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	H8SX
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	EBI/EMI, I²C, IrDA, SCI, SmartCard, USB
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	92
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	40K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x8b
Oscillator Type	External
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-LFQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f61664mn50fpv

Classification	Pin Name	I/O	Description
16-bit timer pulse unit (TPU)	TIOCA5 TIOCB5	Input/ output	Signals for TGRA_5 and TGRB_5. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
	TCLKE TCLKF TCLKG TCLKH	Input	Input pins for external clock signals.
	TIOCA6 TIOCB6 TIOCC6 TIOCD6	Input/ output	Signals for TGRA_6 to TGRD_6. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
	TIOCA7 TIOCB7	Input/ output	Signals for TGRA_7 and TGRB_7. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
	TIOCA8 TIOCB8	Input/ output	Signals for TGRA_8 and TGRB_8. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
	TIOCA9 TIOCB9 TIOCC9 TIOCD9	Input/ output	Signals for TGRA_9 to TGRD_9. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
	TIOCA10 TIOCB10	Input/ output	Signals for TGRA_10 and TGRB_10. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
	TIOCA11 TIOCB11	Input/ output	Signals for TGRA_11 and TGRB_11. These pins are used as input capture inputs, output compare outputs, or PWM outputs.
Programmable pulse generator (PPG)	PO31 to PO0	Output	Output pins for the pulse signals.
8-bit timer (TMR)	TMO0 to TMO3	Output	Output pins for the compare match signals.
	TMCI0 to TMCI3	Input	Input pins for the external clock signals that drive for the counters.
	TMRI0 to TMRI3	Input	Input pins for the counter-reset signals.
Watchdog timer (WDT)	$\overline{\text{WDTOVF}}$	Output	Output pin for the counter-overflow signal in watchdog-timer mode.

Table 2.2 Combinations of Instructions and Addressing Modes (2)

			Addressing Mode							
Classification	Instruction	Size	@ERn	@(d,PC)	@(RnL.B/Rn.W/ERn.L, PC)	@aa:24	@aa:32	@@ aa:8	@@vec:7	—
Branch	BRA/BS, BRA/BC	—		O						
	BSR/BS, BSR/BC	—		O						
	Bcc	—		O						
	BRA	—		O	O					
	BRA/S	—		O*						
	JMP	—	O			O	O	O	O	
	BSR	—		O						
	JSR	—	O			O	O	O	O	
	RTS, RTS/L	—								O
System control	TRAPA	—								O
	RTE, RTE/L	—								O

[Legend]

d:d:8 or d:16

Note: * Only @(d:8, PC) is available.

9.11.11 Controlling Byte and Word Accesses

When 16-bit bus SDRAM is used, byte and word accesses are performed through the control of DQMLU and DQMLL.

Figures 9.69 and 9.70 show control timing examples of the DQM signals in the big endian format. Figure 9.71 shows a connection example when the DQM signals are used for the byte and word control.

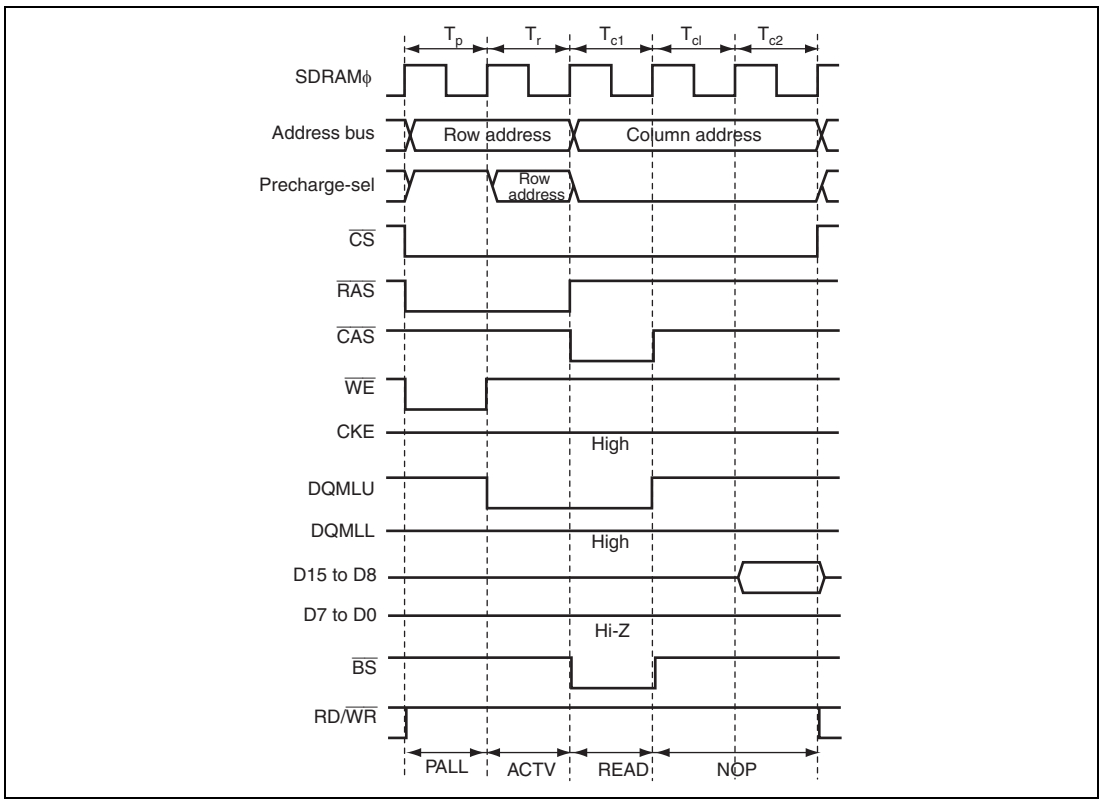


Figure 9.69 Control Timing Example of Byte Control by DQM in 16-Bit Access Space (Read Access with Lowest Bit of Address = B'0)

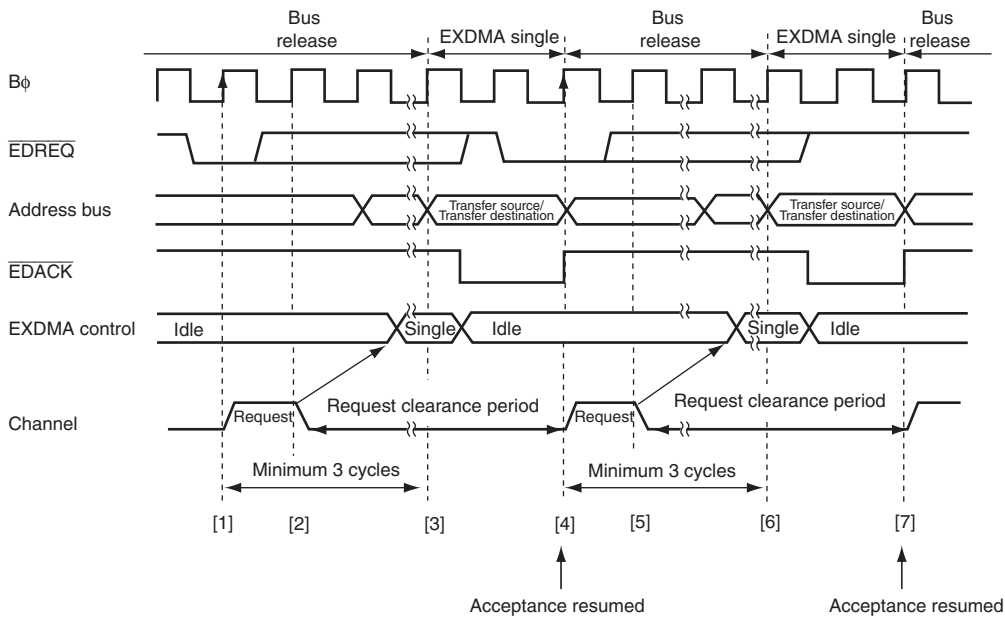
Bit	Bit Name	Initial Value	R/W	Description
30	DACKE	0	R/W	$\overline{\text{DACK}}$ Signal Output Enable Enables/disables the $\overline{\text{DACK}}$ signal output in single address mode. This bit is ignored in dual address mode. 0: Disables $\overline{\text{DACK}}$ signal output 1: Enables $\overline{\text{DACK}}$ signal output
29	TENDE	0	R/W	$\overline{\text{TEND}}$ Signal Output Enable Enables/disables the $\overline{\text{TEND}}$ signal output. 0: Disables $\overline{\text{TEND}}$ signal output 1: Enables $\overline{\text{TEND}}$ signal output
28	—	0	R/W	Reserved Initial value should not be changed.
27	DREQS	0	R/W	$\overline{\text{DREQ}}$ Select Selects whether a low level or the falling edge of the $\overline{\text{DREQ}}$ signal used in external request mode is detected. 0: Low level detection 1: Falling edge detection (the first transfer after a transfer enabled is detected on a low level)
26	NRD	0	R/W	Next Request Delay Selects the accepting timing of the next transfer request. 0: Starts accepting the next transfer request after completion of the current transfer 1: Starts accepting the next transfer request one cycle of Bϕ after completion of the current transfer
25, 24	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
23	ACT	0	R	Active State Indicates the operating state for the channel. 0: Waiting for a transfer request or a transfer disabled state by clearing the DTE bit to 0 1: Active state
22 to 20	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.

(3) $\overline{\text{EDREQ}}$ Pin Falling Edge Activation Timing

Figure 11.36 shows an example of single address mode transfer activated by the $\overline{\text{EDREQ}}$ pin falling edge.

$\overline{\text{EDREQ}}$ pin sampling is performed in each cycle starting at the next rise of Bϕ after the end of the DTE bit write cycle.

When a low level is sampled at the $\overline{\text{EDREQ}}$ pin while acceptance of a transfer request via the $\overline{\text{EDREQ}}$ pin is possible, the request is held within the EXDMAC. Then when activation is initiated within the EXDMAC, the request is cleared, and $\overline{\text{EDREQ}}$ pin high level sampling for edge sensing is started. If $\overline{\text{EDREQ}}$ pin high level sampling is completed by the end of the EXDMA single cycle, acceptance resumes after the end of the single cycle, and $\overline{\text{EDREQ}}$ pin low level sampling is performed again. This sequence of operations is repeated until the end of the transfer.



[1] Acceptance after transfer enabling; $\overline{\text{EDREQ}}$ pin low level is sampled at rise of Bϕ, and request is held.
[2], [5] Request is cleared at end of next bus cycle, and activation is started in EXDMAC.
[3], [6] EXDMA cycle starts; $\overline{\text{EDREQ}}$ pin high level sampling is started at rise of Bϕ.
[4], [7] When $\overline{\text{EDREQ}}$ pin high level has been sampled, acceptance is resumed after completion of write cycle.
(As in [1], $\overline{\text{EDREQ}}$ pin low level is sampled at rise of Bϕ, and request is held.)

Figure 11.36 Example of Single Address Mode Transfer Activated by $\overline{\text{EDREQ}}$ Pin Falling Edge

11.6 Operation in Cluster Transfer Mode

In cluster transfer mode, transfer is performed by the consecutive read and write operations of 1 to 32 bytes using the cluster buffer. A part of the cluster transfer mode function differs from the ordinary transfer mode functions (normal transfer, repeat transfer, and block transfer modes).

11.6.1 Address Mode

(1) Cluster Transfer Dual Address Mode (AMS = 0)

In this mode, both the transfer source and destination addresses are specified for transfer in the EXDMAC internal registers. The transfer source address is set in the source address register (EDSAR), and the transfer destination address is set in the destination address register (EDDAR).

The transfer is processed by performing the consecutive read of a cluster-size from the transfer source address and then the consecutive write of that data to the transfer destination address. One data access size to 32 bytes can be specified as a cluster size. When one data access size is specified as a cluster size, block transfer mode (dual address mode) is used.

The cycles in a cluster-size transfer are indivisible: another bus cycle (external access by another bus master, refresh cycle, or external bus release cycle) does not occur in a cluster-size transfer.

$\overline{\text{ETEND}}$ pin output can be enabled or disabled by means of the ETENDE bit in EDMDR. $\overline{\text{ETEND}}$ is output for the last write cycle. The $\overline{\text{EDACK}}$ signal is not output.

Figure 11.53 shows the data flow in the cluster transfer mode (dual address mode), figure 11.54 shows an example of the timing in cluster transfer dual address mode, and figure 11.55 shows the cluster transfer dual address mode operation.

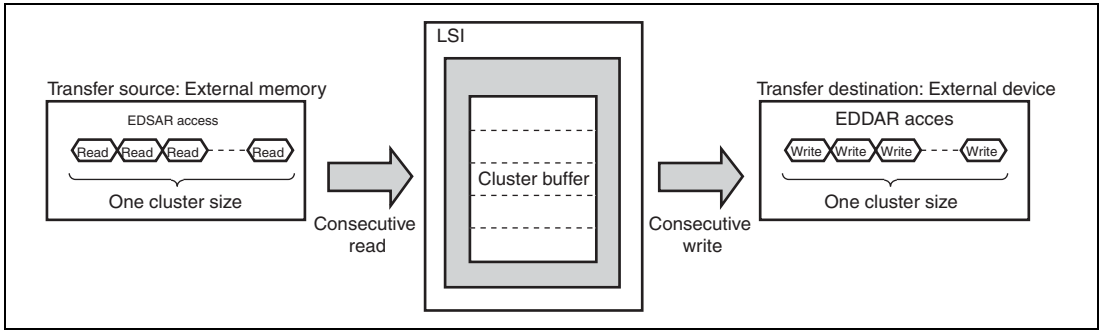


Figure 11.53 Data Flow in Cluster Transfer Dual Address Mode

Bit	Bit Name	Initial Value	R/W	Description
3	ITS11	0	R/W	$\overline{\text{IRQ11}}$ Pin Select Selects an input pin for $\overline{\text{IRQ11}}$. 0: Selects pin P23 as $\overline{\text{IRQ11}}$-A input 1: Selects pin P63 as $\overline{\text{IRQ11}}$-B input
2	ITS10	0	R/W	$\overline{\text{IRQ10}}$ Pin Select Selects an input pin for $\overline{\text{IRQ10}}$. 0: Selects pin P22 as $\overline{\text{IRQ10}}$-A input 1: Selects pin P62 as $\overline{\text{IRQ10}}$-B input
1	ITS9	0	R/W	$\overline{\text{IRQ9}}$ Pin Select Selects an input pin for $\overline{\text{IRQ9}}$. 0: Selects pin P21 as $\overline{\text{IRQ9}}$-A input 1: Selects pin P61 as $\overline{\text{IRQ9}}$-B input
0	ITS8	0	R/W	$\overline{\text{IRQ8}}$ Pin Select Selects an input pin for $\overline{\text{IRQ8}}$. 0: Selects pin P20 as $\overline{\text{IRQ8}}$-A input 1: Selects pin P60 as $\overline{\text{IRQ8}}$-B input

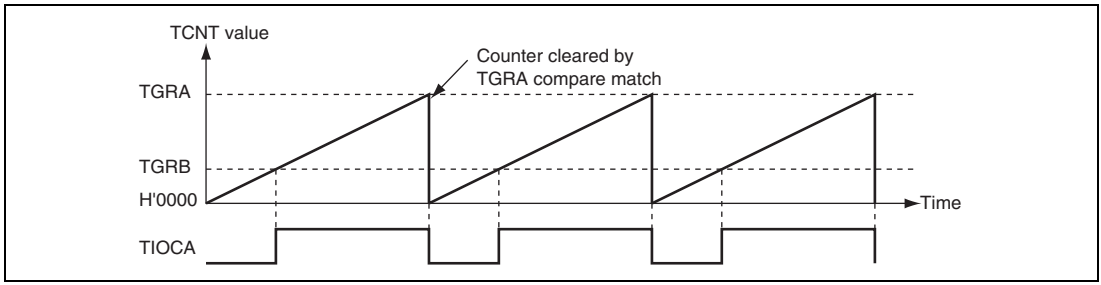


Figure 14.22 Example of PWM Mode Operation (1)

Figure 14.23 shows an example of PWM mode 2 operation.

In this example, synchronous operation is designated for channels 0 and 1, TGRB_1 compare match is set as the TCNT clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGR registers (TGRA_0 to TGRD_0, TGRA_1), to output a 5-phase PWM waveform.

In this case, the value set in TGRB_1 is used as the cycle, and the values set in the other TGRs as the duty cycle.

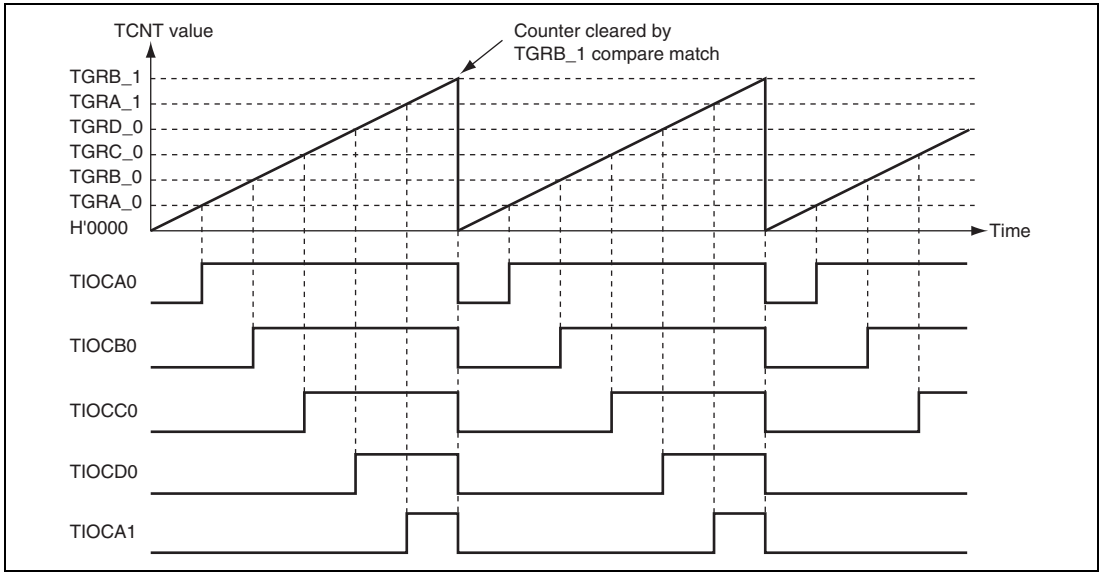
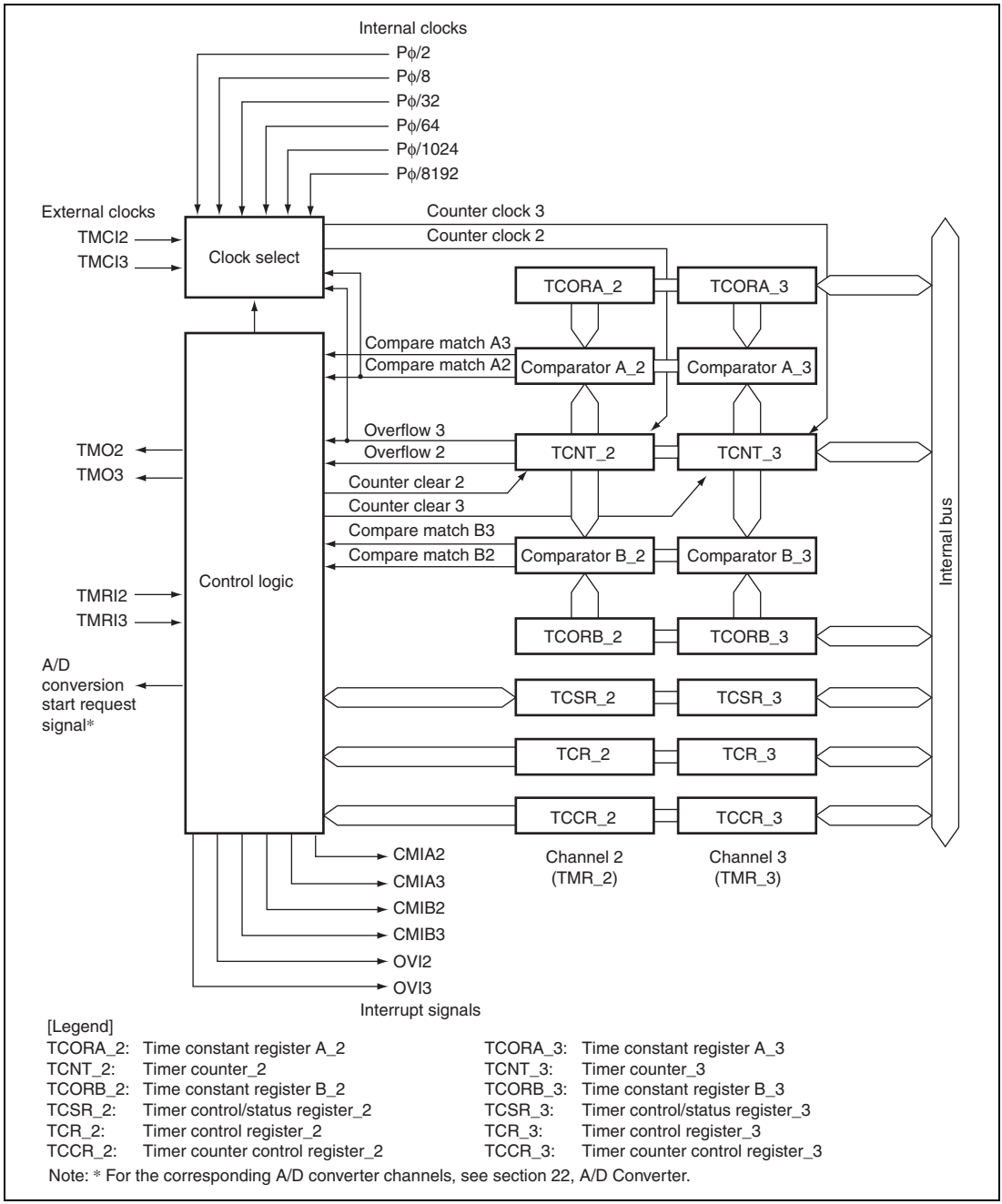


Figure 14.23 Example of PWM Mode Operation (2)

Bit	Bit Name	Initial Value	R/W	Description
3	G3NOV	0	R/W	Group 7 Non-Overlap Selects normal or non-overlapping operation for pulse output group 7. 0: Normal operation (output values updated by compare match A on the selected TPU channel) 1: Non-overlapping operation (output values updated by compare match A or B on the selected TPU channel)
2	G2NOV	0	R/W	Group 6 Non-Overlap Selects normal or non-overlapping operation for pulse output group 6. 0: Normal operation (output values updated by compare match A on the selected TPU channel) 1: Non-overlapping operation (output values updated by compare match A or B on the selected TPU channel)
1	G1NOV	0	R/W	Group 5 Non-Overlap Selects normal or non-overlapping operation for pulse output group 5. 0: Normal operation (output values updated by compare match A on the selected TPU channel) 1: Non-overlapping operation (output values updated by compare match A or B on the selected TPU channel)
0	G0NOV	0	R/W	Group 4 Non-Overlap Selects normal or non-overlapping operation for pulse output group 4. 0: Normal operation (output values updated by compare match A on the selected TPU channel) 1: Non-overlapping operation (output values updated by compare match A or B on the selected TPU channel)

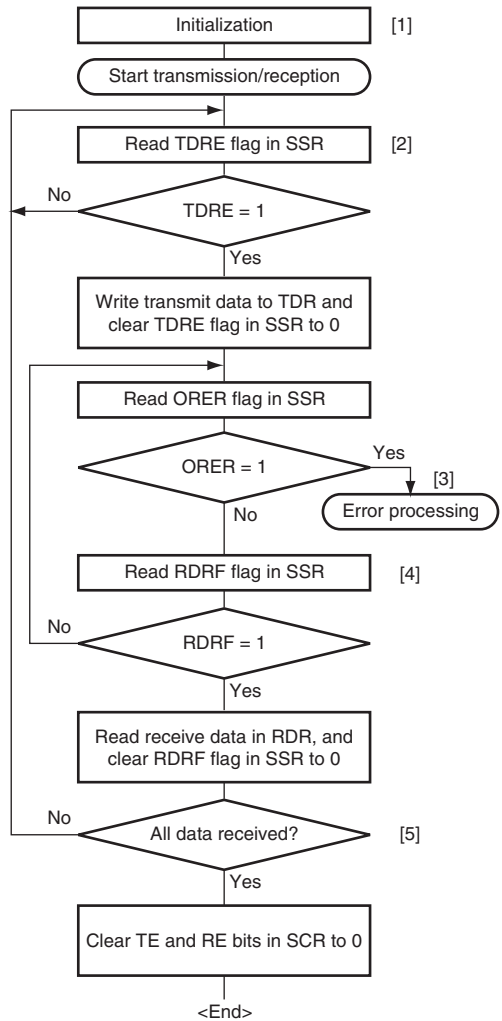


[Legend]

TCORA_2:	Time constant register A_2	TCORA_3:	Time constant register A_3
TCNT_2:	Timer counter_2	TCNT_3:	Timer counter_3
TCORB_2:	Time constant register B_2	TCORB_3:	Time constant register B_3
TCSR_2:	Timer control/status register_2	TCSR_3:	Timer control/status register_3
TCR_2:	Timer control register_2	TCR_3:	Timer control register_3
TCCR_2:	Timer counter control register_2	TCCR_3:	Timer counter control register_3

Note: * For the corresponding A/D converter channels, see section 22, A/D Converter.

Figure 16.2 Block Diagram of 8-Bit Timer Module (Unit 1)



Note: When switching from transmit or receive operation to simultaneous transmit and receive operations, first clear the TE bit and RE bit to 0, then set both these bits to 1 simultaneously.

[1] SCI initialization:
The TxD pin is designated as the transmit data output pin, and the RxD pin is designated as the receive data input pin, enabling simultaneous transmit and receive operations.

[2] SCI state check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0. Transition of the TDRE flag from 0 to 1 can also be identified by a TXI interrupt.

[3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag to 0. Reception cannot be resumed if the ORER flag is set to 1.

[4] SCI state check and receive data read:
Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR and clear the RDRF flag to 0. Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission/reception, before the MSB (bit 7) of the current frame is received, finish reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0. Also, before the MSB (bit 7) of the current frame is transmitted, read 1 from the TDRE flag to confirm that writing is possible. Then write data to TDR and clear the TDRE flag to 0.
However, the TDRE flag is checked and cleared automatically when the DMAC or DTC is initiated by a transmit data empty interrupt (TXI) request and writes data to TDR. Similarly, the RDRF flag is cleared automatically when the DMAC or DTC is initiated by a receive data full interrupt (RXI) and reads data from RDR.

Figure 19.23 Sample Flowchart of Simultaneous Serial Transmission and Reception

20.3.7 Interrupt Enable Register 0 (IER0)

IER0 enables the interrupt requests of interrupt flag register 0 (IFR0). When an interrupt flag is set to 1 while the corresponding bit of each interrupt is set to 1, an interrupt request is sent to the CPU. The interrupt vector number is determined by the contents of interrupt select register 0 (ISR0).

Bit	7	6	5	4	3	2	1	0
Bit Name	BRST	EP1 FULL	EP2 TR	EP2 EMPTY	SETUP TS	EP0o TS	EP0i TR	EP0i TS
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	BRST	0	R/W	Bus Reset
6	EP1 FULL	0	R/W	EP1 FIFO Full
5	EP2 TR	0	R/W	EP2 Transfer Request
4	EP2 EMPTY	0	R/W	EP2 FIFO Empty
3	SETUP TS	0	R/W	Setup Command Receive Complete
2	EP0o TS	0	R/W	EP0o Receive Complete
1	EP0i TR	0	R/W	EP0i Transfer Request
0	EP0i TS	0	R/W	EP0i Transmission Complete

Bit	Bit Name	Initial Value	R/W	Description
1	EP2DMAE	0	R/W	Endpoint 2 DMA Transfer Enable When this bit is set, DMA transfer is enabled from memory to the endpoint 2 transmit FIFO buffer. If there is at least one byte of open space in the FIFO buffer, a DMAC start interrupt signal (USBINTN1) is asserted. In DMA transfer, when 64 bytes are written to the FIFO buffer the EP2 packet enable bit is set automatically, allowing 64 bytes of data to be transferred, and if there is still space in the other side of the two FIFOs, the DMAC start interrupt signal (USBINTN1) is asserted again. However, if the size of the data packet to be transmitted is less than 64 bytes, the EP2 packet enable bit is not set automatically, and so should be set by the CPU with a DMA transfer end interrupt. As EP2-related interrupt requests to the CPU are not automatically masked, interrupt requests should be masked as necessary in the interrupt enable register. • Operating procedure 1. Write of 1 to the EP2 DMAE bit in DMAR 2. Set the DMAC to activate through USBINTN1 3. Transfer count setting in the DMAC 4. DMAC activation 5. DMA transfer 6. DMA transfer end interrupt generated See section 20.8.3, DMA Transfer for Endpoint 2.

21.3.1 I²C Bus Control Register A (ICCRA)

ICCRA enables or disables I²C bus interface, controls transmission or reception, and selects master or slave mode, transmission or reception, and transfer clock frequency in master mode.

Bit	7	6	5	4	3	2	1	0
Bit Name	ICE	RCVD	MST	TRS	CKS3	CKS2	CKS1	CKS0
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	ICE	0	R/W	I²C Bus Interface Enable 0: This module is halted 1: This bit is enabled for transfer operations (SCL and SDA pins are bus drive state)
6	RCVD	0	R/W	Reception Disable This bit enables or disables the next operation when TRS is 0 and ICDRR is read. 0: Enables next reception 1: Disables next reception
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	Transmit/Receive Select When arbitration is lost in master mode, MST and TRS are both reset by hardware, causing a transition to slave receive mode. Modification of the TRS bit should be made between transfer frames. Operating modes are described below according to MST and TRS combination. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
3	CKS3	0	R/W	Transfer Clock Select 3 to 0 These bits are valid only in master mode. Make setting according to the required transfer rate. For details on the transfer rate, see table 21.2.
2	CKS2	0	R/W	
1	CKS1	0	R/W	
0	CKS0	0	R/W	

21.4.2 Master Transmit Operation

In I²C bus format master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device return an acknowledge signal. Figures 21.5 and 21.6 show the operating timings in master transmit mode. The transmission procedure and operations in master transmit mode are described below.

1. Set the ICR bit in the corresponding register to 1. Set the ICE bit in ICCRA to 1. Set the WAIT bit in ICMR and the CKS3 to CKS0 bits in ICCRA to 1. (initial setting)
2. Read the BSSY flag in ICCRB to confirm that the bus is free. Set the MST and TRS bits in ICCRA to select master transmit mode. Then, write 1 to BBSY and 0 to SCP using the MOV instruction. (The start condition is issued.) This generates the start condition.
3. After confirming that TDRE in ICSR has been set, write the transmit data (the first byte shows the slave address and R/W) to ICDRT. After this, when TDRE is automatically cleared to 0, data is transferred from ICDRT to ICDRS. TDRE is set again.
4. When transmission of one byte data is completed while TDRE is 1, TEND in ICSR is set to 1 at the rising of the ninth transmit clock pulse. Read the ACKBR bit in ICIER to confirm that the slave device has been selected. Then, write the second byte data to ICDRT. When ACKBR is 1, the slave device has not been acknowledged, so issue a stop condition. To issue the stop condition, write 0 to BBSY and SCP using the MOV instruction. SCL is fixed to a low level until the transmit data is prepared or the stop condition is issued.
5. The transmit data after the second byte is written to ICDRT every time TDRE is set.
6. Write the number of bytes to be transmitted to ICDRT. Wait until TEND is set (the end of last byte data transmission) while TDRE is 1, or wait for NACK (NACKF in ICSR is 1) from the receive device while ACKE in ICIER is 1. Then, issue the stop condition to clear TEND or NACKF.
7. When the STOP bit in ICSR is set to 1, the operation returns to the slave receive mode.

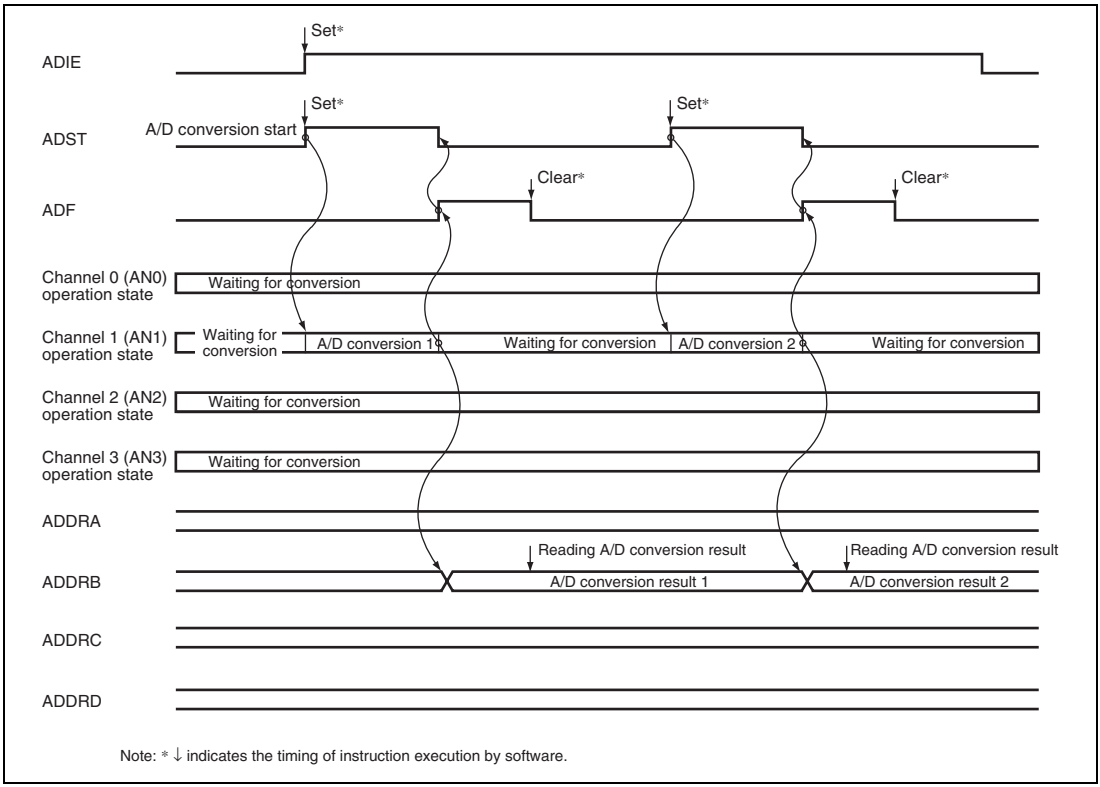


Figure 22.3 Example of A/D Converter Operation (Single Mode, Channel 1 Selected)

22.4.2 Scan Mode

In scan mode, A/D conversion is to be performed sequentially on the analog inputs of the specified channels up to four or eight*[1] channels. Two types of scan mode are provided, that is , continuous scan mode where A/D conversion is repeatedly performed and one-cycle scan mode where A/D conversion is performed for the specified channels for one cycle.

(1) Continuous Scan Mode

1. When the ADST bit in ADCSR is set to 1 by software, TPU*[1], TMR*[2], or an external trigger input, A/D conversion starts on the first channel in the specified channel group. Consecutive A/D conversion on a maximum of four channels (SCANE and SCANS = B'10) or on a maximum of eight channels*[1] (SCANE and SCANS = B'11) can be selected. When consecutive A/D conversion is performed on four channels, A/D conversion starts on AN0 when CH3 and CH2 of unit 0 = B’00, on AN4 when CH3 and CH2 of units 0 and 1 = B'01.

25.7.2 Programming/Erasing Interface Parameters

The programming/erasing interface parameters specify the operating frequency, storage place for program data, start address of programming destination, and erase block number, and exchanges the execution result. These parameters use the general registers of the CPU (ER0 and ER1) or the on-chip RAM area. The initial values of programming/erasing interface parameters are undefined at a reset or a transition to software standby mode.

Since registers of the CPU except for ER0 and ER1 are saved in the stack area during download of an on-chip program, initialization, programming, or erasing, allocate the stack area before performing these operations (the maximum stack size is 128 bytes). The return value of the processing result is written in R0. The programming/erasing interface parameters are used in download control, initialization before programming or erasing, programming, and erasing. Table 25.4 shows the usable parameters and target modes. The meaning of the bits in the flash pass and fail result parameter (FPFR) varies in initialization, programming, and erasure.

Table 25.4 Parameters and Target Modes

Parameter	Download	Initialization	Programming	Erasure	R/W	Initial Value	Allocation
DPFR	O	—	—	—	R/W	Undefined	On-chip RAM*
FPFR	O	O	O	O	R/W	Undefined	R0L of CPU
FPEFEQ	—	O	—	—	R/W	Undefined	ER0 of CPU
FMPAR	—	—	O	—	R/W	Undefined	ER1 of CPU
FMPDR	—	—	O	—	R/W	Undefined	ER0 of CPU
FEBS	—	—	—	O	R/W	Undefined	ER0 of CPU

Note: * A single byte of the start address of the on-chip RAM specified by FTDAR

(a) Download Control

The on-chip program is automatically downloaded by setting the SCO bit in FCCS to 1. The on-chip RAM area to download the on-chip program is the 4-Kbyte area starting from the start address specified by FTDAR. Download is set by the programming/erasing interface registers, and the download pass and fail result parameter (DPFR) indicates the return value.

(2) State Transition Diagram

The state transition after USB boot mode is initiated is shown in figure 25.10.

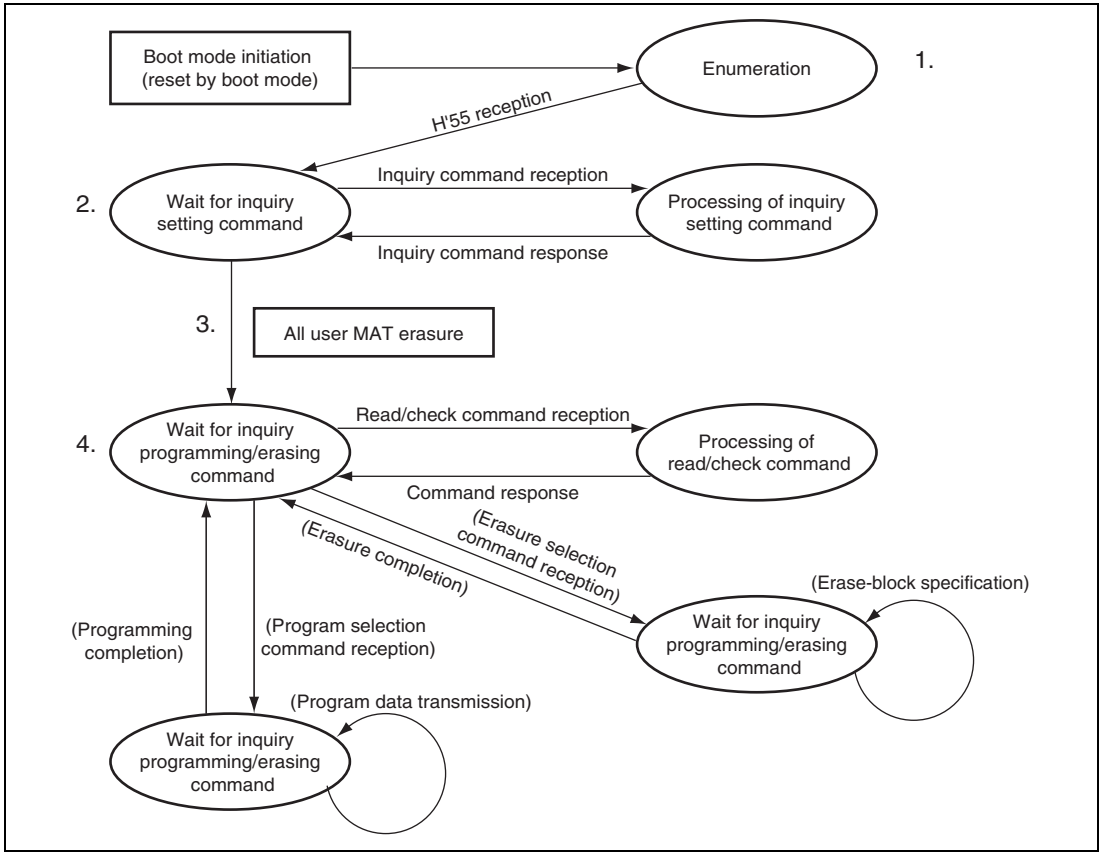


Figure 25.10 USB Boot Mode State Transition Diagram

1. After a transition to the USB boot mode is made, the boot program embedded in this LSI is initialized. This LSI performs enumeration to the host after the USB boot program is initialized.
2. Inquiry information about the size, configuration, start address, and support status of the user MAT is transmitted to the host.
3. After inquiries have finished, all user MAT are automatically erased.

Table 30.2 DC Characteristics (2)

Conditions: V_{CC} = $PLLV_{CC}$ = DrV_{CC} = 3.0 V to 3.6 V, AV_{CC} = 3.0 V to 3.6 V, V_{ref} = 3.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = DrV_{SS} = AV_{SS} = 0 V*[1], T_a = –20°C to +75°C (regular specifications), T_a = –40°C to +85°C (wide-range specifications)

Item			Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Three-state leakage current (off state)	Ports 1, 2, 3, 6, A to F, H, I, J, K, M		| ITSI|	—	—	1.0	µA	V_{in} = 0.5 to V_{CC} – 0.5 V
Input pull-up MOS current	Ports D to F, H, I		$-I_p$	10	—	300	µA	V_{CC} = 3.0 to 3.6 V V_{in} = 0 V
Input capacitance	All input pins		C_{in}	—	—	15	pF	V_{in} = 0 V f = 1 MHz T_a = 25°C
Current consumption*[2]	Normal operation		I_{CC}*[4]	—	50	85	mA	f = 50 MHz
	Sleep mode			—	48	60		
	Subclock operation			—	5	10		32.768-kHz crystal resonator is used.
	Standby mode	Software standby mode*[3]		—	0.15	1.1		$T_a \leq 50°C$
				—	—	3.5		50°C < T_a
		Deep software standby mode	RAM ,USB retained*[3]*[7]	—	20	60	µA	$T_a \leq 50°C$
				—	—	200		50°C < T_a
			RAM, USB power supply halted TM32K halted	—	3	8		$T_a \leq 50°C$
				—	—	26		
			RAM, USB power supply halted TM32K operation	—	9	16		50°C < T_a
				—	—	41		
		Hardware standby mode		—	2	7		$T_a \leq 50°C$
				—	—	25		50°C < T_a
	All-module-clock-stop mode*[5]			—	23	30	mA	
Analog power supply current	During A/D and D/A conversion		AI_{CC}	—	1.0	2.5	mA	
	Standby for A/D and D/A conversion			—	0.5	1.0	µA	

Port Name	MCU Operating Mode	Reset	Hardware Standby Mode	Deep Software Standby Mode IOKEEP = 1/0		Software Standby Mode		Bus Released State
				OPE = 1	OPE = 0	OPE = 1	OPE = 0	
PB6/$\overline{\text{CS6}}$-D/(RD/$\overline{\text{WR}}$)/$\overline{\text{ADTRGH0}}$-B	All	Hi-Z	Hi-Z	[$\overline{\text{CS}}$ output]	[$\overline{\text{CS}}$ output]	[$\overline{\text{CS}}$ output]	[$\overline{\text{CS}}$ output]	[$\overline{\text{CS}}$ output]
				H	Hi-Z	H	Hi-Z	Hi-Z
				[Other than above]	[Other than above]	[Other than above]	[Other than above]	[Other than above]
				Keep	Keep	Keep	Keep	Keep
PB7/SDϕ	SDRAM mode	SDϕ output	Hi-Z	[SDϕ output]	[SDϕ output]	[SDϕ output]	[SDϕ output]	[SDϕ output] SDϕ output
				H	H	H	H	
	Other than SDRAM mode	H	Hi-Z	[Other than above]	[Other than above]	[Other than above]	[Other than above]	[Other than above]
				Keep	Keep	Keep	Keep	Keep
PC2 to PC3	All	Hi-Z	Hi-Z	Keep		Keep		