ePS6800

RISC II
Series MCU

Product Specification

DOC. VERSION 1.0

ELAN MICROELECTRONICS CORP.

November 2010

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 1st Road
Hsinchu Science Park
Hsinchu, TAIWAN 30076
Tel: +886 3 563-9977
Fax: +886 3 563-9966
webmaster@emc.com.tw
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

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

Korea:
Elan Korea Electronics Company, Ltd.
301 Dong-A Building
632 Kojan-Dong, Namdong-ku
Incheon City, KOREA
Tel: +82 32 814-7730
Fax: +82 32 813-7730

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
elan-sz@elanic.com.cn

Shanghai:
Elan Microelectronics Shanghai, Ltd.
3F, Building #13
No. 116, Lane 572, Bibo Road
Zhangjiang Hi-Tech Park
Shanghai, CHINA 201203
Tel: +86 21 5080-3866
Fax: +86 21 5080-4600
elan-sh@elanic.com.cn

Contents

Specification Revision History

Doc. Version	Revision Description	Date
0.0	Initial preliminary version.	2005/11/10
0.1	Modified the LCD size, special RAM, and code option.	2005/11/18
0.2	Modified the register description particularly Bits 0~3 of "LCDCON", and Bits 2~3 of "LCDARH". Modified the LCD Driver: LCD Contrast Adjustment. Added the Frame Frequency Adjustment of LCD (Bit 2~3 of LCDARH). Modified the Pin Assignments (V0~V4 → V0 deleted, V1x~V4x,Vop).	2006/01/24
0.3	Modified the Pin Assignments: VDD, HOSCI, Reset, OSCO, OSCI. Added Input/Output Key with PA and PB. Modified the I/O Electrical Characteristics. Modified the Pin Type Circuit Diagrams.	2006/02/21
0.4	Added Pad Diagram and Pad Locations. Modified the VDD=3.0V Electrical Characteristics entries. Added 1/5 bias for Code Option, LCD driver and Application Circuit.	2007/01/25
0.5	1. Added the Regulator voltage range to Electrical characteristics. 2. Modified the Pin Type Circuit Diagrams. 3. Modified the code example: Pop interrupt register.	2007/06/28
0.6	1. Modified the Pin Assignment and Pin Description. 2. Modified the Electrical characteristics. 3. Modified the Pad Diagram and Pad Locations.	2007/08/29
0.7	1. Modified the Electrical Characteristics. 2. Added a Note on Section 6 Code Option.	2008/11/06
0.8	1. Modified Section 2 Peripheral 2. Modified Section 8 Input/Output Key	2009/08/18
1.0	1. Deleted the VDD=3.0V Electrical Characteristics entries. 2. Modified the Operating voltage to 1.5V only.	2010/11/03

1 General Description

The **ePS6800** is an 8-bit RISC MCU embedded with a 32×98 LCD driver along with two 8-bit timers, one 16-bit general timer, and a Watchdog Timer. It also has on-chip 8K bytes RAM and 64K words program ROM. It is highly ideal for advance scientific calculator application, particularly those requiring high performance and low cost solution.

The MCU core is one of ELAN's second generation RISC based IC's, known as RISC II (RII) series. The core was specifically designed for low power and portable device applications. The ePS6800 also supports Fast, Slow and Idle modes, as well as Sleep mode to enhance its low power consumption features.

IMPORTANT NOTES!

- *Do not use Register BSR (02h) Bit 7 ~ Bit 5.*
- *Do not use Register BSR1 (05h) Bit 7 ~ Bit 5.*
- *Do not use Register BSR2 (17h) Bit 7 ~ Bit 5.*
- *Do not use LCD RAM 62h ~ FFh.*
- *Do not use Registers JDNZ at FSR1 (04h) special register.*
- *Do not use Registers JDNZ at FSR2 (11h) special register.*
- *Don't to use PUSH,POP by "MOV A,r" to avoid effecting S_Z*

1.1 Application

- Scientific calculating gadgets

2 Features

MCU

- 8 bit RISC MCU
- Operating voltage: 1.2V ~ 1.8V
- Clock Source: Dual system clock
  - Low-frequency: 32kHz Internal RC oscillator / External RC oscillator / Crystal oscillator
  - High-frequency: 200kHz / 300kHz / 500kHz / 1MHz Internal RC oscillator / External RC oscillator
- One Instruction cycle time = 2 × System clock time
- Program ROM addressing: 64K words max.

- Provides Special registers and Control registers
- 104 bytes un-banked RAM and Working RAM
- 64×128 bytes banked RAM
- 32-level RAM stack
- Three sets of RAM indirect addressing pointer
- Look-up Table function is fast and efficient when combined with Repeat instruction
- Register-to-Register move instruction
- Compare and Branch in one instruction (two cycles)
- Single Repeat function (max. of 256 repeat times)
- Decimal ADD and SUB instruction
- Full range CALL and JUMP ability (two cycles)

Peripheral

- 36 general I/O pins (Port A, Port B, Port C.0~3, Port D, Port E)
- 32 COM × 98 SEG LCD driver (embedded), 1/16, 1/20, 1/24, 1/28 and 1/32 duty, 1/4 and 1/5 bias
- One 16-bit timer (Timer 0) with Event counter function
- One 8-bit timer (Timer 1) with wake-up function
- One 8-bit timer (Timer 2)
- One 8-bit Watchdog Timer
- Key I/O function with 96 keys max.

Internal Specification

- Watchdog Timer with its own on-chip RC oscillator
- MCU operating modes: Sleep Mode, Idle Mode, Slow Mode, and Fast Mode
- Supports RC oscillation and crystal oscillation for system clock
- MCU Wake-up function consists of input wake up and Timer 1 wake up
- MCU interrupt function consists of Input port interrupt and Timer interrupt (Timers 0 ~ 2)
- MCU reset function includes power-on reset, RSTB pin reset, and Watchdog Timer reset

3 Block Diagram

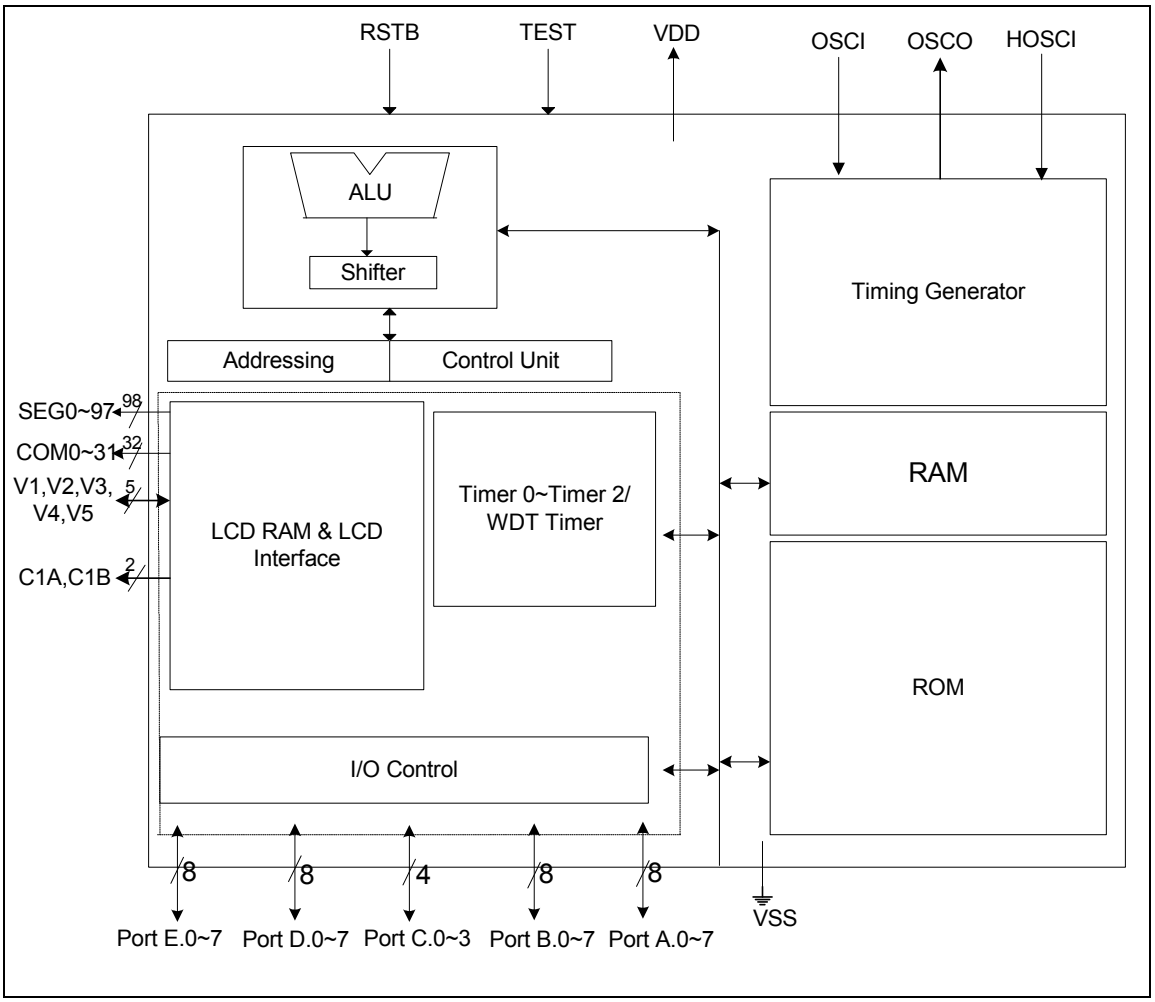


Figure 3-1 ePS6800 Block Diagram

4 Pin Assignment

■ **165-pin Chip form**

No.	Pin Name	No.	Pin Name	No.	Pin Name	No.	Pin Name
1	NC	53	NC	105	NC	157	NC
2	NC	54	NC	106	NC	158	NC
3	NC	55	NC	107	SEG59	159	NC
4	VSS	56	NC	108	SEG60	160	NC
5	C1A	57	NC	109	SEG61	161	NC
6	C1B	58	NC	110	SEG62	162	NC
7	V5	59	NC	111	SEG63	163	NC
8	V4	60	NC	112	SEG64	164	NC
9	V3	61	SEG23	113	SEG65	165	NC
10	V2	62	SEG24	114	SEG66	166	COM25/PortE.1
11	V1	63	SEG25	115	SEG67	167	COM26/PortE.2
12	COM15	64	SEG26	116	SEG68	168	COM27/PortE.3
13	COM14	65	SEG27	117	SEG69	169	COM28/PortE.4
14	COM13	66	SEG28	118	SEG70	170	COM29/PortE.5
15	COM12	67	SEG29	119	SEG71	171	COM30/PortE.6
16	COM11	68	SEG30	120	SEG72	172	COM31/PortE.7
17	COM10	69	SEG31	121	SEG73	173	VSS
18	COM9	70	SEG32	122	SEG74	174	PortA.0
19	COM8	71	SEG33	123	SEG75	175	PortA.1
20	COM7	72	SEG34	124	SEG76	176	PortA.2
21	COM6	73	SEG35	125	SEG77	177	PortA.3
22	COM5	74	SEG36	126	SEG78	178	TEST
23	COM4	75	SEG37	127	SEG79	179	PortA.4
24	COM3	76	SEG38	128	SEG80	180	PortA.5
25	COM2	77	SEG39	129	SEG81	181	PortA.6
26	COM1	78	SEG40	130	SEG82	182	PortA.7
27	COM0	79	SEG41	131	SEG83	183	PortB.0
28	SEG0/Strobe0	80	SEG42	132	SEG84	184	PortB.1
29	SEG1/Strobe1	81	SEG43	133	SEG85	185	PortB.2
30	SEG2/Strobe2	82	SEG44	134	SEG86	186	PortB.3

No.	Pin Name	No.	Pin Name	No.	Pin Name	No.	Pin Name
31	SEG3/Strobe3	83	SEG45	135	SEG87	187	PortB.4
32	SEG4/Strobe4	84	SEG46	136	SEG88	188	PortB.5
33	SEG5/Strobe5	85	SEG47	137	SEG89	189	PortB.6
34	SEG6/Strobe6	86	SEG48	138	SEG90	190	PortB.7
35	SEG7/Strobe7	87	SEG49	139	SEG91	191	PortC.0
36	SEG8/Strobe8	88	SEG50	140	SEG92	192	PortC.1
37	SEG9/Strobe9	89	SEG51	141	SEG93	193	PortC.2
38	SEG10/Strobe10	90	SEG52	142	SEG94	194	PortC.3/EVIN
39	SEG11/Strobe11	91	SEG53	143	SEG95	195	VDD
40	SEG12/Strobe12	92	SEG54	144	SEG96	196	HOSCI
41	SEG13/Strobe13	93	SEG55	145	SEG97	197	RESETB
42	SEG14/Strobe14	94	SEG56	146	COM16/PortD.0	198	OSCO
43	SEG15/Strobe15	95	SEG57	147	COM17/PortD.1	199	OSCI
44	SEG16	96	SEG58	148	COM18/PortD.2	200	NC
45	SEG17	97	NC	149	COM19/PortD.3	201	NC
46	SEG18	98	NC	150	COM20/PortD.4	202	NC
47	SEG19	99	NC	151	COM21/PortD.5	203	NC
48	SEG20	100	NC	152	COM22/PortD.6	204	NC
49	SEG21	101	NC	153	COM23/PortD.7	205	NC
50	SEG22	102	NC	154	COM24/PortE.0	206	NC
51	NC	103	NC	155	NC	207	NC
52	NC	104	NC	156	NC	208	NC

5 Pin Description

5.1 MCU System Pins (8 Pins)

Name	I/O/P Type	Description	Note
VDD	P	Digital and Analog positive power supply, ranging from 1.2V ~ 1.8V	–
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 IC internal test only, normally connect to VSS	Int. Pull Down
VSS	P	Digital and Analog negative power supply	–
OSCI	I	External RC/Crystal oscillator connecting pi	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 (137 Pins)

Name	I/O/P Type	Description	Note
COM0~COM15	O	LCD common signal output pin	–
COM16~COM23 / Port D.0~7	O I/O	LCD common signal output pin General Input/Output port	–
COM24~COM31 / Port E.0~7	O I/O	LCD common signal output pin General Input/Output port	–
SEG0~SEG15	O	LCD segment signal output pin shared with key strobe 0~15	–
SEG16~ SEG97	O	LCD segment signal output pin	–
C1A, C1B	–	LCD voltage charge-pump pin. Connect 1μF between C1A and C1B.	–
V5, V4, V3, V2, V1	O	LCD bias Pin. Connect a 1μF to Vss.	–

5.3 I/O Port (20 Pins)

Name	I/O/P Type	Description	Note
Port A.0~7	I O I	General Input port for special functions, i.e., Wake-up and Interrupt General Output port Bits 6~0: Key matrix input pins	–
Port B.0~7	I/O	General Input/Output port	–
Port C.0~3	I/O I	General Input/Output port Bit 3: Event counter input pin	–

6 Code Option

Located at Address 0x000C~0x000F of Program ROM

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

NOTE

For Initial mode after reset, it is recommended that user set it to “Slow mode”.

- Low Frequency Oscillator:
  - Select “Crystal” oscillator or “Internal RC” oscillator or “External RC” oscillator
- High Frequency Oscillator:
  - Select “Internal RC” oscillator or “External RC” oscillator
- Frequency of Internal High RC Oscillator:
  - Select “200kHz” or “300kHz” or “500kHz” or “1 MHz”
- Operating voltage option:
  - Only has “1.5V”
- RESET pin’s condition:
  - Select “Level hold” or “One shot” for reset pin
- Maximum duty ratio option:
  - Select “1/32 duty” or “1/28 duty” or “1/24 duty” or “1/20 duty” or “1/16 duty”
- LCD bias and LCD regulator voltage option:
  - Select “1/5 bias and 1.1V” or “1/5 bias and 1.2V” or “1/4 bias and 1.2V” or “1/4 bias and 1.3V” or “1/4 bias and 1.4V”
- Port D low nibble control bit (COM16~COM19):
  - Select “LCD common signal output” or “general I/O functions”
- Port D high nibble control bit (COM20~COM23):
  - Select “LCD common signal output” or “general I/O functions”
- Port E low nibble control bit (COM24~COM27):
  - Select “LCD common signal output” or “general I/O functions”
- Port E high nibble control bit (COM28~COM31):
  - Select “LCD common signal output” or “general I/O functions”

7 Functional 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

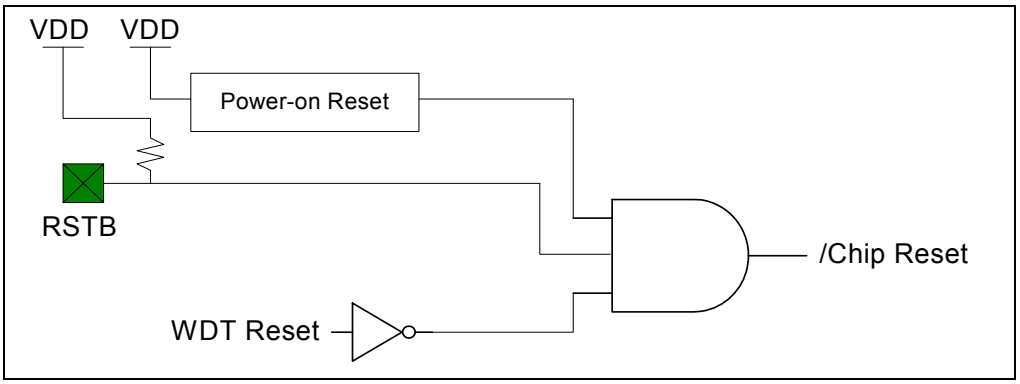


Figure 7-1 On-chip Reset Schematic

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 lowers to below Vpor, a reset will occur.

7.1.2 RSTB Pin

In normal condition, the RSTB pin is pulled up to VDD. Whenever the RSTB is at low condition (level hold or one short), a reset will occur.

7.1.3 WDT Time Out

When the Watchdog Timer is enabled, the WDT time-out will cause the chip to reset. To prevent reset from occurring, user 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 32768Hz Crystal Stable Time

- Power-on Reset Timing:

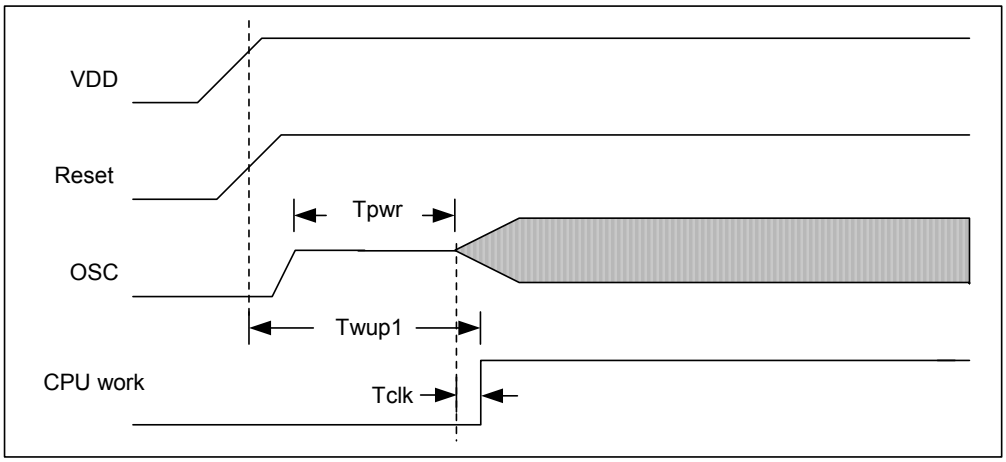


Figure 7-2 Power-on Reset Timing Diagram

- Sleep Mode Wake-up Timing:

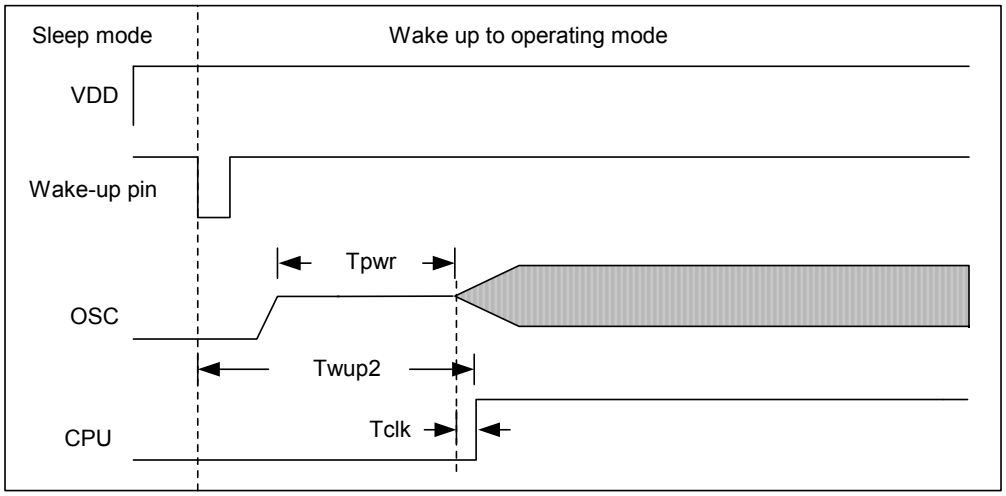


Figure 7-3 Sleep Mode Wake-up Timing Diagram

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

Symbol	Characteristics	Min.	Typ.	Max.	Unit
Tpwr	Oscillator start up time	–	480	780	ms
Twup1	CPU warm up time (Power-on reset)	–	500	800	ms
Twup2	CPU warm up time (Sleep mode wakeup)	–	485	785	ms
Tclk	Detect slow clock time	–	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

Bit 0 (C): Carry flag or inverse of Borrow flag (B)
Under SUB operation, borrow flag is indicated by the inverse of the carry bit (B = /C).

Bit 1 (DC): Auxiliary carry flag.

Bit 2 (Z): Zero flag

Bit 3 (OV): Overflow flag. Used in signed operation when Bit 6 is carried into or borrows from signed bit (Bit 7).

Bit 4 (SLE): Computation result is less than or equal to zero (negative value) after signed arithmetic. This bit is affected by HEX arithmetic instruction only.

Bit 5 (SGE): Computation result is greater than or equal to zero (positive value) after signed arithmetic. This bit is affected by HEX arithmetic instruction only.

NOTE

1. *When OV=1 after signed arithmetic, user can check the 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, user should clear the MSB of Accumulator to obtain the correct value.*
   When underflow took place, user 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 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

Bit 6 (/PD): Reset to 0 when /PD enters Sleep mode. Set to 1 by "WDTC" instruction, power-on reset, or by a Reset pin low condition.

Bit 7 (/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, the 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

- 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 (PCH: PCM: PCL) is cleared to all "0."
- The subsequent table shows the other registers' initial values.

7.1.6.1 Special Register

Addr.	Name	Initial Value	Addr.	Name	Initial Value
00h	INDF0	---- ---- [1]	10h	INDF2	---- ---- [1]
01h	FSR0	0000 0000	11h	FSR2	1000 0000
02h	BSR	0000 0000	12h	BSR2	0000 0000
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	PCH	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	0000 0000	1Dh	General RAM	uuuu uuuu
0Eh	LCDDATA	---- ---- [1]	1Eh	General RAM	uuuu uuuu
0Fh	STATUS	cuxx xxxx [2]	1Fh	General RAM	uuuu uuuu

7.1.6.2 Control Register

Addr.	Name	Initial Value	Addr.	Name	Initial Value
20h	CPUCON	0--- -00c [3]	30h	STBCON	0000 0000
21h	POST_ID	-111 -000	31h	PORTA	xxxx xxxx
22h	LCDARL	0000 0000	32h	PACON	0000 0000
23h	LCDARH	0000 --00	33h	DCRA	1111 1111
24h	INTSTA	---- -000	34h	PAWAKE	0000 0000
25h	TR0CON	--00 0000	35h	PAINTEN	0000 0000
26h	TRL0L	uuuu uuuu	36h	PAINTSTA	0000 0000
27h	TRL0H	uuuu uuuu	37h	PORTB	xxxx xxxx
28h	T0CL	0000 0000	38h	PBCON	0000 0000
29h	T0CH	0000 0000	39h	DCRB	1111 1111
2Ah	TR1CON	0--0 0-00	3Ah	PORTC	---- xxxx
2Bh	TRL1	uuuu uuuu	3Bh	PCCON	---- 0000
2Ch	TR2WCON	0000 0000	3Ch	DCRC	---- 1111
2Dh	TRL2	uuuu uuuu	3Dh	General RAM	uuuu uuuu
2Eh	LCDCON	0000 0000	3Eh	General RAM	uuuu uuuu
2Fh	- [4]	0000 0000	3Fh	General RAM	uuuu uuuu

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 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] *Bit 0 (MS0) of RE (CPUCON) is reloaded from "**INIM**" bit of code option when MCU resets.*

[4] *The register R2F must be fixed to 0x00.*

7.2 Oscillator System

The oscillator system is used to generate the device clock. The oscillator system is composed of an Internal RC, or External RC, or crystal oscillator for slow mode and an Internal RC, or External RC oscillator for Fast mode as shown in the diagram below.

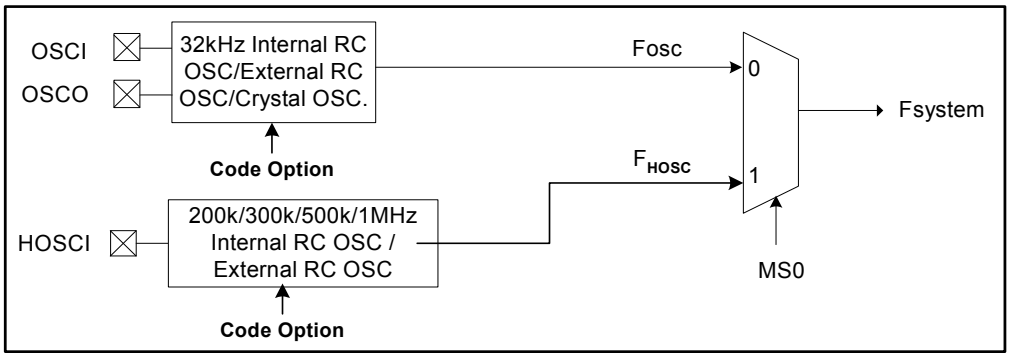


Figure 7-4 Oscillator System Function Block Diagram

The **MS0** bit (mode select bit) of the **CPUCON** register (R20h) is used to set to either 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 the OSCI and OSCO pins to remain floating.

- 32.8kHz RC External oscillator:

  Select “RC oscillator for FOSC” in the code option and allow the OSCO pin to remain floating. A resistor should be connected between the OSCI and VDD pins.

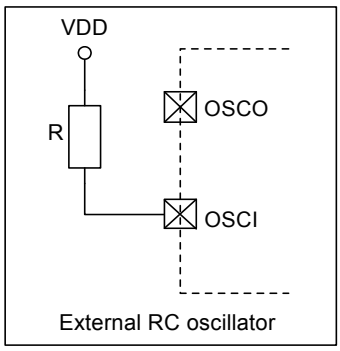


Figure 7-5 Slow Mode RC Oscillators Circuit Diagram

- 32768kHz Crystal oscillator:

  Select “Crystal oscillator for FOSC” in the code option and connect a crystal between the OSCI and OSCO pins. The OSCI and OSCO pins are also connected to ground through a 20 pF capacitor.

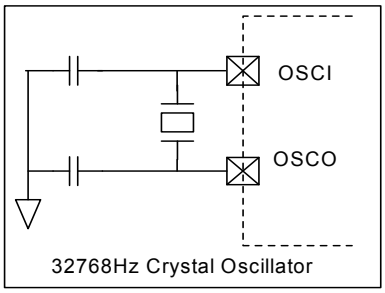


Figure 7-6 Slow Mode Crystal Circuit Diagram

7.2.2 200kHz/300kHz/500kHz/1 MHz RC Oscillator

- 200kHz/300kHz/500kHz/1 MHz RC Internal oscillator:

  Select “RC oscillator for FHOSC” in the code option and allow the HOSCI pin to remain floating.

- 200kHz/300kHz/500kHz/1MHz RC External oscillator:

  Select “RC oscillator for FHOSC” in the code option. A resistor should be connected between the HOSCI and VDD pin.

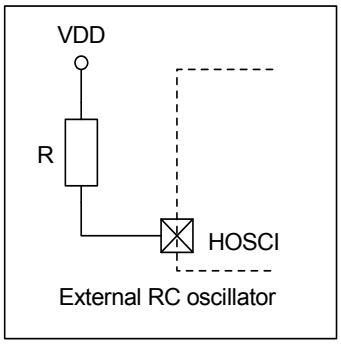


Figure 7-7 Fast Mode RC Oscillators Circuit Diagram

7.3 MCU Operation Mode:

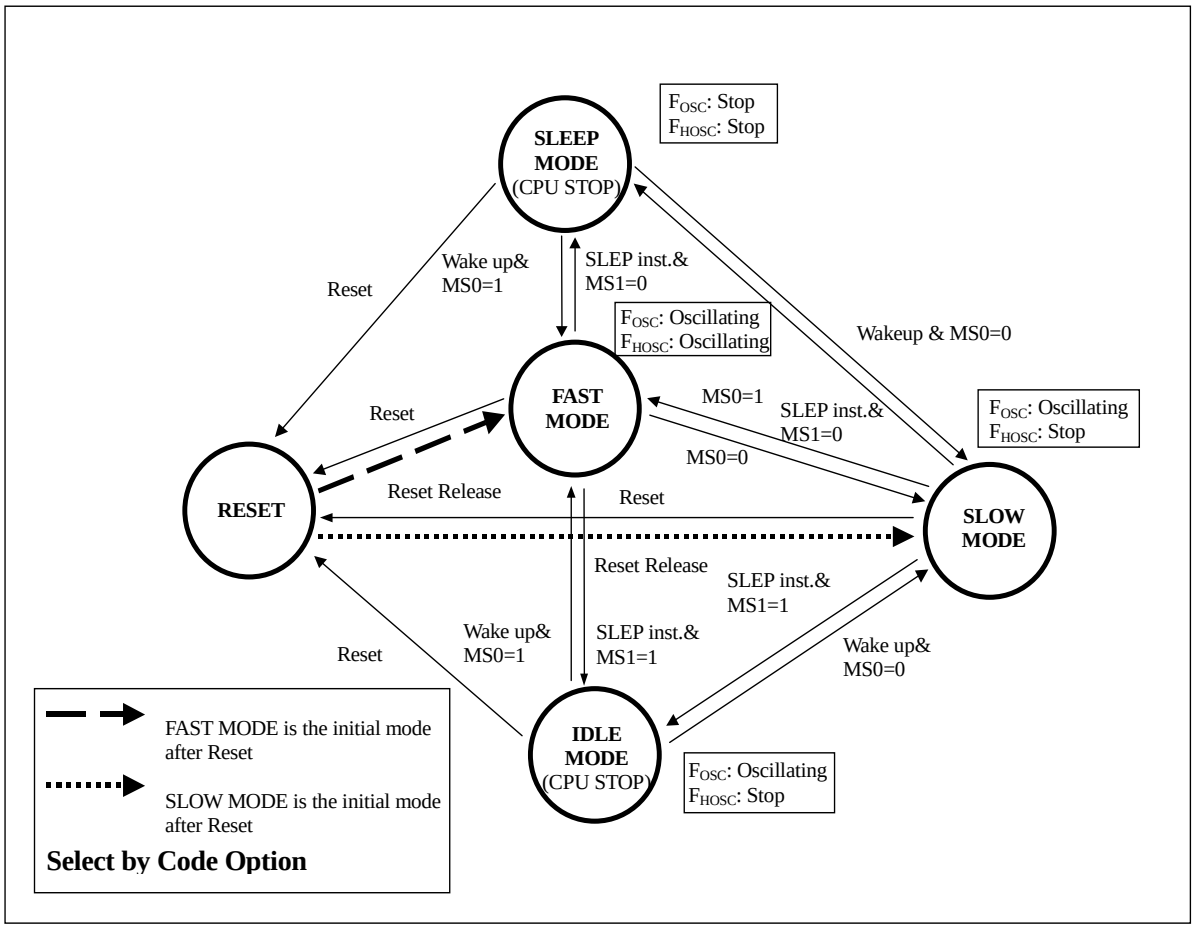


Figure 7-8 MCU Operation Block Diagram

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

Device \ Mode	Sleep	Idle	Slow	Fast
Osc.(32768Hz)	×	✓	✓	✓
Fsystem	×	×	From Osc.	From Hosc.
Timers 0~2	×	×	✓	✓
INT	×*	×*	✓	✓
I/O wake up	✓	✓	×	×
Timer 1 wake up	×	✓	×	×

Legend: ✓ = Function is available if enabled ×: Function is NOT supported

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

7.3.1 Slow, Fast, Sleep, and Idle Modes Operation

- **CPUCON (R20h): MCU Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WBK	–	–	–	–	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 the MS0 bit of the 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.

- **Fast Mode**

  When the MS0 bit of the 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 performance of all the system operations at fast speed, but under highest consumption of power.

- **Idle Mode**

  When the MS1 bit of the 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 32 kHz oscillator. It retains the internal status under low power consumption without stopping the clock function.

  The Idle Mode is awakened by the Timer 1 wake-up or by I/O pins wake-up (if enabled) and returns to either Slow Mode (MS0=0) or Fast Mode (MS0=1)

NOTE

All registers remain unchanged during Idle Mode.

- **Sleep Mode**

  When the 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 suspension of the operation. Sleep Mode operates with very low power consumption and is awakened by I/O pins wake up.

NOTE

- *The /PD bit of the Status Register (RFh) is cleared when the 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:**

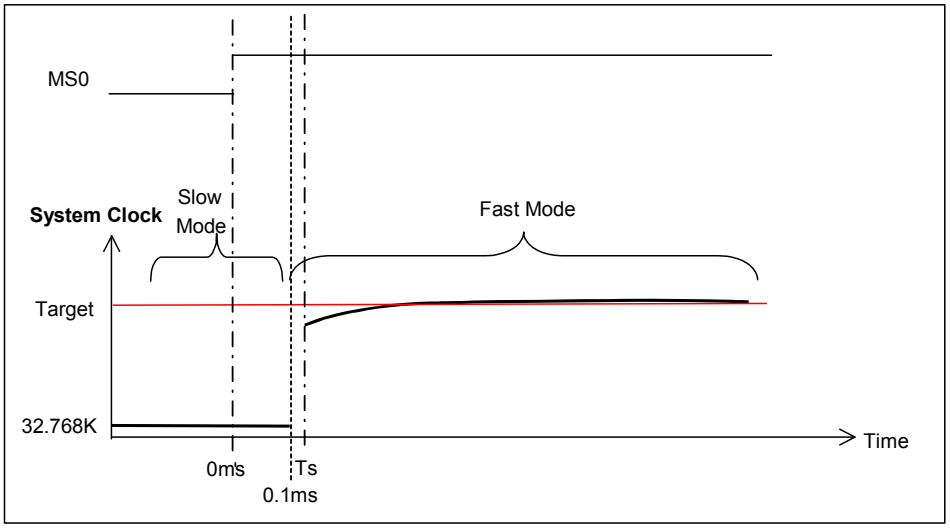


Figure 7-9 Slow Mode to Fast Mode Timing Diagram

NOTE

1. *Slow Mode switches to Fast Mode at Time=0 ms.*
2. *System clock will switch to Fast Mode after a delay of 0.1ms by oscillator and enters into Fast Mode (i.e., system clock will be at 200, 300, or 500kHz).*
3. *High frequency RC will be stabilized at Time=Ts (15 µs~30 µs).*

■ **Code Example**

```
;Enter Fast mode
     BS    CPUCON,MS0

;Entry Slow mode
     BC    CPUCON,MS0
```

```
;Enter Idle mode
     BS      CPUCON,MS1
     SLEP
     NOP

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

7.3.2 Wake-up Operation

The oscillator is off during Sleep Mode. The MCU is waken-up by an input port (Port A), then it returns to Fast Mode or Slow Mode (as determined by the MS0 bit of the CPUCON register as described in the previous section).

When in Idle Mode, the 32kHz oscillator keeps on running. The MCU is waken-up by the input port (Port A) or Timer 1, and then returns to Fast Mode or Slow Mode (as determined by the MS0 bit of the CPUCON register as described in the previous section).

■ **PAWAKE (R34h): 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 (R2Ah): Timer 1 Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1WKEN	–	–	TMR1IE	T1EN	–	T1PSR1	T1PSR0

Bit 7 (T1WKEN): Timer 1 underflow wake-up function control bit in Idle Mode

0: Disable Timer 1 wake-up function

1: Enable Timer 1 wake-up function

7.4 Interrupt

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

7.4.1 Global Interrupt

■ **GLINT Bit of CPUCON (R20h) MCU Control Register**

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

Bit 2 (GLINT): Global interrupt control bit

0: Disable all interrupts, including Level 1 ~ Level 5

1: Enable all un-masked interrupts

■ **Interrupt Vector**

Interrupt Level	Interrupt Source	Start Address	Remark
	RESET	0x00000	–
Level 1	PortA.7 ~ 0	0x00002	PAINT
Level 2	Reserved	0x00004	Reserved
Level 3	Reserved	0x00006	Reserved
Level 4	Timers 0~2	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 (R36h): 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 to **"0"** by software.

- **PAINTEN (R35h): 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 Port A Interrupt
PAINT:
     S0CALL PUSH
     CLR    PAINTSTA
        :
     S0CALL POP
     RETI
```

7.4.3 Timer 0, Timer 1, and Timer 2 Interrupts

7.4.3.1 Timer 0 Interrupt

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

■ **TMR0IE Bit of TR0CON (R25h) Timer 0 Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	T0ENMD	**TMR0IE**	T0EN	T0CS	T0PSR1	T0PSR0

Bit 4 (TMR0IE): Control bit of Timer 0 interrupt

0: Disable Timer 0 interrupt function

1: Enable Timer 0 interrupt function

■ **TMR0I Bit of INTSTA (R24h) Timer Interrupt Status Register**

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

Bit 0 (TMR0I): Status bit of Timer 0 interrupt

Set to **"1"** when Timer 0 counter underflows

Clear to **"0"** by software

7.4.3.2 Timer 1 Interrupt

Timer 1 is an 8-bit timer used for time counting and wake-up functions. When the counting value of Timer 1 underflows, interrupt occurs and the TRL1 value is reloaded to the timer.

■ **TMR1IE Bit of TR1CON (R2Ah) Timer 1 Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1WKEN	–	–	**TMR1IE**	T1EN	–	T1PSR1	T1PSR0

Bit 4 (TMR1IE): Control bit of Timer1 interrupt

0: Disable Timer 1 interrupt function

1: Enable Timer 1 interrupt function

■ **TMR1I Bit of INTSTA (R24h) Timer Interrupt Status Register**

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

Bit 1 (TMR1I): Status bit of Timer 1 interrupt

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

Clear to **"0"** by software.

7.4.3.3 Timer 2 Interrupt

Timer 2 is an 8-bit timer for time counting. When the counting value of Timer 2 underflows, interrupt occurs and the TRL2 value will be reloaded to the timer.

- **TMR2IE Bit of TR2WCON (R2Ch) Timer 2 / Watchdog Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTEN	WDTPSR1	WDTPSR0	**TMR2IE**	T2EN	T2CS	T2PSR1	T2PSR0

Bit 4 (TMR2IE): Control bit of Timer 2 interrupt

0: Disable Timer 2 interrupt function

1: Enable Timer 2 interrupt function

- **TMR2I Bit of INTSTA (R24h) Timer Interrupt Status Register**

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

Bit 2 (TMR2I): Status bit of Time 2 interrupt

Set to **"1"** when Timer 2 counter underflows

Clear to **"0"** by software.

7.4.3.4 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 = 64K Words	
Address	**Description**
0000h | 000Bh	Interrupt Vector (12 words)
000Ch | 000Fh	Code Option (4 words)
0010h | 001Fh	Test Program (16 words)
0020h | 3FFFh	0020h | 3FFFh
4000h | 7FFFh	Segment 2 | Segment 3
8000h | BFFFh	Segment 4 | Segment 5
C000h | FFFFh	Segment 6 | Segment 7

7.6 RAM Map for Special Function and Control Registers

(RAM Size: 104 Bytes + 64 Banks × 128 Bytes = 8296 Bytes)

7.6.1 Special Function 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 Pointer 0							
1	FSR0	R/W							
		File Select Register 0 for INDF0 (R0)							
2	BSR	R	R	R/W	R/W	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	Bank select register (for INDF0 and general)					
3	INDF1	R/W							
		Indirect Addressing Pointer 1							
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/W	R/W	R/W	R/W	R/W	R/W
		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/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PC15	PC14	PC13	PC12	PC11	PC10	PC9	PC8
9	PCH	R	R	R	R	R	R	R	R/W
		Fixed 0	Fixed 0	Fixed 0	Fixed 0	Fixed 0	Fixed 0	Fixed 0	PC16
A	ACC	R/W							
		Accumulator							
B	TABPTRL	R/W							
		Low byte of Table Pointer							
C	TABPTRM	R/W							
		Middle byte of Table Pointer							
D	TABPTRH	R	R	R	R	R	R	R/W	
		Fixed 0	Fixed 0	Fixed 0	Fixed 0	Fixed 0	Fixed 0	High byte of table pointer	
E	LCDDATA	R/W							
		Indirect register to LCD RAM							

Addr.	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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	INDF2	R/W							
		Indirect Addressing Pointer 2							
11	FSR2	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Fixed 1	File select register 1 for INDF2 (R15)						
12	BSR2	R	R	R/W	R/W	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	Bank Select Register 2 (for INDF2)					
20	CPUCON	R/W	-	-	-	-	R/W	R/W	R/W
		WBK	-	-	-	-	GLINT	MS1	MS0
21	POST_ID	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		FSR2_ID	LCD_ID	FSR1_ID	FSR0_ID	FSR2_PE	LCD_PE	FSR1_PE	FSR0_PE
22	LCDARL	R/W							
		LCD RAM Column Address							
23	LCDARH	R/W	R/W	R/W	R/W	-	-	R/W	R/W
		ADJ3	ADJ2	ADJ1	ADJ0	-	-	LCDARH1	LCDARH0
24	INTSTA	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	TMR2I	TMR1I	TMR0I
25	TR0CON	-	-	R/W	R/W	R/W	R/W	R/W	R/W
		-	-	T0ENMD	TMR0IE	T0EN	T0CS	T0PSR1	T0PSR0
26	TRL0L	R/W							
		Timer 0 auto-reload register low byte							
27	TRL0H	R/W							
		Timer 0 auto-reload register high byte							
28	T0CL	R/W							
		Timer 0 counting value register low byte							
29	T0CH	R/W							
		Timer 0 counting value register high byte							
2A	TR1CON	R/W	-	-	R/W	R/W	-	R/W	R/W
		T1WKEN	-	-	TMR1IE	T1EN	-	T1PSR1	T1PSR0
2B	TRL1	R/W							
		Timer 1 auto-reload register							
2C	TR2WCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		WDTEN	WDTPSR1	WDTPSR0	TMR2IE	T2EN	T2CS	T2PSR1	T2PSR0
2D	TRL2	R/W							
		Timer 2 auto-reload register							

Addr.	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2E	LCDCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		R1EN	BLANK	LCDON	SFR2	SFR1	SFR0	LCR1	LCR0
2F	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		0[1]	0[1]	0[1]	0[1]	0[1]	0[1]	0[1]	0[1]
30	STBCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		KE	SCAN	BitST	ALL	STB3	STB2	STB1	STB0
31	PORTA	R/W	R/W	R/W	R/W	R	R	R	R
		Port A.7	Port A.6	Port A.5	Port A.4	Port A.3	Port A.2	Port A.1	Port A.0
32	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
33	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
34	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
35	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
36	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
37	Port B	R/W	R/W	R/W	R/W	R	R	R	R
		Port B.7	Port B.6	Port B.5	Port B.4	Port B.3	Port B.2	Port B.1	Port B.0
38	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
39	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
3A	PORTC	-	-	-	-	R/W	R/W	R/W	R/W
		-	-	-	-	Port C.3	Port C.2	Port C.1	Port C.0
3B	PCCON	-	-	-	-	R/W	R/W	R/W	R/W
		-	-	-	-	PC3PU	PC2PU	PC1PU	PC0PU
3C	DCRC	-	-	-	-	R/W	R/W	R/W	R/W
		-	-	-	-	PC3DC	PC2DC	PC1DC	PC0DC
3D	PORTD	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Port D.7	Port D.6	Port D.5	Port D.4	Port D.3	Port D.2	Port D.1	Port D.0
3E	PORTE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Port E.7	Port E.6	Port E.5	Port E.4	Port E.3	Port E.2	Port E.1	Port E.0
3F	DCRDE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		EHNPU	ELNPU	EHNDC	ELNDC	DHNPU	DLNPU	DHNDC	DLNDC

0[1]: The register R2Fh must be fixed to 0x00.

7.6.2 Other Un-banked General RAM

Address	Unbanked (WBK = "0")	Unbanked (WBK = "1")
00h | 12h	Special register 19 bytes	
13h | 1Fh	General purpose RAM 13 bytes	
20h | 24h	Control register 5 bytes	
25h | 3Fh	Control register 27 bytes	General purpose RAM 27 bytes
40h | 7Fh	General purpose RAM 64 bytes	

7.6.3 Banked General RAM

Address	Bank 0	Bank 1	Bank 2	Bank 3	……	Bank 63
80h | FFh	General Purpose RAM	General Purpose RAM	General Purpose RAM	General Purpose RAM	……	General Purpose RAM

7.7 LCD RAM Map

LCD RAM		LCDARH[1:0]			
RAM Address LCDARL		11 (Page 3)	10 (Page 2)	01 (Page 1)	00 (Psge 0)
		COM31-COM24	COM23-COM16	COM15-COM8	COM7-COM0
		Bit 7 - Bit 0	Bit 7 - Bit 0	Bit 7 - Bit 0	Bit 7 - Bit 0
SEG0	**00H**				
:	**:**				
SEG97	**61H**				

7.8 Special Register Description

7.8.1 ACC (R0Ah): Accumulator Register

Internal data transfer operation, or instruction operand holding usually involves the temporary storage function of the Accumulator. The Accumulator is not an addressable register.

7.8.2 POST_ID (R21h): Post Increase / Decrease Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR2_ID	LCD_ID	FSR1_ID	FSR0_ID	FSR2PE	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 3 (FSR2PE): Enable FSR2 post increase/decrease function. FSR2 will carry into or borrow from BSR2.

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

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

Bit 7 (FSR2_ID): **0:** auto decrease FSR2
1: auto increase FSR2

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 64 banks (Bank 0 ~ Bank 63).

FSR0 (R01h) is an address register for INDF0. User 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 the general register.

FSR1 (R04h) is an address register for INDF1. User 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.

7.8.5 BSR2, FSR2, INDF2 (R12h, R11h, R10h): Indirect Address Pointer 2 Registers

BSR2 (R12h) is a bank register for INDF2. It cannot determine the working bank for the general register.

FSR2 (R11h) is an address register for INDF2. User can select up to 128 bytes (Address: 80 ~ 0FFh). Bit 7 of FSR2 is fixed to "1"

INDF2 (R10h) is not a physically implemented register.

■ **Code Example:**

```
Data transform Bank 0 to Bank 1:
     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 Bank 1
        :
```

```
Data transform Bank 0 to Bank 2:
     MOV    A,#10011001B          ; Enable FSR0 & FSR2 post increase
     MOV    POST_ID,A
     BANK   #0                    ; BSR = 0 working Bank
     MOV    A,#2
     MOV    BSR2,A                ; BSR2 = 1 is Bank 2
     MOV    A,#80H
     MOV    FSR0,A                ; FSR0 = 80H
     CLR    FSR2                  ; FSR2 = 80H
     MOV    A,#80H
     RPT    ACC
     MOVRP  INDF2,INDF0           ; Move 80H ~ 0FFH data to Bank 2
       :
```

■ INDF1 Linear Address Capabilities

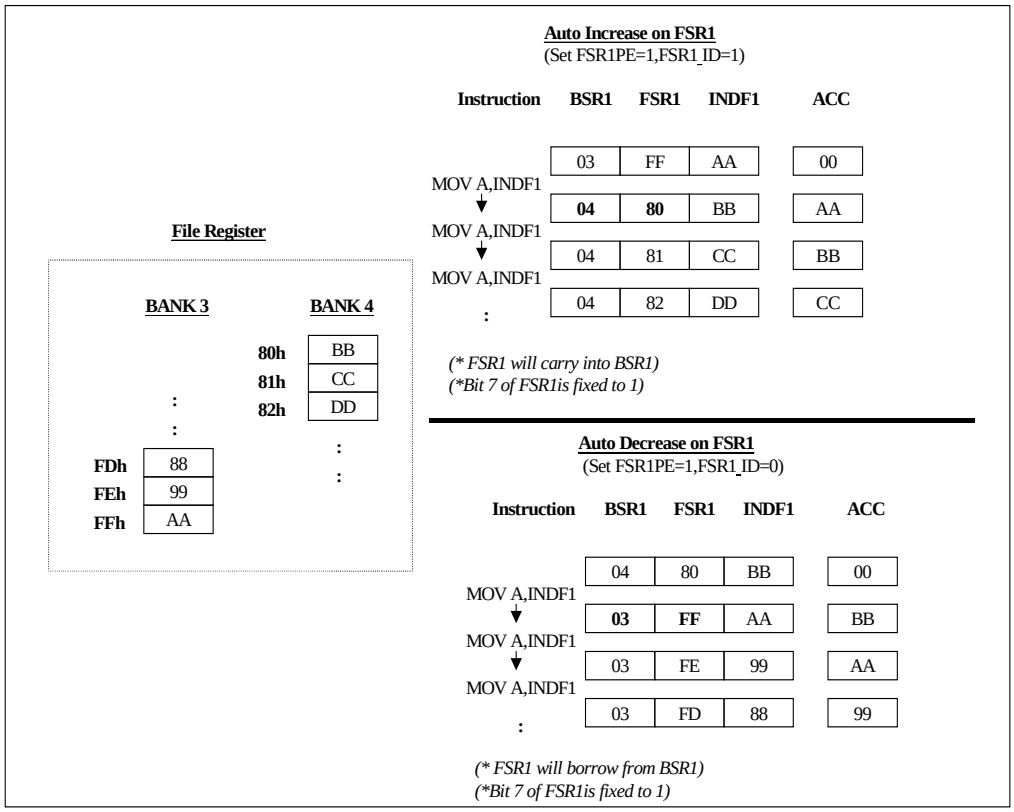


Figure 7-10 INDF1 Linear Address Capabilities Diagram

■ 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
```

■ INDF2 Linear Address Capabilities

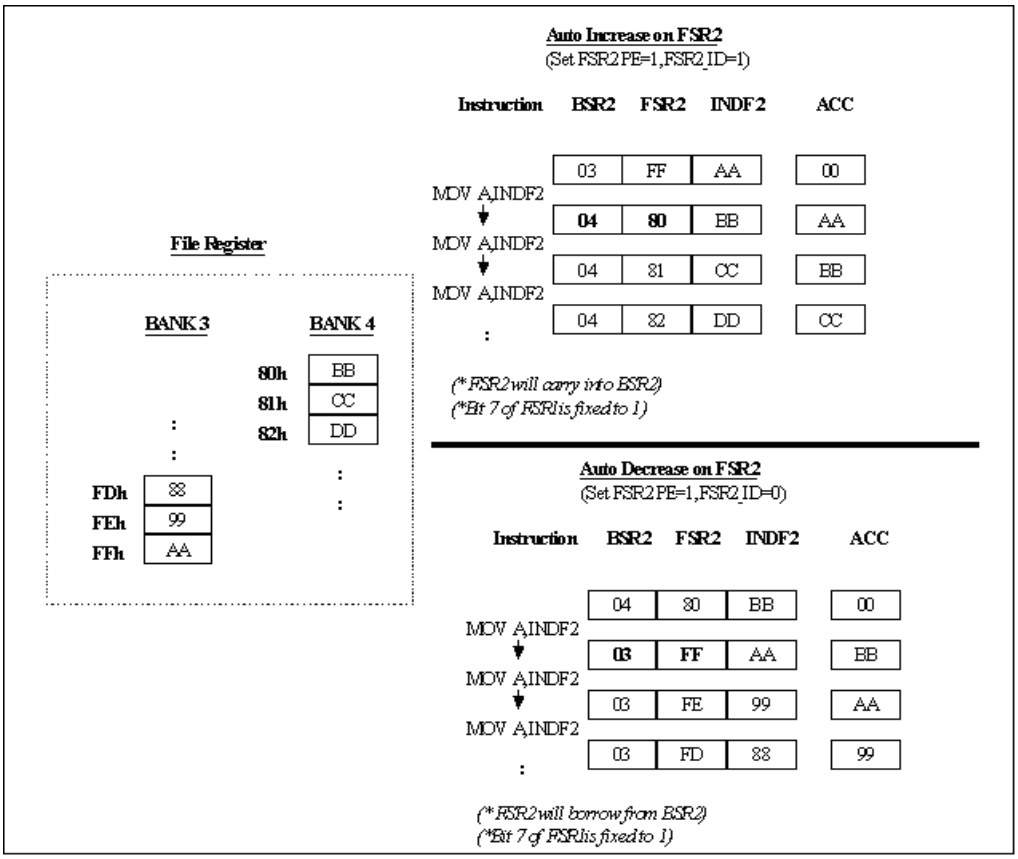


Figure 7-11 INDF2 Linear Address Capabilities Diagram

■ Code Example:

```
;************************************
;* Const => Working bank setting
;* REG => Save or Recall register
;************************************
; ***** RAM stack macro
; *** Initial RAM stack
IniRAMsk  MACRO   #Const
     MOV    A,#Const
     MOV    BSR2,A
     CLR    FSR2
     BS     POST_ID,FSR2PE
     ENDM
; *** Push RAM stack
PushRAM   MACRO  REG
     BS     POST_ID,FSR2_ID
     MOVRP  INDF2,REG
     ENDM
; *** Pop RAM stack
PopRAM    MACRO  REG
     BC     POST_ID,FSR2_ID
     MOVPR  REG,INDF2
     ENDM
```

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

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

7.8.6 STKPTR (R06h): Stack Pointer Register

The initial stack pointer is 00h. Each INT/CALL will stack (the STKPTR will subtract 2) two bytes of address with a total capacity of 32 levels. When the stack overflows, it will replace the first stack level.

NOTE

This Bank RAM does not include the stack RAM. The stack RAM is independent. You cannot see the stack RAM.

7.8.7 PCL, PCM, PCH (R07h, R08h, R09h): Program Counter Registers

Bit 23 ~ Bit17	Bit16	Bit 15 ~ Bit 8	Bit 7 ~ Bit 0
0	PCH	PCM	PCL

Generates up to 64K×16 on-chip ROM addresses for the relative programming instruction codes.

"S0CALL" loads the lower 12 bits of the PC (4K×16 ROM)

"SCALL" or **"SJMP"** loads the lower 13 bits of the PC (8K×16 ROM)

"LCALL" or **"LJMP"** loads the full 14 bits of the PC (16K×16 ROM)

"ADD R7, A" or **"ADC R7, A"** allows a relative address to be added into the current PC. The carry bit of R7 will automatically carry into the PCM.

■ **Code Example:**

```
START:
     MOV    A,entry
     MOV    number,A                    ; number ← entry
     LCALL  Indirect_JUMP
AAA:
       :
       :
Indirect_JUMP:
     MOV    A,number
     ADD    A,ACC                       ; A← 2*A
     ADD    PCL,A                       ; PCL← PCL+A
Function_Table:
     LJMP   Function_Address_1          ; number=0
     LJMP   Function_Address_2          ; number=1
     LJMP   Function_Address_3          ; number=2
     LJMP   Function_Address_4          ; number=3
     LJMP   Function_Address_5          ; number=4
     LJMP   Function_Address_6          ; number=5
     LJMP   Function_Address_7          ; number=6
       :
Function_Address_1:
       :                                ; Function 1 operation
       :
     RET                                ; PC will return to AAA label
```

7.8.8 TABPTRL, TABPTRM, TABPTH (R0Bh, R0Ch, R0Dh): Table Pointer Registers

Bit 23 ~ Bit 17	Bit 16	Bit 15~ Bit 8	Bit 7 ~ Bit 0
0	TABPTRH	TABPTRM	TABPTRL

Program ROM or Internal ROM address register.

Bit 17 ~ Bit 1 are used to point the memory address.

Bit 0 is used to select the low or high byte of the pointed word (see TBRD instruction in the Instruction Set under Section 12)

■ **Code Example:**

```
; *** Program ROM
        :
        :
    TBPTH  #(PROMTabB*2)/10000H
    TBPTM  #(PROMTabB*2)/100H
    TBPTL  #PROMTabB*2
        :
        :
    TBRD   0,ACC                        ; not change
    TBRD   1,ACC                        ; auto-increase
    TBRD   2,ACC                        ; auto-decrease
        :
```

```
; *** Program ROM data
PROMTabB:
    DB     0x00,0x01,0x02,0x03,0x04,0x05
    DB     0x10,0x11,0x12,0x13,0x14,0x15
    DB     0x20,0x21,0x22,0x23,0x24,0x25
```

7.8.9 CPUCON (R20h): MCU Control Register

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

Bit 7 (WBK): Select Working RAM

0: Set un-banked (24h~3Fh) change to Control registers

1: Set un-banked (24h~3Fh) change to General Purpose RAM

Bit 2 (GLINT): Global interrupt control bit

0: Disable all interrupts

1: Enable all un-mask interrupts

Bit 1 (MS1): Select Sleep Mode or Idle Mode after executing “SLEP” instruction

0: Sleep Mode

1: Idle Mode

Bit 0 (MS0): Select Slow Mode or Fast Mode

0: Slow Mode

1: Fast Mode

7.8.10 STBCON (R30): Strobe Output Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
KE	SCAN	BitST	ALL	STB3	STB2	STB1	STB0

Bit 7 (KE): Port A input enable/disable control bit (control at Port A.6 ~ 0)

0: Disable Port A input function

1: Enable Port A input function

Bit 6 (SCAN): Automatic key scan or specify the scan signal bit by bit

0: Key scan is specified as "Bits STB3 ~ 0 defined"

1: Auto strobe scanning

Bit 5 (BitST): Enable SEG0 ~ SEG15 as key strobe pins

0: SEG0 ~ SEG15 are used as LCD segment signal pins only

1: SEG0 ~ SEG15 are used as key strobe pins and LCD segment pins. Strobe signal is STB3 ~ 0 defined

Bit 4 (ALL): Set all strobe

0: Bit strobe

1: All strobe

Bits 3 ~ 0 (STB3 ~ 0): 16 to 1 multiplex selector of key strobe pin (control at SEG15~0)

7.8.11 Port A (R31h): General I/O Pins Register

Port A (R31h) Port A.0 ~ 7 are general I/O pin registers

7.8.12 PACON (R32h): Port A Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7PU	PA6PU	PA5PU	PA4PU	PA3PU	PA2PU	PA1PU	PA0PU

Bits 7 ~ 0 (PA7PU ~ PA0PU): Enable PortA.0 ~ PortA.7 pull-up resistor bits

0: Disable PortA.0 ~ PortA.7 pull-up resistor

1: Enable PortA.0 ~ PortA.7 pull-up resistor

7.8.13 DCRA (R33h): Port A Direction Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7DC	PA6DC	PA5DC	PA4DC	PA3DC	PA2DC	PA1DC	PA0DC

Bit 7 ~ Bit 0 (PA7DC ~ PA0DC): PortA.0~PortA.7 direction control bits

0: Set as output pin

1: Set as input pin

7.8.14 PAWAKE (R34h): Port A Wake-up 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

Bits 7 ~ 0 (WKEN7 ~ WKEN0): Wake up enable control bits of PortA.7~PortA.0.

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

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

NOTE

This function is only available with Port A selected as input pin.

7.8.15 PAINTEN (R35h): Port A Interrupt Enable 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 ~ Bit 0 (PA7IE ~ PA0IE): Interrupt Control bits

0: Disable Port A interrupt function

1: Enable Port A interrupt function

NOTE

This function is only available with Port A selected as input pin.

7.8.16 PAINTSTA (R36h): Port A 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 ~ Bit 0 (PA7I ~ PA0I): INT status of Port A.7 ~ PortA.0 interrupts bits

Set to **"1"** when a pin falling edge is detected.

Clear to **"0"** by software.

7.8.17 Port B (R37h): General I/O Pins Register

Port B (R37h) Port B.0 ~ 7 are general I/O pins register

7.8.18 PBCON (R38h): Port B Pull-up Resistor Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PB7PU	PB6PU	PB5PU	PB4PU	PB3PU	PB2PU	PB1PU	PB0PU

Bit 7 ~ Bit 0 (PB7PU ~ PB0PU): Port B.0 ~ Port B.7 pull-up resistor control bits

0: Disable pull-up resistor

1: Enable pull-up resistor

NOTE

This function is only available with Port B selected as input pin.

7.8.19 DCRB (R39h): Port B Direction Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PB7DC	PB6DC	PB5DC	PB4DC	PB3DC	PB2DC	PB1DC	PB0DC

Bit 7 ~ Bit 0 (PB7DC ~ PB0DC): Port B.0 ~Port B.7 direction control bits

0: Set as output pin

1: Set as input pin

7.8.20 Port C (R3Ah): General I/O Pins Register

Port C (R3Ah) PortC.0 ~ 3 are general I/O pins register

7.8.21 PCCON (R3Bh): Port C Pull-up Resistor Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	PC3PU	PC2PU	PC1PU	PC0PU

Bit 3 ~ Bit 0 (PC3PU ~ PC0PU): Port C.0 ~ PortC.3 pull-up resistor control bits

0: Disable pull-up resistor

1: Enable pull-up resistor

NOTE

This function is only available with Port C selected as input pin.

7.8.22 DCRC (R3Ch): Port C Direction Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	PC3DC	PC2DC	PC1DC	PC0DC

Bit 3 ~ Bit 0 (PC3DC ~ PC0DC): Port C.0 ~PortC.3 direction control bits

0: Set to output pin

1: Set to input pin

7.8.23 Port D, Port E (R3Dh, R3Eh): General I/O Pins Registers

Port D (R3Dh) Port D.0 ~ 7 are general I/O pins register

Port E (R3Eh) Port E.0 ~ 7 are general I/O pins register

7.8.24 DCRDE (R3Fh): Port D/Port E Direction Control and Pull-up Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EHNPU	ELNPU	EHNDC	ELNEC	DHNPU	DLNPU	DHNDC	DLNDC

Bit 1, Bit 5 (DHNDC, EHNDC) and Bit 0, Bit 4 (DLNDC, ELNDC): Port D and E high / low nibbles direction control.

0: Set to output pin

1: Set to input pin

Bit 3, Bit 7 (DHNPU, EHNPU) and Bit 2, Bit 6 (DLNPU, ELNPU): Enable Ports D and E high/low nibbles pull-high resistor.

0: Disable pull-up resistor

1: Enable pull-up resistor

NOTE

1. *The pull-up function is only available with Port D and E selected as input pin.*
2. *When a Ports D and E bits are set to input pin, a 15 µsec delay in reading the Port D and E data must be provided. Otherwise, the read data will be inaccurate. See Example below.*

■ **Code Example:**

```
; *** Set Port D to input pins
     MOV    A,#0X0F
     MOV    DCRDE,A
Read_PD:
     JBS    PORTD,0,Read_PD
        Delay 15usec
     JBS    PORTD,0,Read_PD
     SJMP   Read_PD
; *** Set Port E to input pins
     MOV    A,#0XF0
     MOV    DCRDE,A
Read_PE:
     JBS    PORTE,0,Read_PE
        Delay 15usec
     JBS    PORTE,0,Read_PE
     SJMP   Read_PE
```

8 Peripheral

8.1 Timer 0 (16-Bit Timer with Event Counter Function)

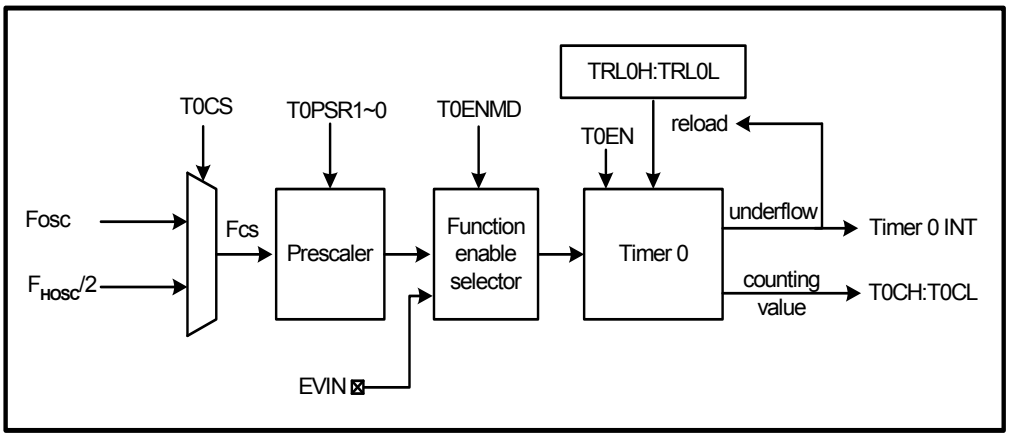


Figure 8-1 Timer 0 Functional Block Diagram

■ **Timer 0 Mode**

Timer 0 is a general-purpose 16 bits down counter used on applications that require time counting with interrupt. The clock source (FCS) is selectable from the oscillator clock (FOSC) or half of the system clock (FHOSC).

A prescaler for the timer is also provided. The T0PSR1 ~ T0PSR0 bits of TR0CON register determine the prescaler ratio and generate different clock rates as clock source for the timer.

Counter value is decremented by one (count down) according to the timer clock source frequency. When underflow occurs, the timer interrupt is triggered if the global interrupt and Timer 0 interrupt are both enabled. At the same time, TRL0H:TRL0L will automatically reload into the 16 bits counter.

$$T = \frac{1}{F_{CS}} \times \mathrm{Pr}\,escaler \times \left(TRL0H : TRL0L + 1\right)$$

■ **Event Counter Mode, EVIN (Port C.3) Pin**

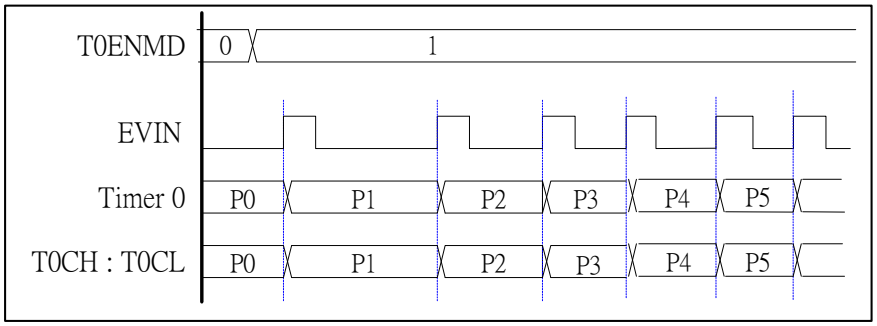
Event counter is a function that allows the 16-bit counter value to decrease by one when an event occurs on the EVIN pin at every rising edge. In other words, the clock source of Timer 0 is from an external event (EVIN pin).

The EVIN pin can be configured into event counter input function by setting the T0ENMD bit of TR0CON register (see next page) to **"1."** The counter value of Timer 0 will be stored in T0CH: T0CL registers.

NOTE

If the program uses the event counter mode, the Port C.3 will be fixed at input pin and Port C.3 cannot be controlled by the DCRC register.

■ **Event Counter Mode Example:**



8.1.1 Timer 0 Registers

■ **TRL0H:TRL0L (R26h, R27h): Timer 0 Reload Registers**

Reload registers are used to store the auto-reload value of Timer 0. When Timer 0 is enabled or underflow occurs, TRL0H:TRL0L registers will automatically reload into 16 bits counter.

■ **T0CH:T0CL(R28h, R29h): Timer 0 Counter Value Register**

Used to store the value compared with Timer 0 register.

■ **TR0CON (R25h): Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	T0ENMD	TMR0IE	T0EN	T0CS	T0PSR1	T0PSR0

Bit 5 (T0ENMD): Timer 0 / Event counter selection bit

0: Timer 0 mode

1: Event counter mode

Bit 4 (TMR0IE): Timer 0 interrupt control bit

0: Disable interrupt function

1: Enable interrupt function

Bit 3 (T0EN): Timer 0 enable control bit

0: Disable

1: Enable

Bit 2 (T0CS): Timer 0 clock source select bit

0: Clock source is from Fosc

1: Clock source is from Fhosc/2

Bit 1 ~ Bit 0 (T0PSR1 ~ T0PSR0): Timer 0 prescaler select bits

T0PSR1 : T0PSR0	Prescaler Value
00	1:1
01	1:4
10	1:16
11	1:64

■ **CPUCON (R20h): MCU Control Register**

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

Bit 2 (GLINT): Global interrupt enable/disable bit

0: Disable all interrupts

1: Enable all un-mask interrupts

■ **INTSTA (R24h): Timer Interrupt Status Register**

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

Bit 0 (TMR0I): When Timer 0 interrupt occurs, this bit will be set. Clear to "0" by software.

■ **Code Example:**

```
; === Timer 0 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR0I,Q_Time
     BC      INTSTA,TMR0I
     BTG     PORTA,7
Q_Time:
     POP
     RETI
; === Timer 0 = [1/(300K/2)] * [1 x(1FFFh + 1)]
Timer0SR:
       :
     System setting 300kHz
     PA.7 setting output pin
       :
     MOV     A,#0B00000100
     AND     TR0CON,A                ; FHOSC and Pre-scale 1:1
     MOV     A,#0X1F
     MOV     TRL0H,A
     MOV     A,#0XFF
     MOV     TRL0L,A                 ; 13.65ms=[1x(8191 + 1)/(300K/2)
     BC      TR0CON,T0ENMD           ; 0=> Timer 0 mode
     BS      TR0CON,T0EN             ; Timer 0 enable.
     BS      TR0CON,TMR0IE           ; Timer 0 interrupt enable.
     BC      INTSTA,TMR0I            ; Clear Timer 0 interrupt status
     BS      CPUCON,GLINT            ; Enable global interrupt
TimeLoop:
     SJMP    TimeLoop
```

```
; === Timer 0 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR0I,Q_Time
     BC      INTSTA,TMR0I
     BTG     PORTA,7
Q_Time:
     POP
     RETI
; === Event counter set
TR0_Event:
       :
     PA.7 setting output pin
       :
     MOV     A,#0XFF                 ; Switch 256 times reload
     MOV     TRL0L,A
     CLR     TRL0H                   ; Count start 00FFh
     BS      TR0CON,T0ENMD           ; 1 => Event counter mode
     BS      TR0CON,T0EN             ; Enable Timer 0
     BS      TR0CON,TMR0IE           ; Enable Timer 0 interrupt
     BS      CPUCON,GLINT            ; Enable global interrupt
EventWait:
     SJMP    EventWait
```

8.2 Timer 1 (8 Bits)

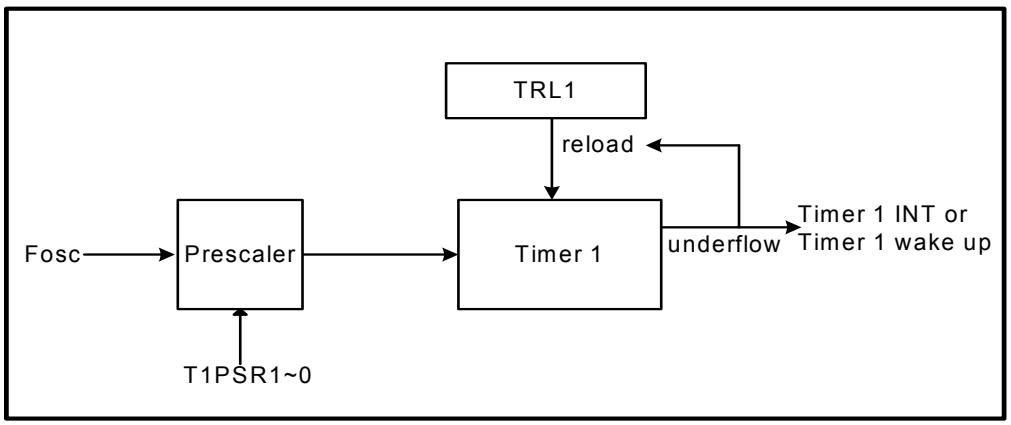


Figure 8-2 Timer 1 Function Block Diagram

Timer 1 is a general-purpose 8-bit down counter used on applications that require time counting with interrupt and wake-up functions. The clock source is from the oscillator clock (FOSC).

A prescaler for the timer is also available. The T1PSR1 ~ T1PSR0 bits of TR1CON register determine the pre-scale ratio and generate different clock rates as clock source for the timer. Setting T1WKEN bit of TR1CON register to “1” will enable the Timer 1 underflow wake-up function in Idle Mode.

The Counter value will be decremented by one (count down) according to the timer clock source frequency. When the counter underflows, the timer interrupt is triggered if the global interrupt and Timer 1 interrupt are both enabled. At the same time, TRL1 will automatically reload into 8 bits counter.

$$T = \frac{1}{F_{OSC}} \times \Pr escaler \times (TRL1 + 1)$$

8.2.1 Timer 1 Registers

- **TRL1 (R2Bh): Timer 1 Reload Register**

This register is used to store the auto-reload value of Timer 1. When Timer 1 is enabled or underflow occurs, TRL1 register will automatically reload into 8 bits counter.

- **CPUCON (R20h): MCU Control Register**

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

Bit 2 (GLINT): Global interrupt enable/disable bit

0: Disable all interrupt

1: Enable all un-mask interrupt

- **TR1CON (R2Ah): Timer 1 Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1WKEN	–	–	**TMR1IE**	**T1EN**	–	**T1PSR1**	**T1PSR0**

Bit 7 (T1WKEN): Enable bit of Timer 1 underflow wake-up function in Idle Mode

0: Disable Timer 1 wake-up function

1: Enable Timer 1 wake-up function

Bit 4 (TMR1IE): Control bit of Timer 1 interrupt

0: Disable interrupt function

1: Enable interrupt function

Bit 3 (T1EN): Timer 1 enable control bit

0: Disable Timer 1 (stop counting)

1: Enable Timer 1

Bit 1 ~ Bit 0 (T1PSR1 ~ T1PSR0): Timer 1 prescaler select bits

T1PSR1 : T1PSR0	Prescaler Value
00	1:4
01	1:16
10	1:64
11	1:256

■ **INTSTA (R24h): Timer Interrupt Status Register**

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

Bit 1 (TMR1I): When Timer 1 interrupt occurs, this bit will be set. Clear to "0" by software.

■ **Code Example:**

```
; === Timer 1 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR1I,Q_Time
     BC      INTSTA,TMR1I
     BTG     PORTA,7
Q_Time:
     POP
     RETI
;  ===  Timer1 = 32.768K / [256 x (3Fh + 1)]
Timer1SR:
       :
     PA.7 setting output pin
       :
     MOV     A,#10000011B
     MOV     TR1CON,A              ; Fosc a Pre-scale 1:256 & wakeup
     MOV     A,#03FH
     MOV     TRL1,A                ; 0.5sec=[256x(63+1)]/32.768K
     BS      TR1CON,T1EN           ; Timer 1 enable.
     BS      TR1CON,TMR1IE         ; Timer 1 interrupt enable.
     BC      INTSTA,TMR1I          ; Clear timer1 interrupt status.
     BS      CPUCON,GLINT          ; Enable global interrupt.
     BS      CPUCON,MS1            ; Idle mode.
T1Wloop:
     SLEP
     NOP
       :
     SJMP    T1Wloop
```

8.3 Timer 2 (8 Bits)

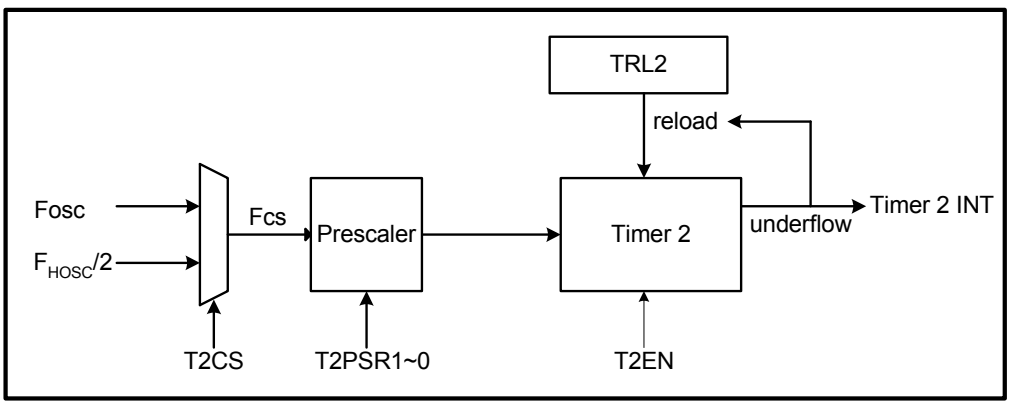


Figure 8-3 Timer 2 Function Block Diagram

Timer 2 is a general-purpose 8-bit down counter used on applications that require a time counter with interrupt. The clock source (FCS) may be selected from the oscillator clock (FOSC) or half of the system clock (FHOSC/2).

A prescaler for the timer is also available. The T2PSR1 ~ T2PSR0 bits of TR2WCON register determine the prescaler ratio and generate different clock rates as clock source for the timer.

Counter value is decreased by one (counting down) according to the timer clock source frequency. When counter value underflows, the timer interrupt is triggered (if Timer 2 interrupt is enabled).

$$T = \frac{1}{F_{CS}} \times \Pr escaler \times (TRL2 + 1)$$

8.3.1 Timer 2 Registers

- **TRL2 (R2Dh): Timer 2 Reload Register**

This register is used to store the auto-reload value of Timer 2. When Timer 2 is enabled or underflow occurs, TRL2 register will automatically reload into 8 bits counter.

- **TR2WCON (R2Ch): Timer 2/Watchdog Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTEN	WDTPSR1	WDTPSR0	**TMR2IE**	**T2EN**	**T2CS**	**T2PSR1**	**T2PSR0**

Bit 4 (TMR2IE): Control bit of Timer 2 interrupt

0: Disable interrupt function

1: Enable interrupt function

Bit 3 (T2EN): Timer 2 enable control bit

0: Disable Timer 2 (stop counting)

1: Enable Timer 2

Bit 2 (T2CS): Timer 2 clock source select bit

0: Clock source is from Fosc

1: Clock source is from Fhosc/2

Bit 1 ~ Bit 0 (T2PSR1 ~ T2PSR0): Timer 2 prescaler select bits

T2PSR1 : T2PSR0	Prescaler Value
00	1:1
01	1:2
10	1:4
11	1:8

■ **CPUCON (R20h): MCU Control Register**

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

Bit 2 (GLINT): Global interrupt enable/disable bit

0: Disable all interrupt

1: Enable all un-mask interrupts

■ **INTSTA (R24h): Timer Interrupt Status Register**

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

Bit 2 (TMR2I): When Timer 2 interrupt occurs, this bit will be set. Clear to **"0"** by software.

■ **Code Example:**

```
; === Timer 2 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR2I,Q_Time
     BC      INTSTA,TMR2I
     BTG     PORTA,7
Q_Time:
     POP
     RETI
; === Timer 2 = (1/32.768K) X [4 x (FFh + 1)]
Timer 2SR:
       :
     PA.7 setting output pin
       :
     MOV     A,#00000010B
     MOV     TR2WCON,A           ; Fosc and Pre-scale=1:4
     MOV     A,#0XFF
     MOV     TRL2,A              ; 31.25ms=[4x(255+1)]/32768
     BS      TR2WCON,T2EN        ; Timer 2 Enable
     BS      TR2WCON,TMR2IE      ; Timer 2 interrupt Enable
     BC      INTSTA,TMR2I        ; Clear Timer 2 interrupt Status
TMR2Loop:
     SJMP    TMR2Loop
```

8.4 Watchdog Timer (WDT)

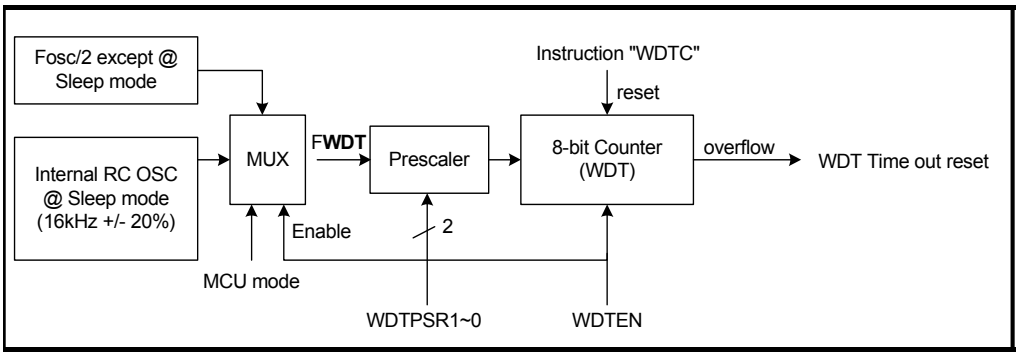


Figure 8-4 Watchdog Timer Functional Block Diagram

The watchdog timer (WDT) clock source comes from an on-chip RC oscillator (16kHz ± 20%, MCU in Sleep mode) or FOSC/2 (MCU in Fast, Slow, or Idle mode). The WDT will keep on running even after the oscillator has been turned off.

The WDTEN bit controls the WDT's enable/disable functions. The initial state of WDT is disabled. When WDT is enabled, its time-out will cause the MCU to reset. You should use "WDTC" instruction to clear the WDT value before WDT time-out. A prescaler is provided to generate different clock rates for the WDT clock source. The prescaler ratio is defined by WDTPSR1 and WDTPSR0.

The WDT time out range is 64ms (prescaler=1:4) to 2.048 second (prescaler=1:128).

$$T = \frac{1}{F_{WDT}} \times \Pr escaler \times (WDT + 1)$$

8.4.1 Watchdog Timer (WDT) Registers

- **TR2WCON (R2Ch): Timer 2/Watchdog Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTEN	**WDTPSR1**	**WDTPSR0**	TMR2IE	T2EN	T2CS	T2PSR1	T2PSR0

Bit 7 (WDTEN): Watchdog Timer enable bit

0: Disable Watchdog Timer (stop running)

1: Enable Watchdog Timer

Bit 6 ~ Bit 5 (WDTPSR1 ~ WDTPSR0): Watchdog Timer prescaler select bits

WDTPSR1: WDTPSR0	Prescaler Value
00	1:4
01	1:16
10	1:64
11	1:128

■ **Code Example:**

```
; === WDT setting 2.048sec
       :
     Timer 1 (0.5sec wakeup)
       :
     BS      TR2WCON,WDTPSR1
     BS      TR2WCON,WDTPSR0              ; Pre-scale 1:128
     BS      TR2WCON,WDTEN                ; WDT enable
     BC      CPUCON,MS1                   ; Change to sleep mode
     WDTC
     SLEP
WDT_Loop:
     SJMP    WDT_Loop
```

8.5 Input/Output Key

■ **Key Matrix:**

8 pins key input (Port A.7 ~ 0) and 12 pins key strobe (Port B.7 ~ 0 and Port C.3 ~ 0) can achieve a maximum of 96 keys matrix.

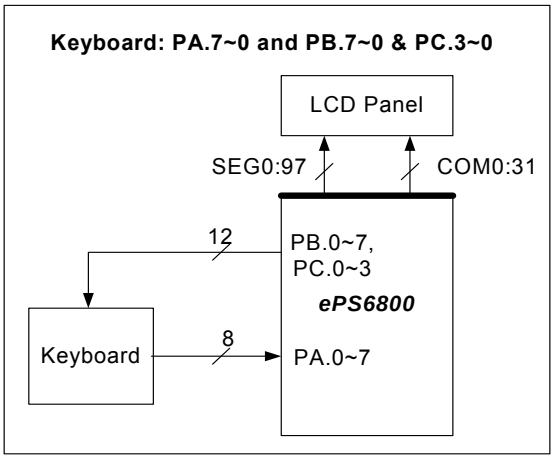


Figure 8-5 Key Function Block Diagram

8.5.1 Key Functions

KE	R1EN	PA6PU~PA0PU	Total Pull-up Resistor	Port A.6 ~ 0	Note
0	×	×	Off	High-Z	
1	0	0	**Off**	**Floating**	**Prohibited**
	0	1	R2	PA.0~.6	
	1	0	R1	PA.0~.6	
	1	1	R1 // R2 [1]	PA.0~.6	

×: Don't care

[1] R1 // R2 = R1xR2 / (R1+R2)

8.5.2 Key Strobe (PA.7 ~ 0 and PB.7 ~ 0 and PC.3 ~ 0)

The following are ways to output a strobe signal:

- **Waiting Key Scan Mode**

  When in Waiting Key Scan mode, Bit 7 ~ 0 of PAINT or PAWAKE must be enabled, and PB.7 ~ 0 and PC.3 ~ 0 must be output Low.

  In Key Scan mode, if any falling edge of the PA.0 ~ 7 pins is detected (when PAINTEN=1), wake-up will occur. Then the CPU runs and interrupt is triggered (if enabled).

- **Software Key Scan Mode**

  Software key scan is used to determine "which key is pressed". Refer to the table below in determining which Port B bit must be output pins with low voltage.

Key Input						Key Output									
KE	R1EN	PA7PU ~ PA0PU	Total Pull-up Resistor	PA.6 ~ 0	PA.7	PBCON	DCRB	Port B							
								Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7
0	×	×	Floating	High-Z	Floating	0×FF	0×FF	1	1	1	1	1	1	1	1
1	0	1	R2	PA.0~6	PA.7	0xFF	0×FE	0	1	1	1	1	1	1	1
							0×FD	1	0	1	1	1	1	1	1
							0×FB	1	1	0	1	1	1	1	1
							0×F7	1	1	1	0	1	1	1	1
							0×EF	1	1	1	1	0	1	1	1
							0×DF	1	1	1	1	1	0	1	1
							0×BF	1	1	1	1	1	1	0	1
							0×7F	1	1	1	1	1	1	1	0

×: Don't care

Software key scan is used to determine "which key is pressed". Refer to the table below in determining which Port C bit must be output pins with low voltage.

Key Intput						Key Output					
KE	R1EN	PA7PU ~ PA0PU	Total Pull-up Resistor	PA.6 ~ 0	PA.7	PCCON	DCRC	Port C			
								Bit0	Bit1	Bit2	Bit3
0	×	×	Floating	High-Z	Floating	0×0F	0×0F	1	1	1	1
1	0	1	R2	PA.0~6	PA.7	0x0F	0x0E	0	1	1	1
							0x0D	1	0	1	1
							0x0B	1	1	0	1
							0x07	1	1	1	0

×: Don't care

8.5.3 Input/Output Key Registers

■ **DCRA (R33): Port A Direction Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7DC	PA6DC	PA5DC	PA4DC	PA3DC	PA2DC	PA1DC	PA0DC

Bit 7 ~ Bit 0: Direction control of Port A.0~7

0: output pin

1: input pin

■ **Port A (R31h): Port A Register**

Bit 6 ~ Bit 0: Port A input is selected by PA6DC~PA0DC bits of the DCRA register (above). The input structure and 2-stage pull-up resistor are controlled together by PA6PU ~ PA0PU bits of PACON register and R1EN, KE bits of the STBCON register (see below).

Bit 7: Port A input is selected by PA7DC bit of the DCRA register (above). The pull-up resistor is controlled by PA7PU bit of the PACON register (see below).

■ **PACON (R32h): Port A Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7PU	PA6PU	PA5PU	PA4PU	PA3PU	PA2PU	PA1PU	PA0PU

Bit 7 ~ Bit 0 (PA7PU ~ PA0PU): Pull-up resistor (R2 large resistor) control bits

0: Disable PortA.0 ~ PortA.7 pull-up resistor

1: Enable PortA.0 ~ PortA.7 pull-up resistor

■ **PAINTEN (R35h): Port A Interrupt Enable 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 ~ Bit 0 (PA7IE ~ PA0IE): Interrupt control bit

0: Disable Interrupt function

1: Enable Interrupt function

NOTE

This function is only available with Port A selected as input pin.

■ **PAINTSTA (R36h): Port A 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 ~ Bit 0 (PA7I ~ PA0I): INT status of Port A interrupt

Set to **"1"** when a pin falling edge is detected.

Clear to **"0"** by software.

■ **LCDCON (R2Eh): LCD Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R1EN	BLANK	LCDON	SFR2	SFR1	SFR0	LCR1	LCR0

Bit 7 (R1EN): R1 pull-up resistor (small resistor) control bit for Port A.6 ~ Port A.0.

0: Disable R1 pull-up resistor

1: Enable R1 pull-up resistor

■ **STBCON (R30h): Strobe Output Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
KE	SCAN	BitST	ALL	STB3	STB2	STB1	STB0

Bit 7 (KE): Port A input enable/disable control bit (control at Port A.6 ~ 0)

0: Disable Port A input function

1: Enable Port A input function

■ **Port B (R37h): Port B Register**

Bit 7 ~ Bit 0: Port B is output that is selected by PB7DC ~ PB0DC bits of DCRB register (see DCRB below). The pull-up resistor is controlled by PB7PU ~ PB0PU bits of PBCON register (see below).

■ **PBCON (R38h): Port B Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PB7PU	PB6PU	PB5PU	PB4PU	PB3PU	PB2PU	PB1PU	PB0PU

Bit 7 ~ Bit 0 (PB7PU ~ PB0PU): Port B.0 ~ Port B.7 pull-up resistor control bits

0: Disable pull-up resistor

1: Enable pull-up resistor

■ **DCRB (R39h): Port B Direction Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PB7DC	PB6DC	PB5DC	PB4DC	PB3DC	PB2DC	PB1DC	PB0DC

Bit 7 ~ Bit 0: Direction control of Port B.0~7

0: output pin

1: input pin

■ **Port C (R3Ah): Port C Register**

Bit 3 ~ Bit 0: Port C output is selected by PC3DC ~ PC0DC bits of DCRC register (see DCRC below). The pull-up resistor is controlled by PC3PU ~ PC0PU bits of the PCCON register (see below).

■ **PCCON (R3Bh): Port C Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	PC3PU	PC2PU	PC1PU	PC0PU

Bit 3 ~ Bit 0 (PC7PU ~ PC0PU): Port C.0 ~ Port C.3 pull-up resistor control bits

0: Disable pull-up resistor

1: Enable pull-up resistor

■ **DCRC (R3Ch): Port C Direction Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	PC3DC	PC2DC	PC1DC	PC0DC

Bit 3 ~ Bit 0: Direction control of Port C.0~3

0: output pin

1: input pin

■ **Code Example:**

```
; Key Matrix (Port A.0~7 and Port B.0~7):
; *** Key Scan function
      :
    LCD display setting
      :
    CLR    PORTB
    CLR    DCRB            ; Set PB output pins
    MOV    A,#0XFF
    MOV    PBCON,A         ; pull-up enable
    MOV    PACON,A         ; R2EN enable
    BS     STBCON,KE
    MOV    PAWAKE,A        ; Port A setting wake-up function
      :
; === Idle mode waiting key scan routine
    BS     CPUCON,MS1      ; Idle mode
KeyIdle:
    SLEP
    NOP
    MOV    A,PORTA      ; Port A input data
    JE     A,#0XFF,KeyIdle
; === Key scan routine
Polling_key_scan:
    MOV    A,#0XFE
    MOV    DCRB,A
KeyPolling:
    LCALL  DLY10US
    MOV    A,PORTA
    JLE    A,#0XFE, KeyScan
    BS     STATUS,F_C
    RLC    DCRB
    JBS    STATUS,F_C,KeyPolling
      :
KeyScan:
; --- Clear key number
    CLR    Key_No
    BC     STATUS,F_C
; --- Key Scan is finish
KeyScan_PA:
    INC    Key_No
    RRC    ACC
    JBC    STATUS,F_C, KeyScanOk
    SJMP   KeyScan_PA
KeyScanOk:
    MOV    A,#0XFF
    XOR    A,PORTB
    ADD    Key_No,A        ; Key_No: XXXX XXXX
      :
```

8.6 LCD Driver

The ePS6800 provides direct drive LCD. It supports multiplexed drive for 98 SEGs × 32 COMs which allows the use of pads as an LCD driver pin or as key input port. The available LCD RAM corresponds directly with LCD Pixel.

This embedded LCD driver generates waveforms to drive the display.

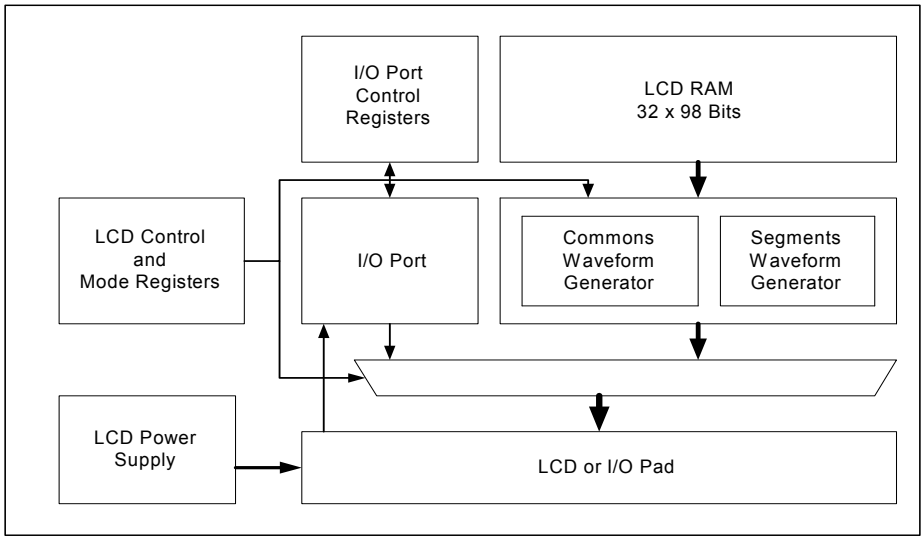


Figure 8-6 LCD Driver Function Block Diagram

The following is the LCD pin configuration:

SEG0 / Key Strobe0 | SEG15 / Key Strobe15
SEG16 | SEG97
COM31/ Port E.7 | COM24/ Port E.0
COM23/ Port D.7 | COM16/ Port D.0
COM15 | COM0

Figure 8-7 LCD Pin Configuration

8.6.1 LCD Driver Registers

■ **LCDCON (R2Eh): LCD Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R1EN	BLANK	LCDON	SFR2	SFR1	SFR0	LCR1	LCR0

Bit 6 (BLANK): LCD Blanking control bit

0: Disable

1: Enable (All SEG pins output "0" signal)

Bit 5 (LCDON): LCD display control bit

0: LCD display off

1: LCD display on

NOTE

All COM and SEG pins are tied to ground when LCD display is off.

Bit 4 ~ Bit 2 (SFR2 ~ SFR0): Frame Frequency Adjustment

CL Frequency vs. Duty Ratio Table:

			CL Frequency				
SFR2	SFR1	SFR0	1/16 Duty	1/20 Duty	1/24 Duty	1/28 Duty	1/32 Duty
0	0	0	Fosc / 26	Fosc / 21	Fosc / 17	Fosc / 15	Fosc / 13
0	0	1	Fosc / 28	Fosc / 22	Fosc / 19	Fosc / 16	Fosc / 14
0	1	0	Fosc / 30	Fosc / 24	Fosc / 20	Fosc / 17	Fosc / 15
0	1	1	Fosc / 32	Fosc / 26	Fosc / 21	Fosc / 18	Fosc / 16
1	0	0	Fosc / 34	Fosc / 27	Fosc / 23	Fosc / 19	Fosc / 17
1	0	1	Fosc / 36	Fosc / 29	Fosc / 24	Fosc / 21	Fosc / 18
1	1	0	Fosc / 38	Fosc / 30	Fosc / 25	Fosc / 22	Fosc / 19
1	1	1	Fosc / 40	Fosc / 32	Fosc / 27	Fosc / 23	Fosc / 20

Bit 1, Bit 0 (LCR1, LCR0): LCD Bias Voltage Charge-pump Rate select bits

LCR1 : LCR0	Charge-Pump Rate (Hz)
00	OFF (LBVON=0, LCD contrast adjustment is off)
01	8K
10	16K
11	32K

■ **LCDARL (R22h): LCD RAM Column Address Register (see LCD RAM Map)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LCDARL7	LCDARL6	LCDARL5	LCDARL4	LCDARL3	LCDARL2	LCDARL1	LCDARL0

■ **LCDARH (R23h): LCD Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADJ3	**ADJ2**	**ADJ1**	**ADJ0**	**-**	**-**	**LCDARH1**	**LCDARH0**

Bit 7 ~ Bit 4 (ADJ0 ~ ADJ3): LCD contrast adjustment

ADJ3	ADJ2	ADJ1	ADJ0	α	Contrast
0	0	0	0	0	Low
0	0	0	1	1	
:	:	:	:	:	:
:	:	:	:	:	:
1	1	1	0	14	
1	1	1	1	15	High

$$VEV = (1 - \frac{(15-\alpha)}{75}) \times Vref$$ (Vref= 1.3V at 25°C)

α= 15 , Vev = 1.3V ; α= 0, Vev = 0.8 × 1.3V = 1.04V

Bit 1, Bit 0 (LCDARH1, LCDARH0): Page address for LCD RAM

NOTE

The Register R2Fh must be fixed to 0x00.

■ **LCDDATA (R0Eh):** LCDDATA register is an Indirect Address Pointer of LCD RAM.

Any instruction that uses LCDDATA as register, actually accesses LCD RAM via the address pointed by LCDARL (see figure below).

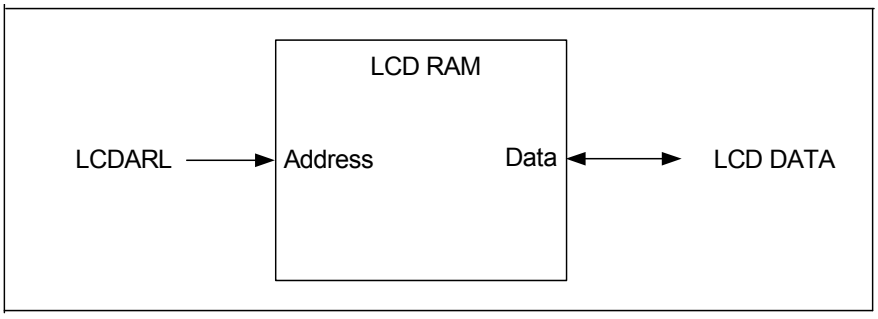


Figure 8-8 LCD Data Register Access through LCD RAM

■ **POST_ID (R21h): Post Increase / Decrease Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR2_ID	**LCD_ID**	FSR1_ID	FSR0_ID	FSR2PE	**LCDPE**	FSR1PE	FSR0PE

After accessing (read or write) the LCD RAM, the LCDARL register can be automatically increased or decreased by setting the POST_ID register.

Bit 6 (LCD_ID): Set to **"1"** to auto-increase the LCDARL register

Reset to **"0"** to auto-decrease the LCDARL register

Bit 2 (LCDPE): Enable LCDARL post increase/decrease function

■ **Code Example:**

```
; === LCD Setting
L_Initial:
; --- LCD Off, Normal Display Mode, Charge-Pump rate=16K
     MOV    A,#00011110B
     MOV    LCDCON,A
     MOV    A,#11110000B
     MOV    LCDARH,A
     CLR    0x2F
     SCALL  DspRAMdot
; --- LCD turn-on
     BS     LCDCON,LCDON
     LCALL  Delay1sec
       :
DspLoop:
; --- LCD Blanking
     BS     LCDCON,BLANK
     LCALL  Delay1sec
; --- Normal display
     BC     LCDCON,BLANK
     LCALL  Delay1sec
       :
     SJMP   DspLoop
```

```
; *** Display LCD RAM is data 55 & AA
DspRAMdot:
; --- LCD increase enable
     BS      POST_ID,LCDPE
     BS      POST_ID,LCD_ID
     CLR     LCDARL
     LCALL   DspRAMd1
     BS      LCDARH,LCDARH0
     LCALL   DspRAMd1
     BS      LCDARH,LCDARH1
     LCALL   DspRAMd1
     BC      LCDARH,LCDARH0
DspRAMd1:
     MOV     A,#0x31
     MOV     LCD_Times,A
; === Write LCD RAM is dot matrix
WrLRAMd:
     MOV     A,#0XAA
     MOV     LCDDATA,A
     MOV     A,#0X55
     MOV     LCDDATA,A
     JDNZ    LCD_Times,WrLRAMd
     CLR     LCDARL
     RET
```

8.6.2 LCD RAM Map

LCD RAM		LCDARH[1:0]			
RAM Address LCDARL		11 (Page 3)	10 (Page 2)	01 (Page 1)	00 (Psge 0)
		COM31-COM24	COM23-COM16	COM15-COM8	COM7-COM0
		Bit 7 - Bit 0	Bit 7 - Bit 0	Bit 7 - Bit 0	Bit 7 - Bit 0
SEG0	**00H**				
:	**:**				
SEG97	**61H**				

8.6.3 LCD Driving Method Circuit

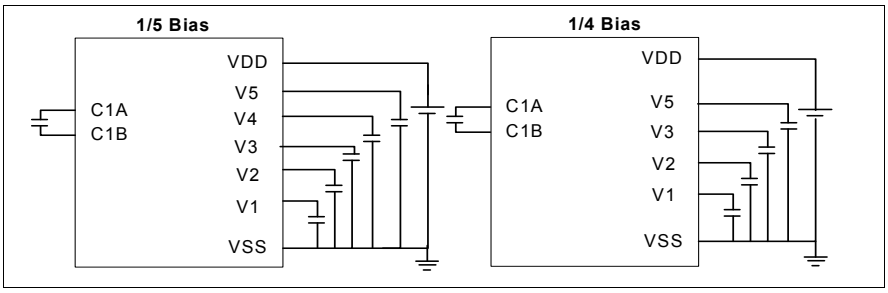


Figure 8-9 LCD Driving Method Circuit

8.6.4 LCD COM Waveforms

8.6.4.1 1/32 Duty and 1/4 Bias

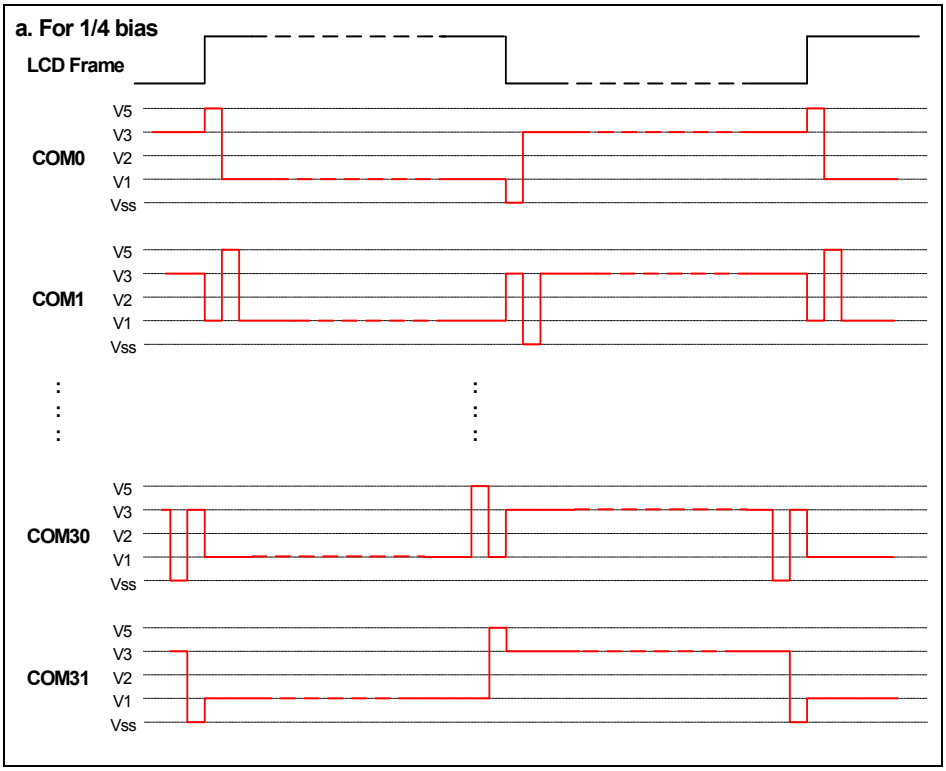


Figure 8-10 LCD COM Waveform for 1/32 Duty and 1/4 Bias

8.6.4.2 1/32 Duty and 1/5 Bias

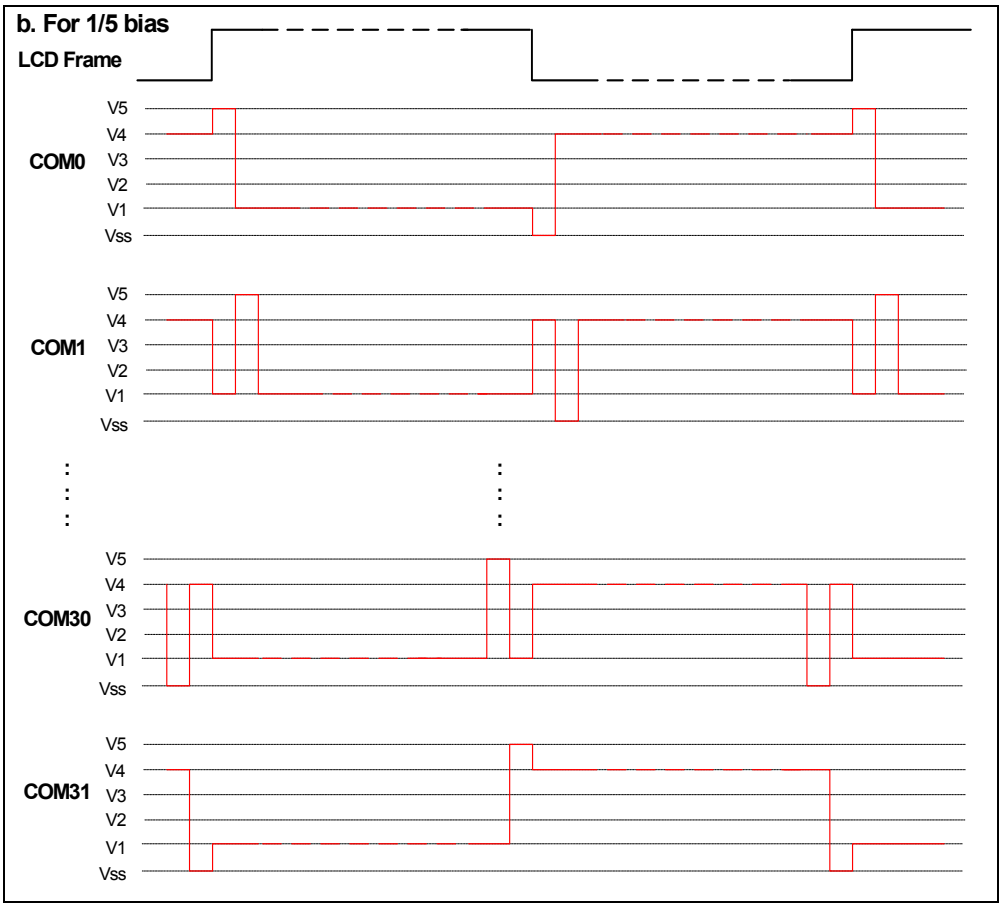


Figure 8-11 LCD COM Waveform for 1/32 Duty and 1/5 Bias

8.6.5 LCD COM and SEG Waveforms

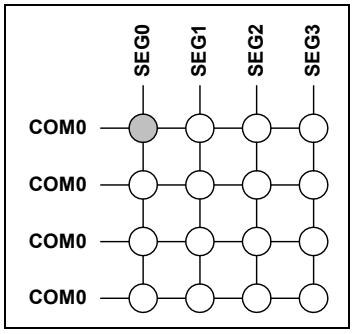


Figure 8-12 LCD COM and SEG Waveform Matrix

8.6.5.1 1/32 Duty and 1/4 Bias

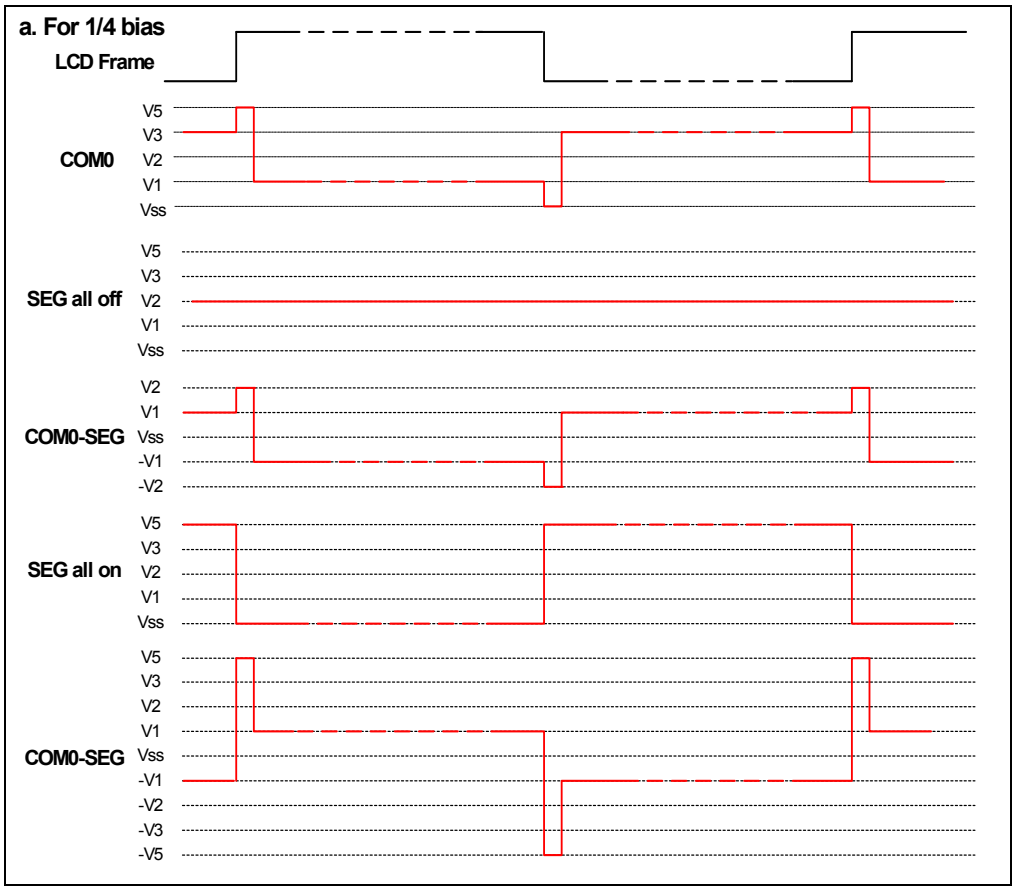


*Figure 8-13 LCD COM **and** SEG Waveform for 1/32 Duty and 1/4 Bias*

8.6.5.2 1/32 Duty and 1/5 Bias

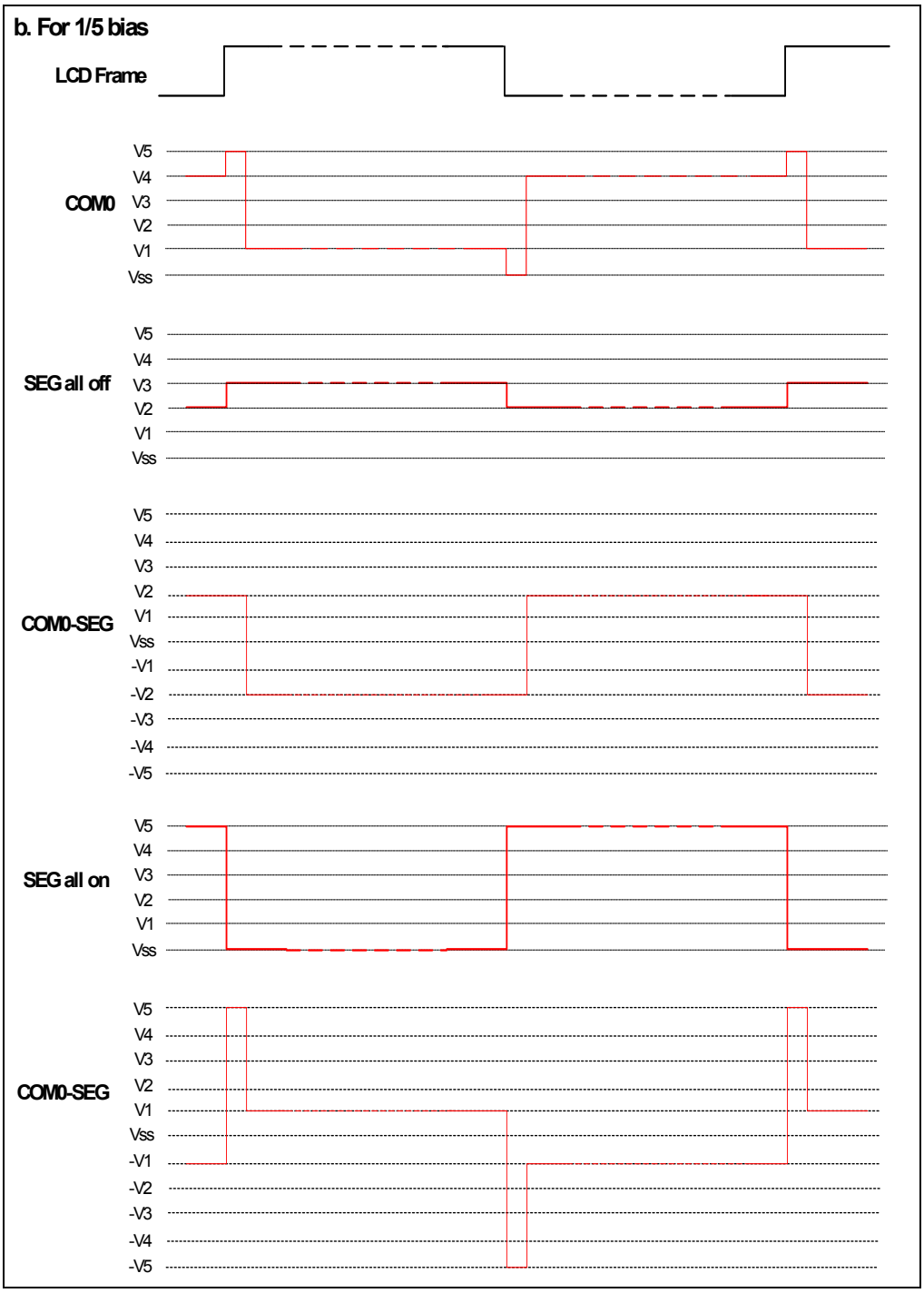


Figure 8-14 LCD COM and SEG Waveform for 1/32 Duty and 1/5 Bias

9 Electrical Characteristics

9.1 VDD = 1.5V Electrical Characteristics

■ Absolute Maximum Ratings

Items	Sym.	Condition	Limits	Unit
Supply voltage	VDD	-	-0.3 to +2.2	V
Supply voltage for ICE	VCC	-	-0.3 to +3.6	
Input voltage (general input port)	VIN	-	-0.5 to VDD +0.5	V
Input voltage for ICE	VIN1	-	-0.5 to VCC +0.5	V
Operating temperature range	TOPR	-	-10 to +70	°C
Storage temperature range	TSTR	-	-55 to +125	°C

■ Recommended Operating Conditions

Items	Sym.	Condition	Limits	Unit
Supply voltage	VDD	-	1.2 to 1.8	V
Supply voltage for ICE	VCC	-	2.2 to 3.6	
Input voltage	VIH	-	VDD x 0.9 to VDD	V
	VIL	-	0 to VDD x 0.1	V
Input voltage for ICE	VIH	-	VDD x 0.9 to VDD	V
	VIL	-	0 to VDD x 0.1	V
Operating temperature	TOPR	-	-10 to +70	°C

■ DC Electrical Characteristics (Condition: Ta=25°C, VDD = 1.5V)

Parameter	Sym.	Condition		Min.	Typ.	Max.	Unit
Clock	F_{HOSC}	Main-clock frequency	RC OSC, R=1.5 MΩ	140	200	260	kHz
			RC OSC, R=1 MΩ	210	300	390	
			RC OSC, R=560 KΩ	350	500	650	
			RC OSC, R=270 KΩ	700	1000	1300	
	Fosc	Sub-clock frequency	RC OSC, R=2.2 MΩ	26.24	32.8	39.36	kHz
			Internal RC OSC	26.24	32.8	39.36	
			Crystal OSC	-	32.768	-	
Supply Current	Idd1	Sleep mode	VDD=1.5V, no load	-	-	1	μA
	Idd2	Idle mode	VDD=1.5V, RC/Crystal OSC, LCD disabled	-	8	15	
	Idd3		VDD=1.5V, RC/Crystal OSC, LCD enabled, no load	-	10	16	
	Idd4	Slow mode	VDD=1.5V, RC/Crystal OSC, LCD disabled	-	10	16	
	Idd5		VDD=1.5V, RC/Crystal OSC, LCD enabled, no load	-	13	20	
	Idd6	Fast mode	VDD=1.5V, FHOSC =200kHz, LCD enabled, no load	-	30	40	
	Idd7		VDD=1.5V, FHOSC =300kHz, LCD enabled, no load	-	40	55	
	Idd8		VDD=1.5V, FHOSC =500kHz, LCD enabled, no load	-	60	85	
	Idd9		VDD=1.5V, FHOSC =1MHz, LCD enabled, no load	-	95	120	

Parameter	Sym.	Condition		Min.	Typ.	Max.	Unit
Input Voltage	VIH1	PA[0:7] , PB[0:7], PC[0:3] (as general input port)		VDD×0.7	-	VDD	V
	VIL1			0	-	VDD×0.3	
Input Threshold Voltage (Schmitt)	VT+	RSTB		0.5×VDD	-	0.75×VDD	V
	VT–			0.2×VDD	-	0.4×VDD	
Large Pull-up Resistance	RPU5	RSTB	Vin=GND	200	450	900	KΩ
Small Pull-up Resistance	RPU6	RSTB	Vin=1V	10	30	60	KΩ
Large Pull-down Resistance	RPD1	TEST	Vin=VDD	250	500	900	KΩ
Small Pull-down Resistance	RPD2	TEST	Vin=0.5V	3	6	12	KΩ
Output Current	IOH1	PA[0:7], PB[0:7], PC[0:3] (as general output port)	VDD=1.5V, VOH=1.2V	-0.6	-0.9	-1.5	mA
	IOL1		VDD=1.5V, VOL=0.2V	0.6	0.9	1.5	
Input leakage current	IIL	All Input port (without pull up/down resistor) Vin= VDD or GND		-	-	±1	μA
Large Pull-up Resistance	RPU1	PA[0:7]	Key high resistance, pulled-up by R2, Vin2=0.5V	80	150	300	KΩ
	RPU3	PB[0:7], PC[0:3]	Vin=0.5V	60	180	400	
Small Pull-up Resistance	RPU2	PA[0:6]	Key high resistance, pulled-up by R2//R1, Vin2=0 V	15	30	60	KΩ
	RPU5	PA[7]	Vin=1V	15	30	60	
Data Retention Voltage	Vret	-		-	1.2	-	V
Power-on reset voltage	Vpor	-		0.9	1.0	1.1	V
LCD Driver							
Regulator Voltage	V1	ADJ=8		V1-2%	V1	V1+2%	V
V5 Voltage	V5	-		-	6	6.5	V
Temperature variation for V5	V5	T=0°C ~40°C		-5	-	+5	mV / °C
Strobe Output ON-resistance	ROP	Seg[0:15] (as key strobe)	V=VDD-0.2V	80	200	1200	KΩ
	RON		V=0.2V	0.7	1.3	2	

10 Pin Type Circuit Diagrams

- **Reset Pin Type**

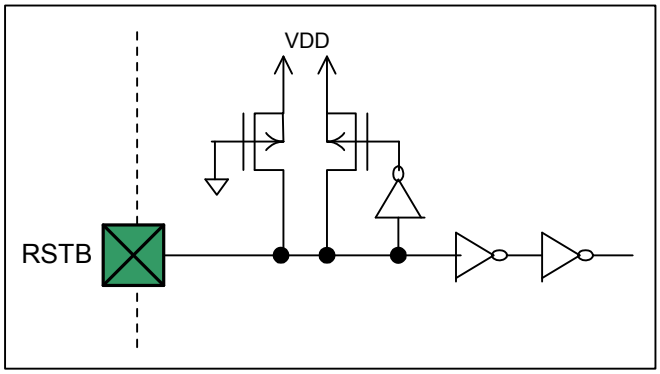


Figure 10-1 Reset Pin Type Circuit Diagram

- **TEST Pin Type**

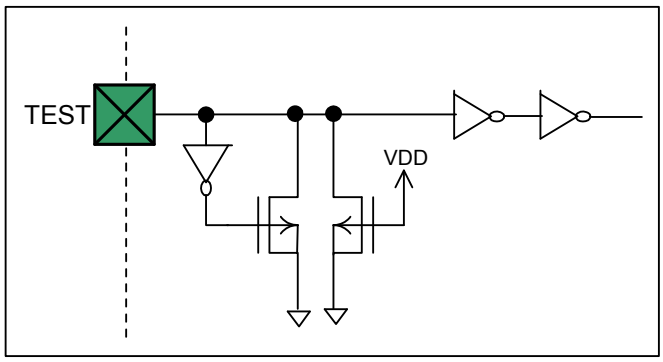


Figure 10-2 Test Pin Type Circuit Diagram

- **Oscillator Pin Type**

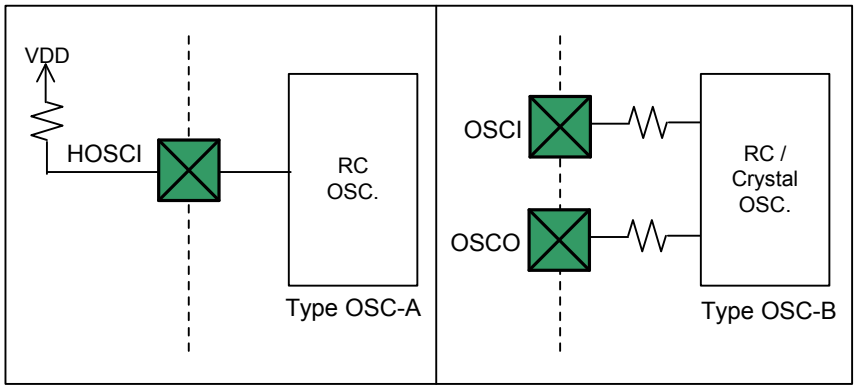


Figure 10-3 Oscillator Pin Type Circuit Diagram

■ **I/O Pin Type (PA.0~6)**

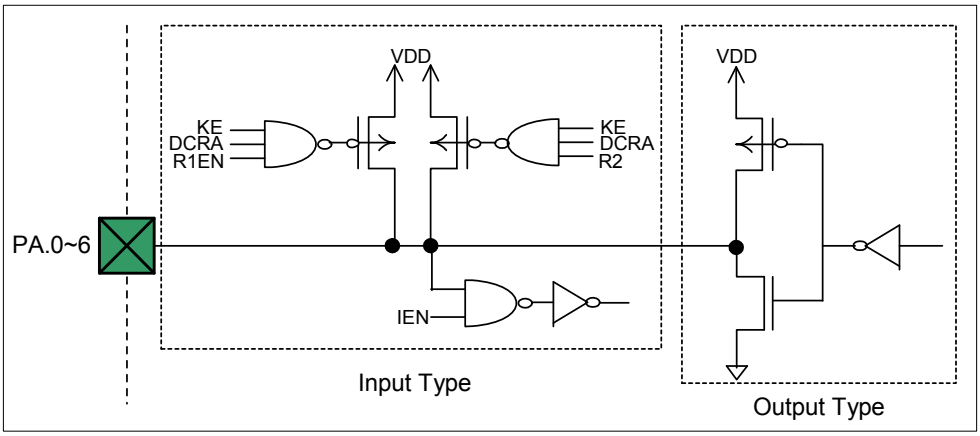


Figure 10-4 I/O Pin Type Circuit Diagram

■ **I/O Pin Type (PA.7)**

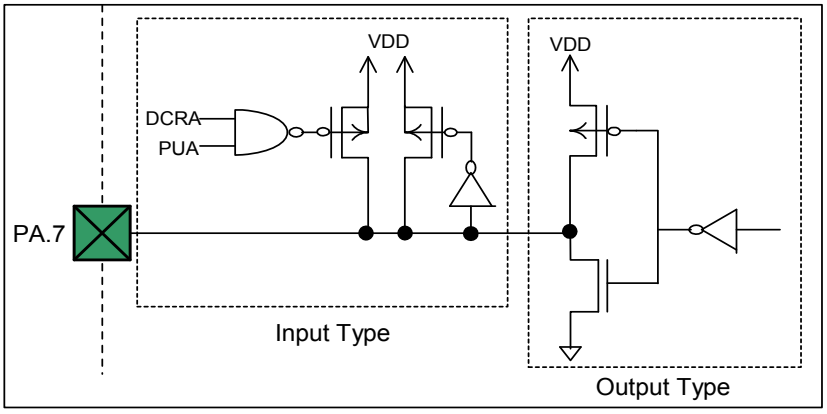


Figure 10-5 I/O Pin Type Circuit Diagram

■ **General SEG and COM Pin Type**

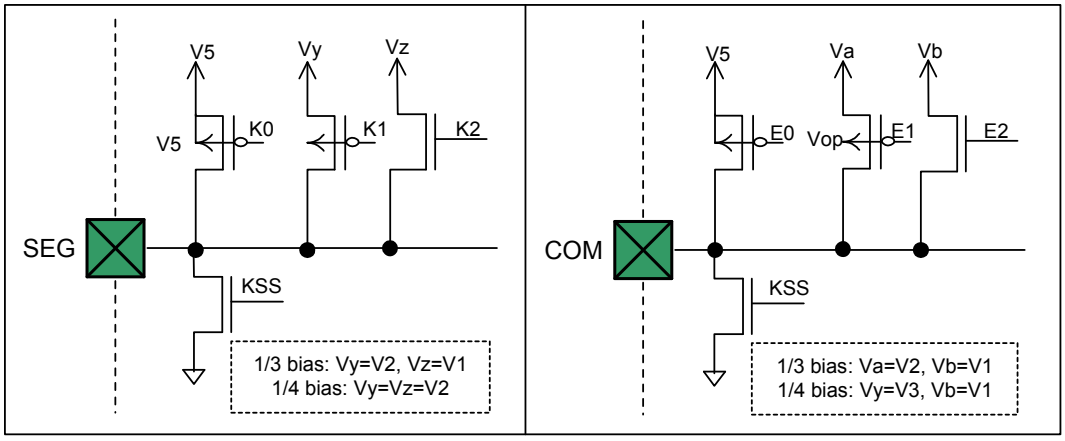


Figure 10-6 General SEG and COM Share Pin Type Circuit Diagram

11 Application Circuits

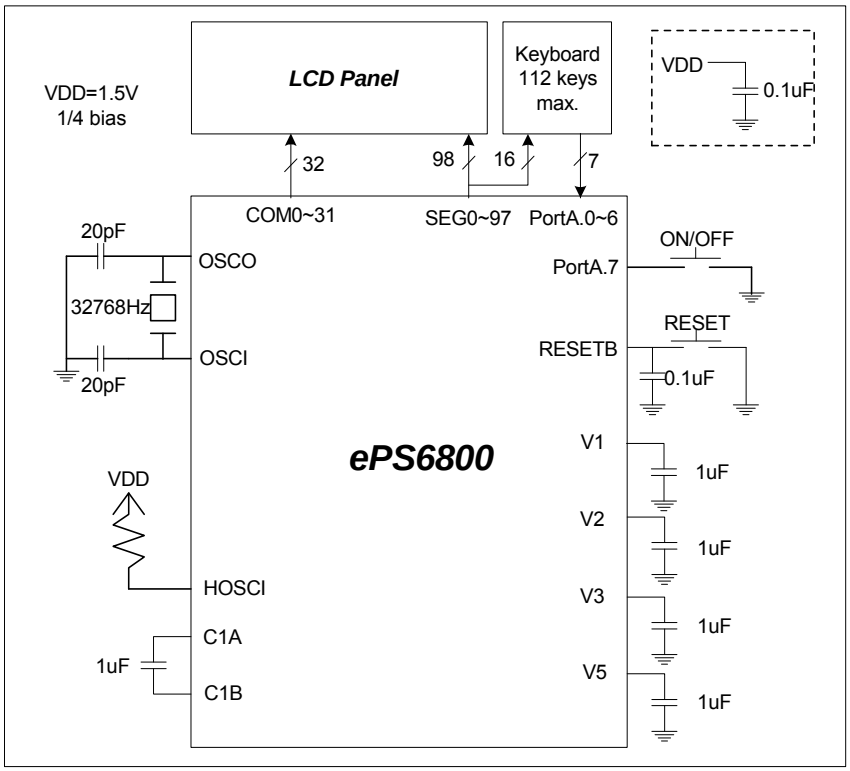


Figure 11-1 Application Circuit Diagram for 1/4 Bias

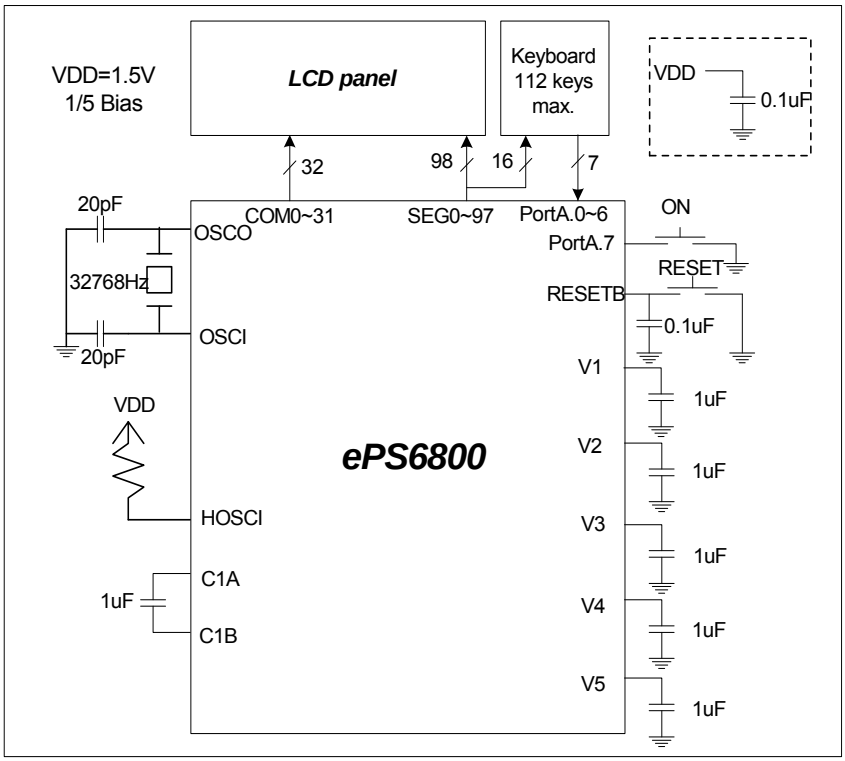


Figure 11-2 Application Circuit Diagram for 1/5 Bias

12 Instruction Set

Legend: **k:** constant **r:** File Register **b:** bit
i: Table pointer control **p:** special file register (0h~1Fh) **addr:** address

Type	Binary Instruction	Mnemonic	Operation	Affected Status	Cycles
System Control	0000 0000 0000 0000	NOP	No operation	None	1
	0000 0000 0000 0001	WDTC	WDT ← 0; /TO←1; /PD←1	None	1
	0000 0000 0000 0010	SLEP	Enter Idle Mode if MS1=1 Enter Sleep Mode if MS1=0	None	1
	0010 0111 rrrr rrrr	RPT r ("r" is the content of register r)	Single repeat (r) times on next instruction	None	1
	0100 0011 kkkk kkkk	BANK #k	BSR ← k	None	1
Subroutine	0011 aaaa aaaa aaaa	S0CALL addr	Top of Stack] ← PC+1 PC[11:0] ← addr PC[12:16] ← 00000 (*5)	None	1
	111a aaaa aaaa aaaa	SCALL addr	[Top of Stack] ← PC+1; PC[12:0] ← addr; PC[13:16] unchanged	None	1
	0000 0000 0011 000a aa aaaa aaaa aaaa	LCALL addr (2 words)	[Top of Stack] ← PC+1; PC ← addr	None	2
	0010 1011 1111 1110	RET	PC ← (Top of Stack)	None	1
	0010 1011 1111 1111	RETI	PC ← (Top of Stack); Enable Interrupt	None	1
Compare	0010 0101 rrrr rrrr	TEST r	Z ← 0 if r<>0;Z ← 1 if r=0	Z	1
Jump	110a aaaa aaaa aaaa	SJMP addr	PC ← addr PC[13..15] unchange	None	1
	0000 0000 0010 000a aaaa aaaa aaaa aaaa	LJMP addr (2 words)	PC ← addr	None	2
Compare and Jump	0101 0000 rrrr rrrr aaaa aaaa aaaa aaaa	JDNZ A,r,addr	A ← r-1, jump to addr if not zero PC[15:0] ← addr (*3)	None	2
	0101 0001 rrrr rrrr aaaa aaaa aaaa aaaa	JDNZ r,addr	r ← r-1, jump to addr if not zero PC[15:0] ← addr (*3)	None	2
	0100 0111 kkkk kkkk aaaa aaaa aaaa aaaa	JGE A,#k,addr	Jump to addr if A ≥ k PC[15:0] ← addr (*3)	None	2
	0100 1000 kkkk kkkk aaaa aaaa aaaa aaaa	JLE A,#k,addr	Jump to addr if A ≤ k PC[15:0] ← addr (*3)	None	2
	0100 1001 kkkk kkkk aaaa aaaa aaaa aaaa	JE A,#k,addr	Jump to addr if A=k PC[15:0] ← addr (*3)	None	2
	0101 0101 rrrr rrrr aaaa aaaa aaaa aaaa	JGE A,r,addr	Jump to addr if A ≥ r PC[15:0] ← addr (*3)	None	2
	0101 0110 rrrr rrrr aaaa aaaa aaaa aaaa	JLE A,r,addr	Jump to addr if A ≤ r PC[15:0] ← addr (*3)	None	2
	0101 0111 rrrr rrrr aaaa aaaa aaaa aaaa	JE A,r,addr	Jump to addr if A=r PC[15:0] ← addr (*3)	None	2

Type	Binary Instruction	Mnemonic	Operation	Affected Status	Cycles
Bit Compare and Jump	0101 1bbb rrrr rrrr aaaa aaaa aaaa aaaa	JBC r,b,addr	If r(b)=0,jump to addr PC[15:0] ← addr (*3)	None	2
	0110 0bbb rrrr rrrr aaaa aaaa aaaa aaaa	JBS r,b,addr	If r(b)=1,jump to addr PC[15:0] ← addr (*3)	None	2
Data Transfer	0010 0000 rrrr rrrr	MOV A,r	A ← r	Z	1
	0010 0001 rrrr rrrr	MOV r,A	r ← A	None	1
	100p pppp rrrr rrrr	MOVRP p,r	Register p ← Register r	None	1
	101p pppp rrrr rrrr	MOVPR r,p	Register r ← Register p	None	1
	0100 1110 kkkk kkkk	MOV A,#k	A ← k	None	1
	0010 0100 rrrr rrrr	CLR r	r ← 0	Z	1
ROM Look-Up Table	0100 0000 kkkk kkkk	TBPTL #k	TABPTRL ← k	None	1
	0100 0001 kkkk kkkk	TBPTM #k	TABPTRM ← k	None	1
	0100 0010 kkkk kkkk	TBPTH #k	TABPTRH ← k	None	1
	0010 11ii rrrr rrrr	TBRD i,r	r←ROM[(TABPTR)] (*1)(*2)	None	2
	0010 1111 rrrr rrrr	TBRD A,r	r←ROM[(TABPTR+ACC)] (*2)	None	2
Logic Operation	0000 0010 rrrr rrrr	OR A,r	A ← A .or. r	Z	1
	0000 0011 rrrr rrrr	OR r,A	r ← r .or. A	Z	1
	0100 0100 kkkk kkkk	OR A,#k	A ← A .or. k	Z	1
	0000 0100 rrrr rrrr	AND A,r	A ← A .and. r	Z	1
	0000 0101 rrrr rrrr	AND r,A	r ← r .and. A	Z	1
	0100 0101 kkkk kkkk	AND A,#k	A ← A .and. k	Z	1
	0000 0110 rrrr rrrr	XOR A,r	A ← A .xor. r	Z	1
	0000 0111 rrrr rrrr	XOR r,A	r ← r .xor. A	Z	1
	0100 0110 kkkk kkkk	XOR A,#k	A ← A .xor. k	Z	1
	0000 1000 rrrr rrrr	COMA r	A ← /r	Z	1
	0000 1001 rrrr rrrr	COM r	r ← /r	Z	1

Type	Binary Instruction	Mnemonic	Operation	Affected Status	Cycles
Arithmetic Operation	0001 1100 rrrr rrrr	INCA r	A ← r+1	C,Z	1
	0001 1101 rrrr rrrr	INC r	r ← r+1	C,Z	1
	0001 0000 rrrr rrrr	ADD A,r	A ← A+r	C,DC,Z,OV,SGE,SLE	1
	0001 0001 rrrr rrrr	ADD r,A	r ← r+A (*4)	C,DC,Z,OV,SGE,SLE	1
	0100 1010 kkkk kkkk	ADD A,#k	A ← A+k	C,DC,Z,OV,SGE,SLE	1
	0001 0010 rrrr rrrr	ADC A,r	A ← A+r+C	C,DC,Z,OV,SGE,SLE	1
	0001 0011 rrrr rrrr	ADC r,A	r ← r+A+C	C,DC,Z,OV,SGE,SLE	1
	0100 1011 kkkk kkkk	ADC A,#k	A ← A+k+C	C,DC,Z,OV,SGE,SLE	1
	0001 1110 rrrr rrrr	DECA r	A ← r-1	C,Z	1
	0001 1111 rrrr rrrr	DEC r	r← r-1	C,Z	1
	0001 0110 rrrr rrrr	SUB A,r	A ← r-A (*6)	C,DC,Z,OV,SGE,SLE	1
	0001 0111 rrrr rrrr	SUB r,A	r ← r-A (*6)	C,DC,Z,OV,SGE,SLE	1
	0100 1100 kkkk kkkk	SUB A,#k	A ← k-A (*6)	C,DC,Z,OV,SGE,SLE	1
	0001 1000 rrrr rrrr	SUBB A,r	A ← r-A-/C (*6)	C,DC,Z,OV,SGE,SLE	1
	0001 1001 rrrr rrrr	SUBB r,A	r ← r-A-/C (*6)	C,DC,Z,OV,SGE,SLE	1
	0100 1101 kkkk kkkk	SUBB A,#k	A ← k-A-/C (*6)	C,DC,Z,OV,SGE,SLE	1
	0001 0100 rrrr rrrr	ADDDC A,r	A ← (Decimal ADD) A+r+C	C,DC,Z	1
	0001 0101 rrrr rrrr	ADDDC r,A	r ← (Decimal ADD) r+A+C	C,DC,Z	1
	0001 1010 rrrr rrrr	SUBDB A,r	A ← (Decimal SUB) r-A-/C	C,DC,Z	1
	0001 1011 rrrr rrrr	SUBDB r,A	r ← (Decimal SUB) r-A-/C	C,DC,Z	1
Rotate	0000 1010 rrrr rrrr	RRCA r	A(n-1) ← r(n); C ← r(0); A(7) ← C	C	1
	0000 1011 rrrr rrrr	RRC r	r(n-1) ← r(n); C ← r(0); r(7) ← C	C	1
	0000 1100 rrrr rrrr	RLCA r	A(n+1) ← r(n); C ← r(7); A(0) ← C	C	1
	0000 1101 rrrr rrrr	RLC r	r(n+1) ← r(n); C ← r(7); r(0) ← C	C	1

Type	Binary Instruction	Mnemonic	Operation	Affected Status	Cycles
Shift	0010 0010 rrrr rrrr	SHRA r	A(n-1)←r(n); A(7)←C	None	1
	0010 0011 rrrr rrrr	SHLA r	A(n+1)←r(n); A(0)←C	None	1
Exchange	0101 0100 rrrr rrrr	EX r	r(7-0) ←→ A(7-0)	None	1
Bit Manipulation	0110 1bbb rrrr rrrr	BC r,b	r(b) ← 0	None	1
	0111 0bbb rrrr rrrr	BS r,b	r(b) ← 1	None	1
	0111 1bbb rrrr rrrr	BTG r,b	r(b) ← /r(b)	None	1
Nibble Operation	0101 0010 rrrr rrrr	EXL r	r(3-0) ←→ A(3-0)	None	1
	0101 0011 rrrr rrrr	EXH r	r(7-4) ←→ A(3-0)	None	1
	0010 0110 rrrr rrrr	MOVL r,A	r(3-0) ← A(3-0)	None	1
	0010 1000 rrrr rrrr	MOVH r,A	r(7-4) ← A(3-0)	None	1
	0010 1001 rrrr rrrr	MOVL A,r	A(3-0) ← r(3-0); A(7-4) ← 0	None	1
	0010 1010 rrrr rrrr	MOVH A,r	A(3-0) ← r(7-4); A(7-4) ← 0	None	1
	0000 0001 rrrr rrrr	SFR4 r	r(7-4)←A(3-0);r(3-0)←r(7-4); A(3-0)←r(3,0)	None	1
	0100 1111 rrrr rrrr	SFL4 r	r(3-0)←A(3-0);r(7-4)←r(3-0); A(3-0)←r(7-4)	None	1
	0000 1111 rrrr rrrr	SWAP r	r(0:3)←→r(4:7)	None	1
	0000 1110 rrrr rrrr	SWAPA r	r(0:3)→A(4:7);r(4:7)→A(0:3)	None	1

(*1) TBRD i, r:

r← ROM [(TABPTR)]

i=00: TABPTR no change

wi=01: TABPTR←TABPTR+1

i=10: TABPTR←TABPTR-1

(*2) TABPTR = (TABPTRH: TABPTRL)

Bit 0 = 0: Low byte of the pointed ROM data

Bit 0 = 1: High byte of the pointed ROM data

NOTE

- *Bit 0 of TABPTRL is used to select either low byte or high byte of the pointed ROM data.*
- *The maximum Look-up Table space is internal 128K bytes (64K words).*

(*3) The maximum jump range is 64K absolute address

(*4) Carry bit of “ADD PCL, A” or “ADD TABPTRL, A” will automatically carry into PCM or TABPTRM. The Instruction cycle of write to PC (program counter) takes two cycles.

(*5) S0CALL addressing ability is from 0x000 to 0xFFF (4K space)

(*6) When in SUB operation, borrow flag is indicated by the inverse of carry bit, that is B=/C

13 Pad Diagram

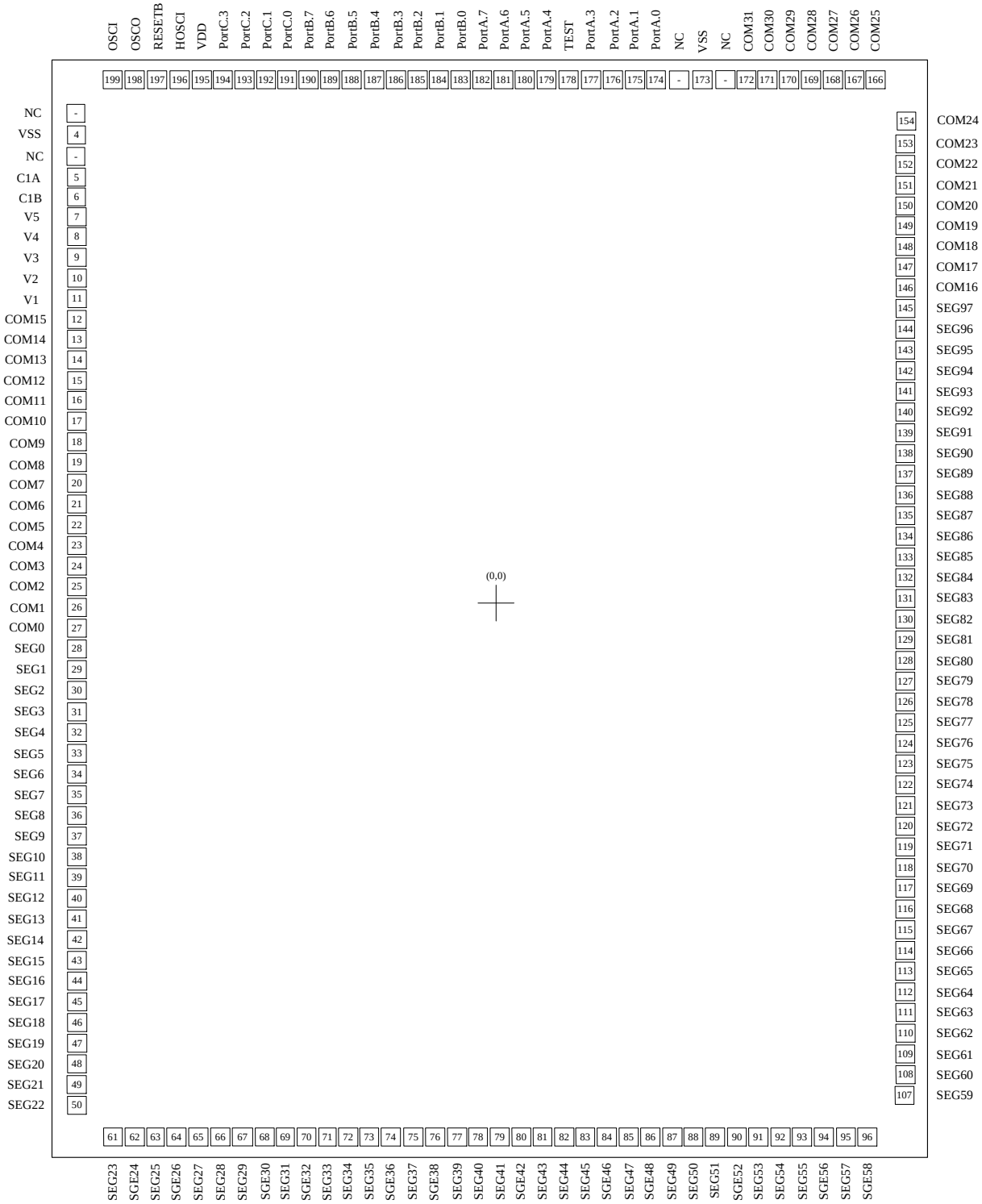


Figure 13-1 ePS6800 Pad Diagram

14 Pad Coordinate

Chip Size: 3930 × 5180 μm²

Pin No.	Symbol	X	Y	Pin No.	Symbol	X	Y
1	NC	-	-	41	SEG13/Strobe13	-1852.5	-1406.2
2	NC	-	-	42	SEG14/Strobe14	-1852.5	-1501.2
3	NC	-	-	43	SEG15/Strobe15	-1852.5	-1596.2
4	VSS	-1852.5	2306	44	SEG16	-1852.5	-1706.2
-	NC	-	-	45	SEG17	-1852.5	-1816.25
5	C1A	-1852.5	2100.8	46	SEG18	-1852.5	-1926.3
6	C1B	-1852.5	1990.8	47	SEG19	-1852.5	-2036.3
7	V5	-1852.5	1882.8	48	SEG20	-1852.5	-2146.35
8	V4	-1852.5	1774.8	49	SEG21	-1852.5	-2258.4
9	V3	-1852.5	1666.8	50	SEG22	-1852.5	-2366.4
10	V2	-1852.5	1558.8	51	NC	-	-
11	V1	-1852.5	1453.8	52	NC	-	-
12	COM15	-1852.5	1348.8	53	NC	-	-
13	COM14	-1852.5	1253.8	54	NC	-	-
14	COM13	-1852.5	1158.8	55	NC	-	-
15	COM12	-1852.5	1063.8	56	NC	-	-
16	COM11	-1852.5	968.8	57	NC	-	-
17	COM10	-1852.5	873.8	58	NC	-	-
18	COM9	-1852.5	778.8	59	NC	-	-
19	COM8	-1852.5	683.8	60	NC	-	-
20	COM7	-1852.5	588.8	61	SEG23	-1725.5	-2477.5
21	COM6	-1852.5	493.8	62	SEG24	-1622.5	-2477.5
22	COM5	-1852.5	398.8	63	SEG25	-1517.5	-2477.5
23	COM4	-1852.5	303.8	64	SEG26	-1412.5	-2477.5
24	COM3	-1852.5	208.8	65	SEG27	-1309.5	-2477.5
25	COM2	-1852.5	113.8	66	SEG28	-1207.5	-2477.5
26	COM1	-1852.5	18.8	67	SEG29	-1112.5	-2477.5
27	COM0	-1852.5	-76.2	68	SEG30	-1017.5	-2477.5
28	SEG0/Strobe0	-1852.5	-171.2	69	SEG31	-922.5	-2477.5
29	SEG1/Strobe1	-1852.5	-266.2	70	SEG32	-827.5	-2477.5
30	SEG2/Strobe2	-1852.5	-361.2	71	SEG33	-732.5	-2477.5
31	SEG3/Strobe3	-1852.5	-456.2	72	SEG34	-637.5	-2477.5
32	SEG4/Strobe4	-1852.5	-551.2	73	SEG35	-542.5	-2477.5
33	SEG5/Strobe5	-1852.5	-646.2	74	SEG36	-447.5	-2477.5
34	SEG6/Strobe6	-1852.5	-741.2	75	SEG37	-352.5	-2477.5
35	SEG7/Strobe7	-1852.5	-836.2	76	SEG38	-257.5	-2477.5
36	SEG8/Strobe8	-1852.5	-931.2	77	SEG39	-162.5	-2477.5
37	SEG9/Strobe9	-1852.5	-1026.2	78	SEG40	-67.5	-2477.5
38	SEG10/Strobe10	-1852.5	-1121.2	79	SEG41	27.5	-2477.5
39	SEG11/Strobe11	-1852.5	-1216.2	80	SEG42	122.5	-2477.5
40	SEG12/Strobe12	-1852.5	-1311.2	-	-	-	-

Pin No.	Symbol	X	Y	Pin No.	Symbol	X	Y
81	SEG43	217.5	-2477.5	121	SEG73	1852.5	-985.3
82	SEG44	312.5	-2477.5	122	SEG74	1852.5	-890.3
83	SEG45	407.5	-2477.5	123	SEG75	1852.5	-795.3
84	SEG46	502.5	-2477.5	124	SEG76	1852.5	-700.3
85	SEG47	597.5	-2477.5	125	SEG77	1852.5	-605.3
86	SEG48	692.5	-2477.5	126	SEG78	1852.5	-510.3
87	SEG49	787.5	-2477.5	127	SEG79	1852.5	-415.3
88	SEG50	882.5	-2477.5	128	SEG80	1852.5	-320.3
89	SEG51	977.5	-2477.5	129	SEG81	1852.5	-225.3
90	SEG52	1072.5	-2477.5	130	SEG82	1852.5	-130.3
91	SEG53	1167.5	-2477.5	131	SEG83	1852.5	-35.3
92	SEG54	1272.5	-2477.5	132	SEG84	1852.5	59.7
93	SEG55	1377.5	-2477.5	133	SEG85	1852.5	154.7
94	SEG56	1482.5	-2477.5	134	SEG86	1852.5	249.7
95	SEG57	1594.85	-2477.5	135	SEG87	1852.5	344.7
96	SEG58	1704.9	-2477.5	136	SEG88	1852.5	439.7
97	NC	-	-	137	SEG89	1852.5	534.7
98	NC	-	-	138	SEG90	1852.5	629.7
99	NC	-	-	139	SEG91	1852.5	724.7
100	NC	-	-	140	SEG92	1852.5	819.7
101	NC	-	-	141	SEG93	1852.5	914.7
102	NC	-	-	142	SEG94	1852.5	1009.7
103	NC	-	-	143	SEG95	1852.5	1104.7
104	NC	-	-	144	SEG96	1852.5	1199.7
105	NC	-	-	145	SEG97	1852.5	1294.7
106	NC	-	-	146	COM16	1852.5	1389.7
107	SEG59	1852.5	-2380.3	147	COM17	1852.5	1484.7
108	SEG60	1852.5	-2276.3	148	COM18	1852.5	1596.1
109	SEG61	1852.5	-2173.3	149	COM19	1852.5	1706.1
110	SEG62	1852.5	-2070.3	150	COM20	1852.5	1816.1
111	SEG63	1852.5	-1967.3	151	COM21	1852.5	1926.1
112	SEG64	1852.5	-1863.3	152	COM22	1852.5	2036.1
113	SEG65	1852.5	-1761.3	153	COM23	1852.5	2146.1
114	SEG66	1852.5	-1659.3	154	COM24	1852.5	2256.1
115	SEG67	1852.5	-1555.3	155	NC	-	-
116	SEG68	1852.5	-1460.3	156	NC	-	-
117	SEG69	1852.5	-1365.3	157	NC	-	-
118	SEG70	1852.5	-1270.3	158	NC	-	-
119	SEG71	1852.5	-1175.3	159	NC	-	-
120	SEG72	1852.5	-1080.3	160	NC	-	-

Pin No.	Symbol	X	Y	Pin No.	Symbol	X	Y
161	NC	-	-	185	PortB.2	-332.75	2477.5
162	NC	-	-	186	PortB.3	-427.75	2477.5
163	NC	-	-	187	PortB.4	-522.75	2477.5
164	NC	-	-	188	PortB.5	-617.75	2477.5
165	NC	-	-	189	PortB.6	-712.75	2477.5
166	COM25	1765.65	2477.5	190	PortB.7	-807.75	2477.5
167	COM26	1635.65	2477.5	191	PortC.0	-902.75	2477.5
168	COM27	1530.65	2477.5	192	PortC.1	-1000.75	2477.5
169	COM28	1425.65	2477.5	193	PortC.2	-1098.75	2477.5
170	COM29	1315.65	2477.5	194	PortC.3	-1196.75	2477.5
171	COM30	1215.65	2477.5	195	VDD	-1296.75	2477.5
172	COM31	1115.65	2477.5	196	HOSCI	-1406.75	2477.5
-	NC	-	-	197	RESETB	-1516.75	2477.5
173	VSS	912.45	2477.5	198	OSCO	-1626.75	2477.5
-	NC	-	-	199	OSCI	-1741.75	2477.5
174	PortA.0	712.25	2477.5	-	NC	-	-
175	PortA.1	617.25	2477.5	200	NC	-	-
176	PortA.2	522.25	2477.5	201	NC	-	-
177	PortA.3	427.25	2477.5	202	NC	-	-
178	TEST	332.25	2477.5	203	NC	-	-
179	PortA.4	237.25	2477.5	204	NC	-	-
180	PortA.5	142.25	2477.5	205	NC	-	-
181	PortA.6	47.25	2477.5	206	NC	-	-
182	PortA.7	-47.75	2477.5	207	NC	-	-
183	PortB.0	-142.75	2477.5	208	NC	-	-
184	PortB.1	-237.75	2477.5				

NOTE

For the PCB layout, the die pad must be connected to Vss (the IC substrate must be connected to Vss or kept floating open).