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, I²C, 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)	4.5V ~ 5.5V
Data Converters	A/D 4x10b
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/w79e633a40pl

Mnemonic: SADDR Address: A9h

SADDR The SADDR should be programmed to the given or broadcast address for serial port to which the slave processor is designated.

ROM Banking Control

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	EN128K	DCP12	DCP11	DCP10

Mnemonic: ROMCON Address: ABh

BIT	NAME	FUNCTION
7-4	-	Reserved
3	EN128K	**On-chip ROM banking enable.** Set this bit to enable APFlash0 and APFlash1 by banking mechanism. The P1.x is selected to be the auxiliary highest address line A16.
2-0	DCP1[2:0]	A16 selection. By banking mechanism, address 16 (A16) indicates where the CPU fetches code from AP0(A16=0) or AP1(AP16=1). By default, P1.7 is defined as A16. See table below

DCP1[2:0]

A16	P1.0	P1.1	P1.2	P1.3	P1.4	P1.5	P1.6	P1.7
DCP1[2:0]	000	001	010	011	100	101	110	111

ISP Address Low Byte

Bit:	7	6	5	4	3	2	1	0
	A7	A6	A5	A4	A3	A2	A1	A0

Mnemonic: SFRAL Address: ACh

Low byte destination address for In System Programming operations.

ISP Address High Byte

Bit:	7	6	5	4	3	2	1	0
	A15	A14	A13	A12	A11	A10	A9	A8

Mnemonic: SFRAH Address: ADh

Low byte destination address for In System Programming operations. (SFRAH, SFRAL) represents the address of the ROM byte that will be erased, programmed or read.

ISP Data Buffer

Bit:	7	6	5	4	3	2	1	0
	D7	D6	D5	D4	D3	D2	D1	D0

Mnemonic: SFRFD Address: AEh

In ISP mode, read/write a specific byte ROM content must go through SFRFD register.

ISP Operation Modes

Bit:	7	6	5	4	3	2	1	0
	BANK	WFWIN	NOE	NCE	CTRL3	CTRL2	CTRL1	CTRL0

Mnemonic: SFRCN Address: AFh

BANK Select APFlash banks for ISP. Set it 1 to access APFlash1, clear it for access to APFlash0.

WFWIN On-chip FLASH EPROM bank select for in-system programming.
0= AP FLASH EPROM bank is selected as destination for re-programming.
1= LD FLASH EPROM bank is selected as destination for re-programming.

NOE Flash EPROM output enable.

NCE Flash EPROM chip enable.

CTRL[3:0] The Flash Control Signals.

ISP MODE	BANK	WFWIN	NOE	NCE	CTRL[3:0]	SFRAH, SFRAL	SFRFD
Erase 4KB LDFlash	0	1	1	0	0010	X	X
Erase 64K APFlash0	0	0	1	0	0010	X	X
Erase 64K APFlash1	1	0	1	0	0010	X	X
Program 4KB LDFlash	0	1	1	0	0001	Address in	Data in
Program 64KB APFlash0	0	0	1	0	0001	Address in	Data in
Program 64KB APFlash1	1	0	1	0	0001	Address in	Data in
Read 4KB LDFlash	0	1	0	0	0000	Address in	Data out
Read 64KB APFlash0	0	0	0	0	0000	Address in	Data out
Read 64KB APFlash1	1	0	0	0	0000	Address in	Data out

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 I/O port. Each pin also has an alternative input or output function, which is described below.

BIT	NAME	FUNCTION
7	P3.7	$\overline{RD}$: strobe for reading from external RAM
6	P3.6	$\overline{WR}$: strobe for writing to external RAM
5	P3.5	T1: Timer 1 external count input
4	P3.4	T0: Timer 0 external count input

STATUS REGISTER

Bit:	7	6	5	4	3	2	1	0
	-	HIP	LIP	-	-	-	-	-

Mnemonic: STATUS Address: C5h

BIT	NAME	FUNCTION
7	-	Reserved.
6	HIP	High-Priority Interrupt Status. When set, it indicates that the software is servicing a high-priority interrupt. This bit is cleared when the program executes the corresponding RETI instruction.
5	LIP	Low-Priority Interrupt Status. When set, it indicates that the software is servicing a low-priority interrupt. This bit is cleared when the program executes the corresponding RETI instruction.
4-0	-	Reserved.

ADC Pin Switch

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

Mnemonic: ADCPS Address: C6h

BIT	NAME	FUNCTION
7-4	-	Must be zeros
3-0	ADCPS.3-0	Switch I/O pins, P1.7~P1.4, to analog input. Analog inputs of ADC0-ADC3 share the I/O pins from P1.4 to P1.7. Setting the bits in ADCPS[3:0] switches the corresponding pins of Port1[7:4] to analog input function.

ADCPS.3-0: Switch P1.7~P1.4 to analog input function

BIT	CORRESPONDING PIN
ADCPS.0	**P1.4**
ADCPS.1	**P1.5**
ADCPS.2	**P1.6**
ADCPS.3	**P1.7**

Timed Access

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

Mnemonic: TA Address: C7h

TA This register controls the access to protected bits. To access protected bits, the program must write AAH, followed immediately by 55H, to TA. This opens a window for three machine cycles, during which the program can write to protected bits.

TIMER 2 CONTROL

Bit:	7	6	5	4	3	2	1	0

Instruction Set for W79E(L)633, continued

OP-CODE	HEX CODE	BYTES	W79E(L)633 MACHINE CYCLE	W79E(L)633 CLOCK CYCLES	8032 CLOCK CYCLES	W79E(L)633 VS. 8032 SPEED RATIO
DEC R5	1D	1	1	4	12	3
DEC R6	1E	1	1	4	12	3
DEC R7	1F	1	1	4	12	3
DEC @R0	16	1	1	4	12	3
DEC @R1	17	1	1	4	12	3
DEC direct	15	2	2	8	12	1.5
MUL AB	A4	1	5	20	48	2.4
DIV AB	84	1	5	20	48	2.4
DA A	D4	1	1	4	12	3
ANL A, R0	58	1	1	4	12	3
ANL A, R1	59	1	1	4	12	3
ANL A, R2	5A	1	1	4	12	3
ANL A, R3	5B	1	1	4	12	3
ANL A, R4	5C	1	1	4	12	3
ANL A, R5	5D	1	1	4	12	3
ANL A, R6	5E	1	1	4	12	3
ANL A, R7	5F	1	1	4	12	3
ANL A, @R0	56	1	1	4	12	3
ANL A, @R1	57	1	1	4	12	3
ANL A, direct	55	2	2	8	12	1.5
ANL A, #data	54	2	2	8	12	1.5
ANL direct, A	52	2	2	8	12	1.5
ANL direct, #data	53	3	3	12	24	2
ORL A, R0	48	1	1	4	12	3
ORL A, R1	49	1	1	4	12	3
ORL A, R2	4A	1	1	4	12	3
ORL A, R3	4B	1	1	4	12	3
ORL A, R4	4C	1	1	4	12	3
ORL A, R5	4D	1	1	4	12	3
ORL A, R6	4E	1	1	4	12	3
ORL A, R7	4F	1	1	4	12	3
ORL A, @R0	46	1	1	4	12	3
ORL A, @R1	47	1	1	4	12	3

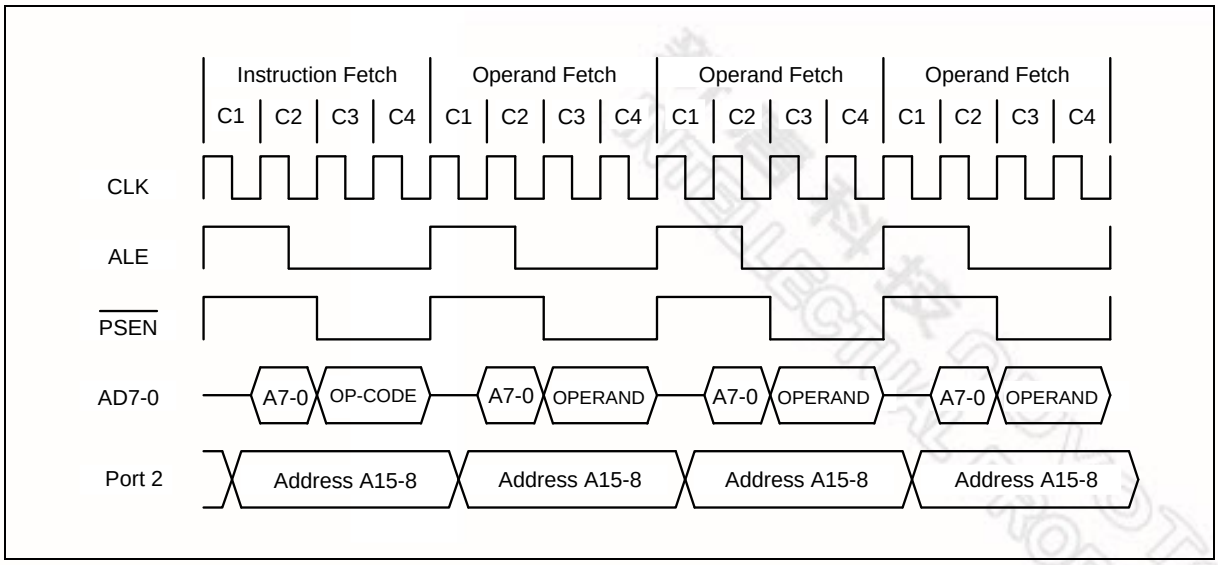


Figure 7-4 Four Cycle Instruction Timing

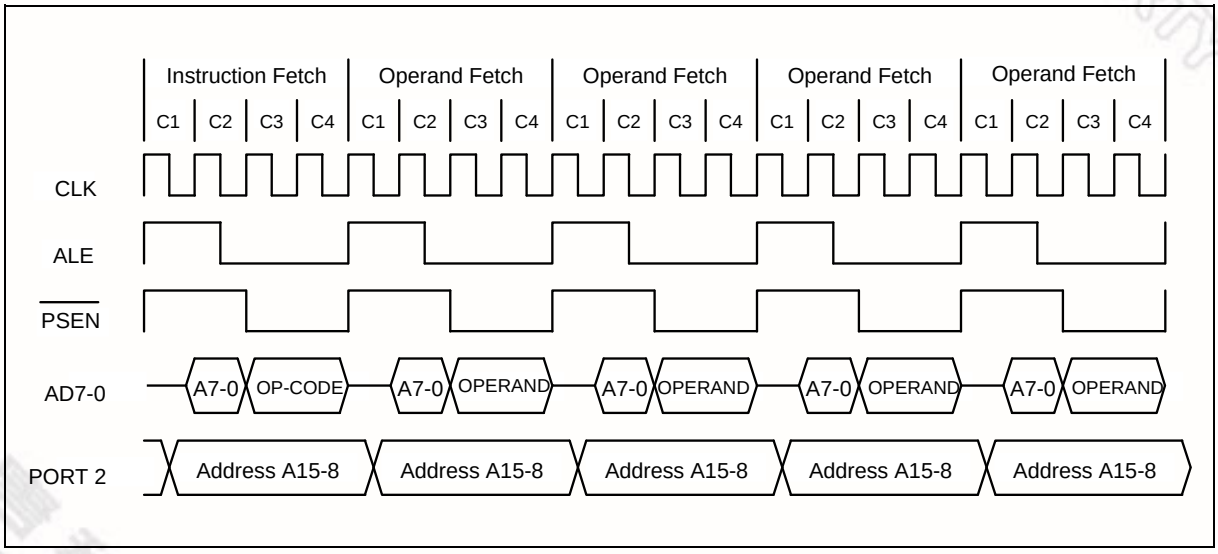


Figure 7-5 Five Cycle Instruction Timing

7.1.1 External Data Memory Access Timing

The timing for the MOVX instruction is another feature of the W79E(L)633. In the standard 8051/52, the MOVX instruction has a fixed execution time of 2 machine cycles. However, in the W79E(L)633, the duration of the access can be controlled by the user.

The instruction starts off as a normal op-code fetch that takes four clocks. In the next machine cycle, the W79E(L)633 puts out the external memory address, and the actual access occurs. The user can control the duration of this access by setting the stretch value in CKCON, bits 2 – 0. As shown in the table below, these three bits can range from zero to seven, resulting in MOVX instructions that take two to nine machine cycles. The default value is one, resulting in a MOVX instruction of three machine cycles.

```
start:
        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
```

Clock Control

WD1, WD0: CKCON.7, CKCON.6 - Watchdog Timer Mode select bits. These two bits select the time-out interval for the Watchdog Timer. The reset interval is 512 clocks longer than the selected interval. The default time-out is 2^{17} clocks, the shortest time-out period.

14. Serial Port

The W79E(L)633 serial port is a full-duplex port, and the W79E(L)633 provides additional features, such as Frame Error Detection and Automatic Address Recognition. The serial port is capable of synchronous and asynchronous communication. In synchronous mode, the W79E(L)633 generates the clock and operates in half-duplex mode. In asynchronous mode, the serial port can simultaneously transmit and receive data. The transmit register and the receive buffer are both addressed as SBUF, but any write to SBUF writes to the transmit register while any read from SBUF reads from the receive buffer. The serial port can operate in four modes, as described below.

14.1 Mode 0

This mode provides half-duplex, synchronous communication with external devices. In this mode, serial data is transmitted and received on the RXD line, and the W79E(L)633 provides the shift clock on TxD, whether the device is transmitting or receiving. Eight bits are transmitted or received per frame, LSB first. The baud rate is 1/12 or 1/4 of the oscillator frequency, as determined by the SM2 bit (SCON.5; 0 = 1/12; 1 = 1/4). This programmable baud rate is the only difference between the standard 8051/52 and the W79E(L)633 in mode 0.

Any write to SBUF starts transmission. The shift clock is activated, and data is shifted out on RxD until all eight bits are transmitted. If SM2 is 1, the data appears on RxD one clock period before the falling edge of the shift clock on TxD. Then, the clock remains low for two clock periods before going high again. If SM2 is 0, the data appears on RxD three clock periods before the falling edge of the shift clock on TxD, and the clock on TxD remains low for six clock periods before going high again. This ensures that, at the receiving end, the data on the RxD line can be clocked on the rising edge of the shift clock or latched when the clock is low. The TI flag is set high in C1 following the end of transmission. The functional block diagram is shown below.

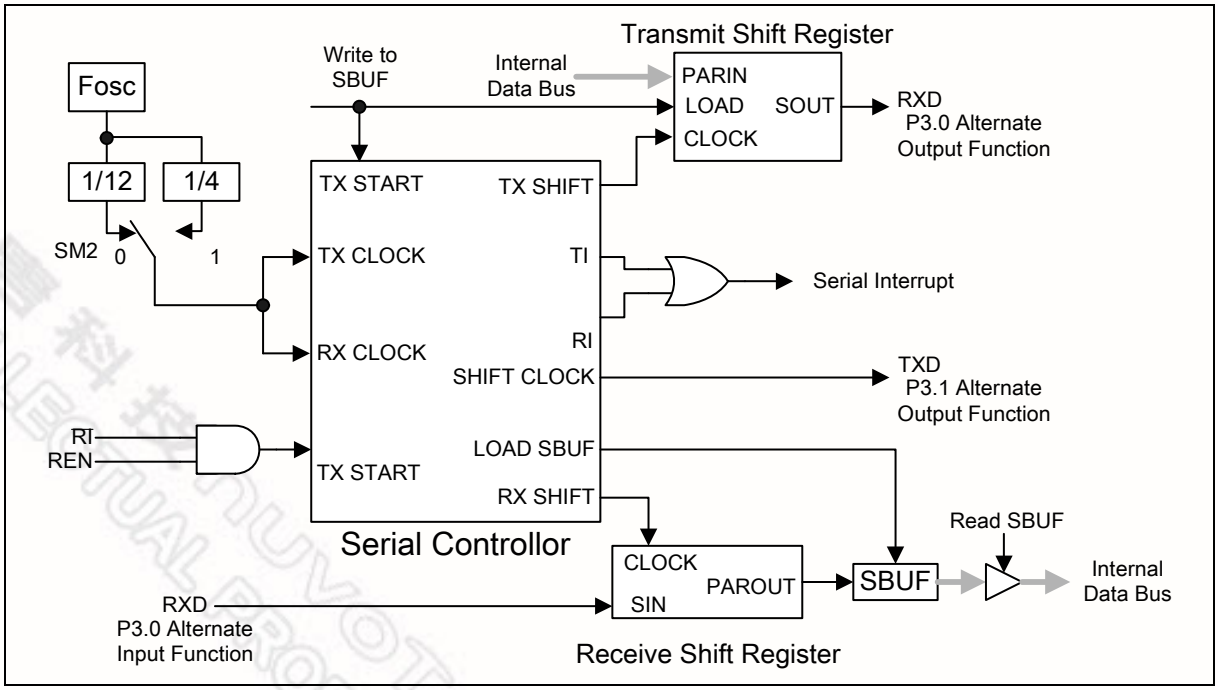


Figure 14-1 Serial Port Mode 0

15.1.4 Status Register, I2STATUSx

I2STATUSx is an 8-bit read-only register. The five most significant bits contain the status code. The three least significant bits are always 0. There are 23 possible status codes. When I2STATUSx contains F8H, no serial interrupt is requested. All other I2STATUSx values correspond to defined I2C ports states. When each of these states is entered, a status interrupt is requested (SI = 1). A valid status code is present in I2STATUSx one machine cycle after SI is set by hardware and is still present one machine cycle after SI has been reset by software.

In addition, state 00H stands for a Bus Error. A Bus Error occurs when a START or STOP condition is present at an illegal position in the format frame. Examples of illegal positions are during the serial transfer of an address byte, a data byte or an acknowledge bit.

15.1.5 I2C Clock Baud Rate Control, I2CLKx

The data baud rate of I2C is determined by I2CLKx register when I2C port is in a master mode. In the slave modes, SIO1 will automatically synchronize with any clock frequency up to 400 KHz from master I2C device.

The data baud rate of I2C setting conforms to the following equation.

Data Baud Rate of I2C = F_{CPU} / (I2CLKx + 1), where $F_{CPU} = F_{OSC}/4$.

For example, if F_{OSC}=16MHz (F_{CPU}=4MHz), the I2CLK=40(28H), the baud rate =4MHz/(40+1) = 97.56K bits/sec.

15.1.6 I2C Time-out Counter, I2Timerx

In W79E(L)633, the I2C logic block provides a 14-bit timer-out counter that helps user to deal with bus pending problem. When SI is cleared user can set ENTI=1 to start the time-out counter. If I2C bus hangs up too long to get any valid signal from devices on the bus, the time-out counter overflows cause TIF=1 to request an I2C interrupt. The I2C interrupt is requested in the condition of either SI=1 or TIF=1. Flags SI and TIF must be cleared by software.

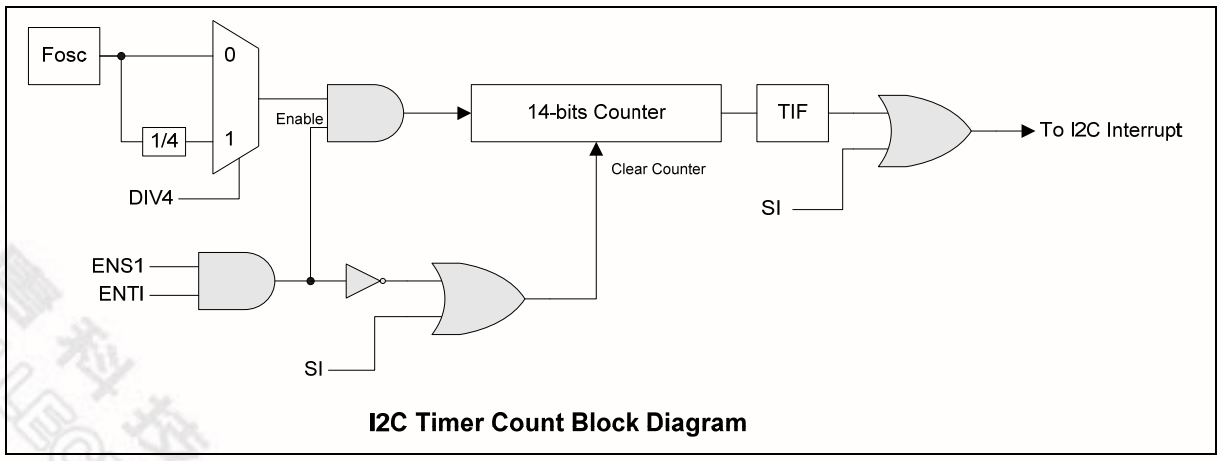


Figure 15-3 I2C Time-out Counter

15.3.1 Master/Transmitter Mode

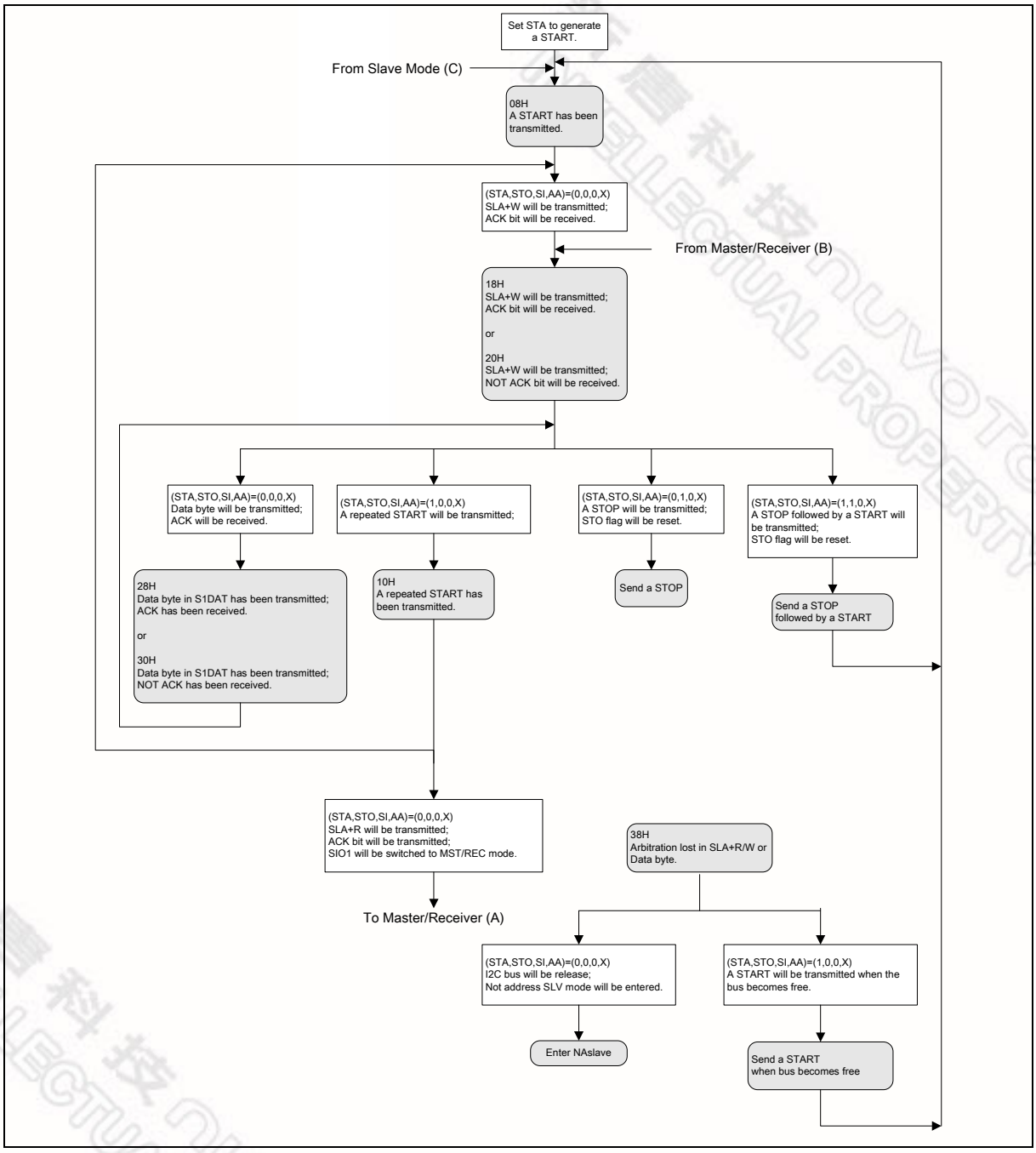


Figure 15-5 Master Transmitter Mode

15.3.3 Slave/Transmitter Mode

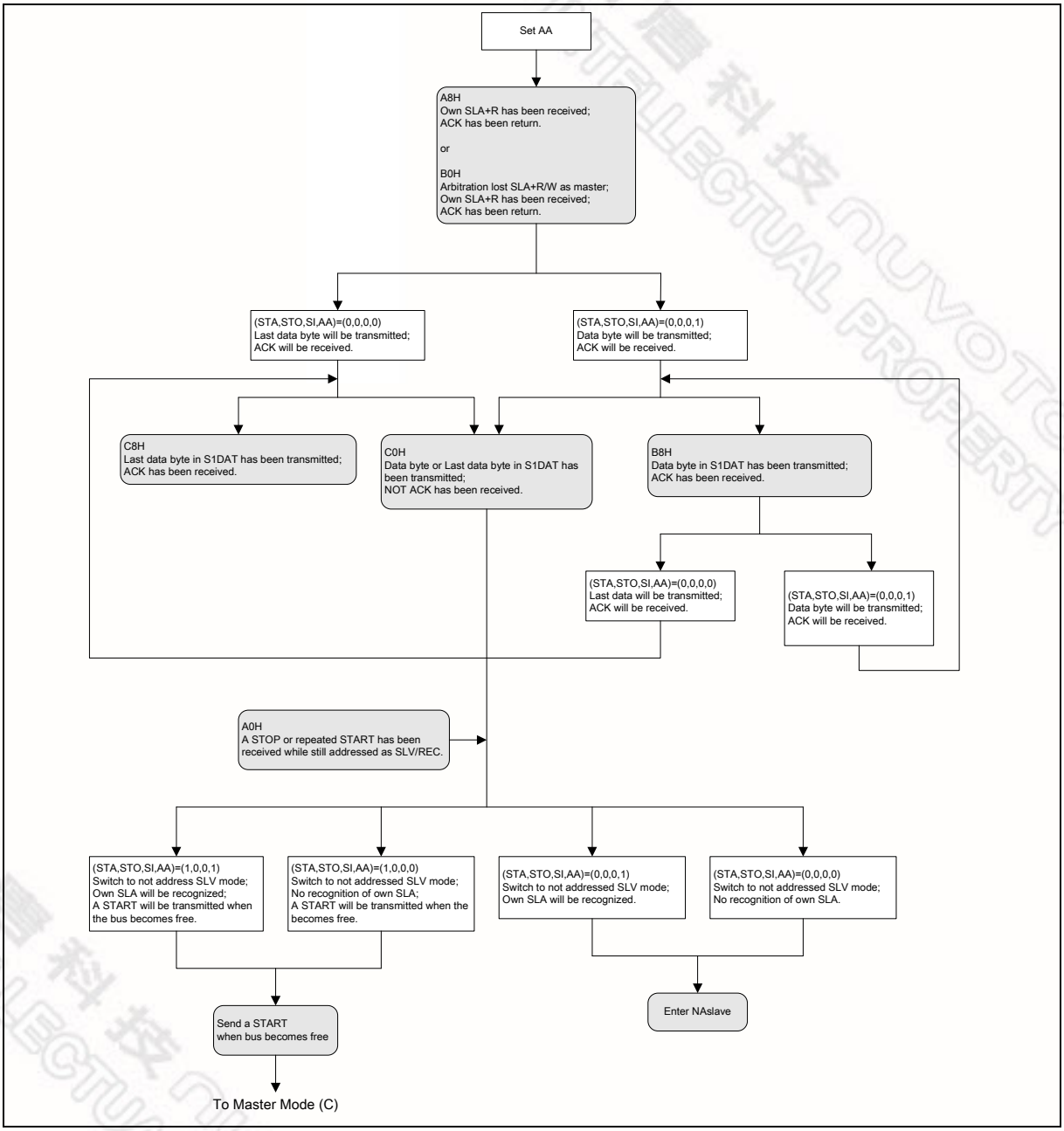


Figure 15-7 Slave Transmitter Mode

15.3.4 Slave/Receiver Mode

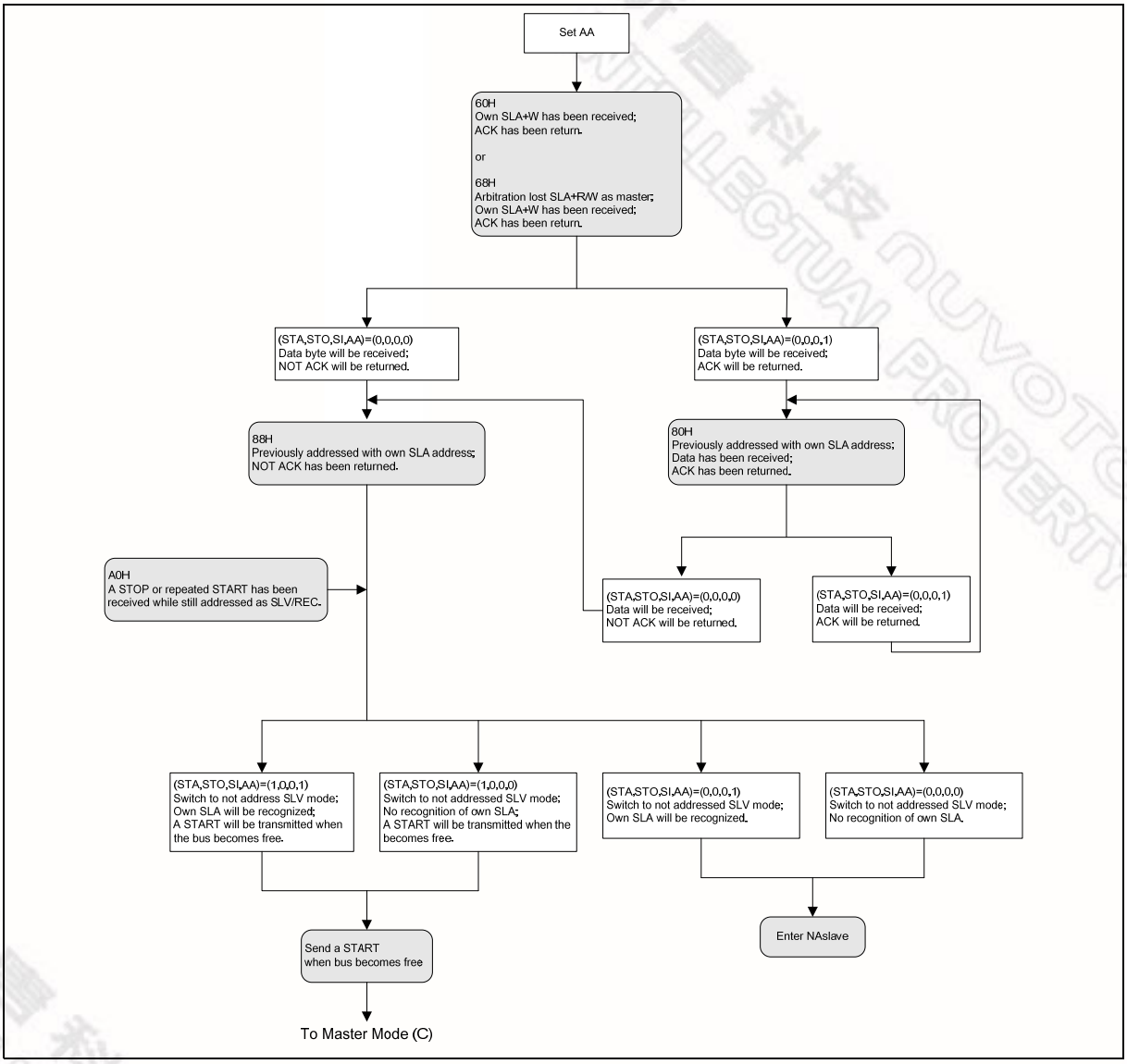


Figure 15-8 Slave Receiver Mode

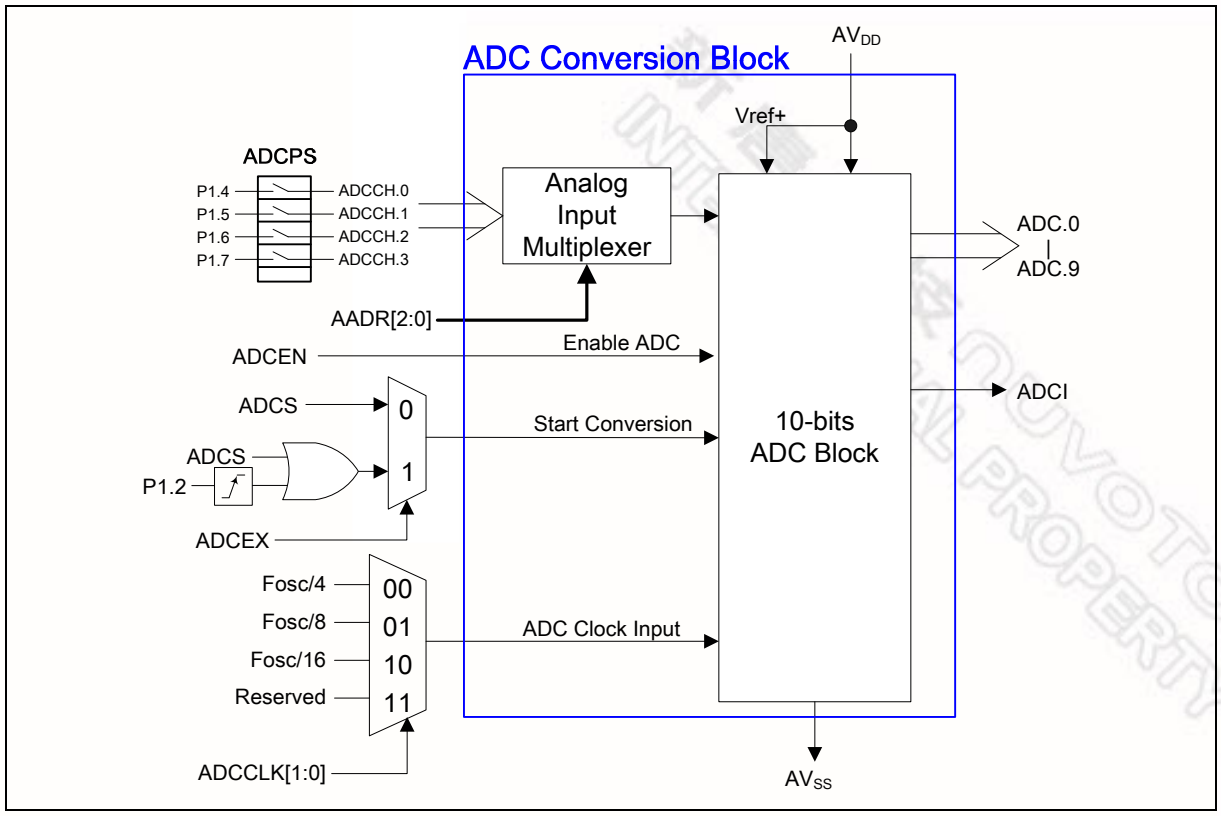


Figure 16-2 ADC Block Diagram

16.2 ADC Resolution and Analog Supply

The ADC circuit has its own supply pins AV_{DD} and AV_{SS}, which are connected to each end of the DAC's resistance-ladder. The ladder has 1023 equally-spaced taps, separated by a resistance of "R". The first tap is located 0.5 x R above AVss, and the last tap is located 0.5 x R below Vref+, giving a total ladder resistance of 1024 x R. This structure ensures that the DAC is monotonic and results in a symmetrical quantization error.

For input voltages between AV_{SS} and [(AV_{SS}) + ½ LSB], the 10-bit result of an A/D conversion will be 0000000000B = 000H. For input voltages between [(AV_{DD}) – 3/2 LSB] and AV_{DD}, the result of a conversion will be 1111111111B = 3FFH. The input voltage (Vin) should be between AV_{DD} and AV_{SS}.

The result can always be calculated from the following formula:

$$\text{Result} = 1024 \times \frac{V_{in}}{AV_{DD}}$$

16.3 ADC Control Registers

ADC Control Register

Bit:	7	6	5	4	3	2	1	0
	ADCEN	-	ADCEX	ADCI	ADCS	AADR2	AADR1	AADR0

Mnemonic: ADCCON Address: C0h

ADCEN Enable A/D Converter Function. Set ADCEN to logic high to enable ADC block.

ADCEX Enable external start control of ADC conversion by a rising edge from P1.2. ADCEX=0: Disable external start. ADCEX=1: Enable external start control.

ADCI A/D Converting Complete/Interrupt Flag. This flag is set when ADC conversion is completed and will cause a hardware interrupt if ADC interrupt is enabled. It is cleared by software only.

ADCS A/D Converting Start. Setting this bit by software starts the conversion of the selected ADC input. ADCS remains high while ADC is converting signal and will be automatically cleared by hardware when ADC conversion is completed.

AADR[2:0] Select and enable analog input channel from ADC0 to ADC3.

The ADCI and ADCS control the ADC conversion as below:

ADCI	ADCS	ADC STATUS
0	0	ADC not busy; A conversion can be started.
0	1	ADC busy; Start of a new conversion is blocked.
1	0	Conversion completed; Start of a new conversion requires ADCI = 0.
1	1	This is an internal temporary state that user can ignore it.

ADC Converter Result Low Register

Bit:	7	6	5	4	3	2	1	0
	ADCLK1	ADCLK0	-	-	-	-	ADC.1	ADC.0

Mnemonic: ADCL Address: C1h

ADCLK[1:0] ADC Clock Frequency Select. The 10-bit ADC needs a clock to drive the converting that the clock frequency may not over 4MHz. ADCLK[1:0] controls the frequency of the clock to ADC block as below table.

ADCLK1	ADCLK0	ADC CLOCK FREQUENCY
0	0	Crystal clock / 4 (Default)
0	1	Crystal clock / 8
1	0	Crystal clock / 16
1	1	Reserved

ADC[1:0] 2 LSB of 10-bit A/D conversion result. The 2 bits are read only.

P43AH, P43AL:

The Base address register for comparator of P4.3. P43AH contains the high-order byte of address, P43AL contains the low-order byte of address.

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 Chip-select Polarity

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

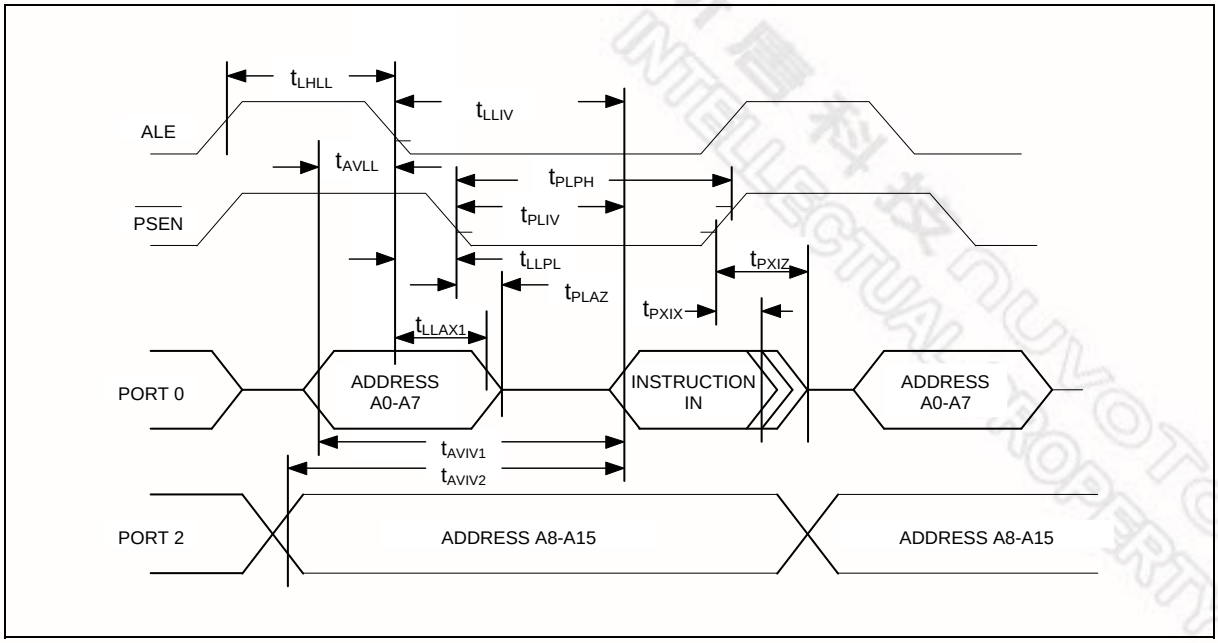
Mnemonic: P4CSIN Address: A2h

P4xINV The active polarity of P4.x when it is set as a chip-select strobe output. High = Active High. Low = Active Low.

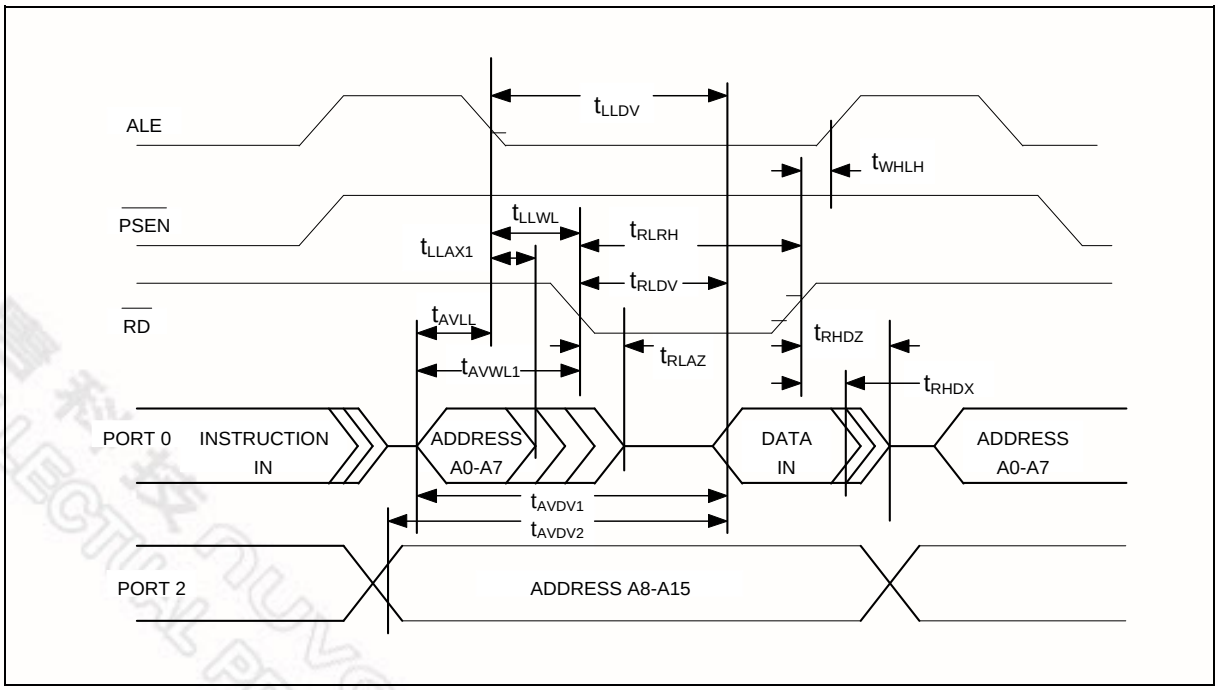
PWDNH Set PWDNH to logic 1 then ALE and PSEN will keep high state, clear this bit to logic 0 then ALE and PSEN will output low during power down mode.

RMWFP Control Read Path of Instruction “Read-Modify-Write”. When this bit is set, the read path of executing “read-modify-write” instruction is from port pin otherwise from SFR.

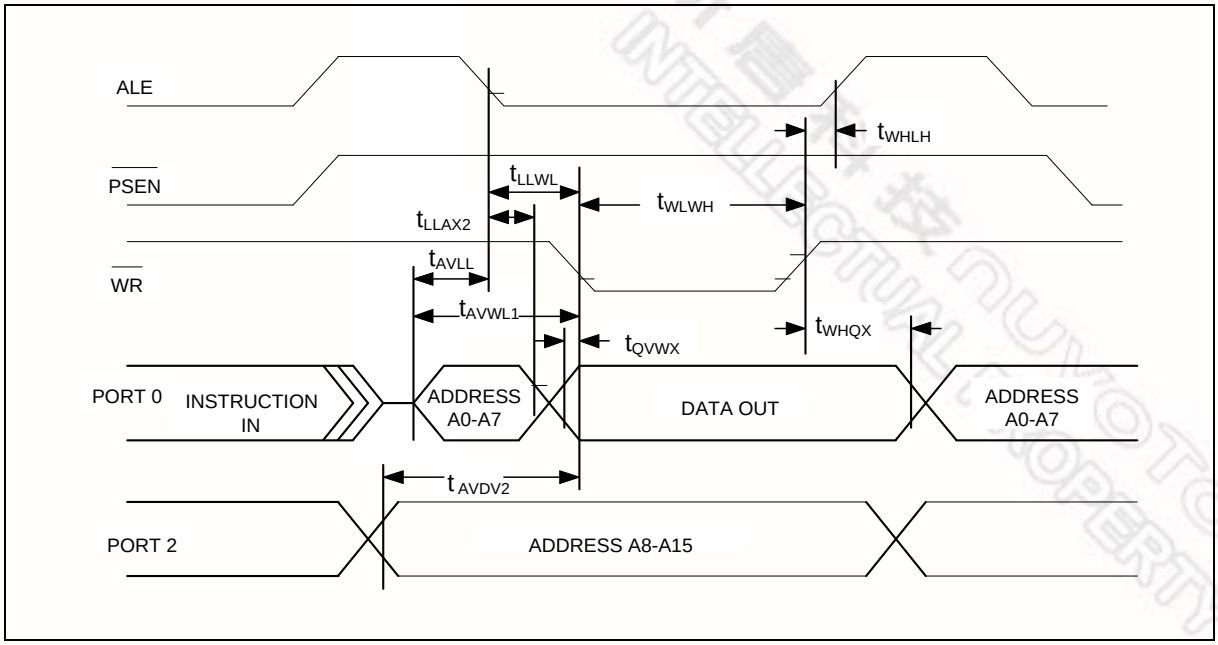
23.6 Program Memory Read Cycle



23.7 Data Memory Read Cycle



23.8 Data Memory Write Cycle



24. Typical Application Circuits

24.1 Expanded External Program Memory and Crystal

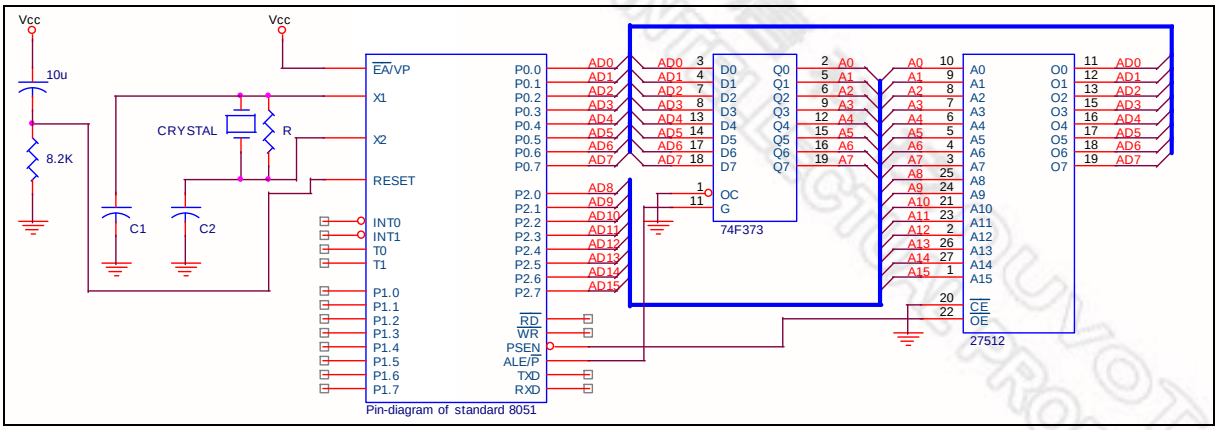


Figure 24-1

CRYSTAL	C1	C2	R
16 MHz	20P	20P	-
24 MHz	12P	12P	-
33 MHz	10P	10P	3.3K
40 MHz	1P	1P	3.3K

The above table shows the reference values for crystal applications.

Note: C1, C2, R components refer to Figure A.

24.2 Expanded External Data Memory and Oscillator

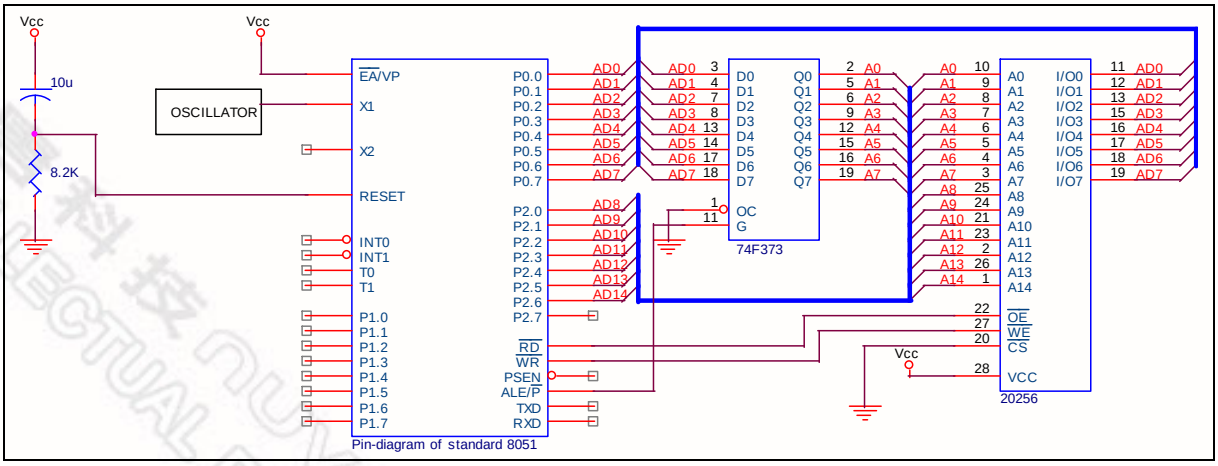


Figure 24-2

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

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 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 APFlash first, then reads the
; new ;* code from external SRAM and program them into APFlash bank. XTAL = 24 MHz
;*************************************************************************************************************
.chip 8052
.RAMCHK OFF
.symbols

CHPCON      EQU     9FH
TA          EQU     C7H
```

```
        MOV  TL0,R6
        MOV  TH0,R7

ERASE_P_4K:
        MOV  SFRCN,#22H     ; SFRCN = 22H, ERASE APFlash0
                            ; SFRCN = A2H, ERASE 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 APFlash0
                            ; SFRCN = 80H, READ 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_APFlashROM

blank_check_error:
        JMP  $

;*************************************************************************
;* RE-PROGRAMMING APFlash BANK
;*************************************************************************
PROGRAM_APFlashROM:
        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 APFlash0
                               ; SFRCN = A1H, PROGRAM APFlash1
        MOV  R6,#9CH           ; SET TIMER FOR PROGRAMMING, ABOUT 50 μS.
        MOV  R7,#FFH
        MOV  TL0,R6
        MOV  TH0,R7

PROG_D_APFlash:
        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_APFlash
        INC  R1
        MOV  SFRAH,R1
        CJNE  R1,#0H,PROG_D_APFlash

;*******************************************************************************
; * VERIFY 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_APFlash:
        MOV  SFRAL,R2          ; SFRAL = LOW ADDRESS
        MOV  TCON,#10H         ; TCON = 10H, TR0 = 1,GO
        MOV  PCON,#01H
        INC  R2
```