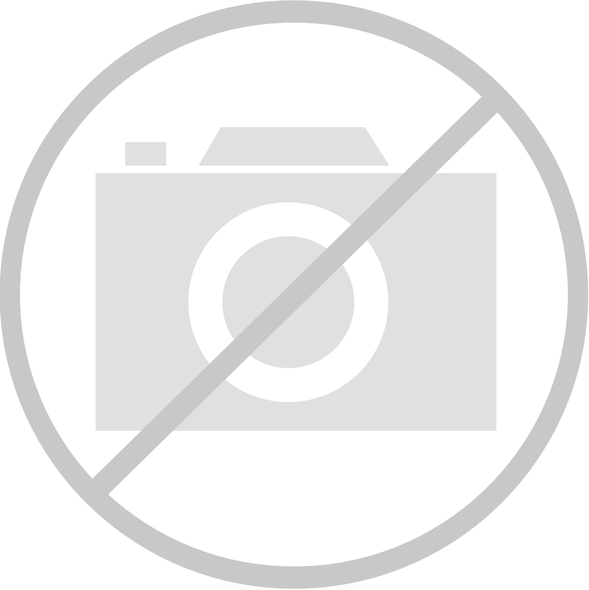
Toshiba Semiconductor and Storage - TMPM366FYFG Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	ARM® Cortex®-M3
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I²C, SIO, UART/USART
Peripherals	DMA, LVD, WDT
Number of I/O	73
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	48K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 12x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/toshiba-semiconductor-and-storage/tmpm366fyfg

Table 1-1 Pin Names and Functions Sorted by Pin (3/7)

Type	Pin No.	Pin Name	Input/ Out-put	Function
Function	29	PC2 SCLK1 A0 TB0OUT $\overline{\text{CTS1}}$	I/O I/O O O I	I/O port Serial channel clock pin Address bus 16-bit timer / event counter output Serial channel handshake input pin
Function/ Control	30	PF0 TB6OUT $\overline{\text{BOOT}}$	O O I	Output port 16-bit timer / event counter output Setting a boot mode: TMPM366FDFG/FYFG/FWFG goes into single boot mode by sampling "Low" at the rising edge of a $\overline{\text{RESET}}$ pin.
Function	31	PF1 $\overline{\text{RD}}$	I/O O	I/O port Read strobe signal
Function	32	PF2 $\overline{\text{WR}}$	I/O O	I/O port Write strobe signal
Function	33	PF3 $\overline{\text{BELL}}$	I/O O	I/O port Byte enable signal as an external 8-bit memory access
Function	34	PF4 $\overline{\text{BELH}}$ INT6 TB5IN0	I/O O I I	I/O port Byte enable signal as an external 8-bit memory access External interrupt pin Inputting 16-bit timer / event counter capture trigger
Function	35	PF5 $\overline{\text{CS1}}$ INT7 TB5IN1	I/O O I I	I/O port Chip select output External interrupt pin Inputting 16-bit timer / event counter capture trigger
Function	36	PF6 $\overline{\text{CS0}}$	I/O O	I/O port Chip select output
Function	37	PF7 ALE	I/O O	I/O port Address latch enable (output disable for noise-reduction can be selected)
PS	38	DVSSA	−	GND pin
PS	39	DVDD3A	−	Power supply pin
Function	40	PA0 D0/AD0	I/O I/O	I/O port Data bus/Address data bus
Function	41	PA1 D1/AD1	I/O I/O	I/O port Data bus/Address data bus
Function	42	PA2 D2/AD2	I/O I/O	I/O port Data bus/Address data bus
Function	43	PA3 D3/AD3	I/O I/O	I/O port Data bus/Address data bus
Function	44	PA4 D4/AD4	I/O I/O	I/O port Data bus/Address data bus
Function	45	PA5 D5/AD5	I/O I/O	I/O port Data bus/Address data bus
Function	46	PA6 D6/AD6	I/O I/O	I/O port Data bus/Address data bus

Table 1-1 Pin Names and Functions Sorted by Pin (4/7)

Type	Pin No.	Pin Name	Input/ Out- put	Function
Function	47	PA7 D7/AD7	I/O I/O	I/O port Data bus/Address data bus
Function	48	PB0 D8/AD8 SP1DO A0	I/O I/O O O	I/O port Data bus/Address data bus SSP DO output Address bus
Function	49	PB1 D9/AD9 SP1DI A1	I/O I/O I O	I/O port Data bus/Address data bus SSP DI input Address bus
Function	50	PB2 D10/AD10 SP1CLK A2	I/O I/O I/O O	I/O port Data bus/Address data bus SSP clock input/ output Address bus
Function	51	PB3 D11/AD11 SP1FSS A3	I/O I/O I/O O	I/O port Data bus/Address data bus SSP FSS input/ output Address bus
Function	52	PB4 D12/AD12 SP2DO A4	I/O I/O O O	I/O port Data bus/Address data bus SSP DO output Address bus
Function	53	PB5 D13/AD13 SP2DI A5	I/O I/O I O	I/O port Data bus/Address bus SSP DI input Address bus
Function	54	PB6 D14/AD14 SP2CLK A6	I/O I/O I/O O	I/O port Data bus/Address data bus SSP clock input/ output Address bus
Function	55	PB7 D15/AD15 SP2FSS A7	I/O I/O I/O O	I/O port Data bus/Address data bus SSP FSS input/ output Address bus
PS	56	DVDD3A	−	Power supply pin
PS	57	DVSSA	−	GND pin
Function	58	PD7 SP0FSS SCOUT	I/O I/O O	I/O port SSP FSS input/ output System clock output
Function	59	PD6 SP0CLK	I/O I/O	I/O port SSP clock input/ output

6.4.2 Internal reset generation

Figure 6-3 shows the internal reset generation (WDMOD<RESCR>="1").

MCU is reset by the overflow of the binary counter. In this case, reset status continues for 32 states. A clock is initialized so that input clock (fsys) is the same as a internal high-speed frequency clock (fosc). This means fsys = fosc.

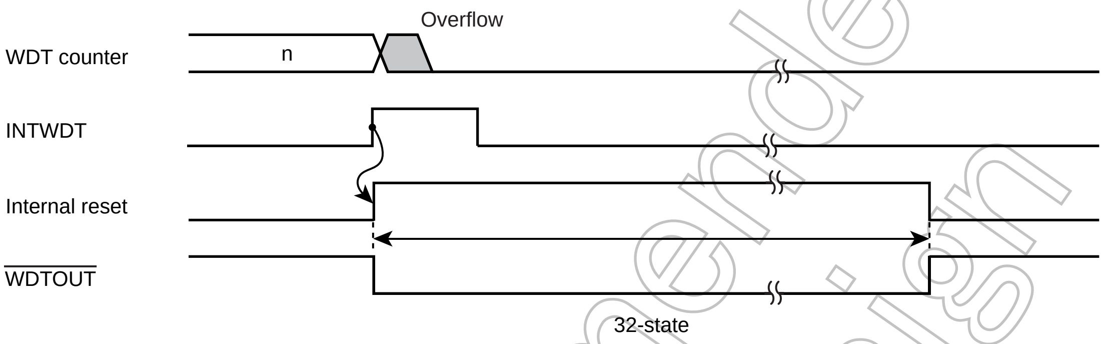


Figure 6-3 Internal reset generation

8.5 Interrupts

This chapter describes routes, sources and required settings of interrupts.

The CPU is notified of interrupt requests by the interrupt signal from each interrupt source.

It sets priority on interrupts and handles an interrupt request with the highest priority.

Interrupt requests for clearing a standby mode are notified to the CPU via the clock generator. Therefore, appropriate settings must be made in the clock generator.

8.5.1 Interrupt Sources

8.5.1.1 Interrupt Route

Figure 8-1 shows an interrupt request route.

The interrupts issued by the peripheral function that is not used to release standby are directly input to the CPU (route1).

The peripheral function interrupts used to release standby (route 2) and interrupts from the external interrupt pin (route 3) are input to the clock generator and are input to the CPU through the logic for releasing standby (route 4 and 5).

If interrupts from the external interrupt pins are not used to release standby, they are directly input to the CPU, not through the logic for standby release (route 6).

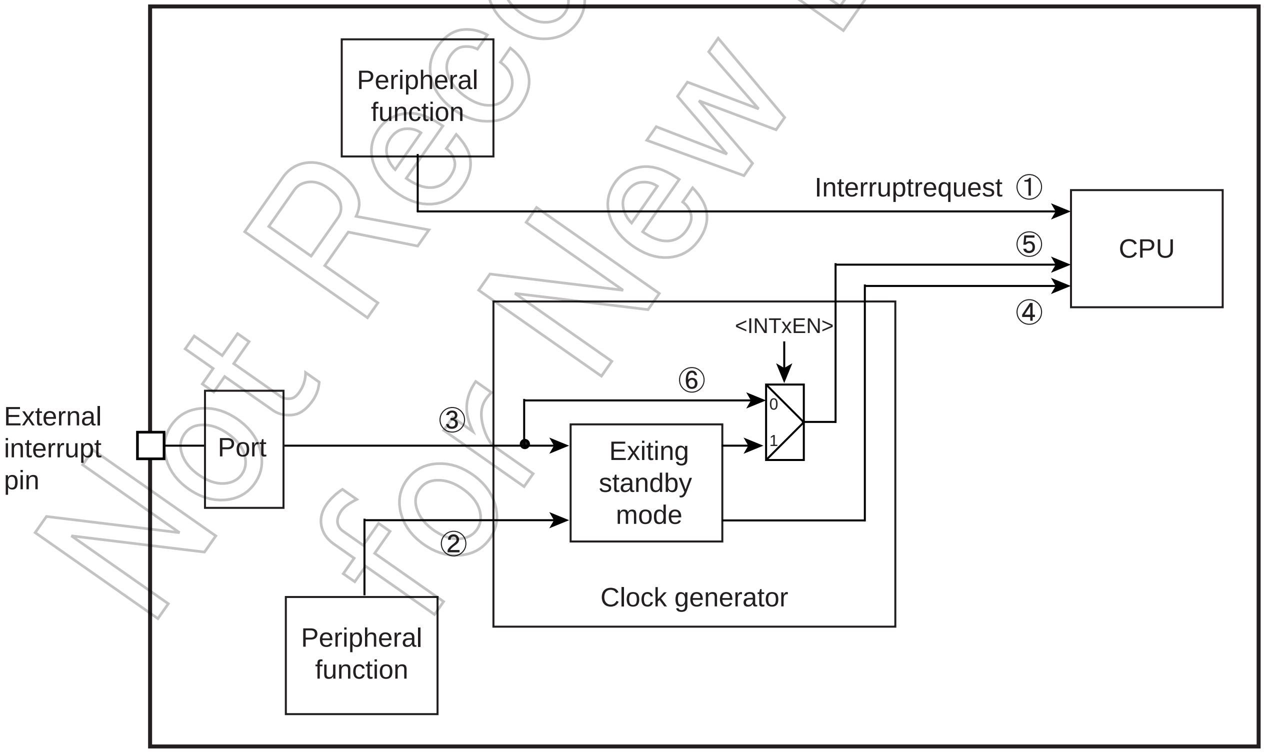


Figure 8-1 Interrupt Route

10.5.11 DMACxSoftSReq (DMAC Software Single Request Register)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
	15	14	13	12	11	10	9	8
bit symbol	SoftSReq15	SoftSReq14	SoftSReq13	SoftSReq12	SoftSReq11	SoftSReq10	SoftSReq9	SoftSReq8
After reset	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SoftSReq7	SoftSReq6	SoftSReq5	SoftSReq4	SoftSReq3	SoftSReq2	SoftSReq1	SoftSReq0
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-16	-	-	Write as zero.
15	SoftSReq15	R/W	DMA single request by software (Request No. [15]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
14	SoftSReq14	R/W	DMA single request by software (Request No. [14]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
13	SoftSReq13	R/W	DMA single request by software (Request No. [13]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
12	SoftSReq12	R/W	DMA single request by software (Request No. [12]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
11	SoftSReq11	R/W	DMA single request by software (Request No. [11]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
10	SoftSReq10	R/W	DMA single request by software (Request No. [10]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
9	SoftSReq9	R/W	DMA single request by software (Request No. [9]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested
8	SoftSReq8	R/W	DMA single request by software (Request No. [8]) Read : 0 :Stopping DMA single transfer 1 : running DMA single transfer Write: 0 : invaild 1 : DMA single requested

11.5.4 Read and Write Recovery Time

If access to external areas occurs consecutively, a dummy cycle can be inserted for recovery time.

A dummy cycle can be inserted in both a read and a write cycle. The dummy cycle insertion setting can be made in the chip select control registers, EXBCSx<WRR[2:0]> (write recovery cycle) and <RDR[2:0]> (read recovery cycle). As for the number of dummy cycles, none, one to six or eight system clocks (internal) can be specified for each channel. Figure 11-16 shows the timing of recovery time insertion.

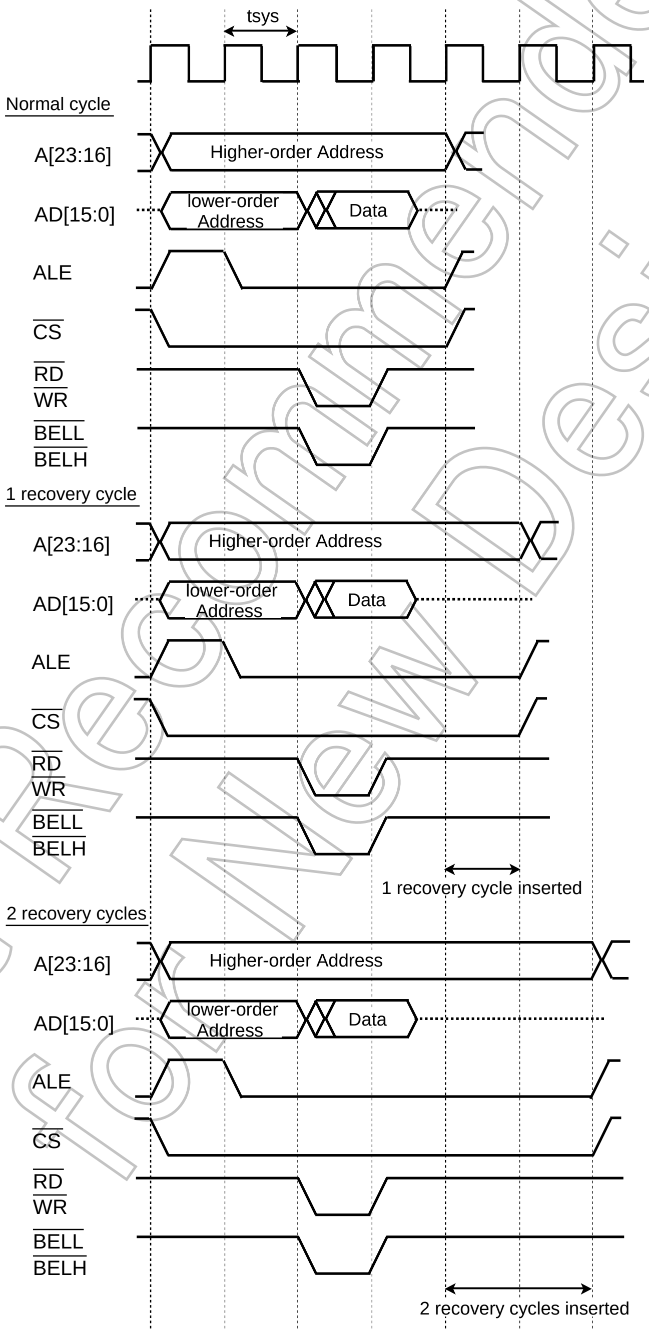


Figure 11-16 Timing of Recovery Time Insertion

11.5.5 Chip select recovery time

If access to external areas occurs consecutively, a dummy cycle can be inserted for recovery time.

11.6.3 EXBASx (External Bus Area and Start Address Configuration Register)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
After reset	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	SA23	SA22	SA21	SA20	SA19	SA18	SA17	SA16
After reset	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
After reset	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	EXAR							
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-24	−	R/W	Always write to 0110_0000
23-16	SA23-SA16	R/W	Chip select Start address (Note) The A[23:16] is specified as start address.
15-8	−	R	Read as 0.
7-0	EXAR[7:0]	R/W	Chip select Address space size The size of address space can be specified nine kind of setting from 64Kbyte up to 16Mbyte. 0000_0000: 16 Mbyte, 0000_0011: 2 Mbyte, 0000_0110: 256 Kbyte, 0000_0001: 8 Mbyte, 0000_0100: 1 Mbyte, 0000_0111: 128 Kbyte, 0000_0010: 4 Mbyte, 0000_0101: 512 Kbyte, 0000_1000: 64 Kbyte, Others: Prohibited

Note: If same address space is specified between CS0 and CS1, the chip selector will be given priority to CS0.

Note: If the access address is exceeded space in 0x6000_0000 to 0x61FF_FFFF, a hard fault error will be generated.

13. USB Device Controller (USBD)

This section describes about USB device controller.

An endpoint is described as EP in this section.

13.1 Outline

1. Conforming to Universal Serial Bus Specification Rev.2.0.
2. Supports Full-Speed (Low-Speed is not supported).
3. USB protocol processing
4. Detects SOF/USB_RESET/SUSPEND/RESUME.
5. Generates and checks packet IDs.
6. Checks CRC5. Generates and checks CRC16.
7. Supports 4 transfer modes (Control/Interrupt /Bulk/ Isochronous).
8. Supports 8 EPs.

Table 13-1 Endpoints

EP0:	Control	64byte × 1 FIFO
EP1:	Control / Interrupt / Bulk / Isochronous (IN)	64byte × 2 FIFO
EP2:	Control / Interrupt / Bulk / Isochronous (OUT)	64byte × 2 FIFO
EP3:	Control / Interrupt / Bulk / Isochronous (IN)	64byte × 2 FIFO
EP4:	Control / Interrupt / Bulk / Isochronous (OUT)	64byte × 2 FIFO
EP5:	Control / Interrupt / Bulk / Isochronous (IN)	64byte × 2 FIFO
EP6:	Control / Interrupt / Bulk / Isochronous (OUT)	64byte × 2 FIFO
EP7:	Control / Interrupt / Bulk / Isochronous (IN)	64byte × 2 FIFO

9. Supports Dual Packets Mode (except for EP 0)
10. Interrupt source signals to the interrupt controller: INTUSB, INTUSBWKUP

13.3 How to connect with the USB bus

The circuit below shows how to connect TMPM366FDFG/FYFG/FWFG with the USB bus.

Input the VBUS signal to USBPON pin to detect the connection of the USB power (VBUS).

The Pull-Up process using the pull-up resistor of the USB-DDP and the in-line damping resistor to the USB-DDM are required. Also, a ON/OFF control using a port needs to be added to the pull-up resistor. The pull-up resistor should be disconnected when voltage is not applied to the VBUS.

If USB-DDP and USB-DDM are unstable, we recommend to add a pull-down resistor to R_1.

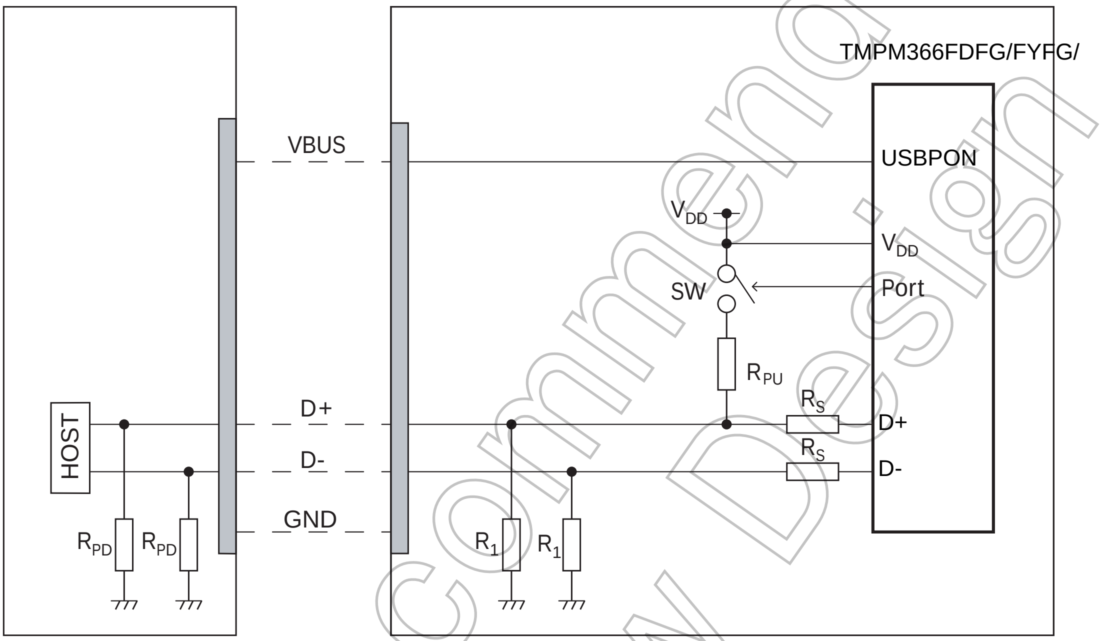


Figure 13-6 Connection example of the USB bus and TMPM366FDFG/FYFG/FWFG.

Note: R_1=500kΩ or higher (recommended value), R_S=33Ω (recommended value), R_{PU}=1.5kΩ (recommended value)

13.4.1.2 UDFSINTSTS (Interrupt Status Register)

	31	30	29	28	27	26	25	24
bit symbol	-	-	int_mw_rerror	int_ powerdetect	-	-	int_dmac_ reg_rd	int_udc2_ reg_rd
After reset	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	int_mr_ahberr	int_mr_ep_ dset	int_mr_end_ add	int_mw_ ahberr	int_mw_ timeout	int_mw_end_ add	int_mw_set_ add	-
After reset	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	int_usb_ reset_end	int_usb_reset	int_suspend_ resume
After reset	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	int_nak	int_ep	int_ep0	int_sof	int_rx_zero	int_status	int_status_ nak	int_setup
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-30	-	R	Read as undefined.
29	int_mw_rerror	R/W	Will be set to 1 when the access to the EP has started Master Write transfer during the setting of common bus access (UDFS2EPxSTS<bus_sel> is 0).(UDFS2EPxSTS<bus_sel> is 0) 0: Not detected 1: EP read error occurred during master write.
28	int_ powerdetect	R/W	When the status of VBUSPOWER input of UDC2AB changed, it is set to "1". 0:No changed 1:Status changed
27-26	-	R	Read as undefined.
25	int_dmac_ reg_rd	R/W	Will be set to 1 when the register access executed by the setting of UDFSDMACRDREQ is completed and the value read to UDFSDMACCRDVL is set. 0: Not detected. 1:Register read completed.
24	int_udc2_reg_ rd	R/W	Will be set to 1 when the UDC2 access executed by the setting of UDFSDMACRDREQ is completed and the value read to UDFSDMACRDLV is set. Also set to 1 when Write access to the internal register of UDC2 is completed. 0: Not detected. 1: Register read/write completed.
23	int_mr_ahberr	R/W	This status will be set to 1 when the AHB error has occurred during the operation of Master Read transfer. After this interrupt has occurred, the Master Read transfer block needs to be reset by the UDFSMSTSET<mr_reset >. 0: Not detected. 1: AHB error occurred.
22	int_mr_ep_dset	R/W	Will be set to 1 when the FIFO of EP for UDC2 Tx to be used for Master Read transfer becomes writable (not full). 0: FIFO is not writable 1: FIFO is writable
21	int_mr_end_ add	R/W	Will be set to 1 when the Master Read transfer has finished. 0: Not detected 1: Master Read transfer finished
20	int_mw_ahberr	R/W	This status will be set to 1 when the AHB error has occurred during the operation of Master Write transfer. After this interrupt has occurred, the Master Write transfer block needs to be reset by UDFSMSTSET<mw_reset>. 0: Not detected. 1: AHB error occurred.

h. 2.5 s after the interrupt is asserted (time required for the signal to stabilize when VBUS is disconnected), check the UDFSPWCTL<pw_detect>. If UDFSPWCTL<pw_detect> is "1", WAKEUP is asserted by resume. If UDFSPWCTL<pw_detect> is "0", WAKEUP is asserted by disconnection of VBUS.

i. If the factor is the resume, perform the sequence written in the "13.5.7.2 Resuming from suspended state (resuming from the USB host)". If the factor is disconnection, perform the sequence below.

j. Zero clear the <phy_suspend> to deassert the PHYSYSPEND output signal.

Set the CGUSBCTL<USBCLKEN> of the clock/mode circuit to "1" to get the CLK_U work.Clear the interrupt factor and <wakeup_en> to deassert the WAKEUP output signal.

k. Set UDFSPWCTL<pw_resetb> using software, initialize the UDC2AB.

13.5.7.4 Remote wakeup from the suspended state

The procedure of remote wakeup from the suspended state and the signal change are shown below.

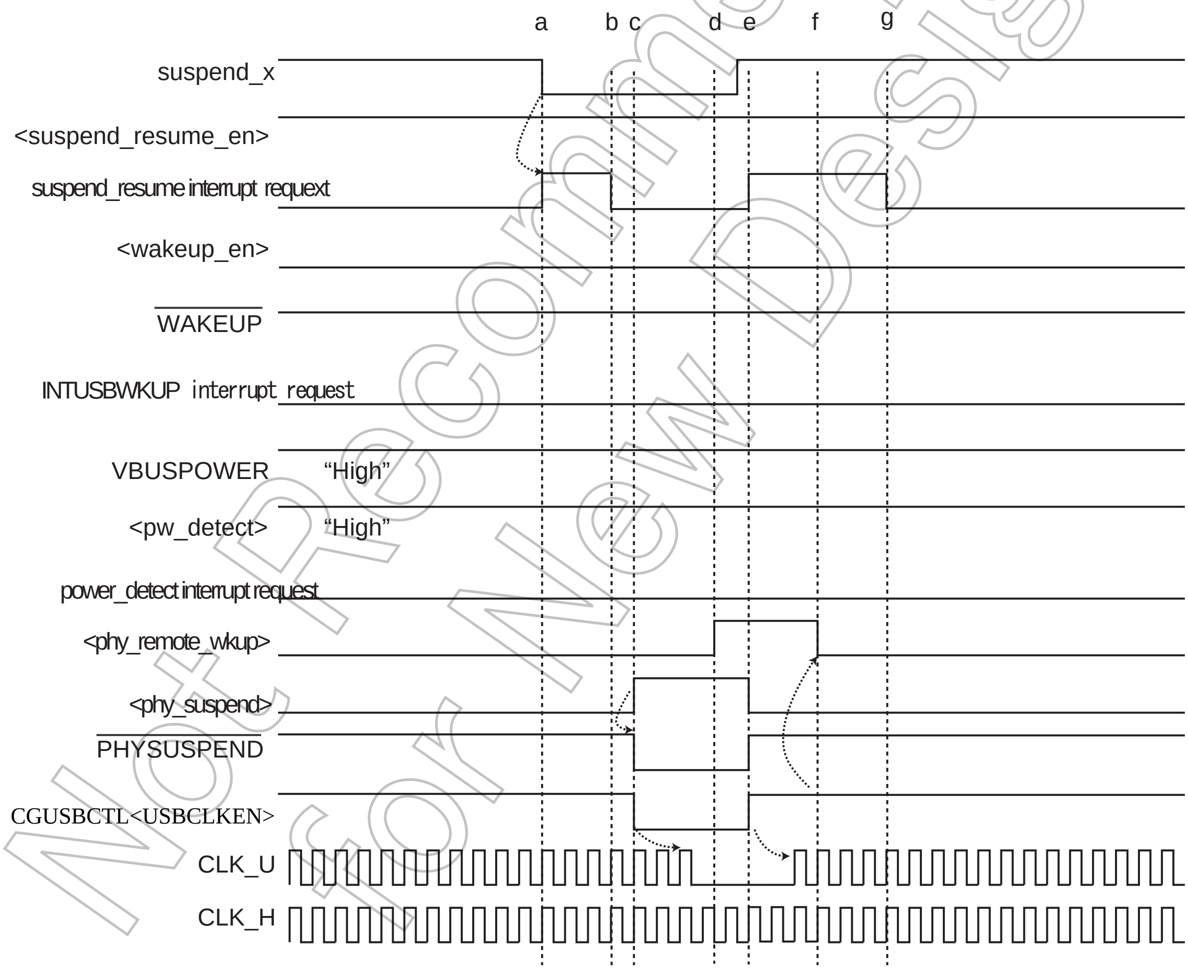


Figure 13-20 Operation of suspend/remote wakeup signals

a. suspend_x of the UDC2 is asserted to 0 by detecting the suspended state on the USB bus and the INTUSB(suspend_resume) interrupt occurs.

b. Clears the interrupt source in the service routine of the INTUSB(suspend_resume) interrupt.

c. Set the UDFSPWCTL<phy_suspend> to "1". PHYSUSPEND output signal is asserted to "0" by setting <phy_suspend> to "1"

13.7 Flow of Control in Transfer of EPs

13.7.1 EP0

EP0 supports Control transfer and is used as device control for enumeration. EP0 supports only Single packet mode.

Control transfers have SETUP-Stage, DATA-Stage and STATUS-Stage

The types of transfer are categorized into the following major types:

- Control-RD transfer
- Control-WR transfer (without DATA-Stag)
- Control-WR transfer (with DATA-Stage)

UDC2 makes control of those three stages by hardware. Flows in each type of transfer are described below.

13.7.1.1 Control-RD transfer

The flow of control in Control-RD transfers is shown below.

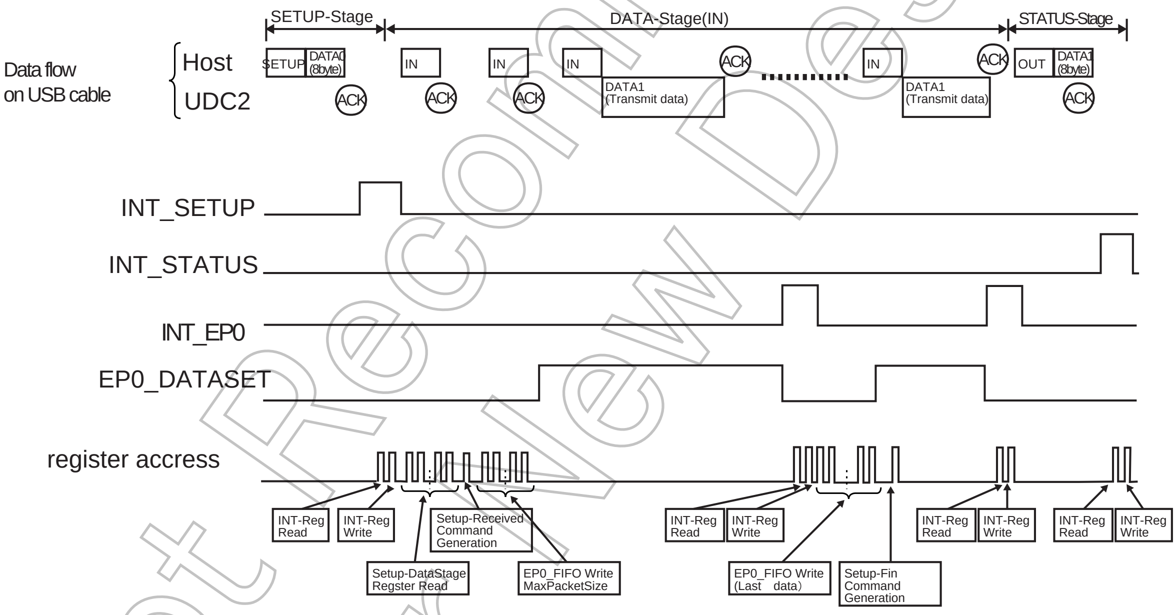


Figure 13-22 Flow of the control in Control-RD transfer

The following description is based on the assumption that the UDFS2EP0MSZ<dset> is set to "EP0_DATASET flag".

(1) SETUP-Stage

UDC2 asserts the INT_SETUP flag when it has received the Setup-Token. This flag can be cleared by writing 1 into the UDFS2INT<i_setup>. In case flags are combined externally, read the UDFS2INT to confirm which flag is asserted and write "1" into the relevant bit.

Then read Setup-Data storage registers (bRequest-bmRequestType, wValue, wIndex, and wLength registers) to determine the request.

15. Asynchronous Serial Channel (UART)

15.1 Overview

This device has the Asynchronous serial channel (UART) with Modem control.

Their features are given in the following.

- Transmit FIFO

  8-bit width/ 32 location deep.
- Receive FIFO

  12-bit width/ 32 location deep.
- Transmit /Receive data format
  - DATA bits: 5,6,7,8 bits can be selected.
  - PARITY : use / no use
  - STOP bit : 1bit / 2 bits
- FIFO ON/OFF
  - ON (FIFO mode)/
  - OFF (characters mode)
- Interrupt
  - Combined interrupt factors are output to interrupt controller.
  - The permission of each interrupt factor is programmable.
- Baud rate generator

  Generates a common transmit and receive internal clock form UART internal reference clock input.

  Supports baud rates of up to 2.95Mbps at fsys = 48MHz.
- DMA
- IrDA 1.0 Function
  - Max data rate : 115.2 kbps (half-duplex).
  - support low power mode
- Control pins

  TXD (IROUT)

  RXD (IRIN)

  $\overline{\text{CTS}}$

  RIN

  $\overline{\text{RTS}}$

  DCD

  DSR

  DTR
- Hardware flow control

  RTS support

  CTS support

16.3.6 SSPxCPSR (Clock prescale register)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
After Reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
After Reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
After Reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
	7	6	5	4	3	2	1	0
bit symbol	CPSDVSR							
After Reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Description
31-8	−	W	Write as "0".
7-0	CPSDVSR[7:0]	R/W	Clock prescale divisor: Set an even number from 2 to 254. Clock prescale divisor: Must be an even number from 2 to 254, depending on the frequency of fsys. The least significant bit always returns zero when read.

17.5.15 Baud Rate Register (SBIxBR0)

The SBIxBR0<I2SBI> register determines if the SBI operates or not when it enters the IDLE mode.

This register must be programmed before executing an instruction to switch to the standby mode.

17.5.16 Software Reset

If the serial bus interface circuit locks up due to external noise, it can be initialized by using a software reset.

Writing "10" followed by "01" to SBIxCR2<SWRST[1:0]> generates a reset signal that initializes the serial bus interface circuit. When writing, set <SBIM[1:0]> to "10"; I2Cbus mode. After a reset, all control registers and status flags are initialized to their reset values. When the serial bus interface is initialized, <SWRST> is automatically cleared to "0".

Note: A software reset causes the SBI operating mode to switch from the I2C mode to the port mode.

18.3.5 ADMOD2 (Mode Control Register 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
After reset	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
After reset	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
After reset	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	HPADCH				ADCH			
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-8	−	R	Read as 0.
7-4	HPADCH[3:0]	R/W	Select analog input channels in highest-priority AD conversion. (See Table 18-1.) (prohibit "1100 to 1111")
3-0	ADCH[3:0]	R/W	Select analog input channels in normal AD conversion. (See Table 18-1.) (prohibit "1100 to 1111")

Table 18-1 Selection of input channels in normal AD conversion or highest-priority AD conversion

<HPADCH[3:0]>	Analog input channels in highest-priority AD conversion	<ADCH[3:0]>	Analog input channels in normal AD conversion
0000	AIN00	0000	AIN00
0001	AIN01	0001	AIN01
0010	AIN02	0010	AIN02
0011	AIN03	0011	AIN03
0100	AIN04	0100	AIN04
0101	AIN05	0101	AIN05
0110	AIN06	0110	AIN06
0111	AIN07	0111	AIN07
1000	AIN08	1000	AIN08
1001	AIN09	1001	AIN09
1010	AIN10	1010	AIN10
1011	AIN11	1011	AIN11

18.4.5 AD Conversion Details

18.4.5.1 Starting AD Conversion

Two types of A/D conversion are supported: normal AD conversion and top-priority AD conversion. Normal AD conversion is activated by setting ADMOD0<ADS> to "1". Highest-priority AD conversion is activated by setting ADMOD0<HPADS> to "1".

Four operation modes are made available to normal AD conversion. In performing normal AD conversion, one of these operation modes must be selected by setting ADMOD3 <REPEAT, SCAN> to an appropriate setting. For highest-priority AD conversion, only one operation mode can be used: fixed channel single conversion mode.

Normal AD conversion can be activated using the H/W activation source selected by ADMOD1<ADHWS>, and highest-priority AD conversion can be activated using the HW activation source selected by ADMOD1<HPADHWS>. If bits of <ADHWS> and <HPADHWS> are "0", normal and highest-priority AD conversions are activated in response to the input of a falling edge through the $\overline{\text{ADTRG}}$ pin. If these bits are "1", normal AD conversion is activated in response to INTCAP50 generated by the 16-bit timer channel 5, and highest-priority AD conversion is activated in response to INTCAP40 generated by the 16-bit timer channel 4.

To permit H/W activation, set ADMOD1 <ADHWE> to "1" for normal AD conversion and set ADMOD1<HPADHWE> to "1" for highest-priority AD conversion.

Software activation is still valid even after H/W activation has been permitted.

Note: When an external trigger is used for the HW activation source of a highest-priority AD conversion, an external trigger cannot be set for activating normal AD conversion H/W start.

18.4.5.2 AD Conversion

When normal AD conversion starts, the AD conversion Busy flag (ADMOD5 <ADBF>) showing that AD conversion is under way is set to "1".

When highest-priority AD conversion starts, the highest-priority AD conversion Busy flag (ADMOD5 <HPADBF>) showing that AD conversion is under way is set to "1".

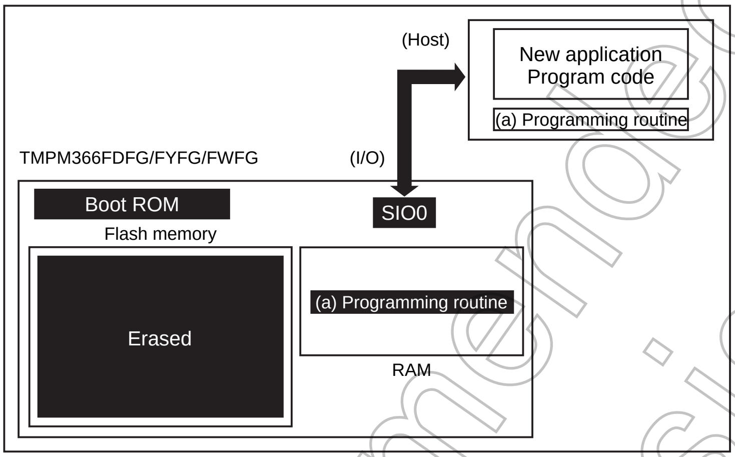
At that time, the value of the Busy flag ADMOD5<ADBF> for normal AD conversion before the start of highest-priority AD conversion is retained.

The value of the conversion completion flag ADMOD5 <EOCF> for normal AD conversion before the start of highest-priority AD conversion is retained.

Note: Normal AD conversion must not be activated when highest-priority AD conversion is under way. If activated when highest-priority AD conversion is under way, the highest-priority AD conversion completion flag cannot be set, and the flag for previous normal A/D conversion cannot be cleared.

(4) Step-4

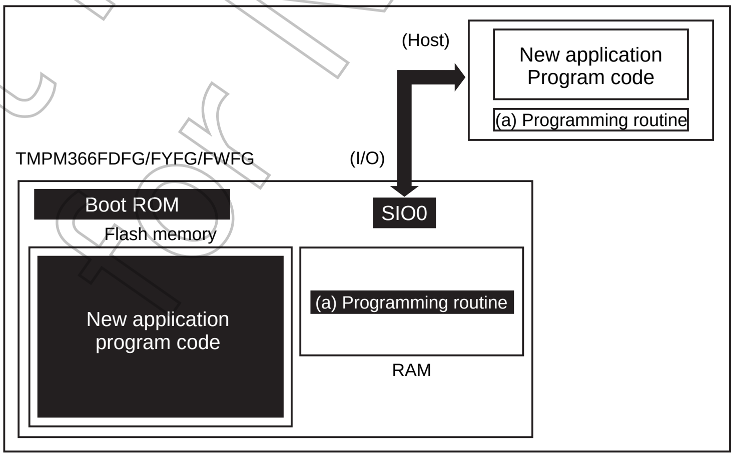
The CPU jumps to the programming routine (a) in the on-chip RAM to erase the flash block containing the old application program code. The Block Erase or Chip Erase command may be used.



(5) Step-5

Next, the programming routine (a) downloads new application program code from the host controller and programs it into the erased flash block. When the programming is completed, the writing or erase protection of that flash block in the user's program area must be set.

In the example below, new program code comes from the same host controller via the same SIO0 channel as for the programming routine. However, once the programming routine has begun to execute, it is free to change the transfer path and the source of the transfer. Create board hardware and a programming routine to suit your particular needs.



(3) Automatic block erase (for each block)

The automatic block erase operation starts when the sixth bus write cycle of the command cycle is completed.

This status of the automatic block erase operation can be checked by monitoring FCFLCS <RDY/BSY> . While no automatic verify operation is performed internally to the device, be sure to read the data to confirm that data has been correctly erased. Any new command sequence is not accepted while it is in an automatic block erase operation. If it is desired to stop operation, use the hardware reset function. In this case, it is necessary to perform the automatic block erase operation again because the data erasing operation has not been normally terminated.

Also, any protected blocks cannot be erased. If an automatic block erase operation has failed, the flash memory is locked in the mode and will not return to the read mode. In this case, execute hardware reset to reset the device.

(4) Automatic programming of protection bits (for each block)

This device is implemented with protection bits. This protection can be set for each block. See Table 19-24 for table of protection bit addresses. This device assigns 1 bit to 1 block as a protection bit. The applicable protection bit is specified by PBA in the seventh bus write cycle. By automatically programming the protection bits, write and/or erase functions can be inhibited (for protection) individually for each block. The protection status of each block can be checked by FCFLCS <BLPRO> to be described later. This status of the automatic programming operation to set protection bits can be checked by monitoring FCFLCS <RDY/BSY> . Any new command sequence is not accepted while automatic programming is in progress to program the protection bits. If it is desired to stop the programming operation, use the hardware reset function. In this case, it is necessary to perform the programming operation again because the protection bits may not have been correctly programmed. If all the protection bits have been programmed, all FCFLCS <BLPRO> are set to "1" indicating that it is in the protected state. This disables subsequent writing and erasing of all blocks.

Note: **Software reset is ineffective in the seventh bus write cycle of the automatic protection bit programming command. FCFLCS <RDY/BSY> turns to "0" after entering the seventh bus write cycle.**

(5) Automatic erasing of protection bits

Different results will be obtained when the automatic protection bit erase command is executed depending on the status of the protection bits and the security bits. It depends on the status of FCFLCS <BLPRO> whether all <BLPRO> are set to "1" or not if FCSECBIT<FCSECBIT> is 0x1. Be sure to check the value of FCFLCS <BLPRO> before executing the automatic protection bit erase command. See the chapter "ROM protection" for details.

- When all the FCFLCS <BLPRO> are set to "1" (all the protection bits are programmed):

  When the automatic protection bit erase command is command written, the flash memory is automatically initialized within the device. When the seventh bus write cycle is completed,the entire area of the flash memory data cells is erased and then the protection bits are erased. This operation can be checked by monitoring FCFLCS <RDY/BSY>. If the automatic operation to erase protection bits is normally terminated, FCFLCS will be set to "0x00000001".While no automatic verify operation is performed internally to the device, be sure to read the data to confirm that it has been correctly erased. For returning to the read mode while the automatic operation after the seventh bus cycle is in progress, it is necessary to use the hardware reset to reset the device. If this is done, it is necessary to check the status of protection bits by FCFLCS <BLPRO> after retuning to the read mode and perform either the automatic protection bit erase, automatic chip erase, or automatic block erase operation, as appropriate.

- When FCFLCS <BLPRO> include "0" (not all the protection bits are programmed):

22.5 12-bit ADC Electrical Characteristics

DVDD3A = DVDD3C = AVDD3 = RVDD3 = AVREFH = 2.7 V to 3.6 V
AVSS = DVSSA = DVSSC = RVSS = AVREFL = 0
Ta = −40 to 85 °C

Parameter		Symbol	Condition	Min.	Typ.	Max	Unit
Analog reference voltage(+)		AVREFH	−	2.7	3.3	3.6	V
Analog input voltage		VAIN	−	AVSS	−	VREFH	V
Power supply current of analog reference voltage	AD conversion	IREF	DVSSA = DVSSC = AVSS	−	1.5	2.0	mA
	Non-AD conversion			−	0.02	0.1	μA
INL error		−	AIN resistance ≤ 600 Ω AIN load capacitance ≥ 0.1 μF Conversion time ≥ 1.0 μs	−	4	6	LSB
DNL error				−	2	6	
Offset error				−	3	6	
Full-scale error				−	3	6	
INL error		−	AIN resistance ≤ 600 Ω AIN load capacitance ≥ 33 pF Conversion time ≥ 1.66 μs	−	3	6	
DNL error				−	2	6	
Offset error				−	4	6	
Full-scale error				−	2	6	
Conversion time		Tconv	−	1.0	−	10	μs

Note 1: **1LSB = (AVREFH − AVSS)/4096 [V]**

Note 2: **Peripheral functions are disable.**