Toshiba Semiconductor and Storage - TMPM332FWUG(C) 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	40MHz
Connectivity	I²C, SIO, UART/USART
Peripherals	POR, WDT
Number of I/O	44
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-20°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/toshiba-semiconductor-and-storage/tmpm332fwug-c

8.2.2.3 PBDATA (Port B data register)
8.2.2.4 PBCR (Port B output control register)
8.2.2.5 PBFR1 (Port B function register 1)
8.2.2.6 PBPUP (Port B pull-up control register)
8.2.2.7 PBIE (Port B input control register)
8.2.3 Port D (PD0 to PD7)....110
8.2.3.1 Port D Circuit Type
8.2.3.2 Port D Register
8.2.3.3 PDDATA (Port D data register)
8.2.3.4 PDFR1 (Port D function register 1)
8.2.3.5 PDPUP (Port D pull-up control register)
8.2.3.6 PDIE (Port D input control register)
8.2.4 Port E (PE0 to PE6)....113
8.2.4.1 Port E Circuit Type
8.2.4.2 Port E Register
8.2.4.3 PEDATA (Port E data register)
8.2.4.4 PECR (Port E output control register)
8.2.4.5 PEFR1(Port E function register 1)
8.2.4.6 PEFR2(Port E function register 2)
8.2.4.7 PEOD (Port E open drain control register)
8.2.4.8 PEPUP (Port E pull-up control register)
8.2.4.9 PEIE (Port E input control register)
8.2.5 Port F (PF4 to PF6)....118
8.2.5.1 Port F Circuit Type
8.2.5.2 Port F Register
8.2.5.3 PFDATA (Port F data register)
8.2.5.4 PFCR (Port F output control register)
8.2.5.5 PFFR1(Port F function register 1)
8.2.5.6 PFOD (Port F open drain control register)
8.2.5.7 PFPUP (Port F pull-up control register)
8.2.5.8 PFIE (Port F input control register)
8.2.6 Port G (PG0 to PG3)....123
8.2.6.1 Port G Circuit Type
8.2.6.2 Port G Register
8.2.6.3 PGDATA (Port G data register)
8.2.6.4 PGCR (Port G output control register)
8.2.6.5 PGFR1(Port G function register 1)
8.2.6.6 PGOD (Port G open drain control register)
8.2.6.7 PGPUP (Port G pull-up control register)
8.2.6.8 PGIE (Port G input control register)
8.2.7 Port H (PH0 to PH3)....128
8.2.7.1 Port H Circuit Type
8.2.7.2 Port H Register
8.2.7.3 PHDATA (Port H data register)
8.2.7.4 PHCR (Port H output control register)
8.2.7.5 PHFR1(Port H function register 1)
8.2.7.6 PHPUP (Port H pull-up control register)
8.2.7.7 PHIE (Port H input control register)
8.2.8 Port I (PI0 to PI5)....132
8.2.8.1 Port I Circuit Type
8.2.8.2 Port I Register
8.2.8.3 PIDATA (Port I data register)
8.2.8.4 PICR (Port I output control register)
8.2.8.5 PIFR1(Port I function register 1)
8.2.8.6 PIPUP (Port I pull-up control register)
8.2.8.7 PIIE (Port I input control register)
8.2.9 Port J (PJ0 to PJ4)....136
8.2.9.1 Port J Circuit Type
8.2.9.2 Port J Register
8.2.9.3 PJDATA (Port J data register)
8.2.9.4 PJCR (Port J output control register)
8.2.9.5 PJFR1(Port J function register 1)
8.2.9.6 PJPUP (Port J pull-up control register)
8.2.9.7 PJIE (Port J input control register)
8.2.10 Port K (PK0 to PK1)....140
8.2.10.1 Port K Circuit Type
8.2.10.2 Port K Register
8.2.10.3 PKDATA (Port K data register)
8.2.10.4 PKCR (Port K output control register)
8.2.10.5 PKFR1(Port K function register 1)
8.2.10.6 PKFR2(Port K function register 2)
8.2.10.7 PKPUP (Port K pull-up control register)
8.2.10.8 PKIE (Port K input control register)
8.3 Block Diagrams of Ports....145
8.3.1 Port Types....145

19.6 AC Electrical Characteristics.....471
19.6.1 AC measurement condition.....471
19.6.2 Serial Channel (SIO/ UART).....471
19.6.2.1 I/O Interface mode
19.6.3 Serial Bus Interface(I2C / SIO).....473
19.6.3.1 I2C Mode
19.6.3.2 Clock-Synchronous 8-Bit SIO mode
19.6.4 Event Counter.....475
19.6.5 Capture.....475
19.6.6 External Interrupt.....475
19.6.7 NMI.....475
19.6.8 SCOUT Pin AC Characteristic.....476
19.6.9 Debug Communication.....476
19.6.10 ETM Trace.....477
19.7 Flash Characteristics.....477
19.7.1 Rewriting.....477
19.8 Recommended Oscillation Circuit.....478
19.8.1 Ceramic oscillator.....478
19.8.2 Crystal oscillator.....478
19.9 Handling Precaution.....479
19.9.1 Solderability.....479
19.9.2 Power-on sequence.....479

20. Port Section Equivalent Circuit Schematic

20.1 PA0, PE1 to 3, PE5 to 6, PF4 to 6, PG0 to 3, PH0 to 3, PJ0 to 3.....481
20.2 PA1.....481
20.3 PA2 to 3, PB0, PE0, PE4, PI0 to 5, PJ4, PK1.....482
20.4 PD4 to 7.....482
20.5 PD0 to 3.....482
20.6 PK0.....483
20.7 NMI, MODE.....483
20.8 RESET.....483
20.9 X1, X2.....484
20.10 XT1, XT2.....484
20.11 VREFH, AVSS.....484

21. Package Dimensions

6.6.7 Warm-up

Mode transition may require the warm-up so that the internal oscillator provides stable oscillation.

In the mode transition from STOP to the NORMAL/ SLOW or from SLEEP to NORMAL, the warm-up counter is activated automatically. And then the system clock output is started after the elapse of configured warm-up time. It is necessary to select a oscillator to be used for warm-up in the CGOSCCR<WUPSEL> and to set a warm-up time in the CGOSCCR<WUPT[2:0]> before executing the instruction to enter the STOP/ SLEEP mode.

Note: **In STOP/ SLEEP modes, the PLL is disabled. When returning from these modes, configure the warm-up time in consideration of the stability time of the PLL and the internal oscillator. It takes approximately 200μs for the PLL to be stabilized.**

In the transition from NORMAL to SLOW/ SLEEP, the warm-up is required so that the internal oscillator to stabilize if the low-speed clock is disabled. Enable the low-speed clock and then activate the warm-up by software.

In the transition from SLOW to NORMAL when the high-speed clock is disabled, enable the high-speed clock and then activate the warm-up.

Table 6-9 shows whether the warm-up setting of each mode transition is required or not.

Table 6-9 Warm-up setting in mode transition

Mode transition	Warm-up setting
NORMAL → IDLE	Not required
NORMAL → SLEEP	(Note1)
NORMAL → SLOW	(Note 1)
NORMAL → STOP	Not required
IDLE → NORMAL	Not required
SLEEP → NORMAL	Auto-warm-up
SLEEP → SLOW	Not required
SLOW → NORMAL	(Note 2)
SLOW → SLEEP	Not required
SLOW → STOP	Not required
STOP → NORMAL	Auto-warm-up (Note 3)
STOP → SLOW	Auto-warm-up

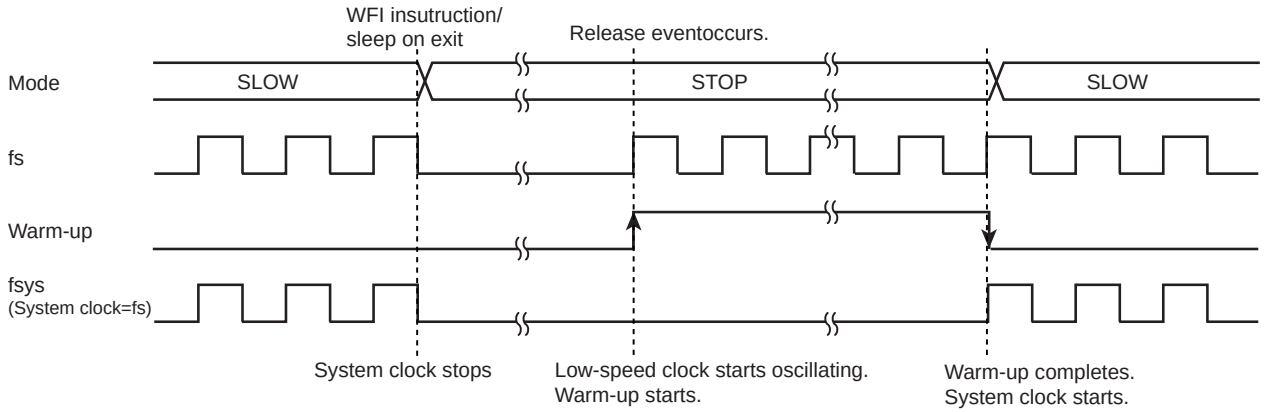
Note 1: **If the low-speed clock is disabled, enable the low-speed clock and then activate the warm-up by software.**

Note 2: **If the high-speed clock is disabled, enable the high-speed clock and then activate the warm-up by software.**

Note 3: **Returning to NORMAL mode by reset does not induce the automatic warm-up. Keep the reset signal valid until the oscillator operation becomes stable.**

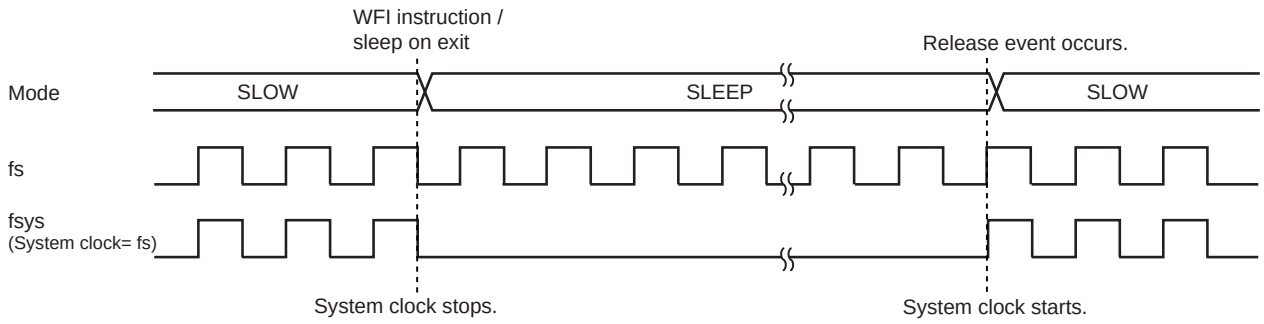
6.6.8.3 Transition of operation modes: SLOW → STOP → SLOW

The warm-up is activated automatically. It is necessary to set the warm-up time before entering the STOP mode.



6.6.8.4 Transition of operation modes: SLOW → SLEEP → SLOW

The low-speed clock continues oscillation in the SLEEP mode. There is no need to make a warm-up setting.



7.3 Non-Maskable Interrupts (NMI)

Non-maskable interrupts are generated from the following two sources.

Use the NMI Flag (CGNMIFLG) Register of the clock generator to identify the source of a non-maskable interrupt.

- External $\overline{\text{NMI}}$ pin

  A non-maskable interrupt is generated when an external $\overline{\text{NMI}}$ pin changes from "High" to "Low".

- Non-maskable interrupt by WDT

  The watchdog timer (WDT) has a non-maskable interrupt generating feature. For details, see the chapter on the WDT.

7.4 SysTick

SysTick provides interrupt features using the CPU's system timer.

When you set a value in the SysTick Reload Value Register and enable the SysTick features in the SysTick Control and Status Register, the counter loads with the value set in the Reload Value Register and begins counting down.When the counter reaches "0", a SysTick exception occurs.You may be pending exceptions and use a flag to know when the timer reaches "0".

The SysTick Calibration Value Register holds a reload value for counting 10 ms with the system timer. The count clock frequency varies with each product, and so the value set in the SysTick Calibration Value Register also varies with each product.

Note: **In this product, the system timer counts based on a clock obtained by dividing the clock input from the X1 pin by 32.The SysTick Calibration Value Register is set to 0x9C4, which provides 10 ms timing when the clock input from X1 is 8 MHz.**

7.5.2.4 Detection by CPU

The CPU detects an interrupt request with the highest priority.

7.5.2.5 CPU processing

On detecting an interrupt, the CPU pushes the contents of PC, PSR, r0-r3, r12 and LR to the stack then enter the ISR.

7.5.2.6 Interrupt Service Routine (ISR)

An ISR requires specific programming according to the application to be used. This section describes what is recommended at the service routine programming and how the source is cleared.

(1) Pushing during ISR

An ISR normally pushes register contents to the stack and handles an interrupt as required. The Cortex-M3 core automatically pushes the contents of PC, PSR, r0-r3, r12 and LR to the stack. No extra programming is required for them.

Push the contents of other registers if needed.

Interrupt requests with higher priority and exceptions such as NMI are accepted even when an ISR is being executed. We recommend you to push the contents of general-purpose registers that might be rewritten.

(2) Clearing an interrupt source

If an interrupt source is used for clearing a standby mode, each interrupt request must be cleared with the CG Interrupt Request Clear (CGICRCG) Register.

If an interrupt source is set as level-sensitive, an interrupt request continues to exist until it is cleared at its source. Therefore, the interrupt source must be cleared. Clearing the interrupt source automatically clears the interrupt request signal from the clock generator.

If an interrupt is set as edge-sensitive, clear an interrupt request by setting the corresponding value in the CGICRCG register. When an active edge occurs again, a new interrupt request will be detected.

7.6.3.4 CGIMCGD(CG Interrupt Mode Control Register D)

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

Bit	Bit Symbol	Type	Function
31	−	R	Read as 0.
30-28	−	R/W	Write any value.
27-25	−	R	Read as 0.
24	−	R/W	Write as 0.
23	−	R	Read as 0.
22-20	−	R/W	Write any value.
19-17	−	R	Read as 0.
16	−	R/W	Write as 0.
15	−	R	Read as 0.
14-12	−	R/W	Write any value.
11-9	−	R	Read as 0.
8	−	R/W	Write as 0.
7	−	R	Read as 0.
6-4	EMCGC[2:0]	R/W	active level setting of INTCECTX standby clear request. **Set it as shown below.** 011: Rising edge
3-2	EMSTC[1:0]	R	active level of INTCECTX standby clear request. 00: − 01: Rising edge 10: Falling edge 11: Both edges
1	−	R	Reads as undefined.
0	INTCEN	R/W	INTCECTX Clear input 0:Disable 1: Enable

Note 1: **<EMSTx> is effective only when <EMCGx[2:0]> is set to "100" for both rising and falling edge. The active level used for the reset of standby can be checked by referring <EMSTx>. If interrupts are cleared with the CGICRCG register, <EMSTx> is also cleared.**

Note 2: **Please specify the bit for the edge first and then specify the bit for the <INTxEN>. Setting them simultaneously is prohibited.**

8.2.3 Port D (PD0 to PD7)

The port D is an 8-bit input port. Besides the general-purpose input function, the port D receives an analog input of the AD converter and a 16-bit timer input.

Reset initializes all bits of the port D as general-purpose input ports with input and pull-up disabled.

Set the PDFR1 and PDIE when you use the port D as input pins of the 16-bit timer.

To use the Port D as an analog input of the AD converter, disable input on PDIE and disable pull-up on PDPUP.

Note: **Unless you use all the bits of port D as analog input pins, conversion accuracy may be reduced. Be sure to verify that this causes no problem on your system.**

8.2.3.1 Port D Circuit Type

	7	6	5	4	3	2	1	0
Type	T17	T17	T17	T17	T18	T18	T18	T18

8.2.3.2 Port D Register

Base Address = 0x4000_00C0

Register name		Address (Base+)
Port D data register	PDDATA	0x0000
Port D function register 1	PDFR1	0x0008
Port D pull-up control register	PDPUP	0x002C
Port D input control register	PDIE	0x0038

8.2.6 Port G (PG0 to PG3)

The port G is a general-purpose, 4-bit input/output port. For this port, inputs and outputs can be specified in units of bits. Besides the general-purpose port function, the port G performs the functions of the serial bus interface and the external interrupt input.

Reset initializes all bits of the port G as general-purpose ports with input, output and pull-up disabled.

To use the external interrupt input for releasing STOP mode, select function in the PGFR register and enable input in the PGIE register.

These settings enable the interrupt input even if the CGSTBYCR<DRVE> bit in the clock/mode control block is set to stop driving of pins during STOP mode.

Note: **In modes other than STOP mode, interrupt input is enabled regardless of the PGFR register setting if input is enabled in PGIE. Make sure to disable unused interrupts when programming the device.**

8.2.6.1 Port G Circuit Type

	7	6	5	4	3	2	1	0
Type	−	−	−	−	T8	T13	T13	T13

8.2.6.2 Port G Register

Base Address = 0x4000_0180

Register name		Address (Base+)
Port G data register	PGDATA	0x0000
Port G output control register	PGCR	0x0004
Port G function register 1	PGFR1	0x0008
Reserved	-	0x0010
Port G open drain control register	PGOD	0x0028
Port G pull-up control register	PGPUP	0x002C
Port G input control register	PGIE	0x0038

Note: **Access to the "reserved" areas is prohibited.**

8.2.6.8 PGIE (Port G input control 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	-	-	-	-	-	-	-	-
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	-	-	-	-	PG3IE	PG2IE	PG1IE	PG0IE
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-8	−	R	Read as 0.
7-4	−	R/W	Write 0.
3-0	PG3IE-PG0IE	R/W	Input 0: Disable 1: Enable

8.2.9.5 PJFR1(Port J function register 1)

	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	-	-	-	PJ4F1	PJ3F1	PJ2F1	PJ1F1	PJ0F1
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-8	−	R	Read as 0.
7-5	−	R/W	Write 0.
4	PJ4F1	R/W	0: PORT 1: TB6OUT
3	PJ3F1	R/W	0: PORT 1: INT3
2	PJ2F1	R/W	0: PORT 1: INT2
1	PJ1F1	R/W	0: PORT 1: INT1
0	PJ0F1	R/W	0: PORT 1: INT0

8.4 Appendix (Port setting List)

The following table shows the register setting for each function.

Initialization of the ports where the [・] does not exist in the "After reset" field is set to "0" for all register settings.

Setting for the bit "x" can be arbitrarily-specified.

8.4.1 Port A Setting

Table 8-5 Port Setting List (Port A)

Pin	Port Type	Function	After reset	PACR	PAFR1	PAPUP	PAPDN	PAIE
PA0	T12	Input Port		0	0	x	0	1
		Output Port		1	0	x	0	0
		SWDIO(I/O)	・	1	1	1	0	1
PA1	T6	Input Port		0	0	0	x	1
		Output Port		1	0	0	x	0
		SWCLK (Input)	・	0	1	0	1	1
PA2	T9	Input Port		0	0	x	0	1
		Output Port		1	0	x	0	0
		TRACECLK (Output)		1	1	x	0	0
PA3	T9	Input Port		0	0	x	0	1
		Output Port		1	0	x	0	0
		TRACEDATA0(Output)		1	1	x	0	0

8.4.2 Port B Setting

Table 8-6 Port Setting List (Port B)

Pin	Port Type	Function	After re-set	PBCR	PBFR1	PBPUP	PBIE
PB0	T11	Input Port		0	0	x	1
		Output Port		1	0	x	0
		SWV (Output)	・	1	1	0	0

8.4.9 Port J Setting

Table 8-13 Port Setting List (Port J)

pin	Port Type	Function	After re-set	PJCR	PJFR1	PJPUP	PJIE
PJ0	T7	Input Port		0	0	x	1
		Output Port		1	0	x	0
		INT0(Input)		0	1	x	1
PJ1	T7	Input Port		0	0	x	1
		Output Port		1	0	x	0
		INT1(Input)		0	1	x	1
PJ2	T7	Input Port		0	0	x	1
		Output Port		1	0	x	0
		INT2(Input)		0	1	x	1
PJ3	T7	Input Port		0	0	x	1
		Output Port		1	0	x	0
		INT3(Input)		0	1	x	1
PJ4	T9	Input Port		0	0	x	1
		Output Port		1	0	x	0
		TB6OUT(Output)		1	1	x	0

8.4.10 Port K Setting

Table 8-14 Port Setting List (Port K)

Pin	Port Type	Function	After re-set	PKCR	PKFR1	PKFR2	PKPUP	PKIE
PK0	T14	Input Port		0	0	0	0	1
		Output Port		1	0	0	0	0
		CEC (Input/Output)		1	1	0	0	1
PK1	T15	Input Port		0	0	0	x	1
		Output Port		1	0	0	x	0
		SCOUT (Output)		1	1	0	x	0
		$\overline{\text{ALARM}}$ (Output)		1	0	1	x	0

Note: **PK0 is an N-ch open drain port.**

9.2 Differences in the Specifications

TMPM332FWUG contains 10-channel of TMRB.

Each channel functions independently and the channels operate in the same way except for the differences in their specification as shown in Table 9-1.

Some of the channels can put the capture trigger and the synchronous start trigger on other channels.

1. The flip-flop output of TMRB 7 through TMRB 9 can be used as the capture trigger of other channels.
   - TB7OUT → available for TMRB0 through TMRB1
   - TB8OUT → available for TMRB2 through TMRB4
   - TB9OUT → available for TMRB5 through TMRB6
2. The start trigger of the timer synchronous mode (with TBxRUN)
   - TMRB0 → can start TMRB0 through TMRB3 synchronously
   - TMRB4 → can start TMRB4 through TMRB7 synchronously

Table 9-1 Differences in the Specifications of TMRB Modules

Specification / Channel	External pins				Trigger function between timers		Interrupt	
	External clock/ capture trigger input pins		Timer flip-flop output pin		Capture trigger	Synchronous start trigger channel	Capture interrupt	TMRB interrupt
	Signal	Port (Pin number)	Signal	Port (Pin number)				
TMRB0	TB0IN0 TB0IN1	PH0 (20) PH1 (21)	TB0OUT	PI0 (24)	TB7OUT	−	INTCAP00 INTCAP01	INTTB0
TMRB1	TB1IN0 TB1IN1	PH2 (22) PH3 (23)	TB1OUT	PI1 (25)	TB7OUT	TMRB0	INTCAP10 INTCAP11	INTTB1
TMRB2	−	−	TB2OUT	PI2 (26)	TB8OUT	TMRB0	−	INTTB2
TMRB3	−	−	TB3OUT	PI3 (30)	TB8OUT	TMRB0	−	INTTB3
TMRB4	−	−	TB4OUT	PI4 (34)	TB8OUT	−	−	INTTB4
TMRB5	TB5IN0 TB5IN1	PD0 (59) PD1 (60)	TB5OUT	PI5 (35)	TB9OUT	TMRB4	INTCAP50 INTCAP51	INTTB5
TMRB6	TB6IN0 TB6IN1	PD2 (61) PD3 (62)	TB6OUT	PJ4 (57)	TB9OUT	TMRB4	INTCAP60 INTCAP61	INTTB6
TMRB7	−	−	−	−	−	TMRB4	−	INTTB7
TMRB8	−	−	−	−	−	−	−	INTTB8
TMRB9	−	−	−	−	−	−	−	INTTB9

10. Serial Channel (SIO/UART)

10.1 Overview

This device has two mode for the serial channel, one is the synchronous communication mode (I/O interface mode), and the other is the asynchronous communication mode (UART mode).

Their features are given in the following.

- Transfer Clock
  - Dividing by the prescaler, from the peripheral clock (ΦT0) frequency into 1/2, 1/8, 1/32, 1/128.
  - Make it possible to divide from the prescaler output clock frequency into 1-16.
  - Make it possible to divide from the prescaler output clock frequency into 1, N+m/16 (N=2-15, m=1-15), 16. (only UART mode)
  - The usable system clock (only UART mode).
- Double Buffer /FIFO

  The usable double buffer function, and the usable FIFO buffers of transmit and receive in all for maximum 4-byte.
- I/O Interface Mode
  - Transfer Mode: the half duplex (transmit/receive), the full duplex
  - Clock: Output (fixed rising edge) /Input (selectable rising/falling edge)
  - Make it possible to specify the interval time of continuous transmission.
- UART Mode
  - Data length: 7 bits, 8bits, 9bits
  - Add parity bit (to be against 9bits data length)
  - Serial links to use wake-up function
  - Handshaking function with $\overline{\text{CTS}}$ pin

In the following explanation, "x" represents channel number.

10.2 Difference in the Specifications of SIO Modules

TMPM332FWUG has two SIO channels.

Each channel functions independently. The used pins and interrupt in each channel are collected in the following.

Table 10-1 Difference in the Specifications of SIO Modules

		Channel 0	Channel 1
Pin name	TXD	PE0(11pin)	PE4(14pin)
	RXD	PE1(12pin)	PE5(15pin)
	$\overline{\text{CTS}}$/SLCK	PE2(13pin)	PE6(16pin)
Interrupt	Receive Interrupt	INTRX0	INTRX1
	Transmit Interrupt	INTTX0	INTTX1

10.4.2 SCxEN (Enable 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	-	-	-	-	-	-	-	-
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	-	-	-	-	-	-	-	SIOE
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-1	−	R	Read as 0.
0	SIOE	R/W	SIO operation 0: Disabled 1: Enabled Specified the SIO operation. To use the SIO, set <SIOE> = "1". When the operation is disabled, no clock is supplied to the other registers in the SIO module. This can reduce the power consumption. If the SIO operation is executed and then disabled, the settings will be maintained in each register except for SCxTFC<TIL>.

11.7.5 SBIxSR (Status Register)

This register serves as SBIxCR2 by writing to it.

	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	-	-	-	-	SIOF	SEF	-	-
After reset	1(Note 1)	1(Note 1)	1(Note 1)	1(Note 1)	0	0	1(Note 1)	1(Note 1)

Bit	Bit Symbol	Type	Function
31-8	-	R	Read as 0.
7-4	-	R	Read as 1.(Note 1)
3	SIOF	R	Serial transfer status monitor. 0: Completed 1: In progress
2	SEF	R	Shift operation status monitor 0: Completed. 1: In progress
1-0	-	R	Read as 1. (Note 1)

Note: **In this document, the value written in the column "after reset" is the value after setting the SIO mode in the initial state.**

13.3.10 RMCRCR4(Receive Control Register 4)

	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	RMCPO	-	-	-	RMCNC			
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-8	−	R	Read as 0.
7	RMCPO	R/W	Remote control input signal 0: Not reversed 1: Reversed
6-4	−	R	Read as 0.
3-0	RMCNC[3:0]	R/W	Specifies noise cancellation time. 0000: No cancellation 0001 to 1111: cancellation Calculating formula of noise cancellation time: <RMCNC> × 1/fs [s]

(4) Settings of Reception Completion

To complete data reception, settings of detecting the maximum data bit cycle and excess low width are required. If multiple factors are specified, reception is completed by the factor detected first. Make sure to configure the reception completion settings.

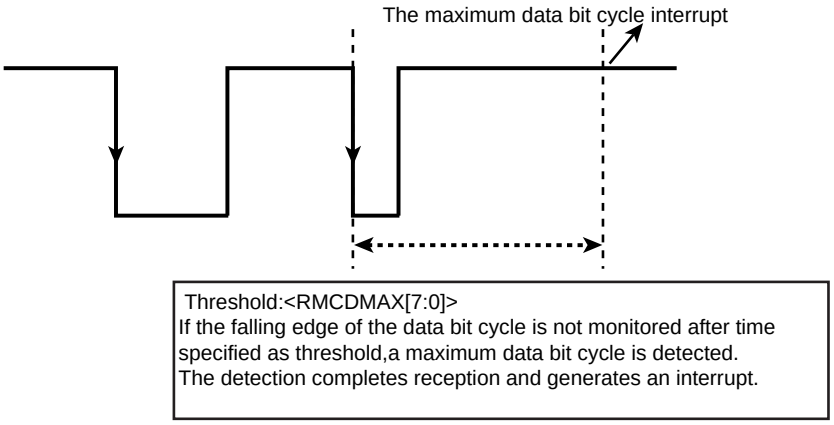
1. Completed by a maximum data bit cycle

   To complete reception by detecting a maximum data bit cycle, you need to configure the RMCRCR2 <RMCDMAX[7:0]> bits.

   If the falling edge of the data bit cycle isn't monitored after time specified as threshold in the <RMCDMAX[7:0]> bits, a maximum data bit cycle is detected.

   The detection completes reception and generates an interrupt.

   After interrupt inputs generated, RMCRSTAT< RMCDMAXIF > bit is set to "1".

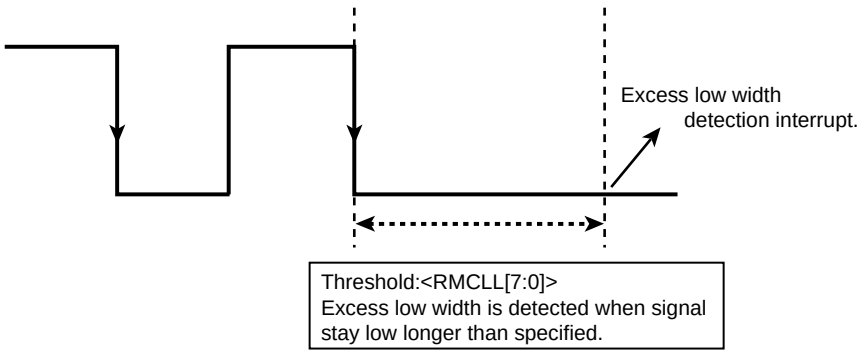


2. Completed by excess low width

   To complete reception by detecting the low width, you need to configure the RMCRCR2 <RMCLL[7:0]> bits.

   After the falling edge of the data bit is detected, if the signal stays low longer than specified, excess low width is detected. The detection completes reception and generates an interrupt.

   After interrupt inputs generated, RMCRSTAT<RMCLOIF> bit is set to "1."



14.3.6 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	EOCFHP	ADBFHP	HPADCE	-	HPADCH			
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-8	−	R	Read as 0.
7	EOCFHP	R	Top-priority AD conversion completion flag (Note1) 0: Before or during conversion 1: Completion
6	ADBFHP	R	Top-priority AD conversion BUSY flag 0: During conversion halts 1: During conversion
5	HPADCE	R/W	Activate top-priority conversion 0: Don't care 1: Start conversion "0" is always read.
4	−	R/W	Write "0".
3-0	HPADCH[3:0]	R/W	Select analog input channel when activating top-priority conversion. (See the table below)

<HPADCH[3:0]>	Analog input channel whene xecuting top-priority conversion
0000	Setting prohibited
0001	Setting prohibited
0010	Setting prohibited
0011	Setting prohibited
0100	AIN4
0101	AIN5
0110	AIN6
0111	AIN7
1000	AIN8
1001	AIN9
1010	AIN10
1011	AIN11
1100	Setting prohibited
1101	
1110	
1111	