Nuvoton Technology Corporation - W79E632A40PL Datasheet

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	8051
Core Size	8-Bit
Speed	40MHz
Connectivity	EBI/EMI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	36
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.25K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°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/w79e632a40pl

Data Pointer Low

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

Mnemonic: DPL Address: 82h

This is the low byte of the standard 8052 16-bit data pointer.

Data Pointer High

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

Mnemonic: DPH Address: 83h

This is the high byte of the standard 8052 16-bit data pointer.

Power Control

Bit:	7	6	5	4	3	2	1	0
	SM0D	SMOD0	-	-	GF1	GF0	PD	IDL

Mnemonic: PCON Address: 87h

SMOD : This bit doubles the serial port baud rate in mode 1, 2, and 3 when set to 1.

SMOD0: Framing Error Detection Enable: When SMOD0 is set to 1, then SCON.7 indicates a Frame Error and acts as the FE flag. When SMOD0 is 0, then SCON.7 acts as per the standard 8052 function.

GF1-0: These two bits are general purpose user flags.

PD: Setting this bit causes the W79E(L)632 to go into the POWER DOWN mode. In this mode all the clocks are stopped and program execution is frozen.

IDL: Setting this bit causes the W79E(L)632 to go into the IDLE mode. In this mode the clocks to the CPU are stopped, so program execution is frozen. But the clock to the serial, timer and interrupt blocks is not stopped, and these blocks continue operating.

Timer Control

Bit:	7	6	5	4	3	2	1	0
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Mnemonic: TCON Address: 88h

TF1: Timer 1 overflow flag: This bit is set when Timer 1 overflows. It is cleared automatically when the program does a timer 1 interrupt service routine. Software can also set or clear this bit.

TR1: Timer 1 run control: This bit is set or cleared by software to turn timer/counter on or off.

TF0: Timer 0 overflow flag: This bit is set when Timer 0 overflows. It is cleared automatically when the program does a timer 0 interrupt service routine. Software can also set or clear this bit.

TR0: Timer 0 run control: This bit is set or cleared by software to turn timer/counter on or off.

IE1: Interrupt 1 edge detect: 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 pin.

IT1: Interrupt 1 type control: Set/cleared by software to specify falling edge/ low level triggered external inputs.

IE0: Interrupt 0 edge detect: 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 pin.

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

GATE: Gating control: When this bit is set, Timer/counter x is enabled only while $\overline{INTx}$ pin is high and TRx control bit is set. When cleared, Timer x is enabled whenever TRx control bit is set.

C / $\overline{T}$: Timer or Counter Select: When cleared, the timer is incremented by internal clocks. When set , the timer counts high-to-low edges of the Tx pin.

M1, M0: Mode Select bits:

M1	M0	MODE
0	0	Mode 0: 8-bits with 5-bit prescale.
0	1	Mode 1: 16-bits, no prescale.
1	0	Mode 2: 8-bits with auto-reload from THx
1	1	Mode 3: (Timer 0) TL0 is an 8-bit timer/counter controlled by the standard Timer 0 control bits. TH0 is a 8-bit timer only controlled by Timer 1 control bits. (Timer 1) Timer/counter is stopped.

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.

WD1	WD0	Interrupt time-out	Reset time-out
0	0	2^{17}	$2^{17} + 512$
0	1	2^{20}	$2^{20} + 512$
1	0	2^{23}	$2^{23} + 512$
1	1	2^{26}	$2^{26} + 512$

T2M: Timer 2 clock select: When T2M is set to 1, timer 2 uses a divide by 4 clock, and when set to 0 it uses a divide by 12 clock.

T1M: Timer 1 clock select: When T1M is set to 1, timer 1 uses a divide by 4 clock, and when set to 0 it uses a divide by 12 clock.

T0M: Timer 0 clock select: When T0M is set to 1, timer 0 uses a divide by 4 clock, and when set to 0 it uses a divide by 12 clock.

MD2–0: Stretch MOVX select bits: These three bits are used to select the stretch value for the MOVX instruction. Using a variable MOVX length enables the user to access slower external memory devices or peripherals without the need for external circuits. The $\overline{RD}$ or $\overline{WR}$ strobe will be stretched by the selected interval. When accessing the on-chip SRAM, the MOVX instruction is always in 2 machine cycles regardless of the stretch setting. By default, the stretch has value of 1. If the user needs faster accessing, then a stretch value of 0 should be selected.

MD2	MD1	MD0	Stretch value	MOVX duration
0	0	0	0	2 machine cycles
0	0	1	1	3 machine cycles *(Default)*
0	1	0	2	4 machine cycles
0	1	1	3	5 machine cycles
1	0	0	4	6 machine cycles
1	0	1	5	7 machine cycles
1	1	0	6	8 machine cycles
1	1	1	7	9 machine cycles

Port 1

Bit:	7	6	5	4	3	2	1	0
	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0

Mnemonic: P1 Address: 90h

P1.7–0: General purpose I/O 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. Some pins also have alternate input or output functions. This alternate functions are described below:

P1.0 : T2 External I/O for Timer/Counter 2
P1.1 : T2EX Timer/Counter 2 Capture/Reload Trigger

Port 4 Control Register A

Bit:	7	6	5	4	3	2	1	0
	P41M1	P41M0	P41C1	P41C0	P40M1	P40M0	P40C1	P40C0

Mnemonic: P4CONA Address: 92h

Port 4 Control Register B

Bit:	7	6	5	4	3	2	1	0
	P43M1	P43M0	P43C1	P43C0	P42M1	P42M0	P42C1	P42C0

Mnemonic: P4CONB Address: 93h

Serial Port Control

Bit:	7	6	5	4	3	2	1	0
	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI

Mnemonic: SCON Address: 98h

SM0/FE: Serial port 0, Mode 0 bit or Framing Error Flag: The SMOD0 bit in PCON SFR determines whether this bit acts as SM0 or as FE. The operation of SM0 is described below. When used as FE, this bit will be set to indicate an invalid stop bit. This bit must be manually cleared in software to clear the FE condition.

SM1: Serial port Mode bit 1:

SM0	SM1	Mode	Description	Length	Baud rate
0	0	0	Synchronous	8	4/12 Tclk
0	1	1	Asynchronous	10	Variable
1	0	2	Asynchronous	11	64/32 Tclk
1	1	3	Asynchronous	11	Variable

SM2: Multiple processors communication. Setting this bit to 1 enables the multiprocessor communication feature in mode 2 and 3. In mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1, then RI will not be activated if a valid stop bit was not received. In mode 0, the SM2 bit controls the serial port clock. If set to 0, then the serial port runs at a divide by 12 clock of the oscillator. This gives compatibility with the standard 8052. When set to 1, the serial clock become divide by 4 of the oscillator clock. This results in faster synchronous serial communication.

REN: Receive enable: When set to 1 serial reception is enabled, otherwise reception is disabled.

TB8: This is the 9th bit to be transmitted in modes 2 and 3. This bit is set and cleared by software as desired.

RB8: In modes 2 and 3 this is the received 9th data bit. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0 it has no function.

TI: Transmit interrupt flag: This flag is set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in all other modes during serial transmission. This bit must be cleared by software.

RI: Receive interrupt flag: This flag is set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bits time in the other modes during serial reception. However the restrictions of SM2 apply to this bit. This bit can be cleared only by software.

Serial Data Buffer

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

Mnemonic: SBUF Address: 99h

SBUF.7-0: Serial data on the serial port 0 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.

Software Reset

Set CHPCON = 0X83, timer and enter IDLE mode. CPU will reset and restart from APFlash after time out.

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

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.

Port 4 Chip-select Polarity

Bit:	7	6	5	4	3	2	1	0
	P43INV	P42INV	P42INV	P40INV	-	-	-	P0UP

Mnemonic: P4CSIN Address: A2h

P4xINV: The active polarity of P4.x when set it as chip-select signal. High = Active High. Low = Active Low.

P0UP: Enable Port 0 weak pull up.

Port 4

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	P4.3	P4.2	P4.1	P4.0

Mnemonic: P4 Address: A5h

P4.3-0: Port 4 is a bi-directional I/O port with internal pull-ups. Port 4 can not use bit-addressable instruction (SETB or CLR).

Interrupt Enable

Bit:	7	6	5	4	3	2	1	0
	EA	ES1	ET2	ES	ET1	EX1	ET0	EX0

Mnemonic: IE Address: A8h

EA: Global enable. Enable/disable all interrupts.
ET2: Enable Timer 2 interrupt.
ES: Enable Serial Port 0 interrupt.
ET1: Enable Timer 1 interrupt
EX1: Enable external interrupt 1
ET0: Enable Timer 0 interrupt
EX0: Enable external interrupt 0

P3.3	$\overline{INT1}$	External interrupt 1
P3.2	$\overline{INT0}$	External interrupt 0
P3.1	TxD	Serial port 0 output
P3.0	RxD	Serial port 0 input

Interrupt Priority

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

Mnemonic: IP Address: B8h

IP.7: This bit is un-implemented and will read high.

PT2: This bit defines the Timer 2 interrupt priority. PT2 = 1 sets it to higher priority level.

PS: This bit defines the Serial port 0 interrupt priority. PS = 1 sets it to higher priority level.

PT1: This bit defines the Timer 1 interrupt priority. PT1 = 1 sets it to higher priority level.

PX1: This bit defines the External interrupt 1 priority. PX1 = 1 sets it to higher priority level.

PT0: This bit defines the Timer 0 interrupt priority. PT0 = 1 sets it to higher priority level.

PX0: This bit defines the External interrupt 0 priority. PX0 = 1 sets it to higher priority level.

Slave Address Mask Enable

Bit:	7	6	5	4	3	2	1	0

Mnemonic: SADEN Address: B9h

SADEN: This register enables the Automatic Address Recognition feature of the Serial port 0. When a bit in the SADEN is set to 1, the same bit location in SADDR will be compared with the incoming serial data. When SADEN.n is 0, then the bit becomes a "don't care" in the comparison. This register enables the Automatic Address Recognition feature of the Serial port 0. When all the bits of SADEN are 0, interrupt will occur for any incoming address.

Power Management Register

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	ALE-OFF	-	DME0

Mnemonic: PMR Address: C4h

ALE0FF: This bit disables the expression of the ALE signal on the device pin during all on-board program and data memory accesses. External memory accesses will automatically enable ALE independent of ALEOFF.

0 = ALE expression is enable; 1 = ALE expression is disable

DME0: This bit determines the on-chip MOVX SRAM to be enabled or disabled. Set this bit to 1 will enable the on-chip 1KB MOVX SRAM.

WTRF: Watchdog Timer Reset Flag. Hardware will set this bit when the watchdog timer causes a reset. Software can read it but must clear it manually. A power-fail reset will also clear the bit. This bit helps software in determining the cause of a reset. If EWT = 0, the watchdog timer will have no affect on this bit.

EWT: Enable Watchdog timer Reset. Setting this bit will enable the Watchdog timer Reset function.

RWT: Reset Watchdog Timer. This bit helps in putting the watchdog timer into a know state. It also helps in resetting the watchdog timer before a time-out occurs. Failing to set the EWT before time-out will cause an interrupt, if EWDI (EIE.4) is set, and 512 clocks after that a watchdog timer reset will be generated if EWT is set. This bit is self-clearing by hardware.

The WDCON SFR is set to a 0x0x0xx0b on an external reset. WTRF is set to a 1 on a Watchdog timer reset, but to a 0 on power on/down resets. WTRF is not altered by an external reset. POR is set to 1 by a power-on reset. EWT is set to 0 on a Power-on reset and unaffected by other resets.

All the bits in this SFR have unrestricted read access. POR, EWT, WDIF and RWT require Timed Access procedure to write. The remaining bits have unrestricted write accesses. Please refer TA register description.

```
TA          EG      C7H
WDCON       REG     D8H
CKCON       REG     8EH
        MOV  TA, #AAH
        MOV  TA, #55H

        SETB  WDCON.0               ; Reset watchdog timer
        ORL  CKCON, #11000000B      ; Select 26 bits watchdog timer
        MOV  TA, #AAH
        MOV  TA, #55H
        ORL  WDCON, #00000010B      ; Enable watchdog
```

Accumulator

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

Mnemonic: ACC Address: E0h

ACC.7-0: The A (or ACC) register is the standard 8052 accumulator.

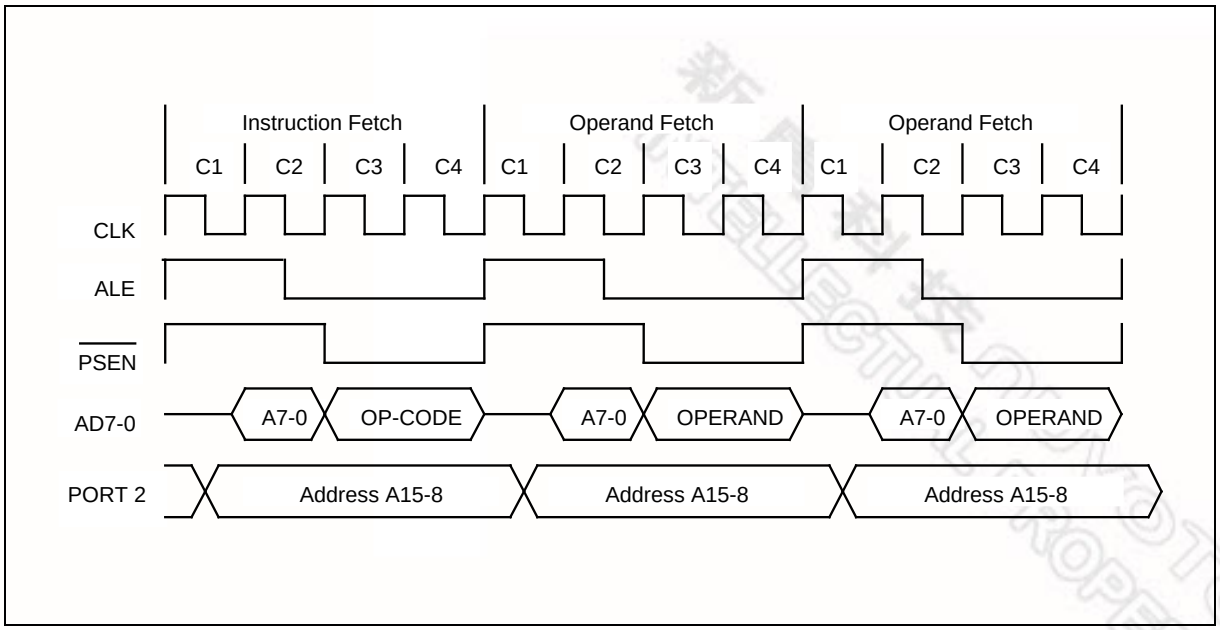

Figure 5. Three Cycle Instruction Timing

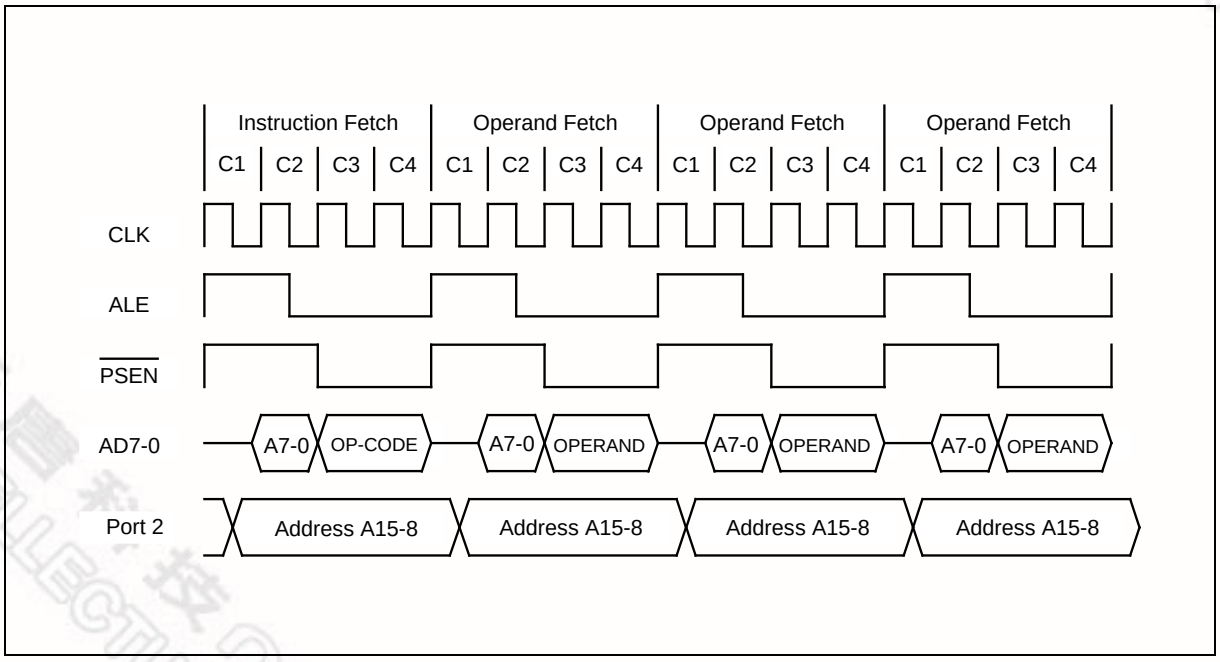

Figure 6. Four Cycle Instruction Timing

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

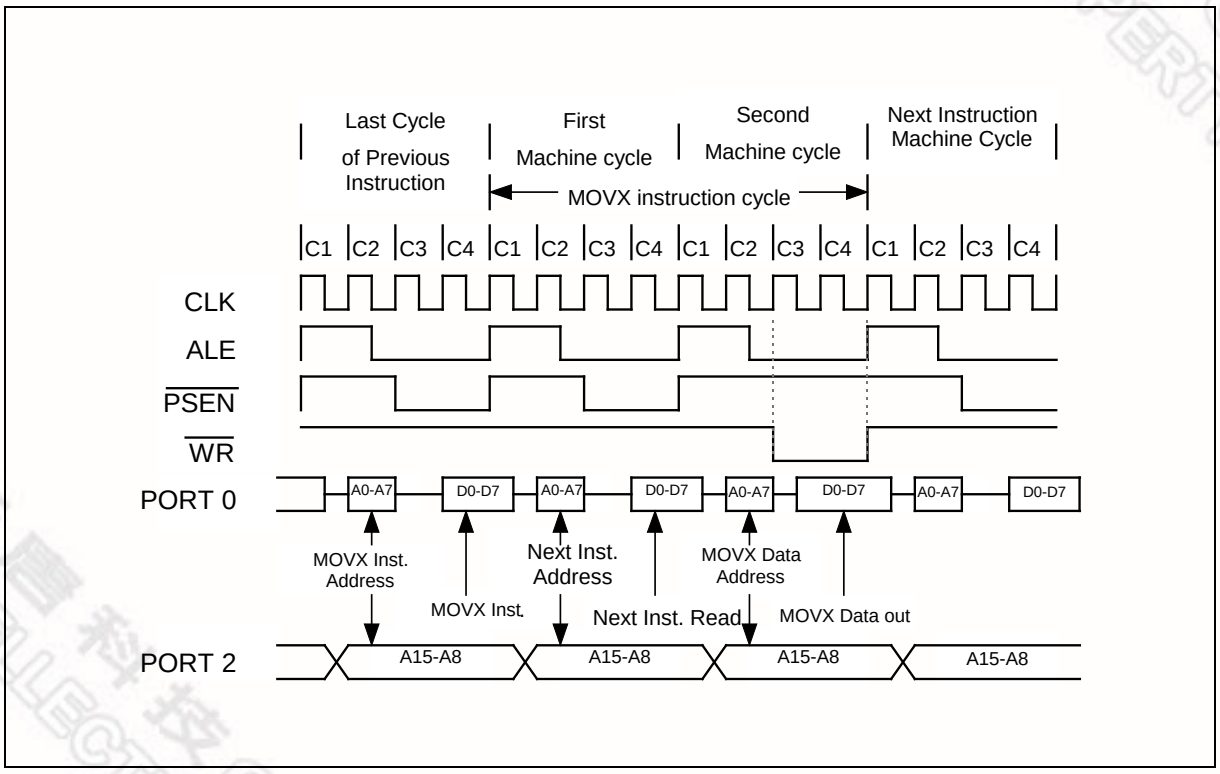

Figure 8. Data Memory Write with Stretch Value = 0

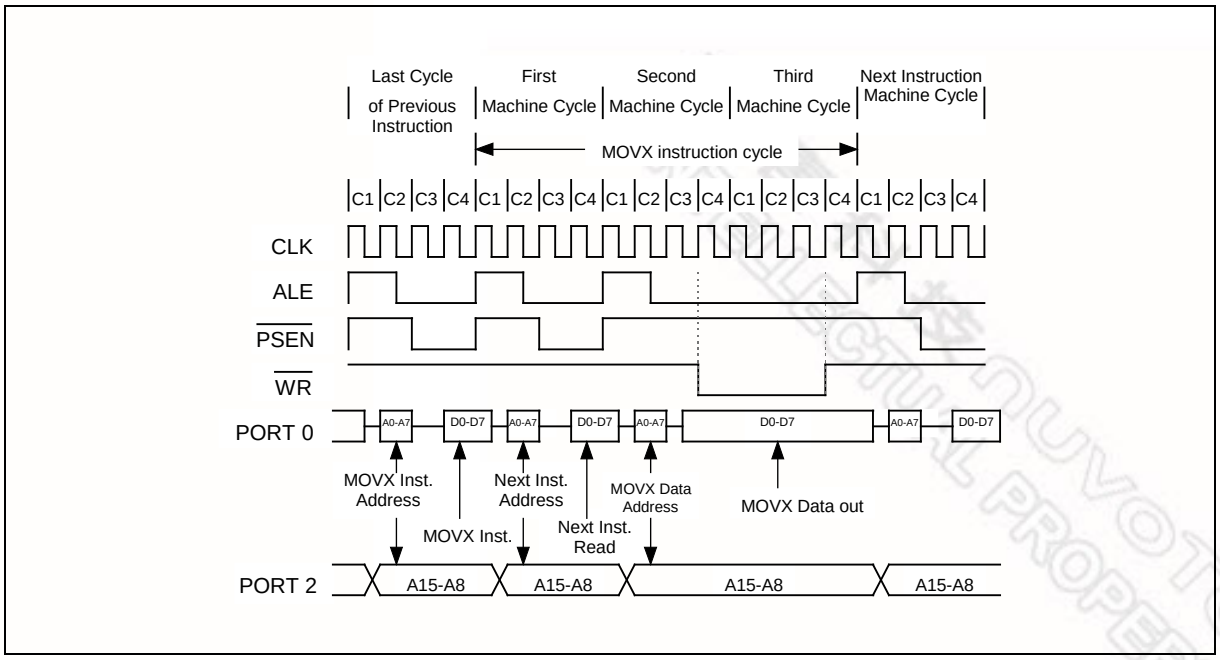

Figure 9. Data Memory Write with Stretch Value = 1

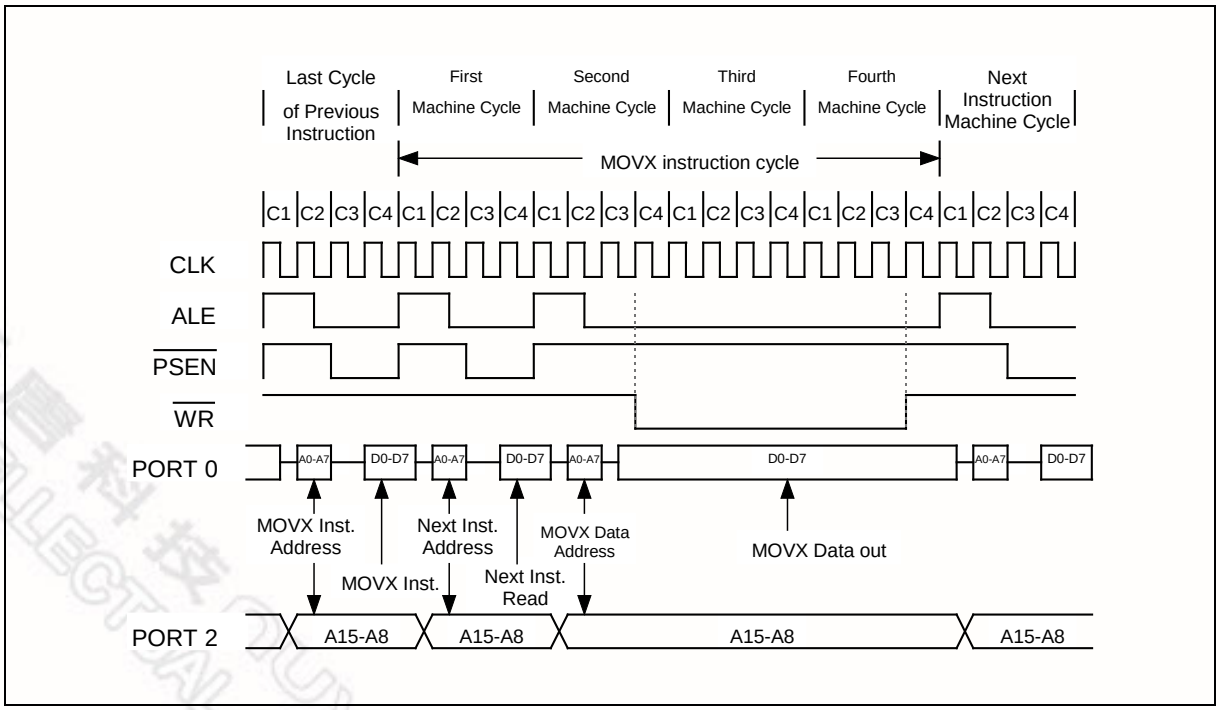

Figure 10. Data Memory Write with Stretch Value = 2

Priority Level Structure

There are three priority levels for the interrupts, highest, high and low. The interrupt sources can be individually set to either high or low levels. Naturally, a higher priority interrupt cannot be interrupted by a lower priority interrupt. However there exists a pre-defined hierarchy amongst the interrupts themselves. This hierarchy comes into play when the interrupt controller has to resolve simultaneous requests having the same priority level. This hierarchy is defined as shown below; the interrupts are numbered starting from the highest priority to the lowest.

Table 7. Priority structure of interrupts

SOURCE	FLAG	VECTOR ADDRESS	PRIORITY LEVEL
External Interrupt 0	IE0	0003h	1(highest)
Timer 0 Overflow	TF0	000Bh	2
External Interrupt 1	IE1	0013h	3
Timer 1 Overflow	TF1	001Bh	4
Serial Port	RI + TI	0023h	5
Timer 2 Overflow	TF2 + EXF2	002Bh	6
Watchdog Timer	WDIF	0063h	7 (lowest)

time-base may be either clock cycles/12 or clock cycles/4 as selected by the bits TxM of the CKCON SFR.

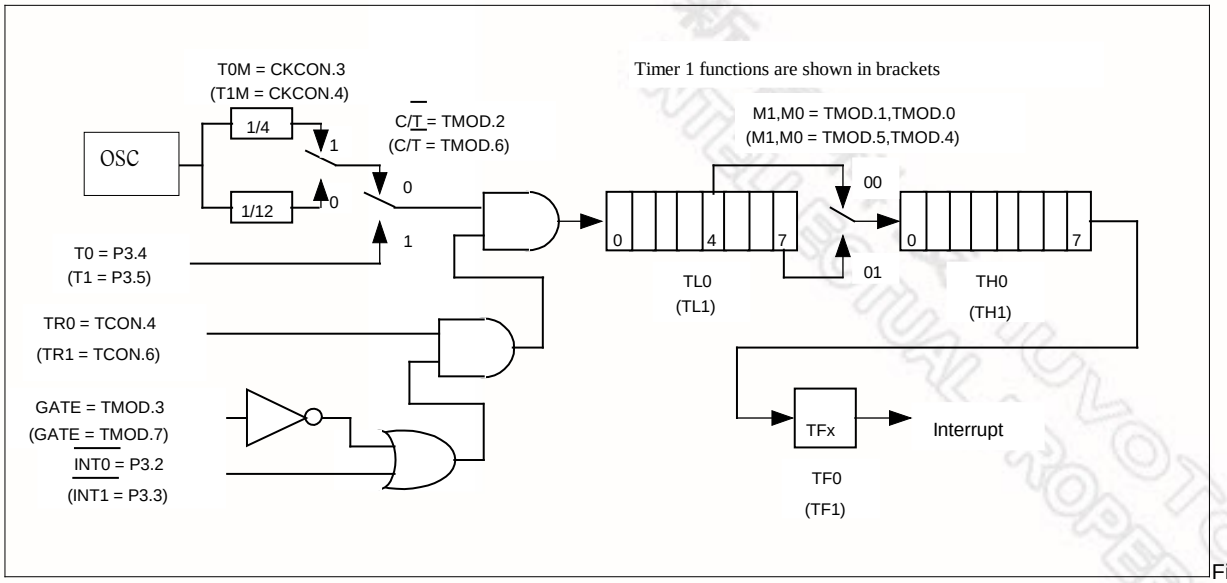

Figure 11. Timer/Counter Mode 0 & Mode 1

Mode 1

Mode 1 is similar to Mode 0 except that the counting register forms a 16 bit counter, rather than a 13 bit counter. This means that all the bits of THx and TLx are used. Roll-over occurs when the timer moves from a count of FFFFh to 0000h. The timer overflow flag TFx of the relevant timer is set and if enabled an interrupt will occur. The selection of the time-base in the timer mode is similar to that in Mode 0. The gate function operates similarly to that in Mode 0.

Mode 2

In Mode 2, the timer/counter is in the Auto Reload Mode. In this mode, TLx acts as a 8 bit count register, while THx holds the reload value. When the TLx register overflows from FFh to 00h, the TFx bit in TCON is set and TLx is reloaded with the contents of THx, and the counting process continues from here. The reload operation leaves the contents of the THx register unchanged. Counting is enabled by the TRx bit and proper setting of GATE and $\overline{\text{INTx}}$ pins. As in the other two modes 0 and 1 mode 2 allows counting of either clock cycles (clock/12 or clock/4) or pulses on pin Tn.

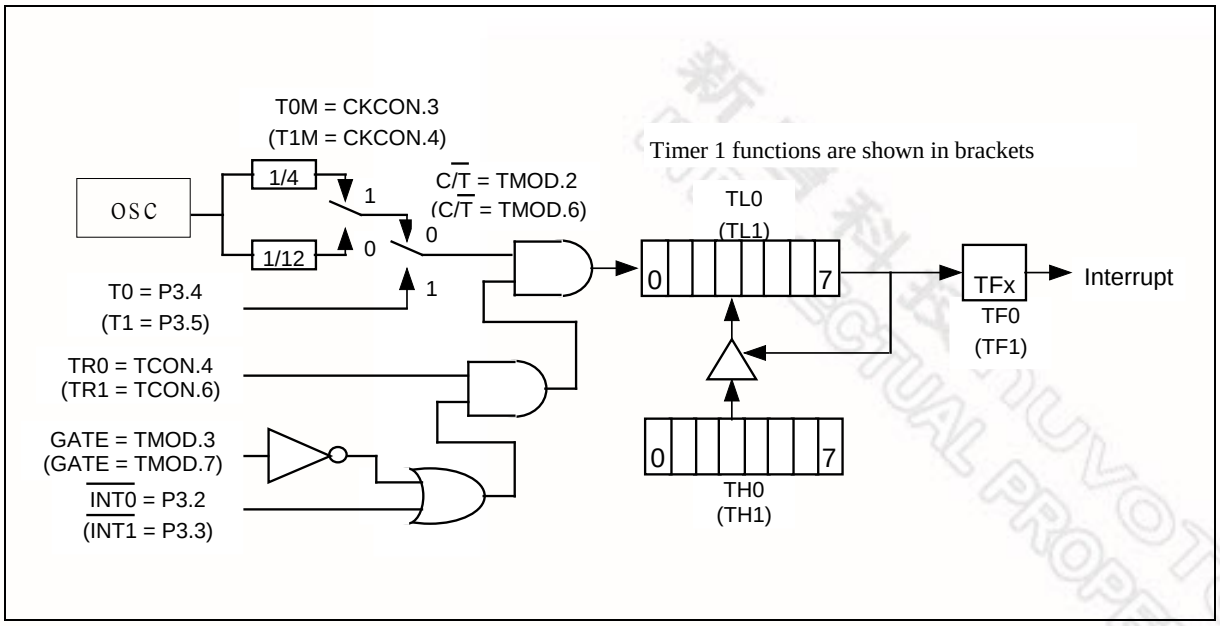

Figure 12. Timer/Counter Mode 2.

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.

Auto-reload Mode, Counting Up/Down

Timer/Counter 2 will be in auto-reload mode as an up/down counter if CP / $\overline{RL2}$ bit in T2CON is cleared and the DCEN bit in T2MOD is set. In this mode, Timer/Counter 2 is an up/down counter whose direction is controlled by the T2EX pin. A 1 on this pin cause the counter to count up. An overflow while counting up will cause the counter to be reloaded with the contents of the capture registers. The next down count following the case where the contents of Timer/Counter equal the capture registers will load an FFFFh into Timer/Counter 2. In either event a reload will set the TF2 bit. A reload will also toggle the EXF2 bit. However, the EXF2 bit can not generate an interrupt while in this mode.

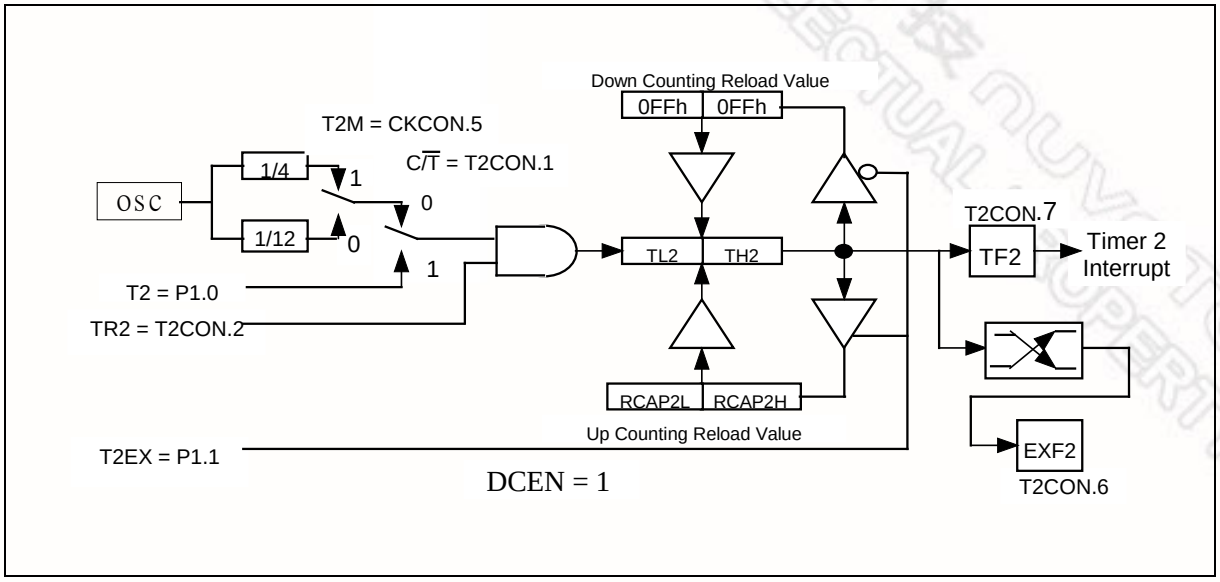

Figure 16. 16-Bit Auto-reload Up/Down Counter

```
        mov     ckcon,#01h      ; select 2 ^ 17 timer
;       mov     ckcon,#61h      ; select 2 ^ 20 timer
;       mov     ckcon,#81h      ; select 2 ^ 23 timer
;       mov     ckcon,#c1h      ; select 2 ^ 26 timer
        mov     TA,#aah
        mov     TA,#55h
        mov     WDCON,#00000011B
        setb    EWDI
        setb    ea
        jmp     $               ; wait time out
```

The TI flag is set high in C1 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 C1 following the last rising edge of the shift clock on TxD. This will stop reception, till the RI is cleared by software.

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 C1 following the first roll-over of divide by 16 counter. The next bit is placed on TxD pin at C1 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 C1 state after the stop bit has been put out on TxD pin. This will be at the 10th rollover of the divide-by-16 counter 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 counter is immediately reset. This helps to align the bit boundaries with the rollovers of the divide- by-16 counter.

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. 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 8 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
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.

11.1 Framing Error Detection

A Frame Error occurs when a valid stop bit is not detected. This could indicate incorrect serial data communication. Typically the frame error is due to noise and contention on the serial communication line. The W79E(L)632 has the facility to detect such framing errors and set a flag which can be checked by software.

The Frame Error FE(FE_1) bit is located in SCON.7. This bit is normally used as SM0 in the standard 8051 family. However, in the W79E(L)632 it serves a dual function and is called SM0/FE. There are actually two separate flags, one for SM0 and the other for FE. The flag that is actually accessed as SCON.7 is determined by SMOD0 (PCON.6) bit. When SMOD0 is set to 1, then the FE flag is indicated in SM0/FE. When SMOD0 is set to 0, then the SM0 flag is indicated in SM0/FE.

The FE bit is set to 1 by hardware but must be cleared by software. Note that SMOD0 must be 1 while reading or writing to FE. If FE is set, then any following frames received without any error will not clear the FE flag. The clearing has to be done by software.

11.2 Multiprocessor Communications

Multiprocessor communications makes use of the 9th data bit in modes 2 and 3. In the W79E(L)632, the RI flag is set only if the received byte corresponds to the Given or Broadcast address. This hardware feature eliminates the software overhead required in checking every received address, and greatly simplifies the software programmer task.

In the multiprocessor communication mode, the address bytes are distinguished from the data bytes by transmitting the address with the 9th bit set high. When the master processor wants to transmit a block of data to one of the slaves, it first sends out the address of the targeted slave (or slaves). All the slave processors should have their SM2 bit set high when waiting for an address byte. This ensures that they will be interrupted only by the reception of a address byte. The Automatic address recognition feature ensures that only the addressed slave will be interrupted. The address comparison is done in hardware not software.

The addressed slave clears the SM2 bit, thereby clearing the way to receive data bytes. With SM2 = 0, the slave will be interrupted on the reception of every single complete frame of data. The unaddressed slaves will be unaffected, as they will be still waiting for their address. In Mode 1, the 9th bit is the stop bit, which is 1 in case of a valid frame. If SM2 is 1, then RI is set only if a valid frame is received and the received byte matches the Given or Broadcast address.

The Master processor can selectively communicate with groups of slaves by using the Given Address. All the slaves can be addressed together using the Broadcast Address. The addresses for each slave are defined by the SADDR and SADEN SFRs. The slave address is an 8-bit value specified in the SADDR SFR. The SADEN SFR is actually a mask for the byte value in SADDR. If a bit position in SADEN is 0, then the corresponding bit position in SADDR is don't care. Only those bit positions in SADDR whose corresponding bits in SADEN are 1 are used to obtain the Given Address. This gives the user flexibility to address multiple slaves without changing the slave address in SADDR.

The following example shows how the user can define the Given Address to address different slaves.

```
;*  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
        INC  SFRAH
        MOV  A,SFRAH
```

```
        CJNE  A,#0H,BLANK_CHECK_LOOP   ; END ADDRESS = FFFFH
        JMP  PROGRAM_64KROM

BLANK_CHECK_ERROR:
        JMP  $

;*******************************************************************************
;
;* RE-PROGRAMMING 64KB APFlash BANK
;*******************************************************************************
;
PROGRAM_64KROM:
        MOV  R2,#00H            ; TARGET LOW BYTE ADDRESS
        MOV  R1,#00H            ; TARGET HIGH BYTE ADDRESS
        MOV  DPTR,#0H
        MOV  SFRAH,R1           ; SFRAH, TARGET  HIGH ADDRESS
        MOV  SFRCN,#21H         ; SFRCN = 21H, PROGRAM 64K APFLASH0
                                ; SFRCN = A1H, PROGRAM 64K APFLASH1
        MOV  R6,#9CH            ; SET TIMER FOR PROGRAMMING, ABOUT 50 μS.
        MOV  R7,#FFH
        MOV  TL0,R6
        MOV  TH0,R7

PROG_D_64K:
        MOV  SFRAL,R2           ; SFRAL = LOW BYTE ADDRESS
        CALL  GET_BYTE_FROM_PC_TO_ACC     ; THIS PROGRAM IS BASED ON USER'S CIRCUIT.
        MOV  @DPTR,A            ; SAVE DATA  INTO SRAM TO VERIFY CODE.
        MOV  SFRFD,A            ; SFRFD = 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_64K
        INC  R1
        MOV  SFRAH,R1
        CJNE  R1,#0H,PROG_D_64K

;*******************************************************************************
;
; * VERIFY 64KB APFLASH BANK
;*******************************************************************************
;
        MOV  R4,#03H            ; ERROR COUNTER
        MOV  R6,#FDH            ; SET TIMER FOR READ VERIFY, ABOUT 1.5 μS.
        MOV  R7,#FFH
        MOV  TL0,R6
        MOV  TH0,R7
        MOV  DPTR,#0H           ; The start address of sample code
        MOV  R2,#0H             ; Target  low byte address
        MOV  R1,#0H             ; Target  high byte address
        MOV  SFRAH,R1           ; SFRAH, Target high address
        MOV  SFRCN,#00H         ; SFRCN = 00H, Read APFlash0
                                ; SFRCN = 80H , Read APFlash1

READ_VERIFY_64K:
        MOV  SFRAL,R2           ; SFRAL = LOW ADDRESS
        MOV  TCON,#10H          ; TCON = 10H, TR0 = 1,GO
        MOV  PCON,#01H
        INC  R2
        MOVX  A,@DPTR
```