**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | 8052 |
| Core Size | 8-Bit |
| Speed | 40MHz |
| Connectivity | EBI/EMI, UART/USART |
| Peripherals | POR, WDT |
| Number of I/O | 36 |
| Program Memory Size | 8KB (8K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 256 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.4V ~ 5.5V |
| Data Converters | - |
| Oscillator Type | External |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 44-LCC (J-Lead) |
| Supplier Device Package | - |
| Purchase URL | https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/w78i052dpg |

## 2 FEATURES

- Fully static design 8-bit CMOS microcontroller
- Optional 12T or 6T mode
  - 12T Mode, 12 clocks per machine cycle operation (default), Speed up to 40 MHz/5V
  - 6T Mode, 6 clocks per machine cycle operation set by the writer, Speed up to 20 MHz/5V
- Wide supply voltage of 2.4V to 5.5V
- Temperature grade is (-40°C~85°C)
- Pin and Instruction-sets compatible with MCS-51
- 256 bytes of on-chip scratchpad RAM
- 16K/8K/4K bytes electrically erasable/programmable Flash EPROM
- 2K bytes LDROM support ISP function (Reference Application Note)
- 64KB program memory address space
- 64KB data memory address space
- Four 8-bit bi-directional ports
- 8-sources, 4-level interrupt capability
- One extra 4-bit bit-addressable I/O port, additional $\overline{INT2}$ / $\overline{INT3}$ (available on PQFP, PLCC and LQFP package)
- Three 16-bit timer/counters
- One full duplex serial port
- Watchdog Timer
- EMI reduction mode
- Software Reset
- Built-in power management with idle mode and power down mode
- Code protection
- Packages:
  - Lead Free (RoHS) DIP 40: W78I054DDG
  - Lead Free (RoHS) PLCC 44: W78I054DPG
  - Lead Free (RoHS) PQFP 44: W78I054DFG
  - Lead Free (RoHS) LQFP 48: W78I054DLG
  - Lead Free (RoHS) DIP 40: W78I052DDG
  - Lead Free (RoHS) PLCC 44: W78I052DPG
  - Lead Free (RoHS) PQFP 44: W78I052DFG
  - Lead Free (RoHS) LQFP 48: W78I052DLG
  - Lead Free (RoHS) DIP 40: W78I051DDG
  - Lead Free (RoHS) PLCC 44: W78I051DPG
  - Lead Free (RoHS) PQFP 44: W78I051DFG
  - Lead Free (RoHS) LQFP 48: W78I051DLG

## 7.6 Data Pointers

The data pointer of W78I054D/W78I052D/W78I051D series is same as standard 8052 that have one 16-bit Data Pointer (DPTR).

## 7.7 Architecture

The W78I054D/W78I052D/W78I051D series are based on the standard 8052 device. It is built around an 8-bit ALU that uses internal registers for temporary storage and control of the peripheral devices. It can execute the standard 8052 instruction set.

### 7.7.1 ALU

The ALU is the heart of the W78I054D/W78I052D/W78I051D series. It is responsible for the arithmetic and logical functions. It is also used in decision making, in case of jump instructions, and is also used in calculating jump addresses. The user cannot directly use the ALU, but the Instruction Decoder reads the op-code, decodes it, and sequences the data through the ALU and its associated registers to generate the required result. The ALU mainly uses the ACC which is a special function register (SFR) on the chip. Another SFR, namely B register is also used Multiply and Divide instructions. The ALU generates several status signals which are stored in the Program Status Word register (PSW).

### 7.7.2 Accumulator

The Accumulator (ACC) is the primary register used in arithmetic, logical and data transfer operations in the W78I054D/W78I052D/W78I051D series. Since the Accumulator is directly accessible by the CPU, most of the high speed instructions make use of the ACC as one argument.

### 7.7.3 B Register

This is an 8-bit register that is used as the second argument in the MUL and DIV instructions. For all other instructions it can be used simply as a general purpose register.

### 7.7.4 Program Status Word

This is an 8-bit SFR that is used to store the status bits of the ALU. It holds the Carry flag, the Auxiliary Carry flag, General purpose flags, the Register Bank Select, the Overflow flag, and the Parity flag.

### 7.7.5 Scratch-pad RAM

The W78I054D/W78I052D/W78I051D series has a 256 byte on-chip scratch-pad RAM. This can be used by the user for temporary storage during program execution. A certain section of this RAM is bit addressable, and can be directly addressed for this purpose.

### 7.7.6 Stack Pointer

The W78I054D/W78I052D/W78I051D series has an 8-bit Stack Pointer which points to the top of the Stack. This stack resides in the Scratch Pad RAM in the W78I054D/W78I052D/W78I051D. Hence the size of the stack is limited by the size of this RAM.

| | | |
|---|---|---|
| 4 | TR0 | Timer 0 Run Control. This bit is set or cleared by software to turn timer/counter on or off. |
| 3 | IE1 | Interrupt 1 Edge Detect Flag: Set by hardware when an edge/level is detected on $\overline{INT1}$. This bit is cleared by hardware when the service routine is vectored to only if the interrupt was edge triggered. Otherwise it follows the inverse of the pin. |
| 2 | IT1 | Interrupt 1 Type Control. Set/cleared by software to specify falling edge/ low level triggered external inputs. |
| 1 | IE0 | Interrupt 0 Edge Detect Flag. Set by hardware when an edge/level is detected on $\overline{INT0}$. This bit is cleared by hardware when the service routine is vectored to only if the interrupt was edge triggered. Otherwise it follows the inverse of the pin. |
| 0 | IT0 | Interrupt 0 Type Control: Set/cleared by software to specify falling edge/ low level triggered external inputs. |

**Timer Mode Control**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 |
| | TIMER1 | | | | TIMER0 | | | |

Mnemonic: TMOD Address: 89h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7 | GATE | Gating control: When this bit is set, Timer/counter 1 is enabled only while the $\overline{INT1}$ pin is high and the TR1 control bit is set. When cleared, the $\overline{INT1}$ pin has no effect, and Timer 1 is enabled whenever TR1 control bit is set. |
| 6 | C/$\overline{T}$ | Timer or Counter Select: When clear, Timer 1 is incremented by the internal clock. When set, the timer counts falling edges on the T1 pin. |
| 5 | M1 | Timer 1 mode select bit 1. See table below. |
| 4 | M0 | Timer 1 mode select bit 0. See table below. |
| 3 | GATE | Gating control: When this bit is set, Timer/counter 0 is enabled only while the $\overline{INT0}$ pin is high and the TR0 control bit is set. When cleared, the $\overline{INT0}$ pin has no effect, and Timer 0 is enabled whenever TR0 control bit is set. |
| 2 | C/$\overline{T}$ | Timer or Counter Select: When clear, Timer 0 is incremented by the internal clock. When set, the timer counts falling edges on the T0 pin. |
| 1 | M1 | Timer 0 mode select bit 1. See table below. |
| 0 | M0 | Timer 0 mode select bit 0. See table below. |

**M1, M0: Mode Select bits:**

| M1 | M0 | MODE |
|---|---|---|
| 0 | 0 | Mode 0: 13-bit timer/counter TLx serves as 5-bit pre-scale. |
| 0 | 1 | Mode 1: 16-bit timer/counter, no pre-scale. |
| 1 | 0 | Mode 2: 8-bit timer/counter with auto-reload from THx. |
| 1 | 1 | Mode 3: (Timer 0) TL0 is an 8-bit timer/counter controlled by the standard Timer0 control bits. TH0 is an 8-bit timer only controlled by Timer1 control bits. (Timer 1) |

| BIT | NAME | FUNCTION |
|---|---|---|
| 7~0 | SBUF | Serial data on the serial port is read from or written to this location. It actually consists of two separate internal 8-bit registers. One is the receive resister, and the other is the transmit buffer. Any read access gets data from the receive data buffer, while write access is to the transmit data buffer. |

**Port 2**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | P2.7 | P2.6 | P2.5 | P2.4 | P2.3 | P2.2 | P2.1 | P2.0 |

Mnemonic: P2 Address: A0h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7-0 | P2.[7:0] | Port 2 is a bi-directional I/O port with internal pull-ups. This port also provides the upper address bits for accesses to external memory. |

**Interrupt Enable**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EA | - | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

Mnemonic: IE Address: A8h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7 | EA | Global enable. Enable/Disable all interrupts. |
| 6 | - | Reserved |
| 5 | ET2 | Enable Timer 2 interrupt. |
| 4 | ES | Enable Serial Port 0 interrupt. |
| 3 | ET1 | Enable Timer 1 interrupt. |
| 2 | EX1 | Enable external interrupt 1. |
| 1 | ET0 | Enable Timer 0 interrupt. |
| 0 | EX0 | Enable external interrupt 0. |

**Port 3**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | P3.7 | P3.6 | P3.5 | P3.4 | P3.3 | P3.2 | P3.1 | P3.0 |

Mnemonic: P3 Address: B0h

P3.7-0: General purpose Input/Output port. Most instructions will read the port pins in case of a port read access, however in case of read-modify-write instructions, the port latch is read. These alternate functions are described below:

| BIT | NAME | FUNCTION |
|---|---|---|
| 7 | P3.7 | $\overline{RD}$ |

| | | |
|---|---|---|
| 6 | P3.6 | $\overline{WR}$ |
| 5 | P3.5 | T1 |
| 4 | P3.4 | T0 |
| 3 | P3.3 | $\overline{INT1}$ |
| 2 | P3.2 | $\overline{INT0}$ |
| 1 | P3.1 | TX |
| 0 | P3.0 | RX |

**Interrupt High Priority**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IPH.7 | IPH.6 | IPH.5 | IPH.4 | IPH.3 | IPH.2 | IPH.1 | IPH.0 |

Mnemonic: IPH Address: B7h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7 | IPH.7 | 1: Interrupt high priority of INT3 is highest priority level. |
| 6 | IPH.6 | 1: Interrupt high priority of INT2 is highest priority level. |
| 5 | IPH.5 | 1: Interrupt high priority of Timer 2 is highest priority level. |
| 4 | IPH.4 | 1: Interrupt high priority of Serial Port 0 is highest priority level. |
| 3 | IPH.3 | 1: Interrupt high priority of Timer 1 is highest priority level. |
| 2 | IPH.2 | 1: Interrupt high priority of External interrupt 1 is highest priority level. |
| 1 | IPH.1 | 1: Interrupt high priority of Timer 0 is highest priority level. |
| 0 | IPH.0 | 1: Interrupt high priority of External interrupt 0 is highest priority level. |

**Interrupt Priority**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | - | - | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

Mnemonic: IP Address: B8h

| BIT | NAME | FUNCTION |
|---|---|---|
| 5 | PT2 | 1: Interrupt priority of Timer 2 is higher priority level. |
| 4 | PS | 1: Interrupt priority of Serial port 0 is higher priority level. |
| 3 | PT1 | 1: Interrupt priority of Timer 1 is higher priority level. |
| 2 | PX1 | 1: Interrupt priority of External interrupt 1 is higher priority level. |
| 1 | PT0 | 1: Interrupt priority of Timer 0 is higher priority level. |
| 0 | PX0 | 1: Interrupt priority of External interrupt 0 is higher priority level. |

**EAPAGE ERASE PAGE Operation Modes**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | EAPG1 | EAPG0 |

Mnemonic: EAPAGE Address: BD

| BIT | NAME | FUNCTION |
|---|---|---|
| 1 | EAPG1 | 1.To ease PAGE1 when ease command is set.(LD flash) |
| 0 | EAPG0 | 1.To ease PAGE0 when ease command is set. (AP Flash) |

```
;CPU Clock = 12MHz/12T mode
READ_TIME           EQU     1
PROGRAM_TIME        EQU     50
ERASE_TIME          EQU     5000
```

Erase_AP Flash:

```
    mov     EAPAGE,#01h         ;set EAPAGE is APROM
    mov     SFRCN,#ERASE_ROM
    mov     TL0,#LOW (65536-ERASE_TIME)
    mov     TH0,#HIGH(65536-ERASE_TIME)
    setb    TR0
    mov     CHPCON,#00000011b
    mov     EAPAGE,#00h         ;clear EAPAGE
    clr     TF0
    clr     TR0
    ret
```

Erase_LD Flash:

```
    mov     EAPAGE,#02h         ;set EAPAGE is LDROM
    mov     SFRCN,#ERASE_ROM
    mov     TL0,#LOW (65536-ERASE_TIME)
    mov     TH0,#HIGH(65536-ERASE_TIME)
    setb    TR0
    mov     CHPCON,#00000011b
    mov     EAPAGE,#00h         ;clear EAPAGE
    clr     TF0
    clr     TR0
    ret
```

**Chip Control**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SWRST | - | - | - | - | - | FBOOTSL | ENP |

Mnemonic: CHPCON Address: BFh

| Bit | Name | Function |
|---|---|---|
| 7 | SWRST | When this bit is set to 1, and both FBOOTSL and ENP are set to 1. It will enforce microcontroller reset to initial condition just like power on reset. This action will re-boot the microcontroller and start to normal operation. To read this bit can determine that the F02KBOOT mode is running.<br>Note: Software Reset only LDROM jump to APROM, APROM can't software reset to LDROM. |
| 1 | FBOOTSL | The Loader Program Location Select.<br>0: The Loader Program locates at the APROM flash memory bank.<br>1: The Loader Program locates at the LD flash memory bank. |
| 0 | ENP | FLASH EPROM Programming Enable.<br>1: Enable in-system programming mode. In this mode, erase, program and read operations are achieved.<br>0: Disable in-system programming mode. The on-chip flash memory is read-only. |

**External Interrupt Control**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PX3 | EX3 | IE3 | IT3 | PX2 | EX2 | IE2 | IT2 |

Mnemonic: XICON Address: C0h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7 | PX3 | External interrupt 3 priority is higher if set this bit to 1 |
| 6 | EX3 | Enable External interrupt 3 if set this bit to 1 |
| 5 | IE3 | If IT3 = 1, IE3 is set/cleared automatically by hardware when interrupt is detected/serviced |
| 4 | IT3 | External interrupt 3 is falling-edge/low-level triggered when this bit is set/cleared by software |
| 3 | PX2 | External interrupt 2 priority is higher if set this to 1 |
| 2 | EX2 | Enable External interrupt 2 if set this bit to 1 |
| 1 | IE2 | If IT2 = 1, IE2 is set/cleared automatically by hardware when interrupt is detected/serviced |
| 0 | IT2 | External interrupt 2 is falling-edge/low-level triggered when this bit is set/cleared by software |

**SFR program of address low**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SFRAL.7 | SFRAL.6 | SFRAL.5 | SFRAL.4 | SFRAL.3 | SFRAL.2 | SFRAL.1 | SFRAL.0 |

Mnemonic: SFRAL Address: C4h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7-0 | SFRAL.[7:0] | The programming address of on-chip flash memory in programming mode. SFRFAL contains the low-order byte of address. |

**SFR program of address high**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SFRAH.7 | SFRAH.6 | SFRAH.5 | SFRAH.4 | SFRAH.3 | SFRAH.2 | SFRAH.1 | SFRAH.0 |

Mnemonic: SFRAH Address: C5h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7-0 | SFRAH.[7:0] | The programming address of on-chip flash memory in programming mode. SFRFAH contains the high-order byte of address. |

**SFR program For Data**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SFRFD.7 | SFRFD.6 | SFRFD.5 | SFRFD.4 | SFRFD.3 | SFRFD.2 | SFRFD.1 | SFRFD.0 |

Mnemonic: SFRFD Address: C6h

| BIT | NAME | FUNCTION |
|---|---|---|
| 7-0 | SFRFD.[7:0] | The programming data for on-chip flash memory in programming mode. |

**SFR for Program Control**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | - | | OEN | CEN | CTRL3 | CTRL2 | CTRL1 | CTRL0 |

Mnemonic: SFRCN Address: C7h

| BIT | NAME | FUNCTION |
|---|---|---|
| 5 | OEN | FLASH EPROM output enable. |
| 4 | CEN | FLASH EPROM chip enable. |
| 3-0 | CTRL[3:0] | CTRL[3:0]: The flash control signals |

| Mode | OEN | CEN | CTRL<3:0> | SFRAH, SFRAL | SFRFD |
|---|---|---|---|---|---|
| Flash Standby | 1 | 1 | X | X | X |
| Read Company ID | 0 | 0 | 1011 | 0FFh, 0FFh | Data out |
| Read Device ID High | 0 | 0 | 1100 | 0FFh, 0FFh | Data out |
| Read Device ID Low | 1 | 0 | 1100 | 0FFh, 0FEh | Data out |
| Erase APROM | 1 | 0 | 0010 | X | X |
| Erase Verify APROM | 0 | 0 | 1001 | Address in | Data out |
| Program APROM | 1 | 0 | 0001 | Address in | Data in |

| | | |
|---|---|---|
| 7-0 | ACC | The A or ACC register is the standard 8052 accumulator. |

**B Register**

| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | B.7 | B.6 | B.5 | B.4 | B.3 | B.2 | B.1 | B.0 |

Mnemonic: B Address: F0h

| Bit | Name | Function |
|---|---|---|
| 7-0 | B | The B register is the standard 8052 register that serves as a second accumulator. |

# 11 INSTRUCTION TIMING

A machine cycle consists of a sequence of 6 states, numbered S1 through S6. Each state time lasts for two oscillator periods. Thus a machine cycle takes 12 oscillator periods or 1us if the oscillator frequency is 12MHz.

Each state is divided into a Phase 1 half and a Phase 2 half. The fetch/execute sequences in states and phases for various kinds of instructions. Normally two program fetches are generated during each machine cycle, even if the instruction being executed doesn't require it. If the instruction being executed doesn't need more code bytes, the CPU simply ignores the extra fetch, and the Program Counter is not incremented. Execution of a one-cycle instruction begins during State 1 of the machine cycle, when the OPCODE is latched into the Instruction Register. A second fetch occurs during S4 of the same machine cycle. Execution is complete at the end of State 6 of this machine cycle.

The MOVX instructions take two machine cycles to execute. No program fetch is generated during the second cycle of a MOVX instruction. This is the only time program fetches are skipped. The fetch/execute sequence for MOVX instructions.

The fetch/execute sequences are the same whether the Program Memory is internal or external to the chip. Execution times do not depend on whether the Program Memory is internal or external.

the signals and timing involved in program fetches when the Program Memory is external. If Program Memory is external, then the Program Memory read strobe PSEN is normally activated twice per machine cycle. If an access to external Data Memory occurs, two PSEN pulse are skipped, because the address and data bus are being used for the Data Memory access. Note that a Data Memory bus cycle takes twice as much time as a Program Memory bus cycle.

# 12 POWER MANAGEMENT

The W78I054D/W78I052D/W78I051D has several features that help the user to control the power consumption of the device. The power saved features have basically the POWER DOWN mode and the IDLE mode of operation.

## 12.1 Idle Mode

The user can put the device into idle mode by writing 1 to the bit PCON.0. The instruction that sets the idle bit is the last instruction that will be executed before the device goes into Idle Mode. In the Idle mode, the clock to the CPU is halted, but not to the Interrupt, Timer, Watchdog timer and Serial port blocks. This forces the CPU state to be frozen; the Program counter, the Stack Pointer, the Program Status Word, the Accumulator and the other registers hold their contents. The port pins hold the logical states they had at the time Idle was activated. The Idle mode can be terminated in two ways. Since the interrupt controller is still active, the activation of any enabled interrupt can wake up the processor. This will automatically clear the Idle bit, terminate the Idle mode, and the Interrupt Service Routine(ISR) will be executed. After the ISR, execution of the program will continue from the instruction which put the device into Idle mode.

The Idle mode can also be exited by activating the reset. The device can put into reset either by applying a high on the external RST pin, a Power on reset condition or a Watchdog timer reset. The external reset pin has to be held high for at least two machine cycles i.e. 24 clock periods to be recognized as a valid reset. In the reset condition the program counter is reset to 0000h and all the SFRs are set to the reset condition. Since the clock is already running there is no delay and execution starts immediately.

## 12.2 Power Down Mode

The device can be put into Power Down mode by writing 1 to bit PCON.1. The instruction that does this will be the last instruction to be executed before the device goes into Power Down mode. In the Power Down mode, all the clocks are stopped and the device comes to a halt. All activity is completely stopped and the power consumption is reduced to the lowest possible value. The port pins output the values held by their respective SFRs.

The W78I054D/W78I052D/W78I051D will exit the Power Down mode with a reset or by an external interrupt pin enabled as level detect. An external reset can be used to exit the Power down state. The high on RST pin terminates the Power Down mode, and restarts the clock. The program execution will restart from 0000h. In the Power down mode, the clock is stopped, so the Watchdog timer cannot be used to provide the reset to exit Power down mode.

The W78I054D/W78I052D/W78I051D can be woken from the Power Down mode by forcing an external interrupt pin activated, provided the corresponding interrupt is enabled, while the global enable(EA) bit is set and the external input has been set to a level detect mode. If these conditions are met, then the high level on the external pin re-starts the oscillator. Then device executes the interrupt service routine for the corresponding external interrupt. After the interrupt service routine is completed, the program execution returns to the instruction after one which put the device into Power Down mode and continues from there.

service routine currently being executed. If the polling cycle is not the last machine cycle of the instruction being executed, then an additional delay is introduced. The maximum response time (if no other interrupt is in service) occurs if the device is performing a write to IE, IP, IPH and then executes a MUL or DIV instruction.

## 13.6 Interrupt Inputs

Since the external interrupt pins are sampled once each machine cycle, an input high or low should hold for at least one machine cycle to ensure proper sampling. If the external interrupt is high for at least one machine cycle, and then hold it low for at least one machine cycle. This is to ensure that the transition is seen and that interrupt request flag IEn is set. IEn is automatically cleared by the CPU when the service routine is called.

If the external interrupt is level-activated, the external source must hold the request active until the requested interrupt is actually generated. If the external interrupt is still asserted when the interrupt service routine is completed another interrupt will be generated. It is not necessary to clear the interrupt flag IEn when the interrupt is level sensitive, it simply tracks the input pin level.

If an external interrupt is enabled when the W78I054D/W78I052D/W78I051D is put into Power Down or Idle mode, the interrupt will cause the processor to wake up and resume operation. Refer to the section on Power Reduction Modes for details.

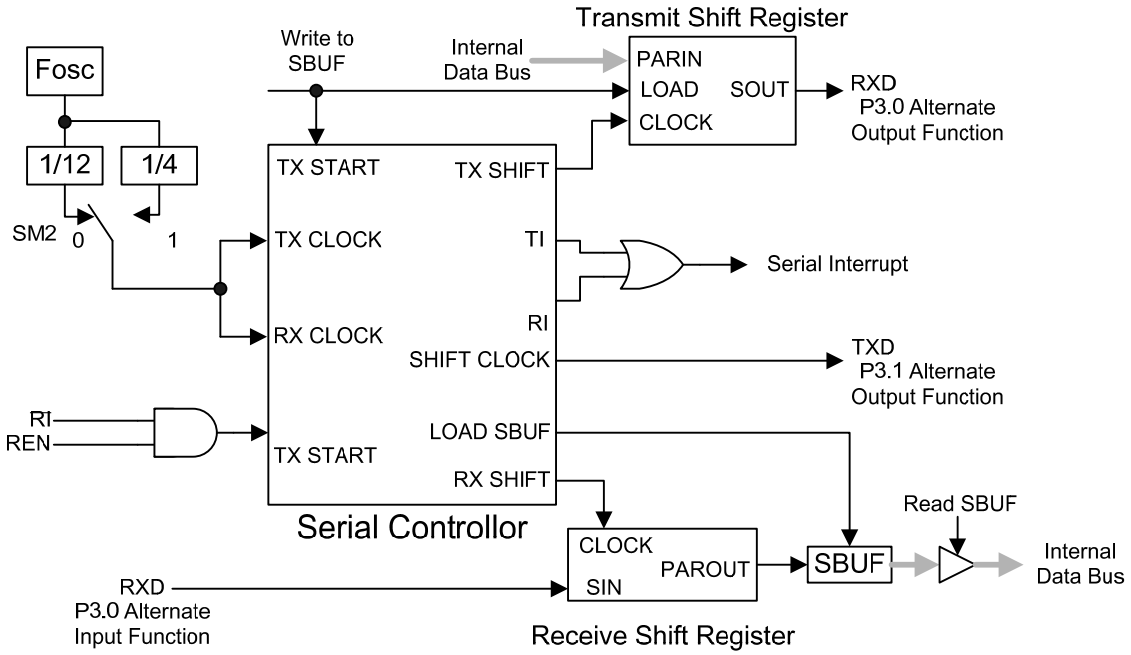



Figure 16- 1 Serial port mode 0

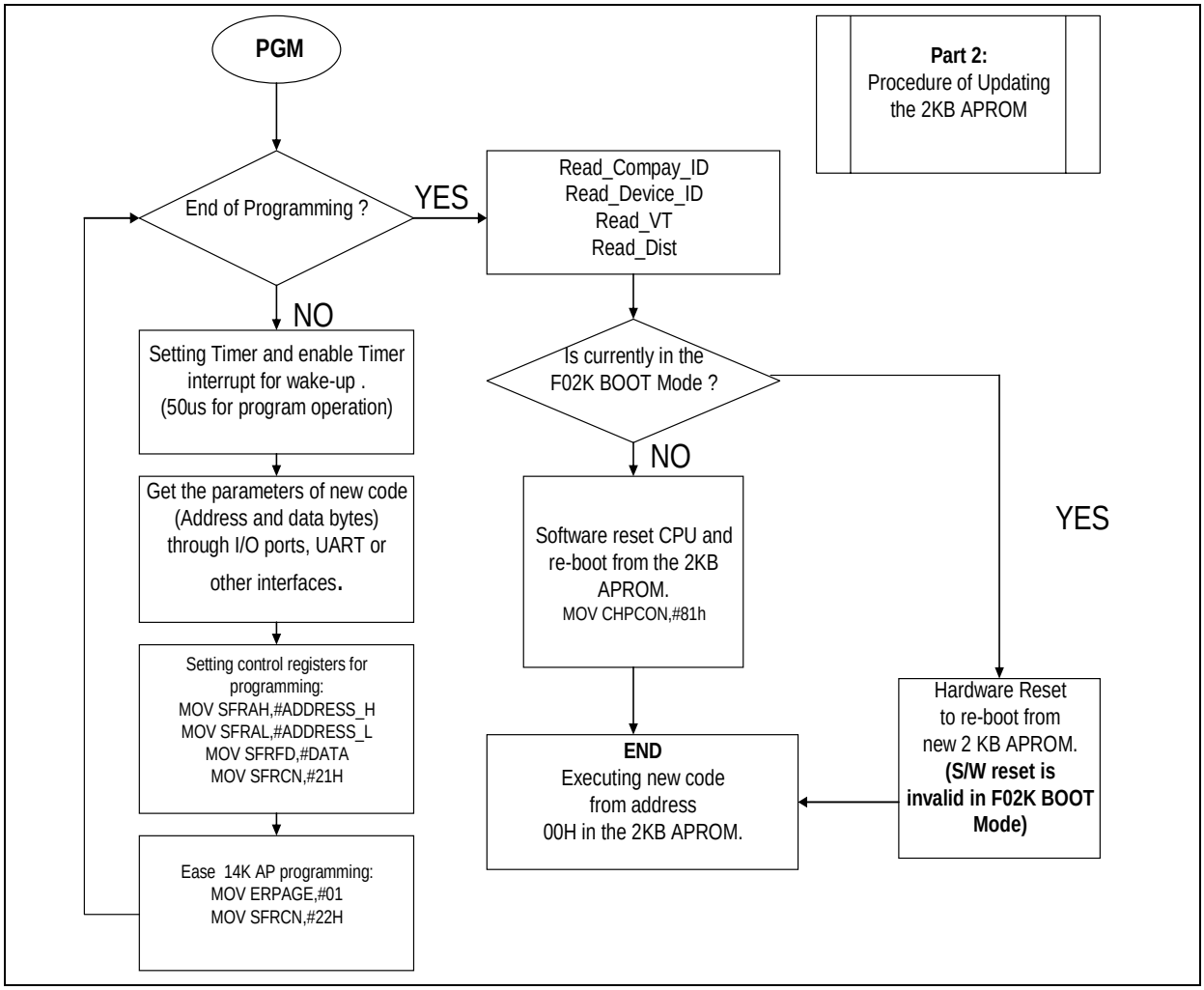
The TI flag is set high in S6P2 following the end of transmission of the last bit. The serial port will receive data when REN is 1 and RI is zero. The shift clock (TxD) will be activated and the serial port will latch data on the rising edge of shift clock. The external device should therefore present data on the falling edge on the shift clock. This process continues till all the 8 bits have been received. The RI flag is set in S6P2 following the last rising edge of the shift clock on TxD. This will stop reception, till the RI is cleared by software.

## 16.2 MODE 1

In Mode 1, the full duplex asynchronous mode is used. Serial communication frames are made up of 10 bits transmitted on TXD and received on RXD. The 10 bits consist of a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in the SFR SCON. The baud rate in this mode is variable. The serial baud can be programmed to be 1/16 or 1/32 of the Timer 1 overflow. Since the Timer 1 can be set to different reload values, a wide variation in baud rates is possible.

Transmission begins with a write to SBUF. The serial data is brought out on to TxD pin at S6P2 following the first roll-over of divide by 16 counter. The next bit is placed on TxD pin at S6P2 following the next rollover of the divide by 16 counter. Thus the transmission is synchronized to the divide by 16 counter and not directly to the write to SBUF signal. After all 8 bits of data are transmitted, the stop bit is transmitted. The TI flag is set in the S6P2 state after the stop bit has been put out on TxD pin. This will be at the 10th rollover of the divide by 16 counters after a write to SBUF.

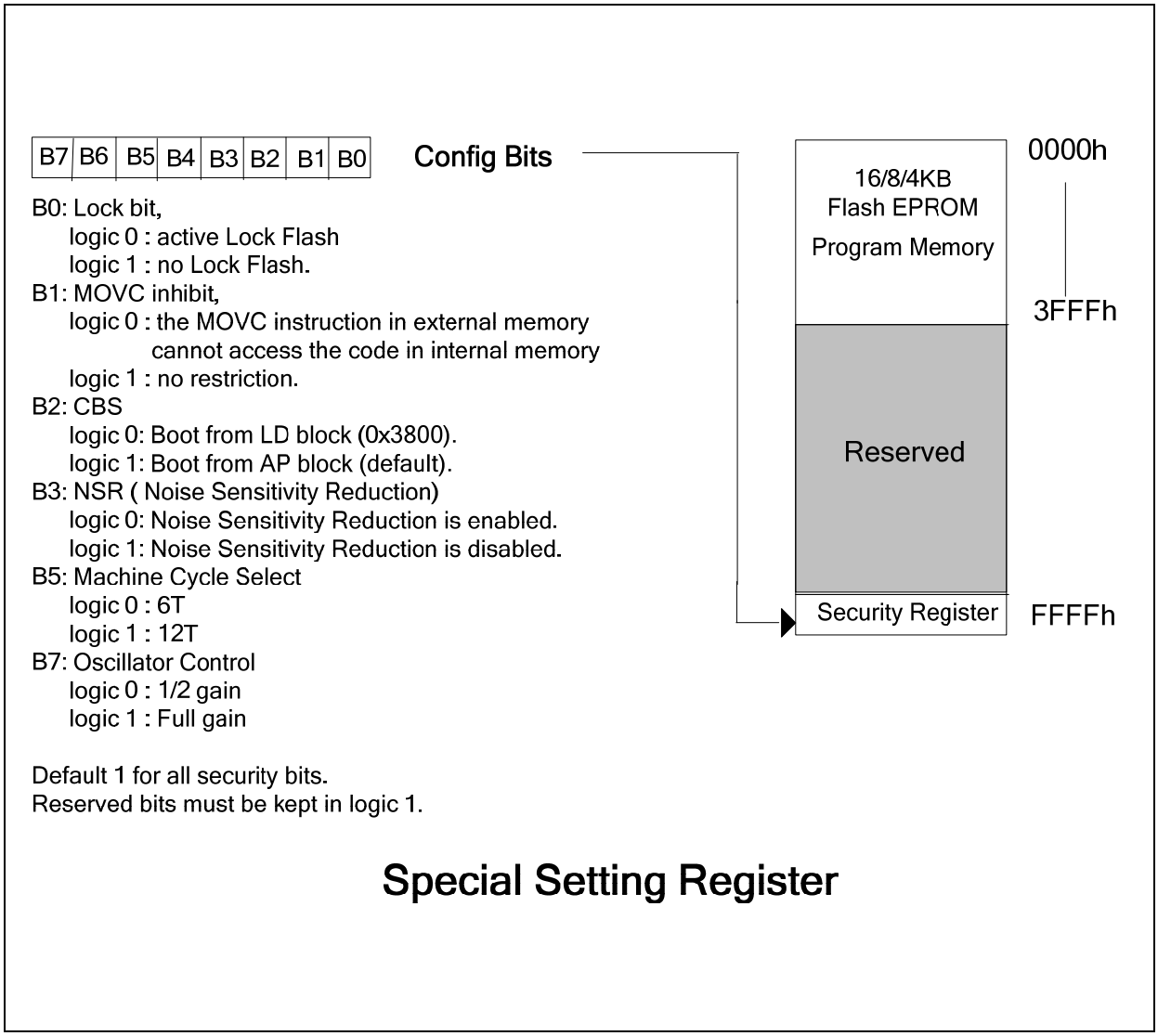
Reception is enabled only if REN is high. The serial port actually starts the receiving of serial data, with the detection of a falling edge on the RxD pin. The 1-to-0 detector continuously monitors the RxD line, sampling it at the rate of 16 times the selected baud rate. When a falling edge is detected, the divide by 16 counters is immediately reset. This helps to align the bit boundaries with the rollovers of the divide by 16 counters.


PGM
End of Programming ?
YES
NO
Read_Compay_ID
Read_Device_ID
Read_VT
Read_Dist
Part 2:
Procedure of Updating
the 2KB APROM
Setting Timer and enable Timer
interrupt for wake-up .
(50us for program operation)
Is currently in the
F02K BOOT Mode ?
NO
YES
Get the parameters of new code
(Address and data bytes)
through I/O ports, UART or
other interfaces.
Software reset CPU and
re-boot from the 2KB
APROM.
MOV CHPCON,#81h
Setting control registers for
programming:
MOV SFRAH,#ADDRESS_H
MOV SFRAL,#ADDRESS_L
MOV SFRFD,#DATA
MOV SFRCN,#21H
Hardware Reset
to re-boot from
new 2 KB APROM.
(S/W reset is
invalid in F02K BOOT
Mode)
END
Executing new code
from address
00H in the 2KB APROM.
Ease 14K AP programming:
MOV ERPAGE,#01
MOV SFRCN,#22H

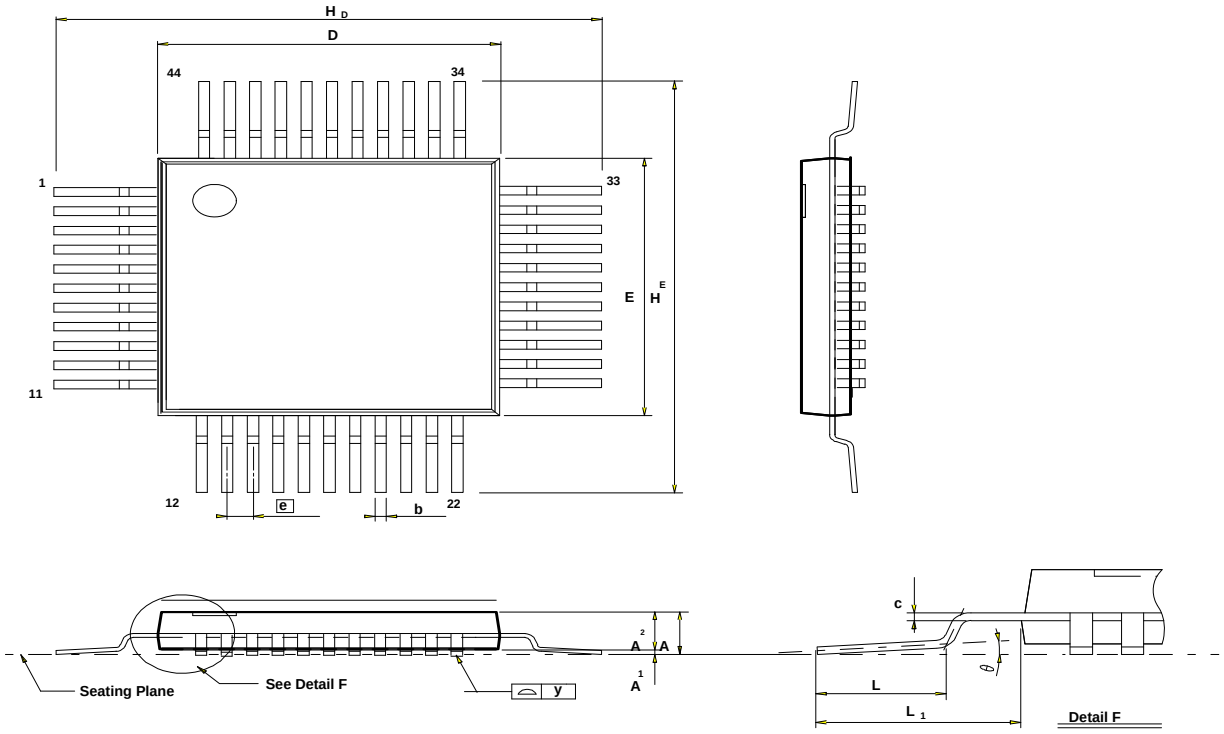
## 19 CONFIG BITS

During the on-chip Flash EPROM operation mode, the Flash EPROM can be programmed and verified repeatedly. Until the code inside the Flash EPROM is confirmed OK, the code can be protected. The protection of Flash EPROM and those operations on it are described below.

The W78I054D/W78I052D/W78I051D has a Special Setting Register, the config Bits, which can not be accessed in normal mode. The Security register can only be accessed from the Flash EPROM operation mode. Those bits of the Security Registers can not be changed once they have been programmed from high to low. They can only be reset through erase-all operation. The Security Register is addressed in the Flash EPROM operation mode by address #0FFFFh.



## 22.3 44-pin PQFP




| Symbol | Dimension in inch | | | Dimension in mm | | |
|---|---|---|---|---|---|---|
| | Min | Nom | Max | Min | Nom | Max |
| A | — | — | - | — | — | - |
| $A_1$ | 0.002 | 0.01 | 0.02 | 0.05 | 0.25 | 0.5 |
| $A_2$ | 0.075 | 0.081 | 0.087 | 1.90 | 2.05 | 2.20 |
| b | 0.01 | 0.014 | 0.018 | 0.25 | 0.35 | 0.45 |
| c | 0.004 | 0.006 | 0.010 | 0.10 | 0.15 | 0.25 |
| D | 0.390 | 0.394 | 0.398 | 9.9 | 10.00 | 10.1 |
| E | 0.390 | 0.394 | 0.398 | 9.9 | 10.00 | 10.1 |
| e | —— | .0315 | —— | —— | 0.80 | —— |
| $H_D$ | 0.510 | 0.520 | 0.530 | 12.95 | 13.20 | 13.45 |
| $H_E$ | 0.510 | 0.520 | 0.530 | 12.95 | 13.20 | 13.45 |
| L | 0.025 | 0.031 | 0.037 | 0.65 | 0.8 | 0.95 |
| $L_1$ | —— | 0.063 | —— | —— | 1.60 | —— |
| y | —— | —— | 0.004 | —— | —— | 0.10 |
| θ | 0 ° | —— | 10 ° | 0 ° | —— | 10 ° |

**Application Note: In-system Programming Software Examples**

This application note illustrates the in-system programmability of the microcontroller. In this example, microcontroller will boot from 2K LDROM bank enter in-system programming mode for programming the contents of APROM, this sample to Erase APROM, Erase Verify APROM, Read one byte for APROM, Write one byte for APROM, Read CID/DID. .

**EXAMPLE: Base on Keil C51 Compiler**

```
$nomod51
#include <reg52.h>

EAPAGE                DATA     0BEh
CHPCON                DATA     0BFh
SFRAL                 DATA     0C4h
SFRAH                 DATA     0C5h
SFRFD                 DATA     0C6h
SFRCN                 DATA     0C7h

;CPU Clock = 12MHz/12T mode
READ_TIME             EQU      1
PROGRAM_TIME          EQU      50
ERASE_TIME            EQU      5000

;For W78E(I)054D
APROM_END_ADDRESS     EQU      03800h
;For W78E(I)052D
;APROM_END_ADDRESS     EQU      02000h
;For W78E(I)051D
;APROM_END_ADDRESS     EQU      01000h

FLASH_STANDBY         EQU      00111111B
READ_CID              EQU      00001011B
READ_DID              EQU      00001100B
ERASE_ROM             EQU      00100010B
ERASE_VERIFY          EQU      00001001B
PROGRAM_ROM           EQU      00100001B
PROGRAM_VERIFY_ROM    EQU      00001010B
READ_ROM              EQU      00000000B

    ORG       03800h
```

```
    mov     SFRCN,#PROGRAM_ROM
    mov     DPTR,#0000h
    mov     A,#055h
wr_lp:
    mov     TH0,#HIGH(65536-PROGRAM_TIME)
    mov     TL0,#LOW (65536-PROGRAM_TIME)
    mov     SFRFD,A
    mov     SFRAL,DPL
    mov     SFRAH,DPH
    setb    TR0
    mov     CHPCON,#00000011b
    clr     TF0
    clr     TR0
    cpl     A
    inc     DPTR
    mov     R0,DPL
    cjne    R0,#LOW (APROM_END_ADDRESS),wr_lp
    mov     R1,DPH
    cjne    R1,#HIGH(APROM_END_ADDRESS),wr_lp
    ret
;*********************************************************************************
;*Program Verify APROM BANK, read APROM 55h,AAh,55h,AAh........
;*********************************************************************************
Program_Verify_APROM:
    mov     SFRCN,#PROGRAM_VERIFY_ROM
    mov     DPTR,#0000h
    mov     B,#055h
rd_lp:
    mov     TH0,#HIGH(65536-READ_TIME)
    mov     TL0,#LOW (65536-READ_TIME)
    mov     SFRAL,DPL
    mov     SFRAH,DPH
    setb    TR0
    mov     CHPCON,#00000011b
    clr     TF0
    clr     TR0
    mov     A,SFRFD
    cjne    A,B,Program_Fail
    mov     A,B
```

## Important Notice

**Nuvoton products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Further more, Nuvoton products are not intended for applications wherein failure of Nuvoton products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.**

**Nuvoton customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nuvoton for any damages resulting from such improper use or sales.**