

MG87FE/L2051/4051/6051

Data Sheet

Version: A1

Features

- 80C51 Central Processing Unit
- MG87FE/L2051 with 2KB flash ROM, 4051/4KB flash ROM; 6051/6KB flash ROM
- Operating voltage: E type: 4.5V~5.5V and L type: 2.4V~3.6V
- Operation frequency : 48MHz(max)@12T and 24MHz@6T mode
 - External crystal mode (Default)
 - Internal RC-oscillator with +/- 1% frequency drift @ 25°C, there are 6 kinds of frequencies selectable (Use only setting by Megawin U1 writer. Please refer U1 writer document).
- ISP memory zone could be optioned as 1KB (Default)
- IAP capability; 1KB IAP memory size
- On-chip 256 bytes data RAM for MG87FE/L2051/4051/6051
- Code protection for flash memory access
- Two 16-bit timer/counter
- PWM-Timer for PWM generator or normal 8-bit timer, selectable interrupt on INT3
- Seven sources, four-level-priority interrupt capability.
- Enhanced UART, provides frame-error detection and hardware address-recognition
- 15 bits Watch-Dog-Timer with 8-bit pre-scalar, one-time enabled by CPU or power-on
- Power control: idle mode and power-down mode, Power-down can be woken-up by INT0(P3.2), INT1(P3.3), INT2(P4.3), INT3(P4.2) and other I/O.
- I/O port list, P1[7:0], P3[7,5:0], P4.2/INT3 on XTAL2, P4.3/INT2 on XTAL1
- Built-in analog comparator with selectable interrupt on INT2. AIN0(V+) on P1.0 and AIN1(V-) on P1.1, output on P3.6
- Package type: PDIP-20, SOP-20

Items	Package Type	Description
MG87Fxy051AE20	PDIP-20	x = E:5.0V , L:3.3V
MG87Fxy051AS20	SOP-20	y = 2, 4, 6 . 2051/4051/6051

Content

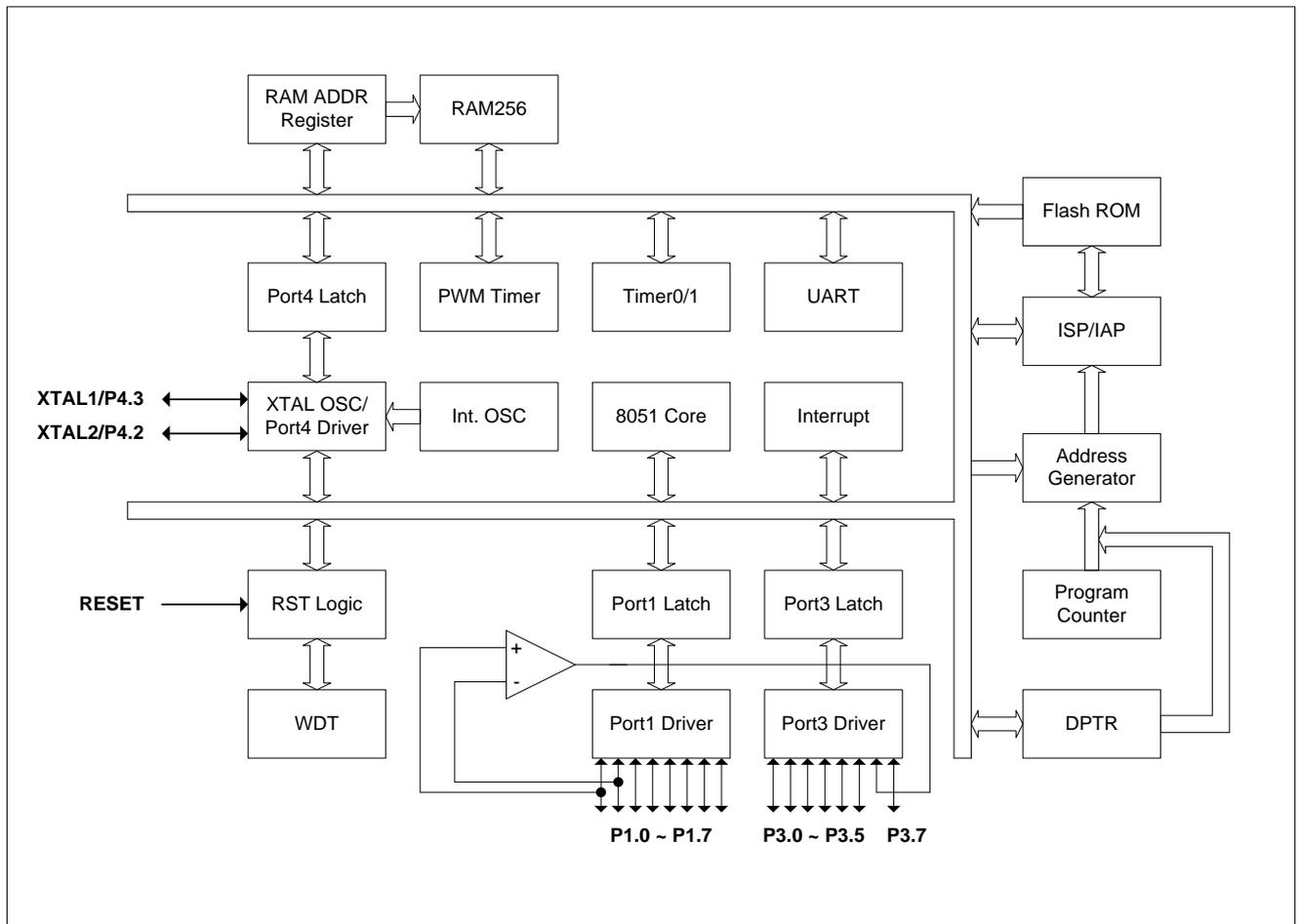
Features.....	3
Content	5
1. General Description.....	7
2. Block Diagram.....	8
3. Pin Configurations	9
3.1. Package Instruction.....	9
3.2. Pin Description (PDIP-20 & SOP-20).....	10
4. 8051 CPU Function Description	11
4.1. CPU Register	11
4.2. CPU Timing.....	11
4.3. CPU Addressing Mode	12
5. Memory Organization	13
5.1. On-Chip Program Flash.....	13
5.2. On-Chip Data RAM.....	14
6. Special Function Register.....	15
6.1. SFR Map.....	15
6.2. SFR Bit Assignment	16
7. Configurable I/O Ports.....	18
7.1. IO Structure.....	18
7.1.1. Port1/3/4 GPIO Structure	18
7.2. Port1 Register	18
7.3. Port3 Register	18
7.4. Port4 Register	19
7.5. GPIO Sample Code.....	20
8. Interrupt.....	21
8.1. Interrupt Structure.....	21
8.2. Interrupt Register.....	22
9. Timers/Counters.....	26
9.1. Timer0 and Timer1	26
9.1.1. Mode 0 Structure.....	26
9.1.2. Mode 1 Structure.....	26
9.1.3. Mode 2 Structure.....	26
9.1.4. Mode 3 Structure.....	27
9.1.5. Timer0/1 Register.....	28
9.2. Timer0/1 Sample Code.....	30
9.3. PWM-Timer	32
9.3.1. PWM-Timer Structure.....	32
9.3.2. PWM-Timer Register.....	33
9.4. PWM Sample Code	35
10. UART	36
10.1. UART Structure	36
10.2. UART Register	37
10.3. Serial Port Sample Code	39
11. Analog Comparator	40
11.1. Analog Comparator Structure	40
11.2. Analog Comparator Register	40
12. Watch Dog Timer (WDT).....	42
12.1. WDT Structure.....	42
12.2. WDT Register.....	42

12.3.WDT Sample Code.....	44
13. Reset.....	45
13.1.Reset Source.....	45
14. Power Management	46
14.1.Power Saving Mode	46
14.1.1. Idle Mode	46
14.1.2. Power-down Mode	46
14.1.3. Interrupt Recovery from Power-down	46
14.1.4. Reset Recovery from Power-down.....	47
14.1.5. GPIO wake-up Recovery from Power-down	47
14.2.Power Control Register.....	47
15. System Clock	49
15.1.Clock Structure.....	49
15.2.Clock Register	49
16. In System Programming (ISP).....	51
17. In Application Programming (IAP)	53
17.1.Demo Code for IAP	54
18. Auxiliary SFRs.....	56
19. Option Setting	58
20. Absolute Maximum Rating.....	59
21. Electrical Characteristics	60
21.1.DC Characteristics.....	60
22. Package Dimension	61
23. Instruction Set	63
24. Revision History	66

1. General Description

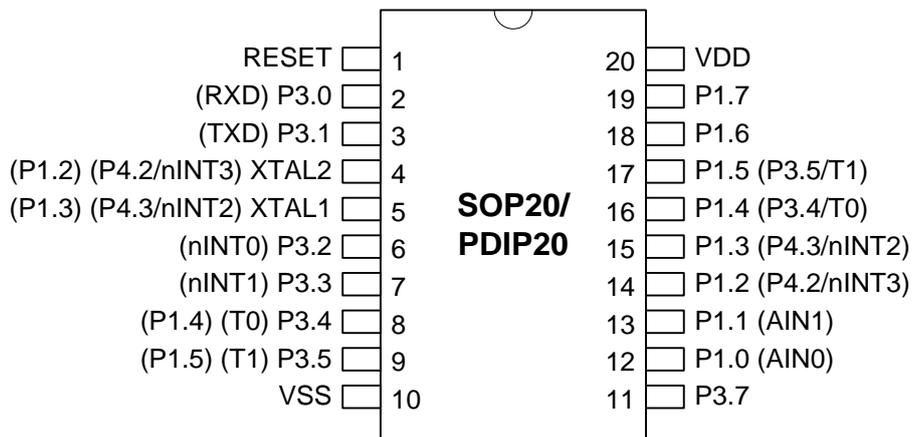
MG87FE/L2051/4051/6051 is single-chip 8-bits microcontroller with the instruction sets fully compatible with industrial-standard 80C51 series microcontroller. 2K/4K/6K bytes flash memory and 256 bytes RAM has been embedded to provide widely field application. In-System-Programming and In-Application-Programming allows the users to download new code or data while the microcontroller sits in the application. This device executes one machine cycle in 6 clock cycles or 12 clock cycles. MG87FE/L2051/4051/6051 has one 8-bit I/O ports (P1), one 7-bit I/O port (P30~P35,P37), two 16-bit timer/counters, one PWM-timer for 8-channel PWM output, a seven -source, four-priority-level interrupt structure, an enhanced UART, a precision analog comparator, on-chip crystal oscillator(combined P42,P43) and a high-precision internal oscillator.

2. Block Diagram



3. Pin Configurations

3.1. Package Instruction



3.2. Pin Description (PDIP-20 & SOP-20)

Pin Name	Pin Number	I/O type	Description
P1.0~P1.7	12~19	I/O	<p>Port1: General-purposed I/O with weak pull-up resistance inside. When 1s are written into Port1, the strong output driving PMOS only turn-on two periods and then the weak pull-up resistance keeps the port high.</p> <p>P1.0 is the comparator positive input. P1.1 is the comparator negative input. P1.2 has a swapped function with P4.2/INT3. P1.3 has a swapped function with P4.3/INT2. P1.4 has a swapped function with P3.4/T0. P1.5 has a swapped function with P3.5/T1.</p>
P3.0~P3.7	2~3,6~9,11	I/O	<p>Port3: General-purposed I/O with weak pull-up resistance inside. When 1s are written into Port1, the strong output driving PMOS only turn-on two periods and then the weak pull-up resistance to keep the port high. Port3 also serves the special function of MG87FE/L2051/4051/6051.</p> <p>P3.4 has a swapped function with P1.4. P3.5 has a swapped function with P1.5.</p>
RESET	1	I	RESET: A high on this pin for at least two machine cycles will reset the device.
XTAL1/P4.3	5	I/O	<p>Crystal1: Input to the inverting oscillator amplifier.</p> <p>XTAL1 has an alternate function for P4.3/INT2. P4.3/INT2 has a swapped function with P1.3.</p>
XTAL2/P4.2	4	I/O	<p>Crystal2: Output from the inverting amplifier.</p> <p>XTAL2 has an alternate function for P42/INT3. P4.2/INT3 has a swapped function with P1.2.</p>
VDD	20	P	POWER
VSS	10	G	GROUND

4. 8051 CPU Function Description

4.1. CPU Register

PSW: Program Status Word

Address=D0H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	F1	P

CY: Carry bit.

AC: Auxiliary carry bit.

F0: General purpose flag 0.

RS1: Register bank select bit 1.

RS0: Register bank select bit 0.

OV: Overflow flag.

F1: General purpose flag 1.

P: Parity bit.

The program status word(PSW) contains several status bits that reflect the current state of the CPU. The PSW, shown above, resides in the SFR space. It contains the Carry bit, the Auxiliary Carry(for BCD operation), the two register bank select bits, the Overflow flag, a Parity bit and two user-definable status flags.

The Carry bit, other than serving the function of a Carry bit in arithmetic operations, also serves as the “Accumulator” for a number of Boolean operations.

The bits RS0 and RS1 are used to select one of the four register banks shown in the on-chip-data-RAM section. A number of instructions refer to these RAM locations as R0 through R7.

The Parity bit reflects the number of 1s in the Accumulator. P=1 if the Accumulator contains an odd number of 1s and otherwise P=0.

SP: Stack Pointer

Address=81H, read/write, Power On + RESET=0000-0111

7	6	5	4	3	2	1	0
SP[7]	SP[6]	SP[5]	SP[4]	SP[3]	SP[2]	SP[1]	SP[0]

DPL: Data Pointer Low

Address=82H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
DPL[7]	DPL[6]	DPL[5]	DPL[4]	DPL[3]	DPL[2]	DPL[1]	DPL[0]

DPH: Data Pointer High

Address=83H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
DPH[7]	DPH[6]	DPH[5]	DPH[4]	DPH[3]	DPH[2]	DPH[1]	DPH[0]

B: B Register

Address=F0H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
B[7]	B[6]	B[5]	B[4]	B[3]	B[2]	B[1]	B[0]

4.2. CPU Timing

A machine cycle is the shortest timing period to achieve an instruction. In MG87FE/L2051/4051/6051, some instructions need 1 machine cycle to achieve, but others need 2 or 4 machine cycles. A machine cycle takes 12 clock periods or 6 clock periods. For 12MHz system clock, it is 1us or 0.5us.

A machine cycle is consisted of six sequential states. The states are from S1 to S6. For each state, it is partitioned into two phase – phase1 and phase2. Each phase is corresponding to 1 clock period. Execution of a one-cycle instruction begins during S1 when the op-code is latched into the instruction register. A second fetch appears during S4 of the same machine cycle. Execution is completed at the end of S6 of the machine cycle.

MOVX instruction is in-active in MG87FE/L2051/4051/6051 because there is no on-chip external RAM and no external access bus. Write operation will have no effect. And read operation will always cause an un-expected operation.

4.3. CPU Addressing Mode

Direct Addressing (DIR)

In direct addressing the operand is specified by an 8-bit address field in the instruction. Only internal data RAM and SFRs can be direct addressed.

Indirect Addressing (IND)

In indirect addressing the instruction specified a register which contains the address of the operand. Both internal and external RAM can be indirectly addressed.

The address register for 8-bit addresses can be R0 or R1 of the selected bank, or the Stack Pointer. The address register for 16-bit addresses can only be the 16-bit data pointer register – DPTR.

Register Instruction (REG)

The register banks, containing registers R0 through R7, can be accessed by certain instructions which carry a 3-bit register specification within the op-code of the instruction. Instructions that access the registers this way are code efficient because this mode eliminates the need of an extra address byte. When such instruction is executed, one of the eight registers in the selected bank is accessed.

Register-Specific Instruction

Some instructions are specific to a certain register. For example, some instructions always operate on the accumulator or data pointer, etc. No address byte is needed for such instructions. The op-code itself does it.

Immediate Constant (IMM)

The value of a constant can follow the op-code in the program memory.

Index Addressing

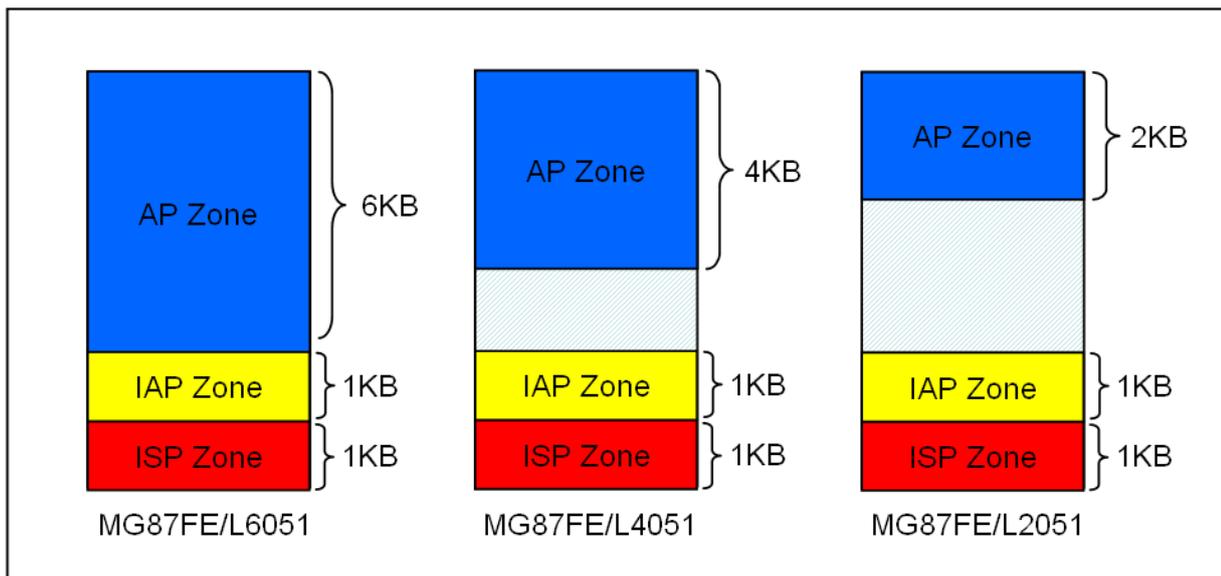
Only program memory can be accessed with indexed addressing and it can only be read. This addressing mode is intended for reading look-up tables in program memory. A 16-bit base register(either DPTR or PC) points to the base of the table, and the accumulator is set up with the table entry number. Another type of indexed addressing is used in the conditional jump instruction.

In conditional jump, the destination address is computed as the sum of the base pointer and the accumulator.

5. Memory Organization

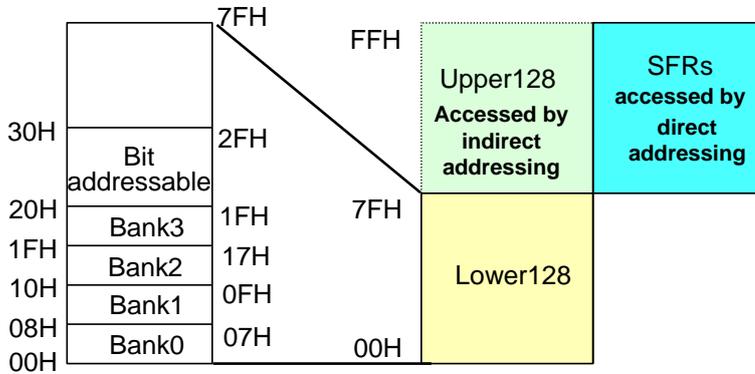
MG87FE/L2051/4051/6051 device has separate address spaces for program and data memory. On-chip data memory can be accessed by 8-bit addresses, which can be quickly stored and manipulated by the 8-bit CPU. Program memory in MG87FE/L2051/4051/6051 can only be read, not written into.

5.1. On-Chip Program Flash



In MG87FE/L2051/4051/6051, the first partition named AP-memory is the space for storing user's application program code. The second one named as IAP-memory and the space which is accessed by CPU for storing the user data. The third is named as ISP-memory and it is special for ISP boot code program.

5.2. On-Chip Data RAM



MG87FE/L2051/4051/6051 has internal data RAM that is mapped into three separate segments. They are lower 128 bytes of RAM, upper 128 bytes of RAM and 128 bytes Special Function Register (SFR).

6.2.1 Lower 128 bytes of RAM: (addresses 0x00 to 0x7F) are accessed by either direct or indirect addressing.

6.2.2 Upper 128 bytes of RAM: (addresses 0x80 to 0xFF) are accessed only by indirect addressing (using R0 or R1).

6.2.3 The Special Function Registers: (addresses 0x80 to 0xFF) are accessed only by direct addressing.

6. Special Function Register

6.1. SFR Map

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8H			CCAP0H 00000000						FFH
F0H	B 00000000								F7H
E8H	P4 XXXX11XX		CCAP0L 00000000						EFH
E0H		WDTCR 0X000000	IFD 11111111	IFADRH 00000000	IFADRL 00000000	IFMT XXXX0000	SCMD XXXXXXXXXX	ISPCR 0000XXXX	E7H
D8H	CCON 00XXXXXX	CMOD 00000000							DFH
D0H	PSW 00000000						P3WKPE 0X000000	P1WKPE 00000000	D7H
C8H									CFH
C0H	XICON 00000000							CKCON XXXXX000	C7H
B8H	IPL XXX00000	SADEN 00000000						CKCON2 XX001010	BFH
B0H	P3 11111111							IPH 00X00000	B7H
A8H	IE 0XX00000	SADDR 00000000							AFH
A0H			AUXR1 0XXX0XXX						A7H
98H	SCON 00000000	SBUF XXXXXXXXXX							9FH
90H	P1 11111111			TSTWD 0X000000				ACSR 0XX00000	97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR 00000000	CKCON3 XXXXXXXX0X	8FH
80H		SP 00000111	DPL 00000000	DPH 00000000				PCON 00010000	87H
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	

6.2. SFR Bit Assignment

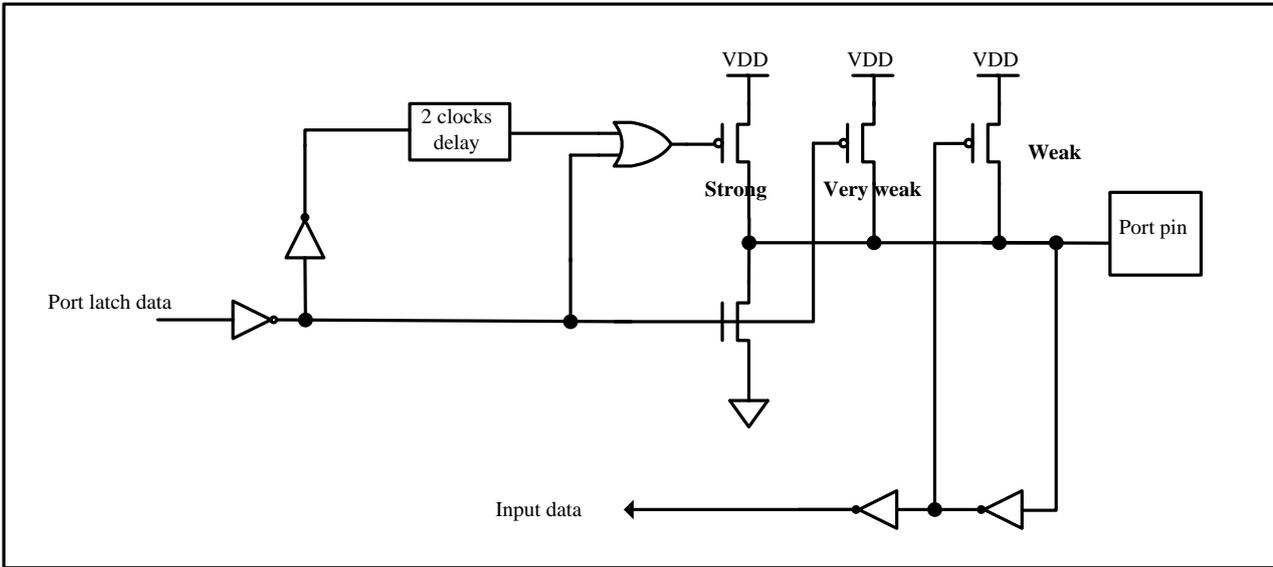
SYMBOL	DESCRIPTION	ADDRESS	BIT ADDRESS AND SYMBOL								LSB	INITIAL VALUE
SP	Stack Pointer	81H										00000111B
DPL	Data Pointer Low	82H										00000000B
DPH	Data Pointer High	83H										00000000B
PCON	Power Control	87H	SMOD	SMOD0	PWMEN	POF	GF1	GF0	PD	IDL		00010000B
TCON	Timer Control	88H	9FH TF1	9EH TR1	9DH TF0	9CH TR0	9BH IE1	9AH IT1	89H IE0	88H IT0		00000000B
TMOD	Timer Mode	89H	GATE	C/T	M1	M0	GATE	C/T	M1	M0		00000000B
TL0	Timer Low 0	8AH										00000000B
TL1	Timer Low 1	8BH										00000000B
TH0	Timer High 0	8CH										00000000B
TH1	Timer High 1	8DH										00000000B
AUXR	Auxiliary	8EH	INT3H	INT2H	P15FS	P14FS	P13FS	P12FS	P11PU	P10PU		00000000B
CKCON3	Clock Control 3	8FH	-	-	-	-	-	-	PWDEX	EN6TR		xxxxxx0xB
P1	Port 1	90H	97H	96H	95H	94H	93H	92H	91H	90H		11111111B
ACSR	Analog Comp. Reg.	97H	ACIDX	-	-	ACF	ACEN	ACM2	ACM1	ACM0		0xx00000B
SCON	Serial Control	98H	9FH SM0 /FE	9EH SM1	9DH SM2	9CH REN	9BH TB8	9AH RB8	99H TI	98H RI		00000000B
SBUF	Serial Buffer	99H										xxxxxxxB
AUXR1	Auxiliary 1	A2H	P14FD	-	-	-	GF2	-	-	-		0xxx0xxxB
IE	Interrupt Enable	A8H	AFH EA	AEH EAC	ADH	ACH ES	ABH ET1	AAH EX1	A9H ET0	A8H EX0		0xx00000B
SADDR	Slave Address	A9H										00000000B
P3	Port 3	B0H	B7H	B6H	B5H T1	B4H T0	B3H INT1	B2H INT0	B1H TXD	B0H RXD		1x111111B
IPH	Interrupt Priority High	B7H	PX3H/ PTCH	PX2H/ PACH		PSH	PT1H	PX1H	PT0H	PX0H		00x00000B
IPL	Interrupt Priority Low	B8H	BFH	BEH PAC	BDH	BCH PS	BBH PT1	BAH PX1	B9H PT0	B8H PX0		xxx00000B
SADEN	Slave Address Mask	B9H										00000000B
CKCON2	Clock Control 2	BFH	OSCDR	EN6TR	XCKS5	XCKS4	XCKS3	XCKS2	XCKS1	XCKS0		xx001010B
XICON	Ext. Interrupt Control	C0H	C7H PX3/ PTC	C6H EX3	C5H IE3	C4H IT3	C3H PX2	C2H EX2	C1H IE2	C0H IT2		00000000B
CKCON	Clock Control	C7H	-	-	-	-	-	SCKS2	SCKS1	SCKS0		xxxxx000B
PSW	Program Status Word	D0H	D7H	D6H	D5H	D4H	D3H	D2H	D1H	D0H		00000000B
P3WKPE	P3 Wake-up Enable	D6H	P37WE		P35WE	P34WE	P33WE	P32WE	P31WE	P30WE		0x000000B
P1WKPE	P1 Wake-up Enable	D7H	P17WE	P16WE	P15WE	P14WE	P13WE	P12WE	P11WE	P10WE		00000000B
CCON	Counter Control Reg.	D8H	CF	CR	-	-	-	-	-	-		00xxxxxxB
CMOD	Counter Mode Reg.	D9H	CIDL	POS2	POS1	POS0	CPS2	CPS1	CPS0	ECF		00000000B
WDTCR	Watch-dog-timer Control register	E1H	WRF	-	ENW	CLW	WIDL	PS2	PS1	PS0		0x000000B
IFD	ISP Flash data	E2H										11111111B
IFADRH	ISP Flash address High	E3H										00000000B
IFADRL	ISP Flash Address Low	E4H										00000000B
IFMT	ISP Mode Table	E5H	-	-	-	-	MS3	MS2	MS1	MS0		xxxx0000B
IAPLB	IAP Low Boundary	Note 1										11111111B
SCMD	ISP Serial Command	E6H										xxxxxxxB
ISPCR	ISP Control Register	E7H	ISPEN	BS	SRST	CFAIL	-	-	-	-		0000xxxxB
P4	Port 4	E8H	-	-	-	-	EBH	EAH				xxxx11xxB
CCAP0L		EAH										00000000B
B	B Register	F0H	F7H	F6H	F5H	F4H	F3H	F2H	F1H	F0H		00000000B
CCAP0H		FAH										00000000B

Note1: The registers are addressed by IFMT and SCMD. Please refer the IFMT register description for more detail information.

7. Configurable I/O Ports

7.1. IO Structure

7.1.1. Port1/3/4 GPIO Structure



By the way, the pull-up resistor is disabled on P10/P11 in default.

7.2. Port1 Register

P1: Port 1 Register

Address=90H, read/write, Power On + RESET=1111-1111

7	6	5	4	3	2	1	0
P17	P16	P15	P14	P13	P12	P11	P10

Bit 7~0: P17~P10 could be set/cleared by CPU. Or it also can be toggled on addressed port channel by PWM-Timer underflow event in PWM mode.

7.3. Port3 Register

P3: Port 3 Register

Address=B0H, read/write, Power On + RESET=1X11-1111

7	6	5	4	3	2	1	0
P37	P36	P35	P34	P33	P32	P31	P30

Bit 7, 5~0: P37, P35~P30 could only be set/cleared by CPU. P36 is read only for CPU from analog comparator output.

7.4. Port4 Register

P4: Port 4 Register

Address=E8H, read/write, Power On + RESET=XXXX-11XX

7	6	5	4	3	2	1	0
-	-	-	-	P43	P42	-	-

Bit 3~2: P43~P42 could be only be set/cleared by CPU.

7.5. GPIO Sample Code

(1). Required Function: Set P1.0 to “1”

Assembly Code Example:

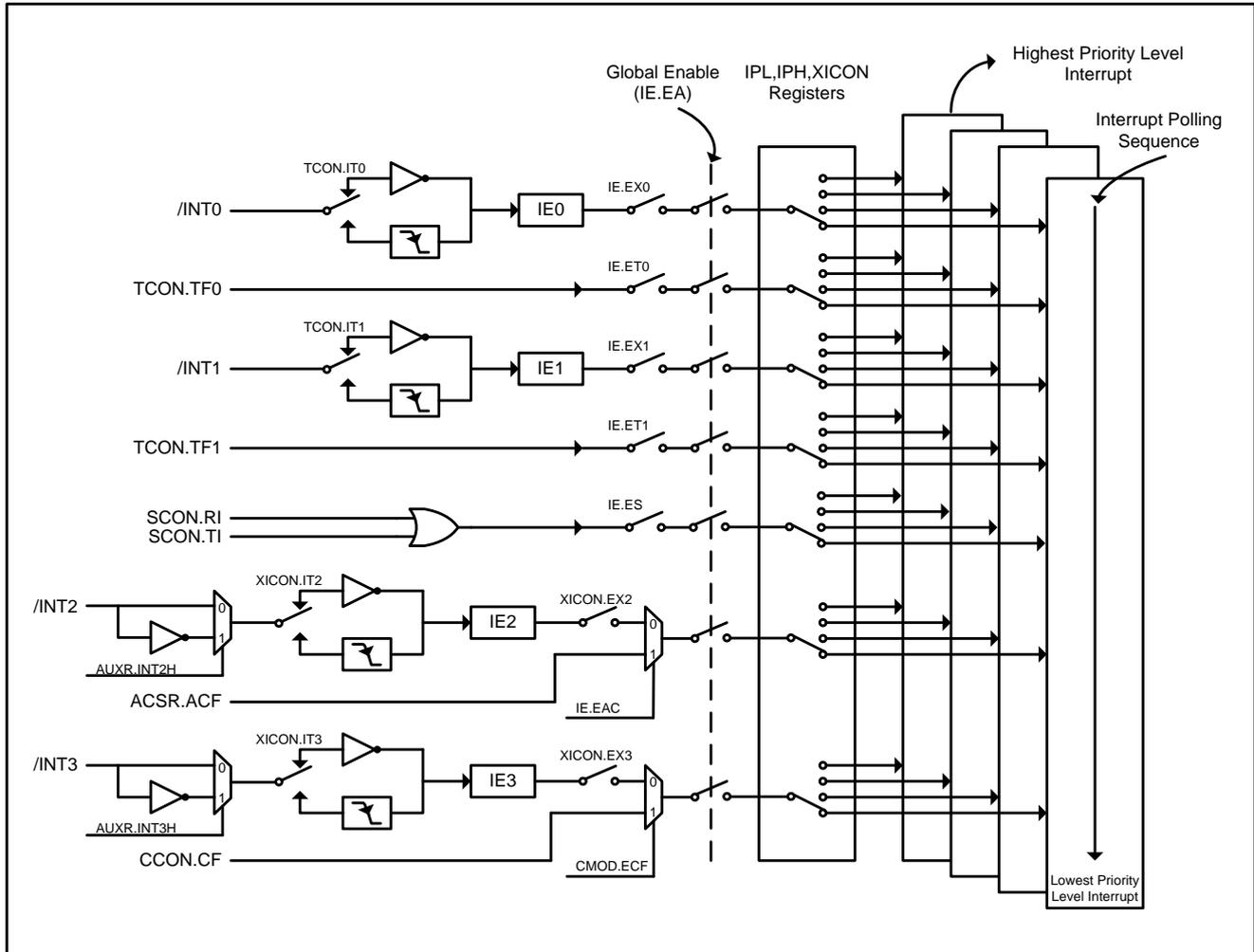
```
SETB    P1.0                ; Set P1.0 data latch to “1”
```

C Code Example:

```
P10 = 1;                    // Set P1.0 data latch to “1”
```

8. Interrupt

8.1. Interrupt Structure



8.2. Interrupt Register

IE: Interrupt Enable Register

Address=E8H, read/write, Power On + RESET=00X0-0000

7	6	5	4	3	2	1	0
EA	EAC	--	ES	ET1	EX1	ET0	EX0

Bit 7: EA, All interrupts enable register.

0: Global disables all interrupts.

1: Global enables all interrupts.

Bit 6: EAC, Analog Comparator interrupt Enable register.

0: Disable analog comparator interrupt and reserve the interrupt vector (33H) to /INT2.

1: Enable analog comparator interrupt and occupy the /INT2 interrupt vector (33H) for analog comparator event. In this mode, IE2 still maintains its original function but it will not generate an interrupt whether EX2 is set or not.

Bit 5: Reserved.

Bit 4: ES, Serial port interrupt enable register.

0: Disable serial port interrupt.

1: Enable serial port interrupt.

Bit 3: ET1, Timer 1 interrupt enable register.

0: Disable Timer 1 interrupt.

1: Enable Timer 1 interrupt.

Bit 2: EX1, External interrupt 1 enable register.

0: Disable external interrupt 1.

1: Enable external interrupt 1.

Bit 1: ET0, Timer 0 interrupt enable register.

0: Disable Timer 0 interrupt.

1: Enable Timer 1 interrupt.

Bit 0: EX0, External interrupt 0 enable register.

0: Disable external interrupt 0.

1: Enable external interrupt 1.

XICON: External Interrupt Control Register

Address=C0H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
PX3/PTC	EX3	IE3	IT3	PX2	EX2	IE2	IT2

Bit 7: PX3, External interrupt 3 priority-L register.

0: Lower priority, setting with PX3H to select priority level.

1: Higher priority, setting with PX3H to select priority level.

Bit 7: PTC, PWM-Timer interrupt priority-L register. PX3 has an alternated function, PTC, that is switched by ECF.

When ECF is set, this bit is PTC function. If cleared, this bit is PX3 function.

0: Lower priority, setting with PTCH to select priority level.

1: Higher priority, setting with PTCH to select priority level.

Bit 6: EX3, external interrupt 3 enable register.

0: Disable external /INT3.

1: Enable external /INT3. This function will be masked when CMOD.ECF is enabled.

Bit 5: IE3, Interrupt 3 Edge flag.

0: Cleared when interrupt start to be serviced. It also could be cleared by CPU.
 1: Set by hardware when external interrupt edge detected. It also could be set by CPU.

Bit 4: IT3, Interrupt 3 type control bit.

0: Cleared by CPU to specify low level triggered on Interrupt 3. If AUXR.INT3H is set, this bit specifies high level triggered on /INT3.
 1: Set by CPU to specify falling edge triggered on Interrupt 3. If AUXR.INT3H is set, this bit specifies rising edge triggered on /INT3.

Bit 3: PX2, External interrupt 2 priority-L register.

0: Lower priority, setting with PX2H to select priority level.
 1: Higher priority, setting with PX2H to select priority level.

Bit 2: EX2, external interrupt 2 enable register.

0: Disable external interrupt 2.
 1: Enable external interrupt 2. This function will be masked when IE.EAC is enabled.

Bit 1: IE2, Interrupt 2 Edge flag.

0: Cleared when interrupt start to be serviced. It also could be cleared by CPU.
 1: Set by hardware when external interrupt edge detected. It also could be set by CPU.

Bit 0: IT2, Interrupt 2 type control bit.

0: Cleared by CPU to specify low level triggered on /INT2. If AUXR.INT2H is set, this bit specifies high level triggered on /INT2.
 1: Set by CPU to specify falling edge triggered on /INT2. If AUXR.INT2H is set, this bit specifies rising edge triggered on /INT2.

IPL: Interrupt Priority Low Register

Address=B8H, read/write, Power On + RESET=X0X0-0000

7	6	5	4	3	2	1	0
-	PAC	--	PS	PT1	PX1	PT0	PX0

Bit 7: reserved.

Bit 6: PAC, Analog Comparator interrupt priority-L register.

Bit 5: Reserved.

Bit 4: PS, Serial port interrupt priority-L register.

Bit 3: PT1, Timer 1 interrupt priority-L register.

Bit 2: PX1, external interrupt 1 priority-L register.

Bit 1: PT0, Timer 0 interrupt priority-L register.

Bit 2: PX0, external interrupt 0 priority-L register.

IPH: Interrupt Priority High Register

Address=B7H, read/write, Power On + RESET=00X0-0000

7	6	5	4	3	2	1	0
PX3H/PTCH	PX2H/PACH	--	PSH	PT1H	PX1H	PT0H	PX0H

Bit 7: PX3H/PTCH, external interrupt 3 priority-H register. It has an alternate function for PWM-Timer interrupt priority-H register when CMOD.ECF is enabled.

Bit 6: PX2H/PACH, external interrupt 2 priority-H register. It has an alternate function for Analog Comparator interrupt priority-H register when IE.EAC is enabled.

Bit 5: Reserved.

Bit 4: PSH, Serial port interrupt priority-H register.

Bit 3: PT1H, Timer 1 interrupt priority-H register.

Bit 2: PX1H, external interrupt 1 priority-H register.

Bit 1: PT0H, Timer 0 interrupt priority-H register.

Bit 2: PX0H, external interrupt 0 priority-H register.

IPL (or XICON) and IPH are combined to form 4-level priority interrupt as the following table.

{IPH.x , IPL.x}	Priority Level
11	1 (highest)
10	2
01	3
00	4

There are seven interrupt sources available in MG87FE/L2051/4051/6051. Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the SFR named IE. This register also contains a global disable bit(EA), which can be cleared to disable all interrupts at once.

Each interrupt source has two corresponding bits to represent its priority. One is located in SFR named IPH and the other in IPL (or XICON) register. Higher-priority interrupt will be not interrupted by lower-priority interrupt request. If two interrupt requests of different priority levels are received simultaneously, the request of higher priority is serviced. If interrupt requests of the same priority level are received simultaneously, an internal polling sequence determine which request is serviced. The following table shows the internal polling sequence in the same priority level and the interrupt vector address.

Source	Vector address	Priority within level
External interrupt 0	03H	1 (highest)
Timer 0	0BH	2
External interrupt 1	13H	3
Timer1	1BH	4
Serial Port	23H	5
-	2BH	-
External interrupt 2 or Comparator	33H	6
External interrupt 3 or PWM-Timer	3BH	7

The external interrupt /INT0, /INT1, /INT2 and /INT3 can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in register TCON, IT2 and IT3 in register XICON. The flags that actually generate these interrupts are bits IE0 and IE1 in TCON, IE2 and IE3 in XICON. When an external interrupt is generated, the flag that generated it is cleared by the hardware when the service routine is vectored to *only if the interrupt was transition –activated*, then the external requesting source is what controls the request flag, rather than the on-chip hardware.

The Timer0 and Timer1 interrupts are generated by TF0 and TF1, which are set by a rollover in their respective Timer/Counter registers in most cases. When a timer interrupt is generated, the flag that generated it is cleared by the on-chip hardware when the service routine is vectored to.

The serial port interrupt is generated by the logical OR of RI and TI. Neither of these flags is cleared by hardware when the service routine is vectored to. The service routine should poll RI and TI to determine which one to request service and it will be cleared by software.

/INT2 and Analog Comparator share the same interrupt vector, 33H. If IE.EAC is enabled, the interrupt vector, 33H will be used for Analog Comparator with the interrupt flag, ACSR.ACF, and IE2 will not be cleared when 33H interrupt vector is addressed to.

If IE.EAC is disabled, the interrupt vector, 33H, will be used for /INT2 and the interrupt flag, XICON.IE2, that will be cleared when EX2 is enabled and the interrupt vector is addressed to.

/INT3 and PWM-Timer share the same interrupt vector, 3BH. If CMOD.ECF is enabled, the interrupt vector, 3BH will be used for PWM-Timer with the interrupt flag, CCON.CF, and IE3 will not be cleared when 3BH interrupt vector is addressed to.

If CMOD.ECF is disabled, the interrupt vector, 3BH, will be used for /INT3 and the interrupt flag, XICON.IE3, that will be cleared when EX3 is enabled and the interrupt vector is addressed to.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it had been set or cleared by hardware. In other words, interrupts can be generated or pending interrupts can be canceled in software.

How hardware see the interrupts

Each interrupt flag is sampled at S5P2 of every machine cycle. The samples are polled during the next S5P2. If one of the flags was in a set condition at S5P2 of the first cycle, the second cycle (polling cycle) will find it and the interrupt system will generate an hardware LCALL to the appropriate service routine as long as it is not blocked by any of the following conditions.

Block conditions:

- An interrupt of equal or higher priority level is already in progress.
- The current cycle (polling cycle) is not the final cycle in the execution of the instruction in progress.
- The instruction in progress is RETI or any write to the IE, IP or IPH registers.

Any of these three conditions will block the generation of the hardware LCALL to the interrupt service routine. Condition 2 ensures that the instruction in progress will be completed before vectoring into any service routine. Condition 3 ensures that if the instruction in progress is RETI or any access to IE or IP, then at least one or more instruction will be executed before any interrupt is vectored to.

The polling cycle is repeated with each machine cycle, and the values polled are the values that were present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but not being responded to for one of the above conditions, if the flag is not still active when the blocking condition is removed, the denied interrupt will not be serviced. In other words, the interrupt flag was once active but not being responded to for one of the above conditions, if the flag is not still active when the blocking condition is removed, the denied interrupt will not be serviced. The interrupt flag was once active but not serviced is not kept in memory. Each polling cycle is new.

9. Timers/Counters

MG87FE/L2051/4051/6051 has two Timers/Counters: Timer 0 and Timer 1. All of them can be configured as timers or event counters.

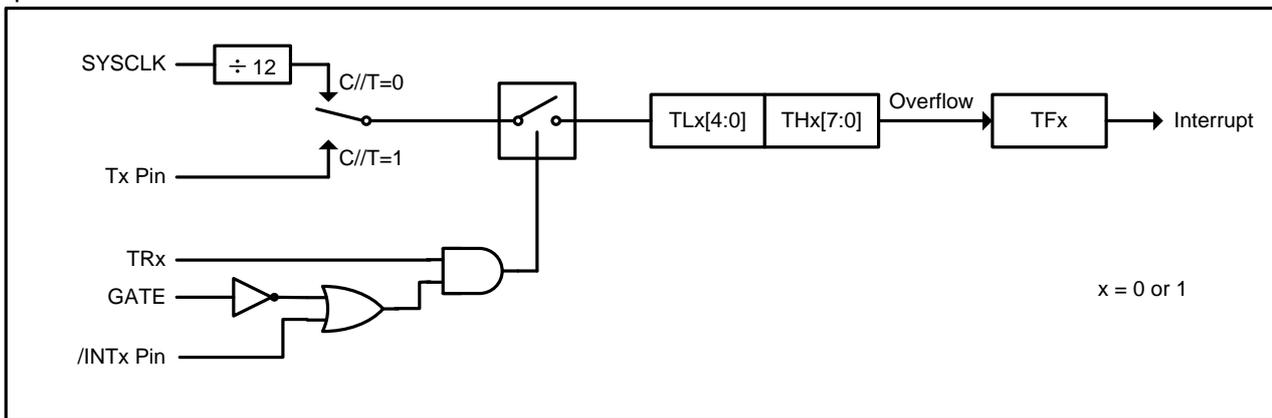
In the “timer” function, the register is incremented every machine cycle. In other words, it is to count the machine cycle. Due to 12(6) oscillator periods in a machine cycle, the count rate is 1/12(1/6) of the oscillator frequency.

In the “counter” function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected.

9.1. Timer0 and Timer1

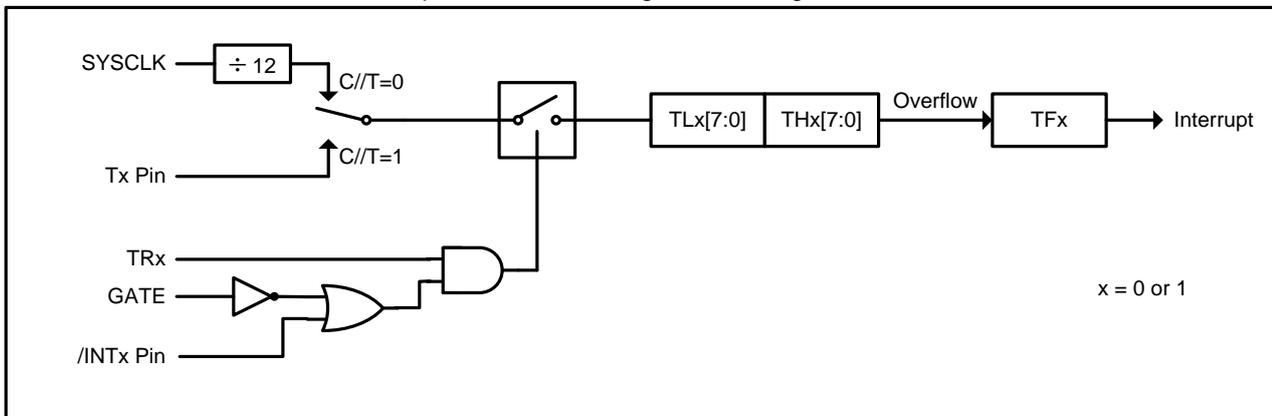
9.1.1. Mode 0 Structure

The timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the timer interrupt flag TFX. The counted input is enabled to the timer when TRx = 1 and either GATE=0 or INTx = 1. Mode 0 operation is the same for Timer0 and Timer1.



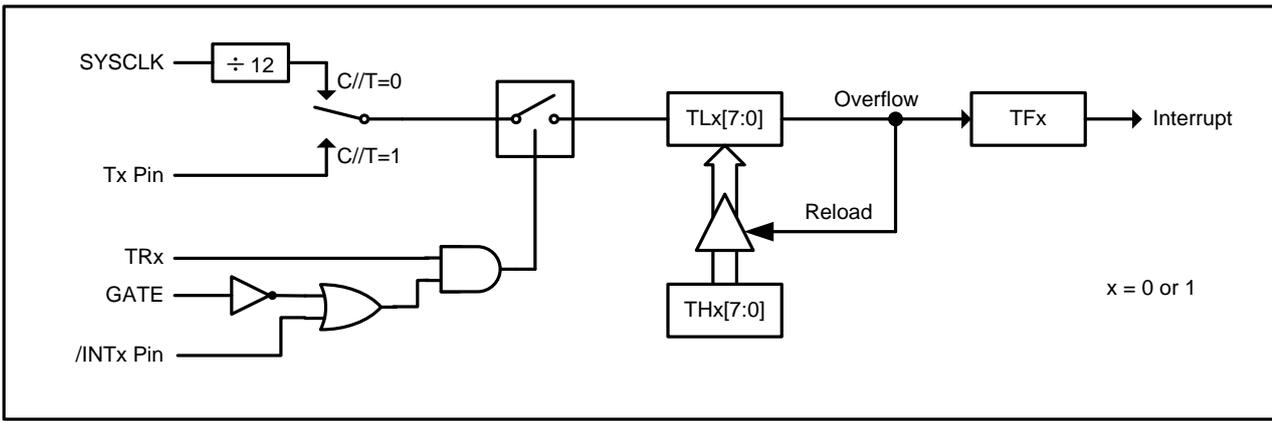
9.1.2. Mode 1 Structure

Mode1 is the same as Mode0, except that the timer register is being run with all 16 bits.



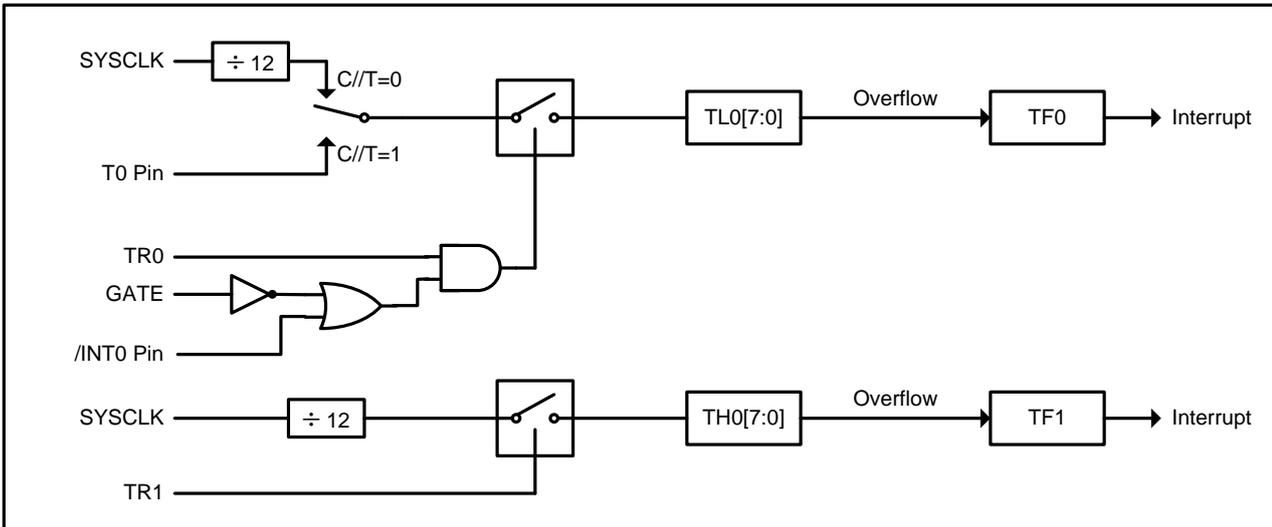
9.1.3. Mode 2 Structure

Mode 2 configures the timer register as an 8-bit counter(TLx) with automatic reload. Overflow from TLx not only set TFX, but also reload TLx with the content of THx, which is determined by software. The reload leaves THx unchanged. Mode 2 operation is the same for Timer0 and Timer1.



9.1.4. Mode 3 Structure

Timer1 in Mode3 simply holds its count, the effect is the same as setting TR1 = 1. Timer0 in Mode 3 enables TL0 and TH0 as two separate 8-bit counters. TL0 uses the Timer0 control bits such like C/T, GATE, TR0, INT0 and TF0. TH0 is locked into a timer function (can not be external event counter) and take over the use of TR1, TF1 from Timer1. TH0 now controls the Timer1 interrupt.



9.1.5. Timer0/1 Register

TMOD: Timer/Counter Mode Control Register

Address=89H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
GATE	C/T	M1	M0	GATE	C/T	M1	M0

|←----- Timer1 ----->|←----- Timer0 ----->|

Bit 7/3: Gate, Gating control for Timer1/0.

0: Disable gating control for Timer1/0.

1: Enable gating control for Timer1/0. When set, Timer1/0 or Counter1/0 is enabled only when /INT1 or /INT0 pin is high and TR1 or TR0 control bit is set.

Bit 6/2: C/T, Timer for Counter function selector.

0: Clear for Timer operation, input from internal system clock.

1: Set for Counter operation, input form T1 input pin.

Bit 5~4/1~0: Operating mode selection.

M1	M0	Operating Mode
0	0	13-bit timer/counter for Timer0 and Timer1
0	1	16-bit timer/counter for Timer0 and Timer1
1	0	8-bit timer/counter with automatic reload for Timer0 and Timer1
1	1 (Timer0)	TL0 is 8-bit timer/counter, TH0 is locked into 8-bit timer
1	1 (Timer1)	Timer/Counter1 Stopped

TCON: Timer/Counter Control Register

Address=88H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Bit 7: TF1, Timer 1 overflow flag.

0: Cleared by hardware when the processor vectors to the interrupt routine, or cleared by software.

1: Set by hardware on Timer/Counter 1 overflow, or set by software.

Bit 6: TR1, Timer 1 Run control bit.

0: Cleared by software to turn Timer/Counter 1 off.

1: Set by software to turn Timer/Counter 1 on.

Bit 5: TF0, Timer 0 overflow flag.

0: Cleared by hardware when the processor vectors to the interrupt routine, or cleared by software.

1: Set by hardware on Timer/Counter 0 overflow, or set by software.

Bit 4: TR0, Timer 0 Run control bit.

0: Cleared by software to turn Timer/Counter 0 off.

1: Set by software to turn Timer/Counter 0 on.

Bit 3: IE1, Interrupt 1 Edge flag.

0: Cleared when interrupt processed on if transition-activated.

1: Set by hardware when external interrupt 1 edge is detected (transmitted or level-activated).

Bit 2: IT1: Interrupt 1 Type control bit.

0: Cleared by software to specify low level triggered external interrupt 1.

1: Set by software to specify falling edge triggered external interrupt 1.

Bit 1: IE0, Interrupt 0 Edge flag.

0: Cleared when interrupt processed on if transition-activated.

1: Set by hardware when external interrupt 0 edge is detected (transmitted or level-activated).

Bit 0: IT0: Interrupt 0 Type control bit.

0: Cleared by software to specify low level triggered external interrupt 0.

1: Set by software to specify falling edge triggered external interrupt 0.

9.2. Timer0/1 Sample Code

(1). Required Function: IDLE mode with T0 wake-up frequency 10KHz, SYSCLK = 12MHz Crystal

Assembly Code Example:		
TOM0	EQU	01h
TOM1	EQU	02h
IDL	EQU	01h
ORG	0000h	
JMP	main	
ORG	0000Bh	
time0_isr:		
to do...		
RETI		
main:		; (unsigned short value)
MOV	TH0,#(256-100)	; Set Timer 0 overflow rate = SYSCLK x 100
MOV	TL0,#(256-100)	;
ANL	TMOD,#0F0h	; Set Timer 0 to Mode 2
ORL	TMOD,#TOM1	;
CLR	TF0	; Clear Timer 0 Flag
SETB	ET0	; Enable Timer 0 interrupt
SETB	EA	; Enable global interrupt
SETB	TR0	; Start Timer 0 running
ORL	PCON,#IDL	; Set MCU into IDLE mode
JMP	\$	
C Code Example:		
#define	TOM0	0x01
#define	TOM1	0x02
#define	IDL	0x01
void	time0_isr(void) interrupt 1	
{		
To do...		
}		
void	main(void)	
{		
TH0 = TL0 = (256-100);		// Set Timer 0 overflow rate = SYSCLK x 100
TMOD &= 0xF0;		// Set Timer 0 to Mode 2
TMOD = TOM1;		
TF0 = 0;		// Clear Timer 0 Flag
ET0 = 1;		// Enable Timer 0 interrupt
EA = 1;		// Enable global interrupt
TR0 = 1;		// Start Timer 0 running
PCON=IDL;		// Set MCU into IDLE mode
while(1);		
}		

(2). Required Function: Select Timer 0 clock source from SYSCLK

Assembly Code Example:		
TOM0	EQU	01h
TOM1	EQU	02h

```

ORG    0000h
JMP    main

ORG    0000Bh
time0_isr:
to do...
RETI

main:
CLR    TF0                ; Clear Timer 0 Flag

SETB   ET0                ; Enable Timer 0 interrupt
SETB   EA                 ; Enable global interrupt

MOV    TH0, #(256 - 240)   ;interrupt interval 20us
MOV    TL0, #(256 - 240)   ;

ANL    TMOD,#0F0h         ; Set Timer 0 to Mode 2
ORL    TMOD,#T0M1        ;

SETB   TR0                ; Start Timer 0 running
JMP    $

```

```

C Code Example:
#define T0M0          0x01
#define T0M1          0x02

TF0 = 0;

ET0 = 1;                // Enable Timer 0 interrupt
EA = 1;                 // Enable global interrupt

TH0 = TL0 = (256 - 240);

TMOD &= 0xF0;          // Set Timer 0 to Mode 2
TMOD |= T0M1;

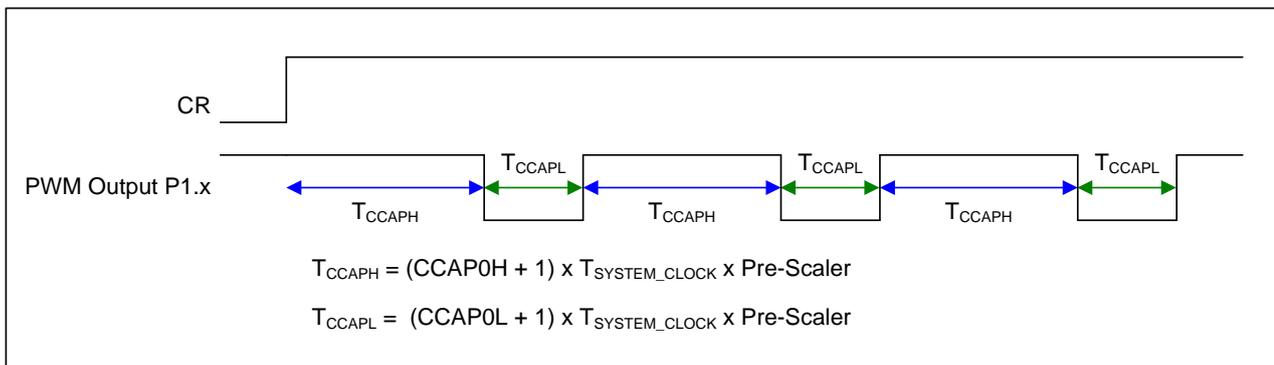
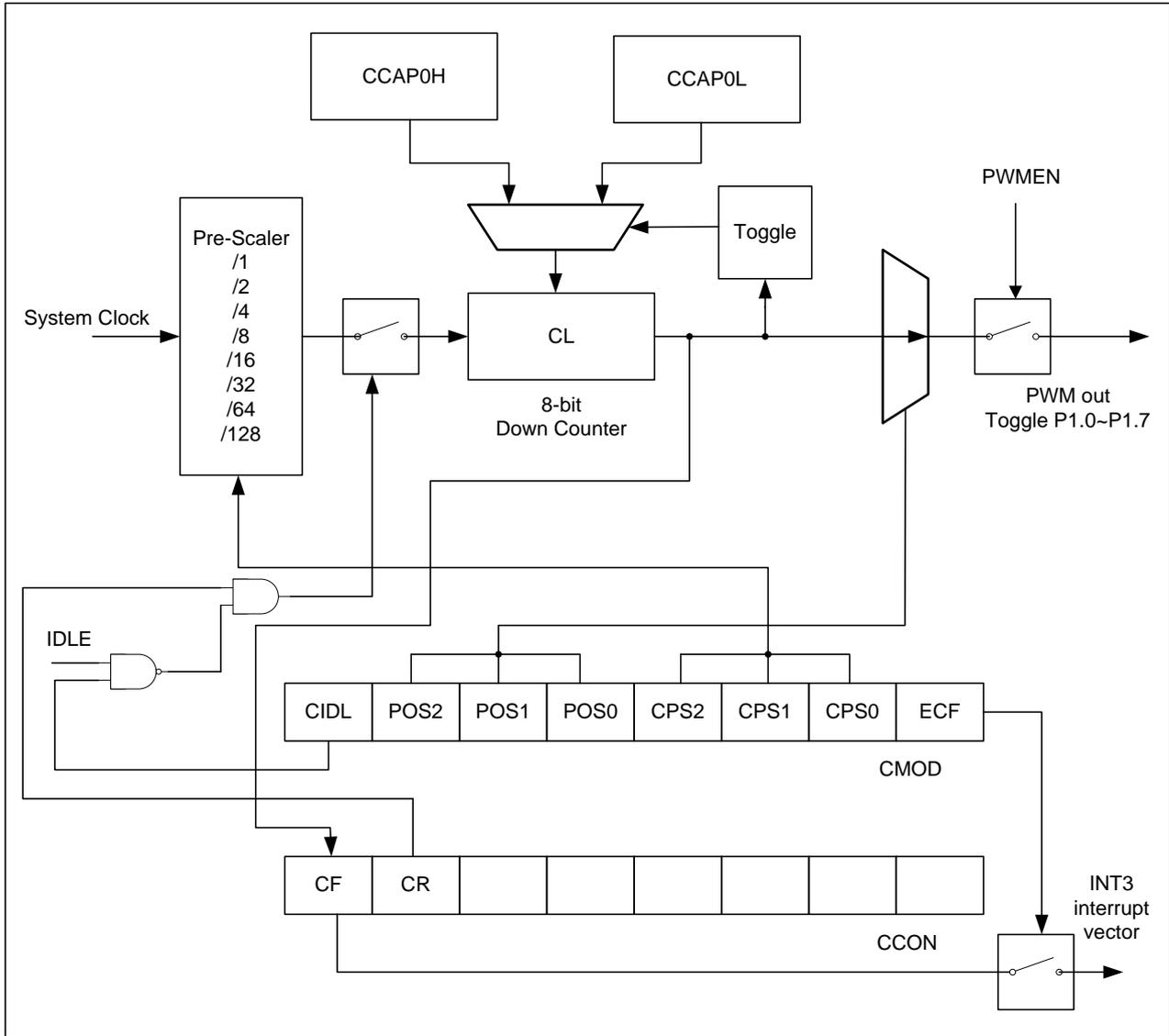
TR0 = 1;                // Start Timer 0 running

```

9.3.PWM-Timer

An 8-bits timer that special designed for PWM generator.

9.3.1.PWM-Timer Structure



9.3.2.PWM-Timer Register

CMOD: PWM-timer Mode Register

Address=D9H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
CIDL	POS2	POS1	POS0	CPS2	CPS1	CPS0	ECF

Bit 7: CIDL, Counter Idle Control.

0: Program the PWM-Timer to continue functioning during IDLE mode.

1: Program the PWM-Timer to be gated off during IDLE mode.

Bit 6-4: POS[2:0], PWM output port select.

POS[2:0]	PWMEN	PWM Output Port
0 0 0	1	P1.0
0 0 1	1	P1.1
0 1 0	1	P1.2
0 1 1	1	P1.3
1 0 0	1	P1.4
1 0 1	1	P1.5
1 1 0	1	P1.6
1 1 1	1	P1.7
X X X	0	Disabled

Bit 3-1: CPS[2:0], Counter Pre-scalar Select.

CPS[2:0]	Pre-scalar
0 0 0	1
0 0 1	2
0 1 0	4
0 1 1	8
1 0 0	16
1 0 1	32
1 1 0	64
1 1 1	128

Bit 0: ECF, Enable PWM-Timer underflow interrupt.

0: Disables CF bit in CCON to generate an interrupt.

1: Enables CF bit in CCON to generate an interrupt.

CCON: PWM-timer Control Register

Address=D8H, read/write, Power On + RESET=00XX-XXXX

7	6	5	4	3	2	1	0
CF	CR	-	-	-	-	-	-

Bit 7: CF, PWM-timer underflow Flag.

0: This flag can only be cleared by software.

1: Set by hardware when the counter rolls under. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software.

Bit 6: CR, PWM-timer Run control bit.

0: Must be cleared by software to turn the PWM-Timer counter off.

1: Set by software to turn the PWM-Timer counter on.

Bit 5-0: Reserved.

CCAP0L: PWM-Timer L-Duty Register

Address=EAH, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

CCAP0H: PWM-Timer H-Duty Register

Address=FAH, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

9.4. PWM Sample Code

(1). Required Function: Set P1.7 output PWM with 75% duty cycle

Assembly Code Example:

```

PWM_P17      EQU      070h
PWM_Pre_scale_DIV_8  EQU      06h
PWMEN      EQU      020h
CF         EQU      080h
CR         EQU      040h

      ORG      0000h
      JMP      main

main:
;

      MOV      CACP0H,#192      ; Set PWM with 75% duty
      MOV      CACP0L,#(255-192) ;
      MOV      CMOD,#PWM_P17    ; Set P1.7 output PWM
      ORL      CMOD,#PWM_Pre_scale_DIV_8 ; Set SYSCLK/8 as PWM clock source
      ANL      CCON,#(0FFh - CF) ; Clear PWM underflow Flag

      ORL      PCON,#PWNEN      ; Enable PWM output from I/O
      ORL      CCON,#CR        ; Start PWM running

      JMP      $

```

C Code Example:

```

#define PWM_P17      0x70
#define PWM_Pre_scale_DIV_8  0x06
#define PWMEN      0x20
#define CF         0x80
#define CR         0x40

void main(void)
{
    CCAP0H = 192;          // Set PWM with 75% duty
    CCAP0L = (255-192);    //
    CMOD |= PWM_P17;      // Set P1.7 output PWM
    CMOD |= PWM_Pre_scale_DIV_8; // Set SYSCLK/8 as PWM clock source
    CCON &= ~CF;          // Clear PWM underflow Flag

    PCON |= PWMEN;        // Enable PWM output from I/O
    CCON |= CR;           // Start PWM running

    while(1);
}

```

10. UART

The serial port(UART) of MG87FE/L2051/4051/6051 support full-duplex transmission. It can transmit and receive simultaneously. The serial port receive and transmit share the same SFR – SBUF, but actually there is two SBUFs in the chip, one is for transmit and the other is for receive. The serial port can be operated in 4 different modes.

10.1. UART Structure

Mode 0

Serial data enters and exits through RXD(P3.0) and TXD(P3.1) outputs the shift clock. 8-bits are transmitted/received with LSB first. The baud rate is fixed at 1/12 the system clock frequency.

$$\text{Baud Rate in Mode 0} = \frac{F_{\text{SYSCLK}}}{12}$$

Mode1

10 bits are transmitted through TXD or received through RXD. The frame data includes a start bit(0), 8 data bits and a stop bit(1). One receive, the stop bit goes into RB8 in SFR – SCON. The baud rate is variable.

$$\text{Baud Rate in Mode 1} = \frac{2^{\text{SMOD}}}{32} \times (\text{timer1 overflow rate})$$

Mode2

11 bits are transmitted through TXD or received through RXD. The frame data includes a start bit(0), 8 data bits, a programmable 9th bit and a stop bit(1). On transmit, the 9th data bit comes from TB8 in SCON. On receive, the 9th data bit goes into RB8 in SCON. The baud rate is programmable to either 1/32 or 1/64 the system clock frequency.

$$\text{Baud Rate in Mode 2} = \frac{2^{\text{SMOD}}}{64} \times F_{\text{SYSCLK}}$$

Mode3

Mode 3 is the same as mode 2 except the baud rate is variable.

$$\text{Baud Rate in Mode 3} = \frac{2^{\text{SMOD}}}{32} \times (\text{timer1 overflow rate})$$

In all four modes, transmission is initiated by any instruction that use SBUF as a destination register. Reception is initiated in mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit with 1-to-0 transition if REN=1.

Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware comparison circuit. This feature improves the overhead of software by eliminating the need in examine every incoming address. This feature is enabled by setting the SM2 bit in SCON. In mode2 and mode3, the receive interrupt flag(RI) will be automatically set when the received byte contains either the “Given” address or the “Broadcast” address. These two modes require the 9th received bit is a 1 to indicate that received information is an address and not the data byte.

In mode1, the RI flag will be set if SM2 is enabled and a valid stop bit is received which the stop bit follows the 8 address bits and the information is either a given or Broadcast address.

In mode 0, SM2 is ignored.

Framing Error Detection

Framing Error Detection allows the serial port to check for valid stop bits in modes 1, 2, or3. A missing stop bit can be caused, for example, by noise on the serial lines, or transmission by two CPUs simultaneously.

If a stop bit is missing, a Framing Error bit FE is set. The FE bit can be checked in software after each reception to detect communication errors. Once set, the FE bit must be cleared in software. A valid stop bit will not clear FE.

The FE bit is located in SCON and shares the same bit address as SM0. Control bit SMOD0 in the PCON register (location PCON.6) determines whether the SM0 or FE bit is accessed. If SMOD0 = 0, then accesses to SCON.7 are to SM0. IF SMOD0 = 1, then accesses to SCON.7 are to FE.

10.2. UART Register

SCON: Serial port Control Register

Address=98H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI

Bit 7: FE, Framing Error bit. The SMOD0 bit must be set to enable access to the FE bit.

0: The FE bit is not cleared by valid frames but should be cleared by software.

1: This bit is set by the receiver when an invalid stop bit is detected.

Bit 7: Serial port mode bit 0, (SMOD0 must = 0 to access bit SM0)

Bit 6: Serial port mode bit 1.

SM0	SM1	Mode	Description	Baud Rate
0	0	0	shift register	F _{SYSClk} /12
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	F _{SYSClk} /64 or F _{SYSClk} /32
1	1	3	9-bit UART	variable

Bit 5: Serial port mode bit 2.

0: Disable SM2 function.

1: Enable the automatic address recognition feature in Modes 2 and 3. If SM2=1, RI will not be set unless the received 9th data bit is 1, indicating an address, and the received byte is a Given or Broadcast address. In mode1, if SM2=1 then RI will not be set unless a valid stop Bit was received, and the received byte is a Given or Broadcast address. In Mode 0, SM2 should be 0.

Bit 4: REN, Enable serial reception.

0: Clear by software to disable reception.

1: Set by software to enable reception.

Bit 3: TB8, The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.

Bit 2: RB8, In Modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.

Bit 1: TI. Transmit interrupt flag.

0: Must be cleared by software.

1: Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission.

Bit 0: RI. Receive interrupt flag.

0: Must be cleared by software.

1: Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2).

SBUF: Serial Buffer Register

Address=99H, read/write, Power On + RESET=XXXX-XXXX

7	6	5	4	3	2	1	0

Bit 7-0: It is used as the buffer register in transmission and reception.

SADDR: Slave Address Register

Address=A9H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0

SCON: Slave Address Mask Register

Address=B9H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0

SADDR register is combined with SADEN register to form Given/Broadcast Address for automatic address recognition. In fact, SADEN functions as the “mask” register for SADDR register. The following is the example for it.

$$\begin{array}{rcl}
 \text{SADDR} & = & 1100\ 0000 \\
 \text{SADEN} & = & 1111\ 1101 \\
 \hline
 \text{Given} & = & 1100\ 00x0 \longrightarrow \text{The Given slave address will be checked except} \\
 & & \text{bit 1 is treated as “don’t care”}
 \end{array}$$

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zero in this result is considered as “don’t care”. Upon reset, SADDR and SADEN are loaded with all 0s. This produces a Given Address of all “don’t care” and a Broadcast Address of all “don’t care”. This disables the automatic address detection feature.

10.3. Serial Port Sample Code

(1). Required Function: IDLE mode with RI wake-up capability

Assembly Code Example:

```

    JMP    main
    ORG    00023h
uart_ri_idle_isr:
    JB     RI,RI_ISR           ;
    JB     TI,TI_ISR          ;
    RETI                          ;

RI_ISR:
; Process
    CLR    RI                  ;
    RETI                          ;

TI_ISR:
; Process
    CLR    TI                  ;
    RETI                          ;

main:
    CLR    TI                  ;
    CLR    RI                  ;
    SETB   SM1                 ;
    SETB   REN                 ; 8bit Mode2, Receive Enable

    CALL   UART_Baud_Rate_Setting ;

    SETB   ES                  ; Enable S0 interrupt
    SETB   EA                  ; Enable global interrupt

    ORL    PCON,#IDL;         ; Set MCU into IDLE mode

```

C Code Example:

```

void uart_ri_idle_isr(void) interrupt 4
{
    if(RI)
    {
        RI=0;
        // to do ...
    }

    if(TI)
    {
        TI=0;
        // to do ...
    }
}

void main(void)
{
    TI = RI = 0;
    SM1 = REN = 1;                // 8bit Mode2, Receive Enable

    UART_Baud_Rate_Setting()     //

    ES = 1;                       // Enable S0 interrupt
    EA = 1;                       // Enable global interrupt

    PCON |= IDL;                 // Set MCU into IDLE mode
}

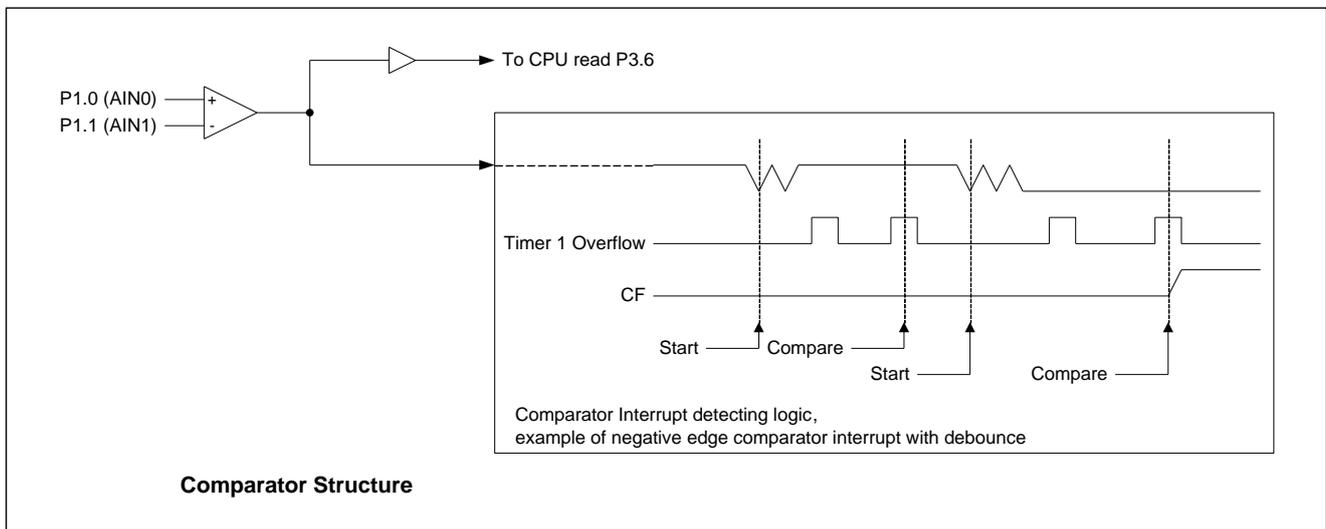
```

11. Analog Comparator

A single analog comparator is provided in the MG87FE/L2051/4051/6051. The comparator operation is such that the output is a logical “HIGH” when the positive input AIN0 (P1.0) is greater than the negative input AIN1 (P1.1). Otherwise the output is “LOW”. Setting the ACEN bit in ACSR enables the comparator. When the comparator is first enabled, the comparator output and interrupt flag are not guaranteed to be stable for 10 microseconds. The corresponding comparator interrupt should not be enabled during that time, and the comparator interrupt flag must be cleared before the interrupt is enabled in order to prevent an immediate interrupt service.

The comparator may be configured to cause an interrupt under a variety of output value conditions by setting the ACM bits in ACSR. The comparator interrupt flag ACF in ACSR is set whenever the comparator output matches the condition specified by ACM. The flag may be polled by firmware or may be used to generate an interrupt and must be cleared by firmware. The analog comparator is always disabled during Idle or Power-down modes.

11.1. Analog Comparator Structure



The comparator output is sampled at every State 4 (S4) of every machine cycle. The conditions on the analog inputs may be such that the comparator output will toggle excessively. This is especially true if applying slow moving analog inputs. Three de-bouncing modes are provided to filter out this noise. In de-bouncing mode, the comparator uses Timer-1 to modulate its sampling time. When a relevant transition occurs, the comparator waits until two Timer-1 overflows have occurred before re-sampling the output. If the new sample agrees with the expected value, ACF is set. Otherwise, the event is ignored. The filter may be tuned by adjusting the timeout period of Timer-1. Because Timer-1 is free running, the de-bouncer must wait for two overflows to guarantee that the sampling delay is at least 1 timeout period. Therefore, after the initial edge event, the interrupt may occur between 1 and 2 timeout periods later.

11.2. Analog Comparator Register

ACSR: Analog Comparator Control & Status Register

Address=97H, read/write, Power On + RESET=0xx0-0000

7	6	5	4	3	2	1	0
ACIDX	-	-	ACF	ACEN	ACM2	ACM1	ACM0

Bit 7: ACIDX, Analog Comparator IDLE control.

- 0: Program the Analog Comparator to be gated off during IDLE mode.
- 1: Program the Analog Comparator to continue functioning during IDLE mode.

Bit 6~5: Reserved.

Bit 4: ACF. Analog Comparator Interrupt Flag.

0: The flag must be cleared by software.

1: Set when the comparator output meets the conditions specified by the ACM [2:0] bits and ACEN is set. The interrupt may be enabled/disabled by setting/clearing bit 6 of IE.

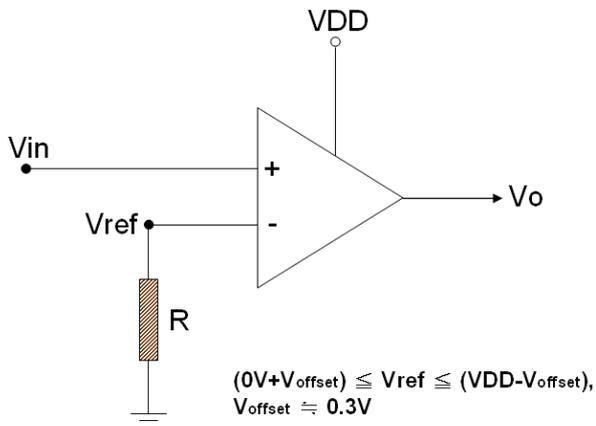
Bit 3: ACEN. Analog Comparator Enable.

0: Clearing this bit will force the comparator output low and prevent further events from setting ACF.

1: Set this bit to enable the comparator.

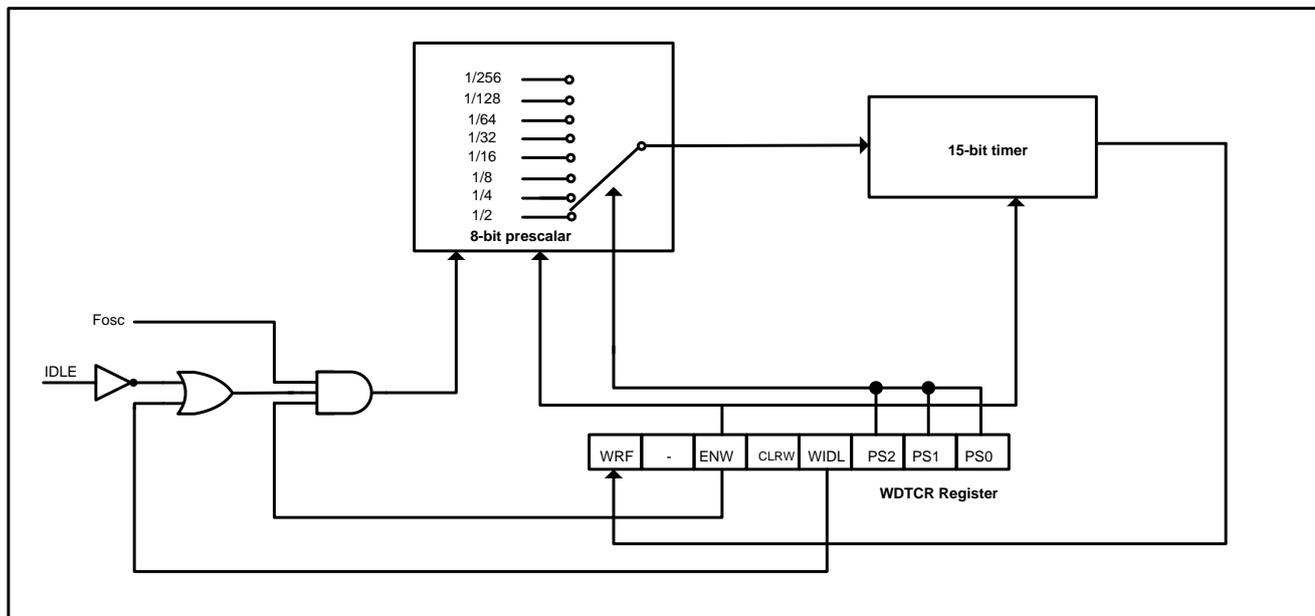
Bit 2~0: ACM2 ~ ACM1, Analog Comparator Interrupt Mode.

ACM[2:0]	Interrupt Mode
0 0 0	Negative (Low) level
0 0 1	Positive edge
0 1 0	Toggle with de-bounce
0 1 1	Positive edge with de-bounce
1 0 0	Negative edge
1 0 1	Toggle
1 1 0	Negative edge with de-bounce
1 1 1	Positive (High) level



12. Watch Dog Timer (WDT)

12.1. WDT Structure



12.2. WDT Register

WDTCR: Watch-Dog-Timer Control Register

Address=E1H, read/write, Power On + Reset =0x00-0000

7	6	5	4	3	2	1	0
WRF	-	ENW	CLRW	WIDL	PS2	PS1	PS0

Bit 7: WRF, WDT reset flag.

0: This bit should be cleared by software.

1: When WDT overflows, this bit is set by hardware.

Bit 6: Reserved.

Bit 5: ENW. Enable WDT.

0: ENW can not be cleared by software.

1: Enable WDT while it is set.

Bit 4: CLRW. Clear WDT counter.

0: Hardware will automatically clear this bit.

1: Clear WDT to recount while it is set.

Bit 3: WIDL. WDT idle control.

0: WDT stops counting while the MCU is in idle mode.

1: WDT keeps counting while the MCU is in idle mode.

Bit 2~0: PS2 ~ PS0, select pre-scalar output for WDT time base input.

PS[2:0]	Pre-scalar Value
0 0 0	2
0 0 1	4
0 1 0	8
0 1 1	16
1 0 0	32
1 0 1	64
1 1 0	128
1 1 1	256

12.3. WDT Sample Code

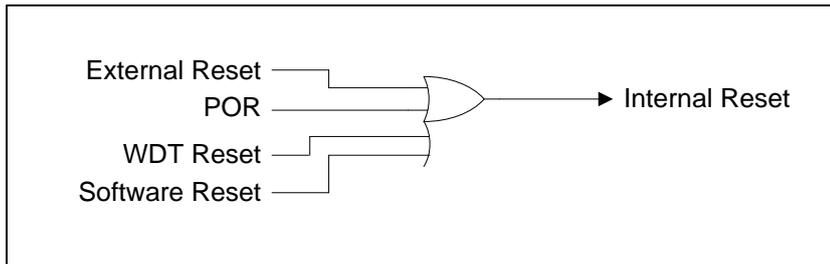
(1) Required function: Enable WDT and select WDT prescaler to 1/32

Assembly Code Example:			
PS0	EQU	01h	
PS1	EQU	02h	
PS2	EQU	04h	
WIDL	EQU	08h	
CLRW	EQU	10h	
ENW	EQU	20h	
WRF	EQU	80h	
ANL	WDTCR,#(0FFh - WRF)		; Clear WRF flag (write "0")
MOV	WDTCR,#(ENW + CLRW + PS2)		; Enable WDT counter and set WDT prescaler to 1/32
C Code Example:			
#define	PS0	0x01	
#define	PS1	0x02	
#define	PS2	0x04	
#define	WIDL	0x08	
#define	CLRW	0x10	
#define	ENW	0x20	
#define	WRF	0x80	
	WDTCR &= ~WRF;		// Clear WRF flag (write "0")
	WDTCR = (ENW CLRW PS2);		// Enable WDT counter and set WDT prescaler to 1/32
			// PS[2:0] WDT prescaler selection
			// 0 1/2
			// 1 1/4
			// 2 1/8
			// 3 1/16
			// 4 1/32
			// 5 1/64
			// 6 1/128
			// 7 1/256

13. Reset

During reset, all I/O Registers are set to their initial values, the port pins are weakly pulled to VDD, and the program starts execution from the Reset Vector, 0000H, or ISP start address by OR setting. MG87FE/ L2051/4051/6051 all have four sources of reset: external reset, power-on reset, WDT reset, and software reset.

13.1. Reset Source



14. Power Management

MG87FE/L2051/4051/6051 supports two power-reducing modes: Idle and Power-down mode. These two modes are accessed through the PCON register.

14.1. Power Saving Mode

14.1.1. Idle Mode

Setting the IDL bit in PCON enters idle mode. Idle mode halts the internal CPU clock. The CPU state is preserved in its entirety, including the RAM, stack pointer, program counter, program status word, and accumulator. The Port pins hold the logical states they had at the time that Idle was activated. Idle mode leaves the peripherals running in order to allow them to wake up the CPU when an interrupt is generated. Timer 0, Timer 1, PWM-Timer and the UART will continue to function during Idle-mode. The analog comparator is disabled during Idle. Any enabled interrupt source or reset may terminate Idle-mode. When exiting Idle-mode with an interrupt, the interrupt will immediately be serviced, and following RETI, the next instruction to be executed will be the one following the instruction that put the device into Idle.

P1.0 and P1.1 should be set to “0” if no external pull-ups are used, or set to “1” if external pull-ups are used, or set to “1” if AUXR.P10PU&P11PU are enabled.

14.1.2. Power-down Mode

Setting the PD bit in PCON0 enters Power-down mode. Power-down mode stops the oscillator and powers down the internal macros in order to minimize power consumption. Only the power-on circuitry will continue to draw power during Power-down. During Power-down the power supply voltage may be reduced to the RAM keep-alive voltage. The RAM contents will be retained; however, the SFR contents are not guaranteed once VDD has been reduced out of chip operating voltage range. Power-down may be exited by external reset, power-on reset, enabled external interrupts, or enabled wake-up GPIOs.

The user should not attempt to enter (or re-enter) the power-down mode for a minimum of 4 μ s until after one of the following conditions has occurred: Start of code execution (after any type of reset), or exit from power-down mode.

14.1.3. Interrupt Recovery from Power-down

Four external interrupts may be configured to terminate Power-down mode. External interrupts /INT0 (P3.2), /INT1 (P3.3), /INT2 (P4.3) and /INT2 (P4.2) may be used to exit Power-down. To wake up by external interrupt /INT0, /INT1, /INT2, or /INT3, the interrupt must be enabled and configured for level-sensitive operation.

If the interrupt vector of /INT2 (P4.3) is occupied by Analog Comparator, low level P4.3 input still have wake-up capability when /INT2 interrupt enable, XICON.EX2, is set (enabled). If the interrupt vector of /INT3 (P4.2) is occupied by PWM-Timer underflow, low level P4.2 input still have wake-up capability when /INT3 interrupt enable, XICON.EX3, is set (enabled).

When terminating Power-down by an interrupt, two different wake-up modes are available. When PWDEX in CKCON3.2 is zero, the wake up period is internally timed. At the falling edge on the interrupt pin, Power-down is exited, the oscillator is restarted, and an internal timer begins counting. The internal clock will not be allowed to propagate and the CPU will not resume execution until after the timer has reached internal counter full. After the timeout period, the interrupt service routine will begin. To prevent the interrupt from re-triggering, the ISR should disable the interrupt before returning. The interrupt pin should be held low until the device has timed out and begun executing.

When PWDEX = 1 the wake-up period is controlled externally by the interrupt. Again, at the falling edge on the interrupt pin, Power-down is exited and the oscillator is restarted. However, the internal clock will not propagate and CPU will not resume execution until the **rising edge** of the interrupt pin. After the rising edge on the pin, the interrupt service routine will begin. The interrupt should be held low long enough for the oscillator to stabilize.

14.1.4. Reset Recovery from Power-down

Wake-up from Power-down through an external reset is similar to the interrupt with PWDEX = 0. At the rising edge of RST, Power-down is exited, the oscillator is restarted, and an internal timer begins counting. The internal clock will not be allowed to propagate to the CPU until after the timer has reached internal counter full. The RST pin must be held high for longer than the timeout period to ensure that the device is reset properly. The device will begin executing once RST is brought low.

It should be noted that when idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

14.1.5. GPIO wake-up Recovery from Power-down

The GPIOs of MG87FE/L2051/4051/6051, P1.7 ~ P1.0 and P3.0 ~ P3.5, P3.7 have wake-up CPU capability that are enabled by individual control bit in P1WKPE and P3WKPE. If the interrupt is disabled on P3.2/INT0 or P3.3/INT1, P3.2 and P3.3 still have the wake-up function from the P3WKPE control. But P4.2/INT3 and P4.3/INT2 can wake-up CPU only when the respective interrupt is enabled.

Wake-up from Power-down through an enabled wake-up GPIO is similar to the interrupt with PWDEX = 0. At the falling edge of enabled wake-up GPIO, Power-down is exited, the oscillator is restarted, and an internal timer begins counting. The internal clock will not be allowed to propagate to the CPU until after the timer has reached internal counter full. After the timeout period, there is no any interrupt and CPU will execute the following command after last power-down instruction. That is, the enabled wake-up GPIOs will only have the capability to wake-up CPU without any interrupt function.

14.2. Power Control Register

PCON: Power Control Register

Address=87H, read/write, Power On + RESET =0001-0000, RESET=000x-0000

7	6	5	4	3	2	1	0
SMOD	SMOD0	PWMEN	POF	GF1	GF0	PD	IDL

Bit 7: SMOD, double Baud rate control bit.
 0: Disable double Baud rate of the UART.
 1: Enable double Baud rate of the UART in mode 1, 2, or 3.

Bit 6: SMOD0, Frame Error select.
 0: SCON.7 is SM0 function.
 1: SCON.7 is FE function. Note that FE will be set after a frame error regardless of the state of SMOD0.

Bit 5: PWMEN, PWM mode enable for PWM-Timer.
 0: Set the PWM-Timer as Timer mode.
 1: Set the PWM-Timer as PWM mode and trigger the output on P1.0 ~ P1.7 by POS[2:0] indexed.

Bit 4: POF. Power-On Flag.
 0: The flag must be cleared by software.
 1: POF is set to "1" during power up (i.e. cold reset). It can be set under software control and is not affected by RESEST (i.e. warm resets).

Bit 3~2: GF1, GF0, General purpose flags.

Bit 1: PD, Power-Down control bit.

0: This bit could be cleared by CPU or any exited power-down event.

1: Setting this bit activates power down operation.

Bit 0: IDL, Idle mode control bit.

0: This bit could be cleared by CPU or any exited Idle mode event.

1: Setting this bit activates idle mode operation.

P1WKPE: Port 1 Wake-up Enable Control Register

Address=D7H, read/write, RESET=0000-0000

7	6	5	4	3	2	1	0
P17WKP	P16WKP	P15WKP	P14WKP	P13WKP	P12WKP	P11WKP	P10WKP

Bit 7~0: Wake-up enable bit for each P1 pins.

0: Disable port pin wake-up function.

1: Enable port pin wake-up function when port input at falling edge in power-down mode and idle mode.

P1WKPE: Port 3 Wake-up Enable Control Register

Address=D6H, read/write, RESET=0000-0000

7	6	5	4	3	2	1	0
P37WKP	-	P35WKP	P34WKP	P33WKP	P32WKP	P31WKP	P30WKP

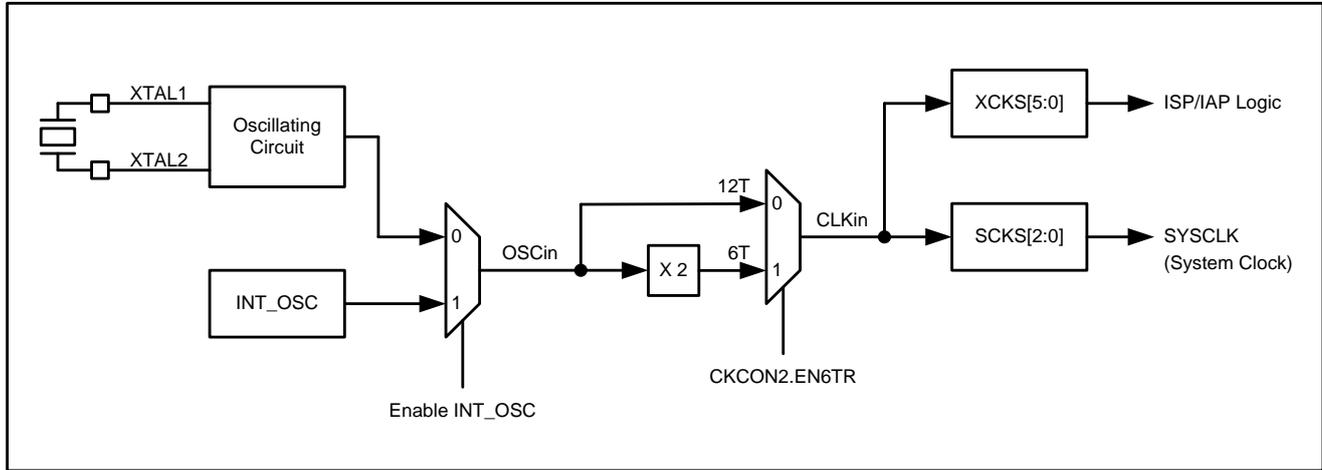
Bit 7, 5~0: Wake-up enable bit for each P3 pins except P3.6.

0: Disable port pin wake-up function.

1: Enable port pin wake-up function when port input at falling edge in power-down mode and idle mode.

15. System Clock

15.1. Clock Structure



Note: Internal RC_OSC frequency +/- 1% frequency drift @ 25°C,
 +/- 2% frequency drift @ -20 ~ 50°C,
 +/- 4% frequency drift @ -40 ~ 85°C,.

15.2. Clock Register

CKCON: Clock Control Register

Address=C7H, read/write, RESET=xxxx-x000

7	6	5	4	3	2	1	0
-	-	-	-	-	SCKS2	SCKS1	SCKS0

Bit 7~3: Reserved.

Bit 2~0: SCKS2 ~ SCKS0, programmable System Clock Selection.

SCKS[2:0]	System Clock (F _{SYSCLK})
0 0 0	CLKin
0 0 1	CLKin /2
0 1 0	CLKin /4
0 1 1	CLKin /8
1 0 0	CLKin /16
1 0 1	CLKin /32
1 1 0	CLKin /64
1 1 1	CLKin /128

CKCON2: Clock Control Register 2

Address=BFH, read/write, RESET=xx00-1010

7	6	5	4	3	2	1	0
OSCDR	EN6TR	XCKS5	XCKS4	XCKS3	XCKS2	XCKS1	XCKS0

Bit 7: OSCDR, OSC Driving control Register. Default value is load from OR1.b4 inverted value. And it could be read/written by CPU.

0: The driving of crystal oscillator is enough for oscillation up to 48MHz.

1: The driving of crystal oscillator is reduced. It will helpful in EMI reduction. Regarding application not needing high frequency clock, it is recommended to do so.

Bit 6: EN6TR, Enable 6T mode control register. Default value is load from OR1.b0 inverted value. And it could be read/written by CPU. The access on this bit will affect CKCON3.EN6TR to corresponding operation and get same control function.

0: MG87FE/L2051/4051/6051 will run in 12T mode.

1: MG87FE/L2051/4051/6051 will run in 6T mode.

Bit 5~0: This is set the crystal frequency value to define the time base of ISP/IAP programming. Fill with a proper value according to OSCin, as listed below.

$[XCKS5-XCKS0] = OSCin - 1$, where $OSCin=1\sim48$ (MHz) in 12T mode.

$[XCKS5-XCKS0] = OSCin \times 2 - 1$, where $OSCin=0.5\sim24$ (MHz) in 6T mode.

For a 12T examples,

(1) If $OSCin=12$ MHz, then fill $[XCKS4-XCKS0]$ with 11, i.e., 001011B.

(2) If $OSCin=6$ MHz, then fill $[XCKS4-XCKS0]$ with 5, i.e., 000101B.

OSCin @ 12T	OSCin @ 6T	XCKS Setting
1MHz	0.5MHz	6'b000000
2MHz	1MHz	6'b000001
3MHz	1.5MHz	6'b000010
4MHz	2MHz	6'b000011
.....
45MHz	22.5MHz	6'b101100
46MHz	23MHz	6'b101101
47MHz	23.5MHz	6'b101110
48MHz	24MHz	6'b101111

The default value of XCKS= 6'b001010 for $OSCin= 11$ MHz at 12T mode.

CKCON3: Clock Control Register 3

Address=8FH, read/write, por+RESET=xxxx-xx0x

7	6	5	4	3	2	1	0
-	-	-	-	-	-	PWDEX	EN6TR

Bit 7~2: Reserved.

Bit 1: PWDEX, Power-down Exit Mode.

0: wake up from Power-down is internally timed.

1: wake up from Power-down is externally controlled.

Bit 0: EN6TR, Enable 6T mode control register. Default value is load from OR1.b0 inverted value. And it could be read/written by CPU. The access on this bit will affect CKCON2.EN6TR to corresponding operation and get same control function.

0: MG87FE/L2051/4051/6051 will run in 12T mode.

1: MG87FE/L2051/4051/6051 will run in 6T mode.

16. In System Programming (ISP)

IFD: ISP/IAP Flash Data Register

Address=E2H, read/write, RESET=1111-1111

7	6	5	4	3	2	1	0
Data							

IFD is the data port register for ISP/IAP operation. The data in IFD will be written into the desired address in operating ISP/IAP write and it is the data window of readout in operating ISP/IAP read.

If IMFT is indexed on IAPLB access, read/write IFD through SCMD flow will access the register content of IAPLB.

IFADRH: ISP/IAP Address for High-byte addressing

Address=E3H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
Address							

IFADRH is the high-byte address port for all ISP/IAP modes.

IFADRL: ISP/IAP Address for Low-byte addressing

Address=E4H, read/write, Power On + RESET=0000-0000

7	6	5	4	3	2	1	0
Address							

IFADRL is the low byte address port for all ISP/IAP modes. In page erase operation, it is ignored.

IFMT: ISP/IAP Flash Mode Table

Address=E5H, read/write, Power On + RESET=XXXX-0000

7	6	5	4	3	2	1	0
Reserved				Mode Selection			

Bit 7~4: Reserved

Bit 3~0: ISP/IAP operating mode selection

Bit[3:0]				Mode
0	0	0	0	Standby
0	0	0	1	AP-memory read
0	0	1	0	AP-memory program
0	0	1	1	AP-memory page erase
0	1	0	0	IAPLB write
0	1	0	1	IAPLB read

IFMT is used to select the flash mode for performing numerous ISP/IAP function.

IAPLB: IAP Low Boundary

Address=indirect, read/write, Power On + RESET=1111-1111

7	6	5	4	3	2	1	0
IAPLB							

Bit 7~0: The IAPLB determines the IAP-memory lower boundary. Since a Flash page has 512 bytes, the IAPLB must be an even number.

To read IAPLB, MCU need to define the IMFT for mode selection on IAPLB Read and set ISPCR.ISPEN. And then write 0x46h & 0xB9h sequentially into SCMD. The IAPLB content is available in IFD. If write IAPLB, MCU will put new IAPLB setting value in IFD firstly. And then select IMFT, enable ISPCR.ISPEN and then set SCMD. The IAPLB content has already finished the updated sequence.

The range of the IAP-memory is determined by IAPLB and the ISP start address as listed below.

*IAP lower boundary = IAPLBx256, and
IAP higher boundary = ISP start address – 1.*

For example, if IAPLB=0x12 and ISP start address is 0x1C00, then the IAP-memory range is located at 0x1200 ~ 0x1BFF.

Additional attention point, the IAP low boundary address must not be higher than ISP start address.

SCMD: Sequential Command Data register / RDID (Read DID register)

Address=E6H, read/write, Power On + RESET=xxxx-xxxx

7	6	5	4	3	2	1	0
SCMD							

SCMD is the command port for triggering ISP/IAP/IAPLB activity. If SCMD is filled with sequential 0x46h, 0xB9h and if ISPCR.7 = 1, ISP/IAP activity will be triggered.

ISPCR: ISP Control Register

Address=E5H, read/write, Power On + RESET= 0000-xxxx

7	6	5	4	3	2	1	0
ISPEN	SWBS	SWRST	CFAIL	-			

Bit 7: ISPEN, ISP/IAP operation enable.

0: Global disable all ISP/IAP program/erase/read function.

1: Enable ISP/IAP program/erase/read function.

Bit 6: SWBS, software boot selection control.

0: Boot from main-memory after reset.

1: Boot from ISP memory after reset.

Bit 5: SWRST, software reset trigger control.

0: No operation

1: Generate software system reset. It will be cleared by hardware automatically.

Bit 4: CFAIL, Command Fail indication for ISP/IAP operation.

0: The last ISP/IAP command has finished successfully.

1: The last ISP/IAP command fails. It could be caused since the access of flash memory was inhibited.

Bit 3~0 : Reserved.

MG87FE/L2051/4051/6051 does not make use of idle-mode to perform ISP operation. Instead, it creates CPU wait-state to release flash memory for ISP control circuit use. Once ISP run over, CPU will be waken-up and advanced to the instruction which follows the previous instruction that invokes ISP activity. During ISP operation, interrupt service is also blocked until ISP run over.

ISP control circuit has a built-in timer for timing sequence control. It is referred from OSC frequency and defined by CKCON2.XCKS[5:0] to get the accuracy erase/program timing.

17. In Application Programming (IAP)

MG87FE/L2051/4051/6051 available program memory size (AP-memory) is restricted to 2K(for 2051), 4K(for 4051), 6K(for 6051). The flash memory between IAPLB and ISP start address could be defined as data flash memory and can be accessed by the ISP operation in field application. The size of IAP flash memory is variable. It is defined by IAPLB.

When MG87FE/L2051/4051/6051 boots from AP-memory, it is restricted to have the capability of accessing IAP data flash memory space only. AP-memory and ISP-memory are protected from abnormal disturbance.

When MG87FE/L2051/4051/6051 boots from ISP-memory, AP-memory and data flash memory(IAP) are opened for ISP access.

17.1. Demo Code for IAP

As mentioned above, all the ISP modes can also be applied to the IAP operation. The demo codes for these modes are shown below.

Demo code for triggering the “Page Erase Mode”

```
MOV    ISPCR,#10000000b ;ISPCR.7=1, enable ISP
MOV    CKCON2,#00010001b ;CKCON2[5:0]=11h, suppose MCU running @11.0592MHz

MOV    IFMT,#03h        ;select Page Erase Mode

MOV    IFADRH,??        ;fill [IFADRH,IFADRL] with page address
MOV    IFADRL,??        ;! the page address must be within the IAP-memory

MOV    SCMD,#46h        ;trigger ISP processing
MOV    SCMD,#0B9h      ;

;Now, MCU will halt here until processing completed
```

Demo code for triggering the “Program Mode”

```
MOV    ISPCR,#10000000b ;ISPCR.7=1, enable ISP
MOV    CKCON2, #00010001b ;CKCON2[5:0]=11h, suppose MCU running @11.0592MHz

MOV    IFMT,#02h        ;select Program Mode

MOV    IFADRH,??        ;fill [IFADRH,IFADRL] with byte address
MOV    IFADRL,??        ;! the byte address must be within the IAP-memory
MOV    IFD,??           ;fill IFD with the data to be programmed

MOV    SCMD,#46h        ;trigger ISP processing
MOV    SCMD,#0B9h      ;

;Now, MCU will halt here until processing completed
```

Demo code for triggering the "Read Mode"

```
MOV    ISPCR,#10000000b ;ISPCR.7=1, enable ISP
MOV    CKCON2, #00010001b ;CKCON2[5:0]=11h, suppose MCU running @11.0592MHz

MOV    IFMT,#01h        ;select Read Mode

MOV    IFADRH,??        ;fill [IFADRH,IFADRL] with byte address
MOV    IFADRL,??        ;! the byte address must be within the IAP-memory

MOV    SCMD,#46h        ;trigger ISP processing
MOV    SCMD,#0B9h      ;

;Now, MCU will halt here until processing completed

MOV    A,IFD            ;now, the read data exists in IFD
...
```

18. Auxiliary SFRs

AUXR: Auxiliary Control Register

Address=8EH, read/write, RESET=0000-0000

7	6	5	4	3	2	1	0
INT3H	INT2H	P15FS	P14FS	P13FS	P12FS	P11PU	P10PU

Bit 7: INT3H, INT3 High/Rising trigger enable.

0: Remain INT3 triggered on low level or falling edge on P4.2.

1: Set INT3 triggered on high level or rising edge on P4.2.

Bit 6: INT2H, INT2 High/Rising trigger enable.

0: Remain INT2 triggered on low level or falling edge on P4.3.

1: Set INT2 triggered on high level or rising edge on P4.3.

Bit 5: P15FS, pin P1.5 function swapped enable.

0: Pin P1.5 and P3.5 reserves original default function.

1: Pin P1.5 function is swapped with P3.5/T1. And Pin P3.5 function is swapped by P1.5.

Bit 4: P14FS, pin P1.4 function swapped enable.

0: Pin P1.4 and P3.4 reserve original default function.

1: Pin P1.4 function is swapped with P3.4/T0. And Pin P3.4 function is swapped by P1.4.

Bit 3: P13FS, pin P1.3 function swapped enable.

0: Pin P1.3 and P4.3 reserve original default function.

1: Pin P1.3 function is swapped with P4.3/INT2. And Pin P4.3 function is swapped by P1.3 if internal OSC is enabled to release XTAL1 for GPIO function.

Bit 2: P12FS, pin P1.2 function swapped enable.

0: Pin P1.2 and P4.2 reserve original default function.

1: Pin P1.2 function is swapped with P4.2/INT3. And Pin P4.2 function is swapped by P1.2 if internal OSC is enabled to release XTAL2 for GPIO function.

Bit 1: P11PU, Enable P1.1 pull-up resistor.

0: P1.1 without Pull-Up resistor in open-drain mode.

1: P1.1 with Pull-Up resistor in open-drain mode.

Bit 0: P10PU, Enable P1.0 pull-up resistor.

0: P1.0 without Pull-Up resistor in open-drain mode.

1: P1.0 with Pull-Up resistor in open-drain mode.

P1.1 & P1.0 is high-impedance input and N-MOS output without pull-up resistor in default mode. P11PU & P10PU in AUXR will enable the pull-up resistor on P1.1/P1.0 individually. If P1.1 & P1.0 are used for GPIO function, CPU could not drive low without external pull-up resistor in power down mode when P11PU & P10PU are enabled.

AUXR1: Auxiliary Control Register 1

Address=A2H, read/write, Power On + RESET=0XXX-0XXX

7	6	5	4	3	2	1	0
P14FD	-	-	-	GF2	-	-	-

Bit 7: P14FD, Enable P14 output with fast driving.

0: P14 has normal driving on output state.

1: Enable P14 output with fast driving.

Bit 6~4: Reserved. **User should always write "0"**

Bit 3: GF2, General purpose Flag 2.

Bit 2~0: Reserved. **User should always write “0”**

19. Option Setting

LOCK	When enabled, dump ROM code function was disabled
HWBS	When enabled, MCU will boot from ISP-memory if ISP-memory is configured after power up.
HWBS2	When enabled, MCU will boot from ISP-memory if ISP-memory is configured after reset by RESET pin.
OSCDN	When enabled, the gain of crystal oscillator is reduced. It will helpful in EMI reduction. Regarding application not needing high frequency clock, it is recommended for system clock under 40MHz..
EN6T	When enabled, MCU will run at 6T mode.

20. Absolute Maximum Rating

For MG87FE2051/4051/6051

Parameter	Rating	Unit
Ambient temperature under bias	-55 ~ +125	°C
Storage temperature	-65 ~ + 150	°C
Voltage on any Port I/O Pin or RESET with respect to Ground	-0.5 ~ VDD + 0.5	V
Voltage on VDD with respect to Ground	-0.5 ~ +6.0	V
Maximum total current through VDD and Ground	400	mA
Maximum output current sunk by any Port pin	40	mA

*Note: stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the devices at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

For MG87FL2051/4051/6051

Parameter	Rating	Unit
Ambient temperature under bias	-55 ~ +125	°C
Storage temperature	-65 ~ + 150	°C
Voltage on any Port I/O Pin or RESET with respect to Ground	-0.3 ~ VDD + 0.3	V
Voltage on VDD with respect to Ground	-0.3 ~ +4.2	V
Maximum total current through VDD and Ground	400	mA
Maximum output current sunk by any Port pin	40	mA

*Note: stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the devices at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

21. Electrical Characteristics

21.1. DC Characteristics

VSS = 0V, TA = 25 °C, VDD = 5.0V and 12 clocks per machine cycle, unless otherwise specified

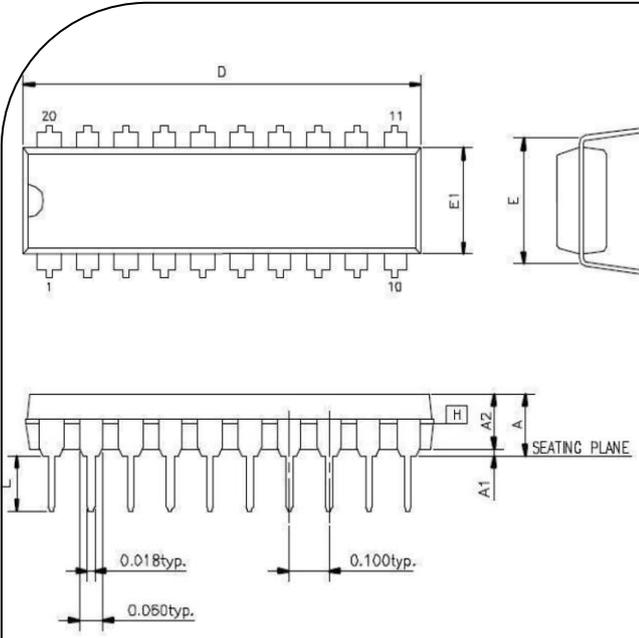
Symbol	Parameter	Test Condition	Limits			Unit
			min	Typ.	max	
V _{IH1}	Input High voltage (Ports 1, 3, 4)		2.0			V
V _{IH2}	Input High voltage (RESET)		3.5			V
V _{IL1}	Input Low voltage (Ports 1, 3, 4)				0.8	V
V _{IL2}	Input Low voltage (RESET)				1.6	V
I _{IH}	Input High Leakage current (Ports 1, 3, 4)	V _{PIN} = VDD		0	10	uA
I _{IL}	Logic 0 input current (Ports 1, 3, 4)	V _{PIN} = 0.4V		20	50	uA
I _{H2L}	Logic 1 to 0 input transition current (Ports 1, 3, 4)	V _{PIN} = 1.8V		250	500	uA
I _{OH1}	Output High current (Ports 1, 3, 4)	V _{PIN} = 2.4V	150	220		uA
I _{OL1}	Output Low current (Ports 1, 3, 4)	V _{PIN} = 0.4V	12			mA
I _{OP}	Operating current	F _{OSC} = 12MHz		8	16	mA
		F _{OSC} = 24MHz		10	20	
I _{IDLE}	Idle mode current	F _{OSC} = 12MHz		4	8	mA
		F _{OSC} = 24MHz		5	10	
I _{PD}	Power down current			0.1	10	uA
R _{RST}	Internal reset pull-down resistance			100		Kohm

VSS = 0V, TA = 25 °C, VDD = 3.3V and 12 clocks per machine cycle, unless otherwise specified

Symbol	Parameter	Test Condition	Limits			Unit
			min	Typ.	max	
V _{IH1}	Input High voltage (Ports 1, 3, 4)		2.0			V
V _{IH2}	Input High voltage (RESET)		2.8			V
V _{IL1}	Input Low voltage (Ports 1, 3, 4)				0.8	V
V _{IL2}	Input Low voltage (RESET)				1.5	V
I _{IH}	Input High Leakage current (Ports 1, 3, 4)	V _{PIN} = VDD		0	10	uA
I _{IL}	Logic 0 input current (Ports 1, 3, 4)	V _{PIN} = 0.4V		7	30	uA
I _{H2L}	Logic 1 to 0 input transition current (Ports 1, 3, 4)	V _{PIN} = 1.8V		100	250	uA
I _{OH1}	Output High current (Ports 1, 3, 4)	V _{PIN} = 2.4V	40	70		uA
I _{OL1}	Output Low current (Ports 1, 3, 4)	V _{PIN} = 0.4V	8			mA
I _{OP}	Operating current	F _{OSC} = 12MHz		6	12	mA
		F _{OSC} = 24MHz		8	16	
I _{IDLE}	Idle mode current	F _{OSC} = 12MHz		2	4	mA
		F _{OSC} = 24MHz		2.5	5	
I _{PD}	Power down current			0.1	50	uA
R _{RST}	Internal reset pull-down resistance			200		Kohm

22. Package Dimension

PDIP-20



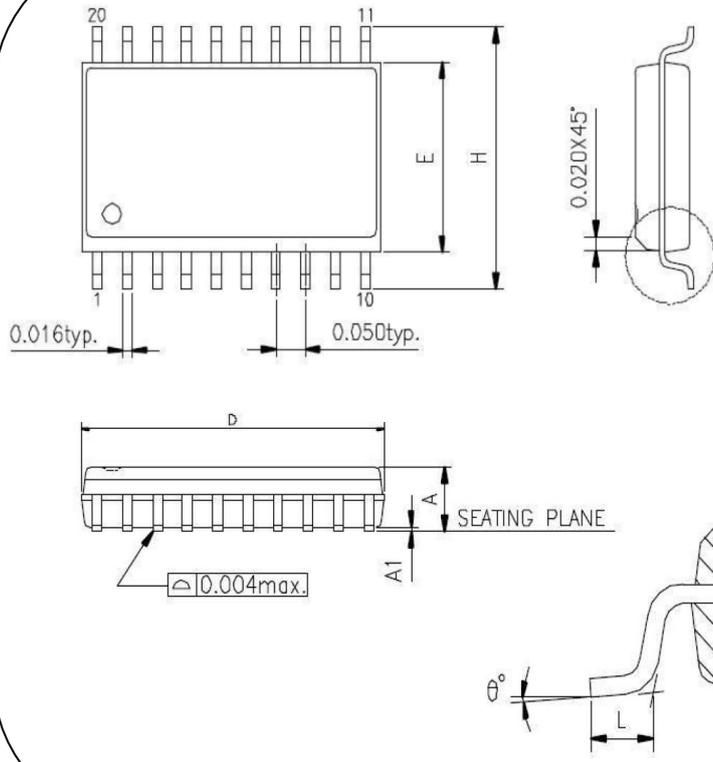
- NOTES:
1. JEDEC OUTLINE : MS-001 AD
 2. "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
 3. eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
 4. POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
 5. DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
 6. DATUM PLANE \square COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY

PDIP-20

Symbol	MIN		NOR		MAX	
A					5.334	0.210
A1	0.381	0.015				
A2	3.175	0.125	3.302	0.130	3.429	0.135
D	24.892	0.980	26.162	1.030	26.924	1.060
E	7.620 / 0.300 BSC					
E1	6.223	0.245	6.350	0.250	6.477	0.255
L	2.921	0.115	3.302	0.130	3.810	0.150
eB	8.509	0.335	9.017	0.355	9.525	0.375
θ°	0°		7°		15°	

mm/inch

SOP-20



- NOTES:
1. JEDEC OUTLINE : MS-013 AC
 2. DIMENSIONS "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED .15mm (.006in) PER SIDE.
 3. DIMENSIONS "E" DOES NOT INCLUDE INTER-LEAD FLASH, OR PROTRUSIONS. INTER-LEAD FLASH AND PROTRUSIONS SHALL NOT EXCEED .25mm (.010in) PER SIDE.

SOP-20

Symbol	MIN		MAX	
A	2.362	0.093	2.642	0.104
A1	0.102	0.004	0.305	0.012
D	12.598	0.496	12.903	0.508
E	7.391	0.291	7.595	0.299
H	10.008	0.394	10.643	0.419
L	0.406	0.016	1.270	0.05
θ°	0°		8°	

mm/inch

23. Instruction Set

MNEMONIC	DESCRIPTION	BYTE	EXECUTION TIME(MC)
DATA TRASFER			
MOV A,Rn	Move register to Acc	1	1
MOV A,direct	Move direct byte o Acc	2	1
MOV A,@Ri	Move indirect RAM to Acc	1	1
MOV A,#data	Move immediate data to Acc	2	1
MOV Rn,A	Move Acc to register	1	1
MOV Rn,direct	Move direct byte to register	2	2
MOV Rn,#data	Move immediate data to register	2	1
MOV direct,A	Move Acc to direct byte	2	1
MOV direct,Rn	Move register to direct byte	2	2
MOV direct,direct	Move direct byte to direct byte	3	2
MOV direct,@Ri	Move indirect RAM to direct byte	2	2
MOV direct,#data	Move immediate data to direct byte	3	2
MOV @Ri,A	Move Acc to indirect RAM	1	1
MOV @Ri,direct	Move direct byte to indirect RAM	2	2
MOV @Ri,#data	Move immediate data to indirect RAM	2	1
MOV DPTR,#data16	Load DPTR with a 16-bit constant	3	2
MOVC A,@A+DPTR	Move code byte relative to DPTR to Acc	1	2
MOVC A,@A+PC	Move code byte relative to PC to Acc	1	2
MOVX A,@Ri	Move external RAM(8-bit address) to Acc	1	2
MOVX A,@DPTR	Move external RAM(16-bit address) to Acc	1	2
MOVX @Ri,A	Move Acc to external RAM(8-bit address)	1	2
MOVX @DPTR,A	Move Acc to external RAM(16-bit address)	1	2
PUSH direct	Push direct byte onto Stack	2	2
POP direct	Pop direct byte from Stack	2	2
XCH A,Rn	Exchange register with Acc	1	1
XCH A,direct	Exchange direct byte with Acc	2	1
XCH A,@Ri	Exchange indirect RAM with Acc	1	1
XCHD A,@Ri	Exchange low-order digit indirect RAM with Acc	1	1
ARITHMETIC OPERATIONS			
ADD A,Rn	Add register to Acc	1	1
ADD A,direct	Add direct byte to Acc	2	1
ADD A,@Ri	Add indirect RAM to Acc	1	1
ADD A,#data	Add immediate data to Acc	2	1
ADDC A,Rn	Add register to Acc with Carry	1	1
ADDC A,direct	Add direct byte to Acc with Carry	2	1
ADDC A,@Ri	Add indirect RAM to Acc with Carry	1	1
ADDC A,#data	Add immediate data to Acc with Carry	2	1
SUBB A,Rn	Subtract register from Acc with borrow	1	1
SUBB A,direct	Subtract direct byte from Acc with borrow	2	1
SUBB A,@Ri	Subtract indirect RAM from Acc with borrow	1	1
SUBB A,#data	Subtract immediate data from Acc with borrow	2	1
INC A	Increment Acc	1	1
INC Rn	Increment register	1	1

INC direct	Increment direct byte	2	1
INC @Ri	Increment indirect RAM	1	1
DEC A	Decrement Acc	1	1
DEC Rn	Decrement register	1	1
DEC direct	Decrement direct byte	2	1
DEC @Ri	Decrement indirect RAM	1	1
INC DPTR	Increment DPTR	1	2
MUL AB	Multiply A and B	1	4
DIV AB	Divide A by B	1	4
DA A	Decimal Adjust Acc	1	1
LOGIC OPERATION			
ANL A,Rn	AND register to Acc	1	1
ANL A,direct	AND direct byte to Acc	2	1
ANL A,@Ri	AND indirect RAM to Acc	1	1
ANL A,#data	AND immediate data to Acc	2	1
ANL direct,A	AND Acc to direct byte	2	1
ANL direct,#data	AND immediate data to direct byte	3	2
ORL A,Rn	OR register to Acc	1	1
ORL A,direct	OR direct byte to Acc	2	1
ORL A,@Ri	OR indirect RAM to Acc	1	1
ORL A,#data	OR immediate data to Acc	2	1
ORL direct,A	OR Acc to direct byte	2	1
ORL direct,#data	OR immediate data to direct byte	3	2
XRL A,Rn	Exclusive-OR register to Acc	1	1
XRL A,direct	Exclusive-OR direct byte to Acc	2	1
XRL A,@Ri	Exclusive-OR indirect RAM to Acc	1	1
XRL A,#data	Exclusive-OR immediate data to Acc	2	1
XRL direct,A	Exclusive-OR Acc to direct byte	2	1
XRL direct,#data	Exclusive-OR immediate data to direct byte	3	2
CLR A	Clear Acc	1	1
CPL A	Complement Acc	1	1
RL A	Rotate Acc Left	1	1
RLC A	Rotate Acc Left through the Carry	1	1
RR A	Rotate Acc Right	1	1
RRC A	Rotate Acc Right through the Carry	1	1
SWAP A	Swap nibbles within the Acc	1	1
BOOLEAN VARIABLE MANIPULATION			
CLR C	Clear Carry	1	1
CLR bit	Clear direct bit	2	1
SETB C	Set Carry	1	1
SETB bit	Set direct bit	2	1
CPL C	Complement Carry	1	1
CPL bit	Complement direct bit	2	1
ANL C,bit	AND direct bit to Carry	2	2
ANL C,/bit	AND complement of direct bit to Carry	2	2
ORL C,bit	OR direct bit to Carry	2	2
ORL C,/bit	OR complement of direct bit to Carry	2	2
MOV C,bit	Move direct bit to Carry	2	1
MOV bit,C	Move Carry to direct bit	2	2
JC rel	Jump if Carry is set	2	2

JNC rel	Jump if Carry not set	2	2
BOOLEAN VARIABLE MANIPULATION			
JB bit,rel	Jump if direct bit is set	3	2
JNB bit,rel	Jump if direct bit not set	3	2
JBC bit,rel	Jump if direct bit is set and then clear bit	3	2
PROGRAM BRACHING			
ACALL addr11	Absolute subroutine call	2	2
LCALL addr16	Long subroutine call	3	2
RET	Return from subroutine	1	2
RETI	Return from interrupt subroutine	1	2
AJMP addr11	Absolute jump	2	2
LJMP addr16	Long jump	3	2
SJMP addr16	Short jump	2	2
JMP @A+DPTR	Jump indirect relative to DPTR	1	2
JZ rel	Jump if Acc is zero	2	2
JNZ rel	Jump if Acc not zero	2	2
CJNE A,direct,rel	Compare direct byte to Acc and jump if not equal	3	2
CJNE A,#data,rel	Compare immediate data to Acc and jump if not equal	3	2
CJNE Rn,#data,rel	Compare immediate data to register and jump if not equal	3	2
CJNE @Ri,#data,rel	Compare immediate data to indirect RAM and jump if not equal	3	2
DJNZ Rn,rel	Decrement register and jump if not equal	2	2
DJNZ direct,rel	Decrement direct byte and jump if not equal	3	2
NOP	No Operation	1	1

Disclaimers

Herein, Megawin stands for “*Megawin Technology Co., Ltd.*”

Life Support — This product is not designed for use in medical, life-saving or life-sustaining applications, or systems where malfunction of this product can reasonably be expected to result in personal injury. Customers using or selling this product for use in such applications do so at their own risk and agree to fully indemnify Megawin for any damages resulting from such improper use or sale.

Right to Make Changes — Megawin reserves the right to make changes in the products - including circuits, standard cells, and/or software - described or contained herein in order to improve design and/or performance. When the product is in mass production, relevant changes will be communicated via an Engineering Change Notification (ECN).