# Renesas Electronics America Inc - HD64F7044F28V Datasheet

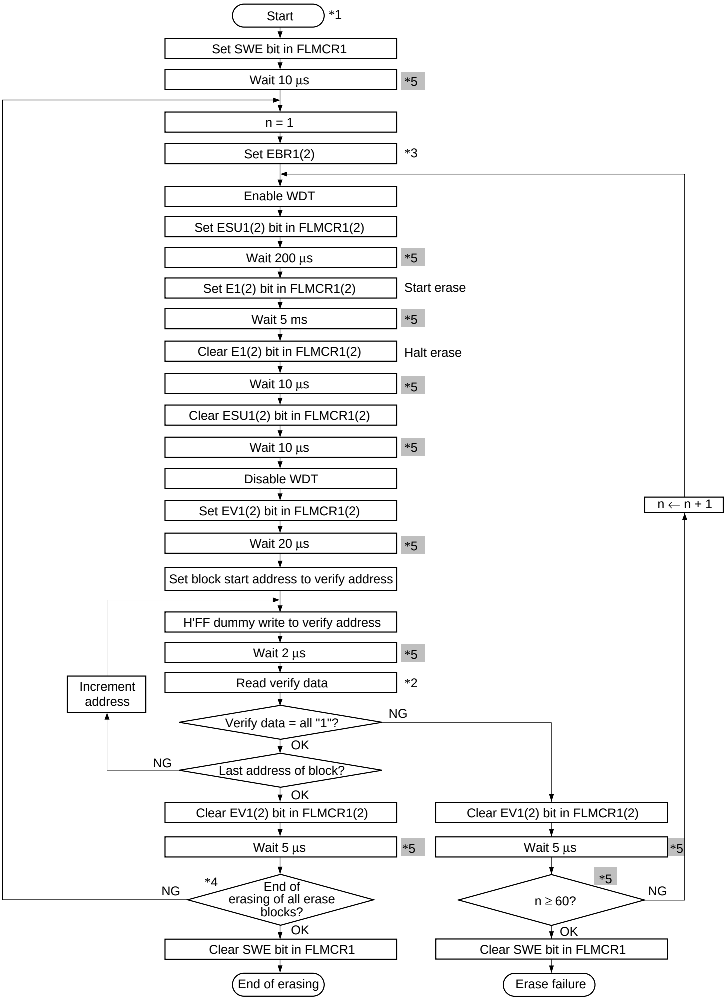
## What is "Embedded - Microcontrollers"?

"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 | Obsolete |
| Core Processor | SH-2 |
| Core Size | 32-Bit Single-Core |
| Speed | 28.7MHz |
| Connectivity | EBI/EMI, SCI |
| Peripherals | DMA, POR, PWM, WDT |
| Number of I/O | 74 |
| Program Memory Size | 256KB (256K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 4K x 8 |
| Voltage - Supply (Vcc/Vdd) | 4.5V ~ 5.5V |
| Data Converters | A/D 8x10b |
| Oscillator Type | Internal |
| Operating Temperature | -20°C ~ 75°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 112-BQFP |
| Supplier Device Package | 112-QFP (20x20) |
| Purchase URL | https://www.e-xfl.com/product-detail/renesas-electronics-america/hd64f7044f28v |

<table>
<tr><th>Section</th><th>Page</th><th>Description</th></tr>
<tr><td>22.7.4 Erase-Verify Mode<br>Figure 22.14 Erase/Erase-Verify Flowchart</td><td>713</td><td>Figure amended



Notes: *1 Preprogramming (setting erase block data to all “0”) is not necessary.<br>
*2 Verify data is read in 32-bit (longword) units.<br>
*3 Set only one bit in EBR1(2). More than one bit cannot be set.<br>
*4 Erasing is performed in block units. To erase a number of blocks, each block must be erased in turn.<br>
*5 Make sure to set the wait times and repetitions as specified. Erasing may not complete correctly if values other than the specified ones are used.</td></tr>
<tr><td>24.4.2 Canceling the Standby Mode</td><td>747</td><td>Cancellation by a Manual Reset deleted</td></tr>
<tr><td>25. Electrical Characteristics (5V, 33.3 MHz Version)</td><td>—</td><td>Deleted</td></tr>
</table>

  - Instruction length: 16-bit fixed length for improved code efficiency
  - Load-store architecture (basic operations are executed between registers)
  - Delayed branch instructions reduce pipeline disruption during branch
  - Instruction set based on C language
- Instruction execution time: one instruction/cycle (35 ns/instruction at 28.7-MHz operation)
- Address space: Architecture supports 4 Gbytes
- On-chip multiplier: multiplication operations (32 bits × 32 bits → 64 bits) and multiplication/accumulation operations (32 bits × 32 bits + 64 bits → 64 bits) executed in two to four cycles
- Five-stage pipeline

**Cache Memory:**

- 1-kbyte instruction cache
- Caching of instruction codes and PC relative read data
- 4-byte line length (1 longword: 2 instruction lengths)
- 256 entry cache tags
- Direct map method
- On-chip ROM/RAM, and on-chip I/O areas not objects of cache
- Used in common with on-chip RAM; 2 kbytes of on-chip RAM used as address array/data array when cache is enabled

**Interrupt Controller (INTC):**

- Nine external interrupt pins (NMI, $\overline{\text{IRQ0}}$–$\overline{\text{IRQ7}}$)
- Forty-three internal interrupt sources (forty-four for A mask)
- Sixteen programmable priority levels

**User Break Controller (UBC):**

- Generates an interrupt when the CPU or DMAC generates a bus cycle with specified conditions
- Simplifies configuration of an on-chip debugger

**Bus State Controller (BSC):**

- Supports external extended memory access
  - 16-bit (QFP-112, TQFP-120), or 32-bit (QFP-144) external data bus
- Memory address space divided into five areas (four areas of SRAM space, one area of DRAM space) with the following settable features:
  - Bus size (8, 16, or 32 bits)
  - Number of wait cycles

  - — Outputs chip-select signals for each area
  - — During DRAM space access:
    - Outputs $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ signals for DRAM
    - Can generate a RAS precharge time assurance Tp cycle
- DRAM burst access function
  - — Supports high-speed access mode for DRAM
- DRAM refresh function
  - — Programmable refresh interval
  - — Supports CAS-before-RAS refresh and self-refresh modes
- Wait cycles can be inserted using an external $\overline{\text{WAIT}}$ signal
- Address data multiplex I/O devices can be accessed

**Direct Memory Access Controller (DMAC) (4 Channels):**

- Supports cycle-steal transfers
- Supports dual address transfer mode
- Can be switched between direct and indirect transfer modes (channel 3 only)
  - — Direct transfer mode: transfers the data at the transfer source address to the transfer destination address
  - — Indirect transfer mode: regards the data at the transfer source address as an address and transfers the data at that address to the transfer destination address

**Data Transfer Controller (DTC):**

- Data transfer independent of the CPU possible through peripheral I/O interrupt requests
- Transfer mode can be set for each interrupt factor (transfer mode set in memory)
- Multiple data transfers possible for one activating factor
- Abundant transfer modes
  - — Normal mode/repeat mode/block transfer mode selectable
- Transfer unit can be set to byte/word/longword
- Interrupts activating the DTC requested of the CPU
  - — Interrupts can be generated to the CPU after completion of one data transfer
  - — Interrupts can be generated to the CPU after completing all designated data transfers
- Transfer can be activated by software

**Multifunction Timer/Pulse Unit (MTU):**

- Maximum 16 types of waveform output or maximum 16 types of pulse I/O processing possible based on 16-bit timer, 5 channels
- 16 dual-use output compare/input capture registers

### Table 1.4 Pin Arrangement by Mode for SH7041, SH7043 (QFP-144 Pin) (cont)

| Pin No. | MCU Mode | PROM Mode |
|---|---|---|
| 121 | PF3/AN3 | $V_{SS}$ |
| 122 | PF4/AN4 | $V_{SS}$ |
| 123 | PF5/AN5 | $V_{SS}$ |
| 124 | $AV_{SS}$ | $V_{SS}$ |
| 125 | PF6/AN6 | $V_{SS}$ |
| 126 | PF7/AN7 | $V_{SS}$ |
| 127 | AVref | $V_{CC}$ |
| 128 | $AV_{CC}$ | $V_{CC}$ |
| 129 | $V_{SS}$ | $V_{SS}$ |
| 130 | PA0/RXD0 | NC |
| 131 | PA1/TXD0 | NC |
| 132 | PA2/SCK0/$\overline{\text{DREQ0}}$ /$\overline{\text{IREQ0}}$ | NC |
| 133 | PA3/RXD1 | NC |
| 134 | PA4/TXD1 | NC |
| 135 | $V_{CC}$ | $V_{CC}$ |
| 136 | PA5 /SCK1/$\overline{\text{DREQ1}}$/$\overline{\text{IREQ1}}$ | NC |
| 137 | PE7/TIOC2B | NC |
| 138 | PE8/TIOC3A | NC |
| 139 | PE9/TIOC3B | NC |
| 140 | PE10/TIOC3C | NC |
| 141 | $V_{SS}$ | $V_{SS}$ |
| 142 | PE11/TIOC3D | NC |
| 143 | PE12/TIOC4A | NC |
| 144 | PE13/TIOC4B /$\overline{\text{MRES}}$ | NC |

**Table 2.10 Classification of Instructions (cont)**

| Classification | Types | Operation Code | Function | No. of Instructions |
|---|---|---|---|---|
| System control | 11 | CLRT | T bit clear | 31 |
| | | CLRMAC | MAC register clear | |
| | | LDC | Load to control register | |
| | | LDS | Load to system register | |
| | | NOP | No operation | |
| | | RTE | Return from exception processing | |
| | | SETT | T bit set | |
| | | SLEEP | Shift into power-down mode | |
| | | STC | Storing control register data | |
| | | STS | Storing system register data | |
| | | TRAPA | Trap exception handling | |
| Total: | 62 | | | 142 |

Table 2.11 shows the format used in tables 2.12 to 2.17, which list instruction codes, operation, and execution states in order by classification.

### 5.3.1 Address Error Exception Processing

When an address error occurs, the bus cycle in which the address error occurred ends. When the executing instruction then finishes, address error exception processing starts up. The CPU operates as follows:

1. The status register (SR) is saved to the stack.
2. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the last executed instruction.
3. The exception service routine start address is fetched from the exception processing vector table that corresponds to the address error that occurred and the program starts executing from that address. The jump that occurs is not a delayed branch.

## 5.4 Interrupts

Table 5.7 shows the sources that start up interrupt exception processing. These are divided into NMI, user breaks, IRQ, and on-chip peripheral modules.

**Table 5.7 Interrupt Sources**

| Type | Request Source | Number of Sources |
|---|---|---|
| NMI | NMI pin (external input) | 1 |
| User break | User break controller | 1 |
| IRQ | $\overline{\text{IRQ0}}$–$\overline{\text{IRQ7}}$ (external input) | 8 |
| On-chip peripheral module | Direct memory access controller (DMAC) | 4 |
| | Multifunction timer/pulse unit (MTU) | 24 |
| | Serial communications interface (SCI) | 8 |
| | A/D converter | 1* |
| | Data transfer controller (DTC) | 1 |
| | Compare match timer (CMT) | 2 |
| | Watchdog timer (WDT) | 1 |
| | Bus state controller (BSC) | 1 |
| | Port | 1 |

Note: * For A mask products, (A/D0, A/D1) is 2

Each interrupt source is allocated a different vector number and vector table offset. See section 6, Interrupt Controller (INTC), and table 6.3, Interrupt Exception Processing Vectors and Priorities, for more information on vector numbers and vector table address offsets.

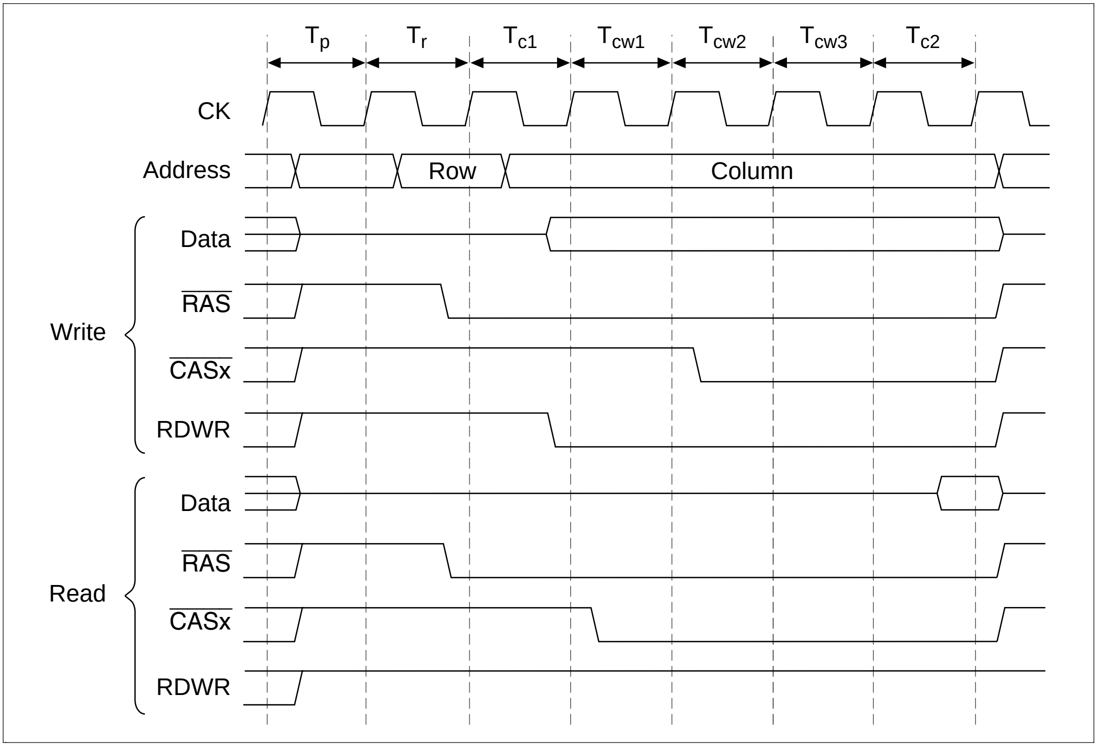


**Figure 10.10 DRAM Bus Cycle (Normal Mode, TPC = 0, RCD = 0, Three Waits)**

Figures 11.17 and 11.18 show cycle steal mode and single address mode. In this case, transfer begins at earliest three cycles after the first $\overline{\text{DREQ}}$ sampling. The second sampling begins from the start of the transfer one bus cycle before the start of the first DMAC transfer. In single address mode, the DACK signal is output during the DMAC transfer period.

**Burst Mode, Single Address, and Edge Detection:** In burst mode with single address and edge detection, $\overline{\text{DREQ}}$ sampling is conducted only on the first cycle. In figure 11.24, a dummy cycle is inserted, at the earliest, three cycles after the timing for the first sampling. During this period, data is undefined, and DACK is not output. Nor is the number of DMAC transfers counted. Thereafter, DMAC transfer continues until the data transfer count set in the DMATCR has ended. $\overline{\text{DREQ}}$ sampling is not conducted during this period. Therefore, DRAK is output on the first cycle only.

When DMAC transfer is resumed after being halted by a NMI or address error, be sure to reinput an edge request. DRAK is output once, and the remaining transfer restarts after output of one dummy cycle.

The DACK output period in burst mode is the same as in cycle steal mode.

### 12.2.6 Timer Counters (TCNT)

The timer counters (TCNT) are 16-bit counters, with one for each channel, for a total of five.

The TCNT are initialized to H'0000 by a power-on reset and when in standby mode. Manual reset does not initialize TCNT. Accessing the TCNT counters in 8-bit units is prohibited. Always access in 16-bit units.

| Channel | Abbreviation | Function |
|---|---|---|
| 0 | TCNT0 | Increment counter |
| 1 | TCNT1 | Increment/decrement counter[*1] |
| 2 | TCNT2 | Increment/decrement counter[*1] |
| 3 | TCNT3 | Increment/decrement counter[*2] |
| 4 | TCNT4 | Increment/decrement counter[*2] |

Notes: *1 Can only be used as an increment/decrement counter in phase counting mode, with other channel overflow/underflow counting. It becomes an increment counter in all other cases.

*2 Can only be used as an increment counter in complementary PWM mode. It becomes an increment counter in all other cases.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| | | | | | | | | |
| 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: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | | | | | | | | |
| 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 |

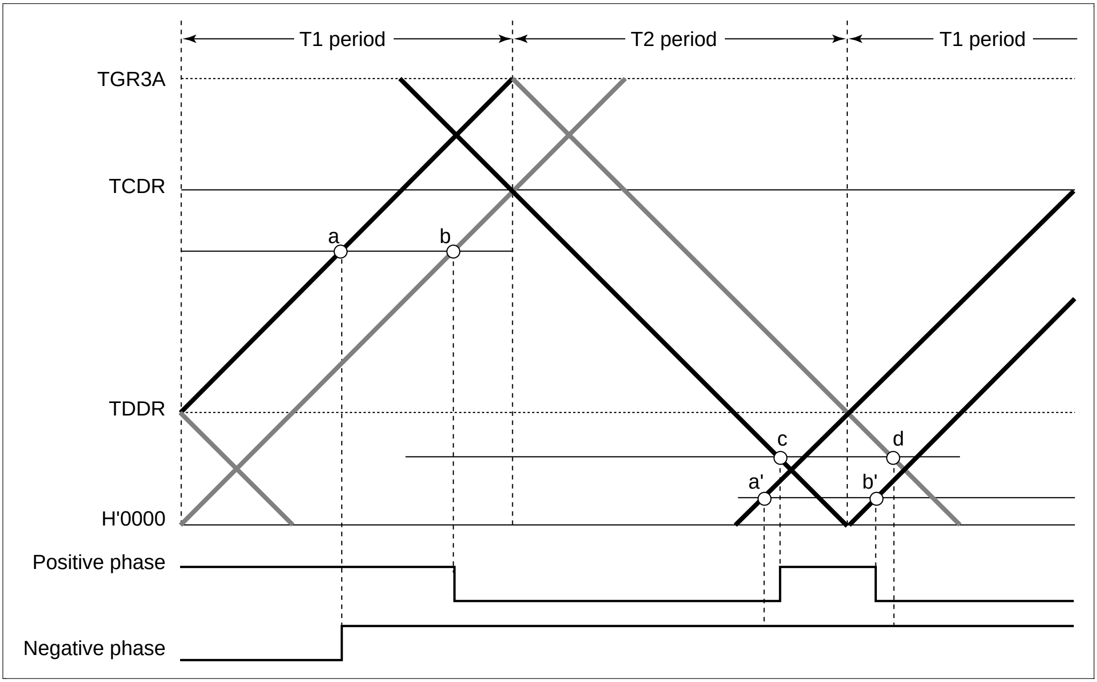


**Figure 12.47 Example of Complementary PWM Mode Waveform Output (3)**

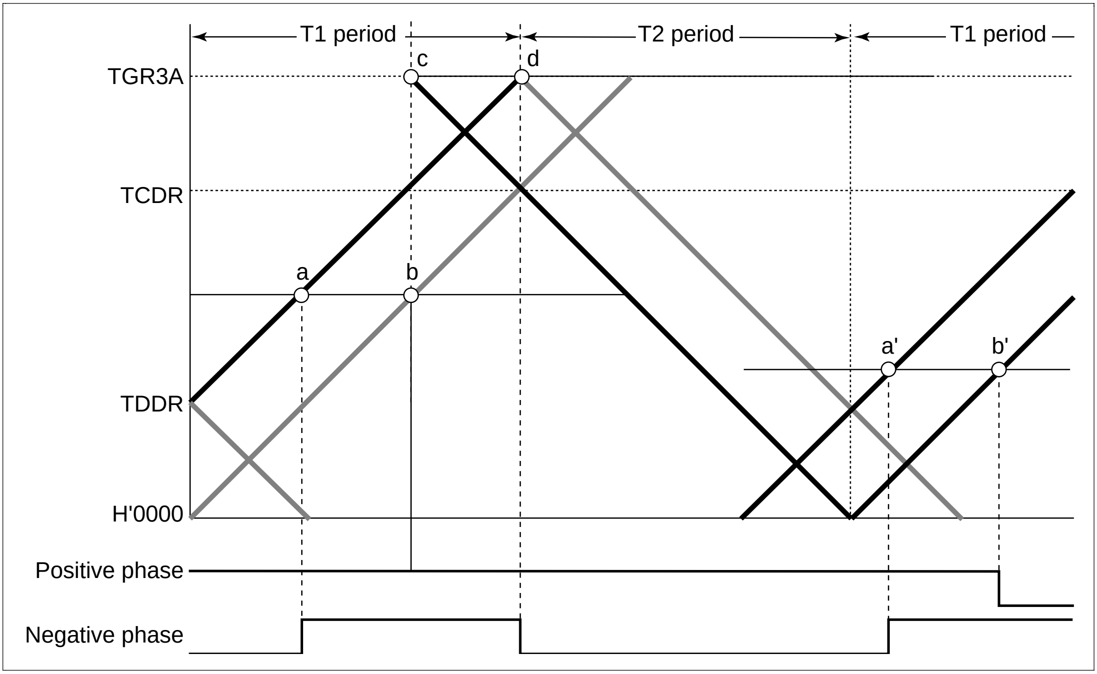


**Figure 12.48 Example of Complementary PWM Mode 0% and 100% Waveform Output (1)**

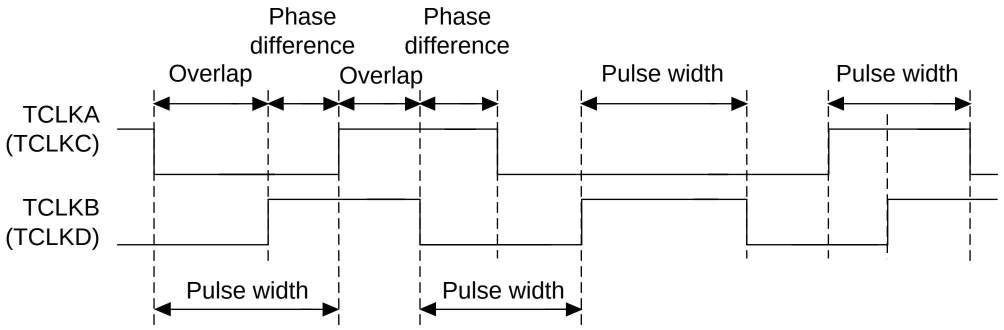
## 12.7 Notes and Precautions

This section describes contention and other matters requiring special attention during MTU operations.

### 12.7.1 Input Clock Limitations

The input clock pulse width, in the case of single edge, must be 1.5 states or greater, and 2.5 states or greater for both edges. Normal operation cannot be guaranteed with lesser pulse widths.

In phase counting mode, the phase difference between the two input clocks and the overlap must be 1.5 states or greater for each, and the pulse width must be 2.5 states or greater. Input clock conditions for phase counting mode are shown in figure 12.76.



Note: Phase difference and overlap: 1.5 states or greater
Pulse width: 2.5 states or greater

**Figure 12.76 Phase Difference, Overlap, and Pulse Width in Phase Count Mode**

### 12.7.2 Note on Cycle Setting

When setting a counter clearing by compare-match, clearing is done in the final state when TCNT matches the TGR value (update timing for count value on TCNT match). The actual number of states set in the counter is given by the following equation:

$$f = \frac{\phi}{(N+1)}$$

(f: counter frequency, $\phi$: operating frequency, N: value set in the TGR)

Note that for channel 0, the TIOCC pin allows both default output setting by TIOR and PWM output when setting buffer operation only for the TGRD register in PWM mode.

When using channel 0 in PWM mode 1 and setting buffer operation, use both the TGRC and TGRD registers as buffer registers.

### 12.7.22 Cautions on Restarting with Sync Clear of Another Channel in Complementary PWM Mode (A Mask Excluded)

The complementation PWM mode operates while waiting for the current, next and the following set values as values of the PWM duty. If clearing sync from another channel during operation, the PWM output will return to the default output and restart.

When restarting with sync clear, the following operations may occur:

1. When restarting with sync clear, the next set value is used for the PWM duty, however, the following set value may be used by mistake.
2. If sync clear and the setting of the value following the next value of PWM duty (write to TGR4D) occurs at the same time, the next set value may be overwritten.

**How to avoid 1**

When selecting the mode to transfer using the crest/trough in the complementary PWM transfer mode, set the value following the next value of the PWM duty (write to TGR4D) while the temporary register is not executing comparisons. Furthermore, set the occurrence timing of sync clear while the temporary register is not executing comparisons.

When selecting the mode to transfer using the crest in the transfer mode, set the value following the next value of the PWM duty (write to TGR4D) while the temporary register is not executing comparisons and while TCNT3 and TCNT4 are counting up. Furthermore, set the occurrence timing of sync clear while the temporary register is not executing comparisons and while TCNT3 and TCNT4 are counting up.

When selecting the mode to transfer using the trough in the transfer mode, set the value following the next value of the PWM duty (write to TGR4D) while the temporary register is not executing comparisons and while TCNT3 and TCNT4 are counting down. Furthermore, set the occurrence timing of sync clear while the temporary register is not executing comparisons and while TCNT3 and TCNT4 are counting down.

**How to avoid 2**

Regardless of the transfer mode, set so that the sync clear and the setting of the value following the next value (write to TGR4D) does not occur at the same time.

**(22) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in PWM Mode 1:** Figure 12.117 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

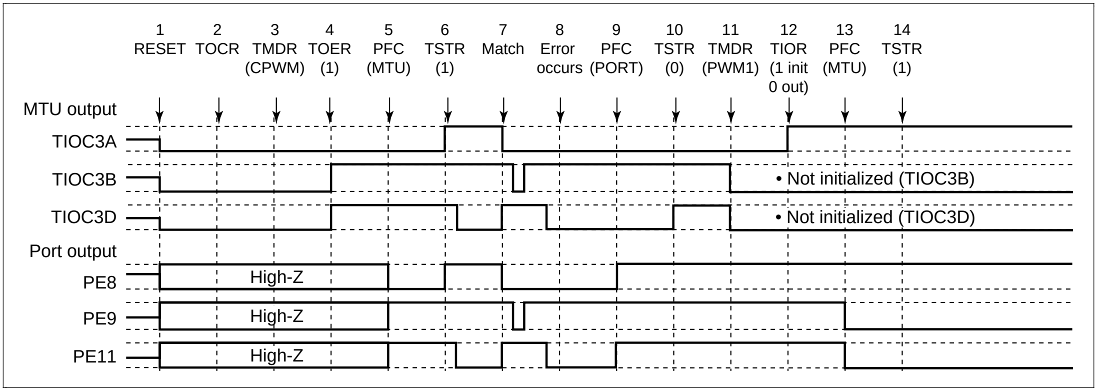


**Figure 12.117 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1**

1 to 10 are the same as in figure 12.116.

11. Set PWM mode 1. (MTU output goes low.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
13. Set MTU output with the PFC.
14. Operation is restarted by TSTR.

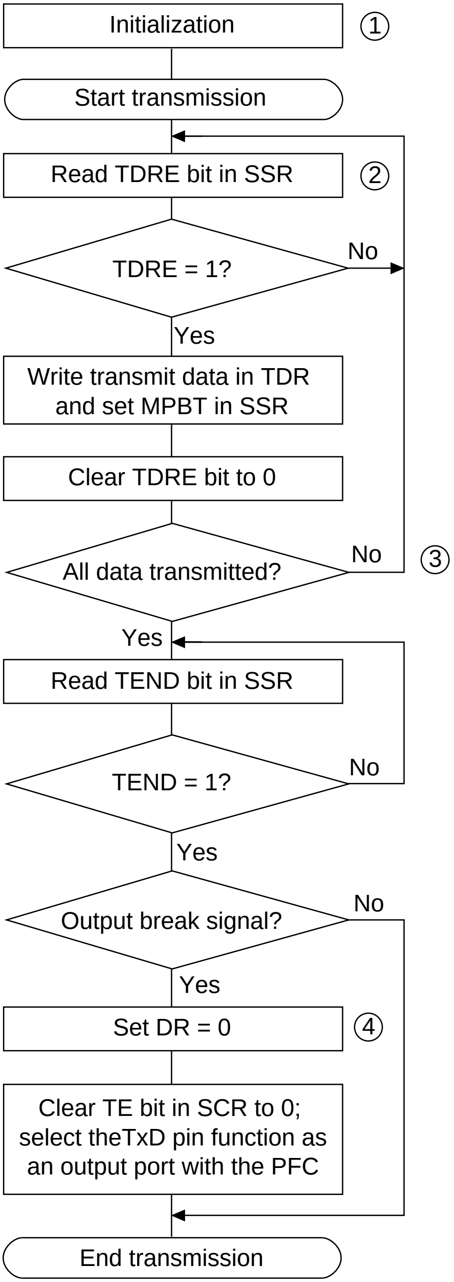


**Figure 14.11 Sample Flowchart for Transmitting Multiprocessor Serial Data**

- Bits 13 and 12—PB6 Mode (PB6MD1 and PB6MD0): PB6MD1 and PB6MD0 select the function of the PB6/$\overline{\text{IRQ4}}$/A18/$\overline{\text{BACK}}$ pin.

| Bit 13: PB6MD1 | Bit 12: PB6MD0 | Description |
|---|---|---|
| 0 | 0 | General input/output (PB6) (initial value) |
| | 1 | Interrupt request input ($\overline{\text{IRQ4}}$) |
| 1 | 0 | Address output (A18) (PB6 in single chip mode) |
| | 1 | Bus right request output ($\overline{\text{BACK}}$) (PB6 in single chip mode) |

- Bits 11 and 10—PB5 Mode (PB5MD1 and PB5MD0): PB5MD1 and PB5MD0 select the function of the PB5/$\overline{\text{IRQ3}}$/$\overline{\text{POE3}}$/RDWR pin.

| Bit 11: PB5MD1 | Bit 10: PB5MD0 | Description |
|---|---|---|
| 0 | 0 | General input/output (PB5) (initial value) |
| | 1 | Interrupt request input ($\overline{\text{IRQ3}}$) |
| 1 | 0 | Port output enable ($\overline{\text{POE3}}$) |
| | 1 | Read/write output (RDWR) |

- Bits 9 and 8—PB4 Mode (PB4MD1 and PB4MD0): PB4MD1 and PB4MD0 select the function of the PB4/$\overline{\text{IRQ2}}$/$\overline{\text{POE2}}$/$\overline{\text{CASH}}$ pin.

| Bit 9: PB4MD1 | Bit 8: PB4MD0 | Description |
|---|---|---|
| 0 | 0 | General input/output (PB4) (initial value) |
| | 1 | Interrupt request input ($\overline{\text{IRQ2}}$) |
| 1 | 0 | Port output enable ($\overline{\text{POE2}}$) |
| | 1 | Column address strobe ($\overline{\text{CASH}}$) (PB4 in single chip mode) |

- Bits 7 and 6—PB3 Mode (PB3MD1 and PB3MD0): PB3MD1 and PB3MD0 select the function of the PB3/$\overline{\text{IRQ1}}$/$\overline{\text{POE1}}$/$\overline{\text{CASL}}$ pin.

| Bit 7: PB3MD1 | Bit 6: PB3MD0 | Description |
|---|---|---|
| 0 | 0 | General input/output (PB3) (initial value) |
| | 1 | Interrupt request input ($\overline{\text{IRQ1}}$) |
| 1 | 0 | Port output enable ($\overline{\text{POE1}}$) |
| | 1 | Column address strobe ($\overline{\text{CASL}}$) (PB3 in single chip mode) |

## 22.2 Overview

### 22.2.1 Block Diagram

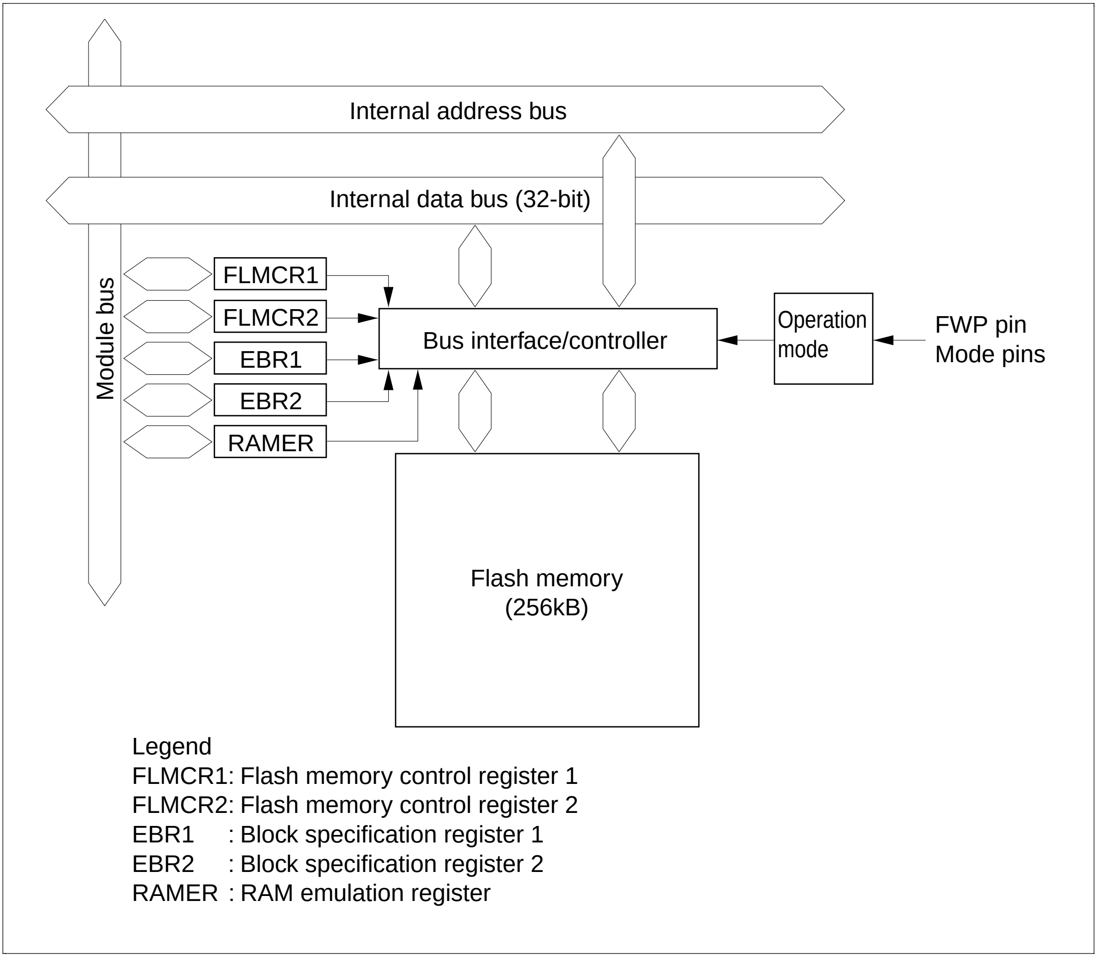
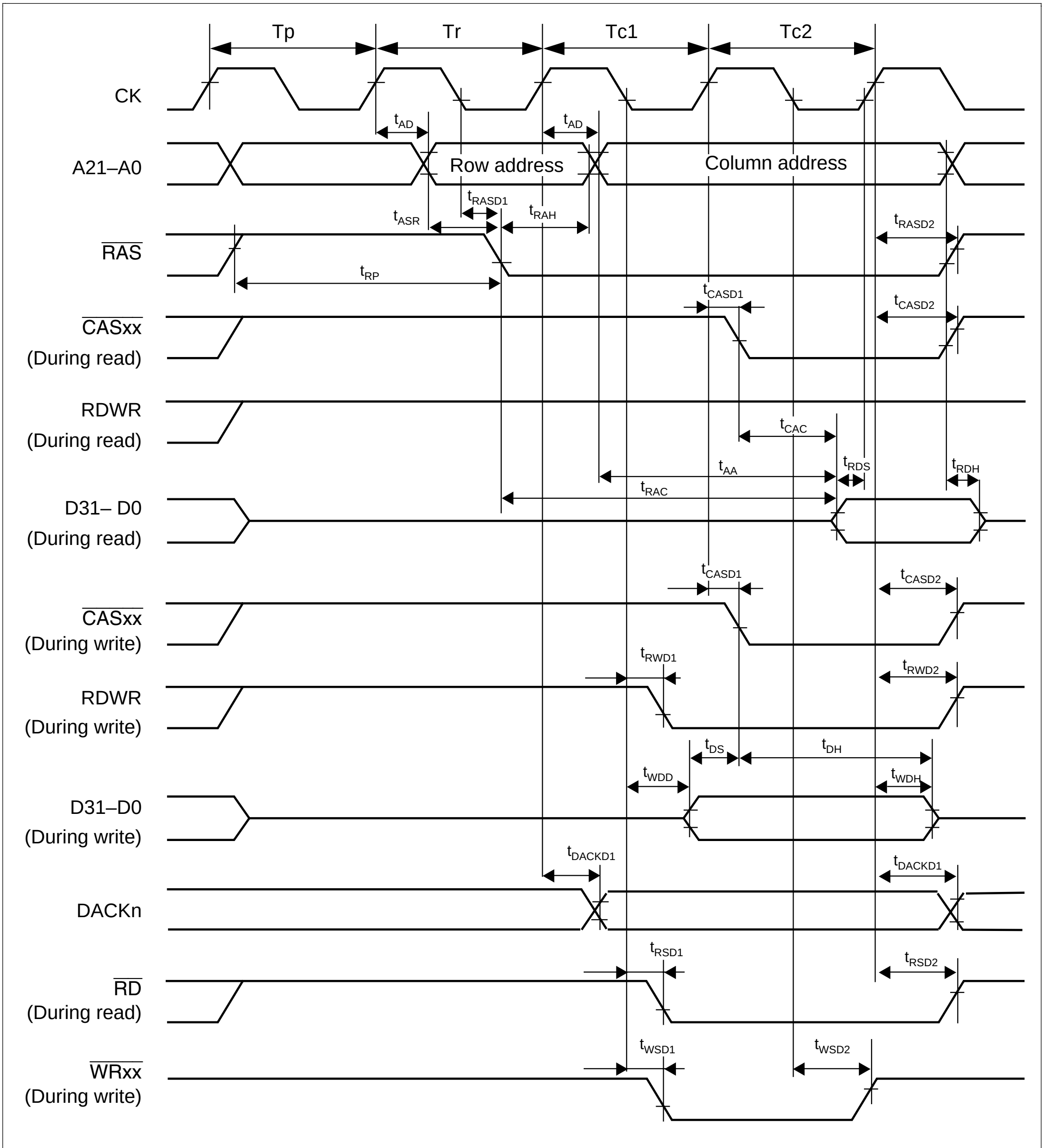

Internal address bus
Internal data bus (32-bit)
Module bus
FLMCR1
FLMCR2
EBR1
EBR2
RAMER
Bus interface/controller
Operation mode
FWP pin
Mode pins
Flash memory
(256kB)
Legend
FLMCR1: Flash memory control register 1
FLMCR2: Flash memory control register 2
EBR1 : Block specification register 1
EBR2 : Block specification register 2
RAMER : RAM emulation register

**Figure 22.1 Flash Memory Block Diagram**



Note: $t_{RDH}$ is specified from fastest negate timing of A21–A0, $\overline{RAS}$, and $\overline{CAS}$.

**Figure 25.11 DRAM Cycle (Normal Mode, No Waits, TPC = 0, RCD = 0)**