

# Preliminary W742E81A/W742C81A



## 4-BIT MICROCONTROLLER

### 1. GENERAL DESCRIPTION

The W742E81A/W742C81A is a high-performance 4-bit microcontroller ( $\mu\text{C}$ ) that provides an LCD driver. The device contains a 4-bit ALU, two 8-bit timers, two dividers (for two oscillators) in dual-clock operation, a  $40 \times 4$  LCD driver, six 4-bit I/O ports (including 1 output port for LED driving), and one channel DTMF generator. There are also five interrupt sources and 16-levels subroutine nesting for interrupt applications. The W742E81A/W742C81A operates on very low current and has two power reduction modes, that is the dual-clock slow operation and STOP mode, which help to minimize power dissipation.

### 2. FEATURES

- Operating voltage: 2.4V–3.8V
- Dual-clock operation or single-clock operation (By option)
- Main-oscillator
  - Connect to 3.58 MHz crystal or 400 KHz that can be selected by option code
  - Crystal or RC oscillator can be selected by code option (W742E81A)
  - Connect to 2 MHz typical RC oscillator (W742C81A)
- Sub-oscillator
  - Connect to 32768 Hz crystal only
- Memory
  - 16384 x 16 bits program flash EEPROM (including 64K x 4 bit look-up table)
  - 2048 x 4 bits data RAM (including 16 nibbles x 16 pages working registers)
  - 40 x 4 LCD data RAM
- 24 input/output pins
  - Port for input only: 1 ports/4 pins(RC)
  - Input/output ports: 3 ports/12 pins(RA, RB & RD)
  - High sink current output port for LED driving: 1 port /4 pins(RE)
  - Port for output only: 1 port/ 4 pins(RF)
- Power-down mode
  - Hold function: no operation (main-oscillator and sub-oscillator still operate)
  - Stop function: no operation (main-oscillator and sub-oscillator are stopped)
  - Dual-clock slow operation mode: system is operated by the sub-oscillator ( $F_{\text{OSC}}=F_s$  and  $F_m$  is stopped)
- Five types of interrupts
  - Four internal interrupts (Divider0, Divider1, Timer 0, Timer 1)
  - One external interrupts (RC Port)

# Preliminary W742E81A/W742C81A



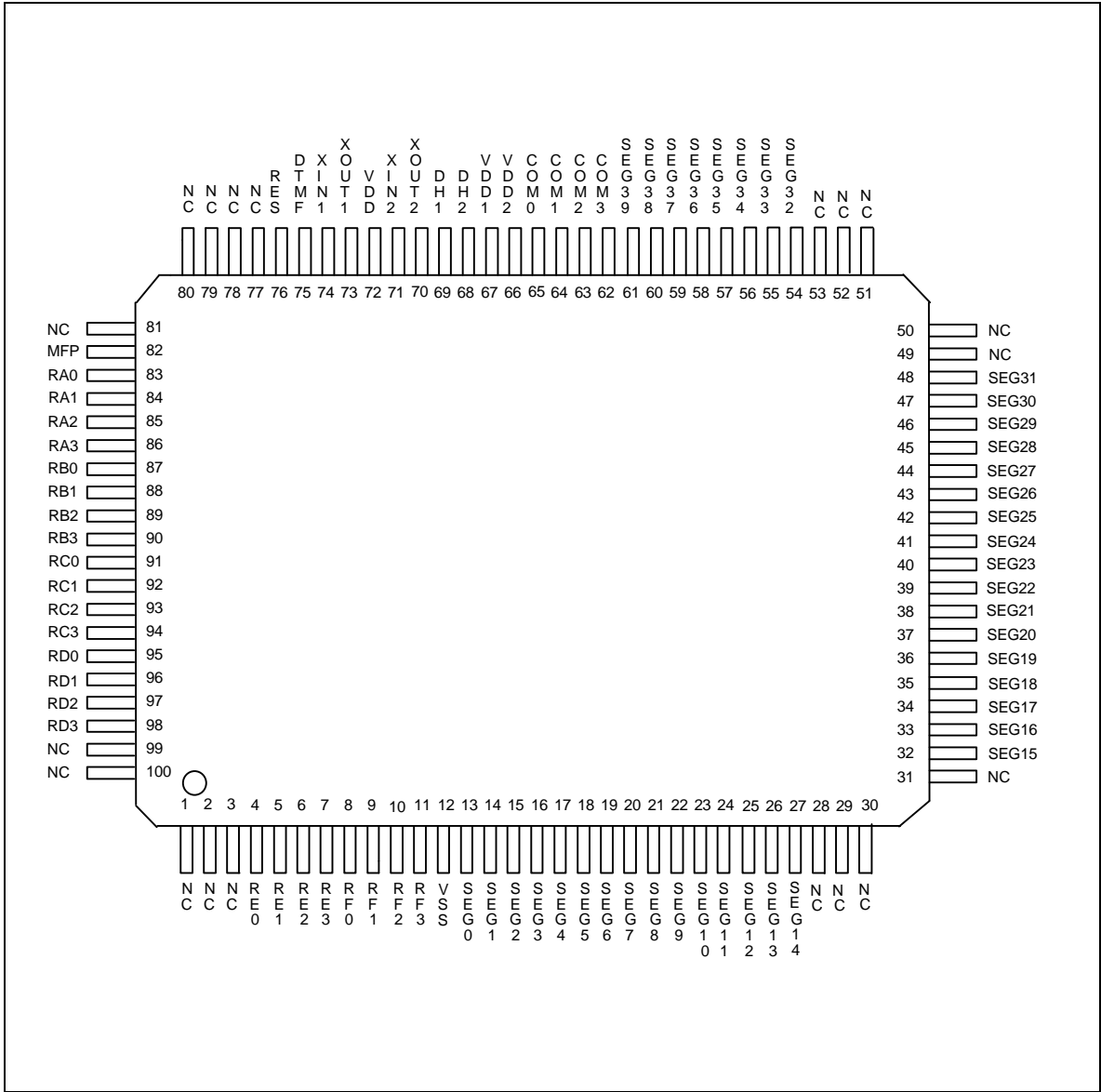
- LCD driver output
  - 40 segments x 4 commons
  - 1/4 duty 1/3 bias driving mode
  - Clock source should be the sub-oscillator clock in the dual-clock operation mode
- MFP output pin
  - Output is software selectable as modulating or nonmodulating frequency
  - Works as frequency output specified by Timer 1
- DTMF output pin
  - Output is one channel Dual Tone Multi-Frequency signal for dialling
- 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 (dual-clock mode) or the  $F_{osc}/128$  (single-clock mode)
- Two built-in 8-bit programmable countdown timers
  - Timer 0: one of two internal clock frequencies ( $F_{OSC}/4$  or  $F_{OSC}/1024$ ) can be selected
  - Timer 1: with auto-reload function and one of three internal clock frequencies ( $F_{OSC}$ ,  $F_{OSC}/64$  or  $F_s$ ) can be selected by MR1 register; and the specified frequency can be delivered to MFP pin
- Built-in 18/15-bit watchdog timer selectable for system reset; enable the watch dog timer or not is determined by code option
- Powerful instruction set: 142 instructions
- 16-levels subroutine (include interrupt) nesting

# Preliminary W742E81A/W742C81A



## 3. PIN CONFIGURATION

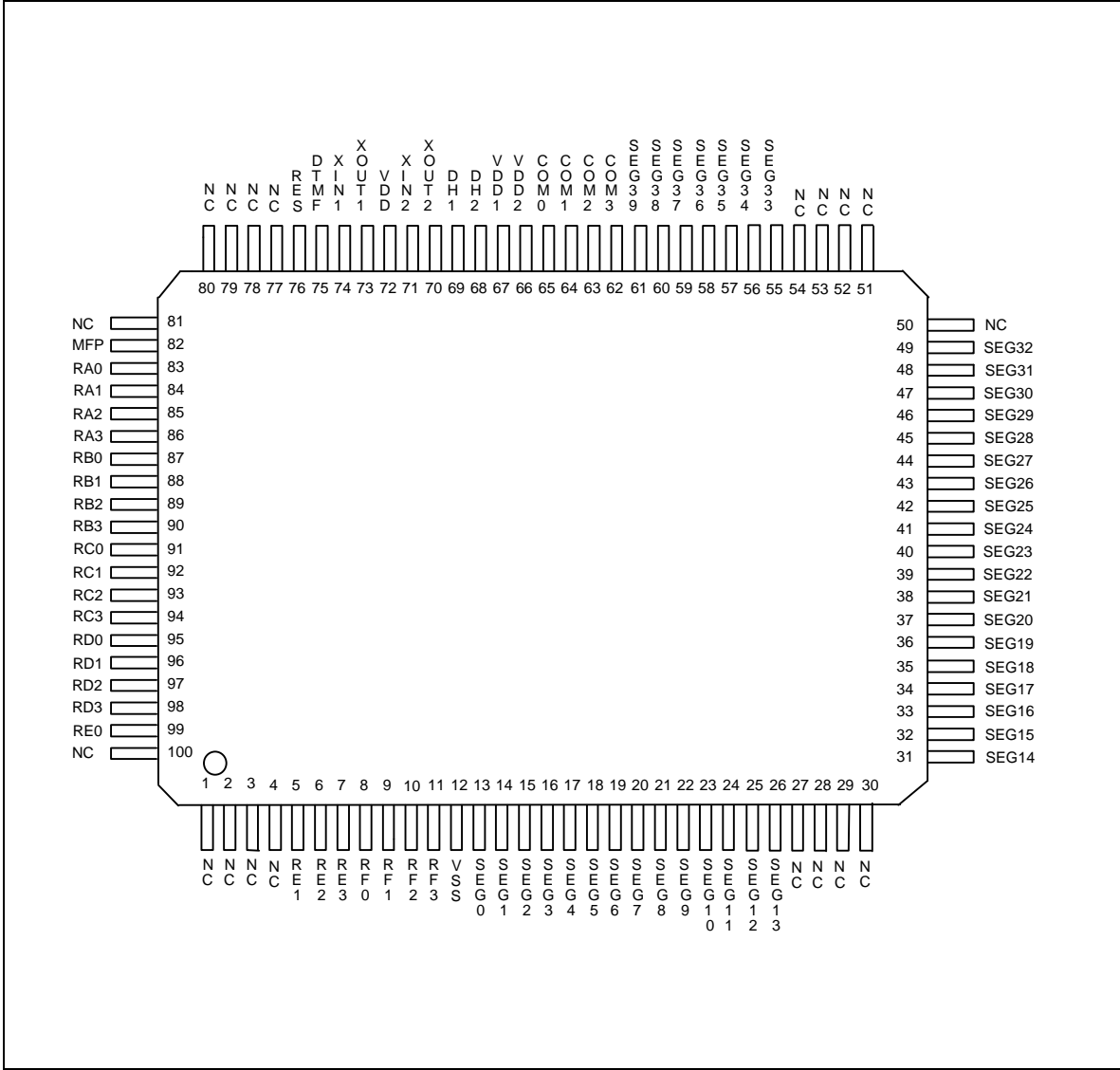
For W742E81A



# Preliminary W742E81A/W742C81A



For W742C810



# Preliminary W742E81A/W742C81A



## 4. PIN DESCRIPTION

| SYMBOL     | I/O | FUNCTION   |
|------------|-----|--|
| XIN2       | I   | Input pin for sub-oscillator.<br>Connected to 32.768 KHz crystal only.   |
| XOUT2      | O   | Output pin for sub-oscillator with internal oscillation capacitor. Connected to 32.768 KHz crystal only.   |
| XIN1       | I   | Input pin for main-oscillator.<br>Connected to 3.58 MHz or 400 KHz crystal or RC to generate system clock.   |
| XOUT1      | O   | Output pin for main-oscillator.<br>Connected to 3.58 MHz or 400 KHz crystal or RC to generate system clock.  |
| RA0–RA3    | I/O | Input/Output port.<br>Input/output mode specified by port mode 1 register (PM1).   |
| RB0–RB3    | I/O | Input/Output port.<br>Input/output mode specified by port mode 2 register (PM2).   |
| RC0–RC3    | I   | 4-bit port for input only.<br>Each pin has an independent interrupt capability.  |
| RD0–RD3    | I/O | Input/Output port.<br>Input/output mode specified by port mode 5 register (PM5).   |
| RE0–RE3    | O   | Output port only. With high sink current capacity for the LED application.   |
| RF0–RF3    | O   | Output port only.  |
| MFP        | O   | Output pin only.<br>This pin can output modulating or nonmodulating frequency, or Timer 1 specified frequency. It can be selected by bit 0 of BUZCR (BUZCR.0). |
| DTMF       | O   | This pin can output dual-tone multifrequency signal for dialling.  |
| RES        | I   | System reset pin with pull-high resistor.  |
| SEG0–SEG39 | O   | LCD segment output pins.   |
| COM0–COM3  | O   | LCD common signal output pins.<br>The LCD alternating frequency can be selected by code option.  |
| DH1, DH2   | I   | Connection terminals for voltage doubler (halver) capacitor.   |
| VDD1       | I   | Positive (+) supply voltage terminal.  |
| VDD2       |     | Refer to Functional Description.   |
| VDD        | I   | Positive power supply (+).   |
| VSS        | I   | Negative power supply (-).   |

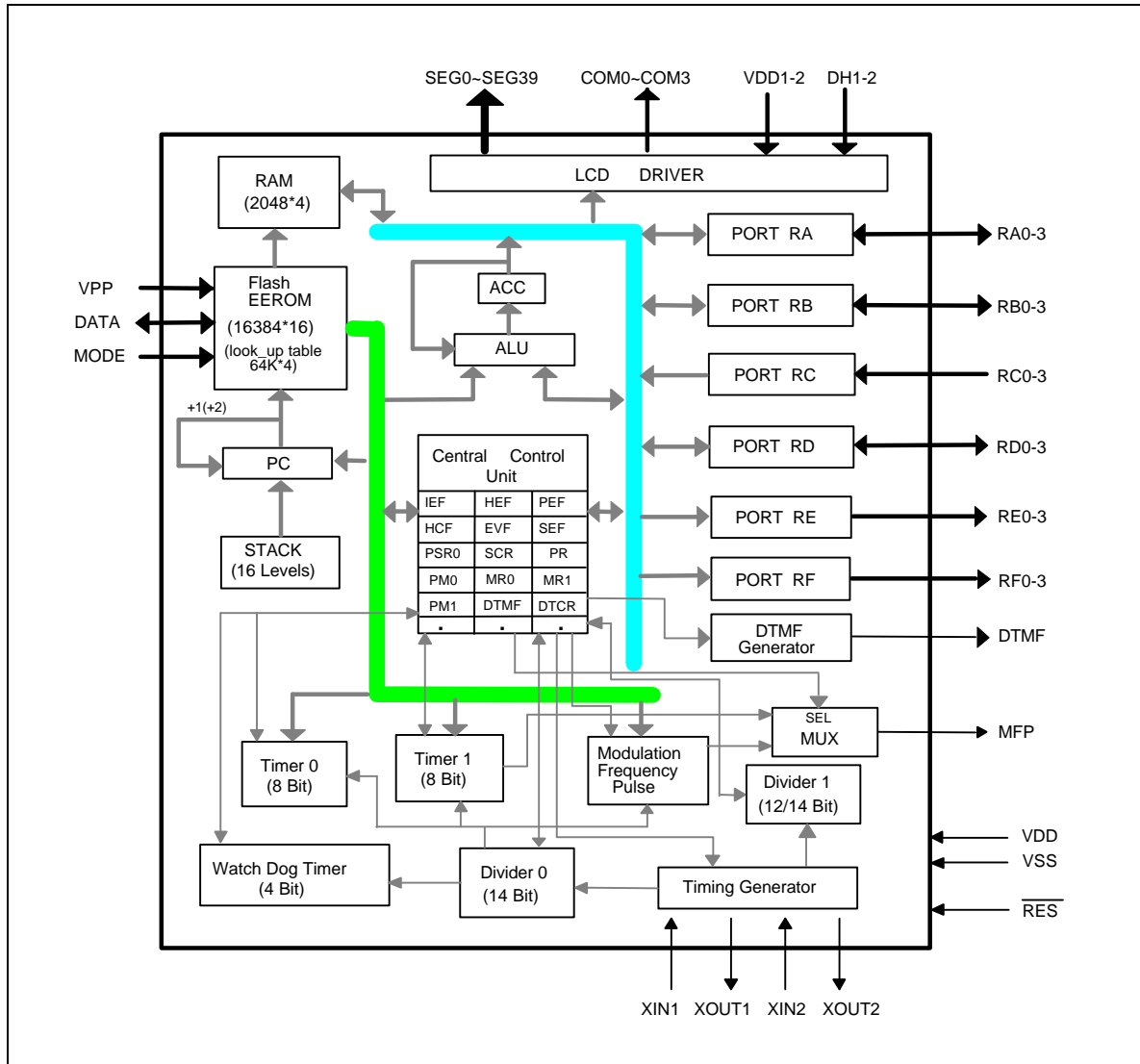
# Preliminary W742E81A/W742C81A



Pin Description, continued

| SYMBOL | I/O | FUNCTION   |
|--------|-----|--|
| VPP    | I   | Voltage control pin for the flash EEPROM programming, erasing and verifying. This pin has the built-in pull-low resistor.  |
| MODE   | I   | This pin can be used as mode selection control; data read/write clock; program/erase control or address counter control in the flash EEPROM erasing, programming or verifying mode. This pin has the built-in pull-low resistor. |
| DATA   | I/O | Data I/O pin with the built-in pull-low resistor.  |

## 5. BLOCK DIAGRAM



# Preliminary W742E81A/W742C81A



## 6. FUNCTIONAL DESCRIPTION

### 6.1 Program Counter (PC)

Organized as an 14-bit binary counter (PC0 to PC13), the program counter generates the addresses of the 16384 × 16 on-chip ROM containing the program instruction words. Before the jump or subroutine call instructions are to be executed, the destination ROM page must be determined firstly. The confirmation of the ROM page can be done by executing the MOV ROMPR, #I or MOV ROMPR, R instruction. When the interrupt or initial reset conditions are to be executed, the corresponding address will be loaded into the program counter directly. The format used is shown below.

Table 1 Vector address and interrupt priority

| ITEM             | ADDRESS | INTERRUPT PRIORITY |
|------------------|---------|--------------------|
| Initial Reset    | 0000H   | -                  |
| INT 0 (Divider0) | 0004H   | 1st                |
| INT 1 (Timer 0)  | 0008H   | 2nd                |
| INT 2 (Port RC)  | 000CH   | 3rd                |
| INT 3 (Divider1) | 0014H   | 4th                |
| INT 4 (Timer 1)  | 0020H   | 5th                |
| JP Instruction   | XXXXH   | -                  |
| Subroutine Call  | XXXXH   | -                  |

### 6.2 Stack Register (STACK)

The stack register is organized as 49 bits x 16 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. (Refer to Table 8) When the stack register is pushed over the sixteen levels, the contents of the first level will be lost. In other words, the stack register is always sixteen levels deep.

### 6.3 Program Memory (ROM)

The read-only memory (ROM) is used to store program codes; and the look-up table is arranged as 65536 x 4 bits. The program ROM is divided into eight pages; the size of each page is 2048 x 16 bits. So the total ROM size is 16384 x 16 bits. Before the jump or subroutine call instructions are to be executed, the destination ROM page must be determined firstly. The ROM page can be selected by executing the MOV ROMPR,#I or MOV ROMPR, R instruction. But the branch decision instructions (e.g. JB0, SKB0, JZ, JC, ...) must jump to the same ROM page which the branch decision instructions are in. The whole ROM 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 65536 elements. Instruction MOVC R is used to read the look-up table content and transfer table data to the RAM. But before reading the addressed look-up table content, the content of the look-up table pointer (TAB) must be determined firstly. The address of the look-up table element is allocated by the content of TAB. The MOV TAB0 (TAB1, TAB2, TAB3), R instructions are used to allocate the address of the wanted look-up table element. The TAB0 register stores the LSB 4 bits of the look-up table address.

# Preliminary W742E81A/W742C81A



The organization of the program memory is shown in Figure 6-1.

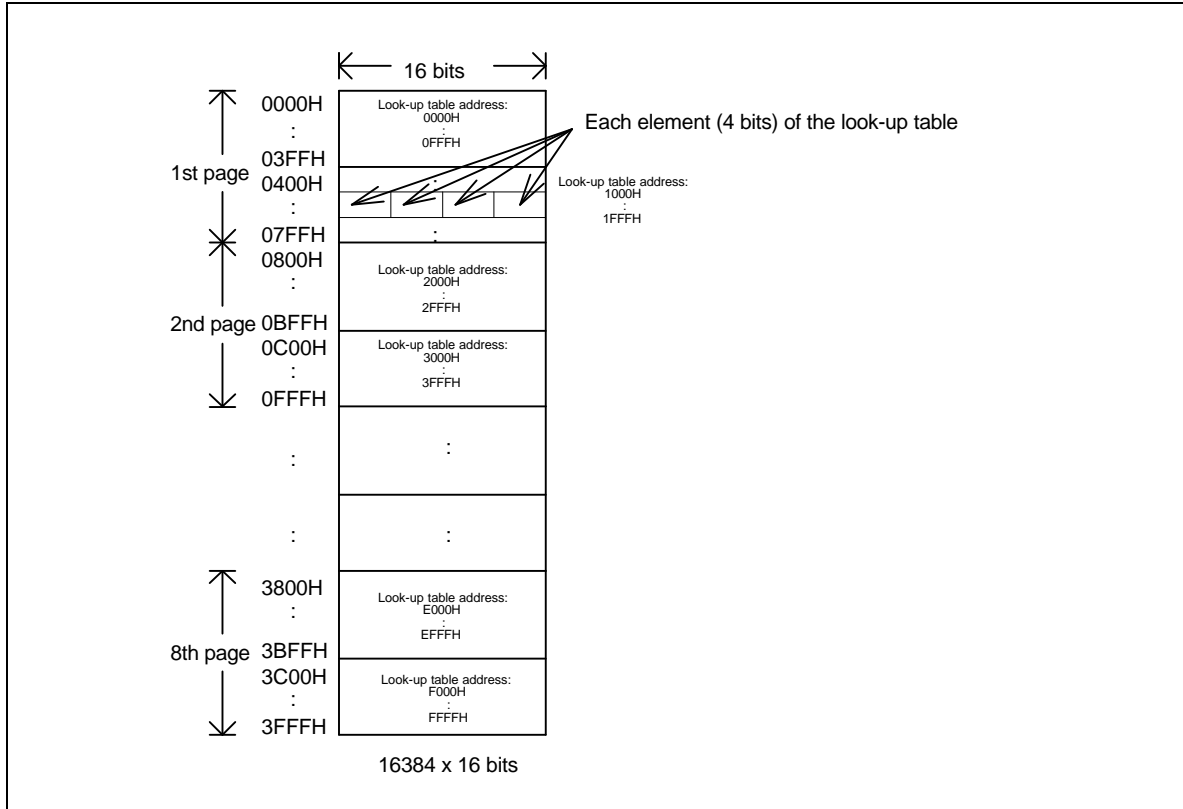


Figure 6-1 Program Memory Organization

## 6.3.1 ROM Page Register (ROMPR)

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

|       |   |   |   |   |
|-------|---|---|---|---|
|       | 3 | 2 | 1 | 0 |
| ROMPR | — | W | W | W |

Note: W means write only.

Bit 3 is reserved.

Bit 2, Bit 1, Bit 0 ROM page preselect bits:

- 000 = ROM page 0 (0000H - 07FFH)
- 001 = ROM page 1 (0800H - 0FFFH)
- 010 = ROM page 2 (1000H - 17FFH)
- 011 = ROM page 3 (1800H - 1FFFH)
- 100 = ROM page 4 (2000H - 27FFH)
- 101 = ROM page 5 (2800H - 2FFFH)
- 110 = ROM page 6 (3000H - 37FFH)
- 111 = ROM page 7 (3800H - 3FFFH)



# Preliminary W742E81A/W742C81A



## 6.4 Data Memory (RAM)

### 6.4.1 Architecture

The static data memory (RAM) used to store data is arranged as  $2048 \times 4$  bits. The data RAM is divided into sixteen banks; each bank has  $128 \times 4$  bits. Executing the MOV DBKR,WR or MOV DBKR,#I instruction can determine which data bank is used. The data memory can be addressed directly or indirectly. But the data bank must be confirmed firstly; and the page in the data bank will be done in the indirect addressing mode, too. In indirect addressing mode, each data bank will be divided into eight pages. Before the data memory is addressed indirectly, the page which the data memory is in must be confirmed. The organization of the data memory is shown in Figure 6-2.

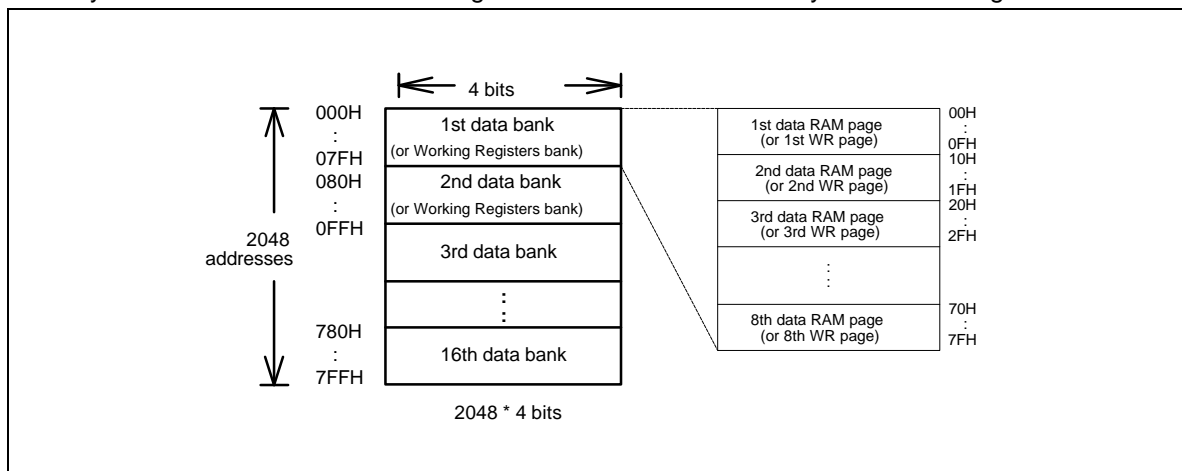
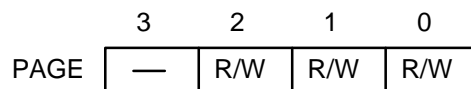


Figure 6-2 Data Memory Organization

The 1st and 2nd data bank (00H to 7FH & 80H to FFH) in the data memory can also be used as the working registers (WR). It is also divided into sixteen pages. Each page contains 16 working registers. When one page is used as WR, the others can be used as the normal data memory. The WR page can be switched by executing the MOV WRP,R or MOV WRP,#I instruction. The data memory cannot operate directly with immediate data, but the WR can do. The relationship between data memory locations and the page register (PAGE) in indirect addressing mode is described in the next subsection.

### 6.4.2 Page Register (PAGE)

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



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)

# Preliminary W742E81A/W742C81A



- 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)

### 6.4.3 WR Page Register (WRP)

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

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
|     | 3   | 2   | 1   | 0   |
| WRP | R/W | R/W | R/W | R/W |

Note: R/W means read/write available.

Bit 3, Bit 2, Bit 1, Bit 0 Working registers page preselect bits:

- 0000 = WR Page 0 (00H - 0FH)
- 0001 = WR Page 1 (10H - 1FH)
- 0010 = WR Page 2 (20H - 2FH)
- 0011 = WR Page 3 (30H - 3FH)
- 0100 = WR Page 4 (40H - 4FH)
- 0101 = WR Page 5 (50H - 5FH)
- 0110 = WR Page 6 (60H - 6FH)
- 0111 = WR Page 7 (70H - 7FH)
- 1000 = WR Page 8 (80H - 8FH)
- 1001 = WR Page 9 (90H - 9FH)
- 1010 = WR Page A (A0H - AFH)
- 1011 = WR Page B (B0H - BFH)
- 1100 = WR Page C (C0H - CFH)
- 1101 = WR Page D (D0H - DFH)
- 1110 = WR Page E (E0H - EFH)
- 1111 = WR Page F (F0H - FFH)

### 6.4.4 Data Bank Register (DBKR)

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

|      |     |     |     |     |
|------|-----|-----|-----|-----|
|      | 3   | 2   | 1   | 0   |
| DBKR | R/W | R/W | R/W | R/W |

Note: R/W means read/write available.

# Preliminary W742E81A/W742C81A



Bit 3, Bit 2, Bit 1, Bit 0 Data memory bank preselect bits:

- 0000 = Data bank 0 (000H - 07FH)
- 0001 = Data bank 1 (080H - 0FFH)
- 0010 = Data bank 2 (100H - 17FH)
- 0011 = Data bank 3 (180H - 1FFH)
- 0100 = Data bank 4 (200H - 27FH)
- 0101 = Data bank 5 (280H - 2FFH)
- 0110 = Data bank 6 (300H - 37FH)
- 0111 = Data bank 7 (380H - 3FFH)
- 1000 = Data bank 8 (400H - 47FH)
- 1001 = Data bank 9 (480H - 4FFH)
- 1010 = Data bank A (500H - 57FH)
- 1011 = Data bank B (580H - 5FFH)
- 1100 = Data bank C (600H - 67FH)
- 1101 = Data bank D (680H - 6FFH)
- 1110 = Data bank E (700H - 77FH)
- 1111 = Data bank F (780H - 7FFH)

## 6.5 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.

## 6.6 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. CF can be read out by executing MOV R, CF.

## 6.7 Main-Oscillator

The W742E81A/W742C81A provides a crystal or RC oscillation circuit to generate the system clock through external connections. If a crystal oscillator is used, The 3.58 MHz or 400KHz crystal must be connected to XIN1 and XOUT1, and a capacitor must be connected to XIN1 and VSS if an accurate frequency is needed.

# Preliminary W742E81A/W742C81A

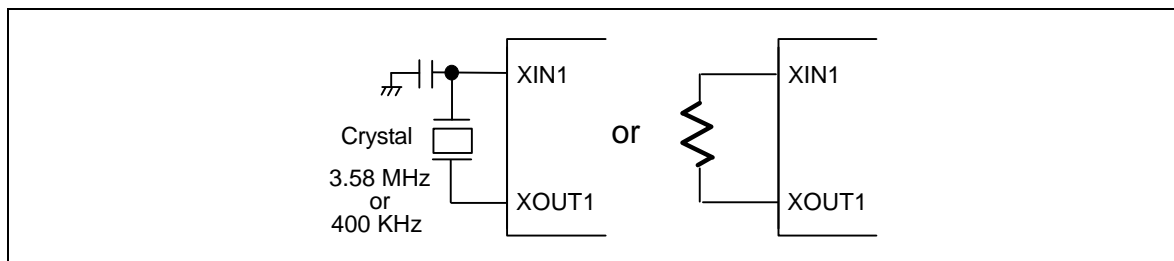


Figure 6-3 System Clock Oscillator Configuration

## 6.8 Sub-Oscillator

The sub-oscillator is used in dual-clock operation mode. In the sub-oscillator application, just only the 32768 Hz crystal could be connected to XIN2 and XOUT2, and it would not be oscillated in STOP mode.

## 6.9 Dividers

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). Then, if the Divider0 interrupt enable flag has been set (IEF.0 = 1), the interrupt is executed, while if the hold release enable flag has been set (HEF.0 = 1), the hold state is terminated. And the last 4-stage of the Divider0 can be reset by executing CLR DIVR0 instruction.

If the sub-oscillator starts action, the Divider1 is incremented by each clock (Fs in dual-clock mode or Fosc/128 in single-clock mode). When an overflow occurs, the Divider1 event flag is set to 1 (EVF.4 = 1). Then, if the Divider1 interrupt enable flag has been set (IEF.4 = 1), the interrupt is executed, while if the hold release enable flag has been set (HEF.4 = 1), the hold state is terminated. And the last 4-stage of the Divider1 can be reset by executing CLR DIVR1 instruction. Same as EVF.0, the EVF.4 is set to 1 periodically. But there are two period time (125 mS & 500mS) that can be selected by setting the SCR.3 bit. When SCR.3 = 0 (default), the 500 mS period time is selected; SCR.3 = 1, the 125 mS period time is selected.

## 6.10 Dual-clock operation

This operation mode is selected by option code. 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 signal-clock mode, the clock source of the LCD frequency selector will be  $F_m/128$  ( $F_m$  : main oscillator clock, See figure 6-4). **So before the STOP instruction is executing, the LCD must be turned off in the signal-clock mode or dual-clock mode.**

In this dual-clock mode, the normal operation is performed by generating the system clock from the main-oscillator clock ( $F_m$ ). As required, the slow operation can be performed by generating the system clock from the sub-oscillator clock ( $F_s$ ). 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 main-oscillator clock; if the SCR.0 is set to 1, the clock source of the system clock generator is 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.

# Preliminary W742E81A/W742C81A



When the SCR is set or reset, we must care the following cases:

1. X000B → X011B: we should not exchange the FOSC from Fm into Fs and disable Fm simultaneously. We could first exchange the FOSC from Fm into Fs, then disable the main-oscillator. So it should be X000B→X001B→X011B.
2. X011B → X000B: we should not enable Fm and exchange the FOSC from Fs into Fm simultaneously. We could first enable the main-oscillator; the 2nd step is calling a delay subroutine to wait the main-oscillator oscillating stably; then exchange the FOSC from Fs into Fm is the last step. So it should be X011B→X001B→delay the Fm oscillating stably time→X000B. The suggestion of the Fm oscillating stably time is 3.5 mS for 455 KHz and 0.8ms for 4 MHz.

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

The organization of the dual-clock operation mode is shown in Figure 6-4.

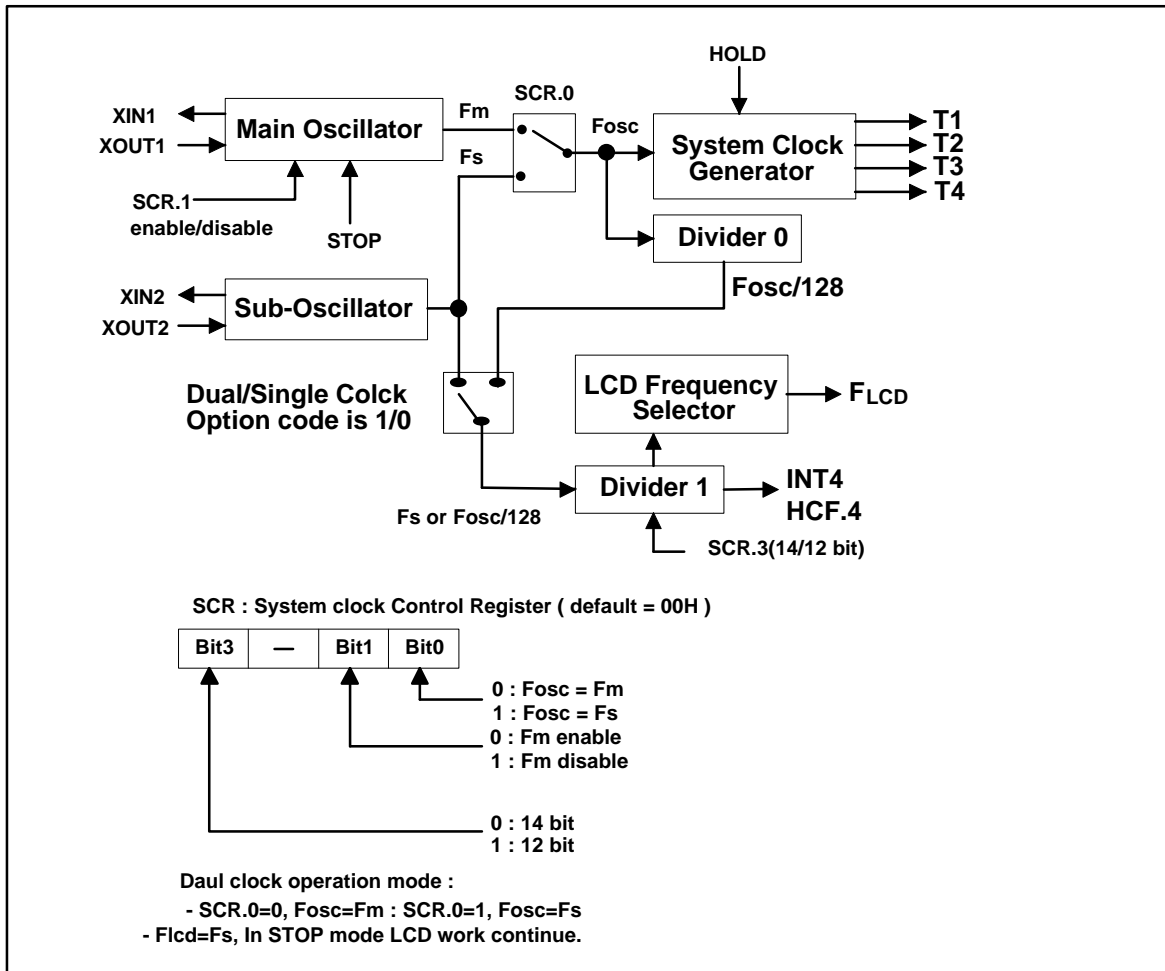


Figure 6-4 Organization of the dual-clock operation mode

# Preliminary W742E81A/W742C81A



## 6.11 WatchDog Timer (WDT) and WatchDog Timer Register(WDTR)

The watchdog timer (WDT) is organized as a 4-bit up counter designed to prevent the program from unknown errors. When the corresponding option code bit of the WDT set to 1, the WDT is enabled, and if the WDT overflows, the chip will be reset. At initial reset, the input clock of the WDT is  $F_{osc}/2048$ . The input clock of the WDT can be switched to  $F_{osc}/16384$  (or  $F_{osc}/2048$ ) by setting  $WDTR.3$  to 1. 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 operation is not under control and the chip will be reset. The WDT overflow period is 1 S when the sub-system clock ( $F_s$ ) is 32 KHz and WDT clock input is  $F_s/2048$ . When the corresponding option code bit of the WDT set to 0, the WDT function is disabled. The organization of the Divider0 and watchdog timer is shown in Figure 6-5.

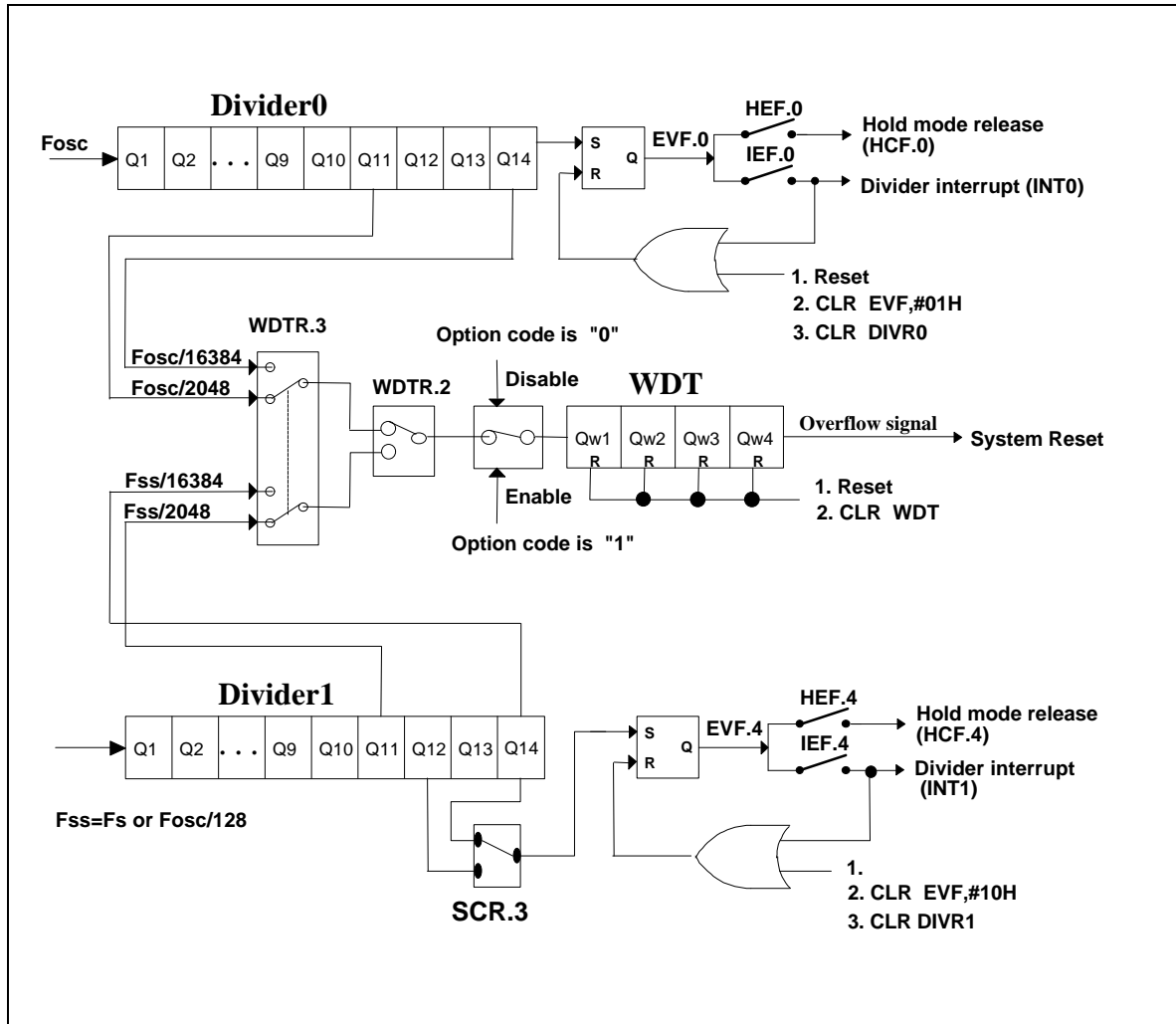


Figure 6-5 Organization of Divider0, Divider1 and WatchDog Timer

# Preliminary W742E81A/W742C81A



|      |     |     |     |   |
|------|-----|-----|-----|---|
|      | 3   | 2   | 1   | 0 |
| WDTR | R/W | R/W | R/W | R |

Note: R/W means read/write available, R means read only.

Power On reset default is : 0000

- Bit 3 = 0 FOSC/2048(Select Divider0) or Fss/2048(Select Divider1) as the WDT source.
- = 1 FOSC/16384(Select Divider0) or Fss/16384(Select Divider1) as the WDT source.
- Bit 2 = 0 Select Divider0.
- = 1 Select Divider1.
- Bit 1 = 0 Refer to Table 2.
- = 1 Refer to Table 2.
- Bit 0 = 0 No time out.
- = 1 Time out.

WDTR.0 will be set to one when WDT time out and can be reset to zero by:

Power On Reset, RESET pin, CLR WDT

Table 2 The bit 1 of WatchDog Timer Register (WDTR) reset item

| RESET ITEM   | WDTR.1 = 1  | WDTR.1 = 0       |
|--|-------------|------------------|
| Program Counter (PC)                               | 0000H       | 0000H            |
| Stack Pointer (SP)                                 | -           | Reset            |
| ROMPR, PAGE, DBKR, WRP, ACC, CF, ZF, SCR registers | -           | Reset            |
| IEF, HEF, SEF, HCF, PEF, EVF flags                 | IEF = Reset | Reset            |
| DIV0, DIV1   | -           | Reset            |
| TM0, TM1, MR0, MR1 registers                       | -           | Reset            |
| Timer 0 input clock                                | -           | FOSC/4           |
| Timer 1 input clock                                | -           | FOSC             |
| MFP output   | -           | Low              |
| PM0 register                                       | -           | Reset            |
| PM1, PM2, PM5 registers                            | -           | Set (1111B)      |
| PSR0 register                                      | -           | Reset            |
| Input/output ports RA, RB, RD                      | -           | Input mode       |
| Output ports RE, RF                                | -           | High             |
| RA, RB ports output type                           | -           | CMOS type        |
| RC port pull-high resistors                        | -           | Disable          |
| Input clock of the watchdog timer                  | -           | FOSC/2048        |
| DTMF output  | -           | Hi-Z             |
| BUZCR register                                     | -           | Reset            |
| FLCD   | -           | Q5 to Q9 Reset   |
| LCD display  | -           | OFF              |
| LCDR   | -           | Reset            |
| Segment output mode                                | -           | LCD drive output |

-: keep the status

Note: SCR.2 is reserved

# Preliminary W742E81A/W742C81A



## 6.12 Timer/Counter

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

If the Timer 0 clock input is FOSC/4:

$$\text{Desired Timer 0 interval} = (\text{preset value} + 1) \times 4 \times 1/\text{Fosc}$$

If the Timer 0 clock input is FOSC/1024:

$$\text{Desired Timer 0 interval} = (\text{preset value} + 1) \times 1024 \times 1/\text{Fosc}$$

Preset value: Decimal number of Timer 0 preset value

FOSC: Clock oscillation frequency

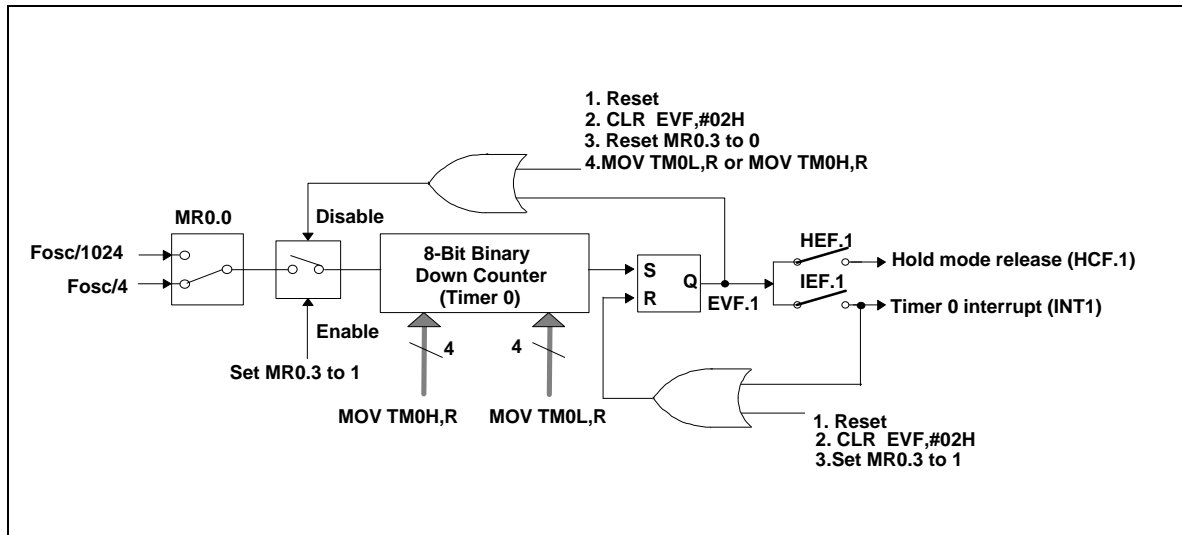


Figure 6-6 Organization of Timer 0



# Preliminary W742E81A/W742C81A



## 6.12.2 Timer 1 (TM1)

Timer 1 (TM1) is also a programmable 8-bit binary down counter, as shown in Figure 6-7. Timer 1 can be used as 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 FS. 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. When the MOV TM1L, R or MOV TM1H, R instruction is executed, the specified data are loaded into the auto-reload buffer; but the TM1 down-counting will keep going on. If the bit 3 of MR1 is set (MR1.3 = 1), the content of the auto-reload buffer will be loaded into the TM1 down counter, and Timer 1 starts to down count, and the event flag 7 is reset (EVF.7 = 0). When the timer decreases and underflow 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. Then, if interrupt enable flag 7 has been set to 1 (IEF.7 = 1), an interrupt is executed; if hold mode release enable flag 7 is set to 1 (HEF.7 = 1), the hold state is terminated. 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.

In a case where Timer 1 clock input is FT:

$$\text{Desired Timer 1 interval} = (\text{preset value} + 1) / FT$$

$$\text{Desired frequency for MFP output pin} = FT \div (\text{preset value} + 1) \div 2 \text{ (Hz)}$$

Preset value: Decimal number of Timer 1 preset value

FOSC: Clock oscillation frequency

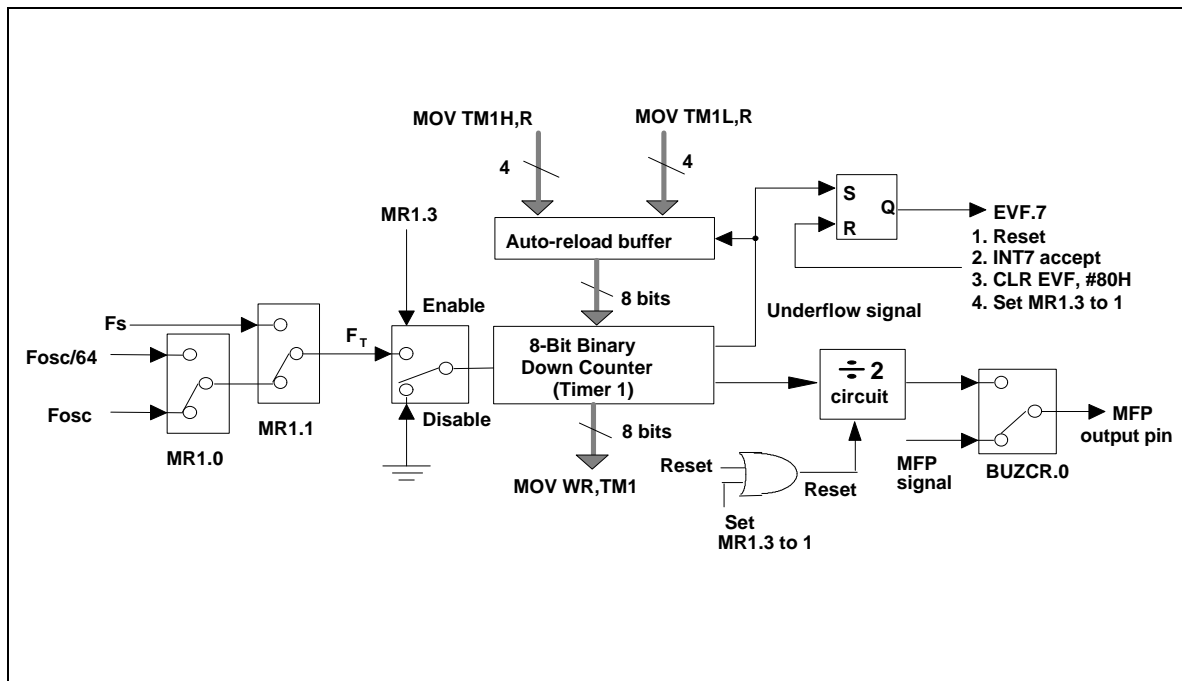


Figure 6-7 Organization of Timer 1

# Preliminary W742E81A/W742C81A



For example, when FT 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.

MOV WR, TM1 can read back the content of TM1, It will save the TM1 MSB to WR and the TM1 LSB to ACC.

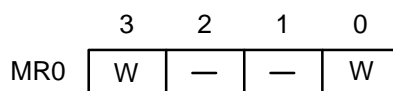
Table 3 The relation between the tone frequency and the present value of TM1

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

Note: Central tone is A4 (440 Hz).

## 6.12.3 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:



Note: W means write only.

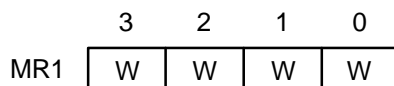
- Bit 0 = 0 The fundamental frequency of Timer 0 is FOSC/4.
- = 1 The fundamental frequency of Timer 0 is FOSC/1024.
- Bit 1 & Bit 2 are reserved
- Bit 3 = 0 Timer 0 stops down-counting.
- = 1 Timer 0 starts down-counting.

# Preliminary W742E81A/W742C81A



## 6.12.4 Mode Register 1 (MR1) & MFP Control Pin (BUZCR)

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:



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.

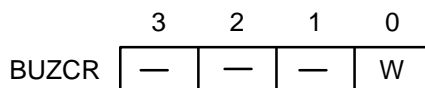
= 1 The fundamental frequency source of Timer 1 is the sub-oscillator frequency Fs (32768 Hz).

Bit 2 is reserved.

Bit 3 = 0 Timer 1 stops down-counting.

= 1 Timer 1 starts down-counting.

MFP control pin is organized as a 4-bit binary register.



Note: W means write only.

Bit 0 = 0 The specified waveform of the MFP generator is delivered to the MFP output pin.

= 1 The specified frequency of Timer 1 is delivered to the MFP output pin.

Bit 1, Bit 2 & Bit 3 are reserved.

## 6.13 Interrupts

The W742E81A/W742C81A provides four internal interrupt sources (Divider 0, Divider 1, Timer 0, Timer 1) and one external interrupt source (port RC). 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  $\mu$ C will enter hold mode again. The operation flow chart is shown in Figure 6-9. The control diagram is shown in Figure 6-9.

# Preliminary W742E81A/W742C81A

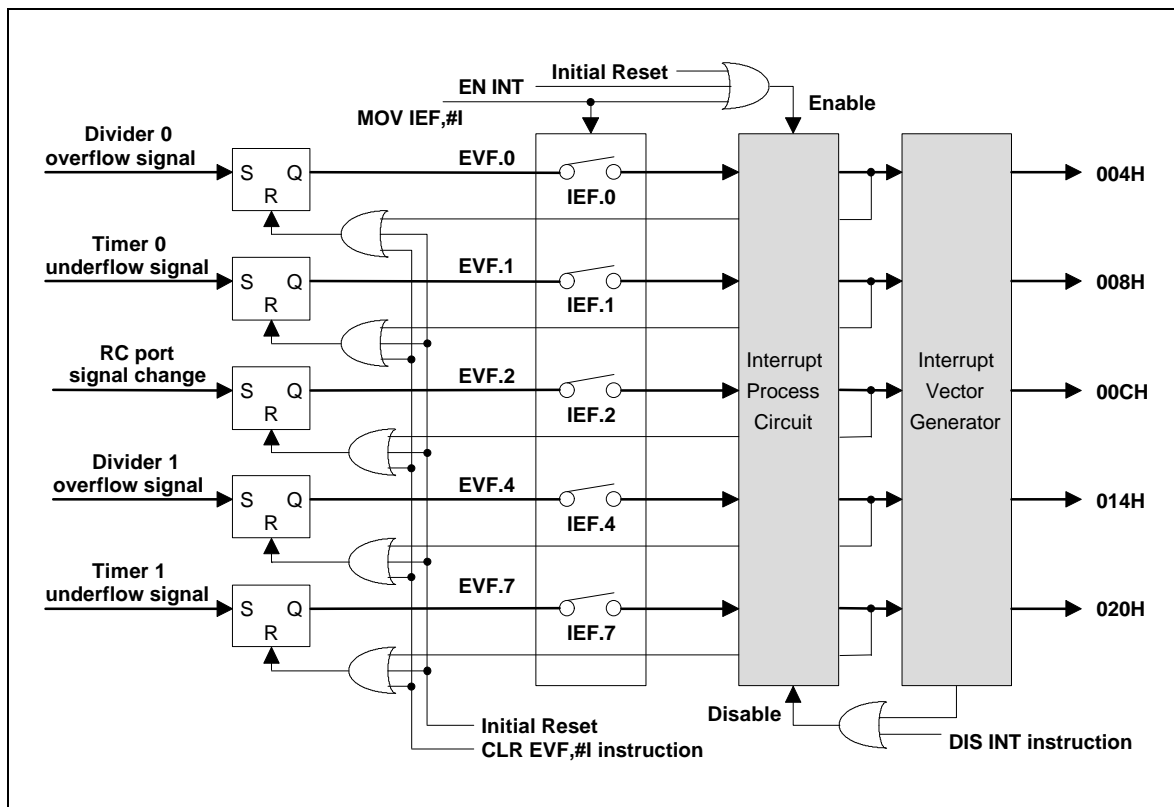


Figure 6-8 Interrupt event control diagram

## 6.14 Stop Mode Operation

In stop mode, all operations of the  $\mu\text{C}$  cease, and the MFP pin is kept to high. The  $\mu\text{C}$  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). When the designated signal is accepted, the  $\mu\text{C}$  awakens and executes the next instruction. To prevent erroneous execution, the NOP instruction should follow the STOP command. But In the dual-clock slow operation mode, the STOP instruction will also disable the sub-oscillator oscillating; all operations of the  $\mu\text{C}$  cease.

### 6.14.1 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, #1 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

# Preliminary W742E81A/W742C81A



- 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

## 6.15 Hold Mode Operation

In hold mode, all operations of the  $\mu$ C cease, except for the operation of the oscillator, Timer, Divider, LCD driver, DTMF generator and MFP generator. The  $\mu$ C 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.

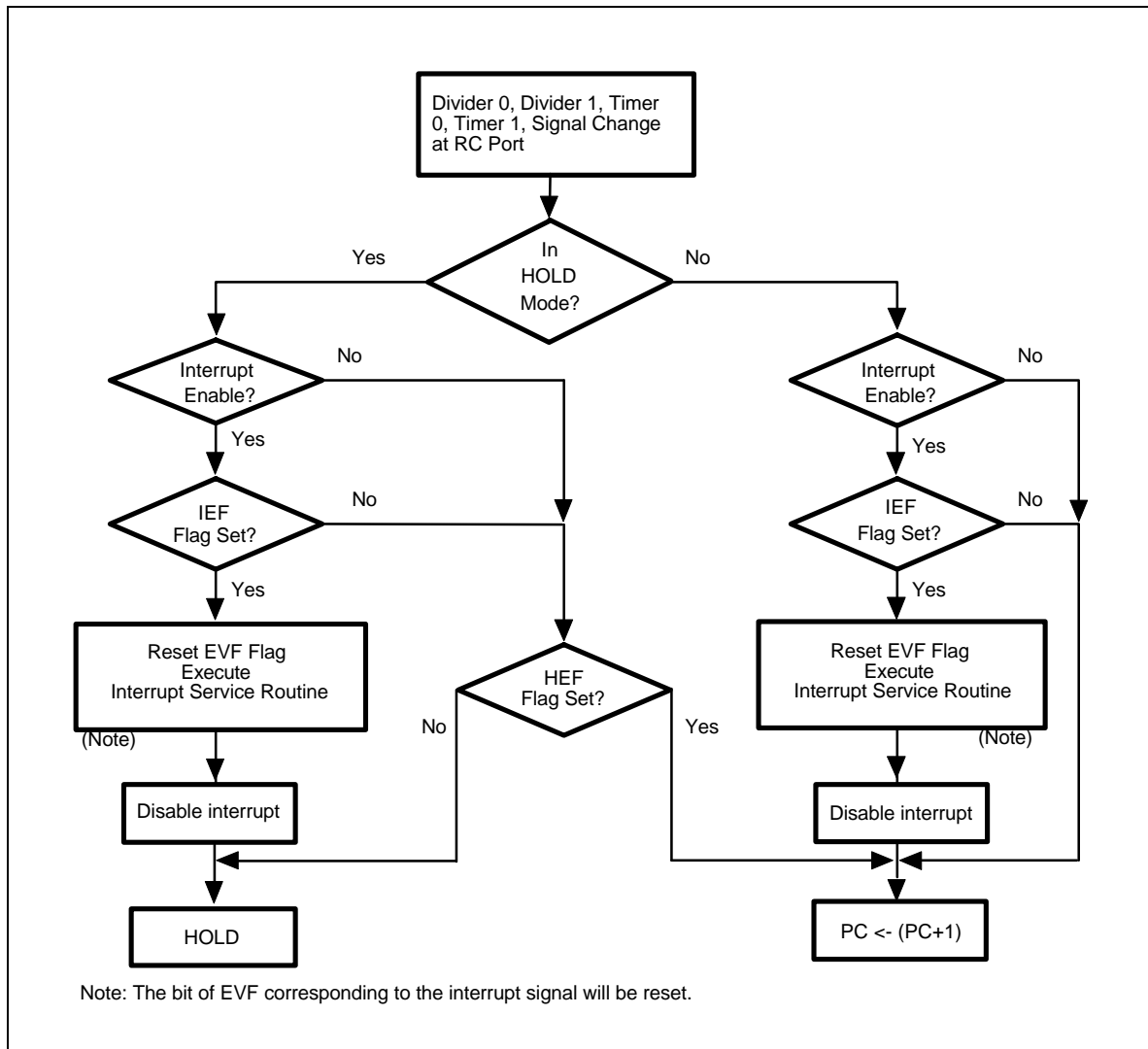


Figure 6-9 Hold Mode and Interrupt Operation Flow Chart

# Preliminary W742E81A/W742C81A



## 6.15.1 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, HEF.5 & HEF.6 are reserved.

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

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

## 6.15.2 Interrupt Enable Flag (IEF)

The interrupt enable flag is organized as a 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 disable till the instruction MOV IEF, #I or EN INT is executed again. Otherwise, these interrupts can be disable 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, IEF.5 & IEF.6 are reserved.

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

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

## 6.15.3 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 preform 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:

# Preliminary W742E81A/W742C81A



|     |   |   |   |   |
|-----|---|---|---|---|
|     | 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.

## 6.15.4 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.
- HCF.5 = 1 Hold mode was released by underflow from the timer 1.
- HCF.6 and HCF.7 are reserved.

## 6.15.5 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.

# Preliminary W742E81A/W742C81A



- 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.
- EVF.5 & EVF.6 are reserved.
- EVF.7 = 1 Underflow from Timer 1 occurred.

## 6.16 Reset Function

The W742E81A/W742C81A is reset either by a power-on reset or by using the external  $\overline{\text{RES}}$  pin. The initial state of the W742E81A/W742C81A after the reset function is executed is described below.

Table 4 The initial state after the reset function is executed

|                                    |            |
|------------------------------------|------------|
| Program Counter (PC)               | 000H       |
| WDTR registers                     | Reset      |
| BUZCR registers                    | Reset      |
| ACC, CF, ZF registers              | Reset      |
| MR0, MR1, PAGE registers           | Reset      |
| PSR0, SCR, TM0, TM1 registers      | Reset      |
| IEF, HEF, HCF, PEF, EVF, SEF flags | Reset      |
| WRP, DBKR, PAGE registers          | Reset      |
| Timer 0 input clock                | FOSC/4     |
| Timer 1 input clock                | FOSC       |
| MFP output                         | Low        |
| DTMF output                        | Hi-Z       |
| Input/output ports RA, RB, RD      | Input mode |
| Output port RE & RF                | High       |
| RA, RB ports output type           | CMOS type  |
| RC ports pull-high resistors       | Disable    |
| Input clock of the watchdog timer  | FOSC/2048  |
| LCD display                        | OFF        |



# Preliminary W742E81A/W742C81A



## 6.17 Input/Output Ports RA, RB & RD

Port RA consists of pins RA.0 to RA.3. Port RB consists of pins RB.0 to RB.3. Port RD consists of pins RD.0 to RD.3. At initial reset, input/output ports RA, RB and RD are all in input mode. When RA, RB are used as output ports, CMOS or NMOS open drain output type can be selected by the PM0 register. But when RD is used as output port, the output type is just fixed to be CMOS output type. Each pin of port RA, RB and RD can be specified as input or output mode independently by the PM1, PM2 and PM5 registers. The MOV A, R, RA or MOV A, R, RB or MOV A, R, RD instructions operate the input functions and the MOV RA, R or MOV RB, R or MOV RD, R operate the output functions. For more details, refer to the instruction table and Figure 6-10 and Figure 6-11.

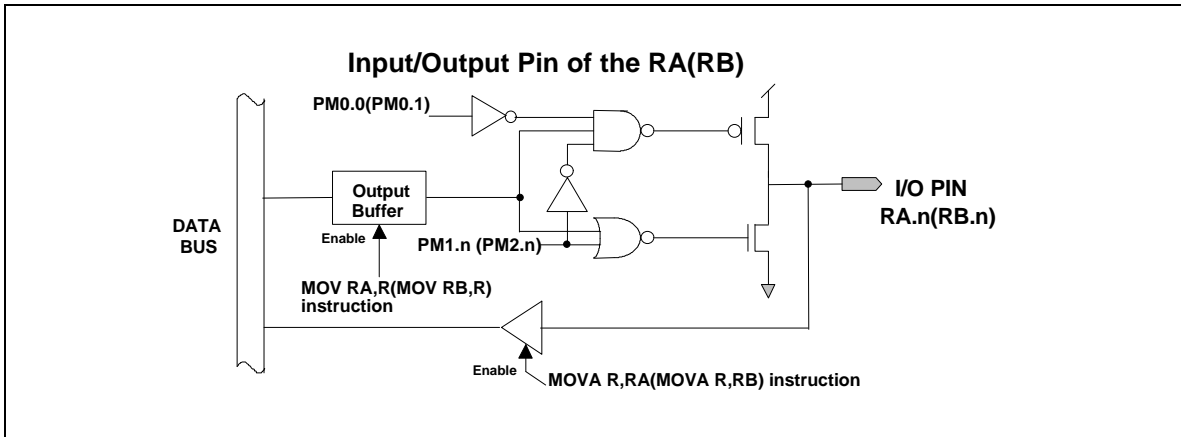


Figure 6-10 Architecture of RA (RB) Input/Output Pins

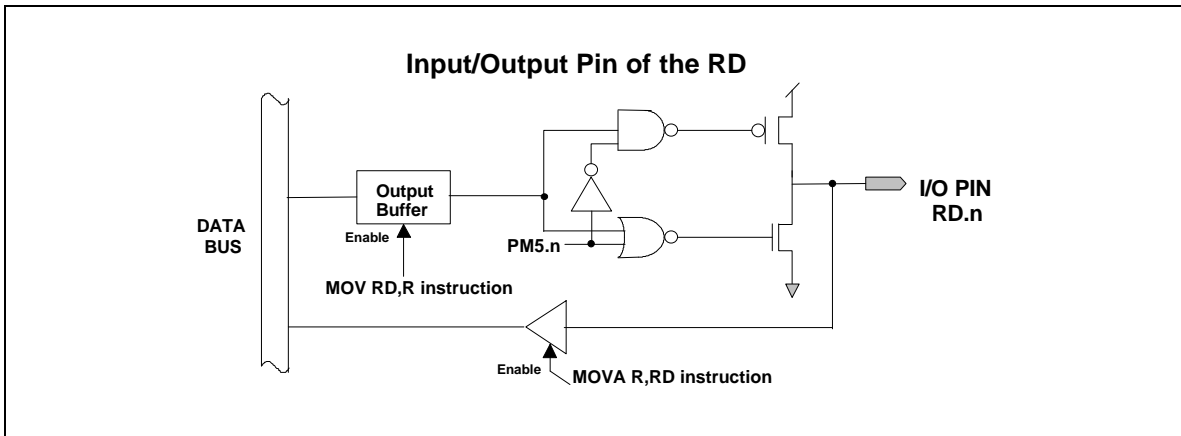


Figure 6-11 Architecture of RD Input/Output pins

### 6.17.1 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

# Preliminary W742E81A/W742C81A



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 is reserved.

## 6.17.2 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).

## 6.17.3 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).

# Preliminary W742E81A/W742C81A



## 6.17.4 Port Mode 5 Register (PM5)

The port mode 5 register is organized as 4-bit binary register (PM5.0 to PM5.3). PM5 can be used to control the input/output mode of port RD. PM5 is controlled by the MOV PM5, #I instruction. The bit descriptions are as follows:

|     |   |   |   |   |
|-----|---|---|---|---|
|     | 3 | 2 | 1 | 0 |
| PM5 | w | w | w | w |

Note: W means write only.

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

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

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

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

At initial reset, the port RD is input mode (PM5 = 1111B).

## 6.18 Input Ports RC

Port RC consists of pins RC.0 to RC.3. Each pin of port RC 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 at 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 ports. 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 6-12 and the instruction table for more details.

# Preliminary W742E81A/W742C81A

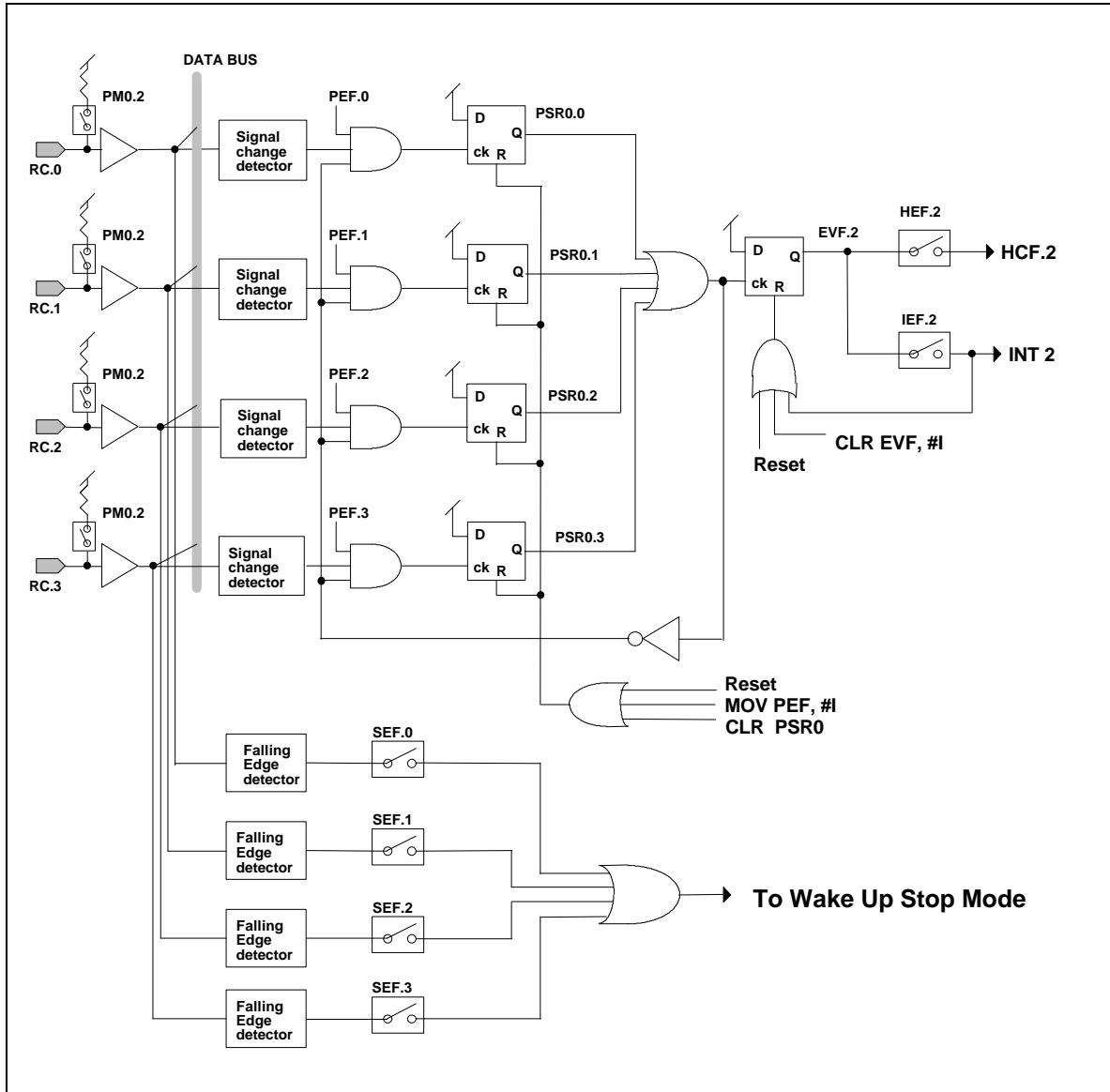


Figure 6-12 Architecture of Input Ports RC

## 6.18.1 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.

# Preliminary W742E81A/W742C81A



- 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

## 6.19 Output Port RE & RF

Output port RE is used as an output of the internal RT port. When the MOV RE, R instruction is executed, the data in the RAM will be output to port RT through port RE. It provides a high sink current to drive an LED. RF port is just used as a output port. When the MOV RF, R instruction is executed, the data in the RAM will be output to RF.

## 6.20 DTMF Output Pin (DTMF)

This pin should output the dual tone multi-frequency signal from the DTMF generator. There is the DTMF register that can specify the wanted low/high frequency. And control whether the dual tone will be output or not. The tones are divided into two groups (Row group and Col group) and one tone from each group is selected to represent a digit. The relation between the DTMF signal and the corresponding touch tone keypad is shown in Figure 6-13.

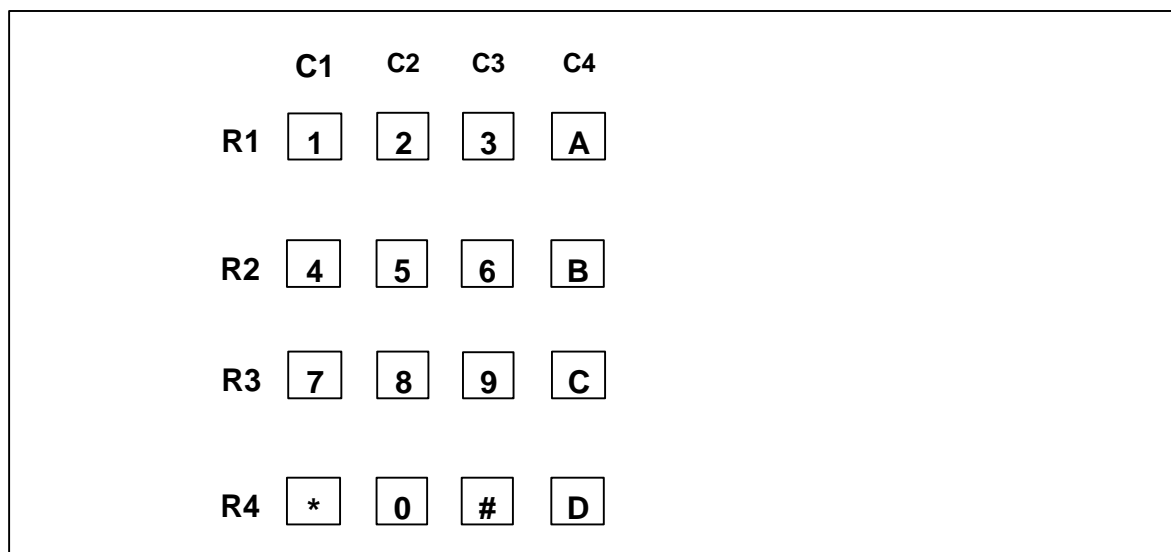
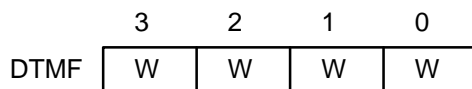


Figure 6-13 The relation between the touch tone keypad and the frequency

### 6.20.1 DTMF register

DTMF register is organized as 4-bit binary register. By controlling the DTMF register, one tone of the low/high group can be selected. The MOV DTMF,R instruction can specify the wanted tones. The bit descriptions are as follows:



Note: W means write only.

# Preliminary W742E81A/W742C81A



|                   | B3 | B2 | B1 | B0 | SELECTED TONE |
|-------------------|----|----|----|----|---------------|
| <b>High Group</b> | X  | X  | 0  | 0  | 1209 Hz       |
|                   | X  | X  | 0  | 1  | 1336 Hz       |
|                   | X  | X  | 1  | 0  | 1477 Hz       |
|                   | X  | X  | 1  | 1  | 1633 Hz       |
| <b>Low Group</b>  | 0  | 0  | X  | X  | 697 Hz        |
|                   | 0  | 1  | X  | X  | 770 Hz        |
|                   | 1  | 0  | X  | X  | 852 Hz        |
|                   | 1  | 1  | X  | X  | 941 Hz        |

Note: X means this bit do not care.

## 6.20.2 Dual Tone Control Register (DTCR)

Dual tone control register is organized as 4-bit binary register. The output of the dual or single tone will be controlled by this register. The MOV DTCR,#I instruction can specify the wanted status. The bit descriptions are as follows:

|      |   |   |   |   |
|------|---|---|---|---|
|      | 3 | 2 | 1 | 0 |
| DTCR | — | W | W | W |

Note: W means write only.

Bit 0 = 1 Low group tone output is enabled.

Bit 1 = 1 High group tone output is enabled.

Bit 2 = 1 DTMF output is enabled. When Bit 2 is reset to 0, the DTMF output pin will be Hi-Z state.

Bit 3 is reserved.

## 6.21 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 bit 0 of BUZCR (BUZCR.0). The organization of MR1 is shown in Figure 6-7. When bit 0 of BUZCR is reset to "0," the MFP output can deliver a modulation output in any combination of one signal from among DC, 4096 Hz, 2048 Hz, 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 in next page.

# Preliminary W742E81A/W742C81A



Table 5 The relation between the MFP output frequency and the data specified by 8-bit operand  
(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  | 0                |
| 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  | 0                |
| 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  | 0                |
| 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  | 0                |

# Preliminary W742E81A/W742C81A



## 6.22 LCD Controller/Driver

The W742E81A/W742C81A can directly drive an LCD with 40 segment output pins and 4 common output pins for a total of  $40 \times 4$  dots. The LCD driving mode is 1/3 bias 1/4 duty. The alternating frequency of the LCD can be set as  $F_w/64$ ,  $F_w/128$ ,  $F_w/256$ , or  $F_w/512$ . The structure of the LCD alternating frequency ( $F_{LCD}$ ) is shown in the Figure 6-14.

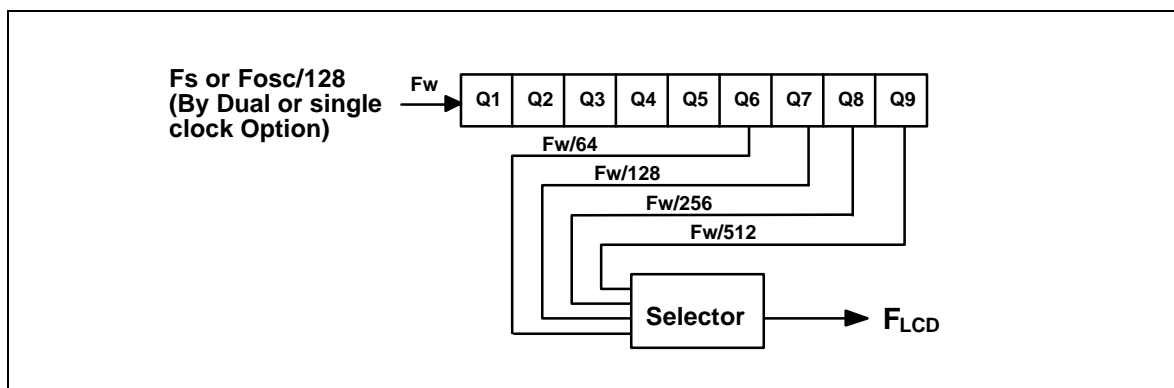


Figure 6-14 LCD alternating frequency ( $F_{LCD}$ ) circuit diagram

$F_w = 32.768$  KHz, the LCD frequency is as shown in the table below.

Table 6 The relationship between the  $F_{LCD}$  and the duty cycle

| LCD FREQUENCY | $F_w/64$ (512Hz) | $F_w/128$ (256Hz) | $F_w/256$ (128Hz) | $F_w/512$ (64Hz) |
|---------------|------------------|-------------------|-------------------|------------------|
| 1/4 DUTY      | 128 Hz           | 64 Hz             | 32 Hz             | 16 Hz            |

Corresponding to the 40 LCD drive output pins, there are 40 LCD data RAM segments. Instructions such as MOV LPL,R, MOV LPH,R, MOV @LP,R, and MOV R,@LP 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 segment39 pins by a direct memory access. The relation between the LCD data RAM and segment/common pins is shown below.

Table 7 The relation between the LCDR and segment/common pins used as LCD drive output pins

|              |            | 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   |
| :            | :          | :     | :     | :     | :     |
| :            | :          | :     | :     | :     | :     |
| LCDR26       | SEG38      | 0/1   | 0/1   | 0/1   | 0/1   |
| LCDR27       | SEG39      | 0/1   | 0/1   | 0/1   | 0/1   |



# Preliminary W742E81A/W742C81A



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 unlit. 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.

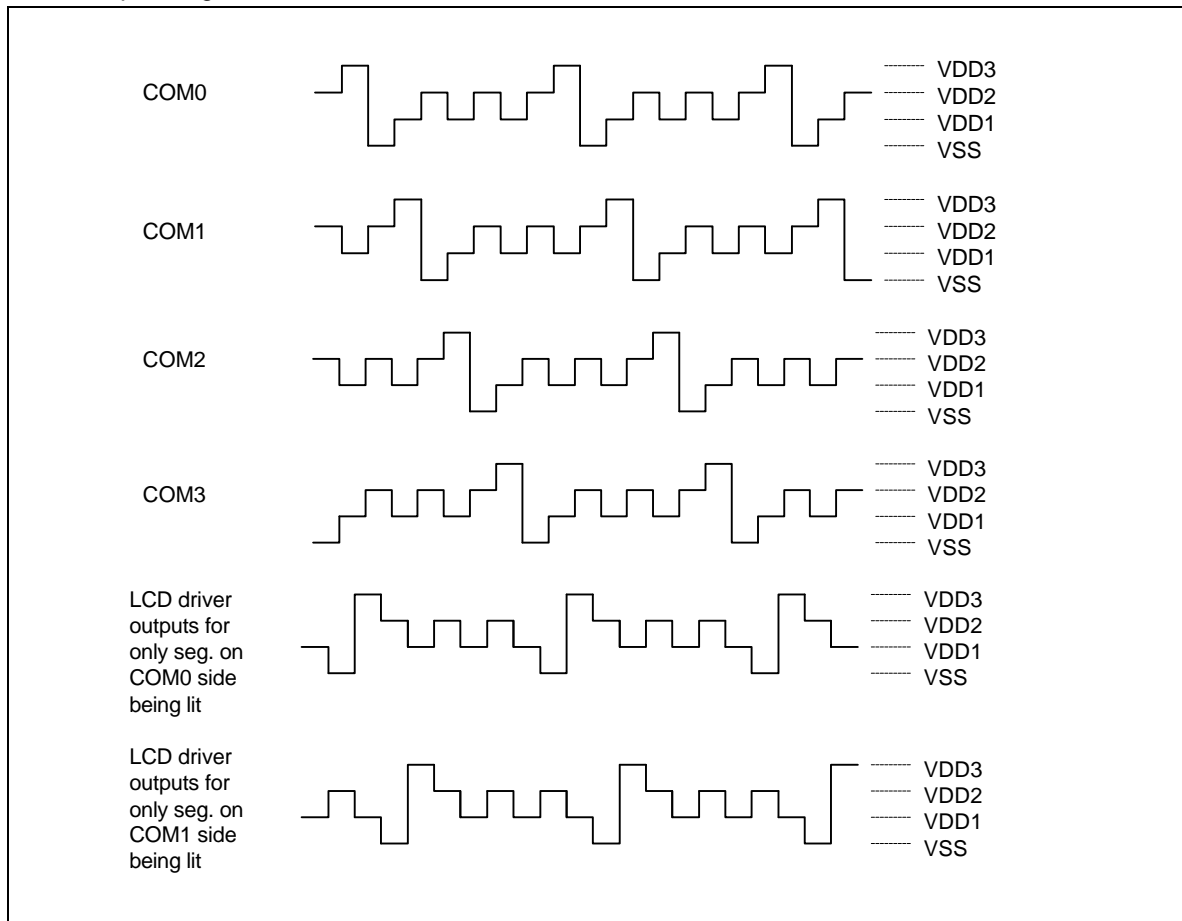
## 6.22.1 LCD RAM addressing method

There are 40 LCD RAMs (LCDR00 - Lcdr27) that should be indirectly addressed. The LCD RAM pointer (LP) is used to point to the address of the wanted LCD RAM. The LP is organized as 6-bit binary register. The MOV LPL,R and MOV LPH,R instructions can load the LCD RAM address to the LP from R. The MOV @LP,R and MOV R,@LP instructions can access the pointed LCD RAM content.

## 6.22.2 The output waveforms for the LCD driving mode

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

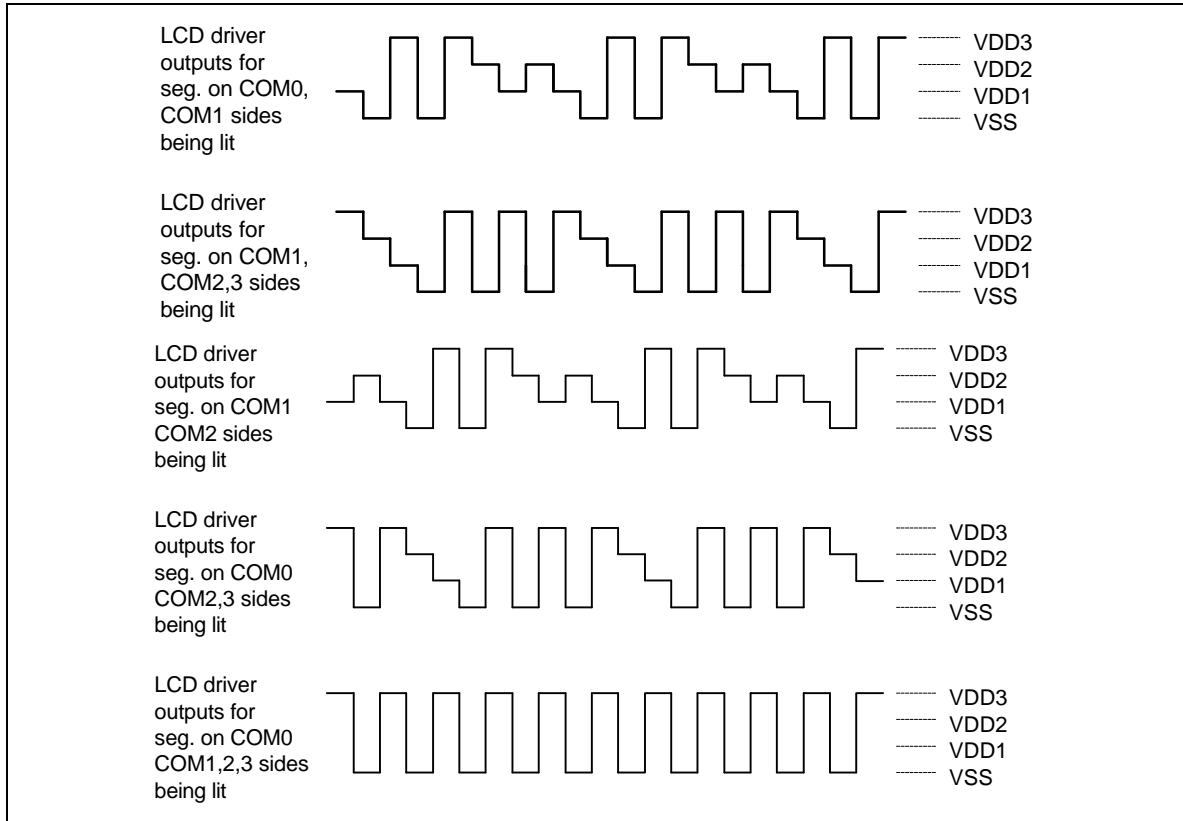
Normal Operating Mode



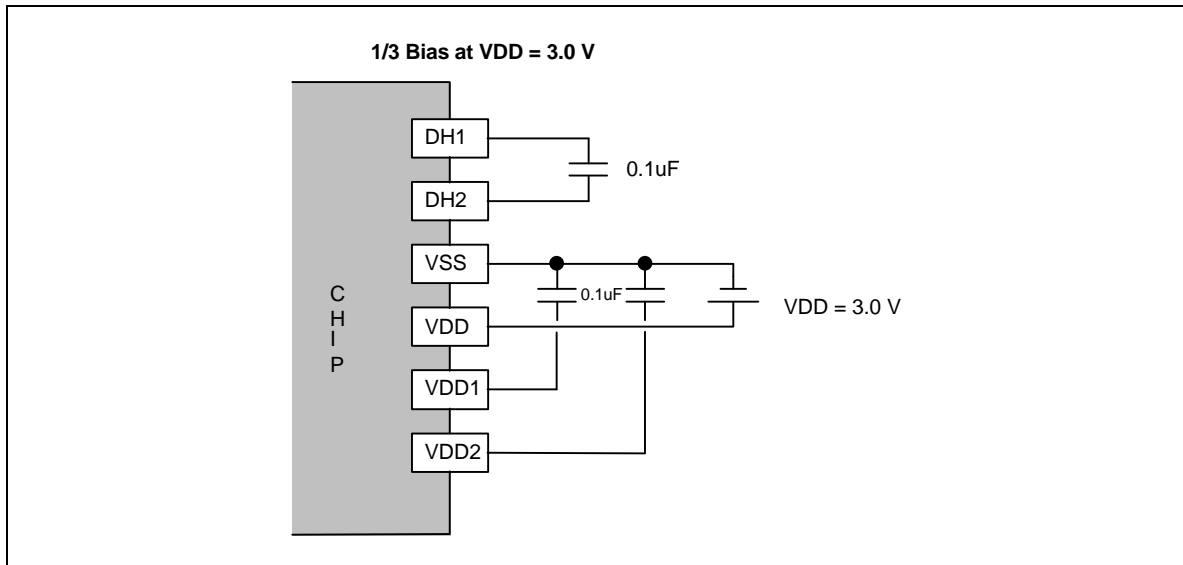
# Preliminary W742E81A/W742C81A



Continued



The power connections for the 1/3 bias 1/4 duty LCD driving mode are shown below.



# Preliminary W742E81A/W742C81A



## 6.23 Mode Description

The built-in program code memory of the W742E81A/W742C81A is the EEPROM structure. This memory can be programmed, erased and verified through the VPP, MODE and DATA pins.

## 7. 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.

## 8. DC CHARACTERISTICS

(V<sub>DD</sub>-V<sub>SS</sub> = 3.0 V, F<sub>m</sub> = 3.58 MHz, F<sub>s</sub> = 32.768 KHz, T<sub>A</sub> = 25° C, LCD on; unless otherwise specified)

| PARAMETER                   | SYM.            | CONDITIONS  | MIN. | TYP. | MAX. | UNIT |
|-----------------------------|-----------------|---|------|------|------|------|
| Op. Voltage                 | V <sub>DD</sub> | -   | 2.4  | -    | 3.8  | V    |
| Op. Current (Crystal type)  | IOP1            | No load (Ext-V)<br>In dual-clock normal operation                                       | -    | 0.9  | 2.5  | mA   |
| Op. Current (Crystal type)  | IOP3            | No load (Ext-V)<br>In dual-clock slow operation and F <sub>m</sub> is stopped           | -    | 20   | 30   | μA   |
| Hold Current (Crystal type) | IHM1            | Hold mode No load (Ext-V)<br>In dual-clock normal operation                             | -    | -    | 450  | μA   |
| Hold Current (Crystal type) | IHM3            | Hold mode No load (Ext-V)<br>In dual-clock slow operation and F <sub>m</sub> is stopped | -    | 15   | 30   | μA   |
| Stop Current (Crystal type) | ISM1            | Stop mode No load (Ext-V)<br>In dual-clock normal operation                             | -    | 1    | 2    | μA   |

# Preliminary W742E81A/W742C81A



DC Characteristics, continued

| PARAMETER                                     | SYM. | CONDITIONS                      | MIN.       | TYP. | MAX.       | UNIT  |
|---|------|---------------------------------|------------|------|------------|-------|
| Input Low Voltage                             | VIL  | -                               | VSS        | -    | 0.3<br>VDD | V     |
| Input High Voltage                            | VIH  | -                               | 0.7<br>VDD | -    | VDD        | V     |
| MFP Output Low Voltage                        | VML  | IOL = 3.5 mA                    | -          | -    | 0.4        | V     |
| MFP Output High Voltage                       | VMH  | IOH = 3.5 mA                    | 2.4        | -    | -          | V     |
| Port RA, RB, RD and RF Output Low Voltage     | VABL | IOL = 2.0 mA                    | -          | -    | 0.4        | V     |
| Port RA, RB, RD and RF Output high Voltage    | VABH | IOH = 2.0 mA                    | 2.4        | -    | -          | V     |
| LCD Supply Current                            | ILCD | All Seg. ON                     | -          | -    | 6          | μA    |
| SEG0-SEG39 Sink Current (Used as LCD output)  | IOL1 | VOL = 0.4V<br>VLCD = 0.0V       | 90         | -    | -          | μA    |
| SEG0-SEG39 Drive Current (Used as LCD output) | IOH1 | VOH = 2.4V<br>VLCD = 3.0V       | 90         | -    | -          | μA    |
| Port RE Sink Current                          | IEL  | VOL = 0.9V                      | 9          | -    | -          | mA    |
| Port RE Source Current                        | IEH  | VOH = 2.4V                      | 0.4        | 1.2  | -          | mA    |
| DTMF Output DC level                          | VTDC | RL = 5 KΩ,<br>VDD = 2.5 to 3.8V | 1.1        | -    | 2.8        | V     |
| DTMF Distortion                               | THD  | RL = 5 KΩ,<br>VDD = 2.5 to 3.8V | -          | -30  | -23        | dB    |
| DTMF Output Voltage                           | VTO  | Low group,<br>RL = 5 KΩ         | 130        | 150  | 170        | mVrms |
| Pre-emphasis                                  |      | Col/Row                         | 1          | 2    | 3          | dB    |
| DTMF Output Sink Current                      | ITL  | VTO = 0.5V                      | 0.2        | -    | -          | mA    |
| Pull-up Resistor                              | RC   | Port RC                         | 100        | 350  | 1000       | KΩ    |
| RES Pull-up Resistor                          | RRES | -                               | 20         | 100  | 500        | KΩ    |

## 9. AC CHARACTERISTICS

| PARAMETER              | SYM. | CONDITIONS        | MIN. | TYP.   | MAX. | UNIT |
|------------------------|------|-------------------|------|--------|------|------|
| Op. Frequency          | FOSC | Crystal type      | -    | 3.58   | -    | MHz  |
| Instruction Cycle time | TI   | One machine cycle | -    | 4/FOSC | -    | S    |
| Reset Active Width     | TRAW | FOSC = 32.768 KHz | 1    | -      | -    | μS   |
| Interrupt Active Width | TIAW | FOSC = 32.768 KHz | 1    | -      | -    | μS   |

# Preliminary W742E81A/W742C81A



## 10. INSTRUCTION SET TABLE

### Symbol Description

|        |                           |
|--------|---------------------------|
| ACC:   | Accumulator               |
| ACC.n: | Accumulator bit n         |
| WR:    | Working Register          |
| WRP:   | WR Page Register          |
| PAGE:  | Page Register             |
| DBKR:  | Data Bank Register        |
| ROMPR: | ROM Page Register         |
| MR0:   | Mode Register 0           |
| MR1:   | Mode Register 1           |
| PM0:   | Port Mode 0               |
| PM1:   | Port Mode 1               |
| PM2:   | Port Mode 2               |
| PM5:   | Port Mode 5               |
| PSR0:  | Port Status Register 0    |
| R:     | Memory (RAM) of address R |
| WDTR:  | WatchDog Timer Register   |
| LPL:   | LCD data RAM pointer      |
| LPH:   | LCD data RAM pointer      |
| R.n:   | Memory bit n of address R |
| SCR:   | System Control Register   |
| BUZCR: | Buzzer Control Register   |
| RA:    | I/O Port RA               |
| RC:    | I/O Port RC               |
| DTMF:  | DTMF Register             |
| DTCR:  | MTMF Control Pin          |
| MFP:   | MFP Output Pin            |

## Preliminary W742E81A/W742C81A



Symbol Description, continued

|                |  |
|----------------|--|
| 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                 |
| TAB0:          | Look-up table address buffer 0                     |
| TAB1:          | Look-up table address buffer 1                     |
| TAB2:          | Look-up table address buffer 2                     |
| TAB3:          | Look-up table address buffer 3                     |
| 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                                 |

# Preliminary W742E81A/W742C81A



| <i>Machine code</i> | <i>Mnemonic</i> | <i>Function</i>           | <i>Flag affected</i> | <i>W/C</i> |
|---------------------|-----------------|---------------------------|----------------------|------------|
| <b>Arithmetic</b>   |                 |                           |                      |            |
| 0001 1000 0xxx xxxx | ADD R, ACC      | ACC←(R) + (ACC)           | ZF, CF               | 1/1        |
| 0001 1100 iii nnn   | ADD WRn, #I     | ACC←(WRn) + I             | ZF, CF               | 1/1        |
| 0001 1001 0xxx xxxx | ADDR R, ACC     | ACC, R←(R) + (ACC)        | ZF, CF               | 1/1        |
| 0001 1101 iii nnn   | ADDR WRn, #I    | ACC, WRn←(WRn) + I        | ZF, CF               | 1/1        |
| 0000 1000 0xxx xxxx | ADC R, ACC      | ACC←(R) + (ACC) + (CF)    | ZF, CF               | 1/1        |
| 0000 1100 iii nnn   | ADC WRn, #I     | ACC←(WRn) + I + (CF)      | ZF, CF               | 1/1        |
| 0000 1001 0xxx xxxx | ADCR R, ACC     | ACC, R←(R) + (ACC) + (CF) | ZF, CF               | 1/1        |
| 0000 1101 iii nnn   | ADCR WRn, #I    | ACC, WRn←(WRn) + I + (CF) | ZF, CF               | 1/1        |
| 0010 1000 0xxx xxxx | ADU R, ACC      | ACC←(R) + (ACC)           | ZF                   | 1/1        |
| 0010 1100 iii nnn   | ADU WRn, #I     | ACC←(WRn) + I             | ZF                   | 1/1        |
| 0010 1001 0xxx xxxx | ADUR R, ACC     | ACC, R←(R) + (ACC)        | ZF                   | 1/1        |
| 0010 1101 iii nnn   | ADUR WRn, #I    | ACC, WRn←(WRn) + I        | ZF                   | 1/1        |
| 0001 1010 0xxx xxxx | SUB R, ACC      | ACC←(R) - (ACC)           | ZF, CF               | 1/1        |
| 0001 1110 iii nnn   | SUB WRn, #I     | ACC←(WRn) - I             | ZF, CF               | 1/1        |
| 0001 1011 0xxx xxxx | SUBR R, ACC     | ACC, R←(R) - (ACC)        | ZF, CF               | 1/1        |
| 0001 1111 iii nnn   | SUBR WRn, #I    | ACC, WR←(WR) - I          | ZF, CF               | 1/1        |
| 0000 1010 0xxx xxxx | SBC R, ACC      | ACC←(R) - (ACC) - (CF)    | ZF, CF               | 1/1        |
| 0000 1110 iii nnn   | SBC WRn, #I     | ACC←(WRn) - I - (CF)      | ZF, CF               | 1/1        |
| 0000 1011 0xxxxxxx  | SBCR R, ACC     | ACC, R←(R) - (ACC) - (CF) | ZF, CF               | 1/1        |
| 0000 1111 iii nnn   | SBCR WRn, #I    | ACC, WRn←(WRn) - I - (CF) | ZF, CF               | 1/1        |
| 0100 1010 0xxx xxxx | INC R           | ACC, R←(R) + 1            | ZF, CF               | 1/1        |
| 0100 1010 1xxx xxxx | DEC R           | ACC, R←(R) - 1            | ZF, CF               | 1/1        |

# Preliminary W742E81A/W742C81A



Instruction set, continued

| <b>Machine code</b> | <b>Mnemonic</b> | <b>Function</b>                        | <b>Flag affected</b> | <b>W/C</b> |
|---------------------|-----------------|--|----------------------|------------|
| <b>Logic</b>        |                 |  |                      |            |
| 0010 1010 0xxx xxxx | ANL R, ACC      | ACC←(R) & (ACC)                        | ZF                   | 1/1        |
| 0010 1110 iii nnn   | ANL WRn, #I     | ACC←(WRn) & I                          | ZF                   | 1/1        |
| 0010 1011 0xxx xxxx | ANLR R, ACC     | ACC, R←(R) & (ACC)                     | ZF                   | 1/1        |
| 0010 1111 iii nnn   | ANLR WRn, #I    | ACC, WRn←(WRn) & I                     | ZF                   | 1/1        |
| 0011 1010 0xxx xxxx | ORL R, ACC      | ACC←(R) ∨ (ACC)                        | ZF                   | 1/1        |
| 0011 1110 iii nnn   | ORL WRn, #I     | ACC←(WRn) ∨ I                          | ZF                   | 1/1        |
| 0011 1011 0xxx xxxx | ORLR R, ACC     | ACC, R←(R) ∨ (ACC)                     | ZF                   | 1/1        |
| 0011 1111 iii nnn   | ORLR WRn, #I    | ACC, WRn←(WRn) ∨ I                     | ZF                   | 1/1        |
| 0011 1000 0xxx xxxx | XRL R, ACC      | ACC←(R) EX (ACC)                       | ZF                   | 1/1        |
| 0011 1100 iii nnn   | XRL WRn, #I     | ACC←(WRn) EX I                         | ZF                   | 1/1        |
| 0011 1001 0xxx xxxx | XRLR R, ACC     | ACC, R←(R) EX (ACC)                    | ZF                   | 1/1        |
| 0011 1101 iii nnn   | XRLR WRn, #I    | ACC, WRn←(WRn) EX I                    | ZF                   | 1/1        |
| <b>Branch</b>       |                 |  |                      |            |
| 0111 0aaa aaaa aaaa | JMP L           | PC12~PC0←(ROMPR)×800H+L10~L0           |                      | 1/1        |
| 1000 0aaa aaaa aaaa | JB0 L           | PC10~PC0←L10~L0; if ACC.0 = "1"        |                      | 1/1        |
| 1001 0aaa aaaa aaaa | JB1 L           | PC10~PC0←L10~L0; if ACC.1 = "1"        |                      | 1/1        |
| 1010 0aaa aaaa aaaa | JB2 L           | PC10~PC0←L10~L0; if ACC.2 = "1"        |                      | 1/1        |
| 1011 0aaa aaaa aaaa | JB3 L           | PC10~PC0←L10~L0; if ACC.3 = "1"        |                      | 1/1        |
| 1110 0aaa aaaa aaaa | JZ L            | PC10~PC0←L10~L0; if ACC = 0            |                      | 1/1        |
| 1100 0aaa aaaa aaaa | JNZ L           | PC10~PC0←L10~L0; if ACC ≠ 0            |                      | 1/1        |
| 1111 0aaa aaaa aaaa | JC L            | PC10~PC0←L10~L0; if CF = "1"           |                      | 1/1        |
| 1101 0aaa aaaa aaaa | JNC L           | PC10~PC0←L10~L0; if CF ≠ "1"           |                      | 1/1        |
| 0100 1000 0xxx xxxx | DSKZ R          | ACC, R←(R) - 1; PC ← (PC) + 2 if ACC = | ZF, CF               | 1/1        |
| 0100 1000 1xxx xxxx | DSKNZ R         | ACC, R←(R) - 1; PC ← (PC) + 2 if ACC ≠ | ZF, CF               | 1/1        |
| 1010 1000 0xxx xxxx | SKB0 R          | PC ← (PC) + 2 if R.0 = "1"             |                      | 1/1        |
| 1010 1000 1xxx xxxx | SKB1 R          | PC ← (PC) + 2 if R.1 = "1"             |                      | 1/1        |
| 1010 1001 0xxx xxxx | SKB2 R          | PC ← (PC) + 2 if R.2 = "1"             |                      | 1/1        |
| 1010 1001 1xxx xxxx | SKB3 R          | PC ← (PC) + 2 if R.3 = "1"             |                      | 1/1        |



# Preliminary W742E81A/W742C81A



Instruction set, continued

| <i>Machine code</i>       | <i>Mnemonic</i> | <i>Function</i>                                 | <i>Flag affected</i> | <i>W/C</i> |
|---------------------------|-----------------|---|----------------------|------------|
| <b>Data move</b>          |                 |   |                      |            |
| 0001 0000 0000 iiiii      | MOV ACC, #I     | ACC←I   | ZF                   | 1/1        |
| 1110 1nmn nxxx xxxx       | MOV WRn, R      | WRn←(R)   |                      | 1/1        |
| 1001 1001 iiiii nnnn      | MOV WRn, #I     | WRn←I   |                      | 1/1        |
| 1111 1nmn nxxx xxxx       | MOV R, WRn      | R←(WRn)   |                      | 1/1        |
| 0110 1nmn nxxx xxxx       | MOVA WRn, R     | ACC, WRn←(R)                                    | ZF                   | 1/1        |
| 0111 1nmn nxxx xxxx       | MOVA R, WRn     | ACC, R←(WRn)                                    | ZF                   | 1/1        |
| 0101 1001 1xxx xxxx       | MOV R, ACC      | R←(ACC)   |                      | 1/1        |
| 0100 1110 1xxx xxxx       | MOV ACC, R      | ACC←(R)   | ZF                   | 1/1        |
| 1011 1i i i i xxx xxxx    | MOV R, #I       | R←I   |                      | 1/1        |
| 1100 1nmn n000 qqqq       | MOV WRn, @WRq   | WRn←[(DBKR)×80H+(PAGE)x10H +(WRq)]              |                      | 1/2        |
| 1101 1nmn n000 qqqq       | MOV @WRq, WRn   | [(DBKR)×80H+(PAGE)x10H +(WRq)]←WRn              |                      | 1/2        |
| 1000 1100 0xxx xxxx       | MOV TAB0, R     | TAB0←(R)  |                      | 1/1        |
| 1000 1100 1xxx xxxx       | MOV TAB1, R     | TAB1←(R)  |                      | 1/1        |
| 1000 1110 0xxx xxxx       | MOV TAB2, R     | TAB2←(R)  |                      | 1/1        |
| 1000 1110 1xxx xxxx       | MOV TAB3, R     | TAB3←(R)  |                      | 1/1        |
| 1000 1101 0xxx xxxx       | MOVC R          | R←[(TAB3)×1000H+(TAB2)x100H+(TAB1)x10H +(TAB0)] |                      | 1/2        |
| <b>Input &amp; Output</b> |                 |   |                      |            |
| 0101 1011 0xxx xxxx       | MOVA R, RA      | ACC, R←[RA]                                     | ZF                   | 1/1        |
| 0101 1011 1xxx xxxx       | MOVA R, RB      | ACC, R←[RB]                                     | ZF                   | 1/1        |
| 0100 1011 0xxx xxxx       | MOVA R, RC      | ACC, R←[RC]                                     | ZF                   | 1/1        |
| 0100 1011 1xxx xxxx       | MOVA R, RD      | ACC, R←[RD]                                     | ZF                   | 1/1        |
| 0101 1010 0xxx xxxx       | MOV RA, R       | [RA]←(R)  |                      | 1/1        |
| 0101 1010 1xxx xxxx       | MOV RB, R       | [RB]←(R)  |                      | 1/1        |
| 0100 1010 0xxx xxxx       | MOV RC, R       | [RC]←(R)  |                      | 1/1        |
| 1010 1100 1xxx xxxx       | MOV RD, R       | [RD]←(R)  |                      | 1/1        |
| 0101 1110 0xxx xxxx       | MOV RE, R       | [RE]←(R)  |                      | 1/1        |
| 1010 1110 0xxx xxxx       | MOV RF, R       | [RF]←(R)  |                      | 1/1        |

Publication Release Date: April 2000

Revision A1

# Preliminary W742E81A/W742C81A



Instruction set, continued

| <i>Machine code</i>        | <i>Mnemonic</i> | <i>Function</i>                                     | <i>Flag affected</i> | <i>W/C</i> |
|----------------------------|-----------------|---|----------------------|------------|
| <b>Flag &amp; Register</b> |                 |   |                      |            |
| 0101 1111 1xxx xxxx        | MOVA R, PAGE    | ACC, R←PAGE (Page Register)                         | ZF                   | 1/1        |
| 0101 1110 1xxx xxxx        | MOV PAGE, R     | PAGE←(R)  |                      | 1/1        |
| 0101 0110 1000 0i i i      | MOV PAGE, #I    | PAGE←I  |                      | 1/1        |
| 1001 1101 1xxx xxxx        | MOV R, WRP      | R←WRP   |                      | 1/1        |
| 1001 1100 1xxx xxxx        | MOV WRP, R      | WRP←(R)   |                      | 1/1        |
| 0011 0101 1000 i i i i     | MOV WRP, #I     | WRP←I   |                      | 1/1        |
| 1001 1101 0000 nmn         | MOV WRn,DBKR    | WRn←DBKR  |                      | 1/1        |
| 1001 1111 0000 nmn         | MOV WRn,TM1     | WRn←TM1.4 - TM1.7, ACC←TM1.0 - TM1.3                |                      | 1/1        |
| 1001 1100 0000 nmn         | MOV DBKR, WRn   | DBKR←WRn  |                      | 1/1        |
| 0011 0101 0000 ii i i      | MOV DBKR, #I    | DBKR←I  |                      | 1/1        |
| 0011 0100 0000 0ii i       | MOV ROMPR, #I   | ROMPR←I   |                      | 1/1        |
| 1000 1000 0xxx xxxx        | MOV ROMPR, R    | ROMPR←(R)   |                      | 1/1        |
| 1000 1001 0xxx xxxx        | MOV R, ROMPR    | R←(ROMPR)   |                      | 1/1        |
| 0101 1001 0xxx xxxx        | MOVA R, CF      | ACC.0, R.0←CF                                       | ZF                   | 1/1        |
| 0101 1000 0xxx xxxx        | MOV CF, R       | CF←(R.0)  | CF                   | 1/1        |
| 0100 1001 0xxx xxxx        | MOVA R, HCF.L   | ACC, R←HCF.0~HCF.3                                  | ZF                   | 1/1        |
| 0100 1001 1xxx xxxx        | MOVA R, HCF.H   | ACC, R←HCF.4~HCF.7                                  | ZF                   | 1/1        |
| 0101 0011 0000 i i i i     | MOV PM0, #I     | Port Mode 0← I                                      |                      | 1/1        |
| 0101 0111 0000 i i i i     | MOV PM1, #I     | Port Mode 1← I                                      |                      | 1/1        |
| 0101 0111 1000 i i i i     | MOV PM2, #I     | Port Mode 2← I                                      |                      | 1/1        |
| 0011 0111 1000 i i i i     | MOV PM5, #I     | Port Mode 5← I                                      |                      | 1/1        |
| 0100 0000 i00i 0i i i      | CLR EVF, #I     | Clear Event Flag if In = 1                          |                      | 1/1        |
| 0101 1101 0xxx xxxx        | MOVA R, EVFL    | R← EVF.0 - EVF.3                                    |                      | 1/1        |
| 0101 1101 1xxx xxxx        | MOVA R, EVFH    | R← EVF.4 - EVF.7                                    |                      | 1/1        |
| 0100 0001 i00i 0i i i      | MOV HEF, #I     | Set/Reset HOLD mode release Enable Flag             |                      | 1/1        |
| 0101 0001 i00i 0i i i      | MOV IEF, #I     | Set/Reset Interrupt Enable Flag                     |                      | 1/1        |
| 0100 0011 0000 i i i i     | MOV PEF, #I     | Set/Reset Port Enable Flag                          |                      | 1/1        |
| 0101 0010 0000 i i i i     | MOV SEF, #I     | Set/Reset STOP mode wake-up Enable Flag for RC port |                      | 1/1        |

# Preliminary W742E81A/W742C81A



Instruction set, continued

| <i>Machine code</i>           | <i>Mnemonic</i> | <i>Function</i>                              | <i>Flag affected</i> | <i>W/C</i> |
|-------------------------------|-----------------|--|----------------------|------------|
| <b>Flag &amp; Register</b>    |                 |  |                      |            |
| 0101 0100 0000 <i>i 0 i i</i> | MOV SCR, #I     | SCR←I  |                      | 1/1        |
| 0100 1111 0xxx xxxx           | MOVA R, PSR0    | ACC, R←Port Status Register 0                | ZF                   | 1/1        |
| 0100 0010 0000 0000           | CLR PSR0        | Clear Port Status Register 0                 |                      | 1/1        |
| 0101 0000 0100 0000           | SET CF          | Set Carry Flag                               | CF                   | 1/1        |
| 0101 0000 0000 0000           | CLR CF          | Clear Carry Flag                             | CF                   | 1/1        |
| 0001 0111 0000 0000           | CLR DIVR0       | Clear the last 4-bit of the Divider 0        |                      | 1/1        |
| 0101 0101 1000 0000           | CLR DIVR1       | Clear the last 4-bit of the Divider 1        |                      | 1/1        |
| 0101 0110 0000 <i>i i i i</i> | MOV WDTR, #I    | WDTR←I                                       |                      | 1/1        |
| 0101 1111 0xxx xxxx           | MOVA R, WDTR    | ACC, R←Watchdog Timer Register               |                      | 1/1        |
| 0001 0111 1000 0000           | CLR WDT         | Clear Watchdog Timer                         |                      | 1/1        |
| <b>DTMF</b>                   |                 |  |                      |            |
| 1001 1110 1xxx xxxx           | MOV DTMF,R      | DTMF←(R)                                     |                      | 1/1        |
| 0011 0100 1000 0iii           | MOV DTCR,I      | DTCR←I                                       |                      | 1/1        |
| <b>Shift &amp; Rotate</b>     |                 |  |                      |            |
| 0100 1101 0xxx xxxx           | SHRC R          | ACC.n, R.n←(R.n+1);<br>ACC.3, R.3←0; CF←R.0  | ZF, CF               | 1/1        |
| 0100 1101 1xxx xxxx           | RRC R           | ACC.n, R.n←(R.n+1);<br>ACC.3, R.3←CF; CF←R.0 | ZF, CF               | 1/1        |
| 0100 1100 0xxx xxxx           | SHLC R          | ACC.n, R.n←(R.n-1);<br>ACC.0, R.0←0; CF←R.3  | ZF, CF               | 1/1        |
| 0100 1100 1xxx xxxx           | RLC R           | ACC.n, R.n←(R.n-1);<br>ACC.0, R.0←CF; CF←R.3 | ZF, CF               | 1/1        |

Publication Release Date: April 2000

# Preliminary W742E81A/W742C81A



Instruction set, continued

| <i>Machine code</i>  | <i>Mnemonic</i> | <i>Function</i>  | <i>Flag affected</i> | <i>W/C</i> |
|----------------------|-----------------|--|----------------------|------------|
| <b>LCD</b>           |                 |  |                      |            |
| 1001 1000 0xxx xxxx  | MOV LPL, R      | LPL←(R)  |                      | 1/1        |
| 1001 1000 1xxx xxxx  | MOV LPH, R      | LPH←(R)  |                      | 1/1        |
| 1001 1010 0xxx xxxx  | MOV @LP, R      | [(LPH)×10H+(LPL)]←(R)  |                      | 1/1        |
| 1001 1011 0xxx xxxx  | MOV R, @LP      | R←[(LPH)×10H+(LPL)]  |                      | 1/1        |
| 0000 0010 0000 0000  | LCDON           | LCD ON   |                      | 1/1        |
| 0000 0010 1000 0000  | LCDOFF          | LCD OFF  |                      | 1/1        |
| <b>MFP</b>           |                 |  |                      |            |
| 0011 0110 0000 000i  | MOV BUZCR, #I   | BUZCR← I   |                      | 1/1        |
| 1000 1010 0xxx xxxx  | MOV BUZCR, R    | BUZCR←(R)  |                      | 1/1        |
| 1000 1011 0xxx xxxx  | MOV R, BUZCR    | R←(BUZCR)  |                      | 1/1        |
| 0001 0010 iiiiiiii   | MOV MFP, #I     | [MFP]← I   |                      | 1/1        |
| <b>Timer</b>         |                 |  |                      |            |
| 1010 1010 0xxx xxxx  | MOV TM0L, R     | TM0L←(R)   |                      | 1/1        |
| 1010 1010 1xxx xxxx  | MOV TM0H, R     | TM0H←(R)   |                      | 1/1        |
| 1010 1011 0xxx xxxx  | MOV TM1L, R     | TM1L←(R)   |                      | 1/1        |
| 1010 1011 1xxx xxxx  | MOV TM1H, R     | TM1H←(R)   |                      | 1/1        |
| 0001 0011 1000 i00i  | MOV MR0, #I     | MR0←(R)  |                      | 1/1        |
| 0001 0011 0000 iiiii | MOV MR1, #I     | MR1←(R)  |                      | 1/1        |
| <b>Other</b>         |                 |  |                      |            |
| 0000 0000 1000 0000  | HOLD            | Enter Hold mode  |                      | 1/1        |
| 0000 0000 1100 0000  | STOP            | Enter Stop mode  |                      | 1/1        |
| 0000 0000 0000 0000  | NOP             | No operation   |                      | 1/1        |
| 0101 0000 1100 0000  | EN INT          | Enable interrupt function  |                      | 1/1        |
| 0101 0000 1000 0000  | DIS INT         | Disable interrupt function   |                      | 1/1        |
| <b>Subroutine</b>    |                 |  |                      |            |
| 0110 0aaa aaaa aaaa  | CALL L          | Push Stack:<br>STACK <- PC+1, TAB0, TAB1, TAB2, TAB3,<br>DBKR, WRP, ROMPR, PAGE, ACC, CF;<br>PC12~PC0 <- (ROMPR)×800H+L10~L0 |                      | 1/1        |
| 0000 0001 iiiiiiii   | RTN #I          | (PC) <- STACK; Pop other register by I Table setting (Refer to Table 8)  |                      | 1/1        |

# Preliminary W742E81A/W742C81A



**Table 8 The bit definition of RTN**

| BIT DEFINITION OF I |  |
|---------------------|--|
| I = 0000 0000       | Pop <b>PC</b> from stack only                              |
| bit0 = 1            | Pop <b>PC</b> and <b>TAB0, TAB1, TAB2, TAB3</b> from stack |
| bit1 = 1            | Pop <b>PC</b> and <b>DBKR</b> from stack                   |
| bit2 = 1            | Pop <b>PC</b> and <b>WRP</b> from stack                    |
| bit3 = 1            | Pop <b>PC</b> and <b>ROMPR</b> from stack                  |
| bit4 = 1            | Pop <b>PC</b> and <b>PAGE</b> from stack                   |
| bit5 = 1            | Pop <b>PC</b> and <b>ACC</b> from stack                    |
| bit6 = 1            | Pop <b>PC</b> and <b>CF</b> from stack                     |



**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

Note: All data and specifications are subject to change without notice.