§ 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:

TTP277 MCU is an easy-used 4-bit CPU base microcontroller. It contains 1984-word ROM(MTP), 144-nibble RAM, timer/counter, time base, watch dog timer, PWM, interrupt service, IO control hardware, 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 system clocks) except read table instruction(RTB)
4. Advance CMOS process
5. Working memory with 1984*16 program ROM(MTP) and 144*4 SRAM
6. 4-level stacks
7. LVR voltage can select 2.0V or 2.7V by register
8. Operating voltage: 3.5V~5.5V (LVR=2.7V); 2.2V~5.5V (LVR=2.0V)
9. System operating frequency: (at VDD=5V)
   . High-speed system oscillator(RC8M):
   - ✧ Built-in RC oscillator: 8MHz (typical) ± 5%
   - ✧ High-speed system clock(OSCH): 4MHz (typical)

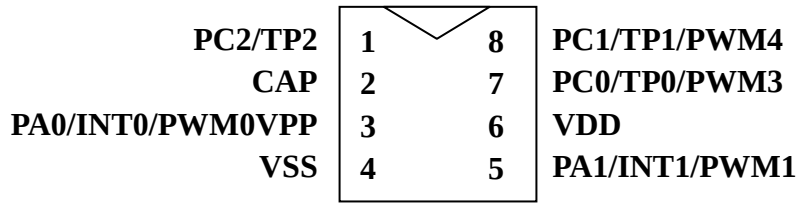
   .Low speed peripheral oscillator(RC32K):
   - ✧ Built-in RC oscillator: 32KHz (typical) ± 30%
   - ✧ Low-speed peripheral clock(OSCL): 32KHz (typical)

10. The MCU will go into low speed operation mode (RC32K on and RC8M off) automatically after power on or reset release.
11. Offer 2 IO + 3 touch pad or 5 general programmable IO
    - ✧ IO port built-in key wake-up feature enable by software setting
    - ✧ Providing external interrupt inputs
    - ✧ Offering internal signal outputs, like PWM
12. Two time base
    - ✧ Time base offers 2 various period interrupt request
13. One 8-bit TCP1 auto-reload timer/counter
    - ✧ 4 kind clock sources selected by software
14. One 8-bit TCP2 auto-reload timer/counter, can improve PWM function
    - ✧ 4 kind clock sources selected by software
15. Built-in 4 set 8-bit PWM output
16. 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
    - ✧ Provide high/low system operating speed, sleep and stop mode for power saving control
17. 3 pins with touch pad detection
18. Provides 8 interrupt sources
    - ✧ External: INT0, INT1 shared with IO pad
    - ✧ Internal: two timer/counter, two time base, two touch pad's interrupt
19. Provide package types
    - ✧ SOP 8/SOT23-6/DFN8L

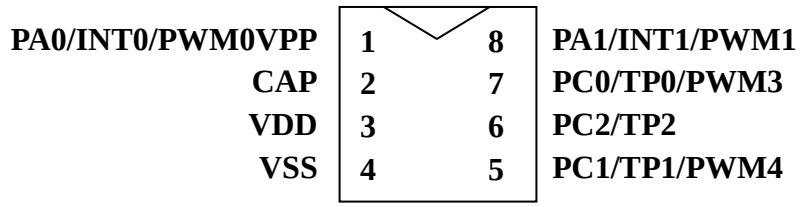
§ Applications:

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

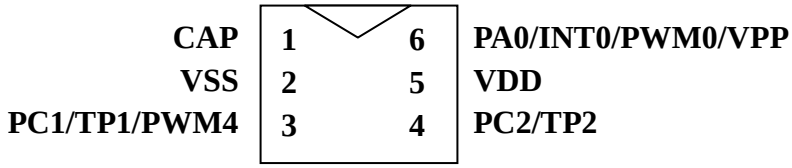
§ Package Description:



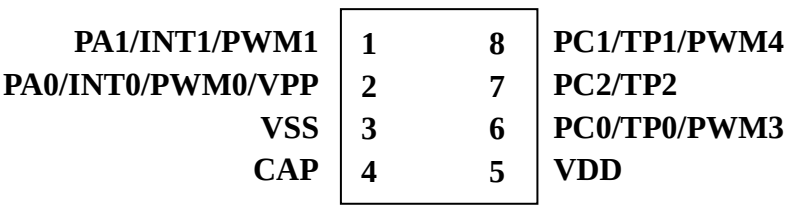
SOP8-A



SOP8-B

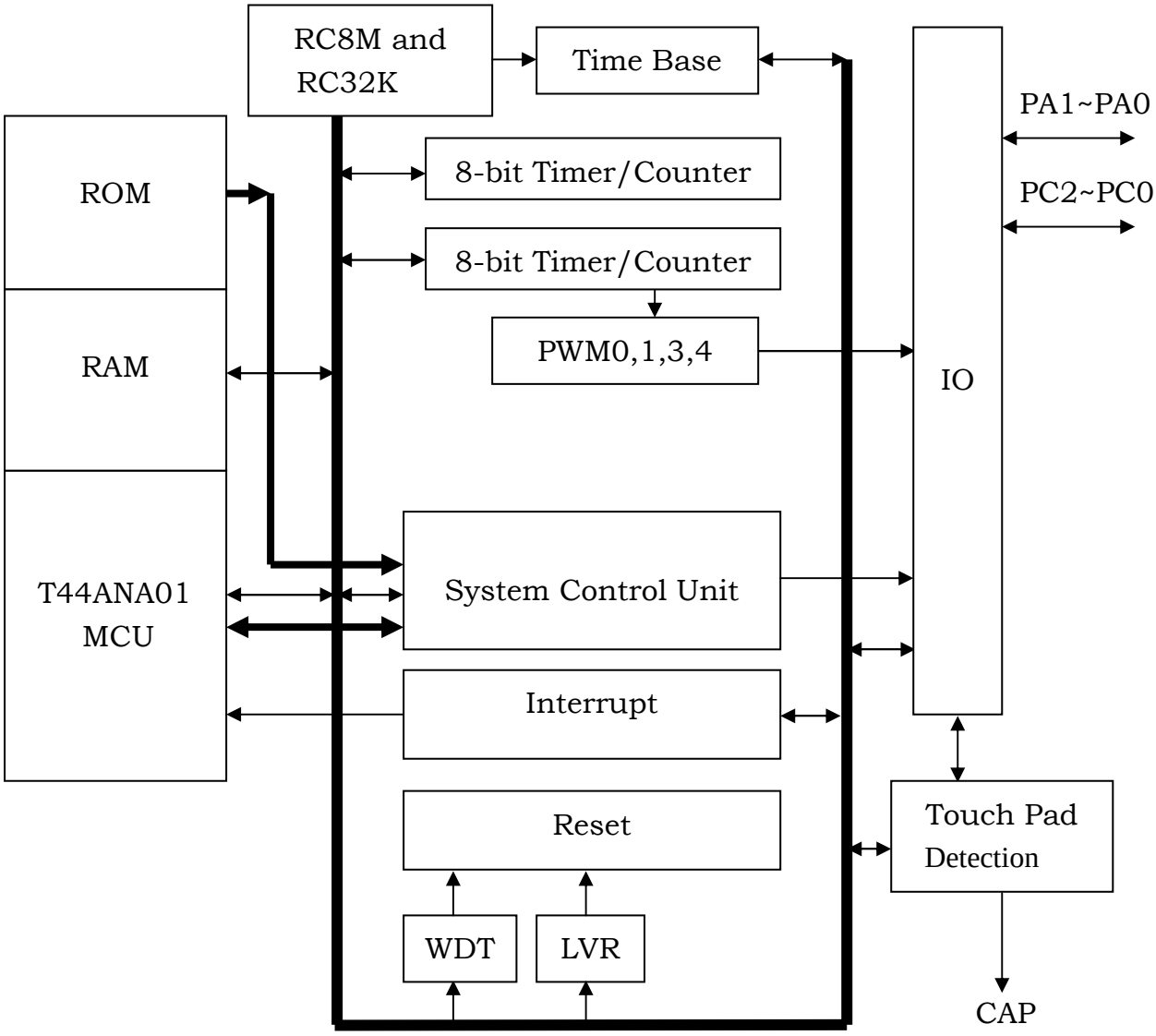


SOT23-6



DFN8L-A

§ Block Diagram:



§ Pin Description:

Pin Name	Share Pin	IO	Pin No.	Mask Option	Pin Description
VDD	-	Power	+1	-	Positive power supply.
VSS	-	Power	+1	-	Negative power supply, ground.
PA0 PA1	INT0/PWM0/VPP INT1/PWM1	IO/I/O IO/I/O	+2	-	IO port with external interrupt input and PWM output. PA0 is shared with external interrupt input, PA0,PA1 is shared with PWM output. PA0 input with pull down resistor and without wake-up function. PA0 output is open drain driver.
PC0 PC1 PC2	TP0/PWM3 TP1/PWM4 TP2	IO/I IO/I IO/I	+3	-	IO port with PWM output or touch pad input. PC0,PC1 is shared with PWM output.
CAP	-	O	+1	-	Touch signal output.
			8	-	

Note: PA0 share with MTP VPP pin, and the input voltage does not exceed the absolute maximum rating Vin VSS-0.3V~VDD + 0.3V.

§ IO Cell Type Description:

Pin Name	IO Cell Type	Description
PA0	Figure IO-C	STD IO with internal PWM output and external interrupt trigger input with pull down resistor and without the wake-up function .
PA1	Figure IO-B	STD IO with internal PWM output and external interrupt trigger
PC0~PC1	Figure IO-D	STD IO with internal PWM output and touch pad input.
PC2	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
Input Voltage	Vin	VSS-0.3~VDD+0.3	V
Human Body Mode	ESD	MIL-STD Class 3A (4KV~8KV)	
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_{CPU}=4MHz, LVR on 2.0V	2.2	-	5.5	V
		F_{CPU}=4MHz, LVR on 2.7V	3.5	-	5.5	
Low Voltage Reset (LVR)	V_{LVR1}	LVR select 2.0V	-	2.0	2.1	V
	V_{LVR2}	LVR select 2.7V	-	2.7	3.3	V
Operating Current (Normal Mode, CPU working, IO no load)	I_{nd1}	VDD=5.0V, no load, RC32K on, RC8M on, F_{CPU}=4MHz LVR on	-	2.5	3.0	mA
	I_{nd2}	VDD=5.0V, no load, RC32K on, RC8M off, F_{CPU}=32KHz LVR on	-	30	50	uA
Operating Current (Sleep Mode, CPU stop, IO no load)	I_{sd1}	VDD=5.0V, no load, RC32K on, RC8M on, LVR on	-	0.9	1.3	mA
	I_{sd2}	VDD=3.0V, no load, RC32K on, RC8M off, LVR on	-	7	12	uA
Standby Current (Stop Mode, CPU stop,IO no load)	I_{sd3}	VDD=5.0V, no load, RC32K off, RC8M off,LVR on	-	-	3	uA
LVR Current	I_{LVR}	VDD=5.0V	-	1	2	uA
Input Ports	V_{IL1}	Input Low Voltage	0	-	0.2	VDD
Input Ports	V_{IH1}	Input High Voltage	0.8	-	1.0	VDD
INT	V_{IL2}	Input Low Voltage	0	-	0.3	VDD
INT	V_{IH2}	Input High Voltage	0.7	-	1.0	VDD
Output port Sink Current (exclude PA0)	I_{OL2}	VDD=5.0V, VOL=0.6V	-	8	-	mA
Output Port Source Current (exclude PA0)	I_{OH2}	VDD=5V, VOH=4.3V	-	-4	-	mA
PA0 Sink Current (open drain)	I_{OL1}	VDD=5.0V, VOL=0.6V	-	8	-	mA
IO Port Pull-up Resistor (exclude PA0)	R_{PH1}	VDD=5.0V	100	150	200	KΩ
PA0 Pull-down Resistor	R_{PD1}	VDD=5.0V	35	50	65	KΩ

§ 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 users' software.		4	-	-	OSCL
System Oscillator Frequency	RC8M (Built-in RC)	VDD=5.0V	7.6M	8M	8.4M	Hz
Peripheral Oscillator Frequency	RC32K (Built-in RC)(OSCL)	VDD=5.0V	22K	32K	42K	Hz
Startup Period Of system clock	T_{OSCH} (Built-in RC)	Wake-up from off mode	8	-	-	OSCH
	T_{OSCL} (Built-in RC)	Wake-up from off mode	8	-	-	OSCL
Stable time Of System Clock Switching	T_{OSCH} (Built-in RC)	OSCL→OSCH & OSCH off	8	-	-	OSCH
	(If H/L=0 then OSCH stop)					
	T_{OSCL} (Built-in RC)	OSCH→OSCL & OSCL on	8	-	-	OSCL
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 ~$7BF_H$	-	Program ROM [1984*16]
-	000_H ~007_H	File Registers
-	008_H~$01F_H$	Peripheral registers (I)
-	020_H~$0AF_H$	Working RAM [144*4]
-	200_H~303_H	Peripheral registers (II)

§ Interrupt Vectors:

Interrupt Vectors	Function Description
000_H	Hardware reset
001_H	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	-000	CPU 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	--00	Extended interrupt enable control register
$00D_H$	INTF1	R/W	--00	Extended interrupt request flag register
$00E_H$	TPINTC	R/W	00--	Touch pad interrupt control register
$00F_H$	TPINTF	R/W	00--	Touch pad interrupt request flag register
010_H	PAC	R/W	--11	Port A IO control register
011_H	PA	R/W	--11	Port A output data register
014_H	PCC	R/W	-111	Port C IO control register
015_H	PC	R/W	-111	Port C output data register
200_H	PAI	R	----	Port A pad data reading address register
202_H	PCI	R	----	Port C pad data reading address register
$20A_H$	TBC	R/W	1111	Time base control register
$20B_H$	CPUFS	R/W	--00	CPU frequency division's register
$20C_H$	INTTS	R/W	0000	INT trigger type selector register
$20D_H$	SPCON0	R/W	000-	Special control 0 register
210_H	TCPFS	R/W	-000	TCP clock source FS pre-scale register
211_H	TCP1C	R/W	0000	TCP1 Timer/counter control register
212_H	TCP1L	R/W	xxxx	TCP1 Timer/counter low nibble data register
213_H	TCP1H	R/W	xxxx	TCP1 Timer/counter high nibble data register
214_H	TCP2C	R/W	0000	TCP2 Timer/counter control register
215_H	TCP2L	R/W	xxxx	TCP2 Timer/counter low nibble data register
216_H	TCP2H	R/W	xxxx	TCP2 Timer/counter high nibble data register
219_H	ADJSTAT	R	---1	Frequency adjustment status flag register
$21A_H$	TBLDRL	R	0000	Time base preload low nibble data register
$21B_H$	TBLDRH	R	1000	Time base preload high nibble data register

$21C_H$	LVRCON	R/W	---0	LVR control register
220_H	PWMSTS0	R/W	0-00	PWM start level selector register 0
221_H	PWMSTS1	R/W	---0	PWM start level selector register 1
224_H	PWMC0	R/W	0-00	PWM control register 0
225_H	PWMC1	R/W	---0	PWM control register 1
226_H	PWM0L	R/W	xxxx	PWM0 duty low nibble data register
227_H	PWM0H	R/W	xxxx	PWM0 duty high nibble data register
229_H	PWM1L	R/W	xxxx	PWM1 duty low nibble data register
$22A_H$	PWM1H	R/W	xxxx	PWM1 duty high nibble data register
$22F_H$	PWM3L	R/W	xxxx	PWM3 duty low nibble data register
230_H	PWM3H	R/W	xxxx	PWM3 duty high nibble data register
232_H	PWM4L	R/W	xxxx	PWM4 duty low nibble data register
233_H	PWM4H	R/W	xxxx	PWM4 duty high nibble data register
240_H	MCKS	R/W	-111	Modulation clock selector register
241_H	TPCON0	R/W	0000	Touch pad control 0 register
243_H	TPCHS0	R/W	-000	Touch pad channel selector 0 register
248_H	TPCTL	R/W	-000	Touch pad control register
249_H	TPCT0	R/W	1111	Touch pad duty counter & latch data 0 register
$24A_H$	TPCT1	R/W	1111	Touch pad duty counter & latch data 1 register
$24B_H$	TPCT2	R/W	1111	Touch pad duty counter & latch data 2 register
300_H	RESETF	R/W	-000	Reset source flag register
301_H	TBRB	W	----	Time base clear address register
302_H	MRO	W	----	Mask option register enable address register
303_H	CLRWDT	W	----	Clear WDT 2nd instruction address 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. The frequency range between 7.6MHz~8.4MHz (typical at VDD=5V). The system clock(OSCH) is system oscillator frequency divided by 2.

S-2: Peripheral oscillator

The low speed oscillator is built-in an internal RC oscillator that is for low power consumption consideration and fixed peripheral device timing control. The frequency range between 22KHz~42KHz (typical at VDD=5V). The peripheral clock(OSCL) is peripheral oscillator frequency.

S-3: CPU clock

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

✧ CPUFS[20BH]: CPU frequency division's register [R/W], power on value [--00]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	--	--	CS1	CS0
Read/Write	--	--	R/W	R/W

CS1~CS0: The selector value of CPU division's register.

CS1~CS0	CPU frequency	OSCH=4MHz
00	OSCH/1	4MHz
01	OSCH/2	2MHz
10	OSCH/4	1MHz
11	OSCH/8	500KHz

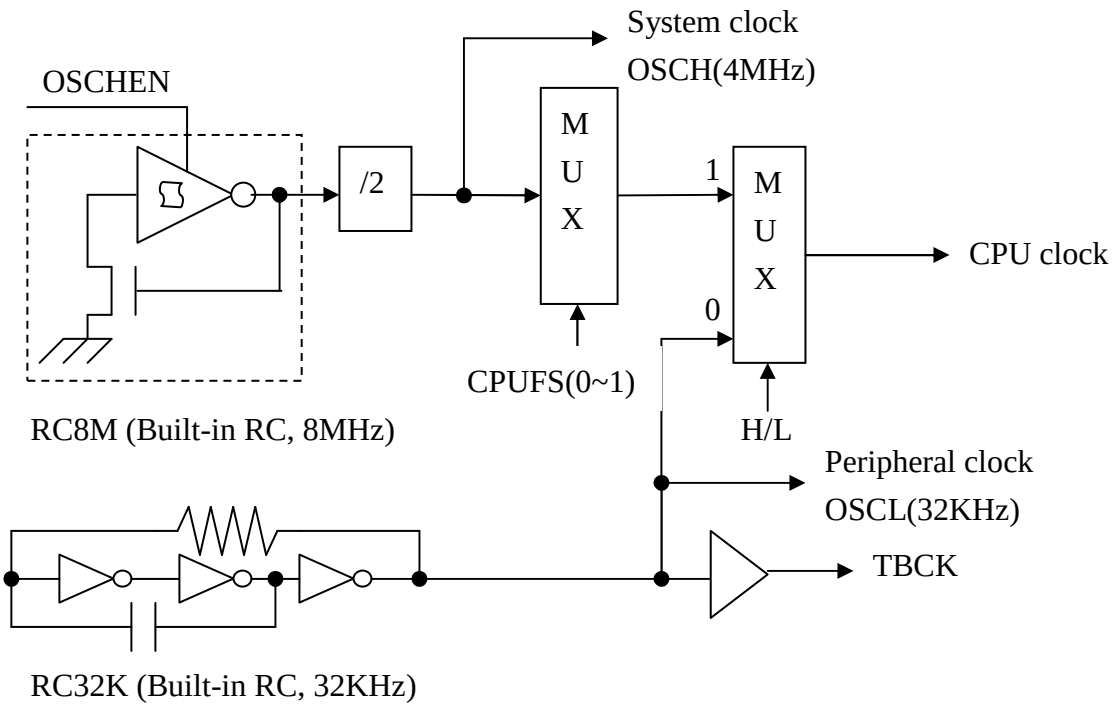


Figure: System/Peripheral Oscillator and CPU Clock Sources

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.

When the power saving mode is switched, do not perform it at the same time as the high and low speed switching. Switch to low speed before setting the power saving mode. The instruction that follow the STOP or SLEEP mode (set by PS register) must be NOP, and two NOP instructions must be followed, it make sure to wake up success.

example: set low speed sleep from high speed.

```
STX     #$0,PS
STX     #$2,PS
NOP
NOP
```

S-5: MCU System Operation Mode

The MCU has 4 operating modes, including high-speed operation, low speed operation, sleep and stop modes. The MCU will go into low speed operation mode (RC32K on and RC8M off) automatically after power on or reset release. After wake up from sleep mode, the MCU will resume the last operation mode.

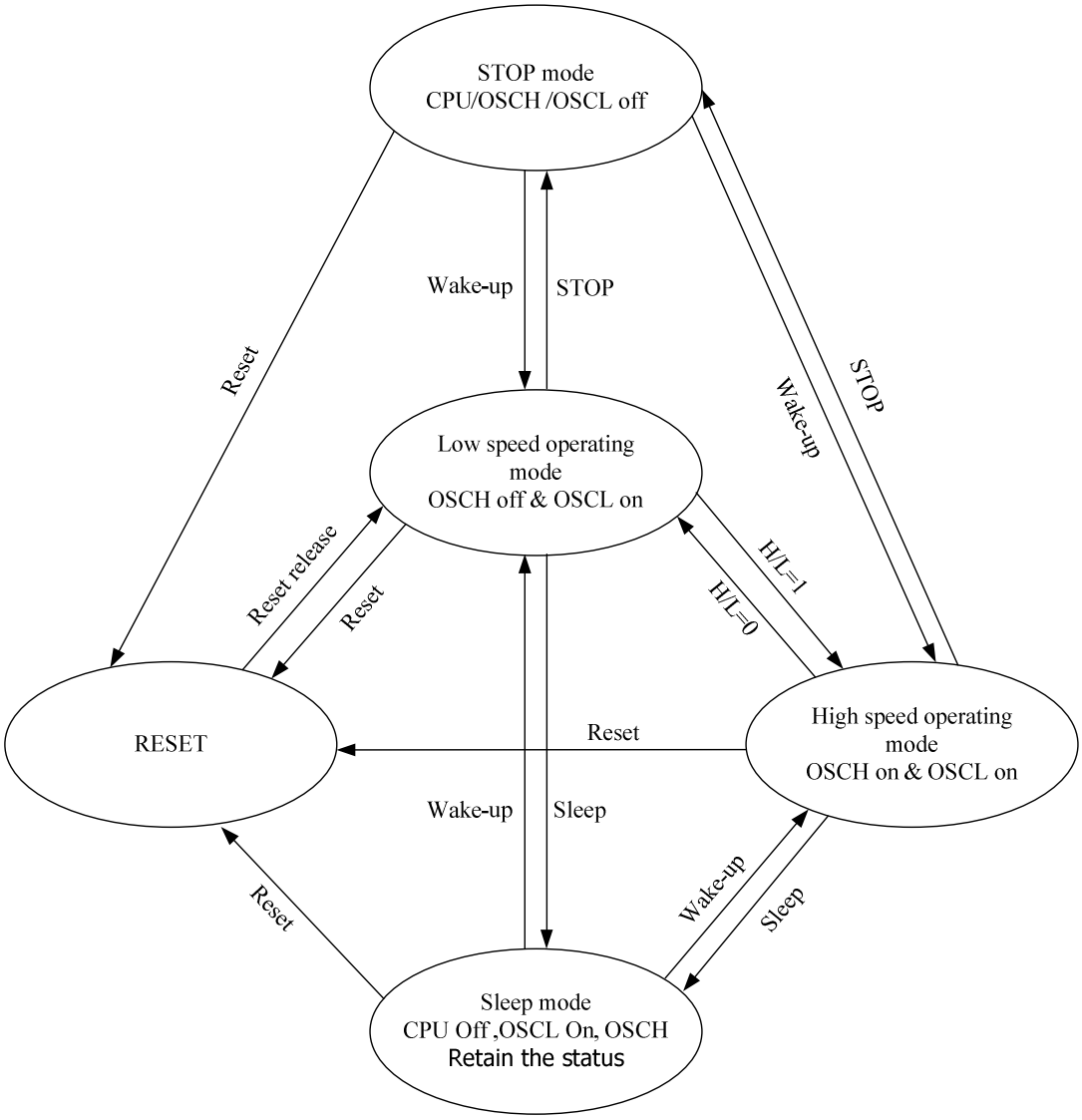


Figure: System Operation State Diagram

* 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_H 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. Watch dog timer will be clear when reset, CPU enter sleep mode or stop mode.

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

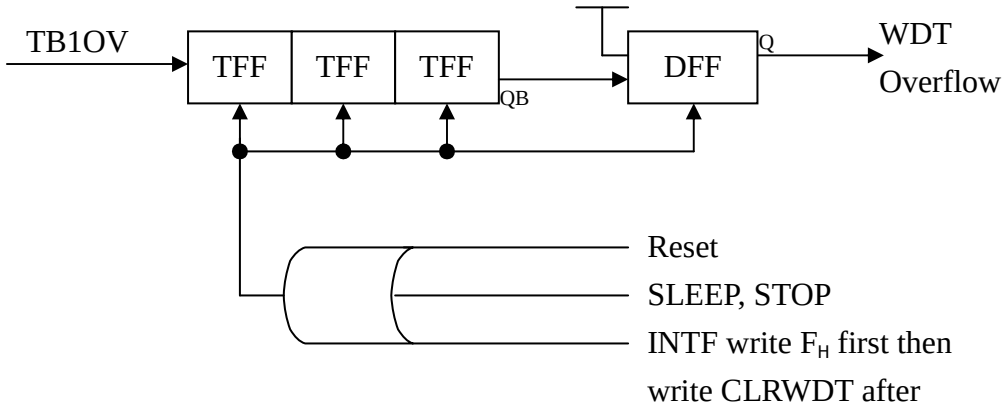


Figure: Watch Dog Timer control circuit

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. This function can not be turn off. The detected voltage locates small than 2.0V or 2.7V is selected by LVRCON register. The voltage in the table would have a little tolerance in different lot of chip and environment.

✧ LVRCON[21CH]: LVR control register [R/W] , default value [---0]

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

LVRVS: Low voltage reset voltage selector. (0: 2.0V; 1: 2.7V)

S-9: Reset

The chip has four kinds of reset sources: POR (power on reset), Watch dog timer overflow reset, LVR (low voltage reset) and Burn out 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 system 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.

.Watch dog timer overflow reset

The system reset signal will generate automatically when the watch dog timer runs overflow. If 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 a system 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 system reset signal (with 8 OSCL de-bounce time) to reset the microcontroller as the VDD power falls below the default setting level VLVR.

.Burn out reset (Program sequence abnormal)

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

✧ RESETF[300H]: Reset source flag register [R/W], power on value [-000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	BOF	LVRF	WDTF
Read/Write	-	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 reset flag. (0: no active; 1: active)
Note: The RESETF is only cleared by power on reset.

S-10. Power Saving Control Register

✧ PS[008H]: Power saving control register [R/W] , default value [-000]

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 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.
When the power saving mode is switched, do not perform it at the same time as the high and low speed switching. Switch to low speed before setting the power saving mode. The instruction that follow the STOP or SLEEP mode (set by PS register) must be NOP, and two NOP instructions must be followed, it make sure to wake up success.
example: set low speed sleep from high speed.

```
STX     #$0,PS
STX     #$2,PS
NOP
NOP
```

S-11. Special control register

✧ SPCON0 [20DH]: Special control 0 register [R/W] , default value [000-]

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

CDSC: Charge and discharge sequence control for touch sensor function

CDSC2 ~ CDSC0	Sequence change clock
000	OFF
001	2
010	4
011	6
100	8
101	12
110	16
111	Reserve

S-12. OST time

The system/peripheral 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

S-13. Interrupts

The CPU provides only 1 interrupt vector (001_H) and no priority, but can expand to multi-sources. Interrupt source includes external interrupts (INTxINT), timer/counter interrupts (TCPxINT), time base interrupt (TBxINT) or other peripheral device interrupt request (PERxINT). The interrupt control registers (INTC or INTC1) contain the interrupt control bits to enable and disable corresponding interrupt request and the corresponding interrupt request flags in the (INTF or INTF1) registers. Before finishing the interrupt service routine, another interrupt request will keep waiting until program return from interrupt routine.

If the interrupt request needs service, the programmer may set the corresponding interrupt enable bit to allow interrupt active. External interrupts can select trigger type by setting INTTS register, and set the related interrupt request flag (INTxF). The internal timer/counter interrupt is setting the TCPxF to 1, resulting from the timer/counter overflow. The time base interrupt 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.

✧ INTTS[20CH]: Interrupt trigger type selector register [R/W], default value [0000]

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

INT0S1~INT0S0: Interrupt 0 (INT0) trigger type selector.

00: Low level trigger.

01: Falling edge trigger.

10: Rising edge trigger.

11: Dual edge trigger.

INT1S1~INT1S0: Interrupt 1 (INT1) trigger type selector.

00: Low level trigger.

01: Falling edge trigger.

10: Rising edge trigger.

11: Dual edge trigger.

✧ 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 [--00]

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

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

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

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

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

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

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

✧ TPINTC[00EH]: 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[00FH]: Touch pad interrupt 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: Timer/Counter clock pre-scale

The System clock is the most high frequency divided by 2 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[210H]: Timer/Counter 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

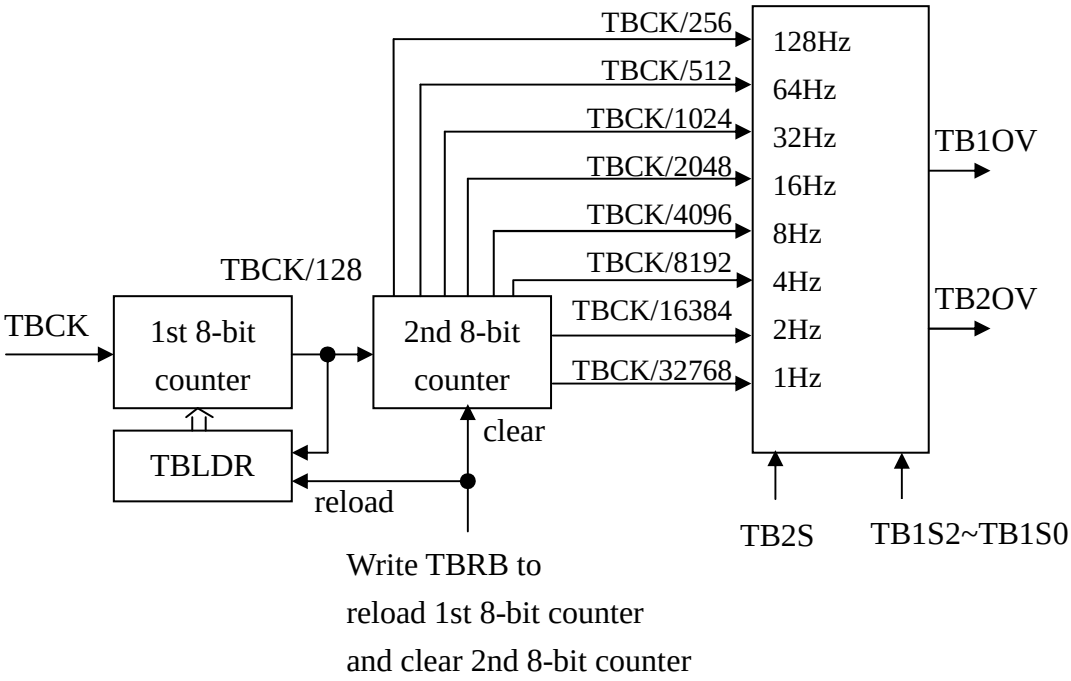
FS2~FS0: The selector of TCPFS.

FS2~FS0	FS	FS2~FS0	FS
000	OSCH/1	100	OSCH/16
001	OSCH/2	101	OSCH/32
010	OSCH/4	110	OSCH/64
011	OSCH/8	111	OSCH/128

OSCH: RC8M/2=4MHz.

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[20AH]: 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~TB1S0	**TB1OV**	**TBCK=32KHz**
000	TBCK/256	128HZ
001	TBCK/512	64HZ
010	TBCK/1024	32HZ
011	TBCK/2048	16HZ
100	TBCK/4096	8HZ
101	TBCK/8192	4HZ
110	TBCK/16384	2HZ
111	TBCK/32768	1HZ

TB2S	**TB2OV**	**TBCK=32KHz**
0	TBCK/1024	32Hz
1	TBCK/2048	16Hz

P-2-1: Adjustment time base

✧ ADJSTAT[219H]: Frequency adjustment status flag register [R], default value [---1]

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

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

✧ TBLDRL[21AH]: Time base preload low nibble data register [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 low nibble data.

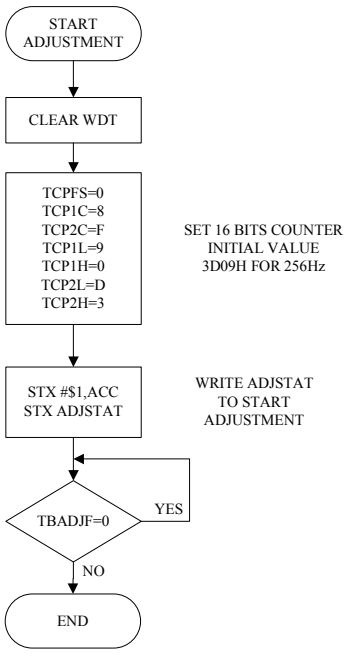
✧ TBLDRH[21BH]: Time base preload high nibble data register [R], default value [1000]

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

TBLDR7~TBLDR4: Time base preload high nibble data.

User can adjustment the time base for accurate 256Hz by modify 1st 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 256Hz, there is using TCP1 and TCP2 cascaded to form a 16-bit timer/counter, chooses clock source FS and set TCPFS=0_H for TCP1, then load $3D09_H$(4MHz/15625=256Hz) to the 16-bit counter and write 1_H to ADJSTAT register to start adjustment, then check TBADJF flag. The adjustment procedure is finished when TBADJF=1.

The adjustment procedure flow chart as follow:



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

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

✧ TCP1C[211H]: 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 enabled. (0: disable; 1: enable)

TCP1LD: TCP1 auto-reload enabled. (0: disable; 1: enable)

TCP1S1~TCP1S0: TCP1 clock source selector.

TCP1S1~TCP1S0	TCP1 clock source
00	FS
01	RC8M
10	TBCK
11	TB1OV

✧ TCP1L[212H]: TCP1 low nibble data register [R/W], default value [xxxx]

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[213H]: TCP1 high nibble data register [R/W], default value [xxxx]

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 [xxH]

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: 8-bit Timer/Counter/PWM for TCP2

One 8-bit timer/counters (TCP2) with 4 kind clock sources and preload data buffer can implement as a timer or counter feature. The clock source of TCP2 is 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 interrupt flag.

✧ TCP2C[214H]: 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 enabled. (0: disable; 1: enable)
TCP2LD: TCP2 auto-reload enabled. (0: disable; 1: enable)
TCP2S1~TCP2S0: TCP2 clock source selector.

TCP2S1~TCP2S0	TCP2 clock source
00	FS
01	RC8M
10	TBCK
11	TCP1OV

✧ TCP2L[215H]: TCP2 low nibble data register [R/W], default value [xxxx]

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.

✧ TCP2H[216H]: TCP2 low high data register [R/W], default value [xxxx]

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 high nibble data.
TCP2D7~TCP2D4: Writing TCP2D high nibble of data buffer.

* TCP2D: Like a 8-bit TCP2 data register [R/W], default value [xxH]

TCP2D	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Bit Name	TCP2D7	TCP2D6	TCP2D5	TCP2D4	TCP2D3	TCP2D2	TCP2D1	TCP2D0

The special R/W function for TCP2 has different Target, AS writing TCP2H/L registers that are updating preload data of the TCP2D. As read TCP2H/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 timer/counter:
Tc = (selected clock cycle) * (256) if TCPxD=00H
Tc = (selected clock cycle) * (TCPxD) otherwise

When 16-bit timer/counter:
Tc = (selected clock cycle) * (65536) if TCP1D=00H and TCP2D=00H
Tc = (selected clock cycle) * (TCP2D*256+TCP1D) otherwise

When user writes data to the TCPxH/L, the data just keep in TCPxH/L latch. During the TCPxEN=1 command executed, the TCPxH/L latch's complement value will load into counter TCPxH/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 for TCPx. At the same time, interrupt request flag TCPxF will set activated, if software enables the corresponding interrupt enable bit, interrupt 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 counter is not enough for counting, user can enable the interrupt and using the data RAM as software counter for extending the counter stage.

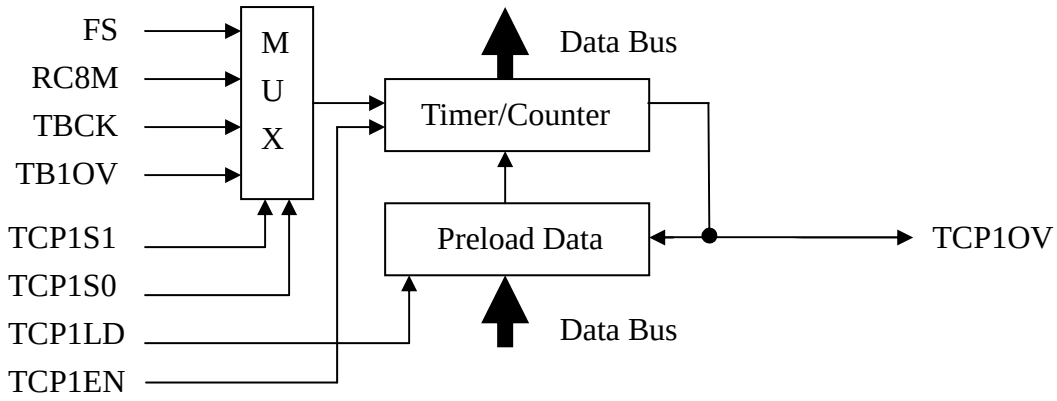


Figure: 8-bit Timer/Counter (TCP1)

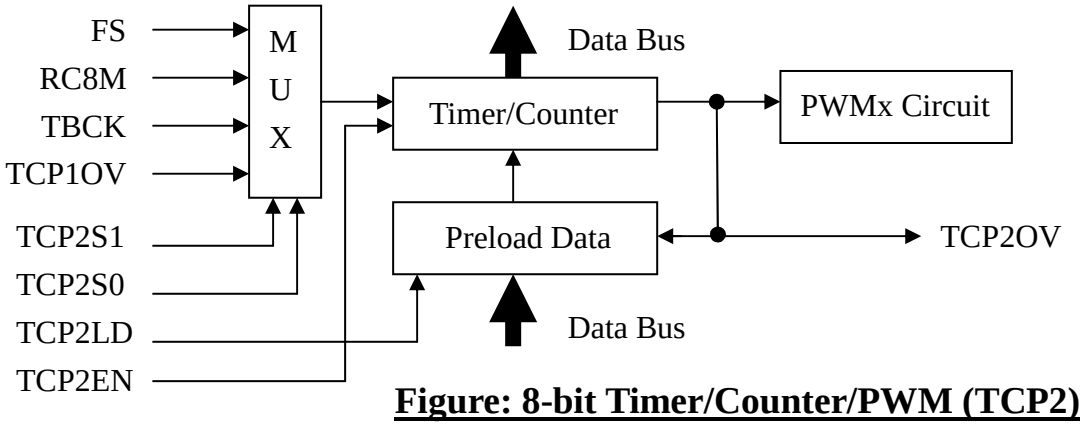


Figure: 8-bit Timer/Counter/PWM (TCP2)

TCP1S1~TCP1S0	TCP1 clock source
00	FS
01	RC8M
10	TBCK
11	TB1OV

TCP2S1~TCP2S0	TCP2 clock source
00	FS
01	RC8M
10	TBCK
11	TCP1OV

	PWM Output
TCP2	PWM0,1,3,4

FS: Clock scaled frequency comes from OSCH.
RC8M: Bulit-in high frequency RC oscillator.
TBCK: Peripheral clock source, 32KHz in the RC mode.
TB1OV: Time base 1st overflow output.
TCP1OV: TCP1 overflow output.
PWM0~1, 3~4: TCP2 cycle time with PWMxD duty output signal.

P-5: 16-bit Timer/Counter (TCP1 and TCP2 cascade)

Two sets TCP can be cascaded to form a 16-bit timer/counter when TCP2 chooses TCP1OV as clock source (TCP2S1=1 and TCP2S0=1). In the 16-bit timer application, user should use TCP1EN to control the starting or stopping counting of 16-bit timer/counter, data load is controlled by writing TCP1EN=1. The rising TCP2OV will reload the contents in the preload register into timer/counter if TCP2LD=1. The interrupt feature is different, in this case, the TCP1 interrupt will be inhibit when TCP1OV occur, the TCP2 interrupt is normally.

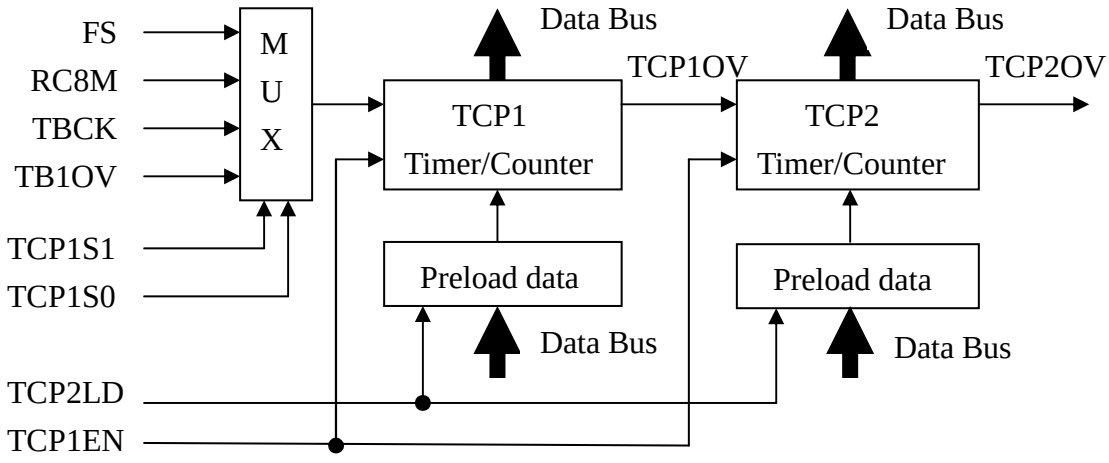


Figure: 16-bit Timer/Counter (TCP1 and TCP2 cascade)

.PWM

The PWM period generated from TCP2. When PWMxEN enable, and PWMOUT pin (PA0, PA1, PC0, PC1 must be output mode) 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 and PWMxH, user write PWMxH first, last write PWMxL. When write the PWMxL, the duty value will be load to PWMxD at the same time. PWM's duty value cannot bigger than TCP2 preload data. If not, PWMOUT is an unexpected signal.

User can select PWMOUT pin start with 1 or start with 0 by PWMSTS register. When TCP2 enable, timer/counter start counting. If timer/counter value equal to PWMx duty's complement value, PWMOUT will change state. The PWMOUT back to start state, When TCP2 is overflow.

User does not use PWM in 16-bit timer/counter mode. If not, PWMOUT is an unexpected signal.

User does not use TCP2D=00_H. If not, PWMOUT is an unexpected signal.

✧ PWMSTS0[220H]: PWM start level selector register 0 [R/W], default value [0-00]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM3S	-	PWM1S	PWM0S
Read/Write	R/W	-	R/W	R/W

PWM0S: PWM0 Start level selector. (0: Start 0; 1: Start 1)

PWM1S: PWM1 Start level selector. (0: Start 0; 1: Start 1)

PWM3S: PWM3 Start level selector. (0: Start 0; 1: Start 1)

✧ PWMSTS1[221H]: PWM start level selector register 1 [R/W], default value [---0]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	-	PWM4S
Read/Write	-	-	-	R/W

PWM4S: PWM4 Start level selector. (0: Start 0; 1: Start 1)

✧ PWMC0[224H]: PWM control register 0 [R/W], default value [0-00]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM3EN	-	PWM1EN	PWM0EN
Read/Write	R/W	-	R/W	R/W

PWM0EN: PWM0 output enabled. (0: disable; 1: enable)

PWM1EN: PWM1 output enabled. (0: disable; 1: enable)

PWM3EN: PWM3 output enabled. (0: disable; 1: enable)

✧ PWMC1[225H]: PWM control register 1 [R/W], default value [---0]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	-	-	-	PWM4EN
Read/Write	-	-	-	R/W

PWM4EN: PWM4 output enabled. (0: disable; 1: enable)

✧ PWM0L[226H]: 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.

✧ PWM0H[227H]: PWM0 duty high 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 high nibble data.

✧ PWM1L[229H]: 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.

✧ PWM1H[22AH]: PWM1 duty high 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 high nibble data.

✧ PWM3L[22FH]: PWM3 duty low nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM3D3	PWM3D2	PWM3D1	PWM3D0
Read/Write	R/W	R/W	R/W	R/W

PWM3D3~PWM3D0: PWM3 duty low nibble data.

✧ PWM3H[230H]: PWM3 duty high nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM3D7	PWM3D6	PWM3D5	PWM3D4
Read/Write	R/W	R/W	R/W	R/W

PWM3D7~PWM3D4: PWM3 duty high nibble data.

✧ PWM4L[232H]: PWM4 duty low nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM4D3	PWM4D2	PWM4D1	PWM4D0
Read/Write	R/W	R/W	R/W	R/W

PWM4D3~PWM4D0: PWM4 duty low nibble data.

✧ PWM4H[233H]: PWM4 duty high nibble data register [R/W], default value [xxxx]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	PWM4D7	PWM4D6	PWM4D5	PWM4D4
Read/Write	R/W	R/W	R/W	R/W

PWM4D7~PWM4D4: PWM4 duty high nibble data.

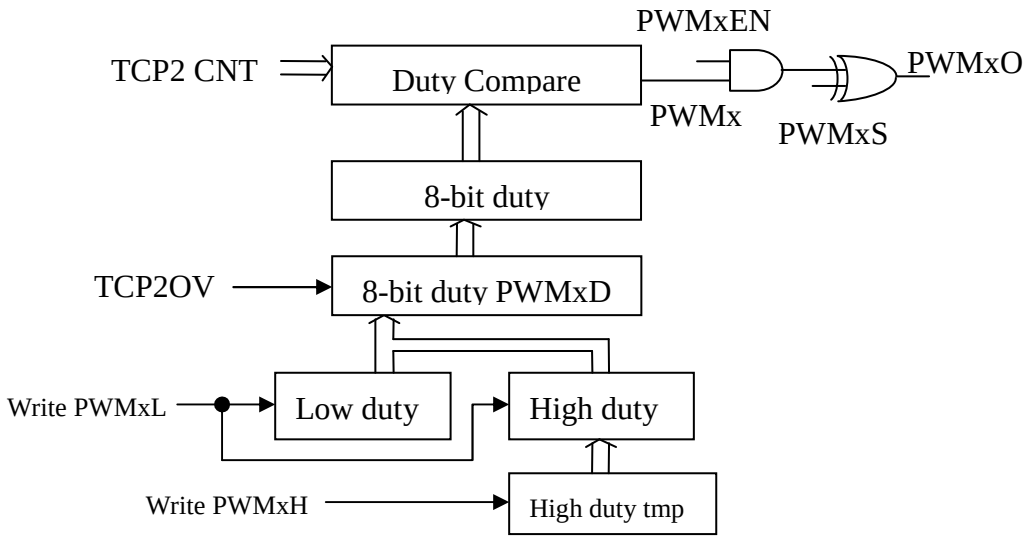


Figure: PWM (TCP2)

PWMxD	PWMx 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 cannot bigger than TCP2D.

2. TCP2 timer's period = (TCP2D) * clock cycle.
3. PWM0~1,3 can start 0 or start 1 by PWMSTS0 register.
4. PWM4 can start 0 or start 1 by PWMSTS1 register.

Table: PWMx duty

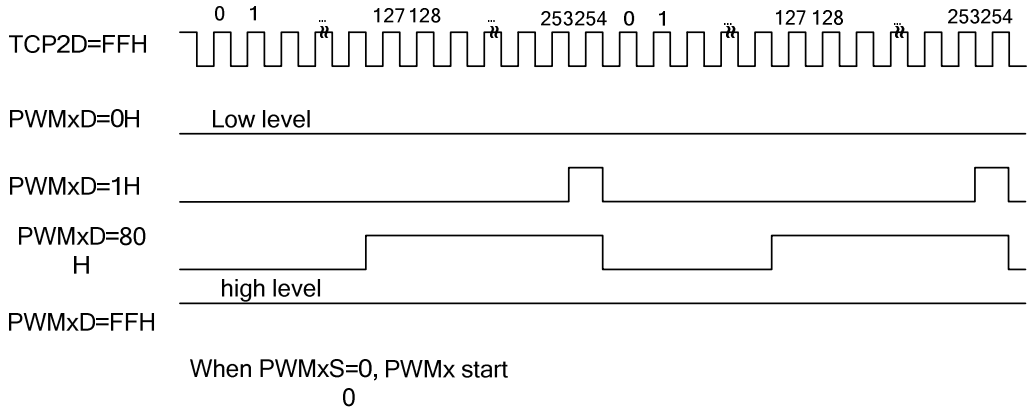


Figure: PWMx output start 0 when PWMxS=0

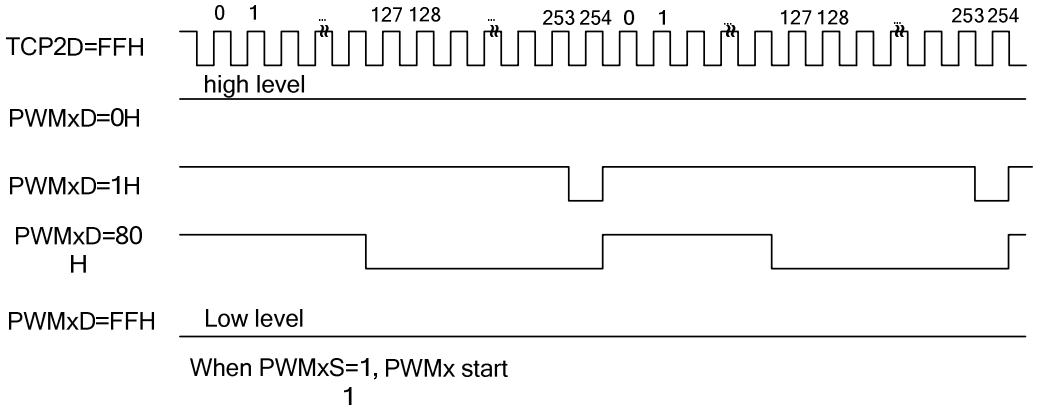


Figure: PWMx output start 1 when PWMxS=1

. IO Pad Cell Structure and Function Description

.. IO port

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 to 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

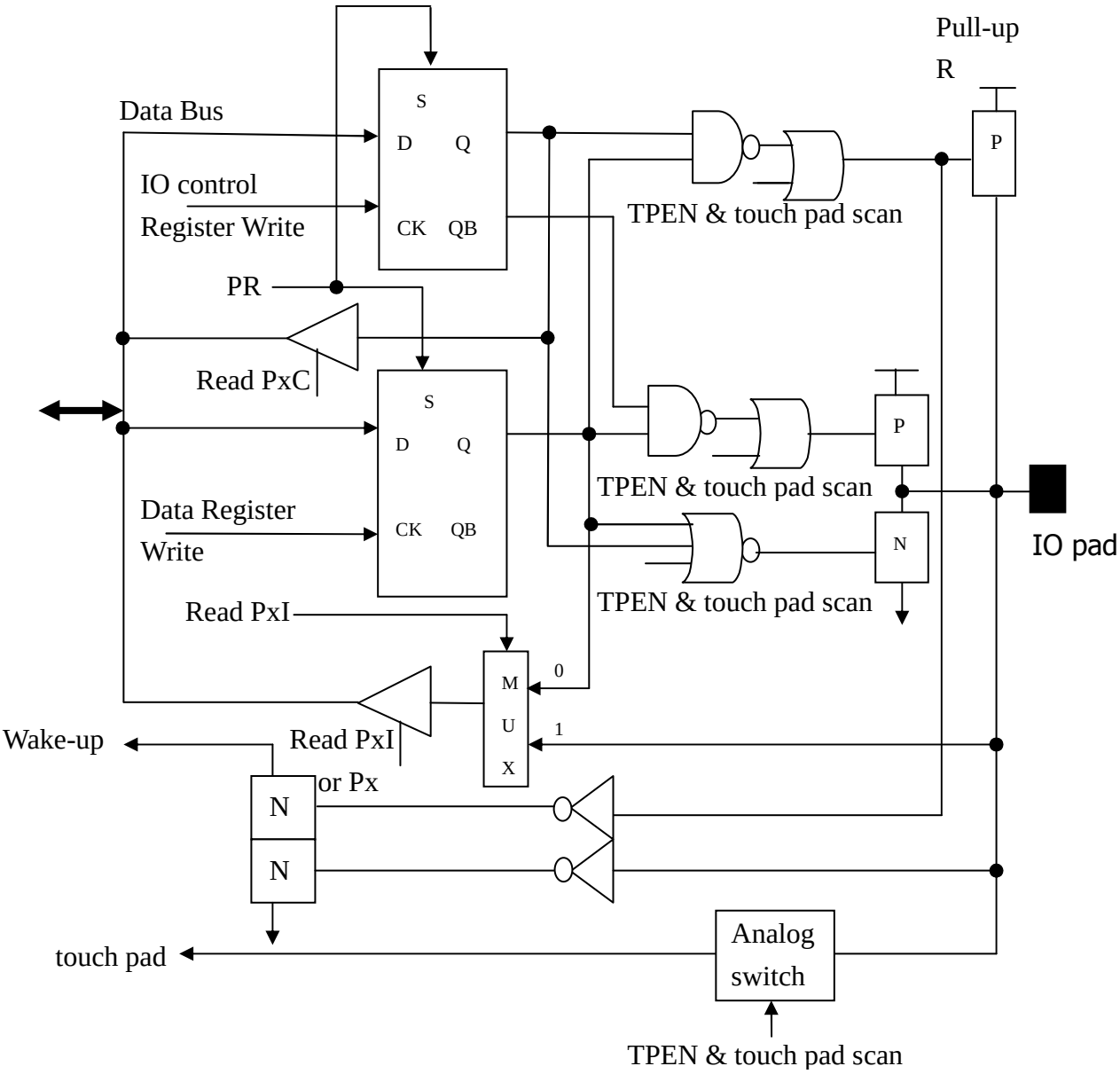


Figure IO-A: Standard IO port with touch pad input

.. IO port with internal PWM output 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

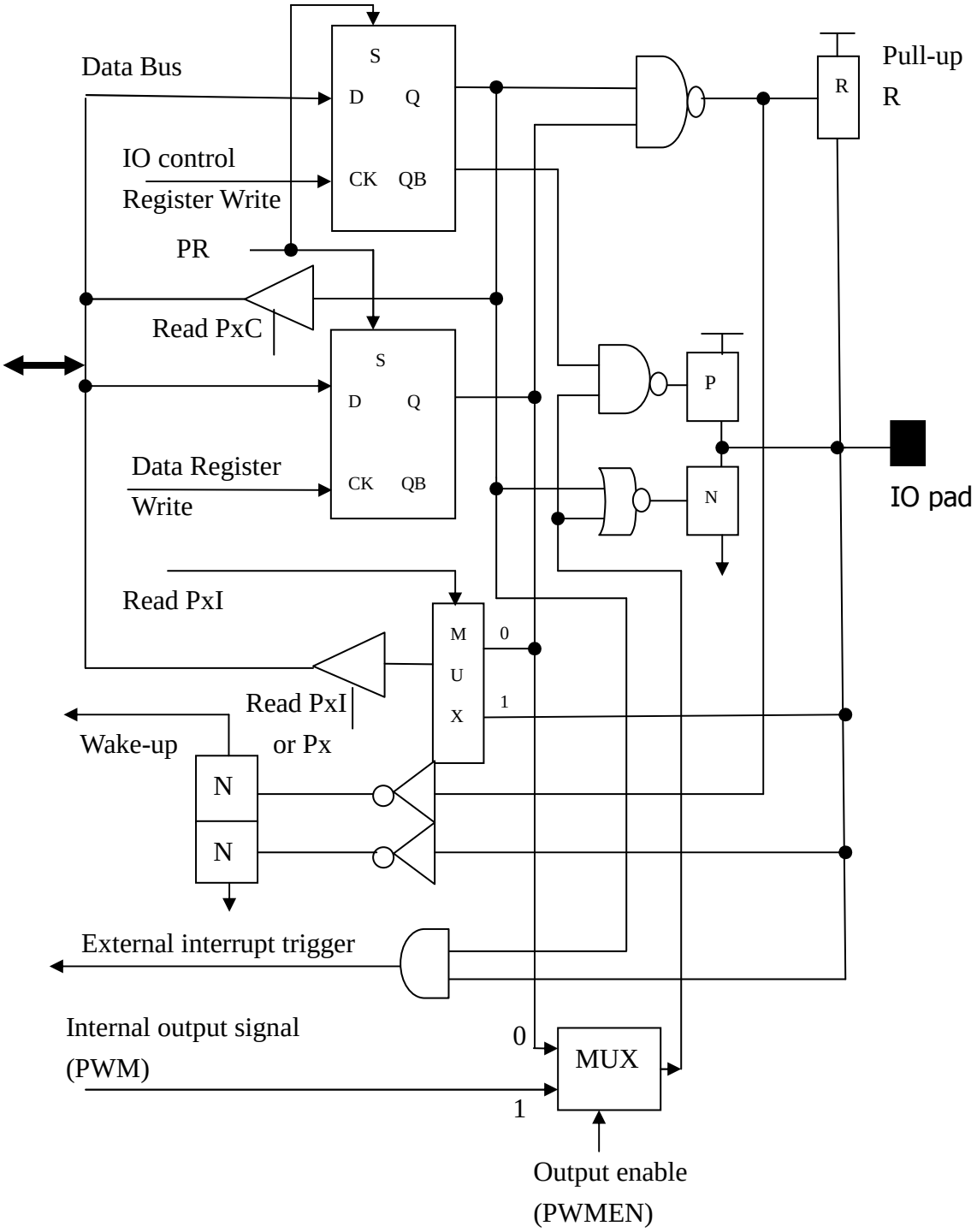


Figure IO-B: Standard IO port with internal PWM output and external interrupt trigger input

.. IO port with internal PWM output and external interrupt trigger input with pull down resistor and without the wake-up function

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-down resistor and without 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-down R	Wake-up feature	External input
0	X	No	No	No
1	0	No	No	Enable
1	1	Enable	No	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

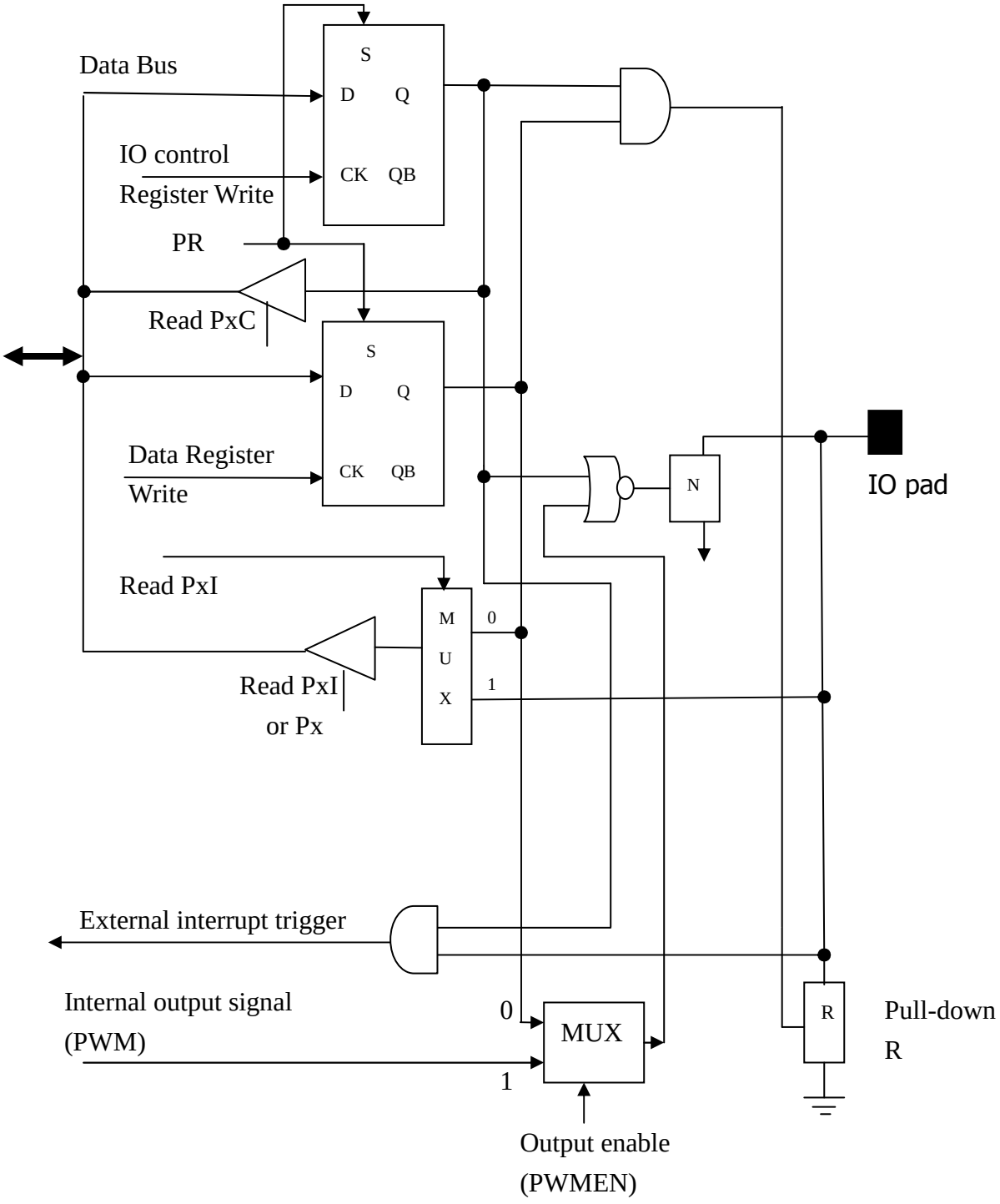


Figure IO-C: Standard IO port with internal PWM output and external interrupt trigger input with pull down resisitor

. IO Pad Cell Structure and Function Description

.. IO port with touch pad input and PWM Output

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

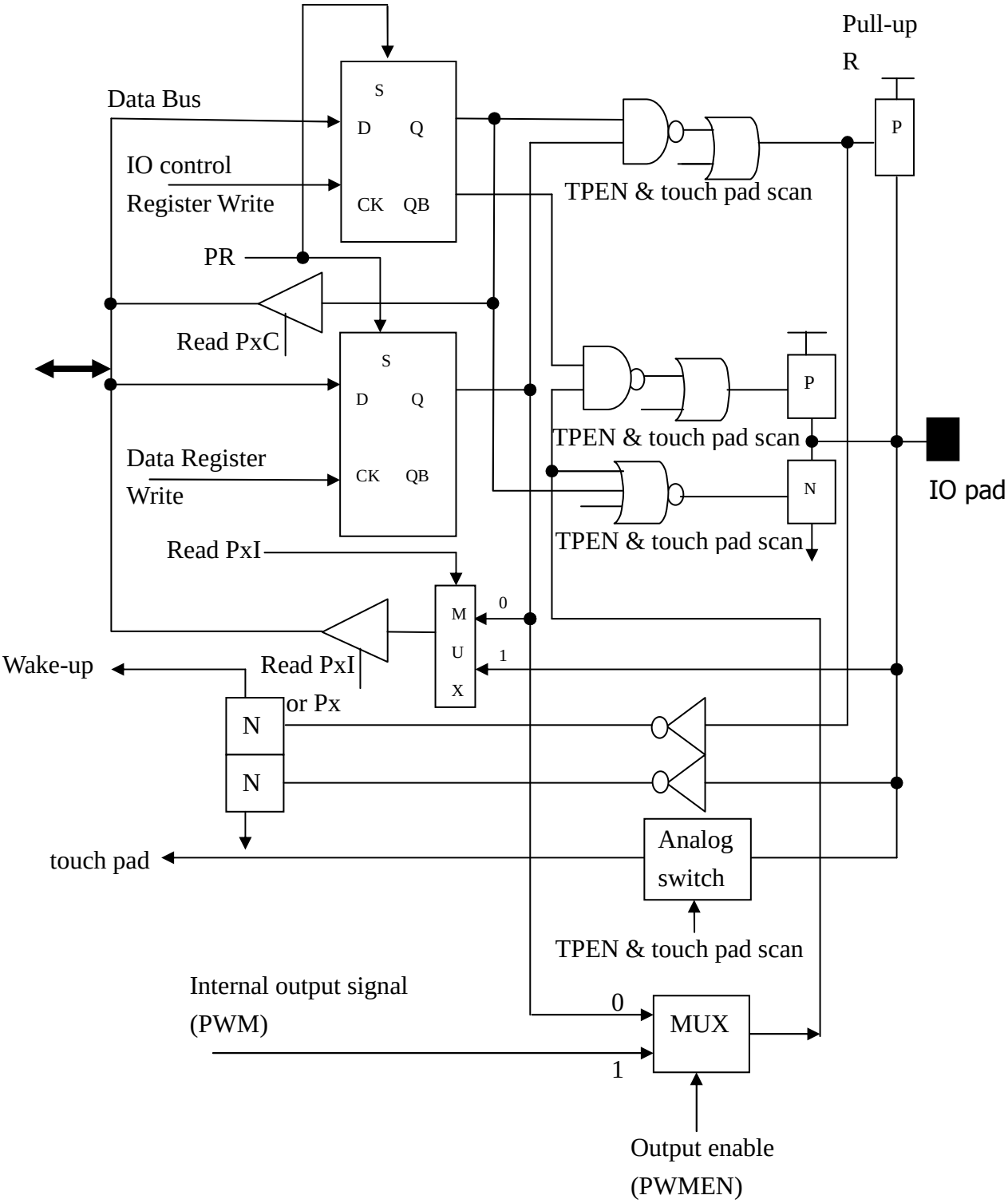


Figure IO-D: Standard IO port with touch pad input and PWM output

3. IO Pad Cells

The main features of 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.

PA0 input with pull down resistor and without wake-up function, the output is open drain driver.

. IO File Register

✧ PAC[010H]: Port A IO control register [R/W], default value [--11]

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

PAC1~PAC0: Port A IO control data.

✧ PA[011H]: Port A output data register [R/W], default value [--11]

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

PA1~PA0: Port A output data.

✧ PCC[014H]: Port C IO control register [R/W], default value [-111]

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

PCC2~PCC0: Port C IO control data.

✧ PC[015H]: Port C output data register [R/W], default value [-111]

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

PC2~PC0: Port C output data.

✧ PAI[200H]: Port A pad data reading address register [R], default value [----]

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

PAI1~PAI0: Port A pad data.

✧ PCI[202H]: Port C pad data reading address register [R], default value [----]

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

PCI2~PCI0: Port C pad data.

3 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.

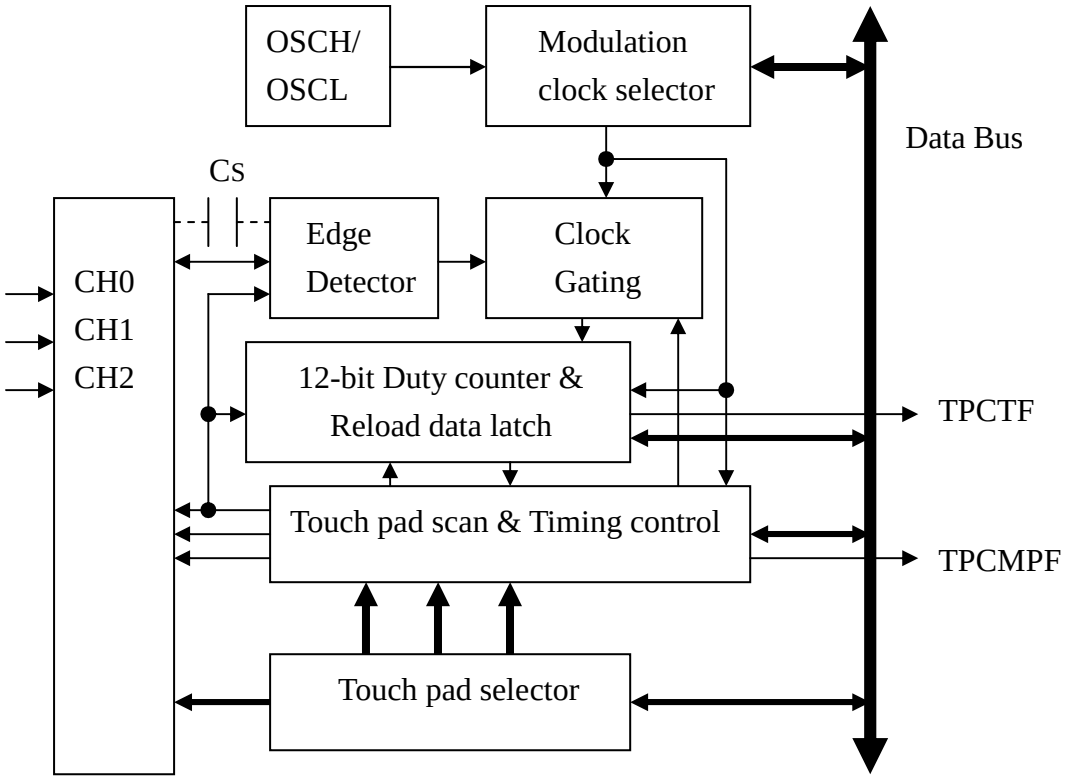
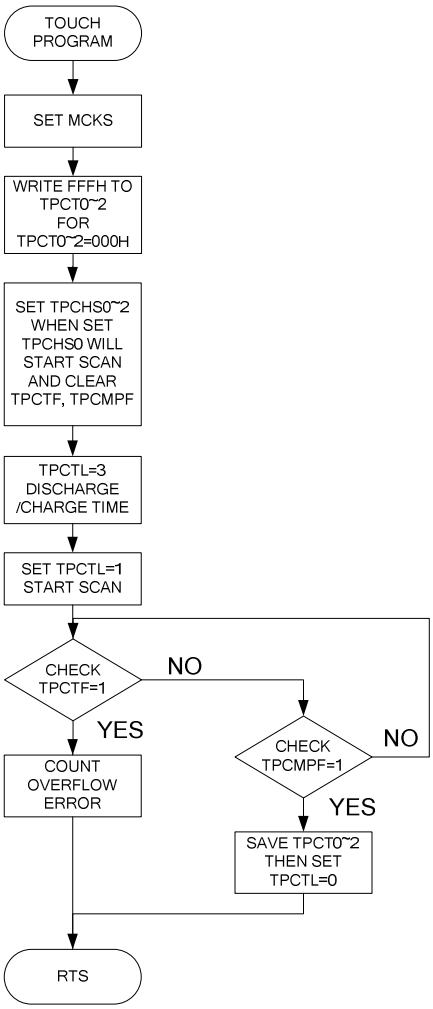


Figure: 3 pads touch pad detector

Parameters	Target Value	Remark
Touch pad clock	OSCH or OSCL	4MHz or 32KHz (typical)
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	1~3 Pad	-
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:



✧ SPCON0 [20DH]: Special control 0 register [R/W] , default value [000-]

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

CDSC2~CDSC0: Charge and discharge sequence control for touch pad detection function.

CDSC2~CDSC0	Sequence change clock
000	OFF
001	2
010	4
011	6
100	8
101	12
110	16
111	Reserve

✧ TPCON0 [241H]: Touch pad control 0 register [R/W], default value [0000]

Register	Bit3	Bit2	Bit1	Bit0
Bit Name	VREFS1	VREFS0	TPNIS	TPMODE
Read/write	R/W	R/W	R/W	R/W

TPMODE: Touch pad mode for touch pad scan.

0: CAP tie to VSS when discharge mode.

1: CAP tie to VDD when charge mode.

TPNIS: Touch pad detection type selector.

0: Touch pad detector use Schmitt trigger output signal.

1: Touch pad detector use comparator output signal.

VREFS1~VREFS0: Voltage reference selector for touch pad detector.

00: 1/2 VDD.

01: 2/3 VDD.

10: 1/3 VDD.

11: 1/3 VDD.

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

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

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

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

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

✧ TPCT0[249H]: Touch pad duty counter & latch data 0 register [R/W], default value [1111]

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 for counter read.

CT3~CT0: 1st nibble of reload latch data.

✧ TPCT1[24AH]: Touch pad duty counter & latch data 1 register [R/W], default value [1111]

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 for counter read.

CT7~CT4: 2nd nibble of reload latch data.

✧ TPCT2[24BH]: Touch pad duty counter & latch data 2 register [R/W], default value [1111]

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 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 any of the TPCHS0 addresses will reload the 12 bit counters and clear the TPCTF & TPCMPF.

MCKS[240H]: Modulation clock selector register [R/W], default value [-111]

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[243H]: Touch pad channel selector 0 register [R/W], default value [-000]

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

TPEN2~TPEN0: Touch pad channel selector 1st nibble.

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

Channel Enable State	TPCHS0 TPEN2~0
TP0	001
TP1	010
TP2	100

When TPCHS0 is writing, TPCTL will be set TP RUN mode, and touch pad begin to scan touch pad.

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

✧ TPCTL[248H]: 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 discharges the external capacitor and enables the sensor clock input until period end.

TPCTL2~TPCTL0	Channel Enable State
000	TP STOP
001	TP RUN
010	-
011	Discharge/Charge
100	
101	-
110	-
111	-

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

TP RUN: TP RUN is touchpad scan start signal, its scan the channel by TPCHS0 select.

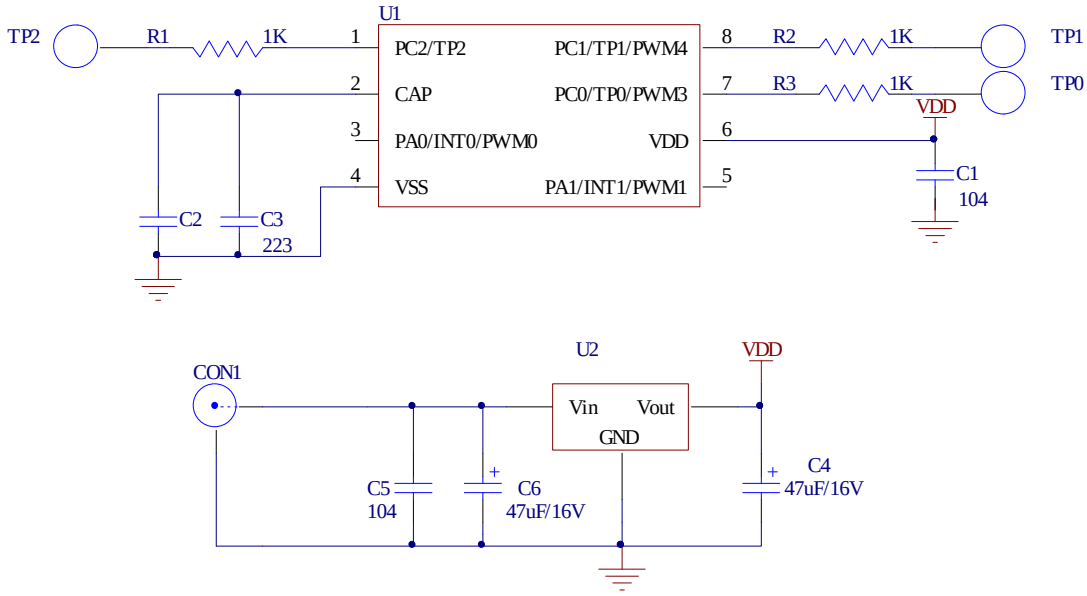
Discharge/Charge: Discharge(charge) can hold touch pad in discharge(charge) state, to avoid discharge(charge) time too short.

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.

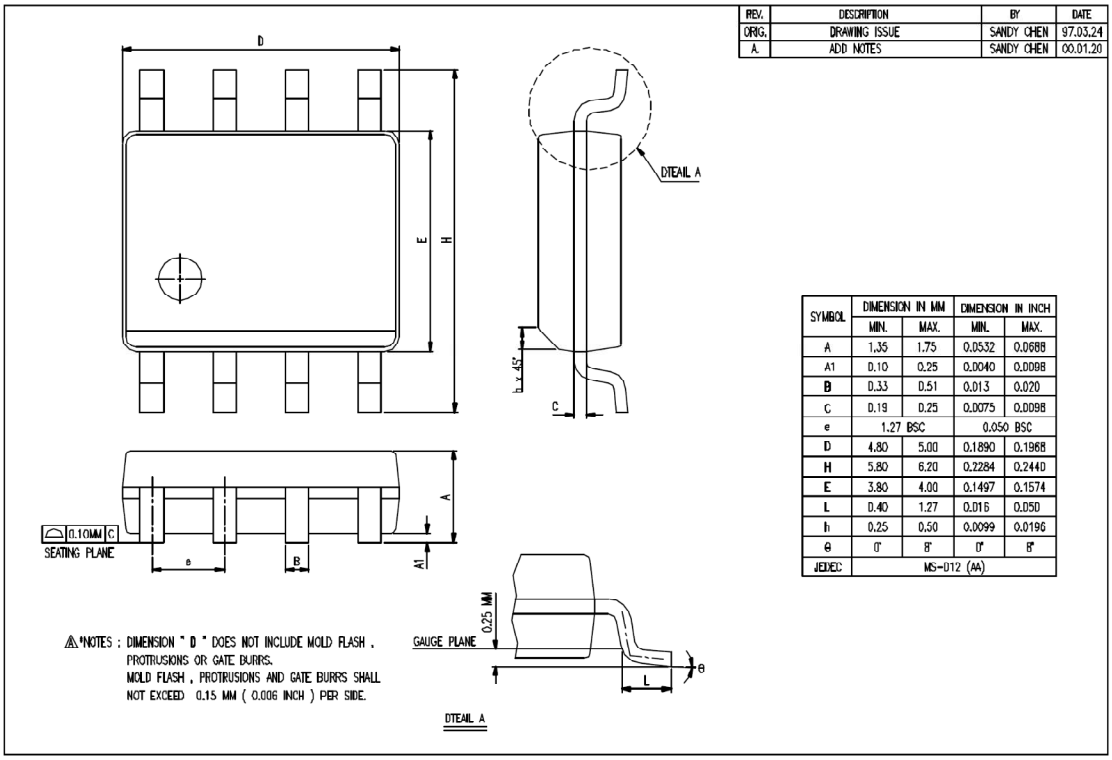
As touch pad analog switch keeps on, the relative IO port is disabled as tri-state by hardware.

§ Application Circuit: Reference only



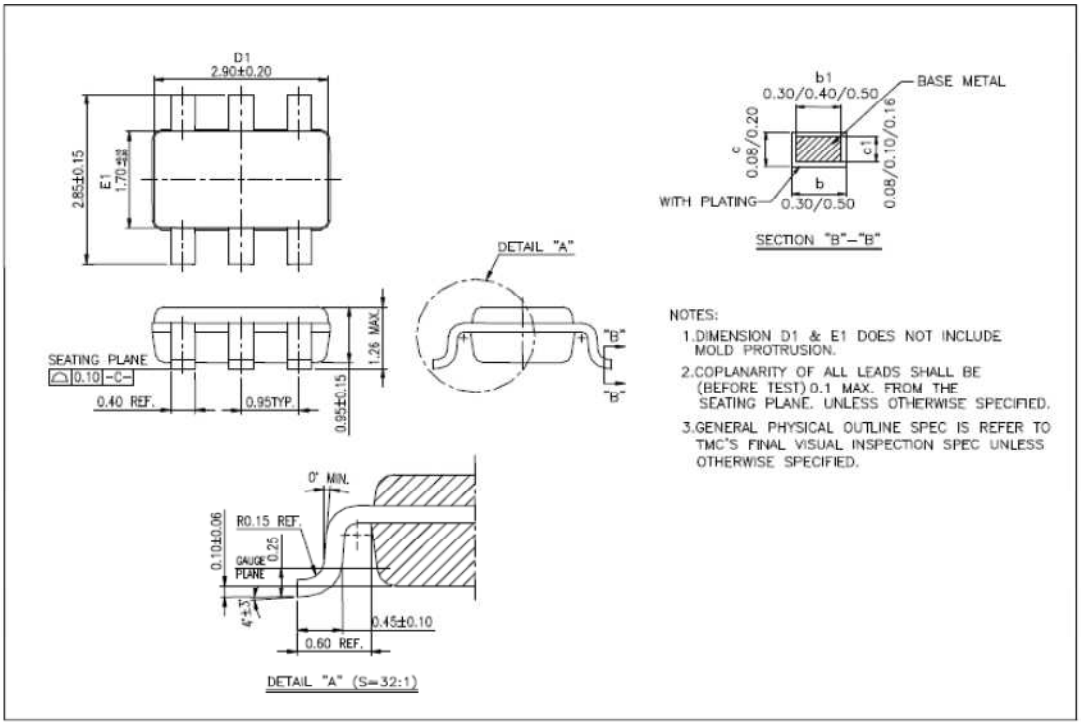
§ Package Information:

- **SOP 8**

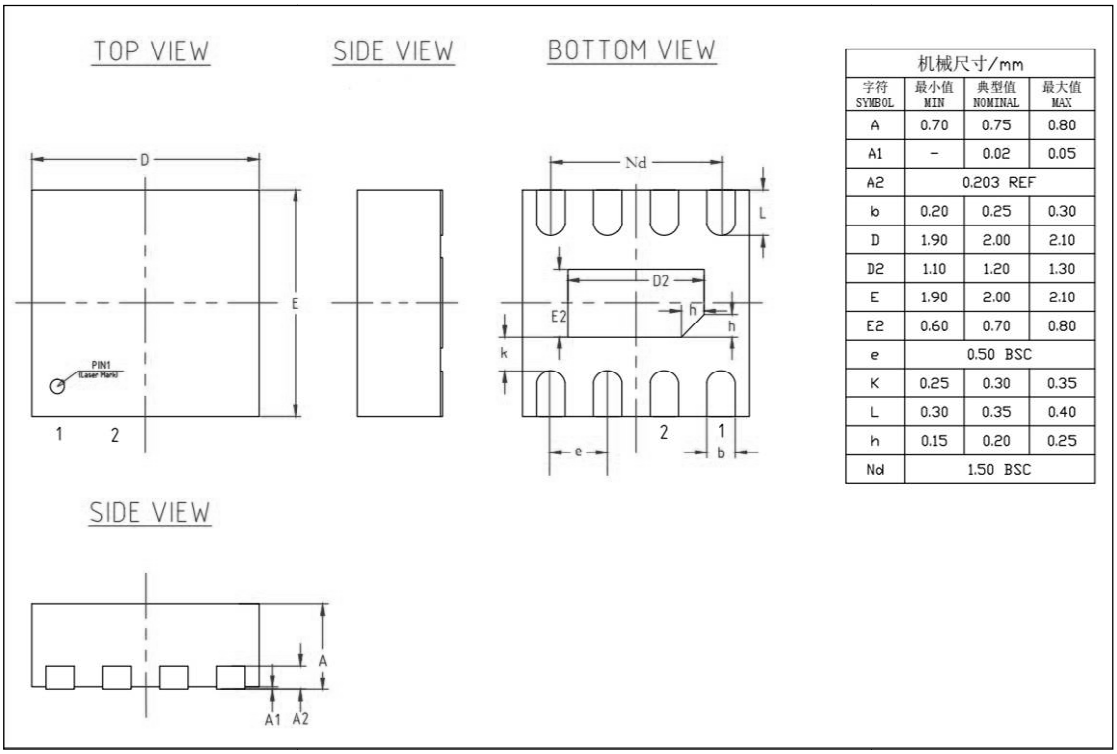


SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN.	MAX.	MIN.	MAX.
A	1.35	1.75	0.0532	0.0688
A1	0.10	0.25	0.0040	0.0098
B	0.33	0.51	0.013	0.020
C	0.19	0.25	0.0075	0.0098
e	1.27 BSC		0.050 BSC	
D	4.80	5.00	0.1890	0.1968
H	5.80	6.20	0.2284	0.2440
E	3.80	4.00	0.1497	0.1574
L	0.40	1.27	0.016	0.050
h	0.25	0.50	0.0099	0.0196
θ	0°	8°	0°	8°
JEDEC	MS-012 (AA)			

- **SOT23-6**



- **DFN8L**



机械尺寸/mm			
字符 SYMBOL	最小值 MIN	典型值 NOMINAL	最大值 MAX
A	0.70	0.75	0.80
A1	–	0.02	0.05
A2	0.203 REF		
b	0.20	0.25	0.30
D	1.90	2.00	2.10
D2	1.10	1.20	1.30
E	1.90	2.00	2.10
E2	0.60	0.70	0.80
e	0.50 BSC		
K	0.25	0.30	0.35
L	0.30	0.35	0.40
h	0.15	0.20	0.25
Nd	1.50 BSC		

§ Ordering Form:

Package Item	Package type
TTP277-AO8N	**SOP8**
TTP277-BO8N	**SOP8**
TTP277-CA6N	**SOT23-6**
TTP277-EB8N	**DFN8L**

REVISION HISTORY：

Body:

2019/05/08：1st version

2019/05/24：Ver. 1.1 fixed typo “SR042” to “TTP277” in Page 1.

2019/08/13：Ver. 1.2 Provide DFN8L package type

2019/08/28：Ver. 1.3 Modify Page1,7　VDD=3.5V~5.5V@LVRmax=3.3V