SINO WEALTH

SH79F1620

Enhanced 8051 Microcontroller with 10bit ADC

1. Features

- 8bits micro-controller with Pipe-line structured 8051 compatible instruction set
- Flash ROM: 16K Bytes
- RAM: internal 256 Bytes, external 256 Bytes
- EEPROM-like: 1024 Bytes
- Operation Voltage:
  V_{DD} = 2V - 5.5V, f_{OSC} = 32.768kHz - 12MHz
- Oscillator (code option)
  - Crystal oscillator: 32.768kHz
  - Crystal oscillator: 2MHz - 12MHz
  - Ceramic oscillator: 2MHz - 12MHz
  - Internal RC: 12MHz (±2%)/128K
- 26/18 CMOS bi-directional I/O pins
- 8/6 Large-current drive port
- Built-in pull-up resistor for input pin
- Powerful interrupt sources:
  - Timer2, 3, 4, 5
  - INT2, 3
  - INT40, INT41, INT42, INT43
  - ADC, SCM , LPD, EUART
  - PWM
- EUART (No EUART in 20 PIN package)
- 8/6channels 10-bits Analog Digital Converter (ADC), with comparator function built-in
- Buzzer
- Four 16-bit timer/counters T2, T3, T4 and T5
- One 12-bit PWM
- Low Voltage Reset (LVR) function (enabled by code option)
  - LVR voltage level 1: 4.3V
  - LVR voltage level 2: 2.1V
- CPU Machine cycle: 1 oscillator clock
- Watch Dog Timer (WDT)
- Warm-up Timer
- Support Low power operation modes:
  - Idle Mode
  - Power-Down Mode
- Flash Type
- Package:
  - SOP28
  - SOP20/TSSOP20

2. General Description

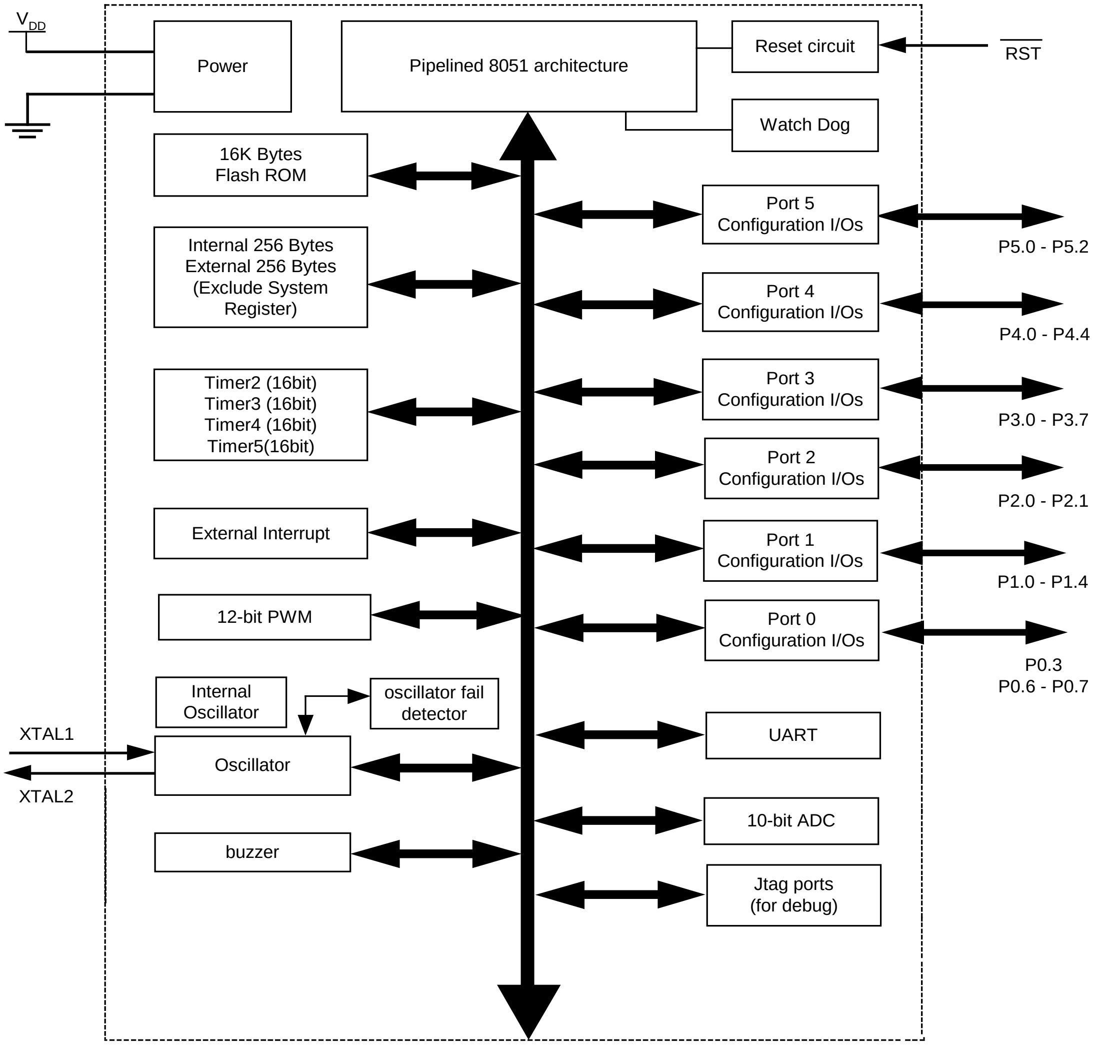
The SH79F1620 is a high performance 8051 compatible micro-controller, regard to its build-in Pipe-line instruction fetch structure, that helps the SH79F1620 can perform more fast operation speed and higher calculation performance, if compare SH79F1620 with standard 8051 at same clock speed.

The SH79F1620 retains most features of the standard 8051. These features include internal 256 bytes RAM, UART and INT2 and INT3. In addition, the SH79F1620 provides external 256 bytes RAM, Four 16-bit timer/counters T2-T5. It also contains 16K bytes Flash memory block both for program and data.

Also the ADC and PWM timer functions are incorporated in SH79F1620.

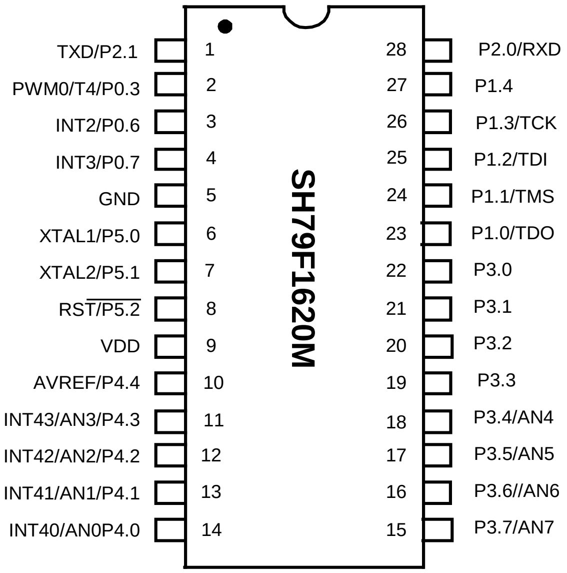
For high reliability and low power consumption, the SH79F1620 builds in Watchdog Timer, Low Voltage Reset function and SCM function. And SH79F1620 also supports two power saving modes to reduce power consumption.

3. Block Diagram



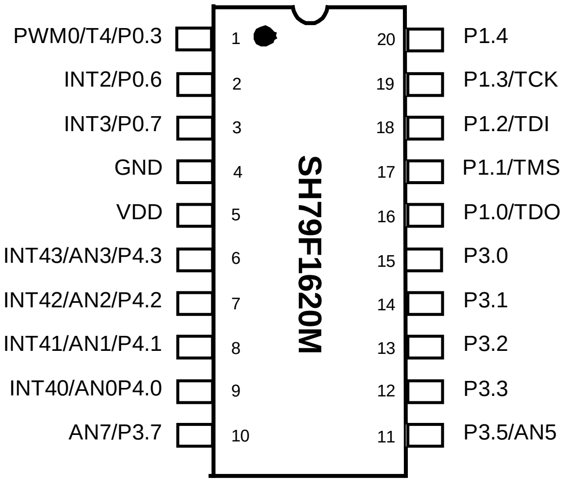
4. Pin Configuration

SOP28



Pin Configuration Diagram SOP28

SOP20/TSSOP20



Pin Configuration Diagram SOP20/TSSOP20

Note:

The out most pin function has the highest priority, and the inner most pin function has the lowest priority (Refer to Pin Configuration Diagram. This means when one pin is occupied by a higher priority function (if enabled) cannot be used as the lower priority functional pin, even when the lower priority function is also enabled. Until the higher priority function is closed by software, can the corresponding pin be released for the lower priority function use.

5. Pin Description

Pin No.	Type	Description
I/O PORT		
P0.3 P0.6 P0.7	I/O	3 bit General purpose CMOS I/O
P1.0 - P1.4	I/O	5 bit General purpose CMOS I/O
P2.0 P2.1	I/O	2 bit General purpose CMOS I/O (NoP2.0 and P2.1 in 20 PIN package)
P3.0 - P3.7	I/O	8 bit General purpose CMOS I/O (No P3.4 and P3.6 in 20 PIN package)
P4.0 - P4.4	I/O	5 bit General purpose CMOS I/O (No P4.4 in 20 PIN package)
P5.0 - P5.2	I/O	3 bit General purpose CMOS I/O (No P5.0-P5.2 in 20 PIN package)
Timer		
T4	I/O	Timer4 external input/output
PWM		
PWM0	O	Output pin for 12-bit PWM timer
EUART (No EUART in 20 PIN package)		
RXD	I	EUART0 data input
TXD	O	EUART0 data output
ADC		
AN0 - AN7	I	ADC input channel (No AN4 and AN6 in 20 PIN package)
AVREF	I	External ADC reference voltage input (No AVREF in 20 PIN package)
Interrupt & Reset & Clock & Power		
INT2 - INT3	I	External interrupt 2-3 input source
INT40 - INT43	I	External interrupt 40-43 input source
$\overline{\text{RST}}$	I	The device will be reset by A low voltage on this pin longer than 10us, an internal resistor about 30kΩ to V_{DD}, So using only an external capacitor to GND can cause a power-on reset. (No PIN_RESET function in 20 PIN package)
XTAL1	I	Oscillator input (No XTAL1 in 20 PIN package)
XTAL2	O	Oscillator output (No XTAL2 in 20 PIN package)
V_{SS}	P	Ground
V_{DD}	P	Power supply (2.0 - 5.5V)
Buzzer		
BUZ	O	Buzzer output pin
Programmer		
TDO (P1.0)	O	Debug interface: Test data out
TMS (P1.1)	I	Debug interface: Test mode select
TDI (P1.2)	I	Debug interface: Test data in
TCK (P1.3)	I	Debug interface: Test clock in
Note: *When P1.0-1.3 used as debug interface, functions of P1.0-1.3 are blocked.*		

6. SFR Mapping

The SH79F1620 provides 256 bytes of internal RAM to contain general-purpose data memory and Special Function Register (SFR). The SFR of the SH79F1620 fall into the following categories:

CPU Core Registers:	ACC, B, PSW, SP, DPL, DPH
Enhanced CPU Core Registers:	AUXC, DPL1, DPH1, INSCON, XPAGE
Power and Clock Control Registers:	PCON, SUSLO
Flash Registers:	IB_OFFSET, IB_DATA, IB_CON1, IB_CON2, IB_CON3, IB_CON4, IB_CON5, FLASHCON
Data Memory Register:	XPAGE
Hardware Watchdog Timer Registers:	RSTSTAT
System Clock Control Register:	CLKCON
Interrupt System Registers:	IEN0, IEN1, IENC, IPH0, IPL0, IPH1, IPL1, EXF0, EXF1
I/O Port Registers:	P0, P1, P2, P3, P4, P5, P0CR, P1CR, P2CR, P3CR, P4CR, P5CR, P0PCR, P1PCR, P2PCR, P3PCR, P4PCR, P5PCR
Timer Registers:	T2CON, T2MOD, TH2, TL2, RCAP2L, RCAP2H, T3CON, TH3, TL3, T4CON, TH4, TL4, SWTHL, T5CON, TH5, TL5
EUART Registers:	SCON, SBUF, SADEN, SADDR, PCON, RxCON
ADC Registers:	ADCON, ADT, ADCH, ADDL, ADDH
PWM Registers:	PWMEN, PWMEN1, PWMLO, PWM0C, PWM0PL, PWM0PH, PWM0DL, PWM0DH
LPD Registers:	LPDCON

Table 6.1 CPU Core SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ACC	E0H	Accumulator	00000000	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
B	F0H	B Register	00000000	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
AUXC	F1H	C Register	00000000	C.7	C.6	C.5	C.4	C.3	C.2	C.1	C.0
PSW	D0H	Program Status Word	00000000	CY	AC	F0	RS1	RS0	OV	F1	P
SP	81H	Stack Pointer	00000111	SP.7	SP.6	SP.5	SP.4	SP.3	SP.2	SP.1	SP.0
DPL	82H	Data Pointer Low byte	00000000	DPL0.7	DPL0.6	DPL0.5	DPL0.4	DPL0.3	DPL0.2	DPL0.1	DPL0.0
DPH	83H	Data Pointer High byte	00000000	DPH0.7	DPH0.6	DPH0.5	DPH0.4	DPH0.3	DPH0.2	DPH0.1	DPH0.0
DPL1	84H	Data Pointer 1 Low byte	00000000	DPL1.7	DPL1.6	DPL1.5	DPL1.4	DPL1.3	DPL1.2	DPL1.1	DPL1.0
DPH1	85H	Data Pointer 1 High byte	00000000	DPH1.7	DPH1.6	DPH1.5	DPH1.4	DPH1.3	DPH1.2	DPH1.1	DPH1.0
INSCON	86H	Data pointer select	-0--00-0	-	BKS0	-	-	DIV	MUL	-	DPS

Table 6.2 Power and Clock control SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCON	87H	Power Control	00--0000	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
SUSLO	8EH	Suspend Mode Control	00000000	SUSLO.7	SUSLO.6	SUSLO.5	SUSLO.4	SUSLO.3	SUSLO.2	SUSLO.1	SUSLO.0

Table 6.3 Flash control SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_OFF SET	FBH Bank0	Low byte offset of flash memory for programming	00000000	IB_OFF SET.7	IB_OFF SET.6	IB_OFF SET.5	IB_OFF SET.4	IB_OFF SET.3	IB_OFF SET.2	IB_OFF SET.1	IB_OFF SET.0
IB_DATA	FCH Bank0	Data Register for programming flash memory	00000000	IB_DATA.7	IB_DATA.6	IB_DATA.5	IB_DATA.4	IB_DATA.3	IB_DATA.2	IB_DATA.1	IB_DATA.0
IB_CON1	F2H Bank0	Flash Memory Control Register 1	00000000	IB_CON1.7	IB_CON1.6	IB_CON1.5	IB_CON1.4	IB_CON1.3	IB_CON1.2	IB_CON1.1	IB_CON1.0
IB_CON2	F3H Bank0	Flash Memory Control Register 2	----0000	-	-	-	-	IB_CON2.3	IB_CON2.2	IB_CON2.1	IB_CON2.0
IB_CON3	F4H Bank0	Flash Memory Control Register 3	----0000	-	-	-	-	IB_CON3.3	IB_CON3.2	IB_CON3.1	IB_CON3.0
IB_CON4	F5H Bank0	Flash Memory Control Register 4	----0000	-	-	-	-	IB_CON4.3	IB_CON4.2	IB_CON4.1	IB_CON4.0
IB_CON5	F6H Bank0	Flash Memory Control Register 5	----0000	-	-	-	-	IB_CON5.3	IB_CON5.2	IB_CON5.1	IB_CON5.0
XPAGE	F7H Bank0	Memory Page	--000000	-	-	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
FLASHCON	A7H Bank0	Flash access control	-------0	-	-	-	-	-	-	-	FAC

Table 6.4 WDT SFR

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RSTSTAT	B1H Bank0	Watchdog Timer Control	0-000000*	WDOF	-	PORF	LVRF	CLRF	WDT.2	WDT.1	WDT.0

***Note:** *RSTSTAT initial value is determined by different RESET.*

Table 6.5 CLKCON SFR

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CLKCON	B2H Bank0	System Clock Control Register	111000--	32k_SPDUP	CLKS1	CLKS0	SCMIF	HFON	FS	-	-

Table 6.6 Interrupt SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IEN0	A8H Bank0	Interrupt Enable Control 0	0000-000	EA	EADC	ET2	ES	-	EX1	ET5	EX0
IEN1	A9H Bank0	Interrupt Enable Control 1	0000000-	ESCM/ELPD	ET4	EPWM	ET3	EX4	EX3	EX2	-
IENC	BAH Bank0	Interrupt 4channel enable control	----0000	-	-	-	-	EXS43	EXS42	EXS41	EXS40
IENC1	BBH Bank0	Interrupt channel enable control 1	------00	-	-	-	-	-	-	ESCM	ELPD
IPH0	B4H Bank0	Interrupt Priority Control High 0	-000--0-	-	PADCH	PT2H	PSH	-	-	PT5H	-
IPL0	B8H Bank0	Interrupt Priority Control Low 0	-000--0-	-	PADCL	PT2L	PSL	-	-	PT5L	-
IPH1	B5H Bank0	Interrupt Priority Control High 1	0000000-	PSCMH	PT4H	PPWMH	PT3H	PX4H	PX3H	PX2H	-
IPL1	B9H Bank0	Interrupt Priority Control Low 1	0000000-	PSCML	PT4L	PPWML	PT3L	PX4L	PX3L	PX2L	-
EXF0	E8H Bank0	External interrupt Control 0	00000000	IT4.1	IT4.0	IT3.1	IT3.0	IT2.1	IT2.0	IE3	IE2
EXF1	D8H Bank0	External interrupt Control 1	----0000	-	-	-	-	IF43	IF42	IF41	IF40

Table 6.7 Port SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0	80H Bank0	3-bit Port 0	00—0---	P0.7	P0.6	-	-	P0.3	-	-	-
P1	90H Bank0	5-bit Port 1	---00000	-	-	-	P1.4	P1.3	P1.2	P1.1	P1.0
P2	A0H Bank0	2-bit Port 2	------00	-	-	-	-	-	-	P2.1	P2.0
P3	B0H Bank0	8-bit Port 3	00000000	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4	C0H Bank0	5-bit Port 4	---00000	-	-	-	P4.4	P4.3	P4.2	P4.1	P4.0
P5	80H Bank1	3-bit Port 5	-----000	-	-	-	-	-	P5.2	P5.1	P5.0
P0CR	E1H Bank0	Port0 input/output direction control	00—0---	P0CR.7	P0CR.6	-	-	P0CR.3	-	-	-
P1CR	E2H Bank0	Port1 input/output direction control	---00000	-	-	-	P1CR.4	P1CR.3	P1CR.2	P1CR.1	P1CR.0
P2CR	E3H Bank0	Port2 input/output direction control	------00	-	-	-	-	-	-	P2CR.1	P2CR.0
P3CR	E4H Bank0	Port3 input/output direction control	00000000	P3CR.7	P3CR.6	P3CR.5	P3CR.4	P3CR.3	P3CR.2	P3CR.1	P3CR.0
P4CR	E5H Bank0	Port4 input/output direction control	---00000	-	-	-	P4CR.4	P4CR.3	P4CR.2	P4CR.1	P4CR.0
P5CR	E1H Bank1	Port5 input/output direction control	-----000	-	-	-	-	-	P5CR.2	P5CR.1	P5CR.0
P0PCR	E9H Bank0	Internal pull-high enable for Port0	00—0---	P0PCR.7	P0PCR.6	-	-	P0PCR.3	-	-	-
P1PCR	EAH Bank0	Internal pull-high enable for Port1	---00000	-	-	-	P1PCR.4	P1PCR.3	P1PCR.2	P1PCR.1	P1PCR.0
P2PCR	EBH Bank0	Internal pull-high enable for Port2	------00	-	-	-	-	-	-	P2PCR.1	P2PCR.0
P3PCR	ECH Bank0	Internal pull-high enable for Port3	00000000	P3PCR.7	P3PCR.6	P3PCR.5	P3PCR.4	P3PCR.3	P3PCR.2	P3PCR.1	P3PCR.0
P4PCR	EDH Bank0	Internal pull-high enable for Port4	---00000	-	-	-	P4PCR.4	P4PCR.3	P4PCR.2	P4PCR.1	P4PCR.0
P5PCR	E9H Bank1	Internal pull-high enable for Port5	-----000	-	-	-	-	-	P5PCR.2	P5PCR.1	P5PCR.0

Table 6.8 Timer SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T2CON	C8H Bank0	Timer/Counter 2 Control	00000000	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
T2MOD	C9H Bank0	Timer/Counter 2 Mode	------00	-	-	-	-	-	-	T2OE	DCEN
RCAP2L	CAH Bank0	Timer/Counter 2 Reload /Capture Low Byte	00000000	RCAP2L.7	RCAP2L.6	RCAP2L.5	RCAP2L.4	RCAP2L.3	RCAP2L.2	RCAP2L.1	RCAP2L.0
RCAP2H	CBH Bank0	Timer/Counter 2 Reload /Capture High Byte	00000000	RCAP2H.7	RCAP2H.6	RCAP2H.5	RCAP2H.4	RCAP2H.3	RCAP2H.2	RCAP2H.1	RCAP2H.0
TL2	CCH Bank0	Timer/Counter 2 Low Byte	00000000	TL2.7	TL2.6	TL2.5	TL2.4	TL2.3	TL2.2	TL2.1	TL2.0
TH2	CDH Bank0	Timer/Counter 2 High Byte	00000000	TH2.7	TH2.6	TH2.5	TH2.4	TH2.3	TH2.2	TH2.1	TH2.0
T3CON	88H Bank1	Timer/Counter 3 Control	0-00-000	TF3	-	T3PS.1	T3PS.0	-	TR3	T3CLKS.1	T3CLKS.0
SWTHL	89H Bank1	Timer/Counter data switch	------00	-	-	-	-	-	-	T5HLCON	T3HLCON
TL3	8CH Bank1	Timer/Counter 3 Low Byte	00000000	TL3.7	TL3.6	TL3.5	TL3.4	TL3.3	TL3.2	TL3.1	TL3.0
TH3	8DH Bank1	Timer/Counter 3 High Byte	00000000	TH3.7	TH3.6	TH3.5	TH3.4	TH3.3	TH3.2	TH3.1	TH3.0
T4CON	C8H Bank1	Timer/Counter 4 Control	00000000	TF4	TC4	T4PS1	T4PS0	T4M1	T4M0	TR4	T4CLKS
TL4	CCH Bank1	Timer/Counter 4 Low Byte	00000000	TL4.7	TL4.6	TL4.5	TL4.4	TL4.3	TL4.2	TL4.1	TL4.0
TH4	CDH Bank1	Timer/Counter 4 High Byte	00000000	TH4.7	TH4.6	TH4.5	TH4.4	TH4.3	TH4.2	TH4.1	TH4.0
T5CON	C0H Bank1	Timer/Counter 5 Control	0-00--0-	TF5	-	T5PS1	T5PS0	-	-	TR5	-
TL5	CEH Bank1	Timer/Counter 5 Low Byte	00000000	TL5.7	TL5.6	TL5.5	TL5.4	TL5.3	TL5.2	TL5.1	TL5.0
TH5	CFH Bank1	Timer/Counter 5 High Byte	00000000	TH5.7	TH5.6	TH5.5	TH5.4	TH5.3	TH5.2	TH5.1	TH5.0

Table 6.9 EUART SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SCON	98H Bank0	Serial Control	00000000	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI
SBUF	99H Bank0	Serial Data Buffer	00000000	SBUF.7	SBUF.6	SBUF.5	SBUF.4	SBUF.3	SBUF.2	SBUF.1	SBUF.0
SADEN	9BH Bank0	Slave Address Mask	00000000	SADEN.7	SADEN.6	SADEN.5	SADEN.4	SADEN.3	SADEN.2	SADEN.1	SADEN.0
SADDR	9AH Bank0	Slave Address	00000000	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
PCON	87H Bank0	Power & serial Control	00--0000	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
RxCON	9FH Bank0	Rxd pin Schmidt voltage Control	------00	-	-	-	-	-	-	RxCON1	RxCON0

Table 6.10 ADC SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCON	93H Bank0	ADC Control	00000000	ADON	ADCIF	EC	REFC	SCH2	SCH1	SCH0	GO/$\overline{\text{DONE}}$
ADT	94H Bank0	ADC Time Configuration	000-0000	TADC2	TADC1	TADC0	-	TS3	TS2	TS1	TS0
ADCH	95H Bank0	ADC Channel Configuration	00000000	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
ADDL	96H Bank0	ADC Data Low Byte	------00	-	-	-	-	-	-	A1	A0
ADDH	97H Bank0	ADC Data High Byte	00000000	A9	A8	A7	A6	A5	A4	A3	A2

Table 6.11 PWM SFRs

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWMEN	CFH Bank0	PWM timer enable	-0--0--0	-	EFLT	-	-	EPWM01	-	-	EPWM0
PWMEN1	B7H Bank0	PWM output enable	-------0	-	-	-	-	-	-	-	PWM0
PWMLO	E7H Bank0	PWM register Lock	00000000	PWMLO.7	PWMLO.6	PWMLO.5	PWMLO.4	PWMLO.3	PWMLO.2	PWMLO.1	PWMLO.0
PWM0C	D2H Bank0	12-bit PWM Control	00---000	PWM0IE	PWM0IF	-	-	-	PWM0S	TnCK01	TnCK00
PWM0PL	D3H Bank0	12-bit PWM Period Control low byte	00000000	PP0.7	PP0.6	PP0.5	PP.4	PP0.3	PP0.2	PP0.1	PP0.0
PWM0PH	D4H Bank0	12-bit PWM Period Control high byte	----0000	-	-	-	-	PP0.11	PP0.10	PP0.9	PP0.8
PWM0DL	D5H Bank0	12-bit PWM Duty Control low byte	00000000	PD0.7	PD0.6	PD0.5	PD0.4	PD0.3	PD0.2	PD0.1	PD0.0
PWM0DH	D6H Bank0	12-bit PWM Duty Control high byte	----0000	-	-	-	-	PD0.11	PD0.10	PD0.9	PD0.8
PWM0DT	D1H Bank0	PWM01 Dead time control	00000000	DT0.7	DT0.6	DT0.5	DT0.4	DT0.3	DT0.2	DT0.1	DT0.0

Table 6.12 LPD SFR

Mnem	Add	Name	POR/WDT/LVR /PIN Reset Value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPDCON	B3H Bank0	LPD control	00000000	LPDEN	LPDF	LPDMD	LPDIF	LPDS3	LPDS2	LPDS1	LPDS0

***Note:** - :Unimplemented*

SFR Map

Bank0

	Bit addressable	**Non Bit addressable**							
	0/8	*1/9*	*2/A*	*3/B*	*4/C*	*5/D*	*6/E*	*7/F*	
F8H				IB_OFFSET	IB_DATA				FFH
F0H	B	AUXC	IB_CON1	IB_CON2	IB_CON3	IB_CON4	IB_CON5	XPAGE	F7H
E8H	EXF0	P0PCR	P1PCR	P2PCR	P3PCR	P4PCR			EFH
E0H	ACC	P0CR	P1CR	P2CR	P3CR	P4CR		PWMLO	E7H
D8H	EXF1								DFH
D0H	PSW		PWM0C	PWM0PL	PWM0PH	PWM0DL	PWM0DH		D7H
C8H	T2CON		RCAP2L	RCAP2H	TL2	TH2		PWMEN	CFH
C0H	P4								C7H
B8H	IPL0	IPL1	IENC	IENC1					BFH
B0H	P3	RSTSTAT	CLKCON	LPDCON	IPH0	IPH1		PWMEN1	B7H
A8H	IEN0	IEN1							AFH
A0H	P2							FLASHCON	A7H
98H	SCON	SBUF	SADDR	SADEN				RxCON	9FH
90H	P1			ADCON	ADT	ADCH	ADDL	ADDH	97H
88H							SUSLO		8FH
80H	P0	SP	DPL	DPH	DPL1	DPH1	INSCON	PCON	87H
	0/8	*1/9*	*2/A*	*3/B*	*4/C*	*5/D*	*6/E*	7/F	

Bank1

	Bit addressable	**Non Bit addressable**							
	0/8	*1/9*	*2/A*	*3/B*	*4/C*	*5/D*	*6/E*	*7/F*	
F8H									FFH
F0H	B	AUXC						XPAGE	F7H
E8H		P5PCR							EFH
E0H	ACC	P5CR							E7H
D8H									DFH
D0H	PSW								D7H
C8H	T4CON				TL4	TH4	TL5	TH5	CFH
C0H	T5CON								C7H
B8H	IPL0	IPL1							BFH
B0H					IPH0	IPH1			B7H
A8H	IEN0	IEN1							AFH
A0H									A7H
98H									9FH
90H									97H
88H	T3CON	SWTHL			TL3	TH3	SUSLO		8FH
80H	P5	SP	DPL	DPH	DPL1	DPH1	INSCON	PCON	87H
	0/8	*1/9*	*2/A*	*3/B*	*4/C*	*5/D*	*6/E*	7/F	

***Note:** The unused addresses of SFR are not available.*

7. Normal Function

7.1 CPU

7.1.1 CPU Core SFR

Feature

- CPU core registers: ACC, B, PSW, SP, DPL, DPH

Accumulator

ACC is the Accumulator register. Instruction system adopts A as mnemonic symbol of accumulator.

B Register

The B register is used during multiply and divide operations. For other instructions it can be treated as another scratch pad register.

Stack Pointer (SP)

The Stack Pointer Register is 8 bits special register, It is incremented before data is stored during PUSH, CALL executions and it is decremented after data is out of stack during POP, RET, RETI executions. The stack may reside anywhere in on-chip internal RAM (00H-FFH). On reset, the Stack Pointer is initialized to 07H causing the stack to begin at location 08H.

Program Status Word Register (PSW)

The PSW register contains program status information.

Data Pointer Register (DPTR)

DPTR consists of a high byte (DPH) and a low byte (DPL). Its intended function is to hold a 16-bit address, but it may be manipulated as a 16-bit register or as two independent 8-bit registers.

Table 7.1 PSW Register

D0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PSW	CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**CY**	**Carry flag bit** 0: no carry or borrow in an arithmetic or logic operation 1: a carry or borrow in an arithmetic or logic operation
6	**AC**	**Auxiliary Carry flag bit** 0: no auxiliary carry or borrow in an arithmetic or logic operation 1: an auxiliary carry or borrow in an arithmetic or logic operation
5	**F0**	**F0 flag bit** Available to the user for general purposes
4-3	**RS[1:0]**	**R0-R7 Register bank select bits** 00: Bank0 (Address to 00H-07H) 01: Bank1 (Address to 08H-0FH) 10: Bank2 (Address to 10H-17H) 11: Bank3 (Address to 18H-1FH)
2	**OV**	**Overflow flag bit** 0: no overflow happen 1: an overflow happen
1	**F1**	**F1 flag bit** Available to the user for general purposes
0	**P**	**Parity flag bit** 0: In the Accumulator,the bits whose value is 1 is even number 1: In the Accumulator,the bits whose value is 1 is odd number

7.1.2 Enhanced CPU core SFRs

- Extended 'MUL' and 'DIV' instructions: 16bit*8bit, 16bit/8bit
- Dual Data Pointer
- Enhanced CPU core registers: AUXC, DPL1, DPH1, INSCON

The SH79F1620 has modified 'MUL' and 'DIV' instructions. These instructions support 16 bit operand. A new register - the register AUXC is applied to hold the upper part of the operand/result.

The AUXC register is used during 16 bit operand multiply and divide operations. For other instructions it can be treated as another scratch pad register.

After reset, the CPU is in standard mode, which means that the 'MUL' and 'DIV' instructions are operating like the standard 8051 instructions. To enable the 16 bit mode operation, the corresponding enable bit in the INSCON register must be set.

	Operation		**Result**		
			A	**B**	**AUXC**
MUL	INSCON.2 = 0; 8 bit mode	(A)*(B)	Low Byte	High Byte	---
	INSCON.2 = 1; 16 bit mode	(AUXC A)*(B)	Low Byte	Middle Byte	High Byte
DIV	INSCON.3 = 0; 8 bit mode	(A)/(B)	Quotient Low Byte	Remainder	---
	INSCON.3 = 1; 16 bit mode	(AUXC A)/(B)	Quotient Low Byte	Remainder	Quotient High Byte

Dual Data Pointer

Using two data pointers can accelerate data memory moves. The standard data pointer is called DPTR and the new data pointer is called DPTR1.

DPTR1 is similar to DPTR, which consists of a high byte (DPH1) and a low byte (DPL1). Its intended function is to hold a 16-bit address, but it may be manipulated as a 16-bit register or as two independent 8-bit registers.

The DPS bit in INSTCON register is used to choose the active pointer by setting 1 or 0. And all DPTR-related instructions will use the currently selected data pointer.

7.1.3 Register

Table 7.2 Data Pointer Select Register

86H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INSCON	-	BKS0	-	-	DIV	MUL	-	DPS
R/W	-	R/W	-	-	R/W	R/W	-	R/W
Reset Value (POR/WDT/LVR/PIN)	-	0	-	-	0	0	-	0

Bit Number	Bit Mnemonic	Description
6	**BKS0**	**SFR Bank Selection Bit** 0: SFR Bank0 selected 1: SFR Bank1 selected
3	**DIV**	**16 bit/8 bit Divide Selection Bit** 0: 8 bit Divide 1: 16 bit Divide
2	**MUL**	**16 bit/8 bit Multiply Selection Bit** 0: 8 bit Multiply 1: 16 bit Multiply
0	**DPS**	**Data Pointer Selection Bit** 0: Data pointer 1: Data pointer1

7.2 RAM

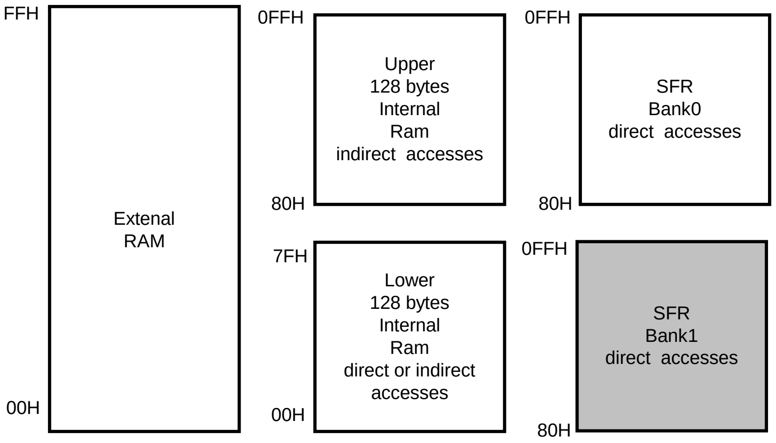
7.2.1 Features

SH79F1620 provides both internal RAM and external RAM for random data storage. The internal data memory is mapped into four separated segments:

- The Lower 128 bytes of RAM (addresses 00H to 7FH) are directly and indirectly addressable.
- The Upper 128 bytes of RAM (addresses 80H to FFH) are indirectly addressable only.
- The Special Function Registers (SFR, addresses 80H to FFH) are directly addressable only.
- The 256 bytes of external RAM(addresses 00H to FFH) are indirectly accessed by MOVX instructions.

The Upper 128 bytes occupy the same address space as SFR, but they are physically separate from SFR space. When an instruction accesses an internal location above address 7FH, the CPU can distinguish whether to access the upper 128 bytes data RAM or to access SFR by different addressing mode of the instruction.

SH79F1620 provides an extra 256 bytes of RAM to support high-level language in external data space.



The Internal and External RAM Configuration

The SH79F1620 provides traditional method for accessing of external RAM. Use *MOVXA, @Ri* or *MOVX@Ri, A;* to access external low 256 bytes RAM; *MOVX A, @DPTR* or *MOVX @DPTR, A* also to access external 256 bytes RAM.

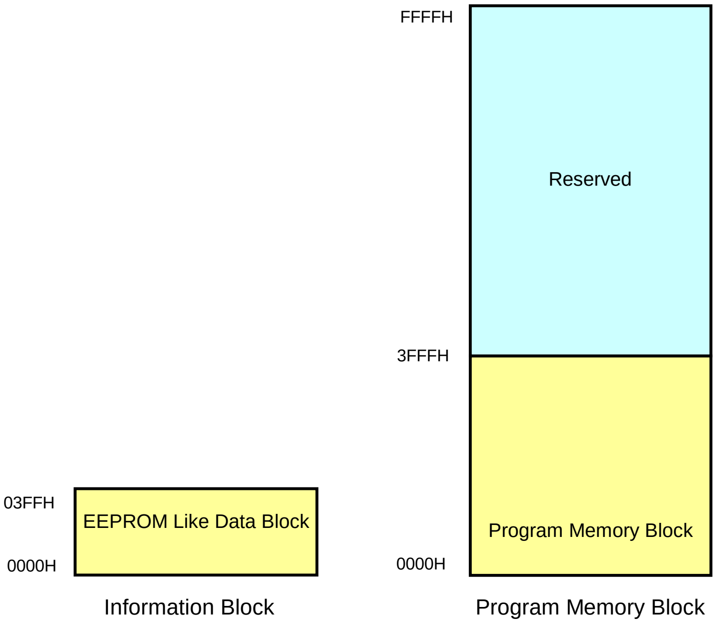
Note:

In SH79F1620 the user can also use XPAGE register to access external RAM only with MOVX A, @Ri or MOVX@Ri, A instructions. The user can use XPAGE to represent the high byte address of RAM above 256 Bytes.

7.3 Flash Program Memory

7.3.1 Features

- The program memory consists 16 X 1KB sectors, total 16KB
- Programming and erase can be done over the full operation voltage range
- Write, read and erase operation are all supported by In-Circuit Programming (ICP)
- Support overall/sector erase and programming, each sector erase time < 3ms, programming time per byte < 30us
- Minimum program/erase cycles: Program area: 1000
  EEPROM-like area: 100,000
- Minimum years data retention: 10
- Low power consumption



The SH79F1620 embeds 16K flash program memory for program code. The flash program memory provides electrical erasure and programming and supports In-Circuit Programming (ICP) mode and Self-Sector Programming (SSP) mode. Every sector is 1024 bytes.

The SH79F1620 also embeds 1024 bytes EEPROM-likea memory block for storing user data.Every sector is 256 bytes.It has 4 sectors.

Flash operation defined:

In-Circuit Programming (ICP):Through the Flash programmer to wipe the Flash memory, read and write operations.

Self-Sector Programming (SSP) mode:User Program code run in Program Memory to wipe the Flash memory, read and write operations.

Flash Memory Supports the Following Operations:

(1) Code Protection Control Mode

SH79F1620 code protection function provides a high-performance security measures for the user. Each partition has two modes are available.
Code protection mode 0: allow/forbid any programmer write/read operations (not including overall erasure).
Code protection mode 1: allow/forbid through MOVC instructions to read operation in other sectors, or through SSP mode to erased/write operation.
The user must use one of the following two ways to complete code protection control mode Settings:
Flash programmer in ICP mode is set to corresponding protection bit to enter the protected mode.
The SSP mode does not support code protection control mode programming.

(2) Overall Erasure

Regardless of the state of the code protection control mode, the overall erasure operation will erase all programs, code options, the code protection bit, but they will not erase EEPROM-like memory block.
The user must use the following way to complete the overall erasure:
Flash programmer in ICP mode send overall erasure instruction to run overall erasure.
The SSP mode does not support overall erasure mode.

(3) Sector Erasure

Sector erasure operations will erase the content of selected sector. The user program (SSP) and Flash programmer can perform this operation.
For user programs to perform the operation, code protection mode 1 in the selected sector must be forbidden.
For Flash programmer to perform the operation,code protection mode 0 in the selected sector must be forbidden.
The user must use one of the following two ways to complete sector erasure:

1. Flash programmer in ICP mode send sector erasure instruction to run sector erasure.
2. Through the SSP function send sector erasure instruction to run sector erasure (see chapter SSP).

(4) EEPROM-like Memory Block Erasure

EEPROM-like memory block erasure operations will erase the content in EEPROM-like memory block.The user program (SSP) and Flash programmer can perform this operation.
The user must use one of the following two ways to complete EEPROM-like memory block erasure:

1. Flash programmer in ICP mode send EEPROM-like memory block erasure instruction to run EEPROM-like memory block erasure.
2. Through the SSP function send EEPROM-like memory block erasure instruction to run EEPROM-like memory block erasure (see chapter SSP).

(5) Write/Read Code

Write/read code operation can read or write code from flash memory block. The user program (SSP) and Flash programmer can perform this operation.
For user programs to perform the operation, code protection mode 1 in the selected sector must be forbidden. Regardless of the security bit Settings or not, the user program can read/write the sector which contains program itself.
For Flash programmer to perform the operation, code protection mode 0 in the selected sector must be forbidden.
The user must use one of the following two ways to complete write/read code:

1. Flash programmer in ICP mode send write/read code instruction to run write/read code.
2. Through the SSP function send write/read code instruction to run write/read code.

(6) Write/Read EEPROM-like Memory Block

EEPROM-like memory block operation can read or write data from EEPROM-like memory block. The user program (SSP) and Flash programmer can perform this operation.
The user must use one of the following two ways to complete write/read EEPROM-like memory block:

1. Flash programmer in ICP mode send write/read EEPROM-like memory block instruction to run write/read EEPROM-like memory block.
2. Through the SSP function send write/read EEPROM-like memory block instruction to run write/read EEPROM-like memory block.

Flash Memory Block Operation Summary

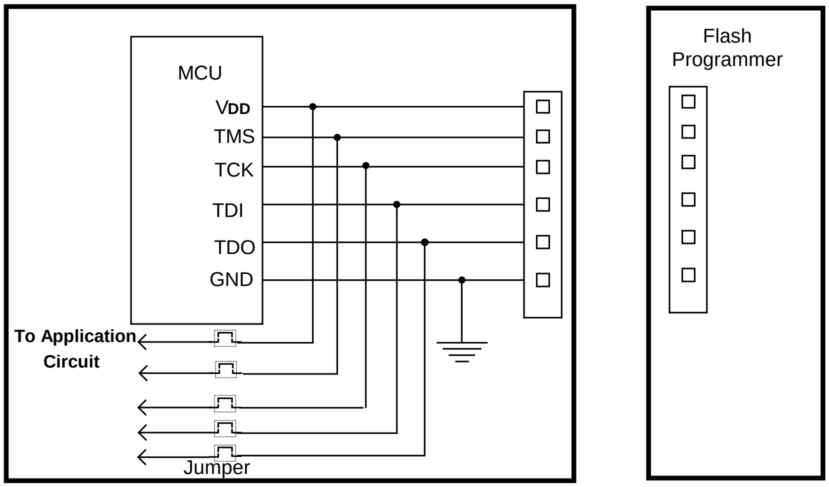
Operation	ICP	SSP
Code protection	support	non support
Sector erasure	support (no security bit)	support (no security bit)
Overall erasure	support	non support
EEPROM-like memory block erasure	support	support
Write/read code	Support (no security bit)	support (no security bit)
Read/write EEPROM-like memory block	support	support

7.3.2 Flash Operation in ICP Mode

ICP mode is performed without removing the micro-controller from the system. In ICP mode, the user system must be power-off, and the programmer can refresh the program memory through ICP programming interface. The ICP programming interface consists of 6 pins (V_{DD}, GND, TCK, TDI, TMS, TDO).

At first the four JTAG pins (TDO, TDI, TCK, TMS) are used to enter the programming mode. Only after the four pins are inputted the specified waveform, the CPU will enter the programming mode. For more detail description please refers to the FLASH Programmer's user guide.

In ICP mode，all the flash operations are completed by the programmer through 6-wire interface. Since the program signal is very sensitive, 6 jumpers are needed (V_{DD}, GND, TDO, TDI, TCK, TMS) to separate the program pins from the application circuit , as show in the following diagram.



The recommended steps are as following:

(1) The jumpers must be open to separate the programming pins from the application circuit before programming.
(2) Connect the programming interface with programmer and begin programming.
(3) Disconnect programmer interface and connect jumpers to recover application circuit after programming is complete.

7.4 SSP Function

The SH79F1620 provides SSP (Self Sector Programming) function, each sector can be sector erased or programmed by the user's code if the selected sector is not be protected. But once sector has been programmed, it cannot be reprogrammed before sector erase.

The SH79F1620 builds in a complex control flow to prevent the code from carelessly modification. If the dedicated conditions are not met (IB_CON2-5), the SSP will be terminated.

7.4.1 SSP Register

Table 7.3 Memory Page Register for Programming (For Flash memory, one sector is 1024 bytes)

F7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
XPAGE	-	-	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
5-2	**XPAGE[5:2]**	Sector of the flash memory to be programmed, 0000 means sector 0, and so on
1-0	**XPAGE[1:0]**	High 2 Address of the flash memory sector to be programmed

Table 7.4 Memory Page Register for Programming (For EEPROM-like memory, one sector is 256 bytes)

F7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
XPAGE	-	-	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
5-2	**XPAGE[5:2]**	Reserved
1-0	**XPAGE[1:0]**	Sector of the flash memory to be programmed, 00 means sector 0, and so on

Table 7.5 Offset of Flash Memory for Programming

FBH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_OFFSET	IB_OFFSET.7	IB_OFFSET.6	IB_OFFSET.5	IB_OFFSET.4	IB_OFFSET.3	IB_OFFSET.2	IB_OFFSET.1	IB_OFFSET.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**IB_OFFSET[7:0]**	Low 8 Address of Offset of the flash memory sector to be programmed

Table 7.6 Data Register for Programming

FCH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_DATA	IB_DATA.7	IB_DATA.6	IB_DATA.5	IB_DATA.4	IB_DATA.3	IB_DATA.2	IB_DATA.1	IB_DATA.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	IB_DATA[7:0]	Data to be programmed

Table 7.7 SSP Type select Register

F2H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_CON1	IB_CON1.7	IB_CON1.6	IB_CON1.5	IB_CON1.4	IB_CON1.3	IB_CON1.2	IB_CON1.1	IB_CON1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	IB_CON1[7:0]	**SSP Type select** 0xE6: Sector Erase(erase time < 3ms) 0x6E: Sector Programming(program time <30us)

Table 7.8 SSP Flow Control Register1

F3H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_CON2	-	-	-	-	IB_CON2.3	IB_CON2.2	IB_CON2.1	IB_CON2.0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	IB_CON2[3:0]	Must be 05H, otherwise Flash Programming will terminate

Table 7.9 SSP Flow Control Register2

F4H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_CON3	-	-	-	-	IB_CON3.3	IB_CON3.2	IB_CON3.1	IB_CON3.0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	IB_CON3[3:0]	Must be 0AH,otherwise Flash Programming will terminate

Table 7.10 SSP Flow Control Register3

F5H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_CON4	-	-	-	-	IB_CON4.3	IB_CON4.2	IB_CON4.1	IB_CON4.0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

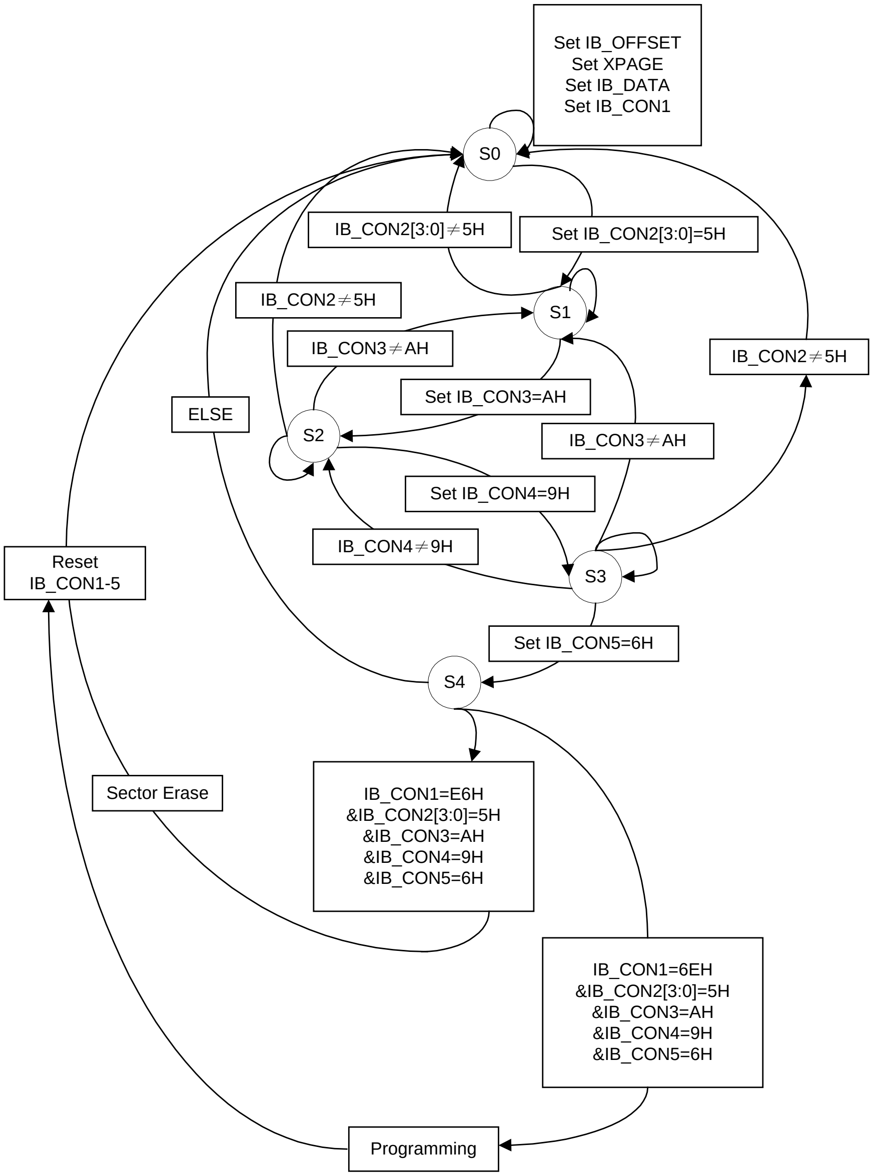
Bit Number	Bit Mnemonic	Description
3-0	**IB_CON4[3:0]**	Must be 09H, otherwise Flash Programming will terminate

Table 7.11 SSP Flow Control Register4

F6H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IB_CON5	-	-	-	-	IB_CON5.3	IB_CON5.2	IB_CON5.1	IB_CON5.0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	**IB_CON5[3:0]**	Must be 06H, otherwise Flash Programming will terminate

7.4.2 Flash Control Flow



7.4.3 SSP Programming Notice

To successfully complete SSP programming, the user’s software must be set as the following the steps:

(1) For Code/Data Programming:

1. Disable interrupt;
2. Fill in the XPAGE, IB_OFFSET for the corresponding address;
3. Fill in IB_DATA if programming is wanted;
4. Fill in IB_CON1-5 sequentially;
5. Add 4 nops for more stable operation;
6. Code/Data programming, CPU will be in IDLE mode;
7. Go to Step 2 if more data are to be programmed;
8. Clear XPAGE; enable interrupt if necessary.

(2) For Sector Erase:

1. Disable interrupt;
2. Fill in the XPAGE for the corresponding sector;
3. Fill in IB_CON1-5 sequentially;
4. Add 4 NOPs for more stable operation;
5. Sector Erase, CPU will be in IDLE mode;
6. Go to step 2 if more sectors are to be erased;
7. Clear XPAGE; enable interrupt if necessary.

(3) For Code Reading:

Just Use “MOVC A, @A+DPTR” or “MOVC A, @A+PC”.

(4) For EEPROM-Like:

Steps is same as code programming,the diffenrences are:
1. Set FAC bit in FLASHCON register before programming or erase EEPROM-Like;
2. One sector of EEPROM-Like is 256 bytes, not 1024 bytes.

Note:

(1) The system clock is not less than 200 KHZ to ensure normal FLASH programming
(2) FAC must be cleared when you don't need to do EEPROM-like operation.

FLASHCON register description is as follows:

Table 7.12 Flash Access Control Register

A7H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
FLASHCON	-	-	-	-	-	-	-	FAC
R/W	-	-	-	-	-	-	-	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	-	0

Bit Number	Bit Mnemonic	Description
7-1	**-**	**Reserved**
0	**FAC**	**FAC: Flash access control** 0: MOVC or SSP access main memory 1: MOVC or SSP access EEPROM-like

7.5 System Clock and Oscillator

7.5.1 Features

- Four oscillator types: 32.768kHz crystal oscillator, crystal oscillator, ceramic oscillator and 12MHz/128kHz internal RC
- 2 Oscillator pin (XTAL1, XTAL2)
- Built-in 12MHz Internal RC
- Built-in 32.768kHz speed up circuit
- Built-in system clock prescaler

Note: *20 PIN package has no external Oscillator pin (XTAL1, XTAL2), only supports two oscillator types: 12MHz and 128kHz internal RC.*

7.5.2 Clock Definition

The SH79F1620 have several internal clocks defined as below:

OSCCLK: the oscillator clock is selected from the four oscillator types (32.768kHz crystal oscillator, crystal oscillator, ceramic oscillator and interal 12M RC/128K oscillator from XTAL input). f_{OSC} is defined as the OSCCLK frequency. t_{OSC} is defined as the OSCCLK period.

OSCXCLK: select internal 12M RC oscillator. f_{OSCX} is defined as the OSCXCLK frequency. t_{OSCX} is defined as the OSCXCLK period.

Note: *OSCXCLK does not exist when code option OP_OSC is not 0011, 0110. (32.768kHz/128kHz RC is not selected, Refer to* ***code option*** *section for details)*

WDTCLK: the internal WDT RC clock. f_{WDT} is defined as the WDTCLK frequency. t_{WDT} is defined as the WDTCLK period.

OSCSCLK: the input clock of system clock frequency prescaler. It can be OSCCLK or OSCXCLK. f_{OSCS} is defined as the OSCSCLK frequency. t_{OSCS} is defined as the OSCSCLK period.

SYSCLK: system clock, the output clock of system clock prescaler. It is the CPU instruction clock. f_{SYS} is defined as the SYSCLK frequency. t_{SYS} is defined as the SYSCLK period.

7.5.3 Description

SH79F1620 has four oscillator types: 32.768kHz crystal oscillator, crystal oscillator/ceramic oscillator (2MHz-12MHz) and internal RC (12MHz, 128KHz), which is selected by code option OP_OSC (Refer to code option section for details). SH79F1620 has 2 oscillator pins (XTAL1, XTAL2), which can generate 1 or 2 clocks from 4 oscillator types. They also are selected by code option OP_OSC (Refer to code option section for details). The oscillator generates the basic clock pulse that provides the system clock to supply CPU and on-chip peripherals.

7.5.4 Register

Table 7.13 System Clock Control Register

B2H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CLKCON	32K_SPDUP	CLKS1	CLKS0	SCMIF	HFON	FS	-	-
R/W	R/W	R/W	R/W	R	R/W	R/W	-	-
Reset Value (POR/WDT/LVR/PIN)	1	1	1	0	0	0	-	-

Bit Number	Bit Mnemonic	Description
7	**32K_SPDUP**	**32.768kHz oscillator speed up mode control bit** 0: 32.768kHz oscillator normal mode, cleared by software. 1: 32.768kHz oscillator speed up mode, set by hardware or software. This control bit is set by hardware automatically in all kinds of RESET such as Power on reset, watch dog reset etc. to speed up the 32.768kHz Oscillator oscillating, shorten the 32.768kHz oscillator start-oscillating time. And this bit also can be set or cleared by software if necessary. Such as set before entering Power-down mode and cleared when Power-down mode wakes up. It should be noticed that turning off 32.768kHz oscillator speed up (clear this bit) could reduce the system power consumption. Only when code option OP_OSC is 1010 or 1101, this bit is valid. (32.768kHz oscillator is selected, Refer to **code option** section for details) (The control bit is invalid in 20 pins package).
6-5	**CLKS [1: 0]**	**SYSCLK Prescaler Register** 00: $f_{SYS} = f_{OSCS}$ 01: $f_{SYS} = f_{OSCS}/2$ 10: $f_{SYS} = f_{OSCS}/4$ 11: $f_{SYS} = f_{OSCS}/12$ If 32.768kHz oscillator is selected as OSCSCLK, these control bits is invalid.
3	**HFON**	**OSCXCLK On-Off control Register** 0: turn off OSCXCLK 1: turn on OSCXCLK Only when code option OP_OSC is 0011, 0110, 1010, this bit is valid. (32.768kHz oscillator/128kHz internal RC is selected, refer to **code option** section for details)
2	**FS**	**Frequency Select Register** 0: 32.768kHz/128kHz is selected as OSCSCLK. 1: OSCXCLK is selected as OSCSCLK. Only when code option OP_OSC is 0011, 0110, 1010. this bit is valid. (32.768kHz oscillator/128kHz internal RC is selected, refer to **code option** section)

Note:

(1) If code option OP_OSC is 0011,0110,1010, OSCXCLK is Internal 12M RC.

(2) HFON and FS is valid only when code option OP_OSC is 0011, 0110, 1010.

(3) When OSCXCLK is used as OSCSCLK (that is HFON = 1 and FS = 1), HFON is can't be cleared by software.

(4) When OSCSCLK changed from 32.768kHz/128kHz to OSCXCLK, if OSCXCLK is off, the setting must be done as the following steps:

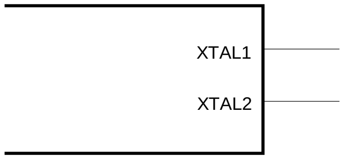
a. Set HFON = 1 to turn on the OSCXCLK
b. Wait at least Oscillator Warm-up timer (Refer to Warm-up Timer section for details)
c. Set FS = 1 to select OSCXCLK as OSCSCLK.

(5). When OSCSCLK changed from OSCXCLK to 32.768kHz/128kHz, the setting must be done as the following steps:

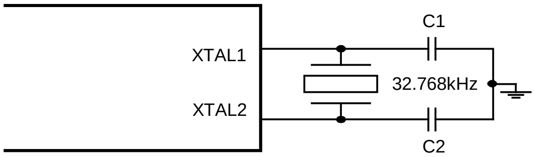
a. Clear FS to select 32.768kHZ/128k as OSCSCLK.
b. Add one NOP
c. Clear HFON(reduce power consumption)
d. Add four NOPs.

7.5.5 Oscillator Type

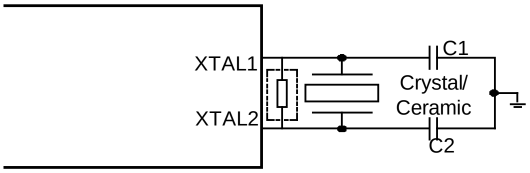
(1) OP_OSC = 0000, 0011: internal RC, XTAL are shared with IO



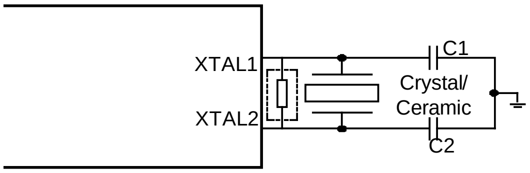
(2) OP_OSC = 1010: 32.768kHz Crystal Oscillator at XTAL, Internal RC can be enabled



(3) OP_OSC = 1110: 2M - 12M Crystal/Ceramic oscillator from XTAL input*



(4) OP_OSC = 0110: 128kHz internal RC, 2M - 12M Crystal/Ceramic oscillator from XTAL input*



*: *If the environment humidity is bigger, use the high frequency oscillator, advice plus 510k feedback resistance.*

Note: *20 PIN package have no these oscillator types: above-mentioned (2), (3) and (4).*

7.5.6 Capacitor Selection for Oscillator

Ceramic Oscillator		
Frequency	**C1**	**C2**
3.58MHz	-	-
4MHz	-	-

Crystal Oscillator		
Frequency	**C1**	**C2**
32.768kHz	10 - 12pF	10 - 12pF
4MHz	8 - 15pF	8 - 15pF
12MHz	8 - 15pF	8 - 15pF

Note :

(1) Capacitor values are used for design guidance only!
(2) These capacitors were tested with the crystals listed above for basic start-up and operation. They are ***not optimized.***
(3) Be careful for the stray capacitance on PCB board, the user should test the performance of the oscillator over the expected V_{DD} and the temperature range for the application.
(4) Before selecting crystal/ceramic, the user should consult the crystal/ceramic manufacturer for appropriate value of external component to get best performance, visit http://www.sinowealth.com for more recommended manufactures.

7.6 System Clock Monitor (SCM)

In order to enhance the system reliability, SH79F1618 contains a system clock monitor (SCM) module. If the system clock breaks down(for example the external oscillator stops oscillating), the built-in SCM will switch the OSCCLK to the internal 32k WDTCLK and set system clock monitor bit (SCMIF) to 1. And the SCM interrupt will be generated when EA and ESCM is enabled. If the external oscillator comes back, SCM will switch the OSCCLK back to the external oscillator and clears the SCMIF automatically.

Notes:

The SCMIF is read only register; it can be clear to 0 or set to 1 by hardware only.

If SCMIF is cleared, the SCM switches the system clock to the state before system clock breaks down automatically.

If Internal RC is selected as OSCSCLK by code option (Refer to ***code option*** *section for detail), the SCM can not work.*

Table 7.14 System Clock Control Register

B2H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CLKCON	-	-	-	SCMIF	-	-	-	-
R/W	-	-	-	R	-	-	-	-
Reset Value (POR/WDT/LVR/PIN)	-	-	-	0	-	-	-	-

Bit Number	Bit Mnemonic	Description
4	**SCMIF**	**System Clock Monitor flag bit** 0: Clear by hardware to indicate system clock is normal 1: Set by hardware to indicate system clock fails

7.7 I/O Port

7.7.1 Features

- 26/18 bi-directional I/O ports
- Share with alternative functions

The SH79F1620 has 26 bi-directional I/O ports. The PORT data is put in Px register. The PORT control register (PxCRy) controls the PORT as input or output. Each I/O port has an internal pull-high resistor, which is controlled by PxPCRy when the PORT is used as input (x = 0-5, y = 0-7).

For SH79F1620, some I/O pins can share with alternative functions. There exists a priority rule in CPU to avoid these functions conflicts when all the functions are enabled. (Refer to **Port Share** Section for details).

7.7.2 Register

Table 7.15 Port Control Register

E1H - E5H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0CR (E1H, Bank0)	P0CR.7	P0CR.6	-	-	P0CR.3	-	-	-
P1CR (E2H, Bank0)	-	-	-	P1CR.4	P1CR.3	P1CR.2	P1CR.1	P1CR.0
P2CR (E3H, Bank0)	-	-	-	-	-	-	P2CR.1	P2CR.0
P3CR (E4H, Bank0)	P3CR.7	P3CR.6	P3CR.5	P3CR.4	P3CR.3	P3CR.2	P3CR.1	P3CR.0
P4CR (E5H, Bank0)	-	-	-	P4CR.4	P4CR.3	P4CR.2	P4CR.1	P4CR.0
P5CR (E1H, Bank1)	-	-	-	-	-	P5CR.2	P5CR.1	P5CR.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**PxCRy** **x = 0-5, y = 0-7**	**Port input/output control Register** 0: input mode 1: output mode

Table 7.16 Port Pull up Resistor Control Register

E9H - ECH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0PCR (E9H, Bank0)	P0PCR.7	P0PCR.6	-	-	P0PCR.3	-	-	-
P1PCR (EAH, Bank0)	-	-	-	P1PCR.4	P1PCR.3	P1PCR.2	P1PCR.1	P1PCR.0
P2PCR (EBH, Bank0)	-	-	-	-	-	-	P2PCR.1	P2PCR.0
P3PCR (ECH, Bank0)	P3PCR.7	P3PCR.6	P3PCR.5	P3PCR.4	P3PCR.3	P3PCR.2	P3PCR.1	P3PCR.0
P4PCR (EDH, Bank0)	-	-	-	P4PCR.4	P4PCR.3	P4PCR.2	P4PCR.1	P4PCR.0
P5PCR (E9H, Bank1)	**-**	**-**	**-**	-	-	P5PCR.2	P5PCR.1	P5PCR.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

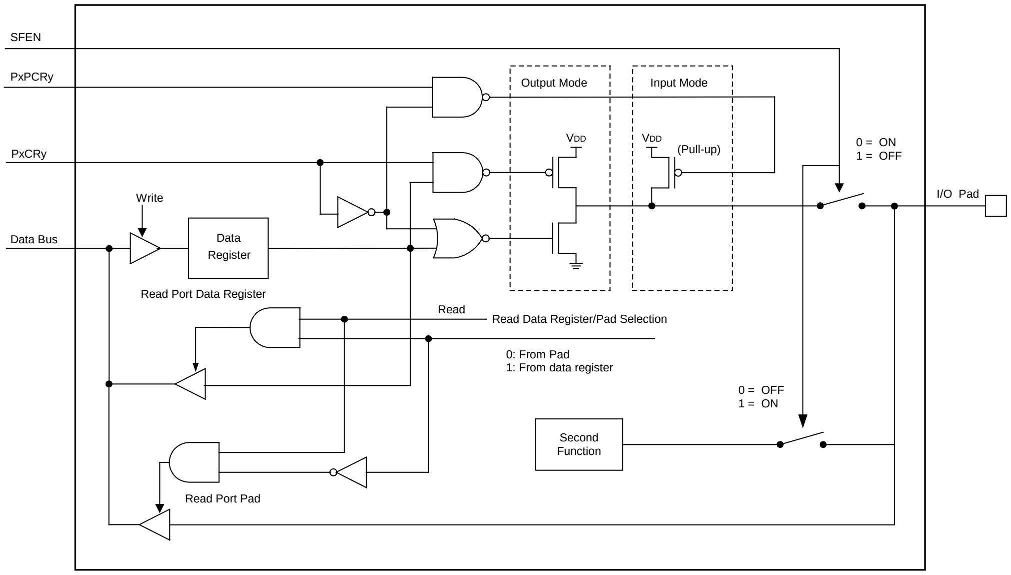
Bit Number	Bit Mnemonic	Description
7-0	**PxPCRy** **x = 0-5, y = 0-7**	**Input Port internal pull-high resistor enable/disable control** 0: internal pull-high resistor disabled 1: internal pull-high resistor enabled

Table 7.17 Port Data Register

80H - C0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0 (80H, Bank0)	P0.7	P0.6	-	-	P0.3	-	-	-
P1 (90H, Bank0)	-	-	-	P1.4	P1.3	P1.2	P1.1	P1.0
P2 (A0H, Bank0)	-	-	-	-	-	-	P2.1	P2.0
P3 (B0H, Bank0)	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4 (C0H, Bank0)	-	-	-	P4.4	P4.3	P4.2	P4.1	P4.0
P5 (80H, Bank1)	-	-	-	-	-	P5.2	P5.1	P5.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**Px.y x = 0-5, y = 0-7**	Port Data Register

7.7.3 Port Diagram



Note:

(1) The input source of reading input port operation is from the input pin directly.

(2) The input source of reading output port operation has two paths, one is from the port data Register, and the other is from the output pin directly.

(3) The read Instruction distinguishes which path is selected: The read-modify-write instruction is for the reading of the data register in output mode, and the other instructions are for reading of the output pin directly.

(4) The destination of writing port operation is the data register regardless of the port shared as the second function or not.

7.7.4 Port Share

The 26/18 bi-directional I/O ports can also share second or third special function. But the share priority should obey the **Outer Most Inner Lest** rule:

The out most pin function in **Pin Configuration** has the highest priority, and the inner most pin function has the lowest priority. This means when one pin is occupied by a higher priority function (if enabled), it cannot be used as the lower priority functional pin, even the lower priority function is also enabled. Only until the higher priority function is closed by hardware or software,can the corresponding pin be released for the lower priority function use. Also the function that need pull up resister is also controlled by the same rule.

When port share function is enabled, the user can modify PxCR, PxPCR (x = 0-5), but these operations will have no effect on the port status until the second function was disabled.

When port share function is enabled, any read or write operation to port will only affect the data register.The value of the port pin kepps unchanged until the second function was disabled.

PORT0:

- PWM0: PWM0 output (P0.3)
- INT2: external inturrupt2 (P0.6)
- INT3: external inturrupt3 (P0.7)
- T4: Timer4 external input/Baud-rate clock output (P0.3)

Table 7.18 PORT0 Share Table

Pin No.		Priority	Function	Enable bit
SOP28	TSSOP/ SOP20			
2	1	1	PWM0	Set **EPWM0** bit in **PWMEN** register
		2	T4	Set TR4&T4CLKS in T4CON to 1(pull-high automatically) ,or set T4CLK in T4CON to 0 and TC4 bit is set as 1 or TR4 is set as 1 in mode 2
		3	P0.3	Above condition is not met
3	2	1	INT2	Set **EX2** bit in **IEN1** Register and Port0.6 is in input mode
		2	P0.6	Above condition is not met
4	3	1	INT3	Set **EX3** bit in **IEN1** Register and Port0.7 is in input mode
		2	P0.7	Above condition is not met

PORT1:

Table 7.19 PORT1 Share Table

Pin No.		Priority	Function	Enable bit
SOP28	TSSOP/ SOP20			
23-27	16-20	1	P1.0-P1.4	Default

PORT2:

- RXD: EUART data input (P2.0)
- TXD: EUART data output (P2.1)

Table 7.20 PORT2 Share Table

Pin No.		Priority	Function	Enable bit
SOP28	TSSOP/ SOP20			
28	-	1	RXD	Set REN bit in **SCON** Register (Auto Pull up)
		2	P2.0	Above condition is not met
1	-	1	TXD	Do write operation to SBUF register
		2	P2.1	Above condition is not met

PORT3:

- AN4-AN7: ADC input channel (P3.4-P3.7)

Table 7.21 PORT3 Share Table

Pin No.		Priority	Function	Enable bit
SOP28	TSSOP/ SOP20			
15-18	**10-11**	1	AN7-AN4	Set **CH7-4** bit in **ADCH** Register and set **ADON** bit in **ADCON** Register, and set **SCH [2:0]** (20 pins package have no AN4&AN6 channels)
		2	P3.7-P3.4	Above condition is not met
19-22	12-15	1	P3.3-P3.0	Default

PORT4:

- INT40-INT43 (P4.0-P4.3): External interrupt input
- AN0-AN3 (P4.0-P4.3): ADC input channel
- AVREF (P4.4): AD reference voltage

Table 7.22 PORT4 Share Table

Pin No.		Priority	Function	Enable bit
SOP28	TSSOP/ SOP20			
14	-	1	AVREF	Set **REFC** bit in **ADCON** register
		2	P4.4	Above condition is not met
15-18	6-9	1	AN3-AN0	Set **CH3-0** bit in **ADCH** Register and set **SCH [2:0]**
		2	INT43-INT40	Set EX4 bit in IEN1 register and EXS43-40 bit in IENC register, P4.3-P4.0 in input mode
		3	P4.3-P4.0	Above condition is not met

PORT5:

- XTAL1 (P5.0): oscillator input
- XTAL2 (P5.1): oscillator output
- $\overline{\text{RST}}$ (P5.2): system reset pin
- BUZ (P5.3): buzzer output
- T3 (P5.3): Timer3 external input

Table 7.23 PORT5 Share Table

Pin No.		Priority	Function	Enable bit
SOP28	TSSOP/ SOP20			
6	-	1	XTAL1	Selected by Code Option
		2	P5.0	Above condition is not met
7	-	1	XTAL2	Selected by Code Option
		2	P5.1	Above condition is not met
8	-	1	$\overline{\text{RST}}$	Selected by Code Option
		2	P5.2	Selected by Code Option

7.8 Timer

7.8.1 Features

- The SH79F1620 has four timers (Timer2, 3, 4, 5)
- Timer2 and Timer4 is a 16-bit auto-reload timer and can be selected as a baud-rate generator
- Timer3 is a 16-bit auto-reload timer and can operate even in Power-Down mode
- Timer5 is a 16-bit auto-reload timer

7.8.2 Timer2

The Timer 2 is implemented as a 16-bit register accessed as two cascaded data registers: TH2 and TL2. It is controlled by the register T2CON and T2MOD. The Timer2 interrupt can be enabled by setting the ET2 bit in the IEN0 register. (Refer to Interrupt Section for details)

Timer2 Modes

Timer2 has 2 operating modes: Auto-reload mode with up counter and Baud Rate Generator. These modes are selected by the combination of RCLK, TCLK and CP/RL2.

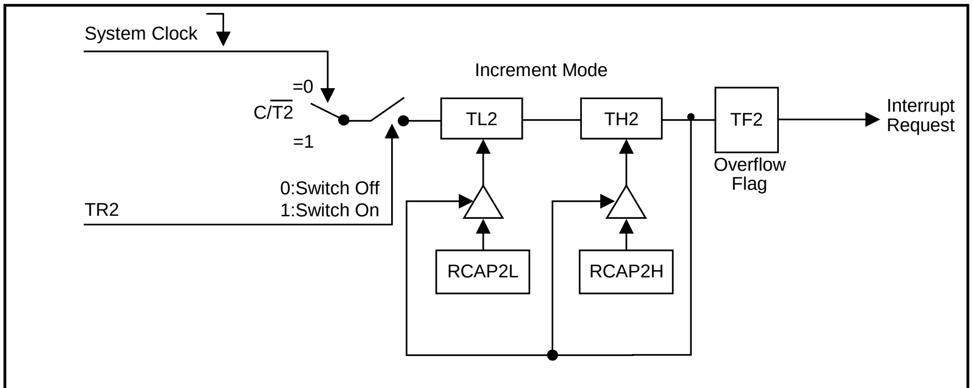
Table 7.24 Timer2 Mode select

C/$\overline{T2}$	T2OE	DCEN	TR2	CP/RL2	RCLK	TCLK	MODE	
0	0	0	1	0	0	0	1	16 bit auto-reload timer
0	0	0	1	0	1	X	2	Baud-Rate generator
					X	1		

Mode1: 16 bit auto-reload Timer

Set the **CP/RL2 = 0, C/$\overline{T2}$ = 0, EXEN2 = 0** and register T2MOD write zero, The Timer 2 is implemented as a 16-bit auto-reload timer with up counter.

Timer2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded to TH2&TL2 registers with the 16-bit value in RCAP2H and RCAP2L, which are pressed by software. The TF2 bit can generate an interrupt if ET2 is enabled.



The Block Diagram of Auto Relode Mode (Mode 0)of Timer2

Mode2: Baud-Rate Generator

Timer2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON. The baud rates for transmit and receive can be different if Timer2 is used for the receiver or transmitter and Timer4 is used for the other.

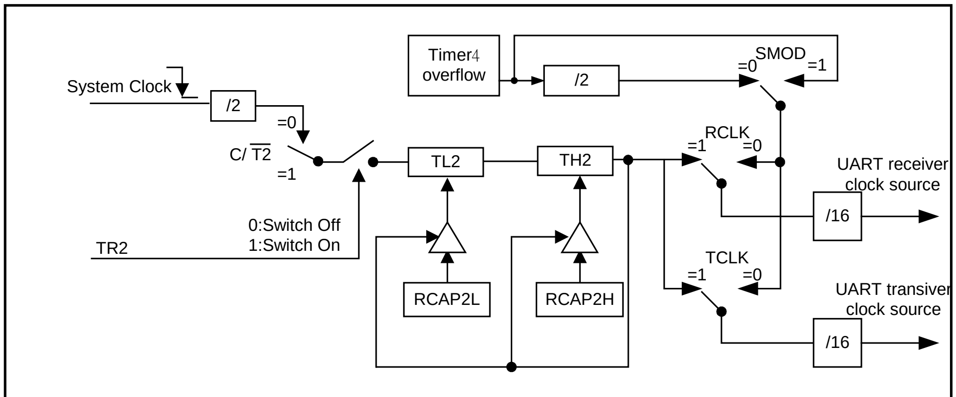
Setting RCLK and/or TCLK will put Timer2 into its baud rate generator mode, which is similar to the auto-reload mode.

Set CP/RL2 = 0,C/$\overline{\text{T2}}$ = 0,EXEN2 = 0 and T2MOD is cleared.

Over flow of Timer2 will causes the Timer2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L that preset by software. But this will not generate an interrupt.

The baud rate in EUART mode 1&3 is determined by the timer 2 overflow ratio according to the following equation.

$$\text{BaudRate} = \frac{1}{2 \times 16} \times \frac{f_{\text{SYS}}}{65536 - [\text{RCAP2H}, \text{RCAP2L}]}\ ;\ \text{C}/\overline{\text{T2}} = 0$$



The Block Diagram of Baud-Rate Generator (Mode 1) of Timer2

Note:

(1) Set TF2 to 1 when incident happens or at anytime by software, only software & hardware reset can clear it.
(2) When EA = 1 and ET2 = 1, setting TF2 to 1 to generate Timer2 interrupt.
(3) When Timer2 is used as baud-rate generator, write in TH2/TL2. Write in RCAPH2/RCAPL2 will effect the veracity of baud rate. It also will cause communication error.

Registers

Table 7.25 Timer2 Control Register

C8H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	$C/\overline{T2}$	$CP/\overline{RL2}$
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**TF2**	**Timer2 overflow flag bit** 0: No overflow(must be cleared by software) 1: Overflow (Set by hardware if RCLK = 0 & TCLK = 0)
6	**EXF2**	**Write this bit 0**
5	**RCLK**	**EUART0 Receive Clock control bit** 0: Timer4 generates receiveing baud-rate 1: Timer2 generates receiveing baud-rate
4	**TCLK**	**EUART0 Transmit Clock control bit** 0: Timer4 generates transmitting baud-rate 1: Timer2 generates transmitting baud-rate
3	**EXEN2**	**Write this bit 0**
2	**TR2**	**Timer2 start/stop control bit** 0: Stop Timer2 1: Start Timer2
1	**$C/\overline{T2}$**	**Write this bit 0**
0	**$CP/\overline{RL2}$**	**Write this bit 0**

Table 7.26 Timer2 Mode Control Register

C9H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T2MOD	-	-	-	-	-	-	T2OE	DCEN
R/W	-	-	-	-	-	-	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0

Bit Number	Bit Mnemonic	Description
1	**T2OE**	**Write this bit 0**
0	**DCEN**	**Write this bit 0**

Table 7.27 Timer2 Reload/Capture & Data Registers

CAH-CDH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RCAP2L	RCAP2L.7	RCAP2L.6	RCAP2L.5	RCAP2L.4	RCAP2L.3	RCAP2L.2	RCAP2L.1	RCAP2L.0
RCAP2H	RCAP2H.7	RCAP2H.6	RCAP2H.5	RCAP2H.4	RCAP2H.3	RCAP2H.2	RCAP2H.1	RCAP2H.0
TL2	TL2.7	TL2.6	TL2.5	TL2.4	TL2.3	TL2.2	TL2.1	TL2.0
TH2	TH2.7	TH2.6	TH2.5	TH2.4	TH2.3	TH2.2	TH2.1	TH2.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	RCAP2L.x	**Timer2 Reload/Capture Data Low & High byte, x = 0 - 7**
	RCAP2H.x	
7-0	TL2.x	**Timer2 Low/High byte counter, x = 0 - 7**
	TH2.x	

7.8.3 Timer3

Timer3 is a 16-bit auto-reload timer. It is accessed as two cascaded data registers: TH3 and TL3. It is controlled by the T3CON register. The Timer3 interrupt can be enabled by setting ET3 bit in IEN1 register (Refer to **Interrupt** Section for details).

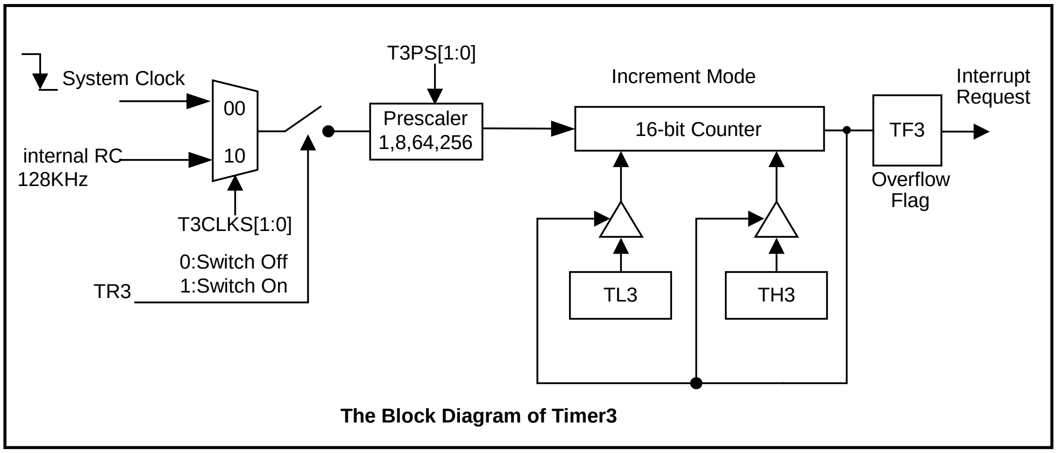
Timer3 has only one operating mode: 16-bit Counter/Timer with auto-reload. Timer3 also supports the following features: selectable pre-scaler setting and Operation during CPU Power-Down mode.

Timer3 consists of a 16-bit Counter/Timer register (TH3, TL3). When writing to TH3 and TL3, they are used as Timer reload register. When reading from TH3 and TL3, they are used as Counter register. Setting the TR3 bit enables Timer 3 to count up. The Timer will overflow from 0xFFFF to 0x0000 and set the TF3 bit. This overflow also causes the 16-bit value written in timer load register to be reloaded into the timer counter register. Writing to TH3 also can cause the 16-bit value written in timer load register to be reloaded into the timer counter register.

Read or write operation to TH3 and TL3 should follow these steps:

Write operation: Low bits first, High bits followed.

Read operation: High bits first, Low bits followed.



The Block Diagram of Timer3

Timer3 can operate in Power-Down mode.

When OP_OSC[3:0] (Refer to Code Option Section for details) is 1010, 0011 or 0110, T3CLKS [1:0] can select 00, or 10. When OP_OSC[3:0] is not 1010, 0011 or 0110, T3CLKS[1:0] can only be selected as 00, and 10 will be an invalid value.

If T3CLKS[1:0] is 00, Timer 3 can't work in Power Down mode. If T3CLKS[1:0] is 10 and OP_OSC[3:0] is 1010, 0011 or 0110, Timer3 can work in CPU normal mode or Power Down mode If T3CLKS[1:0] is 10 and OP_OSC[3:0] is not 1010, 0011 or 0110, Timer3 can't work. It can be described in the following table.

OP_OSC[3:0]	T3CLKS[1:0]	Can work in normal mode	Can work in Power Down mode
1010, 0011 or 0110	00	YES	NO
	10	YES	YES
Not 1010, 0011or 0110	00	YES	NO
	10	NO	NO

***Note:** When TH3 and TL3 read or written, must make sure TR3 = 0.*

Registers

Table 7.28 Timer3 Control Register

88H, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T3CON	TF3	-	T3PS.1	T3PS.0	-	TR3	T3CLKS.1	T3CLKS.0
R/W	R/W	-	R/W	R/W	-	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	-	0	0	-	0	0	0

Bit Number	Bit Mnemonic	Description
7	**TF3**	**Timer3 overflow flag bit** 0: No overflow (cleared by hardware) 1: Overflow (Set by hardware)
5-4	**T3PS[1:0]**	**Timer3 input clock Prescaler Select bits** 00: 1/1 01: 1/8 10: 1/64 11: 1/256
2	**TR3**	**Timer3 start/stop control bit** 0: Stop Timer3 1: Start Timer3
1-0	**T3CLKS[1:0]**	**Timer3 Counter/Timer mode select bits** 00: System clock 01: reserved 10: External 32.768kHz crystal or internal 128kHZ RC 11: reserved

Table 7.29 Timer3 Reload/Counter Data Registers

8CH-8DH, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TL3	TL3.7	TL3.6	TL3.5	TL3.4	TL3.3	TL3.2	TL3.1	TL3.0
TH3	TH3.7	TH3.6	TH3.5	TH3.4	TH3.3	TH3.2	TH3.1	TH3.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**TL3.x**	**Timer3 Low & High byte counter, x = 0 - 7**
	TH3.x	

Table 7.30 Timer3 Reload/Count Data Register

89H, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SWTHL	-	-	-	-	-	-	T5HLCON	T3HLCON
R/W	-	-	-	-	-	-	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0

Bit Number	Bit Mnemonic	Description
0	T3HLCON	0: when read TH3, TL3, return T3 count data 1: when read TH3, TL3, return T3 reload register data

7.8.4 Timer4

Timer4 is a 16-bit auto-reload timer. It is accessed as two cascaded data registers: TH4 and TL4. It is controlled by the T4CON register. The Timer 4 interrupt can be enabled by setting ET4 bit in IEN1 register (Refer to **interrupt** Section for details).

When writing to TH4 and TL4, they are used as timer load register. When reading from TH4 and TL4, they are used as timer counter register. Setting the TR4 bit enables Timer 4 to count up. The timer will overflow from 0xFFFF to 0x0000 and set the TF4 bit. This overflow also causes the 16-bit value written in timer load register to be reloaded into the timer counter register. Writing to TH4 also can cause the 16-bit value written in timer load register to be reloaded into the timer counter register.

Read or write operation to TH4 and TL4 should follow these steps:

Write operation: Low bits first, High bits followed.

Read operation: High bits first, Low bits followed.

Timer4 Modes

Timer4 has three operating modes: 16-bit auto-reload timer, serial port Baud Rate Generator and 16 bit auto-reload timer with T4 edge trig. These modes are selected by T4M[1:0] bits in T4CON Register.

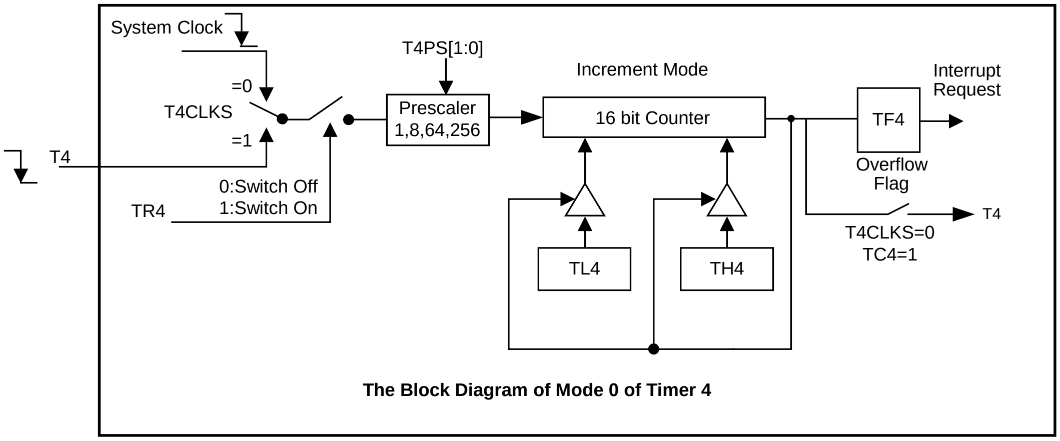
Mode0: 16 bit Auto-Reload Timer

Timer4 operates as 16-bit auto-reload timer in Mode 0. The TH4 register holds the high eight bits of the 16-bit counter/timer, TL4 holds the low eight bits. As the 16-bit timer register increments and overflows from 0xFFFF to 0x0000, the timer overflow flag TF4 (T4CON.7) is set and the 16-bit value in timer load register are reloaded into timer counter register, and an interrupt will occur if Timer 4 interrupts is enabled.

The T4CLKS bit (T4CON.0) selects the counter/timer's clock source. If T4CLKS = 1, external clock from the Pin T4 is selected as Timer4 clock, after prescaled, it will increase the Counter/Timer4 Data register. Else if T4CLKS = 0, the system clock is selected as Timer4 clock.

Setting the TR4 bit (T4CON.1) enables the timer. Setting TR4 does not clear the counter data of Timer4. The timer load register should be loaded with the desired initial value before the timer is enabled.

In Compare mode, the T4 pin is automatically set as output mode by hardware. the internal counter is constantly countered from TH4 and TL4 register value to 0xFFFF. When an overflow occurs, the T4 pin will be inverted. At the same time, interrupt flag bit of Time4 is set. Timer4 must be running in Timer mode (T4CLKS = 0) when compare function enabled.



The Block Diagram of Mode 0 of Timer 4

Mode1: Baud-Rate Generator

Timer4 is selected as the baud rate generator by setting T4MOD bit in T4M[1:0] register. If Timer2 is used for the receiver or transmitter and Timer4 is used for the other, the baud rates for transmit and receive can be different.

The mode is similar to the auto-reload mode. Overflow of Timer4 will causes the Timer4 counter register to be reloaded with the 16-bit value in timer load register. But this will not generate an interrupt.

The baud rates in EUART mode1 and mode3 are determined by Timer4's overflow rate according to the following equation.

$$\text{BaudRate} = \frac{1}{2 \times 16} \times \frac{f_{T4} / \text{PRESCALER}}{65536 - [\text{TH4}, \text{TL4}]}$$

Here, TH4 and TL4 stand for Timer4 reload register.

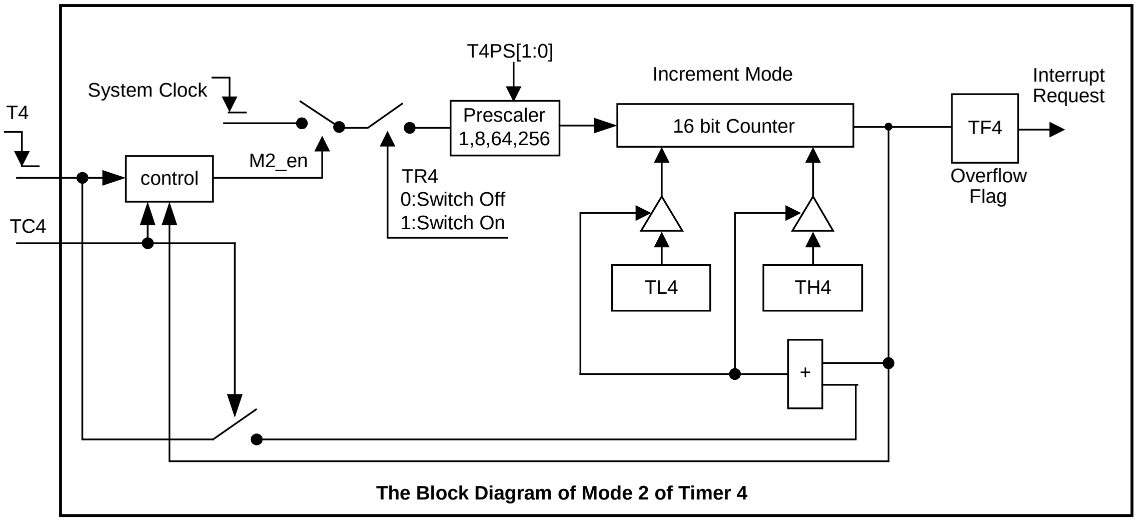
Mode2: 16 bit Auto-Reload Timer with T4 Edge Trig

Timer4 operates as 16-bit timer in Mode2. T4CLKS bit in T4CON.0 will be 0 always.Timer4 can select system clock as clock source.Other setting accords with mode 0.

In Mode2, After Setting the TR4 bit (T4CON.1), Timer4 does not start counting but waits the trig signal (rising or falling edge controlled by T4M[1:0]) from T4. An active trig signal will start the Timer4. When Timer 4 overflows from 0XFFFF to 0x0000, TF4 will be set, TH4 and TL4 will be reloaded from timer load register, and Timer4 holds and waits the next trig edge.

When Timer4 is working, an active trig signal maybe come, if TC4 = 0, the trig signal will be ignored; if TC4 = 1, Timer4 will be re-trigged.

Setting TR4 does not clear the counter data of Timer4. The timer register should be loaded with the desired initial value before the timer is enabled.



The Block Diagram of Mode 2 of Timer 4

control : M2_en set to 1 when T4 edge trig, M2_en set to 0 when counter overflow

Note:

(1) When Timer4 is running (TR4 = 1) as a timer in the baud rate generator mode, TH4 or TL4 should not be written. Because a write might overlap a reload and cause write and/or reload errors. So, the timer 4 must be turned off (TR4 = 0) before accessing the TH4 or TL4 registers.

(2) When Timer4 is used as a counter, the frequency of input signal of T4 pin must be less than half of system clock.

Registers

Table 7.31 Timer4 Control Register

C8H, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T4CON	TF4	TC4	T4PS1	T4PS0	T4M1	T4M0	TR4	T4CLKS
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**TF4**	**Timer4 overflow flag bit** 0: No overflow (cleared by hardware) 1: Overflow (Set by hardware)
6	**TC4**	**Compare function Enable bit** When T4M[1:0] = 00 0: Disable compare function of Timer4 1: Enable compare function of Timer4 When T4M[1:0] = 10 or 11 0: Timer4 can't be re-trigged 1: Timer4 can be re-trigged
5-4	**T4PS[1:0]**	**Timer4 input clock Prescale Select bits** 00: 1/1 01: 1/8 10: 1/64 11: 1/256
3-2	**T4M[1:0]**	**Timer4 Mode Select bit** 00: Mode0, 16-bit auto-reload timer 01: Mode1, baud-rate generator for EUART 10: Mode2 with rising edge trig from pin T4 (system clock only, T4CLKS is invalid) 11: Mode2 with falling edge trig from pin T4 (system clock only, T4CLKS is invalid)
1	**TR4**	**Timer4 start/stop control bit** 0: Stop Timer4 1: Start Timer4
0	**T4CLKS**	**Timer4 Clock Source select bit** 0: System clock, T4 pin is used as I/O port 1: External clock from pin T4 (On the falling edge), the internal pull-up resister is turned on

Table 7.32 Timer4 Reload/Counter Data Registers

CCH-CDH, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TL4	TL4.7	TL4.6	TL4.5	TL4.4	TL4.3	TL4.2	TL4.1	TL4.0
TH4	TH4.7	TH4.6	TH4.5	TH4.4	TH4.3	TH4.2	TH4.1	TH4.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**TL4.x**	**Timer4 Low & High byte counter, x = 0 - 7**
	TH4.x	

7.8.5 Timer5

Timer5 is a 16-bit auto-reload timer. It is accessed as two cascaded data registers: TH5 and TL5. It is controlled by the T5CON register. The interrupt can be enabled by setting ET5 bit in IEN0 register (Refer to **interrupt** Section for details).

When writing to TH5 and TL5, they are used as timer load register. When reading from TH5 and TL5, they are used as timer counter register. Setting the TR5 bit enables Timer5 to count up. The timer will overflow from 0xFFFF to 0x0000 and set the TF5 bit. This overflow also causes the 16-bit value written in timer load register to be reloaded into the timer counter register. Writing to TH4 also can cause the 16-bit value written in timer load register to be reloaded into the timer counter register.

Read or write operation to TH5 and TL5 should follow these steps:

Write operation: Low bits first, High bits followed.

Read operation: High bits first, Low bits followed.

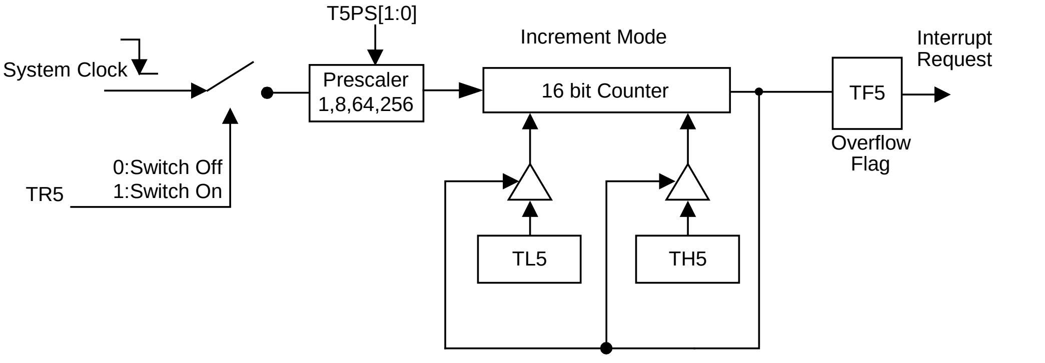
Timer5 Modes

Timer5 has one operating modes: 16-bit auto-reload timer.

Mode0: 16 bit Auto-Reload Counter/Timer

Timer5 operates as 16-bit counter/timer in Mode 0. The TH5 register holds the high eight bits of the 16-bit counter/timer, TL5 holds the low eight bits. As the 16-bit timer register increments and overflows from 0xFFFF to 0x0000, the timer overflow flag TF5 (T5CON.7) is set and the 16-bit value in timer load register are reloaded into timer counter register, and an interrupt will occur if Timer 5 interrupts is enabled.

Setting the TR5 bit (T5CON.1) enables the timer. Setting TR5 does not clear the counter data of Timer4. The timer load register should be loaded with the desired initial value before the timer is enabled.



The Block Diagram of Mode 0 of Timer 5

Registers

Table 7.33 Timer5 Control Register

C0H, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T5CON	TF5	-	T5PS1	T5PS0	-	-	TR5	-
R/W	R/W	-	R/W	R/W	-	-	R/W	-
Reset Value (POR/WDT/LVR/PIN)	0	-	0	0	-	-	0	-

Bit Number	Bit Mnemonic	Description
7	**TF5**	**Timer5 overflow flag bit** 0: No overflow (cleared by hardware) 1: Overflow (Set by hardware)
5-4	**T5PS[1:0]**	**Timer5 input clock Prescale Select bits** 00: 1/1 01: 1/8 10: 1/64 11: 1/256
1	**TR5**	**Timer5 start/stop control bit** 0: Stop Timer5 1: Start Timer5

Table 7.34 Timer5 Reload/Counter Data Registers

CEH-CFH, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TL5	TL5.7	TL5.6	TL5.5	TL5.4	TL5.3	TL5.2	TL5.1	TL5.0
TH5	TH5.7	TH5.6	TH5.5	TH5.4	TH5.3	TH5.2	TH5.1	TH5.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**TL5.x**	**Timer5 Low & High byte counter, x = 0 - 7**
	TH5.x	

Table 7.35 Timer5 Reload/Count Data Register

89H, Bank1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SWTHL	-	-	-	-	-	-	T5HLCON	T3HLCON
R/W	-	-	-	-	-	-	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0

Bit Number	Bit Mnemonic	Description
1	T5HLCON	0: when read TH5, TL5, return T5 count data 1: when read TH5, TL5, return T5 reload register data

7.9 Interrupt

7.9.1 Feature

- 12 interrupt sources
- 4 interrupt priority levels

The SH79F1620 provides total 12 interrupt sources: 3 external interrupts (INT2, INT3, INT4), INT4 has 4 interrupt sources (INT40-43, which share the same vector address), 4 timer interrupts (Timer2, 3, 4, 5), one EUART interrupt, ADC Interrupt, PWM interrupts, SCM interrupt and LPD interrupt.

7.9.2 Interrupt Enable Control

Each interrupt source can be individually enabled or disabled by setting or clearing the corresponding bit in the interrupt enable registers IEN0 or IEN1. The IEN0 register also contains global interrupt enable bit, EA, which can enable/disable all the interrupts at once. Generally, after reset, all interrupt enable bits are set to 0, which means that all the interrupts are disabled.

7.9.3 Register

Table 7.36 Primary Interrupt Enable Register

A8H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IEN0	EA	EADC	ET2	ES0	-	EX1	ET5	EX0
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	-	0	0	0

Bit Number	Bit Mnemonic	Description
7	**EA**	**All interrupt enable bit** 0: Disable all interrupt 1: Enable all interrupt
6	**EADC**	**ADC interrupt enable bit** 0: Disable ADC interrupt 1: Enable ADC interrupt
5	**ET2**	**Timer2 overflow interrupt enable bit** 0: Disable Timer2 overflow interrupt 1: Enable Timer2 overflow interrupt
4	**ES**	**EUART interrupt enable bit** 0: Disable EUART interrupt 1: Enable EUART interrupt
2	**EX1**	**Write this bit 0**
1	**ET5**	**Timer5 overflow interrupt enable bit** 0: Disable Timer5 overflow interrupt 1: Enable Timer5 overflow interrupt
0	**EX0**	**Write this bit 0**

Note: *IEN0[0] and :IEN0[2] must write 0.*

Table 7.37 Secondary Interrupt Enable Register

A9H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IEN1	ESCM/ELPD	ET4	EPWM	ET3	EX4	EX3	EX2	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	-

Bit Number	Bit Mnemonic	Description
7	**ESCM/ELPD**	**SCM/LPD interrupt enable bit** 0: Disable SCM/LPD interrupt 1: Enable SCM/LPD interrupt
6	**ET4**	**Timer4 overflow interrupt enable bit** 0: Disable Timer4 overflow interrupt 1: Enable Timer4 overflow interrupt
5	**EPWM**	**PWM interrupt enable bit** 0: Disable PWM interrupt 1: Enable PWM interrupt
4	**ET3**	**Timer3 overflow interrupt enable bit** 0: Disable timer3 overflow interrupt 1: Enable timer3 overflow interrupt
3	**EX4**	**External interrupt4 enable bit** 0: Disable external interrupt4 1: Enable external interrupt4
2	**EX3**	**External interrupt3 enable bit** 0: Disable external interrupt3 1: Enable external interrupt3
1	**EX2**	**External interrupt2 enable bit** 0: Disable external interrupt2 1: Enable external interrupt2

Note:

(1) To enable External interrupt2/3/4, the corresponding port must be set to input mode before using it.

(2) To enable PWM timer interrupt, the EPWM bit here should be set. Also, the PWMIE in PWM interrupt control register should be set.

Table 7.38 Interrupt channel Enable Register

BAH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IENC	-	-	-	-	EXS43	EXS42	EXS41	EXS40
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	**EXS4x (x = 3-0)**	**External interrupt4 channel select bit (x = 3-0)** 0: Disable external interrupt 4x 1: Enable external interrupt 4x

Table 7.39 Interrupt channel Enable Register1

BBH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IENC1	-	-	-	-	-	-	ESCM1	ELPD
R/W	-	-	-	-	-	-	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0

Bit Number	Bit Mnemonic	Description
1	**ESCM1**	**SCM interrupt enable bit** 0: Disable SCM interrupt 1: Enable SCM interrupt
0	**ELPD**	**LPD interrupt enable bit** 0: Disable LPD interrupt 1: Enable LPD interrupt

7.9.4 Interrupt Flag

Each Interrupt source has its own interrupt flag, when interrupt occurs, corresponding flag will be set by hardware, the interrupt flag bits are listed in interrupt abstract table.

When an external interrupt INT2/3 is generated, if the interrupt was edge trigged, the flag IEx (x = 2-3) that generated this interrupt is cleared by hardware when the service routine is vectored. If the interrupt was level trigged, then the requesting external source directly controls the request flag, rather than the on-chip hardware.

When INT4 generates an interrupt, the flag (IF4x (x = 0-3) in EXF1 register) that generated this interrupt should be cleared by user’s program because the same vector entrance was used in INT4. But if INT4 is set up as level trigged, the flag can’t be cleared by user’s program, it only be controlled by peripheral signal level that connect to INT source pin.

The **Timer2 interrupt** is generated by the logical OR of flag TF2 in T2CON register, which is set by hardware. None of these flags can be cleared by hardware after CPU responses to the interrupt, the flag must be cleared by software.

When the Timer3 counter overflow, set interrupt flag bit TF3 in T3CON to 1 to generate Timer3 interrupt. The flag will be cleared automatically by hardware after CPU responses to the interrupt.

When the Timer4 counter overflow, set interrupt flag bit TF4 in T4CON to 1 to generate Timer4 interrupt. The flag will be cleared automatically by hardware after CPU responses to the interrupt.

When the Timer5 counter overflow, set interrupt flag bit TF5 in T5CON to 1 to generate Timer5 interrupt. The flag will be cleared automatically by hardware after CPU responses to the interrupt.

The **EUART interrupt** is generated by the logical OR of flag RI and TI in SCON register, which is set by hardware. Neither of these flags can be cleared by hardware when the service routine is vectored. In fact, the service routine will normally have to determine whether it was the receive interrupt flag or the transmission interrupt flag that generated the interrupt, so the flag must be cleared by software.

The **ADC interrupt** is generated by ADCIF bit in ADCON. If an interrupt is generated, the converted result in ADCDH/ADCDL will be valid. If continuous compare function in ADC module is Enable, ADCIF will not be clear at each conversion when conversion results is less than the compare value. But if converted result is larger than compare value, ADCIF bit will be 1. The flag must be cleared by software.

The **SCM interrupt** is generated by SCMIF in SCM register, which is set by hardware. And the flag can only be cleared by hardware.

The **LPD interrupt** is generated by LPDF in LPDCON register. And the flag can only be cleared by hardware. By setting the LPDMD, can choose when the V_{DD} voltage is above or below the LPD set generated when the detecting voltage interruption of LPD.

The **PWM interrupts** are generated by PWM0IF in PWM0C. The flags can be cleared by software.

Table 7.40 External Interrupt Flag Register

E8H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EXF0	IT4.1	IT4.0	IT3.1	IT3.0	IT2.1	IT2.0	IE3	IE2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-6	**IT4[1:0]**	**External interrupt4 trigger mode selection bit** 00: Low Level trigger 01: Trigger on falling edge 10: Trigger on rising edge 11: Trigger on both edge IT4 [1:0] is effect on external interrupt 4x at the same mode
5-4	**IT3[1:0]**	**External interrupt3 trigger mode selection bit** 00: Low Level trigger 01: Trigger on falling edge 10: Trigger on rising edge 11: Trigger on both edge
3-2	**IT2[1:0]**	**External interrupt2 trigger mode selection bit** 00: Low Level trigger 01: Trigger on falling edge 10: Trigger on rising edge 11: Trigger on both edge
1	**IE3**	**External interrupt3 request flag bit** 0: No interrupt pending 1: Interrupt is pending
0	**IE2**	**External interrupt2 request flag bit** 0: No interrupt pending 1: Interrupt is pending

Table 7.41 External Interrupt 4 Flag Register

D8H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EXF1	-	-	-	-	IF43	IF42	IF41	IF40
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	**IF4x** **(x = 3-0)**	**External interrupt4 request flag bit** 0: No interrupt pending 1: Interrupt is pending IF4x is cleared by software

7.9.5 Interrupt Vector

When an interrupt occurs, the program counter is pushed onto the stack and the corresponding interrupt vector address is loaded into the program counter. The interrupt vector addresses are listed in **Interrupt Summary table**.

7.9.6 Interrupt Priority

Each interrupt source can be individually programmed to one of four priority levels by setting or clearing corresponding bits in the interrupt priority control registers IPL0, IPH0, IPL1, and IPH1. The interrupt priority service is described below.

An interrupt service routine in progress can be interrupted by a higher priority interrupt, but can not by another interrupt with the same or lower priority.

The highest priority interrupt service cannot be interrupted by any other interrupt source. If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced.

If the same priority interrupt source apply for the interrupt at the beginning of the instruction cycle at the same time, an internal polling sequence determines which request is serviced.

Interrupt Priority		
Priority bits		**Interrupt Lever Priority**
IPHx	**IPLx**	
0	0	Level 0 (lowest priority)
0	1	Level 1
1	0	Level 2
1	1	Level 3 (highest priority)

Table 7.42 Interrupt Priority Control Registers

B8H, B4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IPL0	-	PADCL	PT2L	PSL	-	-	PT5L	-
IPH0	-	PADCH	PT2H	PSH	-	-	PT5H	-
R/W	-	R/W	R/W	R/W	-	-	R/W	-
Reset Value (POR/WDT/LVR/PIN)	-	0	0	0	-	-	0	-
B9H, B5H	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
IPL1	PSCML	PT4L	PPWML	PT3L	PX4L	PX3L	PX2L	-
IPH1	PSCMH	PT4H	PPWMH	PT3H	PX4H	PX3H	PX2H	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	-

Bit Number	Bit Mnemonic	Description
7-0	**PxxxL/H**	Corresponding interrupt source xxx’s priority level selection bits

7.9.7 Interrupt Handling

The interrupt flags are sampled and captured at each machine cycle. All interrupts are sampled at the rising edge of the clock. If one of the flags was set, the CPU will find it and the interrupt system will generate a LCALL to the appropriate service routine, LCALL generated by hardware is not blocked by any of the following conditions:

An interrupt of equal or higher priority is already in progress.

The current cycle is not in the final cycle of the instruction in progress. In other words, any interrupt request can not get response before executing instructions to complete.

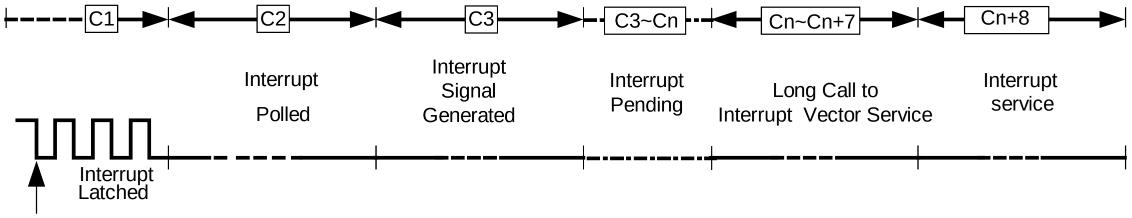
The instruction in progress is RETI or visit the special register IEN01 or IPLH instruction. This ensures that if the instruction in progress is RETI or read and write IEN01 or IPLH then at least one more instruction except RETI will be executed before any interrupt is vectored to; this delay guarantees that the CPU can observe the changes of the interrupt status.

Note:

Since priority change normally needs 2 instructions, it is recommended to disable corresponding Interrupt Enable flag to avoid interrupt between these 2 instructions during the change of priority.

If the flag is no longer active when the blocking condition is removed, the denied interrupt will not be serviced. Every polling cycle interrogates only the valid interrupt requests.

The polling cycle/LCALL sequence is illustrated below:



Interrupt Response Time

The hardware-generated LCALL pushes the contents of the program counter onto the stack (but it does not save the PSW). Then vector address of the corresponding interrupt source (referring to the interrupt vector table) will be stored in the program counter.

Interrupt service execution proceeds from that location until the RETI instruction is encountered. The RETI instruction informs the processor that the interrupt routine is no longer in progress, and then pops the top two bytes from the stack and reloads the program counter. Execution of the interrupted program continues from the point where it was stopped. The RET instruction can also return to the original address to continue, but the interrupt priority control system still think the interrupt in a same priority is responsed, in this case, the same priority or lower priority interrupt will not be responsed.

7.9.8 Interrupt Response Time

If an interrupt is detected, its request flag will be set in every machine cycle after detection. The value will be kept by the internal circuitry until the next machine cycle; the CPU will generate an interrupt at the third machine cycle. If the request is active and conditions are right for it to be acknowledged, hardware instruction LCALL will call service routine which requeste interrupt at the next instruction to be executed. Otherwise the interrupt will pending. The call itself takes 7 machine cycles. Therefore, from the external interrupt request to start the implementation of interrupt program requires at least 3+7 completed machine cycle.

A longer response time would be obtained if the request was blocked by one of the above three previously listed conditions. If an interrupt of equal or higher priority is already in progress, the additional wait time obviously depends on the length of the other interrupt's service routine.

If the instruction in progress is not in its final cycle and the instruction in progress is RETI, the additional wait time is 8 machine cycles. For a single interrupt system, if the next instruction is 20 machine cycles long (the longest instructions DIV & MUL are 20 machine cycles long for 16 bit operation), adding the LCALL instruction 7 machine cycles the total response time is 2+8+20+7 machine cycles.

Thus interrupt response time is always more than 10 machine cycles and less than 37 machine cycles.

7.9.9 External Interrupt Inputs

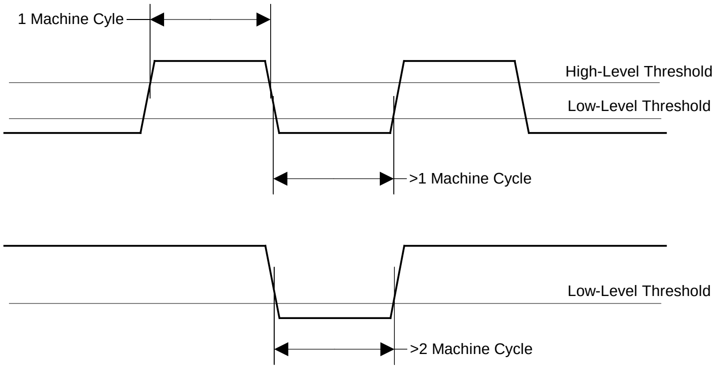
The SH79F1620 has 3 external interrupt inputs. External interrupt 2-3 each has one vector address. External interrupt 4 has 4 inputs; all of them share one vector address. In this mode, a cycle INTx (x = 2, 3) pin continuous sampling is high while the next cycle is low, the interrupt request flag is set to send an interrupt request. Since the external interrupt pin samples one time at every machine cycle, input high or low should be kept for at least one machine cycle to ensure proper sampling.

If the external interrupt is falling-edge-triggered, the external interrupt source has to hold the request pin high level for at least one machine cycle, and then hold it low level for at least one machine cycle. This ensures that the edge can be detected to set IEx. When calling the interrupt service program, CPU will automatically clear IEx.

If the external interrupt is low-level-triggered, the external interrupt source must hold the request active until the requested interrupt is generated, which will take 2 syetem clock cycles. If the external interrupt is still valid when the interrupt service routine is completed, another interrupt will be generated. It is not necessary to clear the interrupt flag IEx (x = 2, 3) when the interrupt is triggered by level, it simply tracks the input pin level.

When SH79F1620 entering idle or power-down mode, the interrupt will wake up the processor to continue to work, see the Power Management chapter.

Note: *External interrupt 2-3 in the execution of an interrupt service routine interrupt flag is automatically cleared by hardware, but the external interrupt flag IF40-43 must be cleared by software.*



7.9.10 Interrupt Summary

Source	Vector Address	Enable bits	Flag bits	Polling Priority	Interrupt number (C51)
Reset	0000H	-	-	0 (higest)	-
Timer5	000BH	ET5	TF5	1	1
EUART	0023H	ES	RI+TI	2	4
Timer2	002BH	ET2	TF2	3	5
ADC	0033H	EADC	ADCIF	4	6
INT2	0043H	EX2	IE2	5	8
INT3	004BH	EX3	IE3	6	9
INT4	0053H	EX4+IENC	IF43-40	7	10
Timer3	005BH	ET3	TF3	8	11
PWM	0063H	EPWM	PWMIF	9	12
Timer4	006BH	ET4	TF4	10	13
SCM/LPD	0073H	ESCM+ESCM1/ELPD	SCMIF/LPDF	11 (lowest)	14

8. Enhanced Function

8.1 PWM (Pulse Width Modulation)

8.1.1 Feature

- Provided overflow and duty interrupt function on every PWM period
- Selectable output polarity
- Lock register provided to avoid PWM control register to be unexpected change

The SH79F1620 has one 12-bit PWM module. Which can provide the pulse width modulation waveform with the period and the duty being controlled individually by corresponding register.

PWM timer also provides 1 interrupts for PWM0. This makes it possible to change period or duty of next cycle in every PWM period.

8.1.2 PWM Module Enable

Table 8.1 PWM Module Enable Register

CFH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWMEN	-	EFLT	-	-	EPWM01	-	-	EPWM0
R/W	-	R/W	-	-	R/W	-	-	R/W
Reset Value (POR/WDT/LVR/PIN)	-	0	-	-	0	-	-	0

Bit Number	Bit Mnemonic	Description
6	**EFLT**	**Write this bit 0**
3	**EPWM01**	**Write this bit 0**
0	**EPWM0**	**Enable 12-bit PWM0** 0: I/O port 1: PWM output

Note:

(1) PWM output will be disable at the same time when the PWM Enable register is clear to 0.
(2) The PWM[3] and PWM[6] must write 0, if PWM function is used.

Table 8.2 PWM Module Enable Register

B7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWMEN1	-	-	-	-	-	-	-	PWM0
R/W	-	-	-	-	-	-	-	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	-	0

Bit Number	Bit Mnemonic	Description
0	**PWM0**	**Enable 12-bit PWM0** 0: PWM output enable 1: PWM output disable, PWM0 as I/O, but PWM timer can work normally, Trigger interrupt

8.1.3 PWM Timer Lock Register

This register is used to control the change of PWM timer enable register, PWM control register and PWM period register and PWM duty register. Only when the data in this register is #55h, it is possible to change these register. Otherwise they cannot be changed.

This register is to enhance the anti-noise ability of SH79F1620.

Table 8.3 PWM Timer Lock Register

E7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWMLO	PWMLO.7	PWMLO.6	PWMLO.5	PWMLO.4	PWMLO.3	PWMLO.2	PWMLO.1	PWMLO.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**PWMLO[7:0]**	**PWM lock register** 55h: enable to change PWM related registers else: disable to change PWM related registers

8.1.4 12-bit PWM Timer

The SH79F1620 has one 12-bit PWM module. The PWM module can provide the pulse width modulation waveform with the period and the duty being controlled, individually. The PWMC is used to control the PWM module operation with proper clocks. The PWMPH/L is used to control the period cycle of the PWM module output. PWMDH/L is used to control the duty in the waveform of the PWM module output.

It is acceptable to change these 3 registers during PWM output Enable. All the change will take affect at the next PWM period.

Table 8.4 12-bit PWM Control Register

D2H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0C	PWM0IE	PWM0IF	-	-	-	PWM0S	TnCK01	TnCK00
R/W	R/W	R/W	-	-	-	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	-	-	-	0	0	0

Bit Number	Bit Mnemonic	Description
7	**PWM0IE**	**PWM0 interrupt enable bit (When EPWM bit in IEN1 is set)** 0: Disable PWM0 interrupt 1: Enable PWM0 interrupt
6	**PWM0IF**	**PWM0 interrupt flag bit** 0: Clear by software 1: Set by hardware to indicate that the PWM0 period counter overflow
2	**PWM0S**	**PWM0 output normal mode of duty cycle** 0: high active, PWM0 output high during duty time, output low during remain period time 1: low active, PWM0 output low during duty time, output high during remain period time
1-0	**TnCK0[1:0]**	**12-bit PWM clock selector:** 00: Oscillator clock/2 01: Oscillator clock/4 10: Oscillator clock/8 11: Oscillator clock/16 ***Note:*** *when OP_OSC is 0000,0011 or 1010, PWM clock is internal RC; when OP_OSC is 1110, PWM clock is crystal/ceramic from XTAL pin; when OP_OSC is 0110, PWM clock is crystal/ceramic from XTALX pin.*

Note: *Inactivate PWM here means PWM0 outputs keep Low (if PWM0S = 0) or High (if PWM0S = 1).*

Table 8.5 PWM Period Control Register (PWM0PL)

D3H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0PL	PP0.7	PP0.6	PP0.5	PP0.4	PP0.3	PP0.2	PP0.1	PP0.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**PP0[7:0]**	12-bit PWM period low 8 bits registers

Table 8.6 PWM Period Control Register (PWM0PH)

D4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0PH	-	-	-	-	PP0.11	PP0.10	PP0.9	PP0.8
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	**PP0[11:8]**	12-bit PWM period high 4 bits registers

PWM output period cycle = [PP0.11, PP0.0] X PWM clock.
When [PP0.11, PP0.0] = 000H, If PWM0S = 0, regardless of the PWM duty cycle, PWM0 output low.
When [PP0.11, PP0.0] = 000H, If PWM0S = 1, regardless of the PWM duty cycle, PWM0 output high.

Table 8.7 PWM Duty Control Register (PWM0DL)

D5H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0DL	PD0.7	PD0.6	PD0.5	PD0.4	PD0.3	PD0.2	PD0.1	PD0.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**PD0[7:0]**	12-bit PWM duty low 8 bits registers

Table 8.8 PWM Duty Control Register (PWM0DH)

D6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0DH	-	-	-	-	PD0.11	PD0.10	PD0.9	PD0.8
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
3-0	**PD0[11:8]**	12-bit PWM duty high 4 bits registers

PWM output duty cycle = [PD0.11, PD0.0] X PWM clock.
If [PP0.11, PP0.0] ≤ [PD0.11, PD0.0], PWM0 outputs high level when the PWM0S bit is set to “0”.
If [PP0.11, PP0.0] ≤ [PD0.11, PD0.0], PWM0 outputs GND level when the PWM0S bit is set to “1”.

Programming Note:

(1) Set PWMLO register to 55H and select the PWM module system clock.

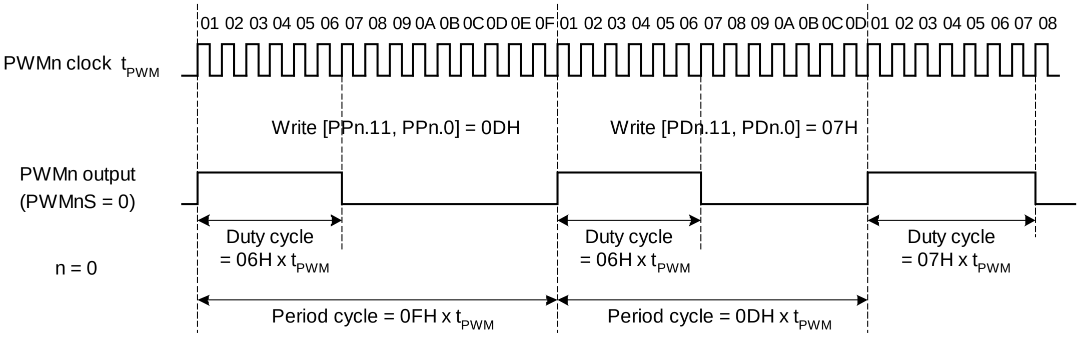
(2) Set the PWM period/duty cycle by writing proper value to the PWM period control register (PWMP) or PWM duty control register (PWMD). First set the low Byte, then the high Byte. Note that even if the high constant value keep unchanged, it also need to rewrite once, otherwise, the low modify is invalid.

(3) Select the PWM output mode (high level valid or low level valid) by writing the PWM0S bit in the PWM control register (PWMC).

(4) In order to output the desired PWM waveform, enable the PWM module by writing “1” to the EPWM0 bit in the PWM control register (PWMC).

(5) If the PWM period cycle or duty cycle is to be changed, the writing flow should be followed as described in step 2 or step 3. The modified reloading counter value will take effect in the next period.

(6) Change the data in PWMLO register not equal to 55h in order to enhance the anti-noise ability.


01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 01 02 03 04 05 06 07 08
PWMn clock t_{PWM}
Write [PPn.11, PPn.0] = 0DH
Write [PDn.11, PDn.0] = 07H
PWMn output
(PWMnS = 0)
n = 0
Duty cycle = 06H x t_{PWM}
Duty cycle = 06H x t_{PWM}
Duty cycle = 07H x t_{PWM}
Period cycle = 0FH x t_{PWM}
Period cycle = 0DH x t_{PWM}

PWM output Period or Duty cycle changing example

8.2 EUART (No EUART in 20 PIN package)

8.2.1 Feature

- The SH79F1620 has one enhanced EUART which are compatible with the conventional 8051
- The baud rate can be selected from the divided clock of the system clock, or Timer4/2 overflow rate
- Enhancements over the standard 8051 the EUART include Framing Error detection and automatic address recognition
- The EUART can be operated in four modes

8.2.2 EUART0 Mode Description

The EUART can be operated in 4 modes. Users must initialize the SCON before any communication can take place. This involves selection of the Mode and the baud rate. The Timer4/2 should also be initialized if the mode 1 or the mode 3 is used.

In all of the 4 modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. This will generate a clock on the TXD pin and shift in 8 bits on the RxD pin. Reception is initiated in the other modes by the input start bit if REN = 1. The external transmitter will start the communication by transmitting the start bit.

EUART Mode Summary

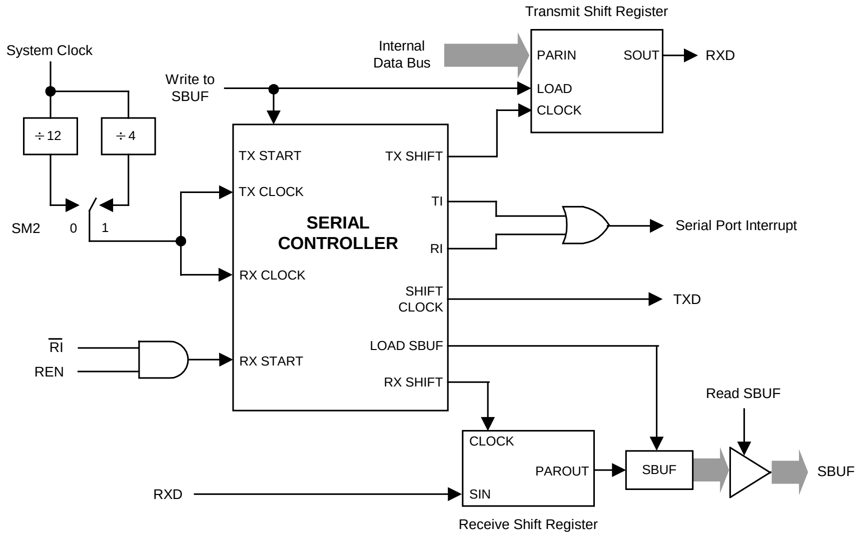
SM0	SM1	Mode	Type	Baud Clock	Frame Size	Start Bit	Stop Bit	9^{th} bit
0	0	0	Synch	f_{SYS}/(4 or 12)	8 bits	None	None	None
0	1	1	Asynch	Timer 4 or 2 overflow rate/(16 or 32)	10 bits	1	1	None
1	0	2	Asynch	f_{SYS}/(32 or 64)	11 bits	1	1	0, 1
1	1	3	Asynch	Timer 4 or 2 overflow rate/(16 or 32)	11 bits	1	1	0, 1

Mode0: Synchronous Mode, Half duplex

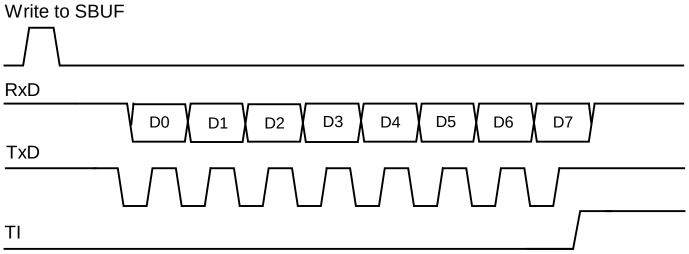
This mode provides synchronous communication with external devices. In this mode serial data is transmitted and received on the RXD pin. TXD is used to output the shift clock. The TXD clock is provided by the SH79F1620 whether the device is transmitting or receiving. Therefore, this mode is a half duplex mode of serial communication. In this mode, 8 bits are transmitted or received per frame. The LSB is transmitted/received first.

The baud rate is programmable to either 1/12 or 1/4 of the system clock. This baud rate is determined in the SM2 bit (SCON.5). When this bit is set to 0, the serial port runs at 1/12 of the system clock. When set to 1, the serial port runs at 1/4 of the system clock. The only difference from standard 8051 is that SH79F1620 in the mode 0 has variable baud rate.

The functional block diagram is shown below. Data enters and exits the serial port on the RXD pin. The TXD pin is used to output the SHIFT CLOCK. The SHIFT CLOCK is used to shift data into and out of the SH79F1620.

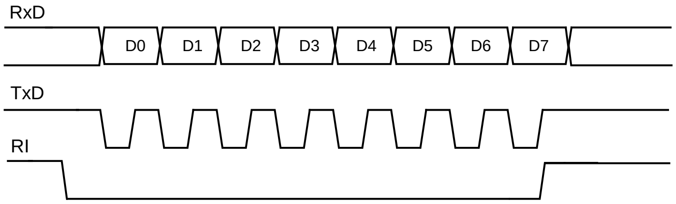


Any instruction that uses SBUF as a destination register ("write to SBUF" signal) will start the transmission. The next system clock tells the Tx control block to commence a transmission. The data shift occurs at the falling edge of the SHIFT CLOCK, and the contents of the transmit shift register is shifted one position from left to the right. As data bits shift to the right, zeros come in from the left. After transmission of all 8 bits in the transmit shift register, the Tx control block will deactivates SEND and sets TI (SCON.1) at the rising edge of the next system clock.



Send Timing of Mode 0

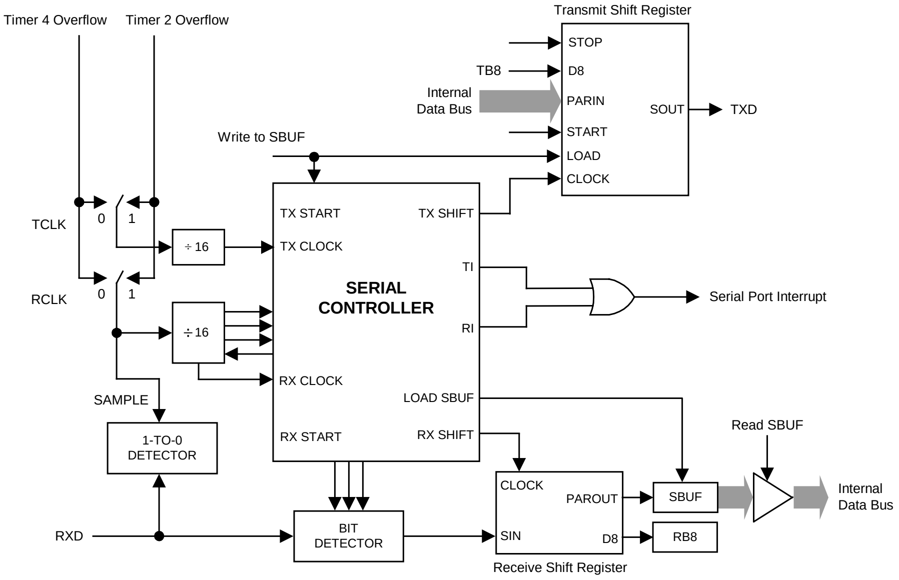
Reception is initiated by the condition REN (SCON.4) = 1 and RI (SCON.0) = 0. The next system clock activates RECEIVE. The data latch occurs at the rising edge of the SHIFT CLOCK, and the contents of the receive shift register are shifted one position to the left. After the receiving of all 8 bits into the receive shift register, the RX control block will deactivates RECEIVE and sets RI at the rising edge of the next system clock, and the reception will not be enabled till the RI is cleared by software.



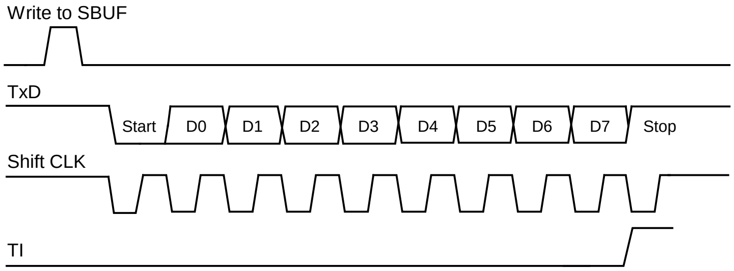
Receive Timing of Mode 0

Mode1: 8-Bit EUART, Variable Baud Rate, Asynchronous Full-Duplex

This mode provides the 10 bits full duplex asynchronous communication. The 10 bits consist of a start bit (logical 0), 8 data bits (LSB first), and a stop bit (logical 1). When receiving, the eight data bits are stored in SBUF and the stop bit goes into RB8 (SCON.2). The serial receive and transmit baud rate is 1/16 of the Timer4/2 overflow (Refer to **Baud Rate** Section for details). The functional block diagram is shown below.



Transmission begins with a “write to SBUF” signal, and it actually commences at the next system clock following the next rollover in the divide-by-16 counter (divide baud-rate by 16), thus, the bit times are synchronized to the divide-by-16 counter, not to the “write to SUBF” signal. The start bit is firstly put out on TxD pin, then are the 8 bits of data. After all 8 bits of data in the transmit shift register are transmitted, the stop bit is put out on the TxD pin, and the TI flag is set at the same time that the stop is send.



Send Timing of Mode 1

Reception is enabled only if REN is high. The serial port actually starts the receiving of serial data with the detection of a falling edge on the RxD pin. For this purpose RxD is sampled at the rate of 16 times baud rate. When a falling edge is detected, the divide-by-16 counter is immediately reset. This helps the divide-by-16 counter to synchronize with the serial datas of RXD pin.

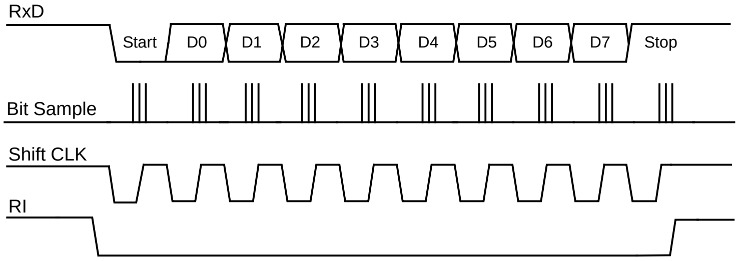
The divide-by-16 counter divides each bit time into 16 states. The bit detector samples the value of RxD at the 7th, 8th and 9th counter states of each bit time. At least 2 the sampling values have no difference in the state of the three samples, data can be received This is done for noise rejection. If the first bit after the falling edge of RxD pin is not 0, which indicates an invalid start bit, and the reception is immediately aborted. The receive circuits are reset and again waiting for a falling edge in the RxD pin. If a valid start bit is detected, then the rest of the bits are also detected and shifted into the shift register. After shifting in 8 data bits and the stop bit, the SBUF and RB8 are loaded and RI are set, if the following conditions are met:

1. RI must be 0

2. Either SM2 = 0, or the received stop bit = 1

If these conditions are met, then the stop bit goes to RB8, the 8 data bits go into SBUF and RI is set. Otherwise the received frame may be lost.

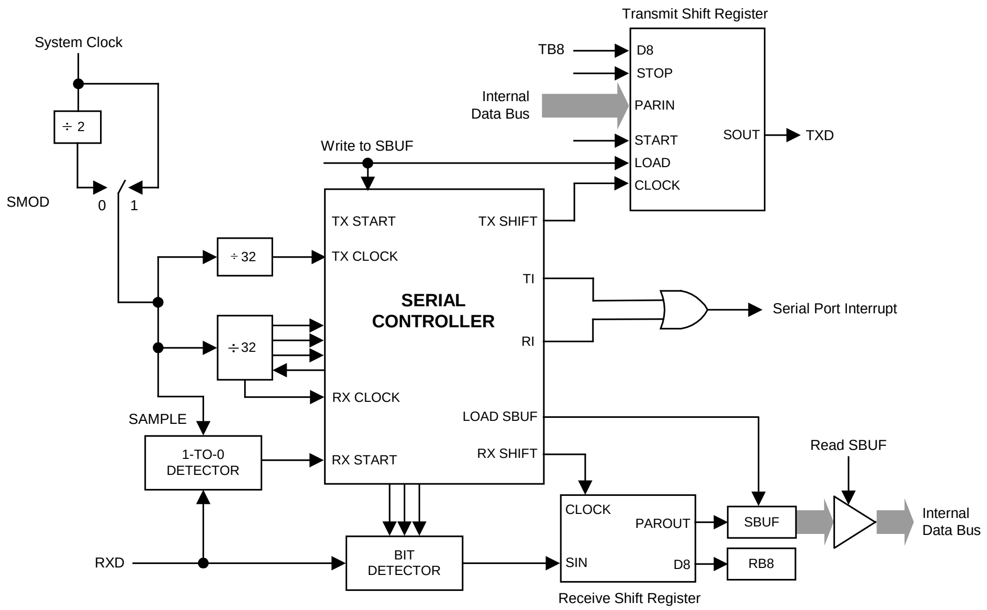
At the time, the receiver goes back to looking for another falling edge on the RxD pin. And the user should clear RI by software for further reception.



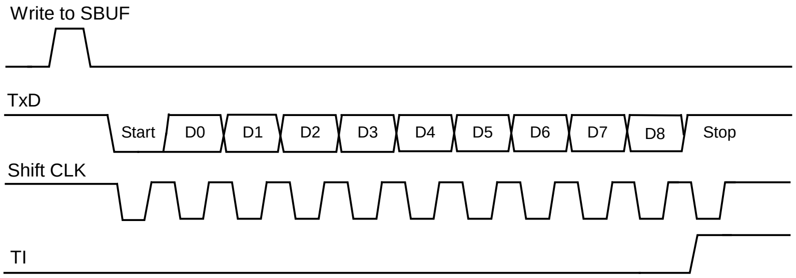
Receive Timing of Mode 1

Mode2: 9-Bit EUART, Fixed Baud Rate, Asynchronous Full-Duplex

This mode provides the 11 bits full duplex asynchronous communication. The 11 bit consists of one start bit (logical 0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (logical 1). Mode 2 supports multiprocessor communications and hardware address recognition (Refer to Multiprocessor Communication Section for details). When data is transmitted, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1, for example, the parity bit P in the PSW or used as data/address flag in multiprocessor communications. When data is received, the 9th data bit goes into RB8 and the stop bit is not saved. The baud rate is programmable to either 1/32 or 1/64 of the system working frequency, as determined by the SMOD bit in PCON. The functional block diagram is shown below:



Transmission begins with a “write to SBUF” signal, the “write to SBUF” signal also loads TB8 into the 9th bit position of the transmit shift register. Transmission actually commences at the next system clock following the next rollover in the divide-by-16 counter (thus, the bit times are synchronized to the divide-by-16 counter, not to the “write to SUBF” signal). The start bit is firstly put out on TxD pin, then are the 9 bits of data. After all 9 bits of data in the transmit shift register are transmitted, the stop bit is put out on the TxD pin, and the TI flag is set at the same time.



Send Timing of Mode 2

Reception is enabled only if REN is high. The serial port actually starts the receiving of serial data, with the detection of a falling edge on the RxD pin. For this purpose RxD is sampled at the rate of 16 times baud rate. When a falling edge is detected, the divide-by-16 counter is immediately reset. This helps the divide-by-16 counter to synchronize with the serial datas of RXD pin.

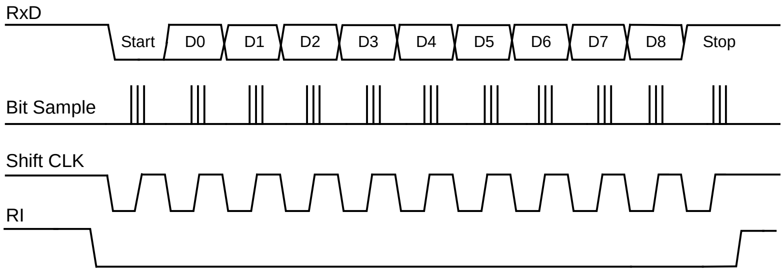
The divide-by-16 counter divides each bit time into 16 states. The bit detector samples the value of RxD at the 7th, 8th and 9th counter state of each bit time. At least 2 the sampling values have no difference in the state of the three samples, data can be received. This is done for noise rejection. If the first bit detected after the falling edge of RxD pin is not 0, which indicates an invalid start bit, and the reception is immediately aborted. The receive circuits are reset and again waiting for a falling edge in the RxD pin. If a valid start bit is detected, then the rest of the bits are also detected and shifted into the shift register. After shifting in 9 data bits and the stop bit, the SBUF and RB8 are loaded and RI is set, if the following conditions are met:

1. RI must be 0

2. Either SM2 = 0, or the received 9th bit = 1 and the received byte accords with Given Address

If these conditions are met, then the 9th bit goes to RB8, the 8 data bits go into SBUF and RI is set. Otherwise the received frame may be lost.

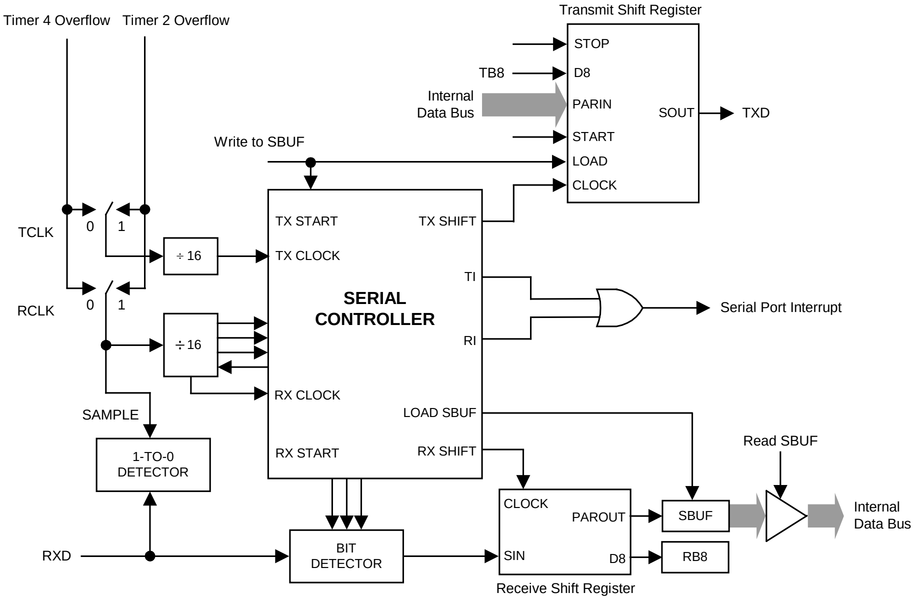
At the time, the receiver goes back to looking for another falling edge on the RxD pin. And the user should clear RI by software for further reception.



Receive Timing of Mode 2

Mode3: 9-Bit EUART, Variable Baud Rate, Asynchronous Full-Duplex

Mode3 uses transmission protocol of the Mode2 and baud rate generation of the Mode1.



8.2.3 Baud Rate Generate

In Mode0, the baud rate is programmable to either 1/12 or 1/4 of the system clock. This baud rate is determined by SM2 bit. When set to 0, the serial port runs at 1/12 of the system clock. When set to 1, the serial port runs at 1/4 of the system clock.

In Mode1 & Mode3, the baud rate can be selected from Timer4/2 overflow rate.

Individually setting TCLK (T2CON.4) and RCLK (T2CON.5) to 1 to select Timer2 as buad clock source of TX & RX (Refer to "Timer" section for details). Whether TCLK or RCLK to logic 1, Timer2 is baud rate generator mode. If TCLK and RCLK are logic 0, Timer4 will be used as the buad clock source of TX & RX.

The Mode1 & 3 baud rate equations are shown below, where [RCAP2H, RCAP2L] is the 16-bit auto-reload register for Timer2, [TH4, TL4] is the 16-bit reload register for Timer4.

$\mathrm{BaudRate} = \frac{1}{2\times16}\times\frac{f_{SYS}}{65536-[\mathrm{RCAP2H},\mathrm{RCAP2L}]}$, Baud Rate using Timer2, the clock source of Timer2 is system clock.

$\mathrm{BaudRate} = \frac{1}{2\times16}\times\frac{f_{T4}/\mathrm{PRESCALER}}{65536-[\mathrm{TH4},\mathrm{TL4}]}$, Baud Rate using Timer4, Timer4 works in Mode1.

In Mode2, the baud rate is programmable to either 1/32 or 1/64 of the system clock. This baud rate is determined by the SMOD bit (PCON.7). When this bit is set to 0, EUART runs at 1/64 of the system clock. When set to 1, EUART runs at 1/32 of the system clock.

$$\mathrm{BaudRate} = 2^{\mathrm{SMOD}}\times(\frac{f_{SYS}}{64})$$

8.2.4 Multi-Processor Communication

Software Address Recognition

Modes2 and 3 of the EUART have a special function for multi-processor communication. In these modes, 9 data bits are received. The 9th bit goes into RB8. Then a stop bit follows. The EUART can be programmed such that when the stop bit is received, the serial port interrupt will be activated (i.e. the request flag RI is set) only if RB8 = 1. This feature is enabled by setting the bit SM2 in SCON.

A way to use this feature in multiprocessor communications is as follows. If the master processor wants to transmit a block of data to one of the several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte by using the 9th bit.The 9th bit is 1 in an address byte ,the 9th bit is 0 in a data byte.

If SM2 is 1,slave will not respond to data byte interrupt. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. After having received a complete message, the slave sets SM2 again. The slaves that were not addressed keep their SM2 setting and go on with their business, ignoring the incoming data bytes.

Note: *In Mode0, SM2 is used to select baud rate doubling. In Mode1, SM2 can be used to check the validity of the stop bit. If SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.*

Automatic (Hardware) Address Recognition

In Mode2 & 3, setting the SM2 bit will configure EUART act as following: when a stop bit is received, EUART will generate an interrupt only if the 9th bit that goes into RB8 is logic 1 (address byte) and the received data byte matches the EUART slave address. Following the received address interrupt, the slave should clear its SM2 bit to enable interrupts to receive the following data byte(s).

The 9-bit mode requires that the 9th information bit is 1 to indicate that the received information is address rather than data. When the master processor wants to transmit a block of data to one of the slaves, it must first send out the address of the targeted slave (or slaves). All the slave processors should have their SM2 bit set high when waiting for an address byte, which ensures that they will be interrupted only by the reception of an address byte. The Automatic address recognition feature further ensures that only the address matching slave will be interrupted. The address comparison is done by hardware not software.

After being interrupted, the address matching slave clears the SM2 bit to receive data bytes. The un-addressed slaves will be unaffected, they will be still waiting for their address. Once the entire message is received, the addressed slave should set its SM2 bit to ignore all not address byte in transmission until it receives the next address byte.

The Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given Address. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave’s address (SADDR) and the address shield (SADEN). The slave address is an 8-bit byte stored in the SADDR register. The SADEN register is actually used to define whether the byte value in SADDR is valid or not. If a bit position in SADEN is 0, then the corresponding bit position in SADDR is i. Only those bit positions in SADDR whose corresponding bits in SADEN are 1 are used to obtain the Given Address. This gives the user flexibility to address multiple slaves without changing the slave address in SADDR. Use of the Given Address allows multiple slaves to be recognized while excluding others.

	Slave 1	Slave 2
SADDR	10100100	10100111
SADEN (bit = 0 will be ignored)	11111010	11111001
Given Address	10100x0x	10100xx1
Broadcast Address (SADDR o rSADEN)	1111111x	11111111

The Given address for slave 1 and 2 differ in the LSB. For slave 1, it is ignore LSB, while for slave 2 LSB is 1. Thus to communicate only with slave 1, the master must send an address with LSB = 0 (10100000). Similarly the bit 1 is 0 for slave 1 and ignore the bit 1 for slave 2. Hence to communicate only with slave 2, the master has to transmit an address with bit 1 = 1 (1010 0011). If the master wishes to communicate with both slaves simultaneously, then the address must have bit 0 = 1 and bit 1 = 0. The bit 2 position is ignored for both the slaves. This allows two different addresses to select both slaves (1010 0001 and 1010 0101).

The master can communicate with all the slaves simultaneously with the Broadcast Address. This address is formed from the logical OR of the SADDR and SADEN. The zeros in the result are defined as neglect. In most cases, the Broadcast Address is FFh, this address will be responded by all slaves.

On reset, the SADDR and SADEN are initialized to 00h. The two results set Given Address and Broadcast Address to XXXXXXXX (all bits are ignored). This effectively removes the multiprocessor communications feature, since any selectivity is disabled. This ensures that the EUART will reply to any address, which it is compatible with the 80C51 microcontrollers that do not support automatic address recognition. So the user may implement multiprocessor communication by software recognition address according to the above mentioned method.

8.2.5 Frame Error Detection

Frame error detection is available when the SSTAT bit in register PCON is set to logic 1.All the 3 error falg bits should be cleared by software after they are set, even when the following frames received without any error will not be cleared automatically.

Note: *The SSTAT bit must be logic 1 to access any of the status bits (FE, RXOV, and TXCOL). The SSTAT bit must be logic 0 to access the Mode Select bits (SM0, SM1, and SM2).*

Transmit Collision

The Transmit Collision bit (TXCOL bit in register SCON) set ‘1’ when a transmission is still in progress and user software writes data to the SBUF register. If collision occurs, the new data will be ignored and the transmit buffer will not be written.

Receive Overflow

The Receive Overflow bit (RXOV in register SCON) set ‘1’ if a new data byte is latched into the receive buffer before software has read the previous byte. The previous data is lost when this happen.

Frame Error

The Frame Error bit (FE in register SCON) set ‘1’ if an invalid (low) STOP bit is detected.

Break Detection

A break is detected when any 11 consecutive bits are detected as low. Since a break condition also satisfies the requirements for a framing error, a break condition will also result in reporting a framing error. Once a break condition has been detected, the UART will go into an idle state and remain in this idle state until a valid stop bit (rising edge on RxD pin) has been received.

8.2.6 Register

Table 8.9 EUART Control & Status Register

98H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SCON	SM0 /FE	SM1 /RXOV	SM2 /TXCOL	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-6	**SM[0:1]**	**EUART Serial mode control bit, when SSTAT = 0** 00: mode 0, Synchronous Mode, fixed baud rate 01: mode 1, 8 bit Asynchronous Mode, variable baud rate 10: mode 2, 9 bit Asynchronous Mode, fixed baud rate 11: mode 3, 9 bit Asynchronous Mode, variable baud rate
7	**FE**	**EUART Frame Error flag, when FE bit is read, SSTAT bit must be set 1** 0: No Frame Error, clear by software 1: Frame error occurs, set by hardware
6	**RXOV**	**EUART Receive Over flag, when RXOV bit is read, SSTAT bit must be set 1** 0: No Receive Over, clear by software 1: Receive over occurs, set by hardware
5	**SM2**	**EUART Multi-processor communication enable bit (9th bit '1' checker), when SSTAT = 0** 0: In Mode0, baud-rate is 1/12 of system clock In Mode1, disable stop bit validation check, any stop bit will set RI to generate interrupt In Mode2 & 3, any byte will set RI to generate interrupt 1: In Mode0, baud-rate is 1/4 of system clock In Mode1, Enable stop bit validation check, only valid stop bit (1) will set RI to generate interrupt In Mode2 & 3, only address byte (9th bit = 1) will set RI to generate interrupt
5	**TXCOL**	**EUART Transmit Collision flag, when TXCOL bit is read, SSTAT bit must be set 1** 0: No Transmit Collision, clear by software 1: Transmit Collision occurs, set by hardware
4	**REN**	**EUART Receiver enable bit** 0: Receive Disable 1: Receive Enable
3	**TB8**	**The 9th bit to be transmitted in Mode2 & 3 of EUART, set or clear by software**
2	**RB8**	**The 9th bit to be received in Mode1, 2 & 3 of EUART** In Mode0, RB8 is not used In Mode1, if receive interrupt occurs, RB8 is the stop bit that was received In Modes2 & 3 it is the 9th bit that was received
1	**TI**	**Transmit interrupt flag of EUART** 0: cleared by software 1: Set by hardware at the end of the 8th bit time in Mode0, or at the beginning of the stop bit in other modes
0	**RI**	**Receive interrupt flag of EUART** 0: cleared by software 1: Set by hardware at the end of the 8th bit time in Mode0, or during the stop bit time in other modes

Table 8.10 EUART Data Buffer Register

99H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SBUF	SBUF.7	SBUF.6	SBUF.5	SBUF.4	SBUF.3	SBUF.2	SBUF.1	SBUF.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**SBUF[7:0]**	This SFR accesses two registers; a transmit shift register and a receive latch register A write of SBUF will send the byte to the transmit shift register and then initiate a transmission A read of SBUF returns the contents of the receive latch

Table 8.11 Power Control Register

87H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCON	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**SMOD**	**Baud rate doubler** Set SMOD in Mode1 & 3, the baud-rate of EUART is doubled if using time4 as baud-rate generator set SMOD in Mode2, the baud-rate of EUART is doubled
6	**SSTAT**	**SCON[7:5] function select bit** 0: SCON[7:5] operates as SM0, SM1, SM2 1: SCON[7:5] operates as FE, RXOV, TXCOL
3-0	**--**	**Other: See "Power Management" chapter**

Table 8.12 EUART Slave Address & Address Mask Register

9AH-9BH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SADDR	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
SADEN	SADEN.7	SADEN.6	SADEN.5	SADEN.4	SADEN.3	SADEN.2	SADEN.1	SADEN.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**SADDR[7:0]**	**SFR SADDR defines the EUART’s slave address**
7-0	**SADEN[7:0]**	**SFR SADEN is a bit mask to determine which bits of SADDR are checked against a received address** 0: Corresponding bit in SADDR is a “don’t care” 1: Corresponding bit in SADDR is checked against a received address

Table 8.13 Rxd Pin Schmidt Voltage Control Register

9FH, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RxCON	-	-	-	-	-	-	RxCON1	RxCON0
R/W	-	-	-	-	-	-	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0

Bit Number	Bit Mnemonic	Description
1-0	**RxCON[1:0]**	**Rxd pin** Schmidt **voltage control** 00: input low voltage is 0.2 V_{DD} 01: input low voltage is 0.4 V_{DD} 10: input low voltage is 0.5 V_{DD} 11: normal IO

Note: *RxCON is available when EUART is enable,input low voltage is measured at 25°C. Refer to* ***Electrical Characteristics*** *for detail.*

8.3 Analog Digital Converter (ADC)

8.3.1 Feature

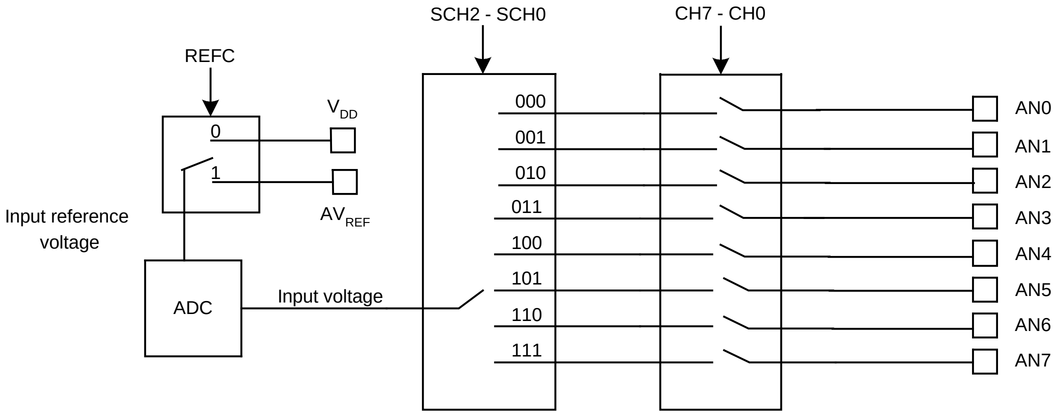
- 10-bit Resolution
- Build in V_{REF}
- Selectable external or built-in V_{REF} (No external V_{REF} in 20 PIN package)
- 8/6 analog Channels inpu

The SH79F1620 includes a single ended, 10-bit SAR Analog to Digital Converter (ADC) with build in reference voltage connected to the V_{DD}, users also can select the AVREF port input reference voltage. The 8/6 ADC channels are shared with 1 ADC module; each channel can be programmed to connect with the analog input individually. Only one channel can be available at one time. GO/$\overline{\text{DONE}}$ signal is available to start convert, and indicate end of convert. When conversion is completed, the data in AD convert data register will be updated and ADCIF bit in ADCON register will be set. If ADC Interrupt is enabled, the ADC interrupt will be generated.

The ADC integrates a digital compare function to compare the value of analog input and the digital value in the AD converter. If this function is enabled (set EC bit in ADCON register) and ADC module is enabled (set ADON bit in ADCON register). When the corresponding digital value of analog input is larger than the compare value in register (ADDH/L), the ADC interrupt will occur, otherwise no interrupt will be generated. The digital comparator can work continuously when GO/$\overline{\text{DONE}}$ bit is set until software clear, which behaviors different with the AD converter operation mode.

The ADC module including digital compare module can wok in Idle mode and the ADC interrupt will wake up the Idle mode, but is disabled in Power-Down mode.

8.3.2 ADC Diagram



ADC Diagram

8.3.3 ADC Register

Table 8.14 ADC Control Register

93H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCON	ADON	ADCIF	EC	REFC	SCH2	SCH1	SCH0	GO/$\overline{\text{DONE}}$
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**ADON**	**ADC Enable bit** 0: Disable the ADC module 1: Enable the ADC module
6	**ADCIF**	**ADC Interrupt Flag bit** 0: No ADC interrupt, cleared by software. 1: Set by hardware to indicate that the AD Convert has been completed, or analog input is larger than ADDATH/L if compare is enabled
5	**EC**	**Compare Function Enable bit** 0: Compare function disabled 1: Compare function enabled
4	**REFC**	**Reference Voltage Select bit** 0: the reference voltage connected to V_{DD} 1: the reference voltage input from V_{REF} pin
3-1	**SCH[2:0]**	**ADC channel Select bits** 000: ADC channel AN0 001: ADC channel AN1 010: ADC channel AN2 011: ADC channel AN3 100: ADC channel AN4 101: ADC channel AN5 110: ADC channel AN6 111: ADC channel AN7
0	**GO/$\overline{\text{DONE}}$**	**ADC status flag bit** 0: Automatically cleared by hardware when AD convert is completed. Clearing this bit during converting time will stop current conversion. If Compare function is enabled, this bit will not be cleared by hardware until software clear. 1: Set to start AD convert or digital compare.

Note:

(1) When select the reference voltage input from V_{REF} pin (REFC = 1), the P4.4 is shared as V_{REF} input rather than AN3 input. No AVREF PIN in 20 PIN package, REFC bit must write 0.

(2) No AN4 and AN6 channel in 20 PIN package, SCH[2:0] can not be set 100 and 110.

Table 8.15 ADC Time Configuration Register

94H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADT	TADC2	TADC1	TADC0	-	TS3	TS2	TS1	TS0
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-5	**TADC[2:0]**	**ADC Clock Period Select bits** 000: ADC Clock Period t_{AD} = 2 t_{SYS} 001: ADC Clock Period t_{AD} = 4 t_{SYS} 010: ADC Clock Period t_{AD} = 6 t_{SYS} 011: ADC Clock Period t_{AD} = 8 t_{SYS} 100: ADC Clock Period t_{AD} = 12 t_{SYS} 101: ADC Clock Period t_{AD} = 16 t_{SYS} 110: ADC Clock Period t_{AD} = 24 t_{SYS} 111: ADC Clock Period t_{AD} = 32 t_{SYS}
3-0	**TS[3:0]**	**Sample time select bits** 2 t_{AD} ≤ Sample time = (TS [3:0]+1) * t_{AD} ≤ 15 t_{AD}

Note:

(1) Make sure that t_{AD} ≧ 1µs;
(2) The minimum sample time is 2 t_{AD}, even TS[3:0] = 0000;
(3) The maximum sample time is 15 t_{AD} , even TS[3:0] = 1111;
(4) Evaluate the series resistance connected with ADC input pin before set TS[3:0];
(5) Be sure that the series resistance connected with ADC input pin is no more than 10kΩ when 2 t_{AD} sample time is selected;
(6) Total conversion time is: 12 t_{AD} + sample time.

For Example

System Clock (SYSCLK)	TADC[2:0]	t_{AD}	TS[3:0]	Sample Time	Conversion Time
32.768kHz	000	30.5 *2=61µs	0000	2*61=122µs	12*61+122=854µs
	000	30.5 *2=61µs	0111	8*61=488µs	12*61+488=1220µs
	000	30.5 *2=61µs	1111	15*61=915µs	12*61+915=1647µs
	111	30.5 *32=976µs	0000	2*976=1952µs	12*976+1952=13664µs
	111	30.5 *32=976µs	0111	8*976=7808µs	12*976+7808=19520µs
	111	30.5 *32=976µs	1111	15*976=14640µs	12*976+14640=26352µs
4MHz	000	0.25 *2=0.5µs	-	-	(t_{AD} < 1µs, Not recommended)
	001	0.25 *4=1µs	0000	2*1=2µs	12*1+2=14µs
	001	0.25 *4=1µs	0111	8*1=8µs	12*1+8=20µs
	001	0.25 *4=1µs	1111	15*1=15µs	12*1+15=27µs
	111	0.25 *32=8µs	0000	2*8=16µs	12*8+16=112µs
	111	0.25 *32=8µs	0111	8*8=64µs	12*8+64=160µs
	111	0.25 *32=8µs	1111	15*8=120µs	12*8+120=216µs
12MHz	000	0.083*2=0.166µs	-	-	t_{AD} < 1µs, Not recommended)
	100	0.083*12=1µs	0000	2*1=2µs	12*1+2=14µs
	100	0.083*12=1µs	0111	8*1=8µs	12*1+8=20µs
	100	0.083*12=1µs	1111	15*1=15µs	12*1+15=27µs
	111	0.083*32=2.7µs	0000	2*2.7=5.4µs	12*2.7+5.4=37.8µs
	111	0.083*32=2.7µs	0111	8*2.7=21.6µs	12*2.7+21.6=54µs
	111	0.083*32=2.7µs	1111	15*2.7=40.5µs	12*2.7+40.5=72.9µs

Table 8.16 ADC Channel Configure Register

95H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCH	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**CH[7:0]**	**Channel Configuration bits** 0: P4.0-P4.3, P3.4-P3.7 are I/O port 1: P4.0-P4.3, P3.4-P3.7 are ADC input port

Note: *No P3.4 and P3.6 PIN in 20 PIN package, CH4 and CH6 must write 0.*

Table 8.17 AD Converter Data Register (Compare Value Register)

96H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADDL	-	-	-	-	-	-	A1	A0
R/W	-	-	-	-	-	-	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0
97H, Bank0	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
ADDH	A9	A8	A7	A6	A5	A4	A3	A2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
1-0 **7-0**	**A9-A0**	**ADC Data register** Digital Value of sampled analog voltage, updated when conversion is completed If ADC Compare function is enabled (EC = 1), this is the value to be compared with the analog input

The Approach for AD Conversion:

(1) Select the analog input channels and reference voltage.
(2) Enable the ADC module with the selected analog channel.
(3) Set GO/$\overline{\text{DONE}}$ = 1 to start the AD conversion.
(4) Wait until GO/$\overline{\text{DONE}}$ = 0 or ADCIF = 1, if the ADC interrupt is enabled, the ADC interrupt will occur, user need clear ADCIF by software.
(5) Acquire the converted data from ADDH/ADDL.
(6) Repeat step 3-5 if another conversion is required.

The Approach for Digital Compare Function:

(1) Select the analog input channels and reference voltage.
(2) Write ADDH/ADDL to set the compare value.
(3) Set EC = 1 to enable compare function.
(4) Enable the ADC module with the selected analog channel.
(5) Set GO/$\overline{\text{DONE}}$ = 1 to start the compare function.
(6) If the analog input is lager than compare value set in ADDH/ADDL, the ADCIF will be set to 1. if the ADC interrupt is enabled, the ADC interrupt will occur,user need clear ADCIF by software.
(7) The compare function will continue work until the GO/$\overline{\text{DONE}}$ bit is cleared to 0.

8.4 Low Power Detect (LPD)

8.4.1 Feature

- Low power detect and generate interrupt
- LPD detect voltage is selectable
- LPD de-bounce timer T_{LPD} is about 30-60µs

The low power detect (LPD) is used to monitor the supply voltage and generate an internal flag if the voltage decrease below the specified value. It is used to inform CPU whether the power is shut off or the battery is used out, so the software may do some protection action before the voltage drop down to the minimal operation voltage.

The LPD interrupt can wake the power down mode up.

8.4.2 Register

Table 8.18 Low Power Detection Control Register

B3H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPDCON	LPDEN	LPDF*	LPDMD	LPDIF	LPDS3	LPDS2	LPDS1	LPDS0
R/W	R/W	R*	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**LPDEN**	**LPD Enable bit** 0: Disable lower power detection 1: Enable lower power detection
6	**LPDF**	**LPD status Flag bit** 0: No LPD happened, clear by hardware, 1: LPD happened, set by hardware
5	**LPDMD**	**LPD mode select bit** 0: When V_{DD} below LPD voltage, LPDF is set 1: When V_{DD} above LPD voltage, LPDF is set
4	**LPDIF**	**LPD interrupt flag bit** 0: No LPD happened, clear by software 1: LPD happened, set by hardware
3-0	**LPDS[3:0]**	**LPD Voltage Select bit** 0000: 2.40V 0001: 2.55V 0010: 2.70V 0011: 2.85V 0100: 3.00V 0101: 3.15V 0110: 3.30V 0111: 3.45V 1000: 3.60V 1001: 3.75V 1010: 3.90V 1011: 4.05V 1100: 4.20V 1101: 4.35V 1110: 4.50V 1111: 4.65V

*: LPDIF can only be write 0, it can't be set to 1.

8.5 Low Voltage Reset (LVR)

8.5.1 Feature

- Enabled by the code option and V_{LVR} is 4.3V or 2.1V
- LVR de-bounce timer T_{LVR} is 30-60μs
- When the power supply voltage is lower than the set voltage V_{LVR}, it will cause the internal reset

The LVR function is used to monitor the supply voltage and generate an internal reset in the device when the supply voltage below the specified value V_{LVR}. The LVR de-bounce timer T_{LVR} is about 30μs-60μs.

The LVR circuit has the following feature when the LVR function is enabled: (t means the time of the supply voltage below V_{LVR})

Generates a system reset when $V_{DD} \leq V_{LVR}$ and $t \geq T_{LVR}$;

Cancels the system reset when $V_{DD} > V_{LVR}$ or $V_{DD} < V_{LVR}$, but $t < T_{LVR}$.

The LVR function is enabled by the code option.

It is typically used in AC line or large capacity battery applications, where heavy loads may be switched on and cause the MCU supply-voltage temporarily falls below the minimum specified operating voltage.

Low voltage reset can be applied to this, protecting system generates valid reset in the below set voltages.

8.6 Watchdog Timer (WDT) and Reset State

8.6.1 Feature

- Auto detect Program Counter (PC) over range, and generate OVL Reset
- WDT runs even in the Power-Down mode
- Selectable different WDT overflow frequency

OVL Reset

To enhance the anti-noise ability, SH79F1618 built in Program Counter (PC) over range detect circuit, if program counter value is larger than flash rom size, or detect operation code equal to A5H which is not exist in 8051 instruction set, a OVL reset will be generate to reset CPU, and set WDOF bit. So, to make use of this feature, you should fill unused flash rom with A5H.

Watchdog Timer

The watchdog timer is a down counter, and its clock source is an independent built-in RC oscillator, so it always runs even in the Power-Down mode. The watchdog timer will generate a device reset when it overflows. It can be enabled or disabled by the code option.

The watchdog timer control bits (WDT.2-0) are used to select different overflow time. The watchdog timer overflow flag (WDOF) will be automatically set to “1” by hardware when overflow happens. To prevent overflow happen, by reading or writing the WDT register RSTSTAT, the watchdog timer should re-count before the overflow happens.

There are also some reset flags in this register as below:

8.6.2 Register

Table 8.19 Reset Control Register

B1H, Bank0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RSTSTAT	WDOF	-	PORF	LVRF	CLRF	WDT.2	WDT.1	WDT.0
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR)	0	-	1	0	0	0	0	0
Reset Value (WDT)	1	-	u	u	u	0	0	0
Reset Value (LVR)	u	-	u	1	u	0	0	0
Reset Value (PIN)	u	-	u	u	1	0	0	0

Bit Number	Bit Mnemonic	Description
7	**WDOF**	**Watch Dog Timer Overflow or OVL Reset Flag** Set by hardware when WDT overflow or OVL reset happened, cleared by software or Power On Reset 0: Watch Dog not overflows or no OVL reset generated 1: Watch Dog overflow or OVL reset occurred
5	**PORF**	**Power On Reset Flag** Set only by Power On Reset, cleared only by software 0: No Power On Reset. 1: Power On Reset occurred.
4	**LVRF**	**Low Voltage Reset Flag** Set only by Low Voltage Reset, cleared by software or Power On Reset 0: No Low Voltage Reset occurs 1: Low Voltage Reset occurred
3	**CLRF**	**Pin Reset Flag** Set only by pin reset, cleared by software or Power On Reset 0: No Pin Reset occurs 1: Pin Reset occurred
2-0	**WDT[2:0]**	**WDT Overflow period control bit** 000: Overflow period minimal value= 4096 ms 001: Overflow period minimal value= 1024 ms 010: Overflow period minimal value = 256 ms 011: Overflow period minimal value = 128 ms 100: Overflow period minimal value = 64ms 101: Overflow period minimal value = 16ms 110: Overflow period minimal value = 4ms 111: Overflow period minimal value = 1ms ***Notes:*** *If WDT_opt is enable in application, you must clear WatchDog periodically, and the interval must be less than the minimum value listed above.*

Note: *No RST PIN in 20 PIN package, CLRF bit is inactive.*

8.7 Power Management

8.7.1 Feature

- ■Two power saving modes: Idle mode and Power-Down mode
- ■Two ways to exit Idle and Power-Down mode: interrupt and reset

To reduce power consumption, SH79F1620 supplies two power saving modes: Idle mode and Power-Down mode. These two modes are controlled by PCON & SUSLO register.

8.7.2 Idle Mode

In this mode, the clock of CPU is frozen, the program execution is halted, and the CPU will stop at a defined state. But the peripherals continue to be clocked. When entering idle mode, all the CPU status before entering will be preserved. Such as: PSW, PC, SFR & RAM are all retained.

By two consecutive instructions: setting SUSLO register as 0x55, and immediately followed by setting the IDL bit in PCON register, will make SH79F1620 enter Idle mode. If the consecutive instruction sequence requirement is not met, the CPU will clear either SUSLO register or IDL bit in the next machine cycle. And the CPU will not enter Idle mode. The setting of IDL bit will be the last instruction that CPU executed.

There are two ways to exit Idle mode:

(1) An interrupt generated. After warm-up time, the clock of the CPU will be restored, and the hardware will clear SUSLO register and IDL bit in PCON register. Then the program will execute the interrupt service routine first, and then jumps to the instruction immediately following the instruction that activated Idle mode.

(2) Reset signal (logic low on the RESET pin, WDT RESET if enabled, LVR RESET if enabled), this will restore the clock of the CPU, the SUSLO register and the IDL bit in PCON register will be cleared by hardware, finally the SH79F1620 will be reset. And the program will execute from address 0000H. The RAM will keep unchanged and the SFR value might be changed according to different function module.

8.7.3 Power-Down Mode

The Power-Down mode places the SH79F1620 in a very low power state.

When single clock signal input (OP_OSC[3:0] is 0000 or 1110), Power-Down mode will stop all the clocks including CPU and peripherals. When double clock signa input (OP_OSC[3:0] is 0011, 0110, 1010 or 1101), if system clock is 32.768kHz or 128kHzRC, Power-Down mode will stop all the clocks including CPU and peripherals. If high frequency oscillator is used as system clock, 32.768kHz or 128kHzRC clock used in Timer3 will be opened in Power-Down mode. In Power-Down mode, if WDT is enabled, WDT block will keep on working. When entering Power-Down mode, all the CPU status before entering will be preserved. Such as: PSW, PC, SFR & RAM are all retained.

By two consecutive instructions: setting SUSLO register as 0x55, and immediately followed by setting the PD bit in PCON register, will make SH79F1620 enter Power-Down mode. If the consecutive instruction sequence requirement is not met, the CPU will clear either SUSLO register or PD bit in the next machine cycle. And the CPU will not enter Power-Down mode.

The setting of PD bit will be the last instruction that CPU executed.

Note: *If IDL bit and PD bit are set simultaneously, the SH79F1620 enters Power-Down mode. The CPU will not go in Idle mode when exiting from Power-Down mode, and the hardware will clear both IDL & PD bit after exit form Power-Down mode.*

There are three ways to exit the Power-Down mode:

(1) An active external Interrupt (such as INT2, INT3 & INT4) and LPD interrupt will make SH79F1620 exit Power-Down mode. The oscillator will start after interrupt happens, after warm-up time, the clocks of the CPU and peripheral will be restored, the SUSLO register and PD bit in PCON register will be cleared by hardware. Program execution resumes with the interrupt service routine. After completion of the interrupt service routine, the instructions which jumped to enter Power-Down mode will continue to run.

(2) Timer3 interrupt will make SH79F1620 exit Power-Down mode when 32.768kHz or 128kHz RC is the clock source. The oscillator will start after interrupt happens, after warm-up time, the clocks of the CPU and peripheral will be restored, the SUSLO register and PD bit in PCON register will be cleared by hardware. Program execution resumes with the interrupt service routine. After completion of the interrupt service routine, the instructions which jumped to enter Power-Down mode will continue to run.

(3) Reset signal (logic low on the RESET pin, WDT RESET if enabled, LVR RESET if enabled). This will restore the clock of the CPU after warm-up time, the SUSLO register and the PD bit in PCON register will be cleared by hardware, finally the SH79F1620 will be reset. And the program will execute from address 0000H. The RAM will keep unchanged and the SFR value might be changed according to different function module.

Note: *In order to entering Idle/Power-Down, it is necessary to add 3 NOPs after setting IDL/PD bit in PCON.*

8.7.4 Register

Table 8.20 Power Control Register

87H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCON	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	-	-	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**SMOD**	**Baud rate double bit**
6	**SSTAT**	**SCON[7:5] function selection bit**
3-2	**GF[1:0]**	**General purpose flags for software use**
1	**PD**	**Power-Down mode control bit** 0: Cleared by hardware when an interrupt or reset occurs 1: Set by software to activate the Power-Down mode
0	**IDL**	**Idle mode control bit** 0: Cleared by hardware when an interrupt or reset occurs 1: Set by software to activate the Idle mode

Table 8.21 Suspend Mode Control Register

8EH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SUSLO	SUSLO.7	SUSLO.6	SUSLO.5	SUSLO.4	SUSLO.3	SUSLO.2	SUSLO.1	SUSLO.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7-0	**SUSLO[7:0]**	This register is used to control the CPU enter suspend mode (Idle or Power-Down). Only consecutive instructions like below will make CPU enter suspend mode. Other wise the either SUSLO, IDL or PD bit will be cleared by hardware in the next machine cycle.

Example

```
IDLE_MODE:
        MOV     SUSLO, #55H
        ORL     PCON, #01H
        NOP
        NOP
        NOP
```

```
POWERDOWN_MODE:
        MOV     SUSLO, #55H
        ORL     PCON, #02H
        NOP
        NOP
        NOP
```

8.8 Warm-up Timer

8.8.1 Feature

- Built-in power on warm-up counter to eliminate unstable state of power on
- Built-in oscillator warm-up counter to eliminate unstable state when oscillation start up

SH79F1620 has a built-in power warm-up counter; it is designed to eliminate unstable state after power on or to do some internal initial operation such as read internal customer code option etc.

SH79F1620 has also a built-in oscillator warm-up counter, it is designed to eliminate unstable state when oscillator starts oscillating in the following conditions: Power-on reset, Pin reset, LVR reset, Watchdog Reset and Wake up from low power consumption mode.

After power-on, SH79F1620 will start power warm-up procedure first, and then oscillator warm-up procedure. Began to run the program after the overflow.

Power Warm-up Time

Power On Reset/ Pin Reset/ Low Voltage Reset		**WDT Reset (Not in Power-Down Mode)**		**WDT Reset (Wakeup from Power-Down Mode)**		**Wakeup from Power-Down Mode (Only for interrupt)**	
TPWRT**	OSC Warm up*	TPWRT**	OSC Warm up*	TPWRT**	OSC Warm up*	TPWRT**	OSC Warm up*
11ms	YES	≈1ms	NO	≈1ms	YES	≈800us	YES

OSC Warm-up Time

Option: OP_WMT **Oscillator Type**	**00**	**01**	**10**	**11**
Ceramic/Crystal	2^{17} X Tosc	2^{14} X Tosc	2^{11} X Tosc	2^{8} X Tosc
32kHz crystal	2^{13} X Tosc			
Internal RC	2^{7} X Tosc			

8.9 Code Option

OP_WDT:

0: Disable WDT function (default)
1: Enable WDT function

OP_WDTPD:

0: Disable WDT function in Power-Down mode (default)
1: Enable WDT function in Power-Down mode

OP_RST:

0: P5.2 used as RST pin (default)
1: P5.2 used as I/O pin
Note: *No reset pin in 20 pin package, please select code option 1.*

OP_WMT: (unavailable for 32k oscillator or Internal RC)

00: longest warm up time (default)
01: longer warm up time
10: shorter warm up time
11: shortest warm up time

OP_OSC:

0000: Oscillator1 is internal 12M RC, oscillator2 is disabled
0011: Oscillator1 is internal 128k RC, oscillator2 is internal 12M RC
0110: Oscillator1 is internal 128k RC, oscillator2 is 2M-12M crystal/ceramic oscillator
1010: Oscillator1 is 32.768k crystal oscillator, oscillator2 is internal 12M RC
1110: Oscillator1 is 2M-12M crystal/ceramic oscillator, oscillator2 is disabled
Others: Oscillator1 is internal 12M RC, oscillator2 is disabled
Note: *No external oscillator pin in 20 PIN package, OP_OSC only can be set 0000 and 0011.*

OP_LVREN:

0: Disable LVR function (default)
1: Enable LVR function

OP_LVRLE:

0: 4.3V LVR level 1 (default)
1: 2.1V LVR level 2

OP_SCM:

0: SCM is invalid in warm up period (default)
1: SCM is valid in warm up period

OP_OSCDRIVE:

011: 8M-12M crystal (default)
001: 4M crystal
111: 12M ceramic
101: 8M ceramic
110: 4M ceramic
100: 2M ceramic

OP_PORTDRIVE:

0: Port drive ability normal mode
1: Port drive ability large mode (default)

OP_P3.3-P3.0:

0: port3 [3:0] sink ability normal mode
1: port3 [3:0] sink ability large mode (default)

OP_P3.7-P3.4:

0: port3 [7:4] sink ability large mode (default)
1: port3 [7:4] sink ability normal mode

8.10 Programming Note

1. Set the registers as follows in programming:
Register P0SS (B6H, Bank0) must be written 0;
Register P1SS (9CH, Bank0) must be written 0;
Register P2SS (9DH, Bank0) must be written 0;
Register P3SS (9EH, Bank0) must be written 0;
Register DISPCON (ABH, Bank0) must be written 0;
Register BUZCON (BDH, Bank0) must be written 0.

Program example:

```
MOV         P0SS, #00H
MOV         P1SS, #00H
MOV         P2SS, #00H
MOV         P3SS, #00H
MOV         DISPCON, #00H
MOV         BUZCON, #00H
```

2. In order to improve reliability, unused bit of PxCR (x = 0-5) register should be written 1, unused bit of PxPCR (x = 0-5) register should be written 0, unused bit of Px (x = 0-5) register should be written 0.

Example Program (P0):

```
MOV P0CR, # 11XX1XXXB   ; P0CR unused bit write 1, X is user's settings
MOV P0PCR, # 00XX0XXXB; P0PCR unused bit write 1, X is user's settings
MOV P0, # 00XX0XXXB       ; P0 unused bit write 1, X is user's settings
```

9. Instruction Set

ARITHMETIC OPERATIONS				
Opcode	**Description**	**Code**	**Byte**	**Cycle**
ADD A, Rn	Add register to accumulator	0x28-0x2F	1	1
ADD A, direct	Add direct byte to accumulator	0x25	2	2
ADD A, @Ri	Add indirect RAM to accumulator	0x26-0x27	1	2
ADD A, #data	Add immediate data to accumulator	0x24	2	2
ADDC A, Rn	Add register to accumulator with carry flag	0x38-0x3F	1	1
ADDC A, direct	Add direct byte to A with carry flag	0x35	2	2
ADDC A, @Ri	Add indirect RAM to A with carry flag	0x36-0x37	1	2
ADDC A, #data	Add immediate data to A with carry flag	0x34	2	2
SUBB A, Rn	Subtract register from A with borrow	0x98-0x9F	1	1
SUBB A, direct	Subtract direct byte from A with borrow	0x95	2	2
SUBB A, @Ri	Subtract indirect RAM from A with borrow	0x96-0x97	1	2
SUBB A, #data	Subtract immediate data from A with borrow	0x94	2	2
INC A	Increment accumulator	0x04	1	1
INC Rn	Increment register	0x08-0x0F	1	2
INC direct	Increment direct byte	0x05	2	3
INC @Ri	Increment indirect RAM	0x06-0x07	1	3
DEC A	Decrement accumulator	0x14	1	1
DEC Rn	Decrement register	0x18-0x1F	1	2
DEC direct	Decrement direct byte	0x15	2	3
DEC @Ri	Decrement indirect RAM	0x16-0x17	1	3
INC DPTR	Increment data pointer	0xA3	1	4
MUL AB 8 X 8 16 X 8	Multiply A and B	0xA4	1	11 20
DIV AB 8 / 8 16 / 8	Divide A by B	0x84	1	11 20
DA A	Decimal adjust accumulator	0xD4	1	1

LOGIC OPERATIONS				
Opcode	**Description**	**Code**	**Byte**	**Cycle**
ANL A, Rn	AND register to accumulator	0x58-0x5F	1	1
ANL A, direct	AND direct byte to accumulator	0x55	2	2
ANL A, @Ri	AND indirect RAM to accumulator	0x56-0x57	1	2
ANL A, #data	AND immediate data to accumulator	0x54	2	2
ANL direct, A	AND accumulator to direct byte	0x52	2	3
ANL direct, #data	AND immediate data to direct byte	0x53	3	3
ORL A, Rn	OR register to accumulator	0x48-0x4F	1	1
ORL A, direct	OR direct byte to accumulator	0x45	2	2
ORL A, @Ri	OR indirect RAM to accumulator	0x46-0x47	1	2
ORL A, #data	OR immediate data to accumulator	0x44	2	2
ORL direct, A	OR accumulator to direct byte	0x42	2	3
ORL direct, #data	OR immediate data to direct byte	0x43	3	3
XRL A, Rn	Exclusive OR register to accumulator	0x68-0x6F	1	1
XRL A, direct	Exclusive OR direct byte to accumulator	0x65	2	2
XRL A, @Ri	Exclusive OR indirect RAM to accumulator	0x66-0x67	1	2
XRL A, #data	Exclusive OR immediate data to accumulator	0x64	2	2
XRL direct, A	Exclusive OR accumulator to direct byte	0x62	2	3
XRL direct, #data	Exclusive OR immediate data to direct byte	0x63	3	3
CLR A	Clear accumulator	0xE4	1	1
CPL A	Complement accumulator	0xF4	1	1
RL A	Rotate accumulator left	0x23	1	1
RLC A	Rotate accumulator left through carry	0x33	1	1
RR A	Rotate accumulator right	0x03	1	1
RRC A	Rotate accumulator right through carry	0x13	1	1
SWAP A	Swap nibbles within the accumulator	0xC4	1	4

DATA TRANSFERS

Opcode	Description	Code	Byte	Cycle
MOV A, Rn	Move register to accumulator	0xE8-0xEF	1	1
MOV A, direct	Move direct byte to accumulator	0xE5	2	2
MOV A, @Ri	Move indirect RAM to accumulator	0xE6-0xE7	1	2
MOV A, #data	Move immediate data to accumulator	0x74	2	2
MOV Rn, A	Move accumulator to register	0xF8-0xFF	1	2
MOV Rn, direct	Move direct byte to register	0xA8-0xAF	2	3
MOV Rn, #data	Move immediate data to register	0x78-0x7F	2	2
MOV direct, A	Move accumulator to direct byte	0xF5	2	2
MOV direct, Rn	Move register to direct byte	0x88-0x8F	2	2
MOV direct1, direct2	Move direct byte to direct byte	0x85	3	3
MOV direct, @Ri	Move indirect RAM to direct byte	0x86-0x87	2	3
MOV direct, #data	Move immediate data to direct byte	0x75	3	3
MOV @Ri, A	Move accumulator to indirect RAM	0xF6-0xF7	1	2
MOV @Ri, direct	Move direct byte to indirect RAM	0xA6-0xA7	2	3
MOV @Ri, #data	Move immediate data to indirect RAM	0x76-0x77	2	2
MOV DPTR, #data16	Load data pointer with a 16-bit constant	0x90	3	3
MOVC A, @A+DPTR	Move code byte relative to DPTR to A	0x93	1	7
MOVC A, @A+PC	Move code byte relative to PC to A	0x83	1	8
MOVX A, @Ri	Move external RAM (8-bit address) to A	0xE2-0xE3	1	5
MOVX A, @DPTR	Move external RAM (16-bit address) to A	0xE0	1	6
MOVX @Ri, A	Move A to external RAM (8-bit address)	0xF2-F3	1	4
MOVX @DPTR, A	Move A to external RAM (16-bit address)	0xF0	1	5
PUSH direct	Push direct byte onto stack	0xC0	2	5
POP direct	Pop direct byte from stack	0xD0	2	4
XCH A, Rn	Exchange register with accumulator	0xC8-0xCF	1	3
XCH A, direct	Exchange direct byte with accumulator	0xC5	2	4
XCH A, @Ri	Exchange indirect RAM with accumulator	0xC6-0xC7	1	4
XCHD A, @Ri	Exchange low-order nibble indirect RAM with A	0xD6-0xD7	1	4

PROGRAM BRANCHES

Opcode	Description	Code	Byte	Cycle
ACALL addr11	Absolute subroutine call	0x11-0xF1	2	7
LCALL addr16	Long subroutine call	0x12	3	7
RET	Return from subroutine	0x22	1	8
RETI	Return from interrupt	0x32	1	8
AJMP addr11	Absolute jump	0x01-0xE1	2	4
LJMP addr16	Long jump	0x02	3	5
SJMP rel	Short jump (relative address)	0x80	2	4
JMP @A+DPTR	Jump indirect relative to the DPTR	0x73	1	6
JZ rel (not taken) (taken)	Jump if accumulator is zero	0x60	2	3 5
JNZ rel (not taken) (taken)	Jump if accumulator is not zero	0x70	2	3 5
JC rel (not taken) (taken)	Jump if carry flag is set	0x40	2	2 4
JNC rel (not taken) (taken)	Jump if carry flag is not set	0x50	2	2 4
JB bit, rel (not taken) (taken)	Jump if direct bit is set	0x20	3	4 6
JNB bit, rel (not taken) (taken)	Jump if direct bit is not set	0x30	3	4 6
JBC bit, rel (not taken) (taken)	Jump if direct bit is set and clear bit	0x10	3	4 6
CJNE A, direct, rel (not taken) (taken)	Compare direct byte to A and jump if not equal	0xB5	3	4 6
CJNE A, #data, rel (not taken) (taken)	Compare immediate to A and jump if not equal	0xB4	3	4 6
CJNE Rn, #data, rel (not taken) (taken)	Compare immediate to reg. and jump if not equal	0xB8-0xBF	3	4 6
CJNE @Ri, #data, rel (not taken) (taken)	Compare immediate to Ri and jump if not equal	0xB6-0xB7	3	4 6
DJNZ Rn, rel (not taken) (taken)	Decrement register and jump if not zero	0xD8-0xDF	2	3 5
DJNZ direct, rel (not taken) (taken)	Decrement direct byte and jump if not zero	0xD5	3	4 6
NOP	No operation	0	1	1

BOOLEAN MANIPULATION				
Opcode	**Description**	**Code**	**Byte**	**Cycle**
CLR C	Clear carry flag	0xC3	1	1
CLR bit	Clear direct bit	0xC2	2	3
SETB C	Set carry flag	0xD3	1	1
SETB bit	Set direct bit	0xD2	2	3
CPL C	Complement carry flag	0xB3	1	1
CPL bit	Complement direct bit	0xB2	2	3
ANL C, bit	AND direct bit to carry flag	0x82	2	2
ANL C, /bit	AND complement of direct bit to carry	0xB0	2	2
ORL C, bit	OR direct bit to carry flag	0x72	2	2
ORL C, /bit	OR complement of direct bit to carry	0xA0	2	2
MOV C, bit	Move direct bit to carry flag	0xA2	2	2
MOV bit, C	Move carry flag to direct bit	0x92	2	3

10. Electrical Characteristics

Absolute Maximum Ratings*

DC Supply Voltage. . . . .-0.3V to +6.0V

Input/Output Voltage. . . . . GND-0.3V to V_{DD}+0.3V

Operating Ambient Temperature. . . . . -40℃ to +85℃

Storage Temperature. . . . .-55℃ to +125℃

FLASH write/erase operating. . . . . 0℃ to +85℃

***Comments**

Stresses exceed those listed under "**Absolute Maximum Ratings**" may cause permanent damage to this device. These are stress ratings only. Functional operation of this device at these or any other conditions above those indicated in the operational sections of this specification is not implied or intended. Exposure to the absolute maximum rating conditions for extended periods may affect device reliability.

DC Electrical Characteristics (V_{DD} = 2.0V - 5.5V, GND = 0V, T_A = +25°C, unless otherwise specified)

Parameter	Symbol	Min.	Typ.*	Max.	Unit	Condition
Operating Voltage	V_{DD}	2.0	5.0	5.5	V	2MHz ≤ f_{OSC} ≤ 12MHz
Operating Current	I_{OP1}	-	5	10	mA	f_{OSC} = 12MHz, V_{DD} = 5.0V All output pins unload (including all digital input pins unfloating) CPU on (execute NOP instruction), WDT on, all other function block off
	I_{OP2}	-	25	35	µA	f_{OSC} = 128kHz, OSCX off,V_{DD} = 5.0V All output pins unload (including all digital input pins unfloating) CPU on (execute NOP instruction), LVR off, WDT off, all other function block off
Stand by Current (IDLE)	I_{SB1}	-	3	5	mA	f_{OSC} = 12MHz, V_{DD} = 5.0V All output pins unload, CPU off (IDLE), all digital input pins unfloating LVR on, WDT off, all other function block off
	I_{SB2}	-	15	20	µA	f_{OSC} = 128kHz, OSCX off, V_{DD} = 5.0V All output pins unload (including all digital input pins unfloating) (IDLE MODE) LVR on, WDT off, all other function block off
Stand by Current (Power-Down)	I_{SB3}	-	2	10	uA	Osc off, V_{DD} = 5.0V All output pins unload (including all digital input pins unfloating) CPU off (Power-Down), LVR on, WDT off, all other function block off
	I_{SB4}	-	4	15	uA	f_{OSC} = 128kHz, OSCX off, V_{DD} = 5.0V All output pins unload (including all digital input pins unfloating) CPU off (Power-Down), LVR on, WDT off, all other function block off
WDT Current	I_{WDT}	-	1	3	µA	All output pins unload, WDT on, V_{DD} = 5.0V
LPD Current	I_{LPD}	-	-	1	µA	V_{DD} = 2.0 - 5.5V
Input Low Voltage 1	V_{IL1}	GND	-	0.3 X V_{DD}	V	I/O Ports , RXD (RxCON[1:0] = 11)
Input High Voltage 1	V_{IH1}	0.7 X V_{DD}	-	V_{DD}	V	I/O Ports
Input Low Voltage 2	V_{IL2}	GND	-	0.2 X V_{DD}	V	$\overline{RST}$, T4, INT2/3/4, RXD (RxCON[1:0] = 00), V_{DD} = 2.4 - 5.5V
		GND	-	0.4 X V_{DD}	V	RXD (RxCON[1:0] = 01), V_{DD} = 2.4 - 5.5V
		GND	-	0.5 X V_{DD}	V	RXD (RxCON[1:0] = 10), V_{DD} = 2.4 - 5.5V
Input High Voltage 2	V_{IH2}	0.8 X V_{DD}	-	V_{DD}	V	$\overline{RST}$, T4, INT2/3/4, RXD, V_{DD} = 2.4 - 5.5V

(to be continued)

(continue)

Parameter	Symbol	Min.	Typ.*	Max.	Unit	Condition
Input Leakage Current	I_{IL}	-1	-	1	μA	Input pin, V_{IN} = V_{DD} or GND
output Leakage Current	I_{OL}	-1	-	1	μA	Open-drain output, V_{DD} = 5.0V V_{OUT} = V_{DD} or GND
Reset pin pull-up resistor	R_{RPH}	-	30	-	kΩ	V_{DD} = 5.0V, V_{IN} = GND
Pull-up resistor	R_{PH}	-	30	-	kΩ	V_{DD} = 5.0V, V_{IN} = GND
Output High Voltage1	V_{OH1}	V_{DD} - 0.7	-	-	V	I/O port (P3), I_{OH} = -10mA, V_{DD} = 5.0V
Output High Voltage2	V_{OH2}	V_{DD} - 0.7	-	-	V	I/O ports (P0, P1, P2, P4, P5), I_{OH} = -10mA, V_{DD} = 5.0V, OP_PORTDRIVE select normal mode (Code Option)
Output High Voltage3	V_{OH3}	V_{DD} - 0.7	-	-	V	I/O ports (P0, P1, P2, P4, P5), I_{OH} = -20mA, V_{DD} = 5.0V, OP_PORTDRIVE select large mode (Code Option)
Output Low Voltage1	V_{OL1}	-	-	GND + 0.6	V	I/O Ports, (P0, P1, P2, P4, P5) I_{OL} = 15mA, V_{DD} = 5.0V
Output Low Voltage2	V_{OL2}	-	-	GND + 0.6	V	I/O ports (P3), I_{OL} = 15mA, V_{DD} = 5.0V OP_P3.3-P3.0 and OP_P3.7-P3.4 both select normal mode (Code Option)
large drive port sink current capability	I_{OL}	120	140	-	mA	I/O port (P3), V_{DD} = 5.0V, V_{OL} = GND + 1.5V OP_P3.3-P3.0 and OP_P3.7-P3.4 both select large mode (Code Option)

Note:

(1) "" Indicates that the typical value is measured at 5.0V, 25°C, unless otherwise noted.*
(2) maximum current value flowing through V_{DD} 5.0V, 25°C must be less than 150mA.
(3) maximum current value flowing through GND 5.0V, 25°C must be less than 200mA.

A/D Converter Electrical Characteristics (V_{DD} = 3V, GND = 0V, T_A = 25°C, Unless otherwise specified)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Supply Voltage	V_{AD}	2.5	3	3.5	V	
Resolution	N_R	-	10	-	bit	GND ≤ V_{AIN} ≤ V_{REF}
A/D Input Voltage	V_{AIN}	GND	-	V_{REF}	V	
A/D Input Resistor*	R_{AIN}	2	-	-	MΩ	V_{IN} = 3.0V
Recommended impedance of analog voltage source	Z_{AIN}	-	-	10	kΩ	
A/D conversion current	I_{AD}	-	1	3	mA	ADC module operating, V_{DD} = 3.0V
A/D Input current	I_{ADIN}	-	-	10	μA	V_{DD} = 3.0V
Differential linearity error	D_{LE}	-	-	±1	LSB	f_{OSC} = 12MHz, V_{DD} = 3.0V
Integral linearity error	I_{LE}	-	-	±2	LSB	f_{OSC} = 12MHz, V_{DD} = 3.0V
Full scale error	E_F	-	±1	±3	LSB	f_{OSC} = 12MHz, V_{DD} = 3.0V
Offset error	E_Z	-	±0.5	±3	LSB	f_{OSC} = 12MHz, V_{DD} = 3.0V
Total Absolute error	E_{AD}	-	-	±3	LSB	f_{OSC} = 12MHz, V_{DD} = 3.0V
Total Conversion time**	T_{CON}	14	-	-	t_{AD}	10 bit Resolution, V_{DD} = 3.0V, t_{AD} = 1μs

Note:

(1) "" Here the A/D input Resistor is the DC input-resistance of A/D itself.*
*(2) "**" Recommendations ADC connected signal source resistance of less than10kΩ.*

AC Electrical Characteristics (V_{DD} = 2.0V - 5.5V, GND = 0V, T_A = +25°C, f_{OSC} = 12MHz, unless otherwise specified)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Oscillator start time	T_{OSC}	-	-	1	s	f_{OSC} = 128kHz
	T_{OSC}	-	1	2	ms	f_{OSC} = 12MHz
RESET pulse width	t_{RESET}	10	-	-	μs	
WDT RC Frequency	f_{WDT}	-	2	3	kHz	
Frequency Stability (RC)	|Δ F|/F	-	±1	±2	%	RC oscillator: |F - 12MHz|/12MHz (V_{DD} = 2.0 - 5.0V, T_A = -40°C~+85°C)
		-	-	±2	%	RC oscillator: |F - 128kHz|/128kHz (V_{DD} = 2.0 - 5.0V, T_A = 25°C)

Low Voltage Reset Electrical Characteristics (V_{DD} = 2.0V - 5.5V, GND = 0V, T_A = +25°C,unless otherwise specified)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
LVR Voltage1	V_{LVR1}	4.2	4.3	4.4	V	LVR Enable, V_{DD} = 2.0V - 5.5V
LVR Voltage2	V_{LVR2}	2.0	2.1	2.2	V	LVR Enable, V_{DD} = 2.0V - 5.5V
Drop-Down Pulse Width for LVR	T_{LVR}	-	60	-	μs	

12MHz Crystal Electrical Characteristics

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Frequency	F_{12M}	-	12	-	MHz	
Capacitor	C_L	-	12.5	-	pF	

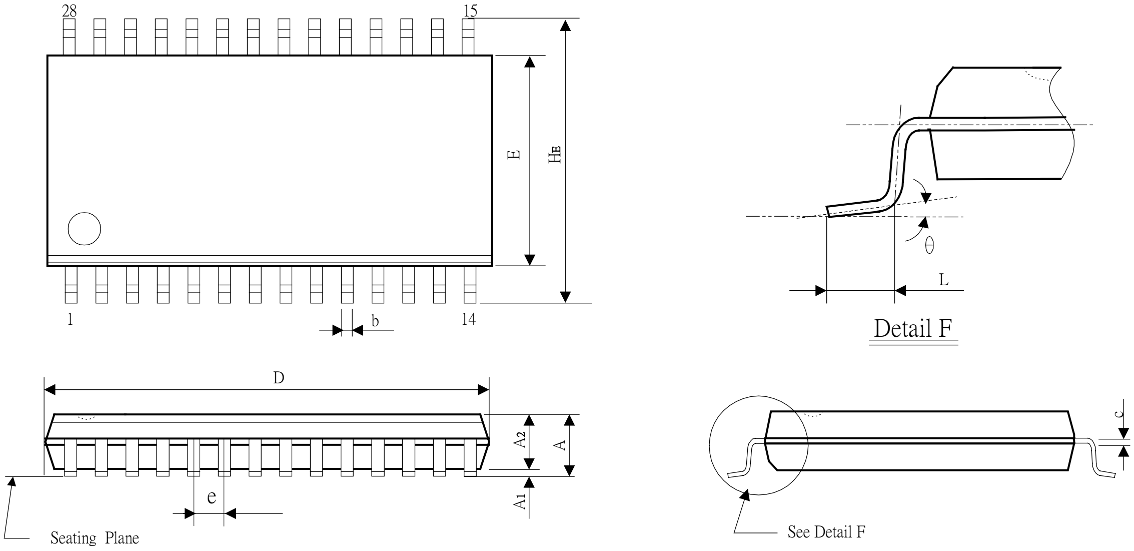
11. Ordering Information

Part No.	Package
SH79F1620M/028MU	SOP28
SH79F1620M/020MU	SOP20
SH79F1620X/020XU	TSSOP20

12. Package Information

SOP28 Outline Dimensions

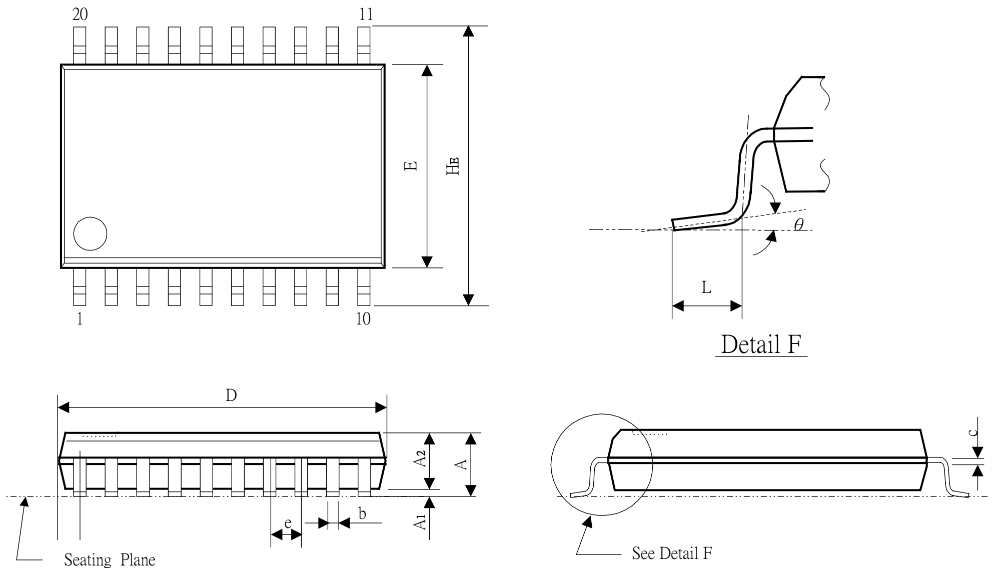
unit: inch/mm



Symbol	Dimensions in inches		Dimensions in mm	
	Min	Max	Min	Max
A	0.085	0.104	2.15	2.65
A1	0.004	0.012	0.10	0.30
A2	0.081	0.098	2.05	2.50
b	0.013	0.02	0.33	0.51
c	0.008	0.014	0.20	0.36
D	0.697	0.715	17.70	18.15
E	0.291	0.3	7.40	7.62
e	0.050(BSC)		1.27(BSC)	
H_E	0.402	0.418	10.21	10.61
L	0.016	0.05	0.40	1.27
θ	0°	8°	0°	8°

SOP20 Outline Dimensions

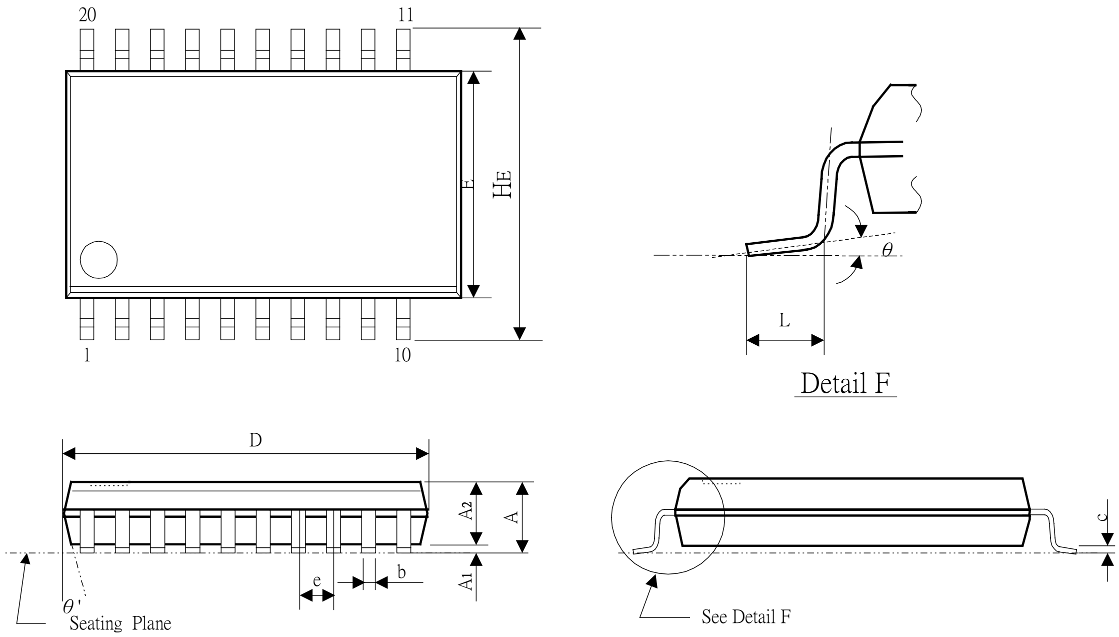
unit: inch/mm



Symbol	Dimensions in inches		Dimensions in mm	
	Min	Max	Min	Max
A	0.093	0.104	2.35	2.65
A1	0.004	0.012	0.10	0.30
A2	0.083	0.098	2.10	2.50
b	0.013	0.020	0.33	0.51
c	0.008	0.013	0.20	0.33
D	0.493	0.516	12.52	13.10
E	0.291	0.299	7.40	7.60
e	0.050(BSC)		1.27(BSC)	
H_E	0.398	0.418	10.11	10.61
L	0.016	0.050	0.40	1.27
θ	0°	8°	0°	8°

TSSOP20 Outline Dimensions

unit: inch/mm



Symbol	Dimensions in inches		Dimensions in mm	
	MIN	MAX	MIN	MAX
A	---	0.048	---	1.2
A1	0.002	0.006	0.05	0.15
A2	0.031	0.041	0.8	1.05
b	0.007	0.012	0.18	0.3
C	0.004	0.008	0.09	0.2
D	0.252	0.26	6.4	6.6
E	0.169	0.177	4.3	4.5
HE	0.246	0.258	6.25	6.55
e	0.026(BSC)		0.65(BSC)	
L	0.018	0.03	0.45	0.75
θ	0°	8°	0°	8°

13. Product SPEC. Change Notice

Version	Content	Date
2.1	Update Package Information	Jul. 2015
2.0	Original	May. 2014

Content

1. FEATURES ... 1
2. GENERAL DESCRIPTION ... 1
3. BLOCK DIAGRAM ... 2
4. PIN CONFIGURATION ... 3
5. PIN DESCRIPTION ... 4
6. SFR MAPPING ... 5
7. NORMAL FUNCTION ... 14
7.1 CPU ... 14
7.1.1 CPU Core SFR ... 14
7.1.2 Enhanced CPU core SFRs ... 15
7.1.3 Register ... 15
7.2 RAM ... 16
7.2.1 Features ... 16
7.3 FLASH PROGRAM MEMORY ... 17
7.3.1 Features ... 17
7.3.2 Flash Operation in ICP Mode ... 19
7.4 SSP FUNCTION ... 20
7.4.1 SSP Register ... 20
7.4.2 Flash Control Flow ... 23
7.4.3 SSP Programming Notice ... 24
7.5 SYSTEM CLOCK AND OSCILLATOR ... 25
7.5.1 Features ... 25
7.5.2 Clock Definition ... 25
7.5.3 Description ... 25
7.5.4 Register ... 26
7.5.5 Oscillator Type ... 27
7.5.6 Capacitor Selection for Oscillator ... 27
7.6 SYSTEM CLOCK MONITOR (SCM) ... 28
7.7 I/O PORT ... 29
7.7.1 Features ... 29
7.7.2 Register ... 29
7.7.3 Port Diagram ... 30
7.7.4 Port Share ... 31
7.8 TIMER ... 33
7.8.1 Features ... 33
7.8.2 Timer2 ... 33
7.8.3 Timer3 ... 37
7.8.4 Timer4 ... 39
7.8.5 Timer5 ... 42
7.9 INTERRUPT ... 44
7.9.1 Feature ... 44
7.9.2 Interrupt Enable Control ... 44
7.9.3 Register ... 44
7.9.4 Interrupt Flag ... 46
7.9.5 Interrupt Vector ... 48
7.9.6 Interrupt Priority ... 48
7.9.7 Interrupt Handling ... 49
7.9.8 Interrupt Response Time ... 49
7.9.9 External Interrupt Inputs ... 50
7.9.10 Interrupt Summary ... 50
8. ENHANCED FUNCTION ... 51
8.1 PWM (PULSE WIDTH MODULATION) ... 51
8.1.1 Feature ... 51
8.1.2 PWM Module Enable ... 51
8.1.3 PWM Timer Lock Register ... 52
8.1.4 12-bit PWM Timer ... 52
8.2 EUART (NO EUART IN 20 PIN PACKAGE) ... 55
8.2.1 Feature ... 55
8.2.2 EUART0 Mode Description ... 55
8.2.3 Baud Rate Generate ... 60

8.2.4 Multi-Processor Communication *60*
8.2.5 Frame Error Detection *61*
8.2.6 Register *62*
8.3 ANALOG DIGITAL CONVERTER (ADC) 65
8.3.1 Feature *65*
8.3.2 ADC Diagram *65*
8.3.3 ADC Register *66*
8.4 LOW POWER DETECT (LPD) 69
8.4.1 Feature *69*
8.4.2 Register *69*
8.5 LOW VOLTAGE RESET (LVR) 70
8.5.1 Feature *70*
8.6 WATCHDOG TIMER (WDT) AND RESET STATE 71
8.6.1 Feature *71*
8.6.2 Register *72*
8.7 POWER MANAGEMENT 73
8.7.1 Feature *73*
8.7.2 Idle Mode *73*
8.7.3 Power-Down Mode *73*
8.7.4 Register *74*
8.8 WARM-UP TIMER 75
8.8.1 Feature *75*
8.9 CODE OPTION 76
8.10 PROGRAMMING NOTE 77
9. INSTRUCTION SET 78
10. ELECTRICAL CHARACTERISTICS 83
11. ORDERING INFORMATION 86
12. PACKAGE INFORMATION 87
13. PRODUCT SPEC. CHANGE NOTICE 90