Toshiba Semiconductor and Storage - TMPM361F10FG(C,J) 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	64MHz
Connectivity	I²C, SIO, UART/USART
Peripherals	DMA, PWM, WDT
Number of I/O	68
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-20°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/tmpm361f10fg-c-j

Revision History

Date	Revision	Comment
2011/6/20	1	First Release
2013/5/31	2	Contents Revised

Table 1-1 Pin Names and Functions Sorted by Pin (4/6)

Type	Pin No.	PIn Name	Input / Output	Function
Function	51	PE0 A17 TB5IN0	I/O Output Input	I/O port Address bus Inputting the 16-bit timer / event countercapture trigger
Function	52	PE1 A18 TB5IN1	I/O Output Input	I/O port Address bus Inputting the 16-bit timer / event countercapture trigger
Function	53	PE2 A19 TB6IN0	I/O Output Input	I/O port Address bus Inputting the 16-bit timer / event countercapture trigger
Function	54	PE3 A20 TB6IN1	I/O Output Input	I/O port Address bus Inputting the 16-bit timer / event countercapture trigger
Function	55	PE4 A21 TXD0	I/O Output Output	I/O port Address bus Serial channel sending serial data
Function	56	PE5 A22 RXD0	I/O Output Input	I/O port Address bus Serial channel receiving serial data
Function	57	PE6 A23 SCLK0 $\overline{CTS0}$	I/O Output I/O Input	I/O port Address bus Serial channel clock pin Serial channel handshake input pin
Function	58	PE7 INT5 SCOUT	I/O Input Output	I/O port External interrupt pin Internal clock output pin
PS	59	DVDD3B	-	Power supply pin
PS	60	DVSS	-	GND pin
Debug	61	SWDIO	I/O	Debug pin
Debug	62	SWCLK	I/O	Debug pin
Function/ Debug	63	PF0 TRACECLK	I/O Output	I/O port Debug pin
Function/ Debug	64	PF1 TRACEDATA0 SWV	I/O Output Output	I/O port Debug pin Debug pin
Function/ Debug	65	PF2 TRACEDATA1	I/O Output	I/O port Debug pin
Function/ Debug	66	PF3 TRACEDATA2	I/O Output	I/O port Debug pin
Function/ Debug	67	PF4 TRACEDATA3	I/O Output	I/O port Debug pin

6.2.5 CGPLLSEL (PLL Selection 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	RS				-	IS		C2S
After reset	0	1	1	1	0	0	1	0
	7	6	5	4	3	2	1	0
bit symbol	ND					-	-	PLLSEL
After reset	0	0	0	1	1	1	1	0

Bit	Bit Symbol	Type	Function
31-16	−	R	Read as "0".
15-12	RS[3:0]	R/W	Clock multiplied by PLL 0111: 4 times 1010: 8 times Others: Reserved
11	−	R	Read as "0".
10-9	IS[1:0]	R/W	Clock multiplied by PLL 00: 8 times 01: 4 times Others: Reserved
8	C2S	R/W	Clock multiplied by PLL 0: 4 times 1: 8 times
7-3	ND[4:0]	R/W	Clock multiplied by PLL 00011: 4 times 00111: 8 times Others: Reserved
2-1	−	R/W	Write as "1".
0	PLLSEL	R/W	Use PLL 0: fosc 1: f_{PLL} Specifies use or disuse of the clock multiplied by the PLL. fosc is automatically set after reset. Resetting is required when using the PLL.

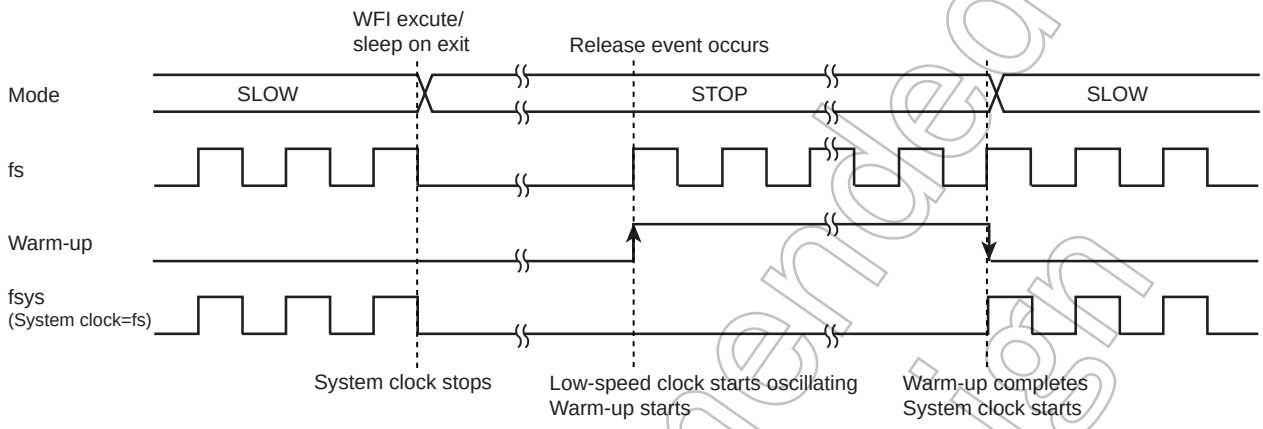
Note 1: Select PLL multiplying value which is shown Table 6-1.

Note 2: Select PLL multiplying value when CGOSCCR<PLLON> = "0" (PLL stop).

Note 3: After setting PLL multiplying value, to keep (CGOSCCR<PLLON> = "0" (PLL stop) over 100 μs is needed as the PLL initializing stable time.

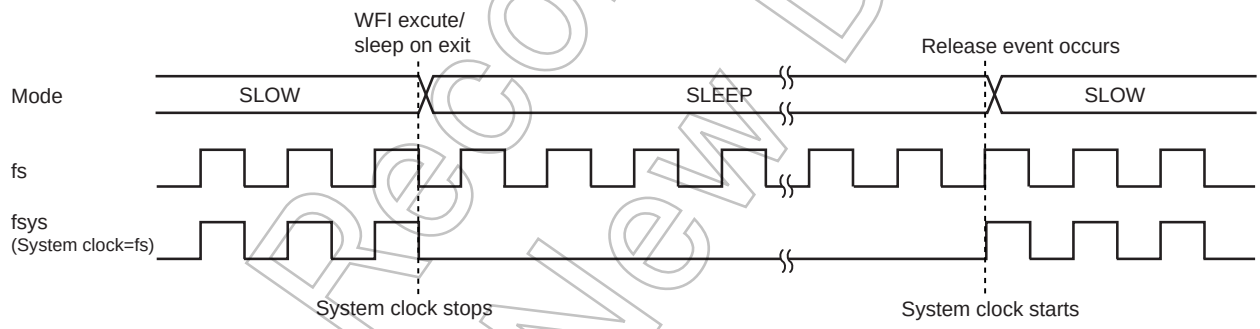
6.6.9.3 Transition of operation modes : SLOW → STOP → SLOW

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



6.6.9.4 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.



7.1.2.4 Exception exit

(1) Execution after returning from an ISR

When returning from an ISR, the CPU takes one of the following actions :

- Tail-chaining

  If a pending exception exists and there are no stacked exceptions or the pending exception has higher priority than all stacked exceptions, the CPU returns to the ISR of the pending exception.

  In this case, the CPU skips the pop of eight registers and push of eight registers when exiting one ISR and entering another. This is called "tail-chaining".

- Returning to the last stacked ISR

  If there are no pending exceptions or if the highest priority stacked exception is of higher priority than the highest priority pending exception, the CPU returns to the last stacked ISR.

- Returning to the previous program

  If there are no pending or stacked exceptions, the CPU returns to the previous program.

(2) Exception exit sequence

When returning from an ISR, the CPU performs the following operations :

- Pop eight registers

  Pops the eight registers (PC, xPSR, r0 to r3, r12 and LR) from the stack and adjust the SP.

- Load current active interrupt number

  Loads the current active interrupt number from the stacked xPSR. The CPU uses this to track which interrupt to return to.

- Select SP

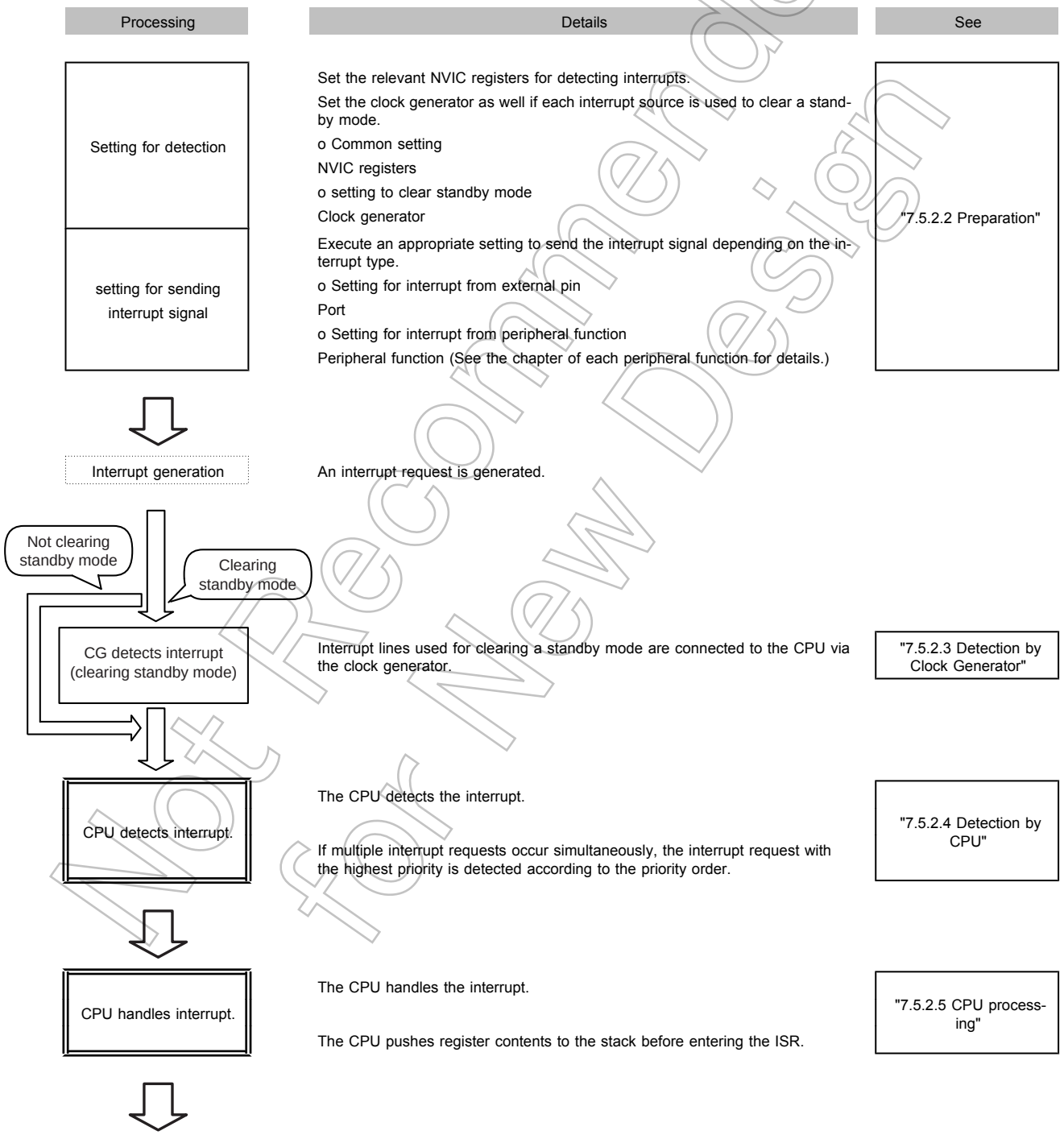
  If returning to an exception (Handler Mode), SP is SP_main. If returning to Thread Mode, SP can be SP_main or SP_process.

7.5.2 Interrupt Handling

7.5.2.1 Flowchart

The following shows how an interrupt is handled.

The following shows how an exception/interrupt is handled. In the following descriptions, ▭ indicates hardware handling. ▭ indicates software handling.



8.3.12 Type T11

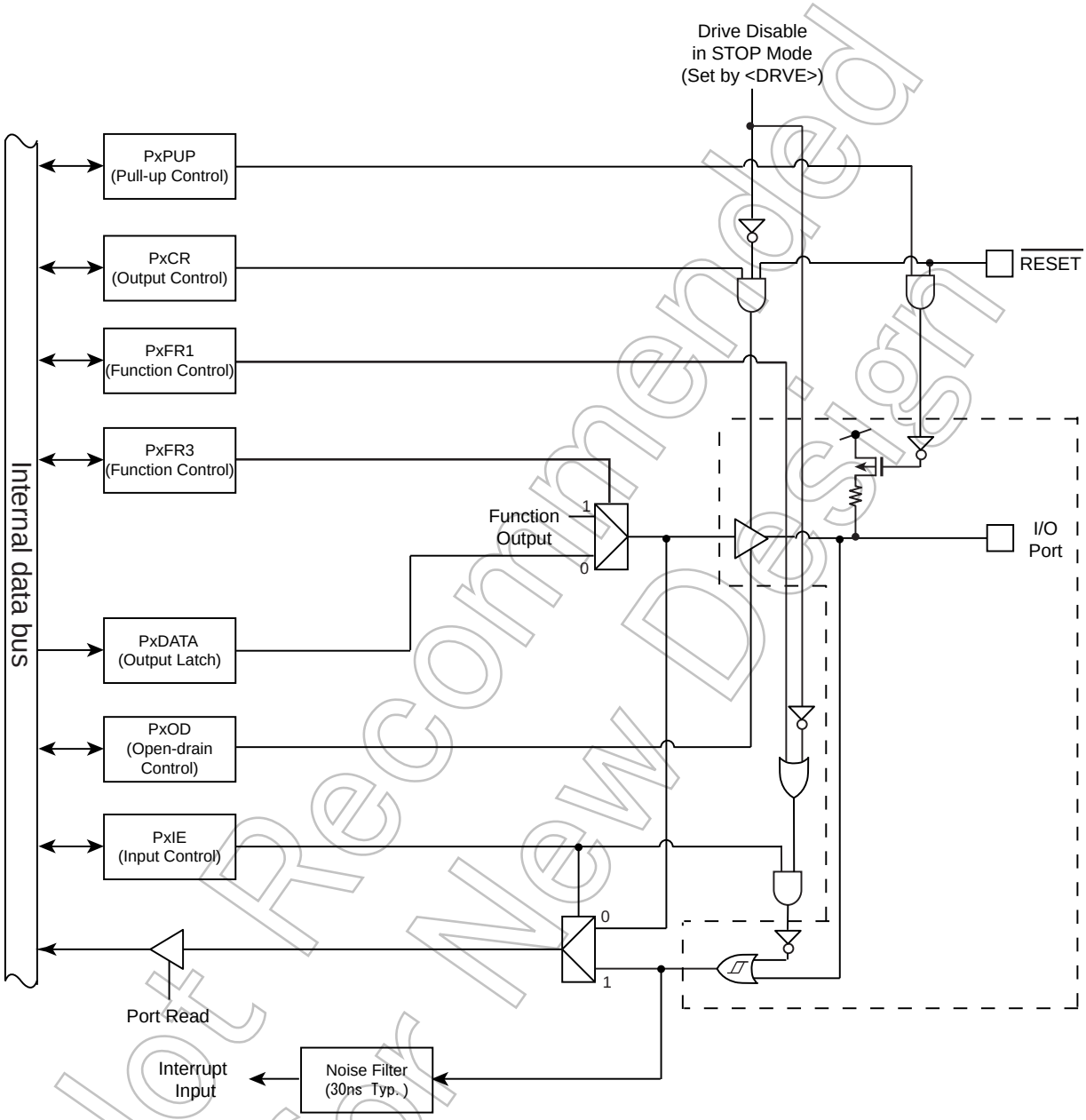


Figure 8-11 Port Type T11

9.5.5 DMACxIntErrorStatus (DMAC Interrupt Error Status 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	-	-	-	-	-	-	IntErrStatus1	IIntErrStatus0
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0

Bit	Bit Symbol	Type	Function
31-2	-	-	Write as zero.
1	IntErrStatus1	R	Status of DMAC channel 1 error interrupt generation. 0 : Interrupt not requested 1 : Interrupt requested Shows error interrupt status after enabled.
0	IntErrStatus0	R	Status of DMAC channel 0 error interrupt generation. 0 : Interrupt not requested 1 : Interrupt requested Shows error interrupt status after enabled.

9.5.7 DMACxRawIntTCStatus (DMAC Raw Interrupt Terminal Count Status 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	-	-	-	-	-	-	RawIntTCS1	RawIntTCS0
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0

Bit	Bit Symbol	Type	Function
31-2	-	-	Write as zero.
1	RawIntTCS1	R	Status of DMAC channel 1 pre-enable transfer end interrupt generation 0 : Interrupt not requested 1 : Interrupt requested
0	RawIntTCS0	R	Status of DMAC channel 0 pre-enable transfer end interrupt generation 0 : Interrupt not requested 1 : Interrupt requested

11.5.4 Capture

This is a circuit that controls the timing of latching values from the UC up-counter into the TBxCP0 and TBxCP1 capture registers. The timing with which to latch data is specified by TBxMOD<TBCPM[1:0]>.

Software can also be used to import values from the UC up-counter into the capture register; specifically, UC values are taken into the TBxCP0 capture register each time "0" is written to TBxMOD<TBCP>.

11.5.5 Capture registers (TBxCP0, TBxCP1)

This register captures an up-counter (UC) value.

11.5.6 Up-counter capture register (TBxUC)

Other than the capturing functions shown above, the current count value of the UC can be captured by reading the TBxUC registers.

11.5.7 Comparators (CP0, CP1)

This register compares with the up-counter (UC) and the value setting of the Timer Register (TBxRG0 and TBxRG1) to detect whether there is a match or not. If a match is detected, INTTBx is generated.

11.5.8 Timer Flip-flop (TBxFF0)

The timer flip-flop (TBxFF0) is reversed by a match signal from the comparator and a latch signal to the capture registers. It can be enabled or disabled to reverse by setting the TBxFFCR<TBC1T1, TBC0T1, TBE1T1, TBE0T1>.

The value of TBxFF0 becomes undefined after a reset. The flip-flop can be reversed by writing "00" to TBxFFCR<TBFF0C[1:0]>. It can be set to "1" by writing "01," and can be cleared to "0" by writing "10."

The value of TBxFF0 can be output to the Timer output pin (TBxOUT). If the timer output is performed, the corresponding port settings must be programmed beforehand.

11.5.9 Capture interrupt (INTCAPx0, INTCAPx1)

Interrupts INTCAPx0 and INTCAPx1 can be generated at the timing of latching values from the UC up-counter into the TBxCP0 and TBxCP1 capture registers. The interrupt timing is specified by the CPU.

12.4.13 SCxTST (TX FIFO Status 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	TUR	-	-	-	-	TLVL		
After reset	1	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-8	−	R	Read as "0".
7	TUR	R	TX FIFO Under run (Note) 0: Not generated 1: Generated
6-3	−	R	Read as "0".
2-0	TLVL[2:0]	R	Status of TX FIFO level 000: Empty 001: 1 byte 010: 2 byte 011: 3 byte 100: 4 byte

Note: **The <TUR> bit is cleared to "0" when transmit data is written to the SCxBUF register.**

Table 12-6 Clock resolution to the Baud Rate Generator fc = 32 MHz, fs = 32.768kHz

φT0 selection CGSYSCR <FPSEL1>	Peripheral clock selection CGSYSCR <FPSEL0>	Clock gear value CGSYSCR <GEAR[2:0]>	Prescaler clock selection CGSYSCR <PRCK[2:0]>	Prescaler output clock resolution			
				φT1	φT4	φT16	φT64
0	1 (fc)	000 (fc)	000 (fperiph/1)	fc/2^1 (0.0625 μs)	fc/2^3 (0.25 μs)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)
			001 (fperiph/2)	fc/2^2 (0.125 μs)	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)
			010 (fperiph/4)	fc/2^3 (0.25 μs)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)
			011 (fperiph/8)	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)
			100 (fperiph/16)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)	fc/2^{11} (64.0 μs)
			101 (fperiph/32)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)	fc/2^{12} (128.0 μs)
		100 (fc/2)	000 (fperiph/1)	–	fc/2^3 (0.25 μs)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)
			001 (fperiph/2)	fc/2^2 (0.125 μs)	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)
			010 (fperiph/4)	fc/2^3 (0.25 μs)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)
			011 (fperiph/8)	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)
			100 (fperiph/16)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)	fc/2^{11} (64.0 μs)
			101 (fperiph/32)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)	fc/2^{12} (128.0 μs)
		101 (fc/4)	000 (fperiph/1)	–	fc/2^3 (0.25 μs)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)
			001 (fperiph/2)	–	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)
			010 (fperiph/4)	fc/2^3 (0.25 μs)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)
			011 (fperiph/8)	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)
			100 (fperiph/16)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)	fc/2^{11} (64.0 μs)
			101 (fperiph/32)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)	fc/2^{12} (128.0 μs)
		110 (fc/8)	000 (fperiph/1)	–	–	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)
			001 (fperiph/2)	–	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)
			010 (fperiph/4)	–	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)
			011 (fperiph/8)	fc/2^4 (0.5 μs)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)
			100 (fperiph/16)	fc/2^5 (1.0 μs)	fc/2^7 (4.0 μs)	fc/2^9 (16.0 μs)	fc/2^{11} (64.0 μs)
			101 (fperiph/32)	fc/2^6 (2.0 μs)	fc/2^8 (8.0 μs)	fc/2^{10} (32.0 μs)	fc/2^{12} (128.0 μs)
1	*	*	*	fs/2 (61μs)	fs/2^3 (244 μs)	fs/2^5 (977 μs)	fs/2^7 (3.91 ms)

Note 1: **The prescaler output clock φTn must be selected so that the relationship "φTn ≤ fsys/2" is satisfied (so that φTn is slower than fsys).**

Note 2: **Do not change the clock gear while SIO is operating.**

Note 3: **The "–" indicates that the setting is prohibited and the "*" indicates don't care in the above table.**

18.3.9 KWUPINT (Interrupt monitor 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	-	-	-	-	KEYINT3	KEYINT2	KEYINT1	KEYINT0
After reset	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	Function
31-4	−	R	Read as "0".
3-0	KEYINT3 to KEYINT0	R	Interrupt 0:No 1:Yes

When KWUPCRn<KEYnEN>="1" and activated signal into the key-on wakeup port, the <KEYINTn> corresponding channel of KWUPINT will be set to "1" for the interrupt execution.

KWUPINT is read only register, due to the reading this register corresponding bit set to "1" and interrupt request will becleared. That one all can be cleared at once by the KWUPCLR register.

When setting the active status to "Level" input by KWUPCRn<KEYn>, KWUPINT corresponding bit for a key-on wakeup is kept "1" in reading without changing a external input setting function to nothing.

(2) Transition to BACKUP mode

1. Setting modes and clearing release source of BACKUP mode

   By the CGSTBYCR<STBY> register, set to the BACKUP STOP mode or BACKUP SLEEP mode.

2. Transition to the BACKUP mode

   Clear the interrupt which releases from BACKUP mode, then execute WFI instruction

Precautions for the use of the BACKUP mode (about debug tool)
The communication with debug tool is disconnected, if MCU changes to the BACKUP mode. In this case, it is necessary to reconnect to debug tool.

(3) Returning from backup mode (Releasing)

1. Releasing source of BACKUP mode

   Releasing source of BACKUP STOP and BACKUP SLEEP shown as below.

BACKUP mode	Releasing source of BACKUP mode
BACKUP STOP	INT0 to 4, INTKWUP (Static)
BACKUP SLEEP	INT0 to 4, INTKWUP (Dynamic / Static), INTRTC, INTCECRX, INTRMCRX0

2. Releasing operation by releasing source of BACKUP mode

   If the event of releasing source are received, regulator starts to supply power to the shut down block. Depending on the returned modes, high-speed oscillator and low-speed oscillator will start operation.

   The warm up timer will starts when the oscillation becomes stable. During warm up time, internal reset signal of power shut down block which returned from BACKUP mode is continuing active level. Internal reset is cleared after warm up time has elapsed, and then MCU returns to the preceding mode of BACKUP mode.

Precaution after BACKUP mode released
・ By reading CGRSTFLG register, it can be found which reset are occurred.
・ Make sure to perform the port A, B, E, F, G, and P setting before releasing port keep function by (CGSTBYCR<PTKEEP>="0").

20.4.5.5 Reactivating normal AD conversion

To reactivate normal AD conversion while the conversion is underway, a software reset (ADMOD3<ADRST>) must be performed before starting AD conversion. The H/W activation method must not be used to reactivate normal AD conversion.

20.4.5.6 Conversion completion

(1) Normal AD conversion completion

When normal AD conversion is completed, the AD conversion completion interrupt (INTAD) is generated. The result of AD conversion is stored in the storage register is the storage register, and two registers change: the register ADMOD0<EOCFN> which indicates the completion of AD conversion and the register ADMOD0<ADBFN>.Interrupt request, conversion register storage register and <EOCFN><ADBFN> change with a different timing according to a mode selected.

In mode other than fixed-channel repeat conversion mode, conversion results are stored in AD conversion result registers (ADREG08 through ADRG7F) corresponding to a channel.

In fixed-channel repeat conversion mode, the conversion results are sequentially stored in storage registers ADREG08 through ADREG7F. However, if interrupt setting on <ITM> is set to be generated each time one AD conversion is completed, the conversion result is stored only in ADREG08. If interrupt setting on <ITM> is set to be generated each time four AD conversions are completed, the conversion results are sequentially stored in ADREG08H through ADREG3B.If interrupt setting on <ITM> is set to be generated each time eight AD conversions are completed, the conversion results are sequentially stored in ADREG08H through ADREG7F.

Interrupt requests, flag changes and conversion result registers in each mode are as shown below.

- Fixed-channel single conversion mode

  After AD conversion completed, ADMOD0<EOCFN> is set to "1", ADMOD0<ADBFN> is cleared to "0", and the interrupt request is generated.

  Conversion results are stored a conversion result register correspond to a channel.

- Channel scan single conversion mode

  After the channel scan conversion is completed, ADMOD0<EOCFN> is set to "1", ADMOD0<ADBFN> is set to "0", and the interrupt request INTAD is generated.

  Conversion results are stored a conversion result register correspond to a channel.

- Fixed-channel repeat conversion mode

  ADMOD0<ADBFN> is not cleared to "0". It remains at "1". The timing with which the interrupt request INTAD is generated can be selected by setting ADMOD0<ITM> to an appropriate setting. ADMOD0<EOCFN> is set with the same timing as this interrupt INTAD is generated.

  a. One conversion

     With <ITM[1:0]> set to "00", an interrupt request is generated each time one AD conversion is completed. In this case, the conversion results are always stored in the storage register ADREG08. After the conversion result is stored, <EOCFN> changes to "1".

  b. Four conversions

22.2.9.2 Show Flash Memory SUM

Table 22-7 Transfer Format for the Show Flash Memory SUM Command

	Byte	Data Transferred from the Controller to the TMPM361F10FG	Baud rate	Data Transferred from the TMPM361F10FG to the Controller
Boot ROM	1 byte	Serial operation mode and baud rate For UART mode : 0x86 For I/O Interface mode : 0x30	Desired baud rate (Note 1)	−
	2 byte	−		ACK for the serial operation mode byte ・For UART mode - Normal acknowledge : 0x86 (The boot program aborts if the baud rate can not be set correctly.) ・For I/O Interface mode - Normal acknowledge :0x30
	3 byte	Command code (0x20)		−
	4 byte	−		ACK for the command code byte (Note 2) - Normal acknowledge : 0x10 - Negative acknowledge : 0xX1 - Communication error : 0xX8
	5 byte	−		SUM (upper byte)
	6 byte	−		SUM (lower byte)
	7 byte	−		Checksum value for byte 5 and 6
	8 byte	(Wait for the next command code.)		−

Note 1: **In I/O Interface mode, the baud rate for the transfers of the first and second bytes must be 1/16 of the desired baud rate.**

Note 2: **In case of any negative acknowledge, the boot program returns to a state in which it waits for a command code (3rd byte). In I/O Interface mode, if a communication error occurs, a negative acknowledge does not occur.**

26.6.2 Static memory controller (SMC)

"T" is 1/2 cycles of an internal bus frequency (fsys) in the Equation of the table.

AC measurement condition

- Output levels : High = 0.7 x DVDD3, Low = 0.3 x DVDD3
- Input levels : High = 0.7 x DVDD3, Low = 0.3 x DVDD3
- Load capacity : CL = 40pF

26.6.2.1 Basic Bus cycle (Read)

Parameter	Symbol	Equation		fsys = 64 MHz T=31.25 N = 4 M = 1 K = 5 L = 2 P = 2 Q = 2	Unit
		Min.	Max.		
SMCCLK	t_{CYC}	31.25	2000	31.3	ns
A1 to A23 Valid → D0 to D15 Input (Multiplex bus mode)	t_{ADL}	−	(N)T − 35.0	90.0	
$\overline{OE}$ falling edge → D0 to D15 Input	t_{OED}	−	(N − M)T − 25.0	68.8	
$\overline{OE}$ Low-level pulse width	t_{OEW}	(N − M)T − 13.0	−	80.8	
A1 ~ A16 Valid →$\overline{OE}$ falling edge (Multiplex bus mode)	t_{AOEL}	(M)T − 15.0	−	16.3	
$\overline{OE}$ rising edge → D0 to D15 Hold	t_{HR}	0.00	−	0.00	
A1 to A23 Valid → D0 to D15 Hold	t_{HA}	0.00	−	0.00	
$\overline{OE}$ High-level pulse width	t_{OEHW}	(M)T − 13.0	−	18.3	
$\overline{ALE}$ Low-level pulse width	t_{LL}	T − 13.0	−	18.3	
A1 to A16 Valid → $\overline{ALE}$ rising edge	t_{AL}	T − 15.0	−	16.3	
$\overline{ALE}$ rising edge → A1to A16 Hold	t_{LA}	T − 10.0	−	21.3	
$\overline{OE}$ rising edge → $\overline{ALE}$ falling edge	t_{CLR}	(P)T − 13.0	−	49.5	
$\overline{OE}$ rising edge → A1to A16 Hold	t_{CAR}	(P)T − 13.0	−	49.5	
$\overline{OE}$ rising edge → A1to A16 Output	t_{RAE}	(P)T − 13.0	−	49.5	

Note:**" Equation Measurement condition:**

N = t_{RC} Cycle ≥ 3 , M = t_{CEOE} Cycle ≥ 1

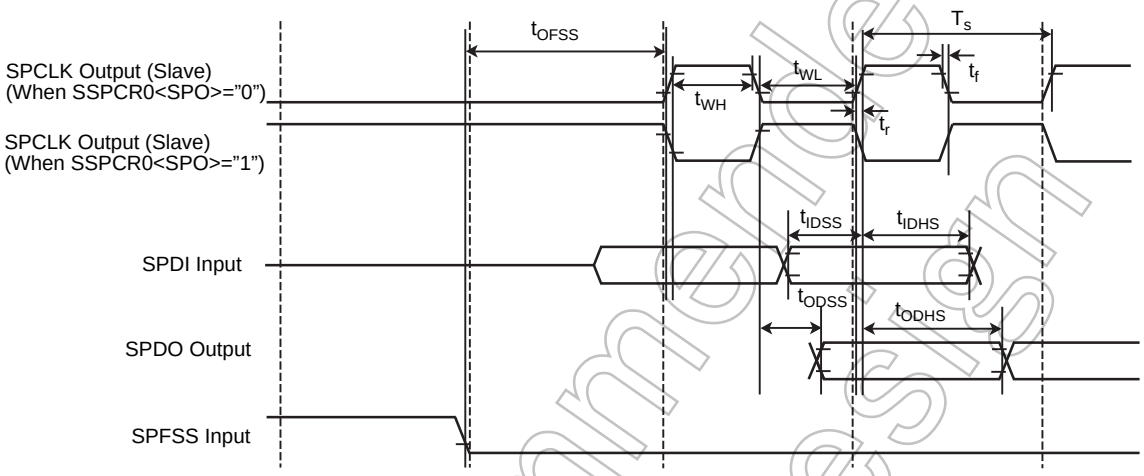
K = t_{WC} Cycle ≥ 3 , L = t_{WP} Cycle ≥ 1

P = t_{TR} Cycle ≥ 1 , Q = t_{PC} Cycle ≥ 1

26.6.5.2 SSP SPI mode (Slave)

- fsys / 12 ≥ f_{SPCLK} ≥ fsys / 65024

(1) Slave SSPCR0<SPH> = "0" (Data is latched on the first edge)



(2) Slave SSPCR0<SPH> = "1" (Data is latched on the second edge)