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	Last Time Buy
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	36
Program Memory Size	64KB (64K 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	Surface Mount
Package / Case	44-BQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/n78e517afg

TABLE OF CONTENTS

4. PIN CONFIGURATIONS

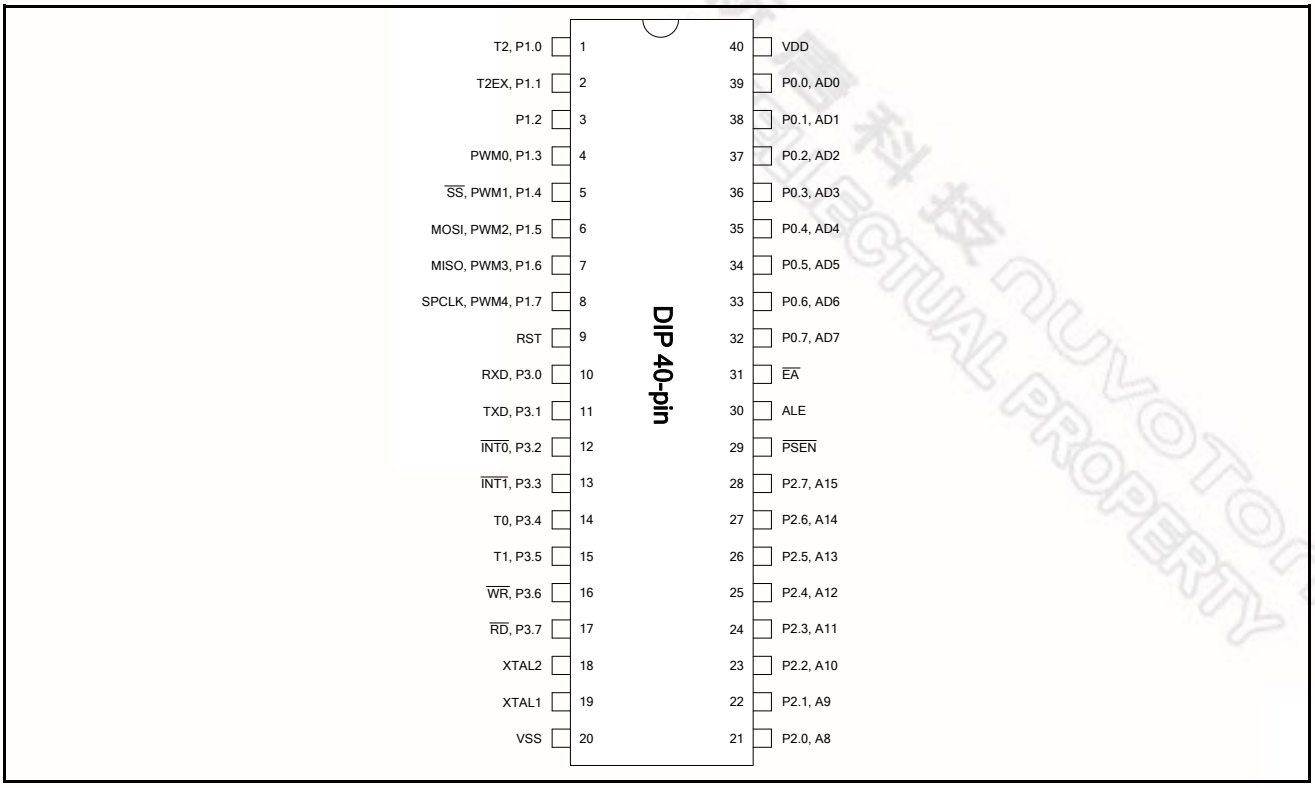

Figure 4–1. Pin Assignment of DIP 40-Pin

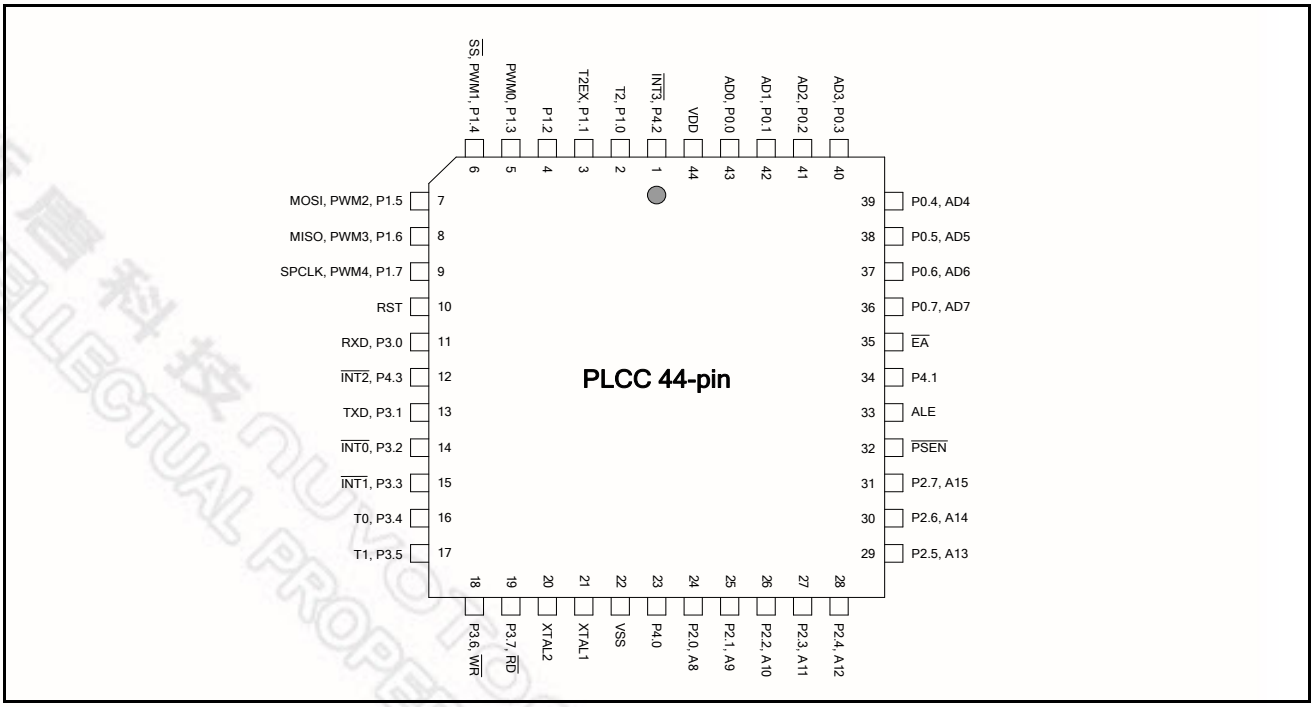

Figure 4–2. Pin Assignment of PLCC 44-Pin

Table 4–1. Pin Description

Pin number				Symbol	Alternate Function		Type[1]	Description
DIP	PLCC	PQFP TQFP	LQFP		1	2		
28	31	25	27	P2.7		A15	I/O	
10	11	5	5	P3.0	RXD		I/O	**PORT3:** Port 3 is an 8-bit quasi bi-directional I/O port. Its multifunction pins are for RXD, TXD, $\overline{INT0}$, $\overline{INT1}$, T0, T1, $\overline{WR}$, and $\overline{RD}$.
11	13	7	7	P3.1	TXD		I/O	
12	14	8	8	P3.2	$\overline{INT0}$		I/O	
13	15	9	9	P3.3	$\overline{INT1}$		I/O	
14	16	10	10	P3.4	T0		I/O	
15	17	11	11	P3.5	T1		I/O	
16	18	12	13	P3.6	$\overline{WR}$		I/O	
17	19	13	14	P3.7	$\overline{RD}$		I/O	
-	23	17	18	P4.0			I/O	**PORT4[3]:** Port 4 is an 8-bit quasi bi-directional I/O port. It also possesses bit-addressable feature as P0-P3. P4.2 and P4.3 are alternative function pins of $\overline{INT3}$ and $\overline{INT2}$.
-	34	28	30	P4.1			I/O	
-	1	39	42	P4.2	$\overline{INT3}$		I/O	
-	12	6	6	P4.3	$\overline{INT2}$		I/O	
-	-	-	48	P4.4			I/O	
-	-	-	12	P4.5			I/O	
-	-	-	24	P4.6			I/O	
-	-	-	36	P4.7			I/O	

[1] I/O type description. I: input, O: output, I/O: quasi bi-direction, D: open-drain, P: power pins, ST: Schmitt trigger.
[2] While switching to 6T mode, ALE will run at 1/3 of Fosc.
[3] A full 8-bit P4 is just on LQPF-48 package. PLCC-44, PQFP-44, and TQFP-44 just have low nibble 4 bits of P4. DIP-40 does not have this additional P4.

5.2 External Program Memory

N78E517A is a 16-bit address-width CPU. It can address 64k-byte program code. Besides the internal Program Memory, the external additional Program Memory is also can be used. The external program addressing will be executed under cases below,

1. The PC (Program Counter) value is beyond the boundary size address of APROM or LDROM while $\overline{EA}$ pin is pulled high during power on. The CPU will continue to fetch the external Program Memory.

2. While $\overline{EA}$ pin is pulled low during power on period, The CPU will run totally 64k-byte code externally.

While the external mode is running, the P0 and P2 will produce address and data signals to fetching external Program Memory. In this case, P0 and P2 cannot be general purpose I/O anymore. $\overline{PSEN}$ will also toggle out to strobe the external Program Memory. For the hardware circuit for external program execution, see Figure 5–2. Program Memory Interface.

For security $\overline{EA}$ pin state will be locked after power on. The user cannot switch the program running internally or externally by $\overline{EA}$ after power on. The other design for data security is MOVC lock enable (MOVCL, CONFIG0.2). While this bit is set 0, The external Program Memory code is inhibited to read internal APROM or LDROM contents through MOVC instruction.

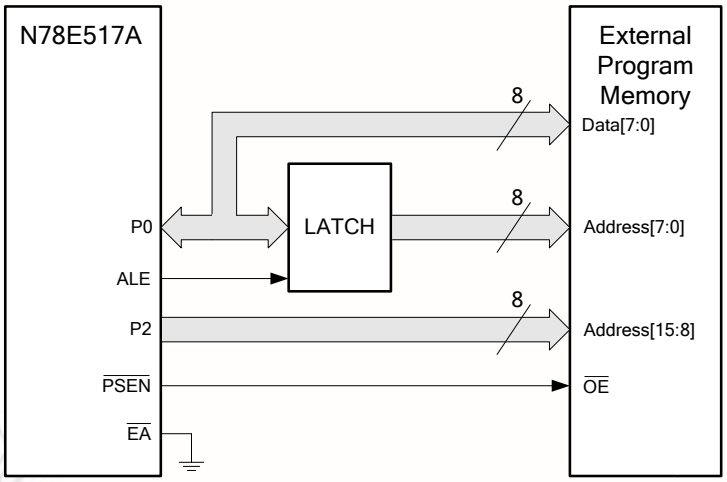

Figure 5–2. Program Memory Interface

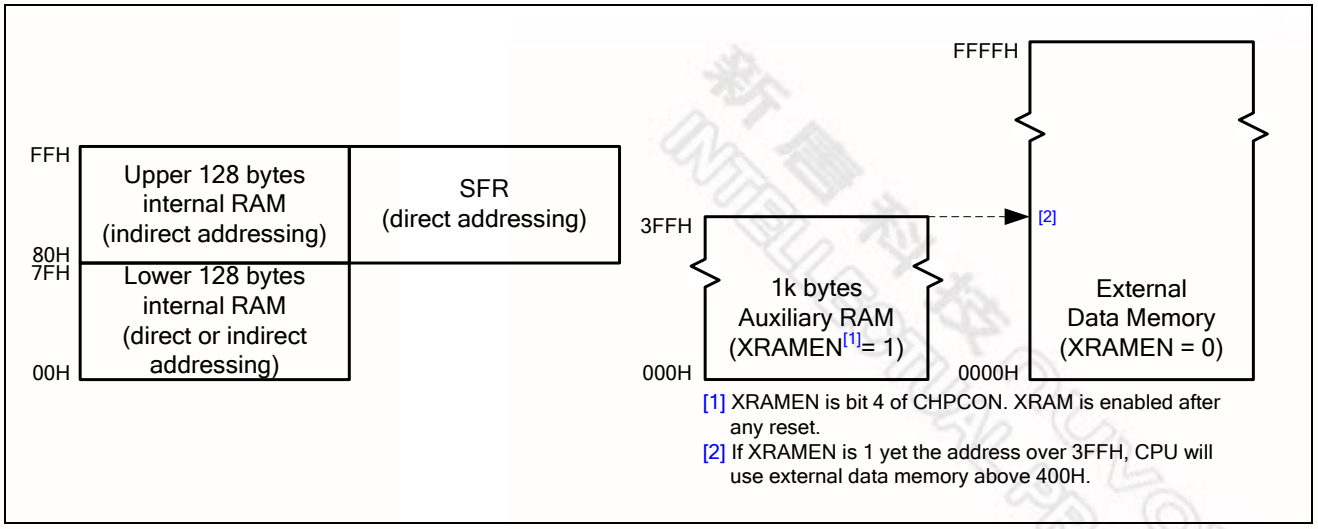

Figure 5–3. N78E517A Data Memory Structure

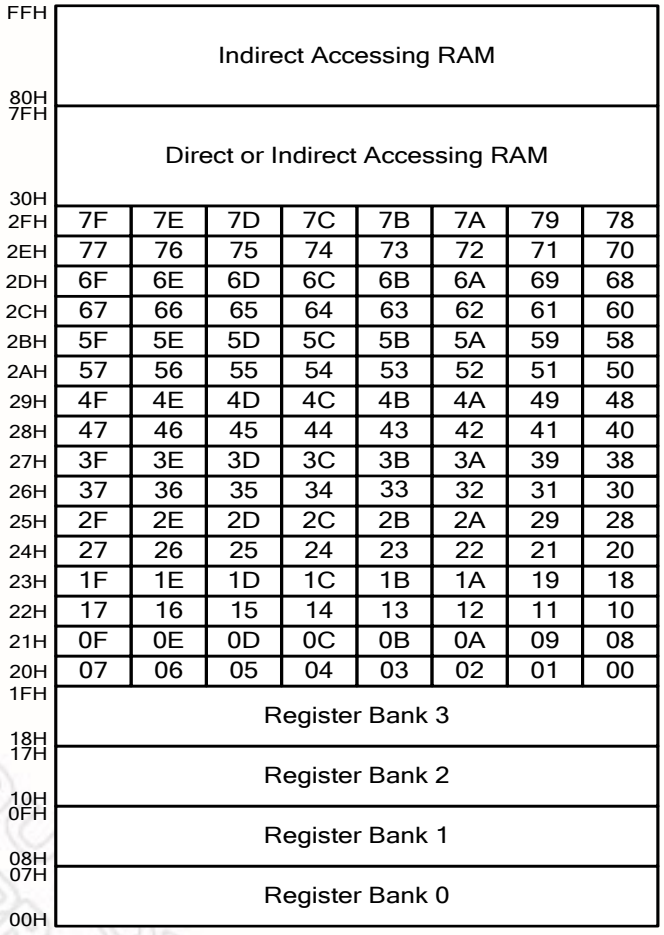

Figure 5–4. 256 bytes Internal RAM Addressing

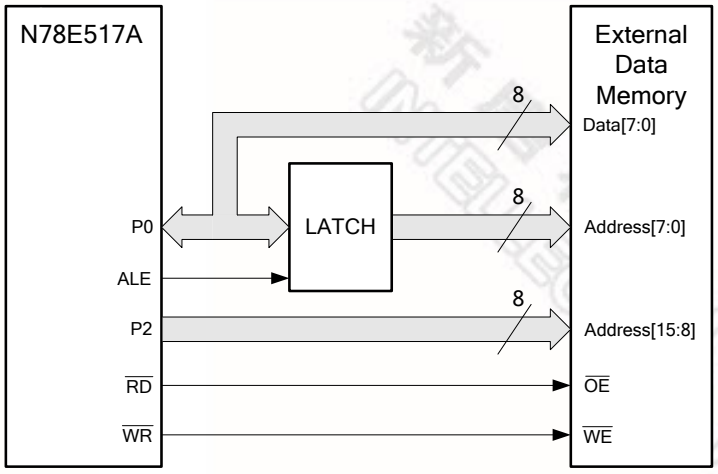

Figure 5–5. Data Memory Interface

5.6 On-chip Non-volatile Data Flash

N78E517A additionally has Data Flash. The Data Flash is non-volatile so that it remains its content even after the power is off. Therefore, in general application the user can write or read data which rules as parameters or constants. Be aware of Data Flash writing endurance of 10,000 cycles. By the software path, the Data Flash can be accessed only through ISP mode. Note that the erasing or writing of Data Flash should not operates under V_{DD} 3.0V for ISP limitation. For Data Flash accessing with ISP, please see Section 18. “IN SYSTEM PROGRAMMING (ISP)” on page 92 for details. For the design for security, ISP is invalid while external Program Memory executes. The Data Flash, therefore, cannot be accessed with external memory code. Of course the Data Flash can be accessed via hardware with parallel Programmer/Writer.

The Data Flash size is software adjustable on N78E517A by updating the content of SHBDA. SHBDA[7:0] represents the high byte of 16-bit Data Flash start address and the low byte are hardware set to 00H. The value of SHBDA is loaded from the content of CONFIG1 (CHBDA) after all resets. The application program can dynamically adjust the Data Flash size by resetting SHBDA value. Once the Data Flash size is changed the APROM size is changed accordingly. SHBDA has time access protect while a write to SHBDA is required. Be aware that if CHBDA is 00H, the Data Flash size will be 64k bytes and there will be no APROM. CPU will execute codes in the external Program Memory.

The CONFIG bit DFEN (CONFIG0.0) should be programmed as a 0 before access the Data Flash block. If DFEN remains its unprogrammed value 1, APROM will occupy whole 64k-byte block.

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

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

Address: 9CH reset value: see Table 6–2. N78E517A 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.

PCON – Power Control

7	6	5	4	3	2	1	0
SMOD	-	-	POF	GF1	GF0	PD	IDL
r/w	-	-	r/w	r/w	r/w	r/w	r/w

Address: 87H reset value: see Table 6–2. N78E517A SFR Description and Reset Values

Bit	Name	Description
3	GF1	**General purpose flag 1.** The general purpose flag that can be set or cleared by the user.
2	GF0	**General purpose flag 0.** The general purpose flag that can be set or cleared by the user.

Bit	Name	Description
3	EXEN2	**Timer 2 external enable.** This bit enables 1-to-0 transitions on T2EX trigger. 0 = 1-to-0 transitions on T2EX is ignored. 1 = 1-to-0 transitions on T2EX will set EXF2 logic 1. If Timer 2 is configured in capture or auto-reload mode, the 1-to-0 transitions on T2EX will cause capture or reload event.
2	TR2	**Timer 2 run control.** 0 = Timer 2 is halted. Clearing this bit will halt Timer 2 and the current count will be preserved in TH2 and TL2. 1 = Timer 2 is enabled.
1	C/$\overline{T2}$	**Timer 2 Counter/Timer select.** 0 = Timer 2 is incremented by internal peripheral clocks. 1 = Timer 2 is incremented by the falling edge of the external pin T2. If Timer 2 would like to be set in clock-out mode, C/$\overline{T2}$ must be 0.
0	CP/$\overline{RL2}$	**Timer 2 Capture or Reload select.** This bit selects whether Timer 2 functions in capture or auto-reload mode. EXEN2 must be logic 1 for 1-to-0 transitions on T2EX to be recognized and used to trigger captures or reloads. If RCLK or TCLK is set, this bit is ignored and Timer 2 will function in auto-reload mode. 0 = Auto-reload on Timer 2 overflow or 1-to-0 transition on T2EX pin. 1 = Capture on 1-to-0 transition at T2EX pin.

T2MOD – Timer 2 Mode

7	6	5	4	3	2	1	0
-	-	-	-	-	-	T2OE	-
-	-	-	-	-	-	r/w	-

Address: C9H reset value: 0000 0000b

Bit	Name	Description
7:2	-	**Reserved.**
1	T2OE	**Timer 2 clock-out enable.** 0 = Disable Timer 2 clock-out function. T2 pin functions either as a standard port pin or as a counter input for Timer 2. 1 = Enable Timer 2 clock-out function. Timer 2 will drive T2 pin with a clock output if C/$\overline{T2}$ is 0.
0	-	**Reserved.**

RCAP2L – Timer 2 Reload/Capture Low Byte

7	6	5	4	3	2	1	0
RCAP2L[7:0]							
r/w							

Address: CAH reset value: 0000 0000b

Bit	Name	Description
7:0	RCAP2L[7:0]	**Timer 2 reload/capture low byte.** This register captures and stores the low byte of Timer 2 when Timer 2 is configured in capture mode. When Timer 2 is in auto-reload mode, baud rate generator mode, or clock-out mode, it holds the low byte of the reload value.

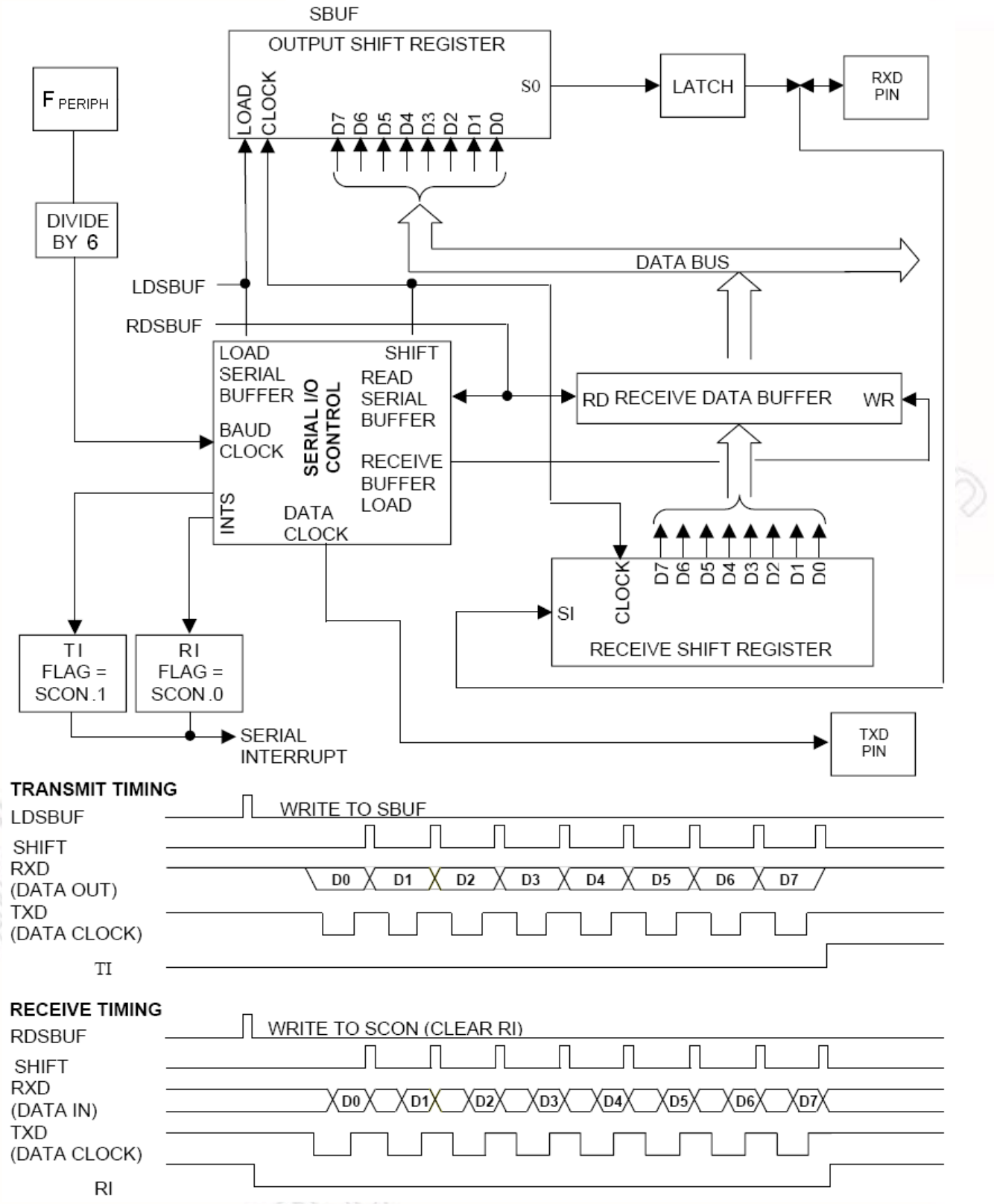

Figure 13–1. Serial Port Mode 0 Function Block and Timing Diagram

By default, SPI data is transferred MSB first. If the LSBFE (SPCR.5) is set, SPI data shifts LSB first. This bit does not affect the position of the MSB and LSB in the data register. Note that all following description and figures are under the condition of LSBFE logic 0. MSB is transmitted and received first.

14.3 Control Registers of SPI

There are three SPI registers to support its operations, they are SPI control register (SPCR), SPI status register (SPSR), and SPI data register (SPDR). These registers provide control, status, data storage functions, and clock rate selection. The following registers relate to SPI function.

SPCR – Serial Peripheral Control Register

7	6	5	4	3	2	1	0
SSOE	SPIEN	LSBFE	MSTR	CPOL	CPHA	SPR1	SPR0
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Address: F3H reset value: 0000 0000b

Bit	Name	Description
7	SSOE	**Slave select output enable.** This bit is used in combination with the DISMODF (SPSR.3) bit to determine the feature of $\overline{SS}$ pin as shown in Table 14–1. Slave Select Pin Configurations. This bit takes effect only under MSTR = 1 and DISMODF = 1 condition. 0 = $\overline{SS}$ functions as a general purpose I/O pin. 1 = $\overline{SS}$ automatically goes low for each transmission when selecting external Slave device and goes high during each idle state to de-select the Slave device.
6	SPIEN	**SPI enable.** 0 = Disable SPI function. 1 = Enable SPI function.
5	LSBFE	**LSB first enable.** 0 = The SPI data is transferred MSB first. 1 = The SPI data is transferred LSB first.
4	MSTR	**Master mode enable.** This bit switches the SPI operating between Master and Slave modes. 0 = The SPI is configured as Slave mode. 1 = The SPI is configured as Master mode.
3	CPOL	**SPI clock polarity select.** CPOL bit determines the idle state level of the SPI clock. See Figure 14–4. SPI Clock Formats. 0 = The SPI clock is low in idle state. 1 = The SPI clock is high in idle state.
2	CPHA	**SPI clock phase select.** CPHA bit determines the data sampling edge of the SPI clock. See Figure 14–4. SPI Clock Formats. 0 = The data is sampled on the first edge of the SPI clock. 1 = The data is sampled on the second edge of the SPI clock.

15. PULSE WIDTH MODULATOR (PWM)

N78E517A provides five pulse width modulated (PWM) output channels to generate pulses of programmable length and interval. Five PWM channels, PWM0-4, shares the same pins with P1.3-P1.7. The PWM period is defined by an 8-bit pre-scalar PWMP, which supplies the clock of the PWM counter. The pre-scalar is common for all PWM channels. The duty of each PWM channel is determined by the value of five registers, PWM0, PWM1, PWM2, PWM3, and PWM4. If the contents of these registers are equal to or less than the 8-bit counter value, the output will be 0. Else the output will be 1 if these registers value are larger than the counter. Set PWMxEN (in PWMCON0[0,1,4,5] and PWMCON1.0) will enable to run or disable to stop each PWM channel respectively. In addition, the PWMxOM (in PWMCON0[2,3,6,7] and PWMCON1.2) must set 1 to output the internal PWM signal to port pins. Without setting PWMxOM, the pins which share with alternative PWM function will be normal general purpose I/O of P1.3-P1.7 even though PWM is enabled. The following registers relate to PWM function.

PWMCON0 – PWM Control 0

7	6	5	4	3	2	1	0
PWM3OE	PWM2OE	PWM3EN	PWM2EN	PWM1OE	PWM0OE	PWM1EN	PWM0EN
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Address: DCH reset value: 0000 0000b

Bit	Name	Description
7	PWM3OE	**PWM3 output enable.** 0 = P1.6 serves as general purpose I/O. 1 = P1.6 serves as output pin of PWM3 signal.
6	PWM2OE	**PWM2 output enable.** 0 = P1.5 serves as general purpose I/O. 1 = P1.5 serves as output pin of PWM2 signal.
5	PWM3EN	**PWM3 enable.** 0 = PWM3 is disabled and stops. 1 = PWM3 is enabled and runs.
4	PWM2EN	**PWM2 enable.** 0 = PWM2 is disabled and stops. 1 = PWM2 is enabled and runs.
3	PWM1OE	**PWM1 output enable.** 0 = P1.4 serves as general purpose I/O. 1 = P1.4 serves as output pin of PWM1 signal.
2	PWM0OE	**PWM0 output enable.** 0 = P1.3 serves as general purpose I/O. 1 = P1.3 serves as output pin of PWM0 signal.
1	PWM1EN	**PWM1 enable.** 0 = PWM1 is disabled and stops. 1 = PWM1 is enabled and runs.

16. TIMED ACCESS PROTECTION (TA)

N78E517A has several features like the Watchdog Timer, the ISP function, Boot select control, etc. are crucial to proper operation of the system. If leaving these control registers unprotected, errant code may write undetermined value into them, it results in incorrect operation and loss of control. In order to prevent this risk, the N78E517A has a protection scheme which limits the write access to critical SFRs. This protection scheme is done using a timed access. The following registers are related to TA process.

TA – Timed Access

7	6	5	4	3	2	1	0
TA[7:0]							
w							

Address: C7H reset value: 0000 0000b

Bit	Name	Description
7:0	TA[7:0]	**Timed access.** The timed access register controls the access to protected SFRs. To access protected bits, the user must first write AAH to the TA and immediately followed by a write of 55H to TA. After these two steps, a writing permission window is opened for three machine-cycles during which the user may write to protected SFRs.

In timed access method, the bits, which are protected, have a timed write enable window. A write is successful only if this window is active; otherwise, the write will be discarded. When the software writes AAH to TA, a counter is started. This counter waits for three machine-cycles looking for a write of 55H to TA. If the second write of 55H occurs within three machine-cycles of the first write of AAH, then the timed access window is opened. It remains open for three machine-cycles during which the user may write to the protected bits. After three machine-cycles, this window automatically closes. Once the window closes, the procedure must be repeated to access the other protected bits. Not that the TA protected SFRs are required timed access for writing. But the reading is not protected. The user may read TA protected SFR without giving AAH and 55H to TA. The suggestion code for opening the timed access window is shown below.

```
(CLR    EA)                 ;if any interrupt is enabled, disable temporally
 MOV    TA,#0AAH
 MOV    TA,#55H
(Instruction that writes a TA protected register)
(SETB   EA)                 ;resume interrupts enabled
```

The writings of AAH, 55H to TA register and the writing-protection register must occur within 3 machine-cycles of each other. Any enabled interrupt should be disabled during this procedure to avoid delay between these three writings. If there is no interrupt enabled, the `CLR EA` and `SETB EA` instructions can be left out. Once the timed access window closes, the procedure must be repeated to access the other protected bits.

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 N78E517A, the TA protected SFRs includes CHPCON (9FH), ISPTRG (A4H), PMC (ACH), SHBDA (9CH), and WDCON (AAH).

pleted. RET would leave the controller still thinking that the service routine is underway, making future interrupts impossible.

17.2 Interrupt Latency

The response time for each interrupt source depends on several factors, such as the nature of the interrupt and the instruction underway. In the case of external interrupts $\overline{INT0}$ and $\overline{INT1}$, they are sampled at every machine-cycle and then their corresponding interrupt flags IE0 or IE1 will be set or reset. The value are not actually polled by the circuit until the next machine-cycle. If a request is active and all three previous conditions are met, then the hardware generated LCALL is executed. This LCALL itself takes 2 machine-cycles to be completed. Thus there is a minimum time of 3 machine-cycles between the interrupt flag being set and the interrupt service routine being executed.

A longer response time should be anticipated if any of the three conditions are not met. If a higher or equal priority is being serviced, then the interrupt latency time obviously depends on the nature of the service routine currently being executed. If the polling cycle is not the last machine-cycle of the instruction being executed, then an additional delay is introduced. The maximum response time (if no other interrupt is in service) occurs if the device is performing a write to IE, IP and then executes a MUL or DIV instruction. From the time an interrupt source is activated, the longest reaction time is 9 machine-cycles. This time includes 1 machine-cycle to detect the interrupt, 2 machine-cycles to complete the IE, EIE, IP, IPH, EIP, or EIPH access, 4 machine-cycles to complete the MUL or DIV instruction and 2 machine-cycles to complete the hardware LCALL to the interrupt vector location.

Thus in a single-interrupt system the interrupt response time will always be more than 3 machine-cycles and not more than 9 machine-cycles.

(2) If the loader code, which controls the ISP procedure, locates in the external Program Memory or runs from the internal into the external, the ISP will not work anymore and set error indicator ISPF for data security.

(3) CONFIG bytes can be ISP fully accessed only when loader code executing in LDROM. New CONFIG bytes other than CBS bit activate after all resets. New CBS bit activates after resets other than software reset.

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

(5) ISP erasing or programming works from V_{DD} 3.0V through 5.5V.

(6) APROM and LDROM can read itself through ISP method.

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

Note that If the user would like to develop your own ISP program, remember always erase and program CONFIG bytes at the last step for data security.

18.4 ISP Demo Code

```
;*****************************************************************************
;      This code illustrates how to do APROM and CONFIG ISP from LDROM.
;      APROM are re-programmed by the code to output P1 as 55h and P2 as aah.
;      The CONFIG3 is also updated to 6T mode.
;      The user should put this code in LDROM and boot from LDROM.
;*****************************************************************************
PAGE_ERASE_AP              EQU          00100010b
BYTE_PROGRAM_AP            EQU          00100001b
BYTE_READ_AP               EQU          00000000b
BYTE_READ_CONFIG           EQU          11000000b
BYTE_PROGRAM_CONFIG        EQU          11100001b
ALL_ERASE_CONFIG           EQU          11100010b

       ORG    0000h

       CALL   Enable_ISP
       CLR    EA                        ;disable all interrupts
       CALL   Erase_AP                  ;erase AP data
       CALL   Erase_AP_Verify           ;verify Erase AP data
       CALL   Program_AP                ;programming AP data
       CALL   Program_AP_Verify         ;verify Programmed AP data
       CALL   Read_Config               ;read back CONFIG3
       CALL   Erase_Config              ;erase CONFIG bytes
       CALL   Program_Config            ;programming CONFIG3 with new data
       CALL   Program_Config_Verify     ;verify Programmed CONFIG3
       CALL   Disable_ISP
       MOV    TA,#0Aah                  ;TA protection
       MOV    TA,#55h                   ;
       ANL    CHPCON,#0FDh              ;BS = 0, reset to APROM
       MOV    TA,#0Aah
       MOV    TA,#55h
       ORL    CHPCON,#80h               ;software reset and reboot from APROM
```

CONFIG3

7	6	5	4	3	2	1	0
CWDTEN	EN6T	ROG	CKF	INTOSCFS	-	FOSC	-
r/w	r/w	r/w	r/w	r/w	-	r/w	-

unprogrammed value: 1111 1111b

Bit	Name	Description
6	EN6T	**Enable 6T mode.** This bit switches MCU between 12T and 6T mode. See Figure 20–1. Clock System Block Diagram for definitions in details. 1 = MCU runs at 12T mode. Each machine-cycle is equal to 12 clocks of system oscillator. The operating mode is the same as a standard 8051 MCU. (F_{CPU} and F_{PERIPH} is a half of F_{OSC}.) 0 = MCU runs at 6T mode. Each machine-cycle is equal to 6 clocks of system oscillator. This mode doubles the whole chip operation compared with the standard 8051. (F_{CPU} and F_{PERIPH} is equal to F_{OSC}.)
5	ROG	**Reducing oscillator gain.** 1 = Use normal gain for crystal oscillating. The crystal frequency can be up to 40MHz. 0 = Use reduced gain for crystal oscillating. The crystal frequency should be lower than 24MHz. In reduced gain mode, it will also help to decrease EMI.
4	CKF	**Clock filter enable.** 1 = Enable clock filter. It increases noise immunity and EMC capacity. 0 = Disable clock filter.
3	INTOSCFS	**Internal RC oscillator frequency select.** 1 = Select 22.1184MHz as the system clock if internal RC oscillator mode is used. It bypasses the divided-by-2 path of internal oscillator to select 22.1184MHz output as the system clock source. 0 = Select 11.0592MHz as the system clock if internal RC oscillator mode is used. The internal RC divided-by-2 path is selected. The internal oscillator is equivalent to 11.0592MHz output used as the system clock.
2	-	**Reserved.**
1	FOSC	**Oscillator selection bit.** This bit selects the source of the system clock. 1 = Crystal, resonator, or external clock input. 0 = Internal RC oscillator.
0	-	**Reserved.**

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. N78E517A SFR Description and Reset Values

Bit	Name	Description
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.

22. RESET CONDITIONS

N78E517A has several options to place device in reset condition. It also offers the software flags to indicate the source, which causes a reset. In general, most SFRs go to their reset value irrespective of the reset condition, but there are several reset source indicating flags whose state depends on the source of reset. The user can read back these flags to determine the cause of reset using software. There are 5 ways of putting the device into reset state. They are power-on reset, RST pin reset, software reset, Watchdog Timer reset, and Brown-out reset.

RSR – Reset Status Register

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

Address: 96H reset value: see Table 6–2. N78E517A SFR Description and Reset Values

Bit	Name	Description
7:3	-	**Reserved.**
2	BORF	**Brown-out reset flag.** When the MCU is reset by Brown-out reset, this bit will be set via hardware. This flag is recommended to be cleared via software.
1	-	**Reserved.**
0	SWRF	**Software reset flag.** When the MCU is reset via software reset, this bit will be set via hardware. This flag is recommended to be cleared via software.

PCON – Power Control

7	6	5	4	3	2	1	0
SMOD	-	-	POF	GF1	GF0	PD	IDL
r/w	-	-	r/w	r/w	r/w	r/w	r/w

Address: 87H reset value: see Table 6–2. N78E517A SFR Description and Reset Values

Bit	Name	Description
4	POF	**Power-on reset flag.** This bit will be set as 1 after a power-on reset. It indicates a cold reset, a power-on reset complete. This bit remains its value after any other resets. This flag is recommended to be cleared via software.

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

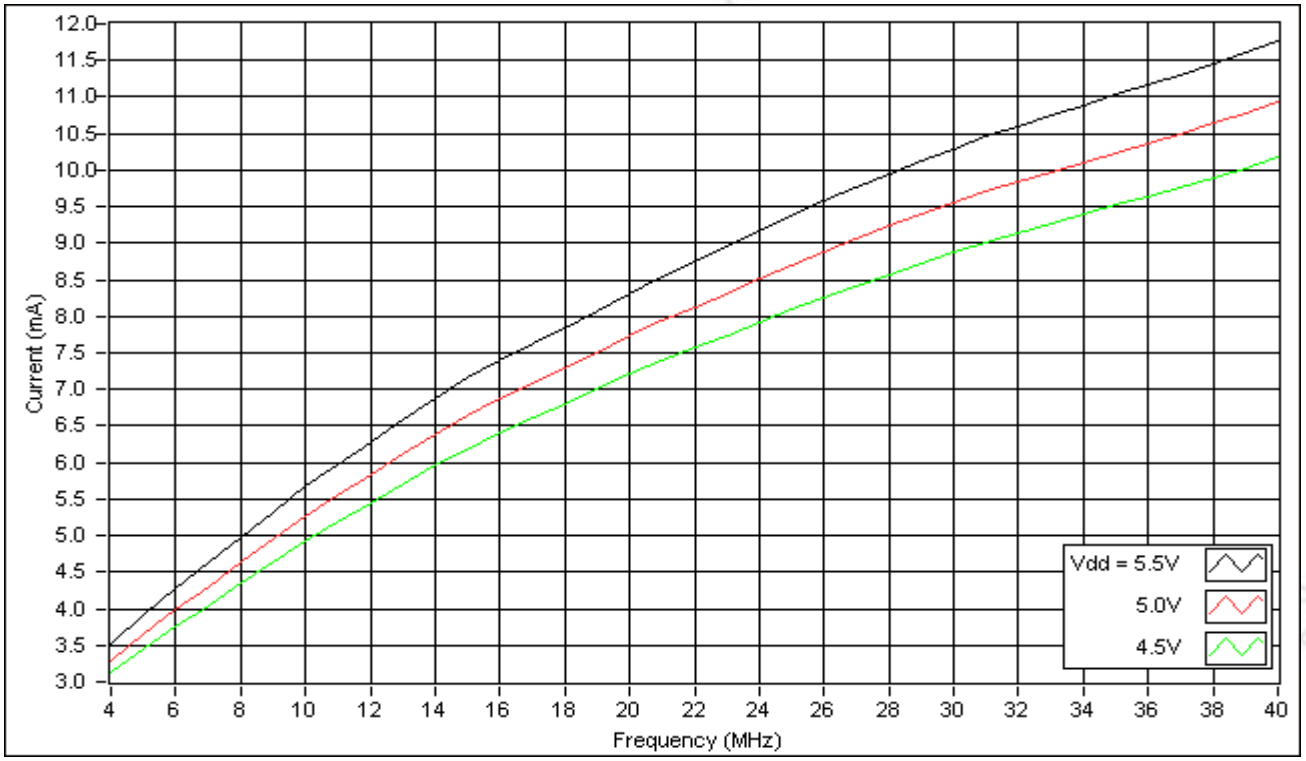

Figure 26–1. Supply Current Under 12T Mode, External Clock (1)

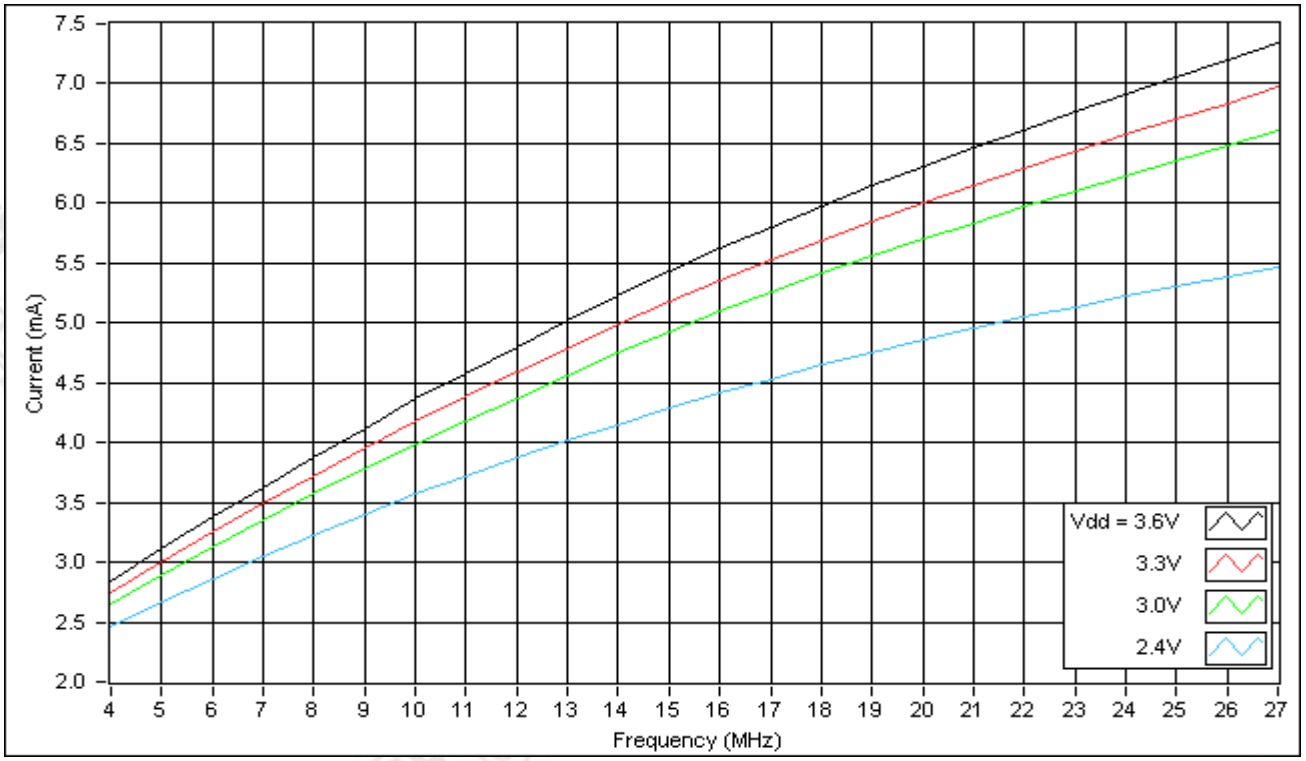

Figure 26–2. Supply Current Under 12T Mode, External Clock (2)

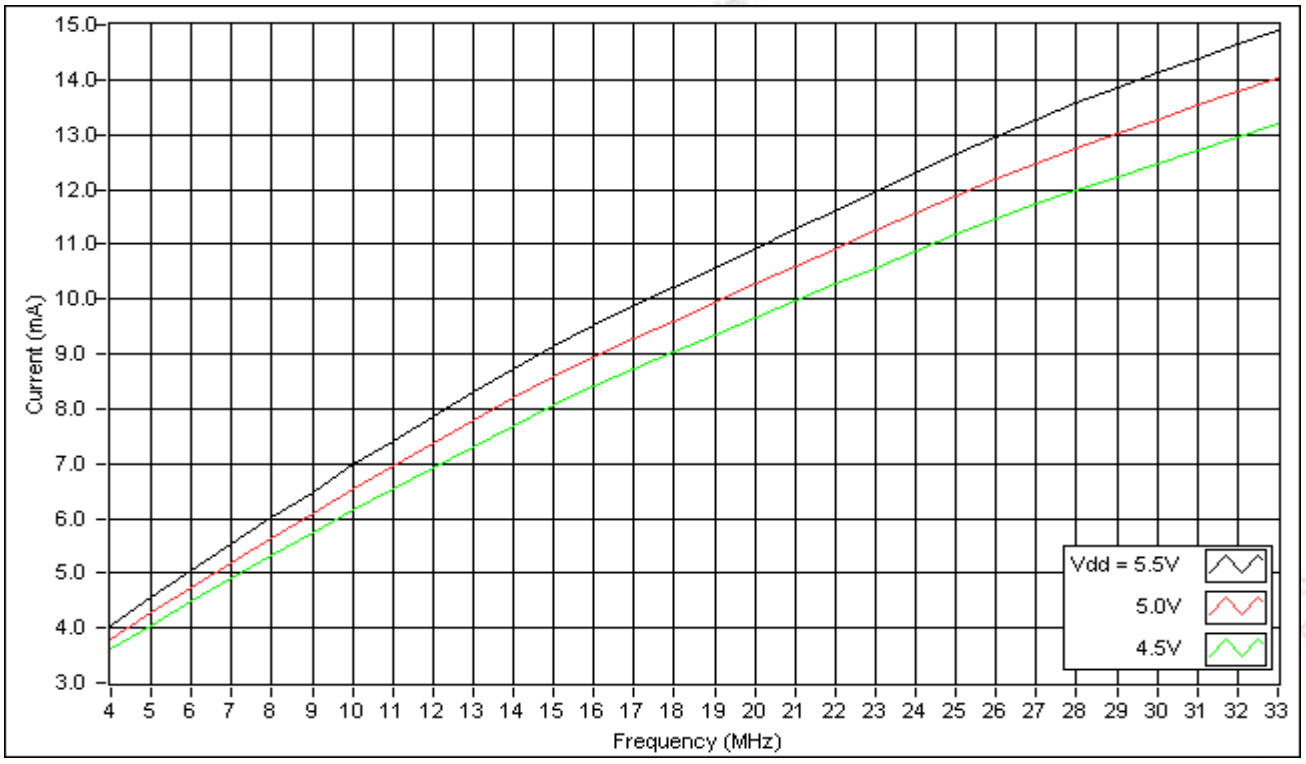

Figure 26–3. Supply Current Under 6T Mode, External Clock (1)

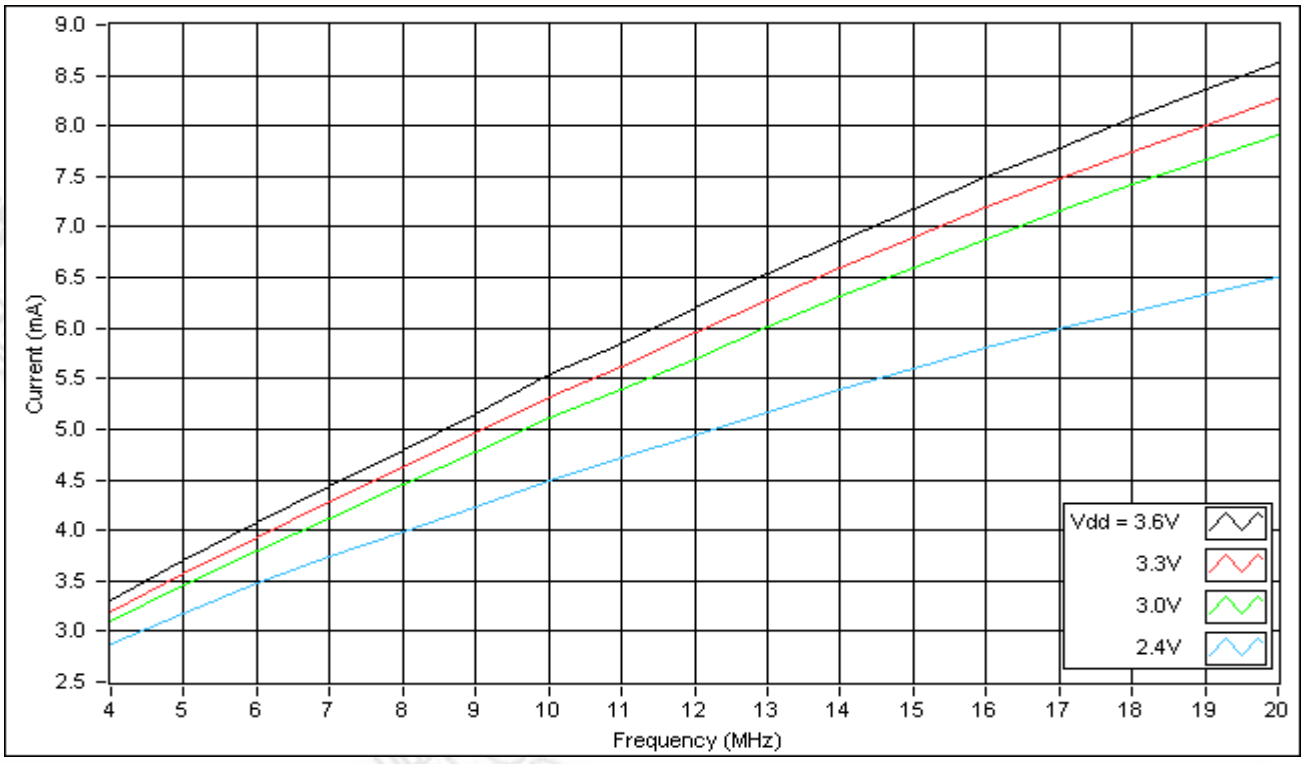

Figure 26–4. Supply Current Under 6T Mode, External Clock (2)