DATA SHEET

KSL81P304

OTP-BASED 8-BIT USB MICROCONTROLLER

Data Sheet

Version 1.30

Tel: 886-3-5510850 Fax: 886-3-5510860

TABLE OF CONTENTS

Revision History:

Date	Revision	Description	Page
2012/02/07	V001	New Create	32
2012/03/12	V002	Add QFN20 package	33
2012/5/24	V1.00	Error correction Add B version information and function figures	35
2012/6/1	V1.10	Error correction Add OTP_BUSY in status register Add typical connection diagram	36
2012/9/3	V1.20	Error correction Add GPIO circuit diagram Add Timer control mode diagram Add interrupt source diagram	45
2013/2/18	V1.3	Add UART circuit diagram Add USB control endpoint implementation	22~29 43~46

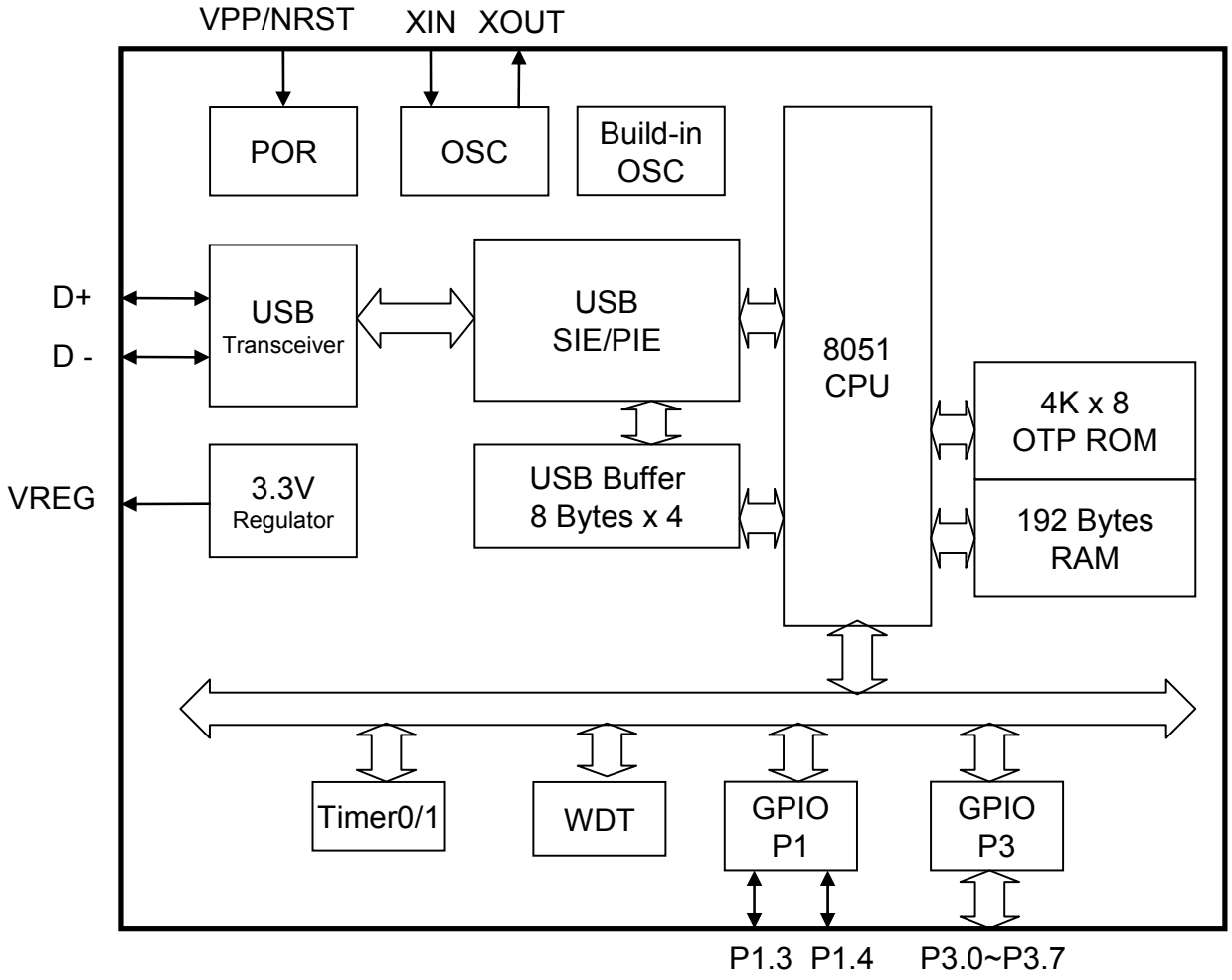
KSL81P304

OTP-Based 8 bit USB Microcontroller

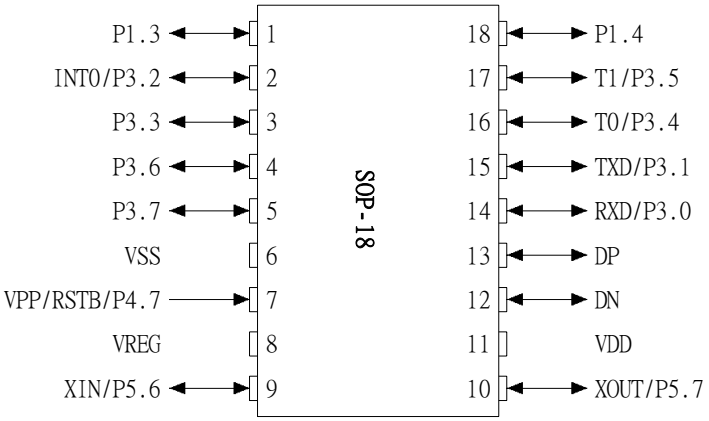
1. Feature

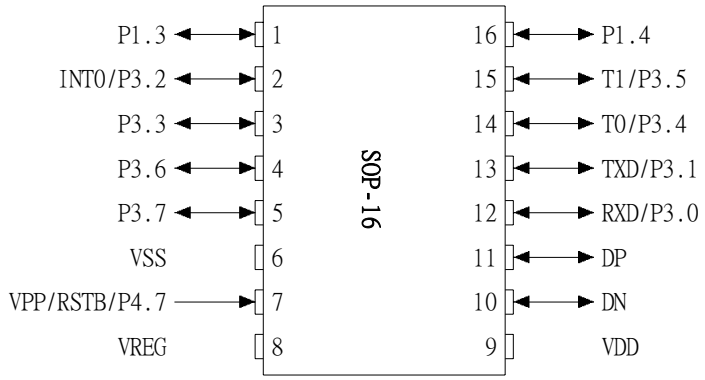
- CPU
  - 8-bit microcontroller compatible with standard 8051.
  - 1T machine cycle runs up to 6MHz.
- Memory
  - 192 bytes internal RAM.
  - 4K x 8 OTP ROM (4064 bytes code area, 32 bytes configuration ROM)
- USB Specification
  - Complies with low speed USB 1.1 Specification.
  - Support three 8-byte endpoints, EP0=Control IN/OUT, EP1/EP2 = Interrupt IN.
  - Support USB Suspend and Resume function.
  - Build-in 1.5K pull-high resistor.
- Reset
  - Power-on Reset
  - External Reset
  - Watchdog Reset
- Power down and wake up function, wake-up trigger by USB or GPIO P3.
- Up to 13 I/O ports.
- Two 16-bit Timers.
- One UART port.
- Build-in 3.3V Regulator output, supply 60mA current.
- Build-in RC oscillator 6MHz (BR6M), <±1.5% after calibration.
- Programmable Watchdog timer.
- 5V power supply only.

2. Block Diagram

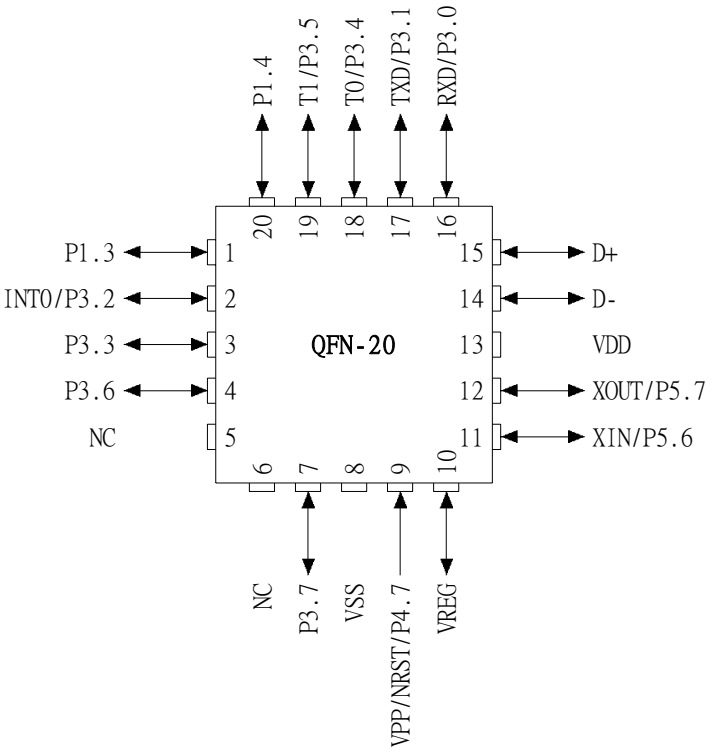


3. Pin Definitions





Note: SOP-16 not support A version



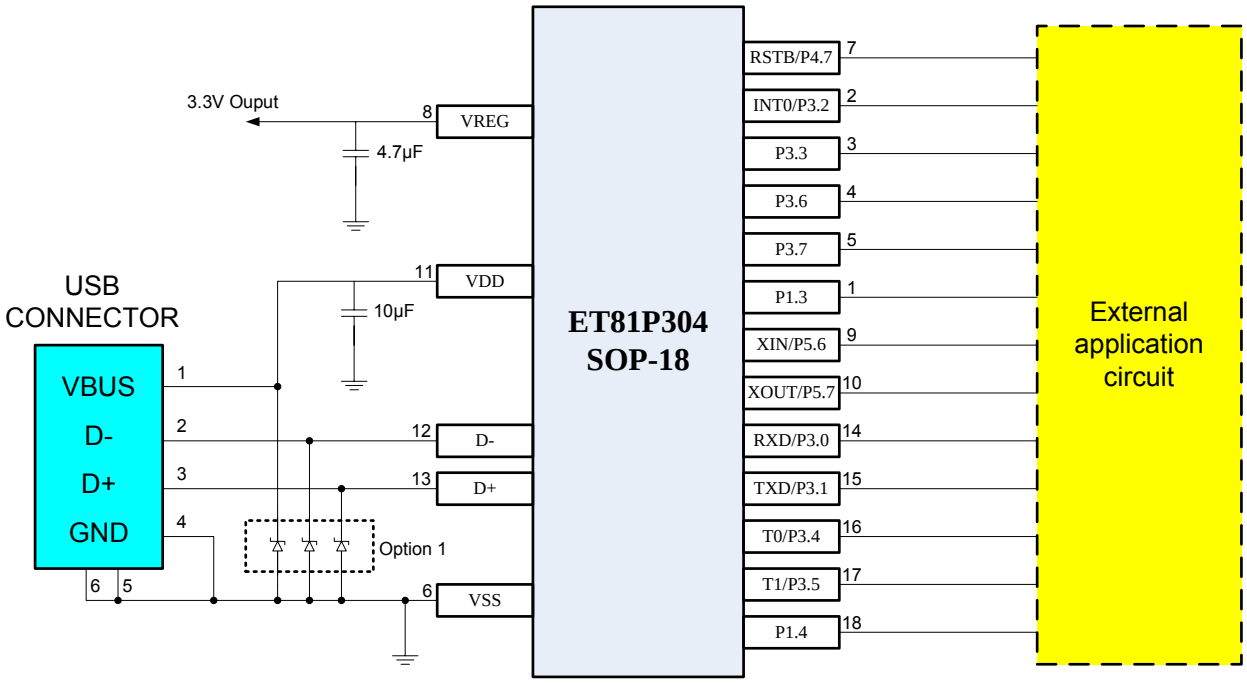
4. Pin Description

Pin Name	Type	16-pin	18-pin	20-pin	Description
P1.3 P1.4	I/O	1,16	1,18	1,20	GPIO Port 1.3, Port 1.4
P3[0:7]	I/O	2~5, 12~15	2~5, 14~17	2~4, 7, 9, 11, 12, 16~19	GPIO Port 3.0 ~ 3.7 with wake-up function pins. RXD(P3.0): Serial Port RXD. TXD(P3.1): Serial Port TXD. INT0(P3.2): External interrupt 0, active low. T0(P3.4): Timer 0 External Input T1(P3.5): Timer 1 External Input
VPP/RSTB/P4.7	I	7	7	9	VPP: OTP ROM Programming voltage supply RSTB: External reset pin, active low. GPIO Port 4.7, input only and pull-up resister always exist.
XIN/P5.6	I/O	N/A	9	11	External crystal input GPIO Port 5.6
XOUT/P5.7	I/O	N/A	10	12	External crystal output GPIO Port 5.7
VSS	G	6	6	8	Ground
VDD	P	9	11	13	Power Supply 5V, need connecting a 10μF bypass capacitor for USB VBUS connection.
VREG	P	8	8	10	3.3V Regulator output, need connecting at least 4.7μF bypass capacitor.
DP	I/O	11	13	15	USB D+
DN	I/O	10	12	14	USB D-

5. Electrical Characteristics

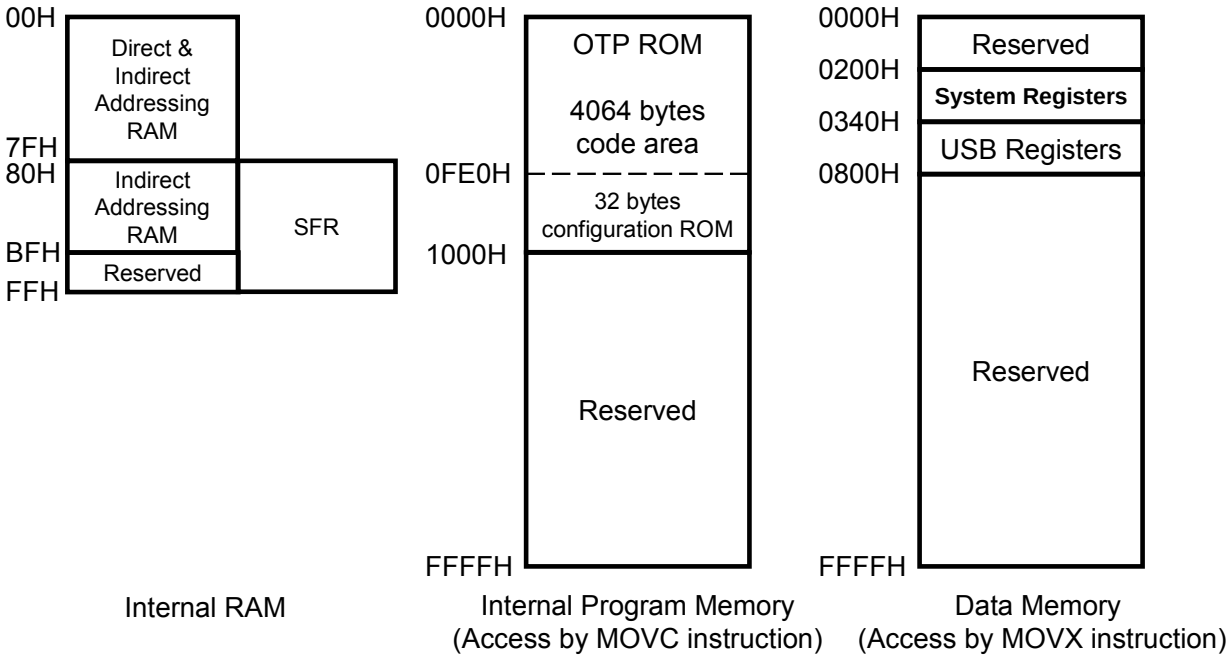
Parameter	Conditions	Min	Typical	Max	Unit
Operation Voltage		4.5	5	5.5	V
Operation Current	VDD5V, BR6M No GPIO Loading		3.2		mA
Suspend Current	Stop system clock, Enable USB pull-up 1.5K Resister		270		uA
VREG Regulator output		3	3.3	3.6	V
VREG Supply current			60		mA
VIH		2.0			V
VIL				0.8	V
VOH	IO = -2mA	2.4			V
VOL	IO = -2mA			0.4	V
Programming Voltage VPP		6.25	6.5	6.75	V

6. Typical Connection Diagram



Option 1: Avalanche transient voltage suppression diode should be added for ESD protection.

7. Memory Map



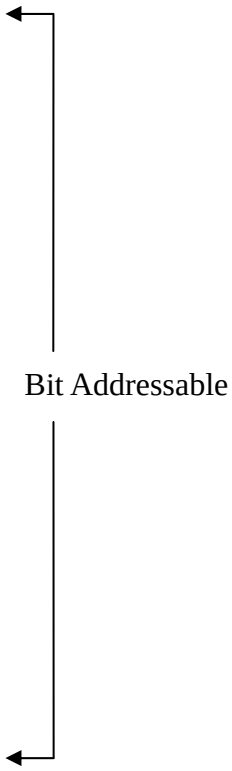
Internal RAM

Internal Program Memory
(Access by MOVC instruction)

Data Memory
(Access by MOVX instruction)

8. Internal RAM Addressing

BFh - 80h	**Indirect RAM**							
7Fh - 30h	**Direct RAM**							
2Fh	**7F**	**7E**	**7D**	**7C**	**7B**	**7A**	**79**	**78**
2Eh	**77**	**76**	**75**	**74**	**73**	**72**	**71**	**70**
2Dh	**6F**	**6E**	**6D**	**6C**	**6B**	**6A**	**69**	**68**
2Ch	**67**	**66**	**65**	**64**	**63**	**62**	**61**	**60**
2Bh	**5F**	**5E**	**5D**	**5C**	**5B**	**5A**	**59**	**58**
2Ah	**57**	**56**	**55**	**54**	**53**	**52**	**51**	**50**
29h	**4F**	**4E**	**4D**	**4C**	**4B**	**4A**	**49**	**48**
28h	**47**	**46**	**45**	**44**	**43**	**42**	**41**	**40**
27h	**3F**	**3E**	**3D**	**3C**	**3B**	**3A**	**39**	**38**
26h	**37**	**36**	**35**	**34**	**33**	**32**	**31**	**30**
25h	**2F**	**2E**	**2D**	**2C**	**2B**	**2A**	**29**	**28**
24h	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**
23h	**1F**	**1E**	**1D**	**1C**	**1B**	**1A**	**19**	**18**
22h	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**
21h	**0F**	**0E**	**0D**	**0C**	**0B**	**0A**	**09**	**08**
20h	**07**	**06**	**05**	**04**	**03**	**02**	**01**	**00**
1Fh - 18h	**Bank 3**							
17h - 10h	**Bank 2**							
0Fh - 08h	**Bank 1**							
07h - 00h	**Bank 0**							



9. Special Function Registers

The ET81P304 uses Special Functions Registers (SFRs) to control and monitor peripherals and their modes.

The SFRs reside in the register locations 80-FFh and are accessed by direct addressing only. Some of the SFRs are bit addressable. This is very useful in cases where one wishes to modify a particular bit without changing the others. The SFRs that are bit addressable are those whose addresses end in 0 or 8. The ET81P304 contains all the SFRs present in the standard 8051. However, some additional SFRs have been added. In some cases unused bit in the original 8051 have been given new functions. The list of SFRs is as follows. The table is condensed with eight locations per row. Empty locations indicate that there are no registers at these addresses.

Table 9-1: Special Function Registers summary

	0	1	2	3	4	5	6	7	
F8									FF
F0	B								F7
E8									EF
E0	ACC								E7
D8	ADCON								DF
D0	PSW								D7
C8									CF
C0									C7
B8	IP		S0RELH						BF
B0	P3	P3DIR	P4	P4DIR	P5	P5DIR			B7
A8	IE		S0RELL						AF
A0	P2	P2DIR	P0DIR						A7
98	SCON	SBUF							9F
90	P1	P1DIR	DPS						97
88	TCON	TMOD	TL0	TL1	TH0	TH1			8F
80	P0	SP	DPL	DPH	DPL1	DPH1		PCON	87

Note: The SFRs in the column with dark borders are bit-addressable.

9.1. GPIO Ports

The I/O registers (P1, P3, P4, P5), I/O control registers (P1DIR, P3DIR, P5DIR) and pull-up control registers (system register 21Ah, 21Ch, 21Eh) are used to manipulate I/O pins. All I/O pins integrates a weak pull-up function and can be set input and output, except RSTB/P4.7 is input only and pull-up resister always exist while can not be control. In output mode, the pull-up will be turned off automatically.

9.1.1. Port 1

P1 register is including P1.3 and P1.4 two general purpose I/O ports with a weak pull-up function (system register 21Ah), P1DIR is I/O control register. The P1 pins circuit diagram is shown in Figure 9-1.

9.1.2. Port 3

P3 is an 8-bit I/O register. Each P3 pin integrates a weak pull-up function (system register 21Ch). When CPU in to power down mode, each P3 pin also have a wake-up function (system register 20Bh.0 = 1) with input mode. The P3 pins circuit diagram is shown in Figure 9-1.

9.1.3. Port 4

P4 register only has one pin is P4.7 that is input only and pull-up resister always exist while can not be control. The RSTB/P4.7 pin also can be reset pin, which function is depends on configuration ROM 0xFE2.5 setting (load into system register 205h.5 when power on). The RSTB/P4.7 pin circuit diagram is shown in Figure 9-2.

9.1.4. Port 5

P5 register is including P5.6 and P5.7 two general purpose I/O ports with a weak pull-up function (system register 21Eh), P5DIR is I/O control register. The XIN/P5.6 and XOUT/5.7 pins also can be configuring external crystal pins, which function is depends on configuration ROM 0xFE2.6 setting (load into system register 205h.6 when power on). The XIN/P5.6 and XOUT/5.7 pins circuit diagram is shown in Figure 9-3.

Table 9-2: GPIO Control Registers Summary

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
80h	P0	P0 Data Reg.	N/A								0000 0000
			N/A								
A2h	P0DIR	P0 Direction Reg.	N/A								0000 0000
			N/A								
90h	P1	P1 Data Reg.	N/A			P1[4:3]		N/A			0000 0000
			N/A			r/w		N/A			
91h	P1DIR	P1 Direction Reg.	N/A			P1DIR[4:3]		N/A			0000 0000
			N/A			r/w		N/A			
A0h	P2	P2 Data Reg.	N/A								0000 0000
			N/A								
A1h	P2DIR	P2 Direction Reg.	N/A								0000 0000
			N/A								
B0h	P3	P3 Data Reg.	P3[7:0]								0000 0000
			r/w								
B1h	P3DIR	P3 Direction Reg.	P3DIR[7:0]								0000 0000
			r/w								
B2h	P4	P4 Data Reg.	P4.7	N/A							0000 0000
			r	N/A							
B3h	P4DIR	P4 Direction Reg.	0	N/A							0000 0000
			r/w	N/A							
B4h	P5	P5 Data Reg.	P5[6:7]		N/A						0000 0000
			r/w		N/A						
B5h	P5DIR	P5 Direction Reg.	P5DIR[6:7]		N/A						0000 0000
			r/w		N/A						

P0~5[7:0]: port 0 ~ 5 data

P0~5DIR[7:0]: select the in/out of port 0 ~ 5

1 = Drives the IO pin on output state

0 = Drives the IO pin on input high-impedance state

Note: P4DIR.7 always set 0 due to P4.7 is input high-impedance state only.

Figure 9-1: Circuit Diagram of P1, P3 Pins

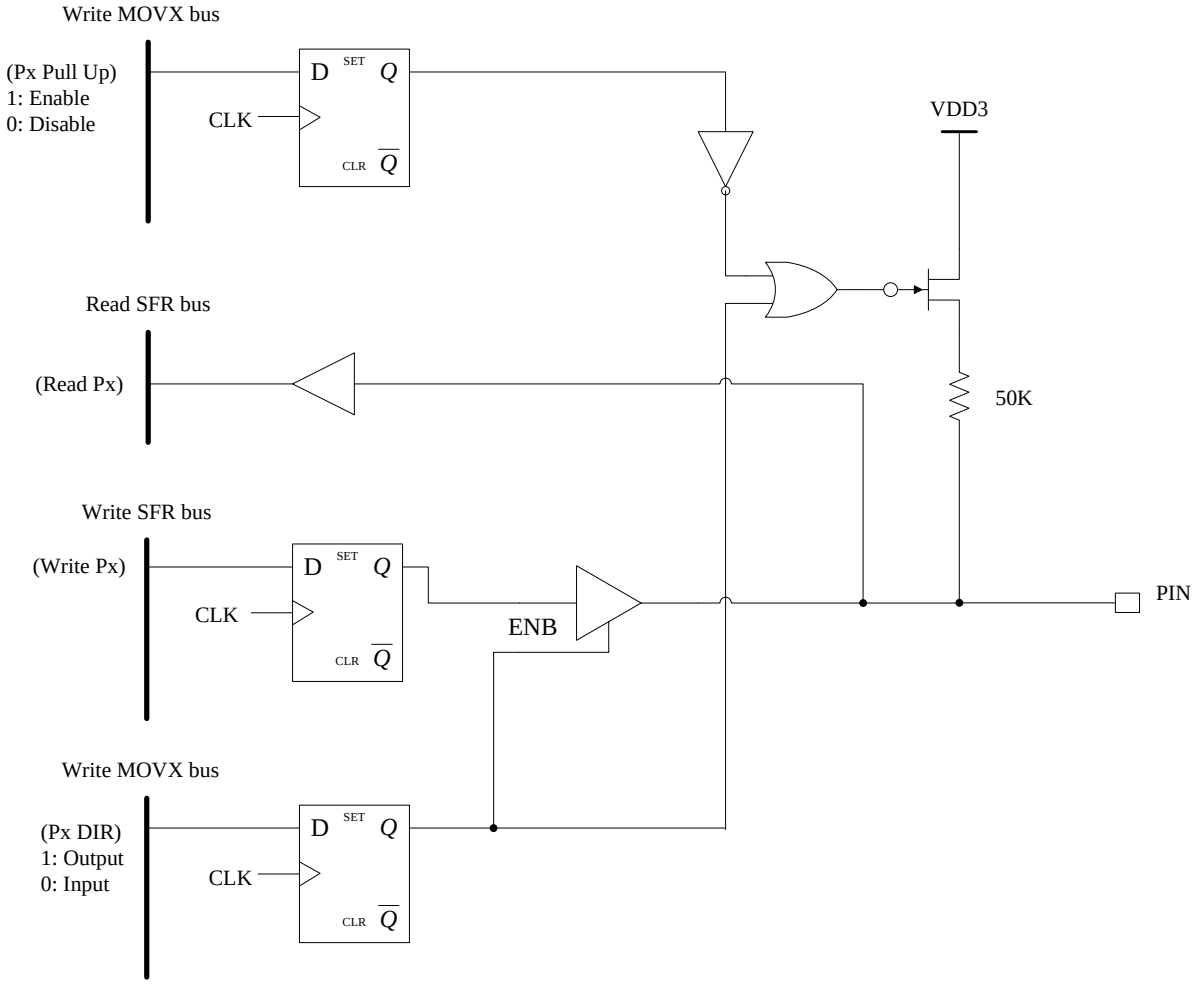


Figure 9-2: Circuit Diagram of RSTB/P4.7 Pins

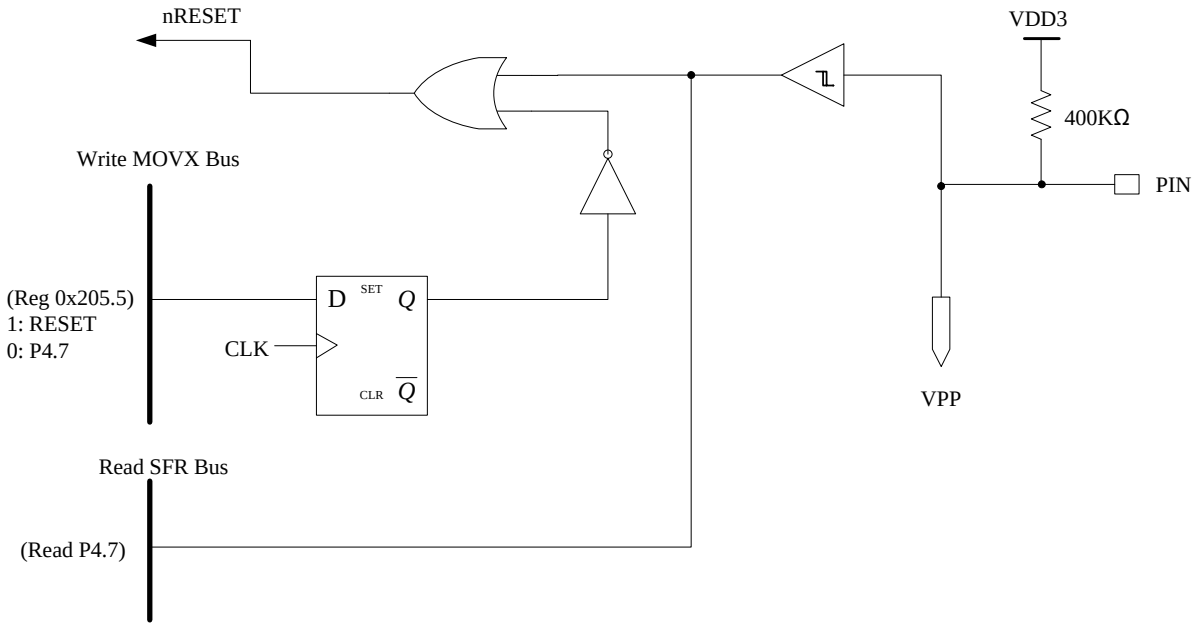
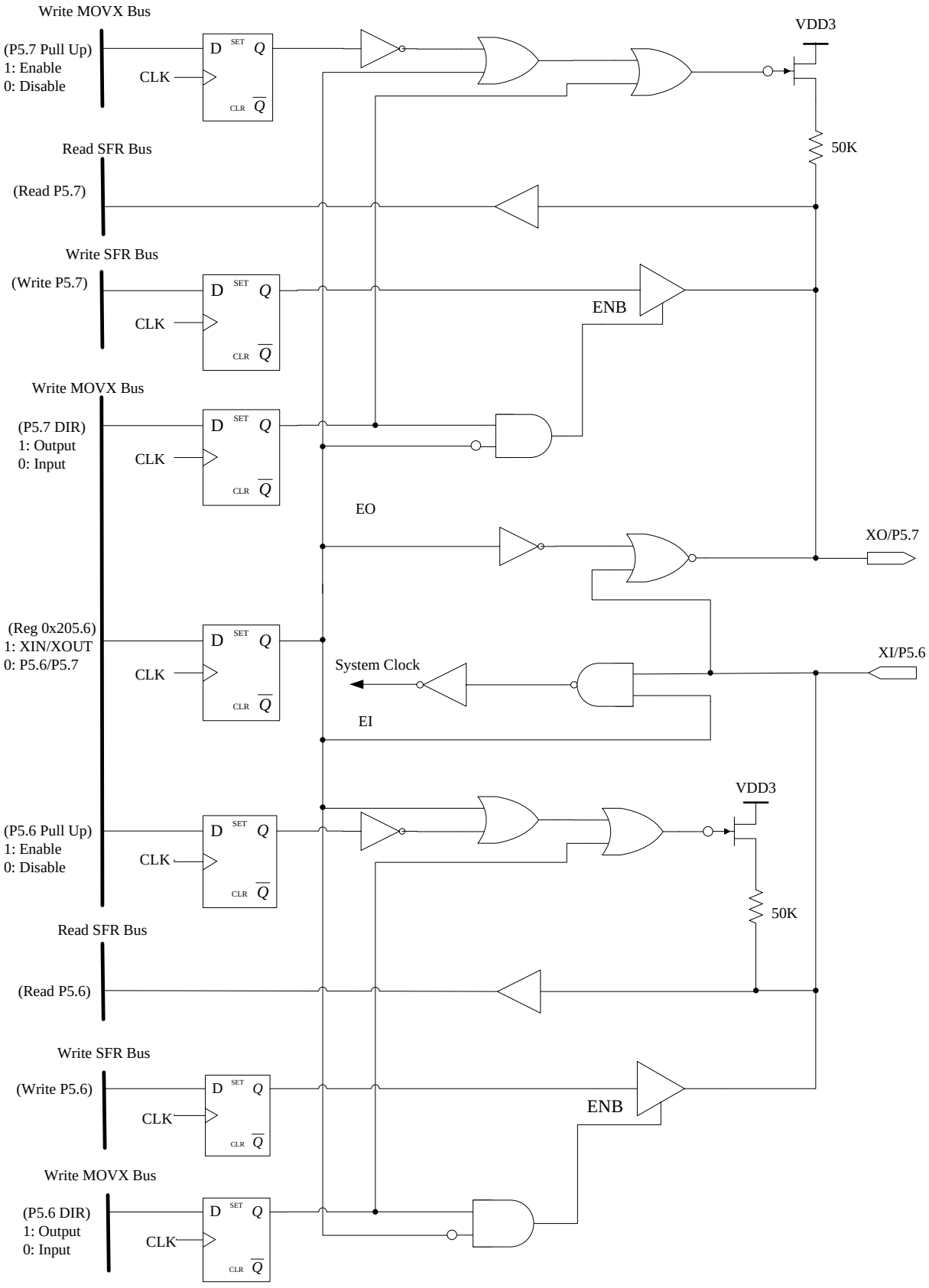


Figure 9-3: Circuit Diagram of XIN/P5.6 XOUT/P5.7 Pins



9.2. Interrupts

The ET81P304 has a two priority level interrupt structure with 5 interrupt sources. Each of the interrupt sources has an individual priority bit, flag, interrupt vector and enable bit. In addition, the interrupts can be globally enabled or disabled.

Interrupt Sources

The ET81P304 has 5 sources of interrupt: INT0 pin, USB, Timer 0, Timer 1, UART 0. When the EA bit is set and an interrupt condition occurs, its corresponding flag will be set to '1', and then to go into Interrupt Service Routine (ISR). After it enters ISR, the hardware will clear its corresponding flag except UART 0 interrupt flag TI and RI, these two UART 0 interrupt flag is cleared by software. The vector of interrupt is shown in Table 9-4.

Priority Level Structure

The 5 sources of interrupt priority are shown in Table 9-4. The ET81P304 has 2 priority levels, each interrupt source can be individually set to either high or low levels priority. Naturally, a higher priority interrupt cannot be interrupted by a lower priority interrupt, but if lower priority interrupt was sets in high level priority and higher priority interrupt was sets in low level priority, that higher priority interrupt can be interrupted by a lower priority interrupt. The interrupt source diagram is shown in Figure 9-4.

Table 9-3: Interrupt Control Registers Summary

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
88h	TCON	Timer Control Reg.	**TF1**	**TR1**	**TF0**	**TR0**	**IE1**	**IT1**	**IE0**	**IT0**	0000 0000
			Timer 0/1 r/w				**INT0/1 r/w**				
A8h	IE	Interrupt Enable Reg.	**EA**	--	--	**ES**	**ET1**	**EX1**	**ET0**	**EX0**	0000 0000
			r/w	--	--	r/w	r/w	r/w	r/w	r/w	
B8h	IP	Interrupt Priority Reg.	--	--	--	**PS**	**PT1**	**PX1**	**PT0**	**PX0**	0000 0000
			r/w	--	--	r/w	r/w	r/w	r/w	r/w	

TCON register bit function description:

TF0/TF1: Timer 0/1 overflows flag, cleared by hardware.

TR0/TR1: Timer 0/1 enables.

IE0/IE1: External interrupt 0/1 edge detects.
Set by hardware when an edge/level is detected on INT0/USB interrupt . This bit is cleared by hardware when the service routine is vectored to only if the interrupt was edge triggered. Otherwise it follows the pin.

IT0/IT1: External interrupt 0/1 type control, 0: low level trigger, 1: falling edge trigger.

IE register bit n=1：Enable interrupt; bit n=0：Disable interrupt

EA: Enable all interrupts

ES: Enable UART0 interrupt

ET1: Enable Timer1 overflow interrupt

EX1: Enable External interrupt 1 (USB interrupt)

ET0: Enable Timer0 overflow interrupt

EX0: Enable External interrupt 0

IP register bit n=1：Higher priority level; bit n=0：Lower priority level

PS: Define UART0 interrupt priority level

PT1: Define Timer1 overflow interrupt priority level

PX1: Define USB interrupt priority level

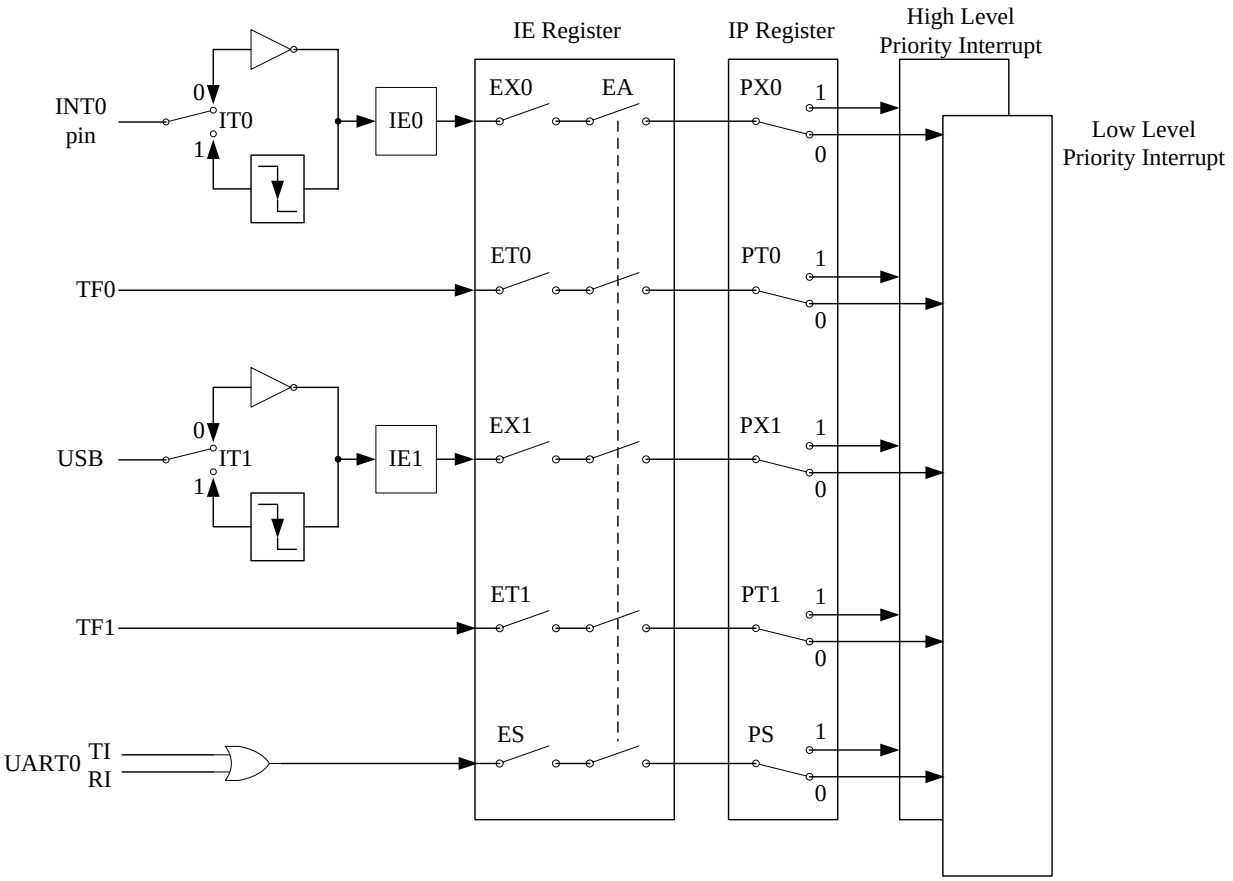
PT0: Define Timer0 overflow interrupt priority level

PX0: Define External interrupt 0 priority level

Table 9-4: Interrupt Vectors and Priority

INTERRUPT SOURCE	VECTOR ADDRESS	PRIORITY
External Interrupt 0	03h	1 (highest)
Timer 0	0Bh	2
USB	13h	3
Timer 1	1Bh	4
UART0	23h	5 (lowest)

Figure 9-4: Interrupt sources diagram



9.3. UART 0

The UART port in the ET81P304 is a full duplex port which is capable of asynchronous communication. In the asynchronous mode, full duplex operation is available. This means that it can simultaneously transmit and receive data. The transmit register and the receive buffer are both addressed as SBUF register. However any write to SBUF will be to the transmit register, while a read from SBUF will be from the receive buffer register. The UART port can operate in four different modes and relation control register as described below.

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
AAh	S0RELL	UART 0 Baud rate generator, low byte	S0REL[7:0]								0000 0000
			r/w								
BAh	S0RELH	UART 0 Baud rate generator, high byte	--	--	--	--	--	--	S0REL[9:8]		0000 0000
			--	--	--	--	--	--	r/w		
98h	SCON	UART 0 Control Reg.	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	0000 0000
			r/w								
99h	SBUF	UART 0 Data Buffer Reg.	SBUF[7:0]								0000 0000
			r/w								
D8h	ADCON	UART 0 Config Reg	BS	--	--	--	--	--	--	UPS	0000 0000
			r/w							r/w	

UPS (ADCON.0): UART port selection, always set to 1 for UART 0 to P3.0/P3.1
S0REL [9:0]: UART 0 Baud rate Generator.
BS (ADCON.7): UART0 Baud rate generator timer selection, 0: TH1, 1:S0REL
SM0, SM1 (SCON [7:6]): Serial Port Mode

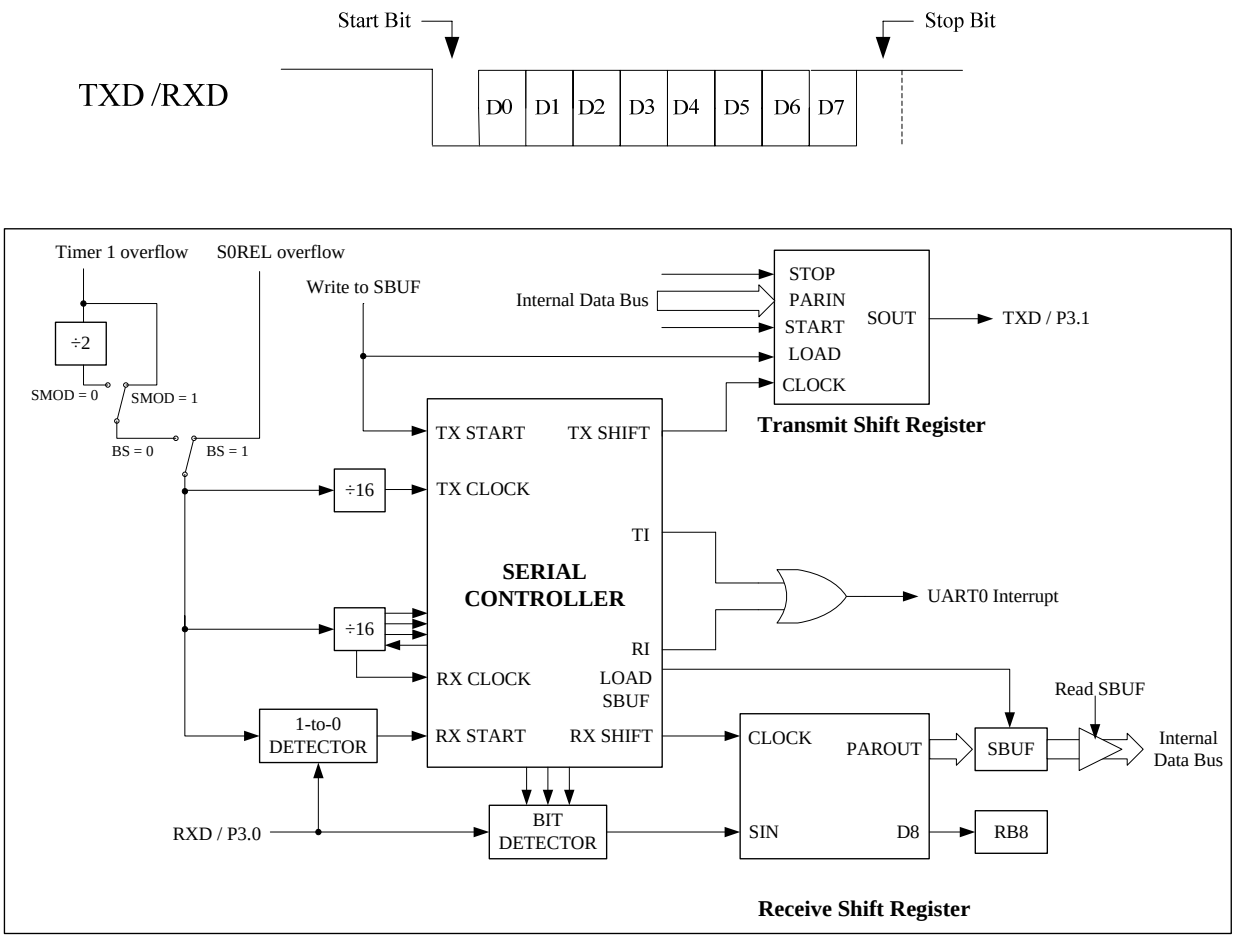
SM0	SM1	Mode	Description	Type	Start Bit	Stop Bit	Baud rate
0	0	N/A	--	--	--	--	--
0	1	1	8 bits UART	Asynch.	1	1	Programmable
1	0	2	9 bits UART	Asynch.	1	1	Fsys / 64 or Fsys / 32
1	1	3	9 bits UART	Asynch.	1	1	Programmable

SM2 (SCON.5): Multiple processors communication.
REN (SCON.4): Receive enable.
TB8 (SCON.3): This is the transmitted 9th bit data in mode 2 and 3.
RB8 (SCON.2): This is the received 9th data bit in modes 2 and 3.
TI (SCON.1): Transmit interrupt flag, cleared by software.
RI (SCON.0): Receive interrupt flag, cleared by software.

9.3.1. Mode 1: 8 bits UART

In mode 1, the full duplex asynchronous mode is used, serial communication frames are made up of 10 bits transmitted on TXD and received on RXD. The 10 bits consist of a start bit (0), 8 data bits (LSB first), stop bit (1). The function diagram and timing is shown as below.

Figure 9-6: Mode 1 timing and block diagram



Baud rate equation:

When BS bit (ADCON.7) = 0, UART0 Baud rate generator timer by TH1.

$$BaudRate = \frac{2^{s\,\mathrm{mod}}\, xFclk}{32x12x(256 - TH1)}$$ (SMOD is register PCON.7)

$$TH1 = 256 - \frac{2^{s\,\mathrm{mod}}\, xFclk}{32x12xBaudRate}$$

When BS bit (ADCON.7) = 1, UART0 Baud rate generator timer by S0REL.

$$BaudRate = \frac{2^{s\,\mathrm{mod}}\, xFclk}{32x(2^{10} - S0REL)}$$ (SMOD is register PCON.7)

$$S0REL = 1024 - \frac{2^{s\,\mathrm{mod}}\, xFclk}{32xBaudRate}$$

Mode 1 setting example:

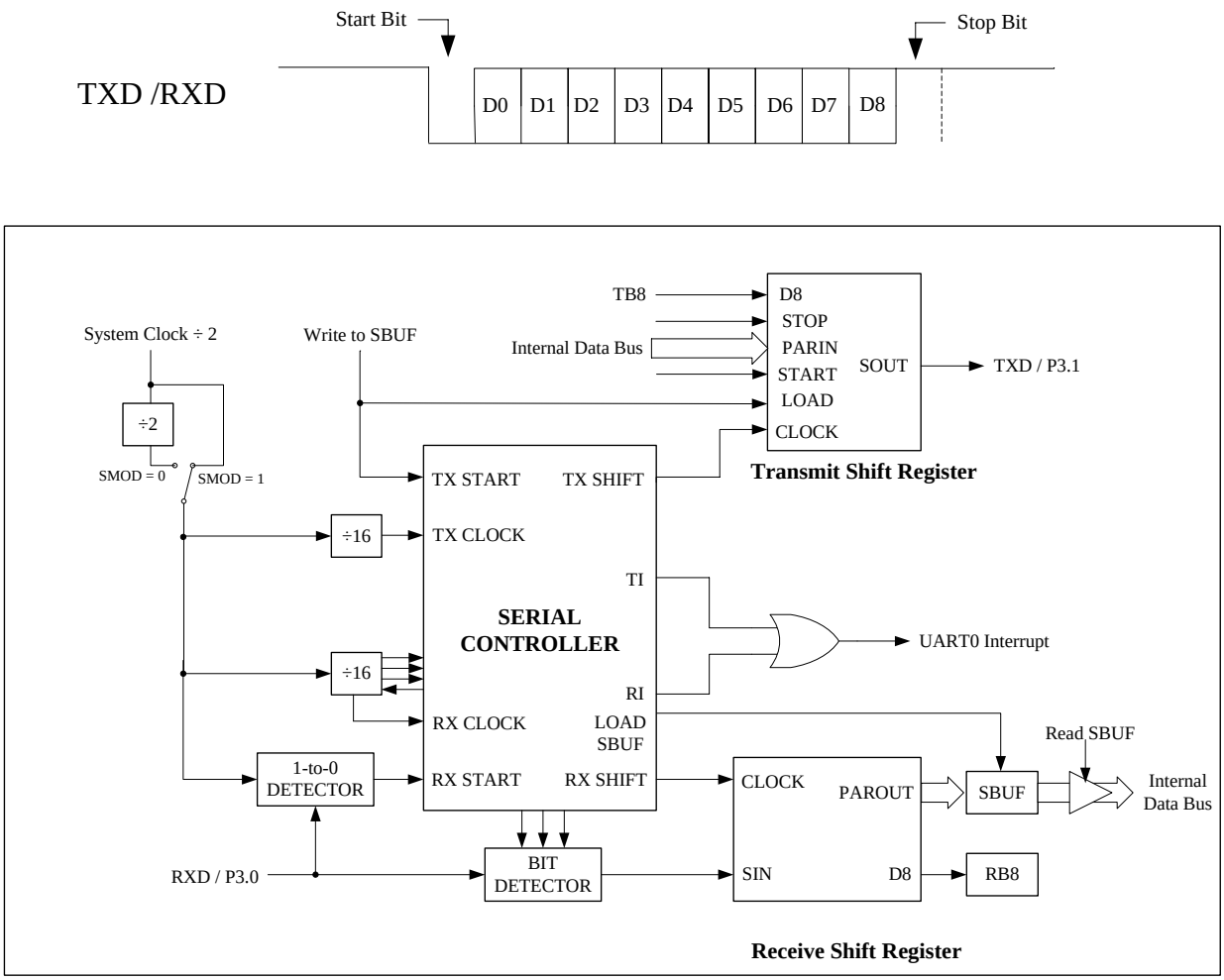
```
MOV     DPTR,#20BH
MOVX    A,@DPTR
SETB    A.7                              ; GPIO 3.0 / 3.1 = RXD / TXD
MOVX    @DPTR,A

MOV     SCON,#50H                        ; mode 1, 8-bit UART, enable RX
MOV     PCON,#80H                        ; SMOD = 1
MOV     S0RELL,#BAUDRATE_LOW_BYTE
MOV     S0RELH,#BAUDRATE_HIGH_BYTE
MOV     ADCON,#81H                  ; ADCON.7 = 1 for UART0 with S0REL timer
                                         ; ADCON.0 = 1 for UART 0 to P3.0/P3.1,
SETB    ES                               ; enable serial interrupt
```

9.3.2. Mode 2: 9 bits UART

This mode uses a total of 11 bit in asynchronous full-duplex communication. The function description is shown in the figure below. The frame consists of one start bit (0), 8 data bits (LSB first), a programmable 9th bit (TB8) and a stop bit (1). The 9th bit received is put into RB8. The baud rate is programmable to 1/32 or 1/64 of the system clock, which is determined by the SMOD bit in PCON SFR. The function diagram and timing is shown as below.

SMOD = 1, Baud rate = 1/32 Fsys
SMOD = 0, Baud rate = 1/64 Fsys

Figure 9-7: Mode 2 timing and block diagram



Mode 2 baud rate = 1/32 system clock setting example:

```
MOV     DPTR,#20BH
MOVX    A,@DPTR
SETB    A.7                          ; GPIO 3.0 / 3.1 = RXD / TXD
MOVX    @DPTR,A

MOV     SCON,#90H                    ; mode 2, 9-bit UART, enable RX
MOV     PCON,#80H                    ; SMOD = 1 for baud rate = 1/32 system clock
MOV     ADCON,#01H                   ; ADCON.0 = 1 for UART 0 to P3.0/P3.1,
SETB    ES                           ; enable serial interrupt
```

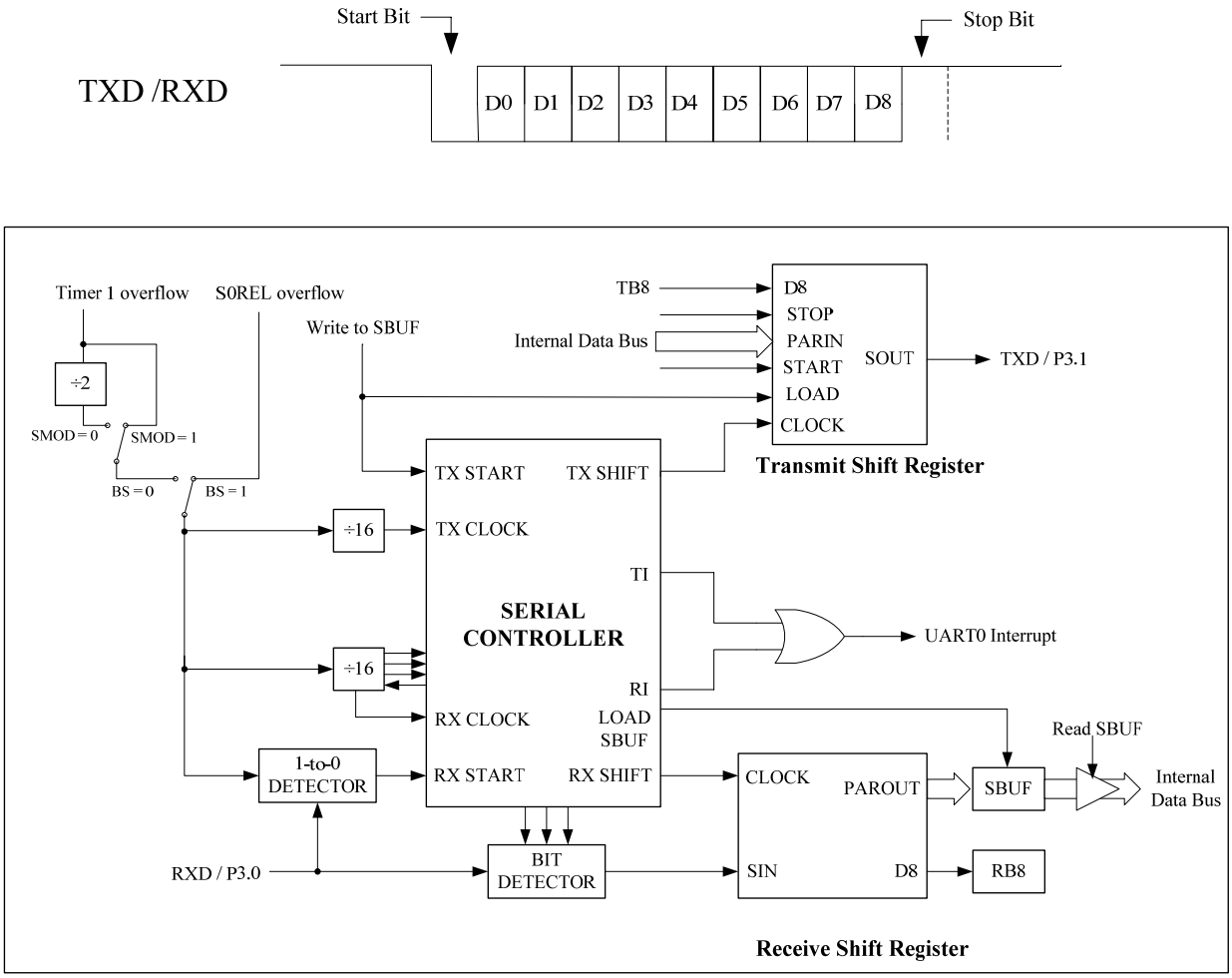
Mode 2 baud rate = 1/64 system clock setting example:

```
MOV     DPTR,#20BH
MOVX    A,@DPTR
SETB    A.7                          ; GPIO 3.0 / 3.1 = RXD / TXD
MOVX    @DPTR,A

MOV     SCON,#90H                    ; mode 2, 9-bit UART, enable RX
MOV     PCON,#00H                    ; SMOD = 0 for baud rate = 1/64 system clock
MOV     ADCON,#01H                   ; ADCON.0 = 1 for UART 0 to P3.0/P3.1,
SETB    ES                           ; enable serial interrupt
```

9.3.3. Mode 3: 9 bits UART (Programmable)

This mode is similar to Mode 2 in all respects, except that the baud rate is programmable like Mode 1. The function diagram and timing is shown as below.

Figure 9-8: Mode 2 timing and block diagram



Baud rate equation:

When BS bit (ADCON.7) = 0, UART0 Baud rate generator timer by TH1.

$$BaudRate = \frac{2^{s\,\mathrm{mod}}\, xFclk}{32x12x(256 - TH1)}$$ (SMOD is register PCON.7)

$$TH1 = 256 - \frac{2^{s\,\mathrm{mod}}\, xFclk}{32x12xBaudRate}$$

When BS bit (ADCON.7) = 1, UART0 Baud rate generator timer by S0REL.

$$BaudRate = \frac{2^{s\,\mathrm{mod}}\, xFclk}{32x(2^{10} - S0REL)}$$ (SMOD is register PCON.7)

$$S0REL = 1024 - \frac{2^{s\,\mathrm{mod}}\, xFclk}{32xBaudRate}$$

Mode 3 setting example:

```
MOV     DPTR,#20BH
MOVX    A,@DPTR
SETB    A.7                                 ; GPIO 3.0 / 3.1 = RXD / TXD
MOVX    @DPTR,A

MOV     SCON,#D0H                           ; mode 3, 8-bit UART, enable RX
MOV     PCON,#80H                           ; SMOD = 1
MOV     S0RELL,#BAUDRATE_LOW_BYTE
MOV     S0RELH,#BAUDRATE_HIGH_BYTE
MOV     ADCON,#81H                  ; ADCON.7 = 1 for UART0 with S0REL timer
                                            ; ADCON.0 = 1 for UART 0 to P3.0/P3.1,
SETB    ES                                  ; enable serial interrupt
```

9.4. Timer 0/1

The ET81P304 has two 16-bit timers that are functionally similar to the timers of the 8051 family, which is up count timer and 12 clocks per count.

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
88h	TCON	Timer Control Reg	**TF1**	**TR1**	**TF0**	**TR0**	**IE1**	**IT1**	**IE0**	**IT0**	0000 0000
			Timer 0/1 r/w				INT0/1 r/w				
89h	TMOD	Timer Mode Control Reg	**0**	**C/T**	**M1**	**M0**	**GATE**	**C/T**	**M1**	**M0**	0000 0000
			Timer 1 r/w				Timer 0 r/w				

TF0/TF1: Timer 0/1 overflows flag, cleared by hardware.
TR0/TR1: Timer 0/1 enables.
GATE = 1: Timer 0 is enabled only while INT0 pin is high and TR0 = 1.
GATE = 0: Timer 0 is enabled whenever TR0 = 1.
C/T = 1: Counter mode, clock source from T0/1 pin.
C/T = 0: Timer mode, clock source from system clock.

	M1	M0	
Mode 0	0	0	8-bit Timer/Counter with 5 bit pre-scale. TH0/1 = 8-bit, TL0/1 = 5-bit (bit 4 ~ 0). Control Block Diagram is shown in Figure 9-9 and 9-10.
Mode 1	0	1	16-bit Timer/Counter, TH0/1 = 8-bit (bit 15 ~ 8), TL0/1 = 8-bit (bit 7 ~ 0). Control Block Diagram is shown in Figure 9-11 and 9-12.
Mode 2	1	0	8-bit Timer/Counter with auto-reload from TH0/1. Control Block Diagram is shown in Figure 9-13 and 9-14.
Mode 3 by Timer 0	1	1	TL0 is an 8-bit timer/counter controlled by the Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits. Control Block Diagram is shown in Figure 9-15.
Mode 3 by Timer 1	1	1	Timer 1 is stopped.

Figure 9-9: Timer 0 Mode 0 control block diagram

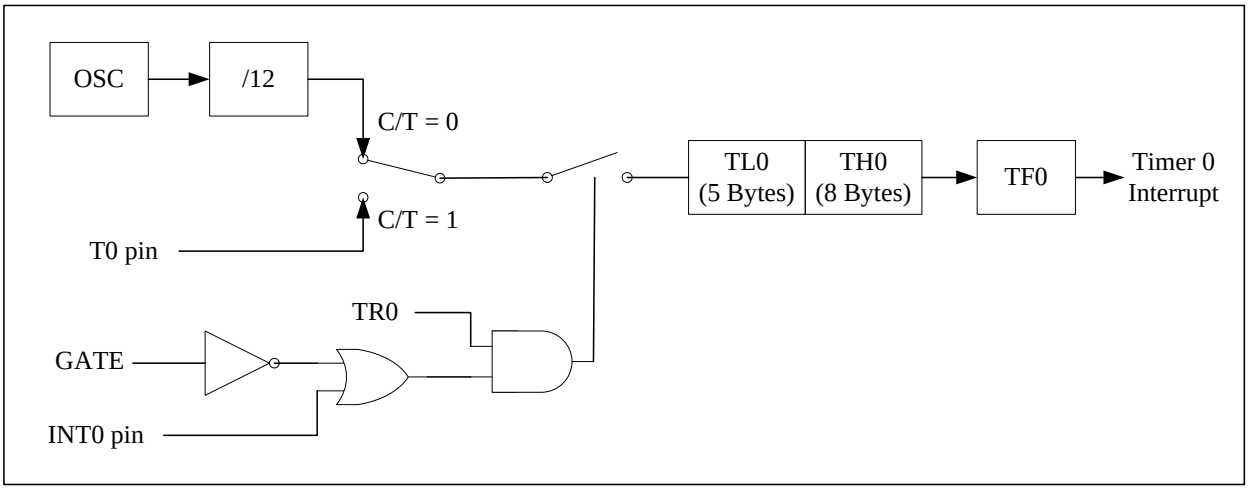


Figure 9-10: Timer 1 Mode 0 control block diagram

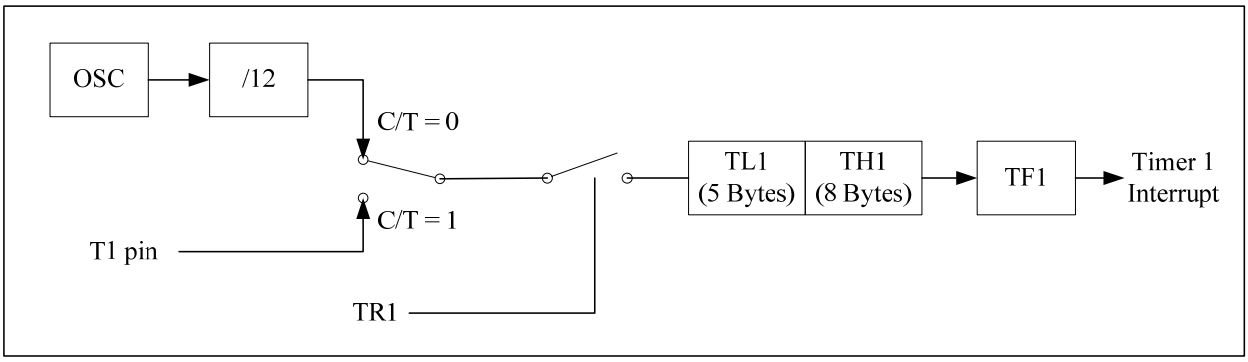


Figure 9-11: Timer 0 Mode 1 control block diagram

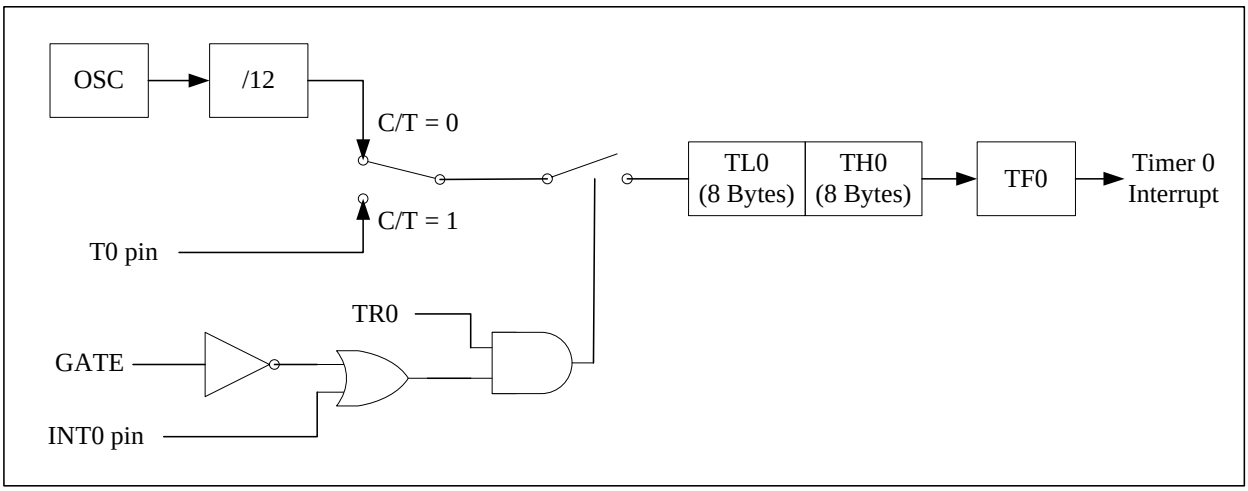


Figure 9-12: Timer 1 Mode 1 control block diagram

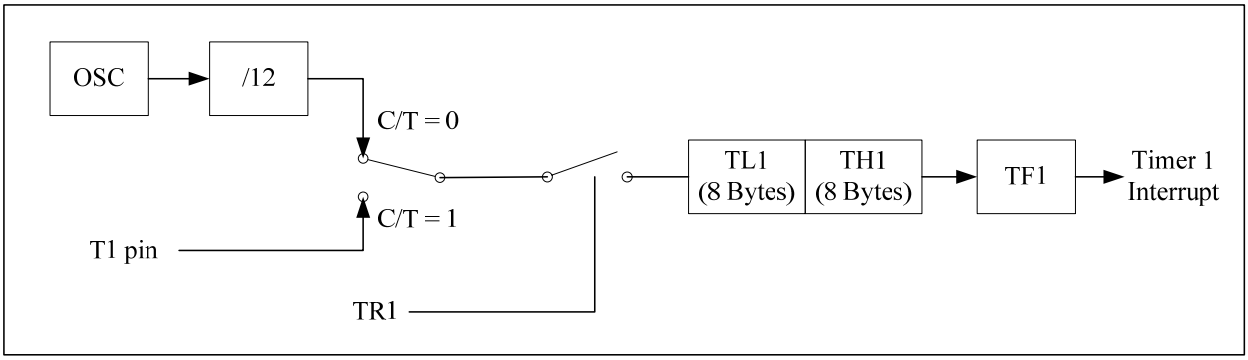


Figure 9-13: Timer 0 Mode 2 control block diagram

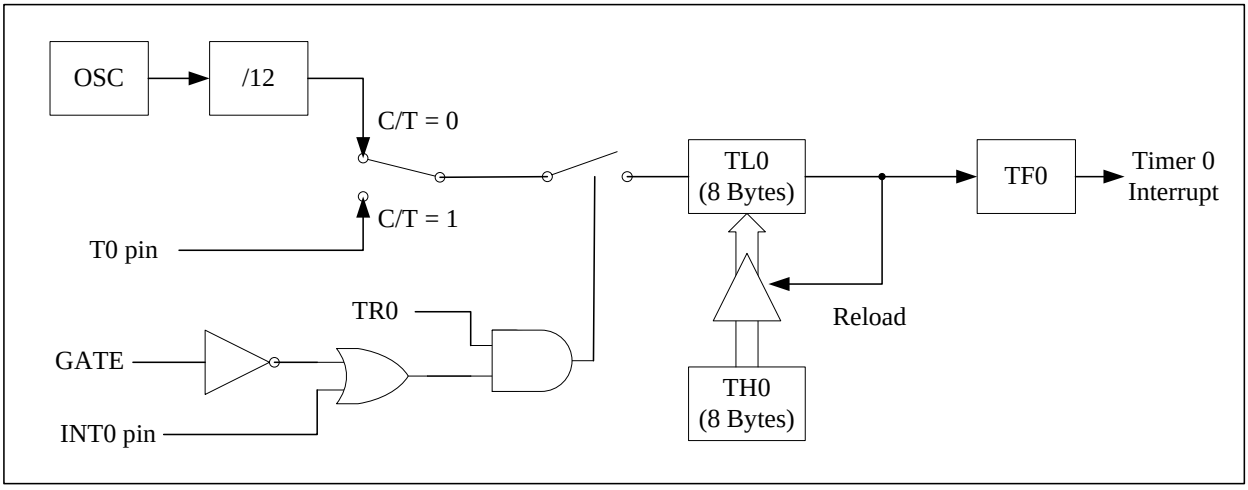


Figure 9-14: Timer 1 Mode 2 control block diagram

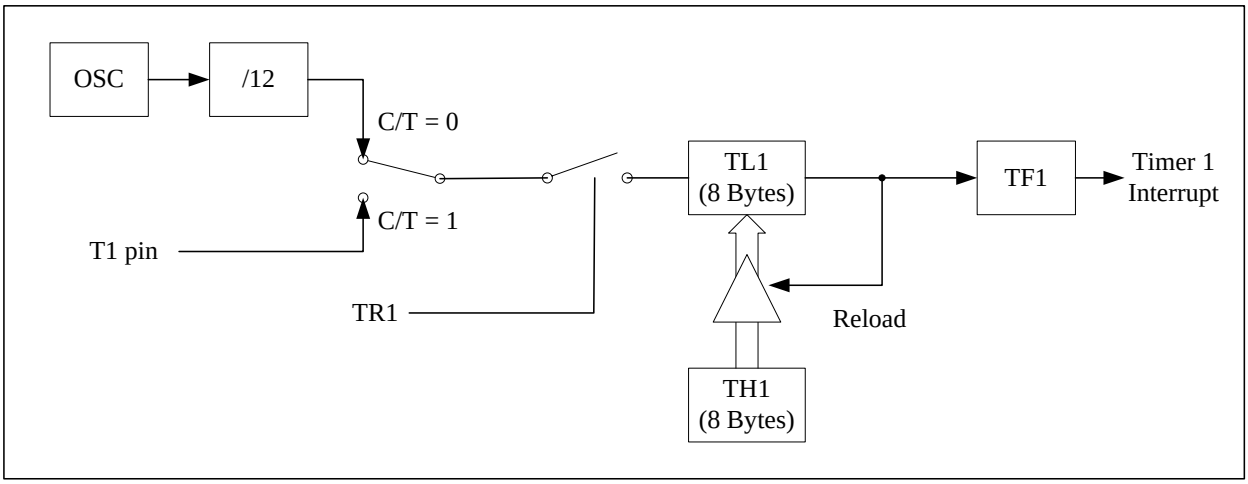
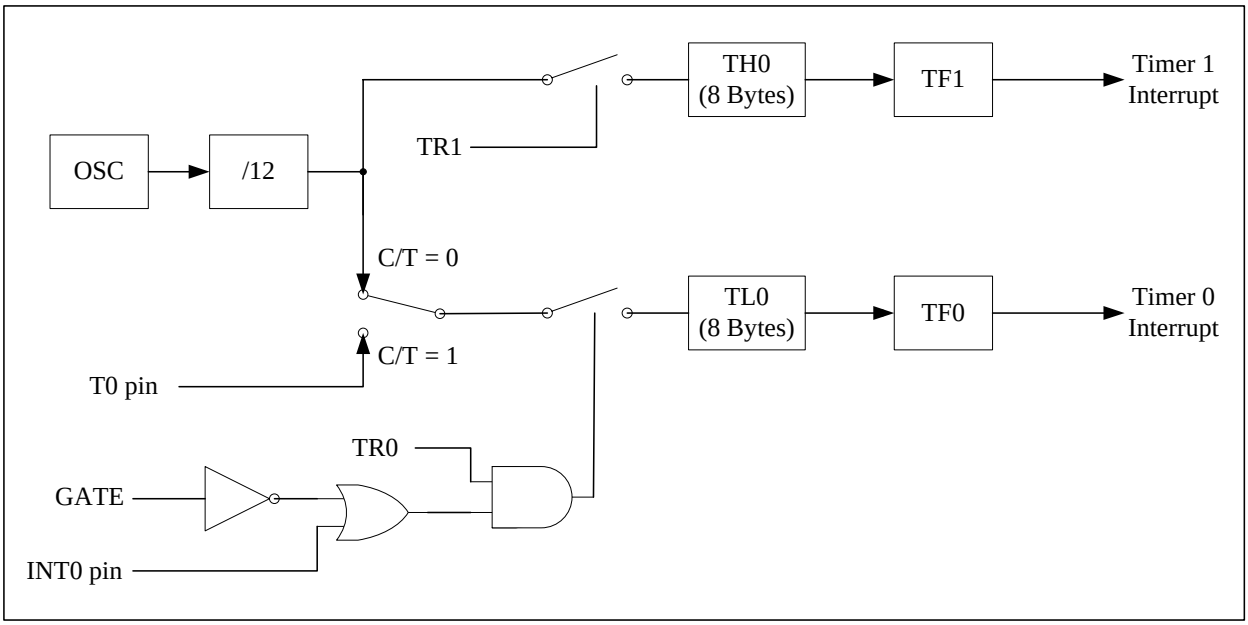


Figure 9-15: Timer 0 mode 3 control block diagram



9.5. Data Pointer

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
82h	DPL	Data pointer low byte	**DPTR[7:0]**								0000 0000
			r/w								
83h	DPH	Data pointer high byte	**DPTR[15:8]**								0000 0000
			r/w								
84h	DPL1	Data pointer 1 low byte	**DPTR 1 [7:0]**								0000 0000
			r/w								
85h	DPH1	Data pointer 2 high byte	**DPTR 1 [15:8]**								0000 0000
			r/w								
92h	DPS	Data pointer select	--	--	--	--	--	--	--	**DPS.0**	0000 0000
			--	--	--	--	--	--	--	r/w	

DPS.0: 0: select DPTR, 1: DPTR 1.

9.6. Power Control

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
87h	PCON	Power Control Reg	**SMOD**	--	--	--	**GF1**	**GF0**	--	--	0000 0000
			r/w								

SMOD: This bit doubles the UART baud rate in mode 1,2 and 3 when set to 1.
GF0/1: General purpose user flags.

9.7. Stack pointer

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
81h	SP	Stack Pointer Reg	SP[7:0]								0000 0000
			r/w								

The Stack Pointer stores the internal RAM address where the stack begins. In other words, it always points to the top of stack.

9.8. Program Status Word

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
D0h	PSW	Program status word	CY	AC	F0	RS1	RS0	OV	--	P	0000 0000
			r/w								

CY: Carry flag.
AC: Auxiliary carry.
F0: General purpose user flags.
OV: Overflow flag.
P: Parity flag.

9.9. Accumulator

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
E0h	ACC	Accumulator	ACC[7:0]								0000 0000
			r/w								

9.10. B Register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
F0h	B	B Register	B[7:0]								0000 0000
			r/w								

10. USB Registers

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7A8h	IVEC	Interrupt vector	--	**IV4**	**IV43**	**IV2**	**IV1**	**IV0**	--	--	0000 0000
			--	r	r	r	r	r	--	--	
7A9h	In07irq	Interrupt flag, in07	--	--	--	--	--	**In2ir**	**In1ir**	**In0ir**	0000 0000
			--	--	--	--	--	rw	rw	rw	
7AAh	Out07irq	Interrupt flag, out07	--	--	--	--	--	--	--	**Out0ir**	0000 0000
			--	--	--	--	--	--	--	rw	
7ABh	Usbirq	Interrupt flag, usb even	--	--	--	**uresir**	**suspir**	**sutokir**	**sofir**	**sudavir**	0000 0000
			--	--	--	rw	rw	--	--	rw	
7ACh	In07ien	Interrupt enable, in07	--	--	--	--	--	**In2ien**	**In1ien**	**In0ien**	0000 0000
			--	--	--	--	--	rw	rw	rw	
7ADh	Out07ien	Interrupt enable, out07	--	--	--	--	--	--	--	**Out0ien**	0000 0000
			--	--	--	--	--	--	--	rw	
7AEh	usbien	Interrupt enable, usb even	--	--	--	**uresie**	**suspie**	**sutokie**	**sofie**	**sudavie**	0000 0000
			--	--	--	rw	rw	--	--	rw	
7AFh	Usbbav	Packet info	**PID [3:0]**				**Data01**	**Data_stage**	**Status_stage**	**0**	0000 0000
			r				r	r	r	rw	
7B4h	Ep0cs	EP0 control	--	--	--	--	**outbsy**	**inbsy**	**hsnak**	**Ep0stall**	0000 0000
			--	--	--	--	r	r	rw	rw	
7B5h	In0bc	EP0 byte count	--	**In0bc[6:0]**							0000 0000
			--	rw							
7B6h	In1cs	EP1 IN control	--	--	--	--	--	--	**In1bsy**	**In1stl**	0000 0000
			--	--	--	--	--	--	rw	rw	
7B7h	In1bc	EP1 IN byte count	--	**In1bc[6:0]**							0000 0000
				rw							
7B8h	In2cs	EP2 IN control	--	--	--	--	--	--	**In2bsy**	**In2stl**	0000 0000
			--	--	--	--	--	--	rw	rw	
7B9h	In2bc	EP2 IN byte count	--	**In2bc[6:0]**							0000 0000
			rw	rw							
7C5h	Out0bc	EP0 OUT byte count	--	**Out0bc[6:0]**							0000 0000
			--	rw							
7D6h	Usbcs	USB even control	**wakesrc**	--	--	--	**discon**	--	--	**resume**	0000 1000
			rw	--	--	--	rw	--	--	rw	
7DEh	In07val	Data valid,	--	--	--	--	--	**In2val**	**In1val**	**In0val**	0000 0000

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
		in07	--	--	--	--	--	rw	Rw	rw	
7DFh	Out07val	Data valid, out07	--	--	--	--	--	--	--	**Out0val**	0000 0001
			--	--	--	--	--	--	--	rw	
7E8h | 7EFh	Setupbuf	STEUP data buffer	8 bytes								0000 0000
700h | 707h	In0buf	EP0 IN data buffer	8 bytes								0000 0000
680h | 687h	In1buf	EP1 IN data buffer	8 bytes								0000 0000
600h | 607h	In2buf	EP2 IN data buffer	8 bytes								0000 0000
6C0h | 6C7h	Out0buf	EP0 OUT data buffer	8 bytes								0000 0000

USB interrupt vector register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7A8h	IVEC	USB Interrupt vector	--	**IV4**	**IV43**	**IV2**	**IV1**	**IV0**	--	--	0000 0000
			--	r	r	r	r	r	--	--	

Vector of the active interrupt is available in the IVEC register. The table below shows the content of the IVEC register for the USB interrupts.

Source of interrupt	Interrupt register bit	Content of IVEC register
USB_SETUP_DATA_VALID_INT	Usbirq.0	0x00
USB_START_OF_FRAME_INT	Usbirq.1	0x04
USB_SETUP_TOKEN_INT	Usbirq.2	0x08
USB_SUSPEND_INT	Usbirq.3	0x0C
USB_RESET_INT	Usbirq.4	0x10
USB_EP0_IN_INT	In07irq.0	0x18
USB_EP0_OUT_INT	Out07irq.0	0x1C
USB_EP1_IN_INT	In07irq.1	0x20
USB_EP2_IN_INT	In07irq.2	0x28

USB IN transfer interrupts request register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7A9h	In07irq	Interrupt flag, in07	--	--	--	--	--	**In2ir**	**In1ir**	**In0ir**	0000 0000
			--	--	--	--	--	rw	rw	rw	

The USB sets the inxir bit to ‘1’ when it transmits IN x data packet and receives an ACK from host.

Write ‘1’ to this bit to clear the interrupt request.

USB OUT transfer interrupts request register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7AAh	Out07irq	Interrupt flag, out07	--	--	--	--	--	--	--	**Out0ir**	0000 0000
			--	--	--	--	--	--	--	rw	

The USB sets the out0ir bit to ‘1’ when it receives an error free OUT 0 data packet.

Write ‘1’ to this bit to clear the interrupt request.

USB Control transfer interrupts request register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7ABh	Usbirq	Interrupt flag, usb even	--	--	--	**uresir**	**suspir**	**sutokir**	**sofir**	**sudavir**	0000 0000
			--	--	--	rw	rw	--	--	rw	

The USB set corresponding bit to ‘1’ when it detect a interrupt request. Write ‘1’ to corresponding bit to clear the interrupt request.

Sudavir: SETUP token data valid interrupt request.

Sofir: Start-of-frame interrupt request.

Sutokir: SETUP token interrupt request.

Suspir: USB suspend interrupt request.

Uresir: USB reset interrupt request.

USB IN transfer interrupts enable register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7ACh	In07ien	Interrupt enable, in07	--	--	--	--	--	**In2ien**	**In1ien**	**In0ien**	0000 0000
			--	--	--	--	--	rw	rw	rw	

Set to ‘1’ is enable, set to ‘0’ is disable.

In0ien: EP0 IN transfer interrupt enable.

In1ien: EP1 IN transfer interrupt enable.

In2ien: EP2 IN transfer interrupt enable.

USB OUT transfer interrupts enable register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7ADh	Out07ien	Interrupt enable, out07	--	--	--	--	--	--	--	**Out0ien**	0000 0000
			--	--	--	--	--	--	--	rw	

Out0ien: EP0 OUT transfer interrupt set to ‘1’ is enable, set to ‘0’ is disable.

USB Control transfer interrupts enable register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7AEh	usbien	Interrupt enable, usb even	--	--	--	**uresie**	**suspie**	**sutokie**	**sofie**	**sudavie**	0000 0000
			--	--	--	rw	rw	--	--	rw	

Set to ‘1’ is enable, set to ‘0’ is disable.

Sudavie: SETUP token data valid interrupt enable.

Sofie: Start-of-frame interrupt enable.

Sutokie: SETUP token interrupt enable.

Suspie: USB suspend interrupt enable.

Uresie: USB reset interrupt enable.

USB packet information register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7AFh	Usbbav	Packet info	PID [3:0]				Data01	Data_stage	Status_stage	0	0000 0000
			r				r	r	r	rw	

PID [3:0]: packet ID code.

Data01: data packet toggle.

Data_stage: data stage is set to ‘1’, other stage is set to ‘0’.

Status_stage: status stage is set to ‘1’, other stage is set to ‘0’.

PID Type	PID Name	PID[3:0]	Description
Token	OUT	0001B	Address + endpoint number in host-to-function transaction
	IN	1001B	Address + endpoint number in function-to-host transaction
	SOF	0101B	Start-of-Frame marker and frame number
	SETUP	1101B	Address + endpoint number in host-to-function transaction for SETUP to a control pipe
Data	DATA0	0011B	Data packet PID even
	DATA1	1011B	Data packet PID odd
Handshake	ACK	0010B	Receiver accepts error-free data packet.
	NAK	1010B	Rx device cannot accept data or Tx device cannot send data.
	STALL	1110B	Endpoint is halted or a control pipe request is not supported.
Special	PRE	1100B	Host-issued preamble. Enables downstream bus traffic to low-speed devices.

Endpoint 0 control and status register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7B4h	Ep0cs	EP0 control	--	--	--	--	**outbsy**	**inbsy**	**hsnak**	**Ep0stall**	0000 0000
			--	--	--	--	r	r	rw	rw	

Ep0stall: Endpoint 0 stall. If this bit is set to '1', the USB sends a STALL handshake for ant IN or OUT token. This is done in the data or handshake phases of the CONTROL transfer.

Hsnak: if this bit is set to '1', the USB responds with a NAK handshake for every packet in the status stage. Hsnak bit is automatically set when a SETUP token arrives. Clear this bit by writing to '1'.

Inbsy: IN 0 Endpoint busy bit. This is read only bit that is automatically cleared when a SETUP token arrives. Set this bit by reloading the ep0inbc register.

Outbsy: OUT 0 Endpoint busy bit. This is read only bit that is automatically cleared when a SETUP token arrives. Set this bit by writing a dummy value to the ep0outbc register.

Endpoint 0 IN transfer byte count register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7B5h	In0bc	EP0 byte count	--	**In0bc[6:0]**							0000 0000
			--	rw							

After loading the IN 0 Endpoint data buffer, write to the in0bc register with the number of loaded bytes then start transfer data. When write this register the inbsy bit also set to '1'.

Endpoint 1 IN transfer control and status register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7B6h	In1cs	EP1 IN control	--	--	--	--	--	--	**In1bsy**	**In1stl**	0000 0000
			--	--	--	--	--	--	rw	rw	

In1stl: IN 1 Endpoint stall. If in1stl = '1', the USB returns a STALL handshake for all requests to the Endpoint 1.

In1bsy: IN 1 Endpoint busy bit. When the host sends an IN token for IN 1 endpoint and the in1bsy bit is set, the USB will respond with in1bc size data packet and clears the in1bsy bit. If in1bsy = '0', the IN 1 endpoint is empty and ready for loading data. When in1bsy = '1', the CPU should not access the IN 1 endpoint buffer.

Endpoint 1 IN transfer byte count register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7B7h	In1bc	EP1 IN byte count	--	**In1bc[6:0]**							0000 0000
				rw							

After loading the IN 1 Endpoint data buffer, write to the in1bc register with the number of loaded bytes then start transfer data. When write this register the in1bsy bit also set to '1'.

Endpoint 2 IN transfer control and status register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7B8h	In2cs	EP2 IN control	--	--	--	--	--	--	**In2bsy**	**In2stl**	0000 0000
			--	--	--	--	--	--	rw	rw	

In2stl: IN 2 Endpoint stall. If in2stl = '1', the USB returns a STALL handshake for all requests to the Endpoint 2.

In2bsy: IN 2 Endpoint busy bit. When the host sends an IN token for IN 2 endpoint and the in2bsy bit is set, the USB will respond with in1bc size data packet and clears the in2bsy bit. If in2bsy = '0', the IN 2 endpoint is empty and ready for loading data. When in2bsy = '1', the CPU should not access the IN 2 endpoint buffer.

Endpoint 2 IN transfer byte count register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7B9h	In2bc	EP2 IN byte count	--	**In2bc[6:0]**							0000 0000
			rw	rw							

After loading the IN 2 Endpoint data buffer, write to the in2bc register with the number of loaded bytes then start transfer data. When write this register the in2bsy bit also set to '1'.

Endpoint 0 OUT transfer byte count register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7C5h	Out0bc	EP0 OUT byte count	--	**Out0bc[6:0]**							0000 0000
			--	rw							

The out0bc register contains the number of bytes sent during the last OUT transfer from the host to an OUT 0 endpoint. The out0bc is a read only register, updated by the USB.

USB control and status register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7D6h	Usbcs	USB even control	**wakesrc**	--	--	--	**discon**	--	--	**resume**	0000 1000
			rw	--	--	--	rw	--	--	rw	

Resume: signal remote device resume. If sets this bit to '1', sets K state on the USB bus.

Discon: USB disconnect. If discon = '1', disconnect internal D- pull-up resistor. If discon = '0', connect internal D- pull-up resistor.

Wakesrc: USB wakeup. This bit indicates that a wakeup signals resumed the USB. Reset this bit by writing to '1'.

Endpoint x IN valid register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7DEh	In07val	Data valid, in07	--	--	--	--	--	**In2val**	**In1val**	**In0val**	0000 0000
			--	--	--	--	--	rw	Rw	rw	

In0val: if set to '1', IN 0 endpoint is active, if set to '0', IN 0 endpoint is inactive.

In1val: if set to '1', IN 1 endpoint is active, if set to '0', IN 1 endpoint is inactive.

In2val: if set to '1', IN 2 endpoint is active, if set to '0', IN 2 endpoint is inactive.

Endpoint 0 OUT valid register

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
7DFh	Out07val	Data valid, out07	--	--	--	--	--	--	--	**Out0val**	0000 0001
			--	--	--	--	--	--	--	rw	

Out0val: if set to '1', OUT 0 endpoint is active, if set to '0', OUT 0 endpoint is inactive.

10.1. The USB control endpoints implementation

10.1.1. Control transfer examples

A Control transfer consists of two or tree stages:

- Setup stage
- Data stage (optional)
- Status stage

a) Control Write Transfer example

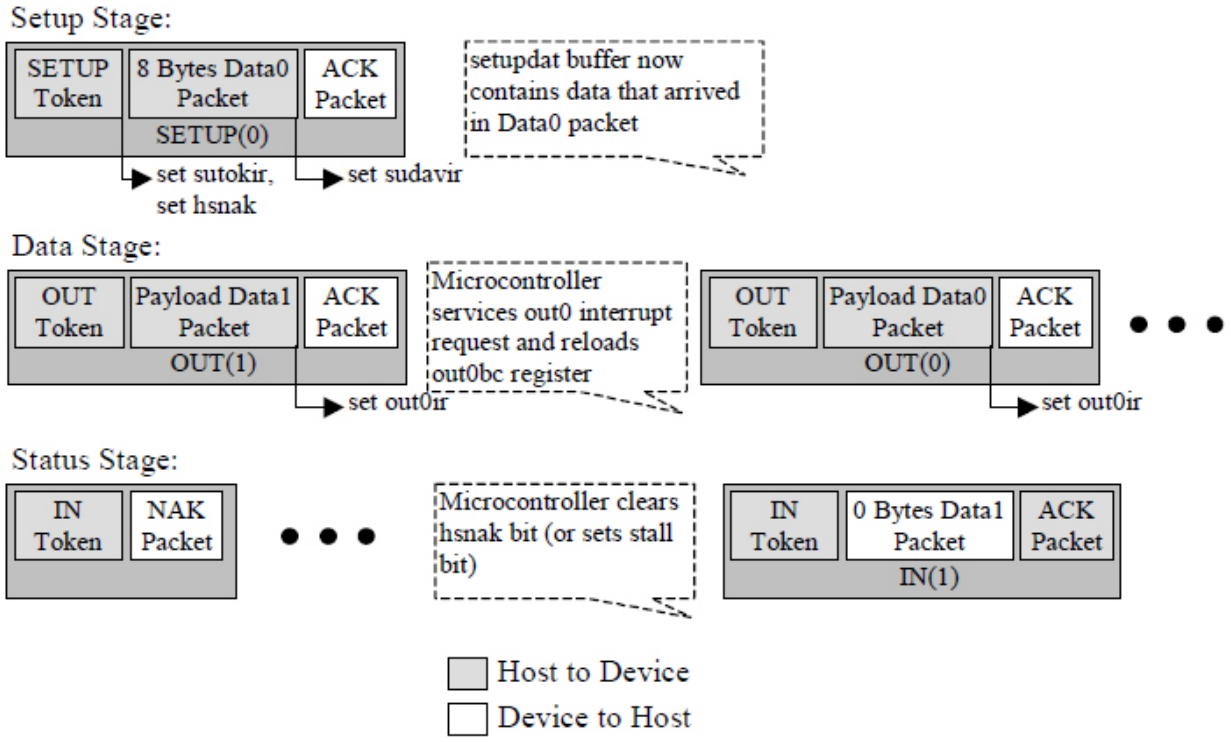


Figure 10-1: Control Write Transfer

After receiving the SETUP token, the ET81P304 USB sets the hsnak and sutokir bits. If an 8-byte data packet is received correctly, the ET81P304 USB sets the sudavir bit. Setting sutokir and (or) sudavir bits generates the appropriate interrupts. The data stage consists of one or more OUT bulk like transactions. The ET81P304 USB generates OUT 0 interrupt request by setting the out0ir bit after each correct OUT transaction during the data stage. Out0bc register contains the number of data bytes received in the last OUT transaction. The microcontroller should service the interrupt request and then prepare the endpoint for the next transaction by reloading the out0bc register with any value (setting outbusy bit). The status stage of a control transfer is the last operation in the sequence. The microcontroller should clear hsnak bit (by writing 1 to it) to instruct the ET81P304 USB to ACK the status stage.

b) Control Read Transfer example

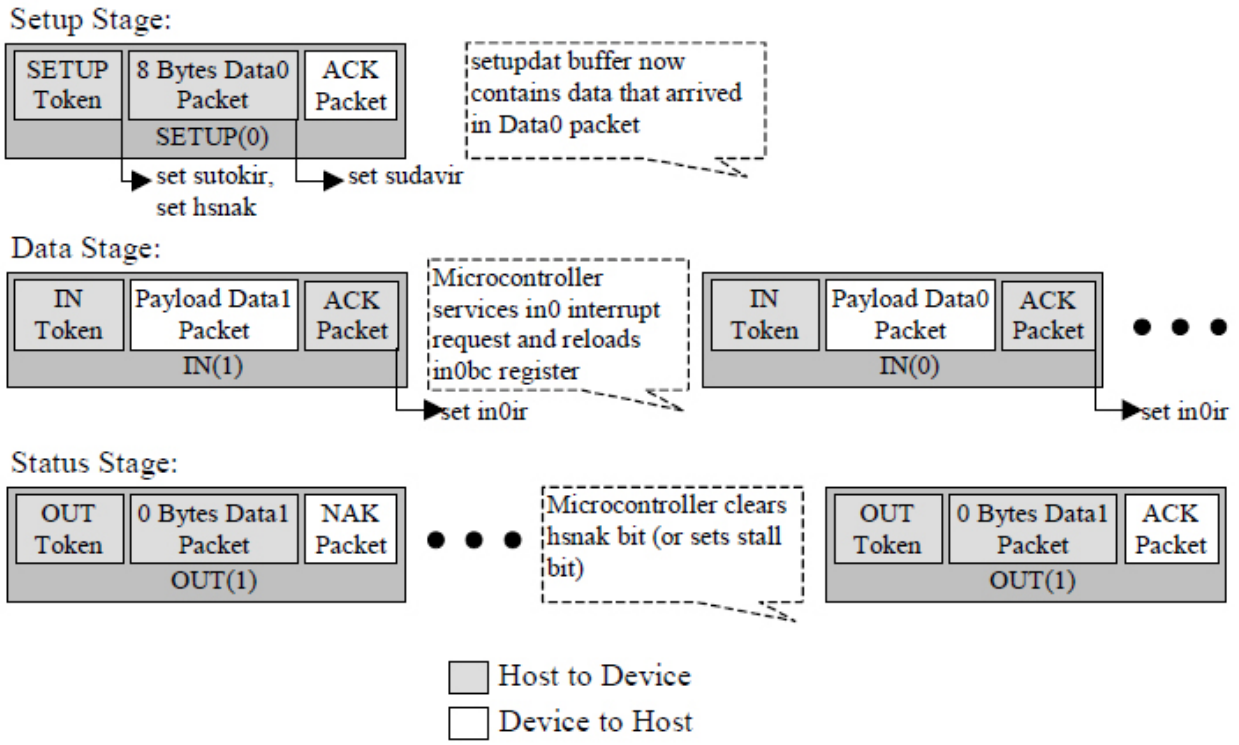


Figure 10-2: Control Read Transfer

Control read transfer is similar to control write transfer. The difference is in the data stage. During data stage of control read transfers the ET81P304 USB generates IN 0 interrupt request by setting in0ir bit. This is done after each acknowledge by the host data packet. The microcontroller should load new data into IN 0 buffer and then reload in0bc register with a valid number of loaded data. Reloading the in0bc register causes setting inbsy bit and arming endpoint for the next IN transaction. The status stage of a control transfer is the last operation in the sequence. The microcontroller should clear hsnak bit (by write 1 to it) to instruct the ET81P304 USB to ACK the status stage.

c) No-Data Control Transfer example

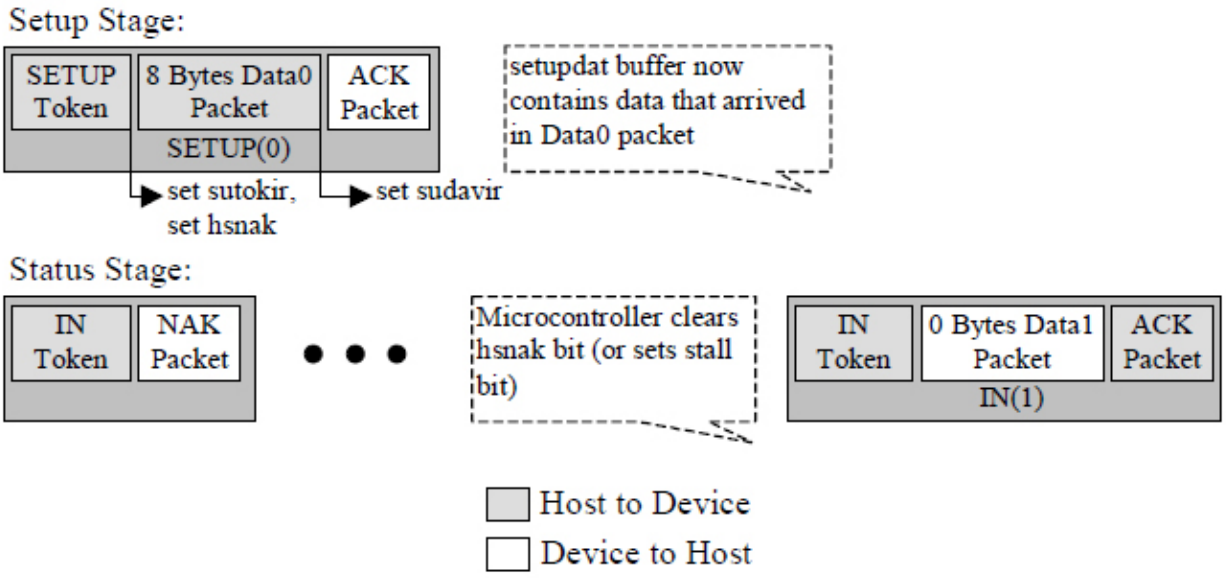


Figure 10-3: No-data Control Transfer

Some control transfer don't have a data stage. In this case the status stage consists of IN data packet. The microcontroller should clear hsnak bit (by write 1 to it) to instruct the ET81P304 USB to the ACK the status stage.

10.1.2. Interrupt IN transfer example

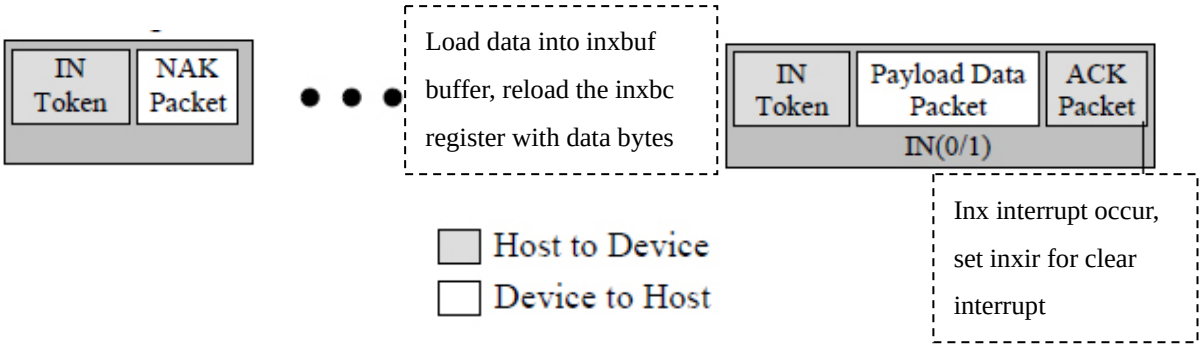


Figure 10-4: Interrupt IN Transfer

When the host wants to receive interrupt data, it issues an IN token. If microcontroller load new data into inxbuf buffer and then reload the inxbc register with valid number of data bytes, if host receives a valid data packet, it responds with and ACK handshake. After receiving a valid ACK handshake from host, the ET81P304 USB will generates an interrupt request for IN x endpoint if inxir bit has been sets. During a service interrupt request the microcontroller should set the inxir for clear interrupt request.

If inxbc register is not reload data byte count, the ET81P304 USB returns NAK handshake for each IN token from the host.

10.2.USB Suspend and Resume

When occur USB Suspend interrupt that system will into power down mode for stop CPU and USB clock, the Suspend and Resume control flow as follows:

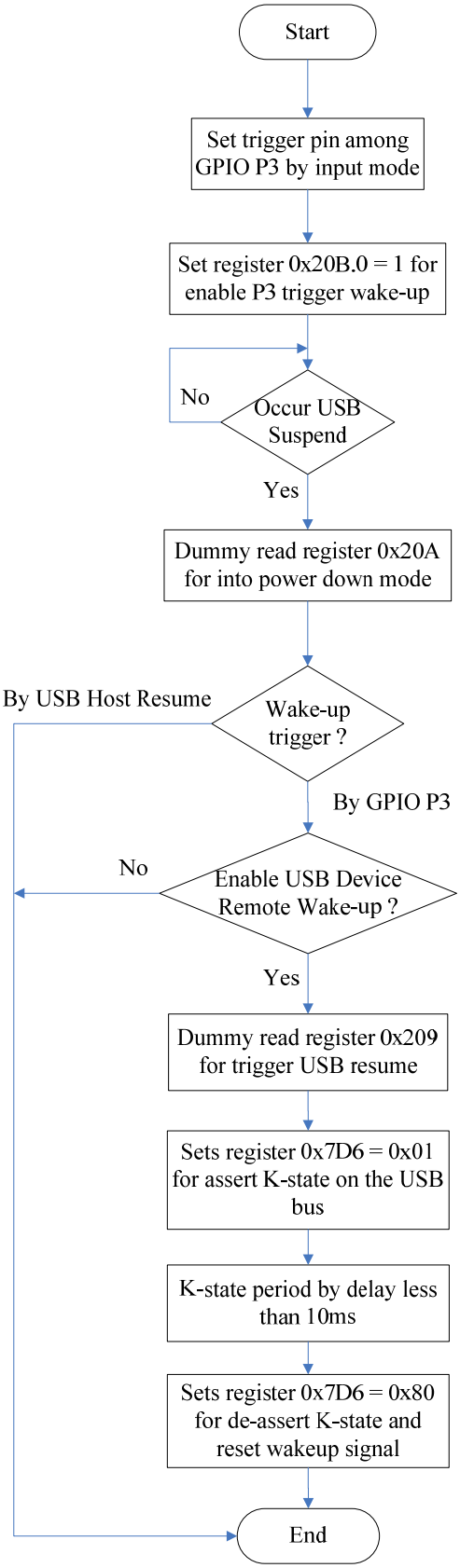
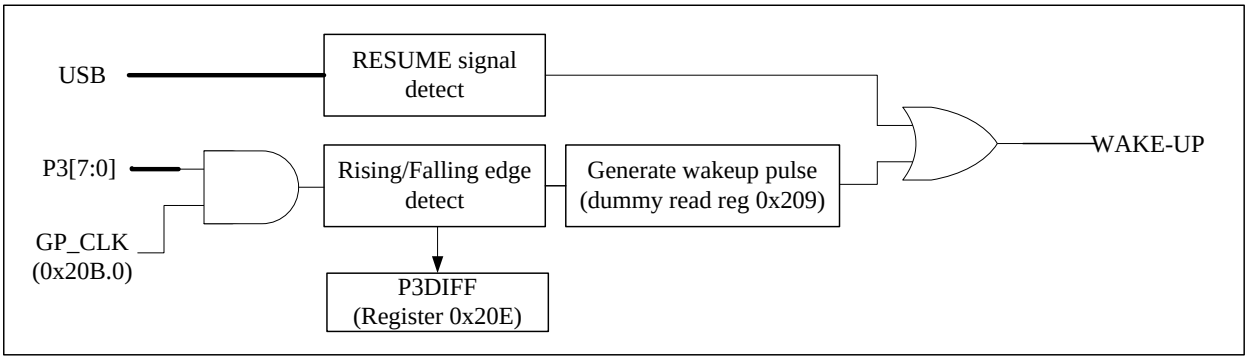


Figure 10-5: Resume and Remote Wake-up Logic



USB Suspend and Resume demo code:

```
void USB_Suspend(void)
{
    Unsigned char temp,TriggerP3;

    FEATURE_ENA |= 0x01;                    //sets register 0x20B bit 0 for enable GPIO wake-up
    TriggerP3 = P3DIFF;                     //dummy read register 0x20E for clear register
    temp = SUSPEND;                         //dummy read register 0x20A for stop system clock

    _nop_();                                //for nothing after stop clock

    TriggerP3 = P3DIFF;                     //read register 0x20E for check port 3 which pin triggered
    if(DeviceRemoteWakeup)                  //if HOST enable Device_Remote_Wakeup ?
    {
        if(TriggerP3)                       //if any P3 pin was triggered ?
        {
            temp = WAKEUP_PULSE;            //dummy read register 0x209 to generate wake-up pulse

            USB_usbcs = 0x01;               //Sets register 0x7D6 = 0x01 for assert K-state on the USB bus
            Delay(80);                                  //delay 8ms for K-state period
            USB_usbcs = 0x80;   //Sets register 0x7D6 = 0x80 for de-assert K-state and reset wakeup signal
        }
    }
}
```

11. System Control Registers

ADR	NAME	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
202h	STATUS	WDT_FLAG	--	--	--	HWCNT	BRCNT	VBUS	TMODE	0000 0000
203h	WDTCON	--	--	--	--	WDTEN	WDTCFG2	WDTCFG1	WDTCFG0	0000 0111
204h	WDTCLR	WDTCLR[7:0]								0000 0000
205h	EXTCLK	OSC_SEL	XIXO_SEL	RSTB_SEL	EN_BR6M	CLK_DIV				1111 1111
206h	BR_CALB	EN_EXTCLK	BR_CALB[5:0]							1110 0000
209h	WK_PUL	WK_PUL[7:0]								0000 0000
20Ah	SUSPEND	SUSPEND[7:0]								0000 0000
20Bh	FEATURE_ENA	UART0_EN	--	--	--	--	--	--	GP_CLK	0000 1001
20Ch	SYSCFG	OST_40K	OST_6M		--	--	--	--	--	1111 1111
20Eh	P3DIFF	P3DIFF[7:0]								0000 0000
21Ah	P1_PU	P1_PU[7:0]								0000 0000
21Ch	P3_PU	P3_PU[7:0]								0000 0000
21Eh	P5_PU	P5_PU[7:0]								0000 0000

11.1. System Status Register

ADR	NAME	Content	b7	b6	b5	b4	b3	B2	b1	b0	Value on POR
202h	STATUS	System status	**WDT_FLAG**	**OTP BUSY**	--	--	**HWCNT**	**BRCNT**	**VBUS**	**TMODE**	0000 0000
			r	r	--	--	r	r	r	r	

TMODE: 1: in test mode, 0: in normal mode.

VBUS: 1: USB connect, 0: USB disconnect.

BRCNT: BR clock counting flag in BR calibration, 1: counting, 0: stop count.

HWCNT: External clock counting flag in BR calibration, 1: counting, 0: stop count.

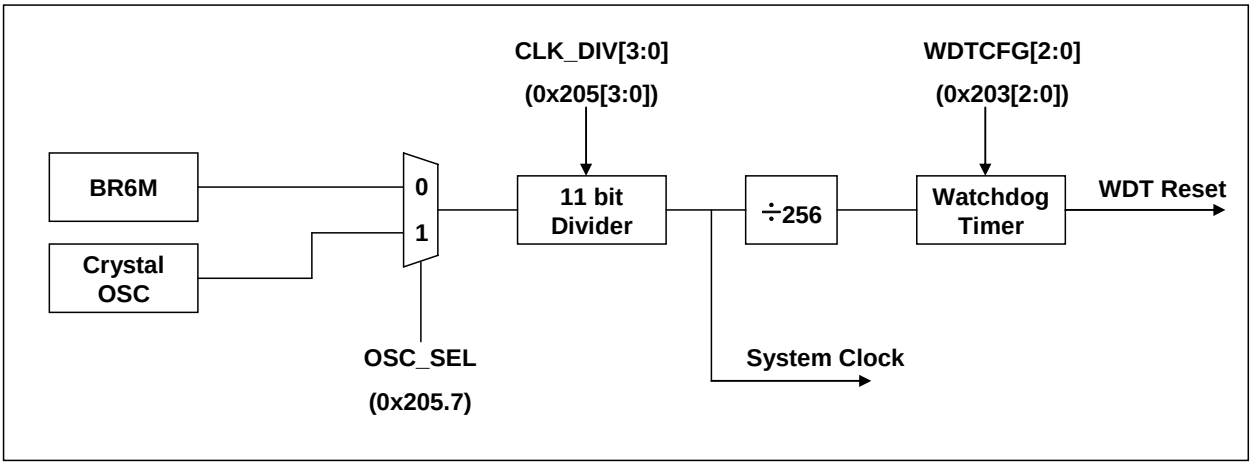
OTP BUSY: busy flag when OTP ROM in write if 1.

WDT_FLAG: Watch Dog Timer occur flag, 1: Timers overflow and reset CPU.

11.2. Watchdog Timer

The clock source of watchdog timer is from system clock, there are two types of clock source for system clock; Build-in 6MHz RC oscillator and crystal oscillator. All clock sources are passing through 11bit divider to whole system. Watchdog timer is shown in Figure 11-1.

Figure 11-1: Watchdog Timer Block Diagram



Watchdog Timer control registers

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
203h	WDT	Watch Dog Timer Control Reg	--	--	--	--	**WDT EN**	**WDT CFG2**	**WDT CFG1**	**WDT CFG0**	0000 0111
			--	--	--	--	r/w	r/w	r/w	r/w	
204h	WDTCLR	Clear Watch Dog Timer	**WDTCLR[7:0]**								0000 0000
			r								

WDTCLR: Dummy read to clear Watch Dog Timer.

WDTEN : Watch dog timer enable bit, 0=Disable, 1 = Enable.

WDTCFG[2:0] : The limitation of WDT counter (unit: 256 CPU clock per count)

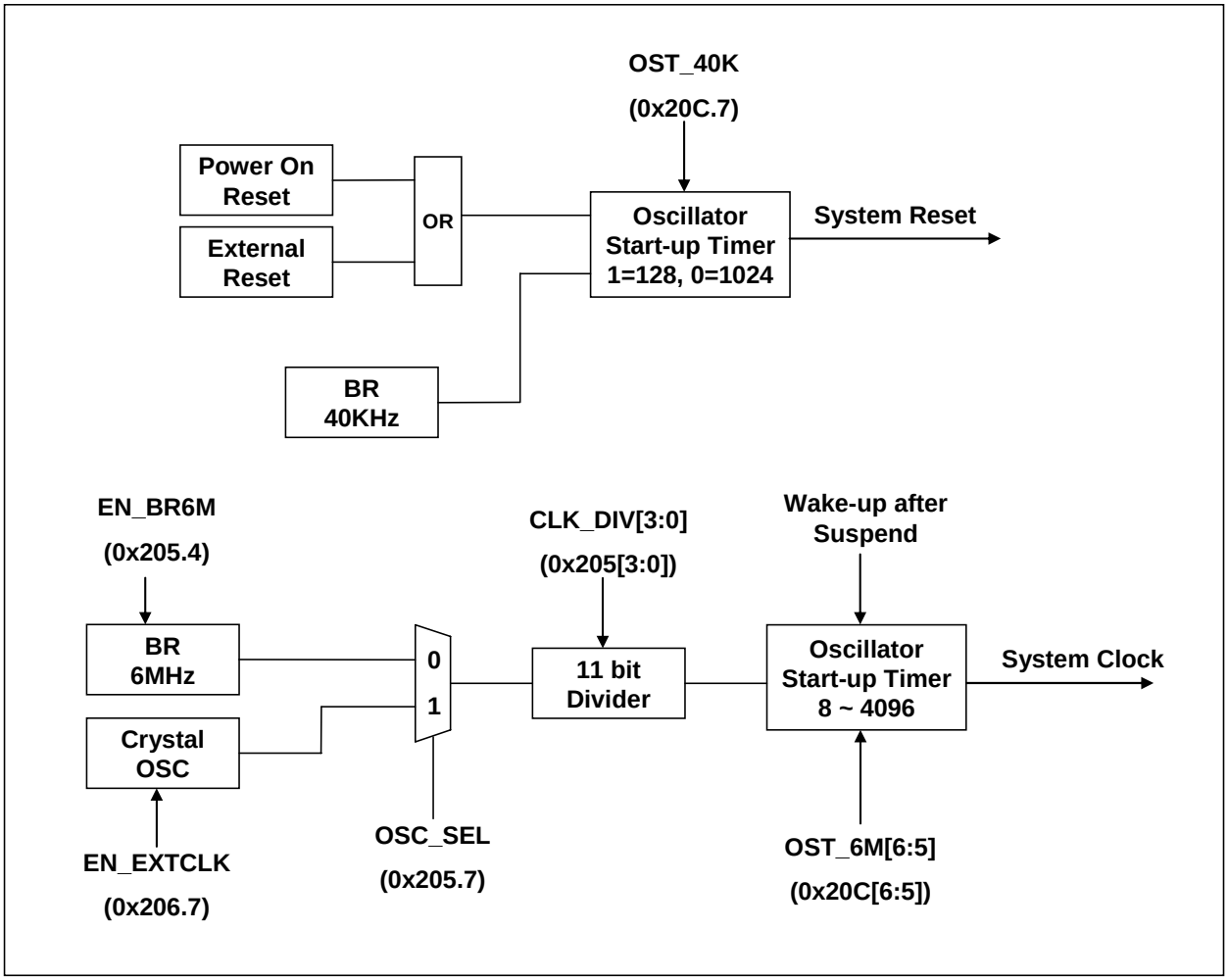
000 : 4 count	100 : 64 count
001 : 8 count	101 : 128 count
010 : 16 count	110 : 256 count
011 : 32 count	111 : 512 count

11.3. Clock System

There are three oscillators integrated in this CPU, include crystal oscillator, build-in 6MHz RC oscillator and build-in 40KHz RC oscillator.

There are two optional oscillator start-up timer (OST), one is used to ensure that the oscillator has started and stabilized after wake up, it provides a delay of 8, 128, 1024 and 4096 cycles after oscillator is enable. The other one is used to ensure system reset successful, it provides a delay of 128 or 1024 cycles in system reset stage. Clock system is shown in Figure 11-2.

Figure 11-2: Clock System Block Diagram



Clock System Control Registers:

ADR	NAME	Content	b7	b6	b5	b4	b3	B2	b1	b0	Value on POR
205h	EXTCLK	Ext Clock Divider	**OSC_SEL**	**XIXO_SEL**	**RSTB_SEL**	**EN_BR6M**	**CLK_DIV**				1111 1111
			r/w	r/w	r/w	r/w	r/w				

CLK_DIV [3:0]: System clock divider.

CPU clock = External clock / 2^{CLK_DIV}, CLK_DIV = 0~11,

Default and other value: CPU clock = external clock / 4.

OSC_SEL: System clock selection, 0 = Build-in 6MHz RC oscillator, 1 = Crystal oscillator.

XIXO_SEL: XIN/P5.6 and XOUT/P5.7 pin selection,

1 = XIN/XOUT

0 = Port P5.6/P5.7 (bit 7 OSC_SEL should be set 0 at same time)

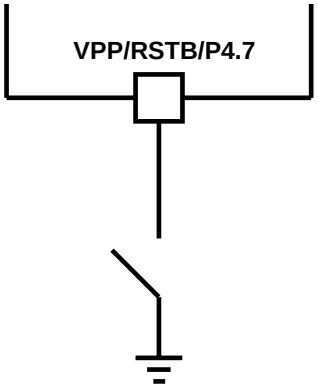
RSTB_SEL: RSTB/P4.7 pin selection, 1 = RESET pin, active low as shown Figure 11-3.

0 = Port P4.7 pin, input only with pull-up resister.

EN_BR6M: Build-in 6MHz RC oscillator enable bit, 1: Enable 0: Disable.

Note: This register setting will mapping from Configuration ROM address 0xFE2 in power on reset stage.

Figure 11-3: External reset configuration



ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
20Ch	SYSCFG	System Configuration	OST_40K	OST_6M		--	--	--	--	--	1111 1111
			r/w	r/w		--	--	--	--	--	

OST_40K: Oscillator 40KHz Start-up Timer Delay

1 = 128 cycles, 0 = 1024 cycles

OST_6M: Oscillator 6MHz Start-up Timer Delay

11 = 8 cycles, 10 = 128 cycles

01 = 1024 cycles, 00 = 4096 cycles

Note: This register setting will mapping from Configuration ROM address 0xFE3 in power on reset stage..

ADR	NAME	Content	b7	b6	b5	b4	b3	B2	b1	b0	Value on POR
206h	BR_CALB	BR Calibration	EN_EXTCLK	BR_CALB[6:0]							1110 0000
			r/w	r/w							

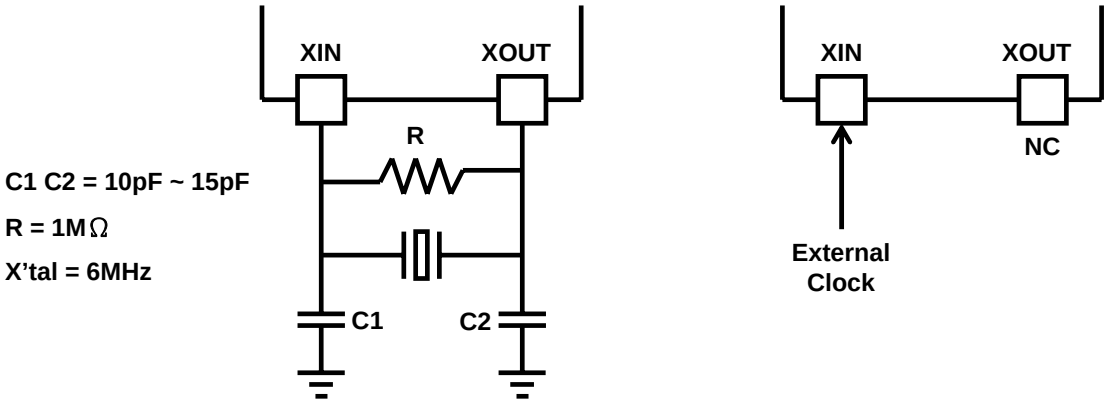
BR_CALB: Build-in 6MHz RC oscillator calibration value.

EN_EXTCLK: External clock enable bit, 1 = Enable, 0 = Disable.

The external clock configuration has shown Figure 11-4.

Note: This register setting will mapping from Configuration ROM address 0xFE0 in power on reset stage.

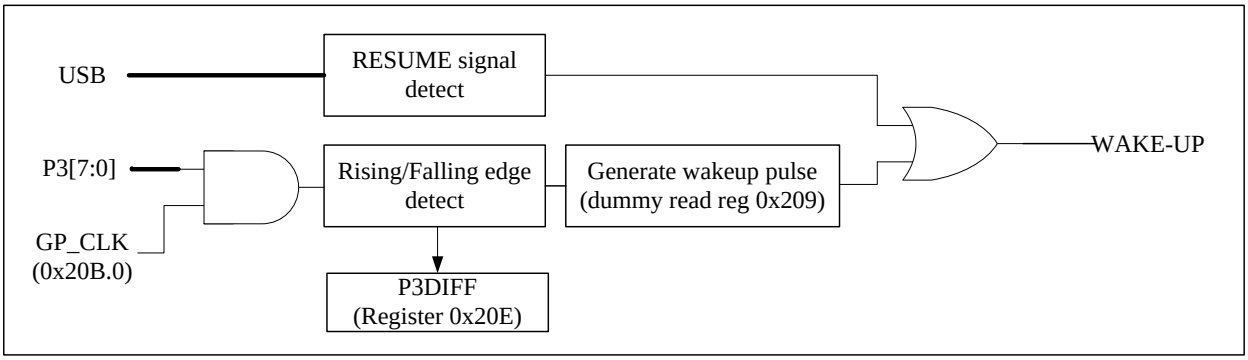
Figure 11-4: External clock configuration



11.4. USB suspend and resume control registers

The Suspend and Resume control registers is control ET81P304 power down and wakeup function. Dummy read register 0x20A will into power down mode that system will stop CPU and USB clock, usually combine with USB suspend interrupt for compliance USB specification. The Wakeup trigger source by GPIO P3 or USB RESUME signal as figure 11.5.

Figure 11-5: Resume and Remote Wake-up Logic



Suspend and resume control registers

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
209h	WK_PUL	Generate a wakeup pulse	**WK_PUL[7:0]**								0000 0000
			r								
20Ah	SUSPEND	Go to suspend state	**SUSPEND[7:0]**								0000 0000
			r								
20Bh	FEATURE_ENA	Feature enable	**UART0_EN**	--	--	--	--	--	--	**GP_CLK**	0000 1001
			r/w	--	--	--	--	--	--	r/w	
20Eh	P3DIFF	P3 changing flag	**P3DIFF[7:0]**								0000 0000
			r								

WK_PUL [7:0]: Dummy read for USB resume when USB into suspend state.

SUSPEND [7:0]: Dummy read into power down mode in USB suspend (stop CPU and USB clock)

GP_CLK: GPIO P3 trigger wake-up when CPU into power down mode

1 = Enable, 0 = Disable.

UART0_EN: Pin P3.0/P3.1 attribution.

1 = UART0 RX/TX.

0 = GPIO port P3.0/P3.1.

P3DIFF [7:0]: P3 changing during power down mode.

11.5. GPIO pull-up control registers

ADR	NAME	Content	b7	b6	b5	b4	b3	b2	b1	b0	Value on POR
21Ah	P1_PU	Port 1 pull-up	--	--	--	**P1_PU[4:3]**		--	--	--	0000 0000
			--	--	--	r/w		--	--	--	
21Ch	P3_PU	Port 3 pull-up	**P3_PU[7:0]**								0000 0000
			r/w								
21Eh	P5_PU	Port 5 pull-up	**P5_PU[7:6]**		--	--	--	--	--	--	0000 0000
			r/w		--	--	--	--	--	--	

GPIO port pull-up resister control, only for input mode.

1: enable.

0: disable.

12. Programming Pin

for "A" version:

VDD: 5V power supply.

VSS: Ground.

VPP: 3.3V for OTP ROM Read, 6.5V ±0.25V for OTP ROM Write

P3.0: UART RXD

P3.1: UART TXD

XIN: External clock = 6MHz.

for "B" version and later version:

VDD: 5V power supply.

VSS: Ground.

VPP: 3.3V for OTP ROM Read, 6.5V ±0.25V for OTP ROM Write

P3.0: UART RXD

P3.1: UART TXD

P1.4: External clock = 6MHz.

13. Configuration ROM

The Configuration ROM contains device configuration data in OTP ROM address 0xFE0~0xFFF, which is addressed only in device programming mode.

Address	Function	B7	B6	B5	B4	B3	B2	B1	B0
0xFE0	BOSC	EN_EXTCLK	BR_CALB[6:0]						
0xFE1	ROM_RD	--	--	--	--	--	--	--	RDEN
0xFE2	EXTCLK	OSC_SEL	XIXO_SEL	RSTB_SEL	EN_BR6M	CLK_DIV			
0xFE3	SYSCFG	OST_40K	OST_6M		--	--	--	--	--

Address 0xFE0: Same as system register 0x206.
BR_CALB: Build-in 6MHz RC oscillator calibration value.
EN_EXTCLK: External clock enable bit, 1 = Enable, 0 = Disable.

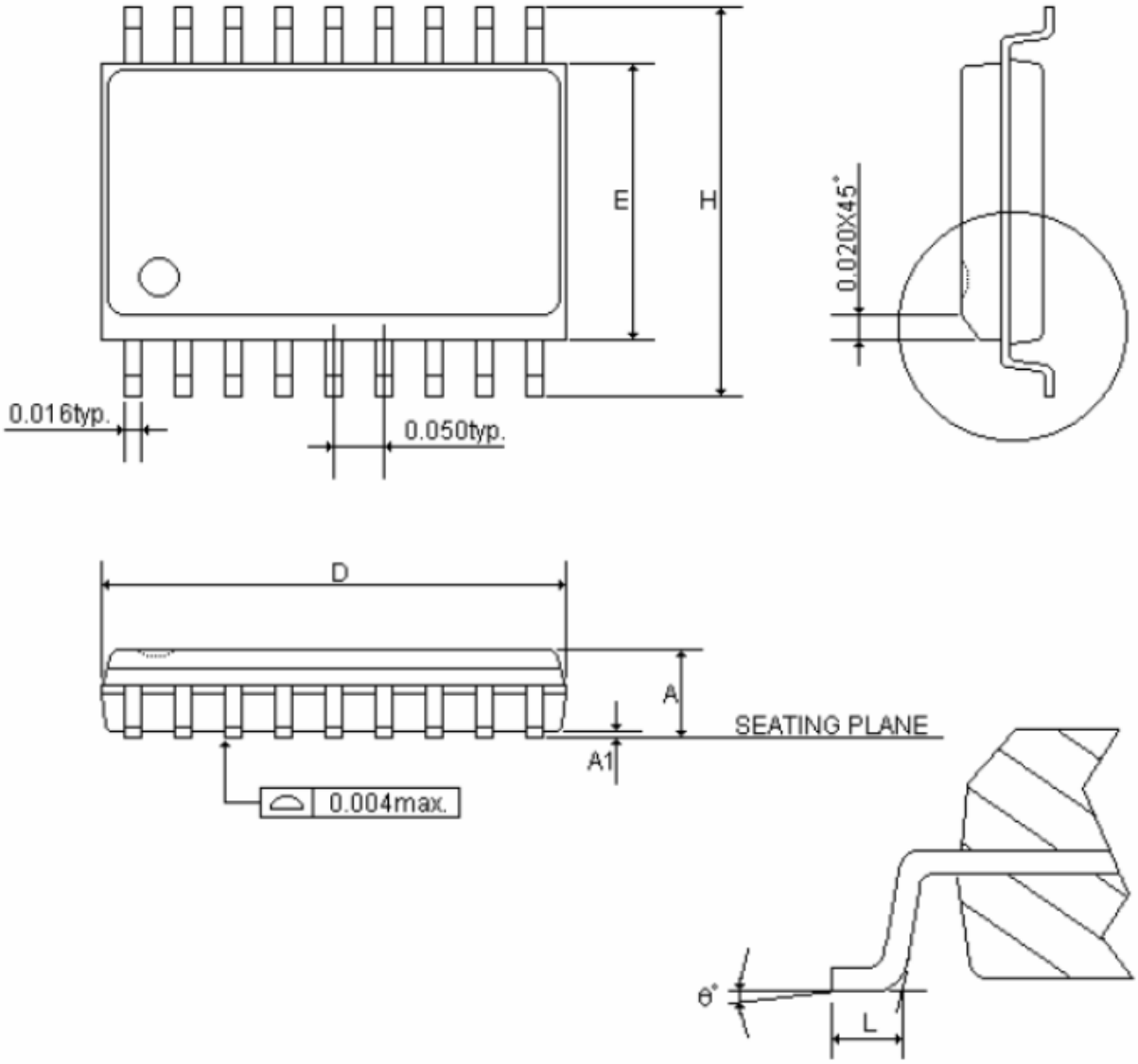
Address 0xFE1: RDEN = 1, OTP ROM read enable.
RDEN = 0, OTP ROM read disable.

Address 0xFE2: Same as system register 0x205.
CLK_DIV [3:0]: System clock divider.
OSC_SEL: 0 = Build-in 6MHz RC oscillator, 1 = Crystal oscillator.
XIXO_SEL: 1 = select XIN/XOUT pins
0 = select Port P5.6/P5.7 (have to set OSC_SEL = 0)
RSTB_SEL: 1 = RESET pin, active low.
0 = Port P4.7 pin, input only with pull-up resister.
EN_BR6M: Build-in 6MHz RC oscillator enable bit, 1: Enable 0: Disable.

Address 0xFE3: Same as system register 0x20C.
OST_40K: Oscillator 40KHz Start-up Timer Delay
1 = 128 cycles, 0 = 1024 cycles
OST_6M: Oscillator 6MHz Start-up Timer Delay
11 = 8 cycles, 10 = 128 cycles
01 = 1024 cycles, 00 = 4096 cycles

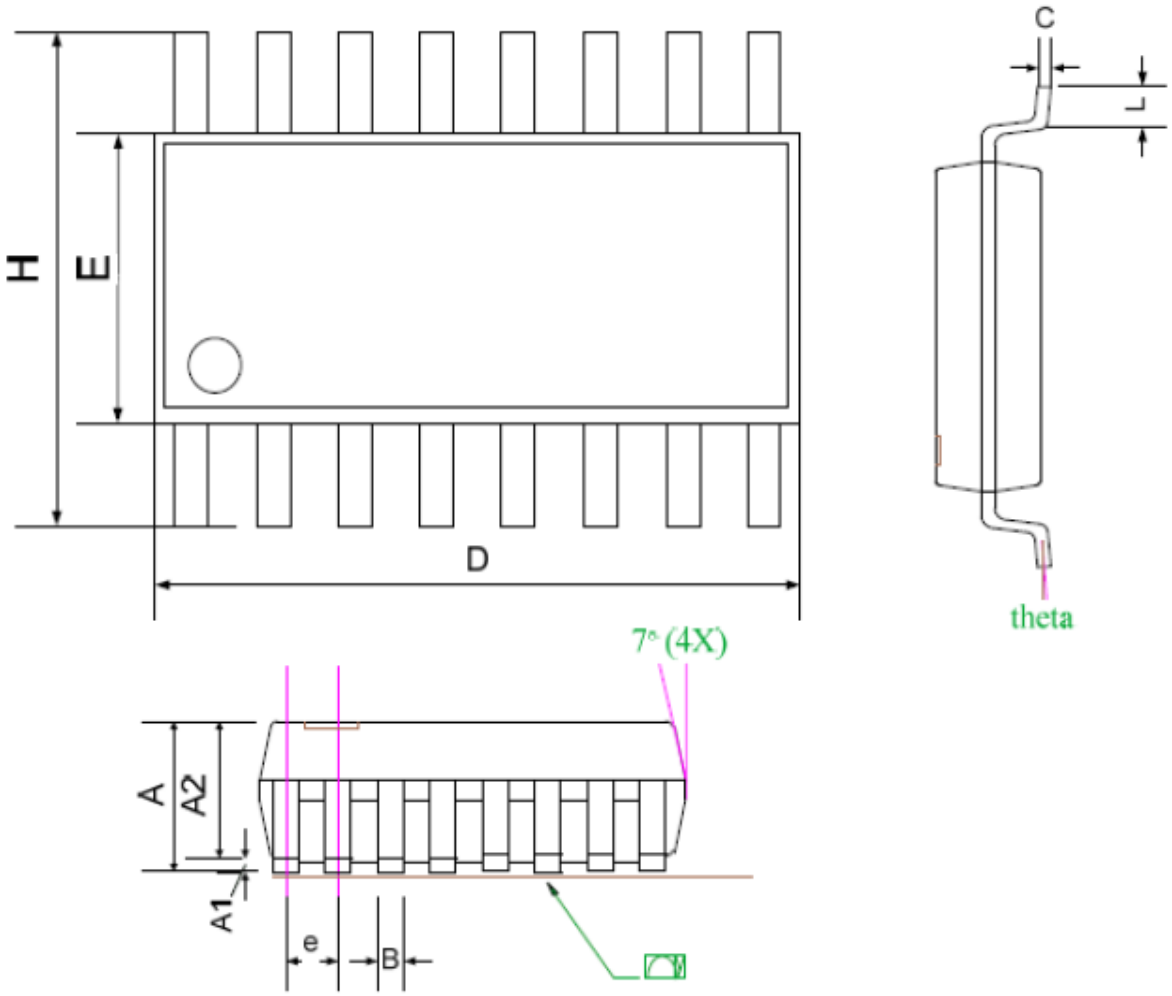
14. Package Outlines

SOP18: plastic small outline package; 18 leads; 300mil



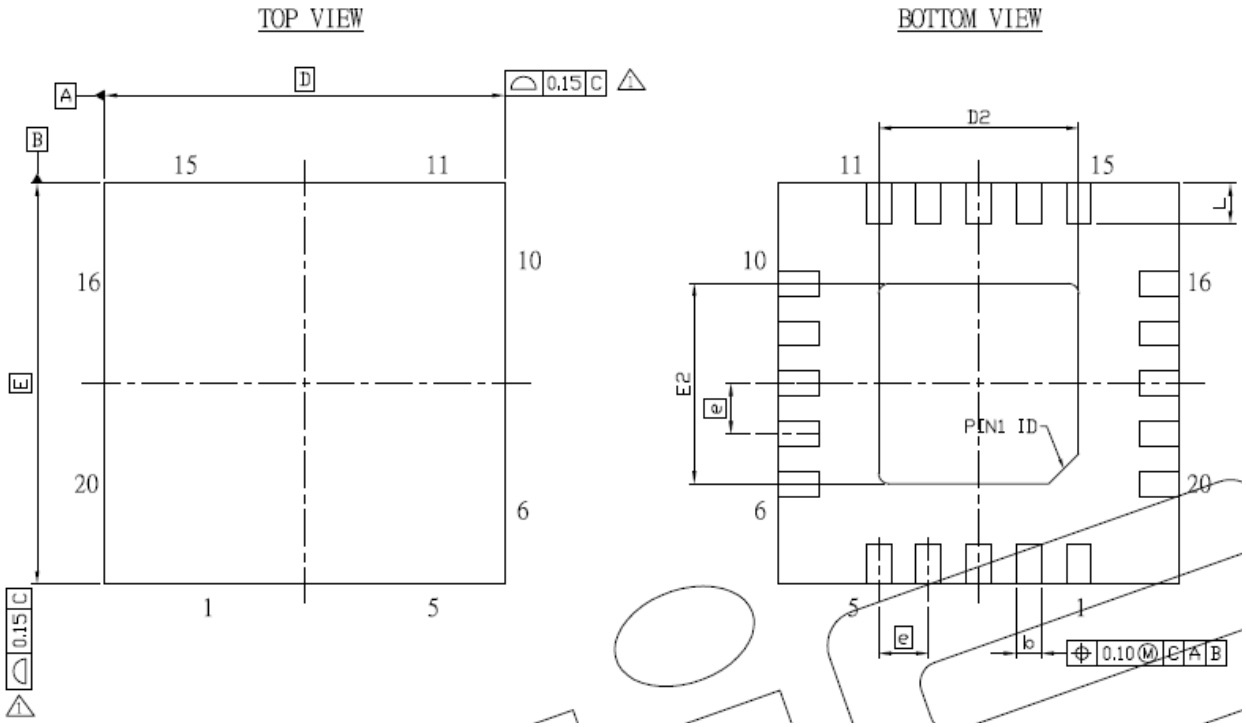
SYMBOLS	MIN	NOR	MAX	MIN	NOR	MAX
	(inch)			(mm)		
A	0.093	0.099	0.104	2.362	2.502	2.642
A1	0.004	0.008	0.012	0.102	0.203	0.305
D	0.447	0.455	0.463	11.354	11.557	11.760
E	0.291	0.295	0.299	7.391	7.493	7.595
H	0.394	0.407	0.419	10.008	10.325	10.643
L	0.016	0.033	0.050	0.406	0.838	1.270
θ°	0°	4°	8°	0°	4°	8°

SOP16: plastic small outline package; 16 leads; 150mil



SYMBOLS	DIMENSIONS IN MILLIMETERS			DIMENSIONS IN INCHES		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.35	1.60	1.75	0.053	0.063	0.069
A1	0.10	---	0.25	0.004	---	0.010
A2	---	1.45	---	---	0.057	---
B	0.33	---	0.51	0.013	---	0.020
C	0.19	---	0.25	0.007	---	0.010
D	9.80	---	10.00	0.386	---	0.394
E	3.80	---	4.00	0.150	---	0.157
e	---	1.27	---	---	0.050	---
H	5.80	---	6.20	0.228	---	0.244
L	0.40	---	1.27	0.016	---	0.050
y	---	---	0.10	---	---	0.004
theta	0°	---	8°	0°	---	8°

QFN20: 4 x 4 x 0.8mm package



SYMBOL	DIMENSION (MM)			DIMENSION (MIL)		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	0.70	0.75	0.80	27.56	29.53	31.50
A1	0	0.02	0.05	0	0.79	1.97
A3	0.203 REF			8 REF		
b	0.18	0.25	0.30	7.09	9.84	11.81
D	3.90	4.00	4.10	153.5	157.5	161.4
D2	1.90	2.00	2.10	74.8	78.7	82.7
E	3.90	4.00	4.10	153.5	157.5	161.4
E2	1.90	2.00	2.10	74.8	78.7	82.7
e	0.50 BSC			19.69 BSC		
L	0.30	0.40	0.50	11.81	15.74	19.69