§ PATENTEN

1. PATENT：『電流源控制及補償觸控電容感測方法及其裝置』
   PAT NO. I339356 (Taiwan)
   PAT NO. ZL 2007 1 0202087. 0 (CHINA)
2. PATENT：『具環境變化校正的電容式觸控感測裝置』
   PAT NO. M383780 (Taiwan)
   PAT NO. ZL 2010 2 0141537. 7 (CHINA)
3. PATENT：『省電型多鍵觸摸開關感測裝置』
   PAT NO. M375250 (Taiwan)
   PAT NO. ZL 2010 2 0302392. 4 (CHINA)

§ General Description:

TTP259 MCU is an easy-used 4-bit CPU base microcontroller. It contains 4032-word ROM、384-nibble RAM、time base、timer/counter、interrupt service、IO control hardware、PWM output、IIC function、LVR and touch pad feature for specified applications. The device is also suitable for diverse simple applications in control appliance and consumer product.

§ Features:

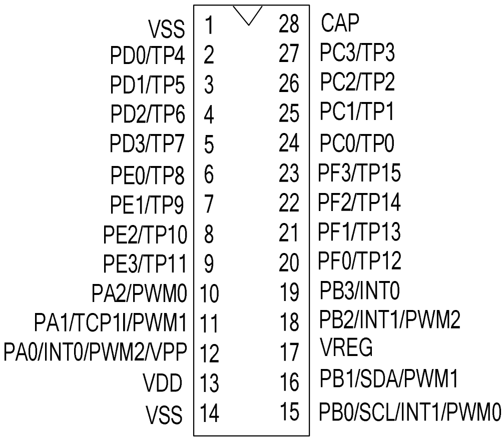
1. Tontek RISC 4-bit CPU core
2. Total 26 crucial instructions and two addressing mode
3. Most instructions need 1 word and 1 machine cycle(2 CPU clocks) except read table instruction(RTB)
4. Advance CMOS process
5. Working memory with 4032*16 program ROM and 384*4 SRAM
6. 4-level stacks
7. Operating voltage: 2.4V~5.5V(LVR=2.2V); 3.3V~5.5V(LVR=3.0V); 2.2V~5.5V(LVR OFF)
8. System operating frequency: (at VDD=5V)
   . High speed system oscillator (OSCH)
     - ✧ Built-in RC oscillator: 4MHz(typical)
   . Low speed peripheral oscillator (OSCL)
     - ✧ Built-in RC oscillator: 16KHz(typical)
9. Provide 7 IO+16 touch pad or 23 general programmable IO
   - ✧ IO port built-in key wake-up feature enable by software setting
   - ✧ Provide external interrupt inputs
   - ✧ Provide internal signal outputs, like PWM

10. TWO time base
    - Time base offers 2 various period interrupt request
11. One 8-bit TCP1 auto-reload timer/counter
    - 4 timer clock sources selected by software
12. One 12-bit TCP2 auto-reload timer/counter, can improve PWM function
    - 4 timer clock sources selected by software
13. Built-in 3 set 12-bit PWM output
14. MCU system protection and power saving controlled mode
    - Built-in watch dog timer (WDT) circuit
    - Built-in low voltage reset (LVR) function
    - Out of user program's range detection
    - ROM code error detection
    - Provide high/low system operating speed, sleep and stop mode for power saving control
15. Provide 16 pins with touch pad detection
16. LDO voltage can select 2.7V or 4.2V output by mask option
17. LVR voltage can select 2.2V or 3.0V by mask option
18. Provide two wire serial interface (IIC-BUS)
19. Provide 10 interrupt sources
    - External: INT0, INT1 shared with IO pad
    - Internal: two time base, two timer/counter
    - Two touch pad's interrupt
    - Two IIC interrupt
20. Provide package types
    - 28SSOP/20TSSOP/16SOP

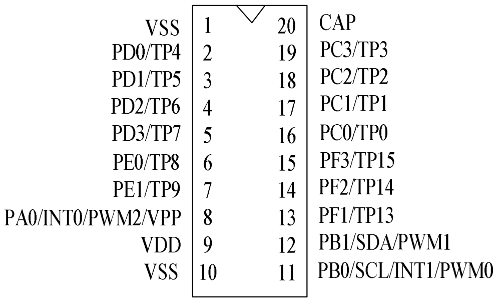
§ Applications:

1. Household electric appliances
2. Consumer products
3. Measurement controller

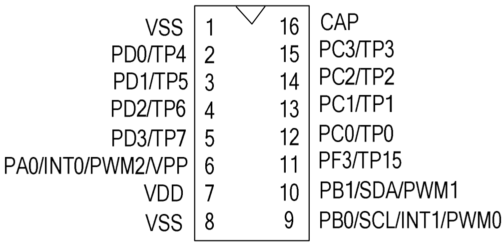
§ Package Description:



28SSOP-A

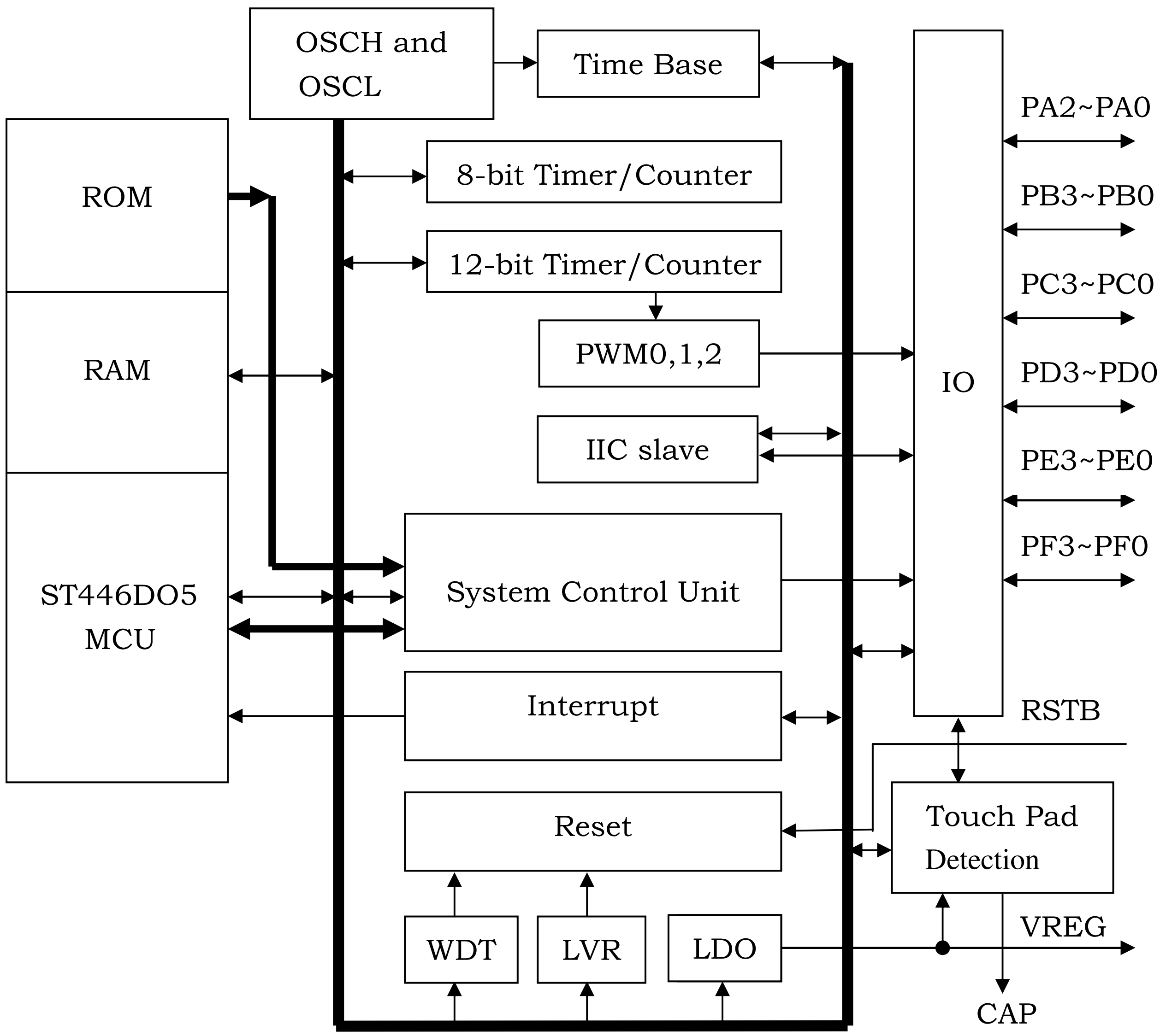


20TSSOP-B



16SOP-B

§ Block Diagram:



§ Pad Description:

Pad Name	Share Pad	IO	Pad	Mask Option	Pad Description
VDD	-	Power	+2	-	Positive power supply.
VSS	-	Power	+4	-	Negative power supply, ground.
RSTB	-	I	+1	-	External reset input, active low.
PA0 PA1 PA2	INT0/PWM2/VPP TCP1I/PWM1 PWM0	IO/I/O IO/I/O IO/O	+3	Yes	IO port with external interrupt input, external clock input and PWM output. PA0 is shared with external interrupt input, PA1 is shared with external clock input, PA0,PA1,PA2 is shared with PWM output.
PB0 PB1 PB2 PB3	SCL/INT1/PWM0 SDA/PWM1 INT1 INT0/PWM2	IO/I/O IO/O IO/I IO/I/O	+4	Yes	IO port with internal IICBUS, external interrupt input and PWM output. PB0,PB1 is shared with internal IICBUS, PB0,PB2,PB3 is shared with external interrupt input, PB0,PB1,PB3 is shared with PWM output.
PC0 PC1 PC2 PC3	TP0 TP1 TP2 TP3	IO/I IO/I IO/I IO/I	+4	-	IO port or touch pad input.
PD0 PD1 PD2 PD3	TP4 TP5 TP6 TP7	IO/I IO/I IO/I IO/I	+4	-	IO port or touch pad input.
PE0 PE1 PE2 PE3	TP8 TP9 TP10 TP11	IO/I IO/I IO/I IO/I	+4	-	IO port or touch pad input.
PF0 PF1 PF2 PF3	TP12 TP13 TP14 TP15	IO/I IO/I IO/I IO/I	+4	-	IO port or touch pad input.
CAP	-	O	+1	-	Touch signal output.
VREG	-	Power	+1	-	LDO voltage output.
Total pad	-	-	32	-	-

§ IO Cell Type Description:

Pin Name	IO Type	Description
PA0	Figure IO-G	STD IO with internal PWM output and external interrupt trigger input.
PA1	Figure IO-C	STD IO with internal PWM output and external TCP1 clock input.
PA2	Figure IO-B	STD IO with internal PWM output.
PB0	Figure IO-E	STD IO with internal PWM output and external interrupt trigger input and IIC.
PB1	Figure IO-F	STD IO with internal PWM output and IIC.
PB2	Figure IO-D	STD IO with internal PWM output and external interrupt trigger input.
PB3	Figure IO-H	STD IO with external interrupt trigger input.
PC0~PC3	Figure IO-A	STD IO with touch pad input.
PD0~PD3	Figure IO-A	STD IO with touch pad input.
PE0~PE3	Figure IO-A	STD IO with touch pad input.
PF0~PF3	Figure IO-A	STD IO with touch pad input.

§ Absolute Maximum ratings:

ITEM	SYMBOL	RATING	UNIT
Operating Temperature	Top	-40~+85	℃
Storage Temperature	Tst	-50~+125	℃
Supply Voltage	VDD	VSS-0.3~VSS+6.0	V
OTP Supply Voltage	VPP	VSS-0.3~VSS+12.5	V
Input Voltage	Vin	VSS-0.3~VDD+0.3	V
Human Body Mode	ESD	>5	KV
Note: VSS symbolizes for system ground.			

§ DC and AC Characteristics

§ DC Characteristics: (Test condition at room temperature=25°C)

Parameter	Symbol	Test Condition	Min.	Typ.	Max.	Unit
Operating Voltage	VDD	F_{OSCH}=4MHz, LVR on 2.2V	2.4	-	5.5	V
		F_{OSCH}=4MHz, LVR on 3.0V	3.3	-	5.5	
		F_{OSCH}=4MHz, LVR off	2.2	-	5.5	
Operating Current (Normal Mode, CPU working, IO no load)	I_{nd1}	VDD=5.0V, no load, F_{OSCL} on, F_{OSCH}=4MHz, LVR off, LDO off	-	3.5	4.0	mA
	I_{nd2}	VDD=5.0V, no load, F_{OSCL} on, F_{OSCH} off, LVR off, LDO off	-	30	50	uA
Operating Current (Sleep Mode, CPU stop, IO no load)	I_{sd1}	VDD=5.0V, no load, F_{OSCL} on, F_{OSCH}=4MHz, LVR off, LDO off	-	0.6	0.8	mA
	I_{sd2}	VDD=3.0V, no load, F_{OSCL} on, F_{OSCH} off, LVR off, LDO off	-	5	10	uA
Standby Current (Stop Mode, CPU stop, IO no load)	I_{sd3}	VDD=5.0V, no load, F_{OSCL} off, F_{OSCH} off, LVR off, LDO off	-	-	1	uA
LVR Current	I_{LVR}	VDD=5.0V		55		uA
LDO Current	I_{LDO}	VDD=5.0V		100		uA
Input Ports	V_{IL}	Input Low Voltage	0	-	0.2	VDD
Input Ports	V_{IH}	Input High Voltage	0.8	-	1.0	VDD
RSTB and INT	V_{IL}	Input Low Voltage	0	-	0.3	VDD
RSTB and INT	V_{IH}	Input High Voltage	0.7	-	1.0	VDD
PA0 Sink Current	I_{OL}	VDD=5.0V, VOL=0.6V	-	2	-	mA
PA0 Source Current	I_{OH}	VDD=5.0V, VOH=4.3V	-	-1	-	mA
Output port Sink Current (PA, PB exclude PA0)	I_{OL}	VDD=5.0V, VOL=0.6V	-	32	-	mA
Output Port Source Current (PA, PB exclude PA0)	I_{OH}	VDD=5.0V, VOH=4.3V	-	-8	-	mA
Output port Sink Current (PC, PD, PE, PF)	I_{OL}	VDD=5.0V, VOL=0.6V	-	16	-	mA
Output Port Source Current (PC, PD, PE, PF)	I_{OH}	VDD=5.0V, VOH=4.3V	-	-8	-	mA
IO Port Pull-up Resistor	R_{PH}	VDD=5.0V	100	150	200	KΩ
RSTB Pull-up Resistor	R_{PH}	VDD=5.0V	30	50	80	KΩ
Low Voltage Reset (LVR)	V_{LVR1}	For AC application	2.7	3.0	3.3	V
	V_{LVR2}		2.0	2.2	2.4	V
LDO Voltage	V_{LDO1}		3.8	4.2	4.6	V
	V_{LDO2}		2.4	2.7	3.0	V
Bandgap Voltage	V_{BGAP}		1.0	1.12	1.23	V

§ AC Characteristics: (Test condition at room temperature=25°C)

Parameter	Test Condition		Min.	Typ.	Max.	Unit
External Reset	Low active pulse width t_{RES}		2	-	-	CPU clock
Interrupt input	Low active pulse width t_{INT}		2	-	-	CPU clock
Wake up input	Low active pulse width t_{WKUP}, Application de-bounce should be manipulated by user' software		2	-	-	OSCL
System Oscillator Frequency	F_{OSCH} (Built-in RC)	VDD=5.0V	-	4M	-	Hz
Peripheral Oscillator Frequency	F_{OSCL} (Built-in RC)	VDD=5.0V	-	16K	-	Hz
Startup Period of Oscillators	T_{OSCH} (Built-in RC)	wake-up from off mode	8	-	-	T_{OSCH}
	T_{OSCL} (Built-in RC)	Wake-up from off mode	8	-	-	T_{OSCL}
Stable Time Of System Clock Switching	T_{OSCH} (Built-in RC)	OSCL→OSCH and OSCH off	8	-	-	T_{OSCL}
	(If H/L=0 then OSCH stop)					
	T_{OSCL} (Built-in RC)	OSCH→OSCL and OSCL on	-	-	-	T_{OSCL}
Timer/Counter input clock frequency	Input frequency rating, no de-bounce circuit built-in, VDD=5V		DC	-	4M	Hz
System Stable Time after Power up	After power up, the system needs to initialize the configured state and OST		-	-	40	ms

§ Memory Map:

ROM ADDRESS	RAM ADDRESS	Function Block
000_H~FBF_H	-	Program ROM [4032*16]
-	000_H~007_H	File Registers
-	008_H~$01F_H$	Peripheral registers (I)
-	020_H~$19F_H$	Working RAM [384*4]
-	200_H~304_H	Peripheral registers (II)

§ Interrupt Vectors:

Interrupt Vectors	Function Description
$000	Hardware reset
$001	Hardware interrupt

§ File registers:

Address	Symbol	R/W	Default	Description
000_H	(DP1)	R/W	----	Indirect addressing register
001_H	ACC	R/W	xxxx	Accumulator and read table 1st data
002_H	TB1	R/W	xxxx	Read table 2nd data
003_H	TB2	R/W	xxxx	Read table 3rd data
004_H	TB3	R/W	xxxx	Read table 4th data
005_H	DPL	R/W	xxxx	Data pointer low nibble data
006_H	DPM	R/W	xxxx	Data pointer middle nibble data
007_H	DPH	R/W	xxxx	Data pointer high nibble data

§ Peripheral registers:

Address	Symbol	R/W	Default	Description
008_H	PS	R/W	-100	CPU power saving control register
009_H	PSP	R/W	0---	Peripheral power saving control register
$00A_H$	INTC	R/W	0000	Interrupt enable control register
$00B_H$	INTF	R/W	0000	Interrupt request flag register
$00C_H$	INTC1	R/W	0000	Extended interrupt enable control register
$00D_H$	INTF1	R/W	0000	Extended interrupt request flag register
$00E_H$	PWMC	R/W	-000	PWM control register
$00F_H$	PWM0L	R/W	xxxx	PWM0 duty low nibble data register
010_H	PWM0M	R/W	xxxx	PWM0 duty middle nibble data register
011_H	PWM0H	R/W	xxxx	PWM0 duty high nibble data register
012_H	PAC	R/W	-111	IO port A control register
013_H	PA	R/W	-111	IO port A output data register
014_H	PBC	R/W	1111	IO port B control register
015_H	PB	R/W	1111	IO port B output data register
016_H	PCC	R/W	1111	IO port C control register
017_H	PC	R/W	1111	IO port C output data register
018_H	PDC	R/W	1111	IO port D control register
019_H	PD	R/W	1111	IO port D output data register
$01A_H$	PEC	R/W	1111	IO port E control register
$01B_H$	PE	R/W	1111	IO port E output data register
$01C_H$	PFC	R/W	1111	IO port F control register
$01D_H$	PF	R/W	1111	IO port F output data register
$01E_H$	TPINTC	R/W	00--	Touch pad interrupt enable control register
$01F_H$	TPINTF	R/W	00--	Touch pad interrupt request flag register

200_H	TCP1C	R/W	0000	TCP1 Timer/counter control register
201_H	TCP1L	R/W	xxxx	TCP1 Timer/counter data low register
202_H	TCP1H	R/W	xxxx	TCP1 Timer/counter data high register
203_H	TCP2C	R/W	0000	TCP2 Timer/counter control register
204_H	TCP2L	R/W	xxxx	TCP2 Timer/counter data low register
205_H	TCP2M	R/W	xxxx	TCP2 Timer/counter data middle register
206_H	TCP2H	R/W	xxxx	TCP2 Timer/counter data high register
207_H	PAI	R	----	Port A pad data reading address
208_H	PBI	R	----	Port B pad data reading address
209_H	PCI	R	----	Port C pad data reading address
$20A_H$	PDI	R	----	Port D pad data reading address
$20B_H$	PEI	R	----	Port E pad data reading address
$20C_H$	PFI	R	----	Port F pad data reading address
$20D_H$	TCPFS	R/W	-000	TCP clock source FS pre-scale register
$20E_H$	TBC	R/W	1111	Time base control register
$20F_H$	-	-	----	-
210_H	TPCHS0	R/W	0000	Touch pad channel selector register 0
211_H	TPCHS1	R/W	0000	Touch pad channel selector register 1
212_H	TPCHS2	R/W	0000	Touch pad channel selector register 2
213_H	TPCHS3	R/W	0000	Touch pad channel selector register 3
214_H	TPCTL	R/W	-000	Touch pad control register
215_H	TPCT0	R/W	1111	Touch pad Duty counter 1st nibble
216_H	TPCT1	R/W	1111	Touch pad Duty counter 2nd nibble
217_H	TPCT2	R/W	1111	Touch pad Duty counter 3rd nibble
218_H	CSAL	R/W	0000	Touch pad C load low nibble
219_H	CSAH	R/W	--00	Touch pad C load high nibble
$21A_H$	MCKS	R/W	-111	Modulation clock selector register
$21B_H$	SPCON0	R/W	0000	Special control register 0
$21C_H$	SPCON1	R/W	0000	Special control register 1
$21D_H$	SPCON2	R/W	--00	Special control register 2
$21E_H$	LDOFLAG	R/W	---0	LDO fail flag
$21F_H$	ODATA	R/W	0000	Touch pad output register for special function
220_H	OSCHADJ	R/W	0001	OSCH frequency adjustment register

221_H	IICCON0	R/W	---1	IIC control register 0
222_H	IICCON1	R/W	0000	IIC control register 1
223_H	IICSTS	R/W	0001	IIC status register
224_H	IICDATL	R/W	xxxx	IIC data low nibble register
225_H	IICDATH	R/W	xxxx	IIC data high nibble registe
226_H	IICRDATL0	R/W	0000	IIC fast read data low nibble register 0
227_H	IICRDATH0	R/W	0000	IIC fast read data high nibble register 0
228_H	IICRDATL1	R/W	0000	IIC fast read data low nibble register 1
229_H	IICRDATH1	R/W	0000	IIC fast read data high nibble register 1
$22A_H$	PWM1L	R/W	xxxx	PWM1 duty low nibble data register
$22B_H$	PWM1M	R/W	xxxx	PWM1 duty middle nibble data register
$22C_H$	PWM1H	R/W	xxxx	PWM1 duty high nibble data register
$22D_H$	PWM2L	R/W	xxxx	PWM2 duty low nibble data register
$22E_H$	PWM2M	R/W	xxxx	PWM2 duty middle nibble data register
$22F_H$	PWM2H	R/W	xxxx	PWM2 duty high nibble data register
230_H	ADJSTAT	R	--11	Frequency Adjustment Status flag register
231_H	TBLDRL	R	0000	Time base preload register low nibble
232_H	TBLDRH	R	1000	Time base preload register high nibble
300_H	RESETF	R/W	0000	Reset flag
301_H	TBRB	W	----	Time base clear address
302_H	MRO	W	----	Mask option register enable address
303_H	CLRWDT	W	----	Clear WDT 2nd instruction
304_H	LVREN	R/W	---0	LVR enable control register

Note: a. Default means initial value after power on or reset.
b. R is "read" only, W is "write" only, R/W is both of "read" and "write".

§ System function description:

S-1: System Oscillator

The high speed oscillator is operated in built-in RC mode. It is fixed 4MHz (typical at VDD=5V).

S-2: Peripheral Oscillator

The low speed oscillator was built-in an internal RC oscillator that is for low power consumption consideration and fixed peripheral device timing control. Built-in RC oscillator and the frequency range between 11 KHz~21 KHz.

S-3: CPU clock

The CPU clock comes from system/peripheral oscillator which was controlled by H/L bit in PS register. The high speed operation frequency comes from system oscillator. The low speed operation frequency comes from peripheral oscillator.

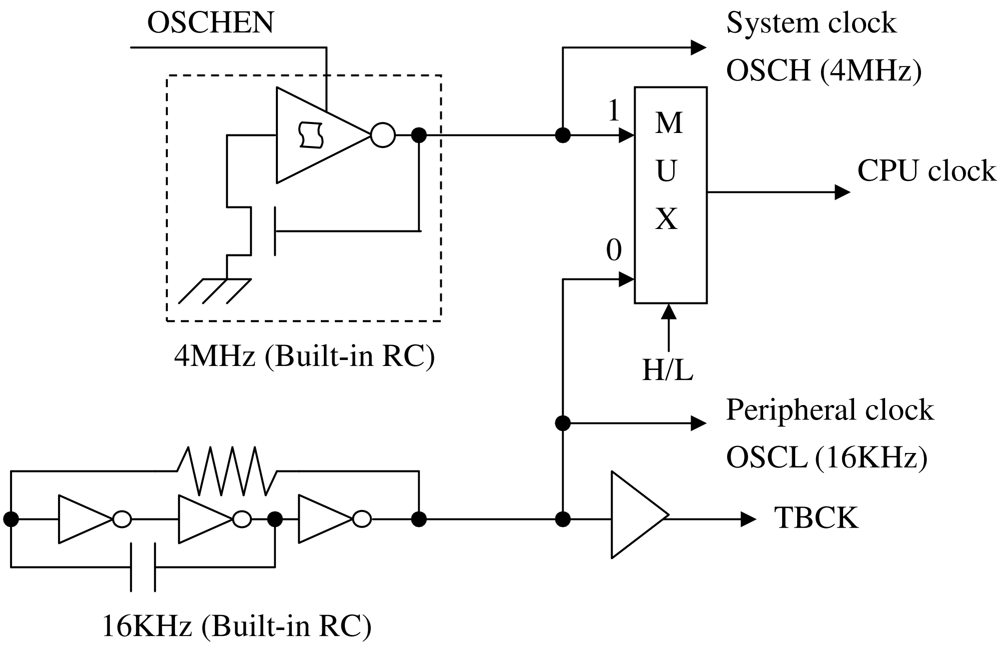


Figure: System/Peripheral Oscillator and CPU Clock Sources

S-4: Power saving mode (Stop mode and Sleep mode)

The CPU enters stop or sleep mode is operated by writing CPU power saving control register (PS). During the power saving mode, CPU holds the internal status of the system. In stop mode, the oscillator clocks will be stopped and system need a warm-up time for the stability of system clock running after wake up.

S-5: MCU System Operating Mode

The MCU has 4 operating modes, including high speed operation, low speed operation, sleep and stop modes. After power on reset, the MCU will go into high speed operation mode automatically. After wake up from sleep mode, the MCU will resume the last operation mode.

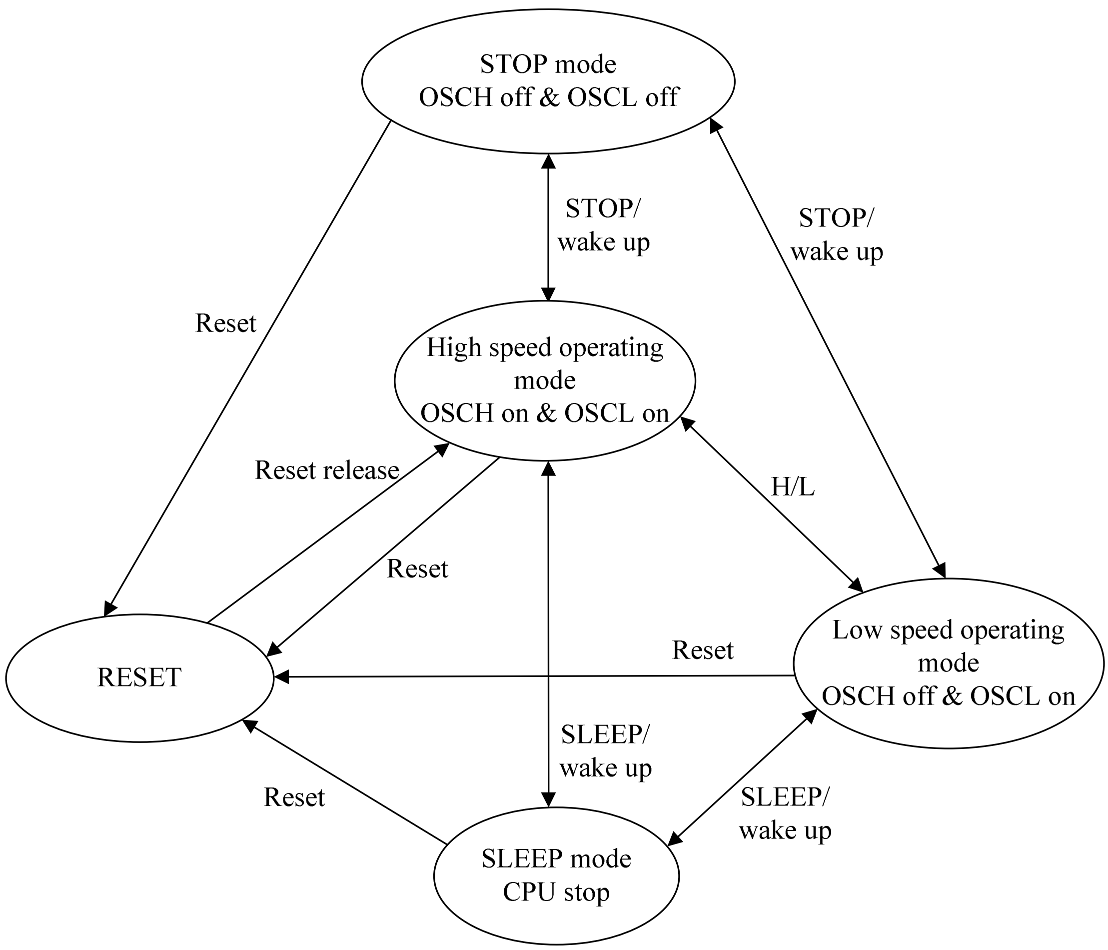


Figure: System Operation State Diagram

* Power saving mode condition and release

Modes	Stop mode	Sleep mode
High speed oscillator	Stopped	Stopped as H/L=0 Keep operating as H/L=1
Low speed oscillator	Stopped	Keep operating
CPU clock	Stopped	Stopped
CPU internal status	Stop and Retain the status	
Memory, Flag, Register, IO	Retain the status	
Program counter	Hold the next executed address	
Peripherals: Time bases, Timers, Interrupts	Stopped and Retain	Keep operating
Watch Dog Timer	Disable and cleared	
Release Condition	Reset, external INT sources, Input wake-up	Reset, internal and external INT sources, Input wake-up

S-6: Watch Dog Timer (WDT)

The clock of watch dog timer comes from time base 1st overflow output (TB1OV).User can use the time up signal to prevent a software malfunction or abnormal sequence from jumping to an unknown memory location causing a system fatal failure. Normally, if the watch dog timer time up signal active that will reset the chip. At the same time, program and hardware can be initialized and resume system under normal operation. The chip also provides 2 steps clear watch dog command as the programmer writes INTF with $F data first that will enable the WDT clear, and then writes CLRWDT register after. Completely finishes the two write steps will clear the watch dog timer. User should well arrange the two command steps for avoiding the dead lock loop.

User should keep in minds that always clear the WDT at main program and never clear the WDT in the interrupt routine.

The maximum period of WDT =(TB1OV cycle time) * 8

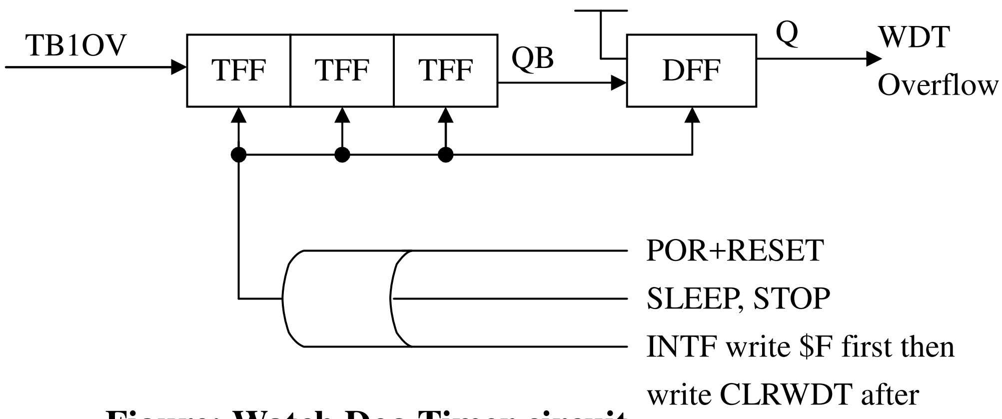


Figure: Watch Dog Timer circuit

S-7: Low Voltage Reset (LVR)

The low voltage reset (LVR) forces the MCU in reset state during power failure, especially as MCU working in AC power application, preventing from abnormal state is the key issue. The LVR voltage can be select 2.2V, 3.0V by mask option.

S-8: Reset

The chip has six kinds of reset sources: POR (power on reset), External reset, Watch dog timer reset, LVR (low voltage reset), Burn out reset and ROM fail reset. The reset feature can be divided into 2 kind groups that one is system reset and the other is CPU reset. The system reset will initialize the CPU and peripheral device with default state. The CPU reset only initializes the CPU state and keeps the peripheral state no change.

.POR (power on reset)

The chip provides automatically reset function when the power is turned on. The VDD should be below 0.5V and its rising slope (from 0.1VDD up to 0.9VDD) needs less than 10ms.

.External reset (RSTB)

This is one kind of system reset signal, but only forced externally. When the chip acknowledged the low level from the pin RSTB exceed 1 us, it will generate the reset procedure to reset CPU and all the peripheral back to their initial state (default values).

.Watch Dog Timer reset

The reset signal will generate automatically when the watch dog timer runs overflow. If the watch dog timer is cleared regularly by users' program, no watch dog timer reset will occur. Unless the MCU is forced into abnormal state, the software controlled procedure is disrupted and causing watch dog timer overflow, then it will generate reset signal to initializes the chip returning to normal operation.

.Low voltage reset (LVR)

The LVR function is used to monitor the supply voltage of MCU, it will generate a reset signal (with 4 OSCL de-bounce time) to reset the microcontroller as the VDD power falls below the default setting level V_{LVR}. It can also be enabled or disabled by programming LVREN bit in LVREN register. User writes $5 to LVREN register, LVREN bit is set to 1 and enable LVR function. User writes $A to LVREN register, LVREN bit is clear to 0 and disable LVR function. If user writes other value to LVREN register, it can't change LVREN bit.

.Burn out reset (Program sequence abnormal)

As CPU out of program area, the CPU can detect the abnormal condition and generate a system reset request.

.ROM fail reset

As ROM fail, the CPU can detect the abnormal condition and generate a system reset request.

✧ RESETF[300H]: Reset source flag register [R/W], power on value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	ROMF	BOF	LVRF	WDTF
Read/Write	R/W	R/W	R/W	R/W

WDTF: Watch dog timer overflow reset flag. (0: no active; 1: active)
LVRF: Low voltage reset flag. (0: no active; 1: active)
BOF: Burn out flag. (0: no active; 1: active)
ROMF: ROM fail flag. (0: no active; 1: active)
Note: The RESETF is only cleared by power on reset and external reset.

S-9. Power saving control register

✧ PS[008H]: Power saving control register [R/W], default value [-100]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	H/L	SLEEP	STOP
Read/write	-	R/W	R/W	R/W

STOP: Into stop mode. (0: disable; 1: enable)
SLEEP: Into sleep mode. (0: disable; 1: enable)
H/L: CPU clock source selector. (1: system clock; 0: peripheral clock)
When H/L=0, system clock oscillator is stopped.

When STOP bit is set to 1, system and peripheral clock oscillator are stopped. When H/L bit is set to 1, system clock oscillator is stopped. The SLEEP bit and STOP bit will be cleared to 0 automatically, when the release conditions occur from reset, interrupt or input wake up.

S-10. Special control register

✧ SPCON0 [21BH]: Special control register 0 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	CDSC2	CDSC1	CDSC0	VREFS
Read/write	R/W	R/W	R/W	R/W

VREFS: Voltage reference selector for touch sensor detection. (0: 1/2 VDD; 1: 2/3 VDD)
CDSC2~CDSC0: Charge and discharge sequence control for touch sensor function.

CDSC2~CDSC0	Sequence change clock
000	OFF
001	8
010	12
011	16
100	24
101	32
110	Reserve
111	Reserve

✧ SPCON1 [21CH]: Special control register 1 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	INTTS	FST2	FST1	FST0
Read/write	R/W	R/W	R/W	R/W

FST2~FST0: Frequency Shift Time selector.

INTTS: INT0 Interrupt input type selector. (0: Schmitt; 1: comparator)

Compare reference voltage use bandgap voltage 1.12V.

FST2~FST0	Frequency Shift Time (us)
000	OFF
001	4
010	8
011	16
100	32
101	64
110	128
111	256

✧ SPCON2 [21DH]: Special control register 2 [R/W], default value [--00]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	TPNIS	CSAMODE
Read/write	-	-	R/W	R/W

CSAMODE: CSA mode selector for touch pad scan.

0: C array as a touch pad capacitance compensation.

1: C array as a touch pad current compensation.

TPNIS: Touch detect circuit type selector.

0: TPNI use Schmitt trigger output signal.

1: TPNI use comparator output signal.

S-11. OST time

The system oscillator generates the system control timing for CPU core or peripheral devices with fixed control phase, so the waveform of oscillator becomes sensitive to noise, abnormal duty especially fatal for CPU. Any switching of clock source needs oscillation stable time (OST) to make sure the oscillation is stable and synchronized with CPU timing phase. The relative OST for different oscillator with reference value as below table:

OST	System clock(OSCH)	Peripheral clock(OSCL)
High speed STOP wakeup	-	8
Low speed STOP wakeup	-	8
High speed SLEEP wakeup	8	-
Low speed SLEEP wakeup	-	8
Low speed to High speed	-	8

✧ PSP[009H]: Peripheral power saving control register [R/W], default value [0---]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	LDOEN	-	-	-
Read/write	R/W	-	-	-

LDOEN: LDO enable. (0:disable; 1:enable)

The LDO voltage can be select 2.7V, 4.2V by mask option.

✧ LVREN[304H]: LVR enable control register [R/W], default value [---0]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	-	LVREN
Read/write	-	-	-	R/W

LVREN: Low voltage reset enable. (0:disable, 1:enable)

When write $5 to this address, LVREN is set to 1; write $A, LVREN is clear to 0.

LDOFLAG[21EH]: LDO flag register [R/W], default value [---0]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	-	LDOFAIL
Read/write	-	-	-	R/W

LDOFAIL: When VDD voltage is smaller than LDO voltage, LDOFAIL will be set. This bit can be clear by write 0.

S-12. Interrupts

The CPU provides only 1 interrupt vector ($001H) and no priority, but can expand to multi-sources. Interrupt source includes external interrupts (INT0,INT1), timer/counter interrupts (TCP1,TCP2), Time base timer interrupt (TBxINT) or other peripheral device interrupt request (PERINT). The interrupt control registers (INTC or INTC1) contain the interrupt control bit to enable and disable corresponding interrupt request and the corresponding interrupt request flags in the (INTF or INTF1) registers. Before finishing the INT service routine, another INT request will keep waiting until program return from interrupt routine.

If the interrupt request needs service, the programmer may set the corresponding INT enable bit to allow interrupt active. External interrupts are triggered by both falling and rising edge trigger and set the related interrupt request flag (INTFx). The internal timer/counter interrupt is setting the TCPxF to 1, resulting from the timer/counter overflow. The time base interrupt TBxINT was provided 2 periodic interrupt request cycles for user operating a periodic routine.

When the corresponding interrupt enable and flag bit is set to 1, the CPU will active the interrupt service routine. Then CPU reads the service flag and check the request priority then proceeds with the relative interrupt service. After CPU writes the corresponding bit to 0 in the INTFx register, the service flag will be cleared to 0(using STX #n,$m instruction). The INTF and INTF1 registers' bit can only write 0 to clear the flag. User writes 1 to flag bit with no effect.

INT0 input type can select Schmitt or comparator by SPCON1 register, if comparator select then the comparator reference voltage is the bandgap voltage(1.12+-10%), it will consumption more current than Schmitt because bandgap turn on. It can be used to detect VDD voltage for battery low and so on.

✧ INTC[00AH]: Interrupt control register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TB2IE	TCP2IE	TCP1IE	TB1IE
Read/Write	R/W	R/W	R/W	R/W

TB1IE: Enable time base 1st interrupt. (0: disable; 1: enable)
TCP1IE: Enable interrupt of TCP1 timer/counter. (0: disable; 1: enable)
TCP2IE: Enable interrupt of TCP2 timer/counter. (0: disable; 1: enable)
TB2IE: Enable time base 2nd interrupt. (0: disable; 1: enable)

✧ INTF[00BH]: Interrupt request flag register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TB2F	TCP2F	TCP1F	TB1F
Read/Write	R/W	R/W	R/W	R/W

TB1F: Time base 1st interrupt request flag. (0: inactive; 1: active)
TCP1F: TCP1 Timer/counter interrupt request flag. (0: inactive; 1: active)
TCP2F: TCP2 Timer/counter interrupt request flag. (0: inactive; 1: active)
TB2F: Time base 2nd interrupt request flag. (0: inactive; 1: active)

✧ INTC1[00CH]: Extended interrupt control register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	STIE	IICIE	INT1IE	INT0IE
Read/Write	R/W	R/W	R/W	R/W

INT0IE: Enable INT0 external interrupt. (0: disable; 1: enable)
INT1IE: Enable INT1 external interrupt. (0: disable; 1: enable)
IICIE: Enable IIC interrupt. (0: disable; 1: enable)
STIE: Enable IIC start signal interrupt. (0: disable; 1: enable)

✧ INTF1[00DH]: Extended interrupt request flag register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	STIF	IICF	INT1F	INT0F
Read/Write	R/W	R/W	R/W	R/W

INT0F: INT0 external interrupt request flag. (0: inactive; 1: active)
INT1F: INT1 external interrupt request flag. (0: inactive; 1: active)
IICF: IIC interrupt request flag. (0: inactive; 1: active)
STIF: IIC start signal interrupt request flag. (0: inactive; 1: active)

INTxS1~INTxS0	Trigger type
00	Low active
01	Falling edge
10	Rising edge
11	Dual edge trigger

Note: INTx Trigger type are selected by mask option.

✧ TPINTC[01EH]: Touch pad interrupt control register [R/W], default value [00--]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCTIE	TPCMPIE	-	-
Read/Write	R/W	R/W	-	-

TPCMPIE: Capacitor overcharge interrupt enable. (0: disable; 1: enable)

TPCTIE: Duty counter overflow interrupt enable. (0: disable; 1: enable)

✧ TPINTF[01FH]: Touch pad request flag register [R/W], default value [00--]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCTF	TPCMPF	-	-
Read/Write	R/W	R/W	-	-

TPCMPF: Capacitor overcharge flag. (0: inactive; 1: active)

TPCTF: Duty counter overflow flag. (0: inactive; 1: active)

§ Peripheral function description:

P-1: System clock pre-scale

The system clock is the most high frequency of MCU. For various peripherals, application needs different clock source divided from system clock. TCPFS register is a selector for choosing suitable frequency (FS).

✧ TCPFS[20DH]: System clock pre-scale register [R/W], default value [-000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	FS2	FS1	FS0
Read/Write	-	R/W	R/W	R/W

FS2~FS0: The selector of TCPFS.

FS2~FS0	FS	FS2~FS0	FS
0	OSCH/1	4	OSCH/16
1	OSCH/2	5	OSCH/32
2	OSCH/4	6	OSCH/64
3	OSCH/8	7	OSCH/128

P-1-1: OSC Frequency Adjustment

✧ ADJSTAT[230H]: Frequency Adjustment Status flag register [R], default value [--11]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	OSCHADJF	TBADJF
Read/Write	-	-	R	R

TBADJF: Time base adjustment status flag. (0: busy, 1: idle)

OSCHADJF: OSCH frequency adjustment status flag. (0: busy, 1: idle)

✧ SPCON1 [21CH]: Special control register 1 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	INTTS	FST2	FST1	FST0
Read/write	R/W	R/W	R/W	R/W

FST2~FST0: Frequency Shift Time selector.

FST2~FST0	Frequency Shift Time (us)
000	OFF
001	4
010	8
011	16
100	32
101	64
110	128
111	256

✧ OSCHADJ [220H]: OSCH frequency adjustment register [R/W], default value [0001]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	ADJ3	ADJ2	ADJ1	ADJ0
Read/write	R/W	R/W	R/W	R/W

ADJ3~ADJ0: OSCH frequency adjustment data.

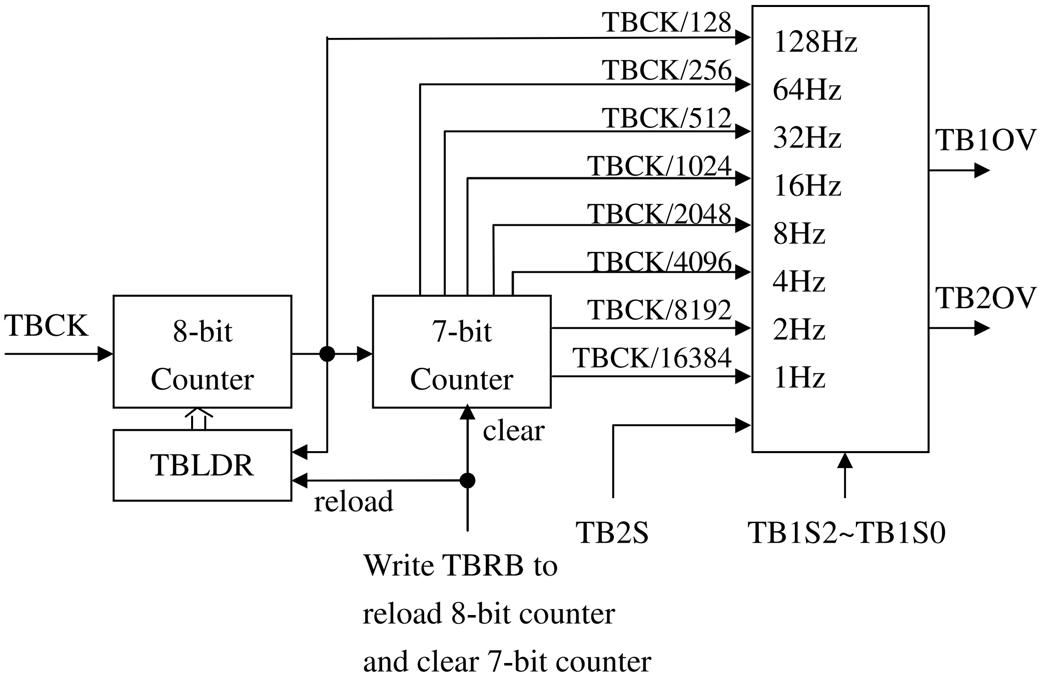
OSCHADJ set the frequency swing range, when the change time in register

FST is set, frequency shift function will be activated. Swing shift back and forth from the center frequency. Set FST=0 to off frequency shift function. When frequency shift function is executing, OSCHADJF will be set 0. Then user can not change OSCHADJ and FST, but only can be set 0 to off function. Frequency shift function will not immediately stop, when FST is set to 0, the need to wait until the frequency back to the original frequency, while OSCHADJF will be set to 1.

OSCHADJ use ranges from 1 to 8, do not use the value out of range.

P-2: Time Base

The time base has 2 interrupt sources and both of them come from the peripheral internal RC oscillator. The time base 1st overflow output (TB1OV) can cause interrupt and the period is selected by TB1S2~TB1S0 in TBC register. The time base 2nd overflow output (TB2OV) also offers two sample frequency options by TB2S bit in the TBC register.



✧ TBC[20EH]: Time base control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TB2S	TB1S2	TB1S1	TB1S0
Read/Write	R/W	R/W	R/W	R/W

TB1S2~TB1S0: Time base 1st overflow frequency selector.

TB2S: Time base 2nd overflow frequency selector.

Note: Every time writing the TBRB will clear the time base.

TB1S2	**TB1S1**	**TB1S0**	**Time Base overflow frequency (TB1OV)**	**TB1OV**
0	0	0	TBCK/128	128HZ
0	0	1	TBCK/256	64HZ
0	1	0	TBCK/512	32HZ
0	1	1	TBCK/1024	16HZ
1	0	0	TBCK/2048	8HZ
1	0	1	TBCK/4096	4HZ
1	1	0	TBCK/8192	2HZ
1	1	1	TBCK/16384	1HZ

TB2S	**Time Base overflow frequency (TB2OV)**	**TB2OV**
0	TBCK/512	32Hz
1	TBCK/1024	16Hz

Note: TB1OV select 128Hz can not be use for TCP1 clock source, the TCP1 will not work. Please use other time base select option.

P-2-1: Adjustment Time base

✧ ADJSTAT[230H]: Frequency Adjustment Status flag register [R], default value [--11]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	OSCHADJF	TBADJF
Read/Write	-	-	R	R

TBADJF: Time base adjustment status flag. (0: busy, 1: idle)

OSCHADJF: OSCH frequency adjustment status flag. (0: busy, 1: idle)

✧ TBLDRL[231H]: Time base preload register low nibble [R], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TBLDR3	TBLDR2	TBLDR1	TBLDR0
Read/Write	R	R	R	R

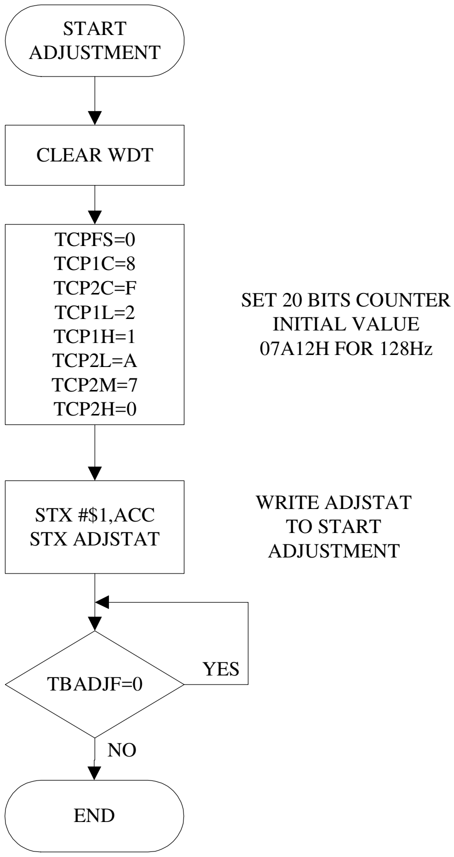
TBLDR3~TBLDR0: Time base preload register low nibble data.

✧ TBLDRH[232H]: Time base preload register low nibble [R], default value [1000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TBLDR7	TBLDR6	TBLDR5	TBLDR4
Read/Write	R	R	R	R

TBLDR7~TBLDR4: Time base preload register high nibble data.

User can adjustment the time base for accurate 128Hz by modify first 8-bit counter preload value, the time base preload counter initial value is 80H in power on, the adjustment procedure will modify the preload counter value to approach 128Hz, there is using TCP1 and TCP2 cascaded to form a 20-bit timer/counter, chooses clock source FS and set TCPFS=0 for TCP1, then load 07A12H(4MHz/31250=128Hz) to the 20-bit counter and write 1H to ADJSTAT register to start adjustment, then check TBADJF flag. It is finished when TBADJF=1. The adjustment procedure flow chart as follow:



P-3: 8-bit Timer/Counter for TCP1

One 8-bit timer/counter (TCP1) with 4 kind clock sources and preload data buffer can implement as a timer or counter feature. The clock sources of TCP1 are selected by TCP1S1~TCP1S0 of TCP1 control register (TCP1C). TCP1OV is the timer or counter overflow signal and the rising edge will set the relative INT flag.

✧ TCP1C[200H]: TCP1 Timer/counter control register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP1LD	TCP1S1	TCP1S0	TCP1EN
Read/Write	R/W	R/W	R/W	R/W

TCP1EN: TCP1 counting enable. (0: disable; 1: enable)
TCP1LD: TCP1 auto-reload enable. (0: disable; 1: enable)
TCP1S1~TCP1S0: TCP1 clock source selector.

TCP1S1	TCP1S0	Selected Clock source
0	0	FS
0	1	TCP1I
1	0	TBCK
1	1	TB1OV

Note: TB1OV select 128Hz can not be use for TCP1 clock source, the TCP1 will not work. Please use other time base select option.

✧ TCP1L[201H]: TCP1 low nibble data register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP1_3/TCP1D3	TCP1_2/TCP1D2	TCP1_1/TCP1D1	TCP1_0/TCP1D0
Read/Write	R/W	R/W	R/W	R/W

TCP1_3~TCP1_0: Reading TCP1 counter low nibble data.
TCP1D3~TCP1D0: Writing TCP1D low nibble of data buffer.

✧ TCP1H[202H]: TCP1 high nibble data register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP1_7/TCP1D7	TCP1_6/TCP1D6	TCP1_5/TCP1D5	TCP1_4/TCP1D4
Read/Write	R/W	R/W	R/W	R/W

TCP1_7~TCP1_4: Reading TCP1 counter high nibble data.
TCP1D7~TCP1D4: Writing TCP1D high nibble of data buffer.

* TCP1D: Like a 8-bit TCP1 data register [R/W], default value [00H]

TCP1D	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP1D7	TCP1D6	TCP1D5	TCP1D4	TCP1D3	TCP1D2	TCP1D1	TCP1D0

The special R/W function for TCP1 has different Target, AS writing TCP1H/L registers that are updating preload data of the TCP1D. As read TCP1H/L registers that are the brand new TCP1 counter value.

P-4: 12-bit Timer/Counter/PWM for TCP2

One 12-bit timer/counter (TCP2) with 4 kind clock sources and preload data buffer can implement as a timer or counter feature. The clock sources of TCP2 are selected by TCP2S1~TCP2S0 of TCP2 control register (TCP2C). TCP2OV is the timer or counter overflow signal and the rising edge will set the relative INT flag.

✧ TCP2C[203H]: TCP2 Timer/counter/PWM control register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP2LD	TCP2S1	TCP2S0	TCP2EN
Read/Write	R/W	R/W	R/W	R/W

TCP2EN: TCP2 counting enable. (0: disable; 1: enable)
TCP2LD: TCP2 auto-reload enable. (0: disable; 1: enable)
TCP2S1~TCP2S0: TCP2 clock source selector.

TCP2S1	TCP2S0	Selected Clock source
0	0	FS
0	1	OSCH
1	0	TBCK
1	1	TCP1OV

✧ TCP2L[204H]: TCP2 low nibble data register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP2_3/TCP2D3	TCP2_2/TCP2D2	TCP2_1/TCP2D1	TCP2_0/TCP2D0
Read/Write	R/W	R/W	R/W	R/W

TCP2_3~TCP2_0: Reading TCP2 counter low nibble data.
TCP2D3~TCP2D0: Writing TCP2D low nibble of data buffer.

✧ TCP2M[205H]:TCP2 middle nibble data register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP2_7/TCP2D7	TCP2_6/TCP2D6	TCP2_5/TCP2D5	TCP2_4/TCP2D4
Read/Write	R/W	R/W	R/W	R/W

TCP2_7~TCP2_4: Reading TCP2 counter middle nibble data.
TCP2D7~TCP2D4: Writing TCP2D middle nibble of data buffer.

✧ TCP2H[206H]: TCP2 high nibble data register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP2_11/TCP2D11	TCP2_10/TCP2D10	TCP2_9/TCP2D9	TCP2_8/TCP2D8
Read/Write	R/W	R/W	R/W	R/W

TCP2_11~TCP2_8: Reading TCP2 counter high nibble data.
TCP2D11~TCP2D8: Writing TCP2D high nibble of data buffer.

* TCP2D: Like a 12-bit TCP2 data register [R/W], default value [000H]

TCP2D	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP2D11	TCP2D10	TCP2D9	TCP2D8	TCP2D7	TCP2D6	TCP2D5	TCP2D4	TCP2D3	TCP2D2	TCP2D1	TCP2D0

The special R/W function for TCP2 has different Target, AS writing TCP2H/M/L registers that are updating preload data of the TCP2D. As read TCP2H/M/L registers that are the brand new TCP2 counter value.

.Timer

When TCPx works as a Timer, user needs give the preload data TCPxD for periodic interrupt. After initial setting, user starts the TCPx counting by setting.

When 8-bit TCP1 timer/counter:
TCP1EN=1, the TCP1 cycle period is:
Tc = (selected clock cycle) * (256) if TCP1D=00H
Tc = (selected clock cycle) * (TCP1D) otherwise

When 12-bit TCP2 timer/counter:
TCP2EN=1, the TCP2 cycle period is:
Tc = (selected clock cycle) * (4096) if TCP2D=000H
Tc = (selected clock cycle) * (TCP2D) otherwise

When 20-bit timer/counter:
Tc = (selected clock cycle) * (1048576) if TCP1D=00H and TCP2D=000H
Tc = (selected clock cycle) * (TCP2D*256+TCP1D) otherwise

When user writes data to the TCPxH/M/L, the data just keep in TCPxH/M/L latch. During the TCPxEN=1 command executed, the TCPxH/M/L latch's complement value will load into counter TCPxH/M/L as initial value and start the timer function. Necessary TCPxLD=1, timer run with reload feature as TCPx up counts and reaches the value of FF_H or 255 for TCP1 or value of FFF_H or 4095 for TCP2. At the same time, interrupt request flag TCPxF will set activated, if software enables the corresponding interrupt enable bit, INT hardware will cause MCU interrupt service routine.

.Counter

Counter feature is implemented only by TCPxLD=0, the TCPxD can be zero or not that depends on software needs. User starts and stops the counter by changing the TCPxEN bit value. On the save side, reading the counter value after stopping the count by disable TCPxEN=0, if reading the counter value during value changing that means clock in happening at the same time. The reading of counter value may disrupt for transient state. If 8-bit counter is not enough for counting, user can enable the interrupt and using the data RAM as software counter for extending the counter stage.

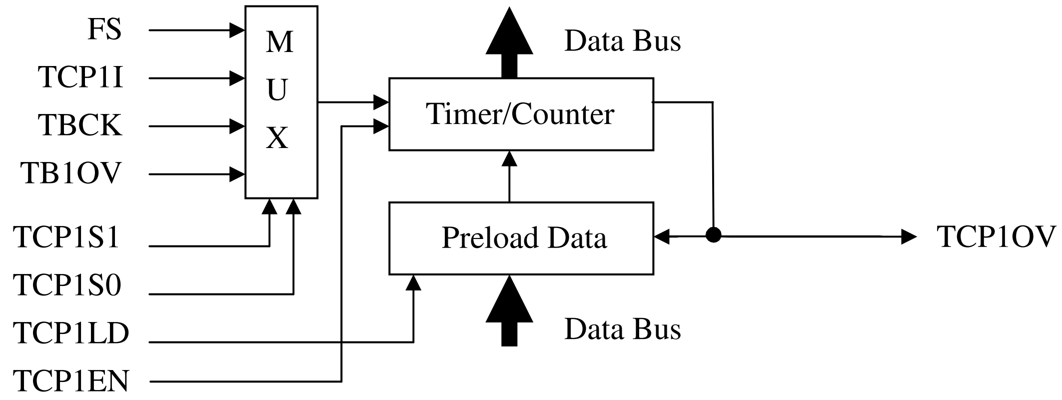


Figure: 8-bit Timer/Counter (TCP1)

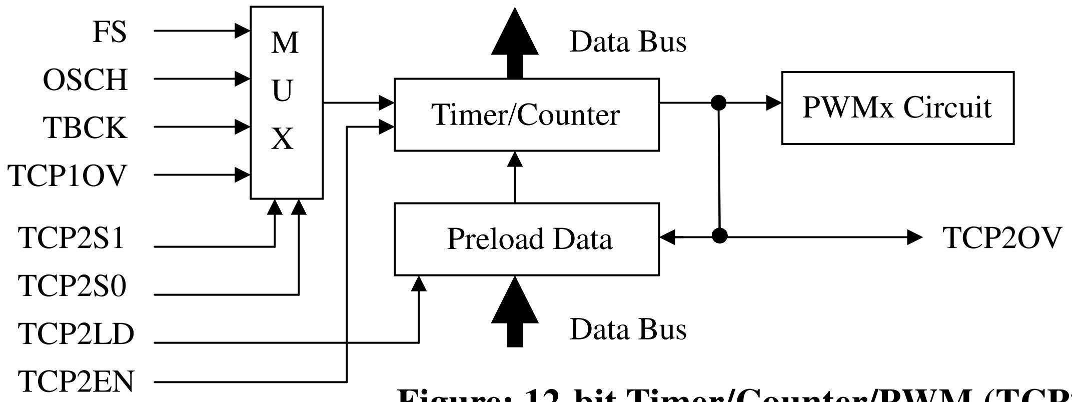


Figure: 12-bit Timer/Counter/PWM (TCP2)

TCP1S1	TCP1S0	**TCP1**
0	0	**FS**
0	1	**TCP1I**
1	0	**TBCK**
1	1	**TB1OV**

	PWM Output
TCP2	PWM0,1,2

TCP2S1	TCP2S0	**TCP2**
0	0	**FS**
0	1	**OSCH**
1	0	**TBCK**
1	1	**TCP1OV**

FS: System scaled frequency.
TCP1I: External clock input (falling edge).
TBCK: Peripheral clock source, 16KHz in the RC mode.
TB1OV: Time base 1st overflow output.
OSCH: System clock source, 4MHz in the RC mode.
TCP1OV: TCP1 overflow output.
PWM0,1,2: TCP2 cycle time with PWMxD duty output signal.

P-5: 20-bit Timer/Counter (TCP1 and TCP2 cascade)

Two sets TCP can be cascaded to form a 20-bit timer/counter when TCP2 chooses TCP1OV as clock source (TCP2S1=1 and TCP2S0=1). In the 20-bit timer application, user should use TCP1EN to control the starting or stopping counting of 20-bit timer/counter, data load is controlled by writing TCP1EN=1. The rising TCP2OV will reload the contents in the pre-load register into timer/counter, if TCP2LD in TCP2C are enabled. The interrupt feature is different, in this case, the TCP1 INT will be inhibit when TCP1OV occur, the TCP2 INT is normally.

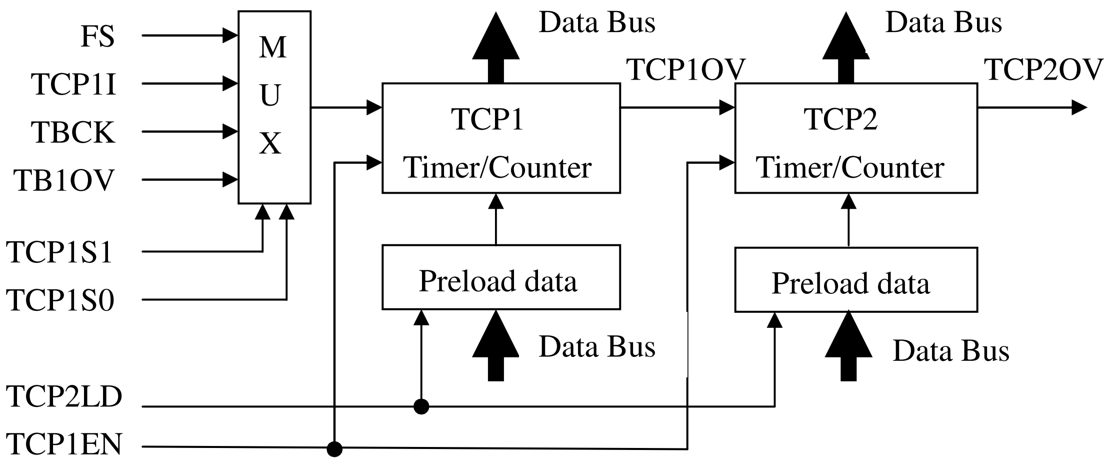


Figure: 20-bit Timer/Counter (TCP1 and TCP2 cascade)

.PWM

The PWM period generated from TCP2. When PWMxEN (PWMC<0~2>) enable, and PWMOUT pin (PA0~PA2,PB0~PB2 must be output mode and select PWM function pin and normal IO by mask option) change to output mode, PWMx signal will output to PWMOUT pin. If TCP2 is running, set PWMxEN=1 will not execute until TCP2OV occur.

The duty of PWMx value is store in PWMxL, PWMxM and PWMxH, user write PWMxH and PWMxM first, last write PWMxL. When write the PWMxL the 12-bit duty value will be load to PWMxD at the same time. PWM's duty value cannot bigger than TCP2 pre-load data. If not, PWMOUT is an unexpected signal.

User can select PWMOUT pin start with 1 or start with 0 by mask option. When TCP2 enable, timer start increment, if timer/counter value bigger than PWM's duty value, PWMOUT will change state. The PWMOUT back to start state, When TCP2 is overflow.

User does not use PWM in 20-bit timer/counter mode. If not, PWMOUT is an unexpected signal.

User does not use TCP2D=000H. If not, PWMOUT is an unexpected signal.

✧ PWMC[00EH]: PWM control register [R/W], default value [-000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	PWM2EN	PWM1EN	PWM0EN
Read/Write	-	R/W	R/W	R/W

PWM0EN: PWM0 output enable. (0: disable; 1: enable)
PWM1EN: PWM1 output enable. (0: disable; 1: enable)
PWM2EN: PWM2 output enable. (0: disable; 1: enable)

✧ PWM0L[00FH]: PWM0 duty low nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM0D3	PWM0D2	PWM0D1	PWM0D0
Read/Write	R/W	R/W	R/W	R/W

PWM0D3~PWM0D0: PWM0 duty low nibble data.

✧ PWM0M[010H]: PWM0 duty middle nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM0D7	PWM0D6	PWM0D5	PWM0D4
Read/Write	R/W	R/W	R/W	R/W

PWM0D7~PWM0D4: PWM0 duty middle nibble data.

✧ PWM0H[011H]: PWM0 duty high nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM0D11	PWM0D10	PWM0D9	PWM0D8
Read/Write	R/W	R/W	R/W	R/W

PWM0D11~PWM0D8: PWM0 duty high nibble data.

✧ PWM1L[22AH]: PWM1 duty low nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM1D3	PWM1D2	PWM1D1	PWM1D0
Read/Write	R/W	R/W	R/W	R/W

PWM1D3~PWM1D0: PWM1 duty low nibble data.

✧ PWM1M[22BH]: PWM1 duty middle nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM1D7	PWM1D6	PWM1D5	PWM1D4
Read/Write	R/W	R/W	R/W	R/W

PWM1D7~PWM1D4: PWM1 duty middle nibble data.

✧ PWM1H[22CH]: PWM1 duty high nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM1D11	PWM1D10	PWM1D9	PWM1D8
Read/Write	R/W	R/W	R/W	R/W

PWM1D11~PWM1D8: PWM1 duty high nibble data.

✧ PWM2L[22DH]: PWM2 duty low nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM2D3	PWM2D2	PWM2D1	PWM2D0
Read/Write	R/W	R/W	R/W	R/W

PWM2D3~PWM2D0: PWM2 duty low nibble data.

✧ PWM2M[22EH]: PWM2 duty middle nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM2D7	PWM2D6	PWM2D5	PWM2D4
Read/Write	R/W	R/W	R/W	R/W

PWM2D7~PWM2D4: PWM2 duty middle nibble data.

✧ PWM2H[22FH]: PWM2 duty high nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM2D11	PWM2D10	PWM2D9	PWM2D8
Read/Write	R/W	R/W	R/W	R/W

PWM2D11~PWM2D8: PWM2 duty high nibble data.

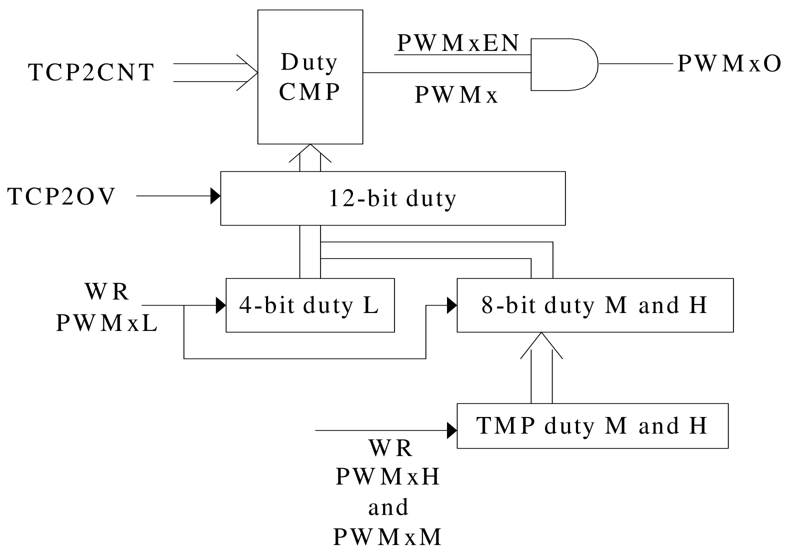


Figure: PWM (TCP2)

PWMxD	PWM duty	Note
0	(0 * clock cycle) / TCP2 timer's period	All off
1	(1 * clock cycle) / TCP2 timer's period	
2	(2 * clock cycle) / TCP2 timer's period	
……	……	
n	((n) * clock cycle) / TCP2 timer's period	
……	……	
TCP2D	((TCP2D) * clock cycle) / TCP2 timer's period	All on

Note: 1. PWMxD can not bigger than TCP2D

2. TCP2 timer's period = (TCP2D) * clock cycle.

3. PWM can start 0 or start 1 by mask option.

Table: PWM duty

P-6: IIC Module

IIC is a 2-wire, bi-directional serial bus, which provides a simple, efficient way for data exchange between devices. This two-wire bus minimizes the interconnection between devices and eliminates the need for address decoders.

IIC CONTROL

Every IIC device must have independent slave address. User can use IICCON1<3:1> (ADR<2:0>) to select one independent slave address, the map address reference follow table.

MODE	ADR2	ADR1	ADR0	IIC device address 7-bit
Normal mode	0	0	0	101 0000 (50H)
	0	0	1	101 0001 (51H)
	0	1	0	101 0010 (52H)
	0	1	1	101 0011 (53H)
	1	0	0	101 0100 (54H)
	1	0	1	101 0101 (55H)
	1	1	0	101 0110 (56H)
	1	1	1	101 0111 (57H)
MODE	ADR2	ADR1	ADR0	IIC device address 8-bit
Fast read mode	0	0	0	1010 0001 (A1H)
	0	0	1	1010 0011 (A3H)
	0	1	0	1010 0101 (A5H)
	0	1	1	1010 0111 (A7H)
	1	0	0	1010 1001 (A9H)
	1	0	1	1010 1011 (ABH)
	1	1	0	1010 1101 (ADH)
	1	1	1	1010 1111 (AFH)

Table: IIC address mapping table

Set IICCON1<0> (IICEN) can enable all IIC block, IICCON0<0> (IICMOD) can select IIC operation in normal mode or fast read mode. The first byte of data transfer immediately following the START signal is the slave address transmitted by the master. This is a seven bit long calling address followed by a R/W bit in Normal mode. Fast read mode only have transmit mode, so R/W bit must be 1.

When START signal is detected, IICSTS<2> (MBB) is set. When STOP signal is detected, MBB is cleared. IICSTS<3> (MAASF) is set, when IIC device match the calling address. When MAASF is set, and INTF1<2> (IICF) is also set. An interrupt is generated if the INTC1<2> (IICIE) be set. User can check IICSTS<1> (SRWB) to know IIC device operate in transmit or receive mode. When IICF is set, an interrupt is generated to the CPU. IICF is set when one of the following event occurs:

1) IIC device address match in normal mode.
2) Completion of one byte of data transfer. It is set at the falling edge of the 9th clock in normal mode.
3) Completion of IICRDAT(L/H)0 data transfer and IICRDAT(L/H)1 load on IICDAT(L/H) in fast read mode.

When IIC device operate in transmit mode, master's acknowledge store in IICSTS<0> (TXACK). If detect acknowledge, this bit set, if not, this bit clear.

IICDAT(L/H) only use in normal mode. In transmit mode, data written into the register to send to the bus automatically, with the most significant bit out first. In receive mode, reading of this register initiates receiving of the next byte data. IICRDAT(L/H)0 and IICRDAT(L/H)1 only use in fast mode. IIC device transmit IICRDAT(L/H)0 first, completion of transfer, IICRDAT(L/H)1 continue transmit to bus automatically.

Whenever IICRDAT(L/H)0 transfer is complete, IICRDAT(L/H)1 will automatically load transfer buffer, and generates an interrupt flag notify updatable. If the data transfer will be more than two bytes, you can create a data counter, interrupt flag is generated every time, data will be placed in IICRDAT(L/H)0 and IICRDAT(L/H)1 by order by software, however, if the master halfway want to re-read the beginning of the data, when the data counter is not starting from scratch, data can not be read correctly, this situation can be set STIE be 1, when receiving the START signal from IIC BUS, an interrupt is generated, by receiving this start signal, the data counter is reset by the software, so you can re-read the data correctly. STIF can only be actived in fast read mode.

✧ INTC1[00CH]: Extended interrupt control register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	STIE	IICIE	INT1IE	INT0IE
Read/Write	R/W	R/W	R/W	R/W

INT0IE: enable INT0 external interrupt. (0: disable; 1: enable)

INT1IE: enable INT1 external interrupt. (0: disable; 1: enable)

IICIE: enable IIC interrupt. (0: disable; 1: enable)

STIE: enable IIC start signal interrupt. (0: disable; 1: enable)

✧ INTF1[00DH]: Extended interrupt request flag register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	STIF	IICF	INT1F	INT0F
Read/Write	R/W	R/W	R/W	R/W

INT0F: INT0 external interrupt request flag. (0: inactive; 1: active)

INT1F: INT1 external interrupt request flag. (0: inactive; 1: active)

IICF: IIC interrupt request flag. (0: inactive; 1: active)

STIF: IIC start signal interrupt request flag. (0: inactive; 1: active)

✧ IICCON0[221H]: IIC control register 0 [R/W], default value [---1]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	-	IICMOD
Read/Write	-	-	-	R/W

IICMOD: IIC Operation mode. (0: normal; 1: fast read)

✧ IICCON1[222H]: IIC control register 1 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	ADR2	ADR1	ADR0	IICEN
Read/Write	R/W	R/W	R/W	R/W

IICEN: IIC function enable. (0: disable; 1: enable)

ADR2~ADR0: IIC slave address.

✧ IICSTS[223H]: IIC status register [R/W], default value [0001]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	MAASF	MBB	SRWB	TXACK
Read/Write	R/W	R	R	R

TXACK: receive master acknowledge. (0: master don't send acknowledge; 1: master send acknowledge)

SRWB: IIC slave read or write select. (0: write data to slave; 1: read data from slave)

MBB: IIC bus busy flag. (0: IIC bus idle; 1: IIC bus busy)

MAASF: IIC slave address match flag. (0: not match; 1: match(must clear by software))

✧ IICDATL[224H]: IIC data low nibble register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	DAT3	DAT2	DAT1	DAT0
Read/Write	R/W	R/W	R/W	R/W

DAT3~DAT0: IIC low nibble data.

✧ IICDATH[225H]: IIC data high nibble register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	DAT7	DAT6	DAT5	DAT4
Read/Write	R/W	R/W	R/W	R/W

DAT7~DAT4: IIC high nibble data.

✧ IICRDATL0[226H]: IIC fast read data low nibble register 0 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	RDAT03	RDAT02	RDAT01	RDAT00
Read/Write	R/W	R/W	R/W	R/W

RDAT03~RDAT00: IIC fast read low nibble data 0.

✧ IICRDATH0[227H]: IIC fast read data high nibble register 0 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	RDAT07	RDAT06	RDAT05	RDAT04
Read/Write	R/W	R/W	R/W	R/W

RDAT07~RDAT04: IIC fast read high nibble data 0.

✧ IICRDATL1[228H]: IIC fast read data low nibble register 1 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	RDAT13	RDAT12	RDAT11	RDAT10
Read/Write	R/W	R/W	R/W	R/W

RDAT13~RDAT10: IIC fast read low nibble data 1.

✧ IICRDATH1[229H]: IIC fast read data high nibble register 1 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	RDAT17	RDAT16	RDAT15	RDAT14
Read/Write	R/W	R/W	R/W	R/W

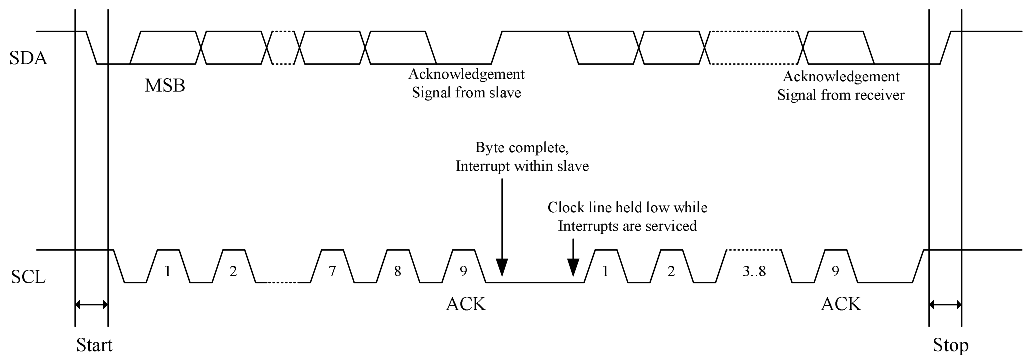
RDAT17~RDAT14: IIC fast read high nibble data 1.

.IIC operation

IIC support normal mode and fast read mode. Normal mode is compatible to standard IIC bus. Fast read mode is support fast read data, for reduce CPU wait time.

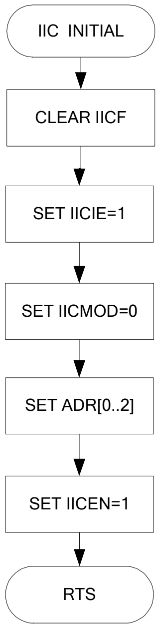
In normal mode, IIC device address match and completion of one byte of data transfer, interrupt will occur. It is set at the falling edge of the 9th clock in normal mode. if master read last data, master does not send acknowledge to IIC device, it can let IIC device know don't need to send data, and do dummy write for release data bus to avoid bus error.

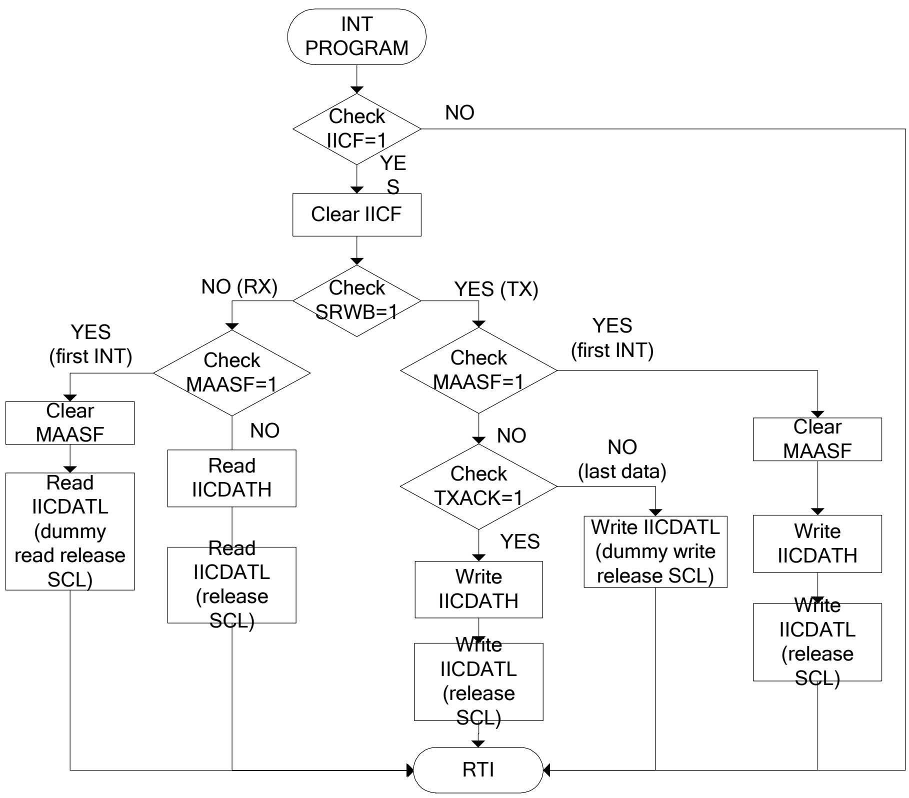
When an interrupt occurs, MCU must first determine the current mode will be receive or transmit, and then determine whether the command mode by MAASF flag. Due to the process by software, so before entering the interrupt will pull low the SCL, to notify the Master waits. In receive mode when reading IICDATL and the transfer mode when writing IICDATL, the SCL pull low will be release. The timing as follow:



1) Normal mode

Follow show a flowchart of IIC device program state machine in normal mode.

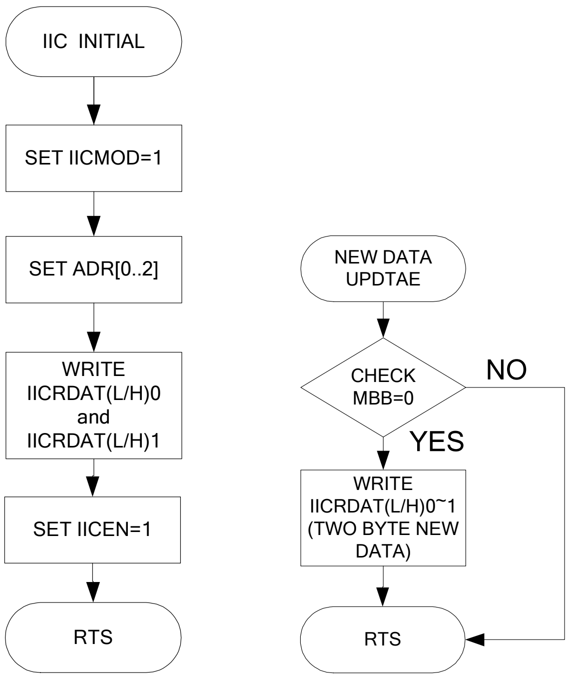




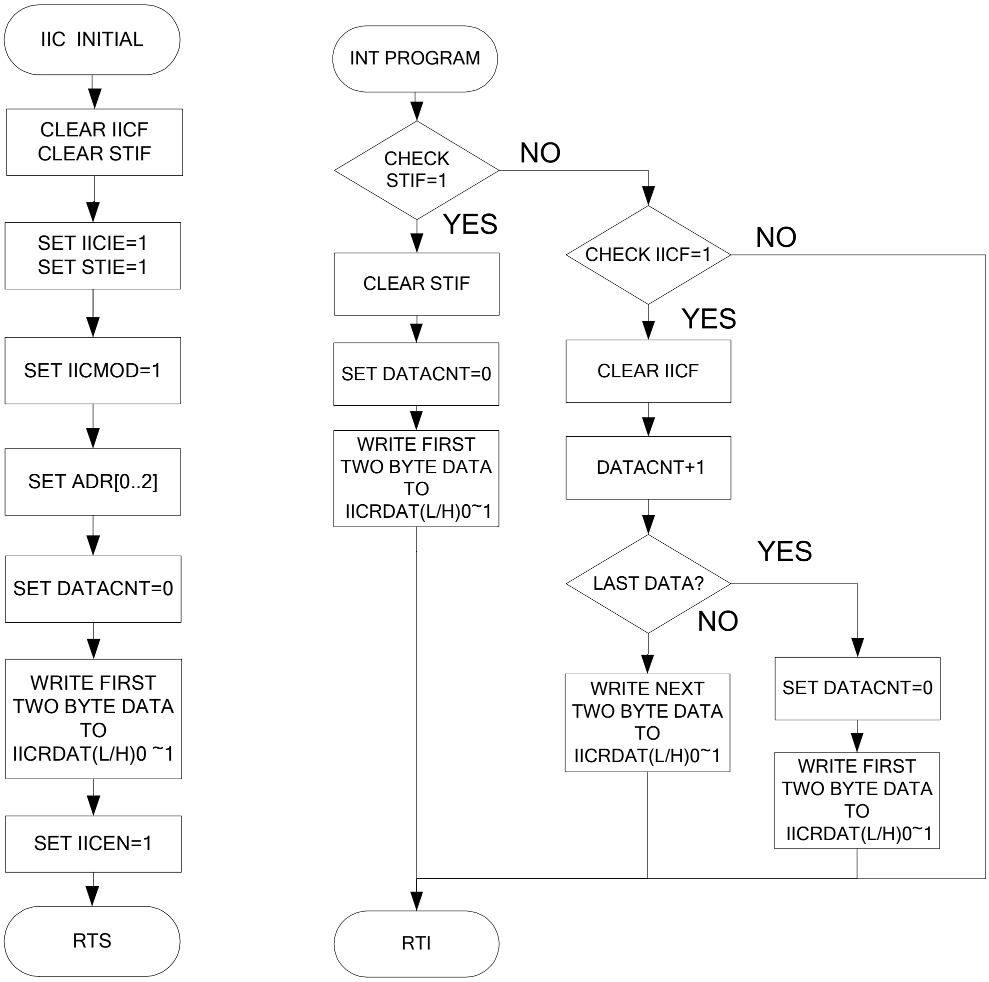
2) Fast read mode

Master read data fast in fast read mode, to reduce CPU process time. Follow shows a flowchart of IIC device program state machine in fast read mode. When master read the first data, IIC device automatically load next data to the sent buffer and generates an interrupt allows users to update new IICRDAT(L/H)0 and IICRDAT(L/H)1 data.

1. For two byte data:



2. For more than two byte data:



. IO Pad Cell Structure and Function Description

.. IO port with touch pad input

The input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control register=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. Software can performs a configuration (output data register=0, changing the IO control register 0 or 1) for open drain type that specifies suitable for key scan application. An additional feature supports the touch pad input.

Extern input	IO control data	Output data	Pull-up R	Wake-up feature
Disable	0	X	No	No
Disable	1	0	No	No
Disable	1	1	Enable	Enable
Enable and touch pad scan	X	X	No	No

X: don't care the value

IO control data	IO pad
0	Output register data
1	IO pad input data

Read PxI	Read input data
1	IO pad data

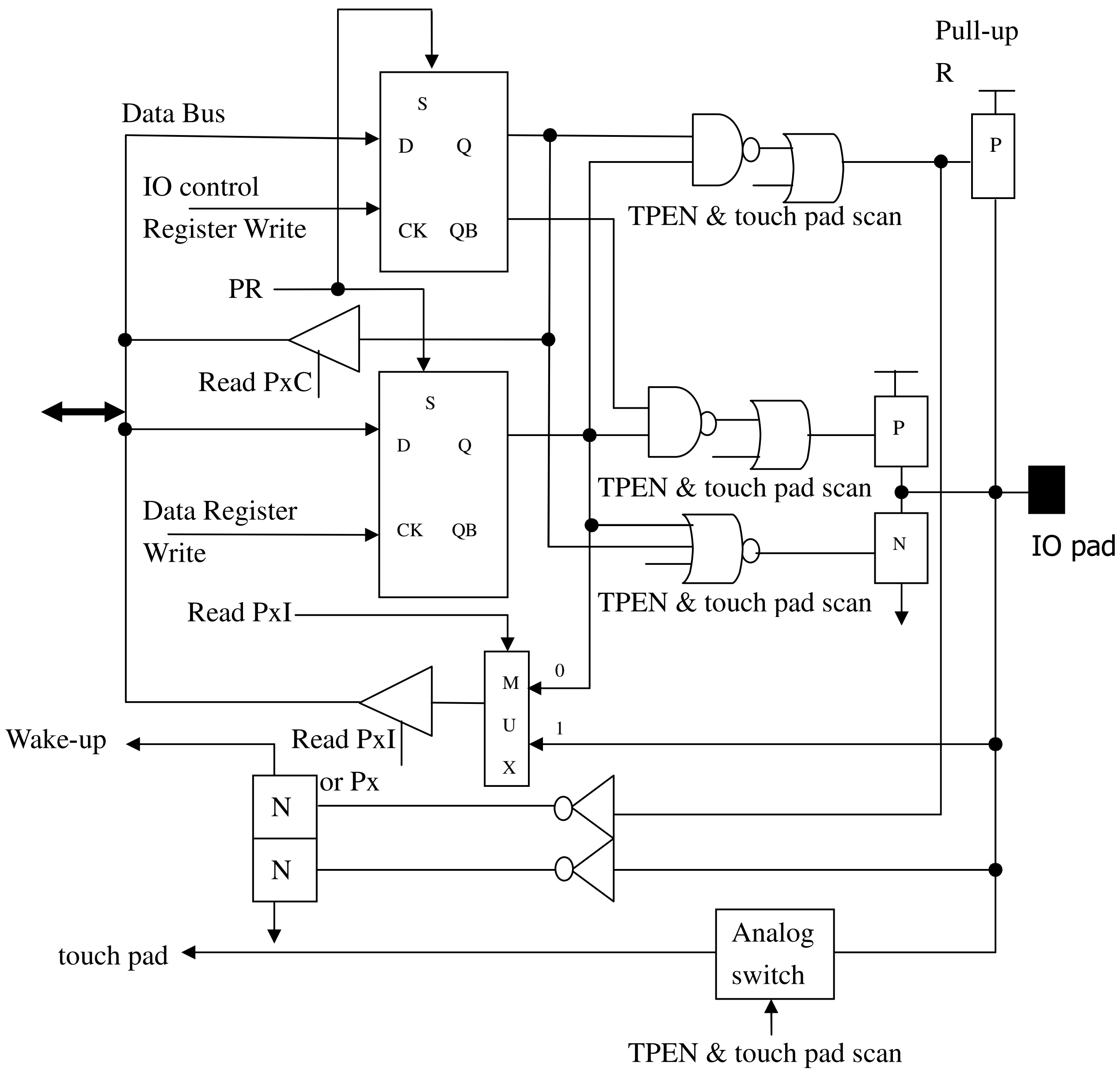


Figure IO-A: Standard IO port with touch pad input

.. IO port with internal PWM output

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the internal PWM output.

IO control data	Output data	Pull-up R	Wake-up feature
0	X	No	No
1	0	No	No
1	1	Enable	Enable

X: don't care the value

IO control data	Internal output	IO pad
0	Enable	Output internal data
0	Disable	Output register data
1	X	IO pad input data

X: don't care the value

Read PxI	Read input data
1	IO pad data

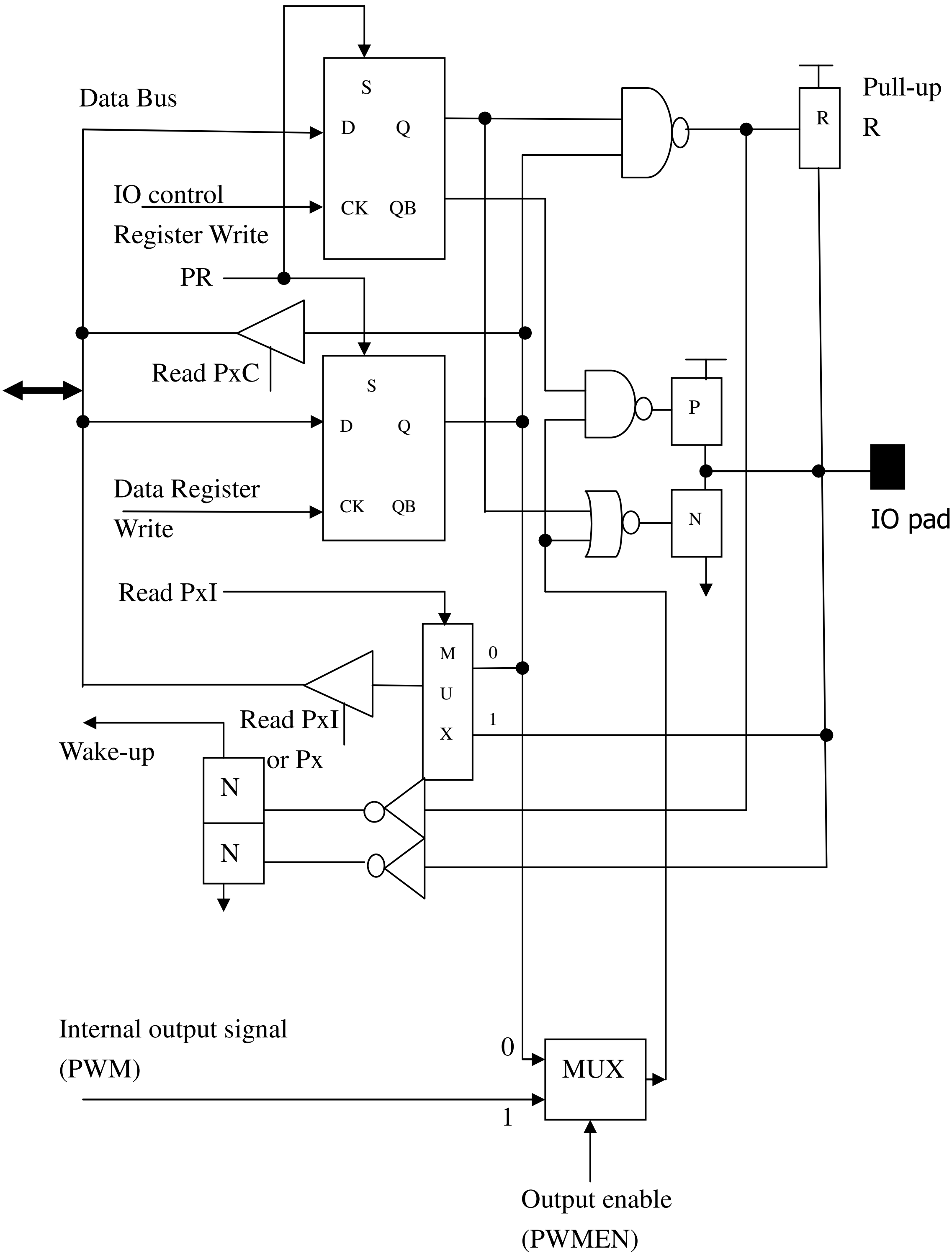


Figure IO-B: Standard IO port with internal PWM output

.. IO port with internal PWM output and external TCP1 clock input

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the internal PWM output and external TCP1 clock input.

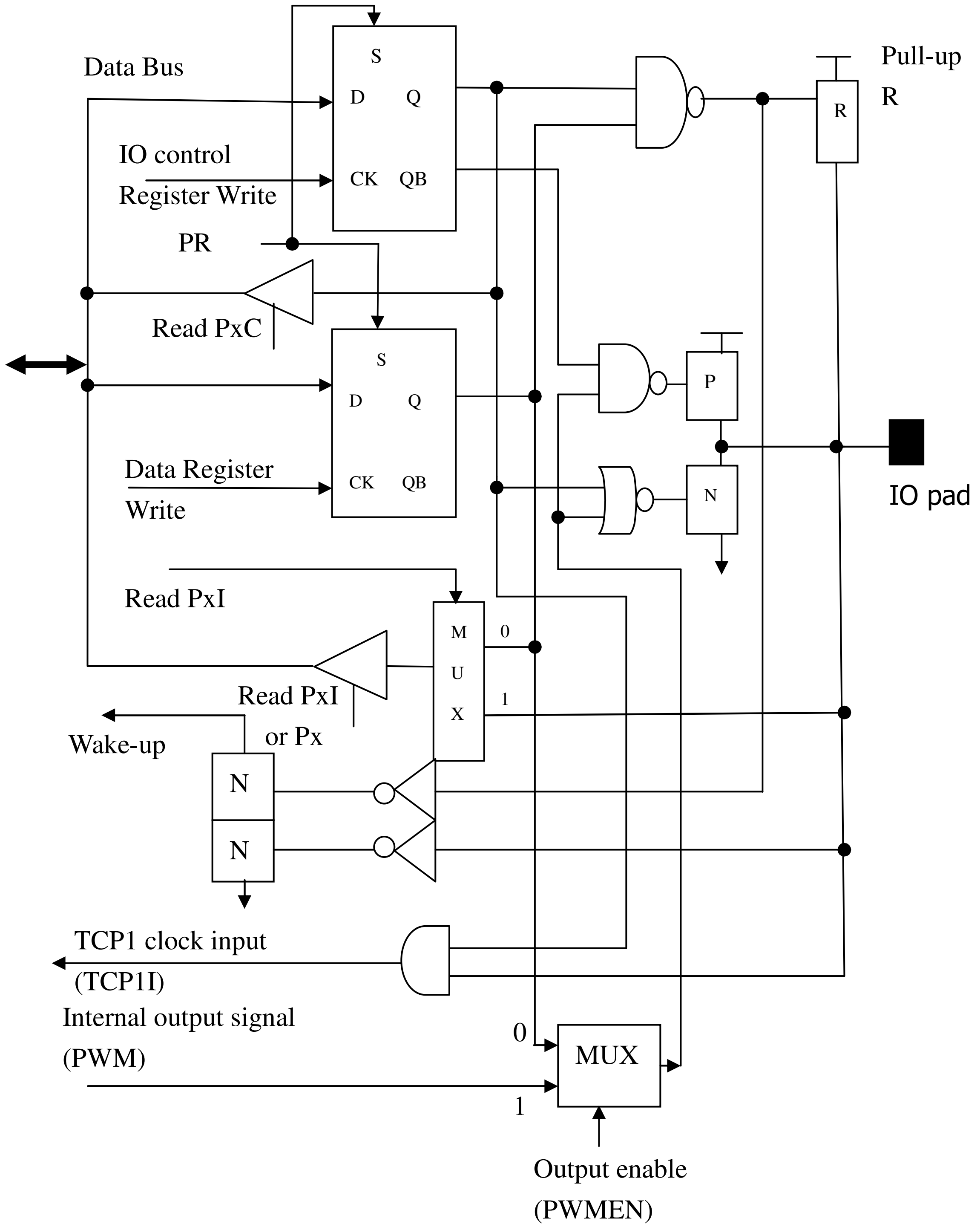
IO control data	Output data	Pull-up R	Wake-up feature	External input
0	X	No	No	No
1	0	No	No	Enable
1	1	Enable	Enable	Enable

X: don't care the value

IO control data	Internal output	IO pad
0	Enable	Output internal data
0	Disable	Output register data
1	X	IO pad input data

X: don't care the value

Read PxI	Read input data
1	IO pad data



<u>Figure IO-C: Standard IO port with internal PWM output and external TCP1 clock input</u>

.. IO port with internal PWM output and external interrupt trigger input

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the internal PWM output and external interrupt trigger input.

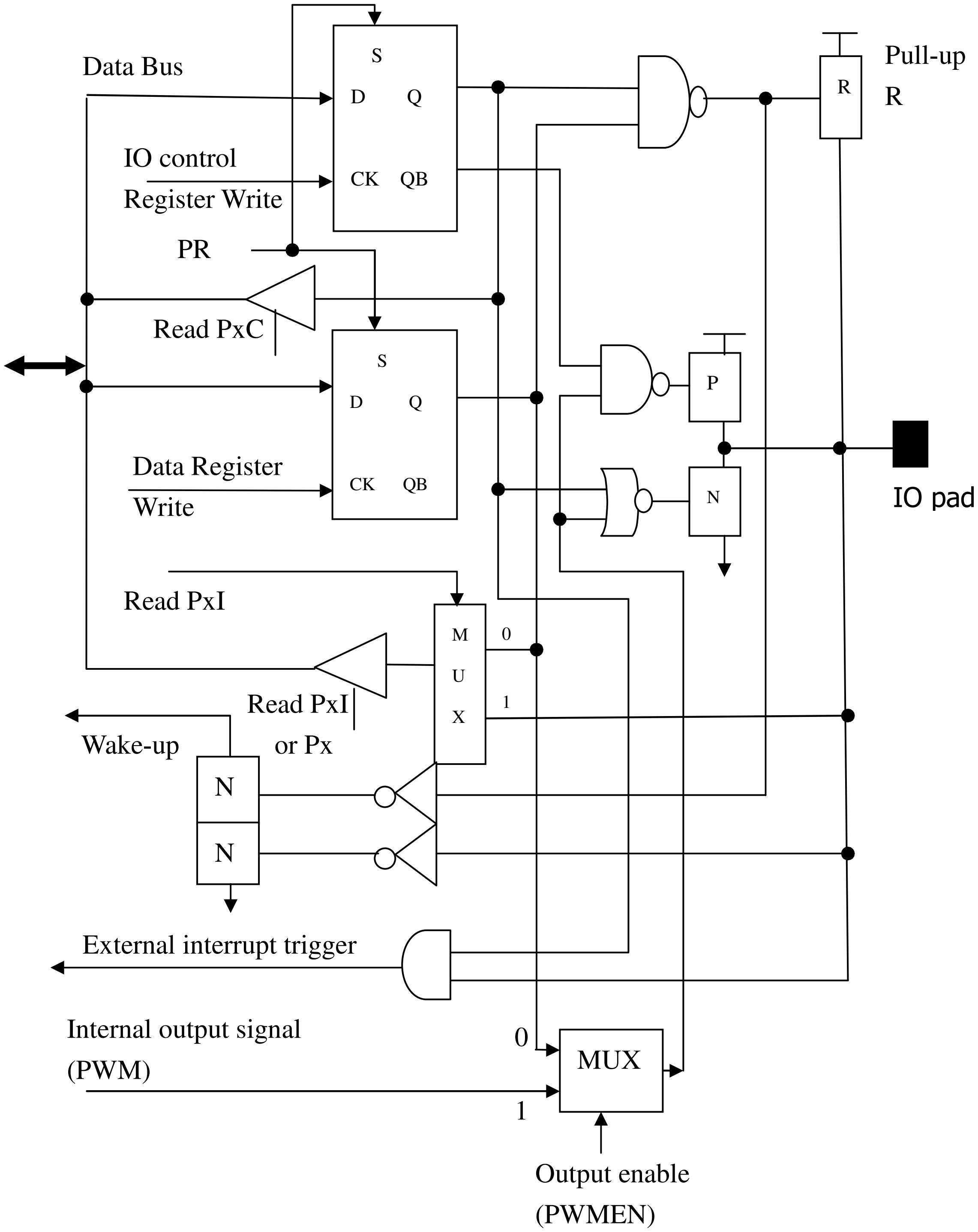
IO control data	Output data	Pull-up R	Wake-up feature	External input
0	X	No	No	No
1	0	No	No	Enable
1	1	Enable	Enable	Enable

X: don't care the value

IO control data	Internal output	IO pad
0	Enable	Output internal data
0	Disable	Output register data
1	X	IO pad input data

X: don't care the value

Read PxI	Read input data
1	IO pad data



<u>Figure IO-D: Standard IO port with internal PWM output and external interrupt trigger input</u>

.. IO port with internal PWM output and external interrupt trigger input and IIC

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the internal PWM output and external interrupt trigger input and IIC.

IO control data	Output data	Pull-up R	Wake-up feature	External input
0	X	No	No	No
1	0	No	No	Enable
1	1	Enable	Enable	Enable

X: don't care the value

IO control data	Internal output	IO pad
0	Enable	Output internal data
0	Disable	Output register data
1	X	IO pad input data

X: don't care the value

Read PxI	Read input data
1	IO pad data

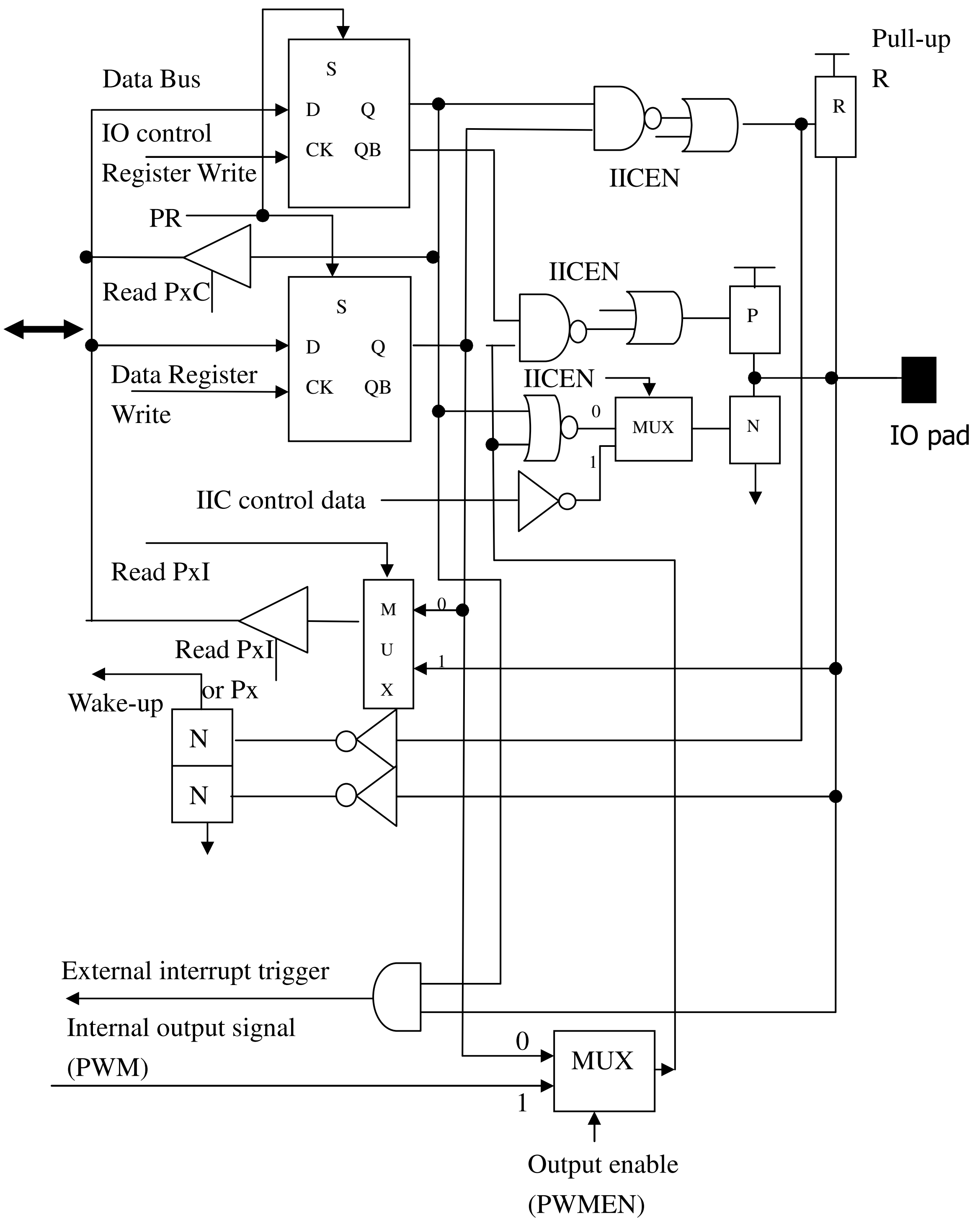


Figure IO-E: Standard IO port with internal PWM output and external interrupt trigger input and IIC

.. IO port with internal PWM output and IIC

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the internal PWM output and IIC.

IO control data	Output data	Pull-up R	Wake-up feature
0	X	No	No
1	0	No	No
1	1	Enable	Enable

X: don't care the value

IO control data	Internal output	IO pad
0	Enable	Output internal data
0	Disable	Output register data
1	X	IO pad input data

X: don't care the value

Read PxI	Read input data
1	IO pad data

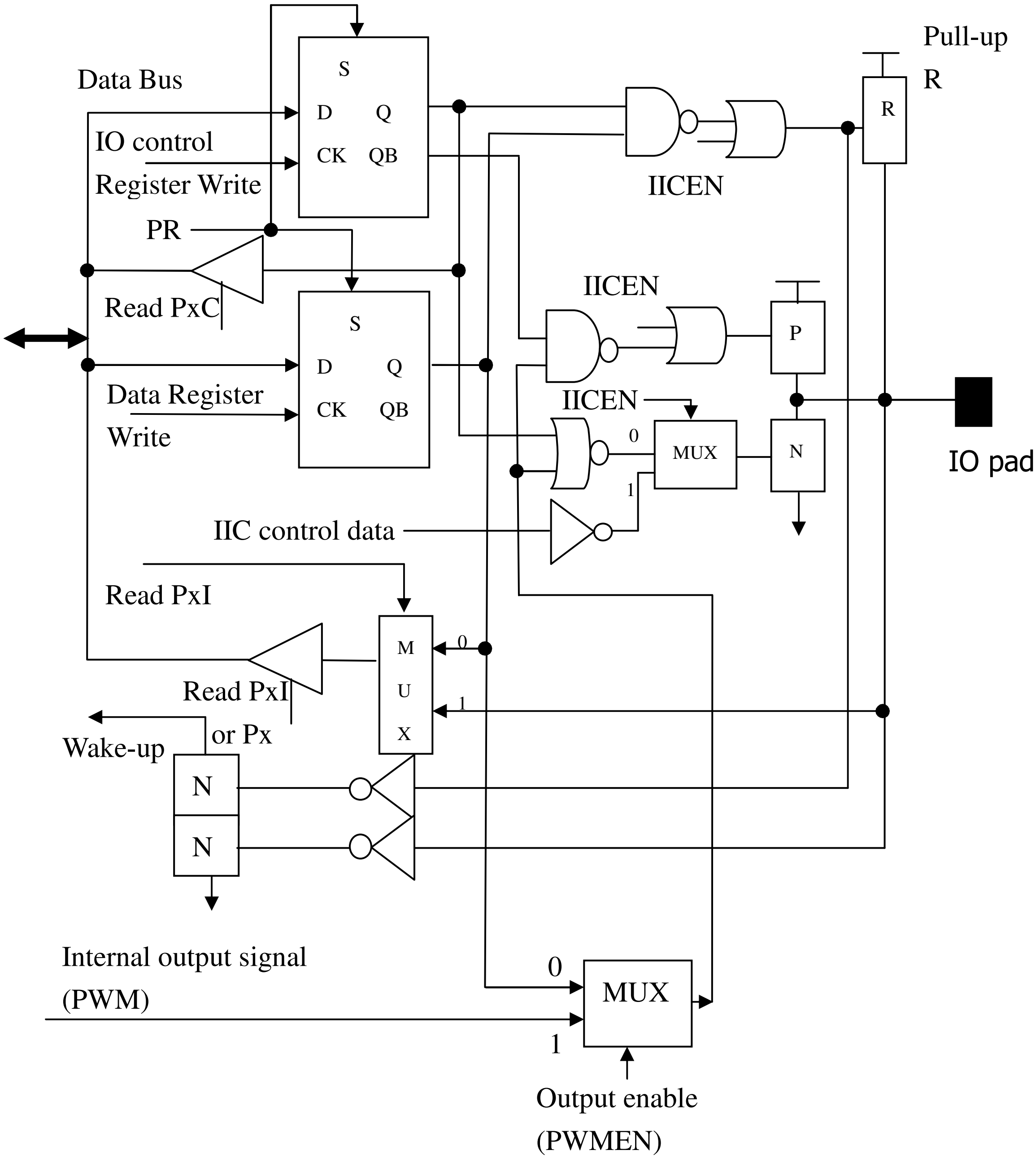


Figure IO-F: Standard IO port with internal PWM output and IIC

.. IO port with internal PWM output and external interrupt trigger input

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the internal PWM output and external interrupt trigger input.

IO control data	Output data	Pull-up R	Wake-up feature	External input
0	X	No	No	No
1	0	No	No	Enable
1	1	Enable	Enable	Enable

X: don't care the value

IO control data	Internal output	IO pad
0	Enable	Output internal data
0	Disable	Output register data
1	X	IO pad input data

X: don't care the value

Read PxI	Read input data
1	IO pad data

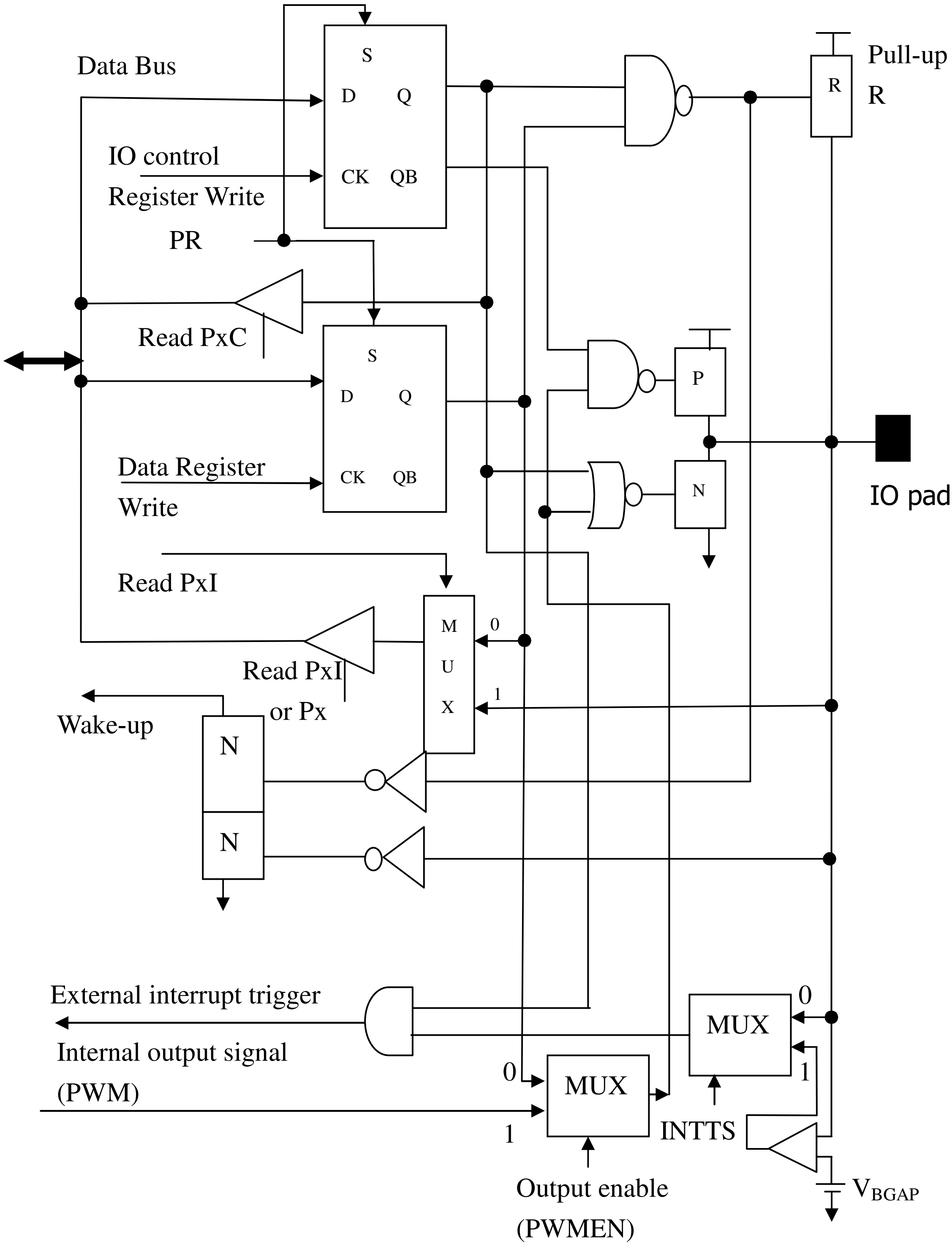


Figure IO-G: Standard IO port with internal PWM output and external interrupt trigger input

.. IO port with external interrupt trigger input

The standard input/output port has the IO control register for switching input or output mode and data register stores the output data in output mode. If IO control data=1 and output data=1, the IO port is programmed as input with pull-up resistor and also actives the wake-up function. User intends to read the port data with differed read instruction. The read PxI is reading data comes from IO pad data. The data register reading result will have the same value with output register data. If enable internal output, the IO port must set as output (IO control data=0). An additional feature supports the external interrupt trigger input.

IO control data	Output data	Pull-up R	Wake-up feature	External input
0	X	No	No	No
1	0	No	No	Enable
1	1	Enable	Enable	Enable

X: don't care the value

IO control data	IO pad
0	Output internal data
1	IO pad input data

Read PxI	Read input data
1	IO pad data

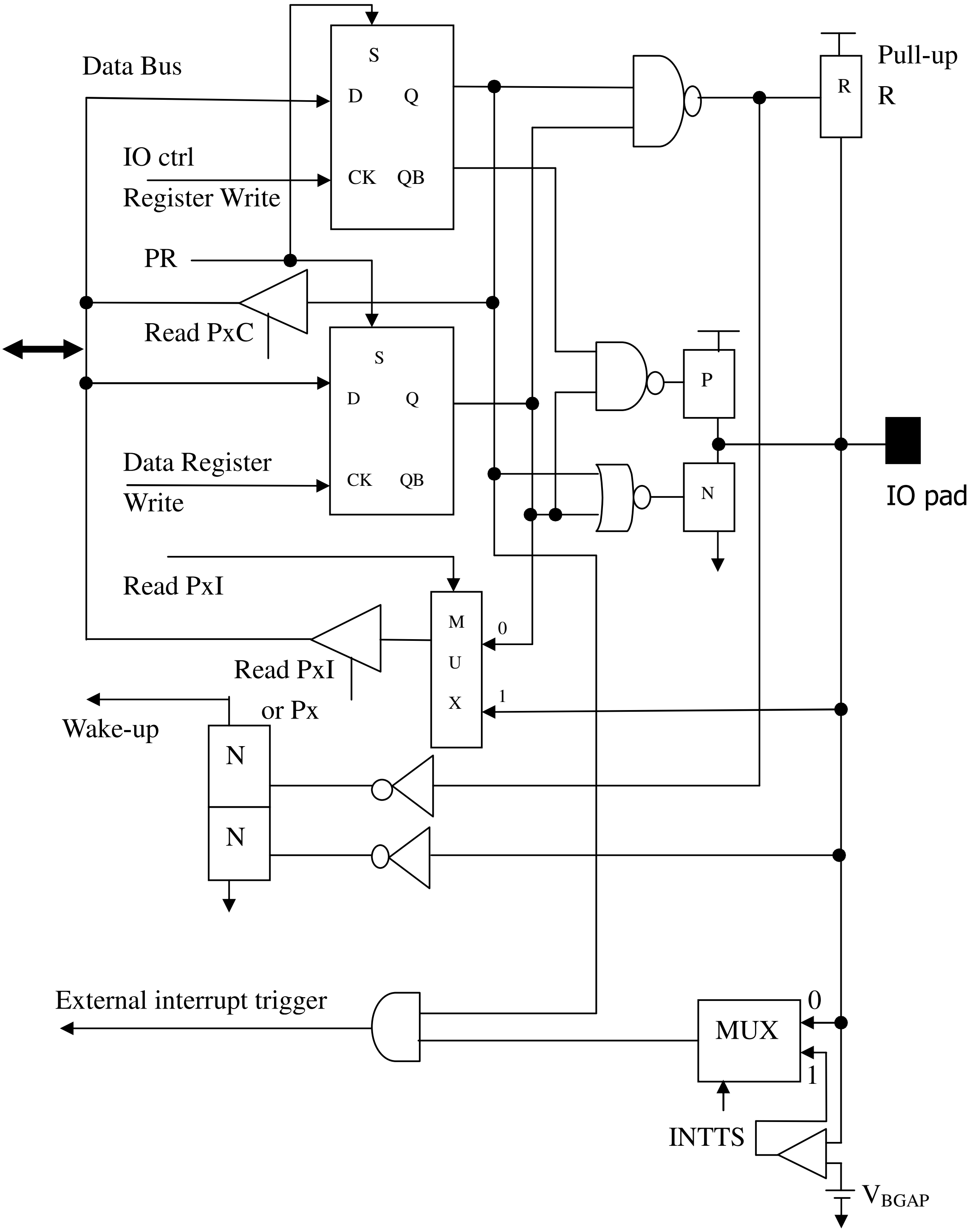


Figure IO-H: Standard IO port with external interrupt trigger input

3. IO Pad Cells

The main features of IO pad cell are including ESD/EFT protection and general IO access. A general IO pad cell can be configured as input with or without pull-up resistor, or working as a CMOS or NMOS output driver. The input pad cell must have pull-up resistor for avoiding a floating state when user doesn't care or not be used. For concerning the standby current, user can use data register or IO control register to fit the application.

. IO File Register

✧ PAC[012H]: Port A IO control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	PAC2	PAC1	PAC0
Read/Write	-	R/W	R/W	R/W

PAC2~PAC0: Port A IO control data.

✧ PA[013H]: Port A output data register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	PA2	PA1	PA0
Read/Write	-	R/W	R/W	R/W

PA2~PA0: Port A output data.

✧ PBC[014H]: Port B IO control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PBC3	PBC2	PBC1	PBC0
Read/Write	R/W	R/W	R/W	R/W

PBC3~PBC0: Port B IO control data.

✧ PB[015H]: Port B output data register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PB3	PB2	PB1	PB0
Read/Write	R/W	R/W	R/W	R/W

PB3~PB0: Port B output data.

✧ PCC[016H]: Port C IO control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PCC3	PCC2	PCC1	PCC0
Read/Write	R/W	R/W	R/W	R/W

PCC3~PCC0: Port C IO control data.

✧ PC[017H]: Port C output data register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PC3	PC2	PC1	PC0
Read/Write	R/W	R/W	R/W	R/W

PC3~PC0: Port C output data.

✧ PDC[018H]: Port D IO control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PDC3	PDC2	PDC1	PDC0
Read/Write	R/W	R/W	R/W	R/W

PDC3~PDC0: Port D IO control data.

✧ PD[019H]: Port D output data register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PD3	PD2	PD1	PD0
Read/Write	R/W	R/W	R/W	R/W

PD3~PD0: Port D output data.

✧ PEC[01AH]: Port E IO control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PEC3	PEC2	PEC1	PEC0
Read/Write	R/W	R/W	R/W	R/W

PEC3~PEC0: Port E IO control data.

✧ PE[01BH]: Port E output data register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PE3	PE2	PE1	PE0
Read/Write	R/W	R/W	R/W	R/W

PE3~PE0: Port E output data.

✧ PFC[01CH]: Port F IO control register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PFC3	PFC2	PFC1	PFC0
Read/Write	R/W	R/W	R/W	R/W

PFC3~PFC0: Port F IO control data.

✧ PF[01DH]: Port F output data register [R/W], default value [1111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PF3	PF2	PF1	PF0
Read/Write	R/W	R/W	R/W	R/W

PF3~PF0: Port F output data.

✧ PAI[207H]: Port A pad data reading address [R], default value [----]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	PAI2	PAI1	PAI0
Read/Write	-	R	R	R

PAI3~PAI0: Port A pad data.

✧ PBI[208H]: Port B pad data reading address [R], default value [----]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PBI3	PBI2	PBI1	PBI0
Read/Write	R	R	R	R

PBI3~PBI0: Port B pad data.

✧ PCI[209H]: Port C pad data reading address [R], default value [----]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PCI3	PCI2	PCI1	PCI0
Read/Write	R	R	R	R

PCI3~PCI0: Port C pad data.

✧ PDI[20AH]: Port D pad data reading address [R], default value [----]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PDI3	PDI2	PDI1	PDI0
Read/Write	R	R	R	R

PDI3~PDI0: Port D pad data.

✧ PEI[20BH]: Port E pad data reading address [R], default value [----]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PEI3	PEI2	PEI1	PEI0
Read/Write	R	R	R	R

PEI3~PEI0: Port E pad data.

✧ PFI[20CH]: Port F pad data reading address [R], default value [----]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PFI3	PFI2	PFI1	PFI0
Read/Write	R	R	R	R

PFI3~PFI0: Port F pad data.

. IO Port's Special function

When SpecIO is selected by mask option, PA0, PB0 and PB1 is special IO function. It can output ODATA register to user. ODATA can be store Key touch information by software. User set PA0 for input, PB0,PB1 for output. User can use this function to get Key touch information.

When using special IO function, do not use IIC function. If not, PB0,PB1 are unexpected signal.

✧ ODATA[21FH]: Touch pad output register for special function [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	ODATA3	ODATA2	ODATA1	ODATA0
Read/Write	R/W	R/W	R/W	R/W

ODATA3~ODATA0: Touch pad information.

PA0 (input)	PB0 (output)	PB1(output)
1	ODATA0	ODATA1
0	ODATA2	ODATA3

16 non-contact inputs touch pad detector

The touch pad detector applies the charge sharing conception. The inputs share the pad with IO ports. Built-in charge sharing control, duty detector and de-bounce feature can response the input with varied output refresh rate that dependant on the system request. For power saving concern, auto power off function and wake up de-bounce capability can support a lower average operating current.

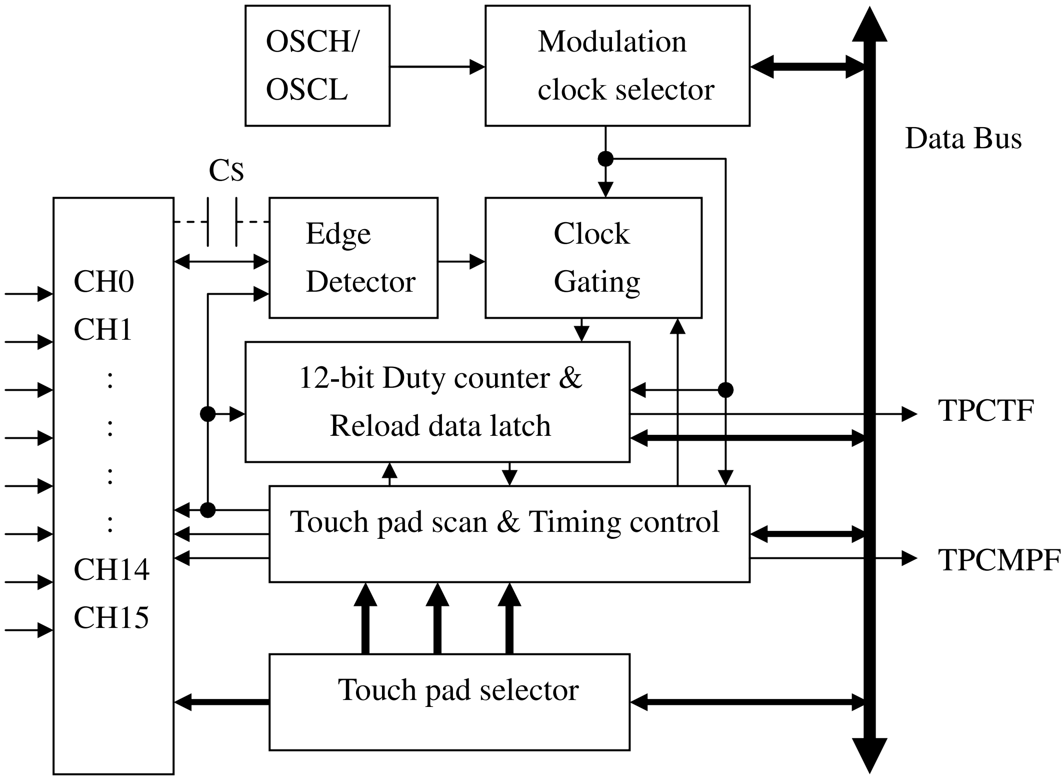
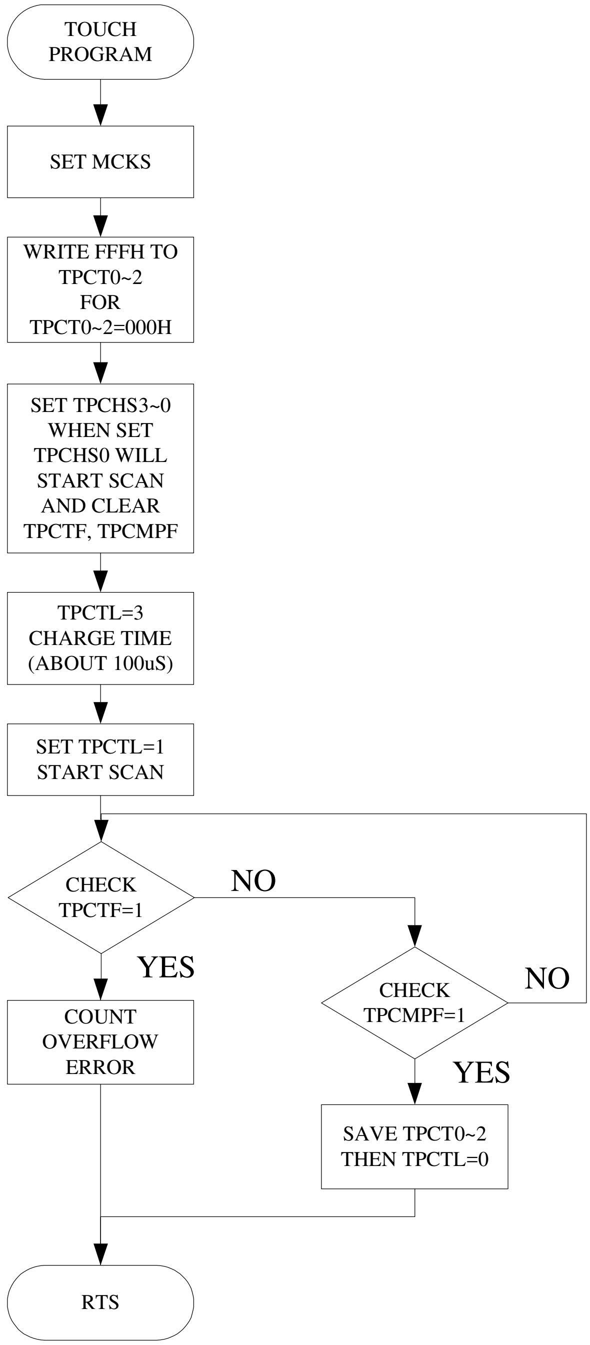


Figure: 16 pads Touch pad detector

Parameters	Target value	Remark
Touch pad OSC	4MHz or 16KHz	Using OSCH or OSCL
Modulation clock	OSCH/N or OSCL	N=1,2,4,8,16,32,64
Duty counter	12-bit	With INT
Reload data latch	12-bit	Write only
Touch pads	1~16 pads	-
Key de-bounce time	s/w implements	By application or cover thickness
Sensitivity level	Offset value by s/w	Resolution=1 modulation clock

The flowchart as follow:



TPINTC[01EH]: Touch pad interrupt control register [R/W], default value [0000]

TPINTC	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCTIE	TPCMPIE	-	-
Read/Write	R/W	R/W	-	-

TPCMPIE: Capacitor overcharge interrupt enable. (0: disable; 1: enable)
TPCTIE: Duty counter overflow interrupt enable. (0: disable; 1: enable)

✧ TPINTF[01FH]: Touch pad request flag register [R/W], default value [0000]

TPINTF	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCTF	TPCMF	-	-
Read/Write	R/W	R/W	-	-

TPCMPF: Capacitor overcharge flag. (0: inactive; 1: active)
TPCTF: Duty counter overflow flag. (0: inactive; 1: active)

✧ TPCT0[215H]: Touch pad duty counter and latch data register 0 [R], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCT3/CT3	TPCT2/CT2	TPCT1/CT1	TPCT0/CT0
Read/Write	R/W	R/W	R/W	R/W

TPCT3~TPCT0: Duty counter 1st nibble data for counter read.
CT3~CT0: 1st nibble of reload latch data.

✧ TPCT1[216H]: Touch pad duty counter and latch data register 1 [R], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCT7/CT7	TPCT6/CT6	TPCT5/CT5	TPCT4/CT4
Read/Write	R/W	R/W	R/W	R/W

TPCT7~TPCT4: Duty counter 2nd nibble data for counter read.
CT7~CT4: 2nd nibble of reload latch data.

✧ TPCT2[217H]: Touch pad duty counter and latch data register 2 [R], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPCT11/CT11	TPCT10/CT10	TPCT9/CT9	TPCT8/CT8
Read/Write	R/W	R/W	R/W	R/W

TPCT11~TPCT8: Duty counter 3rd nibble data for counter read.
CT11~CT8: 3rd nibble of reload latch data.

Duty counter value= TPCT2*256 +TPCT1*16+TPCT0

When user writes data to the TPCT2~TPCT0, the data just keep in TPCT2~TPCT0 latch register. When TPCHS0 is writing, the TPCT2~TPCT0 latch register's complement value will load into TPCT2~TPCT0 duty counter as initial value and start the scan function.

The duty counter will be enabled by writing the TPCHS0 register and will set the TPCTF flag if duty counter overflow. As writing the TPCHS0 register will reload the 12-bit duty counter and clear the TPCTF and TPCMPF.

✧ MCKS[21AH]: Modulation clock selector register [R/W], default value [0111]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	MCKS2	MCKS1	MCKS0
Read/Write	-	R/W	R/W	R/W

MCKS2~MCKS0: Modulation clock selector.

MCKS2~MCKS0	**Sample time**	**MCKS2~MCKS0**	**Sample time**
000	OSCH/1	100	OSCH/16
001	OSCH/2	101	OSCH/32
010	OSCH/4	110	OSCH/64
011	OSCH/8	111	OSCL

The TPCMPF will be set as no modulation clock going into duty counter with de-bounce feature and will also call the interrupt as TPCMPIE=1.

✧ TPCHS0[210H]: Touch pad channel selector register0 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPEN3	TPEN2	TPEN1	TPEN0
Read/Write	R/W	R/W	R/W	R/W

TPEN3~TPEN0: Touch pad channel selector 1st nibble.

✧ TPCHS1[211H]: Touch pad channel selector register1 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPEN7	TPEN6	TPEN5	TPEN4
Read/Write	R/W	R/W	R/W	R/W

TPEN7~TPEN4: Touch pad channel selector 2nd nibble.

✧ TPCHS2[212H]: Touch pad channel selector register2 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPEN11	TPEN10	TPEN9	TPEN8
Read/Write	R/W	R/W	R/W	R/W

TPEN11~TPEN8: Touch pad channel selector 3rd nibble.

✧ TPCHS3[213H]: Touch pad channel selector register3 [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	TPEN15	TPEN14	TPEN13	TPEN12
Read/Write	R/W	R/W	R/W	R/W

TPEN15~TPEN12: Touch pad channel selector 4th nibble.

As program writes the TPCHS0 register, hardware automatically discharges the external capacitor and enable the sensor clock input until period end.

Channel Enable State	**TPCHS0 TPEN3~ TPEN0**	**TPCHS1 TPEN7~ TPEN4**	**TPCHS2 TPEN11~ TPEN8**	**TPCHS3 TPEN15~ TPEN12**
TP0	0001	0000	0000	0000
TP1	0010	0000	0000	0000
TP2	0100	0000	0000	0000
TP3	1000	0000	0000	0000
TP4	0000	0001	0000	0000
TP5	0000	0010	0000	0000
TP6	0000	0100	0000	0000
TP7	0000	1000	0000	0000
TP8	0000	0000	0001	0000
TP9	0000	0000	0010	0000
TP10	0000	0000	0100	0000
TP11	0000	0000	1000	0000
TP12	0000	0000	0000	0001
TP13	0000	0000	0000	0010
TP14	0000	0000	0000	0100
TP15	0000	0000	0000	1000

When TPCHS0 is writing, TPCTL will be set TP RUN mode, and begin to scan the channel by TPCHS3~TPCHS0 select.

Users can enable multi-channel by setting corresponding bit 1, that will turn on all enable channel at the same time.

✧ TPCTL[214H]: Touch pad control register [R/W], default value [-000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	TPCTL2	TPCTL1	TPCTL0
Read/Write	-	R/W	R/W	R/W

TPCTL2~TPCTL0: Touch pad control selector.

As program writes the TPCTL register, hardware automatically charges the external capacitor and reload TPCTx and enable the sensor clock input until period end.

TPCTL2~TPCTL0	**Channel Enable State**
000	TP STOP
001	TP RUN
010	-
011	Charge
100	Inner pad
101	-
110	-
111	-

TP STOP: STOP the touch pad feature and release pad for IO port.

TP RUN: TP RUN is touch pad scan start signal, it's scan the channel by TPCHS3~TPCHS0 select.

Charge: Charge can hold touch pad in charge state, to avoid charge time too short.

Inner pad: Select switch select Inner pad. Inner pad is reference pad, this pad is no bounding to package.

When user writes data to the TPCT2~TPCT0, the data just keep in TPCT2~TPCT0 latch register. When writing the TPCTL register (exclude select TP STOP), the TPCT2~TPCT0 latch register's complement value will load into

TPCT2~TPCT0 duty counter as initial value and start the scan function.

As writing the TPCTL register (exclude select TP STOP) will reload the 12-bit duty counter and clear the TPCTF and TPCMPF.

As touch pad analog switch keeps on, the relative IO port is disabled as tri-state by hardware.

✧ CSAL[218H]: Select Capacity load low nibble register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	CSA3	CSA2	CSA1	CSA0
Read/Write	R/W	R/W	R/W	R/W

CSA3~CSA0: Select Capacity load low nibble data for touch pad.

✧ CSAH[219H]: Select Capacity load high nibble register [R/W], default value [--00]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	CSA5	CSA4
Read/Write	-	-	R/W	R/W

CSA5~CSA4: Select Capacity load high nibble data for touch pad.

CSA5~CSA0	Extra capacity load
00 0000	0 * C array unit
00 0001	1 * C array unit
……	……
11 1110	62 * C array unit
11 1111	63 * C array unit

Note: C array unit = 0.25pf

§ Mask Option Table:

All the OTP mask option register can open for user to reset the initial value, but should enable the MRO. User writes MRO address first then changes the target mask option register data. The MRO enable will be cleared with other writing address. Bit 3 of MOP0 must always set to 1.

✧ MOP0: LVR voltage select option register [R/W], default value [1-0-]

Mask option	Bit3	Bit2	Bit1	Bit0
Bit Name	1	-	LVRVS	-
Read/Write	R/W	-	R/W	-

✧ MOP1: PWM start level option register [R/W], default value [-000]

Mask option	Bit3	Bit2	Bit1	Bit0
Bit Name	-	PWM2S	PWM1S	PWM0S
Read/Write	-	R/W	R/W	R/W

✧ MOP2: INT trigger option register [R/W], default value [0000]

Mask option	Bit3	Bit2	Bit1	Bit0
Bit Name	INT1S1	INT1S0	INT0S1	INT0S0
Read/Write	R/W	R/W	R/W	R/W

✧ MOP3: Function pin select 1st option register [R/W], default value [0000]

Mask option	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM1PS	PWM0PS	INT1PS	INT0PS
Read/Write	R/W	R/W	R/W	R/W

✧ MOP4: Function select option register [R/W], default value [0--0]

Mask option	Bit3	Bit2	Bit1	Bit0
Bit Name	LVREN	-	-	LDOVS
Read/Write	R/W	-	-	R/W

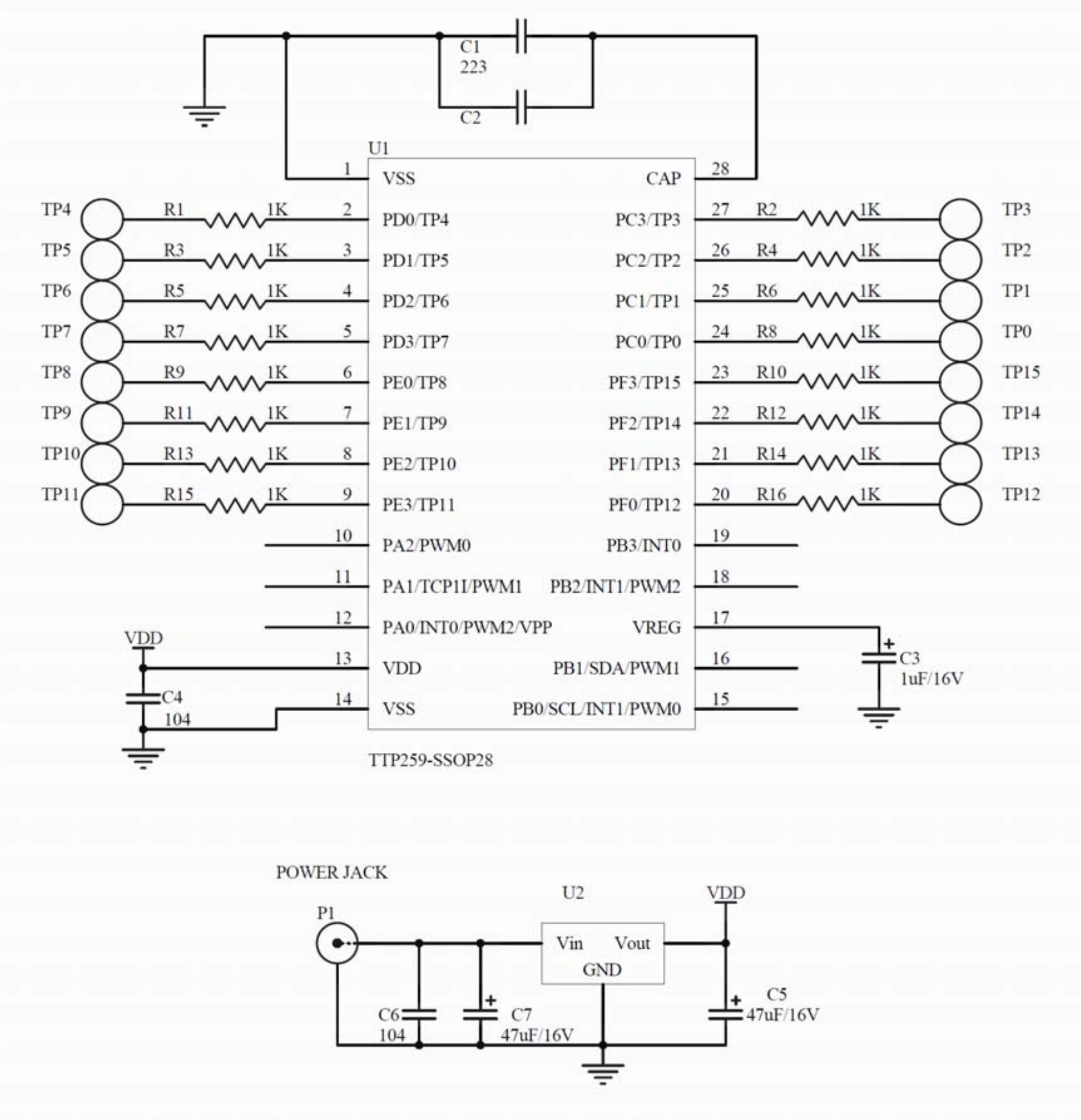
✧ MOP5: Function pin select 2nd option register [R/W], default value [000-]

Mask option	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM2PS1	PWM2PS0	SpecIO	-
Read/Write	R/W	R/W	R/W	-

The following table shows the mask option in this chip. All the mask options must be defined clearly and ensure to meet user's proper function.

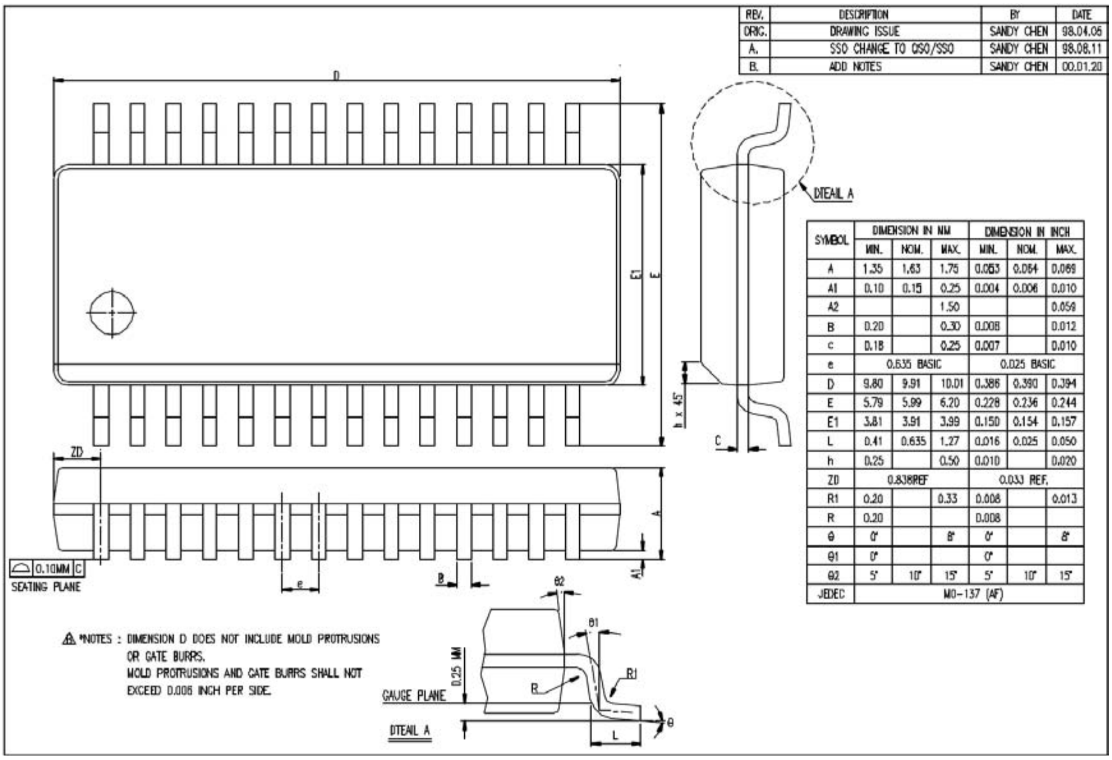
No.	Mask Option	Function Descriptions	
+1	LVR output Voltage select	0	2.2V
		1	3.0V
+1	PWM0S	0	Start 0(active high)
		1	Start 1(active low)
+1	PWM1S	0	Start 0(active high)
		1	Start 1(active low)
+1	PWM2S	0	Start 0(active high)
		1	Start 1(active low)
+2	INT0F trigger type INT0S1,INT0S0	00	Low level trigger
		01	Falling edge trigger
		10	Rising edge trigger
		11	Dual edge trigger
+2	INT1F trigger type INT1S1,INT1S0	00	Low level trigger
		01	Falling edge trigger
		10	Rising edge trigger
		11	Dual edge trigger
+1	INT0 function pin select	0	INT0 select PB3
		1	INT0 select PA0
+1	INT1 function pin select	0	INT1 select PB2
		1	INT1 select PB0
+1	PWM0 function pin select	0	PWM0 select PA2
		1	PWM0 select PB0
+1	PWM1 function pin select	0	PWM1 select PA1
		1	PWM1 select PB1
+1	LDO output Voltage select	0	2.7V
		1	4.2V
+1	LVREN select	0	LVREN disable
		1	LVREN enable
+1	SpecIO	0	PA0,PB0,PB1 is normal IO port
		1	PA0,PB0,PB1 is special function
+2	PWM2 function pin select	00	PA0
		01	PA0
		10	PB2
		11	Both PA0 and PB2

§ Application Circuit：



§ Package and Pad Information:

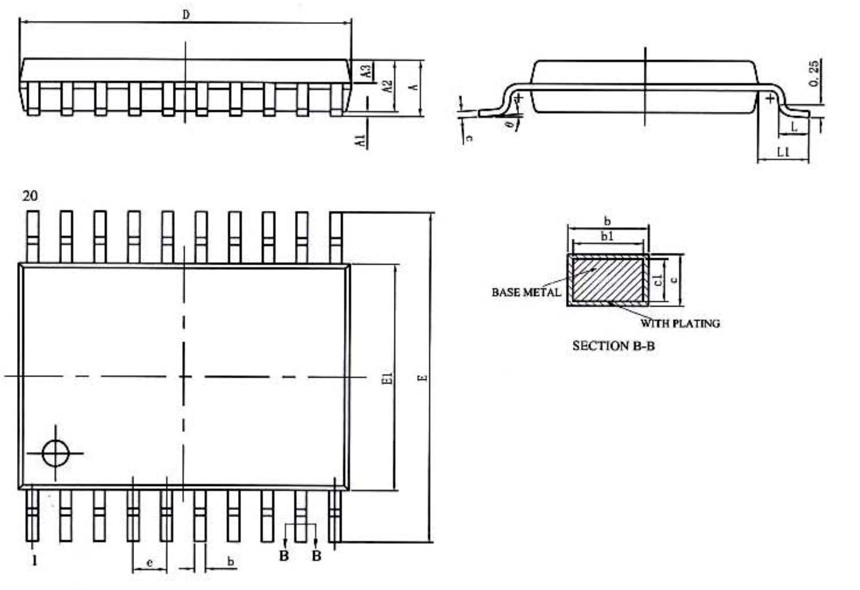
- **SSOP 28**



REV.	DESCRIPTION	BY	DATE
ORIG.	DRAWING ISSUE	SANDY CHEN	98.04.05
A.	SSO CHANGE TO QSO/SSO	SANDY CHEN	98.08.11
B.	ADD NOTES	SANDY CHEN	00.01.20

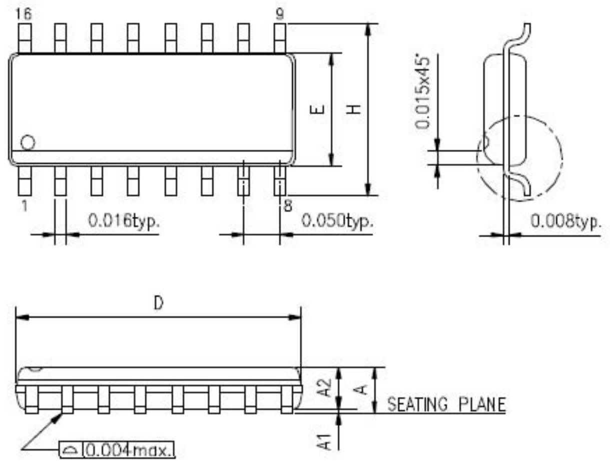
SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	1.35	1.63	1.75	0.053	0.064	0.069
A1	0.10	0.15	0.25	0.004	0.006	0.010
A2			1.50			0.059
B	0.20		0.30	0.008		0.012
c	0.18		0.25	0.007		0.010
e	0.635 BASIC			0.025 BASIC		
D	9.80	9.91	10.01	0.386	0.390	0.394
E	5.79	5.99	6.20	0.228	0.236	0.244
E1	3.81	3.91	3.99	0.150	0.154	0.157
L	0.41	0.635	1.27	0.016	0.025	0.050
h	0.25		0.50	0.010		0.020
ZD	0.838REF			0.033 REF.		
R1	0.20		0.33	0.008		0.013
R	0.20			0.008		
θ	0°		8°	0°		8°
θ1	0°			0°		
θ2	5°	10°	15°	5°	10°	15°
JEDEC	MO-137 (AF)					

- **TSSOP 20**



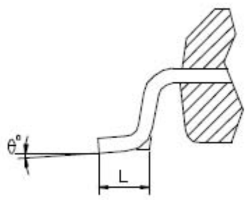
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.20
A1	0.05	—	0.15
A2	0.80	1.00	1.05
A3	0.39	0.44	0.49
b	0.20	—	0.29
b1	0.19	0.22	0.25
c	0.13	—	0.18
c1	0.12	0.13	0.14
D	6.40	6.50	6.60
E1	4.30	4.40	4.50
E	6.20	6.40	6.60
e	0.65BSC		
L	0.45	0.60	0.75
L1	1.00BSC		
θ	0	—	8°

- **SOP 16**



SYMBOLS	MIN.	MAX.
A	0.053	0.069
A1	0.004	0.010
D	0.386	0.394
E	0.150	0.157
H	0.228	0.244
L	0.016	0.050
θ°	0	8

UNIT : INCH



§ Ordering Information：

	Package type
TTP259-ASFN	SSOP28-A
TTP259-DTDN	TSSOP20-B
TTP259 -EOBN	SOP16-B

§ REVISION HISTORY：

2015/05/08：(Ver. 1.0) New build
2015/05/25：(Ver. 1.1) Modify package type TSSOP20-B and SOP16-B
2018/09/20：(Ver. 1.2) Modify TSSOP20 outline diagram
2018/10/02：(Ver. 1.3) Modify Page6 Top=-40~+85