Welcome to E-XFL.COM

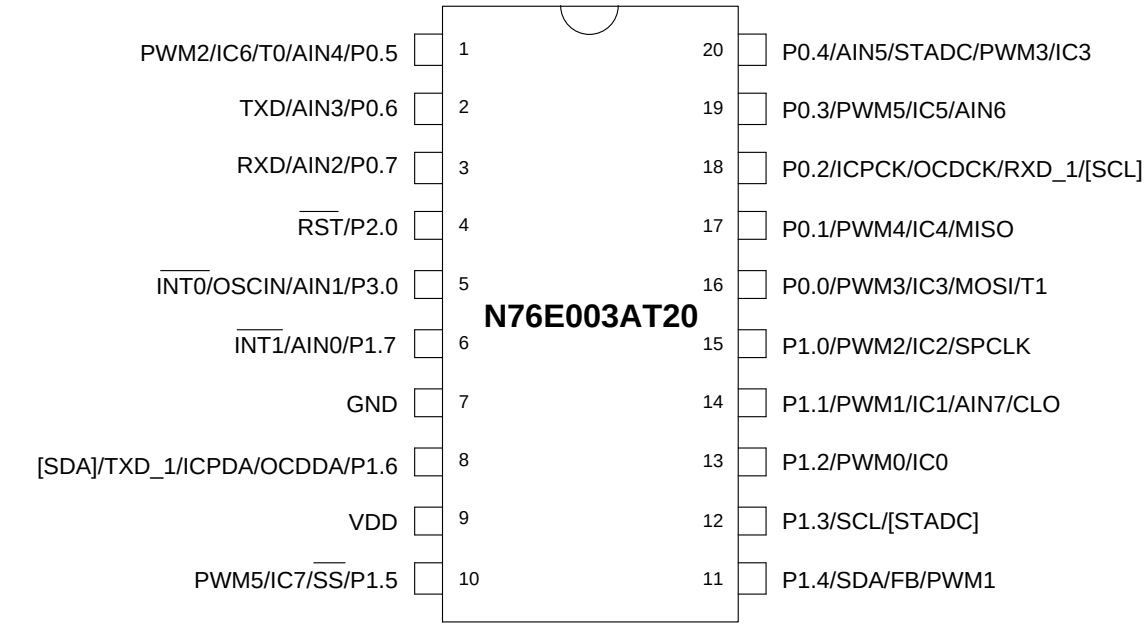
What is "Embedded - Microcontrollers"?

"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	-
Core Size	-
Speed	-
Connectivity	-
Peripherals	-
Number of I/O	-
Program Memory Size	-
Program Memory Type	-
EEPROM Size	-
RAM Size	-
Voltage - Supply (Vcc/Vdd)	-
Data Converters	-
Oscillator Type	-
Operating Temperature	-
Mounting Type	-
Package / Case	-
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/n76e003as20

4. PIN CONFIGURATION



1. [ ] alternate function remapping option (if the same alternate function is shown twice, it indicates an exclusive choice not a duplication of the function).

Figure 4.1. Pin Assignment of TSSOP-20 Package

P1SR – Port 1 Slew Rate

7	6	5	4	3	2	1	0
P1SR.7	P1SR.6	P1SR.5	P1SR.4	P1SR.3	P1SR.2	P1SR.1	P1SR.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: B4H, Page: 1 Reset value: 0000 0000b

Bit	Name	Description
n	P1SR.n	**P1.n slew rate** 0 = P1.n normal output slew rate. 1 = P1.n high-speed output slew rate.

P2S – P20 Setting and Timer01 Output Enable

7	6	5	4	3	2	1	0
P20UP	-	-	-	T1OE	T0OE	-	P2S.0
R/W	-	-	-	R/W	R/W	-	R/W

Address: B5H Reset value: 0000 0000b

Bit	Name	Description
7	P20UP	**P2.0 pull-up enable** 0 = P2.0 pull-up Disabled. 1 = P2.0 pull-up Enabled. This bit is valid only when RPD (CONFIG0.2) is programmed as 0. When selecting as a $\overline{\text{RST}}$ pin, the pull-up is always enabled.
3	T1OE	**Timer 1 output enable** 0 = Timer 1 output Disabled. 1 = Timer 1 output Enabled from T1 pin. Note that Timer 1 output should be enabled only when operating in its “Timer” mode.
2	T0OE	**Timer 0 output enable** 0 = Timer 0 output Disabled. 1 = Timer 0 output Enabled from T0 pin. Note that Timer 0 output should be enabled only when operating in its “Timer” mode.
0	P2S.0	**P2.0 Schmitt triggered input** 0 = TTL level input of P2.0. 1 = Schmitt triggered input of P2.0.

IPH – Interrupt Priority High[2]

7	6	5	4	3	2	1	0
-	PADCH	PBODH	PSH	PT1H	PX1H	PT0H	PX0H
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: B7H, Page0 Reset value: 0000 0000b

Bit	Name	Description
6	PADC	**ADC interrupt priority high bit**
5	PBOD	**Brown-out detection interrupt priority high bit**
4	PSH	**Serial port 0 interrupt priority high bit**

I2DAT – I²C Data

7	6	5	4	3	2	1	0
I2DAT[7:0]							
R/W							

Address: BCH Reset value: 0000 0000b

Bit	Name	Description
7:0	I2DAT[7:0]	**I²C data** I2DAT contains a byte of the I²C data to be transmitted or a byte, which has just received. Data in I2DAT remains as long as SI is logic 1. The result of reading or writing I2DAT during I²C transceiving progress is unpredicted. While data in I2DAT is shifted out, data on the bus is simultaneously being shifted in to update I2DAT. I2DAT always shows the last byte that presented on the I²C bus. Thus the event of lost arbitration, the original value of I2DAT changes after the transaction.

I2STAT – I²C Status

7	6	5	4	3	2	1	0
I2STAT[7:3]					0	0	0
R					R	R	R

Address: BDH Reset value: 1111 1000b

Bit	Name	Description
7:3	I2STAT[7:3]	**I²C status code** The MSB five bits of I2STAT contains the status code. There are 27 possible status codes. When I2STAT is F8H, no relevant state information is available and SI flag keeps 0. All other 26 status codes correspond to the I²C states. When each of these status is entered, SI will be set as logic 1 and a interrupt is requested.
2:0	0	**Reserved** The least significant three bits of I2STAT are always read as 0.

PWMPH – PWM Period High Byte

7	6	5	4	3	2	1	0
PWMP[15:8]							
R/W							

Address: D1H reset value: 0000 0000b

Bit	Name	Description
7:0	PWMP[15:8]	**PWM period high byte** This byte with PWMPL controls the period of the PWM generator signal.

PWM0H – PWM0 Duty High Byte

7	6	5	4	3	2	1	0
PWM0[15:8]							
R/W							

Address: D2H reset value: 0000 0000b

Bit	Name	Description
7:0	PWM0[15:8]	**PWM0 duty high byte** This byte with PWM0L controls the duty of the output signal PG0 from PWM generator.

PWM1H – PWM1 Duty High Byte

7	6	5	4	3	2	1	0
PWM1[15:8]							
R/W							

Address: D3H reset value: 0000 0000b

Bit	Name	Description
7:0	PWM1[15:8]	**PWM1 duty high byte** This byte with PWM1L controls the duty of the output signal PG1 from PWM generator.

PWM2H – PWM2 Duty High Byte

7	6	5	4	3	2	1	0
PWM2[15:8]							
R/W							

Address: D4H reset value: 0000 0000b

Bit	Name	Description
7:0	PWM2[15:8]	**PWM2 duty high byte** This byte with PWM2L controls the duty of the output signal PG2 from PWM generator.

ADCCON2 – ADC Control 2

7	6	5	4	3	2	1	0
ADFBEN	ADCMPOP	ADCMPEN	ADCMPO	-	-	-	ADCDLY.8
R/W	R/W	R/W	R	-	-	-	R/W

Address: E2H Reset value: 0000 0000b

Bit	Name	Description
7	ADFBEN	**ADC compare result asserting Fault Brake enable** 0 = ADC asserting Fault Brake Disabled. 1 = ADC asserting Fault Brake Enabled. Fault Brake is asserted once its compare result ADCMPO is 1. Meanwhile, PWM channels output Fault Brake data. PWMRUN (PWMCON0.7) will also be automatically cleared by hardware. The PWM output resumes when PWMRUN is set again.
6	ADCMPOP	**ADC comparator output polarity** 0 = ADCMPO is 1 if ADCR[11:0] is greater than or equal to ADCMP[11:0]. 1 = ADCMPO is 1 if ADCR[11:0] is less than ADCMP[11:0].
5	ADCMPEN	**ADC result comparator enable** 0 = ADC result comparator Disabled. 1 = ADC result comparator Enabled.
4	ADCMPO	**ADC comparator output value** This bit is the output value of ADC result comparator based on the setting of ACMPOP. This bit updates after every A/D conversion complete.
0	ADCDLY.8	**ADC external trigger delay counter bit 8** See ADCDLY register.

ADCDLY – ADC Trigger Delay Counter

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

Address: E3H Reset value: 0000 0000b

Bit	Name	Description
7:0	ADCDLY[7:0]	**ADC external trigger delay counter low byte** This 8-bit field combined with ADCCON2.0 forms a 9-bit counter. This counter inserts a delay after detecting the external trigger. An A/D converting starts after this period of delay. External trigger delay time = $\frac{ADCDLY}{F_{ADC}}$. Note that this field is valid only when ADCEX (ADCCON1.1) is set. User should not modify ADCDLY during PWM run time if selecting PWM output as the external ADC trigger source.

P3 – Port 3 (Bit-addressable)

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	P3.0
R	R	R	R	R	R	R	R/W

Address: B0H Reset value: 0000 0001b

Bit	Name	Description
7:1	0	**Reserved** The bits are always read as 0.
0	P3.0	**Port 3 bit 0** P3.0 is available only when the internal oscillator is used as the system clock. At this moment, P3.0 functions as a general purpose I/O. If the system clock is not selected as the internal oscillator, P3.0 pin functions as OSCIN. A write to P3.0 is invalid and P3.0 is always read as 0.

7.6.2 Output Mode Control

These registers control output mode which is configurable among four modes: input-only, quasi-bidirectional, push-pull, or open-drain. Each pin can be configured individually. There is also a pull-up control for P2.0 in P2S.7.

P0M1 – Port 0 Mode Select 1[1]

7	6	5	4	3	2	1	0
P0M1.7	P0M1.6	P0M1.5	P0M1.4	P0M1.3	P0M1.2	P0M1.1	P0M1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: B1H, Page: 0 Reset value: 1111 1111b

Bit	Name	Description
7:0	P0M1[7:0]	**Port 0 mode select 1**

P0M2 – Port 0 Mode Select 2[1]

7	6	5	4	3	2	1	0
P0M2.7	P0M2.6	P0M2.5	P0M2.4	P0M2.3	P0M2.2	P0M2.1	P0M2.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: B2H, Page: 0 Reset value: 0000 0000b

Bit	Name	Description
7:0	P0M2[7:0]	**Port 0 mode select 2**

[1] P0M1 and P0M2 are used in combination to determine the I/O mode of each pin of P0. See Table 7-1. Configuration for Different I/O Modes.

UART baud rate generator for it runs without continuous software intervention. Note that only Timer1 can be the baud rate source for UART. Counting is enabled by setting the TR0 (TR1) bit as 1 and proper setting of GATE and $\overline{INT0}$ ($\overline{INT1}$) pins. The functions of GATE and $\overline{INT0}$ ($\overline{INT1}$) pins are just the same as Mode 0 and 1.

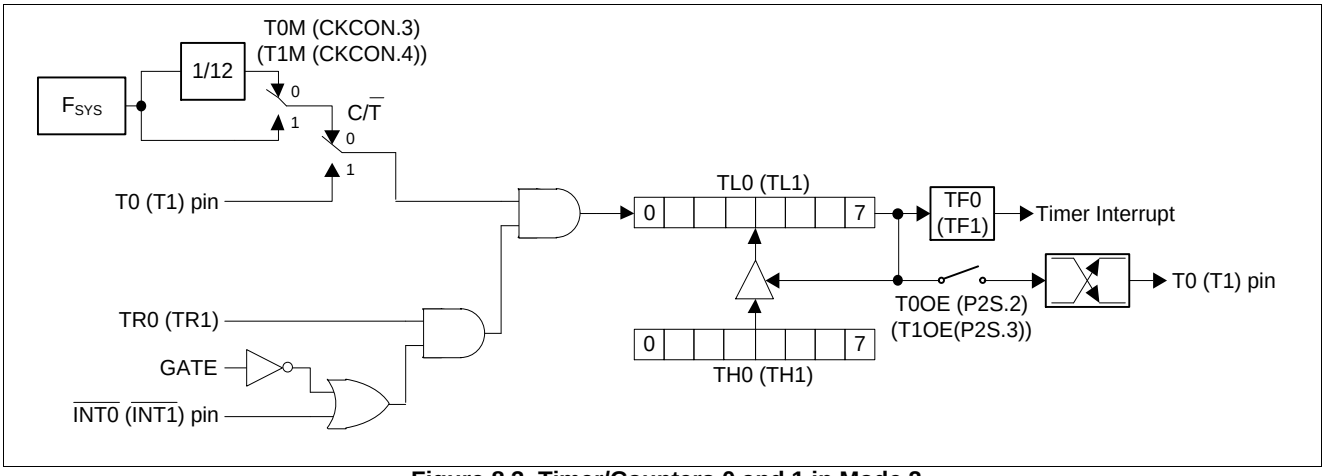


Figure 8.3. Timer/Counters 0 and 1 in Mode 2

8.4 Mode 3 (Two Separate 8-Bit Timers)

Mode 3 has different operating methods for Timer 0 and Timer 1. 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. TL0 uses the Timer/Counter 0 control bits C/$\overline{T}$, GATE, TR0, $\overline{INT0}$, and TF0. The TL0 also can be used as a 1-to-0 transition counter on pin T0 as determined by C/$\overline{T}$ (TMOD.2). TH0 is forced as a clock cycle counter and takes over the usage of TR1 and TF1 from Timer/Counter 1. Mode 3 is used in case that an extra 8 bit timer is needed. If Timer/Counter 0 is configured in Mode 3, Timer/Counter 1 can be turned on or off by switching it out of or into its own Mode 3. It can still be used in Modes 0, 1 and 2 although its flexibility is restricted. It no longer has control over its overflow flag TF1 and the enable bit TR1. However Timer 1 can still be used as a Timer/Counter and retains the use of GATE, $\overline{INT1}$ pin and T1M. It can be used as a baud rate generator for the serial port or other application not requiring an interrupt.

CAPCON1 – Input Capture Control 1

7	6	5	4	3	2	1	0
-	-	CAP2LS[1:0]		CAP1LS[1:0]		CAP0LS[1:0]	
-	-	R/W		R/W		R/W	

Address: 93H Reset value: 0000 0000b

Bit	Name	Description
5:4	CAP2LS[1:0]	**Input capture 2 level select** 00 = Falling edge. 01 = Rising edge. 10 = Either Rising or falling edge. 11 = Reserved.
3:2	CAP1LS[1:0]	**Input capture 1 level select** 00 = Falling edge. 01 = Rising edge. 10 = Either Rising or falling edge. 11 = Reserved.
1:0	CAP0LS[1:0]	**Input capture 0 level select** 00 = Falling edge. 01 = Rising edge. 10 = Either Rising or falling edge. 11 = Reserved.

CAPCON2 – Input Capture Control 2

7	6	5	4	3	2	1	0
-	ENF2	ENF1	ENF0	-	-	-	-
-	R/W	R/W	R/W	-	-	-	-

Address: 94H Reset value: 0000 0000b

Bit	Name	Description
6	ENF2	**Enable noise filer on input capture 2** 0 = Noise filter on input capture channel 2 Disabled. 1 = Noise filter on input capture channel 2 Enabled.
5	ENF1	**Enable noise filer on input capture 1** 0 = Noise filter on input capture channel 1 Disabled. 1 = Noise filter on input capture channel 1 Enabled.
4	ENF0	**Enable noise filer on input capture 0** 0 = Noise filter on input capture channel 0 Disabled. 1 = Noise filter on input capture channel 0 Enabled.

10. TIMER 3

Timer 3 is implemented simply as a 16-bit auto-reload, up-counting timer. The user can select the pre-scale with T3PS[2:0] (T3CON[2:0]) and fill the reload value into RH3 and RL3 registers to determine its overflow rate. User then can set TR3 (T3CON.3) to start counting. When the counter rolls over FFFFH, TF3 (T3CON.4) is set as 1 and a reload is generated and causes the contents of the RH3 and RL3 registers to be reloaded into the internal 16-bit counter. If ET3 (EIE1.1) is set as 1, Timer 3 interrupt service routine will be served. TF3 is auto-cleared by hardware after entering its interrupt service routine.

Timer 3 can also be the baud rate clock source of both UARTs. For details, please see Section 13.5 “Baud Rate” on page 133.

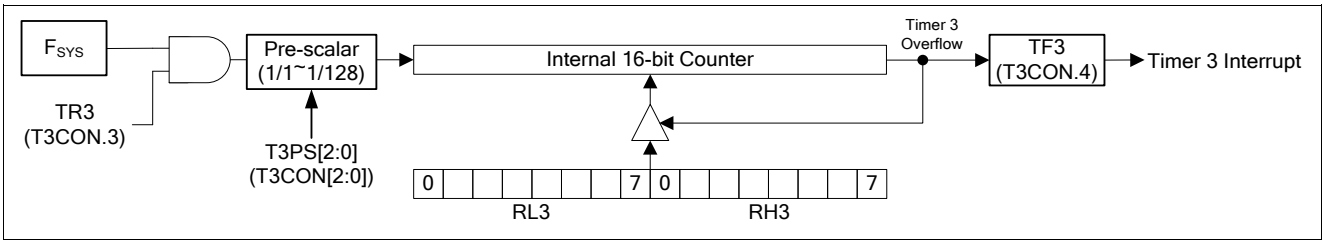


Figure 10.1. Timer 3 Block Diagram

T3CON – Timer 3 Control

7	6	5	4	3	2	1	0
SMOD_1	SMOD0_1	BRCK	TF3	TR3	T3PS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Address: C4H, Page:0 Reset value: 0000 0000b

Bit	Name	Description
4	TF3	**Timer 3 overflow flag** This bit is set when Timer 3 overflows. It is automatically cleared by hardware when the program executes the Timer 3 interrupt service routine. This bit can be set or cleared by software.
3	TR3	**Timer 3 run control** 0 = Timer 3 is halted. 1 = Timer 3 starts running. Note that the reload registers RH3 and RL3 can only be written when Timer 3 is halted (TR3 bit is 0). If any of RH3 or RL3 is written if TR3 is 1, result is unpredictable.

Bit	Name	Description
0	RI_1	**Receiving interrupt flag** This flag is set via hardware when a data frame has been received by the serial port 1 after the 8th bit in Mode 0 or after sampling the stop bit in Mode 1, 2, or 3. SM2_1 bit as logic 1 has restriction for exception. When the serial port 1 interrupt is enabled, setting this bit causes the CPU to execute to the serial port 1 interrupt service routine. This bit must be cleared manually via software.

PCON – Power Control

7	6	5	4	3	2	1	0
SMOD	SMOD0	-	POF	GF1	GF0	PD	IDL
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Address: 87H Reset value: see Table 6-2. SFR Definitions and Reset Values

Bit	Name	Description
7	SMOD	**Serial port 0 double baud rate enable** Setting this bit doubles the serial port baud rate when UART0 is in Mode 2 or when Timer 1 overflow is used as the baud rate source of UART0 Mode 1 or 3. See Table 13-1. Serial Port 0 Mode Description for details.
6	SMOD0	**Serial port 0 framing error flag access enable** 0 = SCON.7 accesses to SM0 bit. 1 = SCON.7 accesses to FE bit.

T3CON – Timer 3 Control

7	6	5	4	3	2	1	0
SMOD_1	SMOD0_1	BRCK	TF3	TR3	T3PS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Address: C4H, Page:0 Reset value: 0000 0000b

Bit	Name	Description
7	SMOD_1	**Serial port 1 double baud rate enable** Setting this bit doubles the serial port baud rate when UART1 is in Mode 2. See Table 13-2. Serial Port 1 Mode Description for details.
6	SMOD0_1	**Serial port 1 framing error access enable** 0 = SCON_1.7 accesses to SM0_1 bit. 1 = SCON_1.7 accesses to FE_1 bit.

Table 13-1. Serial Port 0 Mode Description

Mode	SM0	SM1	Description	Frame Bits	Baud Rate
0	0	0	Synchronous	8	F_{SYS} divided by 12 or by 2[1]
1	0	1	Asynchronous	10	Timer 1/Timer 3 overflow rate divided by 32 or 16[2]
2	1	0	Asynchronous	11	F_{SYS} divided by 32 or 64[2]
3	1	1	Asynchronous	11	Timer 1/Timer 3 overflow rate divided by 32 or 16[2]

[1] While SM2 (SCON.5) is logic 1.
[2] While SMOD (PCON.7) is logic 1.

Example 3, slave 2:

```
SADDR = 11000000b
SADEN = 11111100b
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-cares”, e.g.:

```
SADDR     = 01010110b
SADEN     = 11111100b
Broadcast = 1111111Xb
```

The use of don’t-care bits provides flexibility in defining the Broadcast address, however in most applications, interpreting the “don’t-cares” as all 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 cares” as well as a “Broadcast” address of all XXXXXXXXb (all “don’t care” bits). This ensures that the serial port will reply to any address, and so that it is backwards compatible with the standard 80C51 microcontrollers that do not support automatic address recognition.

SPDR – Serial Peripheral Data Register

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

Address: F5H Reset value: 0000 0000b

Bit	Name	Description
7:0	SPDR[7:0]	**Serial peripheral data** This byte is used for transmitting or receiving data on SPI bus. A write of this byte is a write to the shift register. A read of this byte is actually a read of the read data buffer. In Master mode, a write to this register initiates transmission and reception of a byte simultaneously.

14.2 Operating Modes

14.2.1 Master Mode

The SPI can operate in Master mode while MSTR (SPCR.4) is set as 1. Only one Master SPI device can initiate transmissions. A transmission always begins by Master through writing to SPDR. The byte written to SPDR begins shifting out on MOSI pin under the control of SPCLK. Simultaneously, another byte shifts in from the Slave on the MISO pin. After 8-bit data transfer complete, SPIF (SPSR.7) will automatically set via hardware to indicate one byte data transfer complete. At the same time, the data received from the Slave is also transferred in SPDR. User can clear SPIF and read data out of SPDR.

14.2.2 Slave Mode

When MSTR is 0, the SPI operates in Slave mode. The SPCLK pin becomes input and it will be clocked by another Master SPI device. The $\overline{SS}$ pin also becomes input. The Master device cannot exchange data with the Slave device until the $\overline{SS}$ pin of the Slave device is externally pulled low. Before data transmissions occurs, the $\overline{SS}$ of the Slave device should be pulled and remain low until the transmission is complete. If $\overline{SS}$ goes high, the SPI is forced into idle state. If the $\overline{SS}$ is forced to high at the middle of transmission, the transmission will be aborted and the rest bits of the receiving shifter buffer will be high and goes into idle state.

In Slave mode, data flows from the Master to the Slave on MOSI pin and flows from the Slave to the Master on MISO pin. The data enters the shift register under the control of the SPCLK from the Master device. After one byte is received in the shift register, it is immediately moved into the read data buffer and the SPIF bit is set. A read of the SPDR is actually a read of the read data buffer. To prevent an overrun and the loss of the byte that caused by the overrun, the Slave should read SPDR out and the first SPIF should be cleared before a second transfer of data from the Master device comes in the read data buffer.

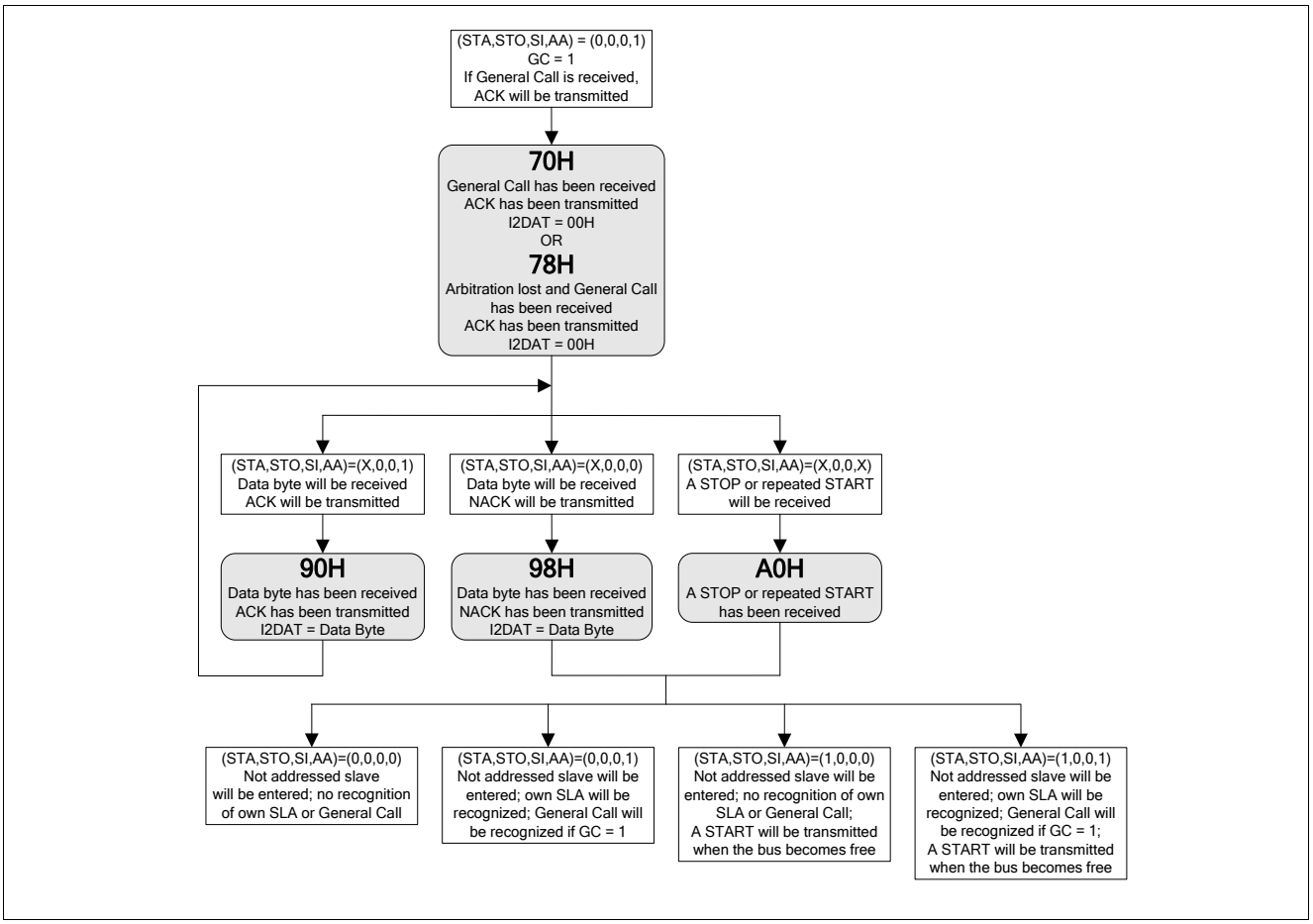


Figure 15.11. Flow and Status of General Call Mode

15.3.6 Miscellaneous States

There are two I2STAT status codes that do not correspond to the 25 defined states, which are mentioned in previous sections. These are F8H and 00H states.

The first status code F8H indicates that no relevant information is available during each transaction. Meanwhile, the SI flag is 0 and no I^2C interrupt is required.

The other status code 00H means a bus error has occurred during a transaction. A bus error is caused by a START or STOP condition appearing temporally at an illegal position such as the second through eighth bits of an address or a data byte, and the acknowledge bit. When a bus error occurs, the SI flag is set immediately. When a bus error is detected on the I^2C bus, the operating device immediately switches to the not addressed salve mode, releases SDA and SCL lines, sets the SI flag, and loads I2STAT as 00H. To recover from a bus

PWMPH – PWM Period High Byte

7	6	5	4	3	2	1	0
PWMP[15:8]							
R/W							

Address: D1H reset value: 0000 0000b

Bit	Name	Description
7:0	PWMP[15:8]	**PWM period high byte** This byte with PWMPL controls the period of the PWM generator signal.

PWM0L – PWM0 Duty Low Byte

7	6	5	4	3	2	1	0
PWM0[7:0]							
R/W							

Address: DAH reset value: 0000 0000b

Bit	Name	Description
7:0	PWM0[7:0]	**PWM0 duty low byte** This byte with PWM0H controls the duty of the output signal PG0 from PWM generator.

PWM0H – PWM0 Duty High Byte

7	6	5	4	3	2	1	0
PWM0[15:8]							
R/W							

Address: D2H reset value: 0000 0000b

Bit	Name	Description
7:0	PWM0[15:8]	**PWM0 duty high byte** This byte with PWM0L controls the duty of the output signal PG0 from PWM generator.

PWM1L – PWM/1 Duty Low Byte

7	6	5	4	3	2	1	0
PWM1[7:0]							
R/W							

Address: DBH reset value: 0000 0000b

Bit	Name	Description
7:0	PWM1[7:0]	**PWM1 duty low byte** This byte with PWM1H controls the duty of the output signal PG1 from PWM generator.

PDTCNT – PWM Dead-time Counter (TA protected)

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

Address: FAH Reset value: 0000 0000b

Bit	Name	Description
7:0	PDTCNT[7:0]	**PWM dead-time counter low byte** This 8-bit field combined with PDTEN.4 forms a 9-bit PWM dead-time counter PDTCNT. This counter is valid only when PWM is under complementary mode and the correspond PDTEN bit for PWM pair is set. PWM dead-time = $\frac{PDTCNT+1}{F_{SYS}}$. Note that user should not modify PDTCNT during PWM run time.

17.1.3.3 Synchronous Mode

Synchronous mode is enabled when PWMMOD[1:0] = [1:0]. In this mode, PG0/2/4 output PWM signals the same as the independent mode. PG1/3/5 output just the same in-phase PWM signals of PG02/4 correspondingly.

17.1.4 Mask Output Control

Each PWM signal can be software masked by driving a specified level of PWM signal. The PWM mask output function is quite useful when controlling Electrical Commutation Motor like a BLDC. PMEN contains six bits, those determine which channel of PWM signal will be masked. PMD set the individual mask level of each PWM channel. The default value of PMEN is 00H, which makes all outputs of PWM channels follow signals from PWM generator. Note that the masked level is reversed or not by PNP setting on PWM output pins.

PMEN – PWM Mask Enable

7	6	5	4	3	2	1	0
-	-	PMEN5	PMEN4	PMEN3	PMEN2	PMEN1	PMEN0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Address: FBH Reset value: 0000 0000b

Bit	Name	Description
n	PMENn	**PWMn mask enable** 0 = PWMn signal outputs from its PWM generator. 1 = PWMn signal is masked by PMDn.

PNP – PWM Negative Polarity

7	6	5	4	3	2	1	0
-	-	PNP5	PNP4	PNP3	PNP2	PNP1	PNP0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Address: D6H Reset value: 0000 0000b

Bit	Name	Description
n	PNPn	**PWMn negative polarity output enable** 0 = PWMn signal outputs directly on PWMn pin. 1 = PWMn signal outputs inversely on PWMn pin.

17.2 PWM Interrupt

The PWM module has a flag PWMF (PWMCON0.5) to indicate certain point of each complete PWM period. The indicating PWM channel and point can be selected by INTSEL[2:0] and INTTYP[1:0] (PWMINTC[2:0] and [5:4]). Note that the center point and the end point interrupts are only available when PWM operates in its center-aligned type. PWMF is cleared by software.

PWMINTC – PWM Interrupt Control

7	6	5	4	3	2	1	0
-	-	INTTYP1	INTTYP0	-	INTSEL2	INTSEL1	INTSEL0
-	-	R/W	R/W	-	R/W	R/W	R/W

Address: B7H, Page:1 Reset value: 0000 0000b

Bit	Name	Description
5:4	INTTYP[1:0]	**PWM interrupt type select** These bit select PWM interrupt type. 00 = Falling edge on PWM0/1/2/3/4/5 pin. 01 = Rising edge on PWM0/1/2/3/4/5 pin. 10 = Central point of a PWM period. 11 = End point of a PWM period. Note that the central point interrupt or the end point interrupt is only available while PWM operates in center-aligned type.
2:0	INTSEL[2:0]	**PWM interrupt pair select** These bits select which PWM channel asserts PWM interrupt when PWM interrupt type is selected as falling or rising edge on PWM0/1/2/3/4/5 pin.. 000 = PWM0. 001 = PWM1. 010 = PWM2. 011 = PWM3. 100 = PWM4. 101 = PWM5. Others = PWM0.

The PWM interrupt related with PWM waveform is shown as figure below.

The effective condition is selected by ETGSEL (ADCCON0[5:4]) and ETGTYP (ADCCON1[3:2]). A trigger delay can also be inserted between external trigger point and A/D conversion. The external trigging ADC hardware with controllable trigger delay makes the N76E003 feasible for high performance motor control. Note that during ADC is busy in converting (ADCS = 1), any conversion triggered by software or hardware will be ignored and there is no warning presented.

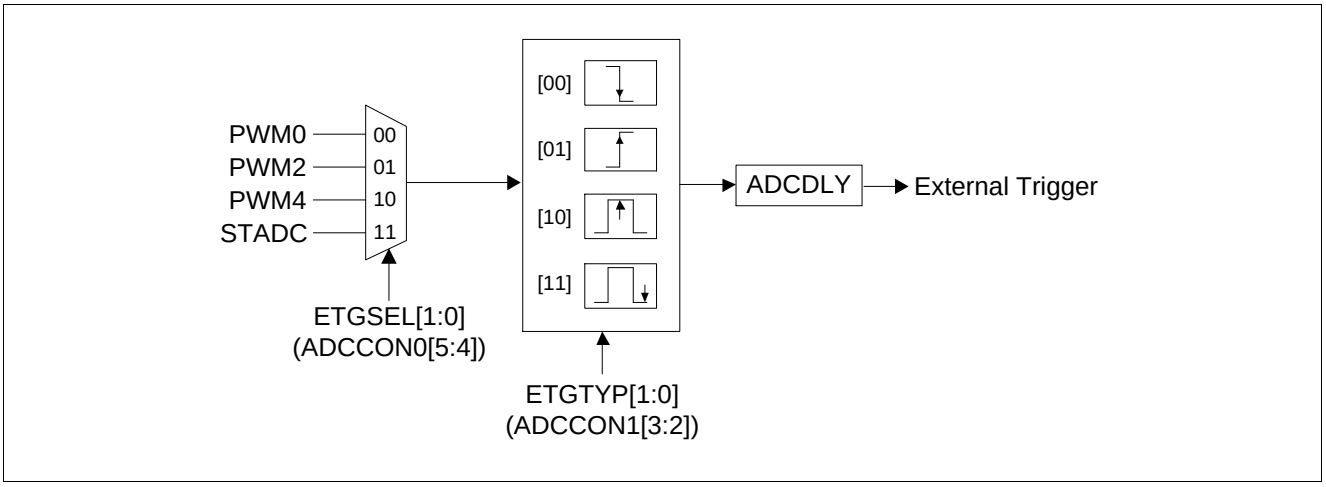


Figure 18.2. External Triggering ADC Circuit

18.1.3 ADC Conversion Result Comparator

The N76E003 ADC has a digital comparator, which compares the A/D conversion result with a 12-bit constant value given in ACMPH and ACMPL registers. The ADC comparator is enabled by setting ADCMPEN (ADCCON2.5) and each compare will be done on every A/D conversion complete moment. ADCMPO (ADCCON2.4) shows the compare result according to its output polarity setting bit ADCMPOP (ADCCON2.6). The ADC comparing result can trigger a PWM Fault Brake output directly. This function is enabled when ADFBEN (ADCCON2.7). When ADCMPO is set, it generates a ADC compare event and asserts Fault Brake. Please also see Sector 18.1.5“Fault Brake” on page 129.

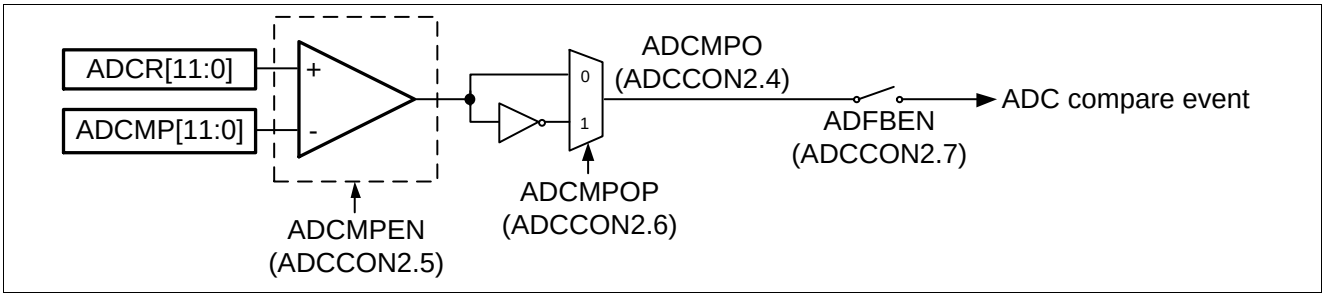


Figure 18.3. ADC Result Comparator

IAP Mode	IAPCN				IAPA[15:0] {IAPAH, IAPAL}	IAPFD[7:0]
	IAPB[1:0]	FOEN	FCEN	FCTRL[3:0]		
CONFIG byte-program	11	1	0	0001	CONFIG0: 0000H CONFIG1: 0001H CONFIG2: 0002H CONFIG4: 0004H	Data in
CONFIG byte-read	11	0	0	0000	CONFIG0: 0000H CONFIG1: 0001H CONFIG2: 0002H CONFIG4: 0004H	Data out

[1] "X" means "don't care".
[2] Each page is 128 Bytes size. Therefore, the address should be the address pointed to the target page.

21.2 IAP User Guide

IAP facilitates the updating flash contents in a convenient way; however, user should follow some restricted laws in order that the IAP operates correctly. Without noticing warnings will possible cause undetermined results even serious damages of devices. Furthermore, this paragraph will also support useful suggestions during IAP procedures.

(1) If no more IAP operation is needed, user should clear IAPEN (CHPCON.0). It will make the system void to trigger IAP unaware. Furthermore, IAP requires the HIRC running. If the external clock source is selected, disabling IAP will stop the HIRC for saving power consumption. Note that a write to IAPEN is TA protected.

(2) When the LOCK bit (CONFIG0.1) is activated, IAP reading, writing, or erasing can still be valid.

During IAP progress, interrupts (if enabled) should be disabled temporally by clearing EA bit for implement limitation.

Do not attempt to erase or program to a page that the code is currently executing. This will cause unpredictable program behavior and may corrupt program data.

21.3 Using Flash Memory as Data Storage

In general application, there is a need of data storage, which is non-volatile so that it remains its content even after the power is off. Therefore, in general application user can read back or update the data, which rules as parameters or constants for system control. The Flash Memory array of the N76E003 supports IAP function and any byte in the Flash Memory array may be read using the MOVC instruction and thus is suitable for use as non-volatile data storage. IAP provides erase and program function that makes it easy for one or more bytes within a page to be erased and programmed in a routine. IAP performs in the application under the

```
        MOV     TA,#0Aah
        MOV     TA,#55h
        ORL     CHPCON,#80h                 ;software reset and reboot from APROM

        SJMP    $

;*****************************************************************
;             IAP Subroutine
;*****************************************************************
Enable_IAP:
        MOV     TA,#0Aah                    ;CHPCON is TA protected
        MOV     TA,#55h
        ORL     CHPCON,#00000001b           ;IAPEN = 1, enable IAP mode
        RET

Disable_IAP:
        MOV     TA,#0Aah
        MOV     TA,#55h
        ANL     CHPCON,#11111110b           ;IAPEN = 0, disable IAP mode
        RET

Enable_AP_Update:
        MOV     TA,#0Aah                    ;IAPUEN is TA protected
        MOV     TA,#55h
        ORL     IAPUEN,#00000001b           ;APUEN = 1, enable APROM update
        RET

Disable_AP_Update:
        MOV     TA,#0Aah
        MOV     TA,#55h
        ANL     IAPUEN,#11111110b           ;APUEN = 0, disable APROM update
        RET

Enable_CONFIG_Update:
        MOV     TA,#0Aah
        MOV     TA,#55h
        ORL     IAPUEN,#00000100b           ;CFUEN = 1, enable CONFIG update
        RET

Disable_CONFIG_Update:
        MOV     TA,#0Aah
        MOV     TA,#55h
        ANL     IAPUEN,#11111011b           ;CFUEN = 0, disable CONFIG update
        RET

Trigger_IAP:
        MOV     TA,#0Aah                    ;IAPTRG is TA protected
        MOV     TA,#55h
        ORL     IAPTRG,#00000001b           ;write '1' to IAPGO to trigger IAP process
        RET

;*****************************************************************
;             IAP APROM Function
;*****************************************************************
Erase_AP:
        MOV     IAPCN,#PAGE_ERASE_AP
        MOV     IAPFD,#0FFh
        MOV     R0,#00h
Erase_AP_Loop:
```

33. DOCUMENT REVISION HISTORY

Revision	Date	Description	
V1.00	2016/10/28	Initial release	
V1.01	2017/6/12	Chapter 31.3 Chapter 31.4 Chapter 13.5 Chapter 24.1 Chapter 31.7	Added HIRC Accuracy vs. Temperature figure. Modified band-gap max value from 1.27 to 1.30. Modified SFR name to RCTRIM0 and RCTRIM1 for modify HIRC application. Added disable POR function after power on description. Modified data retention value at 25℃ condition, added condition at 55℃and 85℃data.
V1.02	2017/6/26	Chapter 18.1.4 Chapter 15.3	Added chapter description of band-gap as ADC input to calculate the VDD value. Added how to clear SI register example code for I^2C transmission issue.
V1.03	2017/10/3	Chapter 4 Chapter 14.1	Add N76E003BQ20 pin assignment Modify N76E003AQ20 pin assignment, remove 21pin GND mark Added SPSR.2, SPI writer data buffer status bit description.
V1.04	2017/12/13	Chapter 4 Chapter 11.1 Chapter 25.5 Chapter 31.6 Chapter 32.3	Modified pin description table, add N76E003BQ20 pin description Added note for WDT reset fail when CKDIV enable and into power down mode Added note for WDT reset fail when CKDIV enable and into power down mode Modify EFT Characteristics, remove BOD enable/disable condition. Added QFN20 package dimensions for N76E003BQ20