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	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	40MHz
Connectivity	EBI/EMI, Serial Port
Peripherals	POR, WDT
Number of I/O	32
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/w77e532a40dl

5. FUNCTIONAL DESCRIPTION

The W77E532A is 8052 pin compatible and instruction set compatible. It includes the resources of the standard 8052 such as four 8-bit I/O Ports, three 16-bit timer/counters, full duplex serial port and interrupt sources.

The W77E532A features a faster running and better performance 8-bit CPU with a redesigned core processor without wasted clock and memory cycles. it improves the performance not just by running at high frequency but also by reducing the machine cycle duration from the standard 8052 period of twelve clocks to four clock cycles for the majority of instructions. This improves performance by an average of 1.5 to 3 times. The W77E532A also provides dual Data Pointers (DPTRs) to speed up block data memory transfers. It can also adjust the duration of the MOVX instruction (access to off-chip data memory) between two machine cycles and nine machine cycles. This flexibility allows the W77E532A to work efficiently with both fast and slow RAMs and peripheral devices. In addition, the W77E532A contains on-chip 1KB MOVX SRAM, the address of which is between 0000H and 03FFH. It only can be accessed by MOVX instruction; this on-chip SRAM is optional under software control.

The W77E532A is an 8052 compatible device that gives the user the features of the original 8052 device, but with improved speed and power consumption characteristics. It has the same instruction set as the 8051 family, with one addition: DEC DPTR (op-code A5H, the DPTR is decreased by 1). While the original 8051 family was designed to operate at 12 clock periods per machine cycle, the W77E532A operates at a much reduced clock rate of only 4 clock periods per machine cycle. This naturally speeds up the execution of instructions. Consequently, the W77E532A can run at a higher speed as compared to the original 8052, even if the same crystal is used. Since the W77E532A is a fully static CMOS design, it can also be operated at a lower crystal clock, giving the same throughput in terms of instruction execution, yet reducing the power consumption.

The 4 clocks per machine cycle feature in the W77E532A is responsible for a three-fold increase in execution speed. The W77E532A has all the standard features of the 8052, and has a few extra peripherals and features as well.

I/O Ports

The W77E532A has four 8-bit ports and one extra 4-bit port. Port 0 can be used as an Address/Data bus when external program is running or external memory/device is accessed by MOVC or MOVX instruction. In these cases, it has strong pull-ups and pull-downs, and does not need any external pull-ups. Otherwise it can be used as a general I/O port with open-drain circuit. Port 2 is used chiefly as the upper 8-bits of the Address bus when port 0 is used as an address/data bus. It also has strong pull-ups and pull-downs when it serves as an address bus. Port 1 and 3 act as I/O ports with alternate functions. Port 4 is only available on 44-pin PLCC/QFP package type. It serves as a general purpose I/O port as Port 1 and Port 3. The P4.0 has an alternate function $CP/\overline{RL2}$ which is the wait state control signal. When wait state control signal is enabled, P4.0 is input only.

Serial I/O

The W77E532A has two enhanced serial ports that are functionally similar to the serial port of the original 8052 family. However the serial ports on the W77E532A can operate in different modes in order to obtain timing similarity as well. **Note that the serial port 0 can use Timer 1 or 2 as baud rate generator, but the serial port 1 can only use Timer 1 as baud rate generator**. The serial ports have the enhanced features of Automatic Address recognition and Frame Error detection.

Special Function Registers

The W77E532A uses Special Function Registers (SFRs) to control and monitor peripherals and their Modes.

The SFRs reside in the register locations 80-FFh and are accessed by direct addressing only. Some of the SFRs are bit addressable. This is very useful in cases where one wishes to modify a particular bit without changing the others. The SFRs that are bit addressable are those whose addresses end in 0 or 8. The W77E532A contains all the SFRs present in the standard 8052. However, some additional SFRs have been added. In some cases unused bits in the original 8052 have been given new functions. The list of SFRs is as follows. The table is condensed with eight locations per row. Empty locations indicate that there are no registers at these addresses. When a bit or register is not implemented, it will read high.

Table 1. Special Function Register Location Table

F8	EIP							
F0	B							
E8	EIE							
E0	ACC							
D8	WDCON							
D0	PSW							
C8	T2CON	T2MOD	RCAP2L	RCAP2H	TL2	TH2		
C0	SCON1	SBUF1	WSCON		PMR	STATUS		TA
B8	IP	SADEN	SADEN1					
B0	P3							
A8	IE	SADDR	SADDR1	ROMCON	SFRAL	SFRAH	SFDFD	SFRCN
A0	P2		P4CSIN			P4		
98	SCON0	SBUF	P42AL	P42AH	P43AL	P43AH		CHPCON
90	P1	EXIF	P4CONA	P4CONB	P40AL	P40AH	P41AL	P41AH
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	
80	P0	SP	DPL	DPH	DPL1	DPH1	DPS	PCON

Note: The SFRs in the column with dark borders are bit-addressable.

Timer 0 LSB

Bit:	7	6	5	4	3	2	1	0
	TL0.7	TL0.6	TL0.5	TL0.4	TL0.3	TL0.2	TL0.1	TL0.0

Mnemonic: TL0 Address: 8Ah

TL0.7–0: Timer 0 LSB

Timer 1 LSB

Bit:	7	6	5	4	3	2	1	0
	TL1.7	TL1.6	TL1.5	TL1.4	TL1.3	TL1.2	TL1.1	TL1.0

Mnemonic: TL1 Address: 8Bh

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

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

TH1.7–0: Timer 1 MSB

Clock Control

Bit:	7	6	5	4	3	2	1	0
	WD1	WD0	T2M	T1M	T0M	MD2	MD1	MD0

Mnemonic: CKCON Address: 8Eh

WD1–0: Watchdog timer mode select bits: These bits determine the time-out period for the watchdog timer. In all four time-out options the reset time-out is 512 clocks more than the interrupt time-out period.

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

B.7-0: The B register is the standard 8052 register that serves as a second accumulator.

Extended Interrupt Priority

Bit:	7	6	5	4	3	2	1	0
	-	-	-	PWDI	PX5	PX4	PX3	PX2

Mnemonic: EIP Address: F8h

EIP.7-5: Reserved bits.

PWDI: Watchdog timer interrupt priority.

PX5: External Interrupt 5 Priority. 0 = Low priority, 1 = High priority.

PX4: External Interrupt 4 Priority. 0 = Low priority, 1 = High priority.

PX3: External Interrupt 3 Priority. 0 = Low priority, 1 = High priority.

PX2: External Interrupt 2 Priority. 0 = Low priority, 1 = High priority.

8. INSTRUCTION TIMING

The instruction timing for the W77E532A is an important aspect, especially for those users who wish to use software instructions to generate timing delays. Also, it provides the user with an insight into the timing differences between the W77E532A and the standard 8032. In the W77E532A each machine cycle is four clock periods long. Each clock period is designated a state. Thus each machine cycle is made up of four states, C1, C2 C3 and C4, in that order. Due to the reduced time for each instruction execution, both the clock edges are used for internal timing. Hence it is important that the duty cycle of the clock be as close to 50% as possible to avoid timing conflicts. As mentioned earlier, the W77E532A does one op-code fetch per machine cycle. Therefore, in most of the instructions, the number of machine cycles needed to execute the instruction is equal to the number of bytes in the instruction. Of the 256 available op-codes, 128 of them are single cycle instructions. Thus more than half of all op-codes in the W77E532A are executed in just four clock periods. Most of the two-cycle instructions are those that have two byte instruction codes. However there are some instructions that have only one byte instructions, yet they are two cycle instructions. One instruction which is of importance is the MOVX instruction. In the standard 8032, the MOVX instruction is always two machine cycles long. However in the W77E532A, the user has a facility to stretch the duration of this instruction from 2 machine cycles to 9 machine cycles. The $\overline{RD}$ and $\overline{WR}$ strobe lines are also proportionately elongated. This gives the user flexibility in accessing both fast and slow peripherals without the use of external circuitry and with minimum software overhead. The rest of the instructions are either three, four or five machine cycle instructions. Note that in the W77E532A, based on the number of machine cycles, there are five different types, while in the standard 8032 there are only three. However, in the W77E532A each machine cycle is made of only 4 clock periods compared to the 12 clock periods for the standard 8032. Therefore, even though the number of categories has increased, each instruction is at least 1.5 to 3 times faster than the standard 8032 in terms of clock periods.

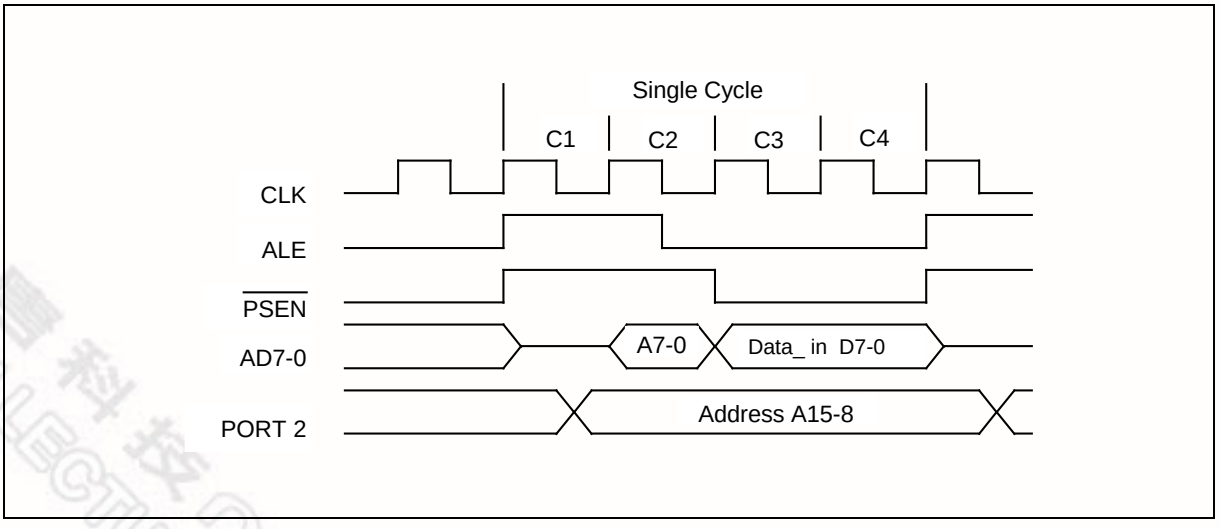


Figure 3. Single Cycle Instruction Timing

MOVX Instruction

The W77E532A, like the standard 8032, uses the MOVX instruction to access external Data Memory. This Data Memory includes both off-chip memory as well as memory mapped peripherals. While the results of the MOVX instruction are the same as in the standard 8032, the operation and the timing of the strobe signals have been modified in order to give the user much greater flexibility.

The MOVX instruction is of two types, the MOVX @Ri and MOVX @DPTR. In the MOVX @Ri, the address of the external data comes from two sources. The lower 8-bits of the address are stored in the Ri register of the selected working register bank. The upper 8-bits of the address come from the port 2 SFR. In the MOVX @DPTR type, the full 16-bit address is supplied by the Data Pointer.

Since the W77E532A has two Data Pointers, DPTR and DPTR1, the user has to select between the two by setting or clearing the DPS bit. The Data Pointer Select bit (DPS) is the LSB of the DPS SFR, which exists at location 86h. No other bits in this SFR have any effect, and they are set to 0. When DPS is 0, then DPTR is selected, and when set to 1, DPTR1 is selected. The user can switch between DPTR and DPTR1 by toggling the DPS bit. The quickest way to do this is by the INC instruction. The advantage of having two Data Pointers is most obvious while performing block move operations. The accompanying code shows how the use of two separate Data Pointers speeds up the execution time for code performing the same task.

Block Move with single Data Pointer:

```
; SH and SL are the high and low bytes of Source Address
; DH and DL are the high and low bytes of Destination Address
; CNT is the number of bytes to be moved
                                                    Machine cycles of W77E532A
                                                                              #
MOV   R2, #CNT          ; Load R2 with the count value                        2
MOV   R3, #SL           ; Save low byte of Source Address in R3               2
MOV   R4, #SH           ; Save high byte of Source address in R4              2
MOV   R5, #DL           ; Save low byte of Destination Address in R5          2
MOV   R6, #DH           ; Save high byte of Destination address in R6         2
LOOP:
MOV   DPL, R3           ; Load DPL with low byte of Source address            2
MOV   DPH, R4           ; Load DPH with high byte of Source address           2
MOVXA, @DPTR            ; Get byte from Source to Accumulator                 2
INC     DPTR            ; Increment Source Address to next byte               2
MOV   R3, DPL           ; Save low byte of Source address in R3               2
MOV   R4, DPH           ; Save high byte of Source Address in R4              2
MOV   DPL, R5           ; Load low byte of Destination Address in DPL         2
MOV   DPH, R6           ; Load high byte of Destination Address in DPH        2
MOVX@DPTR, A            ; Write data to destination                           2
INC     DPTR            ; Increment Destination Address                       2
MOV   DPL, R5           ; Save low byte of new destination address in R5      2
MOV   DPH, R6           ; Save high byte of new destination address in R6     2
DJNZ  R2, LOOP          ; Decrement count and do LOOP again if count <> 0     2
```

the Stretch value is set at 1, giving a MOVX instruction of 3 machine cycles. If desired by the user the stretch value can be set to 0 to give the fastest MOVX instruction of only 2 machine cycles.

Table 4. Data Memory Cycle Stretch Values

M2	M1	M0	MACHINE CYCLES	$\overline{RD}$ OR $\overline{WR}$ STROBE WIDTH IN CLOCKS	$\overline{RD}$ OR $\overline{WR}$ STROBE WIDTH @ 25 MHZ	$\overline{RD}$ OR $\overline{WR}$ STROBE WIDTH @ 40 MHZ
0	0	0	2	2	80 nS	50 nS
0	0	1	3 (default)	4	160 nS	100 nS
0	1	0	4	8	320 nS	200 nS
0	1	1	5	12	480 nS	300 nS
1	0	0	6	16	640 nS	400 nS
1	0	1	7	20	800 nS	500 nS
1	1	0	8	24	960 nS	600 nS
1	1	1	9	28	1120 nS	700 nS

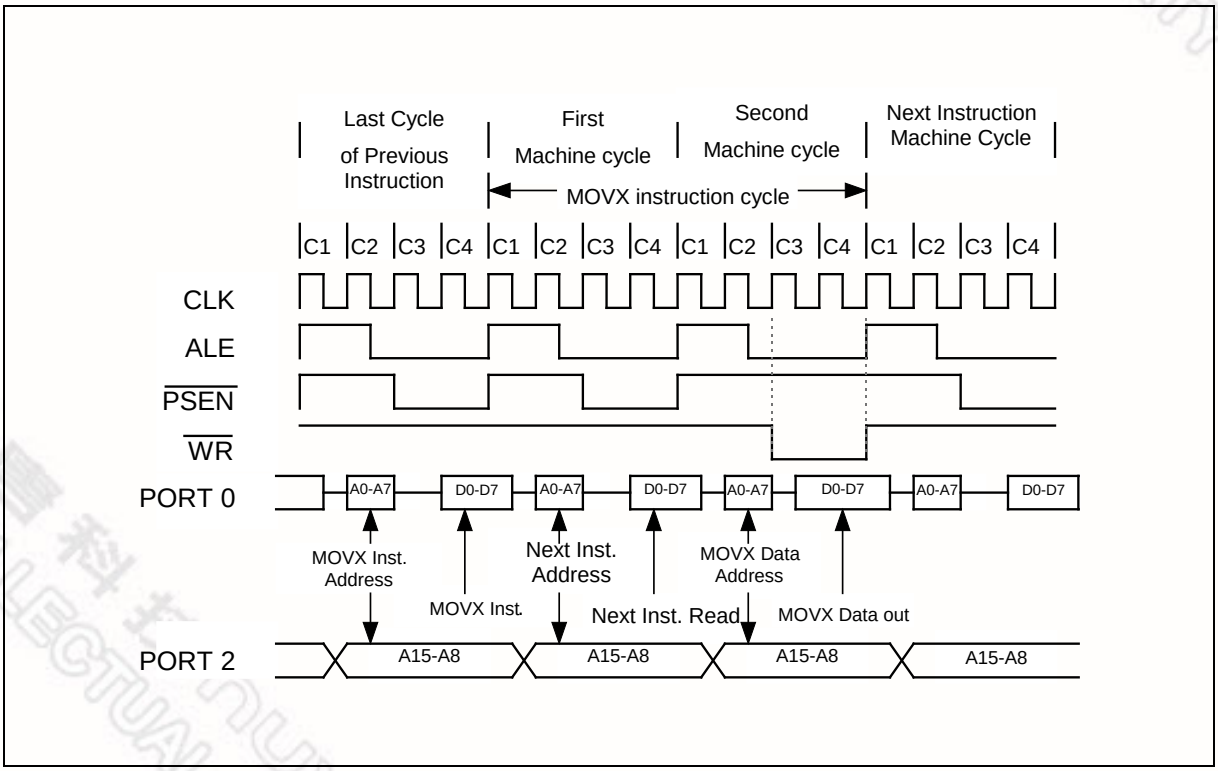


Figure 8. Data Memory Write with Stretch Value = 0

Table 6. SFR Reset Value

SFR NAME	RESET VALUE	SFR NAME	RESET VALUE
P0	11111111b	IE	00000000b
SP	00000111b	SADDR	00000000b
DPL	00000000b	P3	11111111b
DPH	00000000b	IP	x0000000b
DPL1	00000000b	SADEN	00000000b
DPH1	00000000b	T2CON	00000000b
DPS	00000000b	T2MOD	00000x00b
PCON	00xx0000b	RCAP2L	00000000b
TCON	00000000b	RCAP2H	00000000b
TMOD	00000000b	TL2	00000000b
TL0	00000000b	TH2	00000000b
TL1	00000000b	TA	11111111b
TH0	00000000b	PSW	00000000b
TH1	00000000b	WDCON	0x0x0xx0b
CKCON	00000001b	ACC	00000000b
P1	11111111b	EIE	xxx00000b
P4CONA	00000000b	P4CONB	00000000b
P40AL	00000000b	P40AH	00000000b
P41AL	00000000b	P41AH	00000000b
P42AL	00000000b	P42AH	00000000b
P43Al	00000000b	P43AH	00000000b
CHPCON	00000000b	P4CSIN	00000000b
ROMCON	00000111b	SFRAL	00000000b
SFRAH	00000000b	SFRFD	00000000b
SFRCN	00111111b	B	00000000b
SCON	00000000b	EIP	xxx00000b
SBUF	xxxxxxxxb	PC	00000000b
P2	11111111b	SADEN1	00000000b
SADDR1	00000000b	SBUF1	xxxxxxxxb
SCON1	00000000b	PMR	010xx0x0b
WSCON	00000000b	STATUS	000x0000b
EXIF	0000xxx0b		
P4	xxxx1111b		

Table 7. Priority structure of interrupts

SOURCE	FLAG	PRIORITY LEVEL
External Interrupt 0	IE0	1(highest)
Timer 0 Overflow	TF0	2
External Interrupt 1	IE1	3
Timer 1 Overflow	TF1	4
Serial Port	RI + TI	5
Timer 2 Overflow	TF2 + EXF2	6
Serial Port 1	RI_1 + TI_1	7
External Interrupt 2	IE2	8
External Interrupt 3	IE3	9
External Interrupt 4	IE4	10
External Interrupt 5	IE5	11
Watchdog Timer	WDIF	12 (lowest)

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 IP, IE, EIP or EIE 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 Watchdog timer interrupt flag WDIF has to be cleared by software. 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 vector address for the different sources are as follows

11. PROGRAMMABLE TIMERS/COUNTERS

The W77E532A has three 16-bit programmable timer/counters and one programmable Watchdog timer. The Watchdog timer is operationally quite different from the other two timers.

Timer/Counters 0 & 1

The W77E532A has two 16-bit Timer/Counters. Each of these Timer/Counters has two 8 bit registers which form the 16 bit counting register. For Timer/Counter 0 they are TH0, the upper 8 bits register, and TL0, the lower 8 bit register. Similarly Timer/Counter 1 has two 8 bit registers, TH1 and TL1. The two can be configured to operate either as timers, counting machine cycles or as counters counting external inputs.

When configured as a “Timer”, the timer counts clock cycles. The timer clock can be programmed to be thought of as 1/12 of the system clock or 1/4 of the system clock. In the “Counter” mode, the register is incremented on the falling edge of the external input pin, T0 in case of Timer 0, and T1 for Timer 1. The T0 and T1 inputs are sampled in every machine cycle at C4. If the sampled value is high in one machine cycle and low in the next, then a valid high to low transition on the pin is recognized and the count register is incremented. Since it takes two machine cycles to recognize a negative transition on the pin, the maximum rate at which counting will take place is 1/24 of the master clock frequency. In either the “Timer” or “Counter” mode, the count register will be updated at C3. Therefore, in the “Timer” mode, the recognized negative transition on pin T0 and T1 can cause the count register value to be updated only in the machine cycle following the one in which the negative edge was detected.

The “Timer” or “Counter” function is selected by the “$C/\overline{T}$” bit in the TMOD Special Function Register. Each Timer/Counter has one selection bit for its own; bit 2 of TMOD selects the function for Timer/Counter 0 and bit 6 of TMOD selects the function for Timer/Counter 1. In addition each Timer/Counter can be set to operate in any one of four possible modes. The mode selection is done by bits M0 and M1 in the TMOD SFR.

Time-Base Selection

The W77E532A gives the user two modes of operation for the timer. The timers can be programmed to operate like the standard 8051 family, counting at the rate of 1/12 of the clock speed. This will ensure that timing loops on the W77E532A and the standard 8051 can be matched. This is the default mode of operation of the W77E532A timers. The user also has the option to count in the turbo mode, where the timers will increment at the rate of 1/4 clock speed. This will straight-away increase the counting speed three times. This selection is done by the T0M and T1M bits in CKCON SFR. A reset sets these bits to 0, and the timers then operate in the standard 8051 mode. The user should set these bits to 1 if the timers are to operate in turbo mode.

Mode 0

In Mode 0, the timer/counters act as a 8 bit counter with a 5 bit, divide by 32 pre-scale. In this mode we have a 13 bit timer/counter. The 13 bit counter consists of 8 bits of THx and 5 lower bits of TLx. The upper 3 bits of TLx are ignored.

The negative edge of the clock increments the count in the TLx register. When the fifth bit in TLx moves from 1 to 0, then the count in the THx register is incremented. When the count in THx moves from FFh to 00h, then the overflow flag TFx in TCON SFR is set. The counted input is enabled only if TRx is set and either GATE = 0 or $\overline{INTx}$ = 1. When $C/\overline{T}$ is set to 0, then it will count clock cycles, and if $C/\overline{T}$ is set to 1, then it will count 1 to 0 transitions on T0 (P3.4) for timer 0 and T1 (P3.5) for timer 1. When the 13 bit count reaches 1FFFh the next count will cause it to roll-over to 0000h. The timer overflow flag TFx of the relevant timer is set and if enabled an interrupts will occur. Note that when used as a timer, the time-base may be either clock cycles/12 or clock cycles/4 as selected by the bits TxM of the CKCON SFR.

Mode 3

Mode 3 has different operating methods for the two timer/counters. For timer/counter 1, mode 3 simply freezes the counter. Timer/Counter 0, however, configures TL0 and TH0 as two separate 8 bit count registers in this mode. The logic for this mode is shown in the figure. TL0 uses the Timer/Counter 0 control bits $C/\overline{T}$, GATE, TR0, $\overline{INT0}$ and TF0. The TL0 can be used to count clock cycles (clock/12 or clock/4) or 1-to-0 transitions on pin T0 as determined by C/T (TMOD.2). TH0 is forced as a clock cycle counter (clock/12 or clock/4) and takes over the use of TR1 and TF1 from Timer/Counter 1. Mode 3 is used in cases where an extra 8 bit timer is needed. With Timer 0 in Mode 3, Timer 1 can still be used in Modes 0, 1 and 2., but its flexibility is somewhat limited. While its basic functionality is maintained, it no longer has control over its overflow flag TF1 and the enable bit TR1. Timer 1 can still be used as a timer/counter and retains the use of GATE and INT1 pin. In this condition it can be turned on and off by switching it out of and into its own Mode 3. It can also be used as a baud rate generator for the serial port.

Timer/Counter 2

Timer/Counter 2 is a 16 bit up/down counter which is configured by the T2MOD register and controlled by the T2CON register. Timer/Counter 2 is equipped with a capture/reload capability. As with the Timer 0 and Timer 1 counters, there exists considerable flexibility in selecting and controlling the clock, and in defining the operating mode. The clock source for Timer/Counter 2 may be selected for either the external T2 pin (C/T2 = 1) or the crystal oscillator, which is divided by 12 or 4 (C/T2 = 0). The clock is then enabled when TR2 is a 1, and disabled when TR2 is a 0.

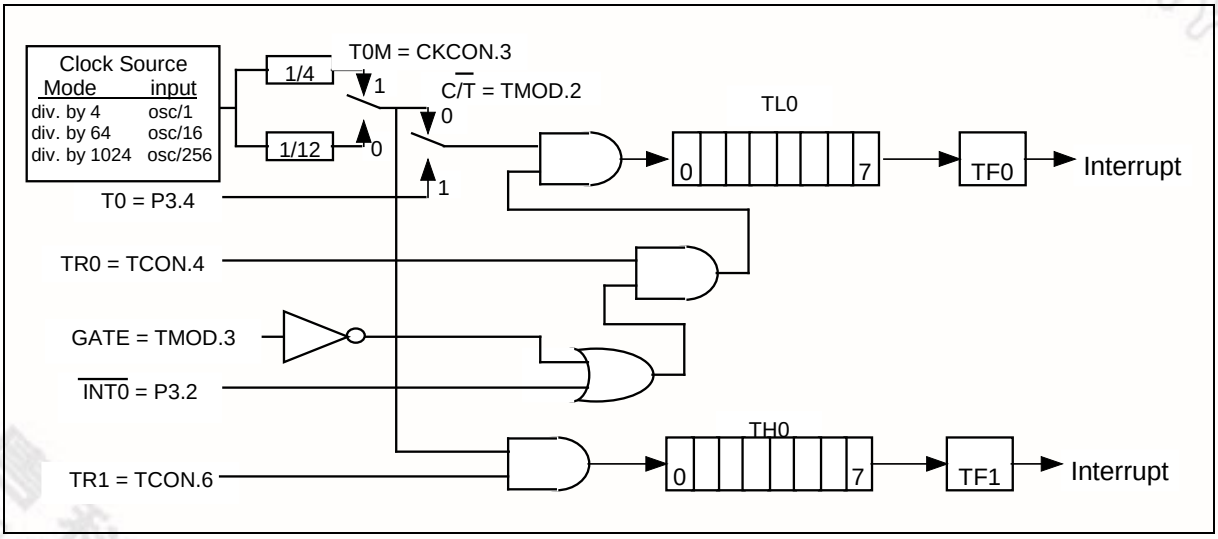


Figure 13. Timer/Counter 0 Mode 3

Capture Mode

The capture mode is enabled by setting the $CP/\overline{RL2}$ bit in the T2CON register to a 1. In the capture mode, Timer/Counter 2 serves as a 16 bit up counter. When the counter rolls over from FFFFh to 0000h, the TF2 bit is set, which will generate an interrupt request. If the EXEN2 bit is set, then a negative transition of T2EX pin will cause the value in the TL2 and TH2 register to be captured by the RCAP2L and RCAP2H registers. This action also causes the EXF2 bit in T2CON to be set, which will also generate an interrupt. Setting the T2CR bit (T2MOD.3), the W77E532A allows hardware to reset timer 2 automatically after the value of TL2 and TH2 have been captured.

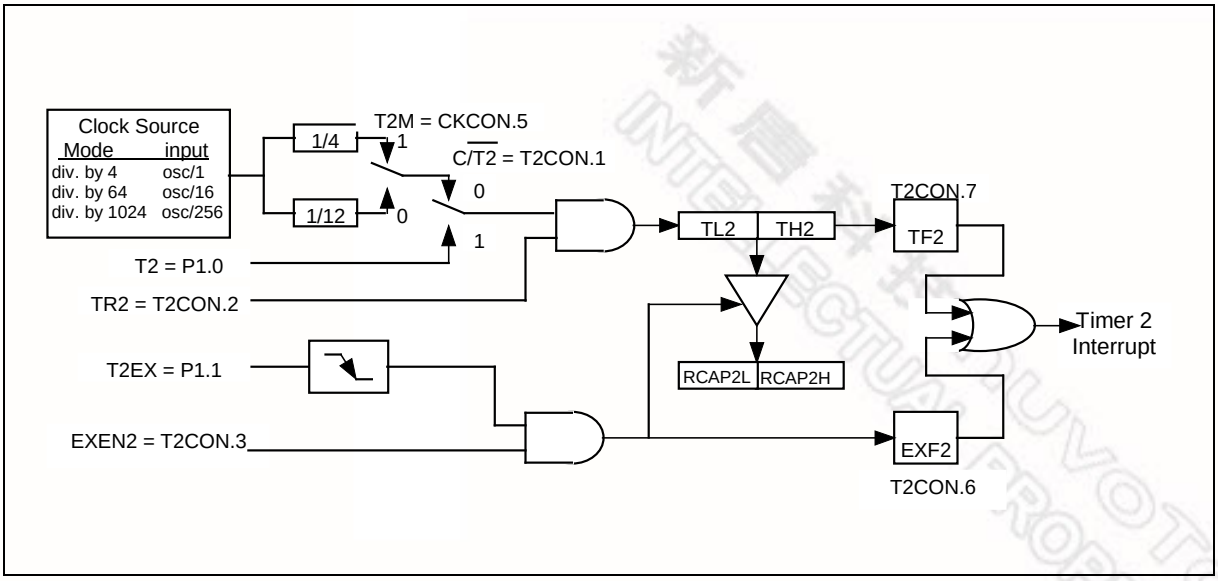


Figure 14. 16-Bit Capture Mode

1. RI must be 0 and
2. Either SM2 = 0, or the received stop bit = 1.

If these conditions are met, then the stop bit goes to RB8, the 8 data bits go into SBUF and RI is set. Otherwise the received frame may be lost. After the middle of the stop bit, the receiver goes back to looking for a 1-to-0 transition on the RxD pin.

Mode 3

This mode is similar to Mode 2 in all respects, except that the baud rate is programmable. The user must first initialize the Serial related SFR SCON before any communication can take place. This involves selection of the Mode and baud rate. The Timer 1 should also be initialized if modes 1 and 3 are used. In all four modes, transmission is started by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. This will generate a clock on the TxD pin and shift in 8 bits on the RxD pin. Reception is initiated in the other modes by the incoming start bit if REN = 1. The external device will start the communication by transmitting the start bit.

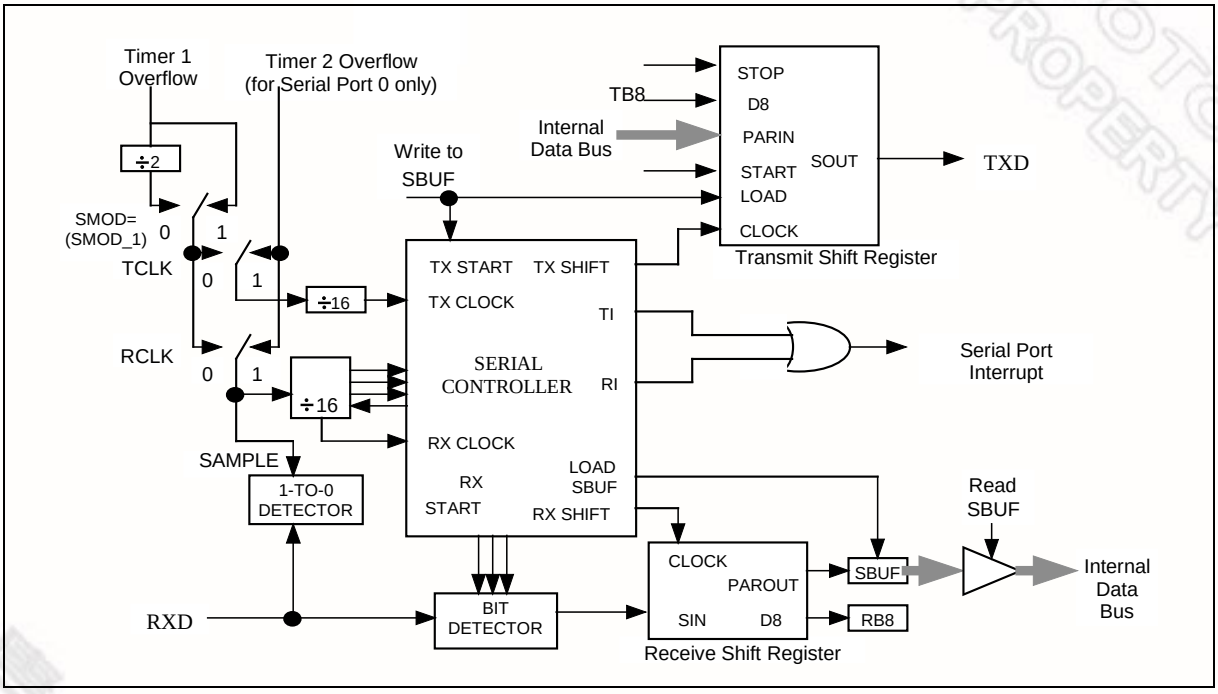


Figure 23. Serial Port Mode 3

Table 10. Serial Ports Modes

SM1	SM0	MODE	TYPE	BAUD CLOCK	FRAME SIZE	START BIT	STOP BIT	9TH BIT FUNCTION
0	0	0	Synch.	4 or 12 TCLKS	8 bits	No	No	None
0	1	1	Asynch.	Timer 1 or 2	10 bits	1	1	None
1	0	2	Asynch.	32 or 64 TCLKS	11 bits	1	1	0, 1
1	1	3	Asynch.	Timer 1 or 2	11 bits	1	1	0, 1

15. IN-SYSTEM PROGRAMMING

15.1 The Loader Program Locates at LDFLASH Memory

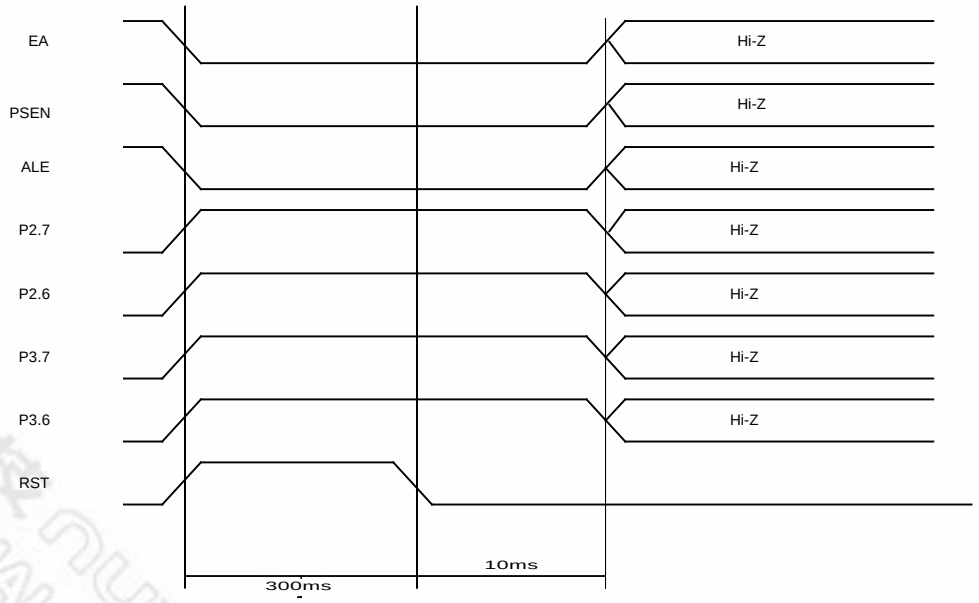
CPU is Free Run at APFLASH memory. CHPCON register had been set #03H value before CPU has entered idle state. CPU will switch to LDFLASH memory and execute a reset action. H/W reboot mode will switch to LDFLASH memory, too. Set SFRCN register where it locates at user's loader program to update APFLASH bank 0 or bank 1 memory. Set a SWRESET (CHPCON=#83H) to switch back APFLASH after CPU has updated APFLASH program. CPU will restart to run program from reset state.

15.2 The Loader Program Locates at APFLASH Memory

CPU is Free Run at APFLASH memory. CHPCON register had been set #01H value before CPU has entered idle state. Set SFRCN register to update LDFLASH or another bank of APFLASH program. CPU will continue to run user's APFLASH program after CPU has updated program. Please refer demonstrative code to understand other detail description.

16. H/W WRITER MODE

The Timing For Entering Flash EPROM Mode on the Programmer



18. ELECTRICAL CHARACTERISTICS

18.1 Absolute Maximum Ratings

SYMBOL	PARAMETER	CONDITION	RATING	UNIT
DC Power Supply	$V_{DD} - V_{SS}$	-0.3	+7.0	V
Input Voltage	V_{IN}	V_{SS} -0.3	V_{DD} +0.3	V
Operating Temperature	T_A	0	+70	°C
Storage Temperatute	Tst	-55	+150	°C

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

18.2 DC Characteristics

($V_{DD} - V_{SS}$ = 5V ±10%, T_A = 25°C, Fosc = 20 MHz, unless otherwise specified.)

PARAMETER	SYMBOL	SPECIFICATION			TEST CONDITIONS
		MIN.	MAX.	UNIT	
Operating Voltage	V_{DD}	4.5	5.5	V	
Operating Current	I_{DD}	-	30	mA	No load V_{DD} = RST = 5.5V
Idle Current	I_{IDLE}	-	24	mA	Idle mode V_{DD} = 5.5V
Power Down Current	I_{PWDN}	-	10	µA	Power-down mode V_{DD} = 5.5V
Input Current P1, P2, P3	I_{IN1}	-50	+10	µA	V_{DD} = 5.5V V_{IN} = 0V or V_{DD}
Input Current RST[*1]	I_{IN2}	-10	+120	µA	V_{DD} = 5.5V 0<V_{IN}<V_{DD}
Input Leakage Current P0, $\overline{EA}$	I_{LK}	-10	+10	µA	V_{DD} = 5.5V 0V<V_{IN}<V_{DD}
Logic 1 to 0 Transition Current P1, P2, P3	I_{TL}[*4]	-500	-200	µA	V_{DD} = 5.5V V_{IN} = 2.0V
Input Low Voltage P0, P1, P2, P3, $\overline{EA}$	V_{IL1}	0	0.8	V	V_{DD} = 4.5V
Input Low Voltage RST[*1]	V_{IL2}	0	0.8	V	V_{DD} = 4.5V
Input Low Voltage XTAL1[*3]	V_{IL3}	0	0.8	V	V_{DD} = 4.5V
Input High Voltage P0, P1, P2, P3, $\overline{EA}$	V_{IH1}	2.4	V_{DD} +0.2	V	V_{DD} = 5.5V
Input High Voltage RST	V_{IH2}	3.5	V_{DD} +0.2	V	V_{DD} = 5.5V
Input High Voltage XTAL1[*3]	V_{IH3}	3.5	V_{DD} +0.2	V	V_{DD} = 5.5V

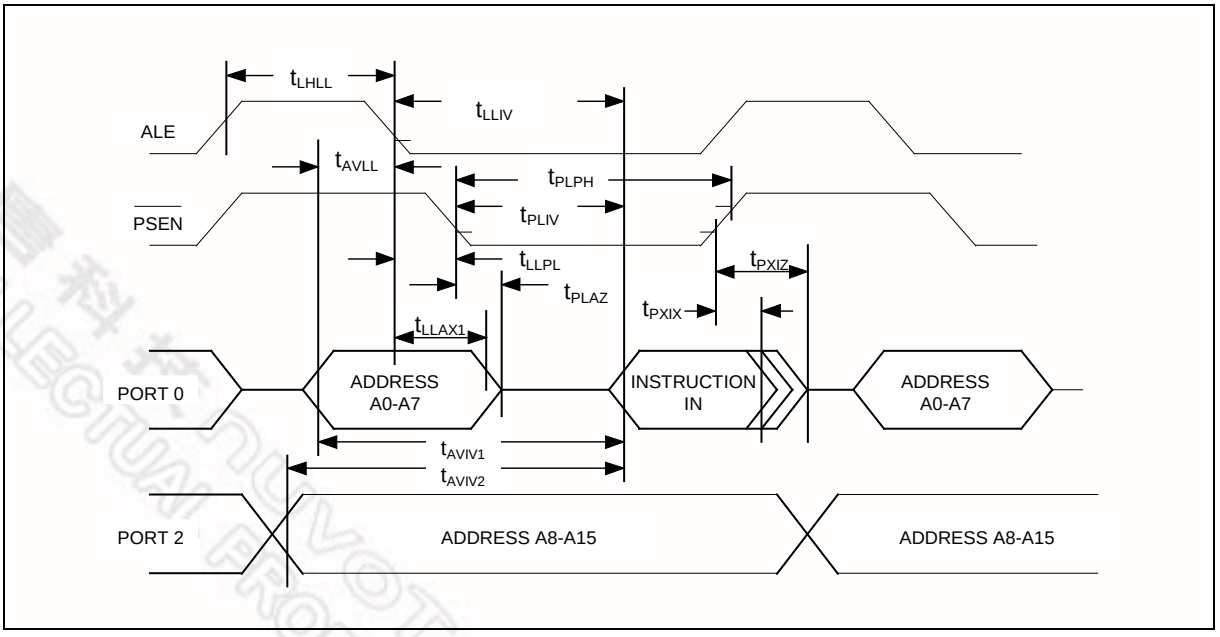
M2	M1	M0	MOVX CYCLES	T_{MCS}
0	0	0	2 machine cycles	0
0	0	1	3 machine cycles	4 t_{CLCL}
0	1	0	4 machine cycles	8 t_{CLCL}
0	1	1	5 machine cycles	12 t_{CLCL}
1	0	0	6 machine cycles	16 t_{CLCL}
1	0	1	7 machine cycles	20 t_{CLCL}
1	1	0	8 machine cycles	24 t_{CLCL}
1	1	1	9 machine cycles	28 t_{CLCL}

Explanation of Logics Symbols

In order to maintain compatibility with the original 8051 family, this device specifies the same parameter for each device, using the same symbols. The explanation of the symbols is as follows.

T	Time	A	Address
C	Clock	D	Input Data
H	Logic level high	L	Logic level low
I	Instruction	P	$\overline{PSEN}$
Q	Output Data	R	$\overline{RD}$ signal
V	Valid	W	$\overline{WR}$ signal
X	No longer a valid state	Z	Tri-state

18.3.3 Program Memory Read Cycle



19.2 Expanded External Data Memory and Oscillator

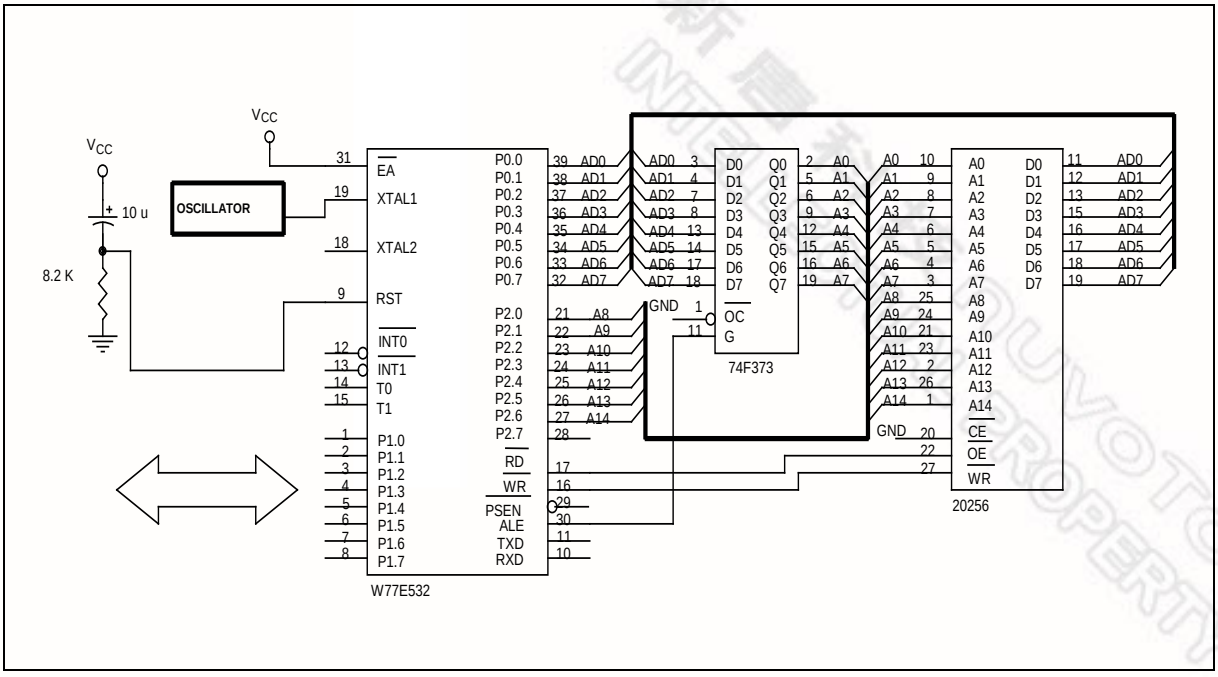


Figure B

```
PROGRAM_64:
      MOV  TA, #AAH              ; CHPCON register is written protect by TA register.
      MOV  TA, #55H
      MOV  CHPCON, #03H          ; CHPCON = 03H, ENTER IN-SYSTEM PROGRAMMING MODE
      MOV  SFRCN, #0H
        MOV  TCON, #00H          ; TR = 0 TIMER0 STOP
      MOV  IP, #00H              ; IP = 00H
      MOV  IE, #82H              ; TIMER0 INTERRUPT ENABLE FOR WAKE-UP FROM IDLE MODE
      MOV  R6, #F0H              ; TL0 = F0H
        MOV  R7, #FFH            ; TH0 = FFH
      MOV  TL0, R6
      MOV  TH0, R7
      MOV  TMOD, #01H            ; TMOD = 01H, SET TIMER0 A 16-BIT TIMER
      MOV  TCON, #10H            ; TCON = 10H, TR0 = 1,GO
      MOV  PCON, #01H            ; ENTER IDLE MODE FOR LAUNCHING THE IN-SYSTEM
PROGRAMMING
;************** ********************************************************************
;* Normal mode 64KB APFLASH program: depending user's application
;*******************************************************************************
NORMAL_MODE:

                                 .
      ; User's application program
                                 .
                                 .
                                 .
```

EXAMPLE 2:

```
;*************************************************************************************************************************** ;*
Example of 4KB LDFLASH program: This loader program will erase the 64KB APFLASH first, then reads the new ;* code from external SRAM and program them into 64KB APFLASH bank. XTAL = 24 MHz
;*************************************************************************************************************************
.chip 8052
.RAMCHK OFF
.symbols

CHPCON    EQU        9FH
TA        EQU        C7H
SFRAL     EQU        ACH
SFRAH     EQU        ADH
SFRFD     EQU        AEH
SFRCN     EQU        AFH

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

```
;*********************************************************************
;*  4KB LDFLASH MAIN PROGRAM
;*********************************************************************
          ORG 100H
MAIN_4K:
        MOV  TA,#AAH
        MOV  TA,#55H
        MOV  CHPCON,#03H        ; CHPCON = 03H, ENABLE IN-SYSTEM PROGRAMMING.
        MOV  SFRCN,#0H
        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,#F0H
             MOV  R7,#FFH
        MOV  TL0,R6
        MOV  TH0,R7
        MOV  TCON,#10H          ; TCON = 10H, TR0 = 1, GO
        MOV  PCON,#01H          ; ENTER IDLE MODE

 UPDATE_64K:
        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,#D0H            ; SET WAKE-UP TIME FOR ERASE OPERATION, ABOUT 15 ms
                                  DEPENDING ON USER'S SYSTEM CLOCK RATE.
        MOV  R7,#8AH
        MOV  TL0,R6
        MOV  TH0,R7

ERASE_P_4K:
        MOV  SFRCN,#22H         ; SFRCN = 22H, ERASE 64K  APFLASH0
                                ; SFRCN = A2H, ERASE 64K APFLASH1
        MOV  TCON,#10H          ; TCON = 10H, TR0 = 1,GO
        MOV  PCON,#01H          ; ENTER IDLE MODE (FOR ERASE OPERATION)

;*********************************************************************
;* BLANK CHECK
;*********************************************************************
        MOV  SFRCN,#0H          ; SFRCN = 00H, READ 64KB APFLASH0
                                ; SFRCN = 80H, READ 64KB APFLASH1
        MOV  SFRAH,#0H          ; START ADDRESS = 0H
        MOV  SFRAL,#0H
        MOV  R6,#FDH            ; SET TIMER FOR READ OPERATION, ABOUT 1.5 μS.
        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
```

```
        MOVX  A,@DPTR
        INC  DPTR
        CJNE  A,SFRFD,ERROR_64K
        CJNE  R2,#0H,READ_VERIFY_64K
        INC  R1
        MOV  SFRAH,R1
        CJNE  R1,#0H,READ_VERIFY_64K

;*****************************************************************************
;* PROGRAMMING COMPLETLY, SOFTWARE RESET CPU
;*****************************************************************************
        MOV  TA,#AAH
        MOV  TA,#55H
        MOV  CHPCON,#83H          ; SOFTWARE RESET. CPU will restart from APFLASH0

ERROR_64K:
        DJNZ  R4,UPDATE_64K       ; IF ERROR OCCURS, REPEAT 3 TIMES.
.                                 ; IN-SYST PROGRAMMING FAIL, USER'S PROCESS TO DEAL WITH IT.
                .
                .
                .
```