MB86R03 'Jade-L'

LSI Product Specifications

November, 2009
The 1.2 edition

FUJITSU

Preface

Objectives and Intended Reader

Thank you very much for your continued special support for Fujitsu Microelectronics semiconductor products.
MB86R03 is LSI product for the graphics applications.
This manual describes functions and operations of MB86R03 for engineers who design products using MB86R03. Read through this manual before use.

Trademarks

ARM is a registered trademark of ARM Limited in UK, USA and Taiwan.
ARM is a trademark of ARM Limited in Japan and Korea.
ARM Powered logo is a registered trademark of ARM Limited in Japan, UK, USA, and Taiwan.
ARM Powered logo is a trademark of ARM Limited in Korea.
ARM926EJ-S and ETM9 are trademarks of ARM Limited.

The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

Hardware Related Manuals

MB86R03 hardware related manuals are shown below. Refer them as the situation demands.

- MB86R03 'Jade-L' LSI product specifications graphics display controller (GDC)
- MB86R03 'Jade-L' LSI product specifications SD memory controller (Note)
- MB86R03 'Jade-L' Data sheet

Note) This specification document is for SD card licensee.

Revision History

Date	Ver.	Contents
2009/09/18	1.0	Newly issued
2009/10/07	1.1	Whole contents • Added model number (MB86R03) 1.6.23. Unused pin • Table 1-31 Revised description of P2, P5, R1, R2, R3, R5, T1, T2, and U3 Added description of *1) and *2)
2009/11/24	1.2	1.6.23. Unused pin • Revised table 1-31 [Pin No.] [JEDEC] [Pin name] [Pin name] 112 N2 VINFID0, GI1[3], MLB_CLK -> VINFID0, GI1[3] 202 M3 VINVSYNC0, GI1[5], MLB_DATA -> VINVSYNC0, GI1[5] 203 N3 VINHSYNC0, GI1[4], MLB_SIG -> VINHSYNC0, GI1[4] 3.3. Register map • Revised table 3-2 [Address] [Register name] [Explanation] [Register name] [Explanation] FFF4_2038$_H$ CMLB MediaLB setting register -> Reserved Access prohibited FFF4_2040$_H$ CUSB USB setting register -> Reserved Access prohibited

Contents

1. Outline 1-1

1.1. Feature 1-1
1.2. Block diagram 1-3
1.3. Function list 1-5
1.4. Package dimension 1-7
1.5. Pin assignment 1-8
1.6. Pin function 1-11
1.6.1. Pin Multiplex 1-11
1.6.2. Pin function 1-16
1.6.3. External bus interface related pin 1-17
1.6.4. SD memory controller related pin 1-17
1.6.5. External interrupt controller related pin 1-17
1.6.6. UART related pin 1-18
1.6.7. CAN related pin 1-18
1.6.8. I2S related pin 1-19
1.6.9. I^2C related pin 1-19
1.6.10. SPI related pin 1-19
1.6.11. PWM related pin 1-20
1.6.12. A/D converter related pin 1-20
1.6.13. DDR2 related pin 1-21
1.6.14. DISPLAY related pin 1-22
1.6.15. Video capture related pin 1-23
1.6.16. System related pin 1-23
1.6.17. JTAG related pin 1-24
1.6.18. ICE related pin 1-24
1.6.19. Multiplex setting related pin 1-24
1.6.20. ETM related pin 1-25
1.6.21. Power supply related pin 1-25
1.6.22. GPIO related pin 1-25
1.6.23. Unused pin 1-26
1.6.24. Unused pin in the duplex case with pin multiplex function 1-34

2. System configuration 2-1

3. Memory map 3-1

3.1. Memory map of LSI 3-1
3.2. Register access 3-3
3.3. Register map 3-4

4. CPU (ARM926EJ-S core part) 4-1

4.1. Outline 4-1
4.2. Feature 4-1
4.3. Block diagram 4-1
4.4. ARM926EJ-S and ETM setting 4-2

5. Clock reset generator (CRG) 5-1

5.1. Outline 5-1
5.2. Feature 5-1
5.3. Block diagram 5-2
5.4. Register 5-3
5.4.1. Register list 5-3
5.4.2. PLL control register (CRPR) 5-5
5.4.3. Watchdog timer control register (CRWR) 5-7
5.4.4. Reset/Standby control register (CRSR) 5-9
5.4.5. Clock divider control register A (CRDA) 5-11
5.4.6. Clock divider control register B (CRDB) 5-13
5.4.7. AHB (A) bus clock gate control register (CRHA) 5-14
5.4.8. APB (A) bus clock gate control register (CRPA) 5-15
5.4.9. APB (B) bus clock gate control register (CRPB) 5-16
5.4.10. AHB (B) bus clock gate control register (CRHB) 5-17
5.4.11. ARM core clock gate control register (CRAM) 5-18
5.5. Operation 5-19
5.5.1. Generation of reset 5-19
5.5.2. Clock generation 5-22

6. Remap boot controller (RBC) 6-1

6.1. Outline 6-1
6.2. Feature 6-1
6.3. Block diagram 6-1
6.4. Supply clock 6-2
6.5. Register 6-2
6.5.1. Register list 6-2
6.5.2. Remap control register (RBREMAP) 6-4
6.5.3. VINITHI control register A (RBVIHA) 6-5
6.5.4. INITRAM control register A (RBITRA) 6-6
6.6. Operation 6-7
6.6.1. RBC reset 6-7
6.6.2. Remap control 6-7
6.6.3. VINITHI control 6-7
6.6.4. INITRAM control 6-8

7. Interrupt controller (IRC) 7-1

7.1. Outline 7-1
7.2. Feature 7-1
7.3. Block diagram 7-2
7.4. Supply clock 7-3
7.5. Interrupt map 7-3
7.5.1. Exception vector to ARM926EJ-S core 7-3
7.5.2. Extension IRQ interrupt vector of IRC0/IRC1IRC0/IRC1 7-4
7.6. Register 7-6
7.6.1. Register list 7-6
7.6.2. IRQ flag register (IR0IRQF/ IR1IRQF) 7-10
7.6.3. IRQ mask register (IR0IRQM/IR1IRQM) 7-11

7.6.4. Interrupt level mask register (IR0ILM/IR1ILM) ... 7-12
7.6.5. ICR monitoring register (IR0ICRMN/IR1ICRMN) ... 7-13
7.6.6. Delay interrupt control register 0 (IR0DICR0) ... 7-14
7.6.7. Delay interrupt control register 1 (IR0DICR1) ... 7-15
7.6.8. Table base register (IR0TBR/IR1TBR) ... 7-16
7.6.9. Interrupt vector register (IR0VCT/IR1VCT) ... 7-17
7.6.10. Interrupt control register (IR0ICR31/IR1ICR31 – IR0ICR00/IR1ICR00) ... 7-18
7.7. Operation ... 7-19
7.7.1. Outline ... 7-19
7.7.2. Initialization ... 7-19
7.7.3. Multiple interrupt process ... 7-20
7.7.4. Example of IRQ interrupt handler ... 7-20
7.7.5. Resume from Stop and standby modes ... 7-22
7.7.6. Notice for using IRC ... 7-22

8. External bus interface ... 8-1

8.1. Outline ... 8-1
8.2. Spec limitation ... 8-1
8.3. Feature ... 8-3
8.4. Block diagram ... 8-3
8.5. Related pin ... 8-4
8.6. Supply clock ... 8-4
8.7. Register ... 8-5
8.7.1. SRAM/Flash mode register 0-7 (MCFMODE0-7) ... 8-5
8.7.2. SRAM/Flash timing register 0-7 (MCFTIM0-7) ... 8-7
8.7.3. SRAM/Flash area register 0-7 (MCFAREA0-7) ... 8-9
8.7.4. Memory controller error register (MCERR) ... 8-11
8.8. Connection example ... 8-12
8.9. Example of access waveform ... 8-14
8.10. Operation ... 8-18
8.10.1. External bus interface ... 8-18
8.10.2. Low-speed device interface function ... 8-18
8.10.3. Endian and byte lane to each access ... 8-19

9. DDR2 controller ... 9-1

9.1. Outline ... 9-1
9.2. Feature ... 9-1
9.3. Block diagram ... 9-2
9.4. Supply clock ... 9-3
9.5. Register ... 9-3
9.5.1. Register list ... 9-3
9.5.2. DRAM initialization control register (DRIC) ... 9-5
9.5.3. DRAM initialization command register [1] (DRIC1) ... 9-7
9.5.4. DRAM initialization command register [2] (DRIC2) ... 9-7
9.5.5. DRAM CTRL ADD register (DRCA) ... 9-8
9.5.6. DRAM control mode register (DRCM) ... 9-9
9.5.7. DRAM CTRL SET TIME1 Register (DRCST1) ... 9-10
9.5.8. DRAM CTRL SET TIME2 register (DRCST2) ... 9-12
9.5.9. DRAM CTRL REFRESH register (DRCR) ... 9-14
9.5.10. DRAM CTRL FIFO register (DRCF) ... 9-15

9.5.11. AXI setting register (DRASR) ... 9-16
9.5.12. DRAM IF MACRO SETTING DLL register (DRIMSD) ... 9-17
9.5.13. DRAM ODT SETTING register (DROS) ... 9-18
9.5.14. IO buffer setting ODT1 (DRIBSODT1) ... 9-19
9.5.15. IO buffer setting OCD (DRIBSOCD) ... 9-20
9.5.16. IO buffer setting OCD2 (DRIBSOCD2) ... 9-21
9.5.17. ODT auto bias adjust register (DROABA) ... 9-22
9.5.18. ODT bias select register (DROBS) ... 9-23
9.5.19. IO monitor register 1 (DRIMR1) ... 9-24
9.5.20. IO monitor register 2 (DRIMR2) ... 9-24
9.5.21. IO monitor register 3 (DRIMR3) ... 9-25
9.5.22. IO monitor register 4 (DRIMR4) ... 9-25
9.5.23. OCD impedance setting Rrgister1 (DROISR1) ... 9-26
9.5.24. OCD impedance setting register2 (DROISR2) ... 9-26
9.6. Operation ... 9-27
9.6.1. DRAM initialization sequence ... 9-27
9.6.2. DRAM initialization procedure ... 9-28
9.6.2.1. SDRAM initialization procedure ... 9-29
9.6.2.2. OCD adjustment procedure ... 9-31
9.6.2.3. ODT setting procedure ... 9-33

10. Built-in SRAM ... 10-1

10.1. Outline ... 10-1
10.2. Feature ... 10-1
10.3. Block diagram ... 10-1
10.4. Supply clock ... 10-1

11. DMA controller (DMAC) ... 11-1

11.1. Outline ... 11-1
11.2. Feature ... 11-1
11.3. Block diagram ... 11-2
11.4. Related pin ... 11-3
11.5. Supply clock ... 11-3
11.6. Register ... 11-4
11.6.1. Register list ... 11-4
11.6.2. DMA configuration register (DMACR) ... 11-6
11.6.3. DMA configuration A register (DMACAx) ... 11-8
11.6.4. DMA configuration B register (DMACBx) ... 11-11
11.6.5. DMAC source address register (DMACSAx) ... 11-14
11.6.6. DMAC destination address register (DMACDAx) ... 11-15
11.7. Operation ... 11-16
11.7.1. Transfer mode ... 11-16
11.7.1.1. Block transfer ... 11-16
11.7.1.2. Burst transfer ... 11-20
11.7.1.3. Demand transfer ... 11-24
11.7.2. Beat transfer ... 11-28
11.7.2.1. Normal and Single transfer ... 11-28
11.7.2.2. Increment and lap transfer ... 11-29
11.7.3. Channel priority control ... 11-30
11.7.3.1. Fixed priority ... 11-30

11.7.3.2. Rotate priority 11-31
11.7.4. Retry, split, and error 11-32
11.7.4.1. Retry and split 11-32
11.7.4.2. Error 11-33
11.8. Example of DMAC setting 11-34
11.8.1. DMA start in Single channel 11-34
11.8.2. DMA start in all channels (in demand transfer mode) 11-35

12. Timer (TIMER) 12-1

12.1. Outline 12-1
12.2. Feature 12-1
12.3. Supply clock 12-1
12.4. Specification 12-1

13. General-purpose input/output port (GPIO) 13-1

13.1. Outline 13-1
13.2. Feature 13-1
13.3. Block diagram 13-1
13.4. Supply clock 13-1
13.5. Register 13-2
13.5.1. Register list 13-2
13.5.2. Port data register 0-2 (GPDR0-2) 13-4
13.5.3. Data direction register 0-2 (GPDDR2-0) 13-6
13.6. Operation 13-8
13.6.1. Direction control 13-8
13.6.2. Data transfer 13-8

14. PWM 14-1

14.1. Outline 14-1
14.2. Feature 14-1
14.3. Block diagram 14-2
14.4. Related pin 14-2
14.5. Supply clock 14-2
14.6. Interrupt 14-2
14.7. Register 14-3
14.7.1. Register list 14-3
14.7.2. PWMx base clock register (PWMxBCR) 14-5
14.7.3. PWMx pulse width register (PWMxTPR) 14-6
14.7.4. PWMx phase register (PWMxPR) 14-7
14.7.5. PWMx duty register (PWMxDR) 14-8
14.7.6. PWMx status register (PWMxCR) 14-9
14.7.7. PWMx start register (PWMxSR) 14-10
14.7.8. PWMx current count register (PWMxCCR) 14-11
14.7.9. PWMx interrupt register (PWMxIR) 14-12
14.8. Example of setting register 14-13

15. A/D converter 15-1

15.1. Outline 15-1

15.2. Feature ... 15-1
15.3. Block diagram ... 15-2
15.4. Related pin ... 15-2
15.5. Supply clock ... 15-2
15.6. Output truth value list ... 15-3
15.7. Analog pin equivalent circuit ... 15-4
15.8. Register ... 15-5
15.8.1. Register list ... 15-5
15.8.2. ADCx data register (ADCxDATA) ... 15-7
15.8.3. ADCx power down control register (ADCxXPD) ... 15-7
15.8.4. ADCx clock selection register (ADCxCKSEL) ... 15-8
15.8.5. ADCx status register (ADCxSTATUS) ... 15-9
15.9. Basic operation flow ... 15-10

16. Graphics display controller (GDC) ... 16-1

17. Serial audio interface (I2S) ... 17-1

17.1. Outline ... 17-1
17.2. Feature ... 17-1
17.3. Block diagram ... 17-2
17.4. Related pin ... 17-3
17.5. Supply clock ... 17-3
17.6. Register ... 17-4
17.6.1. Register list ... 17-4
17.6.2. I2SxRXFDAT register ... 17-6
17.6.3. I2SxTXFDAT register ... 17-7
17.6.4. I2SxCNTREG register ... 17-8
17.6.5. I2SxMCR0REG register ... 17-11
17.6.6. I2SxMCR1REG register ... 17-12
17.6.7. I2SxMCR2REG register ... 17-13
17.6.8. I2SxOPRREG register ... 17-14
17.6.9. I2SxSRST register ... 17-15
17.6.10. I2SxINTCNT register ... 17-16
17.6.11. I2SxSTATUS register ... 17-19
17.6.12. I2SxDMAACT register ... 17-21
17.7. Operation ... 17-22
17.7.1. Outline ... 17-22
17.7.2. Transfer start, stop, and malfunction ... 17-23
17.7.3. Frame construction ... 17-28
17.7.3.1. 1 sub frame construction ... 17-28
17.7.3.2. 2 sub frame construction ... 17-29
17.7.3.3. Bit alignment ... 17-30
17.7.4. FIFO construction and description ... 17-32

18. UART interface ... 18-1

18.1. Outline ... 18-1
18.2. Feature ... 18-1
18.3. Block diagram ... 18-1
18.4. Related pin ... 18-2

18.5. Supply clock 18-2
18.6. Register 18-3
18.6.1. Register list 18-3
18.6.2. Reception FIFO register (URTxRFR) 18-6
18.6.3. Transmission FIFO register (URTxTFR) 18-6
18.6.4. Interrupt enable register (URTxIER) 18-7
18.6.5. Interrupt ID register (URTxIIR) 18-8
18.6.6. FIFO control register (URTxFCR) 18-9
18.6.7. Line control register (URTxLCR) 18-10
18.6.8. Modem control register (URTxMCR) 18-11
18.6.9. Line status register (URTxLSR) 18-12
18.6.10. Modem status register (URTxMSR) 18-13
18.6.11. Divider latch register (URTxDLL&URTxDLM) 18-14
18.7. UART operation 18-16
18.7.1. Example of initial setting 18-16
18.7.2. Example of transfer procedure 18-17
18.7.3. Example of reception procedure 18-18
18.7.4. Basic transmission operation 18-19
18.7.5. Basic reception operation 18-20
18.7.6. Line status 18-21
18.7.7. Character time-out interrupt 18-25

19. I^{2}C bus interface 19-1

19.1. Outline 19-1
19.2. Feature 19-1
19.3. Block diagram 19-2
19.4. Related pin 19-4
19.5. Supply clock 19-4
19.6. Register 19-5
19.6.1. Register list 19-5
19.6.2. Bus status register (I2CxBSR) 19-7
19.6.3. Bus control register (I2CxBCR) 19-9
19.6.4. Clock control register (I2CxCCR) 19-12
19.6.5. Address register (I2CxADR) 19-15
19.6.6. Data register (I2CxDAR) 19-16
19.6.7. Two bus control registers (I2CxBC2R) 19-17
19.6.8. Expansion CS register (I2CxECSR) 19-18
19.6.9. Bus clock frequency register (I2CxBCFR) 19-20
19.7. Operation 19-21
19.7.1. Start condition 19-21
19.7.2. Stop condition 19-22
19.7.3. Addressing 19-23
19.7.4. Synchronous arbitration of SCL 19-24
19.7.5. Arbitration 19-25
19.7.6. Acknowledge/Negative acknowledge 19-26
19.7.7. Bus error 19-27
19.7.8. Initialization 19-28
19.7.9. One byte transfer from master to slave 19-29
19.7.10. One byte transfer from slave to master 19-30
19.7.11. Recover from bus error 19-31
19.7.12. Interrupt process and wait request operation to master device 19-32

19.8. Notice 19-32

20. Serial peripheral interface (SPI) 20-1

20.1. Outline 20-1
20.2. Feature 20-1
20.3. Block diagram 20-2
20.4. Supply clock 20-2
20.5. Transition state 20-3
20.6. Register 20-4
20.6.1. Register list 20-4
20.6.2. SPI control register (SPICR) 20-5
20.6.3. SPI slave control register (SPISCR) 20-7
20.6.4. SPI data register (SPIDR) 20-10
20.6.5. SPI status register (SPISR) 20-11
20.7. Setup procedure flow 20-12

21. CAN interface (CAN) 21-1

21.1. Outline 21-1
21.2. Block diagram 21-1
21.3. Supply clock 21-2
21.4. Register 21-2

22. Chip Control Module (CCNT) 22-1

22.1. Outline 22-1
22.2. Feature 22-1
22.3. Block diagram 22-2
22.4. Supply clock 22-2
22.5. Register 22-3
22.5.1. Register list 22-3
22.5.2. CHIP ID register (CCID) 22-5
22.5.3. Software reset register (CSRST) 22-6
22.5.4. Interrupt status register (CIST) 22-7
22.5.5. Interrupt status mask register （CISTM） 22-9
22.5.6. GPIO interrupt status register (CGPIO_IST) 22-11
22.5.7. GPIO interrupt status mask register (CGPIO_ISTM) 22-11
22.5.8. GPIO interrupt polarity setting register (CGPIO_IP) 22-12
22.5.9. GPIO interrupt mode setting register (CGPIO_IM) 22-12
22.5.10. AXI bus wait cycle setting register (CAXI_BW) 22-13
22.5.11. AXI polarity setting register (CAXI_PS) 22-14
22.5.12. Multiplex mode setting register (CMUX_MD) 22-16
22.5.13. External pin status register (CEX_PIN_ST) 22-17
22.5.14. Byte swap switching register (CBSC) 22-18
22.5.15. DDR2 controller reset control register (CDCRC) 22-20
22.5.16. Software reset register 0 for macro (CMSR0) 22-21
22.5.17. Software reset register 1 for macro (CMSR1) 22-23

23. External interrupt controller (EXIRC) 23-1

23.1. Outline 23-1

23.2. Feature 23-1
23.3. Block diagram 23-2
23.4. Supply clock 23-2
23.5. Register 23-3
23.5.1. Register list 23-3
23.5.2. External interrupt enable register (EIENB) 23-5
23.5.3. External interrupt request register (EIREQ) 23-6
23.5.4. External interrupt level register (EILVL) 23-7
23.6. Operation 23-8
23.7. Operation procedure 23-8
23.8. Instruction for use 23-8

24. SD memory controller (SDMC) 24-1

1. Outline

This chapter describes feature, block diagram, and function of MB86R03.

1.1. Feature

MB86R03 is LSI product for the graphics applications with ARM Limited's CPU ARM926EJ-S and Fujitsu's GDC MB86296 as its core. This product contains peripheral I/O resources, such as in-vehicle LAN; therefore only a single chip of MB86R03 controls main graphics application system which usually requires 2 chips (CPU and GDC.)

MB86R03 has following features:

- CMOS 90nm technology
- Package: PBGA484
- Power-supply voltage: (IO: 3.3 ± 0.3V, core: 1.2 ± 0.1V, DDR2: 1.8 ± 0.1V)
- Operation frequency: 333MHz (CPU), 83MHz (AHB), 41.5MHz (APB)
- CPU core
    - ARM926EJ-S
    - 16KB instruction cache/16KB data cache
    - 16KB ITCM/16KB DTCM
    - ETM9CS Single and JTAG ICE interface
    - Java acceleration (Jazelle technology)
- Bus architecture
    - Multi-layer AHB bus architecture
- Interrupt
- Built-in SRAM
- Clock/Reset control function
- Remap/Boot control function
- 16 bit external bus interface with decoding engine
- 32 bit DDR2 memory interface (target: 166MHz: 333Mbps)
- Graphics display controller
    - 2D/3D rendering engine of Fujitsu MB86296
    - RGB66 video output × 1ch (extensible to RGB888 with using option I/O)
    - ITU RBT-656 video capture × 1ch (extensible to RGB666 with using option I/O)
- SD memory controller (SDIO/CPRM: unsupported) × 1ch
- 10 bit A/D converter (1MS/s) × 2ch
- I^2C (I/O voltage: 3.3V) × 2ch
- UART × 3ch (extensible up to 6ch with using option I/O)
- 32/16 bit timer × 2ch
- DMAC × 8ch

Option I/O (with pin multiplex)

- RGB666 video output is extensible to 2ch
- Video capture is extensible to 2ch
- CAN (I/O voltage: 3.3V) × 2ch is addable
- GPIO is addable up to 24
- SPI × 1ch is addable
- PWM × 2ch is addable
- I2S is addable up to 3ch
- The number of UART channel is extensible up to 6ch
- The data width in the external bus interface is extensible to 32 bit

1.2. Block diagram

Figure 1-1 shows block diagram of MB86R03.

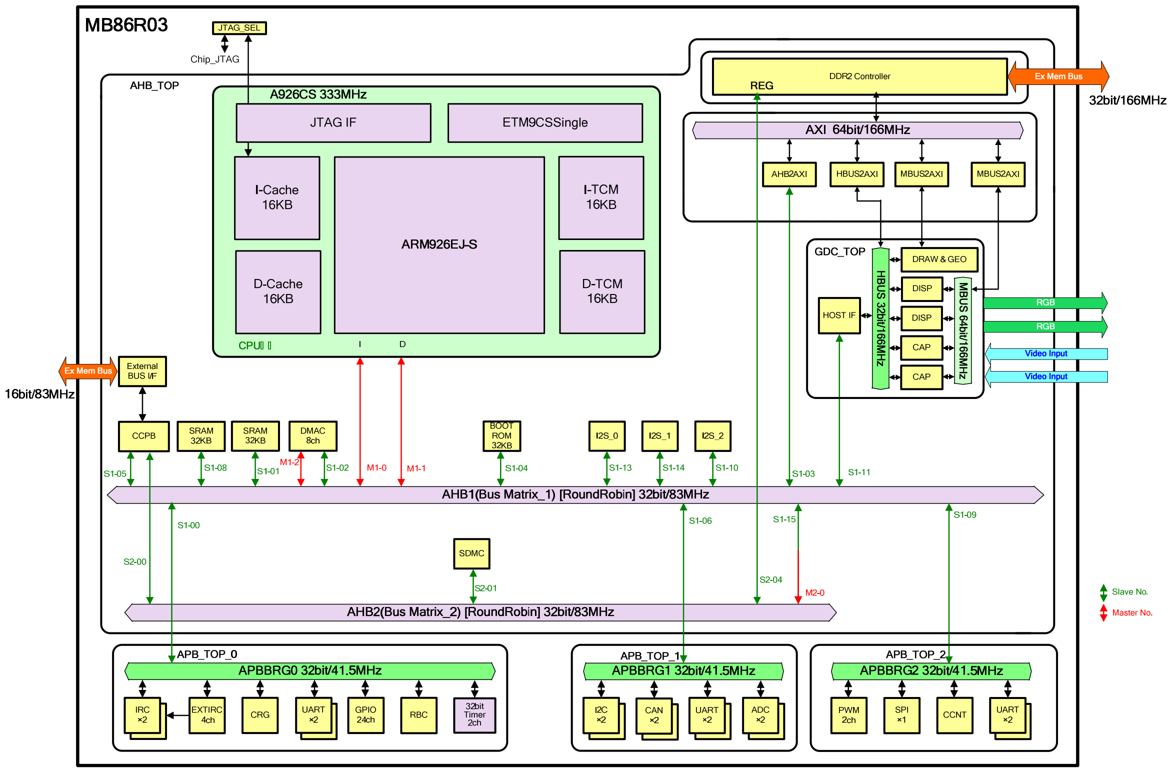


Figure 1-1 Block diagram of MB86R03

CPU core

CPU core block of ARM926EJ-S is connected to each I/O through AHB bus in LSI. Instruction (I)/Data (D) function as a separate bus master for Harvard architecture.

GDC_TOP

MB86296 compatible GDC has 2 functions: AHB slave function which writes required display list for drawing to GDC with having CPU or DMA controller as master, and AXI master function which reads display list arranged in DDR2 memory with having GDC as master.

AXI bus

This bus bridges main memory and internal resource. Following 4 bus masters are connected.

- AHB1: Each bus master of AHB bus such as CPU and DMA controller
- HBUS: HOST IF on GDC
- DRAW & GEO: Draw (2D/3D drawing) and GEO (geometry engine) on GDC
- MBUS: DISP (display controller) and CAP (video capture) on GDC

AHB1 bus

Following resources are connected.

- CPU core: Bus masters of instruction (I)/data (D)
- GDC: GDC register part
- AHB2AXI: AXI port for main memory access
- CCPB: Encrypted ROM decoding block
- External BUS I/F: External bus interface (connected through CCPB)
- SRAM: General purpose internal SRAM 32KB × 2
- DMAC: General purpose DMA × 8ch
  It operates as bus master at data transfer
- Boot ROM: Built-in boot ROM
- I2S_0/1/2: Serial audio controller × 3ch
- AHB2
- APBBRG0/1/2: AHB-APB bridge circuit × 3ch

AHB2 bus

- CCPB: Encrypted ROM decoding block
- SDMC: SD memory controller
- DDR2 controller: DDR2 controller's register part

APB_TOP_0

This block bridges between APBBRG0 bus and the AHB1 bus, and following low-speed peripheral resources are connected.

- Interrupt controller (IRC) × 2ch
- External interrupt controller (EXTIRC)
- Clock reset generator (CRG)
- UART (ch0 and ch1) × 2ch
- Remap boot controller (RBC)
- 32 bit general-purpose timer (32 bit timer) × 2ch

APB_TOP_1

This block bridges between APBBRG1 bus and AHB1 bus, and following low-speed peripheral resources are connected.

- I^2C controller × 2ch
- CAN controller × 2ch
- UART (ch2 and ch3) × 2ch
- A/D converter (ADC) × 2ch

APB_TOP_2

This block bridges between APBBRG2 bus and AHB1 bus, and following low-speed peripheral resources are connected.

- PWM controller (PWM)
- SPI controller (SPI)
- Chip control module (CCNT)
- UART (ch4 and ch5) × 2ch

1.3. Function list

Function list of MB86R03 is shown below.

Table 1-1 MB86R03 function list

Function	Outline
CPU core	• ARM926EJ-S™ processor core • Core operation frequency: 333MHz • 16KB instruction cache • 16KB data cache • Tightly-Coupled memory for 16KB instruction (ITCM) • Tightly-Coupled memory for 16KB data (DTCM) • ETM9CS Single and JTAG ICE debugging interface • Java acceleration (Jazelle technology)
Bus architecture	• Multilayer AHB bus architecture • Speeding up data transfer between main memory and each bus master with 64 bit AXI bus
Interrupt	• High-speed interrupt × 1ch (software interrupt) • Normal interrupt × 64ch (external interrupt × 4ch + built-in internal interrupt × 60ch) • Up to 16 interrupt levels are settable by channel
Clock	• PLL multiplication: selectable from ×15 ~ 49 • Operation frequency: 333MHz (CPU), 83MHz (AHB), 41.5MHz (APB) • Low power consumption mode (clock to ARM and module is stoppable)
Reset	• Hardware reset, software reset, and watchdog reset
Remap	• ROM area is able to be mapping to built-in SRAM area
External bus interface	• Three chip select signals • Provided 32M byte address space in each chip select • Supported 16/32 bit width SRAM/Flash ROM connection • Programmable weight controller • Encrypted ROM compound engine
DDR2 controller	• Supported DDR2SDRAM (DDR2-400) • Connectable capacity: 256 ~ 512M bit × 2 or 256 ~ 512M bit × 1 • I/O width: Selectable from ×16/×32 bit • Max. transfer rate: 166MHz/333Mbps
Built-in SRAM	• Mounted general purpose SRAM of 32KB × 2 (32 bit bus)
DMAC	• AHB connection × 8ch • Transfer mode: Block, burst, and demand
Timer	• 32/16 bit programmable × 2 channels
GPIO(*2)	• Max. 24 is usable • Interrupt function
PWM(*2)	• Built-in 2 channels • Duty ratio and phase are configurable
A/D converter	• 10 bit successive approximation type A/D converter × 2ch • Sampling rate: 648KS/s (max. sampling plate) • Nonlinearity error: ± 2.0LSB (max.)

Function	Outline
GDC (*1)	• Display controller RGB666 or RGB888 output Max. resolution is 1024 × 768 Max. 6 layered display Max. 2 screen output • Digital video capture function BT.601, BT.656, and RGB666 Max. 2 inputs • Geometry engine (MB86296 compatible display list is usable) • 2D/3D drawing function (MB86296 compatible display list is usable)
I^2S (*2)	• Audio output × 3ch (L/R) /Audio input × 3ch (L/R) • Supported three-wire serial (I2S, MSB-Justified) and serial PCM data transfer interface • Master/Slave operations are selectable • Resolution capability: Max. 32 bit/sample
UART (*2)	• Max. 6 channels (dedicated channel: 3ch, option: 3ch) • 1 channel: capable of input/output CTS/RTS signals • 8 bit pre-scaler for baud rate clock generation • Enabled DMA transfer
I2C	• 3.3V pin × 2ch • Supported standard mode (max. 100kbps)/high-speed mode (max. 400kbps)
SPI (*2)	• Full duplex/Synchronous transmission • Transfer data length: 1 bit unit (max. 32 bit) (programmable setting)
CAN (*2)	• Mounted BOSCH C_CAN module × 2ch • Conformed to CAN protocol version 2.0 part A and B • I/O voltage: 3.3V
SD memory	• Conformed to SD memory card physical layer specification 1.0 • Equipped 1 channel • Supported SD memory card and multimedia card • Unsupported SPI mode, SDIO mode, and CPRM
CCNT	• Mode selection of multiplex pin group 2 and 4 • Software reset control • AXI interconnection control (priority and WAIT setting)
JTAG	• Conformed to IEIEEE1149.1 (IEEE Standard Test Access Port and Boundary-Scan Architecture) • Supported JTAG ICE connection

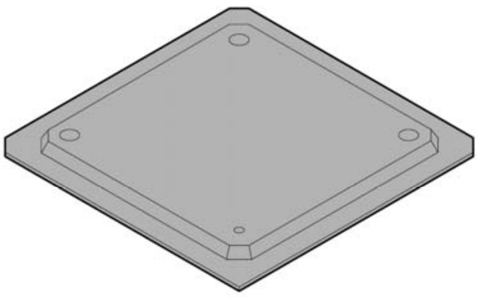
*1: Number of layer of simultaneous display and number of output display as well as capture input for displaying in high resolution may be restricted due to data supply capacity of graphics memory (DDR2 controller).

*2: A part of external pin functions of this LSI is multiplexed. Max. number of usable channel is limited by pin multiplex function setting.

1.4. Package dimension

Package dimension of MB86R03 is shown below.

BGA-484P-M07

484-pin plastic PBGA  (BGA-484P-M07)	Ball pitch	1.00 mm
	Package width × package length	27.00 mm × 27.00 mm
	Lead shape	Ball
	Sealing method	Plastic mold
	Mounting height	2.36 mm Max

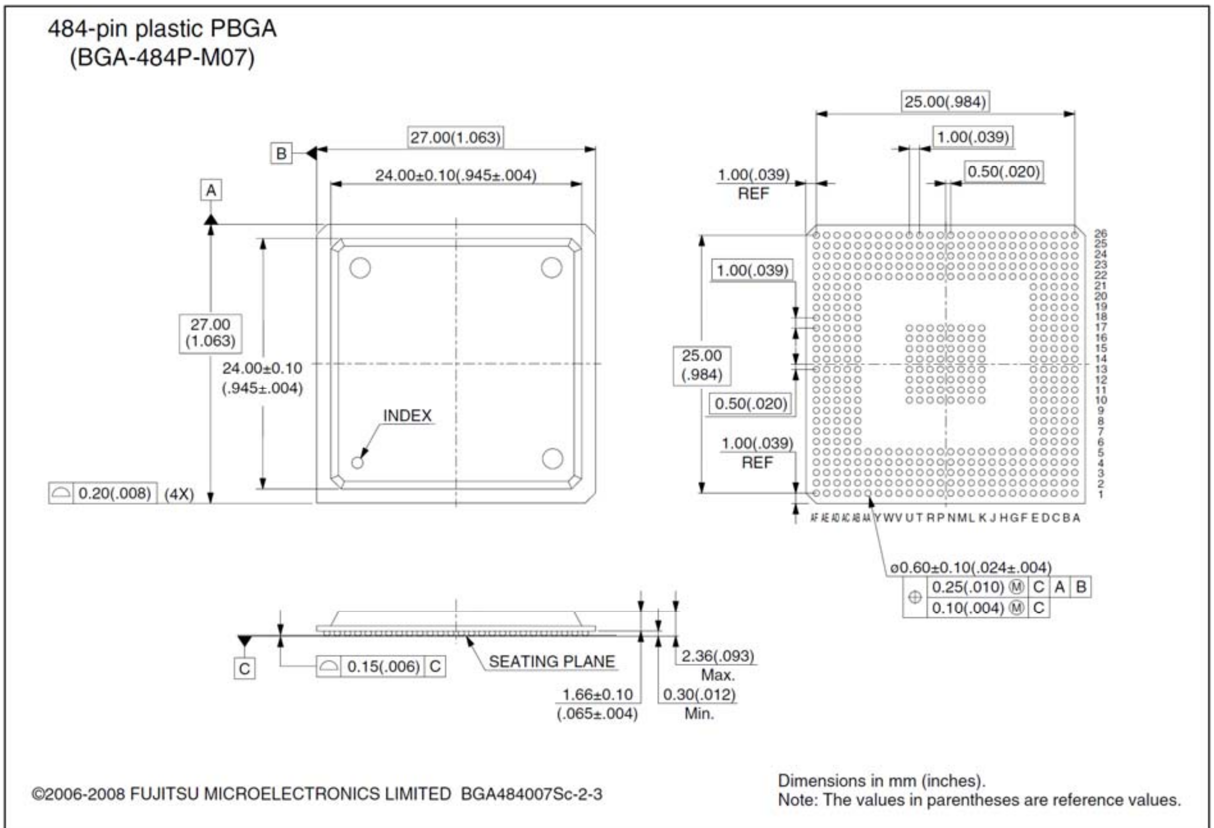


Figure 1-2 BGA-484P-M07 package dimension

1.5. Pin assignment

Pin assignment of MB86R03 is shown below.

(Top view)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
A	1	100	99	98	97	96	95	94	93	92	91	90	89	88	87	86	85	84	83	82	81	80	79	78	77	76
B	2	101	192	191	190	189	188	187	186	185	184	183	182	181	180	179	178	177	176	175	174	173	172	171	170	75
C	3	102	193	276	275	274	273	272	271	270	269	268	267	266	265	264	263	262	261	260	259	258	257	256	169	74
D	4	103	194	277	352	351	350	349	348	347	346	345	344	343	342	341	340	339	338	337	336	335	334	255	168	73
E	5	104	195	278	353	420	419	418	417	416	415	414	413	412	411	410	409	408	407	406	405	404	333	254	167	72
F	6	105	196	279	354																	403	332	253	166	71
G	7	106	197	280	355																	402	331	252	165	70
H	8	107	198	281	356																	401	330	251	164	69
J	9	108	199	282	357																	400	329	250	163	68
K	10	109	200	283	358					421	448	447	446	445	444	443	442					399	328	249	162	67
L	11	110	201	284	359					422	449	468	467	466	465	464	441					398	327	248	161	66
M	12	111	202	285	360					423	450	469	480	479	478	463	440					397	326	247	160	65
N	13	112	203	286	361					424	451	470	481	484	477	462	439					396	325	246	159	64
P	14	113	204	287	362					425	452	471	482	483	476	461	438					395	324	245	158	63
R	15	114	205	288	363					426	453	472	473	474	475	460	437					394	323	244	157	62
T	16	115	206	289	364					427	454	455	456	457	458	459	436					393	322	243	156	61
U	17	116	207	290	365					428	429	430	431	432	433	434	435					392	321	242	155	60
V	18	117	208	291	366																	391	320	241	154	59
W	19	118	209	292	367																	390	319	240	153	58
Y	20	119	210	293	368																	389	318	239	152	57
AA	21	120	211	294	369																	388	317	238	151	56
AB	22	121	212	295	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	316	237	150	55
AC	23	122	213	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	236	149	54
AD	24	123	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	148	53
AE	25	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	52
AF	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51

Figure 1-3 MB86R03 pin assignment (pin number)

(Top view)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
A	VSS	VSS	DCLKO0	VSS	DCLKIN0	DOUTG0 [6]	DOUTG0 [2]	DOUTB0 [4]	XSRST	TRACE DATA[3]	XRST	PLLVSS	PLLVDD	TDO	VSS	CLK	MEM XRD	VSS	MEM EA[20]	MEM EA[16]	MEM EA[12]	MEM EA[8]	MEM EA[4]	MEM EA[1]	VSS	VSS
B	VSS	DE0	HSYNC0	VDDE	DOUTR0 [4]	DOUTG0 [7]	DOUTG0 [3]	DOUTB0 [5]	XTRST	TRACE CTL	TRACE DATA[0]	TMS	VINITHI	CRIPM3	VDDE	MEM XCS[4]	MEM XWR[1]	MEM EA[23]	MEM EA[19]	MEM EA[15]	MEM EA[11]	MEM EA[7]	MEM EA[3]	MEM ED[15]	MEM ED[14]	VSS
C	DOUTB1 [2]	GV0	VSYNC0	DOUTR0 [7]	DOUTR0 [5]	DOUTR0 [2]	DOUTG0 [4]	DOUTB0 [6]	DOUTB0 [2]	TRACE CLK	TRACE DATA[1]	JTAGSEL	TCK	CRIPM2	CRIPM0	MEM XCS[2]	MEM XWR[0]	MEM EA[22]	MEM EA[18]	MEM EA[14]	MEM EA[10]	MEM EA[6]	MEM EA[2]	MEM ED[13]	MEM ED[12]	MEM ED[11]
D	DOUTB1 [6]	DOUTB1 [5]	DOUTB1 [4]	DOUTB1 [3]	DOUTR0 [6]	DOUTR0 [3]	DOUTG0 [5]	DOUTB0 [7]	DOUTB0 [3]	RTCK	TRACE DATA[2]	LLTDTRS	TDI	CRIPM1	MEM RDY	MEM XCS[0]	MEM EA[24]	MEM EA[21]	MEM EA[17]	MEM EA[13]	MEM EA[9]	MEM EA[5]	MEM ED[10]	MEM ED[9]	MEM ED[8]	MEM ED[7]
E	DOUTG1 [4]	DOUTG1 [3]	DOUTG1 [2]	DOUTB1 [7]	VDDE	VSS	VSS	VDDE	VDDE	VDDI	VDDI	VSS	VSS	VDDE	VDDE	VDDI	VDDI	VSS	VSS	VDDE	VDDE	VDDI	MEM ED[6]	MEM ED[5]	MEM ED[4]	MEM ED[3]
F	DOUTR1 [2]	DOUTG1 [7]	DOUTG1 [6]	DOUTG1 [5]	VDDE																	VDDI	MEM ED[2]	MEM ED[1]	MEM ED[0]	VSS
G	DCLKIN 1	DOUTR1 [5]	DOUTR1 [4]	DOUTR1 [3]	VDDI																	VSS	MDQ[30]	MDM[3]	MDQ[31]	MDQS P[3]
H	VSS	VDDE	DOUTR1 [7]	DOUTR1 [6]	VDDI																	VSS	MDQ[25]	MDQ[28]	MDQ[24]	MDQS N[3]
J	DCLKO1	GV1	VSYNC1	HSYNC1	VSS																	DDRVDE	MDQ[27]	MDQ[26]	MDQ[29]	VSS
K	VIN0 [5]	VIN0 [6]	VIN0 [7]	DE1	VSS					VDDI	VDDE	VDDE	VDDI	VDDI	VDDE	VDDE	VDDI					DDRVDE	MDM[2]	MDQ[23]	VREF1	MDQS P[2]
L	VIN0 [1]	VIN0 [2]	VIN0 [3]	VIN0 [4]	VDDE					VDDI	VSS	VSS	VSS	VSS	VSS	VSS	VDDI					DQ22	MDQ[20]	MDQ[17]	MDQ[16]	MDQS N[2]
M	CCLK0	VDDE	VIN VSYNC0	VIN0 [0]	VDDE					VDDE	VSS	VSS	VSS	VSS	VSS	VSS	DDRVDE					VSS	MDQ[19]	MDQ[18]	MDQ[21]	VSS
N	VSS	VINFID0	VIN HSYNC0	VDDI	VDDI					VDDE	VSS	VSS	VSS	VSS	VSS	VSS	DDRVDE					VDDI	ODT	VSS	DDRVDE	MCKP
P	VSS	VSS or VDDI	VSS	VSS	VSS or VDDE					VDDI	VSS	VSS	VSS	VSS	VSS	VSS	VDDI					VDDI	OCD	VSS	DDRVDE	MCKN
R	VSS	VSS	VSS or VDDE	VSS	VSS					VDDI	VSS	VSS	VSS	VSS	VSS	VSS	VDDI					VSS	MDQ[14]	MDM[1]	MDQ[15]	VSS
T	VSS	VSS	VSS	VSS	VSS					VDDE	VSS	VSS	VSS	VSS	VSS	VSS	DDRVDE					DDRVDE	MDQ[12]	MDQ[9]	MDQ[8]	MDQS P[1]
U	VSS	VSS	VSS or VDDI	VSS	VDDI					VDDE	VDDI	VDDI	VDDE	VDDE	VDDI	VDDI	DDRVDE					DDRVDE	MDQ[11]	MDQ[10]	MDQ[13]	MDQS N[1]
V	VSS	VSS	VIN1 [7]	VSS	VDDI																	MDQ[6]	MDM[0]	MDQ[7]	VREF0	VSS
W	VIN1 [6]	VIN1 [5]	VIN1 [4]	VIN1 [3]	VDDE																	VSS	MDQ[4]	MDQ[1]	MDQ[0	MDQS P[0]
Y	VSS	VIN1 [2]	VIN1 [1]	VIN1 [0]	VDDE																	VSS	MDQ[3]	MDQ[5]	MDQ[2]	MDQS N[0]
AA	CCLK1	VDDE	VIN VSYNC1	VIN HSYNC1	VSS																	DDRVDE	MCAS	MRAS	MCKE	VSS
AB	VINFID1	I2S SDO2	I2S SDI2	I2S WS2	VSS	VDDE	VDDE	VDDI	VDDI	VSS	VSS	VDDE	AD VRL0	AD VRL1	VSS	VSS	VSS	VDDE	VDDE	VDDI	VDDI	DDRVDE	MCS	MWE	MBA[0]	MBA[1]
AC	I2S SCK2	PWM_O1	(Unused)	I2S_SDO1	CAN_TX0	GPIO_PD [23]	GPIO_PD [19]	GPIO_PD [15]	(Unused)	(Unused)	MPX MODE_1 [0]	TEST MODE[0]	AD VR0	AD VR1	VDDE	UART SIN2	SD CLK	SD DAT[3]	VPD	INT_A [2]	DDRTYPE	ODTCONT	MA[0]	MA[2]	MA[10]	MA[1]
AD	I2S ECLK2	PWM_O0	I2S_SCK1	I2S_ ECLK1	CAN_RX0	GPIO_PD [22]	GPIO_PD [18]	GPIO_PD [14]	PWM_O1	(Unused)	MPX MODE_1 [1]	PLL BYPASS	AD VIN0	AD VIN1	VDDE	UART SOUT2	SD CMD	SD DAT[2]	(Unused)	I2C SDA0	INT_A [1]	TEST MODE[2]	MA[9]	MA[6]	MA[5]	MA[3]
AE	VSS	VSS	I2S_WS1	(Unused)	CAN_TX1	GPIO_PD [21]	GPIO_PD [17]	GPIO_PD [13]	PWM_O0	(Unused)	MPX MODE_5 [0]	BIGEND	AD VRH0	AD VRH1	UART XRTS0	UART XCTS0	UART SOUT1	SD DAT[1]	SD XMCD	I2C SCL0	INT_A [3]	MCKE START	MA[13]	MA[4]	MA[11]	MA[7]
AF	VSS	VSS	I2S_SDI1	(Unused)	CAN_RX1	GPIO_PD [20]	GPIO_PD [16]	(Unused)	(Unused)	(Unused)	MPX MODE_5 [1]	TEST MODE[1]	AD AVD	AD AVS	UART SOUT0	UART SIN0	UART SIN1	SD DAT[0]	SD WP	I2C SCL1	I2C SDA1	INT_A [0]	MA[8]	MA[12]	VSS	VSS

Figure 1-4 MB86R03 pin assignment (pin name)

Table 1-2 Pin assignment table

Pin NO	JEDEC	PIN NAME	Pin NO	JEDEC	PIN NAME	Pin NO	JEDEC	PIN NAME	Pin NO	JEDEC	PIN NAME	Pin NO	JEDEC	PIN NAME
1	A1	VSS	101	B2	DE0	201	L3	VIN0[3]	301	AC9	(Unused)	401	H22	VSS
2	B1	VSS	102	C2	GV0	202	M3	VINVSYNC0	302	AC10	(Unused)	402	G22	VSS
3	C1	DOUTB1[2]	103	D2	DOUTB1[5]	203	N3	VINHSYNC0	303	AC11	MPX_MODE_1[0]	403	F22	VDDI
4	D1	DOUTB1[6]	104	E2	DOUTG1[3]	204	P3	VSS	304	AC12	TESTMODE[0]	404	E22	VDDI
5	E1	DOUTG1[4]	105	F2	DOUTG1[7]	205	R3	VSS or VDDE	305	AC13	AD_VR0	405	E21	VDDE
6	F1	DOUTR1[2]	106	G2	DOUTR1[5]	206	T3	VSS	306	AC14	AD_VR1	406	E20	VDDE
7	G1	DCLKIN1	107	H2	VDDE	207	U3	VSS or VDDI	307	AC15	VDDE	407	E19	VSS
8	H1	VSS	108	J2	GV1	208	V3	VIN1[7]	308	AC16	UART_SIN2	408	E18	VSS
9	J1	DCLKO1	109	K2	VIN0[6]	209	W3	VIN1[4]	309	AC17	SD_CLK	409	E17	VDDI
10	K1	VIN0[5]	110	L2	VIN0[2]	210	Y3	VIN1[1]	310	AC18	SD_DAT[3]	410	E16	VDDI
11	L1	VIN0[1]	111	M2	VDDE	211	AA3	VINVSYNC1	311	AC19	VPD	411	E15	VDDE
12	M1	CCLK0	112	N2	VINFID0	212	AB3	I2S_SDI2	312	AC20	INT_A[2]	412	E14	VDDE
13	N1	VSS	113	P2	VSS or VDDI	213	AC3	(Unused)	313	AC21	DDRTYPE	413	E13	VSS
14	P1	VSS	114	R2	VSS	214	AD3	I2S_SCK1	314	AC22	ODTCONT	414	E12	VSS
15	R1	VSS	115	T2	VSS	215	AD4	I2S_ECLK1	315	AC23	MA[0]	415	E11	VDDI
16	T1	VSS	116	U2	VSS	216	AD5	CAN_RX0	316	AB23	MCS	416	E10	VDDI
17	U1	VSS	117	V2	VSS	217	AD6	GPIO_PD[22]	317	AA23	MCAS	417	E9	VDDE
18	V1	VSS	118	W2	VIN1[5]	218	AD7	GPIO_PD[18]	318	Y23	MDQ[3]	418	E8	VDDE
19	W1	VIN1[6]	119	Y2	VIN1[2]	219	AD8	GPIO_PD[14]	319	W23	MDQ[4]	419	E7	VSS
20	Y1	VSS	120	AA2	VDDE	220	AD9	PWM_O1	320	V23	MDM[0]	420	E6	VSS
21	AA1	CCLK1	121	AB2	I2S_SDO2	221	AD10	(Unused)	321	U23	MDQ[11]	421	K10	VDDI
22	AB1	VINFID1	122	AC2	PWM_O1	222	AD11	MPX_MODE_1[1]	322	T23	MDQ[12]	422	L10	VDDI
23	AC1	I2S_SCK2	123	AD2	PWM_O0	223	AD12	PLLBYPASS	323	R23	MDQ[14]	423	M10	VDDE
24	AD1	I2S_ECLK2	124	AE2	VSS	224	AD13	AD_VIN0	324	P23	OCD	424	N10	VDDE
25	AE1	VSS	125	AE3	I2S_WS1	225	AD14	AD_VIN1	325	N23	ODT	425	P10	VDDI
26	AF1	VSS	126	AE4	(Unused)	226	AD15	VDDE	326	M23	MDQ[19]	426	R10	VDDI
27	AF2	VSS	127	AE5	CAN_TX1	227	AD16	UART_SOUT2	327	L23	MDQ[20]	427	T10	VDDE
28	AF3	I2S_SDI1	128	AE6	GPIO_PD[21]	228	AD17	SD_CMD	328	K23	MDM[2]	428	U10	VDDE
29	AF4	(Unused)	129	AE7	GPIO_PD[17]	229	AD18	SD_DAT[2]	329	J23	MDQ[27]	429	U11	VDDI
30	AF5	CAN_RX1	130	AE8	GPIO_PD[13]	230	AD19	(Unused)	330	H23	MDQ[25]	430	U12	VDDI
31	AF6	GPIO_PD[20]	131	AE9	PWM_O0	231	AD20	I2C_SDA0	331	G23	MDQ[30]	431	U13	VDDE
32	AF7	GPIO_PD[16]	132	AE10	(Unused)	232	AD21	INT_A[1]	332	F23	MEM_ED[2]	432	U14	VDDE
33	AF8	(Unused)	133	AE11	MPX_MODE_5[0]	233	AD22	TESTMODE[2]	333	E23	MEM_ED[6]	433	U15	VDDI
34	AF9	(Unused)	134	AE12	BIGEND	234	AD23	MA[9]	334	D23	MEM_ED[10]	434	U16	VDDI
35	AF10	(Unused)	135	AE13	AD_VRH0	235	AD24	MA[6]	335	D22	MEM_EA[5]	435	U17	DDRVDE
36	AF11	MPX_MODE_5[1]	136	AE14	AD_VRH1	236	AC24	MA[2]	336	D21	MEM_EA[9]	436	T17	DDRVDE
37	AF12	TESTMODE[1]	137	AE15	UART_XRTS0	237	AB24	MWE	337	D20	MEM_EA[13]	437	R17	VDDI
38	AF13	AD_AVD	138	AE16	UART_XCTS0	238	AA24	MRAS	338	D19	MEM_EA[17]	438	P17	VDDI
39	AF14	AD_AVS	139	AE17	UART_SOUT1	239	Y24	MDQ[5]	339	D18	MEM_EA[21]	439	N17	DDRVDE
40	AF15	UART_SOUT0	140	AE18	SD_DAT[1]	240	W24	MDQ[1]	340	D17	MEM_EA[24]	440	M17	DDRVDE
41	AF16	UART_SIN0	141	AE19	SD_XMCD	241	V24	MDQ[7]	341	D16	MEM_XCS[0]	441	L17	VDDI
42	AF17	UART_SIN1	142	AE20	I2C_SCL0	242	U24	MDQ[10]	342	D15	MEM_RDY	442	K17	VDDI
43	AF18	SD_DAT[0]	143	AE21	INT_A[3]	243	T24	MDQ[9]	343	D14	CRIPM1	443	K16	VDDE
44	AF19	SD_WP	144	AE22	MCKE_START	244	R24	MDM[1]	344	D13	TDI	444	K15	VDDE
45	AF20	I2C_SCL1	145	AE23	MA[13]	245	P24	VSS	345	D12	PLLTDTRST	445	K14	VDDI
46	AF21	I2C_SDA1	146	AE24	MA[4]	246	N24	VSS	346	D11	TRACEDATA[2]	446	K13	VDDI
47	AF22	INT_A[0]	147	AE25	MA[11]	247	M24	MDQ[18]	347	D10	RTCK	447	K12	VDDE
48	AF23	MA[8]	148	AD25	MA[5]	248	L24	MDQ[17]	348	D9	DOUTB0[3]	448	K11	VDDE
49	AF24	MA[12]	149	AC25	MA[10]	249	K24	MDQ[23]	349	D8	DOUTB0[7]	449	L11	VSS
50	AF25	VSS	150	AB25	MBA[0]	250	J24	MDQ[26]	350	D7	DOUTG0[5]	450	M11	VSS
51	AF26	VSS	151	AA25	MCKE	251	H24	MDQ[28]	351	D6	DOUTR0[3]	451	N11	VSS
52	AE26	MA[7]	152	Y25	MDQ[2]	252	G24	MDM[3]	352	D5	DOUTR0[6]	452	P11	VSS
53	AD26	MA[3]	153	W25	MDQ[0]	253	F24	MEM_ED[1]	353	E5	VDDE	453	R11	VSS
54	AC26	MA[1]	154	V25	VREF0	254	E24	MEM_ED[5]	354	F5	VDDE	454	T11	VSS
55	AB26	MBA[1]	155	U25	MDQ[13]	255	D24	MEM_ED[9]	355	G5	VDDI	455	T12	VSS
56	AA26	VSS	156	T25	MDQ[8]	256	C24	MEM_ED[13]	356	H5	VDDI	456	T13	VSS
57	Y26	MDQSN[0]	157	R25	MDQ[15]	257	C23	MEM_EA[2]	357	J5	VSS	457	T14	VSS
58	W26	MDQSP[0]	158	P25	DDRVDE	258	C22	MEM_EA[6]	358	K5	VSS	458	T15	VSS
59	V26	VSS	159	N25	DDRVDE	259	C21	MEM_EA[10]	359	L5	VDDE	459	T16	VSS
60	U26	MDQSN[1]	160	M25	MDQ[21]	260	C20	MEM_EA[14]	360	M5	VDDE	460	R16	VSS
61	T26	MDQSP[1]	161	L25	MDQ[16]	261	C19	MEM_EA[18]	361	N5	VDDI	461	P16	VSS
62	R26	VSS	162	K25	VREF1	262	C18	MEM_EA[22]	362	P5	VSS or VDDE	462	N16	VSS
63	P26	MCKN	163	J25	MDQ[29]	263	C17	MEM_XWR[0]	363	R5	VSS	463	M16	VSS
64	N26	MCKP	164	H25	MDQ[24]	264	C16	MEM_XCS[2]	364	T5	VSS	464	L16	VSS
65	M26	VSS	165	G25	MDQ[31]	265	C15	CRIPM0	365	U5	VDDI	465	L15	VSS
66	L26	MDQSN[2]	166	F25	MEM_ED[0]	266	C14	CRIPM2	366	V5	VDDI	466	L14	VSS
67	K26	MDQSP[2]	167	E25	MEM_ED[4]	267	C13	TCK	367	W5	VDDE	467	L13	VSS
68	J26	VSS	168	D25	MEM_ED[8]	268	C12	JTAGSEL	368	Y5	VDDE	468	L12	VSS
69	H26	MDQSN[3]	169	C25	MEM_ED[12]	269	C11	TRACEDATA[1]	369	AA5	VSS	469	M12	VSS
70	G26	MDQSP[3]	170	B25	MEM_ED[14]	270	C10	TRACECLK	370	AB5	VSS	470	N12	VSS
71	F26	VSS	171	B24	MEM_ED[15]	271	C9	DOUTB0[2]	371	AB6	VDDE	471	P12	VSS
72	E26	MEM_ED[3]	172	B23	MEM_EA[3]	272	C8	DOUTB0[6]	372	AB7	VDDE	472	R12	VSS
73	D26	MEM_ED[7]	173	B22	MEM_EA[7]	273	C7	DOUTG0[4]	373	AB8	VDDI	473	R13	VSS
74	C26	MEM_ED[11]	174	B21	MEM_EA[11]	274	C6	DOUTR0[2]	374	AB9	VDDI	474	R14	VSS
75	B26	VSS	175	B20	MEM_EA[15]	275	C5	DOUTR0[5]	375	AB10	VSS	475	R15	VSS
76	A26	VSS	176	B19	MEM_EA[19]	276	C4	DOUTR0[7]	376	AB11	VSS	476	P15	VSS
77	A25	VSS	177	B18	MEM_EA[23]	277	D4	DOUTB1[3]	377	AB12	VDDE	477	N15	VSS
78	A24	MEM_EA[1]	178	B17	MEM_XWR[1]	278	E4	DOUTB1[7]	378	AB13	AD_VRL0	478	M15	VSS
79	A23	MEM_EA[4]	179	B16	MEM_XCS[4]	279	F4	DOUTG1[5]	379	AB14	AD_VRL1	479	M14	VSS
80	A22	MEM_EA[8]	180	B15	VDDE	280	G4	DOUTR1[3]	380	AB15	VSS	480	M13	VSS
81	A21	MEM_EA[12]	181	B14	CRIPM3	281	H4	DOUTR1[6]	381	AB16	VSS	481	N13	VSS
82	A20	MEM_EA[16]	182	B13	VINITHI	282	J4	HSYNC1	382	AB17	VSS	482	P13	VSS
83	A19	MEM_EA[20]	183	B12	TMS	283	K4	DE1	383	AB18	VDDE	483	P14	VSS
84	A18	VSS	184	B11	TRACEDATA[0]	284	L4	VIN0[4]	384	AB19	VDDE	484	N14	VSS
85	A17	MEM_XRD	185	B10	TRACECTL	285	M4	VIN0[0]	385	AB20	VDDI			
86	A16	CLK	186	B9	XTRST	286	N4	VDDI	386	AB21	VDDI			
87	A15	VSS	187	B8	DOUTB0[5]	287	P4	VSS	387	AB22	DDRVDE			
88	A14	TDO	188	B7	DOUTG0[3]	288	R4	VSS	388	AA22	DDRVDE			
89	A13	PLLVDD	189	B6	DOUTG0[7]	289	T4	VSS	389	Y22	VSS			
90	A12	PLLVSS	190	B5	DOUTR0[4]	290	U4	VSS	390	W22	VSS			
91	A11	XRST	191	B4	VDDE	291	V4	VSS	391	V22	MDQ[6]			
92	A10	TRACEDATA[3]	192	B3	HSYNC0	292	W4	VIN1[3]	392	U22	DDRVDE			
93	A9	XSRST	193	C3	VSYNC0	293	Y4	VIN1[0]	393	T22	DDRVDE			
94	A8	DOUTB0[4]	194	D3	DOUTB1[4]	294	AA4	VINHSYNC1	394	R22	VSS			
95	A7	DOUTG0[2]	195	E3	DOUTG1[2]	295	AB4	I2S_WS2	395	P22	VDDI			
96	A6	DOUTG0[6]	196	F3	DOUTG1[6]	296	AC4	I2S_SDO1	396	N22	VDDI			
97	A5	DCLKIN0	197	G3	DOUTR1[4]	297	AC5	CAN_TX0	397	M22	VSS			
98	A4	VSS	198	H3	DOUTR1[7]	298	AC6	GPIO_PD[23]	398	L22	MDQ[22]			
99	A3	DCLKO0	199	J3	VSYNC1	299	AC7	GPIO_PD[19]	399	K22	DDRVDE			
100	A2	VSS	200	K3	VIN0[7]	300	AC8	GPIO_PD[15]	400	J22	DDRVDE			

1.6. Pin function

External pin function of MB86R03 is described below.

1.6.1. Pin Multiplex

This LSI adopts pin multiplex function, and a part of external pin function is multiplexed.
The external pin function is categorized into following four groups. Each group is able to set the external pin function individually; therefore, the function can be flexibly set depending on the peripheral I/O resource to be used.

1. Pin multiplex group #1 (setting pin: MPX_MODE_1[1:0])
    - Mode 0: Pin related to DISPLAY1
    - Mode 1: Pin related to external bus interface
    - Mode 2: Pin related to I2S0, GPIO, and DISPLAY0 data width extension
2. Pin multiplex group #2 (setting register: CMUX_MD.MPX_MODE_2[2:0])
    - Mode 0: Pin related to CAP1, CAP0 synchronizing signal, PWM, and I2S2
    - Mode 1: Pin related to CAP1 (NRGB666)
    - Mode 2: Pin related to GPIO, CAN, I2S1, and I2S2
    - Mode 3: Pin related to GPIO, CAN, I2S1, and SPI
    - Mode 4: Pin related to GPIO, CAN, I2S1, and I2S2 (input)
3. Pin multiplex group #3 (Reserved)
4. Pin multiplex group #4 (setting register: CMUX_MD.MPX_MODE_4[1:0])
    - Mode 1: Pin related to I2S1, CAN, GPIO, and PWM
5. Pin multiplex group #5 (setting pin: MPX_MODE_5[1:0])
    - Mode 0: Pin related to ETM
    - Mode 1: Pin related to UART3, UART4, and UART5
    - Mode 2: Pin related to UART3, UART4, and PWM

Note:

Mode should be changed when each pin is not in operation.
PWM, I2S1, and CAN pins may be duplicated and allocated to external pin depending on group combination; in this case, use either of them. For unused pin, follow the procedure in 1.6.24, Unused pin in the duplex case with pin multiplex function.

Pin multiplex group #1 (setting pin: MPX_MODE_1 [1:0])

Table 1-3 Pin function of pin multiplex group #1 by mode

Pin No.	JEDEC	Mode 0	Mode 1	Mode 2			
		Pin related to DISPLAY1	Pin related to external bus interface	Pin related to I2S0	Pin related to GPIO	Pin related to DISPLAY0	Pin related to external bus interface
198	H3	DOUTR1[7]	MEM_ED[31]	I2S_ECLK0	-	-	-
281	H4	DOUTR1[6]	MEM_ED[30]	I2S_SCK0	-	-	-
106	G2	DOUTR1[5]	MEM_ED[29]	I2S_WS0	-	-	-
197	G3	DOUTR1[4]	MEM_ED[28]	I2S_SDI0	-	-	-
280	G4	DOUTR1[3]	MEM_ED[27]	I2S_SDO0	-	-	-
6	F1	DOUTR1[2]	MEM_ED[26]	-	GPIO_PD[12]	-	-
105	F2	DOUTG1[7]	MEM_ED[25]	-	GPIO_PD[11]	-	-
196	F3	DOUTG1[6]	MEM_ED[24]	-	GPIO_PD[10]	-	-
279	F4	DOUTG1[5]	MEM_ED[23]	-	GPIO_PD[9]	-	-
5	E1	DOUTG1[4]	MEM_ED[22]	-	GPIO_PD[8]	-	-
104	E2	DOUTG1[3]	MEM_ED[21]	-	GPIO_PD[7]	-	-
195	E3	DOUTG1[2]	MEM_ED[20]	-	GPIO_PD[6]	-	-
278	E4	DOUTB1[7]	MEM_ED[19]	-	-	DOUTR0[1]	-
4	D1	DOUTB1[6]	MEM_ED[18]	-	-	DOUTR0[0]	-
103	D2	DOUTB1[5]	MEM_ED[17]	-	-	DOUTG0[1]	-
194	D3	DOUTB1[4]	MEM_ED[16]	-	-	DOUTG0[0]	-
277	D4	DOUTB1[3]	MEM_XWR[3]	-	-	DOUTB0[1]	-
3	C1	DOUTB1[2]	MEM_XWR[2]	-	-	DOUTB0[0]	-
283	K4	DE1	XDACK[7]	-	-	-	XDACK[7]
282	J4	HSYNC1	DREQ[6]	-	-	-	DREQ[6]
199	J3	VSYNC1	XDACK[6]	-	-	-	XDACK[6]
108	J2	GV1	DREQ[7]	-	-	-	DREQ[7]

Pin multiplex group #1 mode setting

This mode is set with external pin, MPX_MODE_1[1:0].

Table 1-4 Mode setting of pin multiplex group #1

MPX_MODE_1[1] pin	MPX_MODE_1[0] pin	Pin multiplex group #1 mode
"L"	"L"	Mode 0
"L"	"H"	Mode 1
"H"	"L"	Mode 2
"H"	"H"	Mode 0

Pin multiplex group #2 (setting register: PIN MPX Select.MPX_MODE_2 [2:0])

Table 1-5 Pin function of pin multiplex group #2 by mode

Pin No.	JEDEC	Mode 0			Mode 1	Mode 2			Mode 3				Mode 4		
		Pin related to CAP0/1	Pin related to PWM	Pin related to I2S2	Pin related to CAP1 (NRGB666)	Pin related to GPIO	Pin related to CAN	Pin related to I2S1/2	Pin related to GPIO	Pin related to CAN	Pin related to I2S1	Pin related to SPI	Pin related to GPIO	Pin related to CAN	Pin related to I2S1/2
208	V3	VINI[7]	-	-	RI1[7]	GPIO_PD[5]	-	-	GPIO_PD[5]	-	-	-	GPIO_PD[5]	-	-
19	W1	VINI[6]	-	-	RI1[6]	GPIO_PD[4]	-	-	GPIO_PD[4]	-	-	-	GPIO_PD[4]	-	-
118	W2	VINI[5]	-	-	RI1[5]	-	CAN_TX0	-	-	CAN_TX0	-	-	-	CAN_TX0	-
209	W3	VINI[4]	-	-	RI1[4]	-	CAN_RX0	-	-	CAN_RX0	-	-	-	CAN_RX0	-
292	W4	VINI[3]	-	-	RI1[3]	-	CAN_TX1	-	-	CAN_TX1	-	-	-	CAN_TX1	-
119	Y2	VINI[2]	-	-	RI1[2]	-	CAN_RX1	-	-	CAN_RX1	-	-	-	CAN_RX1	-
210	Y3	VINI[1]	-	-	GI1[7]	-	-	I2S_SCK1	-	-	I2S_SCK1	-	-	-	I2S_SCK1
293	Y4	VINI[0]	-	-	GI1[6]	-	-	I2S_WS1	-	-	I2S_WS1	-	-	-	I2S_WS1
211	AA3	VINVSYNC1	-	-	VINVSYNC1	-	-	I2S_ECLK1	-	-	I2S_ECLK1	-	-	-	I2S_ECLK1
294	AA4	VINHSYNC1	-	-	VINHSYNC1	-	-	I2S_SDI1	-	-	I2S_SDI1	-	-	-	I2S_SDI1
22	AB1	VINFID1	-	-	VINFID1	-	-	I2S_SDO1	-	-	I2S_SDO1	-	-	-	I2S_SDO1
202	M3	VINVSYNC0	-	-	GI1[5]	-	-	-	-	-	-	-	-	-	-
203	N3	VINHSYNC0	-	-	GI1[4]	-	-	-	-	-	-	-	-	-	-
112	N2	VINFID0	-	-	GI1[3]	-	-	-	-	-	-	-	-	-	-
123	AD2	-	PWM_O0	-	GI1[2]	GPIO_PD[3]	-	-	GPIO_PD[3]	-	-	-	GPIO_PD[3]	-	-
122	AC2	-	PWM_O1	-	BI1[7]	GPIO_PD[2]	-	-	GPIO_PD[2]	-	-	-	GPIO_PD[2]	-	-
121	AB2	-	-	I2S_SDO2	BI1[6]	-	-	I2S_SDO2	-	-	-	SPI_DO	GPIO_PD[1]	-	-
24	AD1	-	-	I2S_ECLK2	BI1[5]	-	-	I2S_ECLK2	-	-	-	Reserved (入出力)	GPIO_PD[0]	-	-
23	AC1	-	-	I2S_SCK2	BI1[4]	-	-	I2S_SCK2	-	-	-	SPI_SCK	-	-	I2S_SCK2
295	AB4	-	-	I2S_WS2	BI1[3]	-	-	I2S_WS2	-	-	-	SPI_SS	-	-	I2S_WS2
212	AB3	-	-	I2S_SDI2	BI1[2]	-	-	I2S_SDI2	-	-	-	SPI_DI	-	-	I2S_SDI2

Pin multiplex group #2 mode setting

This mode is set with MPX_MODE_2 bit (bit 2-0) in the Multiplex mode setting register (CMUX_MD.)

Table 1-6 Mode setting of pin multiplex group #2

MPX_MODE_2 (bit 2-0) of the CMUX_MD register	Pin multiplex group #2 mode
000	Mode 0
001	Mode 1
010	Mode 2
011	Mode 3
100	Mode 4
101 – 110	Reserved
111	(Initial value)

Pin multiplex group #4 (setting register: PIN_MPX_Select.MPX_MODE_4 [1:0])

Table 1-7 Pin function of pin multiplex group #4 by mode

Pin No.	JEDEC	Mode 1			
		Pin related to I2S1	Pin related to CAN	Pin related to GPIO	Pin related to PWM
28	AF3	I2S_SDI1	-	-	-
125	AE3	I2S_WS1	-	-	-
215	AD4	I2S_ECLK1	-	-	-
296	AC4	I2S_SDO1	-	-	-
214	AD3	I2S_SCK1	-	-	-
297	AC5	-	CAN_TX0	-	-
216	AD5	-	CAN_RX0	-	-
127	AE5	-	CAN_TX1	-	-
30	AF5	-	CAN_RX1	-	-
298	AC6	-	-	GPIO_PD[23]	-
217	AD6	-	-	GPIO_PD[22]	-
128	AE6	-	-	GPIO_PD[21]	-
31	AF6	-	-	GPIO_PD[20]	-
299	AC7	-	-	GPIO_PD[19]	-
218	AD7	-	-	GPIO_PD[18]	-
129	AE7	-	-	GPIO_PD[17]	-
32	AF7	-	-	GPIO_PD[16]	-
300	AC8	-	-	GPIO_PD[15]	-
219	AD8	-	-	GPIO_PD[14]	-
130	AE8	-	-	GPIO_PD[13]	-
220	AD9	-	-	-	PWM_O1
131	AE9	-	-	-	PWM_O0

Pin multiplex group #4 mode setting

This mode is set with MPX_MODE_4 bit (bit 5-4) in the Multiplex mode setting register (CMUX_MD.)

Table 1-8 Mode setting of pin multiplex group #4

MPX_MODE_4 (Bit 5-4) of the CMUX_MD register	Pin multiplex group #4 mode
00	Reserved
01	Mode 1
10	Reserved
11	(Initial value)

Pin multiplex group #5 (setting pin: MPX_MODE_5 [1:0])

Table 1-9 Pin function of pin multiplex group #5 by mode

Pin No.	JEDEC	Mode 0	Mode 1	Mode 2	
		Pin related to ETM	Pin related to UART3/4/5	Pin related to UART3/4	Pin related to PWM
270	C10	TRACECLK	UART_SIN3	UART_SIN3	-
185	B10	TRACECTL	UART_SOUT3	UART_SOUT3	-
92	A10	TRACEDATA[3]	UART_SIN4	UART_SIN4	-
346	D11	TRACEDATA[2]	UART_SOUT4	UART_SOUT4	-
269	C11	TRACEDATA[1]	UART_SIN5	-	PWM_O1
184	B11	TRACEDATA[0]	UART_SOUT5	-	PWM_O0

Pin multiplex group #5 mode setting

This mode is set with external pin, MPX_MODE_5[1:0].

Table 1-10 Mode setting of pin multiplex group #5

MPX_MODE_5[1] pin	MPX_MODE_5[0] pin	Pin multiplex group #5 mode
"L"	"L"	Mode 0
"L"	"H"	Mode 1
"H"	"L"	Mode 2
"H"	"H"	Mode 0

1.6.2. Pin function

Format

Pin function list is shown in the following format.

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description

Meaning of item and sign

Pin name

Name of external pin.

I/O

Input/Output signal's distinction based on this LSI.

- I: Pin that can be used as input
- O: Pin that can be used as output
- IO: Pin that can be used as input and output (interactive pin)

Polarity

Active polarity of external pin's input/output signals

- P: "H" active pin (positive logic)
- N: "L" active pin (negative logic)
- PN: "H" and "L" active pins

Analog/Digital

Signal type of external pin

- A: Analog signal
- D: Digital signal

Type

Input/Output circuit type of external pin.

- CLK: Clock
- POD: Pseudo Open Drain
- PU: Pull Up
- PD: Pull Down
- ST: Schmitt Type
- Tri: Tri-state

Pin status after reset

Pin status after external pin reset

- H: "H" level
- L: "L" level
- HiZ: High impedance
- X: "H" level or "L" level
- A: Clock output

Description

Outline of external pin function

1.6.3. External bus interface related pin

Table 1-11 External bus interface related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
MEM_XCS[4]	O	N	D	-	H	Chip select 4
MEM_XCS[2]	O	N	D	-	H	Chip select 2
MEM_XCS[0]	O	N	D	-	H	Chip select 0
MEM_XRD	O	N	D	-	H	Read strobe
MEM_XWR[3:2]	O	N	D	-	H	Write strobe MEM_XWR[3] → MEM_ED[31:24], MEM_XWR[2] → MEM_ED[23:16] (optional pin)
MEM_XWR[1:0]	O	N	D	-	H	Write strobe MEM_XWR[1] → MEM_ED[15:8] MEM_XWR[0] → MEM_ED[7:0]
MEM_RDY	I	P	D	-	-	Ready input for slow device
MEM_EA[24:1]	O	-	D	-	L	Address bus
MEM_ED[31:16]	IO	-	D	-	HiZ	Bi-directional data bus (optional pin)
MEM_ED[15:0]	IO	-	D	-	HiZ	Bi-directional data bus
DREQ[7:6]	I	-	D	-	-	External DMA request
XDACK[7:6]	O	P	D	-	L	External DMA acknowledge

1.6.4. SD memory controller related pin

Table 1-12 SD memory controller related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
SD_CLK	O	N	D	-	L	Media clock
SD_CMD	IO	-	D	-	HiZ	Media command
SD_DAT[3:0]	IO	-	D	-	HiZ	Media data
SD_WP	I	P	D	-	-	Media write protection
SD_XMCD	I	N	D	-	-	Media card detection

1.6.5. External interrupt controller related pin

Table 1-13 External interrupt controller related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
INT_A[3:0]	I	PN	D	-	-	Asynchronous external interrupt requests

1.6.6. UART related pin

Table 1-14 UART related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
UART_SIN0	I	P	D	-	-	Input data signal
UART_SOUT0	O	P	D	-	H	Output data signal
UART_XCTS0	I	N	D	-	-	Clear to send
UART_XRTS0	O	N	D	-	H	Request to send
UART_SIN1	I	P	D	-	-	Input data signal
UART_SOUT1	O	P	D	-	H	Output data signal
UART_SIN2	I	P	D	-	-	Input data signal
UART_SOUT2	O	P	D	-	H	Output data signal
UART_SIN3	I	P	D	-	-	Input data signal (optional)
UART_SOUT3	O	P	D	-	H	Output data signal (optional)
UART_SIN4	I	P	D	-	-	Input data signal (optional)
UART_SOUT4	O	P	D	-	H	Output data signal (optional)
UART_SIN5	I	P	D	-	-	Input data signal (optional)
UART_SOUT5	O	P	D	-	H	Output data signal (optional)

1.6.7. CAN related pin

Table 1-15 CAN related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
CAN_TX0	O	-	D	PD	H	Transmission (optional)
CAN_RX0	I	-	D	PD	-	Reception (optional)
CAN_TX1	O	-	D	PD	H	Transmission (optional)
CAN_RX1	I	-	D	PD	-	Reception (optional)

1.6.8. I2S related pin

Table 1-16 I2S related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
I2S_ECLK0	I	-	D	-	-	External clock (optional)
I2S_SCK0	IO	-	D	-	HiZ	Clock (optional)
I2S_WS0	IO	PN	D	-	HiZ	Sync (optional)
I2S_SDI0	I	P	D	-	-	Input data signal (optional)
I2S_SDO0	O	P	D	-	Hiz	Output data signal (optional)
I2S_ECLK1	I	-	D	-	-	External clock (optional)
I2S_SCK1	IO	-	D	PD	L	Clock (optional)
I2S_WS1	IO	PN	D	PD	L	Sync(optional)
I2S_SDI1	I	P	D	-	-	Input data signal (optional)
I2S_SDO1	O	P	D	PD	L	Output data signal (optional)
I2S_ECLK2	I	-	D	PD	-	External clock (optional)
I2S_SCK2	IO	-	D	PD	L	Clock (optional)
I2S_WS2	IO	PN	D	PD	L	Sync (optional)
I2S_SDI2	I	P	D	-	-	Input data signal (optional)
I2S_SDO2	O	P	D	PD	L	Output data signal (optional)

1.6.9. I^{2}C related pin

Table 1-17 I^{2}C related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
I2C_SCL0	IO	-	D	POD	HiZ	I2C clock
I2C_SDA0	IO	-	D	POD	HiZ	I2C data
I2C_SCL1	IO	-	D	POD	HiZ	I2C clock
I2C_SDA1	IO	-	D	POD	HiZ	I2C data

1.6.10. SPI related pin

Table 1-18 SPI related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
SPI_DO	O	P	D	PD	L	Serial data output (optional)
SPI_DI	I	P	D	-	-	Serial data input (optional)
SPI_SCK	O	-	D	PD	L	Serial clock (optional)
SPI_SS	O	PN	D	PD	L	Slave select (optional)

1.6.11. PWM related pin

Table 1-19 PWMrelated pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
PWM_O0	O	-	D	PD (*1)	L	PWM out 0 (optional)
PWM_O1	O	-	D	PD (*1)	L	PWM out 1 (optional)

*1: Only PWM pin of the pin multiplex group #2 is with pull-down resistance

1.6.12. A/D converter related pin

Table 1-20 A/D converter related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
AD_VIN0	I	-	A	-	-	A/D analog input
AD_VRH0	I	-	A	-	-	Reference voltage "H" input
AD_VRL0	I	-	A	-	-	Reference voltage "L" input
AD_AVD	I	-	A	-	-	Analog power supply
AD_VR0	O	-	A	-	-	Reference output
AD_VIN1	I	-	A	-	-	A/D analog input
AD_VRH1	I	-	A	-	-	Reference voltage "H" input
AD_VRL1	I	-	A	-	-	Reference voltage "L" input
AD_AVS	I	-	A	-	-	Analog ground
AD_VR1	O	-	A	-	-	Reference output

1.6.13. DDR2 related pin

Table 1-21 DDR2 related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
MA[13:0]	O	P	D	-	H	Address
MBA[1:0]	O	P	D	-	H	Bank address
MDQ[31:0]	IO	P	D	-	H	Data (*5)
MDM[3:0]	O	P	D	-	HiZ	Data mask (*6)
MDQSP[3:0]	IO	P	D	-	HiZ	Data strobe (*5)
MDQSN[3:0]	IO	N	D	-	HiZ	Data strobe (*5)
MCKP	O	P	D	CLK	L	Clock output
MCKN	O	N	D	CLK	H	Clock output
MCKE	O	P	D	-	L	Clock enable
MCS	O	N	D	-	L	Chip select
MRAS	O	N	D	-	H	Row address strobe
MCAS	O	N	D	-	H	Column address strobe
MWE	O	N	D	-	H	Write enable
DDRVDE	I	-	A	-	-	SSTL_18 1.8V power supply
VREF1	I	-	A	-	-	Reference voltage input (DDRVDE/2)
VREF0	I	-	A	-	-	Reference voltage input (DDRVDE/2)
OCD	I	-	A	-	-	Off chip driver reference voltage input (*1)
ODT	I	-	A	-	-	On-die termination reference voltage input (*2)
ODTCONT	O	P	D	-	L	On-die termination control (*3)
MCKE_START	I	P	D	-	-	Set a state of MCKE in reset 0: Low (*4) 1: High (reserved)
DDRTYPE	I	P	D	-	-	Pull up pin to VDDE via high resistance

*1: Pull up the pin to DDRVDE (1.8V power supply), via 200Ω resistance.
*2: PCB impedance Z = 100Ω or 50Ω: Pull up pin to DDRVDE (1.8V power supply), via 180Ω resistance.
PCB impedance Z = 150Ω or 75Ω: Pull up pin to DDRVDE (1.8V power supply), via 240Ω resistance.
*3: It connects it with the ODT pin of DDR2SDRAM.
*4: Pull down pin to VSS, via high resistance.
*5: This is process of unused pin at 16 bit mode. Pull down the pin to VSS via high resistance.
Unused pins at 16 bit mode are as follows:
"MDQ[31:16], MDQSP[3:2], MDQSN[3:2]"
*6: This is process of MDM[3:2] at 16 bit mode. Be sure to open this pin.

1.6.14. DISPLAY related pin

Table 1-22 DISPLAY related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Explanation
HSYNC0	IO	-	D	-	HiZ	Video output interface horizontal sync output Horizontal sync input in external sync mode
VSYNC0	IO	-	D	-	HiZ	Video output interface vertical sync output Vertical sync input in external sync mode
GV0	O	-	D	-	L	Video output interface graphics/video switch
DCLKIN0	I	-	D	CLK	-	Video output interface dot clock input
DCLKO0	O	-	D	CLK	X	Video output interface dot clock output
DE0	O	-	D	-	X	DE/CSYNC
DOUTR0[7:2]	O	-	D	-	X	Digital RGB output0 DataR[7:2]
DOUTR0[1:0]	O	-	D	-	X	Digital RGB output0 DataR[1:0] (optional)
DOUTG0[7:2]	O	-	D	-	X	Digital RGB output0 DataG[7:2]
DOUTG0[1:0]	O	-	D	-	X	Digital RGB output0 DataG[1:0] (optional)
DOUTB0[7:2]	O	-	D	-	X	Digital RGB output0 DataB[7:2]
DOUTB0[1:0]	O	-	D	-	X	Digital RGB output0 DataB[1:0] (optional)
HSYNC1	IO	-	D	-	HiZ	Video output interface horizontal sync output Horizontal sync input in external sync mode
VSYNC1	IO	-	D	-	HiZ	Video output interface vertical sync output Vertical sync input in external sync mode
GV1	O	-	D	-	L	Video output interface graphics/video switch
DCLKIN1	I	-	D	CLK	-	Video output interface dot clock input
DCLKO1	O	-	D	CLK	X	Video output interface dot clock output
DE1	O	-	D	-	X	DE/CSYNC
DOUTR1[7:2]	O	-	D	-	X	Digital RGB output1 DataR[7:2]
DOUTG1[7:2]	O	-	D	-	X	Digital RGB output1 DataG[7:2]
DOUTB1[7:2]	O	-	D	-	X	Digital RGB output1 DataB[7:2]

1.6.15. Video capture related pin

Table 1-23 Video capture related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
VIN0[7:0]	I	-	D	-	-	Video capture Data[7:0]
VINVSYNC0	I	-	D	PD	-	Video capture vertical sync input
VINHSYNC0	I	-	D	PD	-	Video capture horizontal sync input
VINFID0	I	-	D	-	-	Video input field identification signal 0 in odd field
CCLK0	I	-	D	CLK	-	Video capture input clock
VIN1[7:0]	I	-	D	PD	-	Video capture Data[7:0]
VINVSYNC1	I	-	D	-	-	Video capture vertical sync input
VINHSYNC1	I	-	D	-	-	Video capture horizontal sync input
VINFID1	I	-	D	PD	-	Video input field identification signal 0 in odd field
CCLK1	I	-	D	CLK	-	Video capture input clock
RI1[7:2]	I	-	D	PD	-	NRGB666 capture DataR[7:2] (optional)
GI1[7:2]	I	-	D	PD (*1)	-	NRGB666 capture DataG[7:2] (optional)
BI1[7:2]	I	-	D	PD (*2)	-	NRGB666 capture DataB[7:2] (optional)

*1: GI1[3] is not applicable.
*2: BI1[2] is not applicable.

1.6.16. System related pin

Table 1-24 System related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
CLK	I	-	D	CLK	-	Input clock
XRST	I	N	D	ST	-	System reset
CRIPM[3:0]	I	-	D	-	-	PLLMODE setting
VINITHI	I	-	D	-	-	Boot high address
PLLBYPASS	I	-	D	-	-	PLL bypass mode setting
BIGEND	I	-	D	-	-	LSI endian setting Low: Little endian High: Big endian
PLLVSS	I	-	A	-	-	PLL ground
PLLTDTRST	I	-	D	-	-	Test pin Pull up the pin to VDDE, via high resistance
PLLVDD	I	-	A	-	-	PLL power supply

1.6.17. JTAG related pin

Table 1-25 JTAG related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
TCK	I	-	D	ST, PU	-	Test clock
XTRST	I	N	D	ST, PU	-	Test reset
TMS	I	N	D	PU	-	Test mode
TDI	I	-	D	PU	-	Test data input
TDO	O	-	D	Tri	HiZ	Test data output

1.6.18. ICE related pin

Table 1-26 ICE related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
RTCK	O	-	D	-	H	Return test clock
XSRST	IO	N	D	ST, PU	H	System reset

1.6.19. Multiplex setting related pin

Table 1-27 Multiplex setting related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
JTAGSEL	I	-	D	-	-	JTAG selection 1: DFT, 0: Normal Pull it down to VSS, via high resistance
MPX_MODE_5[1:0]	I	-	D	-	-	External pin multiplex mode 5
MPX_MODE_1[1:0]	I	-	D	-	-	External pin multiplex mode 1
TESTMODE[2:0]	I	-	D	-	-	Test mode selection pin Pull it down to VSS, via high resistance
VPD	I	-	D	-	-	Test mode selection pin Pull it down to VSS, via high resistance

1.6.20. ETM related pin

Table 1-28 ETM related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
TRACECLK	O	-	D	-	L	Exported clock for TRACEDATA[3:0] and TRACECTL They are valid on bath edges of TRACECLK for max. integrity.
TRACECTL	O	-	D	-	H	Trace control signal used by the trace tool such as RealView supplied by ARM Limited.
TRACEDATA[3:0]	O	-	D	-	LHHH	Trace data used by the trace tool such as RealView supplied by ARM Limited.

1.6.21. Power supply related pin

Table 1-29 Power supply related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
VSS	I	-	D	-	-	Ground
VDDE	I	-	D	-	-	External pin power supply
VDDI	I	-	D	-	-	Internal power supply

1.6.22. GPIO related pin

Table 1-30 GPIO related pin's function

Pin name	I/O	Polarity	Analog /Digital	Type	Status of pin after reset	Description
GPIO_PD[23:0]	IO	-	D	PD (*1)	HiZ	General purpose I/O port (optional)

*1: GPIO_PD[12:6] is not applicable.

1.6.23. Unused pin

Proceed following processes for unused pin.

Table 1-31 MB86R03 unused pin's process

Pin No.	JEDEC	Pin name	Process
3	C1	DOUTB1[2], MEM_XWR[2], DOUTB0[0]	Pull up to VDDE or pull down to VSS through high resistance.
4	D1	DOUTB1[6], MEM_ED[18], DOUTR0[0]	Pull up to VDDE or pull down to VSS through high resistance.
5	E1	DOUTG1[4], MEM_ED[22], GPIO_PD[8]	Pull up to VDDE or pull down to VSS through high resistance.
6	F1	DOUTR1[2], MEM_ED[26], GPIO_PD[12]	Pull up to VDDE or pull down to VSS through high resistance.
7	G1	DCLKIN1	Pull up to VDDE or pull down to VSS through high resistance.
9	J1	DCLKO1	Keep the pin open.
10	K1	VIN0[5]	Pull up to VDDE or pull down to VSS through high resistance.
11	L1	VIN0[1]	Pull up to VDDE or pull down to VSS through high resistance.
12	M1	CCLK0	Pull up to VDDE or pull down to VSS through high resistance.
14	P1	VSS	Connect to VSS.
15	R1	VSS	Connect to VSS or Pull down to VSS through 10kΩ resistance. (*2)
16	T1	VSS	Connect to VSS or Pull down to VSS through 10kΩ resistance. (*2)
17	U1	VSS	Connect to VSS.
18	V1	VSS	Connect to VSS.
19	W1	VIN1[6], RI1[6], GPIO_PD[4]	Keep the pin open.
21	AA1	CCLK1	Pull up to VDDE or pull down to VSS through high resistance.
22	AB1	VINFID1, I2S_SDO1	Keep the pin open.
23	AC1	I2S_SCK2, BI1[4], SPI_SCK	Keep the pin open.
24	AD1	I2S_ECLK2, BI1[5], Reserved (input/output), GPIO_PD[0]	Keep the pin open.
28	AF3	I2S_SDI1	Pull up to VDDE or pull down to VSS through high resistance.
29	AF4	(Unused)	Keep the pin open.
30	AF5	CAN_RX1	Keep the pin open.
31	AF6	GPIO_PD[20]	Keep the pin open.
32	AF7	GPIO_PD[16]	Keep the pin open.
33	AF8	(Unused)	Keep the pin open.
34	AF9	(Unused)	Keep the pin open.
35	AF10	(Unused)	Keep the pin open.
36	AF11	MPX_MODE_5[1]	Pull up to VDDE or pull down to VSS through high resistance.
38	AF13	AD_AVD	Connect to VSS.
39	AF14	AD_AVS	Connect to VSS.
40	AF15	UART_SOUT0	Keep the pin open.
41	AF16	UART_SIN0	Pull up to VDDE or pull down to VSS through high resistance.
42	AF17	UART_SIN1	Pull up to VDDE or pull down to VSS through high resistance.
43	AF18	SD_DAT[0]	Pull up to VDDE or pull down to VSS through high resistance.
44	AF19	SD_WP	Pull up to VDDE or pull down to VSS through high resistance.
45	AF20	I2C_SCL1	Pull up to VDDE or pull down to VSS through high resistance.

Pin No.	JEDEC	Pin name	Process
46	AF21	I2C_SDA1	Pull up to VDDE or pull down to VSS through high resistance.
47	AF22	INT_A[0]	Pull up to VDDE or pull down to VSS through high resistance.
48	AF23	MA[8]	Keep the pin open.
49	AF24	MA[12]	Keep the pin open.
52	AE26	MA[7]	Keep the pin open.
53	AD26	MA[3]	Keep the pin open.
54	AC26	MA[1]	Keep the pin open.
55	AB26	MBA[1]	Keep the pin open.
57	Y26	MDQSN[0]	Pull down to VSS through high resistance.
58	W26	MDQSP[0]	Pull down to VSS through high resistance.
60	U26	MDQSN[1]	Pull down to VSS through high resistance.
61	T26	MDQSP[1]	Pull down to VSS through high resistance.
63	P26	MCKN	Keep the pin open.
64	N26	MCKP	Keep the pin open.
66	L26	MDQSN[2]	Pull down to VSS through high resistance.
67	K26	MDQSP[2]	Pull down to VSS through high resistance.
69	H26	MDQSN[3]	Pull down to VSS through high resistance.
70	G26	MDQSP[3]	Pull down to VSS through high resistance.
72	E26	MEM_ED[3]	Pull up to VDDE or pull down to VSS through high resistance.
73	D26	MEM_ED[7]	Pull up to VDDE or pull down to VSS through high resistance.
74	C26	MEM_ED[11]	Pull up to VDDE or pull down to VSS through high resistance.
78	A24	MEM_EA[1]	Pull up to VDDE or pull down to VSS through high resistance.
79	A23	MEM_EA[4]	Pull up to VDDE or pull down to VSS through high resistance.
80	A22	MEM_EA[8]	Pull up to VDDE or pull down to VSS through high resistance.
81	A21	MEM_EA[12]	Pull up to VDDE or pull down to VSS through high resistance.
82	A20	MEM_EA[16]	Pull up to VDDE or pull down to VSS through high resistance.
83	A19	MEM_EA[20]	Pull up to VDDE or pull down to VSS through high resistance.
85	A17	MEM_XRD	Pull up to VDDE or pull down to VSS through high resistance.
88	A14	TDO	Keep the pin open.
92	A10	TRACEDATA[3], UART_SIN4	Pull up to VDDE or pull down to VSS through high resistance.
94	A8	DOUTB0[4]	Keep the pin open.
95	A7	DOUTG0[2]	Keep the pin open.
96	A6	DOUTG0[6]	Keep the pin open.
97	A5	DCLKIN0	Pull up to VDDE or pull down to VSS through high resistance.
99	A3	DCLKO0	Keep the pin open.
101	B2	DE0	Keep the pin open.
102	C2	GV0	Keep the pin open.
103	D2	DOUTB1[5], MEM_ED[17], DOUTG0[1]	Pull up to VDDE or pull down to VSS through high resistance.
104	E2	DOUTG1[3], MEM_ED[21], GPIO_PD[7]	Pull up to VDDE or pull down to VSS through high resistance.
105	F2	DOUTG1[7], MEM_ED[25], GPIO_PD[11]	Pull up to VDDE or pull down to VSS through high resistance.
106	G2	DOUTR1[5], MEM_ED[29], I2S_WS0	Pull up to VDDE or pull down to VSS through high resistance.
108	J2	GV1, DREQ[7]	Pull up to VDDE or pull down to VSS through high resistance.

Pin No.	JEDEC	Pin name	Process
109	K2	VIN0[6]	Pull up to VDDE or pull down to VSS through high resistance.
110	L2	VIN0[2]	Pull up to VDDE or pull down to VSS through high resistance.
112	N2	VINFID0, GI1[3]	Pull up to VDDE or pull down to VSS through high resistance.
113	P2	VSS or VDDI	Connect to VSS or VDDI. (*1) Do not open the pin.
114	R2	VSS	Connect to VSS or Pull down to VSS through 10kΩ resistance. (*2)
115	T2	VSS	Connect to VSS or Pull down to VSS through 10kΩ resistance. (*2)
116	U2	VSS	Connect to VSS.
117	V2	VSS	Connect to VSS.
118	W2	VIN1[5], RI1[5], CAN_TX0	Keep the pin open.
119	Y2	VIN1[2], RI1[2], CAN_RX1	Keep the pin open.
121	AB2	I2S_SDO2, BI1[6], SPI_DO, GPIO_PD[1]	Keep the pin open.
122	AC2	PWM_O1, BI1[7], GPIO_PD[2]	Keep the pin open.
123	AD2	PWM_O0, GI1[2], GPIO_PD[3]	Keep the pin open.
125	AE3	I2S_WS1	Keep the pin open.
126	AE4	(Unused)	Keep the pin open.
127	AE5	CAN_TX1	Keep the pin open.
128	AE6	GPIO_PD[21]	Keep the pin open.
129	AE7	GPIO_PD[17]	Keep the pin open.
130	AE8	GPIO_PD[13]	Keep the pin open.
131	AE9	PWM_O0	Keep the pin open.
132	AE10	(Unused)	Keep the pin open.
133	AE11	MPX_MODE_5[0]	Pull up to VDDE or pull down to VSS through high resistance.
135	AE13	AD_VRH0	Connect to VSS.
136	AE14	AD_VRH1	Connect to VSS.
137	AE15	UART_XRTS0	Keep the pin open.
138	AE16	UART_XCTS0	Pull up to VDDE or pull down to VSS through high resistance.
139	AE17	UART_SOUT1	Keep the pin open.
140	AE18	SD_DAT[1]	Pull up to VDDE or pull down to VSS through high resistance.
141	AE19	SD_XMCD	Pull up to VDDE or pull down to VSS through high resistance.
142	AE20	I2C_SCL0	Pull up to VDDE or pull down to VSS through high resistance.
143	AE21	INT_A[3]	Pull up to VDDE or pull down to VSS through high resistance.
144	AE22	MCKE_START	Pull down to VSS through high resistance.
145	AE23	MA[13]	Keep the pin open.
146	AE24	MA[4]	Keep the pin open.
147	AE25	MA[11]	Keep the pin open.
148	AD25	MA[5]	Keep the pin open.
149	AC25	MA[10]	Keep the pin open.
150	AB25	MBA[0]	Keep the pin open.
151	AA25	MCKE	Keep the pin open.
152	Y25	MDQ[2]	Pull down to VSS through high resistance.
153	W25	MDQ[0]	Pull down to VSS through high resistance.

Pin No.	JEDEC	Pin name	Process
154	V25	VREF0	Connect to DDRVDE/2[V]Reference voltage.
155	U25	MDQ[13]	Pull down to VSS through high resistance.
156	T25	MDQ[8]	Pull down to VSS through high resistance.
157	R25	MDQ[15]	Pull down to VSS through high resistance.
160	M25	MDQ[21]	Pull down to VSS through high resistance.
161	L25	MDQ[16]	Pull down to VSS through high resistance.
162	K25	VREF1	Connect to DDRVDE/2[V]Reference voltage.
163	J25	MDQ[29]	Pull down to VSS through high resistance.
164	H25	MDQ[24]	Pull down to VSS through high resistance.
165	G25	MDQ[31]	Pull down to VSS through high resistance.
166	F25	MEM_ED[0]	Pull up to VDDE or pull down to VSS through high resistance.
167	E25	MEM_ED[4]	Pull up to VDDE or pull down to VSS through high resistance.
168	D25	MEM_ED[8]	Pull up to VDDE or pull down to VSS through high resistance.
169	C25	MEM_ED[12]	Pull up to VDDE or pull down to VSS through high resistance.
170	B25	MEM_ED[14]	Pull up to VDDE or pull down to VSS through high resistance.
171	B24	MEM_ED[15]	Pull up to VDDE or pull down to VSS through high resistance.
172	B23	MEM_EA[3]	Pull up to VDDE or pull down to VSS through high resistance.
173	B22	MEM_EA[7]	Pull up to VDDE or pull down to VSS through high resistance.
174	B21	MEM_EA[11]	Pull up to VDDE or pull down to VSS through high resistance.
175	B20	MEM_EA[15]	Pull up to VDDE or pull down to VSS through high resistance.
176	B19	MEM_EA[19]	Pull up to VDDE or pull down to VSS through high resistance.
177	B18	MEM_EA[23]	Pull up to VDDE or pull down to VSS through high resistance.
178	B17	MEM_XWR[1]	Pull up to VDDE or pull down to VSS through high resistance.
179	B16	MEM_XCS[4]	Pull up to VDDE or pull down to VSS through high resistance.
183	B12	TMS	Pull up to VDDE or pull down to VSS through high resistance.
184	B11	TRACEDATA[0], UART_SOUT5, PWM_O0	Pull up to VDDE or pull down to VSS through high resistance.
185	B10	TRACECTL, UART_SOUT3	Keep the pin open.
187	B8	DOUTB0[5]	Keep the pin open.
188	B7	DOUTG0[3]	Keep the pin open.
189	B6	DOUTG0[7]	Keep the pin open.
190	B5	DOUTR0[4]	Keep the pin open.
192	B3	HSYNC0	Pull up to VDDE or pull down to VSS through high resistance.
193	C3	VSYNC0	Pull up to VDDE or pull down to VSS through high resistance.
194	D3	DOUTB1[4], MEM_ED[16], DOUTG0[0]	Pull up to VDDE or pull down to VSS through high resistance.
195	E3	DOUTG1[2], MEM_ED[20], GPIO_PD[6]	Pull up to VDDE or pull down to VSS through high resistance.
196	F3	DOUTG1[6], MEM_ED[24], GPIO_PD[10]	Pull up to VDDE or pull down to VSS through high resistance.
197	G3	DOUTR1[4], MEM_ED[28], I2S_SDI0	Pull up to VDDE or pull down to VSS through high resistance.
198	H3	DOUTR1[7], MEM_ED[31], I2S_ECLK0	Pull up to VDDE or pull down to VSS through high resistance.
199	J3	VSYNC1, XDACK[6]	Pull up to VDDE or pull down to VSS through high resistance.
200	K3	VIN0[7]	Pull up to VDDE or pull down to VSS through high resistance.
201	L3	VIN0[3]	Pull up to VDDE or pull down to VSS through high resistance.
202	M3	VINVSYNC0, GI1[5]	Keep the pin open.

Pin No.	JEDEC	Pin name	Process
203	N3	VINHSYNC0, GI1[4]	Keep the pin open.
204	P3	VSS	Connect to VSS.
205	R3	VSS or VDDE	Connect to VSS or VDDE. (*1) Do not open the pin.
206	T3	VSS	Connect to VSS.
207	U3	VSS or VDDI	Connect to VSS or VDDI. (*1) Do not open the pin.
208	V3	VIN1[7], RI1[7], GPIO_PD[5]	Keep the pin open.
209	W3	VIN1[4], RI1[4], CAN_RX0	Keep the pin open.
210	Y3	VIN1[1], GI1[7], I2S_SCK1	Keep the pin open.
211	AA3	VINVSYNC1, I2S_ECLK1	Pull up to VDDE or pull down to VSS through high resistance.
212	AB3	I2S_SDI2, BI1[2], SPI_DI	Pull up to VDDE or pull down to VSS through high resistance.
213	AC3	(Unused)	Connect to VSS.
214	AD3	I2S_SCK1	Keep the pin open.
215	AD4	I2S_ECLK1	Pull up to VDDE or pull down to VSS through high resistance.
216	AD5	CAN_RX0	Keep the pin open.
217	AD6	GPIO_PD[22]	Keep the pin open.
218	AD7	GPIO_PD[18]	Keep the pin open.
219	AD8	GPIO_PD[14]	Keep the pin open.
220	AD9	PWM_O1	Keep the pin open.
221	AD10	(Unused)	Keep the pin open.
222	AD11	MPX_MODE_1[1]	Pull up to VDDE or pull down to VSS through high resistance.
224	AD13	AD_VIN0	Connect to VSS.
225	AD14	AD_VIN1	Connect to VSS.
227	AD16	UART_SOUT2	Keep the pin open.
228	AD17	SD_CMD	Pull up to VDDE or pull down to VSS through high resistance.
229	AD18	SD_DAT[2]	Pull up to VDDE or pull down to VSS through high resistance.
230	AD19	(Unused)	Keep the pin open.
231	AD20	I2C_SDA0	Pull up to VDDE or pull down to VSS through high resistance.
232	AD21	INT_A[1]	Pull up to VDDE or pull down to VSS through high resistance.
234	AD23	MA[9]	Keep the pin open.
235	AD24	MA[6]	Keep the pin open.
236	AC24	MA[2]	Keep the pin open.
237	AB24	MWE	Keep the pin open.
238	AA24	MRAS	Keep the pin open.
239	Y24	MDQ[5]	Pull down to VSS through high resistance.
240	W24	MDQ[1]	Pull down to VSS through high resistance.
241	V24	MDQ[7]	Pull down to VSS through high resistance.
242	U24	MDQ[10]	Pull down to VSS through high resistance.
243	T24	MDQ[9]	Pull down to VSS through high resistance.
244	R24	MDM[1]	Pull down to VSS through high resistance.
247	M24	MDQ[18]	Pull down to VSS through high resistance.
248	L24	MDQ[17]	Pull down to VSS through high resistance.

Pin No.	JEDEC	Pin name	Process
249	K24	MDQ[23]	Pull down to VSS through high resistance.
250	J24	MDQ[26]	Pull down to VSS through high resistance.
251	H24	MDQ[28]	Pull down to VSS through high resistance.
252	G24	MDM[3]	Pull down to VSS through high resistance.
253	F24	MEM_ED[1]	Pull up to VDDE or pull down to VSS through high resistance.
254	E24	MEM_ED[5]	Pull up to VDDE or pull down to VSS through high resistance.
255	D24	MEM_ED[9]	Pull up to VDDE or pull down to VSS through high resistance.
256	C24	MEM_ED[13]	Pull up to VDDE or pull down to VSS through high resistance.
257	C23	MEM_EA[2]	Pull up to VDDE or pull down to VSS through high resistance.
258	C22	MEM_EA[6]	Pull up to VDDE or pull down to VSS through high resistance.
259	C21	MEM_EA[10]	Pull up to VDDE or pull down to VSS through high resistance.
260	C20	MEM_EA[14]	Pull up to VDDE or pull down to VSS through high resistance.
261	C19	MEM_EA[18]	Pull up to VDDE or pull down to VSS through high resistance.
262	C18	MEM_EA[22]	Pull up to VDDE or pull down to VSS through high resistance.
263	C17	MEM_XWR[0]	Pull up to VDDE or pull down to VSS through high resistance.
264	C16	MEM_XCS[2]	Pull up to VDDE or pull down to VSS through high resistance.
267	C13	TCK	Pull up to VDDE or pull down to VSS through high resistance.
269	C11	TRACEDATA[1], UART_SIN5, PWM_O1	Pull up to VDDE or pull down to VSS through high resistance.
270	C10	TRACECLK, UART_SIN3	Pull up to VDDE or pull down to VSS through high resistance.
271	C9	DOUTB0[2]	Keep the pin open.
272	C8	DOUTB0[6]	Keep the pin open.
273	C7	DOUTG0[4]	Keep the pin open.
274	C6	DOUTR0[2]	Keep the pin open.
275	C5	DOUTR0[5]	Keep the pin open.
276	C4	DOUTR0[7]	Keep the pin open.
277	D4	DOUTB1[3], MEM_XWR[3], DOUTB0[1]	Pull up to VDDE or pull down to VSS through high resistance.
278	E4	DOUTB1[7], MEM_ED[19], DOUTR0[1]	Pull up to VDDE or pull down to VSS through high resistance.
279	F4	DOUTG1[5], MEM_ED[23], GPIO_PD[9]	Pull up to VDDE or pull down to VSS through high resistance.
280	G4	DOUTR1[3], MEM_ED[27], I2S_SDO0	Pull up to VDDE or pull down to VSS through high resistance.
281	H4	DOUTR1[6], MEM_ED[30], I2S_SCK0	Pull up to VDDE or pull down to VSS through high resistance.
282	J4	HSYNC1, DREQ[6]	Pull up to VDDE or pull down to VSS through high resistance.
283	K4	DE1, XDACK[7]	Keep the pin open.
284	L4	VIN0[4]	Pull up to VDDE or pull down to VSS through high resistance.
285	M4	VIN0[0]	Pull up to VDDE or pull down to VSS through high resistance.
287	P4	VSS	Connect to VSS.
288	R4	VSS	Connect to VSS.
289	T4	VSS	Connect to VSS.
292	W4	VIN1[3], RI1[3], CAN_TX1	Keep the pin open.
293	Y4	VIN1[0], GI1[6], I2S_WS1	Keep the pin open.
294	AA4	VINHSYNC1, I2S_SDI1	Pull up to VDDE or pull down to VSS through high resistance.
295	AB4	I2S_WS2, BI1[3], SPI_SS	Keep the pin open.
296	AC4	I2S_SDO1	Keep the pin open.

Pin No.	JEDEC	Pin name	Process
297	AC5	CAN_TX0	Keep the pin open.
298	AC6	GPIO_PD[23]	Keep the pin open.
299	AC7	GPIO_PD[19]	Keep the pin open.
300	AC8	GPIO_PD[15]	Keep the pin open.
301	AC9	(Unused)	Keep the pin open.
302	AC10	(Unused)	Keep the pin open.
303	AC11	MPX_MODE_1[0]	Pull up to VDDE or pull down to VSS through high resistance.
305	AC13	AD_VR0	Connect to VSS.
306	AC14	AD_VR1	Connect to VSS.
308	AC16	UART_SIN2	Pull up to VDDE or pull down to VSS through high resistance.
309	AC17	SD_CLK	Keep the pin open.
310	AC18	SD_DAT[3]	Pull up to VDDE or pull down to VSS through high resistance.
312	AC20	INT_A[2]	Pull up to VDDE or pull down to VSS through high resistance.
313	AC21	DDRTYPE	Pull up to VDDE through high resistance.
314	AC22	ODTCONT	Keep the pin open.
315	AC23	MA[0]	Keep the pin open.
316	AB23	MCS	Keep the pin open.
317	AA23	MCAS	Keep the pin open.
318	Y23	MDQ[3]	Pull down to VSS through high resistance.
319	W23	MDQ[4]	Pull down to VSS through high resistance.
320	V23	MDM[0]	Pull down to VSS through high resistance.
321	U23	MDQ[11]	Pull down to VSS through high resistance.
322	T23	MDQ[12]	Pull down to VSS through high resistance.
323	R23	MDQ[14]	Pull down to VSS through high resistance.
324	P23	OCD	Keep the pin open.
325	N23	ODT	Keep the pin open.
326	M23	MDQ[19]	Pull down to VSS through high resistance.
327	L23	MDQ[20]	Pull down to VSS through high resistance.
328	K23	MDM[2]	Pull down to VSS through high resistance.
329	J23	MDQ[27]	Pull down to VSS through high resistance.
330	H23	MDQ[25]	Pull down to VSS through high resistance.
331	G23	MDQ[30]	Pull down to VSS through high resistance.
332	F23	MEM_ED[2]	Pull up to VDDE or pull down to VSS through high resistance.
333	E23	MEM_ED[6]	Pull up to VDDE or pull down to VSS through high resistance.
334	D23	MEM_ED[10]	Pull up to VDDE or pull down to VSS through high resistance.
335	D22	MEM_EA[5]	Pull up to VDDE or pull down to VSS through high resistance.
336	D21	MEM_EA[9]	Pull up to VDDE or pull down to VSS through high resistance.
337	D20	MEM_EA[13]	Pull up to VDDE or pull down to VSS through high resistance.
338	D19	MEM_EA[17]	Pull up to VDDE or pull down to VSS through high resistance.
339	D18	MEM_EA[21]	Pull up to VDDE or pull down to VSS through high resistance.
340	D17	MEM_EA[24]	Pull up to VDDE or pull down to VSS through high resistance.
341	D16	MEM_XCS[0]	Pull up to VDDE or pull down to VSS through high resistance.

Pin No.	JEDEC	Pin name	Process
342	D15	MEM_RDY	Pull up to VDDE or pull down to VSS through high resistance.
344	D13	TDI	Pull up to VDDE or pull down to VSS through high resistance.
346	D11	TRACEDATA[2], UART_SOUT4	Pull up to VDDE or pull down to VSS through high resistance.
347	D10	RTCK	Keep the pin open.
348	D9	DOUTB0[3]	Keep the pin open.
349	D8	DOUTB0[7]	Keep the pin open.
350	D7	DOUTG0[5]	Keep the pin open.
351	D6	DOUTR0[3]	Keep the pin open.
352	D5	DOUTR0[6]	Keep the pin open.
362	P5	VSS or VDDE	Connect to VSS or VDDE. (*1) Do not open the pin.
363	R5	VSS	Connect to VSS or Pull down to VSS through 10kΩ resistance. (*2)
364	T5	VSS	Connect to VSS.
378	AB13	AD_VRL0	Connect to VSS.
379	AB14	AD_VRL1	Connect to VSS.
391	V22	MDQ[6]	Pull down to VSS through high resistance.
398	L22	MDQ[22]	Pull down to VSS through high resistance.

*1) If any of these pins is connected to VSS, P2, R3, U3 and P5 should be connected to VSS.
If any of these pins is connected to VDD, P2, R3, U3 and P5 should be connected to VDD.
*2) If P2, R3, U3 and P5 are connected to VDD, this pin should be pulled down to VSS through 10kΩ resistance.
If P2, R3, U3 and P5 are connected to VSS, this pin can be connected to VSS or 10kΩ pull-down resistance.

1.6.24. Unused pin in the duplex case with pin multiplex function

PWM, I2S1, and CAN pins may be duplicated and allocated to external pin depending on pin multiplex function's group combination. In this case, follow the procedure below.

Table 1-32 Unused pin process in the duplex case with pin multiplex function

Pin No.	JEDEC	Pin multiplex group: pin name	Process
122	AC2	Pin multiplex group #2:PWM_O1	Keep the pin open.
123	AD2	Pin multiplex group #2:PWM_O0	Keep the pin open.
220	AD9	Pin multiplex group #4:PWM_O1	Keep the pin open.
131	AE9	Pin multiplex group #4:PWM_O0	Keep the pin open.
269	C11	Pin multiplex group #5:PWM_O1	Pull down to VSS through high resistance.
184	B11	Pin multiplex group #5:PWM_O0	Pull down to VSS through high resistance.
118	W2	Pin multiplex group #2:CAN_TX0	Keep the pin open.
292	W4	Pin multiplex group #2:CAN_TX1	Keep the pin open.
209	W3	Pin multiplex group #2:CAN_RX0	Keep the pin open.
119	Y2	Pin multiplex group #2:CAN_RX1	Keep the pin open.
297	AC5	Pin multiplex group #4:CAN_TX0	Keep the pin open.
127	AE5	Pin multiplex group #4:CAN_TX1	Keep the pin open.
216	AD5	Pin multiplex group #4:CAN_RX0	Keep the pin open.
30	AF5	Pin multiplex group #4:CAN_RX1	Keep the pin open.
210	Y3	Pin multiplex group #2:I2S_SCK1	Keep the pin open.
293	Y4	Pin multiplex group #2:I2S_WS1	Keep the pin open.
211	AA3	Pin multiplex group #2:I2S_ECLK1	Pull down to VSS through high resistance.
294	AA4	Pin multiplex group #2:I2S_SDI1	Pull down to VSS through high resistance.
22	AB1	Pin multiplex group #2:I2S_SDO1	Keep the pin open.
28	AF3	Pin multiplex group #4:I2S_SDI1	Pull down to VSS through high resistance.
125	AE3	Pin multiplex group #4:I2S_WS1	Keep the pin open.
215	AD4	Pin multiplex group #4:I2S_ECLK1	Pull down to VSS through high resistance.
214	AD3	Pin multiplex group #4:I2S_SCK1	Keep the pin open.
296	AC4	Pin multiplex group #4:I2S_SDO1	Keep the pin open.

2. System configuration

Figure 2-1 shows system configuration for which this LSI is used to in-vehicle navigation.

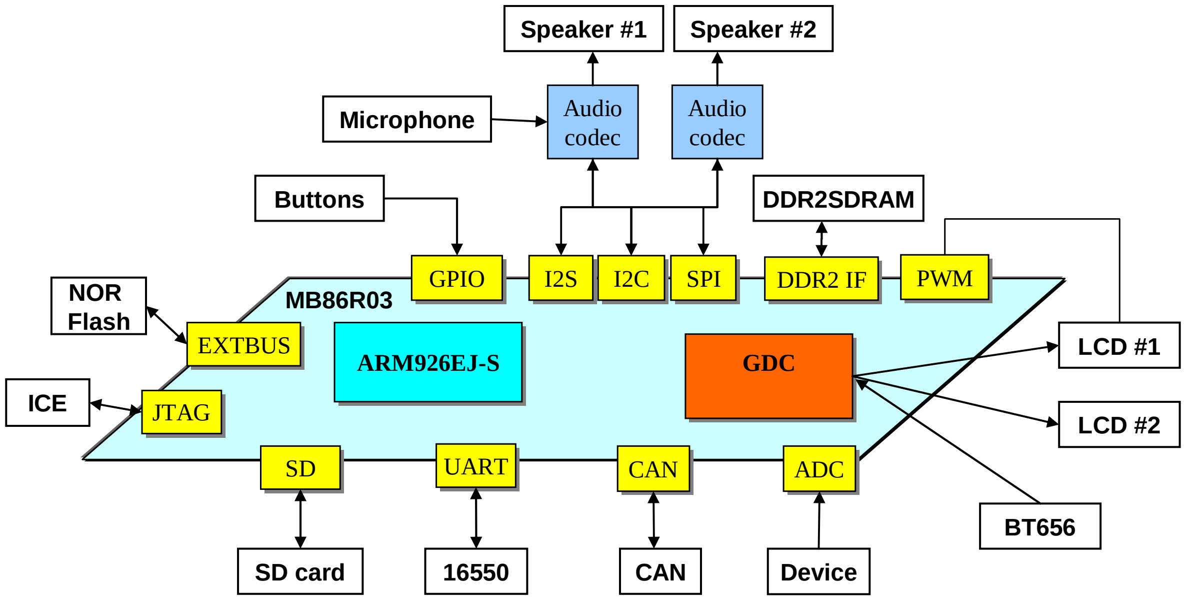


Figure 2-1 Sample of MB86R03 system configuration

3. Memory map

This chapter shows memory map and register map of MB86R03.

3.1. Memory map of LSI

Figure 3-1 shows MB86R03 memory map.
As the memory map indicates, boot operation jumps to user code, external boot ROM (1000_0000$_H$) through built-in boot ROM (0000_0000$_{H.}$) (Setting 1000_0000$_H$ to program counter (PC).)
After the jump, set remap boot controller to remap internal boot ROM area (0000_0000$_H$ ~ 0000_8000$_H$) to internal SRAM_0, then proceed interrupt vector area setting and each register setting.

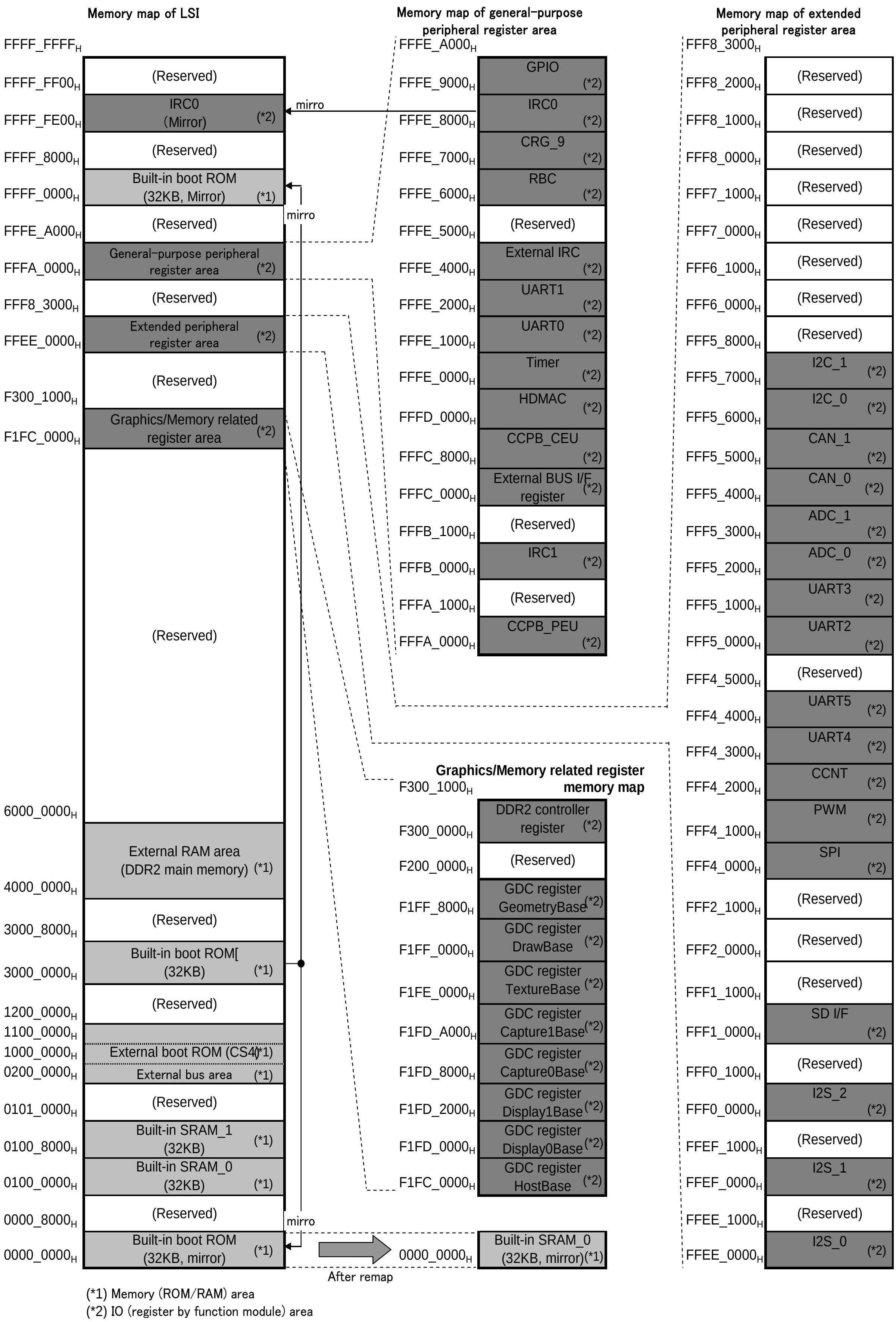


Figure 3-1 Memory map

3.2. Register access

Basically, register in MB86R03 should be accessed by word length except some registers.
Table 3-1 shows valid access data length of each register.

Table 3-1 Valid access data length of register

Module	Register name	Valid data length
DMAC	DMACR	Byte (8 bit) Address follows endian
	DMACA, DMACB, DMACSA, DMACDA	Word (32 bit)/Half-word (16 bit)/Byte (8 bit)
UART	RFR, TFR, DLL	Word (32 bit)/Byte (8 bit) When these registers are accessed by byte long, address follows endian
GPIO	PDR0, PDR1, PDR2	Word (32 bit)/Byte (8 bit). When these registers are accessed by byte long, address follows endian
DDR2 controller	All registers of DDR2 controller	Half-word (16 bit) Address follows endian
SDMC	All registers of SDMC	Byte (8 bit) Address follows endian
Others	All registers other than the above	Word (32 bit)

3.3. Register map

Table 3-2 MB86R03 register map

Module name	Address	Register name	Explanation
GDC	F1FC_0000$_H$ - F1FF_FFFF$_H$	Refer another document, "MB86R03 'Jade-L' LSI product specifications graphics display controller (GDC)" for GDC register	
No module	F200_0000$_H$ - F2FF_FFFF$_H$	Reserved	Access prohibited
DDR2 controller	F300_0000$_H$	DRIC	Initialization control register
	F300_0002$_H$	DRIC1	Initialization control command register 1
	F300_0004$_H$	DRIC2	Initialization control command register 2
	F300_0006$_H$	DRCA	Address control register
	F300_0008$_H$	DRCM	Mode control register
	F300_000A$_H$	DRCST1	Timing setting register 1
	F300_000C$_H$	DRCST2	Timing setting register 2
	F300_000E$_H$	DRCR	Refresh control register
	F300_0010$_H$ - F300_001F$_H$	Reserved	Access prohibited
	F300_0020$_H$	DRCF	FIFO control register
	F300_0022$_H$ - F300_002F$_H$	Reserved	Access prohibited
	F300_0030$_H$	DRASR	AXI operation setting register
	F300_0032$_H$ - F300_004F$_H$	Reserved	Access prohibited
	F300_0050$_H$	DRIMSD	IF setting register
	F300_0052$_H$ - F300_005F$_H$	Reserved	Access prohibited
	F300_0060$_H$	DROS	ODT setting register
	F300_0062$_H$	Reserved	Access prohibited
	F300_0064$_H$	DRIBSODT1	IO ODT1 setting register
	F300_0066$_H$	DRIBSOCD	IO OCD setting register
	F300_0068$_H$	DRIBSOCD2	IO OCD2 setting register
	F300_006A$_H$ - F300_006F$_H$	Reserved	Access prohibited
	F300_0070$_H$	DROABA	ODT bias auto adjustment register
	F300_0072$_H$ - F300_0083$_H$	Reserved	Access prohibited
	F300_0084$_H$	DROBS	ODT bias selection register
	F300_0086$_H$ - F300_008F$_H$	Reserved	Access prohibited
	F300_0090$_H$	DRIMR1	IO monitor register 1
	F300_0092$_H$	DRIMR2	IO monitor register 2
	F300_0094$_H$	DRIMR3	IO monitor register 3
	F300_0096$_H$	DRIMR4	IO monitor register 4
	F300_0098$_H$	DROISR1	OCD impedance setting register 1
	F300_009A$_H$	DROISR2	OCD impedance setting register 2
	F300_009C$_H$ - F300_0FFF$_H$	Reserved	Access prohibited
No module	F300_1000$_H$ - FFED_FFFF$_H$	Reserved	Access prohibited
I2S_0	FFEE_0000$_H$	I2S0RXFDAT	I2S_0 reception FIFO data register
	FFEE_0004$_H$	I2S0TXFDAT	I2S_0 transmission FIFO data register

Module name	Address	Register name	Explanation
I2S_0	$FFEE\_0008_H$	I2S0CNTREG	I2S_0 control register
	$FFEE\_000C_H$	I2S0MCR0REG	I2S_0 channel control register 0
	$FFEE\_0010_H$	I2S0MCR1REG	I2S_0 channel control register 1
	$FFEE\_0014_H$	I2S0MCR2REG	I2S_0 channel control register 2
	$FFEE\_0018_H$	I2S0OPRREG	I2S_0 operation control register
	$FFEE\_001C_H$	I2S0SRST	I2S_0 software reset register
	$FFEE\_0020_H$	I2S0INTCNT	I2S_0 interrupt control register
	$FFEE\_0024_H$	I2S0STATUS	I2S_0 status register
	$FFEE\_0028_H$	I2S0DMAACT	I2S_0 DMA start register
	$FFEE\_002C_H$ - $FFEE\_0FFF_H$	Reserved	Access prohibited
No module	$FFEE\_1000_H$ - $FFEE\_FFFF_H$	Reserved	Access prohibited
I2S_1	$FFEF\_0000_H$	I2S1RXFDAT	I2S_1 reception FIFO data register
	$FFEF\_0004_H$	I2S1TXFDAT	I2S_1 transmission FIFO data register
	$FFEF\_0008_H$	I2S1CNTREG	I2S_1 control register
	$FFEF\_000C_H$	I2S1MCR0REG	I2S_1 channel control register 0
	$FFEF\_0010_H$	I2S1MCR1REG	I2S_1 channel control register 1
	$FFEF\_0014_H$	I2S1MCR2REG	I2S_1 channel control register 2
	$FFEF\_0018_H$	I2S1OPRREG	I2S_1 operation control register
	$FFEF\_001C_H$	I2S1SRST	I2S_1 software reset register
	$FFEF\_0020_H$	I2S1INTCNT	I2S_1 interrupt control register
	$FFEF\_0024_H$	I2S1STATUS	I2S_1 status register
	$FFEF\_0028_H$	I2S1DMAACT	I2S_1 DMA start register
	$FFEF\_002C_H$ - $FFEF\_0FFF_H$	Reserved	Access prohibited
No module	$FFEF\_1000_H$ - $FFEF\_FFFF_H$	Reserved	Access prohibited
I2S_2	$FFF0\_0000_H$	I2S2RXFDAT	I2S_2 reception FIFO data register
	$FFF0\_0004_H$	I2S2TXFDAT	I2S_2 transmission FIFO data register
	$FFF0\_0008_H$	I2S2CNTREG	I2S_2 control register
	$FFF0\_000C_H$	I2S2MCR0REG	I2S_2 channel control register 0
	$FFF0\_0010_H$	I2S2MCR1REG	I2S_2 channel control register 1
	$FFF0\_0014_H$	I2S2MCR2REG	I2S_2 channel control register 2
	$FFF0\_0018_H$	I2S2OPRREG	I2S_2 operation control register
	$FFF0\_001C_H$	I2S2SRST	I2S_2 software reset register
	$FFF0\_0020_H$	I2S2INTCNT	I2S_2 interrupt control register
	$FFF0\_0024_H$	I2S2STATUS	I2S_2 status register
	$FFF0\_0028_H$	I2S2DMAACT	I2S_2 DMA start register
	$FFF0\_002C_H$ - $FFF0\_0FFF_H$	Reserved	Access prohibited
No module	$FFF0\_1000_H$ - $FFF0\_FFFF_H$	Reserved	Access prohibited
SDMC	$FFF1\_0000_H$ - $FFF1\_0FFF_H$	Another specifications	Another specifications
No module	$FFF1\_1000_H$ - $FFF1\_FFFF_H$	Reserved	Access prohibited
No module	$FFF2\_0000_H$ - $FFF2\_0FFF_H$	Reserved	Access prohibited
No module	$FFF2\_1000_H$ - $FFF3\_FFFF_H$	Reserved	Access prohibited
SPI	$FFF4\_0000_H$	SPICR	SPI control register

Module name	Address	Register name	Explanation
SPI	FFF4_0004$_H$	SPISCR	SPI slave control register
	FFF4_0008$_H$	SPIDR	SPI data register
	FFF4_000C$_H$	SPISR	SPI status register
	FFF4_0010$_H$ - FFF4_0FFF$_H$	Reserved	Access prohibited
PWM	FFF4_1000$_H$	PWM0BCR	PWM ch0 base clock register
	FFF4_1004$_H$	PWM0TPR	PWM ch0 pulse width register
	FFF4_1008$_H$	PWM0PR	PWM ch0 phase register
	FFF4_100C$_H$	PWM0DR	PWM ch0 duty register
	FFF4_1010$_H$	PWM0CR	PWM ch0 status register
	FFF4_1014$_H$	PWM0SR	PWM ch0 start register
	FFF4_1018$_H$	PWM0CCR	PWM ch0 current count register
	FFF4_101C$_H$	PWM0IR	PWM ch0 interrupt register
	FFF4_1020$_H$ - FFF4_10FF$_H$	Reserved	Access prohibited
	FFF4_1100$_H$	PWM1BCR	PWM ch1 base clock register
	FFF4_1104$_H$	PWM1TPR	PWM ch1 pulse width register
	FFF4_1108$_H$	PWM1PR	PWM ch1 phase register
	FFF4_110C$_H$	PWM1DR	PWM ch1 duty register
	FFF4_1110$_H$	PWM1CR	PWM ch1 status register
	FFF4_1114$_H$	PWM1SR	PWM ch1 start register
	FFF4_1118$_H$	PWM1CCR	PWM ch1 current count register
	FFF4_111C$_H$	PWM1IR	PWM ch1 interrupt register
	FFF4_1120$_H$ - FFF4_1FFF$_H$	Reserved	Access prohibited
CCNT	FFF4_2000$_H$	CCID	Chip ID register
	FFF4_2004$_H$	CSRST	Software reset register
	FFF4_2008$_H$ - FFF4_200F$_H$	Reserved	Access prohibited
	FFF4_2010$_H$	CIST	Interrupt status register
	FFF4_2014$_H$	CISTM	Interrupt status mask register
	FFF4_2018$_H$	CGPIO_IST	GPIO interrupt status register
	FFF4_201C$_H$	CGPIO_ISTM	GPIO interrupt status mask register
	FFF4_2020$_H$	CGPIO_IP	GPIO interrupt polarity setting register
	FFF4_2024$_H$	CGPIO_IM	GPIO interrupt mode setting register
	FFF4_2028$_H$	CAXI_BW	AXI bus wait cycle setting register
	FFF4_202C$_H$	CAXI_PS	AXI priority setting register
	FFF4_2030$_H$	CMUX_MD	Multiplex mode setting register
	FFF4_2024$_H$	CEX_PIN_ST	External pin status register
	FFF4_2038$_H$	Reserved	Access prohibited
	FFF4_203C$_H$	Reserved	Access prohibited
	FFF4_2040$_H$	Reserved	Access prohibited
	FFF4_2044$_H$ - FFF4_20E7$_H$	Reserved	Access prohibited
	FFF4_20E8$_H$	CBSC	Byte swap switching register
	FFF4_20EC$_H$	CDCRC	DDR2 controller reset control register
	FFF4_20F0$_H$	CMSR0	Software reset register 0 for macro
	FFF4_20F4$_H$	CMSR1	Software reset register 1 for macro
	FFF4_20F8$_H$ - FFF4_2FFF$_H$	Reserved	Access prohibited

Module name	Address	Register name	Explanation
UART4	FFF4_3000$_H$	URT4RFR	Transmission FIFO register (read only at DLAB = 0) When it accesses RFR by byte long in the big endian mode, address becomes FFF4_3003$_H$.
		URT4TFR	Transmission FIFO register (write only at DLAB = 0) When it accesses TFR by byte long in the big endian mode, address becomes FFF4_3003$_H$.
		URT4DLL	Dividing frequency value (lower byte at DLAB = 1) When it accesses DLL by byte long in the big endian mode, address becomes FFF4_3003$_H$.
	FFF4_3004$_H$	URT4IER	DLAB = 0: Interrupt enable register
		URT4DLM	DLAB = 1: Dividing frequency value (upper byte)
	FFF4_3008$_H$	URT4IIR	Interrupt ID register (read only)
		URT4FCR	FIFO control register (write only)
	FFF4_300C$_H$	URT4LCR	Line control register
	FFF4_3010$_H$	URT4MCR	Modem control register
	FFF4_3014$_H$	URT4LSR	Line status register
	FFF4_3018$_H$	URT4MSR	Modem status register
	FFF4_301C$_H$ - FFF4_3FFF$_H$	Reserved	Access prohibited
UART5	FFF4_4000$_H$	URT5RFR	Transmission FIFO register (read only at DLAB = 0) When it accesses RFR by byte long in the big endian mode, address becomes FFF4_4003H.
		URT5TFR	Transmission FIFO register (write only at DLAB = 0) When it accesses TFR by byte long in the big endian mode, address becomes FFF4_4003$_H$.
		URT5DLL	Dividing frequency value (lower byte at DLAB = 1) When it accesses DLL by byte long in the big endian mode, address becomes FFF4_4003$_H$.
	FFF4_4004$_H$	URT5IER	DLAB = 0: Interrupt enable register.
		URT5DLM	DLAB = 1: Dividing frequency value (upper byte)
	FFF4_4008$_H$	URT5IIR	Interrupt ID register (read only)
		URT5FCR	FIFO control register (write only)
	FFF4_400C$_H$	URT5LCR	Line control register
	FFF4_4010$_H$	URT5MCR	Modem control register
	FFF4_4014$_H$	URT5LSR	Line status register
	FFF4_4018$_H$	URT5MSR	Modem status register
	FFF4_401C$_H$ - FFF4_4FFF$_H$	Reserved	Access prohibited
No module	FFF4_5000$_H$ - FFF4_FFFF$_H$	Reserved	Access prohibited
UART2	FFF5_0000$_H$	URT2RFR	Transmission FIFO register (read only at DLAB = 0) When it accesses RFR by byte long in the big endian mode, address becomes FFF5_0003$_H$.
		URT2TFR	Transmission FIFO register (write only at DLAB = 0) When it accesses TFR by byte long in the big endian mode, address becomes FFF5_0003$_H$.
		URT2DLL	Dividing frequency value (lower byte at DLAB = 1) When it accesses DLL by byte long in the big endian mode, address becomes FFF5_0003$_H$.
	FFF5_0004$_H$	URT2IER	DLAB = 0: Interrupt enable register.
		URT2DLM	DLAB = 1: Dividing frequency value (upper byte)
	FFF5_0008$_H$	URT2IIR	Interrupt ID register (read only)
		URT2FCR	FIFO control register (write only)
	FFF5_000C$_H$	URT2LCR	Line control register
	FFF5_0010$_H$	URT2MCR	Modem control register

Module name	Address	Register name	Explanation
UART2	FFF5_0014$_H$	URT2LSR	Line status register
	FFF5_0018$_H$	URT2MSR	Modem status register
	FFF5_001C$_H$ - FFF5_0FFF$_H$	Reserved	Access prohibited
UART3	FFF5_1000$_H$	URT3RFR	Transmission FIFO register (read only at DLAB = 0) When it accesses RFR by byte long in the big endian mode, address becomes FFF5_1003$_H$.
		URT3TFR	Transmission FIFO register (write only at DLAB = 0) When it accesses TFR by byte long in the big endian mode, address becomes FFF5_1003$_H$.
		URT3DLL	Dividing frequency value (lower byte at DLAB = 1) When it accesses DLL by byte long in the big endian mode, address becomes FFF5_1003$_H$.
	FFF5_1004$_H$	URT3IER	DLAB = 0: Interrupt enable register.
		URT3DLM	DLAB = 1: Dividing frequency value (upper byte)
	FFF5_1008$_H$	URT3IIR	Interrupt ID register (read only)
		URT3FCR	FIFO control register (write only)
	FFF5_100C$_H$	URT3LCR	Line control register
	FFF5_1010$_H$	URT3MCR	Modem control register
	FFF5_1014$_H$	URT3LSR	Line status register
	FFF5_1018$_H$	URT3MSR	Modem status register
	FFF5_101C$_H$ - FFF5_1FFF$_H$	Reserved	Access prohibited
ADC_0	FFF5_2000$_H$	ADC0DATA	Data register
	FFF5_2004$_H$	Reserved	Access prohibited
	FFF5_2008$_H$	ADC0XPD	Power down control register
	FFF5_200C$_H$	Reserved	Access prohibited
	FFF5_2010$_H$	ADC0CKSEL	Clock selection register
	FFF5_2014$_H$	ADC0STATUS	Status register
	FFF5_2018$_H$ - FFF5_2FFF$_H$	Reserved	Access prohibited
ADC_1	FFF5_3000$_H$	ADC1DATA	Data register
	FFF5_3004$_H$	Reserved	Access prohibited
	FFF5_3008$_H$	ADC1XPD	Power down control register
	FFF5_300C$_H$	Reserved	Access prohibited
	FFF5_3010$_H$	ADC1CKSEL	Clock selection register
	FFF5_3014$_H$	ADC1STATUS	Status register
	FFF5_3018$_H$ - FFF5_3FFF$_H$	Reserved	Access prohibited
CAN_0	FFF5_4000$_H$ - FFF5_4FFF$_H$	Another specifications	Another specifications
CAN_1	FFF5_5000$_H$ - FFF5_5FFF$_H$	Another specifications	Another specifications
I^2C_0	FFF5_6000$_H$	I2C0BSR	I2C bus status register ch0
	FFF5_6004$_H$	I2C0BCR	I2C bus control register ch0
	FFF5_6008$_H$	I2C0CCR	I2C clock control register ch0
	FFF5_600C$_H$	I2C0ADR	I2C address register ch0
	FFF5_6010$_H$	I2C0DAR	I2C data register ch0
	FFF5_6014$_H$	I2C0ECSR	I2C extension CS register ch0
	FFF5_6018$_H$	I2C0BCFR	I2C bus clock frequency register ch0
	FFF5_601C$_H$	I2C0BC2R	I2C bus control 2 registers ch0
	FFF5_6020$_H$ - FFF5_6FFF$_H$	Reserved	Access prohibited

Module name	Address	Register name	Explanation
I^2C_1	FFF5_7000$_H$	I2C1BSR	I2C bus status register ch1
	FFF5_7004$_H$	I2C1BCR	I2C bus control register ch1
	FFF5_7008$_H$	I2C1CCR	I2C clock control register ch1
	FFF5_700C$_H$	I2C1ADR	I2C address register ch1
	FFF5_7010$_H$	I2C1DAR	I2C data register ch1
	FFF5_7014$_H$	I2C1ECSR	I2C extension CS register ch1
	FFF5_7018$_H$	I2C1BCFR	I2C bus clock frequency register ch1
	FFF5_701C$_H$	I2C1BC2R	I2C bus control 2 registers ch1
	FFF5_7020$_H$ - FFF5_7FFF$_H$	Reserved	Access prohibited
No module	FFF5_8000$_H$ - FFF5_FFFF$_H$	Reserved	Access prohibited
No module	FFF6_0000$_H$ - FFF6_0FFF$_H$	Reserved	Access prohibited
No module	FFF6_1000$_H$ - FFF6_FFFF$_H$	Reserved	Access prohibited
No module	FFF7_0000$_H$ - FFF7_0FFF$_H$	Reserved	Access prohibited
No module	FFF7_1000$_H$ - FFF7_FFFF$_H$	Reserved	Access prohibited
No module	FFF8_0000$_H$ - FFF8_1FFF$_H$	Reserved	Access prohibited
No module	FFF8_2000$_H$ - FFF8_2FFF$_H$	Reserved	Access prohibited
No module	FFF8_3000$_H$ - FFF9_FFFF$_H$	Reserved	Access prohibited
CCPB_PEU	FFFA_0000$_H$ - FFFA_0FFF$_H$	Another specifications	Another specifications
No module	FFFA_1000$_H$ - FFFA_FFFF$_H$	Reserved	Access prohibited
Interrupt controller 1 (IRC1)	FFFB_0000$_H$	IR1IRQF	IRQ flag register
	FFFB_0004$_H$	IR1IRQM	IRQ mask register
	FFFB_0008$_H$	IR1ILM	Interrupt level mask register
	FFFB_000C$_H$	IR1ICRMN	ICR monitoring register
	FFFB_0010$_H$ - FFFB_0018$_H$	Reserved	Access prohibited
	FFFB_001C$_H$	IR1TBR	Table base register
	FFFB_0020$_H$	IR1VCT	Interrupt vector register
	FFFB_0024$_H$ - FFFB_002C$_H$	Reserved	Access prohibited
	FFFB_0030$_H$	IR1ICR0	Interrupt control register 00
	FFFB_0034$_H$	IR1ICR1	Interrupt control register 01
	FFFB_0038$_H$	IR1ICR2	Interrupt control register 02
	FFFB_003C$_H$	IR1ICR3	Interrupt control register 03
	FFFB_0040$_H$	IR1ICR4	Interrupt control register 04
	FFFB_0044$_H$	IR1ICR5	Interrupt control register 05
	FFFB_0048$_H$	IR1ICR6	Interrupt control register 06
	FFFB_004C$_H$	IR1ICR7	Interrupt control register 07
	FFFB_0050$_H$	IR1ICR8	Interrupt control register 08
	FFFB_0054$_H$	IR1ICR9	Interrupt control register 09
	FFFB_0058$_H$	IR1ICR10	Interrupt control register 10
	FFFB_005C$_H$	IR1ICR11	Interrupt control register 11
	FFFB_0060$_H$	IR1ICR12	Interrupt control register 12

Module name	Address	Register name	Explanation
Interrupt controller 1 (IRC1)	FFFB_0064$_H$	IR1ICR13	Interrupt control register 13
	FFFB_0068$_H$	IR1ICR14	Interrupt control register 14
	FFFB_006C$_H$	IR1ICR15	Interrupt control register 15
	FFFB_0070$_H$	IR1ICR16	Interrupt control register 16
	FFFB_0074$_H$	IR1ICR17	Interrupt control register 17
	FFFB_0078$_H$	IR1ICR18	Interrupt control register 18
	FFFB_007C$_H$	IR1ICR19	Interrupt control register 19
	FFFB_0080$_H$	IR1ICR20	Interrupt control register 20
	FFFB_0084$_H$	IR1ICR21	Interrupt control register 21
	FFFB_0088$_H$	IR1ICR22	Interrupt control register 22
	FFFB_008C$_H$	IR1ICR23	Interrupt control register 23
	FFFB_0090$_H$	IR1ICR24	Interrupt control register 24
	FFFB_0094$_H$	IR1ICR25	Interrupt control register 25
	FFFB_0098$_H$	IR1ICR26	Interrupt control register 26
	FFFB_009C$_H$	IR1ICR27	Interrupt control register 27
	FFFB_00A0$_H$	IR1ICR28	Interrupt control register 28
	FFFB_00A4$_H$	IR1ICR29	Interrupt control register 29
	FFFB_00A8$_H$	IR1ICR30	Interrupt control register 30
	FFFB_00AC$_H$	IR1ICR31	Interrupt control register 31
	FFFB_00B0$_H$ - FFFB_FFFF$_H$	Reserved	Access prohibited
External bus interface (External BUS I/F)	FFFC_0000$_H$	MCFMODE0	SRAM/Flash-mode register 0
	FFFC_0004$_H$	MCFMODE1	SRAM/Flash-mode register 1 (access prohibited)
	FFFC_0008$_H$	MCFMODE2	SRAM/Flash-mode register 2
	FFFC_000C$_H$	MCFMODE3	SRAM/Flash-mode register 3 (access prohibited)
	FFFC_0010$_H$	MCFMODE4	SRAM/Flash-mode register 4
	FFFC_0014$_H$	MCFMODE5	SRAM/Flash-mode register 5 (access prohibited)
	FFFC_0018$_H$	MCFMODE6	SRAM/Flash-mode register 6 (access prohibited)
	FFFC_001C$_H$	MCFMODE7	SRAM/Flash-mode register 7 (access prohibited)
	FFFC_0020$_H$	MCFTIM0	SRAM/Flash timing register 0
	FFFC_0024$_H$	MCFTIM1	SRAM/Flash timing register 1 (access prohibited)
	FFFC_0028$_H$	MCFTIM2	SRAM/Flash timing register 2
	FFFC_002C$_H$	MCFTIM3	SRAM/Flash timing register 3 (access prohibited)
	FFFC_0030$_H$	MCFTIM4	SRAM/Flash timing register 4
	FFFC_0034$_H$	MCFTIM5	SRAM/Flash timing register 5 (access prohibited)
	FFFC_0038$_H$	MCFTIM6	SRAM/Flash timing register 6 (access prohibited)
	FFFC_003C$_H$	MCFTIM7	SRAM/Flash timing register 7 (access prohibited)
	FFFC_0040$_H$	MCFAREA0	SRAM/Flash area register 0
	FFFC_0044$_H$	MCFAREA1	SRAM/Flash area register 1
	FFFC_0048$_H$	MCFAREA2	SRAM/Flash area register 2
	FFFC_004C$_H$	MCFAREA3	SRAM/Flash area register 3
	FFFC_0050$_H$	MCFAREA4	SRAM/Flash area register 4
	FFFC_0054$_H$	MCFAREA5	SRAM/Flash area register 5
	FFFC_0058$_H$	MCFAREA6	SRAM/Flash area register 6
	FFFC_005C$_H$	MCFAREA7	SRAM/Flash area register 7

Module name	Address	Register name	Explanation
External bus interface (External BUS I/F)	FFFC_0060$_H$ - FFFC_01FC$_H$	Reserved	Access prohibited
	FFFC_0200$_H$	MCERR	Memory controller error register
	FFFC_0204$_H$ - FFFC_7FFF$_H$	Reserved	Access prohibited
CCPB_CEU	FFFC_8000$_H$ - FFFC_FFFF$_H$	Another specifications	Another specifications
DMAC	FFFD_0000$_H$	DMACR	DMAC configuration register
	FFFD_0004$_H$ - FFFD_000F$_H$	Reserved	Access prohibited
	FFFD_0010$_H$	DMACA0	DMAC0 configuration A register
	FFFD_0014$_H$	DMACB0	DMAC0 configuration B register
	FFFD_0018$_H$	DMACSA0	DMAC0 source address register
	FFFD_001C$_H$	DMACDA0	DMAC0 destination address register
	FFFD_0020$_H$	DMACA1	DMAC1 configuration A register
	FFFD_0024$_H$	DMACB1	DMAC1 configuration B register
	FFFD_0028$_H$	DMACSA1	DMAC1 source address register
	FFFD_002C$_H$	DMACDA1	DMAC1 destination address register
	FFFD_0030$_H$	DMACA2	DMAC2 configuration A register
	FFFD_0034$_H$	DMACB2	DMAC2 configuration B register
	FFFD_0038$_H$	DMACSA2	DMAC2 source address register
	FFFD_003C$_H$	DMACDA2	DMAC2 destination address register
	FFFD_0040$_H$	DMACA3	DMAC3 configuration A register
	FFFD_0044$_H$	DMACB3	DMAC3 configuration B register
	FFFD_0048$_H$	DMACSA3	DMAC3 source address register
	FFFD_004C$_H$	DMACDA3	DMAC3 destination address register
	FFFD_0050$_H$	DMACA4	DMAC4 configuration A register
	FFFD_0054$_H$	DMACB4	DMAC4 configuration B register
	FFFD_0058$_H$	DMACSA4	DMAC4 source address register
	FFFD_005C$_H$	DMACDA4	DMAC4 destination address register
	FFFD_0060$_H$	DMACA5	DMAC5 configuration A register
	FFFD_0064$_H$	DMACB5	DMAC5 configuration B register
	FFFD_0068$_H$	DMACSA5	DMAC5 source address register
	FFFD_006C$_H$	DMACDA5	DMAC5 destination address register
	FFFD_0070$_H$	DMACA6	DMAC6 configuration A register
	FFFD_0074$_H$	DMACB6	DMAC6 configuration B register
	FFFD_0078$_H$	DMACSA6	DMAC6 source address register
	FFFD_007C$_H$	DMACDA6	DMAC6 Destination address register
	FFFD_0080$_H$	DMACA7	DMAC7 configuration A register
	FFFD_0084$_H$	DMACB7	DMAC7 configuration B register
	FFFD_0088$_H$	DMACSA7	DMAC7 source address register
	FFFD_008C$_H$	DMACDA7	DMAC7 destination address register
	FFFD_0090$_H$ - FFFD_FFFF$_H$	Reserved	Access prohibited
Timer	FFFE_0000$_H$	TMR0LD	Timer 1 load value
	FFFE_0004$_H$	TMR0VAL	Timer 1 current value
	FFFE_0008$_H$	TMR0CTL	Timer 1 control register
	FFFE_000C$_H$	TMR0IC	Timer 1 interrupt clear register
	FFFE_0010$_H$	TMR0RIS	Timer 1 interrupt status

Module name	Address	Register name	Explanation
Timer	FFFE_0014$_H$	TMR0MIS	Interrupt status to which Timer 1 masks
	FFFE_0018$_H$	TMR0BGL	Timer 1 background load value
	FFFE_001C$_H$	Reserved	Access prohibited
	FFFE_0020$_H$	TMR1LD	Timer 2 load value
	FFFE_0024$_H$	TMR1VAL	Timer 2 current value
	FFFE_0028$_H$	TMR1CTL	Timer 2 control registers
	FFFE_002C$_H$	TMR1IC	Timer 2 interrupt clear register
	FFFE_0030$_H$	TMR1RIS	Timer 2 interrupt status
	FFFE_0034$_H$	TMR1MIS	Interrupt status to which Timer 2 masks
	FFFE_0038$_H$	TMR1BGL	Timer 2 background load value
	FFFE_003C$_H$ - FFFE_0FFF$_H$	Reserved	Access prohibited
UART0	FFFE_1000$_H$	URT0RFR	Reception FIFO register (read only at DLAB = 0) When it accesses RFR by byte long in the big endian mode, address becomes FFFE_1003$_H$.
		URT0TFR	Transmission FIFO register (write only at DLAB = 0) When it accesses TFR by byte long in the big endian mode, address becomes FFFE_1003$_H$.
		URT0DLL	Dividing frequency value (lower byte at DLAB = 1) When it accesses DLL by byte long in the big endian mode, address becomes FFFE_1003$_H$.
	FFFE_1004$_H$	URT0IER	DLAB = 0: Interrupt enable register
		URT0DLM	DLAB = 1: Dividing frequency value (upper byte)
	FFFE_1008$_H$	URT0IIR	Interrupt ID register (read only)
		URT0FCR	FIFO control register (write only)
	FFFE_100C$_H$	URT0LCR	Line control register
	FFFE_1010$_H$	URT0MCR	Modem control register
	FFFE_1014$_H$	URT0LSR	Line status register
	FFFE_1018$_H$	URT0MSR	Modem status register
	FFFE_101C$_H$ - FFFE_1FFF$_H$	Reserved	Access prohibited
UART1	FFFE_2000$_H$	URT1RFR	Transmission FIFO register (read only at DLAB = 0) When it accesses RFR by byte long in the big endian mode, address becomes FFFE_2003$_H$.
		URT1TFR	Transmission FIFO register (write only at DLAB = 0) When it accesses TFR by byte long in the big endian mode, address becomes FFFE_2003$_H$.
		URT1DLL	Dividing frequency value (lower byte at DLAB = 1) When it accesses DLL by byte long in the big endian mode, address becomes FFFE_2003$_H$.
	FFFE_2004$_H$	URT1IER	DLAB = 0: Interrupt enable register.
		URT1DLM	DLAB = 1: Dividing frequency value (upper byte)
	FFFE_2008$_H$	URT1IIR	Interrupt ID register (read only)
		URT1FCR	FIFO control register (write only)
	FFFE_200C$_H$	URT1LCR	Line control register
	FFFE_2010$_H$	URT1MCR	Modem control register
	FFFE_2014$_H$	URT1LSR	Line status register
	FFFE_2018$_H$	URT1MSR	Modem status register
	FFFE_201C$_H$ - FFFE_3FFF$_H$	Reserved	Access prohibited
External interrupt controller (EXIRC)	FFFE_4000$_H$	EIENB	External interrupt enable register
	FFFE_4004$_H$	EIREQ	External interrupt request register

Module name	Address	Register name	Explanation
External interrupt controller (EXIRC)	FFFE_4008$_H$	EILVL	External interrupt level register
	FFFE_401C$_H$ - FFFE_47FF$_H$	Reserved	Access prohibited
No module	FFFE_4800$_H$ - FFFE_5FFF$_H$	Reserved	Access prohibited
Remap boot controller (RBC)	FFFE_6000$_H$	Reserved	Access prohibited
	FFFE_6004$_H$	RBREMAP	Remap control register
	FFFE_6008$_H$	RBVIHA	VINITHI control register A
	FFFE_600C$_H$	RBITRA	INITRAM control register A
	FFFE_6010$_H$ - FFFE_6FFF$_H$	Reserved	Access prohibited
Clock reset generator (CRG)	FFFE_7000$_H$	CRPR	PLL control register
	FFFE_7004$_H$	Reserved	Access prohibited
	FFFE_7008$_H$	CRWR	Watchdog timer control register
	FFFE_700C$_H$	CRSR	Reset/Standby control register
	FFFE_7010$_H$	CRDA	Clock division control register A
	FFFE_7014$_H$	CRDB	Clock division control register B
	FFFE_7018$_H$	CRHA	AHB(A) bus clock gate control register
	FFFE_701C$_H$	CRPA	APB(A) bus clock gate control register
	FFFE_7020$_H$	CRPB	APB(B) bus clock gate control register
	FFFE_7024$_H$	CRHB	AHB(B) bus clock gate control register
	FFFE_7028$_H$	CRAM	ARM core clock gate control register
	FFFE_702C$_H$ - FFFE_7FFF$_H$	Reserved	Access prohibited
Interrupt controller 0 (IRC0)	FFFE_8000$_H$	IR0IRQF	IRQ flag register
	FFFE_8004$_H$	IR0IRQM	IRQ mask register
	FFFE_8008$_H$	IR0ILM	Interrupt level mask register
	FFFE_800C$_H$	IR0ICRMN	ICR monitoring register
	FFFE_8010$_H$	Reserved	Access prohibited
	FFFE_8014$_H$	IR0SWIR0	Software interrupt control register 0
	FFFE_8018$_H$	IR0SWIR1	Software interrupt control register 1
	FFFE_801C$_H$	IR0TBR	Table base register
	FFFE_8020$_H$	IR0VCT	Interrupt vector register
	FFFE_8024$_H$	Reserved	Access prohibited
	FFFE_8028$_H$	Reserved	Access prohibited
	FFFE_802C$_H$	Reserved	Access prohibited
	FFFE_8030$_H$	IR0ICR0	Interrupt control register 00
	FFFE_8034$_H$	IR0ICR1	Interrupt control register 01
	FFFE_8038$_H$	IR0ICR2	Interrupt control register 02
	FFFE_803C$_H$	IR0ICR3	Interrupt control register 03
	FFFE_8040$_H$	IR0ICR4	Interrupt control register 04
	FFFE_8044$_H$	IR0ICR5	Interrupt control register 05
	FFFE_8048$_H$	IR0ICR6	Interrupt control register 06
	FFFE_804C$_H$	IR0ICR7	Interrupt control register 07
	FFFE_8050$_H$	IR0ICR8	Interrupt control register 08
	FFFE_8054$_H$	IR0ICR9	Interrupt control register 09

Module name	Address	Register name	Explanation
Interrupt controller 0 (IRC0)	FFFE_8058$_H$	IR0ICR10	Interrupt control register 10
	FFFE_805C$_H$	IR0ICR11	Interrupt control register 11
	FFFE_8060$_H$	IR0ICR12	Interrupt control register 12
	FFFE_8064$_H$	IR0ICR13	Interrupt control register 13
	FFFE_8068$_H$	IR0ICR14	Interrupt control register 14
	FFFE_806C$_H$	IR0ICR15	Interrupt control register 15
	FFFE_8070$_H$	IR0ICR16	Interrupt control register 16
	FFFE_8074$_H$	IR0ICR17	Interrupt control register 17
	FFFE_8078$_H$	IR0ICR18	Interrupt control register 18
	FFFE_807C$_H$	IR0ICR19	Interrupt control register 19
	FFFE_8080$_H$	IR0ICR20	Interrupt control register 20
	FFFE_8084$_H$	IR0ICR21	Interrupt control register 21
	FFFE_8088$_H$	IR0ICR22	Interrupt control register 22
	FFFE_808C$_H$	IR0ICR23	Interrupt control register 23
	FFFE_8090$_H$	IR0ICR24	Interrupt control register 24
	FFFE_8094$_H$	IR0ICR25	Interrupt control register 25
	FFFE_8098$_H$	IR0ICR26	Interrupt control register 26
	FFFE_809C$_H$	IR0ICR27	Interrupt control register 27
	FFFE_80A0$_H$	IR0ICR28	Interrupt control register 28
	FFFE_80A4$_H$	IR0ICR29	Interrupt control register 29
	FFFE_80A8$_H$	IR0ICR30	Interrupt control register 30
	FFFE_80AC$_H$	IR0ICR31	Interrupt control register 31
	FFFE_80B0$_H$ - FFFE_8FFF$_H$	Reserved	Access prohibited
GPIO	FFFE_9000$_H$	GPDR0	Port data register 0 When it accesses PDR0 by byte long in the big endian mode, address becomes FFFE_9003$_H$.
	FFFE_9004$_H$	GPDR1	Port data register 1 When it accesses PDR1 by byte long in the big endian mode, address becomes FFFE_9007$_H$.
	FFFE_9008$_H$	GPDR2	Port data register 2 When it accesses PDR2 by byte long in the big endian mode, address becomes FFFE_900B$_H$.
	FFFE_900C$_H$	Reserved	Access prohibited
	FFFE_9010$_H$	GPDDR0	Data direction register 0
	FFFE_9014$_H$	GPDDR1	Data direction register 1
	FFFE_9018$_H$	GPDDR2	Data direction register 2
	FFFE_901C$_H$ - FFFE_9FFF$_H$	Reserved	Access prohibited
No module	FFFE_A000$_H$ - FFFE_FFFF$_H$	Reserved	Access prohibited
No module	FFFF_0000$_H$ - FFFF_FDFF$_H$	Not Register Area	For external area.
Interrupt controller 0 (mirror) (IRC0 mirror)	FFFF_FE00$_H$	IR0IRQF	IRQ flag register
	FFFF_FE04$_H$	IR0IRQM	IRQ mask register
	FFFF_FE08$_H$	IR0ILM	Interrupt level mask register
	FFFF_FE0C$_H$	IR0ICRMN	ICR monitoring register
	FFFF_FE10$_H$	Reserved	Access prohibited

Module name	Address	Register name	Explanation
Interrupt controller 0 (mirror) (IRC0 mirror)	$FFFF\_FE14_H$	IR0DICR0	Software interrupt control register 0
	$FFFF\_FE18_H$	IR0DICR1	Software interrupt control register 1
	$FFFF\_FE1C_H$	IR0TBR	Table base register
	$FFFF\_FE20_H$	IR0VCT	Interrupt vector register
	$FFFF\_FE24_H$ - $FFFF\_FE2F_H$	Reserved	Access prohibited
	$FFFF\_FE30_H$	IR0ICR0	Interrupt control register 00
	$FFFF\_FE34_H$	IR0ICR1	Interrupt control register 01
	$FFFF\_FE38_H$	IR0ICR2	Interrupt control register 02
	$FFFF\_FE3C_H$	IR0ICR3	Interrupt control register 03
	$FFFF\_FE40_H$	IR0ICR4	Interrupt control register 04
	$FFFF\_FE44_H$	IR0ICR5	Interrupt control register 05
	$FFFF\_FE48_H$	IR0ICR6	Interrupt control register 06
	$FFFF\_FE4C_H$	IR0ICR7	Interrupt control register 07
	$FFFF\_FE50_H$	IR0ICR8	Interrupt control register 08
	$FFFF\_FE54_H$	IR0ICR9	Interrupt control register 09
	$FFFF\_FE58_H$	IR0ICR10	Interrupt control register 10
	$FFFF\_FE5C_H$	IR0ICR11	Interrupt control register 11
	$FFFF\_FE60_H$	IR0ICR12	Interrupt control register 12
	$FFFF\_FE64_H$	IR0ICR13	Interrupt control register 13
	$FFFF\_FE68_H$	IR0ICR14	Interrupt control register 14
	$FFFF\_FE6C_H$	IR0ICR15	Interrupt control register 15
	$FFFF\_FE70_H$	IR0ICR16	Interrupt control register 16
	$FFFF\_FE74_H$	IR0ICR17	Interrupt control register 17
	$FFFF\_FE78_H$	IR0ICR18	Interrupt control register 18
	$FFFF\_FE7C_H$	IR0ICR19	Interrupt control register 19
	$FFFF\_FE80_H$	IR0ICR20	Interrupt control register 20
	$FFFF\_FE84_H$	IR0ICR21	Interrupt control register 21
	$FFFF\_FE88_H$	IR0ICR22	Interrupt control register 22
	$FFFF\_FE8C_H$	IR0ICR23	Interrupt control register 23
	$FFFF\_FE90_H$	IR0ICR24	Interrupt control register 24
	$FFFF\_FE94_H$	IR0ICR25	Interrupt control register 25
	$FFFF\_FE98_H$	IR0ICR26	Interrupt control register 26
	$FFFF\_FE9C_H$	IR0ICR27	Interrupt control register 27
	$FFFF\_FEA0_H$	IR0ICR28	Interrupt control register 28
	$FFFF\_FEA4_H$	IR0ICR29	Interrupt control register 29
	$FFFF\_FEA8_H$	IR0ICR30	Interrupt control register 30
	$FFFF\_FEAC_H$	IR0ICR31	Interrupt control register 31
	$FFFF\_FEB0_H$ - $FFFF\_FEFF_H$	Reserved	Access prohibited

4. CPU (ARM926EJ-S core part)

This chapter describes CPU (ARM926EJ-S core part) of MB86R03.

4.1. Outline

ARM926EJ-S core part chiefly includes functional blocks such as ARM926EJ-S, TCM (Tightly Coupled Memory), and ETM9CS Single.

4.2. Feature

ARM926EJ-S core part has following features:

- Five stage pipeline (fetch, decode, execution, memory, and write)
- Harvard architecture
- 16KB instruction cache/16KB data cache
- 16KB instruction TCM (ITCM)/16KB data TCM (DTCM)
- JAVA acceleration (Jazelle technology)
- Coprocessor interface
- Supported MMU (Memory Management Unit)
- Built-in ETM9CS Single for real-time trace
- Corresponded to big endian and little endian

4.3. Block diagram

Figure 4-1 shows ARM926EJ-S core part's block diagram.

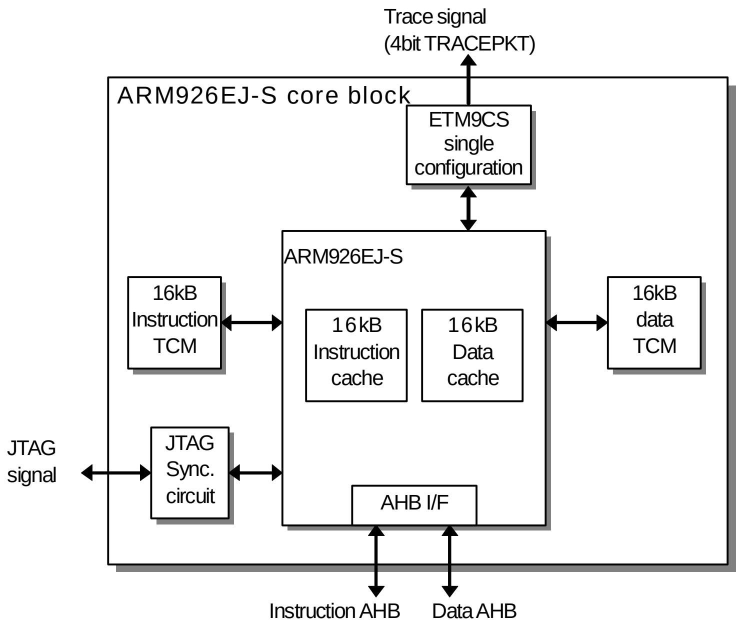


Figure 4-1 Block diagram of ARM926EJ-S core part

4.4. ARM926EJ-S and ETM setting

ARM926EJ-S cache size, both instruction and data, is set to 16KB as well as ITCM and DTCM.

MB86R03 has ETM9CS Single for real-time trace, and 4 bits are supported for TRACEPKT port of ETM9CS Single.

Refer to related material of ARM Ltd. such as shown below for detailed specification of ARM926EJ-S and ETM9CS Single.

ARM926EJ-S

ARM926EJ-S product overview

- ARM926EJ-S (r0p4/r0p5) Technical Reference Manual (DDI0198D)
- ARM9EJ-S Revision r1p2 Technical Reference Manual DDI0222B)
- ARM926EJ-S Product Overview (DVI0035B)

They are found in the following URL.
http://infocenter.arm.com/help/index.jsp

ETM9CS single

- CoreSight ETM9 r0p0 Technical Reference Manual (DDI0315A)
- ETM9 Revision r2p2 Technical Reference Manual (DDI0157F)
- Embedded Trace Macrocell Architecture Specification (IHI0014N)
- CoreSight System Design Guide (DGI0012A)

They are found in the following URL.
http://www.arm.com/documentation/Trace_Debug/index.html

5. Clock reset generator (CRG)

This chapter describes function and operation of clock reset generator (CRG.)

5.1. Outline

CRG controls clock/reset of ARM926EJ-S, AHB, and APB module.

5.2. Feature

CRG has the following features:

- Clock generator
  - Both PLL clock and external input clock (PLL by-pass mode) are operable
  - PLL control
    - a- Control of PLL oscillation and stop
    - b- Control of PLL oscillation stabilization waiting time
  - Clock gear control
  - Clock frequency of ARM core, AXI, AHB, and APB can be changed respectively
  - Supply/Stop control of clock to ARM core, AXI, AHB, and APB module

- Reset generator
  - Generation of internal reset from external reset
  - Generation of software reset
  - Input/Output control of XSRST signal for JTAG ICE
  - Generation of XTRST (TAP controller's reset) signal

- Others
  - Watchdog timer function
  - Corresponding to stop mode which halts all clocks of MB86R01 'Jade'

5.3. Block diagram

Figure 5-1 shows block diagram of clock reset generator (CRG.)

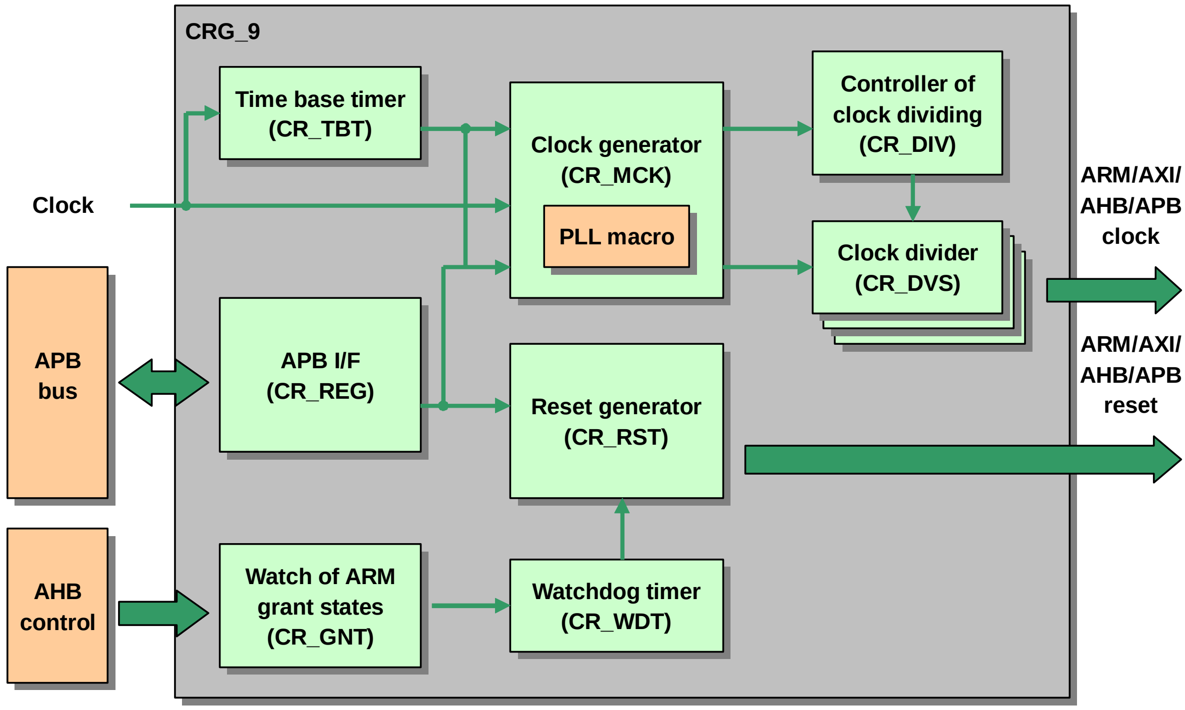


Figure 5-1 Block diagram of clock reset generator (CRG)

Table 5-1 shows function of the block included in CRG.

Table 5-1 Individual block function

Block	Function
CR_RST	Generation of reset signal
CR_MCK	PLL control/bypass
CR_GNT	ARM's grant status watch
CR_DIV	Generation of clock frequency dividing and clock enable signal
CR_DVS	Selection of clock frequency dividing and non clock frequency dividing
CR_TBT	Count of following items: • PLL oscillation stabilization waiting time • PLL reset's pulse width • Watchdog timer's clear timing • Software reset's pulse width
CR_REG	Control register
CR_WDT	Watchdog timer

5.4. Register

This section describes CRG register.

5.4.1. Register list

Table 5-2 shows list of CRG register.

Table 5-2 CRG register list

Address		Register name	Abbreviation	Explanation
Base	Offset			
FFFE_7000$_H$	+ 00$_H$	PLL control register	CRPR	To control PLL
	+ 04$_H$	(Reserved)	–	Reserved area, access prohibited
	+ 08$_H$	Watchdog timer control register	CRWR	To control watchdog timer
	+ 0C$_H$	Reset/Standby control register	CRSR	To control reset/standby
	+ 10$_H$	Clock frequency dividing control register A	CRDA	To control clock divider
	+ 14$_H$	Clock frequency dividing control register B	CRDB	To control clock divider
	+ 18$_H$	AHB(A) bus clock gate control register	CRHA	To control clock gate of AHB(A) bus
	+ 1C$_H$	APB(A) bus clock gate control register	CRPA	To control clock gate of APB(A) bus
	+ 20$_H$	APB(B) bus clock gate control register	CRPB	To control clock gate of APB(B) bus
	+ 24$_H$	AHB(B) bus clock gate control register	CRHB	To control clock gate of AHB(B) bus
	+ 28$_H$	ARM core clock gate control register	CRAM	To control clock gate of ARM core
	+ 2C$_H$ – + FF$_H$	(Reserved)	–	Reserved area, access prohibited

Description format of register

Following format is used for description of register's each bit in "5.4.2 PLL control register (CRPR)" to "5.4.11 ARM core clock gate control register (CRAM)".

Address	**Base address + Offset**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

5.4.2. PLL control register (CRPR)

This register controls PLL.

Address	FFFE_7000$_H$ + 00$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)							PLLRDY	*1	LUWMODE[1:0]		PLLMODE[4:0]				
R/W	R0	R0	R0	R0	R0	R0	R0	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0 *2	1	0	1 *3	1 *3	1 *3	1 *3	1 *3

*1: PLLBYPASS
*2: This follows external pin, PLLBYPASS
*3: This changes according to setting value of external pin, CRIPM[3:0] and PLLBYPASS

Bit field No.	Bit field Name	Description
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-9	(Reserved)	Reserved bits. Write access is ignored, and read value of these bits are always "0".
8	PLLRDY	PLLREADY monitoring This bit monitors internal signal, PLLREADY with external pin CLK clock. The PLLREADY signal shows overflow of the value selected at LUMMODE[1:0] bit by the timer which calculates PLL oscillation stabilization waiting time. 0: PLLREADY signal is "low" (initial value) 1: PLLREADY signal is "high" Write access to this bit is ignored. Note: PLLRDY=1 does not guarantee that PLL is locked and clock supply is ready.
7	PLLBYPASS	PLL bypass mode This bit bypasses PLL. 0: PLL clock is used. 1: PLL is bypassed Note: Do not change PLLBYPASS bit and PLLMODE[4:0] at the same time since clock switch of both external pin CLK and PLL clocks needs to be changed. If they are changed at the same time, CRG detects PLL oscillation frequency change and state becomes PLL oscillation stabilization waiting before PLL bypass mode. Reference: The initial value of this bit is settable with setting external pin, PLLBYPASS.

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>6-5</td><td>LUWMODE[1:0]</td><td>
<u>PLL lockup waiting mode</u>
These bits are used to set PLL oscillation stabilization wait time.
<table>
<tr><td>00</td><td>$T_{CLK} \times (2^{n0} - 2^{m} + 1)$</td></tr>
<tr><td>01</td><td>$T_{CLK} \times (2^{n1} - 2^{m} + 1)$</td></tr>
<tr><td>10</td><td>$T_{CLK} \times (2^{n2} - 2^{m} + 1)$ (initial value)</td></tr>
<tr><td>11</td><td>$T_{CLK} \times (2^{n3} - 2^{m} + 1)$</td></tr>
</table>
T_{CLK}: Cycle time of external pin CLK

n0 = 11
n1 = 12
n2 = 13
n3 = 14

m = 8

The wait time depends on CLK cycle time and PLL lock-up time, moreover it does not need to be changed from the initial value.
</td></tr>
<tr><td>4-0</td><td>PLLMODE[4:0]</td><td>
<u>PLL oscillation mode</u>
These bits are used to set PLL oscillation mode.
Initial value of PLLMODE[4:0] bit changes according to the setting of external pin, CRIPM[3:0]. Initial value of these bits is PLLMODE[4:0] = {"0", CRIPM[3], CRIPM[2], CRIPM[1], CRIPM[0].}
<table>
<tr><td>00000</td><td>$f_{CCLK} = f_{CLK} \times 24.5$ ($49 \times 1/2$)</td></tr>
<tr><td>00001</td><td>$f_{CCLK} = f_{CLK} \times 23$ ($46 \times 1/2$)</td></tr>
<tr><td>00010</td><td>$f_{CCLK} = f_{CLK} \times 18.5$ ($37 \times 1/2$)</td></tr>
<tr><td>00011</td><td>$f_{CCLK} = f_{CLK} \times 10$ ($20 \times 1/2$)</td></tr>
<tr><td>00100</td><td>$f_{CCLK} = f_{CLK} \times 23.5$ ($47 \times 1/2$)</td></tr>
<tr><td>00101</td><td>$f_{CCLK} = f_{CLK} \times 22$ ($44 \times 1/2$)</td></tr>
<tr><td>00110</td><td>$f_{CCLK} = f_{CLK} \times 18$ ($36 \times 1/2$)</td></tr>
<tr><td>00111</td><td>$f_{CCLK} = f_{CLK} \times 9.5$ ($19 \times 1/2$)</td></tr>
<tr><td>01000</td><td>$f_{CCLK} = f_{CLK} \times 19.5$ ($39 \times 1/2$)</td></tr>
<tr><td>01001</td><td>$f_{CCLK} = f_{CLK} \times 19$ ($38 \times 1/2$)</td></tr>
<tr><td>01010</td><td>$f_{CCLK} = f_{CLK} \times 15$ ($30 \times 1/2$)</td></tr>
<tr><td>01011</td><td>$f_{CCLK} = f_{CLK} \times 7.5$ ($15 \times 1/2$)</td></tr>
<tr><td>11111</td><td>PLL stops</td></tr>
<tr><td>Others</td><td>Reserved (setting prohibited)</td></tr>
</table>
f_{CCLK} : Clock frequency of CCLK
f_{CLK} : Clock frequency of external pin CLK

Note: Do not change PLLMODE[4:0] when PLLBYPASS bit is 0.
</td></tr>
</table>

5.4.3. Watchdog timer control register (CRWR)

This register controls watchdog timer.

Address	$FFFE\_7000_H + 08_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								ERST	(Reserved)		TBR	WDRST	WDTSET/ WDTCLR	WDTMODE[1:0]	
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R/W0	R0	R0/W0*	R/W1	R/W0	R/W1	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	0	0	0	X	0	0	0

*: Do not set "1" to bit 5

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31-16</td><td>–</td><td>Unused bits.
Write access is ignored, and read value of these bits is undefined.</td></tr>
<tr><td>15-8</td><td>(Reserved)</td><td>Reserved bits.
Write access is ignored, and read value of these bits are always "0".</td></tr>
<tr><td>7</td><td>ERST</td><td><u>Internal reset of ERSTn monitoring</u>
This bit monitors internal signal of ERSTn.<table><tr><td>0</td><td>ERSTn bit is cleared.</td></tr><tr><td>1</td><td>External reset (XRST) is asserted. (initial value)</td></tr></table>The initial value of this bit is set to 1 by falling edge of ERSTn., and writing "1" is ignored.
This bit is set by ERSTn.</td></tr>
<tr><td>6</td><td>(Reserved)</td><td>Reserved bits.
Write access is ignored, and read value of this bit is always "0".</td></tr>
<tr><td>5</td><td>(Reserved)</td><td>Reserved bit, always write 0.
Read value of this bit is always "0".</td></tr>
<tr><td>4</td><td>TBR</td><td><u>Time based timer reset request</u>
This bit resets the time based timer, and its reset signal is asserted during 1 cycle of APB clock.<table><tr><td>0</td><td>Time based timer is not reset (initial value)</td></tr><tr><td>1</td><td>Time based timer is reset</td></tr></table>Writing 0 is ignored.</td></tr>
<tr><td>3</td><td>WDRST</td><td><u>Watchdog reset monitoring</u>
This bit monitors watchdog reset.<table><tr><td>0</td><td>Watchdog reset is not asserted</td></tr><tr><td>1</td><td>Watchdog reset is asserted</td></tr></table>The initial value of this bit is undefined, and writing 1 is ignored.
When watchdog is reset, this bit is set to "1".</td></tr>
<tr><td>2</td><td>WDTSET
/WDTCLR</td><td><u>Setting and clear of watchdog timer</u>
This bit sets and clears watchdog timer which starts count at writing "1" and clears at writing "1" from the second time.<table><tr><td>0</td><td>The watchdog timer is not set (initial value)</td></tr><tr><td>1</td><td>First time: The watchdog timer starts
Second time and later: The watchdog timer is cleared</td></tr></table>Writing 0 is ignored.</td></tr>
</table>

Bit field		Description
No.	Name	
1-0	WDTMODE[1:0]	These bits set timing to clear watchdog timer. Watchdog reset occurs at following periods when "1" is written to WDTSET/WDTCLR bits at the end. 00: $T_{CLK} \times 2^{n0}$ - $T_{CLK} \times 2^{(n0+1)}$ (initial value) 01: $T_{CLK} \times 2^{n1}$ - $T_{CLK} \times 2^{(n1+1)}$ 10: $T_{CLK} \times 2^{n2}$ - $T_{CLK} \times 2^{(n2+1)}$ 11: $T_{CLK} \times 2^{n3}$ - $T_{CLK} \times 2^{(n3+1)}$ T_{CLK}: Cycle time of external pin CLK n0 = 9 n1 = 12 n2 = 14 n3 = 16 Select the bit that corresponds to the system.

5.4.4. Reset/Standby control register (CRSR)

This register controls reset and standby.

Address	FFFE_7000$_H$ + 0C$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								STOPEN	(Reserved)		Reserved	SRST	SWRST	SWRST REQ	SWRM ODE
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R/W	R0	R0	R/W0	R/W0	R/W0	R/W1	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	X	X	0	0

Bit field No.	Bit field Name	Description
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-8	(Reserved)	Reserved bits. Write access is ignored, and read value of these bits are always "0".
7	STOPEN	Stop mode enable This bit stops all bus clock operations in the standby mode. 0: Bus clock operation in the standby mode does not stop (initial value) 1: All bus clock operations in the standby mode are stopped Note: When changing state to stop mode, write "1" to PLLBYPASS bit of CRPR.
6-5	(Reserved)	Reserved bits. Write access is ignored, and read value of these bits are always "0".
4	(Reserved)	Reserved bit. Always write "0" to write access.
3	SRST	nSRST monitoring This bit monitors nSRST reset from ICE. 0: nSRST is not asserted 1: nSRST is asserted Initial value of this bit is undefined, and writing "0" is ignored. When nSRST occurs, this bit is set to "1".
2	SWRST	Software reset monitoring This bit monitors software reset. 0: Software reset is not asserted 1: Software reset is asserted Initial value of this bit is undefined, and writing "0" is ignored. When software reset occurs, this bit is set to "1".
1	SWRSTREQ	Software reset request This bit asserts software reset. 0: Software reset is not requested (initial value) 1: Software reset is requested Writing 0 is ignored, and this bit is cleared with reset signal.

Bit field		Description
No.	Name	
0	SWRMODE	Pulse width mode of software reset This bit sets pulse width of software reset. 0: $T_{CLK} \times (2^{n0+3}) + T_{CCLK} \times 7$ (initial value) 1: $T_{CLK} \times (2^{n1+3}) + T_{CCLK} \times 7$ T_{XCLK}: Cycle time of external pin CLK T_{CCLK}: Cycle time of internal signal CCLK n0 = 7 n1 = 12 Pulse width of software reset depends on the CLK cycle time and internal operation frequency setting. Select the bit that corresponds to the system.

5.4.5. Clock divider control register A (CRDA)

This register controls clock divider.

Address	$FFFE\_7000_H + 10_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)	ARMBDM[2:0]			ARMADM[2:0]			PBDM[2:0]			PADM[2:0]			HADM[2:0]		
R/W	R0	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
Initial value	0	0	0	1	0	0	0	0	1	1	0	1	1	0	1	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31-16</td><td>–</td><td>Unused bits.
Write access is ignored, and read value of these bits is undefined.</td></tr>
<tr><td>15</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored, and read value of these bits are always "0".</td></tr>
<tr><td>14-12</td><td>ARMBDM[2:0]</td><td>ARMBCLK frequency dividing mode
These bits set frequency dividing ratio of ARMBCLK.
<table>
<tr><td>000</td><td>$f_{ARMBCLK} = f_{CCLK} \times (1/1)$</td></tr>
<tr><td>001</td><td>$f_{ARMBCLK} = f_{CCLK} \times (1/2)$ (initial value)</td></tr>
<tr><td>010</td><td>$f_{ARMBCLK} = f_{CCLK} \times (1/4)$</td></tr>
<tr><td>011</td><td>$f_{ARMBCLK} = f_{CCLK} \times (1/8)$</td></tr>
<tr><td>100</td><td>$f_{ARMBCLK} = f_{CCLK} \times (1/16)$</td></tr>
<tr><td>Others</td><td>Reserved (setting prohibited)</td></tr>
</table>
$f_{ARMBCLK}$: Clock frequency of ARMBCLK
f_{CCLK} : Clock frequency of CCLK</td></tr>
<tr><td>11-9</td><td>ARMADM[2:0]</td><td>ARMACLK dividing mode
These bits set frequency dividing ratio of ARMACLK.
<table>
<tr><td>000</td><td>$f_{ARMACLK} = f_{CCLK} \times (1/1)$ (initial value)</td></tr>
<tr><td>001</td><td>$f_{ARMACLK} = f_{CCLK} \times (1/2)$</td></tr>
<tr><td>010</td><td>$f_{ARMACLK} = f_{CCLK} \times (1/4)$</td></tr>
<tr><td>011</td><td>$f_{ARMACLK} = f_{CCLK} \times (1/8)$</td></tr>
<tr><td>100</td><td>$f_{ARMACLK} = f_{CCLK} \times (1/16)$</td></tr>
<tr><td>Others</td><td>Reserved (setting prohibited)</td></tr>
</table>
$f_{ARMBCLK}$: Clock frequency of ARMACLK
f_{CCLK} : Clock frequency of CCLK</td></tr>
</table>

Bit field No.	Bit field Name	Description
8-6	PBDM[2:0]	PBCLK frequency dividing mode These bits set frequency dividing ratio of PBCLK. 000: $f_{PBCLK} = f_{CCLK} \times (1/1)$ 001: $f_{PBCLK} = f_{CCLK} \times (1/2)$ 010: $f_{PBCLK} = f_{CCLK} \times (1/4)$ 011: $f_{PBCLK} = f_{CCLK} \times (1/8)$ (initial value) 100: $f_{PBCLK} = f_{CCLK} \times (1/16)$ Others: Reserved (setting prohibited) f_{PBCLK} : Clock frequency of PBCLK f_{CCLK} : Clock frequency of CCLK
5-3	PADM[2:0]	PACLK frequency dividing mode These bits set frequency dividing ratio of PACLK. 000: $f_{PACLK} = f_{CCLK} \times (1/1)$ 001: $f_{PACLK} = f_{CCLK} \times (1/2)$ 010: $f_{PACLK} = f_{CCLK} \times (1/4)$ 011: $f_{PACLK} = f_{CCLK} \times (1/8)$ (initial value) 100: $f_{PACLK} = f_{CCLK} \times (1/16)$ Others: Reserved (setting prohibited) f_{PACLK} : Clock frequency of PACLK f_{CCLK} : Clock frequency of CCLK
2-0	HADM[2:0]	HACLK frequency dividing mode These bits set frequency dividing ratio of HACLK. 000: $f_{HACLK} = f_{CCLK} \times (1/1)$ 001: $f_{HACLK} = f_{CCLK} \times (1/2)$ 010: $f_{HACLK} = f_{CCLK} \times (1/4)$ (initial value) 011: $f_{HACLK} = f_{CCLK} \times (1/8)$ 100: $f_{HACLK} = f_{CCLK} \times (1/16)$ Others: Reserved (setting prohibited) f_{HACLK} : Clock frequency of HACLK f_{CCLK} : Clock frequency of CCLK

Note:

ARMACLK must not be slower than HACLK; moreover, HACLK must not be slower than PACLK.

f_{ARMCLK} >= f_{HACLK} >= f_{PACLK}

5.4.6. Clock divider control register B (CRDB)

This register controls clock divider.

Address	$FFFE\_7000_H + 14_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)													HBDM[2:0]		
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit field No.	Bit field Name	Description
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-3	(Reserved)	Reserved bits. Write access is ignored, and read value of these bits are always "0".
2-0	HBDM[2:0]	HBCLK frequency dividing mode These bits set frequency dividing ratio of HBCLK. (see table below) f_{HBCLK} : Clock frequency of HBCLK f_{CCLK} : Clock frequency of CCLK

HBDM[2:0]	Frequency dividing ratio of HBCLK
000	$f_{HBCLK} = f_{CCLK} \times (1/1)$
001	$f_{HBCLK} = f_{CCLK} \times (1/2)$ (initial value)
010	$f_{HBCLK} = f_{CCLK} \times (1/4)$
011	$f_{HBCLK} = f_{CCLK} \times (1/8)$
100	$f_{HBCLK} = f_{CCLK} \times (1/16)$
Others	Reserved (setting prohibited)

5.4.7. AHB (A) bus clock gate control register (CRHA)

This register controls clock gate of AHB (A) bus.

Address	FFFE_7000$_H$ + 18$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HAGATE[15:0]															
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
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit field		Description
No.	Name	
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-0	HAGATE[15:0]	<u>HACLK clock gate control</u> These bits control HACLK clock gate. HAGATE[n] = 0: HACLKn stops HAGATE[n] = 1: HACLKn does not stop (initial value) HACLK0: AHB1, AHB2, APBBRG0, APBBRG1, APBBRG2 HACLK1: External bus I/F, CCPB HACLK2: SRAM HACLK3: HDMAC HACLK4: (Reserved) HACLK5: Boot ROM HACLK6: (Reserved) HACLK7: I2S_0, I2S_1, I2S_2 HACLK8: DMAC HACLK9: (Reserved) HACLK10: SD I/F HACLK11: (Reserved) HACLK12: (Reserved) HACLK13: GDC HACLK14: (Reserved) HACLK15: DDR2 controller

Nested table within description of bits 15-0:

HAGATE[n]	Description
0	HACLKn stops
1	HACLKn does not stop (initial value)

5.4.8. APB (A) bus clock gate control register (CRPA)

This register controls clock gate of APB (A) bus.

Address	$FFFE\_7000_H + 1C_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PAGATE[15:0]															
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
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit field		Description
No.	Name	
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-0	PAGATE[15:0]	<u>PACLK clock gate control</u> These bits control PACLK clock gate. PAGATE[n] / Description: 0: PACLKn stops 1: PACLKn does not stop (initial value) PACLK0: IRC PACLK1: EXIRC PACLK2: UART0, UART1 PACLK3: GPIO PACLK4: RBC PACLK5: 32 bit timer PACLK6: I2C × 2 (I2C_0, I2C_1) PACLK7: CAN × 2 (CAN_0, CAN_1) PACLK8: UART2, UART3 PACLK9: ADC × 2 (ADC0, ADC1) PACLK10: PWM 2ch PACLK11: SPI PACLK12: CCNT PACLK13: UART4, UART5 PACLK14: ETM9CSSingle APB port PACLK15: (Reserved)

5.4.9. APB (B) bus clock gate control register (CRPB)

This register controls clock gate of APB (B) bus.

Address	$FFFE\_7000_H + 20_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PBGATE[15:0]															
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
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit field		Description
No.	Name	
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-0	PBGATE[15:0]	These bits control PBCLK clock gate. This LSI does not use them. <table><tr><th>PBGATE[n]</th><th>Description</th></tr><tr><td>0</td><td>PBCLKn stops</td></tr><tr><td>1</td><td>PBCLKn does not stop (initial value)</td></tr></table>

5.4.10. AHB (B) bus clock gate control register (CRHB)

This register controls clock gate of AHB (B) bus.

Address	FFFE_7000$_H$ + 24$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HBGATE[15:0]															
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
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit field		Description
No.	**Name**	
31-16	–	Unused bits. Write access is ignored, and read value of these bits is undefined.
15-0	HBGATE[15:0]	HBCLK clock gate control These bits control HBCLK clock gate. HBGATE[n] | Description 0 | HBCLKn stops 1 | HBCLKn does not stop (initial value) HBCLK0: GDC (HOST IF) HBCLK1: GDC (DRAW, GEO), MBUS2AXI (DRW) HBCLK2: (Reserved) HBCLK3: GDC (DISP0), MBUS2AXI (DISP) HBCLK4: GDC (DISP1) HBCLK5: GDC (CAP0), MBUS2AXI (CAP) HBCLK6: GDC (CAP1) HBCLK7: AXI, AHB2AXI, HBUS2AXI HBCLK8: DDR2 controller, DDR2 I/F HBCLK9: MLB HBCLK10: (Reserved) HBCLK11: (Reserved) HBCLK12: (Reserved) HBCLK13: (Reserved) HBCLK14: (Reserved) HBCLK15: (Reserved)

5.4.11. ARM core clock gate control register (CRAM)

This register controls clock gate of ARM core.

Address	FFFE_7000$_H$ + 28$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)											ARMBGATE	(Reserved)			ARMAGATE
R/W	R1	R1	R1	R1	R1	R1	R1	R1	R1	R1	R1	R/W	R1	R1	R1	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

Bit field No.	Bit field Name	Description
31-16	–	Unused bits. The write access is ignored, and read value of these bits is undefined.
15-5	(Reserved)	Reserved bits. Write access is ignored, and read value of these bits is always "1".
4	ARMBGATE	ARMBCLK clock gate control This bit controls ARMBCLK clock gate. 0: ARMBCLK stops 1: ARMBCLK does not stop (initial value) This clock is used to ATCLK of ETM9CS Single.
3-1	(Reserved)	Reserved bits. Write access is ignored, and read value of these bits is always "1".
0	ARMAGATE	ARMACLK clock gate control This bit controls ARMACLK clock gate. 0: ARMACLK stops 1: ARMACLK does not stop (initial value) After stopping this clock, proceed system reset to resume operation.

5.5. Operation

This section descries CRG operation.

5.5.1. Generation of reset

Factor

There are following five reset factors.

1. External reset (XRST pin input)
   The entire chip is initialized by the reset input from external pin, XRST. When external pin, PLLBYPASS is set to "L", external reset shifts to PLL oscillation stabilization waiting state.

2. Software reset (reset with register control)
   Software reset occurs with writing "1" to SWRSTREQ bit of the Reset/Standby control register (CRSR). It does not change state to PLL oscillation stabilization even though PLLBYPASS bit of the PLL control register (CRPR) is "0" (setting that uses PLL clock.)
   Moreover, this reset does not change the CRG module register, the VINITHI control register of remap/boot controller (RBC), and the INITRAM control register.
   Clock source of the software reset is time based timer’s count value. It is cleared when software reset is asserted.
   This software reset generates the internal signal, which does not reset as CRSTn.

3. XSRST (reset request from debugging tool)
   This signal is reset request from debugging tool (e.g. MultiICE), and internal reset request is able to transmit to the tool through XSRST pin This module recognizes the reset signal to be the same reset request as external reset’s.

4. XTRST (built-in ICE macro reset request from debugging tool)
   This signal is built-in ICE macro reset request from debugging tool (e.g. MultiICE), and the reset signal is to request reset to built-in ICE macro in ARM9. Although the reset signal is asserted, other peripherals are not initialized. ETM9CS Single is also reset by this signal.

5. Watchdog reset

When WDTSET/WDTCLR bits of the watchdog timer control register (CRWR) are set to "1" after external reset, watchdog timer starts. Writing "1" to the WDTSET/WDTCLR bits at the second time or later clears the timer.
Clock source of the watchdog timer is count value of the time based timer.
Clear operation of time based timer affects on watchdog timer’s count value.
When the timer is cleared, the watchdog timer is also cleared.

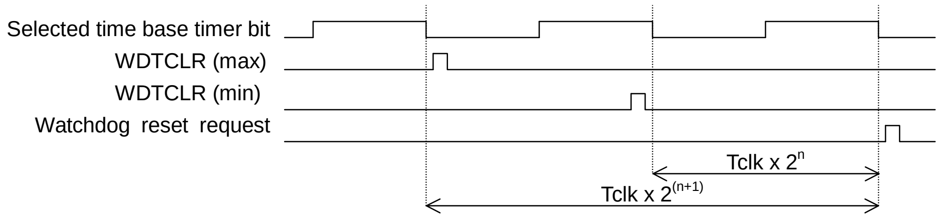


Figure 5-2 Timing of watchdog reset

As shown in Figure 5-2, watchdog reset occurs after second falling edge of selected time based timer bit.
During PLL oscillation stabilization waiting time and ARM9 debug mode (DBGACK = 1), CRG clears watchdog timer. Moreover, it monitors standby mode of ARM9 and clears watchdog timer automatically in the standby mode (standby mode = 1.)

Reset output signal

Reset signal output from the reset generator based on the reset factor is as follows.

HRESETn (AHB/APB bus reset)

This internal reset signal initializes ARM9 and AHB/APB peripherals, and it is output by external reset, software reset or XSRST reset.

XSRST (reset monitoring)

This signal reports to external circuit of ARM’s internal reset source, moreover it is asserted the same as HRESETn signal.

Internal XTRST (built-in ICE macro reset)

This signal initializes built-in macro of ARM9. The macro must be reset at power-on so that this signal is output by external reset or external XTRST reset.

CRSTn (internal reset)

This signal is output by external reset or XSRST reset.

Table 5-3 shows correlation between reset factor and reset output signal.

Table 5-3 Correlation between reset factor and reset output signal

Reset output	Reset factor				
	External reset	Software reset	Input XSRST	XTRST	Watchdog reset
HRESETn	Asserted	Asserted	Asserted	Not asserted	Asserted
Output XSRST	Asserted	Asserted	Not asserted	Not asserted	Asserted
Internal XTRST	Asserted	Not asserted	Not asserted	Asserted	Not asserted
CRSTn	Asserted	Not asserted	Asserted	Not asserted	Asserted

5.5.2. Clock generation

Figure 5-3 shows clock generation chart.

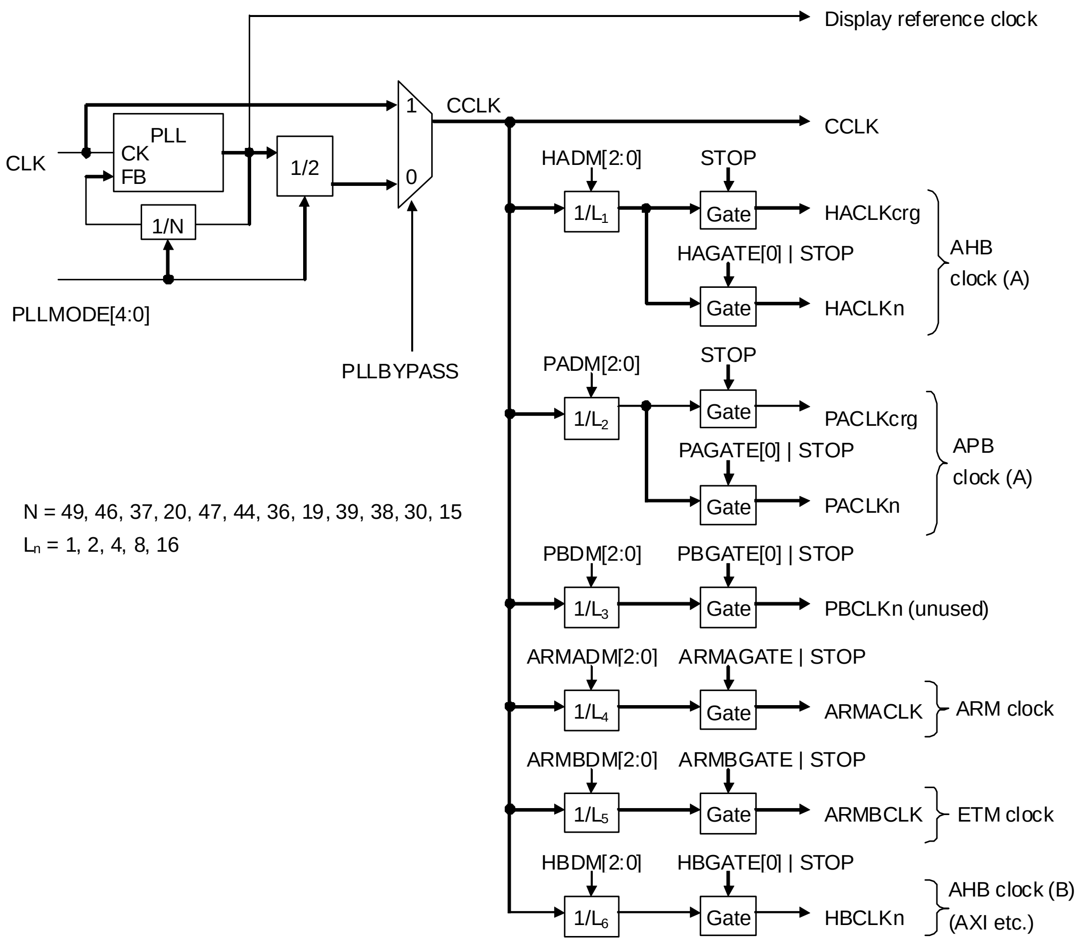


Figure 5-3 Clock generation chart

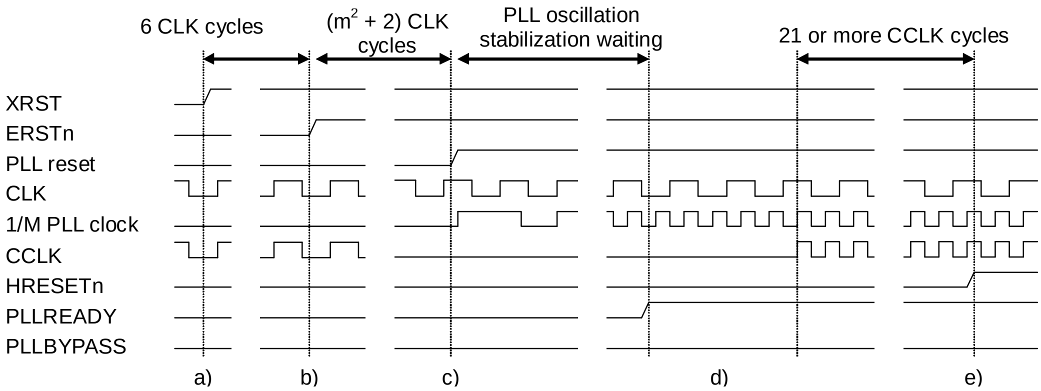
PLL control

Oscillation stabilization waiting

The clock transmission source in oscillation stabilization waiting is count value of the time based timer. Clear operation of time based timer affects on its count value.

When this module state is changed to PLL oscillation stabilization waiting state as shown below, the time based timer is cleared.

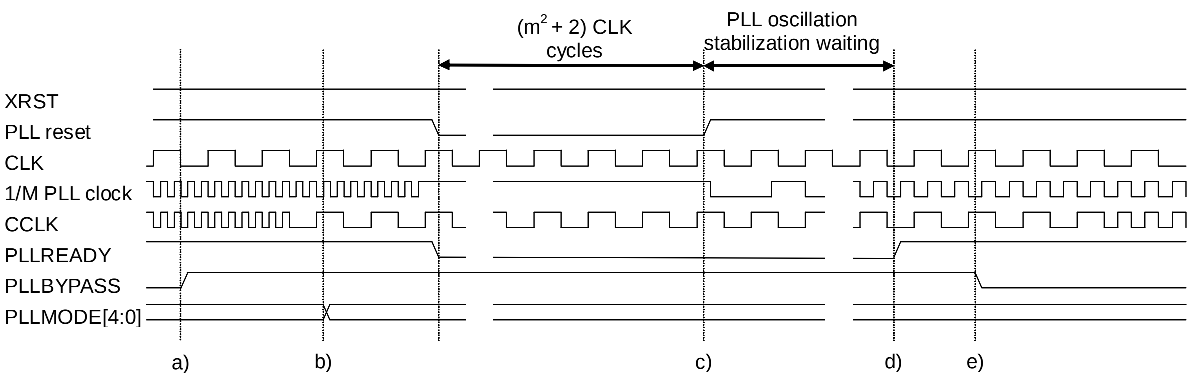
(1) External reset is asserted ("M" in Figure 5-4 and "m" of LUWMODE in the 5.4.2 PLL control register (CRPR))



a) External reset deasserted (XRST)
b) ERSTn reset (CRG internal signal)
c) PLL reset deasserted
d) PLL ready
e) HRESETn deasserted

Figure 5-4 PLL oscillation stabilization waiting state after external reset

(2) PLL oscillation frequency is changed by PLL mode ("M" in Figure 5-5 and "m" of LUWMODE in the 5.4.2 PLL control register (CRPR))



a) Clock source change (write PLLBYPASS bit)
b) Set the PLL oscillation mode (write PLLMODE bits)
c) PLL reset de-asserted
d) PLL ready (PLLREADY can be monitored by PLLRDY bit)
e) Clock source change (write PLLBYPASS bit)

Figure 5-5 PLL oscillation stabilization waiting state by PLL mode change

(3) Returning from stop mode by external interrupt (see Figure 5-9)

(4) Watchdog reset is asserted

Frequency change

Oscillation frequency and frequency dividing ratio (M) of PLL ($f_{CLK} \times N$) are set by PLLMODE[4:0] bit of the PLL control register (CRPR), and the frequency is able to be changed during the operation (see Table 5-4.)

Do not change PLLMODE[4:0] when PLLBYPASS bit of the PLL control register (CRPR) is 0.

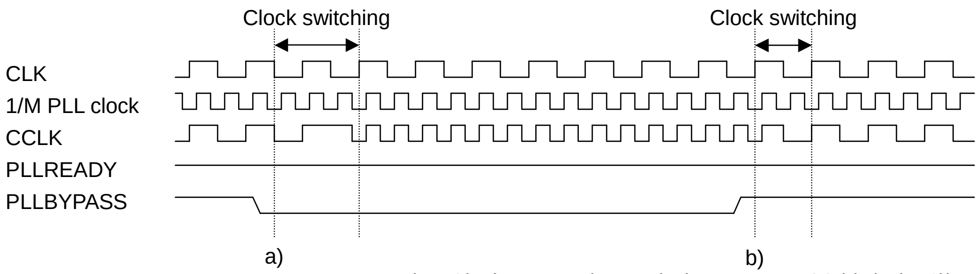
Initial value at start up is determined by external pin, PLLBYPASS and CRIPM[3:0]. To specify PLLSTOP with the initial value, fix external pin, PLLBYPASS to "1" as well.

Table 5-4 Setting example of input frequency and multiple number

Operation frequency	Initial setting: {PLLBYPASS, CRIPM[3:0]} At operation: PLLMODE[4:0]					Multiple number	Input frequency CLK	PLL output /Display reference clock	CCLK	ARMACLK	ARMBCLK	HACLKn	HBCLKn	PACLKn
	4	3	2	1	0									
333M	0	0	0	0	0	49	13.5MHz	661.5MHz	330.8MHz	330.8MHz	165.4MHz	82.7MHz	165.4MHz	41.3MHz
	0	0	0	0	1	46	14.3MHz	658.7MHz	329.4MHz	329.4MHz	164.7MHz	82.3MHz	164.7MHz	41.2MHz
	0	0	0	1	0	37	17.7MHz	656.0MHz	328.0MHz	328.0MHz	164.0MHz	82.0MHz	164.0MHz	41.0MHz
	0	0	0	1	1	20	33.3MHz	666.6MHz	333.3MHz	333.3MHz	166.7MHz	83.3MHz	166.7MHz	41.7MHz
320M	0	0	1	0	0	47	13.5MHz	634.5MHz	317.3MHz	317.3MHz	158.6MHz	79.3MHz	158.6MHz	39.7MHz
	0	0	1	0	1	44	14.3MHz	630.1MHz	315.0MHz	315.0MHz	157.5MHz	78.8MHz	157.5MHz	39.4MHz
	0	0	1	1	0	36	17.7MHz	638.3MHz	319.1MHz	319.1MHz	159.6MHz	79.8MHz	159.6MHz	39.9MHz
	0	0	1	1	1	19	33.3MHz	633.3MHz	316.6MHz	316.6MHz	158.3MHz	79.2MHz	158.3MHz	39.6MHz
266M	0	1	0	0	0	39	13.5MHz	526.5MHz	263.3MHz	263.3MHz	131.6MHz	65.8MHz	131.6MHz	32.9MHz
	0	0	0	1	0	37	14.3MHz	529.8MHz	264.9MHz	264.9MHz	132.5MHz	66.2MHz	132.5MHz	33.1MHz
	0	1	0	1	0	30	17.7MHz	531.9MHz	266.0MHz	266.0MHz	133.0MHz	66.5MHz	133.0MHz	33.2MHz
	0	1	0	1	1	15	33.3MHz	500.0MHz	250.0MHz	250.0MHz	125.0MHz	62.5MHz	125.0MHz	31.2MHz
	1	1	1	1	1	PLL STOP								

PLLBYPASS

Main clock (CCLK) of this module is able to be switched dynamically between PLL clock and external input clock (CLK) by PLLBYPASS bit of the PLL control register (CRPR.)



a) Clock source change (write PLLBYPASS bit (write 0))

b) Clock source change (write PLLBYPASS bit (write 1))

Figure 5-6 Clock switch between PLL clock and external clock

Clock gear

CRG corresponds to the clock gear function with clock enable signal.

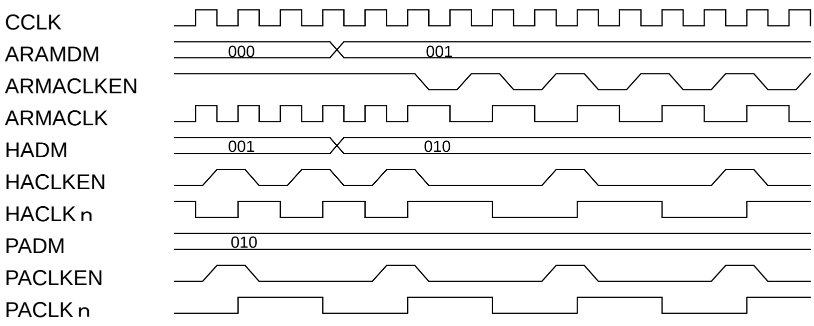


Figure 5-7 Clock gear

Standby mode (standby and stop)

ARM9 and CRG correspond to following two standby modes.

(1) Standby mode

ARM926EJ-S core corresponds to standby mode that is called "Wait for interrupt mode" with CP15. The STANDBYWFI signal is asserted and internal clock gate is closed not to supply input clock to sub module during the standby mode (refer to ARM926EJ-S Technical Reference Manual, "12.1.1 Dynamic power management (wait for interrupt mode)".)

This CRG does not equip function to stop ARMCLK in the standby mode.

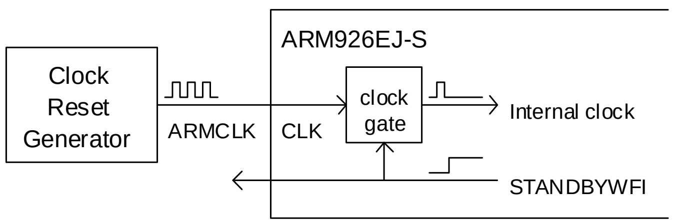


Figure 5-8 STANDBYWFI mode (ARM926EJ-S)

(2) STOP mode

When STANDBYWFI (ARM926EJ-S) signal is set to "1" with STOPEN = 1, the state changes to STOP mode through standby mode (at STOPEN = 1, this module's STANDBYWFI signal is "1".)

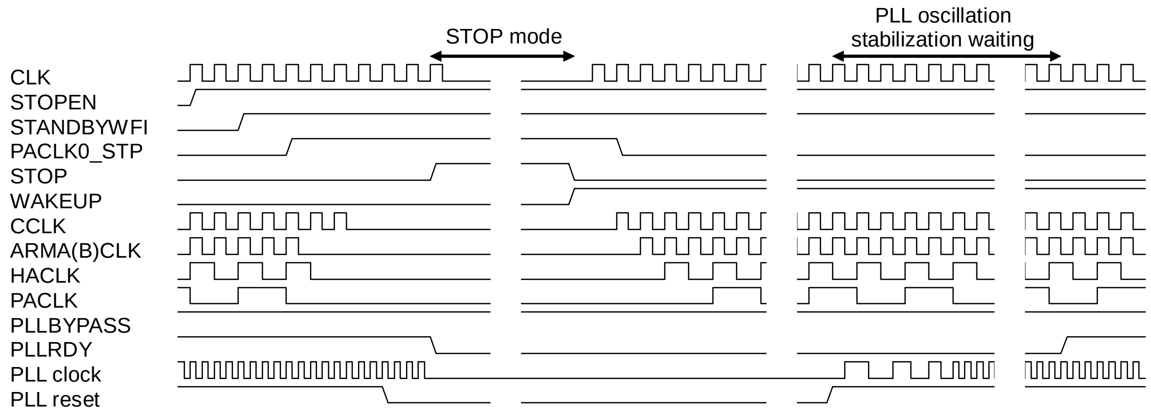
In this mode, CRG stops all clocks and PLL oscillation; moreover, the stop mode is released with external rest or external interrupt.

Figure 5-9 shows STOP mode operation.

Note:

When state is changed to the stop mode, "1" should be written to PLLBYPASS bit of the PLL control register (CRPR.)

Although PLL proceeds oscillation stabilization waiting at STOP mode release, clock is not switched to PLL clock until PLLBYPASS bit becomes "0"; in addition, PLL oscillation stabilization waiting state is skipped when PLLMODE[4:0] is 5'b11111.



* STOP = CLK clock is able to stop while the value is "1"

Figure 5-9 Stop mode

6. Remap boot controller (RBC)

This chapter describes function and operation of remap boot controller (RBC.)

6.1. Outline

RBC is APB slave module. It provides system boot operation control and controls remap sequence of the system, VINITHI signal of ARM926EJ-S™, and INITRAM signal that enable exception vector address change and ITCM reboot after power-on reset.

6.2. Feature

RBC has following features:

- Remap control register
- INITRAM signal control register
- VINITHI signal control register

6.3. Block diagram

Figure 6-1 shows RBC block diagram.

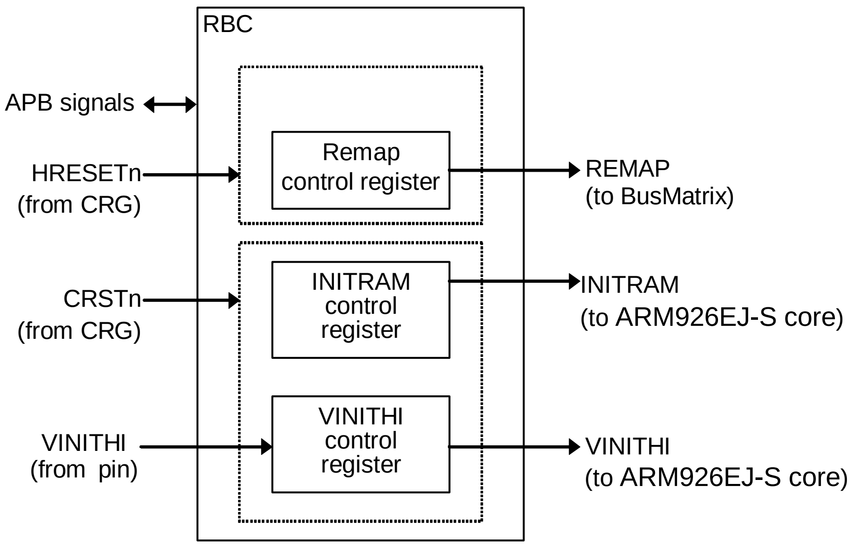


Figure 6-1 RBC block diagram

Table 6-1 shows RBC's external port function.

Table 6-1 RBC external port function list

Signal name	I/O	Description
VINITHI	I	Default value of output port, VINITHI

6.4. Supply clock

APB clock is supplied to RBC. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

6.5. Register

This section describes RBC register.

6.5.1. Register list

RBC is controlled by the register shown in Table 6-2.

Table 6-2 RBC register list

Address		Register name	Abbreviation	Description
Base	Offset			
FFFE_6000$_H$	+ 00$_H$	(Reserved)	–	Reserved area (access prohibited)
	+ 04$_H$	Remap control register	RBREMAP	Remap state control
	+ 08$_H$	VINITHI control register A	RBVIHA	VINITHI output signal control
	+ 0C$_H$	INITRAM control register A	RBITRA	INITRAM output signal control
	+ 10$_H$ – + FFF$_H$	(Reserved)	–	Reserved area (access prohibited)

Description format of register

Following format is used for description of register’s each bit in "6.5.2 Remap control register (RBREMAP)" to "6.5.4 INITRAM control register A (RBITRA)".

Address	Base address + Offset															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field’s value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

6.5.2. Remap control register (RBREMAP)

Remap control register (RBREMAP) controls remap state. Once remap is carried out, its state kept until reset. Write operation to this register is valid only the first time after reset, and its second time or later is ignored.

This register is reset by HRESETn input.

This register should be accessed in word unit.

Address	**GPR0: FFFE_6000$_H$ + 04$_H$**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															REMAP
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		**Description**
No.	**Name**	
31-1	(Reserved)	Reserved bit.
0	REMAP	Remap state is controlled. When write operation to remap register is performed (both "0" and "1" of write data are available) REMAP output signal becomes high. BusMatrix is designed to remap memory map with REMAP output signal. REMAP = Low: Vector area is allocated to internal boot ROM REMAP = High: Vector area is allocated to internal SRAM_0

6.5.3. VINITHI control register A (RBVIHA)

VINITHI control register A (RBVIHA) controls VINITHI output signal. This register is reset by the CRSTn input, and its initial value is determined by input level of external pin, VINITHI.
This register should be accessed in word unit.

Address	GPR0: FFFE_6000$_H$ + 08$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	Determined by input level of external pin, VINITHI															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															VIHA
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial value	Determined by input level of external pin, VINITHI															

Bit field		Description
No.	Name	
31-1	(Reserved)	Reserved bits. Write access is ignored. Reading these bits enable reading the value set by VINITHI.
0	VIHA	VINTHI output signal is controlled.

6.5.4. INITRAM control register A (RBITRA)

INITRAM control register A (RBITRA) controls INITRAM output signal.
This register is reset by the CRSTn input. It should be accessed in word unit.

Address	GPR0: FFFE_6000$_H$ + 0C$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															ITRA
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-1	(Reserved)	Reserved bits. Write access is ignored. Read value of these bits is always "0".
0	ITRA	INTRAM output signal is controlled.

6.6. Operation

This section describes RBC operation.

6.6.1. RBC reset

RBC has two reset input ports.
RBREMAP register is reset by HRESETn input, and RBVIHA and RBITRA registers are reset by CRSTn value.
Table 6-3 shows correlation between these reset and register.

Table 6-3 Correlation between reset and register

Reset input	Register	Description
HRESETn	RBREMAP	This port is reset by HRESETn.
CRSTn	RBVIHA	This port value reflects to value of external pin, VINITHI by CRSTn input.
	RBITRA	This port is reset by CRSTn input.

6.6.2. Remap control

Remap changes vector area (00000000_H - 00008000_H) after power-on.
Vector area is allocated to built-in boot ROM at power-on and the system starts up from it.
With the remap control, the allocated area is changed to built-in SRAM_0; then vector table is able to be overwritten.

6.6.3. VINITHI control

ARM926EJ-S has VINITHI signal which determines exception vector address.
When it is low at reset, the exception vector is located in 00000000_H. On the other hand, when the signal is high at reset, the exception vector is located in $FFFF0000_H$.
Refer to "Technical reference manual" of individual ARM9 provided by ARM Ltd. for detail of VINITHI signal.
The initial value of RBVIHA register is defined by external pin, VINITHI.

6.6.4. INITRAM control

ARM926EJ-S has INITRAM signal. When it is high at reset, instruction TCM automatically becomes valid which enables reboot operation from ITCM.
Refer to "Technical reference manual" of individual ARM9 core provided by ARM Ltd. for detail of INITRAM signal.
RBITRA register is initialized to "0" by CRSTn, however it is not reset by HRESETn. This means, reboot operation from ITCM is able to be proceeded at software reset when exception vector table is copied to ITCM before software reset

7. Interrupt controller (IRC)

This chapter describes function and operation of interrupt controller (IRC.)

7.1. Outline

IRC consists of two channels, IRC0 and IRC1 which determine priority of IRQ source up to 32 factors respectively, and report to ARM core the highest priority IRQ source as IRQ interrupts. Therefore, those channels have priority setting register of IRQ factor and level setting register for the interrupt from ARM core.

Note:
The IRQ interrupt determined by IRC1 is accepted as IRQ6 interrupt factor of IRC0. Therefore, priority of all IRQ sources allocated to IRC1 is determined according to IRC1 and IRC0’s IRQ6 settings.

The IRQ vector defined in ARM926EJ-S is only "0×18", but the vector table factor is extended to 32 by IRC. When IRQ interrupt is asserted to the ARM core, interrupt vector table address corresponding to the IRQ interrupt factor is generated and displayed during the register.
IRQ interrupt handler must refer to the vector table extended further than "0×18".

IRC, connected to APB bus has delay interrupt control circuit and interrupt wake-up circuit from stop/standby mode which is composed of clock control circuit.

7.2. Feature

IRC has following features:

- 2 channels of IRC to correspond up to 32 factors of interrupt request
- Determination of IRQ interrupt priority to transfer to ARM926EJ-S
- Enable/Mask of extension IRQ interrupt
- Extension IRQ vector address display
- Supply of returning signal from stop mode to CRG (clock/reset generator)
- Capability of issuing software interrupt (IRC0_IRQ30/IRC0_FIQ) by register access

7.3. Block diagram

Figure 7-1 shows IRC block diagram and detail of interrupt request signal connection.

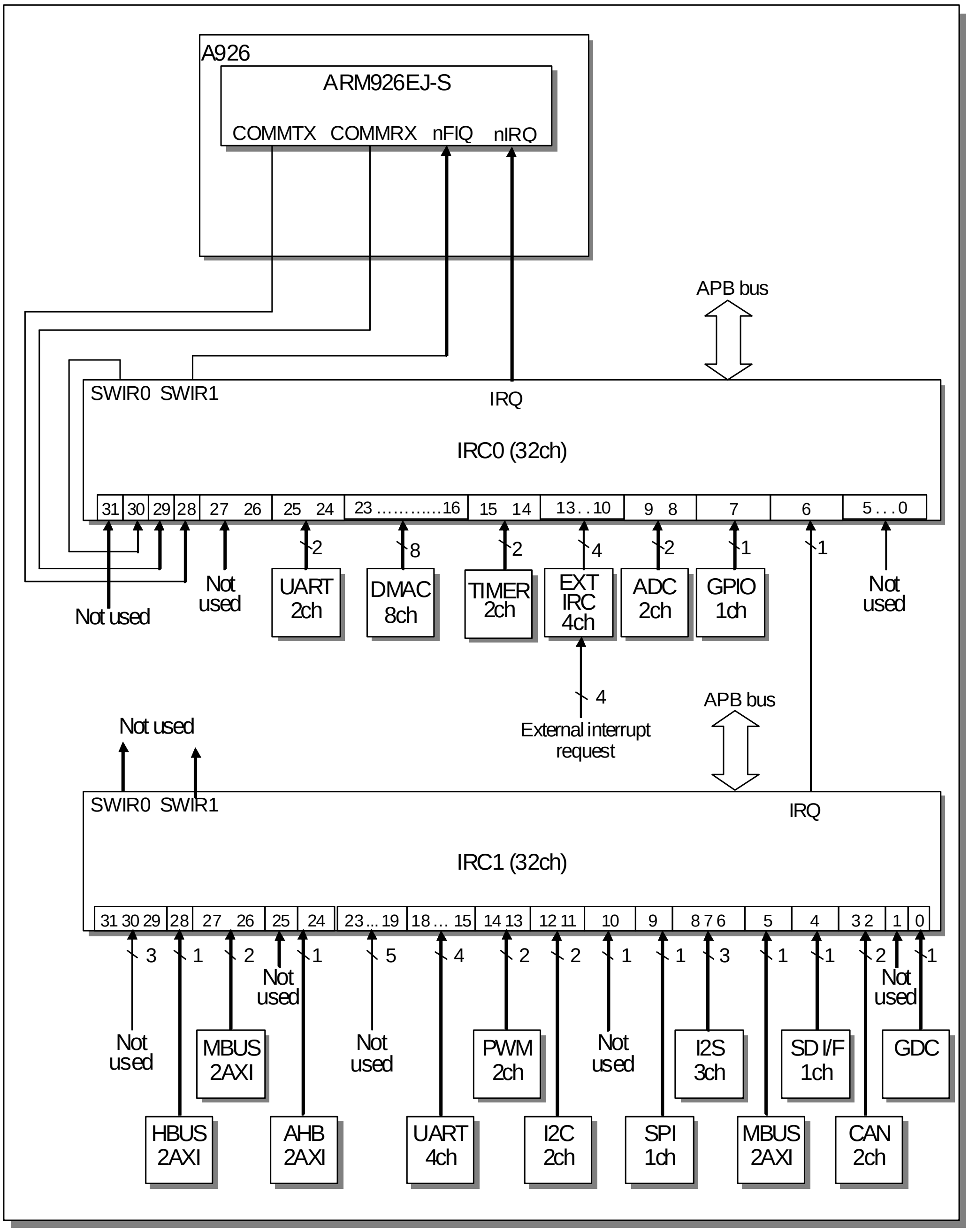


Figure 7-1 Block diagram of IRC

7.4. Supply clock

APB clock is supplied to IRC. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

7.5. Interrupt map

This section describes interrupt map.

7.5.1. Exception vector to ARM926EJ-S core

Table 7-1 shows exception vector defined in the ARM926EJ-S core.
Each interrupt factor input to IRC is notified as final interrupt of either IRQ (0000_0018_{H}/ $FFFF\_0018_{H}$) or FIQ (0000_001C_{H}/ $FFFF\_001C_{H}$) to the core.

Table 7-1 Exception vector defined by ARM926EJ-S

Exception factor	Mode	Vector address At low vector/high vector
Reset	SVC	0000_0000_{H}/$FFFF\_0000_{H}$
Undefined instruction	UND	0000_0004_{H}/$FFFF\_0004_{H}$
Software interrupt	SVC	0000_0008_{H}/$FFFF\_0008_{H}$
Prefetch abort (memory fault at instruction fetch)	Abort	0000_000C_{H}/$FFFF\_000C_{H}$
Data abort (memory fault at data access)	Abort	0000_0010_{H}/$FFFF\_0010_{H}$
Reserved	–	0000_0014_{H}/$FFFF\_0014_{H}$
IRQ (normal) interrupt	IRQ	0000_0018_{H}/$FFFF\_0018_{H}$
FIQ (high speed) interrupt	FIQ	0000_001C_{H}/$FFFF\_001C_{H}$

7.5.2. Extension IRQ interrupt vector of IRC0/IRC1IRC0/IRC1

Table 7-2 and Table 7-3 show IRQ interrupt vector extended by IRC0/IRC1. Base address of the extension vector table is determined with IRC's TBR register.

Table 7-2 Expansion IRQ interrupt vector of IRC0

Exception factor	IRQ interrupt No.		Interrupt control register (level setting)	Correction value	TBR address + correction value (at TBR=0000_0000H)
	Decimal notation	Hexadecimal notation			
IRQ0 (Unused) | IRQ5 (Unused)	0 | 5	00H | 05H	ICR00 | ICR05	20H | 34H	0000_0020H | 0000_0034H
IRQ6 (IRC1 interrupt)	6	06H	ICR06	38H	0000_0038H
IRQ7 (GPIO interrupt)	7	07H	ICR07	3CH	0000_003CH
IRQ8 (ADC ch0 interrupt)	8	08H	ICR08	40H	0000_0040H
IRQ9 (ADC ch1 interrupt)	9	09H	ICR09	44H	0000_0044H
IRQ10 (External interrupt 0)	10	0AH	ICR10	48H	0000_0048H
IRQ11 (External interrupt 1)	11	0BH	ICR11	4CH	0000_004CH
IRQ12 (External interrupt 2)	12	0CH	ICR12	50H	0000_0050H
IRQ13 (External interrupt 3)	13	0DH	ICR13	54H	0000_0054H
IRQ14 (Timer ch0 interrupt)	14	0EH	ICR14	58H	0000_0058H
IRQ15 (Timer ch1 interrupt)	15	0FH	ICR15	5CH	0000_005CH
IRQ16 (DMAC ch0 interrupt)	16	10H	ICR16	60H	0000_0060H
IRQ17 (DMAC ch1 interrupt)	17	11H	ICR17	64H	0000_0064H
IRQ18 (DMAC ch2 interrupt)	18	12H	ICR18	68H	0000_0068H
IRQ19 (DMAC ch3 interrupt)	19	13H	ICR19	6CH	0000_006CH
IRQ20 (DMAC ch4 interrupt)	20	14H	ICR20	70H	0000_0070H
IRQ21 (DMAC ch5 interrupt)	21	15H	ICR21	74H	0000_0074H
IRQ22 (DMAC ch6 interrupt)	22	16H	ICR22	78H	0000_0078H
IRQ23 (DMAC ch7 interrupt)	23	17H	ICR23	7CH	0000_007CH
IRQ24 (UART ch0 interrupt)	24	18H	ICR24	80H	0000_0080H
IRQ25 (UART ch1 interrupt)	25	19H	ICR25	84H	0000_0084H
IRQ26 (Unused)	26	1AH	ICR26	88H	0000_0088H
IRQ27 (Unused)	27	1BH	ICR27	8CH	0000_008CH
IRQ28 (COMMRX interrupt)	28	1CH	ICR28	90H	0000_0090H
IRQ29 (COMMTX interrupt)	29	1DH	ICR29	94H	0000_0094H
IRQ30 (Delay interrupt 0)	30	1EH	ICR30	98H	0000_0098H
IRQ31 (Unused)	31	1FH	ICR31	9CH	0000_009CH

Table 7-3 Extension IRQ interrupt vector of IRC1

Exception factor	IRQ interrupt No.		Interrupt control register (level setting)	Correction value	TBR address + correction value (at TBR=0000_0100$_H$)
	Decimal notation	Hexadecimal notation			
IRQ0 (GDC interrupt)	0	00$_H$	ICR00	20$_H$	0000_0120$_H$
IRQ1 (Unused)	1	01$_H$	ICR01	24$_H$	0000_0124$_H$
IRQ2 (CAN ch0 interrupt)	2	02$_H$	ICR02	28$_H$	0000_0128$_H$
IRQ3 (CAN ch1 interrupt)	3	03$_H$	ICR03	2C$_H$	0000_012C$_H$
IRQ4 (SD I/F interrupt)	4	04$_H$	ICR04	30$_H$	0000_0130$_H$
IRQ5 (MBUS2AXI (Cap) interrupt)	5	05$_H$	ICR05	34$_H$	0000_0134$_H$
IRQ6 (I2S ch0 interrupt)	6	06$_H$	ICR06	38$_H$	0000_0138$_H$
IRQ7 (I2S ch1 interrupt)	7	07$_H$	ICR07	3C$_H$	0000_013C$_H$
IRQ8 (I2S ch2 interrupt)	8	08$_H$	ICR08	40$_H$	0000_0140$_H$
IRQ9 (SPI interrupt)	9	09$_H$	ICR09	44$_H$	0000_0144$_H$
IRQ10 (Unused)	10	0A$_H$	ICR10	48$_H$	0000_0148$_H$
IRQ11 (I2C ch0 interrupt)	11	0B$_H$	ICR11	4C$_H$	0000_014C$_H$
IRQ12 (I2C ch1 interrupt)	12	0C$_H$	ICR12	50$_H$	0000_0150$_H$
IRQ13 (PWM ch0 interrupt)	13	0D$_H$	ICR13	54$_H$	0000_0154$_H$
IRQ14 (PWM ch1 interrupt)	14	0E$_H$	ICR14	58$_H$	0000_0158$_H$
IRQ15 (UART ch2 interrupt)	15	0F$_H$	ICR15	5C$_H$	0000_015C$_H$
IRQ16 (UART ch3 interrupt)	16	10$_H$	ICR16	60$_H$	0000_0160$_H$
IRQ17 (UART ch4 interrupt)	17	11$_H$	ICR17	64$_H$	0000_0164$_H$
IRQ18 (UART ch5 interrupt)	18	12$_H$	ICR18	68$_H$	0000_0168$_H$
IRQ19 (Unused)	19	13$_H$	ICR19	6C$_H$	0000_016C$_H$
IRQ20 (Unused)	20	14$_H$	ICR20	70$_H$	0000_0170$_H$
IRQ21 (Unused)	21	15$_H$	ICR21	74$_H$	0000_0174$_H$
IRQ22 (Unused)	22	16$_H$	ICR22	78$_H$	0000_0178$_H$
IRQ23 (Unused)	23	17$_H$	ICR23	7C$_H$	0000_017C$_H$
IRQ24 (AHB2_AXI (AHBBUS) interrupt)	24	18$_H$	ICR24	80$_H$	0000_0180$_H$
IRQ25 (Unused)	25	19$_H$	ICR25	84$_H$	0000_0184$_H$
IRQ26 (MBUS2AXI (Disp) interrupt)	26	1A$_H$	ICR26	88$_H$	0000_0188$_H$
IRQ27 (MBUS2AXI (Draw) interrupt)	27	1B$_H$	ICR27	8C$_H$	0000_018C$_H$
IRQ28 (HBUS2AXI interrupt)	28	1C$_H$	ICR28	90$_H$	0000_0190$_H$
IRQ29 (Unused)	29	1D$_H$	ICR29	94$_H$	0000_0194$_H$
IRQ30 (Unused)	30	1E$_H$	ICR30	98$_H$	0000_0198$_H$
IRQ31 (Unused)	31	1F$_H$	ICR31	9C$_H$	0000_019C$_H$

7.6. Register

This section describes IRC register.

7.6.1. Register list

Table 7-4 shows IRC0 register list and Table 7-5 shows IRC1 register list.

Table 7-4 IRC0 register list

Address		Register name	Abbreviation	Description
Base	Offset			
FFFF_FE00$_H$ or FFFE_8000$_H$	+ 00$_H$	IRQ flag register	IR0IRQF	IRQ interrupt flag control
	+ 04$_H$	IRQ mask register	IR0IRQM	IRQ interrupt asserted mask control
	+ 08$_H$	Interrupt level mask register	IR0ILM	Valid interrupt level setting from ARM core
	+ 0C$_H$	ICR monitoring register	IR0ICRMN	Current IRQ interrupt source’s interrupt level display
	+ 10$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 14$_H$	Delay interrupt register 0	IR0DICR0	Delay interrupt control for task switch
	+ 18$_H$	Delay interrupt register 1	IR0DICR1	Delay interrupt control
	+ 1C$_H$	Table base register	IR0TBR	High order address (24 bit) setting of IRQ vector
	+ 20$_H$	Interrupt vector register	IR0VCT	Interrupt vector table display
	+ 24$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 28$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 2C$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 30$_H$	Interrupt control register 0	IR0ICR00	IRQ0 interrupt level setting (unused and access prohibited)
	+ 34$_H$	Interrupt control register 1	IR0ICR01	IRQ1 interrupt level setting (unused and access prohibited)
	+ 38$_H$	Interrupt control register 2	IR0ICR02	IRQ2 interrupt level setting (unused and access prohibited)
	+ 3C$_H$	Interrupt control register 3	IR0ICR03	IRQ3 interrupt level setting (unused and access prohibited)
	+ 40$_H$	Interrupt control register 4	IR0ICR04	IRQ4 interrupt level setting (unused and access prohibited)
	+ 44$_H$	Interrupt control register 5	IR0ICR05	IRQ5 interrupt level setting (unused and access prohibited)
	+ 48$_H$	Interrupt control register 6	IR0ICR06	IRQ6 interrupt level setting (IRC1 interrupt)
	+ 4C$_H$	Interrupt control register 7	IR0ICR07	IRQ7 interrupt level setting (GPIO interrupt)
	+ 50$_H$	Interrupt control register 8	IR0ICR08	IRQ8 interrupt level setting (ADC ch0 interrupt)
	+ 54$_H$	Interrupt control register 9	IR0ICR09	IRQ9 interrupt level setting (ADC ch1 interrupt)
	+ 58$_H$	Interrupt control register 10	IR0ICR10	IRQ10 interrupt is set (external interrupt 0)
	+ 5C$_H$	Interrupt control register 11	IR0ICR11	IRQ11 interrupt level setting (external interrupt 1)
	+ 60$_H$	Interrupt control register 12	IR0ICR12	IRQ12 interrupt level setting (external interrupt 2)
	+ 64$_H$	Interrupt control register 13	IR0ICR13	IRQ13 interrupt level setting (external interrupt 3)
	+ 68$_H$	Interrupt control register 14	IR0ICR14	IRQ14 interrupt level setting (timer ch0 interrupt)
	+ 6C$_H$	Interrupt control register 15	IR0ICR15	IRQ15 interrupt level setting (timer ch1 interrupt)
	+ 70$_H$	Interrupt control register 16	IR0ICR16	IRQ16 interrupt level setting (DMAC ch0 interrupt)
	+ 74$_H$	Interrupt control register 17	IR0ICR17	IRQ17 interrupt level setting (DMAC ch1 interrupt)
	+ 78$_H$	Interrupt control register 18	IR0ICR18	IRQ18 interrupt level setting (DMAC ch2 interrupt)
	+ 7C$_H$	Interrupt control register 19	IR0ICR19	IRQ19 interrupt level setting (DMAC ch3 interrupt)
	+ 80$_H$	Interrupt control register 20	IR0ICR20	IRQ20 interrupt level setting (DMAC ch4 interrupt)
	+ 84$_H$	Interrupt control register 21	IR0ICR21	IRQ21 interrupt level setting (DMAC ch5 interrupt)
	+ 88$_H$	Interrupt control register 22	IR0ICR22	IRQ22 interrupt level setting (DMAC ch6 interrupt)
	+ 8C$_H$	Interrupt control register 23	IR0ICR23	IRQ23 interrupt level setting (DMAC ch7 interrupt)
	+ 90$_H$	Interrupt control register 24	IR0ICR24	IRQ24 interrupt level setting (UART ch0 interrupt)
	+ 94$_H$	Interrupt control register 25	IR0ICR25	IRQ25 interrupt level setting (UART ch0 interrupt)
	+ 98$_H$	Interrupt control register 26	IR0ICR26	IRQ26 interrupt level setting (unused and access prohibited)
	+ 9C$_H$	Interrupt control register 27	IR0ICR27	IRQ27 interrupt level setting (unused and access prohibited)
	+ A0$_H$	Interrupt control register 28	IR0ICR28	IRQ28 interrupt level setting (COMMRX interrupt)
	+ A4$_H$	Interrupt control register 29	IR0ICR29	IRQ29 interrupt level setting (COMMTX interrupt)
	+ A8$_H$	Interrupt control register 30	IR0ICR30	IRQ30 interrupt level setting (delay interrupt)
	+ AC$_H$	Interrupt control register 31	IR0ICR31	IRQ31 interrupt level setting (unused and access prohibited)

Table 7-5 IRC1 register list

Address		Register name	Abbreviation	Description
Base	Offset			
FFFB_0000$_H$	+ 00$_H$	IRQ flag register	IR1IRQF	IRQ interrupt flag control
	+ 04$_H$	IRQ mask register	IR1IRQM	IRQ interrupt asserted mask control
	+ 08$_H$	Interrupt level mask register	IR1ILM	Valid interrupt level setting from ARM core
	+ 0C$_H$	ICR monitoring register	IR1ICRMN	Current IRQ interrupt source’s interrupt level display
	+ 10$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 14$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 18$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 1C$_H$	Table base register	IR1TBR	IRQ vector’s high order address (24 bit) setting
	+ 20$_H$	Interrupt vector register	IR1VCT	Interrupt vector table display。
	+ 24$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 28$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 2C$_H$	(Reserved)	–	Reserved (access prohibited)
	+ 30$_H$	Interrupt control register 0	IR1ICR00	IRQ0 interrupt level setting (GDC interrupt)
	+ 34$_H$	Interrupt control register 1	IR1ICR01	IRQ1 interrupt level setting (unused and access prohibited)
	+ 38$_H$	Interrupt control register 2	IR1ICR02	IRQ2 interrupt level setting (CAN ch0 interrupt)
	+ 3C$_H$	Interrupt control register 3	IR1ICR03	IRQ3 interrupt level setting (CAN ch1 interrupt)
	+ 40$_H$	Interrupt control register 4	IR1ICR04	IRQ4 interrupt level setting (SD I/F interrupt)
	+ 44$_H$	Interrupt control register 5	IR1ICR05	IRQ5 interrupt level setting (MBUS2AXI (Cap) interrupt)
	+ 48$_H$	Interrupt control register	IR1ICR06	IRQ6 interrupt level setting (I2S ch0 interrupt)
	+ 4C$_H$	Interrupt control register 7	IR1ICR07	IRQ7 interrupt level setting (I2S ch1 interrupt)
	+ 50$_H$	Interrupt control register 8	IR1ICR08	IRQ8 interrupt level setting (I2S ch2 interrupt)
	+ 54$_H$	Interrupt control register 9	IR1ICR09	IRQ9 interrupt level setting (SPI interrupt)
	+ 58$_H$	Interrupt control register 10	IR1ICR10	IRQ10 interrupt level setting (unused and access prohibited)
	+ 5C$_H$	Interrupt control register 11	IR1ICR11	IRQ11 interrupt level setting (I^2C ch0 interrupt)
	+ 60$_H$	Interrupt control register 12	IR1ICR12	IRQ12 interrupt level setting (I^2C ch1 interrupt)
	+ 64$_H$	Interrupt control register 13	IR1ICR13	IRQ13 interrupt level setting (PWM ch0 interrupt)
	+ 68$_H$	Interrupt control register 14	IR1ICR14	IRQ14 interrupt level setting (PWM ch1 interrupt)
	+ 6C$_H$	Interrupt control register 15	IR1ICR15	IRQ15 interrupt level setting (UART ch2 interrupt)
	+ 70$_H$	Interrupt control register 16	IR1ICR16	IRQ16 interrupt level setting (UART ch3 interrupt)
	+ 74$_H$	Interrupt control register 17	IR1ICR17	IRQ17 interrupt level setting (UART ch4 interrupt)
	+ 78$_H$	Interrupt control register 18	IR1ICR18	IRQ18 interrupt level setting (UART ch5 interrupt)
	+ 7C$_H$	Interrupt control register 19	IR1ICR19	IRQ19 interrupt level setting (unused and access prohibited)
	+ 80$_H$	Interrupt control register 20	IR1ICR20	IRQ20 interrupt level setting (unused and access prohibited)
	+ 84$_H$	Interrupt control register 21	IR1ICR21	IRQ21 interrupt level setting (unused and access prohibited)
	+ 88$_H$	Interrupt control register 22	IR1ICR22	IRQ22 interrupt level setting (unused and access prohibited)
	+ 8C$_H$	Interrupt control register 23	IR1ICR23	IRQ23 interrupt level setting (unused and access prohibited)
	+ 90$_H$	Interrupt control register 24	IR1ICR24	IRQ24 interrupt level setting (AHB2_AXI (AHBBUS) interrupt)
	+ 94$_H$	Interrupt control register 25	IR1ICR25	IRQ25 interrupt level setting (unused and access prohibited)
	+ 98$_H$	Interrupt control register 26	IR1ICR26	IRQ26 interrupt level setting (MBUS2AXI (Disp) interrupt)
	+ 9C$_H$	Interrupt control register 27	IR1ICR27	IRQ27 interrupt level setting (MBUS2AXI (Draw) interrupt)
	+ A0$_H$	Interrupt control register 28	IR1ICR28	IRQ28 interrupt level setting (HBUS2AXI interrupt)
	+ A4$_H$	Interrupt control register 29	IR1ICR29	IRQ29 interrupt level setting (unused and access prohibited)
	+ A8$_H$	Interrupt control register 30	IR1ICR30	IRQ30 interrupt level setting (unused and access prohibited)
	+ AC$_H$	Interrupt control register 31	IR1ICR31	IRQ31 interrupt level setting (unused and access prohibited)

Description format of register

Following format is used for description of register's each bit in "7.6.2 IRQ flag register (IR0IRQF/IR1IRQF)" to "7.6.10 Interrupt control register (IR0ICR31/IR1ICR31 – IR0ICR00/IR1ICR00)".

Address	Base address + Offset address															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

7.6.2. IRQ flag register (IR0IRQF/ IR1IRQF)

IR0IRQF/IR1IRQF registers control IRQ interrupt flag.
When interrupt level is higher than the one set in IR0ILM/IR1ILM registers as a result of determining IRQ interrupt source level, IRQF bit is set and IRQ interrupt is asserted to ARM core.
The interruption to ARM core is negated with "0" writing to the IR0IRQF/IR1IRQF registers.
When IRQF bit is set, interrupt vector is displayed to IR0VCT/IR1VCT registers but its address value is not changed until IRQF bit is set.

Address	**IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 00$_H$**								**IRC1: FFFB_0000$_H$ + 00$_H$**							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	IRQF
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field		**Description**
No.	**Name**	
31-1	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
0	IRQF	IRQ interrupt flag. When interrupt level is higher than the one set in IR0ILM/IR1ILM registers (interrupt level in IR0ICR/IR1ICR registers > interrupt level in IR0ILM/IR1ILM registers), IRQF bit is set to "1" and IRQX (interrupt request) is asserted to ARM core. 0: IRQ is not asserted. 1: IRQ is asserted. This bit is cleared by writing "0", and writing "1" is invalid.

7.6.3. IRQ mask register (IR0IRQM/IR1IRQM)

IR0IRQM/IR1IRQM registers control masking asserted IRQ interrupt.

Address	IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 04$_H$								IRC1: FFFB_0000$_H$ + 04$_H$							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	IRQM
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field		Description
No.	**Name**	
31-1	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
0	IRQM	Asserted IRQ interrupt is masked. 0: Asserted IRQ is masked 1: Asserted IRQ is valid This bit is initialized to "0" by reset.

7.6.4. Interrupt level mask register (IR0ILM/IR1ILM)

IR0ILM/IR1ILM registers set interrupt level enabled by the ARM core. When the IRQ interrupt source is larger than the setting value of this register, IRC notifies the ARM core of the IRQ interrupt.

"Interrupt level of IR0ICR/IR1ICR registers > Interrupt enable level of IR0ILM/IR1ILM registers" -> Generated IRQ interrupt

Address	IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 08$_H$								IRC1: FFFB_0000$_H$ + 08$_H$							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	ILM3	ILM2	ILM1	ILM0
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	1	1	1	1

Bit field		Description
No.	Name	
31-4	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
3-0	ILM3-0	These bits are used to set IRQ interrupt mask level. Its range is from 0000$_B$ the highest to 1111$_B$ the lowest. When 0000$_B$ (highest level) is set, all interrupt requests are masked. These bits are initialized to 1111$_B$ by reset.

7.6.5. ICR monitoring register (IR0ICRMN/IR1ICRMN)

IR0ICRMN/IR1ICRMN registers display interrupt level of the current IRQ interrupt source.
If IRQ interrupt source is less than the setting value of these registers, 1111_B is displayed, and for the case that IRQ interrupt transmission source is larger than the setting value, the highest interrupt source level is displayed.
These registers are updated with setting IRQF bit of IR0IRQF/IR1IRQF "1", and displayed interrupt level is not changed until IRQF bit is cleared.
After it is cleared, interrupt level is set again and the display is updated with the source set the IRQF bit. Register value is not defined if the bit is not set to "1".

Address	**IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 0C$_H$**								**IRC1: FFFB_0000$_H$ + 0C$_H$**							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	ICRMN3	ICRMN2	ICRMN1	ICRMN0
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		**Description**
No.	**Name**	
31-4	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
3-0	ILM3-0	When IRQ interrupt source is larger than the setting value of IR0ILM/IR1ILM registers, the highest interrupt source level is displayed. The initial value of these bits is undefined.

7.6.6. Delay interrupt control register 0 (IR0DICR0)

IR0DICR0 register controls delay interrupt for the task switch.
Writing to this register enables software to issue/cancel IRQ interrupt request.
The delay interrupt is allocated into IRQ30 of IRC0.

Address	**IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 14$_H$**								**IRC1: Reserved area FFFB_0000$_H$ + 14$_H$**							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	DLYI0
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field		**Description**
No.	**Name**	
31-1	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
0	DLYI0	Delay interrupt is controlled. It is cancelled by writing "0" to this bit. 0: Delay interrupt factor is cancelled and interrupt request does not occur. 1: Delay interrupt factor is generated and interrupt request occurs. This bit is initialized to "0" by reset.

7.6.7. Delay interrupt control register 1 (IR0DICR1)

Writing to IR0DICR1 register enables software to issue/cancel FIQ interrupt request.
The delay interrupt is allocated into FIQ of the ARM.

Address	IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 18$_H$								IRC1: Reserved area FFFB_0000$_H$ + 18$_H$							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	DLYI1
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field No.	Bit field Name	Description
31-1	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
0	DLYI1	Delay interrupt is controlled. It is cancelled by writing "0" to this bit. 0: Delay interrupt factor is cancelled but interrupt request does not occur 1: Delay interrupt factor is generated and interrupt request occurs This bit is initialized to "0" by reset.

7.6.8. Table base register (IR0TBR/IR1TBR)

IR0TBR/IR1TBR registers indicate upper address (24 bit) of IRQ vector. When IRC receives IRQ interrupt source, and IRQ is asserted to the ARM core, the address displayed in IR0VCT/IR1VCT registers are as follows.

(IR0TBR/IR1TBR setting value) + Individual IRQ interrupt source vector address

Address	IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 1C$_H$								IRC1: FFFB_0000$_H$ + 1C$_H$							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TBR31	TBR30	TBR29	TBR28	TBR27	TBR26	TBR25	TBR24	TBR23	TBR22	TBR21	TBR20	TBR19	TBR18	TBR17	TBR16
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TBR15	TBR14	TBR13	TBR12	TBR11	TBR10	TBR9	TBR8	Zero	Zero	Zero	Zero	Zero	Zero	Zero	Zero
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-8	TBR31-8	Set upper address (24 bit) of IRQ vector. These bits are initialized to "0" by reset.
7-0	Zero	"0" fixed bit. Writing is invalid and "0" is always read in the read value. These bits are initialized to "0" by reset.

7.6.9. Interrupt vector register (IR0VCT/IR1VCT)

IR0VCT/IR1VCT registers display interrupt vector table to the interrupt source to be processed when IRQ is asserted to ARM core ("1" is set to IRQF bit of IR0IRQF/IR1IRQF registers.)
The priority of vector address is as follows.

- The highest interrupt source vector level in the generated IRQ interrupt source has higher priority
- When interrupt of same level and transmission source occurs at the same time, the one with less address offset value is prioritized

Address	IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 20$_H$								IRC1: FFFB_0000$_H$ + 20$_H$							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VCT31	VCT30	VCT29	VCT28	VCT27	VCT26	VCT25	VCT24	VCT23	VCT22	VCT21	VCT20	VCT19	VCT18	VCT17	VCT16
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VCT15	VCT14	VCT13	VCT12	VCT11	VCT10	VCT9	VCT8	VCT7	VCT6	VCT5	VCT4	VCT3	VCT2	VCT1	VCT0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		Description
No.	Name	
31-0	VCT31-0	Interrupt vector table is displayed to the interrupt source to be processed. The value adding each interrupt factor’s offset value to upper address value set by IR0TBR/IR1TBR registers is displayed as vector value. Refer to "Table 7-2 Expansion IRQ interrupt vector of IRC0" and "Table 7-3 Extension IRQ interrupt vector of IRC1" for correlation of interrupt source, interrupt level register, and vector address. The initial value of these bits is undefined.

After IRQF bit of IR0IRQF/IR1IRQF registers is set to "1", the displayed vector address value is not changed until the IRQF bit is cleared. When the bit is cleared, interrupt level is set again and the display is updated by the source that sets the IRQF bit. Register value is not defined if the bit is not set to "1".

Firmware branches into the address specified by VCT register (branched to extension vector table) with the instruction in IRQ vector (0000_0018$_H$). Then it branches into interrupt handler by the instruction on the address. If IRQF bit is cleared after the branch, asserting IRQ enables to observe whether new IRQ source is higher than the current one in the interrupt handler.

7.6.10. Interrupt control register (IR0ICR31/IR1ICR31 – IR0ICR00/IR1ICR00)

IR0ICR31/IR1ICR31 − IR0ICR00/IR1ICR00 registers are supplied to each IRQ interrupt source, and are able to set interrupt level to the corresponding IRQ interrupt source. When IRQ interrupt source is larger than the setting value of IR0ILM/IR1ILM registers (interrupt level of IR0ICRn/IR1ICRn registers <= interrupt level of IR0ILM/IR1ILM registers), it is masked.

Address	**IRC0: FFFF_FE00$_H$ or FFFE_8000$_H$ + 30$_H$ | FFFF_FE00$_H$ or FFFE_8000$_H$ + AC$_H$**								**IRC1: FFFB_0000$_H$ + 30$_H$ | FFFB_0000$_H$ + AC$_H$**							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	ICR3	ICR2	ICR1	ICR0
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	1	1	1	1

Bit field		**Description**
No.	**Name**	
31-4	–	Unused bit. The write access is ignored. The read value of these bits is undefined.
3-0	ICR3-0	These bits are used to set interrupt level value of each interrupt source. Its range is from "0000$_B$" the highest to "1111$_B$" the lowest.

ICR3	ICR3	ICR1	ICR0	Interrupt level
0	0	0	0	Settable highest level
0	0	0	1	(highest)
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	(lowest)
1	1	1	1	Uninterruptible

These bits are initialized to "1111$_B$" by reset.

7.7. Operation

This section describes IRC operation.

7.7.1. Outline

Interrupt operation process is described with using IRQ24 interrupt as an example.

1. When IRQ interrupt is asserted to ARM core as a result of prioritization of IRQ24 interrupt source with interrupt controller, the ARM core refers instruction of vector address 0000_0018_H.
2. Loading instruction, **LDR PC, [PC, #_0×200]** is written to vector table address 0000_0018_H beforehand. Then extension interruption vector address of IRQ24 (VCT register value) is loaded into PC, and the ARM core refers IRQ24 vector address of extension interrupt vector table.
3. Branch instruction to the IRQ24 interrupt handler should be written to IRQ24 extension interrupt vector address. Then PC branches into the IRQ24 interrupt handler with the branch instruction. All interrupt handlers should be set within ±32MB of the extension interrupt vector table in order to use the branch instruction. If the handler is unable to be set in the range, use load instruction, **LDR PC, [PC, #_0x200]** instead.

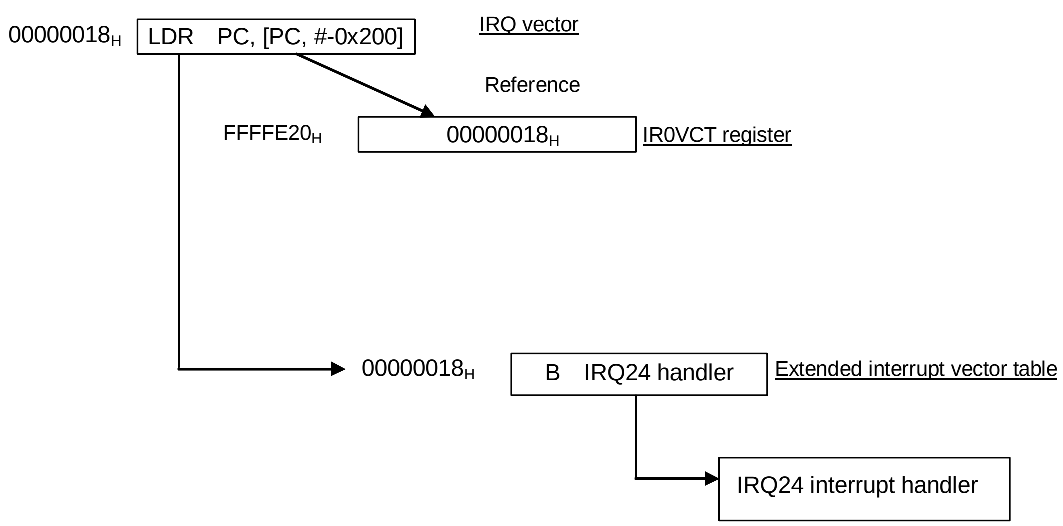


Figure 7-2 IRQ24 interrupt process example

7.7.2. Initialization

1. Determine individual exception table after power-on.
2. Set extension interrupt vector table.
3. Store load instruction, **LDR PC, [PC, #_0x200]** to IRQ vector (00000018_H) in the ARM core.
4. Set base address of the interrupt table to IR0TBR register.
5. Set interrupt level of each interrupt source to IR0ICR31 - 00 registers.
6. Set interrupt level that IRQ interrupt becomes valid to the IR0ILM register.
7. Set I flag of CPSRs register in the ARM core to "0" (to validate IRQ.)
8. Validate interrupt with IR0IRQM register in IRC.

7.7.3. Multiple interrupt process

Example of multiple interrupt process is shown below.

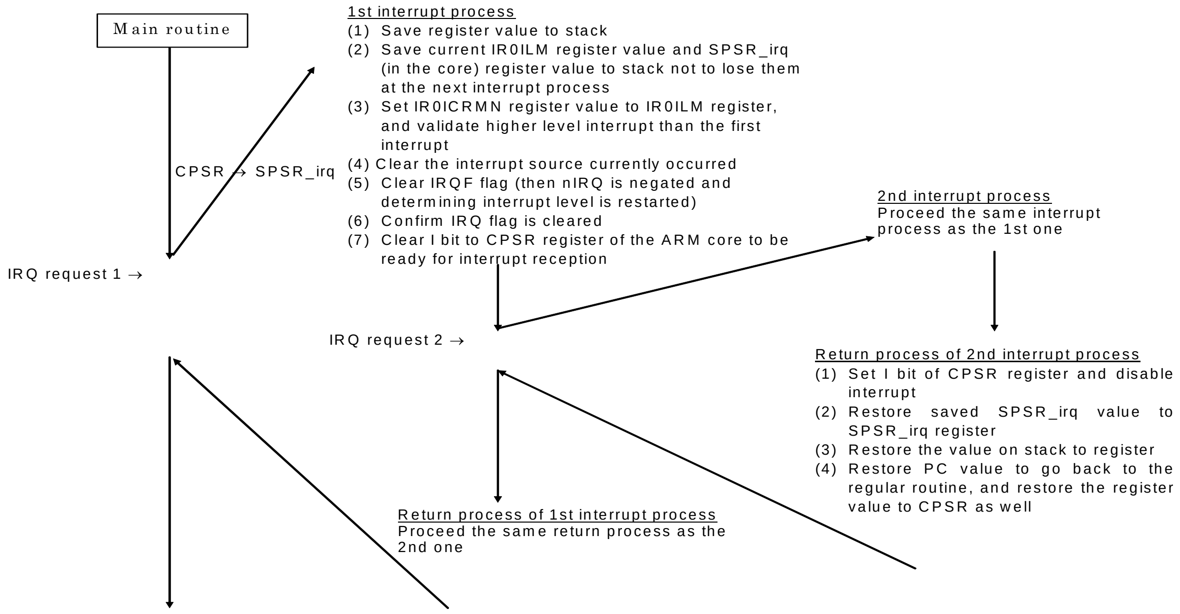


Figure 7-3 Example of multiple IRQ interrupt process

7.7.4. Example of IRQ interrupt handler

IRQ_Handler ROUT

```
STMFD   SP!, {R0-R12, R14}              ;Save register value
MESSAGE "Enter Dummy IRQ Handler"

LDR  R0, = IR0ILM
LDR  R1, [R0]
MRS R2, SPSR
STMFD        SP!, {R1, R2}              ;IR0ILM and SPSR_irq register values are saved

LDR  R2, = IR0ICRMN
LDR  R1, [R2]
STR  R1, [R0]                           ;IR0ICRMN register value is set to IR0ILM register
```

Routine to clear interrupt factor

```
  MOV      R1, #0
  LDR      R0, = IR0IRQF
  STR      R1, [R0];                ;Clear IRQF bit (bit 0) of IR0IRQF register
                                    ;Start the next interrupt level setting operation.
LOOP
  LDR      R1,[R0]                  ;Check IRQF flag clear
  CMP      R1,#0
  BNE LOOP
  ;; Clear ARM IRQ Flag → Enable Interrupt
  MRS      R2, CPSR
  BIC      R2, R2, #I_Bit
  MSR      CPSR_c, R2;              ;Clear I bit of CPSR register (included in the core) and
                                     validate IRQ interrupt (enable)
```

If the IRQ interrupt higher than the current IRQ source occurs, move to the corresponding interrupt handler.

Main routine for this interrupt factor

```
MRS R2, CPSR
ORR R2, R2 #1_Bit
MSR CPSR_c, R2;                     ;Set I bit of CPSR register (included in the core) and
                                     invalidate IRQ interrupt (disable)
LDR  R0, = IR0ILM
LDMFD SP!, {R1, R2}
MSR SPSR_cxsf, R2
STR  R1, [R0];                      ;Resume saved value in IR0ILM and SPSR_irq registers
                                     (included in the core)
LDMFD      SP! {R0-R12, R14};       ;Resume register value
SUBS       PC, R14, #4;             ;CPSR < - SPSR_irq, PC < - R14 –4
```

7.7.5. Resume from Stop and standby modes

Resume from stop and standby modes is able to be instructed to CRG (Clock Reset Controller) with issuing IRQ interrupt from macro.

The resume signal from stop and standby modes, asserted to ARM clock controller is generated by higher IRQ factor than the interrupt level set with IR0ILM register (see Figure 7-1.)

7.7.6. Notice for using IRC

Notice for using IRC is shown below.

Notice for IRQ clear timing

As described in "7.6.2 IRQ flag register (IR0IRQF/ IR1IRQF)", "0" writing to IRQF bit of IR0IRQF/IR1IRQF registers negates IRQX (interrupt request) to the ARM core; however, IRQX is negated during 1 cycle of APB clock after writing "0". Therefore, the ARM core may wrongly goes into IRQ mode again by the IRQX before clear operation if the code (interrupt handler) which may validate ARM core interrupt again is written after "0" writing to the IRQF.
This might occurs especially when ARM core’s clock frequency is faster than the IRC frequency.

In order to prevent such problem, add dummy instruction which accesses to IRC interrupt register after clear instruction of IRQF. In this way, IRQX is cleared properly before interrupt of the ARM core becomes valid again.

8. External bus interface

This chapter describes external bus of MB86R03.

8.1. Outline

MB86R03 has external bus interface for accessing to external memory device such as SRAM and Flash.

8.2. Spec limitation

External bus interface supports 8 chip selects (CS0-7). However, only CS0, CS2, and CS4, which have external pin (MEM_XCS[0/2/4]) are able to be used. The others (CS1, CS3, CS5, CS6, CS7) are not usable since they do not have external pin.

While external bus interface is able to use CS0/2/4 chip selects, address area for other chip selects (CS1/3/5/6/7) are allocated in LSI during initialization (see Figure 8-1.)

0x1100_0000	CS1
0x1000_0000	
0x0F00_0000	
0x0E00_0000	
0x0D00_0000	
0x0C00_0000	
0x0B00_0000	
0x0A00_0000	
0x0900_0000	
0x0800_0000	
0x0700_0000	CS7
0x0600_0000	CS6
0x0500_0000	CS5
0x0400_0000	
0x0300_0000	CS3
0x0200_0000	

Reserved under initialing

CS1: 0x1100_0000-0x11FF_FFFF (16MB)
CS3: 0x0300_0000-0x03FF_FFFF (16MB)
CS5: 0x0500_0000-0x05FF_FFFF (16MB)
CS6: 0x0600_0000-0x06FF_FFFF (16MB)
CS7: 0x0700_0000-0x07FF_FFFF (16MB)

Figure 8-1 Initialization value of chip selection address area (except CS0/2/4) valid in LSI

If address area of CS0/2/4 and CS1/3/5/6/7 is overlapped, CS0/2/4 signals (MEM_XCS[0/2/4] pin output) may not be asserted correctly. Therefore, perform initial setting shown in the next page for using external bus interface.

Initial setting for using external bus interface

CS1/3/5/6/7 address areas should be set out of CS0/2/4 address areas with SRAM/Flash area register 1/3/5/6/7 (MCFAREA1/3/5/6/7.) (See Table 8-1.)

Table 8-1 CS1/3/5/6/7 SRAM/Flash area register 1/3/5/6/7 address and recommended setting value

Chip select	SRAM/Flash area register		Recommended setting value (Note)
	Abbreviation	Address	
CS1	MCFAREA1	0xFFFC0044	0x0000001F
CS3	MCFAREA3	0xFFFC004C	0x0000001F
CS5	MCFAREA5	0xFFFC0054	0x0000001F
CS6	MCFAREA6	0xFFFC0058	0x0000001F
CS7	MCFAREA7	0xFFFC005C	0x0000001F

Note) Since CS1/3/5/6/7 are unable to be used, the same address area is settable.

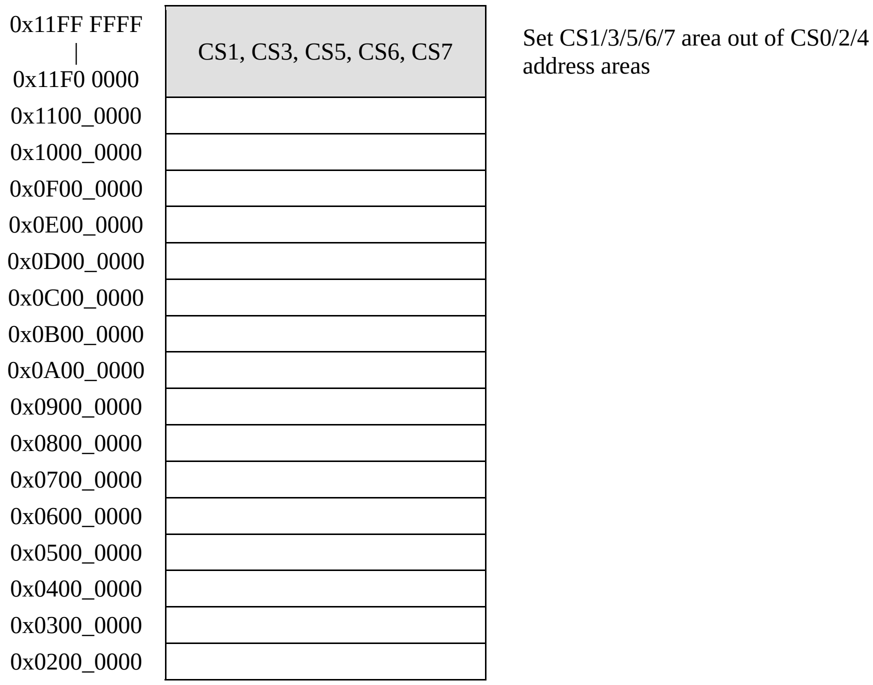


Figure 8-2 CS1/3/5/6/7 address areas

This initial setting enables CS0/2/4 address areas setting in 0x0200_0000 - 0x11EF_FFFF.
For 0x1000_0000 - 0x10FF_FFFF (external boot ROM), address area is fixed in CS4.

Remarks:

CS1/3/5/6/7 address areas are able to set other values than the one indicated in Table 8-1; in this case, make sure that address area of CS0/2/4 and CS1/3/5/6/7 addresses are not overlapped.

8.3. Feature

External bus interface of MB86R03 has the following features.

- Supporting 16/32 bit (32 bit is an option) width of SRAM/Flash
- 3 chip selects for SRAM/Flash (MEM_XCS[4] is for boot operation).
- Parameter setting by individual chip select for SRAM/Flash
- Supporting NOR flash page access
- Supporting Bi-endian

8.4. Block diagram

Figure 8-3 shows block diagram of external bus interface.

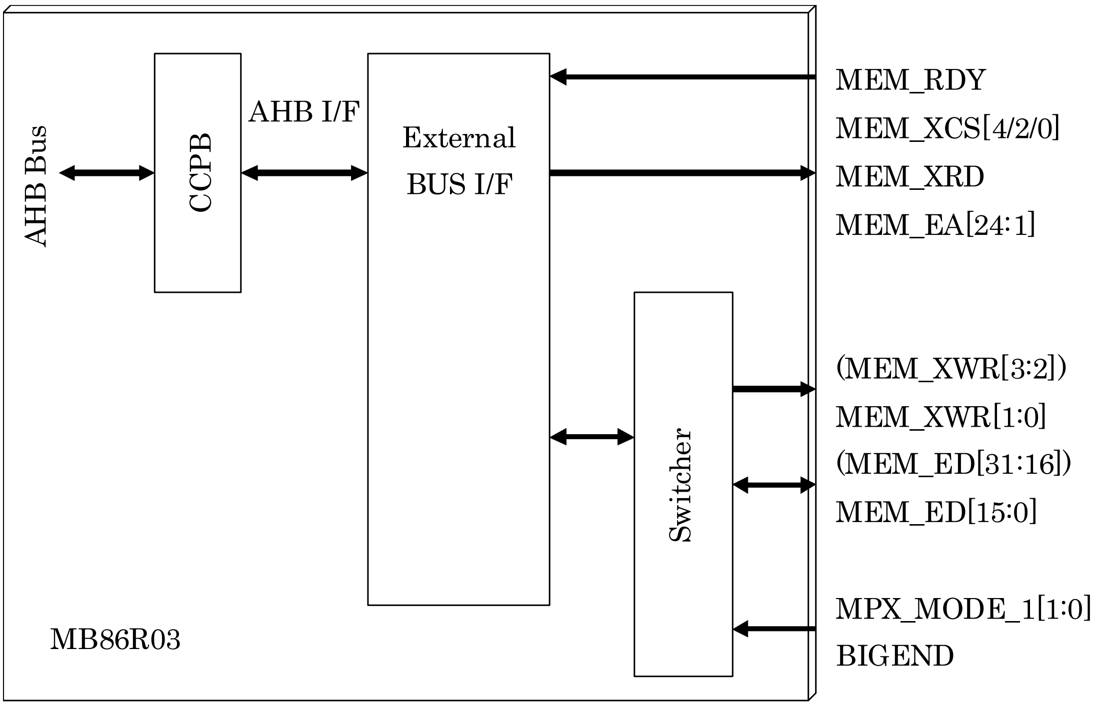


Figure 8-3 Block diagram of external bus interface part

8.5. Related pin

Table 8-2 External interface related pin

Pin	I/O	No. of pin	Function
MEM_EA[24:1]	O	24	Address bus
MEM_XWR[3:0]	O	4	Writing enabled Upper 2 bits are multiplexed pin
MEM_XRD	O	1	Reading enabled
MEM_XCS[4]	O	1	Chip select for boot operation
MEM_XCS[2]	O	1	Chip select
MEM_XCS[0]	O	1	Chip select
MEM_ED[31:0]	IO	32	Data bus Upper 16 bits are multiplexed pin
MEM_RDY	I	1	Ready input for low-speed device

8.6. Supply clock

AHB clock is supplied to external bus interface. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

8.7. Register

This section describes 32 bit width external bus I/F register.
Be sure to access to it in word (32 bit.)

8.7.1. SRAM/Flash mode register 0-7 (MCFMODE0-7)

Register address	BaseAddress+0x0000	MCFMODE0 (External pin: MEM_XCS[0])	
	BaseAddress+0x0004	MCFMODE1 External pin: N/A)	(*1)
	BaseAddress+0x0008	MCFMODE2 (External pin:MEM_XCS[2])	
	BaseAddress+0x000C	MCFMODE3 (External pin: N/A)	(*1)
	BaseAddress+0x0010	MCFMODE4 (External pin:MEM_XCS[4])	
	BaseAddress+0x0014	MCFMODE5 (External pin: N/A)	(*1)
	BaseAddress+0x0018	MCFMODE6 (External pin: N/A)	(*1)
	BaseAddress+0x001C	MCFMODE7 (External pin: N/A)	(*1)

Bit No.	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field name	Reserved															
R/W	R/W0															
Initial value	X															

Bit No.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved									RDY	PAGE	Reserved			WDTH	
R/W	R/W0									R/W	R/W	R/W0			R/W	
Initial value	X									0	0	X	X	X	0 (*2)	

*1: MCFMODE1/3/5/6/7 are access prohibited
*2: Initial value of data width to MEM_XCS[4]
MPX_MODE_1[1:0]=2'b01: 2:32 bit
Others: 1:16 bit

Bit31-7: Reserved

Reserved bits.
Write "0" to these bits. Their read value is undefined.

Bit6: RDY (ready mode)

When handshake is performed with low-speed peripherals that use MEM_RDY signal, set this bit to "1". RDY signal at reading should be asserted to "L" at least 2 cycles from 2 cycles before falling edge of MEM_XRD signal to actual falling edge. For the writing operation, the RDY signal should also be asserted to "L" at least 2 cycles from 2 cycles before falling edge of MEM_XWR signal to actual falling edge.
For accessing to device such as SRAM memory without using the MEM_RDY signal, this bit should be set to "0".

0: READY mode OFF (initial value)
1: READY mode ON

Bit5: PAGE (page access mode) NOR flash page access mode

This bit controls NOR flash page access mode which issues the first address cycle according to FirstReadAddressCycle (FRADC) setting. Then, the access is continuously executed according to Read Access Cycle (RACC) setting until it reaches to 16 byte boundary. In order to select this mode, set Read Address Cycle (RADC) to 0.

0: READY mode OFF (initial value)
1: READY mode ON

Bit4-2: Reserved

Reserved bits.

Write "0" to these bits. Their read value is undefined.

Note:

Writing "1" to these bits are prohibited.

Bit1-0: WDTH (data width)

These bits specify data bit width of the connected device.

0: 8 bit (initial value)

1: 16 bit

2: 32 bit

3: Reserved

8.7.2. SRAM/Flash timing register 0-7 (MCFTIM0-7)

Register address	BaseAddress+0x0020	MCFTIM0 (External pin: MEM_XCS[0])	
	BaseAddress+0x0024	MCFTIM1 (External pin: N/A)	(*1)
	BaseAddress+0x0028	MCFTIM2 (External pin: MEM_XCS[2])	
	BaseAddress+0x002C	MCFTIM3 (External pin: N/A)	(*1)
	BaseAddress+0x0030	MCFTIM4 (External pin: MEM_XCS[4])	
	BaseAddress+0x0034	MCFTIM5 (External pin: N/A)	(*1)
	BaseAddress+0x0038	MCFTIM6 (External pin: N/A)	(*1)
	BaseAddress+0x003C	MCFTIM7 (External pin: N/A)	(*1)

Bit No.	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field name	WIDLC				WWEC				WADC				WACC			
R/W	R/W															
Initial value	0				5				5				15			
Bit No.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	RIDLC				FRADC				RADC				RACC			
R/W	R/W															
Initial value	15				0				0				15			

*1: MCFTIM1/3/5/6/7 are access prohibited

Bit31-28: WIDLC (Write Idle Cycle: Write idle cycle)

These bits set the number of idle cycle after the write access. When RDY bit is set to "1", specify 2 or more value.

0 1 cycle (initial value)
| |
15 16 cycles

Bit27-24: WWEC (Write Enable Cycle)

These bits set the number of write enable assertion cycle. This setting also affects to MEM_XWR[3:0]. When RDY bit is set to "1", the value should be 3 or more (4 cycles or more.)

0 1 cycle
| |
5 6 cycles (initial value)
| |
14 15 cycles
15 Reserved

Bit23-20: WADC (Write Address Setup cycle)

These bits set number of write access setup cycle. Address is output to the cycle; however, write enable is not asserted. When RDY bit is set to "1", the value should be 1 or more (2 cycles or more.)

0 1 cycle
| |
5 6 cycles (initial value)
| |
14 15 cycles
15 Reserved

Bit19-16: WACC (Write Access Cycle)

These bits specify number of cycle required for write access. The address does not change during the cycle specified in these bits. The WACC value should be larger than the total number of Address Setup Cycle (WADC) and Write Enable Cycle (WWEC).

tWACC >= (tWADC+tWWEC)

When RDY bit is set to "1", the value should be 6 or more (7 cycles or more.)

0, 1 Reserved
2 3 cycles
| |
15 16 cycles (initial value)

Bit15-12: RIDLC (Read Idle Cycle)

These bits set number of idle cycle after read access. They are used to prevent data collision that occurs by write access immediately after the read access.

0 1 cycle
| |
15 16 cycles (initial value)

Bit11-8: FRADC (First Read Address Cycle)

These bits are exclusive use for NOR Flash setting that corresponds to page mode access, and are set initial latency in the address of Flash read access.
The address is retained with number of cycle specified by these bits only at the first read access. The subsequent read access is executed according to the number of cycle set in the RACC. MEM_XCS[0/2/4] and MEM_XRD are asserted simultaneously.
When other values than 0 are set to these bits, specify "0" to RADC (Read Address Setup Cycle.)

0 0 cycle (initial value)
| |
15 15 cycles

Bit7-4: RADC (Read Address Setup cycle)

These bits set number of read address setup cycle which asserts MEM_XCS[0/2/4] and its address but not MEM_XRD. When 0 is selected, MEM_XRD and MEM_XCS[0/2/4] are asserted simultaneously. The specifying value should be within number of the read access setup cycle.
When NOR Flash page access mode is applied, set these bits to "0".
When RDY bit is set to "1", the value should be 3 or more (3 cycles or more.)

0 0 cycle (initial value)
| |
15 15 cycles

Bit3-0: RACC (Read Access Cycle)

These bits set number of cycle required for the read access. Although the address does not change during the cycle specified by these bits, data is fetched at the last cycle.
When RDY bit is set to "1", the value should be 3 or more (4 cycles or more.)

0 1 cycle
| |
15 16 cycles (initial value)

8.7.3. SRAM/Flash area register 0-7 (MCFAREA0-7)

Register address	BaseAddress+0x0040	MCFAREA0 (External pin: MEM_XCS[0])	
	BaseAddress+0x0044	MCFAREA1 (External pin: N/A)	(*1)
	BaseAddress+0x0048	MCFAREA2 (External pin: MEM_XCS[2])	
	BaseAddress+0x004C	MCFAREA3 (External pin: N/A)	(*1)
	BaseAddress+0x0050	MCFAREA4 (External pin: MEM_XCS[4])	
	BaseAddress+0x0054	MCFAREA5 (External pin: N/A)	(*1)
	BaseAddress+0x0058	MCFAREA6 (External pin: N/A)	(*1)
	BaseAddress+0x005C	MCFAREA7 (External pin: N/A)	(*1)

Bit No.	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field name	Reserved									MASK						
R/W	R/W0									R/W						
Initial value	X									15 (16MB width)						
Bit No.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved								ADDR							
R/W	R/W0								R/W							
Initial value	X								(in order of MEM_XCS[0/2/4]) 64,32,0							

*1: This must set not to overlap address area of CS0/2/4 and CS1/3/5/6/7 (refer 8.2 Spec limitation)

Bit31-23: Reserved

Reserved bits.

Write "0" to these bits. Their read value is undefined.

Bit22-16: MASK (Address mask)

These bits set mask value of the one set to ADDR. This external bus interface masks ADDR (masked with setting "1") and internal bus mask address according to the specified mask to compare them. When they are matched, external bus interface accesses to MEM_XCS[4/2/0] signal. [22:16] masks each address [26:20].

(Example)

ADDR = 00001000 (b)

MASK = 0000011 (b)

<When the device is selected>

Internal bus address (external interface address): AD = 0x10900000

Mask

ADDR & (!MASK) = 00001000 (b)

AD [27:20] & (!MASK) = 00001000 (b) ….. Matched, and this device is selected

<When the device is not selected>

Internal bus address (external interface address): AD = 0x10c00000

Masking

ADDR & (!MASK) = 00001000 (b)

AD [27:20] & (!MASK) = 00001100 (b) ….. Unmatched, and device is not selected

The masking selects area size; in this example, 0x10800000 - 0x10b00000 (4MB) are selected. The bit specified "1" with masking is lost during mask processing. These bits are invalid even if they are set to ADDR. When LSB in the example is 1 (ADDR = 00001001 (b)), the same address field is selected since it is invalid in masking. The correlation of the size in mask setting and address field is shown below.

0000000 (b) → 1MB
0000001 (b) → 2MB
0000011 (b) → 4MB
0000111 (b) → 8MB
0001111 (b) → 16MB
0011111 (b) → 32MB

Note:

Each address field must not overlapped.

Bit15-8: Reserved

Reserved bits.

Write "0" to these bits. Their value is undefined.

Bit7-0: ADDR (Address)

These bits specify setting address in the corresponding chip select area. These addresses (0x0200_0000 - 0x11FF_FFFF) are allocated by SRAM/Flash interface in 256MB fixed area. Define corresponding value to [27:20] part of the address.

Table 8-3 ADDR (address [27:20]) setting value and chip select area's setting address

ADDR (address[27:20])	Setting address of chip select area
0xFF	0x0FF0_0000 (*1)
0xFE	0x0FE0_0000 (*1)
~	~
0x21	0x0210_0000 (*1)
0x20	0x0200_0000 (*1)
0x1F	0x11F0_0000 (*2)
0x1E	0x11E0_0000 (*2)
~	~
0x01	0x1010_0000 (*2)
0x00	0x1000_0000 (*2)

*1: Address becomes [31:28] = 0 × 0 at ADDR (address [27:20] = 20 - FF setting.

*2: Address becomes [31:28] = 0 × 1 at ADDR (address [27:20] = 00 - 1F setting.

8.7.4. Memory controller error register (MCERR)

Register address	BaseAddress + 0x0200															
Bit No.	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field name	Reserved															
R/W	R/W0															
Initial value	X															
Bit No.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved												Reserved	SFION	Reserved	SFER
R/W	R/W0												R/W0	R/W	R	R/W0
Initial value	X												0	0	0	0

Bit31-4: Reserved

Reserved bits.

Write "0" to these bits. Their value is undefined.

Bit3: Reserved

Reserved bit.

Write "0" to this bit. Its value is undefined.

Note:

Writing "1" to this bit is prohibited.

Bit2: SFION (SRAM/Flash error interrupt: ON)

This bit validates interrupt at SRAM/Flash error.

0: OFF (initial value)

1: ON

Bit1: Reserved

Reserved bit.

Write "0" to this bit. Its value is undefined.

Bit0: SFER (SRAM/Flash error)

This bit indicates that the area without mapping is accessed. In this case, memory controller returns error to internal bus; at the same time, this bit, is set.

When the value is "1", it is cleared by writing "0" Only when "1" is set to this bit, clear operation is available.

0: No error (Initial value)

1: Error

8.8. Connection example

16 bit NOR Flash

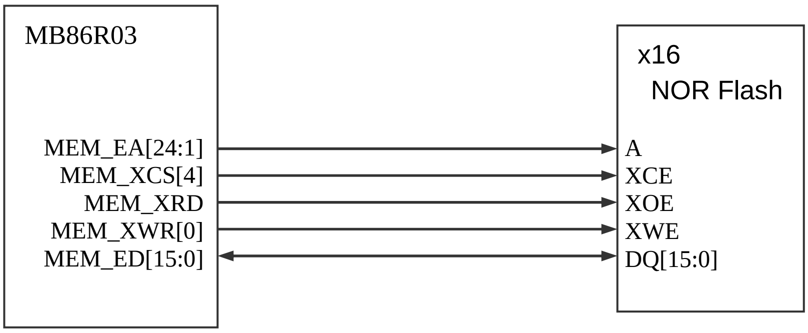


Figure 8-4 Connection example of 16 bit NOR Flash

16 bit NOR Flash + 8 bit SRAM × 2

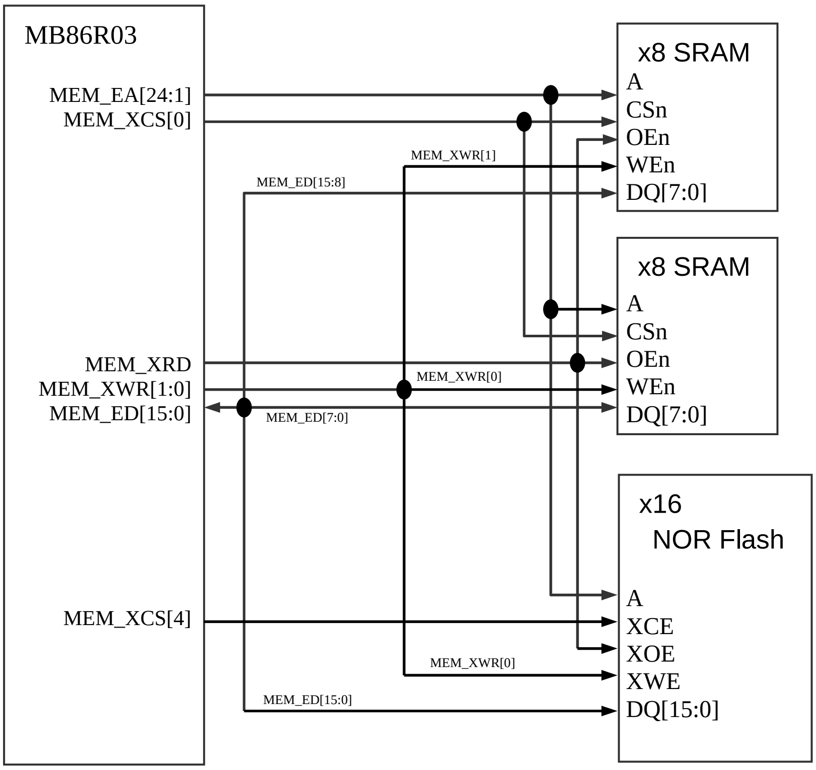


Figure 8-5 Connection example of 16 bit NOR Flash + 8 bit SRAM × 2

32 bit NOR Flash

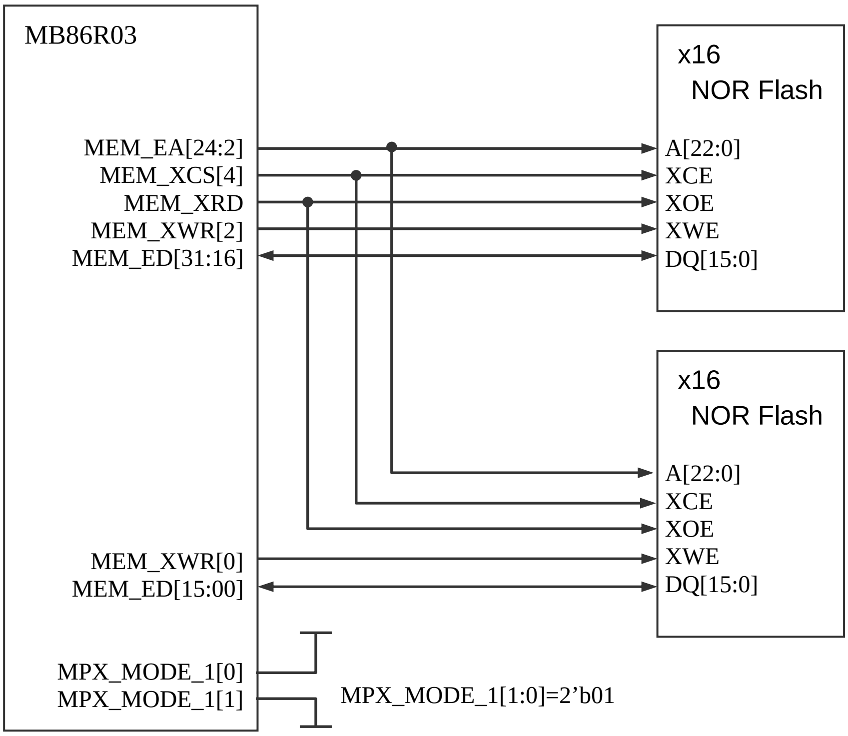


Figure 8-6 Connection example of 32 bit NOR Flash

8.9. Example of access waveform

Word read access to 16 bit width SRAM/NOR Flash

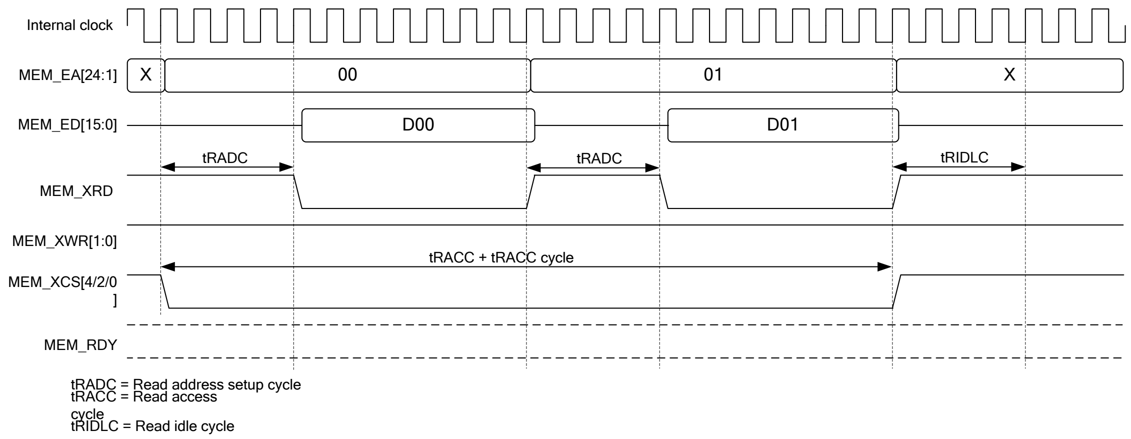


Figure 8-7 Access waveform example (word read access to 16 bit width SRAM/NOR Flash)

Word write access to 16 bit width SRAM/NOR Flash

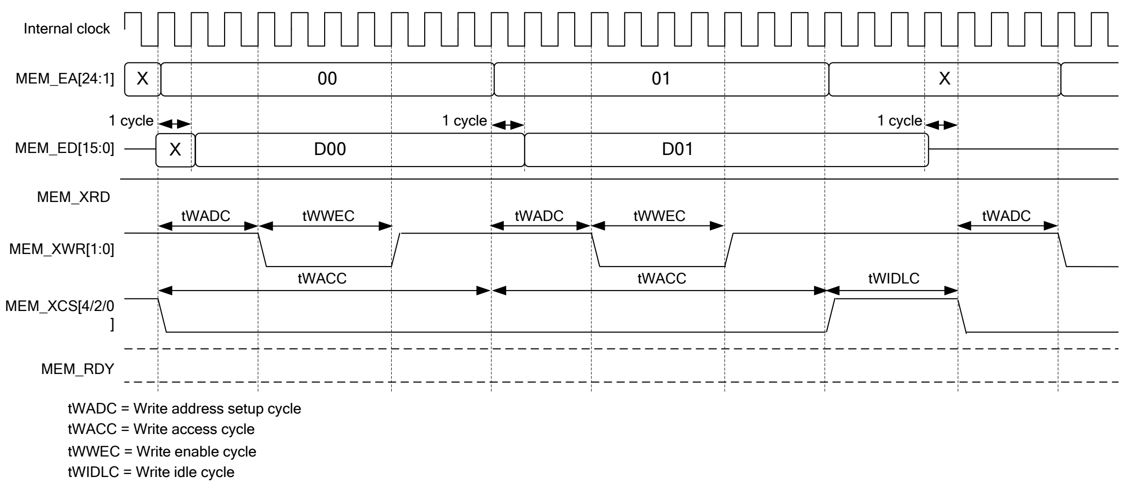


Figure 8-8 Access waveform example (word write access to 16 bit width SRAM/NOR Flash)

Read/Write to low-speed device

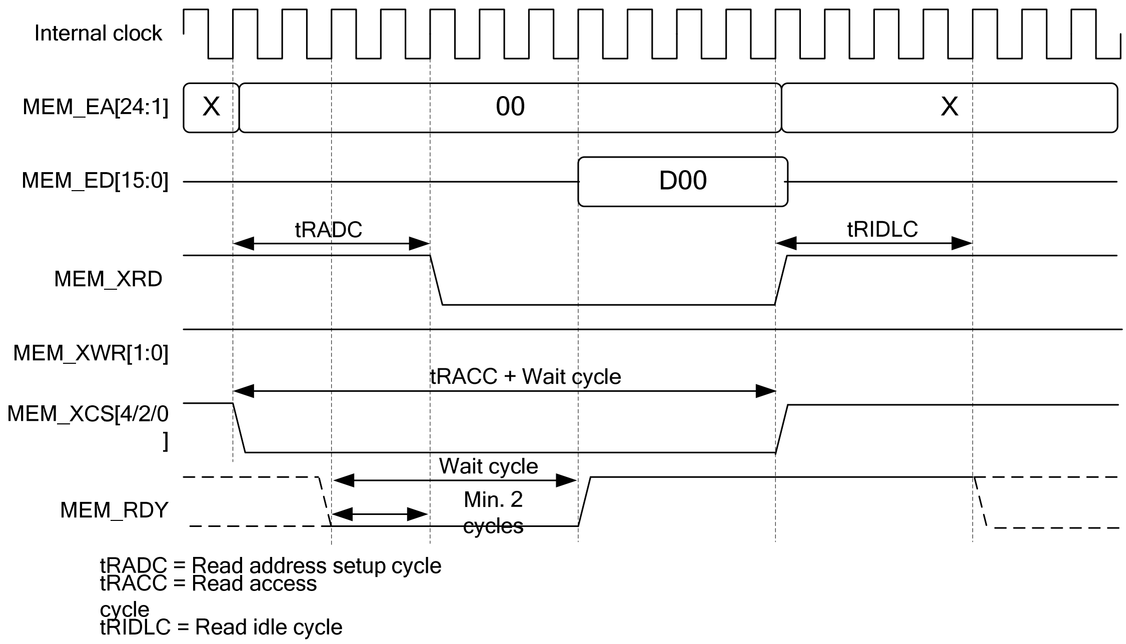


Figure 8-9 Access waveform example (half-word read access to 16 bit width low speed device)

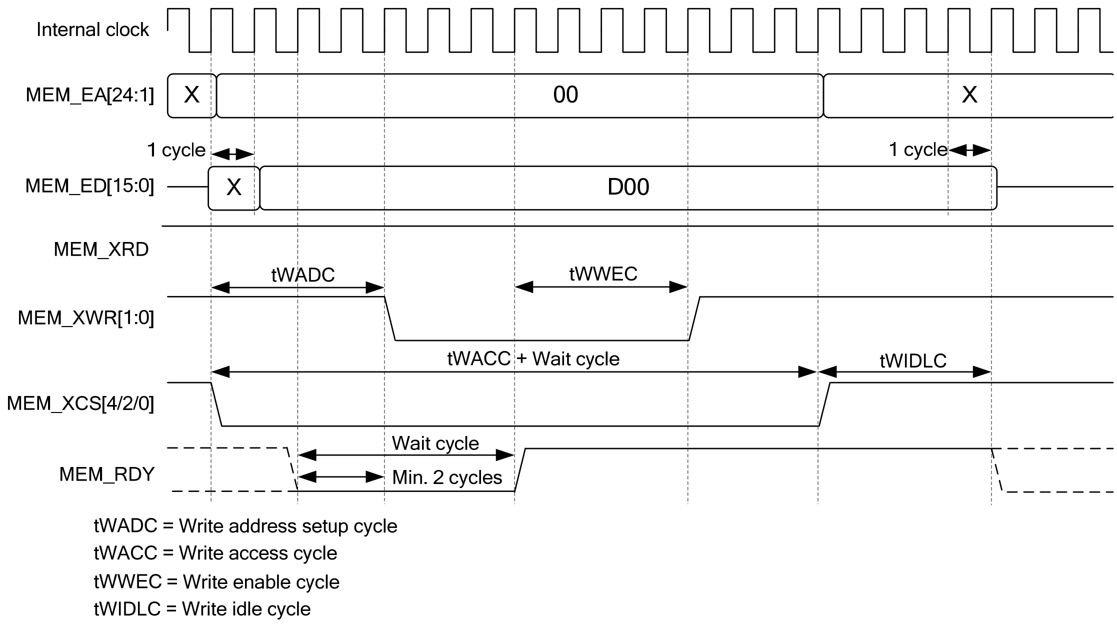


Figure 8-10 Access waveform example (half-word write access to 16 bit width low speed device)

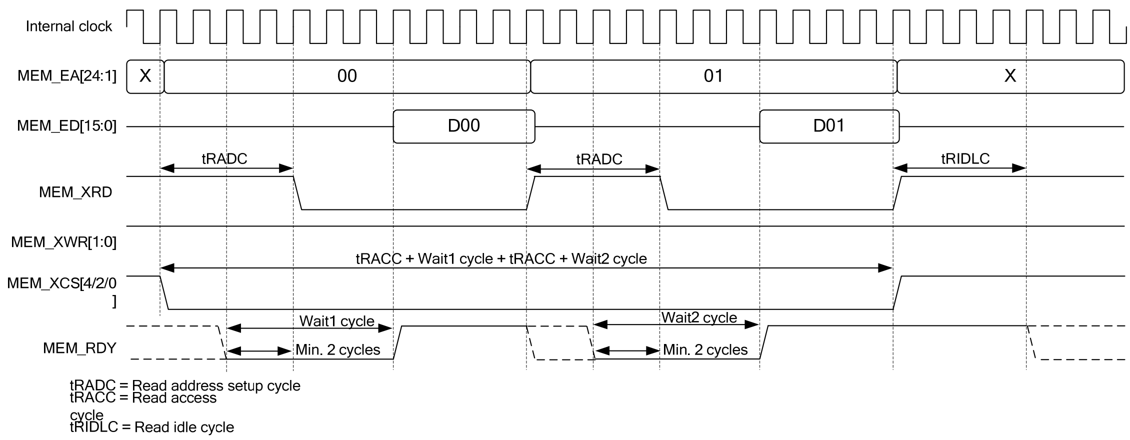


Figure 8-11 Access waveform example (word read access to 16 bit width low speed device)

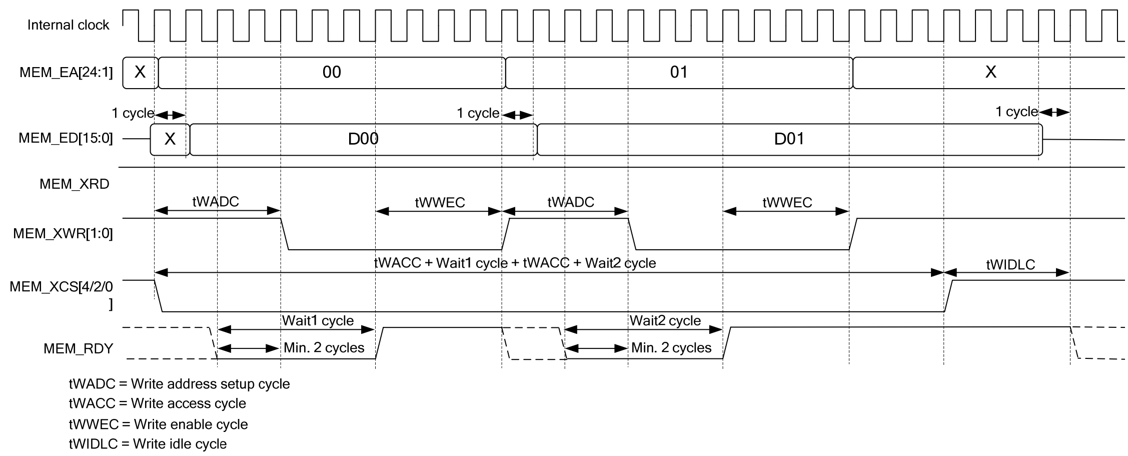


Figure 8-12 Access waveform example (word write access to 16 bit width low speed device)

Page read of 16 bit NOR Flash

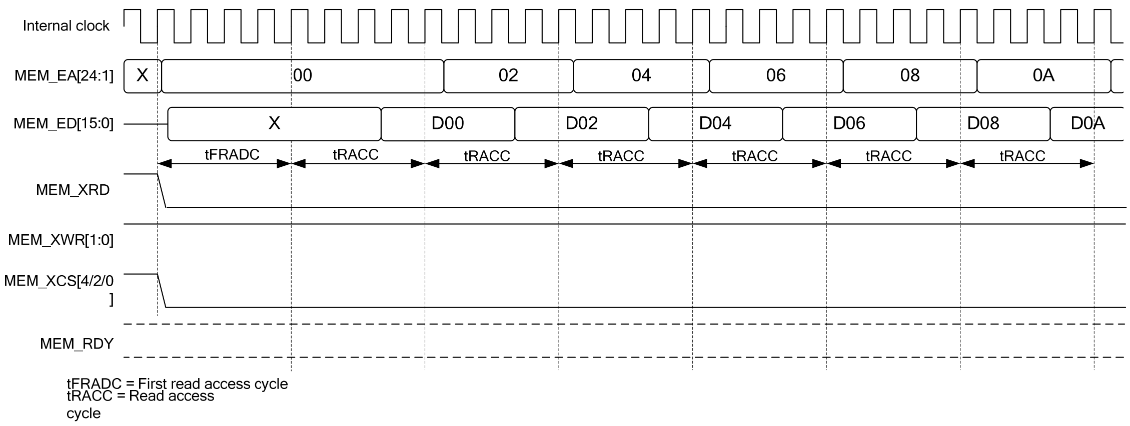


Figure 8-13 Access waveform example (16 bit NOR Flash page read)

8.10. Operation

External bus interface equips 3 chip select signals and controls SRAM and Flash.

8.10.1. External bus interface

This interface has 256MB address space that each address is able to be set arbitrarily (actual max. address size is 32MB with taking bit width of external output address into account.)
Different timing is able to be set to each chip select. NOR Flash is connectable and it accesses in normal SRAM access.
In SRAM access, MEM_XCS[4/2/0] is selected at 1 access.
When access is performed with wider bit width than the target's, it is converted to continuous access.
In continuous access, MEM_XCS[4/2/0] is fixed to L and address is changed.
For instance, the case that 32 bit read access is proceeded from internal bus to 16 bit width device, address is changed from 0 to 2, and the data is continuously fetched from MEM_ED[15:0] according to the transition timing while MEM_XCS[4/2/0] is fixed to L (refer to "8.9 Example of access waveform".)
Then the data suited to endian is returned to the internal bus.
When access is proceeded with narrower bit width than the target's (for instance, the byte access to 16 bit target), byte access is carried out with MEM_XWR[3:0] signal control during writing operation (for external bus interface, only necessary data is output.)

8.10.2. Low-speed device interface function

The external bus interface has interface function with low-speed device and MEM_RDY pin which are used by connecting RDY signal to MEM_RDY pin of this LSI. MEM_RDY pin is available only when wait state is at L and ready state is at H. RDY signal at reading should be asserted to "L" at least 2 cycles from 2 cycles before falling edge of MEM_XRD signal to actual falling edge. For the writing operation, the RDY signal should also be asserted to "L" at least 2 cycles from 2 cycles before falling edge of MEM_WXR signal to actual falling edge.
For the access exceeding external data bus width (e.g. word (32 bit) access to 16 bit device), the access is carried out "Read → Read, Write → Write" continuously until all exceeded bits are covered.
In this case, MEM_XCS[4/2/0] signal is not negated during the access regardless of setting.
When the device using negation of MEM_XCS[4/2/0] signal, the access should be done within the target width. For the device without using RDY function (e.g. SRAM memory), be sure to set "0" to RDY bit of applied chip select.
When RDY signal is H from the access start, the access is carried out in the same method as normal SRAM access.
If RDY becomes L or high pulse during access cycle, the operation is not assured.

* This function cannot be applied to the RDY/$\overline{\text{BUSY}}$ signals of the Flash memory.

8.10.3. Endian and byte lane to each access

The external bus interface corresponds to both little endian and big endian. These switches are set with external pin, BIGEND. External data bus width is set with external pin, MPX_MODE_1[1:0].
Correlation of each endian, external data bus width, and byte lane to each access is shown below.

Table 8-4 Relation of byte lane at little endian

Endian (BIGEND)	Access size	MPX_MODE_1[1:0]	Target width (WDTH)	Internal bus address	Enabled byte lane	Corresponding internal bus data	MEM_XWR [3:2]	MEM_XWR [1:0]	MEM_EA[1]
Little (=1'b0)	Word	16 bit (≠2'b01)	8bit	0	MEM_ED[7:0]	1st: H*DATA[7:0]	not active	10	0
					MEM_ED[7:0]	2nd: H*DATA[15:8]			0
					MEM_ED[7:0]	3rd: H*DATA[23:16]			1
					MEM_ED[7:0]	4th: H*DATA[31:24]			1
			16bit	0	MEM_ED[15:0]	1st: H*DATA[15:0]	not active	00	0
					MEM_ED[15:0]	2nd: H*DATA[31:16]			1
			32bit(prohibited)	-	-	-	-	-	-
		32 bit (=2'b01)	8bit	0	MEM_ED[7:0]	1st: H*DATA[7:0]	not active	10	0
					MEM_ED[7:0]	2nd: H*DATA[15:8]			0
					MEM_ED[7:0]	3rd: H*DATA[23:16]	not active		1
					MEM_ED[7:0]	4th: H*DATA[31:24]			1
			16bit	0	MEM_ED[15:0]	1st: H*DATA[15:0]	not active	00	0
					MEM_ED[15:0]	2nd: H*DATA[31:16]			1
			32bit	0	MEM_ED[31:0]	H*DATA[31:0]	00	00	0
	Half-Word	16 bit (≠2'b01)	8bit	0	MEM_ED[7:0]	1st: H*DATA[7:0]	not active	10	0
					MEM_ED[7:0]	2nd: H*DATA[15:8]			0
				2	MEM_ED[7:0]	1st: H*DATA[23:16]	not active	10	1
					MEM_ED[7:0]	2nd: H*DATA[31:24]			1
			16bit	0	MEM_ED[15:0]	H*DATA[15:0]	not active	00	0
				2	MEM_ED[15:0]	H*DATA[31:16]	not active	00	1
			32bit(prohibited)	-	-	-	-	-	-
		32 bit (=2'b01)	8bit	0	MEM_ED[7:0]	1st: H*DATA[7:0]	not active	10	0
					MEM_ED[7:0]	2nd: H*DATA[15:8]			0
				2	MEM_ED[7:0]	1st: H*DATA[23:16]	not active	10	1
					MEM_ED[7:0]	2nd: H*DATA[31:24]			1
			16bit	0	MEM_ED[15:0]	H*DATA[15:0]	not active	00	0
				2	MEM_ED[15:0]	H*DATA[31:16]	not active	00	1
			32bit	0	MEM_ED[15:0]	H*DATA[15:0]	11	00	0
				2	MEM_ED[31:16]	H*DATA[31:16]	00	11	0
	Byte	16 bit (≠2'b01)	8bit	0	MEM_ED[7:0]	H*DATA[7:0]	not active	10	0
				1	MEM_ED[7:0]	H*DATA[15:8]	not active	10	0
				2	MEM_ED[7:0]	H*DATA[23:16]	not active	10	1
				3	MEM_ED[7:0]	H*DATA[31:24]	not active	10	1
			16bit	0	MEM_ED[7:0]	H*DATA[7:0]	not active	10	0
				1	MEM_ED[15:8]	H*DATA[15:8]	not active	01	0
				2	MEM_ED[7:0]	H*DATA[23:16]	not active	10	1
				3	MEM_ED[15:8]	H*DATA[31:24]	not active	01	1
			32bit(prohibited)	-	-	-	-	-	-
		32 bit (=2'b01)	8bit	0	MEM_ED[7:0]	H*DATA[7:0]	not active	10	0
				1	MEM_ED[7:0]	H*DATA[15:8]	not active	10	0
				2	MEM_ED[7:0]	H*DATA[23:16]	not active	10	1
				3	MEM_ED[7:0]	H*DATA[31:24]	not active	10	1
			16bit	0	MEM_ED[7:0]	H*DATA[7:0]	not active	10	0
				1	MEM_ED[15:8]	H*DATA[15:8]	not active	01	0
				2	MEM_ED[7:0]	H*DATA[23:16]	not active	10	1
				3	MEM_ED[15:8]	H*DATA[31:24]	not active	01	1
			32bit	0	MEM_ED[7:0]	H*DATA[7:0]	11	10	0
				1	MEM_ED[15:8]	H*DATA[15:8]	11	01	0
				2	MEM_ED[23:16]	H*DATA[23:16]	10	11	0
				3	MEM_ED[31:24]	H*DATA[31:24]	01	11	0

H*DATA: HWDATA or HRDATA is internal signals

Table 8-5 Relation of byte lane at big endian

Endian (BIGEND)	Access size	MPX_MODE_1[1:0]	Target width (WDTH)	Internal bus address	Enabled byte lane	Corresponding internal bus data	MEM_XWR [3:2]	MEM_XWR [1:0]	MEM_EA[1]
Big (=1'b1)	Word	16 bit (≠2'b01)	8bit	0	MEM_ED[15:8]	1st: H*DATA[31:24]	not active	01	0
					MEM_ED[15:8]	2nd: H*DATA[23:16]			0
					MEM_ED[15:8]	3rd: H*DATA[15:8]		01	1
					MEM_ED[15:8]	4th: H*DATA[7:0]			1
			16bit	0	MEM_ED[15:0]	1st: H*DATA[31:16]	not active	00	0
					MEM_ED[15:0]	2nd: H*DATA[15:0]			1
			32bit(prohibited)	-	-	-	-	-	-
		32 bit (=2'b01)	8bit	0	MEM_ED[15:8]	1st: H*DATA[31:24]	not active	01	0
					MEM_ED[15:8]	2nd: H*DATA[23:16]			0
					MEM_ED[15:8]	3rd: H*DATA[15:8]		01	1
					MEM_ED[15:8]	4th: H*DATA[7:0]			1
			16bit	0	MEM_ED[15:0]	1st: H*DATA[31:16]	not active	00	0
					MEM_ED[15:0]	2nd: H*DATA[15:0]			1
			32bit	0	MEM_ED[31:0]	H*DATA[31:0]	00	00	0
	Half-Word	16 bit (≠2'b01)	8bit	0	MEM_ED[15:8]	1st: H*DATA[31:24]	not active	01	0
					MEM_ED[15:8]	2nd: H*DATA[23:16]			0
				2	MEM_ED[15:8]	1st: H*DATA[15:8]	not active	01	1
					MEM_ED[15:8]	2nd: H*DATA[7:0]			1
			16bit	0	MEM_ED[15:0]	H*DATA[31:16]	not active	00	0
				2	MEM_ED[15:0]	H*DATA[15:0]	not active	00	1
			32bit(prohibited)	-	-	-	-	-	-
		32 bit (=2'b01)	8bit	0	MEM_ED[15:8]	1st: H*DATA[31:24]	not active	01	0
					MEM_ED[15:8]	2nd: H*DATA[23:16]			0
				2	MEM_ED[15:8]	1st: H*DATA[15:8]	not active	01	1
					MEM_ED[15:8]	2nd: H*DATA[7:0]			1
			16bit	0	MEM_ED[15:0]	H*DATA[31:16]	not active	00	0
				2	MEM_ED[15:0]	H*DATA[15:0]	not active	00	1
			32bit	0	MEM_ED[31:16]	H*DATA[31:16]	00	11	0
				2	MEM_ED[15:0]	H*DATA[15:0]	11	00	0
	Byte	16 bit (≠2'b01)	8bit	0	MEM_ED[15:8]	H*DATA[31:24]	not active	01	0
				1	MEM_ED[15:8]	H*DATA[23:16]	not active	01	0
				2	MEM_ED[15:8]	H*DATA[15:8]	not active	01	1
				3	MEM_ED[15:8]	H*DATA[7:0]	not active	01	1
			16bit	0	MEM_ED[15:8]	H*DATA[31:24]	not active	01	0
				1	MEM_ED[7:0]	H*DATA[23:16]	not active	10	0
				2	MEM_ED[15:8]	H*DATA[15:8]	not active	01	1
				3	MEM_ED[7:0]	H*DATA[7:0]	not active	10	1
			32bit(prohibited)	-	-	-	-	-	-
		32 bit (=2'b01)	8bit	0	MEM_ED[15:8]	H*DATA[31:24]	not active	01	0
				1	MEM_ED[15:8]	H*DATA[23:16]	not active	01	0
				2	MEM_ED[15:8]	H*DATA[15:8]	not active	01	1
				3	MEM_ED[15:8]	H*DATA[7:0]	not active	01	1
			16bit	0	MEM_ED[15:8]	H*DATA[31:24]	not active	01	0
				1	MEM_ED[7:0]	H*DATA[23:16]	not active	10	0
				2	MEM_ED[15:8]	H*DATA[15:8]	not active	01	1
				3	MEM_ED[7:0]	H*DATA[7:0]	not active	10	1
			32bit	0	MEM_ED[31:24]	H*DATA[31:24]	01	11	0
				1	MEM_ED[23:16]	H*DATA[23:16]	10	11	0
				2	MEM_ED[15:8]	H*DATA[15:8]	11	01	0
				3	MEM_ED[7:0]	H*DATA[7:0]	11	10	0

H*DATA: HWDATA or HRDATA is internal signals

9. DDR2 controller

This chapter describes function and operation of DDR2 controller (DDR2C.)

9.1. Outline

DDR2C adopts AHB bus used in the register access as HOST IF and AXI bus used in the memory access. Memory IF supports DDR2SDRAM (DDR2-400.)

9.2. Feature

DDR2C has following features:

a. AHB IF
  - a) Register access by slave function of AHB IF
  - b) Register setting contents
    - a- Operation setting of DDR2C
    - b- Initialization sequence control (DDR IF macro setting, OCD/ODT setting on DDR2C side, SDRAM initialization command issue, and SDRAM control setting)

b. AXI IF
  - a) Storing read/write transactions to internal FIFO by slave function of AHB IF
  - b) Internal FIFO composition
    - a- Address FIFO: Depth = 8 - 28 (controllable with register setting).
    - b- Write data FIFO: Depth = 52
    - c- Read data FIFO: Depth = 62
    - d- Read control FIFO: Depth = 28

c. DRAM IF
  - a) 512M bit/256M bit DDR2SDRAM (SSTL18) × 2pcs. (recommended) or 1pc.
    (DDR2-400/533/667/800 in compliance with JESD79-2C is used as DDR2-400; in addition, SDRAM with ODT=50Ω setting is recommended.)
  - b) Switch of initialization mode and normal operation mode
  - c) SDRAM usage restriction (AL = 0, CL = 3, WL = 2, BL = 4, Bank = 4)
  - d) Automatic issuing function of refresh command
  - e) Max. 166MHz of SDRAM CLK (double edge: 333MHz)

9.3. Block diagram

Figure 9-1 shows block diagram of DDR2 controller (DDR2C.)

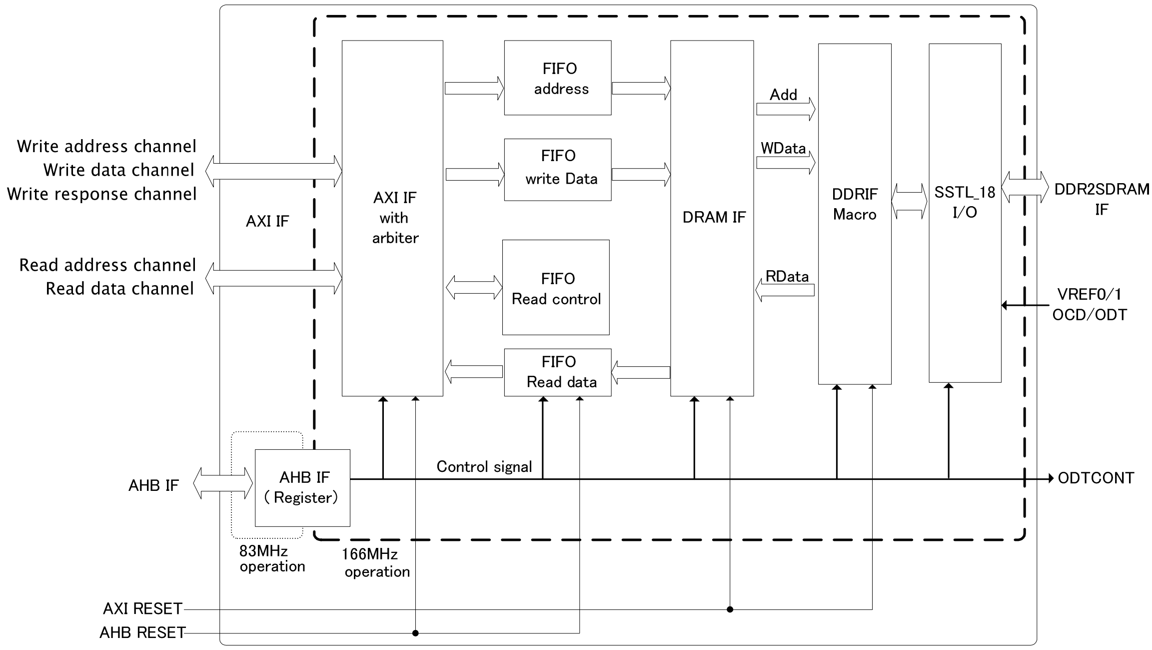


Figure 9-1 Block diagram of DDR2 controller (DDR2C)

Table 9-1 shows each function of the DDR2C block.

Table 9-1 Individual block function

Block	Function
AHB IF	• Slave function of AHB IF • Control register.
AXI IF	• Slave function of AXI IF • FIFO control function
FIFO	• Address/Write • Data/Read • Control/Read • Data storage FIFO
DRAM IF	• DDRIF macro control function • SDRAM IF control function
DDRIF macro	• Connection between DRAM IF module and IO (Read data's importing phase adjustment) • Built-in DLL
SSTL_18 I/O	• STUB series terminated logic for 1.8V single end buffer (OCD and ODT functions are embedded) • STUB series terminated logic for 1.8V differential buffer (OCD and ODT functions are embedded) • ODT auto. adjustment function

9.4. Supply clock

AHB clock is supplied to DDR2 controller. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

9.5. Register

This section describes DDR2 controller (DDR2C) register.

9.5.1. Register list

Table 9-2 shows DDR2C register list.

Table 9-2 DDR2C register list

Address		Register name	Abbreviation	Description
Base	Offset			
F300_0000$_H$	+ 00$_H$	DRAM Initialization Control Register	DRIC	Initialization control register
	+ 02$_H$	DRAM Initialization Command Register [1]	DRIC1	Initialization control command register 1
	+ 04$_H$	DRAM Initialization Command Register [2]	DRIC2	Initialization control command register 2
	+ 06$_H$	DRAM CTRL ADD Register	DRCA	Address control register
	+ 08$_H$	DRAM Control Mode Register	DRCM	Mode control register
	+ 0A$_H$	DRAM CTRL SET TIME1 Register	DRCST1	Timing setting register 1
	+ 0C$_H$	DRAM CTRL SET TIME2 Register	DRCST2	Timing setting register 2
	+ 0E$_H$	DRAM CTRL REFRESH Register	DRCR	Refresh control register
	+ 10$_H$ - + 1F$_H$	(Reserved)	-	Access prohibited
	+ 20$_H$	DRAM CTRL FIFO Register	DRCF	FIFO control register
	+ 22$_H$ - + 2F$_H$	(Reserved)	-	Access prohibited
	+ 30$_H$	AXI Setting	DRASR	AXI operation setting register
	+ 32$_H$ - + 4F$_H$	(Reserved)	-	Access prohibited
	+ 50$_H$	DRAM IF MACRO SETTING DLL Register	DRIMSD	DDRIFmacro setting register
	+ 52$_H$ - + 5F$_H$	(Reserved)	-	Access prohibited
	+ 60$_H$	DRAM ODT SETTING Register	DROS	ODT setting register
	+ 62$_H$ - + 63$_H$	(Reserved)	-	Access prohibited
	+ 64$_H$	IO buffer setting ODT1	DRIBSODT1	IO ODT1 setting register
	+ 66$_H$	IO buffer setting OCD	DRIBSOCD	IO OCD setting register
	+ 68$_H$	IO buffer setting OCD2	DRIBSOCD2	IO OCD2 setting register
	+ 6A$_H$ - + 6F$_H$	(Reserved)	-	Access prohibited
	+ 70$_H$	ODT Auto Bias Adjust	DROABA	ODT bias self adjustment register
	+ 72$_H$ - + 83$_H$	(Reserved)	-	Access prohibited
	+ 84$_H$	ODT Bias Select Register	DROBS	ODT bias selection register
	+ 86$_H$ - + 8F$_H$	(Reserved)	-	Access prohibited
	+ 90$_H$	IO Monitor Register1	DRIMR1	IO monitor register 1
	+ 92$_H$	IO Monitor Register2	DRIMR2	IO monitor register 2
	+ 94$_H$	IO Monitor Register3	DRIMR3	IO monitor register 3
	+ 96$_H$	IO Monitor Register4	DRIMR4	IO monitor register 4
	+ 98$_H$	OCD Impedance Setting Register1	DROISR1	OCD impedance setting register 1
	+ 9A$_H$	OCD Impedance Setting Register2	DROISR2	OCD impedance setting register 2

Description format of register

Following format is used for description of register's each bit in "9.5.2 DRAM initialization control register (DRIC)" to "9.5.24 OCD impedance setting register2 (DROISR2)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

9.5.2. DRAM initialization control register (DRIC)

DRIC register is used to initialize DRAM; in addition, it controls initialization mode setting, issue of initialization command, and others.

Address	F300_0000$_H$ + 00$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRINI	CKEN	-	-	-	-	-	-	-	-	-	-	REFBSY	DDRBSY	CMDRDY	DRCMD
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	R	R	W
Initial value	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field No.	Bit field Name	Description
15	DRINI	This sets DRAM initialization operation mode. 0: Normal operation 1: Initialization mode (initial value) When initialization is completed, this bit becomes 0. Only when DRINI bit is 1, CKEN and DRCMD bits of this register, and the DRAM initialization command register [1]/[2] become valid. When this bit is 0, these registers and bits are don’t care. Note: • Data access and auto. refresh to DRAM are not performed in the initialization operation mode. • Only when there is no access request to DDR, DRINI bit can be changed to 0 → 1. The access request to DDR is able to be judged by DDRBSY (bit 2.) • When DRINI bit is "1", do not access to data from AXI. When data access is requested in the state of DRINI = 1, DDR2 controller may keep occupying the AXI bus. Moreover, the data requested from AXI may be destroyed.
14	CKEN	This is CKE control signal to DDR. Normal operation (DRINI = 0): CKE output always becomes "1" Initialization mode (DRINI = 1): CKE output becomes "1"
13-4	(Reserved)	Reserved bits. Write access is ignored.
3	REFBSY	This bit indicates refresh cycle to DDR. 0: It is not refresh cycle 1: It is refresh cycle
2	DDRBSY	This bit indicates status that data access is requested to DDR. 0: Neither command request to DDR nor access to DDR occurs 1: Command request to DDR or access operation to DDR occurs (busy)
1	CMDRDY	This bit indicates DRAM command is ready. It also shows whether "1" is able to be written to DRCMD bit (writing command bit to DRAM.) 0: 1 cannot be written to DRCMD (bit 0) 1: 1 can be written to DRCMD This bit indicates valid value for only at DRINI = 1. CMDRDY bit becomes "1" in the following cases: • Between writing "1" to DRCMD (bit 0) to completion of the command. • Accessing to DRAM is not completed when DRINI bit is changed to 0 → 1 without reset.

Bit field		Description
No.	Name	
0	DRCMD	This is writing command bit to DRAM. Writing "1" to this bit outputs setting condition of DRAM initialization command register [1]/[2] to DRAM during 1ck period of time. Note: • When DRCMD bit does not issue command in the initialization mode, the state becomes NOP or DSEL to DRAM. • Only when CMDBSY (bit 1) is "0", "1" is able to be written to this bit.

9.5.3. DRAM initialization command register [1] (DRIC1)

This register sets each control signal value of DRAM at the initialization operation.
When "1" is written to DRCMD in the initialization mode (DRINI = 1), the signal corresponding to DRAM bus is driven by this setting value.

Address	F300_0000$_H$ + 02$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	#CS	#RAS	#CAS	#WE	BA2	BA1	BA0
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
Initial value	X	X	X	X	X	X	X	X	X	1	1	1	1	1	1	1

9.5.4. DRAM initialization command register [2] (DRIC2)

This register sets DRAM address signal value at the initialization operation.
When "1" is written to DRCMD in the initialization mode (DRINI = 1), the signal corresponding to DRAM bus is driven by this setting value.

Address	F300_0000$_H$ + 04$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRAM initialization method

All DRAM is initialized by CPU.
DDR2 controller is structured that each signal conductor necessary for the DRAM setting can be driven by the register value in the initialization mode. Set certain value to this register beforehand and "1" to command bit (DRCMD) to execute the setting command to DRAM.

To issue "Precharge all (PALL)" command to DRAM

1) Set "Bit[5:0] = 001000(b)" to the DRAM initialization command register [1].
2) Set "Bit[13:0] = 00010000000000(b)" to the DRAM initialization command register [2].
   (Setting order of these 2 registers is not specified.)
3) Write "1" to bit 0 of the DRAM initialization control register.

The value set at 1) and 2) is output to DRAM for 1ck period of time, and this becomes command to DRAM.

- Command to DRAM without command execution in the initialization mode is NOP or DSEL
- For each control method of DRAM command and initialization, refer applied DRAM data sheet.

9.5.5. DRAM CTRL ADD register (DRCA)

This register sets items such as capacity of DRAM to be connected.
06_H-$0C_H$ register settings related to DDR2 controller's DRAM operation should be fixed before completing DRAM initialization.

Address	$F300\_0000_H + 06_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TYPE		Bus16	-	-	-	BankRange		RowRange				ColRange			
R/W	R/W		R/W	R/W	R/W	R/W	R/W		R/W				R/W			
Initial value	1	1	0	X	X	X	0	1	0	0	1	0	0	0	1	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>15-14</td><td>TYPE</td><td>Operation mode of DRAM control core is set.<table><tr><td>11</td><td>DRAM control core operates in the DDR2SDRAM mode</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
<tr><td>13</td><td>Bus16</td><td>This specifies bus width of DRAM connected to external part.<table><tr><td>0</td><td>32 bit</td></tr><tr><td>1</td><td>16 bit</td></tr></table>Remark:
• Use DQ[15:0], DQS0/1, and DM0/1
• See the pin specifications for process of unused DQ[31:16], DQS2/3, and DM2/3</td></tr>
<tr><td>12-10</td><td>(Reserved)</td><td>Reserved bits.
Write access is ignored.</td></tr>
<tr><td>9-8</td><td>BankRange</td><td>Bank address is set.
Since only 4 banks are applied, these bits are ready only and fixed to 01(b.)</td></tr>
<tr><td>7-4</td><td>RowRange</td><td>Row address range is set.<table><tr><td>0001</td><td>4096 (12 bit)</td></tr><tr><td>0010</td><td>8192 (13 bit)</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
<tr><td>3-0</td><td>ColRange</td><td>Col address range is set.<table><tr><td>0001</td><td>256 (8 bit)</td></tr><tr><td>0010</td><td>512 (9 bit)</td></tr><tr><td>0100</td><td>1024 (10 bit)</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
</table>

9.5.6. DRAM control mode register (DRCM)

This register sets operation mode of DRAM, and the same setting as DRAM should be set. The operation mode is unable to be changed due to DDRIF macro and other restrictions.

Address	F300_0000$_H$ + 08$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	BT	-	AL			-	CL			-	BL		
R/W	R/W	R/W	R/W	R	R/W	R			R/W	R/W			R/W	R/W		
Initial value	X	X	X	0	X	0	0	0	X	0	1	1	X	0	1	0

Bit field No.	Bit field Name	Description
15-13	(Reserved)	Reserved bits. Write access is ignored.
12	BT	Only sequential is applied in the burst type setting. Setting to DRAM should also be "sequential". 0: Sequential (initial value) 1: Reserved (setting prohibited)
11	(Reserved)	Reserved bit. Write access is ignored.
10-8	AL	Additive latency is set. This module operates with AL = 0, and it should also be set to DRAM.
7	(Reserved)	Reserved bit. Write access is ignored.
6-4	CL	CAS latency is specified. 011: CL = 3 (fixed) Others: Reserved (setting prohibited) DRAM setting should also have the same as this register’s.
3	(Reserved)	Reserved bit. Write access is ignored.
2-0	BL	Burst length is specified. 010: BL = 4 (fixed) Others: Reserved (setting prohibited) DRAM setting should also have the same as this register’s.

Note:

- The DRCM register is unable to be used for DRAM initialization.
- Set operation mode of DRAM control core at normal operation to this register. When DRINI bit (bit 15) of the DRAM initialization control register becomes "0" (normal operation mode), DRAM control core operates according to the DRCM register setting. Be sure to complete the setting before "0" is set to the DRINI bit.

9.5.7. DRAM CTRL SET TIME1 Register (DRCST1)

This register sets access timing to DRAM. It should be set with correlation of internal clock frequency and DRAM spec to be used.

Address	F300_0000$_H$ + 0A$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	TRCD			-	TRAS			-	TRP			TRC			
R/W	R/W	R/W			R/W	R/W			R/W	R/W			R/W			
Initial value	X	1	1	1	X	1	1	1	X	1	1	1	1	1	1	1

Bit field No.	Bit field Name	Description
15	(Reserved)	Reserved bit. Write access is ignored.
14-12	TRCD	RAS to CAS delay time (rRCD : Active to read or write command delay) Bit[14:12] / Delay time (number of clock) / 000 / - / Reserved (Setting prohibited) 001 / - / 010 / 2 / 011 / 3 / 100 / 4 / 101 / 5 / 110 / 6 / 111 / 7 / (Initial value)
11	(Reserved)	Reserved bit. Write access is ignored.
10-8	TRAS	RAS active time (rRAS : Active to precharge command) Bit[10:8] / Delay time (number of clock) / 000 / - / Reserved (Setting prohibited) 001 / 5 / 010 / 6 / 011 / 7 / 100 / 8 / 101 / 9 / 110 / 10 / 111 / 11 / (Initial value)
7	(Reserved)	Reserved bit. Write access is ignored.
6-4	TRP	Precharge time (tRP : Precharge period) Bit[6:4] / Delay time (number of clock) / 000 / - / Reserved (Setting prohibited) 001 / 3 / 010 / 4 / 011 / 5 / 100 / 6 / 101 / 7 / 110 / 8 / 111 / 9 / (Initial value)

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>3-0</td><td>TRC</td><td>RAS cycle time (tRC : Active to active/Auto. refresh command time)

<table>
<tr><td>Bit[3:0]</td><td>Delay time (number of clock)</td><td></td></tr>
<tr><td>0000</td><td>-</td><td rowspan="6">Reserved
(Setting prohibited)</td></tr>
<tr><td>0001</td><td>-</td></tr>
<tr><td>0010</td><td>-</td></tr>
<tr><td>0011</td><td>-</td></tr>
<tr><td>0100</td><td>-</td></tr>
<tr><td>0101</td><td>-</td></tr>
<tr><td>0110</td><td>8</td><td></td></tr>
<tr><td>0111</td><td>9</td><td></td></tr>
<tr><td>1000</td><td>10</td><td></td></tr>
<tr><td>1001</td><td>11</td><td></td></tr>
<tr><td>1010</td><td>12</td><td></td></tr>
<tr><td>1011</td><td>13</td><td></td></tr>
<tr><td>1100</td><td>14</td><td></td></tr>
<tr><td>1101</td><td>15</td><td></td></tr>
<tr><td>1110</td><td>16</td><td></td></tr>
<tr><td>1111</td><td>17</td><td>(Initial value)</td></tr>
</table>

For ACT command interval, larger value of either rRC and rRAS+rRP+tWR is used.</td></tr>
</table>

9.5.8. DRAM CTRL SET TIME2 register (DRCST2)

This register sets access timing to DRAM. It should be set by the correlation between DRAM spec and inner clock frequency.

Address	F300_0000$_H$ + 0C$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-				TRFC				-	-	TRRD		-	TWR		
R/W	R/W				R/W				R/W	R/W	R/W		R/W	R/W		
Initial value	X	1	1	0	1	0	1	1	X	X	1	1	X	1	0	1

Bit field No.	Bit field Name	Description
15-12	(Reserved)	Reserved bits. Write access is ignored.
11-8	TRFC	Auto. refresh command period (tRFC : Auto. refresh to active/Auto. refresh command time)
7-6	(Reserved)	Reserved bits. Write access is ignored.
5-4	TRRD	RAS to RAS bank active delay time (tRRD : Active bank A to active bank B command period) Active command interval for when continuously activating RAS in different bank is set in cycle.
3	(Reserved)	Reserved bit. Write access is ignored.

TRFC:

Bit[11:8]	Cycle time (number of clock)	
0000	4	
0001	5	
0010	6	
0011	7	
0100	8	
0101	9	
0110	10	
0111	11	
1000	12	
1001	13	
1010	14	
1011	15	(Initial value)
1100	16	
1101	17	
1110	18	
1111	19	

TRRD:

Bit[5:4]	Cycle time (number of clock)	
11	3	(Initial value)
Others	-	Reserved (setting prohibited)

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>2-0</td><td>TWR</td><td>Write recovery time (tWR : Write recovery time)
Write recovery time of DRAM is set in cycle.

<table>
<tr><td>Bit[2:0]</td><td>Cycle time (number of clock)</td><td></td></tr>
<tr><td>000</td><td>-</td><td>Reserved (setting prohibited)</td></tr>
<tr><td>001</td><td>2</td><td></td></tr>
<tr><td>010</td><td>3</td><td></td></tr>
<tr><td>011</td><td>4</td><td></td></tr>
<tr><td>100</td><td>5</td><td></td></tr>
<tr><td>101</td><td>6</td><td>(Initial value)</td></tr>
<tr><td>110</td><td>-</td><td rowspan="2">Reserved (setting prohibited)</td></tr>
<tr><td>111</td><td>-</td></tr>
</table>
</td></tr>
</table>

9.5.9. DRAM CTRL REFRESH register (DRCR)

This register sets auto. refresh occurrence interval to DRAM. After changing this register value, refresh occurs irregularly.

Address	F300_0000$_H$ + 0E$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	CNTLD	REF_CNT							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W							
Initial value	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
15-9	(Reserved)	Reserved bits. Write access is ignored.
8	CNTLD	Counter load. REF_CNT value is forcibly loaded into internal counter. When this bit is set to 0 → 1, REF_CNT value of bit[7:0] is forcibly loaded into internal refresh counter. This is used when setting value needs to be applied, such as after REF_CNT value change. This bit does not need to be rewritten to 0 immediately after loaded because it is performed after detecting the bit change. However, this bit keeps the writing value. If bit value is not 0 at executing load operation, "1" should be written after writing "0". Although CNTLD is not used after REF_CNT change, it operates with the changed REF_CNT by having the period before setting REF_CNT.
7-0	REF_CNT	Refresh count. Auto. refresh request occurrence is set in 16 cycle. 00$_H$: Refresh request is continuously issued. Priority of refresh is higher than the read/write. Although access request to DRAM occurs, only refresh occurs with this setting. 01$_H$ - FF$_H$: Refresh request occurs in REF_CNT × 16 clock interval. If DRAM data is accessed at refresh request, refresh does not start until the access is completed.

9.5.10. DRAM CTRL FIFO register (DRCF)

This is DDR2C's internal FIFO control related register.

Address	F300_0000$_H$ + 20$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	*1	-	-	-	-	-	-	-	-	-	-	FIFO_CNT				
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				
Initial value	0	X	X	X	X	X	X	X	X	X	X	1	0	1	1	0

*1: FIFO_ARB

Bit field No.	Bit field Name	Description
15	FIFO_ARB	Capture bandwidth is improved. 0: Default 1: Capture bandwidth is improved.
14-5	(Reserved)	Reserved bits. Write access is ignored.
4-0	FIFO_CNT	FIFO FULL count. This is number of stage setting of address FIFO (FULL condition.) When picture flickers due to AXI access latency at using display and capture, it is recovered by reducing number of FIFO stage and decreasing AXI bus latency.

Bit[4:0]	Address FIFO number of stage	
00$_H$ - 01$_H$	-	Reserved (setting prohibited)
02$_H$	8	
03$_H$	9	
04$_H$	10	
05$_H$	11	
06$_H$	12	
07$_H$	13	
08$_H$	14	
09$_H$	15	
0A$_H$	16	
0B$_H$	17	
0C$_H$	18	
0D$_H$	19	
0E$_H$	20	
0F$_H$	21	
10$_H$	22	
11$_H$	23	
12$_H$	24	
13$_H$	25	
14$_H$	26	
15$_H$	27	
16$_H$	28	(Initial value)
17$_H$ - 1F$_H$	-	Reserved (setting prohibited)

9.5.11. AXI setting register (DRASR)

This register sets AXI interface operation.

Address	F300_0000$_H$ + 30$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CACHE
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field		Description
No.	Name	
15-1	(Reserved)	Reserved bits. Write access is ignored.
0	CACHE	CACHE On/Off of cash operation at reading are performed. 0: Cache off (initial value) 1: Cache on When single reading continuously occurs in a single access (16 byte) to DRAM, reading operation from AXI is enabled by the cached data in AXI module instead of accessing to DRAM. However cache is cleared in the following conditions. • Burst reading access occurs to AXI bus in DDR2C • Write access occurs to AXI bus in DR2C

9.5.12. DRAM IF MACRO SETTING DLL register (DRIMSD)

This register is for DDR2-SDRAM interface macro setting which drives macro pin corresponding to each bit by the setting value. This is also for DLL timing setting.

Address	F300_0000H + 50H															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	ISFT_3[2:0]			-	ISFT_2[2:0]			-	ISFT_1[2:0]			-	ISFT_0[2:0]		
R/W	R/W	R/W			R/W	R/W			R/W	R/W			R/W	R/W		
Initial value	X	1	1	0	X	1	1	0	X	1	1	0	X	1	1	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>15</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored.</td></tr>
<tr><td>14-12</td><td>ISFT_3[2:0]</td><td>Value of ISFT_3[2:0]<table><tr><td>110</td><td>(Initial value)</td></tr><tr><td>101</td><td>Normal operation setting value (set to 101 at DRAM initialization)</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
<tr><td>11</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored.</td></tr>
<tr><td>10-8</td><td>ISFT_2[2:0]</td><td>Value of ISFT_2[2:0]<table><tr><td>110</td><td>(Initial value)</td></tr><tr><td>101</td><td>Normal operation setting value (set to 101 at DRAM initialization)</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
<tr><td>7</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored.</td></tr>
<tr><td>6-4</td><td>ISFT_1[2:0]</td><td>Value of ISFT_1[2:0]<table><tr><td>110</td><td>(Initial value)</td></tr><tr><td>101</td><td>Normal operation setting value (set to 101 at DRAM initialization)</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
<tr><td>3</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored.</td></tr>
<tr><td>2-0</td><td>ISFT_0[2:0]</td><td>Value of ISFT_0[2:0]<table><tr><td>110</td><td>(Initial value)</td></tr><tr><td>101</td><td>Normal operation setting value (set to 101 at DRAM initialization)</td></tr><tr><td>Others</td><td>Reserved (setting prohibited)</td></tr></table></td></tr>
</table>

9.5.13. DRAM ODT SETTING register (DROS)

This register sets ODT control signal to DDR2 memory connected to external part.

Address	$F300\_0000_H + 60_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ODT0
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field		Description
No.	Name	
15-1	(Reserved)	Reserved bits. Write access is ignored.
0	ODT0	This is the value of external output pin, ODTCONT. Initial value is 0.

9.5.14. IO buffer setting ODT1 (DRIBSODT1)

ODT related setting of IO buffer is set.

Address	F300_0000$_H$ + 64$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	ZSELN	ODTONN	ZSELP	ODTONP	ZSEL	ODTON
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
Initial value	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
15-6	(Reserved)	Reserved bits. Write access is ignored.
5	ZSELN	This becomes ZSELN value of IO buffer, and this is ODT resistance setting for DQSN. 0: 150Ω or 100Ω (initial value) 1: 75Ω or 50Ω
4	ODTONN	This is ODT setting for DQS's IO, and controls ODTONN of the IO buffer. Initial value is 0. 0: IO buffer's ODTON is always "0" 1: This should be set to use ODT of IO buffer ODTON is set to off in the following case: • To adjust OCD
3	ZSELP	This becomes ZSELP value of the IO buffer, and it is ODT resistance setting of DQSP's IO. 0: 150Ω or 100Ω (initial value) 1: 75Ω or 50Ω
2	ODTONP	This is ODT setting of DQS's IO, and controls ODTONP of the IO buffer. Initial value is 0. 0: IO buffer's ODTON is always "0" 1: This should be set to use ODT of IO buffer ODTON is set to off in the following case: • To adjust OCD
1	ZSEL	This is ZSEL value of the IO buffer that is ODT resistance of IO for DQ and DM. 0: 150Ω or 100Ω (initial value) 1: 75Ω or 50Ω
0	ODTON	This is ODT setting of IO for DQ and DM, and it controls ODTON of IO buffer. Initial value is 0. 0: IO buffer's ODTON is always "0" 1: This should be set to use ODT of IO buffer ODTON is set to off in the following case: • To adjust OCD

9.5.15. IO buffer setting OCD (DRIBSOCD)

Each setting used at impedance adjustment of IO buffer is proceeded.

Address	$F300\_0000_H + 66_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	AFORCE	ADRV	OCDPOL	DIMMCAL	OCDCNT
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0

Bit field		Description
No.	Name	
15-5	(Reserved)	Reserved bits. Write access is ignored.
4	AFORCE	This is control bit to switch IO driver's A input, and "1" is set at impedance adjustment. Initial value is 0. When this bit is "1", ADRV bit value of bit 3 is added to driver input A of IO buffer. Be sure to set "0" at the normal operation.
3	ADRV	This bit combines with AFORCE of bit 4 to use. When AFORCE is "1", this bit value becomes IO driver's A input. When AFORCE is 0, it is don't care.
2	OCDPOL	This becomes OCDPOL value of IO buffer. Initial value is 0.
1	DIMMCAL	This becomes DIMMCAL value of IO buffer. Initial value is 0.
0	OCDCNT	This becomes OCDCNT value of IO buffer. Initial value is 0.

9.5.16. IO buffer setting OCD2 (DRIBSOCD2)

Each setting used at IO buffer's impedance adjustment is proceeded.

Address	$F300\_0000_H + 68_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	-	-	SUSPD	SUSPR	SSEL
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0

Bit field		Description
No.	Name	
15-3	(Reserved)	Reserved bits. Write access is ignored.
2	SUSPD	SUSPD setting of IO buffer. When SSEL = 1, this bit value is supplied to SUSPD of each IO buffer.
1	SUSPR	SUSPR setting of IO buffer. When SSEL = 1, this bit value is supplied to SUSPR of each IO buffer.
0	SSEL	This is selection bit whether to use value of bit1 or bit2 for SUSPR/SUSPD or to control at the internal logic 0: Setting at normal operation DDRIF controls SUSPR/SUSPD 1: Setting value of bit1 and bit2 is used to SUSPR/SUSPD

9.5.17. ODT auto bias adjust register (DROABA)

This register sets auto. adjustment related items of ODT bias.

Address	$F300\_0000_H + 70_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	OCO MPNPOL	OCO MPPPOL	-	-	-	IAVSET		ODTBIAS	
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
Initial value	X	X	X	X	X	X	X	1	0	X	X	X	0	0	0	0

Bit field		Description
No.	Name	
15-9	(Reserved)	Reserved bits. Write access is ignored.
8	OCOMPNPOL	This sets to detect either 0 → 1 or 1 → 0 of OCOCMPN value as valid at bias adjustment operation. 0: 0 → 1 is valid 1: 1 → 0 is valid (initial value)
7	OCOMPPPOL	This sets to detect either 0 → 1 or 1 → 0 of OCOCMPP value as valid at bias adjustment operation. 0: 0 → 1 is valid (initial value) 1: 1 → 0 is valid
6-4	(Reserved)	Reserved bits. Write access is ignored.
3-2	IAVSET	Average number of times of bias adjustment is specified. Adjustment is performed for predetermined number of times to output the average value to ODT of the I/O cell. 00: 32 times (initial value) 01: 64 times 10: 128 times 11: 256 times
1-0	ODTBIAS	Operation of bias auto. adjustment circuit is set. 00: Auto. adjustment circuit of the bias is reset (initial value) 01: Reserved (setting prohibited) 10: Reserved (setting prohibited) 11: Auto. adjustment circuit of the bias is performed

Remark: Each setting of bit2 - 8 should be set after setting ODTBIAS of bit 1 - 0 to "00" and stopping auto. adjustment operation.

9.5.18. ODT bias select register (DROBS)

This register sets ODT.

Address	F300_0000$_H$ + 84$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	AUTO
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0

Bit field		Description
No.	Name	
15-1	(Reserved)	Reserved bits. Write access is ignored.
0	AUTO	This sets whether to use ODT auto. setting value mode. When it is set, the average value calculated with auto. adjustment of the bias is used to ODT value of the I/O cell. 0: The ODT auto. setting value mode is not used 1: The ODT auto. setting value mode is used

9.5.19. IO monitor register 1 (DRIMR1)

This is input level monitor of IO buffer which is used for impedance adjustment of OCD.

Address	$F300\_0000_H + 90_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQX[15:0]															
R/W	R															
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		Description
No.	Name	
15-0	DQX[15:0]	X value of DQ[15:0] can be read.

When input value of IO is read, IO driver should be in the OCD adjustment mode.
The following settings are required:

- Bit 0 of IO buffer setting OCD2 register (68_H) is set to "1".
- Bit 1 of IO buffer setting OCD2 register (68_H) is set to "0".

Remark:
Monitor value is valid only at OCD adjustment.

9.5.20. IO monitor register 2 (DRIMR2)

This is input level monitor of IO buffer which is used for impedance adjustment of OCD.

Address	$F300\_0000_H + 92_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQX[31:16]															
R/W	R															
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		Description
No.	Name	
15-0	DQX[31:16]	X value of DQ[31:16] can be read.

When input value of IO is read, IO driver should be in the OCD adjustment mode.
The following settings are required:

- Bit 0 of IO buffer setting OCD2 register (68_H) is set to "1".
- Bit 1 of IO buffer setting OCD2 register (68_H) is set to "0".

Remark:
Monitor value is valid only at OCD adjustment.

9.5.21. IO monitor register 3 (DRIMR3)

This is input level monitor of IO buffer which is used for impedance adjustment of OCD.

Address	$F300\_0000_H + 94_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	-	DQSX[3:0]			
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	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		Description
No.	Name	
15-4	(Reserved)	Reserved bits. Write access is ignored.
3-0	DQSX[3:0]	X value of DQS[3:0] can be read.

When input value of IO is read, IO driver should be in the OCD adjustment mode.
The following settings are required:

- Bit 0 of IO buffer setting OCD2 register (68_H) is set to "1".
- Bit 1 of IO buffer setting OCD2 register (68_H) is set to "0".

Remark:
Monitor value is valid only at OCD adjustment.

9.5.22. IO monitor register 4 (DRIMR4)

This is input level monitor of IO buffer which is used for impedance adjustment of OCD.

Address	$F300\_0000_H + 96_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	-	-	-	-	-	DMX[3:0]			
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	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		Description
No.	Name	
15-4	(Reserved)	Reserved bits. Write access is ignored.
3-0	DMX[3:0]	X value of DM[3:0] can be read.

When input value of IO is read, IO driver should be in the OCD adjustment mode.
The following settings are required:

- Bit 0 of IO buffer setting OCD2 register (68h) is set to "1".
- Bit 1 of IO buffer setting OCD2 register (68h) is set to "0".

Remark:
Monitor value is valid only at OCD adjustment.

9.5.23. OCD impedance setting Rrgister1 (DROISR1)

This register sets impedance adjustment value.

Address	F300_0000$_H$ + 98$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRVN2				DRVP2				DRVN1				DRVP1			
R/W	R/W				R/W				R/W				R/W			
Initial value	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1

Bit field		Description
No.	Name	
15-12	DRVN2	This register sets DRVN value of DQ[15:8], DQS1, and DM1
11-8	DRVP2	This register sets DRVP value of DQ[15:8], DQS1, and DM1
7-4	DRVN1	This register sets DRVN value of DQ[7:0], DQS0, and DM0
3-0	DRVP1	This register sets DRVP value of DQ[7:0], DQS0, and DM0

9.5.24. OCD impedance setting register2 (DROISR2)

This register sets impedance adjustment value.

Address	F300_0000$_H$ + 9A$_H$															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRVN4				DRVP4				DRVN3				DRVP3			
R/W	R/W				R/W				R/W				R/W			
Initial value	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1

Bit field		Description
No.	Name	
15-12	DRVN4	This register sets DRVN value of DQ[31:24], DQS3, and DM3
11-8	DRVP4	This register sets DRVP value of DQ[31:24], DQS3, and DM3
7-4	DRVN3	This register sets DRVN value of DQ[23:16], DQS2, and DM2
3-0	DRVP3	This register sets DRVP value of DQ[23:16], DQS2, and DM2

9.6. Operation

This section describes DDR2C operation.

9.6.1. DRAM initialization sequence

Initialization sequence at using DDR2SDRAM is described below.
Figure 9-2 shows initialization sequence at using DDR2SDRAM in time chart.
To proceed memory access to DDR2SDRAM, initialization sequence should be performed after power-on. During initialization sequence, DDRIF macro setting, DLL reset release in DDRIF macro, SDRAM initialization, OCD adjustment, ODT setting, and others are processed. Refer to "9.6.2 DRAM initialization procedure" for more detail of initialization sequence.

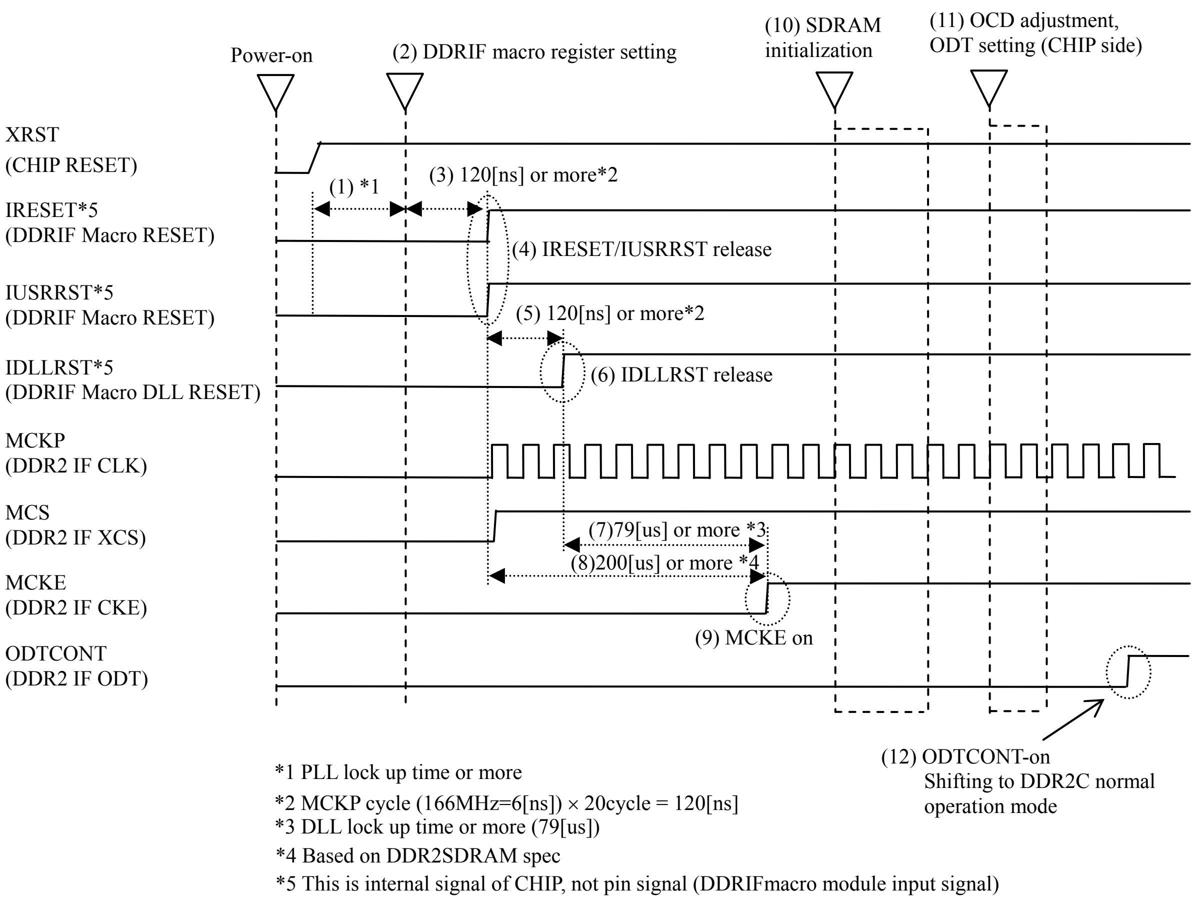


Figure 9-2 DDR2SDRAM initialization time chart

9.6.2. DRAM initialization procedure

The figure below is a whole flow of the register setting procedure for initialization sequence. Each number matches to the one in DDR2SDRAM initialization time chart shown in Figure 9-2.
The procedure showing here is only the register setting relating to the DRAM initialization.

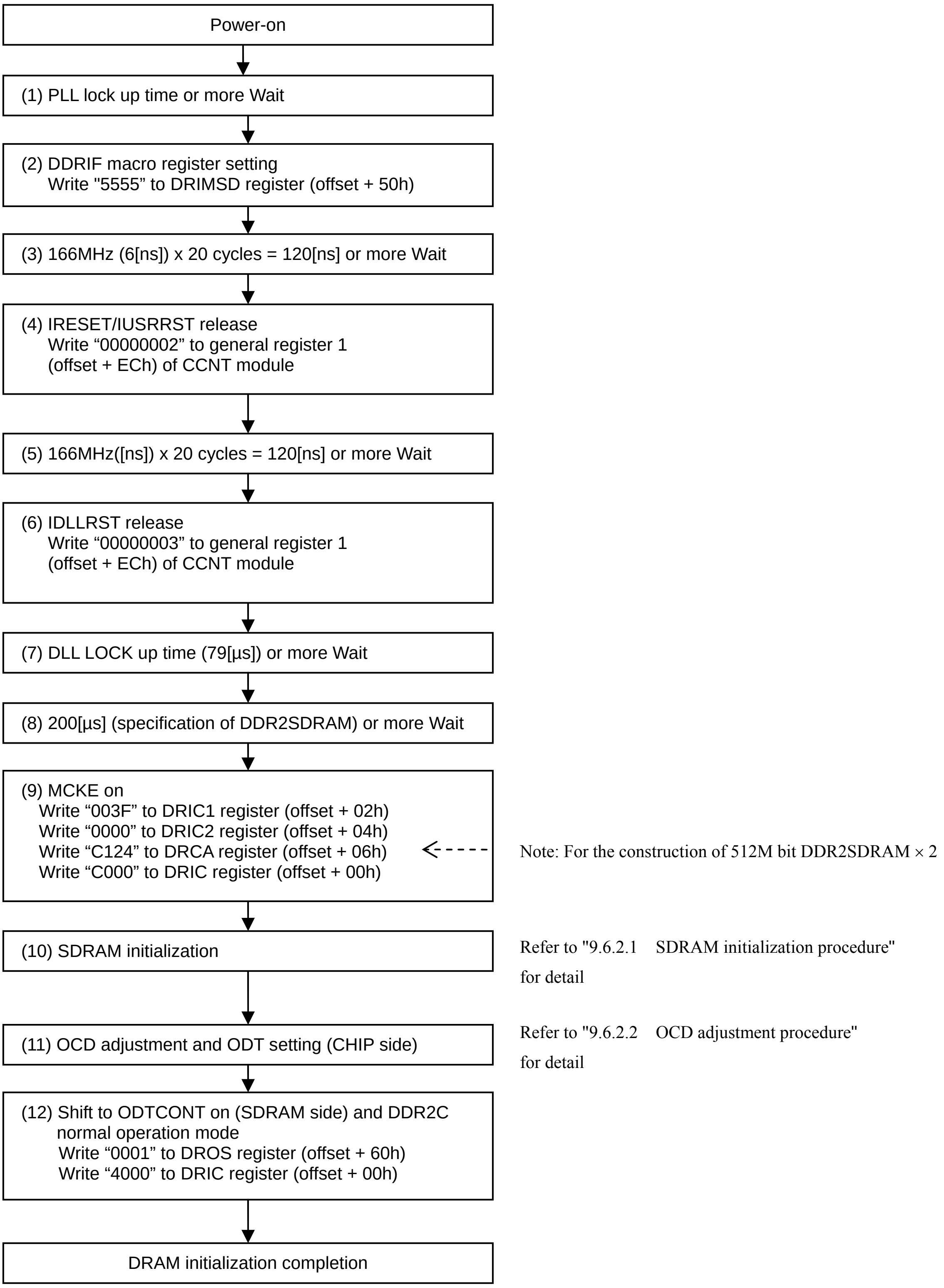


Figure 9-3 DRAM initialization procedure

9.6.2.1. SDRAM initialization procedure

The figure below is DDR2SDRAM initialization setting procedure at DRAM initialization.
DDR2SDRAM initialization sequence's command contents to be issued may change depending on the memory specification connected to this chip.
For each command's issuing contents and DDR2C command issuing timing, be sure to confirm memory spec in use to set properly.

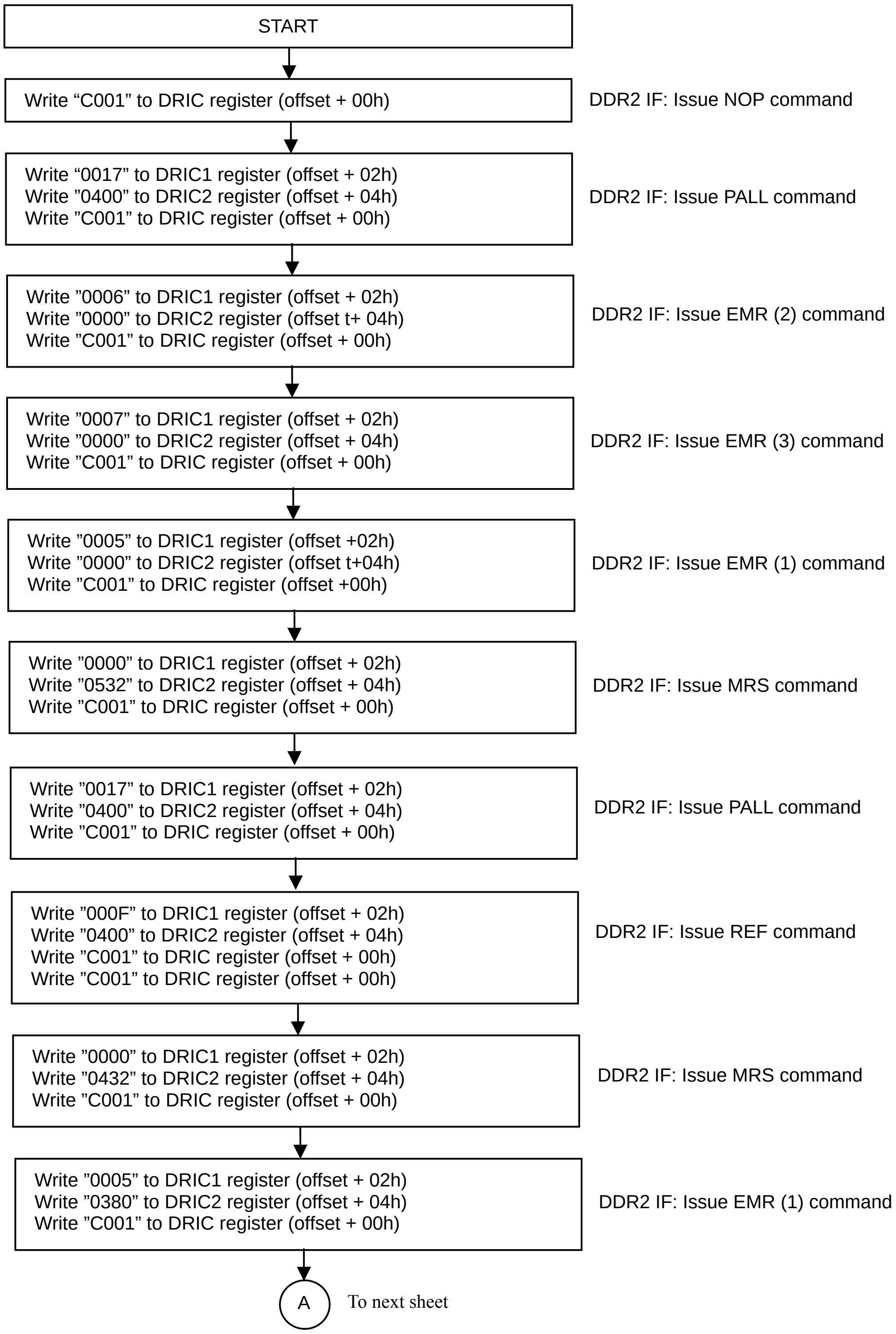


Figure 9-4 DDR2SDRAM initialization procedure

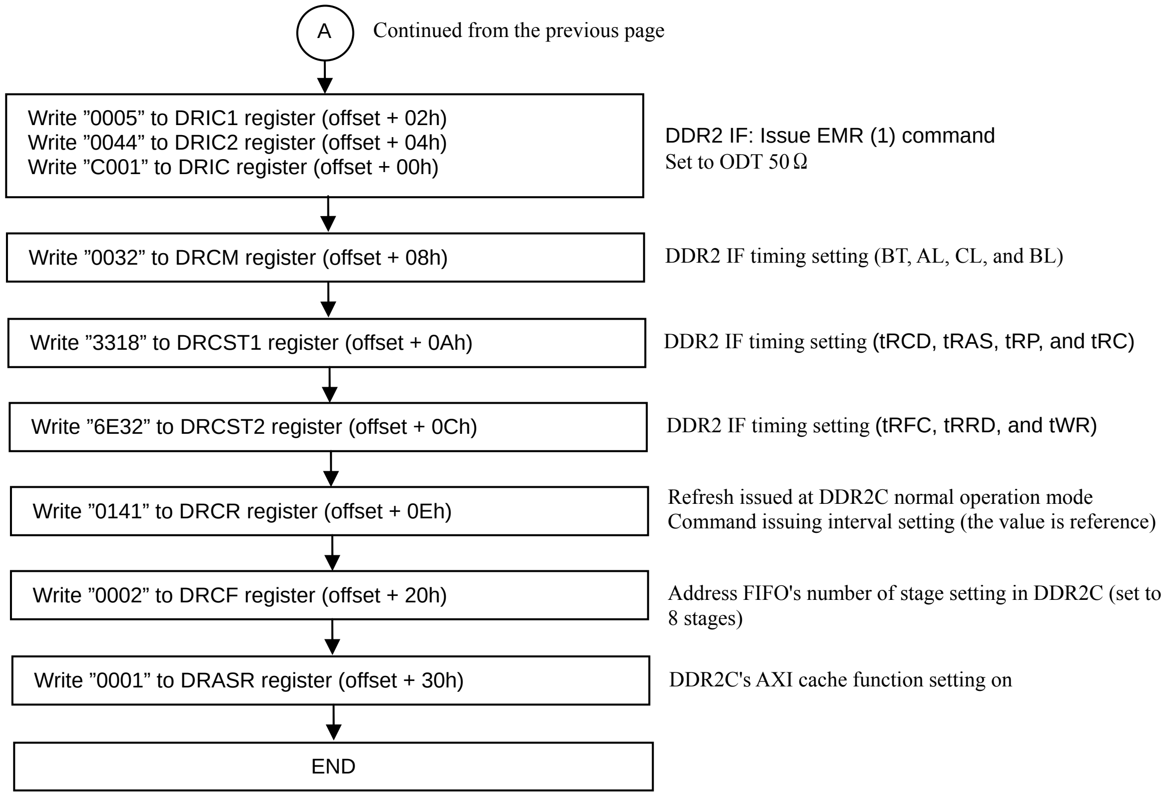


Figure 9-4 DDR2SDRAM initialization procedure

9.6.2.2. OCD adjustment procedure

The figure below is OCD adjustment setting procedure of SSTL_18 IO used for DDR2SDRSAM IF.
The setting adjusts driver output impedance of SSTL_18 IO to the optimum value.
Pin for OCD adjustment is MDQ[31:0], MDM[3:0], MDQS[3:0], and MDQSN[3:0].

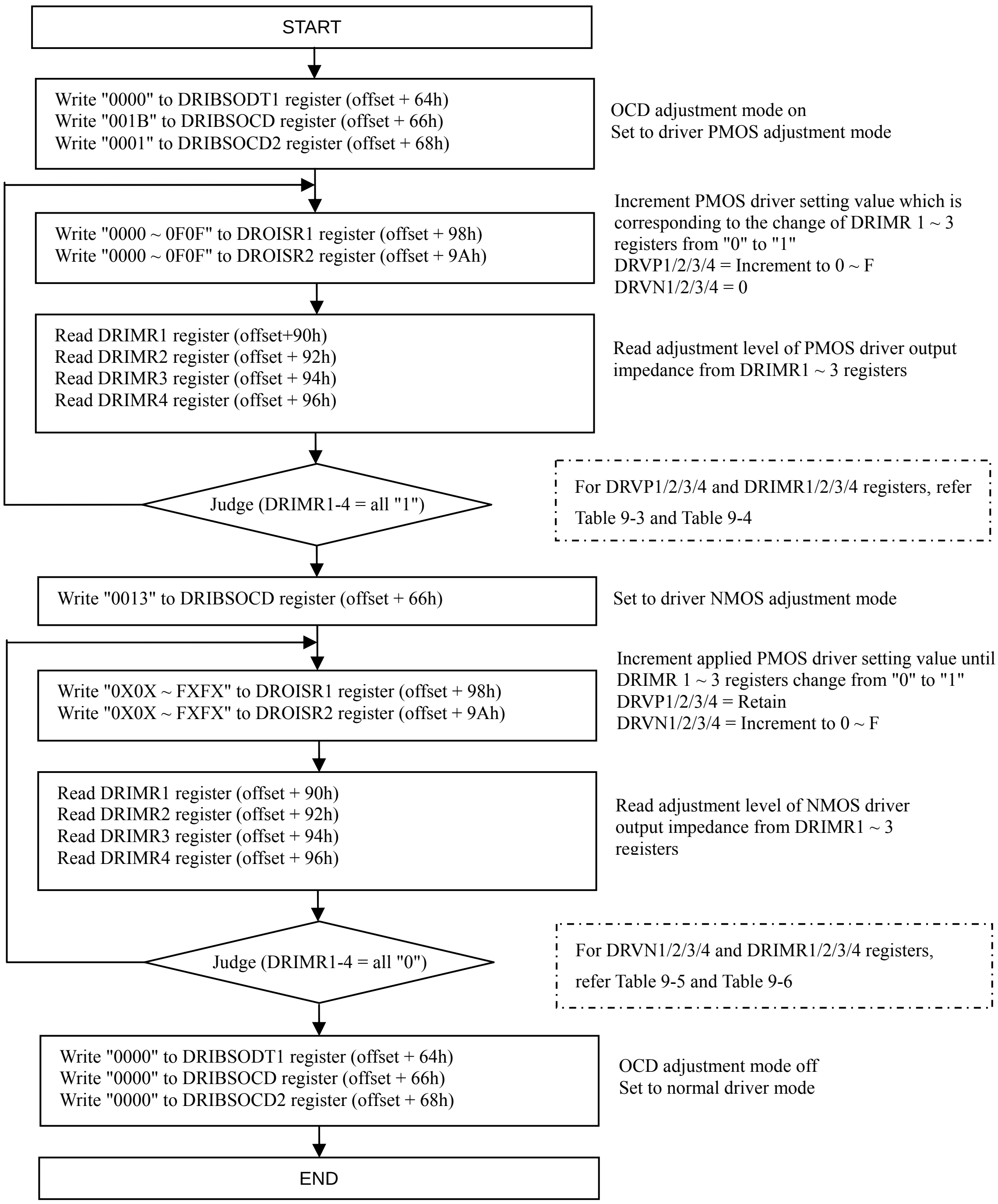


Figure 9-5 OCD adjustment setting procedure of SSTL_18 IO

Table 9-3 Correspondence table of DRVP1/2 and DRIMR1/3/4 registers

DROISR1 register		DRIMR1 register		DRIMR3 register		DRIMR4 register	
11-8	DRVP2	15-8	DQX[15:8]	1	DQSX[1]	1	DMX[1]
3-0	DRVP1	7-0	DQX[7:0]	0	DQSX[0]	0	DMX[0]

Table 9-4 Correspondence table of DRVP3/4 and DRIMR2/3/4 registers

DROISR1 register		DRIMR1 register		DRIMR3 register		DRIMR4 register	
11-8	DRVP4	15-8	DQX[31:24]	3	DQSX[3]	3	DMX[3]
3-0	DRVP3	7-0	DQX[23:16]	2	DQSX[2]	2	DMX[2]

Table 9-5 Correspondence table of DRVN1/2 and DRIMR1/3/4 registers

DROISR1 register		DRIMR1 register		DRIMR3 register		DRIMR4 register	
15-12	DRVN2	15-8	DQX[15:8]	1	DQSX[1]	1	DMX[1]
7-4	DRVN1	7-0	DQX[7:0]	0	DQSX[0]	0	DMX[0]

Table 9-6 Correspondence table of DRVN3/4 and DRIMR2/3/4 registers

DROISR1 register		DRIMR1 register		DRIMR3 register		DRIMR4 register	
15-12	DRVN4	15-8	DQX[31:24]	3	DQSX[3]	3	DMX[3]
7-4	DRVN3	7-0	DQX[23:16]	2	DQSX[2]	2	DMX[2]

9.6.2.3. ODT setting procedure

The figure below is ODT adjustment setting procedure of SSTL_18 IO used for DDR2SDRAM IF.
With proceeding ODT setting, DDR2C automatically adjusts ODT of SSTL_18 IO; moreover, auto. adjustment always operates during memory reading at normal operation.
Pin for ODT adjustment is MDQ[31:0], MDM[3:0], MDQSP[3:0], and MDQSN[3:0].

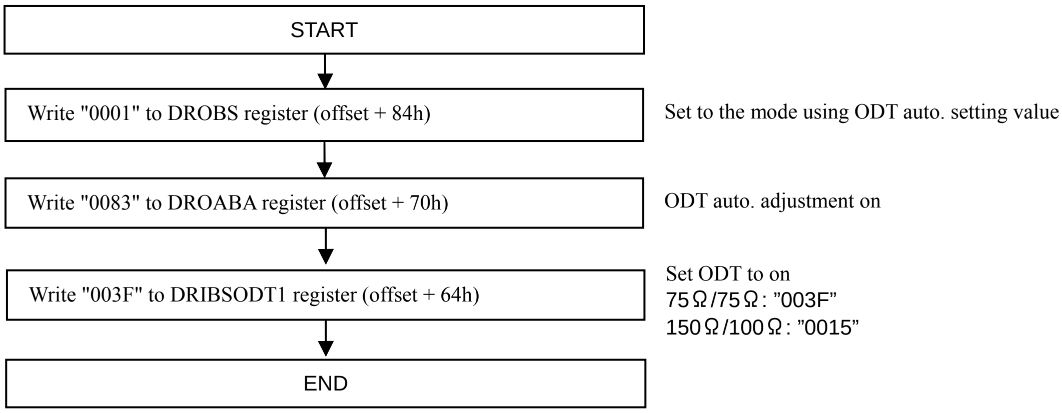


Figure 9-6 ODT adjustment setting procedure of SSTL_18 IO

10. Built-in SRAM

This chapter describes function and operation of built-in SRAM.

10.1. Outline

This SRAM equips 32KB of SRAM that enables storing instruction and data.

10.2. Feature

INTRAM has following features:

- Operation as bus slave of AMBA (AHB)
- 2pcs. of built-in SRAM are accessible from different 2 AHB masters simultaneously
- 32KB of SRAM is equipped to each built-in SRAM

10.3. Block diagram

Figure 10-1 shows block diagram of built-in SRAM.

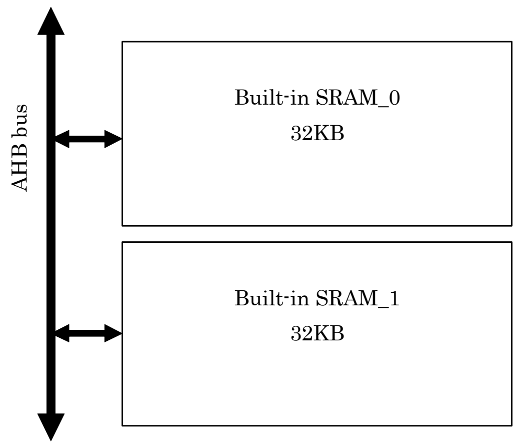


Figure 10-1 Block diagram of built-in SRAM

10.4. Supply clock

AHB clock is supplied to built-in SRAM. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

11. DMA controller (DMAC)

This chapter describes function and operation of DMA controller.

11.1. Outline

DMAC is 8 channel DMA controller.

11.2. Feature

DMAC in MB86R03 has following features:

- Compliant with AMBA v2.0
- 8 DMA channels
- DMA trigger
    - External transfer request (2ch of external DMA request and 6ch of I2S transmission/reception DMA request are available)
    - Peripheral transfer request (12 types of UART transmission/reception DMA request is selectable per channel)
    - Software request (start-up by register writing)
- Beat transfer
  16 word FIFO shared by all channels
  Corresponding to INCR, INCR 4/8/16, and WRAP 4/8/16.
- Transfer mode
    - Block transfer
    - Burst transfer
    - Demand transfer
- 4 bit block register and 16 bit count register are set by programming
- Corresponding to 8, 16, and 32 bit transfer widths
- Corresponding to increment and fixed addressing to source and destination
- Reload count, source address, and destination address register
- Issuing error interrupt and completion interrupt
- Displaying end code of DMA transfer
- Supporting source and destination protection
- Corresponding to fixed priority and rotation priority by hardware.
  In the fixed priority mode, channel 0 has the highest priority, and channel 7 has the lowest priority

11.3. Block diagram

Figure 11-1 shows block diagram of DMA controller.

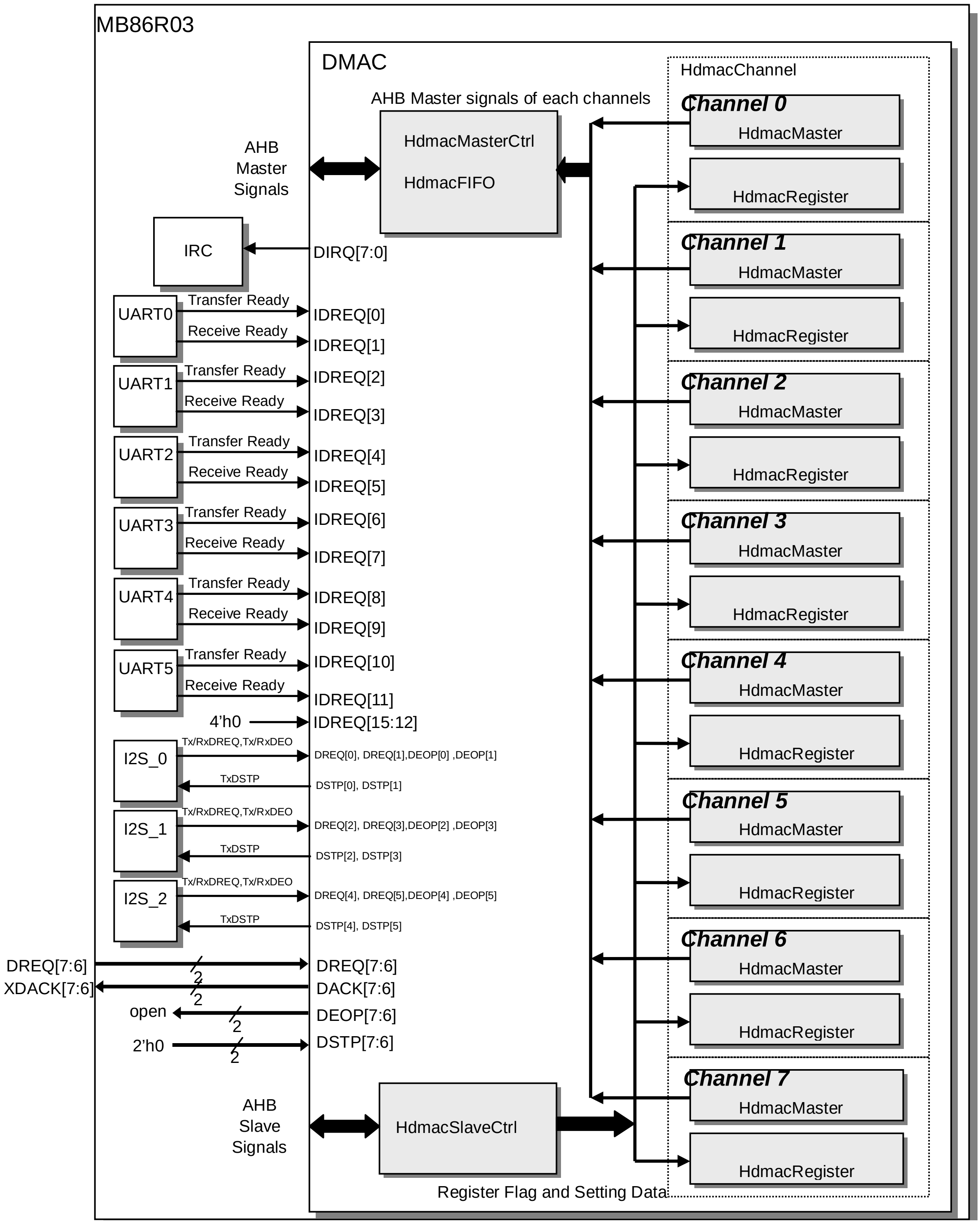


Figure 11-1 Block diagram of DMA controller

Function of individual block

Table 11-1 shows each block function of this module.

Table 11-1 Individual block function

Block	Function
DMAC	Most significant module
HdmacMasterCtrl	Valid channel selector for priority controller and AHB master transaction
HdmacSlaveCtrl	DMAC AHB slave interface controller and valid channel selector I/F for AHB slave transaction
HdmacChannel	DMAC 1 channel module DMAC has 8 modules
HdmacMaster	DMAC AHB master main controller
HdmacRegister	DMAC DMA configuration register controller
HdmacFIFO	DMAC 16 word FIFO

11.4. Related pin

DMAC of MB86R03 has following DMA related pin which is common with other functions. To use the pin, external pin should be set to MPX_MODE_1[1:0] = "LH" or MPX_MODE_1[1:0] = "HL" to select DMA related pin.

Table 11-2 DMAC related pin

Pin	Direction	Qty.	Description
DREQ[6] DREQ[7]	I	2	DMA request pin which is connected as channel 7 of DMAC and channel 6 of external DREQ signal.
XDACK[6] XDACK[7]	O	2	DMA acknowledge pin which is connected as channel 7 of DMAC and channel 6 of external DACK signal.

11.5. Supply clock

AHB clock is supplied to DMA controller. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

11.6. Register

This section describes DMAC register.

11.6.1. Register list

DMAC control related register is shown below.

Table 11-3 DMAC register list

Module	Address	Register	Function
DMAC common	FFFD0000(h)	DMACR	DMAC configuration register
	FFFD0004(h) FFFD000F(h)	Reserved	
DMAC ch0	FFFD0010(h)	DMACA0	DMAC0 configuration A register
	FFFD0014(h)	DMACB0	DMAC0 configuration B register
	FFFD0018(h)	DMACSA0	DMAC0 source address register
	FFFD001C(h)	DMACDA0	DMAC0 Destination address register
DMAC ch1	FFFD0020(h)	DMACA1	DMAC1 configuration A register
	FFFD0024(h)	DMACB1	DMAC1 configuration B register
	FFFD0028(h)	DMACSA1	DMAC1 source address register
	FFFD002C(h)	DMACDA1	DMAC1 Destination address register
DMAC ch2	FFFD0030(h)	DMACA2	DMAC2 configuration A register
	FFFD0034(h)	DMACB2	DMAC2 configuration B register
	FFFD0038(h)	DMACSA2	DMAC2 source address register
	FFFD003C(h)	DMACDA2	DMAC2 Destination address register
DMAC ch3	FFFD0040(h)	DMACA3	DMAC3 configuration A register
	FFFD0044(h)	DMACB3	DMAC3 configuration B register
	FFFD0048(h)	DMACSA3	DMAC3 source address register
	FFFD004C(h)	DMACDA3	DMAC3 Destination address register
DMAC ch4	FFFD0050(h)	DMACA4	DMAC4 configuration A register
	FFFD0054(h)	DMACB4	DMAC4 configuration B register
	FFFD0058(h)	DMACSA4	DMAC4 source address register
	FFFD005C(h)	DMACDA4	DMAC4 Destination address register
DMAC ch5	FFFD0060(h)	DMACA5	DMAC5 configuration A register
	FFFD0064(h)	DMACB5	DMAC5 configuration B register
	FFFD0068(h)	DMACSA5	DMAC5 source address register
	FFFD006C(h)	DMACDA5	DMAC5 Destination address register
DMAC ch6	FFFD0070(h)	DMACA6	DMAC6 configuration A register
	FFFD0074(h)	DMACB6	DMAC6 configuration B register
	FFFD0078(h)	DMACSA6	DMAC6 source address register
	FFFD007C(h)	DMACDA6	DMAC6 Destination address register
DMAC ch7	FFFD0080(h)	DMACA7	DMAC7 configuration A register
	FFFD0084(h)	DMACB7	DMAC7 configuration B register
	FFFD0088(h)	DMACSA7	DMAC7 source address register
	FFFD008C(h)	DMACDA7	DMAC7 Destination address register

Notice for register setting

Note followings for DMAC register setting.

- DMACR, DMACA, DMACB, DMACSA, and DMACDA registers are accessible in byte, half-word, and word size.
- Do not set DMAC register address to DMACSA and DMACDA registers.
- Do not change setting register's channel during DMA transfer except DE/DH bits of DMACR and EB/PB bits of DMACA.

Description format of register

Following format is used for description of register's each bit in "11.6.2 DMA configuration register (DMACR)" to "11.6.6 DMAC destination address register (DMACDAx)".

Address	Base address + Offset address															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

11.6.2. DMA configuration register (DMACR)

Address	FFFD_0000 + 00(h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DE	DS	-	PR	DH[3:0]				(Reserved)							
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31</td><td>DE
(DMA
Enable)</td><td>Transfer is controlled for all DMA channels.
<table><tr><td>0</td><td>All DMA channels are disabled and DMA transfer is not performed until "1" is set to this bit
If the value is cleared to "0" during the transfer, DMA is stopped at transmission gap for the channel in transfer</td></tr><tr><td>1</td><td>DMA transfer starts according to the register setting of each channel</td></tr></table>
[Transfer gap]
The transfer gap is that DMAC de-asserts bus request (HBUSREQ) to the arbiter during DMA transfer (about 4 clocks) by DMAC. Its occurrence is different by transfer mode shown below.
• Block transfer: Transfer gap occurs at BC = 0 (after completing transfer in BC unit)
• Burst transfer: There is no transfer gap.
• Demand transfer: Transfer gap occurs at TC = TC - 1 (after completing 1 DMA transfer), or at transfer request negotiation
This bit can be used to reset all channels of Configuration register at a time during DMA transfer.</td></tr>
<tr><td>30</td><td>DS
(DMA Stop)</td><td>This shows all channels of DMA transfer is stop.
<table><tr><td>0</td><td>Release of disable/halt setting</td></tr><tr><td>1</td><td>DMA transfer stop of all channels by disable/halt setting</td></tr></table>
This bit is set to "1" during DMA transfer by either of following operations:
• DMACR.DE bit is cleared to "0" (all channels are disabled)
• Value other than 4'h0 is set to DMACR.DH bit (all channels are halt)
When the state of disable/halt is cleared, DMAC clears DS bit to "0".
This bit is able to use for confirmation of transfer stop when DMAC stops transfer of all channels by disable/halt setting.</td></tr>
<tr><td>29</td><td>(Reserved)</td><td>Reserved bits. Write access is ignored.
Read value of this bit is always "0".</td></tr>
<tr><td>28</td><td>PR
(Priority
Rotation)</td><td>Prioritization procedure of DMA channel is controlled.
<table><tr><td>0</td><td>"Fixed"
Priority order: Ch0 > Ch1 > Ch2 > Ch3 > Ch4 > Ch5 > Ch6 > Ch7</td></tr><tr><td>1</td><td>"Rotation"
Priority order is rotated</td></tr></table>
Channel switch occurs by the timing of transfer gap.
Refer to DE bit description for the transfer gap.</td></tr>
</table>

Bit field		Description
No.	Name	
27-24	DH[3:0] (DMA Halt)	All channels of DMA stop are controlled. When the value other than 4'b0000 is set to this bit, all DMA channels stop and DMA is not transferred until 4'b0000 is set. If the value other than 4'b0000 is set during DMA transfer, it is stopped at transfer gap. Refer to DE bit description for the transfer gap. These bits are used to stop DMA transfer without resetting each configuration register of all channels. 0000: Stop release Other than 0000: Stop of all channels
23-0	(Reserved)	Reserved bits. Write access is ignored. Read value of this bit is always "0".

11.6.3. DMA configuration A register (DMACAx)

Address	ch0 : FFFD_0000+10 (h) ch4 : FFFD_0000+50 (h)				ch1 : FFFD_0000+20 (h) ch5 : FFFD_0000+60 (h)				ch2 : FFFD_0000+30 (h) ch6 : FFFD_0000+70 (h)				ch3 : FFFD_0000+40 (h) ch7 : FFFD_0000+80 (h)			
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EB	PB	ST	IS[4:0]					BT[3:0]				BC[3:0]			
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TC[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31	EB (Enable Bit)	This bit is used to control DMA channel transfer. When "1" is set to this bit, channel waits for the trigger to start DMA transfer (DMACR/DE bits should be set to "1" beforehand.) DMAC sets "0" to this bit after DMA transfer, then this channel is disabled and DMA transfer is not performed until "1" is set to this bit. If "0" is set to this bit during DMA transfer, DMA stops at transfer gap which is regarded as forcible termination. Refer to DMACR/DE bits description for transfer gap. This bit is able to use for resetting each configuration register of the channel during DMA transfer. 0: This channel is disabled (initial value) 1: This channel is enabled
30	PB (Pause Bit)	This bit is used to discontinue DMA channel transfer. When "1" is set to this bit, this channel stops the transfer, and it is not performed until this bit is cleared. If "1" is set to this bit during DMA transfer, DMA stops at transfer gap. Refer to DMACR/DE bits description for transfer gap. When "1" is set to this bit before receiving transfer request to acquire bus right, DMAC is immediately paused; in this case, DMAC does not hold transfer request during the pause. When "0" is set to this bit during DMA transfer is in pause, it is cleared and DMAC waits for new transfer request. This bit is able to be used to stop DMA transfer without resetting each configuration register of the channel. 0: Initial value 1: This channel is stopped
29	ST (Software Trigger)	This bit is used to generate software trigger. When "1" is set to this bit, DMA transfer starts as software request is received. After the transfer, DMAC sets "0" to this bit. If "0" is set to this bit during DMA transfer by software request, it stops at transfer gap. 0: Initial value 1: Software request

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>28-24</td><td>IS[4:0]
(Input Select)</td><td>This bit is used to select trigger for DMA transfer.

DMA transfer trigger is software request (ST = 1): Set 5’b00000 to IS bit
DMA transfer trigger is external request (DREQ): Set 5’b01110 or 5’b01111 to IS bit
DMA transfer trigger is peripheral request (IDREQ[15:0]): Set 5’b1xxxx to IS bit

External request (DREQ[7:0]) is allocated into each channel, and peripheral request (IDREQ[15:0]) is allocated into all channels. Thus, peripheral request can be selected from all channels.

<table>
<tr><th>IS[4:0]</th><th>Function</th></tr>
<tr><td>0(h)</td><td>Software request</td></tr>
<tr><td>1(h)-B(h)</td><td>Invalid</td></tr>
<tr><td>E(h)</td><td>DREQ "H" active level or rising edge</td></tr>
<tr><td>F(h)</td><td>DREQ "L" active level or falling edge</td></tr>
<tr><td>10(h)</td><td>IDREQ 0 "H" active level or rising edge</td></tr>
<tr><td>11(h)</td><td>IDREQ 1 "H" active level or rising edge</td></tr>
<tr><td>12(h)</td><td>IDREQ 2 "H" active level or rising edge</td></tr>
<tr><td>13(h)</td><td>IDREQ 3 "H" active level or rising edge</td></tr>
<tr><td>14(h)</td><td>IDREQ 4 "H" active level or rising edge</td></tr>
<tr><td>15(h)</td><td>IDREQ 5 "H" active level or rising edge</td></tr>
<tr><td>16(h)</td><td>IDREQ 6 "H" active level or rising edge</td></tr>
<tr><td>17(h)</td><td>IDREQ 7 "H" active level or rising edge</td></tr>
<tr><td>18(h)</td><td>IDREQ 8 "H" active level or rising edge</td></tr>
<tr><td>19(h)</td><td>IDREQ 9 "H" active level or rising edge</td></tr>
<tr><td>1A(h)</td><td>IDREQ 10 "H" active level or rising edge</td></tr>
<tr><td>1B(h)</td><td>IDREQ 11 "H" active level or rising edge</td></tr>
<tr><td>1C(h)</td><td>IDREQ 12 "H" active level or rising edge</td></tr>
<tr><td>1D(h)</td><td>IDREQ 13 "H" active level or rising edge</td></tr>
<tr><td>1E(h)</td><td>IDREQ 14 "H" active level or rising edge</td></tr>
<tr><td>1F(h)</td><td>IDREQ 15 "H" active level or rising edge</td></tr>
</table>

Transfer mode is block transfer or burst transfer: Rising edge is selected.
Transfer mode is demand transfer: "H" active level is selected.

[Note]
• These bits must not be the same as other channels’
• If these bits are changed at asserting DREQ/IDREQ, DMAC regards IS bit change as edge (rising edge/falling edge) detection.</td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>23-20</td><td>BT[3:0]
(Beat Type)</td><td>These bits are used to select beat transfer on AHB.

When these bits are set to Normal or Single, single source access and single destination access are alternately performed.
If these bits are set to INCR* or WRAP*, contiguous source access and contiguous destination access are alternately performed.
DMAC has 64 byte of FIFO that is shared in all channels. FIFO is used for INCR* and WRAP* DMA transfer. Refer to the AMBA specifications (v2.0) for INCR* and WRAP*.
When INCR (undefined length burst) is set, the burst length is specified by the BC bit.

<table><tr><th>BT[3:0]</th><th>Function</th></tr><tr><td>0(h)</td><td>Normal (same as Single)
(Initial value)</td></tr><tr><td>1(h)-7(h)</td><td>Invalid</td></tr><tr><td>8(h)</td><td>Single (same as Normal)</td></tr><tr><td>9(h)</td><td>INCR</td></tr><tr><td>A(h)</td><td>WRAP4</td></tr><tr><td>B(h)</td><td>INCR4</td></tr><tr><td>C(h)</td><td>WRAP8</td></tr><tr><td>D(h)</td><td>INCR8</td></tr><tr><td>E(h)</td><td>WRAP16</td></tr><tr><td>F(h)</td><td>INCR16</td></tr></table>
While DMACB/MS are set to block transfer and burst transfer, fixed length burst (INCR*, WRAP*) and undefined length burst (INCR) are valid.
When DMACB/MS are set to demand transfer, BT should be set to Normal or Single.</td></tr>
<tr><td>19-16</td><td>BC[3:0]
(Block Count)</td><td>These bits are used to specify number of block for block/burst transfer. When transfer mode is demand transfer, be sure to set 4'b0000 to BC. Max. block quantity is 16 (Fh.)
These bits are valid when beat transfer type is Normal, Single, or INCR. When other types of beat (fixed length burst and lap) are set, these bits are ignored. In addition, they are able to be read during DMA transfer. After single source access and single destination access are properly completed, normally BC bit is decremented for 1.

[Note]
These bits are settable even beat type bit (BT[3:0]) is INCR, however, read data of BC after starting DMA transfer is always 4’h0 in INCR DMA transfer so that BC does not need to be monitored during the transfer.

After DMA transfer is completed properly, DMAC sets 4’b0000 to these bits.

<table><tr><th>BC[3:0]</th><th>Function</th></tr><tr><td>x(h)</td><td>Number of block (initial value: 4’b0000)</td></tr></table></td></tr>
<tr><td>15-0</td><td>TC[15:0]
(Transfer Count)</td><td>These bits are used to specify number of block/burst/demand transfer. Max. number of transfer is 65536 (FFFFh.) Any kind of bit type is valid for BT.

These bits are readable during DMA transfer. After BC becomes "0" and DMA transfer is properly completed, normally TC bit is decremented for 1 in the Normal or Single mode (BT = Normal or Single.) In other beat transfer modes (INCR, INCR*, and WRAP*), TC bit is decremented for 1 after completing consecutive source/destination access operation (for example, when 4 consecutive source accesses and 4 consecutive destination accesses are completed, INCR4’s TC bit is decremented for 1.)

After DMA transfer is completed properly, DMAC sets 16’h0000 to these bits.

<table><tr><th>TC[3:0]</th><th>Function</th></tr><tr><td>x(h)</td><td>Number of transfer (initial value: 16’h0000)</td></tr></table></td></tr>
</table>

11.6.4. DMA configuration B register (DMACBx)

Address	ch0 : FFFD_0000+14 (h) ch4 : FFFD_0000+54 (h)				ch1 : FFFD_0000+24 (h) ch5 : FFFD_0000+64 (h)				ch2 : FFFD_0000+34 (h) ch6 : FFFD_0000+74 (h)				ch3 : FFFD_0000+44 (h) ch7 : FFFD_0000+84 (h)			
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TT[1:0]		MS[1:0]		TW[1:0]		FS	FD	RC	RS	RD	EI	CI	SS[2:0]		
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/W0	R/W0	R/W0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SP[3:0]				DP[3:0]				(Reserved)							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-30	TT[1:0] (Transfer Type)	These bits are used to specify transfer type. Currently, only 2 cycle transfer mode is available for DMAC. **TT[1:0]** / **Function** 0(h): 2 cycle transfer (initial value) Other than 0(h): Reserved
29-28	MS[1:0] (Mode Select)	These bits are used to select transfer mode. **MS[1:0]** / **Function** 0(h): Block transmission mode (initial value) 1(h): Burst transmission mode 2(h): Demand transmission mode 3(h): Reserved
27-26	TW[1:0] (Transfer Width)	These bits are used to specify transfer data width. HSIZE of DMAC issues this value on AHB. **TW[1:0]** / **Function** 0(h): Byte (initial value) 1(h): Half-word 2(h): Word 3(h): Reserved
25	FS (Fixed Source)	This bit is used to fix source address. When the address needs to be added after each transfer, "0" must be set to this bit. **FS** / **Function** 0(h): Source address is incremented (initial value) 1(h): Source address is fixed
24	FD (Fixed Destination)	This bit is used to fix destination address. When the address needs to be added after each transfer, "0" must be set to this bit. **FD** / **Function** 0(h): Destination address is incremented (initial value) 1(h): The destination address is fixed

Bit field No.	Bit field Name	Description
23	RC (Reload Count)	This bit is used to control reload function for number of block (DMACA/BC bits) and number of transfer (DMACA/TC bits.) When "1" is set to this bit, DMACA/BC and DMACA/TC are set to the initial value after DMA transfer. **RC** / **Function** 0(h): Reload function for number of transfer is disabled (initial value) 1(h): Reload function for number of transfer is enabled
22	RS (Reload Source)	This bit is used to control reload function of source address (DMACSA.) "1" is set to this bit: DMACSA is set to the initial value after DMA transfer "0" is set to this bit: DMAC sets the next source address to DMACSA after DMA transfer **RS** / **Function** 0(h): Reload function of source address is disabled (initial value) 1(h): Reload function of source address is enabled
21	RD (Reload Destination)	This bit is used to control reload function of destination address (DMACDA.) "1" is set to this bit: DMACDA is set to the initial value after DMA transfer "0" is set to this bit: DMAC sets the next destination address to DMACDA after DMA transfer **RD** / **Function** 0(h): Reload function of destination address is disabled (initial value) 1(h): Reload function of destination address is enabled
20	EI (Error Interrupt)	This bit is used to control issuing interrupt (DIRQ) caused by error. When this bit is set to "1", error interrupt is issued by the following transfer errors. • Address overflow • Transfer stop request from DSTP and IDSTP, or transfer disable with EB or DE bit • Source access error • Destination access error **EI** / **Function** 0(h): Error interrupt issue is disabled (initial value) 1(h): Error interrupt issue is enabled
19	CI (Completion Interrupt)	This bit is used to control issuing interrupt (DIRQ) caused by completion of transfer. When this bit is set to "1", completion interrupt is issued after DMA is transferred properly. **CI** / **Function** 0(h): Completion interrupt is disabled (initial value) 1(h): Completion interrupt is enabled

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>18-16</td><td>SS[2:0]
(Stop Status)</td><td>These bits are used to show end code of DMA transfer which is shown below.
These bits are also used to release interrupt (DIRQ) which is performed by writing 3'b000 to these bits when interrupt becomes error or it is issued by normal termination.
<table>
<tr><th>SS</th><th>Function</th><th>Status type</th></tr>
<tr><td>0(h)</td><td>Initial value</td><td>None</td></tr>
<tr><td>1(h)</td><td>Address overflow</td><td>Error</td></tr>
<tr><td>2(h)</td><td>Transfer stop request</td><td>Error</td></tr>
<tr><td>3(h)</td><td>Source access error</td><td>Error</td></tr>
<tr><td>4(h)</td><td>Destination access error</td><td>Error</td></tr>
<tr><td>5(h)</td><td>Normal termination</td><td>End</td></tr>
<tr><td>6(h)</td><td>Reserved</td><td></td></tr>
<tr><td>7(h)</td><td>DMA discontinuance</td><td>None</td></tr>
</table>
When various errors occur at the same time, end code is displayed by the following priority.
High priority
Reset
Clear by 3'b000 writing
Address overflow
Demand stop
Source access error
Destination access error
Low priority</td></tr>
<tr><td>15-12</td><td>SP[3:0]
(Source Protection)</td><td>These bits are used to control source protection.
HPROT at source access issues this value to AHB; however, it is not performed if source target does not equip protection function.
<table>
<tr><th>SP</th><th>Function</th></tr>
<tr><td>x(h)</td><td>Protection code (initial value: 4'b0000.)</td></tr>
</table></td></tr>
<tr><td>11-8</td><td>DP[3:0]
(Destination Protection)</td><td>These bits are used to control destination protection.
HPROT at destination access issues this value to AHB; however, it is not performed if source target does not equip protection function.
<table>
<tr><th>DP</th><th>Function</th></tr>
<tr><td>x(h)</td><td>Protection code (initial value: 4'b0000.)</td></tr>
</table></td></tr>
<tr><td>7-0</td><td>(Reserved)</td><td>Reserved bits.
Write access is ignored. Read value of this bit is always "0".</td></tr>
</table>

11.6.5. DMAC source address register (DMACSAx)

Address	ch0 : FFFD_0000+18 (h) ch4 : FFFD_0000+58 (h)				ch1 : FFFD_0000+28 (h) ch5 : FFFD_0000+68 (h)				ch2 : FFFD_0000+38 (h) ch6 : FFFD_0000+78 (h)				ch3 : FFFD_0000+48 (h) ch7 : FFFD_0000+88 (h)			
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMACSA[31:16]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMACSA[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-0	DMACSA[31:0] (DMAC Source Address)	These bits are used to specify source address to start DMA transfer, and they are able to be read during DMA transfer. When fixed address function (DMACB/FS) is disabled, these bits are incremented according to the transfer width (DMACB/TB) after completing source address properly. After the DMA transfer, DMAC sets the next source address to these bits. [Note] It is prohibited to set DMAC register address to DMACSA.

DMACSA	Function
x(h)	Source address to start DMA transfer (Initial value: 32'h00000000)

11.6.6. DMAC destination address register (DMACDAx)

Address	ch0 : FFFD0000+1C (h) ch4 : FFFD0000+5C (h)				ch1 : FFFD0000+2C (h) ch5 : FFFD0000+6C (h)				ch2 : FFFD0000+3C (h) ch6 : FFFD0000+7C (h)				ch3 : FFFD0000+4C (h) ch7 : FFFD0000+8C (h)			
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMACDA[31:16]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMACDA[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	**Name**	
31-0	DMACDA[31:0] (DMAC Destination Address)	These bits are used to specify destination address to start DMA transfer, and they are able to be read during DMA transfer. When fixed address function (DMACB/FD) is disabled, these bits are incremented according to the transfer width (DMACB/TB) after completing destination address properly. After DMA transfer, DMAC sets the next destination address to these bits. [Note] It is prohibited to set DMAC register address to DMACDA.

DMACDA	Function
x(h)	Destination address to start DMA transfer (Initial value: 32'h00000000)

11.7. Operation

This section describes operation of DMAC.

11.7.1. Transfer mode

DMAC has 3 types of transfer modes, and they are set with DMACB.MS[1:0].

11.7.1.1. Block transfer

Operation

In the block transfer mode, DMA transfer specified by number of block (DMACA/BC) is executed by 1 transfer request. When number of transfer (DMACA/TC) is set to other values than "0", TC is decremented for 1 after completing DMA transfer of BC. After the last transfer (BC is 4'h0 and TC is 16'h0000), DMA transfer is completed.

Transfer gap

After completing BC transfer, DMAC negates bus request to arbiter for the moment in the block transfer mode. This operation prevents DMAC from occupying the bus.
Transfer gap is able to be used to reflect register setting (e.g. disable/interruption setting) to DMAC during DMA transfer.

Transfer request

Software requirement, external request (DREQ), and peripheral request (IDREQ) are valid in this mode.

- Software request
  Set "1" to DMACA/ST and set 5'b00000 to DMACA/IS
- External request
  Set "0" to DMACA/ST, and set 5'b01110 (rising edge of transfer request) or 5'b01111 (falling edge of transfer request) to DMACA/IS
- Peripheral request
  Set "0" to DMACA/ST, and set 5'b1**** (rising edge of transfer request) to DMACA/IS

When external request or peripheral request is selected, DMAC detects transfer request edge. When BC's DMA transfer is executed by either of those requests, DMAC is unable to detect the next transfer; however, it is able to detect the next transfer request after BC's DMA transfer is completed.

Restrictions

When DMA transfer is performed by external (DREQ) or peripheral (IDREQ) request, there are restrictions for external and peripheral signal pins.

1. DREQ/IDREQ
   DREQ/IDREQ must be asserted at least 2 cycles of AHB clock（HCLK).
   There is no restriction for timing of negating DREQ/IDREQ.
   After asserting DACK/IDACK, DMAC is able to accept new transfer request (edge of DREQ/IDREQ) for the next DMA transfer.

2. DACK/IDACK
   After DMAC transfers data to the destination address, DACK/IDACK are asserted during 1 cycle of AHB clock (HCLK). When access to the destination is proceeded properly, this signal is asserted. If destination issues error, retry, or split responses at AHB, it is not asserted.

   In the block transfer mode, these signals indicate DMAC properly performs destination access.

3. DEOP/IDEOP
   Basically, DEOP/IDEOP asserted for 1 AHB clock (HCLK) cycle when DMAC terminates DMA transfer properly or abnormally. Abnormal DMA transfer includes following cases:
   - Forced termination by DSTP/IDSTP
   - Forced termination by setting 1'b0 to DMACA/EB
   - Receiving error response from source/destination

4. DSTP/IDSTP
   DSTP/IDSTP are used to forcibly terminate DMA transfer, and asserting them during the transfer is valid (it is also valid to assert DSTP/IDSTP while DMA is not transferred due to transfer gap and interruption function.)
   When these signals are used to forcibly terminate DMA transfer, they are not asserted until DEOP/IDEOP are asserted.

5. Exceptional operation of DEOP/IDEOP
   When DSTP/IDSTP are asserted immediately after asserting DREQ/DSTP, DMAC may request bus to execute IDLE transfer. In this case, DMAC may assert DEOP/IDEOP for 2 cycles or more of AHB clock (HCLK.)
   The asserting period of DEOP/IDEOP depends on number of previous master transfer cycle. Figure 11-2 shows example of this exceptional operation.

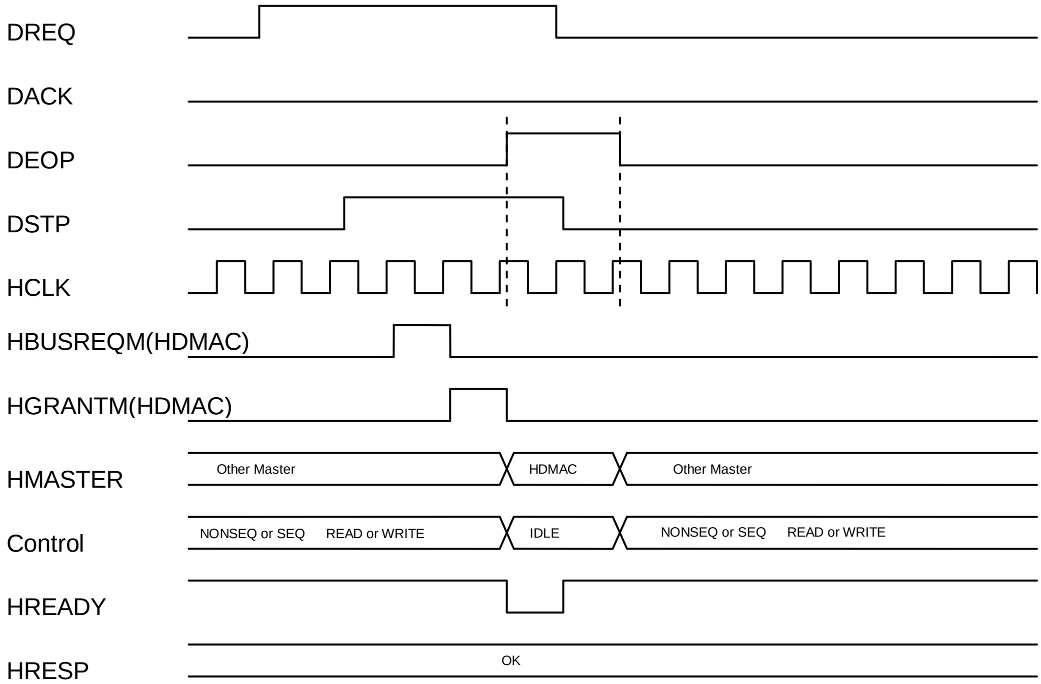


Figure 11-2 Example of exceptional operation for DEOP/IDEOP

When DMA transfer is performed by software reset, DREQ/IDREQ, DACK/IDACK, DEOP/IDEOP, and DSTP/IDSTP are not valid.

Timing chart

Figure 11-3 shows block transfer in timing chart.

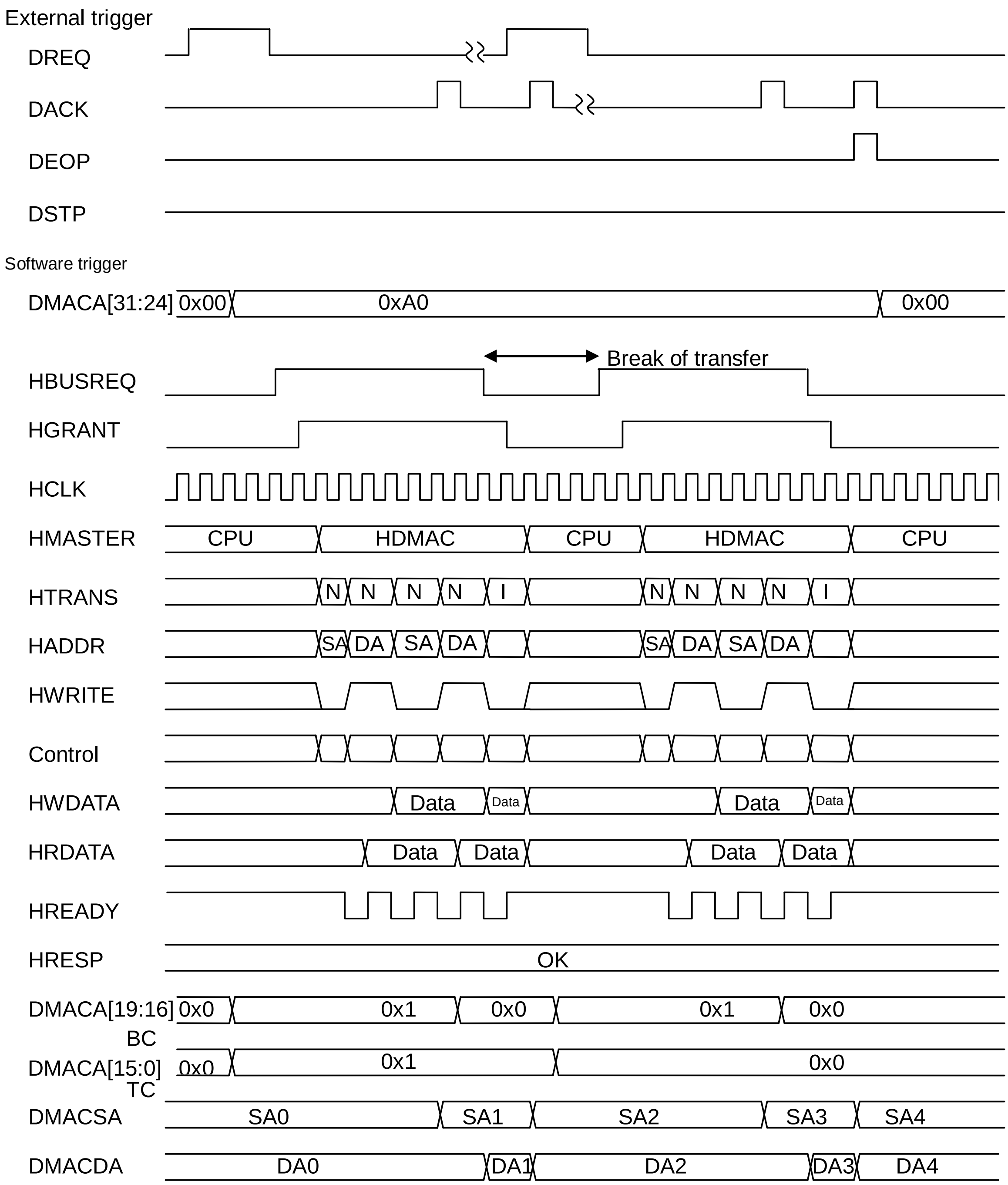


Figure 11-3 Block transfer (for BC = 0x1 and TC = 0x1)

11.7.1.2. Burst transfer

Operation

In the burst transfer mode, DMA transfer is executed for number of block multiplied by number of transfer (DMACA/BC × DMACA/TC) with 1 request.
When number of transfer (DMACA/TC) is set to other values than "0", TC is decremented for 1 after completing DMA transfer. After the last transfer (BC is 4'h0 and TC is 16'h0000), DMA transfer is completed.

Transfer gap

After completing DMA transfer, DMAC negates bus request to arbiter that transfer gap does not occur in the burst transfer mode.
Register setting change during DMA transfer (e.g. disable/interruption setting) is reflected after completing DMA transfer.

Transfer request

Software request, external (DREQ), and peripheral (IDREQ) requests are valid in this mode.

- Software request
  Set "1" to DMACA/ST and set 5'b00000 to DMACA/IS
- External request
  Set "0" to DMACA/ST, and set 5'b01110 (rising edge of transfer request) or 5'b01111 (falling edge of transfer request) to DMACA/IS
- Peripheral request
  Set "0" to DMACA/ST, and set 5'b1**** (rising edge of transfer request) to DMACA/IS

When external request or peripheral request is selected, DMAC detects transfer request edge. When DMA transfer of BC × TC is executed by either of those requests, DMAC is unable to detect the next transfer; however, it is able to detect the next transfer request after DMA transfer of BC × TC is completed.

Restrictions

When DMA transfer is performed by external (DREQ) and peripheral (IDREQ) requests, there are some restrictions for external and peripheral signal pins.

1. DREQ/IDREQ
   DREQ/IDREQ must be asserted at least 2 cycles of AHB clock（HCLK.)
   There is no restriction for timing of negating DREQ/IDREQ.
   After completing DMA transfer in BC × TC and asserting DACK/IDACK and DEOP/IDEOP, new transfer request (edge of DREQ/IDREQ) is able to be accepted for the next DMA transfer.

2. DACK/IDACK
   After DMAC transfers data to the destination address, DACK/IDACK are asserted for 1 cycle of AHB clock (HCLK.)　When access to the destination is proceeded properly, this signal is asserted.
   If destination issues error, retry, or split responses at AHB, this signal is not asserted.

   In the burst transfer mode, these signals indicate that DMAC performs destination access properly.

3. DEOP/IDEOP
   Basically, DEOP/IDEOP are asserted for 1 AHB clock (HCLK) cycle when DMAC ends DMA transfer properly or abnormally.　Abnormal DMA transfer includes following cases:
   - Forced termination by DSTP/IDSTP
   - Forced termination by setting 1'b0 to DMACA/EB
   - Receiving error response from source/destination

4. DSTP/IDSTP
   DSTP/IDSTP are used to forcibly terminate DMA transfer, and asserting them while the transfer is valid (it is also valid to assert DSTP/IDSTP while DMA is not transferred due to transfer gap and interruption function.)
   When these signals are used to forcibly terminate DMA transfer, they are not asserted until DEOP/IDEOP are asserted.

5. Exceptional operation of DEOP/IDEOP
   When DSTP/IDSTP are asserted immediately after DREQ/DSTP are asserted, DMAC may request bus to execute IDLE transfer.　In this case, DMAC may assert DEOP/IDEOP for 2 cycles or more of AHB clock (HCLK.)
   The asserting period of DEOP/IDEOP depends on number of previous master transfer cycle.　Figure 11-4 shows example of this exceptional operation.

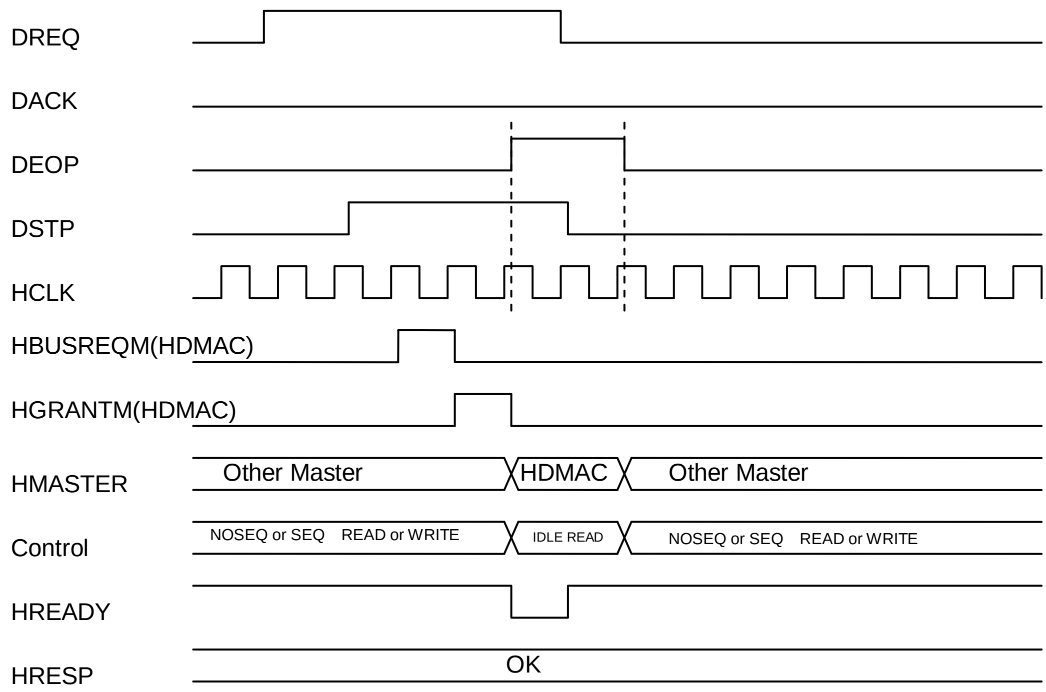


Figure 11-4 Example of exceptional operation of DEOP/IDEOP

When DMA transfer is performed by software reset, DREQ/IDREQ, DACK/IDACK, DEOP/IDEOP, and DSTP/IDSTP are not valid.

Timing chart

Figure 11-5 shows burst transfer in timing chart.

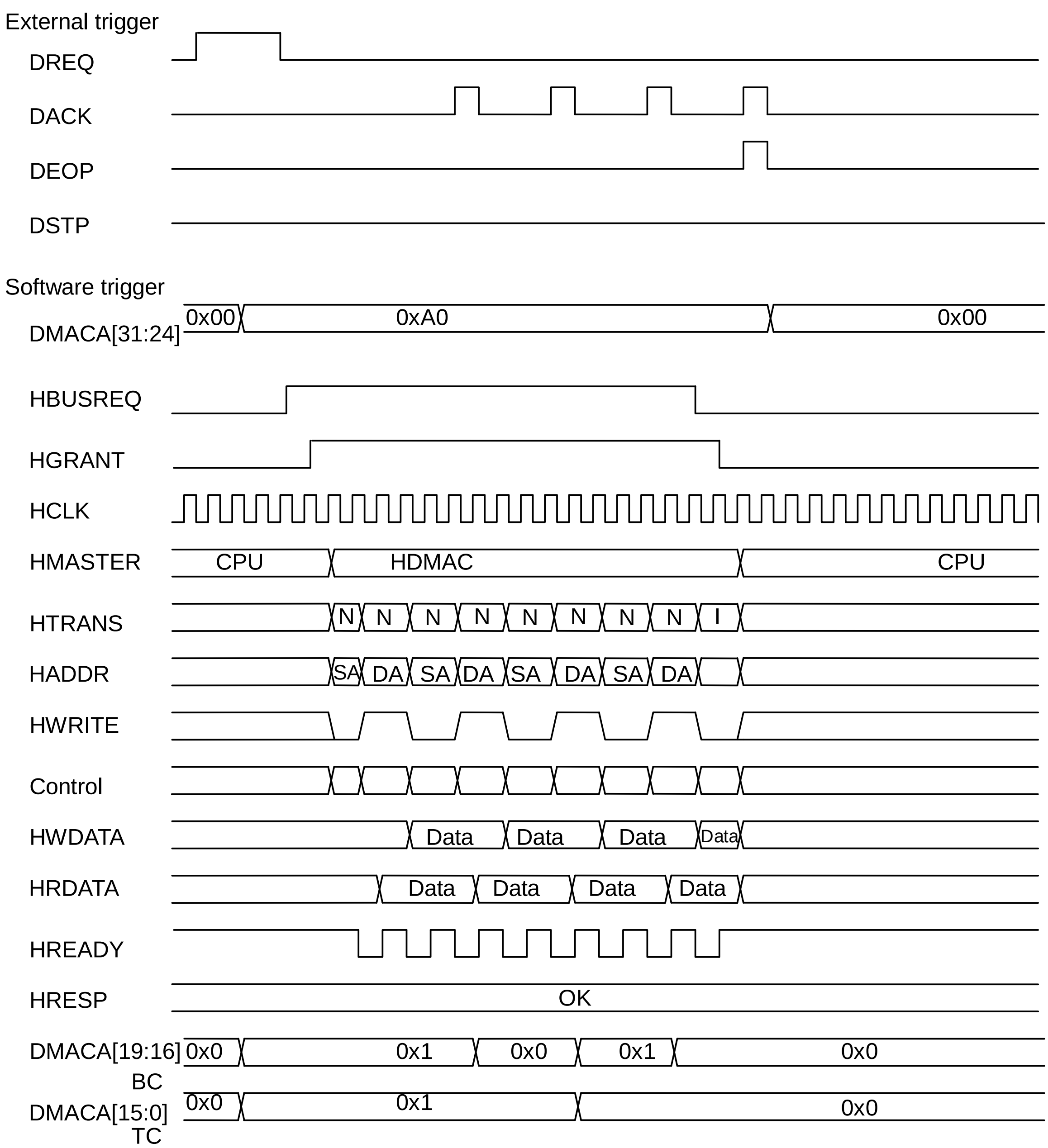


Figure 11-5 Burst transmission (for BC = 0x1 and TC = 0x1)

11.7.1.3. Demand transfer

Operation

In the demand transfer mode, DMA transfer is executed for 1 time transfer when transfer request is asserted, and number of transfer is set to DMACA/TC registers. In this case, DMACA/BC is set to "0". In this mode, DMACA/BC values are ignored. DMACA/TC are decremented for 1 after completing DMA transfer. Therefore DMA transfer ends after the last transfer (TC is16’h0000) is completed.

Transfer gap

After completing 1 transfer, DMAC negates bus request to arbiter for the moment even though transfer request is asserted. This operation prevents DMAC from occupying bus.
Transfer gap is able to be used to reflect register setting (e.g. disable/interruption setting) to DMAC during DMA transfer.

Transfer request

External (DREQ) and peripheral (IDREQ) requests are valid in the demand transfer mode; however, software request setting is prohibited in this mode.

- External request
  Set "0" to DMACA/ST, and set 5'b01110 (H level of transfer request) or 5'b01111 (L level of transfer request) to DMACA/IS
- Peripheral request
  Set "0" to DMACA/ST, and set 5'b1**** (H level of transfer request) to DMACA/IS

When external request or peripheral request is selected, DMAC detects transfer request level.

Restrictions

When DMA transfer is performed by external (DREQ) or peripheral (IDREQ) request, there are some restrictions for the external and peripheral signal pins.

1. DREQ/IDREQ
   DREQ/IDREQ must be asserted until DACK/IDACK are asserted. After they are asserted, DREQ/IDREQ need to be negated within AHB clock (HCLK) cycle of "source access cycle + destination access cycle – 1".
   When negation timing of DREQ/IDREQ is sent against to the restrictions, DMAC may start the next transfer operation.
   After completing 1 DMATE transfer and DACK/IDACK are asserted, DMAC is able to receive new transfer request (DREQ/IDREQ level) for the next DMA transfer after the condition of negating time indicated above.

2. DACK/IDACK
   After DMAC transfers control signal to the source address, DACK/IDACK are asserted during 1 cycle of AHB clock (HCLK.) In the demand transfer mode, these signals indicate that DMAC receives demand transfer request.

3. DEOP/IDEOP
   Basically, DEOP/IDEOP are asserted for 1 AHB clock (HCLK) cycle when DMAC ends DMA transfer properly or abnormally. Abnormal DMA transfer includes following cases:
   - Forced termination by DSTP/IDSTP
   - Forced termination by setting 1'b0 to DMACA/EB
   - Receiving error response from source/destination

4. DSTP/IDSPT
   DSTP/IDSTP are used to forcibly terminate DMA transfer. Asserting them during DMA transfer is valid (it is also valid to assert DSTP/IDSTP while DMA is not transferred due to transfer gap and interrupt function.)
   When these signals are used to forcibly terminate DMA transfer, they are not asserted until DEOP/IDEOP are asserted.

5. Exceptional operation of DEOP/IDEOP
   When DSTP/IDSTP are asserted immediately after DREQ/DSTP are asserted, DMAC may request bus to execute IDLE transfer. In this case, DMAC may assert DEOP/IDEOP for 2 cycles or more of AHB clock (HCLK.)
   The asserting period of DEOP/IDEOP depends on number of previous master transfer cycle. Figure 11-6 shows example of this exceptional operation.

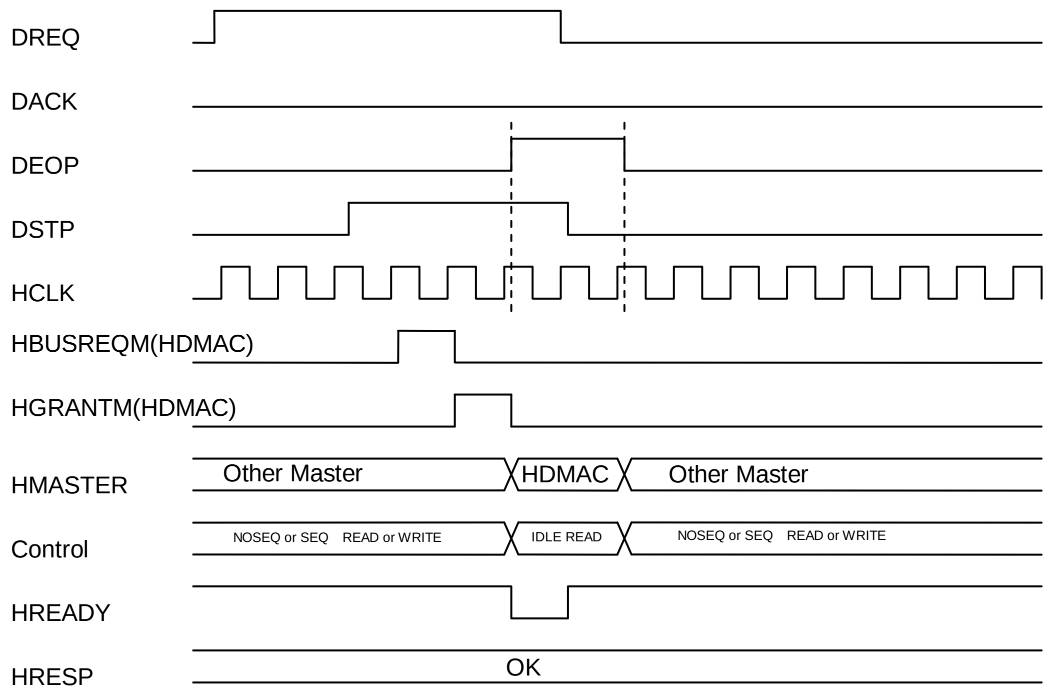


Figure 11-6 Example of exceptional operation of DEOP/IDEOP

Timing chart

Figure 11-7 shows demand transfer in timing chart.

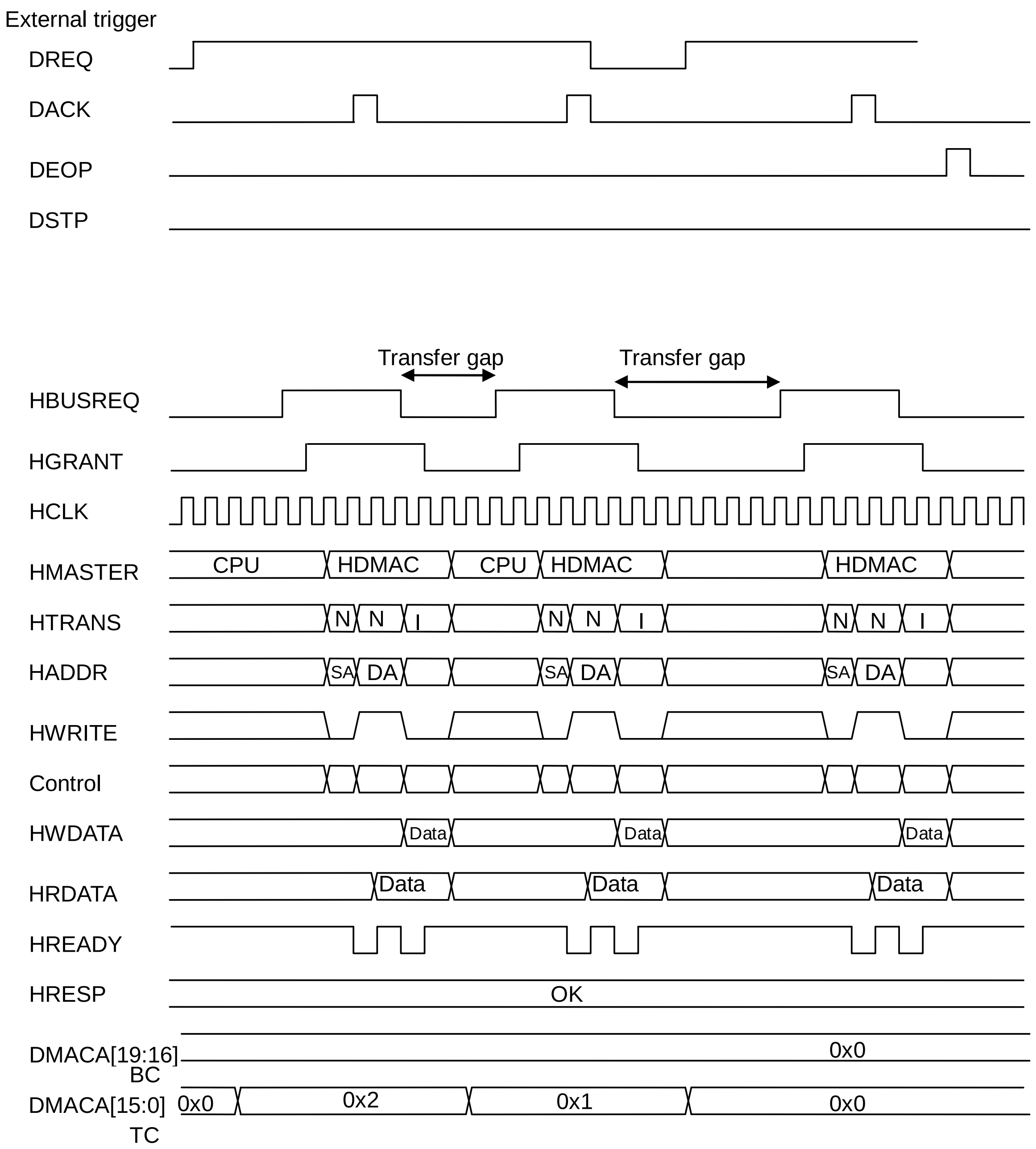


Figure 11-7 Demand transfer (for BC = 0x0 (should be 0) and TC = 0x2)

11.7.2. Beat transfer

DMAC supports beat transfer which means, in this case, increment/lap burst of the AMBA standard. DMAC has 64 byte FIFO shared in all channels, and enables sequential source access and destination access. The beat transfer type is set by DMACA/BT bits.
Correlation to DMACA/BT and AHB of HBURST is shown below.

Table 11-4 DMACA/BT and HBURST

DMACA/BT	Beat transfer type	HBURST	DMACA/MS (mode select)		
			Block	Burst	Demand
4'b0000	Normal	Single	OK	OK	OK
4'b1000	Single	Single	OK	OK	OK
4'b1001	INCR	INCR	OK	OK	NG
4'b1010	WRAP4	WRAP4	OK	OK	NG
4'b1011	INCR4	INCR4	OK	OK	NG
4'b1100	WRAP8	WRAP8	OK	OK	NG
4'b1101	INCR8	INCR8	OK	OK	NG
4'b1110	WRAP16	WRAP16	OK	OK	NG
4'b1111	INCR16	INCR16	OK	OK	NG

In the demand transfer, increment/lap burst (INCR* and WRAP*) is unsupported.

11.7.2.1. Normal and Single transfer

Normal and Single transfer methods are the same. Single source access and single destination access are executed alternately as shown in Figure 11-2 and Figure 11-3.

11.7.2.2. Increment and lap transfer

When increment beat transfer (INCR, INCR4, INCR8 and INCR16) or lap beat transfer (WRAP4, WRAP8, and WRAP16) is set to DMACA/BT, sequential source access and destination access are executed by using 64 byte FIFO of DMAC.

For the case of INCR4 (DMACA/BT = 4'b1011), DMAC performs 4 sequential source accesses.
Output data from the source is stored in FIFO of DMAC, then the data is driven to destination in sequence.

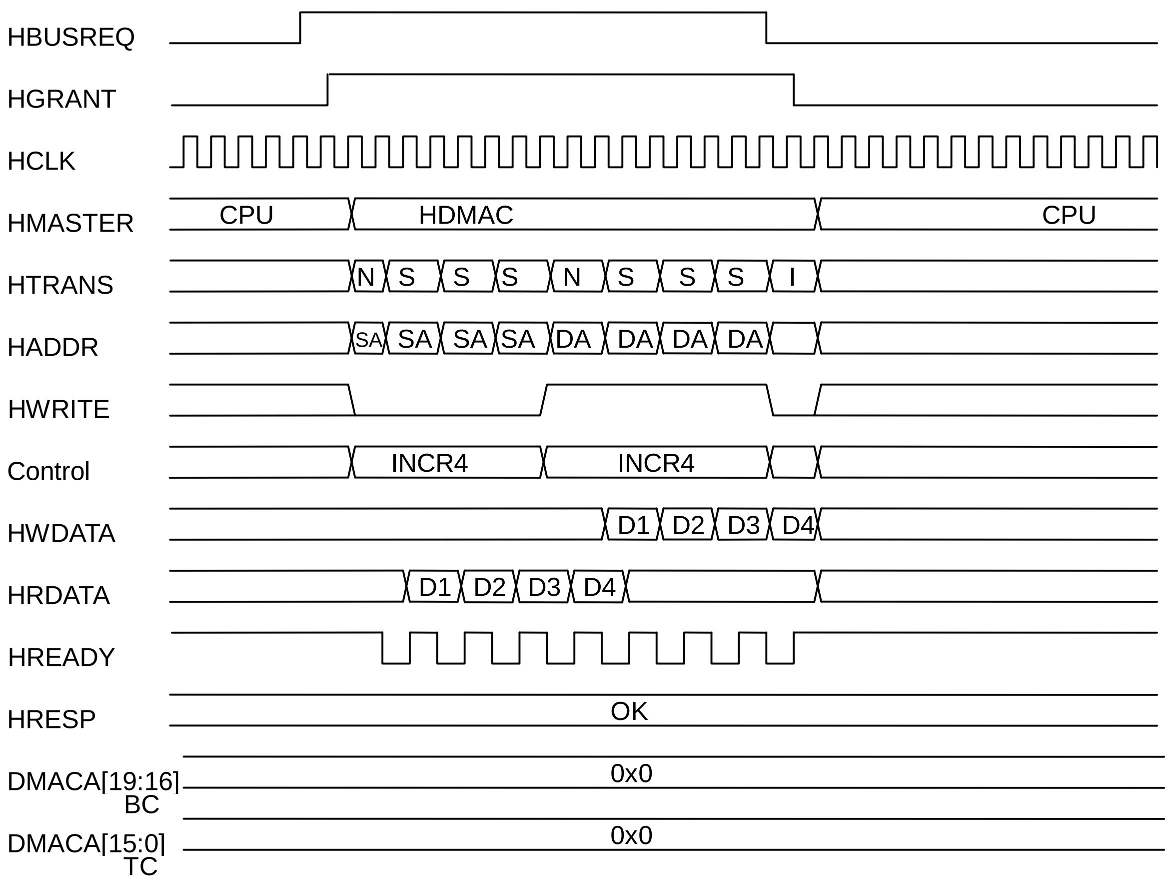


Figure 11-8 Increment/Lap beat transfer (example of INCR4 block transfer)

11.7.3. Channel priority control

DMAC controls priority of each channel by DMACR/PR bits.

11.7.3.1. Fixed priority

When priority is set to DMACR/PR bits, priority order is fixed and bus is given to the lowest figure of channel. Priority controller of DMAC switches channel when active channel is in transfer gap.
Thus, when all channels are active at the same time, the lowest figure of channel (ch0) is able to be selected by priority controller to start transfer. For instance, active channel (ch0) temporarily loses the bus at transfer gap. Then it is given to the second lowest figure of channel (ch1). If ch1 loses bus at transfer gap, it is given to ch0 again.
As a result, those 2 channels are able to preferentially acquire bus in the fixed priority mode.

Figure 11-9 shows defined channel in the fixed priority mode.

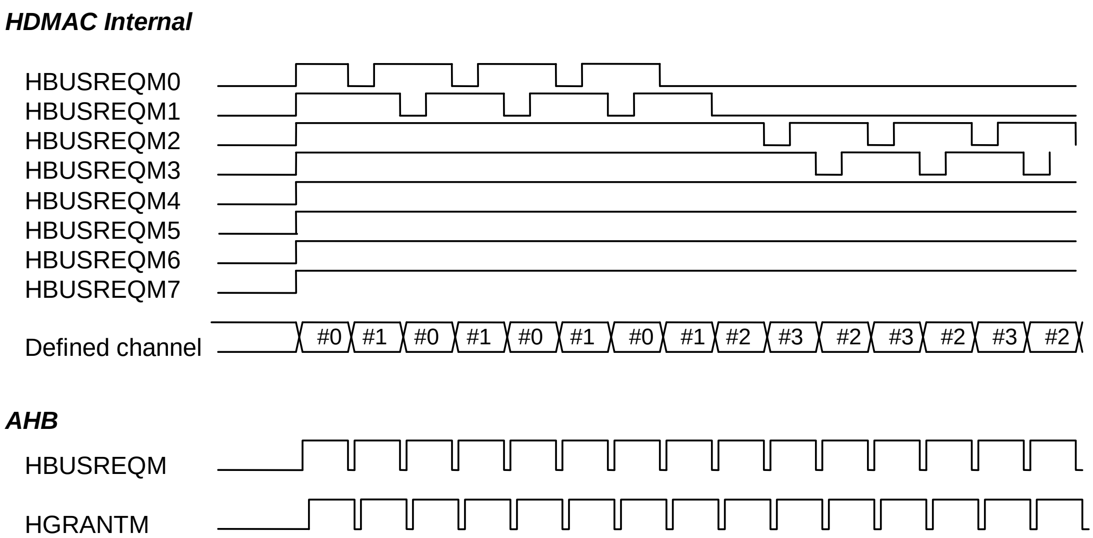


Figure 11-9 Defined channel in the fixed priority

11.7.3.2. Rotate priority

When priority is set to DMACR/PR bits, priority order rotates.
After bus is given to the lowest figure of channel, priority controller of DMAC switches channel at transfer gap of active channel.
Thus, when all channels become active at the same time, the lowest figure of channel (ch0) is selected by priority controller to enable transfer operation.
In the rotate priority mode, all channels are able to acquire bus in rotation. For instance, active channel (ch0) temporarily loses the bus at transfer gap. Then it is given to the second lowest figure of channel (ch1). If ch1 loses bus at transfer gap, it is given to the third lowest figure of channel (ch2.)

Figure 11-10 shows defined channel in the rotate priority mode.

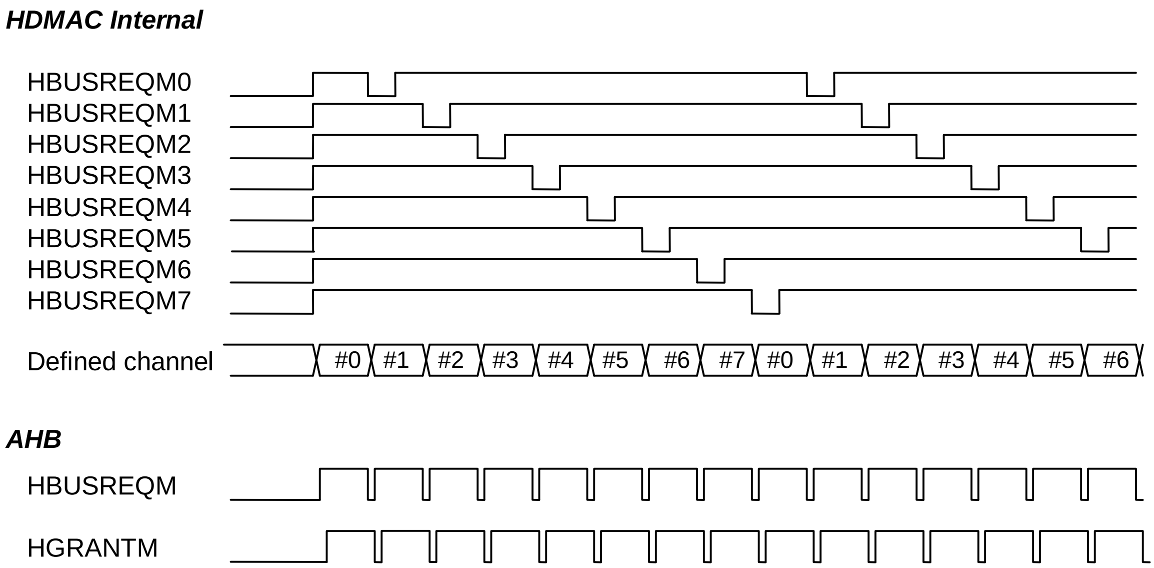


Figure 11-10 Defined channel in the rotate priority

11.7.4. Retry, split, and error

DMAC supports retry and split responses of AHB slave.

11.7.4.1. Retry and split

When DMAC receives retry or split responses from AHB slave during DMA transfer, DMAC negates bus temporarily to construct the contents to be retransmitted.
Figure 11-11 shows example of receiving retry response at INCR4 DMA transfer.

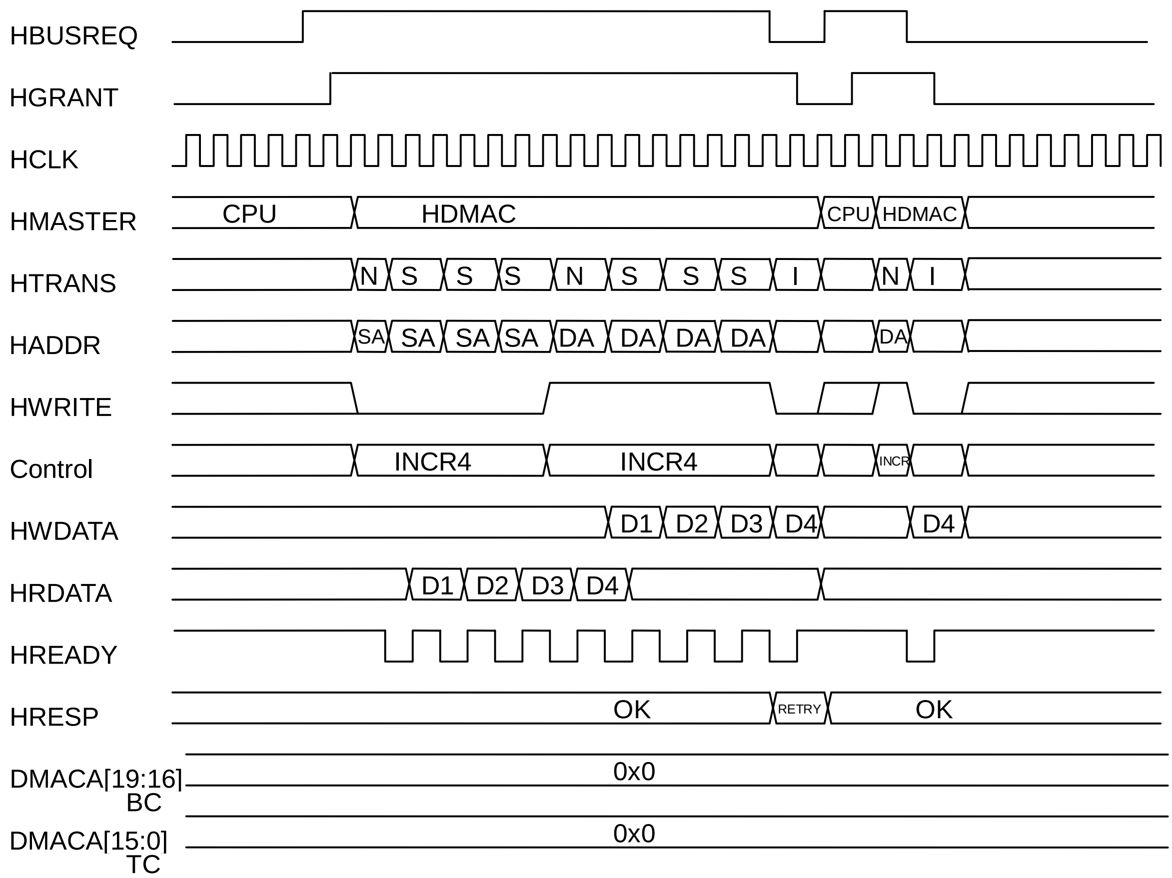


Figure 11-11 Increment/Lap beat transfer (example of INCR4 block transfer)

When DMAC negates bus temporarily, the channel received retry/split responses is continuously selected by DMAC's priority controller that transfer operation is able to start even though higher priority channel requests the bus

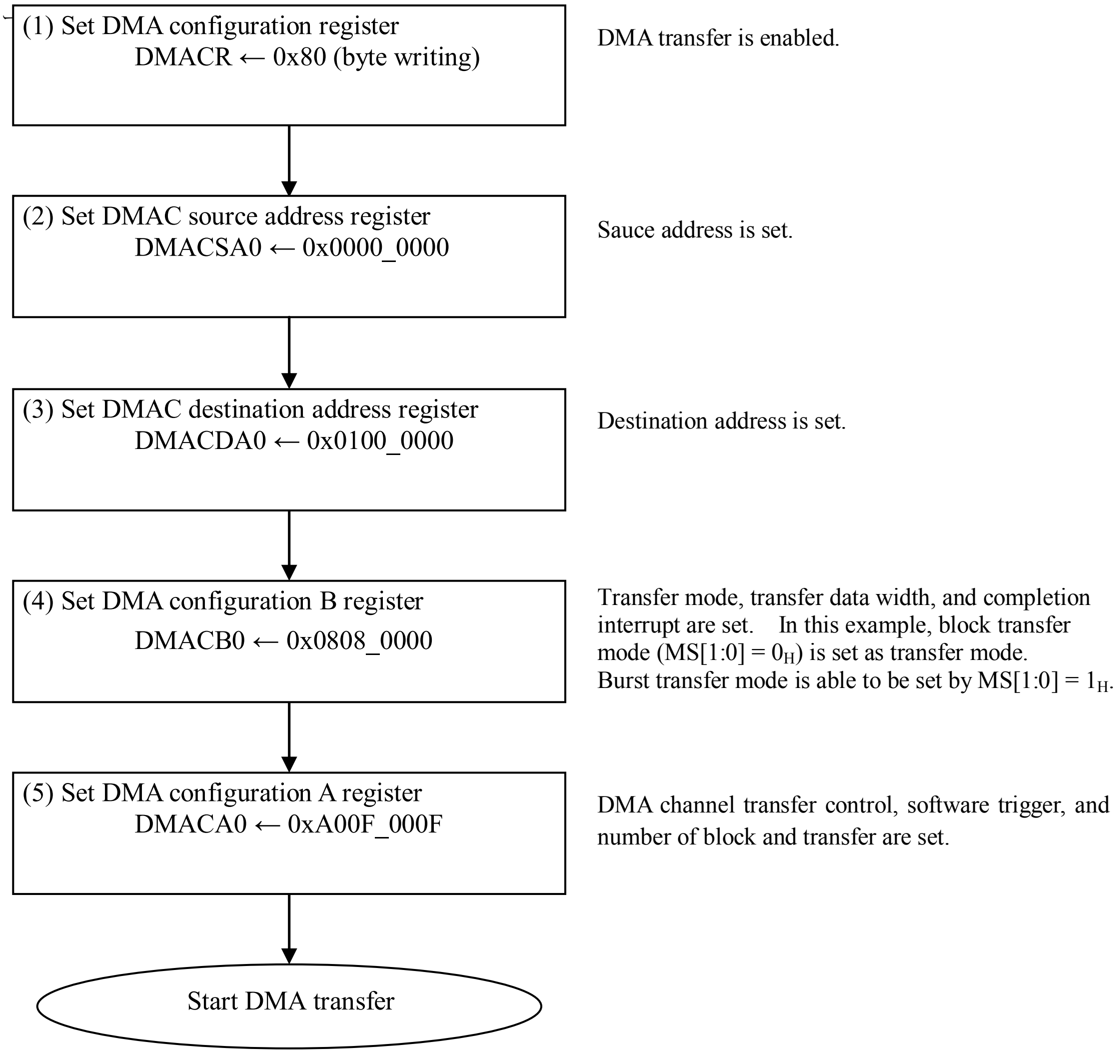
11.7.4.2. Error

When DMAC receives error reply from AHB slave during DMA transfer, DMAC negates bus request and immediately stops the transfer even though it is not completed.
In this case, neither Block/Transfer count register nor Source/Destination address register is updated.

11.8. Example of DMAC setting

11.8.1. DMA start in Single channel

Example of block and burst transfer by software request (with DMAC ch0)



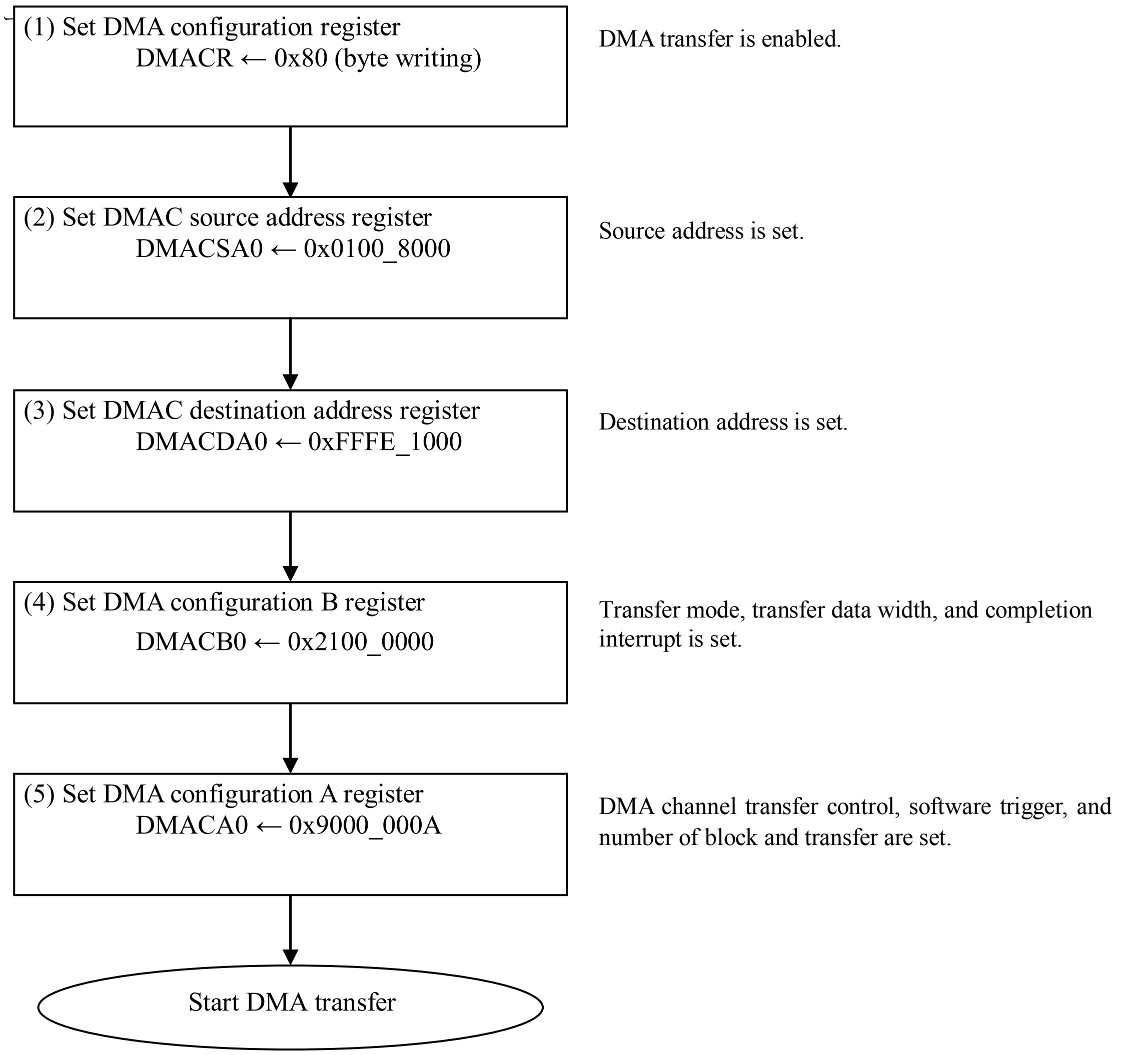
Remark: Setting order of step 1 ~ 4 is arbitrary; however, the one of step 5 is unable to be changed.

Figure 11-12 Example of block and burst transfer by software request (with DMAC ch0)

Note:

- DMA configuration register (DMACR) should be set by byte writing.
- For block and burst transfer with software request, DMA configuration A register (DMACA) should be set at the end.

Example of demand transfer by software request (with DMAC ch0)



Remark: Setting order of step 1 ~ 5 is arbitrary; however, the last setting should be step 1 or 5.

Figure 11-13 Example of demand transfer by software request (with DMAC ch0)

Note:

- DMA configuration register (DMACR) should be set by byte writing.

11.8.2. DMA start in all channels (in demand transfer mode)

All channels are able to start simultaneously by setting DMACR register after setting all DMA channels' register in the demand transfer mode. In this case, DMAC priority controller receives request of all channels at the same time, then transfer starts by selecting channel according to DMA channel priority, which is settable with PR bit of the DMACR.

12. Timer (TIMER)

This chapter describes function and spec of timer.

12.1. Outline

Timer is 2 channel timer module which is able to set 32/16 bit.

12.2. Feature

Timer has following features:

- 32/16 bit counter × 2 (bit width is controllable with register)
- Supplying 2 interrupt request signals to interrupt controller
- Timer clock prescaler unit
- 3 operation modes:
  - Free-run mode
  - Cycle timer mode
  - One-shot mode
- Using APB clock as base clock of the timer

12.3. Supply clock

APB clock is supplied to timer. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

12.4. Specification

Timer in MB86R03 uses ADKr2p0 (AMBA design kit) timer module of ARM Ltd.
Refer to Dual input timer of the AMBA Design Kit Technical Reference Manual for detail spec of the timer.

13. General-purpose input/output port (GPIO)

This chapter describes function and operation of general-purpose input/output port (GPIO.)

13.1. Outline

MB86R03 has max. 24 bit of GPIO port which is in common with other peripheral ports. Refer to "1.6.1 Pin Multiplex" for shared peripherals.
Direction control and data reading/writing of GPIO port is performed with using GPIO control register.

13.2. Feature

GPIO has following features:

- Supplied 24 bit GPIO port
- Composed of following 2 registers
    - Port data register (GPDR)
    - Data direction register (GPDDR)

13.3. Block diagram

Figure 13-1 shows block diagram of GPIO controller. In MB86R03, 24pcs. of these blocks are equipped.

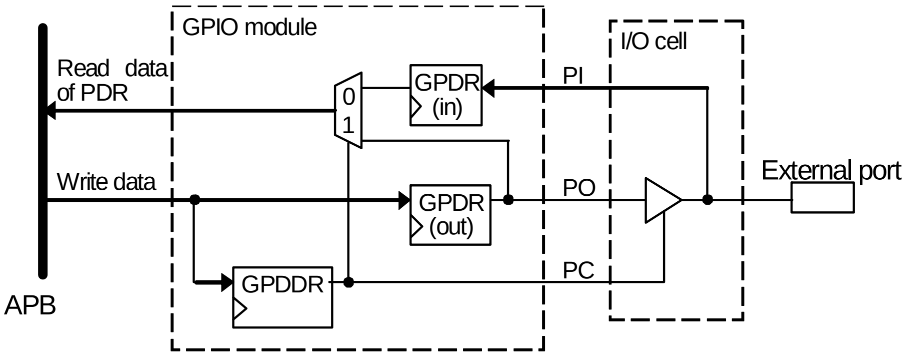


Figure 13-1 Block diagram of GPIO module

13.4. Supply clock

APB clock is supplied to GPIO. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

13.5. Register

This section describes detail of register in GPIO.

13.5.1. Register list

Table 13-1 shows list of GPIO register.

Table 13-1 GPIO register list

Address		Register	Abbreviation	Description
Base	Offset			
FFFE_9000$_H$	+ 00$_H$	Port data register 0	GPDR0	Setting of input/output data of GPIO_PD[7:0] pin
	+ 04$_H$	Port data register 1	GPDR1	Setting of input/output data of GPIO_PD[15:8] pin
	+ 08$_H$	Port data register 2	GPDR2	Setting of input/output data of GPIO_PD[23:16] pin
	+ 0C$_H$	(Reserved)	–	Reserved area (access prohibited)
	+ 10$_H$	Data direction register 0	GPDDR0	Control of input/output direction of GPIO_PD[7:0] pin
	+ 14$_H$	Data direction register 1	GPDDR1	Control of input/output direction of GPIO_PD[15:8] pin
	+ 18$_H$	Data direction register 2	GPDDR2	Control of input/output direction of GPIO_PD[23:16] pin
	+ 1C$_H$– + FFF$_H$	(Reserved)	–	Reserved area (access prohibited)

Description format of register

Following format is used for description of register’s each bit in "13.5.2 Port data register 0-2 (GPDR0-2)" to "13.5.3 Data direction register 0-2 (GPDDR2-0)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field’s value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

13.5.2. Port data register 0-2 (GPDR0-2)

GPDR0 - 2 registers are to set input/output data of GPIO port, and their corresponding GPIO pin is as follows.

- GPDR0: GPIO bit 7 - 0 (GPIO_PD[7:0] pin)
- GPDR1: GPIO bit 15 - 8 (GPIO_PD[15:8] pin)
- GPDR2: GPIO bit 23 - 16 (GPIO_PD[23:16] pin)

Input/Output directions of each GPIO are determined by the corresponding bit of GPDDR0 - 2 registers.

Address	**GPDR0: FFFE_9000$_H$ + 00$_H$** **GPDR1: FFFE_9000$_H$ + 04$_H$** **GPDR2: FFFE_9000$_H$ + 08$_H$**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								PDR0_7 PDR1_15 PDR2_23	PDR0_6 PDR1_14 PDR2_22	PDR0_5 PDR1_13 PDR2_21	PDR0_4 PDR1_12 PDR2_20	PDR0_3 PDR1_11 PDR2_19	PDR0_2 PDR1_10 PDR2_18	PDR0_1 PDR1_9 PDR2_17	PDR0_0 PDR1_8 PDR2_16
R/W	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit field		**Description**
No.	**Name**	
31-8	(Reserved)	Reserved bits. Write access is ignored. Read value of these bits is undefined.
7-0	PDR0_7-0	GPDR0 register's bit field. The register is setting register of GPIO_PD[7:0] pin's input/output data, and each bit corresponds to GPIO pin as follows. • PDR0_7: GPIO_PD[7] pin • PDR0_6: GPIO_PD[6] pin • PDR0_5: GPIO_PD[5] pin • PDR0_4: GPIO_PD[4] pin • PDR0_3: GPIO_PD[3] pin • PDR0_2: GPIO_PD[2] pin • PDR0_1: GPIO_PD[1] pin • PDR0_0: GPIO_PD[0] pin Input/Output directions of GPIO_PD[7] - GPIO_PD[0] pins are determined by the correspondence bit of the GPDDR0 register. Initial value of these bits is undefined.
	PDR1_15-8	GPDR1 register's bit field. This register is setting register of GPIO_PD[15:8] pin's input/output data, and each bit corresponds to GPIO pin as follows. • PDR1_15: GPIO_PD[15] pin • PDR1_14: GPIO_PD[14] pin • PDR1_13: GPIO_PD[13] pin • PDR1_12: GPIO_PD[12] pin • PDR1_11: GPIO_PD[11] pin • PDR1_10: GPIO_PD[10] pin • PDR1_09: GPIO_PD[9] pin • PDR1_08: GPIO_PD[8] pin Input/Output directions of GPIO_PD[15] - GPIO_PD[8] pins are determined by the corresponding bit of the GPDDR1 register. Initial value of these bits is undefined.

Bit field		Description
No.	Name	
7-0	PDR2_23-16	GPDR2 register's bit field. This register is setting register of GPIO_PD[23:16] pin's input/output data, and each bit corresponds to GPIO pin as follows. • PDR2_23: GPIO_PD[23] pin • PDR2_22: GPIO_PD[22] pin • PDR2_21: GPIO_PD[21] pin • PDR2_20: GPIO_PD[20] pin • PDR2_19: GPIO_PD[19] pin • PDR2_18: GPIO_PD[18] pin • PDR2_17: GPIO_PD[17] pin • PDR2_16: GPIO_PD[16] pin Input/Output directions of GPIO_PD[23] - GPIO_PD[16] pins are determined by the corresponding bit of the GPDDR2 register. Initial value of these bits is undefined.

13.5.3. Data direction register 0-2 (GPDDR2-0)

GPDDR0 - 2 registers are to control input/output directions of GPIO port, and their corresponding GPIO pin is as follows.

- GPDDR0: GPIO bit 7 - 0 (GPIO_PD[7:0] pin)
- GPDDR1: GPIO bit 15 - 8 (GPIO_PD[15:8] pin)
- GPDDR2: GPIO bit 23 - 16 (GPIO_PD[23:16] pin)

Address	GPDDR0: FFFE_9000$_H$ + 10$_H$ GPDDR1: FFFE_9000$_H$ + 14$_H$ GPDDR2: FFFE_9000$_H$ + 18$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								DDR0_7 DDR1_15 DDR2_23	DDR0_6 DDR1_14 DDR2_22	DDR0_5 DDR1_13 DDR2_21	DDR0_4 DDR1_12 DDR2_20	DDR0_3 DDR1_11 DDR2_19	DDR0_2 DDR1_10 DDR2_18	DDR0_1 DDR1_9 DDR2_17	DDR0_0 DDR1_8 DDR2_16
R/W	–	–	–	–	–	–	–	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit field		Description
No.	**Name**	
31-8	(Reserved)	Reserved bits. Write access is ignored. Read value of these bits is undefined.
7-0	DDR0_7-0	GPDR0 register's bit field. This register controls input/output directions of GPIO_PD[7:0] pin. 0: GPIO becomes input port 1: GPIO becomes output port GPIO pin corresponding to this register is as follows: • DDR0_7: GPIO_PD[7] pin • DDR0_6: GPIO_PD[6] pin • DDR0_5: GPIO_PD[5] pin • DDR0_4: GPIO_PD[4] pin • DDR0_3: GPIO_PD[3] pin • DDR0_2: GPIO_PD[2] pin • DDR0_1: GPIO_PD[1] pin • DDR0_0: GPIO_PD[0] pin These bits are initialized to "1" by reset.

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td rowspan="2">7-0</td><td>DDR1_15-8</td><td>GPDDR1 register's bit field.
This register controls input/output directions of GPIO_PD[15:8] pin.
<table><tr><td>0</td><td>GPIO becomes input port</td></tr><tr><td>1</td><td>GPIO becomes output port</td></tr></table>GPIO pin corresponding to this register is as follows:
• DDR1_15: GPIO_PD[15] pin
• DDR1_14: GPIO_PD[14] pin
• DDR1_13: GPIO_PD[13] pin
• DDR1_12: GPIO_PD[12] pin
• DDR1_11: GPIO_PD[11] pin
• DDR1_10: GPIO_PD[10] pin
• DDR1_9: GPIO_PD[9] pin
• DDR1_8: GPIO_PD[8] pin
These bits are initialized to "1" by reset.</td></tr>
<tr><td></td><td>GPDDR2 register's bit field.
This register controls input/output directions of GPIO_PD[23:16] pin
<table><tr><td>0</td><td>GPIO becomes input port</td></tr><tr><td>1</td><td>GPIO becomes output port</td></tr></table>GPIO pin corresponding to this register is as follows:
• DDR2_23: GPIO_PD[23] pin
• DDR2_22: GPIO_PD[22] pin
• DDR2_21: GPIO_PD[21] pin
• DDR2_20: GPIO_PD[20] pin
• DDR2_19: GPIO_PD[19] pin
• DDR2_18: GPIO_PD[18] pin
• DDR2_17: GPIO_PD[17] pin
• DDR2_16: GPIO_PD[16] pin
These bits are initialized to "1" by reset.</td></tr>
</table>

13.6. Operation

This section describes GPIO operation.

13.6.1. Direction control

Direction of GPIO port (bit 23 – 0) and its each bit is able to change by the GPDDRx register. Initial direction (DDRx bit's initial value of the GPDDRx register) after reset is "1" (output port.)

Note:

Notice for bus conflict at changing GPIO port direction.

13.6.2. Data transfer

When GPIO port is used as input port (DDRx = 0), the data signal input to the port input signal (PI) is stored to PDRx (in) at rising edge of APB clock (see Figure 13-1.) Reading GPDRx register enables to observe input data. During the period, write access to the GPDRx register is valid that PDRx (out) is changeable except at DDRx = 0.

When GPIO port is used as output port (DDR = 1), GPDRx register value is output to the port output signal (PO); in that time, read data of the register becomes the same value as the port output signal's.

14. PWM

This chapter describes operation and function of PWM (Pulse Width Modulator.)

14.1. Outline

MB86R03 has 2 channels of PWM which is able to output high-precision PWM wave pattern efficiently.

14.2. Feature

PWM has following features:

- Built-in 2 channels
- Individually setting of duty ratio, phase, and polarity
- Specifying one-shot output/continuous output of the pulse

14.3. Block diagram

Figure 14-1 shows block diagram of PWM.

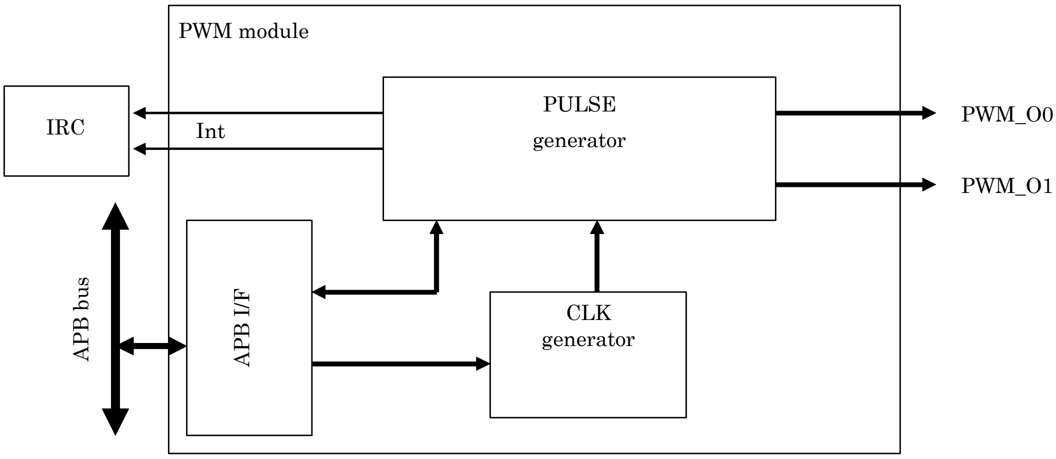


Figure 14-1 PWM block diagram

14.4. Related pin

PWM uses following pins.

Pin	Direction	Qty.	Description
PWM_O0 PWM_O1	OUT	2	PWM0/1 output

PWM pin is common with other peripheral I/O functions. To use the pin, its function should be set by either of followings to be selected to PWM side.

- Set to MPX_MODE_2[2:0] = "000_B" of multiplex mode setting register
- Set to MPX_MODE_4[1:0] = "01_B" of multiplex mode setting register
- Set to MPX_MODE_5[1] pin = "H" and MPX_MODE_5[0] pin = "L"

When these are set in multiples and PWM function is selected, the set pin makes PWM pin valid in parallel.

14.5. Supply clock

APB clock is supplied to PWM. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

14.6. Interrupt

When interrupt factor occurs, PWM notifies it to IRC. Refer to "7. Interrupt controller (IRC)" for more detail.

14.7. Register

This section describes PWM register.

14.7.1. Register list

This LSI equips 2 channels of PWM, and each of them has register shown in Table 14-1.

Table 14-1 PWM register list

Channel	Address		Register	Abbreviation	Description
	Base	Offset			
PWM ch0 (Output pin PWM_O0)	$FFF4\_1000_H$	$+ 00_H$	PWM ch0 base clock register	PWM0BCR	Setting base clock of PWM cycle
		$+ 04_H$	PWM ch0 pulse width register	PWM0TPR	Setting cycle length of 1 pulse
		$+ 08_H$	PWM ch0 phase register	PWM0PR	Setting phase cycle of the pulse
		$+ 0C_H$	PWM ch0 duty register	PWM0DR	Setting duty cycle of the pulse
		$+ 10_H$	PWM ch0 status register	PWM0CR	Setting PWM such as pulse output format and polarity
		$+ 14_H$	PWM ch0 start register	PWM0SR	Setting start/stop of PWM
		$+ 18_H$	PWM ch0 current count register	PWM0CCR	Indicating current count value in the BASECLK base
		$+ 1C_H$	PWM ch0 interrupt register	PWM0IR	Selecting cause of PWM interrupt factor
PWM ch1 (Output pin PWM_O1)	$FFF4\_1100_H$	$+ 00_H$	PWM ch1 base clock register	PWM1BCR	Setting base clock of PWM cycle
		$+ 04_H$	PWM ch1 pulse width register	PWM1TPR	Setting cycle length of 1 pulse
		$+ 08_H$	PWM ch1st place aspect register	PWM1PR	Setting phase cycle of the pulse
		$+ 0C_H$	PWM ch1 duty register	PWM1DR	Setting duty cycle of the pulse
		$+ 10_H$	PWM ch1 status register	PWM1CR	Setting PWM such as pulse output format and polarity
		$+ 14_H$	PWM ch1 start register	PWM1SR	Setting start/stop of PWM
		$+ 18_H$	PWM ch1 current count register	PWM1CCR	Indicating current count value in the BASECLK base
		$+ 1C_H$	PWM ch1 interrupt register	PWM1IR	Selecting cause of PWM interrupt factor

Note:

Access PWM ch0 and PWM ch1 areas with 32 bit (word.)

Description format of register

Following format is used for description of register’s each bit in "14.7.2 PWMx base clock register (PWMxBCR)" to "14.7.9 PWMx interrupt register (PWMxIR)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field’s value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

14.7.2. PWMx base clock register (PWMxBCR)

This register is to set base clock of PWM cycle.

Address	ch0 : FFF4_1000 + 00_H ch1 : FFF4_1100 + 00_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BCR[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-16	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
15-0	BCR	Base clock of the PWM cycle is set.

BCR[15:0]	Base clock
0	0 APBCLK (Setting prohibited)
1	1 APBCLK
|	|
65535	65535 APBCLK

14.7.3. PWMx pulse width register (PWMxTPR)

This register is to set cycle length of 1 pulse.

Address	ch0 : FFF4_1000 + 04_H ch1 : FFF4_1100 + 04_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TPR[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-16	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
15-0	TPR	Cycle length of 1 pulse shown in Figure 14-2 is set.

TPR[15:0]	Pulse cycle length
0	0 BASECLK (Setting prohibited)
1	1 BASECLK (Setting prohibited)
2	2 BASECLK
|	|
65535	65535 BASECLK

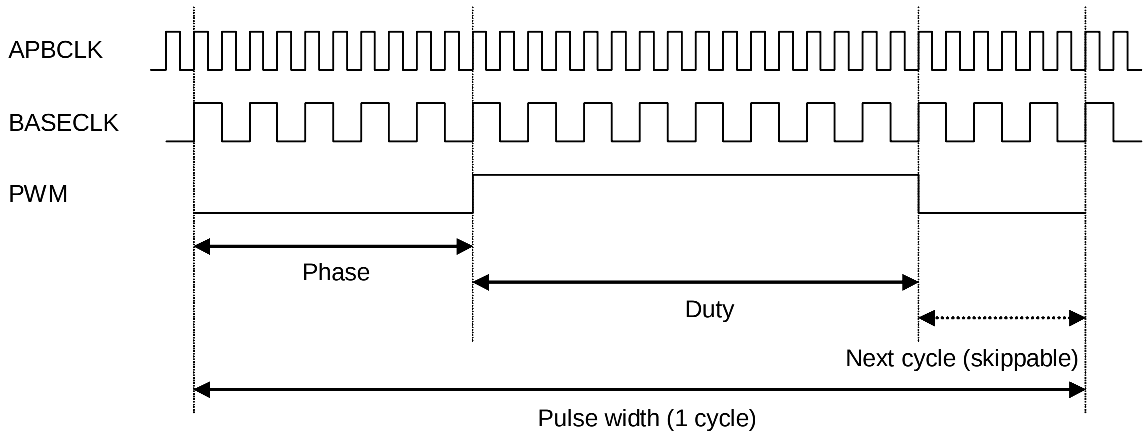


Figure 14-2 Setting parameter

14.7.4. PWMx phase register (PWMxPR)

This register is to set phase cycle of the pulse.

Address	ch0 : FFF4_1000 + 08_H ch1 : FFF4_1100 + 08_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PR[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-16	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
15-0	PR	Phase cycle shown in Figure 14-3 is set.

PR[15:0]	Phase cycle
0	0 BASECLK (Setting prohibited)
1	1 BASECLK
|	|
65535	65535 BASECLK

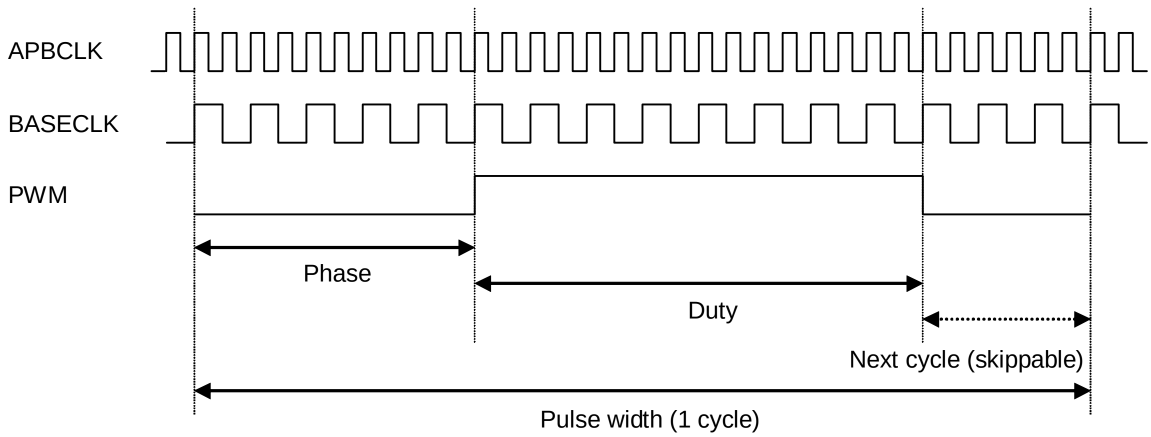


Figure 14-3 Setting parameter

14.7.5. PWMx duty register (PWMxDR)

This register is to set duty cycle of the pulse.

Address	ch0 : FFF4_1000 + $0C_H$ ch1 : FFF4_1100 + $0C_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DR[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-16	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
15-0	DR	Duty cycle shown in Figure 14-4 is set.

DR[15:0]	Duty cycle
0	0 BASECLK (Setting prohibited)
1	1 BASECLK
|	|
65535	65535 BASECLK

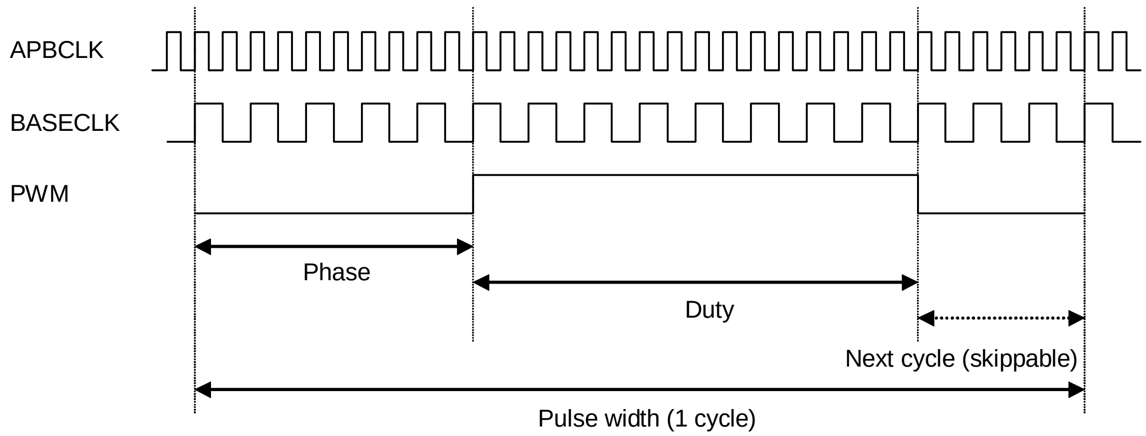


Figure 14-4 Setting parameter

14.7.6. PWMx status register (PWMxCR)

This register is to set PWM such as pulse output format and polarity.

Address	ch0 : FFF4_1000 + 10_H ch1 : FFF4_1100 + 10_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)												ONESHOT	(Reserved)		POL
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-4	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
3	ONESHOT	Pulse output format, either continuous output or one-shot output is set. 0: Continuous output (initial value) 1: One-shot output
2-1	(Reserved)	Reserved bits. Write "0" to these bits. Read value of these bits is undefined. Note: Writing "1" to these bits is prohibited.
0	POL	Polarity of the pulse is set. 0: Negative pulse (initial value) 1: Positive pulse

14.7.7. PWMx start register (PWMxSR)

This register is to set PWM start-up/stop.

Address	ch0 : FFF4_1000 + 14_H ch1 : FFF4_1100 + 14_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															START
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-1	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
0	START	Start-up/Stop of PWM are set. 0: Stop (initial value) 1: Start-up After pulse cycle ends, this bit is cleared to "0" when ONSHOT bit = 1 of PWMxCR register.

14.7.8. PWMx current count register (PWMxCCR)

This register is to indicate current count value in BASECLK base.

Address	ch0 : FFF4_1000 + 18_H ch1 : FFF4_1100 + 18_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCR[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31-16</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>15-0</td><td>CCR</td><td>Current count value in BASECLK base is indicated.
<table><tr><td>CCR[15:0]</td><td>Duty cycle</td></tr><tr><td>0</td><td>0 BASECLK</td></tr><tr><td>1</td><td>1 BASECLK</td></tr><tr><td>|</td><td>|</td></tr><tr><td>65535</td><td>65535 BASECLK</td></tr></table></td></tr>
</table>

14.7.9. PWMx interrupt register (PWMxIR)

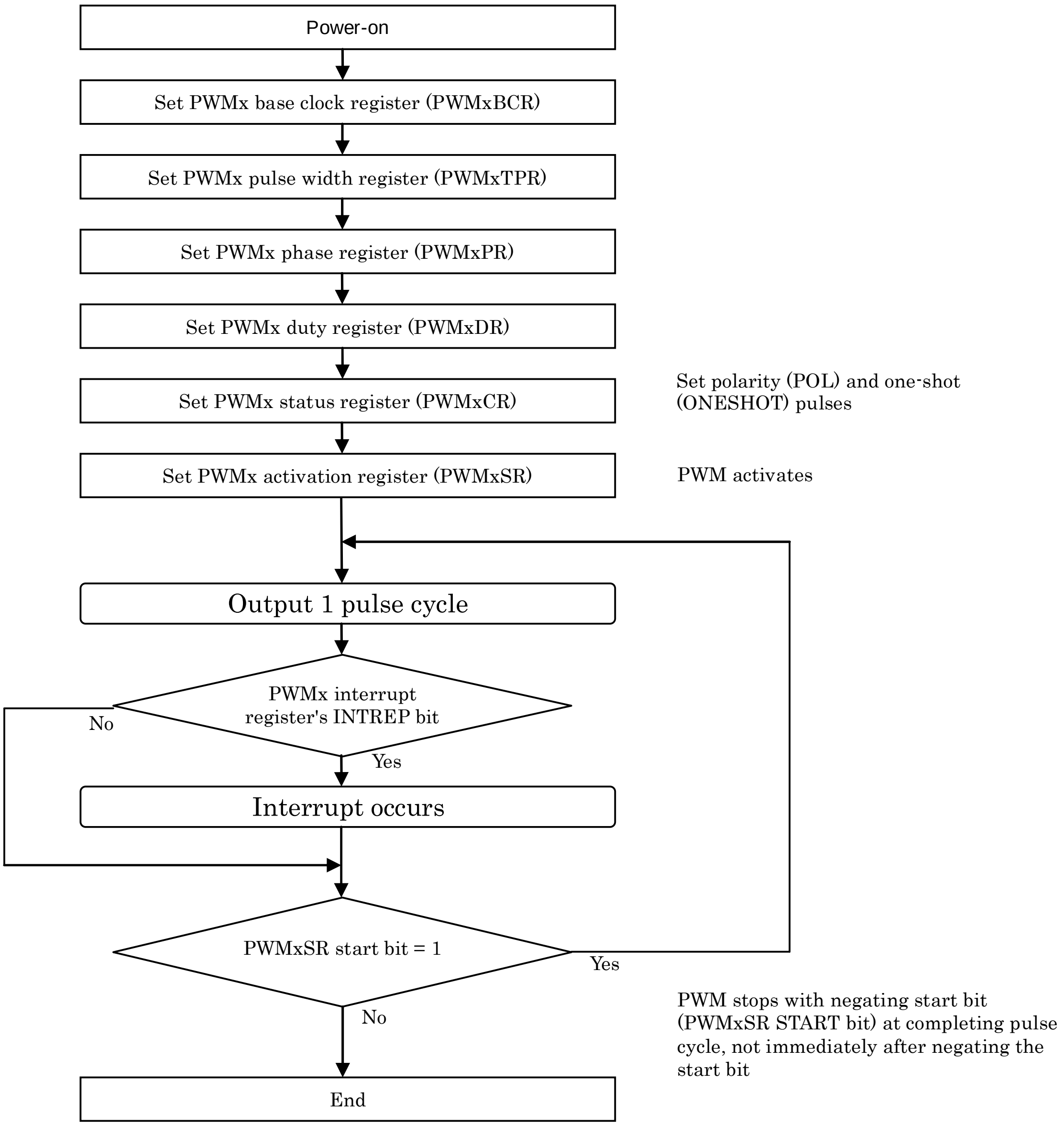
This register is to select cause of PWM interrupt.

Address	ch0 : FFF4_1000 + $1C_H$ ch1 : FFF4_1100 + $1C_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)						INTREP[1:0]		(Reserved)							DONE
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W1	R/W1
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31-10</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>9-8</td><td>INTREP[1:0]</td><td>The bit (DONE bit) which might be the cause of PWM interrupt is selected.
<table>
<tr><th>INTREP[1:0]</th><th>Possible cause bit for PWM interrupt</th></tr>
<tr><td>00</td><td>DONE bit is not selected</td></tr>
<tr><td>01</td><td>DONE bit is selected as cause of interrupt factor</td></tr>
<tr><td>10</td><td>(Setting prohibited)</td></tr>
<tr><td>11</td><td>(Setting prohibited)</td></tr>
</table></td></tr>
<tr><td>7-1</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>0</td><td>DONE</td><td>This bit indicates end of 1 pulse cycle.
<table>
<tr><td>0</td><td>1 pulse is not output (initial value)</td></tr>
<tr><td>1</td><td>1 pulse is output</td></tr>
</table>
This bit is cleared to "0" by writing "1".</td></tr>
</table>

14.8. Example of setting register

This section describes example of register's initial setting.



Set each register in the following condition:

- PWMx base clock register ≥ 1
- PWMx phase register ≥ 1
- PWMx duty register ≥ 1
- PWMx phase register + PWMx duty register ≤ PWMx pulse width register ≥ 2

  (The next phase setting after duty operation is omitted)

Figure 14-5 PWM register initial setting example

15. A/D converter

This chapter describes function and operation of A/D converter.

15.1. Outline

MB86R03 has 2 channels of A/D converter.

15.2. Feature

- Successive approximation A/D converter
- Max. conversion rate: Approx. 648K sample/sec, 10 bit resolution
- Immediate reading operation of A/D value by analog data auto. polling operation
- A/D converter operation clock dividing ratio can be selected
  - 1/4 (APB clock is 41.5MHz: Approx. 648.4K sample/sec)
  - 1/8 (APB clock is 41.5MHz: Approx. 324.1K sample/sec)
  - 1/16 (APB clock is 41.5MHz: Approx. 162.0K sample/sec)
  - 1/32 (APB clock is 41.5MHz: Approx. 81.0K sample/sec)
  - 1/64 (APB clock is 41.5MHz: Approx. 40.5K sample/sec)
  - 1/256 (APB clock is 41.5MHz: Approx. 10.1K sample/sec)
  - 1/1024 (APB clock is 41.5MHz: Approx. 2.5K sample/sec)
  - 1/4096 (APB clock is 41.5MHz: Approx. 0.6K sample/sec)

15.3. Block diagram

Figure 15-1 shows block diagram of A/D converter.

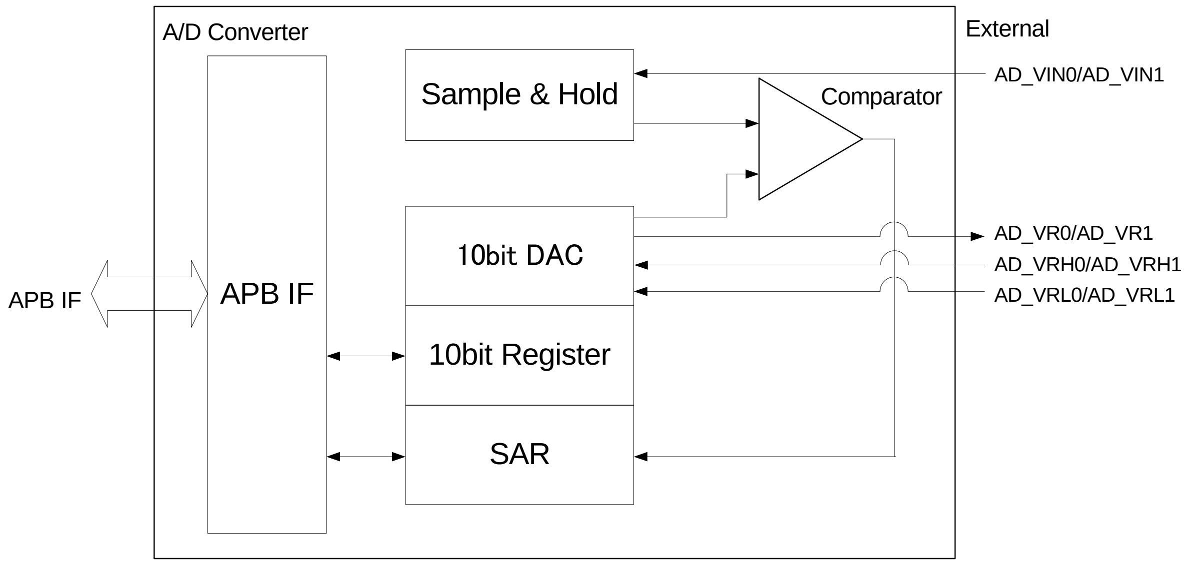


Figure 15-1 Block diagram of A/D converter

15.4. Related pin

A/D converter uses following pins.

Table 15-1 A/D converter related pin

Pin	Direction	Qty.	Description
AD_VIN0	IN	1	A/D analog input pin
AD_VIN1	IN	1	A/D analog input pin
AD_VRH0	IN	1	Reference voltage "H" input pin
AD_VRH1	IN	1	Reference voltage "H" input pin
AD_VRL0	IN	1	Reference voltage "L" input pin
AD_VRL1	IN	1	Reference voltage "L" input pin
AD_VR0	OUT	1	Reference output
AD_VR1	OUT	1	Reference output
AD_AVD0	IN	1	Analog power supply pin
AD_AVS1	IN	1	Analog GND

15.5. Supply clock

APB clock is supplied to A/D converter. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

15.6. Output truth value list

Example of truth value of A/D converter is shown below.

Table 15-2 A/D converter's truth value example list

Ideal input level	Output code									
VIN[V]	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
2.2485 ...	H	H	H	H	H	H	H	H	H	H
2.2471 ... 2.2485	H	H	H	H	H	H	H	H	H	L
2.2456 ... 2.2471	H	H	H	H	H	H	H	H	L	H
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.7515 ... 0.7529	L	L	L	L	L	L	L	L	L	H
... 0.7515	L	L	L	L	L	L	L	L	L	L

Note: AD_AVD0 = 3.0V, AD_VRH0/AD_VRH1 = 2.25V, AD_ VRL0/AD_VRL1 = 0.75V

15.7. Analog pin equivalent circuit

Figure 15-2 shows analog pin's equivalent circuit of A/D converter.

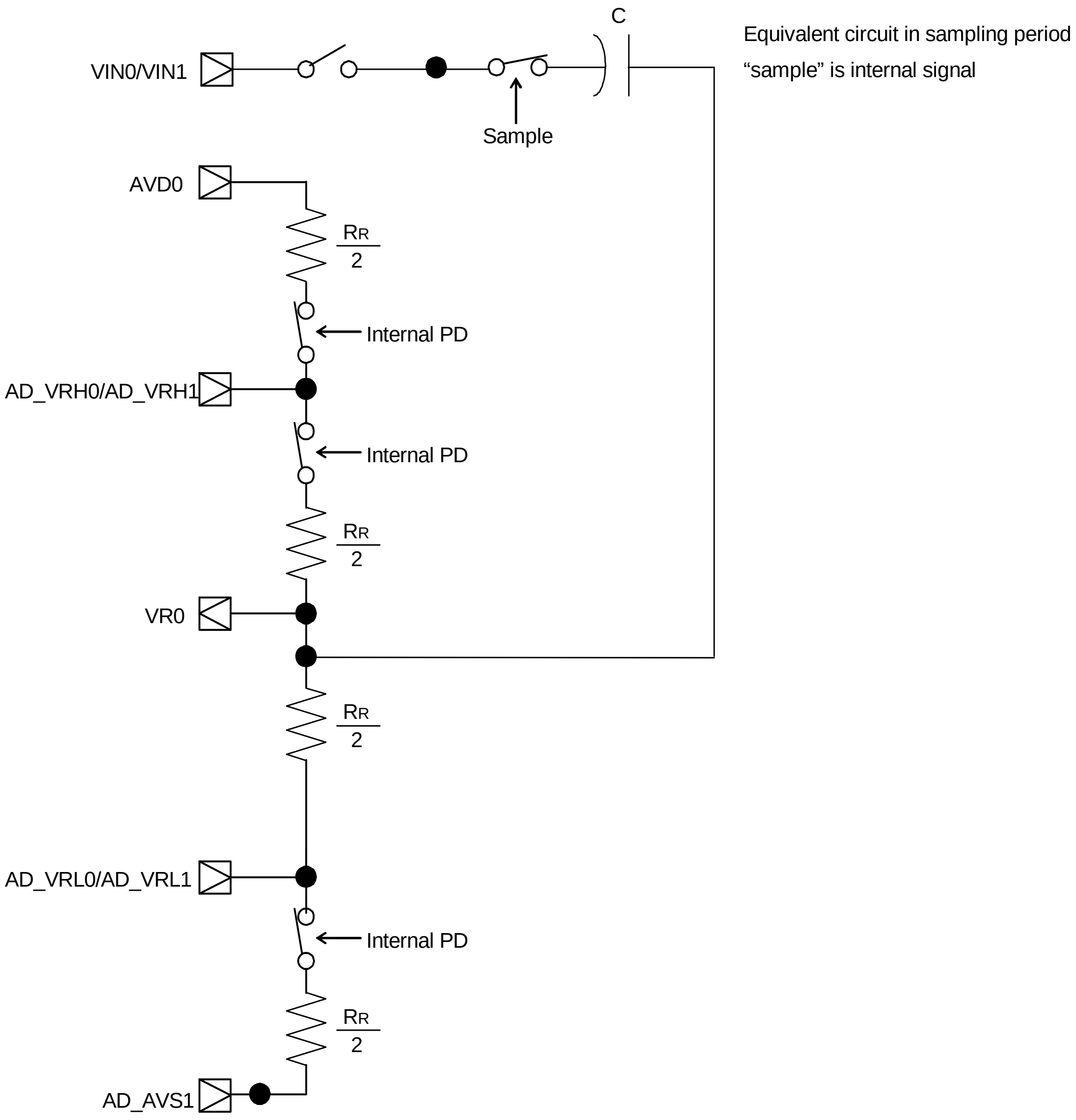


Figure 15-2 Analog pin's equivalent circuit

15.8. Register

This section describes A/D converter register.

15.8.1. Register list

This LSI has 2 channels of A/D converter unit, and each unit has the register shown in Table 15-3.

Table 15-3 ADC register list

Channel	Address		Register	Abbreviation	Description
	Base	Offset			
ADC ch0	$FFF5\_2000_H$	+ 00_H	ADC ch0 data register	ADC0DATA	A/D converted data is stored
		+ 04_H	(Reserved)	–	Reserved area, access prohibited
		+ 08_H	ADC ch0 power down control register	ADC0XPD	Power down mode is set/released
		+ $0C_H$	(Reserved)	–	Reserved area, access prohibited
		+ 10_H	ADC ch0 clock selection register	ADC0CKSEL	Clock frequency is supplied to A/D converter
		+ 14_H	ADC ch0 status register	ADC0STATUS	A/D converted data is stored to data register
ADC ch1	$FFF5\_3000_H$	+ 00_H	ADC ch1 data register	ADC1DATA	A/D converted data is stored
		+ 04_H	(Reserved)	–	Reserved area, access prohibited
		+ 08_H	Down of ADC ch1 power control register	ADC1XPD	Power down mode is set/released
		+ $0C_H$	(Reserved)	–	Reserved area, access prohibited
		+ 10_H	ADC ch1 clock selection register	ADC1CKSEL	Clock frequency is supplied to A/D converter
		+ 14_H	ADC ch1 status register	ADC1STATUS	A/D converted data is stored to data register

Note:

Access ADC ch0 and ADC ch1 areas with 32 bit (word.)

Description format of register

Following format is used for description of register's each bit in "15.8.2 ADCx data register (ADCxDATA)" to "15.8.4 ADCx clock selection register (ADCxCKSEL)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

15.8.2. ADCx data register (ADCxDATA)

This register is to store A/D converted data.

Address	ch0 : FFF5_2000 + 00_H ch1 : FFF5_3000 + 00_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)						DATA[9:0]									
R/W	R0	R0	R0	R0	R0	R0	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-10	(Reserved)	It is a reserved bit. Write access is ignored. Read value of these bits is always "0".
9-0	DATA[9:0]	Output data from A/D converter is stored with polling operation. When power down mode is set to release at ADCx power down control register (ADCxXPD), data is imported to this register.

15.8.3. ADCx power down control register (ADCxXPD)

This register is to control A/D converter operation.

Address	ch0 : FFF5_2000 + 08_H ch1 : FFF5_3000 + 08_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															XPD
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-1	(Reserved)	It is a reserved bit. Write access is ignored. Read value of these bits is always "0".
0	XPD	A/D converter operation is controlled. 0 : Power down mode (initial value) 1 : Release of power down mode When "1" is written to XPD bit, A/D converter's power-down mode is released and A/D data polling starts. Writing "0" to the bit sets A/D converter's power-down mode and A/D data polling stops.

15.8.4. ADCx clock selection register (ADCxCKSEL)

This register is to se to specify ADC clock frequency supplying to A/D converter.
This setting enables sampling plate change.

Address	ch0 : FFF5_2000 + 10_H ch1 : FFF5_3000 + 10_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)													CKSEL[2:0]		
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-3	(Reserved)	It is a reserved bit. Write access is ignored. Read value of these bits is always "0".
2-0	CKSEL[2:0]	Specify clock frequency supplying to A/D converter. (see table below) This clock is made dividing APB clock (41.5MHz.) Analog voltage sampling is carried out every 16 cycles of clock set in this register.

CKSEL[2:0]	Clock frequency setting	Sampling late [sample/sec.]
000_B	1/4096	0.6K
001_B	1/1024	2.5K
010_B	1/256	10.1K
011_B	1/64	40.5K
100_B	1/32	81.0K
101_B	1/16	162.0K
110_B	1/8	324.1K
111_B	1/4	648.4K

15.8.5. ADCx status register (ADCxSTATUS)

This register is to indicate whether A/D data conversion is completed.

Address	ch0 : FFF5_2000 + 14_H ch1 : FFF5_3000 + 14_H															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															CMP
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R/W0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-1	(Reserved)	It is a reserved bit. Write access is ignored. Read value of these bits is always "0".
0	CMP	Whether A/D data conversion is completed is indicated. 0: A/D data conversion is not completed (initial value) 1: A/D data conversion is completed At the time data is set to ADCxDATA, CMP bit becomes "1". Writing "0" to the bit clears register value (although "1" is written to CMP bit, register bit value does not change.) Setting "1" to CMP bit outputs interrupt.

15.9. Basic operation flow

Basic operation flow of ADC is shown below.

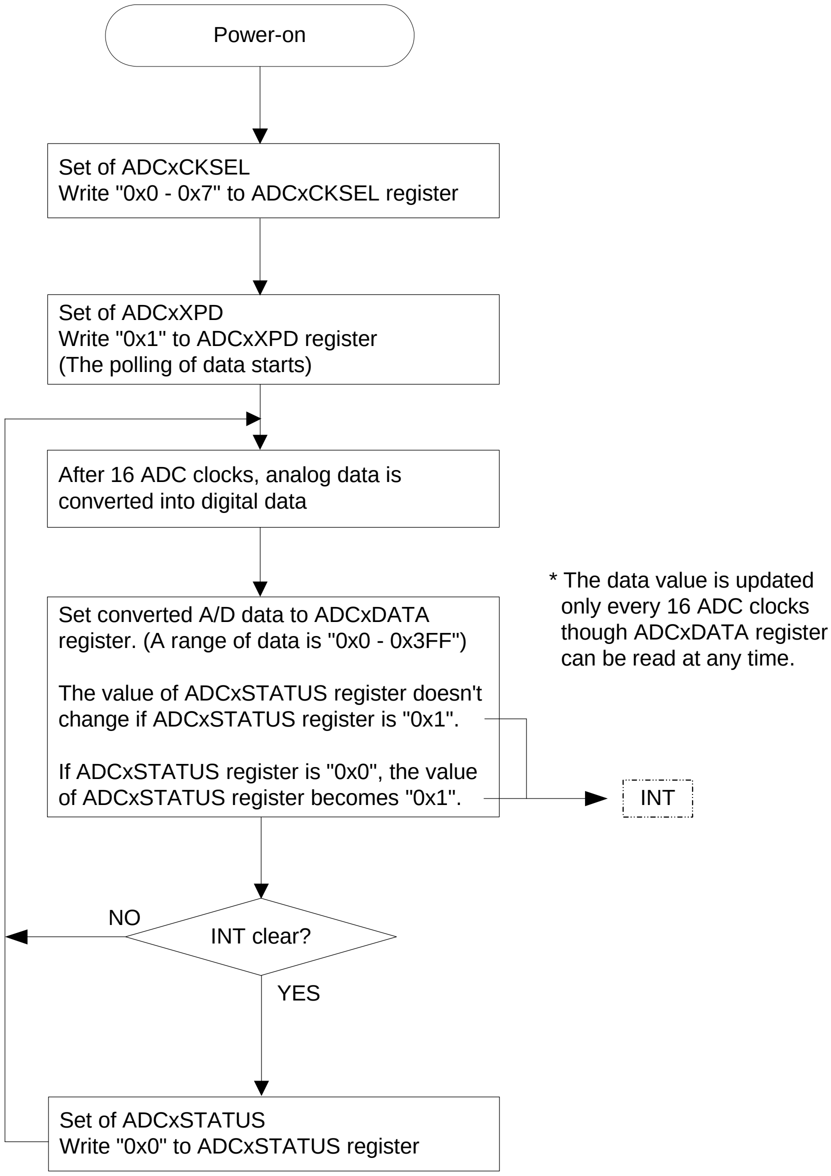


Figure 15-3 AD converter's basic operation flow

16. Graphics display controller (GDC)

Refer to another document, "MB86R03 'Jade-L' LSI product specifications graphics display controller (GDC)" for the controller spec.

17. Serial audio interface (I2S)

This chapter describes function and operation of serial audio interface (hereafter called, I2S.)

17.1. Outline

MB86R03 equips audio I/O interface in I2S format, and up to 3 channels are able to be used.

Note:
I2S is Inter-IC Sound bus advocated by Philips Semiconductors (now NXP).

17.2. Feature

I2S interface in MB86R03 has following features:

- Selecting master/slave operations by programmable
- Supporting state of transmission only, reception only, and simultaneous transmission/reception
- Selecting 1 sub frame and 2 sub frame constructions
- Setting up to 32 channels to each sub frame
- Individually setting number of channel in each sub frame
- Individually setting channel length of each sub frame (number channel bit)
- Individually setting word length in channel of each sub frame (corresponding to MSB-Justified)
- Setting valid/invalid of each channel in each sub frame **(Note 1)**
- Setting word length from 7 to 32 bit
- Programming frequency of frame synchronous signal
- Setting up to 3071 bit in 1 frame
- Programming width of frame synchronous signal (1 bit or 1 channel length)
- Programming phase of frame synchronous signal (0 bits or 1 bit delay)
- Setting polarity of frame synchronous signal
- Setting polarity of serial bit clock
- Programming sampling point of received data
- Selecting clock frequency dividing source of serial bit clock in the master mode (internal and external clock.)
- Setting clock frequency dividing ratio in the master mode
  Frequency of I2S_SCLK = frequency of AHB clock (or external clock)/2 × CKRT[5:0]
  Frequency dividing ratio is settable within 0 – 126 in multiple of 2 (when the ratio is 0, frequency dividing source is by-passed)
- Data transfer to system memory by DMA, interrupt, and polling

Note 1:
Data is not sent or received to invalid channel

17.3. Block diagram

Figure 17-1 shows block diagram of I2S. As shown below, MB86R03 has 3 channels of I2S module.

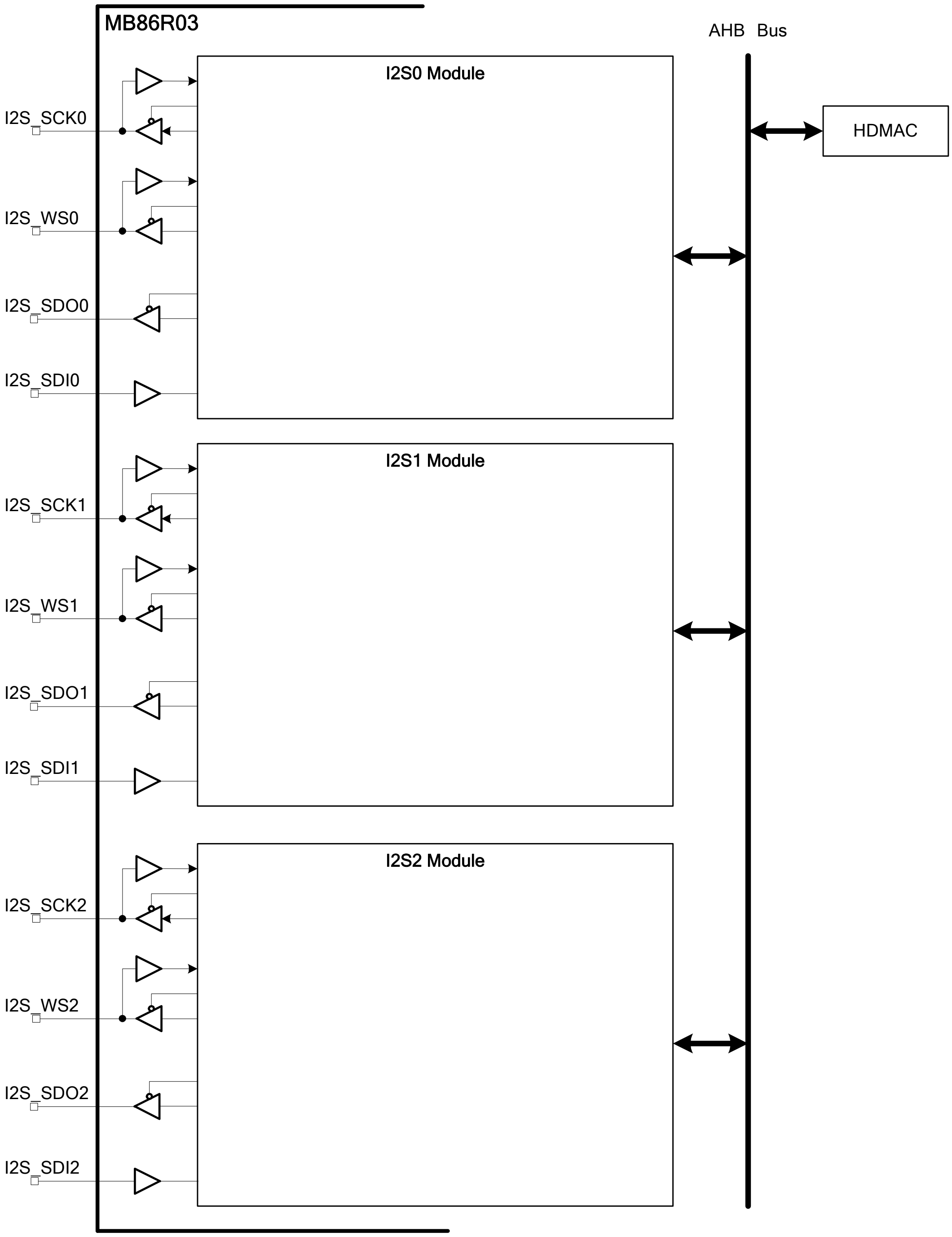


Figure 17-1 Block diagram of I2S

17.4. Related pin

I2S interface uses following pins which are common with other functions. To use this pin, its function should be set to be selected on I2S side to external pin, MPX_MODE_1[1:0] or PIN MPX Select register on CCNT module.

- I2S ch0: Set to MPX_MODE_1[1:0] pin = "HL"
- I2S ch1: Set to MPX_MODE_2[2:0] = "010"/"011"/"100", or MPX_MODE_4[1:0] = "01" of multiplex mode setting register
- I2S ch2: Set to MPX_MODE_2[2:0] = "000"/"010"/"100" of multiplex mode setting register
  For the case of "100" setting, only the pin with input function among I2S ch2 related pins become valid

Table 17-1 I2S related pin

Pin	Direction	Qty.	Description
I2S_ECLK0 I2S_ECLK1 I2S_ECLK2	I	3	External clock input
I2S_SCK0 I2S_SCK1 I2S_SCK2	IO	3	Bit clock input/output signal In the master mode: Clock output In the slave mode: Clock input
I2S_WS0 I2S_WS1 I2S_WS2	IO	3	Input/Output signals of frame synchronization Polarity is settable in the register In the master mode: Frame synch. signal output In the slave mode: Frame synch. signal input
I2S_SDI0 I2S_SDI1 I2S_SDI2	I	3	Serial reception data input signal
I2S_SDO0 I2S_SDO1 I2S_SDO2	O	3	Serial transmission data output signal

17.5. Supply clock

AHB clock is supplied to I2S interface unit. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

17.6. Register

This section describes I2S register.

17.6.1. Register list

Register relating to I2S control is shown below.

Table 17-2 I2S register list

Module	Address	Register	Function
I2S ch0	FFEE_0000	I2S0RXFDAT	Reception FIFO data register
	FFEE_0004	I2S0TXFDAT	Transmission FIFO data register
	FFEE_0008	I2S0CNTREG	Control register
	FFEE_000C	I2S0MCR0REG	Channel control register 0
	FFEE_0010	I2S0MCR1REG	Channel control register 1
	FFEE_0014	I2S0MCR2REG	Channel control register 2
	FFEE_0018	I2S0OPRREG	Operation control register
	FFEE_001C	I2S0SRST	Software reset register
	FFEE_0020	I2S0INTCNT	Interrupt control register
	FFEE_0024	I2S0STATUS	STATUS register
	FFEE_0028	I2S0DMAACT	DMA start-up register
I2S ch1	FFEF_0000	I2S1RXFDAT	Reception FIFO data register
	FFEF_0004	I2S1TXFDAT	Transmission FIFO data register
	FFEF_0008	I2S1CNTREG	Control register
	FFEF_000C	I2S1MCR0REG	Channel control register 0
	FFEF_0010	I2S1MCR1REG	Channel control register 1
	FFEF_0014	I2S1MCR2REG	Channel control register 2
	FFEF_0018	I2S1OPRREG	Operation control register
	FFEF_001C	I2S1SRST	Software reset register
	FFEF_0020	I2S1INTCNT	Interrupt control register
	FFEF_0024	I2S1STATUS	STATUS register
	FFEF_0028	I2S1DMAACT	DMA start-up register
I2S ch2	FFF0_0000	I2S2RXFDAT	Reception FIFO data register
	FFF0_0004	I2S2TXFDAT	Transmission FIFO data register
	FFF0_0008	I2S2CNTREG	Control register
	FFF0_000C	I2S2MCR0REG	Channel control register 0
	FFF0_0010	I2S2MCR1REG	Channel control register 1
	FFF0_0014	I2S2MCR2REG	Channel control register 2
	FFF0_0018	I2S2OPRREG	Operation control register
	FFF0_001C	I2S2SRST	Software reset register
	FFF0_0020	I2S2INTCNT	Interrupt control register
	FFF0_0024	I2S2STATUS	STATUS register
	FFF0_0028	I2S2DMAACT	DMA start-up register

All registers of I2S correspond to access in byte (8 bit), half word (16 bit), and word (32 bit.)

Description format of register

Following format is used for description of register's each bit in "17.6.2 I2SxRXFDAT register" to "17.6.12 I2SxDMAACT register".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

17.6.2. I2SxRXFDAT register

This register is reception FIFO register that is able to maintain up to 18 words (simultaneous transfer mode) or 36 words (reception only mode.)

Address	ch0 : FFEE_0000 (h) ch1 : FFEF_0000 (h) ch2 : FFF0_0000 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RXDATA															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXDATA															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-0	RXDATA[31:0]	The word received from serial bus is written to reception FIFO. When frame is 1 sub frame construction and word length set to S0WDLN of MCR0REG register is 32 bit or less (16 bit when RHLL of CNTREG register is "1"), it is written to reception FIFO after higher order bit is extended. When frame is 2 sub frame construction and word length set to S0WDLN of MCR0REG register is 32 bit or less (16 bit when RHLL of CNTREG register is "1"), reception data of sub frame 0 is written to reception FIFO after higher order bit is extended. For the case that word length set to S1WDL of MCR0REG register is 32 bit or less, reception data of sub frame 1 is written to reception FIFO after higher order bit is extended. When BEXT of CNTREG register is "1", it is extended with MSB of reception word (sign extension). For the case that the value is "0", it is enhanced by "0". Top of the data (First In) of reception FIFO is able to be read by read access, and then the next reception FIFO data is automatically updated. It is able to be accessed regardless of shift register's operation status. When RXNUM of STATUS register is "0", invalid data is able to be read. Writing to RXDATA is ignored.

17.6.3. I2SxTXFDAT register

This register is transmission FIFO register that is able to maintain up to 18 words (simultaneous transfer mode) or 36 words (transmission only mode.)

Address	ch0：FFEE_0004 (h) ch1：FFEF_0004 (h) ch2：FFF0_0004 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TXDATA															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TXDATA															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-0	TXDATA[31:0]	Word to be transmitted is able to be written as long as transmission FIFO is not full. Write access is able to be performed regardless of shift register's operation status. The word written to full transmission FIFO is actually not written. Although writing data is accessed in word, half-word, and byte access, actual number of bit to be transmitted is determined by S0WDL and S1WDL (when frame is 2 sub frame) of MCR0REG register. The data read from TXDATA is invalid one (the data after right justified last written data.)

17.6.4. I2SxCNTREG register

Address	ch0 : FFEE_0008 (h) ch1 : FFEF_0008 (h) ch2 : FFF0_0008 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CKRT						OVHD									
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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	MSKB	MSMD	SBFN	RHLL	ECKM	BEXT	FRUN	MLSB	TXDIS	RXDIS	SMPL	CPOL	FSPH	FSLN	FSPL
R/W	R	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
Initial	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31-26</td><td>CKRT[5:0]</td><td>This sets output clock frequency dividing ratio at master operation.
AHB clock is divided at ECKM = 0, and external clock is divided at ECKM = 1. Only even number of the ratio is supported and output clock's DUTY becomes 50%. CKRT [5:0] × 2 becomes number of AHB clock or external clock cycle included in 1 cycle (I2S_SCKx.)

Setting examples are shown below.

External clock mode and external clock are 24.576MHz:
<table>
<tr><th>CKRT</th><th>Dividing ratio</th><th>I2S_SCKx</th></tr>
<tr><td>0x00</td><td>By pass</td><td>24.576MHz
(external clock is output as it is)</td></tr>
<tr><td>0x01</td><td>1/2</td><td>12.288MHz</td></tr>
<tr><td>0x02</td><td>1/4</td><td>6.144MHz</td></tr>
<tr><td>0x03</td><td>1/6</td><td>4.096MHz</td></tr>
<tr><td>0x04</td><td>1/8</td><td>3.072MHz</td></tr>
<tr><td>0x05</td><td>1/10</td><td>2.458MHz</td></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
</table>

Internal clock mode and AHB clock are 80MHz:
<table>
<tr><th>CKRT</th><th>Dividing ratio</th><th>I2S_SCKx</th></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
<tr><td>0x04</td><td>1/8</td><td>10MHz</td></tr>
<tr><td>0x05</td><td>1/10</td><td>8MHz</td></tr>
<tr><td>0x06</td><td>1/12</td><td>6.67MHz</td></tr>
<tr><td>0x07</td><td>1/14</td><td>5,71MHz</td></tr>
<tr><td>0x08</td><td>1/16</td><td>5MHz</td></tr>
<tr><td>0x09</td><td>1/18</td><td>4.44MHz</td></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
</table></td></tr>
<tr><td>25-16</td><td>OVHD[9:0]</td><td>Frame rate is able to be adjusted by inserting OVHD bit following to valid data of the frame. OVHD section of the transmission frame becomes in high impedance. Up to 0 – 1023 OVHD bit is able to be inserted, and is inserted at the end of the frame.
The value set to OVHD becomes the number of insertion bit.
The following expressions are formed for OVHD and frame synchronous signal cycle (2nd.)

1 sub frame construction:
OVHD = Frame synchronous signal cycle/I2S_SCKx cycle – (S0CHL + 1) × (S0CHN + 1)
2 sub frame construction:
OVHD = Frame synchronous signal cycle/I2S_SCKx cycle – (S0CHL + 1) × (S0CHN + 1) – (S1CHL + 1) × (S1CHN + 1)</td></tr>
<tr><td>15</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>14</td><td>MSKB</td><td>Serial output data of invalid transmission frame is set.
For master operation (MSMD = 1), free-running mode (FRUN = 0), and TXENB = 1:
When transmission FIFO is empty at frame synchronous signal output, MSKB is output to all valid channels of its transmission frame.
For slave operation (MSMD = 0) and TXENB = 1:
When transmission FIFO is empty at frame synchronous signal reception, MSKB is output to all valid channels of its transmission frame.
For the case that transmission word length is shorter than the channel length, MSKB is driven to the rest of bit in transmission channel (channel length -word length.)</td></tr>
<tr><td>13</td><td>MSMD</td><td>Master and slave modes are set.
<table><tr><td>0</td><td>Slave operation</td></tr><tr><td>1</td><td>Master operation</td></tr></table></td></tr>
<tr><td>12</td><td>SBFN</td><td>Sub frame construction (number of sub frame) of the frame is specified.
<table><tr><td>0</td><td>1 sub frame construction (only sub frame 0)</td></tr><tr><td>1</td><td>2 sub frame construction (sub frame 0 and sub frame 1)
Frame starts from the 0th sub frame</td></tr></table></td></tr>
<tr><td>11</td><td>RHLL</td><td>Whether word construction of FIFO is 1 or 2 words is set.
It is considered to be used at protocol, such as I2S and MSB-Justified.
<table><tr><td>0</td><td>32 bit FIFO word is handled as 1 word</td></tr><tr><td>1</td><td>32 bit FIFO word is handled as 2 words at serial bus with dividing 16 bit each to low order and high order. They are transferred by serial bus in order of low order, high order, low order, and high order.
At reception, 2 consecutive words from serial bus is handled as low order and high order, and they are put in 1 word (32 bit) to write to reception FIFO.</td></tr></table></td></tr>
<tr><td>10</td><td>ECKM</td><td>Clock frequency dividing is selected in the master mode.
<table><tr><td>0</td><td>Internal clock (AHB clock) is divided and output</td></tr><tr><td>1</td><td>External clock (2S_ECLKx pin input) is divided and output</td></tr></table></td></tr>
<tr><td>9</td><td>BEXT</td><td>When reception word length is shorter than the word length of FIFO (32 bit when RHLL is "0", and 16 bit when RHLL is "1"), extension mode of upper bit (word length of FIFO - reception word length) should be set.
<table><tr><td>0</td><td>Extended by 0</td></tr><tr><td>1</td><td>Extended by sign bit (for MSB of word is "1", extended by "1" and its "0" is extended by "0")</td></tr></table></td></tr>
<tr><td>8</td><td>FRUN</td><td>Output mode of frame synchronous signal is set.
<table><tr><td>0</td><td>Burst mode
When start bit of OPRREG register is "1", frame synchronous signal is output according to TXENB, RXENB, and transmission/reception FIFO conditions</td></tr><tr><td>1</td><td>Free-running mode
When start bit of OPRREG register is "1", frame synchronous signal proceeds free-running with the set frame rate
When start bit is "0", frame synchronous signal is not output.</td></tr></table></td></tr>
</table>

Bit field		Description
No.	Name	
7	MLSB	Word bit's shift order is set. 0 Shift starts from MSB of the word 1 Shift starts from LSB of the word
6	TXDIS	Transmitting function is enabled or disabled. 0 Transmitting function is enabled 1 Transmitting function is disabled
5	RXDIS	Receiving function is enabled or disabled. 0 Receiving function is enabled 1 Receiving function is disabled
4	SMPL	Sampling point of the data is specified. 0 Sampling at the center of reception data 1 Sampling at the end of reception data
3	CPOL	I2S_SCKx polarity which drives/samples serial data is specified. 0 Data is driven at rising edge of I2S_SCKx, and sampled at falling edge 1 Data is driven at falling edge of I2S_SCKx, and sampled at rising edge
2	FSPH	Phase is specified to I2S_WSx frame data. 0 I2S_WSx becomes valid 1 clock before the first bit of frame data 1 I2S_WSx becomes valid at the same time as the first bit of frame data
1	FSLN	Pulse width of I2S_WSx is specified. 0 Pulse width is 1 cycle/I2S_SCKx long (1 bit) 1 Pulse width is 1 channel long (1 channel) Setting "1" is prohibited when frame length is 1 channel long.
0	FSPL	Polarity of I2S_WSx pin is set. 0 Frame synchronous signal becomes valid with I2S_WSx is "1" The value is "0" at idle 1 Frame synchronous signal becomes valid with I2S_WSx is "0" The value is "1" at idle

Note:

Do not overwrite CNTREG register when start bit of OPRREG register is "1".

17.6.5. I2SxMCR0REG register

Address	ch0 : FFEE_000C (h) ch1 : FFEF_000C (h) ch2 : FFF0_000C (h)														
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	S1CHN					S1CHL					S1WDL				
R/W	R	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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	S0CHN					S0CHL					S0WDL				
R/W	R	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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
30-26	S1CHN[4:0]	Number of channel of sub frame 1 is set. This is valid only when the frame is 2 sub frame construction (SBFN of CNTREG is "1"), and is invalid when the frame is 1 sub frame construction (SBFN of CNTREG is "0".) Up to 32 channels are able to be specified, and S1CHN needs to be set to "number of channel – 1". Example 1 S1CHN = "00011": Sub frame 1 becomes 4 channel construction Example 2 S1CHN = "11111": Sub frame 1 becomes 32 channel construction S1WDL is valid only in 2 sub frame construction (SBFN of CNTREG is "1") and is invalid in 1 sub frame construction (SBFN of CNTREG is "0".)
25-21	S1CHL[4:0]	Channel length of the channel constructing sub frame 1 (bit length of channel) is set. 7 - 32 bit of channel length are available but 1 - 6 bit are prohibited. S1CHN needs to be set to "number of channel – 1". Example 1 S1CHL = "00110": Channel length becomes 7 bit Example 2 S1CHL = "11111": Channel length becomes 32 bit Channel length is able to be set to 32 or less regardless of RHLL value of CNTREG register. S1WDL is valid only in 2 sub frame construction (SBFN of CNTREG is "1") and is invalid in 1 sub frame construction (SBFN of CNTREG is "0".)
20-16	S1WDL[4:0]	Word length of the channel constructing sub frame 1 (bit length of channel) is set. 7 - 32 bit of word length are available but 1 - 6 bit are prohibited. S1WDL needs to be set to "word length – 1". Example 1 S1WDL = "00110": Word length becomes 7 bit Example 2 S1WDL = "11111": Word length becomes 32 bit RHLL of CNTREG register is "1": Set word length to 16 or less and channel length to shorter than the one set to S1CHL RHLL of CNTREG register is "0": Set word length to 32 or less and channel length to shorter than the one set to S1CHL S1WDL is valid only in 2 sub frame construction (SBFN of CNTREG is "1") and is invalid in 1 sub frame construction (SBFN of CNTREG is "0".)
15	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
14-10	S0CHN[4:0]	Number of channel of sub frame 0 is set up to 32 channels. S0CHN needs to be set to "number of channel – 1". Example 1 S0CHN = "00011": Sub frame 0 becomes 4 channel construction Example 2 S0CHN = "11111": Sub frame 0 becomes 32 channel construction

Bit field No.	Bit field Name	Description
9-5	S0CHL[4:0]	Channel length of the channel constructing sub frame 0 (bit length of channel) is set. 4 - 32 bit of channel length are available but 1 - 6 bit are prohibited. S0CHN needs to be set to "channel length – 1". Example 1 S0CHL = "00110": Channel length becomes 7 bit Example 2 S0CHL = "11111": Channel length becomes 32 bit The channel length can be set to 32 or less regardless of RHLL value of CNTREG register.
4-0	S0WDL[4:0]	Word length of the channel constructing sub frame 0 (number of bit in channel) is set. 4 - 32 bit of word length are available but 1-6 bit are prohibited. S0WDL needs to be set to "word length – 1". Example 1 S0WDL = "00110": Word length becomes 7 bit Example 2 S0WDL = "11111": Word length becomes 32 bit RHLL of CNTREG register is "1": Set word length to 16 or less and channel length to shorter than the one set to S0CHL RHLL of CNTREG register is "0": Set word length to 32 or less and channel length to shorter than the one set to S0CHL

17.6.6. I2SxMCR1REG register

This register controls enable and disable functions to each channel of sub frame 0.

Address	ch0 : FFEE_0010 (h) ch1 : FFEF_0010 (h) ch2 : FFF0_0010 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	S0CH 31	S0CH 30	S0CH 29	S0CH 28	S0CH 27	S0CH 26	S0CH 25	S0CH 24	S0CH 23	S0CH 22	S0CH 21	S0CH 20	S0CH 19	S0CH 18	S0CH 17	S0CH 16
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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	S0CH 15	S0CH 14	S0CH 13	S0CH 12	S0CH 11	S0CH 10	S0CH 09	S0CH 08	S0CH 07	S0CH 06	S0CH 05	S0CH 04	S0CH 03	S0CH 02	S0CH 01	S0CH 00
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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-0	S0CH31-S0CH00	Name (S0CHxx) of each bit indicates channel number xx of sub frame 0 (e.g. S0CH00 bit controls 0th channel of sub frame 0.) Thus, S0CH31 bit controls 31st channel of sub frame 0. 0: The corresponding channel is disabled Transmission/Reception are not performed to the disabled channel 1: The corresponding channel is enabled Transmission/Reception are performed to the enabled channel

17.6.7. I2SxMCR2REG register

This register is to control enable and disable functions to each channel of sub frame 1.

Address	ch0 : FFEE_0014 (h) ch1 : FFEF_0014 (h) ch2 : FFF0_0014 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	S1CH 31	S1CH 30	S1CH 29	S1CH 28	S1CH 27	S1CH 26	S1CH 25	S1CH 24	S1CH 23	S1CH 22	S1CH 21	S1CH 20	S1CH 19	S1CH 18	S1CH 17	S1CH 16
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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	S1CH 15	S1CH 14	S1CH 13	S1CH 12	S1CH 11	S1CH 10	S1CH 09	S1CH 08	S1CH 07	S1CH 06	S1CH 05	S1CH 04	S1CH 03	S1CH 02	S1CH 01	S1CH 00
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
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31-0	S1CH31-S1CH00	Name (S1CHxx) of each bit indicates channel number xx of sub frame 1 (e.g. S1CH00 bit controls 0th channel of sub frame 1.) Thus, S1CH31 bit controls 31st channel of sub frame 1. When frame is 1 sub frame construction (SBFN of CNTREG is "0"), this is invalid. 0: The corresponding channel is disabled Transmission/Reception are not performed to the disabled channel 1: The corresponding channel is enabled Transmission/Reception are performed to the enabled channel

17.6.8. I2SxOPRREG register

Address	ch0 : FFEE_0018 (h) ch1 : FFEF_0018 (h) ch2 : FFF0_0018 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)							RXENB	(Reserved)							TXENB
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															start
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-25	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
24	RXENB	Enable/Disable functions of receiving operation is set. 0: Receiving operation is disabled Reception FIFO becomes empty with writing "0" to this bit When RXENB is "0", the data received from serial reception bus is not written to reception FIFO DMA reception channel stops during DMA transfer 1: Receiving operation is enabled
23-17	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
16	TXENB	Enable/Disable functions of transmitting operation is set. 0: Transmitting operation is disabled Reception FIFO becomes empty with writing "0" to this bit When TXENB is "0", the data written to TXFDAT register from CPU or DMA is not written to transmission FIFO DMA reception channel stops during DMA transfer 1: Transmitting operation is enabled
15-1	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
0	start	I2S is enabled/disabled. 0: I2S is stop, and internal transmission/reception FIFO becomes empty by writing "0" to this bit 1: I2S is operable Prohibit overwriting CNTREG, MCR0REG, MCR1REG, and MCR2REG registers when Start is "1".

17.6.9. I2SxSRST register

This register is to control I2S software reset.

Address	**ch0 : FFEE_001C (h) ch1 : FFEF_001C (h) ch2 : FFF0_001C (h)**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															SRST
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		**Description**
No.	**Name**	
31-1	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
0	SRST	Software reset is performed by writing "1". STATUS register and each internal state machine become initial state by software reset, and transmission/reception FIFO becomes empty. There is no influence in registers other than STATUS, INTCNT, and DMAACT registers. When read value is "0" after writing "1", it indicates software reset is completed. "1" indicates software reset is in process.

17.6.10. I2SxINTCNT register

Address	ch0 : FFEE_0020 (h) ch1 : FFEF_0020 (h) ch2 : FFF0_0020 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	TXUD1M	TBERM	FERRM	TXUD0M	TXOVM	TXFDM	TXFIM	(Reserved)		RBERM	RXUDM	RXOVM	EOPM	RXFDM	RXFIM
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Initial	0	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)				TFTH				(Reserved)		RPTMR		RFTH			
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
30	TXUD1M	This is transmission FIFO underflow interrupt mask bit. It becomes "1" by software reset. 0: Interrupt to CPU by TXUDR1 of STATUS register is not masked 1: Interrupt to CPU by TXUDR1 of STATUS register is masked
29	TBERM	This is interrupt mask bit of block size error of transmission channel. It becomes "1" by software reset. 0: Interrupt to CPU by TBERR of STATUS register is not masked 1: Interrupt to CPU by TBERR of STATUS register is masked
28	FERRM	This is frame error interrupt mask bit. It becomes "1" by software reset. 0: Interrupt to CPU by FERR of STATUS register is not masked 1: Interrupt to CPU by FERR of STATUS register is masked.
27	TXUD0M	This is transmission FIFO underflow interrupt mask bit. It becomes "1" by software reset. 0: Interrupt to CPU by TXUDR0 of STATUS register is not masked. 1: Interrupt to CPU by TXUDR0 of STATUS register is masked.
26	TXOVM	This is transmission FIFO overflow interrupt mask bit. It becomes "1" by software reset. 0: Interrupt to CPU by TXOVM of STATUS register is not masked. 1: Interrupt to CPU by TXOVM of STATUS register is masked.
25	TXFDM	This is DMA request mask register bit. It becomes "1" by software reset. 0: DMA transfer is requested when reception data written to transmission FIFO is threshold value or more 1: DMA transfer is not requested even reception data written to transmission FIFO is threshold value or more

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>24</td><td>TXFIM</td><td>This is transmission FIFO interrupt mask bit.
It becomes "1" by software reset.<table><tr><td>0</td><td>Interrupt to CPU by TXFI of STATUS register is not masked</td></tr><tr><td>1</td><td>Interrupt to CPU by TXFI of STATUS register is masked</td></tr></table></td></tr>
<tr><td>23-22</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>21</td><td>RBERM</td><td>This is interrupt mask bit of reception channel block size error.
It becomes "1" by software reset.<table><tr><td>0</td><td>Interrupt to CPU by RBERR of STATUS register is not masked</td></tr><tr><td>1</td><td>Interrupt to CPU by RBERR of STATUS register is masked</td></tr></table></td></tr>
<tr><td>20</td><td>RXUDM</td><td>This is reception underflow interrupt mask bit.
It becomes "1" by software reset.<table><tr><td>0</td><td>Interrupt to CPU by RXUDR of STATUS register is not masked</td></tr><tr><td>1</td><td>Interrupt to CPU by RXUDR of STATUS register is masked</td></tr></table></td></tr>
<tr><td>19</td><td>RXOVM</td><td>This is interrupt mask bit of reception FIFO overflow.
It becomes "1" by software reset.<table><tr><td>0</td><td>Interrupt to CPU by RXOVR of STATUS register is not masked</td></tr><tr><td>1</td><td>Interrupt to CPU by RXOVR of STATUS register is masked</td></tr></table></td></tr>
<tr><td>18</td><td>EOPM</td><td>This is interrupt mask bit by EOPI of STATUS register.
It becomes "1" by software reset.<table><tr><td>0</td><td>Interrupt to CPU by EOPI of STATUS register is not masked</td></tr><tr><td>1</td><td>Interrupt to CPU by EOPI of STATUS register is masked</td></tr></table></td></tr>
<tr><td>17</td><td>RXFDM</td><td>This is reception DMA request mask bit.
It becomes "1" by software reset.<table><tr><td>0</td><td>DMA transfer is requested when reception data written to reception FIFO is threshold value or more</td></tr><tr><td>1</td><td>DMA transfer is not requested though reception data written to reception FIFO is threshold value or more</td></tr></table></td></tr>
<tr><td>16</td><td>RXFIM</td><td>This is reception FIFO interrupt mask bit.
It becomes "1" by software reset.<table><tr><td>0</td><td>Interrupt to CPU by RXFI of STATUS register is not masked</td></tr><tr><td>1</td><td>Interrupt to CPU by RXFI of STATUS register is masked</td></tr></table></td></tr>
<tr><td>15-12</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>11-8</td><td>TFTH[3:0]</td><td>Threshold value of transmission FIFO is set.
Empty space of transmission FIFO is threshold value or more and TXFIM is "0": Interrupt to CPU occurs
Empty space of transmission FIFO is threshold value or more and TXFDM is "0": DMA is requested to DMAC

TFTH is set according to the following expressions.
TFTH = Transmission FIFO threshold – 1</td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>7-6</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>5-4</td><td>RPTMR[1:0]</td><td>This is packet reception completion timer setting bit which sets time-out value of the internal reception completion timer.
Reception FIFO is not empty and number of its data is smaller than threshold value: The timer always counts up
Reception FIFO is empty or the data value is threshold value or more: The timer is cleared. When the timer becomes time-out, EOPI bit of STATUS register is set to "1".
The timer becomes "00" by software reset.
<table><tr><td>00</td><td>0 (the timer is not in operation)</td></tr><tr><td>01</td><td>54000 AHB clock cycles</td></tr><tr><td>10</td><td>108000 AHB clock cycles</td></tr><tr><td>11</td><td>216000 AHB clock cycles</td></tr></table></td></tr>
<tr><td>3-0</td><td>RFTH[3:0]</td><td>Threshold value of reception FIFO is set.
Number of reception word written to reception FIFO is threshold value or more and RXFIM is "0": Interrupt to CPU occurs
Number of reception word written to reception FIFO is threshold value or more and RXFDM is "0": DMA is requested to DMAC

RFTH is set according to the following expressions.
RFTH = Reception FIFO threshold – 1</td></tr>
</table>

17.6.11. I2SxSTATUS register

Address	ch0 : FFEE_0024 (h) ch1 : FFEF_0024 (h) ch2 : FFF0_0024 (h)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TBERR	RBERR	FERR	TXUDR1	TXUDR0	TXOVR	RXUDR	RXOVR	(Reserved)				EOPI	BSY	TXFI	RXFI
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R	R	R
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)		TXNUM						(Reserved)		RXNUM					
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Description
No.	Name	
31	TBERR	In order to set block size of DMA transmission channel to larger value than I2S transmission FIFO threshold (TFTH+1) to operate, this bit is set to "1" and I2S stops the transfer. When TBERR is "1" and TBERM of the INTCNT register is "0", interrupt to CPU occurs. This bit becomes "0" by software and hardware reset.
30	RBERR	In order to set block size of DMA reception channel to larger value than I2S reception FIFO threshold (RFTH+1) to operate, this bit is set to "1" and stop the channel. When RBERR is "1" and RBERM of the INTCNT register is "0", interrupt to CPU occurs. This bit becomes "0" by software and hardware reset.
29	FERR	Occurrence of frame error is indicated. This bit is set to "1" in the following cases: • Frame synchronous signal is not able to be received with the set frame rate in the free-running mode (FRUN = 0 of CNTREG) and the slave mode (MSMD = 0 of CNTREG) • The next frame synchronous signal is received during frame transmission/reception in the slave mode (MSMD = 0 of CNTREG), not free-running mode (FRUN = 1 of CNTREG) When FERR is "1" and FERRM of INTCNT register is "0", interrupt to CPU occurs. Writing "1" from CPU clears the value to "0". This becomes "0" by software reset.
28	TXUDR1	When transmission FIFO underflows at the top of frame, the value is set to "1". Writing "1" from CPU clears the value to "0". This becomes "0" by software reset.
27	TXUDR0	When transmission FIFO underflows during frame transmission (from 2nd bit word to the last frame of the word), the value is set to "1". Writing "1" from CPU clears the value to "0". This becomes "0" by software reset.
26	TXOVR	When transmission FIFO overflows, the value is set to "1" indicating transmission data is written in the condition that transmission FIFO is full. The value "1" indicates 1 word or more of transmission data is deleted. When TXOVR is "1" and TXOVM of INTCNT register is "0", interrupt to CPU occurs. Writing "1" from CPU clears the value to "0". This becomes "0" by software reset.
25	RXUDR	When reception FIFO underflows, the value is set to "1" indicating read access is carried out to reception FIFO in the condition that reception FIFO is empty. Writing "1" from CPU clears the value to "0". This becomes "0" by software reset.
24	RXOVR	When reception FIFO overflows, the value is set to "1" indicating reception is carried out in the condition that reception FIFO is full. The value "1" indicates 1 word or more of reception data is deleted. When RXOVR is "1" and RXOVM of INTCNT register is "0", interrupt to CPU occurs. Writing "1" from CPU clears the value to "0". This becomes "0" by software reset.
23-20	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>19</td><td>EOPI</td><td>This is interrupt flag containing reception timer. The timer is enabled when following conditions are met at the same time:
• RXDIS of CNTREG register is set to "0"
• RXFDM of INTCNT register is set to "0"
• MSMD of CNTREG register is set to "0"
• start bit of OPRREG register is set to "1" and RXENB = "1"

After the reset, operation starts with the 1st word reception. Then the value is cleared every time word is received. When reception FIFO is not empty at the time set to RPTMR of INTCNT register, the value is set to "1". When EOPI is "1" and EOPM of INTCNT register is "0", interrupt to CPU occurs. The value is automatically cleared if reception FIFO data is threshold or more, or it becomes empty.
Writing "1" from CPU clears the value to "0".
This becomes "0" by software reset.</td></tr>
<tr><td>18</td><td>BSY</td><td>Serial transmission control part is busy state. This bit is not affected by software reset.
<table><tr><td>0</td><td>Serial transmission control part is in idle</td></tr><tr><td>1</td><td>Serial transmission control part is in busy</td></tr></table></td></tr>
<tr><td>17</td><td>TXFI</td><td>When empty slot of transmission FIFO is larger than the threshold set in TFTH of INTCNT register, this bit is set to "1".

This bit is "1" and TXFIM bit of INTCNT register is "0": Interrupt to CPU occurs
This bit is "1" and TXFDM bit of INTCNT register is "0": DMA is requested

When number of empty slot of reception FIFO becomes smaller than the threshold by writing to TXFDAT register from CPU or DMAC, this bit is cleared automatically to "0".
The value is also become "0" when start bit of start register is "0" and TXENB bit of OPRREG register is "0". If software reset is performed at start bit = "1" and TXENB bit = "1", the value becomes "0" during software reset then changes to "1" after the process.</td></tr>
<tr><td>16</td><td>RXFI</td><td>When number of reception FIFO data is larger than the threshold set in RFTF of INTCNT register, this bit is set to "1".

This bit is "1" and RXFIM bit of INTCNT register is "0": Interrupt to CPU occurs
This bit is "1" and RXFDM bit of INTCNT register is "0": DMA is requested

When number of data in reception FIFO becomes smaller than the threshold by reading RXFDAT register from CPU or DMAC, this bit is automatically cleared to "0".
When start bit of start register is "0" or RXENB bit of OPRREG register is "0", this bit becomes "0".
This becomes "0" by software reset.</td></tr>
<tr><td>15-14</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>13-8</td><td>TXNUM[5:0]</td><td>The number of data in transmission FIFO is indicated.
This bit is incremented by write access to TXFDAT register and decremented by serial word transfer.
Max. value of 18 can be displayed in the simultaneous transfer mode, and value of 36 in the transmission only mode.
This becomes "0" by software reset.</td></tr>
<tr><td>7-6</td><td>(Reserved)</td><td>Reserved bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>5-0</td><td>RXNUM[5:0]</td><td>The number of data in reception FIFO is indicated.
This bit is incremented by word reception from serial bus and decremented by read access to RXFDAT register.
Max. values of 18 can be displayed in the simultaneous transfer mode, and value of 36 in the reception mode.
This becomes "0" by software reset.</td></tr>
</table>

17.6.12. I2SxDMAACT register

Address	ch0：FFEE_0028 (h) ch1：FFEF_0028 (h) ch2：FFF0_0028 (h)														
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															TDMACT
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															RDMACT
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-17	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
16	TDMACT	Transmission channel of DMAC (DMA controller) is activated. After transfer channel starts, software should write "1" to TDMACT to teach I2S that the transfer channel is active. When TDMACT is "0", transfer request of transmission channel block is not sent to DMAC. I2S automatically clears TDMACT every time DMA packet transmission completes. Writing "0" from CPU clears the value to "0". This becomes "0" by software reset. 0: Transmission channel of DMAC is stop that TXDREQ is unable to be detected 1: Transmission channel of DMAC is activated that TXDREQ is able to be detected
15-1	(Reserved)	Reserved bits. The write access is ignored. The read value of these bits is always "0".
0	RDMACT	The reception channel of DMAC (DMA controller) is activated. After reception channel starts, software should write "1" to RDMACT to teach I2S that the channel is active. When RDMACT is "0", transfer request of reception channel block is not sent to DMAC. I2S automatically clears RDMACT every time DMA packet reception completes. Writing "1" from CPU clears the value to "0". This becomes "0" by software reset. 0: Reception channel of DMAC is stop that RXDREQ is unable to be detected 1: Reception channel of DMAC is active that RXDREQ is able to be detected

17.7. Operation

17.7.1. Outline

This module is synchronous serial interface which enables full duplex and multiplexer channel.
It is also able to correspond to various frame formats by register setting.
(Refer to "17.7.3 Frame construction" for detail.)

This module is also able to operate as master and slave. In the master mode, clock (I2S_SCKx) and frame synchronous signal (I2S_WSx) are output to the external slave. In the slave mode, they are input from the external master.

During the master mode, I2S_SCKx clock can be output by dividing external clock (I2S_external clock x) or internal clock (it is selectable at register). Frame synchronous signal can be generated by free-running or burst mode (generated only when there is transmission data.)

This module equips transmission/reception FIFO, and its depth varies depending on mode: transmission only mode is 36 word × 32 bit constructive transmission FIFO and reception only mode is 18 word × 32 bit constructive reception FIFO. Refer to "17.7.3 Frame construction" for more detail.
Internal transfer between transmission and reception FIFO and internal system memory is able to be performed by DMA, interrupt, and polling.

17.7.2. Transfer start, stop, and malfunction

Transmission only mode

Transfer setting	Operation	Master mode (MSMD = 1)	Slave mode (MSMD = 0)
Transmission only TXDIS = 0 RXDIS = 1	**Start**	**Free-running mode (FRUN = 1):** After start bit becomes "1" and TXENB bit is "1", frame synchronous signal starts to output when transmission FIFO is not empty. From the 2nd time, it outputs frame synchronous signal with the frame rate determined by the register setting. If transfer FIFO is empty, empty frame is output at the same time of frame synchronous signal output. Serial data of the empty frame is able to be set to "0" or "1" by the register setting. **Burst mode (FRUN = 0):** When start bit is "1" and TXENB bit is "1", frame synchronous signal is output if transfer FIFO is not empty. Always confirm transmission FIFO status at the end of 1 frame output or at idle to output the signal if transfer FIFO is not empty.	**Free-running mode (FRUN = 1):** The frame rate determined by the register setting inputs frame synchronous signal. If transmission FIFO is empty at inputting frame synchronous signal with start bit is "1" and TXENB bit is "1", empty frame is output. Serial data of the empty frame is able to be set to "0" or "1" by the register setting. **Burst mode (FRUN=0):** When start bit is "1" and TXENB bit is "1", 1 frame is output every time frame synchronous signal is input. When transmission FIFO is empty at the time of frame synchronous signal input, empty frame is output.
	Stop	At the time of stop, transmission FIFO becomes empty with having no data transfer from internal memory to I2S transmission FIFO. **To maintain start bit to "1"** TXENB = "1": Keep outputting frame synchronous signal in the free-running mode. When transmission FIFO becomes empty, empty frame is output; however, do not output frame synchronous signal in the burst mode. Output empty frame bit to serial data bus. TXENB = "0": When "0" is written to TXENB, transmission FIFO becomes empty that the data in the FIFO at writing "0" is not sent. Although frame synchronous signal continues outputting in the free-running mode, serial bus becomes in high impedance state. In the burst mode, frame synchronous signal is not output and serial data bus becomes in high impedance state. **To make start bit "0"** Write "0" to start bit, then transmission FIFO becomes empty. Stop clock supply to the serial control part regardless of TXENB setting, and do not output clock to external part. Frame synchronous signal output should also be stopped. Serial data bus becomes in high impedance state.	**To maintain start bit to "1"** TXENB = "1": Output empty frame data to serial bus. TXENB = "0": Write "0" to TXENB, then transmission FIFO becomes empty that the data in the FIFO at writing "0" is not sent. Data writing to transmission FIFO and transmission frame detection are stop. Serial data bus becomes in high impedance state. **To make start bit "0"** Write "0" to start bit, then transmission FIFO becomes empty. Writing to transmission FIFO and detection of transmission frame synchronous signal are stop regardless of TXENB setting.

Transfer setting	Operation	Master mode (MSMD = 1)	Slave mode (MSMD = 0)
	Abnormality	When reading to transmission FIFO occurs with having it empty, empty frame is output. When writing to transmission FIFO occurs with having it full, set TXOVR to "1".	When reading to transmission FIFO occurs with having it empty, empty frame is output. However do not set TXUDR to "1" for the 1st output frame after bit becomes Start = "1" and TXENB = "1". When writing to transmission FIFO occurs with having it full, set TXOVR to "1". If it is not input with the frame rate defined frame synchronous signal in the free-running mode, set FERR bit of the register to "1". If the next frame synchronous signal is input before completing 1 frame transmission in the burst mode, set FERR bit of the register to "1"

Note:
1. TXDIS and RXDIS are for setting to enable and disable transmission/reception of CNTREG register.
2. Start, TXENB, and RXENB are operation control bits of OPRREG register.
3. Empty frame bit is determined by MSKB of CNTREG register.

Reception only mode

Transfer setting	Operation	Master mode (MSMD = 1)	Slave mode (MSMD = 0)
Reception only TXDIS = 1 RXDIS = 0	**Start**	**Free-running mode (FRUN = 1):** Frame synchronous signal starts to output after start bit becomes "1" and TXENB bit is "1" when transmission FIFO is not empty. From the 2nd time, output frame synchronous signal with the frame rate determined by the register setting. **Burst mode (FRUN = 0):** When start bit is "1" and RXENB bit is "1", output frame synchronous signal to receive frame if reception FIFO is not full. If the FIFO is full, the signal does not output.	**Free-running mode (FRUN = 1):** When start bit is "1" and RXENB bit is "1", input frame synchronous signal with the frame rate determined by the register setting. Frame should be received every time the signal is input. **Burst mode (FRUN = 0):** When start bit is "1" and RXENB bit is "1", perform frame reception every time frame synchronous signal is input. The signal is input with less speed than the frame rate in the free-running mode.
	Stop	At the time of stop, frame is not imported from serial bus even though reception FIFO is empty in the condition that data transfer from I2S reception FIFO to internal memory is not required. **To maintain start bit to "1"** Write "0" to RXENB and empty reception FIFO. Although frame synchronous signal is kept outputting in the free-running mode, frame is not received. In the burst mode, frame is not received and the signal is not output. **To make start bit "0"** Write "0" to start bit, then reception FIFO becomes empty. Clock supply to the serial control part stops regardless of RXENB setting, and I2S_SCKx supply to the external part is stop as well.	**To maintain start bit to "1"** Reception FIFO becomes empty by "0" writing to RXENB. Ignore the input frame synchronous signal, and do not receive the frame. **To make the start bit "0"** Write "0" to the start bit, then reception FIFO becomes empty. Ignore the input frame synchronous signal regardless of RXENB setting, and do not receive the frame.
	Abnormality	When writing to reception FIFO occurs with having it full, set RXOVR to "1". The bit also should be set to "1" when read access to reception FIFO occurs with having it empty.	When writing to reception FIFO occurs with having it full, set RXOVR of the STATUS register to "1". When read access to reception FIFO occurs with having it empty, set RXUDR of the register to "1". Free-running mode: If frame synchronous signal is not input with the frame rate defined by the register setting, set FERR bit of the register to "1". Burst mode: Set the bit also to "1" if the next frame synchronous signal is input during 1 frame reception.

Note:
1. TXDIS and RXDIS are for setting to enable and disable transmission/reception of CNTREG register.
2. Start, TXENB, and RXENB are operation control bits of OPRREG register.

Simultaneous transfer mode

Transfer setting	Operation	Master mode (MSMD = 1)	Slave mode (MSMD = 0)
Simultaneous transfer TXDIS = 0 RXDIS = 0	**Start**	**Free-running mode (FRUN = 1):** Status of Start = 1, TXENB = 1, and RXENB = 1: The same operation as transmission only mode. Status of Start = 1, TXENB = 0, and RXENB = 1: The same operation as reception only mode. Status of Start = 1, TXENB = 1, and RXENB = 1: Frame synchronous signal is output from the state that transmission FIFO is not empty and reception FIFO is not full. Then output frame synchronous signal with the frame rate defined by the register setting; at the same time, output empty frame if reception FIFO is empty. Empty frame's serial data is able to be set to "0" or "1" at the register setting. Every time frame synchronous signal is output, receive frame. **Burst mode (FRUN = 0):** Status of Start = 1, TXENB = 1, and RXENB = 0: The same operation as transmission only mode. Status of Start = 1, TXENB = 0, and RXENB = 1: The same operation as reception only mode. Status of Start = 1, TXENB = 1, and RXENB = 1: Frame synchronous signal is output from the state that transmission FIFO is not empty and reception FIFO is not full. After completion of 1 frame output or at idle state, always confirm transmission/reception FIFO status. If transmission FIFO is not empty and reception FIFO is not full, output frame synchronous signal to perform frame transmission/reception.	**Free-running mode (FRUN = 1):** Status of Start = 1, TXENB = 1, and RXENB = 0: The same operation as transmission only mode. Status of Start = 1, TXENB = 0, and RXENB = 1: The same operation as reception only mode. Status of Start = 1, TXENB = 1, and RXENB = 1: Frame synchronous signal is input with the frame rate defined by the register setting; at the same time, output empty frame if transmission FIFO is empty. Its serial data is able to be set to "0" or "1" at the register setting. Every time frame synchronous signal is input, receive frame. **Burst mode (FRUN = 0):** Every time frame synchronous signal is input with start bit is "1", transmission and reception for 1 frame is performed. When the signal is input, output empty frame if transmission FIFO is empty.

Transfer setting	Operation	Master mode (MSMD = 1)	Slave mode (MSMD = 0)
	Stop	Stop operation has following states: Transmission stop: Transmission FIFO becomes empty without sending data from internal memory to I2S transmission FIFO. Reception stop: Data does not need to be transferred from I2S reception FIFO to internal memory. **To maintain start bit to "1"** Keep outputting frame synchronous signal in the free-running mode. In the burst mode, do not output the signal when transmission FIFO becomes empty. Transmission stop: TXENB = 1: Keep outputting empty frame bit when transmission FIFO becomes empty. TXENB = 0: Transmission FIFO becomes empty and transmission serial data bus becomes in high impedance. Do not send the data in transmission FIFO at writing "0" to TXENB. Writing to transmission FIFO stops. Reception stop: Write "0" to RXENB, then reception FIFO becomes empty and frame reception operation stops. **To make start bit "0"** Write "0" to start bit, then transmission/reception FIFO becomes empty. The clock supply to the internal serial control part stops regardless of TXENB and RXENB statuses as well as I2S_SCKX output to the external part and frame synchronous signal output.	**To maintain start bit to "1"** Transmission stop: Keep outputting empty frame bit after transmission FIFO becomes empty in order to maintain this bit to TXENB = 1. When the value is changed to "0", transmission FIFO becomes empty and transmission serial data bus becomes in high impedance. Do not send the data in transmission FIFO at writing "0" to TXENB. Stop writing to transmission FIFO. Reception stop: Write "0" to RXENB, then reception FIFO becomes empty and frame reception operation stops. **To make start bit "0"** Write "0" to start bit, then transmission/reception FIFO becomes empty. Stop transmission/ reception regardless of TXENB and RXENB statuses.
	Abnormality	When reading to transmission FIFO occurs with having it empty, output empty frame bit. When writing to transmission FIFO occurs with having it full, set TXOVR to "1". If read access to reception FIFO occurs while it is empty, set RXUDR of STATUS register to "1". If writing to reception FIFO occurs with having it full, set RXOVR of the register to "1".	When reading to transmission FIFO occurs with having it empty, output empty frame bit. When writing to transmission FIFO occurs with having it full, set TXOVR to "1". When read access occurs to reception FIFO with having it empty, set RXUDR of STATUS register to "1". When writing to reception FIFO occurs with having it full, set RXOVR of the register to "1". If it is not input with the frame rate defined frame synchronous signal in the free-running mode, set FERR bit of the register to "1". If the next frame synchronous signal is input before completing 1 frame transmission in the burst mode, set FERR bit of the register to "1".

Note:
1. TXDIS and RXDIS are for setting to enable and disable transmission/reception of CNTREG register.
2. Start, TXENB, and RXENB are operation control bits of OPRREG register.
3. Empty frame bit is determined by MSKB of CNTREG register.

17.7.3. Frame construction

This module supports frame format of multiplexer channel construction. Frame is able to be set to 1 or 2 sub frames; moreover, number of each frame's channel and word length are able to be set individually.

17.7.3.1. 1 sub frame construction

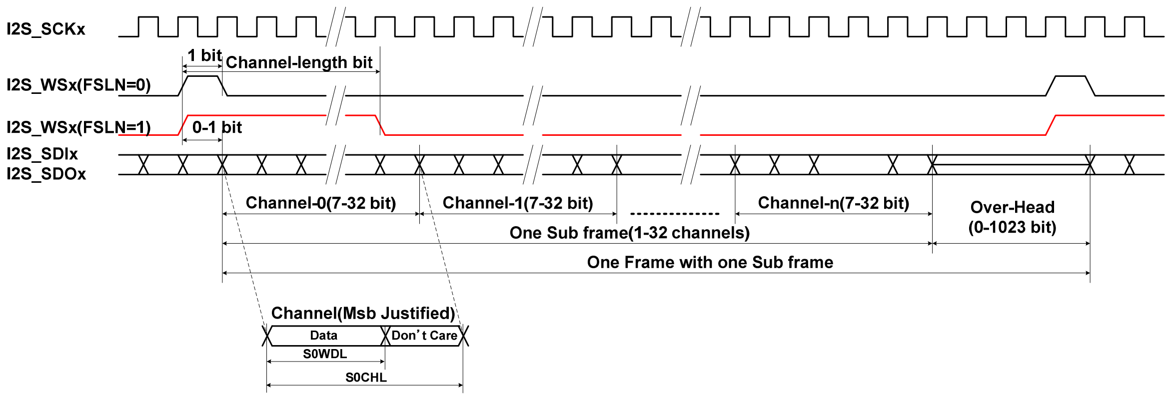


Figure 17-2 1 sub frame composite frame

Description

1. When SBFN bit of CNTREG register is "0", frame becomes 1 sub frame composite.
2. Number of channel of 1 sub frame is determined by S0CHN of MC0REG register.
   Up to 32 channels are settable.
3. Each channel bit length (word length) is determined by S0WDL of MC0REG register.
4. Sub frame channel starts from 0th, and each channel is settable to valid/invalid with the corresponding bit of MC1REG register. Transmission/Reception of data is not performed to invalid channel.
5. Dummy bit can be inserted behind sub frame by setting OVHD of CNTREG register.
   0-1023 bit are insertable.
6. Polarity of I2S_WSx is set with FSPL bit of CNTREG register.
7. Pulse width of I2S_WSx can be set to 1 bit or 1 word length by setting FSLN bit of CNTREG register.
8. Timing from the edge I2S_WSx becomes valid to the first bit of frame is settable to "0" or "1" bit.
9. In this construction, setting of S1CHN of MC0REG register, S1WDL register and MC2REG register are ignored.

17.7.3.2. 2 sub frame construction

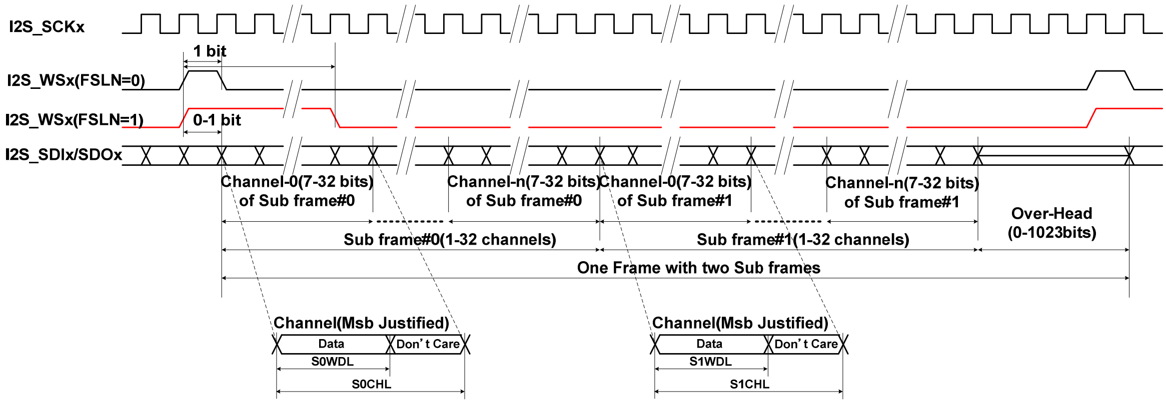


Figure 17-3 2 sub frame composite frame

Description

1. When SBFN bit of CNTREG register is "1", frame becomes 2 sub frame composite that first sub frame is 0 and the next one is 1.
2. Set number of channel of sub frame 0 to S0CHN of MC0REG register, and set number of sub frame 1 channel to S1CHN of the register. Those numbers of channel are individual that they do not need to have the same channel. Up to 32 channels are settable.
3. Channel bit length (word length) of sub frame 0 is determined by S0WDL of MC0REG register. For sub frame 1, they are determined by S1WDL of MC0REG register. Since channel bit length of the sub frame is individual, those channels (word length) do not need to be the same.
4. Sub frame channel starts from 0th. Each channel of sub frame 0 is settable to valid/invalid with the corresponding bit of MC1REG register, and corresponding bit of MC2REG register for sub frame 1.
   Transmission/Reception of data is not performed to invalid channel.
5 Dummy bit can be inserted behind sub frame 1 by setting OVHD of CNTREG.
   0-1023 bit are insertable.
6. Polarity of I2S_WSx is set to FSPL bit of CNTREG register.
7. Pulse width of I2S_WSx can be set to 1 bit or 1 channel length by setting FSLN bit of CNTREG register.
   Channel length setting of 1 channel is determined by the channel length of sub frame 0.
8. Timing from the edge I2S_WSx becomes valid to the first bit of frame is settable to "0" or "1" bit.

17.7.3.3. Bit alignment

(1) Transmission word alignment

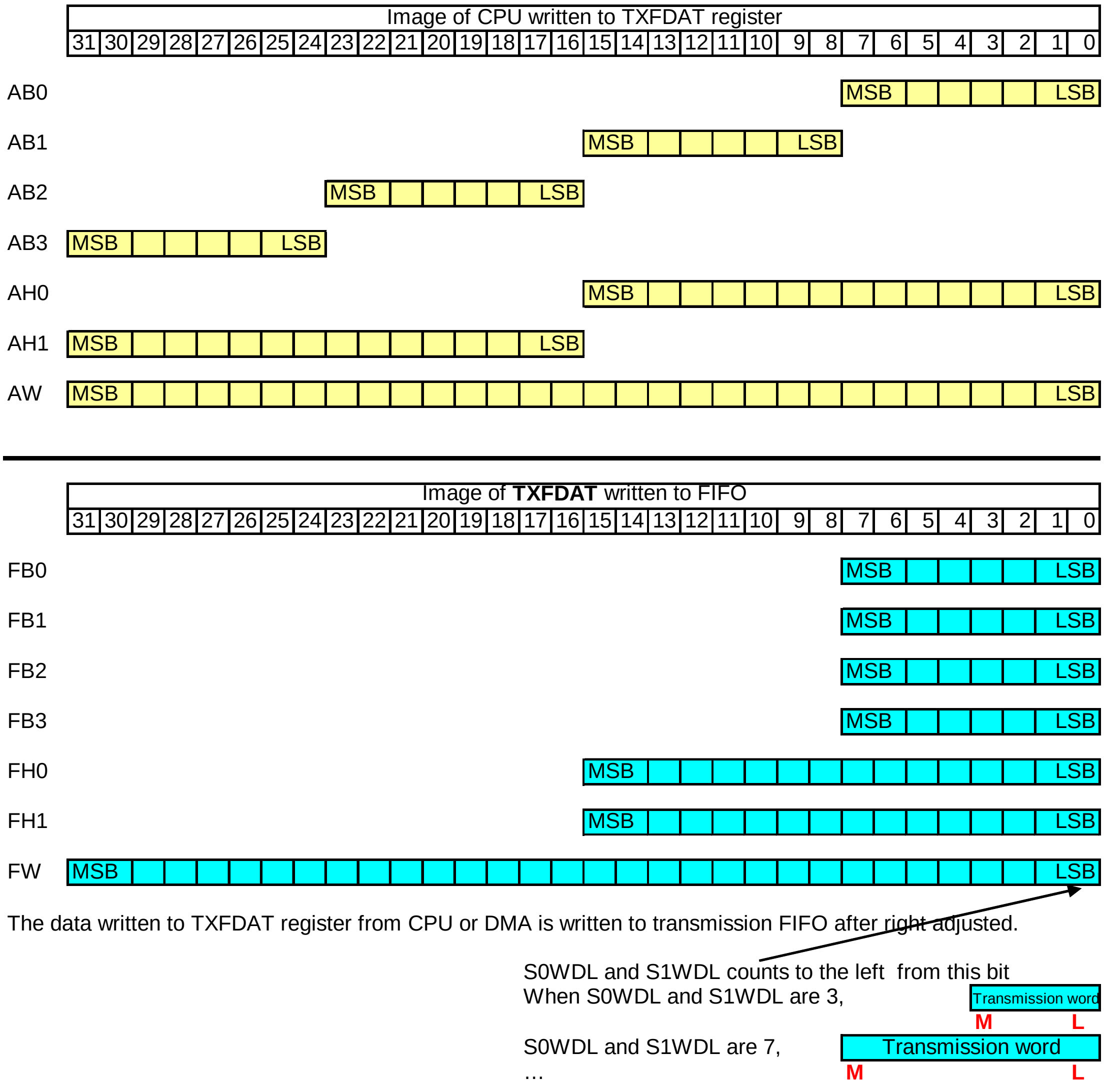


Figure 17-4 Transmission word line chart

When transmission is performed with serial bus, word is sent from M bit when CNTREG register's MLSB is "0" and from L bit when the value is "1". When channel length (set to S0CHL and S1CHL) is longer than the word length (set to S0WDL and S1WDL), remaining bit in the channel becomes CNTREG[MSKB]. If channel length is shorter than the word's, setting is prohibited.

Note:

AB0, AB1, AB2, AB3, AH0, AH1, and AW on the above chart indicate byte 0, byte 1, byte 2, byte 3, half word 0, half word 1, and word at write accessing to TXFDAT on AHB bus.
Each FB0, FB1, FB2, FB3, FH0, FH1, and FW indicate AB0, AB1, AB2, AB3, AH0, AH1, and AW are written to transmission FIFO after they are right justified.

(2) Reception word alignment

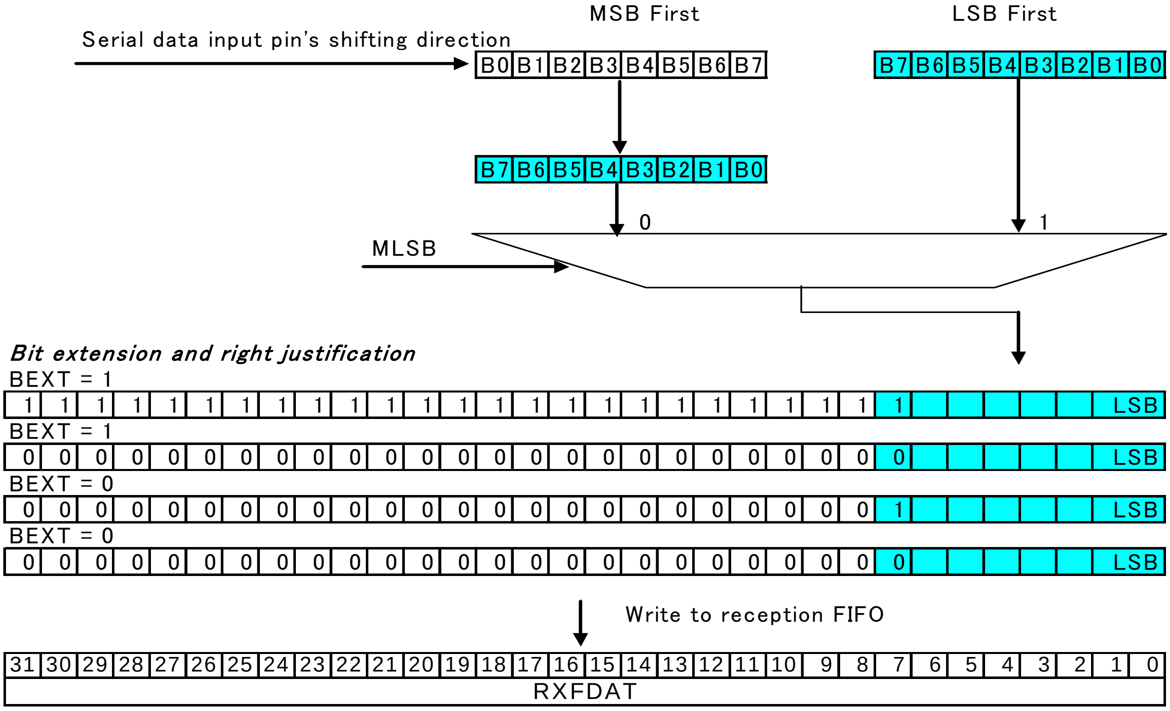


Figure 17-5 Reception word line chart

This chart shows word line example of when word length is 8.
The word received from serial bus is always written to reception FIFO after right-justified.
Therefore, read access should be performed from AHB bus to RXFDAT in order to read as follows:

Word length

- 8 or less: Byte 0
- 9 – 16: Half word 0
- 17 – 32: All words.

17.7.4. FIFO construction and description

Simultaneous transfer mode (TXDIS = 0 and RXDIS = 0)

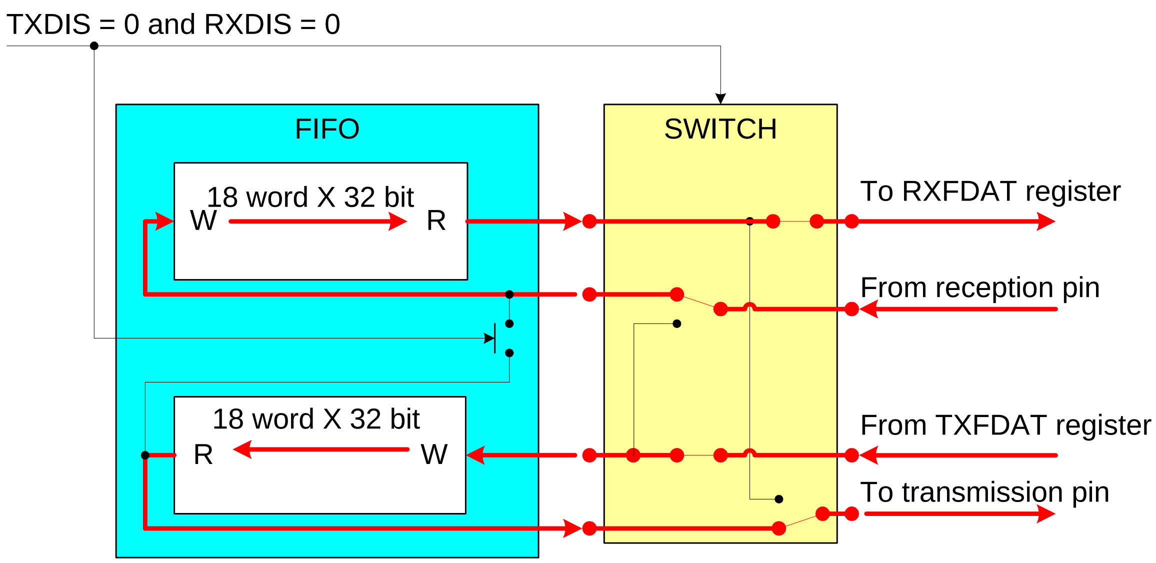


Figure 17-6 Simultaneous transfer mode data flow

With setting TXDIS = 0 and RXDIS = 0 of CNTREG register, the mode becomes simultaneous transfer mode which operates in 18 word × 32 bit transmission FIFO and reception FIFO.

Transmission only mode (TXDIS = 0 and RXDIS = 1)

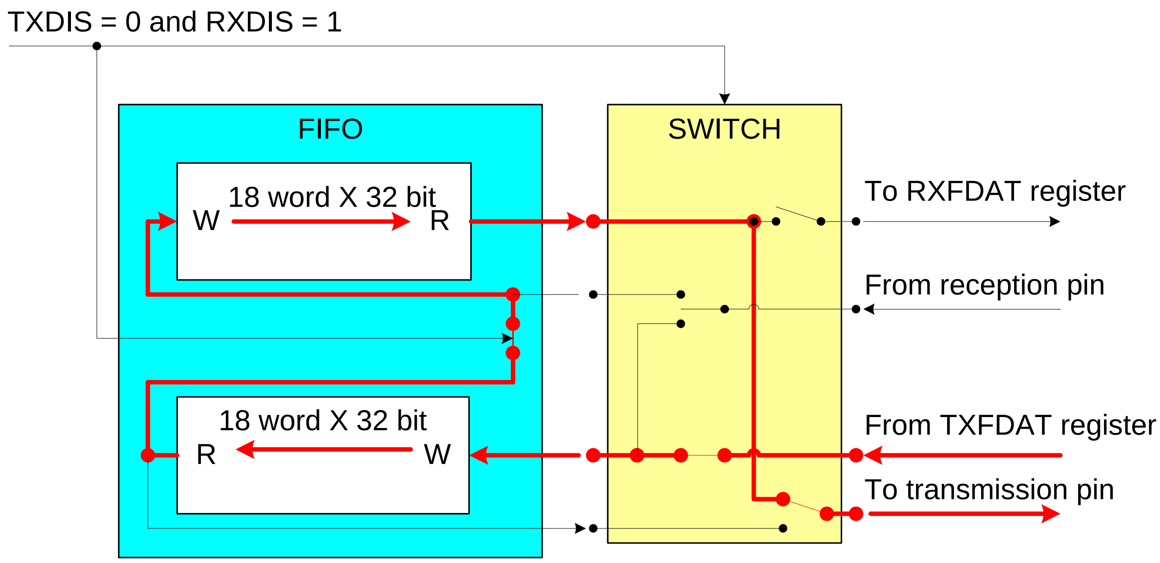


Figure 17-7 Transmission only mode data flow

With setting TXDIS = 0 and RXDIS = 1 of CNTREG register, the mode becomes transmission only mode which operates in 36 word × 32 bit transmission FIFO, and reception is not performed.

Reception only mode (TXDIS = 1 and RXDIS = 0)

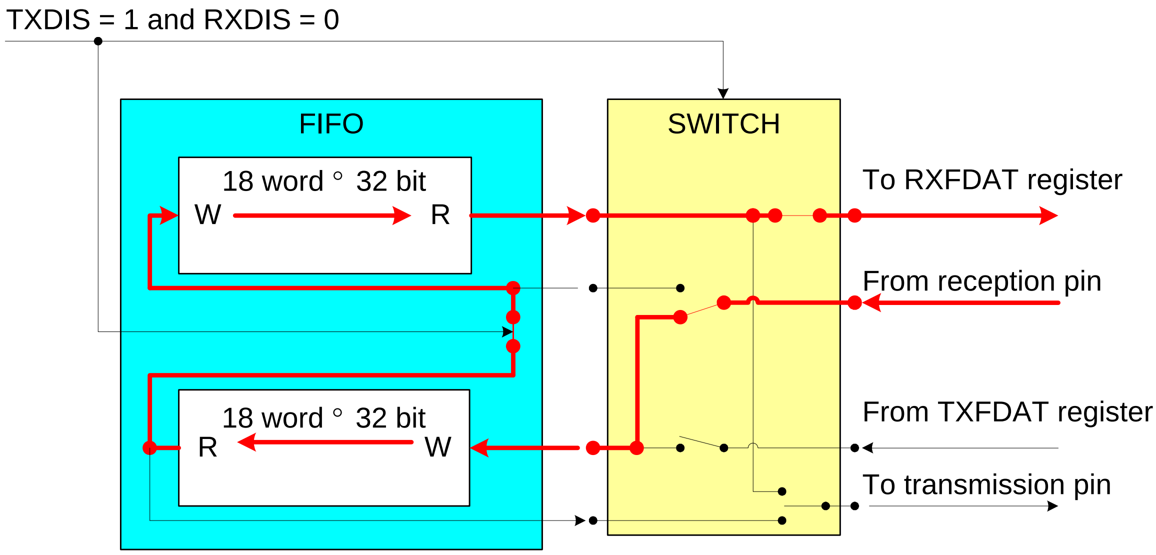


Figure 17-8 Reception only mode data flow

With setting TXDIS = 1 and RXDIS = 0 of CNTREG register, the mode becomes reception only mode which operates in 36 word × 32 bit reception FIFO, and transmission is not performed.

18. UART interface

This chapter describes function and operation of UART.

18.1. Outline

UART is asynchronous transmission/reception serial interface which is controllable by the program. This LSI equips 6 channels of UART.

18.2. Feature

UART has following features:

- Programmable baud rate (baud rate is selectable arbitrarily based on APB clock)
- 16 byte transmission FIFO and 16 byte reception FIFO

18.3. Block diagram

Figure 18-1 shows block diagram of UART.

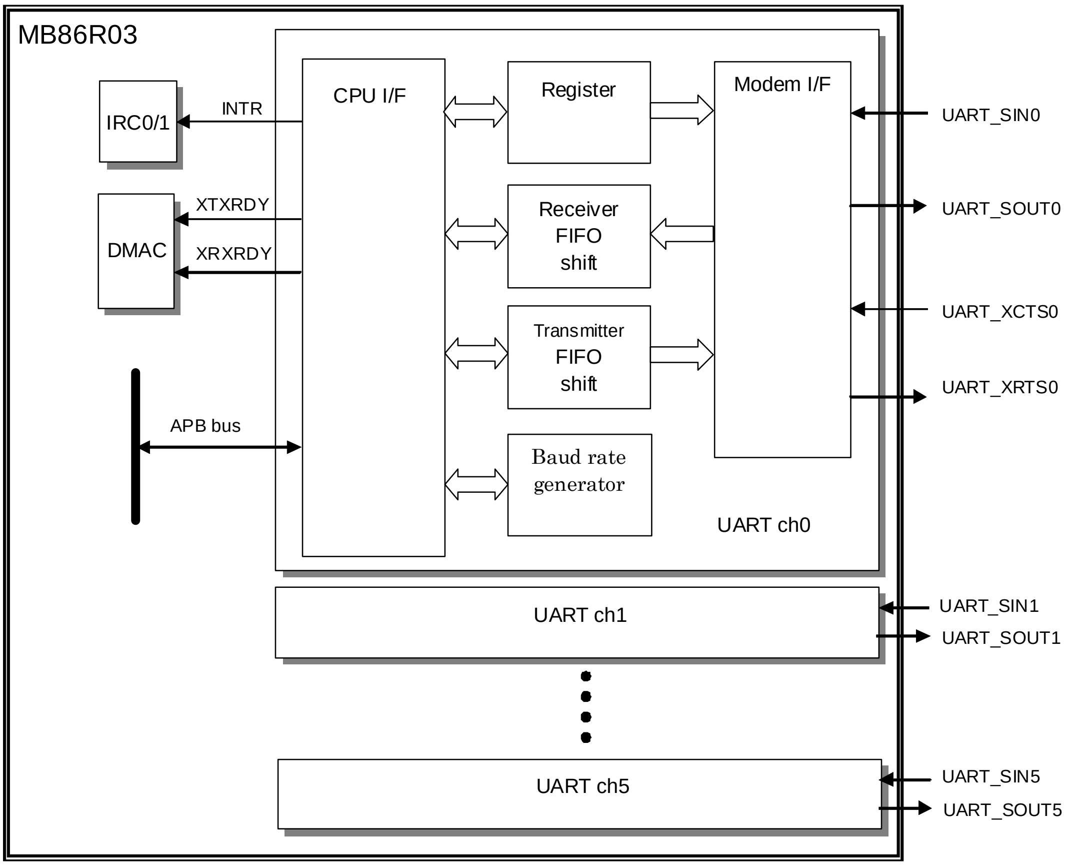


Figure 18-1 Block diagram of UART

18.4. Related pin

UART uses the following pins.

Table 18-1 UART related pin

Pin	Direction	Qty.	Description
UART_SIN0 UART_SIN1 UART_SIN2 UART_SIN3 UART_SIN4 UART_SIN5	IN	6	Input pin of serial data. The umber at the end of pin shows channel number of UART.
UART_SOUT0 UART_SOUT1 UART_SOUT2 UART_SOUT3 UART_SOUT4 UART_SOUT5	OUT	6	Output pin of serial data. The number at the end of pin shows channel number of UART.
UART_XCTS0	IN	1	Input pin of modem control signal, CLEAR TO SEND. Only channel 0 of UART has this pin.
UART_XRTS0	OUT	1	Output pin of modem control signal, REQUEST TO SEND Only channel 0 of UART has this pin.

18.5. Supply clock

APB clock is supplied to UART. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

18.6. Register

This section describes UART interface module's register.

18.6.1. Register list

The LSI has 6 channels (3 dedicated channels and 3 channels of pin multiplex function) of UART interface unit, and each module has the register shown in Table 18-2.

Table 18-2 UART register list

Channel	Address	Register	Description
UART ch0	FFFE1000h	URT0RFR	Reception FIFO register (read only) that is valid in DLAB = 0
		URT0TFR	Transmission FIFO register (write only) that is valid in DLAB = 0
		URT0DLL	Divider latch (low order byte) register that is valid in DLAB = 1
	FFFE1004h	URT0IER	Interrupt enable that is valid in DLAB = 0.
		URT0DLM	Divider latch (high order byte) register that is valid in DLAB = 1
	FFFE1008h	URT0IIR	Interrupt ID register (read only)
		URT0FCR	FIFO control (write only)
	FFFE100Ch	URT0LCR	Line control register
	FFFE1010h	URT0MCR	Modem control register
	FFFE1014h	URT0LSR	Line status register (read only)
	FFFE1018h	URT0MSR	Modem status register (read only)
UART ch1	FFFE2000h	URT1RFR	Reception FIFO register (read only) that is valid in DLAB = 0
		URT1TFR	Transmission FIFO register (write only) that is valid in DLAB = 0
		URT1DLL	Divider latch register (low order byte) that is valid in DLAB = 1
	FFFE2004h	URT1IER	Interrupt enable that is valid in DLAB = 0.
		URT1DLM	Divider latch (high order byte) register that is valid in DLAB = 1
	FFFE2008h	URT1IIR	Interrupt ID register (read only)
		URT1FCR	FIFO control (write only)
	FFFE200Ch	URT1LCR	Line control register
	FFFE2010h	URT1MCR	Modem control register
	FFFE2014h	URT1LSR	Line status register (read only)
	FFFE2018h	URT1MSR	Modem status register (read only)
UART ch2	FFF50000h	URT2RFR	Reception FIFO register (read only) that is valid in DLAB = 0
		URT2TFR	Transmission FIFO register (write only) that is valid in DLAB = 0
		URT2DLL	Divider latch (low order byte) register that is valid in DLAB = 1
	FFF50004h	URT2IER	Interrupt enable that is valid in DLAB = 0.
		URT2DLM	Divider latch (high order byte) register that is valid in DLAB = 1
	FFF50008h	URT2IIR	Interrupt ID register (read only)
		URT2FCR	FIFO control (write only)
	FFF5000Ch	URT2LCR	Line control register
	FFF50010h	URT2MCR	Modem control register
	FFF50014h	URT2LSR	Line status register (read only)
	FFF50018h	URT2MSR	Modem status register (read only)
UART ch3	FFF51000h	URT3RFR	Reception FIFO register (read only) that is valid in DLAB = 0
		URT3TFR	Transmission FIFO register (write only) that is valid in DLAB = 0
		URT3DLL	Divider latch (low order byte) register that is valid in DLAB = 1

Channel	Address	Register	Description
UART ch3	FFF51004h	URT3IER	Interrupt enable that is valid in DLAB = 0.
		URT3DLM	Divider latch (high order byte) register that is valid in DLAB = 1
	FFF51008h	URT3IIR	Interrupt ID register (read only)
		URT3FCR	FIFO control (write only)
	FFF5100Ch	URT3LCR	Line control register
	FFF51010h	URT3MCR	Modem control register
	FFF51014h	URT3LSR	Line status register (read only)
	FFF51018h	URT3MSR	Modem status register (read only)
UART ch4	FFF43000h	URT4RFR	Reception FIFO register (read only) that is valid in DLAB = 0
		URT4TFR	Transmission FIFO register (write only) that is valid in DLAB = 0
		URT4DLL	Divider latch (low order byte) register that is valid in DLAB = 1
	FFF43004h	URT4IER	Interrupt enable that is valid in DLAB = 0.
		URT4DLM	Divider latch (high order byte) register that is valid in DLAB = 1
	FFF43008h	URT4IIR	Interrupt ID register (read only)
		URT4FCR	FIFO control (write only)
	FFF4300Ch	URT4LCR	Line control register
	FFF43010h	URT4MCR	Modem control register
	FFF43014h	URT4LSR	Line status register (read only)
	FFF43018h	URT4MSR	Modem status register (read only)
UART ch5	FFF44000h	URT5RFR	Reception FIFO register (read only) that is valid in DLAB = 0
		URT5TFR	Transmission FIFO register (write only) that is valid in DLAB = 0
		URT5DLL	Divider latch (low order byte) register that is valid in DLAB = 1
	FFF44004h	URT5IER	Interrupt enable that is valid in DLAB = 0.
		URT5DLM	Divider latch (high order byte) register that is valid in DLAB = 1
	FFF44008h	URT5IIR	Interrupt ID register (read only)
		URT5FCR	FIFO control (write only)
	FFF4400Ch	URT5LCR	Line control register
	FFF44010h	URT5MCR	Modem control register
	FFF44014h	URT5LSR	Line status register (read only)
	FFF44018h	URT5MSR	Modem status register (read only)

DLAB: Bit7 of Line control register (LCR)

Note:

Although UART's register length is 8 bit, each register except RFR, TFR, and DLL should be accessed in 32 bit.

PER, TFR, and DLL are able to be accessed in both 32 bit and 8bit lengths; however, note that 8 bit length access is different since register address is endian dependent.

Description format of register

Following format is used for description of register’s each bit in "18.6.2 Reception FIFO register (URTxRFR)" to "18.6.11 Divider latch register (URTxDLL&URTxDLM)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field’s value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

18.6.2. Reception FIFO register (URTxRFR)

Address	ch0 : FFFE_1000 + 00h ch1 : FFFE_2000 + 00h ch2 : FFF5_0000 + 00h ch3 : FFF5_1000 + 00h ch4 : FFF4_3000 + 00h ch5 : FFF4_4000 + 00h (Reading is enabled only at DLAB = 0)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								RFR[7:0]							
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit
7-0	RFR[7:0]	This is FIFO register that is able to maintain up to 16 byte. Reception data is acquired and maintained at the end of reception sequence. This register is able to proceed system reset as well as reset by FCR bit 1 (RxF RST.) RFR register becomes valid at DLAB = 0, and DLL register is assigned at DLAB = 1. RFR register becomes valid only at reading register, and data is written to TFR register (at DLAB = 0) or DLL register (at DLAB = 1) according to the setting value of DLAB when writing.

18.6.3. Transmission FIFO register (URTxTFR)

Address	ch0 : FFFE_1000 + 00h ch1 : FFFE_2000 + 00h ch2 : FFF5_0000 + 00h ch3 : FFF5_1000 + 00h ch4 : FFF4_3000 + 00h ch5 : FFF4_4000 + 00h (Writing is enabled only at DLAB = 0)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								TFR[7:0]							
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit (input "0" at writing)
7:0	TFR[7:0]	This is FIFO register that is able to maintain up to 16 byte. Data is maintained in this register until being transmitted to the Transmission shift register. This register is able to proceed system reset as well as reset by FCR bit 2 (RxF RST.) This register is write only; however, reading operation reads RFR register (at DLAB = 0) or DLL register (at DLAB = 1) according to setting value of DLAB.

18.6.4. Interrupt enable register (URTxIER)

Address	ch0 : FFFE_1000 + 04h ch1 : FFFE_2000 + 04h ch2 : FFF5_0000 + 04h ch3 : FFF5_1000 + 04h ch4 : FFF4_3000 + 04h ch5 : FFF4_4000 + 04h (Accessing is enabled only at DLAB = 0)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)				EDSSI	ELSI	ETBEI	ERBFI
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
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit No.	Bit name	Function
31:4	Unused	Reserved bit (input "0" at writing)
3	EDSSI	Enable Modem Status Interrupt When EDSSI is set to "1" and bit3:0 of the Modem status register is set, interrupt occurs.
2	ELSI	Enable Receiver Status Interrupt When ELSI is set to "1" and bit4:1 of the Line status register is set, interrupt occurs.
1	ETBEI	Enable Transmitter FIFO Register Empty Interrupt After ETBEI is set to "1", interrupt occurs when Transfer FIFO register becomes empty.
0	ERBFI	Enable Receiver FIFO Register When ERBFI is set to "1" and reception FIFO reaches to the trigger level, interrupt occurs. (Interrupt also occurs when character time-out occurs.)

Interrupt can be disabled by setting "0" to all bits of bit3:0.
All interrupt factors of the bit set "1" in bit3:0 become valid.

18.6.5. Interrupt ID register (URTxIIR)

Address	ch0 : FFFE_1000 + 08h ch1 : FFFE_2000 + 08h ch2 : FFF5_0000 + 08h ch3 : FFF5_1000 + 08h ch4 : FFF4_3000 + 08h ch5 : FFF4_4000 + 08h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								FIFO ST1	FIFO ST0	(Reserved)		ID2	ID1	ID0	NINT
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	1	1	0	0	0	0	0	1

Bit No.	Bit name	Function
31:8	Unused	Reserved bit (input "0" at writing)
7:6	FIFO1:0	FIFO status Fixed to "11"
5:4		"00"
3:0	ID2:0, NINT	Interrupt setting 0001: No interrupt 0110: Reception line status (1) Top priority 0100: Reception data existed (2) 1100: Time-out (2) 0010: Transmission FIFO is empty (3) 0000: Modem status (4)

* Bit7:0 = C1h, after the reset
* Numerical value in ( ) is priority level

When character time-out interrupt occurs with having received data, ID2:0, NINT is changed from 0100 to 1100.
Interrupt signal (INTR) is cleared by the following operation.

Priority level:
(1) Read Line status register (LSR)
(2) Read reception FIFO
(3) Read Interrupt ID register (IIR) or write to transmission FIFO
(4) Read Modem status register (MSR)

18.6.6. FIFO control register (URTxFCR)

Address	ch0 : FFFE_1000 + 08h ch1 : FFFE_2000 + 08h ch2 : FFF5_0000 + 08h ch3 : FFF5_1000 + 08h ch4 : FFF4_3000 + 08h ch5 : FFF4_4000 + 08h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								RCVR1	RCVR0	(Reserved)		DMA MODE	TxF RST	RxF RST	(Reserved)
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit (input "0" at writing)
7:6	RCVR1:0	Reception FIFO's trigger level 00: 1 byte 01: 4 byte 10: 8 byte 11: 14 byte
5:4	Unused	Reserved bit
3	DMA MODE	DMA transfer mode (mode of XTXRDY and XRXRDY pins) 0: Single transfer mode 1: Demand transfer mode
2	TxF RST	Transmission FIFO reset 1: Reset
1	RxF RST	Reception FIFO reset 1: Reset
0	Unused	Reserved bit

* Bit7:0 = 00h, after reset

18.6.7. Line control register (URTxLCR)

Address	ch0 : FFFE_1000 + 0Ch ch1 : FFFE_2000 + 0Ch ch2 : FFF5_0000 + 0Ch ch3 : FFF5_1000 + 0Ch ch4 : FFF4_3000 + 0Ch ch5 : FFF4_4000 + 0Ch															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								DLAB	SB	SP	EPS	PEN	STB	WLS1	WLS0
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
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit (input "0" at writing)
7	DLAB	Divisor Latch Access Bit (divider latch access bit) 0: Disable Reception FIFO register reads with address 0 Transmission FIFO register writes with address 0 IER register reads and writes with address 1 1: Enable DLL register reads and writes with address 0 DLM register reads and writes with address 1 TST register writes with address 7
6	SB	Set Break (break transmission) 1: The SOUT signal forcibly becomes "0"
5	SP	Stick Parity (fixed parity) 0: Parity bit is determined by EPS and PEN 1: Parity bit is fixed as follows depending on the status of EPS and PEN (checked at transmission, generation, and reception) Parity is "1" at PEN = 1 and EPS = 0 Parity is "0" at PEN = 1 and EPS = 1
4	EPS	Even Parity Select (parity selection) 0: Odd parity 1: Even parity
3	PEN	Parity Enable (parity enable) 0: Parity is not sent nor checked 1: Parity is sent and checked Parity bit is added to end of data area, and stop bit comes last
2	STB	Number of Stop Bit (stop bit length) 0: 1 bit 1: 1.5 bit (data length: 5) 2 bit (data length: 6 ~ 8)
1:0	WLS1:0	Word Length Select (transmission/reception data length) 00: 5 bit 01: 6 bit 10: 7 bit 11: 8 bit

* Bit7:0 = 00h, after reset

18.6.8. Modem control register (URTxMCR)

Address	ch0 : FFFE_1000 + 10h ch1 : FFFE_2000 + 10h ch2 : FFF5_0000 + 10h ch3 : FFF5_1000 + 10h ch4 : FFF4_3000 + 10h ch5 : FFF4_4000 + 10h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)			LOOP	OUT2	OUT1	RTS	DTR
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
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit (input "0" at writing)
7:5	Unused	Reserved bit (input "0" at writing)
4	LOOP	Loop Back Mode (self-diagnostic mode) When loop is set to "1", following is performed. 1. SOUT becomes "1" 2. SIN is separated from input Shift register of reception 3. Transmission shift register output is connected to input of the Reception shift register 4. Modem status is separated (NCTS, NDSR, NDCD, and NRI) 5. Modem control signal is connected to modem status input CTS – RTS DSR – DTR RI – OUT1 DCD – OUT2
3	OUT2	Control signal "1" makes output pin active.
2	OUT1	
1	RTS	
0	DTR	

* Bit7:0 = 00h, after reset

18.6.9. Line status register (URTxLSR)

Address	ch0 : FFFE_1000 + 14h ch1 : FFFE_2000 + 14h ch2 : FFF5_0000 + 14h ch3 : FFF5_1000 + 14h ch4 : FFF4_3000 + 14h ch5 : FFF4_4000 + 14h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								ERRF	TEMT	THRE	BI	FE	PE	OE	DR
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	0	1	1	0	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit
7	ERRF	Error in RCVR FIFO (error in reception FIFO) This bit is set even 1 error of parity, flaming, or break detection is in reception FIFO. If data including error (except the one set ERRF flag) is not in reception FIFO at reading LSR register, this is reset.
6	TEMT	Transmitter Empty (transmission shift register empty) When both Transmission shift register and Transmission FIFO register become empty, TEMT is set to "1".
5	THRE	Transmitter FIFO Register Empty (transmission register empty) When Transmission FIFO register is empty and ready to accept new data, THRE is set to "1". This bit is cleared at sending data to Transmission shift register.
4	BI	Break Interrupt (break reception) This bit is set when SIN is held in "0" more than transmission time (start bit + data bit + parity + stop bit.) BI is reset by CPU reading this register.
3	FE	Framing Error (flaming error) This bit is set when reception data does not have valid stop bit. FE is reset by CPU reading this register.
2	PE	Parity Error (parity error) This bit is set when reception data does not have valid parity bit. PE is reset by CPU reading this register.
1	OE	Overrun Error (overrunning error) This bit is set when reception FIFO is full and receives the next reception data. OE is reset by CPU reading this register.
0	DR	Data Ready (reception data existed) This bit shows 1 byte or more of data is in FIFO. This bit is set when data is in FIFO and reset after reading all data in FIFO.

* Bit7:0 = 60h, after reset

18.6.10. Modem status register (URTxMSR)

Address	ch0 : FFFE_1000 + 18h ch1 : FFFE_2000 + 18h ch2 : FFF5_0000 + 18h ch3 : FFF5_1000 + 18h ch4 : FFF4_3000 + 18h ch5 : FFF4_4000 + 18h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								DCD	RI	DSR	CTS	DDCD	TERI	DDSR	DCTS
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0

Bit No.	Bit name	Function
31:8	Unused	Reserved bit
7	DCD	Data Carrier Detect Loop = 0: Inversed input signal, XDCD is indicated Loop = 1: It is equal to OUT2 of MCR
6	RI	Ring Indicator Loop = 0: Inversed input signal, XRI is indicated Loop = 1: It is equal to OUT1 of MCR
5	DSR	Data Set Ready Loop = 0: Inversed input signal, XDSR is indicated Loop = 1: It is equal to DTR of MCR
4	CTS	Clear To Send Loop = 0: Inversed input signal, XCTS is indicated Loop = 1: It is equal to RTS of MCR
3	DDCD	Delta Data Carrier Detect This bit is set when DCD signal changes after the last reading by CPU. The bit is reset by reading this register.
2	TERI	Traling Edge of Ring Indicator This bit is set when RI signal changes from 1 to 0 after the last reading by CPU. The bit is reset by reading this register.
1	DDSR	Delta Data Set Ready This bit is set when DSR signal changes after the last reading by CPU. The bit is reset by reading this register.
0	DCTS	Delta Clear To Send This bit is set when CTS signal changes after the last reading by CPU. The bit is reset by reading this register.

* Bit7:0 = x0h, after reset

Bit7:4 is monitor bit of external pin

18.6.11. Divider latch register (URTxDLL&URTxDLM)

This register is frequency dividing latch to generate necessary baud rate from clock input.
Frequency diving latch consists of 16 bit, DLM (high order byte) and DLL (low order byte.)

[DLL]

Address	ch0 : FFFE_1000 + 00h ch1 : FFFE_2000 + 00h ch2 : FFF5_0000 + 00h ch3 : FFF5_1000 + 00h ch4 : FFF4_3000 + 00h ch5 : FFF4_4000 + 00h (Accessing is enabled only at DLAB = 1)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								DL[7:0]							
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
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

[DLM]

Address	ch0 : FFFE_1000 + 04h ch1 : FFFE_2000 + 04h ch2 : FFF5_0000 + 04h ch3 : FFF5_1000 + 04h ch4 : FFF4_3000 + 04h ch5 : FFF4_4000 + 04h (Accessing is enabled only at DLAB = 1))															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								DL[15:0]							
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
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

DLL and DLM are read/written when DLAB bit of LCR is set to "1".

- After the reset, DLL and DLM are 00h
- DLL and DLM values are loaded by writing to either DLL or DLM
- Baud rate is settable in the range that DLM and DLL are FFFFh ~ 0001h

To calculate transfer baud rate

Transfer baud rate (bps) = (APB clock frequency (Hz)/Frequency dividing value)/16

Example of frequency dividing value (DLM and DLL values) and baud rate is shown in Table 18-3.

Table 18-3 Example of frequency dividing value (DLM and DLL values) and baud rate

DLL value (decimal) (DLM = 0)	MB86R03 baud rate APB clock = 41.663(MHz) (external input condition: CLK = 33.33MHz, CRIPM[3:0] = 0011)	The other party's baud rate (error range)
2170	1200	1200 (100%)
1085	2400	2400 (100%)
542	4804	4800 (99.9%)
271	9609	9600 (99.9%)
181	14386	14400 (100.1%)
136	19147	19200 (100.3%)
90	28933	28800 (99.5%)
68	38293	38400 (100.3%)
45	57865	57600 (99.5%)
23	113215	115200 (101.8%)

Transmission baud rate on the other party and baud rate used by macro are able to receive data properly within the permissible error range. Out of the range causes reception error.
Baud rate's permissible error range that macro permits is shown below.

104.1% > Macro baud rate (100%) > 95.3%

When baud rate used by macro is within the reception baud rate's permissible error range of the other party, data is able to be received. Out of the range causes error on the other party side.
After the reset (MR = 1), it takes 1/4 bit of time from setting DLL and DLM to enable start bit detection. Although start bit (SIN = 0) is received in the period, proper start bit detection is not performed.

18.7. UART operation

18.7.1. Example of initial setting

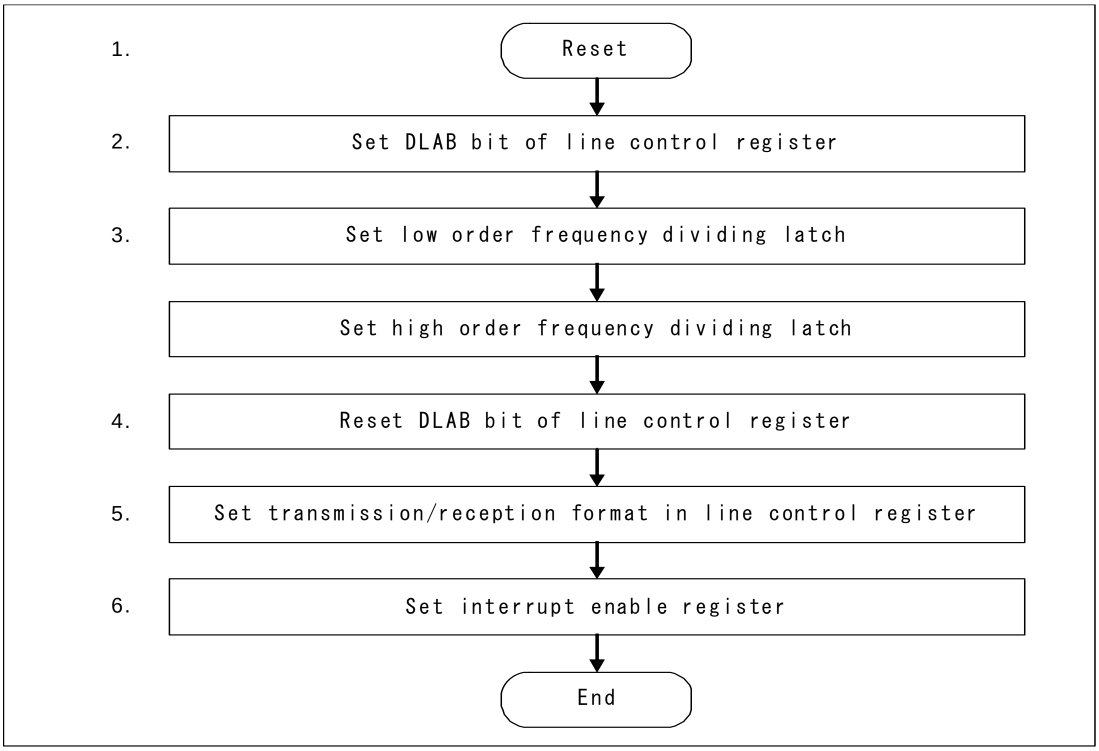


Figure 18-2 Example of initial setting

1. After the power-on, macro's each output pin is undefined. Each output pin level becomes the one shown in the table of chapter 5 by inputting "L" to reset (MR) pin.
2. Divider latch is able to be accessed by setting "1" to DLAB bit in the Line control register (LCR register.)
3. Set baud rate clock (refer to "18.6.11 Divider latch register (URTxDLL&URTxDLM)".)
4. Set "0" to DLAB bit in the Line control register.
5. Set transmission/reception format by setting the Line control register.
6. Control each interrupt by setting the Interrupt enable register (IER register.)

18.7.2. Example of transfer procedure

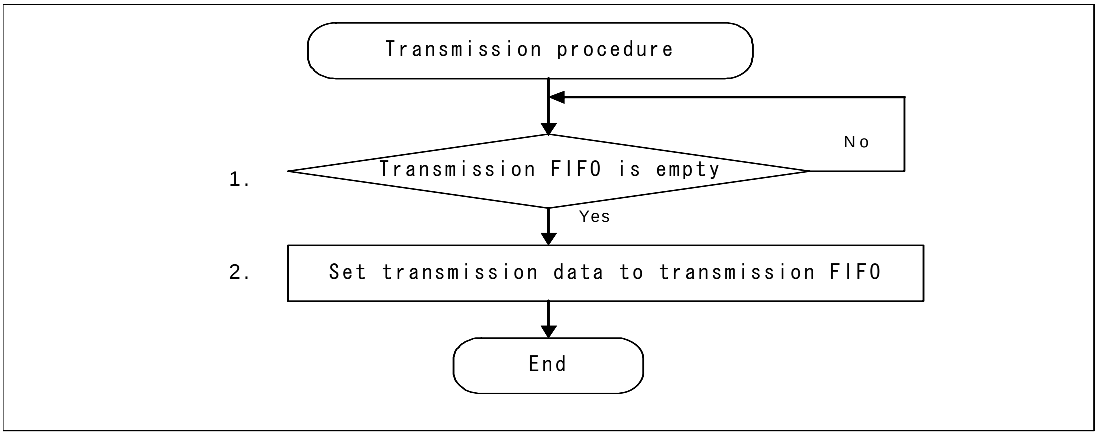


Figure 18-3 Example of transfer procedure

1. Check transmission FIFO is empty with following method:

   a. Polling process of THRE bit in the Line status register (LSR)
   THRE bit shows transmission FIFO status. When the FIFO is empty, the bit becomes "1".

   b. Polling process of TEMT bit in the Line status register (LSR)
   TEMT bit shows transmission FIFO and Transmission shift register statuses that data in transmission process and empty transmission FIFO are able to be confirmed. When they are empty, TEMT becomes "1".

   c. Transmission FIFO empty interrupt process
   When all data in transmission FIFO is moved to the Transmission shift register, this interrupt occurs. It is able to control approval/prohibition in the Interrupt enable register (URTxIER.)

   Note:
   During transmission FIFO empty interrupt process, check THRE bit of the LSR is "1" before writing data to transmission FIFO.
   - THRE = 1: Transmission FIFO is empty that data is able to be written
   - THRE = 0: Transmission FIFO is not empty. Retry from interrupt process to be FIFO empty interrupt status without writing data to transmission FIFO.

2. Set transmission data to transmission FIFO. Up to 16 byte is able to be set in the FIFO at a time.
   In this case, THRE bit of the LSR becomes "0".

   Note:
   The last written data is deleted when writing data to transmission FIFO while it is full.

18.7.3. Example of reception procedure

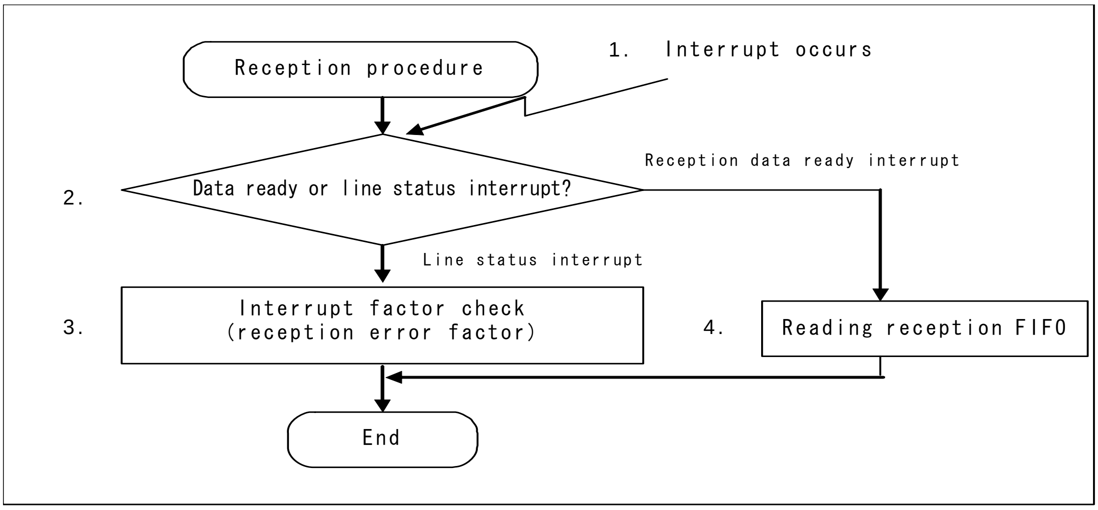


Figure 18-4 Example of reception procedure

1. When certain interrupt is permitted, interrupt occurrence is able to be confirmed with interrupt (INTR) pin (at INTR = "H".)
   Moreover, it is confirmed by polling NINT bit in the Interrupt ID register (IIR register) (at NINT = "0".)
2. Type of interrupt is able to be observed by confirming ID0, ID1 and ID2 bit in the Interrupt ID register.
3. After interrupt type is judged as reception line status interrupt with the process in item 2, reception error information is able to be acquired by reading the Line status register which also releases the interrupt (INTR= "L".)
4. After interrupt type is judged as reception data ready interrupt with the process in item 2, read number of character corresponding to the trigger level to acquire reception character. Reception data ready status is also able to be confirmed by referring DR bit in the Line status register. The interrupt is released when data in FIFO becomes less than the trigger level (INTR= "L".)

18.7.4. Basic transmission operation

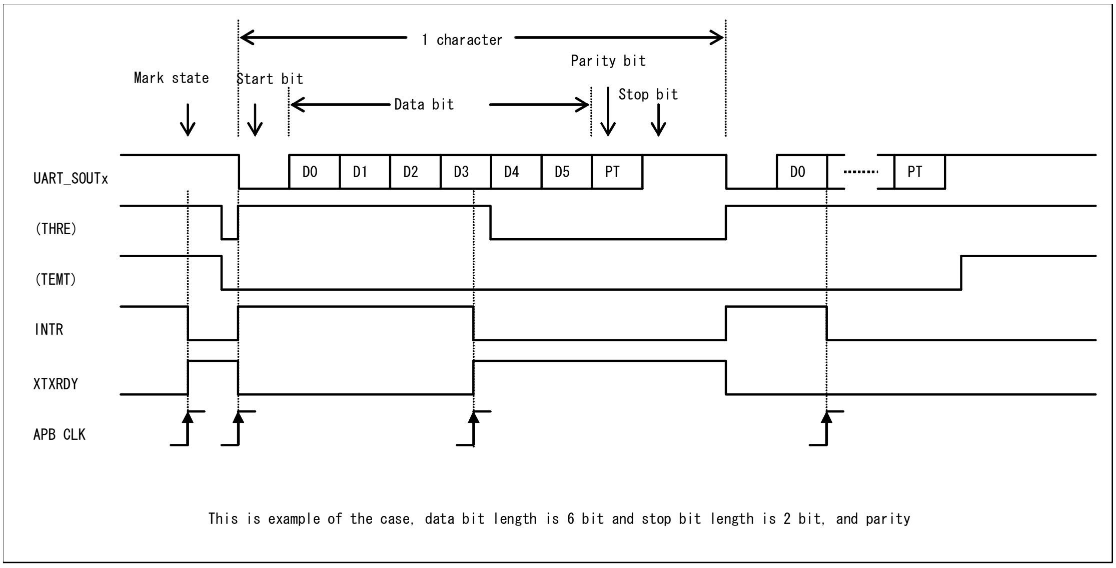


Figure 18-5 Basic transmission operation

When initial reset is completed and transmission data is not written to the Transmission shift register in the transmission control part (mark state), state of "H" level continues applying to serial transmission (SOUT) pin. The data is output from serial transmission (SOUT) pin with adding start bit, parity bit, and stop bit in the transmission control part as shown in Figure 18-5 when transmission data is written from CPU to transmission FIFO.

1 ~ 16 byte of transmission data is able to be consecutively written to transmission FIFO at a time. Transmission FIFO state is able to be confirmed with THRE bit of the LSR register.
When transmission data is written to transmission FIFO though it is full, the last written data is deleted. The data that is already stored in the transmission FIFO is properly transmitted.
THRE bit becomes "0" by writing to transmission FIFO. When the writing data is transferred to the Transmission shift register and FIFO becomes empty, the value becomes "1". If transmission data buffer interrupt is permitted in that time, interrupt (INTR) pin becomes "H" and interrupt occurs. This interrupt is released by writing data to the transmission FIFO again or reading the Interrupt confirmation register.
TEMT bit becomes "0" at the same timing of THRE bit, and the value becomes "1" after transmission of all written data is completed.
XTXRDY is data ready signal that shows possible transmission to DMA controller at using the controller. Single transfer mode is supported when bit 3 of the FCR register is "0" and the demand transfer mode is supported when the bit is "1".

When transmission baud rate used by macro is within the reception baud rate permissible error range, the other party is able to receive data. Out of the range causes reception error on the other party side.

18.7.5. Basic reception operation

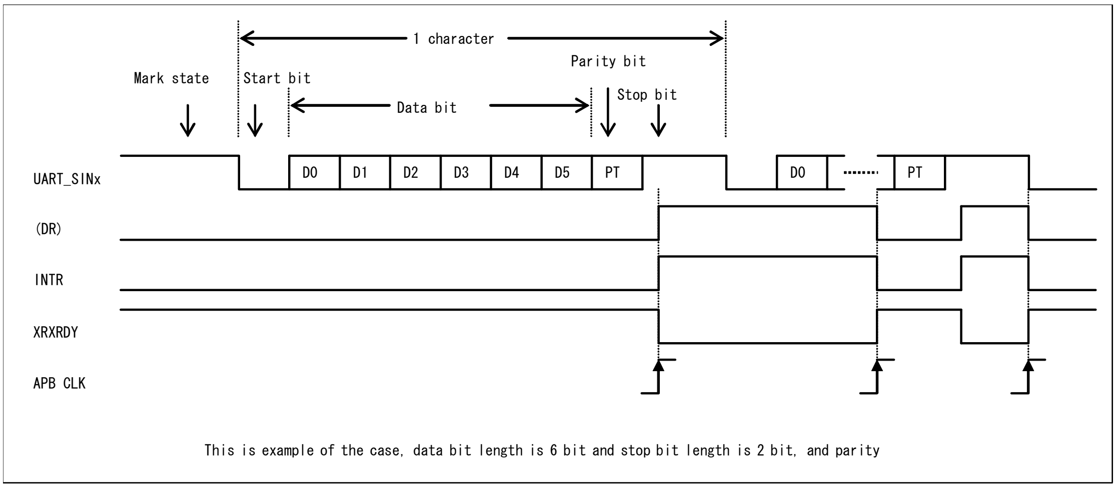


Figure 18-6 Basic reception operation

After detecting received start bit ("L" level) from serial input (SIN) pin, the bit receiving next is regarded as start bit of reception data.
Then, received data is sampled with reception clock, and stop bit is detected after receiving data bit and parity bit. When transmission error occurs during that time, its factor (break detection, flaming error, parity error, and overrunning error) is applied to each data in FIFO, and the status is maintained. Status can be confirmed by CPU at the first data of FIFO.
When reception data ready interrupt is permitted, interrupt (INTR) pin becomes "H" and interrupt occurs by reaching the data in reception FIFO to the trigger level. This interrupt is released when the data in the FIFO becomes less than the trigger level, and interrupt (INTR) pin becomes "L".

XRXRDY is data ready signal that shows possible reception to DMA controller at using the controller. Single transfer mode is supported when bit 3 of the FCR register is "0" and the demand transfer mode is supported when the bit is "1".
When transmission baud rate of the other party and baud rate used by macro are within the reception baud rate permissible error range, data is able to be received properly. Out of the range causes reception error. Baud rate permissible error range that macro permits is as follows.

104.1% > Macro baud rate (100%) > 95.3%

After reset (MR = 1), the time reaching to enable detection of start bit is 1/4 bit after DLL and DLM are set. Even if start bit (SIN=0) is received during this period, normal start bit detection is not performed.

18.7.6. Line status

THRE flag and TEMT flag

Operation example of THRE flag and TEMT flag of bit 5 and 6 in the Line status register (LSR) is shown in Figure 18-7.

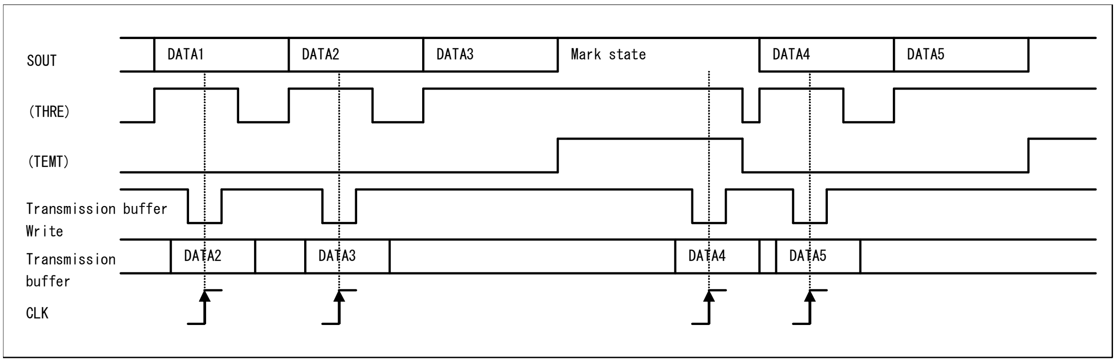


Figure 18-7 Example of operation of THRE flag and TEMT flag

THRE flag = "1" indicates that there is no data in the Transmission FIFO buffer register, and transmission character is able to be written.
TEMT flag becomes "1" when there is no data in the register and Transmission shift register in the transmission control part.
Both flags become "0" at writing "0" to transmission FIFO buffer.

FE flag and BI flag

Operation example of BI flag and of bit 4 and 3 and FE flag in the Line status register (LSR) is shown in Figure 18-8.

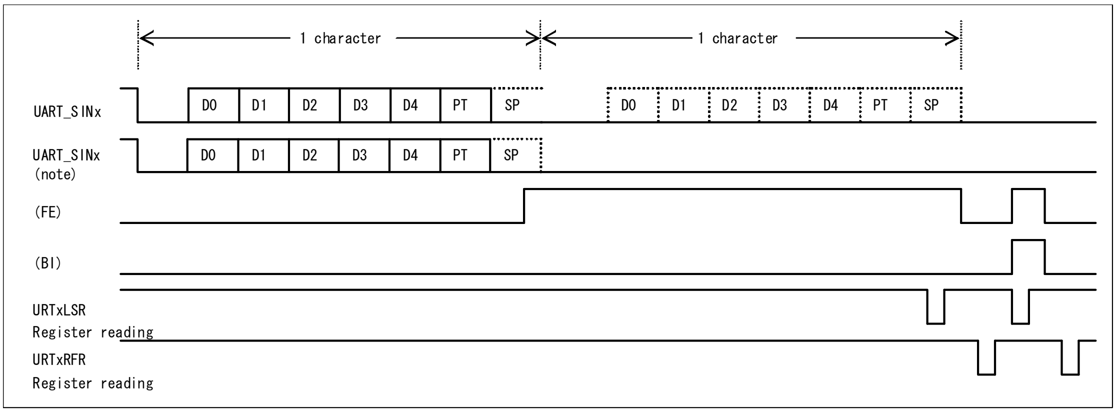


Figure 18-8 Operation example of FE flag and BI flag

If "L" level is received at the stop bit during reception operation, flaming error occurs and FE flag becomes "1". The error flag is reset by reading Line status register.

When "L" level continues during transmission time (start bit, data bit, parity bit, and stop bit) for 1 character, break code is detected. These errors are applied to each data in FIFO, and they are able to be confirmed when CPU reads the first data of FIFO. FE and BI flags are able to be confirmed in the Status register at reading Line status register whose first data includes framing and break detection error. Both flags become "0" by reading Status register.

For the case of break detection error, reception data is stored to FIFO as 0.

When break is detected, macro stops reception, and it restarts the process with detecting SIN's falling edge.

PE flag

Operation example of PE flag of bit 2 in the Line status register (LSR) is shown in Figure 18-9.

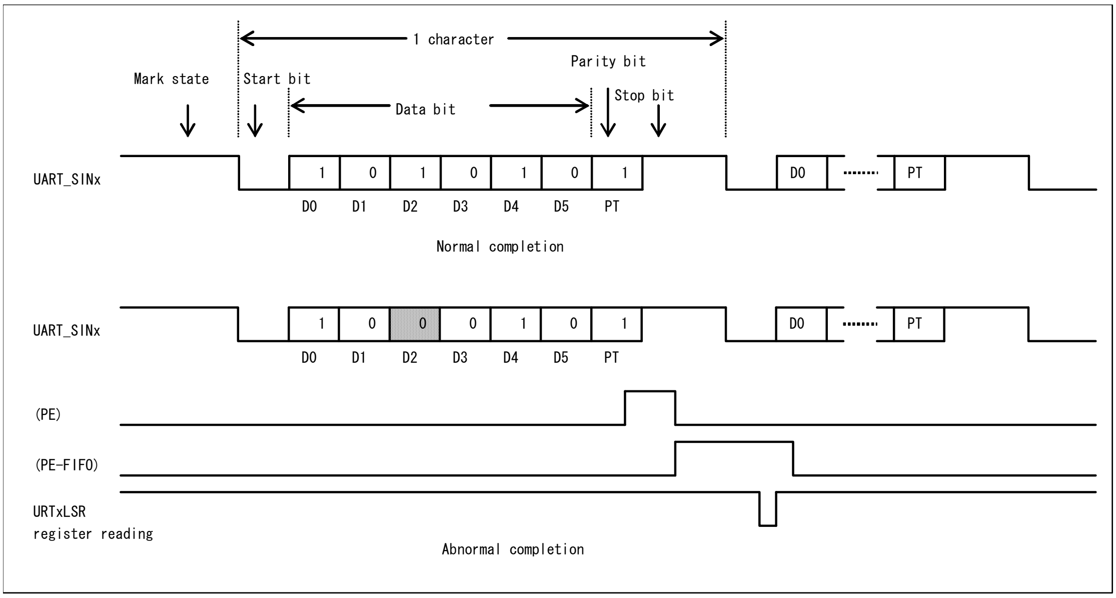


Figure 18-9 Operation example of PE flag (setting even parity)

Parity bit is set to "1" or "0" depending on the number of "1" level bit in the 1 data bit. When it is set to even parity with EPS in the Line control register, the bit is set to "1" or "0" to have total data bit and "1" level parity bit even number. Likewise, when parity bit is set to odd parity, total number of "1" level is set to be odd number.
On reception side, the number of "1" level bit of 1 data including input parity bit is counted, and polarity of the parity set with EPS bit in the Line control register is compared.
For their discrepancy, PE flag of the register becomes "1" by the judgment that problem occurred in transmitting data. Then the flag becomes "0" by reading the Line status register. This error is applied to each data in FIFO, and is able to be confirmed when CPU reads first data of FIFO.

OE flag

Operation example of OE flag of bit 1 in the Line status register (LSR) is shown in Figure 18-10.

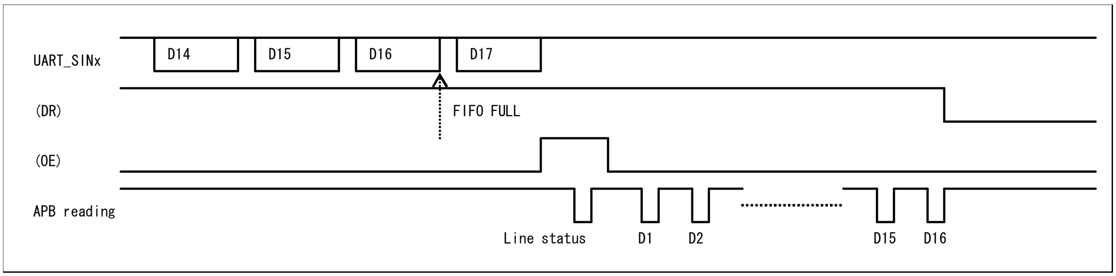


Figure 18-10 Operation example of OE flag

When next character is received completely to the Reception shift register in the status that reception FIFO is full, overrun error occurs. In this case, OE flag of the Line status register is set immediately and interrupt occurs (if it is permitted.)

DR flag

Operation example of DR flag of bit 0 in the Line status register (LSR) is shown in Figure 18-11.

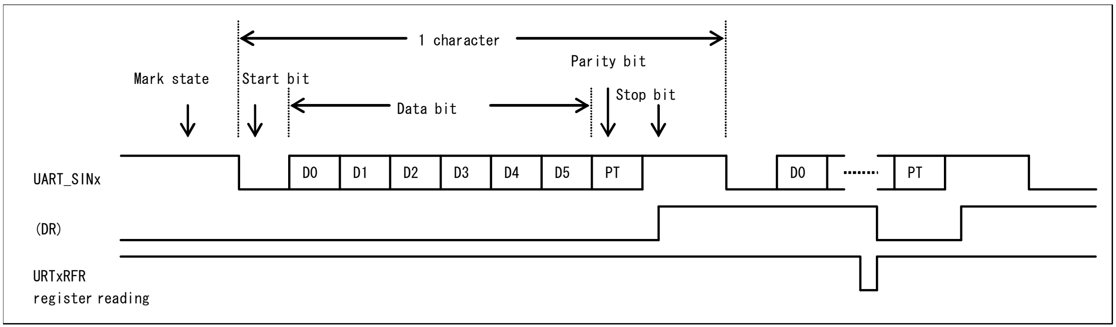


Figure 18-11 Operation example of DR flag

When reception data is received and 1 byte or more of data is stored in reception FIFO, DR flag of the Line status register becomes "1". The flag becomes "0" by reading reception FIFO data and FIFO becomes empty.

ERRF flag

When error (parity, break detection, and flaming) is included in the data stored in reception FIFO, ERRF flag of bit 7 of the Line status register (LSR) is set to "1" during reception operation.

If there is no error data in FIFO except the one set ERRF flag when CPU reads the register, this flag is cleared to "0".

18.7.7. Character time-out interrupt

Character time-out interrupt occurs in the following cases:

- 1 or more data is stored in reception FIFO and the next serial data is still not received after 4 characters of time
- 1 or more data is stored in reception FIFO and CPU still does not read the data after 4 characters of time

When time-out interrupt occurs, INTR pin becomes "H". Moreover, XRXRDY signal becomes "L", showing DMA controller that reception is ready, and requests to read data.
Timer and time-out interrupt are reset by CPU (or DMA controller) reading 1 byte from reception FIFO. If time-out does not occur, it is reset after timer receives new data or CPU (or DMA controller) reads data from reception FIFO.

19. I^{2}C bus interface

This chapter describes function and operation of I^{2}C bus interface.

19.1. Outline

I^{2}C bus is serial bus advocated by Philips Semiconductors (now NXP) that supports data between multiple devices with 2 signals. MB86R03 equips 2 channels of interface corresponding to I^{2}C standard mode (max. 100Kbps)/high-speed mode (max. 400KBps.) External pin, I2C_SDA0, I2C_SDA1, I2C_SCL0, and I2C_SCL1 uses 3.3V exclusive I/O, so that it is able to be used in 3.3V I^{2}C.
I2C_SDA0/I2C_SDA1 are indicated as SDA line, and I2C_SCL0/I2C_SCL1 are indicated as SCL line in this document.

19.2. Feature

I^{2}C has following features:

- Master transmission/reception function
- Slave transmission/reception function
- Arbitration function
- Clock synchronization function
- Slave address detecting function
- General call address detecting function
- Transfer direction detecting function
- Repeat occurrence and detecting function of start condition
- Bus error detecting function
- Corresponding to standard mode (max. 100KBps)/high-speed mode (max. 400KBps)

19.3. Block diagram

Figure 19-1 shows block diagram of I^2C.

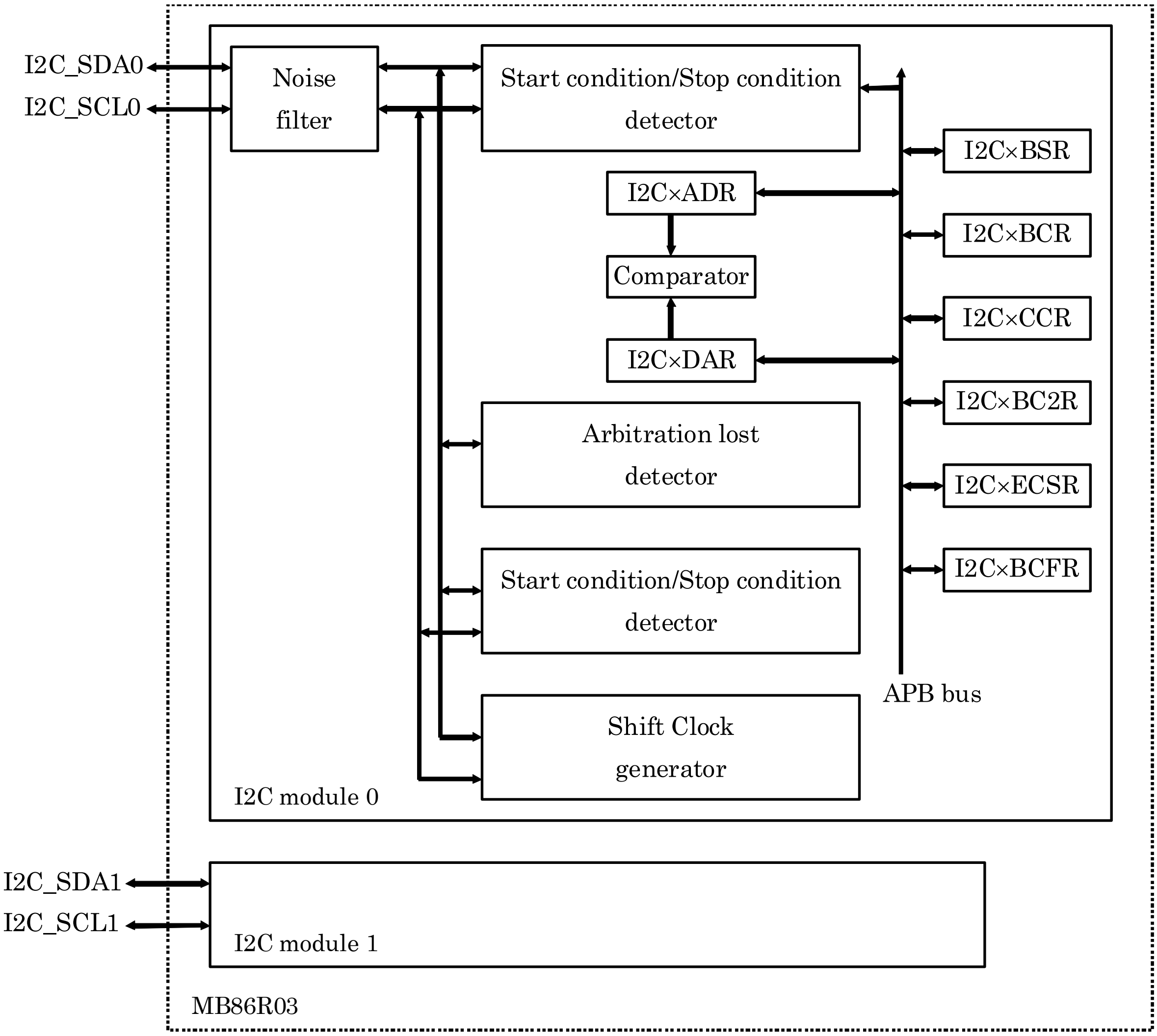


Figure 19-1 Block diagram of I^2C

Block function

Each block function is described below.

Table 19-1 I²C block function

Block	Description
Start condition/Stop condition detector	Start condition and Stop condition are detected from transition state of SDA and SCL lines.
Start condition/Stop condition generator	Start condition and Stop condition are issued from transition state of SDA and SCL lines.
Arbitration lost detector	Output data to SDA line and input data from SDA line are compared at data transmission. If they are unmatched, arbitration lost occurs.
Shift clock generator	Timing count of serial data t transfer clock occurrence and output control of SCL line clock are performed with clock control register setting.
Comparator	Received address and self-address specified to address register, or received address and global address are compared.
I2CxADR	7 bit register that specifies slave address.
I2CxDAR	8 bit register used for serial data transfer.
I2CxBSR	8 bit register with following functions to show I²C bus status and others. • Repeated start condition detection • Arbitration lost detection • Acknowledge bit storage • Direction of data transfer • Addressing detection • General call address detection • First byte detection
I2CxBCR	8 bit register that performs I²C bus control and interrupt control has following functions. • Interrupt request/permission • Start condition occurrence • Master/Slave selection • Acknowledge occurrence permission
I2CxCCR	7 bit register that sets clock frequency of serial data transfer. • Operation permission • Frequency setting of serial clock • Standard/High-speed mode selection
Noise filter	This is noise filter composed of 3 stage shift register circuit. When all 3 values consecutively sampled SCL/SDA line input signals are "1", the filter output becomes "1". When those values are "0", the filter output becomes "0". For other sampling, the state 1 clock before is maintained.
I2CxBC2R	This is the register to drive "L" forcibly and to confirm the line status after noise filter is passed.
I2CxECSR	This is the register to enhance CS bit in I2CxCCR register.
I2CxBCFR	This is the register that specifies frequency range of bus clock to be used.

19.4. Related pin

I^{2}C uses following pins.

Table 19-2 I^{2}C related pin

Pin	Direction	Qty.	Description
I2C_SCL0 I2C_SCL1	IN/OUT	2	Clock pin of I^{2}C bus interface. The last number of the pin name indicates channel number of I^{2}C. Output of this pin is open drain.
I2C_SDA0 I2C_SDA1	IN/OUT	2	Data pin of I^{2}C bus interface. The last number of the pin name indicates channel number of I^{2}C. Output of this pin is open drain.

19.5. Supply clock

APB clock is supplied to I^{2}C. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

19.6. Register

This section describes I^2C bus interface register.

19.6.1. Register list

This LSI equips 2 channels of I^2C bus interface, and each module has the register shown in Table 19-3.

Table 19-3 I^2C register list

Channel	Address	Register	Description
I^2C ch0	FFF56000h	I2C0BSR	Bus status register
	FFF56004h	I2C0BCR	Bus control register
	FFF56008h	I2C0CCR	Clock control register
	FFF5600Ch	I2C0ADR	Address register
	FFF56010h	I2C0DAR	Data register
	FFF56014h	I2C0ECSR	Extension CS register
	FFF56018h	I2C0BCFR	Bus clock frequency register
	FFF5601Ch	I2C0BC2R	Bus control 2 register
I^2C ch1	FFF57000h	I2C1BSR	Bus status register
	FFF57004h	I2C1BCR	Bus control register
	FFF57008h	I2C1CCR	Clock control register
	FFF5700Ch	I2C1ADR	Address register
	FFF57010h	I2C1DAR	Data register
	FFF57014h	I2C1ECSR	Extension CS register
	FFF57018h	I2C1BCFR	Bus clock frequency register
	FFF5701Ch	I2C1BC2R	Bus control 2 register

Note:

Access the area of I^2C ch0 and I^2C ch1 in 32 bit (word)

Description format of register

Following format is used for description of register's each bit in "19.6.2 Bus status register (I2CxBSR)" to "19.6.9 Bus clock frequency register (I2CxBCFR)".

Address	Base address + Offset address															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

19.6.2. Bus status register (I2CxBSR)

Address	ch0 : FFF5_6000 + 00h ch1 : FFF5_7000 + 00h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								BB	RSC	AL	LRB	TRX	AAS	GCA	FBT
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

All bit of this register is cleared during EN bit of I2CxCCR is "0".

Bit 7: BB (bus busy)

This bit shows I^2C bus state.

BB	Status
0	Stop condition is detected
1	Start condition is detected (but is in use)

Bit 6: RSC (Repeated Start Condition)

Repeated start condition detecting bit.

RSC	State
0	Repeated start condition is not detected
1	Start condition is detected again during bus is in use

This bit is cleared by writing "0" to INT bit, start condition detection at bus stop, and stop condition detection as well as addressing is not performed at slave.

Bit 5: AL (Arbitration Lost)

Arbitration lost detecting bit

AL	State
0	Arbitration lost is not detected
1	Arbitration lost occurs during master transmission, or "1" is written to MSS bit while other systems are using bus

This bit is cleared by writing "0" to INT bit.

Restrictions:

In the multi master environment, prohibit other masters to transmit general call address simultaneously with this module, as well as use of arbitration lost by this module at the second byte or later.

Bit 4: LRB (LAST Received Bit)

This bit is to store 9th bit of the data indicating acknowledge (ACK)/negative acknowledge (NACK).

LRB	State
0	Acknowledge (ACK) is detected
1	Negative acknowledge (NACK) is detected

This bit is cleared at start condition detection or stop condition detection.

Bit 3: TRX (Transfer/Receive)

This bit is to indicate transmission/reception state of data transfer.

TRX	State
0	Reception state
1	Transmission state

Bit 2: AAS (Address As Slave)

This is addressing detection bit.

AAS	State
0	Addressing is not performed at slave
1	Addressing is performed at slave

This is cleared at start condition detection or stop condition detection.

Bit 1: GCA (General Call Address)

This is general call address (00h) detecting bit.

GCA	State
0	General call address is not received at slave
1	General call address is received at slave

This bit is cleared at start condition detection or stop condition detection.

Bit 0: FBT (First Byte Transfer)

This is first byte detecting bit.

FBT	State
0	Reception data is not first byte
1	Reception data is the first byte (address data)

Although this is set to "1" at start condition detection, it is cleared if "0" is written to INT bit and addressing is not performed at the salve.

19.6.3. Bus control register (I2CxBCR)

Address	ch0 : FFF5_6000 + 04h ch1 : FFF5_7000 + 04h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								BER	BEIE	SCC	MSS	ACK	GCAA	INTE	INT
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

This is cleared during EN bit of I2CxCCR is "0", except bit 7 and 6 of this register.

Bit 7: BER (Bus ERror)

This is bus error interrupt request flag bit.

At writing

BER	State
0	Bus error interrupt request flag is cleared
1	N/A

At reading

BER	State
0	Bus error is not detected
1	Incorrect start and stop conditions are detected during data transfer

When this bit is set, EN bit of I2CxCCR resister is cleared, this module becomes in halt state, and the data transfer is discontinued.

Bit 6: BEIE (Bus Error Interrupt Enable)

This is buss error interrupt permission bit.

At reading/writing

BEIE	State
0	Bus error interrupt is prohibited
1	Bus error interrupt is permitted

When this bit is "1" and BER bit is "1", interrupt occurs.

Bit 5: SCC (Start Condition Continue)

This is start condition generation bit.

At writing

SCC	State
0	N/A
1	Start condition is generated again at master transfer

This bit is automatically cleared after setting "1".

Bit 4: MSS (Master Slave Select)

This is master/slave selection bit.

At writing

MSS	State
0	Stop condition is generated, and state becomes slave mode after the transfer
1	State becomes master mode, and start condition is generated to start transfer

This bit is cleared when arbitration lost occurs during master transmission, and state becomes slave mode.

Restrictions:

In the multi master environment, prohibit other masters to transmit general call address simultaneously with this module and to use arbitration lost by this module at the second byte or later.

Bit 3: ACK (ACKnowledge)

This is acknowledge permission bit at receiving data.

At reading/writing

ACK	State
0	Acknowledge is not occurred.
1	Acknowledge is occurred.

This bit is disabled at address data reception in the slave mode.

Bit 2: GCAA (General Call Address Acknowledge)

This is acknowledge permission bit at receiving general call address.

At reading/writing

GCAA	State
0	Acknowledge is not occurred.
1	Acknowledge is occurred.

Bit 1: INTE (INTerrupt Enable)

This is interrupt permission bit.

At reading/writing

INTE	State
0	Interrupt is prohibited
1	Interrupt is enabled

When this bit is "1" and INT bit is "1", interrupt occurs.

Bit 0: INT (INTerrupt)

This is transfer end interrupt request flag bit.

At writing

INT	State
0	Transfer end interrupt flag is cleared
1	N/A

At reading

INT	State
0	Transfer is not completed
1	This is set when following conditions are applied at completion of 1 byte transfer which includes acknowledge bit. • Bus master • Addressed slave • General call address is received (only at GCAA ＝ "1") • Arbitration lost occurs (only at bus acquisition state) • Start condition is attempted while other systems use bus

When this bit is "1", SCL line is maintained in "L" level. This is cleared by writing "0" to this bit, then SCL line opens and the next byte is transferred. Moreover, this is cleared to "0" by occurrence of start condition or stop condition at the master mode.

Competition of SCC, MSS, and INT bits

Competition of the next byte transfer, start condition, and stop condition occurs by writing SCC, MSS, and INT bits simultaneously. Priority order in this case is as follows.

1. Occurrence of the next byte transfer and stop condition
   When writing "0" to INT bit and MSS bit simultaneously, MSS bit is prioritized and stop condition occurs.

2. Occurrence of the next byte transfer and start condition
   When writing "0" to INT bit and "1" to SCC bit simultaneously, SCC bit is prioritized and start condition occurs.

3. Occurrence of start condition and stop condition
   Writing "1" to SCC bit and "0" to MSS bit simultaneously is prohibited.

19.6.4. Clock control register (I2CxCCR)

Address	ch0 : FFF5_6000 + 08h ch1 : FFF5_7000 + 08h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)	HSM	EN	CS[4:0]				
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	0	0	X	X	X	X	X

Bit 7: Unused

The value is always "1" at reading.

Bit 6: HSM (High Speed Mode)

This is standard/high-speed setting bit.

At reading/writing

HSM	State
0	Standard mode
1	High-speed mode

Bit 5: EN (ENable)

This is operation permission bit.

At reading/writing

EN	State
0	Operation is prohibited
1	Operation is permitted

When this bit is "0", each bit of I2CxBSR register and I2CxBCR register (excluding BER and BEIE bits) is cleared. When BER bit is set, this bit is cleared.

Bit 4-0: CS4-0 (Clock Period Select 4-0)

This bit is to set frequency of serial transfer clock.

Upper bound of the bus clock frequency is able to be extended by setting I2CxECSR register. Refer to "19.6.8 Expansion CS register (I2CxECSR)" for details.

When I2CxECSR register is not used (using I2CxECSR register in initial state), frequency fscl of serial transfer clock becomes the expression shown below.

At standard mode

$$fscl = \frac{\phi}{(2 \times m) + 2} \qquad \phi : APB\_clock$$

At high-speed mode

$$fscl = \frac{\phi}{\mathrm{int}(1.5 \times m) + 2} \qquad \phi : APB\_clock$$

$$\mathrm{int}(\,) : Round\ off\ \ after\ decimal\ \ point$$

Be sure to set fscl not to exceed the following values at the master operation.

- At standard mode: 100KHz.
- At high-speed mode: 400KHz.

APB clock φ of this module should be used within the range shown below.

When it is less than the range, transmission by max. transfer rate is not guaranteed.

When it exceeds the range, upper bound of the bus clock frequency is able to be extended by setting I2CxECSR register.

- At the master operation: 14MHz ~ 18MHz.
- At the slave operation: 14MHz ~ 18MHz.
- At the register access operation: 14MHz ~ 41.5MHz

Note:

+2 cycle is min. overhead for checking output level change of SCL line. When rising edge delay of SCL line is large or the clock is enlarged with slave device, the value is lager than the above.

The value of m to CS4 ~ 0 is shown in the next page

CS4	CS3	CS2	CS1	CS0	m	
					Standard	High speed
0	0	0	0	0	65	Setting prohibited
0	0	0	0	1	66	Setting prohibited
0	0	0	1	0	67	Setting prohibited
0	0	0	1	1	68	Setting prohibited
0	0	1	0	0	69	Setting prohibited
0	0	1	0	1	70	Setting prohibited
0	0	1	1	0	71	Setting prohibited
0	0	1	1	1	72	Setting prohibited
0	1	0	0	0	73	9
0	1	0	0	1	74	10
0	1	0	1	0	75	11
0	1	0	1	1	76	12
0	1	1	0	0	77	13
0	1	1	0	1	78	14
0	1	1	1	0	79	15
0	1	1	1	1	80	16
1	0	0	0	0	81	17
1	0	0	0	1	82	18
1	0	0	1	0	83	19
1	0	0	1	1	84	20
1	0	1	0	0	85	21
1	0	1	0	1	86	22
1	0	1	1	0	87	23
1	0	1	1	1	88	24
1	1	0	0	0	89	25
1	1	0	0	1	90	26
1	1	0	1	0	91	27
1	1	0	1	1	92	28
1	1	1	0	0	93	29
1	1	1	0	1	94	30
1	1	1	1	0	95	31
1	1	1	1	1	96	32

19.6.5. Address register (I2CxADR)

Address	ch0 : FFF5_6000 + 0Ch ch1 : FFF5_7000 + 0Ch															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)	A[6:0]						
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	1	X	X	X	X	X	X	X

Bit 7: Unused

The value is always "1" at reading.

Bit 6-0: A6-0 (Address 6-0)

This is slave address storage bit.

The comparison with I2CxDAR register is performed after address data reception at slave.
If they are matched, acknowledge is transmitted to master.

19.6.6. Data register (I2CxDAR)

Address	ch0 : FFF5_6000 + 10h　ch1 : FFF5_7000 + 10h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								D[7:0]							
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	X	X	X	X	X	X	X	X

Bit 7-0: D7-0 (Data 7-0)

This is serial data storage bit.

This data register is used for serial transfer transmitted from MSB. When data is received (TRX = 0), the data output becomes "1".

This register's writing side is double buffer that writing data is loaded to serial transfer register at transmission of each byte if bus (BB = 1) is in use.

Since serial transfer register is directly read at reading, received data is valid only when INT bit is set.

19.6.7. Two bus control registers (I2CxBC2R)

Address	ch0 : FFF5_6000 + 1Ch ch1 : FFF5_7000 + 1Ch															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)		SDAS	SCLS	(Reserved)		SDAL	SCLL
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	X	X	0	0	0	0

Bit 7 and 6: Unused

The value is always "00" at reading.

Bit 5: SDAS (SDA status)

Signal level of SDA line after passed noise filter is indicated.
Only reading is valid.

SDAS	State
0	The SDA line is "0"
1	The SDA line is "1"

Bit 4: SCLS (SCL status)

Signal level of SCL line after passed noise filter is indicated.
Only reading is valid.

SCLS	State
0	SCL line is "0"
1	SCL line is "1"

Bit 3 and 2: Unused

The value is always "00" at reading.

Bit 1: SDAL (SDA low drive)

SDAO output is forcibly become "L".
Both reading/writing are valid.

SDAL	State
0	SDAL output is in normal operation
1	SDAL output is forcibly become "L"

Bit 0: SCLL (SCL Low drive)

SCLO output is forcibly become "L".
Both reading/writing are valid.

SCLL	State
0	SCLO output is in normal operation
1	SCLO output is forcibly become "L"

19.6.8. Expansion CS register (I2CxECSR)

Address	ch0 : FFF5_6000 + 14h ch1 : FFF5_7000 + 14h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)		CS[10:5]					
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit 5-0: CS10-5 (Clock Period Select 10-5)

This is set to expand upper bound of bus clock frequency with extending CS4 ~ 0 in the I2CxCCR register.

Initial value of CS10 ~ 5 is "000000", and setting other values goes into frequency upper bound expansion mode.

CS10~5	State
000000	No upper bound expansion of bus clock frequency (only CS4 ~ 0 is used)
Other than 000000	There is upper bound expansion of bus clock frequency

Standard mode:

$$fscl = \frac{\phi}{(2 \times m) + 2}$$

ϕ : APB_clock

m : (Value of CS10～0)+1

High-speed mode:

$$fscl = \frac{\phi}{\mathrm{int}(1.5 \times m) + 2}$$

ϕ : $APBclock$

m : (Value of CS10～0)+1

int( ) : Round off after decimal point

Set fscl not to exceed the following values at master operation.

- Standard mode: 100kHz
- High-speed mode: 400kHz

Use system clock ϕ of this module within the range shown below.
When it is less than the range, transfer in max. transfer rate is not guaranteed.
When it exceeds the range, the operation is not guaranteed.

- Master operation: 14MHz ~ 41.5MHz
- Slave operation: 14MHz ~ 41.5MHz
- Register access operation: 14MHz ~ 41.5MHz

Note:

+2 cycle is min. overhead for checking output level change of SCL line. When rising edge delay of SCL pin is large or the clock is enlarged with slave device, the value is lager than the above.

When extension CS register is used, m value becomes CS10 ~ 0 + 1.

19.6.9. Bus clock frequency register (I2CxBCFR)

Address	ch0 : FFF5_6000 + 18h ch1 : FFF5_7000 + 18h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								(Reserved)				FS[3:0]			
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit 7 and 4: Unused

The value is always "0000" at reading.

Bit 3-0: FS3-0 (Bus Clock Frequency Select 3-0)

Select frequency of the bus clock to be used. Characteristics such as noise filters are set with this register's setting. A standard setting value is shown below; however, adjustment might be required depending on I^2C buffer characteristics and noise state on I^2C bus.

FS3	FS2	FS1	FS0	Frequency [MHz]
0	0	0	0	Setting prohibited
0	0	0	1	14 or more ~ Less than 20
0	0	1	0	20 or more ~ Less than 40
0	0	1	1	40 or more ~ Less than 60
0	1	0	0	–
0	1	0	1	–
0	1	1	0	–
0	1	1	1	–
1	0	0	0	–
1	0	0	1	–
1	0	1	0	–
1	0	1	1	–
1	1	0	0	–
1	1	0	1	–
1	1	1	0	–
1	1	1	1	–

19.7. Operation

I^{2}C bus communicates with 2 interactive bus lines, serial data line (SDA) and serial clock line (SCL.)
This module is connected to SDA and SCL lines through open drain IO cell by wired logic.

19.7.1. Start condition

When "1" is written to MSS bit with bus open (BB = 0), this module becomes master mode, and start condition occurs at the same time. In the master mode, the start condition can be occurred again by writing "1" to SCC bit even if the bus is in use (BB = 1).
There are 2 ways of condition to engender start condition.

1. Writing "1" to MSS bit in status (MSS = 0 & BB = 0 & INT = 0 & AL = 0) that bus is not used
2. Writing "1" to SCC bit in interrupt status (MSS = 1 & BB = 1 & INT = 1 & AL = 0) at bus master

When "1" is written to MSS bit at idling, AL bit is set to "1". Writing "1" to MSS bit and SCC bit in other states than the above is ignored.

Start condition on I^{2}C bus

Changing SDA line from "1" to "0" while SCL line is "1" is called start condition.

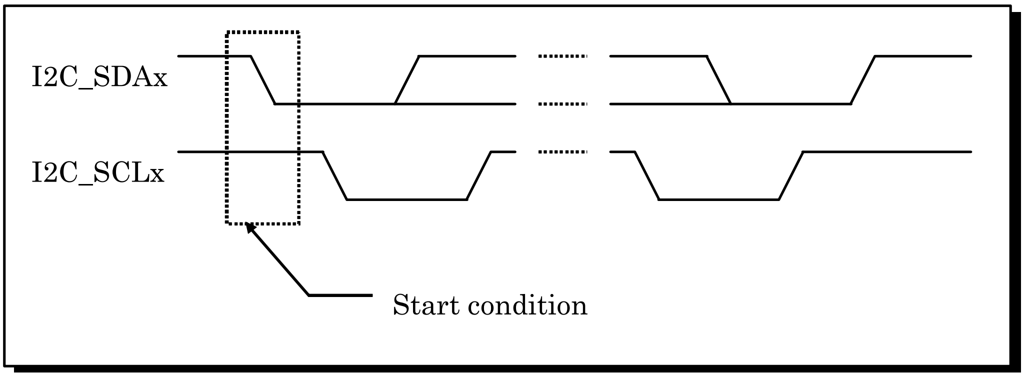


Figure 19-2 Start condition on I^{2}C bus

19.7.2. Stop condition

When "0" is written to MSS bit at master operation (MSS = 1), stop condition occurs and mode becomes slave. Following is condition to engender stop condition.

1. Writing "0" to MSS bit in interrupt status (MSS = 1 & BB = 1 & INT = 1 & AL = 0) at bus master

Writing "1" to MSS bit in other states than the above is ignored.

Stop condition on I2C bus

Changing SDA line from "0" to "1" while SCL line is "1" is called stop condition.

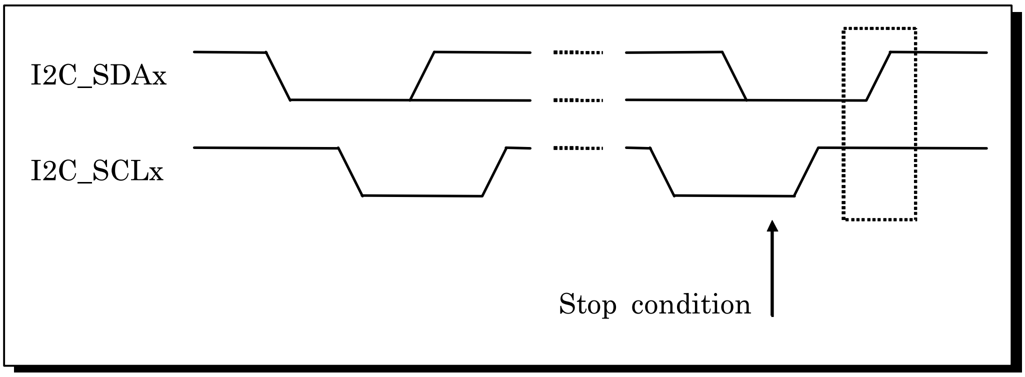


Figure 19-3 Stop condition on I2C bus

19.7.3. Addressing

In the master mode, status is set to BB = "1" and TRX = "1" after start condition occurs, and contents of I2CxDAR register is output from MSB. When acknowledge is received from the slave after sending address data, bit 0 of its data (I2CxDAR register's bit 0 after transmission) is reversed and stored to TRX bit.

In the salve mode, status is set to BB = "1" and TRX = "0" after start condition occurs, and transmission data from the master is received to I2CxDAR register. After receiving address data, I2CxDAR register and I2CxADR register are compared. When they are matched, status is set to AAS = "1" and acknowledge is sent to the master, then bit 0 of the reception data (I2CxDAR register's bit 0 after reception) is stored to TRX bit.

Transfer format of slave address

Transfer format of the slave address is shown below.

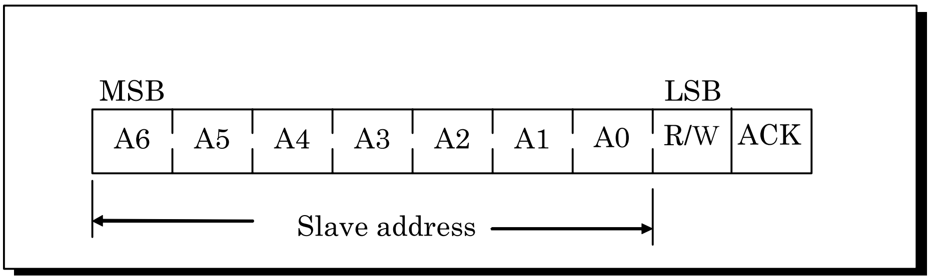


Figure 19-4 Slave address's transfer format

Map of slave address

Slave address map is shown below.

Slave address	R/W	Description
0000 000	0	General call address
0000 000	1	Start byte
0000 001	X	CBUS address
0000 010	X	Reserved
0000 011	X	Reserved
0000 1XX	X	
0001 XXX 1110 XXX	X	Available slave address
1111 0XX	X	10 bit slave address (*1)
1111 1XX	X	Reserved
*1: This module does not support 10 bit slave address		

19.7.4. Synchronous arbitration of SCL

When multiple I^{2}C devices become master device almost the same time to operate SCL line, each device detects SCL line status and automatically adjusts the line's operation timing with keeping the pace to slow device.

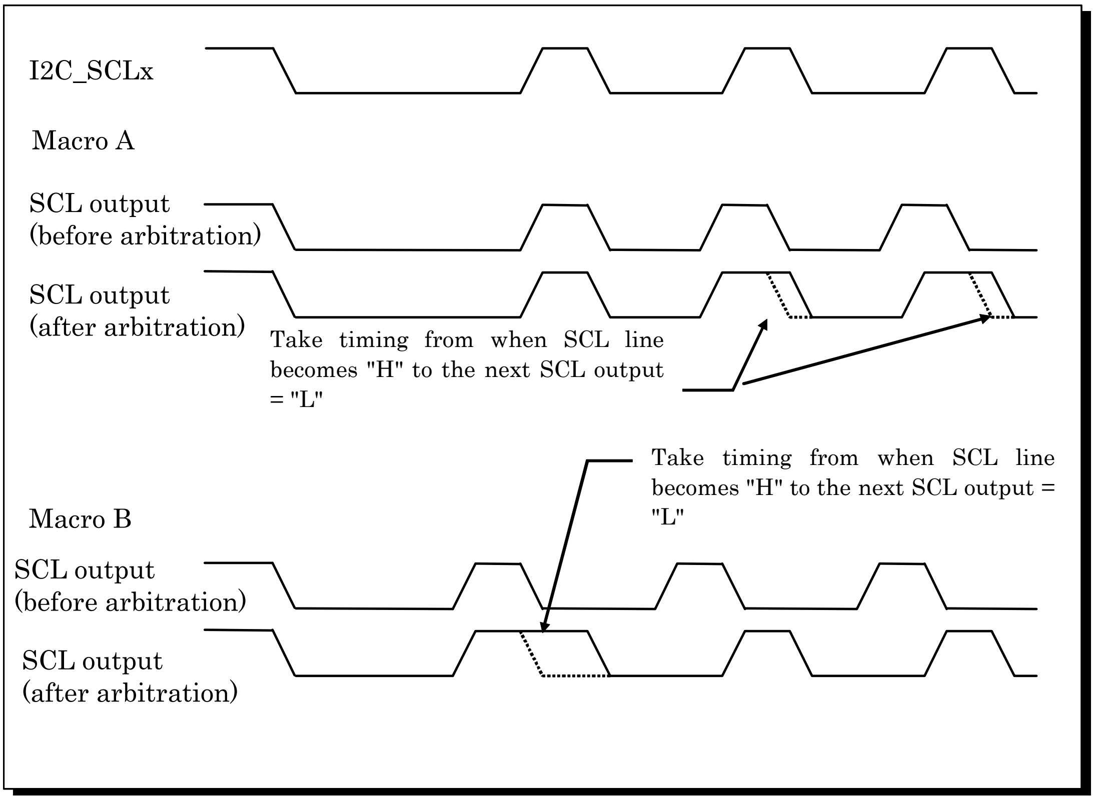


Figure 19-5 SCL output's synchronous arbitration

19.7.5. Arbitration

Arbitration occurs when other masters also transmit data at the same time.

- When own transfer data is "1" and data on SDA line is "0", AL = "1" is set regarding that arbitration is lost.
- When start condition is attempted during other masters are using bus, AL = "1" is set regarding that arbitration is lost.
- When other masters' start condition is detected before starting condition occurs though unused bus is confirmed and MSS = "1" is set, AL = "1" is set regarding that arbitration is lost.

When AL bit is set to "1", status becomes MSS = "0" and TRX = "0" that state becomes slave reception mode. When arbitration is lost (the right to use the bus is lost.), master discontinues drive of SDA. However, drive of SCL is not discontinued until 1 byte of transmission ends and the interrupt is cleared.

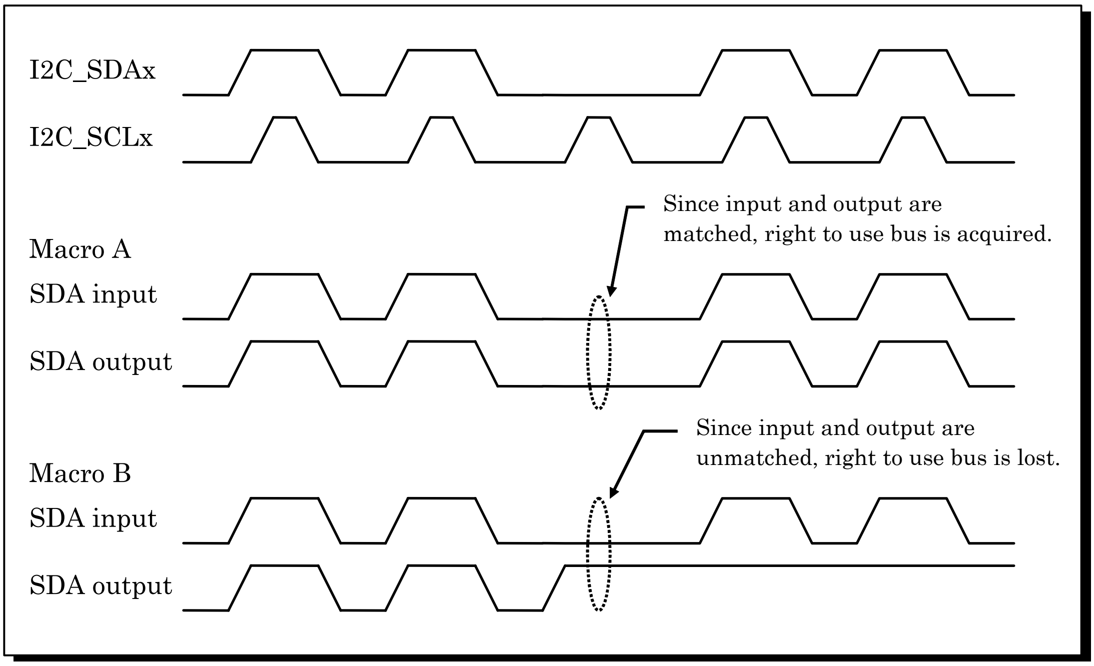


Figure 19-6 Arbitration

19.7.6. Acknowledge/Negative acknowledge

9th bit of data shows acknowledge (ACK)/negative acknowledge (NACK), status of "0" is acknowledge and "1" is negative acknowledge.
The reception side transmits acknowledge/negative acknowledge to transmission side, and they are stored to LRB bit at data reception.
If acknowledge is not received from master reception side at slave transmission (when negative acknowledge is received), the state becomes TRX = "0" and mode becomes slave reception mode. As a result, master is able to generate stop condition when slave opens SCL.

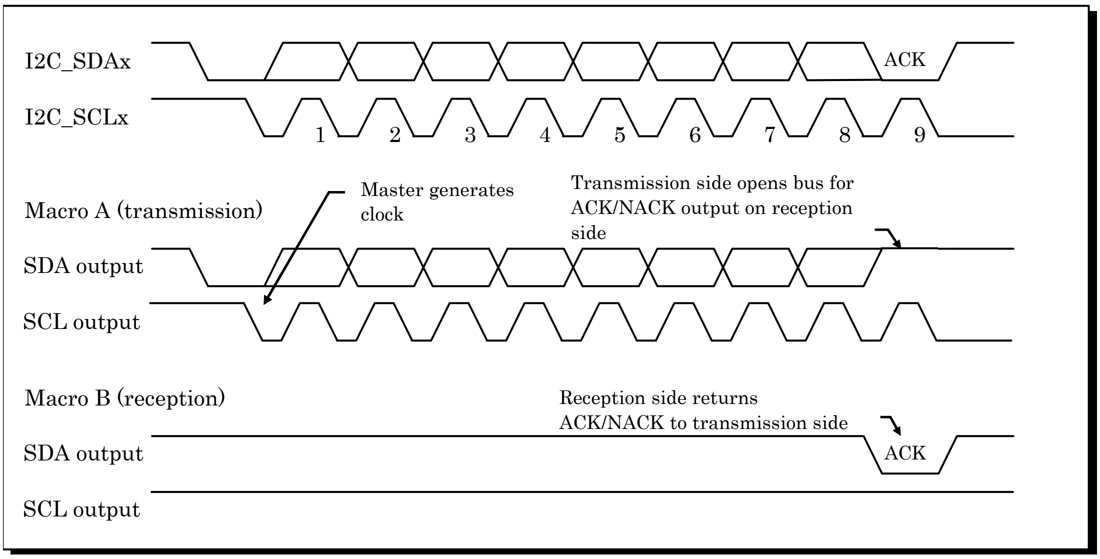


Figure 19-7 Acknowledge/Negative acknowledge

19.7.7. Bus error

When following conditions meet, state is judged as bus error and this module stops.

a. Detection of basic rule violation on I^2C bus in data transmission (including ACK bit)

b. Detection of stop condition at master

c. Detection of basic rule violation on I^2C bus at bus idle

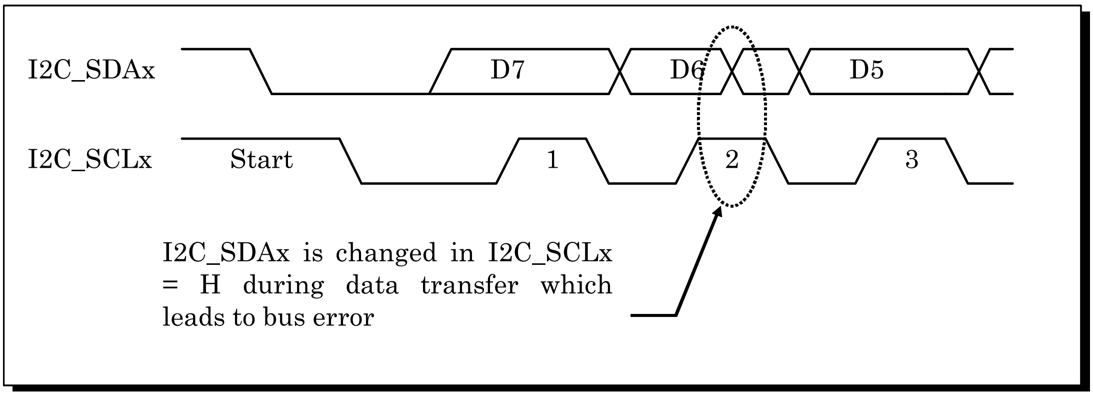


Figure 19-8 Bus error

19.7.8. Initialization

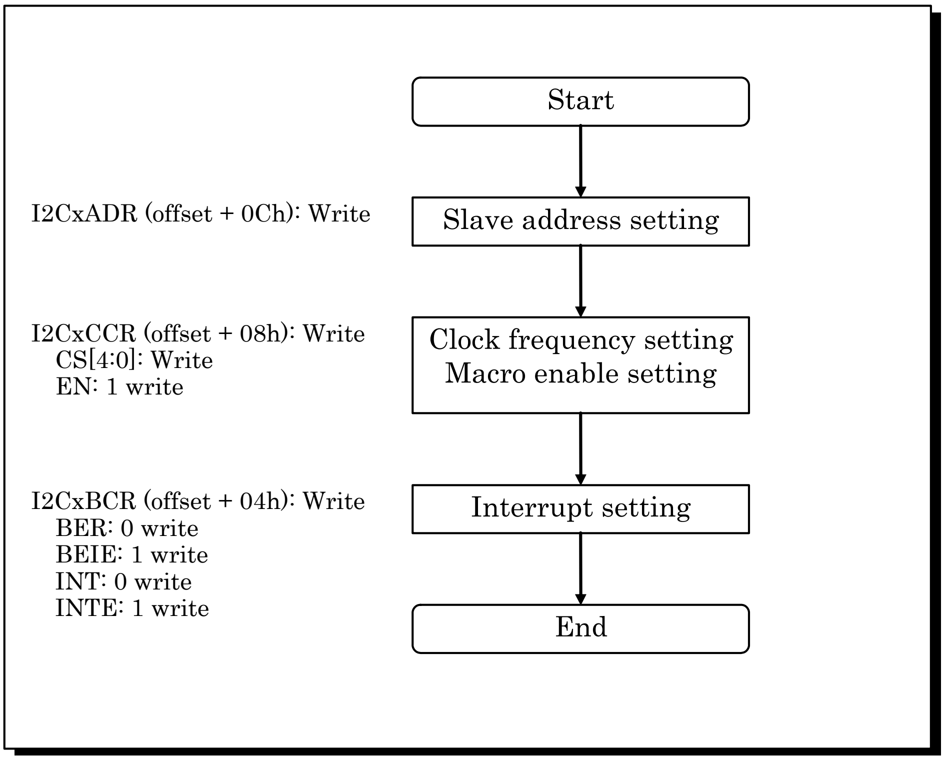


Figure 19-9 I²C initialization

19.7.9. One byte transfer from master to slave

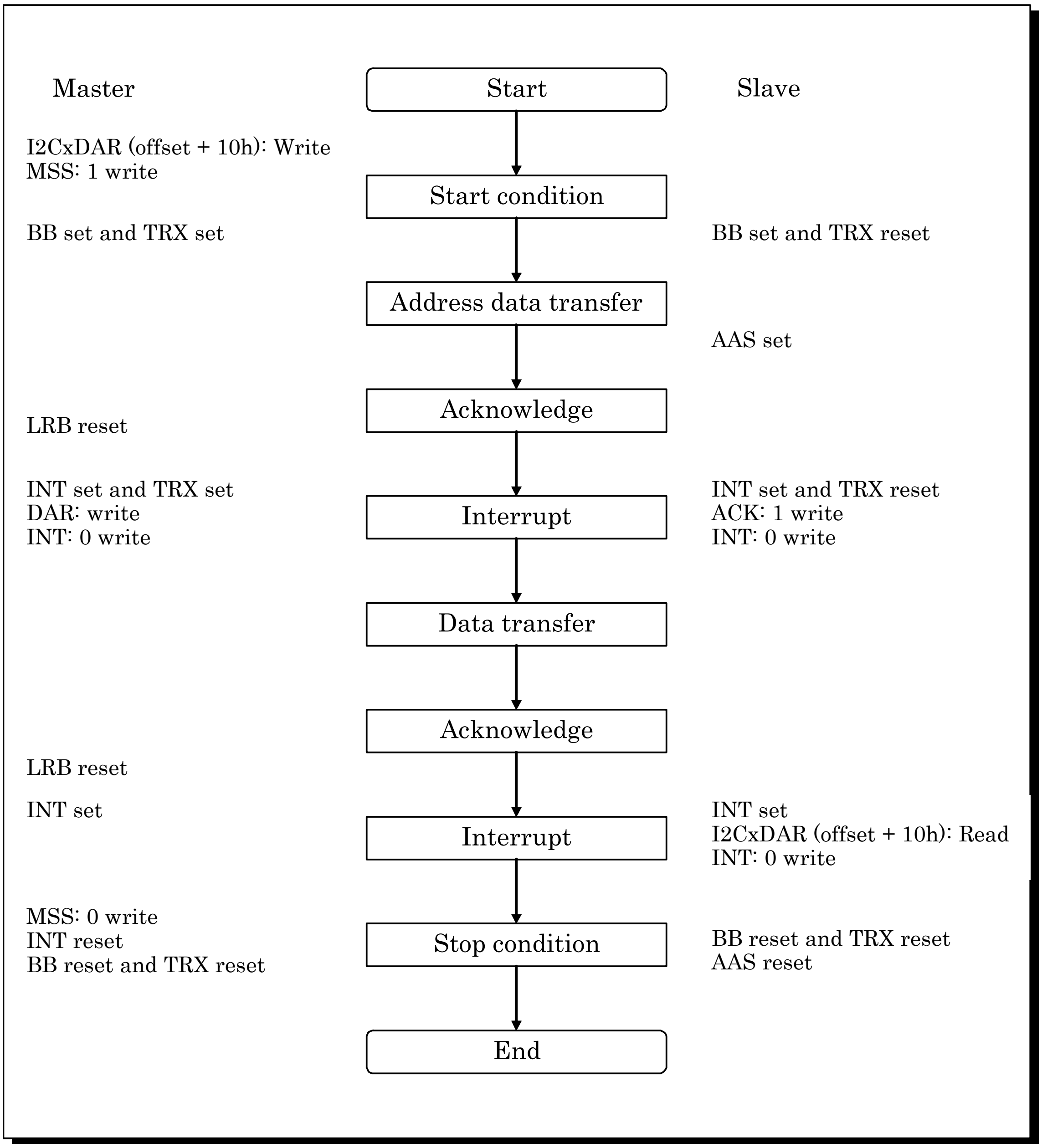


Figure 19-10 1 byte transfer example from master to slave

19.7.10. One byte transfer from slave to master

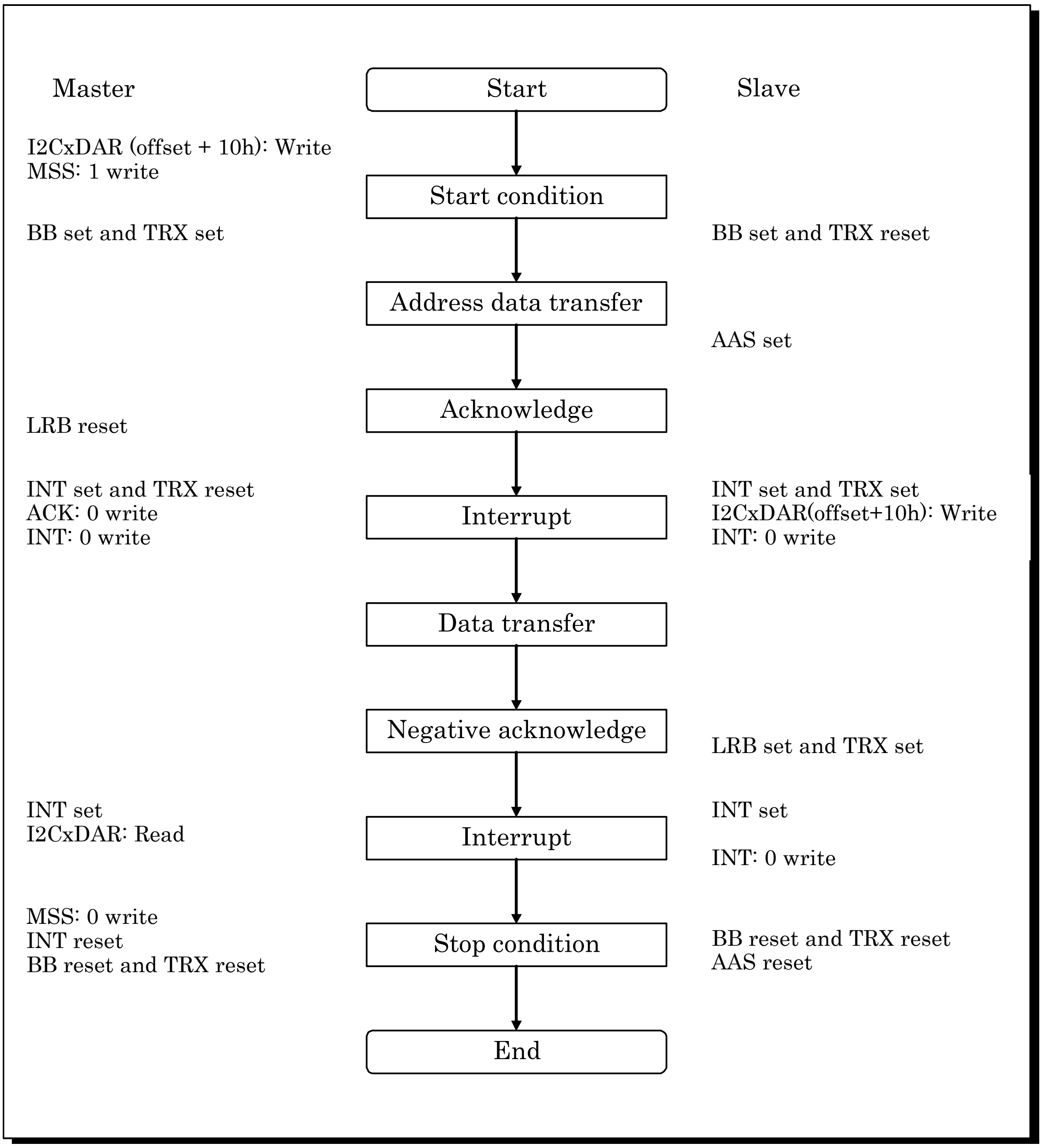


Figure 19-11 1 byte transfer example from slave to master

19.7.11. Recover from bus error

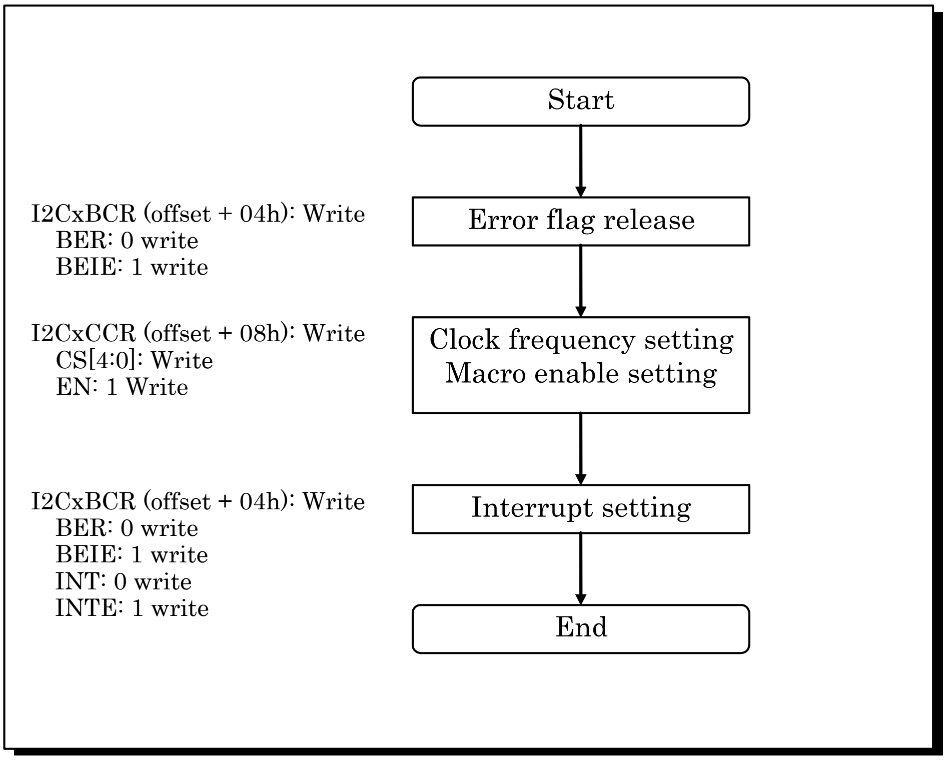


Figure 19-12 Setting example for recovering from bus error

19.7.12. Interrupt process and wait request operation to master device

When INT flag of I2CxBCR register is "H" (during this module engenders interrupt and CPU proceeds interrupt operation), "L" is output to SCL line. While slave side sets "L" to SCL line, master side is unable to generate the next transfer so that slave side puts wait on master side.

19.8. Notice

System clock and fscl of this module

Supply system clock to this module within the following range. The communication with system clock of 18MHz or more needs I2CxCSR setting.

- Master operation: 14MHz ~ 41.5MHz
  Set I2CxCCR not to exceed the following limits on fscl. If it exceeds the upper bound of each mode, normal transfer is not proceeded since it is timing violation on I^2C bus.
  Standard: 100kHz
  High-speed: 400kHz
- Slave operation: 14MHz ~ 41.5MHz
- Register access: 14MHz ~ 41.5MHz

10 bit slave address

This module does not support 10 bit slave address; therefore, do not specify slave address from 78H to 7bH for the module. When wrong address is specified, acknowledge is returned at receiving 1byte; however, normal transfer is not proceeded.

Competition of SCC, MSS, and INT bit

Simultaneous writing of SCC, MSS, and INT bits causes competition of start and stop conditions at the next byte transfer. The priority of this case is as follows.

1. Occurrence of the next byte transfer and stop condition
   When "0" is written to INT bit and MSS bit simultaneously, MSS bit is prioritized and stop condition occurs.
2. Occurrence of the next byte transfer and start condition
   When "0" is written to INT bit and "1" is written to SCC bit simultaneously, SCC bit is prioritized and start condition occurs.
3. Occurrence of start condition and stop condition
   Writing "1" to SCC bit and "0" to MSS bit simultaneously is prohibited.

Serial transfer clock setting

When rising edge delay of SCL line is large or clock is expanded at the slave device, the value may be smaller than the setting value (calculated value) since overhead occurs.

Restrictions in global call address transmission at using multi master

When this module is used at multi master, it is prohibited that other masters send global call address at the same time of this module and it loses arbitration at the 2nd byte or later.

Following usage does not fall under this restriction.

- This module is used in the single master environment.
- This module is used in the multi mater environment; however, it does not send general call address.
- This module is used in the multi master environment; however, other modules do not use general call address transmission.
- Although this module is used in the multi master environment and other masters send general call address simultaneously with this module, it does not lose arbitration at the 2nd byte or later.*

*: Because the larger transmission data causes arbitration lost, the data of the 2nd byte or later must always be smaller than the value of other masters' data.

20. Serial peripheral interface (SPI)

This chapter describes function and operation of serial imperial interface (SPI.)

20.1. Outline

SPI is a serial interface to perform synchronous communication.

20.2. Feature

SPI has following features:

- Serial synchronous transmission of the full duplex
- Transfer format is settable to programmable
  - a) Bit rate
  - b) Data length (1 ~ 32 bit)
  - c) Clock polarity
  - d) Phase
- Supporting 2 types of slave select signals
- Only 1 slave is connectable

Example of SPI connection

Figure 20-1 shows SPI connection example.

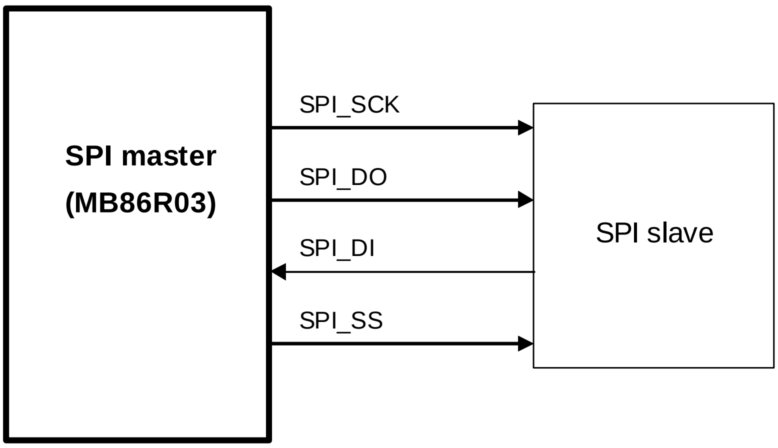


Figure 20-1 Example of SPI connection

Note:

When slave is active, SPI_DI pin may be floating.

20.3. Block diagram

Figure 20-2 shows block diagram of SPI.

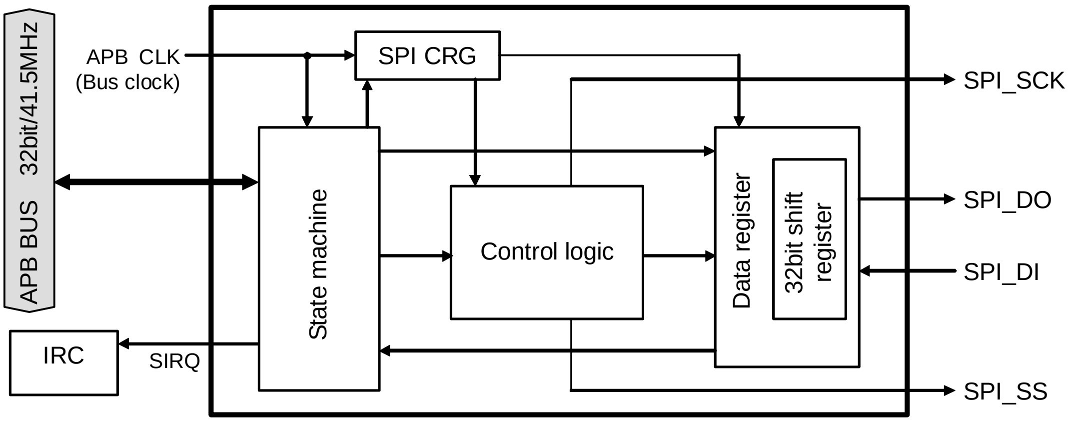


Figure 20-2 Block diagram of SPI

20.4. Supply clock

APB clock is supplied to SPI. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

20.5. Transition state

Figure 20-3 shows SPI transition state chart.

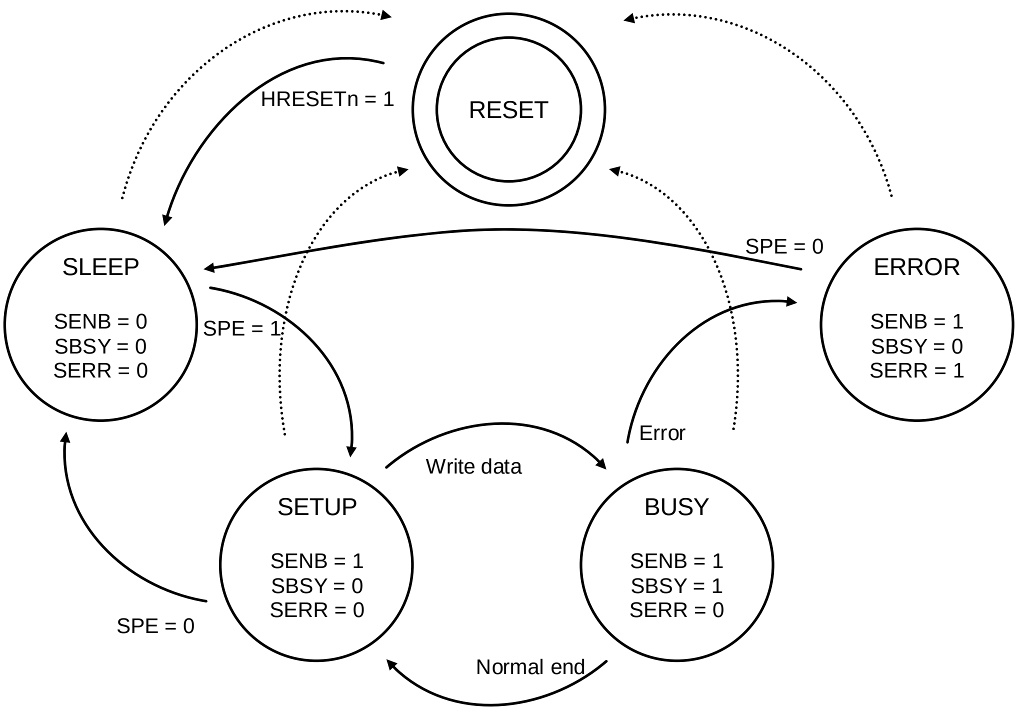


Figure 20-3 SPI state transition chart

Detail of each state shown in Figure 20-3 is as follows. SPI moves to reset state with hardware reset (HRESETn = 0) from all conditions (broken line in the chart.)

SPI state	Description
Sleep (SLEEP)	Initial state of SPI. Clock is not supplied except to state machine. While setup or transition from error state, internal logic is initialized except certain part.
Setup (SETUP)	Stand-by state of communication between master and slave. SPI changes state in the following cases. • SPE bit of SPI slave control register (SPISCR) is set to "1" in the sleep state • Communication completes properly in the busy state Received data should be read in the setup state.
Busy (BUSY)	Communicating state with SPI slave. Writing SPI data register (SPIDR) in the setup state moves to this state; in that time, transmission/reception of the data are performed simultaneously. When 1 bit is output to SPI_DO pin, 1 bit is input from SPI_DI pin. Set SIRQ at the normal termination of the communication.
Error (ERROR)	Performing prohibited register access in the busy state moves to this state. Clearing SPE bit of SPI slave control register (SPISCR) returns to sleep (SLEEP) state.

20.6. Register

This section describes SPI register.

20.6.1. Register list

SPI is controlled by the register shown in Table 20-1.

Table 20-1 SPI register list

Address		Register	Abbreviation	Description
Base	Offset			
FFF4_0000$_H$	+ 00$_H$	SPI control register	SPICR	This sets common setting with SPI
	+ 04$_H$	SPI slave control register	SPISCR	This sets SPI slave fixed setting
	+ 08$_H$	SPI data register	SPIDR	This writes and reads data to be transmitted/received to SPI slave
	+ 0C$_H$	SPI status register	SPISR	This maintains SPI state

Description format of register

Following format is used for description of register's each bit in "20.6.2 SPI control register (SPICR)" to "20.6.5 SPI status register (SPISR)".

Address	Base address + Offset address															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

20.6.2. SPI control register (SPICR)

This register is to set common setting of SPI.
SPICR setting should be carried out in the sleep or setup states, and do not write to this register in the busy state.
Each bit of SPICR is not cleared even the state is changed to sleep by SPE = 0 of SPI slave control register (SPISCR.)

Address	$FFF4\_0000_H + 00_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	SPL0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R/W	R/W	R/W
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	CDV2	CDV1	CDV0	–	–	–	–	–	–	CPOL	CPHA
R/W	R0	R0	R0	R0	R0	R/W	R/W	R/W	R0	R0	R0	R0	R0	R0	R/W	R/W
Initial value	X	X	X	X	X	0	0	0	X	X	X	X	X	X	0	0

(Note) This register should be accessed in 32 bit unit.

Bit field No.	Bit field Name	Description
31-19	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
18-17	–	Unused bits. The write access is ignored.
16	SPL0	Polarity of SPI_SS pin (slave selection pin) is specified. 0: Active-high (initial value) 1: Active-low
15-11	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
10-8	CDV2-0	Frequency dividing ratio of serial clock (SCK) to bus clock (PCLK) is specified. (see table below)
7-2	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
1	CPOL	Polarity of serial clock (SCK) is selected. 0: Positive pulse (initial value) 1: Negative pulse
0	CPHA	Timing of I/O serial data (DI/DO) and serial clock (SCK) are specified. Timing at CPHA = 0 or 1, and CPOL = 0 is shown in Figure 20-4 Timing at CPHA = 0 or 1, and CPOL = 1 is shown in Figure 20-5

CDV2	CDV1	CDV0	Frequency dividing ratio
0	0	0	PCLK × 1/2 (initial value)
0	0	1	PCLK × 1/4
0	1	0	PCLK × 1/8
0	1	1	PCLK × 1/16
1	0	0	PCLK × 1/32
1	0	1	PCLK × 1/64
1	1	0	PCLK × 1/128
1	1	1	PCLK × 1/256

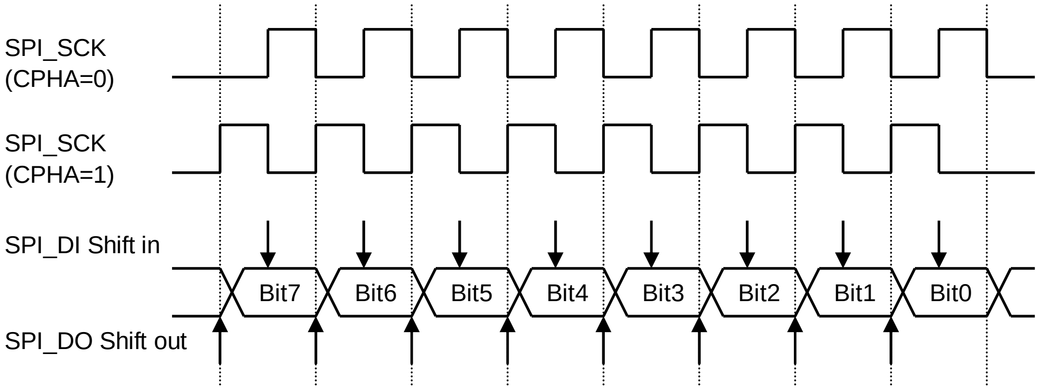


Figure 20-4 Timing of serial data and serial clock (at CPOL = 0)

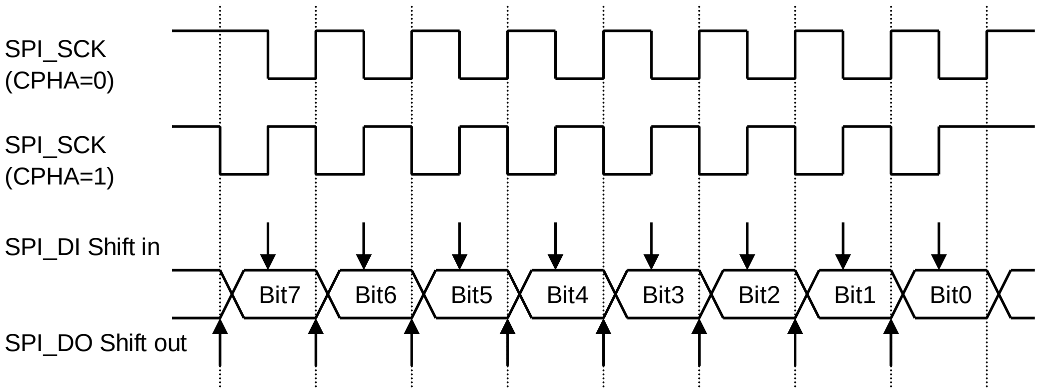


Figure 20-5 Timing of serial data and serial clock (at CPOL = 1)

20.6.3. SPI slave control register (SPISCR)

This register maintains unique setting of SPI slave.
All bits are cleared by moving state to sleep. Set this register at sleep or setup state.

Address	FFF4_0000$_H$ + 04$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	SPE	–	–	–	DRVS	–	–	–	–	STL3	STL2	STL1	STL0
R/W	R0	R0	R0	R/W	R0	R0	R0	R/W	R0	R0	R0	R0	R/W	R/W	R/W	R/W
Initial value	X	X	X	0	X	X	X	0	X	X	X	X	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	DLN4	DLN3	DLN2	DLN1	DLN0	–	–	SMOD	SAUT	–	–	SSP1	SSP0
R/W	R0	R0	R0	R/W	R/W	R/W	R/W	R/W	R0	R0	R/W	R/W	R0	R0	R/W	R/W
Initial value	X	X	X	0	0	0	0	0	X	X	0	0	X	X	0	0

(Note) This register should be accessed in 32 bit unit.

Bit field No.	Bit field Name	Description
31-29	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
28	SPE	SPI's clock supply is controlled. 0: Clock supply to internal logic stops except certain part (initial value) 1: Clock is supplied to all the circuits Write "1" to operate SPI. Its state changes from sleep to setup by setting SPE bit. It changes to sleep by clear; at the same time, internal logic is reset except certain part.
27-25	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
24	DRVS	Transfer order of serial data is specified. 0: MSB --> LSB (initial value) 1: LSB --> MSB
27-25	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
19-16	STL3-0	Strobe width is specified at pulse mode selection (SMOD = 1) in the range of SCK 1 ~ 16 cycles. 0000: SCK 1cycle (initial value) 0001: SCK 2cycles : : 1110: SCK 15cycles 1111: SCK 16cycles
15-13	–	Unused bits. The write access is ignored. The read value of these bits is always "0".

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Description</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>12-8</td><td>DLN4-0</td><td>Data length of transmission/reception serial data is specified in the range of 1 ~ 32 bit.
<table>
<tr><td>00000</td><td>1 bit (initial value)</td></tr>
<tr><td>00001</td><td>2 bit</td></tr>
<tr><td>00010</td><td>3 bit</td></tr>
<tr><td>:</td><td>:</td></tr>
<tr><td>11101</td><td>30 bit</td></tr>
<tr><td>11110</td><td>31 bit</td></tr>
<tr><td>11111</td><td>32 bit</td></tr>
</table></td></tr>
<tr><td>7-6</td><td>–</td><td>Unused bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>5</td><td>SMOD</td><td>Operation mode of slave selection is specified. Slave selection signal is output to SPI_SS pin.
<table>
<tr><td>0</td><td>Selection mode (always active while communication) (initial value)</td></tr>
<tr><td>1</td><td>Pulse mode (after communicating, this becomes active)</td></tr>
</table></td></tr>
<tr><td>4</td><td>SAUT</td><td>Operation timing of slave selection is specified according to the combination of SMOD bit.
<table>
<tr><td>0</td><td>Slave selection synchronizes with SSP bit's setting value regardless of SMOD (see Figure 20-6) (initial value)</td></tr>
<tr><td>1</td><td>1SCK of wait is added from SPI data register (SPIDR) writing to serial data transmission, and from the last data transmission to asserting/negating salve selection (see Figure 20-7)</td></tr>
</table></td></tr>
<tr><td>3-2</td><td>–</td><td>Unused bits.
The write access is ignored. The read value of these bits is always "0".</td></tr>
<tr><td>1-0</td><td>SSP1-0</td><td>Slave selection pin to be active is specified.
<table>
<tr><td>00</td><td>Slave selection pin becomes non-active (initial value)</td></tr>
<tr><td>01</td><td>SPI_SS pin becomes active</td></tr>
<tr><td>10</td><td>Reserved (setting prohibited)</td></tr>
<tr><td>11</td><td>Reserved (setting prohibited)</td></tr>
</table></td></tr>
</table>

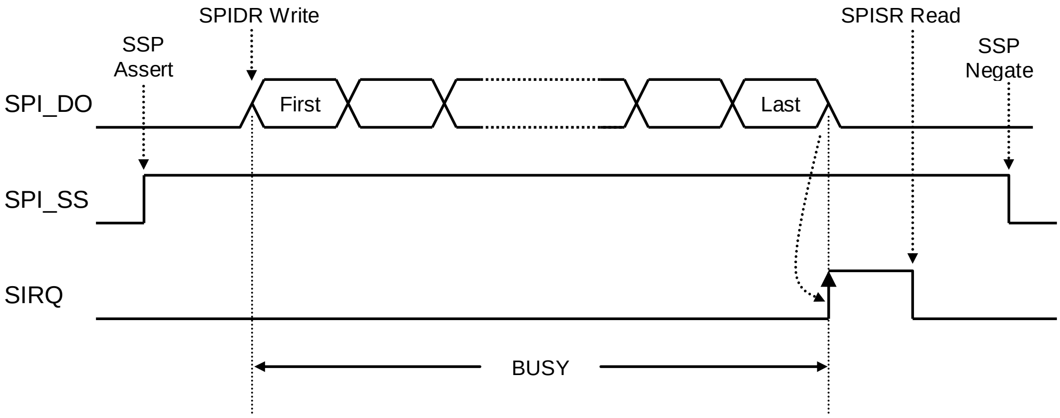


Figure 20-6 Timing chart of SPI_SS pin (at SAUT = 0)

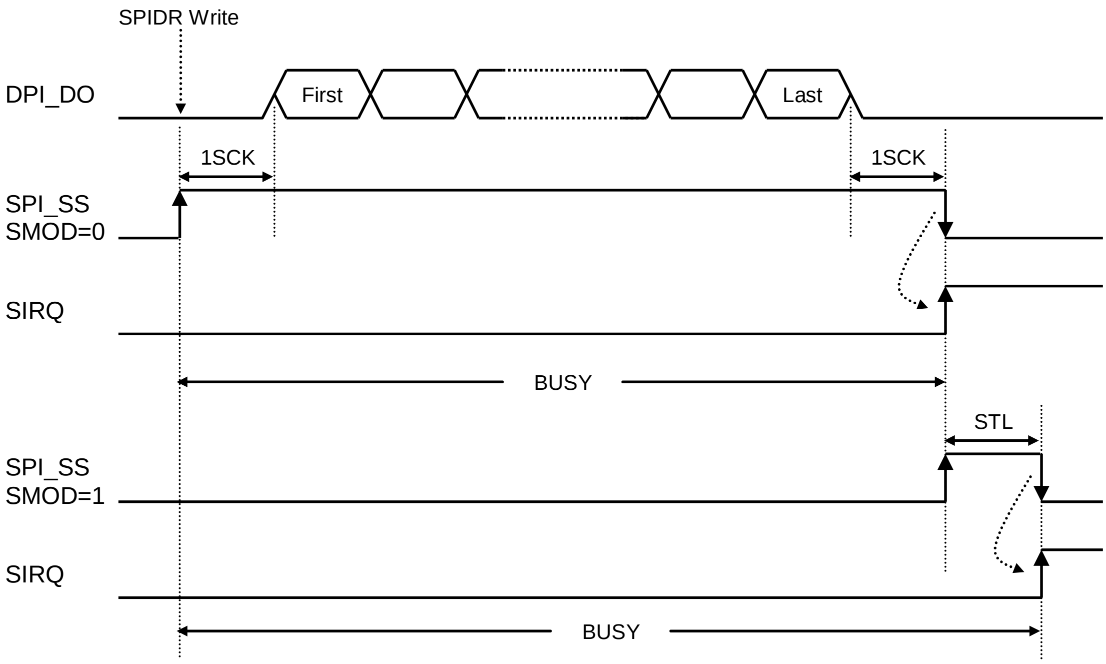


Figure 20-7 Timing chart of SPI_SS pin (at SAUT = 1)

20.6.4. SPI data register (SPIDR)

This register is used to write/read data to be transmitted to/received from SPI slave.

Address	FFF4_0000$_H$ + 08$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	D31	D30	D29	D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(Note) This register should be accessed in 32 bit unit.
Do not operate this register in the busy state.

Bit field		Description
No.	Name	
31-0	D31-0	Transmission/Reception data to SPI slave is stored. SPIDR is reset at moving to the sleep state. Writing to this register in the setup state starts transmission/reception of the data length specified in DLN[4:0] bit of SPI slave control register (SPISCR), and LSB is fixed regardless of the data length.

20.6.5. SPI status register (SPISR)

This register is to maintain SPI state, and it is not able to be written.

Address	FFF4_0000$_H$ + 0C$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	SIRQ	–	–	–	–	SERR	SBSY	SENB
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R	R	R	R	R	R	R	R
Initial value	X	X	X	X	X	X	X	X	0	X	X	X	X	0	0	0

(Note) This register should be accessed in 32 bit unit

Bit field No.	Bit field Name	Description
31-8	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
7	SIRQ	Proper completion of communication between master slaves is indicated. 0: It is under the communication or stand-by (initial value) 1: Communication is completed SIRQ pin outputs this bit. It is cleared by reading SPISR register. Figure 20-6 and Figure 20-7 show timing chart.
6-3	–	Unused bits. The write access is ignored. The read value of these bits is always "0".
2	SERR	Operation error is indicated. 0: Normal operation is in process (initial value) 1: Prohibited operation occurs Clear SPE bit of SPI slave control register (SPISCR) SERR bit is set to "1" by processing other operations than reading SPICR, SPISCR, and SPISR in the busy state. Moreover, this bit is cleared by changing state to sleep with clearing SPE bit of SPISCR.
1	SBSY	Communication with SPI slave is in process. 0: It is standing-by (initial value) 1: It is communicating SBSY is set to "1" by writing to SPI data register (SPIDR.) Do not clear SPE bit of SPISCR in the busy state. This bit is released by either of followings: • SIRQ bit setting • SERR bit setting
0	SENB	SPI circuit is active. 0: Clock supply to internal logic is stop except to certain part (initial value) 1: Clock is supplied to all the circuits

20.7. Setup procedure flow

Figure 20-8 shows SPI setup procedure flow.

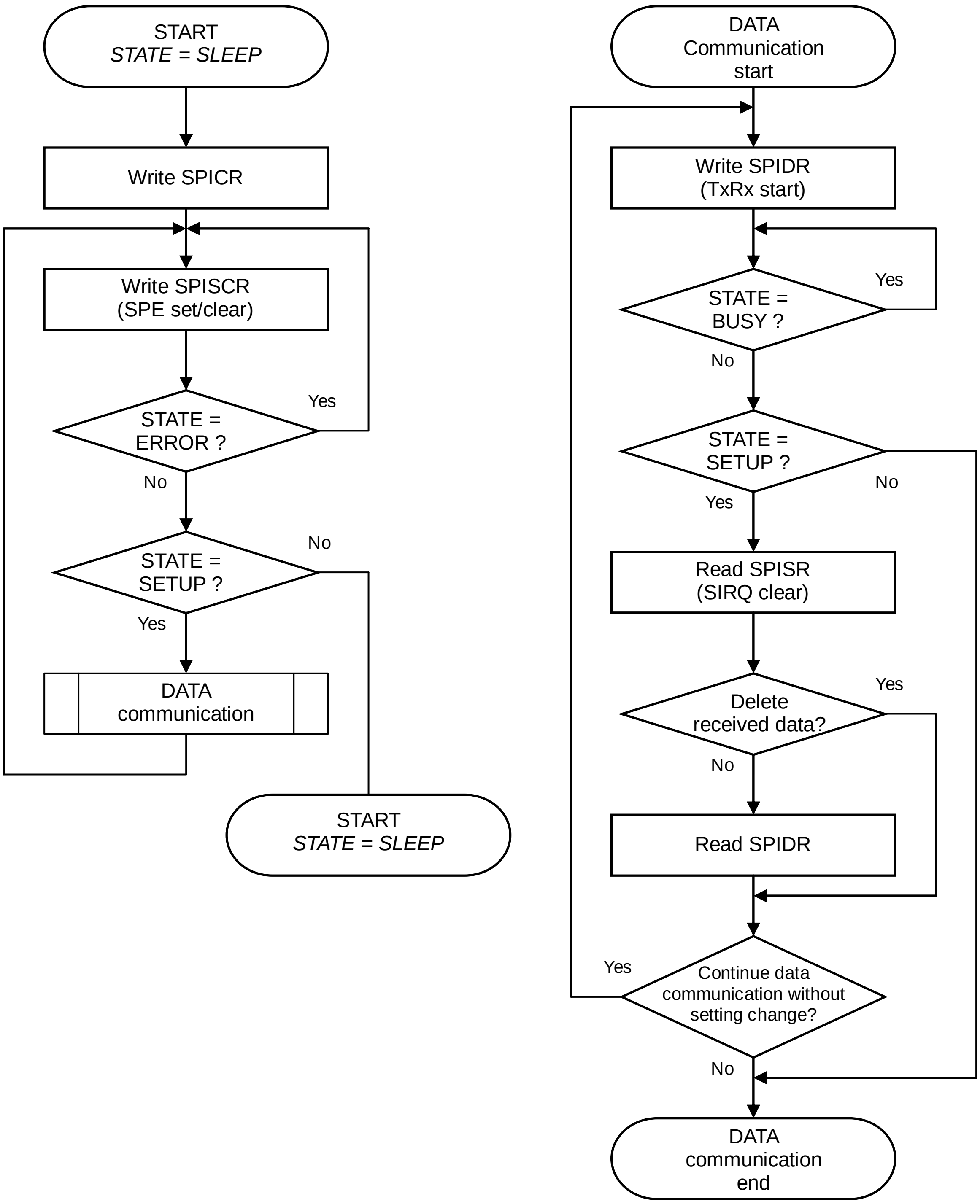


Figure 20-8 SPI setup flow chart

21. CAN interface (CAN)

This chapter describes CAN interface.
Refer following website for CAN module specification.

URL: http://www.semiconductors.bosch.de/en/20/can/products/ccan.asp

21.1. Outline

MB86R03 equips 2 ports of CAN interface which is in compliance with CAN protocol version 2.0 part A and B.

21.2. Block diagram

Figure 21-1 shows block diagram of CAN.

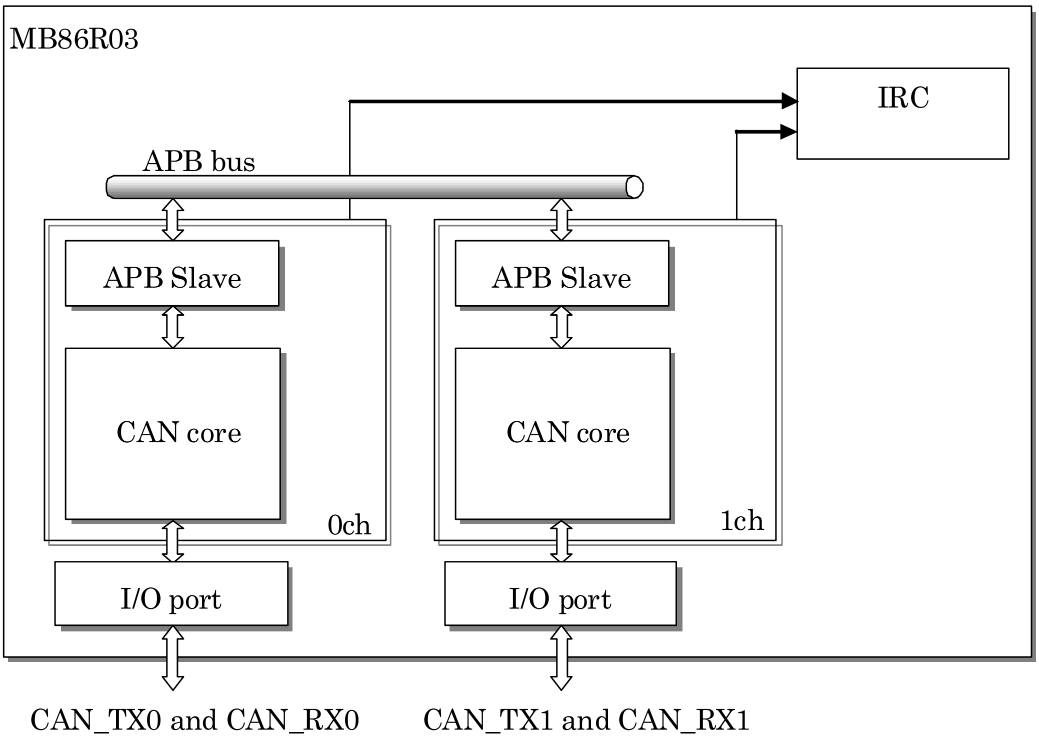


Figure 21-1 Block diagram of CAN

21.3. Supply clock

APB clock is supplied to CAN interface. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

21.4. Register

Register mapping of this LSI is in byte address (8 bit.)
16 bit length of register is allocated by word address unit (32 bit) for local address of CAN; thus valid data in 32 bit width data of APB Bus is 16 bit.

Table 21-1 CAN 0ch register map

Register address	CAN 0ch register address	APB Bus data[31:0]
FFF5_4000h	00h	{0x0000, 16 bit data}
FFF5_4004h	02h	{0x0000, 16 bit data}
FFF5_4008h	04h	{0x0000, 16 bit data}
…	…	…

Table 21-2 CAN 1ch register map

Register address	CAN 1ch register address	APB Bus data [31:0]
FFF5_5000h	00h	{0x0000, 16 bit data}
FFF5_5004h	02h	{0x0000, 16 bit data}
FFF5_5008h	04h	{0x0000, 16 bit data}
…	…	…

22. Chip Control Module (CCNT)

This chapter describes function and operation of Chip Control Module (CCNT.)

22.1. Outline

CCNT performs pin multiplex control, software reset control, AXI interconnect control and others.

22.2. Feature

- Multiplex pin interface:
  Mode selection setting of pin multiplex groups 2 and 4
- Software reset interface:
  Issuing software reset to each module in the register
- External pin interface:
  Indicating signal level of the external pin in status
- AXI interconnect interface:
  Setting AXI wait and priority of bus right
- INT interface:
  Setting interrupt mask and interrupt information clear
- Byte swap interface:
  Setting byte swap of SDMC and I2S
- DDR2 controller interface:
  Reset control in DDR2 controller
- GPIO interface

22.3. Block diagram

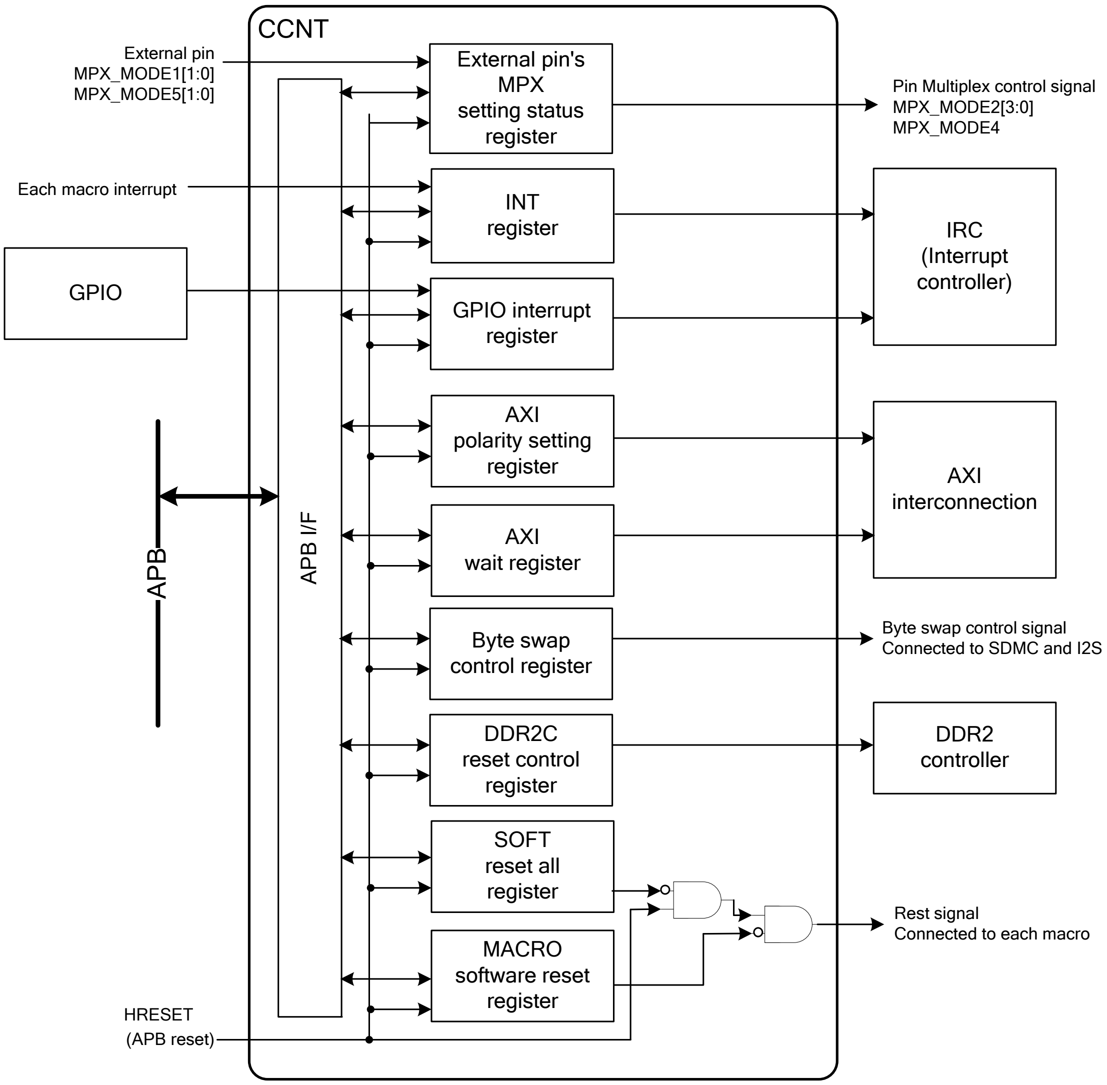


Figure 22-1 Block diagram of CCNT

22.4. Supply clock

AHB clock is supplied to CCNT. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

22.5. Register

This section describes CCNT module register.

22.5.1. Register list

CCNT unit contains register shown in Table 22-1.

Table 22-1 CCNT register list

Address	Register	Description
FFF42000	CCID	Chip ID register
FFF42004	CSRST	Software reset register
FFF42008 – FFF4200F	Reserved	Access prohibited
FFF42010	CIST	Interrupt status register
FFF42014	CISTM	Interrupt status mask register
FFF42018	CGPIO_IST	GPIO interrupt status register
FFF4201C	CGPIO_ISTM	GPIO interrupt status mask register
FFF42020	CGPIO_IP	GPIO interrupt polarity setting register
FFF42024	CGPIO_IM	GPIO interrupt mode setting register
FFF42028	CAXI_BW	AXI bus wait cycle setting register
FFF4202C	CAXI_PS	AXI polarity setting register
FFF42030	CMUX_MD	Multiplex mode setting register
FFF42034	CEX_PIN_ST	External pin status register
FFF42038	Reserved	Access prohibited
FFF4203C	Reserved	Access prohibited
FFF42040	Reserved	Access prohibited
FFF42044 - FFF420E7	Reserved	Access prohibited
FFF420E8	CBSC	Byte swap switching register
FFF420EC	CDCRC	DDR2 controller reset control register
FFF420F0	CMSR0	Software reset register 0 for macro
FFF420F4	CMSR1	Software reset register 1 for macro

Description format of register

Following format is used for description of register's each bit in "22.5.2 CHIP ID register (CCID)" to "22.5.17 Software reset register 1 for macro (CMSR1)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field's value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

22.5.2. CHIP ID register (CCID)

Address	FFF4_2000 + 00h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	YEAR[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHIPNAME[7:0]								VERSION[7:0]							
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0

Bit field		Function
No.	Name	
31-16	YEAR[15:0]	Date of LSI development is indicated in 4 digit dominical year. In this LSI, 2006(h) is read.
15-8	CHIPNAME[7:0]	LSI identification name is indicated in ID number. In this LSI, 10(h) is read.
7-0	VERSION[7:0]	LSI version is indicated. In this LSI, 0A(h) is read.

22.5.3. Software reset register (CSRST)

Address	FFF4_2000 + 04h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)															SFTRST
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Function
31-1	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
0	SFTRST (Software reset)	Writing "1" to this bit outputs reset to macro (GDC, DDR2 controller, CAN, SDMC, I2S, SPI, I2C, PWM, UART, GPIO, and DMAC) in Chip. Since register value is output as it is (level output), "0" should be set again to release reset. 0: Not reset (initial value) 1: Reset

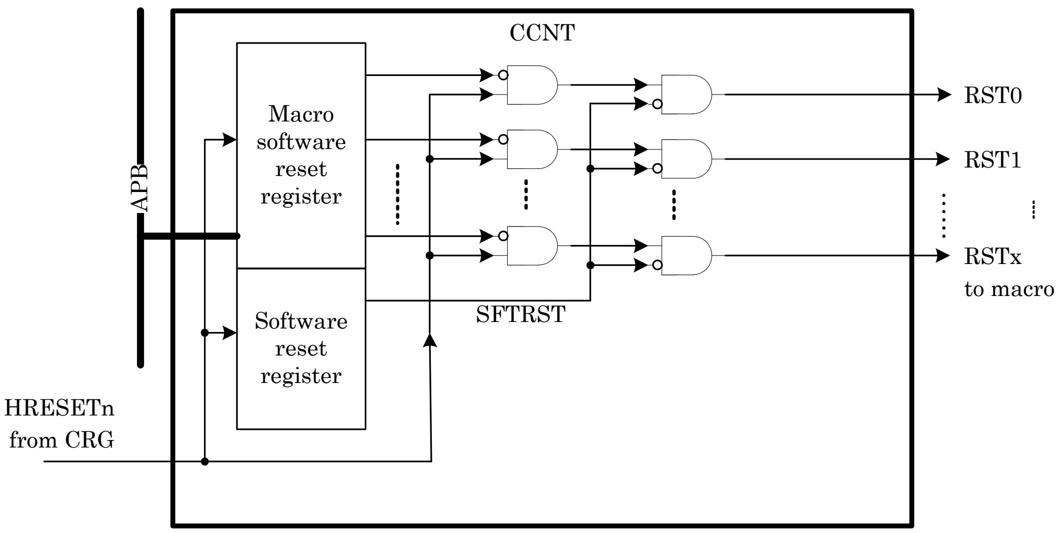


Figure 22-2 Details of software reset

22.5.4. Interrupt status register (CIST)

Address	FFF4_2000 + 10h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)			INT28	INT27	INT26	(Reserved)	INT24	(Reserved)							
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)										INT5	(Reserved)				
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Function
No.	Name	
31	(Reserved)	Reserved bit. Initial value is "0". Setting other values than the initial value is prohibited.
30-29	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
28	INT28	When HBUS2AXI error interrupt occurs, "1" is set. Writing "0" to this bit clears INT information to "0". When bit 28 of the Interrupt status mask register is set to mask "0", this bit is fixed to "0". 0: No interrupt (initial value) 1: Interrupt (HBUS2AXI)
27	INT27	When MBUS2AXI (Draw) error interrupt occurs, "1" is set. Writing "0" to this bit clears INT information to "0". When bit 27 of the Interrupt status mask register is set to mask "0", this bit is fixed to "0". 0: No interrupt (initial value) 1: Interrupt (MBUS2AXI (Draw))
26	INT26	When MBUS2AXI (DispCap) error interrupt occurs, "1" is set. Writing "0" to this bit clears INT information to "0". When bit 26 of the Interrupt status mask register is set to mask "0", this bit is fixed to "0". 0: No interrupt (initial value) 1: Interrupt (MBUS2AXI (DispCap))
25	(Reserved)	Reserved bit. Initial value is "0". Setting other values than the initial value is prohibited.
24	INT24 (AHB2AXI)	When AHB2AXI error interrupt occurs, "1" is set. Writing "0" to this bit clears INT information to "0". When bit 24 of the Interrupt status mask register is set to mask "0", this bit is fixed to "0". 0: No interrupt (initial value) 1: Interrupt (AHB2AXI)
23-6	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>5</td><td>INT5</td><td>When MBUS2AXI (Cap) error interrupt occurs, "1" is set. Writing "0" to this bit clears INT information to "0". When bit 5 of the Interrupt status mask register is set to mask "0", this bit is fixed to "0".
<table><tr><td>0</td><td>No interrupt (initial value)</td></tr><tr><td>1</td><td>Interrupt (MBUS2AXI (Cap))</td></tr></table></td></tr>
<tr><td>4-0</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
</table>

22.5.5. Interrupt status mask register（CISTM）

Address	FFF4_2000 + 14h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	INT31 MASK	(Reserved)		INT28 MASK	INT27 MASK	INT26 MASK	(Reserved)	INT24 MASK	(Reserved)							
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)										INT5 MASK	(Reserved)			INT1 MASK	INT0 MASK
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Function
31	INT31 Mask	Writing "1" to this bit validates MLB_DINT interrupt. 0 Mask (initial value) 1 INT31 is valid (MLB_DINT interrupt)
30-29	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
28	INT28 Mask	Writing "1" to this bit validates HBUS2AXI interrupt. 0 Mask (initial value) 1 INT28 is valid (HBUS2AXI interrupt)
27	INT27 Mask	Writing "1" to this bit validates MBUS2AXI (Draw) interrupt. 0 Mask (initial value) 1 INT27 is valid (MBUS2AXI (Draw))
26	INT26 Mask	Writing "1" to this bit validates MBUS2AXI (Disp) interrupt. 0 Mask (initial value) 1 INT26 is valid (MBUS2AXI (Disp) interrupt)
25	(Reserved)	Reserved bit. Initial value is "0". Setting other values than the initial value is prohibited.
24	INT24 Mask	Writing "1" to this bit validates AHB2AXI interrupt. 0 Mask (initial value) 1 INT24 is valid (AHB2AXI interrupt)
23-6	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
5	INT5 Mask	Writing "1" to this bit validates MBUS2AXI (Cap) interrupt. 0 Mask (initial value) 1 INT5 is valid (MBUS2AXI (Cap) interrupt)
4-2	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>1</td><td>INT1 Mask</td><td>Writing "1" to this bit validates ADC ch1 interrupt.
<table><tr><td>0</td><td>Mask (initial value)</td></tr><tr><td>1</td><td>INT1 is valid (ADC ch1 interrupt)</td></tr></table></td></tr>
<tr><td>0</td><td>INT0 Mask</td><td>Writing "1" to this bit validates ADC ch0 interrupt.
<table><tr><td>0</td><td>Mask (initial value)</td></tr><tr><td>1</td><td>INT0 is valid (ADC ch0 interrupt)</td></tr></table></td></tr>
</table>

22.5.6. GPIO interrupt status register (CGPIO_IST)

This register is to indicate GPIO related interrupt status.

Address	FFF4_2000 + 18h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)								GPIO_INT_status[23:16]							
R/W	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO_INT_status[15:0]															
R/W	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Function
No.	Name	
31-24	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
23-0	GPIO_INT_status (GPIO interrupt status)	This is cleared by "0" writing. GPIO's applied bit indicates interrupt is occurred. 0: Interrupt is not occurred 1: Interrupt is occurred

22.5.7. GPIO interrupt status mask register (CGPIO_ISTM)

This register is to control GPIO related interrupt which is judged by the setting status regardless of input/output. Each setting bit can be set corresponding to each bit one-by-one from MSB to LSB.

Address	FFF4_2000 + 1Ch															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)								GPIO_INT_enable[23:16]							
R/W	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO_INT_enable[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Function
No.	Name	
31-24	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
23-0	GPIO_INT_enable (GPIO interrupt enable)	Whether to generate interrupt with the value sampled external pin, GPIO23-0 in internal clock is set by bit. 0: Interrupt does not occur 1: Interrupt occurs based on the register setting shown from the next page

22.5.8. GPIO interrupt polarity setting register (CGPIO_IP)

This register is to control GPIO related interrupt which is judged by the setting status regardless of input/output. Each setting bit can be set corresponding to each bit one-by-one from MSB to LSB.

Address	**FFF4_2000 + 20h**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)								GPIO_INT_polarity[23:16]							
R/W	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO_INT_polarity[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		**Function**
No.	**Name**	
31-24	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
23-0	GPIO_INT_polarity (GPIO interrupt polarity)	Interrupt occurs with the following value. 0 — Level "0" or negative edge is detected (GPIO_INT_mode dependant) 1 — Level "1" or positive edge is detected (GPIO_INT_mode dependant)

22.5.9. GPIO interrupt mode setting register (CGPIO_IM)

This register is to control GPIO related interrupt which is judged by the setting status regardless of input/output. Each setting bit can be set corresponding to each bit one-by-one from MSB to LSB.

Address	**FFF4_2000 + 24h**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)								GPIO_INT_mode[23:16]							
R/W	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R0/W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO_INT_mode[15:0]															
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		**Function**
No.	**Name**	
31-24	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
23-0	GPIO_INT_polarity (GPIO interrupt polarity)	GPIO_INT_mode (GPIO interrupt mode) 0 — Level sensitive ("0" or "1" is GPIO_INT_polarity dependant) 1 — Edge sensitive ("pos" or "neg" is GPIO_INT_polarity dependant)

22.5.10. AXI bus wait cycle setting register (CAXI_BW)

Address	FFF4_2000 + 28h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Disp_RWait[3:0]				Disp_WWait[3:0]				Draw_RWait[3:0]				Draw_WWait[3:0]			
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								PrimaryAHB_RWait[3:0]				PrimaryAHB_WWait[3:0]			
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Function
No.	Name	
31-28	Disp_RWait (Read Wait)	Wait time of AXI write (between the transactions) is able to be set in the range of 0_H (No Wait) - F_H (15 cycle Wait.) Initial value is 0_H (No Wait.) (Note) 1 cycle is AXI 1 clock.
27-24	Disp_WWAIT (Write Wait)	Wait time of AXI read (between the transactions) is able to be set in the range of 0_H (No Wait) - F_H (15cycle Wait.) Initial value is 0_H (No Wait.) (Note) 1 cycle is AXI 1 clock.
23-20	Draw_RWAIT (Read Wait)	Wait time of AXI write (between the transactions) is able to be set in the range of 0_H (No Wait) - F_H (15cycle Wait.) Initial value is 0_H (No Wait.) (Note) 1 cycle is AXI 1 clock.
19-16	Draw_WWAIT (Write Wait)	Wait time of AXI read (between the transactions) is able to be set in the range of 0_H (No Wait) - F_H (15cycle Wait.) Initial value is 0_H (No Wait.) (Note) 1 cycle is AXI 1 clock.
15-8	(Reserved)	Reserved bit. Initial value is 0_H. Setting other than initial value is prohibited.
7-4	PrimaryAHB_RWAIT (Write Wait)	Wait time of AXI write (between the transactions) is able to be set in the range of 0_H (No Wait) - F_H (15cycle Wait.) Initial value is 0_H (No Wait.) (Note) 1 cycle is AXI 1 clock.
3-0	PrimaryAHB_WWAIT (Read Wait)	Wait time of AXI read (between the transactions) is able to be set in the range of 0_H (No Wait) - F_H (15cycle Wait.) Initial value is 0_H (No Wait.) (Note) 1 cycle is AXI 1 clock.

22.5.11. AXI polarity setting register (CAXI_PS)

This register is to prioritize the bus right on AXI Inter Connect. The priority on the AXI bus is as follows.

PSEL_0 > PSEL_1 > PSEL_2 >PSEL_3 > PSEL_4

Set bus master identification code 0-4 to each setting bit. 5 or more of value and overlapping value are not available; in this case, register writing is ignored and the previous setting value is kept.

Note:

The PSEL_2 setting bit should be fixed to "010". Setting "010" to PSEL_0, PSEL_1, PSEL_3, and PSEL_4 is prohibited.

Address	FFF4_2000 + 2Ch															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)													P_SEL4		
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)	P_SEL3			(Reserved)	P_SEL2			(Reserved)	P_SEL1			(Reserved)	P_SEL0		
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
Initial value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31-19</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>18-16</td><td>P_SEL4
(Priority Select4)</td><td>Priority order of AXI Inter Connect is set.
<table>
<tr><td>000</td><td>DispCap</td></tr>
<tr><td>001</td><td>AHB</td></tr>
<tr><td>010</td><td>(Setting prohibited)</td></tr>
<tr><td>011</td><td>HBUS</td></tr>
<tr><td>100</td><td>DRAW (initial value)</td></tr>
<tr><td>101-111</td><td>(Setting prohibited)</td></tr>
</table></td></tr>
<tr><td>15</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>14-12</td><td>P_SEL3
(Priority Select3)</td><td>Priority order of AXI Inter Connect is set.
<table>
<tr><td>000</td><td>DispCap</td></tr>
<tr><td>001</td><td>AHB</td></tr>
<tr><td>010</td><td>(Setting prohibited)</td></tr>
<tr><td>011</td><td>HBUS (initial value)</td></tr>
<tr><td>100</td><td>DRAW</td></tr>
<tr><td>101-111</td><td>(Setting prohibited)</td></tr>
</table></td></tr>
<tr><td>11</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>10-8</td><td>P_SEL2
(Priority Select2)</td><td>Priority order of AXI Inter Connect is set.
<table><tr><td>000</td><td>(Setting prohibited)</td></tr><tr><td>001</td><td>(Setting prohibited)</td></tr><tr><td>010</td><td>This bit field should be fixed to 010 (initial value).</td></tr><tr><td>011</td><td>(Setting prohibited)</td></tr><tr><td>100</td><td>(Setting prohibited)</td></tr><tr><td>101-111</td><td>(Setting prohibited)</td></tr></table></td></tr>
<tr><td>7</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>6-4</td><td>P_SEL1
(Priority Select1)</td><td>Priority order of AXI Inter Connect is set.
<table><tr><td>000</td><td>DispCap</td></tr><tr><td>001</td><td>AHB (initial value)</td></tr><tr><td>010</td><td>(Setting prohibited)</td></tr><tr><td>011</td><td>HBUS</td></tr><tr><td>100</td><td>DRAW</td></tr><tr><td>101-111</td><td>(Setting prohibited)</td></tr></table></td></tr>
<tr><td>3</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>2-0</td><td>P_SEL0
(Priority Select0)</td><td>Priority order of AXI Inter Connect is set.
<table><tr><td>000</td><td>DispCap (initial value)</td></tr><tr><td>001</td><td>AHB</td></tr><tr><td>010</td><td>(Setting prohibited)</td></tr><tr><td>011</td><td>HBUS</td></tr><tr><td>100</td><td>DRAW</td></tr><tr><td>101-111</td><td>(Setting prohibited</td></tr></table></td></tr>
</table>

22.5.12. Multiplex mode setting register (CMUX_MD)

Address	FFF4_2000 + 30h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)										MPX_MODE_4		(Reserved)	MPX_MODE_2		
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1

Bit field		Function
No.	**Name**	
31-6	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
5-4	MPX_MODE_4	External pin's multiplexed group #4 is set. 00: Mode 0 01: Mode 1 10: Reserved 11: (Initial value)
3	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
2-0	MPX_MODE_2	External pin's multiplexed group #2 is set. 000: Mode 0 001: Mode 1 010: Mode 2 011: Mode 3 100: Mode 4 101 – 0110: Reserved 111: (Initial value)

22.5.13. External pin status register (CEX_PIN_ST)

Address	FFF4_2000 + 34h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)				CRIPM[3:0]				(Reserved)				MPX_MODE_5		MPX_MODE_1	
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Initial value	0	0	0	0	X	X	X	X	0	0	0	0	X	X	X	X

Bit field		Function
No.	Name	
31-12	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
11-8	CRIPM	Status of PLL multiple number setting pin is displayed.
7-4	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
3-2	MPX_MODE_5	Setting pin status for external pin's multiplexed group #5 is displayed. 00: Mode 0 01: Mode 1 10: Mode 2 11: Mode 0
1-0	MPX_MODE_1	Setting pin status for external pin's multiplexed group #1 is displayed. 00: Mode 0 01: Mode 1 10: Mode 2 11: Mode 0

22.5.14. Byte swap switching register (CBSC)

This register is for byte swap switching and is set as follows.

<table>
<tr><td>wSEL</td><td>0 (Little)</td><td colspan="2">1 (Big)</td></tr>
<tr><td>HWSWAP</td><td>- (no swap)</td><td>0 (Swap)</td><td>1 (no swap)</td></tr>
<tr><td>WSWAP</td><td>- (no swap)</td><td>0 (Swap)</td><td>1 (no swap)</td></tr>
</table>

wSEL: Little/Big switching signal
HWSWAP: Hword byte swap switching signal at big endian
WSWAP: Word byte swap switching signal at big endian

<table>
<tr><th>Address</th><th colspan="16">FFF4_2000 + E8h</th></tr>
<tr><td>Bit</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td>Name</td><td colspan="9">(Reserved)</td><td colspan="3">SDMC_Endian[2:0]</td><td>(Reserved)</td><td colspan="3">I2S0_Endian[2:0]</td></tr>
<tr><td>R/W</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
<tr><td>Initial value</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Bit</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Name</td><td>(Reserved)</td><td colspan="3">I2S1_Endian[2:0]</td><td>(Reserved)</td><td colspan="3">I2S2_Endian[2:0]</td><td colspan="8">(Reserved)</td></tr>
<tr><td>R/W</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
<tr><td>Initial value</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>31</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>30-28</td><td>(Reserved)</td><td>Reserved bit.
Initial value is 000_B. Setting other values than the initial value is prohibited.</td></tr>
<tr><td>27</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>26-24</td><td>(Reserved)</td><td>Reserved bit.
Initial value is 000_B. Setting other values than the initial value is prohibited.</td></tr>
<tr><td>23</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>22-20</td><td>SDMC_Endian</td><td>Endian switch of SDMC is controlled.<table><tr><td>Bit 22</td><td>wSEL</td><td>Endian switch 0:Little, 1:Big</td></tr><tr><td>Bit 21</td><td>HWSAP</td><td>Hword byte swap switching signal at Big</td></tr><tr><td>Bit 20</td><td>WSWAP</td><td>Word byte swap switching signal at Big</td></tr></table></td></tr>
<tr><td>19</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>18-16</td><td>I2S0_Endian</td><td>Endian switch of I2S0 is controlled.<table><tr><td>Bit 18</td><td>wSEL</td><td>Endian switch 0:Little, 1:Big</td></tr><tr><td>Bit 17</td><td>HWSAP</td><td>Hword byte swap switching signal at Big</td></tr><tr><td>Bit 16</td><td>WSWAP</td><td>Word byte swap switching signal at Big</td></tr></table></td></tr>
<tr><td>15</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>14-12</td><td>I2S1_Endian</td><td>Endian switch of I2S1 is controlled.<table><tr><td>Bit 14</td><td>wSEL</td><td>Endian switch 0:Little, 1:Big</td></tr><tr><td>Bit 13</td><td>HWSAP</td><td>Hword byte swap switching signal at Big</td></tr><tr><td>Bit 12</td><td>WSWAP</td><td>Word byte swap switching signal at Big</td></tr></table></td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>11</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>10-8</td><td>I2S2_Endian</td><td>Endian switch of I2S2 is controlled.
<table><tr><td>Bit 10</td><td>wSEL</td><td>Endian switch 0:Little, 1:Big</td></tr><tr><td>Bit 9</td><td>HWSAP</td><td>Hword byte swap switching signal at Big</td></tr><tr><td>Bit 8</td><td>WSWAP</td><td>Word byte swap switching signal at Big</td></tr></table></td></tr>
<tr><td>7</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>6-4</td><td>(Reserved)</td><td>Reserved bit.
Initial value is 000_B. Setting other values than the initial value is prohibited.</td></tr>
<tr><td>3-2</td><td>(Reserved)</td><td>Reserved bit.
Write access is ignored. Read value of these bits is always "0".</td></tr>
<tr><td>1-0</td><td>(Reserved)</td><td>Reserved bit.
Initial value is 00_B. Setting other values than the initial value is prohibited.</td></tr>
</table>

22.5.15. DDR2 controller reset control register (CDCRC)

This register is to output reset to DDR-IF macro in DDR2 controller by writing "0" to each bit. Since register value is output as it is (level output), "1" should be set again to release reset.

Address	FFF4_2000 + ECh															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)														*1	*2
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

*1: IRESET&IUSRRST
*2: IDLLRST

Bit field		Function
No.	**Name**	
31-2	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
1	IRESET&IUSRRST	IRESET and IUSRRST to DDR-IF macro in DDR2 controller is controlled. 0: Reset (initial value) 1: Not Reset
0	IDLLRST	IDLLRST to DDR-IF macro in DDR2 controller is controlled. 0: Reset (initial value) 1: Not Reset

22.5.16. Software reset register 0 for macro (CMSR0)

Address	FFF4_2000 + F0h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)						SRST0_25	SRST0_24	(Reserved)							SRST0_16
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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	(Reserved)								SRST0_7	(Reserved)	SRST0_5	SRST0_4	SRST0_3	SRST0_2	SRST0_1	SRST0_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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Function
No.	Name	
31-26	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
25	SRST0_25 (UART1 Software reset)	Reset is output to UART1 macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (Initial value) 1: Software reset
24	SRST0_24 (UART0 Software reset)	Reset is output to UART0 macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (Initial value) 1: Software reset
23-17	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
16	SRST0_16 (DMAC Software reset)	Reset is output to DMAC macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (Initial value) 1: Software reset
15-8	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
7	SRST0_7 (GPIO Software reset)	Reset is output to GPIO macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (Initial value) 1: Software reset
6-5	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
4	SRST0_4 (GDC DISP1 Software reset)	Reset is output to GDC DISP1 macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (Initial value) 1: Software reset

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>3</td><td>SRST0_3 (GDC DISP0 Software reset)</td><td>Reset is output to GDC DISP0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.
<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>2</td><td>SRST0_2 (GDC CAP1 Software reset)</td><td>Reset is output to GDC CAP1 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.
<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>1</td><td>SRST0_1 (GDC CAP0 Software reset)</td><td>Reset is output to GDC CAP0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.
<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>0</td><td>SRST0_0 (GDC Draw Software reset)</td><td>Reset is output to GDC Draw macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.
<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
</table>

22.5.17. Software reset register 1 for macro (CMSR1)

Address	FFF4_2000 + F4h															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	(Reserved)			SRST1_28	SRST1_27	SRST1_26	SRST1_25	SRST0_24	(Reserved)					SRST1_18	SRST1_17	SRST1_16
R/W	R	R	R	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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SRST1_15	SRST1_14	SRST1_13	SRST1_12	SRST1_11	(Reserved)	SRST1_9	SRST1_8	SRST1_7	SRST1_6	SRST1_5	SRST1_4	SRST1_3	SRST1_2	SRST1_1	SRST1_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
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit field		Function
No.	Name	
31-30	(Reserved)	Reserved bit. Write access is ignored. Read value of these bits is always "0".
29	(Reserved)	Reserved bit. Initial value is "0". Setting other values than the initial value is prohibited.
28	SRST1_28 (HBUS2AXI Software reset)	Reset is output to HBUS2AXI macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (initial value) 1: Software reset
27	SRST1_27 (MBUS2AXI(Draw) Software reset)	Reset is output to MBUS2AXI (Draw) macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (initial value) 1: Software reset
26	SRST1_26 (MBUS2AXI(Disp) Software reset)	Reset is output to MBUS2AXI (Disp) macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (initial value) 1: Software reset
25	SRST1_25 (AHB2AXI(CPUroot) Software reset)	Reset is output to AHB2AXI (CPUroot) macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (initial value) 1: Software reset
24	SRST1_24 (AHB2AXI(AHBBus) Software reset)	Reset is output to AHB2AXI (AHB Bus) macro by writing "1" to this bit. Since register value is output as it is (level output), "0" should be set again to release reset. 0: No software reset (initial value) 1: Software reset
23-19	(Reserved)	Reserved bit. Initial value is 00000_B. Setting other values than the initial value is prohibited.

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>18</td><td>SRST1_18
(UART5 Software reset)</td><td>Reset is output to UART5 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>17</td><td>SRST1_17
(UART4 Software reset)</td><td>Reset is output to UART4 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>16</td><td>SRST1_16
(UART3 Software reset)</td><td>Reset is output to UART3 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>15</td><td>SRST1_15
(UART2 Software reset)</td><td>Reset is output to UART2 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>14</td><td>SRST1_14
(PWM_1 Software reset)</td><td>Reset is output to PWM_1 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>13</td><td>SRST1_13
(PWM_0 Software reset)</td><td>Reset is output to PWM_0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>12</td><td>SRST1_12 (I2C_0 Software reset)</td><td>Reset is output to I2C_0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>11</td><td>SRST1_11 (I2C_0 Software reset)</td><td>Reset is output to I2C_0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>10</td><td>(Reserved)</td><td>Reserved bit.
Initial value is "0". Setting other values than the initial value is prohibited.</td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>9</td><td>SRST1_9 (SPI Software reset)</td><td>Reset is output to SPI macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>8</td><td>SRST1_8 (I2S_2 Software reset)</td><td>Reset is output to I2S_2 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>7</td><td>SRST1_7 (I2S_1 Software reset)</td><td>Reset is output to I2S_1 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>6</td><td>SRST1_6 (I2S_0 Software reset)</td><td>Reset is output to I2S_0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (Initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>5</td><td>SRST1_5 (MBUS2AXI(Cap))</td><td>Reset is output to MBUS2AXI (Cap) macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>4</td><td>SRST1_4 (SDMC Software reset)</td><td>Reset is output to SDMC macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>3</td><td>SRST1_3 (CAN1 Software reset)</td><td>Reset is output to CAN1macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>2</td><td>SRST1_2 (CAN0 Software reset)</td><td>Reset is output to CAN0 macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
</table>

<table>
<tr><th colspan="2">Bit field</th><th rowspan="2">Function</th></tr>
<tr><th>No.</th><th>Name</th></tr>
<tr><td>1</td><td>SRST1_1 (DDR2 Software reset)</td><td>Reset is output to DDR2 controller macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
<tr><td>0</td><td>SRST1_0 (GDC Software reset)</td><td>Reset is output to GDC macro by writing "1" to this bit.
Since register value is output as it is (level output), "0" should be set again to release reset.<table><tr><td>0</td><td>No software reset (initial value)</td></tr><tr><td>1</td><td>Software reset</td></tr></table></td></tr>
</table>

23. External interrupt controller (EXIRC)

This chapter describes function and operation of external interrupt controller (EXIRC.)

23.1. Outline

EXIRC is block to control external interrupt as well as external interrupt request input to external pin of INT_A[3] - INT_A [0]. "H" level, "L" level, rising edge, and falling edge are selectable as detected input request level.

23.2. Feature

EXIRC has following features:

- Operating as bus slave of AMBA (APB)
- 4 channels of external interrupt control
- 4 input request level selections
    - "H" level
    - "L" level
    - Rising edge
    - Falling edge
- Utilization of external interrupt as returning factor from Stop mode

23.3. Block diagram

Figure 23-1 shows block diagram of EXIRC.

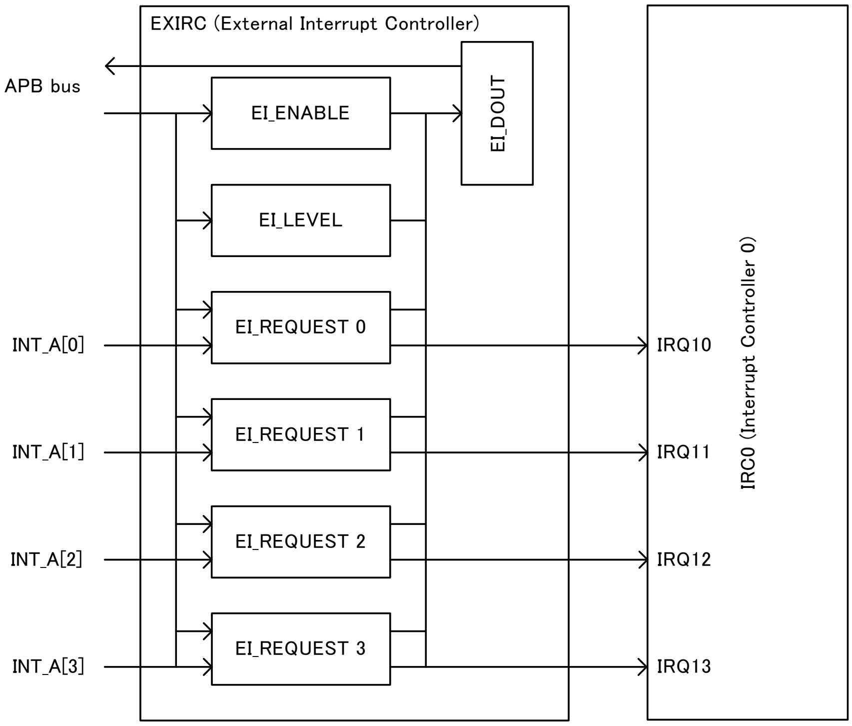


Figure 23-1 Block diagram of EXIRC

Table 23-1 shows block function included in EXIRC.

Table 23-1 Block function included in EXIRC

Block	Function
EI_ENABLE	Enabling external interrupt request for interrupt controller (IRC0)
EI_LEVEL	Setting input request level: "H" level/"L" level/rising edge/falling edge
EI_REQUEST	Synchronizing and maintaining interrupt request
EI_DOUT	Generating data for reading

23.4. Supply clock

APB clock is supplied to EXIRC. Refer to "5. Clock reset generator (CRG)" for frequency setting and control specification of the clock.

23.5. Register

This section describes EXIRC register.

23.5.1. Register list

Table 23-2 shows EXIRC register list.

Table 23-2 EXIRC register list

Address		Register	Abbreviation	Description
Base	Offset			
$FFFE\_4000_H$	$+ 00_H$	External interrupt enable register	EIENB	Enable control of external interrupt request output
	$+ 04_H$	External interrupt request register	EIREQ	Clear function of external interrupt display and interrupt request
	$+ 08_H$	External interrupt level register	EILVL	Selection of input request level detection of external interrupt

Description format of register

Following format is used for description of register’s each bit in "23.5.2 External interrupt enable register (EIENB)" to "23.5.4 External interrupt level register (EILVL)".

Address	**Base address + Offset address**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
R/W																
Initial value																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
R/W																
Initial value																

Meaning of item and sign

Address

Address (base address + offset address) of the register

Bit

Bit number of the register

Name

Bit field name of the register

R/W

Attribution of read/write of each bit field

- R0:Read value is always "0"
- R1: Read value is always "1"
- W0: Write value is always "0", and write access of "1" is ignored
- W1: Write value is always "1", and write access of "0" is ignored
- R: Read
- W: Write

Initial value

Each bit field’s value after reset

- 0: Value is "0"
- 1: Value is "1"
- X: Value is undefined

23.5.2. External interrupt enable register (EIENB)

This register is to control masking external interrupt request output.

Address	FFFE_4000$_H$ + 00$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	ENB3	ENB2	ENB1	ENB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0	R0	R0	R0	R/W	R/W	R/W	R/W
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-8	–	Unused bit. Write access is ignored. Read value of these bits is undefined.
7-4	–	Unused bit. Write access is ignored. Read value of these bits is always "0".
3-0	ENB3-0	Masking external interrupt request output is controlled. 0 External interrupt request is disabled 1 External interrupt request is enabled. The interrupt request output corresponding to the bit written "1" is permitted (ENB0 controls INT_A[0] permission), and the request is output to interrupt controller (IRC0.) Although the pin corresponding to the bit written "0" maintains interrupt factor, interrupt is not requested to the controller. These bits are initialized to "0000$_B$" by reset.

23.5.3. External interrupt request register (EIREQ)

This register is to indicate and clear external interrupt request.

Address	FFFE_4000$_H$ + 04$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	–	–	–	–	REQ3	REQ2	REQ1	REQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0	R0	R0	R0	R/W0	R/W0	R/W0	R/W0
Initial value	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0

Bit field No.	Bit field Name	Description
31-8	–	Unused bit. Write access is ignored. Read value of these bits is undefined.
7-4	–	Unused bit. Write access is ignored. Read value of these bits is always "0".
3-0	REQ3-0	External interrupt request is indicated and cleared. 0: At reading: There is no external interrupt request At writing: External interrupt request is cleared 1: At reading: There is external interrupt request At writing: External interrupt request invalid Read value of "1" shows external interrupt is requested. These bits correspond to external interrupt channel as follows. • REQ0: External interrupt 0 (INT_A[0] pin) • REQ1: External interrupt 1 (INT_A[1] pin) • REQ2: External interrupt 2 (INT_A[2] pin) • REQ3: External interrupt 3 (INT_A[3] pin) When "0" is written to these bits, external interrupt request is cleared. Writing "1" is invalid. These bits are initialized to "0000$_B$" by reset.

23.5.4. External interrupt level register (EILVL)

This register is to select input request level detection.

Address	FFFE_4000$_H$ + 08$_H$															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
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
Initial value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	–	–	–	–	–	–	–	–	LVL3[1]	LVL3[0]	LVL2[1]	LVL2[0]	LVL1[1]	LVL1[0]	LVL0[1]	LVL0[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
Initial value	X	X	X	X	X	X	X	X	0	1	0	1	0	1	0	1

Bit field		Description
No.	Name	
31-8	–	Unused bit. Write access is ignored. Read value of these bits is undefined.
7-0	LVL3[1:0] - LVL0[1:0]	Input request level detection of external interrupt is selected. 2 bit is allocated to each external interrupt channel. This is initialized to "01$_B$" by reset. • LVL0[1:0]: External interrupt 0 (INT_A[0] pin) • LVL1[1:0]: External interrupt 1 (INT_A[1] pin) • LVL2[1:0]: External interrupt 2 (INT_A[2] pin) • LVL3[1:0]: External interrupt 3 (INT_A[3] pin)

LVL3-0[1]	LVL3-0[0]	Input request level
0	0	"L" Level
0	1	"H" Level
1	0	Rising edge
1	1	Falling edge

23.6. Operation

External interrupt controller issues request signal to interrupt controller (IRC0) when input request level of external interrupt is input to corresponding channel after setting EIENB and EILVL registers.
If interrupt from this module is higher than interrupt level set in ILM register and it is highest priority as a result of interrupt prioritization occurred in IRQ level decision circuit, IRQ interrupt request is issued to ARM core.

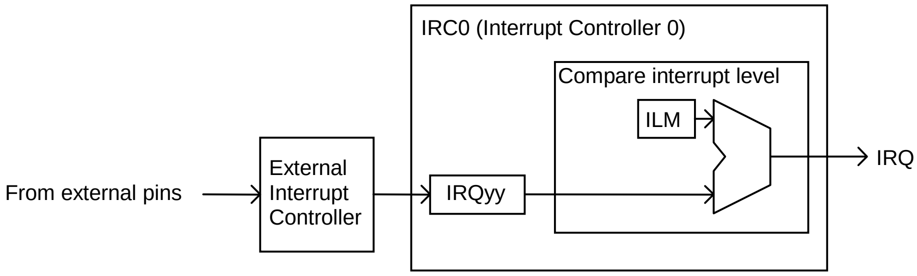


Figure 23-2 Operation of external interrupt

23.7. Operation procedure

External interrupt register setting procedure is as followings.
1. Disable EIENB register related bit
2. Set EILVL register related bit
3. Clear EIREQ register related bit
4. Enable EIENB register related bit

EIENB register must be disabled to set register in the module; moreover, EIREQ register needs to be cleared before EIENB register is enabled. This operation is to prevent accident caused by incidental interrupt source during register setting.

23.8. Instruction for use

This section indicates notice for using external interrupt.

Notice for returning from Stop mode

When external interrupt is used to return from Stop mode, where clock is stopped, set input request level to "H" since "L" level request may cause malfunction. Moreover, the edge request is not able to return from the Stop mode.

24. SD memory controller (SDMC)

Only SD card licensee is disclosed.

FUJITSU