4-BIT FLASH EPROM MICROCONTROLLER

Table of Contents--

Publication Release Date: March 1998
Revision A2

GENERAL DESCRIPTION

The W741E260 is a high-performance 4-bit microcontroller (μC) that provides an LCD driver and the flash EEPROM for the program memory. The device contains a 4-bit ALU, two 8-bit timers, two dividers (for two oscillators) in dual-clock operation, a 32 × 4 LCD driver, and five 4-bit I/O ports (including 1 output port for LED driving). There are also five interrupt sources and 8-level subroutine nesting for interrupt applications. The W741E260 operates on very low current and has three power reduction modes, hold mode and stop mode in single-clock operation and the dual-clock slow operation, which help to minimize power dissipation.

This chip is available for W741C250 and W741C260 bodies, which can be selected by option code. The W741E260 is suitable for end product manufacturer engineering testing and earlier samples before mass production.

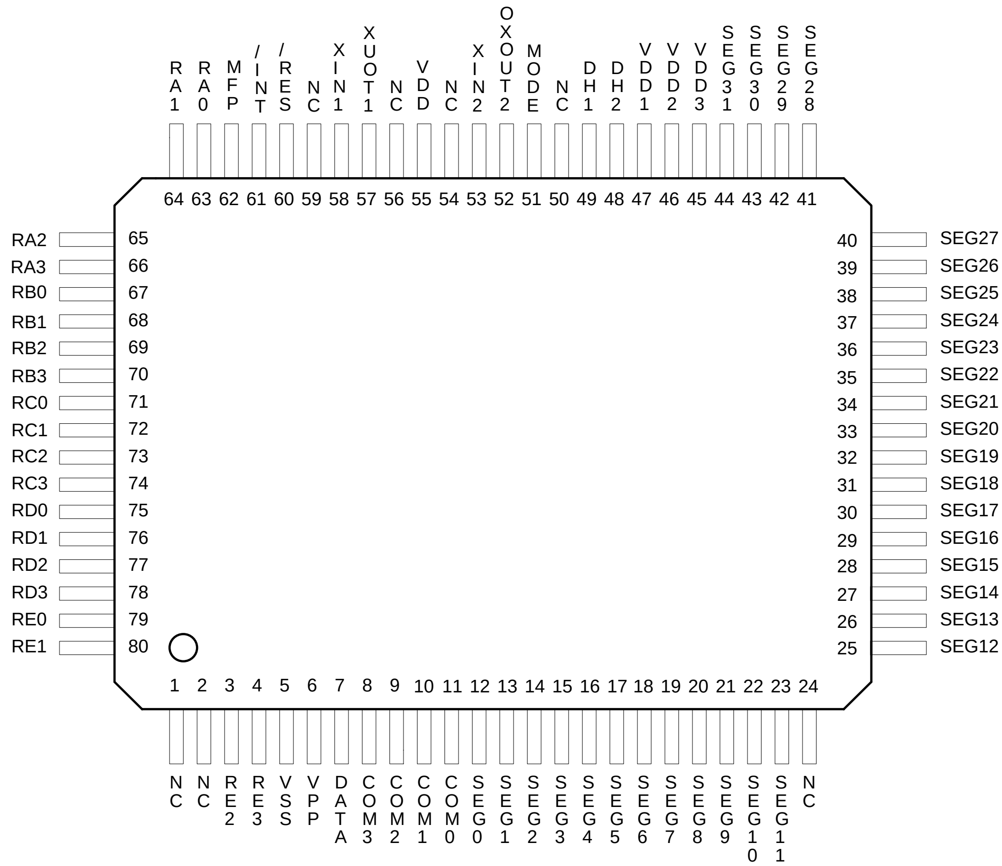
FEATURES

- Operating voltage: 2.4V to 5.5V (LCD drive voltage: 3.0V, or 4.5V)
- Crystal/Ceramic oscillator: up to 4 MHz
- RC oscillator: up to 4 MHz
- Dual-clock operation is selected by code option
- Main oscillator
  - Crystal or RC oscillation circuit can be selected by code option
  - In crystal mode, high-frequency (400 KHz to 4 MHz) or low-frequency (32.768 KHz) oscillation should be selected by code option
  - In RC mode, attention must be paid to the high/low frequency oscillation option, because the LCD driver frequency and the ROM code emulation time are related to this option.
- Sub-oscillator
  - Connect to 32768 Hz crystal only
  - Used in dual-clock operation
- Memory
  - 2048 x 16-bit program flash EEPROM (including 2K x 4-bit look-up table)
  - 128 x 4-bit data RAM (including 16 working registers)
  - 32 x 4 LCD data RAM
- 21 input/output pins
  - Ports for input only: 2 ports/8 pins
  - Input/output ports: 2 ports/8 pins
  - High sink current for LED driving: 1 port/4 pins
  - MFP output pin: 1 pin (MFP)
- Power-down mode
  - Hold function: no operation (excluding main oscillator and sub-oscillator)
  - Stop function: no operation (excluding sub-oscillator)

- Dual-clock slow operation mode: system is operated by the sub-oscillator (FOSC = Fs and Fm is stopped)

- Five types of interrupts
  - Four internal interrupts (Divider0, Divider1, Timer0, Timer1) for W741C260 body; three internal interrupts (Divider0, Timer0, Timer1) for W741C250 body.
  - One external interrupt (RC Port) for W741C260 body; two external interrupts (RC port and $\overline{\text{INT}}$ pin) for W741C250 body.
- LCD driver output
  - 32 segment x 4 common
  - Static, 1/2 duty (1/2 bias), 1/3 duty (1/2 or 1/3 bias), 1/4 duty (1/3 bias) driving mode can be selected
  - LCD driver output pins can be used as DC output port by code option
  - Clock source can be main oscillator clock in the single-clock operation mode, or sub-oscillator clock in the dual-clock operation mode; operation mode is selected by code option
- MFP output pin
  - Output is software selectable as modulating or nonmodulating frequency
  - Works as frequency output specified by Timer 1
- Two built-in 14-bit frequency dividers
  - Divider0: the clock source is the output of the main oscillator
  - Divider1: the clock source is the output of the sub-oscillator
- Two built-in 8-bit programmable countdown timers
  - Timer 0: one of two internal clock frequencies (FOSC/4 or FOSC/1024) can be selected
  - Timer 1: includes an auto-reload function and one of two internal clock frequencies (FOSC or FOSC/64) can be selected, or falling edge of pin RC.0 can be selected (output through MFP pin)
- Built-in 18/14-bit watchdog timer selectable for system reset
  - Enable/Disable the watchdog timer can be controlled by command or by option code; the control source (command or option code) can be determined by another option code
- Powerful instruction set: 118 instructions for W741C260 body
  116 instructions for W741C250 body
- 8-level subroutine (include interrupt) nesting
- Up to 1 µS instruction cycle (with 4 MHz operating frequency)
- Packaged in 80-pin QFP

PIN CONFIGURATION



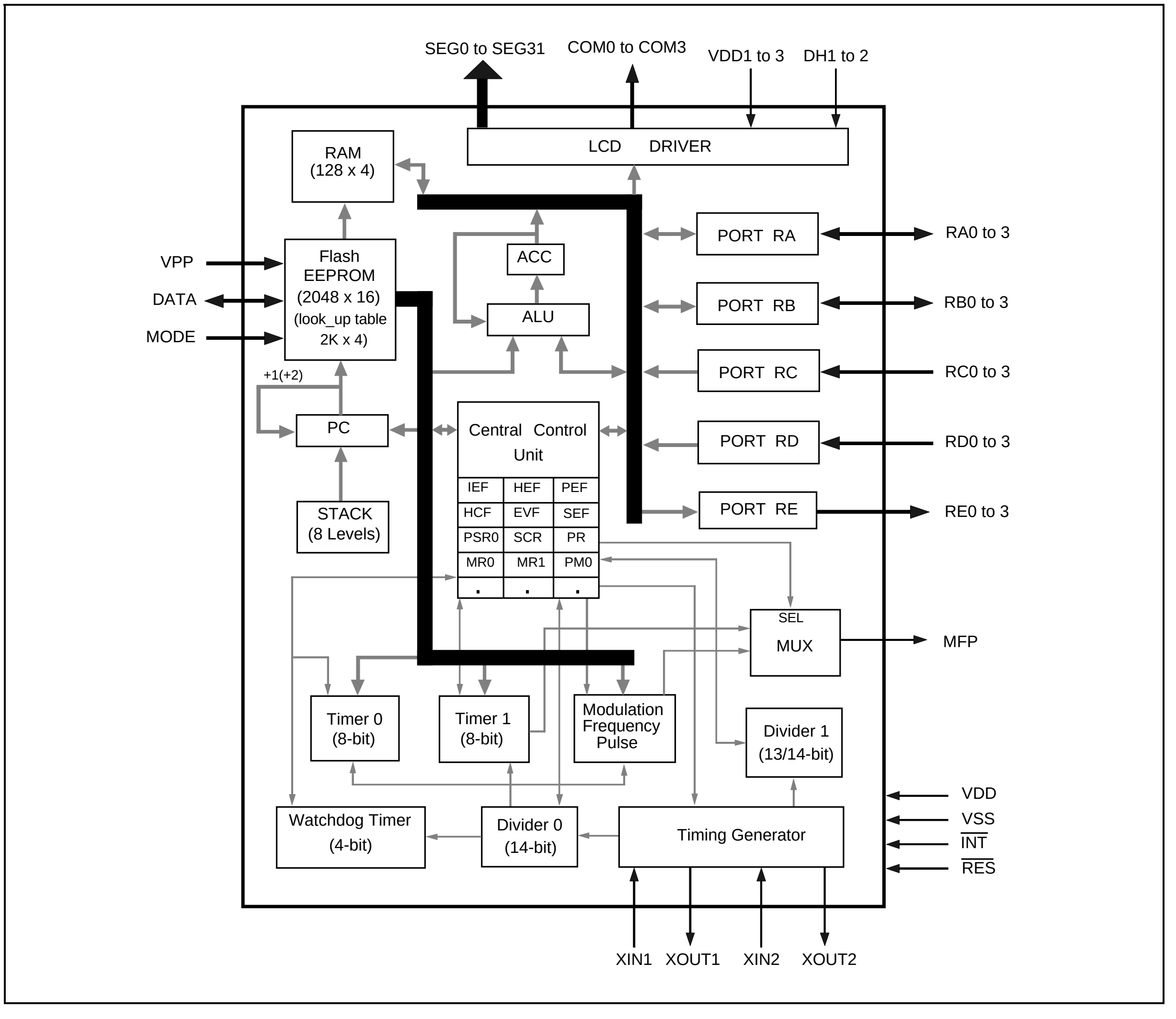
PIN DESCRIPTION

<table>
<tr><th>SYMBOL</th><th>I/O</th><th>FUNCTION</th></tr>
<tr><td>XIN1</td><td>I</td><td>Input pin for main-oscillator.
Connected to crystal or resistor to generate system clock by code option.</td></tr>
<tr><td>XOUT1</td><td>O</td><td>Output pin for main-oscillator.
Connected to crystal or resistor to generate system clock by code option.</td></tr>
<tr><td>XIN2</td><td>I</td><td>Input pin for sub-oscillator. Connected to 32.768 KHz crystal.</td></tr>
<tr><td>XOUT2</td><td>O</td><td>Output pin for sub-oscillator with internal oscillation capacitor. Connected to 32.768 KHz crystal.</td></tr>
<tr><td>RA0 to RA3</td><td>I/O</td><td>Input/Output port.
Input/output mode specified by port mode 1 register (PM1).</td></tr>
<tr><td>RB0 to RB3</td><td>I/O</td><td>Input/Output port.
Input/output mode specified by port mode 2 register (PM2).</td></tr>
<tr><td>RC0 to RC3</td><td>I</td><td>4-bit port for input only.
Each pin has an independent interrupt capability.</td></tr>
<tr><td>RD0 to RD3</td><td>I</td><td>4-bit port for input only.</td></tr>
<tr><td>RE0 to RE3</td><td>O</td><td>Output port only.
This port provides high sink current to drive LEDs.</td></tr>
<tr><td>MFP</td><td>O</td><td>Output pin only.
This pin can output modulating or nonmodulating frequency, or Timer 1 clock output specified by mode register 1 (MR1).</td></tr>
<tr><td>$\overline{RES}$</td><td>I</td><td>System reset pin with pull-high resistor.</td></tr>
<tr><td>$\overline{INT}$</td><td>I</td><td>External interrupt pin with pull-high resistor. This pin is bonding option for the W741C250 body.</td></tr>
<tr><td>SEG0 to SEG31</td><td>O</td><td>LCD segment output pins.
Also can be used as DC output ports specified by option codes.</td></tr>
<tr><td>COM0 to
COM3</td><td>O</td><td>LCD common signal output pins.
<table>
<tr><th></th><th>Static</th><th>1/2 Duty</th><th>1/3 Duty</th><th>1/4 Duty</th></tr>
<tr><td>COM0</td><td>Used</td><td>Used</td><td>Used</td><td>Used</td></tr>
<tr><td>COM1</td><td>Not Used</td><td>Used</td><td>Used</td><td>Used</td></tr>
<tr><td>COM2</td><td>Not Used</td><td>Not Used</td><td>Used</td><td>Used</td></tr>
<tr><td>COM3</td><td>Not Used</td><td>Not Used</td><td>Not Used</td><td>Used</td></tr>
</table>
The LCD alternating frequency can be selected by code option.</td></tr>
<tr><td>DH1, DH2</td><td>I</td><td>Connection terminals for voltage doubler (halver) capacitor.</td></tr>
<tr><td>VDD1,
VDD2,
VDD3</td><td>I</td><td>Positive (+) supply voltage terminal.
Refer to Functional Description.</td></tr>
</table>

Pin description, continued

SYMBOL	I/O	FUNCTION
VDD	I	Positive power supply (+).
VSS	I	Negative power supply (-).
VPP	I	Voltage control pin for the flash EEPROM programming, erasing and verifying. This pin has a built-in pull-low resistor.
MODE	I	Mode selection pin for the flash EEPROM programming, erasing and verifying. This pin has a built-in pull-low resistor.
DATA	I/O	Data I/O pin for the flash EEPROM programming and verifying. This pin has a built-in pull-low resistor.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Program Counter (PC)

Organized as an 11-bit binary counter (PC0 to PC10), the program counter generates the addresses ofthe 2048 × 16 on-chip flash EEPROM containing the program instruction. When the jump or subroutine call instructions or the interrupt or initial reset conditions are to be executed, the address corresponding to the instruction will be loaded into the program counter. The format used is shown below.

ITEM	ADDRESS	INTERRUPT PRIORITY
Initial Reset	000H	-
INT 0 (Divider0)	004H	1st
INT 1 (Timer 0)	008H	2nd
INT 2 (Port RC)	00CH	3rd
INT 3 (Divider1 for W741C260; $\overline{\text{INT}}$ pin for W741C250)	014H	4th
INT 4 (Timer 1)	020H	5th
JMP Instruction	XXXH	-
Subroutine Call	XXXH	-

Stack Register (STACK)

The stack register is organized as 11-bit x 8 levels (first-in, last-out). When either a call subroutine or an interrupt is executed, the program counter will be pushed onto the stack register automatically. At the end of a call subroutine or an interrupt service subroutine, the RTN instruction must be executed to pop the contents of the stack register into the program counter. When the stack register is pushed over the eighth level, the contents of the first level will be lost. In other words, the stack register is always eight levels deep.

Program Memory (flash EEPROM)

The flash EEPROM is used to store program codes; the look-up table is arranged as 2048 × 4 bits. The first three quarters of flash EEPROM (000H to 5FFH) are used to store instruction codes only, but the last quarter (600H to 7FFH) can store both instruction codes and the look-up table. Each look-up table element is composed of 4 bits, so the look-up table can be addressed up to 2048 elements. There are two registers (TABL and TABH) to be used in look-up table addressing and they are controlled by MOV TABH, R and MOV TABL, R instructions. When the instruction MOVC R is executed, the contents of the look-up table location address specified by TABH, TABL and ACC will be read and transferred to the data RAM. Refer to the instruction table for more details. The organization of the program memory is shown in Figure 1.

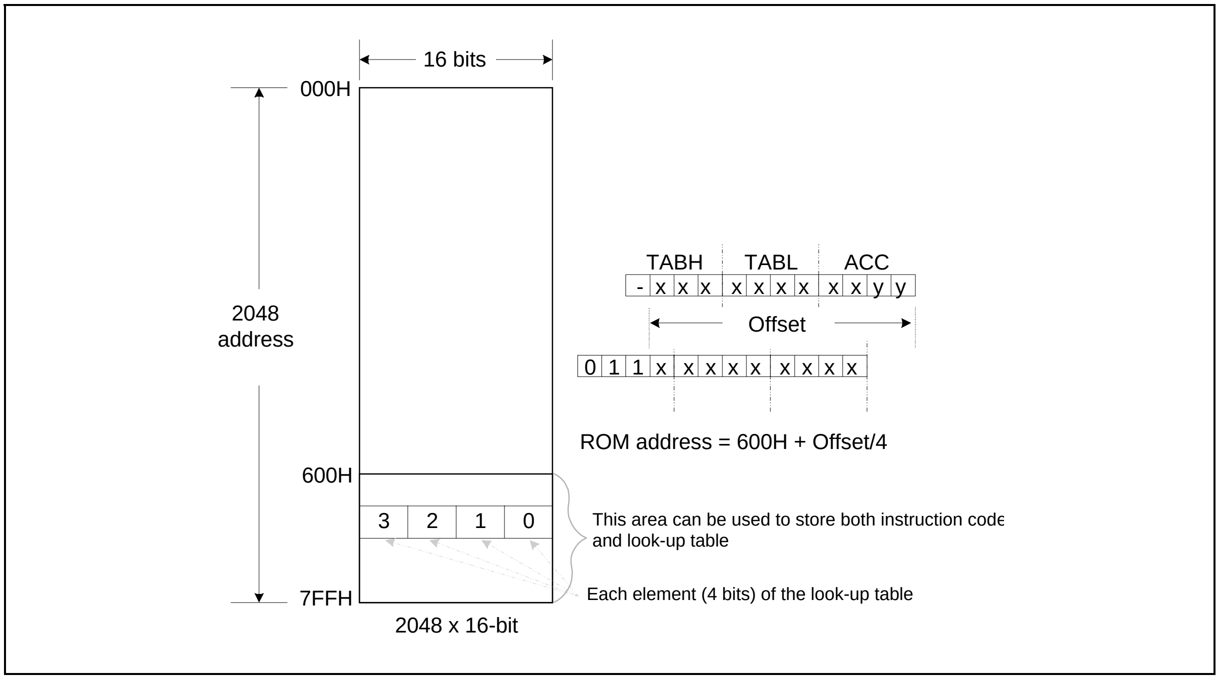


Figure 1. Program Memory Organization

Data Memory (RAM)

1. Architecture

The static data memory (RAM) used to store data is arranged as 128 × 4 bits. The data memory can be addressed directly or indirectly. The organization of the data memory is shown in Figure 2.

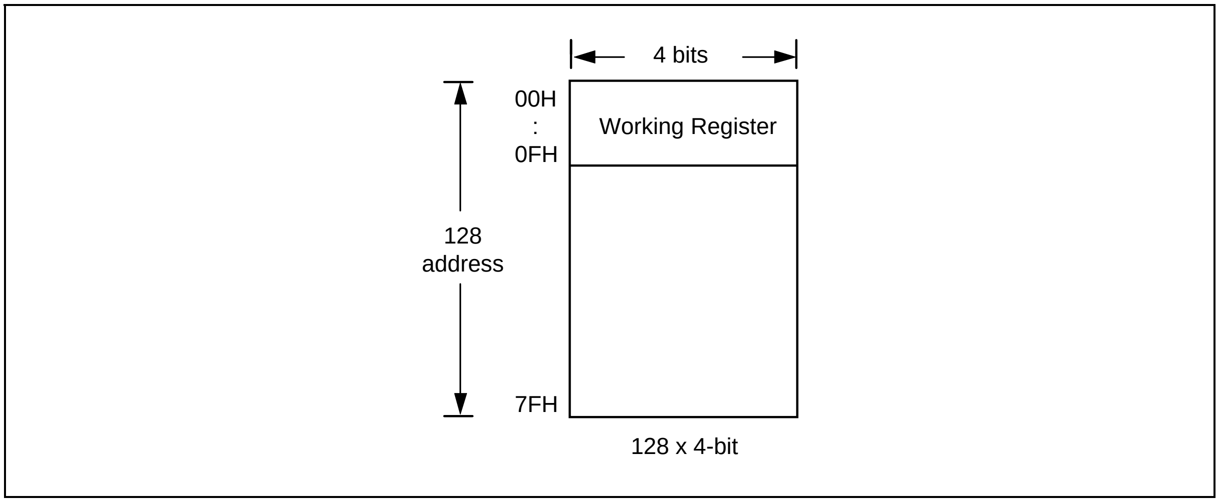


Figure 2. Data Memory Organization

The first sixteen addresses (00H to 0FH) in the data memory are known as the working registers (WR). The other data memory is used as general memory and cannot operate directly with immediate data. The relationship between data memory locations and the page register (PAGE) in indirect addressing mode is described in the next section.

2. Page Register (PAGE)

The page register is organized as a 4-bit binary register. The bit descriptions are as follows:

	3	2	1	0
PAGE	–	R/W	R/W	R/W

Note: R/W means read/write available.

Bit 3 is reserved.

Bit 2, Bit 1, Bit 0 Indirect addressing mode preselect bits:

000 = Page 0 (00H - 0FH)
001 = Page 1 (10H - 1FH)
010 = Page 2 (20H - 2FH)
011 = Page 3 (30H - 3FH)
100 = Page 4 (40H - 4FH)
101 = Page 5 (50H - 5FH)
110 = Page 6 (60H - 6FH)
111 = Page 7 (70H - 7FH)

Accumulator (ACC)

The accumulator (ACC) is a 4-bit register used to hold results from the ALU and transfer data between the memory, I/O ports, and registers.

Arithmetic and Logic Unit (ALU)

This is a circuit which performs arithmetic and logic operations. The ALU provides the following functions:

- Logic operations: ANL, XRL, ORL
- Branch decisions: JB0, JB1, JB2, JB3, JNZ, JZ, JC, JNC, DSKZ, DSKNZ, SKB0, SKB1, SKB2, SKB3
- Shift operations: SHRC, RRC, SHLC, RLC
- Binary additions/subtractions: ADC, SBC, ADD, SUB, ADU, DEC, INC

After any of the above instructions are executed, the status of the carry flag (CF) and zero flag (ZF) is stored in the internal registers. Otherwise CF can be stored or be read out by executing MOVA R, CF or MOV CF, R.

Clock Generator

The W741E260 provides two oscillation circuits, main-oscillator and sub-oscillator. The main-oscillator can select the crystal or RC oscillation circuit by option codes to generate the system clock through external connections. If a crystal oscillator is used, a crystal or a ceramic resonator must be connected to XIN1 and XOUT1, and a capacitor must be connected if an accurate frequency is needed. When the oscillator is used, a high-frequency clock (400 KHz to 4 MHz) or low-frequency clock (32 KHz) can be selected for the system clock by means of option codes. If the RC oscillator is used, a resistor must be connected to XIN1 and XOUT1, and the high/low frequency clock option must be selected to suit the operation frequency. The sub-oscillator must be connected to a 32.768 KHz crystal through XIN2 and XOUT2 external pins when the dual-clock operation mode is selected by option code. The connection is shown in Figure 3. One machine cycle consists of a four-state system clock sequence and can run up to 1 µS with a 4 MHz system clock.

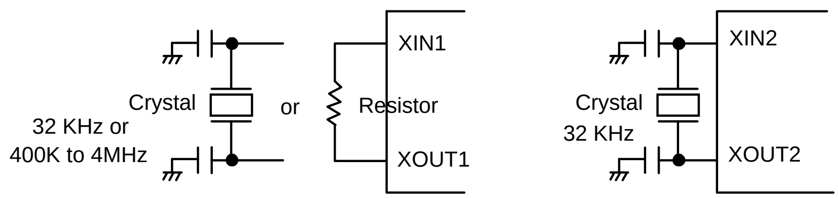


Figure 3. System Clock Oscillator Configuration

Dual-clock operation

This operation mode is selected by code option. In the dual-clock mode, the clock source of the LCD frequency selector should be the sub-oscillator clock (32768 Hz) only. But in the single-clock mode, the clock source of the LCD frequency selector will be Fm or Fm/32 (Fm: main oscillator clock). So when the STOP instruction is executing, the LCD will be turned off in the single-clock mode; but the LCD will keep working in the dual-clock mode.

In this dual-clock mode, the normal operation is performed by generating the system clock from the main-oscillator clock (Fm). As required, the slow operation can be performed by generating the system clock from the sub-oscillator clock (Fs). The exchange of the normal operation and the slow operation is performed by resetting or setting the bit 0 of the system clock control register (SCR). If the SCR.0 is reset to 0, the clock source of the system clock generator is the main-oscillator clock; if the SCR.0 is set to 1, the clock source of the system clock generator is the sub-oscillator clock. In the dual-clock mode, the main-oscillator can stop oscillating when the STOP instruction is executing or the SCR.1 is set to 1. But in the single-clock mode, only the STOP instruction can stop the main-oscillator oscillating, because the SCR would be disabled in the single-clock mode. Therefore, in single-clock mode, the clock source of the system clock generator is the main-oscillator clock (FOSC = Fm).

When the SCR is set or reset, we must pay attention to the following:

1. X000B → X011B: Disable the main-oscillator (Fm) should not be done simultaneously with changing the system clock source (FOSC) from Fm to Fs. The FOSC should be changed first from Fm to Fs before the main-oscillator (Fm) is disabled. The correct sequence is: X000B→X001B→X011B.

2. X011B → X000B: Enabling the main-oscillator (Fm) should not be done simultaneously with changing the FOSC from Fs into Fm. The main-oscillator (Fm) should be enabled first before a delay subroutine is called to allow the main-oscillator to oscillate stably. The FOSC can now be changed from Fs into Fm. The correct sequence is therefore X011B→X001B→delay subroutine→X000B.

   The suggested delay for Fm is 20 mS for 455 KHz ceramic resonator and 10 mS for 4 MHz crystal.

We must remember that the X010B state is inhibitive, because it will induce a system shutdown.

The organization of the dual-clock operation mode is shown below.

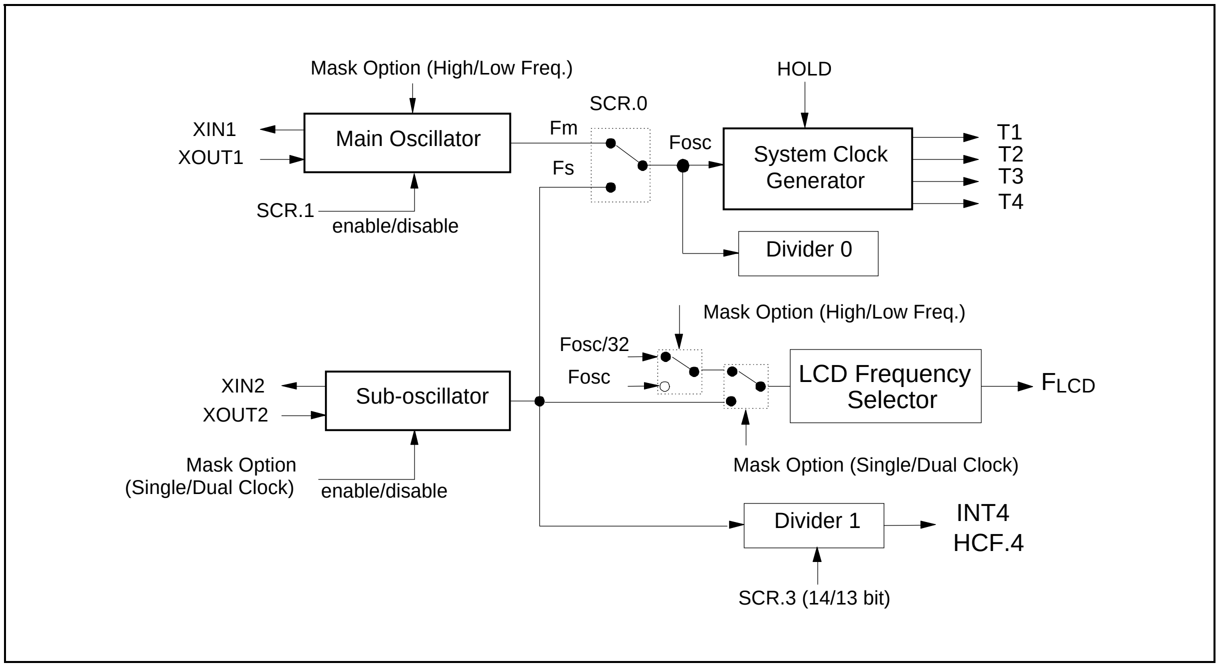


Figure 4. The Dual Clock Operation Mode Control Diagram

	3	2	1	0
SCR	W	–	W	W

Note: W means write only.

Bit 0 = 0   Main oscillator is selected (Fosc = Fm)

      = 1   Sub-oscillator is selected (Fosc = Fs)

Bit 1 = 0   Enable Fm

      = 1   Disable Fm

Bit 2   Reserved

Bit 3 = 0   14-bit Divider1 is selected

      = 1   13-bit Divider1 is selected

Dual clock operation mode:

- Sub-oscillator enable
- SCR.0 = 0, FOSC = Fm
- Flcd = Fs, in STOP mode LCD work continue

Single clock operation mode:

- Sub-oscillator disable
- SCR not use, Main oscillator enable, FOSC = Fm
- Flcd = FOSC (FOSC/32), in STOP mode LCD off

Divider

Each divider is organized as a 14-bit binary up-counter designed to generate periodic interrupts. When the main oscillator starts action, the Divider0 is incremented by each clock (FOSC). When an overflow occurs, the Divider0 event flag is set to 1 (EVF.0 = 1). The interrupt is executed if the Divider0 interrupt enable flag has been set (IEF.0 = 1), and the hold state is terminated if the hold release enable flag has been set (HEF.0 = 1). The last 4-stage of the Divider0 can be reset by executing a CLR DIVR0 instruction. If the main oscillator is connected to the 32768 Hz crystal, the EVF.0 will be set to 1 periodically at each 500 mS interval.

If the sub-oscillator is enabled, the Divider1 is incremented by each clock (Fs). When an overflow occurs, the Divider1 event flag is set to 1 (EVF.4 = 1). The interrupt is executed if the Divider1 interrupt enable flag has been set (IEF.4 = 1), and the hold state is terminated if the hold release enable flag has been set (HEF.4 = 1). There are two time periods (250 mS & 500 mS) that can be selected by setting the SCR.3 bit. When SCR.3 = 0 (default), the 500 mS period time is selected; when SCR.3 = 1, the 250 mS period time is selected.

Watchdog Timer (WDT)

The watchdog timer (WDT) is organized as a 4-bit up counter and is designed to protect the program from unknown errors. The WDT is enabled when the corresponding option code bit of the WDT is set to 1. If the WDT overflows, the chip will be reset. At initial reset, the input clock of the WDT is FOSC/1024. The input clock of the WDT can be switched to FOSC/16384 (or FOSC/1024) by executing the SET PMF, #08H (or CLR PMF, #08H) instruction. The contents of the WDT can be reset by the instruction CLR WDT. In normal operation, the application program must reset WDT before it overflows. A WDT overflow indicates that the operation is not under control and the chip will be reset. The WDT minimun overflow period is 468.75 mS when the system clock (FOSC) is 32 KHz and WDT clock input is FOSC/1024. When the corresponding option code bit of the WDT is set to 0, and the WDT function is disabled. The organization of the Divider0 and watchdog timer is shown in Figure 5.

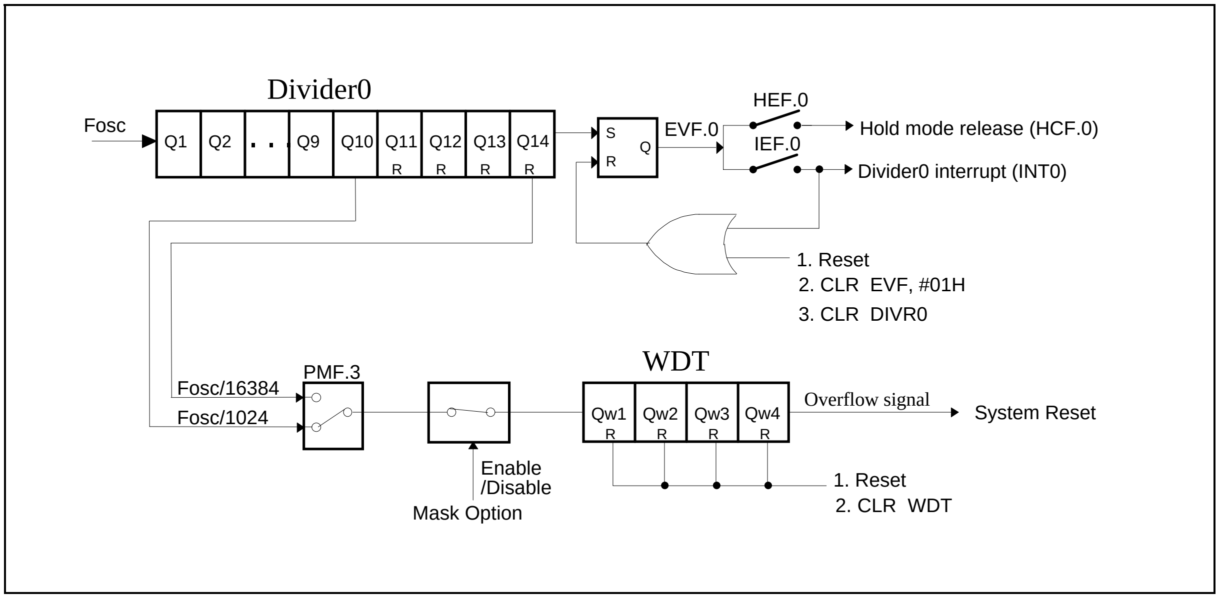


Figure 5. Organization of Divider0 and Watchdog Timer

Parameter Flag (PMF)

The parameter flag is organized as a 4-bit binary register (PMF.0 to PMF.3). The PMF is controlled by the SET PMF, #I or CLR PMF, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PMF	W	–	–	–

Note: W means write only.

Bit 0, Bit1, Bit2 Reserved

Bit 3 = 0 The fundamental frequency of the watchdog timer is FOSC/1024.

= 1 The fundamental frequency of the watchdog timer is FOSC/16384.

Timer/Counter

1. Timer 0 (TM0)

Timer 0 (TM0) is a programmable 8-bit binary down-counter. The specified value can be loaded into TM0 by executing the MOV TM0L (TM0H), R or MOV TM0, #I instructions. When the MOV TM0L (TM0H), R instructions are executed, the TM0 will stop down-counting (if the TM0 is down-counting), the MR0.3 will be reset to 0, and the specified value is loaded into TM0. If MR0.3 is set to 1, the event flag 1 (EVF.1) is reset and the TM0 starts to count. When it decrements to FFH, Timer 0 stops operating and generates an underflow (EVF.1 = 1). The interrupt is executed if the Timer 0 interrupt enable flag has been set (IEF.1 = 1); and the hold state is terminated if the hold release enable flag 1 has been set (HEF.1 = 1). The Timer 0 clock input can be set as FOSC/1024 or FOSC/4 by setting MR0.0 to 1 or by resetting MR0.0 to 0. The default timer value is FOSC/4. The organization of Timer 0 is shown in Figure 6.

If the Timer 0 clock input is FOSC/4:

Desired time interval = (preset value +1) × 4 × 1/FOSC

If the Timer 0 clock input is FOSC/1024:

Desired time interval = (preset value +1) × 1024 × 1/FOSC

Preset value: Decimal number of Timer 0 preset value, and

FOSC: Clock oscillation frequency

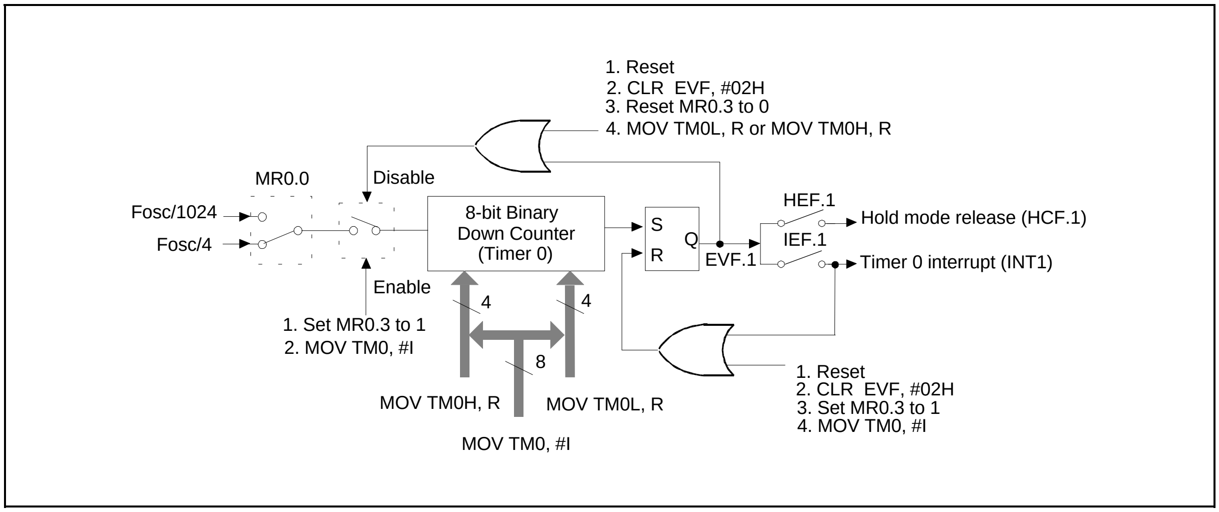


Figure 6. Organization of Timer 0

2. Timer 1 (TM1)

Timer 1 (TM1) is also a programmable 8-bit binary down counter, as shown in Figure . Timer 1 can be used as a counter to count external events or to output an arbitrary frequency to the MFP pin. The input clock of Timer 1 can be one of three sources: FOSC/64, FOSC, or an external clock from the RC.0 input pin. The source can be selected by setting bit 0 and bit 1 of mode register 1 (MR1). At initial reset, the Timer 1 clock input is FOSC. If an external clock is selected as the clock source of Timer 1, the content of Timer 1 is decreased by 1 at the falling edge of RC.0. When the MOV TM1L, R or MOV TM1H, R instruction is executed, the specified data are loaded into the auto-reload buffer and the TM1 down-counting will be disabled (i.e. MR1.3 is reset to 0). If the bit 3 of MR1 is set (MR1.3 = 1), the contents of the auto-reload buffer will be loaded into the TM1 down counter, Timer 1 starts to down count, and the event flag 7 is reset (EVF.7 = 0). When the MOV TM1, #I instruction is executed, the event flag 7 (EVF.7) and MR1.3 are reset and the specified value is loaded into auto-reload buffer and TM1 by the internal hardware, then the MR1.3 is set, that is the TM1 starts to count by the hardware. When the timer decrements to FFH, it will generate an underflow (EVF.7 = 1) and be auto-reloaded with the specified data, after which it will continue to count down. An interrupt is executed if the interrupt enable flag 7 has been set to 1 (IEF.7 = 1), and the hold state is terminated if the hold mode release enable flag 7 is set to 1 (HEF.7 = 1). The specified frequency of Timer 1 can be delivered to the MFP output pin by programming bit 2 of MR1. Bit 3 of MR1 can be used to make Timer 1 stop or start counting.

If the Timer 1 clock input is FT, then:

Desired timer interval = (preset value +1) / F_T

Desired frequency for MFP output pin = F_T ÷ (preset value + 1) ÷ 2 (Hz)

Preset value: Decimal number of Timer 1 preset value, and

FOSC: Clock oscillation frequency

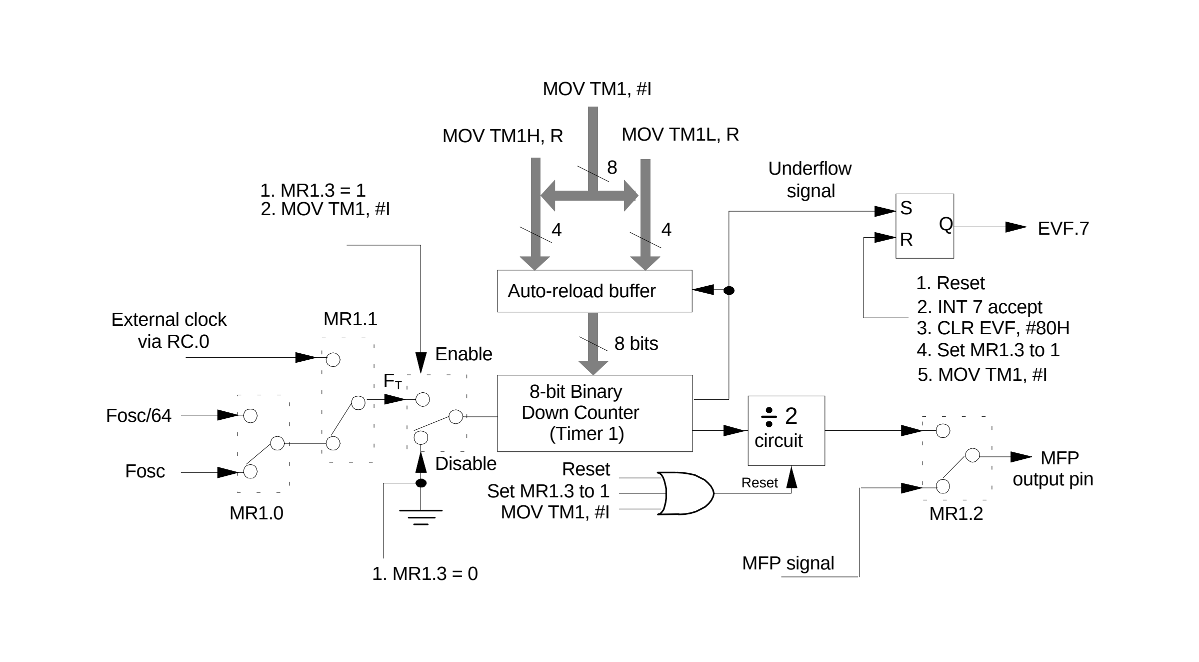


Figure 7. Organization of Timer 1

For example, when F_T equals 32768 Hz, depending on the preset value of TM1, the MFP pin will output a single tone signal in the tone frequency range from 64 Hz to 16384 Hz. The relation between the tone frequency and the preset value of TM1 is shown in the table below.

		3rd octave			4th octave			5th octave		
		Tone frequency	TM1 preset value & MFP frequency		Tone frequency	TM1 preset value & MFP frequency		Tone frequency	TM1 preset value & MFP frequency	
	C	130.81	7CH	131.07	261.63	3EH	260.06	523.25	1EH	528.51
	C#	138.59	75H	138.84	277.18	3AH	277.69	554.37	1CH	564.96
T	**D**	146.83	6FH	146.28	293.66	37H	292.57	587.33	1BH	585.14
	D#	155.56	68H	156.03	311.13	34H	309.13	622.25	19H	630.15
O	**E**	164.81	62H	165.49	329.63	31H	327.68	659.26	18H	655.36
	F	174.61	5DH	174.30	349.23	2EH	372.36	698.46	16H	712.34
N	**F#**	185.00	58H	184.09	369.99	2BH	390.09	739.99	15H	744.72
	G	196.00	53H	195.04	392.00	29H	420.10	783.99	14H	780.19
E	**G#**	207.65	4EH	207.39	415.30	26H	443.81	830.61	13H	819.20
	A	220.00	49H	221.40	**440.00**	**24H**	**442.81**	880.00	12H	862.84
	A#	233.08	45H	234.05	466.16	22H	468.11	932.23	11H	910.22
	B	246.94	41H	248.24	493.88	20H	496.48	987.77	10H	963.76

Note: Central tone is A4 (440 Hz).

Mode Register 0 (MR0)

Mode Register 0 is organized as a 4-bit binary register (MR0.0 to MR0.3). MR0 can be used to control the operation of Timer 0. The bit descriptions are as follows:

	3	2	1	0
MR0	W	–	–	W

Note: W means write only.

Bit 0 = 0 The fundamental frequency of Timer 0 is FOSC/4.

= 1 The fundamental frequency of Timer 0 is FOSC/1024.

Bit 1 & Bit 2 are reserved

Bit 3 = 0 Timer 0 stops down-counting.

= 1 Timer 0 starts down-counting.

Mode Register 1 (MR1)

Mode Register 1 is organized as a 4-bit binary register (MR1.0 to MR1.3). MR1 can be used to control the operation of Timer 1. The bit descriptions are as follows:

	3	2	1	0
MR1	W	W	W	W

Note: W means write only.

Bit 0 = 0 The internal fundamental frequency of Timer 1 is FOSC.
= 1 The internal fundamental frequency of Timer 1 is FOSC/64.

Bit 1 = 0 The fundamental frequency source of Timer 1 is the internal clock.
= 1 The fundamental frequency source of Timer 1 is the external clock from RC.0 input pin.

Bit 2 = 0 The specified waveform of the MFP generator is delivered at the MFP output pin.
= 1 The specified frequency of Timer 1 is delivered at the MFP output pin.

Bit 3 = 0 Timer 1 stops down-counting.
= 1 Timer 1 starts down-counting.

Input/Output Ports RA, RB

Port RA consists of pins RA.0 to RA.3 and Port RB consists of pins RB.0 to RB.3. At initial reset, input/output ports RA and RB are both in input mode. When RA and RB are used as output ports, CMOS or NMOS open drain output type can be selected by the PM0 register. Each pin of port RA or RB can be specified as input or output mode independently by the PM1 and PM2 registers. The MOVA R, RA or MOVA R, RB instructions operate the input functions and the MOV RA, R or MOV RB, R operate the output functions. For more details, refer to the instruction table and Figure 8.

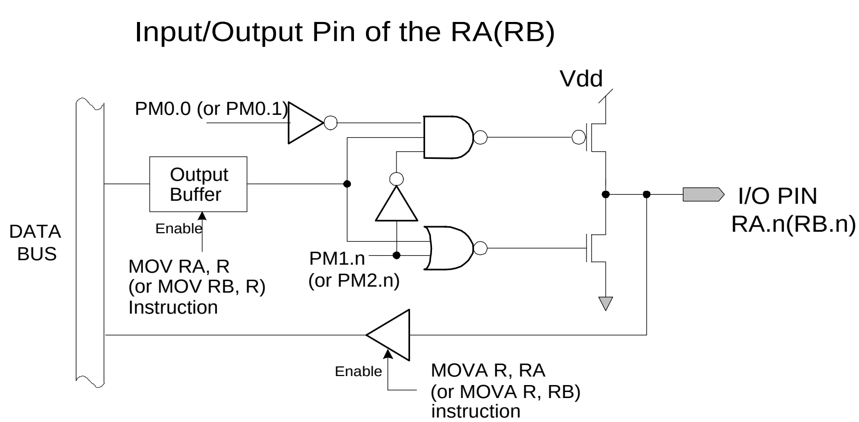


Figure 8. Architecture of Input/Output Pins

Port Mode 0 Register (PM0)

The port mode 0 register is organized as 4-bit binary register (PM0.0 to PM0.3). PM0 can be used to determine the structure of the input/output ports; it is controlled by the MOV PM0, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM0	W	W	W	W

Note: W means write only.

Bit 0 = 0 RA port is CMOS output type. Bit 0 = 1 RA port is NMOS open drain output type.

Bit 1 = 0 RB port is CMOS output type. Bit 1 = 1 RB port is NMOS open drain output type.

Bit 2 = 0 RC port pull-high resistor is disabled. Bit 2 = 1 RC port pull-high resistor is enabled.

Bit 3 = 0 RD port pull-high resistor is disabled. Bit 3 = 1 RD port pull-high resistor is enabled.

Port Mode 1 Register (PM1)

The port mode 1 register is organized as 4-bit binary register (PM1.0 to PM1.3). PM1 can be used to control the input/output mode of port RA. PM1 is controlled by the MOV PM1, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM1	W	W	W	W

Note: W means write only.

Bit 0 = 0 RA.0 works as output pin; Bit 0 = 1 RA.0 works as input pin

Bit 1 = 0 RA.1 works as output pin; Bit 1 = 1 RA.1 works as input pin

Bit 2 = 0 RA.2 works as output pin; Bit 2 = 1 RA.2 works as input pin

Bit 3 = 0 RA.3 works as output pin; Bit 3 = 1 RA.3 works as input pin

At initial reset, port RA is input mode (PM1 = 1111B).

Port Mode 2 Register (PM2)

The port mode 2 register is organized as 4-bit binary register (PM2.0 to PM2.3). PM2 can be used to control the input/output mode of port RB. PM2 is controlled by the MOV PM2, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM2	W	W	W	W

Note: W means write only.

Bit 0 = 0 RB.0 works as output pin; Bit 0 = 1 RB.0 works as input pin

Bit 1 = 0 RB.1 works as output pin; Bit 1 = 1 RB.1 works as input pin

Bit 2 = 0 RB.2 works as output pin; Bit 2 = 1 RB.2 works as input pin

Bit 3 = 0 RB.3 works as output pin; Bit 3 = 1 RB.3 works as input pin

At initial reset, the port RB is input mode (PM2 = 1111B).

Input Ports RC & RD

Port RC consists of pins RC.0 to RC.3, and port RD consists of pins RD.0 to RD.3. Each pin of port RC and port RD can be connected to a pull-up resistor, which is controlled by the port mode 0 register (PM0). When the PEF, HEF, and IEF corresponding to the RC port are set, a signal change on the specified pins of port RC will execute the hold mode release or interrupt subroutine. Port status register 0 (PSR0) records the status of ports RC, i.e., any signal changes on the pins that make up the port. PSR0 can be read out and cleared by the MOV R, PSR0, and CLR PSR0 instructions. In addition, the falling edge signal on the pin of port RC specified by the instruction MOV SEF, #I will cause the device to exit the stop mode. Refer to Figure 9 and the instruction table for more details. The RD port is used as input port only, it has no hold mode release, wake-up stop mode or interrupt functions.

Port Status Register 0 (PSR0)

Port status register 0 is organized as 4-bit binary register (PSR0.0 to PSR0.3). PSR0 can be read or cleared by the MOVA R, PSR0, and CLR PSR0 instructions. The bit descriptions are as follows:

	3	2	1	0
PSR0	R	R	R	R

Note: R means read only.

Bit 0 = 1 Signal change at RC.0

Bit 1 = 1 Signal change at RC.1

Bit 2 = 1 Signal change at RC.2

Bit 3 = 1 Signal change at RC.3

Port Enable Flag (PEF)

The port enable flag is organized as 4-bit binary register (PEF.0 to PEF.3). Before port RC may be used to release the hold mode or perform interrupt function, the content of the PEF must be set first. The PEF is controlled by the MOV PEF, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PEF	W	W	W	W

Note: W means write only.

PEF.0: Enable/disable the signal change at pin RC.0 to release hold mode or perform interrupt.

PEF.1: Enable/disable the signal change at pin RC.1 to release hold mode or perform interrupt.

PEF.2: Enable/disable the signal change at pin RC.2 to release hold mode or perform interrupt.

PEF.3: Enable/disable the signal change at pin RC.3 to release hold mode or perform interrupt.

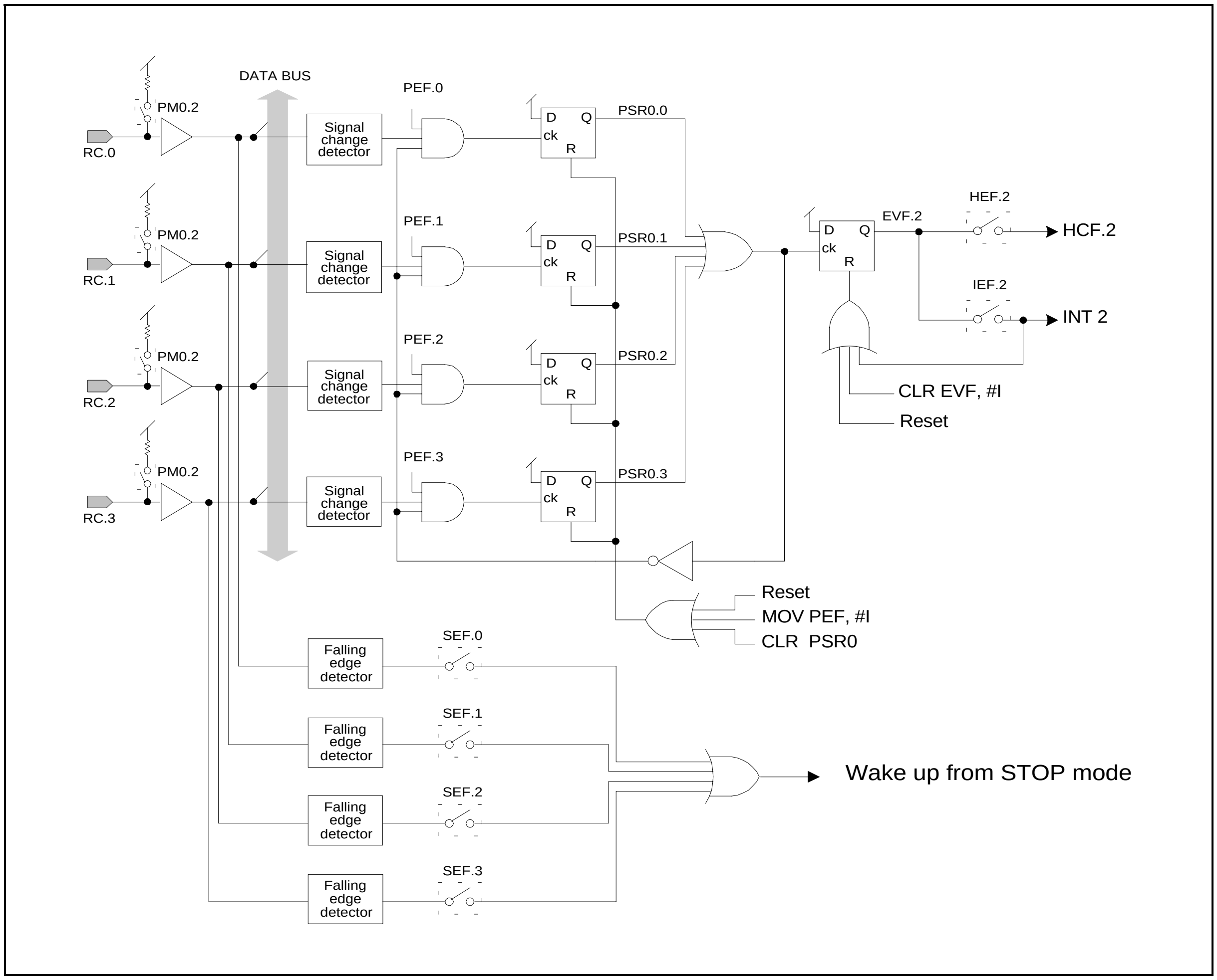


Figure 9. Architecture of Input Port RC

Output Port RE

When the MOV RE, R instruction is executed, the data in the RAM will be output to port RE and it provides a high sink current to drive LEDs.

MFP Output Pin (MFP)

The MFP output pin can output the Timer 1 clock or the modulation frequency; the output of the pin is determined by mode register 1 (MR1). The configuration of MFP is shown in Figure 9. When bit 2 of MR1 is reset to "0," the MFP output can deliver a modulation output in any combination of one signal from among DC, 4096Hz, 2048Hz, and one or more signals from among 128 Hz, 64 Hz, 8 Hz, 4 Hz, 2 Hz, or 1 Hz (when using a 32.768 KHz crystal). The MOV MFP, #I instruction is used to specify the modulation output combination. The data specified by the 8-bit operand and the MFP output pin are shown below.

(Fosc = 32.768 KHz)

R7 R6	R5	R4	R3	R2	R1	R0	FUNCTION
0 0	0	0	0	0	0	0	Low level
	0	0	0	0	0	1	128 Hz
	0	0	0	0	1	0	64 Hz
	0	0	0	1	0	0	8 Hz
	0	0	1	0	0	0	4 Hz
	0	1	0	0	0	0	2 Hz
	1	0	0	0	0	0	1 Hz
0 1	0	0	0	0	0	0	High level
	0	0	0	0	0	1	128 Hz
	0	0	0	0	1	0	64 Hz
	0	0	0	1	0	0	8 Hz
	0	0	1	0	0	0	4 Hz
	0	1	0	0	0	0	2 Hz
	1	0	0	0	0	0	1 Hz
1 0	0	0	0	0	0	0	2048 Hz
	0	0	0	0	0	1	2048 Hz * 128 Hz
	0	0	0	0	1	0	2048 Hz * 64 Hz
	0	0	0	1	0	0	2048 Hz * 8 Hz
	0	0	1	0	0	0	2048 Hz * 4 Hz
	0	1	0	0	0	0	2048 Hz * 2 Hz
	1	0	0	0	0	0	2048 Hz * 1 Hz
1 1	0	0	0	0	0	0	4096 Hz
	0	0	0	0	0	1	4096 Hz * 128 Hz
	0	0	0	0	1	0	4096 Hz * 64 Hz
	0	0	0	1	0	0	4096 Hz * 8 Hz
	0	0	1	0	0	0	4096 Hz * 4 Hz
	0	1	0	0	0	0	4096 Hz * 2 Hz
	1	0	0	0	0	0	4096 Hz * 1 Hz

Interrupts

The W741E260 provides four internal interrupt sources (Divider 0, Divider 1, Timer 0, Timer 1) and one external interrupt source (port RC) for W741C260 body or three internal interrupt sources (Divider 0, Timer 0, Timer 1) and two external interrupt sources (port RC, $\overline{\text{INT}}$ pin) for W741C250 body. Vector addresses for each of the interrupts are located in the range of program memory (ROM) addresses 004H to 020H. The flags IEF, PEF, and EVF are used to control the interrupts. When EVF is set to "1" by hardware and the corresponding bits of IEF and PEF have been set by software, an interrupt is generated. When an interrupt occurs, all of the interrupts are inhibited until the EN INT or MOV IEF, #I instruction is invoked. The interrupts can also be disabled by executing the DIS INT instruction. When an interrupt is generated in hold mode, the hold mode will be released momentarily and interrupt subroutine will be executed. After the RTN instruction is executed in an interrupt subroutine, the µC will enter hold mode again. The operation flow chart is shown in Figure 11. The control diagram is shown below.

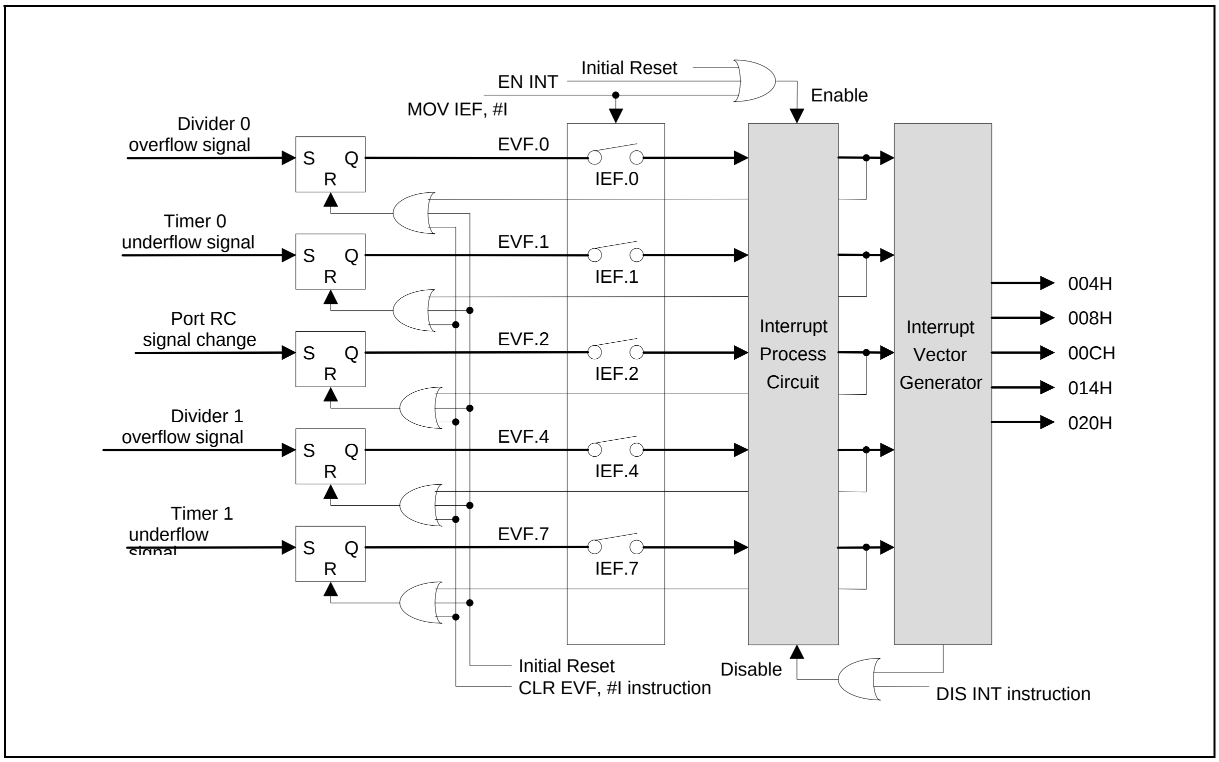


Figure 10. Interrupt Event Control Diagram

Interrupt Enable Flag (IEF)

The interrupt enable flag is organized as an 8-bit binary register (IEF.0 to IEF.7). These bits are used to control the interrupt conditions. It is controlled by the MOV IEF, #I instruction. When one of these interrupts is accepted, the corresponding to the bit of the event flag will be reset, but the other bits are unaffected. In interrupt subroutine, these interrupts will be disabled till the instruction MOV IEF, #I or EN INT is executed again. Otherwise, these interrupts can be disabled by executing DIS INT instruction. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
IEF	W	—	—	W	—	W	W	W

Note: W means write only.

IEF.0 = 1 Interrupt 0 is accepted by overflow from the Divider 0.

IEF.1 = 1 Interrupt 1 is accepted by underflow from the Timer 0.

IEF.2 = 1 Interrupt 2 is accepted by a signal change at port RC.

IEF.3 is reserved.

IEF.4 = 1 Interrupt 4 is accepted by overflow from the Divider 1 for W741C260 body.

Interrupt 4 is accepted by a falling edge signal at the $\overline{\text{INT}}$ pin for W741C250 body.

IEF.5 & IEF.6 are reserved.

IEF.7 = 1 Interrupt 7 is accepted by underflow from Timer 1.

Stop Mode Operation

In stop mode, all operations of the µC cease (excluding the operation of the sub-oscillator and divider 1 when the dual-clock operation mode is selected). The µC enters stop mode when the STOP instruction is executed and exits stop mode when an external trigger is activated (by a falling signal on the RC port for W741C260 body or by a falling signal on the RC port or a low level on the $\overline{\text{INT}}$ pin for W741C250 body). When the designated signal is accepted, the µC awakens and executes the next instruction (if the corresponding bits of IEF and PEF have been set, It will enter the interrupt service routine after stop mode released). To prevent erroneous execution, the NOP instruction should follow the STOP command. But In the dual-clock slow operation mode, the STOP instruction will disable the main-oscillator oscillating; the µC system is still operated by the sub-oscillator.

Stop Mode Wake-up Enable Flag for RC Port (SEF)

The stop mode wake-up flag for port RC is organized as an 4-bit binary register (SEF.0 to SEF.3). Before port RC may be used to make the device exit the stop mode, the content of the SEF must be set first. The SEF is controlled by the MOV SEF, #I instruction. The bit descriptions are as follows:

	3	2	1	0
SEF	W	W	W	W

Note: W means write only.

SEF.0 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.0

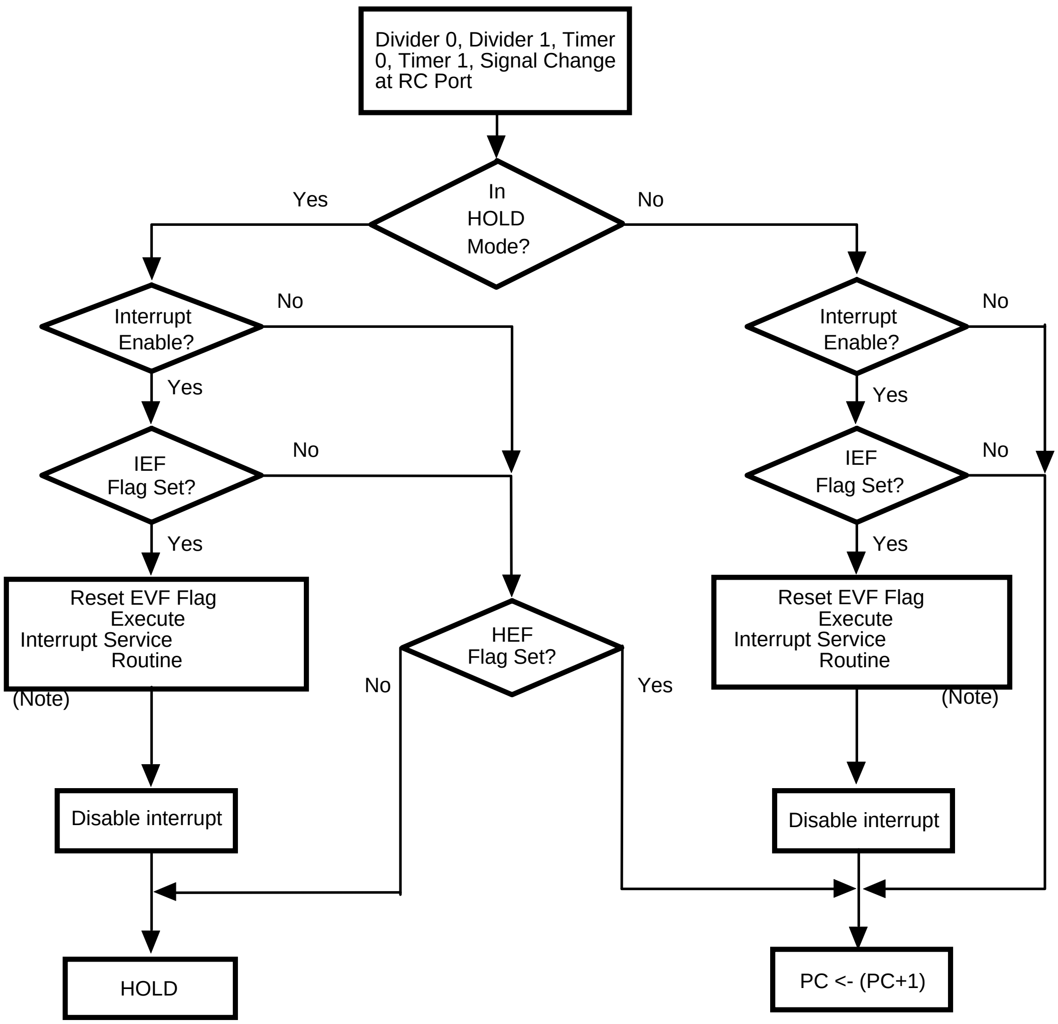
SEF.1 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.1

SEF.2 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.2

SEF.3 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.3

Hold Mode Operation

In hold mode, all operations of the µC cease, except for the operation of the oscillator, Timer, Divider and LCD driver. The µC enters hold mode when the HOLD instruction is executed. The hold mode can be released in one of five ways: by the action of timer 0, timer 1, divider 0, divider 1, the RC port. Before the device enters the hold mode, the HEF, PEF, and IEF flags must be set to define the hold mode release conditions. For more details, refer to the instruction-set table and the following flow chart.



Note: The bit of EVF corresponding to the interrupt signal will be reset.

Figure 11. Hold Mode and Interrupt Operation Flow Chart

Hold Mode Release Enable Flag (HEF)

The hold mode release enable flag is organized as an 8-bit binary register (HEF.0 to HEF.7). The HEF is used to control the hold mode release conditions. It is controlled by the MOV HEF, #I instruction. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
HEF	W	–	–	W	–	W	W	W

Note: W means write only.

HEF.0 = 1   Overflow from the Divider 0 causes Hold mode to be released.

HEF.1 = 1   Underflow from Timer 0 causes Hold mode to be released.

HEF.2 = 1   Signal change at port RC causes Hold mode to be released.

HEF.3 is reserved.

HEF.4 = 1   Overflow from the Divider 1 causes Hold mode to be released for W741C260 body.

Falling edge signal at the $\overline{\text{INT}}$ pin causes Hold mode to be released for W741C250 body.

HEF.5 & HEF.6 are reserved.

HEF.7 = 1   Underflow from Timer 1 causes Hold mode to be released.

Hold Mode Release Condition Flag (HCF)

The hold mode release condition flag is organized as a 8-bit binary register (HCF.0 to HCF.7). It indicates by which interrupt source the hold mode has been released, and is loaded by hardware. The HCF can be read out by the MOVA R, HCFL and MOVA R, HCFH instructions. When any of the HCF bits is "1," the hold mode will be released and the HOLD instruction is invalid. The HCF can be reset by the CLR EVF or MOV HEF, #I (HEF = 0) instructions. When EVF and HEF have been reset, the corresponding bit of HCF is reset simultaneously. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
HCF	—	—	R	R	—	R	R	R

Note: R means read only.

HCF.0 = 1 Hold mode was released by overflow from the divider 0.

HCF.1 = 1 Hold mode was released by underflow from the timer 0.

HCF.2 = 1 Hold mode was released by a signal change at port RC.

HCF.3 is reserved.

HCF.4 = 1 Hold mode was released by overflow from the divider 1 for W741C260 body.

Hold mode was released by a falling edge signal at the $\overline{\text{INT}}$ pin for W741C250 body.

HCF.5 = 1 Hold mode was released by underflow from the timer 1.

HCF.6 and HCF.7 are reserved.

Event Flag (EVF)

The event flag is organized as a 8-bit binary register (EVF.0 to EVF.7). It is set by hardware and reset by CLR EVF, #I instruction or the occurrence of an interrupt. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
EVF	R	—	—	R	—	R	R	R

Note: R means read only.

EVF.0 = 1 Overflow from divider 0 occurred.

EVF.1 = 1 Underflow from timer 0 occurred.

EVF.2 = 1 Signal change at port RC occurred.

EVF.3 is reserved.

EVF.4 = 1 Overflow from divider 1 occurred for W741C260 body.

Falling edge signal at the $\overline{\text{INT}}$ pin occurred for W741C250 body.

EVF.5 & EVF.6 are reserved.

EVF.7 = 1 Underflow from Timer 1 occurred.

Reset Function

The W741E260 is reset either by a power-on reset or by using the external $\overline{\text{RES}}$ pin. The initial state of the W741E260 after the reset function is executed is described below.

Program Counter (PC)	000H
TM0, TM1	Reset
MR0, MR1, PAGE registers	Reset
PSR0 registers	Reset
IEF, HEF, HCF, PEF, EVF, SEF flags	Reset
SCR register	Reset
Timer 0 input clock	FOSC/4
Timer 1 input clock	FOSC
MFP output	Low
Input/output ports RA, RB	Input mode
Output port RE	High
RA & RB ports output type	CMOS type
RC & RD ports pull-high resistors	Disable
Input clock of the watchdog timer	FOSC/1024
LCD display	OFF
Segment output mode	LCD drive output

LCD Controller/Driver

The W741E260 can directly drive an LCD with 32 segment output pins and 4 common output pins for a total of 32 × 4 dots. Option codes can be used to select one of five options for the LCD driving mode: static, 1/2 Bias 1/2 duty, 1/2 Bias 1/3 duty, 1/3 Bias 1/3 duty, or 1/3 Bias 1/4 duty (see Figure 13). The alternating frequency of the LCD can be set as Fw/64, Fw/128, Fw/256, or Fw/512. In addition, option codes can also be used to set up four of the LCD driver output pins (segment 0 to segment 31) as a DC output port. The structure of the LCD alternating frequency (FLCD) is shown in Figure 12.

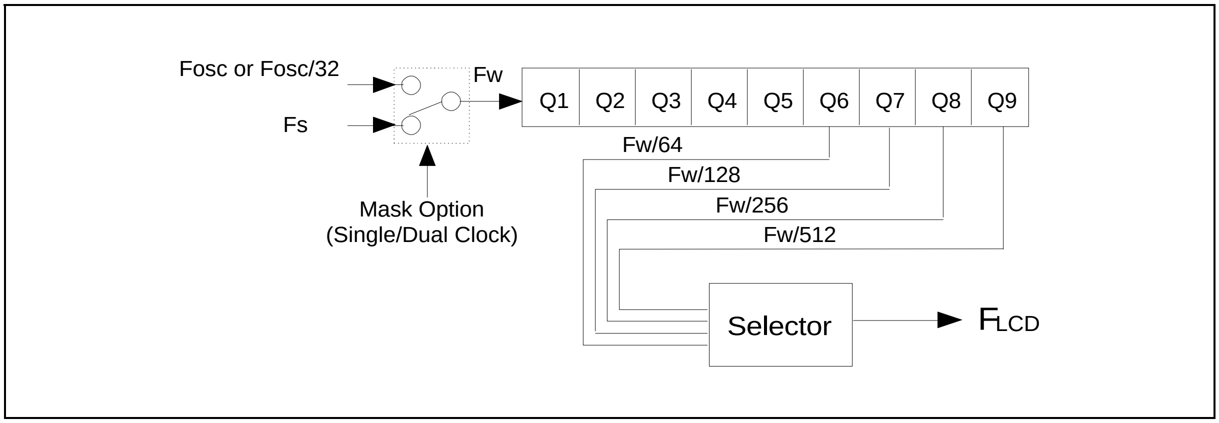


Figure 12. LCD Alternating Frequency (FLCD) Circuit Diagram

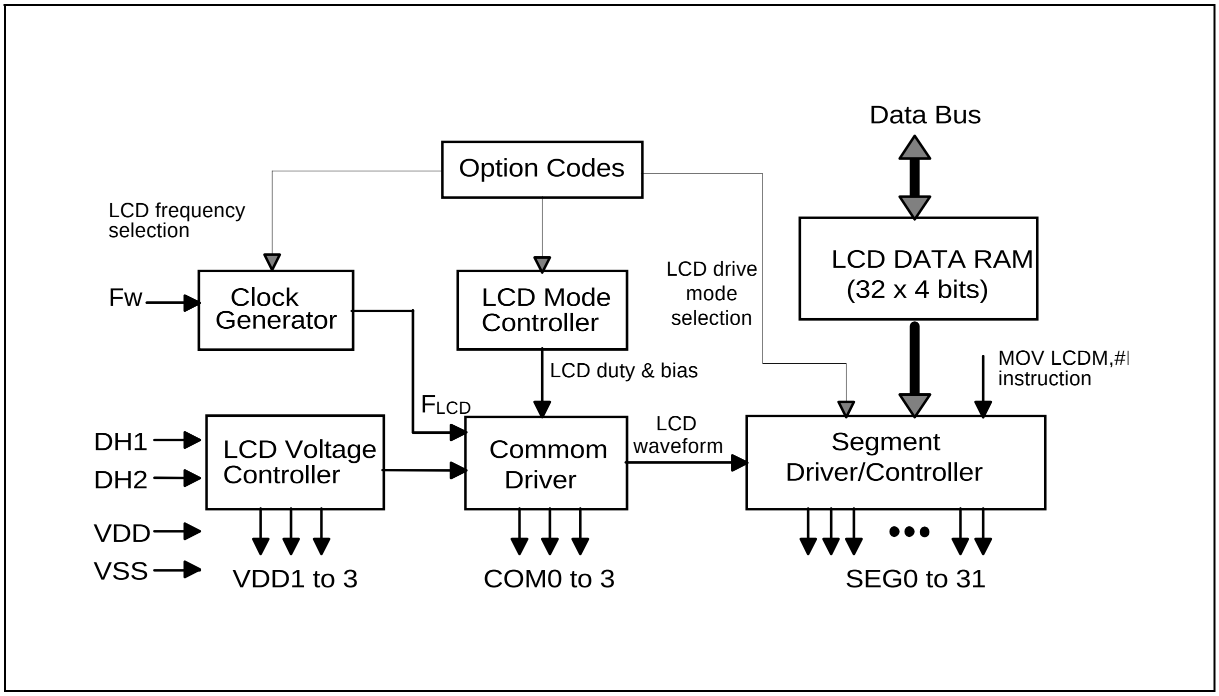


Figure 13. LCD Driver/Controller Circuit Diagram

When Fw = 32.768 KHz, the LCD frequency is as shown in the table below.

LCD Frequency	Static	1/2 Duty	1/3 Duty	1/4 Duty
Fw/512 (64 Hz)	64	32	21	16
Fw/256 (128 Hz)	128	64	43	32
Fw/128 (256 Hz)	256	128	85	64
Fw/64 (512 Hz)	512	256	171	128

Corresponding to the 32 LCD drive output pins, there are 32 LCD data RAM segments (LCDR00 to LCDR1F). Instructions such as MOV LCDR, #I; MOV WR, LCDR; MOV LCDR, WR; and MOV LCDR, ACC are used to control the LCD data RAM. The data in the LCD data RAM are transferred to the segment output pins automatically without program control. When the bit value of the LCD data RAM is "1," the LCD is turned on. When the bit value of the LCD data RAM is "0," LCD is turned off. The contents of the LCD data RAM (LCDR) are sent out through the segment0 to segment31 pins by a direct memory access. The relation between the LCD data RAM and segment/common pins is shown below.

		COM3	COM2	COM1	COM0
LCD Data RAM	Output Pin	bit 3	bit 2	bit 1	bit 0
LCDR00	SEG0	0/1	0/1	0/1	0/1
LCDR01	SEG1	0/1	0/1	0/1	0/1
. . .	. . .	. . .	. . .	. . .	. . .
LCDR1E	SEG30	0/1	0/1	0/1	0/1
LCDR1F	SEG31	0/1	0/1	0/1	0/1

The LCDON instruction turns the LCD display on (even in HOLD mode), and the LCDOFF instruction turns the LCD display off. At initial reset, all the LCD segments are lit. When the initial reset state ends, the LCD display is turned off automatically. To turn on the LCD display, the instruction LCDON must be executed. When the drive output pins are used as DC output ports (setting by option codes, please refer the user's manual of ASM741S assembler for more detail), CMOS output type or NMOS output type can be selected by executing the instruction MOV LCDM, #I. The relationship between the LCD data RAM and segment/common pins is shown below. The data in LCDR00 are transferred to the corresponding segment output port (SEG3 to SEG0) by a direct memory access. The other LCD data RAM segments can be used as normal data RAM to store data.

LCD Data RAM	Output Pin	Bit 3	Bit 2	Bit 1	Bit 0
LCDR00	SEG3-SEG0	SEG3	SEG2	SEG1	SEG0
LCDR03-LCDR01	-	-	-	-	-
LCDR04	SEG7-SEG4	SEG7	SEG6	SEG5	SEG4
LCDR07-LCDR05	-	-	-	-	-
. . .	. . .	. . .	. . .	. . .	. . .
LCDR1C	SEG31-SEG30	SEG31	SEG30	SEG29	SEG28
LCDR1F-LCDR1D	-	-	-	-	-

The relationship between the LCD drive mode and the maximum number of drivable LCD segments is shown below.

LCD Drive Mode	Max. Number of Drivable LCD Segment	Connection at Power Input
STATIC	32 (COM0)	Connect V_{DD3}, V_{DD2} to V_{DD1}
1/2 Bias 1/2 Duty	64 (COM0-COM1)	Connect V_{DD3} to V_{DD2}
1/2 Bias 1/3 Duty	96 (COM0-COM2)	Connect V_{DD3} to V_{DD2}
1/3 Bias 1/3 Duty	96 (COM0-COM2)	-
1/3 Bias 1/4 Duty	128 (COM0-COM3)	-

LCD Output Mode Type Flag (LCDM)

The LCD output mode type flag is organized as an 8-bit binary register (LCDM.0 to LCDM.7). These bits are used to control the LCD output pins architecture. When LCD output pins are set to DC output mode by option codes, the architecture of these output pins (segment 0 to segment 31) can be selected as CMOS or NMOS type. It is controlled by the MOV LCDM, #I instruction. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
LCDM	W	W	W	W	W	W	W	W

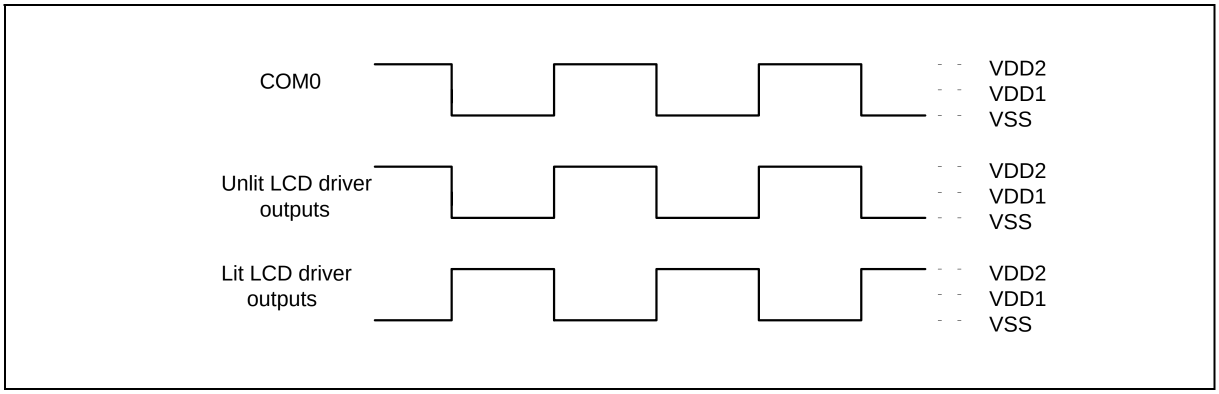
Note: W means write only.

LCDM.0 = 0 SEG0 to SEG3 work as CMOS output type.
= 1 SEG0 to SEG3 work as NMOS output type.
LCDM.1 = 0 SEG4 to SEG7 work as CMOS output type.
= 1 SEG4 to SEG7 work as NMOS output type.
LCDM.2 = 0 SEG8 to SEG11 work as CMOS output type.
= 1 SEG8 to SEG11 work as NMOS output type.
LCDM.3 = 0 SEG12 to SEG15 work as CMOS output type.
= 1 SEG12 to SEG15 work as NMOS output type.
LCDM.4 = 0 SEG16 to SEG19 work as CMOS output type.
= 1 SEG16 to SEG19 work as NMOS output type.
LCDM.5 = 0 SEG20 to SEG23 work as CMOS output type.
= 1 SEG20 to SEG23 work as NMOS output type.
LCDM.6 = 0 SEG24 to SEG27 work as CMOS output type.
= 1 SEG24 to SEG27 work as NMOS output type.
LCDM.7 = 0 SEG28 to SEG31 work as CMOS output type.
= 1 SEG28 to SEG31 work as NMOS output type.

The output waveforms for the five LCD driving modes are shown below.

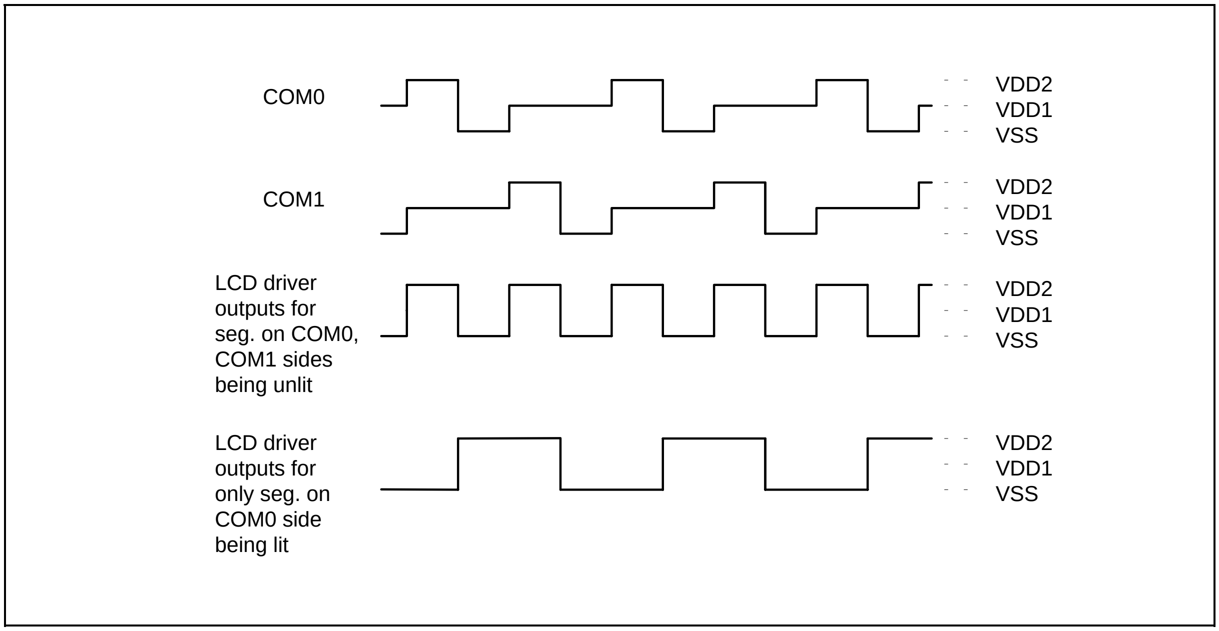
Static Lighting System (Example)

Normal Operating Mode

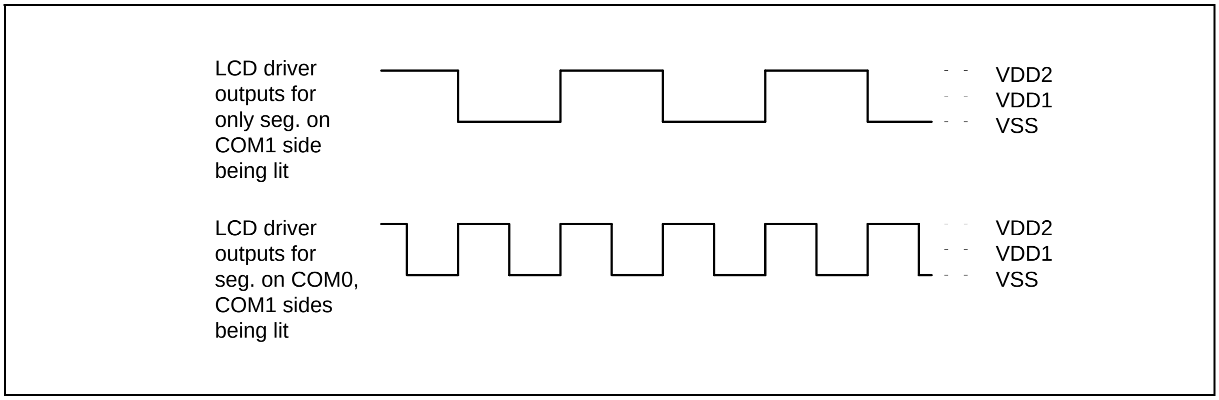


1/2 Bias 1/2 duty Lighting System (Example)

Normal Operating Mode

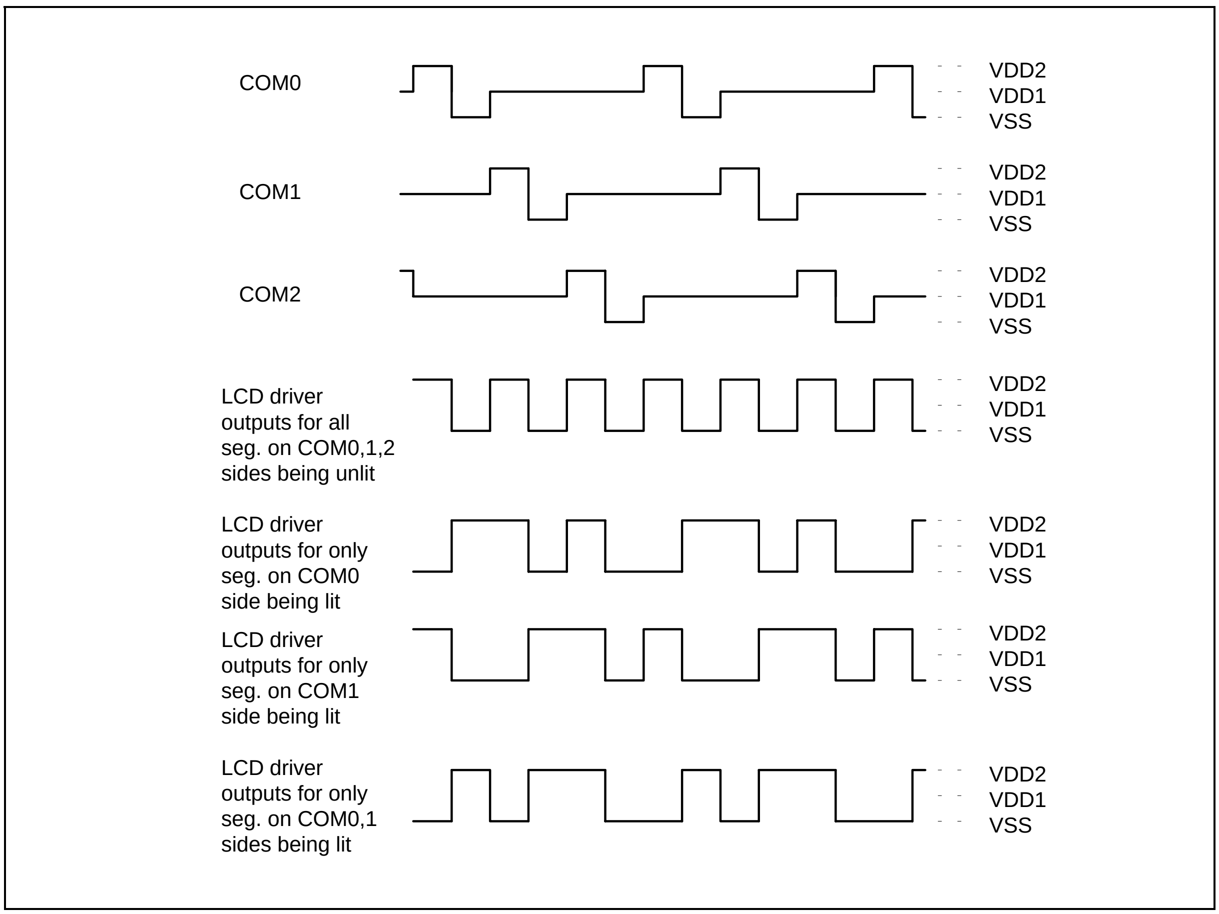


1/2 Bias 1/2 duty Lighting System (Example) - Normal Operating Mode, continued

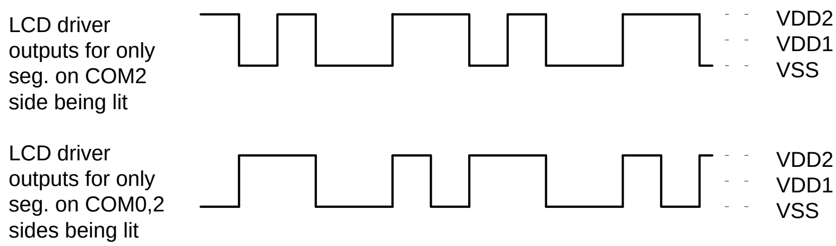


1/2 Bias 1/3 duty Lighting System (Example)

Normal Operating Mode

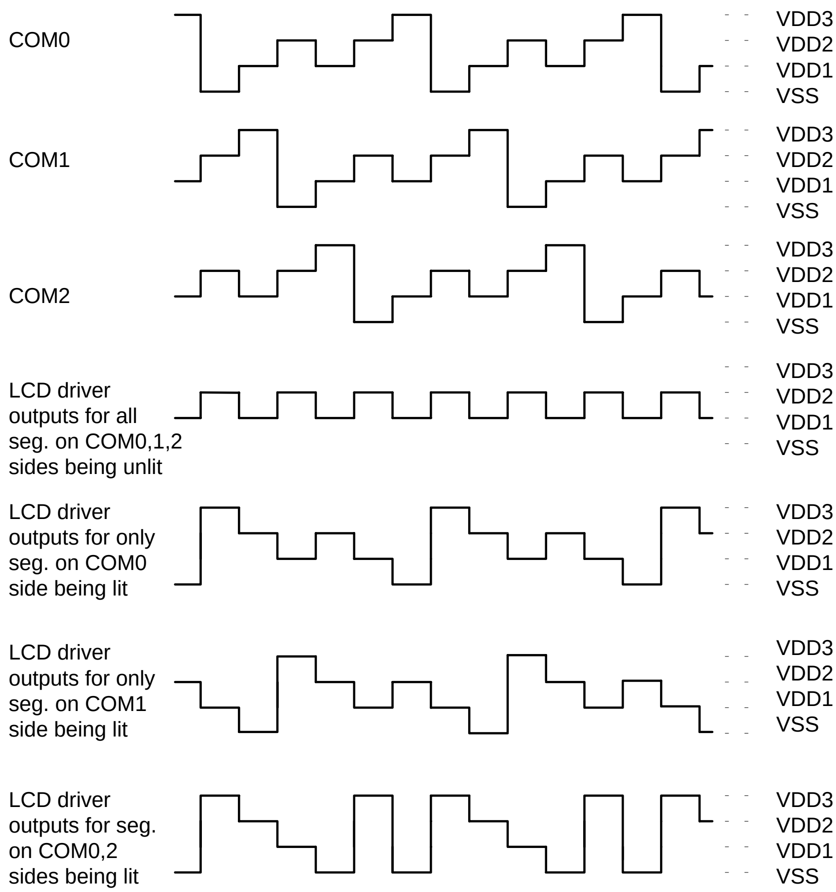


1/2 Bias 1/3 duty Lighting System (Example) - Normal Operating Mode, continued

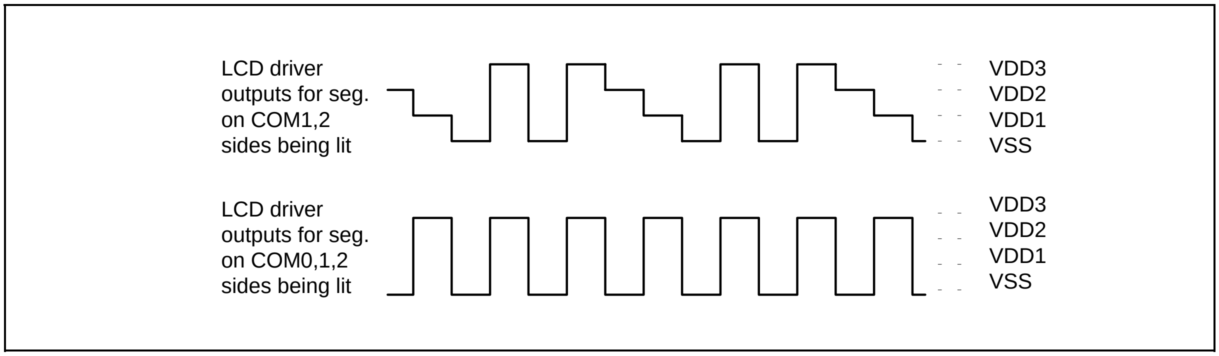


1/3 Bias 1/3 duty Lighting System (Example)

Normal Operating Mode

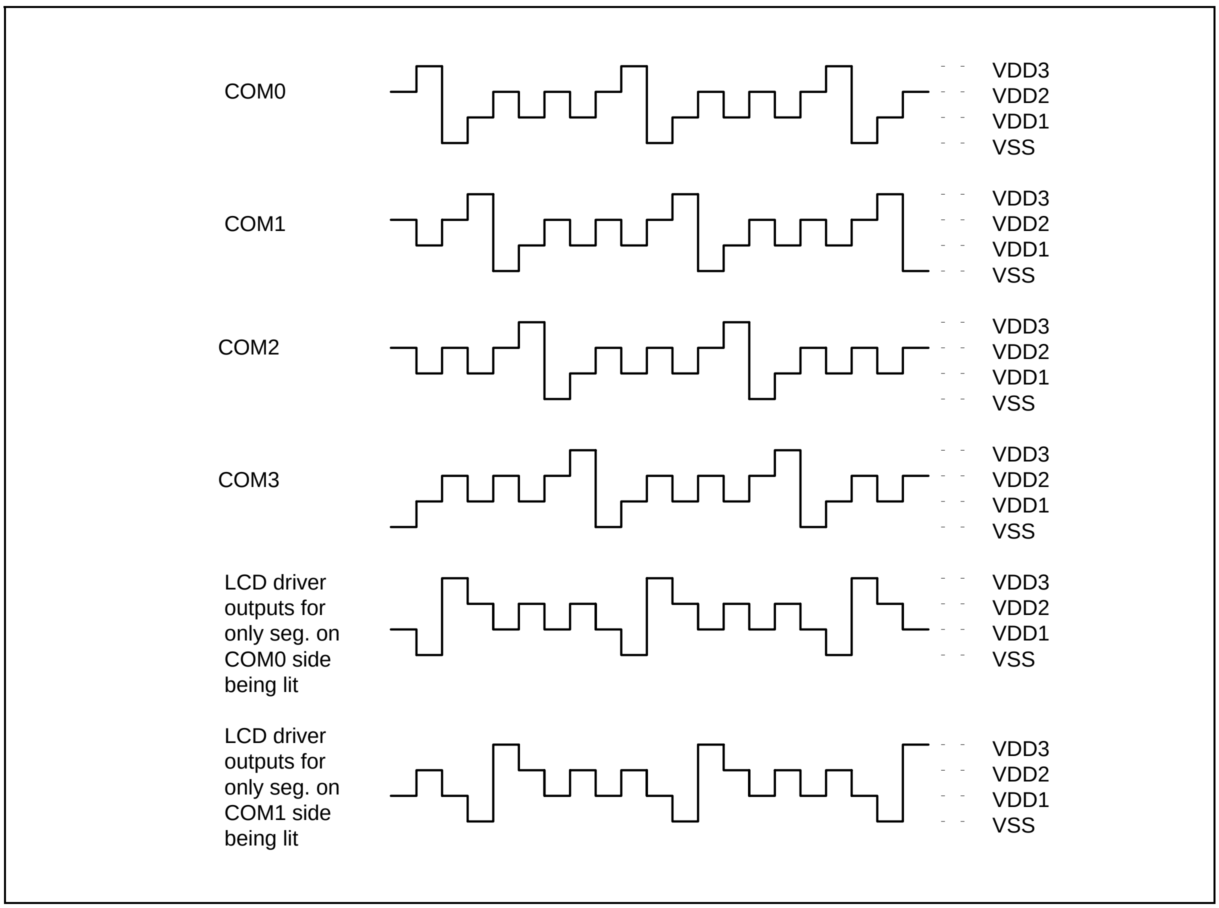


1/3 Bias 1/3 duty Lighting System (Example) - Normal Operating Mode, continued

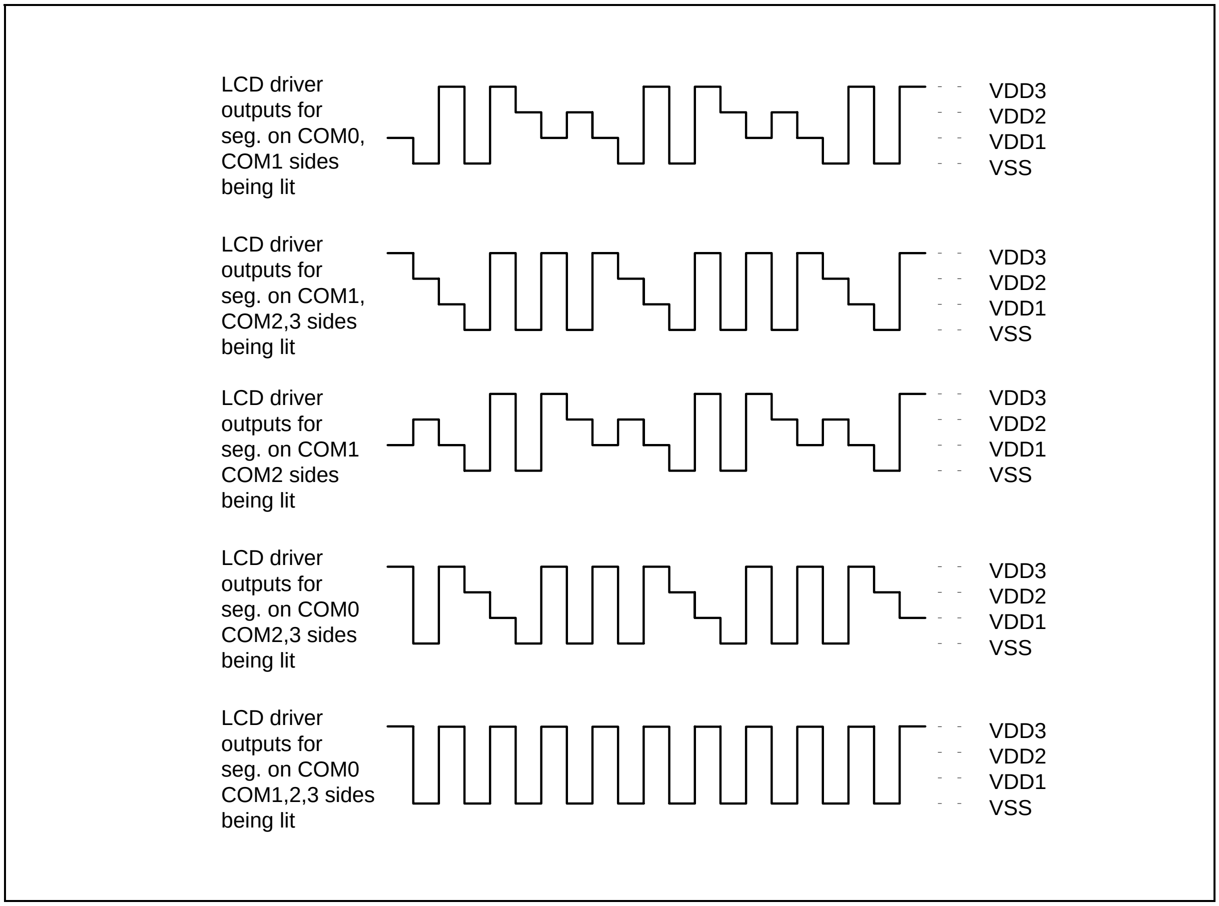


1/3 Bias 1/4 duty Lighting System (Example)

Normal Operating Mode

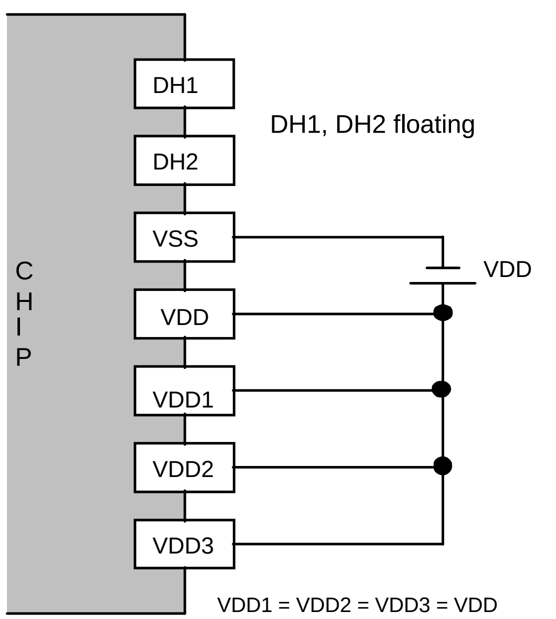


1/3 Bias 1/4 duty Lighting System (Example) - Normal Operating Mode, continued

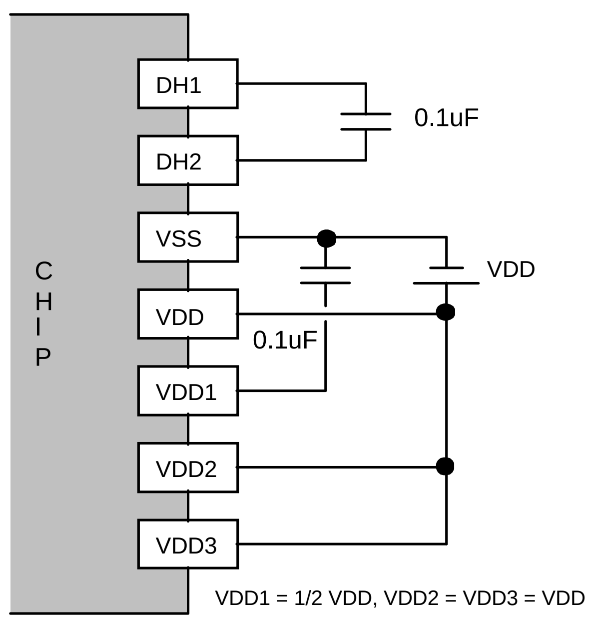


The power connections for each LCD driving mode, which are determined by a mask option, are shown below.

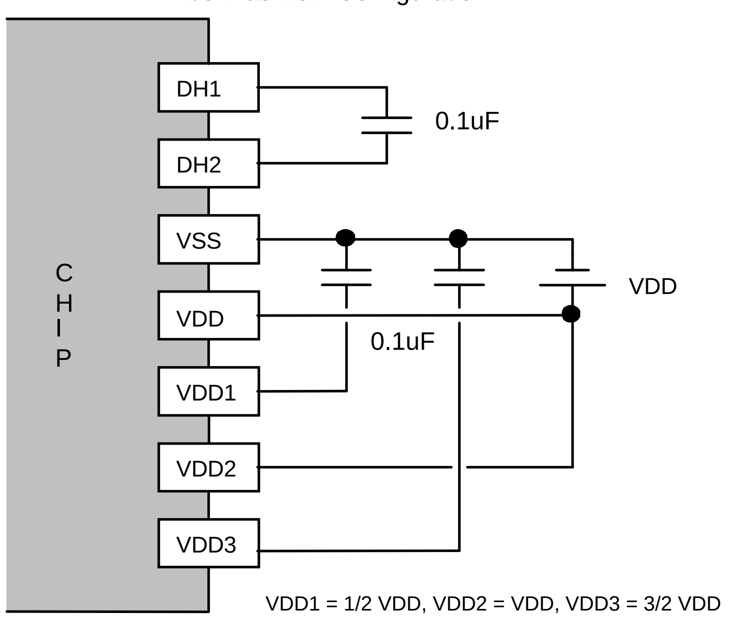
Static LCD Configuration



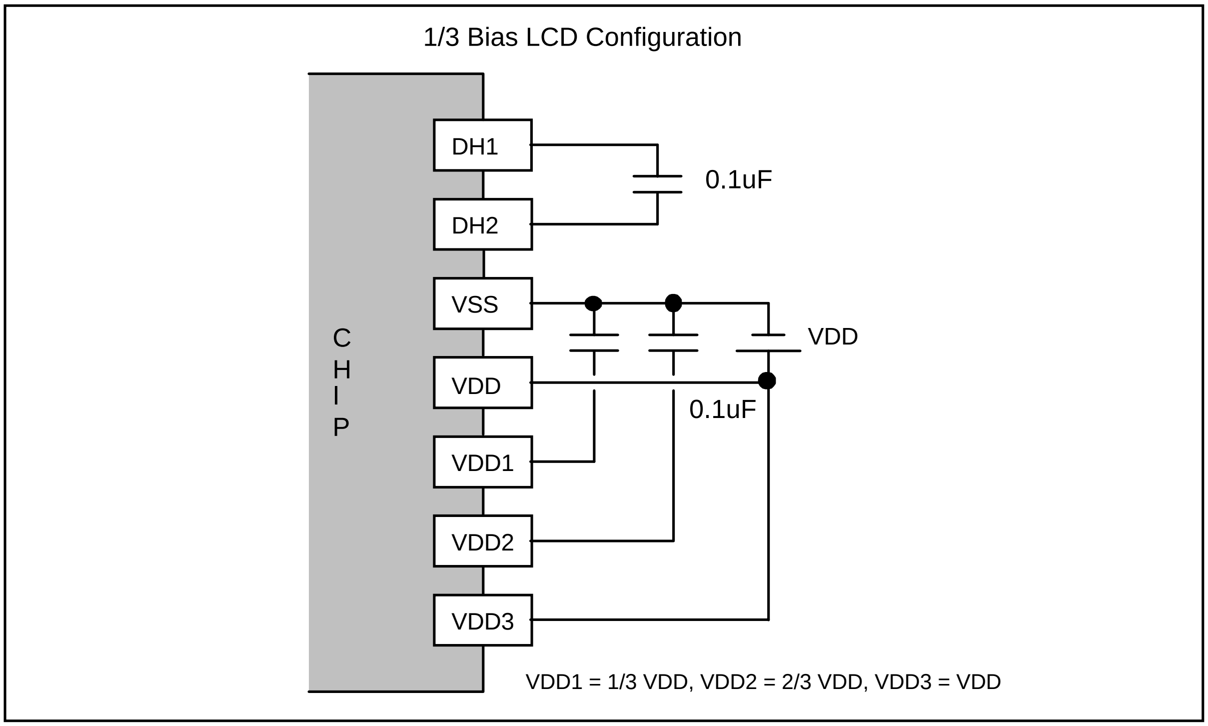
1/2 Bias LCD Configuration



1/3 Bias LCD Configuration



LCD Configuration, continued



EEPROM Program/Erase Description

The built-in program code memory of the W741E260 is the EEPROM structure. This memory can be programmed, erased and verified through the VPP, MODE and DATA pins. The on board program/erase connection is shown below.

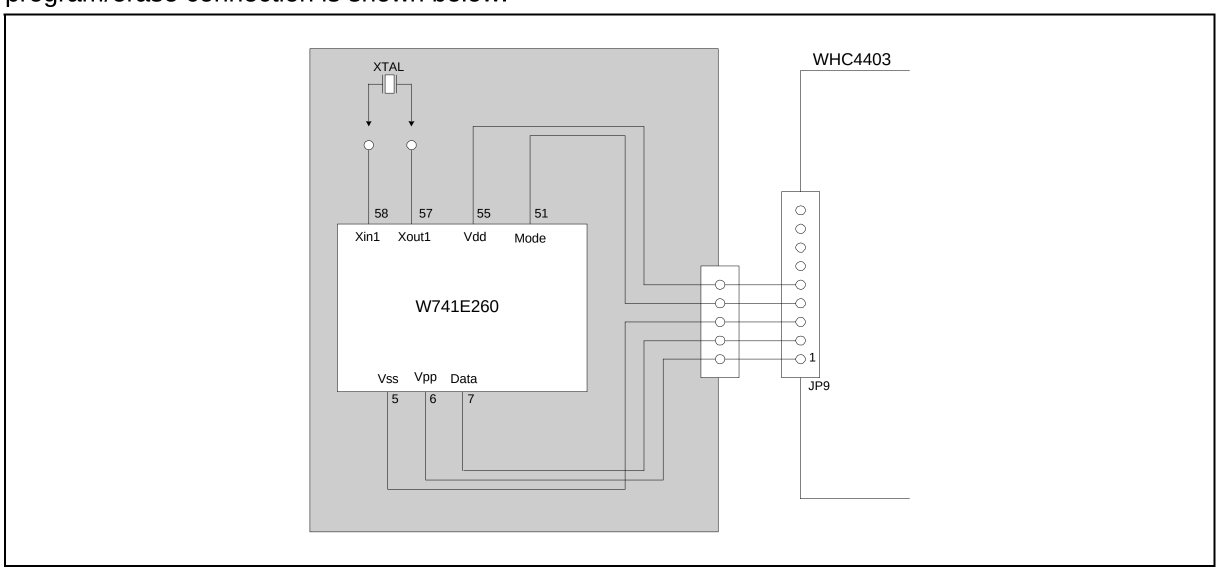


Figure 14. The W741E260 Program/Erase Configuration

ABSOLUTE MAXIMUM RATINGS

PARAMETER	RATING	UNIT
Supply Voltage to Ground Potential	-0.3 to +7.0	V
Applied Input/Output Voltage	-0.3 to +7.0	V
Power Dissipation	120	mW
Ambient Operating Temperature	0 to +70	°C
Storage Temperature	-55 to +150	°C

Note: Exposure to conditions beyond those listed under Absolute Maximum Ratings may adversely affect the life and reliability of the device.

DC CHARACTERISTICS

(V_{DD}-V_{SS} = 3.0V, Fm = 4.19 MHz, Fs = 32.768 KHz, Ta = 25° C, LCD on; unless otherwise specified)

PARAMETER	SYM.	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Op. Voltage	V_{DD}	-	2.4	-	5.5	V
Op. Current (Crystal type)	I_{OP1}	No load (Ext-V) In dual-clock normal operation	-	0.6	2.5	mA
Op. Current (RC type)	I_{OP2}	No load (Ext-V) In dual-clock normal operation	-	1	4	mA
Op. Current (Crystal type)	I_{OP3}	No load (Ext-V) In dual-clock slow operation and Fm is stopped	-	8.5	20	μA
Hold Current (Crystal type)	I_{HM1}	Hold mode No load (Ext-V) In dual-clock normal operation	-	280	450	μA
Hold Current (RC type)	I_{HM2}	Hold mode No load (Ext-V) In dual-clock normal operation	-	500	600	μA
Hold Current (Crystal type)	I_{HM3}	Hold mode No load (Ext-V) In dual-clock slow operation and Fm is stopped	-	4.0	6	μA
Stop Current (Crystal type)	I_{SM1}	Stop mode No load (Ext-V) In dual-clock normal operation	-	4.0	6	μA
Stop Current (Crystal type)	I_{SM2}	Stop mode No load (Ext-V) In single-clock operation	-	0.1	2	μA
Input Low Voltage	V_{IL}	-	V_{SS}	-	0.3 V_{DD}	V

DC Characteristics, continued

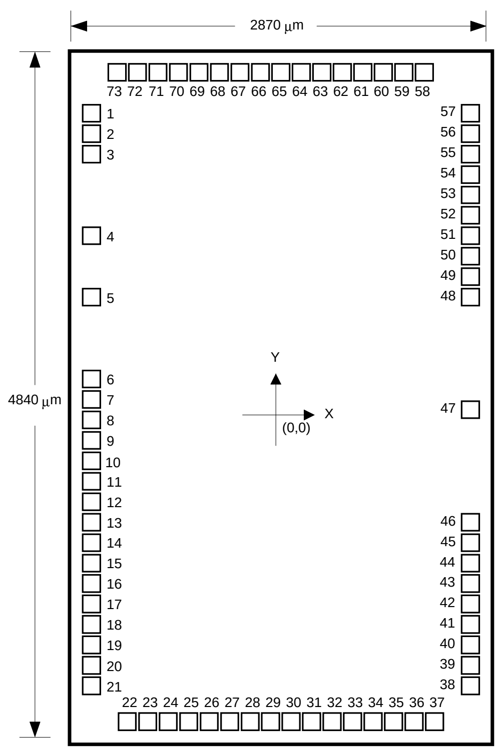
PARAMETER	SYM.	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Input High Voltage	V_{IH}	-	0.7 V_{DD}	-	V_{DD}	V
MFP Output Low Voltage	V_{ML}	I_{OL} = 3.5 mA	-	-	0.4	V
MFP Output High Voltage	V_{MH}	I_{OH} = 3.5 mA	2.4	-	-	V
Port RA, RB Output Low Voltage	V_{ABL}	I_{OL} = 2.0 mA	-	-	0.4	V
Port RA, RB Output high Voltage	V_{ABH}	I_{OH} = 2.0 mA	2.4	-	-	V
LCD Supply Current	I_{LCD}	All Seg. ON	-	-	6	μA
SEG0-SEG31 Sink Current (Used as LCD output)	I_{OL1}	V_{OL} = 0.4V V_{LCD} = 0.0V	0.4	-	-	μA
SEG0-SEG31 Drive Current (Used as LCD output)	I_{OH1}	V_{OH} = 2.4V V_{LCD} = 3.0V	0.3	-	-	μA
Segment output low voltage (Used as DC output)	V_{SL}	I_{OL} = 0.6 mA	-	-	0.4	V
Segment output high voltage (Used as DC output)	V_{SH}	I_{OH} = 3 μA	2.4	-	-	V
Port RE Sink Current	I_{EL}	V_{OL} = 0.9V	9	-	-	mA
Port RE Source Current	I_{EH}	V_{OH} = 2.4V	0.4	1.2	-	mA
Input Port Pull-up Resistor	R_{CD}	Port RC, RD	100	350	1000	KΩ
$\overline{INT}$ Pull-up Resistor	R_{INT}	-	50	250	1000	KΩ
$\overline{RES}$ Pull-up Resistor	R_{RES}	-	20	100	500	KΩ
VPP Pull-down Resistor	R_{VPP}	V_{DD} = 5V	1.5	2	2.5	MΩ
MODE Pull-down Resistor	R_{MODE}	V_{DD} = 5V	1.5	2	2.5	MΩ
DATA Pull-down Resistor	R_{DATA}	V_{DD} = 5V	50	100	150	KΩ

AC CHARACTERISTICS

(V_{DD}-V_{SS} = 3.0V, Ta = 25° C; unless otherwise specified)

PARAMETER	SYM.	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Op. Frequency	FOSC	RC type	-	-	4000	KHz
		Crystal type 1 (Option low speed type)	-	32.768	-	
		Crystal type 2 (Option high speed type)	400	-	4190	
Frequency Deviation by Voltage Drop for RC Oscillator	$\frac{\Delta f}{f}$	$\frac{f(3V)-f(2.4V)}{f(3V)}$	-	-	10	%
Instruction Cycle Time	TI	One machine cycle	-	4/FOSC	-	mS
Reset Active Width	TRAW	FOSC = 32.768 KHz	1	-	-	µS
Interrupt Active Width	TIAW	FOSC = 32.768 KHz	1	-	-	µS

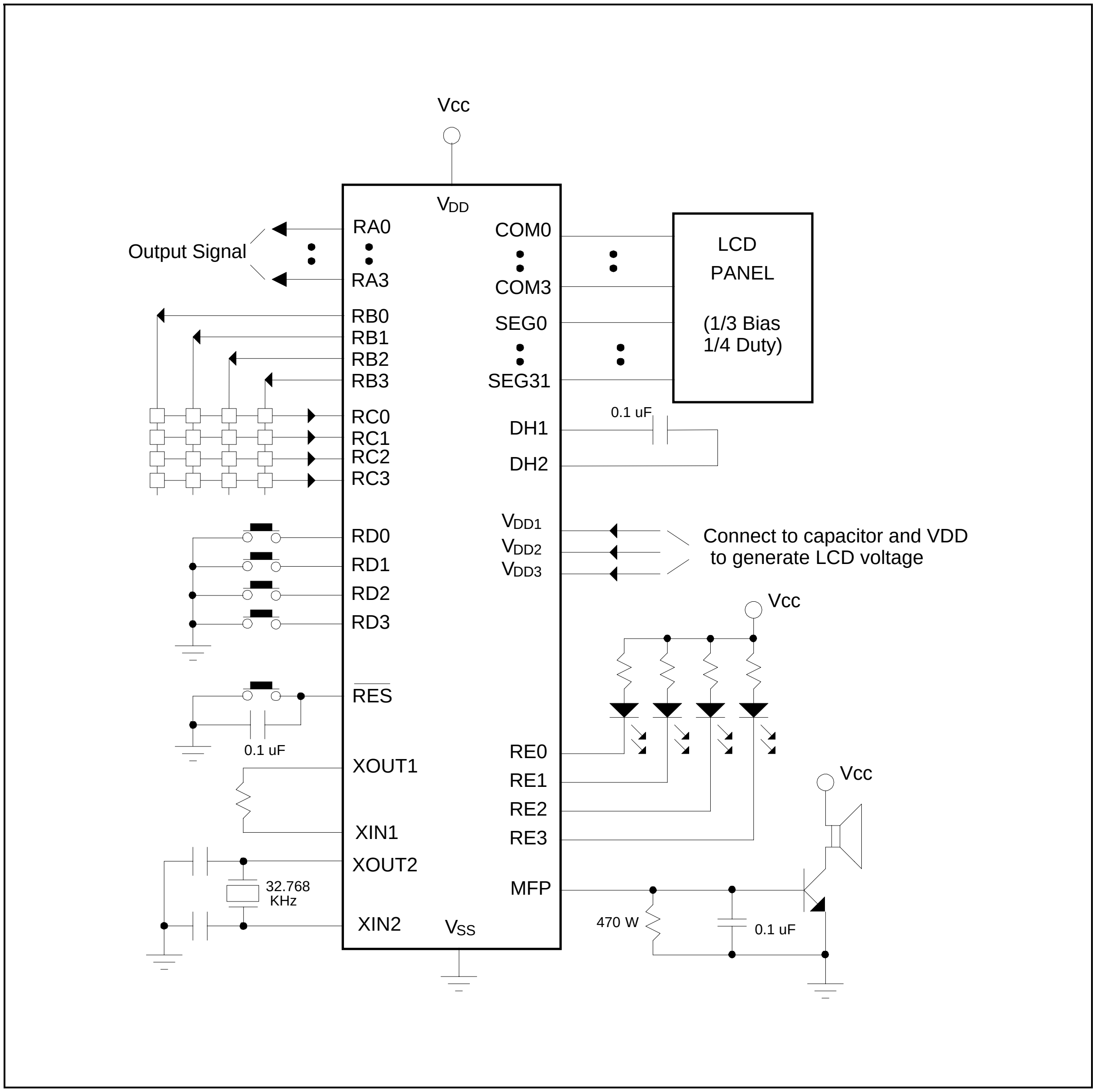
PAD ASSIGNMENT AND POSITIONS



Note: The chip substrate must be connected to system ground (Vss).

PAD NO.	PAD NAME	X	Y	PAD NO.	PAD NAME	X	Y
1	RE2	-1227.80	1810.50	38	SEG28	1226.20	-1913.80
2	RE3	-1227.80	1680.50	39	SEG29	1226.20	-1783.80
3	VSS	-1227.80	1550.50	40	SEG30	1226.20	-1653.80
4	VPP	-1223.25	1079.40	41	SEG31	1226.20	-1523.80
5	DATA	-1234.80	656.85	42	VDD3	1226.20	-1393.80
6	COM3	-1227.80	36.20	43	VDD2	1226.20	-1263.80
7	COM2	-1227.80	-93.80	44	VDD1	1226.20	-1133.80
8	COM1	-1227.80	-223.80	45	DH2	1226.20	-1003.80
9	COM0	-1227.80	-353.80	46	DH1	1226.20	-873.80
10	SEG0	-1227.80	-483.80	47	MODE	1226.20	-112.50
11	SEG1	-1227.80	-613.80	48	XOUT2	1226.20	640.50
12	SEG2	-1227.80	-743.80	49	XIN2	1226.20	770.50
13	SEG3	-1227.80	-873.80	50	VDD	1226.20	900.50
14	SEG4	-1227.80	-1003.80	51	XOUT1	1226.20	1030.50
15	SEG5	-1227.80	-1133.80	52	XIN1	1226.20	1160.50
16	SEG6	-1227.80	-1263.80	53	$\overline{RES}$	1226.20	1290.50
17	SEG7	-1227.80	-1393.80	54	$\overline{INT}$	1226.20	1420.50
18	SEG8	-1227.80	-1523.80	55	MFP	1226.20	1550.50
19	SEG9	-1227.80	-1653.80	56	RA0	1226.20	1680.50
20	SEG10	-1227.80	-1783.80	57	RA1	1226.20	1810.50
21	SEG11	-1227.80	-1913.80	58	RA2	1040.30	2141.70
22	SEG12	-975.80	-2163.80	59	RA3	910.30	2141.70
23	SEG13	-845.80	-2163.80	60	RB0	780.30	2141.70
24	SEG14	-715.80	-2163.80	61	RB1	650.30	2141.70
25	SEG15	-585.80	-2163.80	62	RB2	520.30	2141.70
26	SEG16	-455.80	-2163.80	63	RB3	390.30	2141.70
27	SEG17	-325.80	-2163.80	64	RC0	260.30	2141.70
28	SEG18	-195.80	-2163.80	65	RC1	130.30	2141.70
29	SEG19	-65.80	-2163.80	66	RC2	0.30	2141.70
30	SEG20	64.20	-2163.80	67	RC3	-129.70	2141.70
31	SEG21	194.20	-2163.80	68	RD0	-259.70	2141.70
32	SEG22	324.20	-2163.80	69	RD1	-389.70	2141.70
33	SEG23	454.20	-2163.80	70	RD2	-519.70	2141.70
34	SEG24	584.20	-2163.80	71	RD3	-649.70	2141.70
35	SEG25	714.20	-2163.80	72	RE0	-779.70	2141.70
36	SEG26	844.20	-2163.80	73	RE1	-909.70	2141.70
37	SEG27	974.20	-2163.80				

TYPICAL APPLICATION CIRCUIT



INSTRUCTION SET TABLE

Symbol Description

ACC:	Accumulator
ACC.n:	Accumulator bit n
WR:	Working Register
PAGE:	Page Register
MR0:	Mode Register 0
MR1:	Mode Register 1
PM0:	Port Mode 0
PM1:	Port Mode 1
PM2:	Port Mode 2
PSR0:	Port Status Register 0
PSR1:	Port Status Register 1
R:	Memory (RAM) of address R
LCDR:	LCD data RAM of address LDR
R.n:	Memory bit n of address R
I:	Constant parameter
L:	Branch or jump address
CF:	Carry Flag
ZF:	Zero Flag
PC:	Program Counter
TM0L:	Low nibble of the Timer 0 counter
TM0H:	High nibble of the Timer 0 counter
TM1L:	Low nibble of the Timer 1 counter
TM1H:	High nibble of the Timer 1 counter
TABL:	Low nibble of the look-up table address buffer
TABH:	High nibble of the look-up table address buffer
IEF.n:	Interrupt Enable Flag n
HCF.n:	HOLD mode release Condition Flag n
HEF.n:	HOLD mode release Enable Flag n
SEF.n:	STOP mode wake-up Enable Flag n
PEF.n:	Port Enable Flag n
EVF.n:	Event Flag n
! =:	Not equal
&:	AND
^:	OR
EX:	Exclusive OR
←:	Transfer direction, result
[PAGE*10H+()]:	Contents of address PAGE(bit2, bit1, bit0)*10H+()
[P()]:	Contents of port P

INSTRUCTION SET TABLE 1

Mnemonic		Function	Flag Affected	Cycle
Arithmetic				
ADD	R, ACC	ACC←(R) + (ACC)	ZF, CF	1
ADD	WR, #I	ACC←(WR) + I	ZF, CF	1
ADDR	R, ACC	ACC, R←(R) + (ACC)	ZF, CF	1
ADDR	WR, #I	ACC, WR←(WR) + I	ZF, CF	1
ADC	R, ACC	ACC←(R) + (ACC) + (CF)	ZF, CF	1
ADC	WR, #I	ACC←(WR) + I + (CF)	ZF, CF	1
ADCR	R, ACC	ACC, R←(R) + (ACC) + (CF)	ZF, CF	1
ADCR	WR, #I	ACC, WR←(WR) + I + (CF)	ZF, CF	1
ADU	R, ACC	ACC←(R) + (ACC)	ZF	1
ADU	WR, #I	ACC←(WR) + I	ZF	1
ADUR	R, ACC	ACC, R←(R) + (ACC)	ZF	1
ADUR	WR, #I	ACC, W R←(WR) + I	ZF	1
SUB	R, ACC	ACC←(R) - (ACC)	ZF, CF	1
SUB	WR, #I	ACC←(WR) - I	ZF, CF	1
SUBR	R, ACC	ACC, R←(R) - (ACC)	ZF, CF	1
SUBR	WR, #I	ACC, WR←(WR) - I	ZF, CF	1
SBC	R, ACC	ACC←(R) - (ACC) - (CF)	ZF, CF	1
SBC	WR, #I	ACC←(WR) - I - (CF)	ZF, CF	1
SBCR	R, ACC	ACC, R←(R) - (ACC) - (CF)	ZF, CF	1
SBCR	WR, #I	ACC, WR←(WR) - I - (CF)	ZF, CF	1
INC	R	ACC, R←(R) + 1	ZF, CF	1
DEC	R	ACC, R←(R) - 1	ZF, CF	1

Instruction Set Table 1, continued

Mnemonic		Function	Flag Affected	Cycle
Logic Operations				
ANL	R, ACC	ACC←(R) & (ACC)	ZF	1
ANL	WR, #I	ACC←(WR) & I	ZF	1
ANLR	R, ACC	ACC, R←(R) & (ACC)	ZF	1
ANLR	W, R #I	ACC, WR←(WR) & I	ZF	1
ORL	R, ACC	ACC←(R) ∧ (ACC)	ZF	1
ORL	WR, #I	ACC←(WR) ∧ I	ZF	1
ORLR	R, ACC	ACC, R←(R) ∧ (ACC)	ZF	1
ORLR	WR, #I	ACC, WR←(WR) ∧ I	ZF	1
XRL	R, ACC	ACC←(R) EX (ACC)	ZF	1
XRL	WR, #I	ACC←(WR) EX I	ZF	1
XRLR	R, ACC	ACC, R←(R) EX (ACC)	ZF	1
XRLR	WR, #I	ACC, WR←(WR) EX I	ZF	1
Branch				
JMP	L	PC10~PC0←L10~L0		1
JB0	L	PC10~PC0←L10~L0; if ACC.0 = "1"		1
JB1	L	PC10~PC0←L10~L0; if ACC.1 = "1"		1
JB2	L	PC10~PC0←L10~L0; if ACC.2 = "1"		1
JB3	L	PC10~PC0←L10~L0; if ACC.3 = "1"		1
JZ	L	PC10~PC0←L10~L0; if ACC = 0		1
JNZ	L	PC10~PC0←L10~L0; if ACC ! = 0		1
JC	L	PC10~PC0←L10~L0; if CF = "1"		1
JNC	L	PC10~PC0←L10~L0; if CF != "1"		1
DSKZ	R	ACC, R←(R) - 1; skip if ACC = 0	ZF, CF	1
DSKNZ	R	ACC, R←(R) - 1; skip if ACC != 0	ZF, CF	1
SKB0	R	Skip if R.0 = "1"		1
SKB1	R	Skip if R.1 = "1"		1
SKB2	R	Skip if R.2 = "1"		1
SKB3	R	Skip if R.3 = "1"		1

Instruction Set Table 1, continued

Mnemonic		Function	Flag Affected	Cycle
Data Move				
MOV	WR, R	WR←(R)		1
MOV	R, WR	R←(WR)		1
MOVA	WR, R	ACC, WR←(R)	ZF	1
MOVA	R, WR	ACC, R←(WR)	ZF	1
MOV	R, ACC	R←(ACC)		1
MOV	ACC, R	ACC←(R)	ZF	1
MOV	R, #I	R←I		1
MOV	WR, @R	WR←[PR(bit2, bit1, bit0)x10H +(R)]		2
MOV	@R, WR	[PR(bit2, bit1, bit0)x10H +(R)]←WR		2
MOV	TABL, R	TABL←(R)		1
MOV	TABH, R	TABH←(R)		1
MOVC	R	R←[(TABH) × 10H + (TABL)]		2
MOVC	WR, #I	WR ← [(I6 ~ I0) × 10H + (ACC)]		2
Input & Output				
MOVA	R, RA	ACC, R←[RA]	ZF	1
MOVA	R, RB	ACC, R←[RB]	ZF	1
MOVA	R, RC	ACC, R←[RC]	ZF	1
MOVA	R, RD	ACC, R←[RD]	ZF	1
MOV	RA, R	[RA]←(R)		1
MOV	RB R	[RB]←(R)		1
MOV	RE, R	[RE]←(R)		1
MOV	MFP, #I	[MFP]← I		1
Flag & Register				
MOVA	R, PAGE	ACC, R←PAGE (Page Register)	ZF	1
MOV	PAGE, R	PAGE←(R)		1
MOV	MR0, #I	MR0←I		1
MOV	MR1, #I	MR1←I		1
MOV	PAGE, #I	PAGE←I		1
MOVA	R, CF	ACC.0, R.0←CF	ZF	1
MOV	CF, R	CF←(R.0)	CF	1
MOVA	R,HCFL	ACC, R←HCF0~HCF3	ZF	1
MOVA	R,HCFH	ACC, R←HCF4~HCF7	ZF	1

Instruction Set Table 1, continued

Mnemonic		Function	Flag Affected	Cycle
CLR	PMF, #I	Clear Parameter Flag if In = 1		1 NOTE 2
SET	PMF, #I	Set Parameter Flag if In = 1		1 NOTE 2
MOV	PM0, #I	Port Mode 0← I		1
MOV	PM1, #I	Port Mode 1← I		1
MOV	PM2, #I	Port Mode 2← I		1
CLR	EVF, #I	Clear Event Flag if In = 1		1
MOV	PEF, #I	Set/Reset Port Enable Flag		1
MOV	IEF, #I	Set/Reset Interrupt Enable Flag		1
MOV	HEF, #I	Set/Reset HOLD mode release Enable Flag		1
MOV	SEF, #I	Set/Reset STOP mode wake-up Enable Flag for RC port		1
MOV	SCR, #I	SCR←I		1 NOTE 1
MOVA	R, PSR0	ACC, R←Port Status Register 0	ZF	1
CLR	PSR0	Clear Port Status Register 0		1
SET	CF	Set Carry Flag	CF	1
CLR	CF	Clear Carry Flag	CF	1
CLR	DIVR0	Clear the last 4-bit of the Divider 0		1
CLR	DIVR1	Clear the last 4-bit of the Divider 1		1 NOTE 1
CLR	WDT	Clear WatchDog Timer		1
Shift & Rotate				
SHRC	R	ACC.n, R.n←(R.n+1); ACC.3, R.3←0; CF←R.0	ZF, CF	1
RRC	R	ACC.n, R.n←(R.n+1); ACC.3, R.3←CF; CF←R.0	ZF, CF	1
SHLC	R	ACC.n, R.n←(R.n-1); ACC.0, R.0←0; CF←R.3	ZF, CF	1
RLC	R	ACC.n, R.n←(R.n-1); ACC.0, R.0←CF; CF←R.3	ZF, CF	1

Instruction Set Table 1, continued

Mnemonic		Function	Flag Affected	Cycle
LCD				
MOV	LCDR, #I	LCDR← I		1
MOV	WR, LCDR	WR←(LCDR)		1
MOV	LCDR, WR	LCDR←(WR)		1
MOV	LCDR, ACC	LCDR←(ACC)		1
MOV	LCDM, #I	Select LCD output mode type		1
LCDON		LCD ON		1
LCDOFF		LCD OFF		1
Timer				
MOV	TM0L, R	TM0L←(R)		1
MOV	TM0H, R	TM0H←(R)		1
MOV	TM0, #I	Timer 0 set		1
MOV	TM1L, R	TM1L←(R)		1
MOV	TM1H, R	TM1H←(R)		1
MOV	TM1, #I	Timer 1 set		1
Subroutine				
CALL	L	STACK ← (PC)+1; PC10 ~ PC0 ← L10 ~ L0		1
RTN		(PC)← STACK		1
Other				
HOLD		Enter Hold mode		1
STOP		Enter Stop mode		1
NOP		No Operation		1
EN	INT	Enable Interrupt Function		1
DIS	INT	Disable Interrupt Function		1

Note:

1. These instructions are available in W741C260 body, but inhibited in W741C250 body.
2. The bit0, bit1 and bit2 of PMF are reserved in W741C250 and W741C260 body.

INSTRUCTION SET TABLE 2

ADC R, ACC	Add R to ACC with CF
Machine Code:	0 0 0 0 1 0 0 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) + (ACC) + (CF)
Description:	The contents of the data memory location addressed by R6 to R0, ACC, and CF are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF
ADC WR, #I	**Add immediate data to WR with CF**
Machine Code:	0 0 0 0 1 1 0 0	I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) + I + (CF)
Description:	The contents of the Working Register (WR), I and CF are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF
ADCR R, ACC	**Add R to ACC with CF**
Machine Code:	0 0 0 0 1 0 0 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + (ACC) + (CF)
Description:	The contents of the data memory location addressed by R6 to R0, ACC, and CF are binary added and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

ADCR WR, #I	**Add immediate data to WR with CF**
Machine Code:	0 0 0 0 1 1 0 1 / I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) + I + (CF)
Description:	The contents of the Working Register (WR), I, CF are binary added and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
ADD R, ACC	**Add R to ACC**
Machine Code:	0 0 0 1 1 0 0 0 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF
ADD WR, #I	**Add immediate data to WR**
Machine Code:	0 0 0 1 1 1 0 0 / I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

ADDR R, ACC	**Add R to ACC**
Machine Code:	0 0 0 1 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF
ADDR WR, #I	**Add immediate data to WR**
Machine Code:	0 0 0 1 1 1 0 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
ADU R, ACC	**Add R to ACC and Carry Flag unchange**
Machine Code:	0 0 1 0 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is loaded into the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ADU WR, #I	**Add immediate data to WR and Carry Flag unchange**
Machine Code:	0 0 1 0 1 1 0 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is loaded into the ACC.
Flag Affected:	ZF
ADUR R, ACC	**Add R to ACC and Carry Flag unchange**
Machine Code:	0 0 1 0 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is placed in the ACC and the data memory.
Flag Affected:	ZF
ADUR WR, #I	**Add immediate data to WR and Carry Flag unchange**
Machine Code:	0 0 1 0 1 1 0 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is placed in the WR and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ANL R, ACC	**And R to ACC**
Machine Code:	0 0 1 0 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) & (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ANDed and the result is loaded into the ACC.
Flag Affected:	ZF
ANL WR, #I	**And immediate data to WR**
Machine Code:	0 0 1 0 1 1 1 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) & I
Description:	The contents of the Working Register (WR) and the immediate data I are ANDed and the result is loaded into the ACC.
Flag Affected:	ZF
ANLR R, ACC	**And R to ACC**
Machine Code:	0 0 1 0 1 0 1 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) & (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ANDed and the result is placed in the data memory and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ANLR WR, #I	**And immediate data to WR**
Machine Code:	0 0 1 0 1 1 1 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) & I
Description:	The contents of the Working Register (WR) and the immediate data I are ANDed and the result is placed in the WR and the ACC.
Flag Affected:	ZF
CALL L	**Call subroutine**
Machine Code:	0 1 1 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	STACK ← (PC)+1; PC10 ~ PC0 ← L10 ~ L0
Description:	The next program counter (PC10 to PC0) is saved in the STACK and then the direct address (L10 to L0) is loaded into the program counter. A subroutine is called.
CLR CF	**Clear CF**
Machine Code:	0 1 0 1 0 0 0 0 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Clear CF
Description:	Clear Carry Flag to 0.
Flag Affected:	CF

Instruction Set Table 2, continued

CLR DIVR0	**Reset the last 4 bits of the DIVideR 0**
Machine Code:	0 0 0 1 0 1 1 1 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Reset the last 4 bits of the divider 0
Description:	When this instruction is executed, the last 4 bits of the divider 0 (14 bits) are reset.
CLR DIVR1	**Reset the last 4 bits of the DIVideR 1**
Machine Code:	0 1 0 1 0 1 0 1 | 1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Reset the last 4 bits of the divider 1
Description:	When this instruction is executed, the last 4 bits of the divider 1 (14 bits) are reset. This instruction is available in W741C260 body, but it is inhibited in W741C250 body.
CLR PMF, #I	**Clear ParaMeter Flag**
Machine Code:	0 0 0 1 0 1 1 0 | 1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Clear Parameter Flag
Description:	Description of each flag: I0, I1, I2 : Reserved I3 = 1 : The input clock of the watchdog timer is Fosc/1024.

Instruction Set Table 2, continued

CLR EVF, #I	Clear EVent Flag
Machine Code:	0 1 0 0 0 0 0 0 / I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Clear event flag
Description:	The condition corresponding to the data specified by I7 to I0 is controlled.

I0~I7	Mode after execution of instruction
I0 = 1	EVF0 caused by overflow from the divider 0 is reset.
I1 = 1	EVF1 caused by underflow from the timer 0 is reset.
I2 = 1	EVF2 caused by the signal change at port RC is reset.
I3	Reserved
I4 = 1	EVF4 caused by overflow from the divider 1 is reset for W741C260 body; EVF4 caused by the falling edge signal on INT pin is reset for W741C250 body.
I5 & I6	Reserved
I7 = 1	EVF7 caused by underflow from the timer 1 is reset.

CLR PSR0	Clear Port Status Register 0 (RC port signal change flag)
Machine Code:	0 1 0 0 0 0 1 0 / 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Clear Port Status Register 0 (RC port signal change flag)
Description:	When this instruction is executed, the RC port signal change flag (PSR0) is cleared.

CLR WDT	Reset the last 4 bits of the WatchDog Timer
Machine Code:	0 0 0 1 0 1 1 1 / 1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Reset the last 4 bits of the watchdog timer
Description:	When this instruction is executed, the last 4 bits of the watchdog timer are reset.

Instruction Set Table 2, continued

DEC R	**Decrement R content**
Machine Code:	0 1 0 0 1 0 1 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - 1
Description:	Decrement the data memory content and load result into the ACC and the data memory.
Flag Affected:	CF & ZF
DIS INT	**Disable Interrupt function**
Machine Code:	0 1 0 1 0 0 0 0	1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Disable interrupt function
Description:	Interrupt function is inhibited by executing this instruction.
DSKNZ R	**Decrement R content then skip if ACC ! = 0**
Machine Code:	0 1 0 0 1 0 0 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - 1; PC ← (PC) + 2 if ACC ! = 0
Description:	Decrement the data memory content and load result into the ACC and the data memory. If ACC ! = 0, the program counter is incremented by 2 and produces a skip.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

DSKZ R	**Decrement R content then skip if ACC is zero**
Machine Code:	0 1 0 0 1 0 0 0 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - 1; PC ← (PC) + 2 if ACC = 0
Description:	Decrement the data memory content and load result into the ACC and the data memory. If ACC = 0, the program counter is incremented by 2 and produces a skip.
Flag Affected:	CF & ZF
EN INT	**Enable Interrupt function**
Machine Code:	0 1 0 1 0 0 0 0 / 1 1 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Enable interrupt function
Description:	This instruction enables the interrupt function.
HOLD	**Enter the HOLD mode**
Machine Code:	0 0 0 0 0 0 0 0 / 1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Enter the HOLD mode
Description:	The following two conditions cause the HOLD mode to be released. (1) An interrupt is accepted. (2) The HOLD release condition specified by the HEF is met. In HOLD mode, when an interrupt is accepted the HOLD mode will be released and the interrupt service routine will be executed. After completing the interrupt service routine by executing the RTN instruction, the µC will enter HOLD mode again.

Instruction Set Table 2, continued

INC R	**Increment R content**
Machine Code:	0 1 0 0 1 0 1 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + 1
Description:	Increment the data memory content and load the result into the ACC and the data memory.
Flag Affected:	CF & ZF
JB0 L	**Jump when bit 0 of ACC is "1"**
Machine Code:	1 0 0 0 0 L10 L9 L8	L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.0 = "1"
Description:	If bit 0 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 0 of the ACC is "0," the program counter (PC) is incremented.
JB1 L	**Jump when bit 1 of ACC is "1"**
Machine Code:	1 0 0 1 0 L10 L9 L8	L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.1 = "1"
Description:	If bit 1 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 1 of the ACC is "0," the program counter (PC) is incremented.

Instruction Set Table 2, continued

JB2 L	**Jump when bit 2 of ACC is "1"**
Machine Code:	1 0 1 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.2="1"
Description:	If bit 2 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 2 of the ACC is "0," the program counter (PC) is incremented.
JB3 L	**Jump when bit 3 of ACC is "1"**
Machine Code:	1 0 1 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.3 = "1"
Description:	If bit 3 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 3 of the ACC is "0," the program counter (PC) is incremented.
JC L	**Jump when CF is "1"**
Machine Code:	1 1 1 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if CF = "1"
Description:	If CF is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If the CF is "0," the program counter (PC) is incremented.

Instruction Set Table 2, continued

JMP L	Jump absolutely
Machine Code:	0 1 1 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0
Description:	PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and an unconditional jump occurs.
JNC L	**Jump when CF is not "1"**
Machine Code:	1 1 0 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if CF = "0"
Description:	If CF is "0," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If CF is "1," the program counter (PC) is incremented.
JNZ L	**Jump when ACC is not zero**
Machine Code:	1 1 0 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC ! = 0
Description:	If the ACC is not zero, PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If the ACC is zero, the program counter (PC) is incremented.

Instruction Set Table 2, continued

JZ L	**Jump when ACC is zero**
Machine Code:	1 1 1 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC = 0
Description:	If the ACC is zero, PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If the ACC is not zero, the program counter (PC) is incremented.
LCDON	**LCD ON**
Machine Code:	0 0 0 0 0 0 1 0 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	LCD ON
Description:	Turn on LCD display.
LCDOFF	**LCD OFF**
Machine Code:	0 0 0 0 0 0 1 0 | 1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	LCD OFF
Description:	Turn off LCD display.

Instruction Set Table 2, continued

MOV ACC, R	**Move R content to ACC**
Machine Code:	0 1 0 0 1 1 1 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the ACC.
Flag Affected:	ZF
MOV CF, R	**Move R.0 content to CF**
Machine Code:	0 1 0 1 1 0 0 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	CF ← (R.0)
Description:	The bit 0 content of the data memory location addressed by R6 to R0 is loaded into CF.
Flag Affected:	CF

Instruction Set Table 2, continued

MOV HEF, #I	Set/Reset Hold mode release Enable Flag
Machine Code:	0 1 0 0 0 0 0 1 | I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Hold mode release enable flag control
Description:	

I0~I7	Operation
I0 = 1	The HEF0 is set so that overflow from the divider 0 caused the HOLD mode to be released.
I1 = 1	The HEF1 is set so that underflow from the Timer 0 caused the HOLD mode to be released.
I2 = 1	The HEF2 is set so that signal change at port RC caused the HOLD mode to be released.
I3	Reserved
I4 = 1	The HEF4 is set so that overflow from the divider 1 caused the HOLD mode to be released for W741C260 body; The HEF4 is set so that the falling edge signal at the INT pin caused the HOLD mode to be released for W741C250 body.
I5 & I6	Reserved
I7 = 1	The HEF7 is set so that underflow from the Timer 1 caused the HOLD mode to be released.

Instruction Set Table 2, continued

MOV IEF, #I	**Set/Reset Interrupt Enable Flag**
Machine Code:	0 1 0 1 0 0 0 1 | I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Interrupt Enable flag Control
Description:	The interrupt enable flag corresponding to the data specified by I7-I0 is controlled:

I0~I7	Operation
I0 = 1	The IEF0 is set so that interrupt 0 (overflow from the divider 0) is accepted.
I1 = 1	The IEF1 is set so that interrupt 1 (underflow from the Timer 0) is accepted.
I2 = 1	The IEF2 is set so that interrupt 2 (signal change at port RC) is accepted.
I3	Reserved
I4 = 1	The IEF4 is set so that interrupt 4 (overflow from the divider 1) is accepted for W741C260 body; the IEF4 is set so that interrupt 4 (falling edge signal at the INT pin) is accepted for the W741C250 body.
I5 & I6	Reserved
I7 = 1	The IEF7 is set so that interrupt 7 (underflow from the Timer 1) is accepted.

MOV LCDM, #I	**Select LCD output Mode type**
Machine Code:	0 0 0 0 0 0 1 1 | I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Select LCD output mode type
Description:	When LCD output pins are set to DC output mode, user can select CMOS or NMOS as output type. I0~I7 = 0 => CMOS type; I0~I7 = 1 => NMOS type.

Instruction Set Table 2, continued

MOV LCDR, ACC	**Move ACC content to LCDR**
Machine Code:	0 0 0 0 0 1 1 D4 | D3 D2 D1 D0 0 0 0 0
Machine Cycle:	1
Operation:	LCDR ← (ACC)
Description:	The contents of the ACC are loaded to the LCD data RAM (LCDR) location addressed by D4 to D0.
MOV LCDR, WR	**Load WR content to LCDR**
Machine Code:	0 1 0 0 0 1 0 D4 | D3 D2 D1 D0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	LCDR ← (WR)
Description:	The contents of the WR are loaded to the LCD data RAM (LCDR) location addressed by D4 to D0.
MOV LCDR, #I	**Load immediate data to LCDR**
Machine Code:	0 0 0 0 0 1 0 D4 | D3 D2 D1 D0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	LCDR ← I
Description:	The immediate data I are loaded to the LCD data RAM (LCDR) location addressed by D4 to D0.

Instruction Set Table 2, continued

MOV MFP, #I	**Modulation Frequency Pulse generator**
Machine Code:	0 0 0 1 0 0 1 0 | I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	[MFP] ← I
Description:	If the bit 2 of MR1 is "0," the waveform specified by I7 to I0 is delivered at the MFP output pin (MFP). The relation between the waveform and immediate data I is as follows:

I5~I0	I0 = 1	I1 = 1	I2 = 1	I3 = 1	I4 = 1	I5 = 1
Signal	$\frac{Fosc}{256}$	$\frac{Fosc}{512}$	$\frac{Fosc}{4096}$	$\frac{Fosc}{8192}$	$\frac{Fosc}{16384}$	$\frac{Fosc}{32768}$

I7	I6	Signal
0	0	Low
0	1	High
1	0	Fosc/16
1	1	Fosc/8

Instruction Set Table 2, continued

MOV MR0, #I	**Load immediate data to Mode Register 0 (MR0)**
Machine Code:	0 0 0 1 0 0 1 1 | 1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	MR0 ← I
Description:	The immediate data I are loaded to the MR0.
	MR0 bits description:

bit 0	= 0 The fundamental frequency of Timer 0 is Fosc/4 = 1 The fundamental frequency of Timer 0 is Fosc/1024
bit 1	Reserved
bit 2	Reserved
bit 3	= 0 Timer 0 stop down-counting = 1 Timer 0 start down-counting

Instruction Set Table 2, continued

MOV MR1, #I	**Load immediate data to Mode Register 1 (MR1)**
Machine Code:	0 0 0 1 0 0 1 1 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	MR1 ← I
Description:	The immediate data I are loaded to the MR1. MR1 bit description:

bit0	= 0 The internal fundamental frequency of Timer 1 is Fosc = 1 The internal fundamental frequency of Timer 1 is Fosc/64
bit1	= 0 The fundamental frequency source of Timer 1 is internal clock = 1 The fundamental frequency source of Timer 1 is external clock via RC.0 input pin
bit2	= 0 The specified waveform of the MFP generator is delivered at the MFP output pin = 1 The specified frequency of the Timer 1 is delivered at the MFP output pin
bit3	= 0 Timer 1 stop down-counting = 1 Timer 1 start down-counting

Instruction Set Table 2, continued

MOV PAGE, #I	**Load immediate data to Page Register**
Machine Code:	0 1 0 1 0 1 1 0	1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Page Register ← I
Description:	The immediate data I are loaded to the PR. Bit 3 is reserved. Bit 0, bit 1, and bit 2 indirect addressing mode preselect bits:

bit2	bit1	bit0	
0	0	0	= Page 0 (00H~0FH)
0	0	1	= Page 1 (10H~1FH)
0	1	0	= Page 2 (20H~2FH)
0	1	1	= Page 3 (30H~3FH)
1	0	0	= Page 4 (40H~4FH)
1	0	1	= Page 5 (50H~5FH)
1	1	0	= Page 6 (60H~6FH)
1	1	1	= Page 7 (70H~7FH)

Instruction Set Table 2, continued

MOV PEF, #I	**Set/Reset Port Enable Flag**
Machine Code:	0 1 0 0 0 0 1 1 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Port enable flag control
Description:	The data specified by I can cause HOLD mode to be released or an interrupt to occur. The signal change on port RC is specified.

I0~I7	Signal change at port RC
I0 = 1	RC0
I1 = 1	RC1
I2 = 1	RC2
I3 = 1	RC3

MOV PM0, #I	**Set/Reset Port Mode 0 register**
Machine Code:	0 1 0 1 0 0 1 1 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Set/Reset Port mode 0 register
Description:	I0 = 0: RA port is CMOS type; I0 = 1: RA port is NMOS type. I1 = 0: RB port is CMOS type; I1 = 1: RB port is NMOS type. I2 = 0: RC port pull-high resistor is disabled; I2 = 1: RC port pull-high resistor is enabled. I3 = 0: RD port pull-high resistor is disabled; I3 = 1: RD port pull-high resistor is enabled.

Instruction set table 2, continued

MOV PM1, #I	**RA port independent Input/Output control**
Machine Code:	0 1 0 1 0 1 1 1 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	RA port 4 pins input/output control is independent.
Description:	I0 = 0: RA.0 is output pin; I0 = 1: RA.0 is input pin. I1 = 0: RA.1 is output pin; I1 = 1: RA.1 is input pin. I2 = 0: RA.2 is output pin; I2 = 1: RA.2 is input pin. I3 = 0: RA.3 is output pin; I3 = 1: RA.3 is input pin. Default condition RA port is input mode (PM = 1111B).
MOV PM2, #I	**RB port independent Input/Output control**
Machine Code:	0 1 0 1 0 1 1 1 | 1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	RB port 4 pins input/output control is independent.
Description:	I0 = 0: RB.0 is output pin; I0 = 1: RB.0 is input pin. I1 = 0: RB.1 is output pin; I1 = 1: RB.1 is input pin. I2 = 0: RB.2 is output pin; I2 = 1: RB.2 is input pin. I3 = 0: RB.3 is output pin; I3 = 1: RB.3 is input pin. Default condition RB port is input mode (PM2 = 1111B).

Instruction Set Table 2, continued

MOV R, ACC	**Move ACC content to R**
Machine Code:	0 1 0 1 1 0 0 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	R ← (ACC)
Description:	The contents of the ACC are loaded to the data memory location addressed by R6 to R0.
MOVA R, RA	**Input RA port data to ACC & R**
Machine Code:	0 1 0 1 1 0 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RA]
Description:	The data on port RA are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOVA R, RB	**Input RB port data to ACC & R**
Machine Code:	0 1 0 1 1 0 1 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RB]
Description:	The data on port RB are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

MOVA R, RC	**Input RC port data to ACC & R**
Machine Code:	0 1 0 0 1 0 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RC]
Description:	The input data on the input port RC are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOVA R, RD	**Input RD port data to ACC & R**
Machine Code:	0 1 0 0 1 0 1 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RD]
Description:	The input data on the input port RD are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOV R, WR	**Move WR content to R**
Machine Code:	1 1 1 1 1 W3 W2 W1	W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	R ← (WR)
Description:	The contents of the WR are loaded to the data memory location addressed by R6 to R0.

Instruction Set Table 2, continued

MOV R, #I	**Load immediate data to R**
Machine Code:	1 0 1 1 1 I3 I2 I1 | I0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	R ← I
Description:	The immediate data I are loaded to the data memory location addressed by R6 to R0.
MOV RA, R	**Output R content to RA port**
Machine Code:	0 1 0 1 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	[RA] ← (R)
Description:	The data in the data memory location addressed by R6 to R0 are output to the port RA.
MOV RB, R	**Output R content to RB port**
Machine Code:	0 1 0 1 1 0 1 0 | 1 R6 R5 R4 R3 R2 R1 R0 6
Machine Cycle:	1
Operation:	[RB] ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are output to the port RB.
MOV RE, R	**Output R content to port RE**
Machine Code:	0 1 0 1 1 1 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	[RE] ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are output to port RE.

Instruction Set Table 2, continued

MOV SCR, #I	**System Clock Register control**
Machine Code:	0 1 0 1 0 1 0 0 / 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	System clock control
Description:	If the operation mode is the dual clock operation selected by the option codes, the system clock and oscillator can be arranged by controlling the system clock register. This command is just for the W741C260 body. SCR bits decription:

Bit 0	= 0, Fosc=Fm = 1, Fosc=Fs
Bit 1	= 0, main oscillator is enabled = 1, main oscillator is disabled
Bit 2	Reserved
Bit 3	= 0, divider 1 is 14-stage = 1, divider 1 is 13-stage

MOV SEF, #I	**Set/Reset STOP mode waked-up Enable Flag for port RC**
Machine Code:	0 1 0 1 0 0 1 0 / 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Set/reset STOP mode wake-up enable flag for port RC
Description:	The data specified by I cause a wake-up from the STOP mode. The falling-edge signal on port RC can be specified independently.

I0~I7	Falling edge signal at port RC
I0 = 1	RC0
I1 = 1	RC1
I2 = 1	RC2
I3 = 1	RC3

Instruction Set Table 2, continued

MOV TM0, #I	**Timer 0 set**
Machine Code:	0 0 0 1 0 0 0 0 | I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Timer 0 set
Description:	The data specified by I7 to I0 is loaded to the Timer 0 to start the timer.
MOV TM0L, R	**Move R content to TM0L**
Machine Code:	0 0 0 1 0 1 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM0L ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the TM0L.
MOV TM0H, R	**Move R content to TM0H**
Machine code:	0 0 0 1 0 1 0 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM0H ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the TM0H.
MOV TM1, #I	**Timer 1 set**
Machine Code:	0 0 0 1 0 0 0 1 | I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Timer 1 set
Description:	The data specified by I7 to I0 is loaded to the Timer 1 to start the timer.

Instruction Set Table 2, continued

MOV TM1L, R	**Move R content to TM1L**
Machine Code:	0 0 0 1 0 1 0 1 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM1L ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the TM1L.
MOV TM1H, R	**Move R content to TM1H**
Machine code:	0 0 0 1 0 1 0 1 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM1H ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the TM1H.
MOV WR, LCDR	**Load LCDR content to WR**
Machine Code:	0 1 0 0 0 1 1 D4 D3 D2 D1 D0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	WR ← (LCDR)
Description:	The contents of the LCD data RAM location addressed by D4 to D0 are loaded to the WR.
MOV WR, R	**Move R content to WR**
Machine Code:	1 1 1 0 1 W3 W2 W1 W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	WR ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded to the WR.

Instruction Set Table 2, continued

MOV WR, @R	Indirect load from R to WR
Machine Code:	1 1 0 0 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	2
Operation:	WR ← [PR (bit2, bit1, bit0) × 10H + (R)]
Description:	The data memory contents of address [PR (bit2, bit1, bit0) × 10H + (R)] are loaded to the WR.
MOV @R, WR	**Indirect load from WR to R**
Machine Code:	1 1 0 1 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	2
Operation:	[PR (bit2, bit1, bit0) × 10H + (R)] ← WR
Description:	The contents of the WR are loaded to the data memory location addressed by [PR (bit2, bit1, bit0) × 10H + (R)] .
MOV PAGE, R	**Move R content to Page Register**
Machine Code:	0 1 0 1 1 1 1 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PR ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded to the PR.

Instruction Set Table 2, continued

MOVA R, CF	**Move CF content to ACC.0 & R.0**
Machine Code:	0 1 0 1 1 0 0 1 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.0, R.0 ← (CF)
Description:	The content of CF is loaded to bit 0 of the data memory location addressed by R6 to R0 and the ACC. The other bits of the data memory and ACC are reset to "0."
Flag Affected:	ZF
MOVA R, HCFH	**Move HCF4~7 to ACC & R**
Machine Code:	0 1 0 0 1 0 0 1 / 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← HCF4~7
Description:	The contents of HCF bit 4 to bit 7 (HCF4 to HCF7) are loaded to the data memory location addressed by R6 to R0 and the ACC. The ACC contents and the meaning of the bits after execution of this instruction are as follows:
Flag Affected:	ZF

Bit 0	HCF4: "1" when the HOLD mode is released by overflow from Divide r 1 for the W741C260 body; HCF4: "1" when the HOLD mode is released by the falling edge signal at the INT pin for the W741C250 body.
Bit 1	HCF7: "1" when the HOLD mode is released by underflow from Timer 1.
Bit 2	Reserved.
Bit 3	Reserved.

Instruction Set Table 2, continued

MOVA R, HCFL	Move HCF0~3 to ACC & R
Machine Code:	0 1 0 0 1 0 0 1 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← HCF0~3
Description:	The contents of HCF bit 0 to bit 3 (HCF0 to HCF3) are loaded to the data memory location addressed by R6 to R0 and the ACC. The ACC contents and the meaning of the bits after execution of this instruction are as follows: Bit 0: HCF0: "1" when the HOLD mode is released by overflow from the Divider 0. Bit 1: HCF1: "1" when the HOLD mode is released by underflow from Timer 0. Bit 2: HCF2: "1" when the HOLD mode is released by a signal change on port RC. Bit 3: Reserved.
Flag Affected:	ZF
MOVA R, PAGE	**Move Page Register content to ACC & R**
Machine Code:	0 1 0 1 1 1 1 1 / 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← (Page Register)
Description:	The contents of the Page Register (PR) are loaded to the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

MOVA R, PSR0	**Move Port Status Register 0 content to ACC & R**
Machine Code:	0 1 0 0 1 1 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← RC port signal change flag (PSR0)
Description:	The contents of the RC port signal change flag (PSR0) are loaded to the data memory location addressed by R6 to R0 and the ACC. When the signal changes on any pin of the RC port, the corresponding signal change flag should be set to 1. Otherwise, it should be 0.
Flag Affected:	ZF
MOVA R, WR	**Move WR content to ACC & R**
Machine Code:	0 1 1 1 1 W3 W2 W1	W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (WR)
Description:	The contents of the WR are loaded to the ACC and the data memory location addressed by R6 to R0.
Flag Affected:	ZF
MOVA WR, R	**Move R content to ACC & WR**
Machine Code:	0 1 1 0 1 W3 W2 W1	W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, WR ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded to the WR and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

MOV TABL, R	**Move R content to TABL**
Machine Code:	1 0 0 1 1 0 0 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TABL ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the TABL.
MOV TABH, R	**Move R content to TABH**
Machine code:	1 0 0 1 1 0 0 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TABH ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the TABH.
MOVC R	**Move look-up table ROM addressed by TABL and TABH to R**
Machine code:	1 0 0 1 1 0 0 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	2
Operation:	WR ← [(TABH) × 10H + (TABL)]
Description:	The contents of the look-up table ROM location addressed by TABH and TABL are loaded to R.

Instruction Set Table 2, continued

MOVC WR, #I	**Move look-up table ROM addressed by #I and ACC to WR**
Machine code:	1 0 1 0 1 W3 W2 W1 | W0 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	2
Operation:	WR ← [(I6 ~ I0) × 10H + (ACC)]
Description:	The contents of the look-up table ROM location addressed by I6 to I0 and the ACC are loaded to R.
NOP	**No Operation**
Machine Code:	0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	No Operation
ORL R, ACC	**OR R to ACC**
Machine Code:	0 0 1 1 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) ∧ (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ORed and the result is loaded into the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ORL WR , #I	**OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 1 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) ∧ I
Description:	The contents of the Working Register (WR) and the immediate data I are ORed and the result is loaded into the ACC.
Flag Affected:	ZF
ORLR R, ACC	**OR R to ACC**
Machine Code:	0 0 1 1 1 0 1 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) ∧ (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ORed and the result is placed in the data memory and the ACC.
Flag Affected:	ZF
ORLR WR , #I	**OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 1 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) ∧ I
Description:	The contents of the Working Register(WR) and the immediate data I are ORed and the result is placed in the WR and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

RLC R	**Rotate Left R with CF**
Machine Code:	0 1 0 0 1 1 0 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n-1); ACC.0, R.0 ← CF; CF ← R.3
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are rotated left one bit, bit 3 is rotated into CF, and CF rotated into bit 0 (LSB). The same contents are loaded into the ACC.
Flag Affected:	CF & ZF
RRC R	**Rotate Right R with CF**
Machine Code:	0 1 0 0 1 1 0 1 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n+1); ACC.3, R.3 ← CF; CF ← R.0
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are rotated right one bit, bit 0 is rotated into CF, and CF is rotated into bit 3 (MSB). The same contents are loaded into the ACC.
Flag Affected:	CF & ZF
RTN	**Return from subroutine**
Machine Code:	0 0 0 0 0 0 0 1 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	(PC) ← STACK
Description:	The program counter (PC10 to PC0) is restored from the stack. A return from a subroutine occurs.

Instruction Set Table 2, continued

SBC R, ACC	Subtract ACC from R with Borrow
Machine Code:	0 0 0 0 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) - (ACC) - (CF)
Description:	The contents of the ACC and CF are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is loaded into the ACC.
Flag Affected:	CF & ZF
SBC WR, #I	**Subtract immediate data from WR with Borrow**
Machine Code:	0 0 0 0 1 1 1 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) - I - (CF)
Description:	The immediate data I and CF are binary subtracted from the contents of the WR and the result is loaded into the ACC.
Flag Affected:	CF & ZF
SBCR R, ACC	**Subtract ACC from R with Borrow**
Machine Code:	0 0 0 0 1 0 1 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - (ACC) - (CF)
Description:	The contents of the ACC and CF are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

SBCR WR, #I	**Subtract immediate data from WR with Borrow**
Machine Code:	0 0 0 0 1 1 1 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, R ← (WR) - I - (CF)
Description:	The immediate data I and CF are binary subtracted from the contents of the WR and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
SET CF	**Set CF**
Machine Code:	0 1 0 1 0 0 0 0 | 0 1 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Set CF
Description:	Set Carry Flag to 1.
Flag Affected:	CF
SET PMF, #I	**Set ParaMeter Flag**
Machine Code:	0 0 0 1 0 1 1 0 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Set Parameter Flag
Description:	Description of each flag: I0, I1, I2 : Reserved I3 = 1 : The input clock of the watchdog timer is Fosc/16384.

Instruction Set Table 2, continued

SHLC R	**SHift Left R with CF and LSB = 0**
Machine Code:	0 1 0 0 1 1 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n-1); ACC.0, R.0 ← 0; CF ← R.3
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are shifted left one bit, but bit 3 is shifted into CF, and bit 0 (LSB) is replaced with "0." The same contents are loaded into the ACC.
Flag Affected:	CF & ZF
SHRC R	**SHift Right R with CF and MSB = 0**
Machine Code:	0 1 0 0 1 1 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n+1); ACC.3, R.3 ← 0; CF ← R.0
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are shifted right one bit, but bit 0 is shifted into CF, and bit 3 (MSB) is replaced with "0." The same contents are loaded into the ACC.
Flag Affected:	CF & ZF
SKB0 R	**If bit 0 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.0 = 1“1”
Description:	If bit 0 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 0 of R is not equal to 1, the program counter (PC) is incremented.

Instruction Set Table 2, continued

SKB1 R	**If bit 1 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 0 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.1 = 1"1"
Description:	If bit 1 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 1 of R is not equal to 1, the program counter (PC) is incremented.
SKB2 R	**If bit 2 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 1 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.2 = 1"1"
Description:	If bit 2 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 2 of R is not equal to 1. The program counter (PC) is incremented.
SKB3 R	**If bit 3 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 1 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.3 = 1"1"
Description:	If bit 3 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 3 of R is not equal to 1, the program counter (PC) is incremented.

Instruction Set Table 2, continued

STOP	**Enter the STOP mode**
Machine Code:	0 0 0 0 0 0 0 0 | 1 1 0 0 0 0 0 0
Machine Cycle:	1
Operation:	STOP oscillator
Description:	Device enters STOP mode. When the falling edge signal of RC port is accepted, the µC will wake up and execute the next instruction.
SUB R, ACC	**Subtract ACC from R**
Machine Code:	0 0 0 1 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) - (ACC)
Description:	The contents of the ACC are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is loaded into the ACC.
Flag Affected:	CF & ZF
SUB WR , #I	**Subtract immediate data from WR**
Machine Code:	0 0 0 1 1 1 1 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) - I
Description:	The immediate data I are binary subtracted from the contents of the WR and the result is loaded into the ACC.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

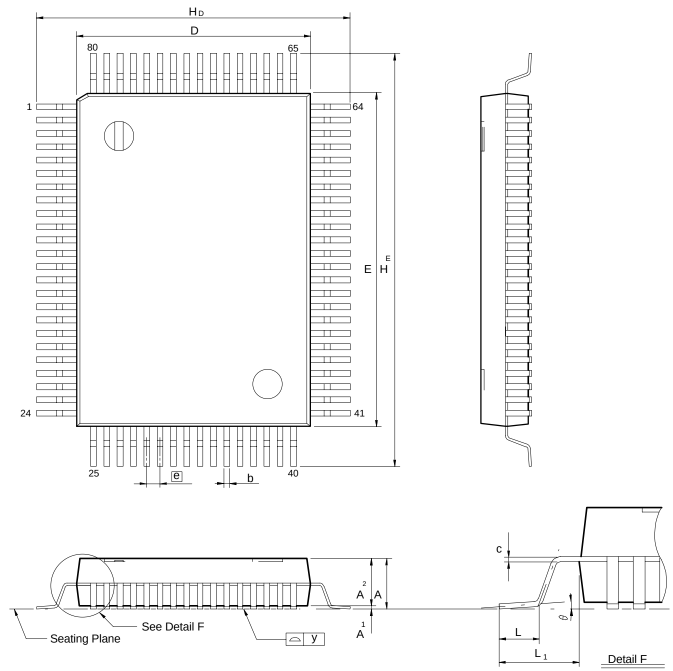
SUBR R, ACC	**Subtract ACC from R**
Machine Code:	0 0 0 1 1 0 1 1 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - (ACC)
Description:	The contents of the ACC are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF
SUBR WR, #I	**Subtract immediate data from WR**
Machine Code:	0 0 0 1 1 1 1 1 / I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) - I
Description:	The immediate data I are binary subtracted from the contents of the WR and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
XRL R, ACC	**Exclusive OR R to ACC**
Machine Code:	0 0 1 1 1 0 0 0 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) EX (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are exclusive-ORed and the result is loaded into the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

XRL WR, #I	**Exclusive OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 0 0 / I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) EX I
Description:	The contents of the Working Register (WR) and the immediate data I are exclusive-ORed and the result is loaded into the ACC.
Flag Affected:	ZF
XRLR R, ACC	**Exclusive OR R to ACC**
Machine Code:	0 0 1 1 1 0 0 1 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) EX (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are exclusive-ORed and the result is placed in the data memory and the ACC.
Flag Affected:	ZF
XRLR WR, #I	**Exclusive OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 0 1 / I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) EX I
Description:	The contents of the Working Register(WR) and the immediate data I are exclusive-ORed and the result is placed in the WR and the ACC.
Flag Affected:	ZF

PACKAGE DIMENSIONS

80-Lead QFP



Symbol	Dimension in inches			Dimension in mm		
	Min.	Nom	Max.	Min.	Nom	Max.
A	—	—	0.130	—	—	3.30
A_1	0.004	—	—	0.10	—	—
A_2	0.107	0.112	0.117	2.73	2.85	2.97
b	0.012	0.014	0.018	0.30	0.35	0.45
c	0.004	0.006	0.010	0.10	0.15	0.25
D	0.546	0.551	0.556	13.87	14.00	14.13
E	0.782	0.787	0.792	19.87	20.00	20.13
e	0.025	0.031	0.037	0.65	0.80	0.95
H_D	0.728	0.740	0.752	18.49	18.80	19.10
H_E	0.964	0.976	0.988	24.49	24.80	25.10
L	0.039	0.047	0.055	1.00	1.20	1.40
L_1	0.087	0.094	0.103	2.21	2.40	2.62
y	—	—	0.004	—	—	0.10
θ	0°	—	12°	0°	—	12°

NOTES:

Headquarters
No. 4, Creation Rd. III,
Science-Based Industrial Park,
Hsinchu, Taiwan
TEL: 886-3-5770066
FAX: 886-3-5792766
http://www.winbond.com.tw/
Voice & Fax-on-demand: 886-2-27197006

Taipei Office
11F, No. 115, Sec. 3, Min-Sheng East Rd.,
Taipei, Taiwan
TEL: 886-2-27190505
FAX: 886-2-27197502

Winbond Electronics (H.K.) Ltd.
Rm. 803, World Trade Square, Tower II,
123 Hoi Bun Rd., Kwun Tong,
Kowloon, Hong Kong
TEL: 852-27513100
FAX: 852-27552064

Winbond Electronics North America Corp.
Winbond Memory Lab.
Winbond Microelectronics Corp.
Winbond Systems Lab.
2727 N. First Street, San Jose,
CA 95134, U.S.A.
TEL: 408-9436666
FAX: 408-5441798