**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 | Obsolete |
| Core Processor | SH-2 DSP |
| Core Size | 32-Bit Single-Core |
| Speed | 62.5MHz |
| Connectivity | EBI/EMI, Ethernet, IrDA, FIFO, SCI, SIO |
| Peripherals | DMA, POR, PWM, WDT |
| Number of I/O | 29 |
| Program Memory Size | - |
| Program Memory Type | ROMless |
| EEPROM Size | - |
| RAM Size | 8K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3V ~ 3.6V |
| Data Converters | - |
| Oscillator Type | Internal |
| Operating Temperature | -20°C ~ 75°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 208-LQFP |
| Supplier Device Package | 208-LQFP (28x28) |
| Purchase URL | https://www.e-xfl.com/product-detail/renesas-electronics-america/hd6417615arfv |

# Contents

Figure 2.2 shows the control registers. Table 2.1 indicates the SR register bits.

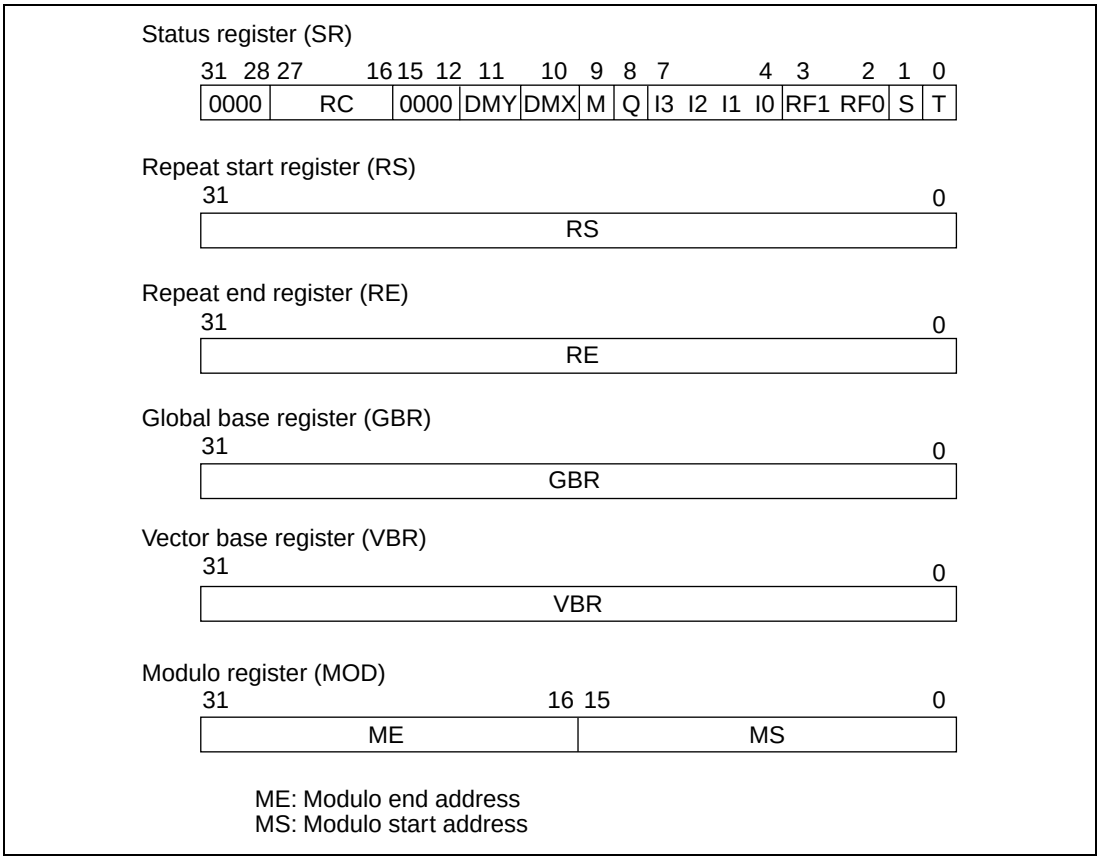



**Figure 2.2 Control Register Configuration**

| Category | Mnemonic | 31–27 | 26 | 25–16 | 15 14 | 13 12 | 11 10 | 9 8 | 7 6 | 5 4 | 3 2 1 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Conditional three operand instructions | (if cc) PSHL Sx, Sy, Dz | 1 | 0 | A field | | 0 0 | 0 0 | if cc | | | |
| | (if cc) PSHA Sx, Sy, Dz | | | | | 0 1 | | | | | |
| | (if cc) PSUB Sx, Sy, Dz | | | | | 1 0 | | | | | |
| | (if cc) PADD Sx, Sy, Dz | | | | | 1 1 | | 01: Uncon-dition | | | |
| | Reserved | | | | | 0 0 | 0 1 | | | | |
| | (if cc) PAND Sx, Sy, Dz | | | | | 0 1 | | | | | |
| | (if cc) PXOR Sx, Sy, Dz | | | | | 1 0 | | | | | |
| | (if cc) POR Sx, Sy, Dz | | | | | 1 1 | | | | | |
| | (if cc) PDEC Sx, Dz | | | | | 0 0 | 1 0 | 10:DCT | | | |
| | (if cc) PINC Sx, Dz | | | | | 0 1 | | | | | |
| | (if cc) PDEC Sy, Dz | | | | | 1 0 | | | | | |
| | (if cc) PINC Sy, Dz | | | | | 1 1 | | | | | |
| | (if cc) PCLR Dz | | | | | 0 0 | 1 1 | 11:DCF | | | |
| | (if cc) PDMSB Sx, Dz | | | | | 0 1 | | | | | |
| | Reserved | | | | | 1 0 | | | | | |
| | (if cc) PDMSB Sy, Dz | | | | | 1 1 | | | | | |
| | (if cc) PNEG Sx, Dz | | | | 1 1 | 0 0 | 1 0 | | | | |
| | (if cc) PCOPY Sx, Dz | | | | | 0 1 | | | | | |
| | (if cc) PNEG Sy, Dz | | | | | 1 0 | | | | | |
| | (if cc) PCOPY Sy, Dz | | | | | 1 1 | | | | | |
| | Reserved | | | | | | | 0 0 | | | |
| | (if cc) PSTS MACH, Dz | | | | | 0 0 | 1 1 | if cc | | | |
| | (if cc) PSTS MACL, Dz | | | | | 0 1 | | | | | |
| | (if cc) PLDS Dz, MACH | | | | | 1 0 | | | | | |
| | (if cc) PLDS Dz, MACL | | | | | 1 1 | | | | | |
| | Reserved*2 | | | | | | | 0 0 | | | |
| | | | | | | | 0 * | | | | |
| | Reserved | 1 | | | | | | | | | |

Notes: 1. System reserved code
2. (if cc): DCT (DC bit true), DCF (DC bit false), or none (unconditional instruction)

## 2.5 Instruction Set

The instructions are divided into three groups: CPU instructions executed by the CPU core, DSP data transfer instructions executed by the DSP unit, and DSP operation instructions. There are a number of CPU instructions for supporting the DSP functions. The instruction set is explained below in terms of each of the three groups.

### 2.5.3 DSP Operation Instruction Set

DSP operation instructions are digital signal processing instructions processed by the DSP unit. These instructions use 32-bit instruction codes, and multiple instructions are executed in parallel. The instruction codes are divided into an A field and a B field; parallel data transfer instructions are designated in the A field, and single or double data operation instructions are designated in the B field. Instructions can be independently designated and execution can also be carried out independently. A parallel data transfer instruction designated in the A field is exactly the same as a double data transfer instruction.

The B field data operation instructions are divided into three groups: double data operation instructions, conditional single data operation instructions, and unconditional single data operation instructions. Table 2.32 lists the instruction formats of the DSP operation instructions. Each of the operands can be independently selected from the DSP registers. Table 2.33 shows the correspondence between the DSP operation instruction operands and registers.

**Table 2.32 DSP Operation Instruction Formats**

| Classification | | Instruction Forms | Instruction |
|---|---|---|---|
| Double data operation instructions (6 operands) | | ALUop. Sx, Sy, Du<br>MLTop. Se, Sf, Dg | PADD PMULS,<br>PSUB PMULS |
| Conditional single data operation instructions | 3 operands | ALUop. Sx, Sy, Dz<br>DCT ALUop. Sx, Sy, Dz<br>DCF ALUop. Sx, Sy, Dz | PADD, PAND, POR, PSHA, PSHL, PSUB, PXOR |
| | 2 operands | ALUop. Sx, Dz<br>DCT ALUop. Sx, Dz<br>DCF ALUop. Sx, Dz<br>ALUop. Sy, Dz<br>DCT ALUop. Sy, Dz<br>DCF ALUop. Sy, Dz | PCOPY, PDEC, PDMSB, PINC, PLDS, PSTS, PNEG |
| | 1 operand | ALUop. Dz<br>DCT ALUop. Dz<br>DCF ALUop. Dz | PCLR, PSHA #imm, PSHL #imm |
| Unconditional single data operation instructions | 3 operands | ALUop. Sx, Sy, Du<br>MLTop. Se, Sf, Dg | PADDC, PSUBC, PMULS |
| | 2 operands | ALUop. Sx, Dz<br>ALUop. Sy, Dz<br>ALUop. Sx, Sy | PCMP, PABS, PRND |
| | 1 operand | ALUop. Dz | PSHA #imm, PSHL #imm |

## 4.4 Interrupts

### 4.4.1 Interrupt Sources

Table 4.7 shows the sources that initiate interrupt exception handling. These are divided into NMI, user breaks, H-UDI, IRL, IRQ, and on-chip peripheral modules.

**Table 4.7 Types of Interrupt Sources**

| Type | Request Source | Number of Sources |
|---|---|---|
| NMI | NMI pin (external input) | 1 |
| User break | User break controller (UBC) | 1 |
| H-UDI | High-performance user debugging interface (H-UDI) | 1 |
| IRL | IRL1 to IRL15 (external input) | 15 |
| IRQ | IRQ0 to IRQ3 (external input) | 4 |
| On-chip peripheral module | Direct memory access controller (DMAC) | 2 |
| | Ethernet controller (EtherC) and Ethernet controller direct memory access controller (E-DMAC) | 1 |
| | 16-bit free-running timer (FRT) | 3 |
| | Watchdog timer (WDT) | 1 |
| | Bus state controller (BSC) | 1 |
| | Serial I/O (SIO) | 4 |
| | Serial communication interface with FIFO (SCIF) | 4 |
| | 16-bit timer pulse unit (TPU) | 13 |

Each interrupt source is allocated a different vector number and vector table address offset. See table 5.4, Interrupt Exception Vectors and Priority Order, in section 5, Interrupt Controller, for more information.

### 5.2.5 IRQ Interrupts

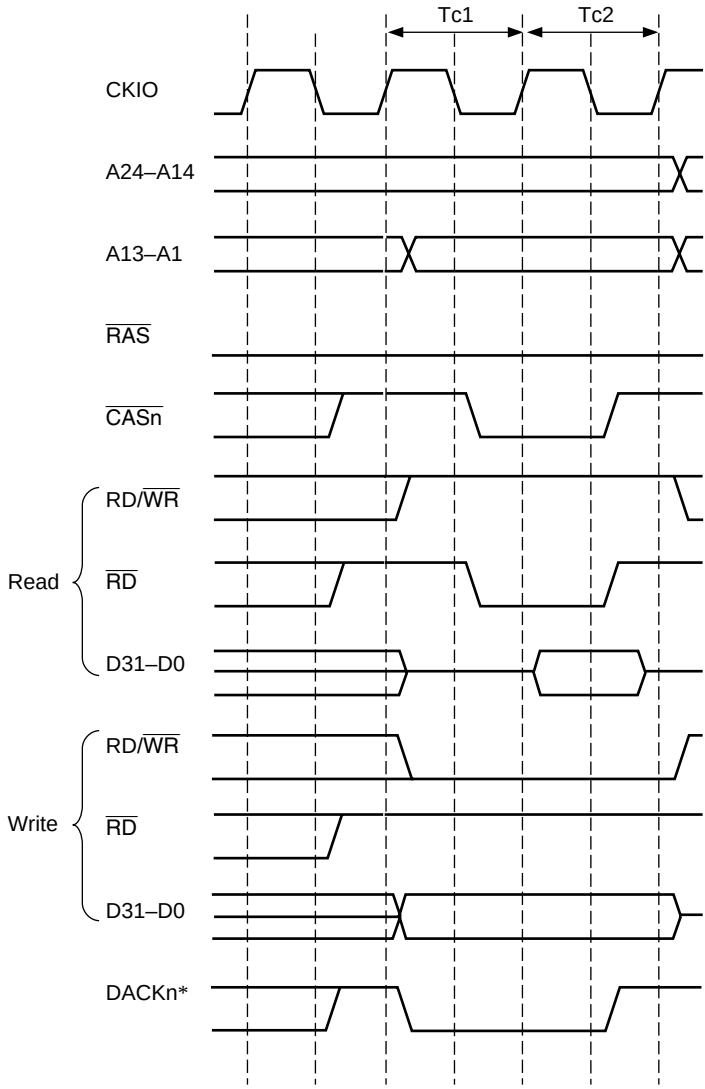
An IRQ interrupt is requested when the external interrupt vector mode select bit (EXIMD) of the interrupt control register (ICR) is set to 1. An IRQ interrupt corresponds to input at one of pins $\overline{\text{IRL3}}$ to $\overline{\text{IRL0}}$. Low-level sensing or rising/falling/both-edge sensing can be selected independently for each pin by the IRQ sense select bits (IRQ31S to IRQ00S) in the IRQ control/status register (IRQCSR), and a priority level of 0 to 15 can be selected independently for each pin by means of interrupt priority register C (IPRC). Set the interrupt vector mode select bit (VECMD) of the interrupt control register (ICR) to enable external input of vector numbers. External vector numbers are 0 to 127, and are input to the external vector input pins (D7 to D0) during the interrupt vector fetch bus cycle. When an external vector is used, 0 is input to D7.

When an IRQ interrupt is accepted in external vector mode, the IRQ interrupt priority level is output from the interrupt acceptance level output pins (A3 to A0). The external vector fetch signal ($\overline{\text{IVECF}}$) is also asserted. The external vector number is read from signals D7 to D0 at this time.

IRQ interrupt exception processing sets the interrupt mask bits (I3 to I0) in the status register (SR) to the priority level value of the IRQ interrupt that was accepted.

**Table 5.3 IRL Interrupt Priority Levels and Auto-Vector Numbers**

| Pin | | | | Priority Level | Vector Number |
|---|---|---|---|---|---|
| $\overline{\text{IRL3}}$ | $\overline{\text{IRL2}}$ | $\overline{\text{IRL1}}$ | $\overline{\text{IRL0}}$ | | |
| 0 | 0 | 0 | 0 | 15 | 71 |
| | | | 1 | 14 | |
| | | 1 | 0 | 13 | 70 |
| | | | 1 | 12 | |
| | 1 | 0 | 0 | 11 | 69 |
| | | | 1 | 10 | |
| | | 1 | 0 | 9 | 68 |
| | | | 1 | 8 | |
| 1 | 0 | 0 | 0 | 7 | 67 |
| | | | 1 | 6 | |
| | | 1 | 0 | 5 | 66 |
| | | | 1 | 4 | |
| | 1 | 0 | 0 | 3 | 65 |
| | | | 1 | 2 | |
| | | 1 | 0 | 1 | 64 |




Note: * DACKn waveform when active-low is specified

**Figure 7.44 RAS Down Mode Same Row Access Timing**

- High-impedance conditions: Not dependent on BCR settings etc. when $\overline{WAIT}$ = L and $\overline{BUSHiZ}$ = L
- Applicable pins: A[24:0], D[31:0], $\overline{CS3}$, RD/WR, $\overline{RD}$, $\overline{RAS}$, $\overline{CAS}$/$\overline{OE}$, DQMLL/$\overline{WE0}$, DQMLU/$\overline{WE1}$, DQMUL/$\overline{WE2}$, DQMUU/$\overline{WE3}$ (total of 66 pins)

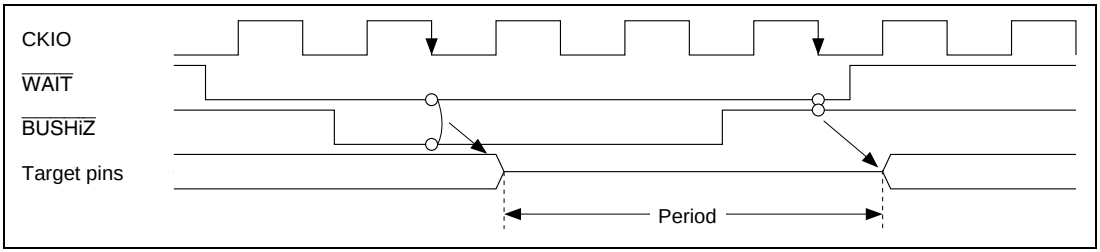



**Figure 7.60 $\overline{BUSHiZ}$ Bus Timing**

1. Can be used when memory is shared by the CPU and an external device.
2. When $\overline{BUSHiZ}$ is asserted after asserting $\overline{WAIT}$, the CPU appears to release the bus.
3. When it becomes possible to access the shared memory, $\overline{BUSHiZ}$ is negated.
4. When the data is ready, $\overline{WAIT}$ is negated.

This procedure allows the CPU and an external device to share memory.

## 7.11 Usage Notes

### 7.11.1 Normal Space Access after Synchronous DRAM Write when Using DMAC

Negation of the DQMn/$\overline{WEn}$ signal in a synchronous DRAM write and $\overline{CSn}$ assertion in an immediately following normal space access both occur at the same rising edge of CKIO (figure 7.61). As there is a risk of an erroneous write to normal space in this case, when synchronous DRAM or a high-speed device is connected to normal space, it is recommended that $\overline{CSn}$ be delayed on the system side.

Cases in which a synchronous DRAM write and normal space access occur consecutively are shown in table 7.10.

**Table 7.10 Access Sequence**

| Write to Synchronous DRAM | Normal Space Access |
|---|---|
| CPU | DMA |
| DMA | CPU |
| DMA | DMA |

Note: When an access by the CPU is performed immediately after a write by the CPU, internally the accesses are not consecutive.

### 9.1.4 Ethernet Controller Register Configuration

The Ethernet controller (EtherC) has the nineteen 32-bit registers shown in table 9.2.

**Table 9.2 EtherC Registers**

| Name | Abbre-viation | R/W | Initial Value | Address |
|---|---|---|---|---|
| EtherC mode register | ECMR | R/W | H'00000000 | H'FFFFFD60 |
| EtherC status register | ECSR | R/W*1 | H'00000000 | H'FFFFFD64 |
| EtherC interrupt permission register | ECSIPR | R/W | H'00000000 | H'FFFFFD68 |
| PHY interface register | PIR | R/W | H'0000000X | H'FFFFFD6C |
| MAC address high register | MAHR | R/W | H'00000000 | H'FFFFFD70 |
| MAC address low register | MALR | R/W | H'00000000 | H'FFFFFD74 |
| Receive flame length register | RFLR | R/W | H'00000000 | H'FFFFFD78 |
| PHY status register | PSR | R | H'00000000 | H'FFFFFD7C |
| Transmit retry over counter register | TROCR | R/W*2 | H'00000000 | H'FFFFFD80 |
| Collision detect counter register | CDCR | R/W*2 | H'00000000 | H'FFFFFD84 |
| Lost carrier counter register | LCCR | R/W*2 | H'00000000 | H'FFFFFD88 |
| Carrier not detect counter register | CNDCR | R/W*2 | H'00000000 | H'FFFFFD8C |
| Illegal frame length counter register | IFLCR | R/W*2 | H'00000000 | H'FFFFFD90 |
| CRC error frame receive counter register | CEFCR | R/W*2 | H'00000000 | H'FFFFFD94 |
| Frame receive error counter register | FRECR | R/W*2 | H'00000000 | H'FFFFFD98 |
| Too-short frame receive counter register | TSFRCR | R/W*2 | H'00000000 | H'FFFFFD9C |
| Too-long frame receive counter register | TLFRCR | R/W*2 | H'00000000 | H'FFFFFDA0 |
| Residual-bit frame counter register | RFCR | R/W*2 | H'00000000 | H'FFFFFDA4 |
| Multicast address frame counter register | MAFCR | R/W*2 | H'00000000 | H'FFFFFDA8 |

Notes: All registers must be accessed as 32-bit units.
Reserved bits in a register should only be written with 0.
The value read from a reserved bit is not guaranteed.
1. Individual bits are cleared by writing 1.
2. Cleared by a write to the register.

Bit 8—Transmit Retry Over (TRO): Indicates that a retry-over condition has occurred during frame transmission. Total 16 transmission retries including 15 retries based on the back-off algorithm are failed after the EtherC transmission starts.

| Bit 8: TRO | Description | |
|---|---|---|
| 0 | Transmit retry-over condition not detected | (Initial value) |
| 1 | Transmit retry-over condition detected (interrupt source) | |

Bit 7—Receive Multicast Address Frame (RMAF): Indicates that a multicast address frame has been received.

| Bit 7: RMAF | Description | |
|---|---|---|
| 0 | Multicast address frame has not been received | (Initial value) |
| 1 | Multicast address frame has been received (interrupt source) | |

Bits 6 and 5—Reserved: These bits are always read as 0. The write value should always be 0.

Bit 4—Receive Residual-Bit Frame (RRF): Indicates that a residual-bit frame has been received.

| Bit 4: RRF | Description | |
|---|---|---|
| 0 | Residual-bit frame has not been received | (Initial value) |
| 1 | Residual-bit frame has been received (interrupt source) | |

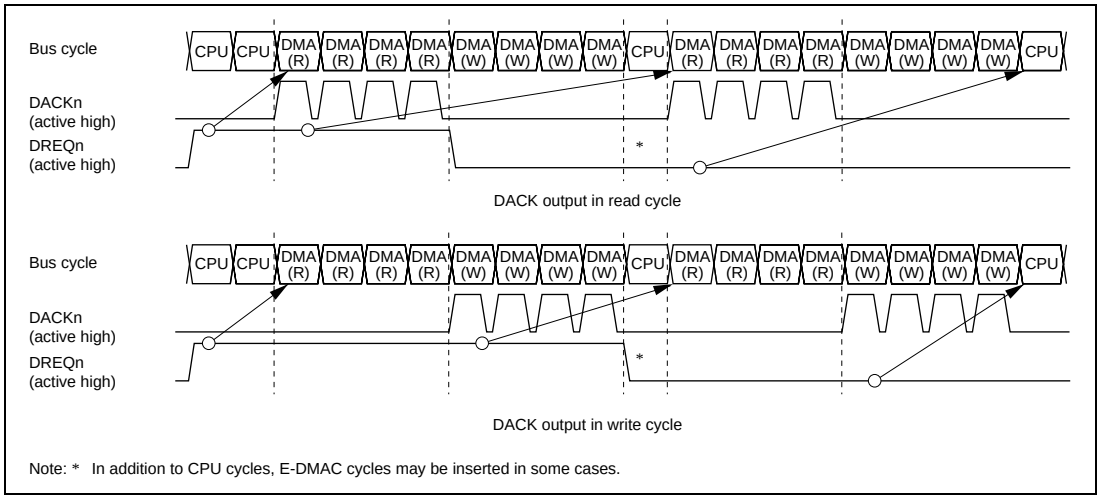
Bit 3—Receive Too-Long Frame (RTLF): Indicates that a frame of 1519 bytes or longer has been received.

| Bit 3: RTLF | Description | |
|---|---|---|
| 0 | Too-long frame has not been received | (Initial value) |
| 1 | Too-long frame has been received (interrupt source) | |

Bit 2—Receive Too-Short Frame (RTSF): Indicates that a frame of fewer than 64 bytes has been received.

| Bit 2: RTSF | Description | |
|---|---|---|
| 0 | Too-short frame has not been received | (Initial value) |
| 1 | Too-short frame has been received (interrupt source) | |

11. The following restrictions apply when using dual address mode for 16-byte transfer in cycle-steal mode:
    a. When external request and level detection are set, do not input DREQn during cycles in which DACKn is not active after the start of DMA transfer.




Note: * In addition to CPU cycles, E-DMAC cycles may be inserted in some cases.

    b. When external request DREQ edge detection is set, if DREQn is input continuously the DMAC continues to operate without insertion of a CPU cycle. (However, a CPU cycle will begin if there is no request from DREQn.)

Conditions:

(1) Conditions for data bus collision during single-address DMA transfer

When the following conditions are all satisfied, a data bus collision occurs during single-address DMA transfer from the external device with DACK to SDRAM.

(a) The clock ratio is set to external clock (Eϕ):internal clock (Iϕ) = 1:1.

(b) SDRAM is used in single write mode.

(c) Immediately after the SH7615 writes data to SDRAM, DMAC transfers data from the external device with DACK to SDRAM in single address mode.

Countermeasures:

(1) Countermeasure against data bus collision during single-address DMA transfer

This problem is avoided by any of the following countermeasures.

(a) Specify a clock ratio other than external clock (Eϕ):internal clock (Iϕ) = 1:1.

(b) Do not write to SDRAM from CPU, Ethernet controller direct memory access controller (E-DMAC), or another channel of the DMAC during single-address DMA transfer from the external device with DACK to SDRAM.

16. DMAC DACK error output

Phenomenon:

(1) DACK error

When DMAC channels 0 and 1 are both set to external request (DREQ0 and DREQ1) mode, the DMAC may execute DMA transfer with DACK1 output on channel 1 while the DREQ1 is not input.

Conditions:

(1) Conditions for DACK error

When the following conditions are all satisfied, the DMAC executes DMA transfer with DACK1 output on channel 1 while the DREQ1 is not input.

(a) DMAC channels 0 and 1 are both enabled.

(b) DMAC channels 0 and 1 both select the external request (DREQ0 and DREQ1) for the transfer request source.

(c) DMAC channels 0 and 1 are both set to cycle-steal mode.

(d) Round-robin mode is specified as the DMAC priority mode.

Countermeasures:

(1) Countermeasure against DACK error

This problem is avoided by any of the following countermeasures.

(a) Set either DMAC channel 0 or 1 to burst mode.

(b) Set the DMAC priority mode to fixed priority mode.

## 12.3 CPU Interface

FRC, OCRA, OCRB, and FICR are 16-bit registers. The data bus width between the CPU and FRT, however, is only 8 bits. Access of these three types of registers from the CPU therefore needs to be performed via an 8-bit temporary register called TEMP.

The following describes how these registers are read from and written to:

- Writing to 16-bit Registers

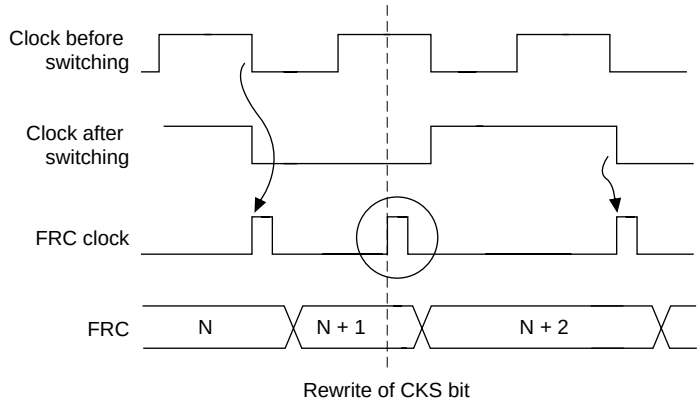
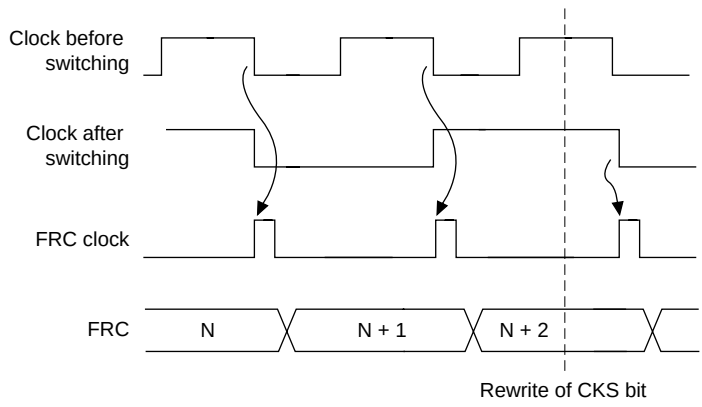
  The upper byte is written, which results in the upper byte of data being stored in TEMP. The lower byte is then written, which results in 16 bits of data being written to the register when combined with the upper byte value in TEMP.

- Reading from 16-bit Registers

  The upper byte of data is read, which results in the upper byte value being transferred to the CPU. The lower byte value is transferred to TEMP. The lower byte is then read, which results in the lower byte value in TEMP being sent to the CPU.

When registers of these three types are accessed, two byte accesses should always be performed, first to the upper byte, then the lower byte. If only the upper byte or lower byte is accessed, the data will not be transferred properly.

Figure 12.2 and 12.3 show the flow of data when FRC is accessed. Other registers function in the same way. When reading OCRA and OCRB, however, both upper and lower-byte data is transferred directly to the CPU without passing through TEMP.

| No. | Timing of Rewrite of CKS1 and CKS0 Bits | FRC Operation |
|---|---|---|
| 3 | High-to-low switch |  |
| 4 | High-to-high switch |  |

Note: Because the switchover is considered a falling edge, FRC starts counting up.

### 12.7.5 Timer Output (FTOA, FTOB)

During a power-on reset, the timer outputs (FTOA, FTOB) will be unreliable until the oscillation stabilizes. The initial value is output after the oscillation settling time has elapsed.

Bits 4 and 3—Clock Edge 1 and 0 (CKEG1, CKEG0): These bits select the input clock edge. When a both-edges count is selected, a clock divided by two from the input clock can be selected. (e.g. Pϕ/4 both edges = Pϕ/2 rising edge). If phase counting mode is used on channels 1, and 2, this setting is ignored and the phase counting mode setting has priority.

| Bit 4: CKEG1 | Bit 3: CKEG0 | Description | |
|---|---|---|---|
| 0 | 0 | Count at rising edge | (Initial value) |
| | 1 | Count at falling edge | |
| 1 | — | Count at both edges | |

Note: Internal clock edge selection is valid when the input clock is Pϕ/4 or slower. If Pϕ/1 is selected for the input clock, this setting is ignored and a rising-edge count is selected.

Bits 2 to 0—Time Prescaler 2 to 0 (TPSC2 to TPSC0): These bits select the TCNT counter clock. The clock source can be selected independently for each channel. Table 16.4 shows the clock sources that can be set for each channel.

**Table 16.4 TPU Clock Sources**

| | Internal Clock | | | | | | External Clock | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Channel | Pϕ/1 | Pϕ/4 | Pϕ/16 | Pϕ/64 | Pϕ/256 | Pϕ/1024 | TCLKA | TCLKB | TCLKC | TCLKD |
| 0 | O | O | O | O | | | O | O | O | O |
| 1 | O | O | O | O | O | | O | O | | |
| 2 | O | O | O | O | | O | O | O | O | |

Notes: O: Setting
Blank: No setting

- Example of input capture operation

  Figure 16.13 shows an example of input capture operation.

  In this example both rising and falling edges have been selected as the TIOCA pin input capture input edge, falling edge has been selected as the TIOCB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT.

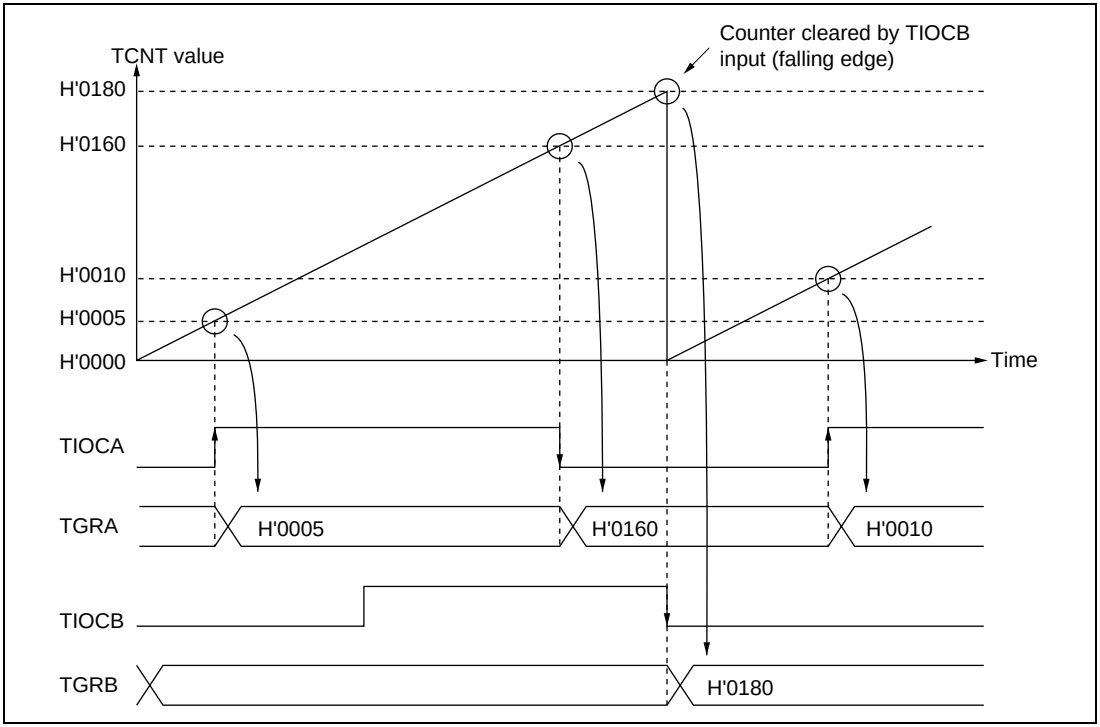
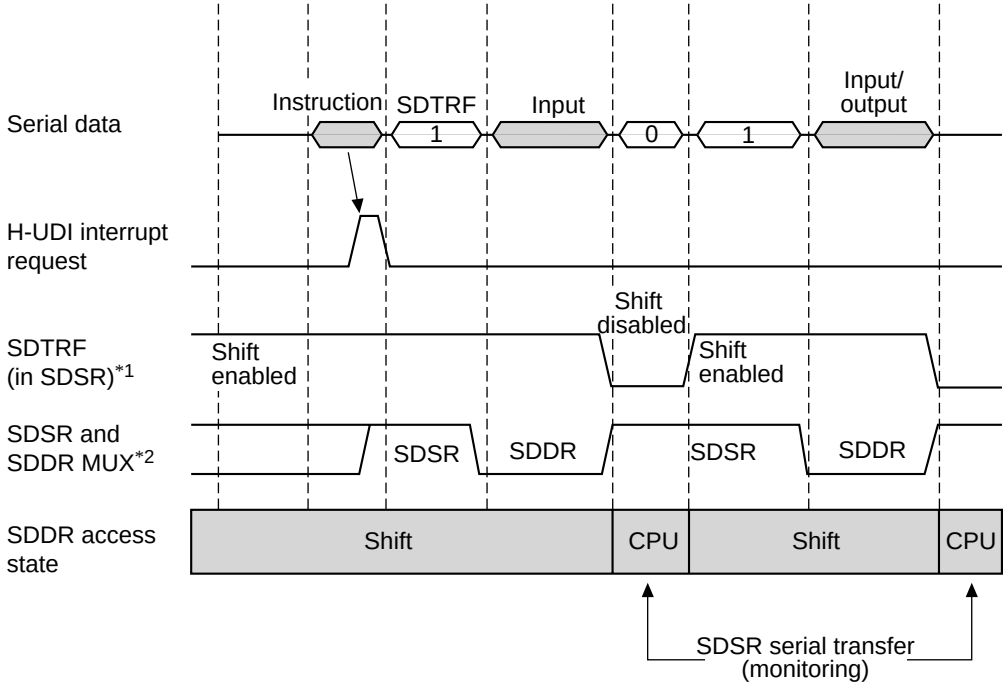



**Figure 16.13 Example of Input Capture Operation**




Notes: 1. SDTRF flag (in SDSR): Indicates whether SDDR access by the CPU or serial transfer data input/output to SDDR is possible.

| | |
|---|---|
| 1 | SDDR is shift-disabled. SDDR access by the CPU is enabled. |
| 2 | SDDR is shift-enabled. Do not access SDDR until SDTRF = 0. |

Conditions:
- SDTRF = 1
  — When $\overline{\text{TRST}}$ = 0
  — When the CPU writes 1
  — In bypass mode
- SDTRF = 0
  — End of SDDR shift access in serial transfer

2. SDSR/SDDR (Update-DR state) internal MUX switchover timing
   - Switchover from SDSR to SDDR: On completion of serial transfer in which SDTRF = 1 is output from TDO
   - Switchover from SDDR to SDSR: On completion of serial transfer to SDDR

**Figure 17.3 Data Input/Output Timing Chart (1)**

### 20.2.2 Standby Control Register 2 (SBYCR2)

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | MSTP11 (TPU) | MSTP10 (SIO2) | MSTP9 (SIO1) | MSTP8 (SIO0) | MSTP7 (SCIF2) | MSTP6 (SCIF1) |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

Standby control register 2 (SBYCR2) is an 8-bit read/write register that sets the power-down mode state. SBYCR2 is initialized to H'00 by a reset.

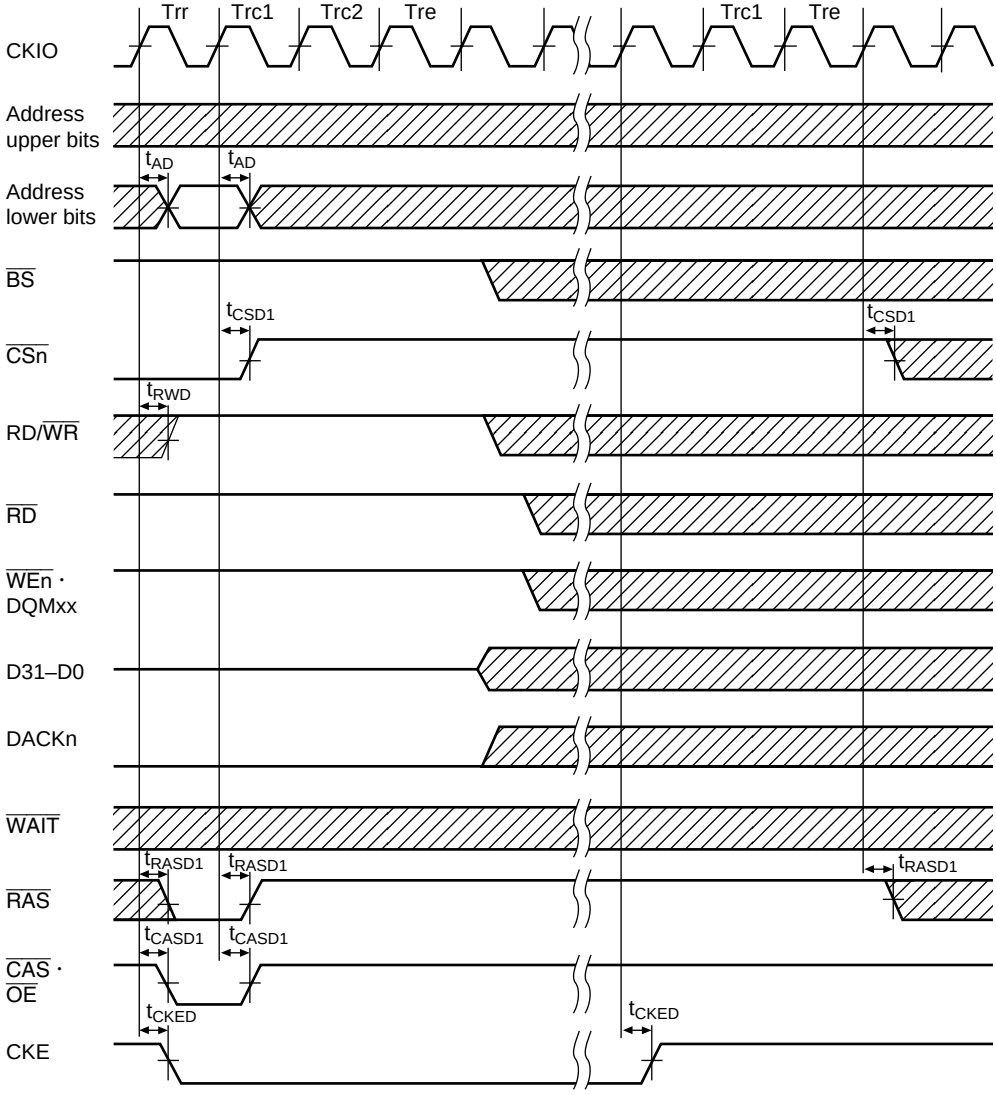
Bits 7 and 6—Reserved: These bits are always read as 0. The write value should always be 0.

Bit 5—Module Stop 11 (MSTP11): Specifies halting the clock supply to the 16-bit timer pulse unit (TPU). When the MSTP11 bit is set to 1, the supply of the clock to the TPU is halted. When the clock halts, the TPU retains its pre-halt state, and the TPU interrupt vector register in the INTC retains its pre-halt value. Therefore, when MSTP11 is cleared to 0 and the clock supply to the TPU is resumed, the TPU starts operating again.

| Bit 5: MSTP11 | Description | |
|---|---|---|
| 0 | TPU running | (Initial value) |
| 1 | Clock supply to TPU halted | |

Bit 4—Module Stop 10 (MSTP10): Specifies halting the clock supply to SIO channel 2. When the MSTP10 bit is set to 1, the supply of the clock to SIO channel 2 is halted. When the clock halts, SIO channel 2 retains its pre-halt state, and the SIO channel 2 interrupt vector register in the INTC retains its pre-halt value. Therefore, when MSTP10 is cleared to 0 and the clock supply to SIO channel 2 is restarted, operation starts again.

| Bit 4: MSTP10 | Description | |
|---|---|---|
| 0 | SIO channel 2 running | (Initial value) |
| 1 | Clock supply to SIO channel 2 halted | |




Note: A self-refresh cycle is always preceded by a precharge cycle. The number of cycles between the two is determined by the number of cycles specified by TRP.

**Figure 21.33 Synchronous DRAM Self-Refresh Cycle (TRAS = 3)**

**Renesas 32-Bit RISC Microcomputer**

**Hardware Manual**

**SH7615 Group**

Publication Date: 1st Edition, September 2000
Rev.2.00, March 17, 2005
Published by: Sales Strategic Planning Div.
Renesas Technology Corp.
Edited by: Technical Documentation & Information Department
Renesas Kodaira Semiconductor Co., Ltd.