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	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 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-BQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/w78e516dfg

1 GENERAL DESCRIPTION

The W78E516D/W78E058D series is an 8-bit microcontroller which has an in-system programmable Flash EPROM for on-chip firmware updating.

The instruction sets of the W78E516D/W78E058D are fully compatible with the standard 8052. The W78E516D/W78E058D series contains a 64K/32K bytes of main Flash EPROM and a 4K bytes of auxiliary Flash EPROM which allows the contents of the 64K/32K bytes main Flash EPROM to be up-dated by the loader program located in the 4K bytes Flash EPROM; a 256 bytes of SRAM; 256 bytes of AUXRAM; four 8-bit bi-directional and bit-addressable I/O ports; an additional 4-bit port P4; three 16-bit timer/counters; a serial port. These peripherals are supported by an 8 sources 2-level interrupt capability. To facilitate programming and verification, the Flash EPROM inside the W78E516D/W78E058D series allows the program memory to be programmed and read electronically. Once the code is confirmed, the user can protect the code for security.

The W78E516D/W78E058D series microcontroller has two power reduction modes, idle mode and power-down mode, both of which are software selectable. The idle mode turns off the processor clock but allows for continued peripheral operation. The power-down mode stops the crystal oscillator for minimum power consumption. The external clock can be stopped at any time and in any state without affecting the processor.

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 2.4 to 5.5V
- Temperature grade is (-40°C~85°C)
- 64K/32K bytes of in-system programmable FLASH EPROM for Application Program (APROM)
- 4K bytes of auxiliary FLASH EPROM for Loader Program (LDROM)
- Low standby current at full supply voltage
- 512 bytes of on-chip RAM. (include 256 bytes of AUX-RAM, software selectable)
- 64K bytes program memory address space and 64K bytes data memory address space
- One 4-bit multipurpose programmable port, additional $\overline{INT2}$ / $\overline{INT3}$
- Support Watch Dog Timer
- Three 16-bit timer/counters
- One full duplex serial port
- 8-sources, 2-level interrupt capability
- Software Reset
- Built-in power management with idle mode and power down mode
- Code protection
- Packages:
  - Lead Free (RoHS) DIP 40: W78E516DDG
  - Lead Free (RoHS) PLCC 44: W78E516DPG
  - Lead Free (RoHS) PQFP 44: W78E516DFG
  - Lead Free (RoHS) LQFP 48: W78E516DLG
  - Lead Free (RoHS) DIP 40: W78E058DDG
  - Lead Free (RoHS) PLCC 44: W78E058DPG
  - Lead Free (RoHS) PQFP 44: W78E058DFG
  - Lead Free (RoHS) LQFP 48: W78E058DLG

as an internal timer, depending on the setting of bit C/T2 in T2CON. Timer 2 has three operating modes: capture, auto-reload, and baud rate generator. The clock speed at capture or auto-reload mode is the same as that of Timers 0 and 1.

7.4.1 Clock

The W78E516D/W78E058D series are designed to be used with either a crystal oscillator or an external clock. Internally, the clock is divided by two before it is used by default. This makes the W78E516D/W78E058D series relatively insensitive to duty cycle variations in the clock.

7.5 Interrupts

The Interrupt structure in the W78E516D/W78E058D series is slightly different from that of the standard 8052. Due to the presence of additional features and peripherals, the number of interrupt sources and vectors has been increased. The W78E516D/W78E058D series provides 8 interrupt resources with two priority level, including four external interrupt sources, three timer interrupts, serial I/O interrupts.

7.6 Data Pointers

The data pointer of W78E516D/W78E058D series is same as standard 8052 that have one 16-bit Data Pointer (DPTR).

7.7 Architecture

The W78E516D/W78E058D 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 W78E516D/W78E058D 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 W78E516D/W78E058D 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.

Mnemonic: TL1 Address: 8Bh

BIT	NAME	FUNCTION
7-0	TL1.[7:0]	Timer 1 LSB.

Timer 0 MSB

Bit:	7	6	5	4	3	2	1	0
	TH0.7	TH0.6	TH0.5	TH0.4	TH0.3	TH0.2	TH0.1	TH0.0

Mnemonic: TH0 Address: 8Ch

BIT	NAME	FUNCTION
7-0	TH0.[7:0]	Timer 0 MSB.

Timer 1 MSB

Bit:	7	6	5	4	3	2	1	0
	TH1.7	TH1.6	TH1.5	TH1.4	TH1.3	TH1.2	TH1.1	TH1.0

Mnemonic: TH1 Address: 8Dh

BIT	NAME	FUNCTION
7-0	TH1.[7:0]	Timer 1 MSB.

AUXR

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

Mnemonic: AUXR Address: 8Eh

BIT	NAME	FUNCTION
0	ALE_OFF	1: Disable ALE output 0: Enable ALE output

Watchdog Timer Control Register

Bit:	7	6	5	4	3	2	1	0
	ENW	CLRW	WIDL	-	-	PS2	PS1	PS0

Mnemonic: WDTC Address: 8FH

BIT	NAME	FUNCTION
7	ENW	Enable watch-dog if set.
6	CLRW	Clear watch-dog timer and Pre-scalar if set. This flag will be cleared automatically.
5	WIDL	If this bit is set, watch-dog is enabled under IDLE mode. If cleared, watch-dog is

BIT	NAME	FUNCTION
7-0	P42AL.[7:0]	The Base address register for comparator of P4.2. P42AL contains the low-order byte of address.

P4.2 Base Address High Byte Register

Bit:	7	6	5	4	3	2	1	0
	P42AH.7	P42AH.6	P42AH.5	P42AH.4	P42AH.3	P42AH.2	P42AH.1	P42AH.0

Mnemonic:P42AH Address: ADh

BIT	NAME	FUNCTION
7-0	P42AH.[7:0]	The Base address register for comparator of P4.2. P42AH contains the High-order byte of address.

Port 2 Expanded Control

Bit:	7	6	5	4	3	2	1	0
	P43CSIN	P42CSIN	P41CSIN	P40CSIN	-	-	-	-

Mnemonic: P2ECON Address:AEh

BIT	NAME	FUNCTION
7	P43CSINV	The active polarity of P4.3 when pin P4.3 is defined as read and/or write strobe signal. 1 : P4.3 is active high when pin P4.3 is defined as read and/or write strobe signal. 0 : P4.3 is active low when pin P4.3 is defined as read and/or write strobe signal.
6	P42CSINV	The similarity definition as P43SINV.
5	P41CSINV	The similarity definition as P43SINV.
4	P40CSINV	The similarity definition as P43SINV.

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

P4.3 Base Address Low Byte Register

Bit:	7	6	5	4	3	2	1	0
	P43AL.7	P43AL.6	P43AL.5	P43AL.4	P43AL.3	P43AL.2	P43AL.1	P43AL.0

Mnemonic: P43AL Address: B4h

BIT	NAME	FUNCTION
7-0	P43AL.[7:0]	The Base address register for comparator of P4.3. P43AL contains the low-order byte of address.

P4.3 Base Address High Byte Register

Bit:	7	6	5	4	3	2	1	0
	P43AH.7	P43AH.6	P43AH.5	P43AH.4	P43AH.3	P43AH.2	P43AH.1	P43AH.0

Mnemonic: P43AH Address: B5h

BIT	NAME	FUNCTION
7-0	P43AH.[7:0]	The Base address register for comparator of P4.3. P43AH contains the High-order byte of address.

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: To set interrupt priority of Timer 2 is higher priority level.
4	PS	1: To set interrupt priority of Serial port 0 is higher priority level.
3	PT1	1: To set interrupt priority of Timer 1 is higher priority level.
2	PX1	1: To set interrupt priority of External interrupt 1 is higher priority level.
1	PT0	1: To set interrupt priority of Timer 0 is higher priority level.
0	PX0	1: To set interrupt priority of External interrupt 0 is higher priority level.

Chip Control

Bit:	7	6	5	4	3	2	1	0
	SWRESET	-	-	ENAUXRAM	0	0	FBOOTSL	FPRGEN

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

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
	-	WFWIN	OEN	CEN	CTRL3	CTRL2	CTRL1	CTRL0

SFRCN Address: C7h

BIT	NAME	FUNCTION
6	WFWIN	On-chip FLASH EPROM bank select for in-system programming. 0: 64K bytes FLASH EPROM bank is selected as destination for re-programming. 1: 4K bytes FLASH EPROM bank is selected as destination for re-programming.
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	CTRL<3:0>	WFWIN	OEN	CEN	SFRAH,SFRAL	SFRFD
Erase APROM	0010	0	1	0	X	X
Program APROM	0001	0	1	0	Address in	Data in

10 INSTRUCTION

The W78E516D/W78E058D series execute all the instructions of the standard 8052 family. The operations of these instructions, as well as their effects on flag and status bits, are exactly the same.

Op-code	HEX Code	Bytes	W78E516D/W78E058D series Clock cycles
NOP	00	1	12
ADD A, R0	28	1	12
ADD A, R1	29	1	12
ADD A, R2	2A	1	12
ADD A, R3	2B	1	12
ADD A, R4	2C	1	12
ADD A, R5	2D	1	12
ADD A, R6	2E	1	12
ADD A, R7	2F	1	12
ADD A, @R0	26	1	12
ADD A, @R1	27	1	12
ADD A, direct	25	2	12
ADD A, #data	24	2	12
ADDC A, R0	38	1	12
ADDC A, R1	39	1	12
ADDC A, R2	3A	1	12
ADDC A, R3	3B	1	12
ADDC A, R4	3C	1	12
ADDC A, R5	3D	1	12
ADDC A, R6	3E	1	12
ADDC A, R7	3F	1	12
ADDC A, @R0	36	1	12
ADDC A, @R1	37	1	12
ADDC A, direct	35	2	12
ADDC A, #data	34	2	12
SUBB A, R0	98	1	12
SUBB A, R1	99	1	12
SUBB A, R2	9A	1	12
SUBB A, R3	9B	1	12
SUBB A, R4	9C	1	12

Op-code	HEX Code	Bytes	W78E516D/W78E058D series Clock cycles
DA A	D4	1	12
ANL A, R0	58	1	12
ANL A, R1	59	1	12
ANL A, R2	5A	1	12
ANL A, R3	5B	1	12
ANL A, R4	5C	1	12
ANL A, R5	5D	1	12
ANL A, R6	5E	1	12
ANL A, R7	5F	1	12
ANL A, @R0	56	1	12
ANL A, @R1	57	1	12
ANL A, direct	55	2	12
ANL A, #data	54	2	12
ANL direct, A	52	2	12
ANL direct, #data	53	3	24
ORL A, R0	48	1	12
ORL A, R1	49	1	12
ORL A, R2	4A	1	12
ORL A, R3	4B	1	12
ORL A, R4	4C	1	12
ORL A, R5	4D	1	12
ORL A, R6	4E	1	12
ORL A, R7	4F	1	12
ORL A, @R0	46	1	12
ORL A, @R1	47	1	12
ORL A, direct	45	2	12
ORL A, #data	44	2	12
ORL direct, A	42	2	12
ORL direct, #data	43	3	24
XRL A, R0	68	1	12
XRL A, R1	69	1	12
XRL A, R2	6A	1	12
XRL A, R3	6B	1	12

Op-code	HEX Code	Bytes	W78E516D/W78E058D series Clock cycles
CPL bit	B2	2	12
ANL C, bit	82	2	24
ANL C, /bit	B0	2	24
ORL C, bit	72	2	24
ORL C, /bit	A0	2	24
MOV C, bit	A2	2	12
MOV bit, C	92	2	24
ACALL addr11	71, 91, B1, 11, 31, 51, D1, F1	2	24
LCALL addr16	12	3	24
RET	22	1	24
RETI	32	1	24
AJMP ADDR11	01, 21, 41, 61, 81, A1, C1, E1	2	24
LJMP addr16	02	3	24
JMP @A+DPTR	73	1	24
SJMP rel	80	2	24
JZ rel	60	2	24
JNZ rel	70	2	24
JC rel	40	2	24
JNC rel	50	2	24
JB bit, rel	20	3	24
JNB bit, rel	30	3	24
JBC bit, rel	10	3	24
CJNE A, direct, rel	B5	3	24
CJNE A, #data, rel	B4	3	24
CJNE @R0, #data, rel	B6	3	24
CJNE @R1, #data, rel	B7	3	24
CJNE R0, #data, rel	B8	3	24
CJNE R1, #data, rel	B9	3	24
CJNE R2, #data, rel	BA	3	24
CJNE R3, #data, rel	BB	3	24
CJNE R4, #data, rel	BC	3	24
CJNE R5, #data, rel	BD	3	24

12 POWER MANAGEMENT

The W78E516D/W78E058D 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 W78E516D/W78E058D will exit the Power Down mode with a reset or by an external interrupt pin enabled as level detects. 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 W78E516D/W78E058D 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.

The interrupt flags are sampled every machine cycle. In the same machine cycle, the sampled interrupts are polled and their priority is resolved. If certain conditions are met then the hardware will execute an internally generated LCALL instruction which will vector the process to the appropriate interrupt vector address. The conditions for generating the LCALL are;

1. An interrupt of equal or higher priority is not currently being serviced.
2. The current polling cycle is the last machine cycle of the instruction currently being executed.
3. The current instruction does not involve a write to IE, IP, XICON registers and is not a RETI.

If any of these conditions are not met, then the LCALL will not be generated. The polling cycle is repeated every machine cycle, with the interrupts sampled in the same machine cycle. If an interrupt flag is active in one cycle but not responded to, and is not active when the above conditions are met, the denied interrupt will not be serviced. This means that active interrupts are not remembered; every polling cycle is new.

The processor responds to a valid interrupt by executing an LCALL instruction to the appropriate service routine. This may or may not clear the flag which caused the interrupt. In case of Timer interrupts, the TF0 or TF1 flags are cleared by hardware whenever the processor vectors to the appropriate timer service routine. In case of external interrupt, /INT0 and /INT1, the flags are cleared only if they are edge triggered. In case of Serial interrupts, the flags are not cleared by hardware. In the case of Timer 2 interrupt, the flags are not cleared by hardware. The hardware LCALL behaves exactly like the software LCALL instruction. This instruction saves the Program Counter contents onto the Stack, but does not save the Program Status Word PSW. The PC is reloaded with the vector address of that interrupt which caused the LCALL. These address of vector for the different sources are as shown on below table. The vector table is not evenly spaced; this is to accommodate future expansions to the device family.

Execution continues from the vectored address till an RETI instruction is executed. On execution of the RETI instruction the processor pops the Stack and loads the PC with the contents at the top of the stack. The user must take care that the status of the stack is restored to what is was after the hardware LCALL, if the execution is to return to the interrupted program. The processor does not notice anything if the stack contents are modified and will proceed with execution from the address put back into PC. Note that a RET instruction would perform exactly the same process as a RETI instruction, but it would not inform the Interrupt Controller that the interrupt service routine is completed, and would leave the controller still thinking that the service routine is underway.

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in registers IE. The IE register also contains a global disable bit, EA, which disables all interrupts at once.

Each interrupt source can be individually programmed to one of 2 priority levels by setting or clearing bits in the IP registers. An interrupt service routine in progress can be interrupted by a higher priority interrupt, but not by another interrupt of the same or lower priority. The highest priority interrupt service cannot be interrupted by any other interrupt source. So, if two requests of different priority levels are received simultaneously, the request of higher priority level is serviced.

If requests of the same priority level are received simultaneously, an internal polling sequence determines which request is serviced. This is called the arbitration ranking. Note that the arbitration ranking is only used to resolve simultaneous requests of the same priority level.

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 W78E516D/W78E058D 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.

grammable 9th bit (TB8) and a stop bit (1). The 9th bit received is put into RB8. The baud rate is programmable to 1/32 or 1/64 of the oscillator frequency, which is determined by the SMOD bit in PCON SFR. 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 the 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 counters, and not directly to the write to SBUF signal. After all 9 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 11th 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. The 16 states of the counter effectively divide the bit time into 16 slices. The bit detection is done on a best of three basis. The bit detector samples the RxD pin, at the 8th, 9th and 10th counter states. By using a majority 2 of 3 voting system, the bit value is selected. This is done to improve the noise rejection feature of the serial port.

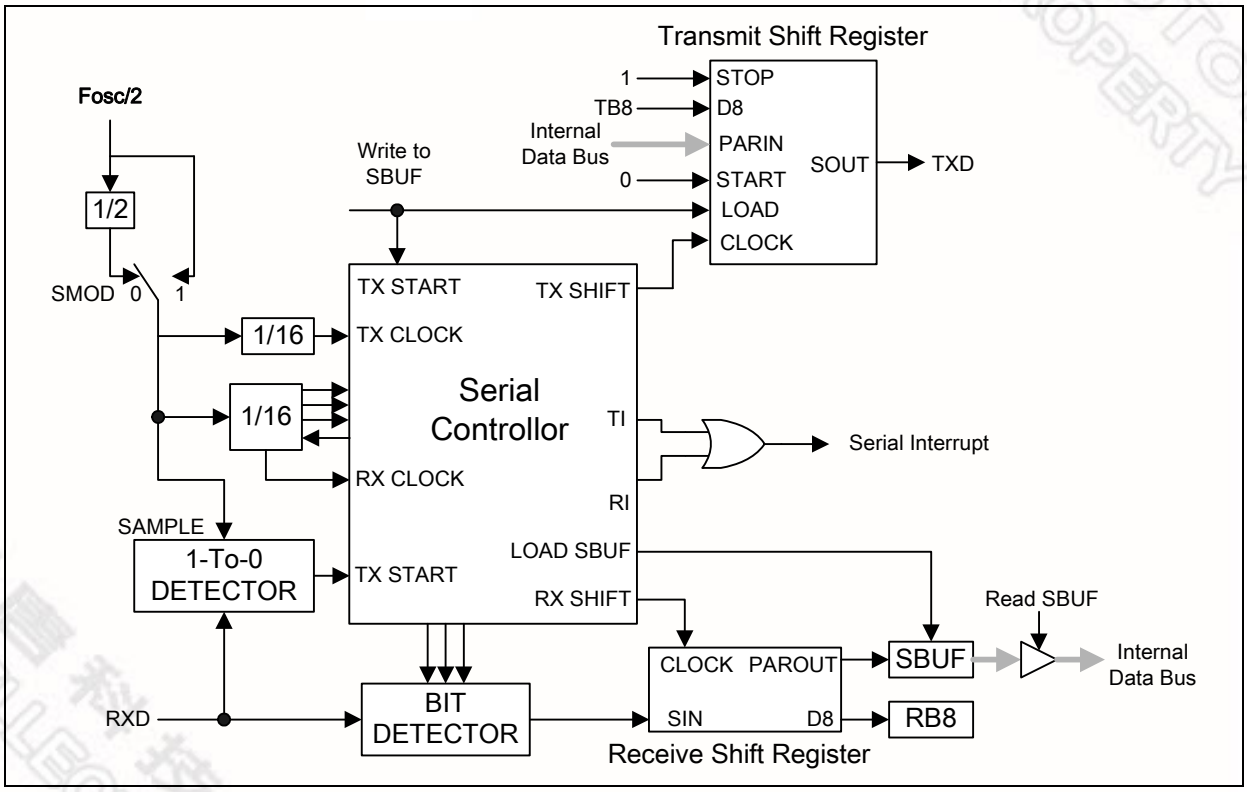

Figure 16- 3 Serial port mode 2

If the first bit detected after the falling edge of RxD pin, is not 0, then this indicates an invalid start bit, and the reception is immediately aborted. The serial port again looks for a falling edge in the RxD line. If a valid start bit is detected, then the rest of the bits are also detected and shifted into the SBUF. After shifting in 9 data bits, there is one more shift to do, after which the SBUF and RB8 are loaded and RI is set. However certain conditions must be met before the loading and setting of RI can be done.

1. RI must be 0 and

gramming function. Then you can use this F04KBOOT mode to force the W78E516D/W78E058D jump to LDROM and run on chip programming procedure. When you design your system, you can connect the pins P26, P27 to switches or jumpers. For example in a CD ROM system, you can connect the P26 and P27 to PLAY and EJECT buttons on the panel. When the APROM program is fail to execute the normal application program. User can press both two buttons at the same time and then switch on the power of the personal computer to force the W78E516D/W78E058D to enter the F04KBOOT mode. After power on of personal computer, you can release both PLAY and EJECT button.

NOTE2: In application system design, user must take care the P2, P3, ALE, /EA and /PSEN pin value at reset to avoid W78E516D/W78E058D entering the programming mode or F04KBOOT mode in normal operation.

18 ISP(IN-SYSTEM PROGRAMMING)

ISP is the ability of programmable MCU to be programmed while F/W code in AP-ROM or LD-ROM (ISP work voltage 3.3-5.5V).

The W78E058D/516D equips one 32K byte of main ROM bank for application program (called APROM) and one 4K byte of auxiliary ROM bank for loader program (called LDROM). In the normal operation, the microcontroller executes the code in the APROM. If the content of APROM needs to be modified, the W78E058D/516D allows user to activate the In-System Programming (ISP) mode by setting the CHPCON register. **The CHPCON is read-only by default, software must write two specific values 87H, then 59H sequentially to the CHPENR register to enable the CHPCON write attribute. Writing CHPENR register with the values except 87H and 59H will close CHPCON register write attribute.** The W78E058D/516D achieves all in-system programming operations including enter/exit ISP Mode, program, erase, read ... etc, during device in the idle mode. Setting the bit CHPCON.0 the device will enter in-system programming mode after a wake-up from idle mode. Because device needs proper time to complete the ISP operations before awaken from idle mode, software may use timer interrupt to control the duration for device wake-up from idle mode. To perform ISP operation for revising contents of APROM, software located at APROM setting the CHPCON register then enter idle mode, after awaken from idle mode the device executes the corresponding interrupt service routine in LDROM. Because the device will clear the program counter while switching from APROM to LDROM, the first execution of RETI instruction in interrupt service routine will jump to 00H at LDROM area. The device offers a software reset for switching back to APROM while the content of APROM has been updated completely. **Setting CHPCON register bit 0, 1 and 7 to logic-1 will result a software reset to reset the CPU**. The software reset serves as a external reset. This in-system programming feature makes the job easy and efficient in which the application needs to update firmware frequently. In some applications, the in-system programming feature make it possible to easily update the system firmware without opening the chassis.

SFRAH, SFRAL: The objective address of on-chip ROM in the in-system programming mode. SFRAH contains the high-order byte of address, SFRAL contains the low-order byte of address.

SFRFD: The programming data for on-chip ROM in programming mode.

SFRCN: The control byte of on-chip ROM programming mode.

SFRCN (C7)

BIT	NAME	FUNCTION
7	-	Reserve.
6	WFWIN	On-chip ROM bank select for in-system programming. 0: 32K/64K bytes ROM bank is selected as destination for re-programming. 1: 4K bytes ROM bank is selected as destination for re-programming.
5	OEN	ROM output enable.
4	CEN	ROM chip enable.
3, 2, 1, 0	CTRL [3:0]	The flash control signals

20 TYPICAL APPLICATION CIRCUITS

External Program Memory and Crystal

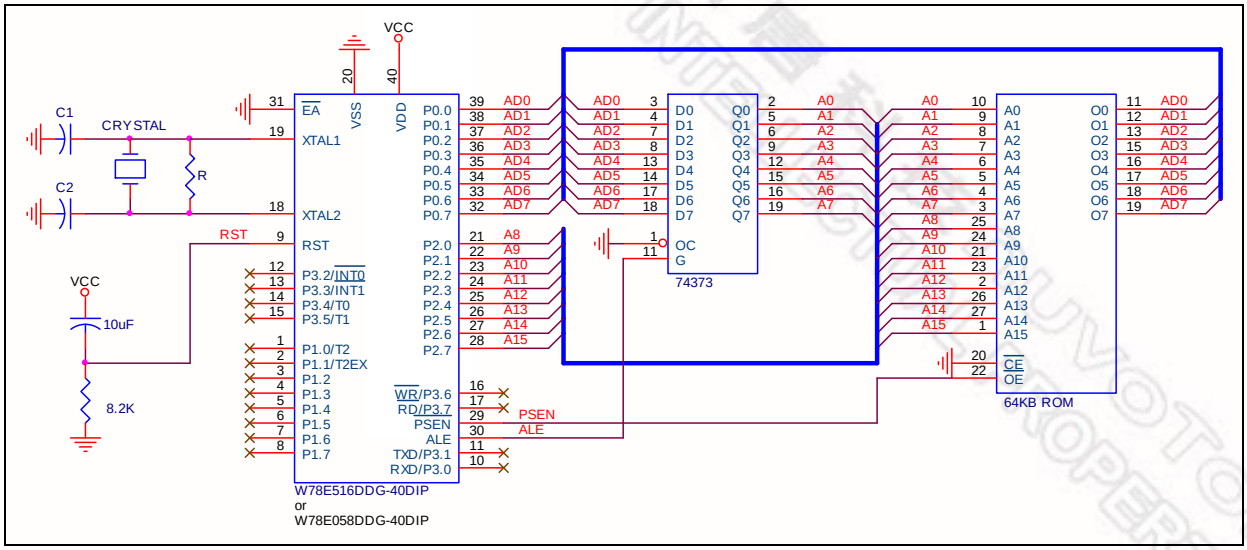

Figure A

Expanded External Data Memory and Oscillator

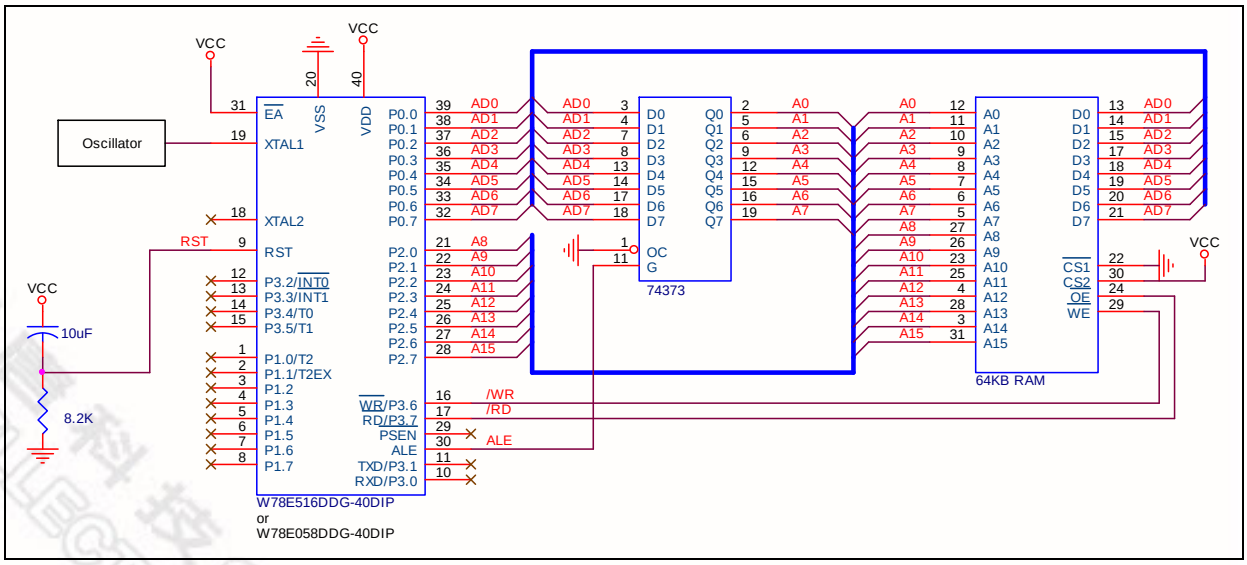

Figure B

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 APROM bank and waiting for a key to enter in-system programming mode for re-programming the contents of APROM. While entering in-system programming mode, microcontroller executes the loader program in 4KB LDROM bank. The loader program erases the APROM then reads the new code data from external SRAM buffer (or through other interfaces) to update the APROM.

EXAMPLE 1:

```
;*****************************************************************************************************************
;* Example of APROM program: Program will scan the P1.0. if P1.0 = 0, enters in-system
;* programming mode for updating the contents of APROM code else executes the current ROM code.
;* XTAL = 40 MHz
;*****************************************************************************************************************

    .chip 8052
    .RAMCHK OFF
    .symbols

    CHPCON          EQU         BFH
    CHPENR          EQU         F6H
    SFRAL           EQU         C4H
    SFRAH           EQU         C5H
    SFRFD           EQU         C6H
    SFRCN           EQU         C7H

            ORG         0H
            LJMP        100H                        ;JUMP TO MAIN PROGRAM
  ;*************************************************************************
  ;*  TIMER0 SERVICE VECTOR ORG=000BH
  ;*************************************************************************
            ORG         00BH
            CLR         TR0                         ;TR0=0,STOP TIMER0
            MOV         TL0,R6
            MOV         TH0,R7
            RETI
  ;*************************************************************************
  ;*   APROM MAIN PROGRAM
  ;*************************************************************************
            ORG         100H

MAIN_:

           MOV          A,P1                        ;SCAN P1.0
           ANL          A,#01H
           CJNE         A,#01H,PROGRAM_             ;IF P1.0=0, ENTER IN-SYSTEM
                                                    ;PROGRAMMING MODE
           JMP          NORMAL_MODE

PROGRAM_:
            MOV         CHPENR,#87H                 ; CHPENR=87H, CHPCON
                                                    ; REGISTER WRTE ENABLE
            MOV         CHPENR,#59H                  ; CHPENR=59H, CHPCON
                                                    ; REGISTER WRITE ENABLE
            MOV         CHPCON,#03H                  ;CHPCON=03H, ENTER IN-SYSTEM
                                                    ; PROGRAMMING MODE
```

```
;*********************************************************************
            ORG 100H

MAIN_4K:

            MOV         CHPENR,#87H             ;CHPENR=87H, CHPCON WRITE ENABLE.
            MOV         CHPENR,#59H             ;CHPENR=59H, CHPCON WRITE ENABLE.
            MOV         7FH,#01H                ;SET F04KBOOT MODE FLAG.

            MOV         A,CHPCON
            ANL         A,#01H
            CJNE        A,#00H,UPDATE_          ;CHECK CHPCON BIT 0
            MOV         7FH,#00H                ;FLAG=0, NOT IN THE F04KBOOT MODE.

            MOV         CHPCON,#01H             ;CHPCON=01H, ENABLE IN-SYSTEM PROGRAMMING.
            MOV         CHPENR,#00H             ;DISABLE CHPCON WRITE ATTRIBUTE

            MOV         TCON,#00H               ;TCON=00H ,TR=0 TIMER0 STOP
            MOV         TMOD,#01H               ;TMOD=01H ,SET TIMER0 A 16BIT TIMER
            MOV         IP,#00H                 ;IP=00H
            MOV         IE,#82H                 ;IE=82H,TIMER0 INTERRUPT ENABLED
            MOV         R6,#FEH
            MOV         R7,#FFH
            MOV         TL0,R6
            MOV         TH0,R7
            MOV         TCON,#10H               ;TCON=10H,TR0=1,GO
            MOV         PCON,#01H               ;ENTER IDLE MODE


UPDATE_:

            MOV         CHPENR,#00H             ;DISABLE CHPCON WRITE-ATTRIBUTE
            MOV         TCON,#00H               ;TCON=00H ,TR=0 TIM0 STOP
            MOV         IP,#00H                 ;IP=00H
            MOV         IE,#82H                 ;IE=82H,TIMER0 INTERRUPT ENABLED
            MOV         TMOD,#01H               ;TMOD=01H ,MODE1

            MOV         R6,#3CH                 ;SET WAKE-UP TIME FOR ERASE OPERATION,
                                                ;ABOUT 15ms.   DEPENDING ON USER'S
                                                ;SYSTEM CLOCK RATE.
            MOV         R7,#B0H
            MOV         TL0,R6
            MOV         TH0,R7

ERASE_P_4K:
            MOV         SFRCN,#22H              ;SFRCN(C7H)=22H  ERASE
            MOV         TCON,#10H               ;TCON=10H,TR0=1,GO
            MOV         PCON,#01H               ;ENTER IDLE MODE( FOR ERASE OPERATION)

;*********************************************************************
;* BLANK CHECK
;*********************************************************************

            MOV         SFRCN,#0H               ;READ APROM MODE
            MOV         SFRAH,#0H               ;START ADDRESS = 0H
            MOV         SFRAL,#0H
            MOV         R6,#FBH                 ;SET TIMER FOR READ OPERATION, ABOUT 1.5us.
            MOV         R7,#FFH
```

```
                MOV         TL0,R6
                MOV         TH0,R7

BLANK_CHECK_LOOP:

                SETB        TR0                     ;ENABLE TIMER 0
                MOV         PCON,#01H               ;ENTER IDLE MODE
                MOV         A,SFRFD                 ;READ ONE BYTE

                CJNE        A,#FFH,BLANK_CHECK_ERROR
                INC         SFRAL                   ;NEXT ADDRESS
                MOV         A,SFRAL
                JNZ         BLANK_CHECK_LOOP
                INC         SFRAH
                MOV         A,SFRAH
                CJNE        A,#C0H,BLANK_CHECK_LOOP   ;END ADDRESS=BFFFH
               JMP          PROGRAM_ROM

BLANK_CHECK_ERROR:

                MOV         P1,#F0H
                MOV         P3,#F0H
                JMP         $

;*******************************************************************************
;* RE-PROGRAMMING APROM BANK
;*******************************************************************************
PROGRAM_ROM:

                MOV         DPTR,#0H                ;THE ADDRESS OF NEW ROM CODE
                MOV         R2,#00H                 ;TARGET LOW BYTE ADDRESS
                MOV         R1,#00H                 ;TARGET HIGH BYTE ADDRESS
                MOV         DPTR,#0H                ;EXTERNAL SRAM BUFFER ADDRESS
                MOV         SFRAH,R1                ;SFRAH, TARGET  HIGH ADDRESS
                 MOV        SFRCN,#21H              ;SFRCN(C7H)=21  (PROGRAM )
                MOV         R6,#0CH                 ;SET TIMER FOR PROGRAMMING, ABOUT 150us.
                MOV         R7,#FEH
                MOV         TL0,R6
                MOV         TH0,R7
PROG_D_:
                MOV         SFRAL,R2                ;SFRAL(C4H)= LOW BYTE ADDRESS
                MOVX        A,@DPTR                 ;READ DATA FROM EXTERNAL SRAM BUFFER
                MOV         SFRFD,A                 ;SFRFD(C6H)=DATA IN
                MOV         TCON,#10H               ;TCON=10H,TR0=1,GO
                MOV         PCON,#01H               ;ENTER IDLE MODE( PRORGAMMING)
                INC         DPTR
                INC         R2
                CJNE        R2,#0H,PROG_D_
                INC         R1
                MOV         SFRAH,R1
                CJNE        R1,#C0H,PROG_D_
;*******************************************************************************
; * VERIFY APROM BANK
;*******************************************************************************
                MOV         R4,#03H                 ;ERROR COUNTER
                MOV         R6,#FBH                 ;SET TIMER FOR READ VERIFY, ABOUT 1.5us.
                MOV         R7,#FFH
```