ePS6500

RISC II
Series MCU

Product Specification

DOC. VERSION 2.6

ELAN MICROELECTRONICS CORP.

May 2014

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

Shenzhen:
Elan Microelectronics Shenzhen, Ltd.
8A Floor, Microprofit Building
Gaoxin South Road 6
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.
6F, Ke Yuan Building
No. 5 Bibo Road
Zhangjiang Hi-Tech Park
Shanghai, CHINA 201203
Tel: +86 21 5080-3866
Fax: +86 21 5080-0273
elan-sh@elanic.com.cn

Contents

Specification Revision History

Doc. Version	Revision Description	Date
1.0	Initial version	2005/04/21
1.1	1. Added Electrical Characteristics section. 2. Added Event Counter to Timer 0.	2005/05/09 2005/06/14
1.2	1. Added 1/3 bias and 1/2 bias to the LCD driver. 2. Modified the LCD 3x46 to 3x40, and modified Port A.0~3 I/O pins to input pins. 3. Added Pin Configuration.	2005/07/29 2005/08/15 2005/08/29
1.3	Added PA.4~7 control bit (Bit 7 “KEH” of LCDCON register)	2005/09/08
1.4	1. Added Pad Diagram and Locations 2. Modified the VDD=1.5V Electrical Characteristics	2005/10/13
1.5	1. Modified the Pad Diagram, Locations (100QFP) and “NC1 and NC2” Pin Description 2. Added Fast to Slow mode Code Example to MCU Operation Mode (Section 7.3.2)	2005/11/01
1.6	1. Modified the Pad Diagram, Pin Description: “NC1 & NC2” modified to “C1A & C1B” 2. Modified the Electrical Characteristics 3. Modified the 32768 Crystal Stable Time 4. Modified the description of DCRA	2006/01/11
1.7	1. Modified the PA3 Pin Description 2. Modified the Key I/O: PA3 has no Auto key scan function, KEL cannot control PA.3 3. Modified the Electrical Characteristics: PA pull-up resistor 4. Deleted the Electrical Characteristics for VDD=3V 5. Modified the Application circuit	2006/01/13
1.8	1. Modified the Electrical Characteristics for VDD=1.5V 2. Added Electrical Characteristics for VDD=3V	2006/06/07
1.9	1. Modified the Electrical Characteristics for VDD=1.5V: PA4-7 small pull-up resistor 2. Added code option for Key matrix set	2006/06/15
2.0	1. Modified the Electrical Characteristics for VDD=1.5V: Strobe Output ON-resistance	2006/06/19
2.1	1. Added the “SCAN” bit Note at Section 8.5.2.1	2006/08/24
2.2	1. Added the Note of the code example (Section 7.4.1) 2. Modified the pop interrupt register of the code example (Section 7.4.1) 3. Modified the push interrupt register of the code example (Section 7.4.1)	2007/03/12 2007/03/13 2007/03/23
2.3	Modified the Note of the Instruction Set	2007/06/27
2.4	1. Modified the max. supply voltage to 2.0V on Section 9.1 2. Modified the supply current on Section 9.1 3. Added a Note on Section 6 Code Option	2008/11/07
2.5	1. Modified the pin description of the OSCI pin in Section 5.1 2. Deleted the Package Pin from Section 4	2010/10/29
2.6	1. Added Notes for the special register “LCDARL” in Section 1, Section 7.1.6 and Section 8.6.1 2. Modified the Electrical Characteristics	2014/05/15

1 General Description

The **ePS6500** is an 8-bit RISC MCU embedded with a 5×40 LCD driver along with two 8-bit timers, one 16-bit general timer, and a watchdog timer. It has also an on-chip 1.2K bytes RAM and 8K 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 ePS6500 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 (05h) Bit 7 ~ Bit 4.*
- *Do not use Register BSR1 (07h) Bit 7 ~ Bit 4.*
- *Check the range of BSR and BSR1, which should be from 0x00~0x09.*
- *Do not use LCD RAM 28h ~ FFh.*
- *Do not use JDNZ at FSR1 (04h) special register.*
- *Do not use JDNZ at LCDARL (09h) special register.*
- *Do not use PUSH, POP by "MOV A,r" to avoid affecting 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 / Crystal oscillator
  - High-frequency:200kHz / 300kHz / 500kHz External RC oscillator
- One Instruction cycle time = 2 × System clock time
- Program ROM addressing**:** Maximum 8K words
- 128 bytes un-banked RAM including special registers and common registers
- 10×128 bytes banked RAM
- Max. of 32-level RAM stack

- Table Lookup function is fast and highly efficient when combined with Repeat instruction
- Register-to-Register move instruction
- Compare and Branch in one instruction (2 cycles)
- Single Repeat function (256 repeat times max.)
- Decimal ADD and SUB instruction
- Full range Call and Jump ability (2 cycles)

Peripheral

- 4 input pins (Port A.0~3) and 12 general I/O pins (Port A.4~7, Port B.0~7)
- 5/4/3 COM × 40 SEG LCD driver (embedded)
- 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 48 keys max (Key matrix: Port A.0~2 and SEG0~15 with automatic key scan)

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 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

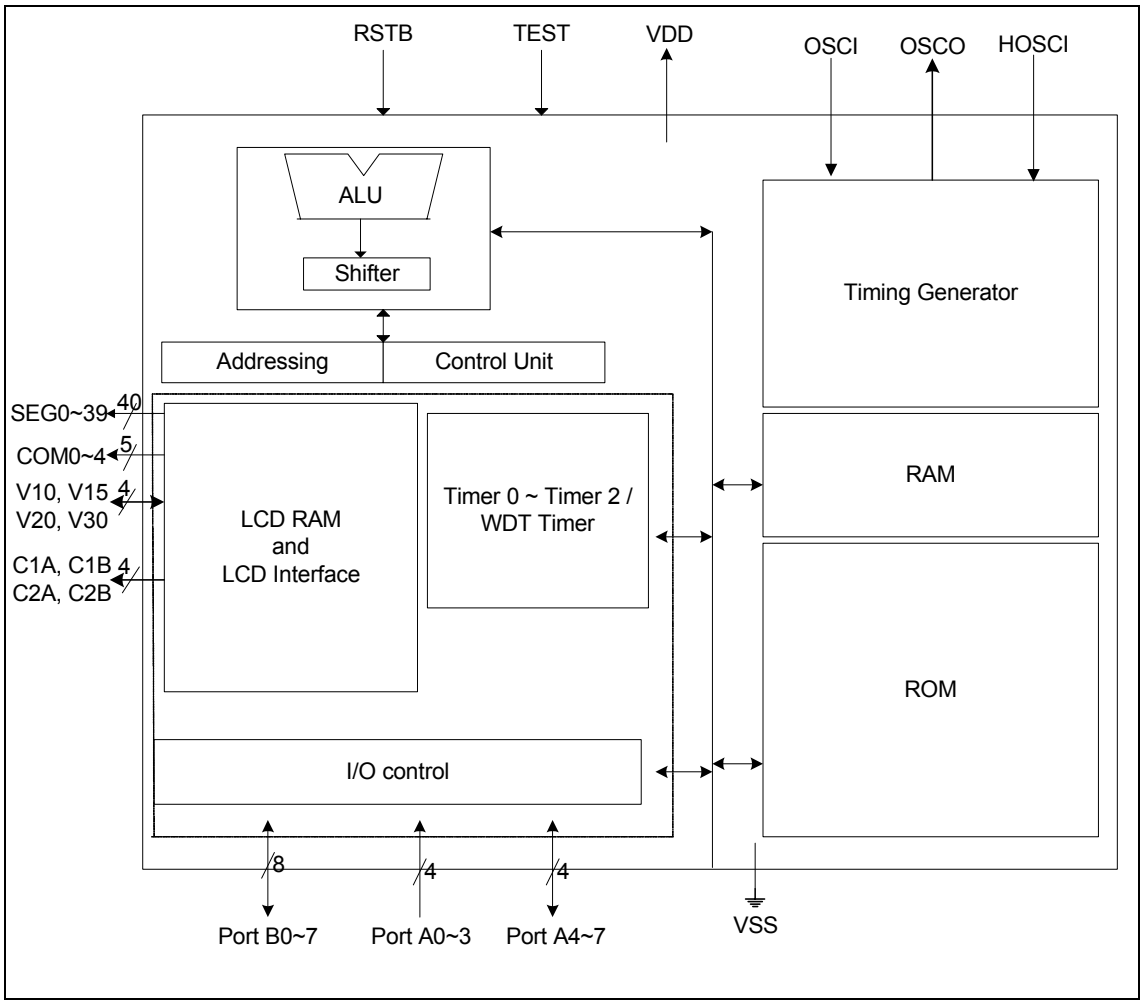


Figure 3-1 ePS6500 Block Diagram

4 Pin Assignment

■ 68-pin Chip Format

No.	Pin Name	No.	Pin Name	No.	Pin Name	No.	Pin Name
1	NC	26	NC	51	NC	76	NC
2	NC	27	NC	52	NC	77	NC
3	NC	28	NC	53	NC	78	NC
4	NC	29	NC	54	NC	79	NC
5	NC	30	NC	55	NC	80	NC
6	NC	31	NC	56	NC	81	NC
7	COM4	32	NC	57	SEG29	82	C1A
8	COM3	33	SEG12/Strobe12	58	SEG30	83	C1B
9	COM2	34	SEG13/Strobe13	59	SEG31	84	C2A
10	COM1	35	SEG14/Strobe14	60	PortA.7	85	C2B
11	COM0	36	SEG15/Strobe15	61	PortA.6	86	V20
12	SEG0/Strobe0	37	SEG16	62	PortA.5	87	V15
13	SEG1/Strobe1	38	SEG17	63	PortA.4	88	V10
14	SEG2/Strobe2	39	SEG18	64	VSS	89	Port A.0
15	SEG3/Strobe3	40	SEG19	65	SEG32/Port B.0	90	Port A.1
16	SEG4/Strobe4	41	SEG20	66	SEG33/Port B.1	91	Port A.2
17	SEG5/Strobe5	42	SEG21	67	SEG34/Port B.2	92	Port A.3
18	SEG6/Strobe6	43	SEG22	68	SEG35/Port B.3	93	TEST
19	SEG7/Strobe7	44	SEG23	69	SEG36/Port B.4	94	RESETB
20	SEG8/Strobe8	45	SEG24	70	SEG37/Port B.5	95	OSCO
21	SEG9/Strobe9	46	SEG25	71	SEG38/Port B.6	96	OSCI
22	SEG10/Strobe10	47	SEG26	72	SEG39/Port B.7 (EVIN)	97	HOSCI
23	SEG11/Strobe11	48	SEG27	73	V30	98	VDD
24	NC	49	SEG28	74	NC	99	NC
25	NC	50	NC	75	NC	100	NC

5 Pin Description

5.1 MCU System Pins (7 Pins)

Name	I/O/P Type	Description	Note
VDD	P	Digital and Analog positive power supply, ranging from 1.2V~1.8V. Connect to VSS through the capacitors (0.1μF).	-
VSS	P	Digital and Analog negative power supply.	-
RSTB	I	System reset pin. Low active, connect 0.1 μF to VSS.	Int. pull-up
TEST	I	Test mode select pin (High active). For chip internal test only, normally connect to VSS.	Int. Pull Down
OSCI	I	Crystal oscillator connecting pin	–
OSCO	O	Crystal oscillator connecting pin	–
HOSCI	I	Hi-Speed RC oscillator connecting pin.	Ext. R to VDD

5.2 Embedded LCD Pins (53 Pins)

Name	I/O/P Type	Description	Note
COM0~COM4	O	LCD common signal output pin	–
SEG0~SEG15	O	LCD segment signal output pin shared with Key Strobe 0~15	–
SEG16~ SEG31	O	LCD segment signal output pin	–
SEG32~SEG39 / Port B.0~7	I/O	LCD segment signal output pin or I/O pin; defined by code option	–
C1A, C1B		LCD voltage charge-pump pin. Connect 0.1 μF between C1A and C1B.	–
C2A, C2B		LCD voltage charge-pump pin. Connect 0.1 μF between C2A and C2B.	–
V30, V20, V15, V10	O	LCD bias Pin. Connect 0.1 μF to Vss	–

5.3 I/O Port (8 Pins)

Port	Bit	Function	I/O Type	Power Source	Description	Note
Port A	Bits 2~0 (for key scan)	General Input	I	VDD	Key input	Int. Pull-up (**R1:** small resistor, **R2:** Large resistor) controllable
		Interrupt and wake up	I	VDD	Input port interrupt and wake-up pin	
	Bit 3	General Input	I	VDD		
		Interrupt and wake up	I	VDD	Input port interrupt and wake-up pin	
	Bits 7~4	General Input	I	VDD		Int. Pull-up (**R2:** Large resistor) controllable
		Interrupt and wake up	I	VDD	Input port interrupt and wake-up pin	
		General Output	O	VDD		

6 Code Option

Located at Address 0x000C~0x000F of Program ROM

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

NOTE

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

- Low Frequency Oscillator**:**
  - Select “Crystal” oscillator or “Internal RC” oscillator
- Reset pin condition**:**
  - Select “Level hold” or “One short” for reset pin
- Maximum duty ratio and LCD bias option**:**
  - Select “1/5 duty and 1/3 bias” or “1/4 duty and 1/3 bias” or “1/3 duty and 1/2 bias”
- Port B low nibble control bit (SEG32~SEG35)**:**
  - Select “LCD segment signal output” or “general I/O functions”
- Port B high nibble control bit (SEG36~SEG39)**:**
  - Select “LCD segment signal output” or “general I/O functions”
- Key matrix combination**:**
  - Select “Port A and SEG” or “Port A and Port B”

7 Function Description

7.1 Reset Function

Reset can be generated by one of the following:

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

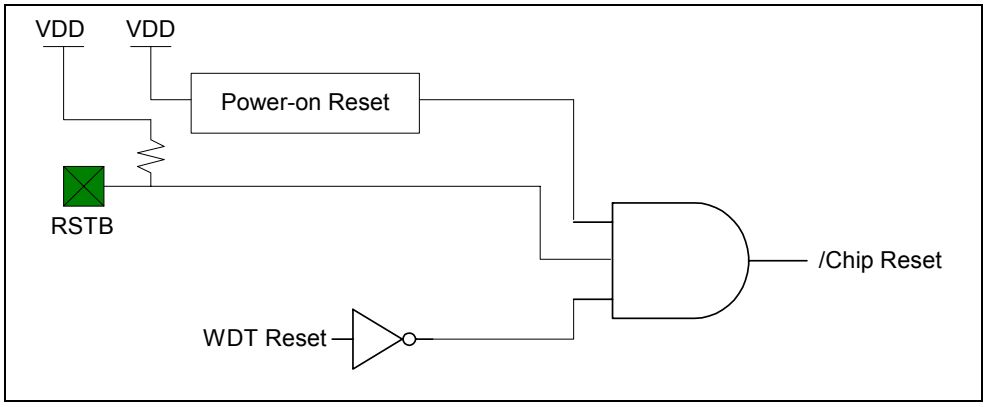


Figure 7-1 On-chip Reset Schematic

7.1.1 Power-on Reset

The power-on reset circuit holds the device under reset condition until VDD is above Vpor (power-on reset voltage). Whenever the voltage supply 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 a 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, the WDT value should be cleared 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**

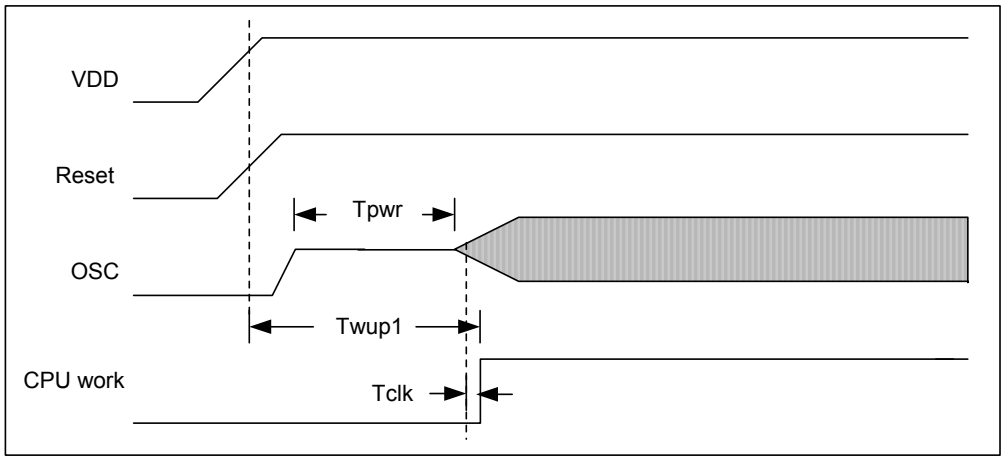


Figure 7-2a Power on Reset Timing Diagram

■ **Sleep Mode Wake-up Timing**

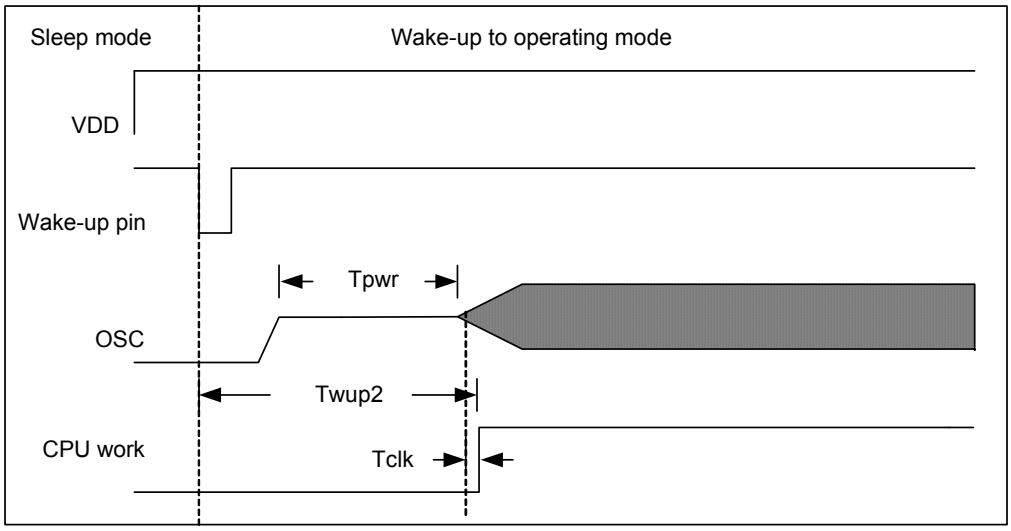


Figure 7-2b Sleep Mode Wake-up Timing Diagram

Condition: Vdd = 1.5V, Cosc = 20pF, 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 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 borrowed from a signed bit (Bit 7).

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

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

NOTE

1. *When OV=1 after a signed arithmetic, 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 occurred, the MSB of the Accumulator should be cleared to obtain the correct value.*
   When underflow occurred, the MSB of the Accumulator should be set 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 a negative value, and the ACC signed bit will be affected.

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

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

ACC = 40h SLE=1, means the result is less than or equal to 0 (negative value)
OV=1, Underflow occurred and the result is borrowed from a 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 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 a reset occurs, the special function registers are reset to their initial value except for the /**TO** and **/PD** bits of the Status register.

Bit 7 (/TO)	Bit 6 (/PD)	Event
0	0	WDT time out reset from Sleep mode
0	1	WDT time out reset (but 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 (PCM: PCL) is clear 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	Port A	xxxx xxxx
01h	FSR0	0000 0000	11h	Port B	xxxx xxxx
02h	BSR	0000 0000	12h	-	-
03h	INDF1	---- ----[1]	13h	General RAM	uuuu uuuu
04h	FSR1	1000 0000	14h	General RAM	uuuu uuuu
05h	BSR1	0000 0000	15h	General RAM	uuuu uuuu
06h	STKPTR	0000 0000	16h	General RAM	uuuu uuuu
07h	PCL	0000 0000	17h	General RAM	uuuu uuuu
08h	PCM	0000 0000	18h	General RAM	uuuu uuuu
09h	LCDARL	xx00 0000 [4]	19h	General RAM	uuuu uuuu
0Ah	ACC	xxxx xxxx	1Ah	General RAM	uuuu uuuu
0Bh	TABPTRL	0000 0000	1Bh	General RAM	uuuu uuuu
0Ch	TABPTRM	0000 0000	1Ch	General RAM	uuuu uuuu
0Dh	TABPTRH	uuuu uuuu	1Dh	General RAM	uuuu uuuu
0Eh	LCDDATA	---- ----[1]	1Eh	General RAM	uuuu uuuu
0Fh	STATUS	cuxx xxxx [2]	1Fh	General RAM	uuuu uuuu

7.1.6.2 Control Register

Addr.	Name	Initial Value	Addr.	Name	Initial Value
20h	STBCON	0000 0000	2Ch	PAINTSTA	0000 0000
21h	INTCON	---- -000	2Dh	DCRA	1111 ----
22h	INTSTA	---- -000	2Eh	PBCON	0000 0000
23h	TR01CON	0000 0000	2Fh	DCRB	1111 1111
24h	TRL0L	uuuu uuuu	30h	–	
25h	TRL0H	uuuu uuuu	31h	–	
26h	TRL1	uuuu uuuu	32h	LCDCON	000- 00-0
27h	TR2WCON	0000 0000	33h	POST_ID	-111 –000
28h	TRL2	uuuu uuuu	34h	CPUCON	---- -00c[3]
29h	PACON	0000 0000	35h	T0CL	0000 0000
2Ah	PAWAKE	0000 0000	36h	T0CH	0000 0000
2Bh	PAINTEN	0000 0000			

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

[1] *Not a physical register.*

[2] *If it is a power-on reset or 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 the code option when the MCU resets.*

[4] *Bits 6~7 of R9 (LCDARL) are* ***unknown values*** *on the ePS6500 (and ePS5001 & ePS5002), and user do not use the instruction "JDNZ" at R9. However Bits 6~7 of R9 (LCDARL) are different from ePSP6000 and PMePS60 board.*

7.2 Oscillator System

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

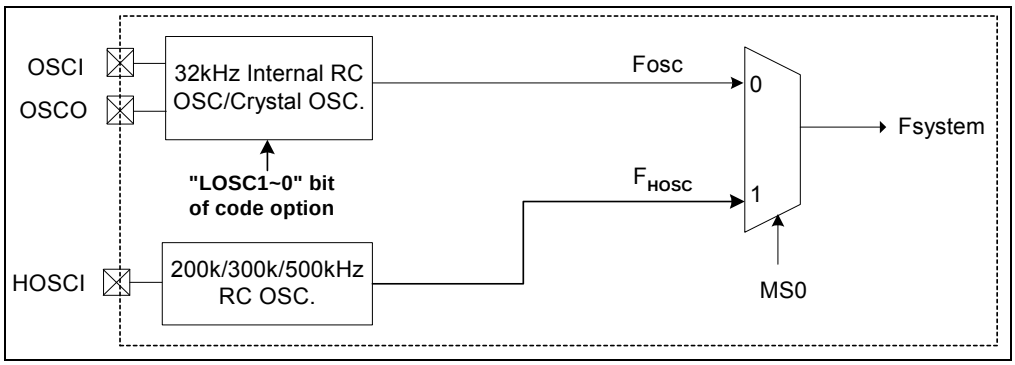


Figure 7-3 Oscillator System Function Block Diagram

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

0: Slow mode (MCU system Clock is from FOSC)

1: Fast mode (MCU system Clock is from FHOSC)

7.2.1 32.8kHz RC or 32768Hz Crystal Oscillator

- 32.8kHz RC Internal oscillator:

  Select “RC oscillator for FOSC” in the code option and allow OSCI and OSCO pins to remain floating.

- 32768kHz Crystal oscillator:

  Select “Crystal oscillator for FOSC” in the code option and connect a crystal and a resistor (10MΩ) between OSCI and OSCO pins. The OSCI and OSCO pins are also connected to ground through a 20pF capacitor respectively.

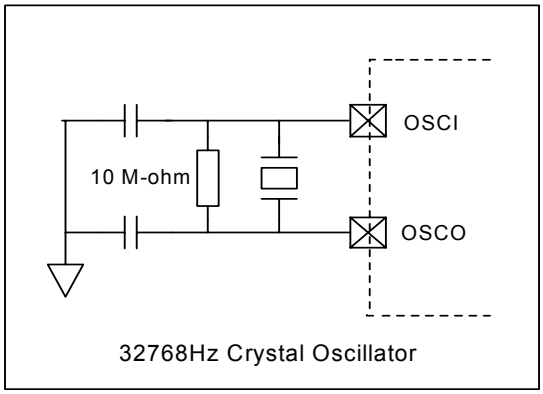


Figure 7-4 Slow Mode Crystal Circuit Diagram

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

A resistor should be connected between HOSCI and Vdd pin.

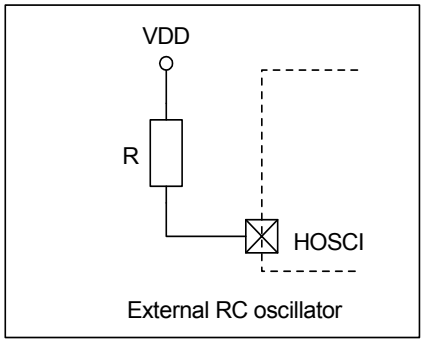


Figure 7-5 Fast Mode RC Oscillators Circuit Diagram

7.3 MCU Operation Mode

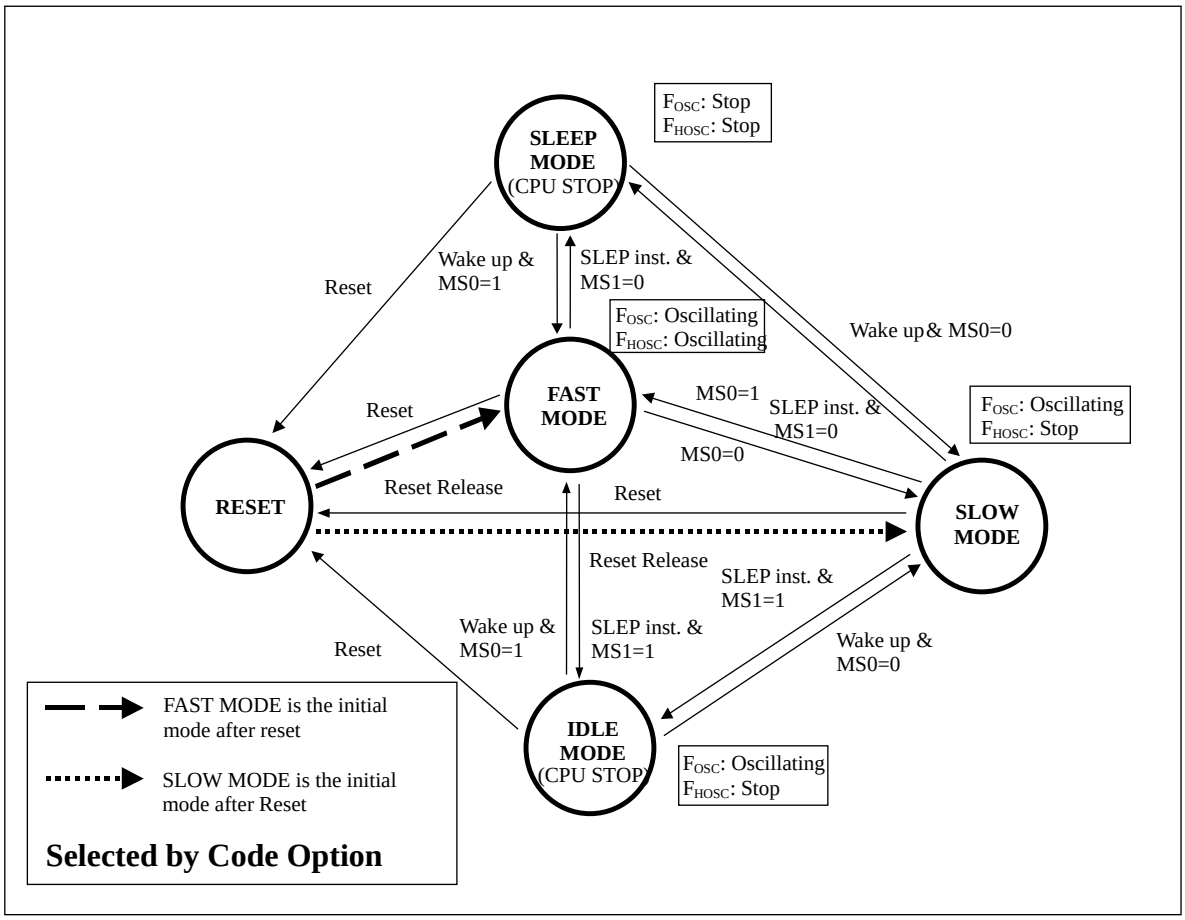


Figure 7-6 MCU Operation Block Diagram

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 NOT supported

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

7.3.1 Slow, Fast, Sleep, and Idle Mode Operation

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

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

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

0: Slow Mode

1: Fast Mode

Bit 1 (MS1): Select Sleep Mode or Idle Mode after executing "SLEP" instruction.

0: Sleep Mode

1: Idle Mode

■ **Slow Mode:**

When 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.

NOTE

The instruction "NOP" should be added after the "BC CPUCON, MS0" instruction when the MCU is made to enter into Slow Mode from Fast Mode. See the code example at the end of this Section.

■ **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 32kHz 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 pin wake up (if enabled) and returns to the either Slow Mode (MS0=0) or Fast Mode (MS0=1)

NOTE

All registers remain unchanged during Sleep Mode.

■ Sleep Mode:

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

NOTE

- *The /PD bit of the Status Register (R0Fh) 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:

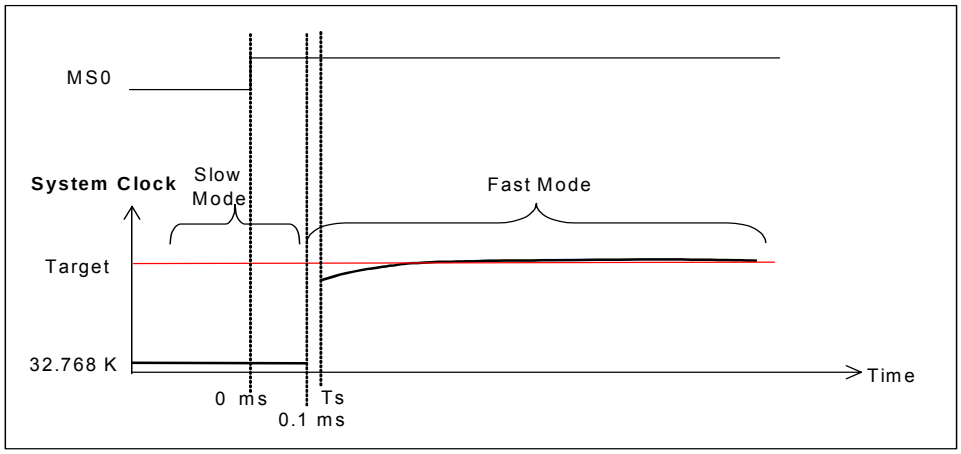


Figure 7-7 Slow Mode to Fast Mode Timing Diagram

NOTE

1. *Slow Mode switches to Fast Mode at Time=0ms.*
2. *System clock will switch to Fast Mode after 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 is stabilized at Time=Ts (around 15 μs~30 μs).*

■ **Code Example:**

```
;Entry FAST mode
BS      CPUCON,MS0

;Entry SLOW mode
BC      CPUCON,MS0

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

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


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

7.3.2 Wake-up Operation

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

When in Idle Mode, the 32kHz oscillator keeps on running. The MCU is awakened by input port (Port A) or Timer 1, 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 (R2Ah): Port A Wake-up Function Control Register**

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

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

0: Disable Port A.7 ~ Port A.0 wake-up function

1: Enable Port A.7 ~ Port A.0 wake-up function

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

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

Bit 7 (T1WKEN): 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 (R34h) MCU Control Register**

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

Bit 2 (GLINT): Global interrupt control bit

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

1: Enables all un-masked interrupts

■ **Interrupt Vector**

Interrupt Level	Interrupt Source	Start Address	Remarks
–	RESET	0x00000	–
Level 1	Port A.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:
     MOVPR  StatusBuf,Status
     MOV    AccBuf,A
     RET

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

RETI
```

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

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

■ **PAINTSTA (R2Ch): Port A.7 ~ Port A.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): Port A.7 ~ Port A.0 Interrupt status

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

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

■ **PAINTEN (R2Bh): Port A.7 ~ Port A.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
       :
     SJMP   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, the Timer 0 interrupt takes place and the TRL0H: TRL0L value is automatically reloaded into the timer.

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

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

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

0: Disable Timer 0 interrupt function

1: Enable Timer 0 interrupt function

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

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

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 INTCON (R21h) Timer Interrupt Control Register**

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

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

0: Disable Timer 1 interrupt function

1: Enable Timer 1 interrupt function

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

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

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

Set to **"1"** when the 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, an interrupt occurs and the TRL2 value will be reloaded to the timer.

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

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

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

0: Disable Timer 2 interrupt function

1: Enable Timer 2 interrupt function

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

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

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 = 8K Words	
Address.	**Description**
0000h | 000Bh	Interrupt Vector (12 words)
000Ch | 000Fh	Code Option (4 words)
0010h | 001Fh	Test Program (16 words)
0020h | 1FFFh	Program or Fixed data region

7.6 RAM Map for Special and Control Registers

RAM Size: 86 Bytes + 10 Banks × 128 Bytes = 1366 Bytes

7.6.1 Special and Control Registers

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

Addr.	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	INDF0	R/W							
		Indirect Addressing Pointer 0							
1	FSR0	R/W							
		File Select Register 0 for INDF0 (R0)							
2	BSR	R	R	R	R	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	Fixed 0	Fixed 0	Bank select register (for INDF0 & general)			
3	INDF1	R/W							
		Indirect Addressing 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	R	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	Fixed 0	Fixed 0	Bank select Register 1 (for INDF1)			
6	STKPTR	R/W							
		Stack Pointer							
7	PCL	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
8	PCM	R	R	R/W	R/W	R/W	R/W	R/W	R/W
		Fixed 0	Fixed 0	PC13	PC12	PC11	PC10	PC9	PC8
9	LCDARL	R/W							
		LCD RAM Column Address							
A	ACC	R/W							
		Accumulator							
B	TABPTRL	R/W							
		Low Byte of Table Pointer							
C	TABPTRM	R	R/W						
		Fixed 0	Middle Byte of Table Pointer						
D	TABPTRH	R/W							
E	LCDDATA	R/W							
		Indirect Register to LCD RAM							
F	STATUS	R	R	R/W	R/W	R/W	R/W	R/W	R/W
		/TO	/PD	SGE	SLE	OV	Z	DC	C
10	Port A	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
11	Port B	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		Port B.7	Port B.6	Port B.5	Port B.4	Port B.3	Port B.2	Port B.1	Port B.0

(Continuation)

Addr.	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
20	STBCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		SCAN	KEL	R1EN	BitST	STB3	STB2	STB1	STB0
21	INTCON	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	TMR2IE	TMR1IE	TMR0IE
22	INTSTA	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	TMR2I	TMR1I	TMR0I
23	TR01CON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		T1WKEN	T1EN	T1PSR1	T1PSR0	T0EN	T0CS	T0PSR1	T0PSR0
24	TRL0L	R/W							
		Timer 0 Auto-reload Register Low Byte							
25	TRL0H	R/W							
		Timer 0 Auto-reload Register High Byte							
26	TRL1	R/W							
		Timer 1 Auto-reload Register							
27	TR2WCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		WDTEN	T0ENMD	WDTPSR1	WDTPSR0	T2EN	T2CS	T2PSR1	T2PSR0
28	TRL2	R/W							
		Timer 2 Auto-reload Register							
29	PACON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7PU	PA6PU	PA5PU	PA4PU	PA3PU	PA2PU	PA1PU	PA0PU
2A	PAWAKE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		WKEN7	WKEN6	WKEN5	WKEN4	WKEN3	WKEN2	WKEN1	WKEN0
2B	PAINTEN	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7IE	PA6IE	PA5IE	PA4IE	PA3IE	PA2IE	PA1IE	PA0IE
2C	PAINTSTA	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PA7I	PA6I	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
2D	DCRA	R/W	R/W	R/W	R/W	-	-	-	-
		PA7DC	PA6DC	PA5DC	PA4DC	-	-	-	-
2E	PBCON	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PB7PU	PB6PU	PB5PU	PB4PU	PB3PU	PB2PU	PB1PU	PB0PU
2F	DCRB	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		PB7DC	PB6DC	PB5DC	PB4DC	PB3DC	PB2DC	PB1DC	PB0DC
32	LCDCON	R/W	R/W	R/W	-	R/W	R/W	-	R/W
		KEH	BLANK	LCDON	-	LCR1	LCR0	-	LBVON
33	POST_ID	-	R/W	R/W	R/W	-	R/W	R/W	R/W
		-	LCD_ID	FSR1_ID	FSR0_ID	-	LCD_PE	FSR1_PE	FSR0_PE
34	CPUCON	-	-	-	-	-	R/W	R/W	R/W
		-	-	-	-	-	GLINT	MS1	MS0
35	T0CL	R							
		Timer 0 Counting Value Low Byte Register							
36	T0CH	R							
		Timer 0 Counting Value High Byte Register							

7.6.2 Other Unbanked General RAM

Address	Unbanked
13h | 1Fh	General Purpose RAM
37h | 7Fh	General Purpose RAM

7.6.3 Banked General RAM

Address	Bank 0	Bank 1	Bank 2	Bank 3	……	Bank 9
80h | FFh	General Purpose RAM	General Purpose RAM	General Purpose RAM	General Purpose RAM	……	General Purpose RAM

7.7 LCD RAM Map

■ **1/5 Duty**

RAM Address LCDARL		COM0	COM1	COM2	COM3	COM4	-	-	-
		Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
SEG0	00H								
:	:								
SEG39	27H								

■ **1/4 Duty**

RAM Address LCDARL		COM0	COM1	COM2	COM3	-	-	-	-
		Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
SEG0	00H								
:	:								
SEG39	27H								

■ **1/3 Duty**

RAM Address LCDARL		COM0	COM1	COM2	-	-	-	-	-
		Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
SEG0	00H								
:	:								
SEG39	27H								

7.8 Special Function Registers

7.8.1 ACC (R0Ah): Accumulator

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

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

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

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

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

Bit 4 (FSR0_ID): **1:** Auto increase FSR0

0: Auto decrease FSR0

Bit 5 (FSR1_ID): **1:** Auto increase FSR1

0: Auto decrease FSR1

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

BSR (R02h) Determines which bank is active (working bank) among the 10 banks (Bank 0 ~ Bank 9).

FSR0 (R01h) Address register for INDF0. Up to 256 bytes (Address: 00 ~ 0FFh) can be selected.

INDF0 (R00h) Not a physically implemented register.

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

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

FSR1 (R04h) Address register for INDF1. Up to 128 bytes (Address: 80 ~ 0FFh) can be selected. Bit 7 of FSR1 is fixed to “1.”

INDF1 (R03h) 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 Bank1
        :
```

■ INDF1 Linear Address Capabilities

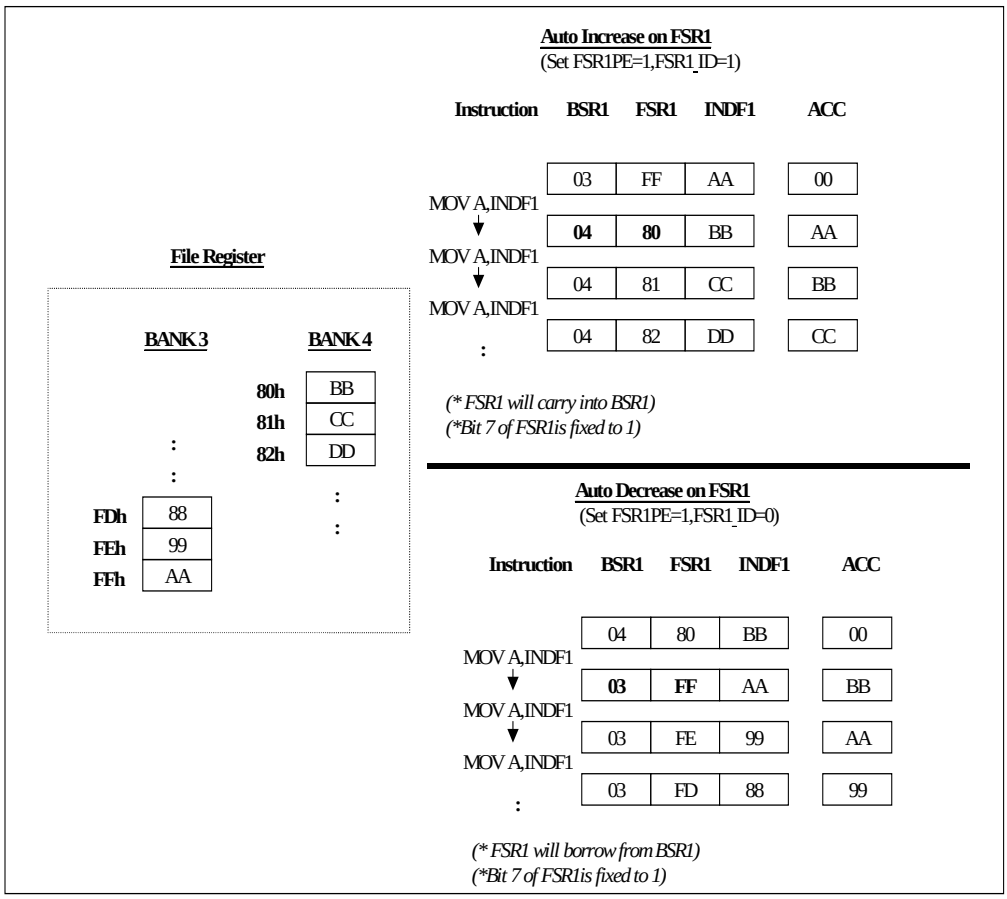


Figure 7-8 INDF1 Linear Address Capabilities Diagram

■ **Code Example:**

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

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

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

7.8.5 STKPTR (R06h): Stack Pointer Register

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

NOTE

This Bank RAM does not include the stack RAM. The stack RAM is independent and cannot be seen.

7.8.6 PCL, PCM (R07h, R08h): Program Counter Registers

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
–	–	–	PCM					PCL							

The configuration structure can generate up to 8K×16 on-chip ROM addresses for the relative programming instruction codes.

"S0CALL" Loads the low 12 bits of the PC (4K×16 ROM)

"SCALL" or **"SJMP"** Loads the low 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 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
```

7.8.7 TABPTRL, TABPTRM, (R0Bh, R0Ch): Table Pointer Registers

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	TABPTRM						TABPTRL							

Program ROM or Internal ROM address register.

Bit 13 ~ Bit 1: Used to point the memory address.

Bit 0: Used to select whether low byte or high byte (see TBRD instruction in the Instruction Set under Section 12)

■ **Code Example:**

```
; *** Program ROM
        :
        :
     TBPTM  #(PROMTabB*2)/100H
     TBPTL  #PROMTabB*2
        :
        :
     TBRD   0,ACC                       ; no 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
```

7.8.8 Port A, Port B (R10h, R11h): General I/O Pin Registers

Port A (R10h): Port A.0~7 are general I/O pin registers

Port B (R11h): Port B.0~7 are general I/O pin registers

7.8.9 STBCON (R20): Strobe Output Control Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCAN	KEL	R1EN	BitST	STB3	STB2	STB1	STB0

Bit 7 (SCAN): Automatic key scan or specify the scan signal bit by bit

0: Key scan is specified as “Bits STB 3~0 as defined”

1: Auto strobe scanning

Bit 6 (KEL): Low nibble key input enable/disable control bit

0: Disable Low nibble key input function (Port A.0~2 do not correspond with Key input in software scan mode)

1: Enable Low nibble key input function (Port A.0~2 correspond with Key input in software scan mode)

Bit 5 (R1EN): R1 pull up resistor (small resistor) control bit for Port A.3 ~ Port A.0.

0: Disable R1 pull up resistor

1: Enable R1 pull up resistor

Bit 4 (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 3 ~ Bit 0 (STB3 ~ 0): 16 to 1 multiplexing selector of key strobe pin

7.8.10 PACON (R29h): 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): Enable PortA.0 ~ Port A.7 pull-up resistor bits

0: Disable Port A.0 ~ Port A.7 pull up resistor

1: Enable Port A.0 ~ Port A.7 pull up resistor

7.8.11 PAWAKE (R2Ah): 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

Bit 7 ~ Bit 0 (WKEN7 ~ WKEN0): Wake-up enable control bits of Port A.7~Port A.0.

0: Disable Port A.7 ~ Port A.0 wake-up function

1: Enable Port A.7 ~ Port A.0 wake-up function

NOTE

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

7.8.12 PAINTEN (R2Bh): 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.

7.8.13 PAINTSTA (R2Ch): 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 ~ Port A.0 interrupt bits

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

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

7.8.14 DCRA (R2Dh): 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	-	-	-	-

Bit 7 ~ Bit 4 (PA7DC ~ PA4DC): PortA.4~Port A.7 direction control bits

0: Set to output pin

1: Set to input pin

7.8.15 PBCON (R2Eh): 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.

7.8.16 DCRB (R2Fh): 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 to output pin

1: Set to input pin

NOTE

When a Port B bit is set to input pin, a 5 µsec delay in reading the Port B data must be provided. Otherwise, read data will be inaccurate. See Example below.

■ **Code Example:**

```
; *** Set PortB to input pins
     MOV    A,#0XFF
     MOV    DCRB,A
     MOV    PBCON,A
Read_PB:
     JBS    PORTB,0,Read_PB
        Delay 5usec
     JBS    PORTB,0,Read_PB
     SJMP   Read_PB
```

8 Peripheral

8.1 Timer 0 (16 Bits Timer with Event Counter Function)

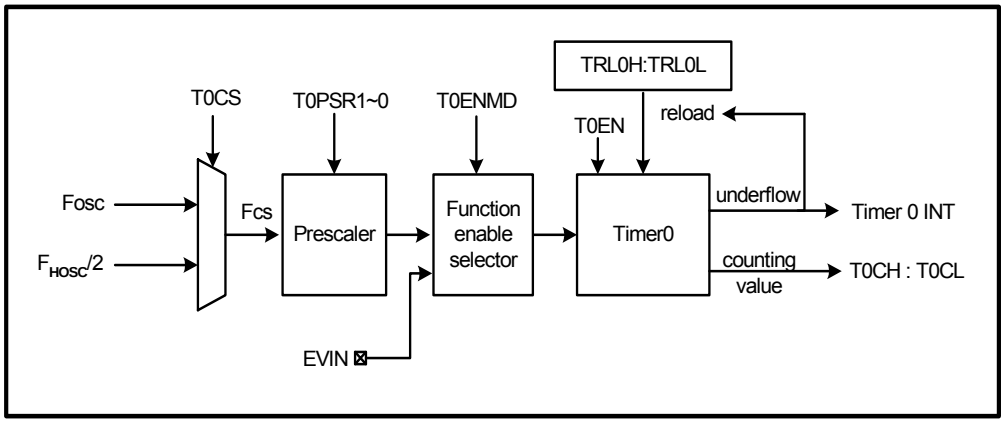


Figure 8-1 Timer 0 Function Block Diagram

■ **Timer 0 Mode**

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

A prescaler for the timer is also provided. The T0PSR1 ~ T0PSR0 bits of TR01CON register determine the prescaler ratio and generate different clock rates as clock source for the timer.

The 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 be reloaded into the 16-bit counter.

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

■ **Event Counter Mode, EVIN (PortB.7) Pin**

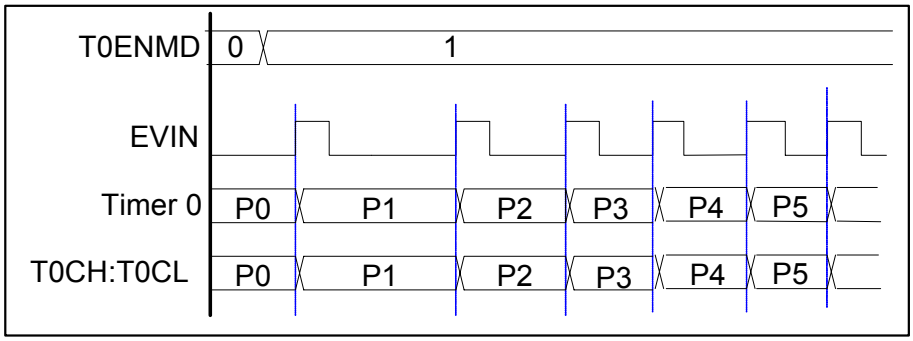
The Event counter is a function that allows the 16-bit counter value to be decremented by one when an event occurs on the EVIN pin at every rising edge. In other words, the clock source of Timer 0 is from an external event (EVIN pin).

The EVIN pin can be configured into event counter input function by setting the T0ENMD bit of the TR2WCON 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, Port B.7 will be fixed as input pin and Port B.7 cannot be controlled by the "Port B high nibble control bit" code option.

■ **Event Counter Mode Example:**



8.1.1 Timer 0 Registers

■ **TRL0H:TRL0L (R25h, R24h): 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 register values will automatically be reloaded into the 16 bits counter.

■ **T0CH:T0CL (R36H, R35H): Timer 0 Counter Value Register**

Used to store the value compared with Timer 0 register.

■ **TR01CON (R23h): Timer 0 and Timer 1 Control Register**

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

Bit 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

■ **TR2WCON (R27h): Timer 2/Watchdog Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTEN	**T0ENMD**	WDTPSR1	WDTPSR0	T2EN	T2CS	T2PSR1	T2PSR0

Bit 6 (T0ENMD): Timer 0/Event counter selection bit

0: Timer 0 mode

1: Event counter mode

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

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

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

0: Disable all interrupts

1: Enable all un-mask interrupts

■ **INTCON (R21h): Timer Interrupt Control Register**

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

Bit 0 (TMR0IE): Timer 0 interrupt control bit

0: Disable Interrupt function

1: Enable Interrupt function

■ **INTSTA (R22h): Timer Interrupt Status Register**

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

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

■ **Code Example:**

```
; === Timer 0 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR0I,Q_Time
     BC      INTSTA,TMR0I
     BTG     PORT A,7
Q_Time:
     POP
     RETI
; === Timer0 = [1/(300K/2)] * [1 x(1FFFh + 1)]
Timer0SR:
      :
     System setting 300KHz
     PA.7 setting output pin
      :
     MOV     A,#0B00000100
     AND     TR01CON,A               ; FHOSC & Pre-scale 1:1
     MOV     A,#0X1F
     MOV     TRL0H,A
     MOV     A,#0XFF
     MOV     TRL0L,A                 ; 13.65ms=[1x(8191 + 1)/(300K/2)
     BC      TR2TWCON,T0ENMD         ; 0=>Timer 0 mode
     BS      TR01CON,T0EN            ; Timer 0 enable
     BS      INTCON,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     PORT A,7
Q_Time:
     POP
     RETI
; === Event counter set
TR0_Event:
      :
     PA.7 setting output pin
       :
```

(Continuation)

```
    MOV     A,#0XFF                 ; Switch 256 times reload
    MOV     TRL0L,A
    CLR     TRL0H                   ; Count start 00FFh
    BS      TR2TWCON,T0ENMD         ; 1=>Event counter mode
    BS      INTCON,TMR0IE           ; Enable Timer 0 interrupt
    BS      CPUCON,GLINT            ; Enable global interrupt
EventWait:

    SJMP    EventWait
```

8.2 Timer 1 (8 Bits)

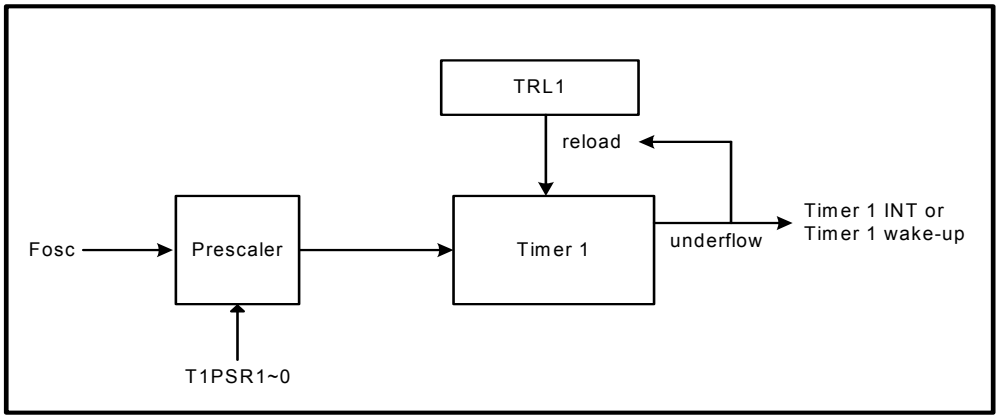


Figure 8-2 Timer 1 Function Block Diagram

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

A prescaler for the timer is also available. The T1PSR1 ~ T1PSR0 bits of TR01CON register determine the pre-scale ratio and generate different clock rates as clock source for the timer. Setting T1WKEN bit of TR01CON 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 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 value will be automatically reloaded into the 8-bit counter.

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

8.2.1 Timer 1 Register

■ **TRL1 (R26h): 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 value will be automatically reloaded into the 8-bit counter.

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

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

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

0: Disable all interrupt

1: Enable all un-masked interrupt

■ **TR01CON (R23h): Timer 0 and Timer 1 Control Register**

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

Bit 7 (T1WKEN): 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 6 (T1EN): Timer 1 enable control bit

0: Disable Timer 1 (stop counting)

1: Enable Timer 1

Bit 5 ~ Bit 4 (T1PSR1 ~ T1PSR0): Timer 1 prescaler select bits

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

■ **INTCON (R21h): Timer Interrupt Control Register**

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

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

0: Disable Interrupt function

1: Enable Interrupt function

■ **INTSTA (R22h): Timer Interrupt Status Register**

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

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

■ **Code Example:**

```
; === Timer 1 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR1I,Q_Time
     BC      INTSTA,TMR1I
     BTG     Port A,7
Q_Time:
     POP
     RETI
; === Timer1 = 32.768K / [256 x (3Fh + 1)]
Timer1SR:
      :
     PA.7 setting output pin
      :
     MOV     A,#10110000B
     MOV     TR01CON,A           ; Fosc & Pre-scale 1:256 & wakeup
     MOV     A,#03FH
     MOV     TRL1,A              ; 0.5sec=[256x(63+1)]/32.768K
     BS      TR01CON,T1EN        ; Timer 1 enable
     BS      INTCON,TMR1IE       ; Timer 1 interrupt enable
     BC      INTSTA,TMR1I        ; Clear Timer 1 interrupt status
     BS      CPUCON,GLINT        ; Enable global interrupt
     BS      CPUCON,MS1          ; Idle mode
T1Wloop:
     SLEP
     NOP
      :
     SJMP    T1Wloop
```

8.3 Timer 2 (8 Bits)

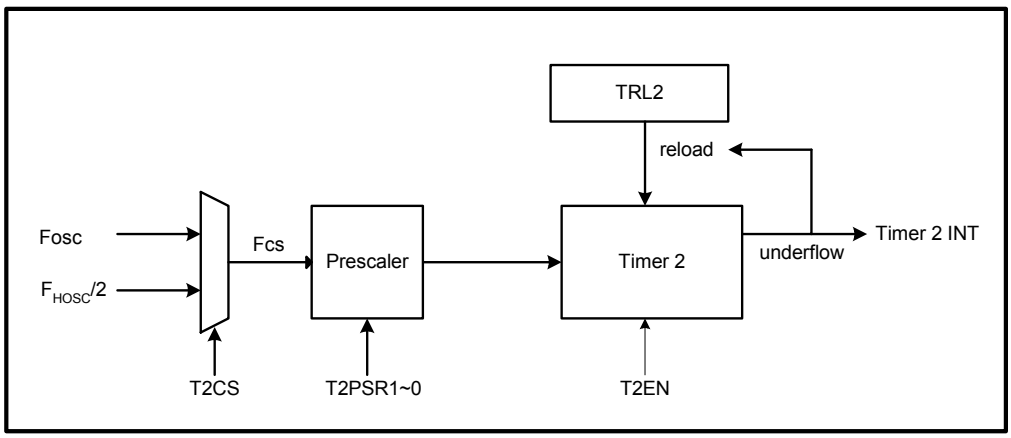


Figure 8-3 Timer 2 Function Block Diagram

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).

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

8.3.1 Timer 2 Registers

■ **TRL2 (R28h): 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 value will automatically reload into the 8-bit counter.

■ **TR2WCON (R27h): Timer 2/Watchdog Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTEN	T0ENMD	WDTPSR1	WDTPSR0	**T2EN**	**T2CS**	**T2PSR1**	**T2PSR0**

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 (R34h): MCU Control Register**

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

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

0: Disable all interrupts

1: Enable all un-masked interrupts

■ **INTCON (R21h): Timer Interrupt Control Register**

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

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

0: Disable Interrupt function

1: Enable Interrupt function

■ **INTSTA (R22h): Timer Interrupt Status Register**

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

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

■ **Code Example:**

```
; === Timer 2 interrupt
TIMERINT:
     PUSH
     JBC     INTSTA,TMR2I,Q_Time
     BC      INTSTA,TMR2I
     BTG     Port A,7
Q_Time:
     POP
     RETI
; === Timer2 = (1/32.768K) X [4 x (FFh + 1)]
Timer2SR:
       :
     PA.7 setting output pin
       :
     MOV     A,#00000010B
     MOV     TR2CON,A            ; Fosc & Pre-scale=1:4
     MOV     A,#0XFF
     MOV     TRL2,A              ; 31.25ms=[4x(255+1)]/32768
     BS      TR2CON,T2EN         ; Timer 2 Enable
     BS      INTCON,TMR2IE       ; Timer 2 Interrupt Enable
     BC      INTSTA,TMR2I        ; Clear Timer 2 Interrupt Status
TMR2Loop:
     SJMP    TMR2Loop
```

8.4 Watchdog Timer (WDT)

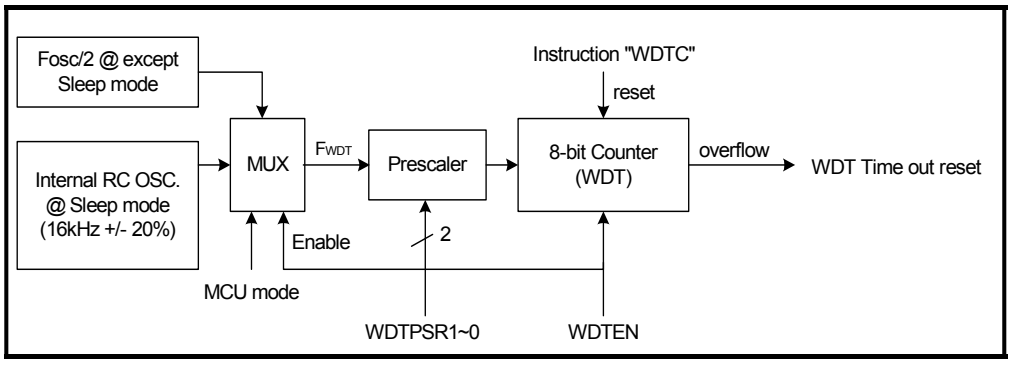


Figure 8-4 Watchdog Timer Function Block Diagram

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). Therefore 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 the WDT is disabled. When WDT is enabled, its time-out will cause the MCU to reset. The "WDTC" instruction should be used 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 (R27h): Timer 2/Watchdog Timer Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTEN	T0ENMD	**WDTPSR1**	**WDTPSR0**	T2EN	T2CS	T2PSR1	T2PSR0

Bit 7 (WDTEN): Watchdog Timer enable bit

0: Disable watchdog timer (stop running)

1: Enable watchdog timer

Bit 5 ~ Bit 4 (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
      :
     Timer1 (0.5sec wakeup)
      :
     BS      TR2WCON,WDTPSR1
     BS      TR2WCON,WDTPSR0       ; Pre-scale 1:128
     BC      CPUCON,MS1            ; Change to sleep mode
     WDTC
     SLEP
WDT_Loop:
     SJMP    WDT_Loop
```

```
; === Timer 1 interrupt 0.5 sec
TIMERINT:
     PUSH
     JBC    INTSTA,TMR1I,Q_Time
     BC     INTSTA,TMR1I
     WDTC
      :
      :
Q_Time:
     POP
     RETI
```

8.5 Input/Output Key

- Three pins key input (Port A.2 ~ 0) and 16 pins key strobe (shared with LCD segment) can achieve a maximum of 48 keys matrix with automatic key scan
- Four pins key input (Port A.7 ~ 4) and 16 pins key strobe (shared with LCD segment) can achieve a maximum of 64 keys matrix without automatic key scan
- Automatic key scan or software key scan
- Interrupt available under automatic key scan mode (SCAN=1)
- Wake-up available when key input falling edge is detected under automatic key scan mode (SCAN=1).

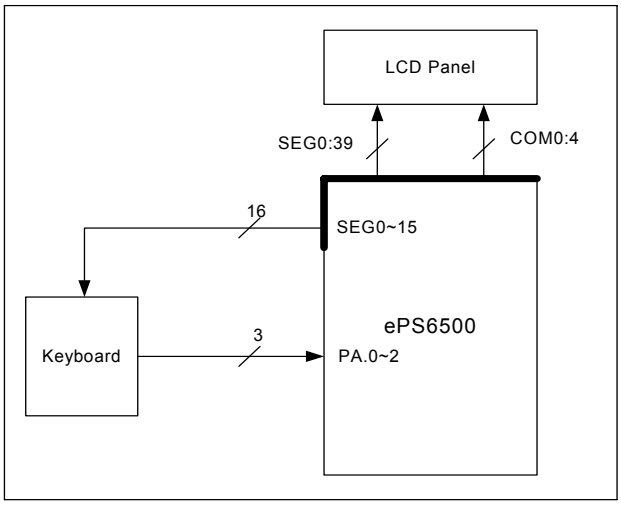


Figure 8-5a Key Function Block Diagram

As shown in the circuit diagram below, it is assumed that the key strobe output has resistance RON, while each key has resistance KON and capacitance C. A long strobe output duration will cause the LCD display to malfunction. Hence, strobe output time should be made as short as possible. Therefore, RIN (pull-up resistance) should be low enough to allow quick charge to capacitor. On the contrary, RIN should be high enough for VIN to be considered as "L" level (RIN >> RON + KON). Therefore, the value of RIN should remain changeable.

The following is the normal key input processing:

1. Output the strobe signal
2. Pull up the input port by the lowest resistance (both R1 and R2 enabled). Capacitance is charged quickly.
3. Pull up the input port by the highest resistance (only R2 is enabled)
4. Read the key
5. Disable the pull-up resistance
6. Stop the strobe signal

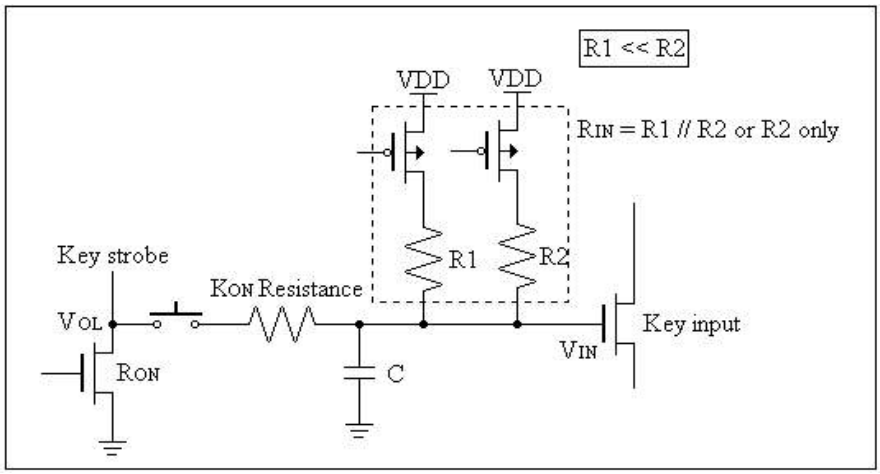


Figure 8-5b Key Circuit Diagram

8.5.1 Key Functions

SCAN	KEL	R1EN	PA2PU ~ PA0PU	IEN[1]	Total Pull-up Resistor	Port A.0 ~ 2	Note
0	0	×	×	0	Floating	High-Z	-
	1	0	0	1	**Floating**	**Floating**	**Prohibited**
		0	1	1	R2	PA.0~.2	-
		1	0	1	R1	PA.0~.2	-
		1	1	1	R1 // R2 [2]	PA.0~.2	-
1	×	0	0	0	Floating	High-Z	A[3]
		1	1	0	R1//R2 [2]	High-Z	B[3]
		0	1	1	R2	PA.0~ 2	C[3]

× : Don't care

[1] *Internal signal. Refer to the Automatic Key Scan Timing Diagram (Figure 8-6) below.*

[2] *R1 // R2 = R1R2 / (R1+R2).*

[3] *Sub clock signal. Refer to the Automatic Key Scan Timing Diagram (Figure 8-6) below.*

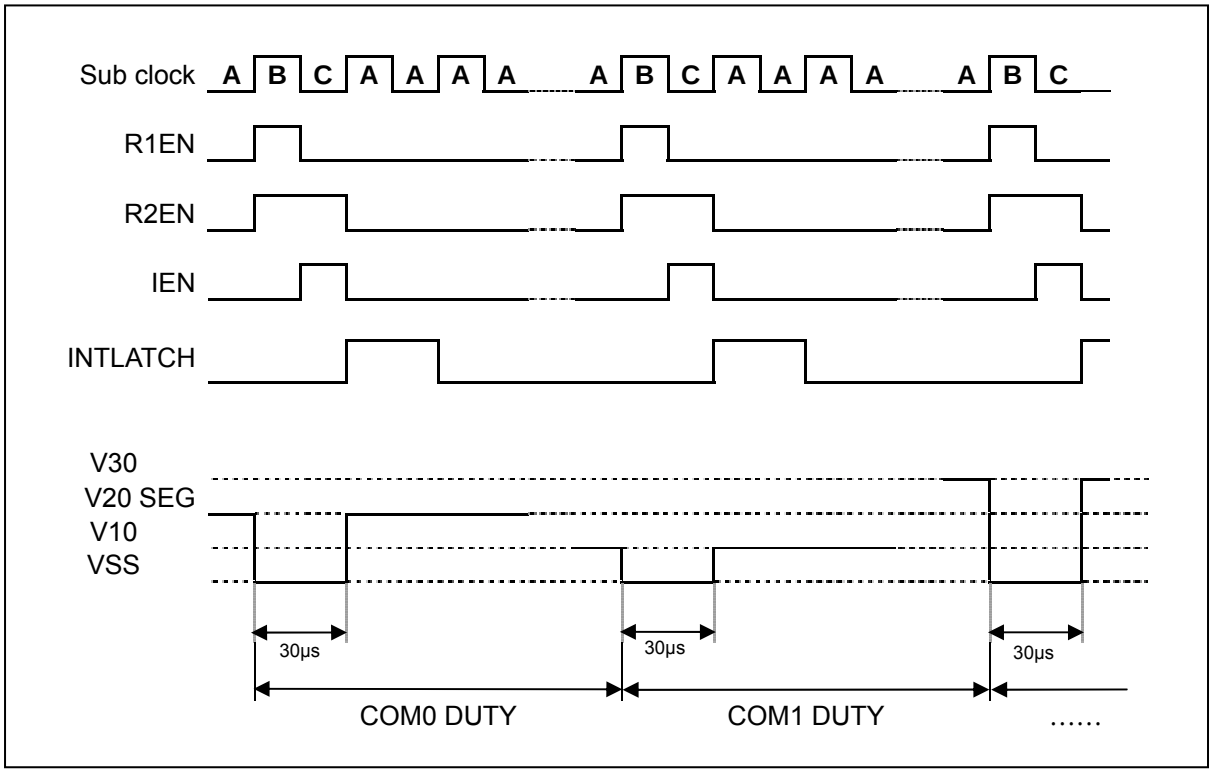


Figure 8-6 Automatic Key Scan Timing (SCAN = 1)

8.5.2 Key Strobe

The key strobe pin shares with the LCD segment pin in the CPU embedded with LCD driver model. When sharing with LCD segment, strobe output should be as short as possible to prevent LCD display error.

There are two ways to output a strobe signal, i.e., by Automatic Key scan and by software Key scan as explained in the following sections.

8.5.2.1 Automatic Key Scan

The LCD waveform has a 30 μs low pulse at the beginning of every common duty signal by setting the SCAN bit of the STBCON register. The strobe timing is shown in the following Figure 8-7 on Automatic Key Scan Strobe Signal.

When in automatic key scan mode, Bits 2 ~ 0 of PAINT or PAWAKE must be enabled. During key scan, wake up and interrupt will occur if any of the falling edge of the key input pins (Port A.2 ~ Port A.0) is detected.

NOTE

The "SCAN" bit turn-on/off in the key loop should be avoided.

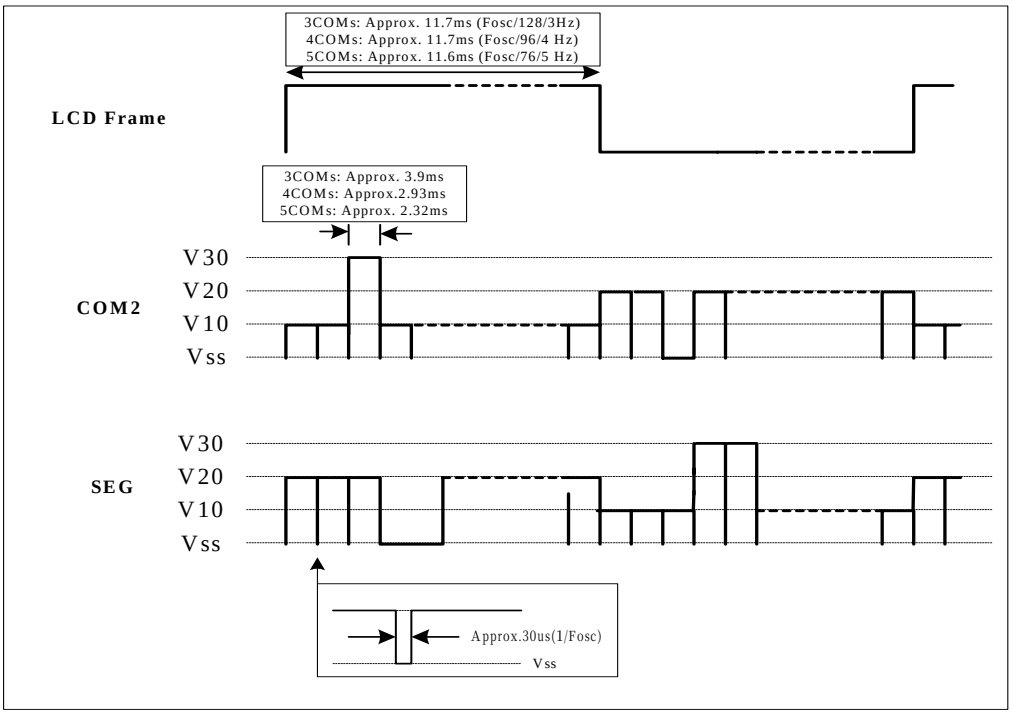


Figure 8-7 Automatic Key Scan Strobe Signal (SCAN = 1)

8.5.2.2 Software Key Scan

The segment is switched to strobe signal temporarily by setting the BitST bit of the STBCON register to "**1**" and the SCAN bit to "**0**." Set the STB3 ~ STB0 bits of the STBCON register to select which pin will be strobed.

■ **In Idle Mode**

During automatic key scanning, if any of the falling edge of the PA.0 ~ 2 pins is detected (when PAINTEN=1), wake-up will occur. Then the CPU runs and interrupt is triggered (if enabled).

■ **In Slow Mode or Fast Mode**

Both automatic and software key scans are applicable.

- Automatic key scan is used to determine "whether any key is pressed." If a key is pressed, PA.0~2 pin falling edge is detected, and then interrupt is triggered.
- Software key scan is used to determine "which key was pressed."

8.5.2.3 Key Strobe Pin Function

STBCON			Key Strobe (Shared with Segments 0 ~ 15)																LCD	
SCAN	BitST	STB3~0	Seg 0	Seg 1	Seg 2	Seg 3	Seg 4	Seg 5	Seg 6	Seg 7	Seg 8	Seg 9	Seg 10	Seg 11	Seg 12	Seg 13	Seg 14	Seg 15	Seg 16:n-1	Com 0:m-1
0	0	××××	Display waveform																	
0	1	0000	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	Display waveform	
		0001	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
		0010	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1		
		0011	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1		
		0100	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1		
		0101	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1		
		0110	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1		
		0111	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1		
		1000	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1		
		1001	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1		
		1010	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1		
		1011	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1		
		1100	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1		
		1101	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1		
		1110	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1		
		1111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0		
1	×	××××	Display waveform with automatic key scan																	

8.5.3 Input/Output Key Registers

■ **DCRA (R2Dh): 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	-	-	-	-

Bit 7 ~ Bit 4 (PA7DC ~ PA4DC): PortA.4~Port A.7 direction control bits

0: Set to output pin

1: Set to input pin

■ **Port A (R10h): Port A Register**

Bit 2 ~ Bit 0: The input structure and two-stage pull-up resistor are controlled together by PA2PU ~ PA0PU bits of the PACON register and R1EN, KEL bits of the STBCON register (see below).

Bit 3: The input structure and two-stage pull up resistor are controlled together by PA3PU bit of the PACON register and R1EN bit of the STBCON register (see below).

Bit 7~ Bit 4: Port A.7 ~ 4 are input ports which are selected by PA7DC~PA4DC bits of DCRA register (see above)

The input structure and one-stage pull-up resistor are controlled together by PA7PU ~ PA4PU bits of the PACON register and KEH bits of the LCDCON register (see below).

■ **PACON (R29h): 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 3 ~ Bit 0 (PA3PU ~ PA0PU): Pull-up resistor (R2 large resistor) control bits

0: Disable Port A.0 ~ Port A.3 pull-up resistor

1: Enable Port A.0 ~ Port A.3 pull-up resistor

Bit 7 ~ Bit 4 (PA7PU ~ PA4PU): Pull-up resistor control bits

0: Disable Port A.4 ~ Port A.7 pull-up resistor

1: Enable Port A.4 ~ Port A.7 pull-up resistor

■ **LCDCON (R32h): LCD Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
KEH	BLANK	LCDCON	–	LCR1	LCR0	–	LBVON

Bit 7 (KEH): High nibble key input enable/disable control bit

0: Disable High nibble key input function

1: Enable High nibble key input function

■ **PAINTEN (R2Bh): 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 (R2Ch): 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 pin falling edge is detected, and

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

■ **STBCON (R20h): Strobe Output Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCAN	KEL	R1EN	BitST	STB3	STB2	STB1	STB0

Bit 7 (SCAN): Key scan or specify the scan signal bit by bit

0: Key scan specified as Bit STB3 ~ 0 defined

1: Auto strobe scanning

Bit 6 (KEL): Low nibble key input enable/disable control bit

0: Disable Low nibble key input function (Port A.0~2 do NOT correspond with Key input in software scan mode)

1: Enable Low nibble key input function (PORTA.0~2 correspond with Key input in software scan mode)

Bit 5 (R1EN): R1 pull up resistor (small resistor) control bit for Port A.3 ~ Port A.0

0: Disable R1 pull-up resistor

1: Enable R1 pull-up resistor

Bit 4 (BitST): Enable bit strobe

0: Display waveform

1: Strobe signal specified as STB3 ~ 0 as defined.

Bit 3 ~ Bit 0 (STB0 ~ STB3): Strobe output selector bits

■ **Code Example:**

```
; Key Matrix 1 (Port A and Ground):
;  ===  Sleep mode
PAIN_SR:
      :
; --- Port A 0~7 input pins
     MOV    A,#0XF0
     MOV    DCRA,A
; --- Port A wakeup
     MOV    A,#11111111B
     MOV    PAWAKE,A
; --- R1EN & R2EN Pull-up & KE enable
     MOV    A,#0XFF
     MOV    PACON,A
     BS     STBCON,R1EN
     BS     STBCON,KEL
     BS     LCDCON,KEH
; --- Port A interrupt enable
     MOV    A,#11111111B
     MOV    PAINTEN,A
     CLR    PAINTSTA
     BS     CPUCON,GLINT
; --- Sleep MODE
     BC     CPUCON,MS1
PAINloop:
     SLEP
     NOP
       :
     SJMP   PAINloop
```

```
; *** Interrupt PortA data
INPTINT:
     PUSH
     MOVRP  A,PAINTSTA
     MOV    Key_No,A
     CLR    PAINTSTA
     POP
     RETI
```

```
; Key matrix 2 (Port A.0~2 and SEG0 ~ SEG15):
; *** Key scan function
      :
     LCD display setting
      :
     MOV    A,#0X0F
     OR     PACON,A            ; R2EN enable
     MOV    PAWAKE,A           ; Port A setting wakeup function
      :
; === Idle mode auto key scan routine
     BS     STBCON,SCAN        ; Auto-key scan enable
     BS     CPUCON,MS1         ; Idle mode
KeyIdle:
     SLEP
     NOP
       :
```

(Continuation)

```
; === Key scan routine
KeyScan:
     CLR    STBCON              ; Auto-key scan disable
KeyLoop:
     BS     STBCON,R1EN         ; R1EN enable
     BS     STBCON,KEL          ; Key enable
     BS     STBCON,BitST        ; Strobe ON
     LCALL  DLY50US
     BC     PACON,R1EN          ; R1EN disable
     MOVL   A,Port A            ; Port A input data
     BC     STBCON,BitST        ; Strobe OFF
     BC     PACON,KEL           ; Key disable
     JLE    A,#0X0E,KeyScan     ; If A <= PORTA Goto KeyScan
     INC    STBCON
     SJMP   KeyLoop
KeyScan:
; --- Clear key number
     CLR    Key_No
;  --- Key Scan is finish
KeyScanOk:
     MOV   Key_No,A
     MOV   A,STBCON
     MOVH  Key_No,A             ; Key_No: XXXX XXXX
      :
```

8.6 LCD Driver

The ePS6500 provides directly driven LCD. It supports multiplexed drive for 40SEGs × 5COMs 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. The LCD voltage is 2xVDD. The LCD frame rate is as follows.

Duty	LCD Frame Rate
1/3	Approx. 11.7ms (Fosc/128/3 Hz)
1/4	Approx. 11.7ms (Fosc/96/4 Hz)
1/5	Approx. 11.6ms (Fosc/76/5 Hz)

This embedded LCD driver generates waveforms to drive the display.

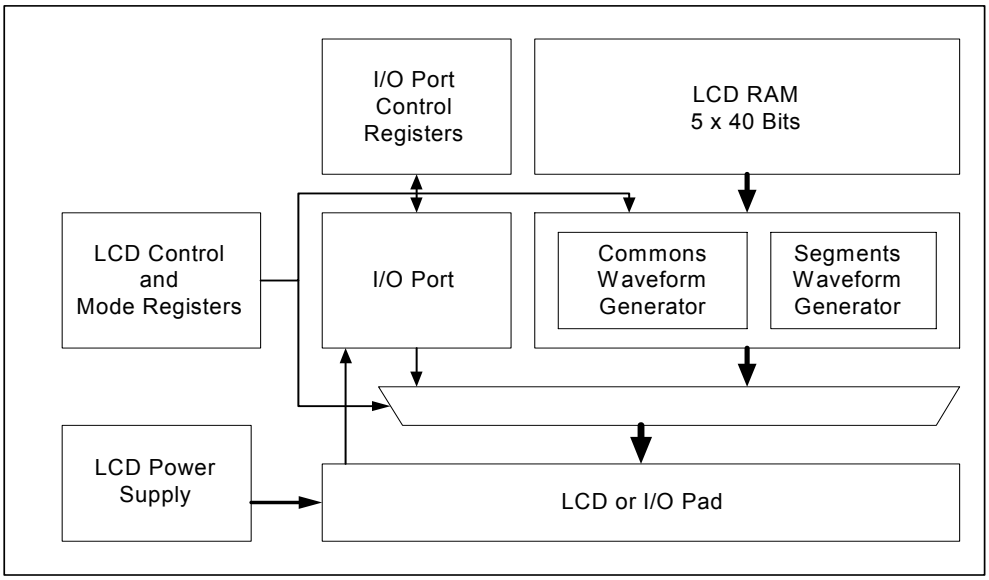


Figure 8-8a LCD Driver Function Block Diagram

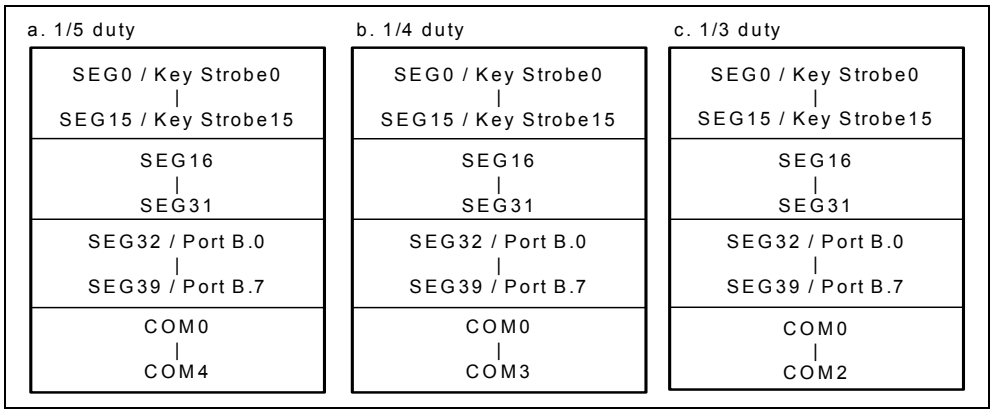


Figure 8-8b LCD Pin Configuration

8.6.1 LCD Driver Registers

■ **LCDCON (R32h): LCD Control Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
KEH	**BLANK**	**LCDCON**	–	**LCR1**	**LCR0**	–	**LBVON**

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 3, Bit 2 (LCR1, LCR0): LCD Bias Voltage Charge-pump Rate select bits

LCR1:LCR0	Charge-Pump Rate(Hz)
00	8K
01	4K
10	2K
11	16K

Bit 0 (LBVON): Bias Voltage Charge-pump control bit

0: Disable

1: Enable

■ LCDARL (R09h): 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

Bit 7~6 (LCDARL7~6): unknown value. (Note: Do not use JDNZ at LCDARL (09h) special register)

■ LCDDATA (R0Eh): LCDDATA Register

This 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).

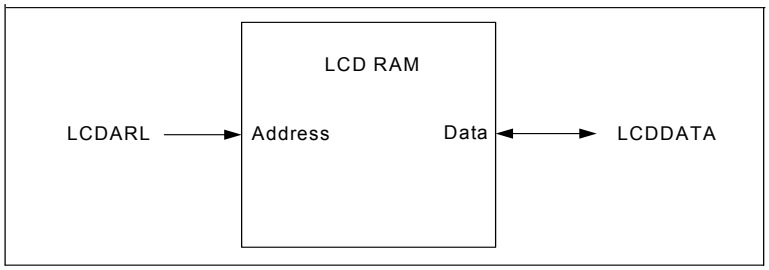


Figure 8-9 LCDDATA Register Access through LCD RAM

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

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

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=8K
     MOV    A,#00000001B
     MOV    LCDCON,A
     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
DspRAMd1:
     CLR    LCDARL
     TBPTH  #0x14
; === Write LCD RAM is dot matrix
WrLRAMd:
     MOV    A,#0XAA
     MOV    LCDDATA,A
     MOV    A,#0X55
     MOV    LCDDATA,A
     JDNZ   TABPTRH, WrLRAMd
     CLR    LCDARL
     RET
```

8.6.2 LCD RAM Map

■ **1/5 Duty**

RAM Address LCDARL		COM 0	COM 1	COM 2	COM 3	COM 4	-	-	-
		Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
SEG0	00H								
:	:								
SEG39	27H								

■ **1/4 Duty**

RAM Address LCDARL		COM 0	COM 1	COM 2	COM 3	-	-	-	-
		Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
SEG0	00H								
:	:								
SEG39	27H								

■ **1/3 Duty**

RAM Address LCDARL		COM 0	COM 1	COM 2	-	-	-	-	-
		Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
SEG0	00H								
:	:								
SEG39	27H								

8.6.3 LCD Driving Method Circuit

■ **1/3 Bias**

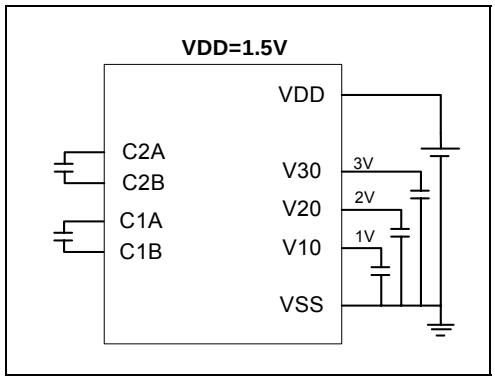


Figure 8-10a LCD Driving Method Circuit for 1/3 Bias

■ **1/2 Bias**

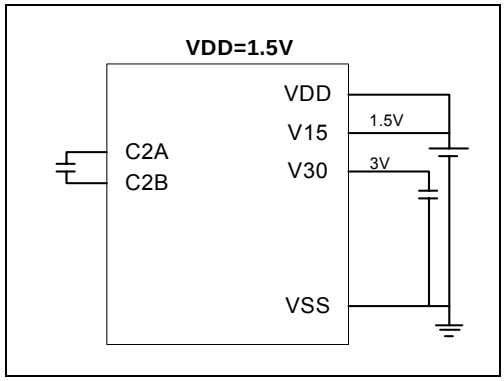


Figure 8-10b LCD Driving Method Circuit for 1/2 Bias

8.6.4 LCD COM Waveforms

8.6.4.1 For 1/5 Duty and 1/3 Bias

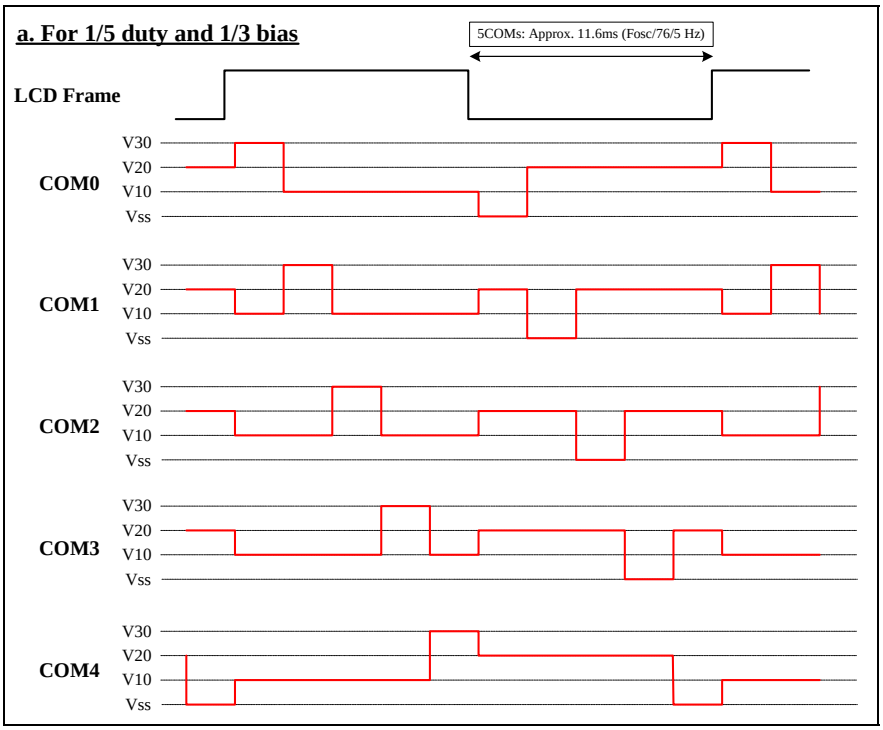


Figure 8-11 LCD COM Waveform for 1/5 Duty and 1/3 Bias

8.6.4.2 For 1/4 Duty and 1/3 Bias

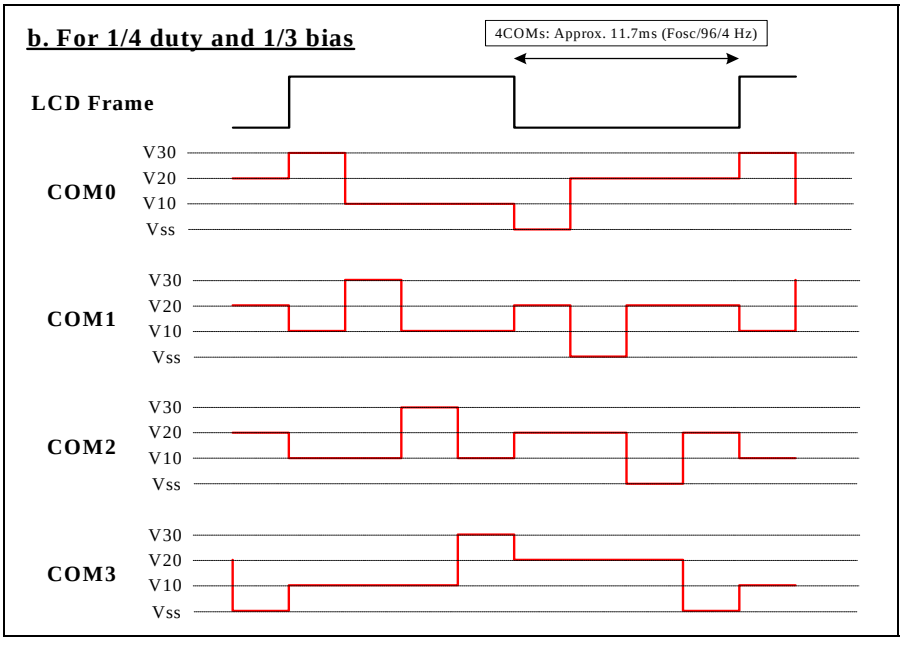


Figure 8-12 LCD COM Waveform for 1/4 Duty and 1/3 Bias

8.6.4.3 For 1/3 Duty and 1/2 Bias

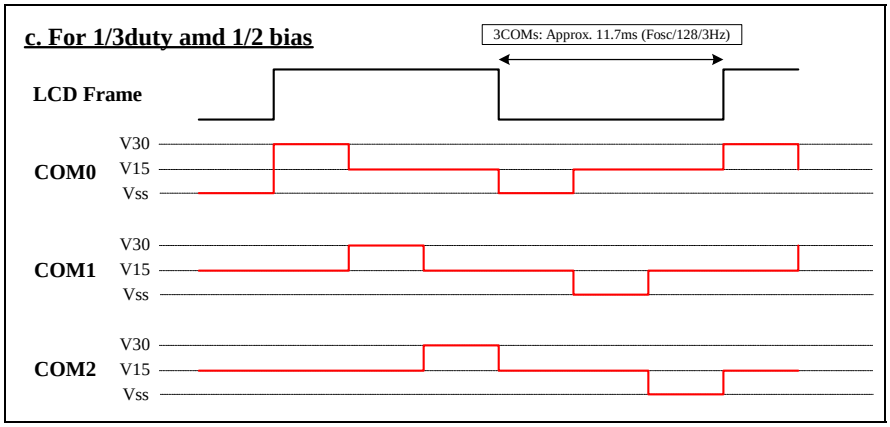


Figure 8-13 LCD COM Waveform for 1/3 Duty and 1/2 Bias

8.6.5 LCD COM and SEG Waveforms

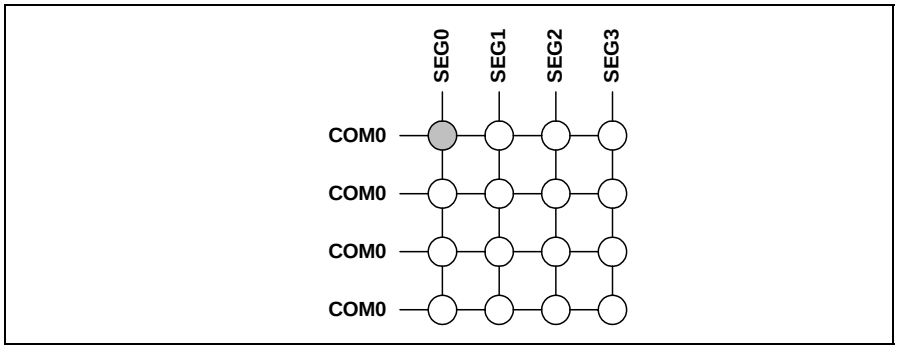


Figure 8-14 LCD COM & SEG Waveform Matrix

8.6.5.1 For 1/5 Duty and 1/3 Bias

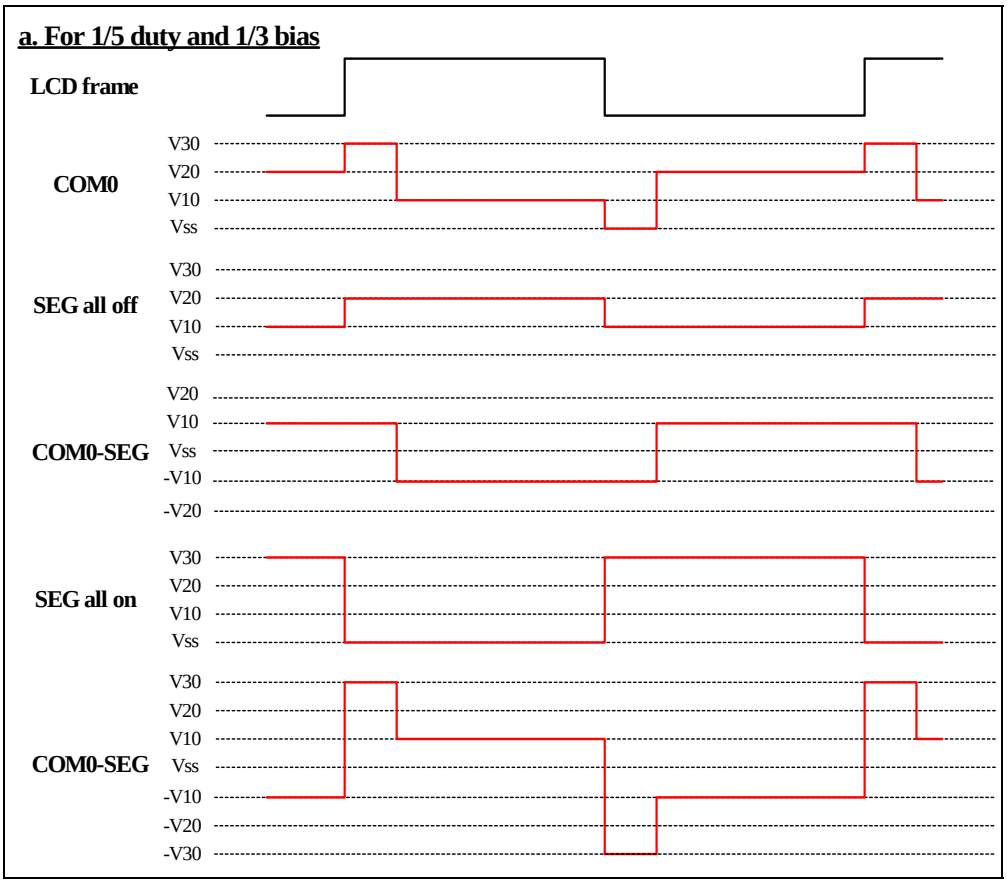


Figure 8-15 LCD COM & SEG Waveform for 1/5 Duty and 1/3 Bias

8.6.5.2 For 1/4 Duty and 1/3 Bias

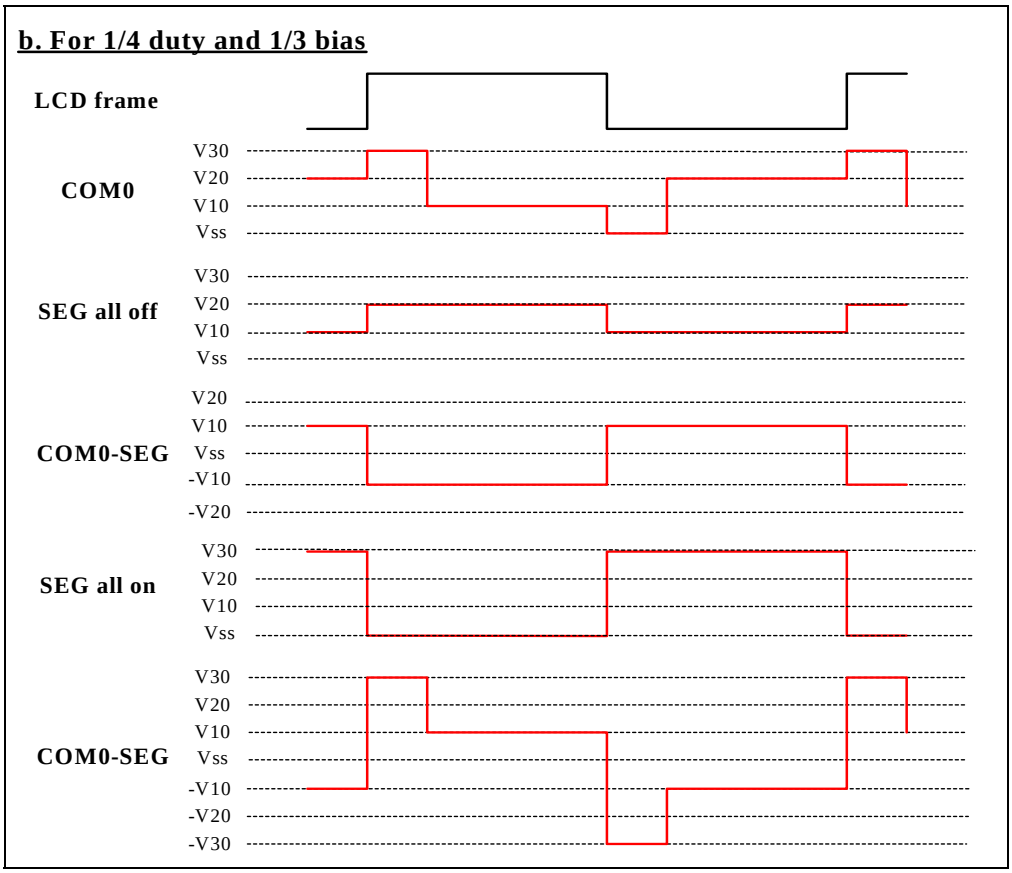


Figure 8-16 LCD COM and SEG Waveform for 1/4 Duty and 1/3 Bias

8.6.5.3 1/3 Duty and 1/2 Bias

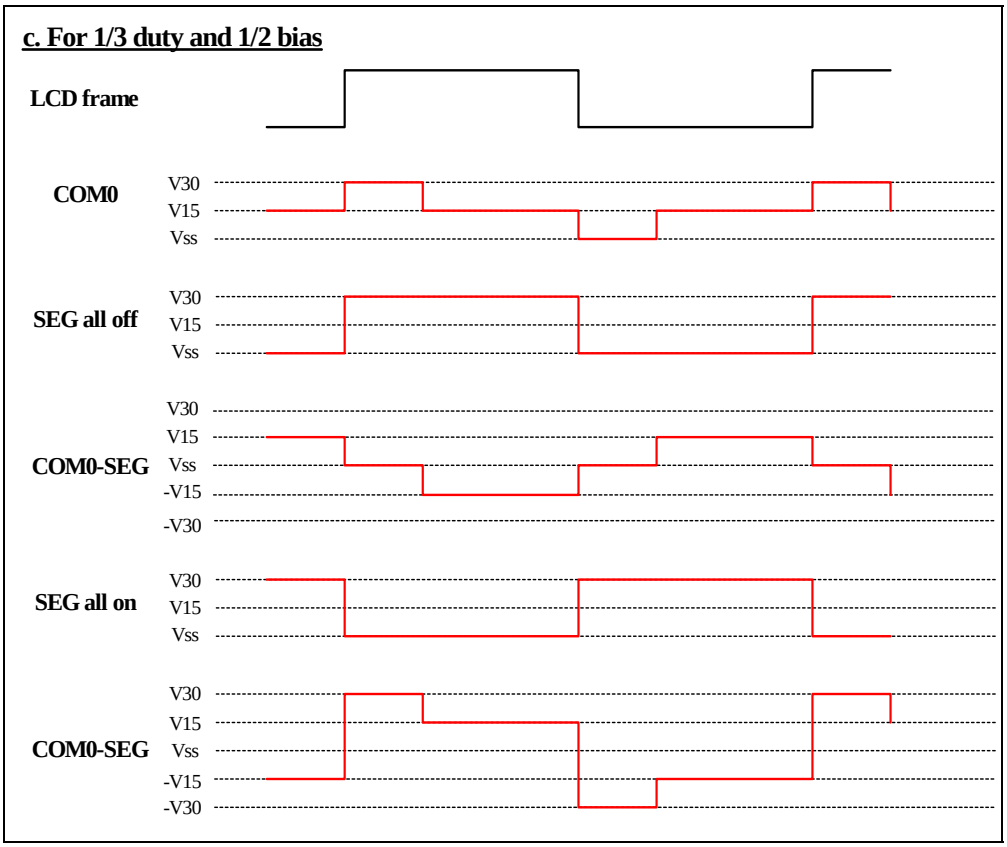


Figure 8-17 LCD COM and SEG Waveform for 1/3 Duty and 1/2 Bias

9 Electrical Characteristics

9.1 VDD=1.5V Electrical Characteristics

■ Absolute Maximum Ratings

Items	Sym.	Condition	Limits	Unit
Supply voltage	VDD	-	–0.3 to +2.0	V
Input voltage (general input port)	VIN	-	–0.5 to VDD +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
Input voltage	VIH	-	VDD x 0.9 to VDD	V
	VIL	-	0 to VDD x 0.1	
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=1.5MΩ	140	200	260	kHz
			RC OSC. R=1MΩ	210	300	390	
			RC OSC. R=560KΩ	350	500	650	
	Fosc	Sub-clock frequency	Internal RC OSC	24.6	32.8	41	kHz
			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 enabled, no load	-	2	3	
	Idd3	Slow mode	VDD=1.5V, RC / Crystal OSC, LCD disabled, no load	-	3	4	
	Idd4		VDD=1.5V, RC / Crystal OSC, LCD enabled, no load	-	3	5	
	Idd5	Fast mode	VDD=1.5V, F_{HOSC} =200kHz, LCD enabled, no load	-	15	20	
	Idd6		VDD=1.5V, F_{HOSC} =300kHz, LCD enabled, no load	-	20	30	
	Idd7		VDD=1.5V, F_{HOSC} =500kHz, LCD enabled, no load	-	30	50	
Input Voltage	VIH1	PA [0:7], PB [0:7] (as general input port)		VDD×0.7	-	VDD	V
	VIL1			0	-	VDD×0.3	

(Continuation)

Parameter	Sym.	Condition		Min.	Typ.	Max.	Unit
Input Threshold Voltage (Schmitt)	VT+	RSTB		0.5×VDD	-	0.75×VDD	V
	VT–			0.2×VDD	-	0.4×VDD	
Input Leakage Current	IIL	ALL Input port (without pull up/down resistor) Vin= VDD or GND		-	-	±1	μA
Large Pull-up Resistance	RPU5	RSTB	Vin=GND	300	450	800	KΩ
Small Pull-up Resistance	RPU6	RSTB	Vin=1V	10	30	60	KΩ
Large Pull-down Resistance	RPD1	TEST	Vin=VDD	250	500	800	KΩ
Small Pull-down Resistance	RPD2	TEST	Vin=0.5V	3	6	12	KΩ
Data Retention Voltage	Vret	-		1.2	-	-	V
Output Current	IOH1	PA [4:7], PB [0:7] (as general output port)	VDD=1.5V, VOH=1.2V, LCD enabled	-0.1	-0.4	-0.8	mA
	IOL1		VDD=1.5V, VOL=0.2V, LCD enabled	0.6	0.9	1.5	
Large Pull-up Resistance	RPU1	PA [0:7]	Key high resistance, pulled up by R2, LCD enabled, Vin2=0.5V	110	280	800	KΩ
	RPU3	PB [0:7]	Vin=0.5V, LCD enabled	100	320	800	
Small Pull-up Resistance	RPU2	PA [0:3]	Key high resistance, pulled up by R2//R1, LCD enabled, Vin2=0 V	6	18	60	KΩ
	RPU4	PA [4:7]	Vin=1V, LCD enabled	15	30	75	
LCD Driver							
LCD Display Output ON-resistance	ROC	Com [0:4]	VOH=V30 ± 0.2V	0.35	0.4	0.5	KΩ
			VOM=V20 ± 0.2V	0.55	0.65	0.75	
			VOM=V10 ± 0.2V	0.35	0.40	0.45	
			VOL=0.2V	0.25	0.3	0.35	
	ROS	Seg [0:39]	VOH=V30 ± 0.2V	0.35	0.4	0.5	KΩ
			VOM=V15 ± 0.2V	0.45	0.6	0.85	
			VOL=0.1V	0.25	0.3	0.35	
Strobe Output ON-resistance	ROP	Seg [0:15] (as key strobe)	V=VDD-0.2V	80	200	500	KΩ
	RON		V=0.2V	0.85	1.1	1.3	

10 Pin Type Circuit Diagrams

■ Reset Pin Type

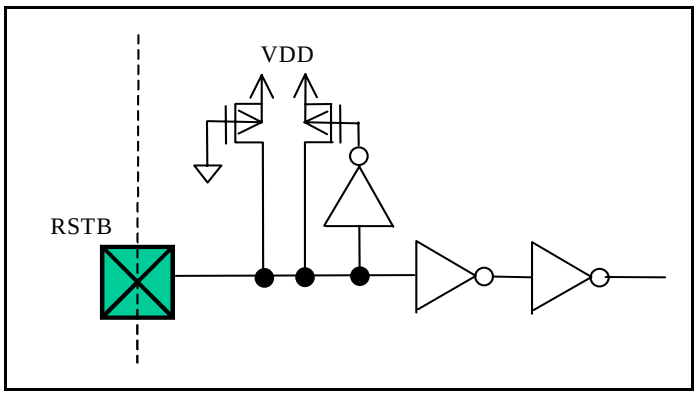


Figure 10-1a Reset Pin Type Circuit Diagram

■ Test Pin Type

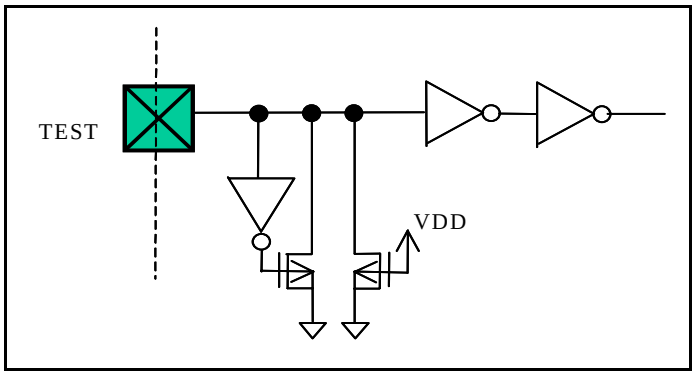


Figure 10-1b Test Pin Type Circuit Diagram

■ Oscillator Pin Type

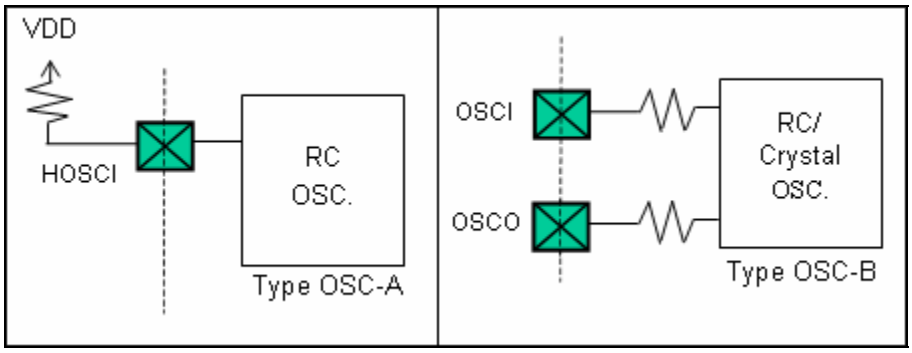


Figure 10-1c Oscillator Pin Type Circuit Diagram

■ **Input Pin Type (PA.0~3)**

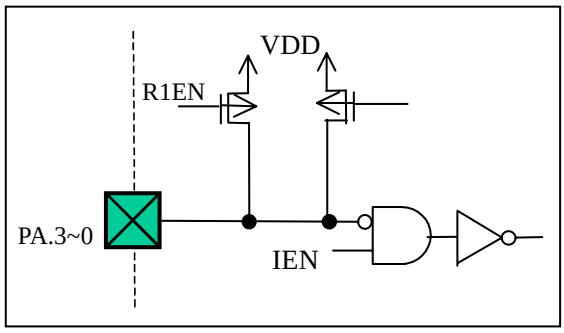


Figure 10-1d Input Pin Type Circuit Diagram

■ **I/O Pin Type (PA.4~7)**

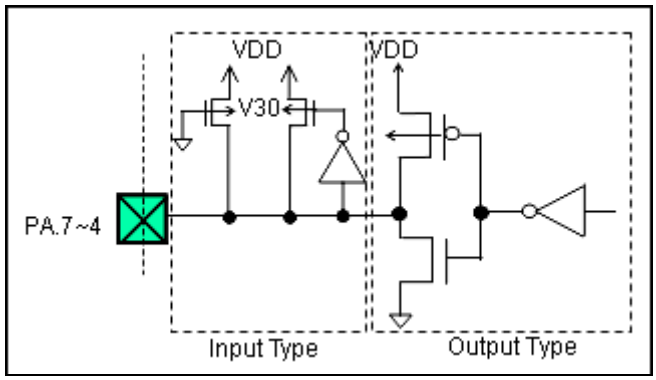


Figure 10-1e I/O Pin Type Circuit Diagram

■ **SEG and I/O Share Pin Type**

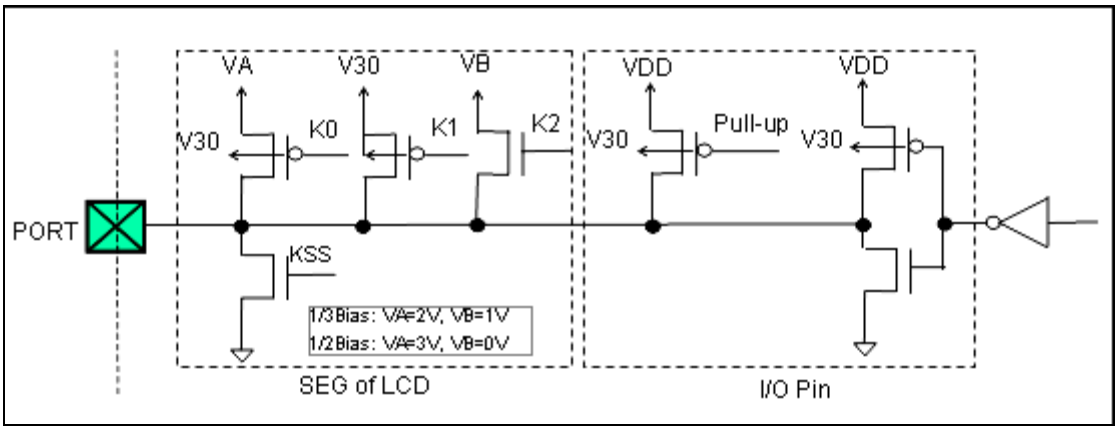


Figure 10-1f SEG and I/O Share Pin Type Circuit Diagram

■ **General SEG and COM Pin Type**

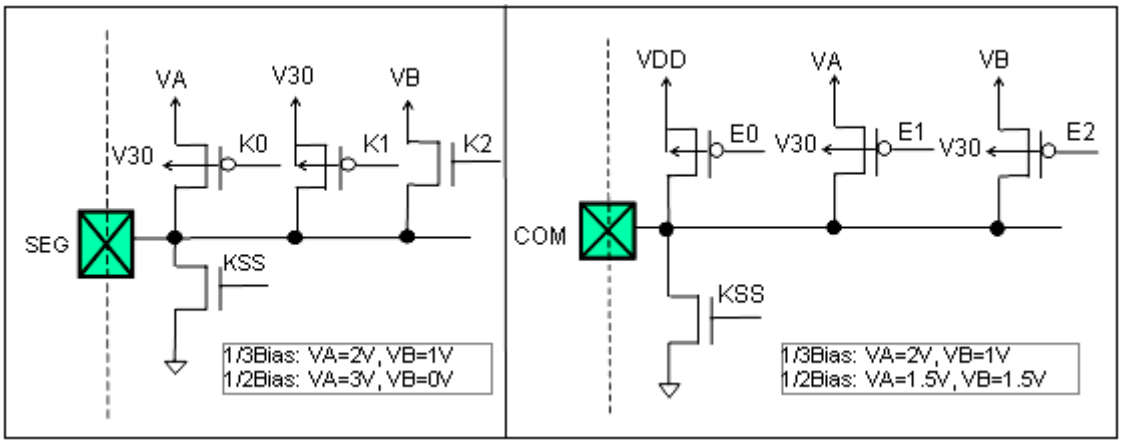


Figure 10-1g General SEG and COM Share Pin Type Circuit Diagram

11 Application Circuits

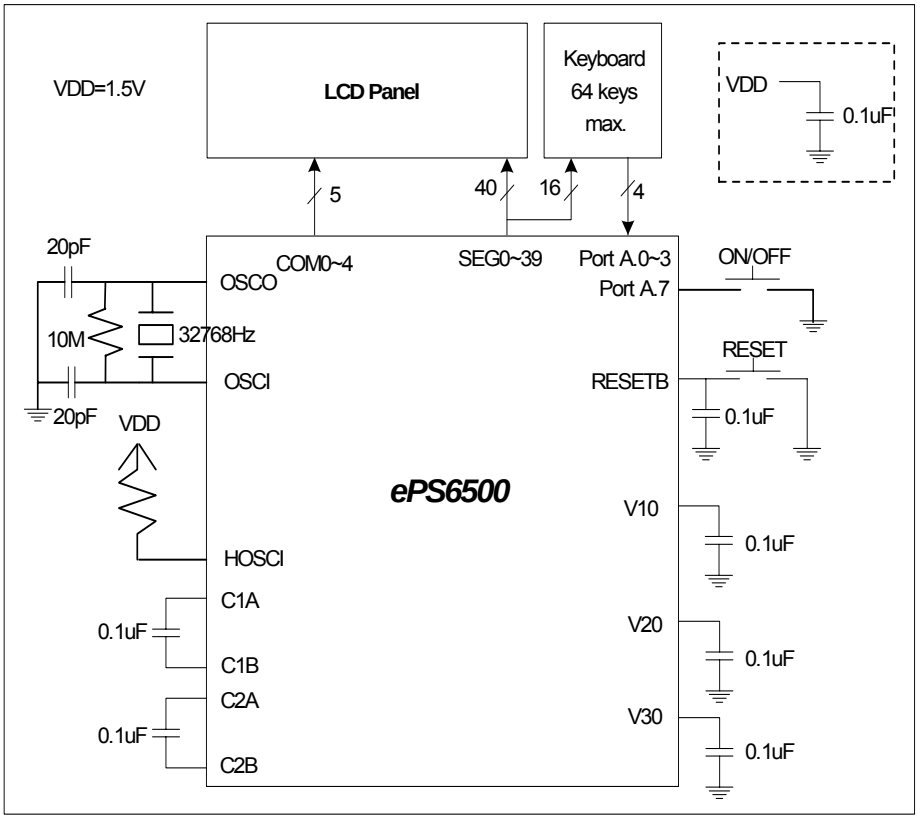


Figure 11-1 VDD=1.5 Application Circuit Diagram

12 Instruction Set

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

Type	Binary Instruction	Mnemonic	Operation	Status Affected	Cycle
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 *1	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 0000 00aa aaaa aaaa aaaa	LCALL addr (two 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 0000 00aa aaaa aaaa aaaa	LJMP addr (two words)	PC ← addr	None	2
Compare & 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 *2	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 *2	None	2
	0100 0111 kkkk kkkk aaaa aaaa aaaa aaaa	JGE A,#k,addr	Jump to addr if A≧k PC[15:0] ← addr *2	None	2
	0100 1000 kkkk kkkk aaaa aaaa aaaa aaaa	JLE A,#k,addr	Jump to addr if A≦k PC[15:0] ← addr *2	None	2
	0100 1001 kkkk kkkk aaaa aaaa aaaa aaaa	JE A,#k,addr	Jump to addr if A=k PC[15:0] ← addr *2	None	2
	0101 0101 rrrr rrrr aaaa aaaa aaaa aaaa	JGE A,r,addr	Jump to addr if A≧r PC[15:0] ← addr *2	None	2
	0101 0110 rrrr rrrr aaaa aaaa aaaa aaaa	JLE A,r,addr	Jump to addr if A≦r PC[15:0] ← addr *2	None	2
	0101 0111 rrrr rrrr aaaa aaaa aaaa aaaa	JE A,r,addr	Jump to addr if A=r PC[15:0] ← addr *2	None	2
Bit Compare & Jump	0101 1bbb rrrr rrrr aaaa aaaa aaaa aaaa	JBC r,b,addr	If r(b)=0,jump to addr PC[15:0] ← addr *2	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 *2	None	2

(Continuation)

Type	Binary Instruction	Mnemonic	Operation	Status Affected	Cycle
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 Table Look Up	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)] *3, *4	None	2
	0010 1111 rrrr rrrr	TBRD A,r	r←ROM[(TABPTR+ACC)] *4	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
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 *5	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

(Continuation)

Type	Binary Instruction	Mnemonic	Operation	Status Affected	Cycle
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
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 S0CALL addressing ability is from 0x000 to 0xFFF (4K words space)

*2 The maximum jump range is 8K words absolute address

*3 TBRD i, r :
r← ROM [(TABPTR)]
i=00: TABPTR no change
wi=01: TABPTR←TABPTR+1
i=10: TABPTR←TABPTR-1

*4 TABPTR = (TABPTRM: 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 table look up space is internal 16K bytes (8K words).*

*5 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.

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

13 Pad Diagram

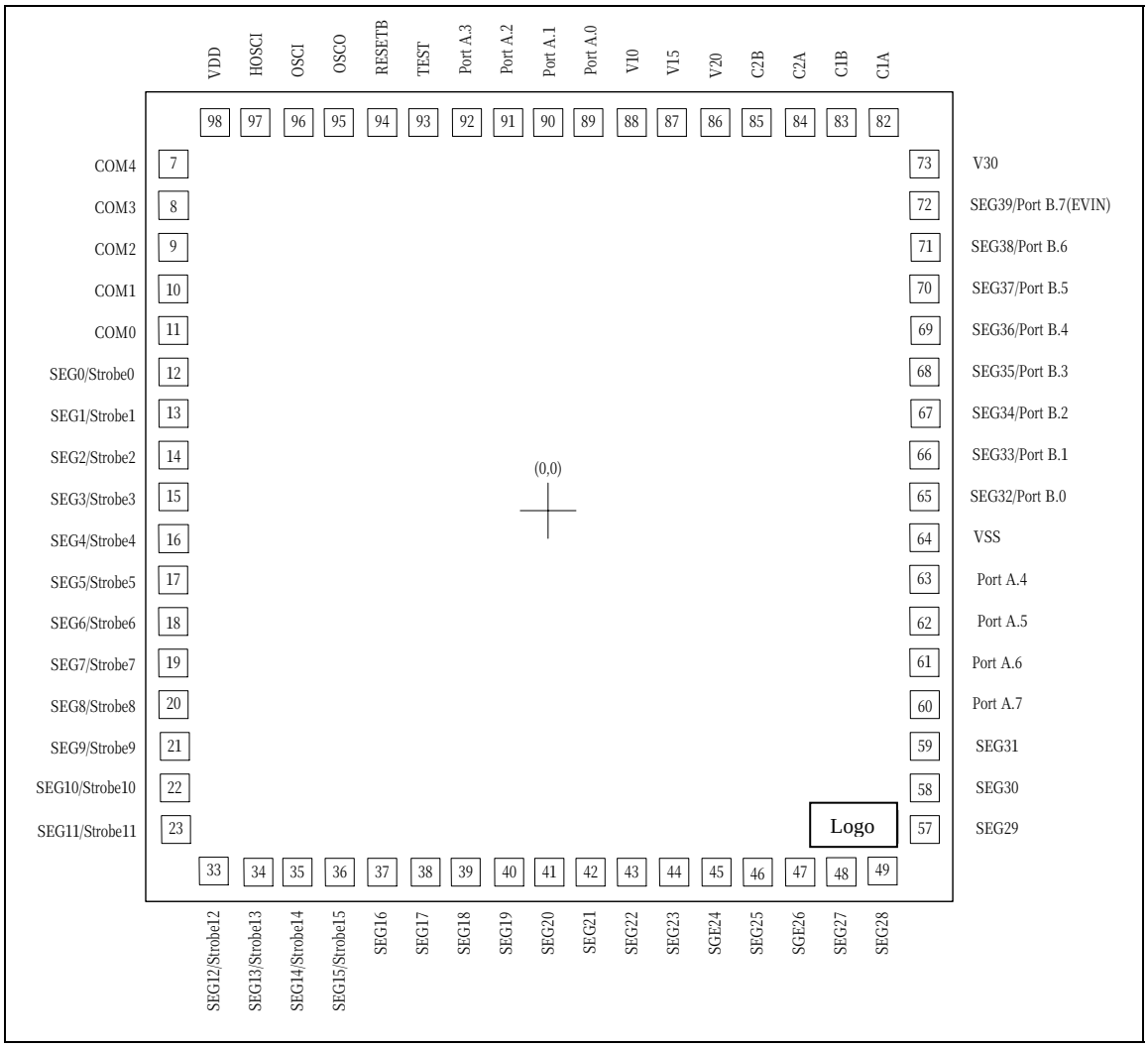


Figure 13-1 ePS6500 Pad Diagram

14 Pad Coordinates

■ Chip Size: 1820 × 1940 μm^2

Pin No.	Symbol	X	Y	Pin No.	Symbol	X	Y
1	NC			41	SEG20	-62.05	-855.6
2	NC			42	SEG21	27.95	-855.6
3	NC			43	SEG22	117.95	-855.6
4	NC			44	SEG23	207.95	-855.6
5	NC			45	SEG24	297.95	-855.6
6	NC			46	SEG25	387.95	-855.6
7	COM4	-795.6	719.55	47	SEG26	477.95	-855.6
8	COM3	-795.6	629.55	48	SEG27	567.95	-855.6
9	COM2	-795.6	539.55	49	SEG28	657.95	-855.6
10	COM1	-795.6	449.55	50	NC		
11	COM0	-795.6	359.55	51	NC		
12	SEG0/Strobe0	-795.6	269.55	52	NC		
13	SEG1/Strobe1	-795.6	179.55	53	NC		
14	SEG2/Strobe2	-795.6	89.55	54	NC		
15	SEG3/Strobe3	-795.6	-0.45	55	NC		
16	SEG4/Strobe4	-795.6	-90.45	56	NC		
17	SEG5/Strobe5	-795.6	-180.45	57	SEG29	795.6	-842.55
18	SEG6/Strobe6	-795.6	-270.45	58	SEG30	795.6	-752.55
19	SEG7/Strobe7	-795.6	-360.45	59	SEG31	795.6	-662.55
20	SEG8/Strobe8	-795.6	-450.45	60	PortA.7	787.5	-557.4
21	SEG9/Strobe9	-795.6	-540.45	61	PortA.6	787.5	-467.4
22	SEG10/Strobe1	-795.6	-630.45	62	PortA.5	787.5	-377.4
23	SEG11/Strobe1	-795.6	-720.45	63	PortA.4	787.5	-287.4
24	NC			64	VSS	787.5	-187.6
25	NC			65	SEG32/PorB.0	787.5	-82.25
26	NC			66	SEG33/PortB.1	787.5	15.85
27	NC			67	SEG34/PortB.2	787.5	114.15
28	NC			68	SEG35/PortB.3	787.5	212.25
29	NC			69	SEG36/PortB.4	787.5	310.55
30	NC			70	SEG37/PortB.5	787.5	408.65
31	NC			71	SEG38/PortB.6	787.5	506.95
32	NC			72	SEG39/ PortB.7	787.5	605.05
33	SEG12/Strobe1	-782.05	-855.6	73	V30	787.5	710.75
34	SEG13/Strobe1	-692.05	-855.6	74	NC		
35	SEG14/Strobe1	-602.05	-855.6	75	NC		
36	SEG15/Strobe1	-512.05	-855.6	76	NC		
37	SEG16	-422.05	-855.6	77	NC		
38	SEG17	-332.05	-855.6	78	NC		
39	SEG18	-242.05	-855.6	79	NC		
40	SEG19	-152.05	-855.6	80	NC		

(Continuation)

Pin No.	Symbol	X	Y	Pin No.	Symbol	X	Y
81	NC			91	PortA.2	-33.7	855.6
82	C1A	776.3	855.6	92	PortA.3	-123.7	855.6
83	C1B	686.3	855.6	93	TEST	-213.7	855.6
84	C2A	596.3	855.6	94	RESETB	-303.7	855.6
85	C2B	506.3	855.6	95	OSCO	-393.7	855.6
86	V20	416.3	855.6	96	OSCI	-483.7	855.6
87	V15	326.3	855.6	97	HOSCI	-573.7	855.6
88	V10	236.3	855.6	98	VDD	-673.6	847.5
89	PortA.0	146.3	855.6	99	NC		
90	PortA.1	56.3	855.6	100	NC		

NOTE

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