TOSHIBA

TOSHIBA Original CMOS 8-Bit Microcontroller

TLCS-870/X Series

TMP88CU74FG

TOSHIBA CORPORATION

Semiconductor Company

Document Change Notification

The purpose of this notification is to inform customers about the launch of the Pb-free version of the device. The introduction of a Pb-free replacement affects the datasheet. Please understand that this notification is intended as a temporary substitute for a revision of the datasheet.

Changes to the datasheet may include the following, though not all of them may apply to this particular device.

1. Part number

   Example: TMPxxxxxxF → TMPxxxxxxFG

   All references to the previous part number were left unchanged in body text. The new part number is indicated on the prelims pages (cover page and this notification).

2. Package code and package dimensions

   Example: LQFP100-P-1414-0.50C → LQFP100-P-1414-0.50F

   All references to the previous package code and package dimensions were left unchanged in body text. The new ones are indicated on the prelims pages.

3. Addition of notes on lead solderability

   Now that the device is Pb-free, notes on lead solderability have been added.

4. RESTRICTIONS ON PRODUCT USE

   The previous (obsolete) provision might be left unchanged on page 1 of body text. A new replacement is included on the next page.

5. Publication date of the datasheet

   The publication date at the lower right corner of the prelims pages applies to the new device.

1. Part number

2. Package code and dimensions

Previous Part Number (in Body Text)	Previous Package Code (in Body Text)	New Part Number	New Package Code	OTP
TMP88CU74F	P-QFP80-1420-0.80B	TMP88CU74FG	QFP80-P-1420-0.80B	TMP88PU74FG

*: For the dimensions of the new package, see the attached Package Dimensions diagram.

3. Addition of notes on lead solderability

The following solderability test is conducted on the new device.

Lead solderability of Pb-free devices (with the G suffix)

Test	Test Conditions	Remark
Solderability	(1) Use of Lead (Pb) ·solder bath temperature = 230°C ·dipping time = 5 seconds ·the number of times = once ·use of R-type flux (2) Use of Lead (Pb)-Free ·solder bath temperature = 245°C ·dipping time = 5 seconds ·the number of times = once ·use of R-type flux	Leads with over 95% solder coverage till lead forming are acceptable.

4. RESTRICTIONS ON PRODUCT USE

The following replaces the "RESTRICTIONS ON PRODUCT USE" on page 1 of body text.

RESTRICTIONS ON PRODUCT USE 20070701-EN

- The information contained herein is subject to change without notice.
- TOSHIBA is continually working to improve the quality and reliability of its products. Nevertheless, semiconductor devices in general can malfunction or fail due to their inherent electrical sensitivity and vulnerability to physical stress. It is the responsibility of the buyer, when utilizing TOSHIBA products, to comply with the standards of safety in making a safe design for the entire system, and to avoid situations in which a malfunction or failure of such TOSHIBA products could cause loss of human life, bodily injury or damage to property.
  In developing your designs, please ensure that TOSHIBA products are used within specified operating ranges as set forth in the most recent TOSHIBA products specifications. Also, please keep in mind the precautions and conditions set forth in the "Handling Guide for Semiconductor Devices," or "TOSHIBA Semiconductor Reliability Handbook" etc.
- The TOSHIBA products listed in this document are intended for usage in general electronics applications (computer, personal equipment, office equipment, measuring equipment, industrial robotics, domestic appliances, etc.).These TOSHIBA products are neither intended nor warranted for usage in equipment that requires extraordinarily high quality and/or reliability or a malfunction or failure of which may cause loss of human life or bodily injury ("Unintended Usage"). Unintended Usage include atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, medical instruments, all types of safety devices, etc.. Unintended Usage of TOSHIBA products listed in his document shall be made at the customer's own risk.
- The products described in this document shall not be used or embedded to any downstream products of which manufacture, use and/or sale are prohibited under any applicable laws and regulations.
- The information contained herein is presented only as a guide for the applications of our products. No responsibility is assumed by TOSHIBA for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patents or other rights of TOSHIBA or the third parties.
- Please contact your sales representative for product-by-product details in this document regarding RoHS compatibility. Please use these products in this document in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances. Toshiba assumes no liability for damage or losses occurring as a result of noncompliance with applicable laws and regulations.
- For a discussion of how the reliability of microcontrollers can be predicted, please refer to Section 1.3 of the chapter entitled Quality and Reliability Assurance/Handling Precautions.

5. Publication date of the datasheet

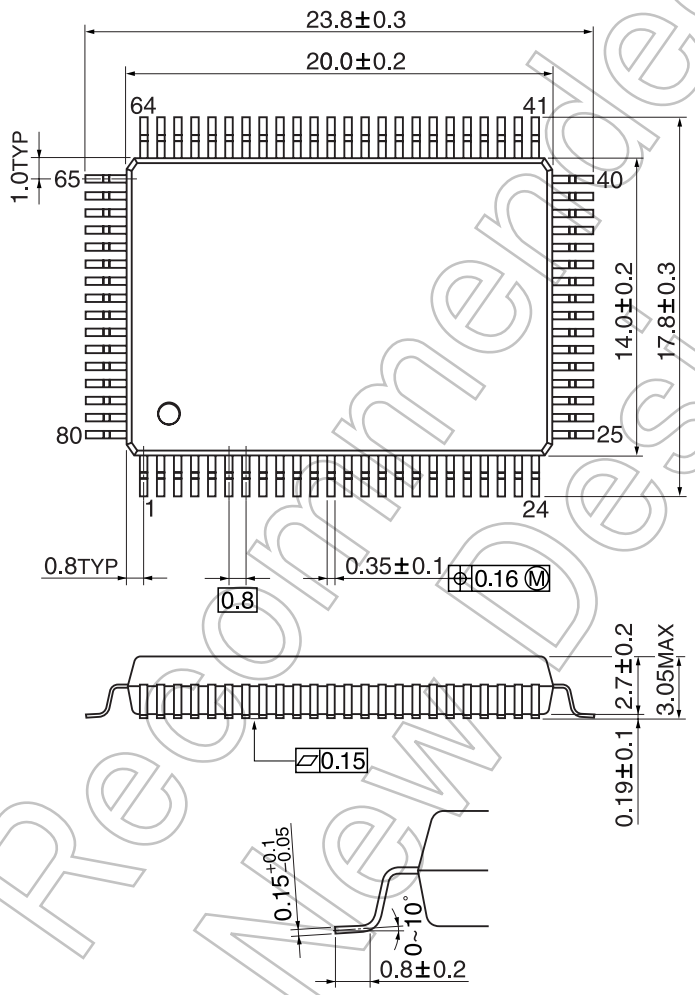
The publication date of this datasheet is printed at the lower right corner of this notification.

(Annex)

Package Dimensions

QFP80-P-1420-0.80B

Unit: mm



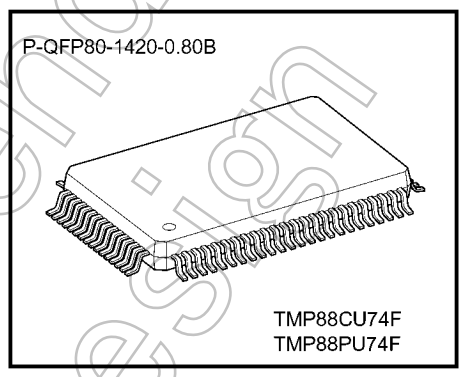
CMOS 8-Bit Microcontroller

TMP88CU74F

The TMP88CU74 are the high speed and high performance 8-bit single chip microcomputers. These MCU contain 8-bit AD conversion inputs and a VFT (Vacuum Fluorescent Tube) driver on a chip.

Product No.	ROM	RAM	Package	OTP MCU
TMP88CU74F	96 Kbytes + 256 bytes	2 Kbytes	P-QFP80-1420-0.80B	TMP88PU74F

Features



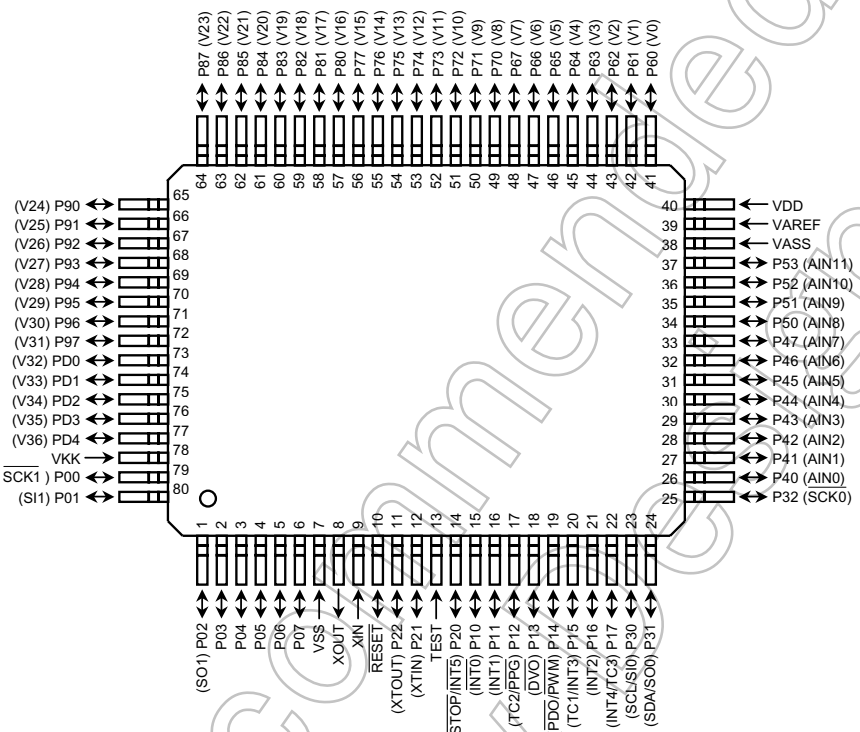
- 8-bit single chip microcomputer TLCS-870/X Series
- Instruction execution time: 0.32 μs (at 12.5 MHz), 122 μs (at 32.768 kHz)
- 842 basic instructions
- General-purpose register: 16 banks
- 15 interrupt sources (External: 6, Internal: 9)
  - All sources have independent latches each
  - Edge-selectable external interrupts with noise reject
  - High-speed task switching by register bank changeover
- Input/Output ports (71 pins)
- 16-bit timer/counters: 2 channels
  - TC1: Timer, Event counter, PPG (Programable Pulse Generator) output, Pulse width measurement, External trigger timer, Window modes.
  - TC2: Timer, Eventcounter, Window modes.
- 8-bit timer/counters: 2 channels
  - TC3: Timer, Eventcounter, Capture (Pulse width/duty measurment)
  - TC4: Timer, PWM output, PDO (Programmable Divider Output) mode

070122EBP

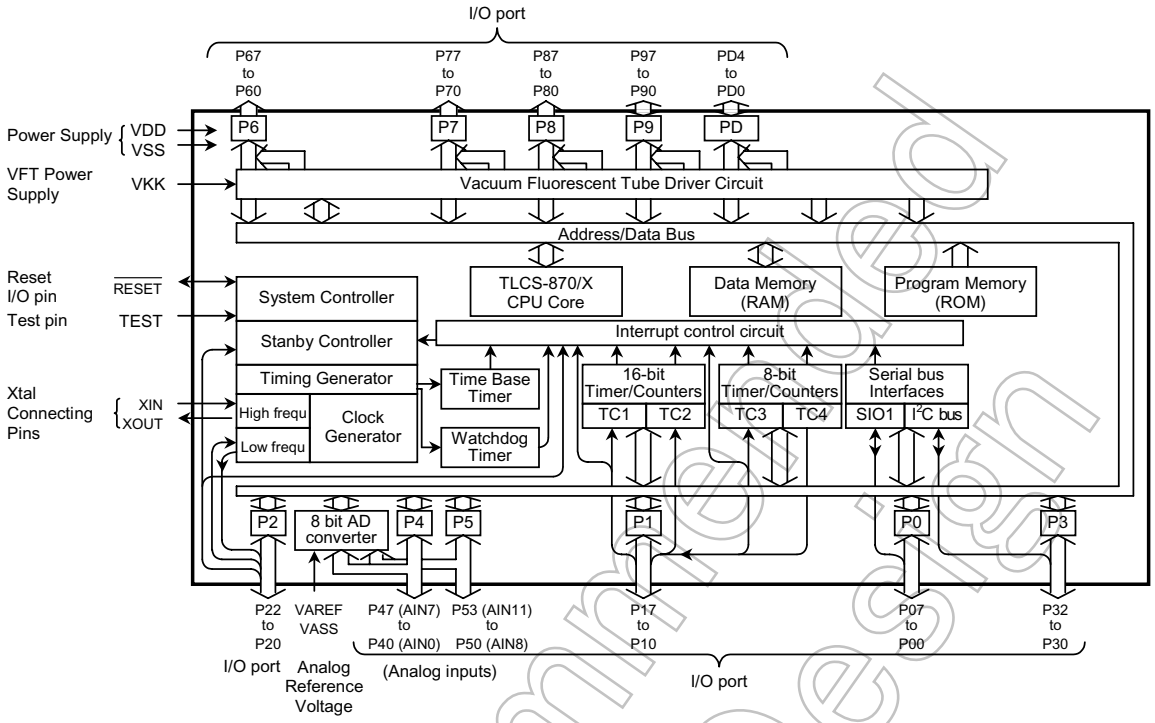
- Time base timer
- Divider output function
- Watchdog timer
  - Interrupt source/reset output (programmable)
- 8-bit serial interface: 1 channel
  - With 8 bytes transmit/receive data buffer
  - Internal/External serial clock, and 4/8-bit mode
- Serial bus interface
  - 8-bit SIO/I²C bus mode
- 8-bit successive approximate type AD converter with sample and hold
  - Analog inputs: 12 channels conversion time: 23 μs at 8 MHz (High-speed conversion mode), 59 μs at 12.5 MHz (Low-speed conversion mode)
- Vacuum fluorescent tube driver (Automatic display)
  - High breakdown voltage ports (Max 40 V × 37 bits)
  - Programmable grid scan output
- Dual clock operation
  - Single/Dual-clock mode (selection)
- Five power saving operating modes
  - STOP mode: Oscillation stops. Battery/Capacitor back-up. Release by stop pin input.
  - SLOW mode: Low power consumption operation using low-frequency clock.
  - IDLE1 mode: CPU stops, and Peripherals operate using high-frequency clock. Release by interrupts.
  - IDLE2 mode: CPU stops, and Peripherals operate using high-and low-frequency clock. Release by interrupts.
  - SLEEP mode: CPU stops, and Peripherals operate using low-frequency clock. Release by interrupts.
- Wide operating voltage: 2.7 to 5.5 V at 32.8 kHz, 4.5 to 5.5 V at 12.5 MHz/32.8 kHz
- Emulation Pod: BM88CU74F0A

Pin Assignments (Top View)

P-QFP80-1420-0.80B



Block Diagram



Pin Functions (1/2)

Pin Name	Input/Output	Function	
P07 to P03	I/O	Two 8-bit programmable input/output ports (tri-state). Each bit of these ports can be individually configured as an input or an output under software control. During reset, all bits are configured as inputs. When used as a PPG output or a divider output, the output latch must be set to “1”.	
P02 (SO1)	I/O (Output)		SIO1 Serial data Output
P01 (SI1)	I/O (Input)		SIO1 Serial data Input
P00 ($\overline{SCK1}$)	I/O (I/O)		SIO1 Serial clock input/output
P17 (INT4/TC3)	I/O (Input)		External interrupt 4 input or Timer Counter 3 input
P16 (INT2)			External interrupt 2 input
P15 (INT3/TC1)			External interrupt 3 input or Timer Counter 1 input
P14 ($\overline{PDO}$ / $\overline{PWM}$)	I/O (Output)		PWM output or programmable divider output
P13 ($\overline{DVO}$)	I/O (Output)		Divider output
P12 (TC2/ $\overline{PPG}$)	I/O (I/O)		Timer counter input 2 or programmable pulse generator output
P11 (INT1)	I/O (Input)		External interrupt input 1
P10 ($\overline{INT0}$)			External interrupt input 0
P22 (XTOUT)	I/O (Output)	3-bit input/output port with latch. When used as an input port, a resonator connecting pin, an external interrupt input, or a STOP mode release input, the output latch must be set to “1”.	Resonator connecting pins (32.8 kHz). For inputting external clock, XTIN is used and XTOUT is opened.
P21 (XTIN)	I/O (Input)		
P20 ($\overline{INT5}$ / $\overline{STOP}$)			External interrupt input 5 or STOP mode release signal input
P32 ($\overline{SCK0}$)	I/O (Input)	3-bit programmable input/output port (tri-state/programmable open drain). Each bit of the port can be individully configured as an input or an output under software control. When used as a serial interface output, the output latch must be set to “1”.	SIO0 clock input/output
P31 (SDA/SO0)	I/O (I/O/Output)		I^2C bus data input/output or SIO0 data output
P30 (SCL/SI0)	I/O (I/O/Input)		I^2C bus clock input/output or SIO0 data input
P47 (AIN7) to P40 (AIN0)	I/O (Input)	8/4-bit programmable input/output port (tri-state). Each bit of the port can be individually configured as an input or output under software control. When used as an analog input set to input mode.	AD converter analog inputs
P53 (AIN13) to P50 (AIN8)	I/O (Input)		
P67 (V7) to P60 (V0)	I/O (Output)	8-bit high breakdown voltage output ports with the latch. When used as an vacuum fluorescent tube driver output, the output latch must be cleared to “0”.	VTF output
P77 (V15) to P70 (V8)			
P87 (V23) to P80 (V16)			
P97 (V31) to P90 (V24)			
PD4(V36) to PD0 (V32)		5-bit high breakdown voltage output ports with the latch. When used as an vacuum fluorescent tube driver output, the latch must be cleared to “0”.	

Pin Functions (2/2)

Pin Name	Input/Output	Function
XIN, XOUT	Input/Output	Resonator connecting pins for high-frequency clock. For inputting external clock, XIN is used an XOUT is opened.
$\overline{\text{RESET}}$	Input/Output	Reset signal input or watchdog timer output/address-reset output/system clock reset output.
TEST	Input	Test pin for out-going teset. Be tied to low.
VDD, VSS	Power Supply	+5 V, 0 V (GND)
VKK		Vacuum fluore scent tube driver voltage pin.
VAREF, VASS		Analog reference voltage input (High, Low)

Operational Description

1. CPU Core Functions

The CPU core consists of a CPU, a system clock controller, an interrupt controller, and a watchdog timer.

This section provides a description of the CPU core, the program memory (ROM), the data memory (RAM), and the reset circuit.

1.1 Memory Address Map

TLCS-870/X Series, the memory is organized 4 address spaces (ROM, RAM, SFR, and DBR). Figure 1.1.1 shows the memory address maps of the TMP88CU74. It uses a memory mapped I/O system, and all I/O registers are mapped in the SFR/DBR address spaces. There are 16 banks of general-purpose registers.

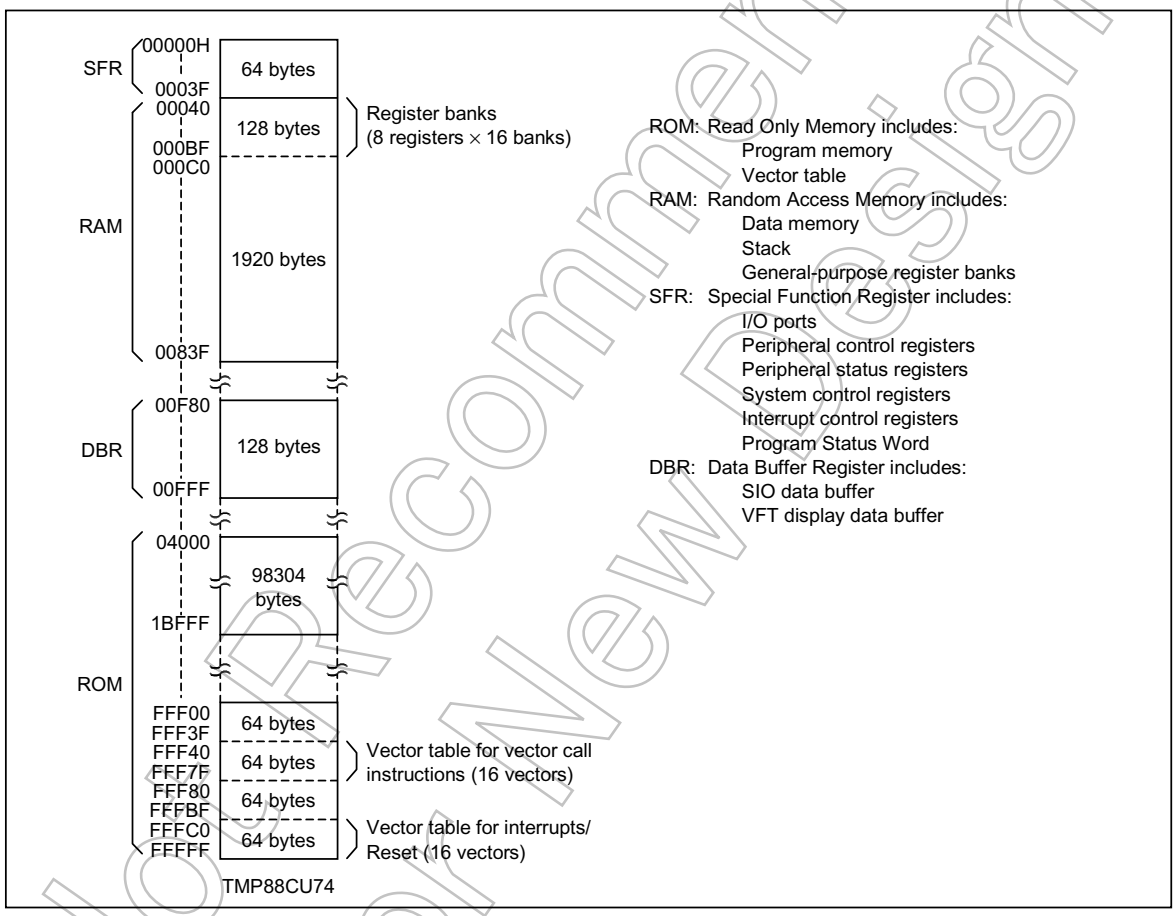


Figure 1.1.1 Memory Address Maps

1.2 Program Memory (ROM)

The TMP88CU74 has a 96 Kbytes (addresses 04000H to 1BFFFH) and 256 bytes (addresses FFF00H to FFFFFH) of program memory (mask programmed ROM). Figure 1.1.1 shown in Memory address maps.

Addresses FFF00H to FFFFFH in the program memory can also be used for special purposes.

1.3 Data Memory (RAM)

The TMP88CU74 has 2 Kbytes of static RAM (address 00040H to 0083FH). The first 128 bytes (00040H to 000BFH) of the internal RAM are also used as general-purpose register banks.

The data memory contents become unstable when the power supply is turned on; therefore, the data memory should be initialized by an initialization routine.

```
Example:  Clears RAM to "00H" except the bank 0.
          LD      HL, 00048H      ;  Sets start address to HL register pair
          LD      A, H            ;  Sets initial data (00H) to A register
          LD      BC, 03F7H       ;  Sets number of byte to BC register pair
SRAMCLR:  LD      (HL+), A
          DEC     BC
          JRS     F, SRAMCLR
```

Note: The general-purpose registers are mapped in the RAM; therefore, do not clear RAM at the current bank addresses.

1.4 System Clock Controller

The system clock controller consists of a clock generator, a timing generator, and a stand-by controller.

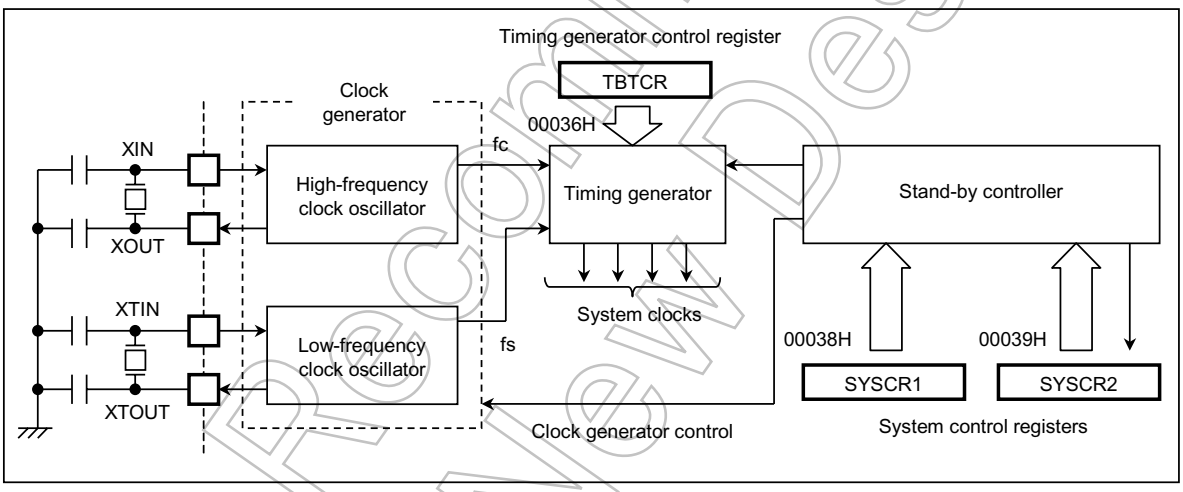


Figure 1.4.1 System Clock Controller

1.4.1 Clock Generator

The clock generator generates the basic clock which provides the system clocks supplied to the CPU core and peripheral hardware. It contains two oscillation circuits: one for the high-frequency clock and one for the low-frequency clock. Power consumption can be reduced by switching of the system clock controller to low-power operation based on the low-frequency clock.

The high-frequency (fc) and low-frequency (fs) clocks can easily be obtained by connecting a resonator between the XIN/XOUT and XTIN/XTOUT pins, respectively. Clock input from an external oscillator is also possible.

In this case, external clock is applied to XIN/XTIN pin with XOUT/XTOUT pin not connected. The TMP88CU74 are not provided an RC oscillation.

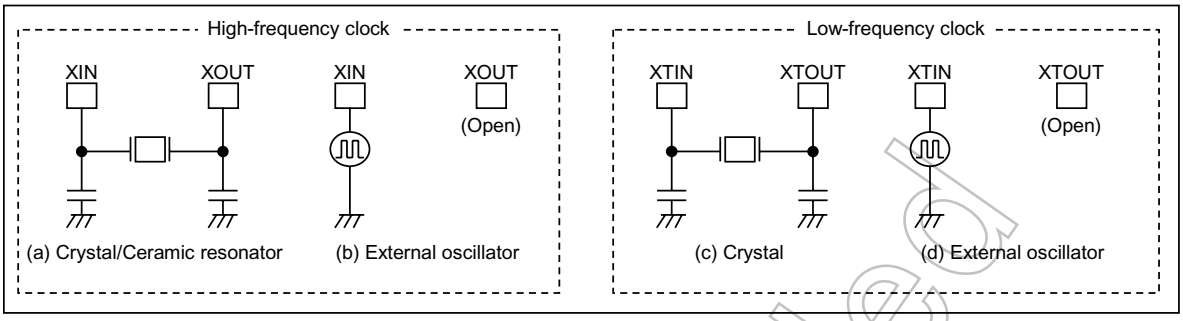


Figure 1.4.2 Examples of Resonator Connection

Note: Accurate Adjustment of the Oscillation Frequency
Although no hardware to externally and directly monitor the basic clock pulse is not provided, the oscillation frequency can be adjusted by making the program to output fixed frequency pulse to the port with disabling all interrupts and watchdog timers, and monitoring this pulse. With a system requiring adjustment of the oscillation frequency, the adjusting program must be created beforehand.

1.4.2 Timing Generator

The timing generator generates the various system clocks supplied to the CPU core and peripheral hardware from the basic clock (fc or fs). The timing generator provides the following functions.

1. Generation of main system clock (fm)
2. Generation of divider output ($\overline{\text{DVO}}$) pulses
3. Generation of source clocks for time base timer
4. Generation of source clocks for watchdog timer
5. Generation of internal source clocks for timer/counters TC1-TC6
6. Generation of internal clocks for serial interfaces SIO and HSO
7. Generation of source clocks for VFT driver circuit
8. Generation of warm-up clocks for releasing STOP mode
9 Generation of a clock for releasing reset output

(1) Configuration of timing generator

The timing generator consists of a 3-stage prescaler, a 21-stage divider, a main system clock generator, and machine cycle counters.

The clock fc/4 or fc/8, that is output from the 2nd stage or the 3rd stage of the prescaler, can be selected as the clock to input to the 1st stage of the divider by DV1CK (bit 5 in CGCR). Inputting fc/8 to the 1st stage of the divider operates the peripheral circuit without the setting change when the operation clock is multiplied by 2. (Example: 8 MHz to 12.5 MHz)

The DV1CK should be set the peripheral circuit prior to starting the peripheral circuits. Do not change the set value after setting.

An input clock to the 7th stage of the divider depends on the operating mode, DV1CK (bit 5 in DVCR), and DV7CK(bit 4 in TBTCR), that is shown in Table 1.4.1. As reset and STOP mode started/canceled, The prescaler and the divider are cleared to “0”.

Table 1.4.1 Input Clock to 7th Stage of the Divider

Single-clock Mode		Dual-clock Mode			
NORMAL1, IDLE1 Mode		NORMAL2, IDLE2 Mode (SYSCK = 0)			SLOW, SLEEP Mode (SYSCK = 1)
DV1CK = 0	DV1CK = 1	DV7CK = 0		DV7CK = 1	
		DV1CK = 0	DV1CK = 1		
$fc/2^8$	$fc/2^9$	$fc/2^8$	$fc/2^9$	fs	fs

Note 1: Do not set DV7CK to "1" in the single clock mode.

Note 2: In SLOW and SLEEP mode, the input clock to the 1st stage of the divider is stopped; output from the 1st to 6th stages is also stopped.

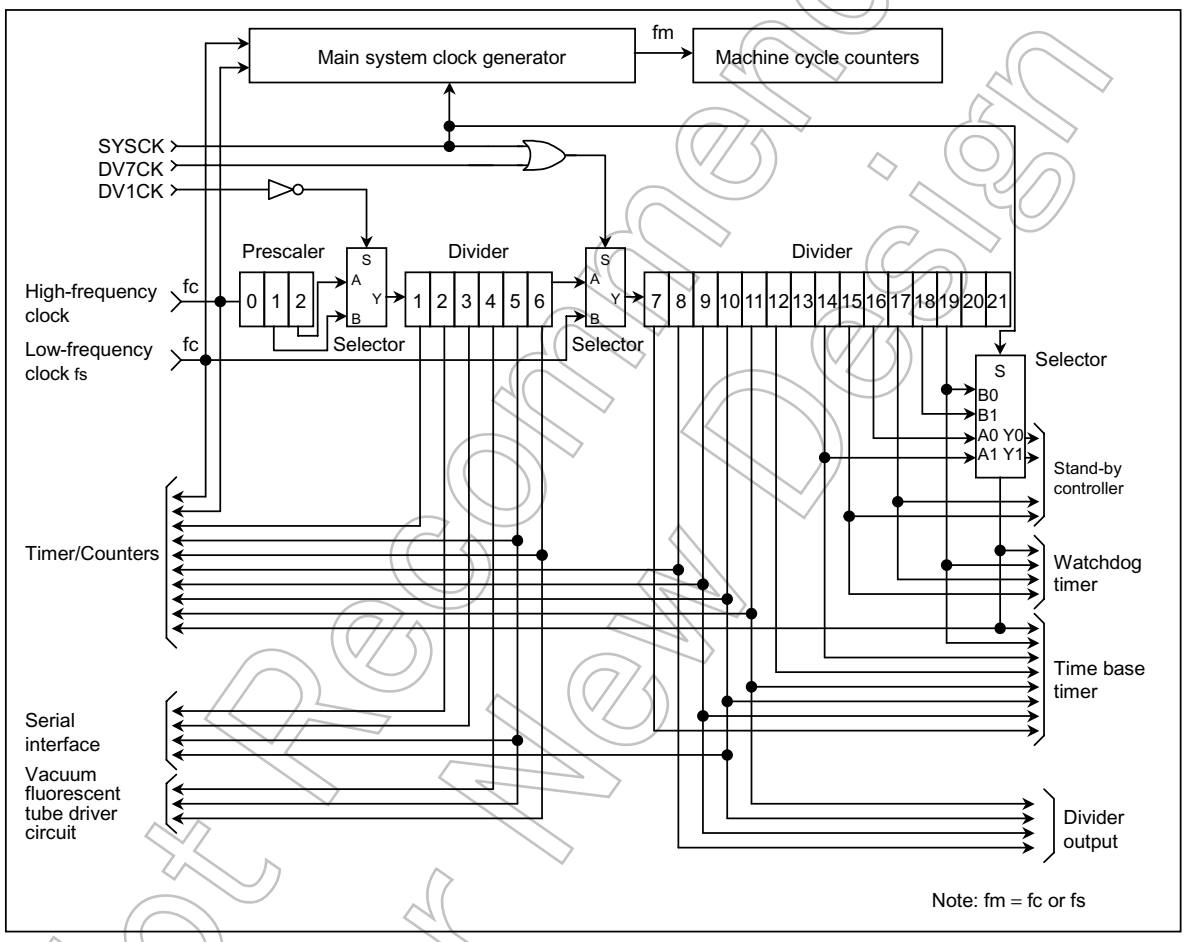


Figure 1.4.3 Configuration of Timing Generator

DVCR (00030H)

7	6	5	4	3	2	1	0
"0"	"0"	DV1CK					

(Initial value: **0* ****)

DV1CK	Selection of input clock to the 1st stage of the divider	0: fc/4 1: fc/8	R/W

Note 1: fc: High-frequency clock [Hz], *: Don't care
Note 2: Bit 4 to 0 in CGCR is always read in as "1" when a read instruction is executed.

Figure 1.4.4 Clock Gear Control Register

TBTCR (00036H)

7	6	5	4	3	2	1	0
(DVOEN)	(DVOCK)		DV7CK	(TBTEN)	(TBTCK)		

(Initial value: 0**0 0***)

DV7CK	Selection of input clock to the 7 th stage of the divide	0: fc/2^8 or fc/2^9 [Hz] 1: fs	R/W

Note 1: fc: High-frequency clock [Hz], *: Don't care
Note 2: Do not set DV7CK to "1" in the single clock mode.
Note 3: Do not set DV7CK to "1" before low-frequency clock is stable in the dual-clock mode.

Figure 1.4.5 Timing Generator Control Register

(2) Machine cycle

Instruction execution and peripheral hardware operation are synchronized with the main system clock.

The minimum instruction execution unit is called an "machine cycle". There are a total of 15 different types of instructions for the TLCS-870/X Series: ranging from 1-cycle instructions which require one machine cycle for execution to 15-cycle instructions which require 15 machine cycles for execution.

A machine cycle consists of 4 states (S0 to S3), and each state consists of one main system clock.

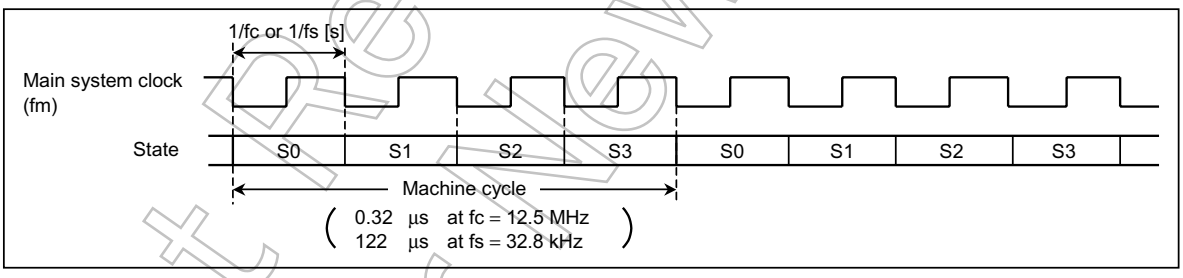


Figure 1.4.6 Machine Cycle

1.4.3 Stand-by Controller

The stand-by controller starts and stops the oscillation circuits for the high-frequency and low-frequency clocks, and switches the main system clock. There are two operating modes: single-clock and dual-clock. These modes are controlled by the system control registers (SYSCR1and SYSCR2).

Figure 1.4.7 shows the operating mode transition diagram and Figure 1.4.8 shows the system control registers.

(1) Single-clock mode

Only the oscillation circuit for the high-frequency clock is used, and P21 (XTIN) and P22 (XTOUT) pins are used as input/output ports. In the single-clock mode, the machine cycle time is 4/fc [s] (0.32 µs at fc = 12.5 MHz).

1. NORMAL1 mode

   In this mode, both the CPU core and on-chip peripherals operate using the high-frequency clock. The TMP88CU74 is placed in this mode after reset.

2. IDLE1 mode

   In this mode, the internal oscillation circuit remains active. The CPU and the watchdog timer are halted; however on-chip peripherals remain active (operate using the high-frequency clock). IDLE1 mode is started by the system control register 2 (SYSCR2), and IDLE1 mode is released to NORMAL1 mode by an interrupt request from the on-chip peripherals or external interrupt inputs. When the IMF (interrupt master enable flag) is "1" (interrupt enable), the execution will resume with the acceptance of the interrupt, and the operation will return to normal after the interrupt service is completed. When the IMF is "0" (interrupt disable), the execution will resume with the instruction which follows the IDLE1 mode start instruction.

3. STOP1 mode

   In this mode, the internal oscillation circuit is turned off, causing all system operations to be halted. The internal status immediately prior to the halt is held with a lowest power consumption during STOP1 mode.

   STOP1 mode is started by the system control register 1 (SYSCR1), and STOP1 mode is released by an inputting (either level-sensitive or edge-sensitive can be programmably selected) to the $\overline{\text{STOP}}$ pin. After the warming-up period is completed, the execution resumes with the instruction which follows the STOP1 mode start instruction.

(2) Dual-clock mode

Both the high-frequency and low-frequency oscillation circuits are used in this mode. P21 (XTIN) and P22 (XTOUT) pins cannot be used as input/output ports. The main system clock is obtained from the high-frequency clock in NORMAL2 and IDLE2 modes, and is obtained from the low-frequency clock in SLOW and SLEEP modes. The machine cycle time is 4/fc [s] in the NORMAL2 and IDLE2 modes, and 4/fs [s] (122 μs at fs = 32.8 kHz) in the SLOW and SLEEP modes.

The TLCS-870/X is placed in the signal-clock mode during reset. To use the dual-clock mode, the low-frequency oscillator should be turned on by executing [SET (SYSCR2), XTEN] instruction.

1. NORMAL2 mode

   In this mode, the CPU core operates using the high-frequency clock. On-chip peripherals operate using the high-frequency clock and/or low-frequency clock.

2. SLOW mode

   This mode can be used to reduce power-consumption by turning off oscillation of the high-frequency clock. The CPU core and on-chip peripherals operate using the low-frequency clock.

   Switching back and forth between NORMAL2 and SLOW modes are performed by the system control register 2 (SYSCR2).

3. IDLE2 mode

   In this mode, the internal oscillation circuit remain active. The CPU and the watchdog timer are halted; however, on-chip peripherals remain active (operate using the high-frequency clock and/or the low-frequency clock). Starting and releasing of IDLE2 mode are the same as for IDLE1 mode, except that operation returns to NORMAL2 mode.

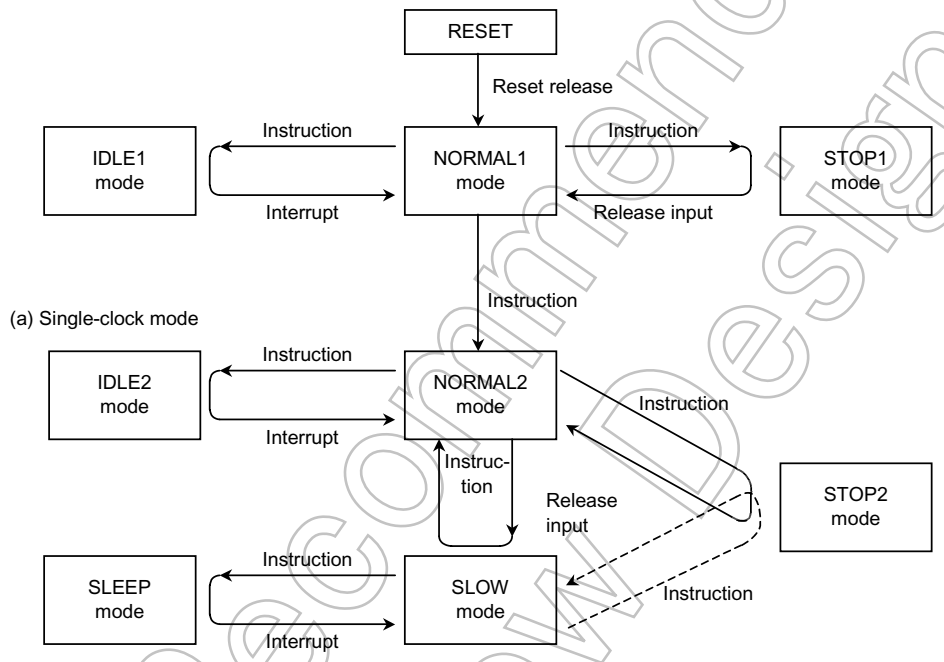
4. SLEEP mode

In this mode, the internal oscillation circuit of the low-frequency clock remains active. The CPU, the watchdog timer, and the internal oscillation circuit of the high-frequency clock are halted; however, on-chip peripherals remain active (operate using the low-frequency clock). Starting and releasing of SLEEP mode are the same as for IDLE1 mode, except that operation returns to SLOW mode.

5. STOP2 mode

As in STOP1 mode, all system operations are halted in this mode.

As in NORMAL2 mode at the start, the operating mode returns to NORMAL2 mode, and as in SLOW mode at the start, it returns to SLOW mode after release.



Note: NORMAL1 and NORMAL2 modes are generically called NORMAL;
STOP1 and STOP2 are called STOP; and IDLE1, IDLE2 and SLEEP are called IDLE.

Operating Mode		Frequency		CPU Core	On-chip Peripherals	Machine Cycle Time
		High-frequency	Low-frequency			
Single-Clock	RESET1	Turning on oscillation	Turning off oscillation	Reset	Reset	4/fc [s]
	NORMAL1			Operate	Operate	
	IDLE1			Halt		
	STOP1	Turning off oscillation			Halt	—
Dual-Clock	NORMAL2	Turning on oscillation	Turning on oscillation	High-frequency	Operate (High and/or Low)	4/fc [s]
	IDLE2			Halt		
	SLOW	Turning off oscillation		Low-frequency	Low-frequency	4/fs [s]
	SLEEP			Halt		
	STOP2		Turning off oscillation		Halt	—

Figure 1.4.7 Operating Mode Transition Diagram

System Control Register 1

SYSCR1 (00038H)

7	6	5	4	3	2	1	0
STOP	RELM	RETM	"1"	WUT			

(Initial value: 0000 00**)

STOP	STOP mode start	0: CPU core and peripherals remain active 1: CPU core and peripherals are halted (start STOP mode)			R/W
RELM	Release method for STOP mode	0: Edge-sensitive release 1: Level-sensitive release			
RETM	Operating mode after STOP mode	0: Return to NORMAL mode 1: Return to SLOW mode			
WUT	Warming-up time at releasing STOP mode		Return to NORMAL mode		Return to SLOW mode
			DV1CK = 0	DV1CK = 1	
		00	3×2^{16}/fc	3×2^{17}/fc	3×2^{13}/fc
		01	2^{16}/fc	2^{17}/fc	2^{13}/fc
		10	3×2^{14}/fc	3×2^{15}/fc	
		11	2^{14}/fc	2^{15}/fc	—

Note 1: Always set RETM to "0" when transiting from NORMAL mode to STOP mode. Always set RETM to "1" when transiting from SLOW mode to STOP mode.

Note 2: When STOP mode is released with $\overline{\text{RESET}}$ pin input, a return is made to NORMAL mode regardless of the RETM contents.

Note 3: fc: High-frequency clock [Hz]
fs: Low-frequency clock [Hz]
*: Don't care

Note 4: Bits 1 and 0 in SYSCR1 are read in as undefined data when a read instruction is executed.

Note 5: Always set bit 4 in SYSCR1 to "1" when STOP mode is started.

System Control Register 2

SYSCR2 (00039H)

7	6	5	4	3	2	1	0
XEN	XTEN	SYSCK	IDLE				

(Initial value: 1000 ****)

XEN	High-frequency oscillator control	0: Turn off oscillation 1: Turn on oscillation	R/W
XTEN	Low-frequency oscillator control	0: Turn off oscillation 1: Turn on oscillation	
SYSCK	Main system clock select (write)/main system clock monitor (read)	0: High-frequency clock 1: Low-frequency clock	
IDLE	IDLE mode start	0: CPU and watchdog timer remain active 1: CPU and watchdog timer are stopped (start IDLE1 mode)	

Note 1: XEN and SYSCK are automatically overwritten in accordance with the contents of RETM (bit 5 in SYSCR1) when STOP mode is released.

RETM	Operating mode after STOP mode	XTEN	SYSCK
0	NORMAL 1/2 mode	1	0
1	SLOW mode	0	1

Note 2: Do not clear XEN to "0" when SYSCK = 0, and do not clear XTEN to "0" when SYSCK = 1.

Note 3: A reset is applied ($\overline{\text{RESET}}$ pin output goes low) if both XEN and XTEN are cleared to "0".

Note 4: *: Don't care

Note 5: Bits 3 to 0 in SYSCR2 are always read in as "1" when a read instruction is executed.

Figure 1.4.8 System Control Registers

1.4.4 Operating Mode Control

(1) STOP mode (STOP1, STOP2)

STOP mode is controlled by the system control register 1 (SYSCR1) and the $\overline{STOP}$ pin input. The $\overline{STOP}$ pin is also used both as a port P20 and an INT5 (external interrupt input 5) pin. STOP mode is started by setting STOP (bit 7 in SYSCR1) to "1". During STOP mode, the following status is maintained.

1. Oscillations are turned off, and all internal operations are halted.
2. The data memory, registers, the program status word and port output latches are all held in the status in effect before STOP mode was entered.
3. The prescaler and the divider of the timing generator are cleared to "0".
4. The program counter holds the address of the instruction but one to the instruction (e.g.[SET (SYSCR1).7]) which started STOP mode.

STOP mode includes a level-sensitive release mode and an edge-sensitive release mode, either of which can be selected with the RELM (bit 6 in SYSCR1).

a. Level-sensitive release mode (RELM = "1")

In this mode, STOP mode is released by setting the $\overline{STOP}$ pin high. This mode is used for capacitor back-up when the main power supply is cut off and long term battery back-up.

When the $\overline{STOP}$ pin input is high, executing an instruction which starts STOP mode will not place in STOP mode but instead will immediately start the release sequence (warm-up). Thus, to start STOP mode in the level-sensitive release mode, it is necessary for the program to first confirm that the $\overline{STOP}$ pin input is low. The following two methods can be used for confirmation.

1. Testing a port P20.
2. Using an external interrupt input $\overline{INT5}$ ($\overline{INT5}$ is a falling edge-sensitive input).

```
Example 1:  Starting STOP mode from NORMAL mode by testing a port P20.
            LD      (SYSCR1), 01010000B     ;   Sets up the level-sensitive release mode
SSTOPH:     TEST    (P2). 0                 ;   Wait until the STOP pin input goes low
                                                level
            JRS     F, SSTOPH
            SET     (SYSCR1). 7             ;   Starts STOP mode

Example 2:  Starting STOP mode from NORMAL mode with an INT5 interrupt.
PINT5:      TEST    (P2). 0                 ;   To reject noise, STOP mode does not
                                                start if port P20 is at high
            JRS     F, SINT5
            LD      (SYSCR1), 01010000B     ;   Sets up the level-sensitive release mode.
            SET     (SYSCR1). 7             ;   Starts STOP mode
SINT5:      RETI
```

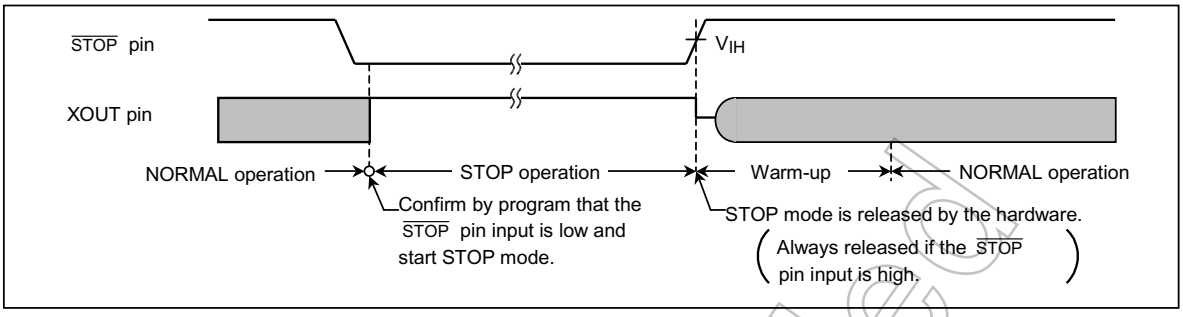


Figure 1.4.9 Level-sensitive Release Mode

Note 1: Even if the $\overline{\text{STOP}}$ pin input is low after warming up start, the STOP mode is not restarted.

Note 2: In this case of changing to the level-sensitive mode from the edge-sensitive mode, the release mode is not switched until a rising edge of the $\overline{\text{STOP}}$ pin input is detected.

b. Edge-sensitive release mode (RELM = "0")

In this mode, STOP mode is released by a rising edge of the STOP pin input. This is used in applications where a relatively short program is executed repeatedly at periodic intervals. This periodic signal (for example, a clock from a low-power consumption oscillator) is input to the $\overline{\text{STOP}}$ pin. In the edge-sensitive release mode, STOP mode is started even when the $\overline{\text{STOP}}$ pin input is high level.

```
Example:  Starting STOP mode from NORMAL mode
          LD      (SYSCR1), 10010000B     ;  Starts after specified to the edge-sensitive
                                             release mode
```

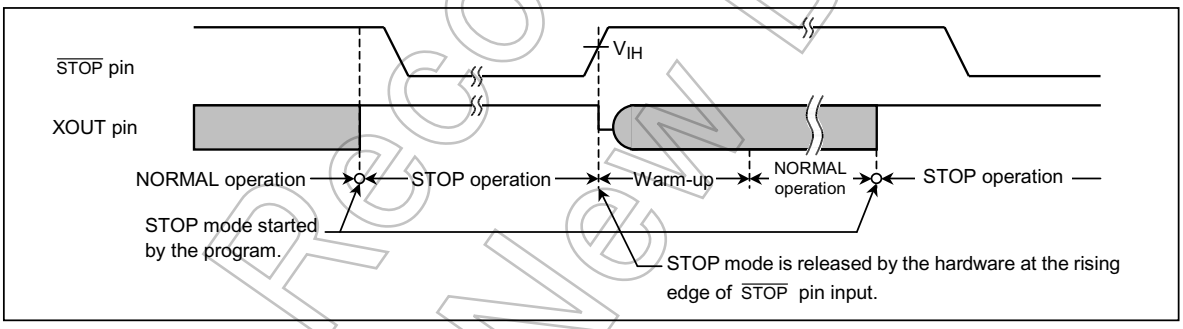


Figure 1.4.10 Edge-sensitive Release Mode

STOP mode is released by the following sequence.

1. In the dual-clock mode, when returning to NORMAL2, both the high-frequency and low-frequency clock oscillators are turned on; when returning to SLOW mode, only the low-frequency clock oscillator is turned on. In the signal-clock mode, only the high-frequency clock oscillator is turned on.
2. A warming-up period is inserted to allow oscillation time to stabilize. During warm-up, all internal operations remain halted. Four different warming-up times can be selected with the WUT (bits 2 and 3 in SYSCR1) in accordance with the resonator characteristics.

3. When the warming-up time has elapsed, normal operation resumes with the instruction following the STOP mode start instruction (e.g. [SET (SYSCR1). 7]). The start is made after the prescaler and the divider of the timing generator are cleared to "0".

Table 1.4.2 Warming-up Time Example (at fc = 12.5 MHz, fs = 32.8 kHz)

WUT	Warming-up Time [ms]		
	Return to NORMAL mode		Return to SLOW mode
	DV1CK = 0	DV1CK = 1	
00	15.729	31.457	750
01	5.243	10.486	250
10	3.932	7.864	—
11	1.311	2.621	—

Note: The warming-up time is obtained by dividing the basic clock by the divider: therefore, the warming-up time may include a certain amount of error if there is any fluctuation of the oscillation frequency when STOP mode is released. Thus, the warming-up time must be considered an approximate value.

STOP mode can also be released by inputting low level on the $\overline{\text{RESET}}$ pin, which immediately performs the normal reset operation.

Note: When STOP mode is released with a low hold voltage, the following cautions must be observed.
The power supply voltage must be at the operating voltage level before releasing STOP mode. The $\overline{\text{RESET}}$ pin input must also be "H" level, rising together with the power supply voltage. In this case, if an external time constant circuit has been connected, the $\overline{\text{RESET}}$ pin input voltage will increase at a slower pace than the power supply voltage. At this time, there is a danger that a reset may occur if input voltage level of the $\overline{\text{RESET}}$ pin drops below the non-inverting high-level input voltage (hysteresis input).

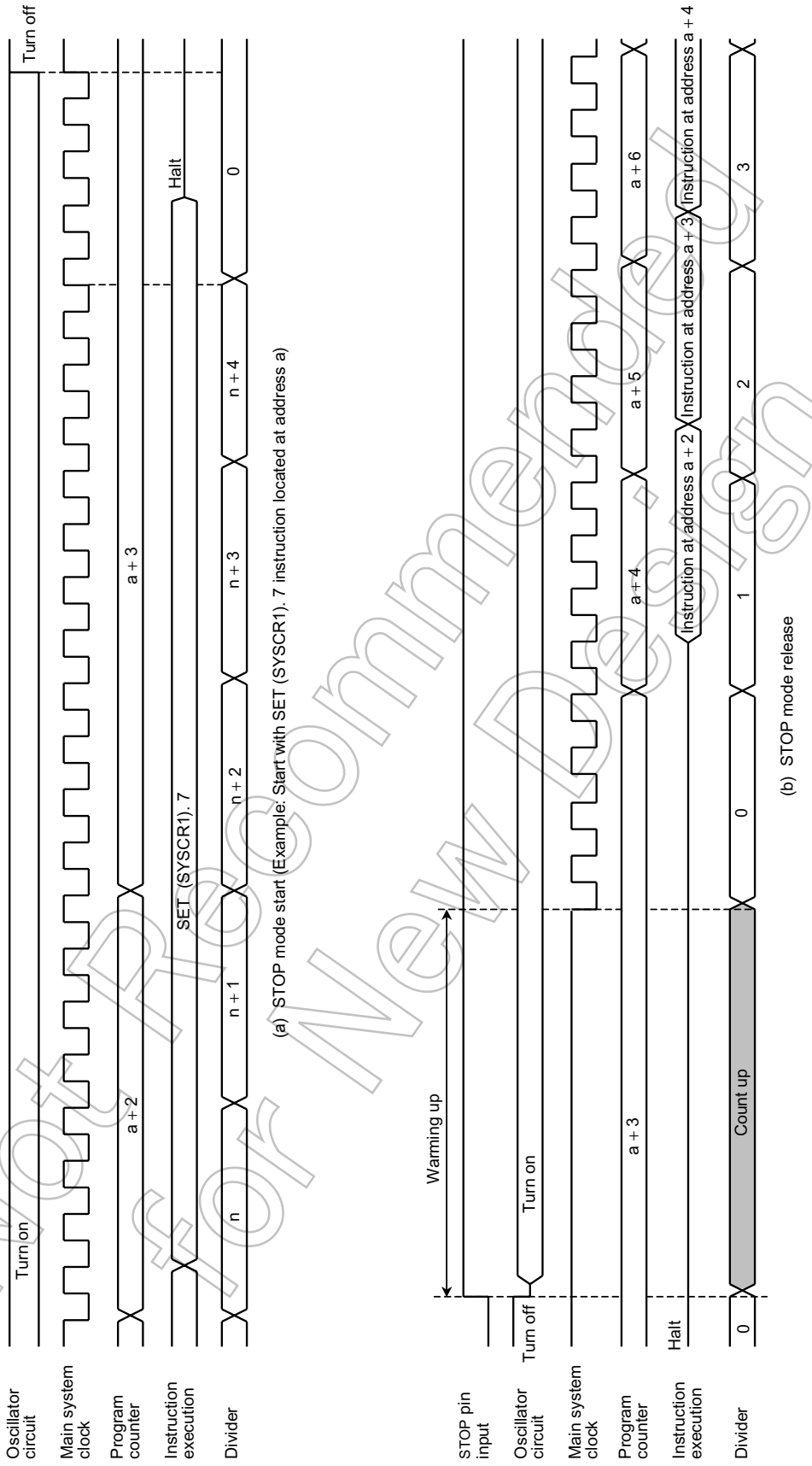


Figure 1.4.11 STOP Mode Start/Release

(2) IDLE mode (IDLE1, IDLE2, SLEEP)

IDLE mode is controlled by the system control register 2 (SYSCR2) and maskable interrupts. The following status is maintained during IDLE mode.

1. Operation of the CPU and watchdog timer (WDT) is halted. On-chip peripherals continue to operate.
2. The data memory, CPU registers, program status word and port output latches are all held in the status in effect before IDLE mode was entered.
3. The program counter holds the address of the second instruction after the instruction which starts IDLE mode.

```
Example:  Starting IDLE mode.
          SET       (SYSCR2). 4          ;  IDLE ← 1
```

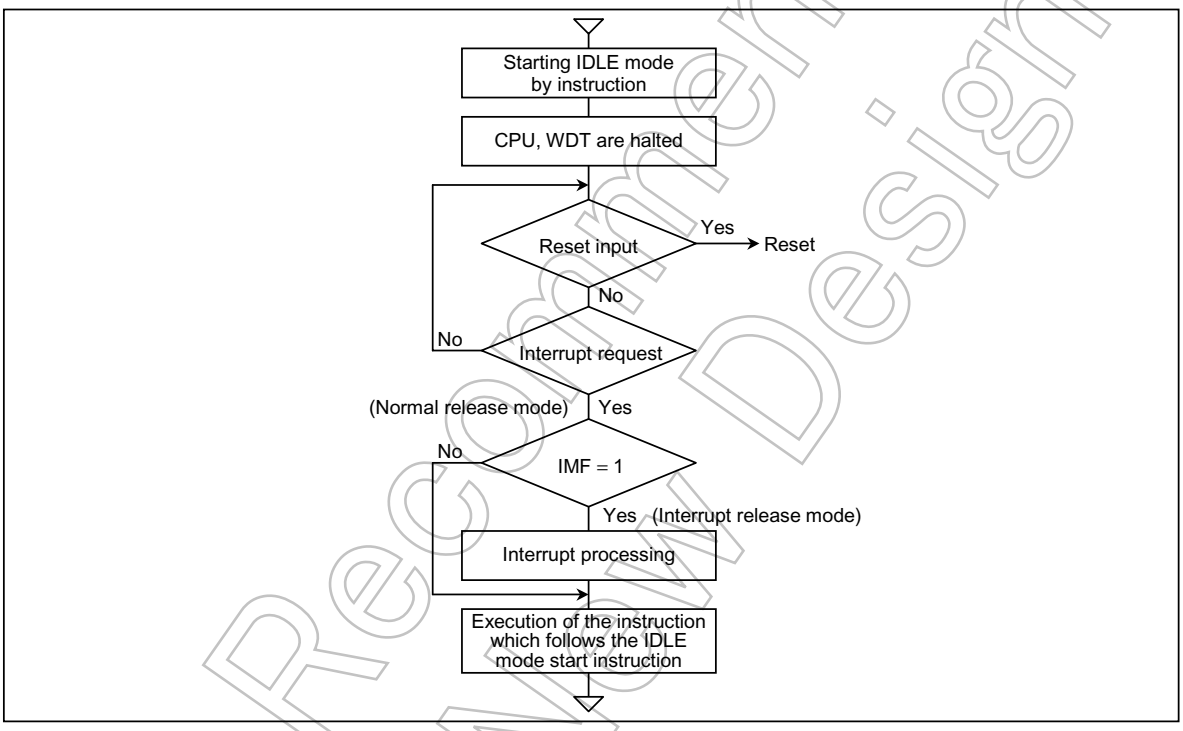


Figure 1.4.12 IDLE Mode

IDLE mode includes a normal release mode and an interrupt release mode. Selection is made with the interrupt master enable flag (IMF). Releasing IDLE mode returns from IDLE1 to NORMAL1, from IDLE2 to NORMAL2, and from SLEEP to SLOW mode.

(I) Normal release mode (IMF = "0")

IDLE mode is released by any interrupt source enabled by the individual interrupt enable flag (EF) or an external interrupt 0 ($\overline{\text{INT0}}$ pin) request. Execution resumes with the instruction following the IDLE mode start instruction (e.g. [SET (SYSCR2), 4]. The interrupt latches (IL) of the interrupt source used for releasing must be cleared to "0" by load instructions.

(II) Interrupt release mode (IMF = "1")

IDLE mode is released and interrupt processing is started by any interrupt source enabled with the individual interrupt enable flag (EF) or an external interrupt 0 ($\overline{\text{INT0}}$ pin) request. After the interrupt is processed, the execution resumes from the instruction following the instruction which starts IDLE mode.

IDLE mode can also be released by inputting low level on the $\overline{\text{RESET}}$ pin, which immediately performs the reset operation. After reset, the TMP88CU74 is placed in NORMAL 1 mode.

Note: When a watchdog timer interrupt is generated immediately before IDLE mode is started, the watchdog timer interrupt will be processed but IDLE mode will not be started.

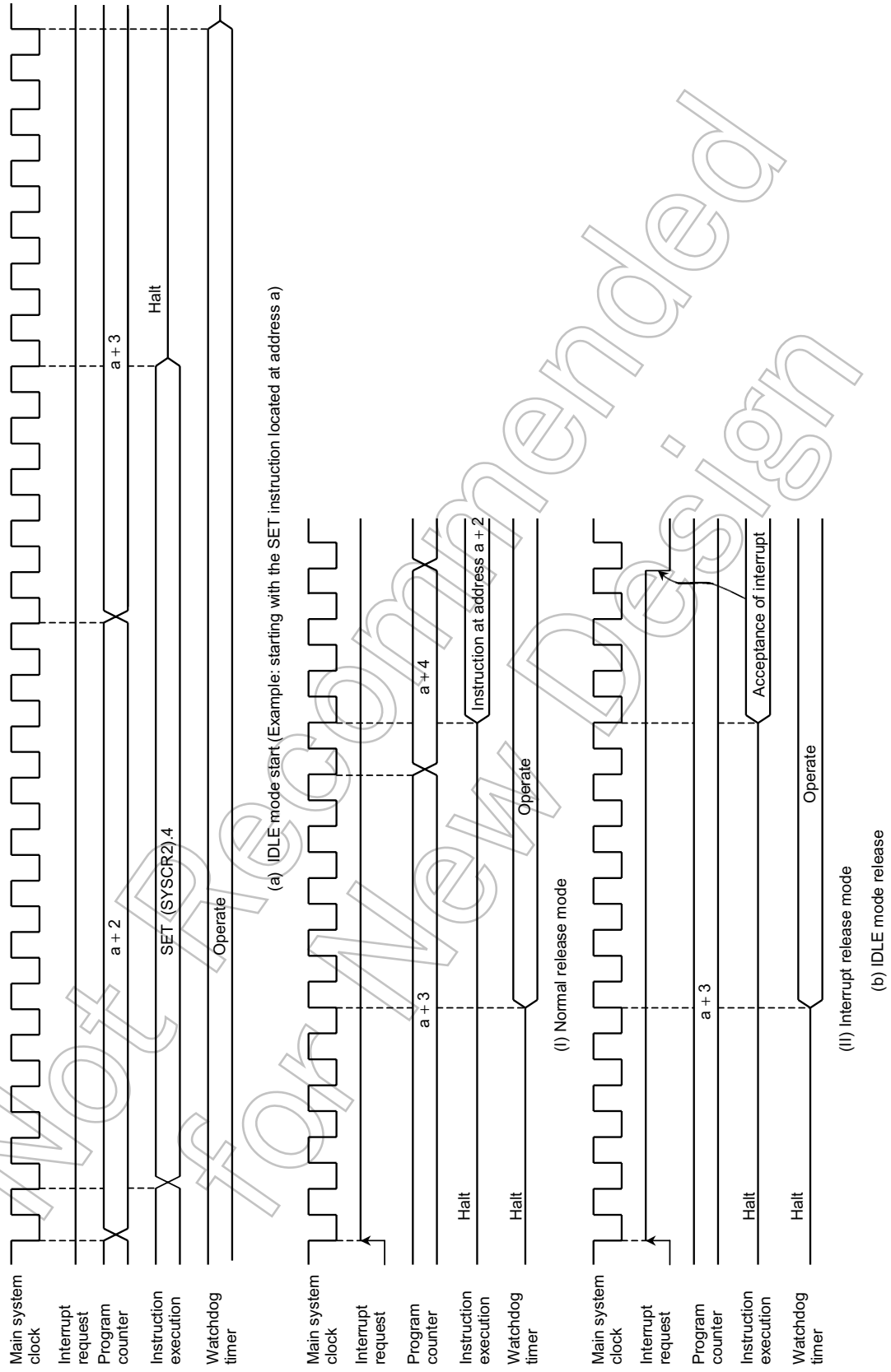


Figure 1.4.13 IDLE Mode Start/Release

(3) SLOW mode

SLOW mode is controlled by the system control register 2 (SYSCR2) and the timer/counter 2 (TC2).

a. Switching from NORMAL2 mode to SLOW mode

First, set SYSCK (bit 5 in SYSCR2) to switch the main system clock to the low-frequency clock. Next, clear XEN (bit 7 in SYSCR2) to turn off high-frequency oscillation.

Note: The high frequency clock can be continued oscillation in order to return to NORMAL2 mode from SLOW mode quickly.
Always turn off oscillation of high frequency clock when switching from SLOW mode to STOP mode.

When the low-frequency clock oscillation is unstable, wait until oscillation stabilizes before performing the above operations. The timer/counter 2 (TC2) can conveniently be used to confirm that low-frequency clock oscillation has stabilized.

Example 1: Switching from NORMAL2 mode to SLOW mode.

```
        SET     (SYSCR2). 5         ;  SYSCK ← 1 (switches the main system
                                       clock to the low-frequency clock)
        CLR     (SYSCR2). 7            XEN ← 0 (turns off high-frequency
                                       oscillation)
```

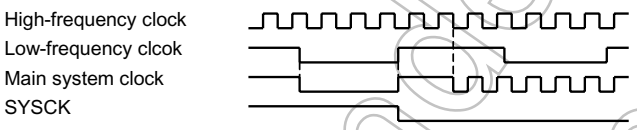
Example 2: Switching to the SLOW mode after low-frequency clock oscillation has stabilized.

```
        LD      (TC2CR), 14H        ;  Sets TC2 mode (timer mode, source
                                       clock: fs)
        LDW     (TREG2), 8000H      ;  Sets warming-up time (according to Xtal
                                       characteristics)
        SET     (EIRH). EF14        ;  Enables INTTC2
        LD      (TC2CR), 34H        ;  Starts TC2
                 :
PINTTC2: LD     (TC2CR), 10H        ;  Stops TC2
        SET     (SYSCR2). 5         ;  SYSCK ← 1 (switches the main system
                                       clock to the low-frequency clock)
        CLR     (SYSCR2). 7         ;  XEN ← 0 (turns off high-frequency
                                       oscillation)
                 :
        RETI
VINTTC2: DL     PINTTC2             ;  INTTC2 vector table
```

b. Switching from SLOW mode to NORMAL2 mode

First, set XEN (bit 7 in SYSCR2) to turn on the high-frequency oscillation. When time for stabilization (warm-up) has been taken by the timer/counter 2 (TC2), clear SYSCK (bit 5 in SYSCR2) to switch the main system clock to the high-frequency clock.

Note 1: After SYSCK is cleared to "0", executing the instructions is continued by the low-frequency clock for the period synchronized with low-frequency and high-frequency clocks.



Note 2: SLOW mode can also be released by inputting low level on the RESET pin, which immediately performs the reset operation. After reset, the TMP88CU74 is placed in NORMAL1 mode.

Example: Switching from SLOW mode to NORMAL2 mode
(fc = 12.5 MHz, warming-up time is 5.8 ms).

```
        SET     (SYSCR2). 7          ;  XEN ← 1 (turns on high-frequency
                                        oscillation)
        LD      (TC2CR), 10H         ;  Sets TC2 mode (timer mode, source
                                        clock: fc)
        LD      (TREG2 + 1), 0F8H    ;  Sets the warming-up time (according to
                                        frequency and Xtal characteristics)
        SET     (EIRH). EF14         ;  Enables INTTC2
        LD      (TC2CR), 30H         ;  Starts TC2
        :
PINTTC2: LD     (TC2CR), 10H         ;  Stops TC2
        CLR     (SYSCR2). 5          ;  SYSCK ← 0 (switches the main system
                                        clock to the high-frequency clock)
        RETI
        :
VINTTC2: DL     PINTTC2              ;  INTTC2 vector table
```

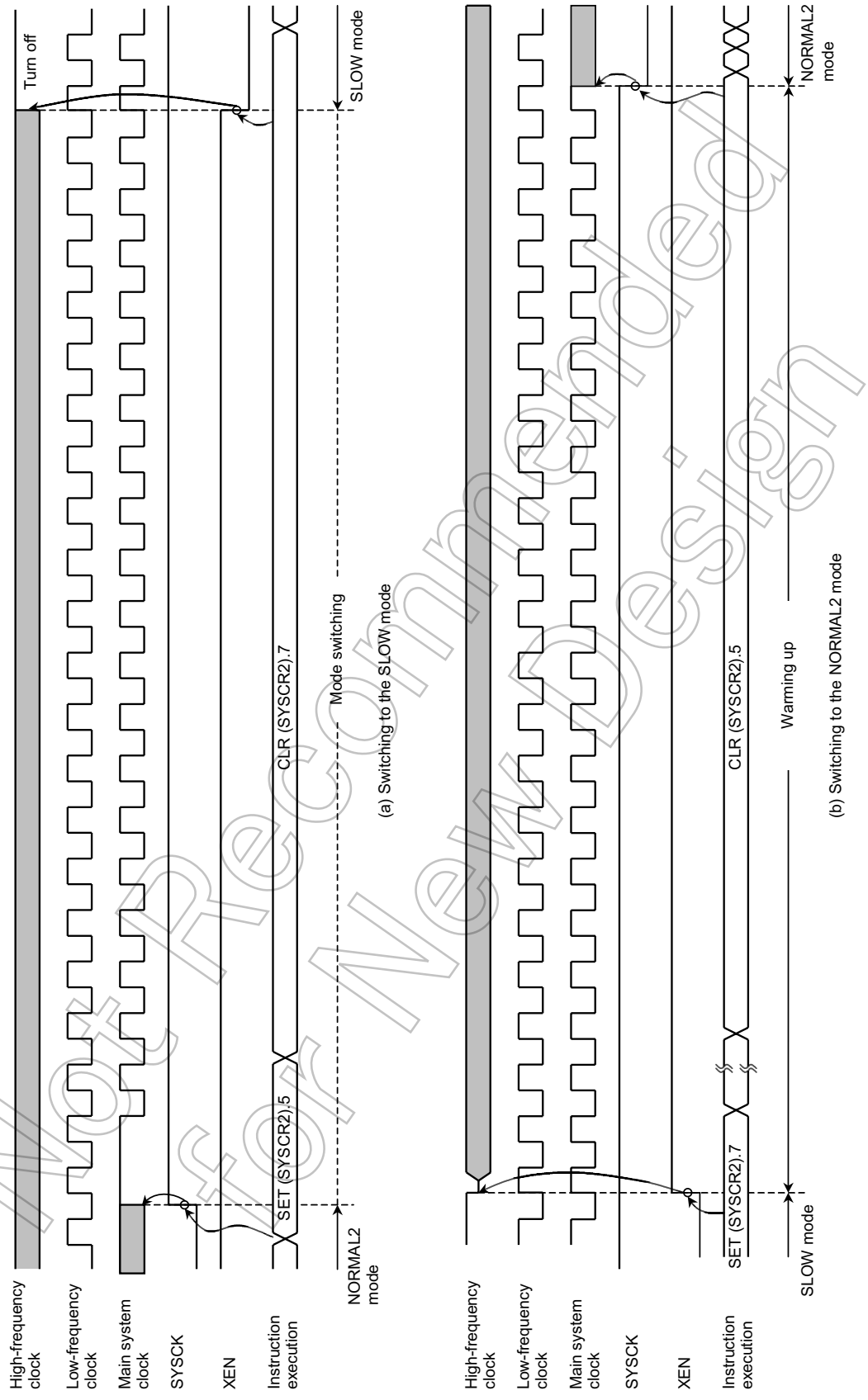


Figure 1.4.14 Switching between the NORMAL2 and SLOW Modes

1.5 Interrupt Controller

The TMP88CU74 each have a total of 15 interrupt sources: 6 externals and 9 internals. Nested interrupt control with priorities is also possible. Two of the internal sources are pseudo non-maskable interrupts; the remainder are all maskable interrupts.

Interrupt latches (IL) that hold the interrupt requests are provided for interrupt sources. Each interrupt vector is independent.

The interrupt latch is set to "1" when an interrupt request is generated and requests the CPU to accept the interrupt. The acceptance of maskable interrupts can be selectively enabled and disabled by the program using the interrupt master enable flag (IMF) and the individual interrupt enable flags (EF). When two or more interrupts are generated simultaneously, the interrupt is accepted in the highest priority order as determined by the hardware. Figure 1.5.1 shows the interrupt controller.

Table 1.5.1 Interrupt Sources

Interrupt Source			Enable Condition	Interrupt Latch	Vector Address	Priority
Internal/ External	(Reset)		Non-Maskable	—	FFFFCH	High 0
Internal	INTSW	(Software interrupt)	Pseudo non-maskable	—	FFFF8H	1
Internal	INTWDT	(Watchdog Timer interrupt)		IL2	FFFF4H	2
External	INT0	(External interrupt 0)	IMF = 1, INT0EN = 1	IL3	FFFF0H	3
Internal	INTTC1	(16-bit TC1 interrupt)	IMF・EF4 = 1	IL4	FFFECH	4
External	INT1	(External interrupt 2)	IMF・EF5 = 1	IL5	FFFE8H	5
Internal	INTTBT	(Time Base Timer interrupt)	IMF・EF6 = 1	IL6	FFFE4H	6
External	INT2	(External interrupt 2)	IMF・EF7 = 1	IL7	FFFE0H	7
Internal	INTTC3	(8-bit TC3 interrupt)	IMF・EF8 = 1	IL8	FFFDCH	8
Internal	INTSIO1	(Serial Interface1 interrupt)	IMF・EF9 = 1	IL9	FFFD8H	9
Internal	INTTC4	(8-bit TC4 interrupt)	IMF・EF10 = 1	IL10	FFFD4H	10
External	INT3	(External interrupt 3)	IMF・EF11 = 1	IL11	FFFD0H	11
Internal	INTKEY	(Key scan interrupt)	IMF・EF12 = 1	IL12	FFFCCH	12
Internal	INTSIO2	(Serial interface2 interrupt)	IMF・EF13 = 1	IL13	FFFC8H	13
Internal	INTTC2	(16-bit TC2 interrupt)	IMF・EF14 = 1	IL14	FFFC4H	14
External	INT5	(External interrupt 5)	IMF・EF15 = 1	IL15	FFFC0H	Low 15

Note: Before you change each enable flag (EF) and/or each interrupt latch (IL), be sure to clear the interrupt master enable flag (IMF) to "0" (to disable interrupts).

a. After a DI instruction is executed

b. When an interrupt is accepted, IMF is automatically cleared to "0". However, to enable nested interrupts, change EF and/or IL before setting IMF to "1" (to enable interrupts).

If the individual enable flags (EF) and interrupt latches (IL) are set under conditions other than the above, the proper operation cannot be guaranteed.

(1) Interrupt latches (IL15 to 2)

Interrupt latches are provided for each source, except for a software interrupt. The latch is set to "1" when an interrupt request is generated, and requests the CPU to accept the interrupt. The latch is cleared to "0" just after the interrupt is accepted. All interrupt latches are initialized to "0" during reset.

Interrupt latches are assigned to addresses 003CH and 003DH in the SFR. Each latch can be cleared to "0" individually by an instruction; however, the read-modify-write instruction such as bit manipulation or operation instructions cannot be used. Thus, interrupt requests can be canceled and initialized by the program. Note that interrupt latches cannot be set to "1" by any instruction.

The contents of interrupt latches can be read out by an instruction. Therefore, testing interrupt requests by software is possible.

```
Example 1:  Clears interrupt latches
            LDW     (ILL), 1110100000111111B    ;   IL12, IL10 to IL6 ← 0

Example 2:  Reads interrupt latches
            LD      WA, (ILL)                   ;   W ← ILH, A ← ILL

Example 3:  Tests an interrupt latch
            TEST    (IL).7                      ;   if IL7 = 1 then jump
            JR      F, SSET
```

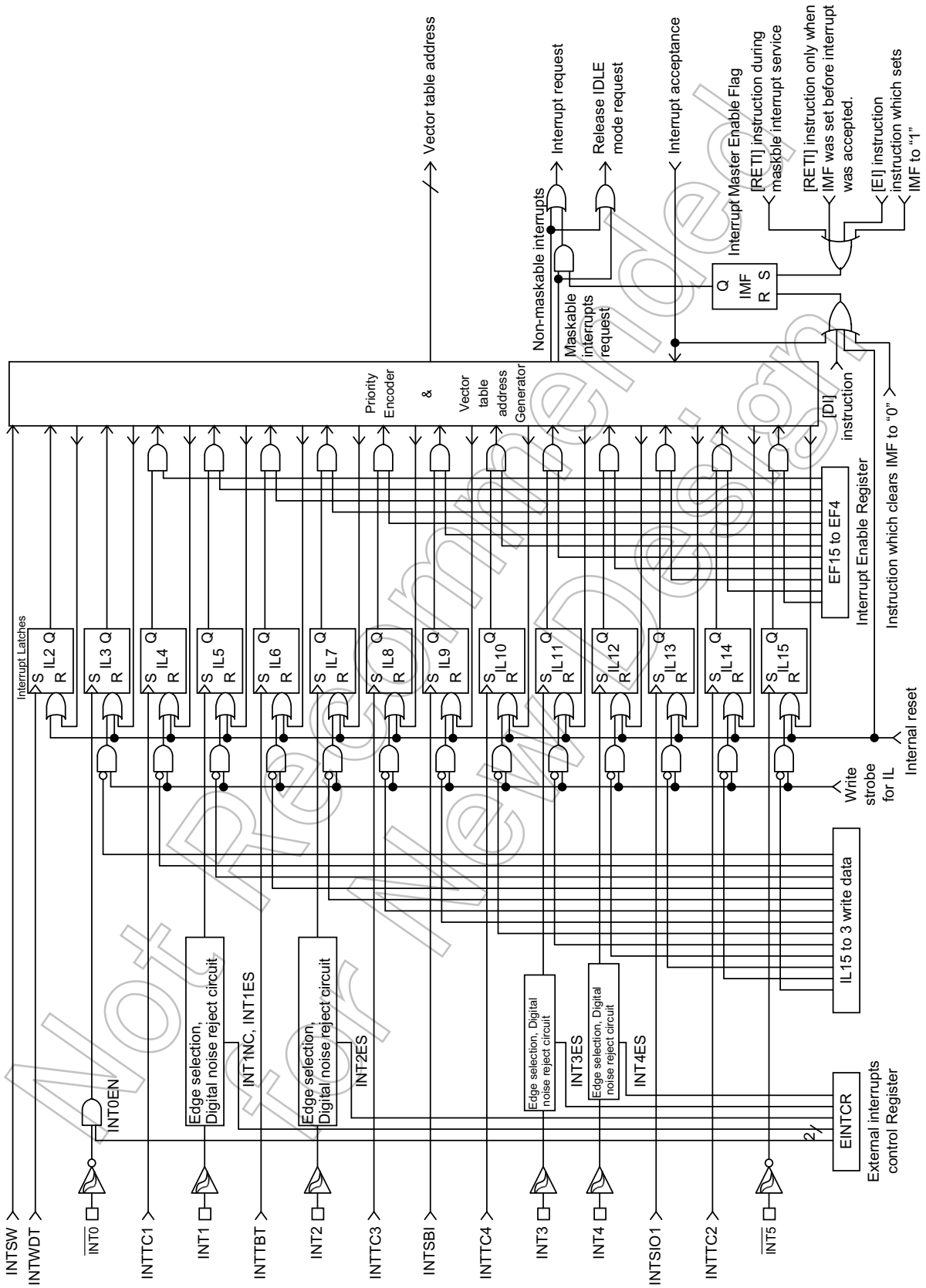


Figure 1.5.1 Interrupt Controller Block Diagram

(2) Interrupt enable register (EIR)

The interrupt enable register (EIR) enables and disables the acceptance of interrupts, except for the pseudo non-maskable interrupts (software and watchdog timer interrupts). Pseudo non-maskable interrupts are accepted regardless of the contents of the EIR; however, the pseudo non-maskable interrupts cannot be nested more than once at the same time. For example, the watchdog timer interrupt is not accepted during the software interrupt service.

The EIR consists of an interrupt master enable flag (IMF) and the individual interrupt enable flags (EF). This register is assigned to addresses 0003AH and 0003BH in the SFR, and can be read and written by an instruction (including read-modify-write instructions such as bit manipulation instructions).

1. Interrupt master enable flag (IMF)

The interrupt master enable flag (IMF) enables and disables the acceptance of all interrupts, except for pseudo non-maskable interrupts. Clearing this flag to "0" disables the acceptance of all maskable interrupts. Setting to "1" enables the acceptance of interrupts.

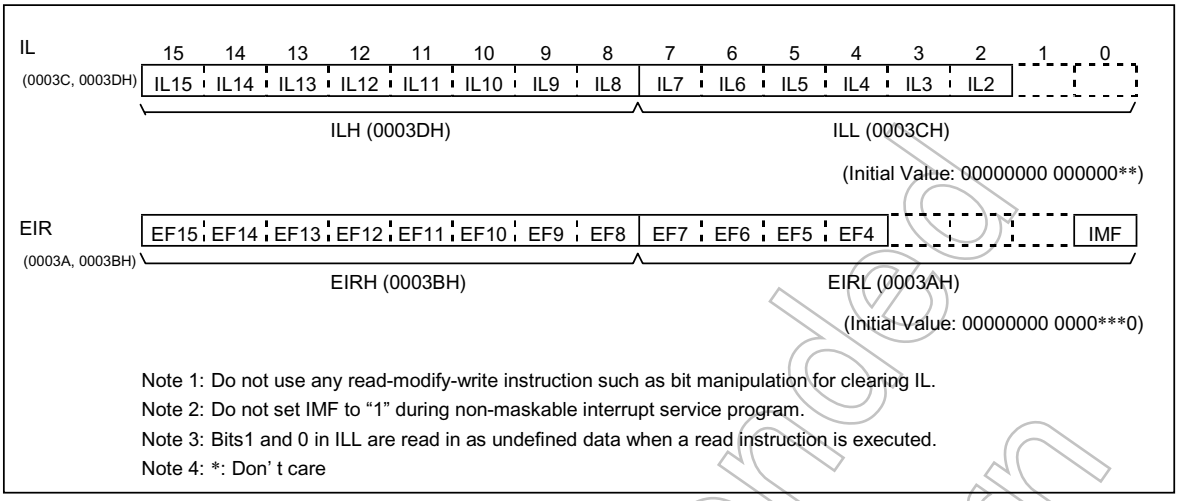
When an interrupt is accepted, this flag is cleared to "0" to temporarily disable the acceptance of maskable interrupts. After execution of the interrupt service program, this flag is set to "1" by the maskable interrupt return instruction [RETI] to again enable the acceptance of interrupts. If an interrupt request has already been occurred, interrupt service starts immediately after execution of the [RETI] instruction.
Pseudo non-maskable interrupts are returned by the [RETN] instruction. In this case, the IMF is set to "1" only when pseudo non-maskable interrupt service is started with interrupt acceptance enabled (IMF = 1). Note that the IMF remains "0" when cleared by the interrupt service program.
The IMF is assigned to bit 0 at address 0003AH in the SFR, and can be read and written by an instruction. The IMF is normally set and cleared by the [EI] and [DI] instructions, and the IMF is initialized to "0" during reset.

2. Individual interrupt enable flags (EF15 to EF4)

These flags enable and disable the acceptance of individual maskable interrupts, except for an external interrupt 0. Setting the corresponding bit of an individual interrupt enable flag to "1" enables acceptance of an interrupt, setting the bit to "0" disables acceptance.

```
Example 1:  Sets EF for individual interrupt enable, and sets IMF to "1".
            LDW     (EIRL), 1110100010100001B   ;   EF15 to EF13, EF11, EF7, EF5,
                                                    IMF ← 1

Example 2:  Sets an individual interrupt enable flag to "1".
            SET     (EIRH).4                    ;   EF12 ← 1
```



Note 1: Do not use any read-modify-write instruction such as bit manipulation for clearing IL.
Note 2: Do not set IMF to "1" during non-maskable interrupt service program.
Note 3: Bits1 and 0 in ILL are read in as undefined data when a read instruction is executed.
Note 4: *: Don' t care

Figure 1.5.2 Interrupt latch (IL) and interrupt enable register (EIR)

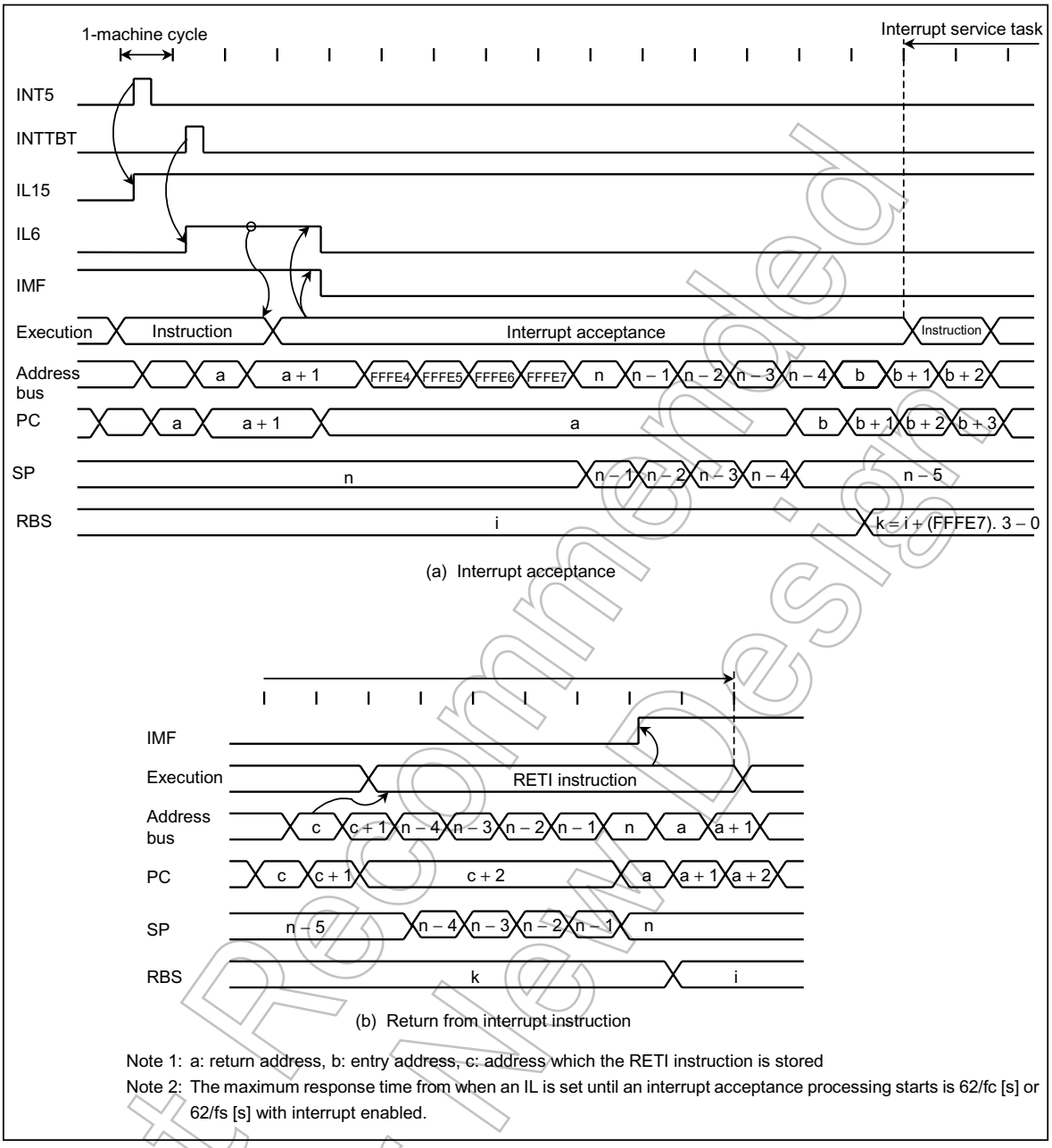
1.5.1 Interrupt Sequence

An interrupt request is held until the interrupt is accepted or the interrupt latch is cleared to "0" by a reset or an instruction. Interrupt acceptance sequence requires 12 machine cycles (3.84 μs at fc = 12.5 MHz in the NORMAL mode) after the completion of the current instruction execution. The interrupt service task terminates upon execution of an interrupt return instruction [RETI] (for maskable interrupts) or [RETN] (for pseudo non-maskable interrupts). Figure 1.5.3 shows the timing chart of interrupt acceptance processing.

(1) Interrupt acceptance

Interrupt acceptance processing is as follows.

1. The interrupt master enable flag (IMF) is cleared to "0" to temporarily disable the acceptance of any following maskable interrupts. When a non-maskable interrupt is accepted, the acceptance of any following interrupts is temporarily disabled.
2. The interrupt latch (IL) for the interrupt source accepted is cleared to "0".
3. The contents of the program counter (return address) and the program status word (PSW) are saved (pushed) on the stack in sequence of PSWH, PSWL, PCE, PCH, PCL. The stack pointer (SP) is decremented five times.
4. The entry address of the interrupt service program is read from the vector table, and set to the program counter.
5. The RBS control code is read from the vector table. The lower 4-bit of this code is added to the RBS.
6. The instruction stored at the entry address of the interrupt service program is executed.

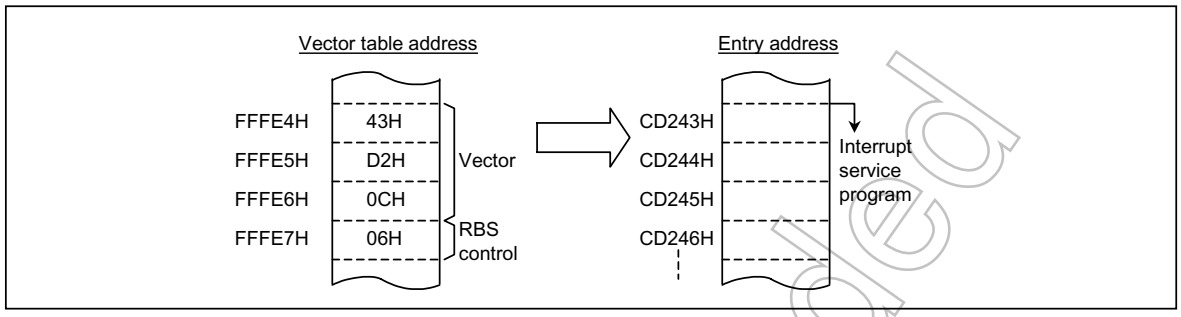


Note 1: a: return address, b: entry address, c: address which the RETI instruction is stored

Note 2: The maximum response time from when an IL is set until an interrupt acceptance processing starts is 62/fc [s] or 62/fs [s] with interrupt enabled.

Figure 1.5.3 Timing Chart of Interrupt Acceptance and Interrupt Return Instruction

Example: Correspondence between vector table address for INTTBT and the entry address of the interrupt service program.



A maskable interrupt is not accepted until the IMF is set to "1" even if the maskable interrupt higher than the level of current servicing interrupt is occurred.

When nested interrupt service is necessary, the IMF is set to "1" in the interrupt service program. In this case, acceptable interrupt sources are selectively enabled by the individual interrupt enable flags. However, an acceptance of external interrupt 0 cannot be disabled by the EF; therefore, if disable is necessary, either the external interrupt function of the $\overline{INT0}$ pin must be disabled with the INT0EN in the external interrupt control register (EINTCR) (the interrupt latch IL3 is not set at INT0EN = 0, therefore, the rising edge of $\overline{INT0}$ pin input can not be detected.) or an interrupt processing must be avoided by the program.

Example 1: Disables an external interrupt 0 using the INT0EN

```
LD      (EINTCR), 00000000B     ;   INT0EN ← 0
```

Example 2: Disables the processing of external interrupt 0 under the software control (using bit 0 at address 000F0H as the interrupt processing disable switch)

```
PINT0:  TEST    (000F0H). 0     ;   Return without interrupt processing if (000F0H)0 = 1
        JRS     T, SINT0
        RETI
SINT0:  Interrupt processing
        RETI
VINT0:  DL      PINT0
```

(2) Saving/Restoring general-purpose registers

During interrupt acceptance processing, the program counter (PC) and the program status word (PSW) are automatically saved on the stack, but not the accumulator and other registers. These registers are saved by the program if necessary. Also, when nesting multiple interrupt services, it is necessary to avoid using the same data memory area for saving registers.

The following method is used to save/restore the general-purpose registers.

1. General-purpose register save/restore by automatic register bank changeover

   The general-purpose registers can be saved at high-speed by switching to a register bank that is not in use. Normally, the bank 0 is used for the main task and the banks 1 to 15 are assigned to interrupt service tasks. To increase the efficiency of data memory utilization, the same bank is assigned for interrupt sources which are not nested.

   The switched bank is automatically restored by executing an interrupt return instruction [RETI] or [RETN]. Therefore, it is not necessary for a program to save the RBS.

```
Example:  Register bank changeover
PINTxx:   Interrupt processing
          RETI
          :
VINTxx:   DP    PINTxx
          DB    1              ;  RBS ← RBS + 1
```

2. General-purpose register save/restore by register bank changeover

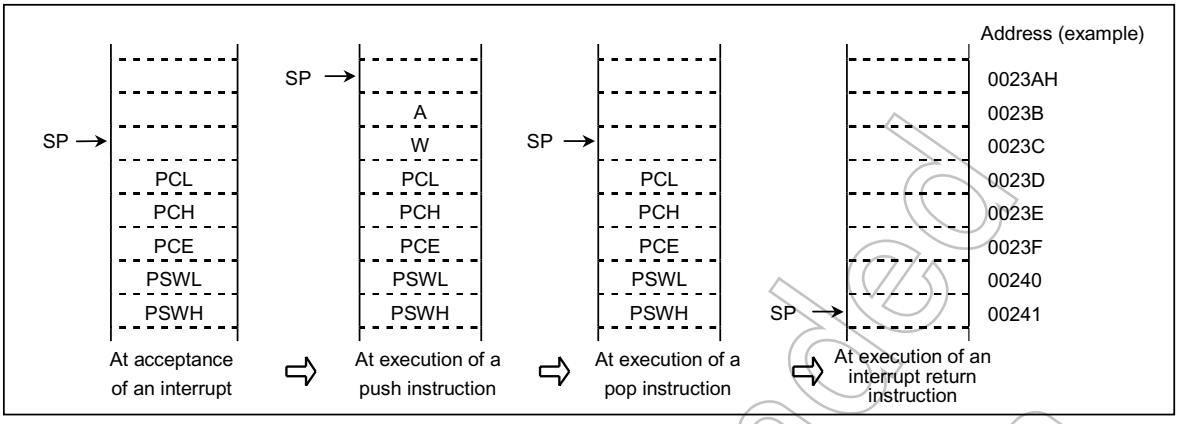
   The general-purpose registers can be saved at high-speed by switching to a register bank that is not in use. Normally, the bank 0 is used for the main task and the banks 1 to 15 are assigned to interrupt service tasks.

```
Example:  Register bank changeover
PINTxx:   LD    RBS, n
          Interrupt processing
          RETI                   ;  Restores bank and Returns
          :
VINTxx:   DP    PINTxx           ;  Interrupt service routine entry address
          DB    0
```

3. General-purpose registers save/restore using push and pop instructions

   To save only a specific register, and when the same interrupt source occurs more than once, the general-purpose registers can be saved/restored using the push/pop instructions.

```
Example:  Register save/restore using push and pop instructions
PINTxx:   PUSH  WA               ;  Save WA register pair
          Interrupt processing
          POP   WA               ;  Restore WA register pair
          RETI                   ;  Return
```



4. General-purpose registers save/restore using data transfer instructions

Data transfer instruction can be used to save only a specific general-purpose register during processing of single interrupt.

```
Example:  Saving/restoring a register using data transfer instructions
PINTxx:   LD        (GSAVA), A      ;  Save A register
          Interrupt processing
          LD        A, (GSAVA)      ;  Restore A register
          RETI                      ;  Return
```

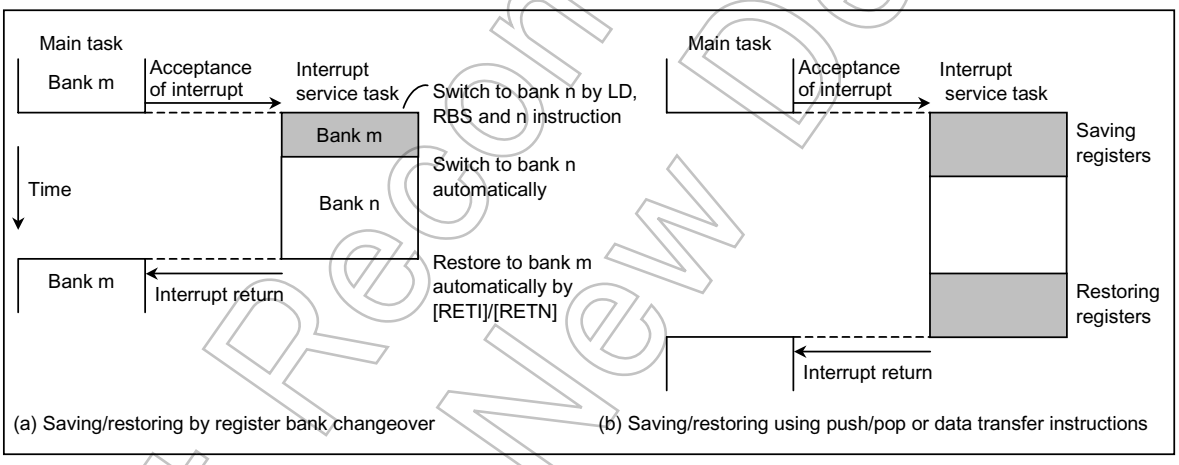


Figure 1.5.4 Saving/Restoring General-purpose Registers

(3) Interrupt return

The interrupt return instructions [RETI]/[RETN] perform the following operations.

[RETI] Maskable interrupt return	[RETN] Non-maskable interrupt return
1. The contents of the program counter and the program status word are restored from the stack.	1. The contents of the program counter and program status word are restored from the stack.
2. The stack pointer is incremented 5 times.	2. The stack pointer is incremented 5 times.
3. The interrupt master enable flag is set to "1".	3. The interrupt master enable flag is set to "1" only when a non-maskable interrupt is accepted in interrupt enable status. However, the interrupt master enable flag remains at "0" when so clear by an interrupt service program.
4. The interrupt nesting counter is decremented, and the interrupt nesting flag is changed.	4. The interrupt nesting counter is decremented, and the interrupt nesting flag is changed.

Interrupt requests are sampled during the final cycle of the instruction being executed. Thus, the next interrupt can be accepted immediately after the interrupt return instruction is executed.

Note: When the interrupt processing time is longer than the interrupt request generation time, the interrupt service task is performed but not the main task.

1.5.2 Software Interrupt (INTSW)

Executing the [SWI] instruction generates a software interrupt and immediately starts interrupt processing (INTSW is highest prioritized interrupt). However, if processing of a non-maskable interrupt is already underway, executing the SWI instruction will not generate a software interrupt but will result in the same operation as the [NOP] instruction.

Use the [SWI] instruction only for detection of the address error or for debugging.

Note: To use the SWI instruction for software break in the development tool, software interrupt always generates even if the non-maskable interrupt is in progress.

1. Address error detection

FFH is read if for some cause such as noise the CPU attempts to fetch an instruction from a non-existent memory address. Code FFH is the SWI instruction, so a software interrupt is generated and an address error is detected. The address error detection range can be further expanded by writing FFH to unused areas of the program memory. Address-trap reset is generated in case that an instruction is fetched from RAM or SFR areas.

2. Debugging

Debugging efficiency can be increased by placing the SWI instruction at the software break point setting address.

1.5.3 External Interrupts

The TMP88CU74 has six external interrupt inputs ($\overline{INT0}$, INT1, INT2, INT3, INT4, $\overline{INT5}$). Four of these are equipped with digital noise reject circuits(pulse inputs of less than a certain time are eliminated as noise).

Edge selection is also possible with INT1 to INT4. The $\overline{INT0}$/P10 pin can be configured as either an external interrupt input pin or an input/output port, and is configured as an input port during reset.

Edge selection, noise reject control and $\overline{INT0}$/P10 pin function selection are performed by the external interrupt control register (EINTCR).

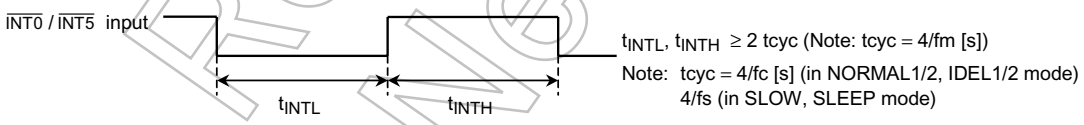
Table 1.5.2 External Interrupts

Source	Pin	Secondary Function Pin	Enable Conditions	Edge	Digital Noise Reject
INT0	$\overline{INT0}$	P10	IMF = 1, INT0EN = 1	Falling edge	— (hysteresis input)
INT1	INT1	P11	IMF・EF5 = 1	Falling edge or Rising edge	Pulses of less than 15/fc or 63/fc [s] are eliminated as noise. Pulses of 48/fc or 192/fc [s] or more are considered to be signals.
INT2	INT2	P16	IMF・EF7 = 1		Pulses of less than 7/fc [s] are eliminated as noise. Pulses of 24/fc [s] or more are considered to be signals.
INT3	INT3	P15/TC1	IMF・EF11 = 1		
INT4	INT4	P17/TC3	IMF・EF12 = 1		
INT5	$\overline{INT5}$	P20/$\overline{STOP}$	IMF・EF15 = 1	Falling edge	— (hysteresis input)

Note 1: The noise reject function is turned off in SLOW and SLEEP modes. Also, the noise reject times are not constant for pulses input while transiting between operating modes.

Note 2: The noise reject function is also affected for timer/counter input (TC1 pin, TC3 pin).

Note 3: The pulse width (both "H" and "L"evel) for input to the $\overline{INT0}$ and $\overline{INT5}$ pins must be over 2 machine cycle.



Note 4: If a noiseless signal is input to the external interrupt pin in the NORMAL 1/2 or IDLE 1/2 mode, the maximum time from the edge of input signal until the IL is set is as follows:

1. INT1 pin 49/fc [s] (INT1NC = 1), 193/fc [s] (INT1NC = 0)
2. INT2 pin 25/fc [s]

Note 5: Even if the falling edge of $\overline{INT0}$ pin input is detected at INT0EN = 0, the interrupt latch IL3 is not set.

EINTCR (00037H)

7	6	5	4	3	2	1	0
INT1NC	INT0EN		INT4ES	INT3ES	INT2ES	INT1ES	

(Initial value: 00*0 000*)

INT1NC	Noise reject time select	0: Pulses of less than 63/fc [s] are eliminated as noise 1: Pulses of less than 15/fc [s] are eliminated as noise	R/W
INT0EN	P10/$\overline{\text{INT0}}$ pin configuration	0: P10 input/output port 1: $\overline{\text{INT0}}$ pin (Port P10 should be set to an input mode)	
INT4ES INT3ES INT2ES INT1ES	INT4 to INT1 edge select	0: Rising edge 1: Falling edge	

Note 1: fc: High-frequency clock [Hz], *: Don't care

Note 2: Edge detection during switching edge selection is invalid.

Note 3: Do not change EINTCR only when IMF = 0. After changing EINTCR, interrupt latches of external interrupt inputs must be cleared to "0" using load instruction.

Note 4: In order to change of external interrupt input by rewriting the contents of INT2ES and INT3ES and INT4ES during NORMAL 1/2 mode, clear interrupt latches of external interrupt inputs (INT2 and INT3 and INT4) after 8 machine cycles from the time of rewriting. During SLOW mode, 3 machine cycles are required.

Figure 1.5.5 External Interrupt Control Register

1.6 Reset Circuit

The TMP88CU74 has four types of reset generation procedures: an external reset input, an address trap reset output, a watchdog timer reset output and a system clock reset output. Table 1.6.1 shows on-chip hardware initialization by reset action.

The malfunction reset output circuit such as watchdog timer reset, address trap reset and system clock reset is not initialized when power is turned on. The $\overline{\text{RESET}}$ pin can output level "L" at the maximum 24/fc [s] (1.92 μs at 12.5 MHz) when power is turned on.

Table 1.6.1 Initializing Internal Status by Reset Action

On-chip Hardware	Initial Value	On-chip Hardwear	Initial Value
Program counter (PC)	(FFFFEH to FFFFCH)	Prescaler and Divider of timing generator	0
Stack pointer (SP)	Not initialized		
General-purpose registers (W, A, B, C, D, E, H, L)	Not initialized		
Register bank selector (RBS)	0	Watchdog timer	Enable
Jump status flag (JF)	1		
Zero flag (ZF)	Not initialized	Output latches of I/O ports	Refer to I/O port circuitry
Carry flag (CF)	Not initialized		
Half carry flag (HF)	Not initialized		
Sign flag (SF)	Not initialized		
Overflow flag (VF)	Not initialized		
Interrupt master enable flag (IMF)	0		
Interrupt individinal enable flags (EF)	0	Control registers	Refer to each of control register
Interrupt latchs (IL)	0		

1.6.1 External Reset Input

The $\overline{\text{RESET}}$ pin contains a Schmitt trigger (hysteresis) with an internal pull-up resistor.

When the $\overline{\text{RESET}}$ pin is held at "L" level for at least 3 machine cycles (12/fc [s]) with the power supply voltage within the operating voltage range and oscillation stable, a reset is applied and the internal state is initialized.

When the $\overline{\text{RESET}}$ pin input goes high, the reset operation is released and the program execution starts at the vector address stored at addresses FFFFC to FFFFEH.

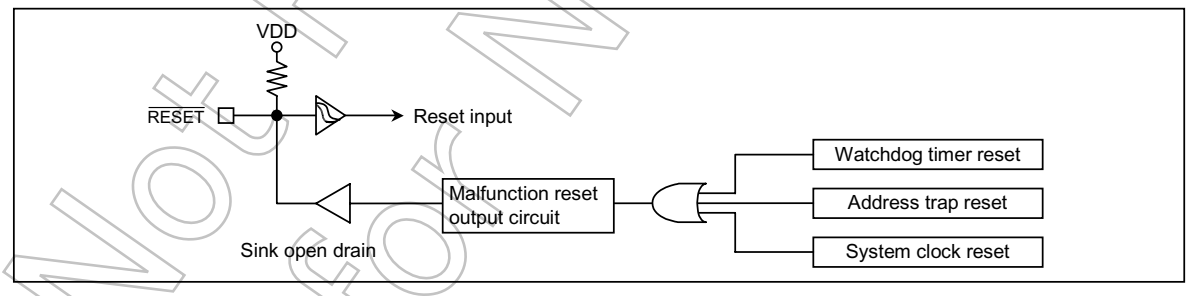
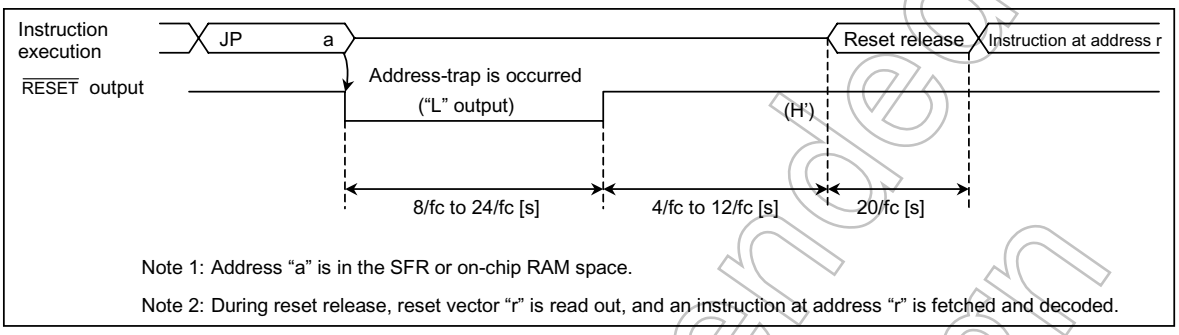


Figure 1.6.1 Reset Circuit

1.6.2 Address-trap-reset

If the CPU should start looping for some cause such as noise and an attempt be made to fetch an instruction from the on-chip RAM or the SFR area, an address-trap-reset will be generated. Then, the $\overline{\text{RESET}}$ pin output will go low. The reset time is about 8/fc to 24/fc [s] (0.64 to 1.92 μs at 12.5 MHz).



Note 1: Address "a" is in the SFR or on-chip RAM space.

Note 2: During reset release, reset vector "r" is read out, and an instruction at address "r" is fetched and decoded.

Figure 1.6.2 Address-trap-reset

1.6.3 Watchdog Timer Reset

Refer to Section "2.4 Watchdog Timer".

1.6.4 System-clock-reset

Clearing both XEN and XTEN (bits 7 and 6 in SYSCR2) to "0", clearing XEN to "0" when SYSCK = 0, or clearing XEN to "0" when SYSCK = 1 stops system clock, and causes the microcomputer to deadlock. This can be prevented by automatically generating a reset signal whenever XEN = XTEN = 0 is detected to continue the oscillation. Then, the $\overline{\text{RESET}}$ pin output goes low from high-impedance. The reset time is about 8/fc to 24/fc [s] (0.64 to 1.92 μs at 12.5 MHz).

2. On-Chip Peripheral Functions

2.1 Special Function Registers (SFR)

The TMP88CU74 uses the memory mapped I/O system, and all peripheral control and data transfers are performed through the special function registers (SFR).

The SFR are mapped to addresses 00000H to 0003FH, and DBR are mapped to address 00F80H to 00FFFH.

Figure 2.1.1 shows the TMP88CU74 SFR and DBR.

Address	Read	Write
00000H	P0 port	
01	P1 port	
02	P2 port	
03	P3 port	
04	P4 port	
05	P5 port	
06	P6 port	
07	P7 port	
08	P8 port	
09	P9 port	
0A	–	P0CR (P0 port I/O output control)
0B	–	P1CR (P1 port I/O output control)
0C	–	P4CR (P4 port I/O output control)
0D	–	P5CR (P5 port I/O output control)
0E	ADCCR (AD converter control)	
0F	ADCDR (AD conv.result)	–
10	–	TREG1AL (Timer register 1A)
11	–	TREG1AH
12	TREG1BL (Timer register1B)	
13	TREG1BH	
14	–	TC1CR (TC1 control)
15	–	TC2CR (TC2 control)
16	–	TREG2L (Timer register 2)
17	–	TREG2H
18	TREG3A (Timer register 3A)	
19	TREG3B (Timer register 3B)	–
1A	–	TC3CR (TC3 control)
1B	–	TREG4 (Timer register 4)
1C	–	TC4CR (TC4 control)
1D	PD port	
1E	Reserved	
1F	Reserved	

Address	Read	Write
00020H	–	SBICR1 (SBI control register)
21	SBIDBR (SBI Data buffer)	
22	–	I^2CAR (I^2C Bus address)
23	SBISR (SBI status)	SBICR2 (SBI control register)
24	Reserved	
25	Reserved	
26	Reserved	
27	SIO1SR (SIO status)	SIO1CR1 (SIO1 control 1)
28		SIO1CR2 (SIO1 control 2)
29	VFTSR (VFT status)	VFTCR1 (VFT control 1)
2A		VFTCR2 (VFT control 2)
2B		P3CR (Port I/O control)
2C	Reserved	
2D	Reserved	
2E	Reserved	
2F	Reserved	
30	DVCR	
31	Reserved	
32	Reserved	
33	Reserved	
34	–	WDTCR1 (Watch-dog timer control)
35	–	WDTCR2
36	TBTCR (TBT/TG/DVO control)	
37	EINTCR (External interrupt control)	
38	SYSCR1 (system control)	
39	SYSCR2	
3A	EIRL (Interrupt enable register)	
3B	EIRH	
3C	ILL (Interrupt latch)	
3D	ILH	
3E	PSWL (program status word)	
3F	PSWH	

(a) Special function registers

Note 1: Do not access reserved areas by the program.

Note 2: −: Cannot be accessed.

Note 3: Write-only registers and interrupt latches cannot use the read-modify-write instructions (bit manipulation instructions such as SET, CLR, etc. and logical operation instructions such as AND, OR, etc.).

Note 4: When defining address 0003FH with assembler symbols, use GRBS.
Address 0003EH must be GPSW/GFLAG.

Figure 2.1.1 SFR and DBR (1/2)

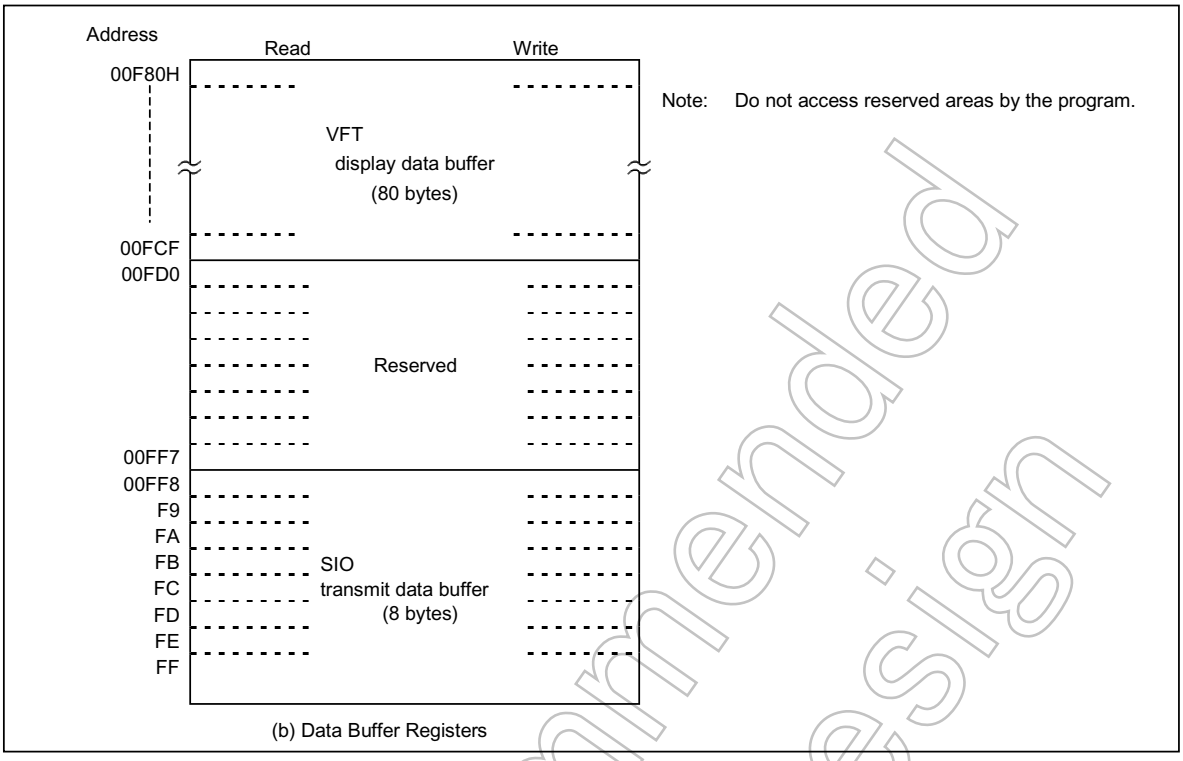


Figure 2.1.2 SFR and DBR (2/2)

2.2 I/O Ports

The TMP88CU74 each have 11 parallel input/output ports (71 pins) each as follows:

1.	Port P0	8-bit I/O port	Serial port input/output
2.	Port P1	8-bit I/O port	External interrupt input, timer/counter input, and divider output
3.	Port P2	3-bit I/O port	Low-frequency resonator connections, external interrupt input, and STOP mode release signal input
4.	Port P3	3-bit I/O port	Serial bus interface input/output
5.	Port P4	8-bit I/O port	Anarog input
6.	Port P5	4-bit I/O port	Anarog input
7.	Port P6	8-bit I/O port	VFT output
8.	Port P7	8-bit I/O port	VFT output
9.	Port P8	8-bit I/O port	VFT output
10.	Port P9	8-bit I/O port	VFT output
11.	Port PD	5-bit I/O port	VFT output

Each output port contains a latch, which holds the output data. Input ports excluding do not have latches, so the external input data should either be held externally until read or reading should be performed several times before processing. Figure 2.2.1 shows input/output timing examples.

External data is read from an I/O port in the S1 state of the read cycle during execution of the read instruction. This timing can not be recognized from outside, so that transient input such as chattering must be processed by the program.

Output data changes in the S2 state of the write cycle during execution of the instruction which writes to an I/O port.

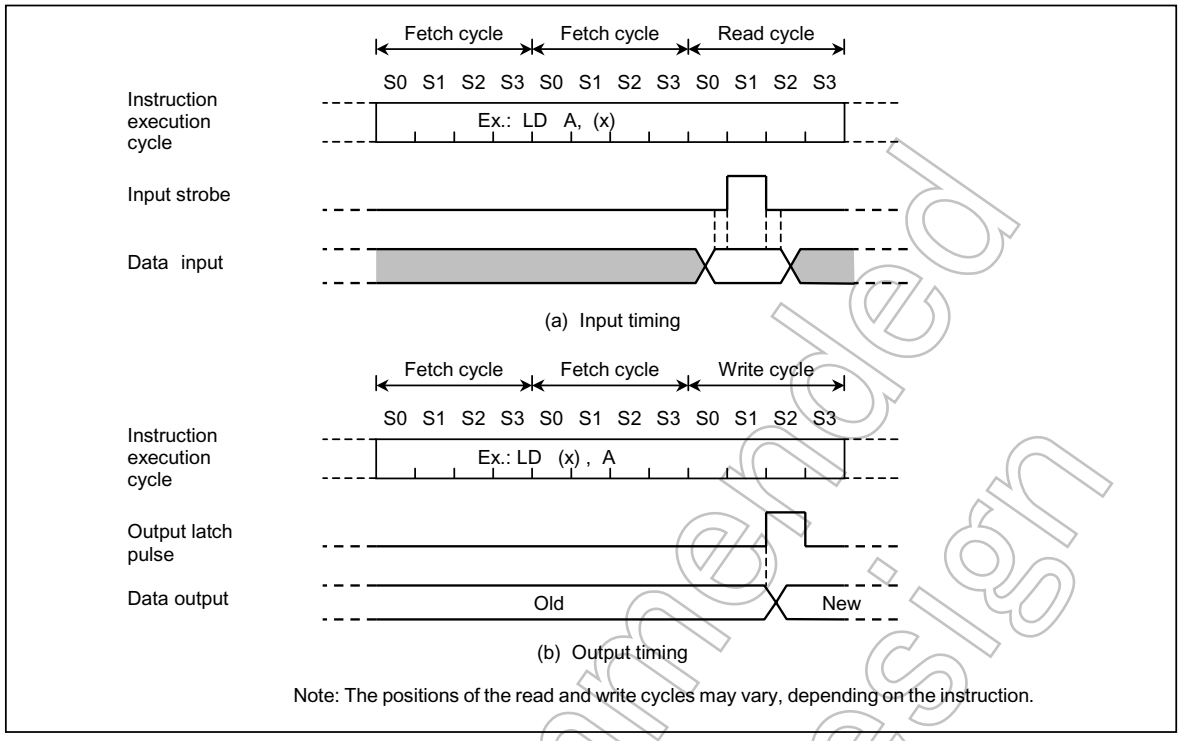


Figure 2.2.1 Input/Output Timing (Example)

When reading an I/O port except programmable I/O ports P0 and P1, whether the pin input data or the output latch contents are read depends on the instructions, as shown below:

(a) Instructions that read the output latch contents

1. XCH r, (src)
2. CLR/SET/CPL (src).b
3. CLR/SET/CPL (pp).g
4. LD (src).b, CF
5. LD (pp).b, CF
6. XCH CF, (src), b
7. ADD/ADDC/SUB/SUBB/AND/OR/XOR (src), n
8. (src) side of ADD/ADDC/SUB/SUBB/AND/OR/XOR (src), (HL)
9. MXOR (src), m

(b) Instructions that read the pin input data

1. (HL) side of ADD/ADDC/SUB/SUBB/AND/OR/XOR (src), (HL)

2.2.1 Port P0 (P07 to P00)

Port P0 is an 8-bit general-purpose input/output port which can be configured as either an input or an output in one-bit unit under software control. Input/output mode is specified by the corresponding bit in the port P0 input/output control register (P0CR). Port P0 is configured as an input if its corresponding P0CR bit is cleared to "0", and as an output if its corresponding P0CR bit is set to "1".

Port P0 is also used as Serial interfase input/output.

When used as a function pins should be set to the input mode. The pin should be set to the output mode and beforehand the output latch should be set to "1".

Note: Input mode port reads the state of input pin. When input/output mode is used to mixed, the contents of output latch setting to the input mode may be overwritten by executing bit manipulation instructions.

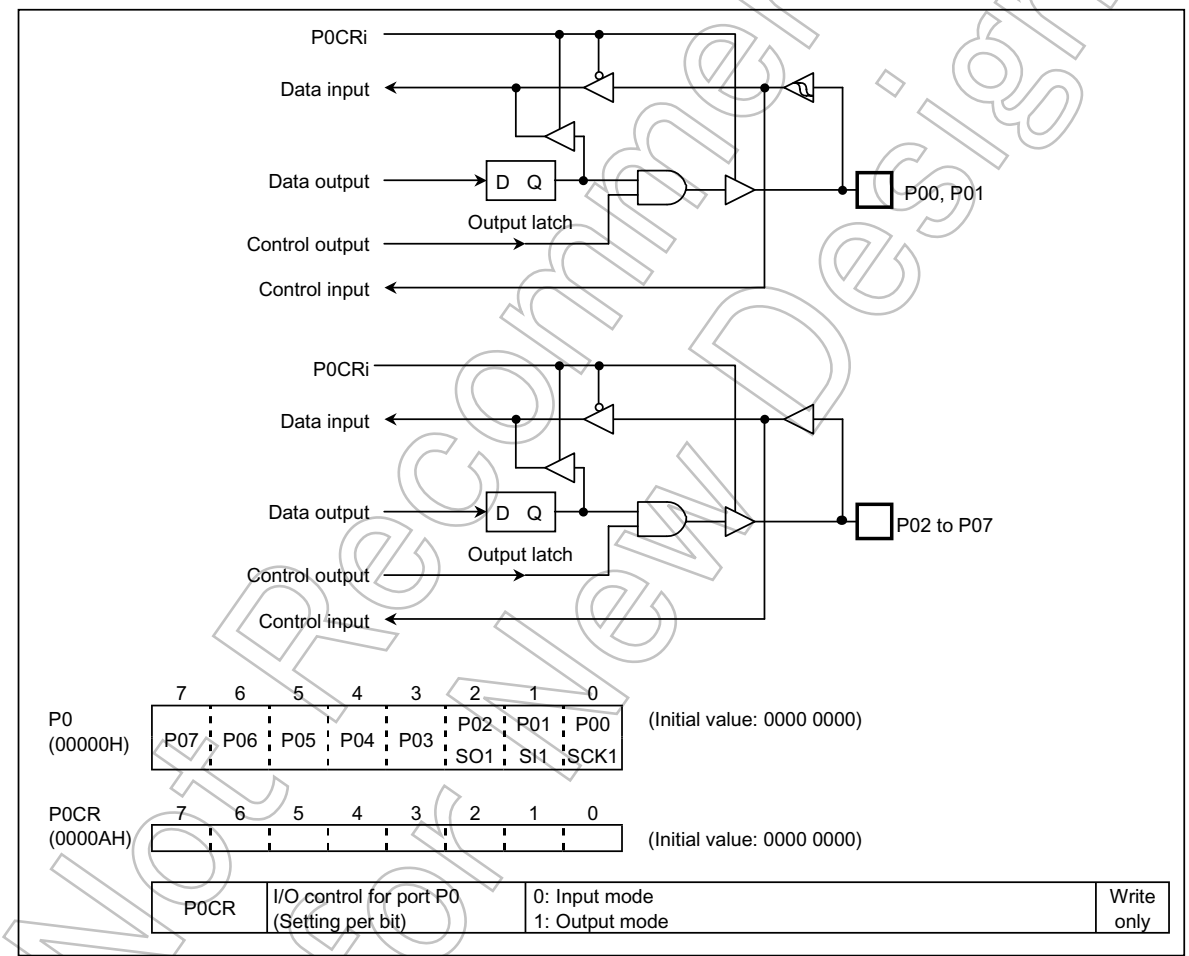


Figure 2.2.2 Port P0 and P0CR

Example: Setting the upper 4 bits of port P0 as an input port and the lower 4 bits as an output port (Initial output data are 1010B).

```
LD      (P0), 00001010B      ;  Sets initial data to P0 output latches
LD      (P0CR), 00001111B    ;  Sets the port P0 input/output mode
```

2.2.2 Port P1 (P17 to P10)

Port P1 is an 8-bit input/output port which can be configured as an input or an output in one-bit unit under software control. Input/output mode is specified by the corresponding bit in the port P1 input/output control register (P1CR). Port P1 is configured as an input if its corresponding P1CR bit is cleared to "0", and as an output if its corresponding P1CR bit is set to "1". During reset, P1CR is initialized to "0", which configures port P1 as an input . The P1 output latches are also initialized to "0".

Data is written into the output latch regardless of P1CR contents. Therefore initial output data should be written into the output latch before setting P1CR. Port P1 is also used as an external interrupt input, a timer/counter input, and a divider output. When used as a secondary function pin, the input pins should be set to the input mode, and the output pins should be set to the output mode and beforehand the output latch should be set to "1".

It is recommended that pins P11 and P15 and P16 and P17 should be used as external interrupt inputs, timer/counter input, or input ports. The interrupt latch is set on the rising or falling edge of the output when used as output ports.

Pin P10 ($\overline{INT0}$) can be configured as either an I/O port or an external interrupt input with INT0EN (bit 6 in EINTCR). During reset, the pin P10 ($\overline{INT0}$) is configured as an input port P10.

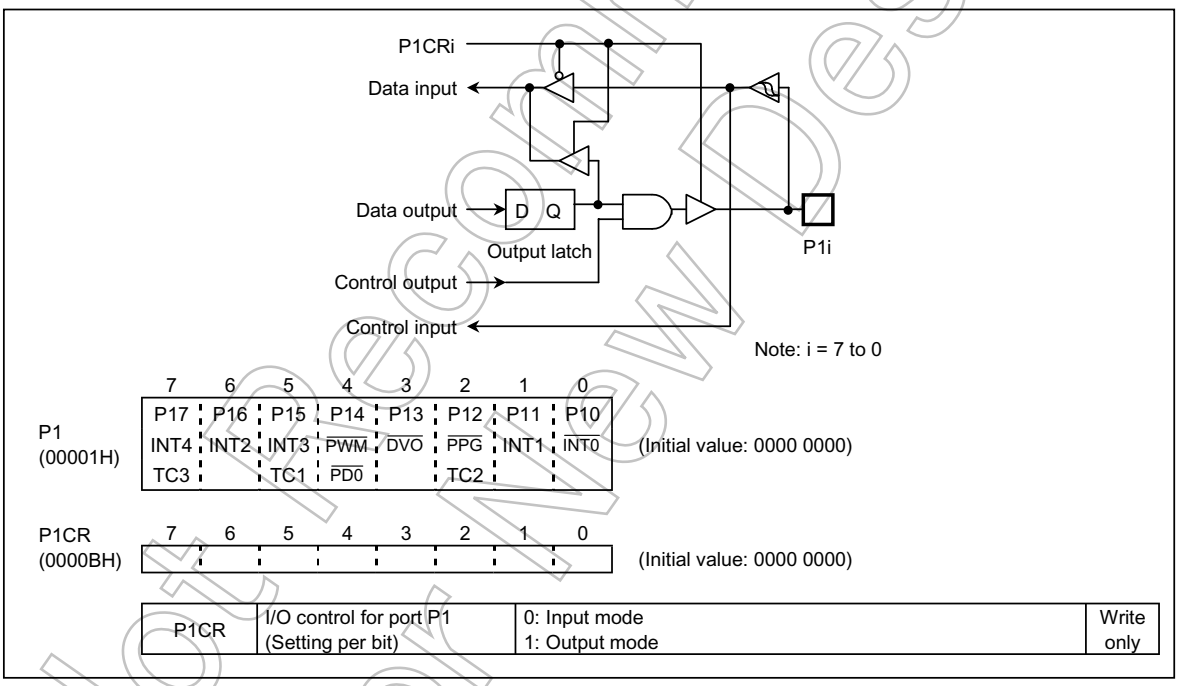


Figure 2.2.3 Port P1 and P1CR

Example: Sets P17, P16 and P14 as output ports, P13 and P11 as input ports, and the others as function pins. Internal output data is "1" for the P17 and P14 pins, and "0" for the P16 pin.

```
LD      (EINTCR), 01000000B     ;   INT0EN ← 1
LD      (P1), 10111111B         ;   P17 ← 1, P14 ← 1, P16 ← 0
LD      (P1CR), 11010000B
```

Note: Input mode port reads the state of input pin. When input/output mode is used to mixed, the contents of output latch setting to the input mode may be overwritten by executing bit manipulation instructions.

2.2.3 Port P2 (P22 to P20)

Port P2 is a 3-bit input/output port. It is also used as an external interrupt input, and low-frequency crystal connection pins When used as an input port, or the secondary function pin, the output latch should be set to "1". During reset, the output latches are initialized to "1".

A low-frequency crystal (32.768 kHz) is connected to pins P21 (XTIN) and P22 (XTOUT) in the dual-clock mode. In the single-clock mode, pins P21 and P22 can be used as normal input/output ports.

It is recommended that the P20 pin should be used as an external interrupt input, a STOP mode release signal input, or an input port. If used as an output port, the interrupt latch is set on the falling edge of the output pulse.

When a read instruction for port P2 is executed, bits 7 to 3 in P2 read in as undefined data.

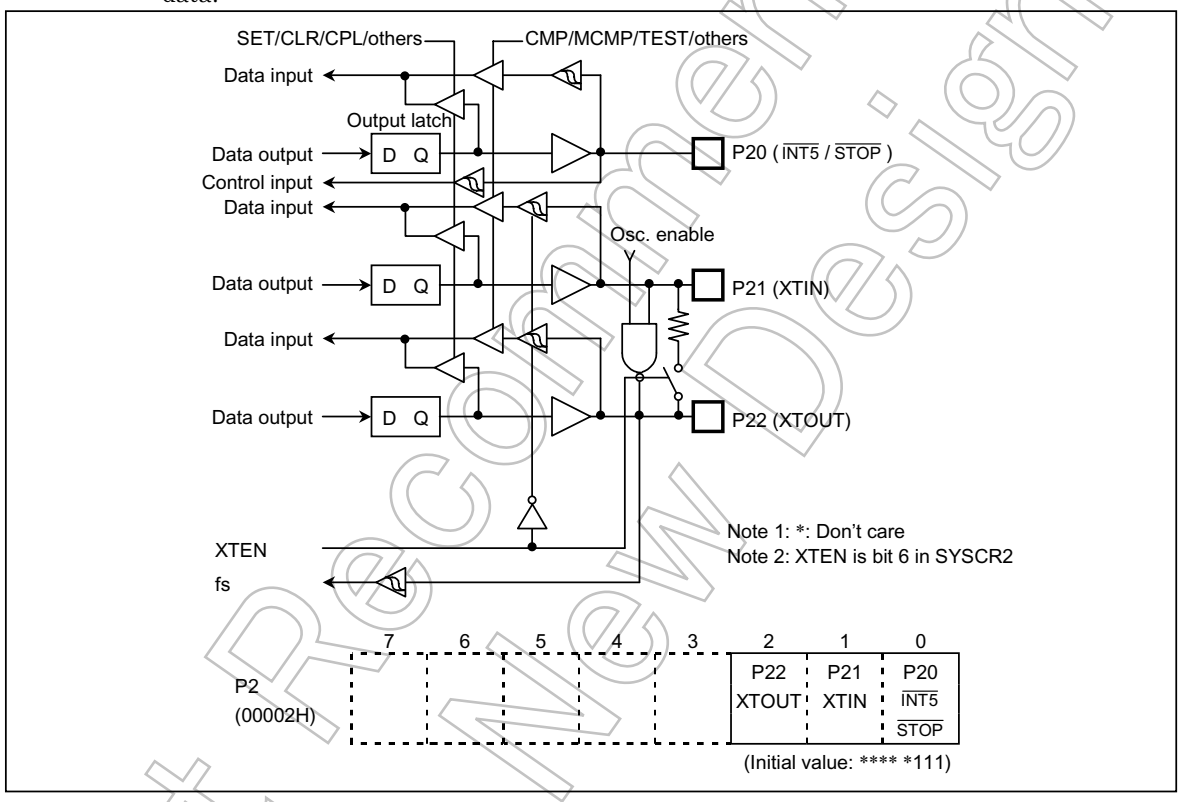


Figure 2.2.4 Port P2

2.2.4 Port P3 (P32 to P30)

Port P3 is an 3-bit input/output port and serial interface (SIO1) input/output. Input/output mode is specified by the port P3 input/output control register (P3CR).

During reset P3CR is initialized to "0" and Port P3 is input mode. The port P3 output latches are initialized to "0".

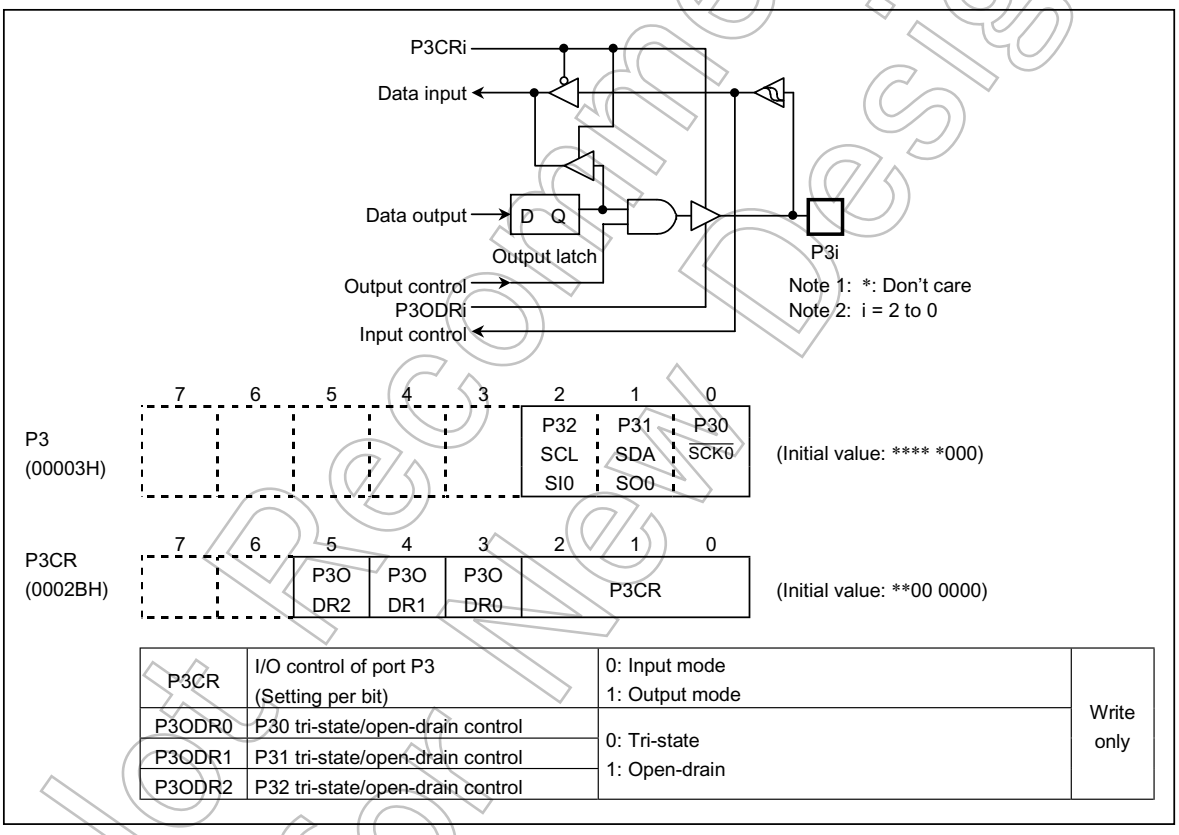
P3 is serial bus interface input/output.

When used as function pins set to output mode by Port P3 I/O control register, and I/O is controlling by output data.

The output buffer can be change to the tri-state or shink-open drain by Port P3 I/O control register (P3CR).

When a read instruction for port P3 is executed, bits 7 to 3 in P3 read is an undefined data.

Note: Input mode port reads the state of input pin. When input/output mode is used to mixed, the contents of output latch setting to the input mode may be overwritten by executing bit manipulation instructions.



P3CR	I/O control of port P3 (Setting per bit)	0: Input mode 1: Output mode	Write only
P3ODR0	P30 tri-state/open-drain control	0: Tri-state 1: Open-drain	
P3ODR1	P31 tri-state/open-drain control		
P3ODR2	P32 tri-state/open-drain control		

Figure 2.2.5 P3 and Port P3 I/O Control Register

2.2.5 Port P4 (P47 to P40)

Ports P4 is an 8-bit input/output port which can be configured as an input or an output in one-bit unit under software control. Input/output mode is specified by the corresponding bit in the port P4 input/output control register (P4CR).

At reset, P4CR is set to 0 and AINDS is cleared to 0. Thus, P4 becomes an analog input port. At the same time, the output latch of port P4 is initialized to 0. P4CR is a write-only register. Pins not used for analog input can be used as I/O ports. But do not execute the output instruction to keep the accuracy in AD conversion. Executing an input instruction on port P4 when the AD converter is in use reads 0 at pins set for analog input: 1 or 0 at pins not set for analog input, depending on the pin input level.

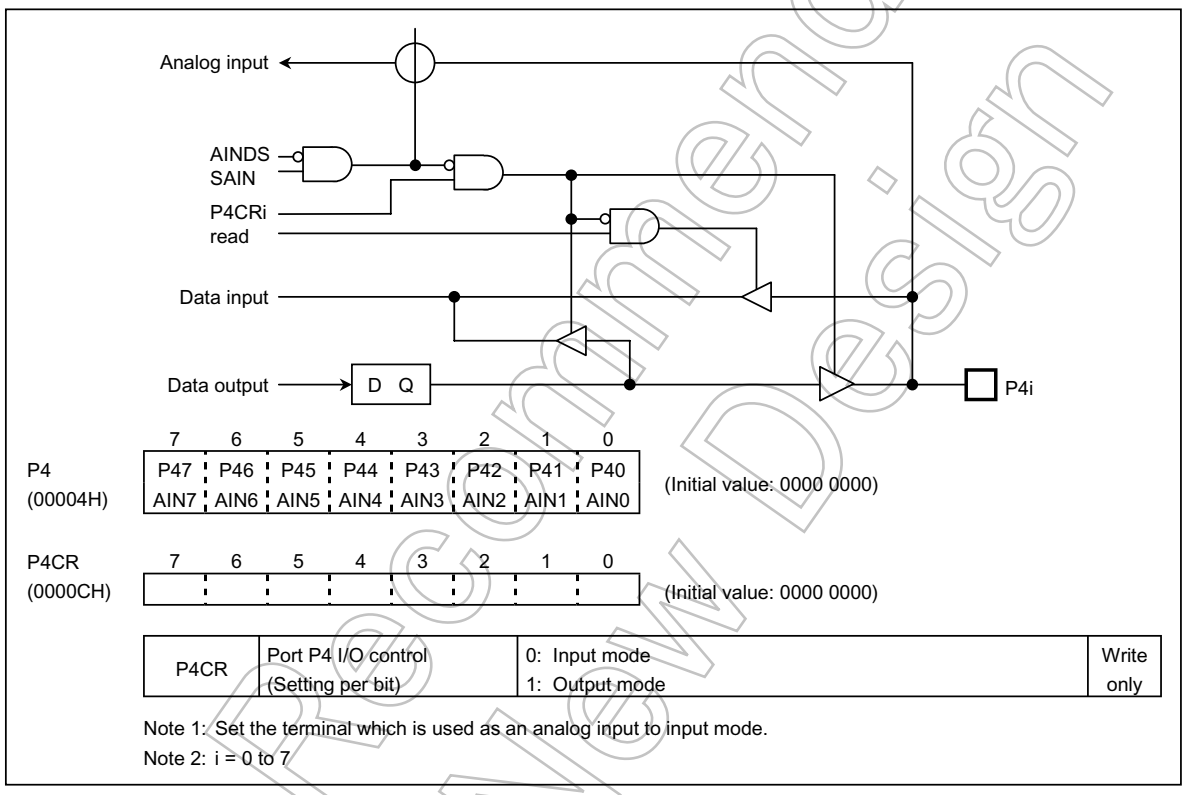


Figure 2.2.6 P4 and Port P4 Control Register

2.2.6 Port P5 (P53 to P50)

Ports P5 is an 8-bit input/output port which can be configured as an input or an output in one-bit unit under software control. Input/Output mode is specified by the corresponding bit in the port P5 input/output control register (P5CR).

At reset, P5CR is set to 0 and AINDS is cleared to 0. Thus, P5 becomes an analog input port . At the same time, the output latch of port P5 is initialized to 0. P5CR is a write-only register. Pins not used for analog input can be used as I/O ports. But do not execute the output instruction to keep the accuracy in AD conversion. Executing an input instruction on port P5 when the AD converter is in use reads 0 at pins set for analog input: 1 or 0 at pins not set for analog input, depending on the pin input level.

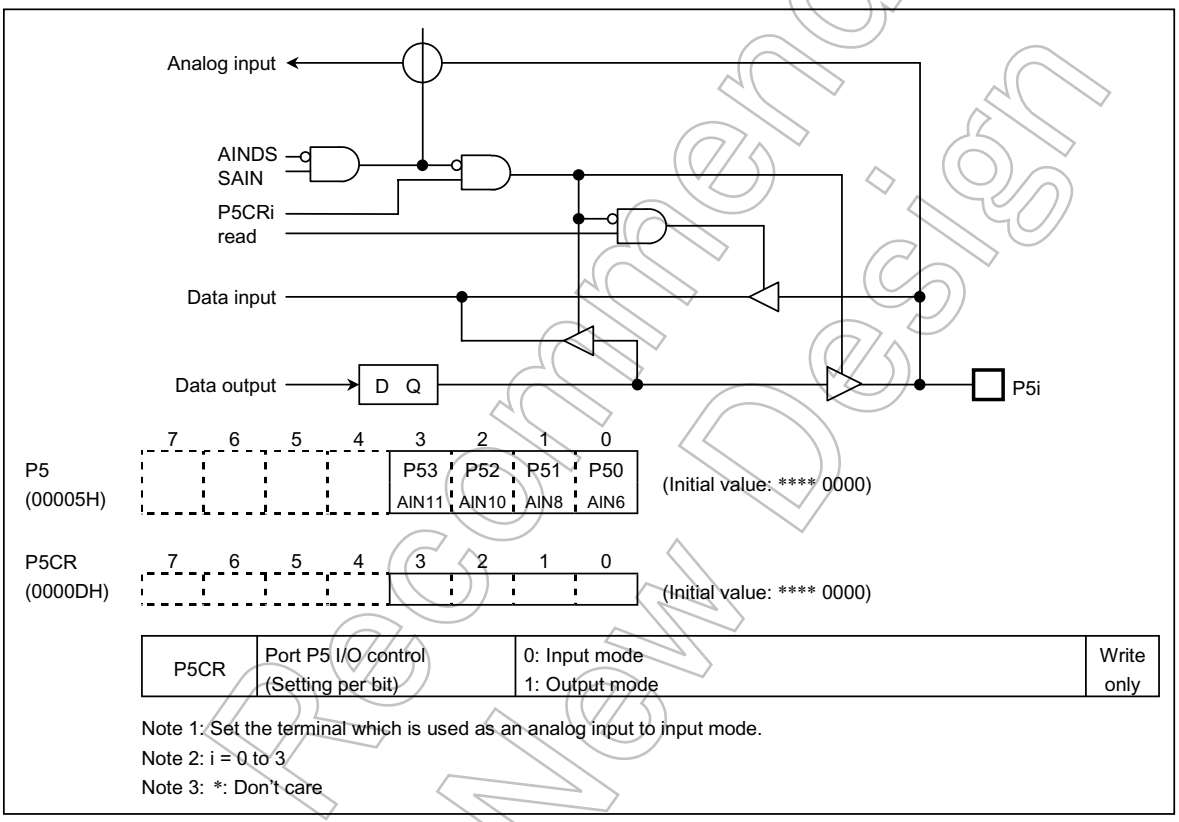


Figure 2.2.7 P5 and Port P5 I/O Control Register

2.2.7 Port 6 (P67 to P60)

Port 6 is an 8-bit high-breakdown voltage input/output port, and also used as a VFT driver output, which can directly drive vacuum fluorescent tube (VFT).

When used as an VFT driver output, the output latch should be cleared to "0".

Pins not used for VFT driver output can be used as I/O ports.

When use an VFT driver and normal input/output at the same time, VFT driver output data buffer memory (DBF) need to cleared to "0".

The output latches are initialized to "0" during reset.

It recommends that port 6 shoud be used to drive directly drive vacuum fluorescent tube (VFT), since this port has a pull down resistance.

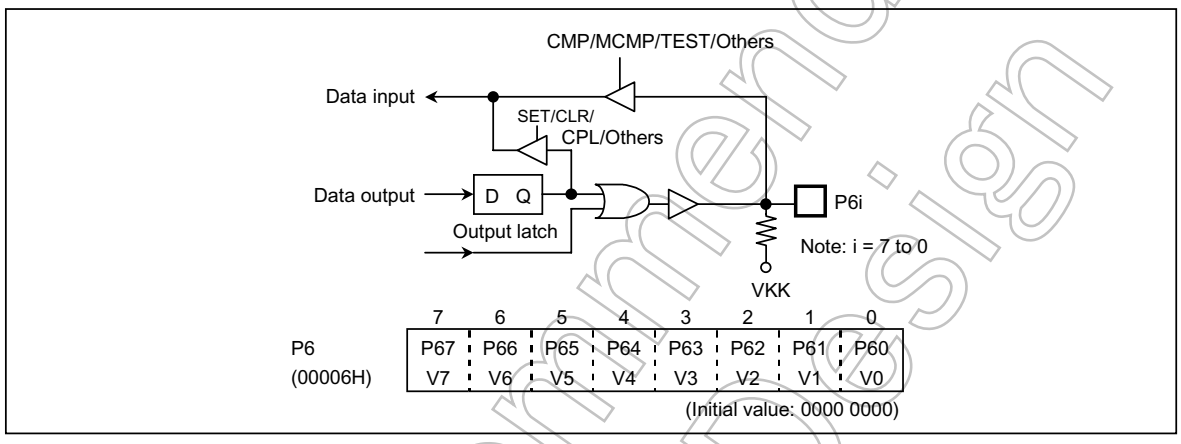


Figure 2.2.8 Port P6

2.2.8 Port 7 (P77 to P70)

Port 7 is an 8-bit high-breakdown voltage input/output port, and also used as a VFT driver output, which can directly drive vacuum fluorescent tube (VFT).

When used as an VFT driver output, the output latch should be cleared to "0".

Pins not used for VFT driver output can be used as I/O ports.

When use an VFT driver and normal input/output at the same time, VFT driver output data buffer memory (DBF) need to cleared to "0".

The output latches are initialized to "0" during reset.

It recommends that port 7 shoud be used to drive directly drive vacuum fluorescent tube (VFT), since this port has a pull down resistance.

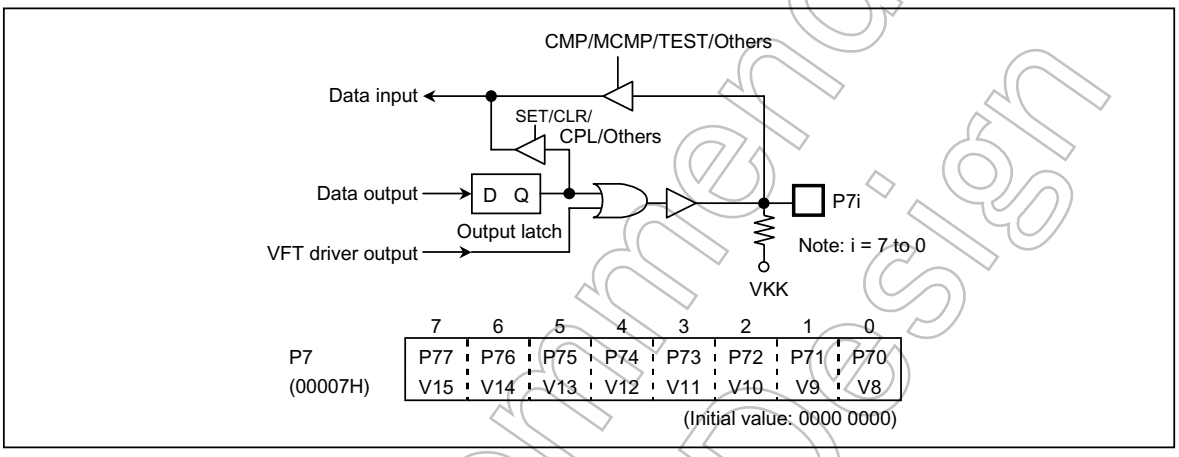


Figure 2.2.9 Port P7

2.2.9 Port 8 (P87 to P80)

Port 8 is an 8-bit high-breakdown voltage input/output port, and also used as a VFT driver output, which can directly drive vacuum fluorescent tube (VFT).

When used as an VFT driver output, the output latch should be cleared to "0".

Pins not used for VFT driver output can be used as I/O ports.

When use an VFT driver and normal input/output at the same time, VFT driver output data buffer memory (DBF) need to cleared to "0".

The output latches are initialized to "0" during reset.

It recommends that port 8 shoud be used to drive directly drive vacuum fluorescent tube (VFT), since this port has a pull down resistance.

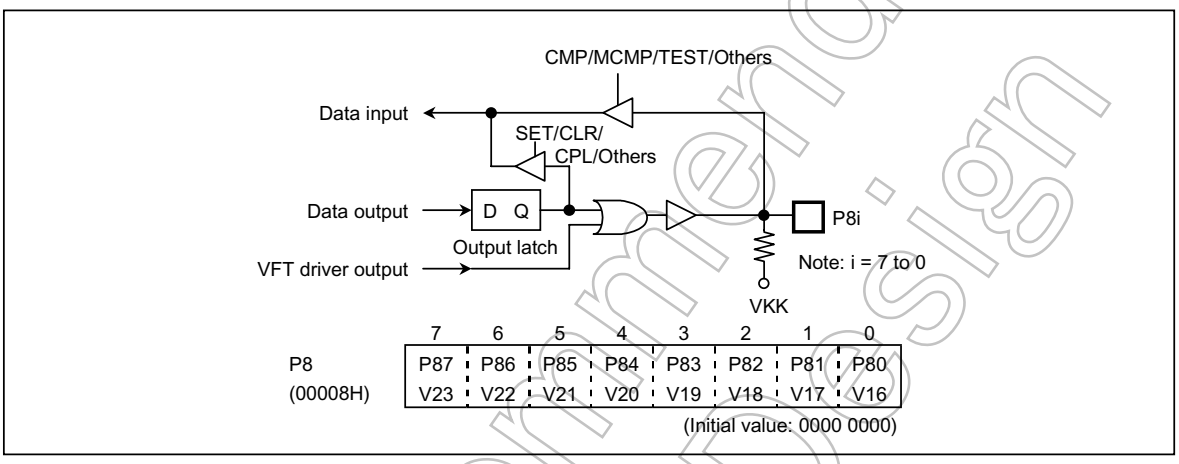


Figure 2.2.10 Port P8

2.2.10 Port 9 (P97 to P90)

Port 9 is an 8-bit high-breakdown voltage input/output port, and also used as a VFT driver output, which can directly drive vacuum fluorescent tube (VFT).

When used as an VFT driver output, the output latch should be cleared to "0".

Pins not used for VFT driver output can be used as I/O ports.

When use an VFT driver and normal input/output at the same time, VFT driver output data buffer memory (DBF) need to cleared to "0".

The output latches are initialized to "0" during reset.

It recommends that port 9 shoud be used to drive directly drive vacuum fluorescent tube (VFT), since this port has a pull down resistance.

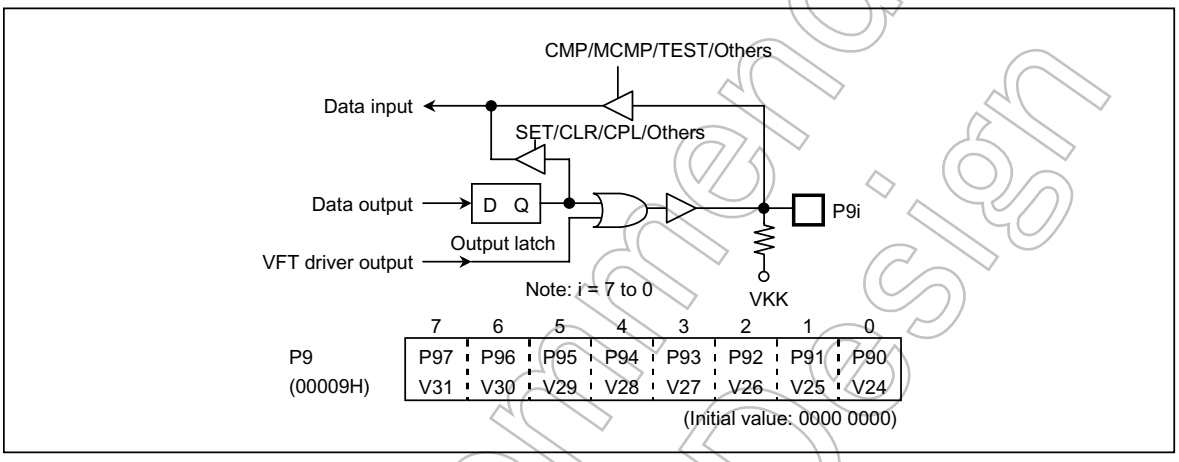


Figure 2.2.11 Port P9

2.2.11 PD (PD4 to PD0)

Port PD is high-breakdown voltage input/output port, and also used as a VFT driver output, which can directly drive vacuum fluorescent tube (VFT).

General-purpose or segment can be selected for each bit by VSEL (bit 4 to 0) in VFT driver control register 1 (VFTCR1). The VSEL is cleared to "0" during reset, which used as an input mode. When used as an input port or VFT driver output, the output latch set to "0". The output latches are initialized to "0" during reset.

When a read instruction for port PD is executed bit 7 to 5 in PD read in as undefined data.

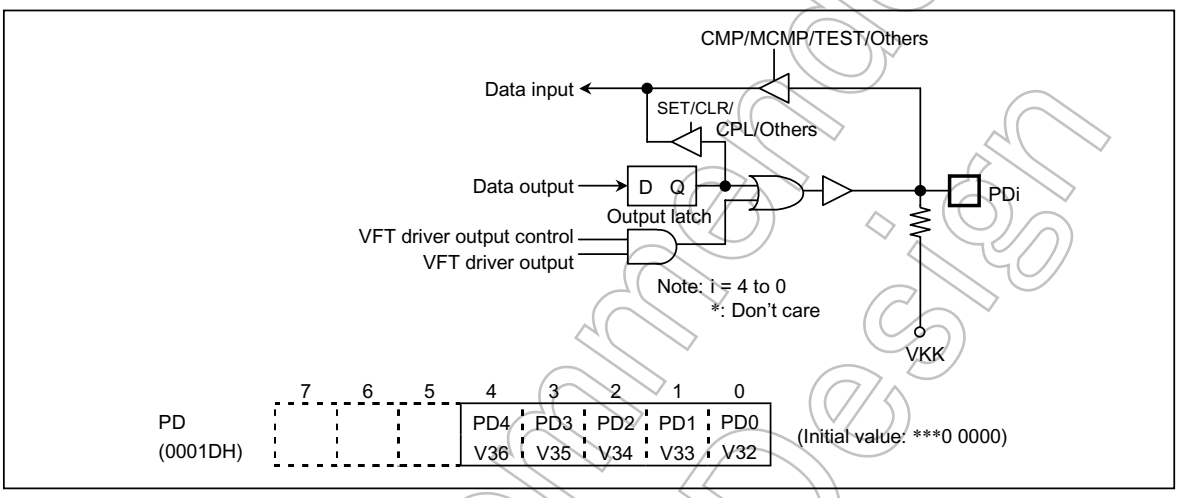


Figure 2.2.12 PD Port

2.3 Time Base Timer (TBT)

The time base timer generates time base for key scanning, dynamic displaying, etc. It also provides a time base timer interrupt (INTTBT).

An INTTBT is generated on the first rising edge of source clock (the divider output of the timing generator) after the time base timer has been enabled. The divider is not cleared by the program; therefore, only the first interrupt may be generated ahead of the set interrupt period (Figure 2.3.1 (b)).

The interrupt frequency (TBTCK) must be selected with the time base timer disabled (the interrupt frequency must not be changed with the disable from the enable state). Both frequency selection and enabling can be performed simultaneously.

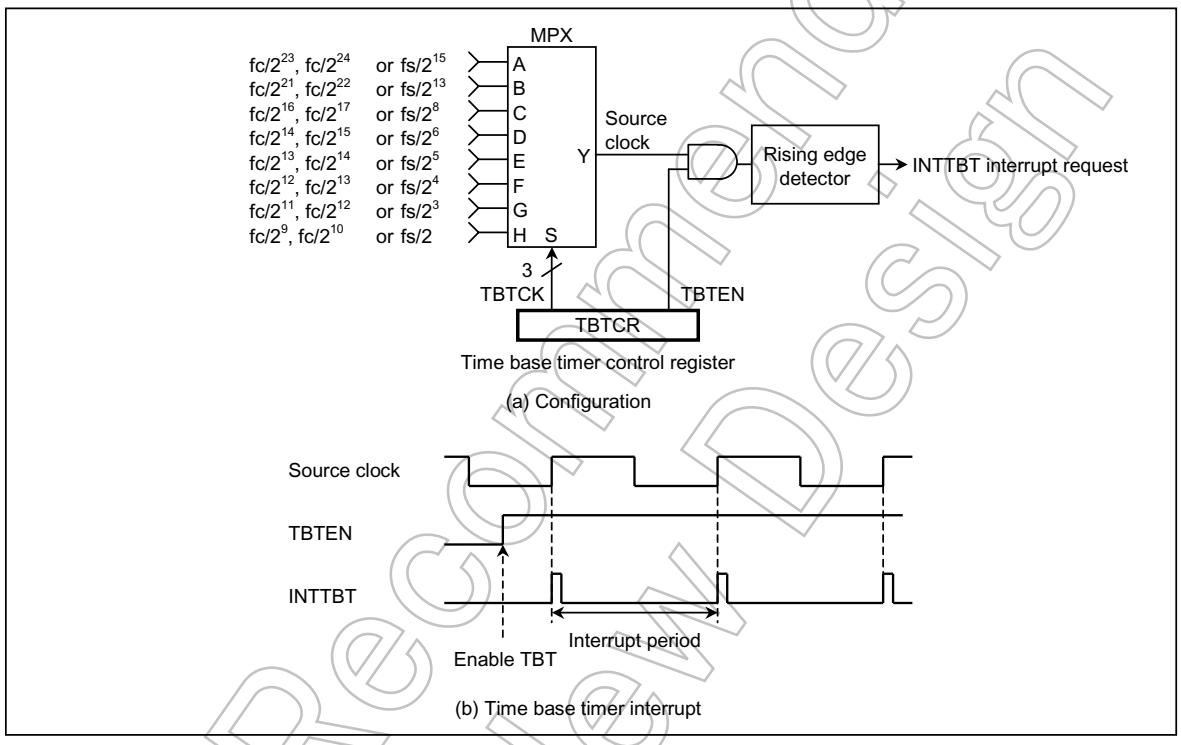


Figure 2.3.1 Time Base Timer

Example: Sets the time base timer frequency to fc/2^{16} [Hz] and enables an INTTBT interrupt.

```
LD      (TBTCR), 00001010B
SET     (EIRL). 6
```

TBTCR (00036H)

7	6	5	4	3	2	1	0
(DVOEN)	(DVOCK)		(DV7CK)	TBTEN	TBTCK		

(Initial value: 0**0 0***)

TBTEN	Time base timer enable/disable	0: Disable 1: Enable						R/W
TBTCK	Time base timer interrupt frequency select [Hz]		NORMAL1/2, IDLE1/2 mode				SLOW, SLEEP mode	
			DV7CK = 0		DV7CK = 1			
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1		
		000	$fc/2^{23}$	$fc/2^{24}$	$fs/2^{15}$	$fs/2^{15}$	$fs/2^{15}$	
		001	$fc/2^{21}$	$fc/2^{22}$	$fs/2^{13}$	$fs/2^{13}$	$fs/2^{13}$	
		010	$fc/2^{16}$	$fc/2^{17}$	$fs/2^{8}$	$fs/2^{8}$	—	
		011	$fc/2^{14}$	$fc/2^{15}$	$fs/2^{6}$	$fs/2^{6}$	—	
		100	$fc/2^{13}$	$fc/2^{14}$	$fs/2^{5}$	$fs/2^{5}$	—	
		101	$fc/2^{12}$	$fc/2^{13}$	$fs/2^{4}$	$fs/2^{4}$	—	
		110	$fc/2^{11}$	$fc/2^{12}$	$fs/2^{3}$	$fs/2^{3}$	—	
		111	$fc/2^{9}$	$fc/2^{10}$	fs/2	fs/2	—	

Note: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz], *: Don't care

Figure 2.3.2 Time Base Timer Control Register

Table 2.3.1 Time Base Timer Interrupt Frequency (Example: fc = 12.5 MHz, fs = 32.8 kHz)

TBTCK	Time Base Timer Interrupt Frequency [Hz]				
	NORMAL1/2, IDLE1/2 Mode				SLOW, SLEEP Mode
	DV7CK = 0		DV7CK = 1		
	DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1	
000	1.49	0.75	1	1	1
001	5.96	2.98	4	4	4
010	190.73	95.37	128	128	—
011	762.94	381.47	512	512	—
100	1525.88	762.94	1024	1024	—
101	3051.76	1525.88	2048	2048	—
110	6103.52	3051.76	4096	4096	—
111	24414.06	12207.03	16384	16384	—

2.4 Watchdog Timer (WDT)

The watchdog timer is a fail-safe system to rapidly detect the CPU malfunctions such as endless looping caused by noise or the like, or deadlock and resume the CPU to the normal state.

The watchdog timer signal for detecting malfunction can be selected either a reset output or a pseudo non-maskable interrupt request. However, selection is possible only once after reset. At first the reset output is selected.

When the watchdog timer is not being used for malfunction detection, it can be used as a timer to generate an interrupt at fixed intervals.

2.4.1 Watchdog Timer Configuration

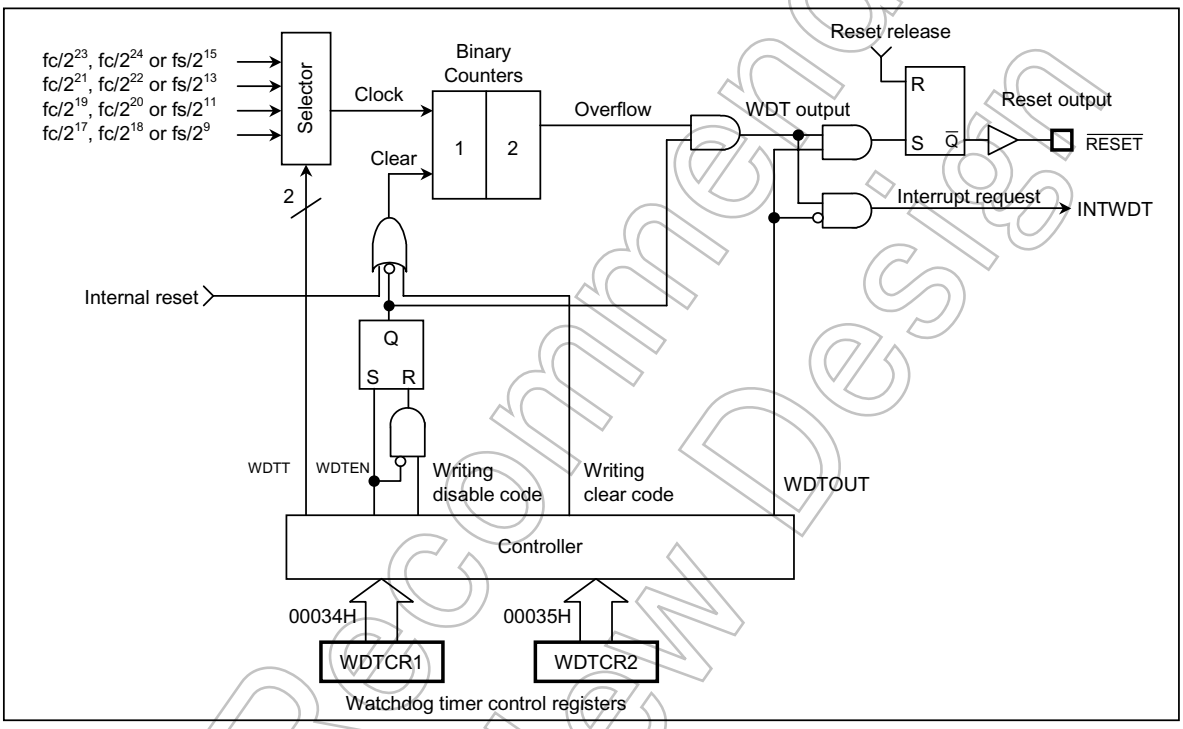


Figure 2.4.1 Watchdog Timer Configuration

2.4.2 Watchdog Timer Control

Figure 2.4.2 shows the watchdog timer control registers (WDTCR1, WDTCR2). The watchdog timer is automatically enabled after reset.

(1) Malfunction detection methods using the watchdog timer

The CPU malfunction is detected at follows.

1. Setting the detection time, selecting output, and clearing the binary counter.
2. Repeatedly clearing the binary counter within the setting detection time

If the CPU malfunctions such as endless looping or deadlock occur for any cause, the watchdog timer output will become active at the rising of an overflow from the binary counters unless the binary counters are cleared. At this time, when WDTOUT = 1 a reset is generated, which drives the $\overline{\text{RESET}}$ pin low to reset the internal hardware and the external circuit. When WDTOUT = 0, a watchdog timer interrupt(INTWDT) is generated.

The watchdog timer temporarily stops counting in STOP mode including warm-up or IDLE mode, and automatically restarts (continues counting) when the STOP/IDLE mode is released.

Note: Just right before disabling the watchdog timer, disable the acceptance of interrupts (DI) and clear the watchdog timer.
If the watchdog timer is disabled under conditions other than the above, the proper operation cannnot be guaranteed.

Example:

```
DI                          ;  Disable interrupt acceptance.
LD    (WDTCR2), 4EH         ;  Clears the watchdog timer.
LDW   (WDTCR1), B100H       ;  Disables the watchdog timer.
EI                          ;  Enables interrupt acceptance.
```

Example: Sets the watchdog timer detection time to 2^{21}/fc [s] and resets the CPU malfunction.

```
LD    (WDTCR2), 4EH         ;  Clears the binary counters
LD    (WDTCR1), 00001101B   ;  WDTT ← 10,  WDTOUT ← 1
Within WDT detection time
LD    (WDTCR2), 4EH         ;  Clears the binary counters (always clear
                               immediately after changing WDTT)
Within WDT detection time
LD    (WDTCR2), 4EH         ;  Clears the binary counters
Within WDT detection time
LD    (WDTCR2), 4EH         ;  Clears the binary counters
```

Watchdog Timer Register 1

WDTCR1 (00034H)

7	6	5	4	3	2	1	0
				WDT EN	WDTT		WDT OUT

(Initial value: **** 1001)

WDTEN	Watchdog timer enable/disable	0: Disable (It is necessary to write the disable code to WDTCR2) 1: Enable						Write only
WDTT	Watchdog timer detection time [s]		NORMA1/2 mode				SLOW mode	
			DV7CK = 0		DV7CK = 1			
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1		
		00	2^{25}/fc	2^{26}/fc	2^{17}/fs	2^{17}/fs	2^{17}/fs	
		01	2^{23}/fc	2^{24}/fc	2^{15}/fs	2^{15}/fs	2^{15}/fs	
		10	2^{21}/fc	2^{22}/fc	2^{13}/fs	2^{13}/fs	2^{13}/fs	
		11	2^{19}/fc	2^{20}/fc	2^{11}/fs	2^{11}/fs	2^{11}/fs	
WDTOUT	Watchdog timer output select	0: Interrupt request 1: Reset output						

Note 1: WDTOUT cannot be set to "1" by program after clearing WDTOUT to "0".
Note 2: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz], *: Don't care
Note 3: WDTCR1 is a write-only register and must not be used with any of read-modify-write instructions.
Note 4: Disable the watchdog timer or clear the counter just before switching to STOP mode.
When the counter is cleared just before switching to STOP mode, clear the counter again subsequently to releasing STOP mode.

Watchdog Timer Register 2

WDTCR2 (00035H)

7	6	5	4	3	2	1	0

(Initial value: **** ****)

WDTCR2	Watchdog timer control code write register	4EH: Watchdog timer binary counter clear (clear code) B1H: Watchdog timer disable (disable code) others: Invalid	Write only

Note 1: The disable code is invalid unless written when WDTEN = 0.
Note 2: *: Don't care
Note 3: The binary counter of the watchdog timer must not be cleared by the interrupt task.

Figure 2.4.2 Watchdog Timer Control Registers

(2) Watchdog timer enable

The watchdog timer is enabled by setting WDTEN (bit 3 in WDTCR1) to "1". WDTEN is initialized to "1" during reset, so the watchdog timer operates immediately after reset is released.

Example : Enables watchdog timer

(3) Watchdog timer disable

The watchdog timer is disabled by writing the disable code (B1H) to WDTCR2 after clearing WDTEN (bit 3 in WDTCR1) to "0". The watchdog timer is not disabled if this procedure is reversed and the disable code is written to WDTCR2 before WDTEN is cleared to "0". During disabling the watchdog timer, the binary counters are cleared to "0".

```
Example:  Disables watchdog timer
          LDW     (WDTCR1), 0B101H      ;  WDTEN ← 0, WDTCR2 ←
                                           Disable code
```

Table 2.4.1 Watchdog Timer Detection Time (Example: fc = 12.5 MHz, fs = 32.8 kHz)

WDTT	Watchdog Timer Detection Time [s]				
	NORMAL1/2 Mode				SLOW Mode
	DV7CK = 0		DV7CK = 1		
	DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1	
00	2.684 s	5.369 s	4 s	4 s	4 s
01	671.089 ms	1.342 s	1 s	1 s	1 s
10	167.772 ms	335.544 ms	250 ms	250 ms	250 ms
11	41.943 ms	83.886 ms	62.5 ms	62.5 ms	62.5 ms

2.4.3 Watchdog Timer Interrupt (INTWDT)

This is a pseudo non-maskable interrupt which can be accepted regardless of the contents of the EIR. If a watchdog timer interrupt or a software interrupt is already accepted, however, the new watchdog timer interrupt waits until the previous interrupt processing is completed (the end of the [RETN] instruction execution).

The stack pointer (SP) should be initialized before using the watchdog timer output as an interrupt source with WDTOUT.

```
Example:  Watchdog timer interrupt setting up
          LD      SP, 0023FH              ;  Sets the stack pointer
          LD      (WDTCR1), 00001000B     ;  WDTOUT ← 0
```

2.4.4 Watchdog Timer Reset

If the watchdog timer output becomes active, a reset is generated, which drives the $\overline{\text{RESET}}$ pin (sink open drain input/output with pull-up) low to reset the internal hardware and external circuits. The reset output time is about 8/fc to 24/fc [s] (0.64 to 1.92 μs at fc = 12.5 MHz, fcgck = fc).

Note: The high-frequency clock oscillator also turns on when a watchdog timer reset is generated in SLOW mode. The reset output time is 8/fc to 24/fc [s]. Therefore, the reset time may include a certain amount of error if there is any fluctuation of the oscillation frequency at starting the high-frequency clock oscillation. Thus, the reset time must be considered an approximated value.

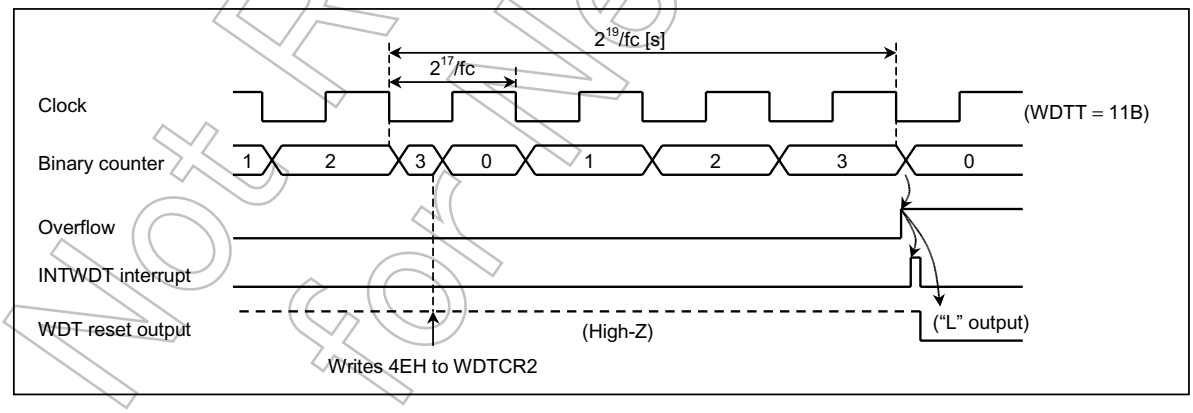


Figure 2.4.3 Watchdog Timer Interrupt/Reset

2.5 Divider Output ($\overline{DVO}$)

Approximately 50% duty pulse can be output using the divider output circuit, which is useful for piezoelectric buzzer drive. Divider output is from pin P13 ($\overline{DVO}$). The P13 output latch should be set to "1" and then the P13 should be configured as an output mode.

TBTCR (00036H)

7	6	5	4	3	2	1	0
DVOEN	DVOCK		(DV7CK)	(TBTEN)		(TBTCK)	

(Initial value: 0**0 0***)

				NORMAL1/2, IDLE1/2 mode				SLOW, SLEEP mode	R/W
				DV7CK = 0		DV7CK = 1			
				DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1		
DVOEN	Divider output enable/disable	0: Disable 1: Enable							
DVOCK	Divider output ($\overline{DVO}$) frequency selection [Hz]	00	$fc/2^{13}$	$fc/2^{14}$	$fs/2^{5}$	$fs/2^{5}$	$fs/2^{5}$		
		01	$fc/2^{12}$	$fc/2^{13}$	$fs/2^{4}$	$fs/2^{4}$	$fs/2^{4}$		
		10	$fc/2^{11}$	$fc/2^{12}$	$fs/2^{3}$	$fs/2^{3}$	$fs/2^{3}$		
		11	$fc/2^{10}$	$fc/2^{11}$	$fs/2^{2}$	$fs/2^{2}$	$fs/2^{2}$		

Note: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz], *: Don't care

Figure 2.5.1 Divider Output Control Register

```
Example:  1.5 kHz pulse output (at fc = 12.5 MHz, DV1CK = 0)
          SET     (P1).3              ;  P13 output latch ← 1
          LD      (P1CR), 00001000B   ;  Configures P13 as an output mode
          LD      (TBTCR), 10000000B  ;  DVOEN ← 1, DVOCK ← 00
```

Table 2.5.1 Divider Output Frequency (Example: at fc = 12.5 MHz, fs = 32.8 kHz)

DVOCK	Divider Output Frequency [kHz]				
	NORMAL1/2, IDLE1/2 MODE				SLOW, SLEEP Mode
	DV7CK = 0		DV7CK = 1		
	DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1	
00	1.526 k	0.763 k	1.024 k	1.024 k	1.024 k
01	3.502	1.526	2.048	2.048	2.048
10	6.104	3.502	4.096	4.096	4.096
11	12.207	6.104	8.192	8.192	8.192

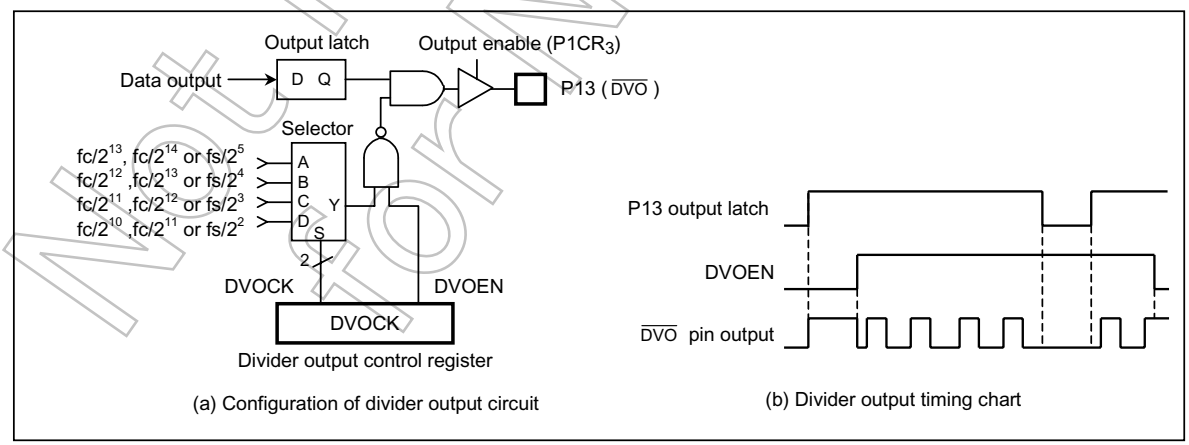


Figure 2.5.2 Divider Output

2.6 16-Bit Timer/Counter 1 (TC1)

2.6.1 Configuration

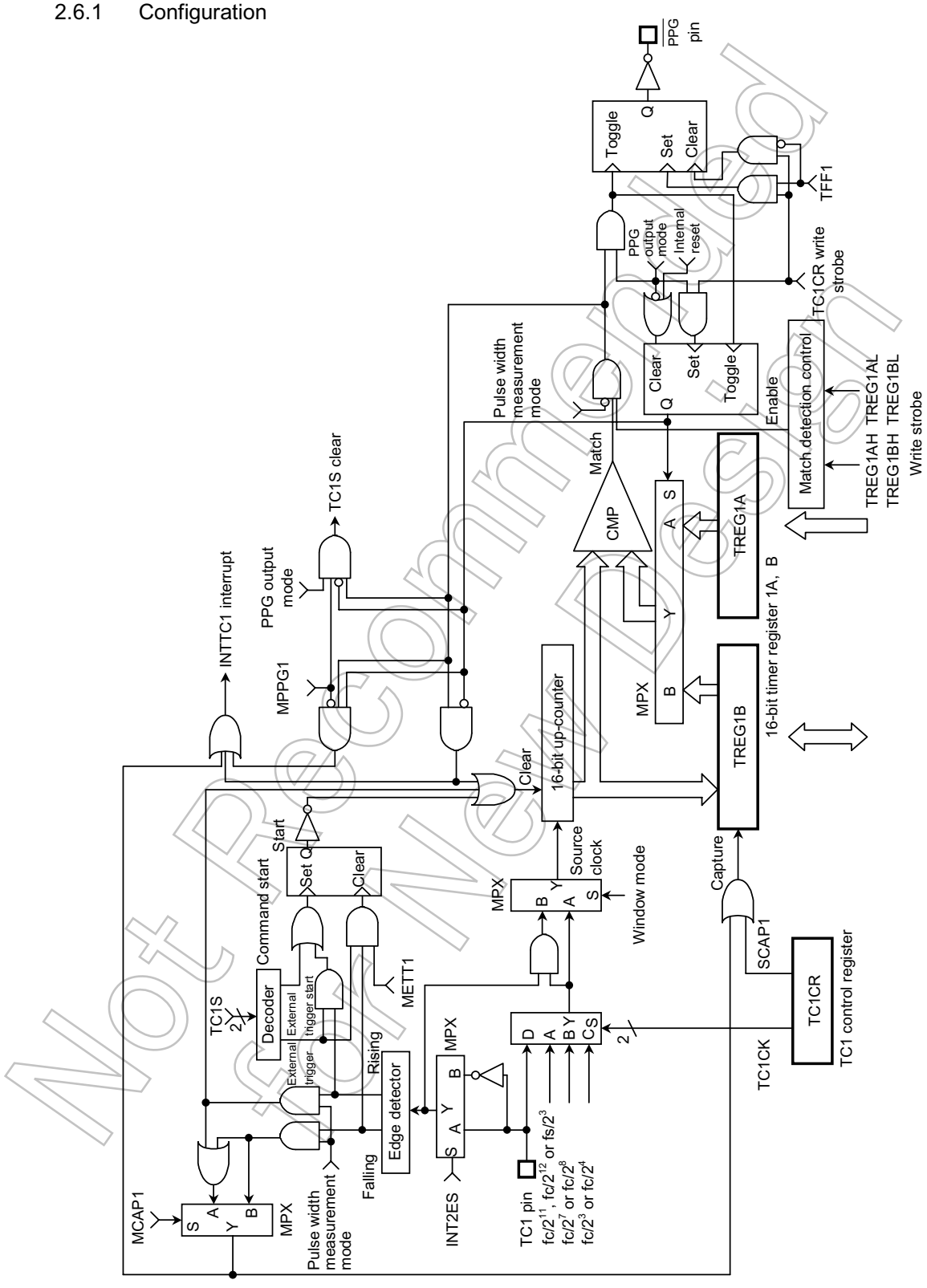


Figure 2.6.1 Timer/Counter 1 (TC1)

2.6.2 Control

The timer/counter 1 is controlled by a timer/counter 1 control register (TC1CR) and two 16-bit timer registers (TREG1A and TREG1B). Reset does not affect TREG1A and TREG1B.

TREG1A (00010, 00011H)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TREG1AH (00011H)								TREG1AL (00010H)							

Write only

TREG1B (00012, 00013H)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TREG1BH (00013H)								TREG1BL (00012H)							

Read/Write (Writing is capable only when PPG output mode)

TC1CR (00014H)

7	6	5	4	3	2	1	0
TFF1	SCAP1 MCAP1 METT1 MPPG1	TC1S		TC1CK		TC1M	

(Initial value: 0000 0000)

TC1M	TC1 operating mode select	00: Timer/external trigger timer/event counter mode 01: Window mode 10: Pulse width measurement mode 11: PPG (Programmable pulse generate) output mode					Write only
TC1CK	TC1 source clock select [Hz]		NORMAL1/2, IDLE1/2 mode			SLOW, SLEEP mode	
			DV7CK = 0		DV7CK = 1		
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1	
		00	$fc/2^{11}$	$fc/2^{12}$	$fs/2^3$	$fs/2^3$	$fs/2^3$
		01	$fc/2^7$	$fc/2^8$	$fc/2^7$	$fc/2^8$	—
		10	$fc/2^3$	$fc/2^4$	$fc/2^3$	$fc/2^4$	—
		11	External clock (TC1 pin input)				
TC1S	TC1 start control	00: Stop and counter clear 01: Command start 10: Reserved 11: External trigger start					
SCAP1	Software capture control	0: —			1: Software capture trigger		
MCAP1	Pulse width measurement mode control	0: Double edge capture			1: Single edge capture		
METT1	External trigger timer mode control	0: Trigger start			1: Trigger start & stop		
MPPG1	PPG output control	0: Pulse			1: Single		
TFF1	Time F/F1 control for PPG output mode	0: Clear			1: Set		

Note 1: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz]

Note 2: Writing to the lower byte of the timer registers (TREG1AL, TREG1BL), the comparison is inhibited until the upper byte(TREG1AH, TREG1BH) is written. Only the lower byte of the timer registers can not be changed. After writing to the upper byte, any match during 1 machine cycle (instruction execution cycle) is ignored.

Note 3: Set the mode, source clock, edge (including INT2ES), PPG control and timer F/F control when TC1 stops (TC1S = 00).

Note 4: Software capture can be used in only timer and event counter modes. SCAP1 is automatically cleared to “0” after capturing.

Note 5: Values to be loaded to timer registers must satisfy the following condition.
TREG1A>TREG1B>0(PPG output mode), TREG1A>0 (others)

Note 6: Always write “0” to TFF1 except PPG output mode.

Note 7: TC1CR and TREG1A are write-only registers and must not be used with any of the read-modify-write instructions such as SET, CLR, etc.

Note 8: Writing to the TREG1B is not possible unless TC1 is set to the PPG output mode.

Note 9: Please use the auto-capture function in the operative condition of TC1.
A captured value may not be fixed if it's read after the execution of the timer stop or auto-capture disable. Please read the capture value in a capture enabled condition.

Note 10: Since the up-counter value is captured into TC1DRB by the source clock of up-counter after setting TC1CR<ACAP1> to “1”. Therefore, wait at least one cycle of the internal source clock before reading TC1DRB for the first time.

Figure 2.6.2 Timer Registers and TC1 Control Register

2.6.3 Function

Timer/Counter 1 has six operating modes: timer, external trigger timer, event counter, window, pulse width measurement, programmable pulse generator output mode.

(1) Timer mode

In this mode, counting up is performed using the internal clock. The contents of TREG1A are compared with the contents of up-counter. If a match is found, an INTTC1 interrupt is generated, and the counter is cleared to "0". Counting up resumes after the counter is cleared. The current contents of up-counter can be transferred to TREG1B by setting SCAP1 (bit 6 in TC1CR) to "1"(software capture function). SCAP1 is automatically cleared after capturing.

Table 2.6.1 Source Clock (Internal clock) for Timer/Counter 1 (Example: at fc = 12.5 MHz, fs = 32.8 kHz)

TC1CK	NORMAL1/2, IDLE1/2 Mode							
	DV7CK = 0				DV7CK = 1			
	DV1CK = 0		DV1CK = 1		DV1CK = 0		DV1CK = 1	
	Resolution [µs]	Maximum Time Setting	Resolution [µs]	Maximum Time Setting	Resolution [µs]	Maximum Time Setting	Resolution [µs]	Maximum Time Setting
00	163.84 µs	10.8 s	327.68 µs	21.5 s	244.14 µs	16.0 s	244.14 µs	16.0 s
01	10.24 µs	0.64 s	20.48 µs	1.28 s	8 µs	0.5 s	16 µs	1.0 s
10	0.64 µs	41.92 ms	1.28 µs	83.84 ms	0.5 µs	32.75 ms	1 µs	65.5 ms

TC1CK	SLOW, SLEEP Mode	
	Resolution [µs]	Maximum Time Setting [s]
00	244.14 µs	16.0 s
01	—	—
10	—	—

Example 1: Sets the timer mode with source clock $fs/2^3$ [Hz] and generates an interrupt 1 later (at fs = 32.8 kHz).

```
LDW     (TREG1A), 1000H     ;  Sets the timer register
                            ;  (1 s ÷ 2^3/fs = 1000H)
SET     (EIRL). EF4
EI                          ;  Enable INTTC1
LD      (TC1CR), 00010000B  ;  Starts TC1
```

Note: TC1CR is a write-only register and must not be used with [SET(TC1CR).4] instruction.

Example 2: Software capture

```
LD      (TC1CR), 01010000B  ;  SCAP1 ← 1 (Captures)
LD      WA, (TREG1B)        ;  Reads the capture value
```

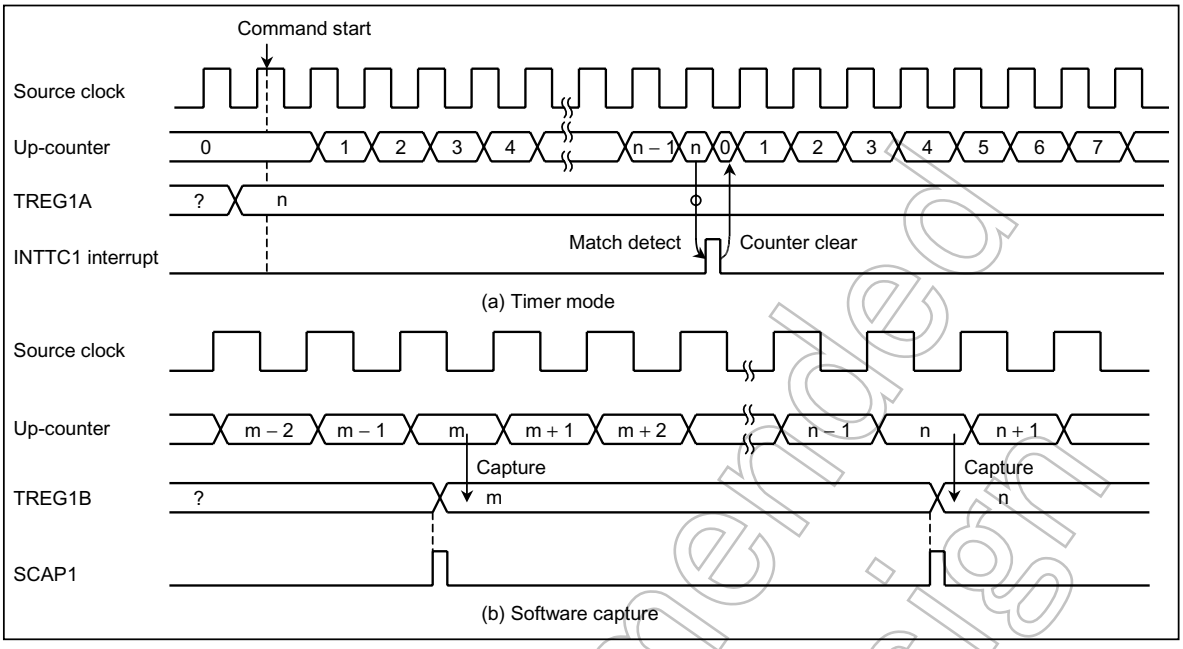


Figure 2.6.3 Timer Mode Timing Chart

(2) External trigger timer mode

In this mode, counting up is started by an external trigger. This trigger is the edge of the TC1 pin input. Either the rising or falling edge can be selected with INT2ES in EINTCR. Edge selection is the same as for INT3 pin. Source clock is an internal clock selected with TC1CK. The contents of TREG1A is compared with the contents of up-counter. If a match is found, an INTTC1 interrupt is generated, and the counter is cleared to "0" and halted. The counter is restarted by the selected edge of the TC1 pin input.

When METT1 (bit 6 in TC1CR) is "1", inputting the edge to the reverse direction of the trigger edge to start counting clears the counter, and the counter is stopped. Inputting a constant pulse width can generate interrupts. When METT1 is "0", the reverse directive edge input is ignored. The TC1 pin input edge before a match detection is also ignored.

The TC1 pin input has the same noise rejection as the INT3 pin; therefore, pulses of 7/fc [s] or less are rejected as noise. A pulse width of 24/fc [s] or more is required for edge detection in NORMAL1, 2 or IDLE1, 2 mode. The noise rejection circuit is turned off in SLOW and SLEEP modes. But, a pulse width of one machine cycle or more is required.

Example 1: Detects rising edge in TC1 pin input and generates an interrupt 100 μs later. (at fc = 12.5 MHz, DV1CK = 1)

```
LD      (EINTCR), 00000000B     ;   INT3ES ← 0 (rising edge)
LDW     (TREG1A), 004EH         ;   100 μs ÷ 2^4/fc = 4EH
SET     (EIRL).EF4              ;   INTTC1 interrupt enable
EI
LD      (TC1CR), 00111000B      ;   TC1 external trigger start, METT1 = 0
```

Example 2: Generates an interrupt, inputting "L" level pulse (pulse width: 4 ms or more) to the TC1 pin. (at fc = 12.5 MHz, DV1CK = 1)

```
LD      (EINTCR), 00000100B     ;   INT2ES ← 1 ("L" level)
LDW     (TREG1A), 00C3H         ;   4 ms ÷ 28/fc = C3H
SET     (EIRL).EF4              ;   INTTC1 interrupt enable
EI
LD      (TC1CR), 01110100B      ;   TC1 external trigger start, METT1 = 1
```

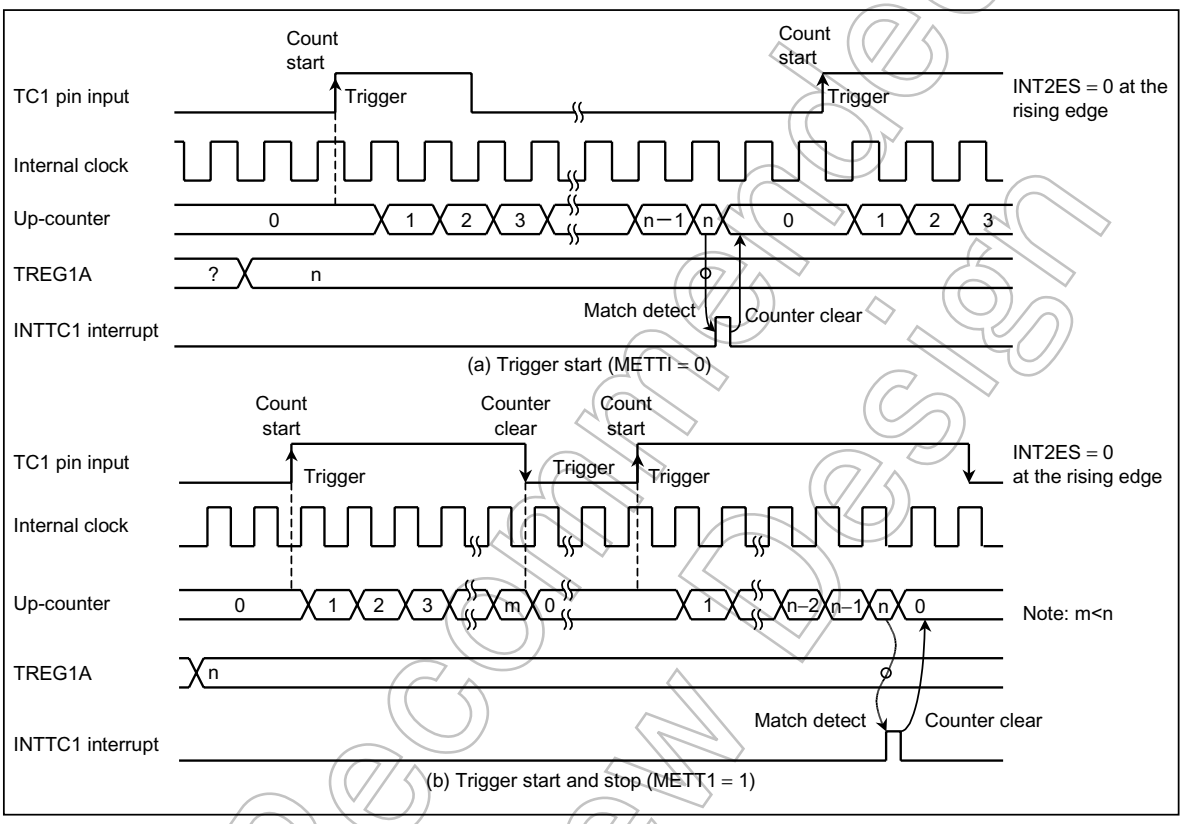


Figure 2.6.4 External Trigger Timer Mode Timing Chart

(3) Event counter mode

In this mode, events are counted at the edge of the TC1 pin input. Either the rising or falling edge can be selected with INT2ES in EINTCR. Edge selection is the same as for INT3 pin. The contents of TREG1A are compared with the contents of up-counter. If a match is found, an INTTC1 interrupt is generated, and the counter is cleared. This maximum applied frequency is shown in Table 2.6.2.

Setting SCAP1 to "1" transfers the current contents of up-counter to TREG1B (software capture function). SCAP1 is automatically cleared after capturing.

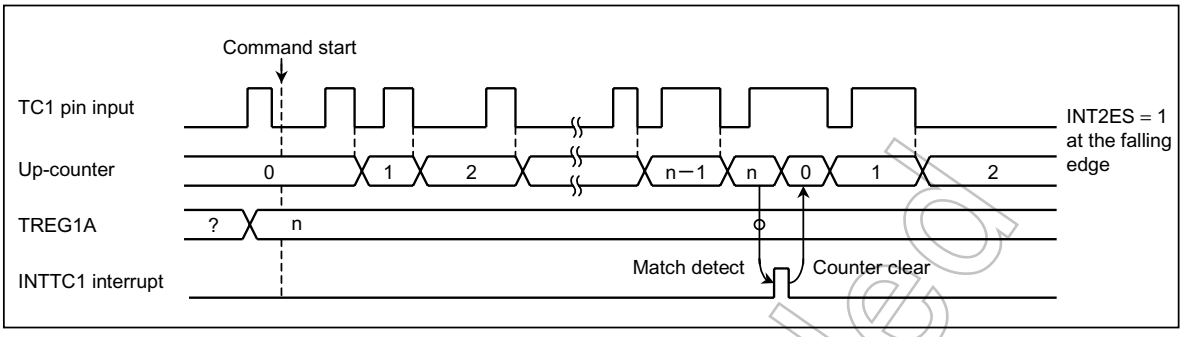


Figure 2.6.5 Event Counter Mode Timing Chart

Table 2.6.2 Timer/Counter 1 External Clock Source

Maximum Applied Frequency [Hz]	
NORMAL1/2, IDLE1/2 Mode	SLOW, SLEEP Mode
$fc/2^4$	$fs/2^4$

(4) Window mode

Counting up is performed on the rising edge of the pulse that is the logical AND-ed product of the TC1 pin input (window pulse) and an internal clock. The contents of TREG1A are compared with the contents of up-counter. If a match is found, an INTTC1 interrupt is generated, and the counter is cleared. Positive or negative logic for the TC1 pin input can be selected. Edge selection is the same as for INT3 pin. Setting SCAP1 to "1" transfers the current contents of up-counter to TREG1B.

It is necessary that the maximum applied frequency be such that the counter value can be analyzed by the program. That is; the frequency must be considerably slower than the selected internal clock.

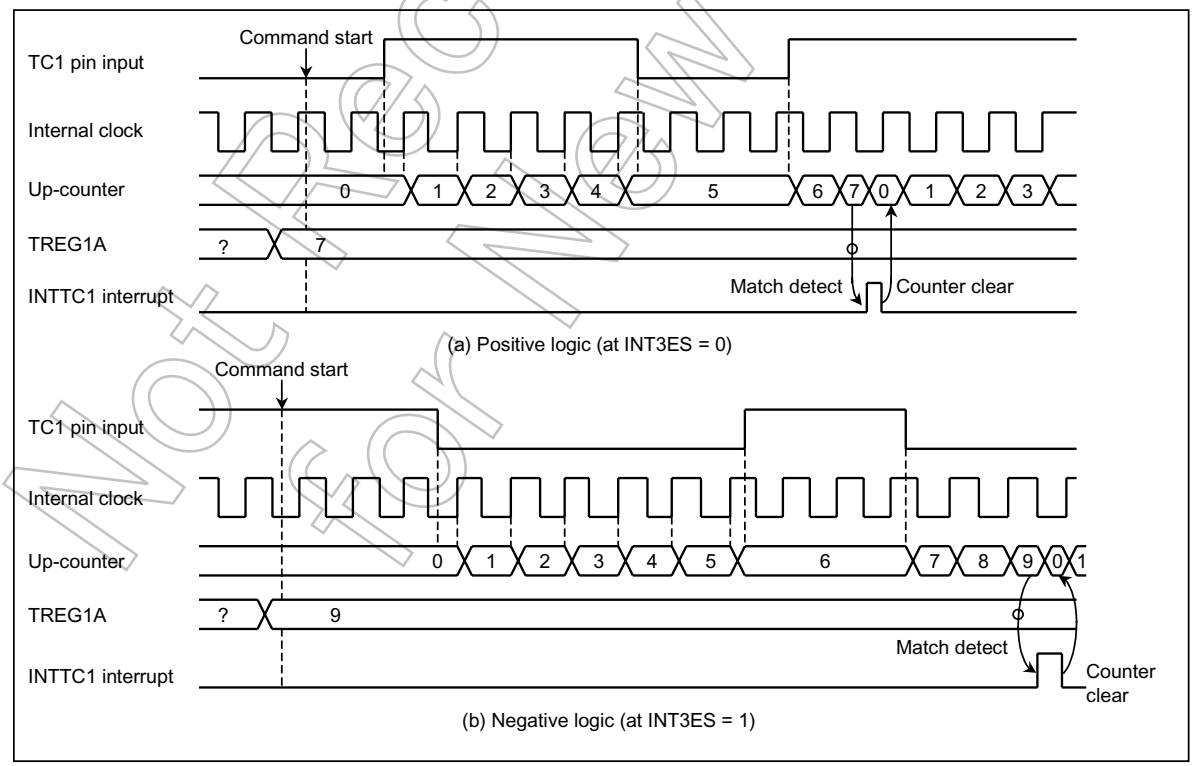


Figure 2.6.6 Window Mode Timing Chart

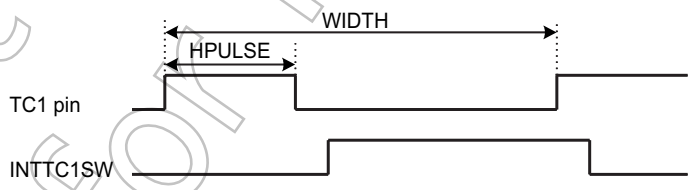
(5) Pulse width measurement mode

Counting is started by the external trigger (set to external trigger start by TC1S). The trigger can be selected either the rising or falling edge of the TC1 pin input. The source clock is used an internal clock. On the next falling (rising) edge, the counter contents are transferred to TREG1B and an INTTC1 interrupt is generated. The counter is cleared when the single edge capture mode is set. When double edge capture is set, the counter continues and, at the next rising (falling) edge, the counter contents are again transferred to TREG1B. If a falling (rising) edge capture value is required, it is necessary to read out TREG1B contents until a rising (falling) edge is detected. Falling or rising edge is selected with INT3ES, and single edge or double edge is selected with MCAP1 (bit 6 in TC1CR).

Note: The first captured value after the timer starts may be read incorrectively, therefore, ignore the first captured value.

Example: Duty measurement (resolution $fc/2^7$ [Hz] DV1CK = 0)

```
                CLR     (INTTC1SW). 0       ;   INTTC1 service switch initial setting
                LD      (EINTCR), 00000000B ;   Sets the rise edge at the INT3 edge
                LD      (TC1CR), 00000110B  ;   Sets the TC1 mode and source clock
                SET     (EIRL). EF4         ;   Enables INTTC1
                EI
                LD      (TC1CR), 00110110B  ;   Starts TC1 with an external trigger at
                :                               MCAP1=0
PINTTC1:        CPL     (INTTC1SW). 0       ;   Complements INTTC1 service switch
                JRS     F, SINTTC1
                LD      (HPULSE), (TREG1BL) ;   Reads TREG1B ("H" level pulse width)
                LD      (HPULSE + 1), (TREG1BH)
                RETI
SINTTC1:        LD      (WIDTH), (TREG1BL)  ;   Reads TREG1B (Period)
                LD      (WIDTH + 1), (TREG1BH)  ;   Duty calculation
                :
                RETI
                :
VINTTC1:        DW      PINTTC1
```



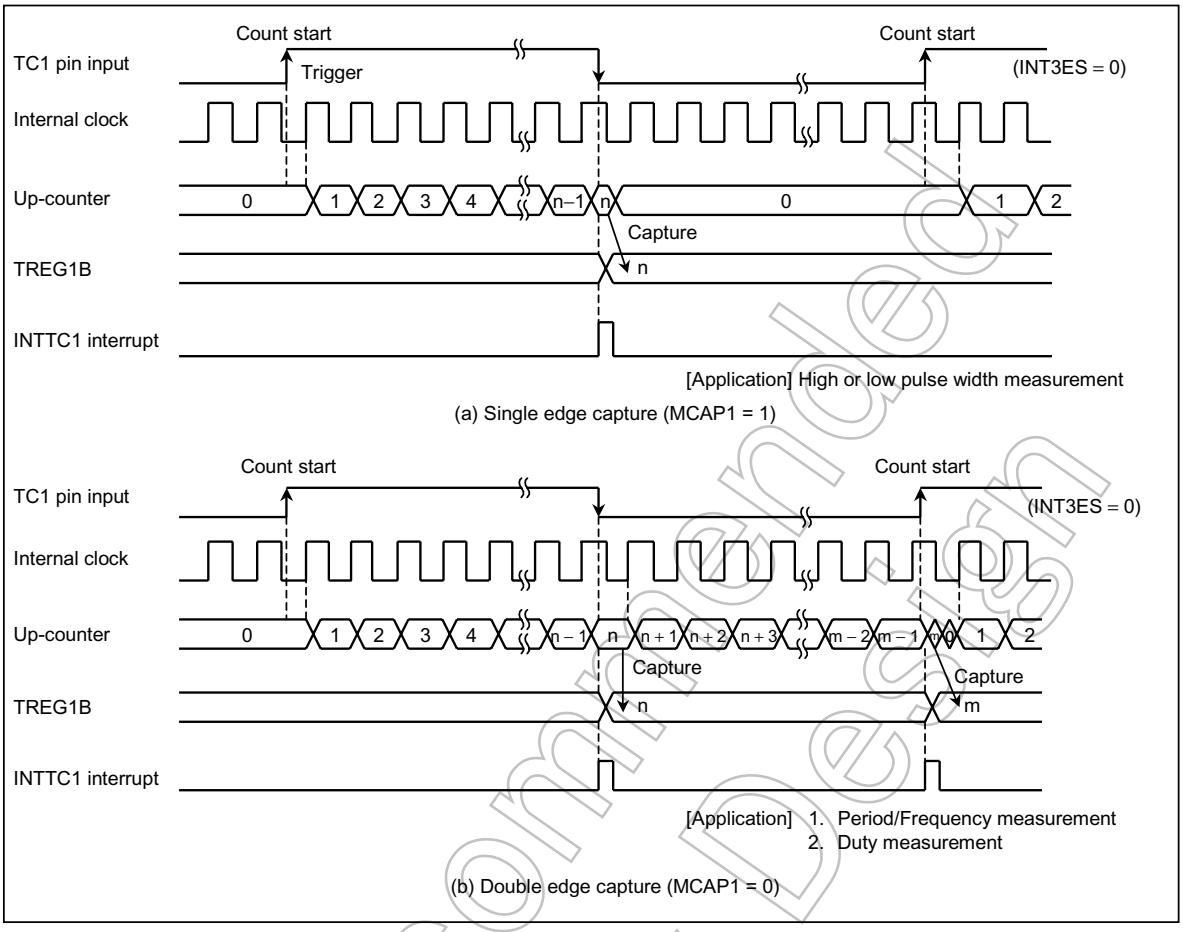


Figure 2.6.7 Pulse Width Measurement Mode Timing Chart

(6) Programmable pulse generate (PPG) output mode

Counting is started by an edge of the TC1 pin input (either the rising or falling edge can be selected) or by a command. The source clock is used an internal clock. First, the contents of TREG1B are compared with the contents of the up-counter. If a match is found, timer F/F1 output is toggled. When MPPG1 = 0, an INTTC1 interrupt is generated. Next, timer F/F is again toggled and the counter is cleared by matching with TREG1A. An INTTC1 interrupt is generated at this time. Timer F/F output is connected to the P12 ($\overline{PPG}$) pin. In the case of $\overline{PPG}$ output, set the P12 output latch to "1" and configure as an output mode. Timer F/F1 is cleared to "0" during reset. The timer F/F 1 value can also be set by TFF1 (bit 7 in TC1CR) and either a positive or negative logic pulse output is available. Also, writing to the TREG1B is not possible unless the timer/counter 1 is set to the PPG output mode.

Example: Pulse output "H" level 800 μs, "L" level 200 μs (at fc = 12.5 MHz, DV1CK = 0)

```
SET     (P1).2                  ;   P12 output latch ← 1
LD      (P1CR), 00000100B       ;   Sets the P12 output mode
LD      (TC1CR), 10001011B      ;   Sets the PPG output mode
LDW     (TREG1A), 07D0H         ;   Sets the period (1 ms ÷ 0.64 μs = 061AH)
LDW     (TREG1B), 0190H         ;   Sets "L" level pulse width
                                    (200 μs ÷ 0.64 μs = 0138H)
LD      (TC1CR), 10011011B      ;   Starts
```

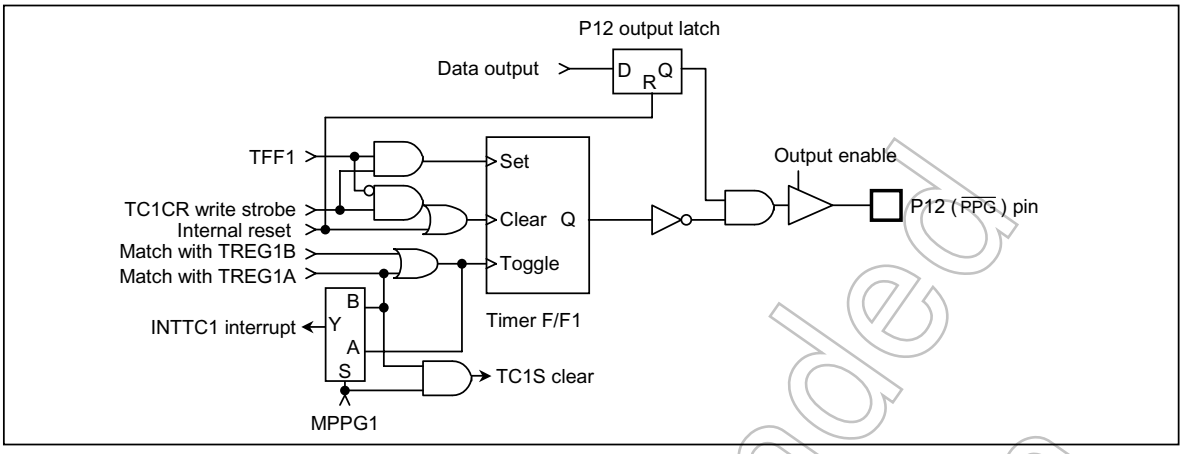


Figure 2.6.8 $\overline{\text{PPG}}$ Output

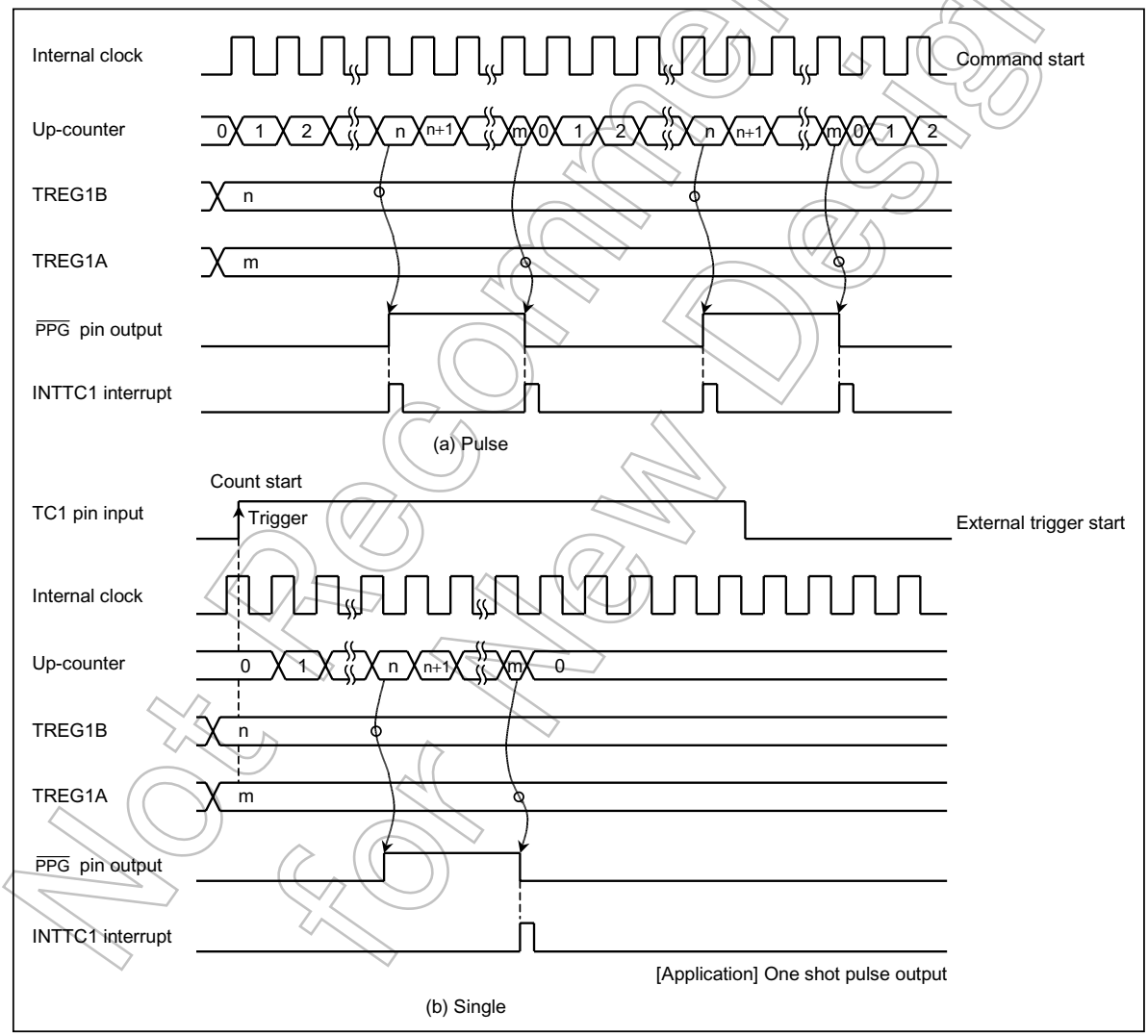


Figure 2.6.9 $\overline{\text{PPG}}$ Output Mode Timing Chart

2.7 16-Bit Timer/Counter 2 (TC2)

2.7.1 Configuration

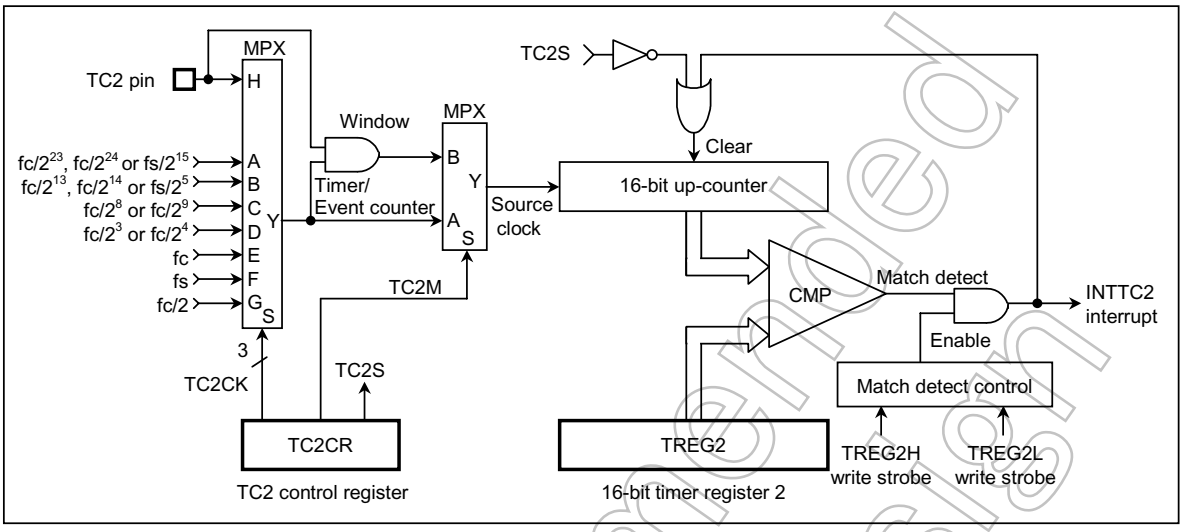


Figure 2.7.1 Timer/Counter 2 (TC2)

2.7.2 Control

The timer/counter 2 is controlled by a timer/counter 2 control register (TC2CR) and a 16-bit timer register 2 (TREG2). Reset does not affect TREG2.

TREG2 (00016, 00017H)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TREG2H (00017H)								TREG2L (00016H)							

Write only

TC2CR (00015H)

7	6	5	4	3	2	1	0
"0"	"0"	TC2S	TC2CK				TC2M

(Initial value: **00 00*0)

TC2M	TC2 operating mode select	0: Timer/Event counter mode 1: Window mode							
TC2CK	TC2 source clock select [Hz]		NORMAL1/2, IDLE1/2 mode				SLOW mode	SLEEP mode	Write only
			DV7CK = 0		DV7CK = 1				
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1			
		000	$fc/2^{23}$	$fc/2^{24}$	$fs/2^{15}$	$fs/2^{15}$	$fs/2^{15}$	$fs/2^{15}$	
		001	$fc/2^{13}$	$fc/2^{14}$	$fs/2^{5}$	$fs/2^{5}$	$fs/2^{5}$	$fs/2^{5}$	
		010	$fc/2^{8}$	$fc/2^{9}$	$fc/2^{8}$	$fc/2^{9}$	—	—	
		011	$fc/2^{3}$	$fc/2^{4}$	$fc/2^{3}$	$fc/2^{4}$	—	—	
		100	—	—	—	—	fc	—	
		101	fs	fs	fs	fs	—	—	
		110	—	—	—	—	fc/2	—	
		111	External clock (TC2 pin input)						
TC2S	TC2 start control	0: Stop and counter clear 1: Start							

Note 1: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz], fcgck: Gear clock [Hz], *: Don't care

Note 2: Writing to the lower byte of timer register 2 (TREG2L), the comparison is inhibited until the upper byte (TREG2H) is written. After writing to the upper byte, any match during 1 machine cycle (instruction execution cycle) is ignored.

Note 3: Set the mode and source clock when the TC2 stops (TC2S = 0).

Note 4: Values to be loaded to the timer register must satisfy the following condition.
$TREG2 > 0$ ($TREG2_{15\ to\ 11} > 0$ at warm-up)

Note 5: "fcgck" can be selected as the source clock only in the timer mode during the SLOW mode.

Note 6: TC2CR and TREG2 are write-only registers and must not be used with any of the read-modify-write instructions such as SET, CLR, etc.

Note 7: It recommends when used as an TC2CK = <100>, at fc ≤ 8 MHz, and used as an TC2CK = <110>, at fc = 12.5 MHz.

Figure 2.7.2 Timer Register 2 and TC2 Control Register

2.7.3 Function

The timer/counter 2 has three operating modes: timer, event counter and window modes. Also timer/counter 2 is used for warm-up when switching from SLOW mode to NORMAL2 mode.

(1) Timer mode

In this mode, the internal clock is used for counting up. The contents of TREG2 are compared with the contents of up-counter. If a match is found, a timer/counter 2 interrupt (INTTC2) is generated, and the counter is cleared. Counting up is resumed after the counter is cleared.

Also, when "fcgck" is selected as the source clock during SLOW mode, the lower 11 bits of TREG2 are ignored and an INTTC2 interrupt is generated by matching the upper 5 bits. Thus, in this case, only the TREG2H setting is necessary.

Table 2.7.1 Source Clock (Internal clock) for Timer/Counter 2 (at fc = 12.5 MHz, fs = 32.8 kHz)

TC2CK	NORMAL1/2, IDLE1/2 Mode							
	DV7CK = 0				DV7CK = 1			
	DV1CK = 0		DV1CK = 1		DV1CK = 0		DV1CK = 1	
	Resolution	Maximum Time Setting	Resolution	Maximum Time Setting	Resolution	Maximum Time Setting	Resolution	Maximum Time Setting
000	671 ms	12.2 h	1.34 s	24.4 h	1 s	18.2 h	1 s	18.2 h
001	655.36 µs	43.0 s	1.31 ms	1.4 min	0.98 ms	1.07 min	0.98 ms	1.07 min
010	20.48 µs	1.34 s	40.96 µs	2.7 s	16 µs	1.05 s	32 µs	2.1 s
011	0.64 µs	41.92 ms	1.28 µs	83.8 ms	0.5 µs	32.75 ms	1 µs	65.5 ms
100	—	—	—	—	—	—	—	—
101	30.5 µs	2 s	30.5 µs	2 s	30.5 µs	2 s	30.5 µs	2 s

TC2CK	SLOW Mode		SLEEP Mode	
	Resolution [s]	Maximum Time Setting	Resolution [s]	Maximum Time Setting
000	1 s	18.2 h	1 s	18.2 h
001	0.98 ms	1.07 min	0.98 ms	1.07 min
01*	—	—	—	—
100	125 ns (Note)	—	—	—
110	160 ns (Note)	—	—	—
101	—	—	—	—

Note: fc and fc/2 can be used only in the timer mode. It is used for warm-up when switching from SLOW mode to NORMAL2 mode. (at fc = 8 MHz, TC2CK = <100>)

Example: Sets the timer mode with source clock $fc/2^4$ [Hz] and generates an interrupt every 25 ms (at fc = 12.5 MHz, DV1CK = 1).

```
LDW     (TREG2), 4C46H          ;  Sets TREG2 (25 ms ÷ 2^4/fc = 4C46H)
SET     (EIRH). EF14            ;  Enables INTTC2 interrupt
EI
LD      (TC2CR), 00101100B      ;  Starts TC2
```

(2) Event counter mode

In this mode, events are counted on the rising edge of the TC2 pin input. The contents of TREG2 are compared with the contents of the up-counter. If a match is found, an INTTC2 interrupt is generated, and the counter is cleared. The maximum frequency applied to the TC2 pin is shown in Table 2.7.2. Two or more machine cycles are required for both the "H" and "L" levels of the pulse width.

```
Example:  Sets the event counter mode and generates an INTTC2 interrupt 640 counts later.
          LDW     (TREG2), 640            ;  Sets TREG2
          SET     (EIRH). EF14            ;  Enables INTTC2 interrupt
          EI
          LD      (TC2CR), 00111100B      ;  Starts TC2
```

Table 2.7.2 Timer/Counter 2 External Clock Source

Maximum Applied Frequency [Hz]	
NORMAL1/2, IDLE1/2 Mode	SLOW, SLEEP Mode
$fc/2^4$	$fs/2^4$

(3) Window mode

In this mode, counting up is performed on the rising edge of an internal clock during TC2 external pin input(window pulse) is "H" level. The contents of TREG2 are compared with the contents of up-counter. If a match is found, an INTTC2 interrupt is generated, and the up-counter is cleared.

The maximum applied frequency (TC2 input) must be considerably slower than the selected internal clock.

```
Example:  Generates an interrupt, inputting "H" level pulse width of 120 ms or more. (at fc = 12.5 MHz,
          DV1CK = 1)
          LDW     (TREG2), 0056H          ;  Sets TREG2 (120 ms ÷ 2^14/fc = 0056H)
          SET     (EIRH). EF14            ;  Enables INTTC2 interrupt
          EI
          LD      (TC2CR), 00100101B      ;  Starts TC2
```

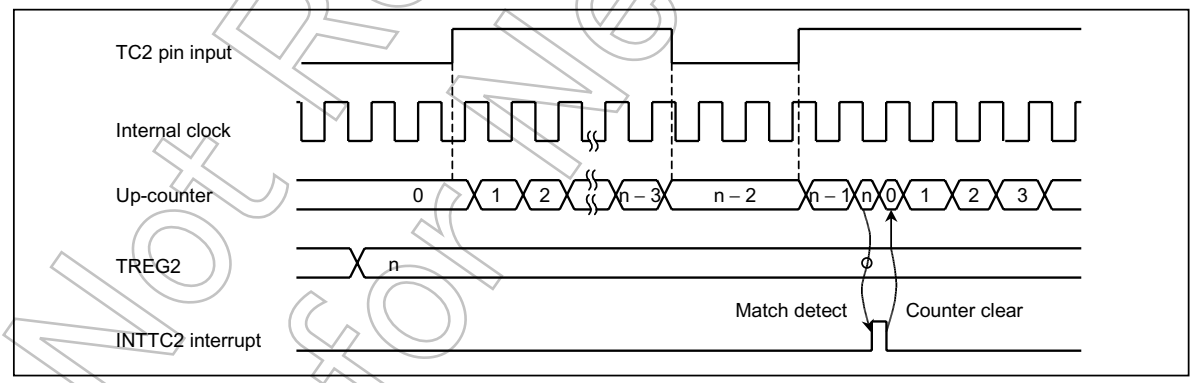


Figure 2.7.3 Window Mode Timing Chart

2.8 8-Bit Timer/Counter 3 (TC3)

2.8.1 Configuration

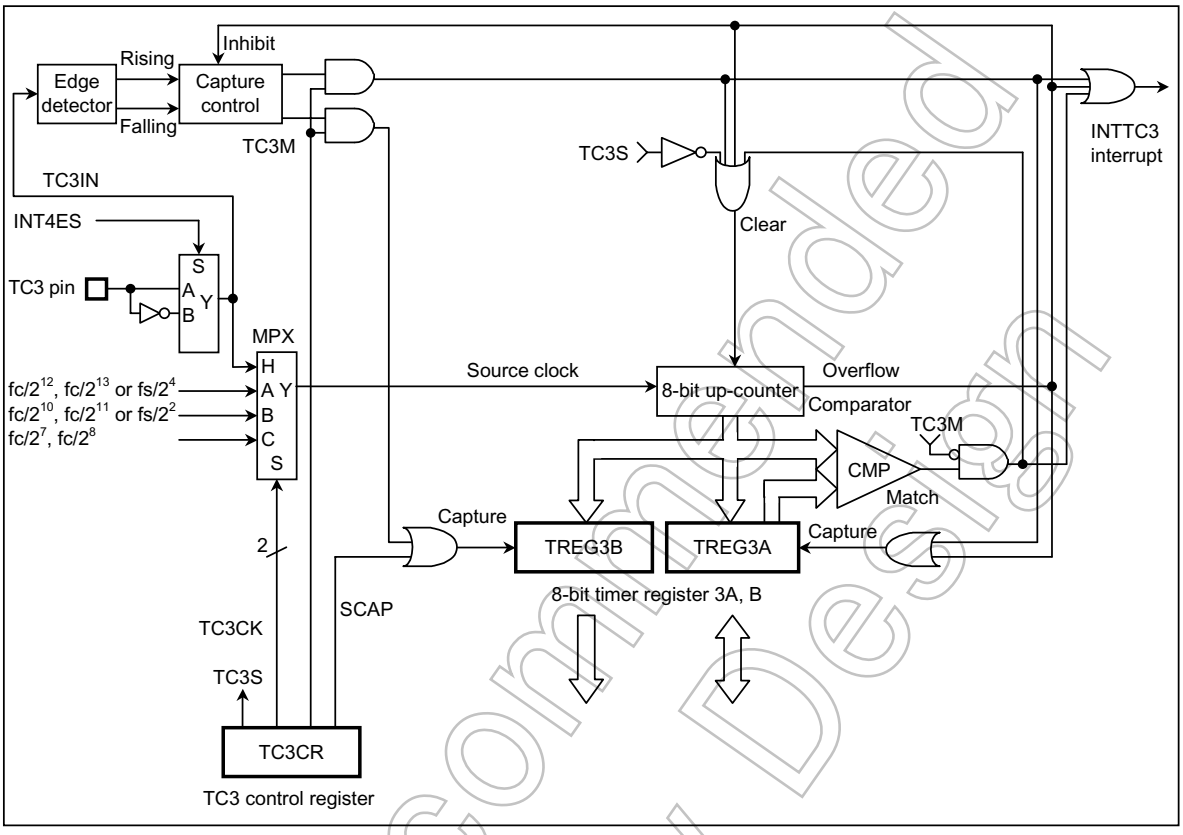


Figure 2.8.1 Timer/Counter 3 (TC3)

2.8.2 Control

The timer/counter 3 is controlled by a timer/counter 3 control register (TC3CR) and two 8-bit timer registers (TREG3A and TREG3B).

TREG3A (00018H): 7 6 5 4 3 2 1 0 — Read/Write

TREG3B (00019H): — Read only

TC3CR (0001AH): 7 "0" | 6 SCAP | 5 "0" | 4 TC3S | 3 2 TC3CK | 1 "0" | 0 TC3M — (Initial value: *0*0 00*0)

TC3M	TC3 operation mode set	0: Timer/event counter 1: Capture					Write only
TC3CK	TC3 source clock select [Hz]		NORMAL1/2, IDLE1/2 mode				
			DV7CK = 0		DV7CK = 1		SLOW, SLEEP mode
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1	
		00	$fc/2^{12}$	$fc/2^{13}$	$fs/2^4$	$fs/2^4$	$fs/2^4$
		01	$fc/2^{10}$	$fc/2^{11}$	$fs/2^2$	$fs/2^2$	—
		10	$fc/2^7$	$fc/2^8$	$fs/2^7$	$fs/2^8$	—
		11	External clock (TC3 pin input)				
TC3S	TC3 start select	0: Stop and clear 1: Start					
SCAP	Software capture control	0: — 1: Software capture trigger					

Note 1: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz], *: Don't care
Note 2: Set the mode, the source clock and the edge selection (INT3ES) when the TC3 stops (TC3S = 0).
Note 3: Software capture can be used only in the timer and event counter mode. SCAP is automatically cleared to "0" after capturing.
Note 4: Values to be loaded into timer register 3A must satisfy the following condition.
TREG3A>0 (in the timer and event counter mode)
Note 5: TC3CR is a write-only register and must not be used with any of read-modify-write instructions such as SET, CLR, etc.

Figure 2.8.2 Timer Register 3 and TC3 Control Register

2.8.3 Function

The timer/counter 3 has three operating modes: timer, event counter, and capture mode. When it is used in the capture mode, the noise rejection time of TC3 pin input can be set by remote control receive control register.

(1) Timer mode

In this mode, the internal clock is used for counting up. The contents of TREG3A are compared with the contents of up-counter. If a match is found, a timer/counter 3 interrupt (INTTC3) is generated, and the up-counter is cleared. Counting up resumes after the up-counter is cleared. The current contents of up-counter are loaded into TREG3B by setting SCAP (bit 6 in TC3CR) to "1". SCAP is automatically cleared after capturing.

Table 2.8.1 Source Clock (Internal Clock) for Timer/Counter 3 (Example: at fc = 12.5 MHz, fs = 32.8 kHz)

TC3CK	NORMAL1/2, IDLE1/2 Mode							
	DV7CK = 0				DV7CK = 1			
	DV1CK = 0		DV1CK = 1		DV1CK = 0		DV1CK = 1	
	Resolution [μs]	Maximum Setting Time [ms]	Resolution [μs]	Maximum Setting Time [ms]	Resolution [μs]	Maximum Setting Time [ms]	Resolution [μs]	Maximum Setting Time [ms]
00	327.68	83.6	655.36	167.8	488.28	124.5	488.28	124.5
01	81.92	20.9	163.84	41.7	122.07	31.1	122.07	31.1
10	10.24	2.6	10.48	5.2	8	2.0	16	4.1

TC3CK	SLOW, SLEEP Mode	
	Resolution [μs]	Maximum Setting Time [ms]
00	488.28	124.5

(2) Event counter mode

In this mode, the TC3 pin input pulses are used for counting up. Either the rising or falling edge can be selected. Edge selection is the same as for INT3 pin. The contents of TREG3A are compared with the contents of the up-counter. If a match is found, an INTTC3 interrupt is generated and the counter is cleared.

The maximum applied frequency is shown in Table 2.8.2. Two or more machine cycles are required for both the high and low levels of the pulse width.

The current contents of up-counter are loaded into TREG3B by setting SCAP (bit 6 in TC3CR) to "1". SCAP is automatically cleared to "0" after capturing.

```
Example:  Generates an interrupt every 0.5 s, inputting 50 Hz pulses to the TC3 pin.
          LD      (TREG3A), 19H          ;  0.5 s ÷ 1/50 = 25 = 19H
          LD      (TC3CR), 00011110B     ;  Starts TC3
```

Table 2.8.2 Source Clock (External Clock) for Timer/Counter

Maximum Applied Frequency [Hz]	
NORMAL1/2, IDLE1/2 Mode	SLOW, SLEEP Mode
$fc/2^4$	$fs/2^4$

(3) Capture mode

The pulse width, period and duty of the TC3 pin input are measured in this mode, which can be used in decoding the remote control signals or distinguishing AC 50/60 Hz, etc. The counter is free running by the internal clock. On the rising (falling) edge of the TC3 pin input, the current contents of counter is loaded into TREG3A, then the up-counter is cleared to "0" and an INTTC4 interrupt is generated. On the falling (rising) edge of the TC3 pin input, the current contents of the counter is loaded into TREG3B. In this case, counting continues. On the next rising (falling) edge of the TC3 pin input, the current contents of counter are loaded into TREG3A, then the counter is cleared again and an interrupt is generated. If the counter overflows before the edge is detected. FF_H is set into TREG3A, and the counter is cleared and an INTTC3 interrupt is generated. During interrupt processing, it can be determined whether or not there is an overflow by checking whether or not the TREG3A value is FF_H. Also, after an interrupt (capture to TREG3A, or overflow detection) is generated, capture and overflow detection are halted until TREG3A has been read out; however, the counter continues. As reading out TREG3A resumes capture/overflow detection, TREG3B must be beforehand read out.

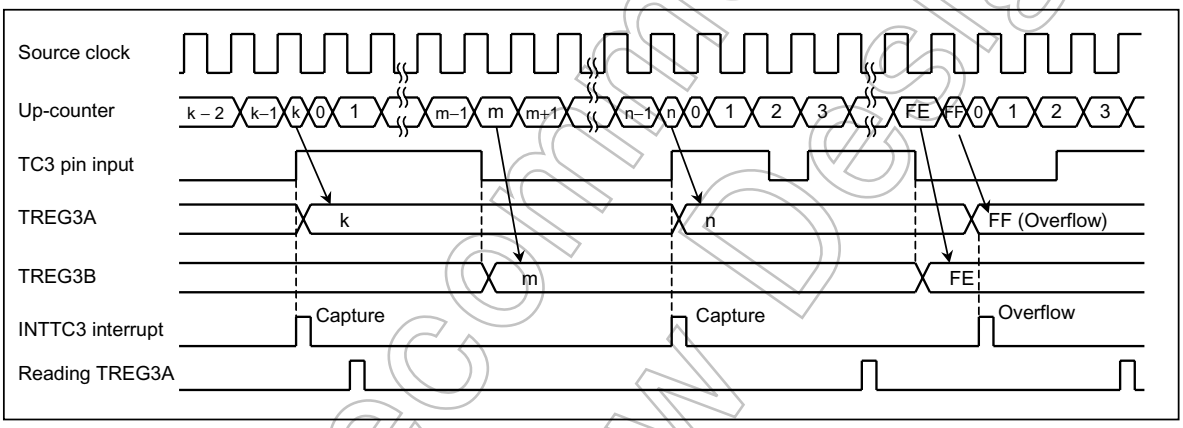


Figure 2.8.3 Capture Mode Timing Chart (at INT4ES = 0)

2.9 8-Bit Timer/Counter 4 (TC4)

2.9.1 Configuration

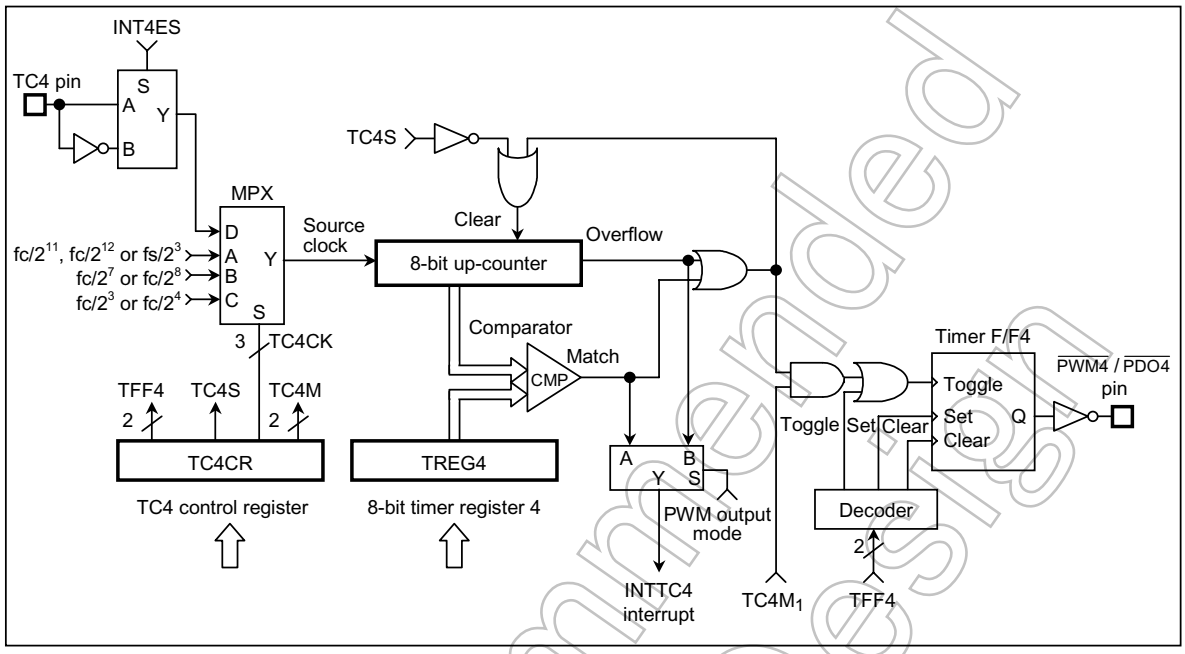


Figure 2.9.1 Timer/Counter 4 (TC4)

2.9.2 Control

The timer/counter 4 is controlled by a timer/counter 4 control register (TC4CR) and an 8-bit timer register 4 (TREG4). Reset does not affect TREG4.

TREG4 (0001BH)

7	6	5	4	3	2	1	0

Write only

TC4CR (0001CH)

7	6	5	4	3	2	1	0
TFF4		"0"	TC4S	TC4CK		TC4M	

(Initial value: 00*0 0000)

TC4M	TC4 operating mode select	00: Timer 01: Reserved 10: Programmable divider output (PDO) mode 11: Pulse width modulation (PWM) output mode						
TC4CK	TC4 source clock select [Hz]		NORMAL1/2, IDLE1/2 mode				SLOW, SLEEP mode	Write only
			DV7CK = 0		DV7CK = 1			
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1		
		00	$fc/2^{11}$	$fc/2^{12}$	$fs/2^{3}$	$fs/2^{3}$	$fs/2^{3}$	
		01	$fc/2^{7}$	$fc/2^{8}$	$fc/2^{7}$	$fc/2^{8}$	—	
		10	$fc/2^{3}$	$fc/2^{4}$	$fc/2^{3}$	$fc/2^{4}$	—	
		11	Reserved					
TC4S	TC4 start control	0: Stop and counter clear 1: Start						
TFF4	Timer F/F4 control	00: Clear 01: Reserved 10: Reserved 11: — (Note 3)						

Note 1: fc: High-frequency clock [Hz], fs: Low-frequency clock [Hz], *: Don't care

Note 2: Set the operating mode, the source clock selection, the edge selection (INT4ES) and timer F/F4 control when the TC4 stops (TC4S = 0)

Note 3: Set TFF4 to "11" in the timer and event counter mode and PWM mode.

Note 4: Values to be loaded to the timer register must satisfy the following condition.
0<TREG4 in others

Note 5: TC4CR is a write-only register and must not be used with any of the read-modify-write instructions such as SET, CLR, etc.

Figure 2.9.2 Timer Register 4 and TC4 Control Register

2.9.3 Function

The timer/counter 4 has four operating modes: timer, event counter, programmable divider output, and PWM output mode.

(1) Timer mode

In this mode, the internal clock is used for counting up. The contents of TREG4 are compared with the contents of up-counter. If a match is found, an INTTC4 interrupt is generated and the up-counter is cleared to "0". Counting up resumes after the up-counter is cleared.

Table 2.9.1 Source Clock (Internal clock) for Timer/Counter 4 (Example: at fc = 12.5 MHz, fs = 32.8 kHz)

TC4CK	NORMAL1/2, IDLE1/2 Mode							
	DV7CK = 0				DV7CK = 1			
	DV1CK = 0		DV1CK = 1		DV1CK = 0		DV1CK = 1	
	Resolution [μs]	Maximum Setting Time [s]	Resolution [μs]	Maximum Setting Time [s]	Resolution [μs]	Maximum Setting Time [s]	Resolution [μs]	Maximum Setting Time [s]
00	163.84 μs	41.7 ms	327.68 μs	83.6 ms	244.14 μs	62.2 ms	244.14 μs	62.2 ms
01	10.24 μs	2.6 ms	20.48 μs	5.2 ms	10.24 μs	2.6 ms	20.48 μs	5.2 ms
10	0.64 μs	163.2 μs	1.28 μs	326 μs	0.64 μs	163.2 μs	1.28 μs	326 μs

TC4CK	SLOW, SLEEP Mode	
	Resolution [μs]	Maximum Setting Time [s]
00	244.14 μs	62.2 ms
01	—	—
10	—	—

(2) Programmable divider output (PDO) mode

The internal clock is used for counting up. The contents of TREG4 are compared with the contents of the up-counter. If a match is found, the timer F/F 4 output is toggled and the counter is cleared. Timer F/F 4 output is inverted and output to the P14 ($\overline{PPO4}$) pin. When programmable divider output is executed, P14 output latch is set to "1". This mode can be used for approximate 50% duty pulse output. Timer F/F 4 can be initialized by program, and it is initialized to "0" during reset. An INTTC4 interrupt is generated each time the $\overline{PDO}$ output is toggled.

```
Example: Output a 1024 Hz pulse (at fc = 12.5 MHz)
        SET     (P1). 4                 ;   P14 output latch ← 1
        LD      (P1CR), 00010000B       ;   Set output mode to P14
        LD      (TREG4), 5FH            ;   1/1024 ÷ 2^7/fc = 5FH
        LD      (TC4CR), 00010010B      ;   Starts TC4
```

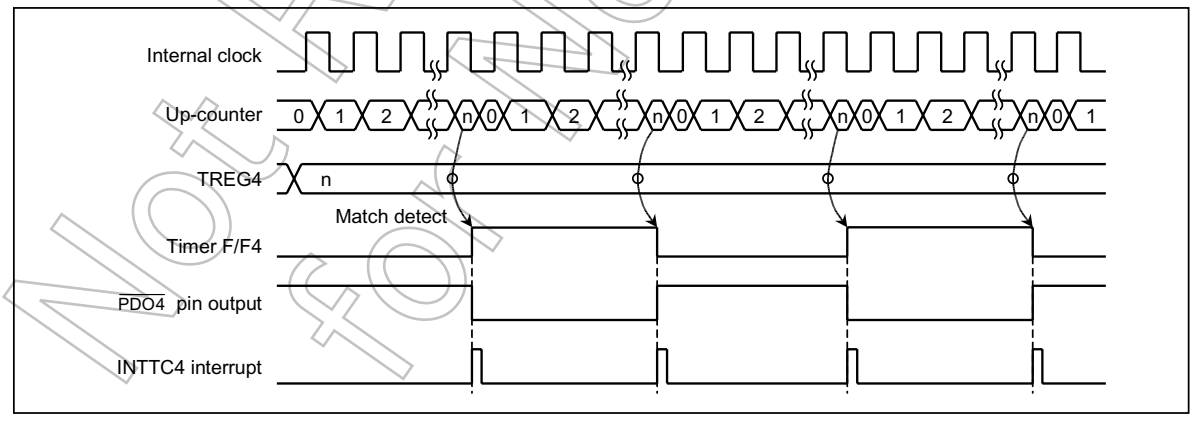


Figure 2.9.3 PDO Mode Timing Chart

(3) Pulse width modulation (PWM) output mode

PWM output with a resolution of 8 bits is possible. The internal clock is used for counting up. The contents of TREG4 are compared with the contents of up-counter. If a match is found, the timer F/F 4 output is toggled. Counting up resumes. And, when an overflow occurs, the timer is again toggled and the counter is cleared. Timer F/F 4 output is inverted and output to the P14 ($\overline{PWM4}$) pin. When programmable divider output is executed, P14 output latch is set to "1". An INTTC4 interrupt is generated when an overflow occurs.

TREG4 is configured a 2-stage shift register and, during output, will not switch until one output cycle is completed even if TREG4 is overwritten; therefore, output can be altered continuously. Also, the first time, TREG4 is shifted by setting TC4S (bit 4 in TC4CR) to "1" after data are loaded to TREG4.

Note: Do not rewrite the contents of TREG4 at only an INTTC4 interrupt generation cycle. The contents of TREG4 is rewritten by the INTTC4 interrupt service routine.

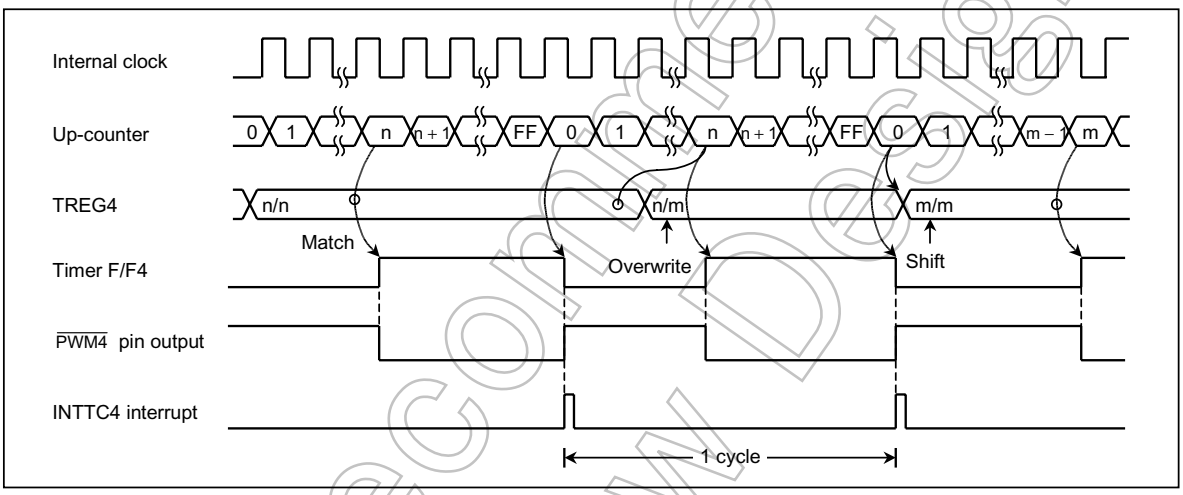


Figure 2.9.4 PWM Output Mode Timing Chart

Table 2.9.2 PWM Output Mode (Example: fc = 12.5 MHz)

TC4CK	NORMAL1/2, IDLE1/2 Mode							
	DV7CK = 0				DV7CK = 1			
	DV1CK = 0		DV1CK = 1		DV1CK = 0		DV1CK = 1	
	Resolution [μs]	Repeat Cycle [ms]	Resolution [μs]	Repeat Cycle [ms]	Resolution [μs]	Repeat Cycle [ms]	Resolution [μs]	Repeat Cycle [ms]
00	163.84 μs	41.7 ms	327.68 μs	83.6 ms	244.14 μs	62.5 ms	244.14 μs	62.5 ms
01	10.24 μs	2.6 ms	20.48 μs	5.2 ms	—	—	—	—
10	0.64 μs	163.2 μs	1.28 μs	326 μs	—	—	—	—

TC4CK	SLOW, SLEEP Mode	
	Resolution [μs]	Repeat Cycle [ms]
00	244.14 μs	62.5 ms
01	—	—
10	—	—

2.10 Serial Bus Interface (SBI-ver.C)

The TMP88CU74 has a 1-channel serial bus interface which employs a clocked-synchronous 8-bit serial bus interface and an I²C bus (a bus system by Philips).

The serial bus interface is connected to an external device through P31 (SDA) and P30 (SCL) in the I²C bus mode; and through P32 ($\overline{SCK0}$), P32 (SO0) and P30 (SI0) in the clocked-synchronous 8-bit SIO mode.

The serial bus interface pins are also used for the P3 port. When used for serial bus interface pins, set the P3 output latches of these pins to "1", and control inputs and outputs of these pins by the I/O control register. When not used for serial bus interface pins, the pin is used as a normal I/O port.

2.10.1 Configuration

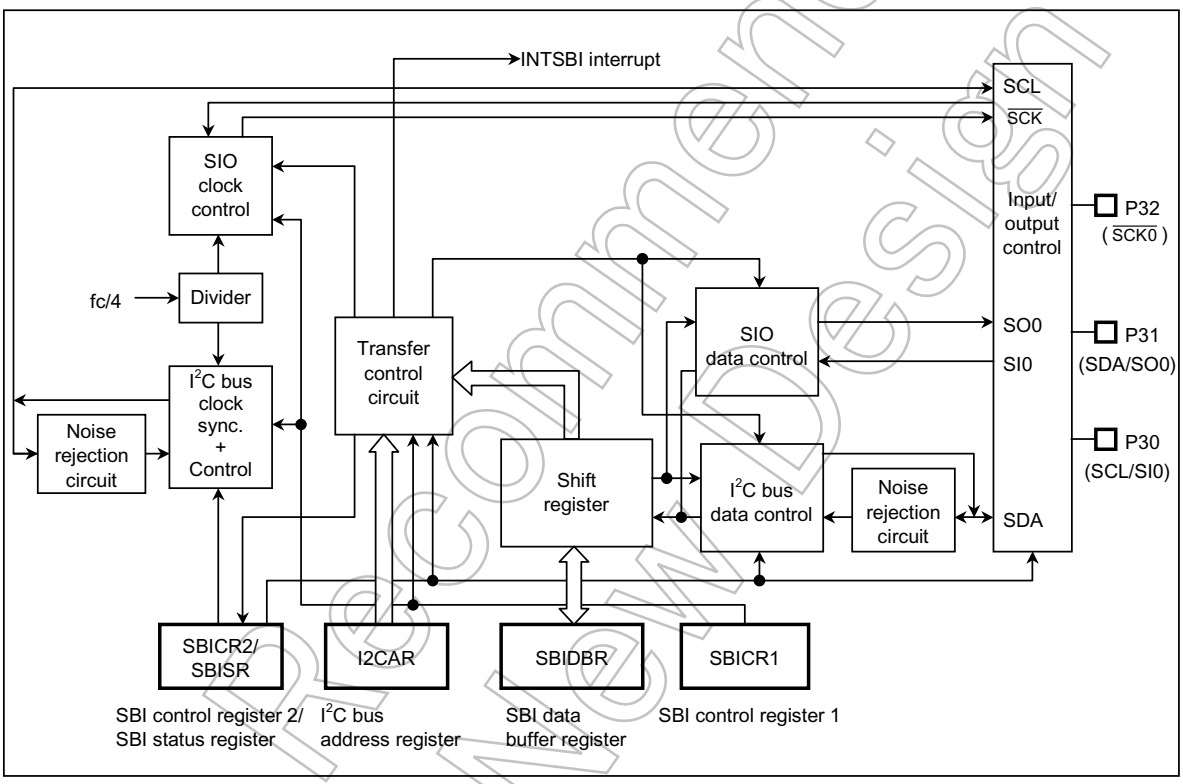


Figure 2.10.1 Serial Bus Interface (SBI-ver.C)

2.10.2 Control

The following registers are used to control the serial bus interface and monitor the operation status.

- Serial bus interface control register 1 (SBICR1)
- Serial bus interface control register 2 (SBICR2)
- Serial bus interface data buffer register (SBIDBR)
- I²C bus address register (I²CAR)
- Serial bus interface status register (SBISR)

The above registers differ depending on an mode to be used. Refer to Section "2.10.4 I²C bus mode control" and "2.10.6 Clocked-synchronous 8-bit SIO mode control".

2.10.3 The Data Format in the I²C Bus Mode

The data format in the I²C bus mode are shown in Figure 2.10.2.

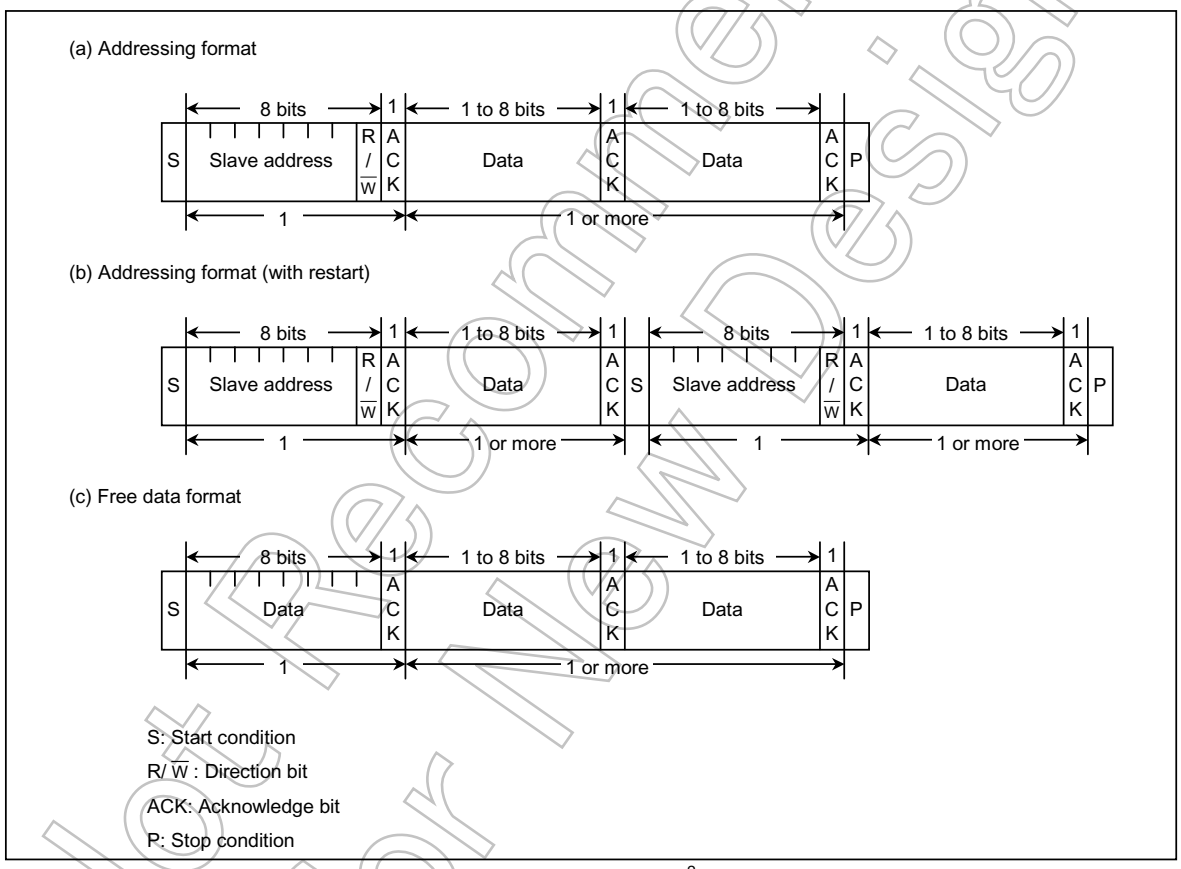


Figure 2.10.2 Data Format in I²C Bus Mode

2.10.4 I²C Bus Mode Control

The following registers are used to control the serial bus interface (SBI-ver.C) and monitor the operation status in the I²C bus mode.

Serial Bus Interface Control Register 1

SBICR1	7	6	5	4	3	2	1	0	
(0020H)	BC			ACK	SWRST	SCK			(Initial value: 0000 0000)

		BC	ACK = 0		ACK = 1		
			Number of Clock	Bits	Number of Clock	Bits	
BC	Number of transferred bits	000	8	8	9	8	Write only
		001	1	1	2	1	
		010	2	2	3	2	
		011	3	3	4	3	
		100	4	4	5	4	
		101	5	5	6	5	
		110	6	6	7	6	
		111	7	7	8	7	
ACK	Acknowledge mode specification	0: Acknowledge not returned to transmitter. 1: Acknowledge returned to transmitter.					Read/ Write
SWRST	Initiate a internal of SBI	0: - 1: Initialized (Clearing "0" after initialized)					Read/ Write
SCK	Serial clock selection	000:Reserved (Note4) 001:Reserved (Note4) 010: 91.9 kHz 011: 47.3 kHz 100: 24.0 kHz 101: 12.1 kHz 110: 6.08 kHz 111: Reserved	} at fc = 12.5 MHz (Output on SCL pin)				Write only

Note 1: fc: High-frequency clock [Hz]

Note 2: Set the BC to “000” before switching to 8-bit SIO bus mode.

Note 3: SBICR1 is write-only registers, which cannot be used with any of read-modify-write instruction such as bit manipulation, etc.

Note 4: This I²C bus circuit does not support the Fast mode. It supports the Standard mode only. Although the I²C bus circuit itself allows the setting of a baud rate over 100 kbps, the compliance with the I²C specification is not guaranteed in that case.

Serial Bus Interface Data Buffer Register

SBIDBR	7	6	5	4	3	2	1	0	
(0021H)									R/W

Note 1: For writing transmitted data, start from the MSB (bit 7).

Note 2: Cannot read the data which was written into SBIDBR, since a write data buffer and a read data buffer are independent in SBIDBR. Therefore, SBIDBR cannot be used with any of read-modify-write instructions such as bit manipulation, etc.

Note3: The data which was written into SBIDBR is cleared to “0” when INTSBI is generated.

I²C bus Address Register

I²CAR	7	6	5	4	3	2	1	0	
	Slave address							ALS	
(0022H)	SA6	SA5	SA4	SA3	SA2	SA1	SA0		(Initial value: 0000 0000)

SA	TMP88CU74 slave address selection		Write only
ALS	Address recognition mode specification	0: Slave address recognition 1: Non slave address recognition	

Note: I²CAR is write-only register, which cannot be used with any of read-modify-write instruction such as bit manipulation, etc.

Figure 2.10.3 Serial Bus Interface Control Register 1, Serial Bus Interface Data Buffer Register and I²C Bus Address Register in the I²C Bus Mode

Serial Bus Interface Control Register 2

SBICR2 (0023H)

7	6	5	4	3	2	1	0
MST	TRX	BB	PIN	SBIM		"0"	"0"

(Initial value: 0001 00**)

MST	Master/Slave selection	0: Slave 1: Master	Write only
TRX	Transmitter/Receiver selection	0: Receiver 1: Transmitter	
BB	Start/Stop generation	0: Stop condition 1: Start condition	
PIN	Cancel interrupt service request	0: — (cannot be cleared to "0") 1: Cancel interrupt service request	
SBIM	Serial bus interface operating mode selection	00: Port mode (serial bus interface output disable) 01: Clocked-synchronous 8-bit SIO mode 10: I^2C bus mode 11: Reserved	

Note 1: *: Don't care

Note 2: Switch a mode to port after confirming that the bus is free.

Note 3: Switch a mode to I2C bus mode after confiming that input signals via port are high level.

Note 4: SBICR2 has write-only register bits, which can not be used with any of read-modify-write instructions such as bit manipulation, etc.

Note 5: Clear bits 1 and 0 in SBICR2 to "0".

SBISR (0023H)

7	6	5	4	3	2	1	0
MST	TRX	BB	PIN	AL	AAS	AD0	LRB

(Initial value: 0001 0000)

MST	Master/Slave selection status monitor	0: Slave 1: Master	Read only
TRX	Transmitter/receiver selection status monitor	0: Receiver 1: Transmitter	
BB	Bus status monitor	0: Bus free 1: Bus busy	
PIN	Interrupt service request status monitor	0: INTSBI occurs 1: INTSBI not occurs	
AL	Arbitration loss detection monitor	0: Arbitration loss undetected 1: Arbitration loss detected	
AAS	Slave address match detection monitor	0: Slave address unmatch or "GENERAL CALL" undetected 1: Slave address match or "GENERAL CALL" detected	
AD0	"GENERAL CALL" detection monitor	0: "GENERAL CALL" undetected 1: "GENERAL CALL" detected	
LRB	Last received bit monitor	0: Last received bit "0" 1: Last received bit "1"	

Figure 2.10.4 Serial Bus Interface Control Register 2 and Serial Bus Interface Status Register in the I^2C Bus Mode

(1) Acknowledge mode specification

Set the ACK (bit 4 in SBICR1) to "1" for operation in the acknowledge mode. The TMP88CU74 generates an additional clock pulse for an acknowledge signal when operating in the master mode. In the transmitter mode during the clock pulse cycle, the SDA pin is released in order to receive the acknowledge signal from the receiver. In the receiver mode during the clock pulse cycle, the SDA pin is set to the low level in order to generate the acknowledge signal.

Reset the ACK for operation in the non-acknowledge mode. The TMP88CU74 does not generate a clock pulse for the acknowledge signal when operating in the master mode.

In the acknowledge mode, the TMP88CU74 counts a clock pulse for the acknowledge signal when operating in the slave mode. During the clock pulse, when the received slave address is the same as the value set at the I2CAR or when a GENERAL CALL is received, the SDA pin is set to the low level in order to generate the acknowledge signal.

In the transmitter mode during the clock pulse cycle after matching the slave addresses or receiving a GENERAL CALL, the SDA pin is released in order to receive the acknowledge signal from the receiver. In the receiver mode during the clock pulse cycle, the SDA pin is set to the low level in order to generate the acknowledge signal.

In non-acknowledge mode, the TMP88CU74 does not count a clock pulse for the acknowledge signal when operating in the slave mode.

(2) Number of transfer bits

The BC (bits 7 to 5 in SBICR1) is used to select a number of bits for transmitting and receiving data.

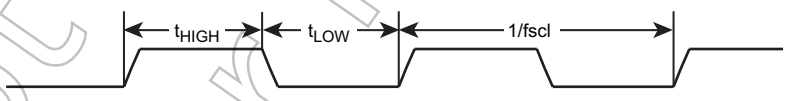
Since the BC is cleared to "000" as a start condition, a slave address and direction bit transmissions are always executed in 8 bits. Other than these, the BC retains a specified value.

(3) Serial clock

a. Clock source

The SCK (bits 2 to 0 in SBICR1) is used to select a maximum transfer frequency output from the SCL pin in the master mode. Set a communication baud rate that meets the I^2C bus specification, such as the shortest pulse width of tLOW, based on the equations shown below.

In both master mode and slave mode, a pulse width of at least 4 machine cycles is required for both high and low levels.



$t_{LOW} = 2^n/fc$

$t_{HIGH} = 2^n/fc + 8/fc$

$fscl = 1/(t_{LOW} + t_{HIGH})$

fc: High-frequency clock

SCK (Bits 2 to 0 in the SBICR1)	n
000	4
001	5
010	6
011	7
100	8
101	9
110	10

Figure 2.10.5 Clock Source

b. Clock synchronization

In the I²C bus mode, in order to drive a bus with a wired AND, a master device which pulls down a clock pulse to low will, in the first place, invalidate a clock pulse of another master device which generates a high-level clock pulse. The master device with a high-level clock pulse needs to detect the situation and implement the following procedure.

The TMP88CU74 has a clock synchronization function for normal data transfer even when more than one master exists on a bus.

The example explains clock synchronization procedures when two masters simultaneously exist on a bus.

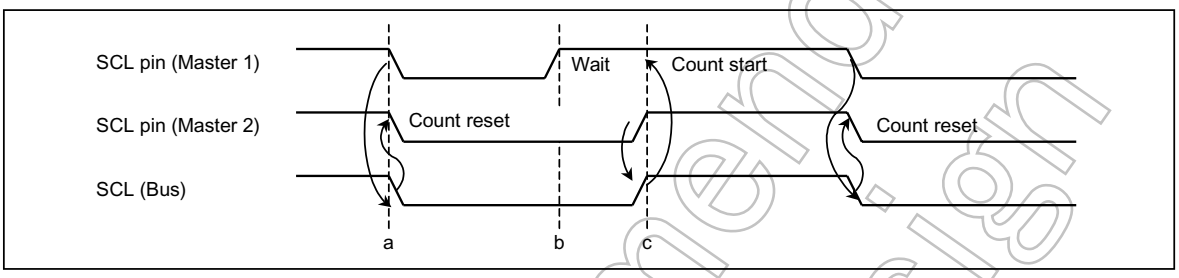


Figure 2.10.6 Clock Synchronization

As Master 1 pulls down the SCL pin to the low level at point "a", the SCL line of the bus becomes the low level. After detecting this situation, Master 2 resets counting a clock pulse in the high level and sets the SCL pin to the low level.

Master 1 finishes counting a clock pulse in the low level at point "b" and sets the SCL pin to the high level. Since Master 2 holds the SCL line of the bus at the low level, Master 1 waits for counting a clock pulse in the high level. After Master 2 sets a clock pulse to the high level at point "c" and detects the SCL line of the bus at the high level, Master 1 starts counting a clock pulse in the high level.

The clock pulse on the bus is deteminded by the master device with the shortest high-level period and the master device with the longest low-level period from among those master devices connected to the bus.

(4) Slave address and address recognition mode specification

When the serial bus interface circuit is used with an addressing format to recognize the slave address, clear the ALS (bit 0 in I²CAR) to "0", and set the SA (bits 7 to 1 in I²CAR) to the slave address.

When the serial bus interface circuit is used with a free data format not to recognize the slave address, set the ALS to "1". With a free data format, the slave address and the direction bit are not recognized, and they are processed as data from immediately after start condition.

(5) Master/slave selection

Set the MST (bit 7 in SBICR2) to "1" for operating the TMP88CU74 as a master device.

Reset the MST for operation as a slave device. The MST is cleared to "0" by the hardware after a stop condition on the bus is detected or arbitration is lost.

(6) Transmitter/Receiver selection

Set the TRX (bit 6 in SBICR2) to "1" for operating the TMP88CU74 as a transmitter. Reset the TRX for operation as a receiver. When data with an addressing format is transferred in the slave mode, the TRX is set to "1" if the direction bit (R/$\overline{W}$) sent from the master device is "1", and is cleared to "0" if the bit is "0". In the master mode, after an acknowledge signal is returned from the slave device with the hardware, the TRX is set to "0" if a transmitted direction bit is "1", and set to "1" if it is "0". When an acknowledge signal is not returned, the current condition is maintained.

The TRX is cleared to "0" by the hardware after a stop condition on the bus is detected or arbitration is lost.

The following table shows TRX changing conditions and TRX value after changing.

Mode	Direction Bit	Conditions	TRX after Changing
Slave mode	0	When the received slave address is the same as I^2CAR	0
	1		1
Master mode	0	When the ACK signal is returned	1
	1		0

When the serial bus interface circuit is used with a free data format, the TRX is not changed by hardware since the slave address and the direction bit are not recognized, and they are processed as data from immediately after start condition.

(7) Start/Stop condition generation

A start condition and 8-bit data are output on the bus by writing "1" to the MST, TRX and BB when the BB (bit 5 in SBICR2) is "0". It is necessary to set the transmitting data to the data buffer register and "1" to ACK beforehand.

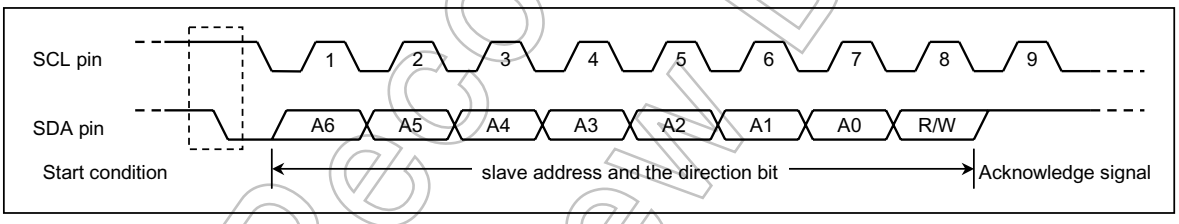


Figure 2.10.7 Start Condition Generation and Slave Address Generation

When the BB is "1", sequence of generating a stop condition is started by writing "1" to the MST, TRX, and PIN, and "0" to the BB. Do not modify the contents of MST, TRX, BB and PIN until a stop condition is generated on a bus.

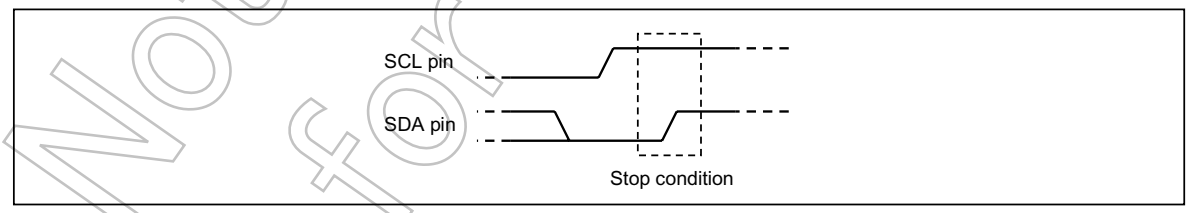


Figure 2.10.8 Stop Condition Generation

The bus condition can be indicated by reading the contetns of the BB (bit 5 in SBISR). The BB is set to "1" when a start condition on a bus is detected and is cleared to "0" when a stop condition is detected.

(8) Interrupt service request cancel

In the master mode, a serial bus interface interrupt request (INTSBI) occurs after the number of clocks which is specified by the BC and ACK has been transmitted.

In the slave mode, when the received slave address is the same as the value set at the I²CAR, after outputting the acknowledge signal when a GENERAL CALL is received, or when data transfer is complete after matching the slave addresses or receiving a GENERAL CALL, an INTSBI interrupt request occurs.

When a serial bus interface interrupt request occurs, the PIN (bit 4 in SBISR) is cleared to "0". During the time that the PIN is "0", the SCL pin is pulled down to the low level.

Either writing/reading data to/from the SBIDBR sets the PIN to "1".

The time from the PIN being set to "1" until the SCL pin is released takes t_{LOW}.

Although the PIN (bit 4 in SBICR2) can be set to "1" by the program, the PIN is not set to "0" when "0" is written.

(9) Serial bus interface operating mode

The SBIM (bits 3, 2 in SBICR2) is used to specify the serial bus interface operation mode. Set the SBIM to "10" after confirming that the serial bus interface pin is set to high level when used in the I²C bus mode.

Switch a mode to port after making sure that a bus is free.

(10) Arbitration lost detection monitor

Since more than one master device can exist simultaneously on a bus in the I²C bus mode, a bus arbitration procedure is implemented in order to guarantee the contents of transferred data.

Data on the SDA line is used for bus arbitration of the I²C bus.

The following shows an example of a bus arbitration procedure when two master devices exist simultaneously on the bus. Master 1 and Master 2 output the same data until point "a". After Master 1 outputs "1" and Master 2, "0", the SDA line of the bus is wired AND and the SDA line is pulled down to the low level by Master 2. When the SCL line of the bus is pulled up at point "b", the slave device reads data on the SDA line, that is, data in Master 2. Data transmitted from Master 1 becomes invalid. The state in Master 1 is called "arbitration lost". A master device which loses arbitration releases the SDA pin and the SCL pin in order not to effect data transmitted from other masters with arbitration. When more than one master sends the same data at the first word, arbitration occurs continuously after the second word.

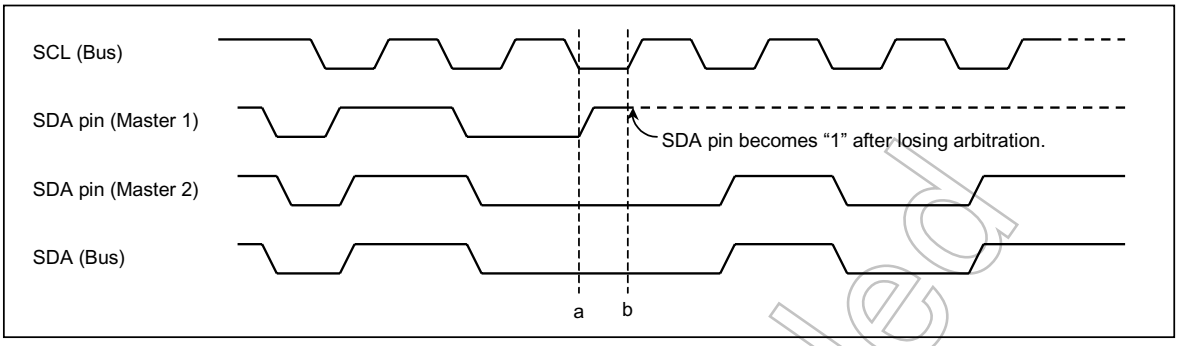


Figure 2.10.9 Arbitration Lost

The TMP88CU74 compares levels of the SDA line of the bus with those of the TMP88CU74 SDA pin at the rising edge of the SCL line. If the levels are unmatched, arbitration is lost and the AL (bit 3 in SBISR) is set to "1".

When the AL is set to "1", the MST and TRX are reset to "0" and the mode is switched to a slave receiver mode.

The AL is reset to "0" by writing/reading data to/from the SBIDBR or writing data to the SBICR2.

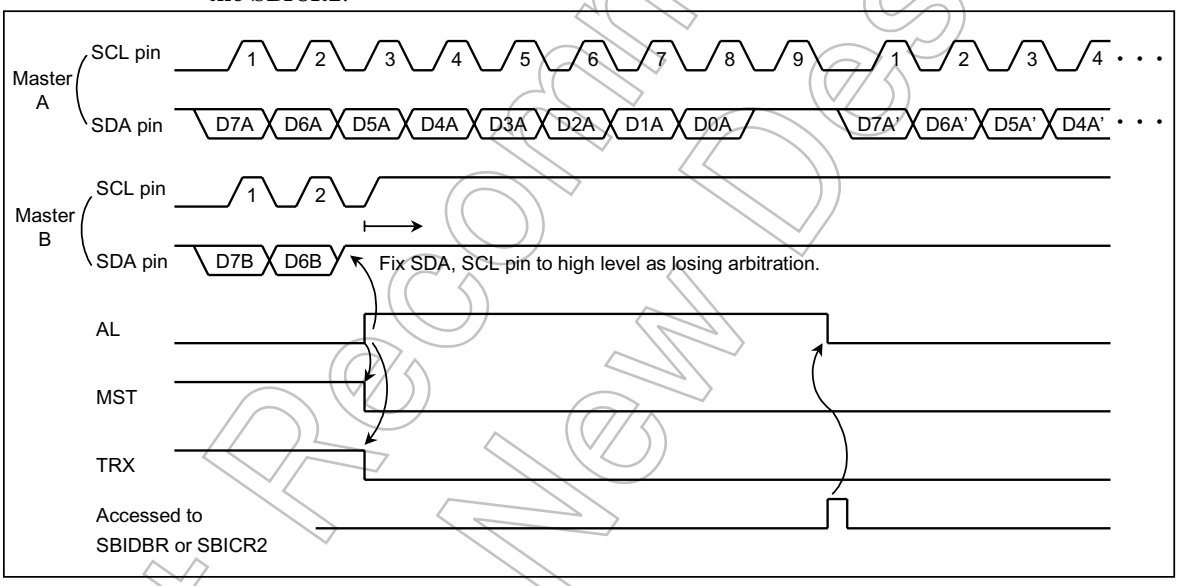


Figure 2.10.10 Example of when TMP88CU74 is a Master B

(11) Slave address match detection monitor

The AAS (bit 2 in SBISR) is set to "1" in the slave mode, in the address recognition mode (ALS = 0), or when receiving a slave address with the same value that sets a GENERAL CALL or I²CAR. When the ALS is "1", the AAS is set to "1" after receiving the first 1-word of data. The AAS is cleared to "0" by after writing/reading data to/from a data buffer register.

(12) GENERAL CALL detection monitor

The AD0 (bit 1 in SBISR) is set to "1" in the slave mode, when all 8-bit data received immediately after a start condition are "0". The AD0 is cleared to "0" when a start or stop condition is detected on the bus.

(13) Last received bit monitor

The SDA value stored at the rising edge of the SCL line is set to the LRB (bit 0 in SBISR). When the contents of the LRB are read immediately after an INTSBI interrupt request is generated in the acknowledge mode, and ACK signal is read.

(14) Software Reset Function

Software reset function is used to initialize SBI, when SBI is rocked by external noise, etc.

SWRST (bit 0 in SBICR) is set to "1", internal reset signal pulse is generated and inputted into SBI circuit.

All command registers and status registers are initialized to an initial value.

SWRST is automatically cleared to "0" after initializing SBI circuit.

2.10.5 Data Transfer in I^2C Bus Mode

(1) Device initialization

First, set the ACK in the SBICR1 to "1", the BC to "000", and the data length to 8-bit to count a clock pulse for the acknowledge signal. In addition, set the transmit frequency to the SCK.

Next, set the slave address to the SA in the I²CAR. Clear the ALS to "0" to set the addressing format.

After confirming that the serial bus interface pin is high level, for specifying the default setting to a slave receiver mode, clear "0" to the MST, TRX, and BB in the SBICR2; "1" to the PIN; "10" to the SBIM; and "0" to bits 1 and 0.

Note: To initialize the serial bus interface circuit, a constant period that the start conditions are not generated for any device is required after all devices which are connected to the bus are initialized. Then, the initialization must be completed during the period. If not, other devices may start transmitting data before the serial bus interface circuit has been initialized. Thus, data can not be normally received.

(2) Start condition and slave address generation

Confirm a bus free status (when BB = 0).

Set the ACK to "1" and specify a slave address and a direction bit to be transmitted to the SBIDBR.

When the BB is "0", the start condition are generated and the slave address and the direction bit which are set to the SBIDBR are output on a bus by writing "1" to the MST, TRX, BB, and PIN. An INTSBI interrupt request occurs at the 9th falling edge of the SCL clock cycle, and the PIN is cleared to "0". The SCL pin is pulled down to the low level while the PIN is "0". When an interrupt request occurs, the TRX changes by the hardware according to the direction bit only when an acknowledge signal is returned from the slave device.

Note 1: The slave address to be output to the SBIDBR must be set after the bus free is detected by software. If setting of the slave address is executed before detection bus free, the current output data may be corrupted.

Note 2: The bus free must be confirmed by software within 98.0 μs (the shortest transmitting time according to the I^2C bus standard) after setting of the slave address to be output. Only when the bus free is confirmed, set "1" to the MST, TRX, BB, and PIN to generate the start conditions. If the start conditions are generated without writing "1" to them, transferring may be executed by other masters between the time when the slave address to be output to the SBIDBR is written and the time when "1" is written to the MST, TRX, BB, and PIN in the SBICR2. Thus, the slave address may be corrupted.

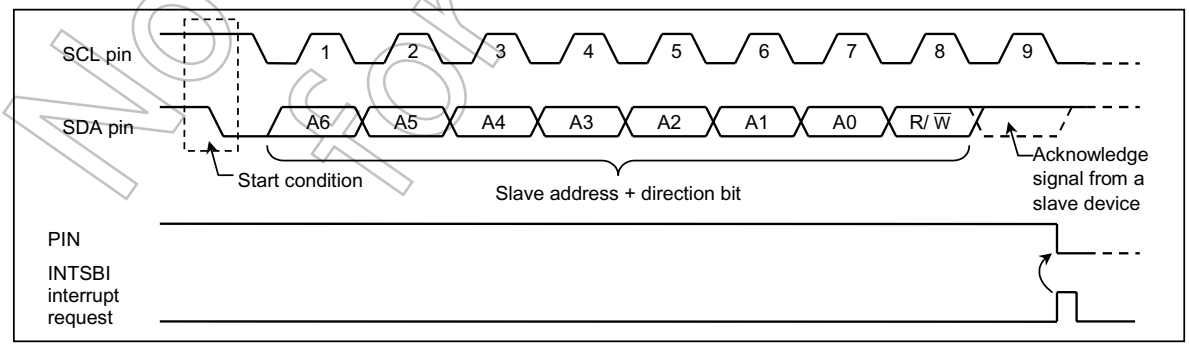


Figure 2.10.11 Start Condition Generation and Slave Address Transfer

(3) 1-word data transfer

Check the MST by the INTSBI interrupt process after a 1-word data transfer is completed, and determine whether the mode is a master or slave.

a. When the MST is "1" (Master mode)

Check the TRX and determine whether the mode is a transmitter or receiver.

1. When the TRX is "1" (Transmitter mode)

Check the LRB. When the LRB is "1", a receiver does not request data. Implement the process to generate a stop condition and terminate data transfer.

When the LRB is "0", the receiver requests new data. When the next transmitted data is other than 8 bits, set the BC and write the transmitted data to the SBIDBR after setting ACK to "1". After writing the data, the PIN becomes "1", a serial clock pulse is generated for transferring a new 1-word of data from the SCL pin, and then the 1-word data is transmitted. After the data is transmitted, an INTSBI interrupt request occurs. The PIN becomes "0" and the SCL pin is pulled down to the low level. If the data to be transferred is more than one word in length, repeat the procedure from the LRB checking above.

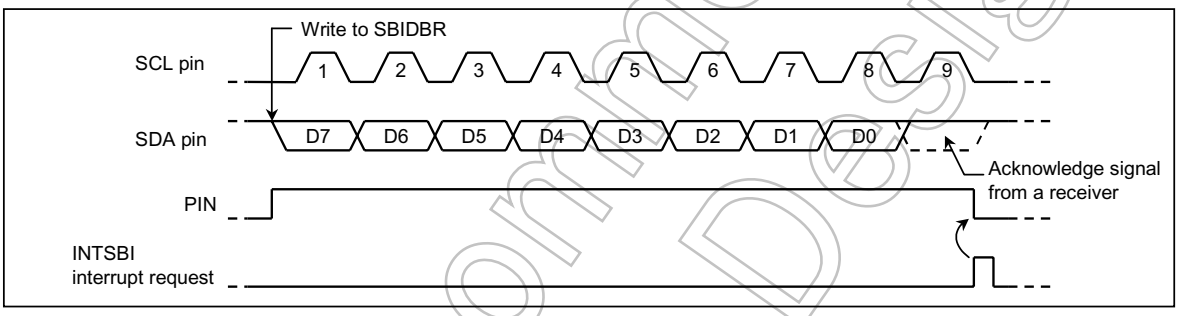


Figure 2.10.12 Example when BC = "000", ACK = "1"

2. When the TRX is "0" (Receiver mode)

When the next transmitted data is other than 8 bits, set the BC again. Set the ACK to "1" and read the received data from the SBIDBR (data which is read immediately after a slave address is sent is undefined). After the data is read, the PIN becomes "1". The TMP88CU74 outputs a serial clock pulse to the SCL to transfer new 1-word of data and sets the SDA pin to "0" at the acknowledge signal timing.

An INTSBI interrupt request occurs and the PIN becomes "0". The TMP88CU74 outputs a clock pulse for 1-word of data transfer and the acknowledge signal each time that received data is read from the SBIDBR.

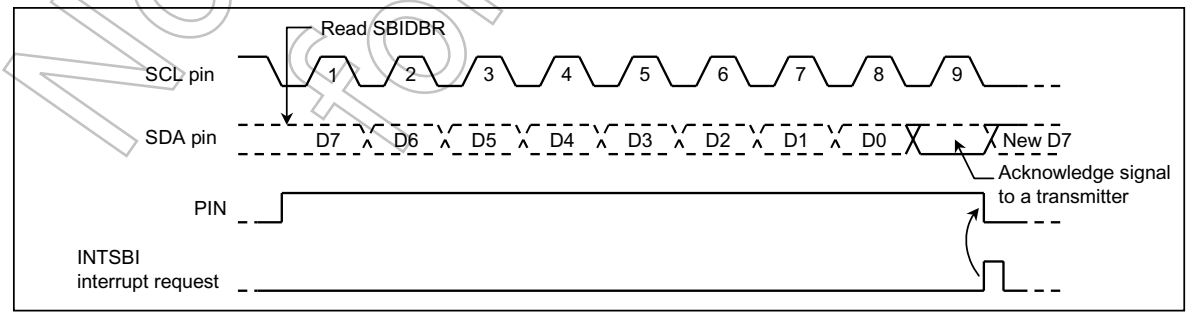


Figure 2.10.13 Example when BC = "000", ACK = "1"

In order to terminate transmitting data to a transmitter, clear the ACK to "0" before reading data which is 1 word before the last data to be received. The last data does not generate a clock pulse for the acknowledge signal. After the data is transmitted and an interrupt request has occurred, set the BC to "001" and read the data. The TMP88CU74 generates a clock pulse for a 1-bit data transfer. Since the master device is a receiver, the SDA line of the bus keeps the high level. The transmitter receives the high-level signal as an ACK signal. The receiver indicates to the transmitter that data transfer is complete.

After 1-bit data is received and an interrupt request has occurred, the TMP88CU74 generates a stop condition and terminates data transfer.

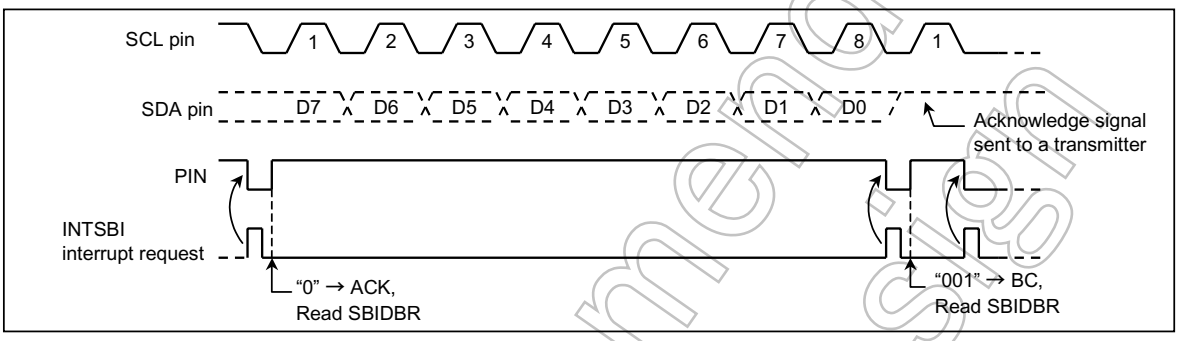


Figure 2.10.14 Termination of Data Transfer in Master Receiver Mode

b. When the MST is "0" (Slave mode)

In the slave mode, the TMP88CU74 operates either in normal slave mode or in slave mode after losing arbitration.

In the slave mode, an INTSBI interrupt request occurs when the TMP88CU74 receives a slave address or a GENERAL CALL from the master device, or when a GENERAL CALL is received and data transfer is complete after matching a received slave address. In the master mode, the TMP88CU74 operates in a slave mode if it is losing arbitration. An INTSBI interrupt request occurs when word data transfer terminates after losing arbitration. When an INTSBI interrupt request occurs, the PIN (bit 4 in the SBICR2) is reset, and the SCL pin is pulled down to the low level. Either reading/writing from/to the SBIDBR or setting the PIN to "1" releases the SCL pin after taking t_{LOW} time.

Check the AL (bit 3 in the SBISR), the TRX (bit 6 in the SBISR), the AAS (bit 2 in the SBISR), and the AD0 (bit 1 in the SBISR) and implements processes according to conditions listed in the next table.

Table 2.10.1 Operation in the Slave Mode

TRX	AL	AAS	AD0	Conditions	Process
1	1	1	0	The TMP88CU74 loses arbitration when transmitting a slave address and receives a slave address of which the value of the direction bit sent from another master is “1”.	Set the number of bits in 1 word to the BC and write transmitted data to the SBIDBR.
	0	1	0	In the slave receiver mode, the TMP88CU74 receives a slave address of which the value of the direction bit sent from the master is “1”.	
		0	0	In the slave transmitter mode, 1-word data is transmitted.	Check the LRB. If the LRB is set to “1”, set the PIN to “1” since the receiver does not request next data. Then, clear the TRX to “0” release the bus. If the LRB is cleared to “0”, set the number of bits in a word to the BC and write transmitted data to the SBIDBR since the receiver requests next data.
0	1	0	1/0	The TMP88CU74 loses arbitration when transmitting a slave address and receives a slave address or GENERAL CALL of which the value of the direction bit sent from another master is “0”.	Read the SBIDBR for setting the PIN to “1” (reading dummy data) or write “1” to the PIN.
		0	0	The TMP88CU74 loses arbitration when transmitting a slave address or data and terminates transferring word data.	
	0	1	1/0	In the slave receiver mode, the TMP88CU74 receives a slave address or GENERAL CALL of which the value of the direction bit sent from the master is “0”.	
		0	1/0	In the slave receiver mode, the TMP88CU74 terminates receiving of 1-word data.	Set the number of bits in a word to the BC and read received data from the SBIDBR.

(4) Stop condition generation

When the BB is “1”, a sequence of generating a stop condition is started by writing “1” to the MST, TRX, and PIN, and “0” to the BB. Do not modify the contents of the MST, TRX, BB, PIN until a stop condition is generated on a bus.

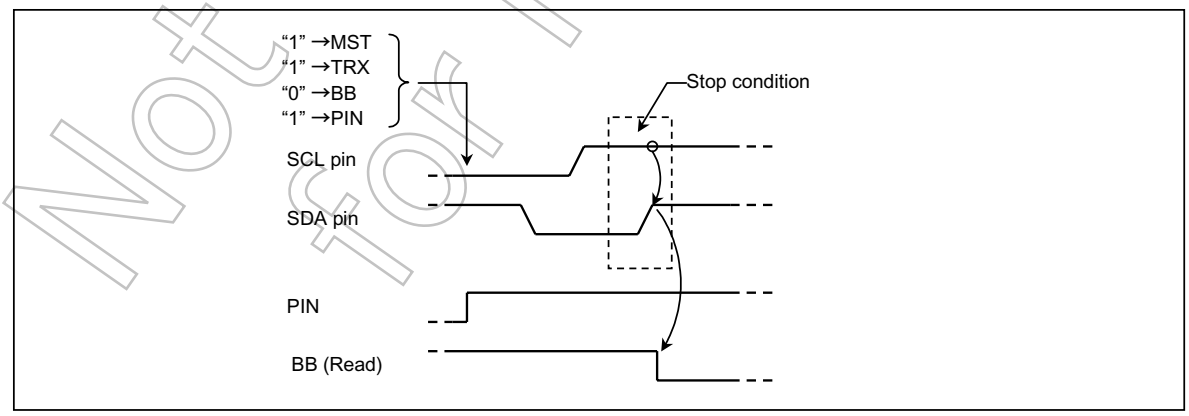


Figure 2.10.15 Stop Condition Generation

(5) Restart

Restart is used to change the direction of data transfer between a master device and a slave device during transferring data. The following explains how to restart when the TMP88CU74 is in the master mode.

Specify "0" to the MST, TRX, and BB and "1" to the PIN and release the bus. The SDA pin retains the high level and the SCL pin is released. Since a stop condition is not generated on a bus, a bus is assumed to be in a busy state from other devices. Check the BB until it becomes "0" to check that the SCL pin of theTMP88CU74 is released. Check the LRB until it becomes "1" to check that the SCL line of a bus is not pulled down to the low level by other devices. After confirming that a bus stays in a free state, generate a start condition with procedure (2).

In order to meet setup time when restarting, take at least 4.7 [μs] of waiting time by software from the time of restarting to confirm that the bus is free until the time to generate the start condition.

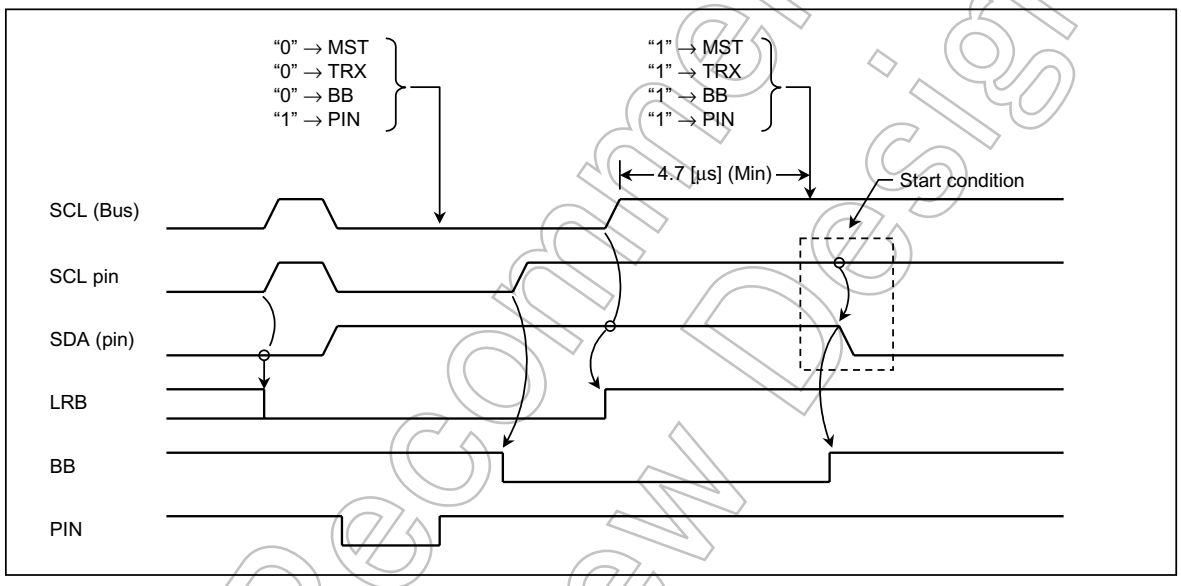


Figure 2.10.16 Timing Diagram when Restarting theTMP88CU74

2.10.6 Clocked-synchronous 8-Bit SIO Mode Control

The following registers are used to control the serial bus interface (SBI-ver.C) and monitor the operation status in the clocked-synchronous 8-bit SIO mode.

Serial Bus Interface Control Register 1

SBICR1 (00020H)

7	6	5	4	3	2	1	0
SIOS	SIO INH	SIOM		"0"	SCK		

(Initial value: 0000 *000)

SIOS	Indicate transfer start/stop	0: Stop 1: Start	Write only
SIOINH	Continue/Abort transfer	0: Continue transfer 1: Abort transfer (automatically cleared after abort)	
SIOM	Transfer mode select	00: 8-bit transmit mode 01: Reserved 10: 8-bit transmit/receive mode 11: 8-bit receive mode	
SCK	Serial clock select	000: fc/2^5 (390 kHz) 001: fc/2^6 (195 kHz) 010: fc/2^7 (97.6 kHz) 011: fc/2^8 (48.8 kHz) 100: fc/2^9 (24.4 kHz) 101: fc/2^{10} (12.2 kHz) 110: fc/2^{11} (6.1 kHz) } at fc = 12.5 MHz (Output on $\overline{SCK}$ pin) 111: External clock (input from $\overline{SCK}$ pin)	

Note 1: *: Don't care

Note 2: Set the SIOS to "0" when setting the transfer mode or serial clock.

Note 3: SBICR1 is write-only register, which cannot be used with any of read-modify-write instruction such as bit manipulation, etc.

Serial Bus Interface Data Buffer Register

SBIDBR (00021H)

7	6	5	4	3	2	1	0

R/W

Note: Cannot read the data which was written into SBIDBR, since a write data buffer and a read buffer are independent in SBIDBR. Therefore, SBIDBR cannot be used with any in read-modify-write instruction such as bit manipulation, etc.

Serial Bus Interface Control Register 2

SBICR2 (00023H)

7	6	5	4	3	2	1	0
"0"	"0"	"0"	"1"	SBIM		"0"	"0"

(Initial value: **** 00**)

SBIM	Serial bus interface operation mode selection	00: Port mode (serial bus interface output disable) 01: Clocked-synchronous 8-bit SIO mode 10: I^2C bus mode 11: Reserved	Write only

Note 1: *: Don't care

Note 2: Switch a mode to port after data transfer is complete.

Note 3: Switch a mode to I2C bus mode or clocked-synchronous 8-bit SIO mode after confirming that input signal via port is high level.

Note 4: SBICR2 is write-only register, which cannot be used with any of read-modify-write instruction such as bit manipulation, etc.

Note 5: Clear bits 7 to 5 and bits 1 to 0 in SBICR2 to "0", and set bit 4 to "1".

Serial Bus Interface Status Register

SBISR (00023H)

7	6	5	4	3	2	1	0
"1"	"1"	"1"	"1"	SIOF	SEF	"1"	"1"

SIOF	Serial transfer operating status monitor	0 : Transfer terminated 1 : Transfer in process	Read only
SEF	Shift operating status monitor	0 : Shift operation terminated 1 : Shift operation in process	

Figure 2.10.17 Control Register/Data Buffer Register/Status Register in SIO Mode

(1) Serial clock

a. Clock source

The SCK(bits 2 to 0 in SBICR1) is used to select the following functions.

1. Internal clock

In an internal clock mode, any of seven frequencies can be selected. The serial clock is output to the outside on the $\overline{SCK0}$ pin. The $\overline{SCK0}$ pin becomes a high level when data transfer starts. When writing (in the transmit mode) or reading (in the receive mode) data cannot follow the serial clock rate, an automatic-wait function is executed to stop the serial clock automatically and hold the next shift operation until reading or writing is complete.

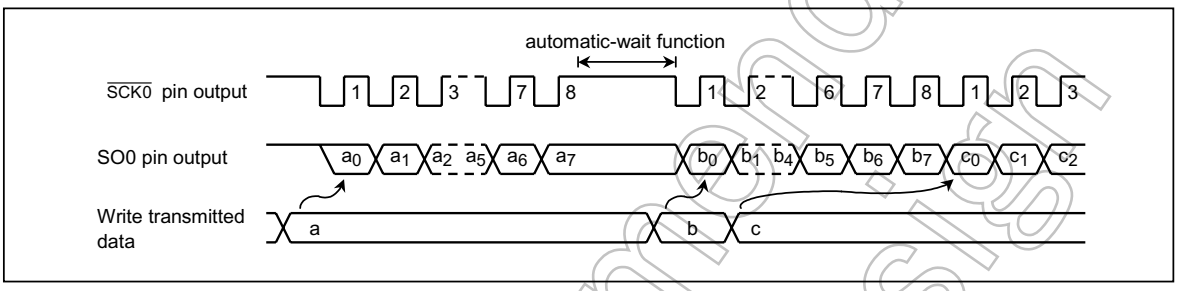


Figure 2.10.18 Automatic Wait Function

2. External clock (SCK = "111")

An external clock supplied to the $\overline{SCK0}$ pin is used as the serial clock. In order to ensure shift operation, a pulse width of at least 4 machine cycles is required for both high and low levels in the serial clock. The maximum data transfer frequency is 390 kHz (fc = 12.5 MHz).

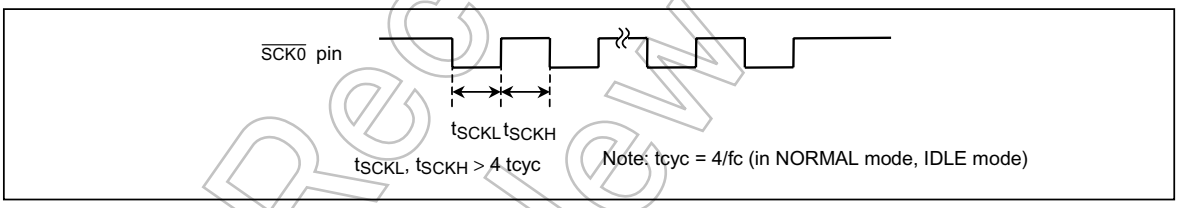


Figure 2.10.19 The Maximum Data Transfer Frequency in the External Clock Input

b. Shift edge

The leading edge is used to transmit data, and the trailing edge is used to receive data.

1. Leading edge

Data is shifted on the leading edge of the serial clock (at a falling edge of the $\overline{SCK0}$ pin input/output).

2. Trailing edge

Data is shifted on the trailing edge of the serial clock (at a rising edge of the $\overline{SCK0}$ pin input/output).

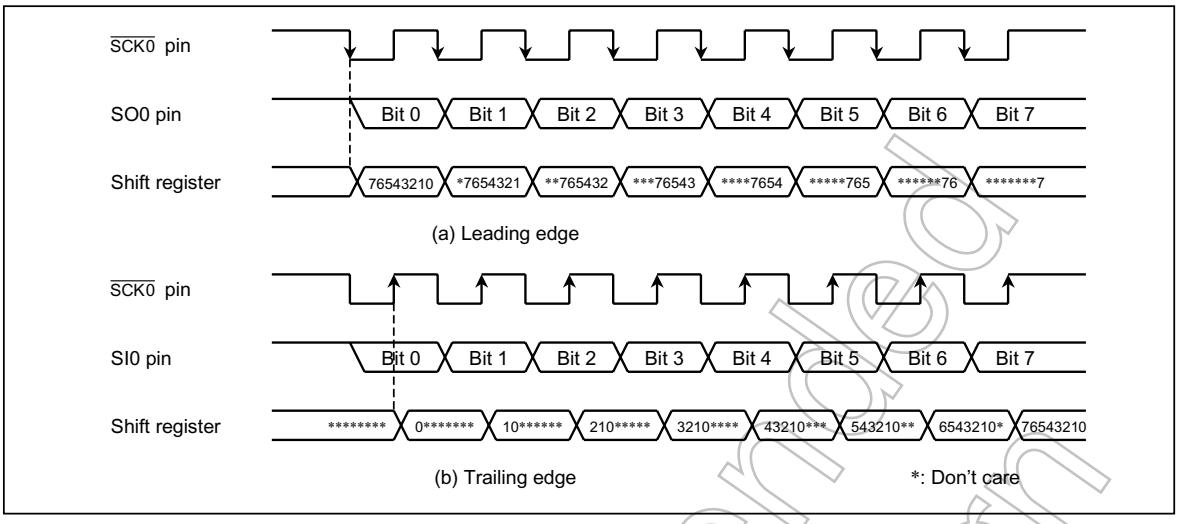


Figure 2.10.20 Shift Edge

(2) Transfer mode

The SIOM (bits 5 and 4 in SIO1CR) is used to select a transmit, receive, or transmit/receive mode.

a. 8-bit transmit mode

Set a control register to a transmit mode and write data to the SBIDBR.

After the data is written, set the SIOS to "1" to start data transfer. The transmitted data is transferred from the SBIDBR to the shift register and output to the SO0 pin in synchronous with the serial clock, starting from the least significant bit (LSB). When the data is transferred to the shift register, the SBIDBR becomes empty. The INTSBI (buffer empty) interrupt request is generated to request new data.

When the internal clock is used, the serial clock will stop and automatic-wait function will be initiated if new data is not loaded to the data buffer register after the specified 8-bit data is transmitted. When new data is written, automatic-wait function is canceled.

When the external clock is used, data should be written to the SBIDBR before new data is shifted. The transfer speed is determined by the maximum delay time between the time when an interrupt request is generated and the time when data is written to the SBIDBR by the interrupt service program.

When the transmit is started, the same value as the final bit of the last data is output until the falling edge of the $\overline{SCK}$ after the SIOF goes "1".

Transmitting data is ended by cleaning the SIOS to "0" by the buffer empty interrupt service program or setting the SIOINH to "1". When the SIOS is cleared, the transmitted mode ends when all data is output. In order to confirm if data is surely transmitted by the program, set the SIOF (bit 3 in the SBISR) to be sensed. The SIOF is cleared to "0" when transmitting is complete. When the SIOINH is set, transmitting data stops. The SIOF turns "0".

When the external clock is used, it is also necessary to clear the SIOS to "0" before new data is shifted; otherwise, dummy data is transmitted and operation ends.

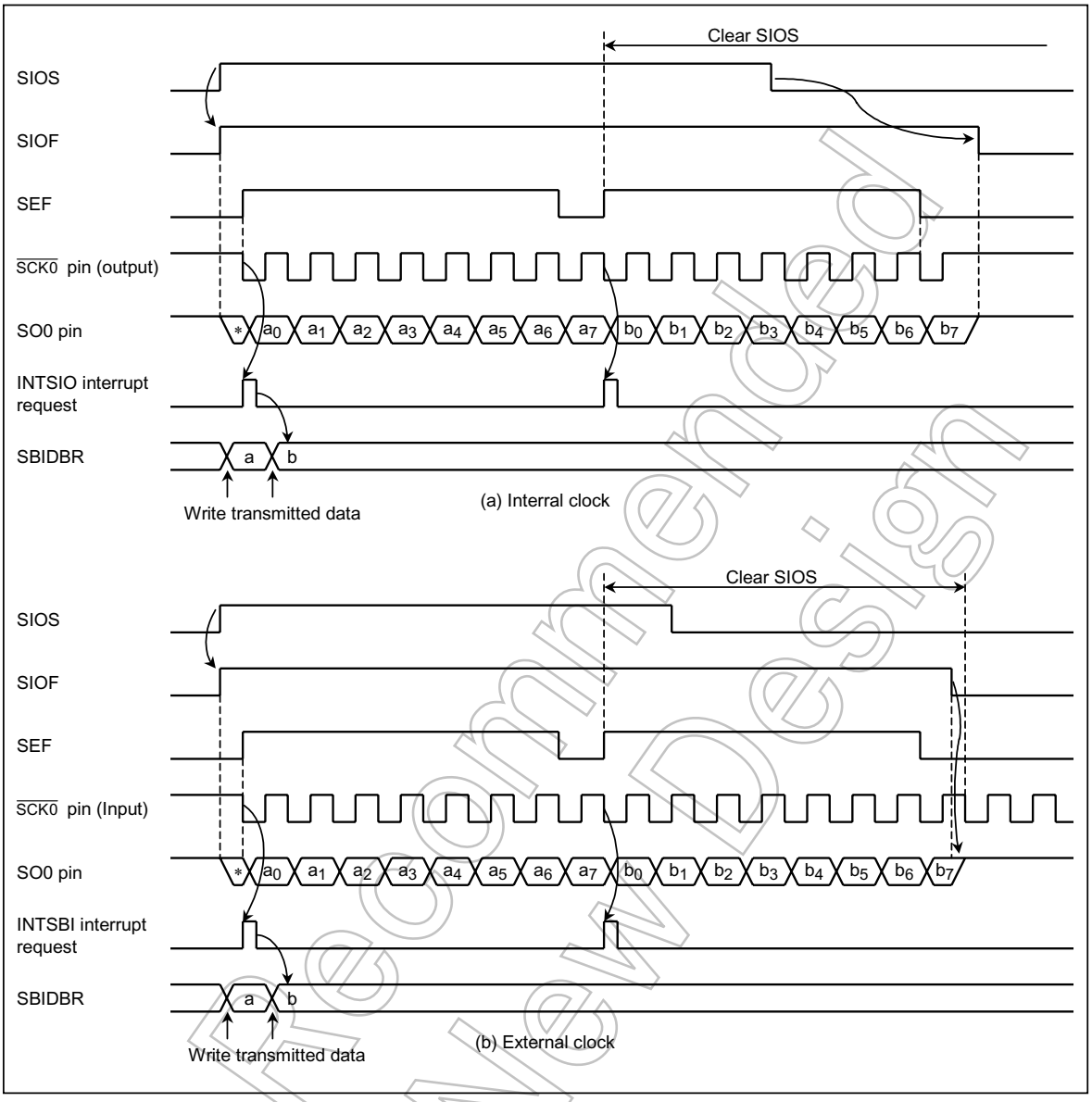


Figure 2.10.21 Transfer Mode

```
Example:  SIO0 transfer end command (External clock)
STEST1:   TEST   (SBISR). SEF        ;  If SEF = 1 then loop
          JRS    F, STEST1
STEST2:   TEST   (P3). 6             ;  If SCK = 0 then loop
          JRS    T, STEST2
          LD     (SBICR1), 00000111B ;  SIOS ← 0
```

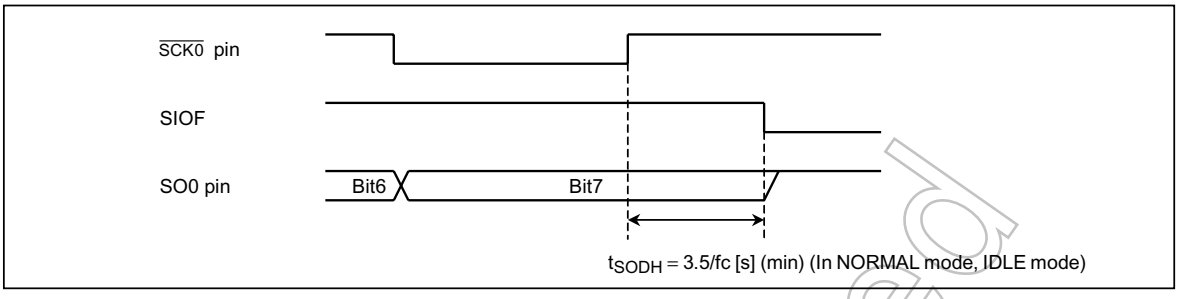


Figure 2.10.22 Transmitted Data Hold Time at End of Transmit

b. 8-bit receive mode

Set a control register to a receive mode and the SIOS to "1" for switching to a receive mode. Data is received from the SI0 pin to the shift register in synchronous with the serial clock, starting from the least significant bit (LSB). When the 8-bit data is received, the data is transferred from the shift register to the SBIDBR. The INTSBI (buffer full) interrupt request is generated to request of reading the received data. The data is then read from the SBIDBR by the interrupt service program.

When the internal clock is used, the serial clock will stop and automatic-wait function will be initiated until the received data is read from the SBIDBR.

When the external clock is used, since shift operation is synchronized with the clock pulse provided externally, the received data should be read before new data is transferred to the SBIDBR. If the received data is not read, further data to be received is canceled. The maximum transfer speed when the external clock is used is determined by the delay time between the time when an interrupt request is generated and the time when received data is read.

Receiving data is ended by clearing the SIOS to "0" by the buffer full interrupt service program or setting the SIOINH to "1". When the SIOS is cleared, received data is transferred to the SBIDBR in complete blocks. The received mode ends when the transfer is complete. In order to confirm if data is surely received by the program, set the SIOF (bit 3 in SBIDBR) to be sensed. The SIOF is cleared to "0" when receiving is complete. After confirming that receiving has ended, the last data is read. When the SIOINH is set, receiving data stops. The SIOF turns "0" (the received data becomes invalid, therefore no need to read it).

Note: When the transfer mode is switched, the SBIDBR contents are lost. In case that the mode needs to be switched, conclude receiving data by clearing the SIOS to "0", read the last data, and then switch the mode.

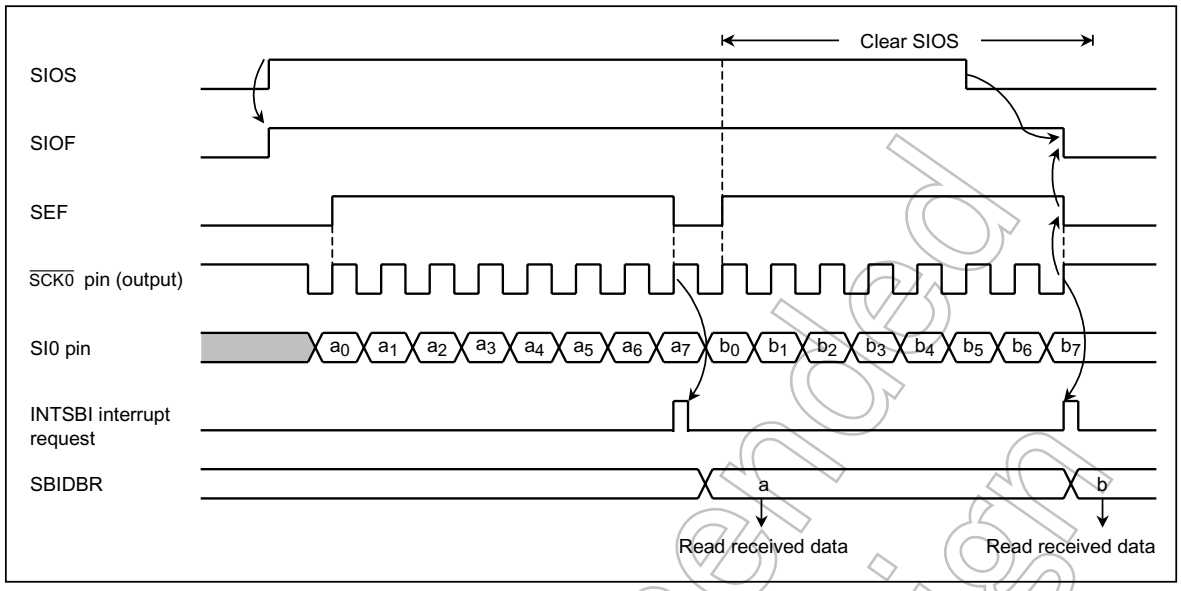


Figure 2.10.23 Receive Mode (Example: Internal Clock)

c. 8-bit transmit/receive mode

Set a control register to a transmit/receive mode and write data to the SBIDBR. After the data is written, set the SIOS to "1" to start transmitting/receiving. When transmitting, the data is output from the SO0 pin on the leading edges in synchronous with the serial clock, starting from the least significant bit (LSB). When receiving, the data is input to the SI0 pin on the trailing edges of the serial clock. 8-bit data is transferred from the shift register to the SBIDBR, and the INTSBI interrupt request occurs. The interrupt service program reads the received data from the data buffer register and writes data to be transmitted. The SBIDBR is used for both transmitting and receiving. Transmitted data should always be written after received data is read.

When the internal clock is used, automatic-wait function is initiated until received data is read and next data is written.

When the external clock is used, since the shift operation is synchronized with the serial clock provided externally, received data is read and transmitted data is written before new shift operation is executed. The maximum transfer speed when the external clock is used is determined by the delay time between the time when an interrupt request is generated and the time when received data is read and transmitted data is written.

When the transmit is started, the same value as the final bit of the last data is output until the falling edge of the $\overline{SCK}$ after the SIOF goes "1".

Transmitting/receiving data is ended by cleaning the SIOS to "0" by the INTSBI interrupt service program or setting the SIOINH to "1". When the SIOS is cleared, received data is transferred to the SBIDBR in complete blocks. The transmit/receive mode ends when the transfer is complete. In order to confirm if data is surely transmitted/received by the program, set the SIOF (bit 3 in SBISR) to be sensed. The SIOF becomes "0" after transmitting/receiving is complete. When the SIOINH is set, transmitting/receiving data stops. The SIOF turns "0".

Note: When the transfer mode is switched, the SBIDBR contents are lost. In case that the mode needs to be switched, conclude transmitting/receiving data by clearing the SIOS to "0", read the last data, and then switch the transfer mode.

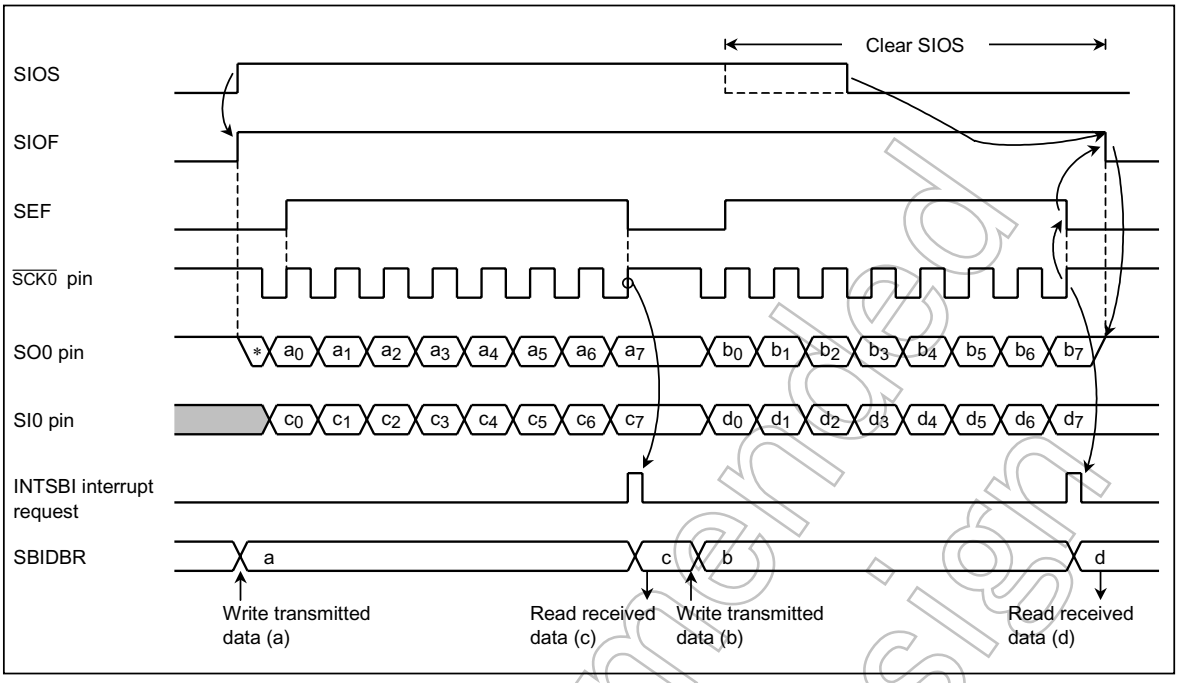


Figure 2.10.24 Transmit/Receive Mode (Example: Internal Clock)

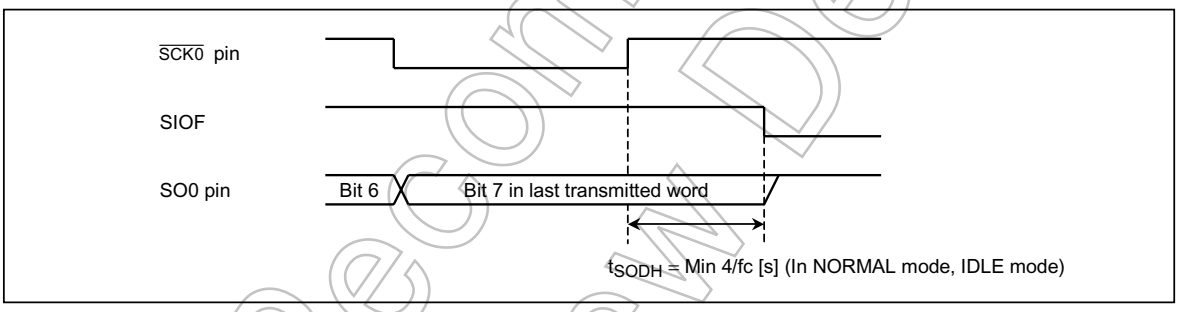


Figure 2.10.25 Transmitted Data Hold Time at End of Transmit/Receive

2.11 Serial Interface (SIO1)

The TMP88CU74 has clocked-synchronous 8-bit serial interfaces (SIO1). The serial interface has an 8-byte transmit and receive data buffer that can automatically and continuously transfer up to 64 bits of data.

The serial interface is connected to external devices via pins P02 (SO1), P01 (SI1), P00 ($\overline{SCK1}$) for SIO1. The serial interface pins are also used as port P0. When used as serial interface pins, the output latches of these pins should be set to "1" and set to input mode by P0CR. In the transmit mode, pins P01 can be used as normal I/O ports, and in the receive mode, the pins P02 can be used as normal I/O ports.

2.11.1 Configuration

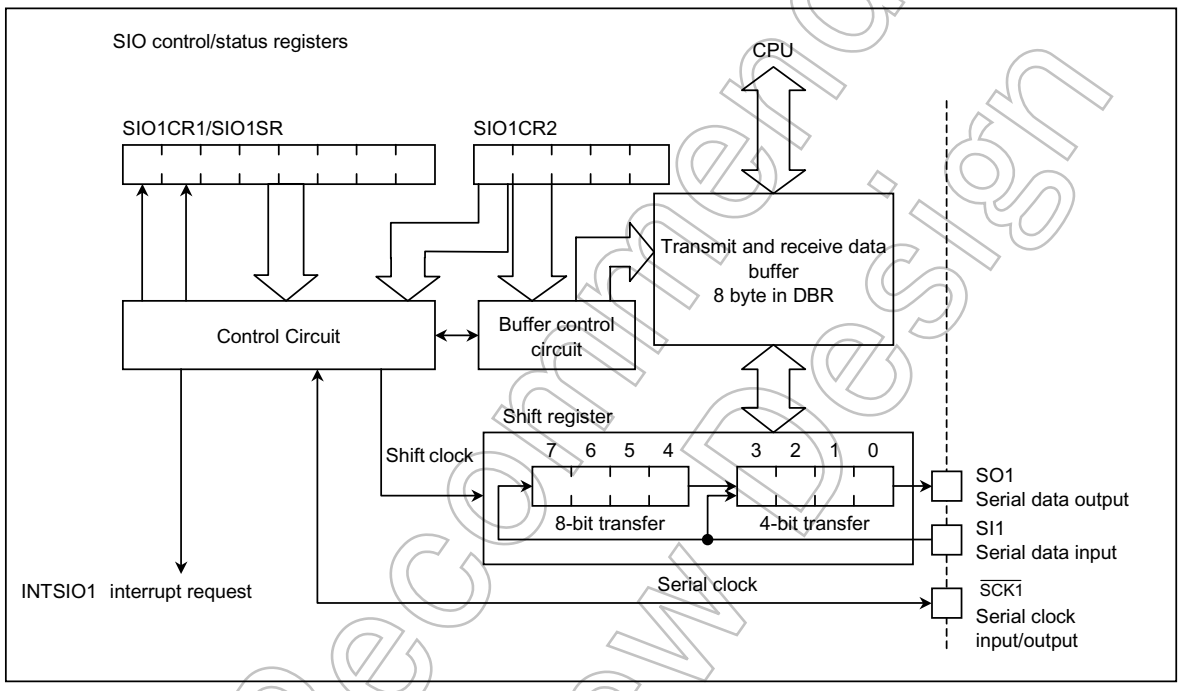


Figure 2.11.1 Serial Interface

2.11.2 Control

The serial interface is controlled by SIO1 control registers (SIO1CR1/SIO1CR2). The serial interface status can be determined by reading SIO1 status register (SIO1SR).

The transmit and receive data buffer is controlled by the BUF (bits 2 to 0 in SIO1CR2). The data buffer is assigned to addresses 0FF8H to 0FFFH for SIO1 in the DBR area, and can continuously transfer up to 8 words (bytes or nibbles) at one time. When the specified number of words has been transferred, a buffer empty (in the transmit mode) or a buffer full (in the receive mode or transmit/receive mode) interrupt (INTSIO1) is generated.

When the internal clock is used as the serial clock in the 8-bit receive mode and the 8-bit transmit/receive mode, a fixed interval wait can be applied to the serial clock for each word transferred. Four different wait times can be selected with WAIT (bits 4 and 3 in SIO1CR2).

SIO Interface 1 Control Register 1

SIO1CR1 (00027H)

7	6	5	4	3	2	1	0
SIOS	SIO INH	SIOM			SCK		

(Initial value: 0000 0000)

SIOS	Indicate transfer start/stop	0: Stop 1: Start							Write only
SIOINH	Continue/Abort transfer	0: Continue transfer 1: Abort transfer (automatically cleared after abort)							
SIOM	Transfer mode select	000: 8-bit transmit mode 010: 4-bit transmit mode 100: 8-bit transmit/receive mode 101: 8-bit receive mode 110: 4-bit receive mode							
SCK	Serial clock select Output on $\overline{SCK}$ 1 pin		NORMAL1/2, IDLE1/2 mode				SLOW, SLEEP mode		
			DV7CK = 0		DV7CK = 1				
			DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1			
		000	$fc/2^{13}$	$fc/2^{14}$	$fs/2^5$	$fs/2^5$	$fs/2^5$		
		001	$fc/2^8$	$fc/2^9$	$fc/2^8$	$fc/2^9$	—		
		010	$fc/2^6$	$fc/2^7$	$fc/2^6$	$fc/2^7$	—		
		011	$fc/2^5$	$fc/2^6$	$fc/2^5$	$fc/2^6$	—		
		111	External clock (input from $\overline{SCK}$ pin)						

Note 1: Set SIOS to “0” and SIOINH to “1” when setting the transfer mode or serial clock.

Note 2: SIO1CR1 is write-only registers, which cannot access any of in read-modify-write instruction such as bit operate, etc.

SIO Interface 1 Control Register 2

SIO1CR2 (00028H)

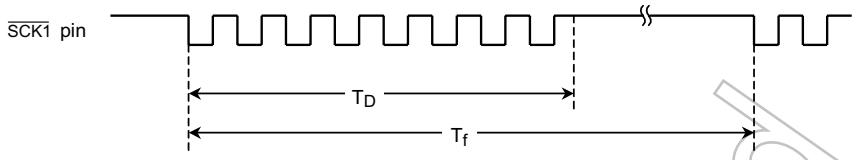
7	6	5	4	3	2	1	0
			WAIT		BUF		

(Initial value: ***0 0000)

WAIT	Wait control	Always “0” (Except 8 bit transmit/receive modes) 00: $T_f = T_D$ (Non-wait) 01: $T_f = 2T_D$ 10: $T_f = 4T_D$ (Wait) 11: $T_f = 8T_D$		Write only
BUF	Number of transfer words		Buffer address used SIO1	
		000: 1 word transfer	00FF8H	
		001: 2 words transfer	00FF8 - 00FF9H	
		010: 3 words transfer	00FF8 - 00FFAH	
		011: 4 words transfer	00FF8 - 00FFBH	
		100: 5 words transfer	00FF8 - 00FFCH	
		101: 6 words transfer	00FF8 - 00FFDH	
		110: 7 words transfer	00FF8 - 00FFEH	
		111: 8 words transfer	00FF8 - 00FFFH	

Figure 2.11.2 SIO Interface 1 Control Registers (1/2)

Note 1: T_f: frame time, T_D: data transfer time



Note 2: The lower 4 bits of each buffer are used during 4-bit transfers. Zeros (0) are stored to the upper 4 bits when receiving.

Note 3: Transmitting starts at the lowest address. Received data are also stored starting from the lowest address to the highest address. For example, in the case of SIO, the first buffer address transmitted is 00FF8H.

Note 4: The value to be loaded to BUF is held after transfer is completed.

Note 5: Set the SIOS to "0" when setting the transfer mode or serial clock.

Note 6: *: Don't care

Note 7: SIO1CR2 is write-only register, which cannot access any of in read-modify-write instruction such as bit operate, etc.

Figure 2.11.3 SIO Interface 1 Control Registers (2/2)

SIO1SR (00027H)

7	6	5	4	3	2	1	0
SIOF	SEF	"1"	"1"	"1"	"1"	"1"	"1"

SIOF	Serial transfer operating status monitor	0: Transfer terminated 1: Transfer in process	Read only
SEF	Shift operating status monitor	0: Shift operation terminated 1: Shift operation in process	

Figure 2.11.4 SIO Interface 1 Status Register

(1) Serial clock

a. Clock source

SCK (bits 2 to 0 in SIO1CR1) is able to select the following:

1. Internal clock

Any of four frequencies can be selected. The serial clock is output to the outside on the $\overline{SCK1}$ pin. The $\overline{SCK1}$ pin goes high when transfer starts.

When data writing (in the transmit mode) or reading (in the receive mode or the transmit/receive mode) cannot keep up with the serial clock rate, there is a wait function that automatically stops the serial clock and holds the next shift operation until the read/write processing is completed.

Table 2.11.1 Serial Clock Rate

Serial clock				SLOW, SLEEP mode	Maximum transfer rate	
DV7CK = 0		DV7CK = 1			fc = 12.5 MHz	fs = 32.768 KHz
DV1CK = 0	DV1CK = 1	DV1CK = 0	DV1CK = 1			
$fc/2^{13}$ [Hz]	$fc/2^{14}$ [Hz]	$fs/2^5$ [Hz]	$fs/2^5$ [Hz]	$fs/2^5$ [Hz]	1.50 Kbit/s	1 Kbit/s
$fc/2^8$	$fc/2^9$	$fc/2^8$	$fc/2^9$	—	48.8	—
$fc/2^6$	$fc/2^7$	$fc/2^6$	$fc/2^7$	—	185	—
$fc/2^5$	$fc/2^6$	$fc/2^5$	$fc/2^6$	—	390	—

Note: 1 Kbit = 1024 bits

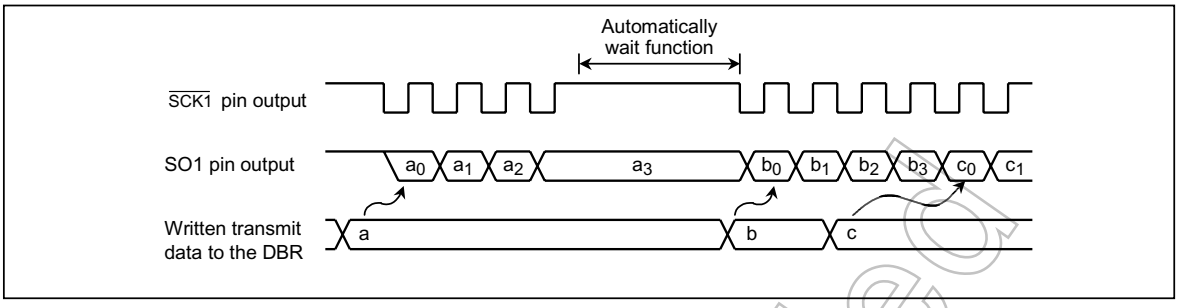
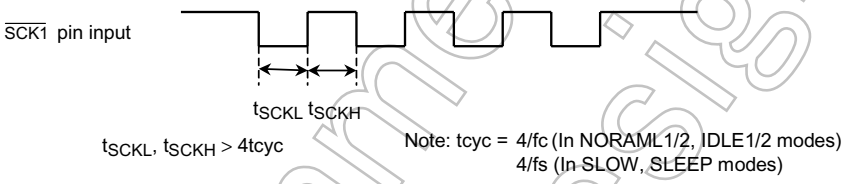


Figure 2.11.5 Clock Source (Internal Clock)

2. External clock

An external clock connected to the $\overline{SCK1}$ pin is used as the serial clock. In this case, the P00 ($\overline{SCK1}$) output latch must be set to "1". To ensure shifting, a pulse width of at least 4 machine cycles is required. Thus, the maximum transfer speed is 390 Kbit/s. (at fc = 12.5 MHz).



b. Shift edge

The leading edge is used to transmit, and the trailing edge is used to receive.

1. Leading edge

Transmitted data are shifted on the leading edge of the serial clock (falling edge of the $\overline{SCK1}$ pin input/output).

2. Trailing edge

Received data are shifted on the trailing edge of the serial clock (rising edge of the $\overline{SCK1}$ pin input/output).

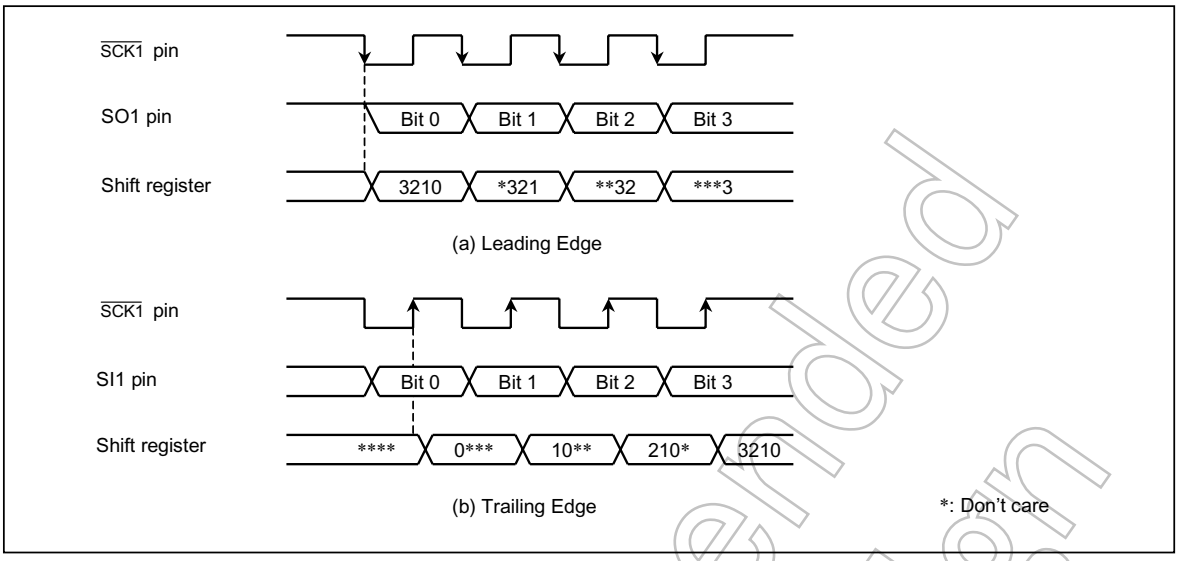


Figure 2.11.6 Shift Edge

(2) Number of bits to transfer

Either 4-bit or 8-bit serial transfer can be selected. When 4-bit serial transfer is selected, only the lower 4 bits of the transmit/receive data buffer register are used. The upper 4 bits are cleared to "0" when receiving.

The data is transferred in sequence starting at the least significant bit (LSB).

(3) Number of words to transfer

Up to 8 words consisting of 4 bits of data (4-bit serial transfer) or 8 bits (8-bit serial transfer) of data can be transferred continuously. The number of words to be transferred is loaded to BUF in SIOBCR.

An INTSIO interrupt is generated when the specified number of words has been transferred. If the number of words is to be changed during transfer, the serial interface must be stopped before making the change.

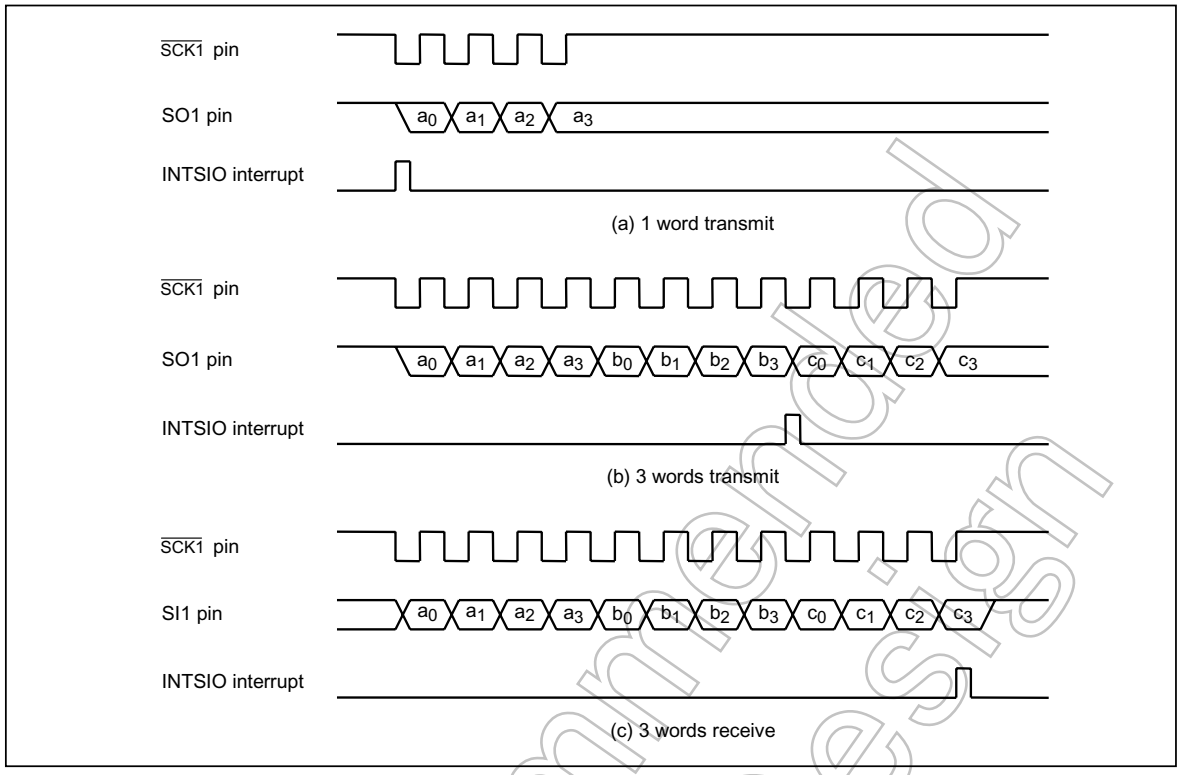


Figure 2.11.7 Number of Bits to Transfer (Example: 4-Bit Serial Transfer)

(4) Transfer mode

SIOM (bits 5 to 3 in SIO1CR) is used to select the transmit, receive, or transmit/receive mode.

a. 4-bit and 8-bit transmit modes

In these modes, the SIO1CR1 is set to the transmit mode and then the data to be transmitted first are written to the data buffer registers (DBR). After the data are written, the transmission is started by setting SIOS to "1". The data are then output sequentially to the SO1 pin in synchronous with the serial clock, starting with the least significant bit (LSB). As soon as the LSB has been output, the data are transferred from the data buffer register to the shift register. When the final data bit has been transferred and the data buffer register is empty, an INTSIO (buffer empty) interrupt is generated to request the next transmitted data.

When the internal clock is used, the serial clock will stop and an automatic-wait will be initiated if the next transmitted data are not loaded to the data buffer register by the time the number of data words specified with the BUF has been transmitted. Writing even one word of data cancels the automatic-wait; therefore, when transmitting two or more words, always write the next word before transmission of the previous word is completed.

Note: Waits are also canceled by writing to a DBR not being used as a transmit data buffer register; therefore, during SIO do not use such DBR for other applications.

When an external clock is used, the data must be written to the data buffer register before shifting next data. Thus, the transfer speed is determined by the maximum delay time from the generation of the interrupt request to writing of the data to the data buffer register by the interrupt service program.

The transmission is ended by clearing SIOS to "0" at the time that the final bit of the data being shifted out has been transferred. That the transmission has ended can be determined from the status of SIOF (bit 7 in SIO1SR) because SIOF is cleared to "0" when a transfer is completed.

When an external clock is used, it is also necessary to clear SIOS to "0" before shifting the next data; otherwise, dummy data will be transmitted and the operation will end.

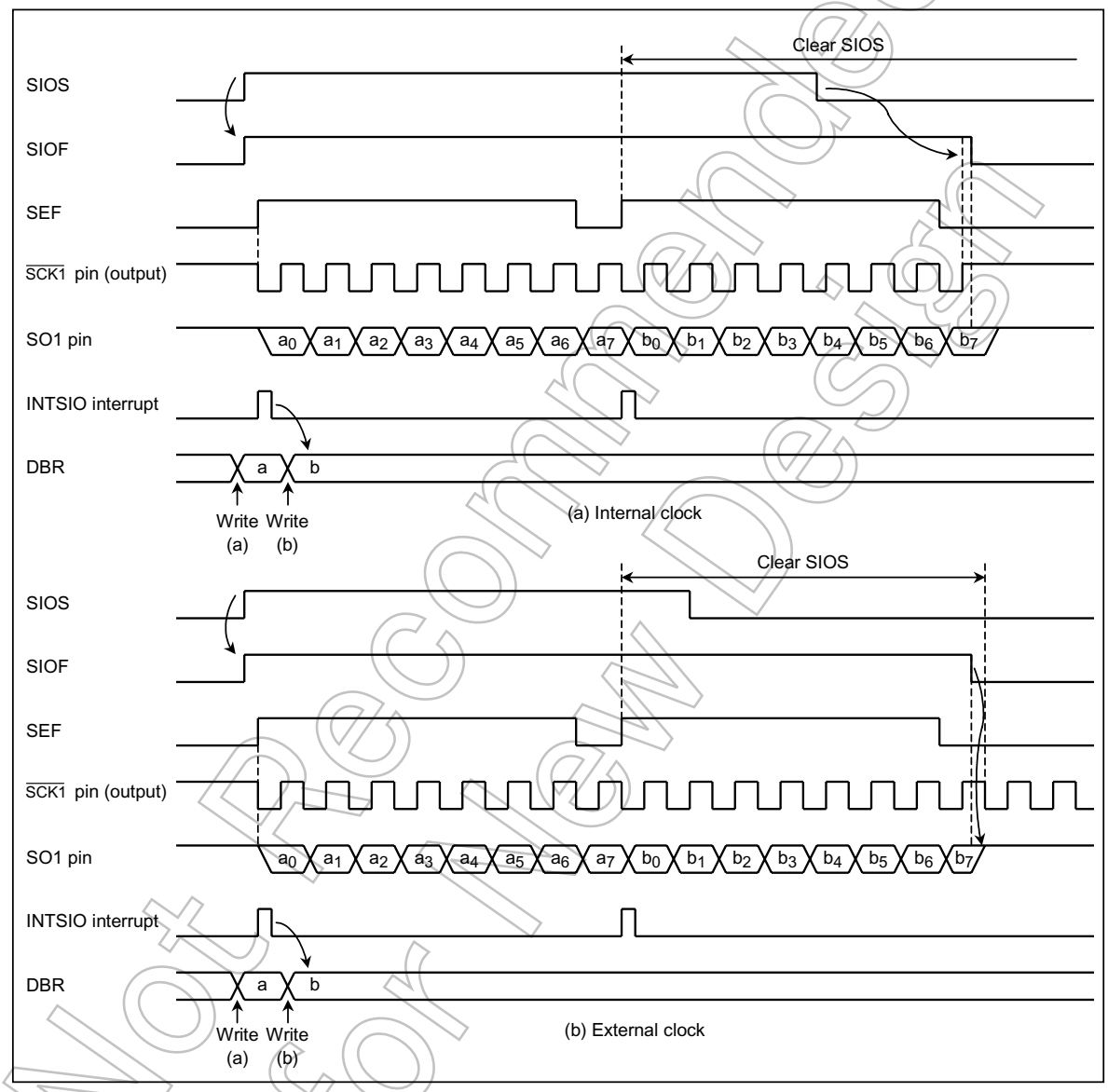


Figure 2.11.8 Transfer Mode (Example: 8-Bit, 1 Word Transfer)

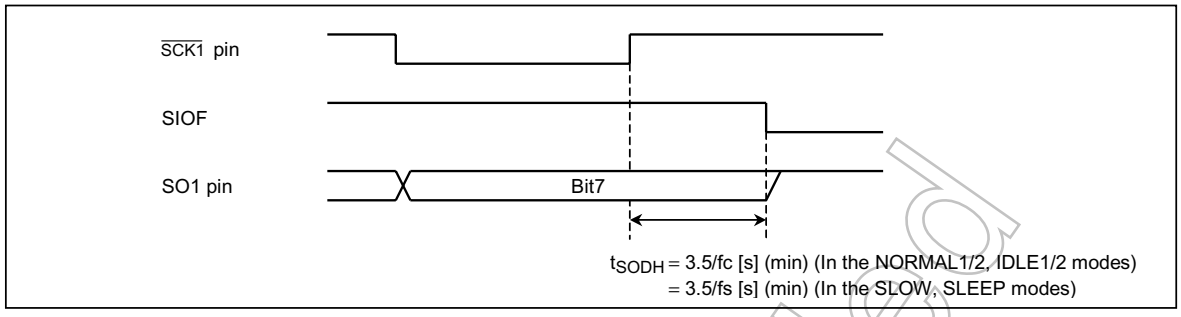


Figure 2.11.9 Transmitted Data Hold Time at End of Transmit

b. 4-bit and 8-bit receive modes

After setting the control registers to the receive mode, set SIOS to "1" to enable receiving. The data are then transferred to the shift register via the SI pin in synchronous with the serial clock. When one word of data has been received, it is transferred from the shift register to the data buffer register (DBR). When the number of words specified with the BUF has been received, an INTSIO (buffer full) interrupt is generated to request that these data be read out. The data are then read from the data buffer registers by the interrupt service program.

When the internal clock is used, and the previous data are not read from the data buffer register before the next data are received, the serial clock will stop and an automatic-wait will be initiated until the data are read. A wait will not be initiated if even one data word has been read.

Note: Waits are also canceled by reading a DBR not being used as a received data buffer register is read; therefore, during SIO do not use such DBR for other applications.

When an external clock is used, the shift operation is synchronized with the external clock; therefore, the previous data are read before the next data are transferred to the data buffer register. If the previous data have not been read, the next data will not be transferred to the data buffer register and the receiving of any more data will be canceled. When an external clock is used, the maximum transfer speed is determined by the delay between the time when the interrupt request is generated and when the data received have been read.

Clear SIOS to "0" to end receiving. When SIOS is cleared, the current data are transferred to the buffer in 4-bit or 8-bit blocks. The receiving mode ends when the transfer is completed. SIOF is cleared to "0" when receiving is ended and thus can be sensed by program to confirm that receiving has ended.

Note: The buffer contents are lost when the transfer mode is switched. If it should become necessary to switch the transfer mode, end receiving by clearing SIOS to "0", read the last data and then switch the transfer mode.

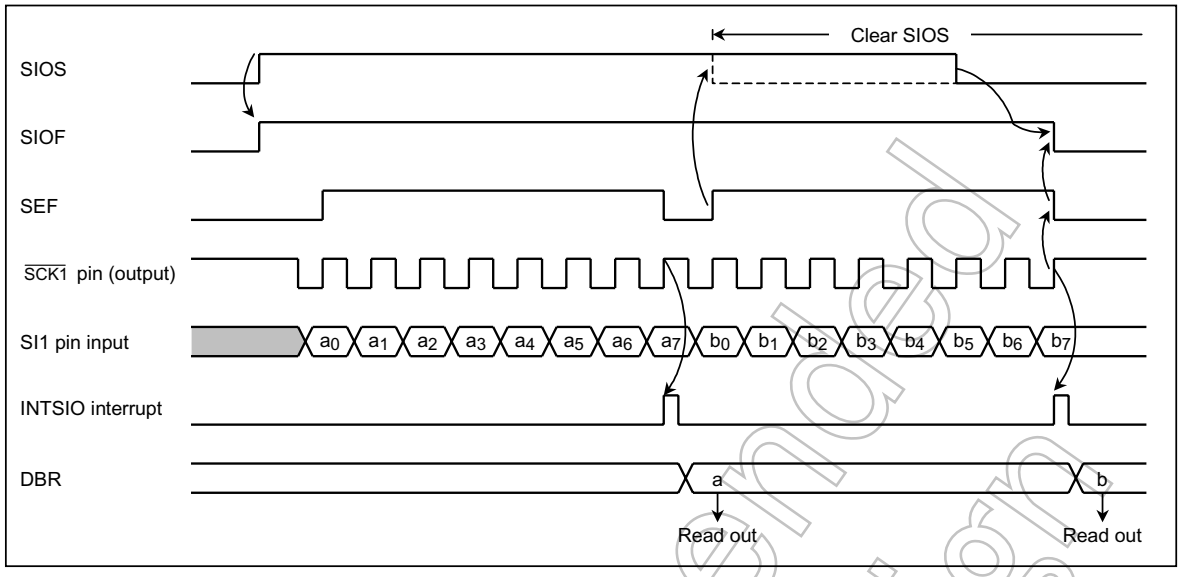


Figure 2.11.10 Receive Mode (Example: 8-Bit, 1 Word, Internal Clock)

c. 8-bit transmit/receive mode

After setting the control registers to the 8-bit transmit/receive mode, write the data to be transmitted first to the data buffer registers (DBR). After that, enable transceiving by setting SIOS to "1". When transmitting, the data are output from the SO1 pin at leading edges of the serial clock. When receiving, the data are input to the SI1 pin at the trailing edges of the serial clock. 8-bit data are transferred from the shift register to the data buffer register. An INTSIO interrupt is generated when the number of data words specified with the BUF has been transferred. The interrupt service program reads the received data from the data buffer register and then writes the data to be transmitted. The data buffer register is used for both transmitting and receiving; therefore, always write the data to be transmitted after reading the received data.

When the internal clock is used, a wait is initiated until the received data are read and the next data are written.

Note: Waits are also canceled by reading a DBR not being used as a received data buffer register is read; therefore, during SIO do not use such DBR for other applications.

When an external clock is used, the shift operation is synchronized with the external clock; therefore, it is necessary to read the received data and write the data to be transmitted next before starting the next shift operation. When an external clock is used, the transfer speed is determined by the maximum delay between generation of an interrupt request and the received data are read and the data to be transmitted next are written.

Clear SIOS to "0" to enable the transmit mode. When SIOS is cleared, the current data are transferred to the data buffer register in 8-bit blocks. The transmit mode ends when the transfer is completed. SIOF is cleared to "0" when receiving is ended and thus can be sensed by program to confirm that receiving has ended.

Note: The buffer contents are lost when the transfer mode is switched. If it should become necessary to switch the transfer mode, end receiving by clearing SIOS to "0", read the last data and then switch the transfer mode.

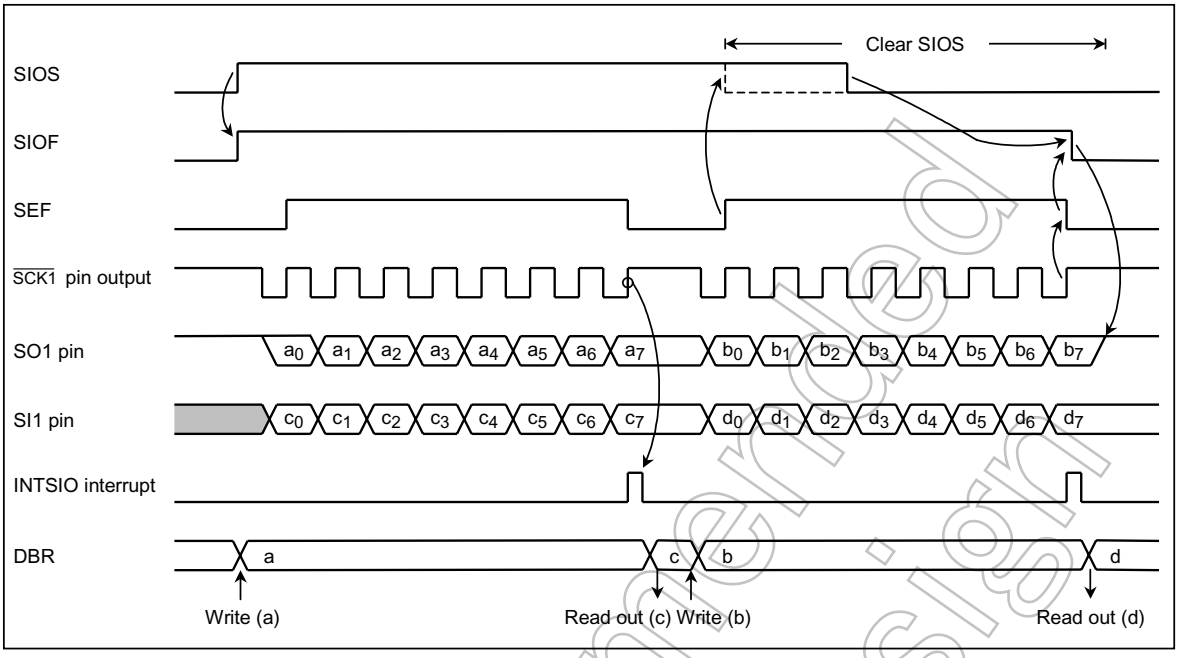


Figure 2.11.11 Transmit/Receive Mode (Example: 8-Bit, 1 Word, Internal Clock)

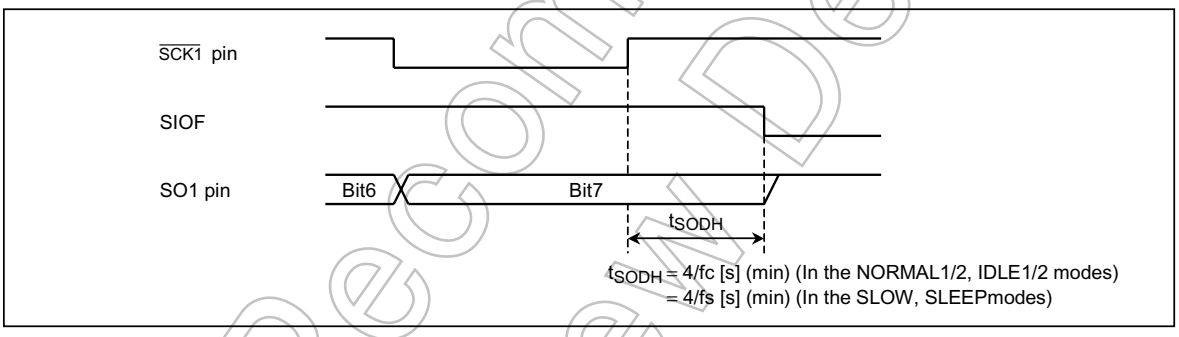


Figure 2.11.12 Transmitted Data Hold Time at End of Transmit/Receive

2.12 8-Bit AD Converter (ADC)

The TMP88CU74 each have an 8-channel multiplexed-input 12-bit successive approximate type AD converter with sample and hold.

2.12.1 Configuration

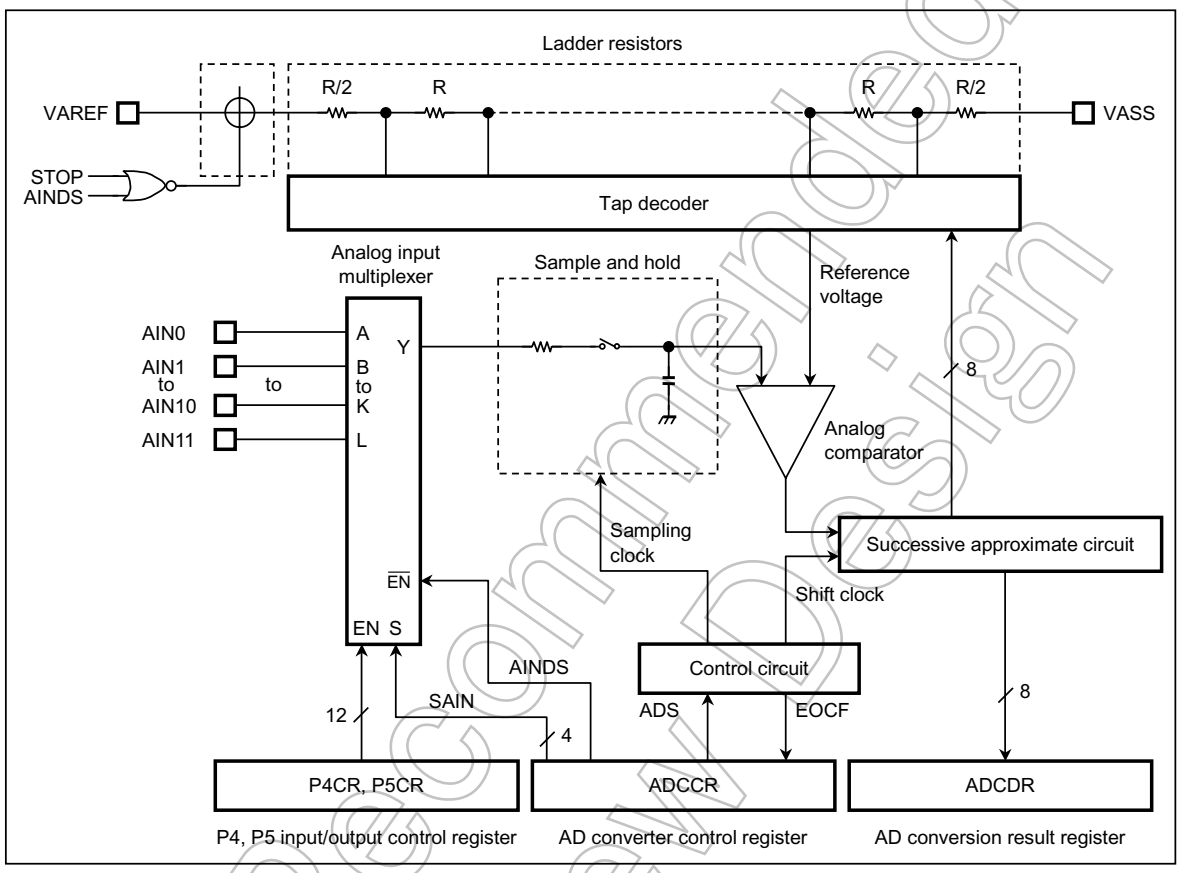


Figure 2.12.1 AD Converter

2.12.2 Control

The AD converter is controlled by the AD converter control register (ADCCR).

The operating state of the AD converter is confirmed by reading EOCF in ADCCR. The AD conversion value is confirmed by reading the contents of AD conversion value registers.

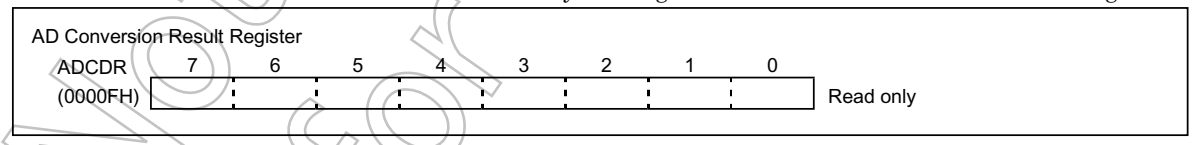


Figure 2.12.2 AD Conversion Result Register

AD Converter Control Register

ADCCR	7	6	5	4	3	2	1	0	
(0000EH)	EOCF	ADS	ACK	AINDS	SAIN				(Initial value: 0000 0000)

SAIN	Analog input selection	0000: AIN0 select 0001: AIN1 select 0010: AIN2 select 0011: AIN3 select 0100: AIN4 select 0101: AIN5 select 0110: AIN6 select 0111: AIN7 select 1000: AIN8 select 1001: AIN9 select 1010: AIN10 select 1011: AIN11 select 1100: Reserved 1101: Reserved 1110: Reserved 1111: Reserved	R/W
AINDS	Analog input control	0: Enable 1: Disable	
ACK	Conversion time selection	0: 184/fc [s]: 23 μs (fc = 8 MHz) 1: 736/fc [s]: 59 μs (fc = 12.5 MHz)	
ADS	AD conversion start	0: — 1: AD conversion start	
EOCF	End of AD conversion flag	0: Under conversion or Before conversion 1: End of conversion	Read only

Note 1: Select analog input when AD converter stops.
Note 2: The ADS is automatically cleared to "0" after starting conversion.
Note 3: The EOCF is cleared to "0" when reading the ADCDR.
Note 4: The EOCF is read-only.
Note 5: (ACK = "0" at fc ≤ 8 MHz)
Note 6: When STOP or SLOW mode is activated, the AD control registers are initialized. After NORMAL mode is resumed, set both the AD control registers again if necessary.

Figure 2.12.3 AD Converter Control Register and AD Conversion Result Register

2.12.3 Operation

Analog reference voltage on high side is applied to the VAREF pin ; on the low side, to the VASS pin. The reference voltage between VAREF and VASS is divided a ladder resistor and compared with the analog voltage input for AD conversion.

(1) Start of AD conversion

First, set the corressponding P4CR and P5CR bit to "0" for analog input. Clear the AINDS (bit 4 in ADCCR) to "0" and select one of twelve analog input AIN11-AIN0 with the SAIN (bits 3 to 0 in ADCCR).

Note: The pin that is not used as an analog input can be used as regular input/output pins. During conversion, do not perform output instruction to maintain a precision for all of the pins.

AD conversion is started by setting the ADS (bit 6 in ADCCR) to "1".

Conversion is accomplished in 59 machine cycles (184/fc [s]).

The EOCF (bit 7 in ADCCR) is set to "1" at end of conversion.

When setting the ADS to "1" under AD conversion, the AD converter circuit is initialized and the AD conversion try again from start.

The sampling of the analog input voltage is excuted at 4 machine cycles after setting the ADS to "1".

Note: The circuit of sample and hold is included in a capacitor of (12 pF (typ.)) through a register (5 kΩ (typ.)).
Therefore, until 4 machine cycles is over, this capacitor must be charged.

(2) Reading of AD conversion result

After the end of conversion, read the conversion result from the ADCDR.

The EOCF is automatically cleared to "0" when reading the ADCDR.

(3) AD conversion in STOP mode

When the MCU places in the STOP mode during the AD conversion, the conversion is terminated and the ADCDR contents become indefinite.

However, if the STOP mode is started after the end of conversion (EOCF = 1), the ADCDR contents are held.

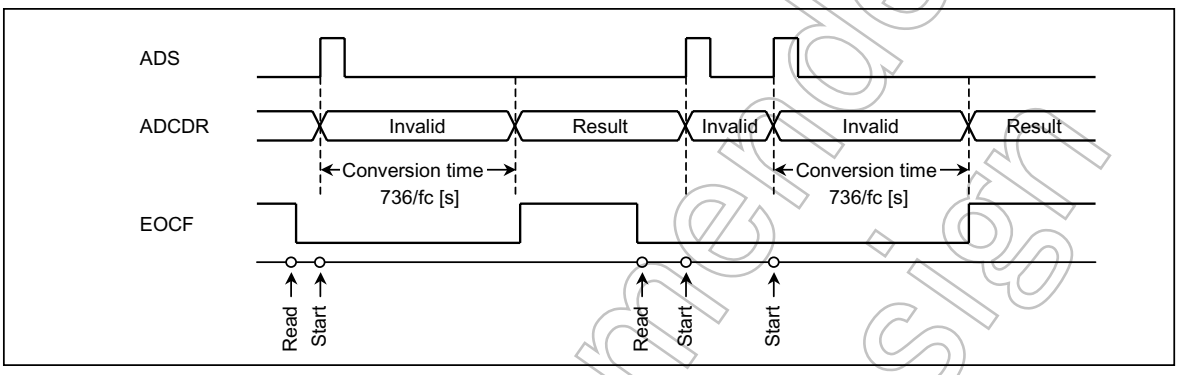


Figure 2.12.4 AD Conversion Timing Chart (ACK = 1)

```
Example:
        ; AIN SELECT
        LD      (ADCCR), 00100100B      ;   Selects AIN4, ACK = 1
        ; AD CONVERT START
        SET     (ADCCR). 6              ;   ADS = 1
SLOOP:  TEST    (ADCCR). 7              ;   EOCF = 1 ?
        JRS     T, SLOOP
        ; RESULT DATA READ
        LD      (9EH), (ADCDR)
```

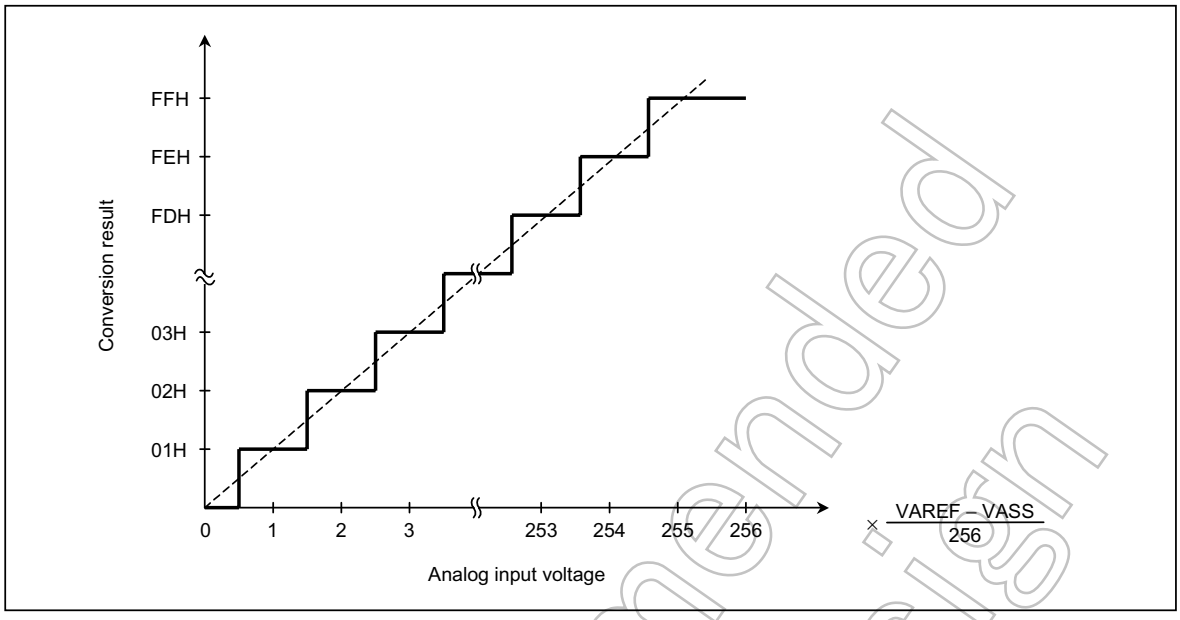


Figure 2.12.5 Analog Input Voltage vs AD Conversion Result (typ.)

2.13 Vacuum Fluorescent Tube (VFT) Driver Circuit

The TMP88CU74 features built-in high-breakdown voltage output buffers for directly driving fluorescent tubes, and a display control circuit used to automatically transfer display data to the output port.

The segment and the digit, as it is the VFT drive circuit which included in the usual products, are not allocated. The segment and the digit can be freely allocated in the timing (T0 to T15) which is specified according to the display tube types and the layout.

2.13.1 Functions

(1) 37 high-breakdown voltage output buffers built-in.

- Large current output pin (typ. 20 mA) 37 (V0 to V36)

There is also the VKK pin used for the VFT drive power supply.

(2) The dynamic lighting system makes it possible to select 1 to 16 digits (T0 to T15) by program.

(3) Pins not used for VFT driver can be used as general-purpose ports.

- Pins can be selected using the VSEL (bits 4 to 0) in VFT control register1 bit by bit.

(4) Display data (112 bytes in DBR) are automatically transferred to the VFT output pin.

(5) Brightness level can be adjusted in 8 steps using the dimmer function.

(6) Table 2.13.1 shown in setting of display time.

Table 2.13.1 Setting of Display Time

DV1CK = 0 [s]	DV1CK = 1 [s]
2^{10}/fc	2^{10}/fc
2^{11}/fc	2^{11}/fc
2^{12}/fc	2^{12}/fc
2^{13}/fc	2^{13}/fc

2.13.2 Configuration

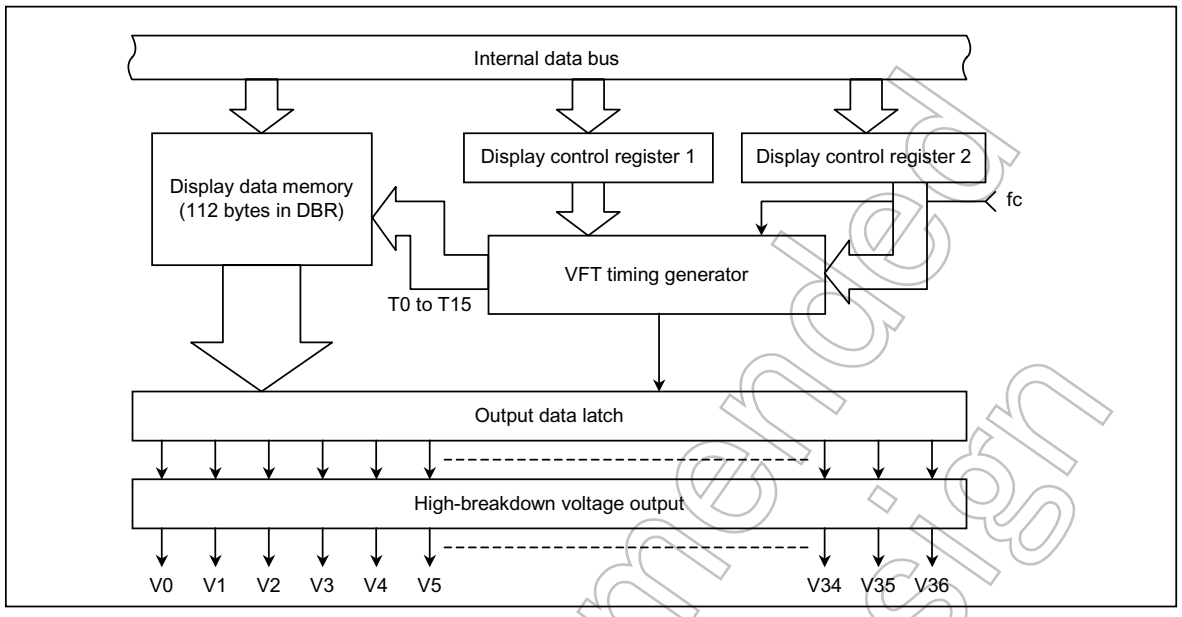


Figure 2.13.1 Vacuum Fluorescent Tube (VFT) Driver Circuit

2.13.3 Control

The VFT driver circuit is controlled by the VFT control registers (VFTCR1, VFTCR2). Reading VFTSR determines the VFT operating status.

Switching the mode from NORMAL1/2 to SLOW or STOP puts the VFT driver circuit into blanking state, and sets segment outputs and digit outputs are cleared to "0". Thus, ports P6 to P9, and PD function as general-purpose output ports with pull-down.

VFT control register 1

VFTCR1 (00029H)

7	6	5	4	3	2	1	0
BLK	SDT			VSEL			

(Initial value: 0000 0000)

BLK	VFT display control	0: Display enable 1: Disable			Write only
SDT	Display time select (tdisp) (Display time of 1 digit)		DV1CK = 0 [s]	DV1CK = 1 [s]	
		00	2^{10}/fc	2^{10}/fc	
		01	2^{11}/fc	2^{11}/fc	
		10	2^{12}/fc	2^{12}/fc	
		11	2^{13}/fc	2^{13}/fc	
VSEL	Automatic display select (When using VFT driver (automatic display), V31 to V0 are only used to output VFT.) Pins which are not selected by the output pins other than the above-mentioned pins can be used as general-purpose input/output pins. (When using as a general-purpose input/output pin, the display data which corresponds to the pin must be set to "0")	00000: 32 (V31 to V0) 00001: 33 (V32 to V0) 00010: 34 (V33 to V0) 00011: 35 (V34 to V0) 00100: 36 (V35 to V0) 00101: 37 (V36 to V0)			

Note 1: fc: High frequency clock

Note 2: VFTCR1 is write-only register, which cannot use any of in read-modify-write instruction such as bit operate, etc.

Figure 2.13.2 VFT Control Register 1

VFT control register 2

VFTCR2 (0002AH)

7	6	5	4	3	2	1	0
DIM				STA			

(Initial value: 000* 0000)

DIM	Dimmer time select	000: (15/16) × tdisp (s) 001: (14/16) × tdisp (s) 010: (12/16) × tdisp (s) 011: (10/16) × tdisp (s) 100: (8/16) × tdisp (s) 101: (6/16) × tdisp (s) 110: (4/16) × tdisp (s) 111: (2/16) × tdisp (s)	Write only
STA	Number of state (display)	0000: 1 display mode (T0) 0001: 2 display mode (T1 to T0) 0010: 3 display mode (T2 to T0) 0011: 4 display mode (T3 to T0) 0100: 5 display mode (T4 to T0) 0101: 6 display mode (T5 to T0) 0110: 7 display mode (T6 to T0) 0111: 8 display mode (T7 to T0) 1000: 9 display mode (T8 to T0) 1001: 10 display mode (T9 to T0) 1010: 11 display mode (T10 to T0) 1011: 12 display mode (T11 to T0) 1100: 13 display mode (T12 to T0) 1101: 14 display mode (T13 to T0) 1110: 15 display mode (T14 to T0) 1111: 16 display mode (T15 to T0)	

Note 1: VFTCR2 is write-only register, which cannot use any of in read-modify-write instruction such as bit operate, etc.

Note 2: Even if a number of the display digit is set a pin which is equal to the digit dose not output.
It is necessary to write data to the data buffer which corresponds to the digit according to the display timing (T0 to T15).

Note 3: *: Don't care

VFTSR (00029H)

7	6	5	4	3	2	1	0
WAIT							

WAIT	VFT operational status monitor	0: VFT display in operation 1: VFT display operation disabled	Read only

Figure 2.13.3 VFT Control Register 2, VFT Status Register

(1) Setting of display mode

VFT display mode is set by VFT control register 1 (VFTCR1) and VFT control register 2 (VFTCR2). VFT control register 1 (VFTCR1) sets 1 display time (tdisp) and the number of display lines (VSEL), and VFT control register 2 (VFTCR2) sets dimmer timer (DIM) and state (STA). (BLK of VFTCR1 must be set to "1".) The segments and the digits are not fixed, so that they can be freely allocated. However the number of states must be specified according to the number of digits of VFT which you use.
(See Display operation in section 2.13.4 for display timing and data setting procedures.)

(2) Display data setting

Data are converted into VFT display data by instructions. The converted data stored in the display data buffer (addresses 00F80 to 00FCF in DBR) are automatically transferred to the VFT driver circuit, then transferred to the high-breakdown voltage output buffer. Thus, to change the display pattern, just change the data in the display data buffer.

Bits in the VFT segment (dot) and display data area correspond one to one. When data are set to 1, the segments corresponding to the bits light. The display data buffer is assigned to the DBR area shown in Figure 2.13.4. (The display data buffer can not be used as data memory)

Bit	0 to 7	0 to 7	0 to 7	0 to 7	0 to 4	Timing
	00F80H	00F90H	00FA0H	00FB0H	00FC0H	T0
	00F81H	00F91H	00FA1H	00FB1H	00FC1H	T1
	00F82H	00F92H	00FA2H	00FB2H	00FC2H	T2
	00F83H	00F93H	00FA3H	00FB3H	00FC3H	T3
	00F84H	00F94H	00FA4H	00FB4H	00FC4H	T4
	00F85H	00F95H	00FA5H	00FB5H	00FC5H	T5
	00F86H	00F96H	00FA6H	00FB6H	00FC6H	T6
	00F87H	00F97H	00FA7H	00FB7H	00FC7H	T7
	00F88H	00F98H	00FA8H	00FB8H	00FC8H	T8
	00F89H	00F99H	00FA9H	00FB9H	00FC9H	T9
	00F8AH	00F9AH	00FAAH	00FBAH	00FCAH	T10
	00F8BH	00F9BH	00FABH	00FBBH	00FCBH	T11
	00F8CH	00F9CH	00FACH	00FBCH	00FCCH	T12
	00F8DH	00F9DH	00FADH	00FBDH	00FCDH	T13
	00F8EH	00F9EH	00FAEH	00FBEH	00FCEH	T14
	00F8FH	00F9FH	00FAFH	00FBFH	00FCFH	T15
Segment	V0 to V7	V8 to V15	V16 to V23	V24 to V31	V32 to V36	

Note: Writing “0” in 7 to 5 bit of address 00FC0H to 00FCFH in DBR.

Figure 2.13.4 VFT Display Data Buffer Memory (DBR)

2.13.4 Display Operation

As the above-mentioned, the segment and the digit are not allocated. After setting of the display timing for the number of digits according to the using VFT and storing the segment and digit data according to the respective timings, clearing BLK in VFTCR1 to 0 starts VFT display.

Figure 2.13.5 shows the VFT drive pulse and Figure 2.13.6, Figure 2.13.7 show the display operation.

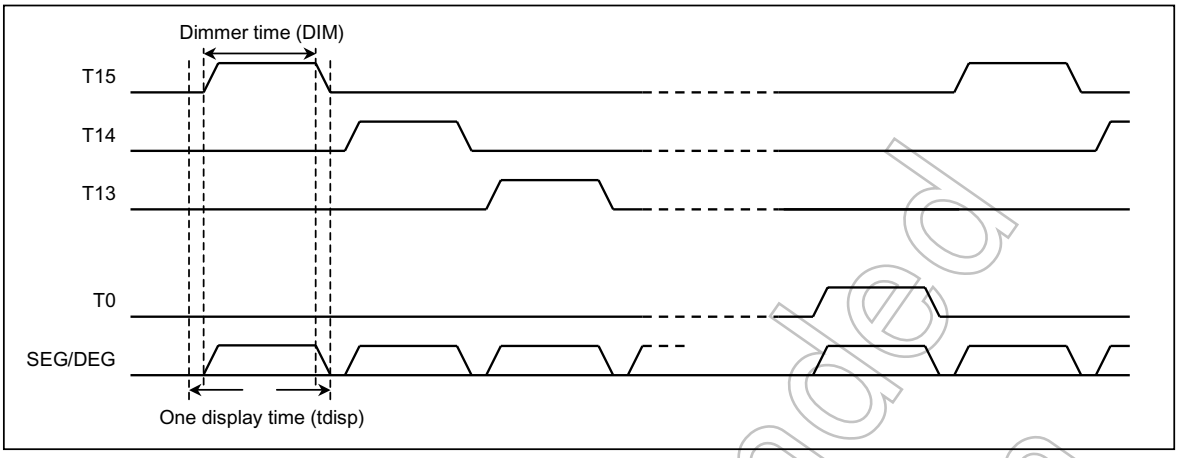


Figure 2.13.5 VFT Drive Waveform and Display Timing

2.13.5 Example of Display Operation

(1) For Conventional type VFT

When using the conventional type VFT, the output timing of the digits is specified to output 1 digit for 1 timing. Data must be set to output the pins which are specified to the digit in sequence. The following figure shows a data allocation of the display data buffer (DBR) and the output timing when VFT of 10 digits is used and V0 to V9 pins are allocated as the digit outputs. (When data is first written by the data buffer which corresponds to the digit pin, it is unnecessary to rewrite the data later.)

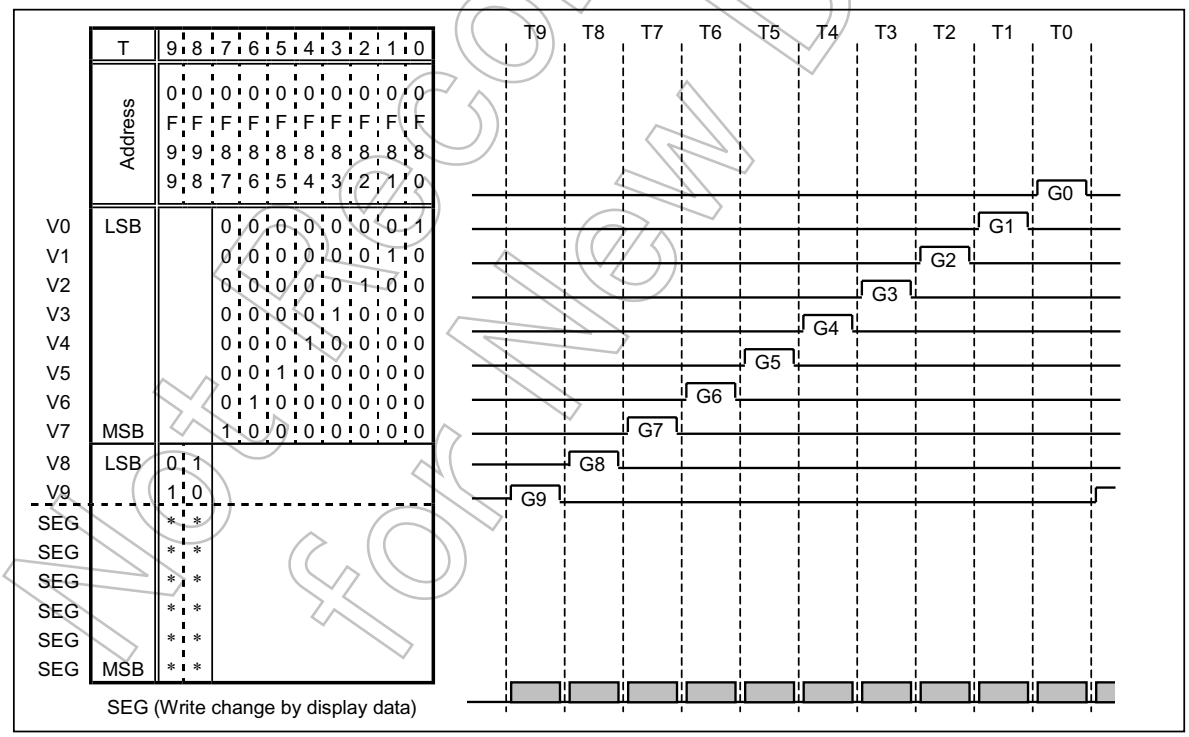


Figure 2.13.6 Example of Conventional Type VFT Driver Pulse

(2) For Grid scan type VFT

When using the grid scan type VFT, two or more grids must be simultaneously selected to turn the display pattern which contains two or more grids on. Additionally, the timing and the data must be determined to set the grid scan mode as follows.

- When the display pattern which is fully set in the respective grids is turned on, only the grids which correspond as ever must be scanned in sequence to turn on the display pattern. (timing of T8 to T3 in the following figure)
- When the display pattern which contains two or more grids is turned on, two or more corresponding grids are simultaneously selected to turn on the display pattern. (timing of T2 to T0 in the following figure)

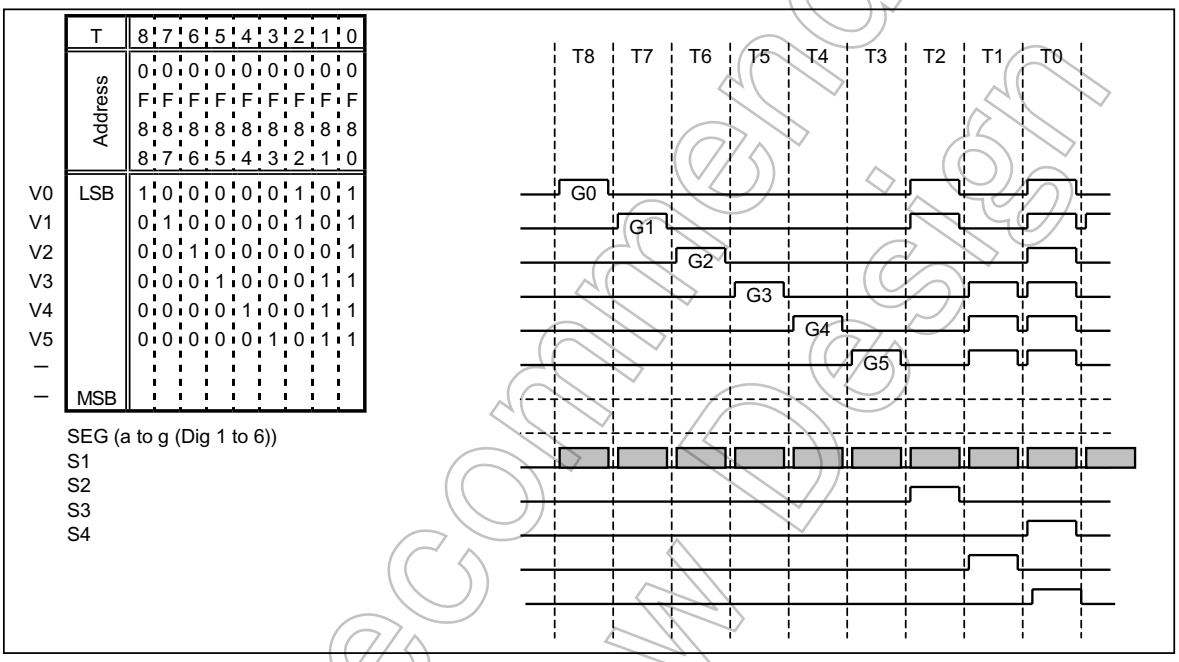


Figure 2.13.7 Grid Scan Type Display Vacuum Fluorescent Tube Ware

2.13.6 Port Function

(1) High-breakdown voltage buffer

To drive fluorescent display tube, clears the port output latch to "0". The port output latch is initialized to 0 at reset.

It is recommended that ports P5, P6, P7, P8 and P9 should be used as VFT driver output. Precaution for using as general-purpose I/O pins are follows.

Note: When not using a pin which is pulled down to pin V_{KK} (RK = typ. 80 kΩ), it must be set to open. It is necessary to clear the port output latch and the data buffer memory (DBR) to "0".

1. Ports P6 to P9

When a part of P6 to P9 is used as the input/output pin (VFT driver in operation), the data buffer memory (DBR) of the segment which is also used as the input/output pin must be cleared to "0".

2. Port PD

VFT output and usual input/output are controlled by VSEL of VFT control register in bits. When a pin which is pulled down to pin V_{KK} is used as usual output or input, the following cautions are required.

(a) When outputting

When level "L" is output, a port which is pulled down to pin V_{KK} is pin V_{KK} voltage. Such processes as clamping with the diode as shown in figure 2.13.8. (a) are necessary to prevent pin V_{KK} voltage applying to the external circuit.

(b) When inputting

When the external data is input, the port output latch is cleared to "0".
The input threshold is the same as that of the other usual input/output port. However it is necessary to drive R_K (typ. 80 kΩ) sufficiently because of pulled down to pin V_{KK}.

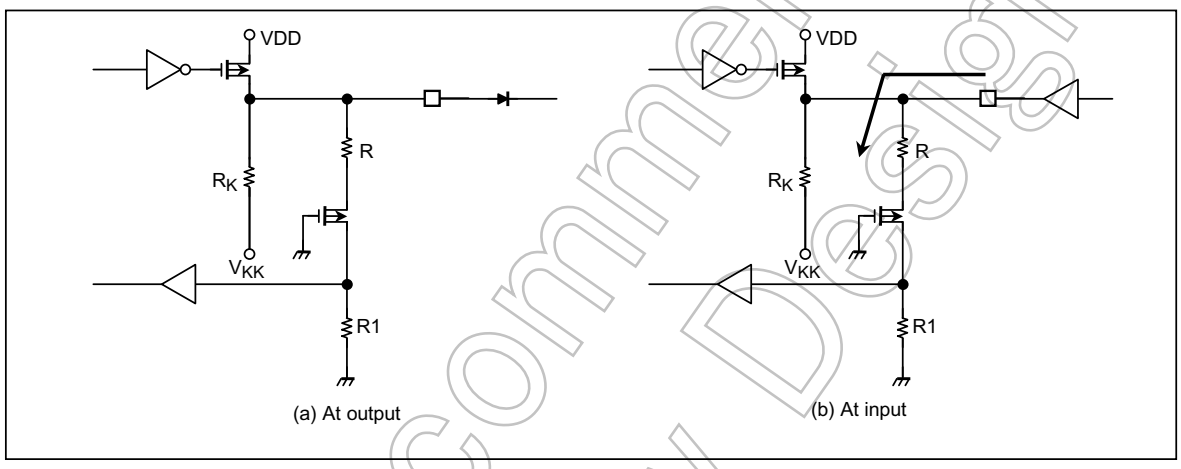
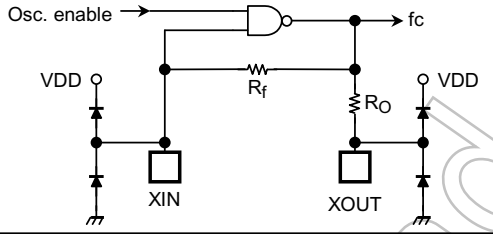
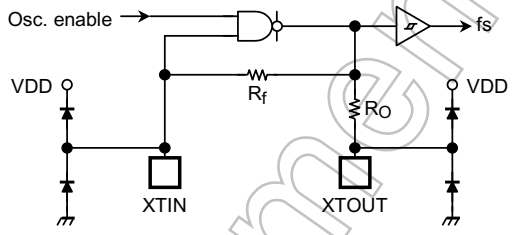
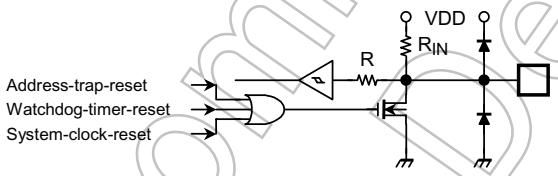
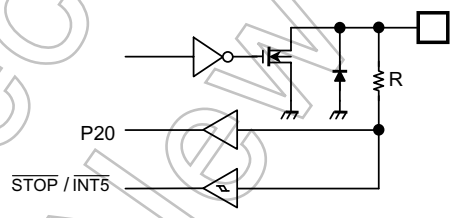
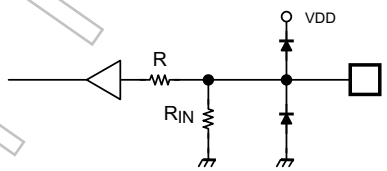


Figure 2.13.8 External Circuit Interface

Input/Output Circuitry

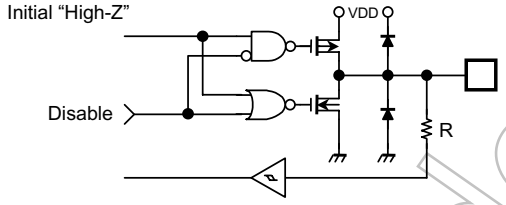
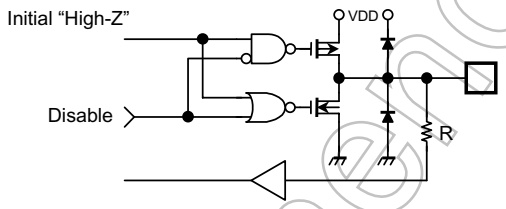
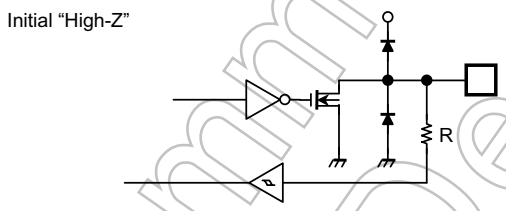
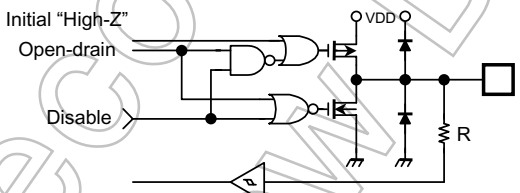
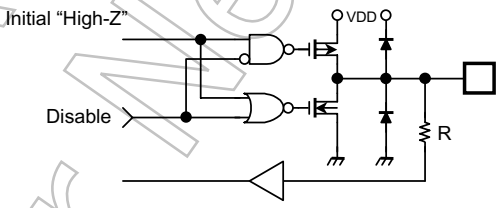
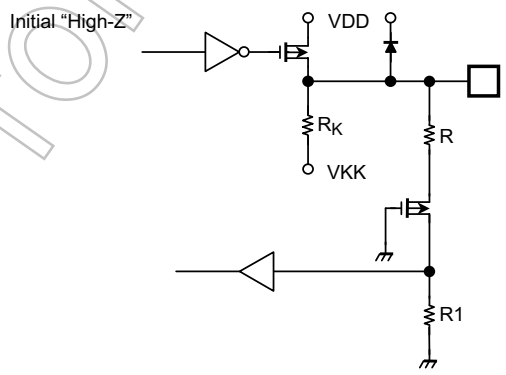
(1) Control pins

The input/output circuitries of the TMP88CU74 control pins are shown below.

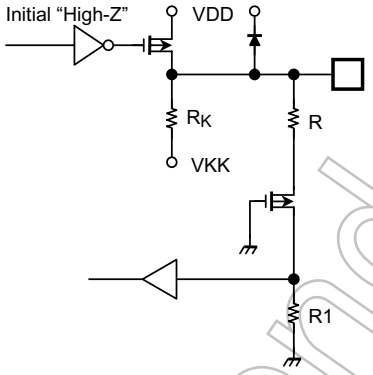
Control Pin	I/O	Circuitry	Remarks
XIN XOUT	Input Output		Resonator connecting pins (high-frequency) R_f = 1.2 MΩ (typ.)
XTIN XTOUT	Input Output		Resonator connecting pins (low-frequency) R_f = 6 MΩ (typ.) R_O = 220 kΩ (typ.)
$\overline{\text{RESET}}$	I/O		Hysteresis input R_{IN} = 220 kΩ (typ.) R = 1 kΩ (typ.)
$\overline{\text{STOP}}$ / $\overline{\text{INT5}}$	Input		Hysteresis input R = 1 kΩ (typ.)
TEST	Input		Pull-down resistor R_{IN} = 70 kΩ (typ.) R = 1 kΩ (typ.)

Note: The TEST pin of the TMP88PU74 does not have a pull-down resistor. Fix the test pin at low-level.

(2) – 1. Input/Output ports

Port	I/O	Input/Output Circuitry	Remarks
P00 P01 P1	I/O		Tri-state I/O Hysteresis input R = 1 kΩ (typ.)
P02 to P07	I/O		Tri-state I/O R = 1 kΩ (typ.)
P2	I/O		Sink open drain output Hysteresis input R = 1 kΩ
P3	I/O		Tri-state I/O Hysteresis input Programmable・Open-drain R = 1 kΩ (typ.)
P4 P5	I/O		Tri-state I/O R = 1 kΩ (typ.)
P6 P7 P8 P9	I/O		Source open drain I/O High-breakdown voltage RK = 80 kΩ (typ.) R = 1 kΩ (typ.) R1 = 200 kΩ (typ.)

(2) – 2. Input/Output ports

Port	I/O	Input/Output Circuitry	Remarks
PD	I/O		Source open drain I/O High-breakdown voltage R_K = 80 kΩ (typ.) R = 1 kΩ (typ.) R1 = 200 kΩ (typ.)

Electrical Characteristics

Absolute Maximum Ratings (V_{SS} = 0 V)

Parameter	Symbol	Pins	Ratings	Unit
Supply Voltage	V_{DD}		−0.3 to 6.5	V
Input Voltage	V_{IN}		−0.3 to V_{DD} + 0.3	
Output Voltage	V_{OUT1}	P2, P3 (at open-drain)	−0.3 to V_{DD} + 0.3	
	V_{OUT2}	P6, P7, P8, P9, PD	V_{DD} − 40 to V_{DD} + 0.3	
Output Current (Per 1 pin)	I_{OUT1}	P0, P1, P2, P3, P4, P5 Ports	3.2	mA
	I_{OUT2}	P6, P7, P8, P9, PD Ports	−25	
Output Current (Total)	ΣI_{OUT1}	P0, P1, P3, P4, P5 Ports	−40	
	ΣI_{OUT2}	P0, P1, P2, P3, P4, P5 Ports	120	
	ΣI_{OUT3}	P6, P7, P8, P9, PD Ports	−160	
Power Dissipation [Topr = 25°C]	PD (Note 2)		1200	mW
Soldering Temperature (time)	Tsld		260 (10 s)	°C
Storage Temperature	Tstg		−55 to + 125	
Operating Temperature	Topr		−30 to + 70	

Note 1: The absolute maximum ratings are rated values which must not be exceeded during operation, even for an instant. Any one of the ratings must not be exceeded. If any absolute maximum rating is exceeded, a device may break down or its performance may be degraded, causing it to catch fire or explode resulting in injury to the user. Thus, when designing products which include this device, ensure that no absolute maximum rating value will ever be exceeded.

Note 2: Power Dissipation (PD); For PD, it is necessary to decrease 14.3 mw/°C.

Recommended Operating Conditions (V_{SS} = 0 V, Topr = −30 to 70°C)

Parameter	Symbol	Pins	Conditions		Min	Max	Unit
Supply Voltage	V_{DD}		fc = 12.5 MHz	NORMAL1, 2 modes	4.5	5.5	V
				IDLE1, 2 modes			
			fs = 32.768 KHz	SLOW modes	2.7		
				SLEEP modes			
				STOP modes	2.0		
Input High Voltage	V_{IH1}	Except hysteresis input	$V_{DD} \geq 4.5$ V		$V_{DD} \times 0.70$	V_{DD}	
	V_{IH2}	Hysteresis input			$V_{DD} \times 0.75$		
	V_{IH3}		$V_{DD} < 4.5$ V		$V_{DD} \times 0.90$		
Input Low Voltage	V_{IL1}	Except hysteresis input	$V_{DD} \geq 4.5$ V		0	$V_{DD} \times 0.30$	
	V_{IL2}	Hysteresis input				$V_{DD} \times 0.25$	
	V_{IL3}		$V_{DD} < 4.5$ V			$V_{DD} \times 0.10$	
Clock Frequency	fc	XIN, XOUT	V_{DD} = 4.5 to 5.5 V (Note 2)		8	12.5	MHz
		XTIN, XTOUT	V_{DD} = 2.7 to 5.5 V		30.0	34.0	kHz

Note 1: The recommended operating conditions for a device are operating conditions under which it can be guaranteed that the device will operate as specified. If the device is used under operating conditions other than the recommended operating conditions (Supply voltage, Operating temperature range, Specified AC/DC values etc.), malfunction may occur. Thus, when designing products which include this device, ensure that the recommended operating conditions for the device are always adhered to.

Note 2: Clock frequency fc: Supply voltage range is specified in NORMAL 1/2 mode and IDLE 1/2 mode.

How to Calculate Power Consumption.

With the TMP88CU74, a pull-down resistor (R_K = 80 kΩ typ.) can be built into a VFT driver using mask option (port by port). The share of VFT driver loss (VFT driver output loss + pull-down resistor (R_K) loss) in power consumption Pmax is high. When using a fluorescent display tube with a large number of segments, the maximum power consumption P_D must not be exceeded.

Power consumption Pmax = operating power consumption + normal output port loss + VFT driver loss

Where,

1. Operating power consumption: $V_{DD} \times I_{DD}$
2. Normal power consumption: $\Sigma I_{OUT2} \times 0.4$
3. VFT driver loss: VFT driver output loss + pull-down resistor (R_K) loss

Example:

When Ta = 10 to 50°C and a fluorescent display tube with segment output = 3 mA, digit output = 15 mA, Vxx = –25 V is used.

Operating conditions: V_{DD} = 5 V±10%, fc = 12.5 MHz, VFT dimmer time (DIM) = (14/16) × tseg:

Power consumption Pmax = (1) + (2) + (3)

Where, segments pin = X grid pin = Y, Y = 2

1. Operating power consumption: $V_{DD} \times I_{DD}$ = 5.5 V × 20 mA = 110 mW
2. Normal output port loss: $\Sigma I_{OUT2} \times 0.4$ V = 120 mA × 0.4 V = 48 mW
3. VFT driver loss: segment pin = 3 mA × 2 V × number of segmentsX = 6 mW × X × number of gridsY
   digit pin = 15 mA × 2 V × 14/16 (DIM) = 52.5 mW
   R_K loss = $(5.5 + 25\ V)^2/50\ k\Omega$ × (number of segments X + number of digits Y) = 18.605 mW × (X + 2)

Therefore, Pmax = 110 mW + 48 mW + 6 mW × X + 52.5 mW + 18.605 mW × (X + 2)
= 253.71 mW + 24.605 X

Maximum power consumption P_D when Ta = 50°C is determined by the following equation:

P_D > Pmax
842.5 mW > 253.71 + 24.605 X
23.9 > X

Thus, a fluorescent display tube with less than 23 segments can be used. If a fluorescent display tube with 23 segments or more is used, either a pull-down resistor must be attached externally , or the number of segments to be lit must be kept to less than 23 by software.

DC Characteristics (V_{SS} = 0 V, Topr = –30 to 70°C)

Parameter	Symbol	Pins	Conditions	Min	Typ.	Max	Unit
Hysteresis Voltage	V_{HS}	Hysteresis input			0.9	—	V
Input Current	I_{IN1}	TEST	V_{DD} = 5.5 V V_{IN} = 5.5 V/0 V	—	—	±2	μA
	I_{IN2}	Open drain ports, Tri-state ports					
	I_{IN3}	$\overline{RESET}$, $\overline{STOP}$					
Input Resistance	R_{IN3}	$\overline{RESET}$		100	220	450	kΩ
Pull-down Resistance	R_K	Source open drain ports	V_{DD} = 5.5 V, V_{KK} = –30 V	50	80	110	
Output Leakage Current	I_{LO1}	Sink open drain ports	V_{DD} = 5.5 V, V_{OUT} = 5.5 V	—	—	2	μA
	I_{LO2}	Source open drain ports	V_{DD} = 5.5 V, V_{OUT} = –32 V	—	—	–2	
	I_{LO3}	Tri-state ports	V_{DD} = 5.5 V, V_{OUT} = 5.5 V/0 V	—	—	2	
Output High Voltage	V_{OH2}	Tri-state ports	V_{DD} = 4.5 V, I_{OH} = –0.7 mA	4.1	—	—	V
Output Low Voltage	V_{OL}	Except XOUT	V_{DD} = 4.5 V, I_{OL} = 1.6 mA	—	—	0.4	
Output High current	I_{OH}	P6, P7, P8, P9, PD Port	V_{DD} = 4.5 V, V_{OH} = 2.4 V	—	–20	—	mA
Supply Current in NORMAL 1, 2 modes	I_{DD}		V_{DD} = 5.5 V V_{IN} = 5.3 V/0.2 V fc = 12.5 MHz fs = 32.768 kHz	—	18	26	
Supply Current in IDLE 1, 2 modes				—	5.5	8.5	
Supply Current in SLOW mode			V_{DD} = 3.0 V V_{IN} = 2.8 V/0.2 V fs = 32.768 kHz	—	30	60	μA
Supply Current in SLEEP mode				—	15	30	
Supply Current in STOP mode			V_{DD} = 5.5 V V_{IN} = 5.3 V/0.2 V	—	0.5	10	

Note 1: Typical values show those at Topr = 25°C, VDD = 5 V.

Note 2: Input Current IIN1,IIN3; The current through resistor is not included, when the input resistor (pull-up/pull-down) is contained.

AD Conversion Characteristics (V_{SS} = 0 V, V_{DD} = 4.5 to 5.5 V, Topr = –30 to 70°C)

Parameter	Symbol	Conditions	Min	Typ.	Max	Unit
Analog Reference Voltage	V_{AREF}		4.5	—	V_{DD}	V
	V_{ASS}		V_{SS}			
Analog Input Voltage	V_{AIN}		V_{ASS}	—	V_{AREF}	
Analog Supply Current	I_{REF}	V_{AREF} = 5.5 V, V_{ASS} = 0.0 V	—	0.5	1.0	mA
Nonlinearity Error		V_{DD} = 5.0 V, V_{SS} = 0.0 V V_{AREF} = 5.000 V V_{ASS} = 0.000 V	—	—	±1	LSB
Zero Point Error			—	—	±1	
Full Scale Error			—	—	±1	
Total Error			—	—	±2	

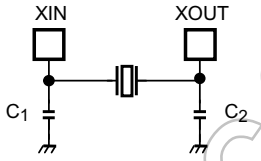
Note: Total errors includes all errors, except quantization error.

AC Characteristics (V_{SS} = 0 V, V_{DD} = 4.5 to 5.5 V, Topr = –30 to 70°C)

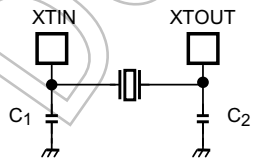
Parameter	Symbol	Conditions	Min	Typ.	Max	Unit
Machine Cycle Time	tcy	In NORMAL1, 2 modes	0.32	—	10	µs
		In IDLE 1, 2 modes				
		In SLOW mode	117.6	—	133.3	
		In SLEEP mode				
High Level Clock Pulse Width	t_{WCH}	For external clock operation (XIN input), fc = 12.5 MHz	33.75	—	—	ns
Low Level Clock Pulse Width	t_{WCL}					
High Level Clock Pulse Width	t_{WSH}	For external clock operation (XTIN input), fs = 32.768 kHz	14.7	—	—	µs
Low Level Clock Pulse Width	t_{WSL}					

Recommended Oscillating Conditions (V_{SS} = 0 V, V_{DD} = 4.5 to 5.5 V, Topr = –30 to 70°C)

Parameter	Oscillator	Oscillation Frequency	Recommended Oscillator		Recommended Constant	
					C_1	C_2
High-frequency Oscillation	Ceramic Resonator	12.5 MHz	Murata	CSA12.5MTZ	30 pF	30 pF
		8 MHz	Murata	CSA8.00MTZ	30 pF	30 pF
	Crystal Oscillator	12.5 MHz	NDK	AT-51	10 pF	10 pF
Low-frequency Oscillation	Crystal Oscillator	32.768 kHz	NDK	MX-38T	15 pF	15 pF



(1) High-frequency Oscillation



(2) Low-frequency Oscillation

Note 1: An electrical shield by metal shield plate on the surface of IC package should be recommendable in order to prevent the device from the high electric fieldstress applied from CRT (Cathode Ray Tube) for continuous reliable operation.

Note 2: The product numbers and specifications of the resonators by Murata Manufacturing Co., Ltd. are subject to change. For up-to-date information, please refer to the following URL;

http://www.murata.co.jp/search/index.html