E·XFL

Renesas Electronics America Inc - R5F72145ADFP#V1 Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	SH2A-FPU
Core Size	32-Bit Single-Core
Speed	200MHz
Connectivity	CANbus, Ethernet, I²C, SCI, SPI, USB
Peripherals	DMA, PWM, WDT
Number of I/O	112
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	176-LQFP
Supplier Device Package	176-LFQFP (24x24)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f72145adfp-v1

Contents

Section 1 Overview....................1
1.1 Features....................1
1.2 Block Diagram....................10
1.3 Pin Assignment....................11
1.4 Pin Functions....................13

Section 2 CPU....................23
2.1 Data Formats....................23
2.2 Register Descriptions....................24
2.2.1 General Registers....................24
2.2.2 Control Registers....................25
2.2.3 System Registers....................27
2.2.4 Floating-Point Registers....................28
2.2.5 Floating-Point System Registers....................29
2.2.6 Register Bank....................32
2.2.7 Initial Values of Registers....................32
2.3 Data Formats....................33
2.3.1 Data Format in Registers....................33
2.3.2 Data Formats in Memory....................33
2.3.3 Immediate Data Format....................34
2.4 Instruction Features....................35
2.4.1 RISC-Type Instruction Set....................35
2.4.2 Addressing Modes....................39
2.4.3 Instruction Format....................44
2.5 Instruction Set....................48
2.5.1 Instruction Set by Classification....................48
2.5.2 Data Transfer Instructions....................55
2.5.3 Arithmetic Operation Instructions....................59
2.5.4 Logic Operation Instructions....................62
2.5.5 Shift Instructions....................63
2.5.6 Branch Instructions....................64
2.5.7 System Control Instructions....................66
2.5.8 Floating-Point Operation Instructions....................68
2.5.9 FPU-Related CPU Instructions....................70
2.5.10 Bit Manipulation Instructions....................70
2.6 Processing States....................72

Items	Specification
User break controller (UBC)	• Four break channels • Addresses, data values, type of access, and data size can all be set as break conditions
User debugging interface (H-UDI)	• E10A emulator support • JTAG-standard pin assignment • Boundary scan test port conforming to IEEE 1149.1 • Realtime branch trace
Advanced user debugger (AUD)	• Six input/output pins • Branch source address/destination address trace • Window data trace • Full trace All trace data can be output by interrupting CPU operation • Realtime trace Trace data can be output within the range where CPU operation is not interrupted
On-chip ROM	• 1 Mbyte, 768 Kbytes, 512 Kbytes
On-chip RAM	• Eight pages, six pages, four pages • 128 Kbytes, 96 Kbytes, 64 Kbytes
Data flash (FLD)	• 32 Kbytes • Programmed in 8-byte units
Power supply voltage	• $V_{CC}Q$: 3.0 to 3.6 V, AV_{CC}: 4.5 to 5.5 V
Packages	• PLQP0176KB-A (0.5-mm pitch) • PLQP0176LB-A (0.4-mm pitch) • PLBG0176GA-A (0.8-mm pitch)

7.3.8 Break Address Mask Register_2 (BAMR_2)

BAMR_2 is a 32-bit readable/writable register. BAMR_2 specifies bits masked in the break address bits specified by BAR_2. BAMR_2 is initialized to H'00000000 by a power-on reset, but retains its previous value by a manual reset or in software standby mode or sleep mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAM2_31	BAM2_30	BAM2_29	BAM2_28	BAM2_27	BAM2_26	BAM2_25	BAM2_24	BAM2_23	BAM2_22	BAM2_21	BAM2_20	BAM2_19	BAM2_18	BAM2_17	BAM2_16
Initial value:	0	0	0	0	0	0	0	0	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	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAM2_15	BAM2_14	BAM2_13	BAM2_12	BAM2_11	BAM2_10	BAM2_9	BAM2_8	BAM2_7	BAM2_6	BAM2_5	BAM2_4	BAM2_3	BAM2_2	BAM2_1	BAM2_0
Initial value:	0	0	0	0	0	0	0	0	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	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAM2_31 to BAM2_0	All 0	R/W	Break Address Mask 2 Specify bits masked in the channel-2 break address bits specified by BAR_2 (BA2_31 to BA2_0). 0: Break address bit BA2_n is included in the break condition 1: Break address bit BA2_n is masked and not included in the break condition

Note: n = 31 to 0

9.4.3 CSn Space Wait Control Register (CSnWCR) (n = 0 to 7)

CSnWCR specifies various wait cycles for memory access. The bit configuration of this register varies as shown below according to the memory type (TYPE2 to TYPE0) specified by the CSn space bus control register (CSnBCR). Specify CSnWCR before accessing the target area. Specify CSnBCR first, then specify CSnWCR.

CSnWCR is initialized to H'00000500 by a power-on reset and retains the value by a manual reset and in software standby mode.

(1) Normal Space, SRAM with Byte Selection, MPX-I/O

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	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	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]				WM	-	-	-	-	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	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	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—*	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

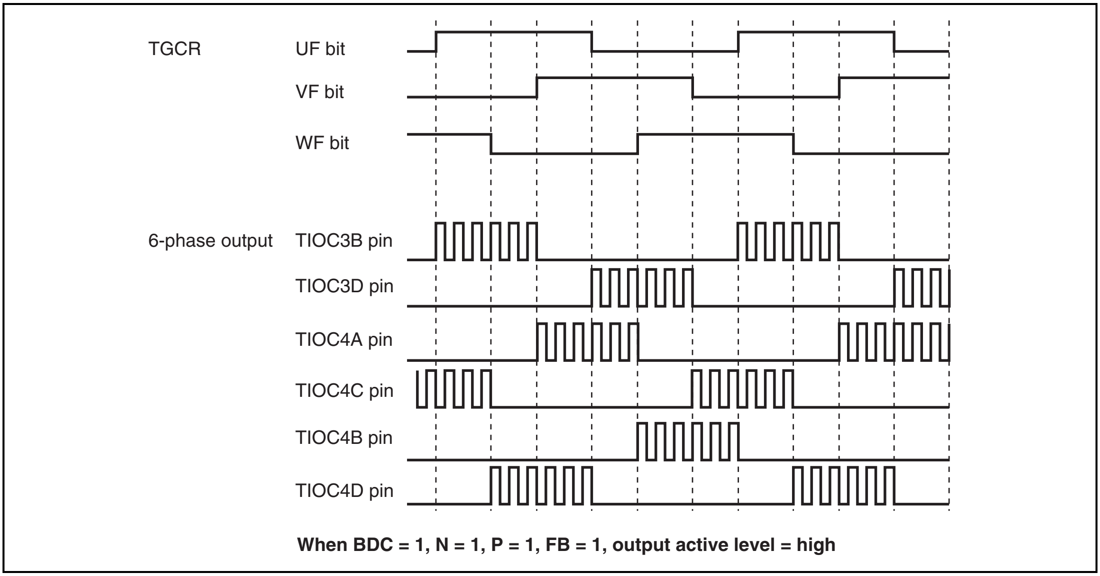


Figure 11.72 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)

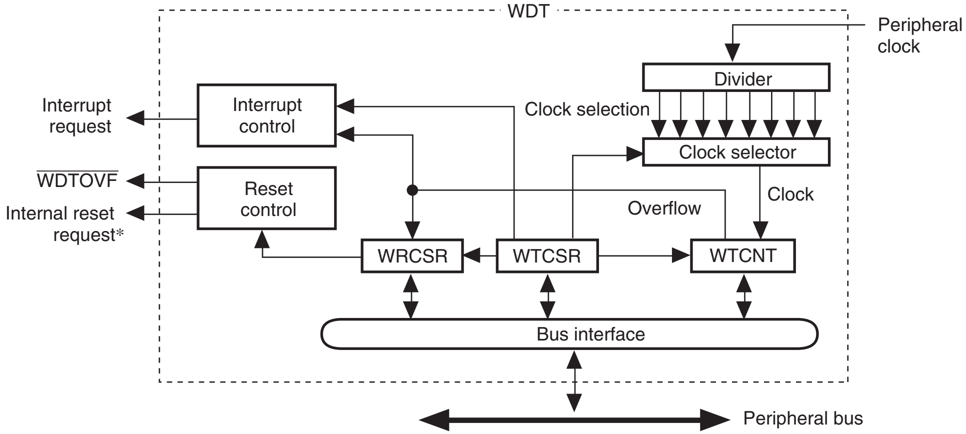
(r) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using a TGRA_3 compare-match, TCNT_4 underflow (trough), or compare-match on a channel other than channels 3 and 4.

When start requests using a TGRA_3 compare-match are specified, A/D conversion can be started at the crest of the TCNT_3 count.

A/D converter start requests can be set by setting the TTGE bit to 1 in the timer interrupt enable register (TIER). To issue an A/D converter start request at a TCNT_4 underflow (trough), set the TTGE2 bit in TIER_4 to 1.

Figure 15.1 shows a block diagram of the WDT.



[Legend]
WTCSR: Watchdog timer control/status register
WTCNT: Watchdog timer counter
WRCSR: Watchdog reset control/status register

Note: * The internal reset signal can be generated by making a register setting.

Figure 15.1 Block Diagram of WDT

15.2 Input/Output Pin

Table 15.1 shows the pin configuration of the WDT.

Table 15.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Watchdog timer overflow	$\overline{\text{WDTOVF}}$	Output	Outputs the counter overflow signal in watchdog timer mode

- Port D Control Register H1 (PDCRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD19MD[2:0]			-	PD18MD[2:0]			-	PD17MD[2:0]			-	PD16MD[2:0]		
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 during the on-chip ROM disabled 32-bit external extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	PD19MD[2:0]	000*	R/W	PD19 Mode Select the function of the PD19/D19/IRQ3/AUDATA3/LNKSTA pin. 000: PD19 I/O (port) 001: D19 I/O (port) 010: Setting prohibited 011: IRQ3 input (INTC) 100: AUDATA3 output (AUD) 101: Setting prohibited 110: Setting prohibited 111: LNKSTA input (Ether)
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	PD18MD[2:0]	000*	R/W	PD18 Mode Select the function of the PD18/D18/IRQ2/AUDATA2/MDIO pin. 000: PD18 I/O (port) 001: D18 I/O (BSC) 010: Setting prohibited 011: IRQ2 input (INTC) 100: AUDATA2 output (AUD) 101: Setting prohibited 110: Setting prohibited 111: MDIO I/O (Ether)

22.3 Usage Notes

1. In this LSI, the same function is available as a multiplexed function on multiple pins. This approach is intended to increase the number of selectable pin functions and to allow the easier design of boards. Note the following points when two or more pins are specified for one function.

- When the pin function is input

  Signals input to several pins are formed as one signal through OR or AND logic and the signal is transmitted into the LSI. Therefore, a signal that differs from the input signals may be transmitted to the LSI depending on the input signals in other pins that have the same functions. Table 22.10 shows the transmit forms of input functions allocated to several pins. When using one of the functions shown below in multiple pins, use it with care of signal polarity considering the transmit forms.

Table 22.10 Transmission Format of Input Function Allocated on Multiple Pins

OR Type	AND Type
TCLKA, TCLKB, TCLKC, TCLKD (MTU2)	IRQ0 to IRQ7 (INTC)
TIOC0A, TIOC0B, TIOC0C, TIOC0D (MTU2)	DREQ0, DREQ1 (DMAC)
TIOC1A, TIOC1B, TIOC2A (MTU2)	$\overline{\mathrm{ADTRG}}$ (ADC)
TIC5U, TIC5V, TIC5W (MTU2)	$\overline{\mathrm{WAIT}}$, $\overline{\mathrm{BREQ}}$ (BSC)
TIOC3AS, TIOC3BS, TIOC3CS, TIOC3DS (MTU2S)	CRx0 (RCAN-ET)
TIOC4AS, TIOC4BS, TIOC4CS, TIOC4DS (MTU2S)	
SCK0 to SCK3, RXD0 to RXD3 (SCI, SCIF)	
$\overline{\mathrm{POE0}}$, $\overline{\mathrm{POE4}}$, $\overline{\mathrm{POE8}}$ (POE2)	
SSLO, MISO, MOSI, RSPCK (RSPI)	
LNKSTA, COL, CRS, MDIO, RX_CLK (Ether)	
MII_RXD0 to MII_RXD3, RX_ER, RX_DV, TX_CLK (Ether)	

OR Type: Signals input to several pins are formed as one signal through OR logic and the signal is transmitted into the LSI.

AND Type: Signals input to several pins are formed as one signal through AND logic and the signal is transmitted into the LSI.

- When the pin function is output

  Each selected pin can output the same function.

24.3.7 USB Interrupt Enable Register 1 (USBIER1)

USBIER1 enables the interrupt requests indicated in the USB interrupt flag register 1 (USBIFR1). When an interrupt flag is set while the corresponding bit in USBIER1 is set to 1, an interrupt request is sent to the CPU. The interrupt vector number is determined by the content of the USB interrupt select register 1 (USBISR1).

Bit:	7	6	5	4	3	2	1	0
	-	-	-	SOFE	SETUP TSE	EP0o TSE	EP0i TRE	EP0i TSE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SOFE	0	R/W	SOF packet detection
3	SETUPTSE	0	R/W	Setup command receive complete
2	EP0oTSE	0	R/W	EP0o receive complete
1	EP0iTRE	0	R/W	EP0i transfer request
0	EP0iTSE	0	R/W	EP0i transmit complete

24.3.16 USBEP0i Data Register (USBEPDR0i)

USBEPDR0i is a 16-byte transmit FIFO buffer for endpoint 0, holding one packet of transmit data for control IN. Transmit data is fixed by writing one packet of data and setting the EP0iPKTE bit in the USB trigger register 0 (USBTRG0). When an ACK handshake is returned from the host after the data has been transmitted, the EP0iTS bit in the USB interrupt flag register 1 (USBIFR1) is set to 1. USBEPDR0i can be initialized by the EP0iCLR bit in the USB FIFO clear register 0 (USBFCLR0). The read value is undefined.

Bit:	7	6	5	4	3	2	1	0
	D7	D6	D5	D4	D3	D2	D1	D0
Initial value:	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	D7 to D0	—	W	Data register for control IN transfer

24.3.17 USBEP0o Data Register (USBEPDR0o)

USBEPDR0o is a 16-byte receive FIFO buffer for endpoint 0 to store endpoint 0 receive data other than setup commands. When data is received normally, the EP0oTS bit in the USB interrupt flag register 1 (USBIFR1) is set, and the number of receive bytes is indicated in the USBEP0o receive data size register (USBEPSZ0o). After the data has been read, setting the EP0oRDFN bit in the USB trigger register 0 (USBTRG0) enables the next packet to be received. USBEPDR0o can be initialized by the EP0oCLR bit in the USB FIFO clear register 0 (USBFCLR0).

Bit:	7	6	5	4	3	2	1	0
	D7	D6	D5	D4	D3	D2	D1	D0
Initial value:	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	D7 to D0	—	R	Data register for control OUT transfer

24.7 Stall Operations

24.7.1 Overview

This section describes stall operations in the USB function module. The USB function module stall function is used in the following cases:

- When the application forcibly stalls an endpoint for some reason
- When a stall is performed automatically within the USB function module due to a USB specification violation

The USB function module has internal status bits that hold the status (stall or non-stall) of each endpoint. When a transaction is sent from the host, the module references these internal status bits and determines whether to return a stall to the host. These bits cannot be cleared by the application. They must be cleared with a Clear Feature command from the host. The internal status bit for EP0 is automatically cleared only when the setup command is received.

24.7.2 Forcible Stall by Application

The application uses the USBEPSTL register to issue a stall request for the USB function module. When the application wishes to stall a specific endpoint, it sets the corresponding bit in USBEPSTL (1-1 in figure 24.14). The internal status bits remain unchanged at this time. When a transaction is sent from the host to the endpoint for which the USBEPSTL bit is set, the USB function module references the internal status bit, and if this is not set, references the corresponding bit in USBEPSTL (1-2 in figure 24.14). If the corresponding bit in USBEPSTL is set, the USB function module sets the internal status bit and returns a stall handshake to the host (1-3 in figure 24.14). If the corresponding bit in USBEPSTL is not set, the internal status bit remains unchanged and the transaction is accepted.

Once an internal status bit is set, it remains set until it is cleared by a Clear Feature command from the host, without regard to the USBEPSTL register. Even after a bit is cleared by the Clear Feature command (3-1 in figure 24.14), the USB function module continues to return a stall handshake while the bit in USBEPSTL is set, since the internal status bit is set each time a transaction is executed for the corresponding endpoint (1-2 in figure 24.14). To clear a stall, therefore, the corresponding bit in USBEPSTL must be cleared by the application, and the internal status bit must be cleared with a Clear Feature command (2-1, 2-2, and 2-3 in figure 24.14).

25.3.2 EtherC Status Register (ECSR)

ECSR is a 32-bit readable/writable register that indicates the status in the EtherC. This status can be notified to the CPU by interrupts. When 1 is written to the PSRTO, LCHNG, MPD, and ICD bits, the corresponding flags can be cleared to 0. Writing 0 does not affect any flags. For bits that generate interrupts, the interrupt can be enabled or disabled by the corresponding bit in ECSIPR.

The interrupts generated due to this status register are reflected in the ECI bit in EESR of the E-DMAC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	BFR	PSRTO	–	LCHNG	MPD	ICD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	BFR	0	R/W	Continuous Broadcast Frame Reception Interrupt (Interrupt Source) Indicates that broadcast frames have been continuously received.
4	PSRTO	0	R/W	PAUSE Frame Retransmit Retry Over Indicates whether the retransmit count for retransmitting a PAUSE frame when flow control is enabled has exceeded the retransmit upper-limit value set in the automatic PAUSE frame retransmit count register (TPAUSER). 0: PAUSE frame retransmit count has not exceeded the upper limit 1: PAUSE frame retransmit count has exceeded the upper limit

Bit	Bit Name	Initial Value	R/W	Description
0	RNR	0	R/W	Receive Request Bit Reset 0: Allows the hardware to reset the receive request (RR) bit in EDRRR automatically upon completion of reception of one frame. This control is possible for each frame. To receive the subsequent receive frame, the RR bit in EDRRR needs to be set again. 1: Allows the higher-level software to control the receive request (RR) bit in EDRRR. Once the RR bit in EDRRR is set to 1, the hardware continues to fetch the receive descriptor and receive frames autonomously until the RR bit in EDRRR is cleared to 0. In other words, continuous reception of multiple frames are possible. Setting this bit to 1 is recommended for continuous reception. However, when a receive descriptor empty is detected, the hardware clears the RR bit in EDRRR automatically.

- Programming/erasing unit

  The data MAT is programmed in 8-byte or 128-byte units and erased in block units (8 Kbytes) in user mode, user program mode, and user boot mode. In boot mode, the data MAT is programmed in 256-Kbyte units and erased in block units (8 Kbytes). The product information MAT is read-only memory and cannot be programmed or erased.

  Figure 28.3 shows the block configuration of the data MAT of this LSI. The data MAT is divided into four 8-Kbyte blocks (DB00 to DB03).

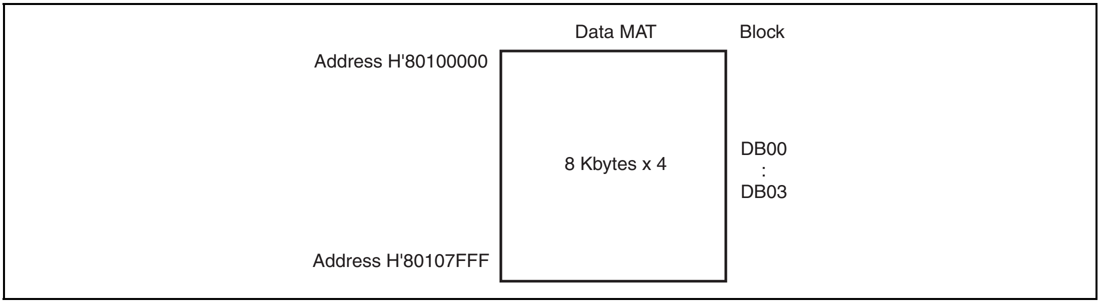


Figure 28.3 Block Configuration of Data MAT

- Blank check function

  If data is read from erased FLD by the CPU, undefined values are read. Using blank check command of the FCU allows checking of whether the FLD is erased (in a blank state). Either an 8 Kbytes (1 erasure block) or 8 bytes of area can be checked by a single execution of the blank check command.

  Blank checking proceeds for areas where erasure has been completed normally to confirm that the data have actually been erased. When erasure or programming in progress is stopped (e.g. by input of the reset signal or shutting down the power), blank checking cannot be used to check whether the data have actually been erased or written.

- Four types of on-board programming modes

  — Boot mode

    The data MAT can be programmed using the SCI. The bit rate for SCI communications between the host and the LSI can be automatically adjusted.

  — User mode/user program mode

    The data MAT can be programmed with a desired interface. The user mode includes the MCU extended mode and MCU single-chip mode (modes 2 and 3) in which the on-chip ROM is enabled.

(1) ROM P/E Mode

The FCU can accept ROM programming and erasing commands in this mode. The FLD cannot be read. The FCU enters this mode when the FENTRYD bit is set to 0 and the FENTRY0 bit is set to 1 in FENTRYR. For details of this mode, refer to section 27.6.2, Conditions for FCU Command Acceptance.

(2) ROM/FLD Read Mode

The FLD can be read through the HPB, and the ROM can be read through the ROM cache at a high speed. The FCU does not accept commands. The FCU enters this mode when the FENTRY0 bit is set to 0 and the FENTRYD bit in FENTRYR is set to 0.

(3) FLD P/E Mode

- FLD P/E normal mode

  The FCU enters this mode when the FENTRYD bit is set to 1 and the FENTRY0 bit is set to 0 in ROM/FLD read mode or ROM P/E mode, or when a normal mode transition command is accepted in FLD P/E mode. Table 28.8 shows the commands that can be accepted in this mode. If the FLD area is read through the P bus, an FLD access error occurs and the FCU enters the command-locked state.
- FLD status read mode

  The FCU enters this mode when the FCU accepts a command that is neither the normal mode transition command nor the lock bit read mode transition command in FLD P/E mode. The FLD status read mode includes the state in which the FRDY bit in FSTATR0 is 0 and the command-locked state after an error has occurred. Table 28.8 shows the commands that can be accepted in this mode. If the FLD area is read through the P bus, the FSTATR0 value is read.
- FLD lock bit read mode

  The FCU enters this mode when the FCU accepts a lock bit read mode transition command in FLD P/E mode. Table 28.8 shows the commands that can be accepted in this mode. Since the FLD has no lock bits, reading an FLD area via the P-bus results in an undefined value. However, no access violation occurs in this case. High-speed read operation is available for ROM.

30.3.2 Standby Control Register 2 (STBCR2)

STBCR2 is an 8-bit readable/writable register that controls the operation of modules in power-down modes. STBCR2 is initialized to H'00 by a power-on reset but retains its previous value by a manual reset or in software standby mode. Only byte access is possible.

Bit:	7	6	5	4	3	2	1	0
	MSTP10	MSTP9	MSTP8	-	-	-	MSTP4	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP10	0	R/W	Module Stop 10 When the MSTP10 bit is set to 1, the supply of the clock to the H-UDI is halted. 0: H-UDI runs. 1: Clock supply to H-UDI halted.
6	MSTP9	0	R/W	Module Stop 9 When the MSTP9 bit is set to 1, the supply of the clock to the UBC is halted. 0: UBC runs. 1: Clock supply to UBC halted.
5	MSTP8	0	R/W	Module Stop 8 When the MSTP8 bit is set to 1, the supply of the clock to the DMAC is halted. 0: DMAC runs. 1: Clock supply to DMAC halted.
4 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	MSTP4	0	R/W	Module Stop 4 When the MSTP4 bit is set to 1, the supply of the clock to the DTC is halted. 0: DTC runs. 1: Clock supply to DTC halted.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Module Name	Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
EtherC	PSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	LMON
	TROCR	TROC31	TROC30	TROC29	TROC28	TROC27	TROC26	TROC25	TROC24
		TROC23	TROC22	TROC21	TROC20	TROC19	TROC18	TROC17	TROC16
		TROC15	TROC14	TROC13	TROC12	TROC11	TROC10	TROC9	TROC8
		TROC7	TROC6	TROC5	TROC4	TROC3	TROC2	TROC1	TROC0
	CDCR	COSDC31	COSDC30	COSDC29	COSDC28	COSDC27	COSDC26	COSDC25	COSDC24
		COSDC23	COSDC22	COSDC21	COSDC20	COSDC19	COSDC18	COSDC17	COSDC16
		COSDC15	COSDC14	COSDC13	COSDC12	COSDC11	COSDC10	COSDC9	COSDC8
		COSDC7	COSDC6	COSDC5	COSDC4	COSDC3	COSDC2	COSDC1	COSDC0
	LCCR	LCC31	LCC30	LCC29	LCC28	LCC27	LCC26	LCC25	LCC24
		LCC23	LCC22	LCC21	LCC20	LCC19	LCC18	LCC17	LCC16
		LCC15	LCC14	LCC13	LCC12	LCC11	LCC10	LCC9	LCC8
		LCC7	LCC6	LCC5	LCC4	LCC3	LCC2	LCC1	LCC0
	CNDCR	CNDC31	CNDC30	CNDC29	CNDC28	CNDC27	CNDC26	CNDC25	CNDC24
		CNDC23	CNDC22	CNDC21	CNDC20	CNDC19	CNDC18	CNDC17	CNDC16
		CNDC15	CNDC14	CNDC13	CNDC12	CNDC11	CNDC10	CNDC9	CNDC8
		CNDC7	CNDC6	CNDC5	CNDC4	CNDC3	CNDC2	CNDC1	CNDC0
	CEFCR	CEFC31	CEFC30	CEFC29	CEFC28	CEFC27	CEFC26	CEFC25	CEFC24
		CEFC23	CEFC22	CEFC21	CEFC20	CEFC19	CEFC18	CEFC17	CEFC16
		CEFC15	CEFC14	CEFC13	CEFC12	CEFC11	CEFC10	CEFC9	CEFC8
		CEFC7	CEFC6	CEFC5	CEFC4	CEFC3	CEFC2	CEFC1	CEFC0
	FRECR	FREC31	FREC30	FREC29	FREC28	FREC27	FREC26	FREC25	FREC24
		FREC23	FREC22	FREC21	FREC20	FREC19	FREC18	FREC17	FREC16
		FREC15	FREC14	FREC13	FREC12	FREC11	FREC10	FREC9	FREC8
		FREC7	FREC6	FREC5	FREC4	FREC3	FREC2	FREC1	FREC0
	TSFRCR	TSFC31	TSFC30	TSFC29	TSFC28	TSFC27	TSFC26	TSFC25	TSFC24
		TSFC23	TSFC22	TSFC21	TSFC20	TSFC19	TSFC18	TSFC17	TSFC16
		TSFC15	TSFC14	TSFC13	TSFC12	TSFC11	TSFC10	TSFC9	TSFC8
		TSFC7	TSFC6	TSFC5	TSFC4	TSFC3	TSFC2	TSFC1	TSFC0
	TLFRCR	TLFC31	TLFC30	TLFC29	TLFC28	TLFC27	TLFC26	TLFC25	TLFC24
		TLFC23	TLFC22	TLFC21	TLFC20	TLFC19	TLFC18	TLFC17	TLFC16
		TLFC15	TLFC14	TLFC13	TLFC12	TLFC11	TLFC10	TLFC9	TLFC8
		TLFC7	TLFC6	TLFC5	TLFC4	TLFC3	TLFC2	TLFC1	TLFC0

Module Name	Register	Power-on Reset	Manual Reset	Software Standby	Module Standby	Sleep
ROM/FLD	FCMDR	Initialized	Retained	Retained	Retained	Retained
	FCPSR	Initialized	Retained	Retained	Retained	Retained
	EEPBCCNT	Initialized	Retained	Retained	Retained	Retained
	FPESTAT	Initialized	Retained	Retained	Retained	Retained
	EEPBCSTAT	Initialized	Retained	Retained	Retained	Retained
	EEPRE0	Initialized	Retained	Retained	Retained	Retained
	EEPWE0	Initialized	Retained	Retained	Retained	Retained
	RCCR	Initialized	Retained	Retained	Retained	Retained
	PCKAR	Initialized	Retained	Retained	Retained	Retained
Power-down mode	STBCR	Initialized	Retained	Retained	—	Retained
	STBCR2	Initialized	Retained	Retained	—	Retained
	SYSCR1	Initialized	Retained	Retained	—	Retained
	SYSCR2	Initialized	Retained	Retained	—	Retained
	STBCR3	Initialized	Retained	Retained	—	Retained
	STBCR4	Initialized	Retained	Retained	—	Retained
	STBCR5	Initialized	Retained	Retained	—	Retained
	STBCR6	Initialized	Retained	Retained	—	Retained
H-UDI*3	SDIR	Retained	Retained	Retained	Retained	Retained

Notes: 1. Retains the previous value after an internal power-on reset by means of the WDT.
2. Bits BN[3:0] are initialized.
3. Initialized by TRST assertion or in the Test-Logic-Reset state of the TAP controller.
4. Initialized after an internal manual reset by means of the WDT.
5. Some bits are not initialized.

Notes: 1. Output pins become high-impedance when the HIZ bit in standby control register 3 (STBCR3) is set to 1.

2. Becomes output when the HIZCNT bit in the common control register (CMNCR) is set to 1.

3. Becomes output when the HIZMEM bit in the common control register (CMNCR) is set to 1.

4. Becomes output when the HIZCKIO bit in the common control register (CMNCR) is set to 1.

5. Becomes high-impedance when the MZIZDH bit in the high-current port control register (HCPCR) is set to 0.

6. Becomes high-impedance when the MZIZDL bit in the high-current port control register (HCPCR) is set to 0.

7. Becomes high-impedance when the MZIZEH bit in the high-current port control register (HCPCR) is set to 0.

8. Becomes high-impedance when the MZIZEL bit in the high-current port control register (HCPCR) is set to 0.

9. Becomes input during a power-on reset. Pull-up to prevent erroneous operation. Pull-down with a resistance of at least 1 MW as required.

10. Pulled-up inside the LSI when there is no input.

TPAUSER 1391
TRIMD 1445
TROCR.. 1378
TRSCER... 1427
TRWER 518
TSFRCR 1384
TSR.. 494
TSTR 511
TSYCRS 504
TSYR.. 513
TWCR.. 537
TXACK0 1100
TXCR0 1099
TXPR1, TXPR0................................ 1097
UMSR0.. 1105
USBCTLR 1306
USBCVR 1305
USBDASTS0................................... 1279
USBDASTS1................................... 1280
USBDASTS2................................... 1281
USBDASTS3................................... 1282
USBDMAR 1302
USBEPDR0i.................................... 1271
USBEPDR0o 1271
USBEPDR0s.................................... 1272
USBEPDR1 1273
USBEPDR2 1273
USBEPDR3 1274
USBEPDR4 1274
USBEPDR5 1275
USBEPDR6 1275
USBEPDR7 1276
USBEPDR8 1276
USBEPDR9 1277
USBEPIR... 1307
USBEPSTL0.................................... 1291
USBEPSTL1.................................... 1292
USBEPSTL2.................................... 1293
USBEPSTL3.................................... 1294
USBEPSZ0o 1277
USBEPSZ1 1278
USBEPSZ4 1278
USBEPSZ7 1279
USBFCLR0....................................... 1287
USBFCLR1....................................... 1288
USBFCLR2....................................... 1289
USBFCLR3....................................... 1290
USBIER0 1261
USBIER1 1262
USBIER2 1263
USBIER3 1264
USBIER4 1265
USBIFR0 1254
USBIFR1 1255
USBIFR2 1257
USBIFR3 1258
USBIFR4 1260
USBISR0 1266
USBISR1 1267
USBISR2 1268
USBISR3 1269
USBISR4 1270
USBSTLSR1.................................... 1296
USBSTLSR2.................................... 1298
USBSTLSR3.................................... 1300
USBTRG0.. 1283
USBTRG1.. 1284
USBTRG2.. 1285
USBTRG3.. 1286
USBTRNTREG0 1312
USBTRNTREG1 1314
USDTENDRR..................................... 141
WRCSR 754
WTCNT 751
WTCSR.. 752
Relationship between RSPI modes and
SPCR and description of each mode....... 931
Renesas serial peripheral interface (RSPI)
.... 897
Repeat transfer mode 231