1.Feature

ROM: 2K x 14 bits

RAM: 96 x 8 bits

STACK: 4 Levels

I/O ports: 20 I/O PAD

Timer/counter: 8bits x1 (TMR0)

Prescaler: 8 Bits

Two IRQ sources: Internal IRQ: (TMR0)
External IRQ: (PA_0)

Watchdog Timer: On chip WDT is based on internal RC oscillator. The shortest period is 20mS; user can extend the WDT overflow period to 2.6S by using prescaler.

Power-On Reset & Power-Down Reset

Reset Timer: 20 mS (5V)

Four external Oscillate modes: RC,LP Crystal,NT Crystal and HS Crystal.

Two operation modes: General mode, and Advanced mode

Operation Voltage: 2.2V 5.5V

Instruction set: 79

Wake-up: Watchdog timer overflow, Port A (PA_3~ PA_0)

Reset vector: 7FFH

IRQ vector: 7FEH

Low voltage reset: voltage shortage will result in reset

2. Pin Definition & Pad Assignment

RTCC	1	28	RESETB/VPP
VDD	2	27	OSC_1
NC	3	26	OSC_2
VSS	4	25	PC_7
NC	5	24	PC_6
PA_0	6	23	PC_5
PA_1	7	22	PC_4
PA_2	8	21	PC_3
PA_3	9	20	PC_2
PB_0	10	19	PC_1
PB_1	11	18	PC_0
PB_2	12	17	PB_7
PB_3	13	16	PB_6
PB_4	14	15	PB_5

Package Types of 28Pin: DIP, SOP.

RESETB/VPP	1	20	OSC_1
RTCC	2	19	OSC_2
VDD	3	18	PC_3
VSS	4	17	PC_2
PA_0	5	16	PC_1
PA_1	6	15	PC_0
PB_0	7	14	PB_7
PB_1	8	13	PB_6
PB_2	9	12	PB_5
PB_3	10	11	PB_4

Package Types of 20Pin: DIP, SOP.

PIN description

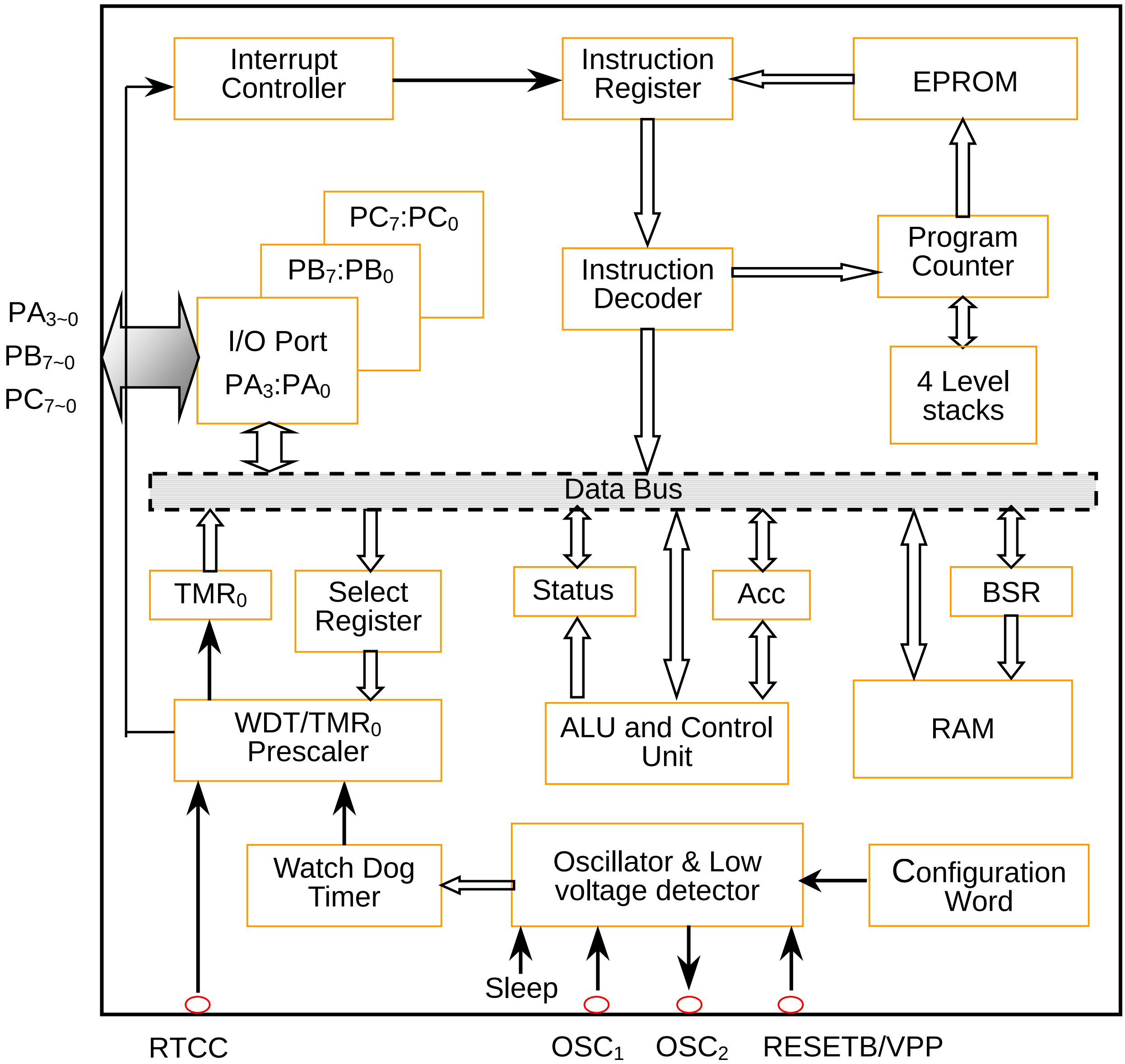
Pin name	I/O	Description
RTCC	I	External clock input to TMR0 counter
PA_0	I/O	I/O port & External IRQ input & wake-up input
PA_{3-1}	I/O	I/O port & wake-up (input mode)
PB_{7-0}	I/O	I/O port
PC_{7-0}	I/O	I/O port
RESETB/VPP	I	System reset signal & VPP (High voltage) input 1 Low voltage: reset mode 2 High voltage: programming mode
OSC_1	I	Oscillator input
OSC_2	O	Oscillator output
VDD	P	Power input
VSS	P	Ground input

I: Input; O: Output; I/O: Bi-direction; P: Power

3. Control Register

Name	Addr	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CONFIG (Instruction)			LV_1	LV_0	TYPE	CPT	WDTE	$FOSC_1$	$FOSC_0$
SELECT				SUR_0	$EDGE_0$	PSA	PS_2	PS_1	PS_0
IAR	\$00	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
TMR0	\$01	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
PC	\$02	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
STATUS	\$03		SA_1	SA_0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
BSR	\$04	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
I/O $Port_A$	\$05					PA_3	PA_2	PA_1	PA_0
I/O $Port_B$	\$06	PB_7	PB_6	PB_5	PB_4	PB_3	PB_2	PB_1	PB_0
I/O $Port_C$	\$07	PC_7	PC_6	PB_5	PC_4	PC_3	PC_2	PC_1	PC_0
WAKE_UP	\$20	WDTS	WUE	EIS		PUH_3	PUH_2	PUH_1	PUH_0
IRQM	\$21	INTM					EXINTM		TMR0M
IRQF	\$22						EXINTF		TMR0F

4. System Block Diagram



5. Memory Map

TM58P20 memory is organized into program memory and data memory.

5.1 Program memory

TM58P20 provides 2 program memory maps, general mode and advanced mode. User can select different mode by setting configuration word.

In general mode, there are only 512 words of the same page that can be directly addressed. Extra program memory can be addressed by setting bit 6~5 of status register. The sequence of instructions is controlled via the program counter (PC), which automatically increases 1. However, the sequence can be changed by “skip”, “call” and “goto” instructions or by moving data to the PC.

In advanced mode, TM58P20 allow directly goto any address in 2K memories without limited by page size. In addition, “lcall” and “lgoto” instructions are employed to provide flexible addressing mode.

TM58P20 has an 11-bits program counter capable of accessing 2K spaces. If accessing address has over 2K, then the address will map to physical 2K memories, i.e. 2K+M will be mapped to M. A NOP at the reset vector location will cause a restart at address 000h. A simple map to induce illustrate ROM organization is shown in figures 5-1.

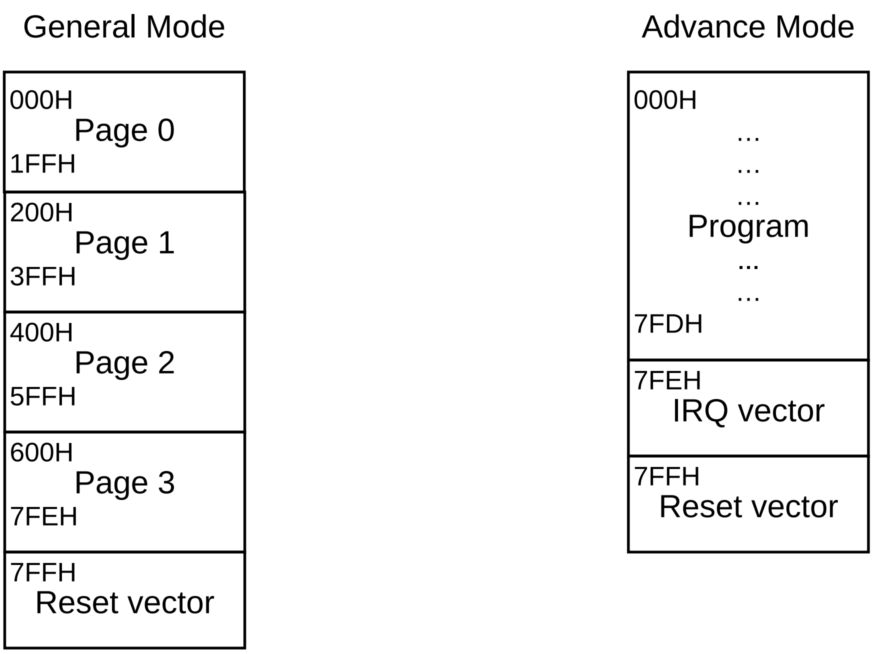


Figure 5-1 The ROM Organization

TM58P20 only provide IRQ function in advanced mode. In this mode, the address 7FEH is reserved for IRQ vector. User can operate advanced mode by setting configuration word. The configuration word is located 800H that contains OSC selection, WDT enable, code protection, operate type selection and low voltage reset selection.

Bit	Symbol	Description			
1~0	FOSC$_1$~FOSC$_0$	Bit$_1$	Bit$_0$	OSC Type	Resonance Frequency
		0	0	LP (low speed)	32~200K hz
		0	1	NT(Normal speed)	200K~10M hz
		1	0	HS (high speed)	10~20M hz
		1	1	RC	32K ~ 6M hz
2	WDTE	WDTE: Watchdog enable/disable control 1: WDT enable 0: WDT disable			
3	CPT	CPT: Code Protection bit 1: OFF 0: ON			
4	TYPE	TYPE: Select operating mode 1: Advanced mode 0: General mode			
6~5	LV$_1$~LV$_0$	LV$_1$	LV$_0$	Detect voltage	
		1	1	Don't use	
		0	1	Don't use	
		1	0	2V	
		0	0	4V	

Figure 5-2 The Configuration Word

5.2 Data memory

Data memory is composed of special function registers and general-purpose ram. The size of data memory is not stationary, it depends on bit 4 of configuration word (general or advanced mode).

5.2.1 General Mode

In general mode, TM58P20 has 72 general-purpose registers that accessed by using a bank select scheme. The special function registers include the program counter (PC), the timer (TMR0) register, the status register, the bank select register, and the I/O port registers. Furthermore, TM58P20 has 3 auxiliary registers that include indirect addressing register (IAR), the select register (Select) and the I/O direction register (IODIR). The register map of general mode is shown in figure 5-3.

<table>
<tr><th></th><th>Bank0</th><th>Bank1</th><th>Bank2</th><th>Bank3</th></tr>
<tr><td>00h</td><td>IAR</td><td colspan="3" rowspan="9">Map back to address in Bank0</td></tr>
<tr><td>01h</td><td>TMR0</td></tr>
<tr><td>02h</td><td>PC</td></tr>
<tr><td>03h</td><td>STATUS</td></tr>
<tr><td>04h</td><td>BSR</td></tr>
<tr><td>05h</td><td>PORTA</td></tr>
<tr><td>06h</td><td>PORTB</td></tr>
<tr><td>07h</td><td>PORTC</td></tr>
<tr><td>08h~0fh</td><td>General Purpose Register</td></tr>
<tr><td>8+16*4=72</td><td>General Purpose Register 10-1F</td><td>General Purpose Register 30-3F</td><td>General Purpose Register 50-5F</td><td>General Purpose Register 70-7F</td></tr>
</table>

Figure 5-3 The Register Map of General Mode

A. The IAR (indirect addressing register) is not a physical register and is used to assist BSR with indirect addressing. Any instruction attempts to access IAR actually mapping to another address that is pointed by BSR. Since IAR is not a material circuit, user reads IAR itself (BSR=00H) will always return 00h at data bus. Writing to IAR itself will like NOP.

B. Select register is used to control WDT and TMR0. It has not assigned a specific address in data memory and can only set control bits by select instruction, i.e. it is write-only register. The context of accumulator will be sent to the select register by executing the select instruction. If select register has never set by program, its default value is 3FH. We drew Figure 5-4 to explain how to set select register.

Bit	Symbol	Description				
		PS_2	PS_1	PS_0	TMR0 rate	WDT rate
2~0	PS_2~PS_0	0	0	0	1:2	1:1
		0	0	1	1:4	1:2
		0	1	0	1:8	1:4
		0	1	1	1:16	1:8
		1	0	0	1:32	1:16
		1	0	1	1:64	1:32
		1	1	0	1:128	1:64
		1	1	1	1:256	1:128
3	PSA	PSA: Prescaler assignment bit 1: Prescaler assigned to WDT 0: Prescaler assigned to TMR0				
4	$EDGE_0$	$EDGE_0$: TMR0 source signal edge control bit 1:increment when H→L transition on external clock 0:increment when L→H transition on external clock				
5	SUR_0	SUR_0: TMR0 clock source bit 1: External clock input 0: (Internal clock)/4 or internal instruction cycle				

Figure 5-4 Select Register

C. The I/O Direction control register is similar to the Select register that is write-only register. To set an I/O port pin as input, the corresponding direction control bit must be high. Similarly, the zero represents output. Any direction control bit can be programmed individually as input or output by using "IODIR" instruction. If the register is not programmed, than all I/O ports always keep input mode.

- PC (program counter) is 11-bit wide binary counter and increases itself for every instruction cycle, except the following instructions.
  1. "call", "goto", "lgoto" and "lcall": the label will move to PC
  2. "retla", "reti" and "ret": the top value of stack will pop to PC

  Incrementing PC when it changes to the next higher page. It should be noted that the page select bits in the status register would not be changed synchronously. The following GOTO, CALL, or MOVAM 02H will return to the previous page, unless the page select bits have been updated in program. In order to reduce the complexity of programming, TM58P20 provides 2 instructions to facilitate subroutine call and branch handling which are LCALL and LGOTO. LCALL and LGOTO can address to anywhere in the ROM, but the page select bits are unnecessary. The attached operands of CALL and GOTO are 8-bit and 9-bit respectively, and so need extra bits (page select bits) to address whole memory. However, LCALL and LGOTO have 11-bit wide operands that are easy to address the total ROM space.

- TMR0 is 8-bit wide binary counter/timer. This register increases by an external signal edge applied to RTCC pin, or by internal instruction cycle. It has the following features.

A. Readable and writeable
B. Synchronize with 2 internal clocks
C. Can use programmable prescaler by setting select register

The other details will be described in follow-up chapter.

- Status register contains page select bits, time out bit, power down bit and the status of ALU. Please note that $\overline{TO}$ and $\overline{PD}$ are controlled by hardware and unchangeable by program.

<table>
<tr><th>Bit</th><th>Symbol</th><th colspan="4">Description</th></tr>
<tr><td rowspan="3">0</td><td rowspan="3">C</td><td colspan="4">Carry and $\overline{Borrow}$ bit</td></tr>
<tr><td colspan="2">ADD instruction</td><td colspan="2">SUB instruction</td></tr>
<tr><td colspan="2">1: a carry occurred from the MSB
0: no carry</td><td colspan="2">1: no borrow (Note1)
0:a borrow occurred from the MSB</td></tr>
<tr><td rowspan="3">1</td><td rowspan="3">DC</td><td colspan="4">Nibble Carry and Nibble $\overline{Borrow}$ bit</td></tr>
<tr><td colspan="2">ADD instruction</td><td colspan="2">SUB instruction</td></tr>
<tr><td colspan="2">1: a carry from the low nibble bits of the result occurred
0: no carry</td><td colspan="2">1: no borrow
0: a borrow from the low nibble bits of the result occurred</td></tr>
<tr><td>2</td><td>Z</td><td colspan="4">Zero bit:
1: the result of a logic operation is zero
0: the result of a logic operation is not zero</td></tr>
<tr><td>3</td><td>$\overline{PD}$</td><td colspan="4">Power down flag bit: (Note2)
1: after power-on or by the CLRWDT instruction
0: execute SLEEP instruction</td></tr>
<tr><td>4</td><td>$\overline{TO}$</td><td colspan="4">Time out flag bit:
1: after power-on or by the CLRWDT or SLEEP instruction
0: Occur WDT time-overflow</td></tr>
<tr><td rowspan="2">6~5</td><td rowspan="2">SA_1~SA_0</td><td>SA1</td><td>SA2</td><td colspan="2">Page Location</td></tr>
<tr><td>0
0
1
1</td><td>0
1
0
1</td><td colspan="2">Page 0 (000H~1FFH)
Page 1 (200H~3FFH)
Page 2 (400H~5FFH)
Page 3 (600H~7FFH)</td></tr>
</table>

Figure 5-5 Status Register

Note1: A SUB instruction is executed by adding the 2's complement of the subtrahend, so C = 1 represents positive result. The Figure 5-5-1 show the relation between C-bit and borrow.

B0H－50H										50H－B0H									
	C	B7	B6	B5	B4	B3	B2	B1	B0		C	B7	B6	B5	B4	B3	B2	B1	B0
		1	0	1	1	0	0	0	0			0	1	0	1	0	0	0	0
＋		1	0	1	1	0	0	0	0	＋		0	1	0	1	0	0	0	0
＝	1	0	1	1	0	0	0	0	0	＝	0	1	0	1	0	0	0	0	0

Figure 5-5-1

Note2: The $\overline{TO}$ and $\overline{PD}$ bits are active low that can be used to determine different causes of reset. The Figure 5-5-2 illustrates the value of $\overline{TO}$ and $\overline{PD}$ after the relative reset events.

$\overline{TO}$	$\overline{PD}$	Reset Event
0	0	WDT time out from sleep mode
0	1	WDT time out from normal mode
1	0	Input a “low” at RESETB from sleep mode
1	1	Power on reset
Unchanged	Unchanged	Input a “low” at RESETB from normal mode

Figure 5-5-2

- BSR (bank select register) is associated with IAR to indirectly access the data memory. The direct addressing must rely on BSR to access bank1 ~ bank3, because there are only 5-bit wide address operands in general mode. The bit 6~5 of BSR are used to select the specifiable memory bank. These address regions 20H~2FH, 40H~4FH and 60H~6FH are not accessible, these address will be mapped to 00H~0FH ($Bank_0$). The addressing map is shown in Figure 5-6.

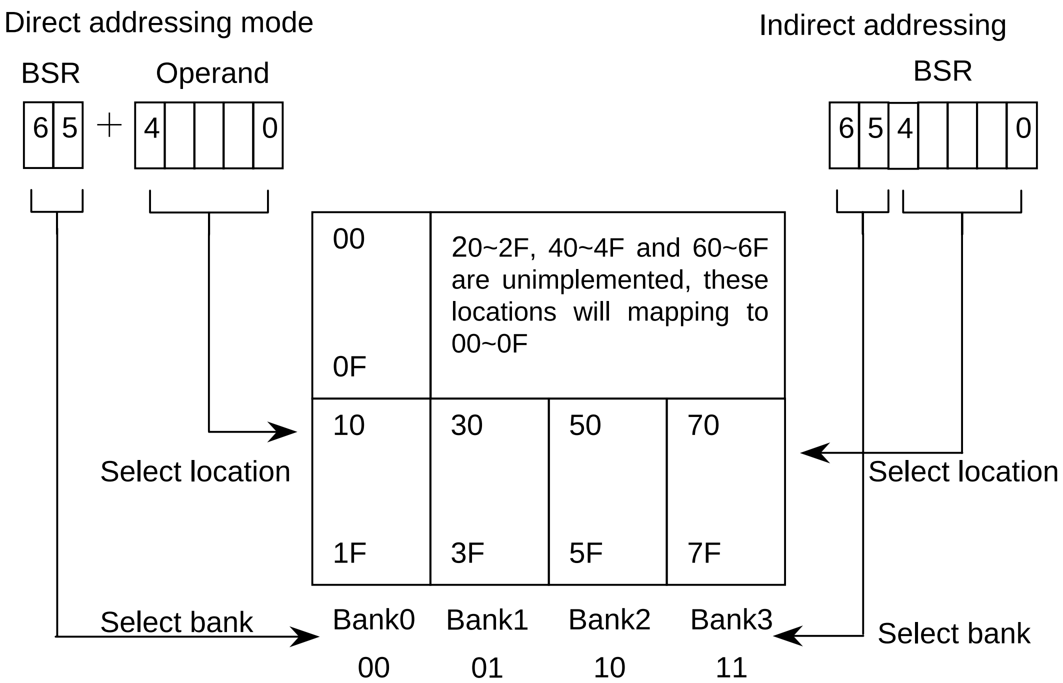


Figure 5-6 The Direct and Indirect Addressing Map

- Port A~C are programmable I/O ports. Please note that read I/O instruction always read the I/O pin even though the pin is output mode. On reset, all I/O pins were set as input mode until IODIR has been changed.

5.2.2 The advanced mode

In advanced mode, we provide IRQ, convenient wake up functions and flexible addressing mode. In addition to extend data memory, we increase 3 extra registers to support IRQ and wake_up. This section will introduce these increased control registers and characteristics. The data memory map of advanced mode and the addressing map are shown in figure 5-7 and figure 5-8.

Advanced mode (Type=1)

	00~1F	20~3F	41~5F	60~7F
00h	IAR	WAKE_UP	Unimplemented	
01h	TMR0	IRQM		
02h	PC	IRQF		
03h	STATUS	Unimplemented		
04h	BSR			
05h	PORTA			
06h	PORTB			
07h	PORTC			
8*4+16*4=96	General Purpose Register 08-0F	General Purpose Register 28-2F	General Purpose Register 48-4F	General Purpose Register 68-6F
	General Purpose Register 10-1F	General Purpose Register 30-3F	General Purpose Register 50-5F	General Purpose Register 70-7F

Figure 5-7 The Data Memory Map of Advanced Mode

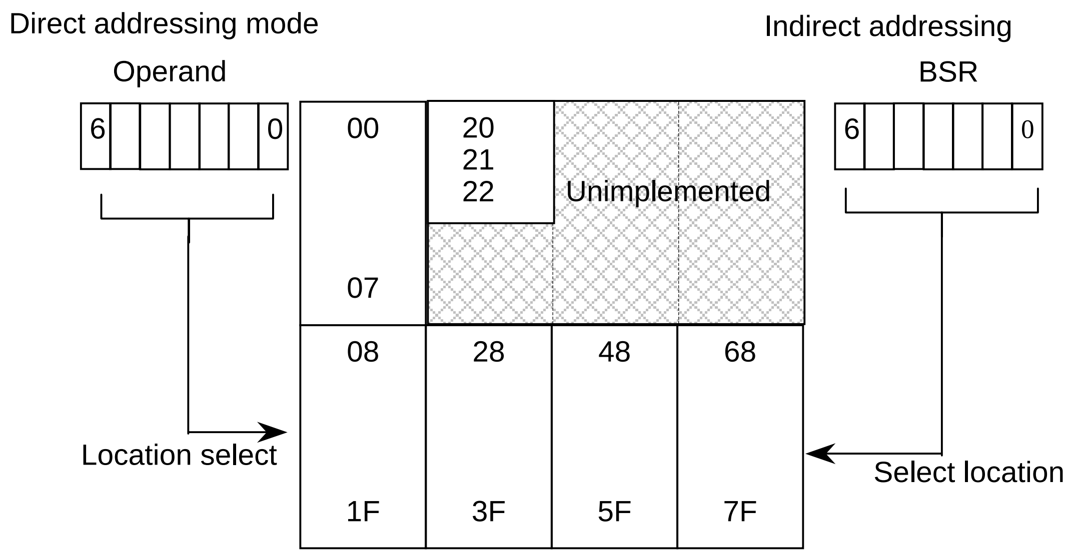


Figure 5-8 The Direct and Indirect Addressing Map

In advanced mode, we locate the increased 24 general-purpose registers on 28~2F, 48~4F and 68~6F which are shadow regions in Fig 5-7. The IRQ and the wake_up control registers (WAKE_UP, IRQM and IRQF) are assigned to 20, 21 and 22, respectively. In general mode, BSR<6,5> are the bank select bits and used to select the bank (00=bank0, 01=bank1, 10=bank2, 11=bank3). The lower 16 bytes of bank1, 2, and 3 are mapped to bank0. In advanced mode,TM58P20 allows 7-bit wide operand to access ram, operand<6:0> can address 00~7F directly. It doesn't need bank select bits, and reduces the complexity of programming

- The wake up control register (WAKE_UP) is used to set watchdog enable and distinguish between external wake-up signal and IRQ. On reset, all bits are defined as '0' that can be programming by software. The scheme of WAKE_UP register is shown in Fig 5-9.

Bit	Symbol	Description
7	WDTS	Watch Dog Timer Software Control bit: TM58P20 has 2 WDT control bits (WDTE and WDTS), WDTE is set in configuration word by hardware and WDTS is set in control register by software. If WDTS is valid only if WDTE has been set, i.e. WDTE has higher priority than WDTS. 1: enable 0: disable
6	WUE	Wake Up Enable bit: 0: don't support external wake-up 1: enable external wake-up function
5	EIS	External Interrupt Select: 1: set PA_0 as an external IRQ pin (Note3) 0: set PA_0 as a bi-directional I/O pin
4	----	Unimplemented
3~1	PUH_3~PUH_1	Pull High Port A bit3 ~1: 0: disable external wake up 1: if (WUE) & (PUH_N) & (input a falling edge signal at PA_N) then wake up chip from sleep. N can be 3, 2 or 1, but it must keep consistent.
0	PUH_0	Pull High Port A bit_0: 0: disable external wake up and external IRQ 1: if (WUE) & (PUH_0) & (input a falling edge signal at PA_0) then wake up chip from sleep. Or if (EIS) & (PUH_0) & (input a falling edge signal at PA_0) then generate an IRQ. Note: If PUH0, WUE and EIS are set as '1', then PA_0 is defined as IRQ input pin.

Figure 5-9 The Scheme of Wake_Up Register

Note3: The IRQ must execute at normal mode. If an IRQ is occurred at sleep model, then the IRQ routine will be performed until this chip has woken by external wake up signal. Other wake methods include (1) power on reset, (2) external reset and (3) WDT overflow (if enabled), the foregoing cases mean the IRQ ought to be abolished.

The Interrupt Mask register and Interrupt Flag register are used to control IRQ handling. TM58P20 supports $timer_0$ and external interrupt but nest-interrupt is not allowed. The schemes of the interrupt mask register and the interrupt flag register are shown in Fig 5-10 and 5-11, respectively.

Bit	Symbol	Description
7	INTM	Global enable bit: The bit has higher priority than EXINTM and TMR0M. 1: enable 0: disable By the way, the RETI instruction will set INTM as ‘1’.
6~3	----	Unimplemented
2	EXINTM	External Interrupt enable: 1:Enable Interrupt 0: Disable Interrupt
1	----	Unimplemented
0	TMR0M	TMR0 Interrupt enable: 1:Enable Interrupt 0: Disable Interrupt

Figure 5-10 Interrupt Mask register

Bit	Symbol	Description
7~3	----	Unimplemented
2	EXINTF	External interrupt flag: 1: the External interrupt be requested by the external interface (Port A_0) (Note4)
1	----	Unimplemented
0	TMR0F	TMR0 interrupt flag: 1: TheTMR0 counter overflow generates an interrupt request.

Figure 5-11 Interrupt Flag register

Note 4: Both interrupt flags are set by hardware, software can only clear flags. It is useless that attempt writing ‘1’ to flag.

The debounce time is the interval that must pass before a second pressing of a key is accepted. User can set this interval with the delay routine (See Example 1).

Key bounce

```
interrup
;----------------------------------------
    btmss   irqf,2        ;; if external IRQ?
    lgoto   int_end
int_nt1                   ;; filter out key begin bounce
    btmsc   ra,0
    lgoto   int_nt1
int_loop1                 ;; filter out key end bounce
    call    delay         ;; worse case 30ms
    btmss   ra,0
    lgoto   int_loop1

    call    delay_routine    ;; such as 30ms
    btmss   ra,0
    lgoto   int_loop1
;-------------------------------------
    bcm     irqf,2
int_end
    reti
```

Example 1 Key_Debounce

6. Functional Description

6.1 TMR0 and Watchdog timer

Fig. 6-1 shows the block diagram of the TMR0/WDT prescaler. As shown in the figure, the prescaler register can be a pre-scaler for TMR0 or be a post-scaler for WDT.

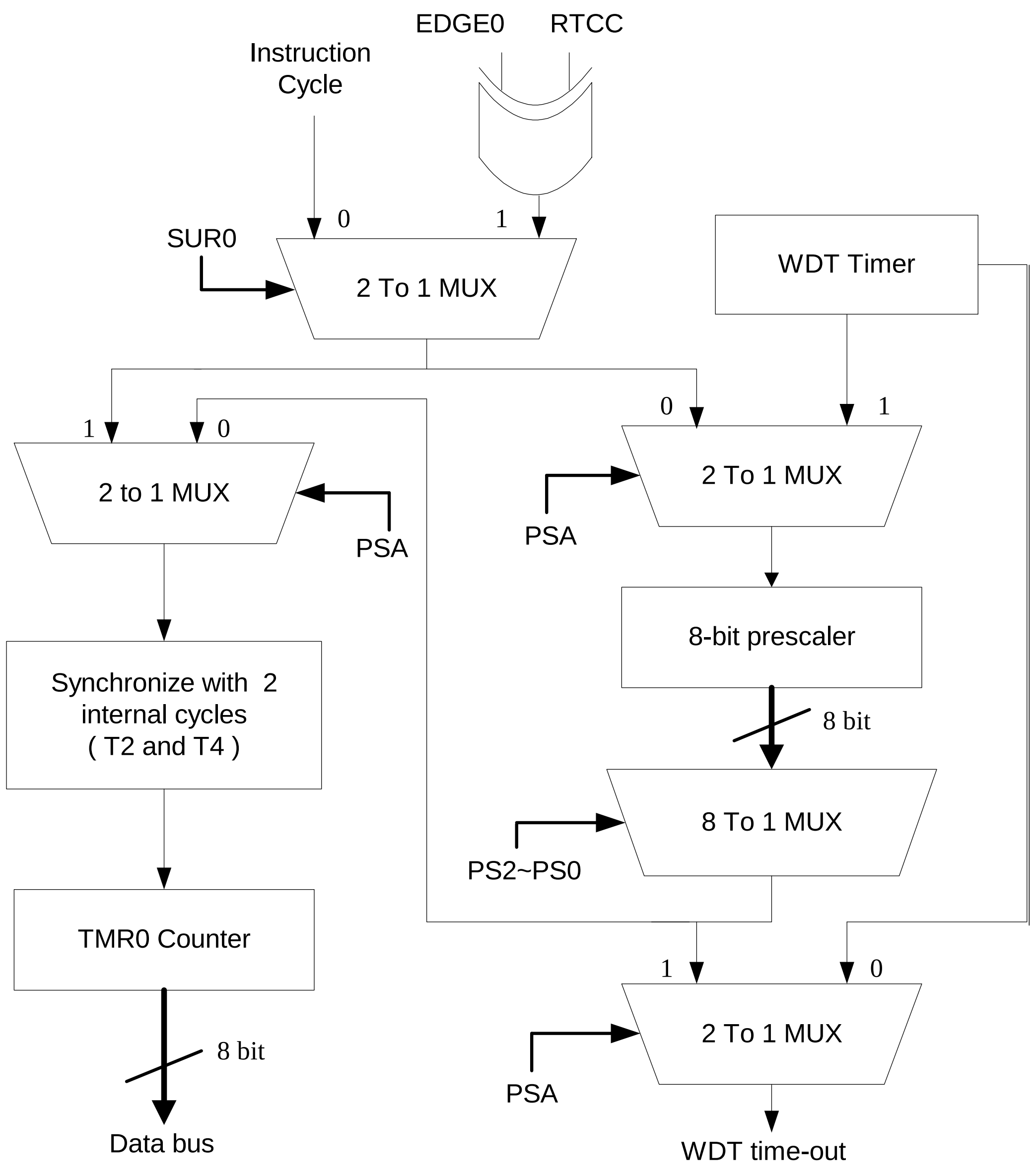


Figure 6-1 Block Diagram of the TMR0/WDT Prescaler

The TMR0 is an 8-bit timer/counter. The clock source of TMR0 can come from the instruction clock or the external clock.

A. To select the instruction clock, the SUR_0 bit of the select register should be clear. When no prescaler is used, TMR0 will increase by 1 at every instruction cycle.

B. To select the external clock, the SUR_0 bit of the select register should be set. In this mode, TMR0 relies on the $EDGE_0$ bit to determine that TMR0 is increased by 1 at every falling or rising edge. When an external clock is used for TMR0, a problem must be noted that the external clock synchronizes with internal clock. TM58P20 synchronizes external clock by sampling internal clock at T2 and T4. If external pulse is smaller than 2 internal cycles, the pulse maybe ignored. Therefore, the external clock must keep stable state (high or low) for at least 2 internal cycles.

The WDT counter is an 8-bit binary counter. The clock source of WDT is provided by an independent on-chip RC oscillator that does not need any external clock. Therefore, the WDT will keep counting even if the chip has slept already. A WDT time-out period which is typically 20ms, it will restart system and set the $\overline{TO}$ bit (bit4 of status register) as "0". The WDT time-out period vary with temperature, power voltage and process. This period can be improved via the prescaler. The maximum division ratio can up to 1:128 by setting PS2~PS0 as "111".

The prescaler can be assigned to either the TMR0 or the WDT via the PSA bit. Note that either WDT or TMR0 can employ the prescaler simultaneously. The following Example(2-3) must be executed when changing PSA form TMR0 to the WDT and form WDT to the TMR0 respectively. These examples can avoid an unintended time-out reset.

```
Clrwdt
Clrm      TMR0;  clear prescaler & TMR0
Movla     B'00xx1111
Select
Clrwdt
Movla     B'00xx1xxx; set prescaler to desired
Select               ; WDT rate
```

Example 2 Changing prescaler form TMR0 to WDT

```
Clrwdt   ; clear prescaler & WDT
Movla      B'00xx0xxx
Select   ; set prescaler to TMR0 with
         ; new rate
```

Example 3 Changing prescaler form WDT to TMR0

When the prescaler is assigned to WDT, “CLRWDT” and “SLEEP” instruction will clear the prescaler and the WDT. When the prescaler is assigned to TMR0, the prescaler will be cleared by any instruction that writes to TMR0.

6.2 Reset

TM58P20 may be reset by one of the following conditions:

(1) Power-on

(2) Power-down (circuit protection), refer to electrical character characteristic.

(3) RESETB/VPP pin input a negative pulse

(4) WDT timer out reset (if enabled).

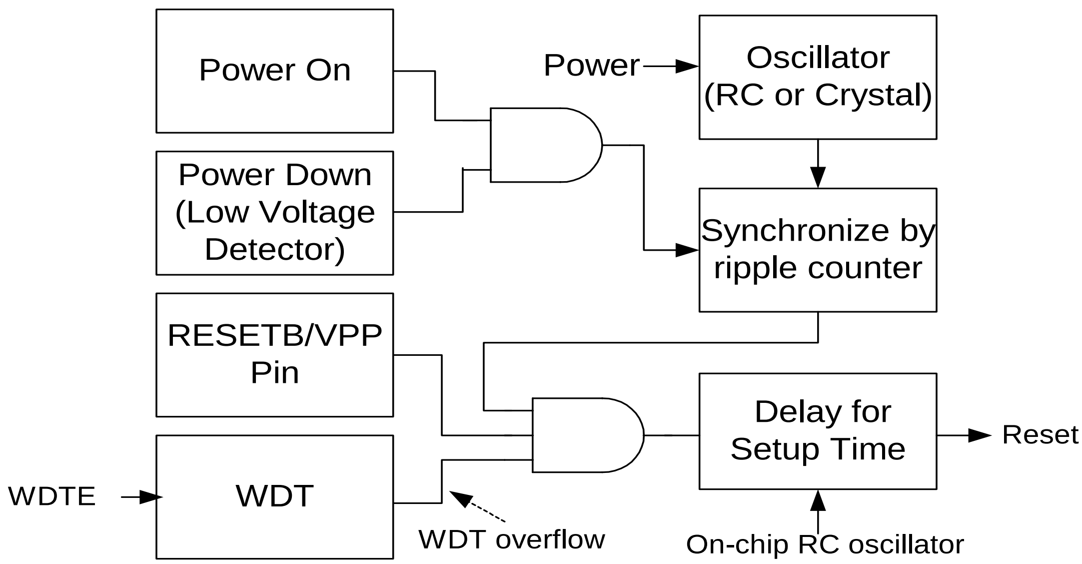


Figure 6-2 Scheme of the Reset Controller

As shown in the figure 6-2, four reset conditions are listed. The power-down event will cause TM58P20 to reset which the voltage ranges is according to the bit6~bit5 in the configuration word. This condition is used to protect chip in deficient power environment. The voltage ranges of power-down are defined in electrical characteristics. Furthermore, the ranges may be influenced by process and temperature variations. In general, we call the first two reset-cases as cold reset. The cold reset time may be too short for slow crystals and RC oscillators that require much longer than setup time (note) to oscillate. In order to insure the system is correct, the events should be synchronized with system clock.

Note: the setup time is approximately 20ms that will affect due to power voltage, process and temperature variations.

The last two cases are called warm reset. The different reset events will affect registers and ram. The $\overline{TO}$ and $\overline{PD}$ bits can be used to determine the type of reset. These relation are listed in figure 6-3

Address	Name	Cold Reset	Warm Reset
N/A	Accumulator	xxxx xxxx	pppp pppp
N/A	IODIR	1111 1111	1111 1111
N/A	Select	--11 1111	--11 1111
00h	IAR	---- ----	---- ----
01h	TMR0	xxxx xxxx	pppp pppp
02h	PC	111 1111 1111	111 1111 1111
03h	STATUS	0001 1xxx	000? ?ppp [1]
04h	BSR	-xxx xxxx	-ppp pppp
05h	PORTA	0000 xxxx	0000 pppp
06h	PORTB	xxxx xxxx	pppp pppp
07h	PORTC	xxxx xxxx	pppp pppp
20h	WAKE_UP	0000 0000	0000 0000
21h	IRQM	0000 0000	0000 0000 [2]
22h	IRQF	0000 0000	0000 0000
	General Purpose RAM	Xxxx xxxx	Pppp pppp

6-3 RESET CONDITIONS

X: unknown; P: previous data ; ?: value depends on condition ; -:unimplemented, read as “0”

7. Instruction Set

Mnemonic Operands	Instruction Code (Advance)	Cycles	Status Affected	OP-code
ADDAM M, m	(M)+(acc) → (M)	1	C, DC, Z	10 0101 1MMM MMMM
ADDAM M, a	(M)+(acc) → (acc)	1	C, DC, Z	10 0101 0MMM MMMM
ANDAM M, m	(M). (acc) → (M)	1	Z	10 0100 1MMM MMMM
ANDAM M, a	(M). (acc) → (acc)	1	Z	10 0100 0MMM MMMM
ANDLA I	Literal . (acc) → (acc)	1	Z	11 1001 iiii iiii
BCM M, b0	Clear bit0 of (M)	1	None	00 1100 0MMM MMMM
BCM M, b1	Clear bit1 of (M)	1	None	00 1100 1MMM MMMM
BCM M, b2	Clear bit2 of (M)	1	None	00 1101 0MMM MMMM
BCM M, b3	Clear bit3 of (M)	1	None	00 1101 1MMM MMMM
BCM M, b4	Clear bit4 of (M)	1	None	00 1110 0MMM MMMM
BCM M, b5	Clear bit5 of (M)	1	None	00 1110 1MMM MMMM
BCM M, b6	Clear bit6 of (M)	1	None	00 1111 0MMM MMMM
BCM M, b7	Clear bit7 of (M)	1	None	00 1111 1MMM MMMM
BSM M, b0	Set bit0 of (M)	1	None	00 1000 0MMM MMMM
BSM M, b1	Set bit1 of (M)	1	None	00 1000 1MMM MMMM
BSM M, b2	Set bit2 of (M)	1	None	00 1001 0MMM MMMM
BSM M, b3	Set bit3 of (M)	1	None	00 1001 1MMM MMMM
BSM M, b4	Set bit4 of (M)	1	None	00 1010 0MMM MMMM
BSM M, b5	Set bit5 of (M)	1	None	00 1010 1MMM MMMM
BSM M, b6	Set bit6 of (M)	1	None	00 1011 0MMM MMMM
BSM M, b7	Set bit7 of (M)	1	None	00 1011 1MMM MMMM
BTMSC M, b0	If bit0 of (M) = 0, skip next instruction	1 + (skip)	None	00 0100 0MMM MMMM
BTMSC M, b1	If bit1 of (M) = 0, skip next instruction	1 + (skip)	None	00 0100 1MMM MMMM
BTMSC M, b2	If bit2 of (M) = 0, skip next instruction	1 + (skip)	None	00 0101 0MMM MMMM
BTMSC M, b3	If bit3 of (M) = 0, skip next instruction	1 + (skip)	None	00 0101 1MMM MMMM
BTMSC M, b4	If bit4 of (M) = 0, skip next instruction	1 + (skip)	None	00 0110 0MMM MMMM
BTMSC M, b5	If bit5 of (M) = 0, skip next instruction	1 + (skip)	None	00 0110 1MMM MMMM

BTMSC M, b6	If bit6 of (M) = 0, skip next instruction	1 + (skip)	None	00 0111 0MMM MMMM
BTMSC M, b7	If bit7 of (M) = 0, skip next instruction	1 + (skip)	None	00 0111 1MMM MMMM
BTMSS M, b0	If bit0 of (M) = 1, skip next instruction	1 + (skip)	None	00 0000 0MMM MMMM
BTMSS M, b1	If bit1 of (M) = 1, skip next instruction	1 + (skip)	None	00 0000 1MMM MMMM
BTMSS M, b2	If bit2 of (M) = 1, skip next instruction	1 + (skip)	None	00 0001 0MMM MMMM
BTMSS M, b3	If bit3 of (M) = 1, skip next instruction	1 + (skip)	None	00 0001 1MMM MMMM
BTMSS M, b4	If bit4 of (M) = 1, skip next instruction	1 + (skip)	None	00 0010 0MMM MMMM
BTMSS M, b5	If bit5 of (M) = 1, skip next instruction	1 + (skip)	None	00 0010 1MMM MMMM
BTMSS M, b6	If bit6 of (M) = 1, skip next instruction	1 + (skip)	None	00 0011 0MMM MMMM
BTMSS M, b7	If bit7 of (M) = 1, skip next instruction	1 + (skip)	None	00 0011 1MMM MMMM
CALL I	Call subroutine	2	None	11 0110 iiii iiii
CLRA	Clear accumulator	1	Z	10 0001 0000 0000
CLRM M	Clear memory M	1	Z	10 0001 1MMM MMMM
CLRWDT	Clear watch-dog register	1	TO, PO	10 0000 0000 0001
COMM M, m	~(M) → (M)	1	Z	10 0010 1MMM MMMM
COMM M, a	~(M) → (acc)	1	Z	10 0010 0MMM MMMM
DECM M, m	Decrement M to M	1	Z	10 0110 1MMM MMMM
DECM M, a	(M) - 1 → (acc)	1	Z	10 0110 0MMM MMMM
DECMSZ M, m	(M) - 1 → (M), skip if (M) = 0	1 + (skip)	None	10 0111 1MMM MMMM
DECMSZ M, a	(M) - 1 → (acc), skip if (M) = 0	1 + (skip)	None	10 0111 0MMM MMMM
GOTO I	Goto branch	2	None	11 101i iiii iiii
INCM M, m	(M) + 1 → (M)	1	Z	10 1000 1MMM MMMM
INCM M, a	(M) + 1 → (acc)	1	Z	10 1000 0MMM MMMM
INCMSZ M, m	(M) + 1 → (M), skip if (M) = 0	1 + (skip)	None	10 1001 1MMM MMMM
INCMSZ M, a	(M) + 1 → (acc), skip if (M) = 0	1 + (skip)	None	10 1001 0MMM MMMM
IODIR M	Set i/o direction	1	None	10 0000 0000 0MMM
IORAM M, m	(M) ior (acc) → (M)	1	Z	10 1111 1MMM MMMM
IORAM M, a	(M) ior (acc) → (acc)	1	Z	10 1111 0MMM MMMM

IORLA I	Literal ior (acc) → (acc)	1	Z	11 0011 iiii iiii
LCALL I	Call subroutine. However, LCALL can addressing 2K address	2	None	01 0iii iiii iiii
LGOTO I	Go branch to any address	2	None	01 1iii iiii iiii
MOVAM m	Move data form acc to memory	1	None	10 0000 1MMM MMMM
MOVLA I	Move literal to accumulator	1	None	11 0001 iiii iiii
MOVM M, m	(M) → (M)	1	Z	10 0011 1MMM MMMM
MOVM M, a	(M) → (acc)	1	Z	10 0011 0MMM MMMM
NOP	No operation	1	None	10 0000 0000 0000
RET	Return	2	None	11 1111 0111 1111
RETI	Return and enable INTM	2	None	11 1111 1111 1111
RETLA I	Return and move literal to accumulator	2	None	11 1100 iiii iiii
RLM M, m	Rotate left from m to itself	1	C	10 1100 1MMM MMMM
RLM M, a	Rotate left from m to acc	1	C	10 1100 0MMM MMMM
RRM M, m	Rotate right from m to itself	1	C	10 1110 1MMM MMMM
RRM M, a	Rotate right from m to acc	1	C	10 1110 0MMM MMMM
SELECT	Set select register	1	None	10 0000 0000 0010
SLEEP	Enter sleep (saving) mode	1	TO, PO	10 0000 0000 0011
SUBAM M, m	(M)–(acc) → (M)	1	C, DC, Z	10 1010 1MMM MMMM
SUBAM M, a	(M) –(acc) → (acc)	1	C, DC, Z	10 1010 0MMM MMMM
SWAPM M, m	Swap data from m to itself	1	None	10 1101 1MMM MMMM
SWAPM M, a	Swap data from m to acc	1	None	10 1101 0MMM MMMM
XORAM M, m	(M) xor (acc) → (M)	1	Z	10 1011 1MMM MMMM
XORAM M, a	(M) xor (acc) → (acc)	1	Z	10 1011 0MMM MMMM
XORLA I	Literal xor (acc) → (acc)	1	Z	11 1000 iiii iiii

8. Electrical Characteristics

8.1 Absolute Maximum Ratings

Supply Voltage …. Vss-0.3V to Vss+5.5V   Storge Temperature ……. –50℃ to 125℃
Input Voltage …… Vss-0.3V to VDD+0.3V   Operating Temperature… 0℃ to 70℃

8.2 DC Characteristics

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		VDD	Conditions				
VDD	Operating Voltage	---		2.2		5.5	V
V_{DVT}	Detect Voltage	5V	Low Voltage Detector (Idd = 3uA) Config bit6.bit5=00		4		V
		3V	Low Voltage Detector (Idd = 1.5uA) Config bit6.bit5=10		2		V
V_{IH}	Input HighVoltage	5V	I/O Port	2		VDD	V
V_{IL}	Input Low Voltage	5V	I/O Port			0.8	V
I_{DD1}	Standby Current	5V	LVD disable, WDT disable		1		uA
			LVD disable, WDT enable		10		
I_{IL}	Input Leakage Current	5V	Vin=VDD, VSS		1		uA
I_{OH}	I/O Port Driving Current	5V	Voh=4.5V		9		mA
			Voh=4V		17		
			Voh=3.5V		23		
I_{OL}	I/O Port Sink Current	5V	Vol=0.5V		20		mA
			Vol=01V		35		
			Vol=1.5V		50		

8.3 AC Characteristics

Symbol	Parameter	Test Conditions		Min	Typ	Max	Unit
		VDD	Conditions				
f_{sys1}	System Clock	5V	LP Crystal mode	32		200	Khz
		3V		32		200	
f_{sys2}	System Clock	5V	NT Crystal mode	0.2		10	Mhz
		3V		0.2		10	
f_{sys3}	System Clock	5V	HS Crystal mode	10		20	Mhz
		3V					
f_{sys4}	System Clock	5V	RC mode			6	Mhz
		3V				6	
T_{wdt}	Watchdog Timer	5V			20		mS
		3V			30		
T_{rht}	Reset Hold Time	5V			20		mS
		3V			30		