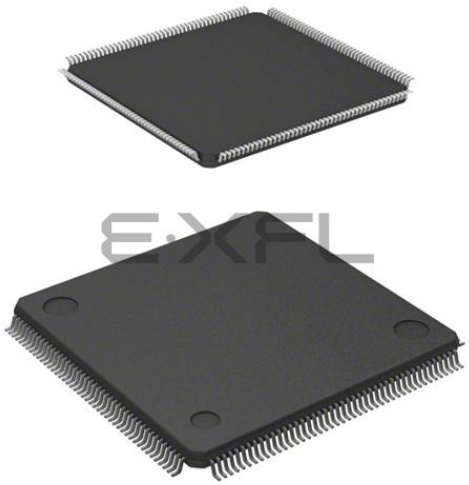
Renesas Electronics America Inc - R5F72165ADFA#V1 Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	SH2A-FPU
Core Size	32-Bit Single-Core
Speed	200MHz
Connectivity	CANbus, Ethernet, I²C, SCI, SPI, USB
Peripherals	DMA, PWM, WDT
Number of I/O	112
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	176-LQFP
Supplier Device Package	176-LFQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f72165adfa-v1

2.5.8 Floating-Point Operation Instructions

Table 2.18 Floating-Point Operation Instructions

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
FABS	FRn	1111nnnn01011101	|FRn| → FRn	1	—	Yes	Yes	
FABS	DRn	1111nnn001011101	|DRn| → DRn	1	—		Yes	
FADD	FRm, FRn	1111nnnnmmmm0000	FRn + FRm → FRn	1	—	Yes	Yes	
FADD	DRm, DRn	1111nnn0mmm00000	DRn + DRm → DRn	6	—		Yes	
FCMP/EQ	FRm, FRn	1111nnnnmmmm0100	(FRn = FRm)? 1:0 → T	1	Comparison result	Yes	Yes	
FCMP/EQ	DRm, DRn	1111nnn0mmm00100	(DRn = DRm)? 1:0 → T	2	Comparison result		Yes	
FCMP/GT	FRm, FRn	1111nnnnmmmm0101	(FRn > FRm)? 1:0 → T	1	Compa-rison result	Yes	Yes	
FCMP/GT	DRm, DRn	1111nnn0mmm00101	(DRn > DRm)? 1:0 → T	2	Comparison result		Yes	
FCNVDS	DRm, FPUL	1111mmm010111101	(float) DRm → FPUL	2	—		Yes	
FCNVSD	FPUL, DRn	1111nnn010101101	(double) FPUL → DRn	2	—		Yes	
FDIV	FRm, FRn	1111nnnnmmmm0011	FRn/FRm → FRn	10	—	Yes	Yes	
FDIV	DRm, DRn	1111nnn0mmm00011	DRn/DRm → DRn	23	—		Yes	
FLDI0	FRn	1111nnnn10001101	0 × 00000000 → FRn	1	—	Yes	Yes	
FLDI1	FRn	1111nnnn10011101	0 × 3F800000 → FRn	1	—	Yes	Yes	
FLDS	FRm, FPUL	1111mmmm00011101	FRm → FPUL	1	—	Yes	Yes	
FLOAT	FPUL,FRn	1111nnnn00101101	(float)FPUL → FRn	1	—	Yes	Yes	
FLOAT	FPUL,DRn	1111nnn000101101	(double)FPUL → DRn	2	—		Yes	
FMAC	FR0,FRm,FRn	1111nnnnmmmm1110	FR0 × FRm+FRn → FRn	1	—	Yes	Yes	
FMOV	FRm, FRn	1111nnnnmmmm1100	FRm → FRn	1	—	Yes	Yes	
FMOV	DRm, DRn	1111nnn0mmm01100	DRm → DRn	2	—		Yes	

6.4.6 On-Chip Peripheral Module Interrupts

On-chip peripheral module interrupts are generated by the following on-chip peripheral modules:

- A/D converter (ADC)
- Controller area network (RCAN-ET)
- Direct memory access controller (DMAC)
- Compare match timer (CMT)
- Bus state controller (BSC)
- Watchdog timer (WDT)
- Ethernet Controller (EtherC, E-DMAC)
- USB function module (USB)
- Multi-function timer pulse unit 2 (MTU2)
- Multi-function timer pulse unit 2S (MTU2S)
- Port output enable 2 (POE2)
- I^2C bus interface 3 (IIC3)
- Renesas serial peripheral interface (RSPI)
- Serial communication interface (SCI)
- Serial communication interface with FIFO (SCIF)

As every source is assigned a different interrupt vector, the source does not need to be identified in the exception service routine. A priority level in a range from 0 to 15 can be set for each module by interrupt priority registers 05 to 19 (IPR05 to IPR19). The on-chip peripheral module interrupt exception handling sets the I3 to I0 bits in SR to the priority level of the accepted on-chip peripheral module interrupt.

8.2.1 DTC Mode Register A (MRA)

MRA selects DTC operating mode. MRA cannot be accessed directly by the CPU.

Bit:	7	6	5	4	3	2	1	0
	MD[1:0]		Sz[1:0]		SM[1:0]		-	-
Initial value:	*	*	*	*	*	*	*	*
R/W:	-	-	-	-	-	-	-	-

*: Undefined

Bit	Bit Name	Initial Value	R/W	Description
7, 6	MD[1:0]	Undefined	—	DTC Mode 1 and 0 Specify DTC transfer mode. 00: Normal mode 01: Repeat mode 10: Block transfer mode 11: Setting prohibited
5, 4	Sz[1:0]	Undefined	—	DTC Data Transfer Size 1 and 0 Specify the size of data to be transferred. 00: Byte-size transfer 01: Word-size transfer 10: Longword-size transfer 11: Setting prohibited
3, 2	SM[1:0]	Undefined	—	Source Address Mode 1 and 0 Specify an SAR operation after a data transfer. 0x: SAR is fixed (SAR write-back is skipped) 10: SAR is incremented after a transfer (by 1 when Sz1 and Sz0 = B'00; by 2 when Sz1 and Sz0 = B'01; by 4 when Sz1 and Sz0 = B'10) 11: SAR is decremented after a transfer (by 1 when Sz1 and Sz0 = B'00; by 2 when Sz1 and Sz0 = B'01; by 4 when Sz1 and Sz0 = B'10)

Bit	Bit Name	Initial Value	R/W	Description
0	ERR	0	R/(W)*	Transfer Stop Flag Indicates that a DTC address error or NMI interrupt has occurred. If a DTC address error or NMI interrupt occurs while the DTC is active, a DTC address error handling or NMI interrupt handling processing is executed after the DTC has released the bus mastership. The DTC halts after a data transfer or a transfer information writing state depending on the NMI input timing. Note that a writing state is not exact, when the DTC halts after a data transfer. When the data is transferred, set a transfer information once again (except that a read skip is performed). 0: No interrupt has occurred 1: An interrupt has occurred [Clearing condition] • When writing 0 after reading 1

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

8.2.9 DTC Vector Base Register (DTCVBR)

DTCVBR is a 32-bit register that specifies the base address for vector table address calculation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12		All 0	R/W	Bits 11 to 0 are always read as 0. The write value should always be 0.
11 to 0	—	All 0	R	

9.4.3 CSn Space Wait Control Register (CSnWCR) (n = 0 to 7)

CSnWCR specifies various wait cycles for memory access. The bit configuration of this register varies as shown below according to the memory type (TYPE2 to TYPE0) specified by the CSn space bus control register (CSnBCR). Specify CSnWCR before accessing the target area. Specify CSnBCR first, then specify CSnWCR.

CSnWCR is initialized to H'00000500 by a power-on reset and retains the value by a manual reset and in software standby mode.

(1) Normal Space, SRAM with Byte Selection, MPX-I/O

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]				WM	-	-	-	-	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	— *	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

(9) Relationship between Refresh Requests and Bus Cycles

If a refresh request occurs during bus cycle execution, the refresh cycle must wait for the bus cycle to be completed. If a refresh request occurs while the bus is released by the bus arbitration function, the refresh will not be executed until the bus mastership is acquired. This LSI has the $\overline{\text{REFOUT}}$ pin to request the bus while waiting for refresh execution. For $\overline{\text{REFOUT}}$ pin function selection, see section 22, Pin Function Controller (PFC). This LSI continues to assert $\overline{\text{REFOUT}}$ (low level) until the bus is acquired.

On receiving the asserted $\overline{\text{REFOUT}}$ signal, the external device must negate the $\overline{\text{BREQ}}$ signal and return the bus. If the external bus does not return the bus for a period longer than the specified refresh interval, refresh cannot be executed and the SDRAM contents may be lost.

If a new refresh request occurs while waiting for the previous refresh request, the previous refresh request is deleted. To refresh correctly, a bus cycle longer than the refresh interval or the bus mastership occupation must be prevented from occurring.

If a bus mastership is requested during self-refresh, the bus will not be released until the refresh is completed.

Bit	Bit Name	Initial Value	R/W	Description
0	I1AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1A pin in the TGRA_2 input capture conditions. 0: Does not include the TIOC1A pin in the TGRA_2 input capture conditions 1: Includes the TIOC1A pin in the TGRA_2 input capture conditions

11.3.9 Timer Synchronous Clear Register S (TSYCRS)

TSYCRS is an 8-bit readable/writable register that specifies conditions for clearing TCNT_3 and TCNT_4 in the MTU2S in synchronization with the MTU2. The MTU2S has one TSYCRS in channel 3 but the MTU2 has no TSYCRS.

Bit:	7	6	5	4	3	2	1	0
	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CE0A	0	R/W	Clear Enable 0A Enables or disables counter clearing when the TGFA flag of TSR_0 in the MTU2 is set. 0: Disables counter clearing by the TGFA flag in TSR_0 1: Enables counter clearing by the TGFA flag in TSR_0
6	CE0B	0	R/W	Clear Enable 0B Enables or disables counter clearing when the TGFB flag of TSR_0 in the MTU2 is set. 0: Disables counter clearing by the TGFB flag in TSR_0 1: Enables counter clearing by the TGFB flag in TSR_0

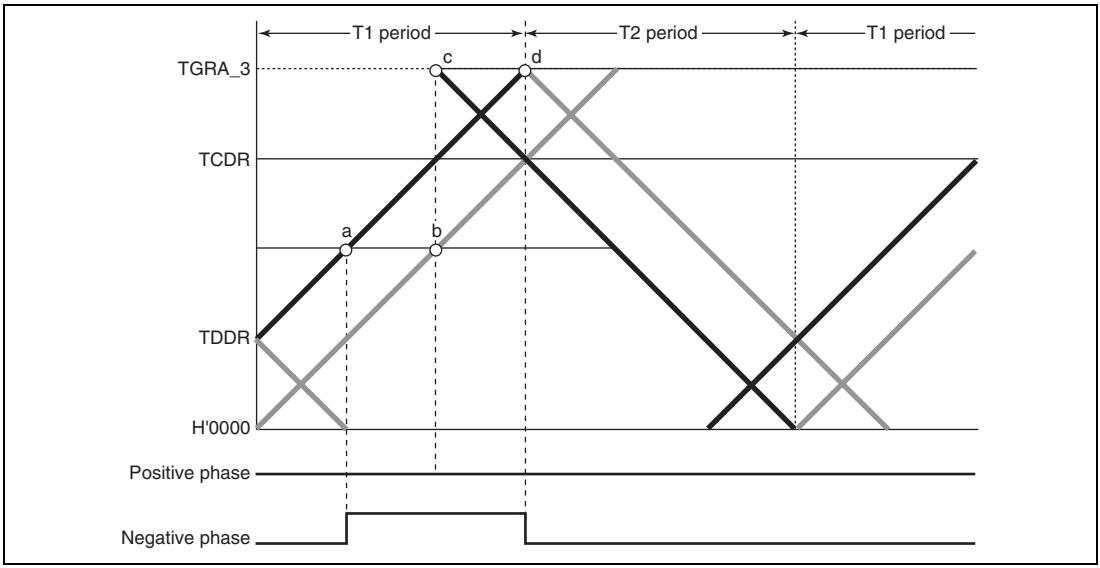


Figure 11.51 Example of Complementary PWM Mode 0% and 100% Waveform Output (3)

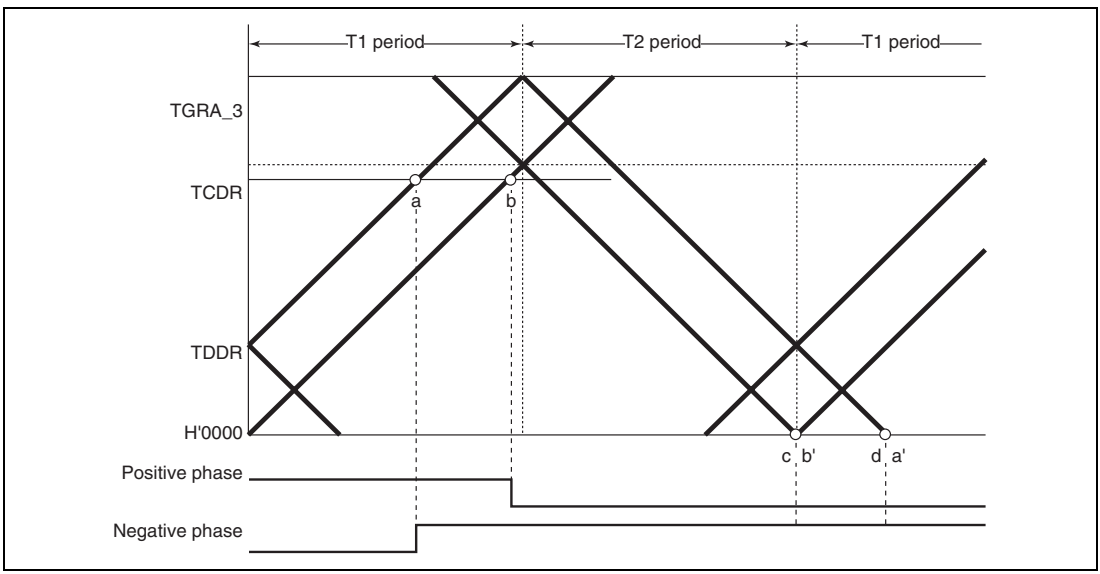


Figure 11.52 Example of Complementary PWM Mode 0% and 100% Waveform Output (4)

17.4.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIF are independent, so full duplex communication is possible. The transmitter and receiver are 16-byte FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmitting and receiving.

Figure 17.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIF monitors the line and starts serial communication when the line goes to the space (low) state, indicating a start bit. One serial character consists of a start bit (low), data (LSB first), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIF synchronizes at the falling edge of the start bit. The SCIF samples each data bit on the eighth pulse of a clock with a frequency 16 times the bit rate. Receive data is latched at the center of each bit.

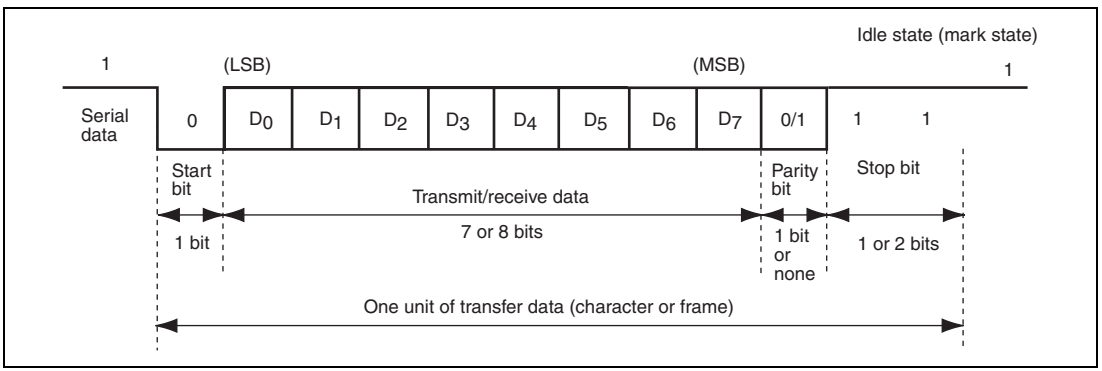


Figure 17.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

18.3.8 RSPI Bit Rate Register (SPBR)

SPBR sets the bit rate in master mode. SPBR can be read from or written to by the CPU. If the contents of SPBR are changed by the CPU while the MSTR and SPE bits in the RSPI control register (SPCR) are 1 with the RSPI function in master mode enabled, operation cannot be guaranteed. When the RSPI is used in slave mode, the bit rate depends on the input clock regardless of the settings of SPBR and BRDV.

Bit:	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The bit rate is determined by combinations of SPBR settings and the bit settings in the BRDV1 and BRDV0 bits in the RSPI command registers (SPCMD0 to SPCMD7). The equation for calculating the bit rate is given below. In the equation, N denotes an SPBR setting (0, 1, 2, …, 255), and n denotes bit settings in the bits BRDV1 and BRDV0 (0, 1, 2, 3).

$$\text{Bit rate} = \frac{f\,(P\phi)}{2 \times (N + 1) \times 2^n}$$

Table 18.3 shows examples of the relationship between the SPBR register and BRDV1 and BRDV0 bit settings.

Table 18.3 Relationship between SPBR and BRDV[1:0] Settings

SPBR (N)	BRDV[1:0] (n)	Division Ratio	Bit Rate				
			Pϕ = 16 MHz	Pϕ = 20 MHz	Pϕ = 32 MHz	Pϕ = 40 MHz	Pϕ = 50 MHz
0	0	2	8.0 Mbps	10.0 Mbps	—	—	—
1	0	4	4.0 Mbps	5.0 Mbps	8.0 Mbps	10.0 Mbps	12.5 Mbps
2	0	6	2.67 Mbps	3.3 Mbps	5.33 Mbps	6.67 Mbps	8.33 Mbps
3	0	8	2.0 Mbps	2.5 Mbps	4.0 Mbps	5.0 Mbps	6.25 Mbps
4	0	10	1.6 Mbps	2.0 Mbps	3.2 Mbps	4.0 Mbps	5.00 Mbps
5	0	12	1.33 Mbps	1.67 Mbps	2.67 Mbps	3.33 Mbps	4.17 Mbps
5	1	24	667 kbps	833 kbps	1.33 Mbps	1.67 Mbps	2.08 Mbps

- Port B Control Register L2 (PBCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PB7MD[2:0]			-	PB6MD[2:0]			-	PB5MD[2:0]			-	PB4MD[2:0]		
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 during the on-chip ROM disabled external extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	PB7MD[2:0]	000*	R/W	PB7 Mode Select the function of the PB7/A23/TEND0/IRQ7/TCLKC/SCK4/RD/$\overline{WR}$ pin. 000: PB7 I/O (port) 001: A23 output (BSC) 010: TEND0 output (DMAC) 011: IRQ7 input (INTC) 100: TCLKC input (MTU2) 101: Setting prohibited 110: SCK4 I/O (SCI) 111: RD/$\overline{WR}$ output (BSC)
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	PB6MD[2:0]	000*	R/W	PB6 Mode Select the function of the PB6/A22/$\overline{WAIT}$/IRQ6/TCLKD/TXD0 pin. 000: PB6 I/O (port) 001: A22 output (BSC) 010: $\overline{WAIT}$ input (BSC) 011: IRQ6 input (INTC) 100: TCLKD input (MTU2) 101: Setting prohibited 110: TXD0 output (SCI) 111: Setting prohibited

Section 24 USB Function Module (USB)

This LSI incorporates a USB function module (USB).

24.1 Features

- Automatic processing of USB protocol with on-chip protocol processor and transceiver conforming to USB2.0

  Automatic processing of USB standard commands for endpoint 0 (some commands and class/vendor commands require decoding and processing by firmware)
- Transfer speed: Full-speed (12 Mbps)
- Endpoint configuration

Endpoint Name	Abbreviation	Transfer Type	Maximum Packet Size	FIFO Buffer Capacity (Byte)	DMA/DTC Transfer
Endpoint 0	EP0s	Setup	8	8	—
	EP0i	Control IN	16	16	—
	EP0o	Control OUT	16	16	—
Endpoint 1	EP1	Bulk OUT	64	128	Possible
Endpoint 2	EP2	Bulk IN	64	128	Possible
Endpoint 3	EP3	Interrupt IN	16	16	—
Endpoint 4	EP4	Bulk OUT	64	128	Possible
Endpoint 5	EP5	Bulk IN	64	128	Possible
Endpoint 6	EP6	Interrupt IN	16	16	—
Endpoint 7	EP7	Bulk OUT	64	64	—
Endpoint 8	EP8	Bulk IN	64	64	—
Endpoint 9	EP9	Interrupt IN	16	16	—

24.3.37 USB Trigger Register 1 (USBTRG1)

USBTRG1 is a write-only register that generates one-shot triggers to control the transmit/receive sequence for each endpoint. The read value of this register is undefined. Do not write a value to this register using the read value, such as a bit manipulation instruction.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	EP3 PKTE	EP2 PKTE	EP1 RDFN
Initial value:	0	0	0	0	0	0	0	0
R/W:	-	-	-	-	-	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	—	Reserved The write value should always be 0.
2	EP3PKTE	0	W	EP3 Packet Enable After one packet of data has been written to the endpoint 3 transmit FIFO buffer, the transmit data is fixed by writing 1 to this bit.
1	EP2PKTE	0	W	EP2 Packet Enable After one packet of data has been written to the endpoint 2 transmit FIFO buffer, the transmit data is fixed by writing 1 to this bit.
0	EP1RDFN	0	W	EP1 Read Complete Write 1 to this bit after one packet of data has been read from the endpoint 1 receive FIFO buffer. The endpoint 1 receive FIFO buffer has a dual-FIFO configuration. Writing 1 to this bit initializes the FIFO that was read, enabling the next packet to be received.

26.2.16 Receive Descriptor Fetch Address Register (RDFAR)

RDFAR stores the descriptor start address required when the E-DMAC fetches descriptor information from the receive descriptor. Which receive descriptor information is used for processing by the E-DMAC can be recognized by monitoring the addresses indicated by this register. The address that the E-DMAC is actually accessing during the descriptor fetch processing is not always equal to the value read from this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDFA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDFA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDFA[31:0]	All 0	R	Receive Descriptor Fetch Address These bits can only be read. Writing is prohibited.

30.5 Usage Notes

30.5.1 Current Consumption during Oscillation Settling Time

While waiting for clock oscillation to settle, the current consumption is increased.

30.5.2 Notes on Writing to Registers

When writing to a register related to power-down modes by the CPU, after the CPU executes the write instruction, it then executes the subsequent instruction without waiting for the actual writing process to the register to finish.

To update the change made by writing to a register while executing the subsequent instruction, perform a dummy read to the same register between the instruction to write to the register and the subsequent instruction.

30.5.3 Notes on Canceling Software Standby Mode with an IRQx Interrupt Request

When canceling software standby mode using an IRQx interrupt request, change the IRQ sense select setting of ICRx in a state in which no IRQx interrupt requests are generated and clear the IRQxF flag in IRQRRx to 0 by the automatic clearing function of the IRQx interrupt processing.

If the IRQxF flag in the IRQ interrupt request register x (IRQRRx) is 1, changing the setting of the IRQ sense select bits in the interrupt control register x (ICRx) or clearing the IRQxF flag in IRQRRx to 0 will clear the relevant IRQx interrupt request but will not clear the software standby cancellation request.

Module Name	Register Name	Abbreviation	Number of Bits	Address	Access Size
MTU2S	Timer start register_5S	TSTR_5S	8	H'FFFE48B4	8
	Timer counter U_5S	TCNTU_5S	16	H'FFFE4880	16, 32
	Timer counter V_5S	TCNTV_5S	16	H'FFFE4890	16, 32
	Timer counter W_5S	TCNTW_5S	16	H'FFFE48A0	16, 32
	Timer general register U_5S	TGRU_5S	16	H'FFFE4882	16
	Timer general register V_5S	TGRV_5S	16	H'FFFE4892	16
	Timer general register W_5S	TGRW_5S	16	H'FFFE48A2	16
	Timer compare match clear register S	TCNTCMPCLRS	8	H'FFFE48B6	8
	Timer start register S	TSTRS	8	H'FFFE4A80	8, 16
	Timer synchronous register S	TSYRS	8	H'FFFE4A81	8
	Timer read/write enable register S	TRWERS	8	H'FFFE4A84	8
	Timer output master enable register S	TOERS	8	H'FFFE4A0A	8
	Timer output control register 1S	TOCR1S	8	H'FFFE4A0E	8, 16
	Timer output control register 2S	TOCR2S	8	H'FFFE4A0F	8
	Timer gate control register S	TGCRS	8	H'FFFE4A0D	8
	Timer cycle data register S	TCDRS	16	H'FFFE4A14	16, 32
	Timer dead time data register S	TDDRS	16	H'FFFE4A16	16
	Timer subcounter S	TCNTSS	16	H'FFFE4A20	16, 32
	Timer cycle buffer register S	TCBRS	16	H'FFFE4A22	16
	Timer interrupt skipping set register S	TITCRS	8	H'FFFE4A30	8, 16
	Timer interrupt skipping counter S	TITCNTS	8	H'FFFE4A31	8
	Timer buffer transfer set register S	TBTERS	8	H'FFFE4A32	8
	Timer dead time enable register S	TDERS	8	H'FFFE4A34	8
	Timer synchronous clear register S	TSYCRS	8	H'FFFE4A50	8
	Timer waveform control register S	TWCRS	8	H'FFFE4A60	8
	Timer output level buffer register S	TOLBRS	8	H'FFFE4A36	8
POE2	Input level control/status register 1	ICSR1	16	H'FFFE5000	16
	Output level control/status register 1	OCSR1	16	H'FFFE5002	16
	Input level control/status register 2	ICSR2	16	H'FFFE5004	16
	Output level control/status register 2	OCSR2	16	H'FFFE5006	16
	Input level control/status register 3	ICSR3	16	H'FFFE5008	16
	Software port output enable register	SPOER	8	H'FFFE500A	8

33.3.5 DMAC Module Timing

Table 33.9 DMAC Module Timing

Conditions: $V_{CC}Q = PLLV_{CC} = DrV_{CC}$= 3.0 to 3.6 V, AV_{CC} = AVREF = 4.5 to 5.5 V,
$V_{SS} = PLLV_{SS} = DrV_{SS}$= AVREFVSS = AV_{SS} = 0 V,
Ta = –40°C to +85°C (Industrial specifications)

Item	Symbol	Min.	Max.	Unit	Figure
DREQ setup time	t_{DRQS}	20	—	ns	Figure 33.38
DREQ hold time	t_{DRQH}	20	—		
DACK, TEND delay time	t_{DACD}	—	20		Figure 33.39

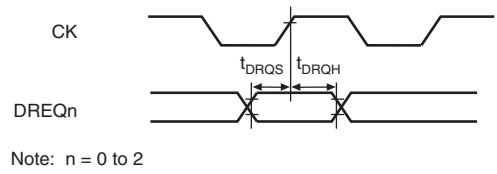


Figure 33.38 DREQ Input Timing

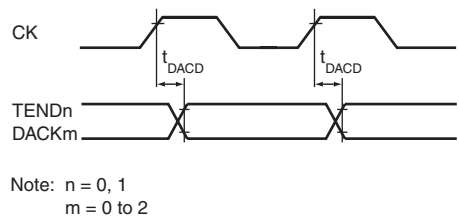


Figure 33.39 DACK, TEND Output Timing

33.3.14 IIC3 Module Timing

Table 33.18 I²C Bus Interface 3 Timing

Conditions: $V_{CC}Q = PLLV_{CC} = DrV_{CC} = 3.0$ to 3.6 V, $AV_{CC} = AVREF = 4.5$ to 5.5V,
$V_{SS} = PLLV_{SS} = DrV_{SS} = AVREFVSS = AV_{SS} = 0$ V,
Ta = –40°C to +85°C (Industrial specifications)

			Specifications				
Item	**Symbol**	**Test Conditions**	**Min.**	**Typ.**	**Max.**	**Unit**	**Figure**
SCL input cycle time	t_{SCL}		12 t_{pcyc} *1 + 600	—	—	ns	Figure 33.55
SCL input high pulse width	t_{SCLH}		3 t_{pcyc} *1 + 300	—	—	ns	
SCL input low pulse width	t_{SCLL}		5 t_{pcyc} *1 + 300	—	—	ns	
SCL, SDA input rise time	t_{Sr}		—	—	300	ns	
SCL, SDA input fall time	t_{Sf}		—	—	300	ns	
SCL, SDA input spike pulse removal time*2	t_{SP}		—	—	1 t_{pcyc} *1	ns	
SDA input bus free time	t_{BUF}		5	—	—	t_{pcyc} *1	
Start condition input hold time	t_{STAH}		3	—	—	t_{pcyc} *1	
Retransmit start condition input setup time	t_{STAS}		3	—	—	t_{pcyc} *1	
Stop condition input setup time	t_{STOS}		3	—	—	t_{pcyc} *1	
Data input setup time	t_{SDAS}		1 t_{pcyc} *1 + 20	—	—	ns	
Data input hold time	t_{SDAH}		0	—	—	ns	
SCL, SDA capacitive load	Cb		0	—	400	pF	
SCL, SDA output fall time*3	t_{Sf}		20 + 0.1 Cb	—	250	ns	

Notes: 1. t_{pcyc} indicates peripheral clock (Pϕ) cycle.
2. Depends on the value of NF2CYC.
3. Indicates the I/O buffer characteristic.