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	Active
Core Processor	H8S/2000
Core Size	16-Bit
Speed	24MHz
Connectivity	SCI, SmartCard, USB
Peripherals	DMA, POR, PWM, WDT
Number of I/O	37
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 6x10b
Oscillator Type	External
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LFQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df2212ufp24v

Item	Page	Revision (See Manual for Details)
17.6 On-Board Programming Modes Table 17.3 Setting On-Board Programming Modes	567	Table amended **Mode** SCI boot mode (HD64F2218, HD64F2212, HD64F2211) USB boot mode (HD64F2218U, HD64F2218CU, HD64F2217CU, HD64F2212U, HD64F2212CU, HD64F2211U, HD64F2211CU, HD64F2210CU)
17.6.1 SCI Boot Mode (HD64F2218, HD64F2212, and HD64F2211)	569	Description amended 5. In boot mode, a part of the on-chip RAM area (four kbytes) is used by the boot program. The area to which the programming control program is transferred from the host is 8 kbytes (H'FFC000 to H'FFDFFF) in the HD64F2218 and HD64F2212 and 4 kbytes (H'FFD000 to H'FFDFFF) in the HD64F2211 . The boot program area cannot be used until the execution state in boot mode switches to the programming control program.
17.6.2 USB Boot Mode (HD64F2218U, HD64F2212U, and HD64F2211U)	573	Description amended 4. In boot mode, the 4-kbyte on-chip RAM area H'FFE000 to H'FFEFBF is used by the boot program. The programming control program is transferred from the host stored in the 8-kbyte area H'FFC000 to H'FFDFFF in the HD64F2218U, HD64F2218CU, HD64F2212U, and HD64F2212CU and the 4-kbyte area H'FFD000 to H'FFDFFF in the HD64F2211U, HD64F2211CU and HD64F2210CU.

Appendix

H8S/2212 Group

ROM	Part No.	ROM	RAM	Remarks
Flash memory Version	HD64F2212	128 kbytes	12 kbytes	SCI boot mode
	HD64F2212U	128 kbytes	12 kbytes	USB boot mode
	HD64F2212CU	128 kbytes	12 kbytes	USB boot mode
	HD64F2211	64 kbytes	8 kbytes	SCI boot mode
	HD64F2211U	64 kbytes	8 kbytes	USB boot mode
	HD64F2211CU	64 kbytes	8 kbytes	USB boot mode
	HD64F2210CU	32 kbytes	8 kbytes	USB boot mode
Masked ROM Version	HD6432211	64 kbytes	8 kbytes	—
	HD6432210	32 kbytes	4 kbytes	—
	HD6432210S	32 kbytes	4 kbytes	—

- General I/O ports
  I/O pins: 69 for the H8S/2218 Group, 37 for the H8S/2212 Group
- Supports various power-down states
- Compact package

Package	Code*	Body Size	Pin Pitch	Remarks
TQFP-100	TFP-100G, TFP-100GV	12.0 × 12.0 mm	0.4 mm	H8S/2218 Group
P-LFBGA-112	BP-112, BP-112V	10.0 × 10.0 mm	0.8 mm	
LQFP-64	FP-64E, FP-64EV	10.0 × 10.0 mm	0.5 mm	H8S/2212 Group
VQFN-64	TNP-64B, TNP-64BV	8.0 × 8.0 mm	0.4 mm	

Note: * A V appended to the end of the package code indicates a lead-free version.

Pin No.		Pin Name*			
TFP-100G, TFP-100GV	**BP-112, BP-112V**	**Modes 4, 5**	**Mode 6**	**Mode 7**	**Programmer Mode**
56	H10	P71/$\overline{CS5}$	P71/$\overline{CS5}$	P71	NC
57	H11	$\overline{STBY}$	$\overline{STBY}$	$\overline{STBY}$	VCC
58	G8	$\overline{RES}$	$\overline{RES}$	$\overline{RES}$	$\overline{RES}$
59	G9	VSS	VSS	VSS	VSS
60	G11	XTAL	XTAL	XTAL	XTAL
61	G10	EXTAL	EXTAL	EXTAL	EXTAL
62	F9	VCC	VCC	VCC	VCC
63	F11	P70/$\overline{CS4}$	P70/$\overline{CS4}$	P70	NC
64	F10	PE0/D0	PE0/D0	PE0	D0
65	F8	PE1/D1	PE1/D1	PE1	D1
66	E11	PE2/D2	PE2/D2	PE2	D2
67	E10	PE3/D3	PE3/D3	PE3	D3
68	E9	PE4/D4	PE4/D4	PE4	D4
69	D11	PE5/D5	PE5/D5	PE5	D5
70	E8	PE6/D6	PE6/D6	PE6	D6
71	D10	PE7/D7	PE7/D7	PE7	D7
72	C11	D8	D8	PD0	NC
73	D9	D9	D9	PD1	NC
74	C10	D10	D10	PD2	NC
75	B11	D11	D11	PD3	NC
76	B10	D12	D12	PD4	NC
77	A10	D13	D13	PD5	NC
78	D8	D14	D14	PD6	NC
79	B9	D15	D15	PD7	NC
80	A9	FWE	FWE	FWE	FWE
81	C8	NMI	NMI	NMI	VCC
82	B8	EMLE/NC	EMLE/NC	EMLE/NC	VSS
83	A8	TDO/NC	TDO/NC	TDO/NC	NC
84	D7	TCK/NC	TCK/NC	TCK/NC	VCC
85	C7	TMS/NC	TMS/NC	TMS/NC	VCC

The instructions BSET, BCLR, BNOT, BST, and BIST perform the following operations in the order shown:

1. Read data in byte units
2. Perform bit manipulation on the read data according to the instruction
3. Write data in byte units

Example: Using the BCLR instruction to clear pin 14 only of P1DDR for port 1

P1DDR is an 8-bit register that contains write-only bits. It is used to specify the I/O setting of the individual pins in port 1. Reading produces invalid data. Attempting to read from P1DDR returns undefined values.

In this example, the BCLR instruction is used to set pin 14 as an input port. Let us assume that pins 17 to 14 are presently set as output pins and pins 13 to 10 are set as input pins. Thus, the value of P1DDR is initially H'F0.

	P17	**P16**	**P15**	**P14**	**P13**	**P12**	**P11**	**P10**
I/O	Output	Output	Output	Output	Input	Input	Input	Input
P1DDR	1	1	1	1	0	0	0	0

To change pin 14 from an output pin to an input pin, the value of bit 4 in P1DDR must be changed from 1 to 0 (H'F0 to H'E0). Now assume that the BCLR instruction is used to clear bit 4 in P1DDR to 0.

```
BCLR    #4, @P1DDR
```

However, using the above bit manipulation instruction on the write-only register P1DDR can cause problems, as described below.

The BCLR instruction first reads data from P1DDR in byte units, but in this case the read values are undefined. These undefined values can be 0 or 1 for each bit in the register, but there is no way of telling which. Since all of the bits in P1DDR are write-only, undefined values are returned for all of the bits when the register is read. In this example the value of P1DDR is H'F0, but we will assume that the value returned when the register was read is H'F8, which would give bit 3 a value of 1.

Section 4 Exception Handling

4.1 Exception Handling Types and Priority

As table 4.1 indicates, exception handling may be caused by a reset, trace, trap instruction, or interrupt. Exception handling is prioritized as shown in table 4.1. If two or more exceptions occur simultaneously, they are accepted and processed in order of priority. Exception sources, the stack structure, and operation of the CPU vary depending on the interrupt control mode. For details on the interrupt control mode, refer to section 5, Interrupt Controller.

Table 4.1 Exception Types and Priority

Priority	Exception Type	Start of Exception Handling
High	Reset	Starts immediately after a low-to-high transition at the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$* pin, or when the watchdog timer overflows. The CPU enters the reset state when the $\overline{\text{RES}}$ pin is low. The CPU enters the manual reset state when the $\overline{\text{MRES}}$ pin* is low.
	Trace	Starts when execution of the current instruction or exception handling ends, if the trace (T) bit in the EXR is set to 1. This is enabled only in trace interrupt control mode 2. Trace exception processing is not performed after RTE instruction execution.
	Interrupt	Starts when execution of the current instruction or exception handling ends, if an interrupt request has been issued. Note that after executing the ANDC, ORC, XORC, or LDC instruction or at the completion of reset exception processing, no interrupt is detected.
Low	Trap instruction (TRAPA)	Started by execution of a trap instruction (TRAPA). Trap exception processing is always accepted in program execution state.

Note: * Supported only by the H8S/2218 Group.

4.2 Exception Sources and Exception Vector Table

Different vector addresses are assigned to different exception sources. Table 4.2 lists the exception sources and their vector addresses. Since the usable modes differ depending on the product, for details on each product, refer to section 3, MCU Operating Modes.

7. The CPU generates a vector address for the accepted interrupt and starts execution of the interrupt handling routine at the address indicated by the contents of the vector address in the vector table.

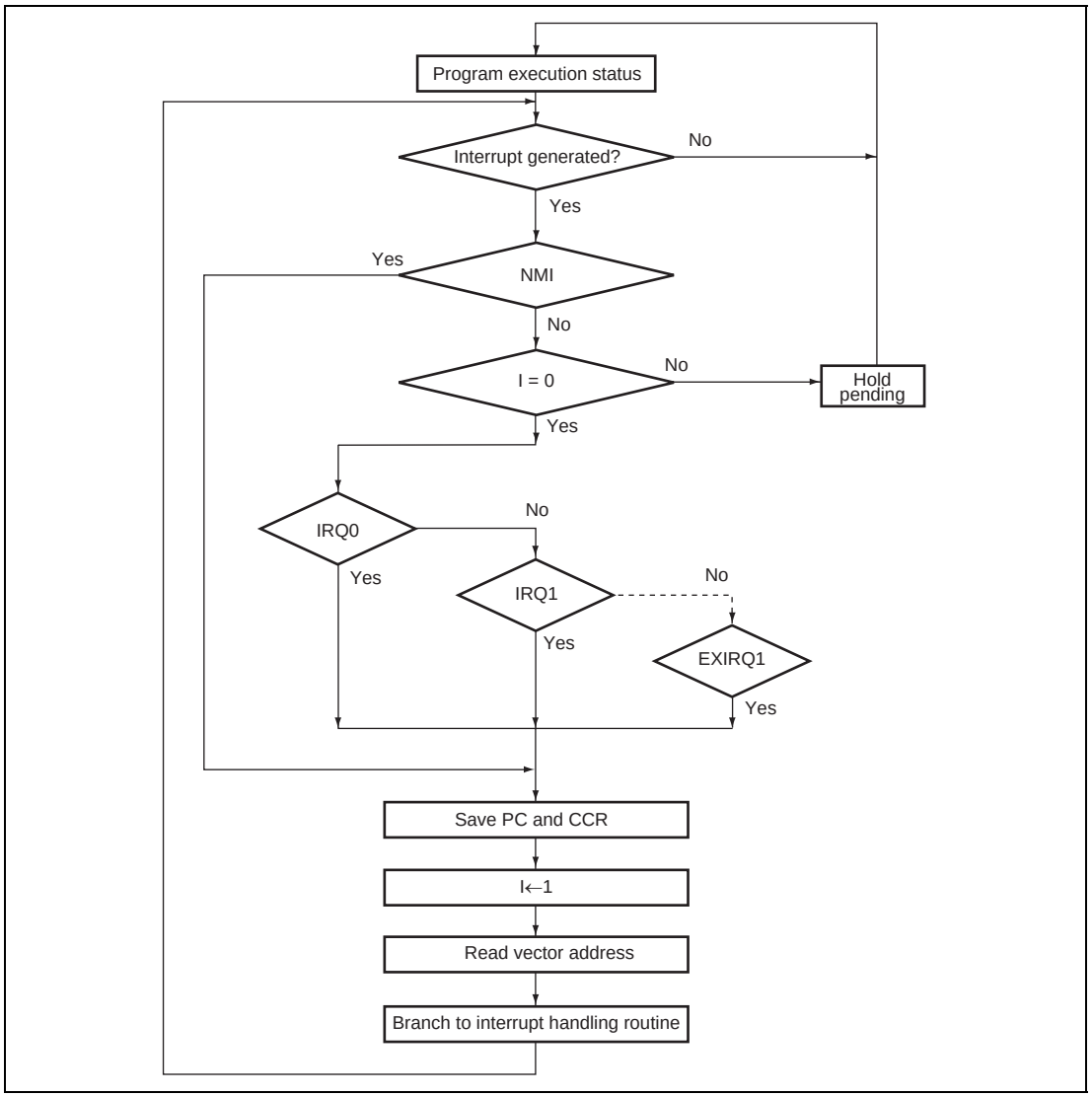


Figure 5.4 Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 0

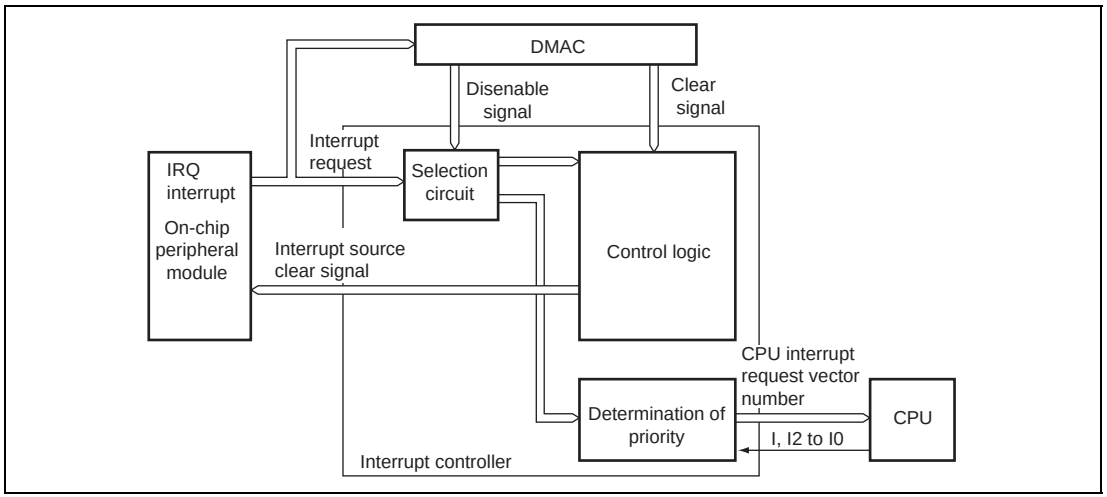


Figure 5.7 Interrupt Control for DMAC

Selection of Interrupt Source: An activation factor is directly input to each channel of the DMAC. The activation factors for each channel of the DMAC are selected by the DTF3 to DTF0 bits of DMACR. The DTA bit of DMABCR can be used to select whether the selected activation factors are managed by the DMAC. By setting the DTA bit to 1, the interrupt factor which was the activation factor for that DMAC cannot act as the CPU interrupt factor.

Interrupt factors other than the interrupts managed by the DMAC is CPU interrupt request.

Determination of Priority: The activation source is directly input to each channel of DMAC.

Operation Order: If the same interrupt is selected as the DMAC activation factor or CPU interrupt factor, these operate independently. They operate in accordance with the respective operating states and bus priorities.

Table 5.6 shows the interrupt factor clear control and selection of interrupt factors by specification of the DTA bit of DMAC's DMABCR.

5.7.6 NMI Interrupt Usage Notes

The NMI interrupt is part of the exception processing performed cooperatively by the LSI's internal interrupt controller and the CPU when the system is operating normally under the specified electrical conditions. No operations, including NMI interrupts, are guaranteed when operation is not normal (runaway status) due to software problems or abnormal input to the LSI's pins. In such cases, the LSI may be restored to the normal program execution state by applying an external reset.

the RAME bit is cleared to 0, the on-chip RAM is disabled and the corresponding space becomes external space.

Only the basic bus interface can be used for the area 7.

6.4.4 Chip Select Signals

In the H8S/2218 Group chip select signals ($\overline{CS0}$ to $\overline{CS5}$) can be output to areas 0 to 5, the signal being driven low when the corresponding external space area is accessed. Figure 6.3 shows an example of $\overline{CSn}$ (n = 0 to 5) output timing. Enabling or disabling of the $\overline{CSn}$ signal is performed by setting the data direction register (DDR) for the port corresponding to the particular $\overline{CSn}$ pin.

In ROM-disabled extended mode, the $\overline{CS0}$ pin is placed in the output state after a power-on reset. Pins $\overline{CS1}$ to $\overline{CS5}$ are placed in the input state after a power-on reset, and so the corresponding DDR should be set to 1 when outputting signals $\overline{CS1}$ to $\overline{CS5}$.

In ROM-enabled extended mode, pins $\overline{CS0}$ to $\overline{CS5}$ are all placed in the input state after a power-on reset, and so the corresponding DDR should be set to 1 when outputting signals $\overline{CS0}$ to $\overline{CS5}$. For details, see section 8, I/O Ports.

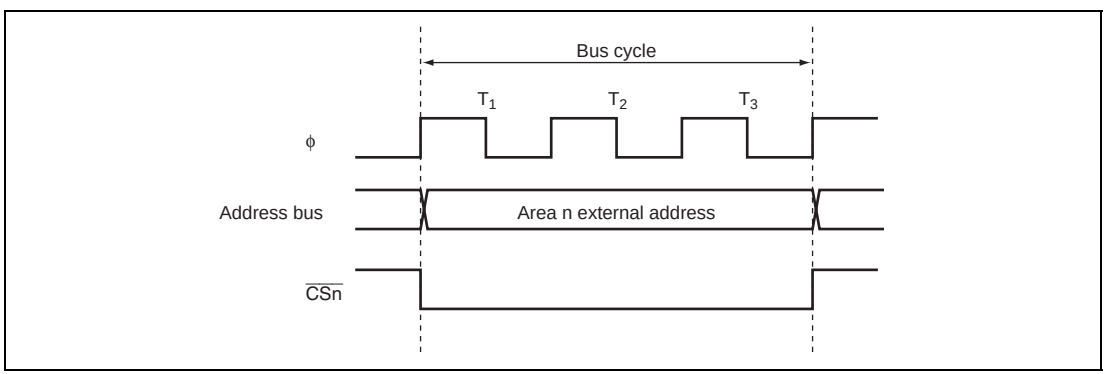


Figure 6.3 $\overline{CSn}$ Signal Output Timing (n = 0 to 5)

Bit	Bit Name	Initial Value	R/W	Description
1	P71DDR	0	W	(H8S/2218 Group)
0	P70DDR	0	W	Setting a P7DDR bit to 1 makes the corresponding port 7 pin an output pin, while clearing the bit to 0 makes the pin an input pin. (H8S/2212 Group) Reserved These bits are undefined and cannot be modified.

8.4.2 Port 7 Data Register (P7DR)

P7DR stores output data for the port 7 pins.

Bit	Bit Name	Initial Value	R/W	Description
7	P77DR	0	R/W	(H8S/2218 Group)
6	P76DR	0	R/W	Reserved These bits are undefined and cannot be modified.
5	P75DR	0	R/W	(H8S/2212 Group) Store output data for the port 7 pins.
4	P74DR	0	R/W	(H8S/2218 Group) Stores output data for the port 7 pins. (H8S/2212 Group) Reserved This bit is undefined and cannot be modified.
3, 2	—	Undefined	—	Reserved These bits are undefined and cannot be modified.
1	P71DR	0	R/W	(H8S/2218 Group)
0	P70DR	0	R/W	Store output data for the port 7 pins. (H8S/2212 Group) Reserved These bits are undefined and cannot be modified.

8.6 Port A

In the H8S/2218 Group, the port A is a 4-bit I/O port also functioning as address bus (A19 to A16) output pins and SCI I/O pins. In the H8S/2212 Group, the port A is a 3-bit I/O port also functioning as SCI I/O pins. The port A has the following registers.

- Port A data direction register (PADDR)
- Port A data register (PADR)
- Port A register (PORTA)
- Port A pull-up MOS control register (PAPCR)
- Port A open-drain control register (PAODR)

8.6.1 Port A Data Direction Register (PADDR)

PADDR specifies input or output for the pins of the port A.

Since PADDR is a write-only register, the bit manipulation instructions must not be used to write PADDR. For details, see section 2.9.4, Accessing Registers Containing Write-Only Bits.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	Undefined	—	Reserved These bits are undefined and cannot be modified.
3	PA3DDR	0	W	(H8S/2218 Group) Modes 4 to 6: If address output is enabled by the setting of bits AE3 to AE0 in PFCR, the corresponding port A pins are address outputs. When address output is disabled, setting a PADDR bit to 1 makes the corresponding port A pin an output port, while clearing the bit to 0 makes the pin an input port. Mode 7: Setting a PADDR bit to 1 makes the corresponding port A pin an output port, while clearing the bit to 0 makes the pin an input port. (H8S/2212 Group) Setting a PADDR bit to 1 makes the corresponding port A pin an output port, while clearing the bit to 0 makes the pin an input port.
2	PA2DDR	0	W	
1	PA1DDR	0	W	
0	PA0DDR*	0	W	

Note: * Reserved in the H8S/2212 Group. If this bit is read, an undefined value will be read. This bit cannot be modified.

Status Flag Clearing Timing: After a status flag is read as 1 by the CPU, it is cleared by writing 0 to it. When the DMAC is activated, the flag is cleared automatically. Figure 9.42 shows the timing for status flag clearing by the CPU, and figure 9.43 shows the timing for status flag clearing by the DMAC.

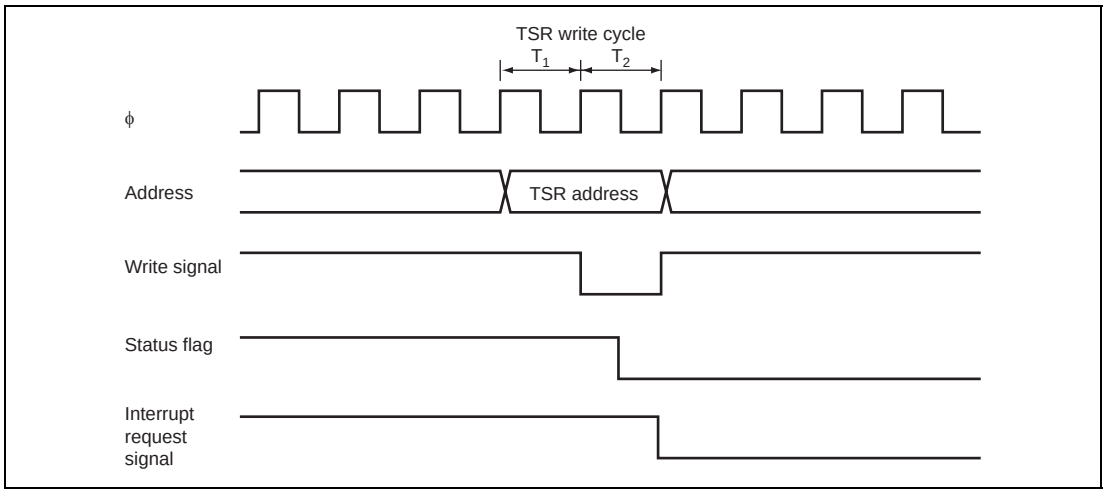


Figure 9.42 Timing for Status Flag Clearing by CPU

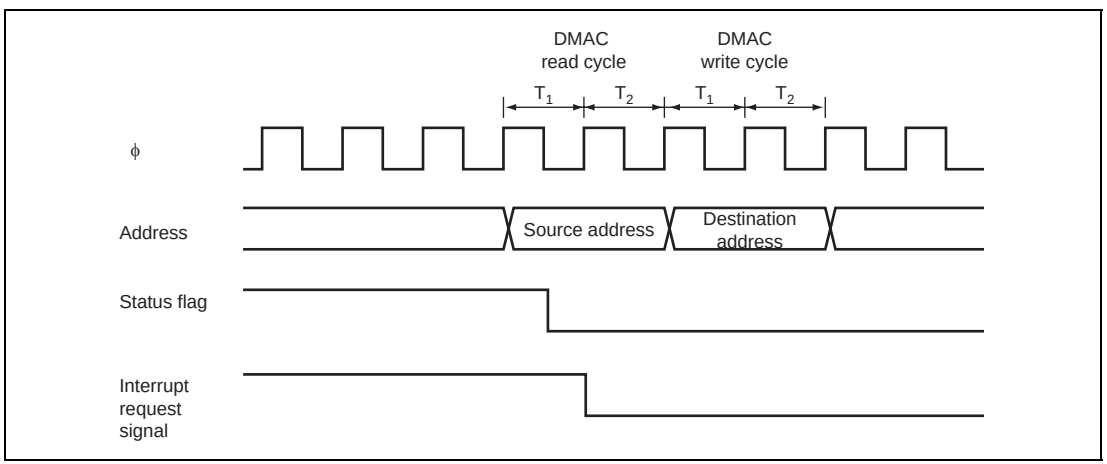
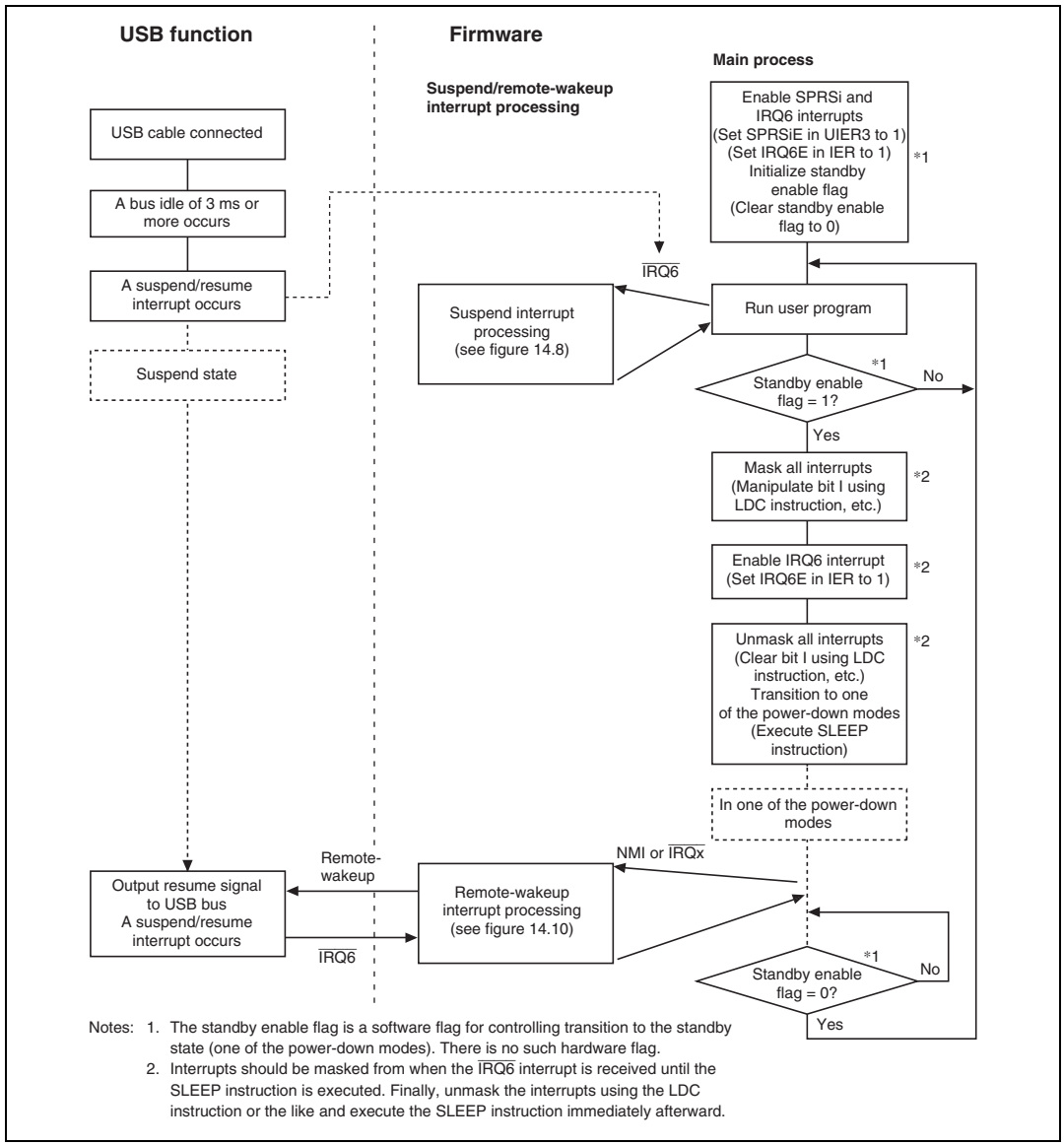


Figure 9.43 Timing for Status Flag Clearing by DMAC Activation

TFP-100G TFP-100GV Pin No.	BP-112 BP-112V Pin No.	Pin Name	I/O	Bit Name
55	H9	P74/$\overline{\text{MRES}}$	IN	60
			Control	59
			OUT	58
56	H10	P71/$\overline{\text{CS5}}$	IN	57
			Control	56
			OUT	55
57	H11	$\overline{\text{STBY}}$	IN	54
58	G8	$\overline{\text{RES}}$	IN	53
63	F11	P70/$\overline{\text{CS4}}$	IN	52
			Control	51
			OUT	50
64	F10	PE0/D0	IN	49
			Control	48
			OUT	47
65	F8	PE1/D1	IN	46
			Control	45
			OUT	44
66	E11	PE2/D2	IN	43
			Control	42
			OUT	41
67	E10	PE3/D3	IN	40
			Control	39
			OUT	38
68	E9	PE4/D4	IN	37
			Control	36
			OUT	35
69	D11	PE5/D5	IN	34
			Control	33
			OUT	32

(3) Suspend and Remote-Wakeup Operations

Figures 14.9 and 14.10 are flowcharts of the suspend and remote-wakeup operations. If the USB bus enters a non-suspend state from the suspend state due to a remote-wakeup signal from this function, perform the operations shown below.



Notes: 1. The standby enable flag is a software flag for controlling transition to the standby state (one of the power-down modes). There is no such hardware flag.
2. Interrupts should be masked from when the IRQ6 interrupt is received until the SLEEP instruction is executed. Finally, unmask the interrupts using the LDC instruction or the like and execute the SLEEP instruction immediately afterward.

Figure 14.9 Example Flowchart of Suspend and Remote-Wakeup Operations

(b) EP2 DMA Transfer Procedure

The DMAC transfer unit should be one packet. Therefore, set the number of transfers so that it is equal to or less than the maximum packet size of each endpoint.

1. Wait for the UIFR1/EP2READY flag to be set.
2. DMAC settings for EP2 data transfer (such as auto-request and address setting). Read value of UESZ2 and specify number of transfers to match size of received data (64 bytes or less).
3. Activate the DMAC (write 1 to DTE after reading DTE as 0).
4. DMA transfer (transfer of 64 bytes or less).
5. Write 1 to the UTRG0/EP2RDFN bit after the DMA transfer is completed.
6. Repeat steps 1 to 5 above.

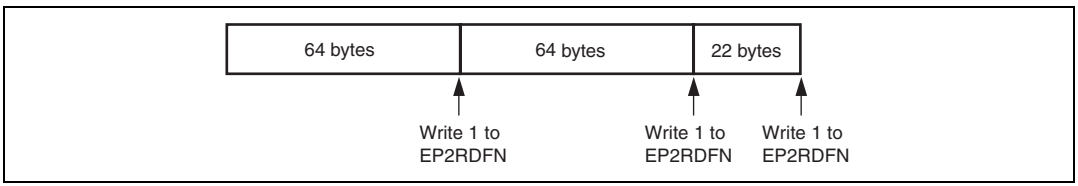


Figure 14.25 EP2RDFN Operation in UTRG0 (Auto-Request)

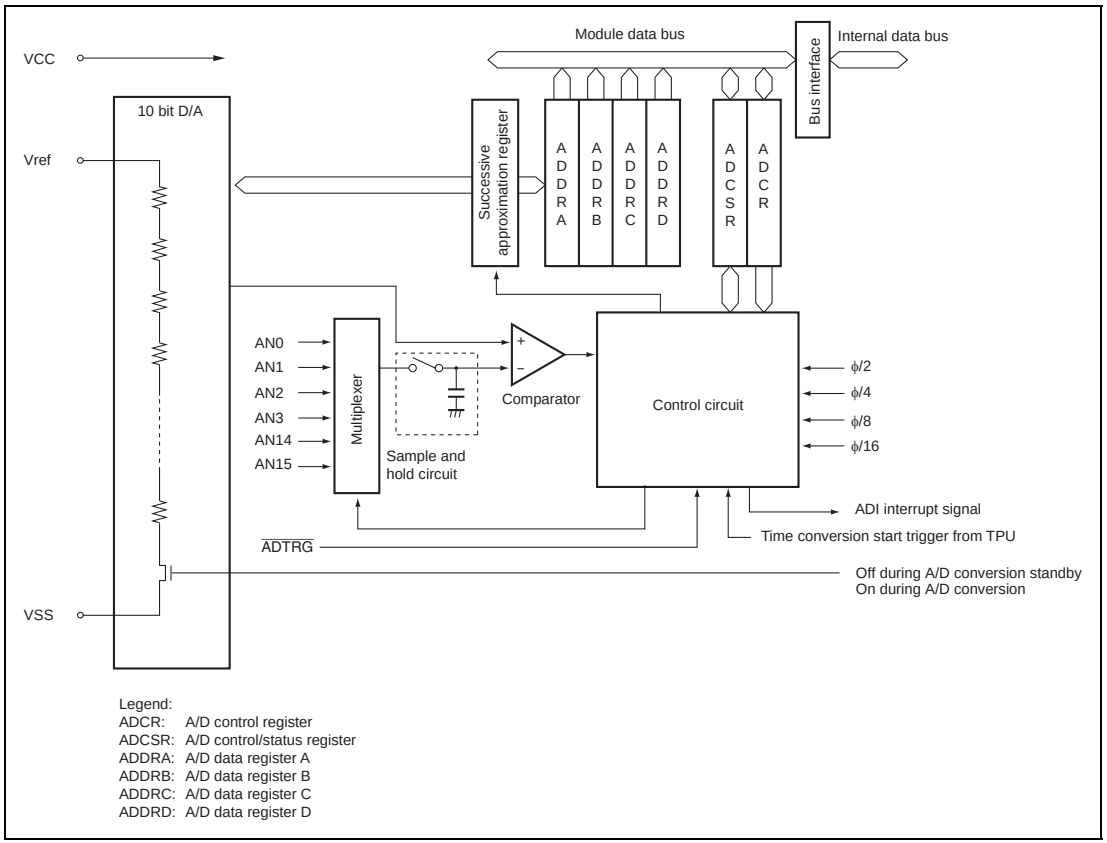


Figure 15.1 Block Diagram of A/D Converter

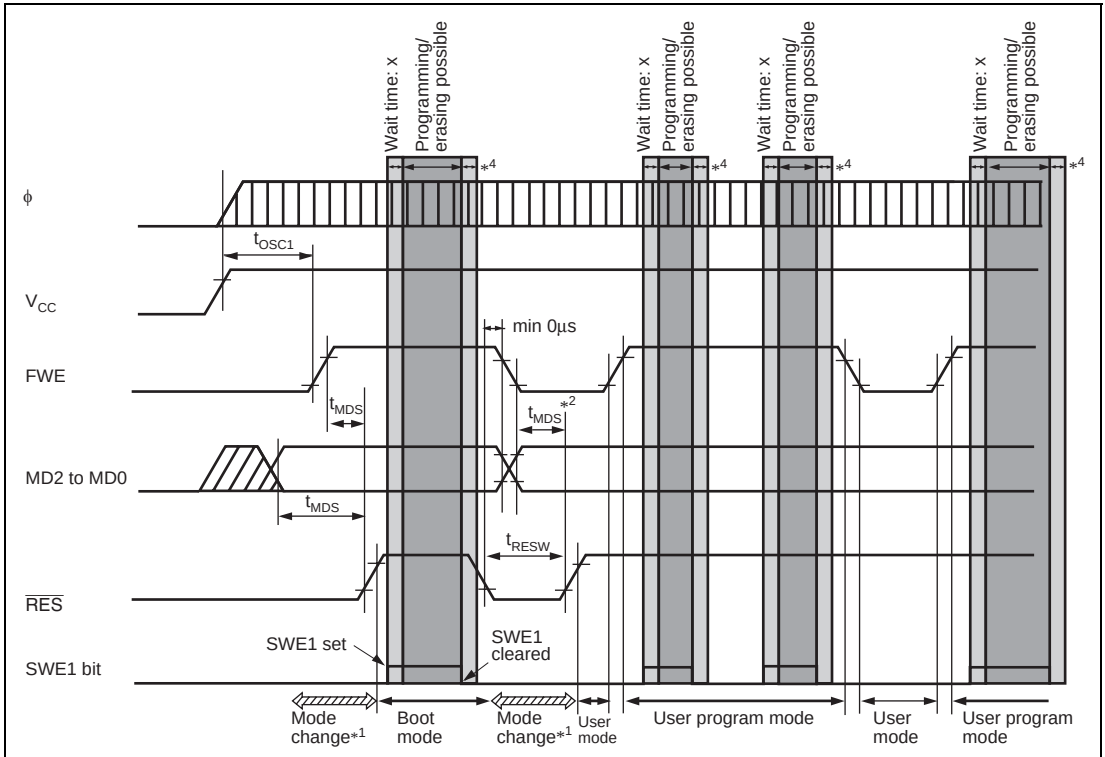
17.6.2 USB Boot Mode (HD64F2218U, HD64F2212U, and HD64F2211U)

- Features
  - Selection of bus-powered mode or self-powered mode
  - Supports the USB operating clock generation by 16 MHz system clock with PLL3 multiplication (FWE = 1, MD2 to MD0 = 011) or 24 MHz system clock with PLL2 multiplication (FWE = 1, MD2 to MD0 = 010)
  - D+ pull up control connection supported for P36 pin only
  - See table 17.6 for enumeration information

Table 17.6 Enumeration Information

USB Standard	Ver.1.1	
Transfer modes	Control (in, out), Bulk (in, out)	
Maximum power	Self power mode ($\overline{\text{UBPM}}$ pin = 1)	100 mA
	Bus power mode ($\overline{\text{UBPM}}$ pin = 0)	500 mA
Endpoint configuration	EP0 Control (in, out) 64 bytes Configuration 1 └ Interface Number 0 └ Alternate Setting 0 ├ EP1 Bulk (in) 64 bytes └ EP2 Bulk (out) 64 bytes	

- Notes on USB Boot Mode Execution
  - Specify 16 MHz or 24 MHz system clock and the FWE and MD2 to MD0 pins correctly.
  - Use the P36 pin for D+ pull-up control connection.
  - To ensure stable power supply during flash memory programming/erasing, do not use cable connection via a bus powered HUB.
  - Note in particular that, in the worst case, the LSI may be permanently damaged if the USB cable is detached during flash memory programming/erasing.
  - A transition is not made to power-down modes even if the USB bus enters suspend mode when in bus power mode.



Notes: 1. When entering boot mode or making a transition from boot mode to another mode, mode switching must be carried out by means of $\overline{RES}$ input
2. When making a transition from boot mode to another mode, a mode programming setup time t_{MDS} (min) of 200 ns is necessary with respect to $\overline{RES}$ clearance timing.
3. See section 22.7, Flash Memory Characteristics.
4. Wait time: θ.

Figure 17.18 Mode Transition Timing
(Example: Boot Mode → User Mode ↔ User Program Mode)

22.5 USB Characteristics

Table 22.8 lists the USB characteristics (USD+ and USD- pins) when the on-chip USB transceiver is used.

Table 22.8 USB Characteristics (USD+ and USD- pins) when On-Chip USB Transceiver Is Used

Conditions: V_{CC} = PLL V_{CC} = Dr V_{CC} = 3.0 V to 3.6 V, V_{SS} = $PLLV_{SS}$ = DrV_{SS} = 0 V, f = 16 MHz, 24 MHz, T_a = –20°C to +75°C (regular specifications), T_a = –40°C to +85°C (wide-range specifications)

Item		Symbol	Min.	Max.	Unit	Test Condition	
Input characteristics	Input high level voltage	V_{IH}	2.0	—	V		Figures 22.21, 22.22
	Input low level voltage	V_{IL}	—	0.8	V		
	Differential input sense	V_{DI}	0.2	—	V	| (D+)-(D-)|	
	Differential common mode range	V_{CM}	0.8	2.5	V		
Output characteristics	Output high level voltage	V_{OH}	2.8	—	V	I_{OH} = -200 μA	
	Output low level voltage	V_{OL}	—	0.3	V	I_{OL} = 2 mA	
	Crossover voltage	V_{CRS}	1.3	2.0	V		
	Rise time	t_R	4	20	ns		
	Fall time	t_F	4	20	ns		
	Rise time/fall time matching	t_{RFM}	90	111.11	%	(T_R/T_F)	
	Output resistance	Z_{DRV}	28	44	Ω	Including Rs = 24 Ω	