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	Last Time Buy
Core Processor	SH-2A
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	CANbus, EBI/EMI, FIFO, I²C, SCI, Serial Sound
Peripherals	DMA, POR, PWM, WDT
Number of I/O	104
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	176-LQFP
Supplier Device Package	176-LFQFP (24x24)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/ds72011rb120fpv

Figure 6.12 shows the timing for saving to a register bank. Saving to a register bank takes place between the start of interrupt exception handling and the start of fetching the first instruction in the exception service routine.

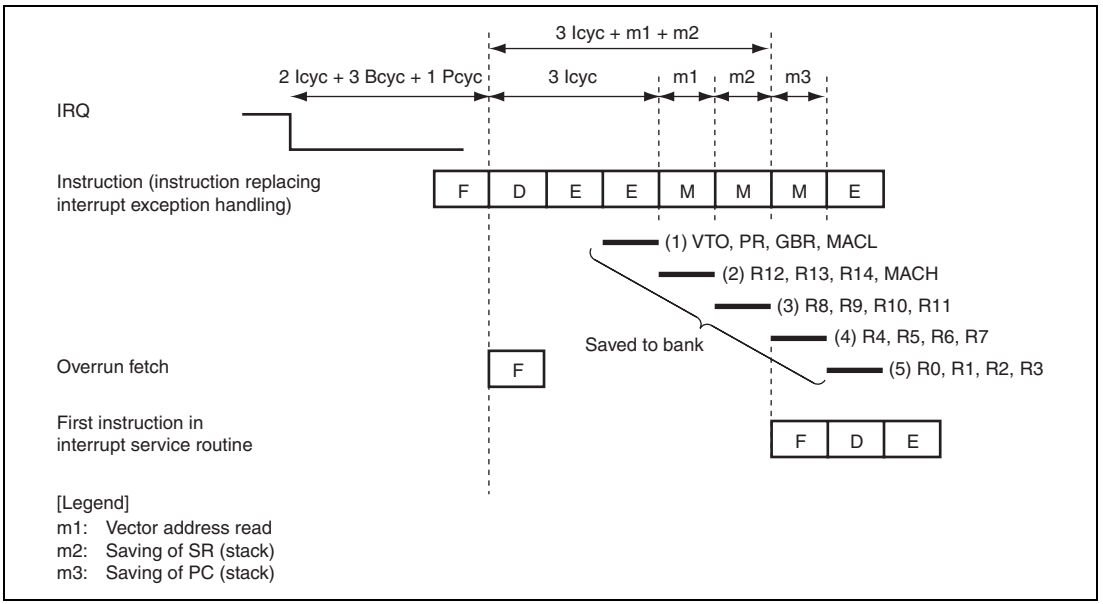


Figure 6.12 Bank Save Timing

(2) Restoration from Bank

The RESBANK (restore from register bank) instruction is used to restore data saved in a register bank. After restoring data from the register banks with the RESBANK instruction at the end of the interrupt service routine, execute the RTE instruction to return from exception handling.

7.5 Usage Notes

1. The CPU can read from or write to the UBC registers via the I bus. Accordingly, during the period from executing an instruction to rewrite the UBC register till the new value is actually rewritten, the desired break may not occur. In order to know the timing when the UBC register is changed, read from the last written register. Instructions after then are valid for the newly written register value.
2. The UBC cannot monitor access to the C bus and I bus cycles in the same channel.
3. When a user break interrupt request and another exception source occur at the same instruction, which has higher priority is determined according to the priority levels defined in table 5.1 in section 5, Exception Handling. If an exception source with higher priority occurs, the user break interrupt request is not received.
4. Note the following when a break occurs in a delay slot.
   If a pre-execution break is set at a delay slot instruction, the user break interrupt request is not received immediately before execution of the branch destination.
5. User breaks are disabled during UBC module standby mode. Do not read from or write to the UBC registers during UBC module standby mode; the values are not guaranteed.
6. Do not set an address within an interrupt exception handling routine whose interrupt priority level is at least 15 (including user break interrupts) as a break address.
7. Do not set break after instruction execution for the SLEEP instruction or for the delayed branch instruction where the SLEEP instruction is placed at its delay slot.
8. When setting a break for a 32-bit instruction, set the address where the upper 16 bits are placed. If the address of the lower 16 bits is set and a break before instruction execution is set as a break condition, the break is handled as a break after instruction execution.
9. Do not set a break after instruction execution for the DIVU or DIVS instruction. If a break after instruction execution is set for the DIVU or DIVS instruction and an exception or interrupt occurs during execution of the DIVU or DIVS instruction, a break after instruction execution occurs even though execution of the DIVU or DIVS instruction is halted.
10. Do not set a pre-execution break for the instruction that comes after the DIVU or DIVS instruction. If a pre-execution break is set for the instruction that comes after the DIVU or DIVS instruction and an exception or interrupt occurs during execution of the DIVU or DIVS instruction, a pre-execution break occurs even though execution of the DIVU or DIVS instruction is halted.
11. Do not set a pre-execution break and a break after instruction execution simultaneously in one address. For example, if a pre-execution break for channel 0 and a break after instruction execution for channel 1 are set simultaneously for one address, a break generated prior to instruction execution for channel 0 can set a condition-match flag after the instruction execution for channel 1.

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	ICF	0	R/W	Instruction Cache Flush Writing 1 flushes all instruction cache entries (clears the V and LRU bits of all instruction cache entries to 0). Always reads 0. Write-back to external memory is not performed when the instruction cache is flushed.
10, 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ICE	0	R/W	Instruction Cache Enable Indicates whether the instruction cache function is enabled or disabled. 0: Instruction cache disabled 1: Instruction cache enabled
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	OCF	0	R/W	Operand Cache Flush Writing 1 flushes all operand cache entries (clears the V, U, and LRU bits of all operand cache entries to 0). Always reads 0. Write-back to external memory is not performed when the operand cache is flushed.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	WT	0	R/W	Write Through Selects write-back mode or write-through mode. 0: Write-back mode 1: Write-through mode
0	OCE	0	R/W	Operand Cache Enable Indicates whether the operand cache function is enabled or disabled. 0: Operand cache disabled 1: Operand cache enabled

(b) 16-Bit Bus Channel

If a 16-bit bus is selected by the external bus width select bits in the CSn control register, A27 to A1 are enabled as address signals for word units and A0 is disabled (fixed low level). Table 9.8 shows the data alignment corresponding to byte addresses for different data sizes.

Pins $\overline{WR1}$ and $\overline{WR0}$ are enabled when byte strobe mode (WRMOD = 0) is selected. Pins $\overline{WR3}$ and $\overline{WR2}$ are disabled. Pins $\overline{BC3}$ to $\overline{BC0}$ are not used.

Only the $\overline{WR1}$ pin is enabled when one-write strobe mode (WRMOD = 1) is selected. A low-level signal is output from the $\overline{WR1}$ pin during write access, regardless of the data size. At this time the $\overline{WR0}$ pin is disabled (fixed high level). The valid byte positions are indicated by pins $\overline{BC1}$ and $\overline{BC0}$.

Table 9.8 Data Alignment (16-Bit Bus Channel)

		DATA				$\overline{WR}/\overline{BC}$			
Data Size	Byte Address (Lower 2 Bits)	[31:24]	[23:16]	[15:8]	[7:0]	[3]	[2]	[1]	[0]
Byte	0	×	×	O	×	*	*	L	H
	1	×	×	×	O	*	*	H	L
	2	×	×	O	×	*	*	L	H
	3	×	×	×	O	*	*	H	L
Word	0	×	×	O	O	*	*	L	L
	2	×	×	O	O	*	*	L	L
Longword	0 (1st)	×	×	O	O	*	*	L	L
	2 (2nd)	×	×	O	O	*	*	L	L

Note: The valid bits in the data bus for each data size are indicated by circles (O).
Crosses (×) indicate bus data bits that are undefined.
Asterisks (*) indicate write/byte control bits that are disabled (fixed high level).

11.8 Determining DMA Channel Priority

11.8.1 Channel Priority Order

Channel priority is allocated in descending order from channel 0; that is priority follows the below relation, where P indicates priority.

$P_{channel\ 0} > P_{channel\ 1} > P_{channel\ 3} \cdots P_{channel\ 6} > P_{channel\ 7}$. This order is fixed.

11.8.2 Operation during Multiple DMA Requests

The DMAC determines the priority every time single operand transfer is performed.

When a DMA request with a higher priority is generated during transfer for one channel, the transfer for the higher-priority channel only starts after the end of the current operand transfer. Figure 11.10 shows overall operation when multiple DMA requests are generated. The thick lines in the figure indicate the periods over which the DMA request signals are at the low level. Here channels 0, 2 and 3 are set to a level sense and channel 1 is set to an edge sense.

1. Since the channel 2 request is masked, it is regarded as non-existent. Thus, transfer on channel 3 starts up.
2. Since channel 0 has the highest priority, transfer on this channel starts up.
3. Since channel 2 has the higher priority of the requests at this point, transfer on this channel restarts.
4. Transfer on channel 3 is restarted as there are no other requests at this point.
5. When the DMA requests are simultaneously generated for channels 0, 1, and 3, transfer on channel 0 starts up because it has the highest priority.
6. After the transfer on channel 0 is complete, transfer on channel 1 starts up because it has the second highest priority.
7. A further DMA request (the selected edge) is received on channel 1 while DMA transfer is in progress. Transfer on channel 1 is thus restarted after completion of the current round of transfer on channel 1. No masking period applies in the case of edge sensing.
8. On completion of the transfer on channel 1, transfer on channel 3 starts up since there are no other requests.
9. No transfer starts up immediately after the end of the unit transfer operation on channel, since channel 3 requests are masked and there are no other requests. Transfer on channel 3 only restarts after the end of the masking period.

Table 12.18 TIORH_4 (Channel 4)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_4 Function**	**TIOC4B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 12.26 TIORH_4 (Channel 4)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_4 Function**	**TIOC4A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

(12) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 12.138 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

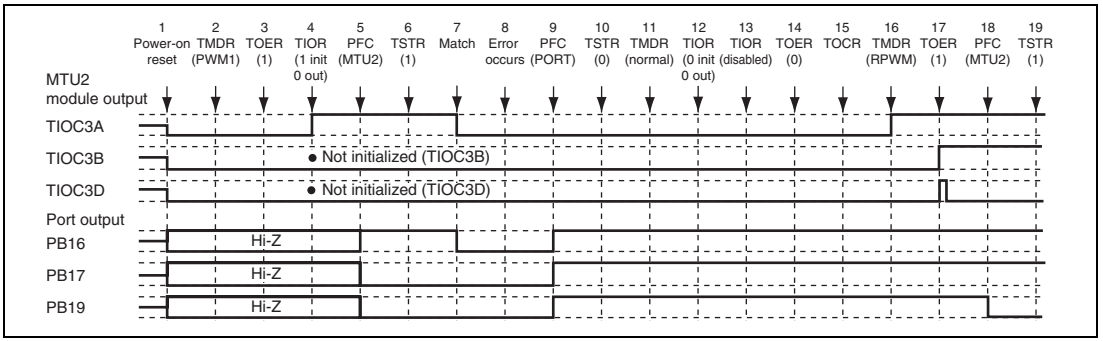


Figure 12.138 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

1 to 14 are the same as in figure 12.137.

15. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
16. Set reset-synchronized PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set MTU2 output with the PFC.
19. Operation is restarted by TSTR.

(24) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 12.150 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using completely new cycle and duty settings).

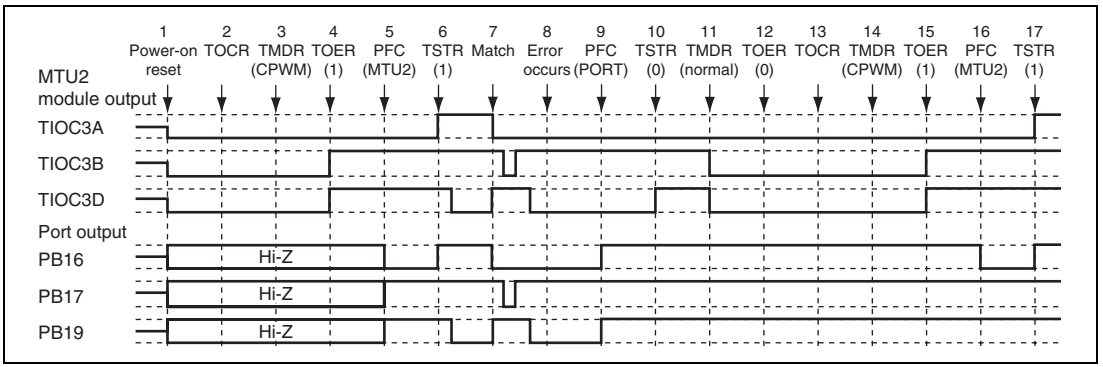


Figure 12.150 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 12.147.

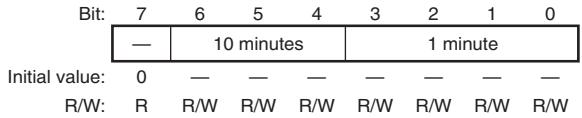
11. Set normal mode and make new settings. (MTU2 output goes low.)
12. Disable channel 3 and 4 output with TOER.
13. Select the complementary PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set complementary PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set MTU2 output with the PFC.
17. Operation is restarted by TSTR.

15.3.3 Minute Counter (RMINCNT)

RMINCNT is used for setting/counting in the BCD-coded minute section. The count operation is performed by a carry for each minute of the second counter.

The assignable range is from 00 through 59 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

RMINCNT is not initialized by a power-on reset or manual reset, or in deep standby and software standby modes.



Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0.The write value should always be 0.
6 to 4	10 minutes	Undefined	R/W	Counting Ten's Position of Minutes Counts on 0 to 5 for 60-minutes counting.
3 to 0	1 minute	Undefined	R/W	Counting One's Position of Minutes Counts on 0 to 9 once per second. When a carry is generated, 1 is added to the ten's position.

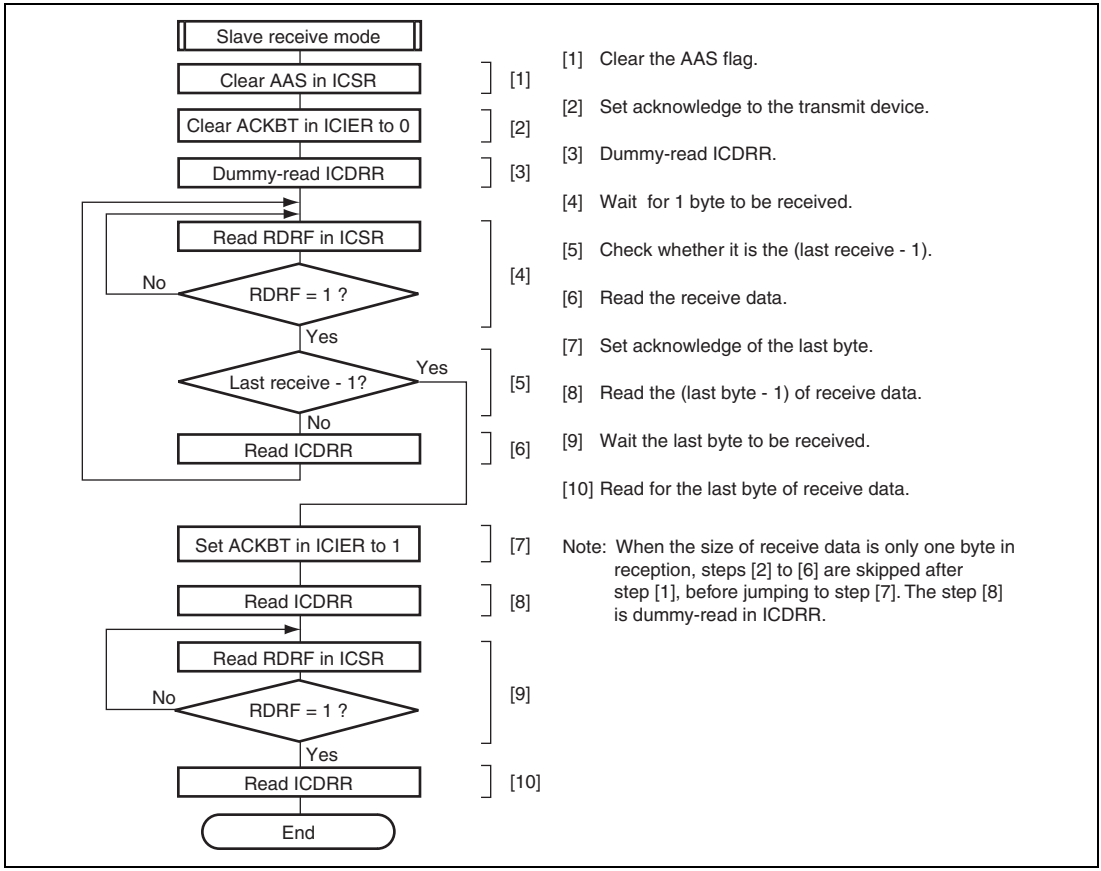


Figure 17.21 Sample Flowchart for Slave Receive Mode

18.4.2 Non-Compressed Modes

The non-compressed modes support all serial audio streams split into channels. It supports I^2S compatible format as well as many more variants on these modes.

(1) Slave Receiver

This mode allows the module to receive serial data from another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of the SSI module, operation is not guaranteed.

(2) Slave Transmitter

This mode allows the module to transmit serial data to another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of the SSI module, operation is not guaranteed.

(3) Master Receiver

This mode allows the module to receive serial data from another device. The clock and word select signals are internally derived from the oversampling clock. The format of these signals is defined in the configuration fields of the SSI module. If the incoming data does not follow the configured format, operation is not guaranteed.

(4) Master Transmitter

This mode allows the module to transmit serial data to another device. The clock and word select signals are internally derived from the oversampling clock. The format of these signals is defined in the configuration fields of the SSI module.

(5) Operating Setting Related to Word Length

All bits related to the SSICR's word length are valid in non-compressed modes. The SSI module supports many configurations, but the formats described below are I^2S compatible, MSB-first left-aligned, and MSB-first right-aligned.

18.5 Usage Notes

18.5.1 Limitations from Overflow during Receive DMA Operation

If an overflow occurs while the receive DMA is in operation, the module should be restarted. The receive buffer in the SSI consists of 32-bit registers that share the L and R channels. Therefore, data to be received at the L channel may sometimes be received at the R channel if an overflow occurs, for example, under the following condition: the control register (SSICR) has a 32-bit setting for both data word length (DWL2 to DWL0) and system word length (SWL2 to SWL0).

If an overflow is confirmed with the overflow error interrupt or overflow error status flag (the OIRQ bit in SSISR), write 0 to the EN bit in SSICR and DMEN bit to disable DMA in the SSI module, thus stopping the operation. (In this case, the controller setting should also be stopped.) After this, write 0 to the OIRQ bit to clear the overflow status, set DMA again and restart the transfer.

18.5.2 Note on Using Oversample Clock

To use the externally input clock as the oversample clock, refer to the section 4.6.1, Note on Inputting External Clock, in which the terms EXTAL and XTAL pins should be replaced by the AUDIO_X1 and AUDIO_X2 pins respectively.

To use the crystal resonator, refer to the section 4.6.2, Note on Using Crystal Resonator, in which the terms EXTAL and XTAL pins should be replaced by the AUDIO_X1 and AUDIO_X2 pins respectively.

Also, see section 4.6.3, Note on Resonator.

18.5.3 Restriction on Stopping Clock Supply

Once the bits MSTP53 and MSTP52 in the standby control register 5 (STBCR5) are cleared to 0 and the SSI operation is started, do not set these bits to 1 (stops clock supply to the SSI).

(3) Mailbox Control

The Mailbox Control handles the following functions:

- For received messages, compare the IDs and generate appropriate RAM addresses/data to store messages from the CAN Interface into the Mailbox and set/clear appropriate registers accordingly.
- To transmit messages, RCAN-ET will run the internal arbitration to pick the correct priority message, and load the message from the Mailbox into the Tx-buffer of the CAN Interface and set/clear appropriate registers accordingly.
- Arbitrates Mailbox accesses between the CPU and the Mailbox Control.
- Contains registers such as TXPR, TXCR, TXACK, ABACK, RXPR, RFPR, UMSR and MBIMR.

(4) CAN Interface

This block conforms to the requirements for a CAN Bus Data Link Controller which is specified in Ref. [3, 5]. It fulfils all the functions of a standard Data Link Controller as specified by the OSI 7 Layer Reference model. This functional entity also provides the registers and the logic which are specific to a given CAN bus, which includes the Receive Error Counter, Transmit Error Counter, the Bit Configuration Registers and various useful Test Modes. This block also contains functional entities to hold the data received and the data to be transmitted for the CAN Data Link Controller.

19.2.3 Input/Output Pins

Table 19.1 shows the pin configuration of the RCAN-ET.

Table 19.1 Pin Configuration

Channel	Name	Abbreviation	I/O	Function
0	Transmit data pin	CTx0	Output	CAN-bus transmit pin
	Receive data pin	CRx0	Input	CAN-bus receive pin
1	Transmit data pin	CTx1	Output	CAN-bus transmit pin
	Receive data pin	CRx1	Input	CAN-bus receive pin

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	PA26MD [2:0]	000	R/W	PA26 Mode These bits control the function of the PA26/A26/ DACT3/PINT2B pin. 000: PA26 I/O (port) 001: A26 output (BSC) 010: DACT3 output (DMAC) 011: PINT2B input (INTC) 100: Setting prohibited 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	PA25MD [2:0]	000	R/W	PA25 Mode These bits control the function of the PA25/A25/ DACK3/PINT1B pin. 000: PA25 I/O (port) 001: A25 output (BSC) 010: DACK3 output (DMAC) 011: PINT1B input (INTC) 100: Setting prohibited 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PA24MD [1:0]	00	R/W	PA24 Mode These bits control the function of the PA24/A24/ DREQ3/PINT0B pin. 00: PA24 I/O (port) 01: A24 output (BSC) 10: DREQ3 input (DMAC) 11: PINT0B input (INTC)

Bit	Bit Name	Initial Value	R/W	Description
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PD8MD[1:0]	00	R/W	PD8 Mode These bits control the function of the PD8/RxD0/DTEND1/TIOC0B pin. 00: PD8 I/O (port) 01: RxD0 input (SCIF) 10: DTEND1 output (DMAC) 11: TIOC0B I/O (MTU2)

(4) Port D Control Register 2 (PDCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PD7MD[1:0]		—	—	PD6MD[1:0]		—	—	PD5MD[1:0]		—	—	PD4MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	PD7MD[1:0]	00	R/W	PD7 Mode These bits control the function of the PD7/TxD0/DACT1/TIOC0A pin. 00: PD7 I/O (port) 01: TxD0 output (SCIF) 10: DACT1 output (DMAC) 11: TIOC0A I/O (MTU2)
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Section 25 Power-Down Modes

This LSI supports sleep mode, software standby mode, deep standby mode, and module standby mode. In power-down modes, functions of CPU, clocks, on-chip memory, or part of on-chip peripheral modules are halted or the power-supply is turned off, through which low power consumption is achieved. These modes are canceled by a reset or interrupt.

25.1 Features

25.1.1 Power-Down Modes

This LSI has the following power-down modes and function:

1. Sleep mode
2. Software standby mode
3. Deep standby mode
4. Module standby function

Table 25.1 shows the transition conditions for entering the modes from the program execution state, as well as the CPU and peripheral module states in each mode and the procedures for canceling each mode.

Register Abbreviation	Bits 31/ 23/15/7	Bits30/ 22/14/6	Bits 29/ 21/13/5	Bits28/ 20/12/4	Bits 27/ 19/11/3	Bits26/ 18/10/2	Bits 25/ 17/9/1	Bits24/ 16/8/0	Module
DMCNTA7	—	—	MDSEL1	MDSEL0	—	—	DSEL1	DSEL0	DMAC
	—	—	—	—	—	—	STRG1	STRG0	
	—	—	—	—	—	BRLOD	SRLOD	DRLOD	
	—	—	DCTG5	DCTG4	DCTG3	DCTG2	DCTG1	DCTG0	
DMCNTB7	—	—	—	—	—	—	—	DEN	
	—	—	—	—	—	—	—	DREQ	
	—	—	—	—	—	—	—	ECLR	
	—	—	—	—	—	—	—	DSCLR	
DMSCNT	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	DMST	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
DMICNT	DINTM_CH0	DINTM_CH1	DINTM_CH2	DINTM_CH3	DINTM_CH4	DINTM_CH5	DINTM_CH6	DINTM_CH7	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
DMICNTA	DINTA_CH0	DINTA_CH1	DINTA_CH2	DINTA_CH3	DINTA_CH4	DINTA_CH5	DINTA_CH6	DINTA_CH7	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
DMISTS	DISTS_CH0	DISTS_CH1	DISTS_CH2	DISTS_CH3	DISTS_CH4	DISTS_CH5	DISTS_CH6	DISTS_CH7	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
DMEDET	DEDET_CH0	DEDET_CH1	DEDET_CH2	DEDET_CH3	DEDET_CH4	DEDET_CH5	DEDET_CH6	DEDET_CH7	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
DMASTS	DASTS_CH0	DASTS_CH1	DASTS_CH2	DASTS_CH3	DASTS_CH4	DASTS_CH5	DASTS_CH6	DASTS_CH7	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
BAR_0	BA0_31	BA0_30	BA0_29	BA0_28	BA0_27	BA0_26	BA0_25	BA0_24	UBC
	BA0_23	BA0_22	BA0_21	BA0_20	BA0_19	BA0_18	BA0_17	BA0_16	
	BA0_15	BA0_14	BA0_13	BA0_12	BA0_11	BA0_10	BA0_9	BA0_8	
	BA0_7	BA0_6	BA0_5	BA0_4	BA0_3	BA0_2	BA0_1	BA0_0	

Table 29.2 DC Characteristics (4) [I²C-Related Pins*]

Conditions: $PV_{CC} = V_{CC}R = PLLV_{CC} = 3.0$ V to 3.6 V, $AV_{CC} = 3.0$ V to 3.6 V,
$PV_{CC} - 0.3$ V $\leq AV_{CC} \leq PV_{CC}$, $AV_{ref} = 3.0$ V to AV_{CC},
$PV_{SS} = V_{SS}R = PLLV_{SS} = AV_{SS} = 0$ V

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	PC22/DREQ2, PC23, PC24, PC25, PD15, PD16	V_{IH}	2.2	—	PV_{CC} + 0.3	V	
Input low voltage		V_{IL}	–0.3	—	0.8	V	
Schmitt trigger input characteristics	IRQ0/SCL0, IRQ1/SDA0, IRQ2/SCL1, IRQ3/SDA1, SDA2, ASCL2	V_T^+ (V_{IH})	$PV_{CC} \times 0.7$	—	5.5	V	
		V_T^- (V_{IL})	–0.3	—	$PV_{CC} \times 0.3$	V	
		$V_T^+ - V_T^-$	$PV_{CC} \times 0.05$	—	—	V	
Output low voltage	SCL0 to SCL2, SDA0 to SDA2	V_{OL}	—	—	0.4	V	I_{OL} = 3.0 mA

Note: * Pins (open-drain pins): PC22/IRQ0/SCL0/DREQ2, PC23/IRQ1/SDA0, PC24/IRQ2/SCL1, PC25/IRQ3/SDA1, PD15/SDA2, and PD16/SCL2

Table 29.3 Permissible Output Currents (1) [Common Items]

Conditions: $PV_{CC} = V_{CC}R = PLLV_{CC} = 3.0$ V to 3.6 V, $AV_{CC} = 3.0$ V to 3.6 V,
$PV_{CC} - 0.3$ V $\leq AV_{CC} \leq PV_{CC}$, $AV_{ref} = 3.0$ V to AV_{CC},
$PV_{SS} = V_{SS}R = PLLV_{SS} = AV_{SS} = 0$ V

Item		Symbol	Min.	Typ.	Max.	Unit
Permissible output low current (per pin)	SCL0 to SCL2, SDA0 to SDA2	I_{OL}	—	—	10	mA
	Other than above				2	
Permissible output low current (total)		ΣI_{OL}	—	—	150	mA
Permissible output high current (per pin)		$-I_{OH}$	—	—	2	mA
Permissible output high current (total)		$\Sigma -I_{OH}$	—	—	50	mA

Caution: To protect the LSI's reliability, do not exceed the output current values in the table above.