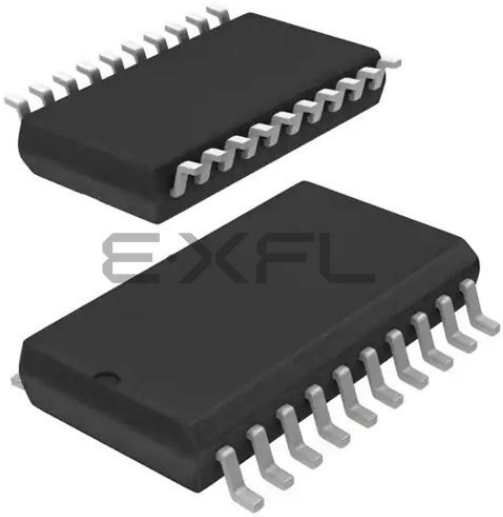

Welcome to E-XFL.COM

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	24MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, WDT
Number of I/O	17
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.4V ~ 5.5V
Data Converters	A/D 7x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOP
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/n79e845asg

- One 16-bit Timer with two channel of input captures
- Watchdog Timer
  - Programmable Watchdog Timer
  - Clock source supported by internal 10kHz ±50% accuracy RC oscillator
- Serial ports (UART, SPI, I^2C)
  - One set of enhanced full duplex UART port with framing error detection and automatic address recognition.
  - One set SPI with master/slave capability.
  - One set I^2C with master/slave capability
- PWM
  - 4 channels 10-bit PWM outputs with one brake/fault input
- KBI
  - 8-keypad interrupt inputs(KBI) with 8 falling/rising/both-edge detection pins selected by software
- ADC
  - 10-bit A/D converter
  - Up to 150 Ksps.(sample per second)
  - 7 analog input channels
- Brown-out Detector
  - 2-level (3.8V/2.7V) BOD detector
  - Supports interrupt and reset options
- POR (Power on Reset)
  - Threshold voltage levels as 2.0V
- Built-in power management.
  - Idle mode
  - Power-down mode with optionally enabled WDT functions
- Development Tools
  - Hardware writer
  - ICP programmer
  - ISP update APROM by UART port

The CONFIG bit DFEN (CONFIG0.0) should be programmed as 0 before accessing the Data Flash block. If DFEN remains its un-programmed value 1, APROM will occupy whole 16K-bytes block in N79E845 DFEN.

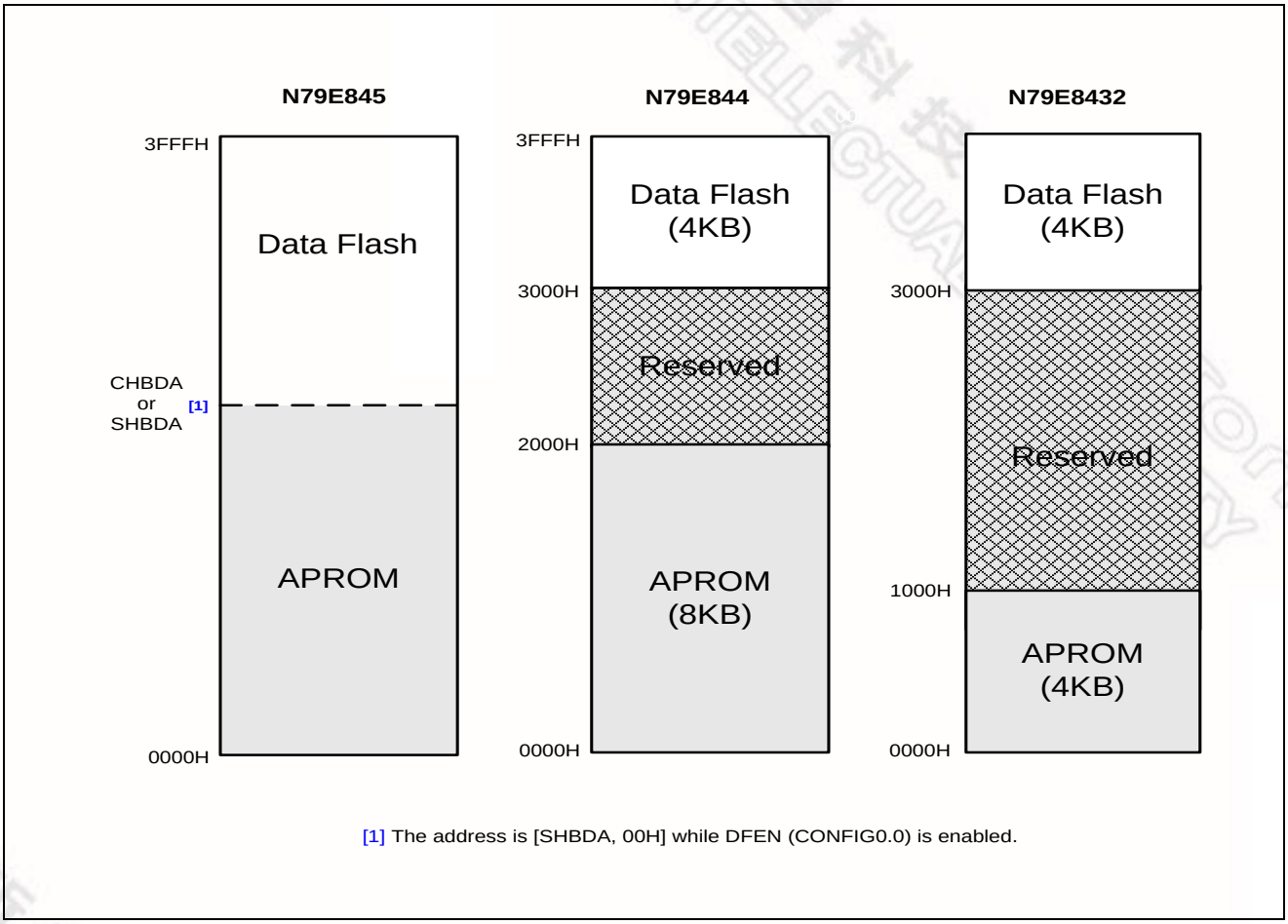

Figure 6–2 N79E845/844/8432 Data Flash

SHBDA – SFR High Byte of Data Flash Starting Address (TA protected, N79E845 Only)

7	6	5	4	3	2	1	0
SHBDA[7:0] [1]							
R/W							

Address: 9CH Reset value: see Table 7–2 N79E845/844/8432 SFR Description and Reset Values

Bit	Name	Description
7:0	SHBDA[7:0]	**SFR high byte of Data Flash starting address** This byte is valid only when DFEN (CONFIG0.0) being 0 condition. It is used to dynamic adjust the starting address of the Data Flash when the application program is executing.

[1] SHBDA is loaded from CONFIG1 after all resets.

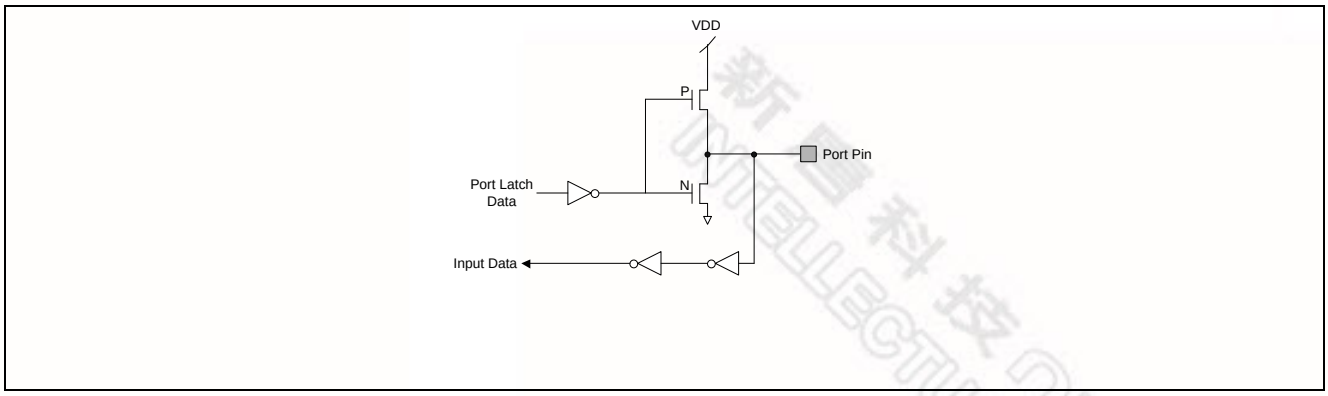

Figure 9-3 Push-Pull Output

9.4 Input Only Configuration

By setting this mode; the ports are only input mode. After setting this mode, the pin will be Hi-Impendence.

P0 – Port 0 (Bit-addressable)

7	6	5	4	3	2	1	0
P07	P06	P05	P04	P03	P02	P01	P00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: 80H Reset value: 1111 1111B

Bit	Name	Description
7:0	P0[7:0]	**Port 0.** Port 0 is an 8-bit quasi bidirectional I/O port.

P1 – Port 1 (Bit-addressable)

7	6	5	4	3	2	1	0
P17	P16	-	P14	P13	P12	P11	P10
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Address: 90H Reset value: 1111 1111B

Bit	Name	Description
7:0	P1[7:0]	**Port 1** These pins are in quasi-bidirectional mode except P1.2 and P1.3 pins. The P1.2 and P1.3 are dedicating open-drain pins for I^2C interface after reset.

Bit	Name	Description
6	SMOD0	**Framing Error Detection Enable** 0 = Framing error detection is disabled. SM0/FE (SCON.7) bit is used as SM0 as standard 80C51 function. 1 = Framing error detection is enabled. SM0/FE bit is used as frame error (FE) status flag.

SBUF – Serial Data Buffer

7	6	5	4	3	2	1	0
SBUF[7:0]							
R/W							

Address: 99H Reset value: 0000 0000B

Bit	Name	Description
7:0	SBUF[7:0]	**Serial Data Buffer** This byte actually consists two separate registers. One is the receiving resister, and the other is the transmitting buffer. When data is moved to SBUF, it goes to the transmitting buffer and is shifted for serial transmission. When data is moved from SBUF, it comes from the receiving buffer. The transmission is initiated through moving a byte to SBUF.

12.1 Mode 0

Mode 0 provides synchronous communication with external devices. Serial data enters and exits through RXD pin. TXD outputs the shift clock. 8 bits are transmitted or received. Mode 0 therefore provides half-duplex communication because the transmitting or receiving data is via the same data line RXD. The baud rate is enhanced to be selected as $F_{SYS}/12$ if SM2 (SCON.5) is 0 or as $F_{SYS}/4$ if SM2 is 1. Note that whenever transmitting or receiving, the serial clock is always generated by the microcontroller. Thus any device on the serial port in Mode 0 should accept the microcontroller as the Master. Figure 12–1 shows a simplified functional diagram of the serial port in Mode 0 and associated timing.

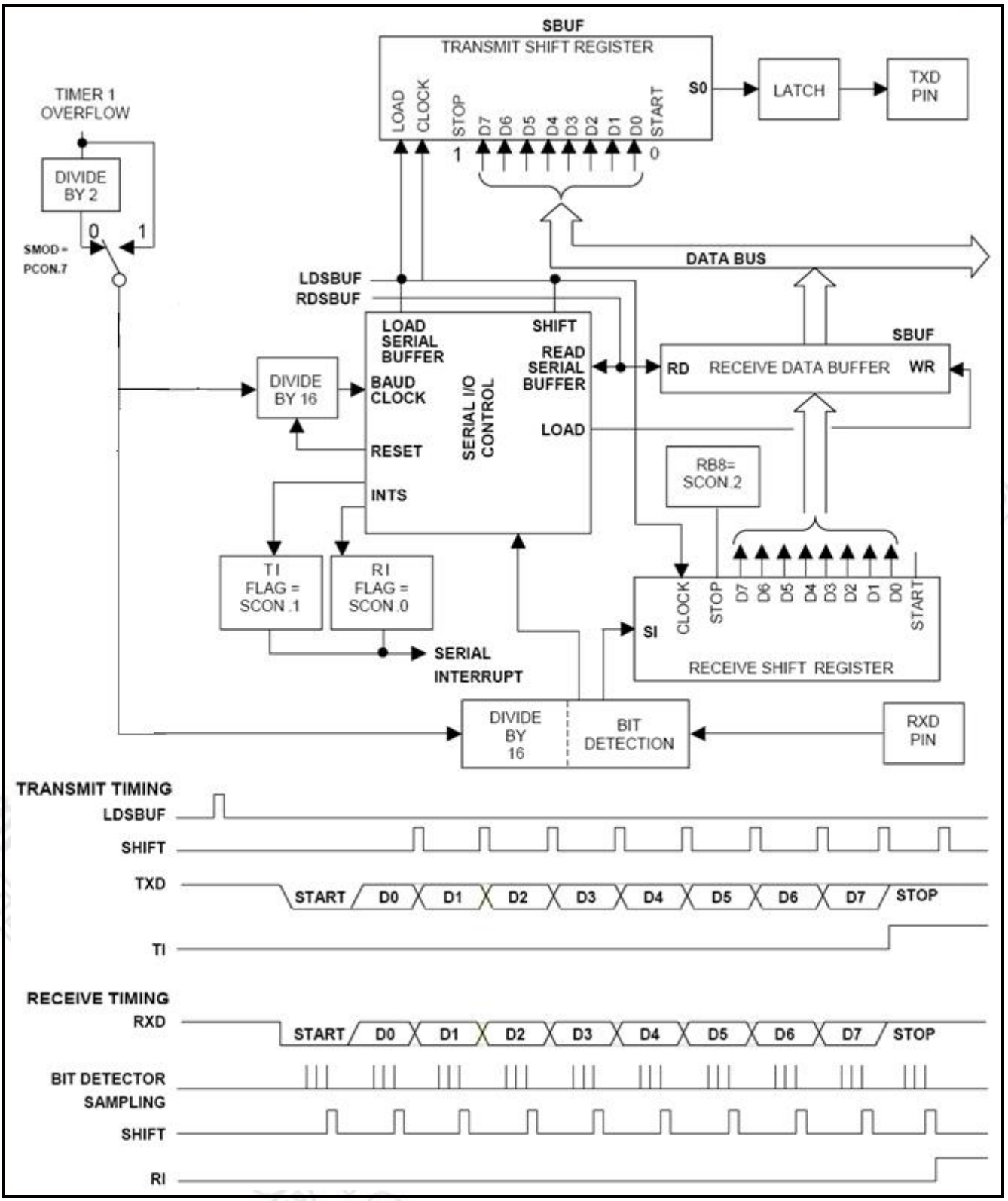

Figure 12–3 Serial Port Mode 1 Function Block and Timing Diagram

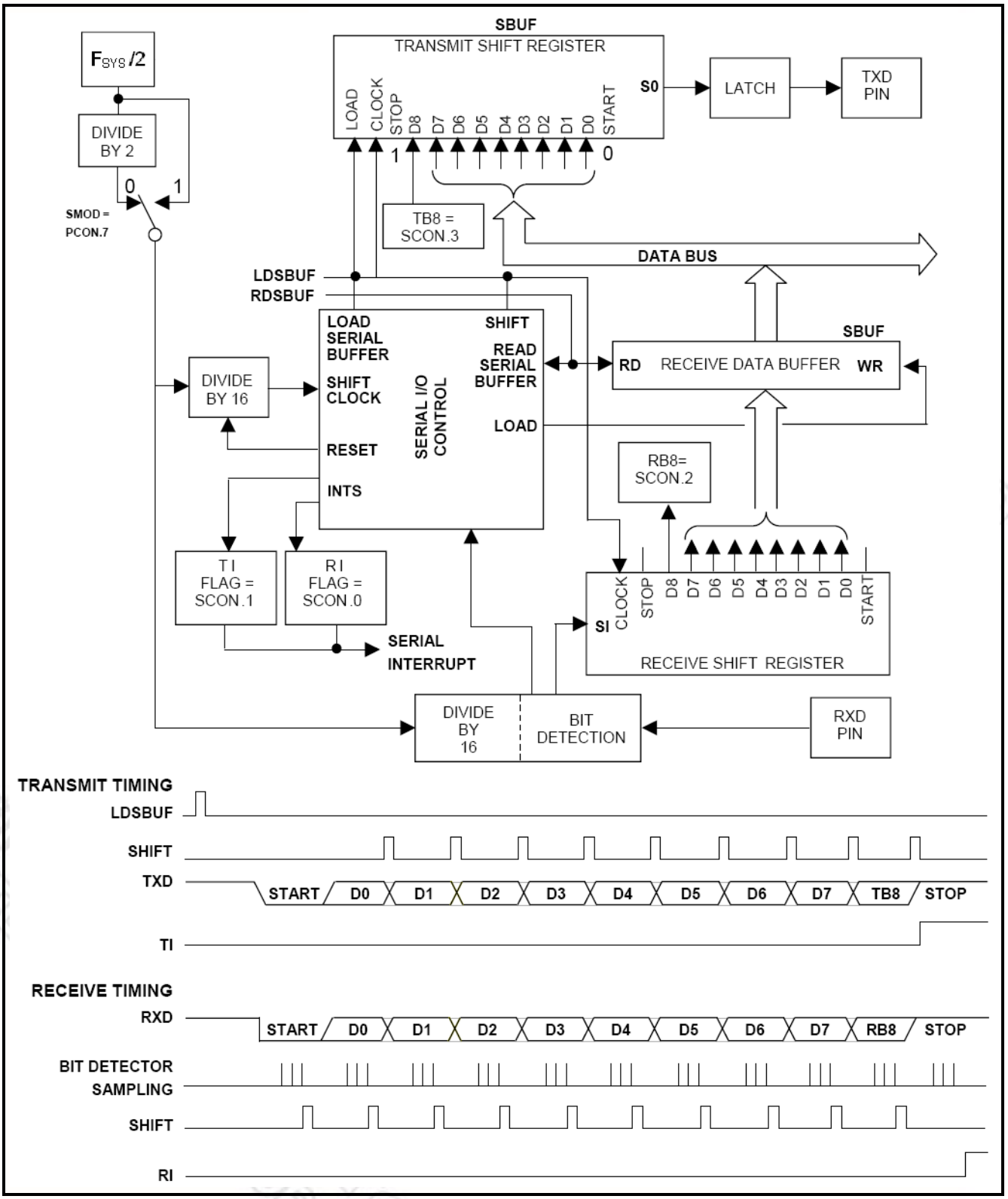

Figure 12–4 Serial Port Mode 2 Function Block and Timing Diagram

```
Given = 110000XXb
```

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 11100110b. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 11100101b. Slave 2 requires that bit 2 = 0 and its unique address is 11100011b. To select Slaves 0 and 1 and exclude Slave 2 use address 11100100b, since it is necessary to make bit 2 = 1 to exclude slave 2. The “Broadcast” address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as “don’t care”. In most cases, interpreting the “don’t care” as ones, the broadcast address will be FFH.

On reset, SADDR and SADEN are initialized to 00H. This produces a “Given” address of all “don’t care” as well as a “Broadcast” address of all XXXXXXXXb (all “don’t care” bits). This effectively disables the automatic addressing mode and allows the microcontroller to use standard UART drivers which do not make use of this feature.

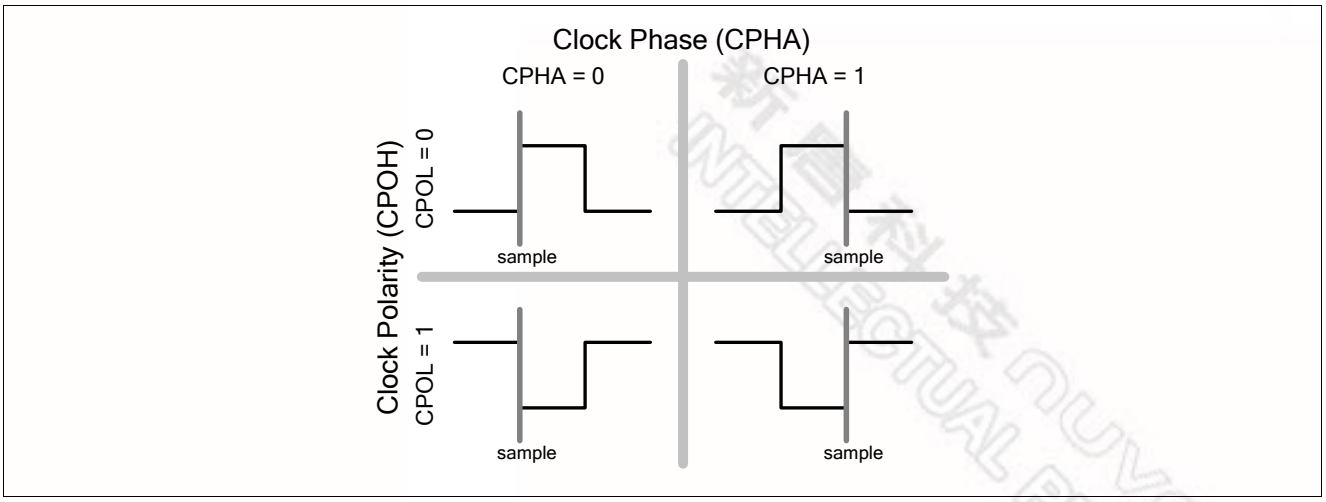

Figure 13–4 SPI Clock Format

In SPI, a Master device always initiates the transfer. If SPI is selected as Master mode (MSTR = 1) and enabled (SPIEN = 1), writing to the SPI data register (SPDR) by the Master device starts the SPI clock and data transfer. After shifting one byte out and receiving one byte in, the SPI clock stops and SPIF (SPSR.7) in both Master and Slave are set. If SPI interrupt enable bit ESPI (EIE.6) is set 1 and global interrupt is enabled (EA = 1), the interrupt service routine (ISR) of SPI will be executed.

Concerning the Slave mode, the $\overline{SS}$ signal needs to be taken care. As shown in Figure 13–4 SPI Clock Format, when CPHA = 0, the first SPCLK edge is the sampling strobe of MSB (for an example of LSBFE = 0, MSB first). Therefore, the Slave should shift its MSB data before the first SPCLK edge. The falling edge of $\overline{SS}$ is used for preparing the MSB on MISO line. The $\overline{SS}$ pin therefore should toggle high and then low between each successive serial byte. Furthermore, if the slave writes data to the SPI data register (SPDR) while $\overline{SS}$ is low, a write collision error occurs.

When CPHA = 1, the sampling edge thus locates on the second edge of SPCLK clock. The Slave uses the first SPCLK clock to shift MSB out rather than the $\overline{SS}$ falling edge. Therefore, the $\overline{SS}$ line can remain low between successive transfers. This format may be preferred in systems having single fixed Master and single fixed Slave. The $\overline{SS}$ line of the unique Slave device can be tied to V_{SS} as long as only CPHA = 1 clock mode is used.

Note: The SPI should be configured before it is enabled (SPIEN = 1), or a change of LSBFE, MSTR, CPOL, CPHA and SPR[1:0] will abort a transmission in progress and force the SPI system into idle state. Prior to any configuration bit changed, SPIEN should be disabled first.

14 Keyboard Interrupt (KBI)

The N79E845/844/8432 provides the 8 keyboard interrupt function to detect keypad status which key is acted, and allow a single interrupt to be generated when any key is pressed on a keyboard or keypad connected to specific pins of the N79E845/844/8432, as shown in the following figure. This interrupt may be used to wake up the CPU from Idle or Power-down mode, after chip is in Power-down or Idle mode.

Keyboard function is supported through by Port 0. It can allow any or all pins of Port 0 to be enabled to cause this interrupt. Port pins are enabled by the setting of bits of KBI0 ~ KBI7 in the KBI register, as shown in the following figure. The Keyboard Interrupt Flag, KBIF[7:0] in the KBIF(EAH), is set when any enabled pin is triggered while the KBI interrupt function is active, an interrupt will be generated if it has been enabled. The KBIF[7:0] bit is set by hardware and should be cleared by software. To determine which key was pressed, the KBI will allow the interrupt service routine to poll port 0.

KBI supports four triggered conditions — low level, falling edge, rising edge and either rising or falling edge detection. The triggered condition of each port pin is individually controlled by two bits KBLS1(ECH).x and KBLS0(EBH).x where x is 0 to 7. After Trigger occurs and two machines pass, KBIF assert.

KBI is generally used to detect an edge transient from peripheral devices like keyboard or keypad. During idle state, the system prefers to enter Power-Down mode to minimize power consumption and waits for event trigger. The N79E845/844/8432 supports KBI interrupt waking up MCU from Power down. Note that if KBI is selected as any of edge trigger mode, restrictions should be followed to make Power down woken up valid. For a falling edge waking up, pin state should be high at the moment of entering Power-Down mode. Respectively, pin state should be low for a rising edge waking up.

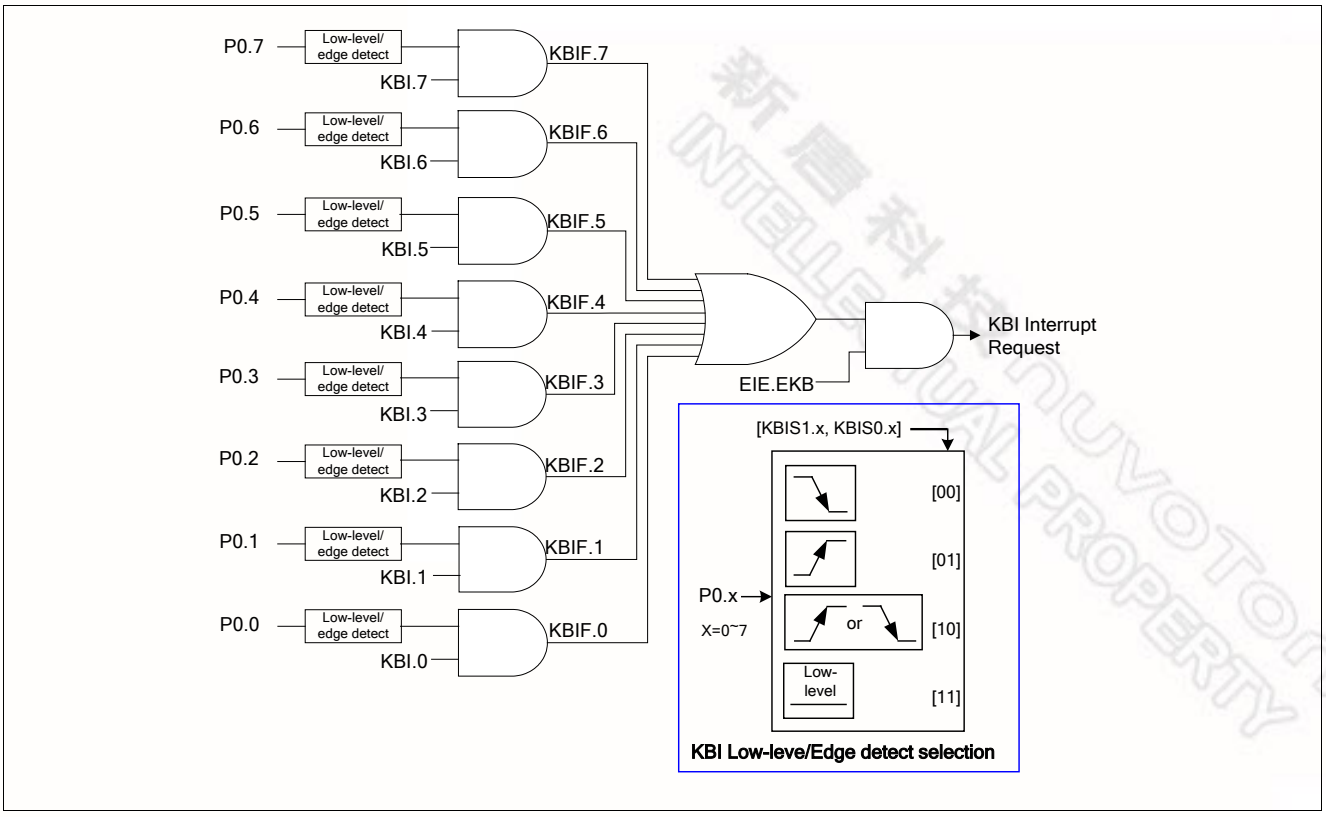

Figure 14-1 Keyboard Interrupt Detection

16 Inter-Integrated Circuit (I^2C)

16.1 Features

The Inter-Integrated Circuit (I^2C) bus serves as a serial interface between the microcontroller and the I^2C devices such as EEPROM, LCD module, and so on. The I^2C bus used two wires design (a serial data line SDA and a serial clock line SCL) to transfer information between devices.

The I^2C bus uses bidirectional data transfer between masters and slaves. There is no central master and the multi-master system is allowed by arbitration between simultaneously transmitting masters. The serial clock synchronization allows devices with different bit rates to communicate via one serial bus. The I^2C bus supports four transfer modes including master transmitter mode, master receiver mode, slave receiver mode, and slave transmitter mode. The I^2C interface only supports 7-bit addressing mode and General Call can be accepted. The I^2C can meet both standard (up to 100kbps) and fast (up to 400kbps) speeds.

16.2 Functional Description

For the bidirectional transfer operation, the SDA and SCL pins should be connected to open-drain pads. This implements a wired-AND function which is essential to the operation of the interface. A low level on a I^2C bus line is generated when one or more I^2C devices output a “0”. A high level is generated when all I^2C devices output “1”, allowing the pull-up resistors to pull the line high.

In the N79E845/844/8432, the user should set output latches of P1.2 and P1.3. as logic 1 before enabling the I^2C function by setting I2CEN (I2CON.6). The P1.2 and P1.3 are configured as the open-drain I/O once the I^2C function is enabled. The P1M2 and P1M1 will be also re-configured. The Schmitt trigger input buffer is strongly recommended to be enabled by setting P1S for improved glitch suppression.

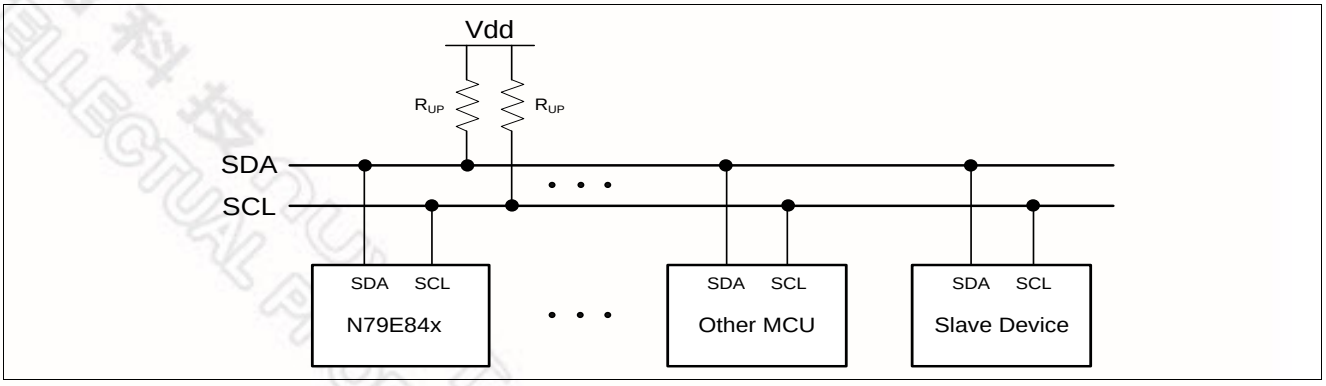

Figure 16–1 I^2C Bus Interconnection

Bit	Name	Description
1	PWM1I	0 = PWM1 output is non-inverted. 1 = PWM1 output is inverted.
0	PWM0I	0 = PWM0 output is non-inverted. 1 = PWM0 output is inverted.

The fact that the transfer from the Counter and PWMn register to the working registers(10-bit Counter and Compare register) only occurs when there is an underflow in the counter results in the need for the user's program to observe the following precautions. If PWMCON0 is written with Load set without Run being enabled the transfer will never take place. Thus if a subsequent write sets Run without Load the compare and counter values will not be those expected. If Load and Run are set, and prior to underflow there is a subsequent LOAD of PWMCON0 which sets Run but not Load, the LOAD will never take place. Again the compare and counter values that existed prior to the update attempt will be used.

As outlined above the Load bit can be polled to determine when the LOAD occurs. Unless there is a compelling reason to do otherwise, it is recommended that both PWMRUN (PWMCON0.7), and Load (PWMCON0.6) be set when PWMCON0 is written.

When the PWMRUN bit, PWMCON0.7 is cleared the PWM outputs take on the state they had just prior to the bit being cleared. In general, this state is not known. To place the outputs in a known state when PWMRUN is cleared the Compare registers can be written to either the "always 1" or "always 0" so the output will have the output desired when the counter is halted. After this PWMCON0 should be written with the Load and Run bits are enabled. After this is done PWMCON0 to polled to find that the Load or CF flag has taken place. Once the LOAD has occurred the Run bit in PWMCON0 can be cleared. The outputs will retain the state they had just prior to the Run being cleared. If the Brake pin (see discussion below in section concerning the operation of PWMCON1) is not used to control the brake function, the "Brake when not running" function can be used to cause the outputs to have a given state when the PWM is halted. This approach should be used only in time critical situations when there is not sufficient time to use the approach outlined above since going from the Brake state to run without causing an undefined state on the outputs is not straightforward. A discussion on this topic is included in the PWMCON1 section.

PWMCON1 – PWM Control Register 1

7	6	5	4	3	2	1	0
BKCH	BKPS	BPEN	BKEN	PWM3B	PWM2B	PWM1B	PWM0B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: DFH Reset value: 0000 0000B

Bit	Name	Description
7	BKCH	See the following table (when BKEN is set).
6	BKPS	0 = Brake is asserted if P0.2 is low. 1 = Brake is asserted if P0.2 is high
5	BPEN	See the following table (when BKEN is set).

not by another interrupt of the same or lower priority. The highest priority interrupt service cannot be interrupted by any other interrupt source. So, if two requests of different priority levels are received simultaneously, the request of higher priority level is serviced.

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

The following table summarizes the interrupt sources, flag bits, vector addresses, enable bits, priority bits, arbitration ranking, and whether each interrupt may wake up the CPU from Power-down mode.

Bit	Name	Description
1	PKB	1 = Set interrupt priority of Keypad as higher priority level.
0	PI2	1 = Set interrupt priority of I^2C as higher priority level.

EIPH – Interrupt High Priority-1 Register

7	6	5	4	3	2	1	0
PT2H	PSPIH	PPWMH	PWDIH	-	-	PKBH	PI2H
R/W	R/W	R/W	R/W	-	-	R/W	R/W

Address: F7H Reset value: 0000 0000B

Bit	Name	Description
7	PT2H	1 = Set interrupt high priority of Timer 2 as the highest priority level.
6	PSPIH	1 = Set interrupt high priority of SPI as the highest priority level.
5	PPWMH	1 = Set interrupt high priority of PWM's external brake pin as the highest priority level.
4	PWDIH	1 = Set interrupt high priority of Watchdog as the highest priority level.
3:2	-	Reserved
1	PKBH	1 = Set interrupt high priority of Keypad as the highest priority level.
0	PI2H	1 = Set interrupt high priority of I^2C as the highest priority level.

TCON – Timer 0 and 1 Control (Bit-addressable)

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: 88H Reset value: 0000 0000B

Bit	Name	Description
3	IE1	**External Interrupt 1 Edge Flag** This flag is set via hardware when an edge/level of type defined by IT1 is detected. If IT1 = 1, this bit will remain set until it is cleared via software or at the beginning of the External Interrupt 1 service routine. If IT1 = 0, this flag is the inverse of the $\overline{INT1}$ input signal's logic level.

```
MOV    ISPAL,#??H
MOV    ISPFD,#??H          ;fill data to be programmed
CALL   Trigger_ISP

CALL   Disable_ISP
```

CONFIG Read (target address in CONFIG area)

```
CALL   Enable_ISP
MOV    ISPCN,#11000000b    ;select “CONFIG Read” mode, (A17,A16)=(1,1) for CONFIG
MOV    ISPAH,#00H          ; fill byte address, 0000H/0001H/0002H/0003H for CONFIG0/1/2/3,
                           ;respectively
MOV    ISPAL,#??H
CALL   Trigger_ISP
MOV    A,ISPFD             ;now, ISPFD contains the CONFIG data, move to ACC for further
                           ;use
CALL   Disable_ISP
```

Bit	Name	Description
4	BORST	**BOD Reset Enable** This bit decides if a BOD reset is caused after a BOD event. 0 = Disable BOD reset when V_{DD} drops below V_{BOD} or V_{DD} rises above V_{BOD}. Chip will assert BOF when V_{DD} drops below V_{BOD}. 1 = Enable BOD reset when V_{DD} drops below V_{BOD} or V_{DD} rises above V_{BOD}.
3	BOF	**BOD Flag** This flag will be set as logic 1 via hardware after a V_{DD} dropping below or rising above V_{BOD} event occurs. If both EBOD (IE.5) and EA (IE.7) are set, a BOD interrupt requirement will be generated. This bit should be cleared via software.
2	-	**It should be set to logic 0.**
1	-	**Reserved**
0	-	**Reserved**

Note: If BOF is 1 after chip reset, it is strongly recommended to initialize the user program by clearing BOF.

24.2 BOD Reset

BOD detection circuit is for monitoring the V_{DD} level during execution. When V_{DD} drops to the selected BOD trigger level (V_{BOD}) or V_{DD} rises over V_{BOD}, the BOD detection logic will reset the CPU if BORST (PMCR.4) setting 1.

PMCR – Power Monitoring Control (TA Protected)

7	6	5	4	3	2	1	0
BODEN	BOV	-	BORST	BOF	-	-	-
R/W	R/W	-	R/W	R/W	-	-	-

Address: A3H Reset value: see Table 7–2 N79E845/844/8432 SFR Description and Reset Values

Bit	Name	Description
7	BODEN	**BOD-detect Function Control** BODEN is initialized by inverted CBODEN (CONFIG2, bit-7) at any resets. 1 = Enable BOD detection. 0 = Disable BOD detection.
6	BOV	**BOD Voltage Select Bits** BOD are initialized at reset with the value of bits CBOV in CONFIG3-bits BOD Voltage Select bits:
5	-	**Reserved**
4	BORST	**BOD Reset Enable** This bit decides if a BOD reset is caused after a BOD event. 0 = Disable BOD reset when V_{DD} drops below V_{BOD} or V_{DD} rises above V_{BOD}. Chip will assert BOF when V_{DD} drops below V_{BOD}. 1 = Enable BOD reset when V_{DD} drops below V_{BOD} or V_{DD} rises above V_{BOD}.
3	BOF	**BOD Flag** This flag will be set as logic 1 via hardware after a V_{DD} dropping below or rising above V_{BOD} event occurs. If both EBOD (IE.5) and EA (IE.7) are set, a BOD interrupt requirement will be generated. This bit should be cleared via software.
2	-	**It should be set to logic 0.**
1	-	**Reserved**
0	-	**Reserved**

CONFIG-bits CBOV	SFR BOV	BOD Voltage
1	0	Enable BOD= 2.7V
0	1	Enable BOD= 3.8V

26 Instruction Sets

The N79E845/844/8432 executes all the instructions of the standard 8051 family. All instructions are coded within an 8-bit field called an OPCODE. This single byte should be fetched from Program Memory. The OPCODE is decoded by the CPU. It determines what action the microcontroller will take and whether more operation data is needed from memory. If no other data is needed, then only one byte was required. Thus the instruction is called a one byte instruction. In some cases, more data is needed, which will be two or three byte instructions.

Table 26–1 lists all instructions in details. The note of the instruction sets and addressing modes are shown below.

Rn (n = 0~7)	Register R0~R7 of the currently selected Register Bank.
	Direct 8-bit internal data location's address. This could be an internal data RAM location (0~127) or a SFR (e.g. I/O port, control register, status register, etc.) (128~255).
@Ri (i = 0, 1)	8-bit internal data RAM location (0~255) addressed indirectly through register R0 or R1.
#data	8-bit constant included in the instruction.
#data16	16-bit constant included in the instruction.
addr16	16-bit destination address. Used by LCALL and LJMP. A branch can be any within the 16 Kbytes Program Memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 Kbytes page of Program Memory as the first byte of the following instruction.
rel	Signed (2's complement) 8-bit offset byte. Used by SJMP and all conditional branches. The range is -128 to +127 bytes relative to first byte of the following instruction.
bit	Direct addressed bit in internal data RAM or SFR.

Table 26–1 Instruction Set for the N79E845/844/8432

Instruction	OPCODE	Bytes	Clock Cycles	N79E845/844/8432 vs. Tradition 80C51 Speed Ratio
NOP	00	1	4	3.0
ADD A, Rn	28~2F	1	4	3.0
ADD A, @Ri	26, 27	1	4	3.0
ADD A, direct	25	2	8	1.5
ADD A, #data	24	2	8	1.5
ADDC A, Rn	38~3F	1	4	3.0
ADDC A, @Ri	36, 37	1	4	3.0
ADDC A, direct	35	2	8	1.5
ADDC A, #data	34	2	8	1.5
SUBB A, Rn	98~9F	1	4	3.0

Table 26–1 Instruction Set for the N79E845/844/8432

Instruction		OPCODE	Bytes	Clock Cycles	N79E845/844/8432 vs. Tradition 80C51 Speed Ratio
SUBB	A, @Ri	96, 97	1	4	3.0
SUBB	A, direct	95	2	8	1.5
SUBB	A, #data	94	2	8	1.5
INC	A	04	1	4	3.0
INC	Rn	08~0F	1	4	3.0
INC	@Ri	06, 07	1	4	3.0
INC	direct	05	2	8	1.5
INC	DPTR	A3	1	8	3.0
DEC	A	14	1	4	3.0
DEC	Rn	18~1F	1	4	3.0
DEC	@Ri	16, 17	1	4	3.0
DEC	direct	15	2	8	1.5
DEC	DPTR	A5	1	8	-
MUL	AB	A4	1	20	2.4
DIV	AB	84	1	20	2.4
DA	A	D4	1	4	3.0
ANL	A, Rn	58~5F	1	4	3.0
ANL	A, @Ri	56, 57	1	4	3.0
ANL	A, direct	55	2	8	1.5
ANL	A, #data	54	2	8	1.5
ANL	direct, A	52	2	8	1.5
ANL	direct, #data	53	3	12	2.0
ORL	A, Rn	48~4F	1	4	3.0
ORL	A, @Ri	46, 47	1	4	3.0
ORL	A, direct	45	2	8	1.5
ORL	A, #data	44	2	8	1.5
ORL	direct, A	42	2	8	1.5
ORL	direct, #data	43	3	12	2.0
XRL	A, Rn	68~6F	1	4	3.0
XRL	A, @Ri	66, 67	1	4	3.0
XRL	A, direct	65	2	8	1.5
XRL	A, #data	64	2	8	1.5
XRL	direct, A	62	2	8	1.5
XRL	direct, #data	63	3	12	2.0
CLR	A	E4	1	4	3.0
CPL	A	F4	1	4	3.0
RL	A	23	1	4	3.0
RLC	A	33	1	4	3.0
RR	A	03	1	4	3.0
RRC	A	13	1	4	3.0

Table 28–3 DC Characteristics

(V_{DD}–V_{SS} = 2.4~5.5V, TA = -40~85°C, unless otherwise specified.)

Sym	Parameter	Test Conditions	MIN	TYP	MAX	Unit
V_{OH1}	Output High Voltage (general purpose I/O, push-pull)	V_{DD}=4.5V I_{OH}= **-28.0**mA[3], [4]	2.4			V
		V_{DD}=3.0V I_{OH}= **-7**mA[3], [4]	2.4			V
		V_{DD}=2.4V I_{OH}= **-3.5**mA[3], [4]	2.0			V
I_{IL}	Logical 0 Input Current (general purpose I/O, quasi bi-direction)	V_{DD}=5.5V, V_{IN}=0.4V		**-40 at 5.5V**	-50	µA
I_{TL}	Logical 1 to 0 Transition Current (general purpose I/O, quasi bi-direction)	V_{DD}=5.5V, V_{IN}=2.0V[2]		**-550 at 5.5V**	-650	µA
I_{LI}	Input Leakage Current (general purpose I/O, open-drain or input only)	0 < V_{IN} < V_{DD}		**<1**	±10	µA
I_{OP}	OP Current (Active mode[5])	XTAL 12MHz, V_{DD}=5.0V		**3.1**		mA
		XTAL 24MHz, V_{DD}=5.5V		**4.3**		mA
		XTAL 12MHz, V_{DD}=3.3V		**1.7**		mA
		XTAL 24MHz, V_{DD}=3.3V		**3.2**		mA
		Internal 22.1184MHz,V_{DD}=5V		**2.3**		mA
		Internal 22.1184MHz,V_{DD}=3.3V		**2.2**		mA
I_{IDLE}	IDLE Current	XTAL 12MHz, V_{DD}=5.0V		**2.7**		mA
		XTAL 24MHz, V_{DD}=5.5V		**3.7**		mA