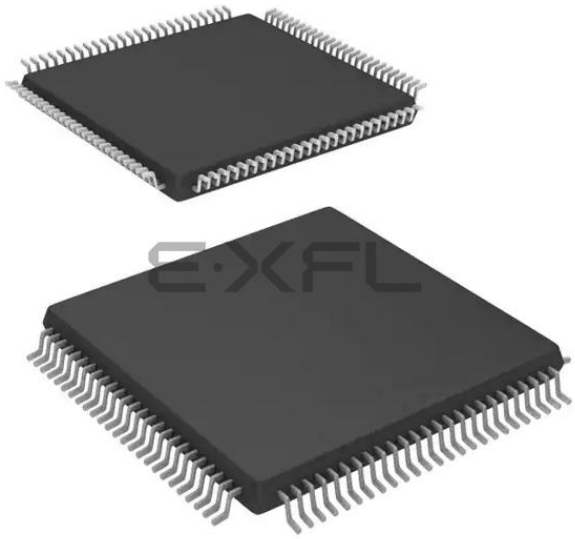

**Welcome to E-XFL.COM**

## 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 | H8S/2000 |
| Core Size | 16-Bit |
| Speed | 25MHz |
| Connectivity | SCI, SmartCard |
| Peripherals | POR, PWM, WDT |
| Number of I/O | 70 |
| Program Memory Size | 128KB (128K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 8K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.7V ~ 3.6V |
| Data Converters | A/D 8x10b; D/A 2x8b |
| Oscillator Type | Internal |
| Operating Temperature | -20°C ~ 75°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 100-TQFP |
| Supplier Device Package | 100-TQFP (14x14) |
| Purchase URL | https://www.e-xfl.com/product-detail/renesas-electronics-america/df2317vtebl25v |

### 5.2.2 Interrupt Priority Registers A to K (IPRA to IPRK)

| Bit | : | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| | | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 |
| Initial value | : | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| R/W | : | — | R/W | R/W | R/W | — | R/W | R/W | R/W |

The IPR registers are eleven 8-bit readable/writable registers that set priorities (levels 7 to 0) for interrupts other than NMI.

The correspondence between IPR settings and interrupt sources is shown in table 5.3.

The IPR registers set a priority (levels 7 to 0) for each interrupt source other than NMI.

The IPR registers are initialized to H'77 by a reset and in hardware standby mode.

**Bits 7 and 3—Reserved:** Read-only bits, always read as 0.

**Table 5.3 Correspondence between Interrupt Sources and IPR Settings**

| | Bits | |
|---|---|---|
| **Register** | **6 to 4** | **2 to 0** |
| IPRA | IRQ0 | IRQ1 |
| IPRB | IRQ2<br>IRQ3 | IRQ4<br>IRQ5 |
| IPRC | IRQ6<br>IRQ7 | DTC |
| IPRD | Watchdog timer | —* |
| IPRE | —* | A/D converter |
| IPRF | TPU channel 0 | TPU channel 1 |
| IPRG | TPU channel 2 | TPU channel 3 |
| IPRH | TPU channel 4 | TPU channel 5 |
| IPRI | 8-bit timer channel 0 | 8-bit timer channel 1 |
| IPRJ | —* | SCI channel 0 |
| IPRK | SCI channel 1 | —* |

Note: * Reserved bits.

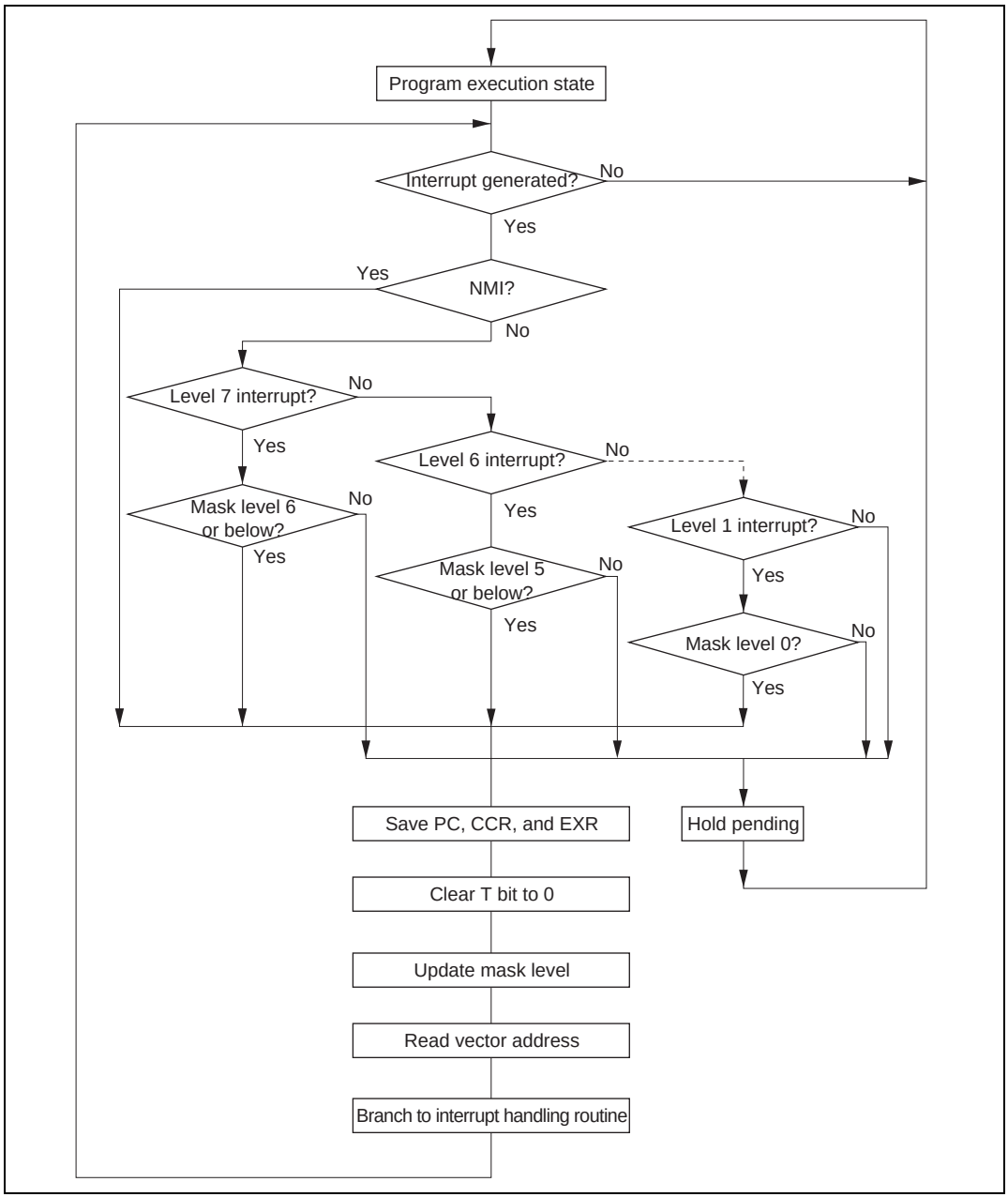


**Figure 5.6 Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 2**

**Bit 7—DTC Software Activation Enable (SWDTE):** Enables or disables DTC activation by software.

| Bit 7 SWDTE | Description |
|---|---|
| 0 | DTC software activation is disabled (Initial value)<br>[Clearing conditions]<br>• When the DISEL bit is 0 and the specified number of transfers have not ended<br>• When 0 is written after a software activation data-transfer-complete interrupt is issued to the CPU |
| 1 | DTC software activation is enabled<br>[Holding conditions]<br>• When the DISEL bit is 1 and data transfer has ended<br>• When the specified number of transfers have ended<br>• During data transfer due to software activation |

**Bits 6 to 0—DTC Software Activation Vectors 6 to 0 (DTVEC6 to DTVEC0):** These bits specify a vector number for DTC software activation.

The vector address is expressed as H'0400 + ((vector number) << 1). <<1 indicates a one-bit left-shift. For example, when DTVEC6 to DTVEC0 = H'10, the vector address is H'0420.

### 7.2.9 Module Stop Control Register (MSTPCR)

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

MSTPCR is a 16-bit readable/writable register that performs module stop mode control.

When the MSTP14 bit in MSTPCR is set to 1, DTC operation stops at the end of the bus cycle and a transition is made to module stop mode. However, 1 cannot be written in the MSTP14 bit while the DTC is operating. For details, see section 19.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not initialized in software standby mode.

## 7.4 Interrupts

An interrupt request is issued to the CPU when the DTC finishes the specified number of data transfers, or a data transfer for which the DISEL bit was set to 1. In the case of interrupt activation, the interrupt set as the activation source is generated. These interrupts to the CPU are subject to CPU mask level and interrupt controller priority level control.

In the case of activation by software, a software activated data transfer end interrupt (SWDTEND) is generated.

When the DISEL bit is 1 and one data transfer has ended, or the specified number of transfers have ended, after data transfer ends, the SWDTE bit is held at 1 and an SWDTEND interrupt is generated. The interrupt handling routine should clear the SWDTE bit to 0.

When the DTC is activated by software, an SWDTEND interrupt is not generated during a data transfer wait or during data transfer even if the SWDTE bit is set to 1.

## 7.5 Usage Notes

**Module Stop:** When the MSTP14 bit in MSTPCR is set to 1, the DTC clock stops, and the DTC enters the module stop state. However, 1 cannot be written to the MSTP14 bit while the DTC is operating.

**On-Chip RAM:** The MRA, MRB, SAR, DAR, CRA, and CRB registers are all located in on-chip RAM. When the DTC is used, the RAME bit in SYSCR must not be cleared to 0.

**DTCE Bit Setting:** For DTCE bit setting, read/write operations must be performed using bit-manipulation instructions such as BSET and BCLR. For the initial setting only, however, when multiple activation sources are set at one time, it is possible to disable interrupts and write after executing a dummy read on the relevant register.

**Chain Transfer:** When chain transfer is used, clearing of the activation source or DTCER is performed when the last of the chain of data transfers is executed. SCI and A/D converter interrupt/activation sources, on the other hand, are cleared when the DTC reads or writes to the prescribed register.

Therefore, when the DTC is activated by an interrupt or activation source, if a read/write of the relevant register is not included in the last chained data transfer, the interrupt or activation source will be retained.

### 8.8.2 Register Configuration

Table 8.13 shows the port C register configuration.

**Table 8.13 Port C Registers**

| Name | Abbreviation | R/W | Initial Value | Address* |
|---|---|---|---|---|
| Port C data direction register | PCDDR | W | H'00 | H'FEBB |
| Port C data register | PCDR | R/W | H'00 | H'FF6B |
| Port C register | PORTC | R | Undefined | H'FF5B |
| Port C MOS pull-up control register | PCPCR | R/W | H'00 | H'FF72 |

Note: * Lower 16 bits of the address.

**Port C Data Direction Register (PCDDR)**

| Bit | : | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| | | PC7DDR | PC6DDR | PC5DDR | PC4DDR | PC3DDR | PC2DDR | PC1DDR | PC0DDR |
| Initial value | : | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | : | W | W | W | W | W | W | W | W |

PCDDR is an 8-bit write-only register, the individual bits of which specify input or output for the pins of port C. PCDDR cannot be read; if it is, an undefined value will be read.

PCDDR is initialized to H'00 by a reset, and in hardware standby mode. It retains its prior state in software standby mode. The OPE bit in SBYCR is used to select whether the address output pins retain their output state or become high-impedance when a transition is made to software standby mode.

- Modes 4 and 5
  The corresponding port C pins are address outputs irrespective of the value of the PCDDR bits.
- Mode 6*
  Setting PCDDR bits to 1 makes the corresponding port C pin address outputs, while clearing the bits to 0 makes the pins input ports.
- Mode 7*
  Setting PCDDR bits to 1 makes the corresponding port C pins an output ports, while clearing the bits to 0 makes the pins input ports.

Note: * Modes 6 and 7 are not available in the ROMless versions.

### 9.2.4 Timer Interrupt Enable Registers (TIER)

**Channel 0: TIER0**

**Channel 3: TIER3**

| Bit | : | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| | | TTGE | — | — | TCIEV | TGIED | TGIEC | TGIEB | TGIEA |
| Initial value | : | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | : | R/W | — | — | R/W | R/W | R/W | R/W | R/W |

**Channel 1: TIER1**

**Channel 2: TIER2**

**Channel 4: TIER4**

**Channel 5: TIER5**

| Bit | : | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| | | TTGE | — | TCIEU | TCIEV | — | — | TGIEB | TGIEA |
| Initial value | : | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | : | R/W | — | R/W | R/W | — | — | R/W | R/W |

The TIER registers are 8-bit registers that control enabling or disabling of interrupt requests for each channel. The TPU has six TIER registers, one for each channel. The TIER registers are initialized to H'40 by a reset and in hardware standby mode.

**Bit 7—A/D Conversion Start Request Enable (TTGE):** Enables or disables generation of A/D conversion start requests by TGRA input capture/compare match.

| Bit 7 TTGE | Description | |
|---|---|---|
| 0 | A/D conversion start request generation disabled | (Initial value) |
| 1 | A/D conversion start request generation enabled | |

**Bit 6—Reserved:** This bit cannot be modified and is always read as 1.

**Contention between TGR Write and Compare Match:** If a compare match occurs in the $T_2$ state of a TGR write cycle, the TGR write takes precedence and the compare match signal is inhibited. A compare match does not occur even if the same value as before is written.

Figure 9.51 shows the timing in this case.

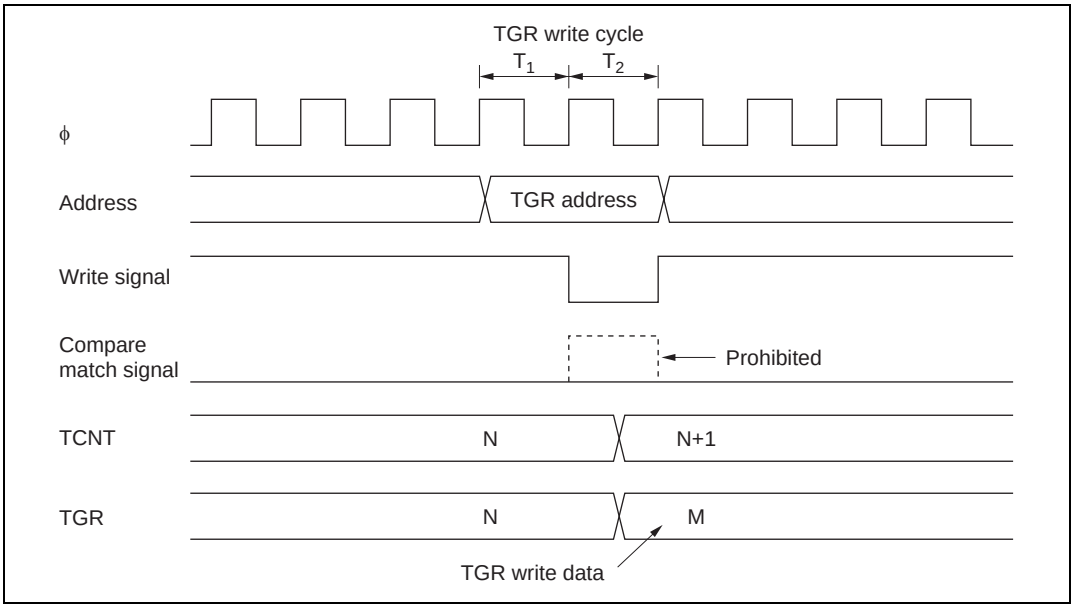


**Figure 9.51 Contention between TGR Write and Compare Match**

**Bit 6—Receive Interrupt Enable (RIE):** Enables or disables receive-data-full interrupt (RXI) request and receive-error interrupt (ERI) request generation when serial receive data is transferred from RSR to RDR and the RDRF flag in SSR is set to 1.

| Bit 6 RIE | Description | |
|---|---|---|
| 0 | Receive-data-full interrupt (RXI) request and receive-error interrupt (ERI) request disabled* | (Initial value) |
| 1 | Receive-data-full interrupt (RXI) request and receive-error interrupt (ERI) request enabled | |

Note:* RXI and ERI interrupt request cancellation can be performed by reading 1 from the RDRF flag, or the FER, PER, or ORER flag, then clearing the flag to 0, or by clearing the RIE bit to 0.

**Bit 5—Transmit Enable (TE):** Enables or disables the start of serial transmission by the SCI.

| Bit 5 TE | Description | |
|---|---|---|
| 0 | Transmission disabled*1 | (Initial value) |
| 1 | Transmission enabled*2 | |

Notes: 1. The TDRE flag in SSR is fixed at 1.

2. In this state, serial transmission is started when transmit data is written to TDR and the TDRE flag in SSR is cleared to 0.

   SMR setting must be performed to decide the transfer format before setting the TE bit to 1.

**Bit 4—Receive Enable (RE):** Enables or disables the start of serial reception by the SCI.

| Bit 4 RE | Description | |
|---|---|---|
| 0 | Reception disabled*1 | (Initial value) |
| 1 | Reception enabled*2 | |

Notes: 1. Clearing the RE bit to 0 does not affect the RDRF, FER, PER, and ORER flags, which retain their states.

2. Serial reception is started in this state when a start bit is detected in asynchronous mode or serial clock input is detected in synchronous mode.

   SMR setting must be performed to decide the transfer format before setting the RE bit to 1.

**Table 12.8 SMR Settings and Serial Transfer Format Selection**

| SMR Settings | | | | | | SCI Transfer Format | | | |
|---|---|---|---|---|---|---|---|---|---|
| Bit 7 | Bit 6 | Bit 2 | Bit 5 | Bit 3 | | Data Length | Multi-processor Bit | Parity Bit | Stop Bit Length |
| C/$\overline{A}$ | CHR | MP | PE | STOP | Mode | | | | |
| 0 | 0 | 0 | 0 | 0 | Asynchronous mode | 8-bit data | No | No | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | | | 1 | 0 | | | | Yes | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | 1 | | 0 | 0 | | 7-bit data | | No | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | | | 1 | 0 | | | | Yes | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | 0 | 1 | — | 0 | Asynchronous mode (multi-processor format) | 8-bit data | Yes | No | 1 bit |
| | | | — | 1 | | | | | 2 bits |
| | 1 | | — | 0 | | 7-bit data | | | 1 bit |
| | | | — | 1 | | | | | 2 bits |
| 1 | — | — | — | — | Synchronous mode | 8-bit data | No | | None |

**Table 12.9 SMR and SCR Settings and SCI Clock Source Selection**

| SMR | SCR Settings | | | SCI Transmit/Receive Clock | |
|---|---|---|---|---|---|
| Bit 7 | Bit 1 | Bit 0 | | Clock Source | SCK Pin Function |
| C/$\overline{A}$ | CKE1 | CKE0 | Mode | | |
| 0 | 0 | 0 | Asynchronous mode | Internal | SCI does not use SCK pin |
| | | 1 | | | Outputs clock with same frequency as bit rate |
| | 1 | 0 | | External | Inputs clock with frequency of 16 times the bit rate |
| | | 1 | | | |
| 1 | 0 | 0 | Synchronous mode | Internal | Outputs serial clock |
| | | 1 | | | |
| | 1 | 0 | | External | Inputs serial clock |
| | | 1 | | | |

The operation sequence is as follows.

[1] When the data line is not in use it is in the high-impedance state, and is fixed high with a pull-up resistor.

[2] The transmitting station starts transfer of one frame of data. The data frame starts with a start bit (Ds, low-level), followed by 8 data bits (D0 to D7) and a parity bit (Dp).

[3] With the smart card interface, the data line then returns to the high-impedance state. The data line is pulled high with a pull-up resistor.

[4] The receiving station carries out a parity check.

If there is no parity error and the data is received normally, the receiving station waits for reception of the next data.

If a parity error occurs, however, the receiving station outputs an error signal (DE, low-level) to request retransmission of the data. After outputting the error signal for the prescribed length of time, the receiving station places the signal line in the high-impedance state again. The signal line is pulled high again by a pull-up resistor.

[5] If the transmitting station does not receive an error signal, it proceeds to transmit the next data frame.

If it does receive an error signal, however, it returns to step [2] and retransmits the data in which the error occurred.

**Block Transfer Mode:** The operation sequence in block transfer mode is as follows.

[1] When the data line is not in use it is in the high-impedance state, and is fixed high with a pull-up resistor.

[2] The transmitting station starts transfer of one frame of data. The data frame starts with a start bit (Ds, low-level), followed by 8 data bits (D0 to D7) and a parity bit (Dp).

[3] With the smart card interface, the data line then returns to the high-impedance state. The data line is pulled high with a pull-up resistor.

[4] The receiving station carries out a parity check, but does not output an error signal even if an error has occurred. Since subsequent receive operations cannot be carried out if an error occurs, the error flag must be cleared to 0 before the parity bit for the next frame is received.

[5] The transmitting station proceeds to transmit the next data frame.

## 17.7 Programming/Erasing Flash Memory

In the on-board programming modes, flash memory programming and erasing is performed by software, using the CPU. There are four flash memory operating modes: program mode, erase mode, program-verify mode, and erase-verify mode. Transition to these modes can be made for the on-chip ROM area by setting the PSU, ESU, P, E, PV, and EV bits in FLMCR1.

The flash memory cannot be read while being programmed or erased. Therefore, the program that controls flash memory programming/erasing (the programming control program) should be located and executed in on-chip RAM or external memory. When the program is located in external memory, an instruction for programming the flash memory and the following instruction should be located in on-chip RAM. The DTC should not be activated before or after the instruction for programming the flash memory is executed.

Notes: 1. Operation is not guaranteed if setting/resetting of the SWE, ESU, PSU, EV, PV, E, and P bits in FLMCR1 is executed by a program in flash memory.
2. When programming or erasing, set FWE to 1 (programming/erasing will not be executed if FWE = 0).
3. Perform programming in the erased state. Do not perform additional programming on previously programmed addresses.

### 17.7.1 Program Mode

Follow the procedure shown in the program/program-verify flowchart in figure 17.15 to write data or programs to flash memory. Performing program operations according to this flowchart will enable data or programs to be written to flash memory without subjecting the device to voltage stress or sacrificing program data reliability. Programming should be carried out 128 bytes at a time.

For the wait times (x, y, z1, z2, z3, $\alpha$, ß, $\gamma$, $\varepsilon$, $\eta$, $\theta$) after bits are set or cleared in flash memory control register 1 (FLMCR1) and the maximum number of programming operations (N), see section 20.3.6, Flash Memory Characteristics.

Following the elapse of (x) $\mu$s or more after the SWE bit is set to 1 in flash memory control register 1 (FLMCR1), 128-byte program data is stored in the program data area and reprogram data area, and the 128-byte data in the reprogram data area is written consecutively to the write addresses. The lower 8 bits of the first address written to must be H'00 or H'80. 128 consecutive byte data transfers are performed. The program address and program data are latched in the flash memory. A 128-byte data transfer must be performed even if writing fewer than 128 bytes; in this case, H'FF data must be written to the extra addresses.

| Response | H'06 |
|---|---|

- Response, H'06, (1 byte): Response to selection of a new bit rate
  When it is possible to set the bit rate, the response will be ACK.

| Error Response | H'BF | ERROR |
|---|---|---|

- Error response, H'BF, (1 byte): Error response to selection of new bit rate
- ERROR: (1 byte): Error code
  H'11: Sum checking error
  H'24: Bit-rate selection error
  The rate is not available.
  H'25: Error in input frequency
  This input frequency is not within the specified range.
  H'26: Multiplication-ratio error*
  The ratio does not match an available ratio.
  H'27: Operating frequency error*
  The frequency is not within the specified range.

Note: * This error does not occur with this LSI.

**Received Data Check**

The methods for checking of received data are listed below.

**(1) Input frequency**

The received value of the input frequency is checked to ensure that it is within the range of minimum to maximum frequencies which matches the clock modes of the specified device. When the value is out of this range, an input-frequency error is generated.

**(2) Multiplication ratio**

The received value of the multiplication ratio or division ratio is checked to ensure that it matches the clock modes of the specified device. When the value is out of this range, an input-frequency error is generated.

Transitions to the reset state, and hardware standby mode are inhibited during programming/erasing. When the reset signal is accidentally input to the chip, a longer period in the reset state than usual (100 μs) is needed before the reset signal is released.

(7) Switching of the MATs by FMATS should be needed when programming/erasing of the user boot MAT is operated in user-boot mode. The program which switches the MATs should be executed from the on-chip RAM. See section 17.27, Switching between User MAT and User Boot MAT. Please make sure you know which MAT is selected when switching between them.

(8) When the data storable area indicated by programming parameter FMPDR is within the flash memory area, an error will occur even when the data stored is normal. Therefore, the data should be transferred to the on-chip RAM to place the address that FMPDR indicates in an area other than the flash memory.

In consideration of these conditions, there are three factors; operating mode, the bank structure of the user MAT, and operations.

The areas in which the programming data can be stored for execution are shown in table 17.26.

**Table 17.72 Executable MAT**

| | Initiated Modes | |
|---|---|---|
| **Operation** | **User Program Mode** | **User Boot Mode*** |
| Programming | Table 17.73 (1) | Table 17.73 (3) |
| Erasing | Table 17.73 (2) | Table 17.73 (4) |

Note: * Programming/Erasing is possible to user MATs.

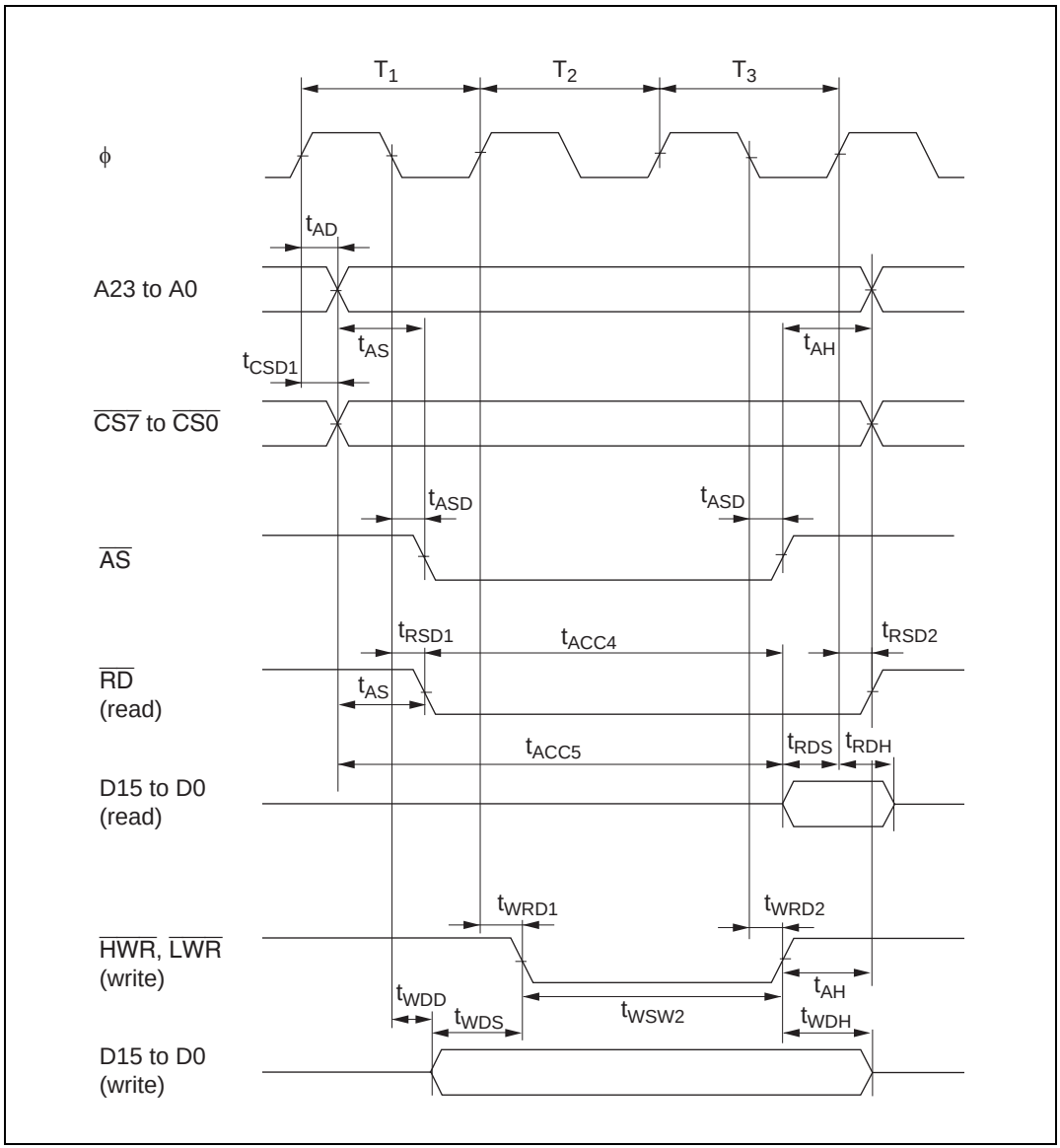


**Figure 20.7 Basic Bus Timing (3-State Access)**

| Item | | | Symbol | Condition A Min | Condition A Max | Condition B Min | Condition B Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|---|---|---|
| SCI | Input clock cycle | Asynchronous | $t_{Scyc}$ | 4 | — | 4 | — | $t_{cyc}$ | Figure 20.20 |
| | | Synchronous | | 6 | — | 6 | — | | |
| | Input clock pulse width | | $t_{SCKW}$ | 0.4 | 0.6 | 0.4 | 0.6 | $t_{Scyc}$ | |
| | Input clock rise time | | $t_{SCKr}$ | — | 1.5 | — | 1.5 | $t_{cyc}$ | |
| | Input clock fall time | | $t_{SCKf}$ | — | 1.5 | — | 1.5 | $t_{cyc}$ | |
| | Transmit data delay time | | $t_{TXD}$ | — | 50 | — | 40 | ns | Figure 20.21 |
| | Receive data setup time (synchronous) | | $t_{RXS}$ | 50 | — | 40 | — | ns | |
| | Receive data hold time (synchronous) | | $t_{RXH}$ | 50 | — | 40 | — | ns | |
| A/D converter | Trigger input setup time | | $t_{TRGS}$ | 30 | — | 30 | — | ns | Figure 20.22 |

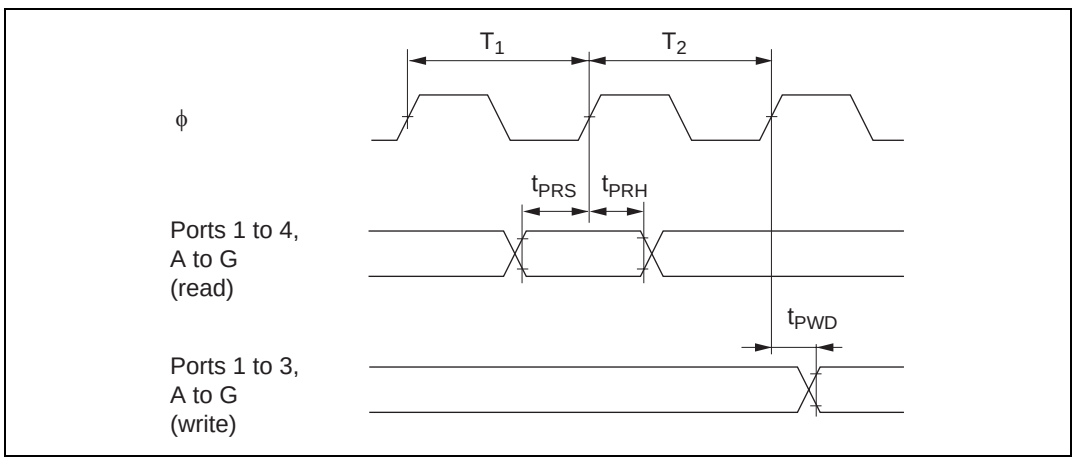


**Figure 20.13 I/O Port Input/Output Timing**

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|---|
| Input pull-up MOS current | Ports A to E | $-I_p$ | 10 | — | 300 | µA | $V_{CC}$ = 3.0 V to 3.6 V, $V_{in}$ = 0 V |
| Input capacitance | $\overline{RES}$ | $C_{in}$ | — | — | 30 | pF | $V_{in}$ = 0 V |
| | NMI | | — | — | 30 | pF | f = 1 MHz |
| | All input pins except $\overline{RES}$ and NMI | | — | — | 15 | pF | $T_a$ = 25°C |
| Current dissipation*2 | Normal operation | $I_{CC}$*4 | — | 25 (3.3 V) | 50 | mA | f = 25 MHz |
| | Sleep mode | | | 17 (3.3 V) | 40 | mA | |
| | Standby mode*3 | | — | 20 | 90 | µA | $T_a \leq 50°C$ |
| | | | — | — | 120 | µA | $50°C < T_a$ |
| Analog power supply voltage | During A/D and D/A conversion | $AI_{CC}$ | — | 1.0 (3.0 V) | 2.0 | mA | |
| | Idle | | — | 1.0 | 5.0 | µA | |
| Reference power supply voltage | During A/D and D/A conversion | $AI_{CC}$ | — | 1.4 (3.0 V) | 3.0 | mA | |
| | Idle | | — | 0.2 | 5.0 | µA | |
| RAM standby voltage | | $V_{RAM}$ | 2.5 | — | — | V | |
| VCC start voltage*5 | | $VCC_{START}$ | — | — | 0.4 | V | |
| VCC rising edge*5 | | SVCC | — | — | 10 | ms/V | |

Notes: 1. **If the A/D and D/A converters are not used, do not leave the $AV_{CC}$, $V_{ref}$, and $AV_{SS}$ pins open**. Connect the $AV_{CC}$ and $V_{ref}$ pins to $V_{CC}$, and the $AV_{SS}$ pin to $V_{SS}$.

2. Current dissipation values are for $V_{IH\ min} = V_{CC} - 0.2$ V and $V_{IL\ max} = 0.2$ V with all output pins unloaded and all MOS input pull-ups in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 3.0$ V, $V_{IH}$ min = $V_{CC} \times 0.9$, and $V_{IL}$ max = 0.3 V.
4. $I_{CC}$ depends on $V_{CC}$ and f as follows:
   $I_{CC}$ max = 0.5 (mA) + 0.55 (mA/(MHz × V)) × $V_{CC}$ × f (normal operation)
   $I_{CC}$ max = 0.4 (mA) + 0.44 (mA/(MHz × V)) × $V_{CC}$ × f (sleep mode)
5. Applies on condition that the $\overline{RES}$ pin is low level at power on.

| Module | Register | Abbreviation | R/W | Initial Value | Address[*1] |
|---|---|---|---|---|---|
| Bus controller | Bus width control register | ABWCR | R/W | H'FF/H'00[*5] | H'FED0 |
| | Access state control register | ASTCR | R/W | H'FF | H'FED1 |
| | Wait control register H | WCRH | R/W | H'FF | H'FED2 |
| | Wait control register L | WCRL | R/W | H'FF | H'FED3 |
| | Bus control register H | BCRH | R/W | H'D0 | H'FED4 |
| | Bus control register L | BCRL | R/W | H'3C | H'FED5 |
| 8-bit timer 0 | Timer control register 0 | TCR0 | R/W | H'00 | H'FFB0 |
| | Timer control/status register 0 | TCSR0 | R/(W)[*7] | H'00 | H'FFB2 |
| | Timer constant register A0 | TCORA0 | R/W | H'FF | H'FFB4 |
| | Timer constant register B0 | TCORB0 | R/W | H'FF | H'FFB6 |
| | Timer counter 0 | TCNT0 | R/W | H'00 | H'FFB8 |
| 8-bit timer 1 | Timer control register 1 | TCR1 | R/W | H'00 | H'FFB1 |
| | Timer control/status register 1 | TCSR1 | R/(W)[*7] | H'10 | H'FFB3 |
| | Timer constant register A1 | TCORA1 | R/W | H'FF | H'FFB5 |
| | Timer constant register B1 | TCORB1 | R/W | H'FF | H'FFB7 |
| | Timer counter 1 | TCNT1 | R/W | H'00 | H'FFB9 |
| All 8-bit timer channels | Module stop control register | MSTPCR | R/W | H'3FFF | H'FF3C |
| WDT | Timer control/status register | TCSR | R/(W)[*9] | H'18 | H'FFBC: Write[*8]<br>H'FFBC: Read |
| | Timer counter | TCNT | R/W | H'00 | H'FFBC: Write[*6]<br>H'FFBD: Read |
| | Reset control/status register | RSTCSR | R/(W)[*9] | H'1F | H'FFBE: Write[*8]<br>H'FFBF: Read |

| | | |
|---|---|---|
| **IPRA—Interrupt Priority Register A** | **H'FEC4** | **Interrupt Controller** |
| **IPRB—Interrupt Priority Register B** | **H'FEC5** | **Interrupt Controller** |
| **IPRC—Interrupt Priority Register C** | **H'FEC6** | **Interrupt Controller** |
| **IPRD—Interrupt Priority Register D** | **H'FEC7** | **Interrupt Controller** |
| **IPRE—Interrupt Priority Register E** | **H'FEC8** | **Interrupt Controller** |
| **IPRF—Interrupt Priority Register F** | **H'FEC9** | **Interrupt Controller** |
| **IPRG—Interrupt Priority Register G** | **H'FECA** | **Interrupt Controller** |
| **IPRH—Interrupt Priority Register H** | **H'FECB** | **Interrupt Controller** |
| **IPRI—Interrupt Priority Register I** | **H'FECC** | **Interrupt Controller** |
| **IPRJ—Interrupt Priority Register J** | **H'FECD** | **Interrupt Controller** |
| **IPRK—Interrupt Priority Register K** | **H'FECE** | **Interrupt Controller** |

| Bit : | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 |
| Initial value : | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| Read/Write : | — | R/W | R/W | R/W | — | R/W | R/W | R/W |

Set priority (levels 7 to 0) for interrupt sources

Correspondence between Interrupt Sources and IPR Settings

| Register | Bits | |
|---|---|---|
| | 6 to 4 | 2 to 0 |
| IPRA | IRQ0 | IRQ1 |
| IPRB | IRQ2 | IRQ4 |
| | IRQ3 | IRQ5 |
| IPRC | IRQ6 | DTC |
| | IRQ7 | |
| IPRD | WDT | —* |
| IPRE | —* | A/D converter |
| IPRF | TPU channel 0 | TPU channel 1 |
| IPRG | TPU channel 2 | TPU channel 3 |
| IPRH | TPU channel 4 | TPU channel 5 |
| IPRI | 8-bit timer channel 0 | 8-bit timer channel 1 |
| IPRJ | —* | SCI channel 0 |
| IPRK | SCI channel 1 | —* |

Note: * Reserved bits.

## TCR1—Timer Control Register 1 H'FFE0 TPU1

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Time Prescaler

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock: counts on $\phi$/1 |
| | | 1 | Internal clock: counts on $\phi$/4 |
| | 1 | 0 | Internal clock: counts on $\phi$/16 |
| | | 1 | Internal clock: counts on $\phi$/64 |
| 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | 1 | External clock: counts on TCLKB pin input |
| | 1 | 0 | Internal clock: counts on $\phi$/256 |
| | | 1 | Counts on TCNT2 overflow/underflow |

Note: This setting is ignored when channel 1 is in phase counting mode.

Clock Edge*

| | | |
|---|---|---|
| 0 | 0 | Count at rising edge |
| | 1 | Count at falling edge |
| 1 | — | Count at both edges |

Note: * This setting is ignored when channel 1 is in phase counting mode. The internal clock edge selection is valid when the input clock is $\phi$/4 or slower. This setting is ignored if $\phi$/1 or overflow/underflow on another channel is selected as the input clock.

Counter Clear

| | | |
|---|---|---|
| 0 | 0 | TCNT clearing disabled |
| | 1 | TCNT cleared by TGRA compare match/input capture |
| 1 | 0 | TCNT cleared by TGRB compare match/input capture |
| | 1 | TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation* |

Note: * Synchronous operation setting is performed by setting the SYNC bit in TSYR to 1.

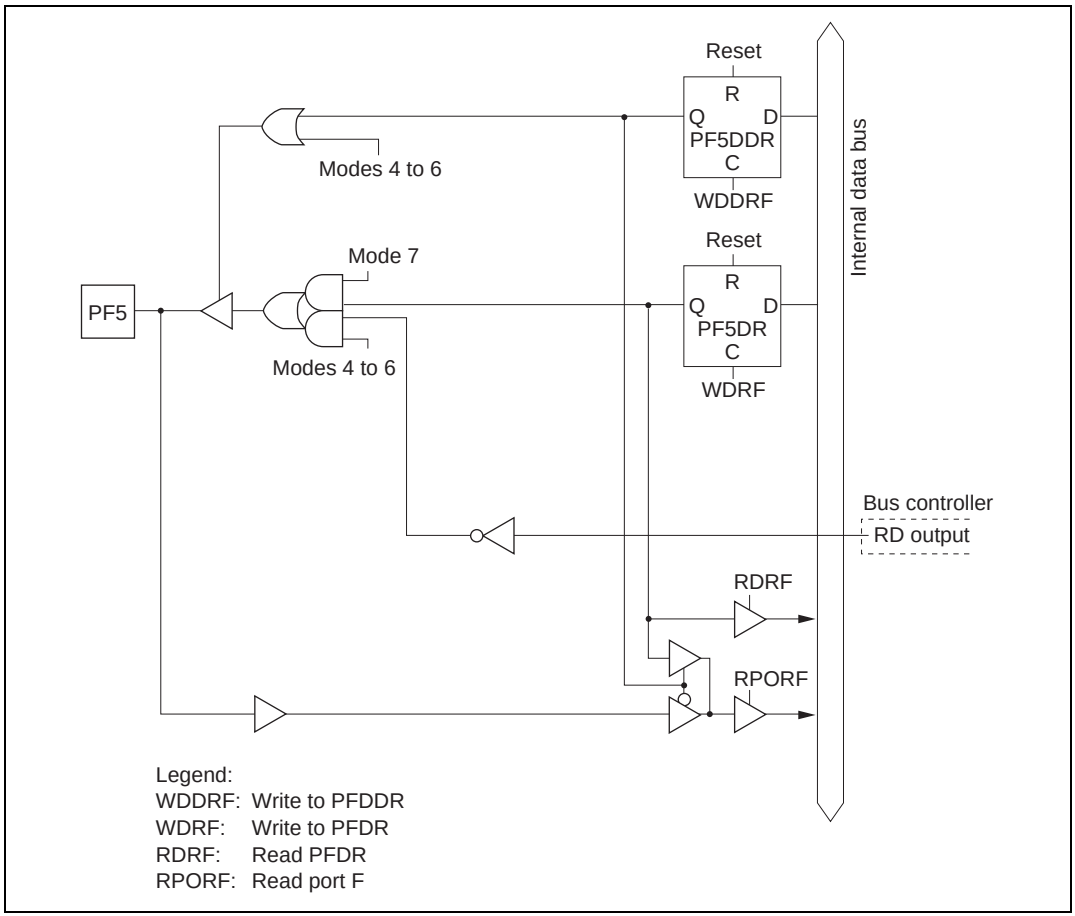


**Figure C.10(f) Port F Block Diagram (Pin PF5)**