ePS6000

RISC II Series Microcontroller

Product Specification

DOC. VERSION 1.7

ELAN MICROELECTRONICS CORP.

August 2008

Trademark Acknowledgments:

IBM is a registered trademark and PS/2 is a trademark of IBM.
Windows is a trademark of Microsoft Corporation.
ELAN and ELAN logo are trademarks of ELAN Microelectronics Corporation.

The contents of this specification are subject to change without further notice. ELAN Microelectronics assumes no responsibility concerning the accuracy, adequacy, or completeness of this specification. ELAN Microelectronics makes no commitment to update, or to keep current the information and material contained in this specification. Such information and material may change to conform to each confirmed order.

In no event shall ELAN Microelectronics be made responsible for any claims attributed to errors, omissions, or other inaccuracies in the information or material contained in this specification. ELAN Microelectronics shall not be liable for direct, indirect, special incidental, or consequential damages arising from the use of such information or material.

The software (if any) described in this specification is furnished under a license or nondisclosure agreement, and may be used or copied only in accordance with the terms of such agreement.

ELAN Microelectronics products are not intended for use in life support appliances, devices, or systems. Use of ELAN Microelectronics product in such applications is not supported and is prohibited.

NO PART OF THIS SPECIFICATION MAY BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT THE EXPRESSED WRITTEN PERMISSION OF ELAN MICROELECTRONICS.

ELAN MICROELECTRONICS CORPORATION

Headquarters:

No. 12, Innovation Road 1
Hsinchu Science Park
Hsinchu, TAIWAN 308
Tel: +886 3 563-9977
Fax: +886 3 563-9966
http://www.emc.com.tw

Hong Kong:

Elan (HK) Microelectronics Corporation, Ltd.
Flat A, 19F., World Tech Centre 95
How Ming Street, Kwun Tong
Kowloon, HONG KONG
Tel: +852 2723-3376
Fax: +852 2723-7780
elanhk@emc.com.hk

USA:

Elan Information Technology Group (U.S.A.)
P.O. Box 601
Cupertino, CA 95015
U.S.A.
Tel: +1 408 366-8225
Fax: +1 408 366-8225

Shenzhen:

Elan Microelectronics Shenzhen, Ltd.
3F, SSMEC Bldg., Gaoxin S. Ave. I
Shenzhen Hi-tech Industrial Park
(South Area), Shenzhen
CHINA 518057
Tel: +86 755 2601-0565
Fax: +86 755 2601-0500

Shanghai:

Elan Microelectronics Shanghai, Ltd.
#23, Zone 115, Lane 572, Bibo Rd.
Zhangjiang Hi-Tech Park
Shanghai, CHINA 201203
Tel: +86 21 5080-3866
Fax: +86 21 5080-4600

Contents

Specification Revision History

Doc. Version	Revision Description	Date
1.0	Modify DC electrical characteristic Modify DC electrical characteristic Add the power current of Fast mode (200kHz) Modify DC electrical characteristic Modify pin assignment	2004/11/30 2004/12/09 2004/12/10 2004/12/27 2004/12/31
1.1	Add LCD driving methods Modify the value for each bias voltage of LCD	2005/01/18 2005/02/03
1.2	Add Pad Diagram and modify pin assignment	2005/03/11
1.3	Modify DC electrical characteristic (VDD=3V), Key I/O code example Add the 1/5 duty to LCD driver	2005/05/30 2005/08/21
1.4	Add Crystal stable timing and Slow to Fast mode Timing diagrams	2005/09/26
1.5	Add Fast to Slow mode Code Example to MCU Operation Mode (Section 7.3.1)	2005/11/01
1.6	Modify DC electrical characteristic (VDD=1.5V): Supply current Modify Application circuit	2006/07/03
1.7	Modify the pop interrupt register of the code example (Section 7.4.1) Modified the max. supply voltage on section 9 Added a Note on Section 6 Code Option	2008/04/07 2008.08.26

1 General Description

IMPORTANT NOTES !!

- *Do not use Register BSR (05h) Bit7 ~ Bit4.*
- *Do not use Register BSR1 (07h) Bit7 ~ Bit4.*
- *Do not use LCD RAM 3Ch ~ 3Fh.*
- *Do not use Registers JDNZ at FSR1 (04h) special register.*

ePS6000 is an 8-bit RISC MCU embedded with an 11*60 LCD driver along with two 8-bit timers, one 16-bit general timer, and a watch dog timer. Furthermore, ePS6000 is equipped with 2K bytes RAM and 16K words program ROM. It is highly ideal for advance scientific calculator application, particularly those that need a high performance at low cost solution.

The MCU core is a one of ELAN's second generation RISC based ICs, known as RISC II (RII) series. The core was specifically designed for low power and portable device applications. The ePS6000 also supports FAST, SLOW, and IDLE modes, as well as SLEEP mode to enhance its low power consumption features.

1.1 Applications

- Scientific calculating machine

2 Features

2.1 MCU Features

- 8 bit RISC MCU
- Operating voltage and speed: 1.2V~3.6V
- Clock Source: Dual clock system
  - Low-frequency: 32KHz Internal RC oscillator / External RC oscillator / Crystal oscillator.
  - High-frequency 200KHz / 300KHz / 500KHz External RC oscillator.
- One Instruction cycle time = 2 * System clock time
- Program ROM addressing: Maximum 16K words
- 128 bytes un-banked RAM including special registers and common registers
- 16 *128 bytes banked RAM
- RAM stack can achieve maximum of 32 stack levels

- TABLE LOOK UP function is fast and highly efficient when combined with REPEAT instruction
- Register to Register move instruction
- Compare and Branch in one instruction (2 cycles)
- Single Repeat function (max. 256 repeat times)
- Decimal ADD & SUB instruction
- Full range CALL and JUMP ability (2 cycles)

2.2 Peripheral

- 24 general I/O pins (PORTA, PORTB, PORTC)
- 11/10/5 COM * 60 SEG LCD driver (embedded)
- One (Timer 0)16-bit timer
- One (Timer1) 8-bit timer with wake-up function
- One (Timer2) 8-bit timer
- One 8-bit watch dog timer
- Key I/O function with maximum 64 keys

2.3 Internal Specification

- Watchdog Timer with its own on-chip RC oscillator
- MCU operating modes: SLEEP MODE, IDLE MODE, SLOW MODE, and FAST MODE
- Support RC oscillation and crystal oscillation for clock system
- MCU Wake-up function consisting of input wake-up and Timer1 wake-up
- MCU interrupt function consisting of Input port interrupt and Timer interrupt (Timer0 ~ 2).
- MCU reset function includes power on reset, RSTB pin reset, and Watchdog timer reset.

3 Block Diagram

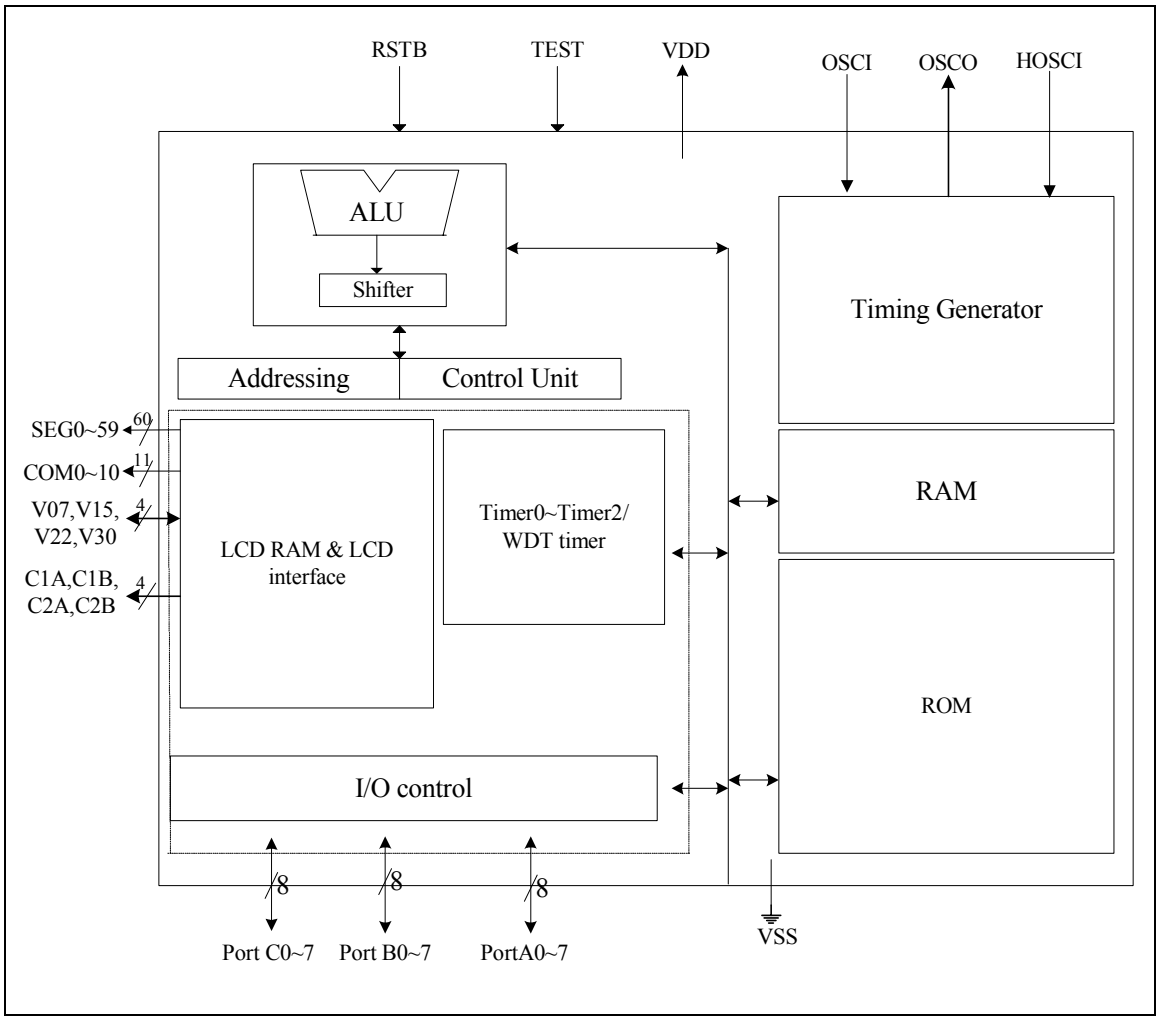


Figure 3-1 ePS6000 Block Diagram

4 Package Pinout and Pin Assignment

4.1 ePS6000 Pinout

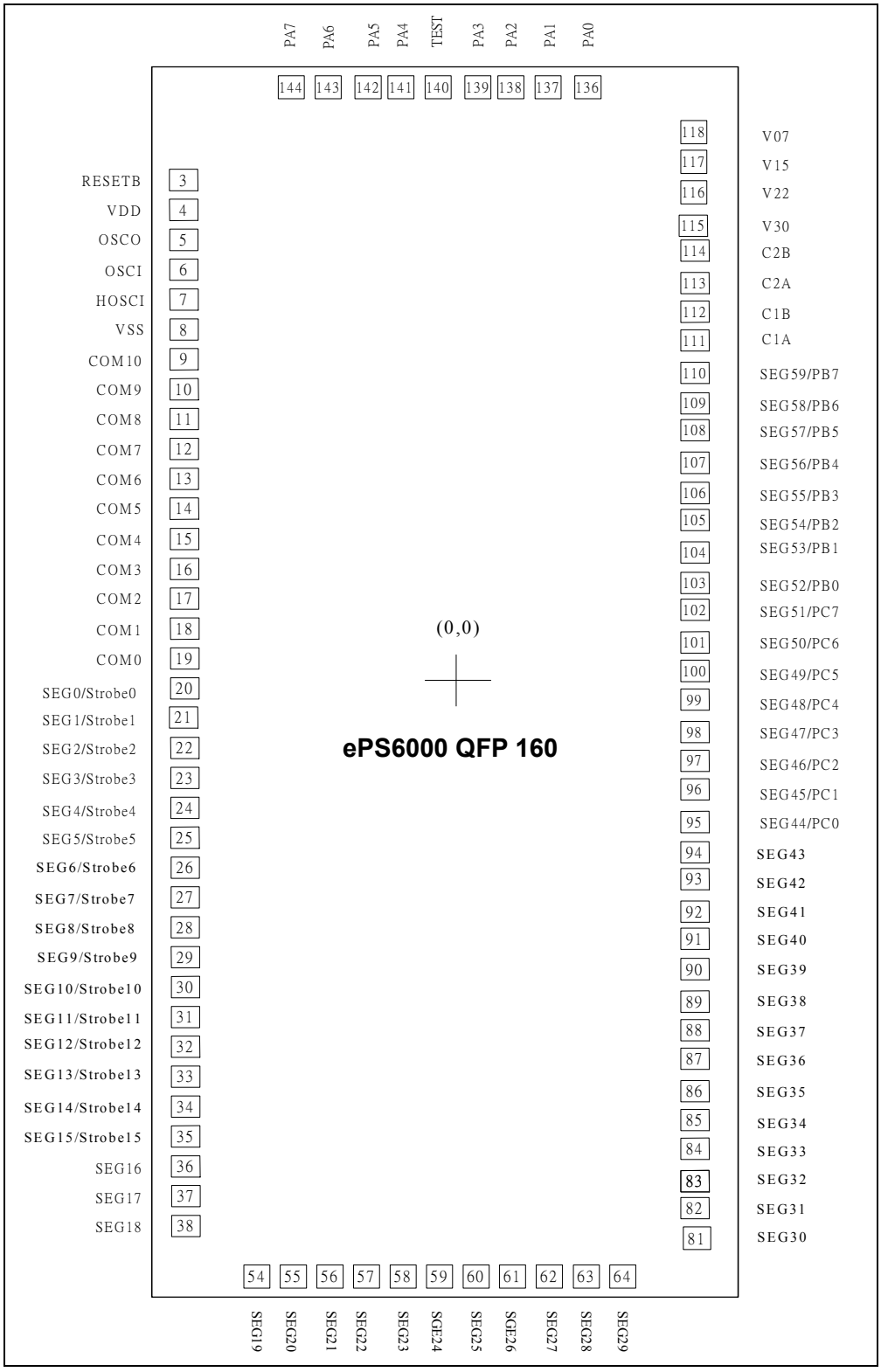
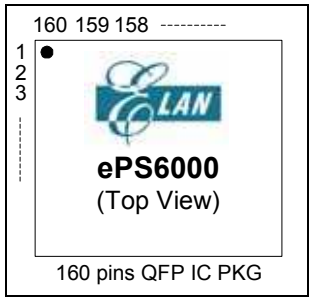


Figure 4-1 ePS6000 QFP Pinout

4.2 Pin Assignment



No.	Pin Name	No.	Pin Name	No.	Pin Name	No.	Pin Name
1	NC	41	NC	81	SEG30	121	NC
2	NC	42	NC	82	SEG31	122	NC
3	RESETB	43	NC	83	SEG32	123	NC
4	VDD	44	NC	84	SEG33	124	NC
5	OSCO	45	NC	85	SEG34	125	NC
6	OSCI	46	NC	86	SEG35	126	NC
7	HOSCI	47	NC	87	SEG36	127	NC
8	VSS	48	NC	88	SEG37	128	NC
9	COM10	49	NC	89	SEG38	129	NC
10	COM9	50	NC	90	SEG39	130	NC
11	COM8	51	NC	91	SEG40	131	NC
12	COM7	52	NC	92	SEG41	132	NC
13	COM6	53	NC	93	SEG42	133	NC
14	COM5	54	SEG19	94	SEG43	134	NC
15	COM4	55	SEG20	95	SEG44/PortC.0	135	NC
16	COM3	56	SEG21	96	SEG45/PortC.1	136	PortA.0
17	COM2	57	SEG22	97	SEG46/PortC.2	137	PortA.1
18	COM1	58	SEG23	98	SEG47/PortC.3	138	PortA.2
19	COM0	59	SEG24	99	SEG48/PortC.4	139	PortA.3
20	SEG0/Strobe0	60	SEG25	100	SEG49/PortC.5	140	TEST
21	SEG1/Strobe1	61	SEG26	101	SEG50/PortC.6	141	PortA.4
22	SEG2/Strobe2	62	SEG27	102	SEG51/PortC.7	142	PortA.5
23	SEG3/Strobe3	63	SEG28	103	SEG52/PortB.0	143	PortA.6
24	SEG4/Strobe4	64	SEG29	104	SEG53/PortB.1	144	PortA.7
25	SEG5/Strobe5	65	NC	105	SEG54/PortB.2	145	NC
26	SEG6/Strobe6	66	NC	106	SEG55/PortB.3	146	NC
27	SEG7/Strobe7	67	NC	107	SEG56/PortB.4	147	NC
28	SEG8/Strobe8	68	NC	108	SEG57/PortB.5	148	NC
29	SEG9/Strobe9	69	NC	109	SEG58/PortB.6	149	NC
30	SEG10/Strobe10	70	NC	110	SEG59/PortB.7	150	NC
31	SEG11/Strobe11	71	NC	111	C1A	151	NC
32	SEG12/Strobe12	72	NC	112	C1B	152	NC
33	SEG13/Strobe13	73	NC	113	C2A	153	NC
34	SEG14/Strobe14	74	NC	114	C2B	154	NC
35	SEG15/Strobe15	75	NC	115	V30	155	NC
36	SEG16	76	NC	116	V22	156	NC
37	SEG17	77	NC	117	V15	157	NC
38	SEG18	78	NC	118	V07	158	NC
39	NC	79	NC	119	NC	159	NC
40	NC	80	NC	120	NC	160	NC

5 Pin Description

5.1 MCU System Pins (7 Pins)

Name	I/O/P Type	Description	Note
VDD	P	Digital and Analog positive power supply, the range is from 1.2V~1.8V/ 2.4V~3.6V.	
VSS	P	Digital and Analog negative power supply.	
RSTB	I	System reset pin. Low active	Int. pull-up
TEST	I	Test mode select pin (High active). For chip internal test only, Normal connect to VSS.	Int. Pull Down
OSCI	I	External RC oscillator /Crystal oscillator connecting pin	Ext. R to VDD
OSCO	O	Crystal oscillator connecting pin	
HOSCI	I	Hi-Speed RC oscillator connecting pin.	Ext. R to VDD

5.2 Embedded LCD Pins (79 Pins)

Name	I/O/P Type	Description	Note
COM0~COM10	O	LCD common signal output pin	
SEG0~SEG15	O	LCD segment signal output pin shared with key strobe 0~15	
SEG16~ SEG43	O	LCD segment signal output pin	
SEG44~SEG51/ PortC.0~7	I/O	LCD segment signal output pin or I/O pin; define by code option	
SEG52~SEG59/ PortB.0~7	I/O	LCD segment signal output pin or I/O pin; define by code option	
C1A,C1B		LCD voltage charge-pump pin. Connect 0.1 uF between C1A and C1B.	
C2A,C2B		LCD voltage charge-pump pin. Connect 0.1 uF between C2A and C2B.	
V30,V22,V15,V07	O	LCD bias Pin. Connect 0.1 uF to Vss	

5.3 I/O Port (8 Pins)

PORT	Bit	Function	I/O Type	Power Source	Description	Note
Port A	Bit3~0 (for key scan)	General Input	I	VDD	Key input	Int. Pull up (R1: small resistor, R2: Large resistor) controllable
		Interrupt and wake-up	I	VDD	Input port interrupt and wake-up pin	
		General Output	O	VDD		
	Bit7~4	General Input	I	VDD		Int. Pull up (R2: Large resistor) controllable
		Interrupt and wake-up	I	VDD	Input port interrupt and wake-up pin	
		General Output	O	VDD		

6 Code Option

Located at Address 0x000C~0x000F of Program ROM

- Initial mode after reset:
  - Select “Slow” mode or “Fast” mode

NOTE
Suggest that user setting “Slow mode” for Initial mode after reset.

- Low Frequency Oscillator:
  - Select “External RC” oscillator or “Crystal” oscillator or “Internal RC” oscillator
- Maximum duty ratio option:
  - Select “1/10” duty or “1/11” duty or “1/5” duty
- Port B.0 control bit (SEG52):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.1 control bit (SEG53):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.2 control bit (SEG54):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.3 control bit (SEG55):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.4 control bit (SEG56):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.5 control bit (SEG57):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.6 control bit (SEG58):
  - Select “LCD segment signal output” or “general I/O function”
- Port B.7 control bit (SEG59):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.0 control bit (SEG44):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.1 control bit (SEG45):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.2 control bit (SEG46):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.3 control bit (SEG47):
  - Select “LCD segment signal output” or “general I/O function”

- Port C.4 control bit (SEG48):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.5 control bit (SEG49):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.6 control bit (SEG50):
  - Select “LCD segment signal output” or “general I/O function”
- Port C.7 control bit (SEG51):
  - Select “LCD segment signal output” or “general I/O function”

7 Function Description

7.1 Reset Function

RESET can be generated by one of the following:

- Power on voltage detector reset and power on reset.
- WDT time out.
- RSTB pin pull low.

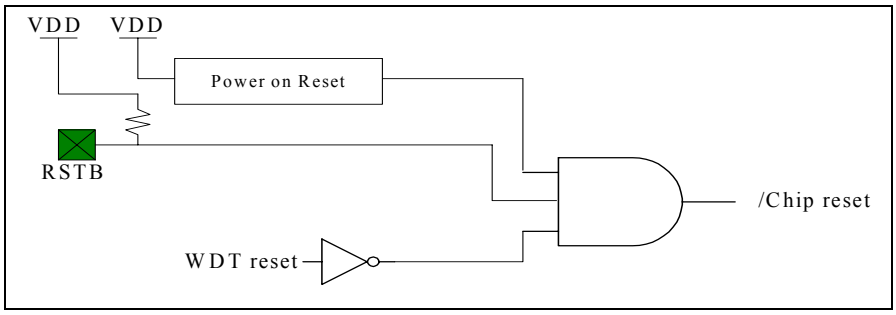


Figure 7-1 On-Chip RESET Schematic

7.1.1 Power on Reset

The power on reset circuit holds the device under reset condition until VDD is above Vpor (power on reset voltage). Whenever the voltage supply decreases to below Vpor, RESET will occur.

7.1.2 RSTB Pin

In normal condition, RSTB pin is pulled up to VDD. Whenever the RSTB is at low condition, RESET will occur.

7.1.3 WDT Time Out

When Watch Dog Timer is enabled, the WDT time-out will cause the chip to reset. To prevent reset from occurring, you should clear the WDT value with the “WDTC” instruction before WDT time-out. WDT time-out can also be used to flag software malfunction.

7.1.4 32768 Crystal Stable Time

■ **Power On Reset Timing:**

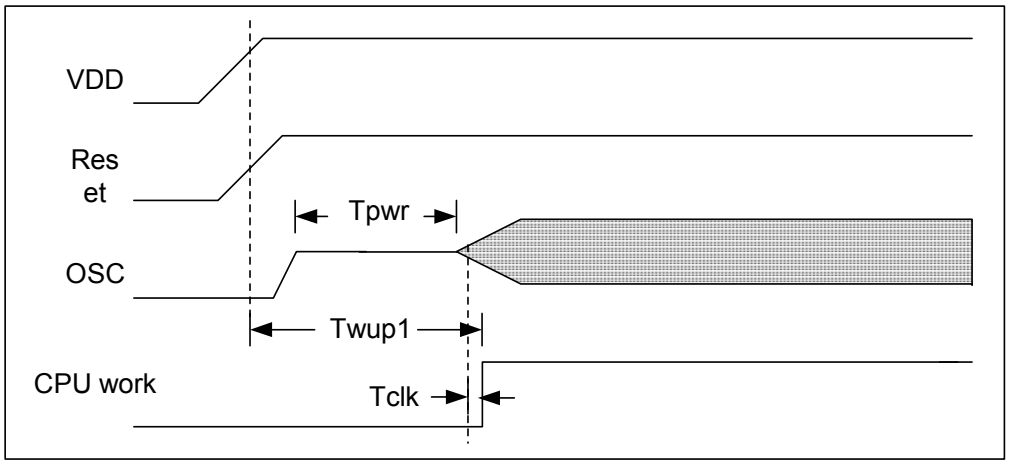


Figure 7-2a Power on Reset Timing Diagram

■ **Sleep Mode Wake-Up Timing:**

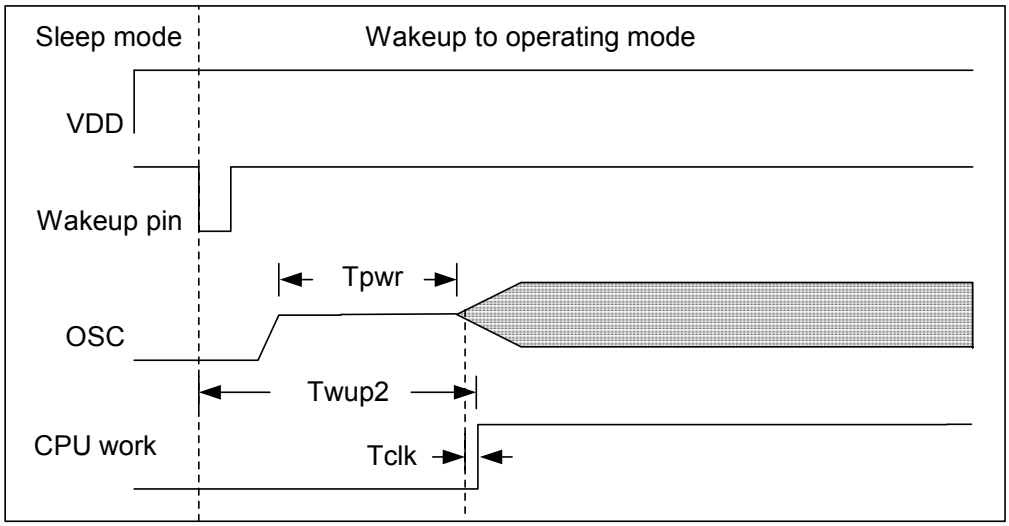


Figure 7-2b Sleep Mode Wake-Up Timing Diagram

Condition: Vdd = 1.5V, Cosc = 20pF & Ta = 25°C

Symbol	Characteristics	Min.	Typ.	Max.	Unit
Tpwr	Oscillator start up time	280	480	780	ms
Twup1	CPU warm up time (Power ON reset)	300	500	800	ms
Twup2	CPU warm up time (Sleep mode wakeup)	285	485	785	ms
Tclk	Detect slow clock time	0.9	1.0	1.1	ms

7.1.5 STATUS (R0Fh)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
/TO	/PD	SGE	SLE	OV	Z	DC	C

Bit0 (C): Carry flag or inverse of Borrow flag (B)
Under SUB operation, borrow flag is indicated by the inverse of carry bit. (B = /C).

Bit1 (DC): Auxiliary carry flag.

Bit2 (Z): Zero flag

Bit3 (OV): Overflow flag. Use in signed operation when Bit6 is carried into or borrows from signed bit (Bit7).

Bit4 (SLE): Computation result is less than or equal to zero (negative value) after signed arithmetic. Affected by HEX arithmetic instruction only.

Bit5 (SGE): Computation result is greater than or equal to zero (positive value) after signed arithmetic. Affected by HEX arithmetic instruction only.

NOTE

1. *When OV=1 after signed arithmetic, you can check SGE bit and SLE bit to verify whether overflow (carry into sign bit) or underflow (borrow from sign bit) occurred.*
   If OV=1 and SGE=1 ➔ overflow occurred.
   If OV=1 and SLE=1 ➔ underflow occurred.
2. *When overflow took place, you should clear the MSB of Accumulator to obtain the correct value.*
   When underflow took place, you should set the MSB of accumulator to obtain the correct value.

Example 1: ADD positive value with a positive value, and the ACC signed bit will be affected.

```
MOV     ACC, #60h       ; Signed number +60h
ADD     ACC, #70h       ; +60h ADD WITH +70h
```

Unsigned bit results after execution of the instruction**:**

ACC = 0D0h SGE=1, means the result is greater than or equal to '**0**' (positive value)
OV=1, means overflow occurred and the result is carried into signed bit (Bit 7)

Signed bit results after execution of the instruction**:**

ACC = 50h (signed bit is cleared)

The actual result = +80h (OV=1) + 50h = +0D0h

Example 2: SUB positive value from negative value, and ACC signed bit will be affected.

```
MOV     ACC, #50h    ; Signed number +50h.
SUB     ACC, #90h    ; +50h SUB from –70h. (Signed number of 90h)
```

Unsigned bit results after execution of the instruction**:**

ACC = 40h SLE=1, means the result is less than or equal to '**0**' (negative value)

OV=1, Underflow occurred and the result is borrowed from signed bit (Bit 7)

Signed bit results after execution of the instruction**:**
ACC = 0C0h (the signed bit is set)

The actual result = –80h (OV=1) + 0C0h (signed number of 0C0h) = 40h

Bit6 (/PD): Reset to "**0**" when /PD enters SLEEP mode. Set to "**1**" by "WDTC" instruction, power-on reset, or by Reset pin low condition.

Bit7 (/TO): Reset to "**0**" at WDT time out reset. Set to "**1**" by "WDTC" instruction; power-on reset, Reset pin low condition, or when MCU enters into SLEEP MODE.

When reset occurs, special function register is reset to initial value except for the **/TO** and **/PD** bits of STATUS register.

Bit 7 (/TO)	Bit 6 (/PD)	Event
0	0	WDT time out reset from SLEEP mode
0	1	WDT time out reset (not from SLEEP mode)
1	0	Reserved.
1	1	Power up or RSTB pin low condition

7.1.6 Initialization after RESET Occurs

- The oscillator is running, or will be started.
- The Watchdog timer is cleared.
- During power on reset or RSTB pin low condition, the **/TO** bit and **/PD** bit of RF (STATUS) are set to "**1**." At WDT time out reset, the /**TO** bit is cleared.
- The program counter (PCM**:** PCL) is cleared to all "**0**."
- The other register initial values are as indicated in the following sections**:**

■ Special Registers

Addr.	Name	Initial Value	Addr.	Name	Initial Value
00h	INDF0	---- ----[1]	10h	PORTA	xxxx xxxx
01h	FSR0	0000 0000	11h	PORTB	xxxx xxxx
02h	BSR	0000 0000	12h	PORTC	xxxx xxxx
03h	INDF1	---- ----[1]	13h	General Ram	uuuu uuuu
04h	FSR1	1000 0000	14h	General Ram	uuuu uuuu
05h	BSR1	0000 0000	15h	General Ram	uuuu uuuu
06h	STKPTR	0000 0000	16h	General Ram	uuuu uuuu
07h	PCL	0000 0000	17h	General Ram	uuuu uuuu
08h	PCM	0000 0000	18h	General Ram	uuuu uuuu
09h	LCDARL	0000 0000	19h	General Ram	uuuu uuuu
0Ah	ACC	xxxx xxxx	1Ah	General Ram	uuuu uuuu
0Bh	TABPTRL	0000 0000	1Bh	General Ram	uuuu uuuu
0Ch	TABPTRM	0000 0000	1Ch	General Ram	uuuu uuuu
0Dh	TABPTRH	uuuu uuuu	1Dh	General Ram	uuuu uuuu
0Eh	LCDDATA	---- ----[1]	1Eh	General Ram	uuuu uuuu
0Fh	STATUS	cuxx xxxx [2]	1Fh	General Ram	uuuu uuuu

■ Control Register

Addr.	Name	Initial Value	Addr.	Name	Initial Value
20h	STBCON	0000 0000	2Bh	PAINTEN	0000 0000
21h	INTCON	---- -000	2Ch	PAINTSTA	0000 0000
22h	INTSTA	---- -000	2Dh	DCRA	1111 1111
23h	TR01CON	0000 0000	2Eh	PBCON	0000 0000
24h	TRL0L	uuuu uuuu	2Fh	DCRB	1111 1111
25h	TRL0H	uuuu uuuu	30h	PCCON	0000 0000
26h	TRL1	uuuu uuuu	31h	DCRC	1111 1111
27h	TR2WCON	0–00 0000	32h	LCDCON	–00– 00–0
28h	TRL2	uuuu uuuu	33h	POST_ID	–111 –000
29h	PACON	0000 0000	34h	CPUCON	---- –00c [3]
2Ah	PAWAKE	0000 0000			

Legend: x: unknown **–:** unimplemented read as “0”
u: unchanged, **c:** value depends on actual condition

[1] *Not a physical register*

[2] *If it is a power-on reset or the RSTB pin is at low condition, the **/TO** bit and **/PD** bit of RF (STATUS) are set to “**1**.” If it is a WDT time out reset, the **/TO** bit is cleared and /PD bit remains unchanged.*

[3] *Bit0 (MS0) of RE (CPUCON) is reloaded from “**INIM**” bit of code option when MCU resets.*

7.2 Oscillator System

The oscillator system is used to generate the device clock. The oscillator system may use an Internal RC, external RC, or a crystal oscillator for SLOW mode and use an external RC oscillator for FAST mode as illustrated in the diagram below.

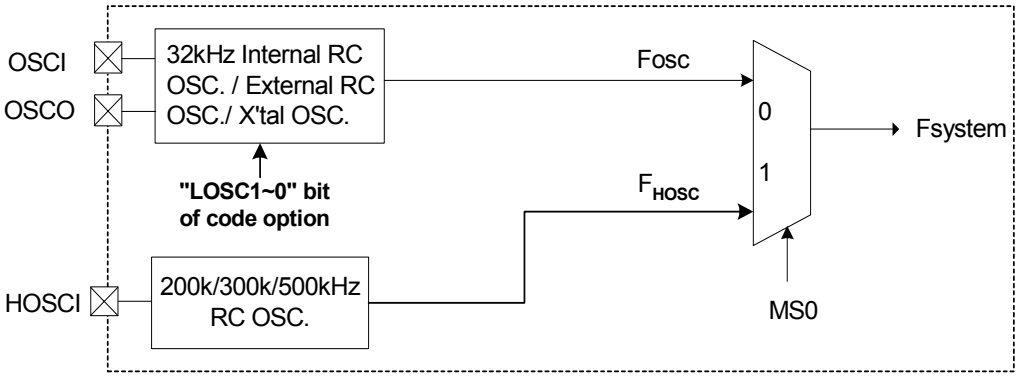


Figure 7-3 Oscillator System Function Block Diagram

The **MS0** bit (mode select bit) of **CPUCON** register (R34h) is used to set the SLOW or FAST mode (see Section 7.3.1).

"0": SLOW mode (MCU system Clock is from FOSC).
"1": FAST mode (MCU system Clock is from FHOSC).

7.2.1 32.8kHz RC or 32768Hz Crystal Oscillator

- 32.8kHz RC Internal oscillator**:**
  Select "RC oscillator for FOSC" in the code option and allow OSCI and OSCO pins to stay floating.
- 32.8kHz RC external oscillator**:**
  Select "RC oscillator for FOSC" in the code option and connect a 2.2MΩ resistor between OSCI and Vdd pin while OSCO pin stays floating.
- 32768kHz Crystal oscillator:
  Select "Crystal oscillator for FOSC" in the code option and connect a crystal between OSCI and OSCO pins. The OSCI and OSCO pins are also connected to ground through a 20pF capacitor respectively.

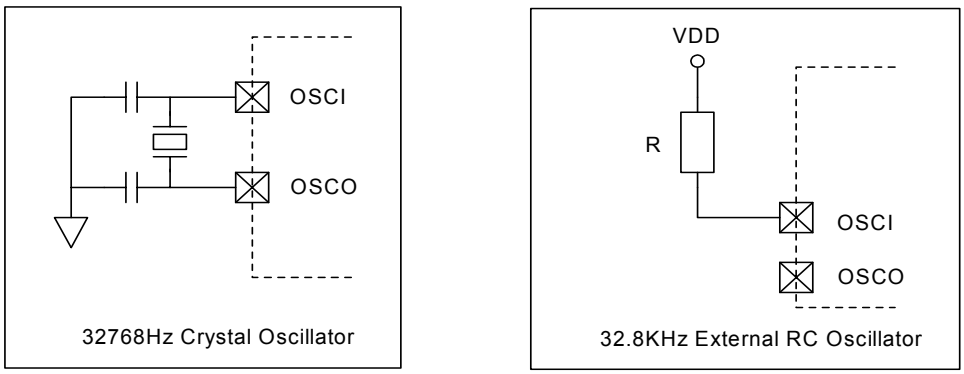


Figure 7-4 Slow Mode Crystal and External RC Oscillators Circuit Diagram

7.2.2 200kHz/300kHz/500kHz RC External Oscillator

A resistor should be connected between HOSCI and Vdd pin.

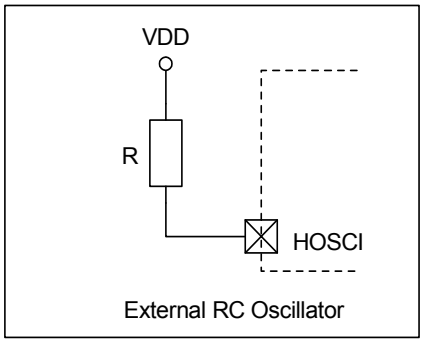


Figure 7-5 Fast Mode RC Oscillators Circuit Diagram

7.3 MCU Operation Mode:

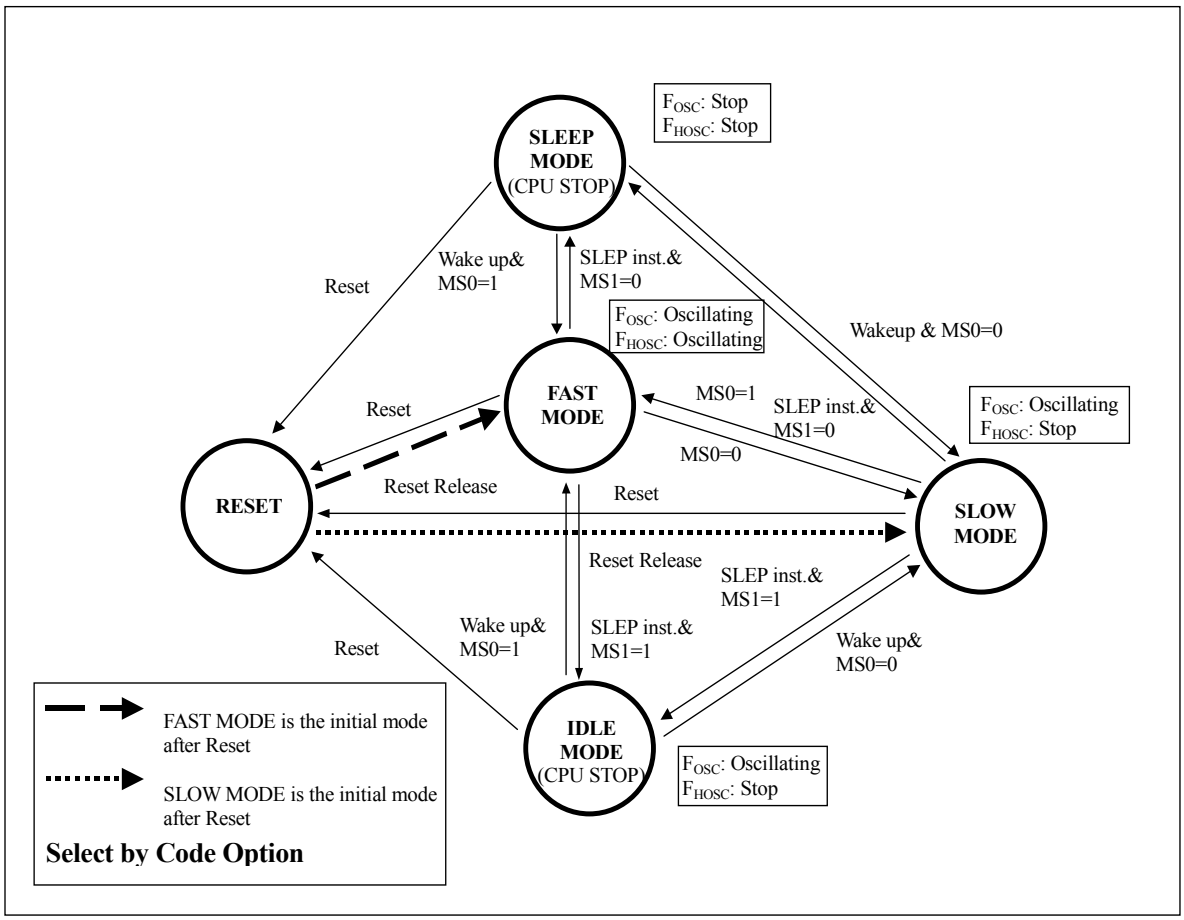


Figure 7-6 ePS6000 MCU Operation Block Diagram

The following table shows the supported device functions for each MCU Mode.

Device \ Mode	SLEEP	IDLE	SLOW	FAST
Osc.(32768Hz)	X	O	O	O
Fsystem	X	X	From Osc.	From Hosc.
Timer0~2	X	X	O	O
INT	X*	X*	O	O
I/O wake-up	O	O	X	X
Timer1 wake-up	X	O	X	X

Legend: O = Function is available if enabled **X:** Function NOT supported

***** *Interrupt flag will be recorded but not executed until MCU wakes up.*

7.3.1 Slow, Fast, Sleep, and Idle Mode of Operations

■ **CPUCON (R34h): MCU Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	GLINT	**MS1**	**MS0**

Bit 0 (MS0) Select SLOW MODE or FAST MODE
"0": SLOW MODE.
"1": FAST MODE

Bit 1 (MS1): Select SLEEP MODE or IDLE MODE after executing "SLEP" instruction.
"0": SLEEP MODE
"1": IDLE MODE

■ **SLOW MODE:**

When MS0 bit of CPUCON register is set to "**0**," the MCU will enter into SLOW MODE and the corresponding system clock is at 32kHz. The SLOW mode feature allows performance of all system operations at reduced power consumption.

NOTE
The instruction "NOP" should be added to follow "BC CPUCON,MS0" instruction when MCU is made to enter into SLOW MODE from FAST MODE. See the code example at the end of this section.

■ **FAST MODE:**

When MS0 bit of CPUCON register is set to "**1**," the MCU will enter into FAST MODE. After setting the MS0 bit, it needs to count 32 clocks from HOSC, then the system clock switches from slow to high frequency. This mode allows fast speed performance of all the system operations, but under highest power consumption.

■ IDLE MODE:

When MS1 bit of CPUCON register is set to "**1**." and the "SLEP" instruction is executed, the MCU will enter into IDLE MODE. The IDLE MODE suspends all system operations except for the 32kHz oscillator. It retains the internal status under low power consumption without stopping the clock function.

The IDLE MODE is awaken by the Timer 1 wake-up or by I/O pins wake-up (if enabled) and returns to the either SLOW MODE (MS0=0) or FAST MODE (MS0=1)

NOTE

All registers remain unchanged during SLEEP MODE.

■ SLEEP MODE:

When MS1 bit of CPUCON register is set to "**0**," and the "SLEP" instruction is executed, the MCU will enter into SLEEP MODE. The SLEEP MODE suspends all system operation and put on hold the internal status immediately before the suspension of operation. SLEEP MODE operates under very low power consumption and is awaken by I/O pins wake-up.

NOTE

- *The /PD bit of STATUS Register (RFh) is cleared when MCU enters SLEEP MODE.*
- *This /PD bit is set to "1" by "WDTC" instruction, power on reset, or by RSTB pin low condition.*
- *All registers remain unchanged during SLEEP MODE.*

■ SLOW MODE to FAST MODE Timing:

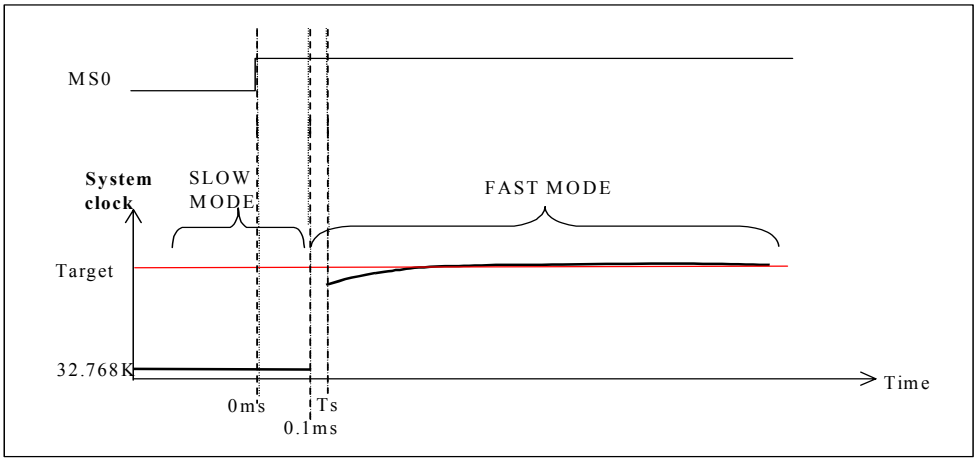


Figure 7-7 SLOW MODE to FAST MODE Timing Diagram

NOTE: *1. SLOW MODE switches to FAST MODE at Time=0ms.*

2. System clock will switch to FAST MODE after delay of 0.1ms by oscillator and enters into FAST MODE (i.e., system clock will be at 200, 300, or 500 kHz).

3. High frequency RC will stabilize at Time=Ts (around 15μs~30μs).

■ **Code Example:**

```
;Entry FAST mode
     BS      CPUCON,MS0

;Entry SLOW mode
     BC      CPUCON,MS0

;FAST mode Entry SLOW mode
     BS      CPUCON,MS0
     :
     :
     BC      CPUCON,MS0
     NOP
```

```
;Entry IDLE mode
     BS      CPUCON,MS1
     SLEP
     NOP

;Entry SLEEP mode
     BC      CPUCON, MS1
     SLEP
     NOP
```

7.3.2 Wake-up Operation

Oscillator is off during SLEEP MODE. The MCU is awaken by input port (Port A), then returns to FAST MODE or SLOW MODE (as determined by MS0 bit of CPUCON register described in previous section).

When in IDLE MODE, the 32khz oscillator keeps on running. The MCU is awaken by input port (Port A) or Timer1, then returns to FAST MODE or SLOW MODE (as determined by MS0 bit of CPUCON register described in the previous section).

■ **PAWAKE (R2Ah): Port A Wake-Up Function Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WKEN7	WKEN6	WKEN5	WKEN4	WKEN3	WKEN2	WKEN1	WKEN0

Bit 7 (WKEN7) ~ Bit 0 (WKEN0): Wake-up function control bit of PortA.7 ~ PortA.0

"0": Disable PortA.7 ~ PortA.0 wake-up function

"1": Enable PortA.7 ~ PortA.0 wake-up function

■ **T1WKEN Bit of (R23h): Timer 0 & Timer 1 Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1WKEN	T1EN	T1PSR1	T1PSR0	T0EN	T0CS	T0PSR1	T0PSR0

Bit 7 (T1WKEN): Timer1 underflow wake-up function control bit under IDLE MODE

"0": Disable Timer1 wake-up function

"1": Enable Timer1 wake-up function.

7.4 Interrupts

When interrupt occurs, the GLINT bit of CPUCON register is reset to "**0**," It disables all interrupts, including LEVELs 1 ~ 5. Setting this bit to "**1**" will enable all un-mask interrupts.

7.4.1 Global Interrupt

■ **GLINT Bit of CPUCON (R34h) MCU Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	**GLINT**	MS1	MS0

Bit 2 (GLINT): Global interrupt control bit

"0": disables all interrupts, including LEVEL 1 ~ LEVEL 5

"1": enables all un-mask interrupts

■ **Interrupt Vector**

Interrupt Level	Interrupt Source	Start Address	Remarks
	RESET	0x00000	
Level 1	PortA.7 ~ 0	0x00002	PAINT
Level 2	reserved	0x00004	Reserved
Level 3	reserved	0x00006	Reserved
Level 4	Timer0~2	0x00008	TMR0I, TMR1I, TMR2I
Level 5	reserved	0x0000A	Reserved

■ **Code Example:**

```
;  *****  Reset program
ResetSEG  CSEG      0X00
     LJMP   RESET          ; (0x00) Initialize
     LJMP   PAINT          ; (0x02) Port A Interrupt
     LJMP   RESERVED       ; (0x04) Reserved
     LJMP   RESERVED       ; (0x06) Reserved
     LJMP   TIMERINT       ; (0x08) Timer-0,1,2 Interrupt
     LJMP   RESERVED       ; (0x0A) Reserved
INT    CSEG    0x20
```

```
;  ---  Push interrupt register
PUSH:
     MOV    AccBuf,A
     MOVPR  StatusBuf,Status
     RET
```

```
;  ---  Pop interrupt register
POP:
     MOV    A,AccBuf
     MOVRP  Status,StatusBuf
     RETI
```

7.4.2 Input Port (PortA.7 ~ PortA.0) Interrupt

PortA.0 ~ PortA.7 are used as external interrupt/wake-up input. If PA7IE ~ PA0IE bits of PAINTEN register are set to "**1**," PortA.0 ~ PortA.7 are the external interrupt input port format.

■ **PAINTSTA (R2Ch): PortA.7 ~ PortA.0 Interrupt Status Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7I	PA6I	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I

Bit 7 (PA7I) ~ Bit 0 (PA0I): PortA.7 ~ PortA.0 Interrupt status

Set to "**1**" when pin falling edge is detected
Clear ("**0**") by software

■ **PAINTEN (R2Bh): PortA.7 ~ PortA.0 Interrupt Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7IE	PA6IE	PA5IE	PA4IE	PA3IE	PA2IE	PA1IE	PA0IE

Bit 7 (PA7IE) ~ Bit 0 (PA0IE): PortA.7 ~ PortA.0 Interrupt control bits

"0": Disable interrupt function
"1": Enable interrupt function

■ **Code Example:**

```
;  ===  Input PortA Interrupt
PAINT:
     S0CALL PUSH
     CLR PAINTSTA
        :
     SJMP   POP
     RETI
```

7.4.3 Timer0 Interrupt

Timer 0 is a 16-bit timer used for general time counting. When the counting value underflows, the Timer0 interrupt takes place and the TRL0H**:** TRL0L value is automatically reloaded into the timer.

■ **TMR0IE Bit of INTCON (R21h) Timer Interrupt Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	TMR2IE	TMR1IE	**TMR0IE**

Bit 0 (TMR0IE): Control bit of Timer0 interrupt

"0": Disable Timer0 interrupt function
"1": Enable Timer0 interrupt function

■ **TMR0I Bit of INTSTA (R22h) Timer Interrupt Status Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	TMR2I	TMR1I	**TMR0I**

Bit0 (TMR0I): Status bit of Timer0 interrupt

Set to "**1**" when Timer0 counter underflows
Clear ("**0**") by software

7.4.4 Timer1 Interrupt

Timer1 is an 8-bit timer used for time counting and wake-up functions. When the counting value of Timer1 underflows, interrupt occurs and the TRL1 value is reloaded to the timer.

■ **TMR1IE Bit of INTCON (R21h) Timer Interrupt Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	TMR2IE	**TMR1IE**	TMR0IE

Bit 1 (TMR1IE): Control bit of Timer1 interrupt

"0": Disable Timer1 interrupt function
"1": Enable Timer1 interrupt function

■ **TMR1I Bit of INTSTA (R22h) Timer Interrupt Status Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	TMR2I	**TMR1I**	TMR0I

Bit1 (TMR1I): Status bit of Timer1 interrupt

Set to "**1**" when Timer1 counter underflows
Clear ("**0**") by software

7.4.5 Timer2 Interrupt

Timer2 is an 8-bit timer for time counting. When the counting value of Timer2 underflows, the interrupt occurs and the TRL2 value is reloaded into the timer.

■ **TMR2IE Bit of INTCON (R21h) Timer Interrupt Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	**TMR2IE**	TMR1IE	TMR0IE

Bit 2 (TMR2IE): Control bit of Timer2 interrupt

"0": Disable Timer2 interrupt function
"1": Enable Timer2 interrupt function

■ **TMR2I Bit of INTSTA (R22h) Timer Interrupt Status Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	–	**TMR2I**	TMR1I	TMR0I

Bit2 (TMR2I): Status bit of Time2 interrupt

Set to "**1**" when Timer2 counter underflows

Clear ("**0**") by software

■ **Code Example:**

```
;  ===  Timer-0,1,2 Interrupt
TIMERINT:
     S0CALL PUSH
     JBS    INTSTA,TMR0I,toTM0INT
     JBS    INTSTA,TMR1I,toTM1INT
     JBS    INTSTA,TMR2I,toTM2INT
     SJMP   POP

;  ---  Timer 0 Interrupt
toTM0INT:
     BC     INTSTA,TMR0I
       :
     SJMP   POP
     RETI
;  ---  Timer 1 Interrupt
toTM1INT:
     BC     INTSTA,TMR1I
       :
     SJMP   POP
     RETI
;  ---  Timer 2 Interrupt
toTM2INT:
     BC     INTSTA,TMR2I
       :
     SJMP   POP
     RETI
```

7.5 Program ROM Map

ROM Size = 16K Words.	
Address.	**Description**
0000h ⋮ 000Bh	Interrupt Vector (12 words)
000Ch ⋮ 000Fh	Code Option (4 words)
0010h ⋮ 001Fh	Test Program (16 words)
0020h ⋮ 3FFFh	Program or Fixed data region

7.6 RAM Map for Special and Control Registers
(RAM Size: 88 Bytes + 16 Banks * 128 Bytes = 2136 Bytes)

7.6.1 Special and Control Registers

Legend: R = Readable bit **W** = Writable bit **–** = Not implemented

Addr.	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	INDF0	R/W							
		Indirect addressing Pointer0.							
1	FSR0	R/W							
		File select register 0 for INDF0 (R0)							
2	BSR	R	R	R	R	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	Fixed 0	Fixed 0	Bank select register (for INDF0 & general)			
3	INDF1	R/W							
		Indirect addressing Pointer1.							
4	FSR1	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Fixed 1	File select register 1 for INDF1 (R3)						
5	BSR1	R	R	R	R	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	Fixed 0	Fixed 0	Bank select Register 1 (for INDF1)			
6	STKPTR	R/W							
		Stack pointer							
7	PCL	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
8	PCM	R	R	R/W	R/W	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	PC13	PC12	PC11	PC10	PC9	PC8
9	LCDARL	R/W							
		LCD Ram column Address							
A	ACC	R/W							
		Accumulator							
B	TABPTRL	R/W							
		Low byte of table pointer							
C	TABPTRM	R	R/W						
		Fixed 0	Middle byte of table pointer						
D	TABPTRH	R/W							
E	LCDDATA	R/W							
		Indirect register to LCD Ram							
F	STATUS	R	R	R/W	R/W	R/W	R/W	R/W	R/W
		/TO	/PD	SGE	SLE	OV	Z	DC	C
10	PORTA	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Port A.7	Port A.6	Port A.5	Port A.4	Port A.3	Port A.2	Port A.1	Port A.0
11	PORTB	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Port B.7	Port B.6	Port B.5	Port B.4	Port B.3	Port B.2	Port B.1	Port B.0
12	PORTC	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Port C.7	Port C.6	Port C.5	Port C.4	Port C.3	Port C.2	Port C.1	Port C.0

(Continuation)

Addr.	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
20	STBCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		SCAN	KE	R1EN	BitST	STB3	STB2	STB1	STB0
21	INTCON	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	TMR2IE	TMR1IE	TMR0IE
22	INTSTA	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	TMR2I	TMR1I	TMR0I
23	TR01CON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		T1WKEN	T1EN	T1PSR1	T1PSR0	T0EN	T0CS	T0PSR1	T0PSR0
24	TRL0L	R/W							
		Timer0 auto-reload register low byte							
25	TRL0H	R/W							
		Timer0 auto-reload register high byte							
26	TRL1	R/W							
		Timer1 auto-reload register							
27	TR2WCON	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
		WDTEN	-	WDTPSR 1	WDTPSR 0	T2EN	T2CS	T2PSR1	T2PSR0
28	TRL2	R/W							
		Timer2 auto-reload register							
29	PACON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7PU	PA6PU	PA5PU	PA4PU	PA3PU	PA2PU	PA1PU	PA0PU
2A	PAWAKE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		WKEN7	WKEN6	WKEN5	WKEN4	WKEN3	WKEN2	WKEN1	WKEN0
2B	PAINTEN	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7IE	PA6IE	PA5IE	PA4IE	PA3IE	PA2IE	PA1IE	PA0IE
2C	PAINTSTA	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7I	PA6I	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
2D	DCRA	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7DC	PA6DC	PA5DC	PA4DC	PA3DC	PA2DC	PA1DC	PA0DC
2E	PBCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PB7PU	PB6PU	PB5PU	PB4PU	PB3PU	PB2PU	PB1PU	PB0PU
2F	DCRB	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PB7DC	PB6DC	PB5DC	PB4DC	PB3DC	PB2DC	PB1DC	PB0DC
30	PCCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PC7PU	PC6PU	PC5PU	PC4PU	PC3PU	PC2PU	PC1PU	PC0PU
31	DCRC	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PC7DC	PC6DC	PC5DC	PC4DC	PC3DC	PC2DC	PC1DC	PC0DC
32	LCDCON	-	R/W	R/W	-	R/W	R/W	-	R/W
		-	BLANK	LCDON	-	LCR1	LCR0	-	LBVON
33	POST_ID	-	R/W	R/W	R/W	-	R/W	R/W	R/W
		-	LCD_ID	FSR1_ID	FSR0_ID	-	LCD_PE	FSR1_PE	FSR0_PE
34	CPUCON	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	GLINT	MS1	MS0

7.6.2 Other Unbanked General RAM

Address	Unbanked
13h ⋮ 1Fh	General purpose RAM
35h ⋮ 7Fh	General purpose RAM

7.6.3 Banked General RAM

Address	Bank 0	Bank 1	Bank 2	Bank 3	……	Bank 15
80h ⋮ FFh	General Purpose RAM	General Purpose RAM	General Purpose RAM	General Purpose RAM	……	General Purpose RAM

7.7 LCD RAM Map

■ **COM0 (Bit0) ~ COM7 (Bit7)**

RAM Address LCDARL		COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7
		Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7
SEG0	**00H**								
⋮	⋮								
SEG59	**3BH**								

■ **COM8 (Bit0) ~ COM10 (Bit2)**

RAM Address LCDARL		COM8	COM9	COM10
		Bit0	Bit1	Bit2
SEG0	**40H**			
⋮	⋮			
SEG59	**7BH**			

7.8 Special Register Descriptions

7.8.1 ACC (R0Ah): Accumulator Register

Internal data transfer or instruction operand holding

7.8.2 POST_ID (R33h): Post Increase / Decrease Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	LCD_ID	**FSR1_ID**	**FSR0_ID**	–	LCDPE	**FSR1PE**	**FSR0PE**

Bit 0 (FSR0PE): Enable FSR0 post increase/decrease function. FSR0 will NOT carry into or borrow from BSR.

Bit 1 (FSR1PE): Enable FSR1 post increase/decrease function. FSR1 will carry into or borrow from BSR1.

Bit 4 (FSR0_ID): **"1":** auto increase FSR0
"0": auto decrease FSR0

Bit 5 (FSR1_ID): **"1":** auto increase FSR1
"0": auto decrease FSR1.

7.8.3 BSR, FSR0, INDF0 (R02h, R01h, R00h): Indirect Address Pointer 0 Registers

BSR (R02h) determines which bank is active (working bank) among the 16 banks (Bank0 ~ Bank15).

FSR0 (R01h) is an address register for INDF0. You can select up to 256 bytes (Address: 00 ~ 0FFh).

INDF0 (R00h) is not a physically implemented register.

7.8.4 BSR1, FSR1, INDF1 (R05h, R04h, R03h): Indirect Address Pointer 1 Registers

BSR1 (R05h) is a bank register for INDF1. It cannot determine the working bank for general register.

FSR1 (R04h) is an address register for INDF1. You can select up to 128 bytes (Address**:** 80 ~ 0FFh). Bit 7 of FSR1 is fixed to "**1**."

INDF1 (R03h) is not a physically implemented register.

■ **Code Example:**

```
Data transform bank0 to bank1:
     MOV    A,#00110011B        ; Enable FSR0 & FSR1 post increase
     MOV    POST_ID,A
     BANK   #0                  ; BSR = 0 working Bank
     MOV    A,#1
     MOV    BSR1,A              ; BSR1 = 1 is Bank 1
     MOV    A,#80H
     MOV    FSR0,A              ; FSR0 = 80H
     CLR    FSR1  ; FSR1 = 80H
     MOV    A,#80H
     RPT    ACC
     MOVRP  INDF1,INDF0         ; Move 80H ~ 0FFH data to Bank1
       :
```

■ INDF1 Linear Address Capabilities

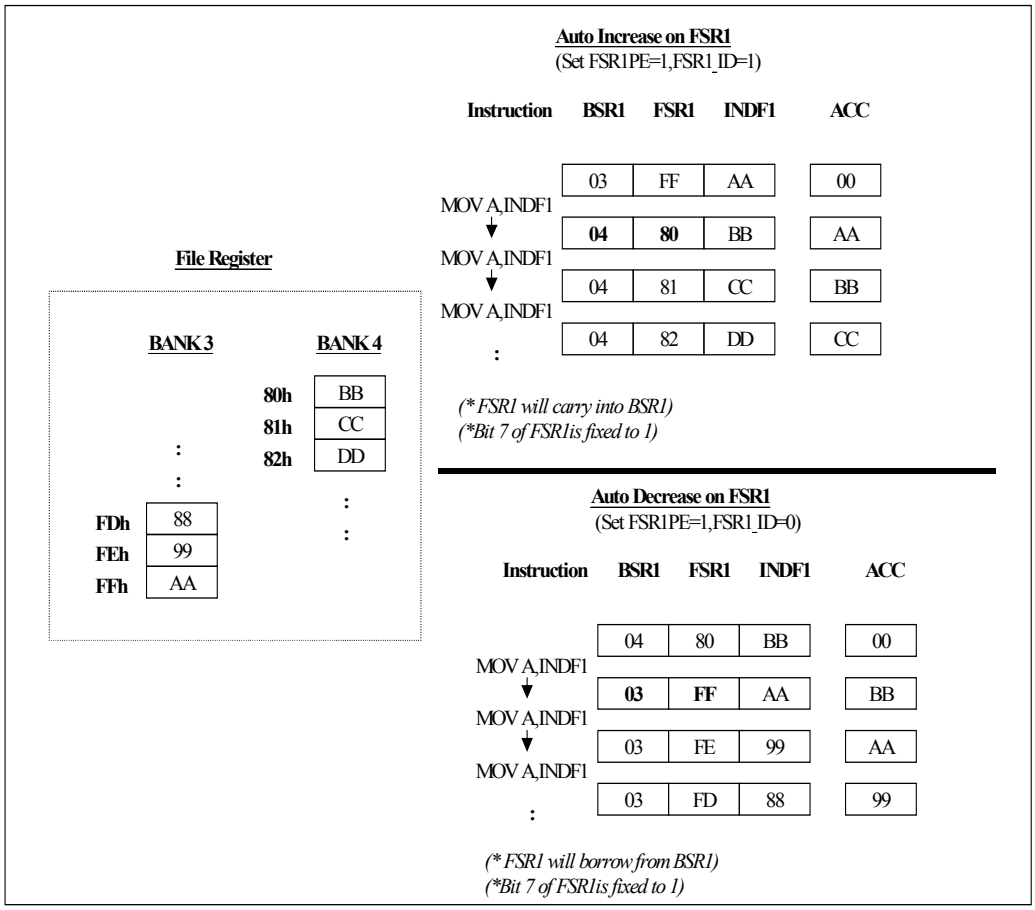


Figure 7-8 INDF1 Linear Address Capabilities Diagram

■ Code Example:

```
;*********************************
;*  Const => Working bank setting
;*  REG => Save or Recall register
;*********************************
;
;  *****  RAM stack macro
;  ***  Initial RAM stack
IniRAMsk  MACRO #Const
     MOV    A,#Const
     MOV    BSR1,A
     CLR    FSR1
     BS     POST_ID,FSR1PE
     ENDM
;  ***  Push RAM stack
PushRAM   MACRO        REG
     BS     POST_ID,FSR1_ID
     MOVRP  INDF1,REG
     ENDM
;  ***  Pop RAM stack
PopRAM    MACRO        REG
     BC     POST_ID,FSR1_ID
     MOVPR  REG,INDF1
     ENDM
```

```
;  ***  Main start program
Mstart:
       :
       :
     IniRAMsk  #29
       :
       :
MnLoop:
       :
       :
     LJMP   MnLoop

;  ***  Interrupt routine
IntSR:
     PushRAM   ACC
     PushRAM   Status
       :
       :
       :
     PopRAM    Status
     PopRAM    ACC
     RETI
```