RA6W1

Arm-Cortex-M33-based Dual Band Wi-Fi 6 MCU

The RA6W1 is a highly integrated ultra-low-power Wi-Fi microcontroller (MCU) integrating an Arm® Cortex®-M33 system processor with a dual band 802.11a/b/g/n/ax Wi-Fi subsystem, on-chip memory, flexible peripheral interfaces, and power management features. This provides a standalone single chip solution which can support complex low-power IoT solutions that require Wi-Fi network connectivity.

Extremely low-power operation is accomplished by dynamically disabling elements of the MCU which are not in use, thus allowing a near zero level of power consumption when not actively transmitting or receiving data. Such low-power operation can extend the battery life up to a year or more depending on the application.

Built from the ground up for the Internet of Things (IoT), the RA6W1 is ideal for door locks, thermostats, sensors, pet trackers, asset trackers, sprinkler systems, connected lighting, video cameras, video doorbells, wearables, and other IoT devices.

Key Features

- **Arm® Cortex®-M33 Core**
  - Armv8-M architecture
  - Maximum operating frequency: 160 MHz
  - Arm Memory Protection Unit (Arm MPU)
    - Protected Memory System Architecture (PMSAv8)
    - MPU: eight regions
  - SysTick timer
- **Full Networking OS and TCP/IP stack**
  - Comprehensive networking software stack
  - Provide TCP/IP stack: in the form of network socket APIs
- **Wi-Fi processor**
  - Operating modes: Station, Soft AP
  - Bluetooth® coexistence
  - WPS-PIN/PBC for easy Wi-Fi provisioning
  - Connection manager for autonomous and fast Wi-Fi connections
  - IEEE 802.11b/g/n/ax, 1×1, 20 MHz channel bandwidth, 2.4 GHz
  - IEEE 802.11a/n/ac/ax, 1×1, 20 MHz channel bandwidth, 5 GHz
  - IEEE 802.11s Wi-Fi mesh
  - On-chip PA, LNA, and RF switch
  - Wi-Fi security: WPA/WPA2-Enterprise/Personal, WPA2 SI, WPA3 SAE, and OWE
  - Vendor EAP types: EAP-TTLS/MSCHAPv2, PEAPv0/EAP-MSCHAPv2, PEAPv1, EAP-FAST, and EAP-TLS
- **Advanced security and encryption**
  - Secure Crypto Engine
  - Symmetric algorithms: AES, DES/3DES, CHACHA
  - Hash/HMAC: SHA1/224/256
  - Asymmetric algorithms: RSA, DH, ECC
  - TRNG
  - Secure boot
  - Secure debug (SWD)
  - Secure asset storage
  - Device lifecycle management
  - TLS/DTLS protocol acceleration
- **Wi-Fi Alliance certifications**
  - Wi-Fi CERTIFIED™ b, g, n
  - WPA™ – Enterprise, Personal
  - WPA2™ – Enterprise, Personal
  - WPA3™ – Enterprise, Personal
  - Wi-Fi Enhanced Open™
  - WMM
  - WMM – Power Save
  - Wi-Fi Protected Setup™
- **Interfaces**
  - SPI Master/Slave interface
  - I2C Master/Slave interface
  - I2S for digital audio streaming
  - PDM/Digital Mic for digital audio streaming
  - General PWM Timer 32-bit × 8
  - Multiplexed GPIO × 28
  - ADC (12-bit SAR) for sensor interfaces × 4
  - eMMC/SD expanded memory
  - SDIO Host/Slave function
  - OQSPI with encrypted XIP for external code/data Flash
  - QSPI for additional PSRAM storage

- UART × 3
- **Power management unit**
  - On-Chip RTC
  - Wake-up control of fast booting or full booting with minimal initialization time
  - Integrated DCDC and LDOs
  - Support for ultra-low-power Sleep modes
- **Clock source**
  - 40 MHz crystal (± 20 ppm) for master clock
  - 32.768 kHz crystal (± 250 ppm) for RTC clock
  - Integrated 32 kHz RC oscillator
- **Memory**
  - SRAM: 704 kB
  - Retention Memory: 64 kB
  - ROM: 256 kB
  - OTP: 2 kB
- **Supply**
  - Single operating voltage: 1.9 V to 3.6 V (typical: 3.3 V)
  - Digital I/O Supply Voltage: 1.8 V/3.3 V
  - Blackout and brownout detector
- **Package type**
  - 5.6 mm × 6.4 mm, 0.4 mm pitch, 66-pin, FCQFN
  - 3.471 mm × 4.065 mm, 0.424 mm pitch, 70-pin, WLCSP
- **Operating temperature range**
  - -40 °C to 85 °C

Applications

- **Home Appliances**
  - Smart air-conditioners
  - Smart dishwashers
  - Smart oven
  - Printers
  - Washing machines
  - Refrigerators
  - Vacuum cleaners
  - Dryers
  - Coffee makers
  - Air purifiers
- **Home Automation**
  - Smoke detectors
  - Smart thermostats
  - Smart cameras
  - Consumer and DIY video
  - Digital voice assistant
  - Smart plugs
  - Smart door locks
  - Smart video doorbells
  - Generic IoT sensors
  - Light control
  - Smart light bulbs
  - Garage door openers
  - Motion sensors
  - Air quality sensors
  - Smart blinds
- **Wearables**
  - Smart watches
  - Smart rings
  - Smart bracelets
  - Smart bands
  - Elderly tracking devices
  - Pet trackers
  - Child location monitors
- **Industrial**
  - Electronic shelf labels
  - Building security access control
  - Connected solar panels
  - Smart EV chargers
  - Agricultural sensors and monitors
  - Environmental sensors: humidity, CO/CO2, methane, and fine particles
  - Point of sales (POS) solutions
  - Retail location trackers
  - Robotics
  - Factory automation devices
  - Industrial wireless sensors
  - Smart building
  - Energy management devices
  - Smart smoke detectors
  - Garage door openers
  - Educational toys
  - Wi-Fi enabled toys
  - Baby monitors
  - Internet music players
  - Live streaming devices

Contents

Figures

Tables

1. Terms and Definitions

ADC	Analog to Digital Converter
AES	Advanced Encryption Standard
API	Application Programming Interface
Bluetooth LE	Bluetooth® Low Energy
CBC	Cipher Block Chaining
ChaCha	ChaCha20 Stream Cipher Algorithm
CRC	Cyclic Redundancy Check
CRG	Clock and Reset Generator
DAC	Digital-to-Analog Converter
DAI	Digital Audio Interface
DES	Data Encryption Standard
DMA	Direct Memory Access
ECC	Elliptic Curve Cryptography
EAP	Extensible Authentication Protocol
ECB	Electronic Codebook
FPU	Floating Point Unit
GPIO	General Purpose Input/Output
HMAC	Hash-based Message Authentication Code
I2C	Inter-Integrated Circuit
I2S	Inter-IC Sound
IoT	Internet of Things
JTAG	Joint Test Action Group
LDO	Low-Dropout Regulator
LLI	Linked-List Item
LNA	Low Noise Amplifier
LSB	Least Significant Bit
MTB	Micro Trace Buffer
NVIC	Nested Vectored Interrupt Controller
NVRAM	Non-Volatile RAM
OTP	One Time Programmable
PA	Power Amplifier
PDM	Pulse-density Modulation
PLL	Phase-Locked Loop
PPA	Programmable Pin Assignment
PRNG	Pseudo Random Number Generator
PWM	Pulse Width Modulation
PSRAM	Pseudo Static Random Access Memory
QSPI	Quad-Lane SPI
RoT	Root of Trust
RTC	Real-Time Clock
RWC	Read Wait Control
MRC	Maximum Ratio Combining
MSL	Moisture Sensitivity Level
RSA	Rivest-Shamir-Adleman Public Key Crypto System

RTM	Retention Memory
SAR ADC	Successive Approximation Analog-to-Digital Converter
SCB	System Control Block
SHA	Secure Hash Algorithm
SoC	System on Chip
SPI	Serial Peripheral Interface
SRC	Sampler Rate Converter
SRAM	Static Random-Access Memory
SWD	Serial Wire Debug
TAP	Test Access Port
TLS	Transport Layer Security
TRNG	True Random Number Generator
UART	Universal Asynchronous Receivers and Transmitter
WFE	Wait for Event
WFI	Wait for Interrupt
WMM	Wi-Fi Multimedia
WPA	Wi-Fi Protected Access
WPS	Wi-Fi Protected Setup
XIP	eXecute in Place

2. Block Diagrams

Figure 1 shows an overview of the hardware features of RA6W1.

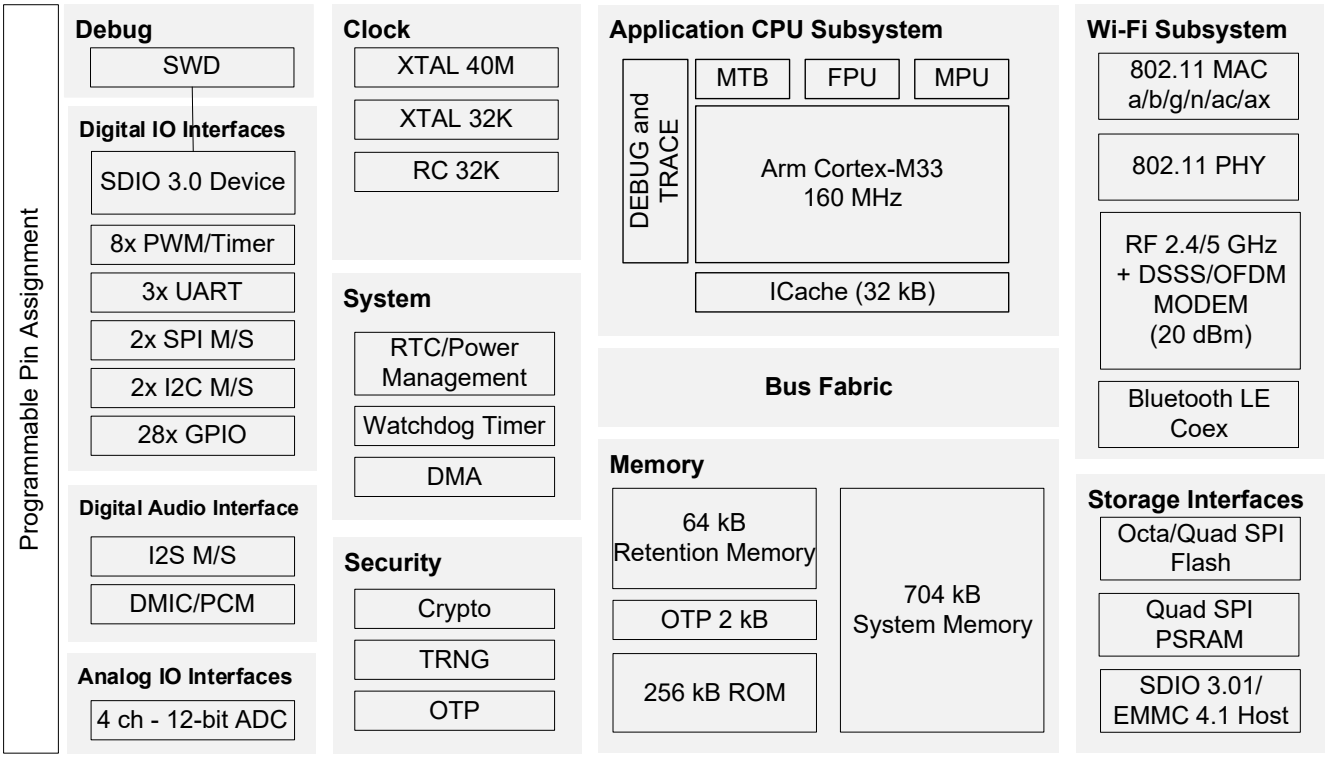


Figure 1. Hardware block diagram

Figure 2 shows an overview of the software features of RA6W1.

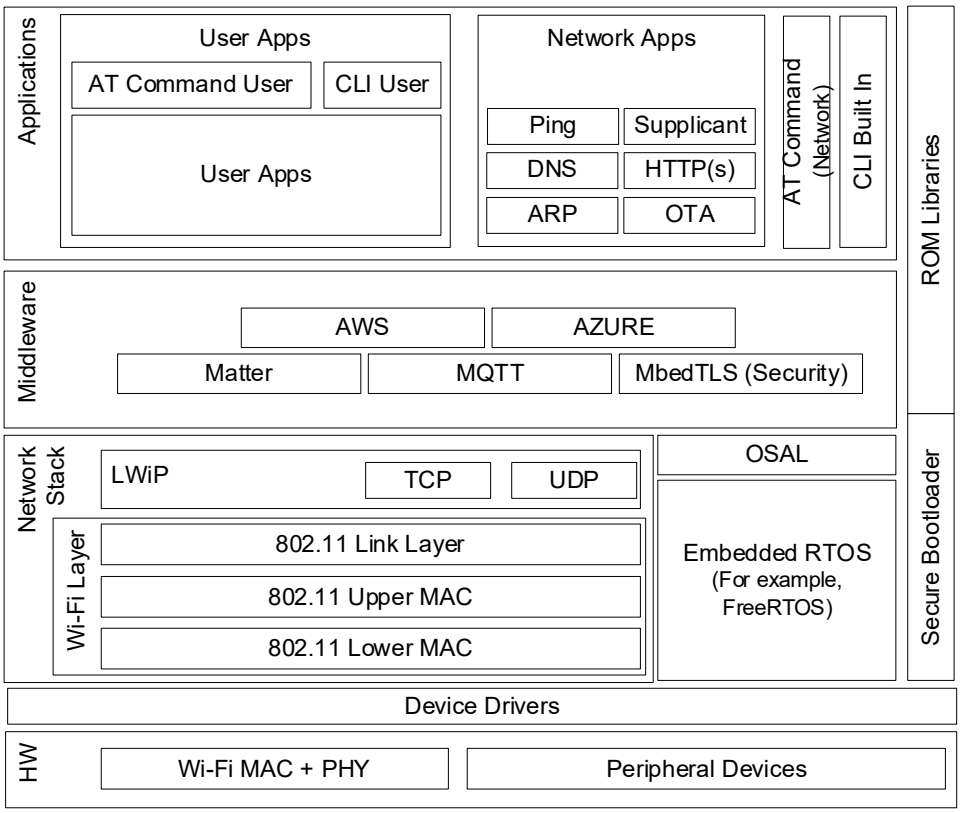


Figure 2. Software system diagram

3. Part Numbering

Figure 3 shows the product part number information, including memory capacity and package type. Table 1 shows a list of products.

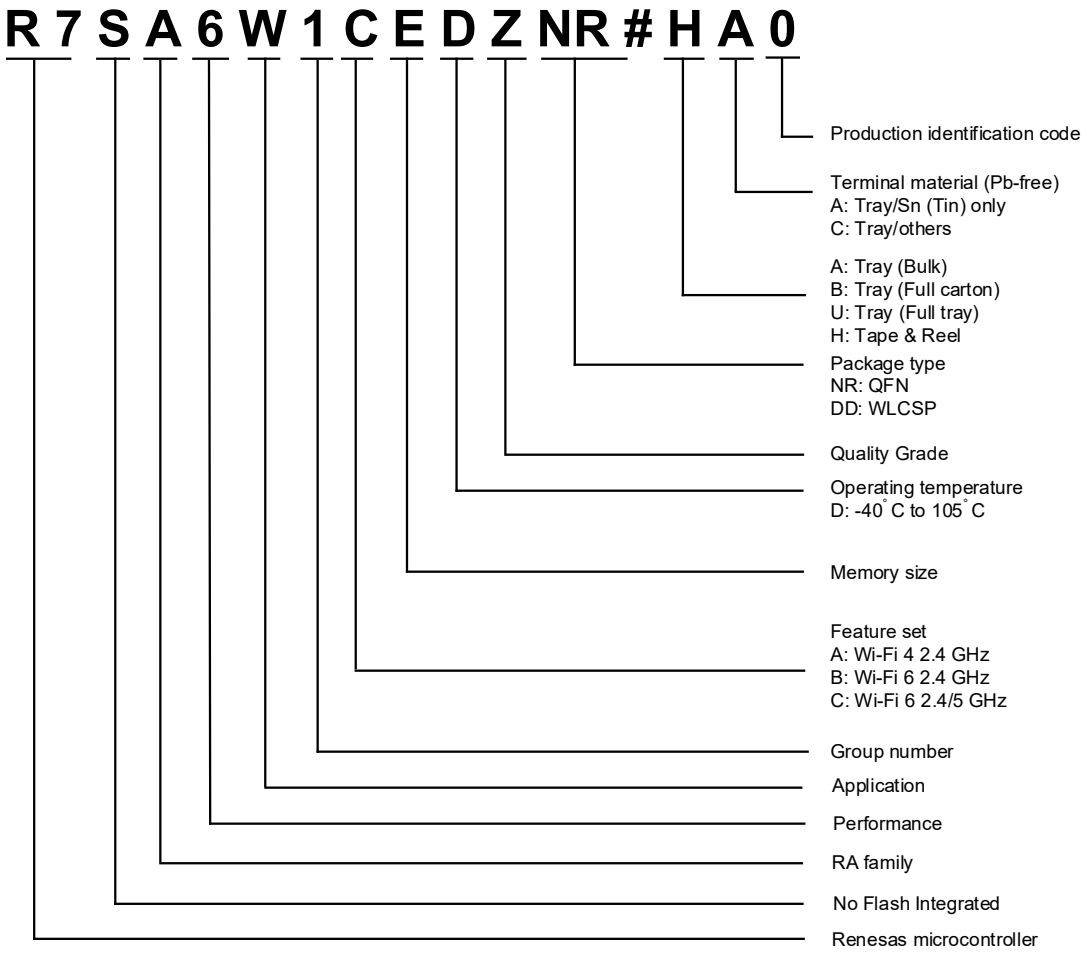


Figure 3. Part numbering scheme

Table 1: Product list

Product part number	Wi-Fi frequency	Wi-Fi version	Package code	SRAM	Operating junction temperature	Shipment form	Pack quantity production
R7SA6W1AEDZNR	2.4 GHz	Wi-Fi 4	FQ0066AA	768 kB	-40 to 105°C	Reel	4000
R7SA6W1BEDZNR	2.4 GHz	Wi-Fi 6					
R7SA6W1CEDZNR	2.4/5 GHz	Wi-Fi 6					
R7SA6W1AEDZDD	2.4 GHz	Wi-Fi 4	WD0070AA				
R7SA6W1BEDZDD	2.4 GHz	Wi-Fi 6					
R7SA6W1CEDZDD	2.4/5 GHz	Wi-Fi 6					

4. Pin Information

The RA6W1 is available in two package variants:

- 66-pin, FCQFN – 5.6 mm × 6.4 mm, 0.4 mm pitch
- 70-pin, WLCSP – 3.471 mm × 4.065 mm, 0.424 mm pitch.

The actual pin/ball assignment is described in the following sections.

4.1 FCQFN Pinout

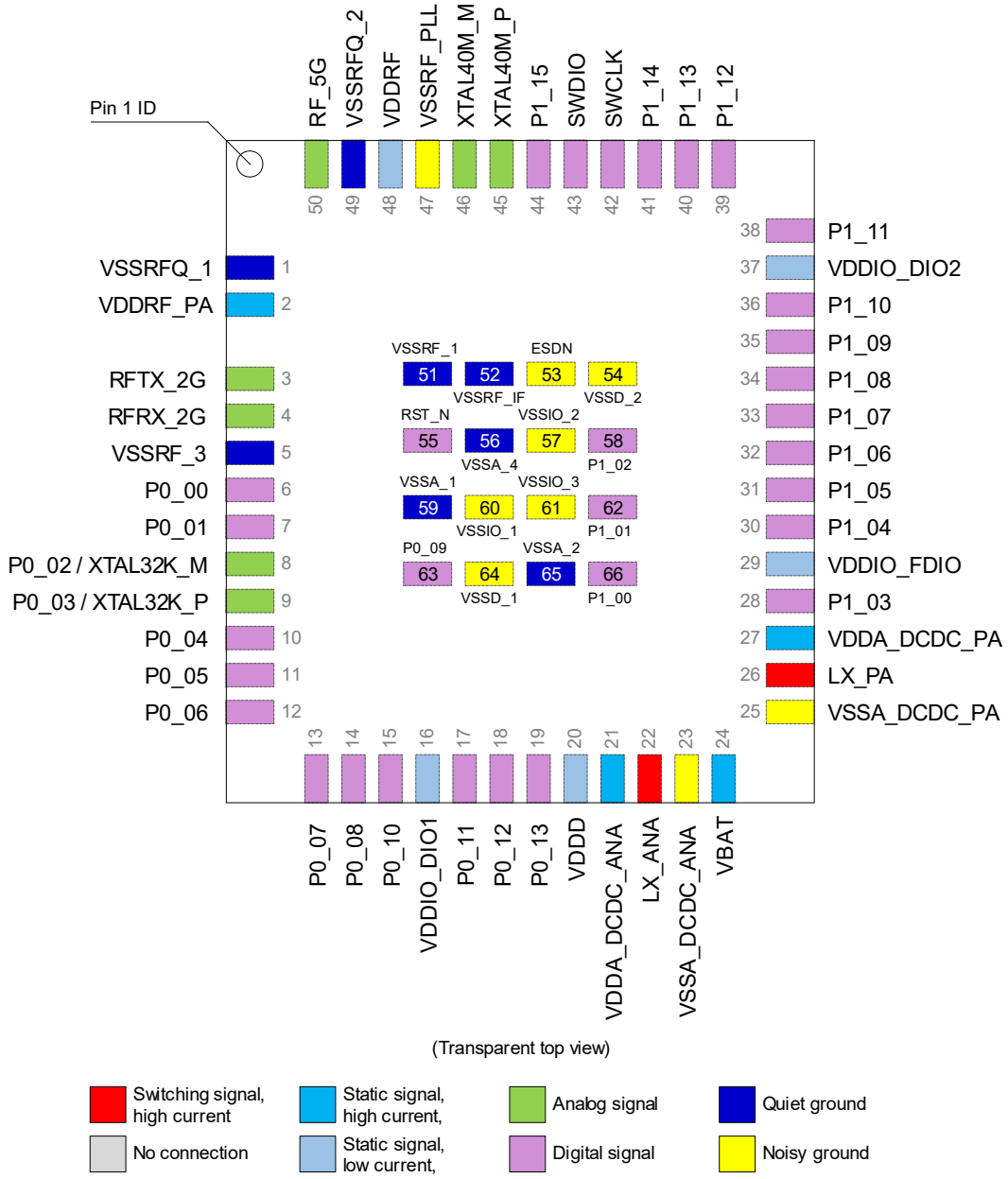


Figure 4. FCQFN66 (top level view)

4.2 WLCSP Pinout

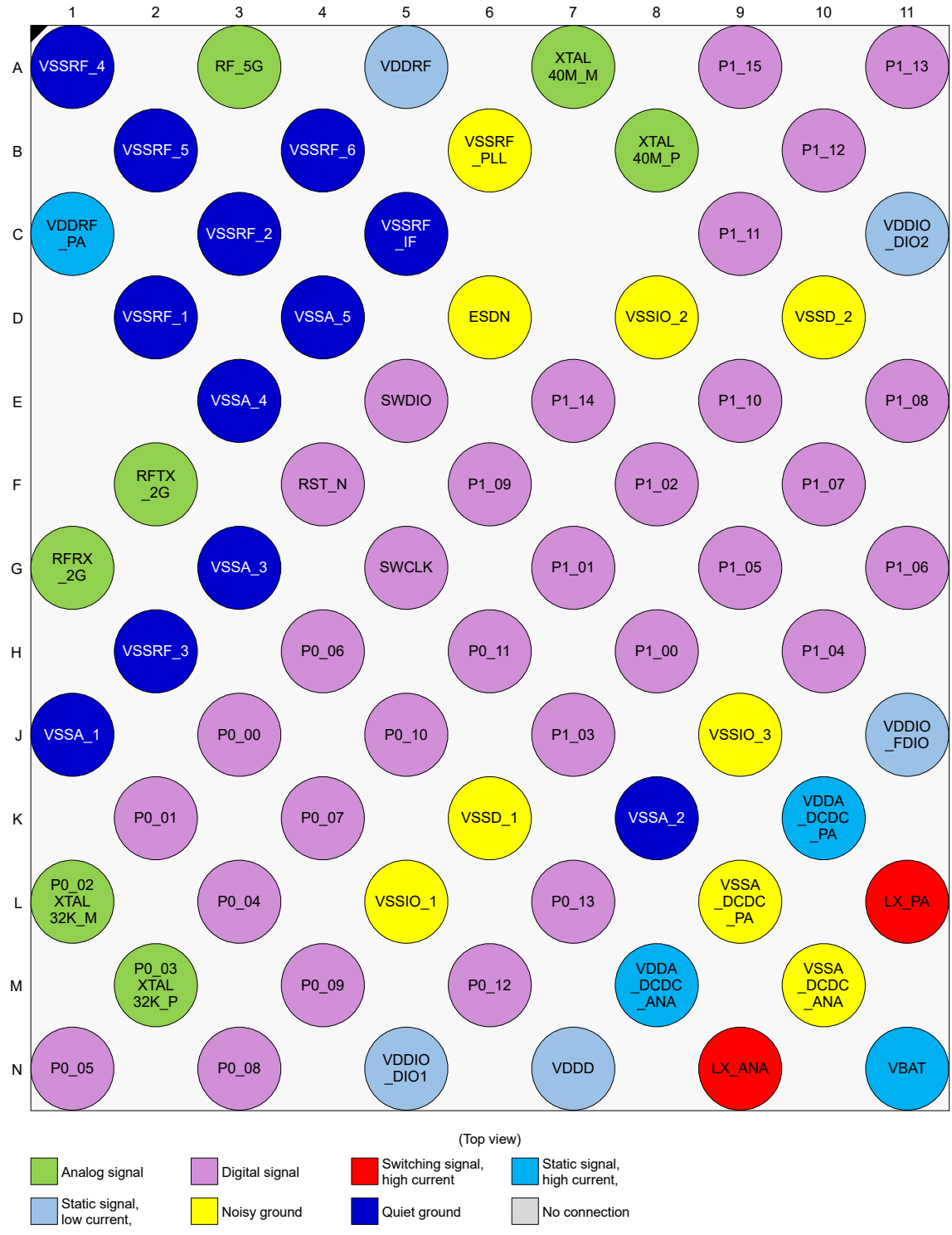


Figure 5. WLCSP70 (top level view)

4.3 Pin Descriptions

Table 2: RA6W1 pin description

FCQFN pin no	WLCSP ball no.	Pin name	Type	Description
Radio				
	A1	VSSRF_4	GND	GND (TX5G)
1		VSSRFQ_1	GND	GND
49		VSSRFQ_2	GND	GND
50	A3	RF_5G	AIO	5G band transmitter output and receiver input, connect to RFSW.
48	A5	VDDRF	AI	rcore power, 1.375 V from DCDC_ANA.
	B2	VSSRF_5	GND	GND (TX5G)
	B4	VSSRF_6	GND	GND (RX5G)
47	B6	VSSRF_PLL	GND	GND (RF_PLL)
2	C1	VDDRF_PA	AI	PA power, 1.35 V from DCDC_PA.
	C3	VSSRF_2	GND	GND (TX5G)
52	C5	VSSRF_IF	GND	GND (RXIF)
51	D2	VSSRF_1	GND	GND (TX2G)
	D4	VSSA_5	GND	GND (rcore)
53	D6	ESDN	GND	GND (RF_DIG)
56	E3	VSSA_4	GND	GND (rcore)
3	F2	RFTX_2G	AIO	2G band transmitter output, connect to inductor.
4	G1	RFRX_2G	AIO	2G band receiver input, connect to inductor.
	G3	VSSA_3	GND	GND (rcore)
5	H2	VSSRF_3	GND	GND (RX2G)
59	J1	VSSA_1	GND	GND (acore)
65	K8	VSSA_2	GND	GND (ana_ldo_buck)
27	K10	VDDA_DCDC_PA	AI	DCDC_PA output: 1.35 V, connect to VDDRF_PA (PA power), external capacitor: 10 µF.
25	L9	VSSA_DCDC_PA	GND	DCDC_PA analog ground.
21	M8	VDDA_DCDC_ANA	AI	DCDC_ANA, connect to VDDRF (1.375 V), external capacitor: 10 µF.
23	M10	VSSA_DCDC_ANA	GND	DCDC_ANA analog ground.
22	N9	LX_ANA	AIO	DCDC_ANA switching: 4.7 µH inductor (recommend: LQM21PN4R7MGH (Murata)).
26	L11	LX_PA	AIO	DCDC_PA switching: 4.7 µH inductor (recommend: LQM21PN4R7MGH (Murata)).
24	N11	VBAT	AI	Battery input: 1.9 ~ 3.6 V.
Oscillator				
46	A7	XTAL40M_M	AIO	40M XTAL oscillator output.
45	B8	XTAL40M_P	AIO	40M XTAL oscillator input.
9	M2	P0_03/XTAL32K_P	AIO	INPUT/OUTPUT with pull-up/down resistor. Analog sharing: XTAL32K_P (VBAT power domain).
8	L1	P0_02/XTAL32K_M	AIO	INPUT/OUTPUT with pull-up/down resistor. Analog sharing: XTAL32K_M (VBAT power domain).
I/O Pins				
44	A9	P1_15	DIOD	INPUT/OUTPUT with pull-up/down resistor. ATB for analog test.
40	A11	P1_13	DIOD	INPUT/OUTPUT with pull-up/down resistor.

FCQFN pin no	WLCSP ball no.	Pin name	Type	Description
39	B10	P1_12	DIOD	INPUT/OUTPUT with pull-up/down resistor.
38	C9	P1_11	DIOD	INPUT/OUTPUT with pull-up/down resistor.
37	C11	VDDIO_DIO2	AI	Digital I/O power: 1.8~3.3 V (typ).
57	D8	VSSIO_2	GND	GND (digital I/O)
54	D10	VSSD_2	GND	GND (dcore)
43	E5	SWDIO	DIOD	INPUT/OUTPUT. Serial Wire Debug data I/O.
41	E7	P1_14	DIOD	INPUT/OUTPUT with pull-up/down resistor. ATB for analog test.
55	F4	RST_N	DIOD	INPUT. Device reset (active LOW). (VBAT power domain).
42	G5	SWCLK	DIOD	INPUT. Serial Wire Debug clock.
12	H4	P0_06	DIOD	INPUT/OUTPUT with pull-up/down resistor. Analog sharing: GPADC input channel 3, ATB for IQ signal test.
17	H6	P0_11	DIOD	INPUT/OUTPUT with pull-up/down resistor.
6	J3	P0_00	DIOD	INPUT/OUTPUT with pull-up/down resistor. Analog sharing (VBAT power domain).
15	J5	P0_10	DIOD	INPUT/OUTPUT with pull-up/down resistor.
61	J9	VSSIO_3	GND	GND (digital I/O)
29	J11	VDDIO_FDIO	AI	Flash I/O power: 1.8 V from FDIO_LDO output, external capacitor 1 µF.
7	K2	P0_01	DIOD	INPUT/OUTPUT with pull-up/down resistor. Analog sharing (VBAT power domain).
13	K4	P0_07	DIOD	INPUT/OUTPUT with pull-up/down resistor. Analog sharing: GPADC input channel 4, ATB for IQ signal test.
64	K6	VSSD_1	GND	GND (dcore)
10	L3	P0_04	DIOD	INPUT/OUTPUT with pull-up/down resistor. Analog sharing: GPADC input channel 1, ATB for IQ signal test.
60	L5	VSSIO_1	GND	GND (digital I/O)
19	L7	P0_13	DIOD	INPUT/OUTPUT with pull-up/down resistor.
63	M4	P0_09	DIOD	INPUT/OUTPUT with pull-up/down resistor.
18	M6	P0_12	DIOD	INPUT/OUTPUT with pull-up/down resistor.
11	N1	P0_05	DIOD	INPUT/OUTPUT with pull-up/down resistor. Analog sharing: GPADC input channel 2, ATB for IQ signal test.
14	N3	P0_08	DIOD	INPUT/OUTPUT with pull-up/down resistor.
16	N5	VDDIO_DIO1	AI	Digital I/O power: 1.8~3.3 V (typ).
20	N7	VDDD	AI	dcore power: 1.1 V from DIG_LDO output, external capacitor 470 nF.
36	E9	P1_10	DIOD	INPUT/OUTPUT with pull-up/down resistor.
QSPI				
35	F6	P1_09	DIOD	INPUT/OUTPUT with pull-up/down resistor.
58	F8	P1_02	DIOD	INPUT/OUTPUT with pull-up/down resistor.
33	F10	P1_07	DIOD	INPUT/OUTPUT with pull-up/down resistor.
62	G7	P1_01	DIOD	INPUT/OUTPUT with pull-up/down resistor.
31	G9	P1_05	DIOD	INPUT/OUTPUT with pull-up/down resistor.
32	G11	P1_06	DIOD	INPUT/OUTPUT with pull-up/down resistor.
66	H8	P1_00	DIOD	INPUT/OUTPUT with pull-up/down resistor.
30	H10	P1_04	DIOD	INPUT/OUTPUT with pull-up/down resistor
28	J7	P1_03	DIOD	INPUT/OUTPUT with pull-up/down resistor.
34	E11	P1_08	DIOD	INPUT/OUTPUT with pull-up/down resistor.

5. Specifications

All Min/Max specification limits are guaranteed by design, production testing and/or statistical characterization. Typical values are based on characterization results at default measurement conditions and are informative only. Default measurement conditions (unless otherwise specified): VBAT1 = VBAT2 = 3.3 V, TA = 25 ˚C. All radio measurements are performed with standard RF measurement equipment providing a source/load impedance of 50 Ω.

The specified Min and Max capacitor values define the range of the effective capacitance, which may vary over the applied voltage because of voltage derating. See the component manufacturer for the capacitor specifications.

5.1 Absolute Maximum Ratings

Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress ratings only, so functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specification are not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

⚠	**CAUTION**
Do not operate at or near the maximum ratings listed for extended periods of time. Exposure to such conditions can adversely impact product reliability and result in failures not covered by the warranty.	

Table 3: Absolute maximum ratings

Parameter	Description	Conditions	Min	Max	Unit
V_{ESD_CDM}	Electrostatic discharge voltage (Charged Device Model)		-500	500	V
V_{ESD_HBM}	Electrostatic discharge voltage (Human Body Model)		-2000	2000	V
V_{BAT_LIM}	Limiting battery supply voltage		-0.2	3.7	V
$V_{PIN\_LIM\_VDDIO\_DIO1}$	Limiting voltage on a pin		-0.2	3.7	V
$V_{PIN\_LIM\_VDDIO\_DIO2}$	Limiting voltage on a pin		-0.2	3.7	V
$V_{PIN\_LIM\_VDDIO\_FDIO}$	Limiting voltage on a pin		-0.2	3.7	V
$V_{PIN\_LIM\_VDDD}$	Limiting voltage on a pin		-0.1	1.22	V
$V_{PIN\_LIM\_VDDRF}$	Limiting voltage on a pin		-0.1	1.55	V
$V_{PIN\_LIM\_VDDRF\_PA}$	Limiting voltage on a pin		-0.1	1.55	V

5.2 Recommended Operating Conditions

Table 4: Recommended operating conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
T_A	Ambient temperature		-40	25	85	°C

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{PIN\_VDDIO\_DIO1}$	Voltage on a pin		1.62		3.6	V
$V_{PIN\_VDDIO\_DIO2}$	Voltage on a pin		1.62		3.6	V
$V_{PIN\_VDDIO\_FIO}$	Voltage on a pin		1.62		3.6	V
V_{PIN_VDDD}	Voltage on a pin			1.1		V
V_{PIN_VDDRF}	Voltage on a pin			1.37		V
$V_{PIN\_VDDRF\_PA}$	Voltage on a pin			1.35		V
V_{BAT}	Battery supply voltage		1.9	3.3	3.6	V

5.3 DC Characteristics

Table 5: DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_OFF\_SLEEP1}$	Off-state battery supply current			0.3		µA
$I_{BAT\_OFF\_SLEEP2}$	Off-state battery supply current			2.5		µA
$I_{BAT\_OFF\_SLEEP3}$	Off-state battery supply current			3.7		µA
$I_{BAT\_OFF\_SLEEP4}$	Off-state battery supply current			200		µA
$I_{BAT\_OFF\_SLEEP5}$	Off-state battery supply current			500 Note 1		µA

Note 1 You can find Sleep mode definitions in the System Overview section.

5.4 Crystal Oscillator 40 MHz – Recommended Operating Conditions

Table 6: Crystal Oscillator 40 MHz - Recommended operating conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{IN\_XTAL40M}$	Input voltage			1.1		V
$f_{XTAL\_XTAL40M}$	Crystal oscillator frequency			40		MHz
$\Delta f_{XTAL\_XTAL40M}$	Crystal frequency tolerance (including aging)		-20		20	ppm
$ESR_{XTAL40M}$	Equivalent series resistance				50	Ω
$C_{L\_XTAL40M}$	Load capacitance		6	8	10	pF

5.5 XTAL32K – Recommended Operating Conditions

Table 7: XTAL32K - Recommended operating conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
$f_{CLK\_EXT\_XTAL32K}$	External clock frequency			32.768		kHz
$\Delta f_{XTAL\_XTAL32K}$	Crystal frequency tolerance (including aging)		-250		250	ppm
$ESR_{XTAL32K}$	Equivalent series resistance			100		kΩ
$C_{L\_XTAL32K}$	Load capacitance			10		pF
$C_{0\_XTAL32K}$	Shunt capacitance			15		pF

5.6 GPADC – DC Characteristics

Table 8: GPADC - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
V_{IN_GPADC}	Input voltage	VDD = 3.3/1.1 V; TA = 25 °C	0		1.4	V

5.7 GPADC – Electrical Performance

Table 9: GPADC - Electrical performance

Parameter	Description	Conditions	Min	Typ	Max	Unit
SNR_{GPADC}	Signal to noise ratio	VDD = 3.3/1.1 V; TA = 25 °C		55.5		dB
THD_{GPADC}	Total harmonic distortion	VDD = 3.3/1.1 V; TA = 25 °C		65.3		dB
$SFDR_{GPADC}$	Spurious-free dynamic range	VDD = 3.3/1.1 V; TA = 25 °C		67.7		dB

5.8 RST_N Digital I/O – Recommended Operating Conditions

Table 10: RST_N Digital IO - Recommended operating conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{BAT\_RST\_N}$	Battery supply voltage		1.62		3.6	V
$V_{IH\_RST\_N\_VBAT\_1V8}$	High-level input voltage		1.13			V
$V_{IL\_RST\_N\_VBAT\_1V8}$	Low-level input voltage				0.92	V
$V_{IH\_RST\_N\_VBAT\_3V3}$	High-level input voltage		2.2			V
$V_{IL\_RST\_N\_VBAT\_3V3}$	Low-level input voltage				1.8	V

5.9 GPIO – Recommended Operating Conditions

Table 11: GPIO - Recommended operating conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
V_{BAT_GPIO}	Battery supply voltage		1.62		3.6	V
$V_{IH\_GPIO\_1V8}$	High-level input voltage		1.26		1.8	V
$V_{IL\_GPIO\_1V8}$	Low-level input voltage		0		0.54	V
$V_{IH\_GPIO\_3V3}$	High-level input voltage		2		3.3	V
$V_{IL\_GPIO\_3V3}$	Low-level input voltage		0		0.8	V

5.10 GPIO – DC Characteristics

Table 12: GPIO - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{OH\_GPIO\_1V8}$	High-level output voltage		1.2		1.8	V
$V_{OL\_GPIO\_1V8}$	Low-level output voltage		0		0.4	V
$R_{PU\_GPIO\_1V8}$	Pull-up resistance		10		60	kΩ
$R_{PD\_GPIO\_1V8}$	Pull-down resistance		10		60	kΩ
$V_{OH\_GPIO\_3V3}$	High-level output voltage		2.4		3.3	V
$V_{OL\_GPIO\_3V3}$	Low-level output voltage		0		0.4	V
$R_{PU\_GPIO\_3V3}$	Pull-up resistance		10		60	kΩ
$R_{PD\_GPIO\_3V3}$	Pull-down resistance		10		60	kΩ
$I_{IH\_GPIO\_3V3}$	High-level input current		-10	0.1	10	nA
$I_{IL\_GPIO\_3V3}$	Low-level input current		-10	-0.1	10	nA
$I_{IH\_GPIO\_1V8}$	High-level input current		-10	0.1	10	nA
$I_{IL\_GPIO\_1V8}$	Low-level input current		-10	-0.1	10	nA

5.11 Radio

5.11.1 WLCSP Package WLAN Radio Characteristics

Table 13: Radio - 802.11.ax RX (2.4 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX2G\_No\_Signal}$	Battery supply current	RX: No Signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		47 Note 1		mA
$I_{BAT\_RX2G\_11b\_1M}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		50 Note 1		mA

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX2G\_11b\_1M}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		61		mA
$I_{BAT\_RX2G\_11n\_MCS7}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		60		mA
$I_{BAT\_RX2G\_11ax\_MCS9}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		60		mA

Note 1 CPU clock = 40 MHz, Low Current Mode for DTIM.

Table 14: Radio - 802.11.ax RX (2.4 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{SENS\_RX2G\_11b\_1M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11b, 1 Mbps DSSS		-97.3		dBm
$P_{SENS\_RX2G\_11b\_2M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11b, 2 Mbps DSSS		-94.3		dBm
$P_{SENS\_RX2G\_11b\_11M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11b, 11 Mbps CKK		-86.7		dBm
$P_{SENS\_RX2G\_11g\_6M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 6 Mbps OFDM		-89.9		dBm
$P_{SENS\_RX2G\_11g\_9M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 9 Mbps OFDM		-89.4		dBm
$P_{SENS\_RX2G\_11g\_18M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 18 Mbps OFDM		-85.9		dBm
$P_{SENS\_RX2G\_11g\_36M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 36 Mbps OFDM		-79.9		dBm
$P_{SENS\_RX2G\_11g\_54M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 54 Mbps OFDM		-75.3		dBm
$P_{SENS\_RX2G\_11n\_MCS0}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11n, MCS0		-89.7		dBm

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{SENS\_RX2G\_11n\_MCS7}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11n, MCS7		-71.8		dBm
$P_{SENS\_RX2G\_11ax\_MCS9}$	Sensitivity (10% PER for 11ax rates)	802.11ax, MCS9		-64.5		dBm
$ACR_{RX2G\_11b\_11M}$	Wanted signal = 2412 MHz/-70 dBm Interferer signal = 2437 MHz Gmode = from AGC	802.11b, 11 Mbps CKK		43.5		dB
$ACR_{RX2G\_11g\_6M}$	Wanted signal = 2412 MHz/-79 dBm Interferer signal = 2437 MHz Gmode = from AGC	802.11g, 6 Mbps OFDM		38.5		dB
$ACR_{RX2G\_11g\_54M}$	Wanted signal = 2412 MHz/-62 dBm Interferer signal = 2437 MHz Gmode = from AGC	802.11g, 54 Mbps OFDM		24.5		dB
$ACR_{RX2G\_11n\_MCS0}$	Wanted signal = 2412 MHz/-79 dBm Interferer signal for ACR= 2437 MHz Gmode = from AGC	802.11n, MCS0		31		dB
$ACR_{RX2G\_11n\_MCS7}$	Wanted signal = 2412 MHz/-61 dBm Interferer signal for ACR = 2437 MHz Gmode = from AGC	802.11n, MCS7		19.5		dB
$ACR_{RX2G\_11ax\_MCS9}$	Wanted signal = 2412 MHz/-54 dBm Interferer signal for ACR = 2432 MHz Gmode = from AGC	802.11ax, MCS9		-3.5		dB

Table 15: Radio - 802.11ax RX (5 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX5G\_No\_Signal}$	Battery supply current	RX: Ch36, No Signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		55 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch36, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		59 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch36, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		67		mA

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX5G\_11ax\_MCS7}$	Battery supply current	RX: Ch36, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		67		mA
$I_{BAT\_RX5G\_No\_Signal}$	Battery supply current	RX: Ch165, No Signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		69 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch165, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		71 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch165, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		68		mA
$I_{BAT\_RX5G\_11ax\_MCS7}$	Battery supply current	RX: Ch165, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		68		mA

Note 1 CPU clock = 40 MHz, Low Current Mode for DTIM.

Table 16: Radio - 802.11ax RX (5 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{SENS\_RX5G\_11a\_6M}$	PSDU length: 1000 octets	802.11a, 6 Mbps OFDM		-89.9		dBm
$P_{SENS\_RX5G\_11a\_9M}$	PSDU length: 1000 octets	802.11a, 9 Mbps OFDM		-89.6		dBm
$P_{SENS\_RX5G\_11a\_18M}$	PSDU length: 1000 octets	802.11a, 18 Mbps OFDM		-86.7		dBm
$P_{SENS\_RX5G\_11a\_36M}$	PSDU length: 1000 octets	802.11a, 36 Mbps OFDM		-80.1		dBm
$P_{SENS\_RX5G\_11a\_54M}$	PSDU length: 1000 octets	802.11a, 54 Mbps OFDM		-73.5		dBm
$P_{SENS\_RX5G\_11n\_MCS0}$	PSDU length 4096 octet; LGI; non-STMC	802.11n, MCS0		-89.7		dBm
$P_{SENS\_RX5G\_11n\_MCS7}$	PSDU length 4096 octet; LGI; non-STMC	802.11n, MCS7		-68.4		dBm
$P_{SENS\_RX5G\_11ac\_MCS7}$	PSDU length: 4096 octet; LGI; non-STMC	802.11ac, MCS7		-68.2		dBm
$P_{SENS\_RX5G\_11ax\_MCS7}$	PSDU length: 4096 octet; LGI; non-STMC	802.11ax, MCS7		-68.3		dBm
$ACR_{RX5G\_11a\_6M}$	Wanted signal = 5500 MHz/-79 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11a, 6 Mbps OFDM		23.5		dB

Parameter	Description	Conditions	Min	Typ	Max	Unit
$ACR_{RX5G\_11a\_54M}$	Wanted signal = 5500 MHz/-62 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11a, 54 Mbps OFDM		8		dB
$ACR_{RX5G\_11n\_MCS0}$	Wanted signal = 5500 MHz/-79 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11n, MCS0		21.4		dB
$ACR_{RX5G\_11n\_MCS7}$	Wanted signal = 5500 MHz/-61 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11n, MCS7		5.4		dB
$ACR_{RX5G\_11ac\_MCS7}$	Wanted signal = 5500 MHz/-61 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11ac, MCS7		5		dB
$ACR_{RX5G\_11ax\_MCS7}$	Wanted signal = 5500 MHz/-54 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11ax, MCS7		4.2		dB

Table 17: Radio - 802.11ax TX (2.4 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_TX\_2G\_11b\_1M}$	Battery supply current measured for EVM and SEM compliance	TX: VBAT = 3.3 V, TA = 25 °C, Ch1, Pout = 17.5 dBm		370		mA
$I_{BAT\_TX\_2G\_11g\_6M}$	Battery supply current measured for EVM and SEM compliance	TX: VBAT = 3.3 V, TA = 25 °C, Ch1, Pout = 16 dBm		285		mA
$I_{BAT\_TX\_2G\_11b\_1M}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 °C, Ch1, Pout = 19 dBm		492		mA
$I_{BAT\_TX\_2G\_11g\_6M}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 °C, Ch1, Pout = 19 dBm		492		mA
$I_{BAT\_TX\_2G\_11n\_MCS7}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 °C, Ch1, Pout = 14.5 dBm		226		mA
$I_{BAT\_TX\_2G\_11ax\_MCS9}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 °C, Ch1, Pout = 10.5 dBm		150		mA

Table 18: Radio - 802.11ax TX (2.4 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{O(MAX)\_11b\_1M}$	Maximum Output Power measured for EVM and SEM compliance	802.11b, 1 Mbps DSSS		16.5		dBm

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{O(MAX)\_11b\_2M}$	Maximum Output Power measured for EVM and SEM compliance	802.11b, 2 Mbps DSSS		16.5		dBm
$P_{O(MAX)\_11b\_5M5}$	Maximum Output Power measured for EVM and SEM compliance	802.11b, 5.5 Mbps CCK		17.5		dBm
$P_{O(MAX)\_11b\_11M}$	Maximum Output Power measured for EVM and SEM compliance	802.11b,11 Mbps CCK		17.5		dBm
$P_{O(MAX)\_11g\_6M}$	Maximum Output Power measured for EVM and SEM compliance	802.11g, 6 Mbps OFDM		17.5		dBm
$P_{O(MAX)\_11g\_54M}$	Maximum Output Power measured for EVM and SEM compliance	802.11g,54 Mbps OFDM		14.5		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS0		15		dBm
$P_{O(MAX)\_11n\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS7		13.5		dBm
$P_{O(MAX)\_11ax\_MCS9}$	Maximum Output Power measured for EVM and SEM compliance	802.11ax, MCS9		10		dBm
$P_{O(MAX)\_11b\_1M}$	Maximum Output Power measured for EVM compliance	802.11b, 1 Mbps DSSS		19.5		dBm
$P_{O(MAX)\_11b\_2M}$	Maximum Output Power measured for EVM compliance	802.11b, 2 Mbps DSSS		19		dBm
$P_{O(MAX)\_11b\_5M5}$	Maximum Output Power measured for EVM compliance	802.11b, 5.5 Mbps CCK		19.5		dBm
$P_{O(MAX)\_11b\_11M}$	Maximum Output Power measured for EVM compliance	802.11b,11 Mbps CCK		19		dBm
$P_{O(MAX)\_11g\_6M}$	Maximum Output Power measured for EVM compliance	802.11g, 6 Mbps OFDM		19		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM compliance	802.11n, MCS0		19.5		dBm

Table 19: Radio - 802.11ax TX (5 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM and SEM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch36, Pout = 16.5 dBm		340		mA
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch36, Pout = 20 dBm		613		mA
$I_{BAT\_TX5G\_11ax\_MCS7}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch36, Pout = 12.5 dBm		224		mA
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM and SEM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch165, Pout = 16.5 dBm		412		mA

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 °C, Ch165, Pout = 19.5 dBm		711		mA
$I_{BAT\_TX5G\_11ax\_MCS7}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 °C, Ch165, Pout = 10.5 dBm		220		mA

Table 20: Radio - 802.11ax TX (5 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{O(MAX)\_11a\_6M}$	Maximum Output Power measured for EVM and SEM compliance	802.11a, 6 Mbps OFDM		17		dBm
$P_{O(MAX)\_11a\_54M}$	Maximum Output Power measured for EVM and SEM compliance	802.11a,54 Mbps OFDM		15		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS0		16.5		dBm
$P_{O(MAX)\_11n\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS7		13		dBm
$P_{O(MAX)\_11ac\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11ac, MCS7		13		dBm
$P_{O(MAX)\_11ax\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11ax, MCS7		13		dBm
$P_{O(MAX)\_11a\_6M}$	Maximum Output Power measured for EVM compliance	802.11a, 6 Mbps OFDM		20.5		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM compliance	802.11n, MCS0		20.5		dBm

5.11.2 FCQFN Package WLAN Radio Characteristics

Table 21: Radio - 802.11.ax RX (2.4 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX2G\_No\_Signal}$	Battery supply current	RX: No Signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		50 Note 1		mA
$I_{BAT\_RX2G\_11b\_1M}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		50 Note 1		mA
$I_{BAT\_RX2G\_11b\_1M}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		61		mA
$I_{BAT\_RX2G\_11n\_MCS7}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		65		mA
$I_{BAT\_RX2G\_11ax\_MCS9}$	Battery supply current	RX: Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		69		mA

Note 1 CPU clock = 40 MHz, Low Current Mode for DTIM.

Table 22: Radio - 802.11.ax RX (2.4 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{SENS\_RX2G\_11b\_1M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11b, 1 Mbps DSSS		-96.9		dBm
$P_{SENS\_RX2G\_11b\_2M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11b, 2 Mbps DSSS		-93.8		dBm
$P_{SENS\_RX2G\_11b\_11M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11b, 11 Mbps CKK		-85.8		dBm
$P_{SENS\_RX2G\_11g\_6M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 6 Mbps OFDM		-88.7		dBm
$P_{SENS\_RX2G\_11g\_9M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 9 Mbps OFDM		-88.2		dBm
$P_{SENS\_RX2G\_11g\_18M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 18 Mbps OFDM		-86.5		dBm
$P_{SENS\_RX2G\_11g\_36M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 36 Mbps OFDM		-80		dBm
$P_{SENS\_RX2G\_11g\_54M}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11g, 54 Mbps OFDM		-73.6		dBm
$P_{SENS\_RX2G\_11n\_MCS0}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11n, MCS0		-89.1		dBm
$P_{SENS\_RX2G\_11n\_MCS7}$	Sensitivity (8% PER for 11b rates, 10% PER for 11g/11n rates)	802.11n, MCS7		-70.5		dBm
$P_{SENS\_RX2G\_11ax\_MCS9}$	Sensitivity (10% PER for 11ax rates)	802.11ax, MCS9		-63.7		dBm
$ACR_{\_RX2G\_11b\_11M}$	Wanted signal = 2412 MHz/-70 dBm Interferer signal = 2437 MHz Gmode = from AGC	802.11b, 11 Mbps CKK		43.5		dB

Parameter	Description	Conditions	Min	Typ	Max	Unit
$ACR_{RX2G\_11g\_6M}$	Wanted signal = 2412 MHz/-79 dBm Interferer signal = 2437 MHz Gmode = from AGC	802.11g, 6 Mbps OFDM		38.5		dB
$ACR_{RX2G\_11g\_54M}$	Wanted signal = 2412 MHz/-62 dBm Interferer signal = 2437 MHz Gmode = from AGC	802.11g, 54 Mbps OFDM		24.5		dB
$ACR_{RX2G\_11n\_MCS0}$	Wanted signal = 2412 MHz/-79 dBm Interferer signal for ACR= 2437 MHz Gmode = from AGC	802.11n, MCS0		31		dB
$ACR_{RX2G\_11n\_MCS7}$	Wanted signal = 2412 MHz/-61 dBm Interferer signal for ACR = 2437 MHz Gmode = from AGC	802.11n, MCS7		19.5		dB
$ACR_{RX2G\_11ax\_MCS9}$	Wanted signal = 2412 MHz/-54 dBm Interferer signal for ACR = 2432 MHz Gmode = from AGC	802.11ax, MCS9		-3.5		dB

Table 23: Radio - 802.11ax RX (5 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX5G\_No\_Signal}$	Battery supply current	RX: Ch36, No Signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		58 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch36, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		58 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch36, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		69		mA
$I_{BAT\_RX5G\_11ax\_MCS7}$	Battery supply current	RX: Ch36, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		72		mA
$I_{BAT\_RX5G\_No\_Signal}$	Battery supply current	RX: Ch165, No Signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		58 Note 1		mA
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch165, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		60 Note 1		mA

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_RX5G\_11a\_6M}$	Battery supply current	RX: Ch165, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		70		mA
$I_{BAT\_RX5G\_11ax\_MCS7}$	Battery supply current	RX: Ch165, Modulated signal input, VBAT = 3.3 V, VDDRF = 1.375 V, TA = 25 °C		72		mA

Note 1 CPU clock = 40 MHz, Low Current Mode for DTIM.

Table 24: Radio - 802.11ax RX (5 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{SENS\_RX5G\_11a\_6M}$	PSDU length: 1000 octets	802.11a, 6 Mbps OFDM		-88.8		dBm
$P_{SENS\_RX5G\_11a\_9M}$	PSDU length: 1000 octets	802.11a, 9 Mbps OFDM		-88.5		dBm
$P_{SENS\_RX5G\_11a\_18M}$	PSDU length: 1000 octets	802.11a, 18 Mbps OFDM		-86		dBm
$P_{SENS\_RX5G\_11a\_36M}$	PSDU length: 1000 octets	802.11a, 36 Mbps OFDM		-79.1		dBm
$P_{SENS\_RX5G\_11a\_54M}$	PSDU length: 1000 octets	802.11a, 54 Mbps OFDM		-72.1		dBm
$P_{SENS\_RX5G\_11n\_MCS0}$	PSDU length: 4096 octet; LGI; non-STMC	802.11n, MCS0		-88.2		dBm
$P_{SENS\_RX5G\_11n\_MCS7}$	PSDU length: 4096 octet; LGI; non-STMC	802.11n, MCS7		-68.3		dBm
$P_{SENS\_RX5G\_11ac\_MCS7}$	PSDU length: 4096 octet; LGI; non-STMC	802.11ac, MCS7		-68.4		dBm
$P_{SENS\_RX5G\_11ax\_MCS7}$	PSDU length: 4096 octet; LGI; non-STMC	802.11ax, MCS7		-66.7		dBm
$ACR_{RX5G\_11a\_6M}$	Wanted signal = 5500 MHz/-79 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11a, 6 Mbps OFDM		23.5		dB
$ACR_{RX5G\_11a\_54M}$	Wanted signal = 5500 MHz/-62 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11a, 54 Mbps OFDM		8		dB
$ACR_{RX5G\_11n\_MCS0}$	Wanted signal = 5500 MHz/-79 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11n, MCS0		21.4		dB

Parameter	Description	Conditions	Min	Typ	Max	Unit
$ACR_{RX5G\_11n\_MCS7}$	Wanted signal = 5500 MHz/-61 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11n, MCS7		5.4		dB
$ACR_{RX5G\_11ac\_MCS7}$	Wanted signal = 5500 MHz/-61 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11ac, MCS7		5		dB
$ACR_{RX5G\_11ax\_MCS7}$	Wanted signal = 5500 MHz/-54 dBm Interferer signal for ACR = 5520 MHz Gmode = from AGC	802.11ax, MCS7		4.2		dB

Table 25: Radio - 802.11ax TX (2.4 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_TX\_2G\_11b\_1M}$	Battery supply current measured for EVM and SEM compliance	TX: VBAT = 3.3 V, TA = 25 ºC, Ch1, Pout = 17.5 dBm		302		mA
$I_{BAT\_TX\_2G\_11g\_6M}$	Battery supply current measured for EVM and SEM compliance	TX: VBAT = 3.3 V, TA = 25 ºC, Ch1, Pout = 15.5 dBm		224		mA
$I_{BAT\_TX\_2G\_11b\_1M}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 ºC, Ch1, Pout = 19.5 dBm		435		mA
$I_{BAT\_TX\_2G\_11g\_6M}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 ºC, Ch1, Pout = 20 dBm		479		mA
$I_{BAT\_TX\_2G\_11n\_MCS7}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 ºC, Ch1, Pout = 14.5 dBm		198		mA
$I_{BAT\_TX\_2G\_11ax\_MCS9}$	Battery supply current measured for EVM compliance	TX: VBAT = 3.3 V, TA = 25 ºC, Ch1, Pout = 10.5 dBm		148		mA

Table 26: Radio - 802.11ax TX (2.4 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{O(MAX)\_11b\_1M}$	Maximum Output Power measured for EVM and SEM compliance	802.11b, 1 Mbps DSSS		18		dBm
$P_{O(MAX)\_11b\_2M}$	Maximum Output Power measured for EVM and SEM compliance	802.11b, 2 Mbps DSSS		18		dBm
$P_{O(MAX)\_11b\_5M5}$	Maximum Output Power measured for EVM and SEM compliance	802.11b, 5.5 Mbps CCK		18		dBm
$P_{O(MAX)\_11b\_11M}$	Maximum Output Power measured for EVM and SEM compliance	802.11b,11 Mbps CCK		18		dBm
$P_{O(MAX)\_11g\_6M}$	Maximum Output Power measured for EVM and SEM compliance	802.11g, 6 Mbps OFDM		15.5		dBm

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{O(MAX)\_11g\_54M}$	Maximum Output Power measured for EVM and SEM compliance	802.11g, 54 Mbps OFDM		15		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS0		14.5		dBm
$P_{O(MAX)\_11n\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS7		14.5		dBm
$P_{O(MAX)\_11ax\_MCS9}$	Maximum Output Power measured for EVM and SEM compliance	802.11ax, MCS9		10.5		dBm
$P_{O(MAX)\_11b\_1M}$	Maximum Output Power measured for EVM compliance	802.11b, 1 Mbps DSSS		20		dBm
$P_{O(MAX)\_11b\_2M}$	Maximum Output Power measured for EVM compliance	802.11b, 2 Mbps DSSS		20		dBm
$P_{O(MAX)\_11b\_5M5}$	Maximum Output Power measured for EVM compliance	802.11b, 5.5 Mbps CCK		20		dBm
$P_{O(MAX)\_11b\_11M}$	Maximum Output Power measured for EVM compliance	802.11b,11 Mbps CCK		20		dBm
$P_{O(MAX)\_11g\_6M}$	Maximum Output Power measured for EVM compliance	802.11g, 6 Mbps OFDM		20		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM compliance	802.11n, MCS0		20.5		dBm

Table 27: Radio - 802.11ax TX (5 GHz) - DC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM and SEM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch36, Pout = 14 dBm		300		mA
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch36, Pout = 19.5 dBm		743		mA
$I_{BAT\_TX5G\_11ax\_MCS7}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch36, Pout = 7.5 dBm		188		mA
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM and SEM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch165, Pout = 13.8 dBm		295		mA
$I_{BAT\_TX5G\_11a\_6M}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch165, Pout = 18.5 dBm		598		mA
$I_{BAT\_TX5G\_11ax\_MCS7}$	Battery supply current measured for EVM compliance	TX: Modulated signal output, VBAT = 3.3 V, TA = 25 ºC, Ch165, Pout = 7 dBm		188		mA

Table 28: Radio - 802.11ax TX (5 GHz) - AC characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{O(MAX)\_11a\_6M}$	Maximum Output Power measured for EVM and SEM compliance	802.11a, 6 Mbps OFDM		14		dBm
$P_{O(MAX)\_11a\_54M}$	Maximum Output Power measured for EVM and SEM compliance	802.11a,54 Mbps OFDM		12		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS0		14		dBm
$P_{O(MAX)\_11n\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11n, MCS7		10.5		dBm
$P_{O(MAX)\_11ac\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11ac, MCS7		9.5		dBm
$P_{O(MAX)\_11ax\_MCS7}$	Maximum Output Power measured for EVM and SEM compliance	802.11ax, MCS7		7.5		dBm
$P_{O(MAX)\_11a\_6M}$	Maximum Output Power measured for EVM compliance	802.11a, 6 Mbps OFDM		18.5		dBm
$P_{O(MAX)\_11n\_MCS0}$	Maximum Output Power measured for EVM compliance	802.11n, MCS0		18.5		dBm

6. System Overview

The RA6W1 contains the following components:

Arm® Cortex®-M33 CPU: The processor is used to run customer applications along with a full Wi-Fi stack (TCP/IP). The M33 provides 1.45 dMIPS/MHz (232 dMIPS at 160 MHz) and has a powerful 32 kB cache controller with 4-way associativity and 16-byte cache line size. Code can be executed from an external Flash device through the secure XIP cache controller, from the internal system or retention memory and from the internal ROM. This gives a high level of flexibility to operate in very low power scenarios.
Depending on the application, the processor can be clocked by the internal PLL configurable from 2.5 MHz up to 160 MHz.

ROM: The internal 256 kB of ROM contains the secure boot loader as well as various library functions such as security and networking which can be used by the firmware to help reduce code size for applications running from internal SRAM.

OTP: The One Time Programmable (OTP) memory is 2 kB in size and consists of two regions. A 256-byte secure region which is used to store sensitive keys used by the crypto subsystem and a 1792-byte general purpose region which can be used by user applications. Various information such as trim values for the clocks and Wi-Fi interface, MAC addresses, and boot configuration scripts are stored in the general-purpose region during production testing. The remaining OTP is available for use by user applications.

System Memory: The system memory is 704 kB of internal SRAM which is primarily used as data memory by the software. It also provides the ability to load and run code for implementing very low power applications. To achieve optimal power consumption, the system memory consists of three memory blocks (two blocks of 256 kB and one block of 192 kB) which can be independently powered off or placed into a low power content retention state. The Cortex M33 can access the system memory at 160 MHz with zero wait cycles.

Retention Memory: The retention memory is a 64 kB block of memory that is retained during the low power operating states of the system. This memory block includes data for maintaining the state of the Wi-Fi interface plus code which can be executed when waking up to perform simple Wi-Fi tasks such as DTIM.

Octa/QSPI Controller: The Octa/QSPI Controller (OQSPIC) interfaces to external Quad or Octa Flash devices which store the applications code and data. The controller provides a zero-overhead secure eXecute In Place (XiP) interface which decrypts the code/data on the fly at up to 80 MHz. This allows for the code/data to be stored securely in Flash and only be accessible internally to the device. The OQSPIC interface supports Flash devices of up to 64 MB.

QSPI Controllers: The QSPI Controller (QSPIC) can interface with an external non secure Flash or PSRAM. For increased read/write performance of the PSRAM, an 8 kB data cache with write-back capabilities can be enabled. The QSPI interface supports Flash or PSRAM devices of up to 64 MB.

Crypto Subsystem: The crypto subsystem provides hardware acceleration of symmetric algorithms (AES 128/192/256, ChaCha20), asymmetric algorithms (RSA, ECC), hash and HMAC algorithms (SHA-1, SHA224/256), and plus support for accelerating specific operations such as key derivation and secure signature processing. There is also a hardware based True Random Number Generator (TRNG) included for secure key generation and support for secure life cycle management.

UART/UART2/UART3: The UART interfaces are asynchronous serial interfaces with hardware flow control. The UART interfaces support both RS-232 and RS-485 physical protocols at data rates up to 5 Mbaud.

SPI/SPI2: The SPI interfaces are serial peripheral interfaces with master and slave capability for connection to SPI devices in master mode or being connected to by a host MCU in slave mode. They have separate RX/TX FIFOs (32 bytes) and support SPI clock rates up to 48 MHz.

I2C/I2C2: The I2C interfaces support master and slave capability with clock stretching and are used for communicating to sensors and/or a host MCU. Each controller includes 32 locations deep FIFO (8-bit RX, 9-bit TX). They support slow, fast, and high-speed modes up to 3.4 Mbps.

Digital Audio Interface (I2S and PDM): The digital audio interface (DAI) consists of both an I2S master/slave interface and a Digital MIC interface supporting PDM for higher quality digital microphone input plus an asynchronous Sampler Rate Converter (SRC) used to convert the input or output data to the required sample rate. A Fractional PLL (FPLL) generates the base frequency for the audio subsystem.
The I2S interface supports either master or slave operation for stereo audio with two 32-bit channels at sample rates of 8 kHz, 12 kHz, 16 kHz, 24 kHz, 32 kHz, 44.1 kHz, and 48 kHz.

The Pulse-density Modulation (PDM) interface supports either master or slave operation for stereo audio sample rates of 24 kHz, 32 kHz, or 48 kHz.

Analog to Digital Converter (ADC): The ADC is a general purpose 4-channel 12-bit analog to digital converter which can be used for voltage measurement or audio sampling. It supports a sampling rate of up to 1 Msps.

General Purpose Timers/Pulse Width Modulators: There are eight general purpose 32-bit timers which support PWM capabilities. These timers all support PWM generation, one capture channel to save a snapshot of the timer, up/down counting with free-running mode, selectable clock source and one-shot pulse generation with configurable width and edge detection counter modes. All timers support One-shot mode and automatic switch from One-shot mode to Counter mode.

Real Time Clock: The RTC is a core feature of the RA6W1 which provides a stable counter that is maintained during low power states such that it can accurately manage the reset and sleep states required to support low power Wi-Fi operation. The RTC can be clocked with a 32.768 kHz crystal or from a calibrated internal 32 kHz RC oscillator. The latter saves component costs. To manage sleep modes, the RTC coordinates the power management of various internal blocks and sequences them as necessary to enter and exit the various low power modes of operation. Wake-up from the sleep modes can be accomplished through a predefined timer event or through external events from various GPIO pins.

Watchdog Timer: The watchdog timer provides a mechanism to protect the system against possible system malfunctions. If there is either a software error or a hardware configuration which disrupts the normal execution of code, the watchdog triggers and performs a reset of the system.

DMA: Two Direct Memory Access (DMA) controllers are included which are designed to improve performance and offload the CPU from moving data through the system. The General Purpose DMA provides 16 channels which can perform memory-memory or memory-peripheral data transfers. The Fast DMA specializes in transferring data between memory and the Wi-Fi interface.

Debug Interface (SWD): A standard Serial Wire Debug (SWD) is provided to allow debugging of user applications during the development phase of a product. When development is complete, the device is set to secure mode and the SWD interface is disabled to protect the device from being tampered with.

GPIO and Programmable Pin Assignment: By default, all digital I/O pins for the RA6W1 are defined as GPIOs to give a total of 28 configurable I/Os. These can be configured by software to be mapped to any of the supported digital peripherals. High performance interfaces such as OQSPIC, QSPIC, SDIO, and eMMC have fixed mapping whereas all other interfaces can be mapped to any pin through the programmable pin assignment (PPA) block.

SD/eMMC Host Controller: The SD/MMC Host Controller supports up to 80 MHz clock output in 1-bit, 4-bit, or 8-bit data bus mode. In 4-bit mode, two SD/SDIO/MMC 4.41 cards can be supported, and one SD card operates at 1.8 V. The SD/MMC Host Controller is compliant with the following features:

- Secure Digital (SD) memory version 3.0 and version 3.01
- Secure Digital I/O (SDIO) version 3.0
- Multimedia Cards (MMC version 4.41, eMMC version 4.5 and version 4.51).

SDIO Device Controller: The SDIO device controller is compliant with version 3.00 of the SD specification. It supports up to 80 MHz operation in 1-bit, 4-bit SD bus mode, and SPI Bus mode. This is used to provide a high bandwidth interface to a host MCU or AP for high-speed communications of commands and Wi-Fi data.

Wi-Fi Subsystem: The Wi-Fi subsystem consists of a 2.4/5 GHz radio, PHY, and MAC supporting the following features:

- Wi-Fi 802.11a/b/g/n/ac/ax in 2.4 GHz and 5 GHz bands
- 802.11a/b/g/n PHY/MAC 1x1 SISO
- Wi-Fi 6 802.11ax PHY/MAC 1x1 SISO
- 20 MHz channels for 2.4 GHz (802.11b/g/n/ax)
- 20 MHz channels for 5 GHz (802.11a/n/ac/ax)
- Compatible to the 802.11ac infrastructure in the 5 GHz band
  - Station mode supports 5 GHz 802.11n mode compatible with 802.11ac
- Support Dynamic Frequency Selection in 20 MHz mode for all 5 GHz DFS bands in all regulatory domains

- Wi-Fi 6 802.11ax IoT features such as TWT
- Adjustable transmit power (1 dB step) where the Access Point (AP) helps set the Station (STA) output power based on signal and noise conditions
- Bluetooth® Coexistence features to coordinate the sharing of the 2.4 GHz band between three devices.

Bluetooth Coexistence: The Bluetooth coexistence interface allows for sharing of the 2.4 GHz band between Wi-Fi and a companion Bluetooth device such that these devices minimize interference with each other. To support smart home systems which may also require coexistence with Zigbee or Thread devices, the coexistence interface provides enhanced features to coordinate the sharing of the 2.4 GHz band between Wi-Fi and two companion devices such as Bluetooth LE and Thread.

Sleep Modes: There are several sleep modes supported by the RA6W1 so that the lowest possible power can be obtained while maintaining the Wi-Fi connection and performing Wi-Fi communications. The low-power modes are defined as follows:

- **Sleep 1 – Deep Sleep/Reset State (~0.3 µA)**
  In this state, power is applied (VBAT), but all internal circuitry is powered off such that when reset is de-asserted, the device can start without waiting for the power to stabilize.
- **Sleep 2 – Low Power Standby (~2.5 µA)**
  In this state, all power domains are off except for the RTC and wake-up pins such that a timer or external event can trigger the device to wake up from sleep.
- **Sleep 3 – Low Power Retention (~3.7 µA)**
  In this state, all power domains are off except for the RTC, wake-up pins, and the retention memory. Retention memory keeps system information such as Wi-Fi state so that a fast wake-up can occur to do simple network management such as DTIM processing. If required, a full system wake-up can also be triggered.
- **Sleep 4/5 – Tickless Idle State (~200 µA/~500 µA)**
  Tickless IDLE state is like retention mode, but it also maintains up to 704 kB of RAM in a low-power state which allows for the full application to maintain its state and wake up without requiring full software reinitialization. The MAC hardware can also be kept alive with a 32 kHz clock to maintain the timers and allow for faster wake-up time depending on power/performance requirements in Sleep 5.

Sensor Monitoring Mode: In Sensor Monitoring mode, the Wi-Fi subsystem is disabled and the Cortex-M33 runs at a very low clock executing code from internal SRAM allowing for a very low power operating mode where simple activities such as monitoring sensors and logging data can be performed.

7. Core System

7.1 Arm® Cortex®-M33

7.1.1 Introduction

The Arm® Cortex®-M33 is a small and highly energy efficient processor that is intended for microcontroller and deeply embedded applications. The processor is based on the Armv8-M architecture and is primarily used in systems where security is an important consideration.

The Cortex-M33 supports low-power operations through the Wait for Event (WFE) or Wait for Interrupt (WFI) functions which stop the CPU clock during times of inactivity. The system can also be placed into Extended Sleep mode by stopping the system clock and clock gating various features including the CPU. The state of the system and CPU are maintained in retention memory during extended sleep and is restored by software upon wake-up.

The register descriptions for NVIC, the System Control Block (SCB), and the System Timer (SysTick) of the Arm Cortex-M33 can be found on the Arm website.

Features

- 160 MHz, 3-stage pipeline
- An in-order issue pipeline
- Armv8, Thumb-2 technology, Little endian
- Floating Point Unit (FPU) supporting single precision arithmetic
  - Combined multiply-add instructions for increased precision (Fused MAC)
  - Hardware support for conversion, addition, subtraction, multiplication with optional accumulation, division, and square root
  - Hardware support for denormal and all IEEE Standard 754-2008 rounding modes
  - 32x32-bit single-precision registers or 16x64-bit double-precision registers
  - Lazy floating-point context save
- DSP/SIMD instructions
  - Single-cycle 16/32-bit MAC
  - Single-cycle dual 16-bit MAC
  - 8/16-bit SIMD arithmetic
- Support for exception-continuable instructions
- Support for SWD and Micro Trace Buffer (MTB)
- Nested Vectored Interrupt Controller (NVIC).

7.1.2 Interrupts

The NVIC is closely integrated with the core to achieve low-latency interrupt processing.

Functions of the NVIC include:

- Configurable levels of interrupt priority from 8 to 256.
- Dynamic reprioritization of interrupts.
- Priority grouping. This enables selection of preempting interrupt levels and non-preempting interrupt levels.
- Support for tail-chaining and late arrival of interrupts. This enables back-to-back interrupt processing without the overhead of state saving and restoration between interrupts.

Table 29: Interrupt list

IRQ #	Name	Description
0	DMA	General Purpose DMA interrupt request.

IRQ #	Name	Description
1	FDMA	Fast DMA interrupt request.
2	UART	UART interrupt request.
3	UART2	UART2 interrupt request.
4	UART3	UART3 interrupt request.
5	I2C	I2C interrupt request.
6	I2C2	I2C2 interrupt request.
7	SPI	SPI interrupt request.
8	SPI2	SPI2 interrupt request.
9	DAI_TX	DAI TX interrupt request.
10	DAI_RX	DAI RX interrupt request.
11	SRC_IN	SRC input interrupt request.
12	SRC_OUT	SRC output interrupt request.
13	SDIO	SDIO interrupt request.
14	SDIO_WKUP	SDEMMC interrupt request.
15	TDES_CBC	TDES interrupt request.
16	PSK_SHA1	PSK_SHA1 interrupt request.
18	CLKCAL	Clock calibration interrupt request.
19	TIMER	TIMER interrupt request.
20	TIMER2	TIMER2 interrupt request.
21	TIMER3	TIMER3 interrupt request.
22	TIMER4	TIMER4 interrupt request.
23	TIMER5	TIMER5 interrupt request.
24	TIMER6	TIMER6 interrupt request.
25	TIMER7	TIMER7 interrupt request.
26	TIMER8	TIMER8 interrupt request.
27	CAPTIMER	GPIO capture interrupt request.
28	SYSPLL_LOCK	PLL480 MHz lock interrupt request.
29	AUX_ADC	AuxADC interrupt request.
30	RTC_IF_EXTWK	External wake-up interrupt request.
31	RTC_IF_BLACK	Voltage blackout interrupt request.
32	RTC_IF_BROWN	Voltage brownout interrupt request.
33	RTC_IF_PCNT	Pulse count interrupt request.
34	RTC_IF_BCF_MSR	Bus clock measure interrupt request.
35	RTC_IF_RTC_ACC	Access to RTC core interrupt request.
36	RTC_IF_EXP	RTC mirror free running count interrupt request.
37	RTC_IF_LMR	FRC to mirroring, loading done interrupt request.
38	MRM	Instruction cache Miss Rate Monitor interrupt request.
39	DCACHE_MRM	Data cache Miss Rate Monitor interrupt request.
40	CC312	CryptoCell-312 interrupt request.
41	GPIO_P0	GPIO port 0 toggle interrupt request.
42	GPIO_P1	GPIO port 1 toggle interrupt request.

IRQ #	Name	Description
43	FPLL_LOCK	FPLL lock interrupt request.
44	WIFI_HSU	Wi-Fi hardware security unit interrupt request.
45	WIFI_MODEM	Wi-Fi modem interrupt request.
46	WIFI_MACTIMER	Wi-Fi MAC TX/RX timer interrupt request.
47	WIFI_MACOTHER	Wi-Fi MAC TX/RX Misc interrupt request.
48	WIFI_MACRX	Wi-Fi MAC RX Trigger interrupt request.
49	WIFI_MACTX	Wi-Fi MAC TX Trigger interrupt request.
50	WIFI_MACPROT	Wi-Fi MAC protocol trigger interrupt request.
51	WIFI_MACINTGEN	Wi-Fi MAC general interrupt request.
52	WIFI_MACBCN	Wi-Fi MAC beacon reception interrupt request.
53	WIFI_RC	Wi-Fi Radio controller interrupt request.
54	SDEMMC	SDEMMC interrupt request.
55	SDEMMC_WKUP	SDEMMC wake-up interrupt request.

7.1.3 Debug

The Cortex M33 provides an SWD interface for real time software debugging.

Features

- Processor halt, single-step, processor core register access, vector catch, unlimited software breakpoints, and full system memory access.
- Hardware breakpoints and watchpoints:
  - A breakpoint unit supporting four to eight instruction comparators.
  - A watchpoint unit supporting two or four data watchpoint comparators.
- MTB provides 8 kB of trace data to support profiling and timestamping of the code execution.

7.2 Internal Memory Architecture

7.2.1 Introduction

The internal memory architecture consists of various memory types including ROM, System RAM, Retention RAM, and OTP.

7.2.2 ROM

ROM is a 256 kB read-only memory that includes the boot loader used to initialize the device and prepare for executing the main firmware. The ROM also includes various library functions for security, DTIM related Wi-Fi MAC drivers/Wi-Fi stack. These are used to reduce the code size of DTIM tasks that may be executing from internal SRAM.

7.2.3 System RAM

System RAM provides 704 kB of internal memory for applications running on the M33 and the Wi-Fi subsystem. The system RAM consists of three blocks which can independently be powered off or put into a low-power retention state to support minimizing power consumption for low-power applications.

Features

- Low latency access
  - 8-way interleaved to minimize latency
- Parallelize data flows from/to various masters in the system:
  - M33, code and data

- DMA
- Wi-Fi subsystem
- SDIO

- Low-power retention mode to minimize power consumption
  - Can be selectively retained during Sleep modes
- Store data or code.

7.2.4 Retention RAM

Retention RAM is a 64 kB block of memory which is used during low-power Sleep modes to maintain the Wi-Fi operating state.

When operating in DPM mode, a small application is also stored in retention memory that is executed directly upon a wake-up event to check if there are any pending Wi-Fi requests. This minimizes wake-up time and reduces the need to power up the full device if there is no activity.

Features

- Low latency access
- Low-power retention mode to minimize power consumption
  - Retained during Sleep modes
- Store data or code.

7.2.5 OTP

The OTP memory is used to store and protect important information essential for mass production and the management of end products, such as boot information, MAC addresses, and serial numbers. It is also used for storing secret information that is used by the advanced security functions like secure boot, secure debug, and secure asset storage. This secret information is programmed during a secure manufacturing process and then locked so that it cannot be accessed directly by CPU read or write operations therefore protecting it from external access.

The OTP memory array supports write accesses of 1 bit and read accesses of 32 bits by executing read/write commands through the OTP controllers register interface.

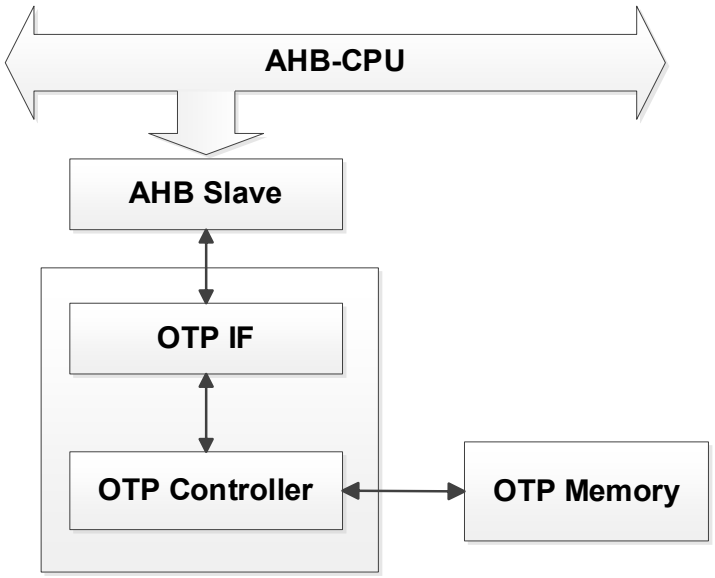


Figure 6. OTP block diagram

The OTP is a 2 kB one-time programmable memory which provides storage for permanent information.

Features

- Storage for device secrets to provide high level of security

  - These device secrets are only accessible by crypto hardware
- Storage for boot and system configuration information
  - MAC addresses
  - Calibration information
  - Configuration script
- Storage for user defined information.

Table 30: OTP memory map

Start index	End index	Size (byte)	Description	Notes
0x000	0x02C	180	Security parameters (lockable)	Hidden Locked for write access
0x02D	0x03F	76	User data (lockable)	Locked for write access
0x040	0x0FF	768	CS script	Configuration parameters (Note 1)
0x100	0x1FF	1024	User data	1 kB user data

Note 1 The configuration parameters consist of both chip manufacturing and customer manufacturing parameters.

The configuration script (CS) is an array of values stored in OTP that is parsed by the boot loader during startup. The values stored in the CS area are either commands which are used to do special configuration or data values such as MAC address, calibration data, or trim values for the Wi-Fi subsystem.

7.2.6 System Address Map

The address map is organized as shown in Table 31.

Table 31: System address map

Map	Region	Start Addr	End Addr	Size (kB)	PD	AMBA	Descriptions
Code	Remapped Devices	0000000	8000000	131072		AHB	
	SYS_RAM	8000000	80B0000	704	PD_SYS_MEM	AHB	MinAddr ~ MinAddr +32 kB*8*2 (8-way interleaved) MinAddr +32 kB*8*2 ~ MaxAddr (2-way interleaved)
	SYS_RAM (SDIO CIS MEM)	80AFC00	80AFE00	0.5	PD_SYS_MEM	AHB	memctrl_apbreg
	SYS_RAM (I-CACHE IVT)	80AFE00	80B0000	0.5	PD_SYS_MEM	AHB	ICACHE remapper
	Retention Memory	8600000	8610000	64	PD_RET	AHB	
	Retention Memory (M33 MTB MEM)	860F000	8610000	4	PD_RET	AHB	M33 MTB base address
	Retention Memory (DEBUG_SECTION)	860FEEC	8610000	0.27	PD_RET	AHB	
	Reserved						
	OQSPIF_C	9000000	A000000	16384	PD_SYS	AHB	
	OQSPIF_M	A000000	E000000	65536	PD_SYS	AHB	
	CACHE_SYS_MON_DATA	E000000	E002000	8	PD_SYS	AHB	
	CACHE_SYS_MON_TAG	E004000	E006000	8	PD_SYS	AHB	
	CACHE_C	E010000	E050000	256	PD_SYS	AHB	

Map	Region	Start Addr	End Addr	Size (kB)	PD	AMBA	Descriptions
	Mask ROM	F020000	F060000	256	PD_SYS	AHB	
Data	SYS_RAM	20000000	200B0000	704	PD_SYS_MEM	AHB	MinAddr ~ MinAddr +32 kB*8*2 (8-way interleaved) MinAddr +32 kB*8*2 ~ MaxAddr (2-way interleaved)
	SYS_RAM (SDIO CIS MEM)	200AFC00	200AFE00	0.5	PD_SYS_MEM	AHB	memctrl_apbreg
	SYS_RAM (I-CACHE IVT)	200AFE00	200B0000	0.5	PD_SYS_MEM	AHB	ICACHE remapper
	DCACHE_C	21000000	21008000	32	PD_SYS_MEM	AHB	
	DCACHE_M	21010000	21018000	32	PD_SYS_MEM	AHB	
	QSPIF2_C	22000000	23000000	16384	PD_SYS	AHB	
	QSPIF2_M	24000000	28000000	65536	PD_SYS	AHB	
	Retention Memory	28600000	28610000	64	PD_RET	AHB	
	Retention Memory (M33 MTB MEM)	2860F000	28610000	4	PD_RET	AHB	M33 MTB base address
	Retention Memory (DEBUG_SECTION)	2860FEEC	28610000	0.26953	PD_RET	AHB	
	Reserved						
	OQSPIF_C	29000000	2A000000	16384	PD_SYS	AHB	
	OQSPIF_M	2A000000	2E000000	65536	PD_SYS	AHB	
	CACHE_C	2E010000	2E050000	256	PD_SYS	AHB	
	AHB_DMA_B (SYSBUS)	2F000000	2F000400	1	PD_SYS	AHB	
Peri.	UART0	40000000	40000100	0.25	PD_SYS	AHB	
	UART1	40001000	40001100	0.25	PD_SYS	AHB	
	UART2	40002000	40002100	0.25	PD_SYS	AHB	
	SRC_FIFO_IF	40003000	40003100	0.25	PD_SYS	AHB	
	Reserved						
	SDIO Device	40010000	40010100	0.25	PD_SYS	AHB	
	SDEMMC Host	40011000	40011100	0.25	PD_SYS	AHB	
	Reserved						
	RTC Interface	40038000	40039000	4	PD_SYS	AHB	
	Reserved						
	VERSION	40070200	40070300	0.25	PD_SYS	APB	
	GPREG	40070300	40070400	0.25	PD_SYS	APB	
	AMBACORE (SYSBUS_ICM)	40070400	40070500	0.25	PD_SYS	APB	
	CRG_SYS	40070500	40070600	0.25	PD_SYS	APB	
	RFMON	40070600	40070700	0.25	PD_SYS	APB	
	DMA	40070700	40070800	0.25	PD_SYS	APB	

Map	Region	Start Addr	End Addr	Size (kB)	PD	AMBA	Descriptions
	KDMA	40070A00	40070B00	0.25	PD_SYS	APB	
	Reserved						
	TIMER/PWM1	40080000	40080100	0.25	PD_SYS	APB	
	TIMER/PWM2	40080100	40080200	0.25	PD_SYS	APB	
	TIMER/PWM3	40080200	40080300	0.25	PD_SYS	APB	
	TIMER/PWM4	40080300	40080400	0.25	PD_SYS	APB	
	TIMER/PWM5	40080400	40080500	0.25	PD_SYS	APB	
	TIMER/PWM6	40080500	40080600	0.25	PD_SYS	APB	
	TIMER/PWM7	40080600	40080700	0.25	PD_SYS	APB	
	TIMER/PWM8	40080700	40080800	0.25	PD_SYS	APB	
	Reserved						
	I2C1	40090000	40090100	0.25	PD_SYS	APB	
	I2C2	40090100	40090200	0.25	PD_SYS	APB	
	SPI1	40090200	40090300	0.25	PD_SYS	APB	
	SPI2	40090300	40090400	0.25	PD_SYS	APB	
	CRG_PER	40090400	40090500	0.25	PD_SYS	APB	
	AuxADC	40090500	40090600	0.25	PD_SYS	APB	
	CLKCAL	40090600	40090700	0.25	PD_SYS	APB	
	Reserved						
	GPIO	400B0000	400B0300	0.75	PD_SYS	APB	P0_04~P0_07: PD_SEN
	MEMCTRL	400B0800	400B0900	0.25	PD_SYS	APB	
	MEMCTRL_RET	400B0900	400B0A00	0.25	PD_SYS	APB	
	Reserved						
	CRG_TOP	400C0000	400C0100	0.25	PD_SYS	APB	
	CRG_COM	400C0200	400C0300	0.25	PD_SYS	APB	
	PLL_DIG (FPLL)	400C0300	400C0400	0.25	PD_SYS	APB	
	Watchdog	400C0700	400C0800	0.25	PD_SYS	APB	
	Reserved						
	CRG_APU	400E0000	400E0100	0.25	PD_SYS	APB	
	APU_AUD	400E0400	400E0500	0.25	PD_SYS	APB	
	APU_DSP	400E0800	400E0900	0.25	PD_SYS	APB	
	DAI	400E0900	400E0A00	0.25	PD_SYS	APB	
	SRC_IF	400E0C00	400E0D00	0.25	PD_SYS	APB	
	Reserved						
Ext.	External RAM	60000000	8FFFFFFF			AHB	(6090000 ~ 60D400FF: reserved)
	External Device	90000000	DFFFFFFF			AHB	

7.3 Clock Generation

7.3.1 Introduction

The Clock Generation block is part of the Clock and Reset Generator (CRG) block. The CRG block provides clocks to the CPU (CM33), bus structure and core clock of several peripherals (I2C, SPI) and the AuxADC. It also generates 240 MHz for the Wi-Fi subsystem, which in turn generates its local derived clocks.

The following clock sources are available to drive the specific subsystems:

- RCX (OSC32): ~ 32 kHz
- XTAL32K: 32.768 kHz external crystal clock source
- XTAL40M: 40 MHz external crystal clock source
- SYS PLL: 480 MHz PLL system clock source
- FPLL: 90.316 ~ 98.304 MHz clock for the digital audio interface.

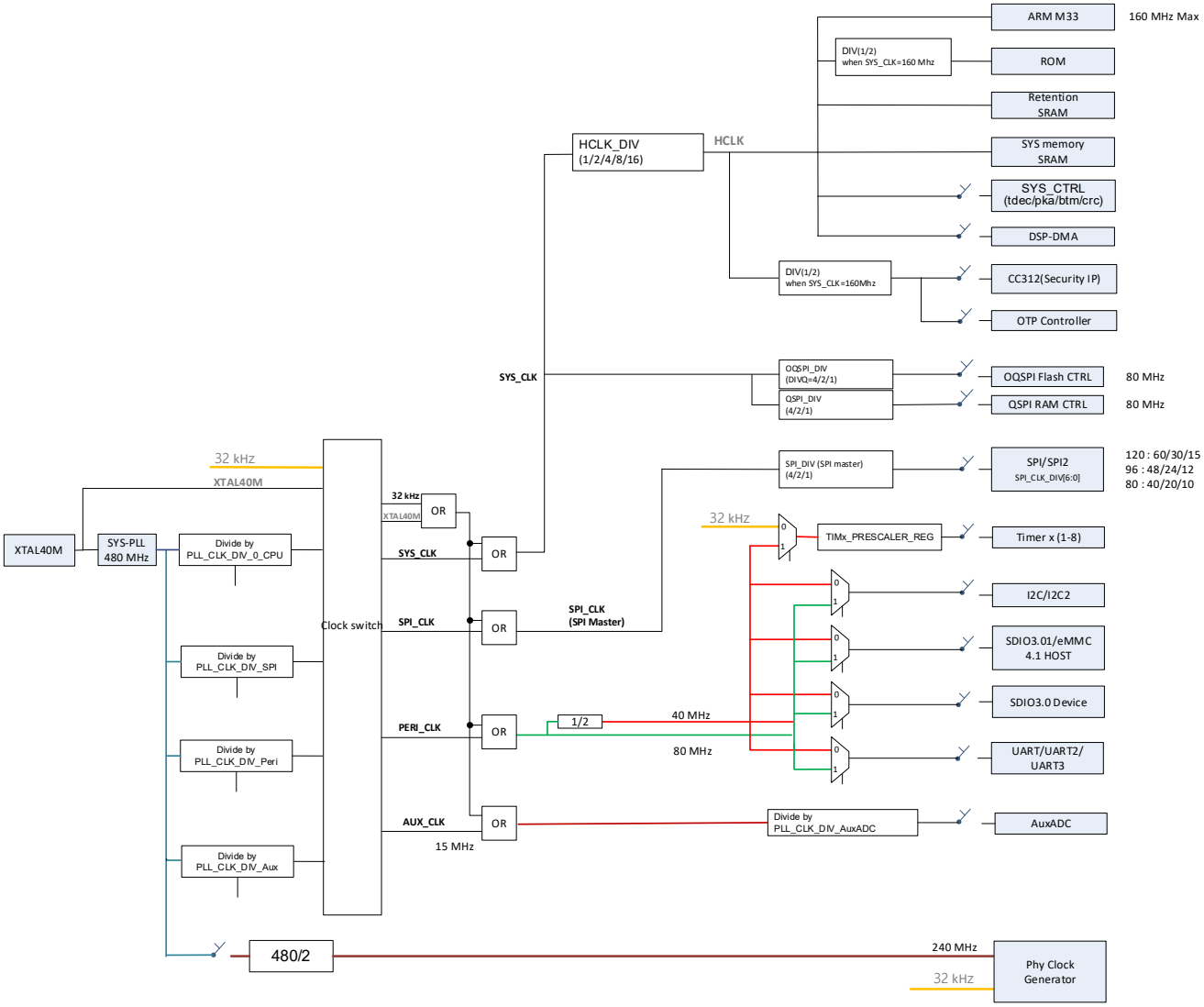


Figure 7. System clocktree diagram

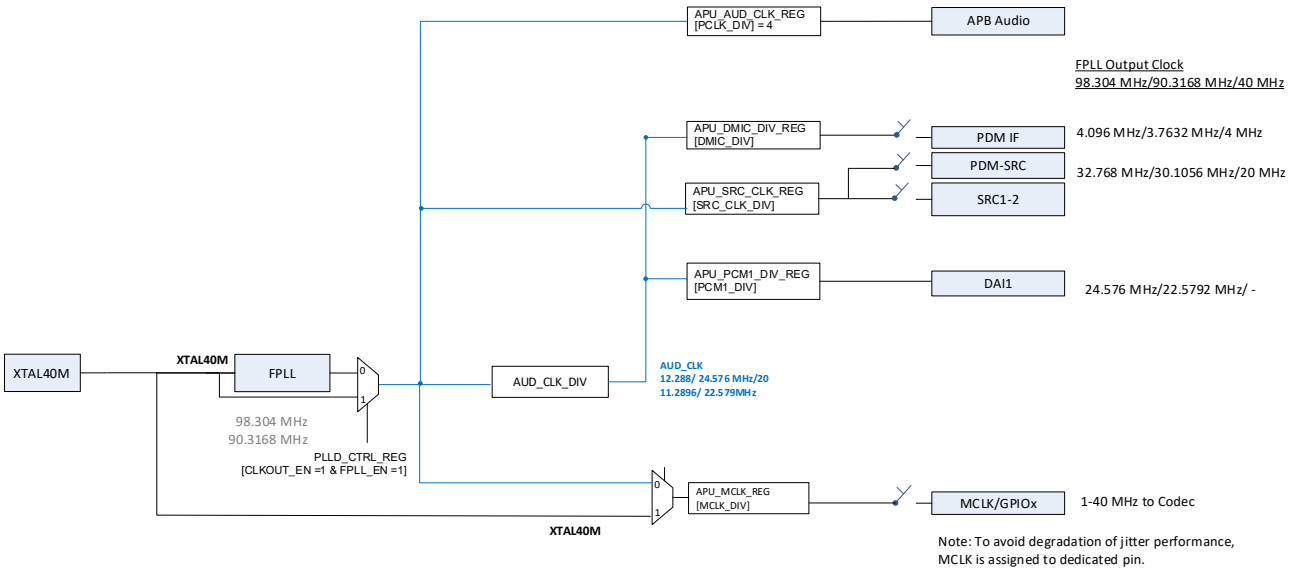


Figure 8. Digital audio interface clocktree diagram

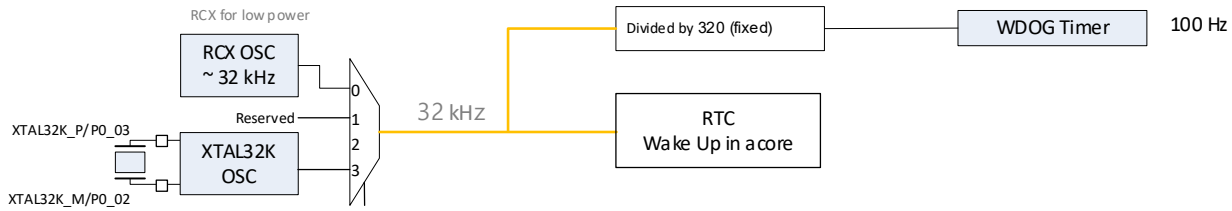


Figure 9. RTC clocktree diagram

7.3.2 System Clock (SYS_CLK, HCLK)

SYS_CLK is the source clock of HCLK divider block and OQSPI and QSPI divider block. It can be 32 kHz, 40 MHz (from the crystal), or a clock divided from the PLL. Table 32 lists the available clock frequencies.

Table 32: Available SYS_CLK frequencies

Clock source	SYS_CLK	HCLK	OQSPI/QSPI
XTAL32K/RC32K	32 kHz	32 kHz	N/A
XTAL40M	40 MHz	SYS_CLK divided by 1,2,4,8,16	SYS_CLK divided by 1,2,4
PLL480	160, 80, 40, 20 MHz	SYS_CLK divided by 1,2,4,8,16	SYS_CLK divided by 1,2,4

7.3.3 Peripheral Clocks (SPI_CLK, PERI_CLK, AUX_CLK)

The peripherals like UART, SDeMMC, SDIO, and SPI require 80 MHz whereas I2C needs 40 MHz. These are derived from the PLL clock. In XTAL mode, the 40 MHz crystal clock is supplied to the peripherals directly. AuxADC is used for internal calibration purposes of circuits such as internal temperature sensor. AUX_CLK is generated for this purpose.

7.3.4 Audio Clocks (AUD_CLK)

Audio clocks are generated from FPLL to cover various specific audio rate requirements such DAI, SRC, DMIC interface, and MCLK for external CODEC support.

7.3.5 RTC Clocks (32 kHz)

RTC clocks can be generated from external crystal (XTAL32K OSC) or internal oscillator (RCX OSC). It is used for watchdog timer and RTC timer.

7.4 Power Management

The RA6W1 has an RTC block which provides power management and function control for low-power operation. In normal operation, the RTC block is always powered on when RST_N is in HIGH state. The RTC block also has a control function for RA6W1 internal power supplying components such as LDOs, DCDCs, and power switches.

7.4.1 Power-On Sequence

Figure 10 shows the sequence after the initial switching from power-off to power-on.

The RST_N of the RA6W1 is an external reset pin that disables the RTC block when the external reset is asserted. When the external reset is released, all the internal regulators are turned on automatically in the sequence predefined by the RTC block. At the initial phase of power-on sequence, RCX 32 kHz is used for RTC clock source.

When the external reset is released, LDOs for both XTAL and digital I/O are turned on shortly and then the DCDC regulator is turned on according to the predefined interval. The enabling intervals can also be modified in the register settings after initial power-up.

The power-on sequence in an initial boot phase is as follows:

1. External reset assertion which makes the bias circuit to be disabled.
2. VBAT_POR asserts a reset to RTC block when external reset case or POR/BOR reset may occur depending on the VBAT voltage.
3. After reset is released from VBAT_POR, the RTC block enables RCX 32 kHz oscillator.
4. The RTC block uses the generated 32 kHz clock as a clock source.
5. The internal counter in RTC generates the enable signal for RCX 32K, DCDC, and FDIO_LDO.
6. FDIO_LDO supplies power for the external flash related blocks, which are used for XiP after boot ROM sequences are completed.
7. DCDC and DIG_LDO generate power for digital core sequentially.
8. After the RTC releases the reset to the digital core, the digital core starts the system boot sequences.

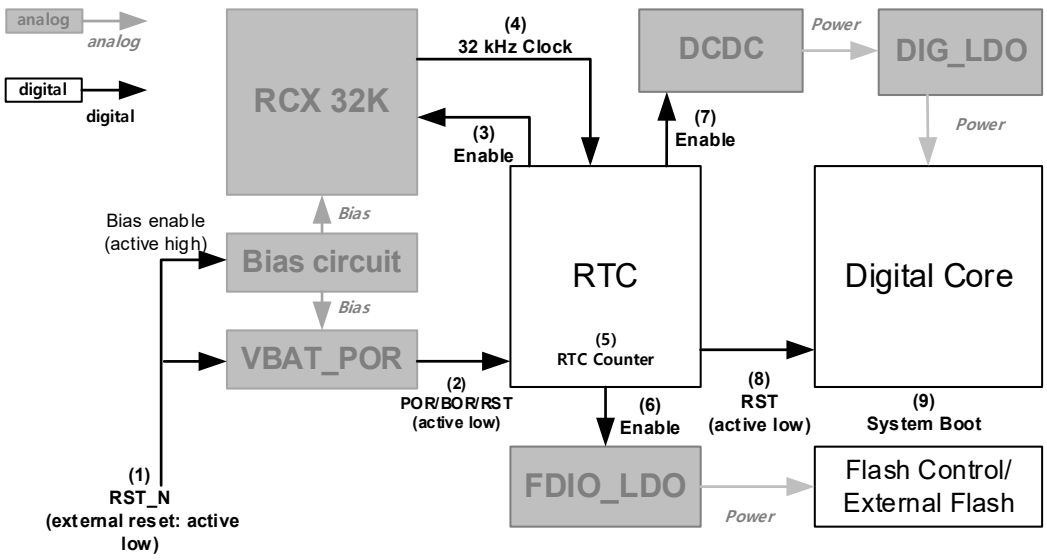


Figure 10. Power-on sequence

7.4.2 Power Management Unit

The RA6W1 has internal DCDC converters and several LDOs to supply power to all internal sub-blocks. Power management does the on-off control of these regulators and is implemented through the register setting inside the RTC block.

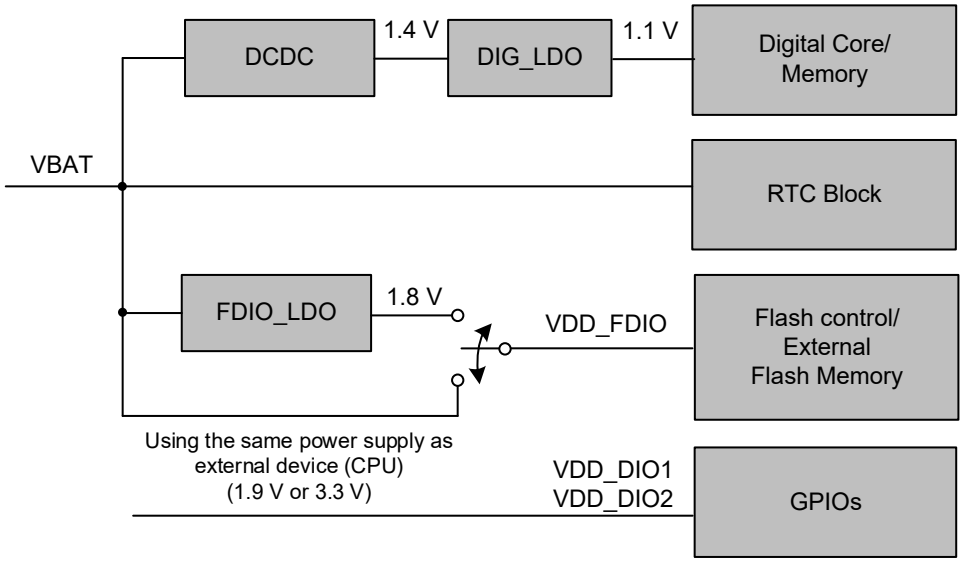


Figure 11. Power management block diagram

Details of the internal DCDC converters and LDOs are explained:

- DCDC converter: From the power supply of external VBAT input, it generates 1.4 V power for the digital LDO.
- DIG_LDO for digital blocks: From the DCDC output, it generates 1.1 V power which is used for digital blocks.
- FDIO_LDO for I/O and external flash memory:
  - This LDO output is used only for 1.8 V digital I/O applications.
  - From external VBAT power input, it generates 1.8 V output voltage which is used for digital I/O power domain in 1.8 V digital I/O applications.
  - It is also used for external flash memory.
  - FDIO_LDO supports only 1.8 V.
  - For 3.3 V digital I/O applications, external power (3.3 V) is directly supplied for digital I/O power.

With the internal DCDC converters and LDOs, all the power necessary for RA6W1 internal sub-blocks is sufficiently generated.

7.5 Sleep Modes

7.5.1 Introduction

There are several Sleep modes supported by the RA6W1 so that the lowest possible power can be obtained while maintaining the Wi-Fi connection and performing Wi-Fi communications.

The low-power modes are defined as follows:

- **Sleep 1 – Deep Sleep/Reset State (~0.3 µA)**
  In this state, power is applied (VBAT), but all internal circuitry is powered off such that when reset is de-asserted, the device can start without waiting for the power to stabilize.
- **Sleep 2 – Low Power Standby (~2.5 µA)**
  In this state, all power domains are off except for the RTC and wake-up pins such that a timer or external event can trigger the device to wake up from sleep.

- **Sleep 3 – Low Power Retention (~3.7 µA)**
  In this state, all power domains are off except for the RTC, wake-up pins, and the retention memory. The retention memory keeps system information such as Wi-Fi state so that a fast wake-up can occur to do simple network management such as DTIM processing. If required, a full system wake-up can also be triggered.
- **Sleep 4/5 – Tickless Idle State (~200 µA/~500 µA)**
  Tickless idle state is like retention mode, but it also maintains up to 704 kB of RAM in a low-power state which allows for the full application to maintain its state and wake up without requiring full software reinitialization. The MAC hardware can also be kept alive with a 32 kHz clock to maintain the timers and allow for faster wake-up time depending on power/performance requirements.

7.5.2 Wake-Up Sources

The RA6W1 supports different Sleep modes to enable the lowest possible application power consumption.

Wake-up from sleep can be triggered by any of the following:

- Program the RTC block to wake up the device at a predefined time interval.
- GPIO trigger (immediate).
- GPIO counter (using the RTC in count mode, waiting to exceed a programmable threshold).
- ADC threshold sensing.

7.5.3 Sleep Mode Active Blocks Overview

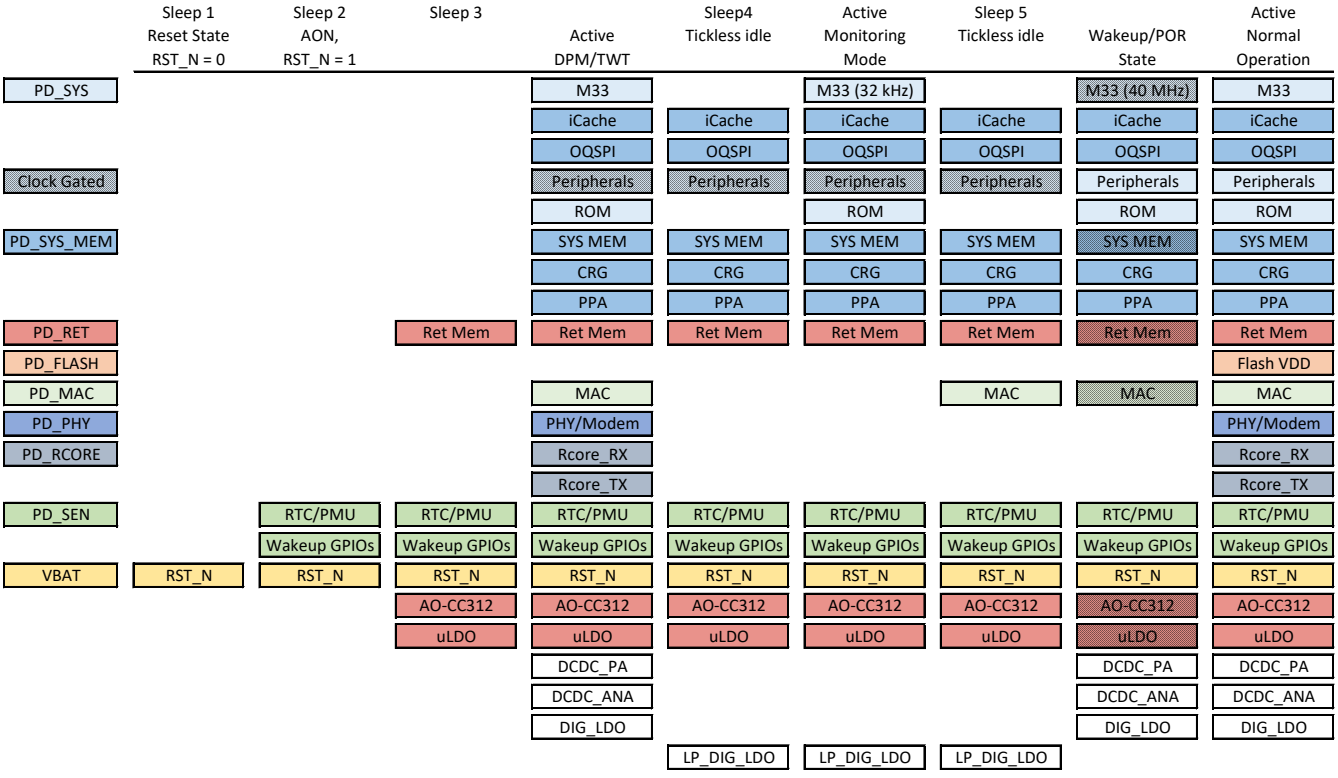


Figure 12. Sleep mode block overview

The blocks which are active during each Sleep mode are shown in Figure 12. Active DPM/TWT, Active Monitoring Mode and Active Normal Operation Mode are application-level examples to help the understanding of the power domain application. The color of each block represents the power domain. Wakeup/POR state explains the state after wake-up from Sleep or POR.

The following descriptions are summaries for each power domain.

- VBAT, DCDC_ANA, DCDC_PA, uLDO, LP_DIG_LDO, DIG_LDO
  VBAT is an external power supply for the overall system. From VBAT, DCDC_ANA, and DCDC_PA generate internal power for analog blocks and RF blocks (Rcore_RX, Rcore_TX). uLDO is connected to VBAT and

supplied for the retention memory and the AO-CC312. This allows the low current in Sleep 3 mode. The LP_DIG_LDO is connected to DCDC_ANA and generates power for Tickless Idle States (Sleep 4/Sleep 5). DIG_LDO supplies power for the most digital block including M33 in Active mode.

- PD_SYS, PD_SYS_MEM, PD_RET

  These power domains are required to execute codes in M33. PD_RET remains in state in Sleep 3. PD_RET domain includes 64 kB retention memory and is used to implement ultra-low-power DTIM.
- PD_FLASH

  This power domain is used to keep power in state for external NVM.
- PD_MAC, PD_PHY, PD_RCORE

  These power domains are required for the communication function of Wi-Fi. In Sleep 4 mode, only PD_MAC power domain is required to be on state.
- PD_SEN

  Excluding Sleep1, other sleep modes require this power domain in state. RTC timer and GPIO wake-up can be supported from PD_SEN domain. To leave Sleep 1 in Sleep 1 mode, use reset.

7.6 DMA

7.6.1 Introduction

There are two DMA controllers available which provide CPU independent transfers of data between peripherals and memory.

The General Purpose DMA (GP-DMA) provides 16 channels which can perform memory-memory or memory-peripheral data transfers.

The Fast DMA is specialized for transferring data from memory-to-memory and between memory and the Wi-Fi interface.

7.6.2 General Purpose DMA

The General-Purpose DMA controller supports data transfers to/from peripherals and memory through an independent DMA bus. It has sixteen channels to support the various peripherals available.

The GP-DMA controller off-loads the processor by performing transfers independent of the processor's execution allowing the processor to perform other operations and be notified by an interrupt after a transfer is completed. The DMA controller has eight levels of priority and is a bus master on the AHB-DMA bus with programmable priority level. The number of peripheral requests is multiplexed on the available channels to increase utilization of the DMA because not all peripherals are active at the same time.

Features

- Programmable source and destination addresses
- Programmable 16-bit transfer length and interrupt generation counters
- Programmable support of 8-bit,16-bit, and 32-bit transfers
- Programmable AHB bursts support (INCR4, INCR8, INCR)
- Request mode (DREQ_MODE) with programmable peripheral selection
  - Support also CIRCULAR, applicable only in DREQ_MODE
- Freeze option, applicable during memory-to-memory transfers as well
- Programmable bus error detection and IRQ generation
- Channel’s start/stop through setting/resetting the DMA_ON bit-field
  - Index and Circular index counters are reset in the next transfer
- Memory protection for general-purpose channels
  - Generate a bus error when the DMA R/W access is not allowed.

Figure 13 shows the mapping of peripherals to DMA channels.

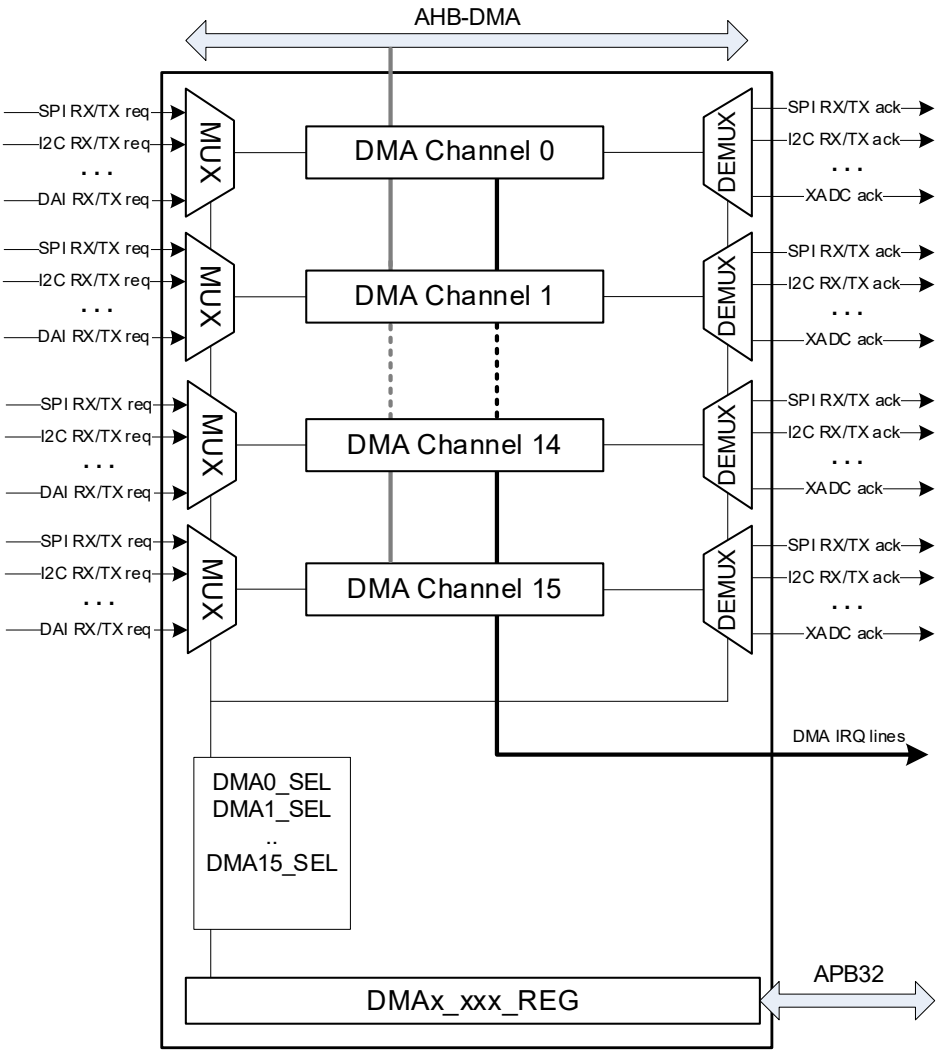


Figure 13. DMA channel mapping

The peripherals that support DMA data transfers are defined in Table 33.

Table 33: List of peripherals that support DMA data transfers

ID	Peripheral
0x0	SPI1_RX (Read)
0x1	SPI1_TX (Write)
0x2	SPI2_RX (Read)
0x3	SPI2_TX (Write)
0x4	UART0_RX (Read)
0x5	UART0_TX (Write)
0x6	UART1_RX (Read)
0x7	UART1_TX (Write)
0x8	UART2_RX (Read)
0x9	UART2_TX (Write)
0xA	I2C1_RX (Read)
0xB	I2C1_TX (Write)
0xC	I2C2_RX (Read)
0xD	I2C2_TX (Write)

ID	Peripheral
0xE	ADC[0]
0xF	ADC[1]
0x10	ADC[2]
0x11	ADC[3]
0x12	SRC_IN
0x13	SRC_OUT
0x14	DAI_TX
0x15	DAI_RX
Not available	Memory Read/Write (Note 1)

Note 1 If a DMA channel is not selected as a peripheral DMA, it can be used as a memory-to-memory DMA.

7.6.2.1 Input/Output Multiplexer

The multiplexing of peripheral requests is controlled by DMA_REQ_MUX_REG. Thus, if DMA_REQ_MUX_REG[DMAxy_SEL] is set to a certain (non-reserved) value, the TX/RX request from the corresponding peripheral should be routed to DMA channels y (TX request) and x (RX request) respectively. Similarly, an acknowledging de-multiplexing mechanism is applied. However, when two or more bit-fields (peripheral selectors) of DMA_REQ_MUX_REG have the same value, the less significant selector is given priority (see also the register's description).

7.6.2.2 DMA Channel Operation

A DMA channel is switched on with bit DMA_ON. This bit is automatically reset if the DMA transfer is finished. The DMA channels can either be triggered by software or by a peripheral DMA request. If DREQ_MODE is 0, then a DMA channel is immediately triggered. If DREQ_MODE is 1, the DMA channel can be triggered by hardware interrupt.

If DMA starts, data is transferred from address DMAx_A_START_REG to address DMAx_B_START_REG for a length of DMAx_LEN_REG, which can be 8-bit, 16-bit, or 32-bit wide. The address increment is realized with an internal 13 bits counter DMAx_IDX_REG, which is set to 0 if the DMA transfer starts and is compared with the DMA_LEN_REG after each transfer. The register value is multiplied according to the AINC, BINC, and BW values before it is added to DMA_B_START_REG and DMA_B_START_REG. AINC or BINC must be 0 for register access.

If at the end of a DMA cycle and the DMA start condition is still true, the DMA continues. The DMA stops if DREQ_MODE is low or if DMAx_LEN_REG is equal to the internal index register. This condition also clears the DMA_ON bit.

If bit CIRCULAR is set to 1, the DMA controller automatically resets the internal index registers and continues from its starting address without intervention of the Arm Cortex-M33. If the DMA controller is started with DREQ_MODE = 0, the DMA should always stop, regardless of the state of CIRCULAR.

Each DMA channel can generate an interrupt if DMAx_INT_REG is equal to DMAx_IDX_REG. After the transfer and before DMAx_IDX_REG is incremented, the interrupt is generated.

For example, if DMA_x_INT_REG = 0 and DMA_x_LEN_REG = 0, there should be one transfer and an interrupt.

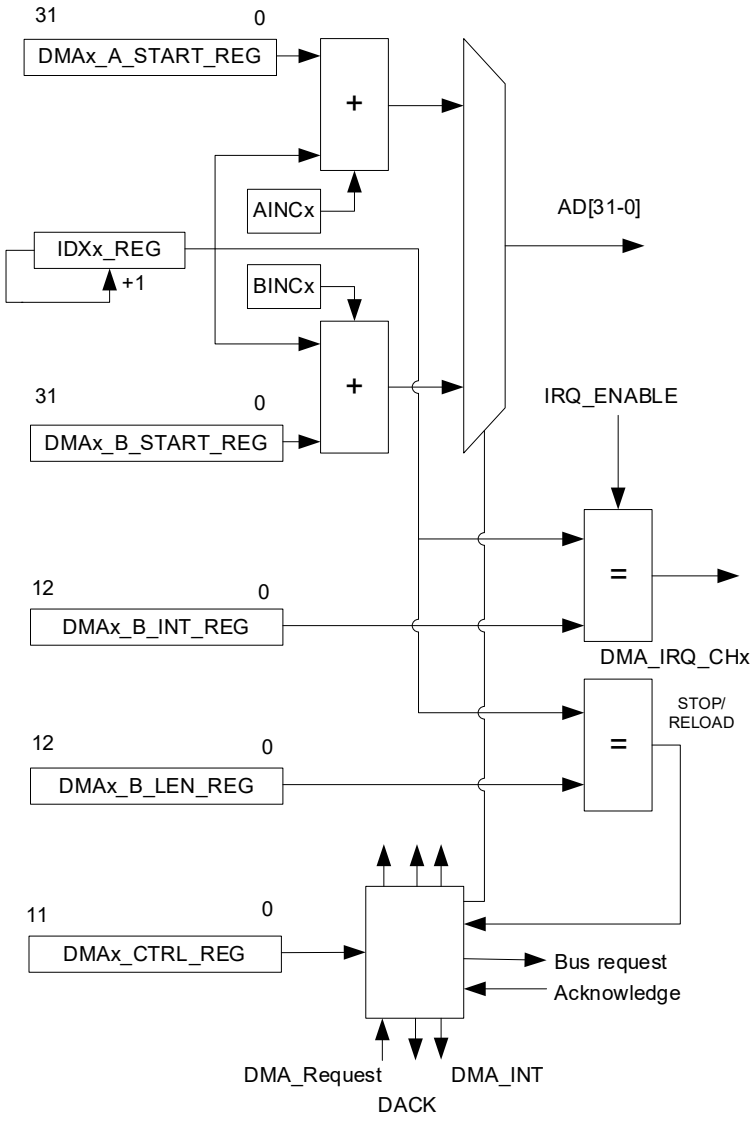


Figure 14. DMA channel diagram

7.6.2.3 DMA Arbitration

The priority level of a DMA channel can be set with bits DMA_PRIO[2-0]. These bits determine which DMA channel is activated in case more than one DMA channel requests DMA. If two or more channels have the same priority, an inherent priority applies (see register description).

With DREQ_MODE = 0, a DMA can be interrupted by a channel with a higher priority if the DMA_IDLE bit is set.

When DMA_INIT is set, however, the DMA channel currently performing the transfer locks the bus and cannot be interrupted by any other channels, until the transfer is completed, regardless of if DMA_IDLE is set. The purpose of DMA_INIT is to initialize a specific memory block with a certain value, fetched also from memory, without any interruption from other active DMA channels that may request the bus at the same time. Consequently, it should be used only for memory initialization, while when the DMA transfers data to/from peripherals, it should be set to 0. Also, AINC must be set to 0 and BINC to 1, when DMA_INIT is enabled.

NOTE
Memory initialization could also be performed without having the DMA_INIT enabled and by simply setting AINC to 0 and BINC to 1, provided that the source address memory value does not change during the transfer. However, it is not guaranteed that the DMA transfer is not interrupted by other channels of higher priority when these request access to the bus at the same time.

7.6.2.4 Freezing DMA Channels

Each channel of the DMA controller can be temporarily disabled by writing a 1 to freeze all channels at SET_FREEZE_REG. To enable the channels again, a 1 to bits at the RESET_FREEZE_REG must be written. There is no hardware protection from erroneous programming of the DMA registers.

The on-going Memory-to-memory transfers (DREQ_MODE = 0) cannot be interrupted. Therefore, in that case, the corresponding DMA channels are frozen after any on-going Memory-to-memory transfer is completed.

7.6.3 Fast DMA

The Fast DMA (F-DMA) controller performs bulk data transfers, reading data from the source address range, and writing the data to the destination address range. The Fast DMA is used for fast data transfers from memory to memory and between memory and the Wi-Fi subsystem.

Features

- Programmable transfer size from 1 byte to 1 MB
- Up to twelve channels can be set at the same time
- Freeze option for each channel
- Support for buffer chaining
- Interrupt enable for each channel.

The basic unit of bus transmission is 32-bit and has a function to automatically correct address aligns, even if the source and destination addresses are not in word units.

For example, assuming that the transfer size is 23 bytes, the source base address is 0x001 for read access, and the destination base address is 0x102 for write access, the number of bytes per transaction is performed as follows:

- Source base address [1:0] = 0x1: the master read port of fast DMA performs read access with the following sequences:
  - 1 > 2 > 4 > 4 > 4 > 4 > 4 bytes
- Destination base address [1:0] = 0x2: the master write port of fast DMA performs write access with the following sequences:
  - 2 > 4 > 4 > 4 > 4 > 4 > 1 bytes

7.7 Hardware Accelerators

7.7.1 CRC Calculation

The cyclic redundancy check (CRC) is an efficient method for detecting corruption in the transmission and storage of data. The CRC calculation consists of an iterative algorithm involving XOR and shifts operations. The CRC calculator is mainly used to check the integrity of the firmware image stored in Flash.

Features

- Operate at a clock frequency up to system clock
- Support 8-bit, 16-bit, and 32-bit data paths
- Be a master on the bus and can operate over a region of memory independent of the CPU
- Support various methods of calculating the CRC as defined by the following generator polynomials:
  - CRC-32:
    $G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$
  - CRC-16 CCITT:
    $G(x) = x^{16} + x^{12} + x^5 + 1$
  - CRC-16 IBM:
    $G(x) = x^{16} + x^{15} + x^2 + 1$

7.7.2 Pseudo Random Number Generation

The Pseudo Random Number Generator (PRNG) is used to generate a stream of pseudo random numbers.

Features

- Operation clock frequency is the same as the system clock
- Support partial parallel processing of 8-bit, 16-bit, and 32-bit units.

Given a seed value from the TRNG, the PRNG generates a stream of random values based on a generator polynomial defined as follows:

$$G(x) = x^{31} + x^{28} + 1$$

If you set the same configuration in the PRNG after POR boot (not Software reset), the PRNG always generates a deterministic sequence which is the same pattern. The PRNG uses SEED input to avoid generating a predictable next pattern. However, this SEED is designed to accumulate the current calculation without resetting the poly calculation, to make its pattern different from the previous pattern. Additionally, the PRNG is designed not to be reset at runtime to prevent predictable patterns within the system as much as possible.

7.8 Watchdog Timer

7.8.1 Introduction

The watchdog timer provides a mechanism to detect if the software executing on the Cortex-M33 has halted because of a software issue or a system fault. This allows the system to recover either through a non-maskable interrupt to the Cortex-M33 or through a system reset.

Features

- A 13-bit down counter clocked at 100 Hz
- Clock sources: RC32K or XTAL32
- Generate:
  - NMI to Cortex-M33 if counter reaches 0
  - Hardware reset if counter reaches -16
- Protected by a lock bit to avoid accidental freeze
- Automatically froze when Cortex-M33 is in debug mode.

The 13-bits System Watchdog is supplied by the always on power domain (PD_AON) and is automatically enabled as soon as the system powers up, see Figure 15. It is decremented by one every 10 ms, clocked by RC32K/XTAL32K. The timer value can be accessed through WATCHDOG_REG that is set to its max value at reset which is a maximum watchdog time-out of 81 seconds.

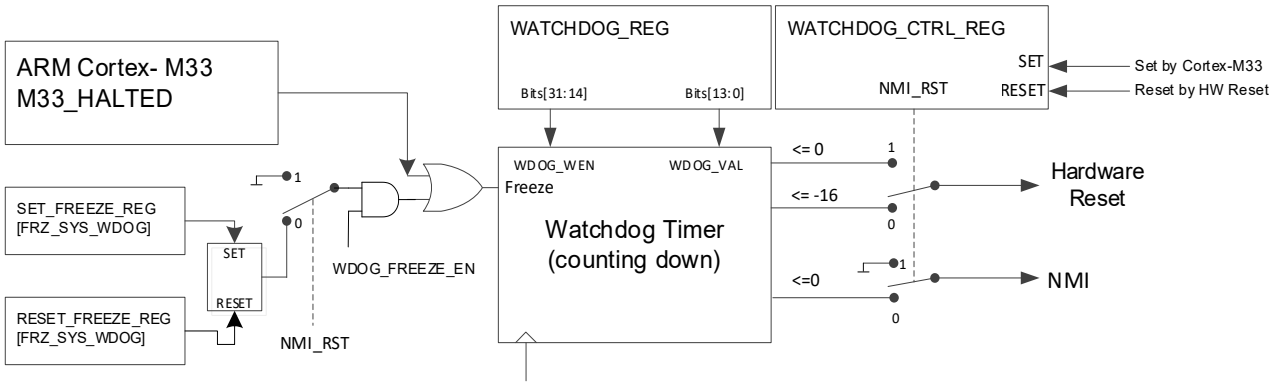


Figure 15. System watchdog block diagram

The watchdog timer can be configured to either generate an interrupt or system reset when a timeout occurs.

If the watchdog timer is configured to generate an interrupt, it generates an NMI to the Cortex-M33 when the watchdog timer reaches 0 and a hardware reset when the counter becomes less than or equal to -16. The NMI

handler must write any value > -16 to the WATCHDOG_REG to prevent the generation of a WDOG reset within 16x10 ms = 160 ms.

If the watchdog timer is configured to reset the system, it generates a system reset if the timer becomes less than or equal to 0.

The system watchdog can be frozen by either the Cortex-M33 directly by writing to a specific register or frozen automatically when the debugger is attached, and the Cortex-M33 CPU is halted during debugging. Even if the watchdog is frozen by the Cortex-M33, the watchdog resumes operation automatically when entering one of the Sleep modes where PD_SYS is turned off.

7.9 Brownout and Blackout Detection

The device enters a brownout or blackout condition whenever the input voltage dips below $V_{brownout}$ or $V_{blackout}$ (see Table 34). This condition must be considered during the design of the power supply routing, especially for battery operated applications. High current operations such as Wi-Fi TX may cause a dip in the supply voltage, potentially triggering a Brownout. When designing a battery-operated system, the overall circuit resistance must be considered which includes the internal resistance of the battery, the contact resistance of the battery holder (for example, four contacts for two AA batteries), the wiring resistance, and the PCB routing resistance.

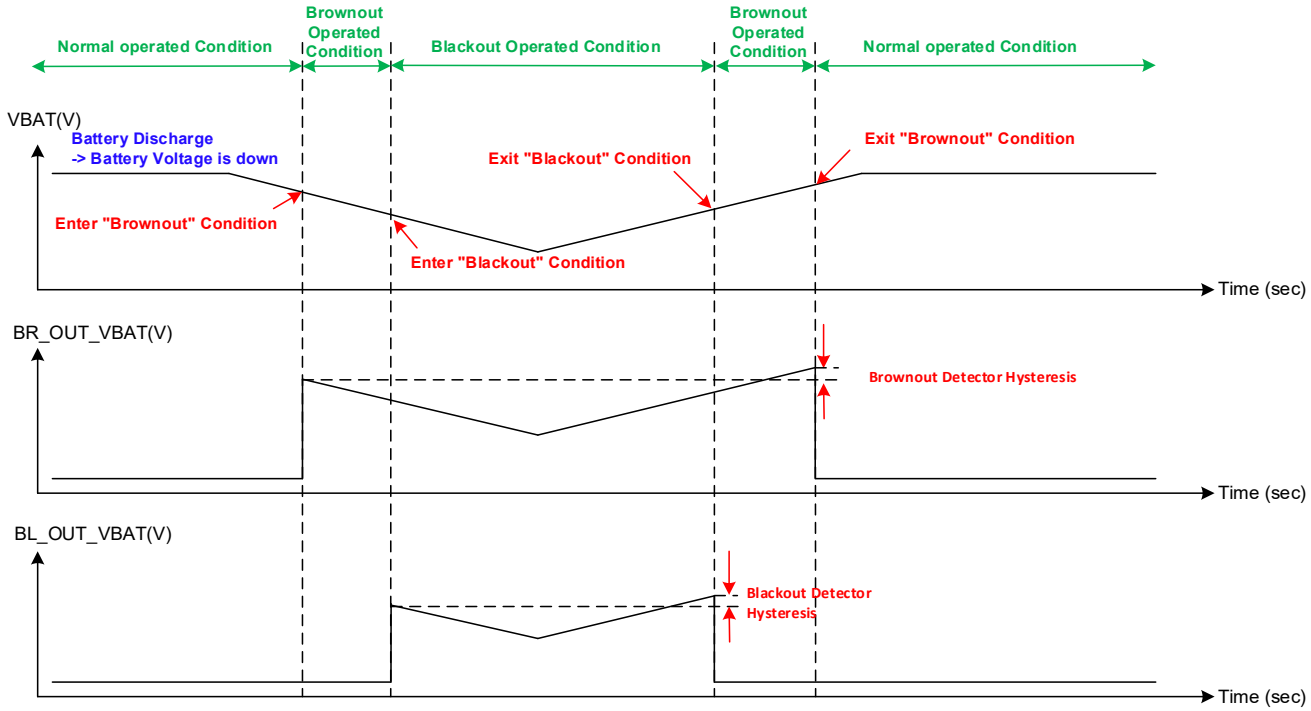


Figure 16. Brownout and blackout levels

Brownout and blackout conditions are monitored only when in normal operating mode (not during low-power Sleep modes). The blackout and brownout condition have hysteresis, and the blackout condition can be used for low battery level indication. The brownout condition can be used as an exit condition for low battery level state.

When blackout events occur, the proper handling is required for the low battery situation. After that, a brownout event occurs, recovering to the normal state is required.

When a brownout or blackout condition is entered, an interrupt can be generated for either case. The voltage level which triggers a brownout or blackout condition is configurable as shown in Table 34.

There is also a hardware POR, which monitors VBAT and asserts overall system reset. It detects 1.68 V rising level and 1.55 V falling level. Figure 17 shows the VBAT voltage level monitoring hardware. VBAT_POR generates a hardware reset and Blackout/Brownout detector generates the related interrupts.

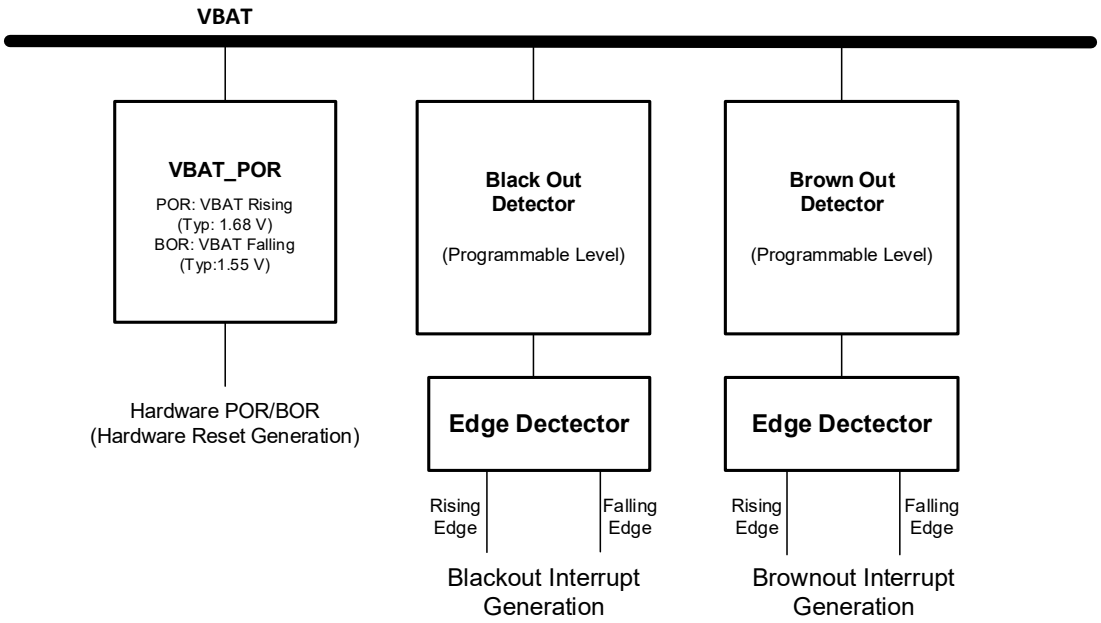


Figure 17. VBAT and POR, blackout/brownout detector

Table 34: $V_{brownout}$ and $V_{blackout}$ voltage levels

CTRL ID	HtoL (Avg.)	V_LtoH (Avg.)	Hysteresis (Avg.)
4	1.733	1.827	0.095
5	1.826	1.923	0.097
6	1.927	2.026	0.098
7	2.022	2.123	0.101
8	2.132	2.234	0.102
9	2.230	2.334	0.103
10	2.332	2.436	0.104
11	2.433	2.537	0.105
12	2.536	2.642	0.105
13	2.638	2.744	0.106
14	2.742	2.848	0.106
15	2.844	2.951	0.107

7.10 Security Features

The RA6W1 provides a high level of security by protecting the code and data that are being processed within the device. Additionally, the security engine provides acceleration of various crypto algorithms and provides protection for a secure Root of Trust (RoT) maintained within the device.

Features

- Secure XIP from Flash
  Secure XIP (eXecute In Place) supports AES-CTR description for XIP operation.
- Secure Crypto Engine
  - Symmetric algorithms: AES, DES/3DES, CHACHA
  - Hash/HMAC: SHA1/224/256
  - Asymmetric algorithms: RSA, DH, ECC
  - TRNG

- Secure boot

  Secure boot and secure loading processes are required to ensure that only authorized software can be executed on a device. Unauthorized boot code should be detected and prevented.
- Secure debug (SWD)

  When the RA6W1 enters the Secure mode, it can extract the SOC_ID. Every deployed device has a unique SOC_ID. If the image has a valid debug certificate, the internal security engine allows the debugger even if it is in Secure mode.
- Secure asset storage

  Secure Asset is a cryptographic service provided to protect data stored in external storage (Serial Flash memory). Data can be encrypted or decrypted with the provisioning key stored in the chip. Production-Line Provisioning is used to protect the data used in the mass production process, and Asset Provisioning is used to protect the data used during system operation.
- Device lifecycle management

  This mechanism enables the device to behave differently in each life cycle, protecting any security assets when they are introduced into the device and reducing the risk of IP theft and reverse engineering.
- TLS/DTLS protocol acceleration

  Hardware engines can support TLS/DTLS acceleration.
- DRBG (CTR_DRBG with AES, HMAC_DRBG)

  The RA6W1 supports hardware based Deterministic Random Bit Generator (DRBG).

7.10.1 Crypto Engine

The hardware crypto engine provides acceleration of many crypto algorithms such as hashing, secret key generation, encryption/decryption, and sign/verify operations.

Table 35 shows the hardware accelerated crypto algorithms supported.

Table 35: Hardware accelerated crypto algorithms

Algorithm	Mode	Key sizes
AES	ECB, CBC, CTR, OFB, CMAC, CBC-MAC, AESCCM, AES-CCM*, AES-GCM	128 bits, 192 bits, and 256 bits
	XTS, CFB128	Supported by software with AES-ECB hardware acceleration
AES key wrapping	N/A	All (MbedTLS)
Chacha20 and Poly1305	N/A	256 bits
Diffie-hellman ▪ ANSI X9.42-2003: Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography ▪ Public-Key Cryptography Standards (PKCS) #3: Diffie Hellman Key Agreement Standard	N/A	1024 bits, 2048 bits, and 3072 bits
ECC key generation	N/A	NIST curves and 25519 curves
ECIES	N/A	NIST curves and 25519 curves
ECDSA	N/A	NIST curves and ED25519
ECDH	N/A	NIST curves and 25519 curves
Hash	SHA1, SHA224, and SHA256	N/A
	SHA384, SHA512, and MD5	Supported by software
HKDF	N/A	N/A

Algorithm	Mode	Key sizes
HMAC	SHA1, SHA224, SHA256, and MD5	N/A
KDF ▪ NIST SP 800-108: Recommendation for Key Derivation Using Pseudorandom Functions	CMAC or HMAC	N/A
KDF ▪ Password based	PBKDF2-HMAC	Supported by software (with SHA1 hardware support)
RSA PKCS#1 operations ▪ Public-Key Cryptography Standards (PKCS) #1 v2.1: RSA Cryptography Specifications ▪ Public-Key Cryptography Standards (PKCS) #1 v1.5: RSA Encryption	Encryption and signature schemes	2048 bits, 3072 bits, and 4096 bits
RSA key generation	N/A	1024 bits to 3072 bits
RSA key validation	N/A	1024 bits to 4096 bits

7.11 Debug Support

The RA6W1 supports the low-pin-count Arm Serial Wire Debug (SWD) interface for real time debugging of the software. The Micro Trace Buffer (MTB) is included to provide support for tracing and real time profiling of code. The SWD protocol provides the same debug features as JTAG.

Features

- Processor halt, single-step, processor core register access, Vector Catch, unlimited software breakpoints, and full system memory access
- Hardware breakpoints and watchpoints:
  - A breakpoint unit supporting eight instruction comparators
  - A watchpoint unit supporting four data watchpoint comparators
- The MTB provides 8 kB of trace data to support profiling and timestamping of the code execution.

Figure 18 shows the SWD timing diagram.

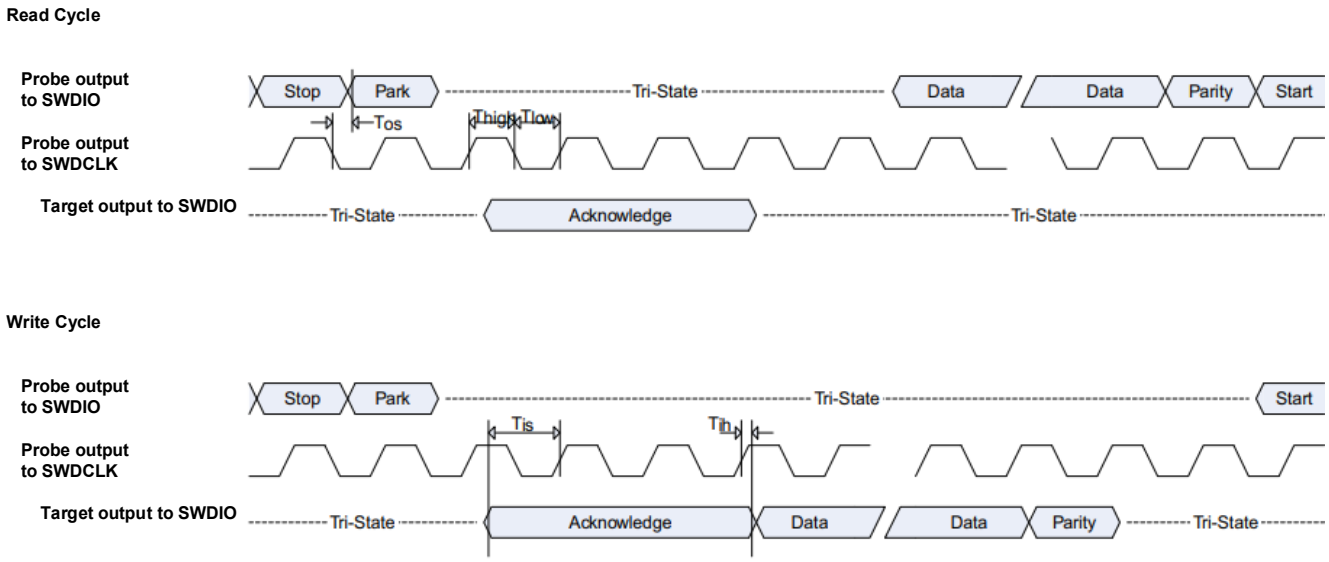


Figure 18. SWD timing diagram

Table 36 shows the SWD timing parameters.

Table 36: SWD timing parameters

Parameter	Parameter name	Min	Max	Unit
T_{high}	SWDCLK High period	10	500	us
T_{low}	SWDCLK Low period	10	500	us
T_{os}	SWDIO output skew to falling edge SWDCLK	-5	5	ns
T_{is}	Input Setup time required between SWDIO and rising edge SWDCLK	4		ns
T_{ih}	Input Hold time required between SWDIO and rising edge SWDCLK	1		ns

Table 37 shows the pin definition of the JTAG interface.

Table 37: SWD pin configuration

Pin name	Default pin assignment	I/O	Description
SWCLK	P1_16	I	Serial Wire Debug clock. The maximum clock frequency is 10 MHz.
SWDIO	P1_17	I/O	Serial Wire Debug data I/O.

Supported debug probes:

- Segger J-Link: https://www.segger.com/products/debug-probes/j-link/
- Segger J-Trace Pro:
  https://www.segger.com/products/debug-trace-probes/
  https://www.segger.com/downloads/jlink/
- Renesas E2 emulator:

https://www.renesas.com/us/en/software-tool/e2-emulator-rte0t00020kce00000r

8. Peripherals

8.1 UART

8.1.1 Introduction

The UART interface provides an industry compliant serial interface for communicating with other devices. Three independently configurable UARTs are available which support the RS-232 and RS-485 protocols.

- **Features**
  - Support RS-232 and RS-485 protocols
  - 32-byte deep transmit and receive FIFOs with a programmable FIFO level interrupt
  - Support for DMA
  - False start bit detection
  - Fully programmable serial interface characteristics:
    - Number of data bits per character (5, 6, 7, or 8 bits)
    - Optional parity bit (with odd or even select) and number of stop bits (1or 2)
    - Programmable flow control (CTS/RTS)
    - Programable baud rate generation:
    - Up to 921600 baud for UART
    - Up to 5 MBaud for RS-485.

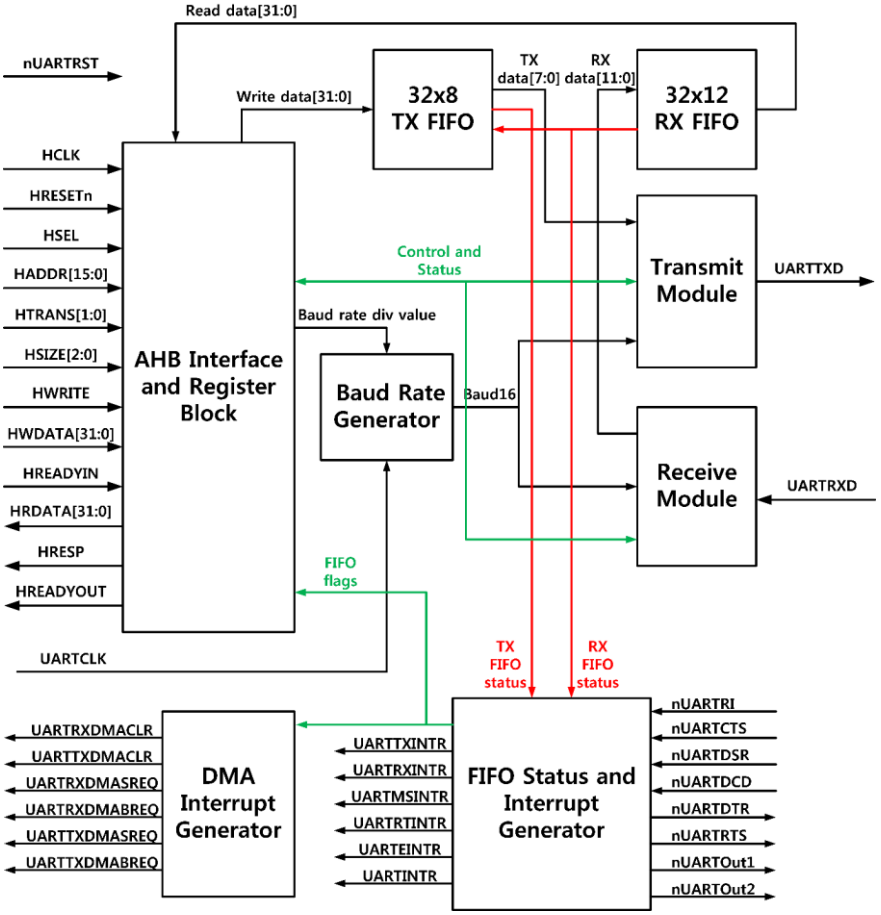


Figure 19. UART block diagram

8.1.2 RS-232

As the serial communication between the UART and the selected device is asynchronous, additional bits (start and stop) are inserted into the data line to indicate the beginning and end. With these bits, two devices can be

synchronized. This structure of serial data accompanied by start and stop bits is referred to as a character, as shown in Figure 20.

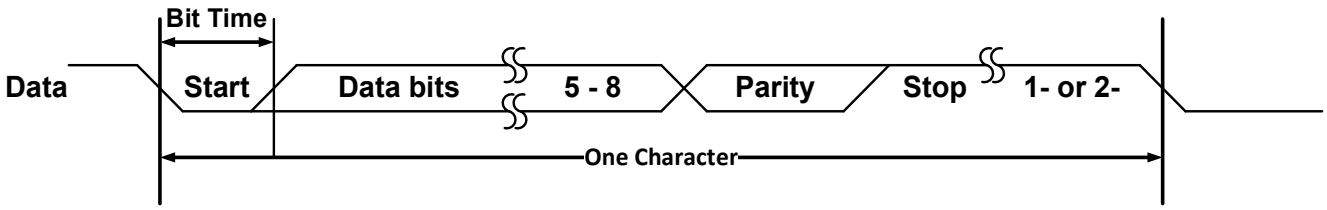


Figure 20. Serial data format

An additional parity bit may be added to the serial character. This bit appears between the last data bit and the stop bit(s) in the character structure. It provides UART with the ability to make simple error-checking on the received data.

The UART Line Control Register is used to control the serial character characteristics. The individual bits of the data word are sent after the start bit, starting with the least significant bit (LSB). These are followed by the optional parity bit, followed by the stop bit(s), which can be 1 or 2.

8.1.3 RS-485

Each UART can be configured to support the RS-485 serial protocol. When in RS-485 mode, an additional UART_TXDOE signal is provided to indicate the TXD active intervals. This signal can be assigned to any of the unused GPIO pins through the PPA.

8.1.4 Baud Rate

The clock that drives the UART block (FUARTCLK) is selectable between 32 kHz, 40 MHz, and 80 MHz depending on the system's operating mode.

The baud rate for each UART can be independently set by configuring its input clock divider with a divisor value consisting of an integer part and a fractional part.

These divisor values can be calculated:

baud rate divisor = (FUARTCLK/(16 × Baud rate))

The resulting baud rate devisor consists of an integer part and a fractional part.

The integer part is calculated:

integer part = integer (baud rate divisor)

Fractional part is calculated:

fractional part = integer ((baud rate divisor × 64) + 0.5)

8.1.5 Hardware Flow Control

The hardware flow control feature is fully selectable, and serial data flow is controlled by using RTS output and CTS input signals. Figure 21 shows how the two different UARTs can communicate using hardware flow control.

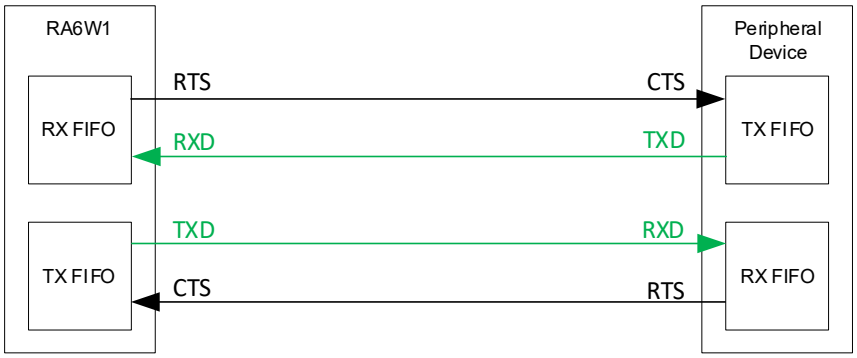


Figure 21. UART hardware flow control

When RTS flow control is enabled, RTS signal is asserted until the receive FIFO is filled up to programmed level. When CTS flow control is enabled, the transmitter can transmit the data when the CTS signal is asserted.

There are four modes of operation for flow control:

- Flow control is disabled
- Both RTS and CTS flow control are enabled
- Only CTS flow control is enabled
- Only RTS flow control is enabled.

8.1.6 Interrupts

Each UART has an associated interrupt in the interrupt vector table which is triggered by the following conditions:

- Overrun error
- Break error
- Parity error
- Framing error
- Receive timeout
- Transmit complete (based on FIFO trigger levels)
- Receive complete (based on FIFO trigger levels).

When an interrupt is triggered, the ISR checks the condition in the respective UART interrupt status register and then the appropriate action is taken.

8.1.7 DMA Interface

Each UART supports a connection to the GP-DMA so that transfers over the UART interface can be performed with minimal CPU overhead. Each UART supports a GP-DMA channel for both the receive and transmit functions.

8.1.8 Pin Configuration

The UART pins can be assigned to any of the unused GPIO pins through the Programmable Pin Assignment (PPA) function. Table 38 shows the pin definitions for the UART interfaces. The PPA provides a multiplexing function for the I/O pins of the on-chip peripherals. Any of the peripheral input or output signals can be freely mapped to any available GPIO port.

Table 38: UART pin configuration

Pin name	Default pin assignment	I/O
UART_RX	PPA	I
UART_TX	PPA	O
UART_CTS	PPA	I
UART_RTS	PPA	O
UART_TXDOE	PPA	I/O
UART1_RX	PPA	I
UART1_TX	PPA	O
UART1_CTS	PPA	I
UART1_RTS	PPA	O
UART1_TXDOE	PPA	I/O
UART2_RX	PPA	I
UART2_TX	PPA	O
UART2_CTS	PPA	I
UART2_RTS	PPA	O

Pin name	Default pin assignment	I/O
UART2_TXDOE	PPA	I/O

8.2 I2C Interface

8.2.1 Introduction

The I2C Interface is a bus that provides communications link between various peripheral devices in a system. It is a simple two-wire bus with a software-defined protocol for system control, which is used in temperature sensors and voltage level translators to EEPROMs, general-purpose I/O, A/D, and D/A converters.

Features

- Two-wire I2C serial interface consisting of a serial data line (SDA) and a serial clock (SCL)
- Three speed modes are supported:
  - Standard mode (0 to 100 kbit/s)
  - Fast mode (<= 400 kbit/s)
  - High speed mode (<= 3.4 Mbit/s)
- Clock synchronization
- 32 entry deep transmit and receive FIFOs (32 x 8-bit RX, 32 x 10-bit TX)
- Master transmit, Master receive operation
- 7-bit or 10-bit addressing
- 7-bit or 10-bit combined format transfers
- Bulk transmit mode
- Configurable slave address (Default address of 0x055)
- Interrupt or Polled mode of operation
- Handle Bit and Byte waiting at both bus speeds
- Programmable SDA hold time
- DMA support.

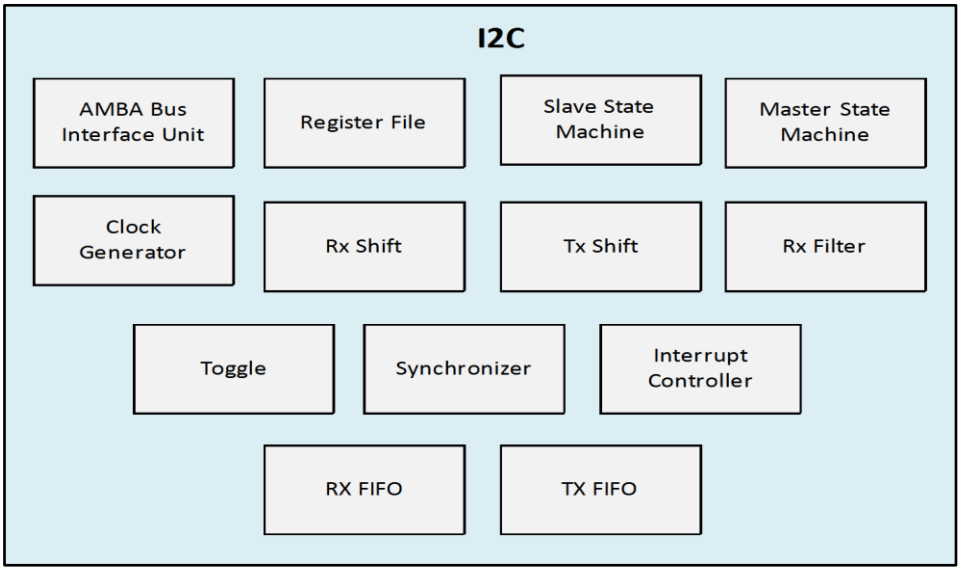


Figure 22. I2C Controller block diagram

8.2.2 I2C Behavior

The I2C can be configured to be an I2C master only, communicating with other I2C slaves.

The master is responsible for generating the clock and controlling the transfer of data. The slave is responsible for receiving data from and sending response data to the master. Data acknowledgement is sent by the device

that receives data, which can be master or slave. The I2C protocol allows multiple masters to reside on the I2C bus. It uses an arbitration procedure to determine bus ownership.

Each slave has a unique address that is determined by the system designer. When a master wants to communicate with a slave, the master transmits a START/RESTART condition that is then followed by the slave's address and a control bit (R/W) to determine whether the master transmits data or receives data from the slave. The slave then sends an acknowledge pulse (ACK) after the address.

If the master (master-transmitter) is writing to the slave (slave-receiver), the receiver gets one byte of data. This transaction continues until the master terminates the transmission with a STOP condition. If the master is reading from a slave (master-receiver), the slave transmits (slave-transmitter) a byte of data to the master, and the master then acknowledges the transaction with the ACK pulse. This transaction continues until the master terminates the transmission by not acknowledging (NACK) the transaction after the last byte is received, and then the master issues a STOP condition or addresses another slave after issuing a RESTART condition. This behavior is shown in Figure 23.

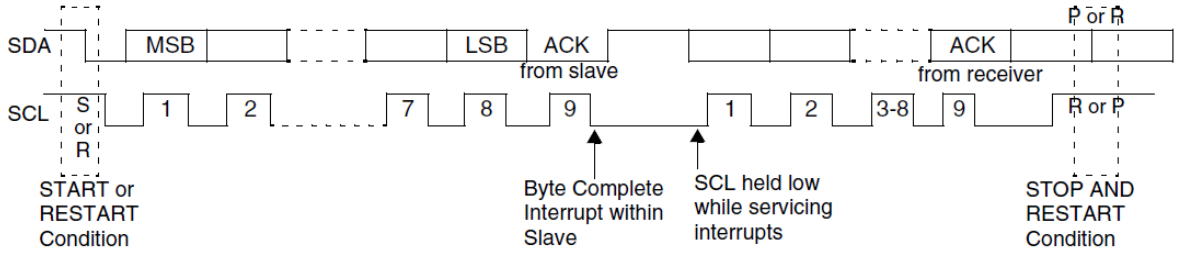


Figure 23. Data transfer on I2C bus

The I2C is a synchronous serial interface. The SDA line is a bidirectional signal and changes only while the SCL line is low, except for STOP, START, and RESTART conditions. The output drivers are open-drain or open-collector to perform wire-AND functions on the bus. The maximum number of devices on the bus is limited by only the maximum capacitance specification of 400 pF. Data is transmitted in byte packages.

8.2.2.1 START and STOP Generation

When operating as an I2C master, putting data into the transmit FIFO causes the I2C Controller to generate a START condition on the I2C bus. Writing a 1 to I2C_DATA_CMD_REG causes the I2C to generate a STOP condition on the I2C bus; a STOP condition is not issued if this bit is not set, even if the transmit FIFO is empty.

When operating as a slave, the I2C Controller does not generate START and STOP conditions, as per the protocol. However, if a read request is made to the I2C Controller, it holds the SCL line low until read data has been supplied to it. This stalls the I2C bus until read data is provided to the slave I2C Controller, or the I2C Controller slave is disabled by writing a 0 to I2C_ENABLE.

8.2.2.2 Combined Formats

The I2C Controller supports mixed read and write combined format transactions in both 7-bit and 10-bit addressing modes.

The I2C Controller does not support mixed address and mixed address format - that is, a 7-bit address transaction followed by a 10-bit address transaction or the other way round – combined format transactions.

To initiate combined format transfers, I2C_CON.I2C_RESTART_EN should be set to 1. With this value set and operating as a master, when the I2C Controller completes an I2C transfer, it checks the transmit FIFO and executes the next transfer. If the direction of this transfer differs from the previous transfer, the combined format is used to issue the transfer. If the transmit FIFO is empty when the current I2C transfer completes, a STOP is issued, and the next transfer is issued following a START condition.

8.2.3 I2C Protocols

The I2C Controller has the following protocols:

- START and STOP Conditions
- Addressing Slave Protocol
- Transmitting and Receiving Protocol
- START Byte Transfer Protocol.

8.2.3.1 START and STOP Conditions

When the bus is idle, both the SCL and SDA signals are pulled high through external pull-up resistors on the bus. When the master wants to start a transmission on the bus, the master issues a START condition. This is defined as a high-to-low transition of the SDA signal while SCL is 1. When the master wants to terminate the transmission, the master issues a STOP condition. This is defined to be a low-to-high transition of the SDA line while SCL is 1. Figure 24 shows the timing of the START and STOP conditions. When data is being transmitted on the bus, the SDA line must be stable when SCL is 1.

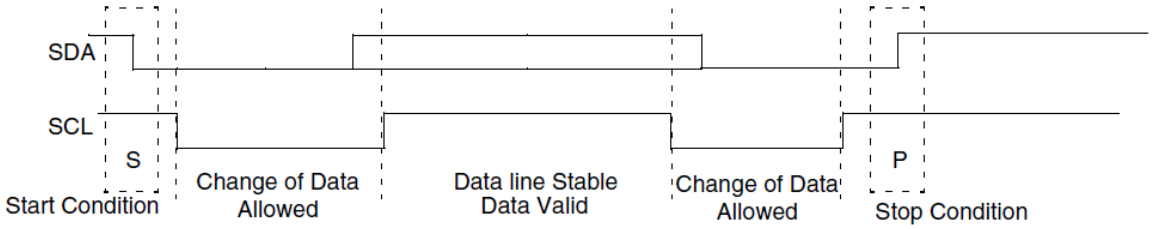


Figure 24. START and STOP conditions

NOTE
The signal transitions for the START/STOP conditions (see Figure 24) reflect those observed at the output signals of the Master driving the I2C bus. Care should be taken when observing the SDA/SCL signals at the input signals of the Slave(s), because unequal line delays may result in an incorrect SDA/SCL timing relationship.

8.2.3.2 Addressing Slave Protocol

There are two address formats: 7-bit address and 10-bit address.

8.2.3.2.1 7-bit Address Format

During the 7-bit address format, the first seven bits (bits 7:1) of the first byte set the slave address and the LSB bit (bit 0) is the R/W bit as shown in Figure 25. When bit 0 (R/W) is set to 0, the master writes to the slave. When bit 0 (R/W) is set to 1, the master reads from the slave.

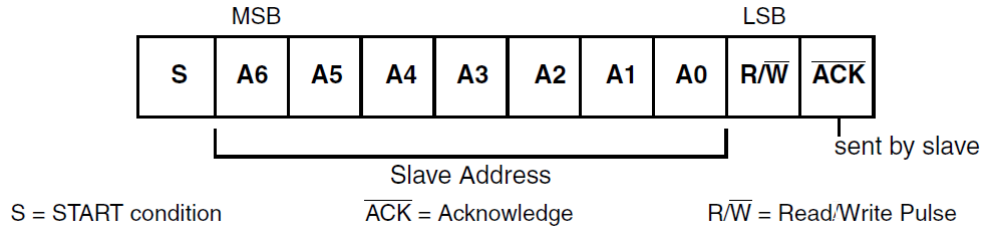


Figure 25. 7-bit address format

8.2.3.2.2 10-bit Address Format

During 10-bit addressing, two bytes are transferred to set the 10-bit address. The transfer of the first byte contains the following bit definition. The first five bits (bits 7:3) notify the slaves that this is a 10-bit transfer followed by the next two bits (bits 2:1), which set the slaves address bits 9:8, and the LSB bit (bit 0) is the R/W bit. The second byte transferred sets bits 7:0 of the slave address. Figure 26 shows the 10-bit address format, and Table 39 defines the special purpose and reserves first byte addresses.

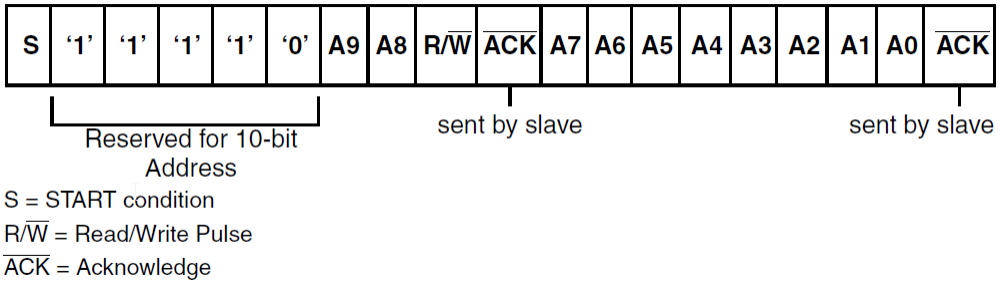


Figure 26. 10-bit address format

Table 39: I2C definition of bits in first byte

Slave address	R/W bit	Description
0000 000	0	▪ General Call Address. ▪ I2C Controller places the data in the receive buffer and issues a General Call interrupt.
0000 000	1	START byte. For more details, see Section 8.2.3.3.3 "START BYTE Transfer Protocol".
0000 001	X	CBUS address. I2C Controller ignores these accesses.
0000 010	X	Reserved (Note 1)
0000 011	X	Reserved (Note 1)
0000 1XX	X	High-speed master code. For more information, see Section 8.2.4 "Multiple Master Arbitration".
1111 1XX	X	Reserved (Note 1)
1111 0XX	X	10-bit slave addressing.

Note 1 Use of the reserved addresses may cause incompatibilities with other I2C devices.

8.2.3.3 Transmitting and Receiving Protocols

The master can initiate data transmission and reception to/from the bus, acting as either a master-transmitter or master-receiver. A slave responds to requests from the master to either transmit data or receive data to/from the bus, acting as either a slave-transmitter or slave-receiver, respectively.

8.2.3.3.1 Master-Transmitter and Slave-Receiver

All data is transmitted in byte format, with no limit on the number of bytes transferred per data transfer. After the master sends the address and R/W bit or the master transmits a byte of data to the slave, the slave-receiver must respond with the acknowledge signal (ACK). When a slave-receiver does not respond with an ACK pulse, the master aborts the transfer by issuing a STOP condition. The slave must leave the SDA line high so that the master can abort the transfer.

If the master-transmitter is transmitting data as shown in Figure 27, then the slave-receiver responds to the master-transmitter with an acknowledge pulse after every byte of data is received.

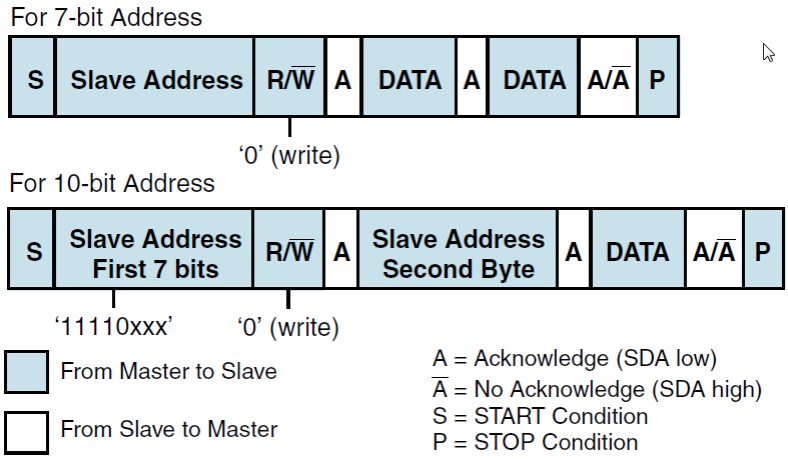


Figure 27. Master-Transmitter protocol

8.2.3.3.2 Master-Receiver and Slave-Transmitter

If the master is receiving data as shown in Figure 28, then the master responds to the slave-transmitter with an acknowledge pulse after a byte of data has been received, except for the last byte. This is the way the master-receiver notifies the slave-transmitter that this is the last byte. The slave-transmitter relinquishes the SDA line after detecting the No Acknowledge (NACK) so that the master can issue a STOP condition.

When a master does not want to relinquish the bus with a STOP condition, the master can issue a RESTART condition. This is identical to a START condition except it occurs after the ACK pulse. The master can then communicate with the same slave or a different slave.

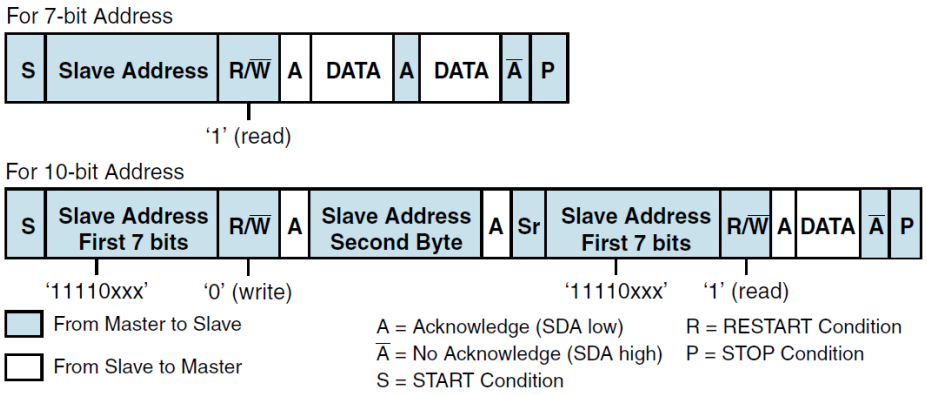


Figure 28. Master-Receiver protocol

8.2.3.3.3 START Byte Transfer Protocol

The START Byte transfer protocol is set up for systems that do not have an on-board dedicated I2C hardware module. When the I2C Controller is addressed as a slave, it always samples the I2C bus at the highest speed supported so that it never requires a START Byte transfer. However, when I2C Controller is a master, it supports the generation of START Byte transfers at the beginning of every transfer in case a slave device requires it. This protocol consists of seven zeros being transmitted followed by a 1, as shown in Figure 29. This allows the processor that is polling the bus to under-sample the address phase until 0 is detected. When the microcontroller detects a 0, it switches from the under-sampling rate to the correct rate of the master.

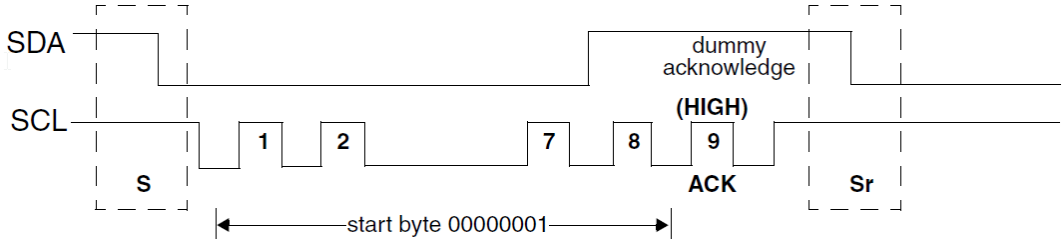


Figure 29. START byte transfer

The START Byte procedure is as follows:

1. Master generates a START condition.
2. Master transmits the START byte (0000 0001).
3. Master transmits the ACK clock pulse (Present only to conform with the byte handling format used on the bus).
4. No slave sets the ACK signal to 0.
5. Master generates a RESTART (R) condition.

A hardware receiver does not respond to the START Byte because it is a reserved address and resets after the RESTART condition is generated.

8.2.4 Multiple Master Arbitration

The I2C Controller bus protocol allows multiple masters to reside on the same bus. If there are two masters on the same I2C bus, there is an arbitration procedure if both try to take control of the bus at the same time by generating a START condition. When a master (for example, a microcontroller) has control of the bus, no other master can take control until the first master sends a STOP condition and places the bus in an idle state.

Arbitration takes place on the SDA line, while the SCL line is 1. The master, which transmits a 1 while the other master transmits 0, loses arbitration and turns off its data output stage. The master that lost arbitration can continue to generate clocks until the end of the byte transfer. If both masters are addressing the same slave device, the arbitration might go into the data phase. Figure 30 shows the timing of when two masters are arbitrating on the bus.

For high-speed mode, the arbitration cannot go into the data phase because each master is programmed with a unique high-speed master code. This 8-bit code is defined by the system designer and is set by writing to the High-Speed Master Mode Code Address Register, I2C_HS_MADDR. Because the codes are unique, only one master can win arbitration, which occurs at the end of the transmission of the high-speed master code.

Control of the bus is determined by address or master code and data sent by competing masters, so there is no central master or any order of priority on the bus.

Arbitration is not allowed between the following conditions:

- A RESTART condition and a data bit
- A STOP condition and a data bit
- A RESTART condition and a STOP condition.

Slaves are not involved in the arbitration process.

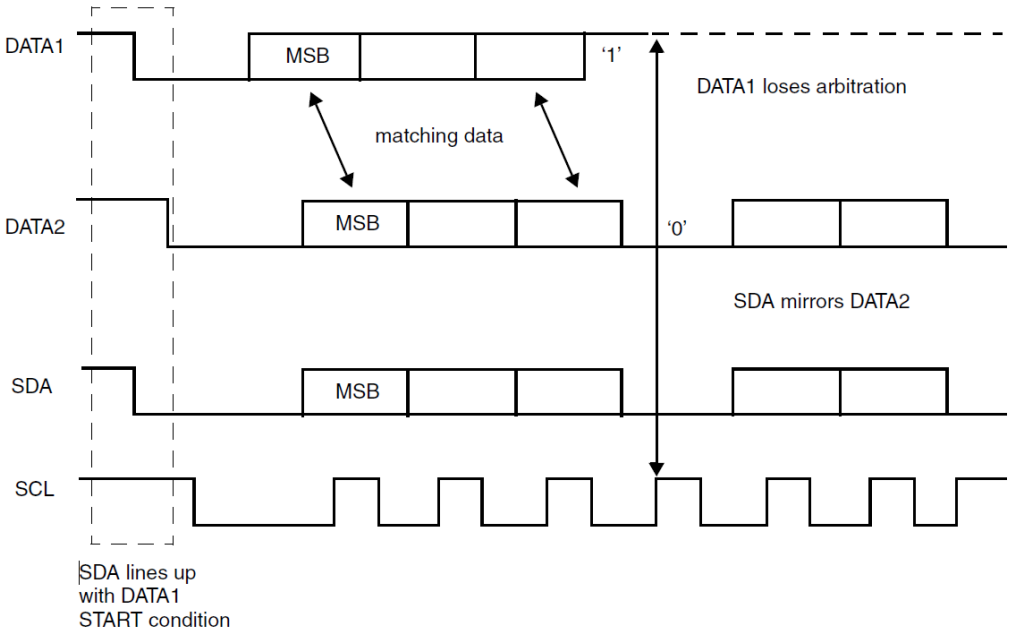


Figure 30. Multiple Master arbitration

8.2.5 Clock Synchronization

When two or more masters try to transfer information on the bus at the same time, they must arbitrate and synchronize the SCL clock. All masters generate their own clock to transfer messages. Data is valid only during the high period of SCL clock. Clock synchronization is performed using the wired-AND connection to the SCL signal. When the master transitions the SCL clock to 0, the master starts counting the low time of the SCL clock and transitions the SCL clock signal to 1 at the beginning of the next clock period. However, if another master is holding the SCL line to 0, then the master goes into a HIGH wait state until the SCL clock line transitions to 1.

All masters then count off their high time, and the master with the shortest high time, transitions the SCL line to 0. The masters then count out their low time. The one with the longest low time, forces the other master into a HIGH wait state. Therefore, a synchronized SCL clock is generated as shown in Figure 31. Optionally, slaves may hold the SCL line low, to slow down the timing on the I2C bus.

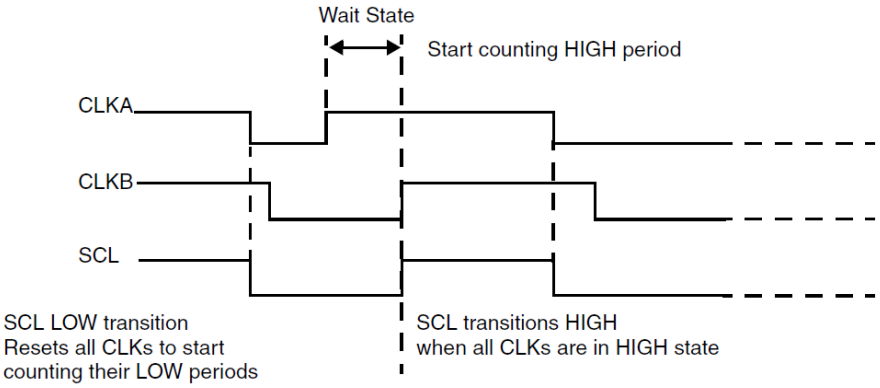


Figure 31. Multiple master clock synchronization

8.3 Digital Audio Interface (I2S and PDM)

8.3.1 Introduction

The Digital Audio Interface (DAI) is a 4-wire synchronous audio interface to Codecs like Renesas DA14740x. The I2S mode supports master or slave operation for stereo audio with two 32-bit channels. The PDM mode supports master or slave operation for sample rates of 24, 32, or 48 kHz.

Features

- I2S Master and Slave mode
- Sample rates: 8, 12, 16, 24, 32, 44.1, and 48 kHz
- Data word length: 16, 20, 24, or 32 bits
- Frame length: 32, 64, 128, 256 bits wide
- Maximum number of slots: 16
- Number of channels: 2
- Sample Rate Converter (SRC)
- Formats: Digital signal processing, left-justified, right-justified, and I^2S
- Time division multiplexed mode, with push-pull in active and Hi-Z is non active slots
- Programmable active rising/falling edges of PCM_WCLK and PCM_BCLK
- Supports Pulse Density Modulation (PDM) for interfacing with DMICs.

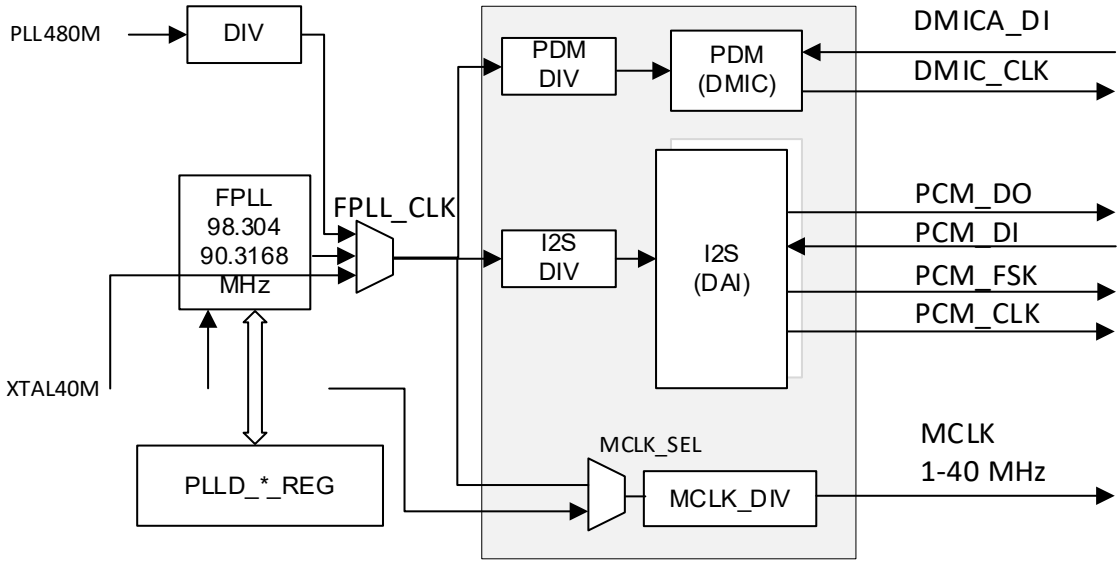


Figure 32. DAI block diagram

8.3.2 Interface Signals

The DAI is a four-wire serial interface. The pins and signals are mapped as shown in Table 40.

Table 40: DAI pins and signals

DAI signals	Description
PCM_BCLK	Bit clock (BCLK) INPUT/OUTPUT Maximum 24.576 MHz
PCM_WCLK	Word clock (WCLK) IN/OUT Maximum 192 kHz
PCM_DI	Input data (DATA_IN)

DAI signals	Description
PCM_DO	Output (DATA_OUT)
MCLK	Master clock (OUTPUT) to Codec (can be disabled if BCLK is selected as MCLK in the Codec
DMICA_DI	Input Data for the Digital Mic interface
DMIC_CLK	Clock for the Digital Mic interface

The MCLK signal is used to clock an external codec. This signal can be derived directly from the FPLL or XTAL40M. The PCM_DO output state can be configured according to Table 41.

Table 41: PCM_DO output states

Pxy_MODE_REG		DAI_DATA_OUT_CTRL_REG DAI2_DATA_OUT_CTRL_REG	PCM_DO state
PUPD	**PPOD**	**DATA_OUT_EN[1-0]**	
0,1,2	Don't care	Don't care	Hi-Z
3	Don't care	0 or 1	Hi-Z
3	0	2	0: Driving all slots (single master) 1: Driving all slots (single master)
3	0	3	**Active slot** 0: Driving 1: Driving **Non-active slot:** Hi-Z
3	1	3	**Active slot** 0: Driving 1: Open drain Hi-Z **Non-active slot:** Hi-Z

8.3.3 Master and Slave Modes

The DAI operates in either Master mode or Slave mode as shown in Figure 33.

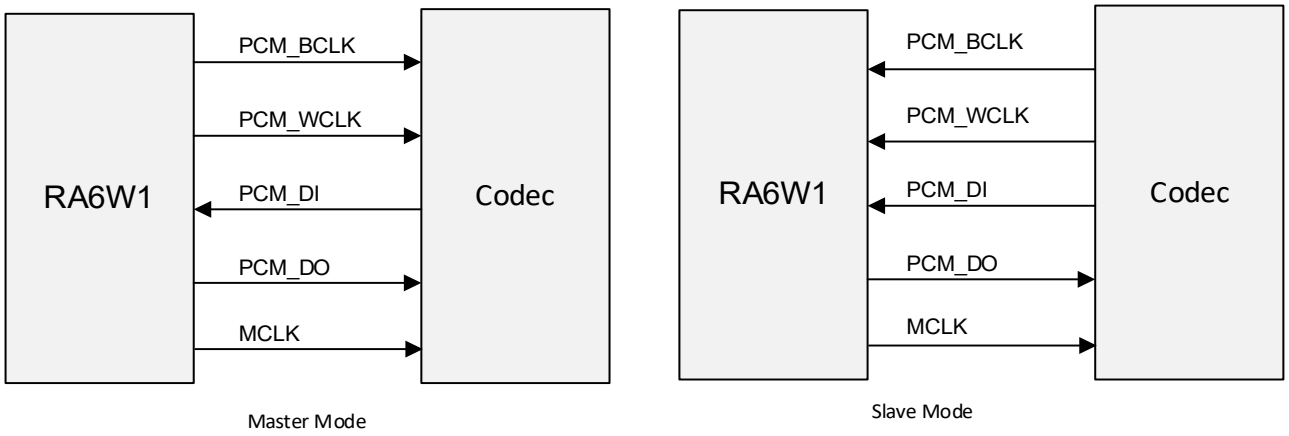


Figure 33. DAI Master and Slave modes

The bit clock (PCM_BCLK) samples receive data into the DAI through the PCM_DI pin and transmit through the PCM_DO pin. The word clock (PCM_WCLK) is the DAI data sample clock, synchronizing the sample frames for the DAI data channels.

In Master mode, the DAI provides synchronization clocks, PCM_BCLK and PCM_WCLK. In Slave mode, BCLK and WCLK must be provided externally.

8.3.4 DAI Slots

The DAI can serve up to 16 slots of 16 bits maximum and up to 8 slots of 32 bits maximum. The maximum frame length of DAI is 256 bits. A configurable slot offset determines the start of frame for Channel 1. The offset prevents conflict when two or more devices are on the bus.

8.3.5 DAI Slot Formats

8.3.5.1 I²S Format

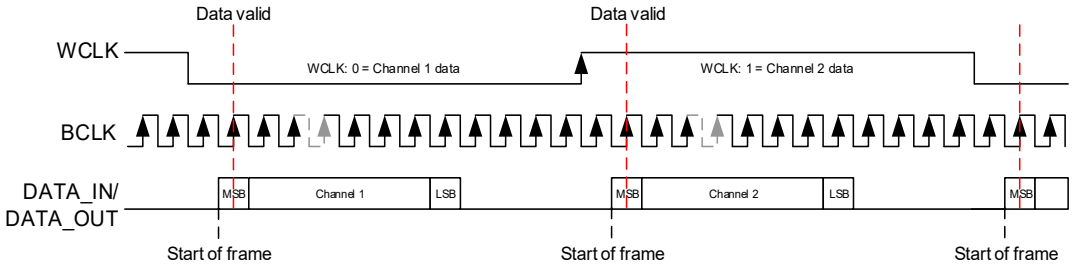


Figure 34. I²S format

In I²S format, the start of frame for Channel 1 is on the second falling edge of BCLK after a falling edge of WCLK. The MSB of Channel 1 is valid on the rising edge of BCLK after the start of frame condition.

The start of frame for Channel 2 is on the second falling edge of BCLK after a rising edge of WCLK. The MSB of Channel 2 is valid on the rising edge of BCLK after the start of frame condition.

8.3.5.2 DSP Format

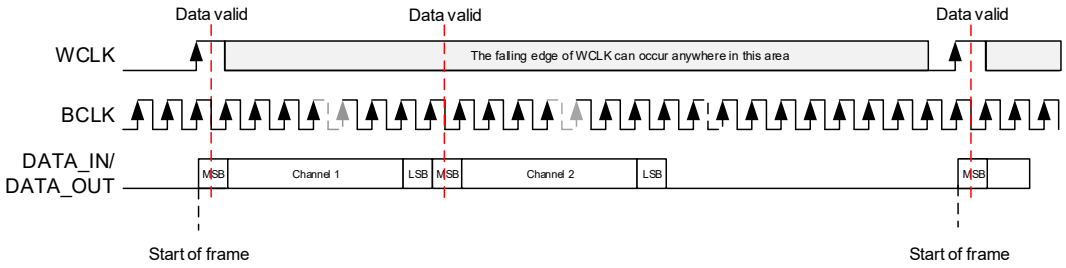


Figure 35. DSP format

In DSP format, the rising edge of WCLK starts the data transfer (start of frame) with the Channel 1 data first, immediately followed by Channel 2 data and any subsequent channels. Each data bit is valid on the falling edge of BCLK.

8.3.5.3 Left-Justified Format

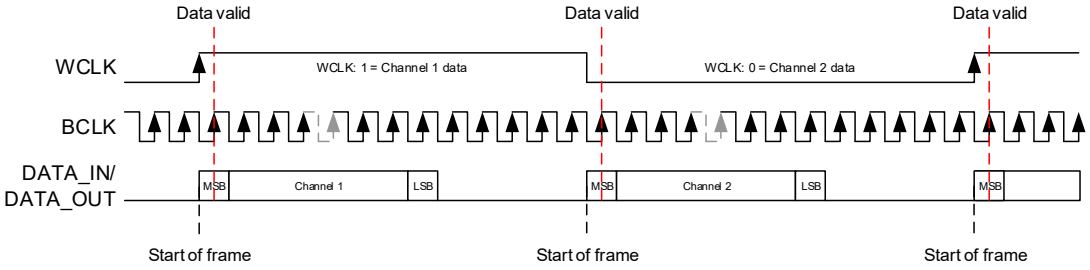


Figure 36. Left-justified format

In left-justified format (LJF), the MSB of Channel 1 is valid on the rising edge of BCLK following the rising edge of WCLK. The MSB of Channel 2 is valid on the rising edge of BCLK following the falling edge of WCLK.

8.3.5.4 Right-Justified Format

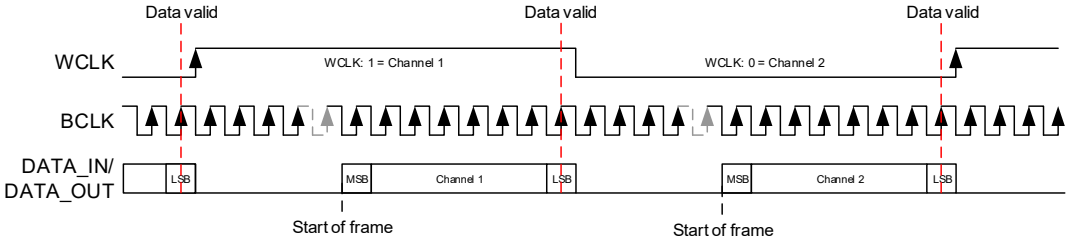


Figure 37. Right-justified format

In right-justified format (RJF), the LSB of Channel 1 is valid on the rising edge of BCLK preceding the falling edge of WCLK. The LSB of Channel 2 is valid on the rising edge of BCLK preceding the rising edge of the WCLK.

8.3.5.5 Time Division Multiplexing Mode

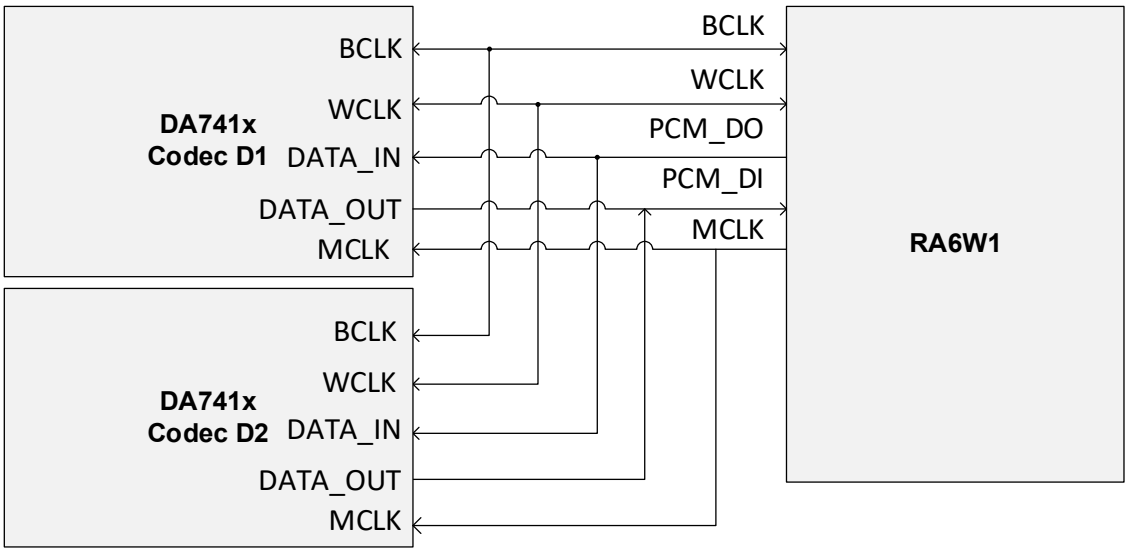


Figure 38. TDM configuration

Time division multiplexing (TDM) mode allows multiple devices to communicate on the same bus without conflicting, see Figure 38. The serial data pin is tri-stated whenever the output is not valid to allow other devices on the bus to drive the data line.

TDM mode is available in both Master and Slave modes. The TDM mode is an extension of LJF, see Figure 39 and for DSP format, see Figure 40.

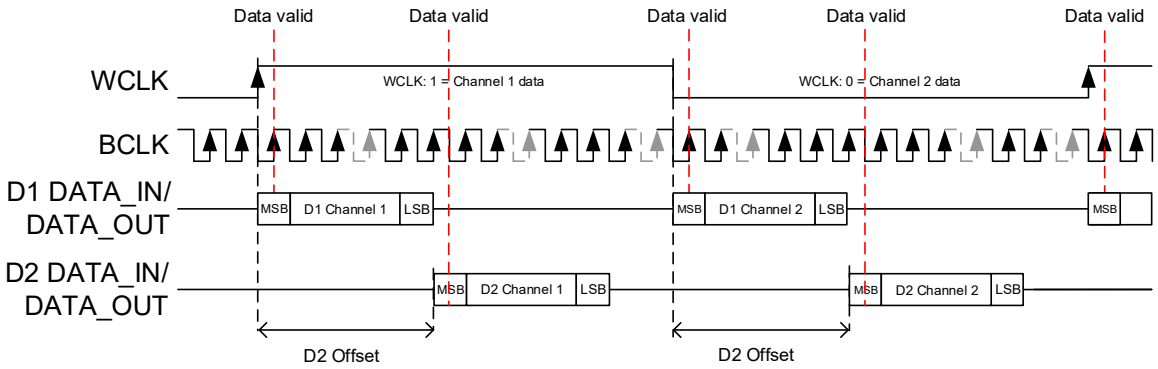


Figure 39. Two devices in LJF mode with TDM mode active

In LJF with TDM mode active, the Device 2 (D2) Channel 1 data is offset by a configurable number of BCLK cycles (D2 Offset) after the rising edge of WCLK. The D2 Channel 2 data is valid for the same number of BLCK cycles (D2 Offset) after the falling edge of WCLK.

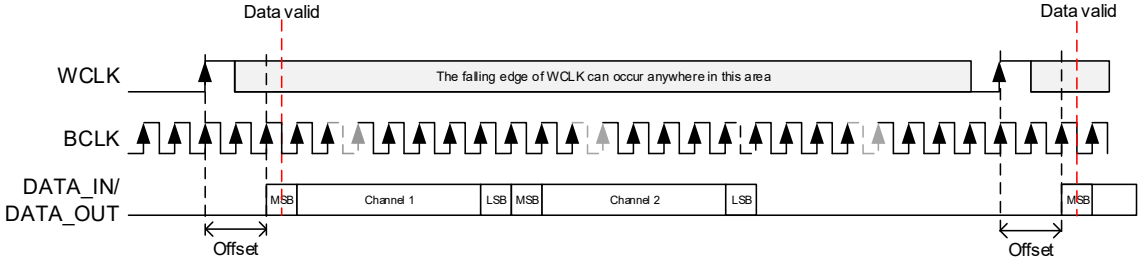


Figure 40. One device in DSP mode with offset from TDM mode

In DSP format with TDM mode active the start of frame is offset by a configurable number of BCLK cycles (Offset) from the rising edge of WCLK. The Channel 1 data is valid on the first falling edge of BCLK after the start of frame condition. Channel 2 data immediately follow Channel 1 data.

8.3.6 DAI Slot Assignment

Table 42 shows the slot assignment for the various formats depending on PCM_WCLK edge, WCLK_POL value, and SLOT_CNT up to 8 are depicted.

Table 42: DAI format summary

SLOT_CNT = 2										
WCLK_POL = 0	WCLKedge	T0	T1	T2	T3	WCLKedge	T4	T5	T6	T7
I2S	Falling	Slot1				Rising	Slot2			-
LJF	Rising	Slot1				Falling	Slot2			-
RJF	Rising				Slot1	Falling				Slot2
DSP	Rising	Slot1	Slot2							
WCLK_POL = 1										
I2S	Rising	Slot1				Falling	Slot2			
LJF	Falling	Slot1				Rising	Slot2			
RJF	Falling				Slot1	Rising				Slot2
DSP	Falling	Slot1	Slot2							

SLOT_CNT = 4										
WCLK_POL = 0	WCLKedge	T0	T1	T2	T3	WCLKedge	T4	T5	T6	T7
I2S	Falling	Slot1	Slot3			Rising	Slot2	Slot4		
LJF	Rising	Slot1	Slot3			Falling	Slot2	Slot4		
RJF	Rising			Slot1	Slot3	Falling			Slot2	Slot4
DSP	Rising	Slot1	Slot2	Slot3	Slot4					
WCLK_POL = 1										
I2S	Rising	Slot1	Slot3			Falling	Slot2	Slot4		
LJF	Falling	Slot1	Slot3			Rising	Slot2	Slot4		
RJF	Falling			Slot1	Slot3	Rising			Slot2	Slot4
DSP	Falling	Slot1	Slot2	Slot3	Slot4					
SLOT_CNT = 8										
WCLK_POL = 0	WCLKedge	T0	T1	T2	T3	WCLKedge	T4	T5	T6	T7
I2S	Falling	Slot1	Slot3	Slot5	Slot7	Rising	Slot2	Slot4	Slot6	Slot8
LJF	Rising	Slot1	Slot3	Slot5	Slot7	Falling	Slot2	Slot4	Slot6	Slot8
RJF	Rising	Slot1	Slot3	Slot5	Slot7	Falling	Slot2	Slot4	Slot6	Slot8
DSP	Rising	Slot1	Slot2	Slot3	Slot4		Slot5	Slot6	Slot7	Slot8
WCLK_POL = 1										
I2S	Rising	Slot1	Slot3	Slot5	Slot7	Falling	Slot2	Slot4	Slot6	Slot8
LJF	Falling	Slot1	Slot3	Slot5	Slot7	Rising	Slot2	Slot4	Slot6	Slot8
RJF	Falling	Slot1	Slot3	Slot5	Slot7	Rising	Slot2	Slot4	Slot6	Slot8
DSP	Falling	Slot1	Slot2	Slot3	Slot4		Slot5	Slot6	Slot7	Slot8
Applying offset (DAI_OFFSET_LSB_REG, DAI_OFFSET_MSB_REG]										
WCLK_POL = 0	WCLKedge	T0	T1	T2	T3	WCLKedge	T4	T5	T6	T7
I2S	Falling	Slot1				Rising	Slot2			
I2S (Offset)	Falling		Slot1			Rising		Slot2		

8.3.7 DAI PCM_CLK Generation

In Master mode, the Slot length can be programmed to be 16, 20, 24, or 32 bits. If the slot length is 24 bits and multiple slots are used, the slots are concatenated without padding bit. If padding bits are required (for example, for a DAI tester) use W_LEN is 32 bits and make the 8 LSB bits 0. In TDM mode with open drain selected, the 8 LSB bits can be set to 1 to allow other bus masters to use these slots.

The MSB of register or SRC input/output is first transmitted and received and the remaining bits in LSBs are not transmitted and received if word length is not equal to 32 bits wide.

In Slave mode, the DAI automatically detects the frame length (number of BLCKs per WCLK). In Master mode, the frame length is configurable to be 32, 64, 128, or 256 bits wide.

The PCM_CLK is automatically set to a maximum of 24.576 MHz, according to this formula:

- For PCM_CLK = DAI_SR * FRAME_LEN

The maximum number of possible slots is:

- For DSP format FRAME_LEN/W_LEN
- For I2S, LJF and RJF format for int(FRAME_LEN/(2*W_LEN))*2

Table 43 and Table 44 show typical examples of PCM clock settings. PCM_DIV has a mandatory value of 1.

Table 43: PCM clock generation examples (FPLL = 98.304 MHz, AUD_CLK_DIV = 4, PCM_DIV =1)

DAI_SR	FRAME_LEN	W_LEN	SLOT_CNT	PCM_CLK (kHz) (Auto set)	PCM_DIV (Mandatory)
0xB: 48 kHz	1: 64 bits	3: 32 bits	2	3072	1
0xB: 48 kHz	2: 128 bits	3: 32 bits	2	6144	1

Table 44: PCM generation example (FPLL = 90.3168 MHz, AUD_CLK_DIV = 4, PCM_DIV = 1)

DAI_SR	FRAME_LEN	W_LEN	SLOT_CNT	PCM_CLK (kHz) (Auto set)	PCM_DIV (Mandatory)
0x0A: 44.1 kHz	1: 64 bits	3: 32 bits	2	2822.4	1

8.4 SPI Master/Slave

8.4.1 Introduction

The Serial Peripheral Interface (SPI™) supports master and Slave modes of operation. The serial interface can transmit and receive from 4 to up to 32 bits in Master/Slave mode. The controller comprises separate TX and RX FIFOs and DMA handshake support. Slave mode clock speed is independent from the system clock speed. The controller can generate an interrupt upon data threshold reached in the TX or RX FIFOs.

Features

- Slave and Master mode for SPI/SPI2
- From 4-bit to up to 32-bit operation
- SPI/SPI2 SPI_CLK Master mode clock line up to 48 MHz
- SPI mode 0, 1, 2, and 3 support (clock edge and phase)
- Built-in separate 8-bit wide and 32-byte deep RX/TX FIFOs for continuous SPI bursts (SPI, SPI2)
- Maskable interrupt generation based on TX or RX FIFO thresholds
- DMA support.

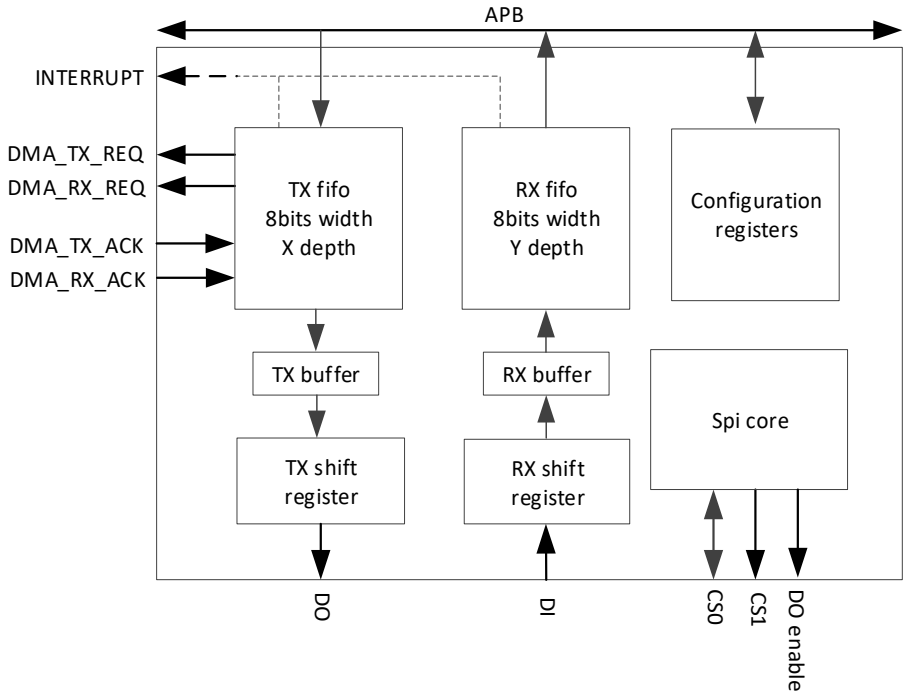


Figure 41. SPI block diagram

The SPI controller is responsible for the serialization/deserialization of the data in the RX and TX streams.

Two separate FIFOs are used to store data for RX and TX streams. Since a SPI word can be configured to be from four bits to up to 32 bits, one to four FIFO positions can be written/read at the same time. FIFOs contain logic implementing programmable thresholds comparison.

The SPI controller supports DMA requests and interrupt generation based on the FIFO thresholds. If enabled, a DMA request and/or interrupt is asserted with whether TX_FIFO level is low, or RX_FIFO level is high.

The SPI interface supports all four modes of operation and the corresponding polarity (CPOL) and phase (CPHA) of the SPI clock (SPI_CLK) are defined in Table 45.

Table 45: SPI modes configuration and SCK states

SPI mode	CPOL	CHPA	TX SPI_CLK	RX SPI_CLK	Idle SPI_CLK
0	0	0	Falling edge	Rising edge	Low
1	0	1	Rising edge	Falling edge	Low
2	1	0	Rising edge	Falling edge	High
3	1	1	Falling edge	Rising edge	High

To read from or to write to an external single byte Flash device in the SPI Master mode, a byte swap mechanism is implemented to allow for a proper placement of the bytes in a 16-bit word for the DMA to write to/read from the internal RAM. More specifically, when the SPI controller is configured as a master with DMA support and a 16-bit word width so that the bus utilization is increased compared to reading from an 8-bit device, the byte swap mechanism brings the least significant byte read and place it in the most significant byte in the 16-bit word. The controller automatically swaps the bytes to allow for placing the first byte read in the least significant byte of the 16-bit word.

8.4.2 SPI Timing

The SPI interface timing for Master and Slave modes is shown in Figure 42.

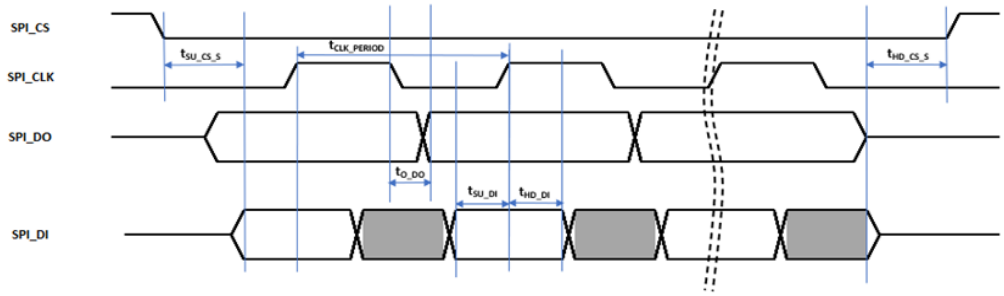


Figure 42. SPI timing (CPOL = 0, CPHA = 0)

Table 46: SPI timing parameters

Description	Parameter	Condition	Min	Typ	Max	Unit
SPI_CLK clock frequency (I/O = 3.3 V)	$t_{CLK\_M\_3V3}$	Master mode			48	MHz
SPI_CLK period (I/O = 3.3 V)	$t_{CLK\_PERIOD\_3V3}$	Master mode	20.84			ns
SPI_CLK clock frequency (I/O = 1.8 V)	$t_{CLK\_M\_1V8}$	Master mode			40	MHz
SPI_CLK period (I/O = 1.8 V)	$t_{CLK\_PERIOD\_1V8}$	Master mode	25			ns
SPI_CLK duty	t_{CLK_DUTY}	Master mode		50		%
SPI_DO Output delay	$t_{O\_DO\_M}$	Master mode	2		4	ns
SPI_DI Setup time	$t_{SU\_DI\_M}$	Master mode	9/4 (Note 1)			ns
SPI_DI Hold time	$t_{HD\_DI\_M}$	Master mode	5			ns
SPI_CLK clock (I/O = 3.3 V)	$t_{CLK\_S\_3V3}$	Slave mode			48	MHz
SPI_CLK clock (I/O = 1.8 V)	$t_{CLK\_S\_1V8}$	Slave mode			40	MHz
SPI_CS Setup time before first transfer	$t_{SU\_CS\_S}$	Slave mode	5			ns

Description	Parameter	Condition	Min	Typ	Max	Unit
SPI_CS Hold time after last transfer	$t_{HD\_CS\_S}$	Slave mode	5			ns
SPI_DO Output delay	$t_{O\_DO\_S}$	Slave mode			8.5	ns
SPI_DI Setup time	$t_{SU\_DI\_S}$	Slave mode	3			ns
SPI_DI Hold time	$t_{HD\_DI\_S}$	Slave mode	2			ns

Note 1 Use next edge decoding.

8.5 SDIO

8.5.1 Introduction

RA6W1 supports an SDIO 3.0 card interface suitable for memory card and I/O card applications with low-power consumption. The SDIO interface supports SPI, 1-bit SD, and 4-bit SD transfer modes at the full clock range of 0 to 80 MHz. The CIS and CSA areas are located inside the internal memory and the SDIO registers (CCCR and FBR) are programmed by the SD host.

Features

- Compliant with SDIO Specification 3.0
- Enhanced power management
- Support Read Wait Control, Suspend/ Resume operations for superior card performance
- Support SPI, 1-bit and 4-bit SD modes
- Support all SDIO form factors including standard, mini, and micro SDIO card
- Embedded SDIO ATA interface code
- Bus Master with Scatter Gather DMA
- Cyclic Redundancy Check (CRC7) (command), CRC16 (data) integrity
- Support direct R/W (IO52) and extended R/W (IO53).

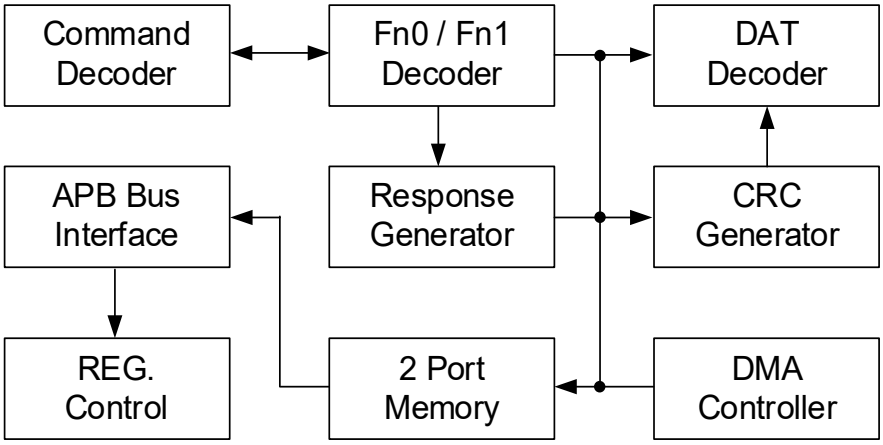


Figure 43. SDIO slave block diagram

The SDIO interface is assigned to specific pins because of the performance requirements of the pin. There are two configurations for the assignment of the SDIO pins (alternate 1 and alternate 2). All pins can also be assigned as either SDIO function or as a peripheral pin through the PPA. The pin configurations for the SDIO interface are shown in Table 47.

Table 47: SDIO pin configuration

Pin name	Pin assignment (Alt 1/Alt 2)	I/O	Description
SDIO_CMD	P1_10/P0_09	I/O	Command line
SDIO_CLK	P1_11/P0_08	I	Input serial clock

Pin name	Pin assignment (Alt 1/Alt 2)	I/O	Description
SDIO_D0	P1_12/P0_10	I/O	Bidirectional data line
SDIO_D1	P1_13/P0_11	I/O	Bidirectional data line
SDIO_D2	P1_14/P0_12	I/O	Bidirectional data line
SDIO_D3	P1_15/P0_13	I/O	Bidirectional data line

Figure 44 shows the timing diagram for the SDIO slave.

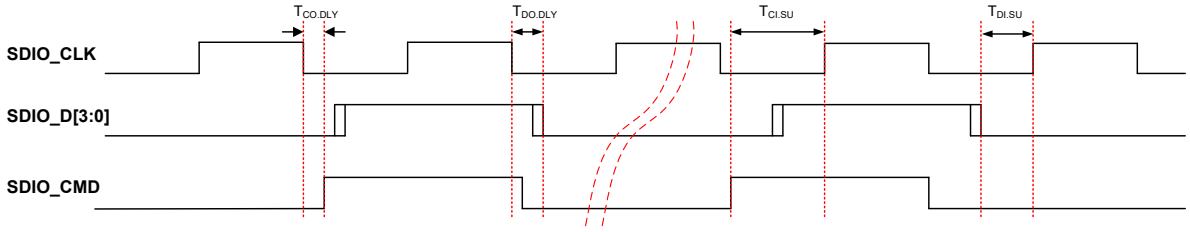


Figure 44. SDIO slave timing diagram

Table 48 lists the timing parameters for the SDIO slave.

Table 48: SDIO slave timing parameters

Parameter	Symbol	Min	Typ	Max	Unit
SDIO_CLK frequency	F_{SCLK}			80	MHz
SDIO_CLK clock duty			50		%
SDIO_CMD input setup time	$T_{CI.SU}$	3			ns
SDIO_CMD output delay time	$T_{CO.DLY}$			11 (Note 1)	ns
SDIO_D[3:0] input setup time	$T_{DI.SU}$	3			ns
SDIO_D[3:0] output delay time	$T_{DO.DLY}$			11 (Note 1)	ns

Note 1 SDIO signals can set previous output from half cycle.

The SDIO interface requires pull-up resistors to be connected between the signal lines and the supply to enable communication.

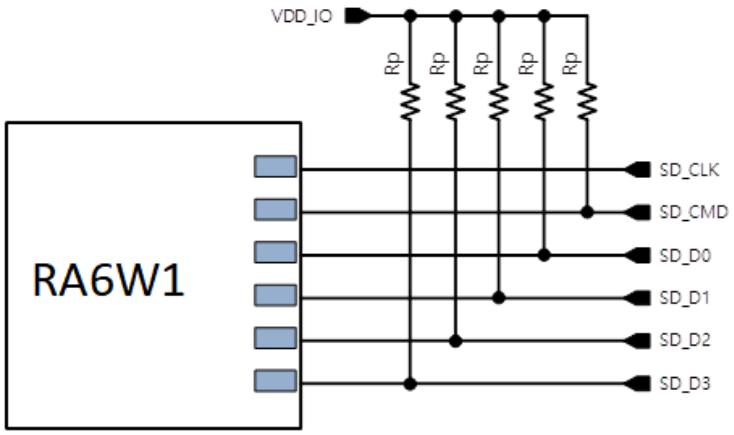


Figure 45. SDIO pull-up resistor

Pull-up resistor values may vary based on the board layout.

8.6 SD/eMMC Host Controller

8.6.1 Introduction

The SD/eMMC host interface of the RA6W1 provides access to SD or eMMC memory cards. The SD/eMMC host interface supports a 4-bit data bus with a maximum clock rate of 80 MHz giving a maximum data rate of 640 Mbps for octa mode, and 320 Mbps for quad mode.

Figure 46 shows the block diagram of the SD/eMMC host interface including the control register, clock control, command/response pipe, data pipe, and AHB master interface blocks.

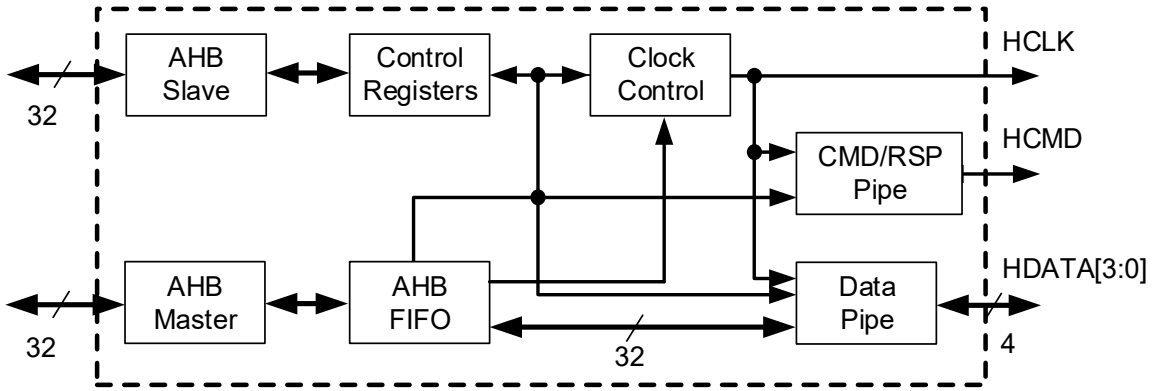


Figure 46. SD/eMMC block diagram

Figure 47 shows the timing diagram for the SD/eMMC master.

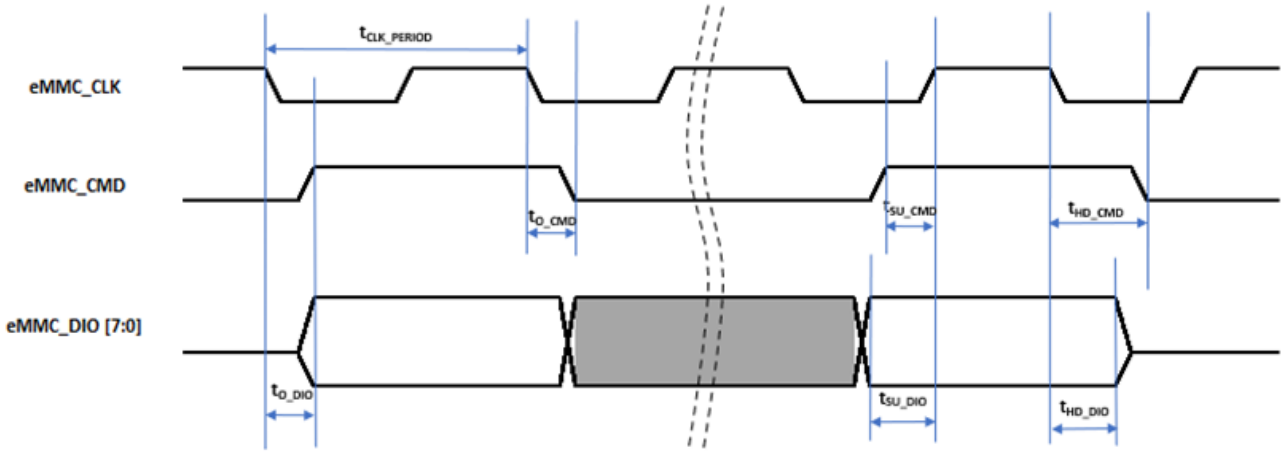


Figure 47. SD/eMMC host timing diagram

Table 49 lists the timing parameters for the SD/eMMC master.

Table 49: SD/eMMC host timing parameters

Description	Parameter	Min	Typ	Max	Unit
eMMC_CLK clock frequency	t_{CLK_M}			80	MHz
eMMC_CLK period	t_{CLK_PERIOD}	12.5			ns
eMMC_CLK duty	t_{CLK_DUTY}		50		%
eMMC_CMD Output delay	t_{O_CMD}	1		2	ns
eMMC_DIO Output delay	t_{O_DIO}	1		2	ns
eMMC_CMD Setup time	t_{SU_CMD}	3			ns
eMMC_CMD Hold time	t_{HD_CMD}	0			ns
eMMC_DIO Setup time	t_{SU_DIO}	2			ns

Description	Parameter	Min	Typ	Max	Unit
eMMC_DIO Hold time	t_{HD_DIO}	0			ns

The SD/eMMC interface is assigned to specific pins because of the performance requirements of the pin. There are two configurations for the assignment of the SD/eMMC pins (alternate 1 and alternate 2). All pins can also be assigned as either SD/eMMC function or as a peripheral pin through the PPA. The pin configurations for the SD/eMMC interface are shown in Table 50.

Table 50: SD/eMMC pin configuration

Pin name	Pin assignment (Alt 1/Alt 2)	I/O	Description
eMMC_CMD	P1_10/P0_09	O	Command line
eMMC_CLK	P1_11/P0_08	O	Output serial clock
eMMC_DIO0	P1_12/P0_10	I/O	Bi-directional data line
eMMC_DIO1	P1_13/P0_11	I/O	Bi-directional data line
eMMC_DIO2	P1_14/P0_12	I/O	Bi-directional data line
eMMC_DIO3	P1_15/P0_13	I/O	Bi-directional data line
eMMC_DIO4	P1_00/P0_04	I/O	Bi-directional data line
eMMC_DIO5	P1_01/P0_05	I/O	Bi-directional data line
eMMC_DIO6	P1_02/P0_06	I/O	Bi-directional data line
eMMC_DIO7	P1_03/P0_07	I/O	Bi-directional data line

8.7 Octa/Quad SPI Flash Controller – with Secure XIP

8.7.1 Introduction

The Octa/Quad SPI Controller (OQSPIC) provides a low pin count interface to standard Serial Peripheral Interface (SPI) and a high-performance Dual/Quad/Octa SPI Interface.

In Automatic mode, the OQSPIC provides transparent access octa/quad flash memory for Execute-In-Place (XIP). In combination with the CPU cache, it provides comparable performance to executing code from standard parallel Flash.

In Manual mode, the serial Flash memory can be accessed by the memory mapped registers of OQSPIC. All instructions supported by Flash memory can be programmed using the registers of the OQSPIC.

Features

- SPI modes:
  - Single: Data transfer using two unidirectional pins
  - Dual: Data transfer using two bidirectional pins
  - Quad: Data transfer using four bidirectional pins
  - Octa: Data transfer using eight bidirectional pins (VFBGA package only)
- Auto mode: up to 64 MB transparent code access for XIP (Execute-In-Place) and Data access with 3-byte and 4-byte addressing modes.
- Manual mode: Direct register access using the QSPIC register file.
- Up to 80 MHz QSPI clock. Clock modes 0 and 3. Master mode only.
- Vendor independent Instruction Sequencer.
- Support for single access and high-performance burst mode in combination with the cache controller (in Auto mode).
- Use of special read instruction for a specific (programmable) wrapping burst access.
- Erase suspend/resume to support for code and data storage.
- Decrypt on-the-fly (AES-256b-CTR) capability while in Auto mode operation.

- Retention mode for I-cache memory.

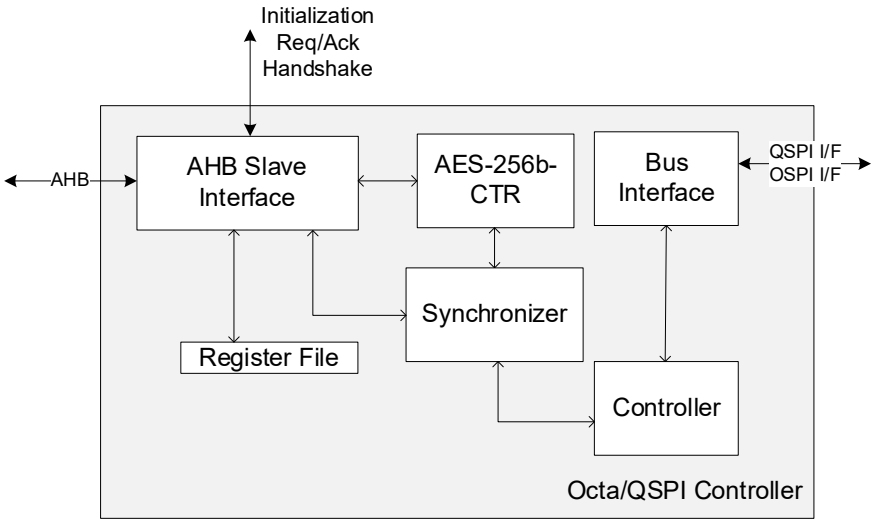


Figure 48. OQSPI flash controller block diagram

The OQSPIC implements all protocols related to the functionality of Flash memory. It contains a finite state machine (FSM) that generates all necessary signaling to the OQSPI bus and realizes all features of the Auto mode operation. Moreover, it manages all data transfers between the two interfaces (the AHB and the OQSPIC).

The Bus Interface block controls the OQSPIC signals at the lowest level while the Synchronizer implements "stretching" or "shortening" of the signals that cross two clock domains.

8.7.1.1 Interface

The OQSPIC interface is assigned to specific pins because of the performance requirements of the pin. All pins can be assigned as either OQSPI function or as a peripheral pin through the PPA. The pin configurations for the OQSPIC interface are shown in Table 51.

Table 51: OQSPI pin configuration

Pin name	Pin assignment	I/O	Description
OQSPI_CLK	P1_08	O	Output serial clock
OQSPI_CS	P1_09	O	Active low output Chip select
OQSPI_D0	P1_04	I/O	▪ MOSI (output) in single SPI mode ▪ D0 (bidirectional) in Quad/Octa SPI mode
OQSPI_D1	P1_05	I/O	▪ MISO (input) in single SPI mode ▪ D1 (bidirectional) in Quad/Octa SPI mode
OQSPI_D2	P1_06	I/O	▪ WPn Write Protect output in single SPI mode ▪ D2 (bidirectional) in Quad/Octa SPI mode
OQSPI_D3	P1_07	I/O	▪ HOLDn/Resetn output in Single SPI mode ▪ D3 (bidirectional) at Quad/Octa SPI mode
OQSPI_D4	P1_00	I/O	D4 (bidirectional) at Octa SPI mode
OQSPI_D5	P1_01	I/O	D5 (bidirectional) at Octa SPI mode
OQSPI_D6	P1_02	I/O	D6 (bidirectional) at Octa SPI mode
OQSPI_D7	P1_03	I/O	D7 (bidirectional) at Octa SPI mode

The OQSPIC drives all data pins constantly except for the case when a read is performed. The time for changing the direction of the pads is at least 1.5 x OQSPIF_CLK (OQSPIF_CLK being the clock that the Flash operates on). In this way, data lines are always terminated, thus reducing unnecessary power consumption.

The default state of the OQSPIF_IOx pins is 1. This state is applied to the pins as soon as the OQSPIC clock is enabled even if no access to the external Flash has yet been triggered. This value is valid only after the OQSPIF_CS is pulled low (an access to the external Flash occurs).

8.7.1.2 SPI Modes

The OQSPIC supports the following SPI standards:

- Single: Data transfer using two unidirectional pins. The OQSPIC supports communication with any single/dual/quad or octa SPI Flash memory. However, the Single SPI interface does not support bus modes 1 and 2, full-duplex communication, or any SPI Slave mode.
- Dual: Data transfer using two bidirectional pins.
- Quad: Data transfer using four bidirectional pins.
- Octa: Data transfer using eight bidirectional pins.

8.7.1.3 Access Modes

The serial Flash connected to the OQSPIC can be accessed in one of the modes:

- Auto mode
- Manual mode.

These modes are mutually exclusive. The serial Flash can operate only in one of the two modes. In Auto mode, 3-byte and 4-byte addressing modes are supported. With OQSPIF_CTRLMODE_REG[QSPIC_USE_32BA] = 0, up to 16 MB OQSPIC (3-byte addressing) can be accessed. If OQSPIF_USE_32BA = 1, 4-byte addressing is enabled for accessing up to 64 MB OQSPIC Flash.

8.7.1.3.1 Auto Mode

In Auto mode, read access from the serial Flash memory is fully transparent to the CPU. A read access at the interface is translated by the OQSPIC into the respective SPI bus control commands needed for the Flash memory access. When the Auto Mode is disabled, any access (reading or writing) is ignored. When Auto mode is enabled, only read access is supported. Write access causes hard faults. A read access can be single access, incremental burst, or wrapping burst. Wrapping burst is supported even when the Flash device does not support any special instruction for wrapping burst. Special read instruction can be used for specific (programmable) wrapping burst access. When Flash supports special instruction for wrapping burst access, it reduces access time (less wait states). For maximizing the utilization of the bus and minimizing the number of wait states, it is recommended to use burst access. However, non-sequential random access is supported with the cost of more wait states.

8.7.1.3.2 Manual Mode

In Manual mode, the Flash memory is controlled by a register file. All instructions that are supported by Flash memory can be programmed using the register file. Moreover, the mode of interface (SPI, Dual SPI, Quad SPI, or Octa SPI) and the mode of operation (Auto or Manual mode) can be configured using this register file. The register file supports the following data sizes for reading and writing accesses: 8 bits, 16 bits, and 32 bits.

8.7.1.4 Endianness

The OQSPIC operates in little-endian mode. For 32-bit or 16-bit access (for read and write operations) to a serial Flash memory, the least significant byte comes first. For 32-bit access, the byte ordering is: data [7:0], data [15:8], data [23:16], data [31:24] while for 16-bit access the byte ordering is: data [7:0], data [15:8].

8.7.1.5 Erase Suspend/Resume

The OQSPI Flash can be used for Data Storage, combining the EEPROM functionality and Program storage on one single device. For this purpose, the Octa/Quad SPI ERASE/SUSPEND ERASE/RESUME commands are automatically executed as shown in Figure 49.

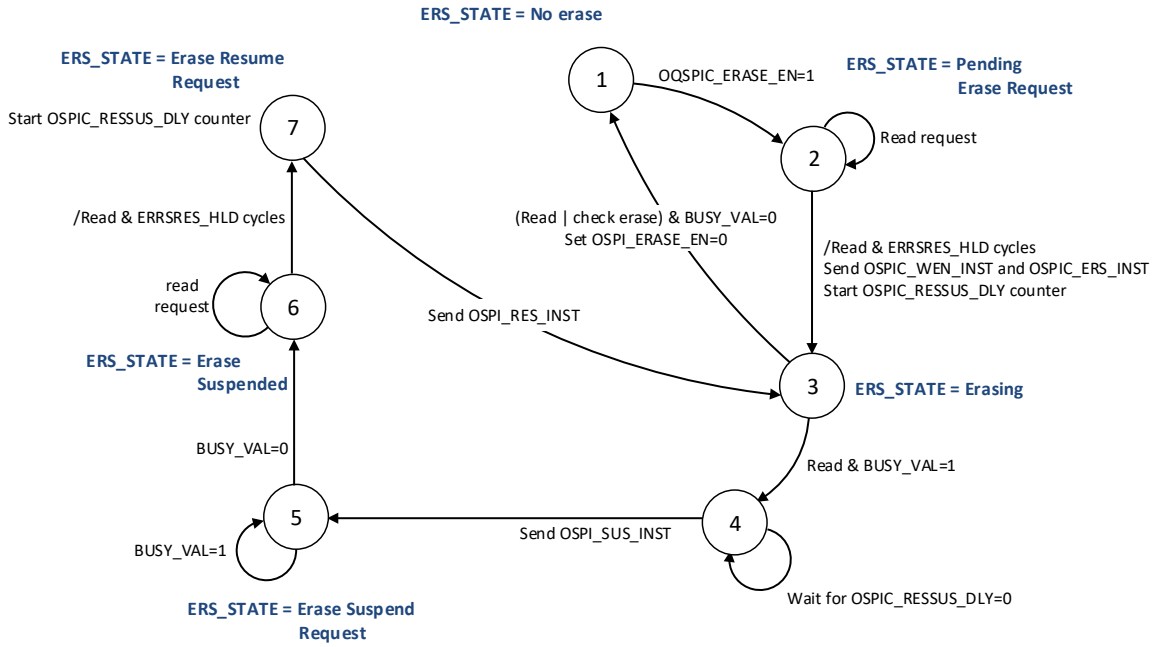


Figure 49. Erase suspend/resume in Auto mode

8.7.1.6 On-the-fly Decryption

The OQSPIC supports decryption of the data retrieved from the Flash device, only when in Auto mode. The Flash contents should be encrypted already, using the same algorithm.

The address range, which is decrypted automatically by the controller, is defined in the OQSPIC by using two configuration registers, one for the start address (OQSPIF_CTR_SADDR_REG register), and the other for the end address (OQSPIF_CTR_EADDR_REG register). The defined address range is 1024 bytes aligned. All addresses, which are outside of this range, are not automatically decrypted but fetched as is.

The on-the-fly decryption feature is based on the CTR mode of the AES encryption algorithm. The AES algorithm process blocks of 128 bits. That means, the input and output block of the AES is 128 bits or 16 bytes and as a result, data to be processed by the algorithm are fragmented into blocks of 16 bytes. The key size that is used for the AES algorithm is 256 bits.

For the AES-256-CTR algorithm, only the cipher part of the AES algorithm is required. The general idea of the encryption process is based on the encryption of a 128-bit counter block (CTR). The initial value of the CTR is labeled as CTR0. The first counter block (CTR0) is encrypted with the help of the AES cipher, and the encrypted result is XORed with the first 16 bytes of the plaintext data (P0) to be encrypted. The counter block is incremented by one (CTR1 = CTR0 + 1) and is encrypted again. The result is XORed with the next 16 bytes of the plaintext (P1) and so on until all plaintext data is encrypted.

The AES CTR decryption is the same process as encryption. By XORing again, the ciphertext with the same encrypted counter value, the plaintext is retrieved. The decryption process can be described by equations:

For j = 1 to m, do $CTR_j = CTR_{j-1} + 1$

For j = 0 to m, do $P_j' = AES\_CIPH_k(CTR_j) \oplus C_j$

Because access to the Flash memory is random, the structure of the CTR block is selected to simplify the process. The total size of the counter block is 128 bits or 16 bytes, namely: $CTRB_0$, $CTRB_1$, $CTRB_2$, $CTRB_3$, …, $CTRB_{14}$, $CTRB_{15}$.

The first 8 bytes ($CTRB_0$ - $CTRB_7$) of the counter block comprise the NONCE value and are programmed in the QSPI Controller in configuration registers (OQSPIF_CTR_NONCE_*_REG). This is typically a random value and is the same for all the CTR_i blocks.

The next four bytes of the counter block ($CTRB_8$-$CTRB_{11}$) are always zero.

The last four bytes of the counter block ($CTRB_{12}$-$CTRB_{15}$) are produced automatically by the hardware based on the 32-bit address offset OFFSET_ADDR [31:0] inside the encrypted range, where the data that should be decrypted are placed. If FLASH_ADDR[31:0] is the absolute address of a specific byte inside the encrypted

range, the offset address is OFFSET_ADDR = FLASH_ADDR – OQSPIF_CTR_SADDR_REG[OSPIC_CTR_SADDR]. The four least significant bits of the address offset are truncated and the four most significant bits of the CTRB12 are padded with zeros. Thus, the zero value in bytes $CTRB_{12}$-$CTRB_{15}$ of the CTR is used for the first 16 bytes of the address range that is encrypted, the value 1 is used for the second 16 bytes of the address range, and so forth.

The final form of the counter block is the following:

{64 bits NONCE, 32 bits 0x0, 4 bits 0x0, OFFSET_ADDR[31:4]}

The four least significant bits of the address offset OFFSET_ADDR[3:0] define which of the AES_CIPHk(CTR_j) byte should be used for the decryption of a specific byte of the encrypted block C_i.

In this way, the CTR block, which should be used for the decryption of a specific byte, can be calculated immediately by the address of the byte in the Flash and the start address of the encrypted range. This counter block supports up to 4 GB data that covers the maximum supported size for the Flash devices.

8.8 Quad SPI RAM/Flash Controller – PSRAM

8.8.1 Introduction

The Quad SPI Controller (QSPIC) provides a low pin count interface to serial QSPIC Flash and RAM devices. The QSPIC supports the standard SPI and a high-performance Dual/Quad SPI Interface. The QSPIC RAM feature provides a low-cost RAM extension for infrequently used data and can be used in combination with the data cache controller.

The QSPIC automatically generates all the control signals for the QSPI bus needed to access data from the serial quad memory. The controller has a vendor independent register file that provides a rich set of control fields for a wide range of Flash and RAMs.

The QSPIC provides transparent memory mapped RAM access.

Features

- SPI modes:
  - Single: Data transfer through two unidirectional pins
  - Dual: Data transfer through two bidirectional pins
  - Quad: Data transfer through four bidirectional pins.
- Auto mode: up to 64 MB memory mapped Read/Write Data access with 3-byte and 4-byte addressing modes
- Manual mode: Direct register access using the QSPIC register file
- Clock modes 0 and 3. Master mode only
- Vendor independent Instruction Sequencer
- In Auto mode, the Flash control signals are fully programmable
- Use of special read instruction for a specific (programmable) wrapping burst access
- Erase suspend/resume to Support for Code and Data storage
- Data Cache controller can be enabled or bypassed.

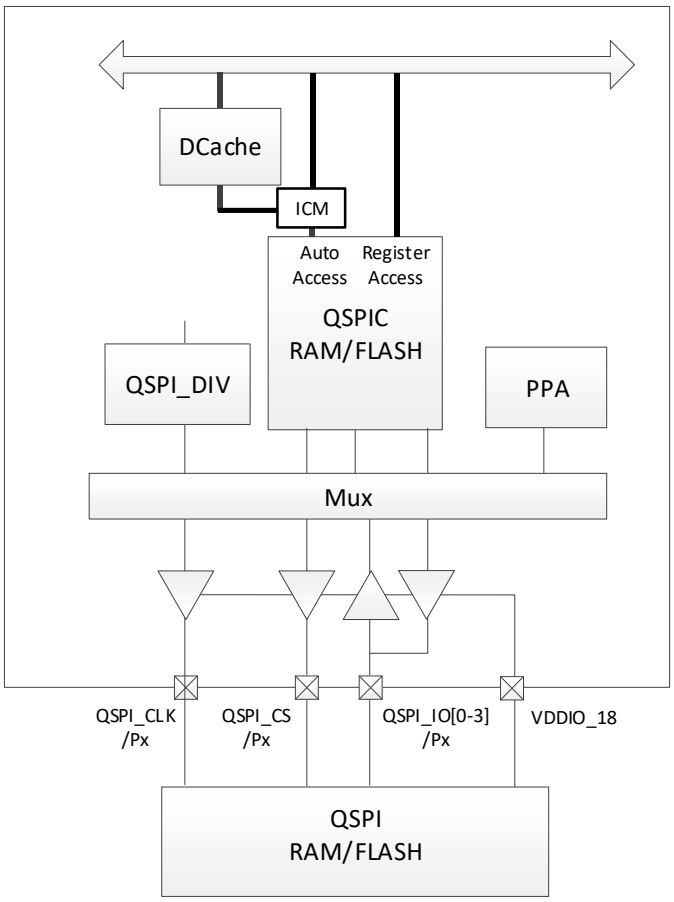


Figure 50. Quad SPI RAM/Flash controller

8.8.2 Interface

The QSPIC interface is assigned to specific pins because of the performance requirements of the pin. All pins can be assigned as either QSPIC function or as a peripheral pin through the PPA. Table 52 shows the pin configurations for the QSPIC interface.

Table 52: QSPI pin configuration

Pin name	Pin assignment	I/O	Description
QSPI_CLK	P0_08	O	Output serial clock
QSPI_CS	P0_09	O	Active low output Chip select
QSPI_D0	P0_10	I/O	MOSI (output) in single SPI mode D0 (bidirectional) in Quad SPI mode
QSPI_D1	P0_11	I/O	MISO (input) in single SPI mode D1 (bidirectional) in Quad SPI mode
QSPI_D2	P0_12	I/O	WPn Write Protect output in single SPI mode D2 (bidirectional) in Quad SPI mode
QSPI_D3	P0_13	I/O	HOLDn/Resetn output in Single SPI mode D3 (bidirectional) at Quad SPI mode

8.8.3 SPI Modes

The QSPIC supports the following SPI standards:

- Single: Data transfer through two unidirectional pins. The QSPIC supports communication with any single/dual or Quad SPI Flash memory. In contradiction to the standard SPI interface, the supported Single SPI interface does not support the bus modes 1 and 2, full-duplex communications, or any SPI Slave mode.

- Dual: Data transfer through two bidirectional pins.
- Quad: Data transfer through four bidirectional pins.

8.8.4 Access Modes

Access to a serial memory (Flash or RAM) connected to the QSPIC can be done in two modes:

- Auto mode
- Manual mode.

These modes are mutually exclusive. The serial memory can be controlled only in one of the two modes. The registers which control the mode of operation can be used at any time.

In Auto mode, 3-byte and 4-byte addressing modes are supported. With QSPIC_USE_32BA=0, up to 16 MB serial memory (3-byte addressing) can be accessed. If QSPIC_USE_32BA = 1, the 4-byte addressing is enabled for accessing up to 64 MB serial memory.

8.8.4.1 Auto Mode Flash Access

In Auto mode (QSPIC_AUTO_MD=1), the read access to a serial Flash memory is performed in a fully transparent way through the SPI bus. A read access to the memory space, where the external memory is mapped, is translated by the controller to the respective SPI bus command sequence, which is needed for the retrieving of the requested data from the serial Flash memory.

When the Auto mode is disabled (QSPIC_AUTO_MD =0), any access (reading or writing) to the mapped memory space is ignored by the controller.

Only read access is supported when the connected external device is a Flash memory (QSPIC_SRAM_EN = 0). Write access causes a hard fault at the CPU.

The read access can be single access or incremental burst or wrapping burst access. The wrapping burst is supported even when the controlled serial Flash does not support any special instruction for wrapping burst. Special read instruction can be used for specific (programmable) wrapping burst access. When a serial Flash supports a special instruction for wrapping burst access, this feature saves access time (less wait states). For maximizing the utilization of the bus and minimizing the number of wait states, it is recommended to be used burst accesses. However, non-sequential random access is supported with the cost of more wait states.

8.8.4.2 Auto Mode RAM Access

When it is connected to a serial RAM device, the QSPIC controller can provide read/write functionality.

The special configuration register must be programmed to enable the RAM functionality (QSPIC_SRAM_EN=1). If the external device is a Flash, the Auto mode must also be enabled (QSPIC_AUTO_MD=1). If the Auto mode is disabled (QSPIC_AUTO_MD=0), any access (reading or writing) in the memory space, where the external device has been mapped, is ignored by the QSPIC.

The read access in the memory space of the external serial RAM, is done in a fully transparent way through the QSPI bus. The capability of the controller to handle the various types of read accesses, is the same as the Flash device. Single access, incremental burst or wrapping burst are all supported.

Write access to the memory space where the external memory is mapped, does not cause a hard fault to the Cortex-M0. In the contrary, the write access is interpreted by the QSPIC in the respective QSPI bus protocol and the write data is stored in the external RAM device.

The controller is capable of handling write accesses of all kinds of burst: single access, incremental burst or wrapping burst access. The throughput that can be achieved varies depending on the burst length, the word width, the cost of the protocol of the external memory device, and the frequency of the QSPI clock.

Burst access provides the highest throughput. The non-sequential random accesses are supported at the cost of more wait states. The maximum throughput that can be achieved depends on the burst length.

8.8.4.3 Manual Mode

In manual mode, the external serial memory is controlled through a register file. All instructions that are supported by the serial memory can be programmed by using the register file. Moreover, the mode of interface (SPI, Dual SPI, Quad SPI) and the mode of operation (Auto or Manual mode), can be configured through this register file. The register file supports the following data sizes for reading and writing accesses: 8 bits, 16 bits, and 32 bits.

8.8.5 Endianness

The QSPI controller operates in little-endian mode. For 32-bit or 16-bit access (for read and write operations) to a serial memory, the least-significant byte comes first. For 32-bit access, the byte ordering is: data [7:0], data [15:8], data [23:16], data [31:24] and for 16-bit access the byte ordering is: data [7:0], data [15:8].

8.8.6 Erase Suspend/Resume

A QSPI Flash memory can be used for data storage, combining the EEPROM functionality with Program storage in one single device.

For this purpose, the QSPI ERASE/SUSPEND ERASE/RESUME are automatically executed as shown in Figure 51.

To store data in QSPI Flash memory, the sector designated for storage must be erased first. The ERASE/SUSPEND ERASE/RESUME process is only meaningful if the external device is a serial Flash memory (QSPIC_SRAM_EN=0).

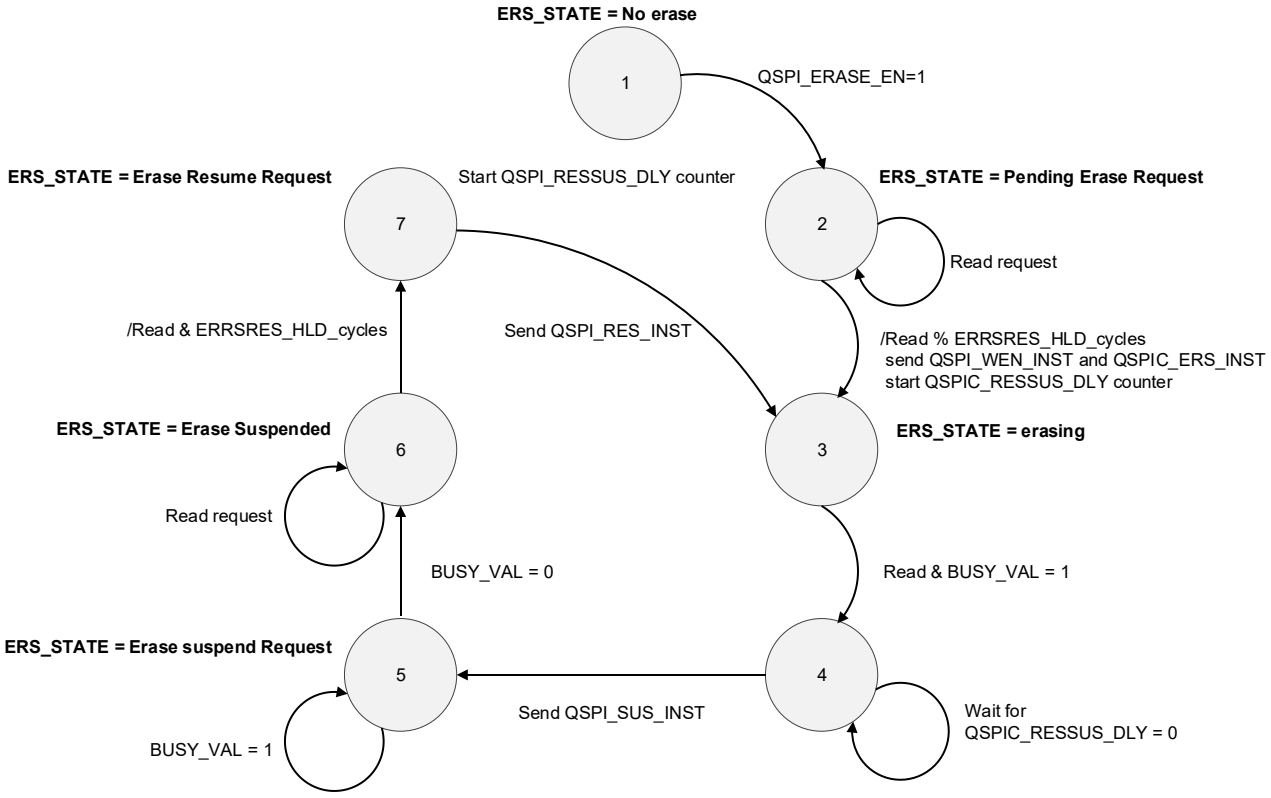


Figure 51. Erase suspend/resume in Auto mode

NOTE
QSPI_RESSTS_DLY is counted with the QSPI_CLK, so before changing the QSPI_CLK, make sure that QSPI_RESSTS_DLY is set large enough to meet the timing parameter requirements.

8.8.7 Low Power Considerations

To reduce the power dissipation in the QSPI Flash, the QSPI_CLK must always be the highest possible system clock to keep the burst access to the Flash as short as possible. The CPU must run as slow as possible for minimum power.

For the lowest power with slow CPU (for example, 2 MHz) and high QSPI_CLK (for example, 40 MHz) bit QSPIC_CTRLMODE_REG[QSPIC_FORCENSEQ_EN] must be set to 1. This enables split burst mode, reducing the power dissipation during active burst only, while disabling the Flash when the burst is done compared to high efficiency burst. These two modes are explained in the following two figures:

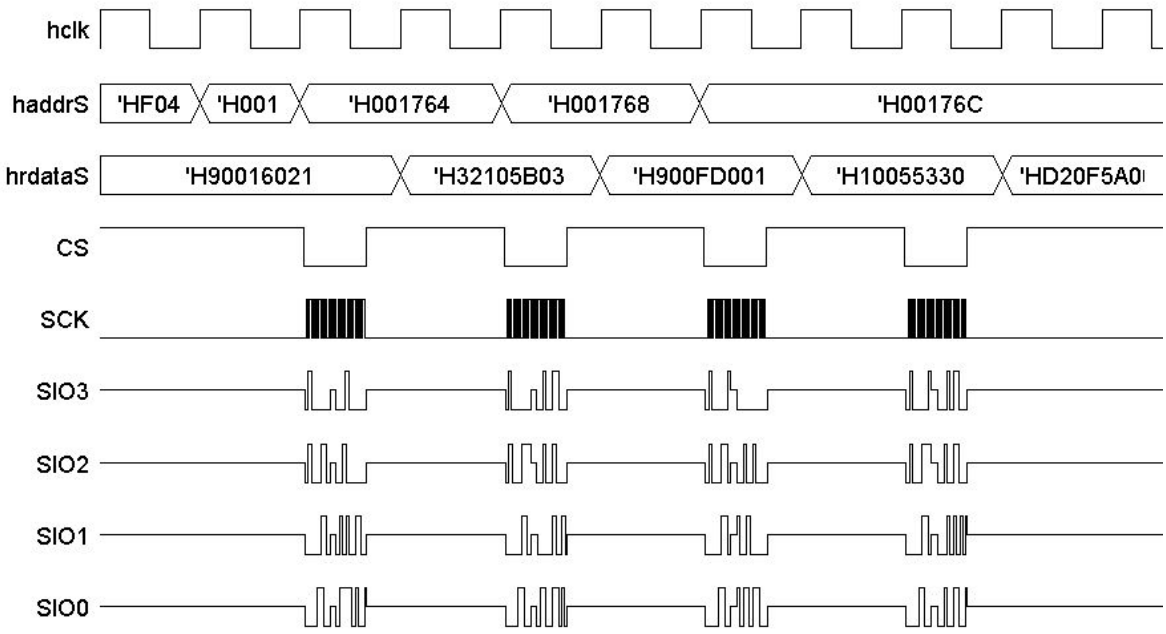


Figure 52. QSPI split burst timing for low power (QSPI_FORENSEQ_EN = 1)

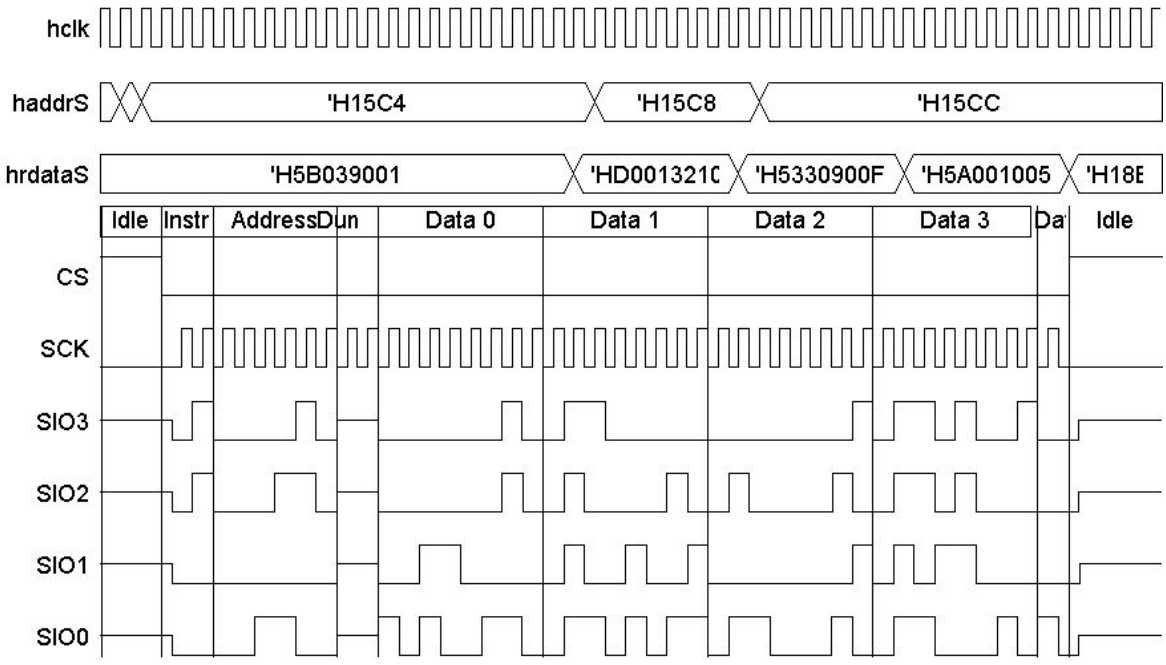


Figure 53. QSPI burst timing for high performance (QSPI_FORENSEQ_EN = 0)

If QSPI_FORCENSE_EN = 0, the QSPIC reads data in a burst address1/extra/dummy/data1, data2, dataN, keeping the QSPI_CS low during the complete burst. If set to 1, the burst is split into non-sequential accesses address1/extra/dummy/data1, address2/extra/dummy/data2, making the QSPI_CS high between the accesses.

8.9 General Purpose Timers/PWMs

8.9.1 Introduction

The Timers block contains eight 32-bit wide programmable timers which include PWM capabilities. All Timers reside in the system power domain.

Features

- Eight 32-bit general purpose timers
- Pulse Width Modulated signal (PWM), one per timer block
- PWM start synchronization
- Dual GPIO multi event capture with interrupt generation (4 GPIOs for Timer)
- One shot mode generated by a GPIO toggle or a register write
- 5-bit clock pre-scaler with selectable clock source XTAL32K or XTAL40M
- Up/down counting capability with free running mode
- Active while system is in Sleep modes 4 and 5
- Dedicated interrupt line per timer, common capture event interrupt
- GPIO edges (positive or negative) counting function
- Timers 1 and 5 support extra functions:
  - Single GPIO single event capture (only first edge)
  - Automatic switching from one-shot to counter mode capability.
  - PWM sync with other timers in the timer group:
    - Timer 1 is grouped with timers 2, 3, and 4
    - Timer 5 is grouped with timers 6, 7, and 8.

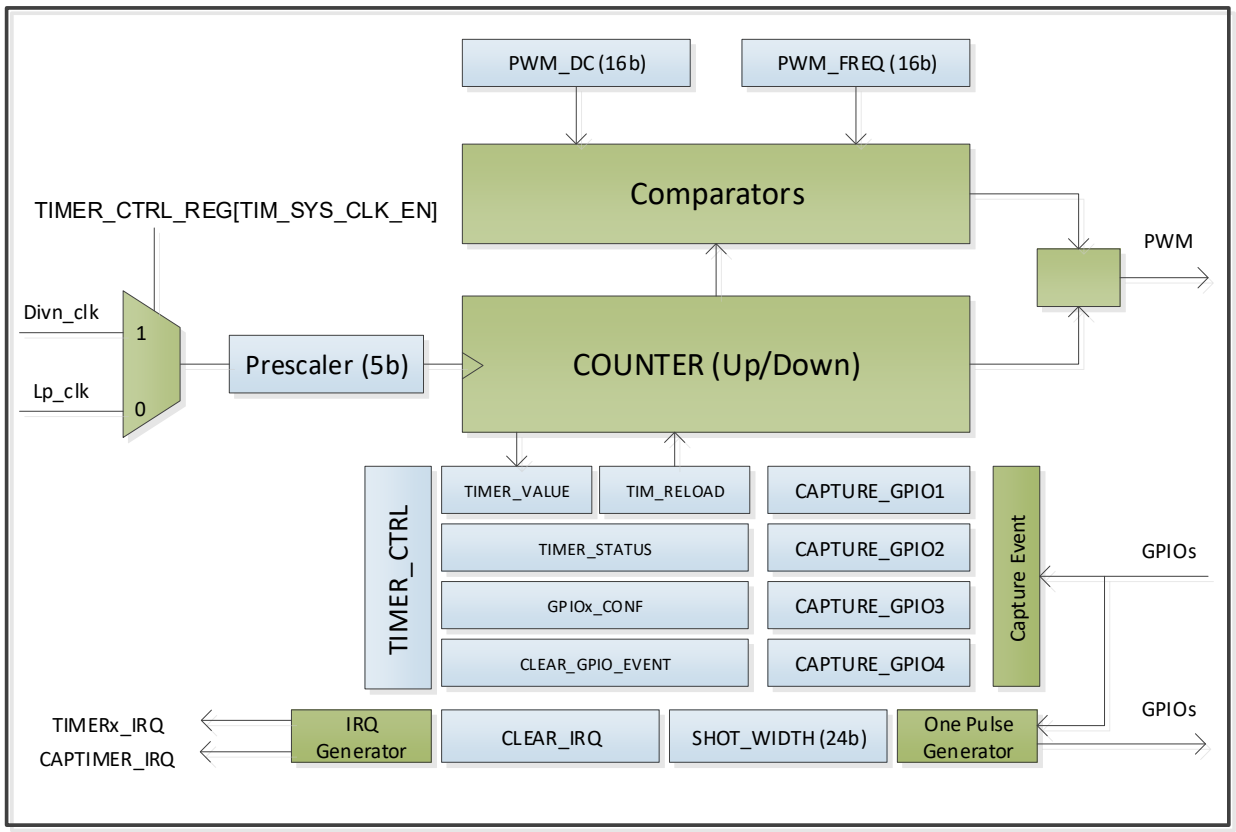


Figure 54. General purpose timers block diagram

There are eight 32-bit timers (Timers 1 – 8) with timer 1 and timer 5 supporting two extra capture channels. The general timer block diagram is shown in Figure 54. All timers reside in the system power domain (PD_SYS) and support the features listed in Table 53.

Table 53: Timer features overview

Feature	Timer (1)	Timer (5)	Timer (2/3/4/6/7/8)
Counter width	32-bit	32-bit	32-bit
Free-Running Counter, PWM generation, and Edge detection counter	✓	✓	✓
PWM start synchronization (Note 1)	✓	✓	✓
Event capturing channels	4	4	2
Event capture IRQ	×	✓	×
Timer IRQ	✓	✓	✓
One-Shot	✓	✓	✓
One-Shot with auto switch	✓	✓	×
First event capture	✓	✓	×

Note 1 PWM sync is grouped Timer 1/2/3/4 and Timer 5/6/7/8.

Each timer can be clocked with either the 40 MHz clock source or the 32 kHz clock for low power operation. Also, for each timer, a five-bit prescaler can be used to further reduce the clock resulting in a minimum timer frequency of 1 kHz and a maximum of 40 MHz.

8.9.2 Timer Modes of Operation

8.9.2.1 Free-Running Counter

Each timer block has a configurable free running up/down counter that triggers an interrupt when passing from the pre-configured value and immediately returns to 0 or pre-load depending on whether it is counting up or down. The timers can be instructed to continuously count upwards and wrap around when reaching its boundaries.

8.9.2.2 PWM Generation

Each timer has a Pulse Width Modulation (PWM) output (TIMx_PWM) which can be mapped to any available GPIO pin without affecting the actual counter while running.

The PWM frequency is defined by further dividing the prescaled clock by a maximum of 2^{16}+1. The duty cycle duration of the generated PWM signal can be configured in TIM_PWM_FREQ steps. If the minimum PWM frequency is selected, only 50% duty cycle is possible.

PWM synchronization is supported for all timers. It is assumed that all the timers are running on the same clock source while PWM frequency and duty cycle might be different.

8.9.2.3 Event Capturing

The GPIO multi event capture provides latching snapshots of the free-running clock into two registers. The difference of these two registers indicates the duration between two trigger events.

NOTE
Timer 1 and Timer 5 have four capture pairs.

A single event capture latches only a single snapshot, in the first capture register, with the first trigger (after programming) of the capture event. Subsequent edges of the triggering GPIO do not change the value of the first capture register and are ignored by the capturing circuitry.

8.9.2.4 One Shot

Upon an input trigger from a GPIO toggle or a register write, a separate GPIO serves as an output, delivering a pulse of configurable width. This feature implements a PWM reply in hardware. There are four different selections for the input trigger:

- External GPIO (TIMx_1SHOT assigned through PPA)
- Software trigger (write to the timer trigger register)
- Either of the two triggers
- None of the two triggers.

Timer 1 and Timer 5 support automatic switching from one-shot to counter mode without the CPU involvement. If no start value is programmed, an interrupt is immediately issued to the CPU.

8.9.2.5 GPIO Pulse Counter

Every timer block supports counting pulses from a programmable GPIO. This operation is available when the system is in Sleep 4 and Sleep 5 modes. The frequency range of the edge train may never exceed 80 MHz for proper edge detection and counter update.

If the threshold is reached, an interrupt is generated, and the counter is automatically reset to zero.

8.9.3 Pin Configuration

Each timer has two I/Os which can be assigned to any available GPIO pin through the PPA function. The pin definitions for the Timer interfaces are shown in Table 53.

8.10 GPIOs and Programmable Pin Assignment

The functions assigned to the GPIO pins are fully configurable and are controlled by the Programmable Pin Assignment (PPA).

- **Features**
- Fully Programmable Pin Assignment
- Selectable 25 kΩ pull-up/pull-down resistors per pin up to selected voltage rail
- Programmable open-drain functionality
- Selectable drive strength: 2 mA, 4 mA, 8 mA, 14 mA
- Fixed assignment for ADCx, QSPI, OQSPI and SWD pins
- Pin state is maintained when the system enters low-power sleep 4 and 5 modes.

The PPA provides a multiplexing function for the I/O pins of the on-chip peripherals. Any of the peripheral input or output signals can be freely mapped to any available GPIO port.

The list of peripheral I/Os that can be mapped through the PPA are shown in Table 54.

Table 54: Pin function list

Function ID	Function name	Pin type (Note 1)
0	GPIO	(I/O)
1	UART_RX	(IA) (Note 2)
2	UART_TX	(OA) (Note 3)
3	UART_CTSN	(IA)
4	UART_RTSN	(OA)
5	UART_TXDOE	(OA)
6	UART1_RX	(IA)
7	UART1_TX	(OA)
8	UART1_CTSN	(IA)
9	UART1_RTSN	(OA)
10	UART1_TXDOE	(OA)
11	UART2_RX	(IA)
12	UART2_TX	(OA)
13	UART2_CTSN	(IA)

Function ID	Function name	Pin type (Note 1)
14	UART2_RTSN	(OA)
15	UART2_TXDOE	(OA)
16	SPI_DI	(IA)
17	SPI_DO	(OA)
18	SPI_CLK	(I/O)
19	SPI_CSN0	(I/O)
20	SPI_CSN1	(OA)
21	SPI2_DI	(IA)
22	SPI2_DO	(OA)
23	SPI2_CLK	(I/O)
24	SPI2_CSN0	(I/O)
25	SPI2_CSN1	(OA)
26	I2C_SCL	(I/O)
27	I2C_SDA	(IO-OD)
28	I2C2_SCL	(IO-OD)
29	I2C2_SDA	(I/O-OD)
30	Analog (Note 4)	(ADC)
31	PCM_DI	(I)
32	PCM_DO	
33	PCM_FSC (Note 5)	(I/O)
34	PCM_CLK (Note 6)	(I/O)
35	DMICA_DI	(I)
36	DMIC_CLK	(I/O)
37	MCLK	
38	TIM_PWM	(OA)
39	TIM2_PWM	(OA)
40	TIM3_PWM	(OA)
41	TIM4_PWM	(OA)
42	TIM5_PWM	(OA)
43	TIM6_PWM	(OA)
44	TIM7_PWM	(OA)
45	TIM8_PWM	(OA)
46	TIM_1SHOT	(OA)
47	TIM2_1SHOT	(OA)
48	TIM3_1SHOT	(OA)
49	TIM4_1SHOT	(OA)
50	TIM5_1SHOT	(OA)
51	TIM6_1SHOT	(OA)
52	TIM7_1SHOT	(OA)
53	TIM8_1SHOT	(OA)
54	CLOCK	Note 7

Function ID	Function name	Pin type (Note 1)
55	FEM_BS	(O)
56	FEM_CS	(O)
57	FEM_CTRL0	(O)
58	FEM_CTRL1	(O)
59	FEM_CTRL2	(O)
60	BT_COEX_CBT	(O)
61	BT_WLAN_ACT	(O)
62	BT_ACT	(I)
63	BT_PRI	(I)
64	RF_SW1	(O)
65	RF_SW2	(O)
66	EXT_INTR	(O)
99	SWCLK	(I)
100	SWDIO	(I/O)
101	WPROTECT	(I)
102	CDETECT	(I)
103	ZB_WLAN_ACT	(O)
104	ZB_ACT	(I)
105	ZB_PRI	(I)
106	BTCOEX_ASC0	(O)
107	BTCOEX_ASC1	(O)
108	BTCOEX_ASC2	(O)

Note 1 (I): Input, (I/O): Input/Output), if nothing mentioned: Output (default).

Note 2 IA: Pad direction is automatically set to input when the function is selected.

Note 3 OA: Pad direction is automatically set to output when the function is selected.

Note 4 For XTAL32K monitoring. See pinout.

Note 5 Also defined as PCM_WCLK.

Note 6 Also defined as PCM_BCLK.

Note 7 See GPIO_CLK_SEL_REG.

Note 8 In other cases, the pad direction can be set through the pin mode register.

Note 9 In output mode and analog mode, the pull-up/down resistors are automatically disabled.

When a pin is configured to function as a GPIO, it has the following configurable features:

- Direction (input/output)
- Push pull/Open drain
- Pull-up/Pull-down
- Drive strength (2 mA, 4 mA, 8 mA, 14 mA)
- Slew rate (Fast/Slow)
- Input selection (CMOS/Schmitt Trigger).

After a power on reset (POR), the default state of the pins is shown in Table 55.

Table 55: Pin configuration

Pin	Support wake-up	Retention group	Power domain	Alternate function 0	Alternate function 1	Alternate function 2	POR default
RST_N			VBAT	RST_N			RST_N
P0_00	Yes	VBAT	VBAT	RTC_WAKE_UP			GPIO
P0_01		VBAT	VBAT	sen_out			sen_out
P0_02		VBAT	VBAT	xtal32k_m			xtal32k_m
P0_03		VBAT	VBAT	xtal32k_p			xtal32k_p
P0_04	ana wake	DIO1_1	VDDIO_DIO1	ADC0		eMMC_DIO4	GPIO
P0_05	ana wake	DIO1_1	VDDIO_DIO1	ADC1		eMMC_DIO5	GPIO
P0_06	ana wake	DIO1_1	VDDIO_DIO1	ADC2		eMMC_DIO6	GPIO
P0_07	ana wake	DIO1_1	VDDIO_DIO1	ADC3	MCLK	eMMC_DIO7	GPIO
P0_08	Yes	DIO1_2	VDDIO_DIO1	QSPIR_CLK	SDIO0_CLK	eMMC_CLK	GPIO
P0_09	Yes	DIO1_2	VDDIO_DIO1	QSPIR_CS	SDIO0_CMD	eMMC_CMD	GPIO
P0_10	Yes	DIO1_2	VDDIO_DIO1	QSPIR_D0	SDIO0_D0	eMMC_DIO0	GPIO
P0_11	Yes	DIO1_2	VDDIO_DIO1	QSPIR_D1	SDIO0_D1	eMMC_DIO1	GPIO
P0_12	Yes	DIO1_2	VDDIO_DIO1	QSPIR_D2	SDIO0_D2	eMMC_DIO2	GPIO
P0_13	Yes	DIO1_2	VDDIO_DIO1	QSPIR_D3	SDIO0_D3	eMMC_DIO3	GPIO
P1_00		FDIO	VDDIO_FDIO	OQSPI_D4	eMMC_DIO4		GPIO
P1_01		FDIO	VDDIO_FDIO	OQSPI_D5	eMMC_DIO5		GPIO
P1_02		FDIO	VDDIO_FDIO	OQSPI_D6	eMMC_DIO6		GPIO
P1_03		FDIO	VDDIO_FDIO	OQSPI_D7	eMMC_DIO7		GPIO
P1_04		FDIO	VDDIO_FDIO	OQSPI_D0			GPIO
P1_05		FDIO	VDDIO_FDIO	OQSPI_D1			GPIO
P1_06		FDIO	VDDIO_FDIO	OQSPI_D2			GPIO
P1_07		FDIO	VDDIO_FDIO	OQSPI_D3			GPIO
P1_08		FDIO	VDDIO_FDIO	OQSPI_CLK			GPIO
P1_09		FDIO	VDDIO_FDIO	OQSPI_CS			GPIO
P1_10	Yes	DIO2	VDDIO_DIO2	eMMC_CMD	SDIO1_CMD		GPIO
P1_11	Yes	DIO2	VDDIO_DIO2	eMMC_CLK	SDIO1_CLK		GPIO
P1_12	Yes	DIO2	VDDIO_DIO2	eMMC_DIO0	SDIO1_D0		GPIO
P1_13	Yes	DIO2	VDDIO_DIO2	eMMC_DIO1	SDIO1_D1		GPIO
P1_14		DIO2	VDDIO_DIO2	eMMC_DIO2	SDIO1_D2		GPIO
P1_15		DIO2	VDDIO_DIO2	eMMC_DIO3	SDIO1_D3		GPIO
P1_16		DIO2		SWCLK			SWCLK
P1_17		DIO2		SWDIO			SWDIO

8.11 ADC/Analog or ADC (Aux 12-bit)

8.11.1 Introduction

The RA6W1 includes high precision, ultra-low-power, and wide dynamic range SAR ADC with a 12-bit resolution. It has a 4-channel single-end ADC. Analog input is measured by four pins from P0_04 to P0_07, and pin selection is changed through the register setting.

Figure 55 shows the control block diagram.

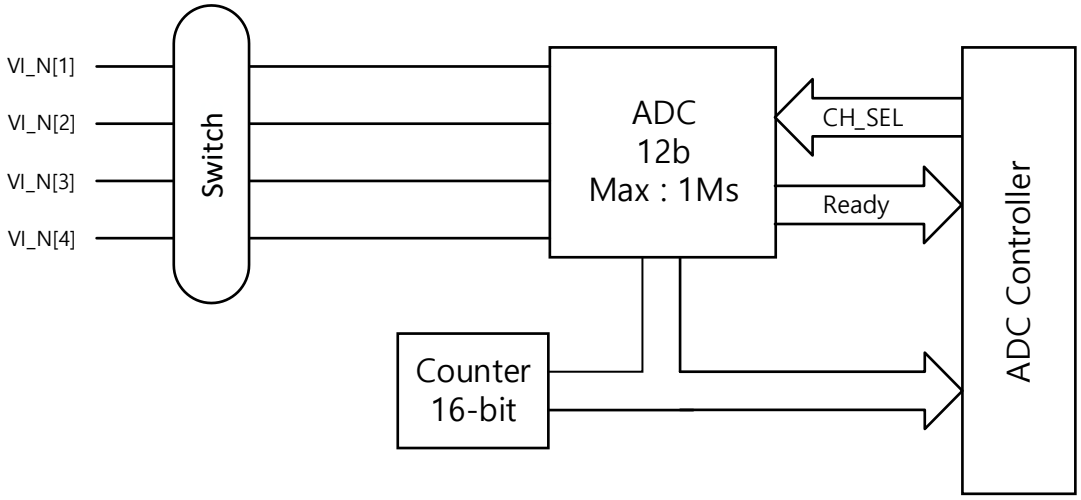


Figure 55. ADC control block diagram

8.11.2 Timing Diagram

The input is digitized at a maximum of 1.0 Msps throughput rate. And the maximum input clock rate is 15 MHz. Figure 56 shows the conversion timing, and Table 56 describes the DC specifications.

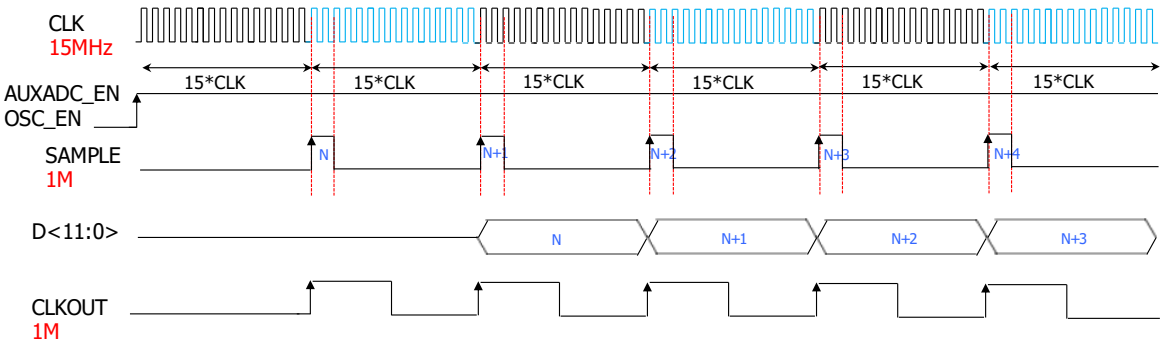


Figure 56. 12-bit ADC timing diagram

Table 56: ADC DC specification

Description	Min	Typ	Max	Unit
Resolution	4	12	12	Bits
Max clock input			15	MHz
Conversion frequency			1	MHz
Accuracy:				
SNR SNDR		67.2 61.7		dB dB
Analog input voltage	0		1.4	V
Reference voltage		0.7		V

8.11.3 DMA Transfer

There are four ADC channel settings available. When the input data of each channel reaches the FIFO level, it is possible to read the data through the DMA path.

8.11.4 Sensor Wake-up

The RA6W1 has an external sensor wake-up function that uses the analog input signal through an Aux ADC. Even in Sleep mode, when a change of an external analog signal is detected, a wake-up event occurs, and normal operation is resumed. This function can be used in up to four channels. Also, when multiple external sensors are used, analog signals are detected while the channel is automatically changed. For example, if all four channels are set as input sources which have their threshold register respectively, the channels are measured sequentially from 0 to 3.

If one of the four ADC channels exceeds the allowed range of values set by the threshold register, the RA6W1 awakes from the Sleep mode. The value setting of the input change can be either over threshold or under threshold.

8.11.5 ADC Pin Configuration

Table 57 shows the pin definition of the ADC.

Table 57: ADC pin configuration

Pin name	Default pin assignment	I/O	Description
ADC0	P0_04	ADC	ADC0 Analog input
ADC1	P0_05	ADC	ADC1 Analog input
ADC2	P0_06	ADC	ADC2 Analog input
ADC3	P0_07	ADC	ADC3 Analog input

8.12 Bluetooth LE/Zigbee Coexistence

The RA6W1 supports a three-wire coexistence interface with up to two external wireless devices, for example, Bluetooth LE and Zigbee. Built-in packet traffic arbitration is supported.

8.12.1 One External Radio Coexistence Interface

The relevant coexistence interface signals are as follows:

- BT_sig0 (oBtCoex_as)
  - Output - indicates if WLAN is currently active
  - Can be used to control an external RF switch
- BT_sig1 (iBtAct)
  - When asserted, the RA6W1 stops any RF activity
- BT_sig2 (iBTPri)
  - Optional
  - When used: if RA6W1 iBtAct = Active while iBTPri = Non-Active, the RA6W1 ignores iBtAct.

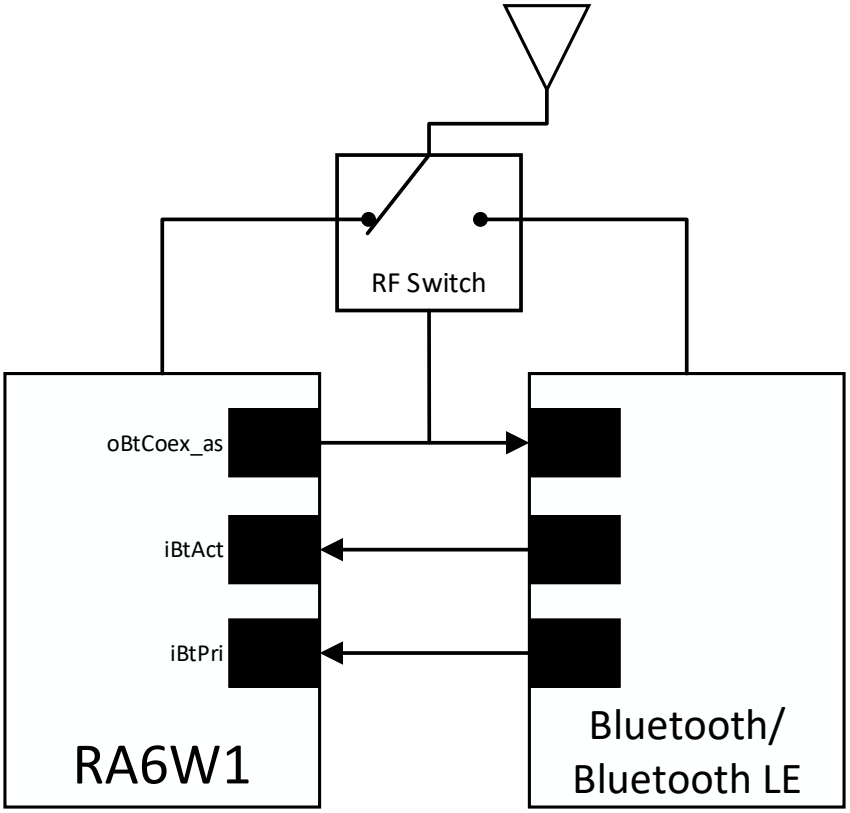


Figure 57. Bluetooth coexistence interface (antenna switch)

In this configuration, the RF antenna can be controlled as a function of different priority settings. For example, Wi-Fi TX may have higher priority than Bluetooth/Bluetooth LE, but Wi-Fi RX may have lower priority.

Table 58: Bluetooth coexistence priority example

WLAN	BT ACT (TX or RX)	Antenna
Idle	Idle	WLAN RX
Idle	ACT	BT ACT
TX	Idle	WLAN TX
TX	ACT	WLAN TX
RX	Idle	WLAN RX
RX	ACT	BT ACT

8.12.2 Two External Radios Coexistence Interface

The RA6W1 supports a wired arbitration interface with two external wireless devices, for example, Bluetooth LE and Zigbee.

In the following example, Wi-Fi 5 GHz is assumed to be fully independent from the 2.4 GHz radios RA6W1, Bluetooth LE and Zigbee. If Wi-Fi operation at 2.4 GHz is required, the RF switch needs to be 3-way to switch the single antenna to either the RA6W1, Bluetooth LE or Zigbee.

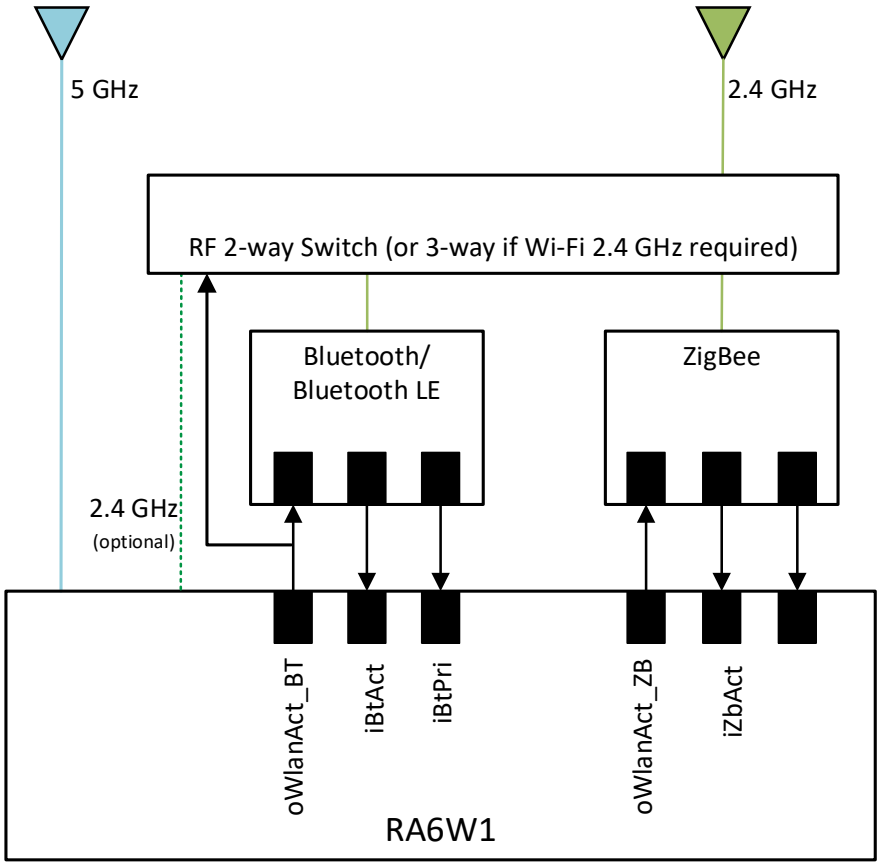


Figure 58. Typical coexistence configuration of RA6W1 with Bluetooth LE and Zigbee

8.13 Antenna Switching Diversity

8.13.1 Introduction

The antenna switching diversity function improves the performance of Wi-Fi in a multi-path environment. A PHY block measures the RSSI of each antenna and selects the antenna with the largest RSSI. The selected antenna is also used for transmission. To use this function, an external switching element is required, and switching control is done through GPIOs. Two GPIOs can be used for switching control, and for this purpose any unused pins among the GPIO pins can be selected. The control signal can be changed by register setting to suit the external switching device.

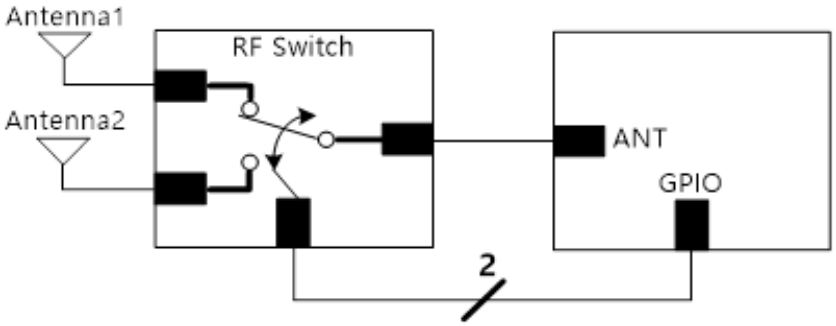


Figure 59. Antenna switching internal block diagram

If the Antenna Switching Diversity function is enabled, the function is automatically done by the PHY hardware block and is based on maximum RSSI (which is different from Maximum Ratio Combining). The basic operation scheme is as follows:

- The antenna's RSSI decision is made for 11b PPDU, except for 11g/n PPDU.
- When PHY hardware detects the existence of 11b PPDU, it stores the RSSI.

- After the switch to another antenna, the RSSI is stored, and a decision is made about which antenna has better RSSI.
- This operation is done during 11b PPDU's preamble duration to protect corruption of 11b PPDU data reception.
- The decided antenna is not changed until there is a new 11b PPDU.

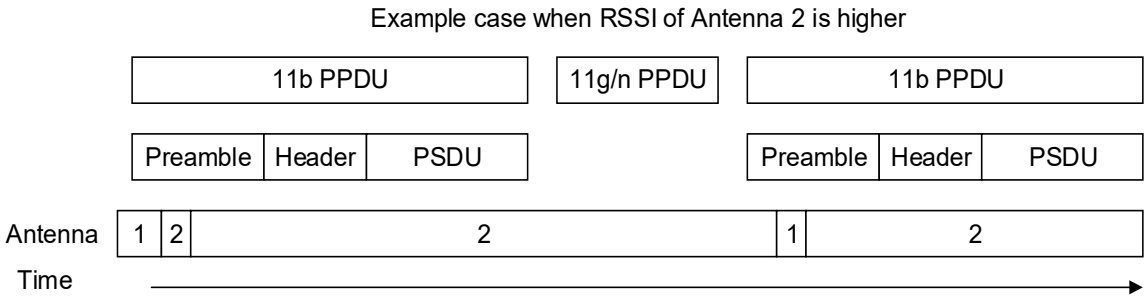


Figure 60. Antenna switching timing diagram

9. Application Information

The RA6W1 can be used in various system configurations. Typical use cases are summarized in Table 59.

Table 59: RA6W1 use cases

Use case	# of antennas	Switch type	Reference
Wi-Fi Applications	1	Diplexer	FCQFN: see Figure 61 WLCSP: see Figure 62
Wi-Fi Applications	1	FEM	FCQFN: see Figure 63 WLCSP: see Figure 64
Wi-Fi/Bluetooth Combo Applications	1	Diplexer +SP2T	FCQFN: see Figure 65 WLCSP: see Figure 66
Wi-Fi/Bluetooth Combo Applications	1	FEM (Bluetooth Support)	FCQFN: see Figure 67 WLCSP: see Figure 68

9.1 Wi-Fi Application

9.1.1 Typical Wi-Fi Application – FCQFN

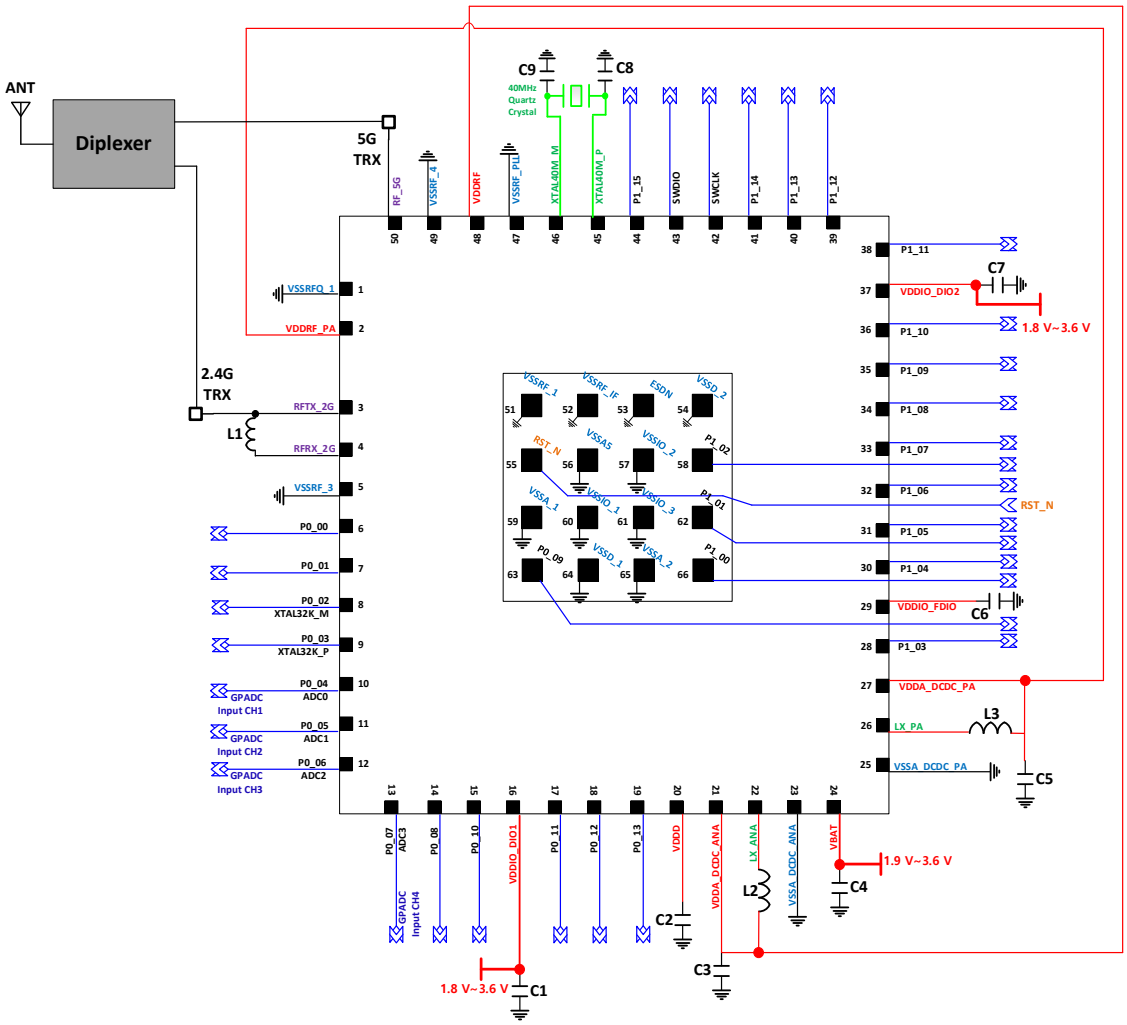


Figure 61. Typical Wi-Fi application – FCQFN

9.1.2 Typical Wi-Fi Application – WLCSP

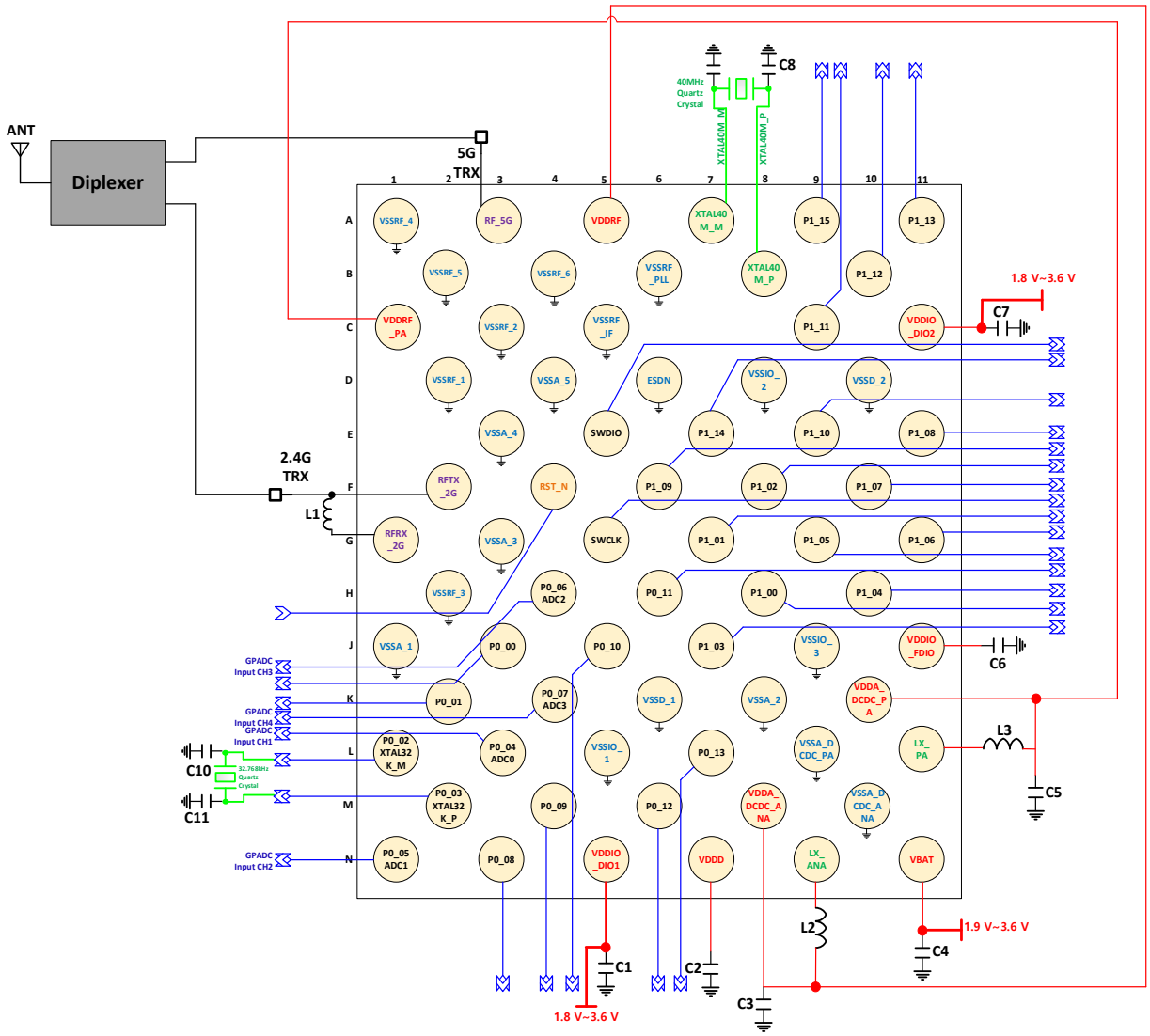


Figure 62. Typical Wi-Fi application – WLCSP

9.1.3 Wi-Fi Application with FEM – FCQFN

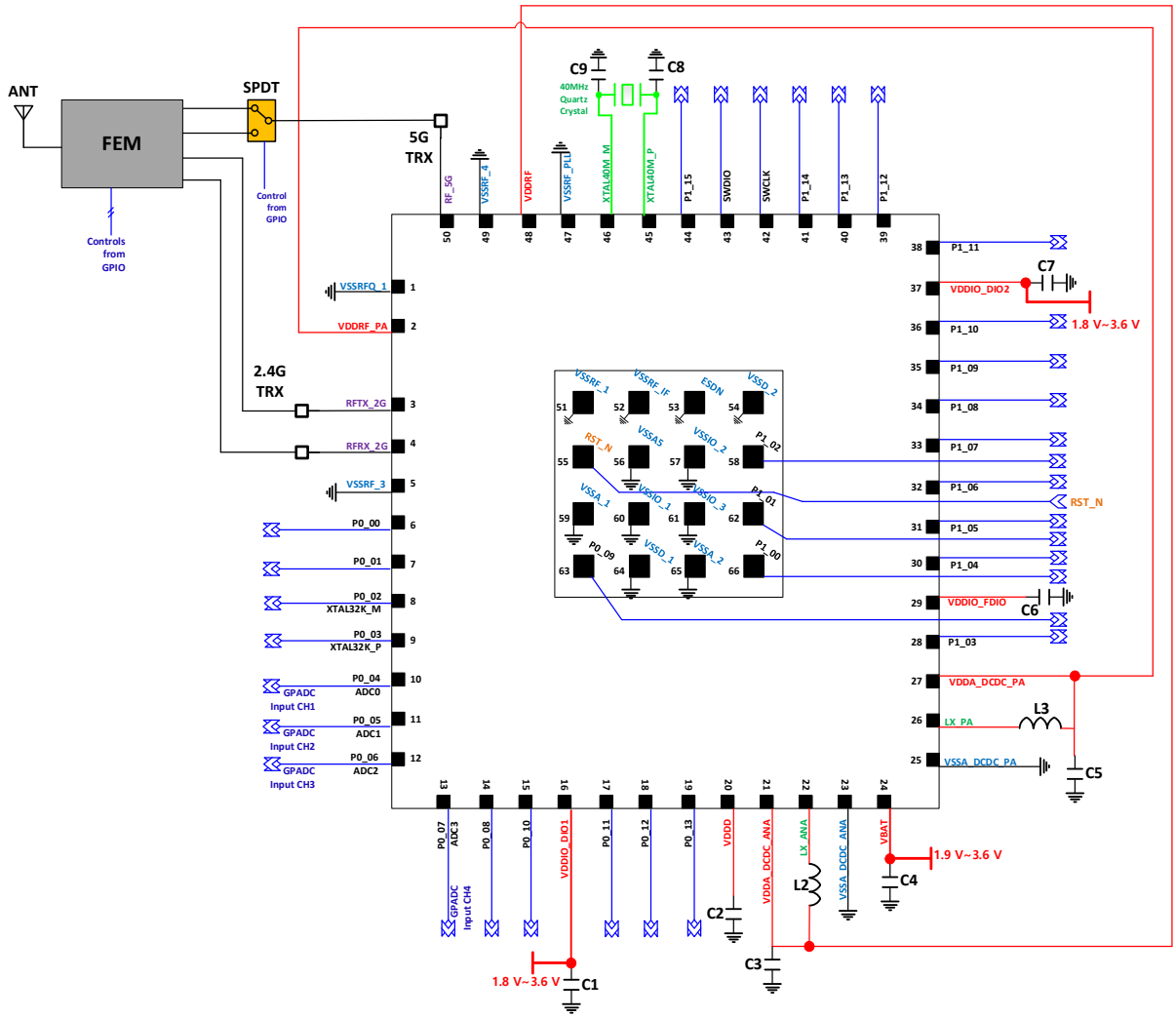


Figure 63. Wi-Fi application with FEM – FCQFN

9.1.4 Wi-Fi Application with FEM – WLCSP

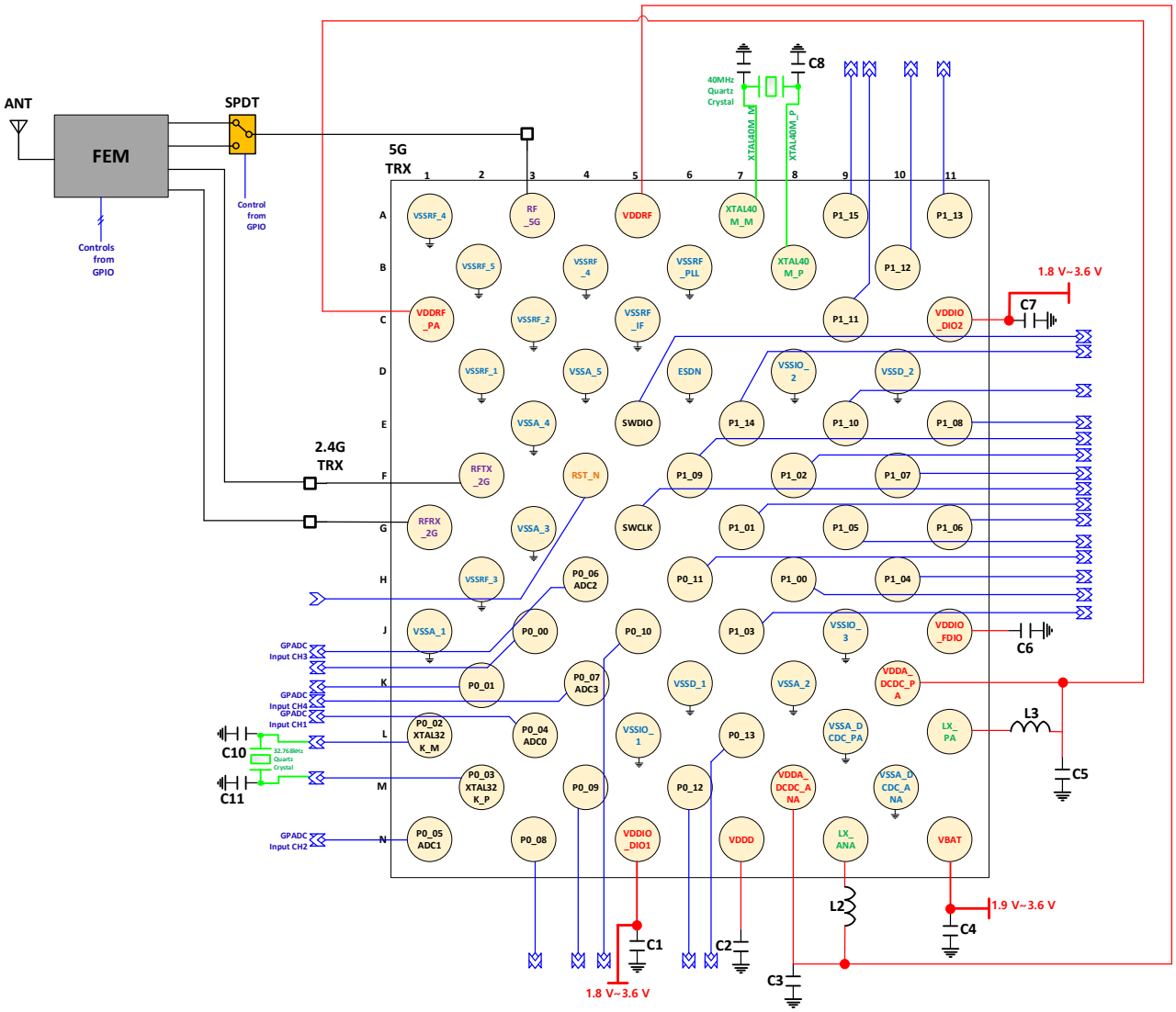


Figure 64. Wi-Fi application with FEM – WLCSP

9.2 Wi-Fi and Bluetooth Combo Application

9.2.1 Typical Wi-Fi and Bluetooth Combo Application – FCQFN

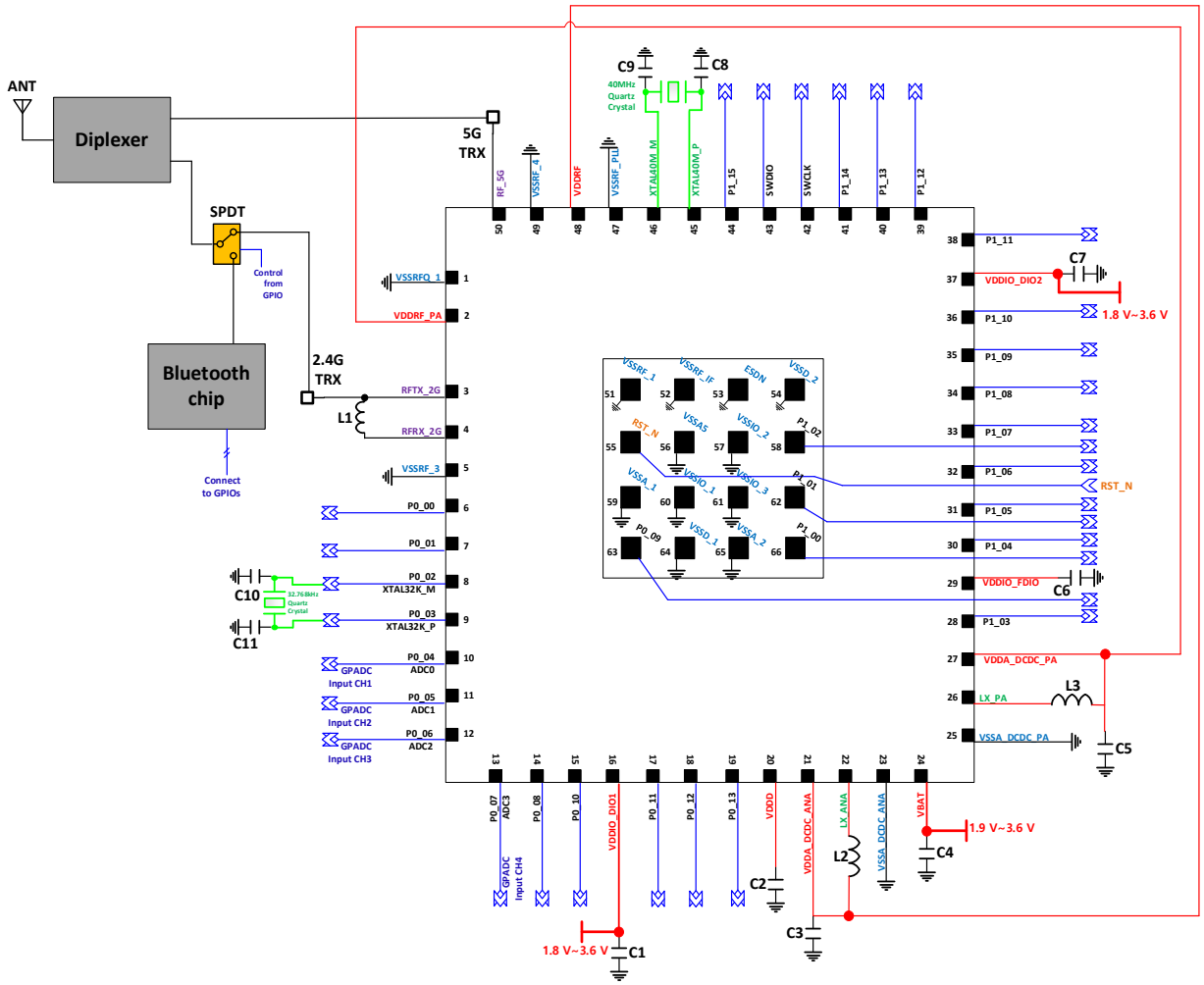


Figure 65. Typical Wi-Fi and Bluetooth combo application – FCQFN

9.2.2 Typical Wi-Fi and Bluetooth Combo Application – WLCSP

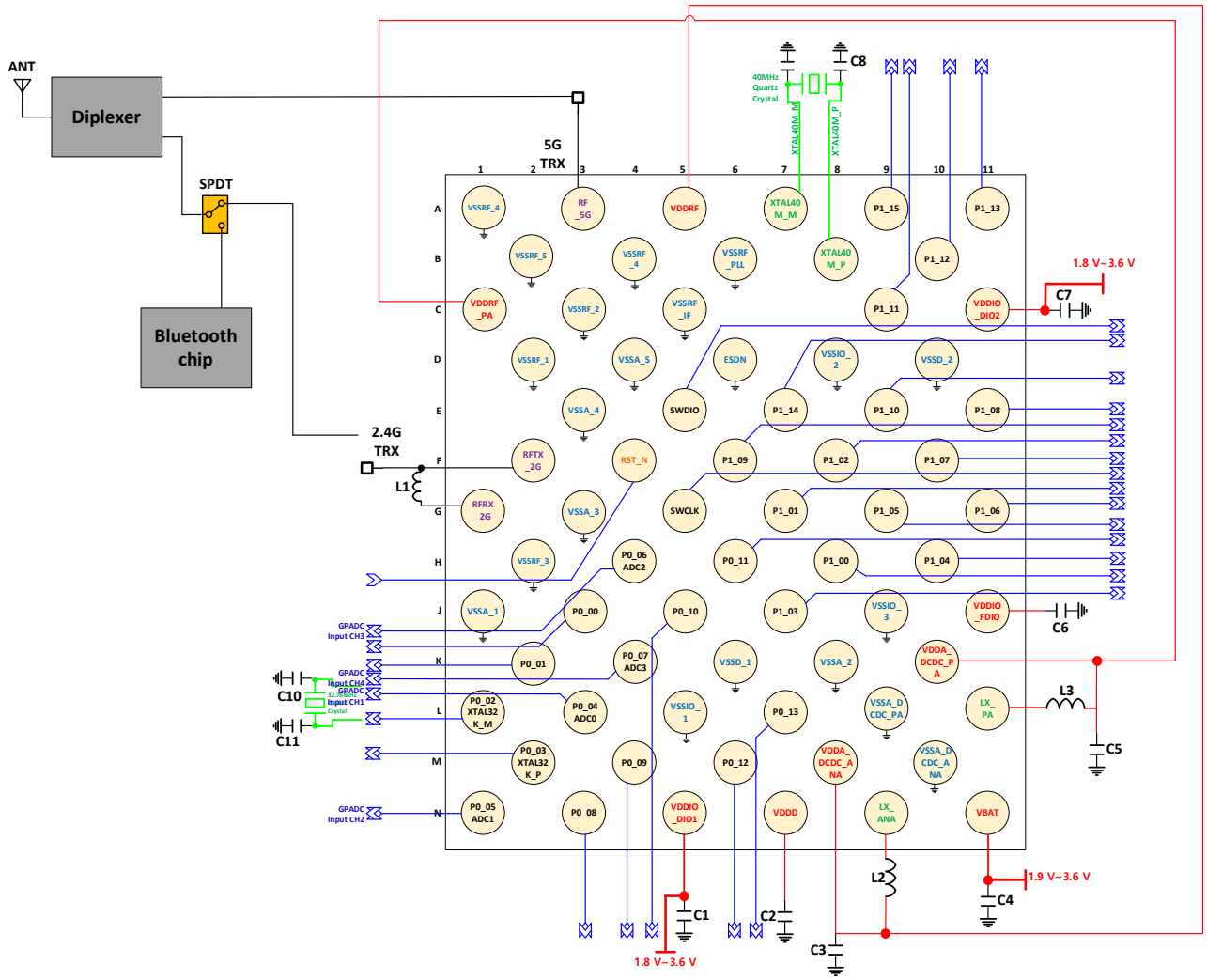


Figure 66. Typical Wi-Fi and Bluetooth combo application – WLCSP

9.2.3 Wi-Fi and Bluetooth Combo Application with FEM – FCQFN

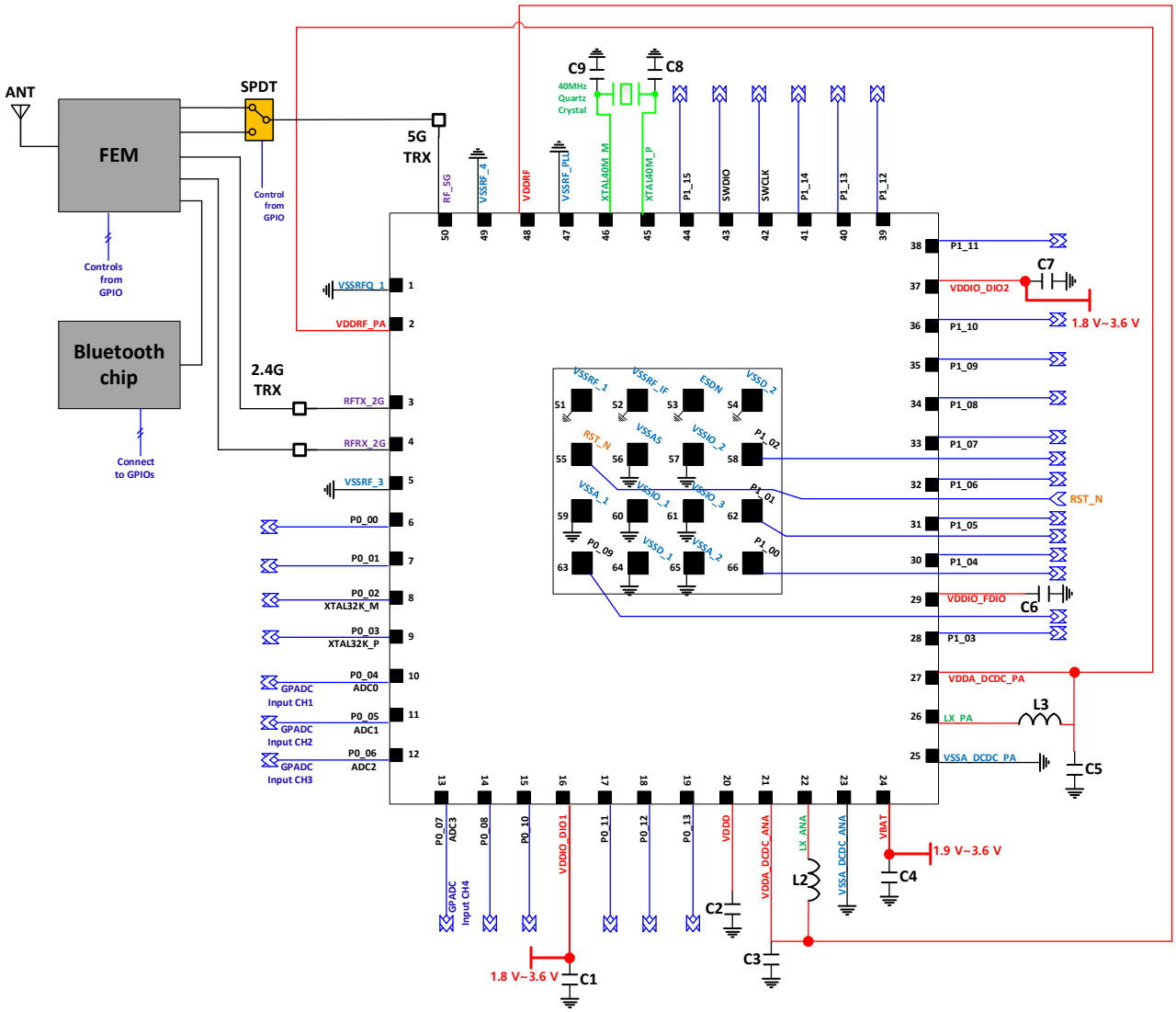


Figure 67. Wi-Fi and Bluetooth combo application with FEM – FCQFN

9.2.4 Wi-Fi and Bluetooth Combo Application with FEM – WLCSP

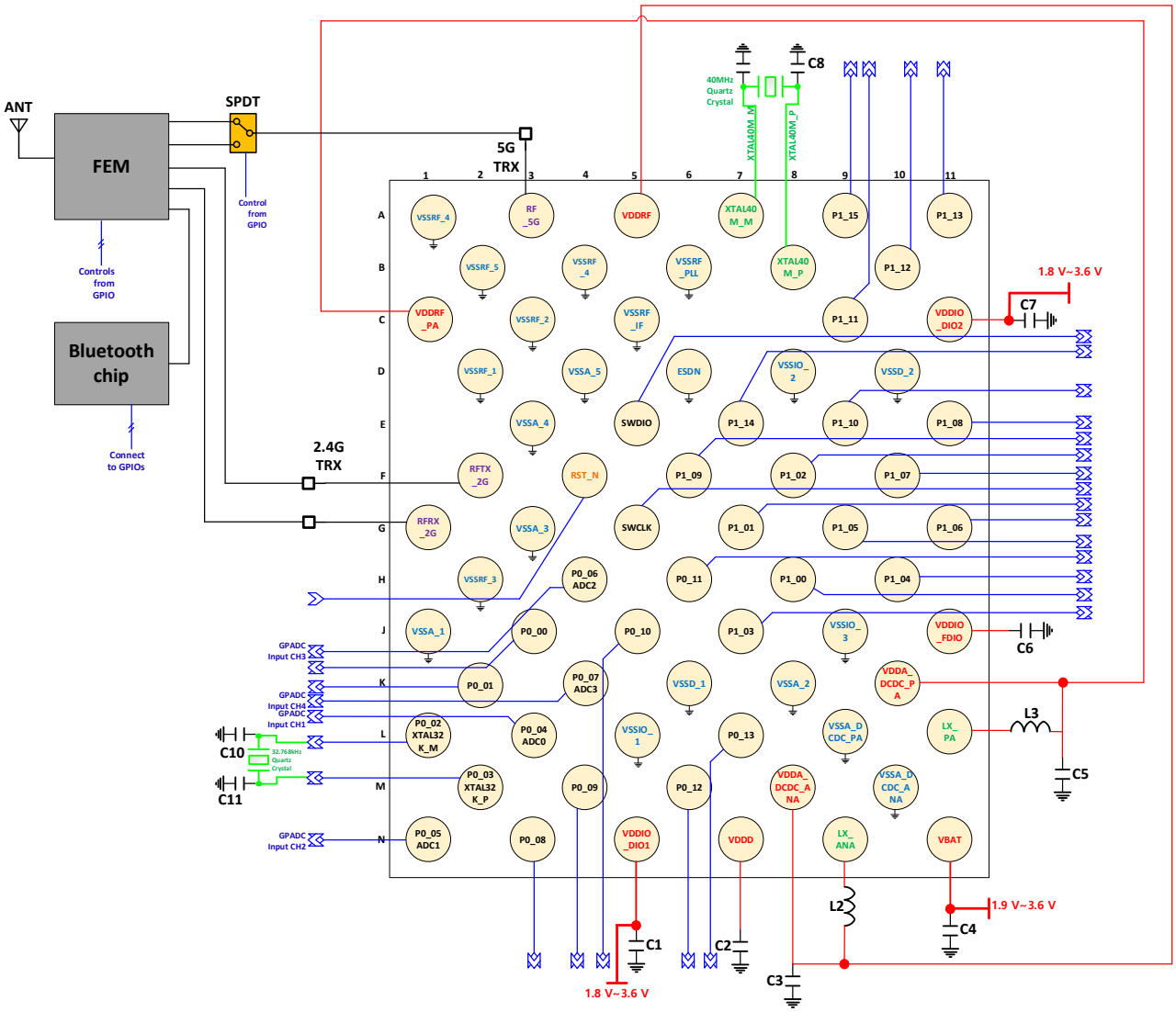


Figure 68. Wi-Fi and Bluetooth combo application with FEM – WLCSP

10. Registers

This section contains a detailed view of the registers. It is organized as follows:

- An overview table is presented initially, which depicts all register names, addresses and descriptions.
- A detailed bit level description of each register follows.

The register file of the Arm Cortex-M33 (SYSCPU) can be found on the Arm website:

- **Devices Generic User Guide:** Arm Cortex-M33 Devices Generic User Guide
- **Technical Reference Manual:** Arm Cortex-M33 Processor Technical Reference Manual

These documents contain the register descriptions for the Nested Vectored Interrupt controller (NVIC), the System control Block (SCB), and the System Timer (SysTick).

10.1 APU Registers

Table 60: Register map APU_AUD

Address	Register	Description
0x400e0400	APU_DMIC_CTRL_REG	DMIC delay and master mode control
0x400e040c	APU_DMIC_DIV_REG	DMIC clock control
0x400e0424	APU_MAIN_DIV_REG	Main divider control
0x400e0428	APU_SYNC_DIV_SEL_REG	APU SYNC control for APU SYNC DIV
0x400e042c	APU_SYNC_DIV_REG	APU SYNC divider control

Table 61: APU_DMIC_CTRL_REG (0x400E0400)

Bit	Mode	Symbol/Description	Reset
19:18	-	- Reserved	0x0
17:16	R/W	DMIC1_IN_DELAY DMIC1 input delay 0: No delay 1: 6 ns 2: 12 ns 3: 18 ns	0x0
15:1	-	- Reserved	0x0
0	R/W	DMIC_MASTER_MODE DMIC master mode. 0: Slave mode (DMICx_IN_DELAY) 1: Master mode (DMICx_PHASE)	0x0

Table 62: APU_DMIC_DIV_REG (0x400E040C)

Bit	Mode	Symbol/Description	Reset
28:24	-	- Reserved	0x0
23:21	-	- Reserved	0x0

Bit	Mode	Symbol/Description	Reset
20:16	R/W	DMIC1_PHASE Clock phase shift to sampling DMIC1 input data. 0x00: Sampled at DMIC CLK 0x01- (DMIC_DIV -2): Sampled at DMIC_CLK delayed with half of DMIC_PHASE cycles Others: Reserved	0x0
15:13	-	- Reserved	0x0
12:8	R/W	DMIC_DIV DMIC out clock divider from AUD_CLK. 0x00: Divided by 32 0x02: Divided by 2 0x04: Divided by 4 .. 0x1E: Divided by 30	0x0
7:2	-	- Reserved	0x0
1	-	- Reserved	0x0
0	R/W	DMIC1_CLK_EN DMIC1 clock divider enable for sampling input data. 0: Disable 1: Enable	0x0

Table 63: APU_MAIN_DIV_REG (0x400E0424)

Bit	Mode	Symbol/Description	Reset
12	R/W	APU_MAIN_DIV_EN APU main divider enable. 0: Disable 1: Enable	0x0
11:0	R/W	APU_MAIN_DIV APU main divider. 0x0000: Divided by 4096 0x0001: Divided by 1 0x0002: Divided by 2 0xFFFF: Divided by 4095	0x0

Table 64: APU_SYNC_DIV_SEL_REG (0x400E0428)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	APU_SYNC_DIV_SEL SYNC source selection for APU SYNC DIV. 0x00: SRC1_in_M 0x01: SRC1_out_M	0x0

Bit	Mode	Symbol/Description	Reset
		0x02: DAI1_SYNC 0x03: MAIN_DIV Note: See the datasheet PCM_SYNC Matrix diagram for invalid combinations.	

Table 65: APU_SYNC_DIV_REG (0x400E042C)

Bit	Mode	Symbol/Description	Reset
8	R/W	APU_SYNC_DIV_EN APU SYNC divider enable. 0: Disable 1: Enable	0x0
7:5	-	- Reserved	0x0
4:0	R/W	APU_SYNC_DIV APU SYNC divider. 0x00: Divided by 32 0x01: Divided by 1 0x1F: Divided by31	0x0

Table 66: Register map APU_DSP

Address	Register	Description
0x400e0800	APU_CTRL_REG	audio IP enable
0x400e0804	APU_MUX_CTRL_REG	APU mux control
0x400e0808	APU_SYNC_SRC1IN_SEL_REG	APU SYNC control for SRC1 IN
0x400e0810	APU_SYNC_SRC1OUT_SEL_REG	APU SYNC control for SRC1 OUT
0x400e0820	APU_PCM_CLK_REG	PCM clock enable
0x400e0824	APU_PCM1_DIV_REG	PCM clock divider control

Table 67: APU_CTRL_REG (0x400E0800)

Bit	Mode	Symbol/Description	Reset
1	-	- Reserved	0x0
0	R/W	DAI1_EN DAI1 enable in APU. 0: Disable 1: Enable	0x0

Table 68: APU_MUX_CTRL_REG (0x400E0804)

Bit	Mode	Symbol/Description	Reset
31:24	R/W	APU_DSP_MUX_CTL bus1_to_src = APU_DSP_MUX_CTL[0]	0x0

Bit	Mode	Symbol/Description	Reset
		bus2_to_src = APU_DSP_MUX_CTL[1] src_to_bus1 = APU_DSP_MUX_CTL[2] src_to_bus2 = APU_DSP_MUX_CTL[3] src_to_dai = APU_DSP_MUX_CTL[4] bus3_to_dai = APU_DSP_MUX_CTL[5] dai_to_src = APU_DSP_MUX_CTL[6] dai_to_bus3 = APU_DSP_MUX_CTL[7]	
23:0	-	- Reserved	0x0

Table 69: APU_SYNC_SRC1IN_SEL_REG (0x400E0808)

Bit	Mode	Symbol/Description	Reset
31:2	-	- Reserved	0x0
1:0	R/W	SRC1_IN_A_SEL SYNC source selection for SRC1_IN_A. 0x00: SRC1_in_M 0x01: SRC1_out_M 0x02: DAI1_SYNC 0x03: SYNC_DIV Note: See the datasheet APU_SYNC Matrix diagram for invalid combinations.	0x0

Table 70: APU_SYNC_SRC1OUT_SEL_REG (0x400E0810)

Bit	Mode	Symbol/Description	Reset
31:2	-	- Reserved	0x0
1:0	R/W	SRC1_OUT_A_SEL SYNC source selection for SRC1_OUT_A same as SRC1_IN_A_SEL	0x0

Table 71: APU_PCM_CLK_REG (0x400E0820)

Bit	Mode	Symbol/Description	Reset
2	R/W	- Reserved	0x0
1	-	- Reserved	0x0
0	R/W	PCM1_CLK_EN DAI clock divider enable. 0: Disable 1: Enable	0x0

Table 72: APU_PCM1_DIV_REG (0x400E0824)

Bit	Mode	Symbol/Description	Reset
6:0	R/W	PCM1_DIV PCM_DIV clock divider. 0x0: Divided by 8 0x1: Divided by 1 0x2: Divided by 2 0x3: Divided by 3 0x4: Divided by 4 0x5: Divided by 5 0x6: Divided by 6 0x7: Divided by 7 0x8-0x7F: Reserved	0x0

10.2 CACHE Registers

Table 73: Register map CACHE

Address	Register	Description
0x0e010020	CACHE_CTRL2_REG	Cache Control Register 2 (only Word (32-bits) access supported).
0x0e010028	CACHE_MRM_HITS_REG	Cache MRM (Miss Rate Monitor) HITS register (only Word (32-bit) access supported).
0x0e01002c	CACHE_MRM_MISSES_REG	Cache MRM (Miss Rate Monitor) MISSES register (only Word (32-bit) access supported).
0x0e010030	CACHE_MRM_CTRL_REG	Cache MRM (Miss Rate Monitor) CONTROL register (only Word (32-bit) access supported).
0x0e010034	CACHE_MRM_TINT_REG	Cache MRM (Miss Rate Monitor) TIME INTERVAL register (only Word (32-bit) access supported).
0x0e010038	CACHE_MRM_MISSES_THRES_REG	Cache MRM (Miss Rate Monitor) THRESHOLD register (only Word (32-bit) access supported).
0x0e01003c	CACHE_MRM_HITS_THRES_REG	Cache MRM (Miss Rate Monitor) HITS THRESHOLD register (only Word (32-bits) access supported).
0x0e010040	CACHE_FLASH_REG	Cache QSPI Flash program size and base address register (only Word (32-bit) access supported).
0x0e010048	CACHE_MRM_HITS1WS_REG	Cache MRM (Miss Rate Monitor) HITS with 1 Wait State register (only Word (32-bit) access supported).
0x0e010050	SWD_RESET_REG	SWD hardware reset control register (only Word (32-bits) access supported).

Table 74: CACHE_CTRL2_REG (0x0E010020)

Bit	Mode	Symbol/Description	Reset
31:23	-	- Reserved	0
22	R/W	CACHE_ROM_REGION_EN Cache enable for the ROM region (0x0F02_0000 ~ 0x0F05_FFFF). This option is valid only if the cache is enabled.	0x0
21	R	CACHE_READY	0

Bit	Mode	Symbol/Description	Reset
		Cache Controller RO status bit. 0: Default. 1: Set to 1 when CACHE_CTRL is enabled, initialized and immediately ready for a cacheable access to service.	
20	R	CACHE_RAM_INIT Cache Controller RO status bit. 0: Default. 1: Set to 1 when SRAM is being initialized (for example, being flushed). Note: The flushing of the cache memory takes 256 HCLK cycles.	0
19	R/W	CACHE_FLUSHED 0: Cache is not flushed yet. 1: Cache is flushed. Note 1: Setting and clearing of this (status) bit field is automatically done by the hardware. It is set on the falling edge and cleared on the rising edge of the (Cache Controller) CACHE_RAM_INIT signal. Note 2: When the Cache is flushed by disabling and enabling the Cache Controller with a SYS_CTRL_REG[CACHERAM_MUX] sequence of 1 -> 0 -> 1, the CACHE_FLUSHED bit can also be cleared first by the software (if needed) with writing a CACHE_CTRL2_REG(CACHE_FLUSHED) sequence of 1 -> 0. This is needed when the software uses the CACHE_FLUSHED bit as a status bit to wait for (for example, in a while-loop).	0
18	R/W	CACHE_FLUSH_DISABLE 0: Default. 1: Flushing of the Cache memory is disabled when SYS_CTRL_REG[CACHERAM_MUX] is switched from 1 to 0. Note: Setting this bit to 1 is only allowed for debugging purposes.	0
17:16	R/W	CACHE_USE_FULL_DB_RANGE Reserved.	0
15	R/W	CACHE_MHCLKEN_DISABLE 0: Default. 1: The m_HCLK_EN input is ignored and the controller avoids inserting m_HTRANS=BUSY because of wait states. Note: This bit is only relevant for executing from QSPI Flash (when set to 1 it improves performance). This bit should be kept on 0 for executing from eFlash.	0
14	R/W	CACHE_CWF_DISABLE 0: Default. 1: The cache line refill is performed with INCR type burst and Critical Word First is disabled. Note: This bit is only relevant for executing from QSPI Flash (when set to 1 it improves performance).This bit should be kept on 0 for executing from eFlash.	0
13	R/W	CACHE_CGEN 0: Cache controller clock gating is not enabled. 1: Cache controller clock gating is enabled (enabling power saving).	0
12	R/W	CACHE_WEN 0: Cache Data and TAG memory read-only. 1: Cache Data and TAG memory read/write. The Data and TAG memory are only updated by the cache controller. There is no hardware protection to prevent unauthorized access by the Arm.	0

Bit	Mode	Symbol/Description	Reset
		Note 1: When accessing the memory mapped Cache Data and TAG memory (which is only allowed for debugging purposes) only 32-bit access is supported. Note 2: SYS_CTRL_REG[CACHERAM_MUX] must be set to 0 before accessing the memory mapped Cache Data and TAG memory. See also the CACHE_CTRL2_REG[CACHE_USE_FULL_DB_RANGE] description.	
11:0	R/W	CACHE_LEN Length of OQSPI Flash cacheable memory. N*64 kB. N = 0 to 2048, incl. 2048 (max. of 128 MB). Setting CACHE_LEN=0 disables the caching. Note 1: The max. relevant CACHE_LEN setting depends on the chosen Flash region (program) size. Note 2: The first block (CACHE_LEN=1) includes the memory space specified by CACHE_FLASH_REG[FLASH_REGION_OFFSET].	0

Table 75: CACHE_MRM_HITS_REG (0x0E010028)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_HITS Contain the amount of cache hits.	0x0

Table 76: CACHE_MRM_MISSES_REG (0x0E01002C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_MISSES Contain the amount of cache misses.	0x0

Table 77: CACHE_MRM_CTRL_REG (0x0E010030)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0
4	R/W	MRM_IRQ_HITS_THRES_STATUS 0: No interrupt is generated. 1: Interrupt (pulse-sensitive) is generated because the number of cache hits reached the programmed threshold (threshold != 0).	0
3	R/W	MRM_IRQ_MISSES_THRES_STATUS 0: No interrupt is generated. 1: Interrupt (pulse-sensitive) is generated because the number of cache misses reached the programmed threshold (threshold != 0).	0
2	R/W	MRM_IRQ_TINT_STATUS 0: No interrupt is generated. 1: Interrupt (pulse-sensitive) is generated because the time interval counter reached the end (time interval != 0).	0
1	R/W	MRM_IRQ_MASK 0: Disables interrupt generation. 1: Enables interrupt generation.	0

Bit	Mode	Symbol/Description	Reset
		Note: The Cache MRM generates a pulse-sensitive interrupt towards the Arm processor.	
0	R/W	MRM_START 0: Freeze the misses/hits counters and reset the time interval counter to the programmed value in CACHE_MRM_TINT_REG. 1: Enables the counters. Note: In case CACHE_MRM_CTRL_REG[MRM_START] is set to 1 and CACHE_MRM_TINT_REG (!=0) is used for the MRM interrupt generation, the time interval counter counts down (on a fixed reference clock of 32 MHz) until it's 0. At that time CACHE_MRM_CTRL_REG[MRM_START] is reset automatically to 0 by the MRM hardware and the MRM interrupt is generated.	0

Table 78: CACHE_MRM_TINT_REG (0x0E010034)

Bit	Mode	Symbol/Description	Reset
31:19	-	- Reserved	0x0
18:0	R/W	MRM_TINT Define the time interval for the monitoring in (bus clock/4) clock cycles. See also the description of CACHE_MRM_CTRL_REG[MRM_IRQ_TINT_STATUS]. Note: When MRM_TINT = 0 (unrealistic value), no interrupt is generated.	0x0

Table 79: CACHE_MRM_MISSES_THRES_REG (0x0E010038)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_MISSES_THRES Define the misses threshold to trigger the interrupt generation. See also the description of CACHE_MRM_CTRL_REG[MRM_IRQ_MISSES_THRES_STATUS]. Note: When MRM_MISSES_THRES = 0 (unrealistic value), no interrupt is generated.	0x0

Table 80: CACHE_MRM_HITS_THRES_REG (0x0E01003C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_HITS_THRES Define the hits threshold to trigger the interrupt generation. See also the description of CACHE_MRM_CTRL_REG[MRM_IRQ_HITS_THRES_STATUS]. Note: When MRM_HITS_THRES = 0 (unrealistic value), no interrupt is generated.	0x0

Table 81: CACHE_FLASH_REG (0x0E010040)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	FLASH_REGION_BASE These bits correspond with the Flash region base address bits [31:16]. Default value is 0x1800. The Flash region base address bits [31:27] are fixed to 0b00011, supporting the range of 0x1800-0x1FFF. These register bits are retained.	0x1800

Bit	Mode	Symbol/Description	Reset
		Note 1: The updated value takes effect only after a software reset. Note 2: The Flash region base address setting depends on the chosen Flash region size.	
15:4	R/W	FLASH_REGION_OFFSET Flash region offset address (in words). This value is added to the Flash (CPU) address bits [13:2]. These register bits are retained. Note 1: The updated value takes effect only after a software reset.	0x0
3:0	R/W	FLASH_REGION_SIZE Flash region size. Default value is 1 (0.5 MB). 0 = 0.25 MB 1 = 0.5 MB 2 = 1 MB 3 = 2 MB 4 = 4 MB 5 = 8 MB 6 = 16 MB 7 = 32 MB 8 = 64 MB 9 = 128 MB These register bits are retained. Note 1: The updated value takes effect only after a software reset. Note 2: See for the max. region (program) size of the memory map.	0x1

Table 82: CACHE_MRM_HITS1WS_REG (0x0E010048)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_HITS1WS Contain the amount of cache hits.	0x0

Table 83: SWD_RESET_REG (0x0E010050)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0
0	R0/W	SWD_HW_RESET_REQ 0: Default. 1: Hardware reset request (from the debugger tool). The register is automatically reset with a HW_RESET. This bit can only be accessed by the debugger software and not by the application.	0

10.3 CRG Registers

Table 84: Register map CRG

Address	Register	Description
0x400c001c	RETAIN_MEM_CTRL_REG	Memory Control register
0x400c0024	RESET_STAT_REG	Reset status register
0x400c0028	PMU_CTRL_REG	Spare register

Table 85: RETAIN_MEM_CTRL_REG (0x400C001C)

Bit	Mode	Symbol/Description	Reset
7	R/W	DCACHE_RETAIN Dcache memory retain. 1: Retain 0: Off	0x0
6	R/W	CACHE_RETAIN Cache memory retain. 1: Retain 0: Off	0x0
5:4	R/W	SYS_RAM3_PWR_CTRL SYS_RAM3 Control: See SYS_RAM1_PWR_CTRL	0x0
3:2	R/W	SYS_RAM2_PWR_CTRL SYS_RAM2 Control: See SYS_RAM1_PWR_CTRL	0x0
1:0	R/W	SYS_RAM1_PWR_CTRL Power state control of the individual RAMs. May only be changed when the memory is not accessed. When SYS_IS_UP: 00: Normal operation 01: Normal operation 10: Retained (no access possible) 11: Off (memory content corrupted) When SYS_IS_DOWN: 00: Retained 01: Off (memory content corrupted) 10: Retained 11: Off (memory content corrupted)	0x0

Table 86: RESET_STAT_REG (0x400C0024)

Bit	Mode	Symbol/Description	Reset
4	R/W	M33_WDOG_STAT Indicate that M33 Watchdog has reached -16 while counting down and triggered a hardware reset. Note that it is also set when a POR occurred.	0x1
3	R/W	SWD_HWRESET_STAT Indicate that a write to SWD_RESET_REG occurred. Note that it is also set when a POR happened.	0x1

Bit	Mode	Symbol/Description	Reset
2	R/W	SWRESET_STAT Indicate that a software reset occurred.	0x1
1	R/W	HWRESET_STAT Indicate that a hardware reset occurred.	0x1
0	R/W	PORESET_STAT Indicate that a POR occurred. All bitfields of RESET_STAT_REG should be read (in order to check the source of reset) and then cleared to 0, allowing the hardware to automatically set to 1 the proper bitfields during the next reset event.	0x1

Table 87: PMU_CTRL_REG (0x400C0028)

Bit	Mode	Symbol/Description	Reset
1	R/W	MAC_SLEEP Wi-Fi: Mac block sleep control	0x1
0	R/W	PHY_SLEEP Wi-Fi: PHY block sleep control	0x1

10.4 CRG APU Registers

Table 88: Register map CRG_APU

Address	Register	Description
0x400e0000	APU_AUD_CLK_REG	AUD_CLK_DIV clock control
0x400e0004	APU_SRC_CLK_REG	SRC_DIV clock control
0x400e000c	AUD_CLK_GATE_REG	AUD_CLK_GATE control
0x400e0010	APU_MCLK_CTRL_REG	MCLK DIV control
0x400e0014	APU_START_CTRL_REG	APU_START control

Table 89: APU_AUD_CLK_REG (0x400E0000)

Bit	Mode	Symbol/Description	Reset
11:8	R/W	AUD_PCLK_DIV APB-32A clock divider. 0x0: Divided by 16 0x1: Divided by 1 0x2: Divided by 2 0x3: Divided by 3 0x4: Divided by 4 0x5: Divided by 5 0x6: Divided by 6 0x7: Divided by 7 0x8: Divided by 8 0x9: Divided by 9 0xA: Divided by 10 0xB: Divided by 11	0x0

Bit	Mode	Symbol/Description	Reset
		0xC: Divided by 12 0xD: Divided by 13 0xE: Divided by 14 0xF: Divided by 15	
7:5	-	- Reserved	0x0
4	R/W	AUD_CLK_EN AUD_CLK_DIV clock enable	0x0
3:0	R/W	AUD_CLK_DIV AUD_CLK_DIV clock divider. 0x0: Divided by 16 0x1: Divided by 1 0x2: Divided by 2 0x3: Divided by 3 0x4: Divided by 4 0x5: Divided by 5 0x6: Divided by 6 0x7: Divided by 7 0x8: Divided by 8 0x9: Divided by 9 0xA: Divided by 10 0xB: Divided by 11 0xC: Divided by 12 0xD: Divided by 13 0xE: Divided by 14 0xF: Divided by 15	0x0

Table 90: APU_SRC_CLK_REG (0x400E0004)

Bit	Mode	Symbol/Description	Reset
12	-	- Reserved	0x0
11	-	- Reserved	0x0
10	-	- Reserved	0x0
9	-	- Reserved	0x0
8	-	- Reserved	0x0
7	R/W	SRC_CLK_EN SRC clock enable. 0: Disabled 1: Enabled	0x0
6	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
5	-	-	0x0
		Reserved	
4	R/W	SRC_DMIC_CLK_EN	0x0
		SRC DMIC clock enable. 0: Disabled 1: Enabled	
3:0	R/W	SRC_CLK_DIV	0x0
		SRC_DIV clock divider. 0x0: Divided by 16 0x1: Divided by 1 0x2: Divided by 2 0x3: Divided by 3 0x4: Divided by 4 0x5: Divided by 5 0x6: Divided by 6 0x7: Divided by 7 0x8: Divided by 8 0x9: Divided by 9 0xA: Divided by 10 0xB: Divided by 11 0xC: Divided by 12 0xD: Divided by 13 0xE: Divided by 14 0xF: Divided by 15	

Table 91: AUD_CLK_GATE_REG (0x400E000C)

Bit	Mode	Symbol/Description	Reset
5	R/W	APU_RST	0x0
		Centralized APU reset. 0: Reset release 1: Reset active	
4	R/W	AUD_CLK_GATE_OVR	0x0
		AUD_CLK_GATE override. 0: Not override 1: Override to 1	
3	R	AUD_CLK_GATE_STAT	0x1
		AUD_CLK_GATE status. 0: AUD_CLK_GATE disabled 1: AUD_CLK_GATE enabled	
2	R/W	AUD_CLK_GATE_SEL	0x0
		AUD_CLK_GATE enable selection. 0: Manual mode using AUD_CLK_GATE_EN 1: APU START pulse triggered	

Bit	Mode	Symbol/Description	Reset
1	-	- Reserved	0x0
0	-	- Reserved	0x0

Table 92: APU_MCLK_CTRL_REG (0x400E0010)

Bit	Mode	Symbol/Description	Reset
9	R/W	MCLK_INV MCLK DIV output inversion. 0: Not inversion 1: Inversion	0x0
8	R/W	MCLK_SEL MCLK source selection. 0: FPLL_CLK selected 1: XTAL40M selected	0x0
7	R/W	MCLK_EN MCLK enable. 0: MCLK Disabled 1: MCLK Enabled	0x0
6:0	R/W	MCLK_DIV MCLK divider. 0: Divide by 128 1: Divide by 1 .. 127: Divide by 127	0x0

Table 93: APU_START_CTRL_REG (0x400E0014)

Bit	Mode	Symbol/Description	Reset
8	R/W	APU_START APU_START control. 0: Disable APU_START 1: Enable APU_START	0x0
7:6	-	- Reserved	0x0
5:0	-	- Reserved	0x0

10.5 CRG PREG Registers

Table 94: Register map crg_preg_common_3095_01

Address	Register	Description
0x400c0204	XTAL40M_CTRL_REG	XTAL40M Control register
0x400c0208	MEMORY_CTRL_REG	LS/RME/RM register

Address	Register	Description
0x400c020c	EXT_INTB_CTRL_REG	External interrupt control register
0x400c0210	EXT_INTB_SET_REG	External interrupt set register
0x400c0280	PSRAM_DEBUG_REG	PSRAM Debug register

Table 95: XTAL40M_CTRL_REG (0x400C0204)

Bit	Mode	Symbol/Description	Reset
23	R	XTAL40M_RDY N.A	0x0
22:16	R/W	XTAL40M_CCTRL 7 bit ppm control, default 0.	0x0
15:11	R/W	- Reserved	0x0
10:8	R/W	XTAL40_GAIN XTAL core gain control, after power on, need to set 7 to 3 0: Min 7: Max, (default code = 7)	0x7
7	R/W	- Reserved	0x0
6	R/W	XTAL40M_ADCCLK_EN It activates the clock for the IQADC. 0: Disable 1: Enable	0x0
5	R/W	XTAL40M_TX_EN 0: Disable 1: Enable	0x0
4	R/W	XTAL40M_FPLL_EN 0: Disable 1: Enable	0x0
3	R/W	XTAL40M_RFPLL_EN 0: Disable 1: Enable	0x0
2	R/W	XTAL40M_DPLL_EN 0: Disable 1: Enable	0x0
1	R/W	XTAL40M_RFEN 0: Disable 1: Enable	0x0
0	R/W	XTAL40M_EN 0: Disable 1: Enable	0x1

Table 96: MEMORY_CTRL_REG (0x400C0208)

Bit	Mode	Symbol/Description	Reset
31:28	R/W	OTHER_MEM Other memories, see bit[3:0] description.	0x7
27:25	R/W	- Reserved	0x0
24	R/W	SYS_CLK_MODE sys_clk operating mode. 0: Manual mode 1: Auto mode	0x0
23:20	R/W	MAC_MEM MAC memory, see bit[3:0] description.	0x7
19:16	R/W	PHY_MEM PHY memory, see bit[3:0] description.	0x7
11:8	R/W	ROM ROM, see bit[3:0] description.	0x7
7:4	R/W	RET_MEM Retention memory, see bit[3:0] description.	0x7
3:0	R/W	CPU_SRAM Main CPU SRAM. bit[3]: LS - light sleep bit[2]: RME - RM enable bit[1:0]: RM - default 2'b11 (fastest) RM is total 4-bit but RM[3:2] is for test so set to 0.	0x7

Table 97: EXT_INTB_CTRL_REG (0x400C020C)

Bit	Mode	Symbol/Description	Reset
7:2	R/W	PULSE_DURATION Counter unit is clock HCLK_FR. Pulse duration, increase 1 after reaching base_counter to 0xFF with clock HCLK_FR, 0 is not permitted.	0x20
1	R/W	INTR_MODE Interrupt mode. 0: Level mode 1: Edge mode	0x0
0	R/W	INTR_POL Interrupt polarity. 0: Low active 1: High active	0x0

Table 98: EXT_INTB_SET_REG (0x400C0210)

Bit	Mode	Symbol/Description	Reset
0	R/W	INTR_SET	0x0

Bit	Mode	Symbol/Description	Reset
		Interrupt set	

Table 99: PSRAM_DEBUG_REG (0x400C0280)

Bit	Mode	Symbol/Description	Reset
12:8	R	PSRAM_STS_MON PSRAM status monitoring.	0x0
7:5	R	- Reserved	0x0
4:0	R	LATENCY_CNT Latency count.	0x0

10.6 DAI Registers

Table 100: Register map DAI

Address	Register	Description
0x400e0904	DAI_MODE_REG	DAI mode
0x400e0908	DAI_SLOT_CNT_REG	DAI maximum slot number
0x400e090c	DAI_CONFIG_REG	DAI configuration
0x400e0910	DAI_W_LEN_REG	DAI word length
0x400e0914	DAI_DATA_OUT_CTRL_REG	DAI output control
0x400e091c	DAI_OFFSET_MSB_REG	DAI offset control [11:8]
0x400e0920	DAI_OFFSET_LSB_REG	DAI offset control [7:0]
0x400e0924	DAI_TX1_CH_REG	DAI slot number mapped to TX channel1
0x400e0928	DAI_TX2_CH_REG	DAI slot number mapped to TX channel2
0x400e0954	DAI_RX1_CH_REG	DAI slot number mapped to RX channel1
0x400e0958	DAI_RX2_CH_REG	DAI slot number mapped to RX channel2
0x400e0988	DAI_TX1_REG	DAI TX1 Data
0x400e098c	DAI_TX2_REG	DAI TX2 Data
0x400e09b8	DAI_RX1_REG	DAI RX1 Data
0x400e09bc	DAI_RX2_REG	DAI RX2 Data
0x400e09e8	DAI_TX_MUX_REG	DAI TX data selection
0x400e09ec	DAI_SR_CONFIG_REG	DAI Sample rate configuration

Table 101: DAI_MODE_REG (0x400E0904)

Bit	Mode	Symbol/Description	Reset
0	R/W	MODE Select between Master and Slave clock generation for the digital audio interface (DAI). 0: Slave mode (DAI receives clocks) 1: Master mode (DAI generates clocks)	0x0

Table 102: DAI_SLOT_CNT_REG (0x400E0908)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	SLOT_CNT Total number of slots. 0: No slots are enabled 1-2: Number of slots 3-31: Reserved	0x0

Table 103: DAI_CONFIG_REG (0x400E090C)

Bit	Mode	Symbol/Description	Reset
4:2	R/W	FRAME_LEN Frame length control for the internal clock generator. The DAI word clock is generated with a 50% duty cycle according to the applied frame length. This register is ignored when the DAI is in Slave mode. 0: 32 bits 1: 64 bits 2: 128 bits 3: 256 bits 4-7: Reserved	0x0
1:0	R/W	FORMAT DAI frame format. 0: I2S 1: LJF (Left justified) 2: RJF (Right justified) 3: DSP	0x3

Table 104: DAI_W_LEN_REG (0x400E0910)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	W_LEN The width of the audio data sent and received over the DAI per Channel. 0: 16 bits per slot 1: 20 bits per slot 2: 24 bits per slot 3: 32 bits per slot	0x0

Table 105: DAI_DATA_OUT_CTRL_REG (0x400E0914)

Bit	Mode	Symbol/Description	Reset
5	R/W	TDM_EARLY_RLS Configure the timing of the DAI data output in TDM mode. 0: Normal (dai_dataout_oe is driven until the end of the slot) 1: Early (dai_dataout_oe is deasserted half of BCLK earlier at the end of the slot)	0x0
4:3	R/W	DATA_OUT_EN DAI output enable.	0x3

Bit	Mode	Symbol/Description	Reset
		0: Hi-Z (Data output is tristate) 1: Hi-Z (Data output is tristate) 2: Enabled (Data driven on all slots) 3: TDM (Data driven only during enabled slots)	
1	R/W	WCLK_POL The WCLK edge defining the start of the PCM frame. 0: Rising (LJF, RJF, DSP), Falling (I2S) 1: Falling (LJF, RJF, DSP), Rising (I2S) Note: The edge is dependent on the DAI Format.	0x1
0	R/W	BCLK_POL The BCLK edge used to sample incoming data (DATA_IN). Outgoing data (DATA_OUT) is driven on the opposite edge. 0: Rising 1: Falling	0x1

Table 106: DAI_OFFSET_MSB_REG (0x400E091C)

Bit	Mode	Symbol/Description	Reset
3:0	R/W	OFFSET_MSB Bits 11-8 of the 12-bit OFFSET, which is the number of BCLK cycles offset relative to the normal data formatting. The minimum offset value is 0, the maximum is equal to the applied frame length.	0x0

Table 107: DAI_OFFSET_LSB_REG (0x400E0920)

Bit	Mode	Symbol/Description	Reset
7:0	R/W	OFFSET_LSB Bits 7-0 of the 12-bit OFFSET, which is the number of BCLK cycles offset relative to the normal data formatting. The minimum offset value is 0, the maximum is equal to the applied frame length. DAI_OFFSET_MSB_REG hold bits 11-8.	0x1

Table 108: DAI_TX1_CH_REG (0x400E0924)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	TX1_CH DAI TX Channel 1 mapping to slots 1 - 16. 0: Disable channel 1-2: Slots 1-16 3-32: Reserved	0x0

Table 109: DAI_TX2_CH_REG (0x400E0928)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	TX2_CH DAI TX Channel 2 mapping to slots 1 - 16. 0: Disable channel 1-2: Slots 1-16 3-32: Reserved	0x0

Table 110: DAI_RX1_CH_REG (0x400E0954)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	RX1_CH DAI RX Channel 1 mapping to slots 1 - 16. 0: Disable channel 1-2: Slots 1-16 3-32: Reserved	0x0

Table 111: DAI_RX2_CH_REG (0x400E0958)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	RX2_CH DAI RX Channel 2 mapping to slots 1 - 16. 0: Disable channel 1-2: Slots 1-16 3-32: Reserved	0x0

Table 112: DAI_TX1_REG (0x400E0988)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TX1_REG DAI TX1 Data register	0xFFFFFFFF

Table 113: DAI_TX2_REG (0x400E098C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TX2_REG DAI TX2 Data register	0xFFFFFFFF

Table 114: DAI_RX1_REG (0x400E09B8)

Bit	Mode	Symbol/Description	Reset
31:0	R	RX1_REG DAI RX1 Data register	0x0

Table 115: DAI_RX2_REG (0x400E09BC)

Bit	Mode	Symbol/Description	Reset
31:0	R	RX2_REG DAI RX2 Data register	0x0

Table 116: DAI_TX_MUX_REG (0x400E09E8)

Bit	Mode	Symbol/Description	Reset
1	R/W	DAI_TX2_SEL DAI TX2 data selection 0: DAI_TX2_REG	0x0

Bit	Mode	Symbol/Description	Reset
		1: DAI_TX2 (from SRC1_OUT2)	
0	R/W	DAI_TX1_SEL DAI TX1 data selection 0: DAI_TX1_REG 1: DAI_TX1 (from SRC1_OUT1)	0x0

Table 117: DAI_SR_CONFIG_REG (0x400E09EC)

Bit	Mode	Symbol/Description	Reset
4:0	R/W	DAI_SR DAI sample rate (kHz) 0x00: Reserved 0x01: 8 0x02: 11.025 0x03: 12 0x04: Reserved 0x05: 16 0x06: 22.05 0x07: 24 0x08: Reserved 0x09: 32 0x0A: 44.1 0x0B: 48 0x0C: Reserved 0x0D: Reserved 0x0E: 88.2 0x0F: 96 0x10: Reserved 0x11: Reserved 0x12: 176.4 0x13: 192 Others: Reserved	0x13

10.7 DCACHE Registers

Table 118: Register map DCACHE

Address	Register	Description
0x21000000	DCACHE_CTRL_REG	Dcache Control Register
0x21000004	DCACHE_BASE_ADDR_REG	Dcache base address for cacheable region
0x21000008	DCACHE_MRM_HITS_REG	Dcache MRM (Miss Rate Monitor) HITS Register
0x2100000c	DCACHE_MRM_MISSES_REG	Dcache MRM (Miss Rate Monitor) MISSES Register
0x21000010	DCACHE_MRM_EVICTS_REG	Dcache MRM (Miss Rate Monitor) EVICTS Register

Address	Register	Description
0x21000014	DCACHE_MRM_CTRL_REG	Dcache MRM (Miss Rate Monitor) CONTROL Register
0x21000018	DCACHE_MRM_TINT_REG	Dcache MRM (Miss Rate Monitor) TIME INTERVAL Register
0x2100001c	DCACHE_MRM_MISSES_THRES_REG	Dcache MRM (Miss Rate Monitor) THRESHOLD Register
0x21000020	DCACHE_MRM_HITS_THRES_REG	Dcache MRM (Miss Rate Monitor) HITS THRESHOLD Register
0x21000024	DCACHE_MRM_EVICTS_THRES_REG	Dcache MRM (Miss Rate Monitor) EVICTS THRESHOLD Register

Table 119: DCACHE_CTRL_REG (0x21000000)

Bit	Mode	Symbol/Description	Reset
31:26	R	-	0x0
		Reserved	
25	R/W	-	0x0
		Reserved	
24	R/W	DCACHE_WBUFFER_FLUSH	0x0
		Write buffer flush. 0: Write buffer is not flushed (default). 1: Write buffer is flushed.	
23	R	DCACHE_WBUFFER_EMPTY	0x1
		Status of the write buffer. 0: Write buffer is not empty. 1: Write buffer is empty.	
22	R/W	DCACHE_WFLUSHED	0x0
		0: DCACHE is not write flushed yet. 1: DCACHE is write flushed. Note 1: Setting and clearing of this (status) bit field is automatically done by the hardware. Note 2: The CACHE_WFLUSHED bit can also be cleared first by the software by writing 0.	
21	R	DCACHE_READY	0x0
		0: DCACHE is not initialized yet. 1: DCACHE initialization is completed.	
20	R0/W	DCACHE_WFLUSH	0x0
		Writing 1 to this field triggers a write flush of the dirty lines. All modified data in the dirty line are written back to the PSRAM. The corresponding dirty bits are cleared. Reading this bit returns 0.	
19	R0/W	DCACHE_INIT	0x0
		Writing 1 to this field triggers an initialization of the cache (0s are written in the TAG area). Reading from this field always returns 0.	
18	R/W	DCACHE_ENABLE	0x0
		Enable the dcache controller hardware block: 0: Disabled, all AHB accesses towards the QSPI are bypassing the hardware block straight into the PSRAM.	

Bit	Mode	Symbol/Description	Reset
		1: Enabled, all AHB access towards the QSPI within the cacheable region are cached.	
17:0	R/W	DCACHE_LEN Length of PSRAM cacheable memory. N*1 kB. N = 0 to 131072 (max. of 128 MB). Setting DCACHE_LEN = 0 disables the caching.	0x0

Table 120: DCACHE_BASE_ADDR_REG (0x21000004)

Bit	Mode	Symbol/Description	Reset
16:0	R/W	DCACHE_BASE_ADDR Base of PSRAM cacheable memory. N*1 kB. N = 0 to 131072 (max. of 128 MB).	0x0

Table 121: DCACHE_MRM_HITS_REG (0x21000008)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_HITS Contain the amount of cache hits.	0x0

Table 122: DCACHE_MRM_MISSES_REG (0x2100000C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_MISSES Contain the amount of cache misses.	0x0

Table 123: DCACHE_MRM_EVICTS_REG (0x21000010)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_EVICTS Contain the amount of cache evicts.	0x0

Table 124: DCACHE_MRM_CTRL_REG (0x21000014)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0
5	R/W	MRM_IRQ_EVICTS_THRES_STATUS 0: No interrupt is generated. 1: Interrupt (pulse-sensitive) is generated because the number of cache evicts reached the programmed threshold (threshold != 0).	0
4	R/W	MRM_IRQ_HITS_THRES_STATUS 0: No interrupt is generated. 1: Interrupt (pulse-sensitive) is generated because the number of cache hits reached the programmed threshold (threshold != 0).	0
3	R/W	MRM_IRQ_MISSES_THRES_STATUS 0: No interrupt is generated.	0

Bit	Mode	Symbol/Description	Reset
		1: Interrupt (pulse-sensitive) is generated because the number of cache misses reached the programmed threshold (threshold != 0).	
2	R/W	MRM_IRQ_TINT_STATUS 0: No interrupt is generated. 1: Interrupt (pulse-sensitive) is generated because the time interval counter reached the end (time interval != 0).	0
1	R/W	MRM_IRQ_MASK 0: Disables interrupt generation. 1: Enables interrupt generation. Note: The Cache MRM generates a pulse-sensitive interrupt towards the Arm processor.	0
0	R/W	MRM_START 0: Freeze the misses/hits counters and reset the time interval counter to the programmed value in CACHE_MRM_TINT_REG. 1: Enables the counters. Note: In case CACHE_MRM_CTRL_REG[MRM_START] is set to 1 and CACHE_MRM_TINT_REG (!=0) is used for the MRM interrupt generation, the time interval counter counts down (on a fixed reference clock of 16 MHz) until it is 0. At that time CACHE_MRM_CTRL_REG[MRM_START] is reset automatically to 0 by the MRM hardware and the MRM interrupt is generated.	0

Table 125: DCACHE_MRM_TINT_REG (0x21000018)

Bit	Mode	Symbol/Description	Reset
31:19	-	- Reserved	0x0
18:0	R/W	MRM_TINT Define the time interval for the monitoring in (bus clock/4) clock cycles. Also, see the description of CACHE_MRM_CTRL_REG[MRM_IRQ_TINT_STATUS]. Note: When MRM_TINT = 0 (unrealistic value), no interrupt is generated.	0x0

Table 126: DCACHE_MRM_MISSES_THRES_REG (0x2100001C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_MISSES_THRES Define the misses threshold to trigger the interrupt generation. Also, see the description of CACHE_MRM_CTRL_REG[MRM_IRQ_MISSES_THRES_STATUS]. Note: When MRM_MISSES_THRES = 0 (unrealistic value), no interrupt is generated.	0x0

Table 127: DCACHE_MRM_HITS_THRES_REG (0x21000020)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_HITS_THRES Define the hits threshold to trigger the interrupt generation. Also, see the description of CACHE_MRM_CTRL_REG[MRM_IRQ_HITS_THRES_STATUS].	0x0

Bit	Mode	Symbol/Description	Reset
		Note: When MRM_HITS_THRES = 0 (unrealistic value), no interrupt is generated.	

Table 128: DCACHE_MRM_EVICTS_THRES_REG (0x21000024)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	MRM_EVICTS_THRES Define the hits threshold to trigger the interrupt generation. Also, see the description of CACHE_MRM_CTRL_REG[MRM_IRQ_EVICTS_THRES_STATUS]. Note: When MRM_EVICTS_THRES = 0 (unrealistic value), no interrupt is generated.	0x0

10.8 DMA Registers

Table 129: Register map DMA

Address	Register	Description
0x40070700	DMA0_A_START_REG	Start address A of DMA channel 0
0x40070704	DMA0_B_START_REG	Start address B of DMA channel 0
0x40070708	DMA0_INT_REG	DMA receive interrupt register channel 0
0x4007070c	DMA0_LEN_REG	DMA receive length register channel 0
0x40070710	DMA0_CTRL_REG	Control register for the DMA channel 0
0x40070714	DMA0_IDX_REG	Index value of DMA channel 0
0x40070720	DMA1_A_START_REG	Start address A of DMA channel 1
0x40070724	DMA1_B_START_REG	Start address B of DMA channel 1
0x40070728	DMA1_INT_REG	DMA receive interrupt register channel 1
0x4007072c	DMA1_LEN_REG	DMA receive length register channel 1
0x40070730	DMA1_CTRL_REG	Control register for the DMA channel 1
0x40070734	DMA1_IDX_REG	Index value of DMA channel 1
0x40070740	DMA2_A_START_REG	Start address A of DMA channel 2
0x40070744	DMA2_B_START_REG	Start address B of DMA channel 2
0x40070748	DMA2_INT_REG	DMA receive interrupt register channel 2
0x4007074c	DMA2_LEN_REG	DMA receive length register channel 2
0x40070750	DMA2_CTRL_REG	Control register for the DMA channel 2
0x40070754	DMA2_IDX_REG	Index value of DMA channel 2
0x40070760	DMA3_A_START_REG	Start address A of DMA channel 3
0x40070764	DMA3_B_START_REG	Start address B of DMA channel 3
0x40070768	DMA3_INT_REG	DMA receive interrupt register channel 3
0x4007076c	DMA3_LEN_REG	DMA receive length register channel 3
0x40070770	DMA3_CTRL_REG	Control register for the DMA channel 3
0x40070774	DMA3_IDX_REG	Index value of DMA channel 3
0x40070780	DMA4_A_START_REG	Start address A of DMA channel 4
0x40070784	DMA4_B_START_REG	Start address B of DMA channel 4
0x40070788	DMA4_INT_REG	DMA receive interrupt register channel 4

Address	Register	Description
0x4007078c	DMA4_LEN_REG	DMA receive length register channel 4
0x40070790	DMA4_CTRL_REG	Control register for the DMA channel 4
0x40070794	DMA4_IDX_REG	Index value of DMA channel 4
0x400707a0	DMA5_A_START_REG	Start address A of DMA channel 5
0x400707a4	DMA5_B_START_REG	Start address B of DMA channel 5
0x400707a8	DMA5_INT_REG	DMA receive interrupt register channel 5
0x400707ac	DMA5_LEN_REG	DMA receive length register channel 5
0x400707b0	DMA5_CTRL_REG	Control register for the DMA channel 5
0x400707b4	DMA5_IDX_REG	Index value of DMA channel 5
0x400707c0	DMA6_A_START_REG	Start address A of DMA channel 6
0x400707c4	DMA6_B_START_REG	Start address B of DMA channel 6
0x400707c8	DMA6_INT_REG	DMA receive interrupt register channel 6
0x400707cc	DMA6_LEN_REG	DMA receive length register channel 6
0x400707d0	DMA6_CTRL_REG	Control register for the DMA channel 6
0x400707d4	DMA6_IDX_REG	Index value of DMA channel 6
0x400707e0	DMA7_A_START_REG	Start address A of DMA channel 7
0x400707e4	DMA7_B_START_REG	Start address B of DMA channel 7
0x400707e8	DMA7_INT_REG	DMA receive interrupt register channel 7
0x400707ec	DMA7_LEN_REG	DMA receive length register channel 7
0x400707f0	DMA7_CTRL_REG	Control register for the DMA channel 7
0x400707f4	DMA7_IDX_REG	Index value of DMA channel 7
0x40070800	DMA8_A_START_REG	Start address A of DMA channel 8
0x40070804	DMA8_B_START_REG	Start address B of DMA channel 8
0x40070808	DMA8_INT_REG	DMA receive interrupt register channel 8
0x4007080c	DMA8_LEN_REG	DMA receive length register channel 8
0x40070810	DMA8_CTRL_REG	Control register for the DMA channel 8
0x40070814	DMA8_IDX_REG	Index value of DMA channel 8
0x40070820	DMA9_A_START_REG	Start address A of DMA channel 9
0x40070824	DMA9_B_START_REG	Start address B of DMA channel 9
0x40070828	DMA9_INT_REG	DMA receive interrupt register channel 9
0x4007082c	DMA9_LEN_REG	DMA receive length register channel 9
0x40070830	DMA9_CTRL_REG	Control register for the DMA channel 9
0x40070834	DMA9_IDX_REG	Index value of DMA channel 9
0x40070840	DMA10_A_START_REG	Start address A of DMA channel 10
0x40070844	DMA10_B_START_REG	Start address B of DMA channel 10
0x40070848	DMA10_INT_REG	DMA receive interrupt register channel 10
0x4007084c	DMA10_LEN_REG	DMA receive length register channel 10
0x40070850	DMA10_CTRL_REG	Control register for the DMA channel 10
0x40070854	DMA10_IDX_REG	Index value of DMA channel 10
0x40070860	DMA11_A_START_REG	Start address A of DMA channel 11
0x40070864	DMA11_B_START_REG	Start address B of DMA channel 11

Address	Register	Description
0x40070868	DMA11_INT_REG	DMA receive interrupt register channel 11
0x4007086c	DMA11_LEN_REG	DMA receive length register channel 11
0x40070870	DMA11_CTRL_REG	Control register for the DMA channel 11
0x40070874	DMA11_IDX_REG	Index value of DMA channel 11
0x40070880	DMA12_A_START_REG	Start address A of DMA channel 12
0x40070884	DMA12_B_START_REG	Start address B of DMA channel 12
0x40070888	DMA12_INT_REG	DMA receive interrupt register channel 12
0x4007088c	DMA12_LEN_REG	DMA receive length register channel 12
0x40070890	DMA12_CTRL_REG	Control register for the DMA channel 12
0x40070894	DMA12_IDX_REG	Index value of DMA channel 12
0x400708a0	DMA13_A_START_REG	Start address A of DMA channel 13
0x400708a4	DMA13_B_START_REG	Start address B of DMA channel 13
0x400708a8	DMA13_INT_REG	DMA receive interrupt register channel 13
0x400708ac	DMA13_LEN_REG	DMA receive length register channel 13
0x400708b0	DMA13_CTRL_REG	Control register for the DMA channel 13
0x400708b4	DMA13_IDX_REG	Index value of DMA channel 13
0x400708c0	DMA14_A_START_REG	Start address A of DMA channel 14
0x400708c4	DMA14_B_START_REG	Start address B of DMA channel 14
0x400708c8	DMA14_INT_REG	DMA receive interrupt register channel 14
0x400708cc	DMA14_LEN_REG	DMA receive length register channel 14
0x400708d0	DMA14_CTRL_REG	Control register for the DMA channel 14
0x400708d4	DMA14_IDX_REG	Index value of DMA channel 14
0x400708e0	DMA15_A_START_REG	Start address A of DMA channel 15
0x400708e4	DMA15_B_START_REG	Start address B of DMA channel 15
0x400708e8	DMA15_INT_REG	DMA receive interrupt register channel 15
0x400708ec	DMA15_LEN_REG	DMA receive length register channel 15
0x400708f0	DMA15_CTRL_REG	Control register for the DMA channel 15
0x400708f4	DMA15_IDX_REG	Index value of DMA channel 15
0x40070900	DMA_REQ_MUX_REG	DMA channel assignments (Channel 0~3)
0x40070904	DMA_REQ_MUX2_REG	DMA channel assignments (Channel 4~7)
0x40070908	DMA_REQ_MUX3_REG	DMA channel assignments (Channel 8~11)
0x4007090c	DMA_REQ_MUX4_REG	DMA channel assignments (Channel 12~15)
0x40070910	DMA_INT_STATUS_REG	DMA interrupt status register
0x40070914	DMA_CLEAR_INT_REG	DMA clear interrupt register
0x40070918	DMA_INT_MASK_REG	DMA Interrupt mask register

Table 130: DMA0_A_START_REG (0x40070700)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA0_A_START	0x0
		Source start address	

Table 131: DMA0_B_START_REG (0x40070704)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA0_B_START Destination start address	0x0

Table 132: DMA0_INT_REG (0x40070708)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA0_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 133: DMA0_LEN_REG (0x4007070C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA0_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 134: DMA0_CTRL_REG (0x40070710)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts, according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0

Bit	Mode	Symbol/Description	Reset
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Therefore, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy, disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel is granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0~7 can be set from 7 to 0, and channel 8~15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0

Bit	Mode	Symbol/Description	Reset
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers and 24/32-bit audio samples) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit is automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel completed the last on-going read-write cycle and stopped. Therefore, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0, before setting again the specific bit-field.	0x0

Table 135: DMA0_IDX_REG (0x40070714)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA0_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON has been cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 136: DMA1_A_START_REG (0x40070720)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA1_A_START Source start address	0x0

Table 137: DMA1_B_START_REG (0x40070724)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA1_B_START Destination start address	0x0

Table 138: DMA1_INT_REG (0x40070728)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA1_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 139: DMA1_LEN_REG (0x4007072C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA1_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 140: DMA1_CTRL_REG (0x40070730)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Therefore, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority.	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	
9:7	R/W	DMA_PRIO The priority level determines which DMA channel is granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0~7 can be set from 7 to 0 and channel 8~15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if Circular mode is not enabled. In Circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel completed the last on-going read-write	0x0

Bit	Mode	Symbol/Description	Reset
		cycle and stopped. Therefore, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	

Table 141: DMA1_IDX_REG (0x40070734)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA1_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON has been cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 142: DMA2_A_START_REG (0x40070740)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA2_A_START Source start address	0x0

Table 143: DMA2_B_START_REG (0x40070744)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA2_B_START Destination start address	0x0

Table 144: DMA2_INT_REG (0x40070748)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA2_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 145: DMA2_LEN_REG (0x4007074C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA2_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 146: DMA2_CTRL_REG (0x40070750)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only).	0x0

Bit	Mode	Symbol/Description	Reset
		000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts, according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Therefore, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy, disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus.	0x0

Bit	Mode	Symbol/Description	Reset
		DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment 1 = Increment according value of BW	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Therefore, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 147: DMA2_IDX_REG (0x40070754)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA2_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON has been cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to	0x0

Bit	Mode	Symbol/Description	Reset
		0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	

Table 148: DMA3_A_START_REG (0x40070760)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA3_A_START Source start address	0x0

Table 149: DMA3_B_START_REG (0x40070764)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA3_B_START Destination start address	0x0

Table 150: DMA3_INT_REG (0x40070768)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA3_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 151: DMA3_LEN_REG (0x4007076C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA3_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 152: DMA3_CTRL_REG (0x40070770)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write	0x1

Bit	Mode	Symbol/Description	Reset
		DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer)	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 153: DMA3_IDX_REG (0x40070774)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA3_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON has been cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 154: DMA4_A_START_REG (0x40070780)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA4_A_START Source start address	0x0

Table 155: DMA4_B_START_REG (0x40070784)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA4_B_START Destination start address	0x0

Table 156: DMA4_INT_REG (0x40070788)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA4_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 157: DMA4_LEN_REG (0x4007078C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA4_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 158: DMA4_CTRL_REG (0x40070790)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0

Bit	Mode	Symbol/Description	Reset
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy, disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0

Bit	Mode	Symbol/Description	Reset
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 159: DMA4_IDX_REG (0x40070794)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA4_IDX This (read-only) register determines the data items already transferred by the DMA channel. Hence, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON has been cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 160: DMA5_A_START_REG (0x400707A0)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA5_A_START Source start address	0x0

Table 161: DMA5_B_START_REG (0x400707A4)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA5_B_START Destination start address	0x0

Table 162: DMA5_INT_REG (0x400707A8)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA5_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 163: DMA5_LEN_REG (0x400707AC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA5_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 164: DMA5_CTRL_REG (0x400707B0)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enables the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy, disabling bus access for any bus master with lower priority.	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-	0x0

Bit	Mode	Symbol/Description	Reset
		write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	

Table 165: DMA5_IDX_REG (0x400707B4)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA5_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 166: DMA6_A_START_REG (0x400707C0)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA6_A_START Source start address	0x0

Table 167: DMA6_B_START_REG (0x400707C4)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA6_B_START Destination start address	0x0

Table 168: DMA6_INT_REG (0x400707C8)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA6_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 169: DMA6_LEN_REG (0x400707CC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA6_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 170: DMA6_CTRL_REG (0x400707D0)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only).	0x0

Bit	Mode	Symbol/Description	Reset
		000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy, disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus.	0x0

Bit	Mode	Symbol/Description	Reset
		DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 171: DMA6_IDX_REG (0x400707D4)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA6_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to	0x0

Bit	Mode	Symbol/Description	Reset
		0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	

Table 172: DMA7_A_START_REG (0x400707E0)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA7_A_START Source start address	0x0

Table 173: DMA7_B_START_REG (0x400707E4)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA7_B_START Destination start address	0x0

Table 174: DMA7_INT_REG (0x400707E8)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA7_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 175: DMA7_LEN_REG (0x400707EC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA7_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 176: DMA7_CTRL_REG (0x400707F0)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write	0x1

Bit	Mode	Symbol/Description	Reset
		DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, etc ... 1 = DMA performs copy of A1 to B1, B2, etc ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy, disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer)	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 177: DMA7_IDX_REG (0x400707F4)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA7_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 178: DMA8_A_START_REG (0x40070800)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA8_A_START Source start address	0x0

Table 179: DMA8_B_START_REG (0x40070804)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA8_B_START Destination start address	0x0

Table 180: DMA8_INT_REG (0x40070808)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA8_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 181: DMA8_LEN_REG (0x4007080C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA8_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 182: DMA8_CTRL_REG (0x40070810)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only) 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0

Bit	Mode	Symbol/Description	Reset
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0

Bit	Mode	Symbol/Description	Reset
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 183: DMA8_IDX_REG (0x40070814)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA8_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 184: DMA9_A_START_REG (0x40070820)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA9_A_START Source start address	0x0

Table 185: DMA9_B_START_REG (0x40070824)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA9_B_START Destination start address	0x0

Table 186: DMA9_INT_REG (0x40070828)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA9_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 187: DMA9_LEN_REG (0x4007082C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA9_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 188: DMA9_CTRL_REG (0x40070830)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only) 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority.	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-	0x0

Bit	Mode	Symbol/Description	Reset
		write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	

Table 189: DMA9_IDX_REG (0x40070834)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA9_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 190: DMA10_A_START_REG (0x40070840)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA10_A_START Source start address	0x0

Table 191: DMA10_B_START_REG (0x40070844)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA10_B_START Destination start address	0x0

Table 192: DMA10_INT_REG (0x40070848)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA10_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 193: DMA10_LEN_REG (0x4007084C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA10_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 194: DMA10_CTRL_REG (0x40070850)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only)	0x0

Bit	Mode	Symbol/Description	Reset
		000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus.	0x0

Bit	Mode	Symbol/Description	Reset
		DMA channel 0 – 7 can be set from 7 to 0 and channel 8 – 15 can be set from 15 to 8.	
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I^2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 195: DMA10_IDX_REG (0x40070854)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA10_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to	0x0

Bit	Mode	Symbol/Description	Reset
		0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	

Table 196: DMA11_A_START_REG (0x40070860)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA11_A_START Source start address	0x0

Table 197: DMA11_B_START_REG (0x40070864)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA11_B_START Destination start address	0x0

Table 198: DMA11_INT_REG (0x40070868)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA11_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 199: DMA11_LEN_REG (0x4007086C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA11_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 200: DMA11_CTRL_REG (0x40070870)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only) 000: 4 B 001: 8 B 010: 12 B 011: 16 B 100: 20 B 101: 24 B 110: 28 B 111: 32 B	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write	0x1

Bit	Mode	Symbol/Description	Reset
		DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, and so forth 1 = DMA performs copy of A1 to B1, B2, and so forth This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0~7 can be set from 7 to 0 and channel 8~15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer)	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = 00; by 2, when BW = 01; by 4, when BW = 10)	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 B (suggested for peripherals like UART and 8-bit SPI) 01 = 2 B (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 B (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if Circular mode is not enabled. In Circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 201: DMA11_IDX_REG (0x40070874)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA11_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 202: DMA12_A_START_REG (0x40070880)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA12_A_START Source start address	0x0

Table 203: DMA12_B_START_REG (0x40070884)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA12_B_START Destination start address	0x0

Table 204: DMA12_INT_REG (0x40070888)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA12_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 205: DMA12_LEN_REG (0x4007088C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA12_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 206: DMA12_CTRL_REG (0x40070890)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only) 000: 4 B 001: 8 B 010: 12 B 011: 16 B 100: 20 B 101: 24 B 110: 28 B 111: 32 B	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0

Bit	Mode	Symbol/Description	Reset
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to '1', while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE='1'.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE=1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0~7 can be set from 7 to 0 and channel 8~15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0

Bit	Mode	Symbol/Description	Reset
2:1	R/W	BW Bus transfer width: 00 = 1 B (suggested for peripherals like UART and 8-bit SPI) 01 = 2 B (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 B (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if Circular mode is not enabled. In Circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel is completed the last on-going read-write cycle and stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 207: DMA12_IDX_REG (0x40070894)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA12_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 208: DMA13_A_START_REG (0x400708A0)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA13_A_START Source start address	0x0

Table 209: DMA13_B_START_REG (0x400708A4)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA13_B_START Destination start address	0x0

Table 210: DMA13_INT_REG (0x400708A8)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA13_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 211: DMA13_LEN_REG (0x400708AC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA13_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 212: DMA13_CTRL_REG (0x400708B0)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enables the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to '1', while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE='1'.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority.	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE='1', DMA_IDLE is don't care.	
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus. DMA channel 0~7 can be set from 7 to 0 and channel 8~15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = '1'). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit will be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by SW while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle	0x0

Bit	Mode	Symbol/Description	Reset
		and has stopped. Thus, the SW has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	

Table 213: DMA13_IDX_REG (0x400708B4)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA13_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 214: DMA14_A_START_REG (0x400708C0)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA14_A_START Source start address	0x0

Table 215: DMA14_B_START_REG (0x400708C4)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA14_B_START Destination start address	0x0

Table 216: DMA14_INT_REG (0x400708C8)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA14_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 217: DMA14_LEN_REG (0x400708CC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA14_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 218: DMA14_CTRL_REG (0x400708D0)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only).	0x0

Bit	Mode	Symbol/Description	Reset
		000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again, in order to perform a new transfer.	0x1
14:13	R/W	BURST_MODE Enables the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 should first be granted access to the bus.	0x0

Bit	Mode	Symbol/Description	Reset
		DMA channel 0~7 can be set from 7 to 0 and channel 8~15 can be set from 15 to 8.	
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 219: DMA14_IDX_REG (0x400708D4)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA14_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to	0x0

Bit	Mode	Symbol/Description	Reset
		0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	

Table 220: DMA15_A_START_REG (0x400708E0)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA15_A_START Source start address	0x0

Table 221: DMA15_B_START_REG (0x400708E4)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	DMA15_B_START Destination start address	0x0

Table 222: DMA15_INT_REG (0x400708E8)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA15_INT Number of transfers until an interrupt is generated. The interrupt is generated after a transfer, if DMAx_INT_REG is equal to DMAx_IDX_REG and before DMAx_IDX_REG is incremented. The bit-field DMA_IRQ_ENABLEx of DMA_INT_MASK_REG must be set to 1 to let the controller generate the interrupt.	0x0

Table 223: DMA15_LEN_REG (0x400708EC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	DMA15_LEN DMA channel's transfer length. DMAx_LEN of value 0, 1, 2, ... results into an actual transfer length of 1, 2, 3, ...	0x0

Table 224: DMA15_CTRL_REG (0x400708F0)

Bit	Mode	Symbol/Description	Reset
18:16	R/W	ADDR_INC Address increment definition for the peripherals (BW 32-bit only). 000: 4 bytes 001: 8 bytes 010: 12 bytes 011: 16 bytes 100: 20 bytes 101: 24 bytes 110: 28 bytes 111: 32 bytes	0x0
15	R/W	BUS_ERROR_DETECT 0 = Ignores bus error response from the AHB bus, so DMA continues normally. 1 = Detects the bus response and tracks any bus error may occur during the transfer. If a bus error is detected, the channel completes the current read-write	0x1

Bit	Mode	Symbol/Description	Reset
		DMA cycle (either in burst or single transfers mode) and then closes the transfer, de-asserting DMA_ON bit automatically. It is noted that the respective bus error detection status bit of DMA_INT_STATUS_REG is automatically cleared as soon as the channel is switched-on again to perform a new transfer.	
14:13	R/W	BURST_MODE Enable the DMA read/write bursts according to the following configuration: 00 = Bursts are disabled 01 = Bursts of 4 are enabled 10 = Bursts of 8 are enabled 11 = Reserved	0x0
12	R/W	REQ_SENSE 0 = DMA operates with level-sensitive peripheral requests (default) 1 = DMA operates with (positive) edge-sensitive peripheral requests	0x0
11	R/W	DMA_INIT 0 = DMA performs copy A1 to B1, A2 to B2, ... 1 = DMA performs copy of A1 to B1, B2, ... This feature is useful for memory initialization to any value. Thus, BINC must be set to 1, while AINC is don't care, as only one fetch from A is done. This process cannot be interrupted by other DMA channels. It is also noted that DMA_INIT should not be used when DREQ_MODE = 1.	0x0
10	R/W	DMA_IDLE 0 = Blocking mode, the DMA performs a fast back-to-back copy disabling bus access for any bus master with lower priority. 1 = Interrupting mode, the DMA inserts a wait cycle after each store allowing the CPU to steal cycles or cache to perform a burst read. If DREQ_MODE = 1, DMA_IDLE is don't care.	0x0
9:7	R/W	DMA_PRIO The priority level determines which DMA channel should be granted access for transferring data, in case more than one channels are active and request the bus at the same time. The greater the value, the higher the priority. In specific: 000 = Lowest priority 111 = Highest priority If different channels with equal priority level values request the bus at the same time, an inherent priority mechanism is applied. According to this mechanism, for example, if both the DMA0 and DMA1 channels have the same priority level, then DMA0 shouldl first be granted access to the bus. DMA channel 0~7 can be set from 7 to 0 and channel 8~15 can be set from 15 to 8.	0x0
6	R/W	CIRCULAR 0 = Normal mode. The DMA channel stops after having completed the transfer of length determined by DMAx_LEN_REG. DMA_ON automatically deasserts when the transfer is completed. 1 = Circular mode (applicable only if DREQ_MODE = 1). In this mode, DMA_ON never deasserts, as the DMA channel automatically resets DMAx_IDX_REG and starts a new transfer.	0x0
5	R/W	AINC Enable increment of source address. 0 = Do not increment (source address stays the same during the transfer)	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	
4	R/W	BINC Enable increment of destination address. 0 = Do not increment (destination address stays the same during the transfer) 1 = Increment according to the value of BW bit-field (by 1, when BW = "00"; by 2, when BW = "01"; by 4, when BW = "10")	0x0
3	R/W	DREQ_MODE 0 = DMA channel starts immediately 1 = DMA channel must be triggered by peripheral DMA request (see also the description of DMA_REQ_MUX_REG)	0x0
2:1	R/W	BW Bus transfer width: 00 = 1 byte (suggested for peripherals like UART and 8-bit SPI) 01 = 2 bytes (suggested for peripherals like I2C and 16-bit SPI) 10 = 4 bytes (suggested for Memory-to-Memory transfers) 11 = Reserved	0x0
0	R/W	DMA_ON 0 = DMA channel is off, clocks are disabled 1 = DMA channel is enabled. This bit should be automatically cleared after the completion of a transfer, if circular mode is not enabled. In circular mode, this bit stays set. Note: If DMA_ON is disabled by software while the DMA channel is active, it cannot be enabled again until the channel has completed the last on-going read-write cycle and has stopped. Thus, the software has to check that the reading of DMAx_CTRL_REG. DMA_ON returns 0 before setting again the specific bit-field.	0x0

Table 225: DMA15_IDX_REG (0x400708F4)

Bit	Mode	Symbol/Description	Reset
15:0	R	DMA15_IDX This (read-only) register determines the data items already transferred by the DMA channel. Therefore, if its value is 1, then the DMA channel has already copied one data item and it is currently performing the next copy. If its value is 2, then two items have already been copied and so on. When the transfer is completed (so when DMAx_CTRL_REG. DMA_ON is cleared) and DMAx_CTRL_REG. CIRCULAR is not set, the register keeps its (last) value (which should be equal to DMAx_LEN_REG) and it is automatically reset to 0 upon starting a new transfer. In CIRCULAR mode, the register is automatically initialized to 0 as soon as the DMA channel starts-over again.	0x0

Table 226: DMA_REQ_MUX_REG (0x40070900)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	- Reserved	0x0
28:24	R/W	DMA3_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 3	0x1F

Bit	Mode	Symbol/Description	Reset
		See DMA0_SEL for the peripherals' mapping.	
23:21	R/W	-	0x0
		Reserved	
20:16	R/W	DMA2_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 2. See DMA0_SEL for the peripherals' mapping.	
15:13	R/W	-	0x0
		Reserved	
12:8	R/W	DMA1_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 1. See DMA0_SEL for the peripherals' mapping.	
7:5	R/W	-	0x0
		Reserved	
4:0	R/W	DMA0_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 0. 0x00: SPI_RX 0x01: SPI_TX 0x02: SPI2_RX 0x03: SPI2_TX 0x04: UART_RX 0x05: UART_TX 0x06: UART2_RX 0x07: UART2_TX 0x08: UART3RX 0x09: UART3_TX 0x0A: I2C_RX 0x0B: I2C_TX 0x0C: I2C2_RX 0x0D: I2C2_TX 0x0E: AUXADC<0> 0x0F: AUXADC<1> 0x10: AUXADC<2> 0x11: AUXADC<3> 0x12: SRC_IN 0x13: SRC_OUT 0x14: DAI_TX 0x15: DAI_RX Note: If any of the 15 available peripheral selector fields for among 4 DMA_REQ_MUX registers (DMA0_SEL, DMA1_SEL, DMA2_SEL, ..., DMA14_SEL and DMA15_SEL) have the same value, the lesser significant selector has higher priority. Consequently, it is suggested to assign the intended peripheral value to a unique selector field.	

Table 227: DMA_REQ_MUX2_REG (0x40070904)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	-	0x0
		Reserved	
28:24	R/W	DMA7_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 7. See DMA0_SEL for the peripherals' mapping.	
23:21	R/W	-	0x0
		Reserved	
20:16	R/W	DMA6_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 6. See DMA0_SEL for the peripherals' mapping.	
15:13	R/W	-	0x0
		Reserved	
12:8	R/W	DMA5_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 5. See DMA0_SEL for the peripherals' mapping.	
7:5	R/W	-	0x0
		Reserved	
4:0	R/W	DMA4_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 4. See DMA0_SEL for the peripherals' mapping.	

Table 228: DMA_REQ_MUX3_REG (0x40070908)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	-	0x0
		Reserved	
28:24	R/W	DMA11_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 11. See DMA0_SEL for the peripherals' mapping.	
23:21	R/W	-	0x0
		Reserved	
20:16	R/W	DMA10_SEL	0x1F
		Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 10. See DMA0_SEL for the peripherals' mapping.	
15:13	R/W	-	0x0
		Reserved	

Bit	Mode	Symbol/Description	Reset
12:8	R/W	DMA9_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 9. See DMA0_SEL for the peripherals' mapping.	0x1F
7:5	R/W	- Reserved	0x0
4:0	R/W	DMA8_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 8. See DMA0_SEL for the peripherals' mapping.	0x1F

Table 229: DMA_REQ_MUX4_REG (0x4007090C)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	- Reserved	0x0
28:24	R/W	DMA15_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 15. See DMA0_SEL for the peripherals' mapping.	0x1F
23:21	R/W	- Reserved	0x0
20:16	R/W	DMA14_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 14. See DMA0_SEL for the peripherals' mapping.	0x1F
15:13	R/W	- Reserved	0x0
12:8	R/W	DMA13_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 13. See DMA0_SEL for the peripherals' mapping.	0x1F
7:5	R/W	- Reserved	0x0
4:0	R/W	DMA12_SEL Select which peripheral is mapped on the DMA channel. Here, the DMA request is mapped on channel 12. See DMA0_SEL for the peripherals' mapping.	0x1F

Table 230: DMA_INT_STATUS_REG (0x40070910)

Bit	Mode	Symbol/Description	Reset
31	R	DMA_BUS_ERR15 0 = No bus error response is detected for channel 15	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Bus error response detected for channel 15 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	
30	R	DMA_BUS_ERR14 0 = No bus error response is detected for channel 14 1 = Bus error response detected for channel 14 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
29	R	DMA_BUS_ERR13 0 = No bus error response is detected for channel 13 1 = Bus error response detected for channel 13 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
28	R	DMA_BUS_ERR12 0 = No bus error response is detected for channel 12 1 = Bus error response detected for channel 12 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
27	R	DMA_BUS_ERR11 0 = No bus error response is detected for channel 11 1 = Bus error response detected for channel 11 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
26	R	DMA_BUS_ERR10 0 = No bus error response is detected for channel 10 1 = Bus error response detected for channel 10 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
25	R	DMA_BUS_ERR9 0 = No bus error response is detected for channel 9 1 = Bus error response detected for channel 9 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
24	R	DMA_BUS_ERR8 0 = No bus error response is detected for channel 8 1 = Bus error response detected for channel 8 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
23	R	DMA_BUS_ERR7 0 = No bus error response is detected for channel 7 1 = Bus error response detected for channel 7 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
22	R	DMA_BUS_ERR6 0 = No bus error response is detected for channel 6 1 = Bus error response detected for channel 6	0x0

Bit	Mode	Symbol/Description	Reset
		Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	
21	R	DMA_BUS_ERR5 0 = No bus error response is detected for channel 5 1 = Bus error response detected for channel 5 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
20	R	DMA_BUS_ERR4 0 = No bus error response is detected for channel 4 1 = Bus error response detected for channel 4 Note This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
19	R	DMA_BUS_ERR3 0 = No bus error response is detected for channel 3 1 = Bus error response detected for channel 3 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
18	R	DMA_BUS_ERR2 0 = No bus error response is detected for channel 2 1 = Bus error response detected for channel 2 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
17	R	DMA_BUS_ERR1 0 = No bus error response is detected for channel 1 1 = Bus error response detected for channel 1 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
16	R	DMA_BUS_ERR0 0 = No bus error response is detected for channel 0 1 = Bus error response detected for channel 0 Note: This bit-field is auto-clear and it is initialized to 0 as soon as a new transfer is started.	0x0
15	R	DMA_IRQ_CH15 0 = IRQ on channel 15 is not set 1 = IRQ on channel 15 is set	0x0
14	R	DMA_IRQ_CH14 0 = IRQ on channel 14 is not set 1 = IRQ on channel 14 is set	0x0
13	R	DMA_IRQ_CH13 0 = IRQ on channel 13 is not set 1 = IRQ on channel 13 is set	0x0
12	R	DMA_IRQ_CH12 0 = IRQ on channel 12 is not set 1 = IRQ on channel 12 is set	0x0
11	R	DMA_IRQ_CH11 0 = IRQ on channel 11 is not set	0x0

Bit	Mode	Symbol/Description	Reset
		1 = IRQ on channel 11 is set	
10	R	DMA_IRQ_CH10 0 = IRQ on channel 10 is not set 1 = IRQ on channel 10 is set	0x0
9	R	DMA_IRQ_CH9 0 = IRQ on channel 9 is not set 1 = IRQ on channel 9 is set	0x0
8	R	DMA_IRQ_CH8 0 = IRQ on channel 8 is not set 1 = IRQ on channel 8 is set	0x0
7	R	DMA_IRQ_CH7 0 = IRQ on channel 7 is not set 1 = IRQ on channel 7 is set	0x0
6	R	DMA_IRQ_CH6 0 = IRQ on channel 6 is not set 1 = IRQ on channel 6 is set	0x0
5	R	DMA_IRQ_CH5 0 = IRQ on channel 5 is not set 1 = IRQ on channel 5 is set	0x0
4	R	DMA_IRQ_CH4 0 = IRQ on channel 4 is not set 1 = IRQ on channel 4 is set	0x0
3	R	DMA_IRQ_CH3 0 = IRQ on channel 3 is not set 1 = IRQ on channel 3 is set	0x0
2	R	DMA_IRQ_CH2 0 = IRQ on channel 2 is not set 1 = IRQ on channel 2 is set	0x0
1	R	DMA_IRQ_CH1 0 = IRQ on channel 1 is not set 1 = IRQ on channel 1 is set	0x0
0	R	DMA_IRQ_CH0 0 = IRQ on channel 0 is not set 1 = IRQ on channel 0 is set	0x0

Table 231: DMA_CLEAR_INT_REG (0x40070914)

Bit	Mode	Symbol/Description	Reset
15	R0/W	DMA_RST_IRQ_CH15 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 15; writing a 0 has no effect.	0x0
14	R0/W	DMA_RST_IRQ_CH14 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 14; writing a 0 has no effect.	0x0

Bit	Mode	Symbol/Description	Reset
13	R0/W	DMA_RST_IRQ_CH13 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 13; writing a 0 has no effect.	0x0
12	R0/W	DMA_RST_IRQ_CH12 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 12; writing a 0 has no effect.	0x0
11	R0/W	DMA_RST_IRQ_CH11 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 11; writing a 0 has no effect.	0x0
10	R0/W	DMA_RST_IRQ_CH10 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 10; writing a 0 has no effect.	0x0
9	R0/W	DMA_RST_IRQ_CH9 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 9; writing a 0 has no effect.	0x0
8	R0/W	DMA_RST_IRQ_CH8 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 8; writing a 0 has no effect.	0x0
7	R0/W	DMA_RST_IRQ_CH7 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 7; writing a 0 has no effect.	0x0
6	R0/W	DMA_RST_IRQ_CH6 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 6; writing a 0 has no effect.	0x0
5	R0/W	DMA_RST_IRQ_CH5 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 5; writing a 0 has no effect.	0x0
4	R0/W	DMA_RST_IRQ_CH4 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 4; writing a 0 has no effect.	0x0
3	R0/W	DMA_RST_IRQ_CH3 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 3; writing a 0 has no effect.	0x0
2	R0/W	DMA_RST_IRQ_CH2 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 2; writing a 0 has no effect.	0x0
1	R0/W	DMA_RST_IRQ_CH1 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 1; writing a 0 has no effect.	0x0
0	R0/W	DMA_RST_IRQ_CH0 Writing a 1 resets the status bit of DMA_INT_STATUS_REG for channel 0; writing a 0 has no effect.	0x0

Table 232: DMA_INT_MASK_REG (0x40070918)

Bit	Mode	Symbol/Description	Reset
15	R/W	DMA_IRQ_ENABLE15 0 = Disable interrupts on channel 15 1 = Enable interrupts on channel 15	0x0
14	R/W	DMA_IRQ_ENABLE14 0 = Disable interrupts on channel 14 1 = Enable interrupts on channel 14	0x0
13	R/W	DMA_IRQ_ENABLE13 0 = Disable interrupts on channel 13 1 = Enable interrupts on channel 13	0x0
12	R/W	DMA_IRQ_ENABLE12 0 = Disable interrupts on channel 12 1 = Enable interrupts on channel 12	0x0
11	R/W	DMA_IRQ_ENABLE11 0 = Disable interrupts on channel 11 1 = Enable interrupts on channel 11	0x0
10	R/W	DMA_IRQ_ENABLE10 0 = Disable interrupts on channel 10 1 = Enable interrupts on channel 10	0x0
9	R/W	DMA_IRQ_ENABLE9 0 = Disable interrupts on channel 9 1 = Enable interrupts on channel 9	0x0
8	R/W	DMA_IRQ_ENABLE8 0 = Disable interrupts on channel 8 1 = Enable interrupts on channel 8	0x0
7	R/W	DMA_IRQ_ENABLE7 0 = Disable interrupts on channel 7 1 = Enable interrupts on channel 7	0x0
6	R/W	DMA_IRQ_ENABLE6 0 = Disable interrupts on channel 6 1 = Enable interrupts on channel 6	0x0
5	R/W	DMA_IRQ_ENABLE5 0 = Disable interrupts on channel 5 1 = Enable interrupts on channel 5	0x0
4	R/W	DMA_IRQ_ENABLE4 0 = Disable interrupts on channel 4 1 = Enable interrupts on channel 4	0x0
3	R/W	DMA_IRQ_ENABLE3 0 = Disable interrupts on channel 3 1 = Enable interrupts on channel 3	0x0
2	R/W	DMA_IRQ_ENABLE2 0 = Disable interrupts on channel 2 1 = Enable interrupts on channel 2	0x0

Bit	Mode	Symbol/Description	Reset
1	R/W	DMA_IRQ_ENABLE1 0 = Disable interrupts on channel 1 1 = Enable interrupts on channel 1	0x0
0	R/W	DMA_IRQ_ENABLE0 0 = Disable interrupts on channel 0 1 = Enable interrupts on channel 0	0x0

10.9 GPIO Registers

Table 233: Register map GPIO

Address	Register	Description
0x400b0000	P0_DATA_REG	P0 Data input/output Register
0x400b0004	P1_DATA_REG	P1 Data input/output Register
0x400b0008	SW_DATA_REG	Software Data input/output Register
0x400b000c	P0_SET_DATA_REG	P0 Set port pins Register
0x400b0010	P1_SET_DATA_REG	P1 Set port pins Register
0x400b0014	SW_SET_DATA_REG	SW Set port pins Register
0x400b0018	P0_RESET_DATA_REG	P0 Reset port pins Register
0x400b001c	P1_RESET_DATA_REG	P1 Reset port pins Register
0x400b0020	SW_RESET_DATA_REG	SW Reset port pins Register
0x400b0024	P0_00_MODE_REG	P0_00 Mode Register
0x400b0028	P0_01_MODE_REG	P0_01 Mode Register
0x400b002c	P0_02_MODE_REG	P0_02 Mode Register
0x400b0030	P0_03_MODE_REG	P0_03 Mode Register
0x400b0034	P0_04_MODE_REG	P0_04 Mode Register
0x400b0038	P0_05_MODE_REG	P0_05 Mode Register
0x400b003c	P0_06_MODE_REG	P0_06 Mode Register
0x400b0040	P0_07_MODE_REG	P0_07 Mode Register
0x400b0044	P0_08_MODE_REG	P0_08 Mode Register
0x400b0048	P0_09_MODE_REG	P0_09 Mode Register
0x400b004c	P0_10_MODE_REG	P0_10 Mode Register
0x400b0050	P0_11_MODE_REG	P0_11 Mode Register
0x400b0054	P0_12_MODE_REG	P0_12 Mode Register
0x400b0058	P0_13_MODE_REG	P0_13 Mode Register
0x400b005c	P1_00_MODE_REG	P1_00 Mode Register
0x400b0060	P1_01_MODE_REG	P1_01 Mode Register
0x400b0064	P1_02_MODE_REG	P1_02 Mode Register
0x400b0068	P1_03_MODE_REG	P1_03 Mode Register
0x400b006c	P1_04_MODE_REG	P1_04 Mode Register
0x400b0070	P1_05_MODE_REG	P1_05 Mode Register
0x400b0074	P1_06_MODE_REG	P1_06 Mode Register
0x400b0078	P1_07_MODE_REG	P1_07 Mode Register

Address	Register	Description
0x400b007c	P1_08_MODE_REG	P1_08 Mode Register
0x400b0080	P1_09_MODE_REG	P1_09 Mode Register
0x400b0084	P1_10_MODE_REG	P1_10 Mode Register
0x400b0088	P1_11_MODE_REG	P1_11 Mode Register
0x400b008c	P1_12_MODE_REG	P1_12 Mode Register
0x400b0090	P1_13_MODE_REG	P1_13 Mode Register
0x400b0094	P1_14_MODE_REG	P1_14 Mode Register
0x400b0098	P1_15_MODE_REG	P1_15 Mode Register
0x400b009c	SW0_MODE_REG	SWCLK pin Register
0x400b00a0	SW1_MODE_REG	SWDIO Data Register
0x400b00a4	GPIO_CLK_SEL_REG	Select which clock to map on ports P0/P1
0x400b00a8	EMMC_MODE_REG	OQSPI PAD control Register
0x400b00ac	SDIO_MODE_REG	QSPI PAD Control Register
0x400b00c0	GPIO_INT_SEL_P0_REG	Select which inputs from P0 port can trigger wake-up counter
0x400b00c4	GPIO_INT_SEL_P1_REG	Select which inputs from P1 port can trigger wake-up counter
0x400b00c8	GPIO_INT_POL_P0_REG	Select the sesitivity polarity for each P0 input
0x400b00cc	GPIO_INT_POL_P1_REG	Select the sesitivity polarity for each P1 input
0x400b00d0	GPIO_INT_STS_P0_REG	Event status register for P0
0x400b00d4	GPIO_INT_STS_P1_REG	Event status register for P1
0x400b00d8	GPIO_INT_CLR_P0_REG	Clear event register for P0
0x400b00dc	GPIO_INT_CLR_P1_REG	Clear event register for P1
0x400b00e0	GPIO_SEL_P0_REG	Enable fast wake-up and enable GPIO_P0_IRQ
0x400b00e4	GPIO_SEL_P1_REG	Enable fast wake-up and enable GPIO_P1_IRQ
0x400b00e8	GPIO_SEL1_P0_REG	Configure to generate level or edge sensitive IRQ on P0 events
0x400b00ec	GPIO_SEL1_P1_REG	Configure to generate level or edge sensitive IRQ on P1 events

Table 234: P0_DATA_REG (0x400B0000)

Bit	Mode	Symbol/Description	Reset
13:0	R/W	P0_DATA	0x0
		Set P0 output register when written; Return the value of P0 port when read.	

Table 235: P1_DATA_REG (0x400B0004)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	P1_DATA	0x0
		Set P1 output register when written; Return the value of P1 port when read.	

Table 236: SW_DATA_REG (0x400B0008)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	SWD_DATA Set SWD_[y] output register when written; Return the value of SWD_[y] port when read.	0x0

Table 237: P0_SET_DATA_REG (0x400B000C)

Bit	Mode	Symbol/Description	Reset
13:0	WS	P0_SET Writing 1 to P0[y] sets P0[y] to 1. Writing 0 is discarded; Reading returns 0.	0x0

Table 238: P1_SET_DATA_REG (0x400B0010)

Bit	Mode	Symbol/Description	Reset
15:0	WS	P1_SET Writing 1 to P1[y] sets P1[y] to 1. Writing 0 is discarded; Reading returns 0.	0x0

Table 239: SW_SET_DATA_REG (0x400B0014)

Bit	Mode	Symbol/Description	Reset
1:0	WS	SWD_SET Writing 1 to SWD_[y] sets SWD_[y] to 1. Writing 0 is discarded; Reading returns 0.	0x0

Table 240: P0_RESET_DATA_REG (0x400B0018)

Bit	Mode	Symbol/Description	Reset
13:0	WS	P0_RESET Writing 1 to P0[y] sets P0[y] to 0. Writing 0 is discarded; Reading returns 0.	0x0

Table 241: P1_RESET_DATA_REG (0x400B001C)

Bit	Mode	Symbol/Description	Reset
15:0	WS	P1_RESET Writing 1 to P1[y] sets P1[y] to 0. Writing 0 is discarded; Reading returns 0.	0x0

Table 242: SW_RESET_DATA_REG (0x400B0020)

Bit	Mode	Symbol/Description	Reset
1:0	WS	SWD_RESET Writing 1 to SWD_[y] sets SWD_[y] to 0. Writing 0 is discarded; Reading returns 0.	0x0

Table 243: P0_00_MODE_REG (0x400B0024)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID 0: GPIO (I/O) 1: UART_RX (IA) 2: UART_TX (OA) 3: UART_CTSN (IA) 4: UART_RTSN (OA) 5: UART_TXDOE (OA) 6: UART1_RX (IA) 7: UART1_TX (OA) 8: UART1_CTSN (IA) 9: UART1_RTSN (OA) 10: UART1_TXDOE (OA) 11: UART2_RX (IA) 12: UART2_TX (OA) 13: UART2_CTSN (IA) 14: UART2_RTSN (OA) 15: UART2_TXDOE (OA)	0x0

Bit	Mode	Symbol/Description	Reset
		16: SPI_DI (IA) 17: SPI_DO (OA) 18: SPI_CLK (I/O) 19: SPI_CSN0 (I/O) 20: SPI_CSN1 (OA) 21: SPI2_DI (IA) 22: SPI2_DO (OA) 23: SPI2_CLK (I/O) 24: SPI2_CSN0 (I/O) 25: SPI2_CSN1 (OA) 26: I2C_SCL (I/O) 27: I2C_SDA (IO-OD) 28: I2C2_SCL (IO-OD) 29: I2C2_SDA (I/O-OD) 30: Analog (ADC, XTAL32Kmonitoring, see Pinout) 31: PCM_DI (I) 32: PCM_DO 33: PCM_FSC (I/O) 34: PCM_CLK (I/O) 35: DMICA_DI (I) 36: DMIC_CLK (I/O) 37: MCLK 38: TIM_PWM (OA) 39: TIM2_PWM (OA) 40: TIM3_PWM (OA) 41: TIM4_PWM (OA) 42: TIM5_PWM (OA) 43: TIM6_PWM (OA) 44: TIM7_PWM (OA) 45: TIM8_PWM (OA) 46: TIM_1SHOT (OA) 47: TIM2_1SHOT (OA) 48: TIM3_1SHOT (OA) 49: TIM4_1SHOT (OA) 50: TIM5_1SHOT (OA) 51: TIM6_1SHOT (OA) 52: TIM7_1SHOT (OA) 53: TIM8_1SHOT (OA) 54: CLOCK (see GPIO_CLK_SEL_REG) 55: FEM_BS (O) 56: FEM_CS (O) 57: FEM_CTRL0 (O) 58: FEM_CTRL1 (O) 59: FEM_CTRL2 (O) 60: BT_COEX_CBT (O) 61: BT_WLAN_ACT (O) 62: BT_ACT (I) 63: BT_PRI (I)	

Bit	Mode	Symbol/Description	Reset
		64: RF_SW1 (O) 65: RF_SW2 (O) 66: EXT_INTR (O) 67 - 98: DBG_WIFI0 - DBG_WIFI31 (O) 99: SWCLK (I) 100: SWDIO (I/O) 101: WPROTECT (I) 102: CDETECT (I) 103: ZB_WLAN_ACT (O) 104: ZB_ACT (I) 105: ZB_PRI (I) 106: BTCOEX_ASC0 (O) 107: BTCOEX_ASC1 (O) 108: BTCOEX_ASC2 (O) (I): Input, (I/O): Input/Output, if nothing mentioned: Output (default) IA: Pad direction is automatically set to input when PID selected OA: Pad direction is automatically set to output when PID selected In other case, set the pad direction with PUPD bits	

Table 244: P0_01_MODE_REG (0x400B0028)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected	0x0

Bit	Mode	Symbol/Description	Reset
		10: Input, pull-down selected 11: Output, no resistors selected	
7	R/W	-	0x0
		Reserved	
6:0	R/W	PID	0x1E
		See P0_00_MODE_REG[PID]	

Table 245: P0_02_MODE_REG (0x400B002C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE	0x0
		Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	
14	R/W	SR	0x0
		Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	
13:12	R/W	DS	0x2
		Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS	0x0
		Standard PAD input selection. 0: CMOS input 1: Schmitt input	
10	R/W	PPOD	0x0
		0: Push pull 1: Open drain	
9:8	R/W	PUPD	0x0
		00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected In ADC mode, these bits are don't care.	
7	R/W	-	0x0
		Reserved	
6:0	R/W	PID	0x1E
		See P0_00_MODE_REG[PID].	

Table 246: P0_03_MODE_REG (0x400B0030)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE	0x0

Bit	Mode	Symbol/Description	Reset
		Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x0
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x1E

Table 247: P0_04_MODE_REG (0x400B0034)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA	0x2

Bit	Mode	Symbol/Description	Reset
		2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 248: P0_05_MODE_REG (0x400B0038)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD	0x2

Bit	Mode	Symbol/Description	Reset
		00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	
7	R/W	-	0x0
		Reserved	
6:0	R/W	PID	0x0
		See P0_00_MODE_REG[PID].	

Table 249: P0_06_MODE_REG (0x400B003C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE	0x0
		Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	
14	R/W	SR	0x0
		Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	
13:12	R/W	DS	0x2
		Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS	0x0
		Standard PAD input selection. 0: CMOS input 1: Schmitt input	
10	R/W	PPOD	0x0
		0: Push pull 1: Open drain	
9:8	R/W	PUPD	0x2
		00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	
7	R/W	-	0x0
		Reserved	
6:0	R/W	PID	0x0
		See P0_00_MODE_REG[PID].	

Table 250: P0_07_MODE_REG (0x400B0040)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 251: P0_08_MODE_REG (0x400B0044)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength.	0x2

Bit	Mode	Symbol/Description	Reset
		0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected In ADC mode, these bits are don't care.	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 252: P0_09_MODE_REG (0x400B0048)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull	0x0

Bit	Mode	Symbol/Description	Reset
		1: Open drain	
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 253: P0_10_MODE_REG (0x400B004C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 254: P0_11_MODE_REG (0x400B0050)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 255: P0_12_MODE_REG (0x400B0054)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength.	0x2

Bit	Mode	Symbol/Description	Reset
		0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 256: P0_13_MODE_REG (0x400B0058)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0

Bit	Mode	Symbol/Description	Reset
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 257: P1_00_MODE_REG (0x400B005C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 258: P1_01_MODE_REG (0x400B0060)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 259: P1_02_MODE_REG (0x400B0064)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength.	0x2

Bit	Mode	Symbol/Description	Reset
		0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 260: P1_03_MODE_REG (0x400B0068)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0

Bit	Mode	Symbol/Description	Reset
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 261: P1_04_MODE_REG (0x400B006C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 262: P1_05_MODE_REG (0x400B0070)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 263: P1_06_MODE_REG (0x400B0074)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength.	0x2

Bit	Mode	Symbol/Description	Reset
		0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 264: P1_07_MODE_REG (0x400B0078)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0

Bit	Mode	Symbol/Description	Reset
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 265: P1_08_MODE_REG (0x400B007C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 266: P1_09_MODE_REG (0x400B0080)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 267: P1_10_MODE_REG (0x400B0084)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength.	0x2

Bit	Mode	Symbol/Description	Reset
		0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 268: P1_11_MODE_REG (0x400B0088)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0

Bit	Mode	Symbol/Description	Reset
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 269: P1_12_MODE_REG (0x400B008C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 270: P1_13_MODE_REG (0x400B0090)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 271: P1_14_MODE_REG (0x400B0094)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength.	0x2

Bit	Mode	Symbol/Description	Reset
		0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected In ADC mode, these bits are don't care.	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 272: P1_15_MODE_REG (0x400B0098)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull	0x0

Bit	Mode	Symbol/Description	Reset
		1: Open drain	
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x0

Table 273: SW0_MODE_REG (0x400B009C)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x63

Table 274: SW1_MODE_REG (0x400B00A0)

Bit	Mode	Symbol/Description	Reset
15	R/W	POE Standard PAD Parametric Output control, parametric inverted data. 0: Disable 1: Enable	0x0
14	R/W	SR Standard PAD slew rate control. 0: Fast slew rate 1: Slow slew rate	0x0
13:12	R/W	DS Standard PAD drive strength. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x2
11	R/W	IS Standard PAD input selection. 0: CMOS input 1: Schmitt input	0x0
10	R/W	PPOD 0: Push pull 1: Open drain	0x0
9:8	R/W	PUPD 00: Input, no resistors selected 01: Input, pull-up selected 10: Input, pull-down selected 11: Output, no resistors selected	0x2
7	R/W	- Reserved	0x0
6:0	R/W	PID See P0_00_MODE_REG[PID].	0x64

Table 275: GPIO_CLK_SEL_REG (0x400B00A4)

Bit	Mode	Symbol/Description	Reset
10	R/W	MCLK_OUTPUT_EN MCLK output enable bit-field. When set, it enables the mapping of MCLK clock on dedicated GPIO (P1_11). The specific GPIO must be configured as GPIO output.	0x0
9	R/W	DPLL480M_OUTPUT_EN DPLL480M output enable bit-field. When set, it enables the mapping of dpll480m clock on dedicated GPIO (P0_13). The specific GPIO must be configured as GPIO output.	0x0
8	R/W	FPLL98M_OUTPUT_EN FPLL98M output enable bit-field. When set, it enables the mapping of fpll98m clock on dedicated GPIO (P0_10). The specific GPIO must be configured as GPIO output.	0x0

Bit	Mode	Symbol/Description	Reset
7	R/W	OSC32K_OUTPUT_EN OSC32K output enable bit-field. When set, it enables the mapping of osc32k clock on dedicated GPIO (P0_12). The specific GPIO must be configured as GPIO output.	0x0
6	R/W	XTAL32K_OUTPUT_EN XTAL32K output enable bit-field. When set, it enables the mapping of xtal32k clock on dedicated GPIO (P0_08). The specific GPIO must be configured as GPIO output.	0x0
5	R/W	RC10M_OUTPUT_EN RC10M_CLK output enable bit-field. When set, it enables the mapping of rc10m clock on dedicated GPIO (P0_09). The specific GPIO must be configured as GPIO output.	0x0
4	R/W	XTAL40M_OUTPUT_EN XTAL40M_CLK output enable bit-field. When set, it enables the mapping of xtal40m clock on dedicated GPIO (P0_11). The specific GPIO must be configured as GPIO output.	0x0
3	R/W	FUNC_CLOCK_EN If set, it enables the mapping of the selected clock signal according to FUNC_CLOCK_SEL bit-field.	0x0
2:0	R/W	FUNC_CLOCK_SEL Select which clock to map when PID = FUNC_CLOCK. 0x1: xtal40m 0x2: rc10m 0x3: xtal32k 0x4: osc32k 0x5: fpll98m 0x6: dpll480m 0x7: divn_clk Others: Reserved	0x0

Table 276: EMMC_MODE_REG (0x400B00A8)

Bit	Mode	Symbol/Description	Reset
29:20	R/W	EMMC_PULL_SEL eMMC PADS mode Pull selection. 1: Pull-up 0: Pull-down bit[20]: eMMC_CLK Pad bit[21]: eMMC_CMD Pad bit[22]: eMMC_DIO0 Pad bit[23]: eMMC_DIO1 Pad bit[24]: eMMC_DIO2 Pad bit[25]: eMMC_DIO3 Pad bit[26]: eMMC_DIO4 Pad bit[27]: eMMC_DIO5 Pad bit[28]: eMMC_DIO6 Pad bit[29]: eMMC_DIO7 Pad	0x0
19:18	R/W	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
17:8	R/W	EMMC_PULL_EN eMMC PADS mode Pull enable. 1: Pull enable 0: Pull disable bit[8]: eMMC_CLK Pad bit[9]: eMMC_CMD Pad bit[10]: eMMC_DIO0 Pad bit[11]: eMMC_DIO1 Pad bit[12]: eMMC_DIO2 Pad bit[13]: eMMC_DIO3 Pad bit[14]: eMMC_DIO4 Pad bit[15]: eMMC_DIO5 Pad bit[16]: eMMC_DIO6 Pad bit[17]: eMMC_DIO7 Pad	0x3FF
7	R/W	EMMC_CLKIN_SEL EMMC rxclk_in selection. 0: clkout signal 1: clkout_inv signal	0x0
6	R/W	EMMC_WP_VAL EMMC Write Protect value.	0x0
5	R/W	EMMC_CD_VAL EMMC Card Detect value.	0x0
4	R/W	EMMC_WP_SEL EMMC Write Protect selection. 0: WP signal from register set value 1: WP signal from PIN	0x0
3	R/W	EMMC_CD_SEL EMMC Card Detect selection. 0: CD signal from register set value 1: CD signal from PIN	0x0
2	R/W	EMMC_OCTA_MODE EMMC Octa mode enable. 0: eMMC Quad mode (default) 1: eMMC Octa mode with P1_[3:0] or P0_[7:4]	0x0
1	R/W	EMMC_PORT_SEL eMMC Port selection when set to high SDIO_ENABLE. 0: eMMC Port P0_[13:4] 1: eMMC Port P1_[3:0] and P1_[15:10] (default)	0x1
0	R/W	EMMC_ENABLE EMMC/GPIO PADs Mode enable. 0: GPIO mode (default) 1: eMMC mode with {P1_[3:0], P1_[15:10]} or P0_[13:4]	0x0

Table 277: SDIO_MODE_REG (0x400B00AC)

Bit	Mode	Symbol/Description	Reset
15:10	R/W	SDIO_PULL_SEL SDIO PADS mode Pull selection. 1: Pull-up 0: Pull-down bit[10]: SDIO_CLK Pad bit[11]: SDIO_CMD Pad bit[12]: SDIO_DIO0 Pad bit[13]: SDIO_DIO1 Pad bit[14]: SDIO_DIO2 Pad bit[15]: SDIO_DIO3 Pad	0x0
9:4	R/W	SDIO_PULL_EN SDIO PADS mode Pull enable. 1: Pull enable 0: Pull disable bit[4]: SDIO_CLK Pad bit[5]: SDIO_CMD Pad bit[6]: SDIO_DIO0 Pad bit[7]: SDIO_DIO1 Pad bit[8]: SDIO_DIO2 Pad bit[9]: SDIO_DIO3 Pad	0x3F
3:2	R/W	CFG_DRV SDIO PADS mode drive strength. 0: 2 mA at 3.3 V 1: 4 mA at 3.3 V 2: 8 mA at3.3 V 3: 14 mA at 3.3 V	0x0
1	R/W	SDIO_PORT_SEL SDIO Port selection when set to high SDIO_ENABLE. 0: SDIO Port P0_[13:8] 1: SDIO Port P1_[15:10]	0x0
0	R/W	SDIO_PAD_ENABLE SDIO/GPIO PADs mode enable. 0: GPIO mode 1: SDIO mode	0x0

Table 278: GPIO_INT_SEL_P0_REG (0x400B00C0)

Bit	Mode	Symbol/Description	Reset
13:0	R/W	GPIO_SELECT_P0 0: Input P0_xx is not enabled for wake-up event. 1: Input P0_xx is enabled for wake-up event.	0x0

Table 279: GPIO_INT_SEL_P1_REG (0x400B00C4)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	GPIO_SELECT_P1	0x0

Bit	Mode	Symbol/Description	Reset
		0: Input P1_xx is not enabled for wake-up event. 1: Input P1_xx is enabled for wake-up event.	

Table 280: GPIO_INT_POL_P0_REG (0x400B00C8)

Bit	Mode	Symbol/Description	Reset
13:0	R/W	GPIO_POL_P0 0: Enabled input P0_xx gives an event if that input goes high 1: Enabled input P0_xx gives an event if that input goes low	0x0

Table 281: GPIO_INT_POL_P1_REG (0x400B00CC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	GPIO_POL_P1 0: Enabled input P1_xx gives an event if that input goes high 1: Enabled input P1_xx gives an event if that input goes low	0x0

Table 282: GPIO_INT_STS_P0_REG (0x400B00D0)

Bit	Mode	Symbol/Description	Reset
13:0	R	GPIO_STAT_P0 Contain the latched value of any toggle of the GPIOs Port P0. WKUP_STAT_P0[0] -> P0_00.	0x0

Table 283: GPIO_INT_STS_P1_REG (0x400B00D4)

Bit	Mode	Symbol/Description	Reset
15:0	R	GPIO_STAT_P1 Contain the latched value of any toggle of the GPIOs Port P1. WKUP_STAT_P1[0] -> P1_00.	0x0

Table 284: GPIO_INT_CLR_P0_REG (0x400B00D8)

Bit	Mode	Symbol/Description	Reset
13:0	W	GPIO_CLEAR_P0 Clear latched value of the GPIOs P0 when corresponding bit is 1.	0x0

Table 285: GPIO_INT_CLR_P1_REG (0x400B00DC)

Bit	Mode	Symbol/Description	Reset
15:0	W	GPIO_CLEAR_P1 Clear latched value of the GPIOs P1 when corresponding bit is 1.	0x0

Table 286: GPIO_SEL_P0_REG (0x400B00E0)

Bit	Mode	Symbol/Description	Reset
13:0	R/W	GPIO_SEL_P0 0: No GPIO_P0_IRQ on input P0_x.	0x0

Bit	Mode	Symbol/Description	Reset
		Fast wake-up is not enabled if the corresponding WKUP_SEL1_GPIO_P0_REG[x] is 0 too. 1: GPIO_P0_IRQ is generated on P0_x input event. If WKUP_SEL1_GPIO_P0_REG[x] is 0, IRQ generation is level sensitive. If WKUP_SEL1_GPIO_P0_REG[x] is 1, IRQ generation is edge sensitive (only if there is a change on P0_x input). Fast wake-up from the corresponding P0_x input is enabled.	

Table 287: GPIO_SEL_P1_REG (0x400B00E4)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	GPIO_SEL_P1 0: No GPIO_P1_IRQ on input P1_x. Fast wake-up is not enabled if the corresponding WKUP_SEL1_GPIO_P1_REG[x] is 0 too. 1: GPIO_P1_IRQ is generated on P1_x input event. If WKUP_SEL1_GPIO_P1_REG[x] is 0, IRQ generation is level sensitive. If WKUP_SEL1_GPIO_P1_REG[x] is 1, IRQ generation is edge sensitive (only if there is a change on P1_x input). Fast wake-up from the corresponding P1_x input is enabled.	0x0

Table 288: GPIO_SEL1_P0_REG (0x400B00E8)

Bit	Mode	Symbol/Description	Reset
13:0	R/W	GPIO_SEL1_P0 0 (level sensitive): If WKUP_SEL_GPIO_P0_REG[x] is 1, generate GPIO_P0_IRQ based on P0_x level. Fast wake-up is not enabled if the corresponding WKUP_SEL_GPIO_P0_REG[x] is 0 too. 1 (edge sensitive): If WKUP_SEL_GPIO_P0_REG[x] is 1, GPIO_P0_IRQ is generated only on rising/falling (defined by WKUP_POL_P0_REG) edge on P0_x. Fast wake-up from the corresponding P0_x input is enabled.	0x0

Table 289: GPIO_SEL1_P1_REG (0x400B00EC)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	GPIO_SEL1_P1 0 (level sensitive): If WKUP_SEL_GPIO_P1_REG[x] is 1, generate GPIO_P1_IRQ based on P1_x level. Fast wake-up is not enabled if the corresponding WKUP_SEL_GPIO_P1_REG[x] is 0 too. 1 (edge sensitive): If WKUP_SEL_GPIO_P1_REG[x] is 1, GPIO_P1_IRQ is generated only on rising/falling (defined by WKUP_POL_P1_REG) edge on P1_x. Fast wake-up from the corresponding P1_x input is enabled.	0x0

10.10 General Purpose System Status Registers

Table 290: Register map GPREG

Address	Register	Description
0x40070308	DEBUG_REG	Various debug information register.
0x4007030c	GP_STATUS_REG	General purpose system status register.

Table 291: DEBUG_REG (0x40070308)

Bit	Mode	Symbol/Description	Reset
31:16	R	-	0x0
		Reserved	
15:5	R	-	0x0
		Reserved	
4	R/W	ETM_TRACE_MAP_ON_PINS_EN	0x0
		1: ETM/TPIU Trace signals mapped on GPIO pins is enabled.	
3	R/W	SYS_CPUWAIT_ON_JTAG	0x0
		1: Stall the processor core out of reset (**only after a wake-up from JTAG**). Debugger access continues when the core is stalled. When set to 0 again, the core resumes instruction execution. This feature is independent of the Power Domain Controller (PDC) settings. If this bit is set and there is SW/JTAG activity during deep sleep, the SYS CPU is stalled after the wake-up. Note: This bit is retained.	
2	R/W	SYS_CPUWAIT	0x0
		1: Stall the processor core out of reset (**always after a wake-up**). Debugger access continue when the core is stalled. When set to 0 again the core resumes instruction execution. Note: This bit is retained.	
1	R	SYS_CPU_IS_HALTED	0x0
		1: SYS CPU (Arm CM33) is halted.	
0	R/W	SYS_CPU_FREEZE_EN	0x1
		1: Enable Freezing on-chip peripherals (see Note 2) by the SYS CPU (Arm CM33). Default 1, freezing of the on-chip peripherals is enabled when the Cortex-M33 is halted in DEBUG state. If 0, freezing of the on-chip peripherals is only depending on [RE]SET_FREEZE_REG except the system watchdog timer. The system watchdog timer is always frozen when the Cortex-M33 is halted in DEBUG state. Note 1: This bit is retained. Note 2: See [RE]SET_FREEZE_REG for the specific on-chip peripherals.	

Table 292: GP_STATUS_REG (0x4007030C)

Bit	Mode	Symbol/Description	Reset
31:2	R	-	0x0
		Reserved	
1	R/W	-	0x0
		Reserved	
0	R/W	CAL_PHASE	0x0
		If 1, it designates that the chip is in Calibration phase, for example, the OTP is initially programmed but no Calibration has occurred.	

10.11 I2C Registers

Table 293: Register map I2C

Address	Register	Description
0x40090000	I2C_CON_REG	I2C Control Register
0x40090004	I2C_TAR_REG	I2C Target Address Register
0x40090008	I2C_SAR_REG	I2C Slave Address Register
0x4009000c	I2C_HS_MADDR_REG	I2C High Speed Master Mode Code Address Register
0x40090010	I2C_DATA_CMD_REG	I2C RX/TX Data Buffer and Command Register
0x40090014	I2C_SS_SCL_HCNT_REG	Standard Speed I2C Clock SCL High Count Register
0x40090018	I2C_SS_SCL_LCNT_REG	Standard Speed I2C Clock SCL Low Count Register
0x4009001c	I2C_FS_SCL_HCNT_REG	Fast Speed I2C Clock SCL High Count Register
0x40090020	I2C_FS_SCL_LCNT_REG	Fast Speed I2C Clock SCL Low Count Register
0x40090024	I2C_HS_SCL_HCNT_REG	High Speed I2C Clock SCL High Count Register
0x40090028	I2C_HS_SCL_LCNT_REG	High Speed I2C Clock SCL Low Count Register
0x4009002c	I2C_INTR_STAT_REG	I2C Interrupt Status Register
0x40090030	I2C_INTR_MASK_REG	I2C Interrupt Mask Register
0x40090034	I2C_RAW_INTR_STAT_REG	I2C Raw Interrupt Status Register
0x40090038	I2C_RX_TL_REG	I2C Receive FIFO Threshold Register
0x4009003c	I2C_TX_TL_REG	I2C Transmit FIFO Threshold Register
0x40090040	I2C_CLR_INTR_REG	Clear Combined and Individual Interrupt Register
0x40090044	I2C_CLR_RX_UNDER_REG	Clear RX_UNDER Interrupt Register
0x40090048	I2C_CLR_RX_OVER_REG	Clear RX_OVER Interrupt Register
0x4009004c	I2C_CLR_TX_OVER_REG	Clear TX_OVER Interrupt Register
0x40090050	I2C_CLR_RD_REQ_REG	Clear RD_REQ Interrupt Register
0x40090054	I2C_CLR_TX_ABRT_REG	Clear TX_ABRT Interrupt Register
0x40090058	I2C_CLR_RX_DONE_REG	Clear RX_DONE Interrupt Register
0x4009005c	I2C_CLR_ACTIVITY_REG	Clear ACTIVITY Interrupt Register
0x40090060	I2C_CLR_STOP_DET_REG	Clear STOP_DET Interrupt Register
0x40090064	I2C_CLR_START_DET_REG	Clear START_DET Interrupt Register
0x40090068	I2C_CLR_GEN_CALL_REG	Clear GEN_CALL Interrupt Register
0x4009006c	I2C_ENABLE_REG	I2C Enable Register

Address	Register	Description
0x40090070	I2C_STATUS_REG	I2C Status Register
0x40090074	I2C_TXFLR_REG	I2C Transmit FIFO Level Register
0x40090078	I2C_RXFLR_REG	I2C Receive FIFO Level Register
0x4009007c	I2C_SDA_HOLD_REG	I2C SDA Hold Time Length Register
0x40090080	I2C_TX_ABRT_SOURCE_REG	I2C Transmit Abort Source Register
0x40090088	I2C_DMA_CR_REG	DMA Control Register
0x4009008c	I2C_DMA_TDLR_REG	DMA Transmit Data Level Register
0x40090090	I2C_DMA_RDLR_REG	I2C Receive Data Level Register
0x40090094	I2C_SDA_SETUP_REG	I2C SDA Setup Register
0x40090098	I2C_ACK_GENERAL_CALL_REG	I2C ACK General Call Register
0x4009009c	I2C_ENABLE_STATUS_REG	I2C Enable Status Register
0x400900a0	I2C_IC_FS_SPKLEN_REG	I2C SS and FS spike suppression limit Size
0x400900a4	I2C_IC_HS_SPKLEN_REG	I2C HS spike suppression limit Size

Table 294: I2C_CON_REG (0x40090000)

Bit	Mode	Symbol/Description	Reset
31:11	-	-	0x0
		Reserved	
10	R	I2C_STOP_DET_IF_MASTER_ACTIVE	0x0
		In Master mode: 1 = Issue the STOP_DET interrupt only when master is active. 0 = Issue the STOP_DET irrespective of whether master is active or not.	
9	R/W	I2C_RX_FIFO_FULL_HLD_CTRL	0x0
		This bit controls whether DW_apb_i2c should hold the bus when the RX FIFO is physically full to its RX_BUFFER_DEPTH. 1 = Hold bus when RX_FIFO is full. 0 = Overflow when RX_FIFO is full.	
8	R/W	I2C_TX_EMPTY_CTRL	0x0
		This bit controls the generation of the TX_EMPTY interrupt as described in the IC_RAW_INTR_STAT register. 1 = Controlled generation of TX_EMPTY interrupt. 0 = Default behaviour of TX_EMPTY interrupt.	
7	R/W	I2C_STOP_DET_IFADDRESSED	0x0
		1 = Slave issues STOP_DET intr only if addressed. 0 = Slave issues STOP_DET intr always. During a general call address, this slave does not issue the STOP_DET interrupt if STOP_DET_IF_ADDRESSED = 1'b1, even if the slave responds to the general call address by generating ACK. The STOP_DET interrupt is generated only when the transmitted address matches the slave address (SAR).	
6	R/W	I2C_SLAVE_DISABLE	0x1

Bit	Mode	Symbol/Description	Reset
		Slave enabled or disabled after reset is applied, which means software does not have to configure the slave. 0 = Slave is enabled. 1 = Slave is disabled. Software should ensure that if this bit is written with 0, then bit 0 should also be written with 0.	
5	R/W	I2C_RESTART_EN Determine whether RESTART conditions may be sent when acting as a master. 0 = Disable 1 = Enable	0x1
4	R/W	I2C_10BITADDR_MASTER Control whether the controller starts its transfers in 7- or 10-bit addressing mode when acting as a master. 0 = 7-bit addressing 1 = 10-bit addressing	0x1
3	R/W	I2C_10BITADDR_SLAVE When acting as a slave, this bit controls whether the controller responds to 7- or 10-bit addresses. 0 = 7-bit addressing 1 = 10-bit addressing	0x1
2:1	R/W	I2C_SPEED These bits control at which speed the controller operates. 1 = Standard mode (100 kbit/s) 2 = Fast mode (400 kbit/s) 3 = High-speed mode	0x3
0	R/W	I2C_MASTER_MODE This bit controls whether the controller master is enabled. 0 = Master disabled 1 = Master enabled Software should ensure that if this bit is written with 1 then bit 6 should also be written with 1.	0x1

Table 295: I2C_TAR_REG (0x40090004)

Bit	Mode	Symbol/Description	Reset
31:12	-	- Reserved	0x0
11	R/W	SPECIAL On read This bit indicates whether software performs a General Call or START BYTE command. 0 = Ignore bit 10 GC_OR_START and use IC_TAR normally. 1 = Perform special I2C command as specified in GC_OR_START bit. On write 1 = Enable programming of GENERAL_CALL or START_BYTE transmission. 0 = Disable programming of GENERAL_CALL or START_BYTE transmission. Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x0

Bit	Mode	Symbol/Description	Reset
10	R/W	GC_OR_START On read If bit 11 (SPECIAL) is set to 1, then this bit indicates whether a General Call or START byte command is to be performed by the controller. 0 = General Call Address. After issuing a General Call, only writes may be performed. Attempting to issue a read command results in setting bit 6 (TX_ABRT) of the IC_RAW_INTR_STAT register. The controller remains in General Call mode until the SPECIAL bit value (bit 11) is cleared. 1 = START BYTE On write 1 = START byte transmission 0 = GENERAL_CALL byte transmission Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x0
9:0	R/W	IC_TAR This is the target address for any master transaction. When transmitting a General Call, these bits are ignored. To generate a START BYTE, the CPU needs to write only once into these bits. Note: If the IC_TAR and IC_SAR are the same, loopback exists but the FIFOs are shared between master and slave, so full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself; it can transmit to a slave only. Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x55

Table 296: I2C_SAR_REG (0x40090008)

Bit	Mode	Symbol/Description	Reset
31:10	-	- Reserved	0x0
9:0	R/W	IC_SAR The IC_SAR holds the slave address when the I2C is operating as a slave. For 7-bit addressing, only IC_SAR[6:0] is used. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE register being set to 0. Write at other times have no effect. Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x55

Table 297: I2C_HS_MADDR_REG (0x4009000C)

Bit	Mode	Symbol/Description	Reset
2:0	R/W	I2C_IC_HS_MAR This bit field holds the value of the I2C HS mode master code. HS-mode master codes are reserved 8-bit codes (00001xxx) that are not used for slave addressing or other purposes. Each master has its unique master code; up to eight high-speed mode masters can be present on the same I2C bus system. Valid values are from 0 to 7. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect.	0x1

Table 298: I2C_DATA_CMD_REG (0x40090010)

Bit	Mode	Symbol/Description	Reset
30:11	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
10	W	I2C_RESTART This bit controls whether a RESTART is issued before the byte is sent or received. 1 = If IC_RESTART_EN is 1, a RESTART is issued before the data is sent/received (according to the value of CMD), regardless of whether or not the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead. 0 = If IC_RESTART_EN is 1, a RESTART is issued only if the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead.	0x0
9	W	I2C_STOP This bit controls whether a STOP is issued after the byte is sent or received. 1 = STOP is issued after this byte, regardless of whether or not the TX FIFO is empty. If the TX FIFO is not empty, the master immediately tries to start a new transfer by issuing a START and arbitrating for the bus. 0 = STOP is not issued after this byte, regardless of whether or not the TX FIFO is empty. If the TX FIFO is not empty, the master continues the current transfer by sending/receiving data bytes according to the value of the CMD bit. If the TX FIFO is empty, the master holds the SCL line low and stalls the bus until a new command is available in the TX FIFO.	0x0
8	W	I2C_CMD This bit controls whether a read or a write is performed. This bit does not control the direction when the I2C Ctrl acts as a slave. It controls only the direction when it acts as a master. 1 = Read 0 = Write When a command is entered in the TX FIFO, this bit distinguishes the write and read commands. In slave-receiver mode, this bit is "don't care" because writes to this register are not required. In slave-transmitter mode, 0 indicates that CPU data is to be transmitted and as DAT or IC_DATA_CMD[7:0]. When programming this bit, you should remember the following: attempting to perform a read operation after a General Call command has been sent results in TX_ABRT interrupt (bit 6 of the I2C_RAW_INTR_STAT_REG), unless bit 11 (SPECIAL) in the I2C_TAR register has been cleared. If 1 is written to this bit after receiving RD_REQ interrupt, then TX_ABRT interrupt occurs. Note: It is possible that while attempting a master I2C read transfer on the controller, RD_REQ interrupt may have occurred simultaneously due to a remote I2C master addressing the controller. In this type of scenario, it ignores the I2C_DATA_CMD write, generates the TX_ABRT interrupt, and waits to service the RD_REQ interrupt.	0x0
7:0	R/W	I2C_DAT This register contains the data to be transmitted or received on the I2C bus. If you are writing to this register and want to perform a read, bits 7:0 (DAT) are ignored by the controller. However, when you read this register, these bits return the value of data received on the controller's interface.	0x0

Table 299: I2C_SS_SCL_HCNT_REG (0x40090014)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_SS_SCL_HCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for standard	0x91

Bit	Mode	Symbol/Description	Reset
		speed. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. Note: This register must not be programmed to a value higher than 65525, because the controller uses a 16-bit counter to flag an I2C bus idle condition when this counter reaches a value of IC_SS_SCL_HCNT + 10.	

Table 300: I2C_SS_SCL_LCNT_REG (0x40090018)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_SS_SCL_LCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for standard speed. This register can be written only when the I2C interface is disabled which corresponds to the I2C_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted, results in 8 being set.	0xAB

Table 301: I2C_FS_SCL_HCNT_REG (0x4009001C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_FS_SCL_HCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. This register can be written only when the I2C interface is disabled, which corresponds to the I2C_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.	0x1A

Table 302: I2C_FS_SCL_LCNT_REG (0x40090020)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_FS_SCL_LCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. This register can be written only when the I2C interface is disabled, which corresponds to the I2C_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set. For designs with APB_DATA_WIDTH = 8, the order of programming is important to ensure the correct operation of the controller. The lower byte must be programmed first. Then the upper byte is programmed.	0x32

Table 303: I2C_HS_SCL_HCNT_REG (0x40090024)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_HS_SCL_HCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high period count for high speed. See "IC_CLK Frequency Configuration". The SCL High time depends on the loading of the bus. For 100 pF loading, the SCL High time is 60 ns; for 400 pF loading, the SCL High time is 120 ns. This register goes away and becomes read-only returning 0 s if IC_MAX_SPEED_MODE != high. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. For designs with APB_DATA_WIDTH = 8, the order of programming is important to ensure the correct operation of the DW_apb_i2c. The lower byte must be programmed first. Then the upper byte is programmed.	0x6

Table 304: I2C_HS_SCL_LCNT_REG (0x40090028)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_HS_SCL_LCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for high speed. For more information, see "IC_CLK Frequency Configuration". The SCL low time depends on the loading of the bus. For 100 pF loading, the SCL low time is 160 ns; for 400 pF loading, the SCL low time is 320 ns. This register goes away and becomes read-only returning 0 s if IC_MAX_SPEED_MODE != high. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set. For designs with APB_DATA_WIDTH == 8, the order of programming is important to ensure the correct operation of the DW_apb_i2c. The lower byte must be programmed first. Then the upper byte is programmed. If the value is less than 8 then the count value gets changed to 8.	0x10

Table 305: I2C_INTR_STAT_REG (0x4009002C)

Bit	Mode	Symbol/Description	Reset
31:15	-	- Reserved	0x0
14	R	R_SCL_STUCK_AT_LOW 1 = R_SCL_STUCK_AT_LOW interrupt is active. 0 = R_SCL_STUCK_AT_LOW interrupt is inactive.	0x0
13	R	R_MASTER_ON_HOLD Indicate whether master is holding the bus and TX FIFO is empty. Enabled only when I2C_DYNAMIC_TAR_UPDATE = 1 and IC_EMPTYFIFO_HOLD_MASTER_EN = 1.	0x0
12	R	R_RESTART_DET	0x0

Bit	Mode	Symbol/Description	Reset
		Indicate whether a RESTART condition has occurred on the I2C interface when DW_apb_i2c is operating in Slave mode and the slave is being addressed. Enabled only when IC_SLV_RESTART_DET_EN = 1. Note: However, in high-speed mode or during a START BYTE transfer, the RESTART comes before the address field as per the I2C protocol. In this case, the slave is not the addressed slave when the RESTART is issued, therefore DW_apb_i2c does not generate the RESTART_DET interrupt.	
11	R	R_GEN_CALL Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling controller or when the CPU reads bit 0 of the I2C_CLR_GEN_CALL register. The controller stores the received data in the RX buffer.	0x0
10	R	R_START_DET Indicate whether a START or RESTART condition occurred on the I2C interface regardless of whether controller is operating in Slave or Master mode.	0x0
9	R	R_STOP_DET Indicate whether a STOP condition occurred on the I2C interface regardless of whether controller is operating in Slave or Master mode.	0x0
8	R	R_ACTIVITY This bit captures I2C Ctrl activity and stays set until it is cleared. There are four ways to clear it: - Disabling the I2C Ctrl - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset When this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller module is idle, this bit remains set until cleared, indicating that there was activity on the bus.	0x0
7	R	R_RX_DONE When the controller is acting as a slave-transmitter, this bit is set to 1, if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	0x0
6	R	R_TX_ABRT This bit indicates if the controller, as an I2C transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I2C master or an I2C slave, and is referred to as a "transmit abort". When this bit is set to 1, the I2C_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. Note: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register I2C_CLR_TX_ABRT is read. When this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	0x0
5	R	R_RD_REQ This bit is set to 1 when the controller is acting as a slave and another I2C master is attempting to read data from the controller. The controller holds the I2C bus in a WAIT state (SCL = 0) until this interrupt is serviced, which means that the slave is addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the I2C_DATA_CMD register. This bit is set to 0 just after the processor reads the I2C_CLR_RD_REQ register.	0x0
4	R	R_TX_EMPTY	0x0

Bit	Mode	Symbol/Description	Reset
		This bit is set to 1 when the transmit buffer is at or below the threshold value set in the I2C_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the Master or Slave state machines. When there is no activity, then with ic_en = 0, this bit is set to 0.	
3	R	R_TX_OVER Set during transmit if the transmit buffer is filled to 32 and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the Master or Slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
2	R	R_RX_FULL Set when the receive buffer reaches or goes above the RX_TL threshold in the I2C_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (I2C_ENABLE[0] = 0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once, the I2C_ENABLE bit 0 is programmed with a 0 regardless of the activity that continues.	0x0
1	R	R_RX_OVER Set if the receive buffer is completely filled to 32 and an additional byte is received from an external I2C device. The controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (I2C_ENABLE[0] = 0), this bit keeps its level until the Master or Slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
0	R	R_RX_UNDER Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (I2C_ENABLE[0] = 0), this bit keeps its level until the Master or Slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0

Table 306: I2C_INTR_MASK_REG (0x40090030)

Bit	Mode	Symbol/Description	Reset
31:15	-	- Reserved	0x0
14	R	M_SCL_STUCK_AT_LOW M_SCL_STUCK_AT_LOW Register field Reserved bits.	0x0
13	R/W	M_MASTER_ON_HOLD These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
12	R/W	M_RESTART_DET These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
11	R/W	M_GEN_CALL These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
10	R/W	M_START_DET These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
9	R/W	M_STOP_DET	0x0

Bit	Mode	Symbol/Description	Reset
		These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	
8	R/W	M_ACTIVITY These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
7	R/W	M_RX_DONE These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
6	R/W	M_TX_ABRT These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
5	R/W	M_RD_REQ These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
4	R/W	M_TX_EMPTY These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
3	R/W	M_TX_OVER These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
2	R/W	M_RX_FULL These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
1	R/W	M_RX_OVER These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
0	R/W	M_RX_UNDER These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1

Table 307: I2C_RAW_INTR_STAT_REG (0x40090034)

Bit	Mode	Symbol/Description	Reset
31:15	-	- Reserved	0x0
14	R	SCL_STUCK_AT_LOW CL_STUCK_AT_LOW Register field Reserved bits.	0x0
13	R	MASTER_ON_HOLD Indicate whether master is holding the bus and TX FIFO is empty. Enabled only when I2C_DYNAMIC_TAR_UPDATE = 1 and IC_EMPTYFIFO_HOLD_MASTER_EN = 1.	0x0
12	R	RESTART_DET Indicate whether a RESTART condition occurred on the I2C interface when DW_apb_i2c is operating in Slave mode and the slave is being addressed. Enabled only when IC_SLV_RESTART_DET_EN = 1. Note: However, in high-speed mode or during a START BYTE transfer, the RESTART comes before the address field as per the I2C protocol. In this case, the	0x0

Bit	Mode	Symbol/Description	Reset
		slave is not the addressed slave when the RESTART is issued, therefore DW_apb_i2c does not generate the RESTART_DET interrupt.	
11	R	GEN_CALL Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling controller or when the CPU reads bit 0 of the I2C_CLR_GEN_CALL register. I2C Ctrl stores the received data in the RX buffer.	0x0
10	R	START_DET Indicate whether a START or RESTART condition occurred on the I2C interface regardless of whether controller is operating in Slave or Master mode.	0x0
9	R	STOP_DET Indicate whether a STOP condition occurred on the I2C interface regardless of whether controller is operating in Save or Mster mode.	0x0
8	R	ACTIVITY This bit captures I2C Ctrl activity and stays set until it is cleared. There are four ways to clear it: - Disabling the I2C Ctrl - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset When this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller module is idle, this bit remains set until cleared indicating that there was activity on the bus.	0x0
7	R	RX_DONE When the controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	0x0
6	R	TX_ABRT This bit indicates if the controller, as an I2C transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I2C master or an I2C slave and is referred to as a "transmit abort". When this bit is set to 1, the I2C_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes place. Note: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register I2C_CLR_TX_ABRT is read. When this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	0x0
5	R	RD_REQ This bit is set to 1 when I2C Ctrl is acting as a slave and another I2C master is attempting to read data from the controller. The controller holds the I2C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the I2C_DATA_CMD register. This bit is set to 0 just after the processor reads the I2C_CLR_RD_REQ register.	0x0
4	R	TX_EMPTY This bit is set to 1 when the transmit buffer is at or below the threshold value set in the I2C_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so	0x0

Bit	Mode	Symbol/Description	Reset
		this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with ic_en=0, this bit is set to 0.	
3	R	TX_OVER Set during transmit if the transmit buffer is filled to 32 and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared .	0x0
2	R	RX_FULL Set when the receive buffer reaches or goes above the RX_TL threshold in the I2C_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (I2C_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the I2C_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.	0x0
1	R	RX_OVER Set if the receive buffer is completely filled to 32 and an additional byte is received from an external I2C device. The controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
0	R	RX_UNDER Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0

Table 308: I2C_RX_TL_REG (0x40090038)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	RX_TL Receive FIFO Threshold Level Controls the level of entries (or above) that triggers the RX_FULL interrupt (bit 2 in I2C_RAW_INTR_STAT register). The valid range is 0-31, with the additional restriction that hardware does not allow this value to be set to a value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 1 entry, and a value of 31 sets the threshold for 32 entries.	0x0

Table 309: I2C_TX_TL_REG (0x4009003C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TX_TL Transmit FIFO Threshold Level Controls the level of entries (or below) that trigger the TX_EMPTY interrupt (bit 4 in I2C_RAW_INTR_STAT register). The valid range is 0-31, with the additional restriction that it may not be set to value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 0 entries, and a value of 31 sets the threshold for 32 entries.	0x0

Table 310: I2C_CLR_INTR_REG (0x40090040)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_INTR	0x0
		Read this register to clear the combined interrupt, all individual interrupts, and the I2C_TX_ABRT_SOURCE register. This bit does not clear hardware clearable interrupts but software clearable interrupts. See Bit 9 of the I2C_TX_ABRT_SOURCE register for an exception to clearing I2C_TX_ABRT_SOURCE.	

Table 311: I2C_CLR_RX_UNDER_REG (0x40090044)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RX_UNDER	0x0
		Read this register to clear the RX_UNDER interrupt (bit 0) of the I2C_RAW_INTR_STAT register.	

Table 312: I2C_CLR_RX_OVER_REG (0x40090048)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RX_OVER	0x0
		Read this register to clear the RX_OVER interrupt (bit 1) of the I2C_RAW_INTR_STAT register.	

Table 313: I2C_CLR_TX_OVER_REG (0x4009004C)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_TX_OVER	0x0
		Read this register to clear the TX_OVER interrupt (bit 3) of the I2C_RAW_INTR_STAT register.	

Table 314: I2C_CLR_RD_REQ_REG (0x40090050)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RD_REQ	0x0
		Read this register to clear the RD_REQ interrupt (bit 5) of the I2C_RAW_INTR_STAT register.	

Table 315: I2C_CLR_TX_ABRT_REG (0x40090054)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_TX_ABRT	0x0
		Read this register to clear the TX_ABRT interrupt (bit 6) of the IC_RAW_INTR_STAT register, and the I2C_TX_ABRT_SOURCE register. This also releases the TX FIFO from the flushed/reset state, allowing more writes to the TX FIFO. See Bit 9 of the I2C_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE.	

Table 316: I2C_CLR_RX_DONE_REG (0x40090058)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RX_DONE	0x0
		Read this register to clear the RX_DONE interrupt (bit 7) of the I2C_RAW_INTR_STAT register.	

Table 317: I2C_CLR_ACTIVITY_REG (0x4009005C)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_ACTIVITY	0x0
		Reading this register clears the ACTIVITY interrupt if the I2C is not active anymore. If the I2C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the IC_RAW_INTR_STAT register.	

Table 318: I2C_CLR_STOP_DET_REG (0x40090060)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_STOP_DET	0x0
		Read this register to clear the STOP_DET interrupt (bit 9) of the IC_RAW_INTR_STAT register.	

Table 319: I2C_CLR_START_DET_REG (0x40090064)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	

Bit	Mode	Symbol/Description	Reset
0	R	CLR_START_DET Read this register to clear the START_DET interrupt (bit 10) of the IC_RAW_INTR_STAT register.	0x0

Table 320: I2C_CLR_GEN_CALL_REG (0x40090068)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	CLR_GEN_CALL Read this register to clear the GEN_CALL interrupt (bit 11) of I2C_RAW_INTR_STAT register.	0x0

Table 321: I2C_ENABLE_REG (0x4009006C)

Bit	Mode	Symbol/Description	Reset
31:3	-	- Reserved	0x0
2	R/W	I2C_TX_CMD_BLOCK In Master mode: 1 = Block the transmission of data on I2C bus even if TX FIFO has data to transmit. 0 = The transmission of data starts on I2C bus automatically as soon as the first data is available in the TX FIFO.	0x0
1	R/W	I2C_ABORT The software can abort the I2C transfer in Uaster mode by setting this bit. The software can set this bit only when ENABLE is already set; otherwise, the controller ignores any write to ABORT bit. The software cannot clear the ABORT bit once set. In response to an ABORT, the controller issues a STOP and flushes the TX FIFO after completing the current transfer, then sets the TX_ABORT interrupt after the abort operation. The ABORT bit is cleared automatically after the abort operation.	0x0
0	R/W	I2C_EN Control whether the controller is enabled. 0 = Disable the controller (TX and RX FIFOs are held in an erased state) 1 = Enable the controller Software can disable the controller while it is active. However, it is important that care be taken to ensure that the controller is disabled properly. When the controller is disabled, the following occurs: The TX FIFO and RX FIFO get flushed. Status bits in the IC_INTR_STAT register are still active until the controller goes into IDLE state. If the module is transmitting, it stops as well as deletes the contents of the transmit buffer after the current transfer is complete. If the module is receiving, the controller stops the current transfer at the end of the current byte and does not acknowledge the transfer. There is a two ic_clk delay when enabling or disabling the controller.	0x0

Table 322: I2C_STATUS_REG (0x40090070)

Bit	Mode	Symbol/Description	Reset
31:11	-	-	0x0
		Reserved	
10	R	LV_HOLD_RX_FIFO_FULL	0x0
		This bit indicates the BUS Hold in Slave mode due to RX FIFO is Full and an additional byte has been received. 1 = Slave holds the bus due to RX FIFO is full. 0 = Slave is not holding the bus or Bus hold is not due to RX FIFO is full.	
9	R	SLV_HOLD_TX_FIFO_EMPTY	0x0
		This bit indicates the BUS Hold in Slave mode for the Read request when the TX FIFO is empty. The Bus is in hold until the TX FIFO has data to Transmit for the read request. 1 = Slave holds the bus due to TX FIFO is empty. 0 = Slave is not holding the bus or Bus hold is not due to TX FIFO is empty.	
8	R	MST_HOLD_RX_FIFO_FULL	0x0
		This bit indicates the BUS Hold in Master mode due to RX FIFO is Full and additional byte is received. 1 = Master holds the bus due to RX FIFO is full. 0 = Master is not holding the bus or Bus hold is not due to RX FIFO is full.	
7	R	MST_HOLD_TX_FIFO_EMPTY	0x0
		The DW_apb_i2c master stalls the write transfer when TX FIFO is empty, and the the last byte does not have the Stop bit set. This bit indicates the BUS hold when the master holds the bus because of the TX FIFO being empty, and the the previous transferred command does not have the Stop bit set. 1 = Master holds the bus due to TX FIFO is empty. 0 = Master is not holding the bus or Bus hold is not due to TX FIFO is empty.	
6	R	SLV_ACTIVITY	0x0
		Slave FSM Activity Status. When the Slave Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0 = Slave FSM is in IDLE state so the Slave part of the controller is not Active. 1 = Slave FSM is not in IDLE state so the Slave part of the controller is Active.	
5	R	MST_ACTIVITY	0x0
		Master FSM Activity Status. When the Master Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0 = Master FSM is in IDLE state so the Master part of the controller is not Active. 1 = Master FSM is not in IDLE state so the Master part of the controller is Active.	
4	R	RFF	0x0
		Receive FIFO Completely Full. When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0 = Receive FIFO is not full. 1 = Receive FIFO is full.	
3	R	RFNE	0x0
		Receive FIFO Not Empty. This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0 = Receive FIFO is empty. 1 = Receive FIFO is not empty.	

Bit	Mode	Symbol/Description	Reset
2	R	TFE Transmit FIFO Completely Empty. When the transmit FIFO is completely empty, this bit is set. When it contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0 = Transmit FIFO is not empty. 1 = Transmit FIFO is empty.	0x1
1	R	TFNF Transmit FIFO Not Full. Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0 = Transmit FIFO is full. 1 = Transmit FIFO is not full.	0x1
0	R	I2C_ACTIVITY I2C Activity Status.	0x0

Table 323: I2C_TXFLR_REG (0x40090074)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R	TXFLR Transmit FIFO Level. Contain the number of valid data entries in the transmit FIFO. Size is constrained by the TXFLR value.	0x0

Table 324: I2C_RXFLR_REG (0x40090078)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R	RXFLR Receive FIFO Level. Contain the number of valid data entries in the receive FIFO. Size is constrained by the RXFLR value.	0x0

Table 325: I2C_SDA_HOLD_REG (0x4009007C)

Bit	Mode	Symbol/Description	Reset
23:16	R/W	I2C_SDA_RX_HOLD Set the required SDA hold time in units of ic_clk period when receiving.	0x0
15:0	R/W	I2C_SDA_TX_HOLD Set the required SDA hold time in units of ic_clk period when transmitting.	0x1

Table 326: I2C_TX_ABRT_SOURCE_REG (0x40090080)

Bit	Mode	Symbol/Description	Reset
16	R	ABRT_USER_ABRT Master-Transmitter: This is a master-mode-only bit. Master detected the transfer abort (IC_ENABLE[1]).	0x0
15	R	ABRT_SLVRD_INTX	0x0

Bit	Mode	Symbol/Description	Reset
		Slave-Transmitter: When the processor side responds to a Slave mode request for data to be transmitted to a remote master and a user writes a 1 in CMD (bit 8) of 2IC_DATA_CMD register. 1 = Slave trying to transmit to remote master in Read mode. 0 = Slave trying to transmit to remote master in Read mode - scenario not present.	
14	R	ABRT_SLV_ARBLOST Slave-Transmitter: Slave lost the bus while transmitting data to a remote master. I2C_TX_ABRT_SOURCE[12] is set at the same time. Note: Even though the slave never "owns" the bus, something could go wrong on the bus. This is a fail safe check. For instance, during a data transmission at the low-to-high transition of SCL, if what is on the data bus is not what is supposed to be transmitted, then the controller no longer own the bus. 1 = Slave lost arbitration to remote master. 0 = Slave lost arbitration to remote master - scenario not present.	0x0
13	R	ABRT_SLVFLUSH_TXFIFO Slave-Transmitter: Slave received a read command and some data exists in the TX FIFO so the slave issues a TX_ABRT interrupt to flush old data in TX FIFO. 1 = Slave flushes existing data in TX-FIFO upon getting read command. 0 = Slave flushes existing data in TX-FIFO upon getting read command - scenario not present.	0x0
12	R	ARB_LOST Master-Transmitter or Slave-Transmitter: Master lost arbitration, or if I2C_TX_ABRT_SOURCE[14] is also set, then the slave transmitter has lost arbitration. Note: I2C can be both master and slave at the same time. 1 = Master or Slave-Transmitter lost arbitration. 0 = Master or Slave-Transmitter lost arbitration - scenario not present.	0x0
11	R	ABRT_MASTER_DIS Master-Transmitter or Master-Receiver: User tries to initiate a Master operation with the Master mode disabled. 1 = User intitating master operation when MASTER disable. 0 = User initiating master operation when MASTER disabled - scenario not present.	0x0
10	R	ABRT_10B_RD_NORSTRT Master-Receiver: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the master sends a read command in 10-bit Addressing mode. 1 = Master trying to read in 10-bit Addressing mode when RESTART disabled. 0 = Master not trying to read in 10-bit Addressing mode when RESTART disabled.	0x0
9	R	ABRT_SBYTE_NORSTRT Master: To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; restart must be enabled (I2C_CON[5] = 1), the SPECIAL bit must be cleared (I2C_TAR[11]), or the GC_OR_START bit must be cleared (I2C_TAR[10]). After the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, bit 9 clears for one cycle and then gets re-asserted. 1: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the user is trying to send a START Byte. 1 = User is trying to send START byte when RESTART disabled. 0 = User is trying to send START byte when RESTART disabled - scenario not present.	0x0

Bit	Mode	Symbol/Description	Reset
8	R	ABRT_HS_NORSTRT Master-Transmitter or Master-Receiver: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the user is trying to use the master to transfer data in High Speed mode. 1 = User trying to switch Master to HS mode when RESTART disabled. 0 = User trying to switch Master to HS mode when RESTART disabled - scenario not present.	0x0
7	R	ABRT_SBYTE_ACKDET Master: Master sent a START Byte and the START Byte was acknowledged (wrong behavior). 1 = ACK detected for START byte. 0 = ACK detected for START byte - scenario not present.	0x0
6	R	ABRT_HS_ACKDET Master: Master is in High Speed mode and the High Speed Master code was acknowledged (wrong behavior). 1 = HS Master code ACKed in HS Mode. 0 = HS Master code ACKed in HS Mode - scenario not present.	0x0
5	R	ABRT_GCALL_READ Master-Transmitter: The controller in Master mode sent a General Call but the user programmed the byte following the General Call to be a read from the bus (IC_DATA_CMD[9] is set to 1). 1 = GCALL is followed by read from bus. 0 = GCALL is followed by read from bus - scenario not present.	0x0
4	R	ABRT_GCALL_NOACK Master-Transmitter: The controller in master mode sent a General Call and no slave on the bus acknowledged the General Call. 1 = GCALL not ACKed by any slave. 0 = GCALL not ACKed by any slave - scenario not present.	0x0
3	R	ABRT_TXDATA_NOACK Master-Transmitter: This is a Master-mode only bit. Master received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledge from the remote slave(s). 1 = Transmitted data not ACKed by addressed slave. 0 = Transmitted data non-ACKed by addressed slave - scenario not present.	0x0
2	R	ABRT_10ADDR2_NOACK Master-Transmitter or Master-Receiver: Master is in 10-bit address mode and the second address byte of the 10-bit address was not acknowledged by any slave. 1 = Byte 2 of 10-bit address not ACKed by any slave. 0 = This abort is not generated.	0x0
1	R	ABRT_10ADDR1_NOACK Master-Transmitter or Master-Receiver: Master is in 10-bit Address mode and the first 10-bit address byte was not acknowledged by any slave. 1 = Byte 1 of 10-bit address not ACKed by any slave. 0 = This abort is not generated.	0x0
0	R	ABRT_7B_ADDR_NOACK Master-Transmitter or Master-Receiver: Master is in 7-bit Addressing mode and the address sent was not acknowledged by any slave. 1 = This abort is generated because of NOACK for 7-bit address.	0x0

Bit	Mode	Symbol/Description	Reset
		0 = This abort is not generated.	

Table 327: I2C_DMA_CR_REG (0x40090088)

Bit	Mode	Symbol/Description	Reset
1	R/W	TDMAE Transmit DMA Enable. This bit enables/disables the transmit FIFO DMA channel. 0 = Transmit DMA disabled. 1 = Transmit DMA enabled.	0x0
0	R/W	RDMAE Receive DMA Enable. This bit enables/disables the receive FIFO DMA channel. 0 = Receive DMA disabled. 1 = Receive DMA enabled.	0x0

Table 328: I2C_DMA_TDLR_REG (0x4009008C)

Bit	Mode	Symbol/Description	Reset
4:0	R/W	DMATDL Transmit Data Level. This bit field controls the level at which a DMA request is made by the transmit logic. It is equal to the watermark level; that is, the dma_tx_req signal is generated when the number of valid data entries in the transmit FIFO is equal to or below this field value, and TDMAE = 1.	0x0

Table 329: I2C_DMA_RDLR_REG (0x40090090)

Bit	Mode	Symbol/Description	Reset
4:0	R/W	DMARDL Receive Data Level. This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or more than this field value + 1, and RDMAE =1. For instance, when DMARDL is 0, then dma_rx_req is asserted when 1 or more data entries are present in the receive FIFO.	0x0

Table 330: I2C_SDA_SETUP_REG (0x40090094)

Bit	Mode	Symbol/Description	Reset
15:8	-	- Reserved	0x0
7:0	R/W	SDA_SETUP SDA Setup. This register controls the amount of time delay (number of I2C clock periods) between the rising edge of SCL and SDA changing by holding SCL low when I2C block services a read request while operating as a slave-transmitter. The relevant I2C requirement is tSU:DAT (Note 4) as detailed in the I2C Bus Specification. This register must be programmed with a value equal to or greater than 2. It is recommended that if the required delay is 1000 ns, then for an I2C frequency of 10 MHz, IC_SDA_SETUP should be programmed to a value of 11. Write to this register succeed only when IC_ENABLE[0] = 0.	0x64

Table 331: I2C_ACK_GENERAL_CALL_REG (0x40090098)

Bit	Mode	Symbol/Description	Reset
15:1	-	-	0x0
		Reserved	
0	R/W	ACK_GEN_CALL	0x0
		ACK General Call. When set to 1, I2C Ctrl responds with a ACK (by asserting ic_data_oe) when it receives a General Call. When set to 0, the controller does not generate General Call interrupts. 1 = Generate ACK for a General Call. 0 = Generate NACK for General Call.	

Table 332: I2C_ENABLE_STATUS_REG (0x4009009C)

Bit	Mode	Symbol/Description	Reset
15:3	-	-	0x0
		Reserved	
2	R	SLV_RX_DATA_LOST	0x0
		Slave Received Data Lost. This bit indicates if a Slave-Receiver operation is aborted with at least one data byte received from an I2C transfer due to the setting of IC_ENABLE from 1 to 0. When read as 1, the controller is deemed to have actively engaged in an aborted I2C transfer (with matching address) and the data phase of the I2C transfer is entered, even though a data byte is responded with a NACK. NOTE: If the remote I2C master terminates the transfer with a STOP condition before the controller has a chance to NACK a transfer, and IC_ENABLE is set to 0, then this bit is also set to 1. When read as 0, the controller is deemed to have disabled without being actively involved in the data phase of a Slave-Receiver transfer. Note: The CPU can safely read this bit when IC_EN (bit 0) is read as 0. 1 = Slave RX Data is lost. 0 = Slave RX Data is not lost.	
1	R	SLV_DISABLED_WHILE_BUSY	0x0
		Slave Disabled While Busy (Transmit, Receive). This bit indicates if a potential or active Slave operation has aborted due to the setting of the IC_ENABLE register from 1 to 0. This bit is set when the CPU writes a 0 to the IC_ENABLE register while: (a) I2C Ctrl is receiving the address byte of the Slave-Transmitter operation from a remote master; OR, (b) address and data bytes of the Slave-Receiver operation from a remote master. When read as 1, the controller is deemed to have forced a NACK during any part of an I2C transfer, irrespective of whether the I2C address matches the slave address set in I2C Ctrl (IC_SAR register) OR if the transfer is completed before IC_ENABLE is set to 0, but has not taken effect. Note: If the remote I2C master terminates the transfer with a STOP condition before the the controller has a chance to NACK a transfer, and IC_ENABLE is set to 0, then this bit is also set to 1. When read as 0, the controller is deemed to be disabled when there is master activity, or when the I2C bus is idle. Note: The CPU can safely read this bit when IC_EN (bit 0) is read as 0. 1 = Slave is disabled when it is active. 0 = Slave is disabled when it is idle.	
0	R	IC_EN	0x0

Bit	Mode	Symbol/Description	Reset
		ic_en Status. This bit always reflects the value driven on the output port ic_en. When read as 1, the controller is deemed to be in an enabled state. When read as 0, the controller is deemed completely inactive. Note: The CPU can safely read this bit anytime. When this bit is read as 0, the CPU can safely read SLV_RX_DATA_LOST (bit 2) and SLV_DISABLED_WHILE_BUSY (bit 1). 1 = I2C enabled. 0 = I2C disabled.	

Table 333: I2C_IC_FS_SPKLEN_REG (0x400900A0)

Bit	Mode	Symbol/Description	Reset
15:8	-	- Reserved	0x0
7:0	R/W	I2C_FS_SPKLEN This register must be set before any I2C bus transaction can take place to ensure stable operation. This register sets the duration, measured in ic_clk cycles, of the longest spike in the SCL or SDA lines that are filtered out by the spike suppression logic. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 1; hardware prevents values less than this being written, and if attempted results in 1 being set.	0x1

Table 334: I2C_IC_HS_SPKLEN_REG (0x400900A4)

Bit	Mode	Symbol/Description	Reset
7:0	R/W	I2C_HS_SPKLEN This register must be set before any I2C bus transaction can take place to ensure stable operation. This register sets the duration, measured in ic_clk cycles, of the longest spike in the SCL or SDA lines that are filtered out by the spike suppression logic. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect. The minimum valid value is 1; hardware prevents values less than this being written, and if attempted results in 1 being set.	0x1

Table 335: Register map I2C2

Address	Register	Description
0x40090100	I2C2_CON_REG	I2C Control Register
0x40090104	I2C2_TAR_REG	I2C Target Address Register
0x40090108	I2C2_SAR_REG	I2C Slave Address Register
0x4009010c	I2C2_HS_MADDR_REG	I2C High Speed Master Mode Code Address Register
0x40090110	I2C2_DATA_CMD_REG	I2C RX/TX Data Buffer and Command Register
0x40090114	I2C2_SS_SCL_HCNT_REG	Standard Speed I2C Clock SCL High Count Register
0x40090118	I2C2_SS_SCL_LCNT_REG	Standard Speed I2C Clock SCL Low Count Register
0x4009011c	I2C2_FS_SCL_HCNT_REG	Fast Speed I2C Clock SCL High Count Register

Address	Register	Description
0x40090120	I2C2_FS_SCL_LCNT_REG	Fast Speed I2C Clock SCL Low Count Register
0x40090124	I2C2_HS_SCL_HCNT_REG	High Speed I2C Clock SCL High Count Register
0x40090128	I2C2_HS_SCL_LCNT_REG	High Speed I2C Clock SCL Low Count Register
0x4009012c	I2C2_INTR_STAT_REG	I2C Interrupt Status Register
0x40090130	I2C2_INTR_MASK_REG	I2C Interrupt Mask Register
0x40090134	I2C2_RAW_INTR_STAT_REG	I2C Raw Interrupt Status Register
0x40090138	I2C2_RX_TL_REG	I2C Receive FIFO Threshold Register
0x4009013c	I2C2_TX_TL_REG	I2C Transmit FIFO Threshold Register
0x40090140	I2C2_CLR_INTR_REG	Clear Combined and Individual Interrupt Register
0x40090144	I2C2_CLR_RX_UNDER_REG	Clear RX_UNDER Interrupt Register
0x40090148	I2C2_CLR_RX_OVER_REG	Clear RX_OVER Interrupt Register
0x4009014c	I2C2_CLR_TX_OVER_REG	Clear TX_OVER Interrupt Register
0x40090150	I2C2_CLR_RD_REQ_REG	Clear RD_REQ Interrupt Register
0x40090154	I2C2_CLR_TX_ABRT_REG	Clear TX_ABRT Interrupt Register
0x40090158	I2C2_CLR_RX_DONE_REG	Clear RX_DONE Interrupt Register
0x4009015c	I2C2_CLR_ACTIVITY_REG	Clear ACTIVITY Interrupt Register
0x40090160	I2C2_CLR_STOP_DET_REG	Clear STOP_DET Interrupt Register
0x40090164	I2C2_CLR_START_DET_REG	Clear START_DET Interrupt Register
0x40090168	I2C2_CLR_GEN_CALL_REG	Clear GEN_CALL Interrupt Register
0x4009016c	I2C2_ENABLE_REG	I2C Enable Register
0x40090170	I2C2_STATUS_REG	I2C Status Register
0x40090174	I2C2_TXFLR_REG	I2C Transmit FIFO Level Register
0x40090178	I2C2_RXFLR_REG	I2C Receive FIFO Level Register
0x4009017c	I2C2_SDA_HOLD_REG	I2C SDA Hold Time Length Register
0x40090180	I2C2_TX_ABRT_SOURCE_REG	I2C Transmit Abort Source Register
0x40090188	I2C2_DMA_CR_REG	DMA Control Register
0x4009018c	I2C2_DMA_TDLR_REG	DMA Transmit Data Level Register
0x40090190	I2C2_DMA_RDLR_REG	I2C Receive Data Level Register
0x40090194	I2C2_SDA_SETUP_REG	I2C SDA Setup Register
0x40090198	I2C2_ACK_GENERAL_CALL_REG	I2C ACK General Call Register

Address	Register	Description
0x4009019c	I2C2_ENABLE_STATUS_REG	I2C Enable Status Register
0x400901a0	I2C2_IC_FS_SPKLEN_REG	I2C SS and FS spike suppression limit Size
0x400901a4	I2C2_IC_HS_SPKLEN_REG	I2C HS spike suppression limit Size

Table 336: I2C2_CON_REG (0x40090100)

Bit	Mode	Symbol/Description	Reset
31:11	-	-	0x0
		Reserved	
10	R	I2C_STOP_DET_IF_MASTER_ACTIVE	0x0
		In Master mode: 1 = Issue the STOP_DET interrupt only when master is active. 0 = Issue the STOP_DET irrespective of whether master is active or not.	
9	R/W	I2C_RX_FIFO_FULL_HLD_CTRL	0x0
		This bit controls whether DW_apb_i2c should hold the bus when the RX FIFO is physically full to its RX_BUFFER_DEPTH. 1 = Hold bus when RX_FIFO is full. 0 = Overflow when RX_FIFO is full.	
8	R/W	I2C_TX_EMPTY_CTRL	0x0
		This bit controls the generation of the TX_EMPTY interrupt as described in the IC_RAW_INTR_STAT register. 1 = Controlled generation of TX_EMPTY interrupt. 0 = Default behaviour of TX_EMPTY interrupt.	
7	R/W	I2C_STOP_DET_IFADDRESSED	0x0
		1 = Slave issues STOP_DET intr only if addressed. 0 = Slave issues STOP_DET intr always. During a general call address, this slave does not issue the STOP_DET interrupt if STOP_DET_IF_ADDRESSED = 1'b1, even if the slave responds to the general call address by generating ACK. The STOP_DET interrupt is generated only when the transmitted address matches the slave address (SAR).	
6	R/W	I2C_SLAVE_DISABLE	0x1
		Slave enabled or disabled after reset is applied, which means software does not have to configure the slave. 0 = Slave is enabled. 1 = Slave is disabled. Software should ensure that if this bit is written with 0, then bit 0 should also be written with 0.	
5	R/W	I2C_RESTART_EN	0x1
		Determine whether RESTART conditions may be sent when acting as a master. 0 = Disable 1 = Enable	
4	R/W	I2C_10BITADDR_MASTER	0x1
		Control whether the controller starts its transfers in 7- or 10-bit addressing mode when acting as a master.	

Bit	Mode	Symbol/Description	Reset
		0 = 7-bit addressing 1 = 10-bit addressing	
3	R/W	I2C_10BITADDR_SLAVE When acting as a slave, this bit controls whether the controller responds to 7- or 10-bit addresses. 0 = 7-bit addressing 1 = 10-bit addressing	0x1
2:1	R/W	I2C_SPEED These bits control at which speed the controller operates. 1 = Standard mode (100 kbit/s) 2 = Fast mode (400 kbit/s) 3 = High-speed mode	0x3
0	R/W	I2C_MASTER_MODE This bit controls whether the controller master is enabled. 0 = Master disabled 1 = Master enabled Software should ensure that if this bit is written with 1 then bit 6 should also be written with 1.	0x1

Table 337: I2C2_TAR_REG (0x40090104)

Bit	Mode	Symbol/Description	Reset
31:12	-	- Reserved	0x0
11	R/W	SPECIAL On read This bit indicates whether software performs a General Call or START BYTE command. 0 = Ignore bit 10 GC_OR_START and use IC_TAR normally. 1 = Perform special I2C command as specified in GC_OR_START bit. On write 1 = Enable programming of GENERAL_CALL or START_BYTE transmission. 0 = Disable programming of GENERAL_CALL or START_BYTE transmission. Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x0
10	R/W	GC_OR_START On read If bit 11 (SPECIAL) is set to 1, then this bit indicates whether a General Call or START byte command is to be performed by the controller. 0 = General Call Address. After issuing a General Call, only writes may be performed. Attempting to issue a read command results in setting bit 6 (TX_ABRT) of the IC_RAW_INTR_STAT register. The controller remains in General Call mode until the SPECIAL bit value (bit 11) is cleared. 1 = START BYTE On write 1 = START byte transmission 0 = GENERAL_CALL byte transmission Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x0

Bit	Mode	Symbol/Description	Reset
9:0	R/W	IC_TAR This is the target address for any master transaction. When transmitting a General Call, these bits are ignored. To generate a START BYTE, the CPU needs to write only once into these bits. Note: If the IC_TAR and IC_SAR are the same, loopback exists but the FIFOs are shared between master and slave, so full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself; it can transmit to a slave only. Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x55

Table 338: I2C2_SAR_REG (0x40090108)

Bit	Mode	Symbol/Description	Reset
31:10	-	- Reserved	0x0
9:0	R/W	IC_SAR The IC_SAR holds the slave address when the I2C is operating as a slave. For 7-bit addressing, only IC_SAR[6:0] is used. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE register being set to 0. Write at other times have no effect. Write to this register succeed only when IC_ENABLE[0] is set to 0.	0x55

Table 339: I2C2_HS_MADDR_REG (0x4009010C)

Bit	Mode	Symbol/Description	Reset
2:0	R/W	I2C_IC_HS_MAR This bit field holds the value of the I2C HS mode master code. HS-mode master codes are reserved 8-bit codes (00001xxx) that are not used for slave addressing or other purposes. Each master has its unique master code; up to eight high-speed mode masters can be present on the same I2C bus system. Valid values are from 0 to 7. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect.	0x1

Table 340: I2C2_DATA_CMD_REG (0x40090110)

Bit	Mode	Symbol/Description	Reset
30:11	-	- Reserved	0x0
10	W	I2C_RESTART This bit controls whether a RESTART is issued before the byte is sent or received. 1 = If IC_RESTART_EN is 1, a RESTART is issued before the data is sent/received (according to the value of CMD), regardless of whether or not the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead. 0 = If IC_RESTART_EN is 1, a RESTART is issued only if the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead.	0x0
9	W	I2C_STOP This bit controls whether a STOP is issued after the byte is sent or received.	0x0

Bit	Mode	Symbol/Description	Reset
		1 = STOP is issued after this byte, regardless of whether or not the TX FIFO is empty. If the TX FIFO is not empty, the master immediately tries to start a new transfer by issuing a START and arbitrating for the bus. 0 = STOP is not issued after this byte, regardless of whether or not the TX FIFO is empty. If the TX FIFO is not empty, the master continues the current transfer by sending/receiving data bytes according to the value of the CMD bit. If the TX FIFO is empty, the master holds the SCL line low and stalls the bus until a new command is available in the TX FIFO.	
8	W	I2C_CMD This bit controls whether a read or a write is performed. This bit does not control the direction when the I2C Ctrl acts as a slave. It controls only the direction when it acts as a master. 1 = Read 0 = Write When a command is entered in the TX FIFO, this bit distinguishes the write and read commands. In slave-receiver mode, this bit is "don't care" because writes to this register are not required. In slave-transmitter mode, 0 indicates that CPU data is to be transmitted and as DAT or IC_DATA_CMD[7:0]. When programming this bit, you should remember the following: attempting to perform a read operation after a General Call command has been sent results in TX_ABRT interrupt (bit 6 of the I2C_RAW_INTR_STAT_REG), unless bit 11 (SPECIAL) in the I2C_TAR register has been cleared. If 1 is written to this bit after receiving RD_REQ interrupt, then TX_ABRT interrupt occurs. Note: It is possible that while attempting a master I2C read transfer on the controller, RD_REQ interrupt may have occurred simultaneously due to a remote I2C master addressing the controller. In this type of scenario, it ignores the I2C_DATA_CMD write, generates the TX_ABRT interrupt, and waits to service the RD_REQ interrupt.	0x0
7:0	R/W	I2C_DAT This register contains the data to be transmitted or received on the I2C bus. If you are writing to this register and want to perform a read, bits 7:0 (DAT) are ignored by the controller. However, when you read this register, these bits return the value of data received on the controller's interface.	0x0

Table 341: I2C2_SS_SCL_HCNT_REG (0x40090114)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_SS_SCL_HCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for standard speed. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. Note: This register must not be programmed to a value higher than 65525, because the controller uses a 16-bit counter to flag an I2C bus idle condition when this counter reaches a value of IC_SS_SCL_HCNT + 10.	0x91

Table 342: I2C2_SS_SCL_LCNT_REG (0x40090118)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_SS_SCL_LCNT	0xAB

Bit	Mode	Symbol/Description	Reset
		This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for standard speed. This register can be written only when the I2C interface is disabled which corresponds to the I2C_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted, results in 8 being set.	

Table 343: I2C2_FS_SCL_HCNT_REG (0x4009011C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_FS_SCL_HCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. This register can be written only when the I2C interface is disabled, which corresponds to the I2C_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.	0x1A

Table 344: I2C2_FS_SCL_LCNT_REG (0x40090120)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_FS_SCL_LCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. This register can be written only when the I2C interface is disabled, which corresponds to the I2C_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set. For designs with APB_DATA_WIDTH = 8, the order of programming is important to ensure the correct operation of the controller. The lower byte must be programmed first. Then the upper byte is programmed.	0x32

Table 345: I2C2_HS_SCL_HCNT_REG (0x40090124)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_HS_SCL_HCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high period count for high speed. See "IC_CLK Frequency Configuration". The SCL High time depends on the loading of the bus. For 100 pF loading, the SCL High time is 60 ns; for 400 pF loading, the SCL High time is 120 ns. This register goes away and becomes read-only returning 0 s if IC_MAX_SPEED_MODE != high. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. For designs with	0x6

Bit	Mode	Symbol/Description	Reset
		APB_DATA_WIDTH = 8, the order of programming is important to ensure the correct operation of the DW_apb_i2c. The lower byte must be programmed first. Then the upper byte is programmed.	

Table 346: I2C2_HS_SCL_LCNT_REG (0x40090128)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	IC_HS_SCL_LCNT This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for high speed. For more information, see "IC_CLK Frequency Configuration". The SCL low time depends on the loading of the bus. For 100 pF loading, the SCL low time is 160 ns; for 400 pF loading, the SCL low time is 320 ns. This register goes away and becomes read-only returning 0 s if IC_MAX_SPEED_MODE != high. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set. For designs with APB_DATA_WIDTH == 8, the order of programming is important to ensure the correct operation of the DW_apb_i2c. The lower byte must be programmed first. Then the upper byte is programmed. If the value is less than 8 then the count value gets changed to 8.	0x10

Table 347: I2C2_INTR_STAT_REG (0x4009012C)

Bit	Mode	Symbol/Description	Reset
31:15	-	- Reserved	0x0
14	R	R_SCL_STUCK_AT_LOW 1 = R_SCL_STUCK_AT_LOW interrupt is active. 0 = R_SCL_STUCK_AT_LOW interrupt is inactive.	0x0
13	R	R_MASTER_ON_HOLD Indicate whether master is holding the bus and TX FIFO is empty. Enabled only when I2C_DYNAMIC_TAR_UPDATE = 1 and IC_EMPTYFIFO_HOLD_MASTER_EN = 1.	0x0
12	R	R_RESTART_DET Indicate whether a RESTART condition has occurred on the I2C interface when DW_apb_i2c is operating in Slave mode and the slave is being addressed. Enabled only when IC_SLV_RESTART_DET_EN = 1. Note: However, in high-speed mode or during a START BYTE transfer, the RESTART comes before the address field as per the I2C protocol. In this case, the slave is not the addressed slave when the RESTART is issued, therefore DW_apb_i2c does not generate the RESTART_DET interrupt.	0x0
11	R	R_GEN_CALL Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling controller or when the CPU reads bit 0 of the I2C_CLR_GEN_CALL register. The controller stores the received data in the RX buffer.	0x0
10	R	R_START_DET	0x0

Bit	Mode	Symbol/Description	Reset
		Indicate whether a START or RESTART condition occurred on the I2C interface regardless of whether controller is operating in Slave or Master mode.	
9	R	R_STOP_DET Indicate whether a STOP condition occurred on the I2C interface regardless of whether controller is operating in Slave or Master mode.	0x0
8	R	R_ACTIVITY This bit captures I2C Ctrl activity and stays set until it is cleared. There are four ways to clear it: - Disabling the I2C Ctrl - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset When this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller module is idle, this bit remains set until cleared, indicating that there was activity on the bus.	0x0
7	R	R_RX_DONE When the controller is acting as a slave-transmitter, this bit is set to 1, if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	0x0
6	R	R_TX_ABRT This bit indicates if the controller, as an I2C transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I2C master or an I2C slave, and is referred to as a "transmit abort". When this bit is set to 1, the I2C_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. Note: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register I2C_CLR_TX_ABRT is read. When this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	0x0
5	R	R_RD_REQ This bit is set to 1 when the controller is acting as a slave and another I2C master is attempting to read data from the controller. The controller holds the I2C bus in a WAIT state (SCL = 0) until this interrupt is serviced, which means that the slave is addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the I2C_DATA_CMD register. This bit is set to 0 just after the processor reads the I2C_CLR_RD_REQ register.	0x0
4	R	R_TX_EMPTY This bit is set to 1 when the transmit buffer is at or below the threshold value set in the I2C_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the Master or Slave state machines. When there is no activity, then with ic_en = 0, this bit is set to 0.	0x0
3	R	R_TX_OVER Set during transmit if the transmit buffer is filled to 32 and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the Master or Slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
2	R	R_RX_FULL	0x0

Bit	Mode	Symbol/Description	Reset
		Set when the receive buffer reaches or goes above the RX_TL threshold in the I2C_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (I2C_ENABLE[0] = 0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once, the I2C_ENABLE bit 0 is programmed with a 0 regardless of the activity that continues.	
1	R	R_RX_OVER Set if the receive buffer is completely filled to 32 and an additional byte is received from an external I2C device. The controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (I2C_ENABLE[0] = 0), this bit keeps its level until the Master or Slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
0	R	R_RX_UNDER Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (I2C_ENABLE[0] = 0), this bit keeps its level until the Master or Slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0

Table 348: I2C2_INTR_MASK_REG (0x40090130)

Bit	Mode	Symbol/Description	Reset
31:15	-	- Reserved	0x0
14	R	M_SCL_STUCK_AT_LOW M_SCL_STUCK_AT_LOW Register field Reserved bits.	0x0
13	R/W	M_MASTER_ON_HOLD These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
12	R/W	M_RESTART_DET These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
11	R/W	M_GEN_CALL These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
10	R/W	M_START_DET These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
9	R/W	M_STOP_DET These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
8	R/W	M_ACTIVITY These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
7	R/W	M_RX_DONE These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
6	R/W	M_TX_ABRT	0x1

Bit	Mode	Symbol/Description	Reset
		These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	
5	R/W	M_RD_REQ These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
4	R/W	M_TX_EMPTY These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
3	R/W	M_TX_OVER These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
2	R/W	M_RX_FULL These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
1	R/W	M_RX_OVER These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
0	R/W	M_RX_UNDER These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1

Table 349: I2C2_RAW_INTR_STAT_REG (0x40090134)

Bit	Mode	Symbol/Description	Reset
31:15	-	- Reserved	0x0
14	R	SCL_STUCK_AT_LOW CL_STUCK_AT_LOW Register field Reserved bits.	0x0
13	R	MASTER_ON_HOLD Indicate whether master is holding the bus and TX FIFO is empty. Enabled only when I2C_DYNAMIC_TAR_UPDATE = 1 and IC_EMPTYFIFO_HOLD_MASTER_EN = 1.	0x0
12	R	RESTART_DET Indicate whether a RESTART condition occurred on the I2C interface when DW_apb_i2c is operating in Slave mode and the slave is being addressed. Enabled only when IC_SLV_RESTART_DET_EN = 1. Note: However, in high-speed mode or during a START BYTE transfer, the RESTART comes before the address field as per the I2C protocol. In this case, the slave is not the addressed slave when the RESTART is issued, therefore DW_apb_i2c does not generate the RESTART_DET interrupt.	0x0
11	R	GEN_CALL Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling controller or when the CPU reads bit 0 of the I2C_CLR_GEN_CALL register. I2C Ctrl stores the received data in the RX buffer.	0x0
10	R	START_DET	0x0

Bit	Mode	Symbol/Description	Reset
		Indicate whether a START or RESTART condition occurred on the I2C interface regardless of whether controller is operating in Slave or Master mode.	
9	R	STOP_DET Indicate whether a STOP condition occurred on the I2C interface regardless of whether controller is operating in Save or Mster mode.	0x0
8	R	ACTIVITY This bit captures I2C Ctrl activity and stays set until it is cleared. There are four ways to clear it: - Disabling the I2C Ctrl - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset When this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller module is idle, this bit remains set until cleared indicating that there was activity on the bus.	0x0
7	R	RX_DONE When the controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	0x0
6	R	TX_ABRT This bit indicates if the controller, as an I2C transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I2C master or an I2C slave and is referred to as a "transmit abort". When this bit is set to 1, the I2C_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes place. Note: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register I2C_CLR_TX_ABRT is read. When this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	0x0
5	R	RD_REQ This bit is set to 1 when I2C Ctrl is acting as a slave and another I2C master is attempting to read data from the controller. The controller holds the I2C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the I2C_DATA_CMD register. This bit is set to 0 just after the processor reads the I2C_CLR_RD_REQ register.	0x0
4	R	TX_EMPTY This bit is set to 1 when the transmit buffer is at or below the threshold value set in the I2C_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with ic_en=0, this bit is set to 0.	0x0
3	R	TX_OVER Set during transmit if the transmit buffer is filled to 32 and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared .	0x0
2	R	RX_FULL	0x0

Bit	Mode	Symbol/Description	Reset
		Set when the receive buffer reaches or goes above the RX_TL threshold in the I2C_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (I2C_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the I2C_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.	
1	R	RX_OVER Set if the receive buffer is completely filled to 32 and an additional byte is received from an external I2C device. The controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
0	R	RX_UNDER Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0

Table 350: I2C2_RX_TL_REG (0x40090138)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	RX_TL Receive FIFO Threshold Level Controls the level of entries (or above) that triggers the RX_FULL interrupt (bit 2 in I2C_RAW_INTR_STAT register). The valid range is 0-31, with the additional restriction that hardware does not allow this value to be set to a value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 1 entry, and a value of 31 sets the threshold for 32 entries.	0x0

Table 351: I2C2_TX_TL_REG (0x4009013C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TX_TL Transmit FIFO Threshold Level Controls the level of entries (or below) that trigger the TX_EMPTY interrupt (bit 4 in I2C_RAW_INTR_STAT register). The valid range is 0-31, with the additional restriction that it may not be set to value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 0 entries, and a value of 31 sets the threshold for 32 entries.	0x0

Table 352: I2C2_CLR_INTR_REG (0x40090140)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	CLR_INTR	0x0

Bit	Mode	Symbol/Description	Reset
		Read this register to clear the combined interrupt, all individual interrupts, and the I2C_TX_ABRT_SOURCE register. This bit does not clear hardware clearable interrupts but software clearable interrupts. See Bit 9 of the I2C_TX_ABRT_SOURCE register for an exception to clearing I2C_TX_ABRT_SOURCE.	

Table 353: I2C2_CLR_RX_UNDER_REG (0x40090144)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RX_UNDER	0x0
		Read this register to clear the RX_UNDER interrupt (bit 0) of the I2C_RAW_INTR_STAT register.	

Table 354: I2C2_CLR_RX_OVER_REG (0x40090148)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RX_OVER	0x0
		Read this register to clear the RX_OVER interrupt (bit 1) of the I2C_RAW_INTR_STAT register.	

Table 355: I2C2_CLR_TX_OVER_REG (0x4009014C)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_TX_OVER	0x0
		Read this register to clear the TX_OVER interrupt (bit 3) of the I2C_RAW_INTR_STAT register.	

Table 356: I2C2_CLR_RD_REQ_REG (0x40090150)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_RD_REQ	0x0
		Read this register to clear the RD_REQ interrupt (bit 5) of the I2C_RAW_INTR_STAT register.	

Table 357: I2C2_CLR_TX_ABRT_REG (0x40090154)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	

Bit	Mode	Symbol/Description	Reset
0	R	CLR_TX_ABRT Read this register to clear the TX_ABRT interrupt (bit 6) of the IC_RAW_INTR_STAT register, and the I2C_TX_ABRT_SOURCE register. This also releases the TX FIFO from the flushed/reset state, allowing more writes to the TX FIFO. See Bit 9 of the I2C_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE.	0x0

Table 358: I2C2_CLR_RX_DONE_REG (0x40090158)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	CLR_RX_DONE Read this register to clear the RX_DONE interrupt (bit 7) of the I2C_RAW_INTR_STAT register.	0x0

Table 359: I2C2_CLR_ACTIVITY_REG (0x4009015C)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	CLR_ACTIVITY Reading this register clears the ACTIVITY interrupt if the I2C is not active anymore. If the I2C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the IC_RAW_INTR_STAT register.	0x0

Table 360: I2C2_CLR_STOP_DET_REG (0x40090160)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	CLR_STOP_DET Read this register to clear the STOP_DET interrupt (bit 9) of the IC_RAW_INTR_STAT register.	0x0

Table 361: I2C2_CLR_START_DET_REG (0x40090164)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	CLR_START_DET Read this register to clear the START_DET interrupt (bit 10) of the IC_RAW_INTR_STAT register.	0x0

Table 362: I2C2_CLR_GEN_CALL_REG (0x40090168)

Bit	Mode	Symbol/Description	Reset
31:1	-	-	0x0
		Reserved	
0	R	CLR_GEN_CALL	0x0
		Read this register to clear the GEN_CALL interrupt (bit 11) of I2C_RAW_INTR_STAT register.	

Table 363: I2C2_ENABLE_REG (0x4009016C)

Bit	Mode	Symbol/Description	Reset
31:3	-	-	0x0
		Reserved	
2	R/W	I2C_TX_CMD_BLOCK	0x0
		In Master mode: 1 = Block the transmission of data on I2C bus even if TX FIFO has data to transmit. 0 = The transmission of data starts on I2C bus automatically as soon as the first data is available in the TX FIFO.	
1	R/W	I2C_ABORT	0x0
		The software can abort the I2C transfer in Uaster mode by setting this bit. The software can set this bit only when ENABLE is already set; otherwise, the controller ignores any write to ABORT bit. The software cannot clear the ABORT bit once set. In response to an ABORT, the controller issues a STOP and flushes the TX FIFO after completing the current transfer, then sets the TX_ABORT interrupt after the abort operation. The ABORT bit is cleared automatically after the abort operation.	
0	R/W	I2C_EN	0x0
		Control whether the controller is enabled. 0 = Disable the controller (TX and RX FIFOs are held in an erased state) 1 = Enable the controller Software can disable the controller while it is active. However, it is important that care be taken to ensure that the controller is disabled properly. When the controller is disabled, the following occurs: The TX FIFO and RX FIFO get flushed. Status bits in the IC_INTR_STAT register are still active until the controller goes into IDLE state. If the module is transmitting, it stops as well as deletes the contents of the transmit buffer after the current transfer is complete. If the module is receiving, the controller stops the current transfer at the end of the current byte and does not acknowledge the transfer. There is a two ic_clk delay when enabling or disabling the controller.	

Table 364: I2C2_STATUS_REG (0x40090170)

Bit	Mode	Symbol/Description	Reset
31:11	-	-	0x0
		Reserved	
10	R	LV_HOLD_RX_FIFO_FULL	0x0

Bit	Mode	Symbol/Description	Reset
		This bit indicates the BUS Hold in Slave mode due to RX FIFO is Full and an additional byte has been received. 1 = Slave holds the bus due to RX FIFO is full. 0 = Slave is not holding the bus or Bus hold is not due to RX FIFO is full.	
9	R	SLV_HOLD_TX_FIFO_EMPTY This bit indicates the BUS Hold in Slave mode for the Read request when the TX FIFO is empty. The Bus is in hold until the TX FIFO has data to Transmit for the read request. 1 = Slave holds the bus due to TX FIFO is empty. 0 = Slave is not holding the bus or Bus hold is not due to TX FIFO is empty.	0x0
8	R	MST_HOLD_RX_FIFO_FULL This bit indicates the BUS Hold in Master mode due to RX FIFO is Full and additional byte is received. 1 = Master holds the bus due to RX FIFO is full. 0 = Master is not holding the bus or Bus hold is not due to RX FIFO is full.	0x0
7	R	MST_HOLD_TX_FIFO_EMPTY The DW_apb_i2c master stalls the write transfer when TX FIFO is empty, and the the last byte does not have the Stop bit set. This bit indicates the BUS hold when the master holds the bus because of the TX FIFO being empty, and the the previous transferred command does not have the Stop bit set. 1 = Master holds the bus due to TX FIFO is empty. 0 = Master is not holding the bus or Bus hold is not due to TX FIFO is empty.	0x0
6	R	SLV_ACTIVITY Slave FSM Activity Status. When the Slave Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0 = Slave FSM is in IDLE state so the Slave part of the controller is not Active. 1 = Slave FSM is not in IDLE state so the Slave part of the controller is Active.	0x0
5	R	MST_ACTIVITY Master FSM Activity Status. When the Master Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0 = Master FSM is in IDLE state so the Master part of the controller is not Active. 1 = Master FSM is not in IDLE state so the Master part of the controller is Active.	0x0
4	R	RFF Receive FIFO Completely Full. When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0 = Receive FIFO is not full. 1 = Receive FIFO is full.	0x0
3	R	RFNE Receive FIFO Not Empty. This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0 = Receive FIFO is empty. 1 = Receive FIFO is not empty.	0x0
2	R	TFE Transmit FIFO Completely Empty. When the transmit FIFO is completely empty, this bit is set. When it contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0 = Transmit FIFO is not empty.	0x1

Bit	Mode	Symbol/Description	Reset
		1 = Transmit FIFO is empty.	
1	R	TFNF Transmit FIFO Not Full. Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0 = Transmit FIFO is full. 1 = Transmit FIFO is not full.	0x1
0	R	I2C_ACTIVITY I2C Activity Status.	0x0

Table 365: I2C2_TXFLR_REG (0x40090174)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R	TXFLR Transmit FIFO Level. Contain the number of valid data entries in the transmit FIFO. Size is constrained by the TXFLR value.	0x0

Table 366: I2C2_RXFLR_REG (0x40090178)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R	RXFLR Receive FIFO Level. Contain the number of valid data entries in the receive FIFO. Size is constrained by the RXFLR value.	0x0

Table 367: I2C2_SDA_HOLD_REG (0x4009017C)

Bit	Mode	Symbol/Description	Reset
23:16	R/W	I2C_SDA_RX_HOLD Set the required SDA hold time in units of ic_clk period when receiving.	0x0
15:0	R/W	I2C_SDA_TX_HOLD Set the required SDA hold time in units of ic_clk period when transmitting.	0x1

Table 368: I2C2_TX_ABRT_SOURCE_REG (0x40090180)

Bit	Mode	Symbol/Description	Reset
16	R	ABRT_USER_ABRT Master-Transmitter: This is a master-mode-only bit. Master detected the transfer abort (IC_ENABLE[1]).	0x0
15	R	ABRT_SLVRD_INTX Slave-Transmitter: When the processor side responds to a Slave mode request for data to be transmitted to a remote master and a user writes a 1 in CMD (bit 8) of 2IC_DATA_CMD register. 1 = Slave trying to transmit to remote master in Read mode. 0 = Slave trying to transmit to remote master in Read mode - scenario not present.	0x0

Bit	Mode	Symbol/Description	Reset
14	R	ABRT_SLV_ARBLOST Slave-Transmitter: Slave lost the bus while transmitting data to a remote master. I2C_TX_ABRT_SOURCE[12] is set at the same time. Note: Even though the slave never "owns" the bus, something could go wrong on the bus. This is a fail safe check. For instance, during a data transmission at the low-to-high transition of SCL, if what is on the data bus is not what is supposed to be transmitted, then the controller no longer own the bus. 1 = Slave lost arbitration to remote master. 0 = Slave lost arbitration to remote master - scenario not present.	0x0
13	R	ABRT_SLVFLUSH_TXFIFO Slave-Transmitter: Slave received a read command and some data exists in the TX FIFO so the slave issues a TX_ABRT interrupt to flush old data in TX FIFO. 1 = Slave flushes existing data in TX-FIFO upon getting read command. 0 = Slave flushes existing data in TX-FIFO upon getting read command - scenario not present.	0x0
12	R	ARB_LOST Master-Transmitter or Slave-Transmitter: Master lost arbitration, or if I2C_TX_ABRT_SOURCE[14] is also set, then the slave transmitter has lost arbitration. Note: I2C can be both master and slave at the same time. 1 = Master or Slave-Transmitter lost arbitration. 0 = Master or Slave-Transmitter lost arbitration - scenario not present.	0x0
11	R	ABRT_MASTER_DIS Master-Transmitter or Master-Receiver: User tries to initiate a Master operation with the Master mode disabled. 1 = User intitating master operation when MASTER disable. 0 = User initiating master operation when MASTER disabled - scenario not present.	0x0
10	R	ABRT_10B_RD_NORSTRT Master-Receiver: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the master sends a read command in 10-bit Addressing mode. 1 = Master trying to read in 10-bit Addressing mode when RESTART disabled. 0 = Master not trying to read in 10-bit Addressing mode when RESTART disabled.	0x0
9	R	ABRT_SBYTE_NORSTRT Master: To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; restart must be enabled (I2C_CON[5] = 1), the SPECIAL bit must be cleared (I2C_TAR[11]), or the GC_OR_START bit must be cleared (I2C_TAR[10]). After the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, bit 9 clears for one cycle and then gets re-asserted. 1: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the user is trying to send a START Byte. 1 = User is trying to send START byte when RESTART disabled. 0 = User is trying to send START byte when RESTART disabled - scenario not present.	0x0
8	R	ABRT_HS_NORSTRT Master-Transmitter or Master-Receiver: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the user is trying to use the master to transfer data in High Speed mode. 1 = User trying to switch Master to HS mode when RESTART disabled.	0x0

Bit	Mode	Symbol/Description	Reset
		0 = User trying to switch Master to HS mode when RESTART disabled - scenario not present.	
7	R	ABRT_SBYTE_ACKDET Master: Master sent a START Byte and the START Byte was acknowledged (wrong behavior). 1 = ACK detected for START byte. 0 = ACK detected for START byte - scenario not present.	0x0
6	R	ABRT_HS_ACKDET Master: Master is in High Speed mode and the High Speed Master code was acknowledged (wrong behavior). 1 = HS Master code ACKed in HS Mode. 0 = HS Master code ACKed in HS Mode - scenario not present.	0x0
5	R	ABRT_GCALL_READ Master-Transmitter: The controller in Master mode sent a General Call but the user programmed the byte following the General Call to be a read from the bus (IC_DATA_CMD[9] is set to 1). 1 = GCALL is followed by read from bus. 0 = GCALL is followed by read from bus - scenario not present.	0x0
4	R	ABRT_GCALL_NOACK Master-Transmitter: The controller in master mode sent a General Call and no slave on the bus acknowledged the General Call. 1 = GCALL not ACKed by any slave. 0 = GCALL not ACKed by any slave - scenario not present.	0x0
3	R	ABRT_TXDATA_NOACK Master-Transmitter: This is a Master-mode only bit. Master received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledge from the remote slave(s). 1 = Transmitted data not ACKed by addressed slave. 0 = Transmitted data non-ACKed by addressed slave - scenario not present.	0x0
2	R	ABRT_10ADDR2_NOACK Master-Transmitter or Master-Receiver: Master is in 10-bit address mode and the second address byte of the 10-bit address was not acknowledged by any slave. 1 = Byte 2 of 10-bit address not ACKed by any slave. 0 = This abort is not generated.	0x0
1	R	ABRT_10ADDR1_NOACK Master-Transmitter or Master-Receiver: Master is in 10-bit Address mode and the first 10-bit address byte was not acknowledged by any slave. 1 = Byte 1 of 10-bit address not ACKed by any slave. 0 = This abort is not generated.	0x0
0	R	ABRT_7B_ADDR_NOACK Master-Transmitter or Master-Receiver: Master is in 7-bit Addressing mode and the address sent was not acknowledged by any slave. 1 = This abort is generated because of NOACK for 7-bit address. 0 = This abort is not generated.	0x0

Table 369: I2C2_DMA_CR_REG (0x40090188)

Bit	Mode	Symbol/Description	Reset
1	R/W	TDMAE Transmit DMA Enable. This bit enables/disables the transmit FIFO DMA channel. 0 = Transmit DMA disabled. 1 = Transmit DMA enabled.	0x0
0	R/W	RDMAE Receive DMA Enable. This bit enables/disables the receive FIFO DMA channel. 0 = Receive DMA disabled. 1 = Receive DMA enabled.	0x0

Table 370: I2C2_DMA_TDLR_REG (0x4009018C)

Bit	Mode	Symbol/Description	Reset
4:0	R/W	DMATDL Transmit Data Level. This bit field controls the level at which a DMA request is made by the transmit logic. It is equal to the watermark level; that is, the dma_tx_req signal is generated when the number of valid data entries in the transmit FIFO is equal to or below this field value, and TDMAE = 1.	0x0

Table 371: I2C2_DMA_RDLR_REG (0x40090190)

Bit	Mode	Symbol/Description	Reset
4:0	R/W	DMARDL Receive Data Level. This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or more than this field value + 1, and RDMAE =1. For instance, when DMARDL is 0, then dma_rx_req is asserted when 1 or more data entries are present in the receive FIFO.	0x0

Table 372: I2C2_SDA_SETUP_REG (0x40090194)

Bit	Mode	Symbol/Description	Reset
15:8	-	- Reserved	0x0
7:0	R/W	SDA_SETUP SDA Setup. This register controls the amount of time delay (number of I2C clock periods) between the rising edge of SCL and SDA changing by holding SCL low when I2C block services a read request while operating as a slave-transmitter. The relevant I2C requirement is tSU:DAT (Note 4) as detailed in the I2C Bus Specification. This register must be programmed with a value equal to or greater than 2. It is recommended that if the required delay is 1000 ns, then for an I2C frequency of 10 MHz, IC_SDA_SETUP should be programmed to a value of 11. Write to this register succeed only when IC_ENABLE[0] = 0.	0x64

Table 373: I2C2_ACK_GENERAL_CALL_REG (0x40090198)

Bit	Mode	Symbol/Description	Reset
15:1	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
0	R/W	ACK_GEN_CALL ACK General Call. When set to 1, I2C Ctrl responds with a ACK (by asserting ic_data_oe) when it receives a General Call. When set to 0, the controller does not generate General Call interrupts. 1 = Generate ACK for a General Call. 0 = Generate NACK for General Call.	0x0

Table 374: I2C2_ENABLE_STATUS_REG (0x4009019C)

Bit	Mode	Symbol/Description	Reset
15:3	-	- Reserved	0x0
2	R	SLV_RX_DATA_LOST Slave Received Data Lost. This bit indicates if a Slave-Receiver operation is aborted with at least one data byte received from an I2C transfer due to the setting of IC_ENABLE from 1 to 0. When read as 1, the controller is deemed to have actively engaged in an aborted I2C transfer (with matching address) and the data phase of the I2C transfer is entered, even though a data byte is responded with a NACK. NOTE: If the remote I2C master terminates the transfer with a STOP condition before the controller has a chance to NACK a transfer, and IC_ENABLE is set to 0, then this bit is also set to 1. When read as 0, the controller is deemed to have disabled without being actively involved in the data phase of a Slave-Receiver transfer. Note: The CPU can safely read this bit when IC_EN (bit 0) is read as 0. 1 = Slave RX Data is lost. 0 = Slave RX Data is not lost.	0x0
1	R	SLV_DISABLED_WHILE_BUSY Slave Disabled While Busy (Transmit, Receive). This bit indicates if a potential or active Slave operation has aborted due to the setting of the IC_ENABLE register from 1 to 0. This bit is set when the CPU writes a 0 to the IC_ENABLE register while: (a) I2C Ctrl is receiving the address byte of the Slave-Transmitter operation from a remote master; OR, (b) address and data bytes of the Slave-Receiver operation from a remote master. When read as 1, the controller is deemed to have forced a NACK during any part of an I2C transfer, irrespective of whether the I2C address matches the slave address set in I2C Ctrl (IC_SAR register) OR if the transfer is completed before IC_ENABLE is set to 0, but has not taken effect. Note: If the remote I2C master terminates the transfer with a STOP condition before the the controller has a chance to NACK a transfer, and IC_ENABLE is set to 0, then this bit is also set to 1. When read as 0, the controller is deemed to be disabled when there is master activity, or when the I2C bus is idle. Note: The CPU can safely read this bit when IC_EN (bit 0) is read as 0. 1 = Slave is disabled when it is active. 0 = Slave is disabled when it is idle.	0x0
0	R	IC_EN ic_en Status. This bit always reflects the value driven on the output port ic_en. When read as 1, the controller is deemed to be in an enabled state. When read as 0, the controller is deemed completely inactive.	0x0

Bit	Mode	Symbol/Description	Reset
		Note: The CPU can safely read this bit anytime. When this bit is read as 0, the CPU can safely read SLV_RX_DATA_LOST (bit 2) and SLV_DISABLED_WHILE_BUSY (bit 1). 1 = I2C enabled. 0 = I2C disabled.	

Table 375: I2C2_IC_FS_SPKLEN_REG (0x400901A0)

Bit	Mode	Symbol/Description	Reset
15:8	-	- Reserved	0x0
7:0	R/W	I2C_FS_SPKLEN This register must be set before any I2C bus transaction can take place to ensure stable operation. This register sets the duration, measured in ic_clk cycles, of the longest spike in the SCL or SDA lines that are filtered out by the spike suppression logic. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE register being set to 0. Write at other times have no effect. The minimum valid value is 1; hardware prevents values less than this being written, and if attempted results in 1 being set.	0x1

Table 376: I2C2_IC_HS_SPKLEN_REG (0x400901A4)

Bit	Mode	Symbol/Description	Reset
7:0	R/W	I2C_HS_SPKLEN This register must be set before any I2C bus transaction can take place to ensure stable operation. This register sets the duration, measured in ic_clk cycles, of the longest spike in the SCL or SDA lines that are filtered out by the spike suppression logic. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE[0] register being set to 0. Write at other times have no effect. The minimum valid value is 1; hardware prevents values less than this being written, and if attempted results in 1 being set.	0x1

10.12 KDMA Registers

Table 377: Register map KDMA

Address	Register	Description
0x40070a00	KDMA_ENABLE_REG	Enable kDMA
0x40070a04	KDMA_RESET_REG	Software reset of kDMA
0x40070a08	KDMA_CFG_DESCRIPTOR_ADDR_REG	Base address of first task descriptor memory area
0x40070a0c	KDMA_CHANNEL_ENABLE_REG	Enable the channel of kDMA
0x40070a10	KDMA_IRQ_DONE_TYPE_REG	dma_done Interrupt type
0x40070a14	KDMA_SW_REQUEST_REG	kDMA software request
0x40070a18	KDMA_IRQ_DONE_REG	Indicator of which channel invokes kdma_done
0x40070a1c	KDMA_IRQ_DONE_CLR_REG	Clear KDMA_IRQ_DONE

Address	Register	Description
0x40070a20	KDMA_IRQ_ERR_REG	Indicator of which channel invokes kdma_err
0x40070a24	KDMA_IRQ_ERR_CLR_REG	Clear KDMA_IRQ_ERR
0x40070a28	KDMA_STATUS_REG	Current status of kDMA
0x40070a2c	KDMA_STATUS_DESC_ADDR_REG	Address of current task descriptor
0x40070a30	KDMA_STATUS_COUNTER_REG	Counters of current DMA task
0x40070a34	KDMA_STATUS_DESC_REG	Current task descriptor
0x40070a38	KDMA_STATUS_DESC_ADDR_PRE_REG	Address of previous task descriptor
0x40070a3c	KDMA_AHB_HPROT_3_TO_0_REG	HPROT signal of AHB bus (Channel 0~3)
0x40070a40	KDMA_AHB_HPROT_7_TO_4_REG	HPROT signal of AHB bus (Channel 4~7)
0x40070a44	KDMA_AHB_HPROT_11_TO_8_REG	HPROT signal of AHB bus (Channel 8~11)
0x40070a48	KDMA_LLI_COUNTER_REG	kDMA software request

Table 378: KDMA_ENABLE_REG (0x40070A00)

Bit	Mode	Symbol/Description	Reset
31:1	R	-	0x0
		Reserved	
0	R/W	DMA_ENABLE	0x0
		Write 1: Enable DMA	
		Write 0: Disable DMA	

Table 379: KDMA_RESET_REG (0x40070A04)

Bit	Mode	Symbol/Description	Reset
31:1	R	-	0x0
		Reserved	
0	R/W	DMA_RESET	0x0
		Write 1: Reset DMA, and automatically return to 0	

Table 380: KDMA_CFG_DESCRIPTOR_ADDR_REG (0x40070A08)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	CFG_DESCRIPTOR_ADDR	0x0
		Base address of first task descriptor memory area	

Table 381: KDMA_CHANNEL_ENABLE_REG (0x40070A0C)

Bit	Mode	Symbol/Description	Reset
31:12	R	-	0x0
		Reserved	
11:0	R/W	CHANNEL_ENABLE	0x0
		Enable DMA channel 1: Enable channel 0: Disable channel	

Table 382: KDMA_IRQ_DONE_TYPE_REG (0x40070A10)

Bit	Mode	Symbol/Description	Reset
31:24	R	-	0x0
		Reserved	
23:22	R/W	CHANNEL11_DONE_TYPE	0x0
		dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	
21:20	R/W	CHANNEL10_DONE_TYPE	0x0
		dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	
19:18	R/W	CHANNEL9_DONE_TYPE	0x0
		dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	
17:16	R/W	CHANNEL8_DONE_TYPE	0x0
		dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	
15:14	R/W	CHANNEL7_DONE_TYPE	0x0
		dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done	

Bit	Mode	Symbol/Description	Reset
		2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	
13:12	R/W	CHANNEL6_DONE_TYPE dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	0x0
11:10	R/W	CHANNEL5_DONE_TYPE dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	0x0
9:8	R/W	CHANNEL4_DONE_TYPE dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	0x0
7:6	R/W	CHANNEL3_DONE_TYPE dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	0x0
5:4	R/W	CHANNEL2_DONE_TYPE dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	0x0
3:2	R/W	CHANNEL1_DONE_TYPE dma_done interrupt type 0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	0x0
1:0	R/W	CHANNEL0_DONE_TYPE dma_done interrupt type	0x0

Bit	Mode	Symbol/Description	Reset
		0: dma_done interrupt enable when DMA chain is done 1: dma_done interrupt enable when DMA task is done 2: dma_done interrupt enable when REQ_MODE = 0 and arbitration period is ended 3: Reserved	

Table 383: KDMA_SW_REQUEST_REG (0x40070A14)

Bit	Mode	Symbol/Description	Reset
31:12	R	-	0x0
		Reserved	
11:0	W	SW_REQUEST	0x0
		Write 1: send request signal, and automatically return to 0	

Table 384: KDMA_IRQ_DONE_REG (0x40070A18)

Bit	Mode	Symbol/Description	Reset
31:12	R	-	0x0
		Reserved	
11:0	R	IRQ_DONE	0x0
		Indicator of which channel invokes dma_done	

Table 385: KDMA_IRQ_DONE_CLR_REG (0x40070A1C)

Bit	Mode	Symbol/Description	Reset
31:12	W	-	0x0
		Reserved	
11:0	W	IRQ_DONE_CLR	0x0
		Write 1: clear KDMA_IRQ_DONE, and automatically return to 0	

Table 386: KDMA_IRQ_ERR_REG (0x40070A20)

Bit	Mode	Symbol/Description	Reset
31:12	R	-	0x0
		Reserved	
11:0	R	IRQ_ERR	0x0
		Indicator of which channel invokes dma_err	

Table 387: KDMA_IRQ_ERR_CLR_REG (0x40070A24)

Bit	Mode	Symbol/Description	Reset
31:12	W	-	0x0
		Reserved	
11:0	W	IRQ_ERR_CLR	0x0
		Write 1: clear KDMA_IRQ_ERR, and automatically return to 0	

Table 388: KDMA_STATUS_REG (0x40070A28)

Bit	Mode	Symbol/Description	Reset
31:24	R	-	0x0
		Reserved	
23:20	R	CURRENT_STATE	0x0
		Current state on FSM	
19:16	R	CURRENT_ACTIVE_CH	0x0
		Currently active channel number	
15:12	R	-	0x0
		Reserved	
11:0	R	PENDING_CH	0x0
		Currently pending channel number	

Table 389: KDMA_STATUS_DESC_ADDR_REG (0x40070A2C)

Bit	Mode	Symbol/Description	Reset
31:0	R	ADDRESS	0x0
		Address of current task descriptor	

Table 390: KDMA_STATUS_COUNTER_REG (0x40070A30)

Bit	Mode	Symbol/Description	Reset
31	R	ARB_LAST_FLAG	0x0
		The flag to check whether the arbitration is last period in current task	
30:16	R	ARB_DONE_COUNTER	0x0
		The number of done arbitration period in current task	
15	R	TRANSFER_LAST_FLAG	0x0
		The flag to check whether the transfer is last in current arbitration period	
14:0	R	TRANSFER_DONE_COUNTER	0x0
		The number of done transfer in current arbitration period	

Table 391: KDMA_STATUS_DESC_REG (0x40070A34)

Bit	Mode	Symbol/Description	Reset
31:0	R	TASK_DESCRIPTOR	0x0
		Current task descriptor	

Table 392: KDMA_STATUS_DESC_ADDR_PRE_REG (0x40070A38)

Bit	Mode	Symbol/Description	Reset
31:0	R	ADDRESS	0x0
		Address of previous task descriptor	

Table 393: KDMA_AHB_HPROT_3_TO_0_REG (0x40070A3C)

Bit	Mode	Symbol/Description	Reset
31:28	R/W	CH3_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
27:24	R/W	CH3_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
23:20	R/W	CH2_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
19:16	R/W	CH2_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
15:12	R/W	CH1_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
11:8	R/W	CH1_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
7:4	R/W	CH0_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
3:0	R/W	CH0_DST_HPROT HPROT signal of AHB bus	0x1

Bit	Mode	Symbol/Description	Reset
		[3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	

Table 394: KDMA_AHB_HPROT_7_TO_4_REG (0x40070A40)

Bit	Mode	Symbol/Description	Reset
31:28	R/W	CH7_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
27:24	R/W	CH7_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
23:20	R/W	CH6_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
19:16	R/W	CH6_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
15:12	R/W	CH5_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
11:8	R/W	CH5_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
7:4	R/W	CH4_SRC_HPROT HPROT signal of AHB bus	0x1

Bit	Mode	Symbol/Description	Reset
		[3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	
3:0	R/W	CH4_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1

Table 395: KDMA_AHB_HPROT_11_TO_8_REG (0x40070A44)

Bit	Mode	Symbol/Description	Reset
31:28	R/W	CH11_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
27:24	R/W	CH11_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
23:20	R/W	CH10_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
19:16	R/W	CH10_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
15:12	R/W	CH9_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
11:8	R/W	CH9_DST_HPROT HPROT signal of AHB bus	0x1

Bit	Mode	Symbol/Description	Reset
		[3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	
7:4	R/W	CH8_SRC_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1
3:0	R/W	CH8_DST_HPROT HPROT signal of AHB bus [3]: Modifiable (1: Cacheable, 0: Non-cacheable) [2]: Bufferable (1: Bufferable, 0: Non-bufferable) [1]: Privileged (1: Privileged, 0: User access) [0]: Data/Opcode (1: Data access, 0: Opcode fetch)	0x1

Table 396: KDMA_LLI_COUNTER_REG (0x40070A48)

Bit	Mode	Symbol/Description	Reset
31:20	R	- Reserved	0x0
19:16	R	LAST_DONE_CHANNEL Last transfer done channel number, reset when the next transfer is started	0x0
15:0	R	LLI_COUNTER Total LLI count for the last transfer done, reset when the next transfer is started	0x0

10.13 MEMCTRL Registers

Table 397: Register map MEMCTRL

Address	Register	Description
0x400b0800	MEMCTRL_STALL_REG	Maximum Stall cycles Control Register
0x400b0804	MEMCTRL_STATUS_REG	RAM cells Status Register
0x400b0808	MEMCTRL_PRIO_ARB_REG	Priority Control Register for RAM cells from 0 to 23
0x400b080c	MEMCTRL_STATIC_CLK_OFF_REG	Memory Static Clock Off register
0x400b0810	MEMCTRL_DYNAMIC_CLK_ON_REG	Memory Dynamic Clock On register
0x400b0814	MEMCTRL_MST_CLK_EN_REG	Memory Master Clock En register
0x400b0818	MEMCTRL_SYS_ARB_REG	APB_SYS ICM Priority level
0x400b081c	MEMCTRL_AUD_ARB_REG	APB_AUDIO ICM Priority level

Address	Register	Description
0x400b0820	MEMCTRL_CC312_ARB_REG	APB_CC312 ICM Priority level
0x400b0824	MEMCTRL_CIS_BASE_ADDR_REG	

Table 398: MEMCTRL_STALL_REG (0x400B0800)

Bit	Mode	Symbol/Description	Reset
23:20	R/W	CIS_MAX_STALL Maximum allowed number of stall cycles for the SDIO CIS interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
19:16	R/W	WIFI_HSU_MAX_STALL Maximum allowed number of stall cycles for the Wi-Fi HSU AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
15:12	R/W	WIFI_MAC_MAX_STALL Maximum allowed number of stall cycles for the Wi-Fi MAC AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
11:8	R/W	AHB_DMA_MAX_STALL Maximum allowed number of stall cycles for the DMA AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
7:4	R/W	AHB_CPUS_MAX_STALL Maximum allowed number of stall cycles for the CPUS AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
3:0	R/W	AHB_CPUC_MAX_STALL Maximum allowed number of stall cycles for the CPUC AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle	0xF

Bit	Mode	Symbol/Description	Reset
		15: max 15 stall cycles	

Table 399: MEMCTRL_STATUS_REG (0x400B0804)

Bit	Mode	Symbol/Description	Reset
23	RW1C	RAM23_OFF_BUT_ACCESS Reading 1 indicates RAM23 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
22	RW1C	RAM22_OFF_BUT_ACCESS Reading 1 indicates RAM22 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
21	RW1C	RAM21_OFF_BUT_ACCESS Reading 1 indicates RAM21 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
20	RW1C	RAM20_OFF_BUT_ACCESS Reading 1 indicates RAM20 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
19	RW1C	RAM19_OFF_BUT_ACCESS Reading 1 indicates RAM19 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
18	RW1C	RAM18_OFF_BUT_ACCESS Reading 1 indicates RAM18 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
17	RW1C	RAM17_OFF_BUT_ACCESS Reading 1 indicates RAM17 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
16	RW1C	RAM16_OFF_BUT_ACCESS Reading 1 indicates RAM16 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
15	RW1C	RAM15_OFF_BUT_ACCESS Reading 1 indicates RAM15 was off but still access was performed. Writing 1 will clear the status back to 0.	0x0
14	RW1C	RAM14_OFF_BUT_ACCESS Reading 1 indicates RAM14 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
13	RW1C	RAM13_OFF_BUT_ACCESS Reading 1 indicates RAM13 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
12	RW1C	RAM12_OFF_BUT_ACCESS Reading 1 indicates RAM12 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
11	RW1C	RAM11_OFF_BUT_ACCESS Reading 1 indicates RAM11 was off but still access was performed. Writing 1 clears the status back to 0.	0x0

Bit	Mode	Symbol/Description	Reset
10	RW1C	RAM10_OFF_BUT_ACCESS Reading 1 indicates RAM10 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
9	RW1C	RAM9_OFF_BUT_ACCESS Reading 1 indicates RAM9 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
8	RW1C	RAM8_OFF_BUT_ACCESS Reading 1 indicates RAM8 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
7	RW1C	RAM7_OFF_BUT_ACCESS Reading 1 indicates RAM7 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
6	RW1C	RAM6_OFF_BUT_ACCESS Reading 1 indicates RAM6 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
5	RW1C	RAM5_OFF_BUT_ACCESS Reading 1 indicates RAM5 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
4	RW1C	RAM4_OFF_BUT_ACCESS Reading 1 indicates RAM4 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
3	RW1C	RAM3_OFF_BUT_ACCESS Reading 1 indicates RAM3 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
2	RW1C	RAM2_OFF_BUT_ACCESS Reading 1 indicates RAM2 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
1	RW1C	RAM1_OFF_BUT_ACCESS Reading 1 indicates RAM1 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
0	RW1C	RAM0_OFF_BUT_ACCESS Reading 1 indicates RAM0 was off but still access was performed. Writing 1 clears the status back to 0.	0x0

Table 400: MEMCTRL_PRIO_ARB_REG (0x400B0808)

Bit	Mode	Symbol/Description	Reset
11:10	R/W	PRIO_ARB_CIS_RAM Priority of RAM0~RAM23 for the SDIO CIS interface. 00: low priority 01: mid priority (default) 1x: highest	0x3
9:8	R/W	PRIO_ARB_WIFI_HSU_RAM Priority of RAM0~RAM23 for the Wi-Fi HSU AHB interface.	0x0

Bit	Mode	Symbol/Description	Reset
		00: low priority 01: mid priority (default) 1x: highest	
7:6	R/W	PRIO_ARB_WIFI_MAC_RAM Priority of RAM0~RAM23 for the Wi-Fi MAC AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x0
5:4	R/W	PRIO_ARB_DMA_RAM Priority of RAM0~RAM23 for the DMA AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x0
3:2	R/W	PRIO_ARB_CPUS_RAM Priority of RAM0~RAM23 for the CPUS AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x1
1:0	R/W	PRIO_ARB_CPUC_RAM Priority of RAM0~RAM23 for the CPUC AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x1

Table 401: MEMCTRL_STATIC_CLK_OFF_REG (0x400B080C)

Bit	Mode	Symbol/Description	Reset
23	R/W	SRAM23 Static clock off - SRAM23	0x0
22	R/W	SRAM22 Static clock off - SRAM22	0x0
21	R/W	SRAM21 Static clock off - SRAM21	0x0
20	R/W	SRAM20 Static clock off - SRAM20	0x0
19	R/W	SRAM19 Static clock off - SRAM19	0x0
18	R/W	SRAM18 Static clock off - SRAM18	0x0
17	R/W	SRAM17 Static clock off - SRAM17	0x0
16	R/W	SRAM16 Static clock off - SRAM16	0x0
15	R/W	SRAM15	0x0

Bit	Mode	Symbol/Description	Reset
		Static clock off - SRAM15	
14	R/W	SRAM14	0x0
		Static clock off - SRAM14	
13	R/W	SRAM13	0x0
		Static clock off - SRAM13	
12	R/W	SRAM12	0x0
		Static clock off - SRAM12	
11	R/W	SRAM11	0x0
		Static clock off - SRAM11	
10	R/W	SRAM10	0x0
		Static clock off - SRAM10	
9	R/W	SRAM9	0x0
		Static clock off - SRAM9	
8	R/W	SRAM8	0x0
		Static clock off - SRAM8	
7	R/W	SRAM7	0x0
		Static clock off - SRAM7	
6	R/W	SRAM6	0x0
		Static clock off - SRAM6	
5	R/W	SRAM5	0x0
		Static clock off - SRAM5	
4	R/W	SRAM4	0x0
		Static clock off - SRAM4	
3	R/W	SRAM3	0x0
		Static clock off - SRAM3	
2	R/W	SRAM2	0x0
		Static clock off - SRAM2	
1	R/W	SRAM1	0x0
		Static clock off - SRAM1	
0	R/W	SRAM0	0x0
		Static clock off - SRAM0	

Table 402: MEMCTRL_DYNAMIC_CLK_ON_REG (0x400B0810)

Bit	Mode	Symbol/Description	Reset
5	R/W	MEM_UNIT	0x0
		Dynamic clock on - Memory Units	
4	R/W	-	0x0
		Reserved	
3	R/W	MMI2MEM	0x0
		Dynamic clock on - MMI2MEM	
2	R/W	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
1	R/W	ARBITER Dynamic clock on - Arbiters	0x0
0	R/W	AHB2MEM Dynamic clock on - AHB2MEM	0x0

Table 403: MEMCTRL_MST_CLK_EN_REG (0x400B0814)

Bit	Mode	Symbol/Description	Reset
5	R/W	MST_CIS_CLK_EN Clock Enable for memctrl's master - SDIO CIS If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
4	R/W	MST_WIFI_HSU_CLK_EN Clock Enable for memctrl's master - Wi-Fi HSU If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
3	R/W	MST_WIFI_MAC_CLK_EN Clock Enable for memctrl's master - Wi-Fi MAC If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
2	R/W	MST_DMA_CLK_EN Clock Enable for memctrl's master - DMA If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
1	R/W	MST_CPUS_CLK_EN Clock Enable for memctrl's master - CPUS If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
0	R/W	MST_CPUC_CLK_EN Clock Enable for memctrl's master - CPUC If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1

Table 404: MEMCTRL_SYS_ARB_REG (0x400B0818)

Bit	Mode	Symbol/Description	Reset
1	R/W	APB_AHB_CPUS_PRIO Priority AHB_CPUS layer system bus 0x0: Highest priority 0x1: Second priority	0x1
0	R/W	APB_AHB_DMA_PRIO Priority AHB_DMA layer system bus 0x0: Highest priority 0x1: Second priority	0x0

Table 405: MEMCTRL_AUD_ARB_REG (0x400B081C)

Bit	Mode	Symbol/Description	Reset
1	R/W	APB_AHB_CPUS_PRIO Priority AHB_CPUS layer system bus 0x0: Highest priority 0x1: Second priority	0x1
0	R/W	APB_AHB_DMA_PRIO Priority AHB_DMA layer system bus 0x0: Highest priority 0x1: Second priority	0x0

Table 406: MEMCTRL_CC312_ARB_REG (0x400B0820)

Bit	Mode	Symbol/Description	Reset
1	R/W	APB_AHB_CPUS_PRIO Priority AHB_CPUS layer system bus 0x0: Highest priority 0x1: Second priority	0x1
0	R/W	APB_AHB_DMA_PRIO Priority AHB_DMA layer system bus 0x0: Highest priority 0x1: Second priority	0x0

Table 407: MEMCTRL_CIS_BASE_ADDR_REG (0x400B0824)

Bit	Mode	Symbol/Description	Reset
22:0	R/W	MEM_CIS_BASE_ADDR Base address of SDIO CIS interface. Full_32bit_address = { MEM_CIS_BASE_ADDR[22:0], cis_addr[6:0], 2'b00 }; So Full_32bit_address [31:9] = MEM_CIS_BASE_ADDR[22:0]; ex1) When MEM_CIS_BASE_ADDR[22:0] = 0x10057E (reset value) Then Full_32bit_address base address by MEM_CIS_BASE_ADDR is 0x200A_FC00. Full_32bit_address = 0x200A_FC00 When cis_addr[6:0] = 0x00. Full_32bit_address = 0x200A_FDFC When cis_addr[6:0] = 0x7F. So, Full_32bit_address range is 0x200AFC00 ~ 0x200AFDFF. ex2) When MEM_CIS_BASE_ADDR[22:0] = 0x000001 (reset value) Then Full_32bit_address base address by MEM_CIS_BASE_ADDR is 0x0000_0200. Full_32bit_address = 0x0000_0200 When cis_addr[6:0] = 0x00. Full_32bit_address = 0x0000_03FC When cis_addr[6:0] = 0x7F. So, Full_32bit_address range is 0x00000200 ~ 0x000003FF. And MEM_CIS_BASE_ADDR's unit address is 0x200 = 512 Byte.	0x10057E

10.14 OQSPI Registers

Table 408: Register map OQSPIF

Address	Register	Description
0x29000000	OQSPIF_CTRLBUS_REG	SPI Bus control register for the Manual mode
0x29000004	OQSPIF_CTRLMODE_REG	Mode Control register
0x29000008	OQSPIF_RECVDATA_REG	Received data for the Manual mode
0x2900000c	OQSPIF_BURSTCMDA_REG	The way of reading in Auto mode (command register A)
0x29000010	OQSPIF_BURSTCMDB_REG	The way of reading in Auto mode (command register B)
0x29000014	OQSPIF_STATUS_REG	The status register of the OSPI controller
0x29000018	OQSPIF_WRITEDATA_REG	Write data to SPI Bus for the Manual mode
0x2900001c	OQSPIF_READDATA_REG	Read data from SPI Bus for the Manual mode
0x29000020	OQSPIF_DUMMYDATA_REG	Send dummy clocks to SPI Bus for the Manual mode
0x29000024	OQSPIF_ERASECTRL_REG	OSPI Erase control register
0x29000028	OQSPIF_ERASECMDA_REG	The way of erasing in Auto mode (command register A)
0x2900002c	OQSPIF_ERASECMDB_REG	The way of erasing in Auto mode (command register B)
0x29000030	OQSPIF_ERASECMDC_REG	The way of erasing in Auto mode (command register C)
0x29000034	OQSPIF_BURSTBRK_REG	Read break sequence in Auto mode
0x29000038	OQSPIF_STATUSCMD_REG	The way of reading the status of external device in Auto mode
0x2900003c	OQSPIF_CHCKERASE_REG	Check erase progress in Auto mode
0x29000040	OQSPIF_GP_REG	OSPI General Purpose control register
0x29000100	OQSPIF_CTR_CTRL_REG	Control register for the decryption engine of the OSPIC
0x29000104	OQSPIF_CTR_SADDR_REG	Start address of the encrypted content in the OSPI flash
0x29000108	OQSPIF_CTR_EADDR_REG	End address of the encrypted content in the OSPI flash
0x2900010c	OQSPIF_CTR_NONCE_0_3_REG	Nonce bytes 0 to 3 for the AES-CTR algorithm
0x29000110	OQSPIF_CTR_NONCE_4_7_REG	Nonce bytes 4 to 7 for the AES-CTR algorithm
0x29000114	OQSPIF_CTR_KEY_0_3_REG	Key bytes 0 to 3 for the AES-CTR algorithm
0x29000118	OQSPIF_CTR_KEY_4_7_REG	Key bytes 4 to 7 for the AES-CTR algorithm

Address	Register	Description
0x2900011c	OQSPIF_CTR_KEY_8_11_REG	Key bytes 8 to 11 for the AES-CTR algorithm
0x29000120	OQSPIF_CTR_KEY_12_15_REG	Key bytes 12 to 15 for the AES-CTR algorithm
0x29000124	OQSPIF_CTR_KEY_16_19_REG	Key bytes 16 to 19 for the AES-CTR algorithm
0x29000128	OQSPIF_CTR_KEY_20_23_REG	Key bytes 20 to 23 for the AES-CTR algorithm
0x2900012c	OQSPIF_CTR_KEY_24_27_REG	Key bytes 24 to 27 for the AES-CTR algorithm
0x29000130	OQSPIF_CTR_KEY_28_31_REG	Key bytes 28 to 31 for the AES-CTR algorithm

Table 409: OQSPIF_CTRLBUS_REG (0x29000000)

Bit	Mode	Symbol/Description	Reset
31:6	-	-	0x0
		Reserved	
5	W	OSPIC_DIS_CS	0x0
		Write 1 to disable the chip select (active low), when the controller is in Manual mode.	
4	W	OSPIC_EN_CS	0x0
		Write 1 to enable the chip select (active low), when the controller is in Manual mode.	
3	W	OSPIC_SET_OCTAL	0x0
		Write 1 to set the bus mode in Octal mode, when the controller is in Manual mode.	
2	W	OSPIC_SET_QUAD	0x0
		Write 1 to set the bus mode in Quad mode, when the controller is in Manual mode.	
1	W	OSPIC_SET_DUAL	0x0
		Write 1 to set the bus mode in Dual mode, when the controller is in Manual mode.	
0	W	OSPIC_SET_SINGLE	0x0
		Write 1 to set the bus mode in Single SPI mode, when the controller is in Manual mode.	

Table 410: OQSPIF_CTRLMODE_REG (0x29000004)

Bit	Mode	Symbol/Description	Reset
31:28	R/W	OSPIC_IO_UH_DAT	0x0
		The value of OSPI_IO4-7 pads if OSPI_IO_UH_OEN is 1.	
27	R/W	OSPIC_IO_UH_OEN	0x0
		Forces the output enable for the upper half of the OSPI bus (OSPI_IO4-7). Set this bit to 1 only in SPI, Dual or Quad SPI mode to control the upper half of the OSPI bus. When the Octal SPI is enabled in the flash device, set this bit to zero. 0: The OSPI_IO4-7 pad direction is decided by the controller. 1: The OSPI_IO4-7 pad are outputs. The output values are defined by the corresponding OSPIC_IO_UH_DAT bits.	
26:19	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
18	R/W	OSPIC_INC_LIM_EN This bit has meaning only for the read in Auto mode and only when the read access in the AHB bus is an incremental burst of unspecified length. 0: The length of the burst is considered as unspecified. The access in the flash device is implemented as is defined by the OSPIC BUF_LIM_EN bit. 1: The length of the burst is considered as equal to 8 bytes. The access in the flash device is implemented by the controller as one or more different bursts, until to be served the access in the AHB bus. Each burst in the flash devicehas maximum length of 8 bytes. The setting OSPIC_INC_LIM_EN=1 is useful, if we know that the masters that make use of the incremental burst of unspecified length, require no more than 8 bytes.	0x0
17	R/W	OSPIC_RD_ERR_EN Controls the generation of AHB bus error response when a read is performed in the address space where the flash device is mapped and Auto mode is not enabled. 0: The controller ignores the access. There is no error response due to the read access. 1: The controller responds with an AHB error response.	0x0
16	R/W	OSPIC_MAN_DIRCHG_MD Selection of the direction change method in Manual mode. 0: The bus direction goes to input after each access. 1: The bus direction goes to input only after a dummy access.	0x0
15	R/W	OSPIC_DMY_MD Define the clock cycle where the bus turns in Hi-Z during the transmission of dummy bytes. This is applicable in both Manual and Auto mode. 0: The bus becomes Hi-Z on the last clock. 1: The bus becomes Hi-Z on the last two clocks.	0x0
14	R/W	OSPIC_CMD_X2_EN Define the number of bytes that consist the instruction code in the command sequences that produced by the OSPIC during Auto mode. 0: The instruction code is one byte only. 1: The instruction code is two bytes. The second byte of the instruction code is the inverse of the first byte. The command sequence that is produced by the OSPIC_BURSTBRK_REG is not affected by this setting.	0x0
13	R/W	OSPIC_USE_32BA Controls the length of the address that the external memory device uses. 0: The external memory device uses 24 bits address. 1: The external memory device uses 32 bits address. The controller uses this bit in order to decide the number of the address bytes that has to transfer to the external device during Auto mode.	0x0
12	R/W	OSPIC_BUF_LIM_EN This bit has meaning only for the read in Auto mode. Defines the behavior of the controller when the internal buffer is full and there are more data to be retrieved for the current burst.	0x0

Bit	Mode	Symbol/Description	Reset
		0: The access in the flash device is not terminated when the internal buffer has no empty space. In this case the OSPI_SCK clock is blocked until to free space in the internal buffer. 1: The access in the flash device is terminated when the internal buffer has no empty space. A new access in the flash device is initiated when addresses that are not present in the internal buffer are requested. In both cases the access in the flash device is terminated when there is no any read request.	
11:9	R/W	OSPIC_PCLK_MD Read pipe clock delay relative to the falling edge of OSPI_SCK. See OSPI Timing for timing parameters and recommended 0-7 values.	0x0
8	R/W	OSPIC_RPIPE_EN Control the use of the data read pipe. 0: The read pipe is disabled. The sampling clock is defined according to the OSPIC_RXD_NEG setting. 1: The read pipe is enabled. The delay of the sampling clock is defined according to the OSPICI_PCLK_MD setting (recommended).	0x0
7	R/W	OSPIC_RXD_NEG Define the clock edge that is used for the capturing of the received data, when the read pipe is not active (OSPIC_RPIPE_EN = 0). 0: Sampling of the received data with the positive edge of the OSPI_SCK. 1: Sampling of the received data with the negative edge of the OSPI_SCK. The internal OSPI_SCK clock that is used by the controller for the capturing of the received data has a skew in respect of the OSPI_SCK that is received by the external memory device.To improve the timing requirements of the read path, the controller supports the read pipe register with a programmable clock delay. See also the OSPIC_RPIPE_EN register.	0x0
6	R/W	OSPIC_HRDY_MD This configuration bit is useful when the frequency of the OSPI clock is lower than that of the AMBA bus, to avoid locking the AMBA bus for an extended period. 0: Add wait states through hready signal when an access is performed on the OSPIC_CTRLBUS_REG, OSPIC_WRITEDATA, OSPIC_READDATA and OSPIC_DUMMYDATA registers. It is not needed to check the OSPIC_BUSY of the OSPIC_STATUS_REG. 1: The controller does not add wait states through the hready signal when is performed access on the OSPIC_CTRLBUS_REG, OSPIC_WRITEDATA, OSPIC_READDATA and OSPIC_DUMMYDATA registers. The OSPIC_BUSY bit of the OSPIC_STATUS_REG must be checked to detect the completion of the requested access. It is applicable only when the controller is in Manual mode. In Auto mode, the controller always adds wait states through the hready signal.	0x0
5	R/W	OSPIC_IO3_DAT The value of OSPI_IO3 pad if OSPI_IO3_OEN is 1.	0x0
4	R/W	OSPIC_IO2_DAT The value of OSPI_IO2 pad if OSPI_IO2_OEN is 1.	0x0
3	R/W	OSPIC_IO3_OEN Force the output enable of the OSPI_IO3. Set this bit to 1 only in SPI or Dual SPI mode to control the /HOLD signal. When the Quad or Octal SPI is enabled in the flash device, set this bit to zero. 0: The OSPI_IO3 pad direction is decided by the controller.	0x0

Bit	Mode	Symbol/Description	Reset
		1: The OSPI_IO3 pad is output. The output value is defined by the OSPIC_IO3_DAT.	
2	R/W	OSPIC_IO2_OEN Force the output enable of the OSPI_IO2. Set this bit to 1 only in SPI or Dual SPI mode to control the /WP signal. When the Quad or Octal SPI is enabled in the flash device, set this bit to zero. 0: The OSPI_IO2 pad direction is decided by the controller. 1: The OSPI_IO2 pad is output. The output value is defined by the OSPIC_IO2_DAT.	0x0
1	R/W	OSPIC_CLK_MD Mode of the generated OSPI_SCK clock. 0: Use Mode 0 for the OSPI_CLK. The OSPI_SCK is low when OSPI_CS is high. 1: Use Mode 3 for the OSPI_CLK. The OSPI_SCK is high when OSPI_CS is high.	0x0
0	R/W	OSPIC_AUTO_MD Mode of operation. 0: Manual mode is selected. 1: Auto mode is selected. While erasing, the OSPIC_AUTO_MD goes in Read-only mode (see OSPIC_ERASE_EN).	0x0

Table 411: OQSPIF_RECVDATA_REG (0x29000008)

Bit	Mode	Symbol/Description	Reset
31:0	R	OSPIC_RECVDATA This register contains the received data when the OSPIC_READDATA_REG register is used in Manual mode, to retrieve data from the external memory device and OSPIC_HRDY_MD=1 && OSPIC_BUSY=0.	0x0

Table 412: OQSPIF_BURSTCMDA_REG (0x2900000C)

Bit	Mode	Symbol/Description	Reset
31:30	R/W	OSPIC_DMY_TX_MD It describes the mode of the SPI bus during the Dummy bytes phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Octal	0x0
29:28	R/W	OSPIC_EXT_TX_MD It describes the mode of the SPI bus during the Extra bytes phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Octal	0x0
27:26	R/W	OSPIC_ADR_TX_MD It describes the mode of the SPI bus during the address phase. 0x0: Single SPI 0x1: Dual 0x2: Quad	0x0

Bit	Mode	Symbol/Description	Reset
		0x3: Octal	
25:24	R/W	OSPIC_INST_TX_MD It describes the mode of the SPI bus during the Instruction phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Octal	0x0
23:16	R/W	OSPIC_EXT_BYTE The value of an extra byte which is transferred after address (only if OSPIC_EXT_BYTE_EN= 1). Usually it is the Mode Bits in Dual/Quad/Octal SPI I/O instructions.	0x0
15:8	R/W	OSPIC_INST_WB Instruction value for Wrapping Burst. This value is the selected instruction when OSPIC_WRAP_MD is equal to 1 and the access is a wrapping burst of length and size described by the bit fields OSPIC_WRAP_LEN and OSPIC_WRAP_SIZE respectively.	0x0
7:0	R/W	OSPIC_INST Instruction value for Incremental Burst or Single read access. This value is the selected instruction at the cases of incremental burst or single read access. Also this value is used when a wrapping burst is not supported (OSPIC_WRAP_MD)	0x0

Table 413: OQSPIF_BURSTCMDB_REG (0x29000010)

Bit	Mode	Symbol/Description	Reset
31:19	-	- Reserved	0x0
18:16	R/W	OSPIC_CS_HIGH_MIN Between the transmissions of two different instructions to the flash memory, the SPI bus stays in idle state (OSPI_CS high) for at least this number of OSPI_SCK clock cycles. See the OSPIC_ERS_CS_HI register for some exceptions.	0x0
15:14	R/W	OSPIC_WRAP_SIZE It describes the selected data size of a wrapping burst (OSPIC_WRAP_MD). 0x0: Byte access (8-bit) 0x1: Half word access (16-bit) 0x2: Word access (32-bit) 0x3: Reserved	0x0
13:12	R/W	OSPIC_WRAP_LEN It describes the selected length of a wrapping burst (OSPIC_WRAP_MD). 0x0: 4-beat wrapping burst 0x1: 8-beat wrapping burst 0x2: 16-beat wrapping burst 0x3: Reserved	0x0
11	R/W	OSPIC_WRAP_MD Wrap mode 0: The OSPIC_INST is the selected instruction at any access. 1: The OSPIC_INST_WB is the selected instruction at any wrapping burst access of length and size described by the registers OSPIC_WRAP_LEN and OSPIC_WRAP_SIZE respectively. In all other cases the OSPIC_INST is the	0x0

Bit	Mode	Symbol/Description	Reset
		selected instruction. Use this feature only when the serial FLASH memory supports a special instruction for wrapping burst access.	
10	R/W	OSPIC_INST_MD Instruction mode 0: Transmit instruction at any burst access. 1: Transmit instruction only in the first access after the selection of Auto mode.	0x0
9	R/W	OSPIC_DMY_EN Dummy bytes enable 0: Do not send the dummy bytes 1: Send the dummy bytes. The number of the dummy bytes is defined by the OSPIC_DMY_NUM.	0x0
8:4	R/W	OSPIC_DMY_NUM Number of dummy bytes (minus 1). Can be set 1-32 dummy bytes (o-31 values). The dummy bytes are applied only when OSPIC_DMY_EN = 1.	0x0
3	R/W	OSPIC_EXT_HF_DS Extra half disable output. 0: If OSPIC_EXT_BYTE_EN = 1, then transmit the complete OSPIC_EXT_BYTE. 1: If OSPIC_EXT_BYTE_EN = 1, then the output is disabled (Hi-Z) during the transmission of bits [3:0] of OSPIC_EXT_BYTE. This setting has no meaning if the extra byte is transferred in Octal mode. In this case keep this bit to zero value.	0x0
2	R/W	OSPIC_EXT_BYTE_EN Extra byte enable 0: Do not send the OSPIC_EXT_BYTE 1: Send the OSPIC_EXT_BYTE	0x0
1:0	R/W	OSPIC_DAT_RX_MD It describes the mode of the SPI bus during the data phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Octal	0x0

Table 414: OQSPIF_STATUS_REG (0x29000014)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	OSPIC_BUSY The status of the SPI Bus. 0: The SPI Bus is idle 1: The SPI Bus is active. Read data, write data or dummy data activity is in progress. Has meaning only in Manual mode and only when OSPIC_HRDY_MD = 1.	0x0

Table 415: OQSPIF_WRITEDATA_REG (0x29000018)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_WRITEDATA Writing to this register is generating a data transfer from the controller to the external memory device. The data written in this register, is then transferred to the memory using the selected mode of the SPI bus (Single SPI, Dual SPI, Quad SPI or Octal SPI). The data size of the access to this register can be 32-bit/16-bit/8-bit and is equal to the number of the transferred bits. This register has meaning only when the controller is in Manual mode.	0x0

Table 416: OQSPIF_READDATA_REG (0x2900001C)

Bit	Mode	Symbol/Description	Reset
31:0	R	OSPIC_READDATA A read access at this register generates a data transfer from the external memory device to the OSPIC controller. The data is transferred using the selected mode of the SPI bus (Single SPI, Dual SPI, Quad SPI or Octal SPI). The data size of the access to this register can be 32-bit/16-bit/8-bit and is equal to the number of the transferred bits. This register has meaning only when the controller is in Manual mode.	0x0

Table 417: OQSPIF_DUMMYDATA_REG (0x29000020)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_DUMMYDATA Writing to this register generates a number of clock pulses to the SPI bus. During the last clock of this activity in the SPI bus, the OSPI_IOx data pads are in Hi-Z state (see also the OSPIC_DMY_MD). The data size of the access to this register can be 32-bit/16-bit/8-bit. The number of generated pulses is equal to: (size of AHB bus access)/(size of SPI bus). The size of SPI bus is equal to 1, 2, 4 or 8 for Single, Dual, Quad or Octal SPI mode respectively. This register has meaning only when the controller is in Manual mode.	0x0

Table 418: OQSPIF_ERASECTRL_REG (0x29000024)

Bit	Mode	Symbol/Description	Reset
31:29	-	- Reserved	0x0
28	R/W	OSPIC_ERS_RES_DIS This configuration bit has meaning when an erase is suspended. Normally the erase is resumed when the flash stays idle (without read accesses) for a predefined number of clock cycles (see OSPIC_ERASECMDB_REG [OSPIC_ERSRES_HLD]). By setting this bit the execution of the erase resume process can be postponed. 0: A suspended erase is resumed based on the setting in the OSPIC_ERSRES_HLD. 1: The erase is not resumed even after the expiration of the OSPIC_ERSRES_HLD. The erase can be resumed again only when the OSPIC_ERS_RES_DIS=0.	0x0
27:25	R	OSPIC_ERS_STATE It shows the progress of sector/block erasing (read-only). 0x0: No Erase.	0x0

Bit	Mode	Symbol/Description	Reset
		0x1: Pending erase request 0x2: Erase procedure is running 0x3: Suspended Erase procedure 0x4: Finishing the Erase procedure 0x5..0x7: Reserved	
24	R/W	OSPIC_ERASE_EN During Manual mode (OSPIC_AUTO_MD = 0). This bit is in Read-only mode. During Auto mode (OSPIC_AUTO_MD = 1). To request the erasing of the block/sector (OSPIC_ERS_ADDR, 12'b0) write 1 to this bit. This bit is cleared automatically with the end of the erasing. Until the end of erasing the OSPIC_ERASE_EN remains in Read-only mode. During the same time interval the controller remains in Auto mode (OSPIC_AUTO_MD goes in Read-only mode).	0x0
23:4	R/W	OSPIC_ERS_ADDR Defines the address of the block/sector that is requested to be erased. If OSPIC_USE_32BA = 0 (24 bits addressing), bits OSPIC_ERASECTRL_REG[23-12] determine the block/sector address bits [23-12]. The OSPIC_ERASECTRL_REG[11-4] are ignored by the controller. If OSPIC_USE_32BA = 1 (32 bits addressing) bits OSPIC_ERASECTRL_REG[23-4] determine the block/sectors address bits [31:12]	0x0
3:0	-	- Reserved	0x0

Table 419: OQSPIF_ERASECMDA_REG (0x29000028)

Bit	Mode	Symbol/Description	Reset
31:24	R/W	OSPIC_RES_INST The code value of the erase resume instruction	0x0
23:16	R/W	OSPIC_SUS_INST The code value of the erase suspend instruction.	0x0
15:8	R/W	OSPIC_WEN_INST The code value of the write enable instruction.	0x0
7:0	R/W	OSPIC_ERS_INST The code value of the erase instruction.	0x0

Table 420: OQSPIF_ERASECMDB_REG (0x2900002C)

Bit	Mode	Symbol/Description	Reset
31:24	R/W	OSPIC_RESSUS_DLY Define a timer that counts the minimum allowed delay between an erase suspend command and the previous erase resume command (or the initial erase command). 0: Do not wait. The controller starts immediately to suspend the erase procedure. 1..255: The controller waits for at least this number of 278 kHz clock cycles before the suspension of erasing. Time starts counting after the end of the previous erase resume command (or the initial erase command). 278 kHz is divn clock divided by 144.	0x0
23:20	-	- Reserved	0x0

Bit	Mode	Symbol/Description	Reset
19:16	R/W	OSPIC_ERSRES_HLD The controller must stay without flash memory reading requests for this number of AMBA hclk clock cycles, before to perform the command of erase or erase resume. 15 - 0	0x0
15	-	- Reserved	0x0
14:10	R/W	OSPIC_ERS_CS_HI After the execution of instructions: write enable, erase, erase suspend and erase resume, the OSPI_CS remains high for at least this number of OSPI bus clock cycles.	0x0
9:8	R/W	OSPIC_EAD_TX_MD The mode of the OSPI Bus during the address phase of the erase instruction. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
7:6	R/W	OSPIC_RES_TX_MD The mode of the OSPI Bus during the transmission of the resume instruction. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
5:4	R/W	OSPIC_SUS_TX_MD The mode of the OSPI Bus during the transmission of the suspend instruction. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
3:2	R/W	OSPIC_WEN_TX_MD The mode of the OSPI Bus during the transmission of the write enable instruction. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
1:0	R/W	OSPIC_ERS_TX_MD The mode of the OSPI Bus during the instruction phase of the erase instruction. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0

Table 421: OQSPIF_ERASECMDC_REG (0x29000030)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	OSPIC_SUSSTS_DLY	0x0

Bit	Mode	Symbol/Description	Reset
		Define a timer that counts the minimum allowed delay between an erase suspend command and the next read status command. 0: Do not wait. The controller starts immediately to read the status of the flash device. 1..63: The controller waits for at least this number of 278 kHz clock cycles before to read the status of the flash device. Time starts counting when the erase resume command is applied.	

Table 422: OQSPIF_BURSTBRK_REG (0x29000034)

Bit	Mode	Symbol/Description	Reset
31:24	-	- Reserved	0x0
23	R/W	OSPIC_BRK_EN Controls the application of a special command (read burst break sequence) that is used to force the device to abandon the continuous Read mode. 0: The special command is not applied 1: The special command is applied This special command is applied by the controller to the external device under the following conditions: - The controller is in Auto mode. - The OSPIC_INST_MD = 1. - The previous command that is applied in the external device was read. - The controller wants to apply to the external device a command different than the read.	0x0
22	R/W	OSPIC_SEC_HF_DS Disables output during the transmission of the second half (OSPIC_BRK_WRD[3:0]). Setting this bit is only useful if OSPIC_BRK_EN = 1 and OSPIC_BRK_SZ >= 1. It is not applicable when the sequence is transferred in Octal mode (OSPIC_BRK_TX_MD = 3). 0: The controller drives the OSPI bus during the transmission of the OSPIC_BRK_WRD[3:0]. 1: The controller leaves the OSPI bus in Hi-Z during the transmission of the OSPIC_BRK_WORD[3:0].	0x0
21:20	R/W	OSPIC_BRK_TX_MD The mode of the OSPI Bus during the transmission of the burst break sequence. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
19:16	R/W	OSPIC_BRK_SZ The size of Burst Break Sequence 0: One byte (Send OSPIC_BRK_WRD[15:8]) 1: Two bytes (Send OSPIC_BRK_WRD[15:0]) 2-15: Three up to 16 bytes are transferred. All the bytes that are transferred, have the value of the OSPIC_BRK_WRD[15:8], except of the last byte that is the OSPIC_BRK_WRD[7:0].	0x0
15:0	R/W	OSPIC_BRK_WRD	0x0

Bit	Mode	Symbol/Description	Reset
		This is the value of a special command (read burst break sequence) that is applied by the controller to the external memory device, to force the memory device to abandon the continuous Read mode.	

Table 423: OQSPIF_STATUSCMD_REG (0x29000038)

Bit	Mode	Symbol/Description	Reset
31	-	-	0x0
		Reserved	
30	R/W	OSPIC_RSTAT_DMY_ZERO	0x0
		Defines the value of that is transferred on the OSPI bus during the phase of the dummy bytes. 0: The controller keeps the data on the bus unchanged until the bus direction is changed in Input mode. 1: Forces the dummy bytes to get the zero value (only for the cycles that are not in Input mode). Only the IO pins that are related with the Transfer mode of the dummy bytes (OSPIC_RSTAT_DMY_TX_MD) get zero value.	
29:28	R/W	OSPIC_RSTAT_DMY_TX_MD	0x0
		It describes the mode of the OSPI bus during the dummy bytes phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Octal	
27:24	R/W	OSPIC_RSTAT_DMY_NUM	0x0
		Number of dummy bytes (minus 1). Can be set 1-6 dummy bytes (values 0 up to 15). The dummy bytes are applied only when OSPIC_RSTAT_DMY_EN=1.	
23	R/W	OSPIC_RSTAT_DMY_EN	0x0
		Enables the transmission of dummy bytes, immediately after the instruction code of the read status command. 0: Do not send the dummy bytes 1: Send the dummy bytes. The number of the dummy bytes is defined by the OSPIC_RSTAT_DMY_NUM.	
22	R/W	OSPIC_STSDLY_SEL	0x0
		Defines the timer which is used to count the delay that it has to wait before to read the FLASH Status Register, after an erase or an erase resume command. 0: The delay is controlled by the OSPIC_RESSTS_DLY which counts on the OSPI_CLK clock. 1: The delay is controlled by the OSPIC_RESSUS_DLY which counts on the 222 kHz clock.	
21:16	R/W	OSPIC_RESSTS_DLY	0x0
		Defines a timer that counts the minimum required delay between the reading of the status register and of the previous erase or erase resume instruction. 0: Do not wait. The controller starts reading the Flash memory status register immediately. 1..63: The controller waits for at least this number of OSPI_CLK cycles and afterwards it starts to reading the Flash memory status register. The timer starts to count after the end of the previous erase or erase resume command. The actual timer that is used by the controller before the reading of the Flash memory status register is defined by the OSPIC_STSDLY_SEL.	

Bit	Mode	Symbol/Description	Reset
15	R/W	OSPIC_BUSY_VAL Defines the value of the Busy bit which means that the flash is busy. 0: The flash is busy when the Busy bit is equal to 0. 1: The flash is busy when the Busy bit is equal to 1.	0x0
14:12	R/W	OSPIC_BUSY_POS It describes who from the bits of status represents the Busy bit (7 - 0).	0x0
11:10	R/W	OSPIC_RSTAT_RX_MD The mode of the OSPI Bus during the receive status phase of the read status instruction 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
9:8	R/W	OSPIC_RSTAT_TX_MD The mode of the OSPI Bus during the instruction phase of the read status instruction. 0x0: Single 0x1: Dual 0x2: Quad 0x3: Octal	0x0
7:0	R/W	OSPIC_RSTAT_INST The code value of the read status instruction. It is transmitted during the instruction phase of the read status instruction.	0x0

Table 424: OQSPIF_CHCKERASE_REG (0x2900003C)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CHCKERASE Writing any value to this register during erasing, forces the controller to read the flash memory status register. Depending on the value of the Busy bit, it updates the OSPIC_ERASE_EN.	0x0

Table 425: OQSPIF_GP_REG (0x29000040)

Bit	Mode	Symbol/Description	Reset
4:3	R/W	OSPIC_PADS_SLEW OQSPI pads slew rate control. 00: Fast slew rate 11: Slow slew rate	0x0
2:1	R/W	OSPIC_PADS_DRV OQSPI pads drive current 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x0
0	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	

Table 426: OQSPIF_CTR_CTRL_REG (0x29000100)

Bit	Mode	Symbol/Description	Reset
0	R/W	OSPIC_CTR_EN Controls the AES-CTR decryption feature of the OSPIC, which enables the decryption (on-the-fly) of the data that is retrieved from the flash memory device. 0: The AES-CTR decryption is disabled. 1: The controller decrypts the content of the flash memory device that is placed in the address space that is defined by the OSPIC_CTR_SADDR_REG and OSPIC_CTR_EADDR_REG registers. The data that is placed outside the previous space, is not decrypted by the OSPIC. The decryption is performed by using the AES-CTR algorithm. The AES key is defined by the OSPIC_CTR_KEY_x_y_REG registers and the nonce value by the OSPIC_CTR_NONCE_x_y_REG registers. This configuration bit has meaning only while the controller is in Auto mode. The on-the-fly decryption is not provided in Manual mode.	0x0

Table 427: OQSPIF_CTR_SADDR_REG (0x29000104)

Bit	Mode	Symbol/Description	Reset
31:10	R/W	OSPIC_CTR_SADDR Defines the bits [31:10] of the start address in the flash memory, where an encrypted image is placed. The bits [9:0] are considered always as zero. This has meaning only when the decryption is active. See also the register OSPIC_CTR_CTRL_REG[OSPIC_CTR_EN].	0x0
9:0	-	- Reserved	0x0

Table 428: OQSPIF_CTR_EADDR_REG (0x29000108)

Bit	Mode	Symbol/Description	Reset
31:10	R/W	OSPIC_CTR_EADDR Defines the bits [31:10] of the end address in the flash memory, where an encrypted image is placed. The bits [9:0] are considered always as 0x3ff. This has meaning only when the decryption is active. See also the register OSPIC_CTR_CTRL_REG[OSPIC_CTR_EN].	0x0
9:0	-	- Reserved	0x3FF

Table 429: OQSPIF_CTR_NONCE_0_3_REG (0x2900010C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	OSPIC_CTR_NONCE_0_3 Defines the 8 bytes of the nonce value (N0 - N7) that is used by the AES-CTR algorithm to construct the counter block (CTRB). The total size of the counter block is 128 bits or 16 bytes: CTRB0 CTRB1 CTRB2 CTRB3...CTRB14 CTRB15. The first 8 bytes (CTRB0 - CTRB7) of the counter block consisted by the nonce value.	0x0

Bit	Mode	Symbol/Description	Reset
		The next 8 bytes of the counter block (CTRB8-CTRB15) are produced automatically by the hardware based on the address offset inside the encrypted image, from where the requested data are retrieved. The mapping of the nonce bytes to the corresponding OSPIC_NONCE_X_Y_REG registers is the following: {CTRB0, CTRB1, CTRB2, CTRB3} = {N0, N1, N2, N3} = OSPIC_NONCE_0_3_REG[31:0] {CRB4, CTRB5, CTRB6, CTRB7} = {N4, N5, N6, N7} = OSPIC_NONCE_4_7_REG[31:0] All these registers make sense only when OSPIC_CTR_CTRL_REG[OSPIC_CTR_EN] = 1. Do not perform access to an encrypted address range while the updating process of the nonce value is in progress.	

Table 430: OQSPIF_CTR_NONCE_4_7_REG (0x29000110)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	OSPIC_CTR_NONCE_4_7 See the description in the OSPIC_NONCE_0_3.	0x0

Table 431: OQSPIF_CTR_KEY_0_3_REG (0x29000114)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_0_3 Defines the key that is used by the AES-CTR algorithm, when the on-the-fly decryption is enabled (OSPIC_CTR_CTRL_REG[OSPIC_CTR_EN] = 1). The size of the decryption key is 256 bits or 32 bytes: K0 K1 K2 K3...K30 K31. The mapping of the bytes to the corresponding OSPIC_CTR_KEY_X_Y_REG registers is the following: {K0, K1, K2, K3} = OSPIC_CTR_KEY_0_3_REG[31:0] {K4, K5, K6, K7} = OSPIC_CTR_KEY_4_7_REG[31:0] {K8, K9, K10, K11} = OSPIC_CTR_KEY_8_11_REG[31:0] {K12, K13, K14, K15} = OSPIC_CTR_KEY_12_15_REG[31:0] {K16, K17, K18, K19} = OSPIC_CTR_KEY_16_19_REG[31:0] {K20, K21, K22, K23} = OSPIC_CTR_KEY_20_23_REG[31:0] {K24, K25, K26, K27} = OSPIC_CTR_KEY_24_27_REG[31:0] {K28, K29, K30, K31} = OSPIC_CTR_KEY_28_31_REG[31:0] All these registers make sense only when OSPIC_CTR_CTRL_REG[OSPIC_CTR_EN] = 1. Do not perform access to an encrypted address range while the updating process of the decryption key is in progress.	0x0

Table 432: OQSPIF_CTR_KEY_4_7_REG (0x29000118)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_4_7 See the description in the OSPIC_CTR_KEY_0_3.	0x0

Table 433: OQSPIF_CTR_KEY_8_11_REG (0x2900011C)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_8_11 See the description in the OSPIC_CTR_KEY_0_3.	0x0

Table 434: OQSPIF_CTR_KEY_12_15_REG (0x29000120)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_12_15 See the description in the OSPIC_CTR_KEY_0_3.	0x0

Table 435: OQSPIF_CTR_KEY_16_19_REG (0x29000124)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_16_19 See the description in the OSPIC_CTR_KEY_0_3.	0x0

Table 436: OQSPIF_CTR_KEY_20_23_REG (0x29000128)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_20_23 See the description in the OSPIC_CTR_KEY_0_3.	0x0

Table 437: OQSPIF_CTR_KEY_24_27_REG (0x2900012C)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_24_27 See the description in the OSPIC_CTR_KEY_0_3.	0x0

Table 438: OQSPIF_CTR_KEY_28_31_REG (0x29000130)

Bit	Mode	Symbol/Description	Reset
31:0	W	OSPIC_CTR_KEY_28_31 See the description in the OSPIC_CTR_KEY_0_3.	0x0

10.15 QSPI Registers

Table 439: Register map QSPIC

Address	Register	Description
0x22000000	QSPIC_CTRLBUS_REG	SPI Bus control register for Manual mode
0x22000004	QSPIC_CTRLMODE_REG	Mode Control Register
0x22000008	QSPIC_RECVDATA_REG	Received data for Manual mode
0x2200000c	QSPIC_BURSTCMDA_REG	The way of reading in Auto mode (command register A)
0x22000010	QSPIC_BURSTCMDB_REG	The way of reading in Auto mode (command register B)

Address	Register	Description
0x22000014	QSPIC_STATUS_REG	The status register of the QSPI controller
0x22000018	QSPIC_WRITEDATA_REG	Write data to SPI Bus for Manual mode
0x2200001c	QSPIC_READDATA_REG	Read data from SPI Bus for Manual mode
0x22000020	QSPIC_DUMMYDATA_REG	Send dummy clocks to SPI Bus for Manual mode
0x22000024	QSPIC_ERASECTRL_REG	Erase control register
0x22000028	QSPIC_ERASECMDA_REG	The way of erasing in Auto mode (command register A)
0x2200002c	QSPIC_ERASECMDB_REG	The way of erasing in Auto mode (command register B)
0x22000030	QSPIC_BURSTBRK_REG	Read break sequence in Auto mode
0x22000034	QSPIC_STATUSCMD_REG	The way of reading the status of external device in Auto mode
0x22000038	QSPIC_CHCKERASE_REG	Check erase progress in Auto mode
0x2200003c	QSPIC_GP_REG	General Purpose control register
0x22000040	QSPIC_AWRITECMD_REG	The way of writing in Auto mode when the external device is a serial SRAM
0x22000044	QSPIC_MEMBLEN_REG	External memory burst length configuration

Table 440: QSPIC_CTRLBUS_REG (0x22000000)

Bit	Mode	Symbol/Description	Reset
31:5	-	-	0x0
		Reserved	
4	W	QSPIC_DIS_CS	0x0
		Write 1 to disable the chip select (active low) when the controller is in Manual mode.	
3	W	QSPIC_EN_CS	0x0
		Write 1 to enable the chip select (active low) when the controller is in Manual mode.	
2	W	QSPIC_SET_QUAD	0x0
		Write 1 to set the bus mode in Quad mode when the controller is in Manual mode.	
1	W	QSPIC_SET_DUAL	0x0
		Write 1 to set the bus mode in Dual mode when the controller is in Manual mode.	
0	W	QSPIC_SET_SINGLE	0x0
		Write 1 to set the bus mode in Single SPI mode when the controller is in Manual mode.	

Table 441: QSPIC_CTRLMODE_REG (0x22000004)

Bit	Mode	Symbol/Description	Reset
31:17	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
16	R/W	QSPIC_CLK_FREE_EN Control the behavior of the QSPI_SCK when the QSPI_CS is high and QSPIC_CS_MD = 1. 0: Produces one QSPI_SCK clock pulse after each 0 to 1 transition in the QSPI_CS. 1: The QSPI_SCK clock remains always active, while the QSPI_CS is inactive. This setting has meaning only when the QSPIC_CS_MD = 1.	0x0
15	R/W	QSPIC_CS_MD Control the clock edge with which is produced the QSPI_CS signal. 0: The QSPI_CS is produced with the rising edge of the QSPI_SCK. The QSPI_SCK is always inactive while the QSPI_CS is high. 1: The QSPI_CS is produced with the falling edge of the QSPI_SCK. The behavior of the QSPI_SCK while the QSPI_CS is high is controlled by the QSPIC_CLK_FREE_EN.	0x0
14	R/W	QSPIC_SRAM_EN Define the type of the external device that is connected on the QSPIC controller. 0: The external memory device is a serial Flash. 1: The external memory device is a serial SRAM. When the external device is a serial SRAM, the erase suspend/ resume functionality of the controller is disabled. In this case the writing of the QSPIC_ERASECTRL_REG[QSPIC_ERASE_EN] bit has no effect. Also, the memory space where the external device is mapped is considered as writable.	0x0
13	R/W	QSPIC_USE_32BA Control the length of the address that the external memory device uses. 0: The external memory device uses 24 bits address. 1: The external memory device uses 32 bits address. The controller uses this bit to decide the number of the address bytes that has to transfer to the external device during Auto mode.	0x0
12	R/W	QSPIC_FORCENSEQ_EN Control the way in which a burst request from the AMBA bus is addressed by the QSPI controller. 0: The controller translates a burst access on the AMBA bus as a burst access on the QSPI bus. That results to the minimum number of command/address phases. 1: The controller splits a burst access on the AMBA bus into a number of single accesses on the QSPI bus. That results to a separate command for each beat of the burst. For example, a 4-beat word incremental AMBA read access is split into four different sequences on the QSPI bus: command/address/extra clock/read data. The QSPI_CS is low only for the time that is needed for each of these single access. This configuration bit is useful when the clock frequency of the QSPI bus is much higher than the clock of the AMBA bus. In this case the interval for which the CS remains low is minimized, achieving lower power dissipation with respect of the case where the QSPIC_FORCENSEQ_EN = 0, at cost of performance.	0x0
11:9	R/W	QSPIC_PCLK_MD Control the read pipe clock delay relative to the falling edge of QSPI_SCK. Refer to QSPI Timing for timing parameters.	0x0
8	R/W	QSPIC_RPIPE_EN Control the use of the data read pipe.	0x0

Bit	Mode	Symbol/Description	Reset
		0: The read pipe is disabled, the sampling clock is defined according to the QSPIC_RXD_NEG setting. 1: The read pipe is enabled. The delay of the sampling clock is defined according to the QSPI_PCLK_MD setting (Recommended).	
7	R/W	QSPIC_RXD_NEG Define the clock edge that is used for the capturing of the received data when the read pipe is not active (QSPIC_RPIPE_EN = 0). 0: Sampling of the received data with the positive edge of the QSPI_SCK. 1: Sampling of the received data with the negative edge of the QSPI_SCK. The internal QSPI_SCK clock that is used by the controller for the capturing of the received data has a skew in respect of the QSPI_SCK that is received by the external memory device. To improve the timing requirements of the read path, the controller supports a read pipe register with programmable clock delay. See also the QSPIC_RPIPE_EN register.	0x0
6	R/W	QSPIC_HRDY_MD This configuration bit is useful when the frequency of the QSPI clock is much lower than the clock of the AMBA bus in order not to lock the AMBA bus for a long time. 0: Adds wait states via hready signal when an access is performed on QSPIC_WRITEDATA, QSPIC_READDATA and QSPIC_DUMMYDATA registers. It is not necessary to check the QSPIC_BUSY of the QSPIC_STATUS_REG. 1: The controller does not add wait states via the hready signal when the access is performed on QSPIC_WRITEDATA, QSPIC_READDATA and QSPIC_DUMMYDATA registers. The QSPIC_BUSY bit of the QSPIC_STATUS_REG must be checked to be detected the completion of the requested access. It is applicable only when the controller is in Manual mode. In case of Auto mode, the controller always adds wait states via the hready signal.	0x0
5	R/W	QSPIC_IO3_DAT The value of QSPI_IO3 pad if QSPI_IO3_OEN is 1.	0x0
4	R/W	QSPIC_IO2_DAT The value of QSPI_IO2 pad if QSPI_IO2_OEN is 1.	0x0
3	R/W	QSPIC_IO3_OEN QSPI_IO3 output enable. Use this only in SPI or Dual SPI mode to control/hold signal. When the Auto Mode is selected (QSPIC_AUTO_MD = 1) and the QUAD SPI is used, set this bit to zero. 0: The QSPI_IO3 pad is input. 1: The QSPI_IO3 pad is output.	0x0
2	R/W	QSPIC_IO2_OEN QSPI_IO2 output enable. Use this only in SPI or Dual SPI mode to control /WP signal. When the Auto Mode is selected (QSPIC_AUTO_MD = 1) and the QUAD SPI is used, set this bit to zero. 0: The QSPI_IO2 pad is input. 1: The QSPI_IO2 pad is output.	0x0
1	R/W	QSPIC_CLK_MD Mode of the generated QSPI_SCK clock. 0: Use Mode 0 for the QSPI_CLK. The QSPI_SCK is low when QSPI_CS is high. 1: Use Mode 3 for the QSPI_CLK. The QSPI_SCK is high when QSPI_CS is high. See also the QSPIC_CS_MD register and the QSPIC_CLK_FREE_EN register.	0x0
0	R/W	QSPIC_AUTO_MD	0x0

Bit	Mode	Symbol/Description	Reset
		Mode of operation. 0: Manual mode is selected. 1: Auto Mmde is selected. During an erasing, the QSPIC_AUTO_MD goes in read-only mode (see QSPIC_ERASE_EN).	

Table 442: QSPIC_RECVDATA_REG (0x22000008)

Bit	Mode	Symbol/Description	Reset
31:0	R	QSPIC_RECVDATA This register contains the received data when the QSPIC_READDATA_REG register is used in Manual mode, to retrieve data from the external memory device and QSPIC_HRDY_MD = 1 and QSPIC_BUSY = 0.	0x0

Table 443: QSPIC_BURSTCMDA_REG (0x2200000C)

Bit	Mode	Symbol/Description	Reset
31:30	R/W	QSPIC_DMY_TX_MD It describes the mode of the SPI bus during the Dummy bytes phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
29:28	R/W	QSPIC_EXT_TX_MD It describes the mode of the SPI bus during the Extra Byte phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
27:26	R/W	QSPIC_ADR_TX_MD It describes the mode of the SPI bus during the address phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
25:24	R/W	QSPIC_INST_TX_MD It describes the mode of the SPI bus during the instruction phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
23:16	R/W	QSPIC_EXT_BYTE The value of an extra byte which is transferred after address (only if QSPIC_EXT_BYTE_EN = 1). Usually this is the Mode Bits in Dual/Quad SPI I/O instructions.	0x0
15:8	R/W	QSPIC_INST_WB Instruction Value for Wrapping Burst. This value is the selected instruction when QSPIC_WRAP_MD is equal to 1 and the access is a wrapping burst of length and	0x0

Bit	Mode	Symbol/Description	Reset
		size described by the bit fields QSPIC_WRAP_LEN and QSPIC_WRAP_SIZE respectively.	
7:0	R/W	QSPIC_INST Instruction Value for Incremental Burst or Single read access. This value is the selected instruction at the cases of incremental burst or single read access. Also this value is used when a wrapping burst is not supported (QSPIC_WRAP_MD)	0x0

Table 444: QSPIC_BURSTCMDB_REG (0x22000010)

Bit	Mode	Symbol/Description	Reset
31:16	-	- Reserved	0x0
15	R/W	QSPIC_DMY_FORCE By setting this bit, the number of dummy bytes is forced to be equal to 3. In this case the QSPIC_DMY_NUM field is overruled and has no function. 0: The number of dummy bytes is controlled by the QSPIC_DMY_NUM field. 1: Three dummy bytes are used. The QSPIC_DMY_NUM is overruled.	0x0
14:12	R/W	QSPIC_CS_HIGH_MIN Between the transmission of two different instructions to the flash memory, the QSPI Bus stays in idle state (QSPI_CS high) for at least this number of QSPI_SCK clock cycles. See the QSPIC_ERS_CS_HI and the QSPIC_WR_CS_HIGH_MIN registers for some exceptions.	0x0
11:10	R/W	QSPIC_WRAP_SIZE It describes the selected data size of a wrapping burst (QSPIC_WRAP_MD). 0x0: Byte access (8-bits) 0x1: Half word access (16 bits) 0x2: Word access (32-bits) 0x3: Reserved	0x0
9:8	R/W	QSPIC_WRAP_LEN It describes the selected length of a wrapping burst (QSPIC_WRAP_MD). 0x0: 4 beat wrapping burst 0x1: 8 beat wrapping burst 0x2: 16 beat wrapping burst 0x3: Reserved	0x0
7	R/W	QSPIC_WRAP_MD Wrap mode 0: The QSPIC_INST is the selected instruction at any access. 1: The QSPIC_INST_WB is the selected instruction at any wrapping burst access of length and size described by the registers QSPIC_WRAP_LEN and QSPIC_WRAP_SIZE respectively. In all other cases the QSPIC_INST is the selected instruction. Use this feature only when the serial Flash memory supports a special instruction for wrapping burst access.	0x0
6	R/W	QSPIC_INST_MD Instruction mode 0: Transmit instruction at any burst access. 1: Transmit instruction only in the first access after the selection of Auto mode.	0x0
5:4	R/W	QSPIC_DMY_NUM	0x0

Bit	Mode	Symbol/Description	Reset
		Number of Dummy Bytes 0x0: Zero Dummy Bytes (Do not Send Dummy Bytes) 0x1: Send 1 Dummy Byte 0x2: Send 2 Dummy Bytes 0x3: Send 4 Dummy Bytes When QSPIC_DMY_FORCE is enabled, the QSPIC_DMY_NUM is overruled. In this case the number of dummy bytes is defined by QSPIC_DMY_FORCE and is equal to 3, independent of the value of QSPIC_DMY_NUM.	
3	R/W	QSPIC_EXT_HF_DS Extra Half Disable Output. 0: If QSPIC_EXT_BYTE_EN = 1, then transmit the complete QSPIC_EXT_BYTE. 1: If QSPIC_EXT_BYTE_EN = 1, then disable (Hi-Z) output during the transmission of bits [3:0] of QSPIC_EXT_BYTE.	0x0
2	R/W	QSPIC_EXT_BYTE_EN Extra Byte Enable 0: Do not Send QSPIC_EXT_BYTE 1: Send QSPIC_EXT_BYTE	0x0
1:0	R/W	QSPIC_DAT_RX_MD It describes the mode of the SPI bus during the data phase. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0

Table 445: QSPIC_STATUS_REG (0x22000014)

Bit	Mode	Symbol/Description	Reset
31:1	-	- Reserved	0x0
0	R	QSPIC_BUSY The status of the SPI Bus. 0: The SPI Bus is idle 1: The SPI Bus is active. Read data, write data or dummy data activity is in progress. This register has meaning only in Manual mode and only when QSPIC_HRDY_MD = 1.	0x0

Table 446: QSPIC_WRITEDATA_REG (0x22000018)

Bit	Mode	Symbol/Description	Reset
31:0	W	QSPIC_WRITEDATA Writing to this register generates a data transfer from the controller to the external memory device. The data written in this register is then transferred to the memory using the selected mode of the SPI Bus (SPI, Dual SPI, Quad SPI). The data size of the access to this register can be 32-bits/16-bits/8-bits and is equal to the number of the transferred bits. This register has meaning only when the controller is in Manual mode.	0x0

Table 447: QSPIC_READDATA_REG (0x2200001C)

Bit	Mode	Symbol/Description	Reset
31:0	R	QSPIC_READDATA A read access at this register generates a data transfer from the external memory device to the QSPIC controller. The data is transferred using the selected mode of the SPI Bus (SPI, Dual SPI, Quad SPI). The data size of the access to this register can be 32-bits/16-bits/8-bits and is equal to the number of the transferred bits. This register has meaning only when the controller is in Manual mode.	0x0

Table 448: QSPIC_DUMMYDATA_REG (0x22000020)

Bit	Mode	Symbol/Description	Reset
31:0	W	QSPIC_DUMMYDATA Writing to this register generates a number of clock pulses to the SPI Bus. During the last clock of this activity in the SPI Bus, the QSPI_IOx data pads are in Hi-Z state. The data size of the access to this register can be 32-bits/16-bits/8-bits. The number of generated pulses is equal to: (size of AHB bus access)/(size of SPI bus). The size of SPI Bus is equal to 1, 2, or 4 for Single, Dual, or Quad SPI mode respectively. This register has meaning only when the controller is in Manual mode.	0x0

Table 449: QSPIC_ERASECTRL_REG (0x22000024)

Bit	Mode	Symbol/Description	Reset
31:28	-	- Reserved	0x0
27:25	R	QSPIC_ERS_STATE It shows the progress of sector/block erasing (read-only) 0x0: No Erase 0x1: Pending erase request 0x2: Erase procedure is running 0x3: Suspended Erase procedure 0x4: Finishing the Erase procedure 0x5...0x7: Reserved	0x0
24	R/W	QSPIC_ERASE_EN This bit has meaning only when the external device is a serial Flash (QSPIC_SRAM_EN = 0). During Manual mode (QSPIC_AUTO_MD = 0): This bit is in read-only mode. During Auto mode (QSPIC_AUTO_MD = 1): To request the erasing of the block/sector (QSPIC_ERS_ADDR, 12'b0), write 1 to this bit. This bit is cleared automatically with the end of erasing. Until the end of erasing the QSPIC_ERASE_EN remains in read-only mode. During the same period of time, the controller remains in Auto mode (QSPIC_AUTO_MD goes in read-only mode). In the case where the external device is a serial SRAM (QSPIC_SRAM_EN = 1) this bit is in read-only mode.	0x0
23:4	R/W	QSPIC_ERS_ADDR Defines the address of the block/sector that is requested to be erased. If QSPIC_USE_32BA = 0 (24 bits addressing), bits QSPIC_ERASECTRL_REG[23-12] determine the block/sector address bits [23-12]. QSPIC_ERASECTRL_REG[11-4] are ignored by the controller.	0x0

Bit	Mode	Symbol/Description	Reset
		If QSPIC_USE_32BA = 1 (32 bits addressing) bits QSPIC_ERASECTRL_REG[23-4] determine the block/sectors address bits [31:12]	
3:0	-	-	0x0
		Reserved	

Table 450: QSPIC_ERASECMDA_REG (0x22000028)

Bit	Mode	Symbol/Description	Reset
31:24	R/W	QSPIC_RES_INST	0x0
		The code value of the erase resume instruction.	
23:16	R/W	QSPIC_SUS_INST	0x0
		The code value of the erase suspend instruction.	
15:8	R/W	QSPIC_WEN_INST	0x0
		The code value of the write enable instruction.	
7:0	R/W	QSPIC_ERS_INST	0x0
		The code value of the erase instruction.	

Table 451: QSPIC_ERASECMDB_REG (0x2200002C)

Bit	Mode	Symbol/Description	Reset
31:30	-	-	0x0
		Reserved	
29:24	R/W	QSPIC_RESSUS_DLY	0x0
		Defines a timer that counts the minimum allowed delay between an erase suspend command and the previous erase resume command (or the initial erase command). 0x00: Do not wait. The controller starts immediately to suspend the erase procedure. 0x01..0x3F: The controller waits for at least this number of 288 kHz clock cycles before the suspension of erasing. Time starts counting after the end of the previous erase resume command (or the initial erase command).	
23:20	-	-	0x0
		Reserved	
19:16	R/W	QSPIC_ERSRES_HLD	0x0
		The controller must stay without Flash memory reading requests for this number of AMBA hclk clock cycles, before to perform the command of erase or erase resume. Allowable range: 0xF - 0x0	
15	-	-	0x0
		Reserved	
14:10	R/W	QSPIC_ERS_CS_HI	0x0
		After the execution of instructions: write enable, erase, erase suspend and erase resume, the QSPI_CS remains high for at least this number of QSPI_SCK clock cycles.	
9:8	R/W	QSPIC_EAD_TX_MD	0x0
		The mode of the SPI Bus during the address phase of the erase instruction. 0x0: Single SPI	

Bit	Mode	Symbol/Description	Reset
		0x1: Dual 0x2: Quad 0x3: Reserved	
7:6	R/W	QSPIC_RES_TX_MD The mode of the SPI Bus during the transmission of the resume instruction. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
5:4	R/W	QSPIC_SUS_TX_MD The mode of the SPI Bus during the transmission of the suspend instruction. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
3:2	R/W	QSPIC_WEN_TX_MD The mode of the SPI Bus during the transmission of the write enable instruction. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
1:0	R/W	QSPIC_ERS_TX_MD The mode of the SPI Bus during the instruction phase of the erase instruction 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0

Table 452: QSPIC_BURSTBRK_REG (0x22000030)

Bit	Mode	Symbol/Description	Reset
31:21	-	- Reserved	0x0
20	R/W	QSPIC_SEC_HF_DS Disable output during the transmission of the second half (QSPIC_BRK_WRD[3:0]). Setting this bit is only useful if QSPIC_BRK_EN = 1 and QSPIC_BRK_SZ = 1. 0: The controller drives the SPI Bus during the transmission of the QSPIC_BRK_WRD[3:0]. 1: The controller leaves the SPI Bus in Hi-Z during the transmission of the QSPIC_BRK_WORD[3:0].	0x0
19:18	R/W	QSPIC_BRK_TX_MD The mode of the SPI Bus during the transmission of the read break sequence. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0

Bit	Mode	Symbol/Description	Reset
17	R/W	QSPIC_BRK_SZ The size of the read break sequence. 0: One byte (Send QSPIC_BRK_WRD[15:8]) 1: Two bytes (Send QSPIC_BRK_WRD[15:0])	0x0
16	R/W	QSPIC_BRK_EN Controls the application of a special command (read break sequence) that is used to force the device to abandon the continuous read mode. 0: The special command is not applied 1: The special command is applied This special command is applied by the controller to the external device under the following conditions: - the controller is in Auto mode - the QSPIC_INST_MD = 1 - the previous command that has been applied in the external device was read - the controller want to apply to the external device a command different than the read.	0x0
15:0	R/W	QSPIC_BRK_WRD This is the value of a special command (read break sequence) that is applied by the controller to the external memory device, to force the memory device to abandon the continuous read mode.	0x0

Table 453: QSPIC_STATUSCMD_REG (0x22000034)

Bit	Mode	Symbol/Description	Reset
31:23	-	- Reserved	0x0
22	R/W	QSPIC_STSDLY_SEL Defines the timer which is used to count the delay that it has to wait before to read the Flash Status Register, after an erase or an erase resume command. 0: The delay is controlled by the QSPIC_RESSTS_DLY which counts on the QSPI clock. 1: The delay is controlled by the QSPIC_RESSUS_DLY which counts on the 288 kHz clock.	0x0
21:16	R/W	QSPIC_RESSTS_DLY Defines the timer that counts the minimum required delay between the reading of the status register and of the previous erase or erase resume instruction. 0x00: Do not wait. The controller starts to reading the Flash memory status register immediately. 0x01...0x3F: The controller waits for at least this number of QSPI_CLK cycles and afterwards it starts to reading the Flash memory status register. The timer starts to count after the end of the previous erase or erase resume command. The actual timer that is used by the controller before the reading of the Flash memory status register is defined by the QSPIC_STSDLY_SEL.	0x0
15	R/W	QSPIC_BUSY_VAL Defines the value of the Busy bit which means that the Flash is busy. 0: The Flash is busy when the Busy bit is equal to 0. 1: The Flash is busy when the Busy bit is equal to 1.	0x0
14:12	R/W	QSPIC_BUSY_POS	0x0

Bit	Mode	Symbol/Description	Reset
		Defines the bit of the Flash status register which represents the Busy bit (0x7 - 0x0).	
11:10	R/W	QSPIC_RSTAT_RX_MD The mode of the SPI Bus during the reception phase of the read status instruction, where the value of status register is retrieved. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
9:8	R/W	QSPIC_RSTAT_TX_MD The mode of the SPI Bus during the instruction phase of the read status instruction. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
7:0	R/W	QSPIC_RSTAT_INST The code value of the read status instruction. It is transmitted during the instruction phase of the read status instruction.	0x0

Table 454: QSPIC_CHCKERASE_REG (0x22000038)

Bit	Mode	Symbol/Description	Reset
31:0	W	QSPIC_CHCKERASE Writing any value to this register during erasing, forces the controller to read the Flash memory status register. Depending on the value of the Busy bit, it updates the QSPIC_ERASE_EN. This register has meaning only when the controller is in Auto mode and there is an erase in progress (QSPIC_ERASE_EN = 1). It has no meaning when the external device is a serial SRAM.	0x0

Table 455: QSPIC_GP_REG (0x2200003C)

Bit	Mode	Symbol/Description	Reset
15:5	R/W	- Reserved	0x0
4:3	R/W	QSPIC_PADS_SLEW QSPIC pads slew rate control. 00: Fast slew rate 11: Slow slew rate	0x0
2:1	R/W	QSPIC_PADS_DRV QSPIC pads drive current. 0: 2 mA 1: 4 mA 2: 8 mA 3: 14 mA	0x0
0	R/W	QSPIC_SELECT	0x0

Bit	Mode	Symbol/Description	Reset
		QSPIC enable. 0: Not active 1: Active	

Table 456: QSPIC_AWRITECMD_REG (0x22000040)

Bit	Mode	Symbol/Description	Reset
31:19	-	- Reserved	0x0
18:14	R/W	QSPIC_WR_CS_HIGH_MIN After the execution of the write command, the QSPI_CS remains high for at least this number of QSPI_SCK clock cycles.	0x0
13:12	R/W	QSPIC_WR_DAT_TX_MD The mode of the SPI Bus during the data phase of the write command. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
11:10	R/W	QSPIC_WR_ADR_TX_MD The mode of the SPI Bus during the address phase of the write command. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
9:8	R/W	QSPIC_WR_INST_TX_MD The mode of the SPI Bus during the instruction phase of the write command. 0x0: Single SPI 0x1: Dual 0x2: Quad 0x3: Reserved	0x0
7:0	R/W	QSPIC_WR_INST This is the value of the instruction that is used to be programmed the external SRAM device.	0x0

Table 457: QSPIC_MEMBLEN_REG (0x22000044)

Bit	Mode	Symbol/Description	Reset
31:14	-	- Reserved	0x0
13:4	R/W	QSPIC_T_CEM_CC Defines the maximum allowed time tCEM for which the QSPIC_CS can stay active (QSPI_CS = 0). It has meaning only when QSPIC_T_CEM_EN is equal to 1. See also the description of the QSPIC_T_CEM_EN for more details. The tCEM is expressed in number of QSPI clock cycles and can be calculated as follows: tCEM/(qspi_clock_period) If the result of the above equation is higher than 0x3FF, use the value 0x3FF.	0x0

Bit	Mode	Symbol/Description	Reset
3	R/W	QSPIC_T_CEM_EN This bit enables the controlling of the maximum time tCEM for which the QSPI_CS remains active. It has meaning only when Auto mode is active (QSPIC_AUTO_MD = 1) and the external device is a serial SRAM (QSPIC_SRAM_EN = 1). In the case where the external device is a serial Flash (QSPIC_SRAM_EN = 0) or the controller is in Manual mode (QSPIC_AUTO_MD = 0), this field has no any effect. This feature is useful when the external serial device is a dynamic RAM that requires refresh. If the refresh is applied only when the device is in the IDLE state (QSPI_CS = 1), the time for which the device remains in the ACTIVE state (QSPI_CS = 0) should be limited by a maximum threshold. 0:There is no any constraint regarding the maximum allowed time for which QSPI_CS can stay active. This is the case also when QSPIC_SRAM_EN = 0 or QSPIC_AUTO_MD = 0. 1:There is a maximum allowed time interval tCEM for which QSPI_CS can stay active during a burst access (for reading or writing of data). For the controller, this is considered as equal to QSPIC_T_CEM_CC x qspi_clock_period. In the case where the data transfer requires QSPI_CS to stays active for more than QSPIC_T_CEM_CC QSPI clock cycles, the QSPI controller splits the access on the SPI Bus in more than one bursts, by inserting inactive periods (QSPI_CS = 0) between them. This costs extra clock cycles for the realization of the original access, due to the additional commands that are required in the SPI Bus. The value in QSPIC_T_CEM_CC should be updated every time where the frequency of the QSPI clock is modified. The QSPI clock frequency should not be decreased more than a lowest frequency. This is the lowest frequency that enables to be performed a 32-bit word read and write access, without violating the tCEM timing requirement (the QSPI controller allows to be performed at least the transferring of one beat of the requested burst, independent of the QSPIC_T_CEM_CC limit).	0x0
2:0	R/W	QSPIC_MEMBLEN In this register, the expected behavior of the external memory device regarding the length of a burst operation is defined: 0x0: The external memory device is capable to implement incremental burst of unspecified length. 0x1: The external memory device implements a wrapping burst of length 4 bytes. 0x2: The external memory device implements a wrapping burst of length 8 bytes. 0x3: The external memory device implements a wrapping burst of length 16 bytes. 0x4: The external memory device implements a wrapping burst of length 32 bytes. 0x5: The external memory device implements a wrapping burst of length 64 bytes. 0x6...0x7: Reserved This setting is used by the QSPI controller when Auto mode is enabled (QSPIC_AUTO_MD = 1) to handle the various burst requests of the AHB Bus, in respect of the requirements of the external memory device. The external memory device may need to be configured by applying special instruction to be defined the kind of the burst operation. This can be implemented by applying this special instruction with the QSPI controller in Manual mode (QSPIC_AUTO_MD = 1). Refer to the datasheet of the external device for more information.	0x0

10.16 Retention Memory Control Registers

Table 458: Register map RETMEMCTRL

Address	Register	Description
0x400b0900	RETMEMCTRL_STALL_REG	Maximum Stall cycles Control Register

Address	Register	Description
0x400b0904	RETMEMCTRL_STATUS_REG	RAM cells Status Register
0x400b0908	RETMEMCTRL_PRIO_ARB_REG	Priority Control Register for RAM cells from 0 to 23
0x400b090c	RETMEMCTRL_STATIC_CLK_OFF_REG	Memory Static Clock Off register
0x400b0910	RETMEMCTRL_DYNAMIC_CLK_ON_REG	Memory Dynamic Clock On register
0x400b0914	RETMEMCTRL_MST_CLK_EN_REG	Memory Master Clock En register

Table 459: RETMEMCTRL_STALL_REG (0x400B0900)

Bit	Mode	Symbol/Description	Reset
11:8	R/W	AHB_DMA_MAX_STALL Maximum allowed number of stall cycles for the DMA AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
7:4	R/W	AHB_CPUS_MAX_STALL Maximum allowed number of stall cycles for the CPUS AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF
3:0	R/W	AHB_CPUC_MAX_STALL Maximum allowed number of stall cycles for the CPUC AHB interface. If exceeded, the interface will get high priority. Valid for a single access so the next access (of a burst) might end up in the queue for the same number of wait cycles. 0: don't use, not feasible and can block other interfaces 1: max 1 stall cycle 15: max 15 stall cycles	0xF

Table 460: RETMEMCTRL_STATUS_REG (0x400B0904)

Bit	Mode	Symbol/Description	Reset
1	RW1C	RAM1_OFF_BUT_ACCESS Reading 1 indicates RAM1 was off but still access was performed. Writing 1 clears the status back to 0.	0x0
0	RW1C	RAM0_OFF_BUT_ACCESS Reading 1 indicates RAM0 was off but still access was performed. Writing 1 clears the status back to 0.	0x0

Table 461: RETMEMCTRL_PRIO_ARB_REG (0x400B0908)

Bit	Mode	Symbol/Description	Reset
5:4	R/W	PRIO_ARB_DMA_RAM Priority of RAM0~RAM23 for the DMA AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x0
3:2	R/W	PRIO_ARB_CPUS_RAM Priority of RAM0~RAM23 for the CPUS AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x1
1:0	R/W	PRIO_ARB_CPUC_RAM Priority of RAM0~RAM23 for the CPUC AHB interface. 00: low priority 01: mid priority (default) 1x: highest	0x1

Table 462: RETMEMCTRL_STATIC_CLK_OFF_REG (0x400B090C)

Bit	Mode	Symbol/Description	Reset
2	R/W	SRAM2 Static clock off - SRAM2	0x0
1	R/W	SRAM1 Static clock off - SRAM1	0x0
0	R/W	SRAM0 Static clock off - SRAM0	0x0

Table 463: RETMEMCTRL_DYNAMIC_CLK_ON_REG (0x400B0910)

Bit	Mode	Symbol/Description	Reset
5	R/W	MEM_UNIT Dynamic clock on - Memory Units	0x0
4	R/W	- Reserved	0x0
3	R/W	MMI2MEM Dynamic clock on - MMI2MEM	0x0
2	R/W	- Reserved	0x0
1	R/W	ARBITER Dynamic clock on - Arbiters	0x0
0	R/W	AHB2MEM Dynamic clock on - AHB2MEM	0x0

Table 464: RETMEMCTRL_MST_CLK_EN_REG (0x400B0914)

Bit	Mode	Symbol/Description	Reset
3	R/W	MST_M33_MTB_CLK_EN Clock Enable for memctrl's master - M33 MTB If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
2	R/W	MST_DMA_CLK_EN Clock Enable for memctrl's master - DMA If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
1	R/W	MST_CPUS_CLK_EN Clock Enable for memctrl's master - CPUS If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1
0	R/W	MST_CPUC_CLK_EN Clock Enable for memctrl's master - CPUC If this value is 0, memctrl's master to memory interface conversion block's clock is off.	0x1

10.17 RTC Registers

Table 465: Register map rtc_if_3095_01

Address	Register	Description
0x40038000	RTC_REQ_REG	RTC Request register
0x40038004	RTC_MIRROR_REG	RTC Enable Mirror register
0x40038008	RTC_IF_REG	RTC Interface register
0x4003800c	RTC_IRQ_EN_REG	RTC Interrupt enable register
0x40038010	RTC_CLK_INV_REG	RTC Clock inverse register
0x40038014	RTC_CLK_GR_CYC_REG	RTC Clock cycle register
0x4003801c	RTC_EDGE_REG	RTC Edge mode register
0x40038020	RTC_IRQ_STATUS_REG	RTC Interrupt Status register
0x40038024	RTC_MR_SEL_8040_REG	RTC Selection Mirror register
0x40038028	RTC_MR_FRC0_REG	RTC FreeRun Counter0 Mirror register
0x4003802c	RTC_MR_FRC1_REG	RTC FreeRun Counter1 Mirror register
0x40038030	RTC_FRC_STATUS_REG	RTC FRC Status register
0x40038060	RTC_EXP_CLR_IRQ_REG	RTC Exp Interrupt clear register
0x40038064	RTC_EXP_IRQ_EN_REG	RTC Exp Interrupt enable register
0x40038068	RTC_EXP_TH0_REG	RTC Exp Threshold0 register
0x4003806c	RTC_EXP_TH1_REG	RTC Exp Threshold1 register
0x40038070	RTC_EXP_OP_STS_REG	RTC Exp OP Status register
0x40038080	RTC_ACC_REQ_REG	RTC Access Request clear register

Address	Register	Description
0x40038084	RTC_ACC_AUTO_EN_REG	RTC Access Auto mode register
0x40038088	RTC_ACC_OP_TYPE_REG	RTC Access OP type register
0x40038090	RTC_ACC_ADDR_REG	RTC Access Address register
0x40038094	RTC_ACC_WDATA_REG	RTC Access Write Data register
0x400380a0	RTC_ACC_BUSY_REG	RTC Access Busy register
0x40038100	WAKEUP_CNT_0_REG	Wake-up Counter0 register
0x40038104	WAKEUP_CNT_1_REG	Wake-up Counter1 register
0x40038108	GPIO_WAKEUP0_REG	DWAKEUP0 register
0x4003810c	GPIO_WAKEUP1_REG	DWAKEUP1 register
0x40038118	RTM_CONTROL_REG	RTM Control register
0x40038128	WAKEUP_SRC_CLR_SIG_REG	Wake-up source clear signal register
0x40038138	FRC_CNT_0_REG	FRC Counter0 register
0x4003813c	FRC_CNT_1_REG	FRC Counter1 register
0x40038158	WAKEUP_SRC_CLR_REG	Wake-up source clear signal read register
0x4003815c	WDOG_CNT_BIT_POS_REG	Watchdog Counter bit position register
0x40038160	XADC12_CNTL_REG	AuxADC12 Control register
0x40038164	XADC12_THR01_REG	AuxADC12 Threshold Level2 register
0x40038168	XADC12_THR23_REG	AuxADC12 Threshold Level1 register
0x4003816c	XADC12_SP_NUM_REG	AuxADC12 Sample Number register
0x40038170	XADC12_TIMER_SET_REG	AuxADC12 Timer Set register
0x40038174	COMP_INT_REG	Compare Interrupt register
0x40038178	INT_THR_REG	Interrupt Threshold register
0x4003817c	PULSE_CNT_REG	Pulse Counter register

Table 466: RTC_REQ_REG (0x40038000)

Bit	Mode	Symbol/Description	Reset
3	W	RTC_REQ_LOAD_MR Load Request of the FRC to mirroring, active high with auto clear function by loading done interrupt	0x0
2	W	RTC_REQ_CLR_IRQ Clear Request of the RTC IRQ Status, active high with auto clear function	0x0
1	W	RTC_REQ_CLR_MR Clear Request of the mirroring FRC, active high with auto clear function	0x0
0	W	RTC_REQ_CLR Clear Request of the RTC Interface, active high with auto clear function	0x0

Table 467: RTC_MIRROR_REG (0x40038004)

Bit	Mode	Symbol/Description	Reset
1	R/W	RTC_MR_EN Monitoring Enable of the RTC Free-Running-Counter(FRC)	0x0
0	R/W	RTC_OP_EN Operation Enable of the RTC Interface	0x1

Table 468: RTC_IF_REG (0x40038008)

Bit	Mode	Symbol/Description	Reset
24	R/W	RTC_IF_TYPE Operation Type of the RTC Interface Clock (wr_en or rd_en) 0: Falling edge of the bus clock (HCLK) 1: Rising edge of the bus clock (HCLK)	0x0
18	R/W	RTC_IF_2_DL_EN Operation Enable of the second logic zero duration	0x1
17	R/W	RTC_IF_1_DL_EN Operation Enable of the first logic high duration	0x1
16	R/W	RTC_IF_0_DL_EN Operation Enable of the first logic zero duration	0x1
11:8	R/W	RTC_IF_2_DL Second logic zero duration length for RTC interface signals (wr_en or rd_en) 0: 1 cycle 1: 2 cycles ... F: 16 cycles	0x0
7:4	R/W	RTC_IF_1_DL First logic high duration length for RTC interface signals (wr_en or rd_en) 0: 1 cycle 1: 2 cycles ... F: 16 cycles	0xA
3:0	R/W	RTC_IF_0_DL First logic zero duration length for RTC interface signals (wr_en or rd_en) 0: 1 cycle 1: 2 cycles ... F: 16 cycles	0x9

Table 469: RTC_IRQ_EN_REG (0x4003800C)

Bit	Mode	Symbol/Description	Reset
1:0	R/W	RTC_IRQ_EN Mask Enable of the RTC Interface's interrupt output bit[1]: Access Done of the Edge Enable bit[0]: Loading Done of the Free-Running-Counter to mirroring	0x0

Table 470: RTC_CLK_INV_REG (0x40038010)

Bit	Mode	Symbol/Description	Reset
0	R/W	RTC_CLK_INV RTC clock (32.768 kHz) inversion 0: Bypass 1: Inversion	0x0

Table 471: RTC_CLK_GR_CYC_REG (0x40038014)

Bit	Mode	Symbol/Description	Reset
3:0	R/W	RTC_CLK_GR_CYC Glitch Removal Cycles of RTC Clock (32.768 kHz) Delay cells or D-F/Fs type	0x0

Table 472: RTC_EDGE_REG (0x4003801C)

Bit	Mode	Symbol/Description	Reset
8	R/W	RTC_EDGE_AUTO_N Automatic edge enable to access RTC Core Registers (active low) 0: Automatic 1: Manual (for debug)	0x0
1:0	R/W	RTC_EDGE_EN Edge Enable to access RTC Core Registers in Manual mode bit[1]: Rising edge enable bit[0]: Falling edge enable	0x0

Table 473: RTC_IRQ_STATUS_REG (0x40038020)

Bit	Mode	Symbol/Description	Reset
1:0	R	RTC_IRQ_STATUS Interrupt Status of the RTC Interface bit[1]: Access Done of the Edge Enable bit[0]: Loading Done of the Free-Running-Counter to mirroring	0x0

Table 474: RTC_MR_SEL_8040_REG (0x40038024)

Bit	Mode	Symbol/Description	Reset
0	R	RTC_MR_SEL_8040 Mirroring Register of the RTC SEL_8040 signal	0x0

Table 475: RTC_MR_FRC0_REG (0x40038028)

Bit	Mode	Symbol/Description	Reset
31:0	R	RTC_MR_FRC0 Mirroring Registers of the RTC Free-Running-Counter[31:0]	0x0

Table 476: RTC_MR_FRC1_REG (0x4003802C)

Bit	Mode	Symbol/Description	Reset
3:0	R	RTC_MR_FRC1 Mirroring Registers of the RTC Free-Running-Counter[35:32]	0x0

Table 477: RTC_FRC_STATUS_REG (0x40038030)

Bit	Mode	Symbol/Description	Reset
2:0	R	RTC_FRC_STATUS Free-Running-Counter Status of the RTC Interface bit[2]: FRC valid data 0: Valid 1: Not valid bit[1:0]: Retry number of the current FRC read access 0: Direct 1: 1 retry 2: 2 retry (happen to glitch or not) 3: 3 retry (happen to glitch or not)	0x0

Table 478: RTC_EXP_CLR_IRQ_REG (0x40038060)

Bit	Mode	Symbol/Description	Reset
1	W	RTC_EXP_CLR_IRQ Clear Request of the RTC EXP IRQ Status, active high with auto clear function	0x0
0	W	RTC_EXP_START Start Request to operate RTC EXP IRQ, active high with auto clear function by operation done interrupt	0x0

Table 479: RTC_EXP_IRQ_EN_REG (0x40038064)

Bit	Mode	Symbol/Description	Reset
2:0	R/W	RTC_EXP_IRQ_EN Enable of Interrupt Operation for the RTC Expiration Timer bit[2]: At the time of setting RTC_EXP_START, the interrupt occurs when the RTC mirror FRC is bigger than the threshold value. bit[1]: Interrupt occurs when RTC mirror FRC is turn around. bit[0]: Interrupt occurs when RTC mirror FRC is equal to threshold value.	0x0

Table 480: RTC_EXP_TH0_REG (0x40038068)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	RTC_EXP_TH0 Threshold value [31:0] to trigger the interrupt for the Expiration Timer	0x0

Table 481: RTC_EXP_TH1_REG (0x4003806C)

Bit	Mode	Symbol/Description	Reset
3:0	R/W	RTC_EXP_TH1	0x0

Bit	Mode	Symbol/Description	Reset
		Threshold value [35:32] to trigger the interrupt for the Expiration Timer	

Table 482: RTC_EXP_OP_STS_REG (0x40038070)

Bit	Mode	Symbol/Description	Reset
8	R	RTC_EXP_OP_STATUS Operation Status of the RTC Expiration Timer 0: Idle 1: Running when RTC mirror FRC is equal to threshold value.	0x0
2:0	R	RTC_EXP_IRQ_STATUS	0x0

Table 483: RTC_ACC_REQ_REG (0x40038080)

Bit	Mode	Symbol/Description	Reset
1	W	RTC_ACC_REQ_CLR Clear Request of the AHB Master to access the RTC Core, active high with auto clear function	0x0
0	W	RTC_ACC_REQ_START Start Request of the AHB Master to access the RTC Core manually, active high with auto clear function by operation done	0x0

Table 484: RTC_ACC_AUTO_EN_REG (0x40038084)

Bit	Mode	Symbol/Description	Reset
0	R/W	RTC_ACC_AUTO_EN Automatic Operation Enable of the AHB Master by CPU WatchDog	0x0

Table 485: RTC_ACC_OP_TYPE_REG (0x40038088)

Bit	Mode	Symbol/Description	Reset
2:0	R/W	RTC_ACC_OP_TYPE Operation Enable Type to access the RTC Core bit[2]: Read enable to hold previous values (wired and) bit[1]: Read enable to hold previous values (wired or) bit[0]: Write enable to save wanted values	0x1

Table 486: RTC_ACC_ADDR_REG (0x40038090)

Bit	Mode	Symbol/Description	Reset
6:0	R/W	RTC_ACC_ADDR Address to access the RTC Core	0x1C

Table 487: RTC_ACC_WDATA_REG (0x40038094)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	RTC_ACC_WDATA	0x3

Bit	Mode	Symbol/Description	Reset
		Write Data to access the RTC Core	

Table 488: RTC_ACC_BUSY_REG (0x400380A0)

Bit	Mode	Symbol/Description	Reset
0	R	RTC_ACC_BUSY Operation Status of the AHB Bus Master 0: Idle 1: Busy	0x0

Table 489: WAKEUP_CNT_0_REG (0x40038100)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	WAKEUP_CNT_0 Count_value [31:0]: Down count: using mask_en	0xFFFFFFFF

Table 490: WAKEUP_CNT_1_REG (0x40038104)

Bit	Mode	Symbol/Description	Reset
3:0	R/W	WAKEUP_CNT_1 Count_value [35:32]: Down count: using mask_en	0xF

Table 491: GPIO_WAKEUP0_REG (0x40038108)

Bit	Mode	Symbol/Description	Reset
10:0	R/W	GPIO_WAKEUP_EDGE_INVERSION_SEL Edge sel: inversion of selected signal by [26:16] of DWAKEUP1_REG 0: High active 1: Low active	0x0

Table 492: GPIO_WAKEUP1_REG (0x4003810C)

Bit	Mode	Symbol/Description	Reset
26:16	R/W	GPIO_WAKEUP_EN_SEL Wake-up enable of selected signal : matching same bit number [16]: GPIO_P0_00 [17]: GPIO_P0_08 [18]: GPIO_P0_09 [19]: GPIO_P0_10 [20]: GPIO_P0_11 [21]: GPIO_P0_12 [22]: GPIO_P0_13 [23]: GPIO_P1_10 [24]: GPIO_P1_11	0x0

Bit	Mode	Symbol/Description	Reset
		[25]: GPIO_P1_12 [26]: GPIO_P1_13	
10:0	R	GPIO_WAKEUP_SRC GPIO wake-up source [10:0]	0x0

Table 493: RTM_CONTROL_REG (0x40038118)

Bit	Mode	Symbol/Description	Reset
20:16	R/W	RTM_CTRL_IO_RETEN_CTRL I/O retention control bit [20]: reton_vbat: P0_00 ~ P0_03 [19]: reton_fdio: P1_00 ~ P1_09 [18]: reton_dio2: P1_10 ~ P1_15, SWCLK,SWDIO [17]: reton_dio1_2: P0_08 ~ P0_13 [16]: reton_dio1_1: P0_04 ~ P0_07	0x0
13:11	R/W	RTM_CTRL_RET_RET RET_RET[2:0] Retention memory retention signal enable	0x0
10:8	R/W	RTM_CTRL_RET_SLR RET_SLR[2:0] Retention Sleep Enable: shutdown mode - contents are lost.	0x0
6:3	R/W	RTM_CTRL_PWR_DN_INFO Powerdown information	0x0
2	R/W	RTM_CTRL_PDP_ISO_SHARED_IO PDB_ISO_shared_io - default 0 1: Isolation cell disable access to GPIO4~7 Note: [24] when SEN_VDD, bit[2] should be set as well	0x0
1	R/W	RTM_INFO RTM_INFORM	0x0
0	R/W	RTM_CTRL_PDB_ISO PDB_ISO - default 1 1: Isolation cell disable, access to RTM 0: Isolation cell enable, not access to RTM	0x1

Table 494: WAKEUP_SRC_CLR_SIG_REG (0x40038128)

Bit	Mode	Symbol/Description	Reset
11:8	R	ADC_WAKEUP_STATUS ADC Sensor Wake-up status: [11]: Sensor Wake-up3 [10]: Sensor Wake-up 2 [9]: Sensor Wake-up 1 [8]: Sensor Wake-up 0	0x0
6	R/W	TIMER_IRQ_DETECT Timers IRQ detected in sleep4/5. Oring: timers irq/wdog_nmi_sync/mac_timer_irq	0x0
5	R/W	PULSE_CNT_DETECT	0x0

Bit	Mode	Symbol/Description	Reset
		Pulse CNT detect	
4	R/W	ADC_DETECT Sensor (ADC) detect	0x0
3	R/W	WATCHDOG_DETECT WatchDog detect	0x0
2	R/W	POR_INDICATOR POR indicator	0x0
1	R/W	FRC_COMPARE_DETECT FRC compare detect	0x0
0	R/W	GPIO_WAKE_UP_DETECT GPIO Wake-up signal detect: see GPIO_WAKEUP1_REG	0x0

Table 495: FRC_CNT_0_REG (0x40038138)

Bit	Mode	Symbol/Description	Reset
31:0	R	FRC_CNT_0 RTC Free runing count read value[31:0] Down count: use mask_en	0x0

Table 496: FRC_CNT_1_REG (0x4003813C)

Bit	Mode	Symbol/Description	Reset
3:0	R	FRC_CNT_1 RTC Free runing count read value[35:32] Down count: use mask_en	0x0

Table 497: WAKEUP_SRC_CLR_REG (0x40038158)

Bit	Mode	Symbol/Description	Reset
6	R	WAKEUP_TIMER_IRQ_DETECT_CLEAR Timers IRQ detect clear	0x0
5	R	WAKEUP_PULSE_CNT_DETECT_CLEAR Pulse CNT detect clear	0x0
4	R	WAKEUP_SENSOR_DETECT_CLEAR Sensor detect clear	0x0
3	R	WAKEUP_WATCHDOGCLR WatchDogClr	0x0
2	R	WAKEUP_PORCLR PorClr	0x0
1	R	WAKEUP_EQCLR EqClr	0x0
0	R	WAKEUP_GPIOWKCLR ExtWkClr	0x0

Table 498: WDOG_CNT_BIT_POS_REG (0x4003815C)

Bit	Mode	Symbol/Description	Reset
6:5	R	WATCHDOG_CNT_READ_VAL	0x0
		WatchDog Count read value	
4:0	R/W	FRC35_14_BIT_SEL	0x16
		FRC[35:14] bit selection	

Table 499: XADC12_CNTL_REG (0x40038160)

Bit	Mode	Symbol/Description	Reset
15:14	R/W	AUXADC12_BITNUM	0x0
		AuxADC12_BITNUM [1:0]	
13:12	R/W	AUXADC12_IVREF	0x0
		AuxADC12_IVREF [1:0]	
11:10	R/W	AUXADC12_ICOMPS	0x0
		AuxADC12_ICOMPS [1:0]	
9:8	R/W	AUXADC12_ICOMP	0x0
		AuxADC12_ICOMP [1:0]	
7	R/W	SS_EN_OUTPUT_VAL	0x0
		SS_EN output value @Test Control[4]	
6:4	R/W	AUXADC12_TRIM	0x0
		AuxADC12_TRIM [2:0]	
3:2	R/W	-	0x0
		Reserved	
1	R/W	AUXADC12_PDB	0x0
		AuxADC12_PDB	
0	R/W	AUXADC12_RESET	0x1
		AuxADC12_RESET	

Table 500: XADC12_THR01_REG (0x40038164)

Bit	Mode	Symbol/Description	Reset
29:28	R/W	XADC12_THR_CONFIG_CH1	0x0
		Threshold mode selection 00: Over threshold 01: Under threshold 10: Different threshold	
27:16	R/W	XADC12_THR_LEVEL1_CH1	0x0
		Threshold Level1[11:0] for Channel-1	
13:12	R/W	XADC12_THR_CONFIG_CH0	0x0
		Threshold mode selection 00: Over threshold 01: Under threshold 10: Different threshold	

Bit	Mode	Symbol/Description	Reset
11:0	R/W	XADC12_THR_LEVEL0_CH0 Threshold Level0[11:0] for Channel-0	0x0

Table 501: XADC12_THR23_REG (0x40038168)

Bit	Mode	Symbol/Description	Reset
29:28	R/W	XADC12_THR_CONFIG_CH3 Threshold mode selection 00: Over threshold 01: Under threshold 10: Different threshold	0x0
27:16	R/W	XADC12_THR_LEVEL3_CH3 Threshold Level3[11:0] for Channel-3	0x0
13:12	R/W	XADC12_THR_CONFIG_CH2 Threshold mode selection 00: Over threshold 01: Under threshold 10: Different threshold	0x0
11:0	R/W	XADC12_THR_LEVEL2_CH2 Threshold Level2[11:0] for Channel-2	0x0

Table 502: XADC12_SP_NUM_REG (0x4003816C)

Bit	Mode	Symbol/Description	Reset
7:3	R/W	ADC_SAMPLE_RANGE_SEL ADC sample select range - value of ADC data summation in any timing range	0x0
2:0	R/W	ADC_SMLP_NUM_AVR Sample number for average (sample count = 2n+2) 000 = 4-sample processing 001 = 8-sample processing 111 = 512-sample processing	0x0

Table 503: XADC12_TIMER_SET_REG (0x40038170)

Bit	Mode	Symbol/Description	Reset
28:24	R/W	ASWCH_CTRL ASWCH_CTRL [28]: Auto-switch activate [27:24]: Channel selection [27]=Ch-3, [26]=Ch-2, [25]=Ch-1, [24]=Ch-0	0x0
19:16	R/W	EXT_SEN_ACTIVATE_TMR_VAL EXT_SEN_ACTIVATE timer value : External sensor on time : Should be set under REG_AX12B_TIMER_SET[3:0] value valid range N = 0x0 ~ 0xE, max. = 0xE	0x1

Bit	Mode	Symbol/Description	Reset
13:8	R/W	IP3_ACTIVATE_TMR_VAL IP3_ACTIVATE timer value : IP3 Power on time (sub_cnt) : delta T = N x X12_Clk + 8/32.768 kHz x sub_cnt valid range = 0x00 ~ 0x3B, max. = 0x3B	0x1
7	R/W	SENSOR_DETECT_ACT Sensor detect activate 1 = Sensor mode enable 0 = Normal mode	0x0
6:4	R/W	X12_CLK_TMR_CNT_SRC X12_Clk: Timer count clock source (base = 32.768 kHz) 3'b000 = 7.81-ms period 3'b001 = 31.25-ms period 3'b010 = 62.5-ms period 3'b011 = 250-ms period 3'b100 = 1000-ms period 3'b101 = 4000-ms period 3'b110 = 16000-ms period 3'b111 = 64000-ms period	0x0
3:0	R/W	AX12B_TIMER_VAL REG_AX12B_TIMER_SET[3:0] Timer value (main_cnt) - periodic cycle = X12_Clk x main_cnt - valid range = 0x0 ~ 0xF, max value = 0xF @bit[6:4] 3'b000: range = 7.8 ~ 124.8 ms 3'b001: range = 15.6 ~ 250.0 ms 3'b010: range = 62.5 ~ 1000.0 ms 3'b011: range = 250 ~ 4000 ms 3'b100: range = 1000 ~ 16000 ms 3'b101: range = 4000 ~ 64000 ms (max. 64 s) 3'b110: range = 16000 ~ 256000 ms (max. 4.2 min) 3'b111: range = 64000 ~ 1024000 ms (max. 17.1 min)	0xF

Table 504: COMP_INT_REG (0x40038174)

Bit	Mode	Symbol/Description	Reset
23	R	COMPINT Interrtup threshold. Counter Compare Value	0x0
20	R/W	CLK_SEL Pulse counter operation clock select bit 0: 32 kHz 1: 15 MHz	0x0
19:16	R/W	PULSE_SELECT Input source Selection for PulseCnt 0: GPIO_P0_00 1: GPIO_P0_08	0xF

Bit	Mode	Symbol/Description	Reset
		2: GPIO_P0_09 3: GPIO_P0_10 4: GPIO_P0_11 5: GPIO_P0_12 6: GPIO_P0_13 7: GPIO_P1_10 8: GPIO_P1_11 9: GPIO_P1_12 10: GPIO_P1_13	
15	R/W	INT_CLR Interrupt clear	0x0
14	R/W	INT_EN Interrupt Enable 0: Disable 1: Enable	0x0
13	R/W	- Reserved	0x0
12:8	R/W	G_FILTER_THR Filtering threshold	0x0
4	R/W	G_FILTER_EN Ignore short glitch 0: Disable 1: Enable	0x0
3	R/W	EDGE_SEL Count up edge 0: Rising 1: Falling	0x0
2	R/W	CNT_RST Pulse counter Reset	0x0
1	R/W	CNT_ENABLE Pulse counter Enable	0x0
0	R/W	OSC_15M_ENABLE OSC_15Mhz Enable: default enable 1 Due to prevent glitch attack, OSC15 MHz turns on by hardware.	0x1

Table 505: INT_THR_REG (0x40038178)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	INT_THR Interrupt threshold	0x0

Table 506: PULSE_CNT_REG (0x4003817C)

Bit	Mode	Symbol/Description	Reset
31:0	R	PULSE_CNT	0x0

Bit	Mode	Symbol/Description	Reset
		Counter value at the present	

10.18 SDEMMC Registers

Table 507: Register map SDEMMC

Address	Register	Description
0x40011000	SDEMMC_SDMA_SYS_ADDR__ARGU_2_REG	SDEMMC_SDMA_SYSTEM_ADDRESS_ARGUMENT2_REG : SDMA_System_Address_Argument2
0x40011004	SDEMMC_BLOCK_REG	SDEMMC_BLOCK_REG: BlockSize_BlockCount
0x40011008	SDEMMC__ARGU_1_REG	SDEMMC_ARGUMENT1_REG: Argument1
0x4001100c	SDEMMC_TRANSFERMODE_COMMAND_REG	SDEMMC_TRANSFERMODE_COMMAND_REG: TransferMode_Command
0x40011010	SDEMMC_RESPONSE0_REG	SDEMMC_RESPONSE0_REG: Response
0x40011014	SDEMMC_RESPONSE1_REG	SDEMMC_RESPONSE1_REG: Response
0x40011018	SDEMMC_RESPONSE2_REG	SDEMMC_RESPONSE2_REG: Response
0x4001101c	SDEMMC_RESPONSE3_REG	SDEMMC_RESPONSE3_REG: Response
0x40011020	SDEMMC_BUFFERDATAPORT_REG	SDEMMC_BUFFERDATAPORT_REG: BufferDataPort
0x40011024	SDEMMC_PRESENTSTATE_REG	SDEMMC_PRESENTSTATE_REG: PresentState
0x40011028	SDEMMC_HOST_CTRL_1_REG	SDEMMC_HOSTCONTROL1_REG: HostControl1
0x4001102c	SDEMMC_CLK_CTRL_REG	SDEMMC_CLOCKCONTROL_REG: ClockControl
0x40011030	SDEMMC_NORMAL_ITNR_STATUS_REG	SDEMMC_NORMALINTERRUPTSTATUS_REG: NormalInterruptStatus
0x40011034	SDEMMC_NORMAL_ITNR_STATUS_EN_REG	SDEMMC_NORMALINTERRUPTSTATUSENABLE_REG: NormalInterruptStatusEnable
0x40011038	SDEMMC_NORMAL_ITNR_SIGNAL_EN_REG	SDEMMC_NORMALINTERRUPTSIGNALENABLE_REG: NormalInterruptSignalEnable
0x4001103c	SDEMMC_AUTOCMDERRORSTATUS_REG	SDEMMC_AUTOCMDERRORSTATUS_REG: AutoCMDErrorStatus
0x40011040	SDEMMC_CAPABILITIES0_REG	SDEMMC_CAPABILITIES0_REG: Capabilities0
0x40011044	SDEMMC_CAPABILITIES1_REG	SDEMMC_CAPABILITIES1_REG: Capabilities1
0x40011048	SDEMMC_MAX_CUR_CAPABILITIES_REG	SDEMMC_MAXIMUMCURRENTCAPABILITIES_REG : MaximumCurrentCapabilities
0x40011050	SDEMMC_FORCE_EVENET_ERRSTAT_REG	SDEMMC_FORCEEVENTFORAUTOCMDERRORSTATUS_REG: ForceEventforAutoCMDErrorStatus
0x40011054	SDEMMC_ADMAERRORSTATUS_REG	SDEMMC_ADMAERRORSTATUS_REG: ADMAErrorStatus

Address	Register	Description
0x40011058	SDEMMC_ADMA_SYS_ADDR_LOWER_REG	SDEMMC_ADMASYSTEMADDRESSLOWBITS_REG: ADMASystemAddressLowbits
0x4001105c	SDEMMC_ADMA_SYS_ADDR_UPPER_REG	SDEMMC_ADMASYSTEMADDRESSUPPERBITS_REG: ADMASystemAddressUpperbits
0x40011060	SDEMMC_PRESETVALUE0_REG	SDEMMC_PRESETVALUE0_REG: PresetValueInit_DefSpeed
0x40011064	SDEMMC_PRESETVALUE1_REG	SDEMMC_PRESETVALUE1_REG: PresetValueHighSpd_SDR12
0x40011068	SDEMMC_PRESETVALUE2_REG	SDEMMC_PRESETVALUE2_REG: PresetValueSDR25_SDR50
0x4001106c	SDEMMC_PRESETVALUE3_REG	SDEMMC_PRESETVALUE3_REG: PresetValueSDR104_DDR50
0x40011070	SDEMMC_BOOTTIMEOUT_CTRL_REG	SDEMMC_BOOTTIMEOUTCONTROL_REG: BootTimeoutControl
0x40011074	SDEMMC_DEBUGSELECTION_REG	SDEMMC_DEBUGSELECTION_REG: DebugSelection
0x40011078	SDEMMC_VENDOR_REG	SDEMMC_VENDOR_REG: Vendor
0x400110e0	SDEMMC_SHAREDBUS_CTRL_REG	SDEMMC_SHAREDBUSCONTROL_REG: SharedBusControl
0x400110fc	SDEMMC_SLOT_ITNR_STATUS_REG	SDEMMC_SLOTINTERRUPTSTATUS_REG: SlotInterruptStatus
0x40011400	SDEMMC_GLB_ITNR_STATUS_REG	SDEMMC_GLOBALINTERRUPTSTATUS_REG: GlobalInterruptStatus
0x40011404	SDEMMC_GLB_ITNR_STATUS_EN_REG	SDEMMC_GLOBALINTERRUPTSTATUSENABLE_REG: GlobalInterruptStatusEnable
0x40011408	SDEMMC_GLB_ITNR_SIGNAL_EN_REG	SDEMMC_GLOBALINTERRUPTSIGNALENABLE_REG: GlobalInterruptSignalEnable

Table 508: SDEMMC_SDMA_SYS_ADDR__ARGU_2_REG (0x40011000)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	SDMA_System_Address_Argument2 This register contains the physical system memory address used for DMA transfers or the second argument for the Auto CMD23. (1) SDMA System Address This register contains the system memory address for a SDMA transfer. When the Host Controller stops a SDMA transfer, this register points to the system address of the next contiguous data position. It can be accessed only if no transaction is executing (after a transaction stops). Read operations during transfers may return an invalid value. The Host Driver initializes this register before starting a SDMA transaction. After SDMA stops, the next system address of the next contiguous data position can be read from this register. The SDMA transfer waits at the every boundary specified by the Host SDMA Buffer Boundary in the Block Size register. The Host Controller generates DMA Interrupt to request the Host Driver to update this register. The Host Driver sets the next system address of the next data position to this register. When the most upper byte of this register (003h) is written, the Host Controller restarts the SDMA transfer. When restarting SDMA by the Resume command or by setting Continue Request in the Block Gap Control register, the Host Controller starts at the next contiguous	0x0

Bit	Mode	Symbol/Description	Reset
		address stored here in the SDMA System Address register. ADMA does not use this register. (2) Argument 2 This register is used with the Auto CMD23 to set a 32-bit block count value to the argument of the CMD23 while executing Auto CMD23. If Auto CMD23 is used with ADMA, the full 32-bit block count value can be used. If Auto CMD23 is used without AMDA, the available block count value is limited by the Block Count register. 65535 blocks is the maximum value in this case.	

Table 509: SDEMMC_BLOCK_REG (0x40011004)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	BlockCountForCurrentTransfer This register is enabled when Block Count Enable in the Transfer Mode register is set to 1 and is valid only for multiple block transfers. The HC decrements the block count after each block transfer and stops when the count reaches zero. It can be accessed only if no transaction is executing (after a transaction has stopped). Read operations during transfer return an invalid value and write operations are ignored. When saving transfer context as a result of Suspend command, the number of blocks yet to be transferred can be determined by reading this register. When restoring transfer context prior to issuing a Resume command, the HD restores the previously save block count. 0000h: Stop Count 0001h: 1 block 0002h: 2 blocks --- --- FFFFh: 65535 blocks	0x0
15	R/W	- Reserved	0x0
14:12	R/W	HostSDMABufferBoundry To perform long DMA transfer, System Address register is updated at every system boundary during DMA transfer. These bits specify the size of contiguous buffer in the system memory. The DMA transfer waits at the every boundary specified by these fields and the HC generates the DMA Interrupt to request the HD to update the System Address register. These bits support when the DMA Support in the Capabilities register is set to 1 and this function is active when the DMA Enable in the Transfer Mode register is set to 1. 000b: 4 kB (Detects A11 Carry out) 001b: 8 kB (Detects A12 Carry out) 010b: 16 kB (Detects A13 Carry out) 011b: 32 kB (Detects A14 Carry out) 100b: 64 kB (Detects A15 Carry out) 101b:128 kB (Detects A16 Carry out) 110b: 256 kB (Detects A17 Carry out) 111b: 512 kB (Detects A18 Carry out	0x0
11:0	R/W	TransferBlockSize This register specifies the block size for block data transfers for CMD17, CMD18, CMD24, CMD25, and CMD53. It can be accessed only if no transaction is	0x0

Bit	Mode	Symbol/Description	Reset
		executing (after a transaction has stopped). Read operations during transfer return an invalid value and write operations are ignored. 0000h: No Data Transfer 0001h: 1 Byte 0002h: 2 Bytes 0003h: 3 Bytes 0004h: 4 Bytes --- --- 01FFh: 511 Bytes 0200h: 512 Bytes --- --- 0800h: 2048 Bytes	

Table 510: SDEMMC__ARGU_1_REG (0x40011008)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	Argument1 The SD Command Argument is specified as bit39-8 of Command-Format.	0x0

Table 511: SDEMMC_TRANSFERMODE_COMMAND_REG (0x4001100C)

Bit	Mode	Symbol/Description	Reset
31:30	R/W	- Reserved	0x0
29:24	R/W	CommandIndex This bit is set to the command number (CMD0-63, ACMD0-63)	0x0
23:22	R/W	CommandType There are three types of special commands. Suspend, Resume and Abort. These bits are set to 00b for all other commands. Suspend Command If the Suspend command succeeds, the HC assumes the SD Bus has been released and that it is possible to issue the next command which uses the DAT line. The HC de-asserts Read Wait for read transactions and stops checking busy for write transactions. The Interrupt cycle starts in 4-bit mode. If the Suspend command fails, the HC maintains its current state and the HD restarts the transfer by setting Continue Request in the Block Gap Control Register. Resume Command The HD re-starts the data transfer by restoring the registers in the range of 000-00Dh. The HC checks for busy before starting write transfers. Abort Command If this command is set when executing a read transfer, the HC stops reads to the buffer. If this command is set when executing a write transfer, the HC stops driving the DAT line. After issuing the Abort command, the HD should issue a software reset 00b: Normal 01b: Suspend 10b: Resume 11b: Abort	0x0

Bit	Mode	Symbol/Description	Reset
21	R/W	DataPresentSelect This bit is set to 1 to indicate that data is present and is transferred using the DAT line. If is set to 0 for the following: 1. Commands using only CMD line (ex. CMD52) 2. Commands with no data transfer but using busy signal on DAT[0] line (R1b or R5b ex. CMD38) 3. Resume Command 0: No Data Present 1: Data Present	0x0
20	R/W	CommandIndexCheckEnable If this bit is set to 1, the HC checks the index field in the response to see if it has the same value as the command index. If it is not, it is reported as a Command Index Error. If this bit is set to 0, the Index field is not checked. 0: Disable 1: Enable	0x0
19	R/W	CommandCRCCheckEnable If this bit is set to 1, the HC checks the CRC field in the response. If an error is detected, it is reported as a Command CRC Error. If this bit is set to 0, the CRC field is not checked. 0: Disable 1: Enable	0x0
18	R/W	- Reserved	0x0
17:16	R/W	ResponseTypeSelect Response Type Select 00: No Response 01: Response length 136 10: Response length 48 11: Response length 48 check Busy after response	0x0
15:6	R/W	- Reserved	0x0
5	R/W	MultiSingleBlockSelect This bit enables multiple block data transfers. 0: Single Block 1: Multiple Block	0x0
4	R/W	DataTransferDirectionSelect This bit defines the direction of data transfers. 0: Write (Host to Card) 1: Read (Card to Host)	0x0
3:2	R/W	AutoCMDEnable This field determines use of auto command functions 00b: Auto Command Disabled 01b: Auto CMD12 Enable 10b: Auto CMD23 Enable	0x0

Bit	Mode	Symbol/Description	Reset
		11b: Reserved There are two methods to stop Multiple-block read and write operation. (1) Auto CMD12 Enable Multiple-block read and write commands for memory require CMD12 to stop the operation. When this field is set to 01b, the Host Controller issues CMD12 automatically when last block transfer is completed. Auto CMD12 error is indicated to the Auto CMD Error Status register. The Host Driver does not set this bit if the command does not require CMD12. (2) Auto CMD23 Enable When this bit field is set to 10b, the Host Controller issues a CMD23 automatically before issuing a command specified in the Command Register. The following conditions are required to use the Auto CMD23. - Auto CMD23 Supported (Host Controller Version is 3.00 or later) - A memory card that supports CMD23 (SCR[33] = 1) - If DMA is used, it is ADMA. - Only when CMD18 or CMD25 is issued By writing the Command register, the Host Controller issues a CMD23 first and then issues a command specified by the Command Index in Command register 32-bit block count value for CMD23 is set to SDMA System Address/Argument 2 register.	
1	R/W	BlockCountEnable This bit is used to enable the Block count register, which is only relevant for multiple block transfers. When this bit is 0, the Block Count register is disabled, which is useful in executing an infinite transfer. 0: Disable 1: Enable	0x0
0	R/W	DMAEnable DMA can be enabled only if DMA Support bit in the Capabilities register is set. If this bit is set to 1, a DMA operation begins when the HD writes to the upper byte of Command register (00Fh). 0: Disable 1: Enable	0x0

Table 512: SDEMMC_RESPONSE0_REG (0x40011010)

Bit	Mode	Symbol/Description	Reset
31:0	R	Reponse0	0x0

Table 513: SDEMMC_RESPONSE1_REG (0x40011014)

Bit	Mode	Symbol/Description	Reset
31:0	R	Reponse1	0x0

Table 514: SDEMMC_RESPONSE2_REG (0x40011018)

Bit	Mode	Symbol/Description	Reset
31:0	R	Reponse2	0x0

Bit	Mode	Symbol/Description	Reset

Table 515: SDEMMC_RESPONSE3_REG (0x4001101C)

Bit	Mode	Symbol/Description	Reset
31:0	R	Reponse3	0x0

Table 516: SDEMMC_BUFFERDATAPORT_REG (0x40011020)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	BufferData The Host Controller Buffer can be accessed through this 32-bit Data Port Register.	0x0

Table 517: SDEMMC_PRESENTSTATE_REG (0x40011024)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	- Reserved	0x0
28:25	R	DAT7TO4LineSignalLevel This status is used to check DAT line level to recover from errors, and for debugging. D28 - DAT[7] D27 - DAT[6] D26 - DAT[5] D25 - DAT[4]	0xF
24	R	CMDLineSignalLevel This status is used to check CMD line level to recover from errors, and for debugging.	0x1
23:20	R	DAT3TO0LineSignalLevel This status is used to check DAT line level to recover from errors, and for debugging. This is especially useful in detecting the busy signal level from DAT[0]. D23 - DAT[3] D22 - DAT[2] D21 - DAT[1] D20 - DAT[0]	0xF
19	R	WriteProtectSwitchPinLevel The Write Protect Switch is supported for memory and combo cards. This bit reflects the SDWP# pin. 0: Write protected (SDWP# = 0) 1: Write enabled (SDWP# = 1)	0x0
18	R	CardDetectPinLevel This bit reflects the inverse value of the SDCD# pin. 0: No Card present (SDCD# = 1) 1: Card present (SDCD# = 0)	0x0

Bit	Mode	Symbol/Description	Reset
17	R	CardStateStable This bit is used for testing. If it is 0, the Card Detect Pin Level is not stable. If this bit is set to 1, it means the Card Detect Pin Level is stable. The Software Reset For All in the Software Reset Register does not affect this bit. 0: Reset of Debouncing 1: No Card or Inserted	0x0
16	R	CardInserted This bit indicates whether a card has been inserted. Changing from 0 to 1 generates a Card Insertion interrupt in the Normal Interrupt Status register and changing from 1 to 0 generates a Card Removal Interrupt in the Normal Interrupt Status register. The Software Reset For All in the Software Reset register should not affect this bit. If a Card is removed while its power is on and its clock is oscillating, the HC clears SD Bus Power in the Power Control register and SD Clock Enable in the Clock control register. In addition, the HD should clear the HC by the Software Reset For All in Software register. The card detect is active regardless of the SD Bus Power. 0: Reset or Debouncing or No Card 1: Card Inserted	0x0
15:12	R/W	- Reserved	0x0
11	R	BufferReadEnable This status is used for non-DMA read transfers. This read only flag indicates that valid data exists in the host side buffer status. If this bit is 1, readable data exists in the buffer. A change of this bit from 1 to 0 occurs when all the block data is read from the buffer. A change of this bit from 0 to 1 occurs when all the block data is ready in the buffer and generates the Buffer Read Ready Interrupt. 0: Read Disable 1: Read Enable.	0x0
10	R	BufferWriteEnable This status is used for non-DMA write transfers. This read only flag indicates if space is available for write data. If this bit is 1, data can be written to the buffer. A change of this bit from 1 to 0 occurs when all the block data is written to the buffer. A change of this bit from 0 to 1 occurs when top of block data can be written to the buffer and generates the Buffer Write Ready Interrupt. 0: Write Disable 1: Write Enable	0x0
9	R	ReadTransferActive This status is used for detecting completion of a read transfer. This bit is set to 1 for either of the following conditions: - After the end bit of the read command - When writing a 1 to continue Request in the Block Gap Control register to restart a read transfer This bit is cleared to 0 for either of the following conditions: - When the last data block as specified by block length is transferred to the system. - When all valid data blocks have been transferred to the system and no current block transfers are being sent as a result of the Stop At Block Gap Request set to 1. A transfer complete interrupt is generated when this bit changes to 0.	0x0

Bit	Mode	Symbol/Description	Reset
		1: Transferring data 0: No valid data	
8	R	WriteTransferActive This status indicates a write transfer is active. If this bit is 0, it means no valid write data exists inthe HC. This bit is set in either of the following cases: - After the end bit of the write command. - When writing a 1 to Continue Request in the Block Gap Control register to restart a writetransfer. This bit is cleared in either of the following cases: - After getting the CRC status of the last data block as specified by the transfer count (Single or Multiple) - After getting a CRC status of any block where data transmission is about to be stopped by a Stop At Block Gap Request. During a write transaction, a Block Gap Event interrupt is generated when this bit is changed to 0, as a result of the Stop At Block Gap Requestbeing set. This status is useful for the HD in determining when to issue commands during write busy. 1: transferring data 0: No valid data	0x0
7:4	R/W	- Reserved	0x0
3	R	ReTuningRequest Re-Tuning Request Host Controller may request Host Driver to execute re-tuning sequence by setting this bit when the data window is shifted by temperature drift and a tuned sampling point does not have a good margin to receive correct data. This bit is cleared when a command is issued with setting Execute Tuning in the Host Control 2 register. Changing of this bit from 0 to 1 generates Re-Tuning Event. Refer to Normal Interrupt registers for more detail. This bit isn't set to 1 if Sampling Clock Select in the Host Control 2 register is set to 0 (using fixed sampling clock). 1: Sampling clock needs re-tuning 0: Fixed or well tuned sampling clock	0x0
2	R	DATLineActive This bit indicates whether one of the DAT line on SD bus is in use. 1: DAT line active 0: DAT line inactive	0x0
1	R	CommandInhibitDAT This status bit is generated if either the DAT Line Active or the Read transfer Active is set to 1. If this bit is 0, it indicates the HC can issue the next SD command. Commands with busy signal belong to Command Inhibit (DAT) (ex. R1b, R5b type). Changing from 1 to 0 generates a Transfer Complete interrupt in the Normal interrupt status register. Note: The SD Host Driver can save registers in the range of 000-00Dh for a suspend transaction after this bit has changed from 1 to 0. 1: Cannot issue command which uses the DAT line 0: Can issue command which uses the DAT line	0x0

Bit	Mode	Symbol/Description	Reset
0	R	CommandInhibitCMD If this bit is 0, it indicates the CMD line is not in use and the HC can issue a SD command using the CMD line. This bit is set immediately after the Command register (00Fh) is written. This bit is cleared when the command response is received. Even if the Command Inhibit (DAT) is set to 1, Commands using only the CMD line can be issued if this bit is 0. Changing from 1 to 0 generates a Command complete interrupt in the Normal Interrupt Status register. If the HC cannot issue the command because of a command conflict error or because of Command Not Issued By Auto CMD12 Error, this bit remains 1 and the Command Complete is not set. Status issuing Auto CMD12 is not read from this bit. Auto CMD12 and Auto CMD23 consist of two responses. In this case, this bit is not cleared by the response of CMD12 or CMD23 but cleared by the response of a read/write command. Status issuing Auto CMD12 is not read from this bit. So if a command is issued during Auto CMD12 operation, Host Controller manages to issue two commands: CMD12 and a command set by Command register.	0x0

Table 518: SDEMMC_HOST_CTRL_1_REG (0x40011028)

Bit	Mode	Symbol/Description	Reset
31:27	R/W	- Reserved	0x0
26	R/W	WakeupEventEnableOnSDCardRemoval This bit enables wake-up event via Card Removal assertion in the Normal Interrupt Status register. FN_WUS (Wake-up Support) in CIS does not affect this bit. 1: Enable 0: Disable	0x0
25	R/W	WakeupEventEnableOnSDCardInsertion This bit enables wake-up event via Card Insertion assertion in the Normal Interrupt Status register. FN_WUS (Wake-up Support) in CIS does not affect this bit. 1: Enable 0: Disable	0x0
24	R/W	WakeupEventEnableOnCardInterrupt This bit enables wake-up event via Card Interrupt assertion in the Normal Interrupt Status register. This bit can be set to 1 if FN_WUS (Wake-up Support) in CIS is set to 1. 1: Enable 0: Disable	0x0
23	R/W	BootAckChk To check for the boot acknowledge in boot operation. 1: Wait for boot ack from eMMC card 0: Will not wait for boot ack from eMMC card	0x0
22	R/W	AltBootEn To start boot code access in alternative mode. 1: To start alternate boot mode access 0: To stop alternate boot mode access	0x0
21	R/W	BootEn	0x0

Bit	Mode	Symbol/Description	Reset
		To start boot code access 1: To start boot code access 0: To stop boot code access	
20	R/W	SpiMode SPI mode enable bit. 1: SPI mode 0: SD mode	0x0
19	R/W	InterruptAtBlockGap This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. Setting to 1 enables interrupt detection at the block gap for a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit should be set to 0. When the HD detects an SD card insertion, it sets this bit according to the CCCR of the SDIO card.	0x0
18	R/W	ReadWaitControl The read wait function is optional for SDIO cards. If the card supports read wait, set this bit to enable use of the read wait protocol to stop read data using DAT[2] line. Otherwise, the HC has to stop the SD clock to hold read data, which restricts commands generation. When the HD detects an SD card insertion, it must set this bit according to the CCCR of the SDIO card. If the card does not support read wait, this bit must never be set to 1, otherwise, DAT line conflict may occur. If this bit is set to 0, Suspend/Resume cannot be supported. 1: Enable Read Wait Control 0: Disable Read Wait Control	0x0
17	R/W	ContinueRequest This bit is used to restart a transaction which was stopped using the Stop At Block Gap Request. To cancel stop at the block gap, set Stop At block Gap Request to 0 and set this bit to restart the transfer. The HC automatically clears this bit in either of the following cases: 1) In the case of a read transaction, the DAT Line Active changes from 0 to 1 as a read transaction restarts. 2) In the case of a write transaction, the Write transfer active changes from 0 to 1 as the write transaction restarts. Therefore, it is not necessary for Host driver to set this bit to 0. If Stop At Block Gap Request is set to 1, any write to this bit is ignored. 1: Restart 0: Ignored	0x0
16	R/W	StopAtBlockGapRequest This bit is used to stop executing a transaction at the next block gap for non-DMA, SDMA, and ADMA transfers. Until the transfer complete is set to 1, indicating a transfer completion the HD leaves this bit set to 1. Clearing both the Stop At Block Gap Request and Continue Request does not cause the transaction to restart. Read Wait is used to stop the read transaction at the block gap. The HC must honour Stop At Block Gap Request for write transfers, but for read transfers it requires that the SD card support Read Wait. Therefore, the HD must not set this bit during read transfers unless the SD card supports Read Wait and has set Read Wait Control to 1. In case of write transfers in which the HD writes data to the Buffer Data Port register, the HD sets this bit after all block data is written. If this bit is set to 1, the HD must not write data to Buffer data port register. This bit affects Read Transfer Active, Write Transfer Active, DAT line active and Command Inhibit (DAT) in the Present State register. 1: Stop 0: Transfer	0x0

Bit	Mode	Symbol/Description	Reset
15:13	R/W	- Reserved	0x0
12	R/W	HardwareRst Hardware reset signal is generated for eMMC card when this bit is set 1: Drives the hardware reset pin as ZERO (Active LOW to eMMC card) 0: Deassert the hardware reset pin	0x0
11:9	R/W	SDBusVoltageSelect By setting these bits, the HD selects the voltage level for the SD card. Before setting this register, the HD checks the voltage support bits in the capabilities register. If an unsupported voltage is selected, the Host System must not supply SD bus voltage 111b: 3.3 V (Typ.) 110b: 3.0 V (Typ.) 101b: 1.8 V (Typ.) 100b - 000b: Reserved	0x0
8	R/W	SDBusPower Before setting this bit, the SD host driver sets SD Bus Voltage Select. If the HC detects the No Card State, this bit is cleared. 1: Power on 0: Power off	0x0
7	R/W	CardDetectSignalSelection This bit selects source for card detection. 1: The card detect test level is selected 0: SDCD# is selected (for normal use)	0x0
6	R/W	CardDetectTestLevel This bit is enabled while the Card Detect Signal Selection is set to 1 and it indicates card inserted or not. Generates (card ins or card removal) interrupt when the normal int sts enable bit is set. 1: Card Inserted 0: No Card	0x0
5	R/W	ExtendedDataTransferWidth This bit controls 8-bit bus width mode for embedded device. Support of this function is indicated in 8-bit Support for Embedded Device in the Capabilities register. If a device supports 8-bit bus mode, this bit may be set to 1. If this bit is 0, bus width is controlled by Data Transfer Width in the Host Control 1 register.This bit is not effective when multiple devices are installed on a bus slot (Slot Type is set to 10b in the Capabilities register). In this case, each device bus width is controlled by Bus Width Preset field in the Shared Bus register. 1: 8-bit Bus Width 0: Bus Width is Selected by Data Transfer Width	0x0
4:3	R/W	DMASelect One of supported DMA modes can be selected. The host driver checks support of DMA modes by referring the Capabilities register. 00: SDMA is selected 01: 32-bit Address ADMA1 is selected 10: 32-bit Address ADMA2 is selected 11: 64-bit Address ADMA2 is selected	0x0

Bit	Mode	Symbol/Description	Reset
2	R/W	HighSpeedEnable This bit is optional. Before setting this bit, the HD checks the High Speed Support in the capabilities register. If this bit is set to 0 (default), the HC outputs CMD line and DAT lines at the falling edge of the SD clock (up to 25 MHz/20 MHz for MMC). If this bit is set to 1, the HC outputs CMD line and DAT lines at the rising edge of the SD clock (up to 50 MHz for SD/52 MHz for MMC)/208 MHz (for SD3.0). If Preset Value Enable in the Host Control 2 register is set to 1, Host Driver needs to reset SD Clock Enable before changing this field to avoid generating clock glitches. After setting this field, the Host Driver sets SD Clock Enable again. 1: High Speed Mode 0: Normal Speed Mode	0x0
1	R/W	DataTransferWidth This bit selects the data width of the HC. The HD selects it to match the data width of the SD card. 1: 4 bit mode 0: 1 bit mode	0x0
0	R/W	LEDControl This bit is used to caution the user not to remove the card while the SD card is being accessed. If the software is going to issue multiple SD commands, this bit can be set during all transactions. It is not necessary to change for each transaction. 1: LED on 0: LED off	0x0

Table 519: SDEMMC_CLK_CTRL_REG (0x4001102C)

Bit	Mode	Symbol/Description	Reset
31:27	R/W	- Reserved	0x0
26	R/W	SoftwareResetForDATLine Only part of data circuit is reset. The following registers and bits are cleared by this bit: **Buffer Data Port Register** - Buffer is cleared and initialized. **Present State register** - Buffer read Enable - Buffer write Enable - Read Transfer Active - Write Transfer Active - DAT Line Active - Command Inhibit (DAT) **Block Gap Control register** - Continue Request - Stop At Block Gap Request **Normal Interrupt Status register** - Buffer Read Ready - Buffer Write Ready - Block Gap Event - Transfer Complete	0x0

Bit	Mode	Symbol/Description	Reset
		1: Reset 0: Work	
25	R/W	SoftwareResetForCMDLine Only part of command circuit is reset. The following registers and bits are cleared by this bit: - Present State register - Command Inhibit (CMD) - Normal Interrupt Status register - Command Complete 1: Reset 0: Work	0x0
24	R/W	SoftwareResetForAll This reset affects the entire HC except for the card detection circuit. Register bits of type ROC, RW, RW1C, RWAC are cleared to 0. During its initialization, the HD sets this bit to 1 to reset the HC. The HC resets this bit to 0 when capabilities registers are valid and the HD can read them. Additional use of Software Reset For All may not affect the value of the Capabilities registers. If this bit is set to 1, the SD card resets itself and must be re initialized by the HD. 1: Reset 0: Work	0x0
23:20	R/W	- Reserved	0x0
19:16	R/W	DataTimeoutCounterValue This value determines the interval by which DAT line time-outs are detected. See the Data Time-out Error in the Error Interrupt Status register for information on factors that dictate time-out generation. Time-out clock frequency is generated by dividing the sdclockTMCLK by this value. When setting this register, prevent inadvertent time-out events by clearing the Data Time-out Error Status Enable (in the Error Interrupt Status Enable register) 1111: Reserved 1110: TMCLK * 2^27 --- 0001: TMCLK * 2^14 0000: TMCLK * 2^13	0x0
15:8	R/W	SDCLKFrequencySelect This register is used to select the frequency of the SDCLK pin. The frequency is not programmed directly; rather this register holds the divisor of the Base Clock Frequency For SD clock in the capabilities register. Only the following settings are allowed. (1) 8-bit Divided Clock Mode 80h: Base clock divided by 256 40h: Base clock divided by 128 20h: Base clock divided by 64 10h: Base clock divided by 32 08h: Base clock divided by 16 04h: Base clock divided by 8 02h: Base clock divided by 4 01h: Base clock divided by 2	0x0

Bit	Mode	Symbol/Description	Reset
		00h: Base clock(10 MHz-63 MHz) Setting 00h specifies the highest frequency of the SD Clock. When setting multiple bits, the most significant bit is used as the divisor. But multiple bits should not be set. The two default divider values can be calculated by the frequency that is defined by the Base Clock Frequency For SD Clock in the Capabilities register. 1) 25 MHz divider value 2) 400 kHz divider value The frequency of the SDCLK is set by the following formula: Clock Frequency = (Base clock)/divisor. Thus choose the smallest possible divisor which results in a clock frequency that is less than or equal to the target frequency. Maximum Frequency for SD = 50 MHz (base clock) Maximum Frequency for MMC = 52 MHz (base clock) Minimum Frequency = 195.3125 kHz (50 MHz / 256), same calculation for MMC also (2) 10-bit Divided Clock Mode Host Controller Version 3.00 supports this mandatory mode instead of the 8-bit Divided Clock Mode. The length of divider is extended to10 bits and all divider values are supported. 3FFh: 1/2046 Divided Clock N: 1/2N Divided Clock (Duty 50%) 002h: 1/4 Divided Clock 001h: 1/2 Divided Clock 000h: Base Clock (10 MHz-254 MHz)	
7:6	R/W	UpperBitsofSDCLKFrequencySelect Bit 07-06 is assigned to bit 09-08 of clock divider in SDCLK Frequency Select	0x0
5	R/W	ClockGeneratorSelect This bit is used to select the clock generator mode in SDCLK Frequency Select. If the Programmable Clock Mode is supported (non-zero value is set to Clock Multiplier in the Capabilities register), this bit attribute is RW, and if not supported, this bit attribute is RO and zero is read. This bit depends on the setting of Preset Value Enable in the Host Control 2 register. If the Preset Value Enable = 0, this bit is set by Host Driver. If the Preset Value Enable = 1, this bit is automatically set to a value specified in one of Preset Value registers. 1: Programmable Clock Mode 0: Divided Clock Mode	0x0
4:3	R/W	- Reserved	0x0
2	R/W	SDClockEnable The HC stops SDCLK when writing this bit to 0. SDCLK frequency Select can be changed when this bit is 0. Then, the HC maintains the same clock frequency until SDCLK is stopped (Stop at SDCLK = 0). If the HC detects the No Card state, this bit is cleared. 1: Enable 0: Disable	0x0
1	R	InternalClockStable	0x0

Bit	Mode	Symbol/Description	Reset
		This bit is set to 1 when SD clock is stable after writing to Internal Clock Enable in this register to 1. The SD Host Driver waits to set SD Clock Enable until this bit is set to 1. Note: This is useful when using PLL for a clock oscillator that requires setup time. 1: Ready 0: Not Ready	
0	R/W	InternalClockEnable This bit is set to 0 when the HD is not using the HC or the HC awaits a wake-up event. The HC should stop its internal clock to go very low-power state. Still, registers can be read and written. Clock starts to oscillate when this bit is set to 1. When clock oscillation is stable, the HC sets Internal Clock Stable in this register to 1. This bit does not affect card detection. 1: Oscillate 0: Stop	0x0

Table 520: SDEMMC_NORMAL_ITNR_STATUS_REG (0x40011030)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	VendorSpecificErrorStatus Reserved	0x0
28	R/W	TargetResponseError Occurs when detecting ERROR in m_hresp(dma transaction) 0: No error 1: Error	0x0
27:26	R/W	- Reserved	0x0
25	R/W	ADMAError This bit is set when the Host Controller detects errors during ADMA based data transfer. The state of the ADMA at an error occurrence is saved in the ADMA Error Status Register. 1: Error 0: No error	0x0
24	R/W	AutoCMDError Auto CMD12 and Auto CMD23 use this error status. This bit is set when detecting that one of the bits D00-D04 in Auto CMD Error Status register changed from 0 to 1. In case of Auto CMD12, this bit is set to 1, not only when the errors in Auto CMD12 occur but also when Auto CMD12 is not executed due to the previous command error. 0: No error 1: Error	0x0
23	R/W	CurrentLimitError By setting the SD Bus Power bit in the Power Control Register, the HC is requested to supply power for the SD Bus. If the HC supports the Current Limit Function, it can be protected from an Illegal card by stopping power supply to the card in which case this bit indicates a failure status. Reading 1 means the HC is not supplying power to SD card due to some failure. Reading 0 means that the HC is supplying power and no error has occurred. This bit must always be set to 0, if the HC does not support this function. 0: No error	0x0

Bit	Mode	Symbol/Description	Reset
		1: Power fail	
22	R/W	DataEndBitError Occurs when detecting 0 at the end bit position of read data which uses the DAT line or the end bit position of the CRC status. 0: No error 1: Error	0x0
21	R/W	DataCRCError Occurs when detecting CRC error when transferring read data which uses the DAT line or when detecting the Write CRC Status having a value of other than 010. 0: No error 1: Error	0x0
20	R/W	DataTimeoutError Occurs when detecting one of following timeout conditions. 1. Busy Timeout for R1b, R5b type. 2. Busy Timeout after Write CRC status 3. Write CRC status Timeout 4. Read Data Timeout 0: No Error 1: Timeout	0x0
19	R/W	CommandIndexError Occurs if a Command Index error occurs in the Command Response. 0: No error 1: Error	0x0
18	R/W	CommandEndBitError Occurs when detecting that the end bit of a command response is 0. 0: No error 1: End bit error generated	0x0
17	R/W	CommandCRCError Command CRC Error is generated in two cases. 1. If a response is returned and the Command Time-out Error is set to 0, this bit is set to 1 when detecting a CRT error in the command response. 2. The HC detects a CMD line conflict by monitoring the CMD line when a command is issued. If the HC drives the CMD line to 1 level, but detects 0 level on the CMD line at the next SDCLK edge, then the HC aborts the command (Stop driving CMD line) and set this bit to 1. The Command Timeout Error must also be set to 1 to distinguish CMD line conflict. 0: No error 1: CRC error generated	0x0
16	R/W	CommandTimeoutError Occurs only if the no response is returned within 64 SDCLK cycles from the end bit of the command. If the HC detects a CMD line conflict, in which case Command CRC Error is also set. This bit must be set without waiting for 64 SDCLK cycles because the command will be aborted by the HC. 0: No error 1: Timeout	0x0
15	R	ErrorInterrupt	0x0

Bit	Mode	Symbol/Description	Reset
		If any of the bits in the Error Interrupt Status Register are set, then this bit is set. Therefore, the HD can test for an error by checking this bit first. 0: No error. 1: Error.	
14	R	BootTerminateInterrupt This status is set if the boot operation get terminated 0: Boot operation is not terminated. 1: Boot operation is terminated	0x0
13	R	BootAckRcv This status is set if the boot acknowledge is received from device. 0: Boot ack is not received. 1: Boot ack is received.	0x0
12	R	ReTuningEvent This status is set if Re-Tuning Request in the Present State register changes from 0 to 1. Host Controller requests Host Driver to perform re-tuning for next data transfer. Current data transfer (not large block count) can be completed without re-tuning. 1: Re-Tuning should be performed 0: Re-Tuning is not required	0x0
11	R	INT_C This status is set if INT_C is enabled and INT_C# pin is in low level. Writing this bit to 1 does not clear this bit. It is cleared by resetting the INT_C interrupt factor.	0x0
10	R	INT_B This status is set if INT_B is enabled and INT_B# pin is in low level. Writing this bit to 1 does not clear this bit. It is cleared by resetting the INT_B interrupt factor.	0x0
9	R	INT_A This status is set if INT_A is enabled and INT_A# pin is in low level. Writing this bit to 1 does not clear this bit. It is cleared by resetting the INT_A interrupt factor.	0x0
8	R	CardInterrupt Writing this bit to 1 does not clear this bit. It is cleared by resetting the SD card interrupt factor. In 1-bit mode, the HC detects the Card Interrupt without SD Clock to support wake-up. In 4-bit mode, the card interrupt signal is sampled during the interrupt cycle, so there are some sample delays between the interrupt signal from the card and the interrupt to the Host system. when this status is set and the HD needs to start this interrupt service, Card Interrupt Status Enable in the Normal Interrupt Status register must be set to 0 to clear the card interrupt statuses latched in the HC and stop driving the Host System. After completion of the card interrupt service (the reset factor in the SD card and the interrupt signal may not be asserted), set Card Interrupt Status Enable to 1 and start sampling the interrupt signal again. Interrupt detected by DAT[1] is supported when there is a card per slot. In case of shared bus, interrupt pins are used to detect interrupts. If 000b is set to Interrupt Pin Select in the Shared Bus Control register, this status is effective. Non-zero value is set to Interrupt Pin Select, INT_A, INT_B or INT_C is then used to device interrupts. 0: No Card Interrupt 1: Generate Card Interrupt	0x0
7	R/W	CardRemoval	0x0

Bit	Mode	Symbol/Description	Reset
		This status is set if the Card Inserted in the Present State register changes from 1 to 0. When the HD writes this bit to 1 to clear this status the status of the Card Inserted in the Present State register should be confirmed. Because the card detect may possibly be changed when the HD clear this bit an Interrupt event may not be generated. 0: Card State Stable or Debouncing 1: Card Removed	
6	R/W	CardInsertion This status is set if the Card Inserted in the Present State register changes from 0 to 1. When the HD writes this bit to 1 to clear this status the status of the Card Inserted in the Present State register should be confirmed. Because the card detect may possibly be changed when the HD clear this bit an Interrupt event may not be generated. 0: Card State Stable or Debouncing 1: Card Inserted	0x0
5	R/W	BufferReadReady This status is set if the Buffer Read Enable changes from 0 to 1. Buffer Read Ready is set to 1 for every CMD19 execution in tuning procedure. 0: Not Ready to read Buffer. 1: Ready to read Buffer.	0x0
4	R/W	BufferWriteReady This status is set if the Buffer Write Enable changes from 0 to 1. 0: Not Ready to Write Buffer. 1: Ready to Write Buffer.	0x0
3	R/W	DMAInterrupt This status is set if the HC detects the Host DMA Buffer Boundary in the Block Size regiser. 0: No DMA Interrupt 1: DMA Interrupt is Generated	0x0
2	R/W	BlockGapEvent If the Stop At Block Gap Request in the Block Gap Control Register is set, this bit is set. **Read Transaction:** This bit is set at the falling edge of the DAT Line Active Status (When the transaction is stopped at SD Bus timing. The Read Wait must be supported inorder to use this function). **Write Transaction:** This bit is set at the falling edge of Write Transfer Active Status (After getting CRC status at SD Bus timing). 0: No Block Gap Event 1: Transaction stopped at Block Gap	0x0
1	R/W	TransferComplete This bit is set when a read / write transaction is completed. **Read Transaction:** This bit is set at the falling edge of Read Transfer Active Status.	0x0

Bit	Mode	Symbol/Description	Reset
		There are two cases in which the Interrupt is generated. The first is when a data transfer is completed as specified by data length (After the last data has been read to the Host System). The second is when data has stopped at the block gap and completed the data transfer by setting the Stop At Block Gap Request in the Block Gap Control Register (After valid data has been read to the Host System). **Write Transaction:** This bit is set at the falling edge of the DAT Line Active Status. There are two cases in which the Interrupt is generated. The first is when the last data is written to the card as specified by data length and Busy signal is released. The second is when data transfers are stopped at the block gap by setting Stop At Block Gap Request in the Block Gap Control Register and data transfers completed. (After valid data is written to the SD card and the busy signal is released). Note: Transfer Complete has higher priority than Data Time-out Error. If both bits are set to 1, the data transfer can be considered complete. Note: While performing tuning procedure (Execute Tuning is set to 1), Transfer Complete is not set to 1. 0: No Data Transfer Complete 1: Data Transfer Complete	
0	R/W	CommandComplete This bit is set when we get the end bit of the command response (Except Auto CMD12 and Auto CMD23) Note: Command Time-out Error has higher priority than Command Complete. If both are set to 1, it can be considered that the response was not received correctly. 0: No Command Complete 1: Command Complete	0x0

Table 521: SDEMMC_NORMAL_ITNR_STATUS_EN_REG (0x40011034)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	VendorSpecificErrorStatusEnb Reserved	0x0
28	R/W	TargetResponseErrorHostErrorStatusEnb 0: Masked 1: Enabled	0x0
27	R/W	- Reserved	0x0
26	R/W	TuningErrorStatusEnb 0: Masked 1: Enabled	0x0
25	R/W	ADMAErrorStatusEnb 0: Masked 1: Enabled	0x0
24	R/W	AutoCMD12ErrorStatusEnb 0: Masked 1: Enabled	0x0
23	R/W	CurrentLimitErrorStatusEnb	0x0

Bit	Mode	Symbol/Description	Reset
		0: Masked 1: Enabled	
22	R/W	DataEndBitErrorStatusEnb 0: Masked 1: Enabled	0x0
21	R/W	DataCRCErrorStatusEnb 0: Masked 1: Enabled	0x0
20	R/W	DataTimeoutErrorStatusEnb 0: Masked 1: Enabled	0x0
19	R/W	CommandIndexErrorStatusEnb 0: Masked 1: Enabled	0x0
18	R/W	CommandEndBitErrorStatusEnb 0: Masked 1: Enabled	0x0
17	R/W	CommandCRCErrorStatusEnb 0: Masked 1: Enabled	0x0
16	R/W	CommandTimeoutErrorStatusEnb 0: Masked 1: Enabled	0x0
15	R	FixedTo0 The HC controls error Interrupts using the Error Interrupt Status Enable register.	0x0
14	R/W	BootTerminateInterruptStatusEnb 0: Masked 1: Enabled	0x0
13	R/W	BootAckRcvStatusEnb 0: Masked 1: Enabled	0x0
12	R/W	ReTuningEventStatusEnable 0: Masked 1: Enabled	0x0
11	R/W	INT_CStatusEnb If this bit is set to 0, the Host Controller clears the interrupt request to the System. The Host Driver may clear this bit before servicing the INT_C and may set this bit again after all interrupt requests to INT_C pin are cleared to prevent inadvertent interrupts.	0x0
10	R/W	INT_BStatusEnb If this bit is set to 0, the Host Controller clears the interrupt request to the System. The Host Driver may clear this bit before servicing the INT_B and may set this bit again after all interrupt requests to INT_B pin are cleared to prevent inadvertent interrupts.	0x0

Bit	Mode	Symbol/Description	Reset
9	R/W	INT_AStatusEnb If this bit is set to 0, the Host Controller clears the interrupt request to the System. The Host Driver may clear this bit before servicing the INT_A and may set this bit again after all interrupt requests to INT_A pin are cleared to prevent inadvertent interrupts.	0x0
8	R/W	CardInterruptStatusEnb If this bit is set to 0, the HC clears Interrupt request to the System. The Card Interrupt detection is stopped when this bit is cleared and restarted when this bit is set to 1. The HD may clear the Card Interrupt Status Enable before servicing the Card Interrupt and may set this bit again after all Interrupt requests from the card are cleared to prevent inadvertent Interrupts. 0: Masked 1: Enabled	0x0
7	R/W	CardRemovalStatusEnb 0: Masked 1: Enabled	0x0
6	R/W	CardInsertionStatusEnb 0: Masked 1: Enabled	0x0
5	R/W	BufferReadReadyStatusEnb 0: Masked 1: Enabled	0x0
4	R/W	BufferWriteReadyStatusEnb 0: Masked 1: Enabled	0x0
3	R/W	DMAInterruptStatusEnb 0: Masked 1: Enabled	0x0
2	R/W	BlockGapEventStatusEnb 0: Masked 1: Enabled	0x0
1	R/W	TransferCompleteStatusEnb 0: Masked 1: Enabled	0x0
0	R/W	CommandCompleteStatusEnb 0: Masked 1: Enabled	0x0

Table 522: SDEMMC_NORMAL_ITNR_SIGNAL_EN_REG (0x40011038)

Bit	Mode	Symbol/Description	Reset
31:29	R/W	VendorSpecificErrorSignalEnb Reserved	0x0
28	R/W	TargetResponseErrorSignalEnb 0: Masked 1: Enabled	0x0

Bit	Mode	Symbol/Description	Reset
27	R/W	- Reserved	0x0
26	R/W	TuningErrorSignalEnb 0: Masked 1: Enabled	0x0
25	R/W	ADMAErrorSignalEnb 0: Masked 1: Enabled	0x0
24	R/W	AutoCMDErrorSignalEnb 0: Masked 1: Enabled	0x0
23	R/W	CurrentLimitErrorSignalEnb 0: Masked 1: Enabled	0x0
22	R/W	DataEndBitErrorSignalEnb 0: Masked 1: Enabled	0x0
21	R/W	DataCRCErrorSignalEnb 0: Masked 1: Enabled	0x0
20	R/W	DataTimeoutErrorSignalEnb 0: Masked 1: Enabled	0x0
19	R/W	CommandIndexErrorSignalEnb 0: Masked 1: Enabled	0x0
18	R/W	CommandEndBitErrorSignalEnb 0: Masked 1: Enabled	0x0
17	R/W	CommandCRCErrorSignalEnb 0: Masked 1: Enabled	0x0
16	R/W	CommandTimeoutErrorSignalEnb 0: Masked 1: Enabled	0x0
15	R	FixedTo0 The HD controls error Interrupts using the Error Interrupt Signal Enable register.	0x0
14	R/W	BootTerminateInterruptSignalEnable 0: Masked 1: Enabled	0x0
13	R/W	BootAckRcvSignalEnable 0: Masked 1: Enabled	0x0

Bit	Mode	Symbol/Description	Reset
12	R/W	ReTuningEventSignalEnable 0: Masked 1: Enabled	0x0
11	R/W	INT_CSignalEnb 0: Masked 1: Enabled	0x0
10	R/W	INT_BSignalEnb 0: Masked 1: Enabled	0x0
9	R/W	INT_ASignalEnb 0: Masked 1: Enabled	0x0
8	R/W	CardInterruptSignalEnb 0: Masked 1: Enabled	0x0
7	R/W	CardRemovalSignalEnb 0: Masked 1: Enabled	0x0
6	R/W	CardInsertionSignalEnb 0: Masked 1: Enabled	0x0
5	R/W	BufferReadReadySignalEnb 0: Masked 1: Enabled	0x0
4	R/W	BufferWriteReadySignalEnb 0: Masked 1: Enabled	0x0
3	R/W	DMAInterruptSignalEnb 0: Masked 1: Enabled	0x0
2	R/W	BlockGapEventSignalEnb 0: Masked 1: Enabled	0x0
1	R/W	TransferCompleteSignalEnb 0: Masked 1: Enabled	0x0
0	R/W	CommandCompleteSignalEnb 0: Masked 1: Enabled	0x0

Table 523: SDEMMC_AUTOCMDERRORSTATUS_REG (0x4001103C)

Bit	Mode	Symbol/Description	Reset
31	R/W	PresetValueEnable Host Controller Version 3.00 supports this bit. As the operating SDCLK frequency and I/O driver strength depend on the Host System implementation, it is difficult to determine these parameters in the Standard Host Driver. When Preset Value Enable is set to automatic. This bit enables the functions defined in the Preset Value registers. 1: Automatic Selection by Preset Value are Enabled 0: SDCLK and Driver Strength are controlled by Host Driver If this bit is set to 0, SDCLK Frequency Select, Clock Generator Select in the Clock Control register and Driver Strength Select in Host Control 2 register are set by Host Driver. If this bit is set to 1, SDCLK Frequency Select, Clock Generator Select in the Clock Control register and Driver Strength Select in Host Control 2 register are set by Host Controller as specified in the Preset Value registers.	0x0
30	R/W	AsynchronousInterruptEnable This bit can be set to 1 if a card support asynchronous interrupt and Asynchronous Interrupt Support is set to 1 in the Capabilities register. Asynchronous interrupt is effective when DAT[1] interrupt is used in 4-bit SD mode (and zero is set to Interrupt Pin Select in the Shared Bus Control register). If this bit is set to 1, the Host Driver can stop the SDCLK during asynchronous interrupt period to save power. During this period, the Host Controller continues to deliver Card Interrupt to the host when it is asserted by the Card. 1: Enabled 0: Disabled	0x0
29:24	R/W	- Reserved	0x0
23	R/W	SamplingClockSelect This bit is set by tuning procedure when Execute Tuning is cleared. Writing 1 to this bit is meaningless and ignored. Setting 1 means that tuning is completed successfully and setting 0 means that tuning is failed. Host Controller uses this bit to select sampling clock to receive CMD and DAT. This bit is cleared by writing 0. Change of this bit is not allowed while the Host Controller is receiving response or a read data block. 1: Tuned clock is used to sample data 0: Fixed clock is used to sample data	0x0
22	R/W	ExecuteTuning This bit is set to 1 to start tuning procedure and automatically cleared when tuning procedure is completed. The result of tuning is indicated to Sampling Clock Select. Tuning procedure is aborted by writing 0 for more detail about tuning procedure. 1: Execute Tuning, 0: Not Tuned or Tuning Completed	0x0
21:20	R/W	DriveStrengthSelect Host Controller output driver in 1.8V signaling is selected by this bit. In 3.3V signaling, this field is not effective. This field can be set depends on Driver Type A, C and D support bits in the Capabilities register. This bit depends on setting of Preset Value Enable. If Preset Value Enable = 0, this field is set by Host Driver. If Preset Value Enable = 1, this field is automatically set by a value specified in the one of Preset Value registers. 00b: Driver Type B is Selected (Default) 01b: Driver Type A is Selected	0x0

Bit	Mode	Symbol/Description	Reset
		10b: Driver Type C is Selected 11b: Driver Type D is Selected	
19	R/W	SignalingEnable1V8 This bit controls voltage regulator for I/O cell. 3.3 V is supplied to the card regardless of signaling voltage. Setting this bit from 0 to 1 starts changing signal voltage from 3.3 V to 1.8 V. 1.8 V regulator output is stable within 5 ms. Host Controller clears this bit if switching to 1.8 V signaling fails. Clearing this bit from 1 to 0 starts changing signal voltage from 1.8 V to 3.3 V. 3.3 V regulator output is stable within 5 ms. Host Driver can set this bit to 1 when Host Controller supports 1.8 V signaling (One of support bits is set to 1: SDR50, SDR104, or DDR50 in the Capabilities register) and the card or device supports UHS-I 1: 1.8 V Signaling 0: 3.3 V Signaling	0x0
18:16	R/W	UHSModeSelect This field is used to select one of UHS-I modes and effective when 1.8 V Signaling Enable is set to 1. If Preset Value Enable in the Host Control 2 register is set to 1, Host Controller sets SDCLK Frequency Select, Clock Generator Select in the Clock Control register and Driver Strength Select according to Preset Value registers. In this case, one of preset value registers is selected by this field. Host Driver needs to reset SD Clock Enable before changing this field to avoid generating clock glitch. After setting this field, Host Driver sets SD Clock Enable again. 000b: SDR12 001b: SDR25 010b: SDR50 011b: SDR104 100b: DDR50 101b ~ 111b: Reserved When SDR50, SDR104 or DDR50 is selected for SDIO card, interrupt detection at the block gap is not used. Read Wait timing is changed for these modes. See the SDIO Specification Version 3.00 for more details.	0x0
15:8	R/W	- Reserved	0x0
7	R	CommandNotIssuedByAutoCMD12Error Setting this bit to 1 means CMD_wo_DAT is not executed due to an Auto CMD12 error (D04 - D01) in this register. This bit is set to 0 when Auto CMD Error is generated by Auto CMD23 0: No Error 1: Not Issued	0x0
6:5	R/W	- Reserved	0x0
4	R	AutoCMDIndexError Occurs if the Command Index error occurs in response to a command. 0: No Error 1: Error	0x0
3	R	AutoCMDEndBitError	0x0

Bit	Mode	Symbol/Description	Reset
		Occurs when detecting that the end bit of command response is 0. 0: No Error 1: End Bit Error Generated	
2	R	AutoCMDCRCError Occurs when detecting a CRC error in the command response. 0: No Error 1: CRC Error Generated	0x0
1	R	AutoCMDTimeoutError Occurs if the no response is returned within 64 SDCLK cycles from the end bit of the command. If this bit is set to 1, the other error status bits (D04 - D02) are meaningless. 0: No Error 1: Timeout	0x0
0	R	AutoCMD12NotExecuted If memory multiple block data transfer is not started due to command error, this bit is not set because it is not necessary to issue Auto CMD12. Setting this bit to 1 means the HC cannot issue Auto CMD12 to stop memory multiple block transfer due to some error. If this bit is set to 1, other error status bits (D04 - D01) are meaningless. This bit is set to 0 when Auto CMD Error is generated by Auto CMD23 0: Executed 1: Not Executed	0x0

Table 524: SDEMMC_CAPABILITIES0_REG (0x40011040)

Bit	Mode	Symbol/Description	Reset
31:30	R/W	SlotType This field indicates usage of a slot by a specific Host System. (A host controller register set is defined per slot.) Embedded slot for one device (01b) means that only one non-removable device is connected to a SD bus slot. Shared Bus Slot (10b) can be set if Host Controller supports Shared Bus Control register. The Standard Host Driver controls only a removable card or one embedded device is connected to a SD bus slot. If a slot is configured for shared bus (10b), the Standard Host Driver does not control embedded devices connected to a shared bus. Shared bus slot is controlled by a specific host driver developed by a Host System. 00b: Removable Card Slot 01b: Embedded Slot for One Device 10b: Shared Bus Slot 11b: Reserved	0x0
29	R/W	AsynchronousInterruptSupport See SDIO Specification Version 3.00 about asynchronous interrupt. 1: Asynchronous Interrupt Supported 0: Asynchronous Interrupt Not Supported	0x0
28	R/W	SystemBusSupport64bit 1: Supports 64 bit system address 0: Does not support 64 bit system address	0x0
27	R/W	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
26	R/W	VoltageSupport1V8 0: 1.8 V Not Supported 1: 1.8 V Supported	0x0
25	R/W	VoltageSupport3V0 0: 3.0 V Not Supported 1: 3.0 V Supported	0x0
24	R/W	VoltageSupport3V3 0: 3.3 V Not Supported 1: 3.3 V Supported	0x0
23	R/W	SuspendResumeSupport This bit indicates whether the HC supports Suspend/Resume functionality. If this bit is 0, the Suspend and Resume mechanism are not supported and the HD shall not issue either Suspend/Resume commands. 0: Not Supported 1: Supported	0x0
22	R/W	SDMASupport This bit indicates whether the HC is capable of using DMA to transfer data between system memory and the HC directly. 0: SDMA Not Supported 1: SDMA Supported	0x0
21	R/W	HighSpeedSupport This bit indicates whether the HC and the Host System support High Speed mode and they can supply SD Clock frequency from 25 MHz to 50 MHz (for SD)/ 20 MHz to 52 MHz (for MMC). 0: High Speed Not Supported 1: High Speed Supported	0x0
20	R/W	- Reserved	0x0
19	R/W	ADMA2Support 1: ADMA2 support. 0: ADMA2 not support	0x0
18	R/W	ExtendedMediaBusSupport This bit indicates whether the Host Controller is capable of using 8-bit bus width mode. This bit is not effective when Slot Type is set to 10b. In this case, see Bus Width Preset in the Shared Bus resister. 1: Extended Media Bus Supported 0: Extended Media Bus not Supported	0x0
17:16	R/W	MaxBlockLength This value indicates the maximum block size that the HD can read and write to the buffer in the HC. The buffer transfers this block size without wait cycles. Three sizes can be defined as indicated below. 00: 512 byte 01: 1024 byte 10: 2048 byte	0x0

Bit	Mode	Symbol/Description	Reset
		11: 4096 byte	
15:8	R/W	BaseClockFrequencyForSDClock (1) 6-bit Base Clock Frequency This mode is supported by the Host Controller Version 1.00 and 2.00. Upper 2-bit is not effective and always 0. Unit values are 1 MHz. The supported clock range is 10 MHz to 63 MHz. 11xx xxxxb: Not supported 0011 1111b: 63 MHz 0000 0010b: 2 MHz 0000 0001b: 1 MHz 0000 0000b: Get information via another method (2) 8-bit Base Clock Frequency This mode is supported by the Host Controller Version 3.00. Unit values are 1 MHz. The supported clock range is 10 MHz to 255 MHz. FFh: 255 MHz 02h: 2 MHz 01h: 1 MHz 00h: Get information via another method If the real frequency is 16.5 MHz, the lager value is set 0001 0001b (17 MHz) because the Host Driver use this value to calculate the clock divider value (see the SDCLK Frequency Select in the Clock Control register.) and it does not exceed upper limit of the SD Clock frequency. If these bits are all 0, the Host System has to get information via another method.	0x0
7	R/W	TimeoutClockUnit This bit shows the unit of base clock frequency used to detect Data Timeout Error. 0: kHz 1: MHz	0x0
6	R/W	- Reserved	0x0
5:0	R/W	TimeoutClockFrequency This bit shows the base clock frequency used to detect Data Timeout Error. Not 0: 1 kHz to 63 kHz or 1 MHz to 63 MHz 000000b: Get Information via another method.	0x0

Table 525: SDEMMC_CAPABILITIES1_REG (0x40011044)

Bit	Mode	Symbol/Description	Reset
31:26	R/W	- Reserved	0x0
25	R/W	SPI_BLOCK_MODE SPI Block mode 0: Not Supported 1: Supported	0x0
24	R/W	SPI_MODE Spi mode 0: Not Supported	0x0

Bit	Mode	Symbol/Description	Reset
		1: Supported	
23:16	R/W	ClockMultiplier This field indicates clock multiplier value of programmable clock generator. See Clock Control register. Setting 00h means that Host Controller does not support programmable clock generator. FFh: Clock Multiplier M = 256 .... 02h: Clock Multiplier M = 3 01h: Clock Multiplier M = 2 00h: Clock Multiplier is Not Supported	0x0
15:14	R/W	ReTuningModes This field defines the re-tuning capability of a Host Controller and how to manage the data transfer length and a Re-Tuning Timer by the Host Driver 00: Mode1 01: Mode2 10: Mode3 11: Reserved There are two re-tuning timings: Re-Tuning Request and expiration of a Re-Tuning Timer. By receiving either timing, the Host Driver executes the re-tuning procedure just before a next command issue.	0x0
13	R/W	UseTuningforSDR50 If this bit is set to 1, this Host Controller requires tuning to operate SDR50. (Tuning is always required to operate SDR104.) 1: SDR50 requires tuning 0: SDR50 does not require tuning	0x0
12	R/W	- Reserved	0x0
11:8	R/W	TimerCountforReTuning This field indicates an initial value of the Re-Tuning Timer for Re-Tuning Mode 1 to 3. 0h: Get information via other source 1h: 1 s 2h: 2 s 3h: 4 s 4h: 8 s -- n: $2^{(n-1)}$ s -- Bh: 1024 s Ch - Fh: Reserved	0x0
7	R/W	- Reserved	0x0
6	R/W	DriverTypeDSupport This bit indicates support of Driver Type D for 1.8 Signaling. 1: Driver Type D is Supported 0: Driver Type D is Not Supported	0x0

Bit	Mode	Symbol/Description	Reset
5	R/W	DriverTypeCSupport This bit indicates support of Driver Type C for 1.8 Signaling. 1: Driver Type C is Supported 0: Driver Type C is Not Supported	0x0
4	R/W	DriverTypeASupport This bit indicates support of Driver Type A for 1.8 Signaling. 1: Driver Type A is Supported 0: Driver Type A is Not Supported	0x0
2	R/W	DDR50Support 1: DDR50 is Supported 0: DDR50 is Not Supported	0x0
1	R/W	SDR104Support SDR104 requires tuning. 1: SDR104 is Supported 0: SDR104 is Not Supported	0x0
0	R/W	SDR50Support If SDR104 is supported, this bit must be set to 1. Bit 40 indicates whether SDR50 requires tuning or not. 1: SDR50 is Supported 0: SDR50 is Not Supported	0x0

Table 526: SDEMMC_MAX_CUR_CAPABILITIES_REG (0x40011048)

Bit	Mode	Symbol/Description	Reset
31:24	R/W	- Reserved	0x0
23:16	R	MaximumCurrentFor1V8 Maximum Current for 1.8V	0x0
15:8	R	MaximumCurrentFor3V0 Maximum Current for 3.0V	0x0
7:0	R	MaximumCurrentFor3V3 Maximum Current for 3.3V	0x0

Table 527: SDEMMC_FORCE_EVENET_ERRSTAT_REG (0x40011050)

Bit	Mode	Symbol/Description	Reset
31:28	W	ForceEventForVendorSpecificError Reserved	0x0
27:26	R/W	- Reserved	0x0
25	W	ForceEventForADMAError Force Event for ADMA Error 1: Interrupt is generated 0: No interrupt	0x0

Bit	Mode	Symbol/Description	Reset
24	W	ForceEventForAutoCMDError Force Event for Auto CMD Error 1: Interrupt is generated 0: No interrupt	0x0
23	W	ForceEventForCurrentLimitError Force Event for Current Limit Error 1: Interrupt is generated 0: No interrupt	0x0
22	W	ForceEventForDataEndBitError Force Event for Data End Bit Error 1: Interrupt is generated 0: No interrupt	0x0
21	W	ForceEventForDataCRCError Force Event for Data CRC Error 1: Interrupt is generated 0: No interrupt	0x0
20	W	ForceEventForDataTimeoutError Force Event for Data Timeout Error 1: Interrupt is generated 0: No interrupt	0x0
19	W	ForceEventForCommandIndexError Force Event for Command Index Error 1: Interrupt is generated 0: No interrupt	0x0
18	W	ForceEventForCommandEndBitError Force Event for Command End Bit Error 1: Interrupt is generated 0: No interrupt	0x0
17	W	ForceEventForCommandCRCError Force Event for Command CRC Error 1: Interrupt is generated 0: No interrupt	0x0
16	W	ForceEventForCommandTimeoutError Force Event for Command Timeout Error 1: Interrupt is generated 0: No interrupt	0x0
15:8	R/W	- Reserved	0x0
7	W	ForceEventForCommandNotIssuedByAutoCMD12Error 1: Interrupt is generated 0: No interrupt	0x0
6:5	R/W	- Reserved	0x0
4	W	ForceEventForAutoCMDIndex	0x0

Bit	Mode	Symbol/Description	Reset
		1: Interrupt is generated 0: No interrupt	
3	W	ForceEventForAutoCMDEndBit 1: Interrupt is generated 0: No interrupt	0x0
2	W	ForceEventForAutoCMDCRC 1: Interrupt is generated 0: No interrupt	0x0
1	W	ForceEventForAutoCMDTimeout 1: Interrupt is generated 0: No interrupt	0x0
0	W	ForceEventForAutoCMD12NotExec 1: Interrupt is generated 0: No interrupt	0x0

Table 528: SDEMMC_ADMAERRORSTATUS_REG (0x40011054)

Bit	Mode	Symbol/Description	Reset
31:3	R/W	- Reserved	0x0
2	R	LengthMismatchError This error occurs in the following two cases. While Block Count Enable being set, the total data length specified by the Descriptor table is different from that specified by the Block Count and Block Length. Total data length can not be divided by the block length. 1: Error 0: No error	0x0
1:0	R	ADMAErrorState This field indicates the state of ADMA when error is occurred during ADMA data transfer. This field never indicates 10 because ADMA never stops in this state. D01 - D00: ADMA Error State when error occurred Contents of SYS_SDR register 00: ST_STOP (Stop DMA) Points to the next of the error descriptor 01: ST_FDS (Fetch Descriptor) Points to the error descriptor 10: Never set this state (Not used) 11: ST_TFR (Transfer Data) Points to the next of the error descriptor	0x0

Table 529: SDEMMC_ADMA_SYS_ADDR_LOWER_REG (0x40011058)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	ADMASystemAddressLower This register holds byte address of executing command of the Descriptor table. 32-bit Address Descriptor uses lower 32- bit of this register. At the start of ADMA, the Host Driver sets start address of the Descriptor table. The ADMA increments this register address, which points to next line, when every fetching a Descriptor line. When the ADMA Error Interrupt is generated, this register holds valid Descriptor address depending on the ADMA state. The Host Driver programs	0x0

Bit	Mode	Symbol/Description	Reset
		Descriptor Table on 32-bit boundary and set 32-bit boundary address to this register. ADMA2 ignores lower 2-bit of this register and assumes it to be 00b. 32-bit Address ADMA Register Value 32-bit System Address xxxxxxxx 00000000h 00000000h xxxxxxxx 00000004h 00000004h xxxxxxxx FFFFFFFCh FFFFFFFCh	

Table 530: SDEMMC_ADMA_SYS_ADDR_UPPER_REG (0x4001105C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	ADMASystemAddressUpper This register holds byte address of executing command of the Descriptor table. 32-bit Address Descriptor uses lower 32-bit of this register. At the start of ADMA, the Host Driver sets start address of the Descriptor table. The ADMA increments this register address, which points to next line, when every fetching a Descriptor line. When the ADMA Error Interrupt is generated, this register holds valid Descriptor address depending on the ADMA state. The Host Driver programs Descriptor Table on 32-bit boundary and set 32-bit boundary address to this register. ADMA2 ignores lower 2-bit of this register and assumes it to be 00b. 32-bit Address ADMA Register Value 32-bit System Address xxxxxxxx 00000000h 00000000h xxxxxxxx 00000004h 00000004h xxxxxxxx FFFFFFFCh FFFFFFFCh	0x0

Table 531: SDEMMC_PRESETVALUE0_REG (0x40011060)

Bit	Mode	Symbol/Description	Reset
31:30	R	DefSpdDriveStrengthSelectValue Preset Values for Default Speed: Signal Voltage 3.3 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0
29:27	R/W	- Reserved	0x0
26	R	DefSpdClockGeneratorSelectValue Preset Values for Default Speed: Signal Voltage 3.3 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
25:16	R	DefSpdSDCLKFrequenceSelectValue Preset Values for Default Speed: Signal Voltage 3.3 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0

Bit	Mode	Symbol/Description	Reset
15:14	R	InitDriveStrengthSelectValue Preset Values for Initialization: Signal Voltage 3.3 V or 1.8 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0
13:11	R/W	- Reserved	0x0
10	R	InitClockGeneratorSelectValue Preset Values for Initialization: Signal Voltage 3.3 V or 1.8 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
9:0	R	InitSDCLKFrequenceSelectValue Preset Values for Initialization: Signal Voltage 3.3 V or 1.8 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0

Table 532: SDEMMC_PRESETVALUE1_REG (0x40011064)

Bit	Mode	Symbol/Description	Reset
31:30	R	SDR12DriveStrengthSelectValue Preset Values for SDR12: Signal Voltage 1.8 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0
29:27	R/W	- Reserved	0x0
26	R	SDR12ClockGeneratorSelectValue Preset Values for SDR12: Signal Voltage 1.8 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
25:16	R	SDR12SDCLKFrequenceSelectValue Preset Values for SDR12: Signal Voltage 1.8 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0
15:14	R	HighSpdDriveStrengthSelectValue Preset Values for High Speed: Signal Voltage 3.3 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling.	0x0

Bit	Mode	Symbol/Description	Reset
		11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	
13:11	R/W	- Reserved	0x0
10	R	HighSpdClockGeneratorSelectValue Preset Values for High Speed: Signal Voltage 3.3 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
9:0	R	HighSpdSDCLKFrequenceSelectValue Preset Values for High Speed: Signal Voltage 3.3 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0

Table 533: SDEMMC_PRESETVALUE2_REG (0x40011068)

Bit	Mode	Symbol/Description	Reset
31:30	R	SDR50DriveStrengthSelectValue Preset Values for SDR50: Signal Voltage 1.8 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0
29:27	R/W	- Reserved	0x0
26	R	SDR50ClockGeneratorSelectValue Preset Values for SDR50: Signal Voltage 1.8 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
25:16	R	SDR50SDCLKFrequenceSelectValue Preset Values for SDR50: Signal Voltage 1.8 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0
15:14	R	SDR25DriveStrengthSelectValue Preset Values for SDR25: Signal Voltage 1.8 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0

Bit	Mode	Symbol/Description	Reset
13:11	R/W	- Reserved	0x0
10	R	SDR25ClockGeneratorSelectValue Preset Values for SDR25: Signal Voltage 1.8 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
9:0	R	SDR25SDCLKFrequenceSelectValue Preset Values for SDR25: Signal Voltage 1.8 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0

Table 534: SDEMMC_PRESETVALUE3_REG (0x4001106C)

Bit	Mode	Symbol/Description	Reset
31:30	R	DDR50DriveStrengthSelectValue Preset Values for DDR50: Signal Voltage 1.8 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0
29:27	R/W	- Reserved	0x0
26	R	DDR50ClockGeneratorSelectValue Preset Values for DDR50: Signal Voltage 1.8 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	0x0
25:16	R	DDR50SDCLKFrequenceSelectValue Preset Values for DDR50: Signal Voltage 1.8 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0
15:14	R	SDR104DriveStrengthSelectValue Preset Values for SDR104: Signal Voltage 1.8 V Driver Strength is supported by 1.8 V signaling bus speed modes. This field is meaningless for 3.3 V signaling. 11b: Driver Type D is Selected 10b: Driver Type C is Selected 01b: Driver Type A is Selected 00b: Driver Type B is Selected	0x0
13:11	R/W	- Reserved	0x0
10	R	SDR104ClockGeneratorSelectValue	0x0

Bit	Mode	Symbol/Description	Reset
		Preset Values for SDR104: Signal Voltage 1.8 V This bit is effective when Host Controller supports programmable clock generator. 1: Programmable Clock Generator 0: Host Controller Ver2.00 Compatible Clock Generator	
9:0	R	SDR104SDCLKFrequenceSelectValue Preset Values for SDR104: Signal Voltage 1.8 V 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.	0x0

Table 535: SDEMMC_BOOTTIMEOUT_CTRL_REG (0x40011070)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	BootDataTimeoutCounterValue This value determines the interval by which DAT line time-outs are detected during boot operation for eMMC card. The value is in number of sd clock.	0x0

Table 536: SDEMMC_DEBUGSELECTION_REG (0x40011074)

Bit	Mode	Symbol/Description	Reset
31:1	R/W	- Reserved	0x0
0	R/W	DebugSel	0x0

Table 537: SDEMMC_VENDOR_REG (0x40011078)

Bit	Mode	Symbol/Description	Reset
31:2	R/W	- Reserved	0x0
1	R/W	Delay_cmdin_datin_dis Chicken bit added to enable/disable the rtl fix made to delay the sampling of cmd_in and data_in.	0x0
0	R/W	AutoGateSDCLK If this bit is 0, SD_CLK to card is not gated automatically, when there is no transfer. If this bit set to 1, SD_CLK to card is gated automatically,when there is no transfer.	0x0

Table 538: SDEMMC_SHAREDBUS_CTRL_REG (0x400110E0)

Bit	Mode	Symbol/Description	Reset
31	R/W	- Reserved	0x0
30:24	R/W	BackEndPowerControl	0x0
23	R/W	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
22:20	R/W	InterruptPinSelect	0x0
19	R/W	-	0x0
		Reserved	
18:16	R/W	ClockPinSelect	0x0
15	R/W	-	0x0
		Reserved	
14:8	R	BusWidthPreset	0x0
7:6	R/W	-	0x0
		Reserved	
5:4	R	NumberOfInterruptInputPins	0x0
3	R/W	-	0x0
		Reserved	
2:0	R	NumberOfClockPins	0x0

Table 539: SDEMMC_SLOT_ITNR_STATUS_REG (0x400110FC)

Bit	Mode	Symbol/Description	Reset
31:24	R	VendorVersionNumber	0x10
		The Vendor Version Number is set to 0x10 (1.0)	
23:16	R	SpecificationVersionNumber	0x2
		The Host Controller Version Number is set to 0x02 (SD Host Specification Version 3.00).	
15:1	R/W	-	0x0
		Reserved	
0	R	InterruptSignalForEachSlot0	0x0
		This status bit indicates the OR of Interrupt signal and Wake-up signal for slot	

Table 540: SDEMMC_GLB_ITNR_STATUS_REG (0x40011400)

Bit	Mode	Symbol/Description	Reset
31:3	R/W	-	0x0
		Reserved	
2	R/W	IntSrcSD	0x0
1	R/W	IntSrcXD	0x0
0	R/W	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	

Table 541: SDEMMC_GLB_ITNR_STATUS_EN_REG (0x40011404)

Bit	Mode	Symbol/Description	Reset
31:3	R/W	- Reserved	0x0
2	R/W	IntSrcSD	0x0
1	R/W	IntSrcXD	0x0
0	R/W	- Reserved	0x0

Table 542: SDEMMC_GLB_ITNR_SIGNAL_EN_REG (0x40011408)

Bit	Mode	Symbol/Description	Reset
31:3	R/W	- Reserved	0x0
2	R/W	IntSrcSD	0x0
1	R/W	IntSrcXD	0x0
0	R/W	- Reserved	0x0

10.19 SDIO Registers

Table 543: Register map SDIO

Address	Register	Description
0x40010000	SDIO_CLOCK_WAKEUP_REG	Clock Wake-up
0x40010004	SDIO_CCCR_REG	CCCR
0x40010008	SDIO_ADMA_SYSTEM_ADDRESS_REG	This register holds descriptor Pointer of the Descriptor table. At the start of ADMA Arm programs start address of the Descriptor table.
0x4001000c	SDIO_CARDRDY_REG	Card Ready
0x40010010	SDIO_FUNRDY_REG	Function Ready
0x40010014	SDIO_AHB_FN0_INT_ENABLE_REG	Function 0 interrupt enable
0x40010018	SDIO_AHB_FN0_INT_REG	Function 0 interrupt
0x4001001c	SDIO_SOFT_RST_AHB_REG	Soft Reset
0x40010020	SDIO_GLB_INT_ENA_REG	Global Interrupt Enable

Address	Register	Description
0x40010024	SDIO_GLB_INT_STS_REG	Global Interrupt Enable
0x40010028	SDIO_CSA_POINTER_REG	CSA pointer
0x4001002c	SDIO_IO_ACC_MODE_REG	IO access mode
0x40010030	SDIO_LAST_FETCH_ADDR_REG	This register holds descriptor Pointer of the Descriptor table. At the start of ADMA Arm programs start address of the Descriptor table.
0x40010034	SDIO_UHS_SUPPORT_REG	UHS Support
0x40010038	SDIO_CLK_DELAY_TIMER_REG	This value used as wait time for the Controller before releasing cmd, data line for cmd11 voltage switch.
0x4001003c	SDIO_POWER_CONTROL_REG	Power Control
0x40010040	SDIO_POWER_STATE_REG	Power State
0x40010104	SDIO_OCR_REG	OCR
0x40010108	SDIO_INTERRUPT_REG	Interrupt Identification
0x4001010c	SDIO_AHB_TRANSCOUNT_REG	AHB transaction counter
0x40010110	SDIO_AHB_SDIOTRANSCOUNT_REG	AHB SDIO transaction counter
0x40010114	SDIO_AHB_FN1_INT_REG	Function 1 interrupt
0x40010118	SDIO_AHB_FN1_INT_ENABLE_REG	Function 1 interrupt enable
0x4001011c	SDIO_FBR_REG	FBR
0x40010120	SDIO_IOR_REG	IOR
0x40010124	SDIO_SD_HOST_GP_REG	SD Host General Purpose Register
0x40010128	SDIO_ARM_GP_REG	Arm General Purpose Register
0x4001012c	SDIO_RDDATRDY_REG	Read Data Ready
0x40010130	SDIO_BLKSIZE_REG	Block Size
0x40010134	SDIO_ARGUMENT_REG	Argument
0x40010138	SDIO_WRBLKCNT_REG	Write Block Count
0x4001013c	SDIO_RDBLKCNT_REG	Read Block Count

Table 544: SDIO_CLOCK_WAKEUP_REG (0x40010000)

Bit	Mode	Symbol/Description	Reset
1	R/W	MANUAL_CLK_ENABLE 1 - The ahb_clk_wkup signal is always high indicating to the Arm for keeping the clock active. This is used to allow Manual mode from SD Host.	0x0
0	R/W	AUTO_CLK_ENABLE When this bit is programmed to 1, SDIO_AHB Controller controls the ahb_clk_wkup signal depending on the data transfer activity on the bus automatically.	0x1

Table 545: SDIO_CCCR_REG (0x40010004)

Bit	Mode	Symbol/Description	Reset
27:22	R	CMD_INDEX These 6 bit register contains the command index value for the commands issued on the sd side.	0x0
21	R/W	SHS Support high speed - This flag bit reports the card's ability to operate in high speed mode.	0x1
20	R/W	SMPC This bit tells the host if the card supports master power control. SMPC = 0. The total card current is less than 200 mA, even if all functions are active (IOEx = 1) EMPC, SPS, and EPS are zero. SMPC = 1. The total card current may exceed 200 mA. EMPC, SPS, and EPS are available.	0x1
19	R/W	FOURBLS This bit denotes that the SDIO card is a low speed card and supports 4-bit data transfer.	0x0
18	R/W	LSC If this bit is set, it indicates that the SDIO card is a low speed device. If this bit is cleared the SDIO card is a full speed device.	0x0
17	R/W	S4MI This flag reports the SDIO card's ability to generate interrupts during a 4-bit multi block data transfer.	0x1
16	R	SBS This flag bit reports the card's ability to Support the Suspend/Resume operations at the request of the Host. If this bit is set, all functions except 0 accept a request to suspend operations and esume under host control.	0x1
15	R/W	SRW This flag bit reports the card's ability to Support the Read Wait Control (RWC) operation.	0x1
14	R/W	SMB This flag bit reports the card's ability to execute CMD53 in Block Mode.	0x1
13	R/W	SDC This flag bit reports the card's ability to execute CMD52 while data transfer is in progress.	0x1
12	R/W	SCSI Support Continuous SPI interrupt.	0x1
11:8	R/W	SD_REVISION SD Format Revision. These 4-bits contain the version of the SD Physical specification that this card supports. 00h SD Physical Specification 1.01 01h SD Specification 1.10 Physical 02h SD Physical Specification 2.00 03h SD Physical Specification 3.00	0x2

Bit	Mode	Symbol/Description	Reset
		04h-0Fh Reserved for Future Use.	
7:4	R/W	SDIO_REVISION SDIO Specification Revision number. These 4-bits contain the version of the SDIO specification that this card supports. 00h SDIO Specification Version 1.00 01h SDIO Specification Version 1.10 02h SDIO Specification Version 1.20 (Unreleased) 03h SDIO Specification Version2.00 04h SDIO Specification Version3.00 05h-0Fh Reserved for Future Use.	0x3
3:0	R/W	CCCR_REVISION CCCR Format Version number. These 4-bits contains the version of the CCCR and FBR format that this card supports. 00h CCCR/FBR Version 1.00 CCCR/FBR 01h Version 1.10 02h CCCR/FBR Version 2.00 CCCR/FBR 03h Version 3.00 04h-0Fh Reserved for Future Use.	0x2

Table 546: SDIO_ADMA_SYSTEM_ADDRESS_REG (0x40010008)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	ADMA_SYSTEM_ADDRESS This register holds descriptor Pointer of the Descriptor table. At the start of ADMA Arm programs start address of the Descriptor table.	0x0

Table 547: SDIO_CARDRDY_REG (0x4001000C)

Bit	Mode	Symbol/Description	Reset
0	R	CARD_RDY Card ready bit indicates Power on reset is synchronously deasserted on SD clock domain. This bit is set to 1 after power on reset indicating that the SDIO_AHB Controller is ready to program.	0x0

Table 548: SDIO_FUNRDY_REG (0x40010010)

Bit	Mode	Symbol/Description	Reset
0	R/W	CARD_RDY Arm Processor set this bit to indicate function is ready to operate.	0x0

Table 549: SDIO_AHB_FN0_INT_ENABLE_REG (0x40010014)

Bit	Mode	Symbol/Description	Reset
11	R/W	FN0_ADMA_ERR_EN 1 - fn0_adma_err interrupt is enabled 0 - fn0_adma_err interrupt is disabled	0x0

Bit	Mode	Symbol/Description	Reset
10	R/W	FN0_ADMA_INT_EN 1 - fn0_adma_int interrupt is enabled 0 - fn0_adma_int interrupt is disabled	0x0
9	R/W	FN0_ADMA_END_EN 1 - fn0_adma_end_bit interrupt is enabled 0 - fn0_adma_end_bit interrupt is disabled	0x0
8	R/W	FN0_RD_ERR_EN 1 - fn0 read trn err interrupt is enabled 0 - fn0 read trn err interrupt is disabled	0x0
7	R/W	FN0_RD_OVR_EN 1 - fn0 read trn over interrupt is enabled 0 - fn0 read trn over interrupt is disabled	0x0
6	R/W	FN0_RD_STRT_EN 1 - fn0 read start interrupt is enabled 0 - fn0 read start interrupt is disabled	0x0
5	R/W	FN0_WR_TRN_OVR_EN 1 - fn0 write trn over interrupt is enabled 0 - fn0 write trn over interrupt is disabled	0x0
4	R/W	FN0_WR_STRT_EN 1 - fn0 write start interrupt is enabled 0 - fn0 write start interrupt is disabled	0x0
3	R/W	CMD19_RD_TRN_OVR_EN 1 - cmd19 rd trans over interrupt is enabled 0 - cmd19 rd trans over interrupt is disabled	0x0
2	R/W	CMD19_RD_STRT_EN 1 - cmd19 rd start interrupt is enabled 0 - cmd19 rd start interrupt is disabled	0x0
1	R/W	VOLT_SWITCH_CMD_EN 1 - volt switch cmd interrupt is enabled 0 - volt switch cmd interrupt is disabled	0x0
0	R/W	AHBSOFT_RST_EN Interrupt enable for cmd52 soft reset interrupt.	0x0

Table 550: SDIO_AHB_FN0_INT_REG (0x40010018)

Bit	Mode	Symbol/Description	Reset
11	R/W	FN0_ADMA_ERR This bit is set by Arasan SDIO_AHB bridge when Arm sets valid bit as 0 in the attribute field during the fetch operation.	0x0
10	R/W	FN0_ADMA_INT This bit is set by Arasan SDIO_AHB bridge when int bit is set in the attribute field during fetch operation.	0x0
9	R/W	FN0_ADMA_END	0x0

Bit	Mode	Symbol/Description	Reset
		This bit is set when the adma endbit is set during descriptor fetch indicate end of fetch. Then Arm processor ready for next fetch (with new updated address).	
8	R/W	FN0_RD_ERR This interrupt is issued if SD Host issue abort command during function 0 read transaction.	0x0
7	R/W	FN0_RD_OVR This interrupt corresponds to the interrupt issued by SD DMA after it completes the read operation of a Function0.	0x0
6	R/W	FN0_RD_STRT This bit is set when the SD Host issues CMD53 Fun0 read command (Arm side transaction). This bit is set by SDIO AHB bridge.	0x0
5	R/W	FN0_WR_TRN_OVR This interrupt corresponds to the interrupt issued by SD DMA after it completes the write operation of a Function0 Arm side transaction. Until this bit is cleared, busy signal is asserted in dat0 line, the SD DMA engine is prevented from performing an AHB write access from a new cmd53 write/read. No valid data in the FIFO is overwritten.	0x0
4	R/W	FN0_WR_STRT This bit is set when the SD Host issues CMD53 Fun0 write command (For Arm side transaction). This bit is set by SDIO AHB bridge.	0x0
3	R/W	CMD19_RD_TRN_OVR This bit is set when cmd19 read transaction is over.	0x0
2	R/W	CMD19_RD_STRT This bit is set when SD Host issues cmd19.	0x0
1	R/W	VOLT_SWITCH_CMD This bit is set by Arasan SDIO_AHB bridge when SD Host issues cmd11.	0x0
0	R/W	AHBSOFT_RST This soft reset interrupt is asserted to Arm when host issues cmd52 soft reset for SDIO controller.	0x0

Table 551: SDIO_SOFT_RST_AHB_REG (0x4001001C)

Bit	Mode	Symbol/Description	Reset
0	R/W	AHBSOFT_VALID When Arm receives cmd52_rst interrupt it clears the interrupt and set this bit to 1. After that the controller asserts rstsoft_ahb_n for 1 AHB clk and clear the AHB domain flops.	0x0

Table 552: SDIO_GLB_INT_ENA_REG (0x40010020)

Bit	Mode	Symbol/Description	Reset
1	R/W	FN1_INT_TO_ARM_EN 1 - Interrupt asserted from AHB fn1 interrupt Register enabled.	0x0

Bit	Mode	Symbol/Description	Reset
		0 - Interrupt asserted from AHB fn1 interrupt Register masked.	
0	R/W	FN0_INT_TO_ARM_EN 1 - Interrupt asserted from AHB fn0 interrupt Register enabled. 0 - Interrupt asserted from AHB fn0 interrupt Register masked.	0x0

Table 553: SDIO_GLB_INT_STS_REG (0x40010024)

Bit	Mode	Symbol/Description	Reset
1	R/W	FN1_INT_TO_ARM Interrupt asserted from AHB fn1 interrupt Register	0x0
0	R/W	FN0_INT_TO_ARM Interrupt asserted from AHB fn0 interrupt Register	0x0

Table 554: SDIO_CSA_POINTER_REG (0x40010028)

Bit	Mode	Symbol/Description	Reset
23:0	R	CSA_POINTER CSA pointer updated by SD Host.	0x0

Table 555: SDIO_IO_ACC_MODE_REG (0x4001002C)

Bit	Mode	Symbol/Description	Reset
6	R/W	SAI Support bit of Asynchronous Interrupt. If the card supports asynchronous interrupt in SD 4-bit mode (interrupt can be asserted without SD clock during specified period), this bit is set to 1.	0x0
5	R/W	SDTD This bit indicates support of Driver Type D. Support bit of SDR50 0b: Driver Type D is not supported 1b: Driver Type D is supported	0x0
4	R/W	SDTC This bit indicates support of Driver Type C. Support bit of SDR50 0b: Driver Type C is not 1b: Drive 1b: Driver Type C is supported	0x0
3	R/W	SDTA This bit indicates support of Driver Type A. Support bit of SDR50 0b: Driver Type A is not supported 1b: Driver Type A is supported	0x0
2	R/W	SDDR50 This bit indicates support of DDR50. Support bit of DDR50 0b: DDR50 is not supported 1b: DDR50 is supported	0x0

Bit	Mode	Symbol/Description	Reset
1	R/W	SSDR104 This bit indicates support of SDR104. Support bit of SDR104 0b: SDR104 is not supported 1b: SDR104 is supported	0x0
0	R/W	SSDR50 This bit indicates support of SDR50. Support bit of SDR50 0b: SDR50 is not supported 1b: SDR50 is supported	0x0

Table 556: SDIO_LAST_FETCH_ADDR_REG (0x40010030)

Bit	Mode	Symbol/Description	Reset
31:0	R	LAST_FETCH_ADDR This register holds descriptor Pointer of the Descriptor table. At the start of ADMA Arm programs start address of the Descriptor table.	0x0

Table 557: SDIO_UHS_SUPPORT_REG (0x40010034)

Bit	Mode	Symbol/Description	Reset
4	R/W	SD_CMD_LINE_SWITCHED This bit is set by Arm and it indicates cmd line switched to 1.8 V successfully	0x0
3	R/W	SD_CLK_LINE_SWITCHED This bit is set by Arm and it indicates sd_clk line is switched to 1.8 V successfully	0x0
2	R/W	CARD_VOLT_ACCEPTED This bit is set by Arm and it indicates card switched to 1.8 V successfully	0x0
1	R/W	DDR_DLY_SELECT This bit field indicates delay select value for DDR mode read transaction. 0 - 1.5 ns delay value 1 - 3 ns delay value	0x0
0	R/W	UHS_SUPPORT This bit is set by the Arm to indicate the card supports the UHS mode for SD3.0 support which requires 1.8 V signal. Setting this to one will also make S18A bit to one in R4 Response	0x0

Table 558: SDIO_CLK_DELAY_TIMER_REG (0x40010038)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	CLK_DELAY_TIMER This value used as wait time for the Controller before releasing cmd, data line for cmd11 voltage switch.	0x0

Table 559: SDIO_POWER_CONTROL_REG (0x4001003C)

Bit	Mode	Symbol/Description	Reset
9	R/W	FASTINIT_REQUEST It should write for waking the IP up from power down mode. This indicates that the chip has received a wake-up request and the IP has to retain the register values from the retention RAM.	0x0
8	R/W	POWERDOWN_REQUEST It should write to initiate power down sequence, which the IP should start writing the register values into the RAM	0x0
7:2	-	- Reserved	0x0
1	R	EPS_FUN1 EPS = 0 (default): The function operates in Higher Current Mode The maximum current for the function is given in TPLFE_HP_MAX_PWR_3.3V EPS = 1: The function works in Lower Current Mode The maximum current for the function is given in TPLFE_LP_MAX_PWR_3.3V.	0x0
0	R	EMPC EMPC = 0(default): The total card current should be less than 200 mA EMPC = 1: The total card current may exceed 200 mA.	0x0

Table 560: SDIO_POWER_STATE_REG (0x40010040)

Bit	Mode	Symbol/Description	Reset
3:0	R	PWR_STATE_FN1 If PS[3:0] is set to 0, TPL_CODE CISTPL_FUNCE (22h) extension 01h is used and the card power is controlled by EMPC and EPS (SDIO Version 2.00 Compatible). Power State control is defined by SDIO Version 3.00 and is effective when EMPC is set to 1 and PS[3:0] is set to larger than 0. In this case, a list of card supported power states (current) is described in the Fun1 CIS tuple	0x0

Table 561: SDIO_OCR_REG (0x40010104)

Bit	Mode	Symbol/Description	Reset
23:0	R/W	OCR IO Operational condition register is programmed by the Arm which is used to match with the operating voltage range of the SD Host.	0xFF8000

Table 562: SDIO_INTERRUPT_REG (0x40010108)

Bit	Mode	Symbol/Description	Reset
11	R/W	ACK_TO_SD_HOST_EN 1: Ack to SD Host interrupt is enabled 0: Ack to SD Host interrupt is disabled	0x0
10	R/W	MESSAGE_FROM_ARM_EN 1: Message from Arm interrupt is enabled 0: Message from Arm interrupt is disabled	0x0
9	R/W	READ_ERROR_EN	0x0

Bit	Mode	Symbol/Description	Reset
		1: Read error interrupt is enabled 0: Read error interrupt is disabled	
8	R/W	READ_DATA_READY_INT_EN 1: Read data ready interrupt is enabled 0: Read data ready interrupt is disabled	0x0
7:4	-	- Reserved	0x0
3	R/W	ACK_TO_SD_HOST Acknowledgment to SD Host. If set to 1, it indicates that the Arm has read the message.	0x0
2	R/W	MESSAGE_FROM_ARM Arm Message Interrupt If set to 1, it indicates that the Arm has programmed the Arm General Purpose Register (Message from Arm to SD Host). On receiving this Interrupt, the SD Host reads the Arm General purpose register to find the message from Arm.	0x0
1	R/W	READ_ERROR Read Error Interrupt If set to 1, it indicates that the Arm has send an error response during data transaction and the host has to retry the same transaction to prevent data loss.	0x0
0	R/W	READ_DATA_READY_INT Read Ready Interrupt: If set to 1, it indicates that the Arm has data ready to send to the SD Host. The bit remains set to 1 until the SD Host writes 1 to the corresponding bit in the Interrupt identification register. This bit is set to one whenever Arm sets the read_data_rdy bit in read_data_rdy register.	0x0

Table 563: SDIO_AHB_TRANSCOUNT_REG (0x4001010C)

Bit	Mode	Symbol/Description	Reset
20:0	R/W	AHB_XFER_CNT This register contains the transfer count value programmable by Arn for a read transaction.	0x0

Table 564: SDIO_AHB_SDIOTRANSCOUNT_REG (0x40010110)

Bit	Mode	Symbol/Description	Reset
20:0	R	XFER_CNT_REG SDIO_AHB bridge writes the number of bytes transferred to Arm processor during write transfer.	0x0

Table 565: SDIO_AHB_FN1_INT_REG (0x40010114)

Bit	Mode	Symbol/Description	Reset
13	R/W	FUN1_EN	0x0

Bit	Mode	Symbol/Description	Reset
		This bit is set by Arasan SDIO_AHB bridge when SD Host enables the function enable bit in CCCR register.	
12	R/W	ADMA_ERR This bit is set by Arasan SDIO_AHB bridge when Arm sets valid bit as 0 in the attribute field during fetch operation.	0x0
11	R/W	ADMA_INT This bit is set by Arasan SDIO_AHB bridge when int bit is set in the attribute field during fetch operation.	0x0
10	R/W	RESUME This bit is set when resume command is issued for the function1.	0x0
9	R/W	FN1_SUSPEND This bit is set when suspend command is issued for the function1.	0x0
8	R/W	ADMA_END_INT This bit is set when the adma endbit is set during descriptor fetch indicate end of fetch. Then Arm processor ready for next fetch (with new updated address).	0x0
7	R/W	FN1_SDIO_WR_START This bit is set when the SD Host issues CMD53 write command. This bit is set by SDIO AHB bridge.	0x0
6	R/W	FN1_SDIO_RD_START This bit is set when the SD Host issues CMD53 read command. This bit is set by SDIO AHB bridge.	0x0
5	R/W	FN1_ACK_TO_ARM Acknowledgment to Arm. If set to 1, it indicates that the SD Host has read the message.	0x0
4	R/W	SD_HOST_FN1_MSG_RDY SD Host Message Interrupt If set to 1, it indicates that the SD Host has programmed the SD Host General Purpose Register (Message from SD Host to Arm). On receiving this Interrupt, the Arm reads the SD Host General purpose register to find the message from SD Host.	0x0
3	R/W	FUN1_RST IOEx,is a CCCR bit (Enable Functionx). If host disable IOEx bit in CCCR register, this interrupt is asserted to inform the Arm, that funx is disabled by the host. This is used for per function reset error recovery and this signal does not affect the flops inside the IP.	0x0
2	R/W	FN1_RD_ERROR This interrupt is issued if SD Host issue abort command during function 1 read transaction.	0x0
1	R/W	FN1_RD_OVER This interrupt corresponds to the interrupt issued by SD DMA after it completes the read operation of a Function1.	0x0
0	R/W	FN1_WR_OVER This interrupt corresponds to the interrupt issued by SD DMA after it completes the write operation of a Function1. Until this bit is cleared, busy signal is asserted in data line, the SD DMA engine is prevented from performing an AHB write access from a new cmd53 write/read.	0x0

Bit	Mode	Symbol/Description	Reset
		No valid data in the FIFO is overwritten.	

Table 566: SDIO_AHB_FN1_INT_ENABLE_REG (0x40010118)

Bit	Mode	Symbol/Description	Reset
13	R/W	FUN1_EN_INT_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
12	R/W	ADMA_ERR_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
11	R/W	ADMA_INT_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
10	R/W	RESUME_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
9	R/W	FN1_SUSPEND_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
8	R/W	ADMA_END_INT_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
7	R/W	FN1_SDIO_WR_START_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
6	R/W	FN1_SDIO_RD_START_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
5	R/W	FN1_ACK_TO_ARM_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
4	R/W	SD_HOST_FN1_MSG_RDY_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
3	R/W	FUN1_RST_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
2	R/W	FN1_RD_ERROR_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0
1	R/W	FN1_RD_OVER_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0

Bit	Mode	Symbol/Description	Reset
0	R/W	FN1_WR_OVER_EN 1: Interrupt is enabled 0: Interrupt is disabled	0x0

Table 567: SDIO_FBR_REG (0x4001011C)

Bit	Mode	Symbol/Description	Reset
13	R/W	SDIO_SPS Function1 supports high power.	0x0
12:5	R/W	EXTENDED_IO_DEVICE_CODE1 This is the Extension of the Standard I/O Device Code for Function 1.	0x0
4	R/W	FUN_CSA_SUP CSA support bit 1 - CSA Supported. 0 - CSA is not Supported.	0x0
3:0	R/W	IO_DEVICE_CODE1 This value denotes the SDIO standard Inter- face supported by this function (function 1). 0 - No SDIO standard Interface supported bythis function.	0x7

Table 568: SDIO_IOR_REG (0x40010120)

Bit	Mode	Symbol/Description	Reset
0	R/W	IOR_REG Set this bit to indicate that the function is ready. This is used to set IOR1 bit in cccr	0x0

Table 569: SDIO_SD_HOST_GP_REG (0x40010124)

Bit	Mode	Symbol/Description	Reset
31:0	R	SD_HOST_GP SD Host General Purpose Register. An Interrupt is asserted to the Arm, whenever SD Host writes into this register indicating that there is a message for Arm.	0x0

Table 570: SDIO_ARM_GP_REG (0x40010128)

Bit	Mode	Symbol/Description	Reset
31:0	W	ARM_GP Arm General-Purpose Register. An Interrupt is asserted to the SD Host, whenever Arm writes into this register indicating that there is a message for SD Host.	0x0

Table 571: SDIO_RDDATRDY_REG (0x4001012C)

Bit	Mode	Symbol/Description	Reset
0	R/W	FN1_RDDATRDY	0x0

Bit	Mode	Symbol/Description	Reset
		Arm sets this bit to indicate that the function1 read data is ready. This bit is auto cleared. If Arm sets this bit to 1, then it should wait for the fn_read_over or fn_rd_error interrupt.	

Table 572: SDIO_BLKSIZE_REG (0x40010130)

Bit	Mode	Symbol/Description	Reset
12	R	SIN_MUL_BLK This bit indicates current transaction is Block/Byte mode transaction. For cmd 53 block mode this value is 1. For cmd53 byte mode this value is 0	0x0
11:0	R	BLK_SIZE Block Size Register. The Arasan SDIO-AHB Controller read this register based on the block size programmed by the SD host	0x0

Table 573: SDIO_ARGUMENT_REG (0x40010134)

Bit	Mode	Symbol/Description	Reset
31:0	R	ARG_REG IO transaction: [8:0] -> Byte/Block count In case of Block mode it indicates Block count value. In case of Byte mode it indicates Byte count value. [25:9] -> Register address field in CMD53. [26] -> Opcode [27] -> Blockmode. 1 - Block mode. 0 - Byte mode. [30:28] -> Function Number field in CMD53 [31] -> R/W flag field in CMD53. 1 -> Write CMD53. 0 -> Read CMD53	0x0

Table 574: SDIO_WRBLKCNT_REG (0x40010138)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	WR_BLK_CNT This register has the value of how many blocks are transferred from SD Host to Arm. On receiving fn1_wr_over, Arm should clear this register.	0x0

Table 575: SDIO_RDBLKCNT_REG (0x4001013C)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	RD_BLK_CNT This register has the value of how many blocks are transferred from Arm to SD Host. On receiving the fn1_read_over interrupt, Arm should clear this register	0x0

10.20 SPI Registers

Table 576: Register map SPI

Address	Register	Description
0x40090200	SPI_CTRL_REG	SPI control register

Address	Register	Description
0x40090204	SPI_CONFIG_REG	SPI control register
0x40090208	SPI_CLOCK_REG	SPI clock register
0x4009020c	SPI_FIFO_CONFIG_REG	SPI FIFO configuration register
0x40090210	SPI_IRQ_MASK_REG	SPI interrupt mask register
0x40090214	SPI_STATUS_REG	SPI status register
0x40090218	SPI_FIFO_STATUS_REG	SPI RX/TX FIFO status register
0x4009021c	SPI_FIFO_READ_REG	SPI RX FIFO read register
0x40090220	SPI_FIFO_WRITE_REG	SPI TX FIFO write register
0x40090224	SPI_CS_CONFIG_REG	SPI CS configuration register
0x4009022c	SPI_TXBUFFER_FORCE_REG	SPI TX buffer force low value

Table 577: SPI_CTRL_REG (0x40090200)

Bit	Mode	Symbol/Description	Reset
7	R/W	SPI_SWAP_BYTES 0 = Normal operation 1 = LSB and MSB are swaped in APB interface In case of 8-bit SPI interface, DMA/SPI can be configured in 16-bit mode to off load the bus. Enabling SPI_SWAP_BYTES bytes to read/write correctly	0x0
6	R/W	SPI_CAPTURE_AT_NEXT_EDGE 0 = SPI captures data at correct clock edge 1 = SPI captures data at next clock edge. (only for Master mode and high clock)	0x0
5	R/W	SPI_FIFO_RESET 0 = FIFO normal operation 1 = FIFO in reset state	0x0
4	R/W	SPI_DMA_RX_EN Applicable only when SPI_RX_EN = 1 0 = No DMA request for RX 1 = DMA request when SPI_STATUS_RX_FULL = 1	0x0
3	R/W	SPI_DMA_TX_EN Applicable only when SPI_TX_EN = 1 0 = No DMA request for TX 1 = DMA request when SPI_STATUS_TX_EMPTY = 1	0x0
2	R/W	SPI_RX_EN 0 = RX path is disabled 1 = RX path is enabled Note: if SPI mode = 1 or SPI mode = 3 readonly is not supported	0x0
1	R/W	SPI_TX_EN 0 = TX path is disabled 1 = TX path is enabled	0x0
0	R/W	SPI_EN 0 = SPI module is disabled	0x0

Bit	Mode	Symbol/Description	Reset
		1 = SPI module is enabled	

Table 578: SPI_CONFIG_REG (0x40090204)

Bit	Mode	Symbol/Description	Reset
7	R/W	SPI_SLAVE_EN 0 = SPI module master mode 1 = SPI module slave mode	0x0
6:2	R/W	SPI_WORD_LENGTH Define the SPI word length = 1+ SPI_WORD_LENGTH (range 4 to 32) Note: Should be changed with SPI_EN = 0	0x0
1:0	R/W	SPI_MODE Define the SPI mode (CPOL, CPHA) 0 = New data on falling, capture on rising, CLK low in idle state 1 = New data on rising, capture on falling, CLK low in idle state 2 = New data on rising, capture on falling, CLK high in idle state 3 = New data on falling, capture on rising CLK high in idle state	0x0

Table 579: SPI_CLOCK_REG (0x40090208)

Bit	Mode	Symbol/Description	Reset
6:0	R/W	SPI_CLK_DIV Applicable only in master mode Defines the SPI clock frequency in master only mode SPI_CLK = MODULE_CLK/2*(SPI_CLK_DIV+1) when SPI_CLK_DIV not 0x7F if SPI_CLK_DIV = 0x7F, then SPI_CLK = MODULE_CLK	0x0

Table 580: SPI_FIFO_CONFIG_REG (0x4009020C)

Bit	Mode	Symbol/Description	Reset
15:8	R/W	SPI_RX_TL Receive FIFO threshold level in bytes. Control the level of bytes in FIFO that triggers the RX_FULL interrupt. IRQ is occurred when FIFO level is more or equal to SPI_RX_TL+1. Valid FIFO level is from 0 to 32	0x0
7:0	R/W	SPI_TX_TL Transmit FIFO threshold level in bytes. Control the level of bytes in FIFO that triggers the TX_EMPTY interrupt. IRQ is occurred when FIFO level is less or equal to SPI_TX_TL. Valid FIFO level is from 0 to 32	0x0

Table 581: SPI_IRQ_MASK_REG (0x40090210)

Bit	Mode	Symbol/Description	Reset
1	R/W	SPI_IRQ_MASK_RX_FULL 0 = FIFO RX full IRQ is masked 1 = FIFO RX full IRQ is enabled	0x0
0	R/W	SPI_IRQ_MASK_TX_EMPTY	0x0

Bit	Mode	Symbol/Description	Reset
		0 = FIFO TX empty IRQ is masked 1 = FIFO TX empy IRQ is enabled	

Table 582: SPI_STATUS_REG (0x40090214)

Bit	Mode	Symbol/Description	Reset
1	R	SPI_STATUS_RX_FULL Auto clear 0 = RX FIFO level is less than SPI_RX_TL+1 1 = RX FIFO level is more or equal to SPI_RX_TL+1	0x0
0	R	SPI_STATUS_TX_EMPTY Auto clear 0 = TX FIFO level is larger than SPI_TX_TL 1 = TX FIFO level is less or equal to SPI_TX_TL	0x1

Table 583: SPI_FIFO_STATUS_REG (0x40090218)

Bit	Mode	Symbol/Description	Reset
15	R	SPI_TRANSACTION_ACTIVE In master mode 0 = SPI transaction is inactive 1 = SPI transaction is active	0x0
14	R	SPI_RX_FIFO_OVFL When 1, receive data is not written to FIFO because FIFO was full and interrupt was generated. It clears with SPI_CTRL_REG.SPI_FIFO_RESET	0x0
13	R	SPI_STATUS_TX_FULL 0 = TX FIFO is not full 1 = TX FIFO is full	0x0
12	R	SPI_STATUS_RX_EMPTY 0 = RX FIFO is not empty 1 = RX FIFO is empty	0x1
11:6	R	SPI_TX_FIFO_LEVEL Gives the number of bytes in TX FIFO	0x0
5:0	R	SPI_RX_FIFO_LEVEL Gives the number of bytes in RX FIFO	0x0

Table 584: SPI_FIFO_READ_REG (0x4009021C)

Bit	Mode	Symbol/Description	Reset
31:0	R	SPI_FIFO_READ Read from RX FIFO Read access is permitted only if SPI_RX_FIFO_EMPTY = 0.	0x0

Table 585: SPI_FIFO_WRITE_REG (0x40090220)

Bit	Mode	Symbol/Description	Reset
31:0	R0/W	SPI_FIFO_WRITE	0x0

Bit	Mode	Symbol/Description	Reset
		Write to TX FIFO. Write access is permitted only if SPI_TX_FIFO_FULL is 0	

Table 586: SPI_CS_CONFIG_REG (0x40090224)

Bit	Mode	Symbol/Description	Reset
2:0	R/W	SPI_CS_SELECT Control the cs output in master mode 0 = No slave device selected 1 = Selected slave device connected to GPIO with FUNC_MODE=SPI_CS0 2 = Selected slave device connected to GPIO with FUNC_MODE=SPI_CS1 4 = Selected slave device connected to GPIO with FUNC_MODE=GPIO	0x0

Table 587: SPI_TXBUFFER_FORCE_REG (0x4009022C)

Bit	Mode	Symbol/Description	Reset
31:0	W	SPI_TXBUFFER_FORCE Write the TX buffer directly. It must to be used only in slave mode.	0x0

10.21 SPI2 Registers

Table 588: Register map SPI2

Address	Register	Description
0x40090300	SPI2_CTRL_REG	SPI control register
0x40090304	SPI2_CONFIG_REG	SPI control register
0x40090308	SPI2_CLOCK_REG	SPI clock register
0x4009030c	SPI2_FIFO_CONFIG_REG	SPI FIFO configuration register
0x40090310	SPI2_IRQ_MASK_REG	SPI interrupt mask register
0x40090314	SPI2_STATUS_REG	SPI status register
0x40090318	SPI2_FIFO_STATUS_REG	SPI RX/TX FIFO status register
0x4009031c	SPI2_FIFO_READ_REG	SPI RX FIFO read register
0x40090320	SPI2_FIFO_WRITE_REG	SPI TX FIFO write register
0x40090324	SPI2_CS_CONFIG_REG	SPI CS configuration register
0x4009032c	SPI2_TXBUFFER_FORCE_REG	SPI TX buffer force low value

Table 589: SPI2_CTRL_REG (0x40090300)

Bit	Mode	Symbol/Description	Reset
7	R/W	SPI_SWAP_BYTES 0 = Normal operation 1 = LSB and MSB are swaped in APB interface In case of 8-bit SPI interface, DMA/SPI can be configured in 16-bit mode to off load the bus. Enabling SPI_SWAP_BYTES bytes to read/write correctly	0x0
6	R/W	SPI_CAPTURE_AT_NEXT_EDGE	0x0

Bit	Mode	Symbol/Description	Reset
		0 = SPI captures data at correct clock edge 1 = SPI captures data at next clock edge. (only for Master mode and high clock)	
5	R/W	SPI_FIFO_RESET 0 = FIFO normal operation 1 = FIFO in reset state	0x0
4	R/W	SPI_DMA_RX_EN Applicable only when SPI_RX_EN = 1 0 = No DMA request for RX 1 = DMA request when SPI_STATUS_RX_FULL = 1	0x0
3	R/W	SPI_DMA_TX_EN Applicable only when SPI_TX_EN = 1 0 = No DMA request for TX 1 = DMA request when SPI_STATUS_TX_EMPTY = 1	0x0
2	R/W	SPI_RX_EN 0 = RX path is disabled 1 = RX path is enabled Note: if SPI mode = 1 or SPI mode = 3 readonly is not supported	0x0
1	R/W	SPI_TX_EN 0 = TX path is disabled 1 = TX path is enabled	0x0
0	R/W	SPI_EN 0 = SPI module is disabled 1 = SPI module is enabled	0x0

Table 590: SPI2_CONFIG_REG (0x40090304)

Bit	Mode	Symbol/Description	Reset
7	R/W	SPI_SLAVE_EN 0 = SPI module master mode 1 = SPI module slave mode	0x0
6:2	R/W	SPI_WORD_LENGTH Define the SPI word length = 1+ SPI_WORD_LENGTH (range 4 to 32) Note: Should be changed with SPI_EN = 0	0x0
1:0	R/W	SPI_MODE Define the SPI mode (CPOL, CPHA) 0 = New data on falling, capture on rising, CLK low in idle state 1 = New data on rising, capture on falling, CLK low in idle state 2 = New data on rising, capture on falling, CLK high in idle state 3 = New data on falling, capture on rising CLK high in idle state	0x0

Table 591: SPI2_CLOCK_REG (0x40090308)

Bit	Mode	Symbol/Description	Reset
6:0	R/W	SPI_CLK_DIV	0x0

Bit	Mode	Symbol/Description	Reset
		Applicable only in master mode Defines the SPI clock frequency in master only mode SPI_CLK = MODULE_CLK/2*(SPI_CLK_DIV+1) when SPI_CLK_DIV not 0x7F if SPI_CLK_DIV = 0x7F, then SPI_CLK = MODULE_CLK	

Table 592: SPI2_FIFO_CONFIG_REG (0x4009030C)

Bit	Mode	Symbol/Description	Reset
15:8	R/W	SPI_RX_TL Receive FIFO threshold level in bytes. Control the level of bytes in FIFO that triggers the RX_FULL interrupt. IRQ is occurred when FIFO level is more or equal to SPI_RX_TL+1. Valid FIFO level is from 0 to 32	0x0
7:0	R/W	SPI_TX_TL Transmit FIFO threshold level in bytes. Control the level of bytes in FIFO that triggers the TX_EMPTY interrupt. IRQ is occurred when FIFO level is less or equal to SPI_TX_TL. Valid FIFO level is from 0 to 32	0x0

Table 593: SPI2_IRQ_MASK_REG (0x40090310)

Bit	Mode	Symbol/Description	Reset
1	R/W	SPI_IRQ_MASK_RX_FULL 0 = FIFO RX full IRQ is masked 1 = FIFO RX full IRQ is enabled	0x0
0	R/W	SPI_IRQ_MASK_TX_EMPTY 0 = FIFO TX empty IRQ is masked 1 = FIFO TX empy IRQ is enabled	0x0

Table 594: SPI2_STATUS_REG (0x40090314)

Bit	Mode	Symbol/Description	Reset
1	R	SPI_STATUS_RX_FULL Auto clear 0 = RX FIFO level is less than SPI_RX_TL+1 1 = RX FIFO level is more or equal to SPI_RX_TL+1	0x0
0	R	SPI_STATUS_TX_EMPTY Auto clear 0 = TX FIFO level is larger than SPI_TX_TL 1 = TX FIFO level is less or equal to SPI_TX_TL	0x1

Table 595: SPI2_FIFO_STATUS_REG (0x40090318)

Bit	Mode	Symbol/Description	Reset
15	R	SPI_TRANSACTION_ACTIVE In master mode 0 = SPI transaction is inactive 1 = SPI transaction is active	0x0
14	R	SPI_RX_FIFO_OVFL	0x0

Bit	Mode	Symbol/Description	Reset
		When 1, receive data is not written to FIFO because FIFO was full and interrupt was generated. It clears with SPI_CTRL_REG.SPI_FIFO_RESET	
13	R	SPI_STATUS_TX_FULL 0 = TX FIFO is not full 1 = TX FIFO is full	0x0
12	R	SPI_STATUS_RX_EMPTY 0 = RX FIFO is not empty 1 = RX FIFO is empty	0x1
11:6	R	SPI_TX_FIFO_LEVEL Gives the number of bytes in TX FIFO	0x0
5:0	R	SPI_RX_FIFO_LEVEL Gives the number of bytes in RX FIFO	0x0

Table 596: SPI2_FIFO_READ_REG (0x4009031C)

Bit	Mode	Symbol/Description	Reset
31:0	R	SPI_FIFO_READ Read from RX FIFO Read access is permitted only if SPI_RX_FIFO_EMPTY = 0.	0x0

Table 597: SPI2_FIFO_WRITE_REG (0x40090320)

Bit	Mode	Symbol/Description	Reset
31:0	R0/W	SPI_FIFO_WRITE Write to TX FIFO. Write access is permitted only if SPI_TX_FIFO_FULL is 0	0x0

Table 598: SPI2_CS_CONFIG_REG (0x40090324)

Bit	Mode	Symbol/Description	Reset
2:0	R/W	SPI_CS_SELECT Control the cs output in master mode 0 = No slave device selected 1 = Selected slave device connected to GPIO with FUNC_MODE=SPI_CS0 2 = Selected slave device connected to GPIO with FUNC_MODE=SPI_CS1 4 = Selected slave device connected to GPIO with FUNC_MODE=GPIO	0x0

Table 599: SPI2_TXBUFFER_FORCE_REG (0x4009032C)

Bit	Mode	Symbol/Description	Reset
31:0	W	SPI_TXBUFFER_FORCE Write the TX buffer directly. It must to be used only in slave mode.	0x0

10.22 Source FIFO Registers

Table 600: Register map SRC_FIFO_IF

Address	Register	Description
0x40003000	APU_FIFO_STATUS_REG	APU FIFO Status
0x40003010	APU_SRC_FIFO_IN1_REG	SRC data in 1
0x40003014	APU_SRC_FIFO_IN2_REG	SRC data in 2
0x40003018	APU_SRC_FIFO_OUT1_REG	SRC data out 1
0x4000301c	APU_SRC_FIFO_OUT2_REG	SRC data out 2
0x40003020	APU_DAI_FIFO_IN1_REG	DAI data in 1
0x40003024	APU_DAI_FIFO_IN2_REG	DAI data in 2
0x40003028	APU_DAI_FIFO_OUT1_REG	DAI data out 1
0x4000302c	APU_DAI_FIFO_OUT2_REG	DAI data out 2

Table 601: APU_FIFO_STATUS_REG (0x40003000)

Bit	Mode	Symbol/Description	Reset
23	R/W	DAI_CH2_IN_FULL	0
		DAI input FIFO full	
22	R/W	DAI_CH2_IN_EMPTY	1
		DAI input FIFO empty	
21	R/W	DAI_CH2_OUT_FULL	0
		DAI output FIFO full	
20	R/W	DAI_CH2_OUT_EMPTY	1
		DAI output FIFO empty	
19	R/W	DAI_CH1_IN_FULL	0
		DAI input FIFO full	
18	R/W	DAI_CH1_IN_EMPTY	1
		DAI input FIFO empty	
17	R/W	DAI_CH1_OUT_FULL	0
		DAI output FIFO full	
16	R/W	DAI_CH1_OUT_EMPTY	1
		DAI output FIFO empty	
7	R/W	SRC_CH2_OUT_FULL	0
		SRC output FIFO full	
6	R/W	SRC_CH2_OUT_EMPTY	1
		SRC output FIFO empty	

Bit	Mode	Symbol/Description	Reset
5	R/W	SRC_CH2_IN_FULL SRC input FIFO full	0
4	R/W	SRC_CH2_IN_EMPTY SRC input FIFO empty	1
3	R/W	SRC_CH1_OUT_FULL SRC output FIFO full	0
2	R/W	SRC_CH1_OUT_EMPTY SRC output FIFO empty	1
1	R/W	SRC_CH1_IN_FULL SRC input FIFO full	0
0	R/W	SRC_CH1_IN_EMPTY SRC input FIFO empty	1

Table 602: APU_SRC_FIFO_IN1_REG (0x40003010)

Bit	Mode	Symbol/Description	Reset
31:0	W	SRC_IN SRC_FIFO_IN1	0

Table 603: APU_SRC_FIFO_IN2_REG (0x40003014)

Bit	Mode	Symbol/Description	Reset
31:0	W	SRC_IN SRC_FIFO_IN2	0

Table 604: APU_SRC_FIFO_OUT1_REG (0x40003018)

Bit	Mode	Symbol/Description	Reset
31:0	R	SRC_OUT SRC_FIFO_OUT1	0

Table 605: APU_SRC_FIFO_OUT2_REG (0x4000301C)

Bit	Mode	Symbol/Description	Reset
31:0	R	SRC_OUT SRC_FIFO_OUT2	0

Table 606: APU_DAI_FIFO_IN1_REG (0x40003020)

Bit	Mode	Symbol/Description	Reset
31:0	W	DAI_IN DAI_FIFO_OUT1	0

Table 607: APU_DAI_FIFO_IN2_REG (0x40003024)

Bit	Mode	Symbol/Description	Reset
31:0	W	DAI_IN DAI_FIFO_OUT2	0

Table 608: APU_DAI_FIFO_OUT1_REG (0x40003028)

Bit	Mode	Symbol/Description	Reset
31:0	R	DAI_OUT DAI_FIFO_IN1	0

Table 609: APU_DAI_FIFO_OUT2_REG (0x4000302C)

Bit	Mode	Symbol/Description	Reset
31:0	R	DAI_OUT DAI_FIFO_IN2	0

10.23 Source Interface Registers

Table 610: Register map SRC_IF

Address	Register	Description
0x400e0c20	APU_SRC_CTRL_REG	SRC control register
0x400e0c24	APU_SRC_IN_FS_REG	SRC Sample input rate
0x400e0c28	APU_SRC_OUT_FS_REG	SRC Sample output rate
0x400e0c2c	APU_SRC_IN1_REG	SRC data in 1
0x400e0c30	APU_SRC_IN2_REG	SRC data in 2
0x400e0c34	APU_SRC_OUT1_REG	SRC data out 1
0x400e0c38	APU_SRC_OUT2_REG	SRC data out 2
0x400e0c40	APU_SRC_FIFO_CTRL_REG	SRC FIFO control
0x400e0c44	APU_DAI_FIFO_CTRL_REG	DAI FIFO control
0x400e0ce0	APU_SRC_COEF10_SET1_REG	SRC coefficient 1,0 set 1
0x400e0ce4	APU_SRC_COEF32_SET1_REG	SRC coefficient 3,2 set 1
0x400e0ce8	APU_SRC_COEF54_SET1_REG	SRC coefficient 5,4 set 1
0x400e0cec	APU_SRC_COEF76_SET1_REG	SRC coefficient 7,6 set 1
0x400e0cf0	APU_SRC_COEF98_SET1_REG	SRC coefficient 9,8 set 1
0x400e0cf4	APU_SRC_COEF0A_SET1_REG	SRC coefficient 10 set 1
0x400e0d00	APU_SRC_COEF10_SET2_REG	SRC coefficient 1,0 set 2

Address	Register	Description
0x400e0d04	APU_SRC_COEF32_SET2_REG	SRC coefficient 3,2 set 2
0x400e0d08	APU_SRC_COEF54_SET2_REG	SRC coefficient 5,4 set 2
0x400e0d0c	APU_SRC_COEF76_SET2_REG	SRC coefficient 7,6 set 2
0x400e0d10	APU_SRC_COEF98_SET2_REG	SRC coefficient 9,8 set 2
0x400e0d14	APU_SRC_COEF0A_SET2_REG	SRC coefficient 10 set 2

Table 611: APU_SRC_CTRL_REG (0x400E0C20)

Bit	Mode	Symbol/Description	Reset
31	-	-	0x0
		Reserved	
30	R/W	SRC_IN_COEF_SET	0x0
		SRC_IN Filter coefficient set selection 0: Set 1 is intended for wideband and fullband audio (16 kHz - 48 kHz). The filters are optimized for a narrow transition band and excellent anti-aliasing suppression. 1: Set 2 is for intended for narrowband and wideband voice calls (8 kHz - 16 kHz). The filters are optimized for low-latency.	
29:28	-	-	0x0
		Reserved	
27	R/W	SRC_FIFO_OUT2_ENABLE	0x0
		0: FIFO_OUT2 disable. On each SRC request, one sample is serviced 1: FIFO_OUT2 enable. FIFO is used to store samples from SRC SRC supports only DMA burst size 4 when FIFO is enabled, else no burst	
26	R/W	SRC_FIFO_OUT1_ENABLE	0x0
		0: FIFO_OUT1 disable. On each SRC request, one sample is serviced 1: FIFO_OUT1 enable. FIFO is used to store samples from SRC SRC supports only DMA burst size 4 when FIFO is enabled, else no burst	
25	R0/W	SRC_OUT_FLOWCLR	0
		Writing 1 clears the SRC_OUT overflow/underflow bits 23 - 22. No more over/underflow indications while bit is 1. Keep 1 until the over/under flow bit is cleared.	
24	W	SRC_IN_FLOWCLR	0
		Writing 1 clears the SRC_IN overflow/underflow bits 21-20. No more over/underflow indications while bit is 1. Keep 1 until the over/under flow bit is cleared.	
23	R	SRC_OUT_UNFLOW	0
		1: SRC_OUT underflow occurred	
22	R	SRC_OUT_OVFLOW	0
		1: SRC_OUT overflow occurred	
21	R	SRC_IN_UNFLOW	0
		1: SRC_IN underflow occurred	

Bit	Mode	Symbol/Description	Reset
20	R	SRC_IN_OVFLOW 1: SRC_IN overflow occurred	0
19	R0/W	SRC_RESYNC 1: SRC restarts synchronisation	0
18	R	SRC_OUT_OK SRC_OUT Status 0: Aquisition in progress 1: Aquisition ready (In manual mode, this bit is always 1)	0
17:16	R/W	SRC_OUT_US SRC_OUT upsampling IIR filters setting 00: For sample rates up to 48 kHz 01: For sample rates of 96 kHz 10: Reserved 11: For sample rates of 192 kHz	0
15	R/W	SRC_OUT_SET SRC_OUT filter coefficient set selection 0: Set 1 is intended for wide band and full band audio (16 kHz - 48 kHz). The filters are optimized for a narrow transition band and excellent anti-aliasing suppression. 1: Set 2 is for intended for narrowband and wide band voice calls (8 kHz - 16 kHz). The filters are optimized for low-latency.	0x0
14	R/W	SRC_OUT_CAL_BYPASS SRC_OUT1 upsampling filter bypass 0: Do not bypass 1: Bypass filter	0
13	R/W	SRC_OUT_AMODE SRC_OUT1 automatic conversion mode 0: Manual mode 1: Automatic mode	0
12	R/W	SRC_PDM_EN PDM input enable	0x0
11	R/W	SRC_FIFO_IN2_ENABLE 0: FIFO_IN2 disable. On each SRC request, one sample is serviced 1: FIFO_IN2 enable. FIFO is used to store samples to SRC SRC supports only DMA burst size 4 when FIFO is enable else no burst	0x0
10	R/W	SRC_FIFO_IN1_ENABLE 0: FIFO_IN1 disable. On each SRC request, one sample is serviced 1: FIFO_IN1 enable. FIFO is used to store samples to SRC SRC supports only DMA burst size 4 when FIFO is enable else no burst	0x0
9:8	R/W	SRC_PCM_GAIN PCM input gain	0x0
7	R/W	SRC_DITHER_DISABLE Dithering feature 0: Enabled 1: Disabled	0x0

Bit	Mode	Symbol/Description	Reset
6	R	SRC_IN_OK SRC_IN status 0: Acquisition in progress 1: Acquisition ready	0
5:4	R/W	SRC_IN_DS SRC_IN upsampling IIR filters setting 00: For sample rates up to 48 kHz 01: For sample rates of 96 kHz 10: Reserved 11: For sample rates of 192 kHz	0
3	R/W	SRC_PDM_IN_INV Invert PDM_CLK, see timing diagrams	0x0
2	R/W	SRC_IN_CAL_BYPASS SRC_IN upsampling filter bypass 0: Do not bypass 1: Bypass filter	0
1	R/W	SRC_IN_AMODE SRC_IN Automatic conversion mode 0: Manual mode 1: Automatic mode	0
0	R/W	SRC_EN SRC_IN and SRC_OUT enable 0: Disabled 1: Enabled	0

Table 612: APU_SRC_IN_FS_REG (0x400E0C24)

Bit	Mode	Symbol/Description	Reset
31:24	-	- Reserved	0
23:0	R/W	SRC_IN_FS SRC_IN Sample rate SRC_IN_FS = 4096*Sample_rate/100*(32768/SRC_CLK) Sample_rate upper limit is 192 kHz. For 96 kHz and 192 kHz SRC_CTRLx_REG[SRC_IN_DS] must be set as shown below: For SRC_CLK = 32.768 kHz, SRC_DIV = 1: Sample_rate SRC_IN_FS SRC_IN_DS Audio bandwidth 8000 Hz 0x050000 0 4000 Hz 11025 Hz 0x06E400 0 5512 Hz 16000 Hz 0x0A0000 0 8000 Hz 22050 Hz 0x0DC800 0 11025 Hz 32000 Hz 0x140000 0 16000 Hz 44100 Hz 0x1B9000 0 22050 Hz 48000 Hz 0x1E0000 0 24000 Hz 96000 Hz 0x1E0000 1 24000 Hz	0

Bit	Mode	Symbol/Description	Reset
		192000 Hz 0x1E0000 3 24000 Hz In manual SRC mode, SRC_IN_FS can be set and adjusted to the desired sample rate at any time. In automatic mode, the SRC returns the final sample rate as soon as SRC_IN_OK. Note that SRC_DS is not calculated in automatic mode and must be set manually automatic mode with Sample_rate of 96 kHz and 192 kHz.	

Table 613: APU_SRC_OUT_FS_REG (0x400E0C28)

Bit	Mode	Symbol/Description	Reset
31:24	-	- Reserved	0
23:0	R/W	SRC_OUT_FS SRC_OUT Sample rate SRC_IN_FS = 4096*Sample_rate/100*(32768/SRC_CLK) Sample_rate upper limit is 192 kHz. For 96 kHz and 192 kHz SRC_CTRLx_REG[SRC_OUT_US] must be set as shown below: For SRC_CLK = 32.768 kHz, SRC_DIV = 1: Sample_rate SRC_OUT_FS SRC_OUT_US Audio bandwidth 8000 Hz 0x050000 0 4000 Hz 11025 Hz 0x06E400 0 5512 Hz 16000 Hz 0x0A0000 0 8000 Hz 22050 Hz 0x0DC800 0 11025 Hz 32000 Hz 0x140000 0 16000 Hz 44100 Hz 0x1B9000 0 22050 Hz 48000 Hz 0x1E0000 0 24000 Hz 96000 Hz 0x1E0000 1 24000 Hz 192000 Hz 0x1E0000 3 24000 Hz In manual SRC mode, SRC_OUT_FS can be set and adjusted to the desired sample rate at any time. In automatic mode, the SRC returns the final sample rate as soon as SRC_OUT_OK. Note that SRC_DS is not calculated in automatic mode and must be set manually automatic mode with Sample_rate of 96 kHz and 192 kHz.	0

Table 614: APU_SRC_IN1_REG (0x400E0C2C)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	SRC_IN SRC_IN1	0

Table 615: APU_SRC_IN2_REG (0x400E0C30)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	SRC_IN SRC_IN2	0

Table 616: APU_SRC_OUT1_REG (0x400E0C34)

Bit	Mode	Symbol/Description	Reset
31:0	R	SRC_OUT	0
		SRC_OUT1	

Table 617: APU_SRC_OUT2_REG (0x400E0C38)

Bit	Mode	Symbol/Description	Reset
31:0	R	SRC_OUT	0
		SRC_OUT2	

Table 618: APU_SRC_FIFO_CTRL_REG (0x400E0C40)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	FIFO_CTRL	0
		Bit0: FIFO enable Bit10: Input channel 1 enabled Bit11: Input channel 2 enabled Bit26: Output channel 1 enabled Bit27: Output channel 2 enabled	

Table 619: APU_DAI_FIFO_CTRL_REG (0x400E0C44)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	FIFO_CTRL	0
		Bit0: FIFO enabled Bit10: Input channel 1 enabled Bit11: Input channel 2 enabled Bit26: Output channel 1 enabled Bit27: Output channel 2 enabled	

Table 620: APU_SRC_COEF10_SET1_REG (0x400E0CE0)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF1	0x7A20
		Coefficient 1	
15:0	R/W	SRC_COEF0	0x8EC4
		Coefficient 0	

Table 621: APU_SRC_COEF32_SET1_REG (0x400E0CE4)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF3	0x70FD
		Coefficient 3	
15:0	R/W	SRC_COEF2	0x8936
		Coefficient 2	

Table 622: APU_SRC_COEF54_SET1_REG (0x400E0CE8)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF5	0x9758
		Coefficient 5	
15:0	R/W	SRC_COEF4	0xB686
		Coefficient 4	

Table 623: APU_SRC_COEF76_SET1_REG (0x400E0CEC)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF7	0x89C4
		Coefficient 7	
15:0	R/W	SRC_COEF6	0x7DF5
		Coefficient 6	

Table 624: APU_SRC_COEF98_SET1_REG (0x400E0CF0)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF9	0x8F18
		Coefficient 9	
15:0	R/W	SRC_COEF8	0x7771
		Coefficient 8	

Table 625: APU_SRC_COEF0A_SET1_REG (0x400E0CF4)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	SRC_COEF10	0x497D
		Coefficient 10	

Table 626: APU_SRC_COEF10_SET2_REG (0x400E0D00)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF1	0x7300
		Coefficient 1	
15:0	R/W	SRC_COEF0	0x8E9C
		Coefficient 0	

Table 627: APU_SRC_COEF32_SET2_REG (0x400E0D04)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF3	0x3517
		Coefficient 3	
15:0	R/W	SRC_COEF2	0xA082
		Coefficient 2	

Table 628: APU_SRC_COEF54_SET2_REG (0x400E0D08)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF5 Coefficient 5	0x93CF
15:0	R/W	SRC_COEF4 Coefficient 4	0xF519

Table 629: APU_SRC_COEF76_SET2_REG (0x400E0D0C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF7 Coefficient 7	0x9087
15:0	R/W	SRC_COEF6 Coefficient 6	0x7B4E

Table 630: APU_SRC_COEF98_SET2_REG (0x400E0D10)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	SRC_COEF9 Coefficient 9	0xCB49
15:0	R/W	SRC_COEF8 Coefficient 8	0x60AF

Table 631: APU_SRC_COEF0A_SET2_REG (0x400E0D14)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	SRC_COEF10 Coefficient 10	0xAFA

10.24 Watchdog Timer Control Registers

Table 632: Register map SYS_WDOG

Address	Register	Description
0x400c0700	WATCHDOG_REG	Watchdog timer register.
0x400c0704	WATCHDOG_CTRL_REG	

Table 633: WATCHDOG_REG (0x400C0700)

Bit	Mode	Symbol/Description	Reset
31:14	R0/W	WDOG_WEN If Bit [31:14] = 0, then write enable for watchdog timer, else write disable. This filter prevents unintentional presetting the watchdog with a software runaway.	0x0
13	R/W	WDOG_VAL_NEG 0 = Watchdog timer value is positive. 1 = Watchdog timer value is negative.	0x0

Bit	Mode	Symbol/Description	Reset

12:0	R/W	WDOG_VAL
		Write: Watchdog timer reload value. Note that all bits [31-14] must be 0 to reload this register.
		Read: Actual Watchdog timer value. Decremented by 1 every ~10 ms (RC32K) or ~29 ms (RCX), the Watchdog timer clock tick.
		Bit 13 indicates a negative counter value. 2, 1, 0, $3FFF_{16}$, $3FFE_{16}$ and so forth. An NMI or WDOG (SYS) reset is generated under the following conditions:
		If WATCHDOG_CTRL_REG[NMI_RST] = 0 then
		If WDOG_VAL = 0 -> NMI (Non Maskable Interrupt)
		if WDOG_VAL =$3FF0_{16}$ -> WDOG reset -> reload $1FFF_{16}$
		If WATCHDOG_CTRL_REG[NMI_RST] = 1 then
		if WDOG_VAL <= 0 -> WDOG reset -> reload $1FFF_{16}$
		Note 1: The programmed value WDOG_VAL is updated in the (independent) Watchdog timer at the second next RC32K or RCX clock tick.
		Note 2: Select RC32K or RCX with CLK_RCX_REG[RCX_ENABLE]. The RC32K is selected by default.
		Note 3: If WATCHDOG_CTRL_REG[NMI_RST] = 0, the time between the NMI generation and the WDOG reset generation is 15 Watchdog timer clock ticks.
			0x1FFF

Table 634: WATCHDOG_CTRL_REG (0x400C0704)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3	R	WRITE_BUSY 0 = A new WATCHDOG_REG[WDOG_VAL] can be written. 1 = No new WATCHDOG_REG[WDOG_VAL] can be written. Note: It takes some time before the programmed WDOG_VAL is updated in the (independent) Watchdog timer. During this time, it is not possible to write a new value to WATCHDOG_REG[WDOG_VAL].	0x0
2	R/W	WDOG_FREEZE_EN 0 = Watchdog timer cannot be frozen when NMI_RST = 0. 1 = Watchdog timer can be frozen/resumed using SET_FREEZE_REG[FRZ_WDOG]/RESET_FREEZE_REG[FRZ_WDOG] when NMI_RST = 0.	0x1
1	R/W	- Reserved	0x1
0	R/W	NMI_RST 0 = Watchdog timer generates NMI at value 0, and WDOG (SYS) reset at <= -16. Timer can be frozen/resumed using SET_FREEZE_REG[FRZ_WDOG] / RESET_FREEZE_REG[FRZ_WDOG]. 1 = Watchdog timer generates a WDOG (SYS) reset at value 0 and cannot be frozen by Software. Note that this bit can only be set to 1 by Software and only be reset with a WDOG (SYS) reset or Software reset. The watchdog is always frozen when the Cortex-M33 is halted in DEBUG state.	0x0

10.25 AHB DMA Registers

Table 635: Register map SYSBUS

Address	Register	Description
0x2f000000	AHB_DMA_PL1_REG	AHB-DMA layer priority level RFMON
0x2f000004	AHB_DMA_PL2_REG	AHB-DMA layer priority level GEN-DMA
0x2f000008	AHB_DMA_PL3_REG	AHB-DMA layer priority level kDMA
0x2f00000c	AHB_DMA_PL4_REG	AHB-DMA layer priority level SDIO_M
0x2f000010	AHB_DMA_PL5_REG	AHB-DMA layer priority level SDeMMC_M
0x2f000014	AHB_DMA_PL6_REG	AHB-DMA layer priority level CC312_M
0x2f000018	AHB_DMA_PL7_REG	AHB-DMA layer priority level CPU-S to AHB_DMA arbiter registers
0x2f000048	AHB_DMA_DFLT_MASTER_REG	Default master ID number (AHB DMA layer only)
0x2f00004c	AHB_DMA_WTEN_REG	Weighted-Token Arbitration Scheme Enable (AHB DMA layer only)
0x2f000050	AHB_DMA_TCL_REG	Master clock refresh period (AHB DMA layer only)
0x2f000054	AHB_DMA_CCLM1_REG	RFMON Master clock tokens
0x2f000058	AHB_DMA_CCLM2_REG	GEN-DMA Master clock tokens
0x2f00005c	AHB_DMA_CCLM3_REG	kDMA Master clock tokens
0x2f000060	AHB_DMA_CCLM4_REG	SDIO Master clock tokens
0x2f000064	AHB_DMA_CCLM5_REG	SDeMMC Master clock tokens
0x2f000068	AHB_DMA_CCLM6_REG	CPU-S Master clock tokens
0x2f00006c	AHB_DMA_CCLM7_REG	CPU-S Master clock tokens
0x2f000090	AHB_DMA_VERSION_REG	Version ID (AHB DMA layer only)

Table 636: AHB_DMA_PL1_REG (0x2F000000)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL1 Arbitration priority for master RFMON. 0: Disables the master 1: The lowest ... 15: The highest	0xF

Table 637: AHB_DMA_PL2_REG (0x2F000004)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL2 Arbitration priority for master GEN-DMA. 0: Disables the master 1: The lowest	0xE

Bit	Mode	Symbol/Description	Reset
		... 15: The highest	

Table 638: AHB_DMA_PL3_REG (0x2F000008)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL3 Arbitration priority for master kDMA. 0: Disables the master 1: The lowest ... 15: The highest	0xD

Table 639: AHB_DMA_PL4_REG (0x2F00000C)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL4 Arbitration priority for master SDIO. 0: Disables the master 1: The lowest ... 15: The highest	0xC

Table 640: AHB_DMA_PL5_REG (0x2F000010)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL5 Arbitration priority for master SDeMMC. 0: Disables the master 1: The lowest ... 15: The highest	0xB

Table 641: AHB_DMA_PL6_REG (0x2F000014)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL6 Arbitration priority for master CC312	0xA

Bit	Mode	Symbol/Description	Reset
		0: Disables the master 1: The lowest ... 15: The highest	

Table 642: AHB_DMA_PL7_REG (0x2F000018)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_PL7 Arbitration priority for master CPU-S. 0: Disables the master 1: The lowest ... 15: The highest	0x9

Table 643: AHB_DMA_DFLT_MASTER_REG (0x2F000048)

Bit	Mode	Symbol/Description	Reset
31:4	R	- Reserved	0x0
3:0	R/W	AHB_DMA_DFLT_MASTER Default master ID number register. The default master is the master that is granted by the bus when no master has requested ownership. 0: Dummy master 1: RFMON 2: GP-DMA 3: kDMA 4: CPU-S	0x0

Table 644: AHB_DMA_WTEN_REG (0x2F00004C)

Bit	Mode	Symbol/Description	Reset
31:1	R	- Reserved	0x0
0	R/W	AHB_DMA_WTEN Weighted-token arbitration scheme enable.	0x0

Table 645: AHB_DMA_TCL_REG (0x2F000050)

Bit	Mode	Symbol/Description	Reset
31:16	R	- Reserved	0x0
15:0	R/W	AHB_DMA_TCL Master clock refresh period, counting clock cycles. An arbitration period is defined over this number of tokens. When a new arbitration period starts, the master	0xFFFF

Bit	Mode	Symbol/Description	Reset
		counters are reloaded. Recommended value is the sum of the AHB_DMA_CCLMx_REG valuesplus 2 tokens for each master.	

Table 646: AHB_DMA_CCLM1_REG (0x2F000054)

Bit	Mode	Symbol/Description	Reset
31:16	R	-	0x0
		Reserved	
15:0	R/W	AHB_DMA_CCLM	0xF
		Number of tokens (counted in AHB clock cycles) that a master can use on the bus before it has to arbitrate on a bus master with low priority and having tokens. Masters with tokens remaining have priority over masters that have used all of their tokens. User should configure all the token values ensuring that the sum does not exceeds the total allocated number of tokens. If a value of zero is configured, then the bus is deemed to have infinite tokens and will always operate in the upper-tier of arbitration.	

Table 647: AHB_DMA_CCLM2_REG (0x2F000058)

Bit	Mode	Symbol/Description	Reset
31:16	R	-	0x0
		Reserved	
15:0	R/W	AHB_DMA_CCLM	0xF
		Refer to AHB_DMA_CCLM1_REG	

Table 648: AHB_DMA_CCLM3_REG (0x2F00005C)

Bit	Mode	Symbol/Description	Reset
31:16	R	-	0x0
		Reserved	
15:0	R/W	AHB_DMA_CCLM	0xF
		Refer to AHB_DMA_CCLM1_REG	

Table 649: AHB_DMA_CCLM4_REG (0x2F000060)

Bit	Mode	Symbol/Description	Reset
31:16	R	-	0x0
		Reserved	
15:0	R/W	AHB_DMA_CCLM	0xF
		Refer to AHB_DMA_CCLM1_REG	

Table 650: AHB_DMA_CCLM5_REG (0x2F000064)

Bit	Mode	Symbol/Description	Reset
31:16	R	-	0x0
		Reserved	
15:0	R/W	AHB_DMA_CCLM	0xF

Bit	Mode	Symbol/Description	Reset
		Refer to AHB_DMA_CCLM1_REG	

Table 651: AHB_DMA_CCLM6_REG (0x2F000068)

Bit	Mode	Symbol/Description	Reset
31:16	R	- Reserved	0x0
15:0	R/W	AHB_DMA_CCLM Refer to AHB_DMA_CCLM1_REG	0xF

Table 652: AHB_DMA_CCLM7_REG (0x2F00006C)

Bit	Mode	Symbol/Description	Reset
31:16	R	- Reserved	0x0
15:0	R/W	AHB_DMA_CCLM Refer to AHB_DMA_CCLM1_REG	0xF

Table 653: AHB_DMA_VERSION_REG (0x2F000090)

Bit	Mode	Symbol/Description	Reset
31:0	R	AHB_DMA_VERSION	0x3231342A

10.26 AHB Arbitration Registers

Table 654: Register map SYSBUS_ICM

Address	Register	Description
0x40070400	QSPIRAM_ARB_REG	QSPIRAM ICM Priority level
0x40070404	AHBSYS_ARB_REG	AHBSYS Peri ICM Priority level
0x40070408	OTP_ARB_REG	OTP ICM Priority level

Table 655: QSPIRAM_ARB_REG (0x40070400)

Bit	Mode	Symbol/Description	Reset
1	R/W	QSPIRAM_AHB_CPUS_PRIO Priority AHB CPUS layer system bus 0x0: Highest priority 0x1: Second priority	0x1
0	R/W	QSPIRAM_AHB_DMA_PRIO Priority AHB DMA layer system bus 0x0: Highest priority 0x1: Second priority	0x0

Table 656: AHBSYS_ARB_REG (0x40070404)

Bit	Mode	Symbol/Description	Reset
1	R/W	AHBSYS_AHB_CPUS_PRIO Priority AHB CPUS layer system bus 0x0: Highest priority 0x1: Second priority	0x1
0	R/W	AHBSYS_AHB_DMA_PRIO Priority AHB DMA layer system bus 0x0: Highest priority 0x1: Second priority	0x0

Table 657: OTP_ARB_REG (0x40070408)

Bit	Mode	Symbol/Description	Reset
1	R/W	OTP_AHB_DMA_PRIO Priority AHB_DMA layer system bus 0x0: Highest priority 0x1: Second priority	0x1
0	R/W	OTP_AHB_CPUS_PRIO Priority AHB_CPUS layer system bus 0x0: Highest priority 0x1: Second priority	0x0

10.27 Timer Control Registers

Table 658: Register map TIMER

Address	Register	Description
0x40080000	TIMER_CTRL_REG	Timer control register
0x40080004	TIMER_TIMER_VAL_REG	Timer counter value
0x40080008	TIMER_STATUS_REG	Timer status register
0x4008000c	TIMER_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080010	TIMER_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080014	TIMER_SETTINGS_REG	Timer reload value and Delay in shot mode
0x40080018	TIMER_SHOTWIDTH_REG	Timer Shot duration in shot mode
0x4008001c	TIMER_PRE_SETTINGS_REG	Timer reload value and Delay in shot mode
0x40080020	TIMER_CAPTURE_GPIO1_REG	Timer value for event on GPIO1
0x40080024	TIMER_CAPTURE_GPIO2_REG	Timer value for event on GPIO2
0x40080028	TIMER_PRESCALER_VAL_REG	Timer prescaler counter value

Address	Register	Description
0x4008002c	TIMER_PWM_CTRL_REG	Timer pwm frequency register
0x40080030	TIMER_PULSE_GPIO_SEL_REG	Timer pulse counter ctrl register
0x40080034	TIMER_GPIO3_CONF_REG	Timer GPIO3 selection
0x40080038	TIMER_GPIO4_CONF_REG	Timer GPIO4 selection
0x4008003c	TIMER_CAPTURE_GPIO3_REG	Timer value for event on GPIO1
0x40080040	TIMER_CAPTURE_GPIO4_REG	Timer value for event on GPIO1
0x40080044	TIMER_PULSE_CNT_CTRL_REG	Timer pulse counter ctrl register
0x40080048	TIMER_ONESHOT_TRIGGER_REG	Timer oneshot trigger register
0x4008004c	TIMER_PWM_SYNC_REG	Timer pwm synchronisation register
0x40080050	TIMER_CLEAR_GPIO_EVENT_REG	Timer clear GPIO event register
0x40080054	TIMER_CLEAR_IRQ_REG	Timer clear interrupt
0x40080058	TIMER_CLEAR_IRQ_PULSE_REG	Timer clear pulse interrupt

Table 659: TIMER_CTRL_REG (0x40080000)

Bit	Mode	Symbol/Description	Reset
31:21	-	-	0x0
		Reserved	
20	R/W	TIM_SINGLE_EVENT_CAPTURE	0x0
		When this bit is set, only the first event on captimer1 is captured	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Select on which edge the edge detection should react 0: The counter is triggered on a rising edge 1: The counter is triggered on a falling edge Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Enable edge detection counter. Note: In sleep only 80 MHz can be reached at 900 mV	
17	R/W	TIM_ONESHOT_SWITCH	0x0
		Automatically switch after the completion of the pulse output without the CPU programming anything. 0: No automated switch from OneShot to Counter mode 1: Automated switch from OneShot to Counter mode and start counting down. In case no start value has been programmed (reload=0), the timer keeps generating interrupts until the timer clock is disabled	

Bit	Mode	Symbol/Description	Reset
16:15	R/W	TIM_ONESHOT_TRIGGER Oneshot trigger source 00: Select external GPIO as the trigger for one shot 01: Select a register write as the trigger of one shot 10: Either of the two triggers one shot 11: None of the two triggers one shot	0x0
14	R/W	TIM_CAP_GPIO4_IRQ_EN 0 = Event on GPIO4 does not create a CAPTIM interrrupt 1 = Event on GPIO4 creates a CAPTIM interrrupt	0x0
13	R/W	TIM_CAP_GPIO3_IRQ_EN 0 = Event on GPIO3 does not create a CAPTIM interrrupt 1 = Event on GPIO3 creates a CAPTIM interrrupt	0x0
12	R/W	TIM_CAP_GPIO2_IRQ_EN 0 = Event on GPIO2 does not create a CAPTIM interrrupt 1 = Event on GPIO2 creates a CAPTIM interrrupt	0x0
11	R/W	TIM_CAP_GPIO1_IRQ_EN 0 = Event on GPIO1 does not create a CAPTIM interrrupt 1 = Event on GPIO1 creates a CAPTIM interrrupt	0x0
10	R/W	TIM_IN4_EVENT_FALL_EN Event input 4 edge type 1 = Falling edge 0 = Rising edge	0x0
9	R/W	TIM_IN3_EVENT_FALL_EN Event input 3 edge type 1 = Falling edge 0 = Rising edge	0x0
8	R/W	TIM_CLK_EN Timer clock enable 1 = Clock enabled 0 = Clock disabled	0x0
7	R/W	TIM_SYS_CLK_EN Select clock 1 = Timer uses the DIVN clock 0 = Timer uses the IP clock Note: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8)	0x0
6	R/W	TIM_FREE_RUN_MODE_EN Valid when timer counts up, if it is 1 timer does not become zero when it reaches the reload value. It becomes zero only when it reaches the max value.	0x0
5	R/W	TIM_IRQ_EN Interrupt mask 1 = Timer IRQ is unmasked 0 = Timer IRQ is masked	0x0
4	R/W	TIM_IN2_EVENT_FALL_EN	0x0

Bit	Mode	Symbol/Description	Reset
		Event input 2 edge type 1 = Falling edge 0 = Rising edge	
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge type 1 = Falling edge 0 = Rising edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction 1 = Down 0 = Up Note: Only change counter direction when timer is not enabled	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode 1 = One shot enabled 0 = Counter enabled	0x0
0	R/W	TIM_EN Timer enable 1 = On 0 = Off	0x0

Table 660: TIMER_TIMER_VAL_REG (0x40080004)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE Gives the current timer value	0x0

Table 661: TIMER_STATUS_REG (0x40080008)

Bit	Mode	Symbol/Description	Reset
31:16	-	- Reserved	0x0
15	R	TIM_CAPTIM_DETECTED Indicates that a single event is detected when SINGLE_EVENT_CAPTURE is set to 1. Every next event does not update the capture timer register.	0x0
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13	R	TIM_IN4_STATE Gives the logic level of the IN4.	0x0
12	R	TIM_IN3_STATE Gives the logic level of the IN3.	0x0
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock was switched to divn clock.	0x0
10	R	TIM_PWM_BUSY	0x0

Bit	Mode	Symbol/Description	Reset
		Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	
9	R	TIM_TIMER_BUSY Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	0x0
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ occured, this bit is 1.	0x0
7	R	TIM_GPIO4_EVENT_PENDING When 1, GPIO4 event is pending.	0x0
6	R	TIM_GPIO3_EVENT_PENDING When 1, GPIO3 event is pending.	0x0
5	R	TIM_GPIO2_EVENT_PENDING When 1, GPIO2 event is pending.	0x0
4	R	TIM_GPIO1_EVENT_PENDING When 1, GPIO1 event is pending.	0x0
3:2	R	TIM_ONESHOT_PHASE OneShot phase 0 = Wait for event 1 = Delay phase 2 = Start Shot 3 = Shot phase	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 662: TIMER_GPIO1_CONF_REG (0x4008000C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09	0x0

Bit	Mode	Symbol/Description	Reset
		11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 663: TIMER_GPIO2_CONF_REG (0x40080010)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02	0x0

Bit	Mode	Symbol/Description	Reset
		18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 664: TIMER_SETTINGS_REG (0x40080014)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode, Delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles)	0x0

Table 665: TIMER_SHOTWIDTH_REG (0x40080018)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in oneshot mode	0x0

Table 666: TIMER_PRE_SETTINGS_REG (0x4008001C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency / (TIM_PRESCALER+1)	0x0

Table 667: TIMER_CAPTURE_GPIO1_REG (0x40080020)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12 Gives the Capture time for event on GPIO1	0x0

Table 668: TIMER_CAPTURE_GPIO2_REG (0x40080024)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22	0x0

Bit	Mode	Symbol/Description	Reset
		Gives the Capture time for event on GPIO2	

Table 669: TIMER_PRESCALER_VAL_REG (0x40080028)

Bit	Mode	Symbol/Description	Reset
31:5	-	-	0x0
		Reserved	
4:0	R	TIM_PRESCALER_VAL	0x0
		Gives the current prescaler counter value	

Table 670: TIMER_PWM_CTRL_REG (0x4008002C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC	0x0
		Defines the PWM duty cycle. TIM_PWM_DC / (TIM_PWM_FREQ+1)	
15:0	R/W	TIM_PWM_FREQ	0x0
		Defines the PWM frequency. Timer clock frequency / (TIM_PWM_FREQ+1) Timer clock is clock after prescaler	

Table 671: TIMER_PULSE_GPIO_SEL_REG (0x40080030)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2	0x0
		Select one of the 32 GPIOs as input for the pulse counter, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07	

Bit	Mode	Symbol/Description	Reset
		23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG	

Table 672: TIMER_GPIO3_CONF_REG (0x40080034)

Bit	Mode	Symbol/Description	Reset
31:6	-	-	0x0
		Reserved	
5:0	R/W	TIM_GPIO3_CONF	0x0
		Select one of the 32 GPIOs as IN3, Valid value 0-32. 1 for the first GPIO 32 for the last GPIO 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11	

Bit	Mode	Symbol/Description	Reset
		27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 673: TIMER_GPIO4_CONF_REG (0x40080038)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO4_CONF Select one of the 32 GPIOs as IN4, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	0x0

Table 674: TIMER_CAPTURE_GPIO3_REG (0x4008003C)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO32 Gives the Capture time for event on GPIO3	0x0

Table 675: TIMER_CAPTURE_GPIO4_REG (0x40080040)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO42 Gives the Capture time for event on GPIO4	0x0

Table 676: TIMER_PULSE_CNT_CTRL_REG (0x40080044)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an irq is fired for the pulse counter n-2. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG	0x0

Table 677: TIMER_ONESHOT_TRIGGER_REG (0x40080048)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_ONESHOT_TRIGGER_SW Trigger oneshot	0x0

Table 678: TIMER_PWM_SYNC_REG (0x4008004C)

Bit	Mode	Symbol/Description	Reset
6	R/W	TIMER6_SYNC Enable PWM start synchronisation of timer6	0x0
5	R/W	TIMER4_SYNC Enable PWM start synchronisation of timer4	0x0
4	R/W	TIMER3_SYNC Enable PWM start synchronisation of timer3	0x0
3	R/W	TIMER2_SYNC Enable PWM start synchronisation of timer2	0x0
2	R/W	TIMER_SYNC Enable PWM synchronisation of timer	0x0
1	R/W	SYNC_ENABLE Enable PWM start synchronisation of the selected timers	0x0
0	R/W	PWM_START Start PWM of the selected timers	0x0

Table 679: TIMER_CLEAR_GPIO_EVENT_REG (0x40080050)

Bit	Mode	Symbol/Description	Reset
3	R0/W	TIM_CLEAR_GPIO4_EVENT	0x0

Bit	Mode	Symbol/Description	Reset
		1 = Clear GPIO4 event. Return always 0	
2	R0/W	TIM_CLEAR_GPIO3_EVENT	0x0
		1 = Clear GPIO3 event. Return always 0	
1	R0/W	TIM_CLEAR_GPIO2_EVENT	0x0
		1 = Clear GPIO2 event. Return always 0	
0	R0/W	TIM_CLEAR_GPIO1_EVENT	0x0
		1 = Clear GPIO1 event. Return always 0	

Table 680: TIMER_CLEAR_IRQ_REG (0x40080054)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ	0x0
		Write any value clear interrupt	

Table 681: TIMER_CLEAR_IRQ_PULSE_REG (0x40080058)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ	0x0
		Write any value will clear irq pulse interrupt	

10.28 Timer2 Control Registers

Table 682: Register map TIMER2

Address	Register	Description
0x40080100	TIMER2_CTRL_REG	Timer Control Register
0x40080104	TIMER2_TIMER_VAL_REG	Timer Counter Value
0x40080108	TIMER2_STATUS_REG	Timer Status Register
0x4008010c	TIMER2_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080110	TIMER2_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080114	TIMER2_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080118	TIMER2_SHOTWIDTH_REG	Timer Shot Duration in Shot mode
0x4008011c	TIMER2_PRE_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080120	TIMER2_CAPTURE_GPIO1_REG	Timer Value for Event on GPIO1
0x40080124	TIMER2_CAPTURE_GPIO2_REG	Timer Value for Event on GPIO2
0x40080128	TIMER2_PRESCALER_VAL_REG	Timer Prescaler Counter Value
0x4008012c	TIMER2_PWM_CTRL_REG	Timer PWM Frequency Register

Address	Register	Description
0x40080130	TIMER2_PULSE_GPIO_SEL_REG	Timer Pulse Counter CTRL Register
0x40080144	TIMER2_PULSE_CNT_CTRL_REG	Timer Pulse Counter CTRL Register
0x40080154	TIMER2_CLEAR_IRQ_REG	Timer Clear Interrupt
0x40080158	TIMER2_CLEAR_IRQ_PULSE_REG	Timer Clear Pulse Interrupt

Table 683: TIMER2_CTRL_REG (0x40080100)

Bit	Mode	Symbol/Description	Reset
31:20	-	-	0x0
		Reserved	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Counter trigger edge selection: 0: The counter is triggered on a rising edge. 1: The counter is triggered on a falling edge. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG.	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Edge detection counter: 0: Disable 1: Enable	
17:9	-	-	0x0
		Reserved	
8	R/W	TIM_CLK_EN	0x0
		Timer clock enable: 0: Clock is disabled. 1: Clock is enabled.	
7	R/W	TIM_SYS_CLK_EN	0x0
		Timer clock selection: 0: LP_CLK clock 1: DIVN clock NOTE: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8).	
6	R/W	TIM_FREE_RUN_MODE_EN	0x0
		Valid when timer counts up, if it is 1 timer does not zero when reaches to reload value. it becomes zero only when it reaches the max value.	
5	R/W	TIM_IRQ_EN	0x0
		Interrupt mask: 0: Timer IRQ is disable. 1: Timer IRQ is enable.	
4	R/W	TIM_IN2_EVENT_FALL_EN	0x0
		Event input 2 edge selection: 0: Rising edge 1: Falling edge	

Bit	Mode	Symbol/Description	Reset
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge selection: 0: Rising edge 1: Falling edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction: 0: Up 1: Down NOTE: Only change this bit when timer is disabled.	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode: 0: Counter is enabled. 1: One shot is enabled.	0x0
0	R/W	TIM_EN Timer enable: 0: Off 1: On	0x0

Table 684: TIMER2_TIMER_VAL_REG (0x40080104)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE2 Gives the current timer value.	0x0

Table 685: TIMER2_STATUS_REG (0x40080108)

Bit	Mode	Symbol/Description	Reset
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13:12	-	- Reserved	0x0
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock is switched to divn clock.	0x0
10	R	TIM_PWM_BUSY Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	0x0
9	R	TIM_TIMER_BUSY Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	0x0
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ has occured, this bit is 1.	0x0
7:4	-	- Reserved	0x0

Bit	Mode	Symbol/Description	Reset
3:2	R	TIM_ONESHOT_PHASE Oneshot phase: 0: Wait for event. 1: Delay phase. 2: Start shot. 3: Shot phase.	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 686: TIMER2_GPIO1_CONF_REG (0x4008010C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13	0x0

Bit	Mode	Symbol/Description	Reset
		29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 687: TIMER2_GPIO2_CONF_REG (0x40080110)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	0x0

Table 688: TIMER2_SETTINGS_REG (0x40080114)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode; delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles).	0x0

Table 689: TIMER2_SHOTWIDTH_REG (0x40080118)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in Oneshot mode.	0x0

Table 690: TIMER2_PRE_SETTINGS_REG (0x4008011C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency/(TIM_PRESCALER+1).	0x0

Table 691: TIMER2_CAPTURE_GPIO1_REG (0x40080120)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12 Gives the capture time for event on GPIO1.	0x0

Table 692: TIMER2_CAPTURE_GPIO2_REG (0x40080124)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22 Gives the capture time for event on GPIO2.	0x0

Table 693: TIMER2_PRESCALER_VAL_REG (0x40080128)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R	TIM_PRESCALER_VAL Gives the current prescaler counter value.	0x0

Table 694: TIMER2_PWM_CTRL_REG (0x4008012C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC Defines the PWM duty cycle. TIM_PWM_DC/(TIM_PWM_FREQ+1).	0x0
15:0	R/W	TIM_PWM_FREQ Defines the PWM frequency. Timer clock frequency/(TIM_PWM_FREQ+1).	0x0

Bit	Mode	Symbol/Description	Reset
		Timer clock is the clock after prescaler.	

Table 695: TIMER2_PULSE_GPIO_SEL_REG (0x40080130)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2 Select one of the 32 GPIOs as input for the pulse counter, valid values 0-32: 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 696: TIMER2_PULSE_CNT_CTRL_REG (0x40080144)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an IRQ is fired for the pulse counter. NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 697: TIMER2_CLEAR_IRQ_REG (0x40080154)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ Write any value to clear interrupt.	0x0

Table 698: TIMER2_CLEAR_IRQ_PULSE_REG (0x40080158)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ Write any value to clear IRQ pulse interrupt.	0x0

10.29 Timer3 Control Registers

Table 699: Register map TIMER3

Address	Register	Description
0x40080200	TIMER3_CTRL_REG	Timer Control Register
0x40080204	TIMER3_TIMER_VAL_REG	Timer Counter Value
0x40080208	TIMER3_STATUS_REG	Timer Status Register
0x4008020c	TIMER3_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080210	TIMER3_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080214	TIMER3_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080218	TIMER3_SHOTWIDTH_REG	Timer Shot Duration in Shot mode
0x4008021c	TIMER3_PRE_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080220	TIMER3_CAPTURE_GPIO1_REG	Timer Value for Event on GPIO1
0x40080224	TIMER3_CAPTURE_GPIO2_REG	Timer Value for Event on GPIO2
0x40080228	TIMER3_PRESCALER_VAL_REG	Timer Prescaler Counter Value
0x4008022c	TIMER3_PWM_CTRL_REG	Timer PWM Frequency Register
0x40080230	TIMER3_PULSE_GPIO_SEL_REG	Timer Pulse Counter CTRL Register
0x40080244	TIMER3_PULSE_CNT_CTRL_REG	Timer Pulse Counter CTRL Register
0x40080254	TIMER3_CLEAR_IRQ_REG	Timer Clear Interrupt
0x40080258	TIMER3_CLEAR_IRQ_PULSE_REG	Timer Clear Pulse Interrupt

Table 700: TIMER3_CTRL_REG (0x40080200)

Bit	Mode	Symbol/Description	Reset
31:20	-	-	0x0
		Reserved	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Counter trigger edge selection: 0: The counter is triggered on a rising edge. 1: The counter is triggered on a falling edge. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG.	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Edge detection counter: 0: Disable 1: Enable	
17:9	-	-	0x0
		Reserved	
8	R/W	TIM_CLK_EN	0x0
		Timer clock enable: 0: Clock is disabled. 1: Clock is enabled.	
7	R/W	TIM_SYS_CLK_EN	0x0
		Timer clock selection: 0: LP_CLK clock 1: DIVN clock NOTE: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8).	
6	R/W	TIM_FREE_RUN_MODE_EN	0x0
		Valid when timer counts up, if it is 1 timer does not zero when reaches to reload value. it becomes zero only when it reaches the max value.	
5	R/W	TIM_IRQ_EN	0x0
		Interrupt mask: 0: Timer IRQ is disable. 1: Timer IRQ is enable.	
4	R/W	TIM_IN2_EVENT_FALL_EN	0x0
		Event input 2 edge selection: 0: Rising edge 1: Falling edge	
3	R/W	TIM_IN1_EVENT_FALL_EN	0x0
		Event input 1 edge selection: 0: Rising edge 1: Falling edge	
2	R/W	TIM_COUNT_DOWN_EN	0x0
		Timer count direction: 0: Up 1: Down NOTE: Only change this bit when timer is disabled.	

Bit	Mode	Symbol/Description	Reset
1	R/W	TIM_ONESHOT_MODE_EN Timer mode: 0: Counter is enabled. 1: One shot is enabled.	0x0
0	R/W	TIM_EN Timer enable: 0: Off 1: On	0x0

Table 701: TIMER3_TIMER_VAL_REG (0x40080204)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE2 Gives the current timer value.	0x0

Table 702: TIMER3_STATUS_REG (0x40080208)

Bit	Mode	Symbol/Description	Reset
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13:12	-	- Reserved	0x0
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock is switched to divn clock.	0x0
10	R	TIM_PWM_BUSY Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	0x0
9	R	TIM_TIMER_BUSY Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	0x0
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ has occured, this bit is 1.	0x0
7:4	-	- Reserved	0x0
3:2	R	TIM_ONESHOT_PHASE Oneshot phase: 0: Wait for event. 1: Delay phase. 2: Start shot. 3: Shot phase.	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE	0x0

Bit	Mode	Symbol/Description	Reset
		Gives the logic level of the IN2.	

Table 703: TIMER3_GPIO1_CONF_REG (0x4008020C)

Bit	Mode	Symbol/Description	Reset
31:6	-	-	0x0
		Reserved	
5:0	R/W	TIM_GPIO1_CONF	0x0
		Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 704: TIMER3_GPIO2_CONF_REG (0x40080210)

Bit	Mode	Symbol/Description	Reset
31:6	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	0x0

Table 705: TIMER3_SETTINGS_REG (0x40080214)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode; delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles).	0x0

Table 706: TIMER3_SHOTWIDTH_REG (0x40080218)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2	0x0

Bit	Mode	Symbol/Description	Reset
		Shot phase duration in Oneshot mode.	

Table 707: TIMER3_PRE_SETTINGS_REG (0x4008021C)

Bit	Mode	Symbol/Description	Reset
31:5	-	-	0x0
		Reserved	
4:0	R/W	TIM_PRESCALER2	0x0
		Defines the timer count frequency. CLOCK frequency/(TIM_PRESCALER+1).	

Table 708: TIMER3_CAPTURE_GPIO1_REG (0x40080220)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12	0x0
		Gives the capture time for event on GPIO1.	

Table 709: TIMER3_CAPTURE_GPIO2_REG (0x40080224)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22	0x0
		Gives the capture time for event on GPIO2.	

Table 710: TIMER3_PRESCALER_VAL_REG (0x40080228)

Bit	Mode	Symbol/Description	Reset
31:5	-	-	0x0
		Reserved	
4:0	R	TIM_PRESCALER_VAL	0x0
		Gives the current prescaler counter value.	

Table 711: TIMER3_PWM_CTRL_REG (0x4008022C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC	0x0
		Defines the PWM duty cycle. TIM_PWM_DC/(TIM_PWM_FREQ+1).	
15:0	R/W	TIM_PWM_FREQ	0x0
		Defines the PWM frequency. Timer clock frequency/(TIM_PWM_FREQ+1). Timer clock is the clock after prescaler.	

Table 712: TIMER3_PULSE_GPIO_SEL_REG (0x40080230)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2	0x0
		Select one of the 32 GPIOs as input for the pulse counter, valid values 0-32: 0: Disable input 1: P0_00	

Bit	Mode	Symbol/Description	Reset
		2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	

Table 713: TIMER3_PULSE_CNT_CTRL_REG (0x40080244)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an IRQ is fired for the pulse counter. NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 714: TIMER3_CLEAR_IRQ_REG (0x40080254)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ Write any value to clear interrupt.	0x0

Table 715: TIMER3_CLEAR_IRQ_PULSE_REG (0x40080258)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ Write any value to clear IRQ pulse interrupt.	0x0

10.30 Timer4 Control Registers

Table 716: Register map TIMER4

Address	Register	Description
0x40080300	TIMER4_CTRL_REG	Timer Control Register
0x40080304	TIMER4_TIMER_VAL_REG	Timer Counter Value
0x40080308	TIMER4_STATUS_REG	Timer Status Register
0x4008030c	TIMER4_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080310	TIMER4_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080314	TIMER4_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080318	TIMER4_SHOTWIDTH_REG	Timer Shot Duration in Shot mode
0x4008031c	TIMER4_PRE_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080320	TIMER4_CAPTURE_GPIO1_REG	Timer Value for Event on GPIO1
0x40080324	TIMER4_CAPTURE_GPIO2_REG	Timer Value for Event on GPIO2
0x40080328	TIMER4_PRESCALER_VAL_REG	Timer Prescaler Counter Value
0x4008032c	TIMER4_PWM_CTRL_REG	Timer PWM Frequency Register
0x40080330	TIMER4_PULSE_GPIO_SEL_REG	Timer Pulse Counter CTRL Register
0x40080344	TIMER4_PULSE_CNT_CTRL_REG	Timer Pulse Counter CTRL Register
0x40080354	TIMER4_CLEAR_IRQ_REG	Timer Clear Interrupt
0x40080358	TIMER4_CLEAR_IRQ_PULSE_REG	Timer Clear Pulse Interrupt

Table 717: TIMER4_CTRL_REG (0x40080300)

Bit	Mode	Symbol/Description	Reset
31:20	-	- Reserved	0x0
19	R/W	TIM_EDGE_DET_CNT_FALL_EN Counter trigger edge selection: 0: The counter is triggered on a rising edge.	0x0

Bit	Mode	Symbol/Description	Reset
		1: The counter is triggered on a falling edge. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG.	
18	R/W	TIM_EDGE_DET_CNT_EN Edge detection counter: 0: Disable 1: Enable	0x0
17:9	-	- Reserved	0x0
8	R/W	TIM_CLK_EN Timer clock enable: 0: Clock is disabled. 1: Clock is enabled.	0x0
7	R/W	TIM_SYS_CLK_EN Timer clock selection: 0: LP_CLK clock 1: DIVN clock NOTE: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8).	0x0
6	R/W	TIM_FREE_RUN_MODE_EN Valid when timer counts up, if it is 1 timer does not zero when reaches to reload value. it becomes zero only when it reaches the max value.	0x0
5	R/W	TIM_IRQ_EN Interrupt mask: 0: Timer IRQ is disable. 1: Timer IRQ is enable.	0x0
4	R/W	TIM_IN2_EVENT_FALL_EN Event input 2 edge selection: 0: Rising edge 1: Falling edge	0x0
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge selection: 0: Rising edge 1: Falling edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction: 0: Up 1: Down NOTE: Only change this bit when timer is disabled.	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode: 0: Counter is enabled. 1: One shot is enabled.	0x0
0	R/W	TIM_EN Timer enable: 0: Off	0x0

Bit	Mode	Symbol/Description	Reset
		1: On	

Table 718: TIMER4_TIMER_VAL_REG (0x40080304)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE2 Gives the current timer value.	0x0

Table 719: TIMER4_STATUS_REG (0x40080308)

Bit	Mode	Symbol/Description	Reset
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13:12	-	- Reserved	0x0
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock is switched to divn clock.	0x0
10	R	TIM_PWM_BUSY Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	0x0
9	R	TIM_TIMER_BUSY Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	0x0
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ has occured, this bit is 1.	0x0
7:4	-	- Reserved	0x0
3:2	R	TIM_ONESHOT_PHASE Oneshot phase: 0: Wait for event. 1: Delay phase. 2: Start shot. 3: Shot phase.	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 720: TIMER4_GPIO1_CONF_REG (0x4008030C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0

Bit	Mode	Symbol/Description	Reset
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	0x0

Table 721: TIMER4_GPIO2_CONF_REG (0x40080310)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02	0x0

Bit	Mode	Symbol/Description	Reset
		4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 722: TIMER4_SETTINGS_REG (0x40080314)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode; delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles).	0x0

Table 723: TIMER4_SHOTWIDTH_REG (0x40080318)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in Oneshot mode.	0x0

Table 724: TIMER4_PRE_SETTINGS_REG (0x4008031C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0

Bit	Mode	Symbol/Description	Reset
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency/(TIM_PRESCALER+1).	0x0

Table 725: TIMER4_CAPTURE_GPIO1_REG (0x40080320)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12 Gives the capture time for event on GPIO1.	0x0

Table 726: TIMER4_CAPTURE_GPIO2_REG (0x40080324)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22 Gives the capture time for event on GPIO2.	0x0

Table 727: TIMER4_PRESCALER_VAL_REG (0x40080328)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R	TIM_PRESCALER_VAL Gives the current prescaler counter value.	0x0

Table 728: TIMER4_PWM_CTRL_REG (0x4008032C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC Defines the PWM duty cycle. TIM_PWM_DC/(TIM_PWM_FREQ+1).	0x0
15:0	R/W	TIM_PWM_FREQ Defines the PWM frequency. Timer clock frequency/(TIM_PWM_FREQ+1). Timer clock is the clock after prescaler.	0x0

Table 729: TIMER4_PULSE_GPIO_SEL_REG (0x40080330)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2 Select one of the 32 GPIOs as input for the pulse counter, valid values 0-32: 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07	0x0

Bit	Mode	Symbol/Description	Reset
		9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	

Table 730: TIMER4_PULSE_CNT_CTRL_REG (0x40080344)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an IRQ is fired for the pulse counter. NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 731: TIMER4_CLEAR_IRQ_REG (0x40080354)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ Write any value to clear interrupt.	0x0

Table 732: TIMER4_CLEAR_IRQ_PULSE_REG (0x40080358)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ Write any value to clear IRQ pulse interrupt.	0x0

10.31 Timer5 Control Registers

Table 733: Register map TIMER5

Address	Register	Description
0x40080400	TIMER5_CTRL_REG	Timer control register
0x40080404	TIMER5_TIMER_VAL_REG	Timer counter value
0x40080408	TIMER5_STATUS_REG	Timer status register
0x4008040c	TIMER5_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080410	TIMER5_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080414	TIMER5_SETTINGS_REG	Timer reload value and Delay in shot mode
0x40080418	TIMER5_SHOTWIDTH_REG	Timer Shot duration in shot mode
0x4008041c	TIMER5_PRE_SETTINGS_REG	Timer reload value and Delay in shot mode
0x40080420	TIMER5_CAPTURE_GPIO1_REG	Timer value for event on GPIO1
0x40080424	TIMER5_CAPTURE_GPIO2_REG	Timer value for event on GPIO2
0x40080428	TIMER5_PRESCALER_VAL_REG	Timer prescaler counter value
0x4008042c	TIMER5_PWM_CTRL_REG	Timer pwm frequency register
0x40080430	TIMER5_PULSE_GPIO_SEL_REG	Timer pulse counter ctrl register
0x40080434	TIMER5_GPIO3_CONF_REG	Timer GPIO3 selection
0x40080438	TIMER5_GPIO4_CONF_REG	Timer GPIO4 selection
0x4008043c	TIMER5_CAPTURE_GPIO3_REG	Timer value for event on GPIO1
0x40080440	TIMER5_CAPTURE_GPIO4_REG	Timer value for event on GPIO1
0x40080444	TIMER5_PULSE_CNT_CTRL_REG	Timer pulse counter ctrl register
0x40080448	TIMER5_ONESHOT_TRIGGER_REG	Timer oneshot trigger register
0x4008044c	TIMER5_PWM_SYNC_REG	Timer pwm synchronisation register
0x40080450	TIMER5_CLEAR_GPIO_EVENT_REG	Timer clear GPIO event register
0x40080454	TIMER5_CLEAR_IRQ_REG	Timer clear interrupt
0x40080458	TIMER5_CLEAR_IRQ_PULSE_REG	Timer clear pulse interrupt

Table 734: TIMER5_CTRL_REG (0x40080400)

Bit	Mode	Symbol/Description	Reset
31:21	-	-	0x0
		Reserved	
20	R/W	TIM_SINGLE_EVENT_CAPTURE	0x0
		When this bit is set, only the first event on captimer1 is captured	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Select on which edge the edge detection should react 0: The counter is triggered on a rising edge 1: The counter is triggered on a falling edge Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Enable edge detection counter. Note: In sleep only 80 MHz can be reached at 900 mV	
17	R/W	TIM_ONESHOT_SWITCH	0x0
		Automatically switch after the completion of the pulse output without the CPU programming anything. 0: No automated switch from OneShot to Counter mode 1: Automated switch from OneShot to Counter mode and start counting down. In case no start value has been programmed (reload=0), the timer keeps generating interrupts until the timer clock is disabled	
16:15	R/W	TIM_ONESHOT_TRIGGER	0x0
		Oneshot trigger source 00: Select external GPIO as the trigger for one shot 01: Select a register write as the trigger of one shot 10: Either of the two triggers one shot 11: None of the two triggers one shot	
14	R/W	TIM_CAP_GPIO4_IRQ_EN	0x0
		0 = Event on GPIO4 does not create a CAPTIM interrrupt 1 = Event on GPIO4 creates a CAPTIM interrrupt	
13	R/W	TIM_CAP_GPIO3_IRQ_EN	0x0
		0 = Event on GPIO3 does not create a CAPTIM interrrupt 1 = Event on GPIO3 creates a CAPTIM interrrupt	
12	R/W	TIM_CAP_GPIO2_IRQ_EN	0x0
		0 = Event on GPIO2 does not create a CAPTIM interrrupt 1 = Event on GPIO2 creates a CAPTIM interrrupt	
11	R/W	TIM_CAP_GPIO1_IRQ_EN	0x0
		0 = Event on GPIO1 does not create a CAPTIM interrrupt 1 = Event on GPIO1 creates a CAPTIM interrrupt	
10	R/W	TIM_IN4_EVENT_FALL_EN	0x0
		Event input 4 edge type 1 = Falling edge 0 = Rising edge	
9	R/W	TIM_IN3_EVENT_FALL_EN	0x0
		Event input 3 edge type	

Bit	Mode	Symbol/Description	Reset
		1 = Falling edge 0 = Rising edge	
8	R/W	TIM_CLK_EN Timer clock enable 1 = Clock enabled 0 = Clock disabled	0x0
7	R/W	TIM_SYS_CLK_EN Select clock 1 = Timer uses the DIVN clock 0 = Timer uses the IP clock Note: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8)	0x0
6	R/W	TIM_FREE_RUN_MODE_EN Valid when timer counts up, if it is 1 timer does not become zero when it reaches the reload value. It becomes zero only when it reaches the max value.	0x0
5	R/W	TIM_IRQ_EN Interrupt mask 1 = Timer IRQ is unmasked 0 = Timer IRQ is masked	0x0
4	R/W	TIM_IN2_EVENT_FALL_EN Event input 2 edge type 1 = Falling edge 0 = Rising edge	0x0
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge type 1 = Falling edge 0 = Rising edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction 1 = Down 0 = Up Note: Only change counter direction when timer is not enabled	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode 1 = One shot enabled 0 = Counter enabled	0x0
0	R/W	TIM_EN Timer enable 1 = On 0 = Off	0x0

Table 735: TIMER5_TIMER_VAL_REG (0x40080404)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE	0x0

Bit	Mode	Symbol/Description	Reset
		Gives the current timer value	

Table 736: TIMER5_STATUS_REG (0x40080408)

Bit	Mode	Symbol/Description	Reset
31:16	-	-	0x0
		Reserved	
15	R	TIM_CAPTIM_DETECTED	0x0
		Indicates that a single event is detected when SINGLE_EVENT_CAPTURE is set to 1. Every next event does not update the capture timer register.	
14	R	TIM_IRQ_PULSE_STATUS	0x0
		Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	
13	R	TIM_IN4_STATE	0x0
		Gives the logic level of the IN4.	
12	R	TIM_IN3_STATE	0x0
		Gives the logic level of the IN3.	
11	R	TIM_SWITCHED_TO_DIVN_CLK	0x0
		Indicates that timer clock was switched to divn clock.	
10	R	TIM_PWM_BUSY	0x0
		Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	
9	R	TIM_TIMER_BUSY	0x0
		Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	
8	R	TIM_IRQ_STATUS	0x0
		IRQ status bit. When an IRQ occured, this bit is 1.	
7	R	TIM_GPIO4_EVENT_PENDING	0x0
		When 1, GPIO4 event is pending.	
6	R	TIM_GPIO3_EVENT_PENDING	0x0
		When 1, GPIO3 event is pending.	
5	R	TIM_GPIO2_EVENT_PENDING	0x0
		When 1, GPIO2 event is pending.	
4	R	TIM_GPIO1_EVENT_PENDING	0x0
		When 1, GPIO1 event is pending.	
3:2	R	TIM_ONESHOT_PHASE	0x0
		OneShot phase 0 = Wait for event 1 = Delay phase 2 = Start Shot 3 = Shot phase	
1	R	TIM_IN2_STATE	0x0
		Gives the logic level of the IN1.	

Bit	Mode	Symbol/Description	Reset
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 737: TIMER5_GPIO1_CONF_REG (0x4008040C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	0x0

Table 738: TIMER5_GPIO2_CONF_REG (0x40080410)

Bit	Mode	Symbol/Description	Reset
31:6	-	-	0x0
		Reserved	
5:0	R/W	TIM_GPIO2_CONF	0x0
		Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 739: TIMER5_SETTINGS_REG (0x40080414)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2	0x0
		Reload or max value in timer mode, Delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles)	

Table 740: TIMER5_SHOTWIDTH_REG (0x40080418)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in oneshot mode	0x0

Table 741: TIMER5_PRE_SETTINGS_REG (0x4008041C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency / (TIM_PRESCALER+1)	0x0

Table 742: TIMER5_CAPTURE_GPIO1_REG (0x40080420)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12 Gives the Capture time for event on GPIO1	0x0

Table 743: TIMER5_CAPTURE_GPIO2_REG (0x40080424)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22 Gives the Capture time for event on GPIO2	0x0

Table 744: TIMER5_PRESCALER_VAL_REG (0x40080428)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R	TIM_PRESCALER_VAL Gives the current prescaler counter value	0x0

Table 745: TIMER5_PWM_CTRL_REG (0x4008042C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC Defines the PWM duty cycle. TIM_PWM_DC / (TIM_PWM_FREQ+1)	0x0
15:0	R/W	TIM_PWM_FREQ Defines the PWM frequency. Timer clock frequency / (TIM_PWM_FREQ+1) Timer clock is clock after prescaler	0x0

Table 746: TIMER5_PULSE_GPIO_SEL_REG (0x40080430)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2 Select one of the 32 GPIOs as input for the pulse counter, Valid values 0-32.	0x0

Bit	Mode	Symbol/Description	Reset
		0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG	

Table 747: TIMER5_GPIO3_CONF_REG (0x40080434)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO3_CONF Select one of the 32 GPIOs as IN3, Valid value 0-32. 1 for the first GPIO 32 for the last GPIO 0: Disable input 1: P0_00 2: P0_01 3: P0_02	0x0

Bit	Mode	Symbol/Description	Reset
		4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 748: TIMER5_GPIO4_CONF_REG (0x40080438)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO4_CONF Select one of the 32 GPIOs as IN4, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09	0x0

Bit	Mode	Symbol/Description	Reset
		11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 749: TIMER5_CAPTURE_GPIO3_REG (0x4008043C)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO32 Gives the Capture time for event on GPIO3	0x0

Table 750: TIMER5_CAPTURE_GPIO4_REG (0x40080440)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO42 Gives the Capture time for event on GPIO4	0x0

Table 751: TIMER5_PULSE_CNT_CTRL_REG (0x40080444)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an irq is fired for the pulse counter n-2. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG	0x0

Table 752: TIMER5_ONESHOT_TRIGGER_REG (0x40080448)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_ONESHOT_TRIGGER_SW Trigger oneshot	0x0

Table 753: TIMER5_PWM_SYNC_REG (0x4008044C)

Bit	Mode	Symbol/Description	Reset
6	R/W	TIMER6_SYNC	0x0
		Enable PWM start synchronisation of timer6	
5	R/W	TIMER4_SYNC	0x0
		Enable PWM start synchronisation of timer4	
4	R/W	TIMER3_SYNC	0x0
		Enable PWM start synchronisation of timer3	
3	R/W	TIMER2_SYNC	0x0
		Enable PWM start synchronisation of timer2	
2	R/W	TIMER_SYNC	0x0
		Enable PWM synchronisation of timer	
1	R/W	SYNC_ENABLE	0x0
		Enable PWM start synchronisation of the selected timers	
0	R/W	PWM_START	0x0
		Start PWM of the selected timers	

Table 754: TIMER5_CLEAR_GPIO_EVENT_REG (0x40080450)

Bit	Mode	Symbol/Description	Reset
3	R0/W	TIM_CLEAR_GPIO4_EVENT	0x0
		1 = Clear GPIO4 event. Return always 0	
2	R0/W	TIM_CLEAR_GPIO3_EVENT	0x0
		1 = Clear GPIO3 event. Return always 0	
1	R0/W	TIM_CLEAR_GPIO2_EVENT	0x0
		1 = Clear GPIO2 event. Return always 0	
0	R0/W	TIM_CLEAR_GPIO1_EVENT	0x0
		1 = Clear GPIO1 event. Return always 0	

Table 755: TIMER5_CLEAR_IRQ_REG (0x40080454)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ	0x0
		Write any value clear interrupt	

Table 756: TIMER5_CLEAR_IRQ_PULSE_REG (0x40080458)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ	0x0
		Write any value will clear irq pulse interrupt	

10.32 Timer6 Registers

Table 757: Register map TIMER6

Address	Register	Description
0x40080500	TIMER6_CTRL_REG	Timer Control Register
0x40080504	TIMER6_TIMER_VAL_REG	Timer Counter Value
0x40080508	TIMER6_STATUS_REG	Timer Status Register
0x4008050c	TIMER6_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080510	TIMER6_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080514	TIMER6_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080518	TIMER6_SHOTWIDTH_REG	Timer Shot Duration in Shot mode
0x4008051c	TIMER6_PRE_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080520	TIMER6_CAPTURE_GPIO1_REG	Timer Value for Event on GPIO1
0x40080524	TIMER6_CAPTURE_GPIO2_REG	Timer Value for Event on GPIO2
0x40080528	TIMER6_PRESCALER_VAL_REG	Timer Prescaler Counter Value
0x4008052c	TIMER6_PWM_CTRL_REG	Timer PWM Frequency Register
0x40080530	TIMER6_PULSE_GPIO_SEL_REG	Timer Pulse Counter CTRL Register
0x40080544	TIMER6_PULSE_CNT_CTRL_REG	Timer Pulse Counter CTRL Register
0x40080554	TIMER6_CLEAR_IRQ_REG	Timer Clear Interrupt
0x40080558	TIMER6_CLEAR_IRQ_PULSE_REG	Timer Clear Pulse Interrupt

Table 758: TIMER6_CTRL_REG (0x40080500)

Bit	Mode	Symbol/Description	Reset
31:20	-	-	0x0
		Reserved	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Counter trigger edge selection: 0: The counter is triggered on a rising edge. 1: The counter is triggered on a falling edge. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG.	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Edge detection counter: 0: Disable 1: Enable	

Bit	Mode	Symbol/Description	Reset
17:9	-	- Reserved	0x0
8	R/W	TIM_CLK_EN Timer clock enable: 0: Clock is disabled. 1: Clock is enabled.	0x0
7	R/W	TIM_SYS_CLK_EN Timer clock selection: 0: LP_CLK clock 1: DIVN clock NOTE: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8).	0x0
6	R/W	TIM_FREE_RUN_MODE_EN Valid when timer counts up, if it is 1 timer does not zero when reaches to reload value. it becomes zero only when it reaches the max value.	0x0
5	R/W	TIM_IRQ_EN Interrupt mask: 0: Timer IRQ is disable. 1: Timer IRQ is enable.	0x0
4	R/W	TIM_IN2_EVENT_FALL_EN Event input 2 edge selection: 0: Rising edge 1: Falling edge	0x0
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge selection: 0: Rising edge 1: Falling edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction: 0: Up 1: Down NOTE: Only change this bit when timer is disabled.	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode: 0: Counter is enabled. 1: One shot is enabled.	0x0
0	R/W	TIM_EN Timer enable: 0: Off 1: On	0x0

Table 759: TIMER6_TIMER_VAL_REG (0x40080504)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE2	0x0

Bit	Mode	Symbol/Description	Reset
		Gives the current timer value.	

Table 760: TIMER6_STATUS_REG (0x40080508)

Bit	Mode	Symbol/Description	Reset
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13:12	-	- Reserved	0x0
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock is switched to divn clock.	0x0
10	R	TIM_PWM_BUSY Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	0x0
9	R	TIM_TIMER_BUSY Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	0x0
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ has occured, this bit is 1.	0x0
7:4	-	- Reserved	0x0
3:2	R	TIM_ONESHOT_PHASE Oneshot phase: 0: Wait for event. 1: Delay phase. 2: Start shot. 3: Shot phase.	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 761: TIMER6_GPIO1_CONF_REG (0x4008050C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02	0x0

Bit	Mode	Symbol/Description	Reset
		4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 762: TIMER6_GPIO2_CONF_REG (0x40080510)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09	0x0

Bit	Mode	Symbol/Description	Reset
		11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 763: TIMER6_SETTINGS_REG (0x40080514)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode; delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles).	0x0

Table 764: TIMER6_SHOTWIDTH_REG (0x40080518)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in Oneshot mode.	0x0

Table 765: TIMER6_PRE_SETTINGS_REG (0x4008051C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency/(TIM_PRESCALER+1).	0x0

Table 766: TIMER6_CAPTURE_GPIO1_REG (0x40080520)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12	0x0

Bit	Mode	Symbol/Description	Reset
		Gives the capture time for event on GPIO1.	

Table 767: TIMER6_CAPTURE_GPIO2_REG (0x40080524)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22 Gives the capture time for event on GPIO2.	0x0

Table 768: TIMER6_PRESCALER_VAL_REG (0x40080528)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R	TIM_PRESCALER_VAL Gives the current prescaler counter value.	0x0

Table 769: TIMER6_PWM_CTRL_REG (0x4008052C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC Defines the PWM duty cycle. TIM_PWM_DC/(TIM_PWM_FREQ+1).	0x0
15:0	R/W	TIM_PWM_FREQ Defines the PWM frequency. Timer clock frequency/(TIM_PWM_FREQ+1). Timer clock is the clock after prescaler.	0x0

Table 770: TIMER6_PULSE_GPIO_SEL_REG (0x40080530)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2 Select one of the 32 GPIOs as input for the pulse counter, valid values 0-32: 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01	0x0

Bit	Mode	Symbol/Description	Reset
		17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	

Table 771: TIMER6_PULSE_CNT_CTRL_REG (0x40080544)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an IRQ is fired for the pulse counter. NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 772: TIMER6_CLEAR_IRQ_REG (0x40080554)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ Write any value to clear interrupt.	0x0

Table 773: TIMER6_CLEAR_IRQ_PULSE_REG (0x40080558)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ Write any value to clear IRQ pulse interrupt.	0x0

10.33 Timer7 Control Registers

Table 774: Register map TIMER7

Address	Register	Description
0x40080600	TIMER7_CTRL_REG	Timer Control Register
0x40080604	TIMER7_TIMER_VAL_REG	Timer Counter Value
0x40080608	TIMER7_STATUS_REG	Timer Status Register
0x4008060c	TIMER7_GPIO1_CONF_REG	Timer GPIO1 selection

Address	Register	Description
0x40080610	TIMER7_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080614	TIMER7_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080618	TIMER7_SHOTWIDTH_REG	Timer Shot Duration in Shot mode
0x4008061c	TIMER7_PRE_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080620	TIMER7_CAPTURE_GPIO1_REG	Timer Value for Event on GPIO1
0x40080624	TIMER7_CAPTURE_GPIO2_REG	Timer Value for Event on GPIO2
0x40080628	TIMER7_PRESCALER_VAL_REG	Timer Prescaler Counter Value
0x4008062c	TIMER7_PWM_CTRL_REG	Timer PWM Frequency Register
0x40080630	TIMER7_PULSE_GPIO_SEL_REG	Timer Pulse Counter CTRL Register
0x40080644	TIMER7_PULSE_CNT_CTRL_REG	Timer Pulse Counter CTRL Register
0x40080654	TIMER7_CLEAR_IRQ_REG	Timer Clear Interrupt
0x40080658	TIMER7_CLEAR_IRQ_PULSE_REG	Timer Clear Pulse Interrupt

Table 775: TIMER7_CTRL_REG (0x40080600)

Bit	Mode	Symbol/Description	Reset
31:20	-	-	0x0
		Reserved	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Counter trigger edge selection: 0: The counter is triggered on a rising edge. 1: The counter is triggered on a falling edge. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG.	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Edge detection counter: 0: Disable 1: Enable	
17:9	-	-	0x0
		Reserved	
8	R/W	TIM_CLK_EN	0x0
		Timer clock enable: 0: Clock is disabled. 1: Clock is enabled.	
7	R/W	TIM_SYS_CLK_EN	0x0
		Timer clock selection:	

Bit	Mode	Symbol/Description	Reset
		0: LP_CLK clock 1: DIVN clock NOTE: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8).	
6	R/W	TIM_FREE_RUN_MODE_EN Valid when timer counts up, if it is 1 timer does not zero when reaches to reload value. it becomes zero only when it reaches the max value.	0x0
5	R/W	TIM_IRQ_EN Interrupt mask: 0: Timer IRQ is disable. 1: Timer IRQ is enable.	0x0
4	R/W	TIM_IN2_EVENT_FALL_EN Event input 2 edge selection: 0: Rising edge 1: Falling edge	0x0
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge selection: 0: Rising edge 1: Falling edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction: 0: Up 1: Down NOTE: Only change this bit when timer is disabled.	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode: 0: Counter is enabled. 1: One shot is enabled.	0x0
0	R/W	TIM_EN Timer enable: 0: Off 1: On	0x0

Table 776: TIMER7_TIMER_VAL_REG (0x40080604)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE2 Gives the current timer value.	0x0

Table 777: TIMER7_STATUS_REG (0x40080608)

Bit	Mode	Symbol/Description	Reset
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13:12	-	-	0x0

Bit	Mode	Symbol/Description	Reset
		Reserved	
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock is switched to divn clock.	0x0
10	R	TIM_PWM_BUSY Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	0x0
9	R	TIM_TIMER_BUSY Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	0x0
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ has occured, this bit is 1.	0x0
7:4	-	- Reserved	0x0
3:2	R	TIM_ONESHOT_PHASE Oneshot phase: 0: Wait for event. 1: Delay phase. 2: Start shot. 3: Shot phase.	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 778: TIMER7_GPIO1_CONF_REG (0x4008060C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11	0x0

Bit	Mode	Symbol/Description	Reset
		13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 779: TIMER7_GPIO2_CONF_REG (0x40080610)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04	0x0

Bit	Mode	Symbol/Description	Reset
		20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 780: TIMER7_SETTINGS_REG (0x40080614)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode; delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles).	0x0

Table 781: TIMER7_SHOTWIDTH_REG (0x40080618)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in Oneshot mode.	0x0

Table 782: TIMER7_PRE_SETTINGS_REG (0x4008061C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency/(TIM_PRESCALER+1).	0x0

Table 783: TIMER7_CAPTURE_GPIO1_REG (0x40080620)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12 Gives the capture time for event on GPIO1.	0x0

Table 784: TIMER7_CAPTURE_GPIO2_REG (0x40080624)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22 Gives the capture time for event on GPIO2.	0x0

Table 785: TIMER7_PRESCALER_VAL_REG (0x40080628)

Bit	Mode	Symbol/Description	Reset
31:5	-	-	0x0
		Reserved	
4:0	R	TIM_PRESCALER_VAL	0x0
		Gives the current prescaler counter value.	

Table 786: TIMER7_PWM_CTRL_REG (0x4008062C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC	0x0
		Defines the PWM duty cycle. TIM_PWM_DC/(TIM_PWM_FREQ+1).	
15:0	R/W	TIM_PWM_FREQ	0x0
		Defines the PWM frequency. Timer clock frequency/(TIM_PWM_FREQ+1). Timer clock is the clock after prescaler.	

Table 787: TIMER7_PULSE_GPIO_SEL_REG (0x40080630)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2	0x0
		Select one of the 32 GPIOs as input for the pulse counter, valid values 0-32: 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11	

Bit	Mode	Symbol/Description	Reset
		27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	

Table 788: TIMER7_PULSE_CNT_CTRL_REG (0x40080644)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an IRQ is fired for the pulse counter. NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 789: TIMER7_CLEAR_IRQ_REG (0x40080654)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ Write any value to clear interrupt.	0x0

Table 790: TIMER7_CLEAR_IRQ_PULSE_REG (0x40080658)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ Write any value to clear IRQ pulse interrupt.	0x0

10.34 Timer8 Control Registers

Table 791: Register map TIMER8

Address	Register	Description
0x40080700	TIMER8_CTRL_REG	Timer Control Register
0x40080704	TIMER8_TIMER_VAL_REG	Timer Counter Value
0x40080708	TIMER8_STATUS_REG	Timer Status Register
0x4008070c	TIMER8_GPIO1_CONF_REG	Timer GPIO1 selection
0x40080710	TIMER8_GPIO2_CONF_REG	Timer GPIO2 selection
0x40080714	TIMER8_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080718	TIMER8_SHOTWIDTH_REG	Timer Shot Duration in Shot mode
0x4008071c	TIMER8_PRE_SETTINGS_REG	Timer Reload Value and Delay in Shot mode
0x40080720	TIMER8_CAPTURE_GPIO1_REG	Timer Value for Event on GPIO1

Address	Register	Description
0x40080724	TIMER8_CAPTURE_GPIO2_REG	Timer Value for Event on GPIO2
0x40080728	TIMER8_PRESCALER_VAL_REG	Timer Prescaler Counter Value
0x4008072c	TIMER8_PWM_CTRL_REG	Timer PWM Frequency Register
0x40080730	TIMER8_PULSE_GPIO_SEL_REG	Timer Pulse Counter CTRL Register
0x40080744	TIMER8_PULSE_CNT_CTRL_REG	Timer Pulse Counter CTRL Register
0x40080754	TIMER8_CLEAR_IRQ_REG	Timer Clear Interrupt
0x40080758	TIMER8_CLEAR_IRQ_PULSE_REG	Timer Clear Pulse Interrupt

Table 792: TIMER8_CTRL_REG (0x40080700)

Bit	Mode	Symbol/Description	Reset
31:20	-	-	0x0
		Reserved	
19	R/W	TIM_EDGE_DET_CNT_FALL_EN	0x0
		Counter trigger edge selection: 0: The counter is triggered on a rising edge. 1: The counter is triggered on a falling edge. Note: Only change this when EDGE_DET_CNT_EN = 0 in TIMER_CTRL_REG.	
18	R/W	TIM_EDGE_DET_CNT_EN	0x0
		Edge detection counter: 0: Disable 1: Enable	
17:9	-	-	0x0
		Reserved	
8	R/W	TIM_CLK_EN	0x0
		Timer clock enable: 0: Clock is disabled. 1: Clock is enabled.	
7	R/W	TIM_SYS_CLK_EN	0x0
		Timer clock selection: 0: LP_CLK clock 1: DIVN clock NOTE: When switching clock, the timer clock should be disabled (TIM_CLK_EN, bit 8).	
6	R/W	TIM_FREE_RUN_MODE_EN	0x0
		Valid when timer counts up, if it is 1 timer does not zero when reaches to reload value. it becomes zero only when it reaches the max value.	
5	R/W	TIM_IRQ_EN	0x0
		Interrupt mask:	

Bit	Mode	Symbol/Description	Reset
		0: Timer IRQ is disable. 1: Timer IRQ is enable.	
4	R/W	TIM_IN2_EVENT_FALL_EN Event input 2 edge selection: 0: Rising edge 1: Falling edge	0x0
3	R/W	TIM_IN1_EVENT_FALL_EN Event input 1 edge selection: 0: Rising edge 1: Falling edge	0x0
2	R/W	TIM_COUNT_DOWN_EN Timer count direction: 0: Up 1: Down NOTE: Only change this bit when timer is disabled.	0x0
1	R/W	TIM_ONESHOT_MODE_EN Timer mode: 0: Counter is enabled. 1: One shot is enabled.	0x0
0	R/W	TIM_EN Timer enable: 0: Off 1: On	0x0

Table 793: TIMER8_TIMER_VAL_REG (0x40080704)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_TIMER_VALUE2 Gives the current timer value.	0x0

Table 794: TIMER8_STATUS_REG (0x40080708)

Bit	Mode	Symbol/Description	Reset
14	R	TIM_IRQ_PULSE_STATUS Status bit of IRQ pulse counter. When the pulse counter reaches the threshold value, this bit is 1.	0x0
13:12	-	- Reserved	0x0
11	R	TIM_SWITCHED_TO_DIVN_CLK Indicates that timer clock is switched to divn clock.	0x0
10	R	TIM_PWM_BUSY Busy with synchronizing PWM_FREQ_REG and PWM_DC_REG. Do not write a new value to these registers when this bit is high.	0x0
9	R	TIM_TIMER_BUSY	0x0

Bit	Mode	Symbol/Description	Reset
		Busy with synchronizing PRESCALER_REG, RELOAD_REG and SHOTWIDTH_REG. Do not write a new value to these registers when this bit is high.	
8	R	TIM_IRQ_STATUS IRQ status bit. When an IRQ has occured, this bit is 1.	0x0
7:4	-	- Reserved	0x0
3:2	R	TIM_ONESHOT_PHASE Oneshot phase: 0: Wait for event. 1: Delay phase. 2: Start shot. 3: Shot phase.	0x0
1	R	TIM_IN2_STATE Gives the logic level of the IN1.	0x0
0	R	TIM_IN1_STATE Gives the logic level of the IN2.	0x0

Table 795: TIMER8_GPIO1_CONF_REG (0x4008070C)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO1_CONF Select one of the 32 GPIOs as IN1, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05	0x0

Bit	Mode	Symbol/Description	Reset
		21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 796: TIMER8_GPIO2_CONF_REG (0x40080710)

Bit	Mode	Symbol/Description	Reset
31:6	-	- Reserved	0x0
5:0	R/W	TIM_GPIO2_CONF Select one of the 32 GPIOs as IN2, Valid values 0-32. 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12	0x0

Bit	Mode	Symbol/Description	Reset
		28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17	

Table 797: TIMER8_SETTINGS_REG (0x40080714)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_RELOAD2 Reload or max value in timer mode; delay phase duration in oneshot mode. Actual delay is the register value plus synchronization time (3 clock cycles).	0x0

Table 798: TIMER8_SHOTWIDTH_REG (0x40080718)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	TIM_SHOTWIDTH2 Shot phase duration in Oneshot mode.	0x0

Table 799: TIMER8_PRE_SETTINGS_REG (0x4008071C)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R/W	TIM_PRESCALER2 Defines the timer count frequency. CLOCK frequency/(TIM_PRESCALER+1).	0x0

Table 800: TIMER8_CAPTURE_GPIO1_REG (0x40080720)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO12 Gives the capture time for event on GPIO1.	0x0

Table 801: TIMER8_CAPTURE_GPIO2_REG (0x40080724)

Bit	Mode	Symbol/Description	Reset
31:0	R	TIM_CAPTURE_GPIO22 Gives the capture time for event on GPIO2.	0x0

Table 802: TIMER8_PRESCALER_VAL_REG (0x40080728)

Bit	Mode	Symbol/Description	Reset
31:5	-	- Reserved	0x0
4:0	R	TIM_PRESCALER_VAL Gives the current prescaler counter value.	0x0

Table 803: TIMER8_PWM_CTRL_REG (0x4008072C)

Bit	Mode	Symbol/Description	Reset
31:16	R/W	TIM_PWM_DC Defines the PWM duty cycle. TIM_PWM_DC/(TIM_PWM_FREQ+1).	0x0
15:0	R/W	TIM_PWM_FREQ Defines the PWM frequency. Timer clock frequency/(TIM_PWM_FREQ+1). Timer clock is the clock after prescaler.	0x0

Table 804: TIMER8_PULSE_GPIO_SEL_REG (0x40080730)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	PULSE_CNT_GPIO_SEL2 Select one of the 32 GPIOs as input for the pulse counter, valid values 0-32: 0: Disable input 1: P0_00 2: P0_01 3: P0_02 4: P0_03 5: P0_04 6: P0_05 7: P0_06 8: P0_07 9: P0_08 10: P0_09 11: P0_10 12: P0_11 13: P0_12 14: P0_13 15: P1_00 16: P1_01 17: P1_02 18: P1_03 19: P1_04 20: P1_05 21: P1_06 22: P1_07 23: P1_08 24: P1_09 25: P1_10 26: P1_11 27: P1_12 28: P1_13 29: P1_14 30: P1_15 31: P1_16 32: P1_17 NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 805: TIMER8_PULSE_CNT_CTRL_REG (0x40080744)

Bit	Mode	Symbol/Description	Reset
31:0	R/W	PULSE_CNT_THRESHOLD2 Select after how many pulses an IRQ is fired for the pulse counter. NOTE: Only change this when EDGE_DET_CNT_EN=0 in TIMER_CTRL_REG.	0x0

Table 806: TIMER8_CLEAR_IRQ_REG (0x40080754)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_IRQ Write any value to clear interrupt.	0x0

Table 807: TIMER8_CLEAR_IRQ_PULSE_REG (0x40080758)

Bit	Mode	Symbol/Description	Reset
0	R0/W	TIM_CLEAR_PULSE_IRQ Write any value to clear IRQ pulse interrupt.	0x0

10.35 UART Registers

Table 808: Register map UART

Address	Register	Description
0x40000000	UART_DR_REG	UART Data Register
0x40000004	UART_RSR_REG	UART Receive Status (Read), Error Clear (Write) Register
0x40000018	UART_FR_REG	UART Flag Register
0x40000024	UART_IBRD_REG	UART Integer Baud Rate Divisor Register
0x40000028	UART_FBRD_REG	UART Fractional Baud Rate Divisor Register
0x4000002c	UART_LCR_H_REG	UART Line Control Register (High Byte)
0x40000030	UART_CR_REG	UART Control Register
0x40000034	UART_IFLS_REG	UART Interrupt FIFO Level Select Register
0x40000038	UART_IMSC_REG	UART Interrupt Mask Set/Clear Register
0x4000003c	UART_RIS_REG	UART Raw Interrupt Status Register
0x40000040	UART_MIS_REG	UART Masked Interrupt Status Register
0x40000044	UART_ICR_REG	UART Interrupt Clear Register
0x40000048	UART_DMACR_REG	UART DMA Control Register
0x4000004c	UART_WA_REG	UART Word Access Enable Register
0x40000050	UART_SWFC_REG	UART Software Flow Control Enable Register
0x40000054	UART_RS485EN_REG	UART RS485 Mode Enable Register

Table 809: UART_DR_REG (0x40000000)

Bit	Mode	Symbol/Description	Reset
15:12	R/W	- Reserved	0x0
11	R/W	DR_OE	0x0

Bit	Mode	Symbol/Description	Reset
		Overrun error. This bit is set to 1 if data is received and the receive FIFO is already full. This is cleared to 0 when there is an empty space in the FIFO and a new character can be written to it.	
10	R/W	DR_BE Break error. This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state), and the next valid start bit is received.	0x0
9	R/W	DR_PE Parity error. When set to 1, it indicates that the parity of the received data character does not match the parity that the EPS and SPS bits in the Line Control Register, UARTLCR_H. In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0
8	R/W	DR_FE Framing error. When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0
7:0	R/W	DR_DATA Receive (read) data charater Transmit (write) data character	0x0

Table 810: UART_RSR_REG (0x40000004)

Bit	Mode	Symbol/Description	Reset
7:4	R/W	- Reserved	0x0
3	R/W	RSR_OE Overrun error. This bit is set to 1 if data is received and the FIFO is already full. This bit is cleared to 0 by a write to the register. The FIFO contents remain valid because no more data is written when the FIFO is full, only the contents of the shift register are overwritten. The CPU must now read the data, to empty the FIFO.	0x0
2	R/W	RSR_BE Break error. This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity, and stop bits). This bit is cleared to 0 after a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state) and the next valid start bit is received.	0x0
1	R/W	RSR_PE Parity error. When set to 1, it indicates that the parity of the received data character does not match the parity that the EPS and SPS bits in the Line Control Register, UARTLCR_H.	0x0

Bit	Mode	Symbol/Description	Reset
		This bit is cleared to 0 by a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO.	
0	R/W	RSR_FE Framing error. When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). This bit is cleared to 0 by a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0

Table 811: UART_FR_REG (0x40000018)

Bit	Mode	Symbol/Description	Reset
15:9	R	- Reserved	0x0
8	R	RI Ring indicator. This bit is the complement of the UART ring indicator, nUARTRI, modem status input. That is, the bit is 1 when nUARTRI is LOW.	0x0
7	R	TXFE Transmit FIFO empty. The meaning of this bit depends on the state of the FEN bit in the Line Control Register, UARTLCR_H If the FIFO is disabled, this bit is set when the transmit holding register is empty. If the FIFO is enabled, the TXFE bit is set when the transmit FIFO is empty. This bit does not indicate if there is data in the transmit shift register.	0x1
6	R	RXFF Receive FIFO full. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the receive holding register is full. If the FIFO is enabled, the RXFF bit is set when the receive FIFO is full.	0x0
5	R	TXFF Transmit FIFO full. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the transmit holding register is full. If the FIFO is enabled, the TXFF bit is set when the transmit FIFO is full.	0x0
4	R	RXFE Receive FIFO empty. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the receive holding register is empty. If the FIFO is enabled, the RXFE bit is set when the receive FIFO is empty.	0x1
3	R	BUSY UART busy. If this bit is set to 1, the UART is busy transmitting data. This bit remains set until the complete byte, including all the stop bits, has been sent from the shift register. This bit is set as soon as the transmit FIFO becomes non-empty, regardless of whether the UART is enabled or not.	0x0
2	R	DCD Data carrier detect. This bit is the complement of the UART data carrier detect, nUARTDCD, modem status input.	0x0

Bit	Mode	Symbol/Description	Reset
		That is, the bit is 1 when nUARTDCD is LOW.	
1	R	DSR Data set ready. This bit is the complement of the UART data set ready, nUARTDSR, modem status input. That is, the bit is 1 when nUARTDSR is LOW.	0x0
0	R	CTS Clear to send. This bit is the complement of the UART clear to send, nUARTCTS, modem status input. That is, the bit is 1 when nUARTCTS is LOW.	0x1

Table 812: UART_IBRD_REG (0x40000024)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	BAUD_DIVINT The integer baud rate divisor. These bits are cleared to 0 on reset.	0x0

Table 813: UART_FBRD_REG (0x40000028)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	BAUD_DIVFRAC The fractional baud rate divisor. These bits are cleared to 0 on reset.	0x0

Table 814: UART_LCR_H_REG (0x4000002C)

Bit	Mode	Symbol/Description	Reset
15:8	R/W	- Reserved	0x0
7	R/W	SPS Stick parity select. 0 = stick parity is disabled 1 = either: If the EPS bit is 0 then the parity bit is transmitted and checked as a 1 If the EPS bit is 1 then the parity bit is transmitted and checked as a 0. This bit has no effect when the PEN bit disables parity checking and generation.	0x0
6:5	R/W	WLEN Word length. These bits indicate the number of data bits transmitted or received in a frame as follows: b11 = 8 bits b10 = 7 bits b01 = 6 bits b00 = 5 bits.	0x0
4	R/W	FEN Enable FIFOs: 0 = FIFOs are disabled (character mode) that is, the FIFOs become 1-byte-deep holding registers	0x0

Bit	Mode	Symbol/Description	Reset
		1 = transmit and receive FIFO buffers are enabled (FIFO mode).	
3	R/W	STP2 Two stop bits select. If this bit is set to 1, two stop bits are transmitted at the end of the frame. The receive logic does not check for two stop bits being received.	0x0
2	R/W	EPS Even parity select. Controls the type of parity the UART uses during transmission and reception: 0 = odd parity. The UART generates or checks for an odd number of 1s in the data and parity bits. 1 = even parity. The UART generates or checks for an even number of 1s in the data and parity bits. This bit has no effect when the PEN bit disables parity checking and generation.	0x0
1	R/W	PEN Parity enable: 0 = parity is disabled and no parity bit added to the data frame 1 = parity checking and generation is enabled.	0x0
0	R/W	BRK Send break. If this bit is set to 1, a low-level is continually output on the UARTTXD output, after completing transmission of the current character. For the proper execution of the break command, the software must set this bit for at least two complete frames. For normal use, this bit must be cleared to 0.	0x0

Table 815: UART_CR_REG (0x40000030)

Bit	Mode	Symbol/Description	Reset
15	R/W	CTSEn CTS hardware flow control enable. If this bit is set to 1, CTS hardware flow control is enabled. Data is only transmitted when the nUARTCTS signal is asserted.	0x0
14	R/W	RTSEn RTS hardware flow control enable. If this bit is set to 1, RTS hardware flow control is enabled. Data is only requested when there is space in the receive FIFO for it to be received.	0x0
13	R/W	Out2 This bit is the complement of the UART Out2 (nUARTOut2) modem status output. That is, when the bit is programmed to a 1, the output is 0. For DTE this can be used as Ring Indicator (RI).	0x0
12	R/W	Out1 This bit is the complement of the UART Out1 (nUARTOut1) modem status output. That is, when the bit is programmed to a 1 the output is 0. For DTE this can be used as Data Carrier Detect (DCD).	0x0
11	R/W	RTS Request to send. This bit is the complement of the UART request to send, nUARTRTS, modem status output. That is, when the bit is programmed to a 1 then nUARTRTS is LOW	0x0
10	R/W	DTR	0x0

Bit	Mode	Symbol/Description	Reset
		Data transmit ready. This bit is the complement of the UART data transmit ready, nUARTDTR, modem status output. That is, when the bit is programmed to a 1 then nUARTDTR is LOW	
9	R/W	RXE Receive enable. If this bit is set to 1, the receive section of the UART is enabled.	0x1
8	R/W	TXE Transmit enable. If this bit is set to 1, the transmit section of the UART is enabled.	0x1
7	R/W	LBE Loopback enable. If this bit is set to 1, the UARTTXD path is fed through to the UARTRXD path. When this bit is set, the modem outputs are also fed through to the modem inputs. This bit is cleared to 0 on reset, to disable loopback.	0x0
6:1	R/W	- Reserved	0x0
0	R/W	UARTEN UART enable: 0 = UART is disabled. If the UART is disabled in the middle of transmission or reception, it completes the current character before stopping. 1 = the UART is enabled. Data transmission and reception occurs.	0x0

Table 816: UART_IFLS_REG (0x40000034)

Bit	Mode	Symbol/Description	Reset
15:6	R/W	- Reserved	0x0
5:3	R/W	RXIFLSEL Receive interrupt FIFO level select. The trigger points for the receive interrupt are as follows: b000 = Receive FIFO becomes 1/8 full b001 = Receive FIFO becomes 1/4 full b010 = Receive FIFO becomes 1/2 full b011 = Receive FIFO becomes 3/4 full b100 = Receive FIFO becomes 7/8 full b101-b111 = Reserved.	0x2
2:0	R/W	TXIFLSEL Transmit interrupt FIFO level select. The trigger points for the transmit interrupt are as follows: b000 = Transmit FIFO becomes 1/8 full b001 = Transmit FIFO becomes 1/4 full b010 = Transmit FIFO becomes 1/2 full b011 = Transmit FIFO becomes 3/4 full b100 = Transmit FIFO becomes 7/8 full b101-b111 = Reserved.	0x2

Table 817: UART_IMSC_REG (0x40000038)

Bit	Mode	Symbol/Description	Reset
15:11	R/W	-	0x0
		Reserved	
10	R/W	OEIM	0x0
		Overrun error interrupt mask. A read returns the current mask for the UARTOEINTR interrupt. On a write of 1, the mask of the UARTOEINTR interrupt is set. A write of 0 clears the mask.	
9	R/W	BEIM	0x0
		Break error interrupt mask. A read returns the current mask for the UARTBEINTR interrupt. On a write of 1, the mask of the UARTBEINTR interrupt is set. A write of 0 clears the mask.	
8	R/W	PEIM	0x0
		Parity error interrupt mask. A read returns the current mask for the UARTPEINTR interrupt. On a write of 1, the mask of the UARTPEINTR interrupt is set. A write of 0 clears the mask.	
7	R/W	FEIM	0x0
		Framing error interrupt mask. A read returns the current mask for the UARTFEINTR interrupt. On a write of 1, the mask of the UARTFEINTR interrupt is set. A write of 0 clears the mask.	
6	R/W	RTIM	0x0
		Receive timeout interrupt mask. A read returns the current mask for the UARTRTINTR interrupt. On a write of 1, the mask of the UARTRTINTR interrupt is set. A write of 0 clears the mask.	
5	R/W	TXIM	0x0
		Transmit interrupt mask. A read returns the current mask for the UARTTXINTR interrupt. On a write of 1, the mask of the UARTTXINTR interrupt is set. A write of 0 clears the mask.	
4	R/W	RXIM	0x0
		Receive interrupt mask. A read returns the current mask for the UARTRXINTR interrupt. On a write of 1, the mask of the UARTRXINTR interrupt is set. A write of 0 clears the mask.	
3	R/W	DSRMIM	0x0
		nUARTDSR modem interrupt mask. A read returns the current mask for the UARTDSRINTR interrupt. On a write of 1, the mask of the UARTDSRINTR interrupt is set. A write of 0 clears the mask.	
2	R/W	DCDMIM	0x0
		nUARTDCD modem interrupt mask. A read returns the current mask for the UARTDCDINTR interrupt. On a write of 1, the mask of the UARTDCDINTR interrupt is set. A write of 0 clears the mask.	

Bit	Mode	Symbol/Description	Reset
1	R/W	CTSMIM nUARTCTS modem interrupt mask. A read returns the current mask for the UARTCTSINTR interrupt. On a write of 1, the mask of the UARTCTSINTR interrupt is set. A write of 0 clears the mask.	0x0
0	R/W	RIMIM nUARTRI modem interrupt mask. A read returns the current mask for the UARTRIINTR interrupt. On a write of 1, the mask of the UARTRIINTR interrupt is set. A write of 0 clears the mask.	0x0

Table 818: UART_RIS_REG (0x4000003C)

Bit	Mode	Symbol/Description	Reset
15:11	R	- Reserved	0x0
10	R	OERIS Overrun error interrupt status. Returns the raw interrupt state of the UARTOEINTR interrupt.	0x0
9	R	BERIS Break error interrupt status. Returns the raw interrupt state of the UARTBEINTR interrupt.	0x0
8	R	PERIS Parity error interrupt status. Returns the raw interrupt state of the UARTPEINTR interrupt.	0x0
7	R	FERIS Framing error interrupt status. Returns the raw interrupt state of the UARTFEINTR interrupt.	0x0
6	R	RTRIS Receive timeout interrupt status. Returns the raw interrupt state of the UARTRTINTR interrupt.	0x0
5	R	TXRIS Transmit interrupt status. Returns the raw interrupt state of the UARTTXINTR interrupt.	0x0
4	R	RXRIS Receive interrupt status. Returns the raw interrupt state of the UARTRXINTR interrupt.	0x0
3	R	DSRRMIS nUARTDSR modem interrupt status. Returns the raw interrupt state of the UARTDSRINTR interrupt.	0x1
2	R	DCDRMIS nUARTDCD modem interrupt status. Returns the raw interrupt state of the UARTDCDINTR interrupt.	0x1
1	R	CTSRMIS nUARTCTS modem interrupt status. Returns the raw interrupt state of the UARTCTSINTR interrupt.	0x0

Bit	Mode	Symbol/Description	Reset
0	R	RIRMIS nUARTRI modem interrupt status. Returns the raw interrupt state of the UARTRIINTR interrupt.	0x1

Table 819: UART_MIS_REG (0x40000040)

Bit	Mode	Symbol/Description	Reset
15:11	R	- Reserved	0x0
10	R	OEMIS Overrun error masked interrupt status. Returns the masked interrupt state of the UARTOEINTR interrupt.	0x0
9	R	BEMIS Break error masked interrupt status. Returns the masked interrupt state of the UARTBEINTR interrupt.	0x0
8	R	PEMIS Parity error masked interrupt status. Returns the masked interrupt state of the UARTPEINTR interrupt.	0x0
7	R	FEMIS Framing error masked interrupt status. Returns the masked interrupt state of the UARTFEINTR interrupt.	0x0
6	R	RTMIS Receive timeout masked interrupt status. Returns the masked interrupt state of the UARTRTINTR interrupt.	0x0
5	R	TXMIS Transmit masked interrupt status. Returns the masked interrupt state of the UARTTXINTR interrupt.	0x0
4	R	RXMIS Receive masked interrupt status. Returns the masked interrupt state of the UARTRXINTR interrupt.	0x0
3	R	DSRMMIS nUARTDSR modem masked interrupt status. Returns the masked interrupt state of the UARTDSRINTR interrupt.	0x0
2	R	DCDMMIS nUARTDCD modem masked interrupt status. Returns the masked interrupt state of the UARTDCDINTR interrupt.	0x0
1	R	CTSMMIS nUARTCTS modem masked interrupt status. Returns the masked interrupt state of the UARTCTSINTR interrupt.	0x0
0	R	RIMMIS nUARTRI modem masked interrupt status. Returns the masked interrupt state of the UARTRIINTR interrupt.	0x0

Table 820: UART_ICR_REG (0x40000044)

Bit	Mode	Symbol/Description	Reset
15:11	W	-	0x0
		Reserved	
10	W	OEIC	0x0
		Overrun error interrupt clear. Clears the UARTOEINTR interrupt.	
9	W	BEIC	0x0
		Break error interrupt clear. Clears the UARTBEINTR interrupt.	
8	W	PEIC	0x0
		Parity error interrupt clear. Clears the UARTPEINTR interrupt.	
7	W	FEIC	0x0
		Framing error interrupt clear. Clears the UARTFEINTR interrupt.	
6	W	RTIC	0x0
		Receive timeout interrupt clear. Clears the UARTRTINTR interrupt.	
5	W	TXIC	0x0
		Transmit interrupt clear. Clears the UARTTXINTR interrupt.	
4	W	RXIC	0x0
		Receive interrupt clear. Clears the UARTRXINTR interrupt.	
3	W	DSRMIC	0x0
		nUARTDSR modem interrupt clear. Clears the UARTDSRINTR interrupt.	
2	W	DCDMIC	0x0
		nUARTDCD modem interrupt clear. Clears the UARTDCDINTR interrupt.	
1	W	CTSMIC	0x0
		nUARTCTS modem interrupt clear. Clears the UARTCTSINTR interrupt.	
0	W	RIMIC	0x0
		nUARTRI modem interrupt clear. Clears the UARTRIINTR interrupt.	

Table 821: UART_DMACR_REG (0x40000048)

Bit	Mode	Symbol/Description	Reset
15:3	R/W	-	0x0
		Reserved	
2	R/W	DMAONERR	0x0
		DMA on error. If this bit is set to 1, the DMA receive request outputs, UARTRXDMASREQ or UARTRXDMABREQ, are disabled when the UART error interrupt is asserted	
1	R/W	TXDMAE	0x0
		Transmit DMA enable. If this bit is set to 1, DMA for the transmit FIFO is enabled.	
0	R/W	RXDMAE	0x0
		Receive DMA enable. If this bit is set to 1, DMA for the receive FIFO is enabled.	

Table 822: UART_WA_REG (0x4000004C)

Bit	Mode	Symbol/Description	Reset
0	R/W	WAE DMA word access enable bit. If this bit is set to 1, DMA word access mode is enabled.	0x1

Table 823: UART_SWFC_REG (0x40000050)

Bit	Mode	Symbol/Description	Reset
0	R/W	SWFCE DMA software flow control enable bit. If this bit is set to 1, DMA software flow control mode is enabled.	0x0

Table 824: UART_RS485EN_REG (0x40000054)

Bit	Mode	Symbol/Description	Reset
0	R/W	RS485E DMA RS485 mode enable bit. If this bit is set to 1, DMA RS485 mode is enabled.	0x0

10.36 UART2 Registers

Table 825: Register map UART2

Address	Register	Description
0x40001000	UART2_DR_REG	UART Data Register
0x40001004	UART2_RSR_REG	UART Receive Status (Read), Error Clear (Write) Register
0x40001018	UART2_FR_REG	UART Flag Register
0x40001024	UART2_IBRD_REG	UART Integer Baud Rate Divisor Register
0x40001028	UART2_FBRD_REG	UART Fractional Baud Rate Divisor Register
0x4000102c	UART2_LCR_H_REG	UART Line Control Register (High Byte)
0x40001030	UART2_CR_REG	UART Control Register
0x40001034	UART2_IFLS_REG	UART Interrupt FIFO Level Select Register
0x40001038	UART2_IMSC_REG	UART Interrupt Mask Set/Clear Register
0x4000103c	UART2_RIS_REG	UART Raw Interrupt Status Register
0x40001040	UART2_MIS_REG	UART Masked Interrupt Status Register
0x40001044	UART2_ICR_REG	UART Interrupt Clear Register
0x40001048	UART2_DMACR_REG	UART DMA Control Register
0x4000104c	UART2_WA_REG	UART Word Access Enable Register
0x40001050	UART2_SWFC_REG	UART Software Flow Control Enable Register
0x40001054	UART2_RS485EN_REG	UART RS485 Mode Enable Register

Table 826: UART2_DR_REG (0x40001000)

Bit	Mode	Symbol/Description	Reset
15:12	R/W	- Reserved	0x0

Bit	Mode	Symbol/Description	Reset
11	R/W	DR_OE Overrun error. This bit is set to 1 if data is received and the receive FIFO is already full. This is cleared to 0 when there is an empty space in the FIFO and a new character can be written to it.	0x0
10	R/W	DR_BE Break error. This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state), and the next valid start bit is received.	0x0
9	R/W	DR_PE Parity error. When set to 1, it indicates that the parity of the received data character does not match the parity that the EPS and SPS bits in the Line Control Register, UARTLCR_H. In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0
8	R/W	DR_FE Framing error. When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0
7:0	R/W	DR_DATA Receive (read) data charater Transmit (write) data character	0x0

Table 827: UART2_RSR_REG (0x40001004)

Bit	Mode	Symbol/Description	Reset
7:4	R/W	- Reserved	0x0
3	R/W	RSR_OE Overrun error. This bit is set to 1 if data is received and the FIFO is already full. This bit is cleared to 0 by a write to the register. The FIFO contents remain valid because no more data is written when the FIFO is full, only the contents of the shift register are overwritten. The CPU must now read the data, to empty the FIFO.	0x0
2	R/W	RSR_BE Break error. This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity, and stop bits). This bit is cleared to 0 after a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state) and the next valid start bit is received.	0x0
1	R/W	RSR_PE	0x0

Bit	Mode	Symbol/Description	Reset
		Parity error. When set to 1, it indicates that the parity of the received data character does not match the parity that the EPS and SPS bits in the Line Control Register, UARTLCR_H. This bit is cleared to 0 by a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO.	
0	R/W	RSR_FE Framing error. When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). This bit is cleared to 0 by a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0

Table 828: UART2_FR_REG (0x40001018)

Bit	Mode	Symbol/Description	Reset
15:9	R	- Reserved	0x0
8	R	RI Ring indicator. This bit is the complement of the UART ring indicator, nUARTRI, modem status input. That is, the bit is 1 when nUARTRI is LOW.	0x0
7	R	TXFE Transmit FIFO empty. The meaning of this bit depends on the state of the FEN bit in the Line Control Register, UARTLCR_H If the FIFO is disabled, this bit is set when the transmit holding register is empty. If the FIFO is enabled, the TXFE bit is set when the transmit FIFO is empty. This bit does not indicate if there is data in the transmit shift register.	0x1
6	R	RXFF Receive FIFO full. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the receive holding register is full. If the FIFO is enabled, the RXFF bit is set when the receive FIFO is full.	0x0
5	R	TXFF Transmit FIFO full. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the transmit holding register is full. If the FIFO is enabled, the TXFF bit is set when the transmit FIFO is full.	0x0
4	R	RXFE Receive FIFO empty. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the receive holding register is empty. If the FIFO is enabled, the RXFE bit is set when the receive FIFO is empty.	0x1
3	R	BUSY UART busy. If this bit is set to 1, the UART is busy transmitting data. This bit remains set until the complete byte, including all the stop bits, has been sent from the shift register. This bit is set as soon as the transmit FIFO becomes non-empty, regardless of whether the UART is enabled or not.	0x0

Bit	Mode	Symbol/Description	Reset
2	R	DCD Data carrier detect. This bit is the complement of the UART data carrier detect, nUARTDCD, modem status input. That is, the bit is 1 when nUARTDCD is LOW.	0x0
1	R	DSR Data set ready. This bit is the complement of the UART data set ready, nUARTDSR, modem status input. That is, the bit is 1 when nUARTDSR is LOW.	0x0
0	R	CTS Clear to send. This bit is the complement of the UART clear to send, nUARTCTS, modem status input. That is, the bit is 1 when nUARTCTS is LOW.	0x1

Table 829: UART2_IBRD_REG (0x40001024)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	BAUD_DIVINT The integer baud rate divisor. These bits are cleared to 0 on reset.	0x0

Table 830: UART2_FBRD_REG (0x40001028)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	BAUD_DIVFRAC The fractional baud rate divisor. These bits are cleared to 0 on reset.	0x0

Table 831: UART2_LCR_H_REG (0x4000102C)

Bit	Mode	Symbol/Description	Reset
15:8	R/W	- Reserved	0x0
7	R/W	SPS Stick parity select. 0 = stick parity is disabled 1 = either: If the EPS bit is 0 then the parity bit is transmitted and checked as a 1 If the EPS bit is 1 then the parity bit is transmitted and checked as a 0. This bit has no effect when the PEN bit disables parity checking and generation.	0x0
6:5	R/W	WLEN Word length. These bits indicate the number of data bits transmitted or received in a frame as follows: b11 = 8 bits b10 = 7 bits b01 = 6 bits b00 = 5 bits.	0x0
4	R/W	FEN	0x0

Bit	Mode	Symbol/Description	Reset
		Enable FIFOs: 0 = FIFOs are disabled (character mode) that is, the FIFOs become 1-byte-deep holding registers 1 = transmit and receive FIFO buffers are enabled (FIFO mode).	
3	R/W	STP2 Two stop bits select. If this bit is set to 1, two stop bits are transmitted at the end of the frame. The receive logic does not check for two stop bits being received.	0x0
2	R/W	EPS Even parity select. Controls the type of parity the UART uses during transmission and reception: 0 = odd parity. The UART generates or checks for an odd number of 1s in the data and parity bits. 1 = even parity. The UART generates or checks for an even number of 1s in the data and parity bits. This bit has no effect when the PEN bit disables parity checking and generation.	0x0
1	R/W	PEN Parity enable: 0 = parity is disabled and no parity bit added to the data frame 1 = parity checking and generation is enabled.	0x0
0	R/W	BRK Send break. If this bit is set to 1, a low-level is continually output on the UARTTXD output, after completing transmission of the current character. For the proper execution of the break command, the software must set this bit for at least two complete frames. For normal use, this bit must be cleared to 0.	0x0

Table 832: UART2_CR_REG (0x40001030)

Bit	Mode	Symbol/Description	Reset
15	R/W	CTSEn CTS hardware flow control enable. If this bit is set to 1, CTS hardware flow control is enabled. Data is only transmitted when the nUARTCTS signal is asserted.	0x0
14	R/W	RTSEn RTS hardware flow control enable. If this bit is set to 1, RTS hardware flow control is enabled. Data is only requested when there is space in the receive FIFO for it to be received.	0x0
13	R/W	Out2 This bit is the complement of the UART Out2 (nUARTOut2) modem status output. That is, when the bit is programmed to a 1, the output is 0. For DTE this can be used as Ring Indicator (RI).	0x0
12	R/W	Out1 This bit is the complement of the UART Out1 (nUARTOut1) modem status output. That is, when the bit is programmed to a 1 the output is 0. For DTE this can be used as Data Carrier Detect (DCD).	0x0
11	R/W	RTS	0x0

Bit	Mode	Symbol/Description	Reset
		Request to send. This bit is the complement of the UART request to send, nUARTRTS, modem status output. That is, when the bit is programmed to a 1 then nUARTRTS is LOW	
10	R/W	DTR Data transmit ready. This bit is the complement of the UART data transmit ready, nUARTDTR, modem status output. That is, when the bit is programmed to a 1 then nUARTDTR is LOW	0x0
9	R/W	RXE Receive enable. If this bit is set to 1, the receive section of the UART is enabled.	0x1
8	R/W	TXE Transmit enable. If this bit is set to 1, the transmit section of the UART is enabled.	0x1
7	R/W	LBE Loopback enable. If this bit is set to 1, the UARTTXD path is fed through to the UARTRXD path. When this bit is set, the modem outputs are also fed through to the modem inputs. This bit is cleared to 0 on reset, to disable loopback.	0x0
6:1	R/W	- Reserved	0x0
0	R/W	UARTEN UART enable: 0 = UART is disabled. If the UART is disabled in the middle of transmission or reception, it completes the current character before stopping. 1 = the UART is enabled. Data transmission and reception occurs.	0x0

Table 833: UART2_IFLS_REG (0x40001034)

Bit	Mode	Symbol/Description	Reset
15:6	R/W	- Reserved	0x0
5:3	R/W	RXIFLSEL Receive interrupt FIFO level select. The trigger points for the receive interrupt are as follows: b000 = Receive FIFO becomes 1/8 full b001 = Receive FIFO becomes 1/4 full b010 = Receive FIFO becomes 1/2 full b011 = Receive FIFO becomes 3/4 full b100 = Receive FIFO becomes 7/8 full b101-b111 = Reserved.	0x2
2:0	R/W	TXIFLSEL Transmit interrupt FIFO level select. The trigger points for the transmit interrupt are as follows: b000 = Transmit FIFO becomes 1/8 full b001 = Transmit FIFO becomes 1/4 full b010 = Transmit FIFO becomes 1/2 full b011 = Transmit FIFO becomes 3/4 full b100 = Transmit FIFO becomes 7/8 full b101-b111 = Reserved.	0x2

Table 834: UART2_IMSC_REG (0x40001038)

Bit	Mode	Symbol/Description	Reset
15:11	R/W	-	0x0
		Reserved	
10	R/W	OEIM	0x0
		Overrun error interrupt mask. A read returns the current mask for the UARTOEINTR interrupt. On a write of 1, the mask of the UARTOEINTR interrupt is set. A write of 0 clears the mask.	
9	R/W	BEIM	0x0
		Break error interrupt mask. A read returns the current mask for the UARTBEINTR interrupt. On a write of 1, the mask of the UARTBEINTR interrupt is set. A write of 0 clears the mask.	
8	R/W	PEIM	0x0
		Parity error interrupt mask. A read returns the current mask for the UARTPEINTR interrupt. On a write of 1, the mask of the UARTPEINTR interrupt is set. A write of 0 clears the mask.	
7	R/W	FEIM	0x0
		Framing error interrupt mask. A read returns the current mask for the UARTFEINTR interrupt. On a write of 1, the mask of the UARTFEINTR interrupt is set. A write of 0 clears the mask.	
6	R/W	RTIM	0x0
		Receive timeout interrupt mask. A read returns the current mask for the UARTRTINTR interrupt. On a write of 1, the mask of the UARTRTINTR interrupt is set. A write of 0 clears the mask.	
5	R/W	TXIM	0x0
		Transmit interrupt mask. A read returns the current mask for the UARTTXINTR interrupt. On a write of 1, the mask of the UARTTXINTR interrupt is set. A write of 0 clears the mask.	
4	R/W	RXIM	0x0
		Receive interrupt mask. A read returns the current mask for the UARTRXINTR interrupt. On a write of 1, the mask of the UARTRXINTR interrupt is set. A write of 0 clears the mask.	
3	R/W	DSRMIM	0x0
		nUARTDSR modem interrupt mask. A read returns the current mask for the UARTDSRINTR interrupt. On a write of 1, the mask of the UARTDSRINTR interrupt is set. A write of 0 clears the mask.	
2	R/W	DCDMIM	0x0
		nUARTDCD modem interrupt mask. A read returns the current mask for the UARTDCDINTR interrupt.	

Bit	Mode	Symbol/Description	Reset
		On a write of 1, the mask of the UARTDCDINTR interrupt is set. A write of 0 clears the mask.	
1	R/W	CTSMIM nUARTCTS modem interrupt mask. A read returns the current mask for the UARTCTSINTR interrupt. On a write of 1, the mask of the UARTCTSINTR interrupt is set. A write of 0 clears the mask.	0x0
0	R/W	RIMIM nUARTRI modem interrupt mask. A read returns the current mask for the UARTRIINTR interrupt. On a write of 1, the mask of the UARTRIINTR interrupt is set. A write of 0 clears the mask.	0x0

Table 835: UART2_RIS_REG (0x4000103C)

Bit	Mode	Symbol/Description	Reset
15:11	R	- Reserved	0x0
10	R	OERIS Overrun error interrupt status. Returns the raw interrupt state of the UARTOEINTR interrupt.	0x0
9	R	BERIS Break error interrupt status. Returns the raw interrupt state of the UARTBEINTR interrupt.	0x0
8	R	PERIS Parity error interrupt status. Returns the raw interrupt state of the UARTPEINTR interrupt.	0x0
7	R	FERIS Framing error interrupt status. Returns the raw interrupt state of the UARTFEINTR interrupt.	0x0
6	R	RTRIS Receive timeout interrupt status. Returns the raw interrupt state of the UARTRTINTR interrupt.	0x0
5	R	TXRIS Transmit interrupt status. Returns the raw interrupt state of the UARTTXINTR interrupt.	0x0
4	R	RXRIS Receive interrupt status. Returns the raw interrupt state of the UARTRXINTR interrupt.	0x0
3	R	DSRRMIS nUARTDSR modem interrupt status. Returns the raw interrupt state of the UARTDSRINTR interrupt.	0x1
2	R	DCDRMIS nUARTDCD modem interrupt status. Returns the raw interrupt state of the UARTDCDINTR interrupt.	0x1
1	R	CTSRMIS	0x0

Bit	Mode	Symbol/Description	Reset
		nUARTCTS modem interrupt status. Returns the raw interrupt state of the UARTCTSINTR interrupt.	
0	R	RIRMIS nUARTRI modem interrupt status. Returns the raw interrupt state of the UARTRIINTR interrupt.	0x1

Table 836: UART2_MIS_REG (0x40001040)

Bit	Mode	Symbol/Description	Reset
15:11	R	- Reserved	0x0
10	R	OEMIS Overrun error masked interrupt status. Returns the masked interrupt state of the UARTOEINTR interrupt.	0x0
9	R	BEMIS Break error masked interrupt status. Returns the masked interrupt state of the UARTBEINTR interrupt.	0x0
8	R	PEMIS Parity error masked interrupt status. Returns the masked interrupt state of the UARTPEINTR interrupt.	0x0
7	R	FEMIS Framing error masked interrupt status. Returns the masked interrupt state of the UARTFEINTR interrupt.	0x0
6	R	RTMIS Receive timeout masked interrupt status. Returns the masked interrupt state of the UARTRTINTR interrupt.	0x0
5	R	TXMIS Transmit masked interrupt status. Returns the masked interrupt state of the UARTTXINTR interrupt.	0x0
4	R	RXMIS Receive masked interrupt status. Returns the masked interrupt state of the UARTRXINTR interrupt.	0x0
3	R	DSRMMIS nUARTDSR modem masked interrupt status. Returns the masked interrupt state of the UARTDSRINTR interrupt.	0x0
2	R	DCDMMIS nUARTDCD modem masked interrupt status. Returns the masked interrupt state of the UARTDCDINTR interrupt.	0x0
1	R	CTSMMIS nUARTCTS modem masked interrupt status. Returns the masked interrupt state of the UARTCTSINTR interrupt.	0x0
0	R	RIMMIS nUARTRI modem masked interrupt status. Returns the masked interrupt state of the UARTRIINTR interrupt.	0x0

Table 837: UART2_ICR_REG (0x40001044)

Bit	Mode	Symbol/Description	Reset
15:11	W	-	0x0
		Reserved	
10	W	OEIC	0x0
		Overrun error interrupt clear. Clears the UARTOEINTR interrupt.	
9	W	BEIC	0x0
		Break error interrupt clear. Clears the UARTBEINTR interrupt.	
8	W	PEIC	0x0
		Parity error interrupt clear. Clears the UARTPEINTR interrupt.	
7	W	FEIC	0x0
		Framing error interrupt clear. Clears the UARTFEINTR interrupt.	
6	W	RTIC	0x0
		Receive timeout interrupt clear. Clears the UARTRTINTR interrupt.	
5	W	TXIC	0x0
		Transmit interrupt clear. Clears the UARTTXINTR interrupt.	
4	W	RXIC	0x0
		Receive interrupt clear. Clears the UARTRXINTR interrupt.	
3	W	DSRMIC	0x0
		nUARTDSR modem interrupt clear. Clears the UARTDSRINTR interrupt.	
2	W	DCDMIC	0x0
		nUARTDCD modem interrupt clear. Clears the UARTDCDINTR interrupt.	
1	W	CTSMIC	0x0
		nUARTCTS modem interrupt clear. Clears the UARTCTSINTR interrupt.	
0	W	RIMIC	0x0
		nUARTRI modem interrupt clear. Clears the UARTRIINTR interrupt.	

Table 838: UART2_DMACR_REG (0x40001048)

Bit	Mode	Symbol/Description	Reset
15:3	R/W	-	0x0
		Reserved	
2	R/W	DMAONERR	0x0
		DMA on error. If this bit is set to 1, the DMA receive request outputs, UARTRXDMASREQ or UARTRXDMABREQ, are disabled when the UART error interrupt is asserted	
1	R/W	TXDMAE	0x0
		Transmit DMA enable. If this bit is set to 1, DMA for the transmit FIFO is enabled.	
0	R/W	RXDMAE	0x0
		Receive DMA enable. If this bit is set to 1, DMA for the receive FIFO is enabled.	

Table 839: UART2_WA_REG (0x4000104C)

Bit	Mode	Symbol/Description	Reset
0	R/W	WAE DMA word access enable bit. If this bit is set to 1, DMA word access mode is enabled.	0x1

Table 840: UART2_SWFC_REG (0x40001050)

Bit	Mode	Symbol/Description	Reset
0	R/W	SWFCE DMA software flow control enable bit. If this bit is set to 1, DMA software flow control mode is enabled.	0x0

Table 841: UART2_RS485EN_REG (0x40001054)

Bit	Mode	Symbol/Description	Reset
0	R/W	RS485E DMA RS485 mode enable bit. If this bit is set to 1, DMA RS485 mode is enabled.	0x0

Table 842: Register map UART3

Address	Register	Description
0x40002000	UART3_DR_REG	UART Data Register
0x40002004	UART3_RSR_REG	UART Receive Status (Read), Error Clear (Write) Register
0x40002018	UART3_FR_REG	UART Flag Register
0x40002024	UART3_IBRD_REG	UART Integer Baud Rate Divisor Register
0x40002028	UART3_FBRD_REG	UART Fractional Baud Rate Divisor Register
0x4000202c	UART3_LCR_H_REG	UART Line Control Register (High Byte)
0x40002030	UART3_CR_REG	UART Control Register
0x40002034	UART3_IFLS_REG	UART Interrupt FIFO Level Select Register
0x40002038	UART3_IMSC_REG	UART Interrupt Mask Set/Clear Register
0x4000203c	UART3_RIS_REG	UART Raw Interrupt Status Register
0x40002040	UART3_MIS_REG	UART Masked Interrupt Status Register
0x40002044	UART3_ICR_REG	UART Interrupt Clear Register
0x40002048	UART3_DMACR_REG	UART DMA Control Register
0x4000204c	UART3_WA_REG	UART Word Access Enable Register
0x40002050	UART3_SWFC_REG	UART Software Flow Control Enable Register
0x40002054	UART3_RS485EN_REG	UART RS485 Mode Enable Register

Table 843: UART3_DR_REG (0x40002000)

Bit	Mode	Symbol/Description	Reset
15:12	R/W	- Reserved	0x0
11	R/W	DR_OE	0x0

Bit	Mode	Symbol/Description	Reset
		Overrun error. This bit is set to 1 if data is received and the receive FIFO is already full. This is cleared to 0 when there is an empty space in the FIFO and a new character can be written to it.	
10	R/W	DR_BE Break error. This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state), and the next valid start bit is received.	0x0
9	R/W	DR_PE Parity error. When set to 1, it indicates that the parity of the received data character does not match the parity that the EPS and SPS bits in the Line Control Register, UARTLCR_H. In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0
8	R/W	DR_FE Framing error. When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0
7:0	R/W	DR_DATA Receive (read) data charater Transmit (write) data character	0x0

Table 844: UART3_RSR_REG (0x40002004)

Bit	Mode	Symbol/Description	Reset
7:4	R/W	- Reserved	0x0
3	R/W	RSR_OE Overrun error. This bit is set to 1 if data is received and the FIFO is already full. This bit is cleared to 0 by a write to the register. The FIFO contents remain valid because no more data is written when the FIFO is full, only the contents of the shift register are overwritten. The CPU must now read the data, to empty the FIFO.	0x0
2	R/W	RSR_BE Break error. This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity, and stop bits). This bit is cleared to 0 after a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state) and the next valid start bit is received.	0x0
1	R/W	RSR_PE Parity error. When set to 1, it indicates that the parity of the received data character does not match the parity that the EPS and SPS bits in the Line Control Register, UARTLCR_H.	0x0

Bit	Mode	Symbol/Description	Reset
		This bit is cleared to 0 by a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO.	
0	R/W	RSR_FE Framing error. When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). This bit is cleared to 0 by a write to the register. In FIFO mode, this error is associated with the character at the top of the FIFO.	0x0

Table 845: UART3_FR_REG (0x40002018)

Bit	Mode	Symbol/Description	Reset
15:9	R	- Reserved	0x0
8	R	RI Ring indicator. This bit is the complement of the UART ring indicator, nUARTRI, modem status input. That is, the bit is 1 when nUARTRI is LOW.	0x0
7	R	TXFE Transmit FIFO empty. The meaning of this bit depends on the state of the FEN bit in the Line Control Register, UARTLCR_H If the FIFO is disabled, this bit is set when the transmit holding register is empty. If the FIFO is enabled, the TXFE bit is set when the transmit FIFO is empty. This bit does not indicate if there is data in the transmit shift register.	0x1
6	R	RXFF Receive FIFO full. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the receive holding register is full. If the FIFO is enabled, the RXFF bit is set when the receive FIFO is full.	0x0
5	R	TXFF Transmit FIFO full. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the transmit holding register is full. If the FIFO is enabled, the TXFF bit is set when the transmit FIFO is full.	0x0
4	R	RXFE Receive FIFO empty. The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H Register. If the FIFO is disabled, this bit is set when the receive holding register is empty. If the FIFO is enabled, the RXFE bit is set when the receive FIFO is empty.	0x1
3	R	BUSY UART busy. If this bit is set to 1, the UART is busy transmitting data. This bit remains set until the complete byte, including all the stop bits, has been sent from the shift register. This bit is set as soon as the transmit FIFO becomes non-empty, regardless of whether the UART is enabled or not.	0x0
2	R	DCD Data carrier detect. This bit is the complement of the UART data carrier detect, nUARTDCD, modem status input.	0x0

Bit	Mode	Symbol/Description	Reset
		That is, the bit is 1 when nUARTDCD is LOW.	
1	R	DSR Data set ready. This bit is the complement of the UART data set ready, nUARTDSR, modem status input. That is, the bit is 1 when nUARTDSR is LOW.	0x0
0	R	CTS Clear to send. This bit is the complement of the UART clear to send, nUARTCTS, modem status input. That is, the bit is 1 when nUARTCTS is LOW.	0x1

Table 846: UART3_IBRD_REG (0x40002024)

Bit	Mode	Symbol/Description	Reset
15:0	R/W	BAUD_DIVINT The integer baud rate divisor. These bits are cleared to 0 on reset.	0x0

Table 847: UART3_FBRD_REG (0x40002028)

Bit	Mode	Symbol/Description	Reset
5:0	R/W	BAUD_DIVFRAC The fractional baud rate divisor. These bits are cleared to 0 on reset.	0x0

Table 848: UART3_LCR_H_REG (0x4000202C)

Bit	Mode	Symbol/Description	Reset
15:8	R/W	- Reserved	0x0
7	R/W	SPS Stick parity select. 0 = stick parity is disabled 1 = either: If the EPS bit is 0 then the parity bit is transmitted and checked as a 1 If the EPS bit is 1 then the parity bit is transmitted and checked as a 0. This bit has no effect when the PEN bit disables parity checking and generation.	0x0
6:5	R/W	WLEN Word length. These bits indicate the number of data bits transmitted or received in a frame as follows: b11 = 8 bits b10 = 7 bits b01 = 6 bits b00 = 5 bits.	0x0
4	R/W	FEN Enable FIFOs: 0 = FIFOs are disabled (character mode) that is, the FIFOs become 1-byte-deep holding registers	0x0

Bit	Mode	Symbol/Description	Reset
		1 = transmit and receive FIFO buffers are enabled (FIFO mode).	
3	R/W	STP2 Two stop bits select. If this bit is set to 1, two stop bits are transmitted at the end of the frame. The receive logic does not check for two stop bits being received.	0x0
2	R/W	EPS Even parity select. Controls the type of parity the UART uses during transmission and reception: 0 = odd parity. The UART generates or checks for an odd number of 1s in the data and parity bits. 1 = even parity. The UART generates or checks for an even number of 1s in the data and parity bits. This bit has no effect when the PEN bit disables parity checking and generation.	0x0
1	R/W	PEN Parity enable: 0 = parity is disabled and no parity bit added to the data frame 1 = parity checking and generation is enabled.	0x0
0	R/W	BRK Send break. If this bit is set to 1, a low-level is continually output on the UARTTXD output, after completing transmission of the current character. For the proper execution of the break command, the software must set this bit for at least two complete frames. For normal use, this bit must be cleared to 0.	0x0

Table 849: UART3_CR_REG (0x40002030)

Bit	Mode	Symbol/Description	Reset
15	R/W	CTSEn CTS hardware flow control enable. If this bit is set to 1, CTS hardware flow control is enabled. Data is only transmitted when the nUARTCTS signal is asserted.	0x0
14	R/W	RTSEn RTS hardware flow control enable. If this bit is set to 1, RTS hardware flow control is enabled. Data is only requested when there is space in the receive FIFO for it to be received.	0x0
13	R/W	Out2 This bit is the complement of the UART Out2 (nUARTOut2) modem status output. That is, when the bit is programmed to a 1, the output is 0. For DTE this can be used as Ring Indicator (RI).	0x0
12	R/W	Out1 This bit is the complement of the UART Out1 (nUARTOut1) modem status output. That is, when the bit is programmed to a 1 the output is 0. For DTE this can be used as Data Carrier Detect (DCD).	0x0
11	R/W	RTS Request to send. This bit is the complement of the UART request to send, nUARTRTS, modem status output. That is, when the bit is programmed to a 1 then nUARTRTS is LOW	0x0
10	R/W	DTR	0x0

Bit	Mode	Symbol/Description	Reset
		Data transmit ready. This bit is the complement of the UART data transmit ready, nUARTDTR, modem status output. That is, when the bit is programmed to a 1 then nUARTDTR is LOW	
9	R/W	RXE Receive enable. If this bit is set to 1, the receive section of the UART is enabled.	0x1
8	R/W	TXE Transmit enable. If this bit is set to 1, the transmit section of the UART is enabled.	0x1
7	R/W	LBE Loopback enable. If this bit is set to 1, the UARTTXD path is fed through to the UARTRXD path. When this bit is set, the modem outputs are also fed through to the modem inputs. This bit is cleared to 0 on reset, to disable loopback.	0x0
6:1	R/W	- Reserved	0x0
0	R/W	UARTEN UART enable: 0 = UART is disabled. If the UART is disabled in the middle of transmission or reception, it completes the current character before stopping. 1 = the UART is enabled. Data transmission and reception occurs.	0x0

Table 850: UART3_IFLS_REG (0x40002034)

Bit	Mode	Symbol/Description	Reset
15:6	R/W	- Reserved	0x0
5:3	R/W	RXIFLSEL Receive interrupt FIFO level select. The trigger points for the receive interrupt are as follows: b000 = Receive FIFO becomes 1/8 full b001 = Receive FIFO becomes 1/4 full b010 = Receive FIFO becomes 1/2 full b011 = Receive FIFO becomes 3/4 full b100 = Receive FIFO becomes 7/8 full b101-b111 = Reserved.	0x2
2:0	R/W	TXIFLSEL Transmit interrupt FIFO level select. The trigger points for the transmit interrupt are as follows: b000 = Transmit FIFO becomes 1/8 full b001 = Transmit FIFO becomes 1/4 full b010 = Transmit FIFO becomes 1/2 full b011 = Transmit FIFO becomes 3/4 full b100 = Transmit FIFO becomes 7/8 full b101-b111 = Reserved.	0x2

Table 851: UART3_IMSC_REG (0x40002038)

Bit	Mode	Symbol/Description	Reset
15:11	R/W	-	0x0
		Reserved	
10	R/W	OEIM	0x0
		Overrun error interrupt mask. A read returns the current mask for the UARTOEINTR interrupt. On a write of 1, the mask of the UARTOEINTR interrupt is set. A write of 0 clears the mask.	
9	R/W	BEIM	0x0
		Break error interrupt mask. A read returns the current mask for the UARTBEINTR interrupt. On a write of 1, the mask of the UARTBEINTR interrupt is set. A write of 0 clears the mask.	
8	R/W	PEIM	0x0
		Parity error interrupt mask. A read returns the current mask for the UARTPEINTR interrupt. On a write of 1, the mask of the UARTPEINTR interrupt is set. A write of 0 clears the mask.	
7	R/W	FEIM	0x0
		Framing error interrupt mask. A read returns the current mask for the UARTFEINTR interrupt. On a write of 1, the mask of the UARTFEINTR interrupt is set. A write of 0 clears the mask.	
6	R/W	RTIM	0x0
		Receive timeout interrupt mask. A read returns the current mask for the UARTRTINTR interrupt. On a write of 1, the mask of the UARTRTINTR interrupt is set. A write of 0 clears the mask.	
5	R/W	TXIM	0x0
		Transmit interrupt mask. A read returns the current mask for the UARTTXINTR interrupt. On a write of 1, the mask of the UARTTXINTR interrupt is set. A write of 0 clears the mask.	
4	R/W	RXIM	0x0
		Receive interrupt mask. A read returns the current mask for the UARTRXINTR interrupt. On a write of 1, the mask of the UARTRXINTR interrupt is set. A write of 0 clears the mask.	
3	R/W	DSRMIM	0x0
		nUARTDSR modem interrupt mask. A read returns the current mask for the UARTDSRINTR interrupt. On a write of 1, the mask of the UARTDSRINTR interrupt is set. A write of 0 clears the mask.	
2	R/W	DCDMIM	0x0
		nUARTDCD modem interrupt mask. A read returns the current mask for the UARTDCDINTR interrupt. On a write of 1, the mask of the UARTDCDINTR interrupt is set. A write of 0 clears the mask.	

Bit	Mode	Symbol/Description	Reset
1	R/W	CTSMIM nUARTCTS modem interrupt mask. A read returns the current mask for the UARTCTSINTR interrupt. On a write of 1, the mask of the UARTCTSINTR interrupt is set. A write of 0 clears the mask.	0x0
0	R/W	RIMIM nUARTRI modem interrupt mask. A read returns the current mask for the UARTRIINTR interrupt. On a write of 1, the mask of the UARTRIINTR interrupt is set. A write of 0 clears the mask.	0x0

Table 852: UART3_RIS_REG (0x4000203C)

Bit	Mode	Symbol/Description	Reset
15:11	R	- Reserved	0x0
10	R	OERIS Overrun error interrupt status. Returns the raw interrupt state of the UARTOEINTR interrupt.	0x0
9	R	BERIS Break error interrupt status. Returns the raw interrupt state of the UARTBEINTR interrupt.	0x0
8	R	PERIS Parity error interrupt status. Returns the raw interrupt state of the UARTPEINTR interrupt.	0x0
7	R	FERIS Framing error interrupt status. Returns the raw interrupt state of the UARTFEINTR interrupt.	0x0
6	R	RTRIS Receive timeout interrupt status. Returns the raw interrupt state of the UARTRTINTR interrupt.	0x0
5	R	TXRIS Transmit interrupt status. Returns the raw interrupt state of the UARTTXINTR interrupt.	0x0
4	R	RXRIS Receive interrupt status. Returns the raw interrupt state of the UARTRXINTR interrupt.	0x0
3	R	DSRRMIS nUARTDSR modem interrupt status. Returns the raw interrupt state of the UARTDSRINTR interrupt.	0x1
2	R	DCDRMIS nUARTDCD modem interrupt status. Returns the raw interrupt state of the UARTDCDINTR interrupt.	0x1
1	R	CTSRMIS nUARTCTS modem interrupt status. Returns the raw interrupt state of the UARTCTSINTR interrupt.	0x0

Bit	Mode	Symbol/Description	Reset
0	R	RIRMIS nUARTRI modem interrupt status. Returns the raw interrupt state of the UARTRIINTR interrupt.	0x1

Table 853: UART3_MIS_REG (0x40002040)

Bit	Mode	Symbol/Description	Reset
15:11	R	- Reserved	0x0
10	R	OEMIS Overrun error masked interrupt status. Returns the masked interrupt state of the UARTOEINTR interrupt.	0x0
9	R	BEMIS Break error masked interrupt status. Returns the masked interrupt state of the UARTBEINTR interrupt.	0x0
8	R	PEMIS Parity error masked interrupt status. Returns the masked interrupt state of the UARTPEINTR interrupt.	0x0
7	R	FEMIS Framing error masked interrupt status. Returns the masked interrupt state of the UARTFEINTR interrupt.	0x0
6	R	RTMIS Receive timeout masked interrupt status. Returns the masked interrupt state of the UARTRTINTR interrupt.	0x0
5	R	TXMIS Transmit masked interrupt status. Returns the masked interrupt state of the UARTTXINTR interrupt.	0x0
4	R	RXMIS Receive masked interrupt status. Returns the masked interrupt state of the UARTRXINTR interrupt.	0x0
3	R	DSRMMIS nUARTDSR modem masked interrupt status. Returns the masked interrupt state of the UARTDSRINTR interrupt.	0x0
2	R	DCDMMIS nUARTDCD modem masked interrupt status. Returns the masked interrupt state of the UARTDCDINTR interrupt.	0x0
1	R	CTSMMIS nUARTCTS modem masked interrupt status. Returns the masked interrupt state of the UARTCTSINTR interrupt.	0x0
0	R	RIMMIS nUARTRI modem masked interrupt status. Returns the masked interrupt state of the UARTRIINTR interrupt.	0x0

Table 854: UART3_ICR_REG (0x40002044)

Bit	Mode	Symbol/Description	Reset
15:11	W	-	0x0
		Reserved	
10	W	OEIC	0x0
		Overrun error interrupt clear. Clears the UARTOEINTR interrupt.	
9	W	BEIC	0x0
		Break error interrupt clear. Clears the UARTBEINTR interrupt.	
8	W	PEIC	0x0
		Parity error interrupt clear. Clears the UARTPEINTR interrupt.	
7	W	FEIC	0x0
		Framing error interrupt clear. Clears the UARTFEINTR interrupt.	
6	W	RTIC	0x0
		Receive timeout interrupt clear. Clears the UARTRTINTR interrupt.	
5	W	TXIC	0x0
		Transmit interrupt clear. Clears the UARTTXINTR interrupt.	
4	W	RXIC	0x0
		Receive interrupt clear. Clears the UARTRXINTR interrupt.	
3	W	DSRMIC	0x0
		nUARTDSR modem interrupt clear. Clears the UARTDSRINTR interrupt.	
2	W	DCDMIC	0x0
		nUARTDCD modem interrupt clear. Clears the UARTDCDINTR interrupt.	
1	W	CTSMIC	0x0
		nUARTCTS modem interrupt clear. Clears the UARTCTSINTR interrupt.	
0	W	RIMIC	0x0
		nUARTRI modem interrupt clear. Clears the UARTRIINTR interrupt.	

Table 855: UART3_DMACR_REG (0x40002048)

Bit	Mode	Symbol/Description	Reset
15:3	R/W	-	0x0
		Reserved	
2	R/W	DMAONERR	0x0
		DMA on error. If this bit is set to 1, the DMA receive request outputs, UARTRXDMASREQ or UARTRXDMABREQ, are disabled when the UART error interrupt is asserted	
1	R/W	TXDMAE	0x0
		Transmit DMA enable. If this bit is set to 1, DMA for the transmit FIFO is enabled.	
0	R/W	RXDMAE	0x0
		Receive DMA enable. If this bit is set to 1, DMA for the receive FIFO is enabled.	

Table 856: UART3_WA_REG (0x4000204C)

Bit	Mode	Symbol/Description	Reset
0	R/W	WAE DMA word access enable bit. If this bit is set to 1, DMA word access mode is enabled.	0x1

Table 857: UART3_SWFC_REG (0x40002050)

Bit	Mode	Symbol/Description	Reset
0	R/W	SWFCE DMA software flow control enable bit. If this bit is set to 1, DMA software flow control mode is enabled.	0x0

Table 858: UART3_RS485EN_REG (0x40002054)

Bit	Mode	Symbol/Description	Reset
0	R/W	RS485E DMA RS485 mode enable bit. If this bit is set to 1, DMA RS485 mode is enabled.	0x0

10.37 ID Registers

Table 859: Register map Version

Address	Register	Description
0x40070200	CHIP_ID1_REG	Chip identification register 1.
0x40070204	CHIP_ID2_REG	Chip identification register 2.
0x40070208	CHIP_ID3_REG	Chip identification register 3.
0x4007020c	CHIP_ID4_REG	Chip identification register 4.
0x40070210	CHIP_SWC_REG	Software compatibility register.
0x40070214	CHIP_REVISION_REG	Chip revision register.
0x400702f8	CHIP_TEST1_REG	Chip test register 1.
0x400702fc	CHIP_TEST2_REG	Chip test register 2.

Table 860: CHIP_ID1_REG (0x40070200)

Bit	Mode	Symbol/Description	Reset
31:8	R	- Reserved	0x0
7:0	R	CHIP_ID1 First character of device type "3095" in ASCII.	0x33

Table 861: CHIP_ID2_REG (0x40070204)

Bit	Mode	Symbol/Description	Reset
31:8	R	- Reserved	0x0
7:0	R	CHIP_ID2	0x30

Bit	Mode	Symbol/Description	Reset
		Second character of device type "3095" in ASCII.	

Table 862: CHIP_ID3_REG (0x40070208)

Bit	Mode	Symbol/Description	Reset
31:8	R	-	0x0
		Reserved	
7:0	R	CHIP_ID3	0x39
		Third character of device type "3095"" in ASCII.	

Table 863: CHIP_ID4_REG (0x4007020C)

Bit	Mode	Symbol/Description	Reset
31:8	R	-	0x0
		Reserved	
7:0	R	CHIP_ID4	0x35
		Fourth character of device type "3095" in ASCII.	

Table 864: CHIP_SWC_REG (0x40070210)

Bit	Mode	Symbol/Description	Reset
31:4	R	-	0x0
		Reserved	
3:0	R	CHIP_SWC	0x0
		Software compatibility code. Integer (default = 0) which is incremented if a silicon change has impact on the CPU Firmware. Can be used by software developers to write silicon revision dependent code.	

Table 865: CHIP_REVISION_REG (0x40070214)

Bit	Mode	Symbol/Description	Reset
31:8	R	-	0x0
		Reserved	
7:0	R	CHIP_REVISION	0x42
		Chip version, corresponds to type number in ASCII. 0x41 = A, 0x42 = B	

Table 866: CHIP_TEST1_REG (0x400702F8)

Bit	Mode	Symbol/Description	Reset
31:8	R	-	0x0
		Reserved	
7:0	R	CHIP_LAYOUT_REVISION	0x41
		Chip layout revision, corresponds to type number in ASCII. 0x41 = A, 0x43 = C, and so forth for the WLCSP package	

Bit	Mode	Symbol/Description	Reset
		0x42 = B, 0x44 = D, and so forth for the VFBGA package	

Table 867: CHIP_TEST2_REG (0x400702FC)

Bit	Mode	Symbol/Description	Reset
31:4	R	-	0x0
		Reserved	
3:0	R	CHIP_METAL_OPTION	0x0
		Chip metal option value.	

11. Package Information

11.1 Moisture Sensitivity Level (MSL)

The MSL is an indicator for the maximum allowable time period (floor lifetime) in which a moisture sensitive plastic device, once removed from the dry bag, can be exposed to an environment with a maximum temperature of 30 °C and a maximum relative humidity of 60% relative humidity (RH) before the solder reflow process.

- FCQFN66 package is qualified for MSL 3.
- WLCSP70 is qualified for MSL 1.

Table 868: MSL definitions

MSL level	Floor lifetime
MSL 4	72 hours
MSL 3	168 hours
MSL 2A	4 weeks
MSL 2	1 year
MSL 1	Unlimited at 30 °C/85%RH

11.2 WLCSP Handling

For WLCSP, underfill material is required for post SMT applications to have a good board level reliability such as drop test.

11.3 Soldering Information

Refer to the JEDEC standard J-STD-020 for relevant soldering information. This document can be downloaded from http://www.jedec.org.

11.4 Package Outline Drawings

Information on the latest version of the package dimensions or mountings is accessible on the Renesas Electronics website – 66-FCQFN and 70-WLCSP.

12. Revision History

Revision	Date	Description
1.02	Apr 16, 2026	▪ Updated Section 7.6.3.
1.01	Jan 28, 2026	▪ Updated Figure 51, Figures 61-68. ▪ Updated Table 11, Table 20, and Table 21.
1.00	Nov 25, 2025	▪ Updated pin information, Figure 1, Section 8. ▪ Added ECAD information as appendix. ▪ Updated Radio Characteristics and typos in registers.
0.22	Sept 2, 2025	▪ Updated product naming. ▪ RF specification updated for QFN and WLCSP packages.
0.21	Mar 13, 2025	Editorial changes and template updates.
0.20	Feb 28, 2025	▪ VESD_CDM values are changed to +/-450 V. ▪ VBAT operating range low limit changed to 1.9 V from 1.8 V. ▪ RF parameters are refined. ▪ OTP map refined and register map added. ▪ Ordering information and package information are added.
0.10	Jan 31, 2025	▪ VESD_CDM values are changed. ▪ SPI maximum supported clock frequency added at IO = 1.8 V condition. ▪ Radio characteristics are updated. ▪ P0_01 wake-up support description is removed. ▪ XTAL32K_P/XTAL32K_M locations are changed.
0.09	Nov 07, 2024	▪ Updated pin information for WLCSP package. ▪ Updated Radio Characteristics. ▪ Typo fixed for O/QSPI, GP Timer. ▪ POD added for WLCSP/FCQFN packages. ▪ Updated for PRNG descriptions. ▪ Typo fixed for SDIO/eMMC CLK/CMD pin map. ▪ Typo fixed for PWM generation part (from 216 to 216) ▪ AC/DC parameters are updated.
0.04	Dec 13, 2023	Added Core System and Peripheral sections.
0.03	Jul 07, 2023	▪ Changed DA16400 to RRQ61000. ▪ Updated Software System Diagram.
0.02	Jun 02, 2023	▪ Changed M33 max clock to 160 MHz. ▪ Changed system RAM to 704 kB.
0.01	Mar 15, 2023	Preliminary release.

RoHS Compliance

Renesas Electronics' suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.

Appendix A ECAD Design Information

This appendix contains information that supports the development of the PCB ECAD model for this device. It is intended to be used by PCB designers.

A.1 Part Number Indexing

Orderable Part Number	Number of Pins	Package Type	Package Code/POD Number
R7SA6W1AEDZNR	66	QFN	FQ0066AA/PSC-4987-01
R7SA6W1BEDZNR	66	QFN	FQ0066AA/PSC-4987-01
R7SA6W1CEDZNR	66	QFN	FQ0066AA/PSC-4987-01
R7SA6W1AEDZDD	70	WLCSP	WD0070AA/PSC-5064-01
R7SA6W1BEDZDD	70	WLCSP	WD0070AA/PSC-5064-01
R7SA6W1CEDZDD	70	WLCSP	WD0070AA/PSC-5064-01

A.2 Symbol Pin Information

A.2.1 66-QFN

Pin Number	Primary Pin Name	Primary electrical type	Alternate Pin Name(s)
1	VSSRFQ_1	Power	-
2	VDDRF_PA	Power	-
3	RFTX_2G	Output	-
4	RFRX_2G	Input	-
5	VSSRF_3	Power	-
6	P0_00	I/O	RTC_WAKE_UP
7	P0_01	I/O	sen_out
8	P0_02	I/O	XTAL32K_M
9	P0_03	I/O	XTAL32K_P
10	P0_04	I/O	ADC0/eMMC_DIO4
11	P0_05	I/O	ADC1/eMMC_DIO5
12	P0_06	I/O	ADC2/eMMC_DIO6
13	P0_07	I/O	ADC3/MCLK/eMMC_DIO7
14	P0_08	I/O	QSPIR_CLK/SDIO0_CLK/eMMC_CLK
15	P0_10	I/O	QSPIR_D0/SDIO0_D0/eMMC_DIO0
16	VDDIO_DIO1	Power	-
17	P0_11	I/O	QSPIR_D1/SDIO0_D1/eMMC_DIO1
18	P0_12	I/O	QSPIR_D2/SDIO0_D2/eMMC_DIO2
19	P0_13	I/O	QSPIR_D3/SDIO0_D3/eMMC_DIO3
20	VDDD	Power	-
21	VDDA_DCDC_ANA	Power	-
22	LX_ANA	Power	-
23	VSSA_DCDC_ANA	Power	-
24	VBAT	Power	-
25	VSSA_DCDC_PA	Power	-
26	LX_PA	Power	-

Pin Number	Primary Pin Name	Primary electrical type	Alternate Pin Name(s)
27	VDDA_DCDC_PA	Power	-
28	P1_03	I/O	OQSPI_D7/eMMC_DIO7
29	VDDIO_FDIO	Power	-
30	P1_04	I/O	OQSPI_D0
31	P1_05	I/O	OQSPI_D1
32	P1_06	I/O	OQSPI_D2
33	P1_07	I/O	OQSPI_D3
34	P1_08	I/O	OQSPI_CLK
35	P1_09	I/O	OQSPI_CS
36	P1_10	I/O	eMMC_CMD/SDIO1_CMD
37	VDDIO_DIO2	Power	-
38	P1_11	I/O	eMMC_CLK/SDIO1_CLK
39	P1_12	I/O	eMMC_DIO0/SDIO1_D0
40	P1_13	I/O	eMMC_DIO1/SDIO1_D1
41	P1_14	I/O	eMMC_DIO2/SDIO1_D2
42	SWCLK	Input	-
43	SWDIO	I/O	-
44	P1_15	I/O	eMMC_DIO3/SDIO1_D3
45	XTAL40M_P	Input	-
46	XTAL40M_M	Output	-
47	VSSRF_PLL	Power	-
48	VDDRF	Power	-
49	VSSRFQ_2	Power	-
50	RF_5G	I/O	-
51	VSSRF_1	Power	-
52	VSSRF_IF	Power	-
53	ESDN	Power	-
54	VSSD_2	Power	-
55	RST_N	Input	-
56	VSSA_4	Power	-
57	VSSIO_2	Power	-
58	P1_02	I/O	OQSPI_D6/eMMC_DIO6
59	VSSA_1	Power	-
60	VSSIO_1	Power	-
61	VSSIO_3	Power	-
62	P1_01	I/O	OQSPI_D5/eMMC_DIO5
63	P0_09	I/O	QSPIR_CS/SDIO0_CMD/eMMC_CMD
64	VSSD_1	Power	-
65	VSSA_2	Power	-
66	P1_00	I/O	OQSPI_D4/eMMC_DIO4

A.2.2 70-WLCSP

Pin Number	Primary Pin Name	Primary electrical type	Alternate Pin Name(s)
A1	VSSRF_4	Power	-
A3	RF_5G	I/O	-
A5	VDDRF	Power	-
A7	XTAL40M_M	Output	-
A9	P1_15	I/O	eMMC_DIO3/SDIO1_D3
A11	P1_13	I/O	eMMC_DIO1/SDIO1_D1
B2	VSSRF_5	Power	-
B4	VSSRF_6	Power	-
B6	VSSRF_PLL	Power	-
B8	XTAL40M_P	Input	-
B10	P1_12	I/O	eMMC_DIO0/SDIO1_D0
C1	VDDRF_PA	Power	-
C3	VSSRF_2	Power	-
C5	VSSRF_IF	Power	-
C9	P1_11	I/O	eMMC_CLK/SDIO1_CLK
C11	VDDIO_DIO2	Power	-
D2	VSSRF_1	Power	-
D4	VSSA_5	Power	-
D6	ESDN	Power	-
D8	VSSIO_2	Power	-
D10	VSSD_2	Power	-
E3	VSSA_4	Power	-
E5	SWDIO	I/O	-
E7	P1_14	I/O	eMMC_DIO2/SDIO1_D2
E9	P1_10	I/O	eMMC_CMD/SDIO1_CMD
E11	P1_08	I/O	OQSPI_CLK
F2	RFTX_2G	Output	-
F4	RST_N	Input	-
F6	P1_09	I/O	OQSPI_CS
F8	P1_02	I/O	OQSPI_D6/eMMC_DIO6
F10	P1_07	I/O	OQSPI_D3
G1	RFRX_2G	Input	-
G3	VSSA_3	Power	-
G5	SWCLK	Input	-
G7	P1_01	I/O	OQSPI_D5/eMMC_DIO5
G9	P1_05	I/O	OQSPI_D1
G11	P1_06	I/O	OQSPI_D2
H2	VSSRF_3	Power	-
H4	P0_06	I/O	ADC2/eMMC_DIO6
H6	P0_11	I/O	QSPIR_D1/SDIO0_D1/eMMC_DIO1

Pin Number	Primary Pin Name	Primary electrical type	Alternate Pin Name(s)
H8	P1_00	I/O	OQSPI_D4/eMMC_DIO4
H10	P1_04	I/O	OQSPI_D0
J1	VSSA_1	Power	-
J3	P0_00	I/O	RTC_WAKE_UP
J5	P0_10	I/O	QSPIR_D0/SDIO0_D0/eMMC_DIO0
J7	P1_03	I/O	OQSPI_D7/eMMC_DIO7
J9	VSSIO_3	Power	-
J11	VDDIO_FDIO	Power	-
K2	P0_01	I/O	sen_out
K4	P0_07	I/O	ADC3/MCLK/eMMC_DIO7
K6	VSSD_1	Power	-
K8	VSSA_2	Power	-
K10	VDDA_DCDC_PA	Power	-
L1	P0_02	I/O	XTAL32K_M
L3	P0_04	I/O	ADC0/eMMC_DIO4
L5	VSSIO_1	Power	-
L7	P0_13	I/O	QSPIR_D3/SDIO0_D3/eMMC_DIO3
L9	VSSA_DCDC_PA	Power	-
L11	LX_PA	Power	-
M2	P0_03	I/O	XTAL32K_P
M4	P0_09	I/O	QSPIR_CS/SDIO0_CMD/eMMC_CMD
M6	P0_12	I/O	QSPIR_D2/SDIO0_D2/eMMC_DIO2
M8	VDDA_DCDC_ANA	Power	-
M10	VSSA_DCDC_ANA	Power	-
N1	P0_05	I/O	ADC1/eMMC_DIO5
N3	P0_08	I/O	QSPIR_CLK/SDIO0_CLK/eMMC_CLK
N5	VDDIO_DIO1	Power	-
N7	VDDD	Power	-
N9	LX_ANA	Power	-
N11	VBAT	Power	-

A.3 Symbol Parameters

Orderable Part Number	Min Input Voltage	Max Input Voltage	Min Operating Temperature	Max Operating Temperature	Max Frequency	RAM Size	Interface	Number of A/D Converters	Number of SPI Channels	Number of I2C Channels	Number of I2S Channels	Number of UART Channels	Number of Timers/ Counters	Number of Programmable I/O
R7SA6W1AEDZNR	1.62 V	3.6 V	-40 °C	85 °C	160 MHz	768 kB	Radio, ADC, xSPI, I2C, I2S, SDIO, MMC, UART	4	2	2	1	3	8	28
R7SA6W1BEDZNR	1.62 V	3.6 V	-40 °C	85 °C	160 MHz	768 kB	Radio, ADC, xSPI, I2C, I2S, SDIO, MMC, UART	4	2	2	1	3	8	28
R7SA6W1CEDZNR	1.62 V	3.6 V	-40 °C	85 °C	160 MHz	768 kB	Radio, ADC, xSPI, I2C, I2S, SDIO, MMC, UART	4	2	2	1	3	8	28
R7SA6W1AEDZDD	1.62 V	3.6 V	-40 °C	85 °C	160 MHz	768 kB	Radio, ADC, xSPI, I2C, I2S, SDIO, MMC, UART	4	2	2	1	3	8	28
R7SA6W1BEDZDD	1.62 V	3.6 V	-40 °C	85 °C	160 MHz	768 kB	Radio, ADC, xSPI, I2C, I2S, SDIO, MMC, UART	4	2	2	1	3	8	28
R7SA6W1CEDZDD	1.62 V	3.6 V	-40 °C	85 °C	160 MHz	768 kB	Radio, ADC, xSPI, I2C, I2S, SDIO, MMC, UART	4	2	2	1	3	8	28

1. Available RAM for CPU (M33).
2. Maximum available I/Os when alternate functions (SPI, I2C, I2S) are not used.

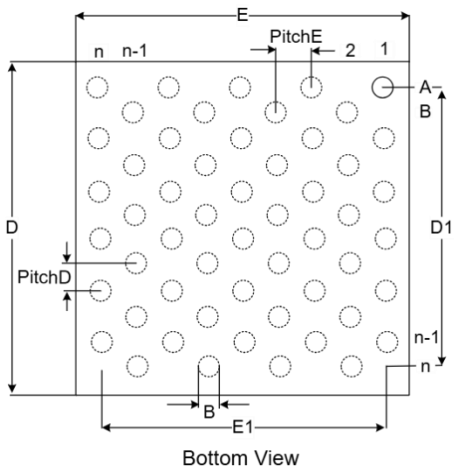
A.4 Footprint Design Information

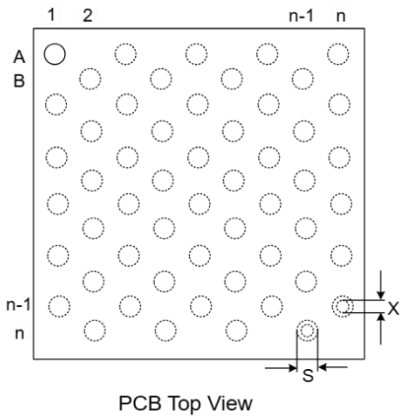
A.4.1 66-QFN

Follow the POD drawing for footprint generation for QFN package.

A.4.2 70-WLCSP

IPC Footprint Type	Package Code/POD Number	Number of Pins
WLCSP	WD0070AA/PSC-5064-01	70

Description	Dimension	Value (mm)	Diagram
Minimum body Length (vertical side)	Dmin	4.045	
Maximum body Length (vertical side)	Dmax	4.085	
Average length of grid (vertical side)	D1ave	3.600	
Minimum body Width (horizontal side)	Emin	3.451	
Maximum body Width (horizontal side)	Emax	3.491	
Average length of grid (horizontal side)	E1ave	3.000	
Average ball diameter	Bnom	0.225	
Distance between the center of any two adjacent balls (vertical side)	PitchD	0.424	
Distance between the center of any two adjacent balls (horizontal side)	PitchE	0.424	
P = Plain Grid, S = Staggered Grid	GridType	S	
F = Full Matrix, P = Perimeter, SD = Selectively Depopulated, TE = Thermally Enhanced	MatrixType	SD	
Number of balls (vertical side)	Rows	13	
Number of balls (horizontal side)	Columns	11	
Maximum number of ball positions (Rows x Columns)	Nmax	143	
Number of actual balls present	PinCount	70	
Ball positions removed from matrix. Example: C5-H10,B6-B9,A1	DepopulateBalls	C7, E1	
Ball positions added back into depopulated matrix. Example: C8,D6-F9	RepopulateBalls	-	
Minimum Standoff Height	A1min	0.15	
Maximum Height	Amax	0.525	

Recommended Land Pattern (NSMD Design)			
Description	**Dimension**	**Value (mm)**	**Reference Diagram[1]**
Diameter of pad. If specified this overrides the calculated value. This can be used to specify a manufacturer's recommended pad size.	X	0.18	
Solder Mask Expansion.	S	0.28	

1. The diagrams show table attribute referencing. For the exact diagrams, see the package outline drawing.