Renesas Electronics America Inc - DF2215UBR16V 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	H8S/2000
Core Size	16-Bit
Speed	16MHz
Connectivity	SCI, SmartCard, USB
Peripherals	DMA, POR, PWM, WDT
Number of I/O	68
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 6x10b; D/A 2x8b
Oscillator Type	External
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LFBGA
Supplier Device Package	112-LFBGA (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df2215ubr16v

Section 6 Bus Controller

Section 7 DMA Controller (DMAC)

Section 8 Data Transfer Controller (DTC)

Section 9 I/O Ports

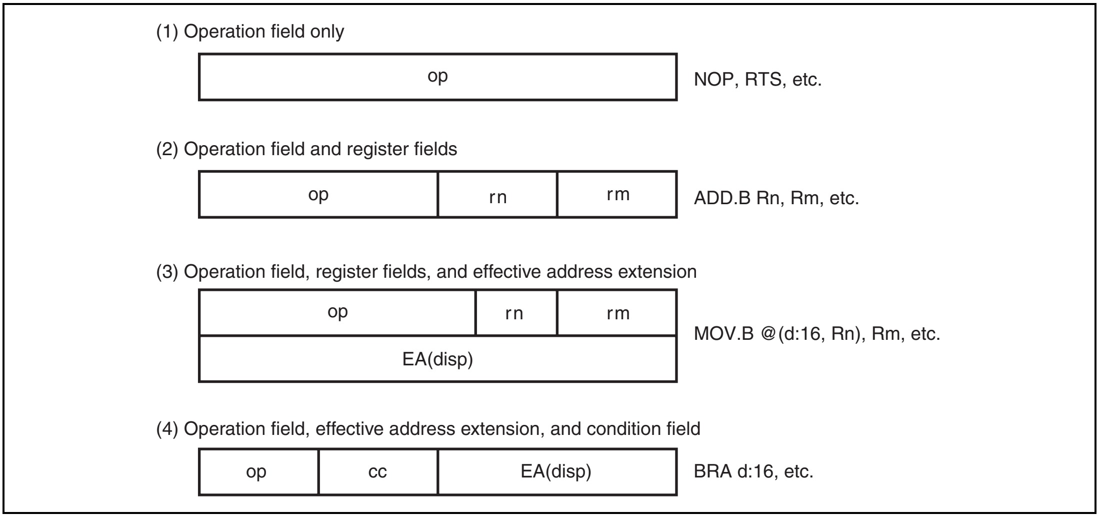

(1) Operation field only
op
NOP, RTS, etc.
(2) Operation field and register fields
op
rn
rm
ADD.B Rn, Rm, etc.
(3) Operation field, register fields, and effective address extension
op
rn
rm
EA(disp)
MOV.B @(d:16, Rn), Rm, etc.
(4) Operation field, effective address extension, and condition field
op
cc
EA(disp)
BRA d:16, etc.

Figure 2.11 Instruction Formats (Examples)

2.7 Addressing Modes and Effective Address Calculation

The H8S/2000 CPU supports the eight addressing modes listed in table 2.11. Each instruction uses a subset of these addressing modes. Arithmetic and logic instructions can use the register direct and immediate modes. Data transfer instructions can use all addressing modes except program-counter relative and memory indirect. Bit manipulation instructions use register direct, register indirect, or absolute addressing mode to specify an operand, and register direct (BSET, BCLR, BNOT, and BTST instructions) or immediate (3-bit) addressing mode to specify a bit number in the operand.

Table 2.11 Addressing Modes

No.	Addressing Mode	Symbol
1	Register direct	Rn
2	Register indirect	@ERn
3	Register indirect with displacement	@(d:16,ERn)/@(d:32,ERn)
4	Register indirect with post-increment Register indirect with pre-decrement	@ERn+ @–ERn
5	Absolute address	@aa:8/@aa:16/@aa:24/@aa:32
6	Immediate	#xx:8/#xx:16/#xx:32
7	Program-counter relative	@(d:8,PC)/@(d:16,PC)
8	Memory indirect	@@aa:8

Figure 6.1 shows a block diagram of the bus controller.

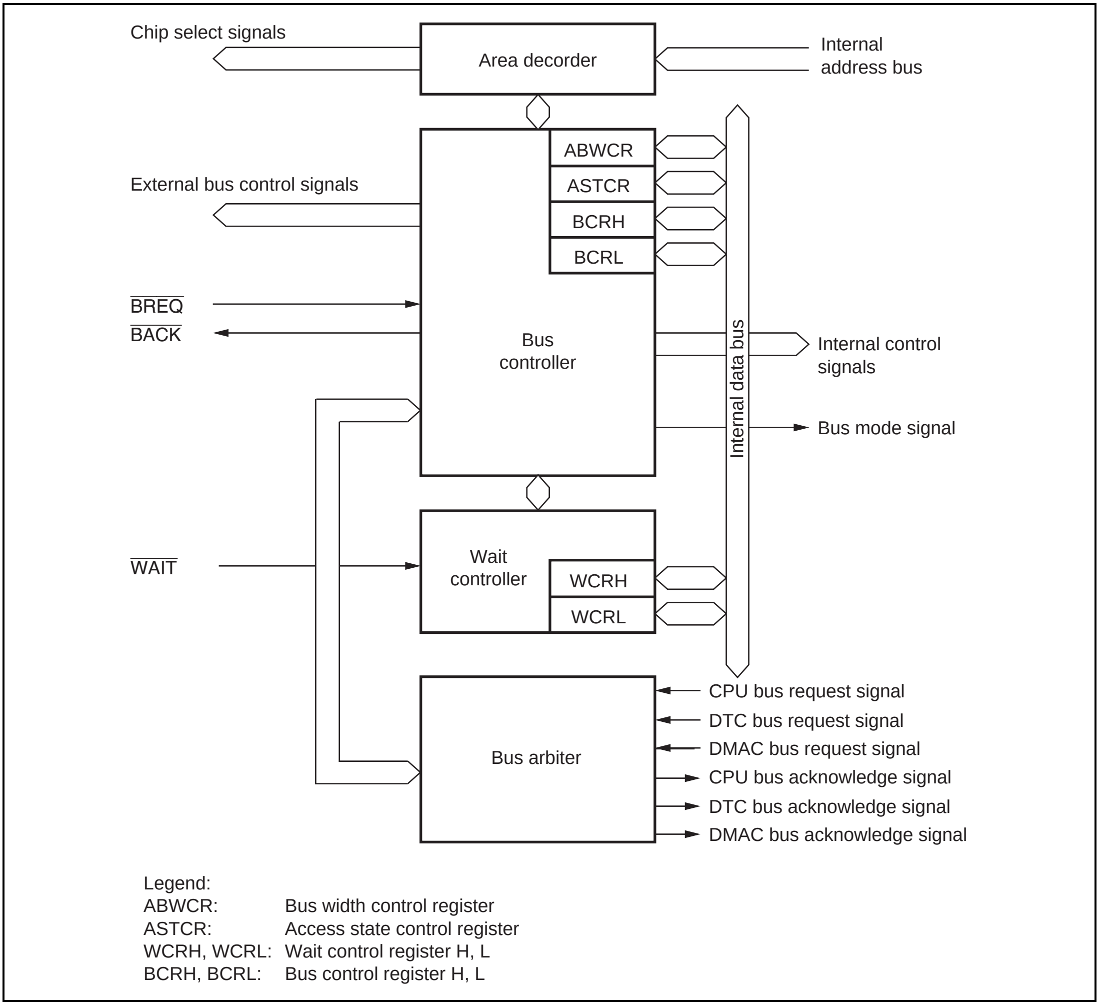
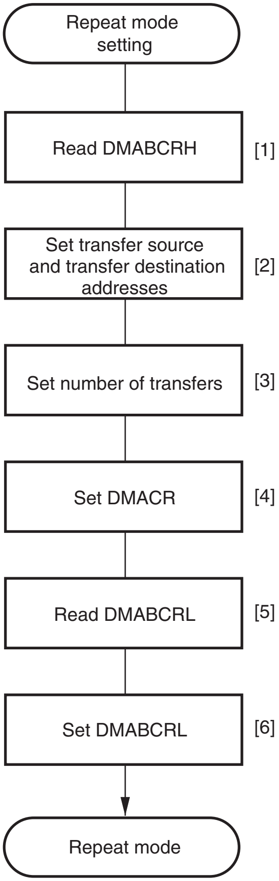


Figure 6.1 Block Diagram of Bus Controller

Bit	Bit Name	Initial Value	R/W	Description
9	DTA0	0	R/W	Data Transfer Acknowledge 0 Enables or disables clearing, when DMA transfer is performed, of the internal interrupt source selected by the channel 0 data transfer factor setting. 0: Clearing of selected internal interrupt source at time of DMA transfer is disabled 1: Clearing of selected internal interrupt source at time of DMA transfer is enabled
8	—	0	R/W	Reserved Although this bit is readable/writable, only 0 should be written here.
				Data Transfer Master Enable 1 Together with the DTE bit, this bit controls enabling or disabling of data transfer on the relevant channel. When both the DTME bit and the DTE bit are set to 1, transfer is enabled for the channel. If the relevant channel is in the middle of a burst mode transfer when an NMI interrupt is generated, the DTME bit is cleared, the transfer is interrupted, and bus mastership passes to the CPU. When the DTME bit is subsequently set to 1 again, the interrupted transfer is resumed. In block transfer mode, however, the DTME bit is not cleared by an NMI interrupt, and transfer is not interrupted. The conditions for the DTME bit being cleared to 0 are as follows: • When initialization is performed • When NMI is input in burst mode • When 0 is written to the DTME bit The condition for DTME being set to 1 is as follows: • When 1 is written to DTME after DTME is read as 0
7	DTME1	0	R/W	Data Transfer Master Enable 1 Enables or disables data transfer on channel 1 0: Data transfer disabled. In burst mode, cleared to 0 by an NMI interrupt 1: Data transfer enabled

Transfer requests (activation sources) consist of A/D conversion end interrupt, SCI transmission complete and reception complete interrupts, and TPU channel 0 to 2 compare match/input capture A interrupts. External requests can be set for channel B only. Figure 7.8 shows an example of the setting procedure for repeat mode.



[1] Set each bit in DMABCRH.
- Clear the FAE bit to 0 to select short address mode.
- Specify enabling or disabling of internal interrupt clearing with the DTA bit.

[2] Set the transfer source address and transfer destination address in MAR and IOAR.

[3] Set the number of transfers in ETCR.

[4] Set each bit in DMACR.
- Set the transfer data size with the DTSZ bit.
- Specify whether MAR is to be incremented or decremented with the DTID bit.
- Set the RPE bit to 1.
- Specify the transfer direction with the DTDIR bit.
- Select the activation source with bits DTF3 to DTF0.

[5] Read the DTE bit in DMABCRL as 0.

[6] Set each bit in DMABCRL.
- Clear the DTIE bit to 1.
- Set the DTE bit to 1 to enable transfer.

Figure 7.8 Example of Repeat Mode Setting Procedure

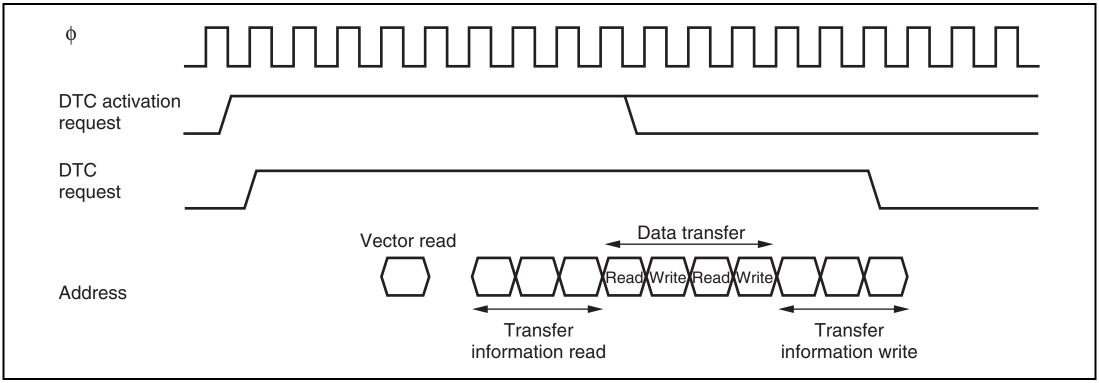


Figure 8.11 DTC Operation Timing (Example of Block Transfer Mode, with Block Size of 2)

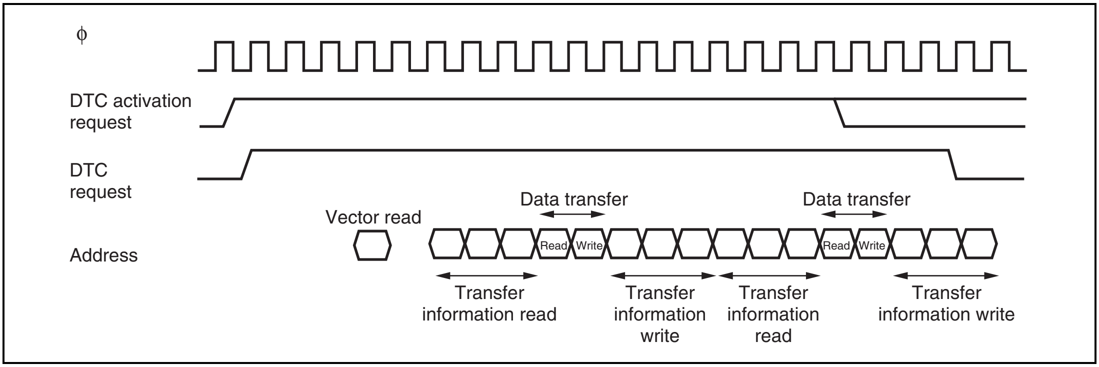

φ
DTC activation request
DTC request
Data transfer
Data transfer
Vector read
Address
Read
Write
Read
Write
Transfer information read
Transfer information write
Transfer information read
Transfer information write

Figure 8.12 DTC Operation Timing (Example of Chain Transfer)

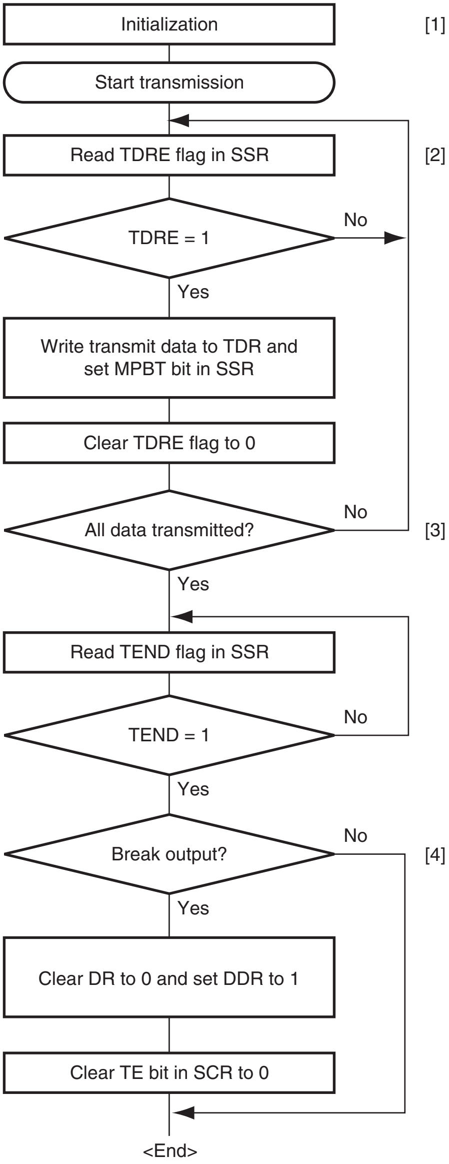
8.5.7 Number of DTC Execution States

Table 8.7 lists execution status for a single DTC data transfer, and table 8.8 shows the number of states required for each execution status.

Bit	Bit Name	Initial Value	R/W	Description
3	MPIE	0	R/W	Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 in asynchronous mode) Write 0 to this bit in Smart Card interface mode. When receive data including MPB = 0 is received, receive data transfer from RSR to RDR, receive error detection, and setting of the RERF, FER, and ORER flags in SSR, are not performed. When receive data including MPB = 1 is received, the MPB bit in SSR is set to 1, the MPIE bit is cleared to 0 automatically, and generation of RXI and ERI interrupts (when the TIE and RIE bits in SCR are set to 1) and FER and ORER flag setting are enabled.
2	TEIE	0	R/W	Transmit End Interrupt Enable Write 0 to this bit in Smart Card interface mode. TEI cancellation can be performed by reading 1 from the TDRE flag in SSR, then clearing it to 0 and clearing the TEND flag to 0, or clearing the TEIE bit to 0.
1	CKE1	0	R/W	Clock Enable 0 and 1
0	CKE0	0		Enables or disables clock output from the SCK pin. The clock output can be dynamically switched in GSM mode. For details, refer to section 13.7.9, Clock Output Control. When the GM bit in SMR is 0: 00: Output disabled (SCK pin can be used as an I/O port pin) 01: Clock output 1X: Reserved When the GM bit in SMR is 1: 00: Output fixed low 01: Clock output 10: Output fixed high 11: Clock output

Legend:

X: Don’t care



[1] SCI initialization:
The TxD pin is automatically designated as the transmit data output pin.
After the TE bit is set to 1, a frame of 1s is output, and transmission is enabled.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR. Set the MPBT bit in SSR to 0 or 1. Finally, clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:
To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear the TDRE flag to 0. Checking and clearing of the TDRE flag is automatic when the DMAC or the DTC* is activated by a transmit data empty interrupt (TXI) request, and data is written to TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the port DDR to 1, clear DR to 0, then clear the TE bit in SCR to 0.

Note: * Checking and clearing of the TDRE flag are performed automatically by the DTC when the DTC's DISEL bit is cleared to 0 and the transfer counter value is not 0. Consequently, it is necessary to use the CPU to clear the TDRE flag if DISEL is set to 1 or if the transfer counter value is 0.

Figure 13.15 Sample Multiprocessor Serial Transmission Flowchart

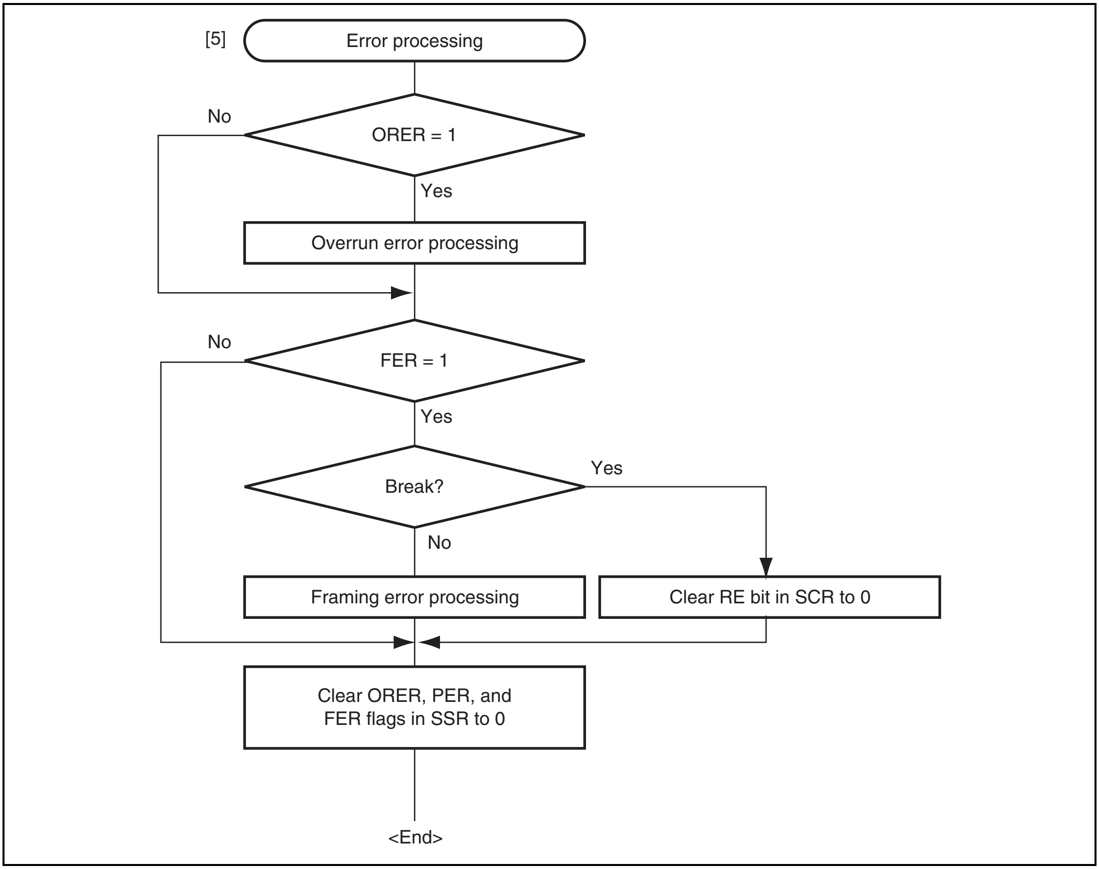


Figure 13.17 Sample Multiprocessor Serial Reception Flowchart (2)

13.7.7 Serial Data Transmission (Except for Block Transfer Mode)

As data transmission in Smart Card interface mode involves error signal sampling and retransmission processing, the operations are different from those in normal serial communication interface mode (except for block transfer mode). Figure 13.30 illustrates the retransfer operation when the SCI is in transmit mode.

1. If an error signal is sent back from the receiving end after transmission of one frame is complete, the ERS bit in SSR is set to 1. If the RIE bit in SCR is enabled at this time, an ERI interrupt request is generated. The ERS bit in SSR should be cleared to 0 by the time the next parity bit is sampled.
2. The TEND bit in SSR is not set for a frame in which an error signal indicating an abnormality is received. Data is retransferred from TDR to TSR, and retransmitted automatically.
3. If an error signal is not sent back from the receiving end, the ERS bit in SSR is not set. Transmission of one frame, including a retransfer, is judged to have been completed, and the TEND bit in SSR is set to 1. If the TIE bit in SCR is enabled at this time, a TXI interrupt request is generated. Writing transmit data to TDR transfers the next transmit data.

Figure 13.32 shows a flowchart for transmission. A sequence of transmit operations can be performed automatically by specifying the DTC or the DMAC to be activated with a TXI interrupt source. In a transmit operation, the TDRE flag is set to 1 at the same time as the TEND flag in SSR is set, and a TXI interrupt will be generated if the TIE bit in SCR has been set to 1. If the TXI request is designated beforehand as a DTC* or the DMAC activation source, the DTC* or the DMAC will be activated by the TXI request, and transfer of the transmit data will be carried out. The TDRE and TEND flags are automatically cleared to 0 when data is transferred by the DTC* or the DMAC. In the event of an error, the SCI retransmits the same data automatically. During this period, the TEND flag remains cleared to 0 and the DTC* or the DMAC is not activated. Therefore, the SCI and DTC* or the DMAC will automatically transmit the specified number of bytes in the event of an error, including retransmission. However, the ERS flag is not cleared automatically when an error occurs, and so the RIE bit should be set to 1 beforehand so that an ERI request will be generated in the event of an error, and the ERS flag will be cleared.

When performing transfer using the DMAC or the DTC, it is essential to set and enable the DMAC or the DTC* before carrying out SCI setting. For details of the DMAC or the DTC* setting procedures, refer to section 8, Data Transfer Controller (DTC) or section 7, DMA controller (DMAC).

Note: * The Flags are automatically cleared by the DTC when the DTC's DISEL bit is cleared to 0 and the transfer counter value is not 0.

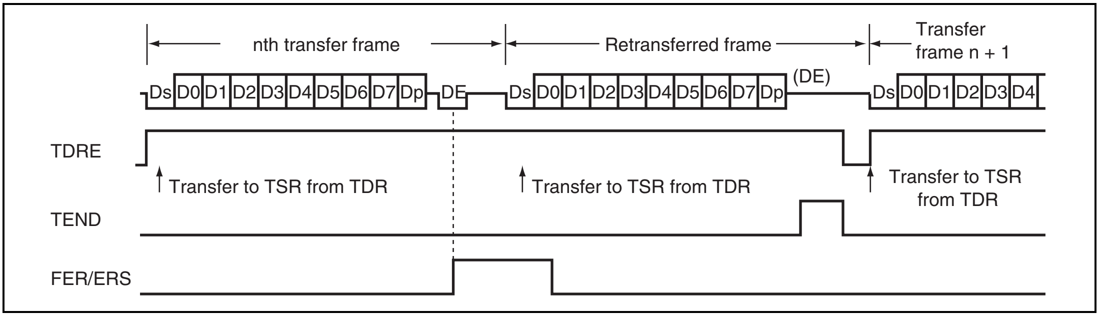


Figure 13.30 Retransfer Operation in SCI Transmit Mode

The timing for setting the TEND flag depends on the value of the GM bit in SMR. The TEND flag set timing is shown in figure 13.31.

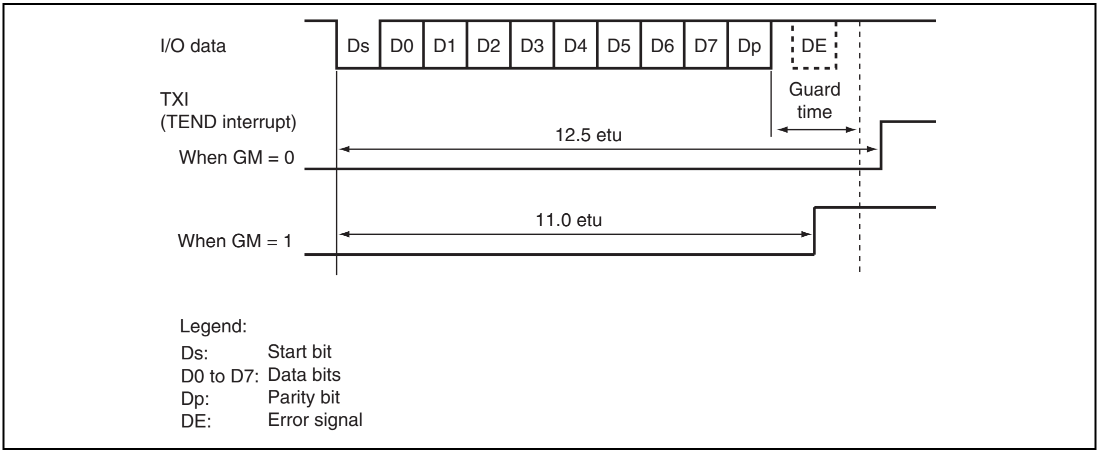


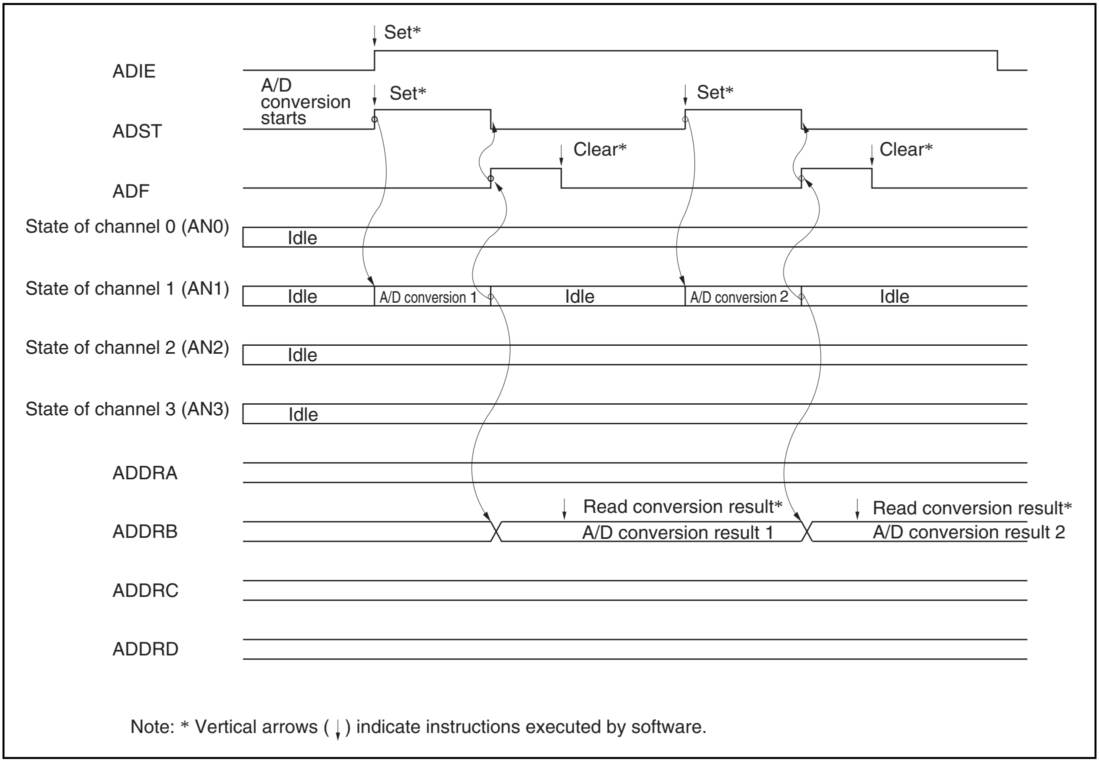
Figure 13.31 TEND Flag Generation Timing in Transmission Operation

TFP-120,	**BP-112,**			
TFP-120V	**BP-112V**			
Pin No.	**Pin No.**	**Pin Name**	**I/O**	**Bit Name**
30	J3	PA0/A16	IN	120
			Control	119
			OUT	118
31	K2	PA1/A17/TxD2	IN	117
			Control	116
			OUT	115
32	L2	PA2/A18/RxD2	IN	114
			Control	113
			OUT	112
33	H4	PA3/A19/SCK2/SUSPND	IN	111
			Control	110
			OUT	109
35	K3	P10/TIOCA0/A20/VM	IN	108
			Control	107
			OUT	106
36	L3	P11/TIOCB0/A21/VP	IN	105
			Control	104
			OUT	103
37	J4	P12/TIOCC0/TCLKA/A22/RCV	IN	102
			Control	101
			OUT	100
38	K4	P13/TIOCD0/TCLKB/A23/VPO	IN	99
			Control	98
			OUT	97
39	L4	P14/TIOCA1/$\overline{\text{IRQ0}}$	IN	96
			Control	95
			OUT	94
40	H5	P15/TIOCB1/TCLKC/FSE0	IN	93
			Control	92
			OUT	91
41	J5	P16/TIOCA2/$\overline{\text{IRQ1}}$	IN	90
			Control	89
			OUT	88

15.3.2 USB Control Register (UCTLR)

UCTLR is used to select USB data input/output pin and USB operating clock, specify SOF marker function, and controls the USB module reset. UCTLR can be read from or written to even in USB module stop mode. For details on UCTLR setting procedure, refer to section 15.5, Communication Operation.

Bit	Bit Name	Initial Value	R/W	Description
7	FADSEL	0	R/W	I/O Analog or Digital Selection Selects USB function data I/O pins 0: USD+ and USD- are used as data I/O pins 1: Control I/O ports 1 and A compatible with Philips Corp. transceiver are connected to data I/O pins. P17 (output) → $\overline{\text{OE}}$: Output enable P15 (output) → FSE0: SE0 setting P13 (output) → VPO: Data+ output P12 (input) ← PCV: Differential input P11 (input) ← VP: Data+ input P10 (input) ← VM: Data– input PA3 (output) → SUSPND: Suspend enable Ports 1 and A are prioritized to address outputs. Accordingly, before setting FADSEL to 1, disable A23 to A19 output via PFCR. In addition, FADSEL must be set during USB module stop mode.
6	SFME	0	R/W	Start Of Frame (SOF) Marker Function Enable Controls the SOF marker function. If SFME is set to 1, the SOF interrupt flag can be set to 1 every 1ms even if the SOF packet has been broken. Note, however, that UTSR stores a time stamp when the correct SOF packet is received. The USB does not support UTSR automatic update function when the SOF packet is broken. To set SFME the first time, SFME must be set after SOF flag detection. SFME must be cleared to 0 when the suspension is detected. To set SFME after resume detection, SFME must also be set after SOF flag detection. 0: Disables the SOF marker function 1: Enables the SOF marker function



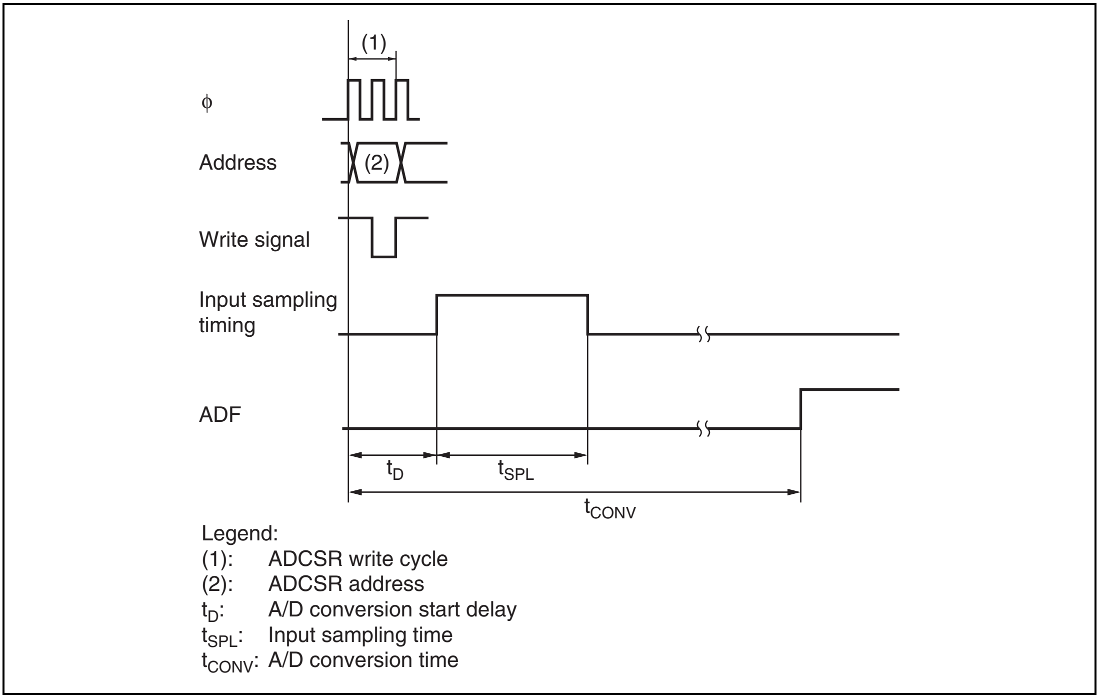
Note: * Vertical arrows (↓) indicate instructions executed by software.

Figure 16.3 A/D Conversion Timing (Single-Chip Mode, Channel 1 Selected)

16.5.2 Scan Mode

In scan mode, A/D conversion is to be performed sequentially on the specified channels (four channels maximum). The operations are as follows.

1. When the ADST bit is set to 1 by software, TPU, or external trigger input, A/D conversion starts on the first channel in the group (AN0 when CH3 and CH2 = 00, AN4 when CH3 and CH2 = 01, or AN8 when CH3 and CH2 = 10).
2. When A/D conversion for each channel is completed, the result is sequentially transferred to the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF flag is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested after A/D conversion ends. Conversion of the first channel in the group starts again.
4. Steps 2 to 3 are repeated as long as the ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion stops.



Legend:
(1): ADCSR write cycle
(2): ADCSR address
t_D: A/D conversion start delay
t_{SPL}: Input sampling time
t_{CONV}: A/D conversion time

Figure 16.5 A/D Conversion Timing

Table 16.3 A/D Conversion Time (Single Mode)

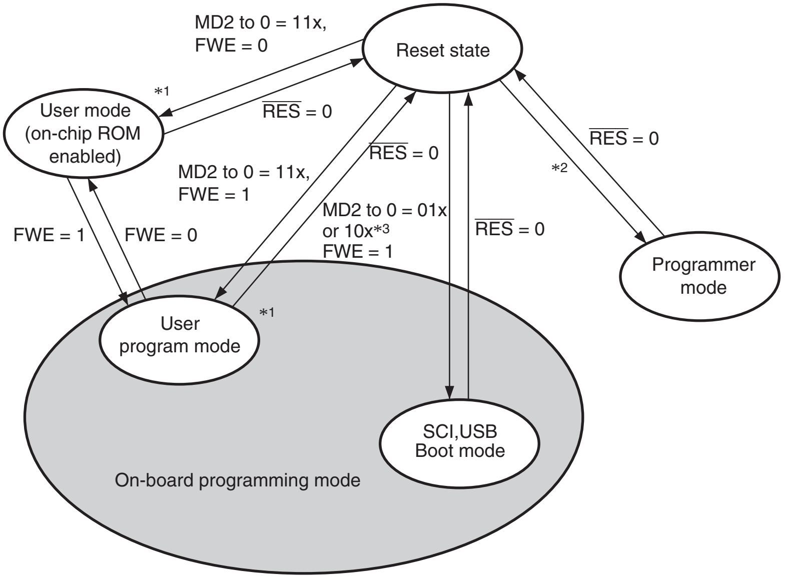
Item	Symbol	CKS1 = 0						CKS1 = 1					
		CKS0 = 0			CKS0 = 1			CKS0 = 0			CKS0 = 1		
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max
A/D conversion start delay	t_D	18	—	33	10	—	17	6	—	9	4	—	5
Input sampling time	t_{SPL}	—	127	—	—	63	—	—	31	—	—	15	—
A/D conversion time	t_{CONV}	515	—	530	259	—	266	131	—	134	67	—	68

Note: All values represent the number of states.

19.2 Mode Transitions

When the mode pins and the FWE pin are set in the reset state and a reset-start is executed, this LSI enters an operating mode as shown in figure 19.2. In user mode, flash memory can be read but not programmed or erased. The boot and user program modes are provided as modes to write and erase the flash memory.

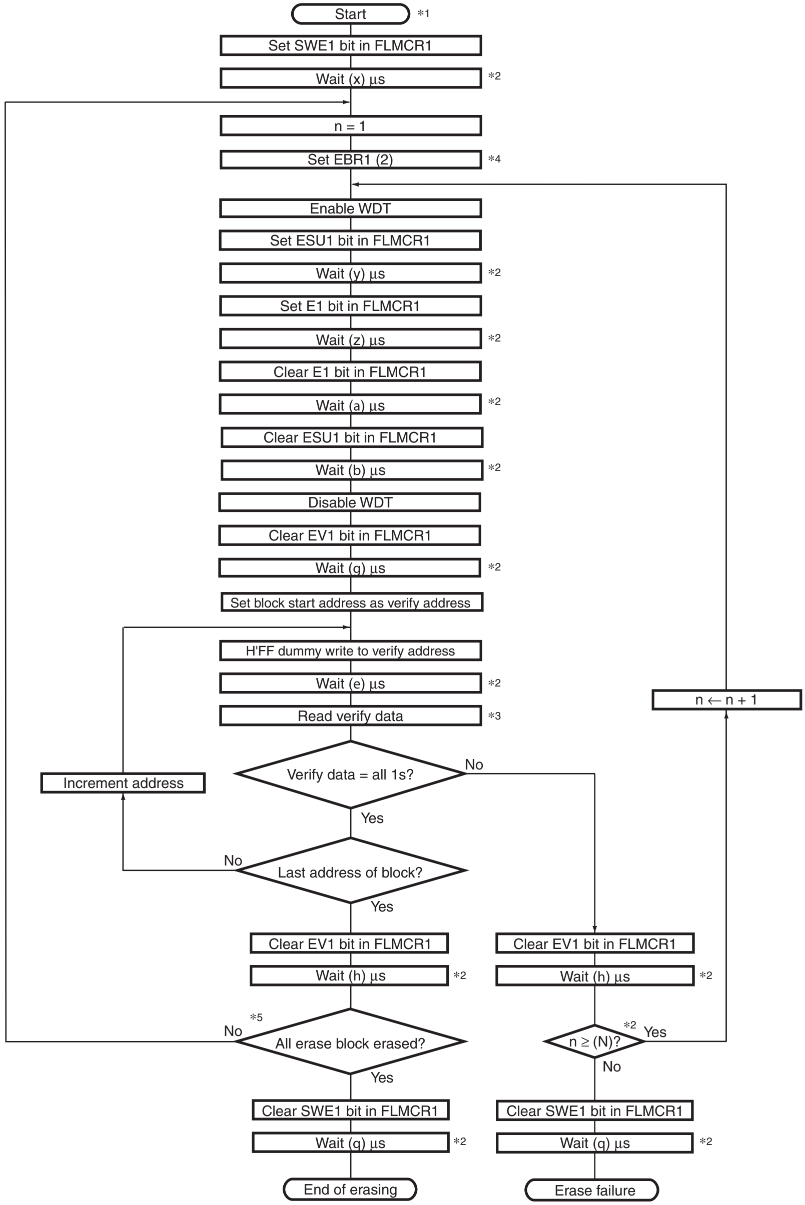
The differences between boot mode and user program mode are shown in table 19.1. Boot mode and user program mode operations are shown in figures 19.3 and 19.4, respectively.



Notes: Only make a transition between user mode and user program mode when the CPU is not accessing the flash memory.

1. RAM emulation possible
2. MD2 to MD0 = 000, PF3, PF0, P16, P14 = 1100
3. 10x applies only to the HD64F2215RU, HD64F2215TU and HD64F2215CU with 24-MHz system clock.

Figure 19.2 Flash Memory State Transitions



Notes: 1. Pre-write (clearing data in the block to be erased to 0) is not required.
2. x, y, z, a, b, g, e, h, q, and N are shown in section 24.8, Flash Memory Characteristics.
3. Veryfy data is read in 16 bits.
4. Only 1 bit in the EBR register must be set. Two or more bits in EBR cannot be set.
5. Erasure is performed in block units. To erase multiple blocks, each block must be erased sequentially.

Figure 19.12 Erase/Erase-Verify Flowchart

The FLMCR1, FLMCR2, EBR1 and EBR2 settings are retained, but program mode or erase mode is aborted at the point at which the error occurred. Program mode or erase mode cannot be re-entered by re-setting the P1 or E1 bit. However, PV1 and EV1 bit setting is enabled, and a transition can be made to verify mode.

19.10 Interrupt Handling when Programming/Erasing Flash Memory

All interrupts, including NMI interrupt is disabled when flash memory is being programmed or erased (when the P1 or E1 bit is set in FLMCR1), and while the boot program is executing in boot mode*[1], to give priority to the program or erase operation. There are three reasons for this:

1. Interrupt during programming or erasing might cause a violation of the programming or erasing algorithm, with the result that normal operation could not be assured.
2. In the interrupt exception handling sequence during programming or erasing, the vector would not be read correctly*[2], possibly resulting in CPU runaway.
3. If interrupt occurred during boot program execution, it would not be possible to execute the normal boot mode sequence.

Notes: 1. Interrupt requests must be disabled inside and outside the CPU until the programming control program has completed programming.

2. The vector may not be read correctly in this case for the following two reasons:
    - If flash memory is read while being programmed or erased (while the P1 or E1 bit is set in FLMCR1), correct read data will not be obtained (undetermined values will be returned).
    - If the interrupt entry in the vector table has not been programmed yet, interrupt exception handling will not be executed correctly.

19.11 Programmer Mode

In programmer mode, a PROM programmer can perform programming/erasing via a socket adapter, just like for a discrete flash memory. Use a PROM programmer which supports the Renesas Electronics 256-kbyte flash memory on-chip MCU device type. Memory map in programmer mode is shown in figure 19.13.

Register Name	Abbreviation	Number of Bits	Address	Data Bus Width	Number of Access States	Module
DTC transfer count register A	CRA	16	H'EBC0 to H'EFBF	16/32	1	DTC
DTC transfer count register B	CRB	16		16/32	1	
D/A data register_0	DADR_0	8	H'FDAC	8	2	D/A
D/A data register _1	DADR_1	8	H'FDAD	8	2	
D/A control register	DACR	8	H'FDAE	8	2	
Serial control register X	SCRX	8	H'FDB4	8	2	FLASH
Standby control register	SBYCR	8	H'FDE4	8	2	SYSTEM
System control register	SYSCR	8	H'FDE5	8	2	
System clock control register	SCKCR	8	H'FDE6	8	2	
Mode control register	MDCR	8	H'FDE7	8	2	
Module stop control register A	MSTPCRA	8	H'FDE8	8	2	
Module stop control register B	MSTPCRB	8	H'FDE9	8	2	
Module stop control register C	MSTPCRC	8	H'FDEA	8	2	
Pin function control register	PFCR	8	H'FDEB	8	2	BSC
Low power control register	LPWRCR	8	H'FDEC	8	2	SYSTEM
Serial extended mode register_0 (In H8S/2215)	SEMR_0	8	H'FDF8	8	2	SCI_0
Serial extended mode register A_0 (In H8S/2215R, H8S/2215T and H8S/2215C)	SEMRA_0	8	H'FDF8	8	2	
Serial extended mode register B_0 (In H8S/2215R, H8S/2215T and H8S/2215C)	SEMRB_0	8	H'FDF9	8	2	
IRQ sense control register H	ISCRH	8	H'FE12	8	2	INT
IRQ sense control register L	ISCRL	8	H'FE13	8	2	
IRQ enable register	IER	8	H'FE14	8	2	
IRQ status register	ISR	8	H'FE15	8	2	
DTC enable register A	DTCERA	8	H'FE16	8	2	DTC
DTC enable register B	DTCERB	8	H'FE17	8	2	
DTC enable register C	DTCERC	8	H'FE18	8	2	
DTC enable register D	DTCERD	8	H'FE19	8	2	
DTC enable register E	DTCERE	8	H'FE1A	8	2	
DTC enable register F	DTCERF	8	H'FE1B	8	2	
DTC vector register	DTVECR	8	H'FE1F	8	2	
Port 1 data direction register	P1DDR	8	H'FE30	8	2	PORT
Port 3 data direction register	P3DDR	8	H'FE32	8	2	
Port 7 data direction register	P7DDR	8	H'FE36	8	2	

23.2 Register Bits

Register addresses and bit names of the on-chip peripheral modules are described below.

Each line covers eight bits, so 16-bit registers are shown as two lines and 32-bit registers as four lines.