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	SH-2
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	CANbus, SCI
Peripherals	POR, PWM, WDT
Number of I/O	53
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	100-BFQFP
Supplier Device Package	100-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/hd64f7047f50v

Bit	Bit Name	Initial Value	R/W	Description
7	DTS	Undefined	—	DTC Transfer Mode Select Specifies whether the source or the destination is set to be a repeat area or block area, in repeat mode or block transfer mode. 0: Destination is repeat area or block area 1: Source is repeat area or block area
6	CHNE	Undefined	—	DTC Chain Transfer Enable When this bit is set to 1, a chain transfer will be performed. 0: Chain transfer is canceled 1: Chain transfer is set In data transfer with CHNE set to 1, determination of the end of the specified number of transfers, clearing of the activation source flag, and clearing of DTER is not performed.
5	DISEL	Undefined	—	DTC Interrupt Select When this bit is set to 1, a CPU interrupt request is generated for every DTC transfer. When this bit is set to 0, a CPU interrupt request is generated at the time when the specified number of data transfer ends.
4	NMIM	Undefined	—	DTC NMI Mode This bit designates whether to terminate transfers when an NMI is input during DTC transfers. 0: Terminate DTC transfer upon an NMI 1: Continue DTC transfer until end of transfer being executed
3 to 0	—	Undefined	—	Reserved These bits have no effect on DTC operation and should always be written with 0.

[Legend]

X: Don’t care

Chain Transfer: Setting the CHNE bit to 1 enables a number of data transfers to be performed consecutively in a single activation source. DTSAR, DTDAR, DTMR, DTCRA, and DTCRB can be set independently.

Figure 8.9 shows the chain transfer.

When activated, the DTC reads the register information start address stored at the vector address, and then reads the first register information at that start address. After the data transfer, the CHNE bit will be tested. When it has been set to 1, DTC reads next register information located in a consecutive area and performs the data transfer. These sequences are repeated until the CHNE bit is cleared to 0.

In the case of transfer with CHNE set to 1, an interrupt request to the CPU is not generated at the end of the specified number of transfers or by setting of the DISEL bit to 1, and the interrupt source flag for the activation source is not affected.

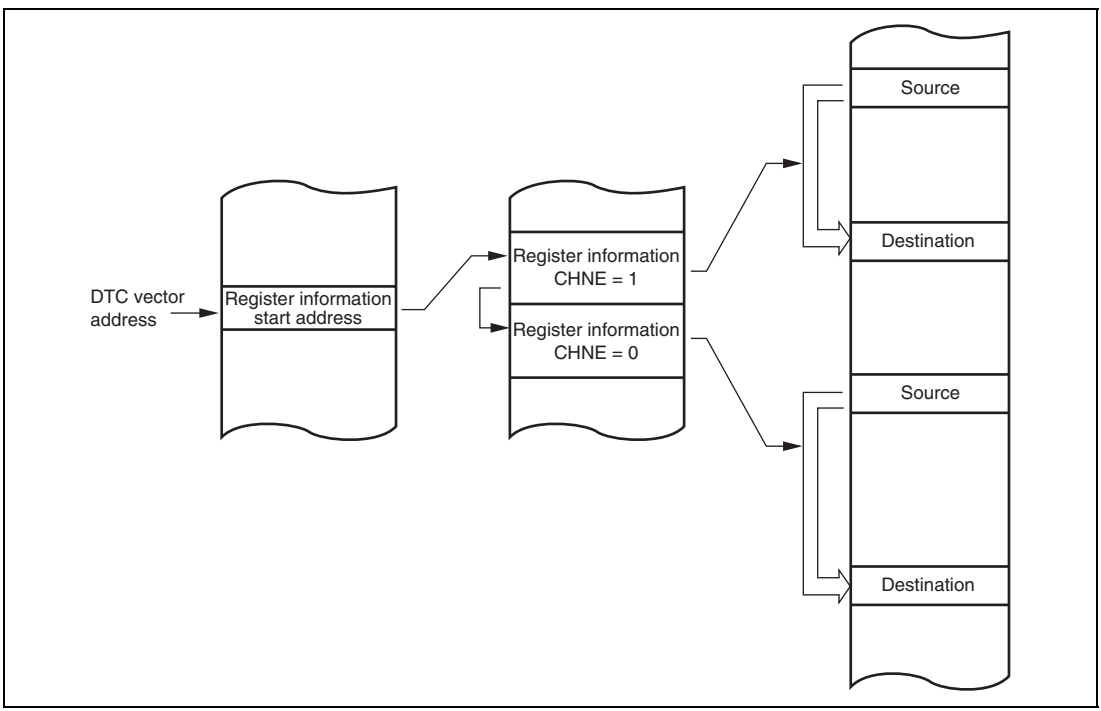

Figure 8.9 Chain Transfer

9.4 Address Map

Figure 9.2 shows the address format used by this LSI.

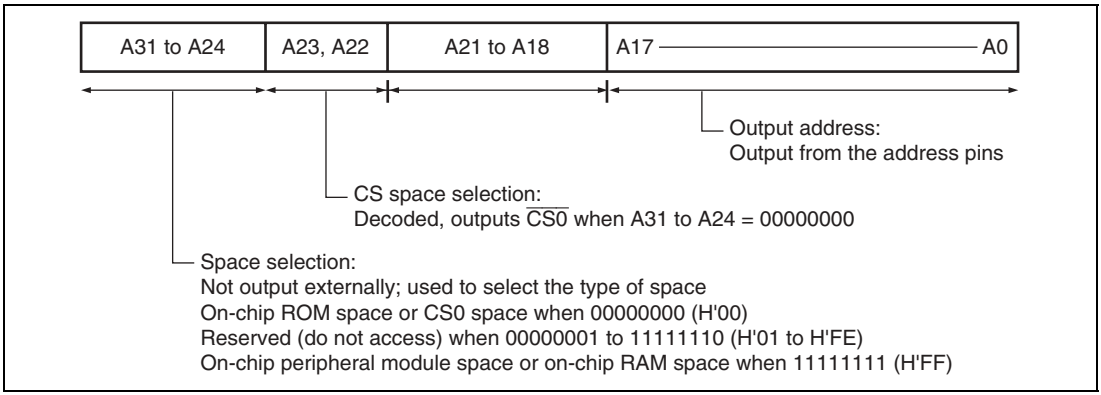

Figure 9.2 Address Format

This chip uses 32-bit addresses:

- Bits A31 to A24 are used to select the type of space and are not output externally.
- Bits A23 and A22 are decoded and output as chip select signals ($\overline{CS0}$) for the corresponding areas when bits A31 to A24 are 00000000.
- A17 to A0 are output externally. A21 to A18 are not output externally.

Table 9.2 shows the address map.

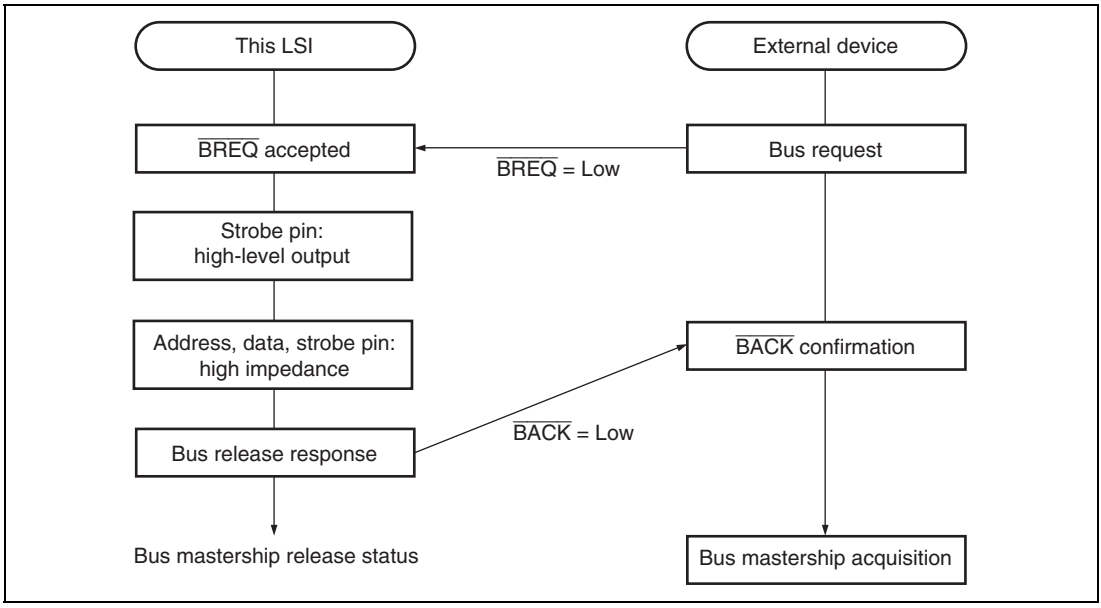

Figure 9.8 Bus Mastership Release Procedure

9.9 Memory Connection Example

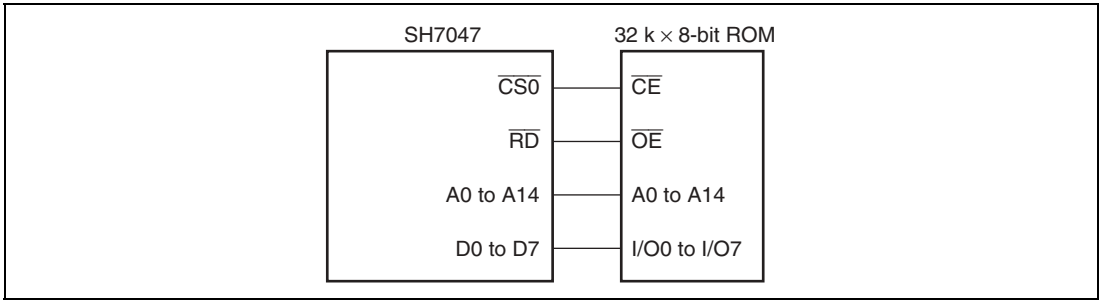

Figure 9.9 Example of 8-bit Data Bus Width ROM Connection

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
DTC activation	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow underflow
A/D converter start trigger	TGRA_0 compare match or input capture	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture
Interrupt sources	5 sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Overflow	4 sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	4 sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	5 sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3C • Compare match or input capture 3D • Overflow	5 sources • Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Overflow or Underflow

Notes:

○ : Possible

— : Not possible

10.3.2 Timer Mode Register (TMDR)

The TMDR registers are 8-bit readable/writable registers that are used to set the operating mode of each channel. The MTU has five TMDR registers, one for each channel. TMDR register settings should be changed only when TCNT operation is stopped.

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1, and should only be written with 1.
5	BFB	0	R/W	Buffer Operation B Specifies whether TGRB is to operate in the normal way, or TGRB and TGRD are to be used together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare is not generated. In channels 1 and 2, which have no TGRD, bit 5 is reserved. It is always read as 0, and should only be written with 0. 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation
4	BFA	0	R/W	Buffer Operation A Specifies whether TGRA is to operate in the normal way, or TGRA and TGRC are to be used together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare is not generated. In channels 1 and 2, which have no TGRC, bit 4 is reserved. It is always read as 0, and should only be written with 0. 0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation
3	MD3	0	R/W	Modes 0 to 3
2	MD2	0	R/W	These bits are used to set the timer operating mode.
1	MD1	0	R/W	See table 10.9 for details.
0	MD0	0	R/W	

10.4.4 Cascaded Operation

In cascaded operation, two 16-bit counters for different channels are used together as a 32-bit counter.

This function works by counting the channel 1 counter clock upon overflow/underflow of TCNT_2 as set in bits TPSC0 to TPSC2 in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

Table 10.30 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for channel 1 or 4, the counter clock setting is invalid and the counters operates independently in phase counting mode.

Table 10.30 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
Channels 1 and 2	TCNT_1	TCNT_2

Example of Cascaded Operation Setting Procedure: Figure 10.18 shows an example of the setting procedure for cascaded operation.

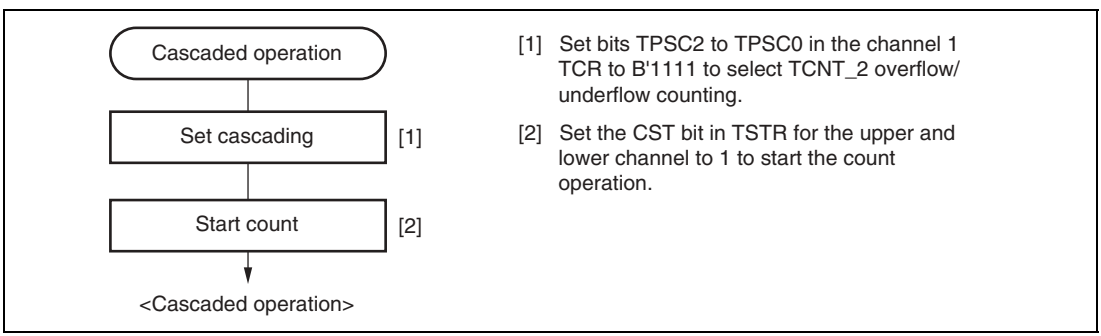

Figure 10.18 Cascaded Operation Setting Procedure

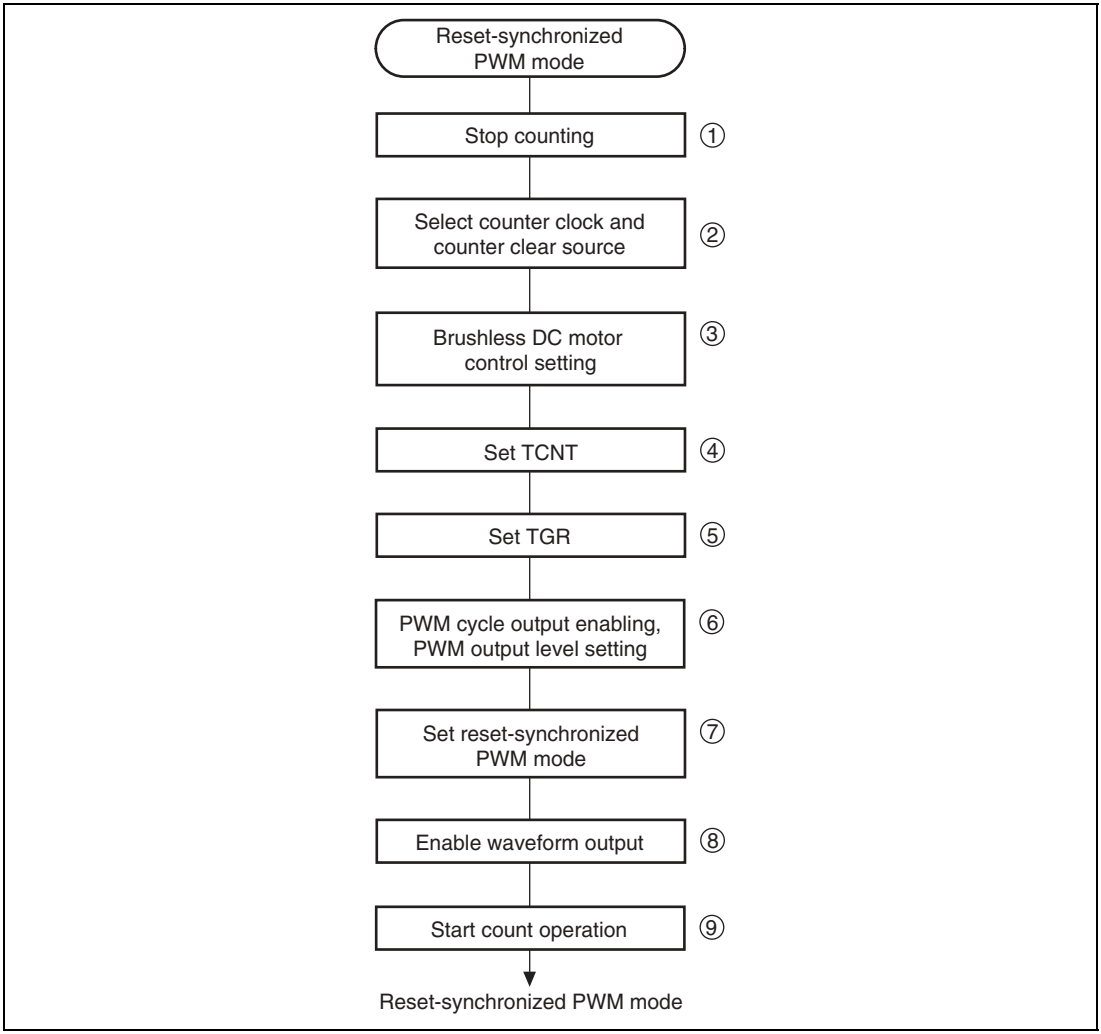

Figure 10.30 Procedure for Selecting the Reset-Synchronized PWM Mode

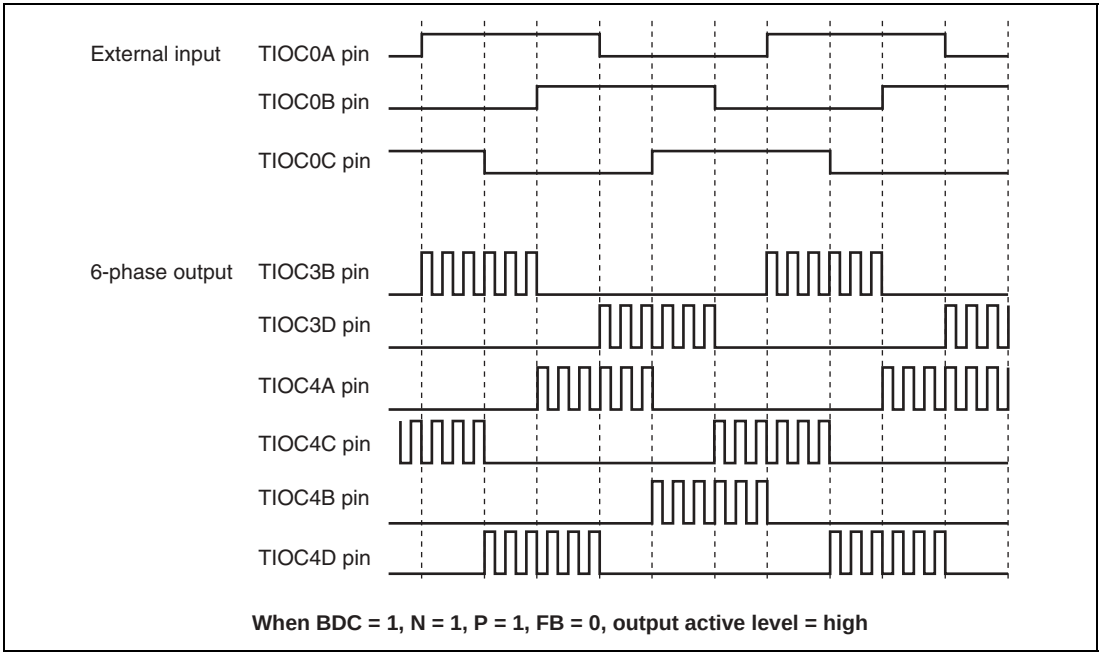

Figure 10.51 Example of Output Phase Switching by External Input (2)

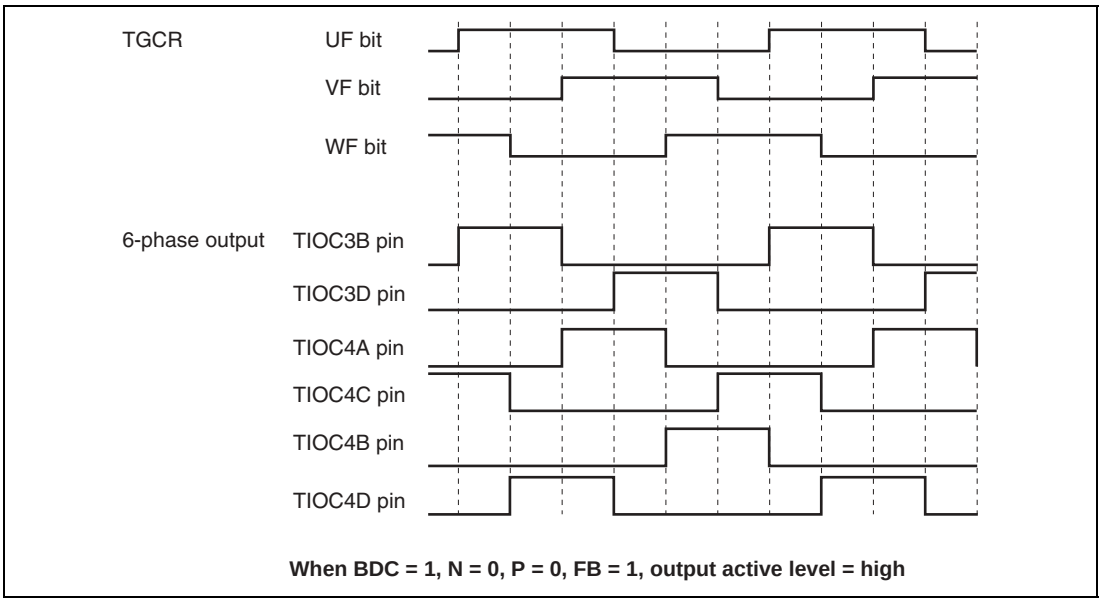

Figure 10.52 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)

10.5 Interrupts

10.5.1 Interrupts and Priorities

There are three kinds of MTU interrupt source; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own status flag and enable/disabled bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt request is generated, the corresponding status flag in TSR is set to 1. If the corresponding enable/disable bit in TIER is set to 1 at this time, an interrupt is requested. The interrupt request is cleared by clearing the status flag to 0.

Relative channel priorities can be changed by the interrupt controller, however the priority order within a channel is fixed. For details, see section 6, Interrupt Controller (INTC).

Table 10.42 lists the TPU interrupt sources.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 10.86 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

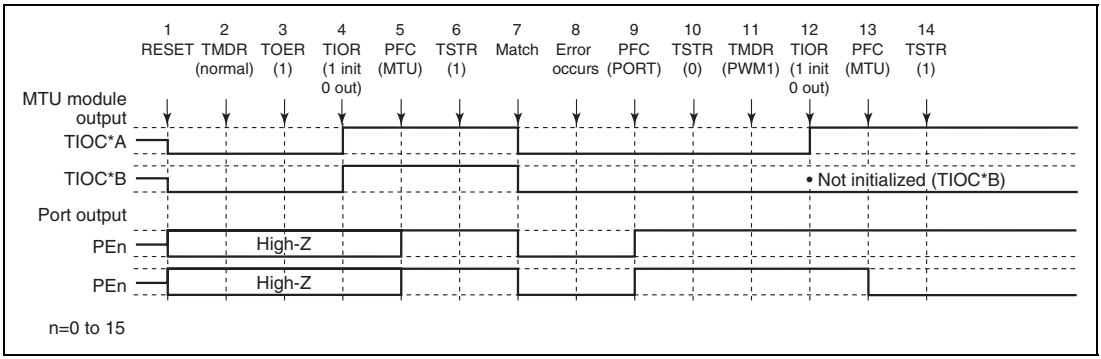

Figure 10.86 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

1 to 10 are the same as in Figure 10.85.

11. Set PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized. If initialization is required, initialize in normal mode, then switch to PWM mode 1.)
13. Set MTU output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 2: Figure 10.93 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

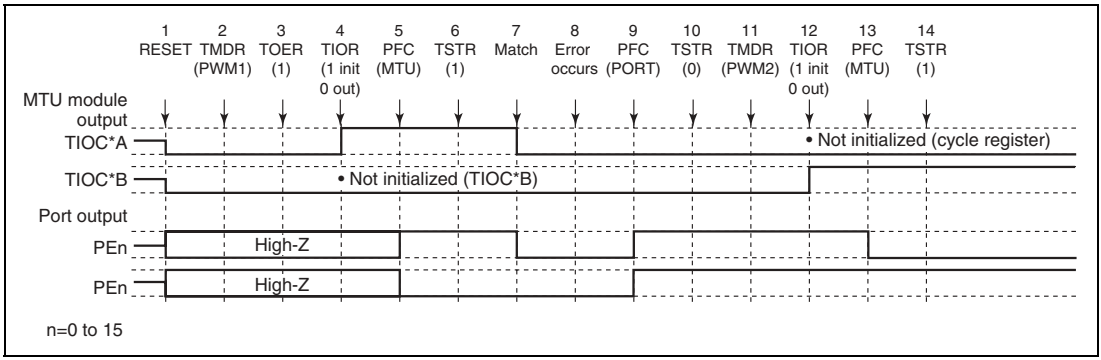

Figure 10.93 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

1 to 10 are the same as in Figure 10.91.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
13. Set MTU output with the PFC.
14. Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0–2, and therefore TOER setting is not necessary.

12.4 Operation in Asynchronous Mode

Figure 12.2 shows the general format for asynchronous serial communication. One frame consists of a start bit (low level), followed by data, a parity bit, and finally stop bits (high level). In asynchronous serial communication, the transmission line is usually held in the mark state (high level). The SCI monitors the communication line, and when it goes to the space state (low level), recognizes a start bit and starts serial communication. Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communication. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transfer.

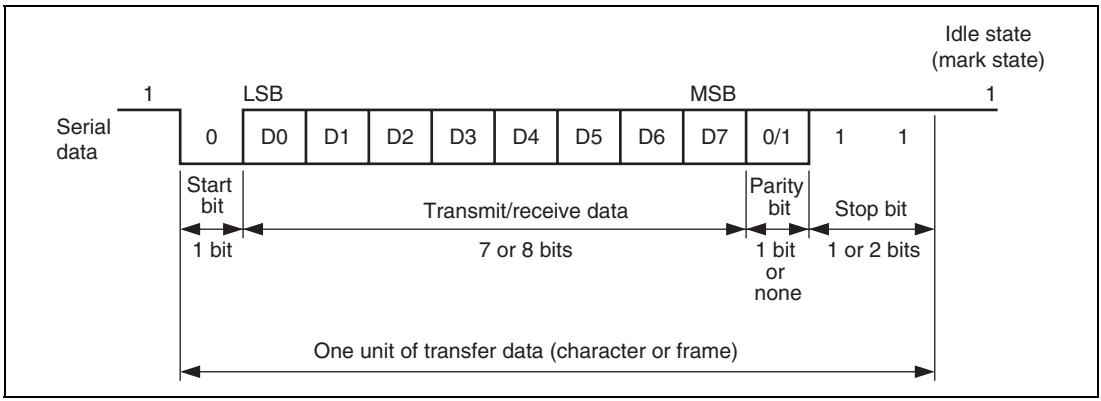

Figure 12.2 Data Format in Asynchronous Communication (Example with 8-Bit Data, Parity, Two Stop Bits)

12.4.1 Data Transfer Format

Table 12.8 shows the data transfer formats that can be used in asynchronous mode. Any of 12 transfer formats can be selected according to the SMR setting. For details on the multiprocessor bit, refer to section 12.5, Multiprocessor Communication Function.

Mailbox Initial Settings: Mailboxes are held in RAM, and so their initial values are undefined after power is supplied. Initial values must therefore be set in all the mailboxes (by writing 0s or 1s).

Mailbox Transmit/Receive Settings: The HCAN2 has 32 mailboxes. Mailbox 31 and 0 are receive-only, while mailboxes 1 to 30 can be set for transmission or reception.

Use MBC[2:0] bits in the mailbox to set the corresponding mailbox for transmission or reception use. When setting mailboxes for reception, in order to improve message reception efficiency, high-priority messages should be set in mailboxes with high mailbox number.

Set MBC[2:0] bits of unused mailboxes to B'111 and do not access them.

Note: Restrictions apply to the use of the mailbox 31 for transmission. Carefully read section 15.8, Usage Notes.

Message Transmission Method Setting : The following two kinds of message transmission methods are available.

- Transmission order determined by message identifier priority
- Transmission order determined by mailbox number priority

Either of the message transmission methods can be selected with the message transmission method bit (MCR2) in the master control register (MCR): When messages are set to be transmitted according to the message identifier priority, if several messages are designated as waiting for transmission (TXPR = 1), depending on the settings of the message identifier, IDE, EXT-ID, and RTR bit, the message with the highest priority (set values of the identifier, IDE, EXT-ID, and RTR bit are low) is stored in the transmit buffer. CAN bus arbitration is then carried out for the message stored in the transmit buffer, and the message is transmitted when the transmission right is acquired. When the TXPR bit is set, the highest-priority message is found and stored in the transmit buffer.

When messages are set to be transmitted according to the mailbox number proiority, if several messages are designated as waiting for transmission (TXPR = 1), the message with the highest mailbox number is stored in the transmit buffer. CAN bus arbitration is then carried out for the message stored in the transmit buffer, and the message is transmitted when the transmission right is acquired.

Section 18 I/O Ports

This LSI has five ports: A, B, D, E, and F. Port A is a 16-bit port, port B is a 6-bit port, port D is a 9-bit port, and port E is a 22-bit port, all supporting both input and output. Port F is a 16-bit input-only port.

All the port pins are multiplexed as general input/output pins and special function pins. The functions of the multiplex pins are selected by means of the pin function controller (PFC). Each port is provided with a data register for storing the pin data.

18.1 Port A

Port A is an input/output port with the 16 pins shown in figure 18.1.

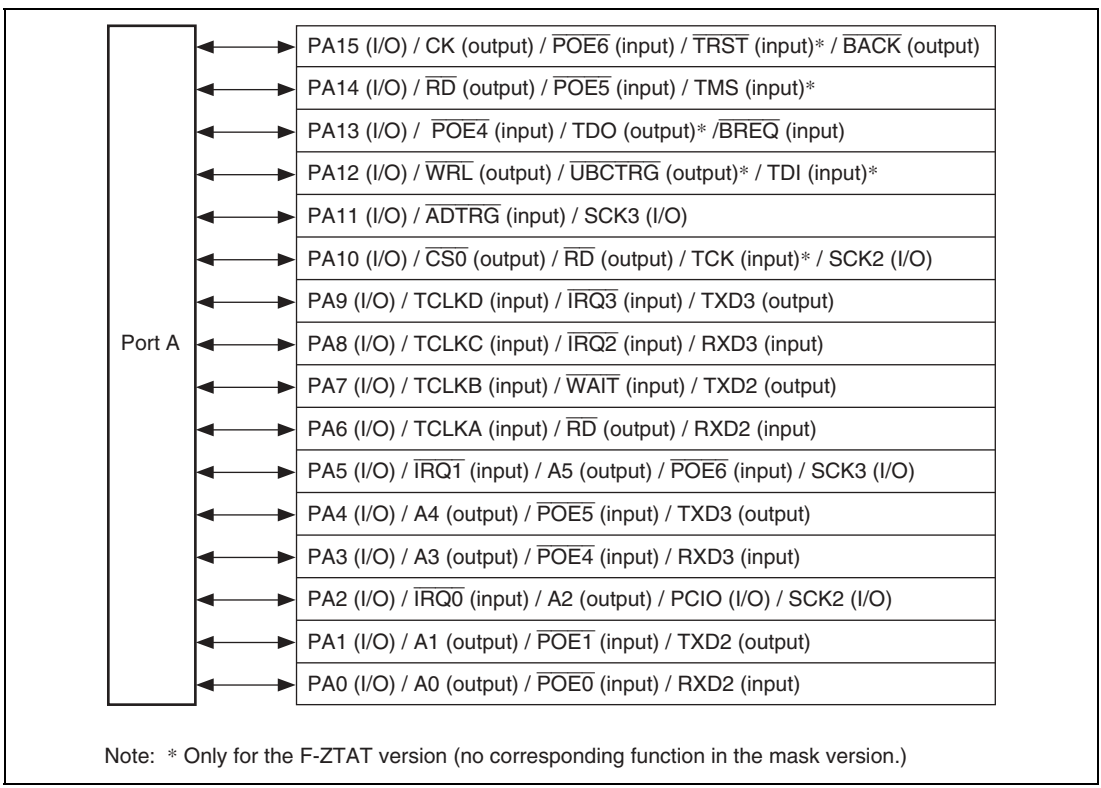

Figure 18.1 Port A

19.5.2 Flash Memory Control Register 2 (FLMCR2)

FLMCR2 is a register that displays the state of flash memory programming/erasing.

Bit	Bit Name	Initial Value	R/W	Description
7	FLER	0	R	Indicates that an error has occurred during an operation on flash memory (programming or erasing). When flash memory goes to the error-protection state, FLER is set to 1. See section 19.9.3, Error Protection, for details.
6 to 0	—	All 0	R	Reserved These bits are always read as 0.

19.5.3 Erase Block Register 1 (EBR1)

EBR1 specifies the flash memory erase block. EBR1 is initialized to H'00 when a high level is input to the FWP pin. It is also initialized to H'00, when the SWE bit in FLMCR1 is 0 regardless of value in the FWP pin. Do not set more than one bit at a time in EBR1 and EBR2, as this will cause all the bits in EBR1 and EBR2 to be automatically cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
7	EB7	0	R/W	When this bit is set to 1, 4 kbytes of EB7 (H'007000 to H'007FFF) are to be erased.
6	EB6	0	R/W	When this bit is set to 1, 4 kbytes of EB6 (H'006000 to H'006FFF) are to be erased.
5	EB5	0	R/W	When this bit is set to 1, 4 kbytes of EB5 (H'005000 to H'005FFF) are to be erased.
4	EB4	0	R/W	When this bit is set to 1, 4 kbytes of EB4 (H'004000 to H'004FFF) are to be erased.
3	EB3	0	R/W	When this bit is set to 1, 4 kbytes of EB3 (H'003000 to H'003FFF) are to be erased.
2	EB2	0	R/W	When this bit is set to 1, 4 kbytes of EB2 (H'002000 to H'002FFF) are to be erased.
1	EB1	0	R/W	When this bit is set to 1, 4 kbytes of EB1 (H'001000 to H'001FFF) are to be erased.
0	EB0	0	R/W	When this bit is set to 1, 4 kbytes of EB0 (H'000000 to H'000FFF) are to be erased.

applied or disconnected, fix the FWP pin level at V_{cc} and place the flash memory in the hardware protection state in advance.

Conditions for this power-on and power-off timing should also be applied in the event of a power failure and subsequent recovery.

FWP application/disconnection (see figures 19.11 to 19.13): If V_{cc} is on or off while low level is applied to FWP pin, a voltage surge from low level on the RESET pin may cause unintentional programming or erasing of flash memory. Applying voltage to FWP should be carried out while MCU operation is in a stable condition. If MCU operation is not stable, fix the FWP pin high and set the protection state. The following points must be observed concerning FWP application and disconnection to prevent unintentional programming or erasing of flash memory:

- Apply voltage to FWP while the V_{cc} voltage is stable enough to satisfy the specification voltage range.
- In boot mode, apply voltage to FWP or disconnect it during a reset.
- Prior to applying voltage while FWP pin is in low level in boot mode, ensure that the RESET pin level is surely kept low despite the applying voltage is rising to V_{cc}. Note that in a case where ICs for reset are used, the voltage level of RESET pin can transiently exceed 1/2 V_{cc} while V_{cc} is rising.
- In user program mode, FWP can be switched between high and low level regardless of the reset state. FWP input can also be switched during execution of a program in flash memory.
- Apply voltage to FWP while programs are not running away.
- Disconnect FWP only when the SWE, ESU, PSU, EV, PV, P, and E bits in FLMCR1 are cleared. Make sure that the SWE, ESU, PSU, EV, PV, P, and E bits are not set by mistake when applying voltage to FWP pin or disconnecting.

Do not apply a constant low level to the FWP pin: If a program runs away while low level is applied to FWP pin, incorrect programming or erasing may occur. Apply a low level to the FWP pin only when programming or erasing flash memory. Avoid creating a system configuration in which a low level is constantly applied to the FWP pin. Also, while a low level is applied to the FWP pin, the watchdog timer should be activated to prevent excess programming or excess erasing due to program runaway, etc.

Use the recommended algorithm when programming and erasing flash memory: The recommended algorithm enables programming and erasing to be carried out without subjecting the device to voltage stress or sacrificing program data reliability. When setting the P or E bit in FLMCR1, the watchdog timer should be set beforehand as a precaution against program runaway, etc.

Do not set or clear the SWE bit during execution of a program in flash memory: Wait for at least 100 µs after clearing the SWE bit before executing a program or reading data in flash memory. When the SWE bit is set, data in flash memory can be rewritten. Access flash memory only for verify operations (verification during programming/erasing). Also, do not clear the SWE

Item			Symbol	Min	Typ	Max	Unit	Measurement Conditions
Three-state leak current (while OFF)	Port A, B, D, E		$\lvert I_{tsi} \rvert$	—	—	1.0	µA	Vin = 0.5 to V_{CC} –0.5 V
Output high-level voltage	All output pins		V_{OH}	V_{CC} – 0.5	—	—	V	I_{OH} = –200 µA
				3.5	—	—	V	I_{OH} = –1 mA
Output low-level voltage	All output pins		V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
	PE9, PE11 to PE21			—	—	1.5	V	I_{OL} = 15 mA
Input capacitance	$\overline{RES}$		C_{in}	—	—	80	pF	Vin = 0 V
	NMI			—	—	50	pF	ϕ = 1 MHz
	All other input pins			—	—	20	pF	Ta = 25°C
Current consumption*[2]	Normal operation	Clock 1:1	I_{CC}	—	180	200	mA	ϕ = 40 MHz
				—	120	140	mA	ϕ = 25 MHz
		Clock 1:1/2		—	220	235	mA	ϕ = 50 MHz
				—	160	180	mA	ϕ = 40 MHz
	Sleep	Clock 1:1		—	140	190	mA	ϕ = 40 MHz
		Clock 1:1/2		—	150	200	mA	ϕ = 50 MHz
	Standby			—	3	100	µA	$T_a \leq 50°C$
				—	—	500	µA	$50°C < T_a$
	Write operation	Clock 1:1		—	180	200	mA	V_{CC} = 5.0 V, ϕ = 40 MHz
		Clock 1:1/2		—	220	235	mA	V_{CC} = 5.0 V, ϕ = 50 MHz
Analog supply current	During A/D conversion, A/D converter idle state		AI_{CC}	—	2	5	mA	
	During standby			—	—	5	µA	
RAM standby voltage			V_{RAM}	2.0	—	—	V	V_{CC}

[Operating precautions]

1. When the A/D converter is not used, do not leave the AV_{CC}, and AV_{SS} pins open.

Notes: 1. See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

2. The current consumption is measured when V_{IH}min = V_{CC} – 0.5 V, V_{IL} = 0.5 V, with all output pins unloaded.

Register Abbreviation	Power-On Reset	Manual Reset	Hardware Standby	Software Standby	Module Standby	Sleep	Module
TGRA_1	Initialized	Held	Initialized	Initialized	Initialized	Held	MTU (channel 2)
TGRB_1	Initialized	Held	Initialized	Initialized	Initialized	Held	
TCR_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TMDR_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TIOR_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TIER_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TSR_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TCNT_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TGRA_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
TGRB_2	Initialized	Held	Initialized	Initialized	Initialized	Held	
IPRA	Initialized	Initialized	Initialized	Held	—	Held	INTC
IPRD	Initialized	Initialized	Initialized	Held	—	Held	
IPRE	Initialized	Initialized	Initialized	Held	—	Held	
IPRF	Initialized	Initialized	Initialized	Held	—	Held	
IPRG	Initialized	Initialized	Initialized	Held	—	Held	
IPRH	Initialized	Initialized	Initialized	Held	—	Held	
ICR1	Initialized	Initialized	Initialized	Held	—	Held	
ISR	Initialized	Initialized	Initialized	Held	—	Held	
IPRI	Initialized	Initialized	Initialized	Held	—	Held	
IPRJ	Initialized	Initialized	Initialized	Held	—	Held	
IPRK	Initialized	Initialized	Initialized	Held	—	Held	
ICR2	Initialized	Initialized	Initialized	Held	—	Held	
PADRL	Initialized	Held	Initialized	Held	—	Held	Port A
PAIORL	Initialized	Held	Initialized	Held	—	Held	
PACRL3	Initialized	Held	Initialized	Held	—	Held	
PACRL1	Initialized	Held	Initialized	Held	—	Held	
PACRL2	Initialized	Held	Initialized	Held	—	Held	
PBDR	Initialized	Held	Initialized	Held	—	Held	Port B
PBIOR	Initialized	Held	Initialized	Held	—	Held	
PBCR1	Initialized	Held	Initialized	Held	—	Held	
PBCR2	Initialized	Held	Initialized	Held	—	Held	
PDDRL	Initialized	Held	Initialized	Held	—	Held	Port D
PDIORL	Initialized	Held	Initialized	Held	—	Held	
PDCRL1	Initialized	Held	Initialized	Held	—	Held	
PDCRL2	Initialized	Held	Initialized	Held	—	Held	
PEDRL	Initialized	Held	Initialized	Held	—	Held	Port E

Register Abbreviation	Power-On Reset	Manual Reset	Hardware Standby	Software Standby	Module Standby	Sleep	Module
ADCR_1	Initialized	Held	Initialized	Initialized	Initialized	Held	A/D
FLMCR1	Initialized	Initialized	Initialized	Initialized	Initialized	Held	FLASH
FLMCR2	Initialized	Initialized	Initialized	Initialized	Initialized	Held	
EBR1	Initialized	Initialized	Initialized	Initialized	Initialized	Held	
EBR2	Initialized	Initialized	Initialized	Initialized	Initialized	Held	
UBARH	Initialized	Held	Initialized	Held	Initialized	Held	UBC
UBARL	Initialized	Held	Initialized	Held	Initialized	Held	
UBAMRH	Initialized	Held	Initialized	Held	Initialized	Held	
UBAMRL	Initialized	Held	Initialized	Held	Initialized	Held	
UBBR	Initialized	Held	Initialized	Held	Initialized	Held	
UBCR	Initialized	Held	Initialized	Held	Initialized	Held	
TCSR	Initialized	Initialized	Initialized	Initialized/ Held*1	—	Held	WDT
TCNT	Initialized	Initialized	Initialized	Initialized	—	Held	
RSTCSR	Initialized/ Held*2	Held	Initialized	Initialized	—	Held	
SBYCR	Initialized	Initialized	Initialized	Held	—	Held	Power-down state
SYSCR	Initialized	Held	Initialized	Held	—	Held	
MSTCR1	Initialized	Held	Initialized	Held	—	Held	
MSTCR2	Initialized	Held	Initialized	Held	—	Held	
BCR1	Initialized	Held	Initialized	Held	—	Held	BSC
BCR2	Initialized	Held	Initialized	Held	—	Held	
WCR1	Initialized	Held	Initialized	Held	—	Held	
RAMER	Initialized	Held	Initialized	Held	—	Held	FLASH
DTEA	Initialized	Held	Initialized	Initialized	Initialized	Held	DTC
DTEB	Initialized	Held	Initialized	Initialized	Initialized	Held	
DTEC	Initialized	Held	Initialized	Initialized	Initialized	Held	
DTED	Initialized	Held	Initialized	Initialized	Initialized	Held	
DTCSR	Initialized	Held	Initialized	Initialized	Initialized	Held	
DTBR	Undefined	Held	Held	Held	Held	Held	
DTEE	Initialized	Held	Initialized	Initialized	Initialized	Held	
DTEF	Initialized	Held	Initialized	Initialized	Initialized	Held	
ADTSR	Initialized	Held	Initialized	Held	—	Held	A/D

Notes: 1. The bits 7 to 5 (OVF, WT/$\overline{IT}$, and TME) in TCSR are initialized and the bits 2 to 0 (CKS2 to CKS0) are retained.

2. RSTCSR is retained in spite of power-on reset by WDT overflow.