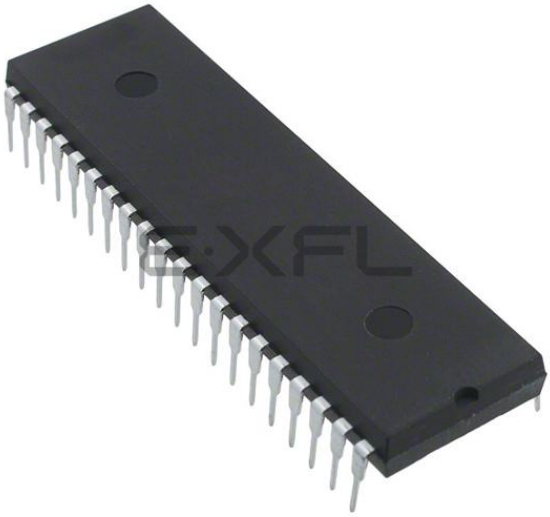
Nuvoton Technology Corporation - N78E055ADG Datasheet



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, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	32
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.25K x 8
Voltage - Supply (Vcc/Vdd)	2.4V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/n78e055adg

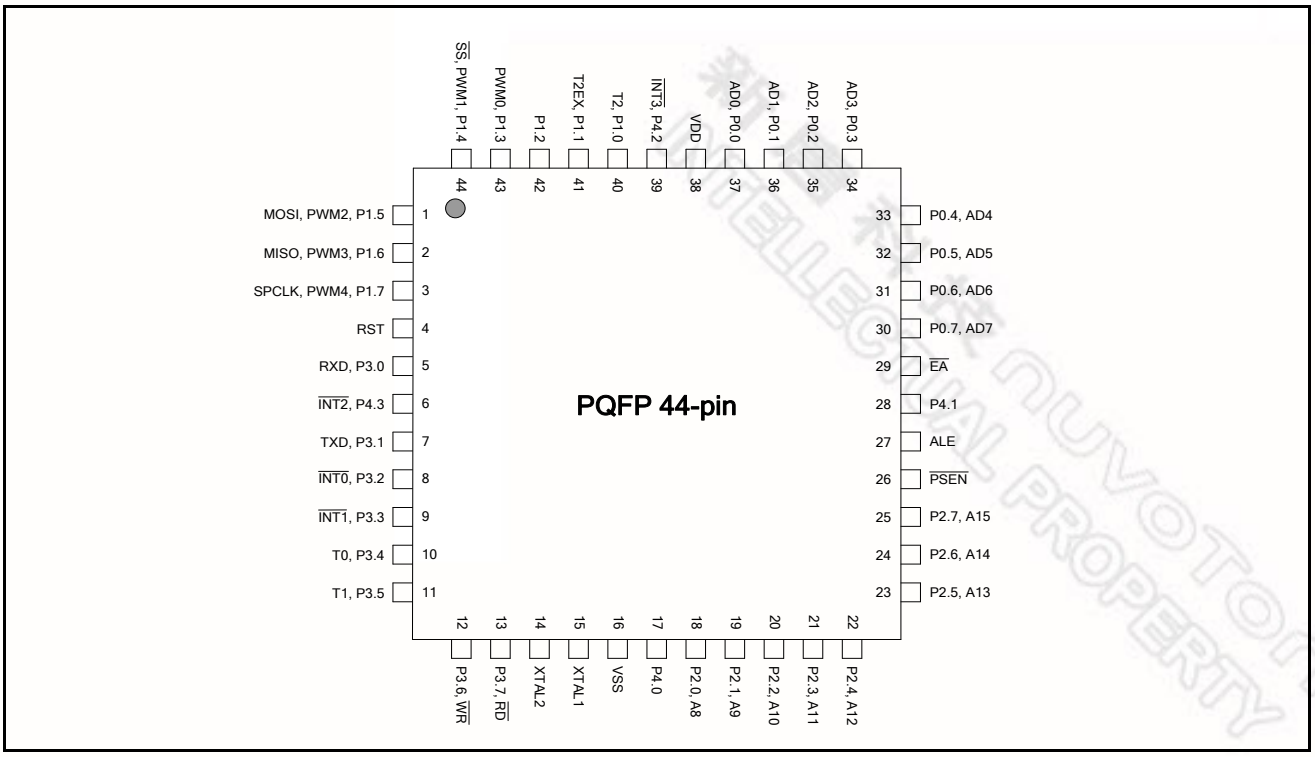


Figure 4–3. Pin Assignment of PQFP 44-Pin

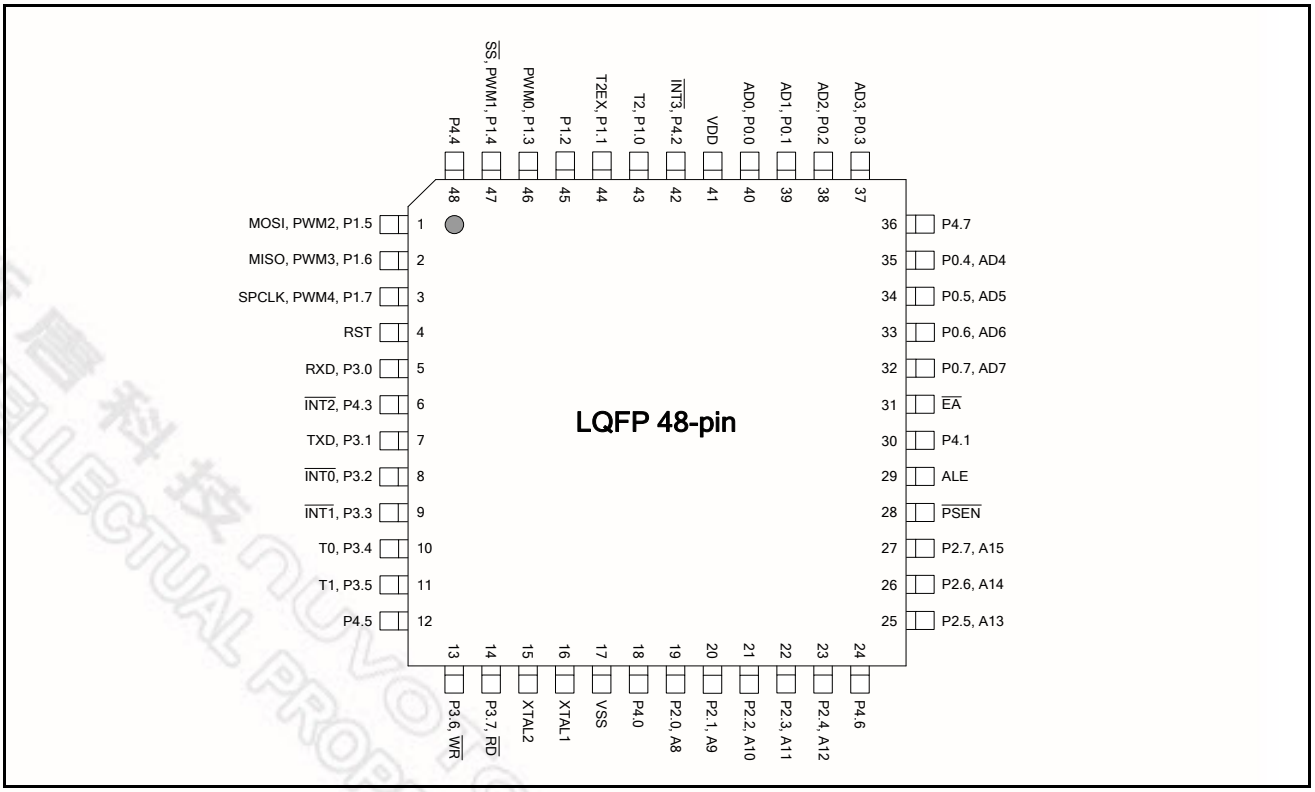


Figure 4–4. Pin Assignment of LQFP 48-Pin

```
MOVX   A,@R0

MOV    DPTR,#0123H          ;write #5BH to XRAM with address @0123H.
MOV    A,#5BH
MOVX   @DPTR,A

MOV    DPTR,#0123H          ;read from XRAM with address @0123H.
MOVX   A,@DPTR
```

10. TIMERS/COUNTERS

N78E059A/N78E055A has three 16-bit programmable timers/counters.

10.1 Timer/Counters 0 and 1

Timer/Counter 0 and 1 on N78E059A/N78E055A are two 16-bit Timer/Counters. Each of them has two 8 bit registers which form the 16 bit counting register. For Timer/Counter 0 they are TH0, the upper 8 bits register, and TL0, the lower 8 bit register. Similarly Timer/Counter 1 has two 8 bit registers, TH1 and TL1. TCON and TMOD can configure modes of Timer/Counter 0 and 1.

The Timer or Counter function is selected by the C/$\overline{T}$ bit in TMOD. Each Timer/Counter has its own selection bit. TMOD.2 selects the function for Timer/Counter 0 and TMOD.6 selects the function for Timer/Counter 1

When configured as a “Timer”, the timer counts clock cycles. The timer clock is 1/6 of the peripheral clock (F_{PERIPH}). In the “Counter” mode, the register increases on the falling edge of the external input pins T0 for Timer 0 and T1 for Timer 1. If the sampled value is high in one machine-cycle and low in the next, a valid 1 to 0 transition on the pin is recognized and the count register increases.

In addition, each Timer/Counter can be set to operate in any one of four possible modes. Bits M0 and M1 in TMOD do the mode selection.

TMOD – Timer 0 and 1 Mode

7	6	5	4	3	2	1	0
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Address: 89H reset value: 0000 0000b

Bit	Name	Description
7	GATE	**Timer 1 gate control.** 0 = Timer 1 will clock when TR1 = 1 regardless of $\overline{INT1}$ logic level. 1 = Timer 1 will clock only when TR1 = 1 and $\overline{INT1}$ is logic 1.
6	C/$\overline{T}$	**Timer 1 Counter/Timer select.** 0 = Timer 1 is incremented by internal peripheral clocks. 1 = Timer 1 is incremented by the falling edge of the external pin T1.
5	M1	**Timer 1 mode select.**
4	M0	M1 M0 Timer 1 Mode 0 0 Mode 0: 8-bit Timer/Counter with 5-bit pre-scalar (TL1[4:0]) 0 1 Mode 1: 16-bit Timer/Counter 1 0 Mode 2: 8-bit Timer/Counter with auto-reload from TH1 1 1 Mode 3: Timer 1 halted

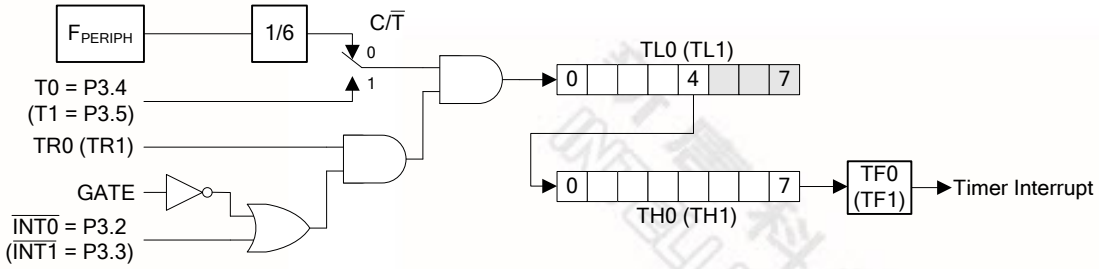


Figure 10–1. Timer/Counters 0 and 1 in Mode 0

10.1.2 Mode 1 (16-bit Timer)

Mode 1 is similar to Mode 0 except that the counting registers are fully used as a 16-bit counter. Roll-over occurs when a count moves FFFFH to 0000H. The Timer overflow flag TFx of the relevant Timer/Counter is set and an interrupt will occurs if enabled.

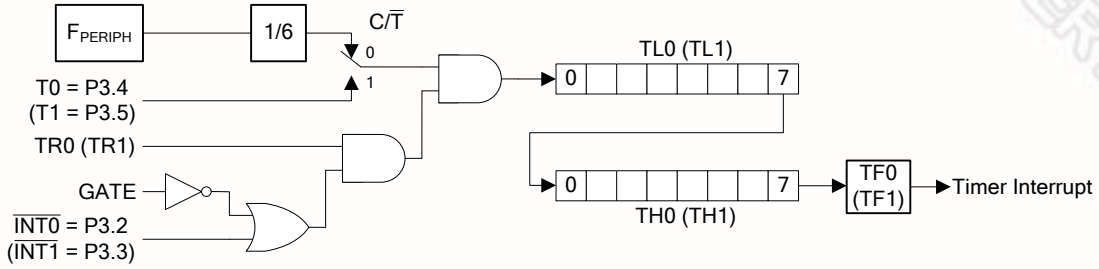


Figure 10–2. Timer/Counters 0 and 1 in Mode 1

10.1.3 Mode 2 (8-bit Auto-reload Timer)

In Mode 2, the Timer/Counter is in auto-reload mode. In this mode, TLx acts as an 8-bit count register whereas THx holds the reload value. When the TLx register overflows from FFH to 00H, the TFx bit in TCON is set, 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. This feature is best suitable for 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 the TRx bit and proper setting of GATE and $\overline{INTx}$ pins. The functions of GATE and $\overline{INTx}$ pins are just the same as Mode 0 and 1.

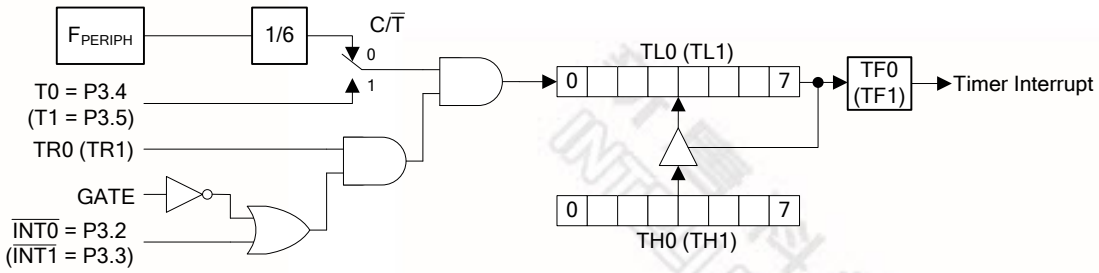


Figure 10–3. Timer/Counter 0 and 1 in Mode 2

10.1.4 Mode 3 (Two Separate 8-bit Timers)

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. 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 which 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 and $\overline{INT1}$ pin. It can be used as a baud rate generator for the serial port or other application not requiring an interrupt.

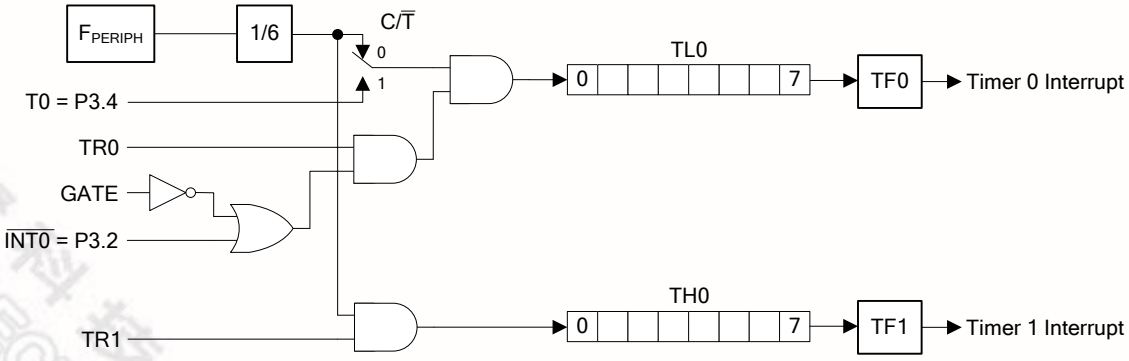


Figure 10–4. Timer/Counter 0 in Mode 3

10.2 Timer/Counter 2

Timer/Counter 2 is a 16-bit up counter, which is configured by the T2MOD and T2CON registers. The count stores in two 8-bit cascade registers TH2 and TL2. Timer/Counter 2 is additionally equipped with a capture or reload capability. It also can be configured as the baud rate generator for UART or a square wave generator. The features listed above could be achieved because of the addition Timer/Counter 2 capture registers RCAP2H and RCAP2L. As with the Timer 0 and Timer 1 counters, there exists considerable flexibility in selecting and controlling the clock and in defining the operating mode. The clock source for Timer/Counter 2 may be selected from either the external T2 pin ($C/\overline{T2}$ (T2CON.1) = 1) or the crystal oscillator ($C/\overline{T2}$ = 0). The clock is then enabled when TR2 (T2CON.2) is a 1, and disabled when TR2 is a 0. The following registers are related to Timer/Counters 2 function.

T2CON – Timer 2 Control (bit-addressable)

7	6	5	4	3	2	1	0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	$C/\overline{T2}$	$CP/\overline{RL2}$
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Address: C8H reset value: 0000 0000b

Bit	Name	Description
7	TF2	**Timer 2 overflow flag.** This bit is set when Timer 2 overflows. If the Timer 2 interrupt and the global interrupt are enable, setting this bit will make CPU execute Timer 2 interrupt service routine. This bit is not automatically cleared via hardware and must be cleared via software. TF2 will not be set while Timer 2 is configured in the baud rate generator or clock-out mode.
6	EXF2	**Timer 2 external flag.** This bit is set via hardware when a 1-to-0 transition on the T2EX input pin occurs and EXEN2 is logic 1. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to execute the Timer 2 Interrupt service routine. This bit is not automatically cleared via hardware and must be cleared via software.
5	RCLK	**Receive clock flag.** This bit selects which Timer is used for the UART's receive clock in serial Mode 1 or 3. 0 = Timer 1 overflows is used for UART receive baud rate clock. 1 = Timer 2 overflows is used for UART receive baud rate clock.
4	TCLK	**Transmit clock flag.** This bit selects which Timer is used for the UART's transmit clock in serial Mode 1 or 3. 0 = Timer 1 overflows is used for UART transmit baud rate clock. 1 = Timer 2 overflows is used for UART transmit baud rate clock.

Bit	Name	Description
3	DISMODF	**Disable Mode Fault error detection.** This bit is used in combination with the SSOE (SPCR.7) bit to determine the feature of $\overline{SS}$ pin as shown in Table 14–1. Slave Select Pin Configurations. DISMODF affects only in Master mode (MSTR = 1). 0 = Mode Fault detection is not disabled. $\overline{SS}$ serves as input pin for Mode Fault detection disregard of SSOE. 1 = Mode Fault detection is disabled. The feature of $\overline{SS}$ follows SSOE bit.
2:0	-	**Reserved.**

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 of 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.

14.4 Operating Modes

14.4.1 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. The user can clear SPIF and read data out of SPDR.

14.4.2 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 must be pulled and remain low until the transmission is complete. If $\overline{SS}$ goes

Bit	Name	Description
0	PWM0EN	**PWM0 enable.** 0 = PWM0 is disabled and stops. 1 = PWM0 is enabled and runs.

PWMCON1 – PWM Control 1

7	6	5	4	3	2	1	0
-	-	-	-	-	PWM4OE	-	PWM4EN
-	-	-	-	-	r/w	-	r/w

Address: CEH reset value: 0000 0000b

Bit	Name	Description
7:3	-	**Reserved.**
2	PWM4OE	**PWM4 output enable.** 0 = P1.7 serves as general purpose I/O. 1 = P1.7 serves as output pin of PWM4 signal.
1	-	**Reserved.**
0	PWM4EN	**PWM0 enable.** 0 = PWM4 is disabled and stops. 1 = PWM4 is enabled and runs.

PWMP – PWM Period

7	6	5	4	3	2	1	0
PWMP[7:0]							
r/w							

Address: D9H reset value: 0000 0000b

Bit	Name	Description
7:0	PWMP[7:0]	**PWM period.** This byte controls the period of the PWM output of PWM0~PWM4 channels.

PWM0 – PWM0 Duty

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.** This byte controls the duty of the PWM0 output.

This gives a repetition frequency range of 122Hz to 31.25kHz (F_{PERIPH} = 16MHz). By loading the PWMx registers with either 00H or FFH, the PWM channels will generate a constant low or high level output, respectively.

When a compare register PWMx is loaded with a new value, the associated output updated immediately. It does not have to wait until the end of the current counter period.

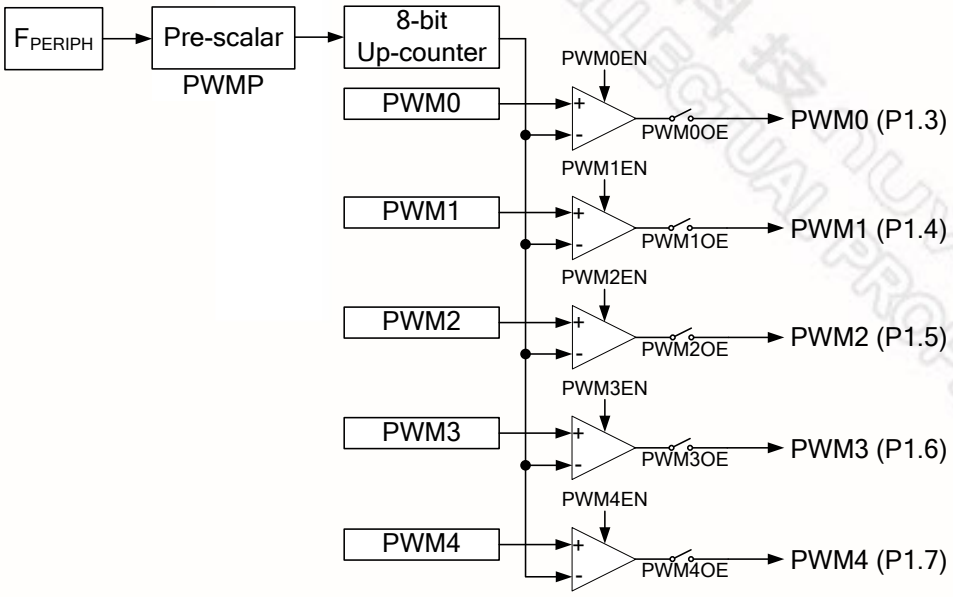


Figure 15–1. PWM Function Block

PWM demo code,

```
MOV     PWMP,#128               ;determine PWM period
MOV     PWM0,#0H                ;duty = 0%
MOV     PWM1,#40H               ;duty = 25%
MOV     PWM2,#80H               ;duty = 50%
MOV     PWM3,#0C0H              ;duty = 75%
MOV     PWM4,#0FFH              ;duty = 100%
ORL     PWMCON0,#00110011b      ;enable PWM0~3
ORL     PWMCON1,#00000001b      ;enable PWM4
ORL     PWMCON0,#11001100b      ;output enable PWM0~3
ORL     PWMCON1,#00000100b      ;output enable PWM4
```

Examples of timed assessing are shown to illustrate correct or incorrect writing processes.

Example 1,

```
MOV     TA,#0AAH            ;2 machine-cycles.
MOV     TA,#55H             ;2 machine-cycles.
ORL     CHPCON,#data        ;2 machine-cycles.
```

Example 2,

```
MOV     TA,#0AAH            ;2 machine-cycles.
MOV     TA,#55H             ;2 machine-cycles.
NOP                         ;1 machine-cycle.
NOP                         ;1 machine-cycle.
ANL     ISPTRG,#data        ;2 machine-cycles.
```

Example 3,

```
MOV     TA,#0AAH            ;2 machine-cycles.
NOP                         ;1 machine-cycle.
MOV     TA,#55H             ;2 machine-cycles.
MOV     WDCON,#data1        ;2 machine-cycles.
ORL     PMC,#data2          ;2 machine-cycles.
```

Example 4,

```
MOV     TA,#0AAH            ;2 machine-cycles.
NOP                         ;1 machine-cycle.
NOP                         ;1 machine-cycle.
MOV     TA,#55H             ;2 machine-cycles.
ANL     WDCON,#data         ;2 machine-cycles.
```

In the first examples, the writing to the protected bits is done before the three-machine-cycle window closes. In example 2, however, the writing to ISPTRG does not complete during the window opening, there will be no change of the value of ISPTRG. In example 3, the WDCON is successful written but the PMC access is out of the three-machine-cycle window. Therefore PMC value will not change either. In Example 4, the second write 55H to TA completes after three machine-cycles of the first write TA of AAH, therefore the timed access window in not opened at all, and the write to the protected bit fails.

In N78E059A/N78E055A, the TA protected SFRs includes CHPCON (9FH), ISPTRG (A4H), PMC (ACH), and WDCON (AAH).

XICON – External Interrupt Control (bit-addressable)

7	6	5	4	3	2	1	0
PX3[1]	EX3	IE3	IT3	PX2[1]	EX2	IE2	IT2
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Address: C0H reset value: 0000 0000b

Bit	Name	Description
7	PX3	**External interrupt 3 priority low bit.**
6	EX3	**Enable external interrupt 3.** 0 = Disable external interrupt 3. 1 = Enable interrupt generated by $\overline{INT3}$ pin (P4.2).
5	IE3	**External interrupt 3 edge flag.** This flag is set via hardware when an edge/level of type defined by IT3 is detected. If IT3 = 1, this bit will remain set until cleared via software or at the beginning of the External Interrupt 3 service routine. If IT3 = 0, this flag is the inverse of the $\overline{INT3}$ input signal's logic level.
4	IT3	**External interrupt 3 type select.** This bit selects whether the $\overline{INT3}$ pin will detect falling edge or low level triggered interrupts. 0 = $\overline{INT3}$ is low level triggered. 1 = $\overline{INT3}$ is falling edge triggered.
3	PX2	**External interrupt 2 priority low bit.**
2	EX2	**Enable external interrupt 2.** 0 = Disable external interrupt 2. 1 = Enable interrupt generated by $\overline{INT2}$ pin (P4.3).
1	IE2	**External interrupt 2 edge flag.** This flag is set via hardware when an edge/level of type defined by IT2 is detected. If IT2 = 1, this bit will remain set until cleared via software or at the beginning of the External Interrupt 2 service routine. If IT2 = 0, this flag is the inverse of the $\overline{INT2}$ input signal's logic level.
0	IT2	**External interrupt 2 type select.** This bit selects whether the $\overline{INT2}$ pin will detect falling edge or low level triggered interrupts. 0 = $\overline{INT2}$ is low level triggered. 1 = $\overline{INT2}$ is falling edge triggered.

[1] PX2 and PX3 are used in combination with the PX2H (IPH.6) and PX3H (IPH.7) respectively to determine the priority of external interrupt 2 and 3. See Table 17–2. Interrupt Priority Level Setting for correct interrupt priority configuration.

The External Interrupts $\overline{INT0}$ and $\overline{INT1}$ can be either edge or level triggered depending on bits IT0 (TCON.0) and IT1 (TCON.2). The bits IE0 (TCON.1) and IE1 (TCON.3) are the flags which are checked to generate the interrupt. In the edge triggered mode, the $\overline{INT0}$ or $\overline{INT1}$ inputs are sampled in every machine-cycle. If the sample is high in one cycle and low in the next, then a high to low transition is detected and the interrupts request flag IE0 or IE1 will be set. Since the external interrupts are sampled every machine-cycle, they have to be held high or low for at least one complete machine-cycle. The IE0 and IE1 are automatically cleared when the interrupt service routine is called. If the level triggered mode is selected, then the requesting source has to

hold the pin low till the interrupt is serviced. The IE0 and IE1 will not be cleared by the hardware on entering the service routine. In the level triggered mode, IE0 and IE1 follows the inverse value of $\overline{INT0}$ and $\overline{INT1}$ pins. If interrupt pins continue to be held low even after the service routine is completed, the processor will acknowledge another interrupt request from the same source. N78E059A/N78E055A (on PLCC-44, PQFP-44, and LQFP-48 packages) possessed other two external interrupts $\overline{INT2}$ and $\overline{INT3}$. Their setting and operation are just the same as interrupt 0 and 1. All configuring bits locate in XICON. The individual interrupt flag corresponding to external interrupt 2 to 3 will also be automatically cleared via hardware once its own interrupt service routine is executed.

The Timer 0 and 1 Interrupts are generated by the TF0 and TF1 flags. These flags are set by the overflow in the Timer 0 and Timer 1 and automatically cleared by the hardware when the timer interrupt is serviced. TF2 or EXF2 flag generates the Timer 2 interrupt. These flags are set by overflow, capture, or reload events in the Timer 2 operation. The hardware will not clear these flags when a Timer 2 interrupt service routine executes. Software has to resolve the cause of the interrupt between TF2 and EXF2 and clear the appropriate flag.

The serial port can generate interrupts on reception or transmission. There are two interrupt sources from the serial port block, which are obtained by the RI and TI bits in the SCON. These bits are not automatically cleared by the hardware. The user has to clear these bits via software.

The Power Down waking-up timer can be used as a simple timer. The Power Down waking-up timer interrupt flag PDTF (PDCON.5) is set once an overflow occurs. If the interrupt is enabled by the enable bit EPDT (EIE.1), then an interrupt will occur.

Brown-out detection, if enabled, can cause Brown-out flag BOF (PMC.3) to be asserted if power voltage drop below Brown-out voltage level. The interrupt will occur if BORST (PMC.4) is 0 and EBOD (EIE.2) is 1.

SPI asserts interrupt flag SPIF (SPSR.7) on completion of data transfer with an external device. If SPI interrupt enable bit ESPI (EIE.0), a serial peripheral interrupt generates. SPIF flag is software clear. MODF (SPSR.4) and SPIOVF (SPSR.5) will also generate SPI interrupt. They share the same vector address with SPIF. When interrupt is generated, the user should tell which flag requires the interrupt.

All the bits that generate interrupts can be set or reset via hardware, and thereby software initiated interrupts can be generated. Each of the individual interrupts can be enabled or disabled by setting or clearing its controlling bit in the IE or EIE. IE also has a global enable bit EA (IE.7) which can be cleared to disable all the interrupts at once. It is set to enable all individually enabled interrupt.

Note that every interrupts, if enabled, is generated by a setting as a logic 1 of its interrupt flag no matter by hardware or software. The user should take care of each interrupt flag in its own interrupt service routine (ISR).

CHPCON – Chip Control (TA protected)

7	6	5	4	3	2	1	0
SWRST	ISPF	LDUEN	XRAMEN	-	-	BS	ISPEN
w	r/w	r/w	r/w	-	-	r/w	r/w

Address: 9FH reset value: see Table 6–2. N78E059A/N78E055A SFR Descriptions and Reset Values

Bit	Name	Description
6	ISPF	**ISP fault flag.** The hardware will set this bit when any of the following condition is met: 1. The accessing area is illegal, such as, (a) Erasing or programming APROM itself when APROM code runs. (b) Erasing or programming LDROM when APROM code runs but LDUEN is 0. (c) Erasing, programming, or reading CONFIG bytes when APROM code runs. (d) Erasing or programming LDROM itself when LDROM code runs. (e) Accessing oversize. 2. The ISP operating runs from internal Program Memory into external one. This bit should be cleared via software.
5	LDUEN	**Updating LDROM enable.** 0 = The LDROM is inhibited to be erased or programmed when APROM code runs. LDROM remains read-only. 1 = The LDROM is allowed to be fully accessed when APROM code runs.
0	ISPEN	**ISP enable.** 0 = Enable ISP function. 1 = Disable ISP function. To enable ISP function will start the internal 22.1184MHz RC oscillator for timing control. To clear ISPEN should always be the last instruction after ISP operation in order to stop internal RC for reducing power consumption.

ISPCN – ISP Control

7	6	5	4	3	2	1	0
ISPA.17	ISPA.16	FOEN	FCEN	FCTRL.3	FCTRL.2	FCTRL.1	FCTRL.0
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Address: AFH reset value: 0000 0000b

Bit	Name	Description
7:6	ISPA[17:16]	**ISP control.** This byte is for ISP controlling command to decide ISP destinations and actions. For details, see Table 18–1. ISP Modes and Command Codes.
5	FOEN	
4	FCEN	
3:0	FCTRL[3:0]	

18.2 ISP Commands

N78E059A/N78E055A provides a wide application to perform ISP to APROM, LDROM or on-chip Data Flash. The ISP action mode and the destination of the flash block are defined by ISP control register ISPCN.

Table 18–1. ISP Modes and Command Codes

ISP Mode	ISPCN				ISPAH, ISPAL ISPA[15:0]	ISPFD[7:0]
	ISPA.17, ISPA.16	FOEN	FCEN	FCTRL[3:0]		
Standby	X, X[1]	1	1	X	X	X
APROM and Data Flash Page Erase	0, 0	1	0	0010	Address in[2]	X
LDROM Page Erase	0, 1	1	0	0010	Address in[2]	X
APROM and Data Flash Program	0, 0	1	0	0001	Address in	Data in
LDROM Program	0, 1	1	0	0001	Address in	Data in
APROM and Data Flash Read	0, 0	0	0	0000	Address in	Data out
LDROM Read	0, 1	0	0	0000	Address in	Data out
All CONFIG bytes Erase	1, 1	1	0	0010	00XXH	X
CONFIG Program	1, 1	1	0	0001	CONFIG0: 0000H CONFIG2: 0002H CONFIG3: 0003H	Data in
CONFIG Read	1, 1	0	0	0000	CONFIG0: 0000H CONFIG2: 0002H CONFIG3: 0003H	Data out

[1] "x" means "don't care".
[2] Each page is 256-byte size. Therefore, the address for Page Erase should be 0000H, 0100H, 0200H, 0300H, etc., which is incremented by one of high byte address.

18.3 User Guide of ISP

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

(1) If no more ISP operation needs, the user must clear ISPEN (CHPCON.0) to zero. It will make the system void to trigger ISP unaware. Furthermore, ISP requires internal 22.1184MHZ RC oscillator running. If the external clock source is chosen, disabling ISP will stop internal 22.1184MHz RC for saving power consumption. Note that a write to ISPEN is TA protected.

```
       ORL    CHPCON,#80h                ;software reset and reboot from APROM

       SJMP   $
;*********************************************************************
;             ISP Function
;*********************************************************************
Enable_ISP:
       MOV    TA,#0Aah                   ;CHPCON is TA protected
       MOV    TA,#55h
       ORL    CHPCON,#00000001b          ;ISPEN = 1, enable ISP mode
       RET
Disable_ISP:
       MOV    TA,#0Aah                   ;CHPCON is TA protected
       MOV    TA,#55h
       ANL    CHPCON,#11111110b          ;ISPEN = 0, disable ISP mode
       RET
Trigger_ISP:
       MOV    TA,#0Aah
       MOV    TA,#55h
       ORL    ISPTRG,#00000001b          ;write '1' to ISPGO to trigger ISP process
       RET
;*********************************************************************
;             ISP AP Function
;*********************************************************************
Erase_AP:
       MOV    ISPCN,#PAGE_ERASE_AP
       MOV    ISPAL,#00h
       MOV    R0,#00h
Erase_AP_Loop:
       MOV    ISPAH,R0
       CALL   Trigger_ISP
       INC    R0
       CJNE   R0,#0,Erase_AP_Loop
       RET
Erase_AP_Verify:
       MOV    ISPCN,#BYTE_READ_AP
       MOV    ISPAH,#00h
       MOV    ISPAL,#00h
Erase_AP_Verify_Loop:
       MOV    ISPFD,#00h                 ;clear ISPFD Data
       CALL   Trigger_ISP
       MOV    A,ISPFD
       CJNE   A,#0FFh,Erase_AP_Verify_Error
       INC    ISPAL
       MOV    A,ISPAL
       CJNE   A,#0,Erase_AP_Verify_Loop
       INC    ISPAH
       MOV    A,ISPAH
       CJNE   A,#0,Erase_AP_Verify_Loop
       RET
Erase_AP_Verify_Error:
       CALL   Disable_ISP
       mov    P0,#00h
       SJMP   $
Program_AP:
       MOV    ISPCN,#BYTE_PROGRAM_AP
       MOV    ISPAH,#00h
       MOV    ISPAL,#00h
       MOV    DPTR,#AP_code
Program_AP_Loop:
```

```
        MOV     A,#0
        MOVC    A,@A+DPTR
        MOV     ISPFD,A
        CALL    Trigger_ISP
        INC     DPTR
        INC     ISPAL
        MOV     A,ISPAL
        CJNE    A,#8,Program_AP_Loop
        RET
Program_AP_Verify:
        MOV     ISPCN,#BYTE_READ_AP
        MOV     ISPAH,#00h
        MOV     ISPAL,#00h
        MOV     DPTR,#AP_code
Program_AP_Verify_Loop:
        MOV     ISPFD,#00h                  ;clear ISPFD Data
        CALL    Trigger_ISP
        MOV     A,#0
        MOVC    A,@A+DPTR
        MOV     B,A
        MOV     A,ISPFD
        CJNE    A,B,Program_AP_Verify_Error
        INC     DPTR
        INC     ISPAL
        MOV     A,ISPAL
        CJNE    A,#8,Program_AP_Verify_Loop
        RET
Program_AP_Verify_Error:
        CALL    Disable_ISP
        mov     P0,#00h
        SJMP    $
;*********************************************************************
;             ISP Config Function
;*********************************************************************
Erase_Config:
        MOV     ISPCN,#ALL_ERASE_CONFIG
        MOV     ISPAH,#00h
        CALL    Trigger_ISP
        RET
Read_Config:
        MOV     ISPCN,#BYTE_READ_CONFIG
        MOV     ISPAH,#00h
        MOV     ISPAL,#03h
        CALL    Trigger_ISP
        MOV     A,ISPFD
        RET
Program_Config:
        MOV     ISPCN,#BYTE_PROGRAM_CONFIG
        MOV     ISPAH,#00h
        MOV     ISPAL,#03h
        ANL     A,#10111111b
        MOV     ISPFD,A                     ;switch to 6T mode
        MOV     R0,A                        ;temp data
        CALL    Trigger_ISP
        RET
Program_Config_Verify:
        MOV     ISPCN,#BYTE_READ_CONFIG
        MOV     ISPAH,#00h
        MOV     ISPAL,#03h
        MOV     ISPFD,#00h                  ;clear ISPFD Data
```

CONFIG2

7	6	5	4	3	2	1	0
CBODEN	CBOV1	CBOV0	CBORST	-	-	-	-
r/w	r/w	r/w	r/w	-	-	-	-

unprogrammed value: 1111 1111b

Bit	Name	Description
7	CBODEN	**CONFIG Brown-out detect enable.** 1 = Enable Brown-out detection. 0 = Disable Brown-out detection.
6 5	CBOV1 CBOV0	**CONFIG Brown-out voltage select.** These two bits select one of four Brown-out voltage level. CBOV1 CBOV0 Brown-out Voltage 1 1 2.2V 1 0 2.7V 0 1 3.8V 0 0 4.5V
4	CBORST	**CONFIG Brown-out reset enable.** This bit decides if a Brown-out reset is caused after a Brown-out event. 1 = Enable Brown-out reset when V_{DD} drops below V_{BOD}. 0 = Disable Brown-out reset when V_{DD} drops below V_{BOD}.

PMC – Power Monitoring Control (TA protected)

7	6	5	4	3	2	1	0
BODEN[1]	-	-	BORST[1]	BOF	LPBOD	-	BOS
r/w	-	-	r/w	r/w	r/w	-	r

Address: ACH reset value: see Table 6–2. N78E059A/N78E055A SFR Descriptions and Reset Values

Bit	Name	Description
7	BODEN	**Brown-out detect enable.** 0 = Disable Brown-out detection. 1 = Enable Brown-out detection.
6:5	-	**Reserved.**
4	BORST	**Brown-out reset enable.** This bit decides if a Brown-out reset is caused after a Brown-out event. 0 = Disable Brown-out reset when V_{DD} drops below V_{BOD}. 1 = Enable Brown-out reset when V_{DD} drops below V_{BOD}.
3	BOF	**Brown-out flag.** This flag will be set as a logic 1 via hardware after a V_{DD} dropping below or rising above V_{BOD} event occurs. If both EBOD (EIE.2) and EA (IE.7) are set, a Brown-out interrupt requirement will be generated. This bit must be cleared via software.
3	LPBOD	**Low power Brown-out detection enable.** This bit switches the Brown-out detection into a power saving mode. This bit is only effective while BODEN = 1. 0 = Disable Brown-out power saving mode. Brown-out detection operates in normal mode if enabled. The detection is always on. 1 = Enable Brown-out power saving mode. Brown-out detection operates in power saving mode if enabled. Enable this bit will switch on internal 10kHz RC to be a timer for about 12.8ms interval of detection. The discrete detection will save much power but the hysteresis feature disappears.

23. AUXILIARY FEATURES

ALE is used to enable the address latch that separates the address from the data on Port 0. ALE runs at 1/6 of the Fosc in 12T mode. An ALE pulse is omitted always. The user can turn ALE signal off via setting ALEOFF to reduce EMI. ALEOFF enable will just make ALE activating during external memory access through a MOVC or MOVX instruction. ALE will stay high in other conditions.

AUXR – Auxiliary Register

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	ALEOFF
-	-	-	-	-	-	-	r/w

Address: 8EH reset value: 0000 0000b

Bit	Name	Description
7:1	-	**Reserved.**
0	ALEOFF	**ALE output off.** 0 = ALE is emitted always. 1 = ALE is off normally and active only during external memory access through a MOVX or MOVC instruction.

Figures below shows supply and Idle mode current under 12T/6T with internal program memory mode.

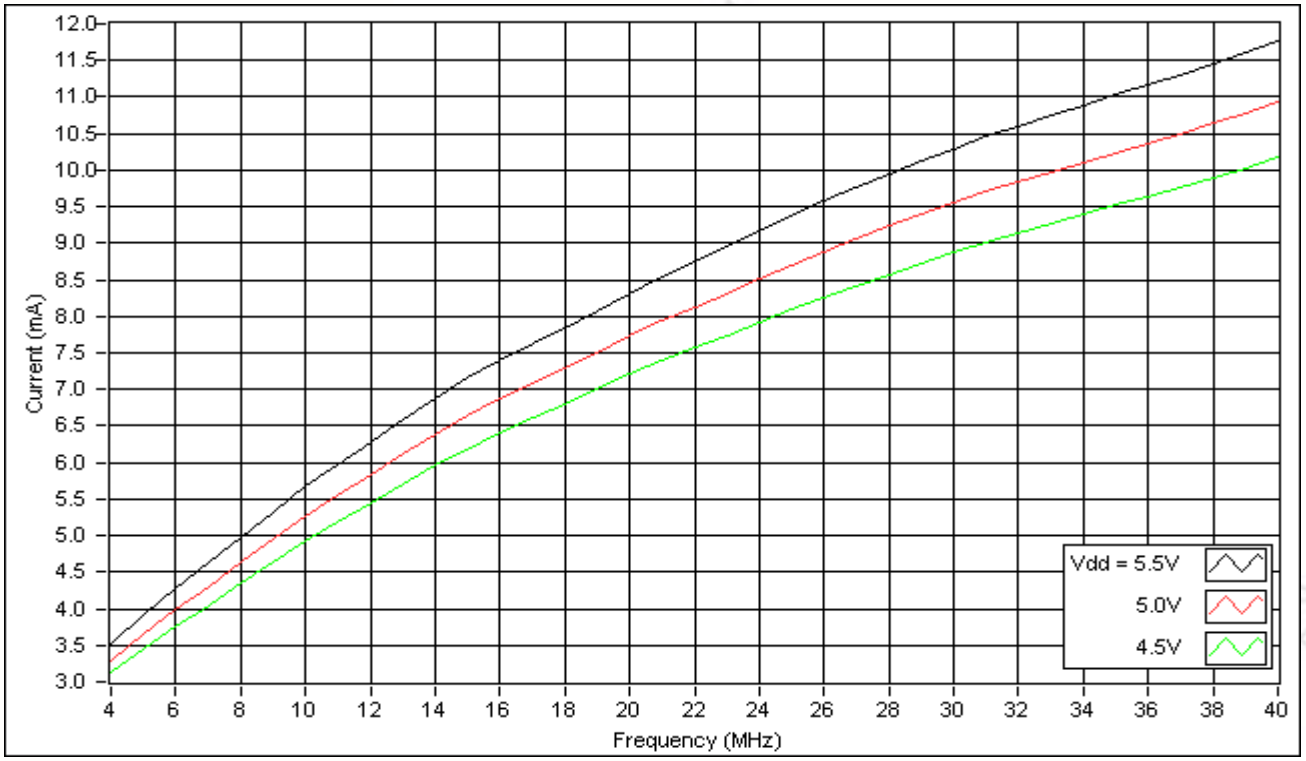


Figure 26–1. Supply Current Under 12T Mode, External Clock (1)

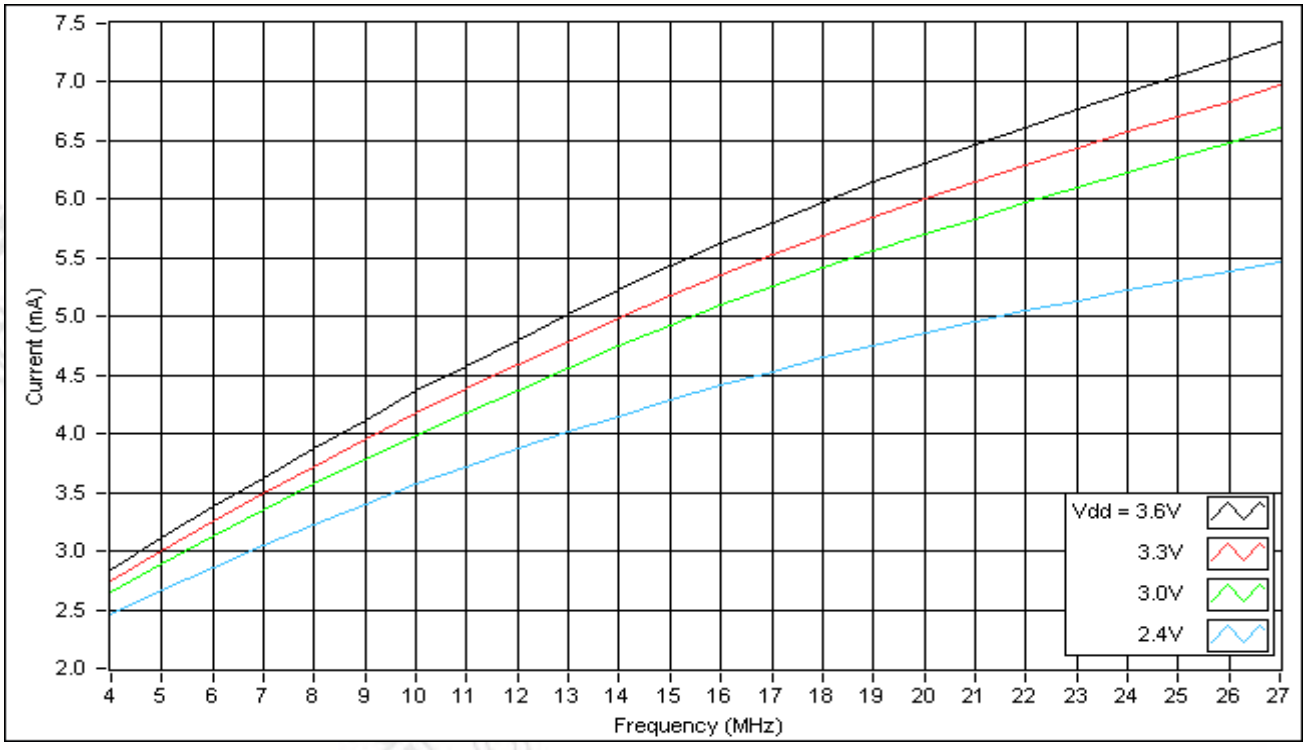
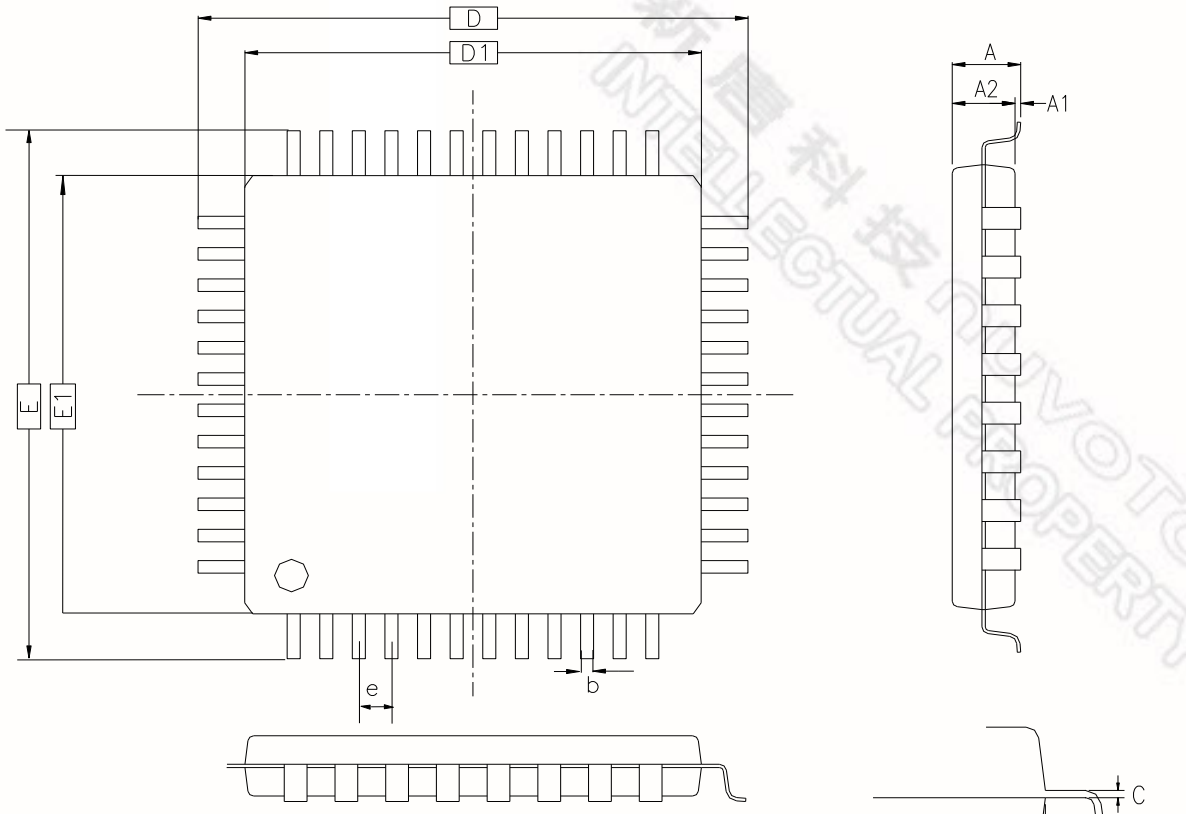


Figure 26–2. Supply Current Under 12T Mode, External Clock (2)



COTROL DIMENSIONS ARE IN MILLIMETERS.

SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	—	—	1.60	—	—	0.063
A_1	0.05	0.10	0.15	0.002	0.004	0.006
A_2	1.35	1.40	1.45	0.053	0.055	0.057
D1	6.90	7.00	7.10	0.272	0.276	0.260
E1	6.90	7.00	7.10	0.272	0.276	0.260
e	0.35	0.50	0.65	0.014	0.020	0.260
D	8.9	9.00	9.10	0.350	0.354	0.358
E	8.9	9.00	9.10	0.350	0.354	0.358
L	0.45	0.60	0.75	0.018	0.024	0.030
L1	—	1.00	—	—	0.039	—
C	0.09	—	0.20	0.0035	—	0.0079
θ	0°	—	7°	0°	—	7°
b	0.17	0.22	0.27	0.007	0.0087	0.011

Figure 27–4. LQFP-48 Package Dimention