Date: Mar. 13, 2026

RENESAS TECHNICAL UPDATE

TOYOSU FORESIA, 3-2-24, Toyosu, Koto-ku, Tokyo 135-0061, Japan
Renesas Electronics Corporation

Product Category	MPU/MCU		Document No.	TN-RZ*-A0153A/E	Rev.	1.00
Title	Restrictions for DDRSS, Ethernet MAC and A-Format, and change for electrical characteristic		Information Category	Technical Notification		
Applicable Product	RZ/T2H and RZ/N2H Groups	Lot No. All	Reference Document	RZ/T2H and RZ/N2H Groups User's Manual: Hardware Rev.1.10 (R01UH1039EJ0110)		

This document informs HW issues / restrictions for DDRSS register access, Ethernet MAC and A-Format, and addition / change for electrical characteristic. These are reflected in User's Manual: Hardware Rev.1.20.

1. Target product

Product	Part Number	Package	Cortex-A55	Cortex-R52	Security
RZ/T2H	R9A09G077M48GBG#AC1 R9A09G077M48GBG#BC1	729-pin FCBGA	Quad cores	Two CPUs	Available
	R9A09G077M28GBG#AC1 R9A09G077M28GBG#BC1	729-pin FCBGA	Dual cores	Two CPUs	Available
	R9A09G077M08GBG#AC1 R9A09G077M08GBG#BC1	729-pin FCBGA	Single core	Two CPUs	Available
	R9A09G077M44GBG#AC1 R9A09G077M44GBG#BC1	729-pin FCBGA	Quad cores	Two CPUs	Not Available
	R9A09G077M24GBG#AC1 R9A09G077M24GBG#BC1	729-pin FCBGA	Dual cores	Two CPUs	Not Available
	R9A09G077M04GBG#AC1 R9A09G077M04GBG#BC1	729-pin FCBGA	Single core	Two CPUs	Not Available
RZ/N2H	R9A09G087M48GBG#AC1 R9A09G087M48GBG#BC1	576-pin FCBGA	Quad cores	Two CPUs	Available
	R9A09G087M28GBG#AC1 R9A09G087M28GBG#BC1	576-pin FCBGA	Dual cores	Two CPUs	Available
	R9A09G087M08GBG#AC1 R9A09G087M08GBG#BC1	576-pin FCBGA	Single core	Two CPUs	Available
	R9A09G087M44GBG#AC1 R9A09G087M44GBG#BC1	576-pin FCBGA	Quad cores	Two CPUs	Not Available
	R9A09G087M24GBG#AC1 R9A09G087M24GBG#BC1	576-pin FCBGA	Dual cores	Two CPUs	Not Available
	R9A09G087M04GBG#AC1 R9A09G087M04GBG#BC1	576-pin FCBGA	Single core	Two CPUs	Not Available

2. DDRSS register access

Issue	Workaround
When the DDRAPB bit in the SSTPCR4 register is set to 0 and the DDRSS is in the reset state which means MRCTLM24 = 1 or MRCTLM25 = 1 in the MRCTLM register, the DDRSS register access causes a hang-up.	The DDRAPB bit must be set to 1 when the DDRSS is in the reset state. In this case, read data is always 0. After the DDRSS is released from reset which means MRCTLM24 = 0 and MRCTLM25 = 0, the DDRAPB bit can be set to 0 for DDRSS register access permission.

3. Ethernet MAC (GMAC)

Issues	Workarounds
When Credit Based Shaper (CBS) is enabled for a Transmit Queue (TXQ), the packet that is available in a TXQ is scheduled for transmission when the TXQ accumulates zero or positive credit. When the current credit is negative or when a packet is available in the Transmit Queue and is waiting for the opportunity to be scheduled for transmission, the MAC increments the credit. Equally, during the actual packet transfer, the MAC decrements the credit. The rate at which the MAC increments or decrements the credit depends on the percentage of the total bandwidth that is reserved for the TXQ. As per the IEEE 802.1Qav standard, the MAC must decrement the credit also when the packet overheads are transmitted. These overheads include the preamble bytes before the start of packet, CRC/FCS bytes, and the minimum 12-byte Inter Packet Gap (IPG) after the end of the packet data transfer. However, the MAC decrements the credit only up to the last byte of the packet data (that is, the last byte of the Frame Check Sequence (FCS) transfer) and increments the credit during the subsequent nominal IPG period that is associated with that packet.	You must compute the additional/extra bandwidth that the TXQ consumes due to the defective algorithm. For this, you must use the below formula and example, substituting the values with the average length of the packets that are transmitted for the use case. Then, you must program the parameters that determine the fractional bandwidth for this TXQ to reflect the lesser percentage, such that the effective programmed fractional bandwidth is closer to the desired one. Formula: Additional/Extra bandwidth = ((# of packets × 12 bytes) / (Total number of bytes that is transmitted in that window, including the preamble bytes of each packet)) x Fractional bandwidth that is programmed for the TXQ. Example: If you program a 30% bandwidth for a TXQ which transfers a stream of 100 packets of 128 bytes length, the additional consumed bandwidth is equal to (30% × (100 × 12) / (100 × (8 + 128))) ≈ 2.65%. Effectively, this TXQ then receives 32.65% of total bandwidth instead of the programmed 30%.
When the Extended Gate Control feature is enabled, the software programs the Cycle Time Register (CTR) to a value that exceeds either of the following: • The sum of Time Intervals (TIs) on the Gate Control List (GCL) rows (that is, the extended gate controls). • The subsequent reprogramming of the number of GCL rows (LLR) and/or TIs. During the time interval between the end of the last row of the GCL and the value of the CTR (that is, the extended gate controls time interval), the MAC opens the gate controls for all the Transmit Queues (TxQs). Any eligible packet that is in the transmit queues during this interval is scheduled based on the priority of the TxQs. When the extended gate controls interval ends, the MAC executes the next iteration of the GCL. However, if the difference between the extended gate controls interval and the time that is required to transmit data packets with an individual total size (including the transmission overheads) of 16384 bytes, is less than nine periods of the PTP clock, the next or the running Base Time Register (BTR) value is additionally updated after the first row of next iteration of the GCL, which is incorrect. As a result, the next iteration of the GCL is incorrectly truncated after the first row, as well as any alternate iteration of the GCL.	The software must split the extended gate controls interval into the following: • An additional GCL row, and increase the LLR value by one. • The remaining extended gate controls interval. The additional GCL row must be programmed with all the gates open and with a TI that: • Is at least nine periods of the PTP clock. • Allows the remaining extended gate controls interval to be less than the time that is required to transmit data packets whose individual total size (including the transmission overheads) is 16384 bytes. For example, in 1-Gbps speed mode, if the software wants to program the extended gate controls interval to a value of 131074 ns, a value that is greater by 2 ns than the time that is required to transmit data packets with an individual total size of 16384 bytes (that is, 131072 ns). For a PTP clock of 125 MHz, the software can program an additional GCL row of 72 ns (that is, nine periods of the PTP clock). This way, the resulting extended gate controls interval is 131002 ns (131074 ns – 72 ns), which is less than 131072 ns.

4. A-Format (AFMT)

Restriction
Supported protocol version is Ver.2.0 only. Ver.1.x is not supported since serial data transmission timing was changed.

5. Electrical Characteristics

Items in red are added or changed.

Table 58.13 Ethernet PHY reference clock output timing

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
ETHn_REFCLK cycle time	tCK	—	40	—	—	ns
ETHn_REFCLK frequency	—	—	25.00 ± 50 ppm			MHz
		EtherCAT in use	**25.00 ± 25 ppm**			**MHz**
ETHn_REFCLK duty	—	—	45	—	55	%
		*1	**35**	**—**	**65**	**%**
ETHn_REFCLK rising/falling time	tCKr / tCKf	—	0.5	—	4.0	ns
RMIIn_REFCLK cycle time	tCK	—	20	—	—	ns
RMIIn_REFCLK frequency	—	—	50.00 ± 50 ppm			MHz
RMIIn_REFCLK duty	—	—	45	—	55	%
RMIIn_REFCLK rising/falling time	tCKr / tCKf	—	0.5	—	3.5	ns

Note 1. When main clock oscillator is used as reference clock input (by connecting a resonator) and SCKCR.PHYSEL is set to 1 (main clock oscillator is selected as reference clock output).

Table 58.33 xSPI timing

Conditions:

V_{OH} = VDD1833 × 0.5, V_{OL} = VDD1833 × 0.5, C = 15 pF (VDD1833 = 1.8 V or 3.3 V)

Parameter		Symbol	1.8 V Min.	1.8 V Max.	3.3 V Min.	3.3 V Max.	Unit	Reference figure
Cycle time	SDR	t_{PERIOD}	7.5	—	10.0	—	ns	Figure 58.56
	DDR		7.5	—	10.0	—	ns	
Clock output slew rate		t_{SRck}	0.75/0.56[*2]	—	0.56	—	V/ns	
Clock duty cycle distortion		t_{CKDCD}	0.0	t_{PERIOD} × 0.05	0.0	t_{PERIOD} × 0.05	ns	
Clock minimum pulse width		t_{CKMPW}	t_{PERIOD} × 0.45	—	t_{PERIOD} × 0.45	—	ns	
Differential clock crossing voltage		$V_{OX(AC)}$	0.4 × VCC18	0.6 × VCC18	—	—	V	
DS duty cycle distortion		t_{DSDCD}	0.0	t_{PERIOD} × 0.04	0.0	t_{PERIOD} × 0.04	ns	
DS minimum pulse width		t_{DSMPW}	t_{PERIOD} × 0.41	—	t_{PERIOD} × 0.41	—	ns	
Data input/output slew rate		t_{SR}	0.75/0.56[*2]	—	0.56	—	V/ns	
Data input setup time (to CK)	SDR	t_{SU}	2.0	—	2.0	—	ns	Figure 58.57
Data input hold time (to CK)		t_H	1.0	—	1.0	—	ns	
Data output delay time		t_{OD}	—	1.0[*3]	—	2.0[*3]	ns	
Data output hold time		t_{OH}	-1.0	—	-2.0	—	ns	
Data output buffer off time		t_{BOFF}	-1.0	—	-2.0	—	ns	
Data input setup time (to DS)	DDR[*1] [*3]	t_{SU}	-0.4/-0.6[*2]	—	-0.3	—	ns	Figure 58.58, Figure 58.59
Data input hold time (to DS)		t_H	t_{PERIOD} × 0.41 0.4/0.6[*2]	—	t_{PERIOD} × 0.41 - 0.3	—	ns	
Data output setup time (to CK)		t_{SUO}	0.8/1.0[*2]	—	1.0	—	ns	
Data output hold time (to CK)		t_{HO}	0.8/1.0[*2]	—	1.0	—	ns	
CS low to clock high		t_{CSLCKH}	6.0/8.0[*2] [*4]	—	8.0[*4]	—	ns	Figure 58.57 to Figure 58.59
Clock low to CS high		t_{CKLCSH}	6.0/8.0[*2]	—	8.0	—	ns	
CS high time		t_{CSTD}	1	16	1	16	t_{PERIOD}	
DS low to CS high		t_{DSLCSH}	6.0/8.0[*2] [*5]	—	10.6[*5]	—	ns	Figure 58.60
CS high to DS tri-state		t_{CSHDST}	0.0	t_{PERIOD}	0.0	t_{PERIOD}	ns	
CS low to DS low[*8]		t_{CSLDSL}	0.0	**16.0[*9]**	0.0	**20.0[*9]**	ns	
DS tri-state to CS low		t_{DSTCSL}	0.0	—	0.0	—	ns	
CK low to DS low[*6]		**t_{CKLDSL}**	**—**	**(0.45 + e) × t_{PERIOD} − 2 [*7]**	**—**	**(0.45 + e) × t_{PERIOD} − 2 [*7]**	**ns**	

Note 1. The DS shift setting (WRAPCFG.DSSFTCSx[4:0]) is 01001b for 133 MHz and 01100b for 100 MHz.

Note 2. Specification at 133 MHz / Specification at 100 MHz

Note 3. These are the values when the OEN assertion is extended in the Output Enable Asserting extension bit (COMCFG.OEASTEX = 1).

Note 4. These are the values when the CS assertion is extended in the CS asserting extension bit (LIOCFGCSn.CSASTEX = 1).

Note 5. These are the values when the t_{CKLDSL} constraint is satisfied.

Note 6. This constraint is necessary only to satisfy the t_{DSLCSH} requirement in JESD251, which specifies that t_{DSLCSH} must be at least 80% of t_{PERIOD}. Set LIOCFGCSn.CSNEGEX to the appropriate value to ensure the memory specification complies with this constraint.

Note 7. e = LIOCFGCSn.CSNEGEX

Note 8. If the DS is high during the command & modifier phase when using JESD251 Profile 2.0 memory, the time from CS low to DS high must also meet this specification.

Note 9. When using JESD251 Profile 1.0 memory or JESD251 Profile 2.0 memory with LIOCFGCSn.LATEMD set to 0, this constraint does not apply if the internal pull-down resistor of the DS pin is enabled.

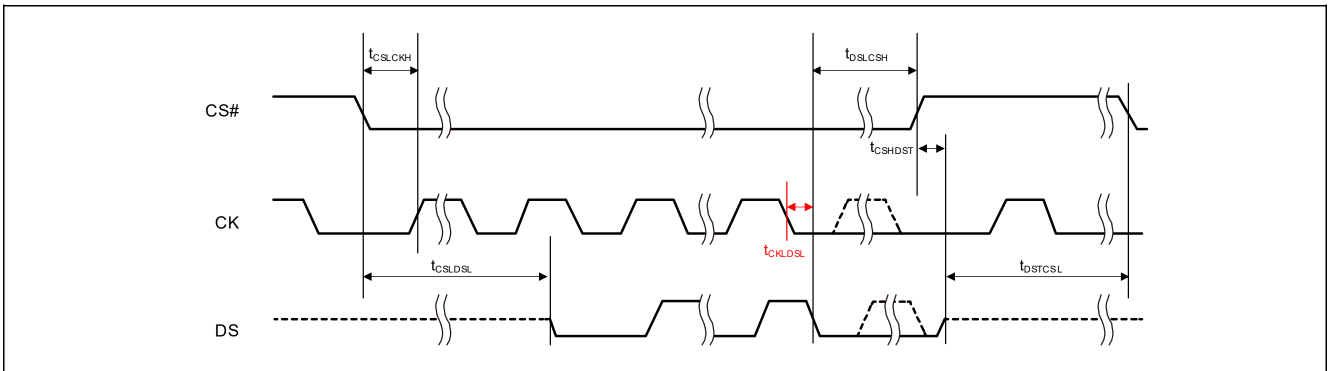

Figure 58.60 DS to CS signal timing

Table 58.39 1.8 V SDHI timing

Conditions: V_{OH} = VDD1833 × 0.5, V_{OL} = VDD1833 × 0.5 (VDD1833 = 1.8 V)

Parameter		Symbol	Conditions	Min.	Max.	Unit	Reference figure
(SD) SDR104 (eMMC) HS200	SD_CLK clock cycle	TSDCYC	C = 15 pF Drive Strength CLK: Ultra-high Others: High	5		ns	Figure 58.72
	SD_CLK clock high level width	TSDWH		1.5	—	ns	
	SD_CLK clock low level width	TSDWL		1.5	—	ns	
	SD_CLK clock rise time	TSDLH		—	1	ns	
	SD_CLK clock fall time	TSDHL		—	1	ns	
	SD_CMD, SD_DATA output delay	TSDODLY		−1.7	0.9	ns	
	SD_CMD, SD_DATA input set up time	TSDIS		—	—	ns	
	SD_CMD, SD_DATA input hold time	TSDIH		—	—	ns	
	SD_CMD, SD_DATA input data width	TSDIDW		2.88	—	ns	
(SD) SDR50, SDR25, SDR12 (eMMC) High Speed SDR, Backwards Compatibility	SD_CLK clock cycle	TSDCYC	C = 20 pF Drive Strength CLK: Ultra-high Others: High	10	—	ns	Figure 58.72
	SD_CLK clock high level width	TSDWH		3	—	ns	
	SD_CLK clock low level width	TSDWL		3	—	ns	
	SD_CLK clock rise time	TSDLH		—	2	ns	
	SD_CLK clock fall time	TSDHL		—	2	ns	
	SD_CMD, SD_DATA output delay	TSDODLY		−4.2	1.6	ns	
	SD_CMD, SD_DATA input set up time	TSDIS		1.1	—	ns	
	SD_CMD, SD_DATA input hold time	TSDIH		1.8	—	ns	
	SD_CMD, SD_DATA input data width	TSDIDW		—	—	ns	
(SD) DDR50 (eMMC) High Speed DDR	SD_CLK clock cycle	TSDCYC	C = 25 pF Drive Strength CLK: High Others: High	20	—	ns	Figure 58.73
	SD_CLK clock high level width	TSDWH		9	11	ns	
	SD_CLK clock low level width	TSDWL		9	11	ns	
	SD_CLK clock rise time	TSDLH		—	3	ns	
	SD_CLK clock fall time	TSDHL		—	3	ns	
	SD_CMD output delay (SDR)	TSDODLY		−6	3	ns	
	SD_CMD input set up time (SDR)	TSDIS		4.8	—	ns	
	SD_CMD input hold time (SDR)	TSDIH		2.5	—	ns	
	SD_DATA output delay (DDR)	TSDODLY_DDR		2.5	6	ns	
	SD_DATA input set up time (DDR)	TSDIS_DDR		1.5	—	ns	
	SD_DATA input hold time (DDR)	TSDIH_DDR		1.5	—	ns	

RENESAS

Table 58.40 3.3 V SDHI timing

Conditions: V_{OH} = VDD1833 × 0.5, V_{OL} = VDD1833 × 0.5 (VDD1833 = 3.3 V)

Parameter		Symbol	Conditions	Min.	Max.	Unit	Reference figure
(SD) High Speed (eMMC) High Speed SDR,	SD_CLK clock cycle	TSDCYC	C = 40 pF Drive Strength CLK: High Others: Middle	20	—	ns	Figure 58.72
	SD_CLK clock high level width	TSDWH		7	—	ns	
	SD_CLK clock low level width	TSDWL		7	—	ns	
	SD_CLK clock rise time	TSDLH		—	3	ns	
	SD_CLK clock fall time	TSDHL		—	3	ns	
	SD_CMD, SD_DATA output delay	TSDODLY		−6.2	2.5	ns	
	SD_CMD, SD_DATA input set up time	TSDIS		4	—	ns	
	SD_CMD, SD_DATA input hold time	TSDIH		2	—	ns	
	SD_CMD, SD_DATA input data width	TSDIDW		—	—	ns	
(SD) Default Speed (eMMC) Backwards Compatibility	SD_CLK clock cycle	TSDCYC	C = 40 pF Drive Strength CLK: High Others: Middle	40	—	ns	Figure 58.72
	SD_CLK clock high level width	TSDWH		10	—	ns	
	SD_CLK clock low level width	TSDWL		10	—	ns	
	SD_CLK clock rise time	TSDLH		—	10	ns	
	SD_CLK clock fall time	TSDHL		—	10	ns	
	SD_CMD, SD_DATA output delay	TSDODLY		−7.5	2.5	ns	
	SD_CMD, SD_DATA input set up time	TSDIS		4	—	ns	
	SD_CMD, SD_DATA input hold time	TSDIH		2	—	ns	
	SD_CMD, SD_DATA input data width	TSDIDW		—	—	ns	
(eMMC) High Speed DDR	SD_CLK clock cycle	TSDCYC	C = 30 pF Drive Strength CLK: High Others: High	20	—	ns	Figure 58.73
	SD_CLK clock high level width	TSDWH		9	11	ns	
	SD_CLK clock low level width	TSDWL		9	11	ns	
	SD_CLK clock rise time	TSDLH		—	3	ns	
	SD_CLK clock fall time	TSDHL		—	3	ns	
	SD_CMD output delay (SDR)	TSDODLY		−6	6	ns	
	SD_CMD input set up time (SDR)	TSDIS		4.8	—	ns	
	SD_CMD input hold time (SDR)	TSDIH		2.5	—	ns	
	SD_DATA output delay (DDR)	TSDODLY_DDR		2.5	6.5	ns	
	SD_DATA input set up time (DDR)	TSDIS_DDR		1.7	—	ns	
	SD_DATA input hold time (DDR)	TSDIH_DDR		1.5	—	ns	