IPS2200

Contents

Figures

Tables

1. Introduction

The IPS2200 inductive position sensor ICs is used for high-speed absolute position sensing in automotive, industrial, medical, and consumer applications. The default configuration can be adapted for specific applications via programming.

The Non-Volatile Memory (NVM) of IPS2200 can be accessed through two independent interfaces: I2C or SPI. These two interfaces can be used only to program the IPS2200 settings or read the diagnostic flags.

The target position can be read through the sine and cosine output interface, which can be single ended or differential.

With a specific IC setting, it is possible to turn the sine and cosine signals into a quadrature signals providing 4 positions per electrical rotation (for more details, see the IPS2200 Datasheet).

This document provides instructions for programming the IPS2200 via the I2C and SPI interfaces and utilizing the diagnostic features.

The programming of the IPS2200 is needed for the Transmitter settings, the complete signal path, the output configuration, the diagnostics and the communication protocol to ensure excellent sensing performance.

Recommendation: Before using this document, read the *IPS2200 Datasheet* for product details, such as features, pin descriptions, functionality, and circuit descriptions and the IPS2200 Eval Kit User Manual before reading this Manual.

Default configuration:

- The default interface is I2C with pin 1 (CSn_IRQN) configured as address input.
- The default I2C slave address is 24dec (18hex) if address in is ‘high’, and 16dec (10 hex) if the address pin is ‘low’.

Note that for the I2C interface, the most significant two bits of the memory address are always ‘1’ (D7,D6) and register address bits are D5…D0. For details see section 4.4.2.2.

2. IPS2200 Block Diagram

The block diagram below shows the main blocks associated with the registers described in this document.

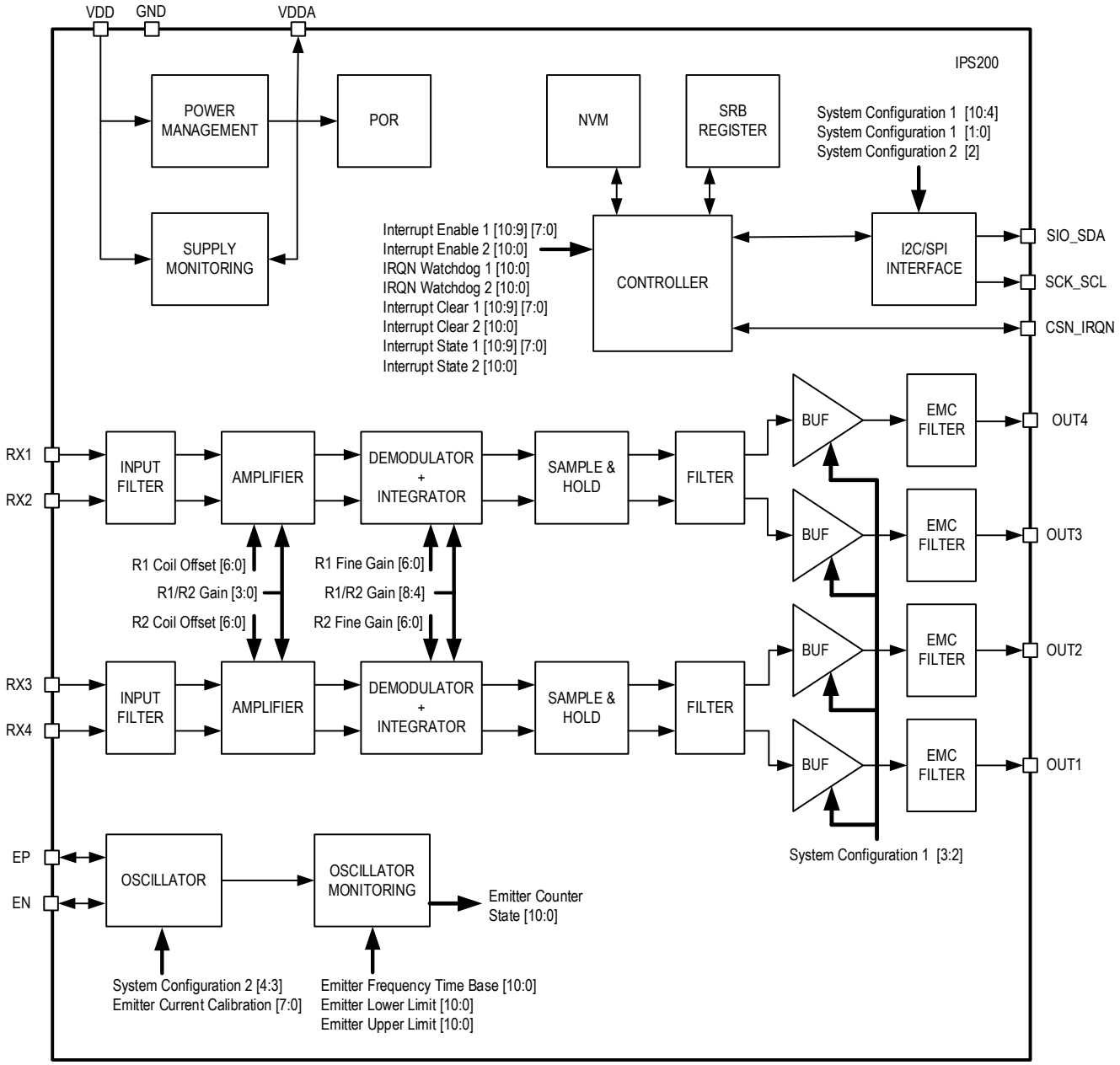


Figure 1. Block Diagram

3. Memory Architecture

The memory architecture of the IPS2200 consists of three different storage blocks with a total address range of 6 bits as describe below:

- Nonvolatile Memory (NVM) 00_{HEX} through $1F_{HEX}$
- Shadow Register Bank (SRB) 20_{HEX} through 33_{HEX}
- Special Function Register (SFR) 34_{HEX} through $3F_{HEX}$

All storage blocks are accessible by read/write access via the I2C and SPI interface.

Note that some registers can be modified by internal processes.

After power-on-reset (POR); the NVM contents of the address range 00_{HEX} to 12_{HEX} are copied to the SRB address range 20_{HEX} to 32_{HEX} within the startup time window of 3ms.

Note: after writing to an FTP register, a programming time of minimum 3ms is required before a new command can be sent.

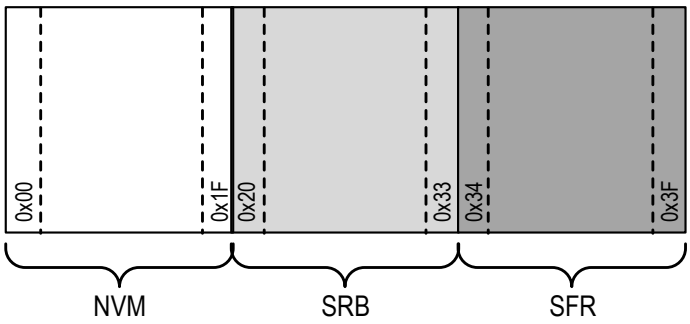


Figure 2. Memory Architecture

3.1 Nonvolatile Memory (NVM)

The nonvolatile memory has a size of 32 words × 16-bits. For enhanced reliability, the stored information can be checked for bit-errors. This has been implemented by adding a redundant 5-bit parity word concatenated with the stored information. The parity word can be considered as an error-correction code (ECC, even-parity Hamming code).

If a bit error in the NVM occurs in the addressed word, an automatic single-error correction is performed providing correct data to the outputs and flagged as a diagnostics SED (single-error detection) warning. If two bits in the addressed word have errors, a warning flag DED (double-error detection) is asserted.

The NVM memory organization is as follows:

$$N = n + p$$

Where:

N= code-word length (16 bit)

n = information length (11 bit)

p = parity length (5 bit)

The maximum parameter length that can be stored in one NVM word is 11 bits. The parity bit extension, single-bit error correction, and double-bit error check are done automatically with each write/read access to the corresponding address range.

Only the 11-bit word (Data 10 to Data 0; see Table 1) can be read/written by the I2C/SPI interface with the exception of register addresses $0F_{HEX}$ to $1F_{HEX}$, which are read only.

3.2 Shadow Register Bank

Some parameters of the NVM (see Table 3) are copied into the corresponding Shadow Register Bank (SRB) address range after a POR within the startup time window of 3ms. All these parameters are protected by error correction and detection mechanisms, which calculate a 5-bit error code (see section 3.1). If these parameters are modified by the external communication interface (I2C, SPI), the error-code is evaluated automatically and concatenated. Single-bit errors are corrected and flagged, double-bit errors are flagged.

Only the 11-bit word (Data 10 to Data 0) can be read/written by the I2C/SPI interface with the exception of read-only register addresses 20_{HEX}, 21_{HEX}, $2F_{HEX}$, 30_{HEX}, and 31_{HEX}.

Table 1. NVM to SRB Address Mapping

NVM Address	SRB Address	Description	Bit Position		
			Bits 15 to 12	Bit 11	Bits 10 to 0
00_{HEX}	20_{HEX}	System configuration 1	ECC	P_{DED}	Data 10 to Data 0
01_{HEX}	21_{HEX}	System configuration 2	ECC	P_{DED}	Data 10 to Data 0
02_{HEX}	22_{HEX}	R1/R2 gain	ECC	P_{DED}	Data 10 to Data 0
03_{HEX}	23_{HEX}	System configuration 3	ECC	P_{DED}	Data 10 to Data 0
04_{HEX}	24_{HEX}	R2 coil offset	ECC	P_{DED}	Data 10 to Data 0
05_{HEX}	25_{HEX}	internal use	ECC	P_{DED}	Data 10 to Data 0
06_{HEX}	26_{HEX}	R1 coil offset	ECC	P_{DED}	Data 10 to Data 0
07_{HEX}	27_{HEX}	Transmitter calibration	ECC	P_{DED}	Data 10 to Data 0
08_{HEX}	28_{HEX}	Transmitter frequency timebase	ECC	P_{DED}	Data 10 to Data 0
09_{HEX}	29_{HEX}	Transmitter frequency lower limit	ECC	P_{DED}	Data 10 to Data 0
$0A_{HEX}$	$2A_{HEX}$	Transmitter frequency upper limit	ECC	P_{DED}	Data 10 to Data 0
$0B_{HEX}$	$2B_{HEX}$	Iren 1	ECC	P_{DED}	Data 10 to Data 0
$0C_{HEX}$	$2C_{HEX}$	Iren 2	ECC	P_{DED}	Data 10 to Data 0
$0D_{HEX}$	$2D_{HEX}$	IRQN watchdog 1	ECC	P_{DED}	Data 10 to Data 0
$0E_{HEX}$	$2E_{HEX}$	IRQN watchdog 2	ECC	P_{DED}	Data 10 to Data 0
$0F_{HEX}$	$2F_{HEX}$	Renesas internal configuration registers	ECC	P_{DED}	Data 10 to Data 0
10_{HEX}	30_{HEX}	Renesas internal configuration registers	ECC	P_{DED}	Data 10 to Data 0
11_{HEX}	31_{HEX}	Renesas internal configuration registers	ECC	P_{DED}	Data 10 to Data 0
12_{HEX}	32_{HEX}	R1 fine gain	ECC	P_{DED}	Data 10 to Data 0
13_{HEX}	33_{HEX}	R2 fine gain	ECC	P_{DED}	Data 10 to Data 0
14_{HEX}		Not used	Read as ‘x		
15_{HEX}		Not used	Read as ‘x		
16_{HEX}		Not used	Read as ‘x		
17_{HEX}		Not used	Read as ‘x		
18_{HEX}		Not used	Read as ‘x		
19_{HEX}	n.a.	Product identifier	Renesas internal		
$1A_{HEX}$	n.a.	Product identifier	Renesas internal		
$1B_{HEX}$	n.a.	Product identifier	Renesas internal		
$1C_{HEX}$	n.a.	Product identifier	Renesas internal		
$1D_{HEX}$	n.a.	Product identifier	Renesas internal		
$1E_{HEX}$	n.a.	Product identifier	Renesas internal		
$1F_{HEX}$	n.a.	Product identifier	Renesas internal		

3.3 Special Function Register

Special Function Register (SFR) has all the diagnostic registers. It can be used in real-time and has two main functions:

- If there is an interrupt state, a diagnostic flag (dependent on the failure event) is asserted.
- If the failure event disappears, SFR updates the register and clears the flag only when the NMV is set to:
  Addr: 0x03 bit [10] - intr_volatile_mode
  Depending on this bit setting the flags in the interrupt registers either clear automatically after some debounce timeout or have to be cleared externally by writing 1 to irclr (interrupt clear) register.

The NVM is not copied to the SFR during the power up of IC. The flags, write and read can be effected dynamically without interruption of the sensor functionality. Volatile parameters (for example, transmitter counter state) are not protected by an error-code and are stored in the SFR. The contents of these registers can be modified during operation by internal processes and by the digital interface (I2C, SPI) read/write access.

The product identifier version code (address 3EHEX) is hardcoded and does not change.

Table 2. SFR Address Mapping

Address	Description	Bit Position	
		Bits 15 to 11	Bits 10 to 0
34_{HEX}	Interrupt clear 1	Not used	Data 10 to Data 0
35_{HEX}	Interrupt clear 2	Not used	Data 10 to Data 0
36_{HEX}	Interrupt state 1	Not used	Data 10 to Data 0
37_{HEX}	Interrupt state 2	Not used	Data 10 to Data 0
38_{HEX}	Transmitter counter state	Not used	Data 10 to Data 0
$3A_{HEX}$	NVM ECC state	Not used	Data 10 to Data 0
[a] SFR addresses 39, 3B, 3C, 3D, 3E, and 3F are Renesas internal.			

3.4 Main Registers Overview

Table 3. Register Overview

NVM Address	SRB Address	Description	Main Functions
00_{HEX}	20_{HEX}	System configuration 1	I2C and SPI mode settings. The default interface is I2C with pin 1 (CSn_IRQN) configured as address input.
01_{HEX}	21_{HEX}	System configuration 2	• Output Interrupt – mode and signal type • Quadrature mode selection • Security of reading/writing via I2C/SPI • Transmitter settings • Supply voltage range selection
02_{HEX}	22_{HEX}	R1/R2 gain	Sets the gain stage and the number of integration cycles of the integrator.
03_{HEX}	23_{HEX}	System configuration 3	Used for dynamic interrupt signal by 1-bit setting.
04_{HEX}	24_{HEX}	R2 coil offset	Sets the offset value for R2 coil and its sign.
06_{HEX}	26_{HEX}	R1 coil offset	Sets the offset value for R1 coil and its sign.
07_{HEX}	27_{HEX}	Transmitter current calibration	Used for LC oscillator current settings.
08_{HEX}	28_{HEX}	Transmitter frequency timebase	Used for counter settings.
09_{HEX}	29_{HEX}	Transmitter frequency lower limit	Used for lower limit of counter.
$0A_{HEX}$	$2A_{HEX}$	Transmitter frequency upper limit	Used for upper limit of counter.
$0B_{HEX}$	$2B_{HEX}$	Interrupt1 Enables	Used for diagnostics of: • VDD upper and lower voltage • SPI and I2C fails • SRB error detection • NVM error detection • LC oscillator detection range • Oscillator stuck check
$0C_{HEX}$	$2C_{HEX}$	Interrupt2 Enables	Used for diagnostics of: • VDDA under-voltage • Low amplitude • Bus transfer • Mechanical failure • Sin/Cosine voltage levels • Coils short
$0D_{HEX}$	$2D_{HEX}$	IRQN watchdog 1	

NVM Address	SRB Address	Description	Main Functions
$0E_{HEX}$	$2E_{HEX}$	IRQN watchdog 2	
12_{HEX}	32_{HEX}	R1 fine gain	Sets the amplitude of the R1 channel for correcting the mismatch between the two receivers.
13_{HEX}	33_{HEX}	R2 fine gain	Sets the amplitude of the R2 channel for correcting the mismatch between the two receivers.
19_{HEX}	n.a.	Product identifier	
$1A_{HEX}$	n.a.	Product identifier	
$1B_{HEX}$	n.a.	Product identifier	
$1C_{HEX}$	n.a.	Product identifier	
$1D_{HEX}$	n.a.	Product identifier	
$1E_{HEX}$	n.a.	Product identifier	
$1F_{HEX}$	n.a.	Product identifier	

3.5 Register Descriptions

3.5.1. System Configuration 1

This register configures the interface protocol, the SPI or I2C selection, and the output mode (Analog or Digital Quadrature mode, see *IPS2200 Datasheet* for detailed explanation on output modes).

Table 4. System Configuration 1 Register Details

Block	NVM	SRB	SFR
Address	00_{HEX}	20_{HEX}	–
Default	0323_{HEX}	0323_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

.

Table 5. System Configuration 1 Register Bit Descriptions

Bit	Symbol	Default	Type	Description
sys_conf1.15 sys_conf1.14 sys_conf1.13 sys_conf1.12	ECC	0000_{BIN}	R	Parity word for error correction. For the internal use of the IC.
sys_conf1.11	P_{DED}	0_{BIN}		Parity bit for double-bit error detection.
sys_conf1.10	SPI data order	0_{BIN}	R/W	SPI data order (DORD): 0_{BIN} = MSB (Most Significant Bit?) received first; LSB (Least Significant Bit?) received last 1_{BIN} = LSB received first; MSB received last
sys_conf1.9 sys_conf1.8	SPI modes	11_{BIN}	R/W	Clock polarity (CPOL) and clock phase (CPHA) of SPI interface. sys_conf1.9 = CPHA sys_conf1.8 = CPOL CPHA: 0_{BIN} = first data bit written at CSN falling edge CPHA and CPOL: 00_{BIN} = sample edge rising, drive edge falling 01_{BIN} = sample edge falling, drive edge rising 10_{BIN} = sample edge falling, drive edge rising 11_{BIN} = sample edge rising, drive edge falling
sys_conf1.7 sys_conf1.6 sys_conf1.5 sys_conf1.4	I2C MSN slave address	0010_{BIN}	R/W	I2C slave programmable address bits, see Table 6 for details.[a]

Bit	Symbol	Default	Type	Description
sys_conf1.3 sys_conf1.2	Outputs mode protocol	00_{BIN}	R/W	Outputs have two protocol modes: • Analog: ○ 00_{BIN}: SIN, SINN, COS; COSN ○ 01_{BIN}: SIN, REF, COS, REF • Digital Quadrature Mode ○ 10_{BIN}: A, AN, B, BN ○ 11_{BIN}: A, B
sys_conf1.1 sys_conf1.0	System protocol	11_{BIN}	R/W	Protocol mode of communication interface between sensor and processing unit: 00_{BIN} = SPI half duplex 01_{BIN} = SPI half duplex and interrupt 10_{BIN} = I2C and interrupt 11_{BIN} = I2C and addressing pin

[a] If modifying the I2C MSN address field, there are two restricted address groups: 0000_{BIN} and 1111_{BIN} (refer to section 3.5.1).

Table 6. I2C Slave Address Configuration

Configuration [a]	A6	A5	A4	A3	A2	A1	A0
Protocol mode = 10_{BIN}	sys_conf1.7 to sys_conf1.4				0	0	0
Protocol mode = 11_{BIN}	sys_conf1.7 to sys_conf1.5			Pin 1 state [b]	0	0	0

[a] Protocol mode = sys_conf1.1 and sys_conf1.0, see Table 5.

[b] Pin 1 state = logical state of Pin 1 input pin (CSN_IRQN).

3.5.2. System Configuration 2

The main functions of this register is to define the following settings:

- Output interrupt enable – interrupt by using the output SIN/COSINE
- Quadrature mode selection
- Security of reading/writing via I2C or SPI
- Transmitter settings
- SPI/I2C Protocol integrity check
- Supply voltage range selection

Table 7. System Configuration 2 Register Details

Block	NVM	SRB	SFR
Address	01_{HEX}	21_{HEX}	–
Default Value	0101_{HEX}	0101_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 8. System Configuration 2 Register Bit Descriptions

Bit	Symbol	Default	Type	Description
sys_conf2.15 sys_conf2.14 sys_conf2.13 sys_conf2.12	ECC	0000_{BIN}	R	Parity word for error correction. For the internal use of the IC.
sys_conf2.11	P_{DED}	0_{BIN}		Parity bit for double bit error detection. For the internal use of the IC.
sys_conf2.10	Pulse mode configuration	0_{BIN}	R/W	Function of Pulse Output. For this function the Output Mode must be set to Quadrature Differential (set sys_conf1.3, sys_conf1.2 to 10BIN: A, AN, B, BN) and Pulse Mode (sys_conf2.5) must be enabled (sys_conf2.5 = 1_{BIN}). • 0_{BIN} = 1 pulse per phase (signal A) • 1_{BIN} = 2 pulses per phase (signal A XOR B)
sys_conf2.9 sys_conf2.8	Renesas internal configuration	01_{BIN}	R/W	Renesas internal feature. Do not change the default setting.

Bit	Symbol	Default	Type	Description
sys_conf2.7	Output interrupt enable	0_{BIN}	R/W	Enable Interrupt signaling at output interface: • 0_{BIN} = interrupt signaling disabled • 1_{BIN} = interrupt signaling enabled If the interface is not used, the output is also intended for the interrupts. If the interrupt is enabled and a failure occurs, the output has the following values: • SIN/A = 0_{DEC}/0bit • SINN/AN =0_{DEC}/0bit • COS/B =1_{DEC}/1bit • COSN/BN =1_{DEC}/1bit
sys_conf2.6	Cyber security	0_{BIN}	R/W	The cyber security bit determines the transfer direction of the allowed I2C/SPI access mode. 0_{BIN} = I2C/SPI read/write access 1_{BIN} = I2C/SPI read-only access
sys_conf2.5	Quadrature Mode protocol	0_{BIN}	R/W	Select between AB and Pulse mode. For this function the Output Mode must be set to Quadrature Differential (set sys_conf1.3, sys_conf1.2 to 10BIN: A, AN, B, BN) 0_{BIN} = AB mode: A, AN, B, BN signals at output pins 1_{BIN} = Pulse mode; Pulse output at SINN pin 13
sys_conf2.4	Transmitter cp enable	0_{BIN}	R/W	Transmitter charge pump enable. 0_{BIN} = off 1_{BIN} = on It must be on, if VDD = 3.3V.
sys_conf2.3	Transmitter amplitude control	0_{BIN}	R/W	Transmitter amplitude control 0_{BIN} = off 1_{BIN} = on
sys_conf2.2	Protocol integrity check	0_{BIN}	R/W	I2C/SPI interface support integrity check of the serial data stream (Rx and Tx). 0_{BIN} = integrity check off 1_{BIN} = integrity check on
sys_conf2.1	Renesas internal configuration	0_{BIN}	R/W	Renesas internal feature. Do not change the default setting.
sys_conf2.0	Supply voltage	1_{BIN}	R/W	Supply voltage mode for VDD regulator: • 0_{BIN} = 3.3V typical VDD supply, the result on VDDA is 4V The 'Transmitter cp enable' must be set to 1_{BIN} to use this feature. • 1_{BIN} = 5.0V typical VDD supply, the result on VDDA is 4V

3.5.3. Receiver 1/2 Gain

This register is used for setting the gain stage and the number of integration cycles of the integrator.

The gain stage and the integration cycles are used for increasing or decreasing the amplitude of the input signal.

Table 9. Receiver 1/2 Gain Register Details

Block	NVM	SRB	SFR
Address	02_{HEX}	22_{HEX}	–
Default value	0056_{HEX}	0056_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 10. Receiver 1/2 Gain

Bit	Symbol	Default	Type	Description
r12_gain.15 r12_gain.14 r12_gain.13 r12_gain.12	ECC	0000_{BIN}		Parity word for error correction. For the internal use of the IC.
r12_gain.11	P_{DED}	0_{BIN}		Parity bit for double bit error detection.
r12_gain.10 r12_gain.9	Renesas internal configuration	00_{BIN}	R/W	Renesas internal feature. Do not change the default setting.

<table>
<tr><th>Bit</th><th>Symbol</th><th>Default</th><th>Type</th><th>Description</th></tr>
<tr><td>r12_gain.8
r12_gain.7
r12_gain.6
r12_gain.5
r12_gain.4</td><td>Integration cycles</td><td>00101_{BIN}</td><td>R/W</td><td>The integration cycles parameter defines the integration time of the integrator in excitation clock units.
The minimum setting is 5 cycles.
<table>
<tr><th>Setting</th><th>Integration Time</th></tr>
<tr><td>00000_{BIN} to 00101_{BIN}</td><td>5 period of LC oscillator</td></tr>
<tr><td>00110_{BIN}</td><td>6 period of LC oscillator</td></tr>
<tr><td>00111_{BIN}</td><td>7 period of LC oscillator</td></tr>
<tr><td>….</td><td>….</td></tr>
<tr><td>11111_{BIN}</td><td>31 period of LC oscillator</td></tr>
</table></td></tr>
<tr><td>r12_gain.3
r12_gain.2
r12_gain.1
r12_gain.0</td><td>gain stage</td><td>0110_{BIN}</td><td>R/W</td><td>The gain stage parameter defines the input resistor of the integrator by this formula:
$Gain = 2^{\frac{n}{2}}$ $where$ $n = dec(\text{gain stage})$
A total 14 stages in multiplication steps of sqrt(2) are implemented.
E_{HEX} and F_{HEX} are unused.
For example:
• 0_{HEX}, n = 0 , Gain = 1
• 1_{HEX}, n = 1, Gain = $2^{\frac{1}{2}}$
• 2_{HEX}, n= 2, Gain = $2^{\frac{2}{2}}$
until
D_{HEX} , n=13, Gain = $2^{\frac{13}{2}}$ = 64 x $2^{\frac{1}{2}}$</td></tr>
</table>

3.5.4. System Configuration 3

Only used for dynamic interrupt signaling by the sys_conf3.10 bit. All other bits are for Renesas internal usage.

Table 11. System Configuration 3 Register Details

Block	NVM	SRB	SFR
Address	03_{HEX}	23_{HEX}	–
Default value		0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 12. System Configuration 3 Register Bit Definitions

Bit	Symbol	Default	Type	Description
sys_conf3.15 sys_conf3.14 sys_conf3.13 sys_conf3.12	ECC			Parity word for error correction. For the internal use of the IC.
sys_conf3.11	P_{DED}			Parity bit for double bit error detection.
sys_conf3.10	dynamic_intr_enable	0_{BIN}	R/W	Enable volatile interrupt signaling: • 0_{BIN} = Disable; IRQN cleared by microcontroller host • 1_{BIN} = Enable; IRQN event type
sys_conf3.9	Renesas internal configuration	0_{BIN}	R/W	Renesas internal feature. Do not change the default setting.
sys_conf3.8	Renesas internal configuration	0_{BIN}	R/W	Renesas internal feature. Do not change the default setting.
sys_conf3.7	–	–	–	Not used; read as 0.

Bit	Symbol	Default	Type	Description
sys_conf3.6 sys_conf3.5 sys_conf3.4 sys_conf3.3 sys_conf3.2 sys_conf3 1 sys_conf3.0	Renesas internal configuration	$000\ 0000_{BIN}$	R/W	Renesas internal feature. Do not change the default setting.

3.5.5. R2 Coil Offset

The coil offset is defined as the constant volt value added or subtracted to the input signal. This offset is generated from the non-ideality of the coil design and non-uniformity of the electromagnetic field. This register enables to set the offset value and its sign (positive or negative).

The offset can be cancelled by adding or subtracting a constant value on the input signal.

For example:

During a complete sweep of the target, the offset on the receiver is seen as a shift of the signal with respect to the x-axis, see Figure 3 where the sine with peak amplitude 1 is shifted by +0.5.

The compensation in this example should be

Offset compensation = - 0.5

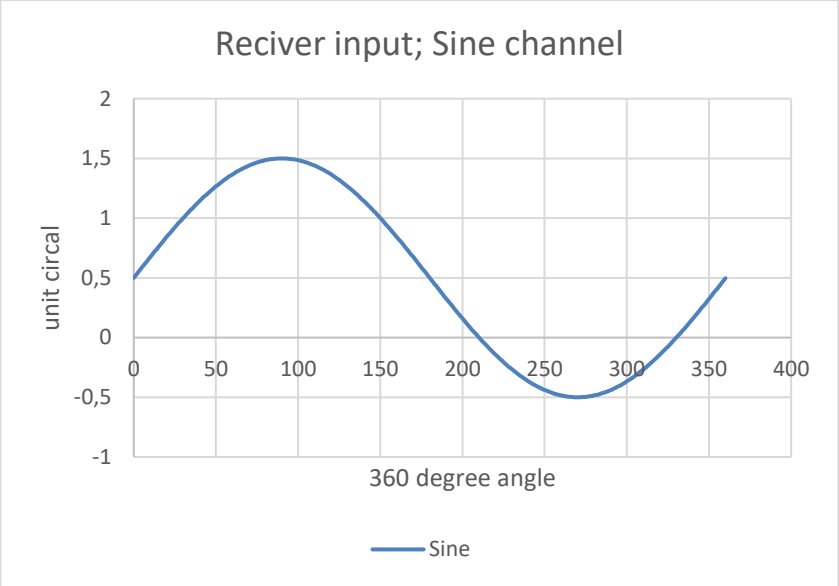


Figure 3. Receiver Input Signal with 0.5 Offset

Table 13. R2 Coil Offset Register Details

Block	NVM	SRB	SFR
Address	04_{HEX}	24_{HEX}	–
Default value	–	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 14. R2 Coil Offset Register Bit Definitions

Bit	Symbol	Default	Type	Description
r2_coil_offset.15 r2_coil_offset.14 r2_coil_offset.13 r2_coil_offset.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
r2_coil_offset.11	P_{DED}	–	–	Parity bit for double-bit error detection.

Bit	Symbol	Default	Type	Description
r2_coil_offset.10	–	–	–	Not used; read as 0.
r2_coil_offset.9	–	–	–	Not used; read as 0.
r2_coil_offset.8	–	–	–	Not used; read as 0.
r2_coil_offset.7	–	–	–	Not used; read as 0.
r2_coil_offset.6	R2 offset polarity	0_{BIN}	R/W	Define the applied offset polarity: • 0_{BIN} = Addition • 1_{BIN} = Subtraction
r2_coil_offset.5 r2_coil_offset.4 r2_coil_offset.3 r2_coil_offset.2 r2_coil_offset 1 r2_coil_offset.0	R2 offset value	00 0000_{BIN}	R/W	The R2 offset compensation value, relative to the Transmitter LC oscillator voltage amplitude is: • 0x00 - no correction • 0x01 - 0.003% • 0x02 - 0.006% • 0x03 - 0.009% until • 0x3E - 0.197% • 0x3F - 0.200% Overall range is 0.2%, step is 0.003%

3.5.6. R1 Coil Offset

The Coil offset is defined as the constant volt value added or subtracted to the input signal. This offset is generated from the non-ideality of the coil design and non-uniformity of the electromagnetic field. This register enables to set the offset value and its sign (positive or negative).

The offset can be cancelled by adding or subtracting a constant value on the input signal.

Table 15. R1 Coil Offset Register Details

Block	NVM	SRB	SFR
Address	06_{HEX}	26_{HEX}	–
Default value	–	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 16. R1 Coil Offset Register Bit Definitions

Bit	Symbol	Default	Type	Description
r1_coil_offset.15 r1_coil_offset.14 r1_coil_offset.13 r1_coil_offset.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
r1_coil_offset.11	P_{DED}	–	–	Parity bit for double bit error detection.
r1_coil_offset.10	–	–	–	Not used; read as 0.
r1_coil_offset.9	–	–	–	Not used; read as 0.
r1_coil_offset.8	–	–	–	Not used; read as 0.
r1_coil_offset.7	–	–	–	Not used; read as 0.
r1_coil_offset.6	R1 offset polarity	0_{BIN}	R/W	Define the applied offset polarity: • 0_{BIN} = Addition • 1_{BIN} = Subtraction
r1_coil_offset.5 r1_coil_offset.4 r1_coil_offset.3 r1_coil_offset.2 r1_coil_offset 1 r1_coil_offset.0	R1 offset value	00 0000_{BIN}	R/W	The R1 offset compensation value, relative to the Transmitter LC oscillator voltage amplitude is: • 0x00 - no correction • 0x01 - 0.003% • 0x02 - 0.006% • 0x03 - 0.009% until • 0x3E - 0.197% • 0x3F - 0.200%" Overall range is 0.2%, step is 0.003%

3.5.7. Transmitter Current Calibration

The Transmitter current bias (I_{BIAS}) influences the AC current of the LC oscillator. Increasing the AC current, increases proportionally the magnitude of the transmitter signal. The I_{BIAS} optimum setting can be calculated according to the following formula:

$I_{BIAS} = VDD / (35 \times L \times Q \times F)$

Where:

VDD = Supply voltage in Volt

L = Inductance of transmit coil in Henry

F = Transmit oscillator frequency in Hz

Q = Quality factor of the Transmitter coil, it is calculated according to the following formula:

$$Q = \frac{1}{R}\sqrt{\frac{L}{C}}$$

Where:
R = Resistance of the Transmitter Coil
C = Capacitance of the transmitter resonator

For Example:

If L = 6µH, R = 1Ohm, C = 345pF (such that the transmitter oscillates at F = 3.5MHz) and VDD = 5V

According to the formula above:

Q = 132

I_{BIAS} = 51.5µA

As shown in Table 19, the closest I_{BIAS} setting to 51.5µA is 52µA corresponding to the following setting:

LC Current Multiplication = 01BIN

LC Current Base = 001101BIN

Table 17. Transmitter Current Calibration Register Details

Block	NVM	SRB	SFR
Address	07_{HEX}	27_{HEX}	–
Default value	$00BE_{HEX}$	$00BE_{HEX}$	–
Access	READ/WRITE	READ/WRITE	–

Table 18. Transmitter Current Calibration Register Bit Definitions

Bit	Symbol	Default	Type	Description
transmitter_calib.15 transmitter_calib.14 transmitter_calib.13 transmitter_calib.12	ECC	0000_{BIN}	–	Parity word for error correction. For the internal use of the IC.
transmitter_calib.11	P_{DED}	0_{BIN}	–	Parity bit for double-bit error detection.
transmitter_calib.10	–	0_{BIN}	–	Not used; read as 0.
transmitter_calib.9	–	0_{BIN}	–	Not used; read as 0.
transmitter_calib.8	–	0_{BIN}	–	Not used; read as 0.
transmitter_calib.7 transmitter_calib.6	LC current multiplication	10_{HEX}	R/W	See Table 19.

Bit	Symbol	Default	Type	Description
transmitter_calib.5 transmitter_calib.4 transmitter_calib.3 transmitter_calib.2 transmitter_calib.1 transmitter_calib.0	LC current base	111110_{BIN}	R/W	See Table 19.

Table 19. Transmitter Bias Current Settings

0	LC Current Multiplication = 00	LC Current Multiplication = 01	LC Current Multiplication = 10	LC Current Multiplication = 11
LC Current Base = 000000BIN	0µA	0µA	0µA	0µA
LC Current Base = 000001BIN	0.5µA	2µA	8µA	32µA
LC Current Base = 000010BIN	1µA	4µA	16µA	64µA
LC Current Base = 000011BIN	1.5µA	6µA	24µA	96µA
LC Current Base = 000100BIN	2µA	8µA	32µA	128µA
LC Current Base = 000101BIN	2.5µA	10µA	40µA	160µA
LC Current Base = 000110BIN	3µA	12µA	48µA	192µA
LC Current Base = 000111BIN	3.5µA	14µA	56µA	224µA
LC Current Base = 001000BIN	4µA	16µA	64µA	256µA
LC Current Base = 001001BIN	4.5µA	18µA	72µA	288µA
LC Current Base = 001010BIN	5µA	20µA	80µA	320µA
LC Current Base = 001011BIN	5.5µA	22µA	88µA	352µA
LC Current Base = 001100BIN	6µA	24µA	96µA	384µA
LC Current Base = 001101BIN	6.5µA	26µA	104µA	416µA
LC Current Base = 001110BIN	7µA	28µA	112µA	448µA
LC Current Base = 001111BIN	7.5µA	30µA	120µA	480µA
LC Current Base = 010000BIN	8µA	32µA	128µA	512µA
LC Current Base = 010001BIN	8.5µA	34µA	136µA	544µA
LC Current Base = 010010BIN	9µA	36µA	144µA	576µA
LC Current Base = 010011BIN	9.5µA	38µA	152µA	608µA
LC Current Base = 010100BIN	10µA	40µA	160µA	640µA
LC Current Base = 010101BIN	10.5µA	42µA	168µA	672µA
LC Current Base = 010110BIN	11µA	44µA	176µA	704µA
LC Current Base = 010111BIN	11.5µA	46µA	184µA	736µA
LC Current Base = 011000BIN	12µA	48µA	192µA	768µA
LC Current Base = 011001BIN	12.5µA	50µA	200µA	800µA
LC Current Base = 011010BIN	13µA	52µA	208µA	832µA
LC Current Base = 011011BIN	13.5µA	54µA	216µA	864µA
LC Current Base = 011100BIN	14µA	56µA	224µA	896µA
LC Current Base = 011101BIN	14.5µA	58µA	232µA	928µA
LC Current Base = 011110BIN	15µA	60µA	240µA	960µA
LC Current Base = 011111BIN	15.5µA	62µA	248µA	992µA
LC Current Base = 100000BIN	16µA	64µA	256µA	1024µA
LC Current Base = 100001BIN	16.5µA	66µA	264µA	1056µA
LC Current Base = 100010BIN	17µA	68µA	272µA	1088µA
LC Current Base = 100011BIN	17.5µA	70µA	280µA	1120µA

LC Current Base = 100100BIN	18µA	72µA	288µA	1152µA
LC Current Base = 100101BIN	18.5µA	74µA	296µA	1184µA
LC Current Base = 100110BIN	19µA	76µA	304µA	1216µA
LC Current Base = 100111BIN	19.5µA	78µA	312µA	1248µA
LC Current Base = 101000BIN	20µA	80µA	320µA	1280µA
LC Current Base = 101001BIN	20.5µA	82µA	328µA	1312µA
LC Current Base = 101010BIN	21µA	84µA	336µA	1344µA
LC Current Base = 101011BIN	21.5µA	86µA	344µA	1376µA
LC Current Base = 101100BIN	22µA	88µA	352µA	1408µA
LC Current Base = 101101BIN	22.5µA	90µA	360µA	1440µA
LC Current Base = 101110BIN	23µA	92µA	368µA	1472µA
LC Current Base = 101111BIN	23.5µA	94µA	376µA	1504µA
LC Current Base = 110000BIN	24µA	96µA	384µA	1536µA
LC Current Base = 110001BIN	24.5µA	98µA	392µA	1568µA
LC Current Base = 110010BIN	25µA	100µA	400µA	1600µA
LC Current Base = 110011BIN	25.5µA	102µA	408µA	1632µA
LC Current Base = 110100BIN	26µA	104µA	416µA	1664µA
LC Current Base = 110101BIN	26.5µA	106µA	424µA	1696µA
LC Current Base = 110110BIN	27µA	108µA	432µA	1728µA
LC Current Base = 110111BIN	27.5µA	110µA	440µA	1760µA
LC Current Base = 111000BIN	28µA	112µA	448µA	1792µA
LC Current Base = 111001BIN	28.5µA	114µA	456µA	1824µA
LC Current Base = 111010BIN	29µA	116µA	464µA	1856µA
LC Current Base = 111011BIN	29.5µA	118µA	472µA	1888µA
LC Current Base = 111100BIN	30µA	120µA	480µA	1920µA
LC Current Base = 111101BIN	30.5µA	122µA	488µA	1952µA
LC Current Base = 111110BIN	31µA	124µA	496µA	1984µA
LC Current Base = 111111BIN	31.5µA	126µA	504µA	2016µA

3.5.8. Transmitter Frequency Time Base

This register contains a number for the LC oscillator frequency. The measurement time for checking the transmitter frequency is as follows:

t_M = TimeBaseCntr × t_{osc_dig}

Where

t_{osc_dig} = Digital clock period

The internal oscillator is trimmed to 7MHz (142.86ns) and divided by 2, therefore t_{osc_dig} = 285.72ns.

Table 20. Transmitter Frequency Time Base Register Details

Block	NVM	SRB	SFR
Address	08_{HEX}	28_{HEX}	–
Default value	0000_{HEX}	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 21. Transmitter Frequency Time Base Register Bit Descriptions

Bit	Symbol	Default	Type	Description
transmitter_wdg.15 transmitter_wdg.14 transmitter_wdg.13 transmitter_wdg.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
transmitter_wdg.11	P_{DED}	–	–	Parity bit for double bit error detection.
transmitter_wdg.10 transmitter_wdg.9 transmitter_wdg.8 transmitter_wdg.7 transmitter_wdg.6 transmitter_wdg.5 transmitter_wdg.4 transmitter_wdg.3 transmitter_wdg.2 transmitter_wdg.1 transmitter_wdg.0	Time base counter	$000\ 0000\ 0000_{BIN}$	R/W	Time base for excitation frequency measurement: • $000\ 0000\ 0000_{BIN}$ = Counter off • $000\ 0000\ 0001_{BIN}$ = 1 osc_dig cycle • $000\ 0000\ 0010_{BIN}$ = 2 osc_dig cycle … • $111\ 1111\ 1111_{BIN}$ = 2047 osc_dig cycles

3.5.9. Transmitter Lower Limit

A digital comparator checks the LC oscillator counter state for minimum value (see section 3.5.22). If the check fails, the diagnostic flag irstate1.1 (lc_osc_freq_fail) is asserted.

Table 22. Transmitter Lower Limit Register Details

Block	NVM	SRB	SFR
Address	09_{HEX}	29_{HEX}	–
Default value	–	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 23. Transmitter Lower Limit Register Bit Descriptions

Bit	Symbol	Default	Type	Description
transmitter_low.15 transmitter_low.14 transmitter_low.13 transmitter_low.12	ECC			Parity word for error correction. For the internal use of the IC.
transmitter_low.11	P_{DED}			Parity bit for double bit error detection.
transmitter_low.10 transmitter_low.9 transmitter_low.8 transmitter_low.7 transmitter_low.6 transmitter_low.5 transmitter_low.4 transmitter_low.3 transmitter_low.2 transmitter_low 1 transmitter_low.0	LC oscillator lower limit	$000\ 0000\ 0000_{BIN}$	R/W	Lower limit for excitation frequency counter: • $000\ 0000\ 0000_{BIN}$ = Counter off • $000\ 0000\ 0001_{BIN}$ = 1 Oscillator cycle • $000\ 0000\ 0010_{BIN}$ = 2 Oscillator cycles … • $111\ 1111\ 1111_{BIN}$ = 2047 Oscillator cycles

3.5.10. Transmitter Upper Limit

A digital comparator checks the LC oscillator counter state for the maximum value (see section 3.5.22). If the check fails, the diagnostic flag irstate1.1 (lc_osc_freq_fail) is asserted.

Table 24. Transmitter Upper Limit Register Details

Block	NVM	SRB	SFR
Address	$0A_{HEX}$	$2A_{HEX}$	–
Default value	0000_{HEX}	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 25. Transmitter Upper Limit Register Bit Descriptions

Bit	Symbol	Default	Type	Description
transmitter_up.15 transmitter_up.14 transmitter_up.13 transmitter_up.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
transmitter_up.11	P_{DED}	–	–	Parity bit for double-bit error detection.
transmitter_up.10 transmitter_up.9 transmitter_up.8 transmitter_up.7 transmitter_up.6 transmitter_up.5 transmitter_up.4 transmitter_up.3 transmitter_up.2 transmitter_up 1 transmitter_up.0	LC oscillator upper limit	$111\ 1111\ 1111_{BIN}$	R/W	Upper limit for excitation frequency counter: • $000\ 0000\ 0000_{BIN}$ = Counter off • $000\ 0000\ 0001_{BIN}$ = 1 oscillator cycle • $000\ 0000\ 0010_{BIN}$ = 2 oscillator cycles … • $111\ 1111\ 1111_{BIN}$ = 2047 oscillator cycles

3.5.11. Interrupt Enable 1

The interrupt function of IPS2200 is divided into static or dynamic events. The static event is considered when the diagnostics is latched until POR, while the dynamic event is active only during the presence of the diagnostics.
In the present of the dynamic event, choose either of the following options:

- Use a microcontroller to clear the register. The benefit of this method is to understand the process causing the interrupt and the type of diagnostics causing the error.
- Use self-clear form the IC by setting: Addr: 0x03 bit [10] - intr_volatile_mode.

The following six registers describe the diagnostics and the error causing the interrupt:

- Interrupt Enable 1/2: they control the Interrupt pin of the IC, see sections 3.5.11 and 3.5.12.
- Interrupt Clear 1/2: they are used by the microcontroller to clear the event and to check the type of the diagnostics, see sections 3.5.18 and 3.5.19.
- Interrupt State 1/2: they show whether there is a diagnostic event, see sections 3.5.20 and 3.5.21.

For the interrupt handling of the IPS2200, interrupt request sources can be individually enabled or disabled by the corresponding iren1.x flag.

Iren1 is used for VDD upper and lower value diagnostics, SPI and I2C fails, SRB error detection, NVN error detectionsm LC oscillator detection range check and oscillator stuck check.

Table 26. Interrupt Enable 1 Register Details

Block	NVM	SRB	SFR
Address	$0B_{HEX}$	$2B_{HEX}$	–
Default value	0000_{HEX}	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 27. Interrupt Enable 1 Register Bit Descriptions

Bit	Symbol	Default	Type	Description
iren1.15 iren1.14 iren1.13 iren1.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
iren1.11	P_{DED}	–	–	Parity bit for double-bit error detection.
iren1.10	VDD over-voltage	0_{BIN}	R/W	VDD over-voltage detection: • 0_{BIN} = Over-voltage interrupt disabled • 1_{BIN} = Over-voltage interrupt enabled

Bit	Symbol	Default	Type	Description
iren1.9	VDD under-voltage	0_{BIN}	R/W	VDD under-voltage detection: • 0_{BIN} = VDD under-voltage interrupt disabled • 1_{BIN} = VDD under-voltage interrupt enabled
iren1.8	Renesas internal configuration	0_{BIN}	R/W	Renesas internal configuration. Do not change the default setting.
iren1.7	data_access_fail	0_{BIN}	R/W	SPI/I2C data access timeout: • 0_{BIN} = Data access fail interrupt disabled • 1_{BIN} = Data access fail interrupt enabled
iren1.6	spi_I2C_protocol_fail	0_{BIN}	R/W	SPI/I2C interface serial received data fail/data access aborted (for error handling details, see Table 50): • 0_{BIN} = SPI/I2C protocol fail interrupt disabled • 1_{BIN} = SPI/I2C protocol fail interrupt enabled
iren1.5	srb_ded	0_{BIN}	R/W	SRB double-bit error detection/SFR test mode register 1/2 parity bit error detection: • 0_{BIN} = SRB double bit error interrupt disabled • 1_{BIN} = SRB double bit error interrupt enabled
iren1.4	srb_sed	0_{BIN}	R/W	SRB single-bit error detection and correction: • 0_{BIN} = SRB single bit error interrupt disabled • 1_{BIN} = SRB single bit error interrupt enabled
iren1.3	nvm_ded	0_{BIN}	R/W	NVM double bit error detection: • 0_{BIN} = NVM double bit error interrupt disabled • 1_{BIN} = NVM double bit error interrupt enabled
iren1.2	nvm_sed	0_{BIN}	R/W	NVM single bit error detection and correction: • 0_{BIN} = NVM single bit error interrupt disabled • 1_{BIN} = NVM single bit error interrupt enabled
iren1.1	lc_osc_freq_fail	0_{BIN}	R/W	LC oscillator frequency range check (see registers 07_{HEX}, 08_{HEX}, $0A_{HEX}$ for the limits): • 0_{BIN} = LC oscillator frequency error interrupt disabled • 1_{BIN} = LC oscillator frequency error interrupt enabled
iren1.0	lc_osc_stuck	0_{BIN}	R/W	LC oscillator stuck check: • 0_{BIN} = LC oscillator stuck interrupt disabled • 1_{BIN} = LC oscillator stuck interrupt enabled

3.5.12. Interrupt Enable 2

The interrupt function of IPS2200 is divided into static or dynamic events. The static event is considered when the diagnostics is latched until POR, while the dynamic event is active only during the presence of the diagnostics.
In the present of the dynamic event, choose either of the following options:

- Use a microcontroller to clear the register. The benefit of this method is to understand the process causing the interrupt and the type of diagnostics causing the error.
- Use self-clear form the IC by setting: Addr: 0x03 bit [10] - intr_volatile_mode.

The following six registers describe the diagnostics and the error causing the interrupt:

- Interrupt Enable 1/2: they control the Interrupt pin of the IC, see sections 3.5.11 and 3.5.12.
- Interrupt Clear 1/2: they are used by the microcontroller to clear the event and to check the type of the diagnostics, see sections 3.5.18 and 3.5.19.
- Interrupt State 1/2: they show whether there is a diagnostic event, see sections 3.5.20 and 3.5.21.

For the interrupt handling of the IPS2200, interrupt request sources can be individually enabled or disabled by the corresponding iren2.x flag.

Table 28. Interrupt Enable 2 Register Details

Block	NVM	SRB	SFR
Address	$0C_{HEX}$	$2C_{HEX}$	–
Default value	0000_{HEX}	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 29. Interrupt Enable 2 Register Bit Descriptions

Bit	Symbol	Default	Type	Description
iren2.15 iren2.14 iren2.13 iren2.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
iren2.11	P_{DED}	–	–	Parity bit for double-bit error detection.
iren2.10	vdda_csn_data	0_{BIN}	R/W	VDDA under-voltage or CSN/DATA pin over-voltage: • 0_{BIN} = VDDA under-voltage or CSN/DATA over-voltage interrupt disabled • 1_{BIN} = VDDA under-voltage or CSN/DATA over-voltage interrupt enabled
iren2.9	low_amplitude	0_{BIN}	R/W	Low amplitude: • 0_{BIN} = Low signal amplitude interrupt disabled • 1_{BIN} = Low signal amplitude interrupt enabled
iren2.8	apb_transfer_fail	0_{BIN}	R/W	Internal bus transfer failure. Any write access of a read-only register is rejected and flagged by the internal bus control (refer to sys_conf2.6 in Table 8). • 0_{BIN} = Internal bus transfer fail interrupt disabled • 1_{BIN} = Internal bus transfer fail interrupt enabled
iren2.7	irqn watchdog	0_{BIN}	R/W	Timeout of interrupt watchdog: • 0_{BIN} = Watchdog timeout interrupt disabled • 1_{BIN} = Watchdog timeout interrupt enabled
iren2.6	chip_mechancial_failure	0_{BIN}	R/W	Chip mechanical failure: • 0_{BIN} = Chip mechanical failure interrupt disabled • 1_{BIN} = Chip mechanical failure interrupt enabled
iren2.5	Renesas internal configuration	0_{BIN}	R	Renesas internal feature. Do not change the default setting.
iren2.4	outbuf_vcm_fail	0_{BIN}	R	VCM voltage level of SINP/N or COSP/N not equal to VDD/2: • 0_{BIN} = Common mode output voltage failure interrupt disabled • 1_{BIN} = Common mode output voltage failure interrupt enabled
irenl2.3	outbuf_current_fail	0_{BIN}	R	Output buffer SINP, SINN, COSP, COSN exceeds the limit: • 0_{BIN} = Output over-current interrupt disabled • 1_{BIN} = Output over-current interrupt enabled
iren2.2	r1r2_coil_short	0_{BIN}	R	Coil short between R1 and R2: • 0_{BIN} = Short between receive coils interrupt disabled • 1_{BIN} = Short between receive coils interrupt enabled
iren2.1	rc2_coil_fail	0_{BIN}	R	Receiver coil 2 short to GND/VDD, or the RX3/RX4 is open: • 0_{BIN} = Receive coil 2 failure interrupt disabled • 1_{BIN} = Receive coil 2 failure interrupt enabled
iren2.0	rc1_coil_fail	0_{BIN}	R	Receiver coil 1 short to GND/VDD, or the RX1/RX2 is open: • 0_{BIN} = Receive coil 1 fail interrupt disabled • 1_{BIN} = Receive coil 1 fail interrupt enabled

3.5.13. IRQN Watchdog 1

The watchdog is sending an interrupt signal periodically for checking whether the overall interrupt system is working correctly.

The interrupt request signaling can be asserted by a 22-bit watchdog timer. The clock driving the watchdog, is the calibrated internal 7MHz clock divided by two (digital clock). The timer pDefault value is defined by the IRQN watchdog LSB and MSB parameters (see register address $0E_{HEX}$ in Table 33). If the IRQN watchdog LSB parameter is not equal to zero, the timer is set to pDefault and decrement. Once the timer expires (counter state = zero), the interrupt is sent. The watchdog counter automatically pDefault again and decrement independently of the interrupt state. If the default of the IRQN watchdog parameter is set to zero, the restart of the watchdog is disabled.

Table 30. IRQN Watchdog 1 Register Details

Block	NVM	SRB	SFR
Address	$0D_{HEX}$	$2D_{HEX}$	–
Default value	–	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 31. IRQN Watchdog 1 Register Bit Descriptions

Bit	Symbol	Default	Type	Description
irqn1.15 irqn1.14 irqn1.13 irqn1.12	ECC			Parity word for error correction. For the internal use of the IC.
irqn1.11	P_{DED}			Parity bit for double-bit error detection.
irqn1.10 irqn1.9 irqn1.8 irqn1.7 irqn1.6 irqn1.5 irqn1.4 irqn1.3 irqn1.2 irqn1.1 irqn1.0	IRQN watchdog LSB	$000\ 0000\ 0000_{BIN}$	R/W	LSB of interrupt request watchdog timer.

3.5.14. IRQN Watchdog 2

The watchdog is sending an interrupt signal periodically for checking whether the overall interrupt system is working correctly.

The interrupt request signaling can be asserted by a 22-bit watchdog timer. The clock driving the watchdog, is the calibrated internal 7MHz clock divided by two (digital clock). The timer pDefault value is defined by the IRQN watchdog LSB and MSB parameters (see register address $0D_{HEX}$ in Table 31). Once the timer expires (counter state = zero), the interrupt is sent. The watchdog counter automatically pDefault again and decrement independently of the interrupt state. If the default of the IRQN watchdog parameter is set to zero, the restart of the watchdog is disabled.

Table 32. IRQN Watchdog 2 Register Details

Block	NVM	SRB	SFR
Address	$0E_{HEX}$	$2E_{HEX}$	–
Default value	–	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 33. IRQN Watchdog 2 Register Bit Descriptions

Bit	Symbol	Default	Type	Description
irqn2.15 irqn2.14 irqn2.13 irqn2.12	ECC			Parity word for error correction. For the internal use of the IC.
irqn2.11	P_{DED}			Parity bit for double-bit error detection.
irqn2.10 irqn2.9 irqn2.8 irqn2.7 irqn2.6 irqn2.5 irqn2.4 irqn2.3 irqn2.2 irqn2.1 irqn2.0	IRQN watchdog MSB	$000\ 0000\ 0000_{BIN}$	R/W	MSB of interrupt request watchdog timer.

3.5.15. R1 Fine Gain

Production tolerances, non-idealities in the coil design and mechanical setup could generate a mismatch between the two output signal amplitudes.

To minimize this difference between the R1 and R2 signal amplitudes, the integrator gain can be modified by bit configuration. The bit configuration is linked with the tuning of the capacitive component of the integrator.

Table 34. R1 Fine Gain Register Details

Block	NVM	SRB	SFR
Address	12_{HEX}	32_{HEX}	–
Default value	0000_{HEX}	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 35. R1 Fine Gain Register Bit Definitions

Bit	Symbol	Default	Type	Description
r1_fine_gain.15 r1_fine_gain.14 r1_fine_gain.13 r1_fine_gain.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
r1_fine_gain.11	P_{DED}	–	–	Parity bit for double bit error detection.
r1_fine_gain.10	–	–	–	Not used, read as 0.
r1_fine_gain.9	–	–	–	Not used, read as 0.
r1_fine_gain.8	–	–	–	Not used, read as 0.
r1_fine_gain.7	–	–	–	Not used; read as 0.
r1_fine_gain.6 r1_fine_gain.5 r1_fine_gain.4 r1_fine_gain.3 r1_fine_gain.2 r1_fine_gain 1 r1_fine_gain.0	R1 gain correction	$000\ 0000_{BIN}$	R/W	R1 channel swapping amplitude mismatch correction: • 0x00 - no correction • 0x01 - 0.09% • 0x02 - 0.18% • 0x03 - 0.27% ... • 0x7E - 10.91% • 0x7F - 11.00% Overall range is 11%, step is 0.09%

3.5.16. R2 Fine Gain

The difference between the amplitudes of the two input signals (R1 and R2) are generated from:

- the amplifiers have different characteristics due to production tolerance
- coils design and the non–uniformity of the EMF

To minimize this difference between the R1 and R2, the integrator gain is modified by bit configuration. The bit configuration is linked with the tuning of the capacitive component of the integrator.

Table 36. R2 Fine Gain Register Details

Block	NVM	SRB	SFR
Address	13_{HEX}	33_{HEX}	–
Default value	0000_{HEX}	0000_{HEX}	–
Access	READ/WRITE	READ/WRITE	–

Table 37. R2 Fine Gain Register Bit Definitions

Bit	Symbol	Default	Type	Description
r2_fine_gain.15 r2_fine_gain.14 r2_fine_gain.13 r2_fine_gain.12	ECC	–	–	Parity word for error correction. For the internal use of the IC.
r2_fine_gain.11	P_{DED}	–	–	Parity bit for double bit error detection.
r2_fine_gain.10	–	–	–	Not used, read as 0
r2_fine_gain.9	–	–	–	Not used, read as 0

Bit	Symbol	Default	Type	Description
r2_fine_gain.8	–	–	–	Not used, read as 0
r2_fine_gain.7	–	–	–	Not used; read as 0.
r2_fine_gain.6 r2_fine_gain.5 r2_fine_gain.4 r2_fine_gain.3 r2_fine_gain.2 r2_fine_gain 1 r2_fine_gain.0	R2 gain correction	$000\ 0000_{BIN}$	R/W	R2 Channel swapping amplitude mismatch correction: • 0x00 - no correction • 0x01 - 0.09% • 0x02 - 0.18% • 0x03 - 0.27% ... • 0x7E - 10.91% • 0x7F - 11.00% Overall range is 11%, step is 0.09%

3.5.17. Product Identification

Important: Registers at NVM address 19_{HEX} to $1F_{HEX}$ are for Renesas internal product identification, and have read-only access.

3.5.18. Interrupt Clear 1

The interrupt function of IPS2200 is divided into static or dynamic events. The static event is considered when the diagnostics is latched until POR, while the dynamic event is active only during the presence of the diagnostics.
In the present of the dynamic event, choose either of the following options:

- Use a microcontroller to clear the register. The benefit of this method is to understand the process causing the interrupt and the type of diagnostics causing the error.
- Use self-clear form the IC by setting: Addr: 0x03 bit [10] - intr_volatile_mode.

The following six registers describe the diagnostics and the error causing the interrupt:

- Interrupt Enable 1/2: they control the Interrupt pin of the IC, see sections 3.5.11 and 3.5.12.
- Interrupt Clear 1/2: they are used by the microcontroller to clear the event and to check the type of the diagnostics, see sections 3.5.18 and 3.5.19.
- Interrupt State 1/2: they show whether there is a diagnostic event, see sections 3.5.20 and 3.5.21.

For the interrupt handling of the module, all interrupt status bits can be individually cleared by the corresponding ircl1.x bits.

Table 38. Interrupt Clear 1 Register Details

Block	NVM	SRB	SFR
Address	–	–	34_{HEX}
Default value	–	–	–
Access	–	–	Write-only; read as '0.'

Table 39. Interrupt Clear 1 Register Bit Definitions

Bit	Symbol	Default	Type	Description
ircl1.15	–	–	–	Not used.
ircl1.14	–	–	–	Not used.
ircl1.13	–	–	–	Not used.
ircl1.12	–	–	–	Not used.
ircl1.11	–	–	–	Not used.
ircl1.10	Over-voltage	0_{BIN}	W	Over-voltage detection: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.10
ircl1.9	Under-voltage	0_{BIN}	W	Under-voltage detection: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.9
ircl1.8	Renesas internal configuration	0_{BIN}	W	Renesas internal feature. Do not change the default setting.

Bit	Symbol	Default	Type	Description
ircl1.7	data_access_fail	0_{BIN}	W	SPI/I2C data access timeout: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.7
ircl1.6	spi_I2C_protocol_fail	0_{BIN}	W	SPI/I2C interface serial received data fail/data access aborted (for error handling details, see Table 50): • 0_{BIN} = No action • 1_{BIN} = Default irstate1.6
ircl1.5	srb_ded	0_{BIN}	W	SRB double-bit error detection/SFR test mode register 1/2 parity bit error detection: • 0_{BIN}= No action • 1_{BIN}= Default irstate1.5
ircl1.4	srb_sed	0_{BIN}	W	SRB single-bit error detection and correction: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.4
ircl1.3	nvm_ded	0_{BIN}	W	NVM double-bit error detection: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.3
ircl1.2	nvm_sed	0_{BIN}	W	NVM single-bit error detection and correction: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.2
ircl1.1	lc_osc_freq_fail	0_{BIN}	W	LC oscillator frequency range check (see registers 07_{HEX}, 08_{HEX}, $0A_{HEX}$ for the limits): • 0_{BIN} = No action • 1_{BIN} = Default irstate1.1
ircl1.0	lc_osc_stuck	0_{BIN}	W	LC oscillator stuck check: • 0_{BIN} = No action • 1_{BIN} = Default irstate1.0

3.5.19. Interrupt Clear 2

The interrupt function of IPS2200 is divided into static or dynamic events. The static event is considered when the diagnostics is latched until POR, while the dynamic event is active only during the presence of the diagnostics.
In the present of the dynamic event, choose either of the following options:

- Use a microcontroller to clear the register. The benefit of this method is to understand the process causing the interrupt and the type of diagnostics causing the error.
- Use self-clear form the IC by setting: Addr: 0x03 bit [10] - intr_volatile_mode.

The following six registers describe the diagnostics and the error causing the interrupt:

- Interrupt Enable 1/2: they control the Interrupt pin of the IC, see sections 3.5.11 and 3.5.12.
- Interrupt Clear 1/2: they are used by the microcontroller to clear the event and to check the type of the diagnostics, see sections 3.5.18 and 3.5.19.
- Interrupt State 1/2: they show whether there is a diagnostic event, see sections 3.5.20 and 3.5.21.

For the interrupt handling of the module, all interrupt status bits can be individually cleared by the corresponding ircl2.x bits.

Table 40. Interrupt Clear 2 Register Details

Block	NVM	SRB	SFR
Address	–	–	35_{HEX}
Default value	–	–	–
Access	–	–	Write-only; read as '0.'

Table 41. Interrupt Clear 2 Register Bit Definitions

Bit	Symbol	Default	Type	Description
ircl2.15	–	–	–	Not used; read as 0.
ircl2.14	–	–	–	Not used; read as 0.

Bit	Symbol	Default	Type	Description
ircl2.13	–	–	–	Not used; read as 0.
ircl2.12	–	–	–	Not used; read as 0.
ircl2.11	–	–	–	Not used; read as 0.
ircl2.10	vdda_csn_data	0_{BIN}	R	VDDA under-voltage or CSN/DATA pin over-voltage: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.10
ircl2.9	low_amplitude	0_{BIN}	R	Low amplitude: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.9
ircl2.8	apb_transfer_fail	0_{BIN}	R	Internal bus transfer failure. Any write access of a read-only register is rejected and flagged by the internal bus control (see sys_conf2.6). • 0_{BIN} = No action • 1_{BIN} = Default irstate2.8
ircl2.7	irqn watchdog	0_{BIN}	R	Timeout of interrupt watchdog: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.7
ircl2.6	chip_mechancial_failure	0_{BIN}	R	Chip mechanical failure: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.6
ircl2.5	Renesas internal configuration	0_{BIN}	R	Renesas internal feature. Do not change the default setting.
ircl2.4	outbuf_vcm_fail	0_{BIN}	R	VCM voltage level of SINP/N or COSP/N not equal to VDD/2: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.4
ircl2.3	outbuf_current_fail	0_{BIN}	R	Output buffer SINP, SINN, COSP, COSN exceeds the limit: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.3
ircl2.2	r1r2_coil_short	0_{BIN}	R	Coil short between R1 and R2 • 0_{BIN} = No action • 1_{BIN} = Default irstate2.2
ircl2.1	rc2_coil_fail	0_{BIN}	R	Receiver coil 2 short to GND/VDD, or the RX3/RX4 is open: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.1
ircl2.0	rc1_coil_fail	0_{BIN}	R	Receiver coil 1 short to GND/VDD, or the RX1/RX2 is open: • 0_{BIN} = No action • 1_{BIN} = Default irstate2.0

3.5.20. Interrupt State 1

The interrupt function of IPS2200 is divided into static or dynamic events. The static event is considered when the diagnostics is latched until POR, while the dynamic event is active only during the presence of the diagnostics.
In the present of the dynamic event, choose either of the following options:

- Use a microcontroller to clear the register. The benefit of this method is to understand the process causing the interrupt and the type of diagnostics causing the error.
- Use self-clear form the IC by setting: Addr: 0x03 bit [10] - intr_volatile_mode.

The following six registers describe the diagnostics and the error causing the interrupt:

- Interrupt Enable 1/2: they control the Interrupt pin of the IC, see sections 3.5.11 and 3.5.12.
- Interrupt Clear 1/2: they are used by the microcontroller to clear the event and to check the type of the diagnostics, see sections 3.5.18 and 3.5.19.
- Interrupt State 1/2: they show whether there is a diagnostic event, see sections 3.5.20 and 3.5.21.

Table 42. Interrupt State 1 Register Details

Block	NVM	SRB	SFR
Address	–	–	36_{HEX}
Default value	–	–	–
Access	–	–	Read -only

Table 43. Interrupt State 1 Register Bit Definitions

Bit	Symbol	Default	Type	Description
irstate1.15	–	–	–	Not used; read as 0.
irstate1.14	–	–	–	Not used; read as 0.
irstate1.13	–	–	–	Not used; read as 0.
irstate1.12	–	–	–	Not used; read as 0.
irstate1.11	–	–	–	Not used; read as 0.
irstate1.10	Over-voltage	0_{BIN}	R	Over-voltage detection: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate1.9	Under-voltage	0_{BIN}	R	Under-voltage detection: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate1.8	Renesas internal	0_{BIN}	R	Renesas internal feature. Do not change the default setting.
irstate1.7	data_access_fail	0_{BIN}	R	SPI/I2C data access timeout: • 0_{BIN}= Idle • 1_{BIN} = Interrupt request
irstate1.6	spi_I2C_protocol_fail	0_{BIN}	R	SPI/I2C interface serial received a data fail/data access aborted (for error handling details, see Table 50): • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate1.5	srb_sfr_ded	0_{BIN}	R	SRB double-bit error detection/SFR test mode register 1/2 parity bit error detection: • 0_{BIN}= Idle • 1_{BIN} = Interrupt request
irstate1.4	srb_sed	0_{BIN}	R	SRB single bit error detection and correction: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate1.3	nvm_ded	0_{BIN}	R	NVM double-bit error detection: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate1.2	nvm_sed	0_{BIN}	R	NVM single-bit error detection and correction: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request

Bit	Symbol	Default	Type	Description
irstate1.1	lc_osc_freq_fail	0_{BIN}	R	LC oscillator frequency range check (see registers 07_{HEX}, 08_{HEX}, $0A_{HEX}$ for the limits): • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate1.0	lc_osc_stuck	0_{BIN}	R	LC oscillator stuck check: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request

3.5.21. Interrupt State 2

The interrupt function of IPS2200 is divided into static or dynamic events. The static event is considered when the diagnostics is latched until POR, while the dynamic event is active only during the presence of the diagnostics.
In the present of the dynamic event, choose either of the following options:

- Use a microcontroller to clear the register. The benefit of this method is to understand the process causing the interrupt and the type of diagnostics causing the error.
- Use self-clear form the IC by setting: Addr: 0x03 bit [10] - intr_volatile_mode.

The following six registers describe the diagnostics and the error causing the interrupt:

- Interrupt Enable 1/2: they control the Interrupt pin of the IC, see sections 3.5.11 and 3.5.12.
- Interrupt Clear 1/2: they are used by the microcontroller to clear the event and to check the type of the diagnostics, see sections 3.5.18 and 3.5.19.
- Interrupt State 1/2: they show whether there is a diagnostic event, see sections 3.5.20 and 3.5.21.

Table 44. Interrupt State 2 Register Details

Block	NVM	SRB	SFR
Address	–	–	37_{HEX}
Default value	–	–	–
Access	–	–	Read-only

Table 45. Interrupt State 2 Register Bit Definitions

Bit	Symbol	Default	Type	Description
irstate2.15	–	–	–	Not used; read as 0.
irstate2.14	–	–	–	Not used; read as 0.
irstate2.13	–	–	–	Not used; read as 0.
irstate2.12	–	–	–	Not used; read as 0.
irstate2.11	–	–	–	Not used; read as 0.
irstate2.10	vdda_csn_data	0_{BIN}	R	VDDA under-voltage or CSN/DATA pin over-voltage: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.9	low_amplitude	0_{BIN}	R	Low amplitude: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.8	apb_transfer_fail	0_{BIN}	R	Internal bus transfer failure. Any write access of read-only register is rejected and flagged by the internal bus control (see sys_conf2.6 in section 3.5.2): • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.7	irqn watchdog	0_{BIN}	R	Timeout of interrupt watchdog: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.6	chip_mechancial_failure	0_{BIN}	R	Chip mechanical failure (enable check; see test2.11): • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.5	Renesas internal configuration	0_{BIN}	R	Renesas internal feature. Do not change the default setting.

Bit	Symbol	Default	Type	Description
irstate2.4	outbuf_vcm_fail	0_{BIN}	R	VCM voltage level of SIN/SINN or COS/COSN is not equal VDD/2: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.3	outbuf_current_fail	0_{BIN}	R	Output buffer SIN, SINN, COS, COSN exceeds the limit: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.2	r1r2_coil_short	0_{BIN}	R	Coil short between R1 and R2 coils: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.1	rc2_coil_fail	0_{BIN}	R	Receiver coil 2 short to GND/VDD, or the RX3/RX4 is open: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request
irstate2.0	rc1_coil_fail	0_{BIN}	R	Receiver coil 1 short to GND/VDD, or the RX1/RX2 is open: • 0_{BIN} = Idle • 1_{BIN} = Interrupt request

3.5.22. Transmitter Counter State

The transmitter counter is driven by the transmitter clock and gated/enabled by the transmitter time base counter state, see section 3.5.8. The relation between these two counters allows calculating the excitation time period/frequency:

t_{exc} = (TimeBaseCntr / TransmitterCntr) × t_{osc_dig}

Where:

TimeBaseCntr = Transmitter frequency time base, address 08_{HEX}, SRB 28_{HEX}

TransmitterCntr = SFR 38_{HEX} (this register)

t_{osc_dig} = Digital clock period

The internal oscillator is trimmed to 7MHz (142.86ns) and divided by 2: t_{osc_dig} = 285.72ns.

To read the T_X frequency, do set the following:

- Write timebase 350_{DEC} into 0x0028.
- Read 0x0038 Transmitter Counter state, for example 416DEC means 4.16Mhz.

Table 46. Transmitter Counter State Register Details

Block	NVM	SRB	SFR
Address	–	–	38_{HEX}
Default value	–	–	0000_{HEX}
Access	–	–	Read-only

Table 47. Transmitter Counter State Register Bit Definitions

Bit	Symbol	Default	Type	Description
transmitter_cntr.15	–	–	–	Not used; read as 0.
transmitter_cntr.14	–	–	–	Not used; read as 0.
transmitter_cntr.13	–	–	–	Not used; read as 0.
transmitter_cntr.12	–	–	–	Not used; read as 0.
transmitter_cntr.11	–	–	–	Not used; read as 0.

Bit	Symbol	Default	Type	Description
transmitter_cntr.10 transmitter_cntr.9 transmitter_cntr.8 transmitter_cntr.7 transmitter_cntr.6 transmitter_cntr.5 transmitter_cntr.4 transmitter_cntr.3 transmitter_cntr.2 transmitter_cntr.1 transmitter_cntr.0	LC oscillator counter state	$000\ 0000\ 0000_{BIN}$	R	Excitation frequency counter: • $000\ 0000\ 0000_{BIN}$ = 0 excitation cycle • $000\ 0000\ 0001_{BIN}$ = 1 excitation cycle • $000\ 0000\ 0010_{BIN}$ = 2 excitation cycles … • $111\ 1111\ 1111_{BIN}$ = 2047 excitation cycles

3.5.23. NVM ECC Fail State

The NVM 11-bit data word is protected by a 5-bit Hamming code. If a single-bit error occurs by reading the NVM addressed word, a single-bit error correction is done and flagged by the SEC diagnostic. If a double-bit error occurs, a DED diagnostic flag is asserted. If an SEC or DED assertion occurs, the parity protection bits and the corresponding word address of the last failed read is stored in the status register and can be monitored for NVM failure analysis.

Table 48. NVM ECC Status Register Details

Block	NVM	SRB	SFR
Address	–	–	$3A_{HEX}$
Default value	–	–	0000_{HEX}
Access	–	–	Read-only

Table 49. NVM ECC Status Register Bit Definitions

Bit	Symbol	Default	Type	Description
test1.15	–	–	–	Not used; read as 0.
test1.14	–	–	–	Not used; read as 0.
test1.13	–	–	–	Not used; read as 0.
test1.12	–	–	–	Not used; read as 0.
test1.11	–	–	–	Not used; read as 0.
test1.10	DED flag	0_{BIN}	R	Double-bit error detection flag: • 0_{BIN} = Idle • 1_{BIN} = Asserted
test1.9	SEC flag	0_{BIN}	R	Single-bit error correction flag: • 0_{BIN} = Idle • 1_{BIN} = Asserted
test1.8	SEC_ID8	0_{BIN}	R	Renesas internal use.
test1.7	SEC_ID4	0_{BIN}	R	Renesas internal use.
test1.6	SEC_ID2	0_{BIN}	R	Renesas internal use.
test1.5	SEC_ID1	0_{BIN}	R	Renesas internal use.
test1.4 test1.3 test1.2 test1.1 test1.0	NVM address	$0\ 0000_{BIN}$	R	NVM address with SEC or DED detection.

3.5.24. SFR Internal Registers

SFR registers $3B_{HEX}$ to $3F_{HEX}$ are for Renesas internal use only and they are read by SPI/I2C as 0.

4. Programming Interfaces

In order to program the device and to enable fast diagnostics without interrupting the analog high speed signal path, an additional digital serial interface is available.

The IPS2200 offers four modes of digital communication for the diagnostics and programming interface:

- I2C interface with address select (default setting)
- I2C interface with interrupt (programming option)
- Half duplex SPI interface with interrupt (programming option)
- Half duplex SPI interface (programming option)

4.1 Half-Duplex SPI Interface

This is a standard bi-directional, half-duplex SPI interface. The SPI slave module is activated by the SPI 3-wire master, which initiates the transaction by pulling the chip-select pin low (CSN_IRQN, pin 1). A serial clock (SCK_SCL, pin 15), is driven by the master. The Serial Data In/Out line (SIO_SDA, pin 16) is a bidirectional data line between master and slave. In a typical scenario, the master transmits a command with a specified length of 8-bit over the SIO line. If it is a write command, the master keeps transmitting data over the same line. If the first bits were a READ command, the slave transmits a fixed length of data over the SIO line to the master.

Note: In the following figures, for IPS2200 pins that have dual functions, the function that is active is shown in **bold** font**.**

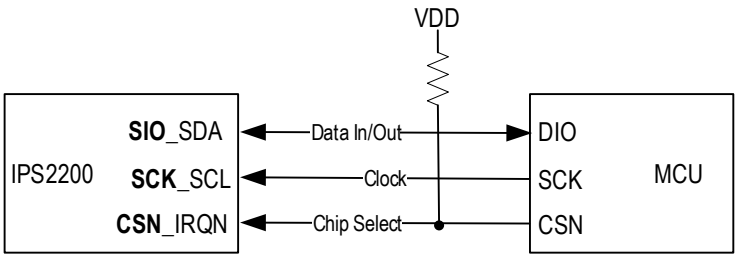


Figure 4. Half Duplex 3-3 Wire SPI Interface

A master can communicate with multiple slaves. Each slave device has an independent CSN line but shares the SCL and SIO lines with all slaves. A slave is only addressed when the corresponding CSN pin is pulled low.

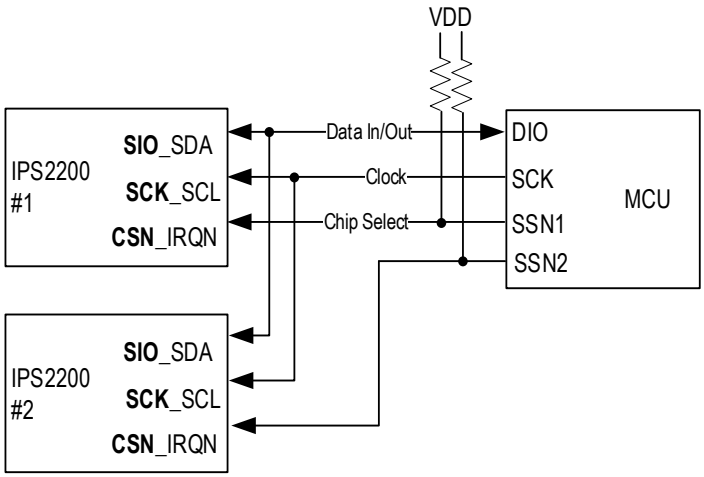


Figure 5. Half Duplex 3-3 Wire SPI Multi-slave Interface

The SPI interface has four modes of operation, based on two parameters:

- Clock polarity (CPOL)
- Clock phase (CPHA)

Master and slave must use the same mode to communicate articulately. If CPOL is zero, then SCK is normally low, and the first clock edge is a rising edge. If CPOL is one, SCL is normally high, and the first clock edge is a falling edge. CPHA defines the data alignment. If CPHA is zero then the first data bit is written on the CSN falling edge and read on the first SCK edge. If CPHA is one, data is written on the first edge and read on the second SCK edge.

Another programming option for SPI mode is the order at which the data is received:

- MSB first, LSB last
- LSB first, MSB last

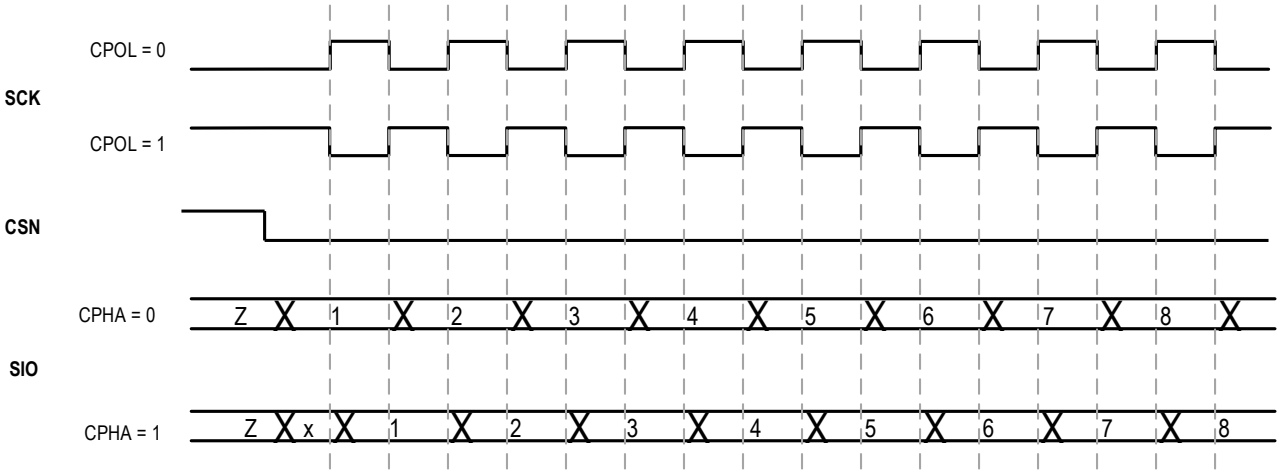


Figure 6. SPI 3-Wire Operating Mode

4.1.1. Interfacing the IPS2200 with a 4-Wire SPI Master

The IPS2200, using a 3-wire bi-directional half-duplex SPI interface, can be connected to a 4-wire SPI master by connecting the MOSI (master out, slave in) output to the MISO (master in, slave out) with a series resistor as shown in Figure 7. In this mode, the master must ensure that the MOSI output stays high while the MISO input is receiving data from the IPS2200 slave.

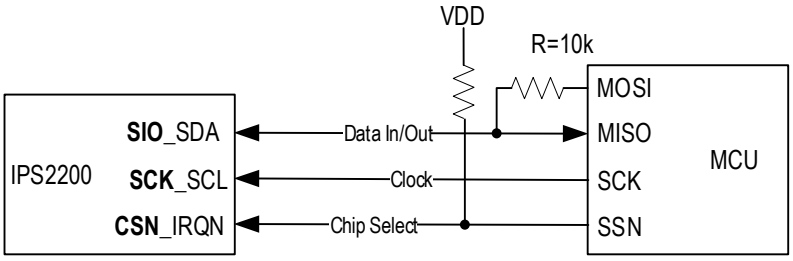


Figure 7. Half Duplex 3-4 Wire SPI Interface

All data transfers are framed by the CSN signal, which must be active low for any data transfer to occur. The beginning of each data transfer is defined by a high to low transition on the CSN signal. The first 8-bits sent after CSN is pulled low by the master comprise the Address/Command byte, which tells the slave device if the data transfer is a READ or a WRITE cycle and which register will be read from or written to; see Figure 8.

4.1.2. Bit Error Checks

Bit 7 of the first byte in the data frame defines the data integrity check.

If the SPI data integrity check is activated, the MSB (A7) is the even parity bit (P) of the Address/Command byte. If the parity check fails, the data bits of the write access is disabled and the diagnostic flag “protocol_integrity_fail” is asserted.

For data transfers without the bit error check, the MSB (A7) must be set to A7= logic ‘1’. If it is zero, WRITE access to the memory is disabled and the diagnostic flag “protocol_integrity_fail” is asserted.

If a READ access fails, the SIO line is pulled low for the complete access frame and the diagnostic flag “protocol_integrity_fail” is asserted.

A6 defines the memory area (NVM or register). A5 to A1 specify the designated address space, and the LSB (A0) specifies the direction of the subsequent 2 bytes:

- A0 = logic ‘0’ = WRITE operation = data input
- A0 = logic ‘1’ = READ operation = data output

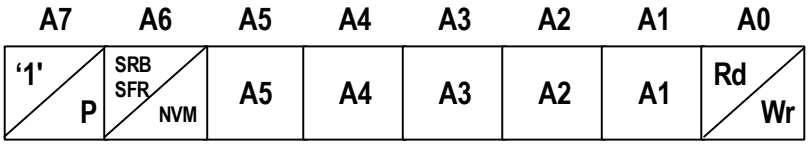


Figure 8. SPI 3-Wire Address/Command Byte

Figure 8, Figure 9, and Figure 10 show the SPI operation mode CPOL= ‘0’, CPHA= ‘0’ and DODR= ‘0’ (MSB first).

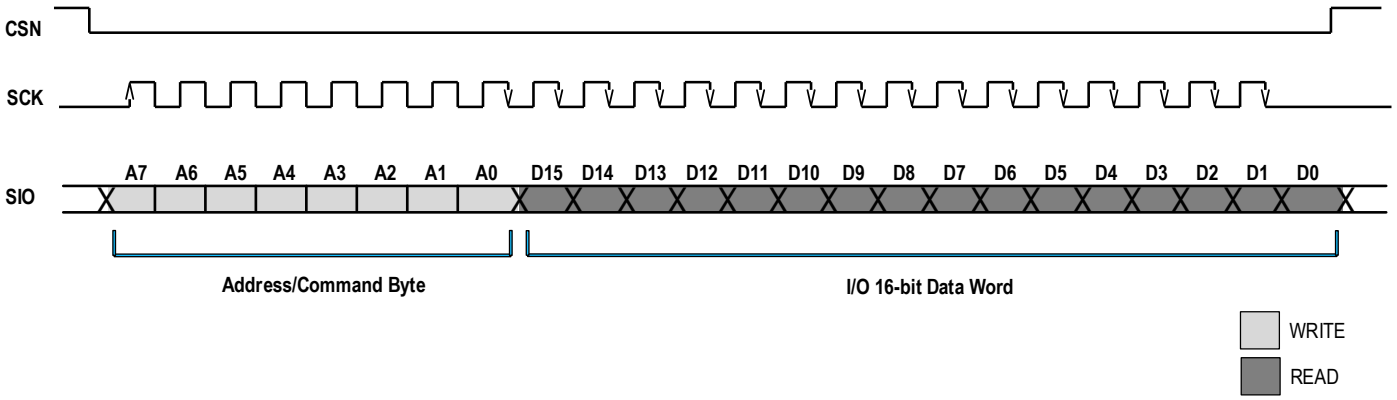
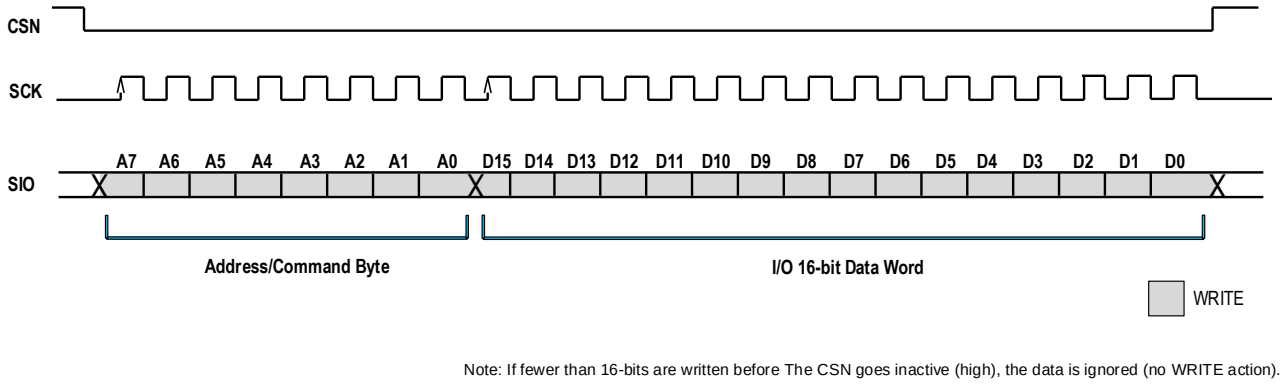


Figure 9. SPI Single Word Read



Note: If fewer than 16-bits are written before The CSN goes inactive (high), the data is ignored (no WRITE action).

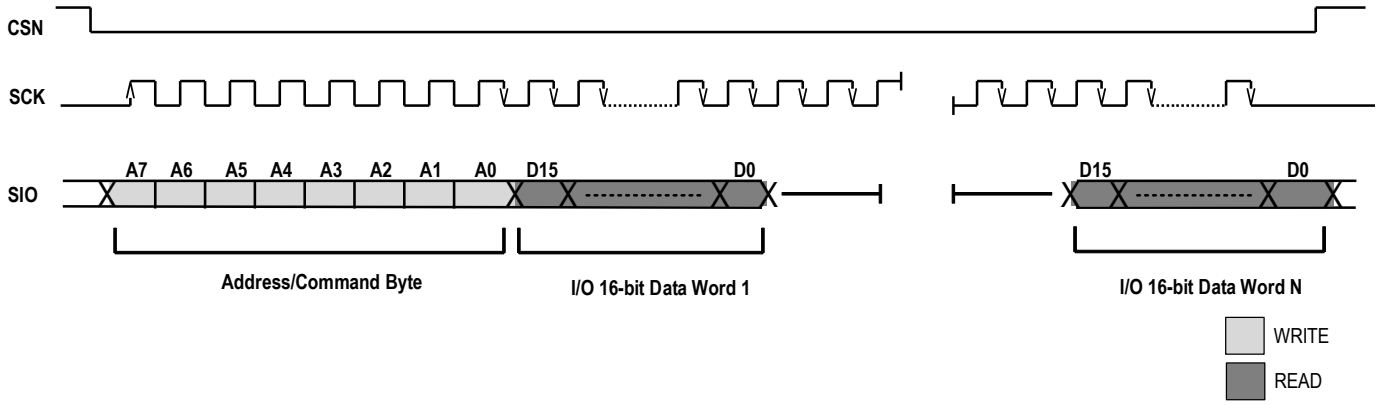
Figure 10. SPI Single Word Write

4.1.3. Burst READ Operation

In a burst READ operation, multiple consecutive registers or memory cells can be read in a single READ operation. The Address/Command byte is considered as the start-address, and by keeping CSN low, several successive 16-bit words of data are read. Data in consecutive addresses beginning with the starting address are read as long as CSN is low and SCK continues clocking. If the master sends more clocks than the number required for a specific burst location, the address burst counter rolls over and data from the original starting burst address is repeated

Note: A READ operation from a write-only register results in reading logic 0's.

Figure 11. SPI Burst READ



4.1.4. Burst WRITE Operation

Similar to a burst READ operation, it is also possible to write data to consecutive addresses of registers. For a burst WRITE operation, a distinctive Address/Command byte is written by the master (A6 = '1') indicating a WRITE operation to a shadow register or a special function register. The register address range is written to consecutive addresses beginning at the starting address, as long as CSN is low and SCK continues clocking. If the master sends more clocks than the number required for a specific burst location, the address burst counter wraps around, and writing data to the original starting burst address is repeated.

Note that due to the requirements for a minimum programming time, burst WRITE operations to NVM memory are not possible. If the master attempts a burst WRITE to a NVM memory location, only the first memory address will be programmed, and the consecutive WRITE addresses is not programmed.

Each data word has a length of 16-bit. If fewer than 16 bits of the last data word are written before CSN goes inactive (high), the data is ignored (no WRITE action).

Note: Data written into read-only registers is ignored.

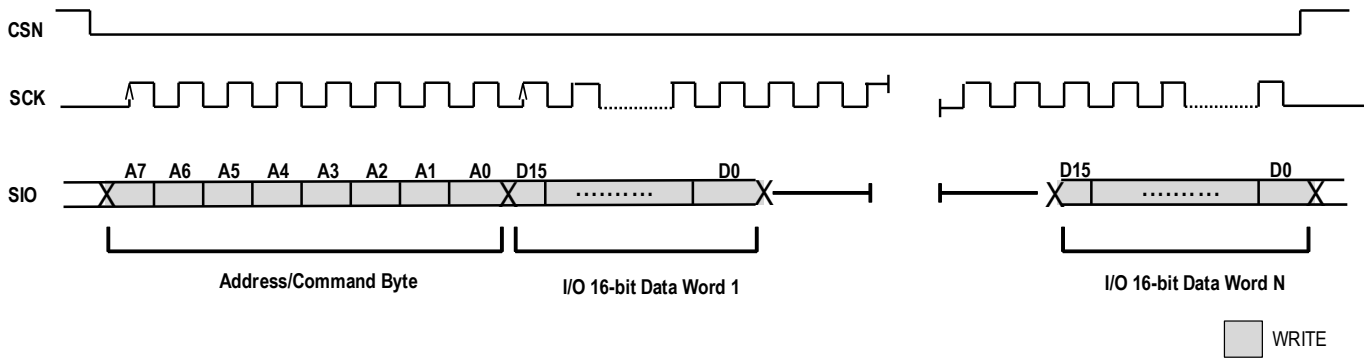


Figure 12. SPI Burst WRITE

4.2 SPI Data Word Definition

Each SPI data word has 16 bits. The first 5 bits are used for error code correction, and the remaining 11 bits are used for data.

The SPI data transfer is byte-oriented with an Address/Command byte, and each word with a size of two bytes.

The 16-bit data word has the following field structure:

- D15 to D5 <CmdData> Memory Information
- D4 to D0 <Reserved> (Always all 1's)

The <Reserved> field is checked for all 1's, and if the check fails, the <CmdData> field is not written to the addressed memory space (no WRITE action) and the diagnostic flag "protocol_integrity_fail" is asserted.

4.2.1. SPI Data Integrity Check

The Data Integrity Check is an extension of the SPI protocol for fail-safe communication between SPI master and slave device.

This check is activated by setting the NVM parameter "Protocol Integrity Check" = 1.

The SPI slave supports an even-parity check of the Address/Command byte and a 3-bit CRC validation of the 16-bit data word. If these checks fail, the diagnostic status "protocol_integrity_fail" is asserted.

The 3-bit CRC field is used to detect the presence of errors in the transmission of each 16-bit data word. If the transferred data does not pass CRC verification, no write action takes place by the slave.

The polynomial used to calculate the 3-bit CRC is:

CRC(x) = x3 + x1 + x0

The Slave transmitter uses the CRC-3 polynomial on the first 13 bits (D15 to D3) and makes a 3-bit augmentation of 000_{BIN} to 16-bit.

- D15 to D5 <CmdData> Memory Information
- D4 to D3 <Reserved> (Always all 1's = 11_{BIN})
- D2 to D0 000_{BIN}

The CRC is generated from the data structure above and replaces the lower 3 bits (D2 to D0):

The transmitted data structure is

- D15 to D5 <CmdData> Memory Information
- D4 to D3 <Reserved> (Always all 1's = 11_{BIN})
- D2 to D0 <CRC>

The receiver then takes the entire 16-bit data word, including the CRC itself, and uses the same polynomial to confirm integrity of the data by verifying the CRC.

The CRC checksum for SPI belongs to the group of cyclic binary codes. The characteristic of cyclic binary code is that each control bit can be considered as a parity bit of a certain group of message bits, which depends on the generator polynomial.

It follows for the control bit calculation (transmitter):

D_2 = XOR (D_{13}, D_{12}, D_{11}, D_9, D_6, D_5, D_4)

D_1 = XOR (D_{15}, D_{12}, D_{11}, D_{10}, D_8, D_5, D_4, D_3)

D_0 = XOR (D_{14}, D_{13}, D_{12}, D_{10}, D_7, D_6, D_5, D_3)

It follows for the CRC verification (receiver):

XOR (D_{13}, D_{12}, D_{11}, D_9, D_6, D_5, D_4, D_2) = 0

XOR (D_{15}, D_{12}, D_{11}, D_{10}, D_8, D_5, D_4, D_3, D_1) = 0

XOR (D_{14}, D_{13}, D_{12}, D_{10}, D_7, D_6, D_5, D_3, D_0) = 0

Table 50. SPI Address / Command Byte Integrity Check

Protocol Integrity Check	A7	A6 to A1	A0	Diagnostic	Description
0_{BIN}	0_{BIN}	xxxxxx$_{BIN}$	0_{BIN}	Asserted	No write action
0_{BIN}	0_{BIN}	xxxxxx$_{BIN}$	1_{BIN}	Idle	Read action
0_{BIN}	1_{BIN}	xxxxxx$_{BIN}$	0_{BIN}	Idle	Write action
0_{BIN}	1_{BIN}	xxxxxx$_{BIN}$	1_{BIN}	Idle	Read action
1_{BIN}	P_{pass}	xxxxxx$_{BIN}$	0_{BIN}	Idle	Parity check pass, write action
1_{BIN}	P_{pass}	xxxxxx$_{BIN}$	1_{BIN}	Idle	Parity check pass, read action
1_{BIN}	P_{fail}	xxxxxx$_{BIN}$	0_{BIN}	Asserted	Parity check fail, no action
1_{BIN}	P_{fail}	xxxxxx$_{BIN}$	1_{BIN}	Asserted	Parity check fail, no action Read I/O word = 0x0000

4.3 Half-Duplex SPI Interface with Interrupt (Programming Option)

This is a standard bi-directional SPI interface with an additional interrupt output function at the CSN input.

In addition to the standard half-duplex data transmission described in section 4.1, this mode allows an additional fast diagnostic alarm interrupt signaling from the IPS2200 sensor IC via the chip select line (CSN_IRQN):

- In normal operation, when no data is being transmitted, the CSN_IRQN line is high.
- Data transfer is initiated from the SPI master (MCU) by pulling the CSN_IRQN line low.
  - As soon as the MCU has pulled CSN_IRQN low, the SPI slave (sensor IC) is selected and ready to transmit and receive data.
  - During data transmission, the CSN_IRQN pin of the sensor IC is a logic input.
  - When data transfer is completed the SPI master returns CSN_IRQN to the high level.
  - In this state, when no data is transmitted, the MCU port (CSN_IRQN) switches to digital input and enables the pin as an active low interrupt output; see Figure 15.
- The open drain output on the CSN_IRQN has a built-in current limiter (current source) and an external pull-up resistor to avoid damage to the pin in the event of a data collision of two outputs (if the CSN_IRQN output is driven by the MCU = high and at the same time IRQN is driven by the SPI slave = low).
- In the alarm state, CSN_IRQN stays low until the first rising edge of the clock signal SCK at the beginning of a data transmission occurs on pin SCK_SCL. The IPS2200 deactivates the alarm state to allow detection of a CSN low signal from the controller. If CSN_IRQN goes high, incoming data on pin DIO is ignored, and the alarm on CSN_IRQN is activated again on the next falling edge of the SCK signal. If CSN_IRQN stays low when the alarm mode is turned off, the slave is selected and incoming data is acquired.

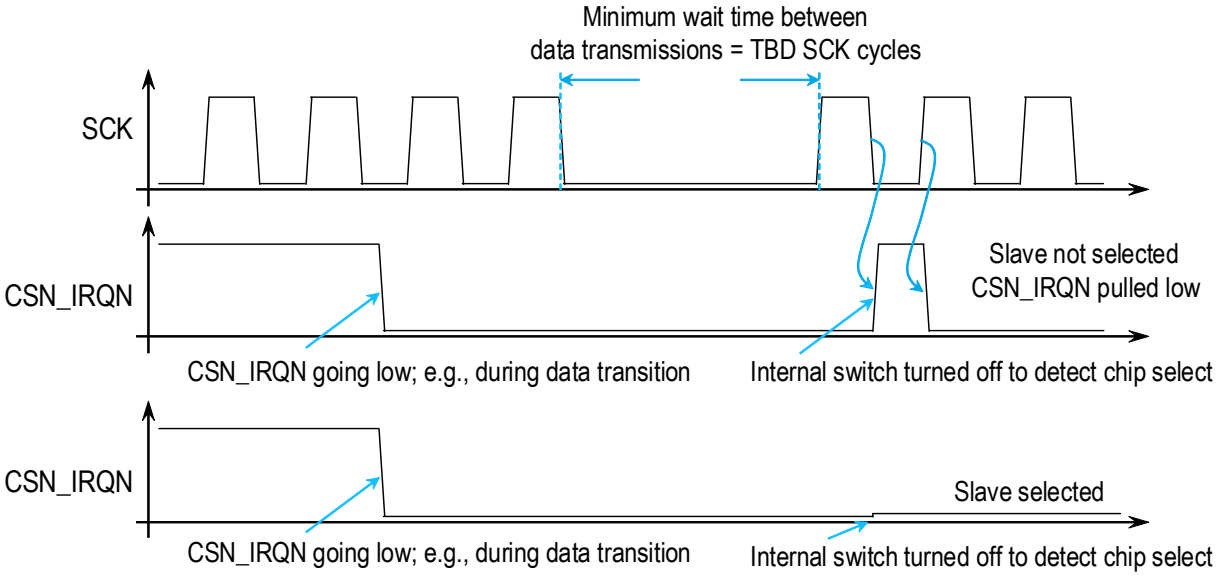


Figure 13. Half Duplex SPI IRQN Diagnostic Detection – Beginning of the Frame

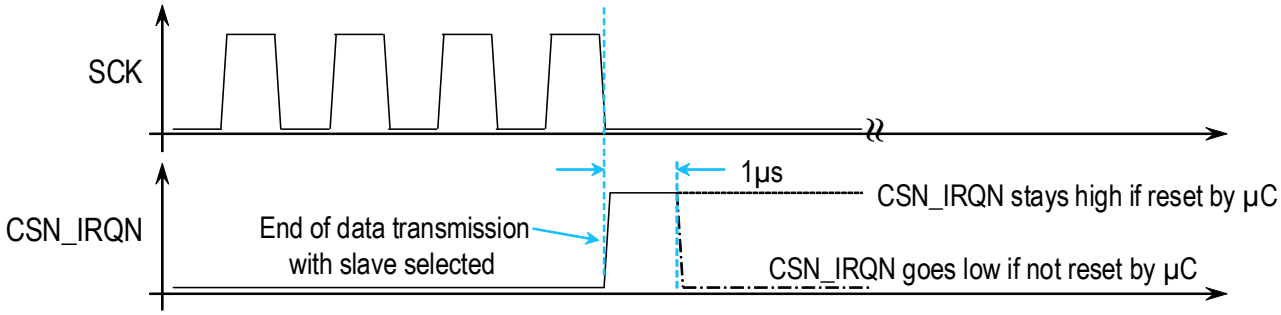


Figure 14. Half Duplex SPI IRQN Diagnostic Detection – End of the Frame

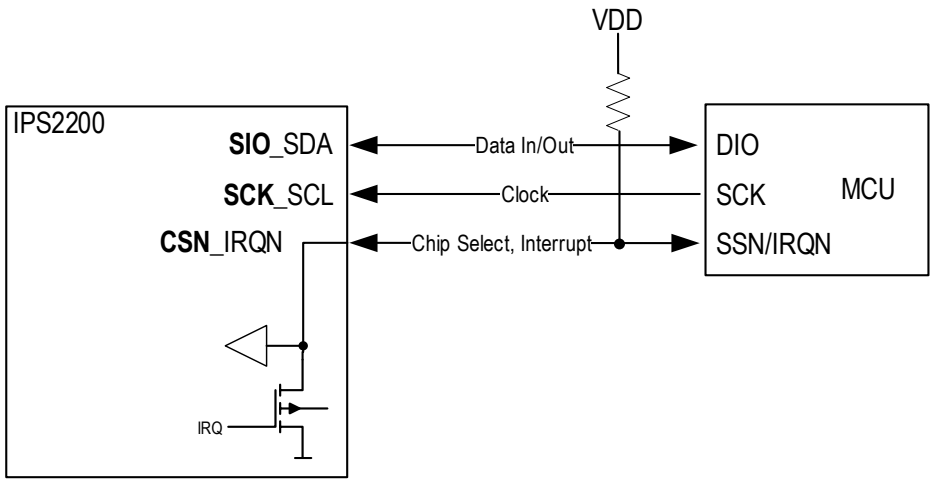


Figure 15. Half Duplex 3-3 Wire SPI Interface with Interrupt

On the MCU side, the chip select output and interrupt are usually separate pins. In this case, the two pins can be tied together and connected to the CSN_IRQN pin of the IPS2200.

The MCU software must ensure that a normal READ/WRITE operation, where CSN_IRQN is pulled low, does not generate an interrupt (disable IRQN before a normal READ/WRITE operation; enable IRQN after completion of a READ/WRITE operation).

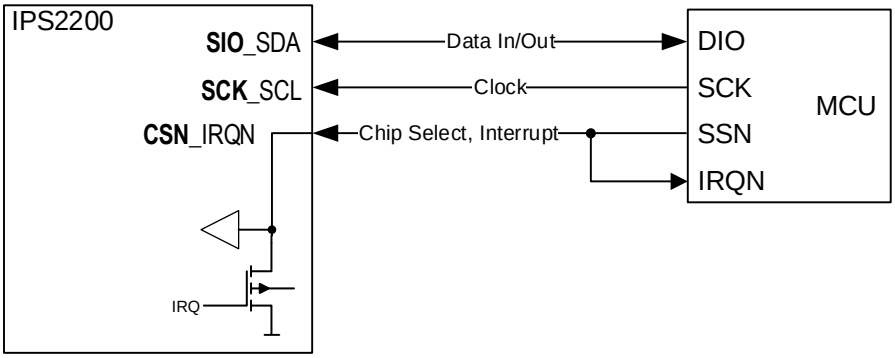


Figure 16. Half Duplex 3-3 Wire SPI Interface with CSN_IRQN and IRQN Separated at MCU

4.4 I2C Interface

The default interface is standard I2C with pin 1 (CSn_IRQN) configured as address input. The I2C address is programmable, the default I2C slave address is 24dec (18hex) if address in is 'high' and 16dec (10 hex) if the address pin is 'low'. In addition, a third pin (#1) allows either an I2C address selection (SEL) or is configured as interrupt output (IRQN). The IPS2200 is configured as I2C slave, several slaves may be connected in parallel on the I2C bus.

Two wires, serial data (SIO_SDA, pin 16) and serial clock (SCK_SCL, pin 15), carry information between the devices connected to the bus. Both SDA and SCL are bi-directional lines, connected to the positive supply voltage VDD via an external pull-up resistor. When the bus is free, both lines are high. The output stages of devices connected to the bus must have an open-drain or open-collector to perform the wired-AND function.

An external master (host controller) initiates a transfer, generates clock signals, and terminates a transfer. The implementation supports the I2C slave function, which is addressed by the master and supports the I2C bus specification version 2.1.

The data on the SDA line must be stable during the high period of the clock. The high or low state of the data line can only change when the clock signal on the SCL line is low.

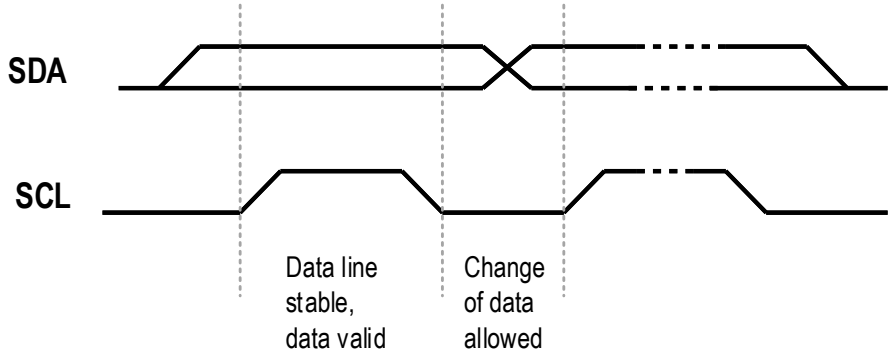


Figure 17. I2C Bit Transfer Signal State

In the procedures for the I2C bus, the START condition (S) is defined as a high to low transition on the SDA line while SCL is high and a STOP condition (P) is defined as a low to high transition on the SDA line while SCL is high, see Figure 18.

START and STOP conditions are always generated by the master. The bus is considered to be busy after the START condition. The bus is considered to be free again after the STOP condition.

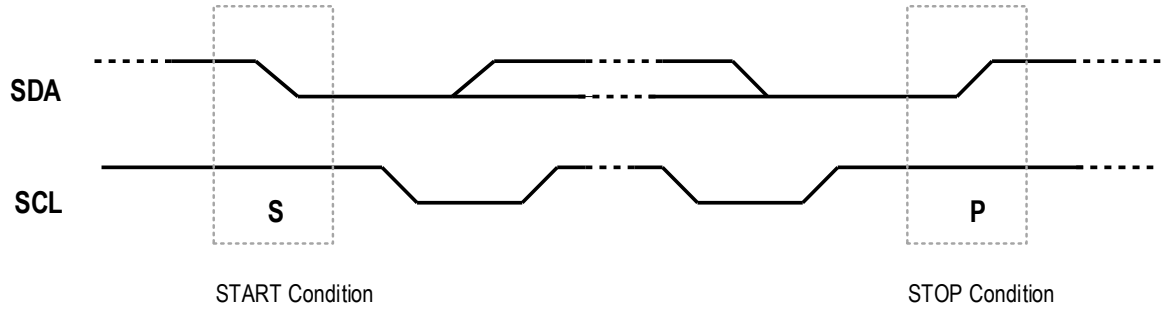


Figure 18. I2C START and STOP Conditions

If a repeated START (Sr) is generated instead of a STOP condition, the bus stays busy. In this respect, the START and repeated START conditions are functionally identical.

Every transfer block on the SDA line must have a length of 1 byte (8 bits). The number of bytes that can be transmitted per transfer is unrestricted. Each byte must be followed by an acknowledge bit by the I2C master (see Figure 19). Data is transferred with the most significant bit (MSB) first.

Data transfer with an acknowledgement is mandatory.

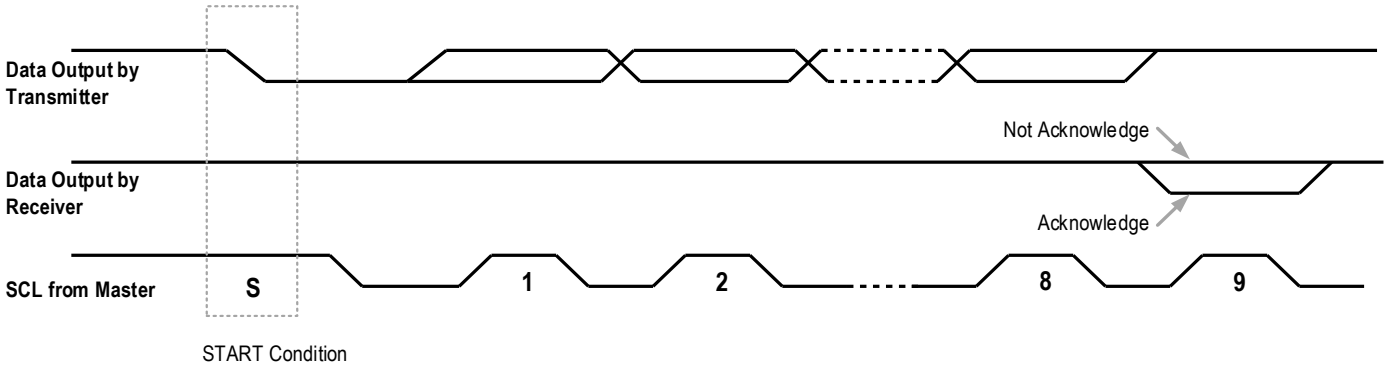


Figure 19. I2C Acknowledge Signal State

4.4.1. I2C Data Format

The I2C data transfer format is shown in Figure 21. After the START condition (S), a 7-bit slave address, followed by a READ (high) or WRITE (low) bit is sent by the master as shown in Figure 20.

Address bits A6 to A3 (shaded orange in Figure 20) can be individually programmed by the user, allowing up to 14 individual I2C slave addresses, ranging from 0001_{BIN} to 1110_{BIN} . Restricted addresses are 0000_{BIN} and 1111_{BIN}. Address bits A2 to A0 (shaded green in Figure 20) are reserved bits and always ‘0’. The programming of A6 to A3 can be done using an external programming tool.

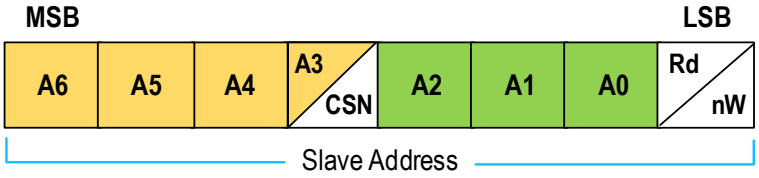


Figure 20. I2C Slave Address Byte

The subsequent bytes contain the requested data; each byte is followed by an acknowledge bit (ACK).

Data transfer is terminated by a STOP condition (P) generated by the master. However, if a master will be continuing to communicate on the bus, it can generate a repeated START condition (Sr) and address another slave without first generating a STOP condition. Various combinations of READ/WRITE formats are then possible within such transfers.

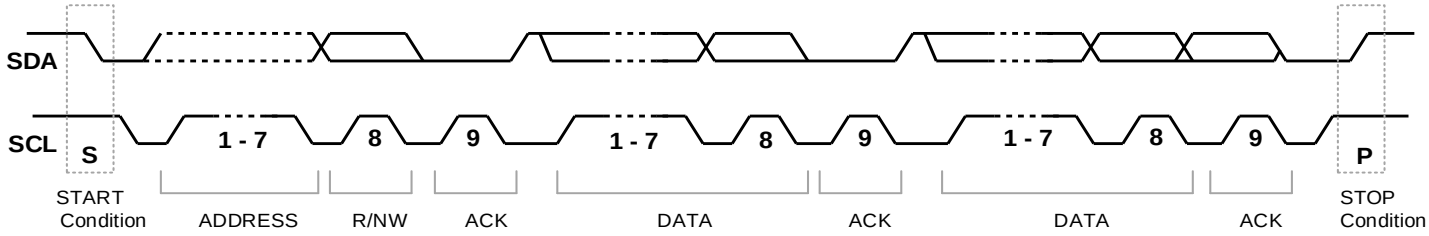


Figure 21. I2C Data Format

Possible data transfer formats are

- WRITE command to slave; the transfer direction is not changed (see Figure 24)
- BURST WRITE command to slave; the transfer direction is not changed (see Figure 26)
- READ from one or more slaves with repeated START condition (see Figure 26)

4.4.2. I2C Data Frame Structure

The I2C data frame consists of 6 key elements (see Table 3):

1. A start bit
1. Byte #1: Writes the slave address and direction of data flow
2. Byte #2: Writes the memory address of the selected slave
3. For read commands, a repeated start followed by Byte #1 changes the direction of data flow
4. Byte #3: Reads from or writes to the upper 8 bits of the selected memory address
5. Byte #4: Reads from or writes to the lower 3 bits of the selected memory address
6. A stop bit

4.4.2.1. Byte 1: I2C Slave Address and Direction of Data Flow

The addressing format of the I2C slave is a 7-bit address. The first byte after the START condition determines the slave address selected by the master.

The first seven bits of the first byte define the slave address; the eighth bit determines the direction of the data transfer.

LSB = low = the master will write information to a selected slave.

LSB = high = the master will read information from the slave.

4.4.2.2. Byte 2: Memory Address in Selected Slave

The first data byte after the slave address byte is defined as the memory address in the selected slave. The memory contains diagnostic status information or user configuration parameters.

D7 to D6 — These bits must always be written as 1’s; otherwise, the remaining 6 address bits are invalid and the slave does not respond with an acknowledge (ACK) bit.

D5 to D0 — These bits form the 6-bit memory address. Bit 5 (A6) selects either the nonvolatile memory (NVM) if A6 is low or a shadow register bank (SRB) or a special function register (SFR) if A6 is high.

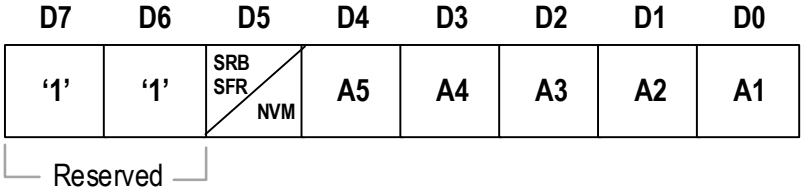


Figure 22. Memory Address in Selected I2C Slave

4.4.2.3. Bytes 3 and 4: 11-Bit Data of Selected Memory Address

Byte 3 contains the upper 8 bits of data of the selected memory address.

Byte 4 contains the lower 3 bits of data of the selected memory address.

In byte 4, bits [D4:D3] are reserved and fixed as ‘1’ and bits [D2:D0] are the CRC bits.

Data is read with the MSB first and LSB last.

For a WRITE command, data bits D3 to D4 must always be written as 1’s; otherwise the WRITE command is not executed and the slave does not respond with an acknowledgement (ACK) bit.

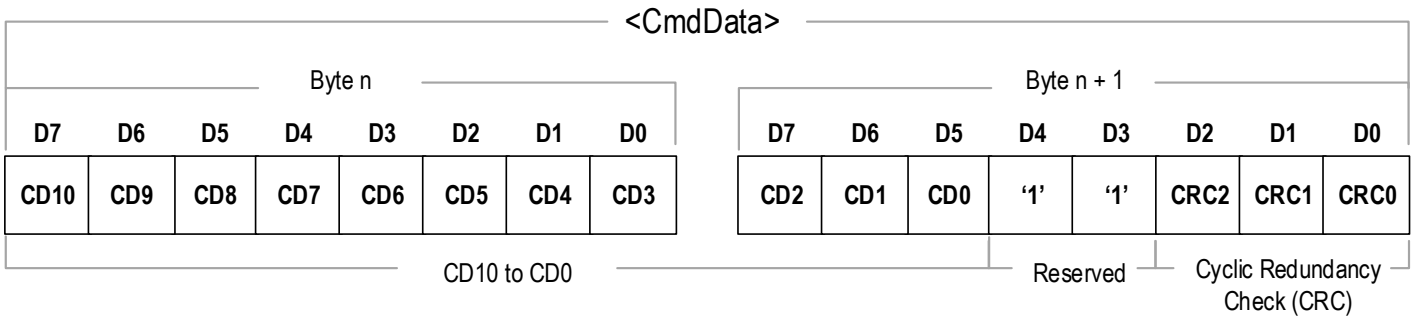


Figure 23. I2C 11-Bit Data Structure

4.4.3. WRITE Operations

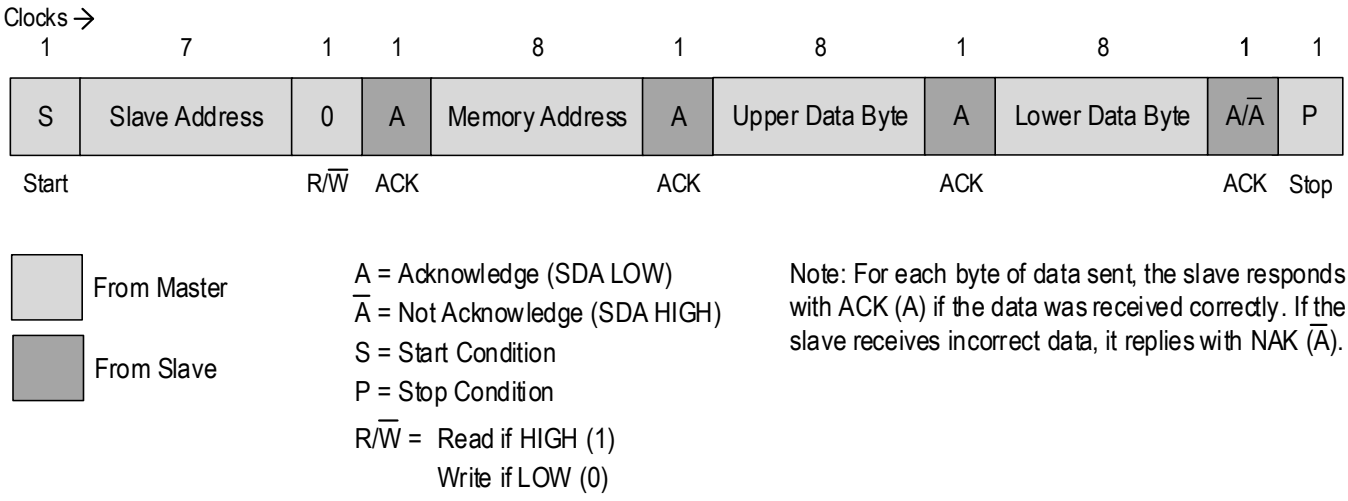


Figure 24. I2C Data WRITE to Slave(s)

Table 51. I2C Single WRITE to Memory

Byte Number	Direction	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0
Start Condition	Write								
1: Slave Address	Write	Slave Address: A6 to A3 (programmable)				Slave Address A2 to A0 (reserved)			DIR: Write=0
Acknowledge	Read								
2: Memory Address	Write	1	1	Memory Address in Selected Slave					
Acknowledge	Read								
3: Data Upper 8 bits	Write	Data Bits D10 to D3 to be Written to Selected Memory Address							
Acknowledge	Read								
4: Data Lower 3 bits	Write	D2 to D0 to Memory Address			1	1	1	1	1
Acknowledge	Read								
Stop Condition	Write								

[a] This mode can be applied to all registers and memory cells.

[b] For direct programming of NVM memory cells, only this mode can be used.

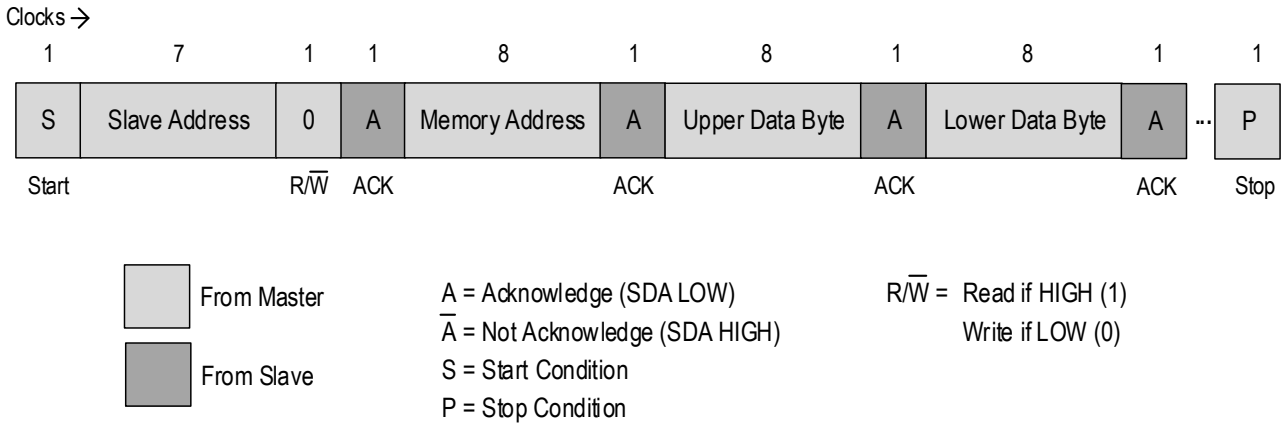


Figure 25. I2C Data BURST WRITE to Slave(s)

The burst write mode is used for writing data to multiple (= n) consecutive memory addresses. It can be applied to all registers and memory cells, except for direct writing to NVM memory. For direct programming of NVM memory cells, the single memory cell WRITE command sequence must be used, see Table 51

Table 52. I2C Burst WRITE to Memory

Byte Number	Direction	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0
Start Condition	Write								
1: Slave Address	Write	Slave Address: A6 to A3 (programmable)				Slave Address A2 to A0 (reserved)			DIR: Write=0
Acknowledge	Read								
2: Memory Address	Write	1	1	Memory Address in Selected Slave					
Acknowledge	Read								
3: Data Upper 8 Bits	Write	Data Bits D10 to D3 to be Written to Selected Memory Address							
Acknowledge	Read								
4: Data Lower 3 Bits	Write	D2 to D0 to Memory Address			1	1	1	1	1
Acknowledge	Read								
5: Data Upper 8 Bits	Write	Data Bits D10 to D3 Write to the Selected Memory Address + 1…n							
Acknowledge	Read								
6: Data Lower 3 Bits	Write	D2 to D0 to Memory Address +2…n			1	1	1	1	1
Acknowledge	Read								
…	…								
Stop Condition	Write								

[a] Tan shading indicates optional further write operations to additional incremental memory addresses.

For a write operation to random addresses, the data frame includes a combined format where a different slave address may be selected after sending a repeated start (Sr) bit. A repeated start bit has the same composition as a regular start bit (falling edge on SDA with SCL = high), but it is sent before a stop bit (rising edge on SDA with SCL = high); see Figure 18 for further details.

Note: This mode is used to write data to multiple (= n) consecutive memory addresses. It can be applied to all registers and memory cells.

4.4.4. READ Operations

For a read access with random memory addresses, the data frame includes a combined format where the data flow direction changes from writing to a register to reading from a register. This data flow requires a repeated start (Sr) bit as shown in Figure 26 and detailed in Table 53. A repeated start bit has the same composition as a regular start bit (falling edge on SDA with SCL = high), but it is sent before a stop bit (rising edge on SDA with SCL = high); see Figure 18 for further details. Data read without repeated start is not supported.

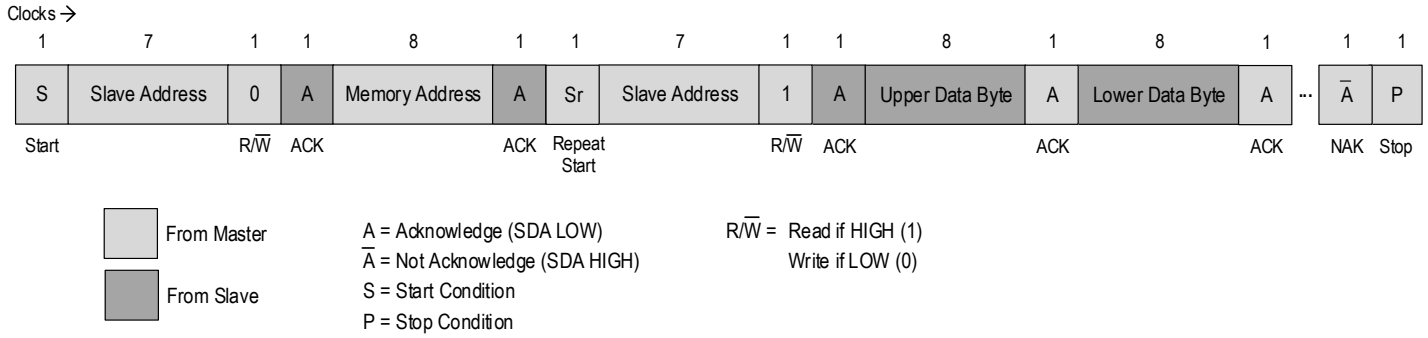


Figure 26. I2C Data READ from Slave(s) with Repeated START

Note: This mode is used for reading data from multiple (= n) consecutive memory addresses. It can be applied to all registers and memory cells.

Table 53. I2C Burst READ from Memory with Repeated Start

Byte Number	Direction	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0
Start Condition	Write								
1: Slave Address	Write	Slave Address: A6 to A3 (programmable)				Slave Address A2 to A0 (reserved)			DIR: Write = 0
Acknowledge	Read								
2: Memory Address	Write	1	1	Memory Address in Selected Slave					
Acknowledge	Read								
(Repeated) Start	Write								
3: Slave Address	Write	Slave Address: A6 to A3 (programmable)				Slave Address A2 to A0 (reserved)			DIR: Read = 1
Acknowledge	Read								
4: Data Upper 8 Bits	Read	Data Bits D10 to D3 Read from the Selected Memory Address							
Acknowledge	Write								
5: Data Lower 3 Bits	Read	D2 to D0 from Memory Address			1	1	1	1	1
Acknowledge	Write								
6: Data Upper 8 Bits	Read	Data Bits D10 to D3 Read from the Selected Memory Address + 1…n							
Acknowledge	Write								
7: Data Lower 3 Bits	Read	D2 to D0 from Memory Address +1…n			1	1	1	1	1
Acknowledge	Write								
…	…								
Not Acknowledge (NAK)	Write								
Stop Condition	Write								

[a] Tan shading indicates optional further write operations to additional incremental memory addresses

4.4.5. I2C Integrity Check

The Data Integrity Check is an extension of the I2C protocol for fail safe communication between I2C master and slave. It can be enabled by NVM configuration.

The I2C slave supports an even-parity check of the memory address in the field <MemAddr> and an even-parity check of the received first byte <SlaveAddr>. If the check fails, the 6-bit memory address counter is not updated, the slave receiver does not acknowledge, and the consecutive WRITE action is suppressed. Consequently a diagnostic status alert “protocol_integrity_fail” is asserted.

- PSA ParitySlaveAddress byte, even parity bit of preceding received first byte (slave address and R/nW)
- PMA ParityMemoryAddress; even parity bit of (D5 down to D0)

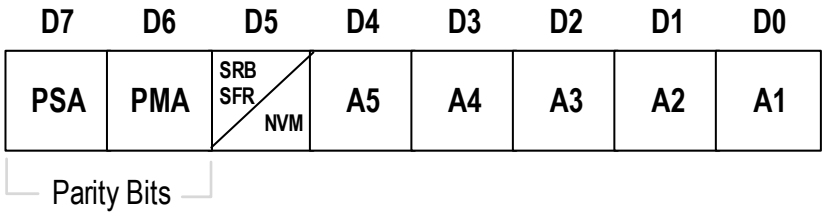


Figure 27. Memory Address of Selected I2C Slave with Integrity Check

Furthermore, a 3-bit CRC field is defined in the 16-bit field <CmdData> (see Figure 23) to identify transmission errors in each 16-bit data word. If the transferred data does not pass CRC verification, no WRITE action is taken, the slave receiver does not acknowledge, and the diagnostic status alert “protocol_integrity_fail” is asserted.

The polynomial used to calculate the 3-bit CRC is $CRC(x) = x^3 + x^1 + x^0$

The master/slave transmitter uses the CRC-3 polynomial on the first 13-bits (D15 to D3), and adds 000_{BIN} extending the length to 16-bit.

- D15 to D5 = <CmdData> Memory Information
- D4 to D3 = <Reserved> (Always all 1’s)
- D2 to D0 = 000_{BIN}

It generates the CRC code to send. The receiver then takes the entire 16-bit data word, including the CRC itself, and uses the same polynomial to confirm integrity of the data by verifying the CRC.

The CRC checksum for I2C belongs to the group of cyclic binary codes. The characteristic of cyclic binary code is that each control bit can be considered as a parity bit of a certain group of message bits, which depends on the generator polynomial.

It follows for the control bit calculation (transmitter):

$D_2 = XOR (D_{13}, D_{12}, D_{11}, D_9, D_6, D_5, D_4)$

$D_1 = XOR (D_{15}, D_{12}, D_{11}, D_{10}, D_8, D_5, D_4, D_3)$

$D_0 = XOR (D_{14}, D_{13}, D_{12}, D_{10}, D_7, D_6, D_5, D_3)$

It follows for the CRC verification (receiver):

$XOR (D_{13}, D_{12}, D_{11}, D_9, D_6, D_5, D_4, D_2) = 0$

$XOR (D_{15}, D_{12}, D_{11}, D_{10}, D_8, D_5, D_4, D_3, D_1) = 0$

$XOR (D_{14}, D_{13}, D_{12}, D_{10}, D_7, D_6, D_5, D_3, D_0) = 0$

The data field structure is as follows:

Data Byte N: D7 to D0 <CmdData (11 down to 3)>

Data Byte N+1: D7 to D5 <CmdData (2 down to 0)>

D4 to D3 <Reserved> (Always all 1's)

D2 to D0 <CRC>

Table 54. I2C Slave Address Byte and Memory Address Field Integrity Check

Protocol Integrity Check	D7	D6	D5 to D0	Diagnostic	Description
0_{BIN}	0_{BIN}	0_{BIN}	0bxxxxxx	Asserted	No action, no slave acknowledge
0_{BIN}	0_{BIN}	1_{BIN}	0bxxxxxx	Asserted	No action, no slave acknowledge
0_{BIN}	1_{BIN}	0_{BIN}	0bxxxxxx	Asserted	No action, no slave acknowledge
0_{BIN}	1_{BIN}	1_{BIN}	0bxxxxxx	Idle	Read/write action
1_{BIN}	PSA_{pass}	PSM_{pass}	0bxxxxxx	Idle	Parity check pass, read/write action
1_{BIN}	PSA_{pass}	PSM_{fail}	0bxxxxxx	Asserted	Parity check fail, no action, no slave acknowledge
1_{BIN}	PSA_{fail}	PSM_{pass}	0bxxxxxx	Asserted	Parity check fail, no action, no slave acknowledge
1_{BIN}	PSA_{fail}	PSM_{fail}	0bxxxxxx	Asserted	Parity check fail, no action, no slave acknowledge

4.4.6. I2C with Address Selection (Programming Option)

In this mode, the third pin (#1) is used for selecting the I2C slave address by hardware.

The default I2C slave address is 24dec (18hex) if address in is 'high' and 16dec (10 hex) if the address pin is 'low'.

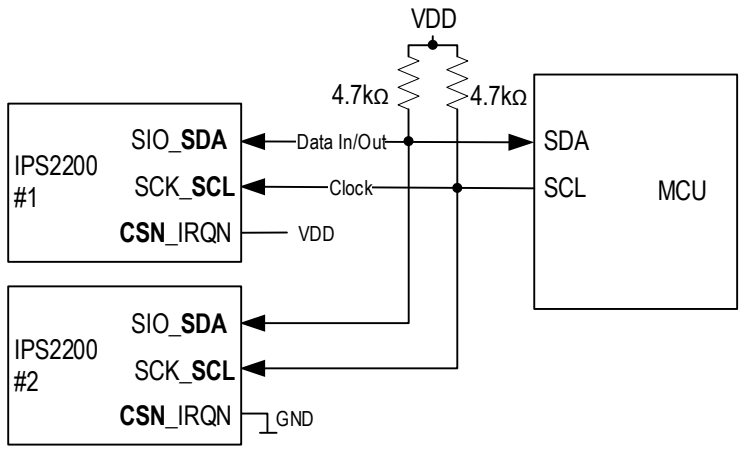


Figure 28. I2C Interface with Address Select

4.4.7. I2C Interface with Interrupt (Programming Option)

This is a standard I2C interface. The I2C address is programmable. In addition, a third pin (IRQN) is used as an interrupt output for fast signaling of a diagnostic event.

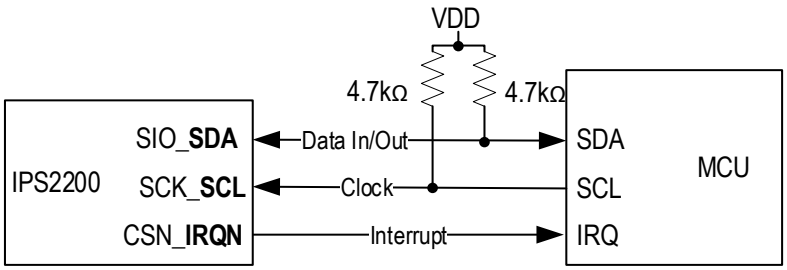
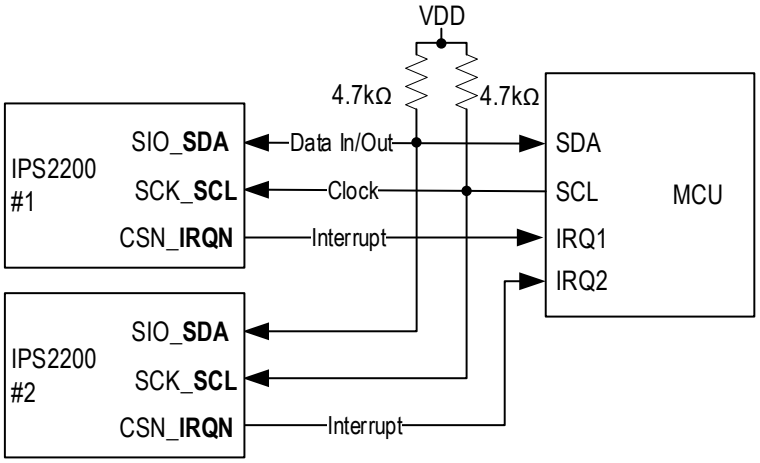


Figure 29. I2C Interface Configuration with Interrupt on a Single Slave



Note: In this mode, several I2C slaves are connected in parallel, all of them must have an individual I2C address.

Figure 30. I2C Interface Configuration with Multi-slave Interrupt

5. Glossary

Term	Description
I2C	Inter-Integrated Circuit; serial two-wire data bus
LSB	Least Significant Bit
MSB	Most Significant Bit
MSN	Most Significant Nibble
NVM	Nonvolatile Memory
POR	Power-On Reset

6. Revision History

Revision Date	Description of Change
Jun.28.23	Minor update to CRC polynomial for I2C.
May.11.23	Minor update to memory architecture description.
Oct.25.22	System configuration values updated.
Sep.07.22	Polynomial for CRC corrected
Mar.28.22	I2C defaults added.
Sep.07.21	Default mode of digital communication updated.
Sep.16.20	Register default value updated.
Jun.24.20	Transmitter sections updated.
Mar.30.20	Initial release.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Disclaimer Rev.5.0-1)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.