

	ใบเนื้อหา	หน้าที่ 1
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	
ชื่อเรื่อง การใช้งานโมดูล Timer/Counter		
<div>หน่วยที่ 6 การใช้งานโมดูล Timer/Counter</div> <div>การใช้งานโมดูล Timer/Counter</div> <div>1. คุณสมบัติของโมดูล Timer/Counter</div> <div>โมดูล Timer/Counter คือโมดูลพิเศษของไมโครคอนโทรลเลอร์ที่มีคุณสมบัติในการทำหน้าที่นับสัญญาณ โดยแบ่งหน้าที่ออกเป็น 2 ลักษณะ คือ หน้าที่ในการทำงานรูปแบบ Timer จะเป็นการนับสัญญาณนาฬิกาภายในของไมโครคอนโทรลเลอร์ที่เกิดจากคาบเวลาของการประมวลผล หรือที่เรียกว่า คาบเวลาของแมชชีนไซเคิล ส่วนการทำงานอีกรูปแบบหนึ่ง คือ หน้าที่ในการทำงานรูปแบบ Counter จะเป็นการนับสัญญาณพัลส์จากภายนอกไมโครคอนโทรลเลอร์ผ่านขาสัญญาณที่เชื่อมต่อกับโมดูล Timer/Counter นั้น ๆ</div> <div>ความละเอียดในการนับของโมดูล Timer/Counter แต่ละตระกูลจะมีค่าไม่เท่ากัน ขึ้นอยู่กับลักษณะเฉพาะของโมดูล Timer/Counter ตัวนั้น ๆ แต่ สามารถแบ่งความละเอียดในการนับของ Timer/Counter ได้ออกเป็น 2 กลุ่ม คือ กลุ่ม Timer/Counter ที่มีความละเอียดในการนับ 8 บิต และกลุ่ม Timer/Counter ที่มีความละเอียดในการนับ 16 บิต</div> <div>คาบเวลาของแมชชีนไซเคิลของไมโครคอนโทรลเลอร์ตระกูล MCS-51 หาได้จากสมการ $T = 1/(XTAL/12)$ ตระกูล PIC16F หาได้จากสมการ $T = 1/(XTAL/4)$ และตระกูล AVR หาได้จากสมการ $T = 1/XTAL$</div> <div>ลักษณะการนับของโมดูล Timer/Counter คือเมื่อสัญญาณผ่านไป 1 ลูกหรือ 1 แมชชีนไซเคิล โมดูล Timer/Counter จะทำการนับขึ้น 1 ค่า ถ้าเป็นโมดูล Timer/Counter ขนาด 8 บิต จะทำการนับค่า 0 – 255 เมื่อนับเพิ่มขึ้นอีก 1 ค่า ก็จะทำให้ค่าวนกลับมาเป็น 0 ส่วนโมดูล Timer/Counter ขนาด 16 บิต ก็จะสามารถนับค่าได้ตั้งแต่ 0 – 65535 เมื่อนับเพิ่มขึ้นอีก 1 ค่า ก็จะทำให้ค่าวนกลับมาเป็น 0 เช่นเดียวกัน</div> <div>โมดูล Timer/Counter นอกจากจะใช้งานเป็นโมดูลสำหรับนับสัญญาณ ในบางกรณีอาจจะใช้งานร่วมกับโมดูลพิเศษอื่น ๆ เพื่อใช้เป็นแหล่งกำเนิดสัญญาณนาฬิกาของโมดูลพิเศษนั้น ๆ เช่นใช้งานร่วมกับโมดูล UART ,โมดูล PWM เป็นต้น</div> <div>ไมโครคอนโทรลเลอร์เบอร์ AT89C51ED2 จะมีโมดูล Timer/Counter อยู่จำนวน 3 ตัวได้แก่ Timer0 , Timer1 และ Timer2 โดย Timer/Counter ทั้ง 3 ตัวมีความละเอียดในการนับได้สูงสุด 16 บิต</div> <div>ไมโครคอนโทรลเลอร์เบอร์ PIC16F887 จะมีโมดูล Timer/Counter อยู่จำนวน 3 ตัวได้แก่ Timer0 , Timer1 และ Timer2 โดย Timer/Counter 0 และ 2 มีความละเอียดในการนับได้สูงสุด 8 บิต ส่วน Timer/Counter 1 มีความละเอียดในการนับได้สูงสุด 16 บิต</div> <div>ไมโครคอนโทรลเลอร์เบอร์ ATMEGA32 จะมีโมดูล Timer/Counter อยู่จำนวน 3 ตัวได้แก่ Timer0 , Timer1 และ Timer2 โดย Timer/Counter 0 และ 2 มีความละเอียดในการนับได้สูงสุด 8 บิต ส่วน Timer/Counter 1 มีความละเอียดในการนับได้สูงสุด 16 บิต</div>		

หน้าที 2

หน่วยที่ 6

ชื่อหน่วย การใช้งานโมดูล Timer/Counter

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

2. รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter

การใช้งานโมดูล Timer/Counter ของไมโครคอนโทรลเลอร์แต่ละตระกูล จะต้องมีการกำหนดคุณลักษณะของการทำงานต่าง ๆ ผ่านรีจิสเตอร์ที่เกี่ยวข้อง โดยแยกตามเบอร์ของไมโครคอนโทรลเลอร์ดังนี้

2.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ AT89C51ED2

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของไมโครคอนโทรลเลอร์ AT89C51ED2 แยกออกมาตามโมดูล Timer/Counter ได้ดังนี้

2.1.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 0 ของ AT89C51ED2

TCON คือรีจิสเตอร์ที่ควบคุมการทำงานของ Timer0 และ Timer1 โดยจะใช้บิต TR0 ในการปิดเปิดการทำงานของโมดูล Timer0 และบิต TF0 เพื่อตรวจสอบสถานะการนับเกินของโมดูล Timer0 ซึ่งภายในรีจิสเตอร์ TCON ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.1

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
-----	-----	-----	-----	-----	-----	-----	-----

TF1	TCON. 7 Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine.
-----	---

TR1 TCON. 6 Timer 1 run control bit. Set/cleared by software to turn Timer/Counter 1 ON/OFF.

TF0	TCON. 5	Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine.
-----	---------	---

TR0 TCON. 4 Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF.

รูปที่ 1.1 แสดงข้อมูลบิตภายในรีจิสเตอร์ TCON และคำอธิบายการใช้งาน

TMOD คือรีจิสเตอร์ที่ทำหน้าที่ในการกำหนดโหมดการทำงานของ Timer0 และ Timer1 โดยจะแบ่งออกเป็น 2 กลุ่ม คือ 4 บิตบน ได้แก่ บิต 7 ถึง บิต 4 ทำหน้าที่ในการกำหนดโหมดการทำงานของ Timer1 ส่วน 4 บิตล่าง ได้แก่ บิต 3 ถึง บิต 0 ทำหน้าที่ในการกำหนดโหมดการทำงานของ Timer0 ซึ่งภายในรีจิสเตอร์ TMOD ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.2

GATE	C/\bar{T}	M1	M0	GATE	C/\bar{T}	M1	M0
------	-------------	----	----	------	-------------	----	----

TIMER 1

TIMER 0

GATE When TR_x (in TCON) is set and GATE = 1, TIMER/COUNTER_x will run only while INT_x pin is high (hardware control). When GATE = 0, TIMER/COUNTER_x will run only while TR_x = 1 (software control).

C/ \overline{T} Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin).


M1	Mode selector bit. (NOTE 1)
----	-----------------------------

M0	Mode selector bit. (NOTE 1)
----	-----------------------------

NOTE 1:

M1	M0	Operating Mode
0	0	0 13-bit Timer (MCS-48 compatible)
0	1	1 16-bit Timer/Counter
1	0	2 8-bit Auto-Reload Timer/Counter
1	1	3 (Timer 0) TLO is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits, TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.
1	1	3 (Timer 1) Timer/Counter 1 stopped.


รูปที่ 1.2 แสดงข้อมูลบิตภายในรีจิสเตอร์ TMOD และคำอธิบายการใช้งาน

	ใบเนื้อหา		หน้าที่ 3
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter			
TH0 และ TL0 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 0 โดย TH0 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TL0 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง			
2.1.2 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 1 ของ AT89C51ED2			
TCON และ TMOD เป็นรีจิสเตอร์ที่มีหน้าที่ควบคุม และกำหนดโหมดการทำงานของ Timer1 ซึ่งมีรายละเอียดหน้าที่การทำงานของรีจิสเตอร์เช่นเดียวกับหัวข้อ 2.1.1			
TH1 และ TL1 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 0 โดย TH0 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TL0 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง			
2.1.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 2 ของ AT89C51ED2			
T2CON คือรีจิสเตอร์ที่ทำหน้าที่ในการควบคุม และเลือกโหมดการทำงานของ Timer/Counter 2 โดยบิต TF2 แสดงสถานการณ์นับเกินของโมดูล Timer2 ,บิต TR2 ทำหน้าที่ในการปิดเปิดการทำงานของโมดูล Timer2 และบิต C/T ทำหน้าที่ในการเลือกโหมดการทำงานของ Timer2 ซึ่งภายในรีจิสเตอร์ T2CON ประกอบด้วย บิตอะไรบ้างแสดงดังรูปที่ 1.3			

TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
TF2	T2CON. 7	Timer 2 overflow flag set by hardware and cleared by software. TF2 cannot be set when either RCLK = 1 or CLK = 1					
EXF2	T2CON. 6	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX, and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software.					
RCLK	T2CON. 5	Receive clock flag. When set, causes the Serial Port to use Timer 2 overflow pulses for its receive clock in modes 1 & 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.					
TCLK	T2CON. 4	Transmit clock flag. When set, causes the Serial Port to use Timer 2 overflow pulses for its transmit clock in modes 1 & 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.					
EXEN2	T2CON. 3	Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of negative transition on T2EX if Timer 2 is not being used to clock the Serial Port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.					
TR2	T2CON. 2	Software START/STOP control for Timer 2. A logic 1 starts the Timer.					
C/T2	T2CON. 1	Timer or Counter select. 0 = Internal Timer. 1 = External Event Counter (falling edge triggered).					
CP/RL2	T2CON. 0	Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, Auto-Reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the Timer is forced to Auto-Reload on Timer 2 overflow.					

รูปที่ 1.3 แสดงข้อมูลบิตภายในรีจิสเตอร์ T2CON และคำอธิบายการใช้งาน			
TH2 และ TL2 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 2 โดย TH2 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TL2 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง			

	ใบเนื้อหา		หน้าที่ 4
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

2.2 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ PIC16F887

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของไมโครคอนโทรลเลอร์ PIC16F887 แยกออกตามโมดูล Timer/Counter ได้ดังนี้

2.2.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 0 ของ PIC16F887

OPTION_REG คือรีจิสเตอร์ที่มีหน้าที่ในการกำหนดคุณสมบัติของโมดูลพิเศษภายในของไมโครคอนโทรลเลอร์ PIC16F887 และมีบางบิตที่ใช้ในการกำหนดคุณสมบัติของโมดูล Timer0 ได้แก่บิต T0CS , T0SE , PSA และ PS2:PS0 ดังแสดงในรูปที่ 1.4

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7		bit 0					

bit 5

T0CS: TMR0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (Fosc/4)

bit 4

T0SE: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3

PSA: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0


PS<2:0>: Prescaler Rate Select bits

BIT VALUE	TMR0 RATE	WDT RATE
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

รูปที่ 1.4 แสดงข้อมูลบิตภายในรีจิสเตอร์ OPTION_REG และคำอธิบายการใช้งานที่เกี่ยวข้องกับ Timer0

TMR0 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 0 มีความละเอียดในการนับขนาด 8 บิต

INTCON คือรีจิสเตอร์ที่ทำหน้าที่ควบคุมการอินเทอร์รัพต์ของไมโครคอนโทรลเลอร์ PIC16F887 และมีส่วนที่เกี่ยวข้องกับการทำงานของ Timer0 คือบิต TOIF ซึ่งมีหน้าที่แสดงสถานะการนับเกินของโมดูล Timer0 และบิต TOIE ซึ่งเป็นที่ยอมให้เกิดการอินเทอร์รัพต์จาก Timer0 ส่วนข้อมูลของบิตต่าง ๆ ภายในรีจิสเตอร์ INTCON แสดงดังรูปที่ 1.5

	ใบเนื้อหา						หน้าที่ 5
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004						หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter						

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	TOIE	INTE	RBIE ^(1,3)	TOIF ⁽²⁾	INTF	RBIF
bit 7							bit 0

รูปที่ 1.5 แสดงข้อมูลบิตภายในรีจิสเตอร์ INTCON

2.2.2 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 1 ของ PIC16F887

TMR1H และ TMR1L คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 1 โดย TMR1H ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TMR1L ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง

T1CON คือรีจิสเตอร์ที่ควบคุมการทำงานของโมดูล Timer/Counter 1 โดยมีบิตที่สำคัญได้แก่ บิต TMR1ON คือบิตที่ควบคุมการเปิดปิดการทำงานของโมดูล Timer1 , บิต TMR1CS คือบิตที่ทำหน้าที่เลือกแหล่งกำเนิดสัญญาณนาฬิกาเพื่อการนับ , บิต T1CKPS1:T1CKPS0 คือบิตที่ทำหน้าที่ในการกำหนดค่าปรีสเกลเลอร์ของโมดูล Timer1 ซึ่งภายในรีจิสเตอร์ T1CON ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.6

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T1GINV ⁽¹⁾	TMR1GE ⁽²⁾	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON
bit 7							bit 0

bit 7

T1GINV: Timer1 Gate Invert bit⁽¹⁾
1 = Timer1 gate is active-high (Timer1 counts when gate is high)
0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 6

TMR1GE: Timer1 Gate Enable bit⁽²⁾
If TMR1ON = 0:
This bit is ignored
If TMR1ON = 1:
1 = Timer1 is on if Timer1 gate is not active
0 = Timer1 is on

bit 5-4

T1CKPS<1:0>: Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale Value
10 = 1:4 Prescale Value
01 = 1:2 Prescale Value
00 = 1:1 Prescale Value

bit 3

T1OSCEN: LP Oscillator Enable Control bit
1 = LP oscillator is enabled for Timer1 clock
0 = LP oscillator is off

bit 2

T1SYNC: Timer1 External Clock Input Synchronization Control bit
TMR1CS = 1:
1 = Do not synchronize external clock input
0 = Synchronize external clock input
TMR1CS = 0:
This bit is ignored. Timer1 uses the internal clock


bit 1


TMR1CS: Timer1 Clock Source Select bit
1 = External clock from T1CKI pin (on the rising edge)
0 = Internal clock (Fosc/4)

bit 0

TMR1ON: Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

รูปที่ 1.6 แสดงข้อมูลบิตภายในรีจิสเตอร์ T1CON และคำอธิบายการใช้งานที่เกี่ยวข้องกับ Timer1

	ใบเนื้อหา		หน้าที่ 6																								
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6																								
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter																										
ชื่อเรื่อง การใช้งานโมดูล Timer/Counter																											
<p>PIR1 คือรีจิสเตอร์ที่ทำหน้าที่แสดงสถานะการทำงานของโมดูลพิเศษภายในไมโครคอนโทรลเลอร์ PIC16F887 ซึ่งมีบิตที่เกี่ยวข้องกับการทำงานของ Timer1 คือบิต TMR1IF คือเมื่อ Timer1 เกิดเหตุการณ์การนับเกินจะทำบิตนี้มีค่าเป็นลอจิก ‘1’</p> <p>2.2.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 2 ของ PIC16F887</p> <p>TMR2 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 2 มีความละเอียดในการนับขนาด 8 บิต</p> <p>T2CON คือรีจิสเตอร์ที่ควบคุมการทำงานของโมดูล Timer/Counter 2 โดยมีบิตที่สำคัญได้แก่ บิต TMR2ON คือบิตที่ควบคุมการเปิดปิดการทำงานของโมดูล Timer1 , บิต T2CKPS1:T2CKPS0 คือบิตที่ทำหน้าที่ในการกำหนดค่าปรีสเกลเลอร์ของโมดูล Timer2 ที่มีแหล่งกำเนิดสัญญาณมาจากค่าแมกซ์ซิมัซเกิล , บิต TOUTPS3:TOUTPS0 คือบิตที่ทำหน้าที่ในการกำหนดค่าโพสสเกลเลอร์ของโมดูล Timer2 ซึ่งภายในรีจิสเตอร์ T2CON ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.7</p> <table border="1"><thead><tr><th>U-0</th><th>R/W-0</th><th>R/W-0</th><th>R/W-0</th><th>R/W-0</th><th>R/W-0</th><th>R/W-0</th><th>R/W-0</th></tr></thead><tbody><tr><td>—</td><td>TOUTPS3</td><td>TOUTPS2</td><td>TOUTPS1</td><td>TOUTPS0</td><td>TMR2ON</td><td>T2CKPS1</td><td>T2CKPS0</td></tr><tr><td>bit 7</td><td></td><td></td><td></td><td></td><td></td><td></td><td>bit 0</td></tr></tbody></table> <p>bit 7 Unimplemented: Read as ‘0’</p> <p>bit 6-3 TOUTPS<3:0>: Timer2 Output Postscaler Select bits</p> <p>0000 = 1:1 Postscaler 0001 = 1:2 Postscaler 0010 = 1:3 Postscaler 0011 = 1:4 Postscaler 0100 = 1:5 Postscaler 0101 = 1:6 Postscaler 0110 = 1:7 Postscaler 0111 = 1:8 Postscaler 1000 = 1:9 Postscaler 1001 = 1:10 Postscaler 1010 = 1:11 Postscaler 1011 = 1:12 Postscaler 1100 = 1:13 Postscaler 1101 = 1:14 Postscaler 1110 = 1:15 Postscaler 1111 = 1:16 Postscaler</p> <p>bit 2 TMR2ON: Timer2 On bit</p> <p>1 = Timer2 is on 0 = Timer2 is off</p> <p>bit 1-0 T2CKPS<1:0>: Timer2 Clock Prescale Select bits</p> <p>00 = Prescaler is 1 01 = Prescaler is 4 1x = Prescaler is 16</p> <p>รูปที่ 1.7 แสดงข้อมูลบิตภายในรีจิสเตอร์ T2CON และคำอธิบายการใช้งานที่เกี่ยวข้องกับ Timer2</p>				U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	bit 7							bit 0
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0																				
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0																				
bit 7							bit 0																				

	ใบเนื้อหา		หน้าที่ 8
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

2.2.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 1 ของ ATMEGA32

TCNT1H และ TCNT1L คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 1 โดย TCNT1H ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TCNT1L ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง

TCCR1B คือรีจิสเตอร์ที่มีหน้าที่ในการกำหนดโหมดในการทำงานของโมดูล Timer1 เลือกแหล่งกำเนิดสัญญาณนาฬิกาเป็นต้น โดยรีจิสเตอร์นี้จะมีบิตที่เกี่ยวข้องกับการทำงานของ Timer1 คือบิต ICNC1 ทำหน้าที่ในการกำหนดการกรองสัญญาณรบกวนในขณะทำงานในลักษณะแค็ปเจอร์สัญญาณอินพุต , บิต ICES1 ทำหน้าที่ในการเลือกขอบขาสัญญาณที่ต้องการจะทำการแค็ปเจอร์ และบิต CS12:CS10 ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ให้แก่ Timer1 ซึ่งภายในรีจิสเตอร์ TCCR1B ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.10

7	6	5	4	3	2	1	0
ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

- **Bit 7 – ICNC1: Input Capture Noise Canceler**
Setting this bit (to one) activates the Input Capture Noise Canceler. When the Noise Canceler is activated, the input from the Input Capture Pin (ICP1) is filtered. The filter function requires four successive equal valued samples of the ICP1 pin for changing its output. The Input Capture is therefore delayed by four Oscillator cycles when the Noise Canceler is enabled.
- **Bit 6 – ICES1: Input Capture Edge Select**
This bit selects which edge on the Input Capture Pin (ICP1) that is used to trigger a capture event. When the ICES1 bit is written to zero, a falling (negative) edge is used as trigger, and when the ICES1 bit is written to one, a rising (positive) edge will trigger the capture.

CS12	CS11	CS10	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk _{IO} /1 (No prescaling)
0	1	0	clk _{IO} /8 (From prescaler)
0	1	1	clk _{IO} /64 (From prescaler)
1	0	0	clk _{IO} /256 (From prescaler)

CS12	CS11	CS10	Description
1	0	1	clk _{IO} /1024 (From prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.

รูปที่ 1.10 แสดงข้อมูลบิตภายในรีจิสเตอร์ TCCR1B และคำอธิบายการใช้งานที่เกี่ยวข้องกับ Timer1

TIFR คือรีจิสเตอร์ที่ทำหน้าที่แสดงสถานะการทำงานของโมดูล Timer/Counter ภายในตัวไมโครคอนโทรลเลอร์ ATMEGA32 ซึ่งมีบิตที่เกี่ยวข้องกับการทำงานของ Timer1 คือบิต TOV1 จะทำหน้าที่แสดงสถานะการนับเกินของโมดูล Timer1 ซึ่งภายในรีจิสเตอร์ TIFR ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.9

	ใบเนื้อหา		หน้าที่ 9
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

2.2.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 2 ของ ATMEGA32

TCCR2 คือรีจิสเตอร์ขนาด 8 บิตที่ทำหน้าที่ในการกำหนดโหมดการทำงานของโมดูล Timer2 ให้ทำงานในลักษณะต่าง ๆ และกำหนดค่า Prescale ของ Timer2 โดยบิตที่เกี่ยวข้องกับการกำหนดให้ Timer2 ทำงานในรูปแบบ Timer/Counter คือบิต CS22:CS20 ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ให้แก่ Timer2 ซึ่งภายในรีจิสเตอร์ TCCR2 ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.11

7	6	5	4	3	2	1	0
FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20
W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CS22	CS21	CS20	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk _{T2S} /(No prescaling)
0	1	0	clk _{T2S} /8 (From prescaler)
0	1	1	clk _{T2S} /32 (From prescaler)
1	0	0	clk _{T2S} /64 (From prescaler)
1	0	1	clk _{T2S} /128 (From prescaler)
1	1	0	clk _{T2S} /256 (From prescaler)
1	1	1	clk _{T2S} /1024 (From prescaler)

รูปที่ 1.11 แสดงข้อมูลบิตภายในรีจิสเตอร์ TCCR2 และคำอธิบายการใช้งานที่เกี่ยวข้องกับ Timer2

TCNT2 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 2 มีความละเอียดในการนับขนาด 8 บิต


TIFR คือรีจิสเตอร์ที่ทำหน้าที่แสดงสถานะการทำงานของโมดูล Timer/Counter ภายในตัวไมโครคอนโทรลเลอร์ ATMEGA32 ซึ่งมีบิตที่เกี่ยวข้องกับการทำงานของ Timer2 คือบิต TOV2 จะทำหน้าที่แสดงสถานะการนับเกินของโมดูล Timer2 ซึ่งภายในรีจิสเตอร์ TIFR ประกอบด้วยบิตอะไรบ้างแสดงดังรูปที่ 1.9

3. การทำงานของโมดูล Timer/Counter ในลักษณะ Counter

การทำงานของโมดูล Timer/Counter ในลักษณะ Counter คือการกำหนดคุณสมบัติของรีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter นั้น ๆ ให้ทำงานโดยใช้สัญญาณอินพุตจากแหล่งภายนอกที่ตรงกับโมดูลนั้น

3.1 การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ AT89C51ED2

ไมโครคอนโทรลเลอร์เบอร์ AT89C51ED2 ของตระกูล MCS-51 มีโมดูล Timer/Counter อยู่ภายในจำนวน 3 ตัว โดยแต่ละตัวทำงานแยกขาดออกจากกัน และมีความละเอียดในการนับได้สูงสุด 16 บิตทั้ง 3 ตัว และยังสามารถกำหนดให้ทำงานในลักษณะของ Counter ได้ทั้ง 3 เช่นเดียวกัน ดังนี้

	ใบเนื้อหา		หน้าที่ 10
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

3.1.1 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 0 ใน AT89C51ED2

โมดูล Timer/Counter 0 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 4 รูปแบบแสดงดังรูปที่ 1.12 ด้วยการกำหนดที่รีจิสเตอร์ TMOD ส่วนสัญญาณนาฬิกาภายนอกจะถูกต่อที่ขา P3.4

MODE	COUNTER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	04H	0CH
1	16-bit Timer	05H	0DH
2	8-bit Auto-Reload	06H	0EH
3	one 8-bit Counter	07H	0FH

NOTES:

1. The Timer is turned ON/OFF by setting/clearing bit TR0 in the software.

2. The Timer is turned ON/OFF by the 1 to 0 transition on INT0 (P3.2) when TR0 = 1 (hardware control).

รูปที่ 1.12 แสดงตารางการกำหนดรูปแบบการทำงานในโหมด Counter ของ Timer0

1. จากรูปที่ 1.12 โหมด 0 คือการนับสัญญาณนาฬิกาภายนอกขนาด 13 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับ 5 บิตล่าง ส่วน TH0 จะทำหน้าที่นับ 8 บิตบน หรืออาจกล่าวได้ว่าการนับแบบนี้เป็นการนับแบบ MOD32

2. จากรูปที่ 1.12 โหมด 1 คือการนับสัญญาณนาฬิกาภายนอกขนาด 16 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับ 8 บิตล่าง ส่วน TH0 จะทำหน้าที่นับ 8 บิตบน


3. จากรูปที่ 1.12 โหมด 2 คือการนับสัญญาณนาฬิกาภายนอกขนาด 8 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับค่า 8 บิตที่ถูกกำหนดโดยรีจิสเตอร์ TH0 และเมื่อ TLO นับจนเกิดโอเวอร์โฟลว์ หรือ TLO วนมานับค่า 0 จะทำให้เกิดเหตุการณ์ที่เรียกว่าออดิตรีโหลด คือ TLO จะเริ่มต้นนับค่าใหม่จากค่าเริ่มต้นที่ TH0

4. จากรูปที่ 1.12 โหมด 3 คือการนับสัญญาณนาฬิกาภายนอกขนาด 8 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับค่า 8 บิต

ส่วนการควบคุมการนับสามารถกระทำได้ 2 รูปแบบคือ กระทำด้วยรูปแบบกระบวนการทางฮาร์ดแวร์ หรือ กระทำด้วยรูปแบบกระบวนการทางซอฟต์แวร์ ซึ่งกำหนดได้ด้วยบิต GATE ของรีจิสเตอร์ TMOD ถ้า บิต GATE = ‘0’ เป็นการนับที่ถูกควบคุมด้วยกระบวนการทางซอฟต์แวร์ เป็นต้น

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ AT89C51ED2 ทำงานในลักษณะ Counter โหมด 3 คือนับสัญญาณนาฬิกาที่ขา P3.4 และสามารถนับค่าได้สูงสุด 8 บิต

```
void init_timer0_counter8bit_mode3(){
    TMOD = (TMOD & 0xf0) | 0x07;
    TF0 = 0;
    TLO = 0;
    TR0 = 1;
}
```

	ใบเนื้อหา		หน้าที่ 11
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

3.1.2 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 1 ใน AT89C51ED2

โมดูล Timer/Counter 1 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 3 รูปแบบแสดงดังรูปที่ 1.13 ด้วยการกำหนดที่รีจิสเตอร์ TMOD ส่วนสัญญาณนาฬิกาภายนอกจะถูกต่อที่ขา P3.5

MODE	COUNTER 1 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	40H	C0H
1	16-bit Timer	50H	D0H
2	8-bit Auto-Reload	60H	E0H
3	not available	—	—

NOTES:

1. The Timer is turned ON/OFF by setting/clearing bit TR1 in the software.

2. The Timer is turned ON/OFF by the 1 to 0 transition on INT1 (P3.3) when TR1 = 1 (hardware control).

รูปที่ 1.13 แสดงตารางการกำหนดรูปแบบการทำงานในโหมด Counter ของ Timer1

1. จากรูปที่ 1.13 โหมด 0 คือการนับสัญญาณนาฬิกาภายนอกขนาด 13 บิต โดยรีจิสเตอร์ TL1 จะทำหน้าที่นับ 5 บิตล่าง ส่วน TH1 จะทำหน้าที่นับ 8 บิตบน หรืออาจกล่าวได้ว่าการนับแบบนี้เป็นการนับแบบ MOD32


2. จากรูปที่ 1.13 โหมด 1 คือการนับสัญญาณนาฬิกาภายนอกขนาด 16 บิต โดยรีจิสเตอร์ TL1 จะทำหน้าที่นับ 8 บิตล่าง ส่วน TH1 จะทำหน้าที่นับ 8 บิตบน

3. จากรูปที่ 1.13 โหมด 2 คือการนับสัญญาณนาฬิกาภายนอกขนาด 8 บิต โดยรีจิสเตอร์ TL1 จะทำหน้าที่นับค่า 8 บิตที่ถูกกำหนดโดยรีจิสเตอร์ TH1 และเมื่อ TL1 นับจนเกิดโอเวอร์โฟลล์ หรือ TL1 วนมานับค่า 0 จะทำให้เกิดเหตุการณ์ที่เรียกว่าอู่อัฒติโรลด์ คือ TL1 จะเริ่มต้นนับค่าใหม่จากค่าเริ่มต้นที่ TH1

ส่วนการควบคุมการนับสามารถกระทำได้ 2 รูปแบบคือ กระทำด้วยรูปแบบกระบวนการทางฮาร์ดแวร์ หรือ กระทำด้วยรูปแบบกระบวนการทางซอฟต์แวร์ ซึ่งกำหนดได้ด้วยบิต GATE ของรีจิสเตอร์ TMOD ถ้าบิต GATE = ‘0’ เป็นการนับที่ถูกควบคุมด้วยกระบวนการทางซอฟต์แวร์ เป็นต้น

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ AT89C51ED2 ทำงานในลักษณะ Counter โหมด 1 คือนับสัญญาณนาฬิกาที่ขา P3.5 และสามารถนับค่าได้สูงสุด 16 บิต

```
void init_timer1_counter16bit_mode1(){
    TMOD = (TMOD & 0x0f) | 0x50;
    TF1 = 0;
    TH1 = 0;
    TL1 = 0;
    TR1 = 1;
}
```

	ใบเนื้อหา	หน้าที่ 12
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

3.1.3 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 2 ใน AT89C51ED2

โมดูล Timer/Counter 2 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 2 รูปแบบแสดงดังรูปที่ 1.14 ด้วยการกำหนดที่รีจิสเตอร์ T2CON ส่วนสัญญาณนาฬิกาภายนอกจะถูกต่อที่ขา P1.0

MODE	TMOD	
	INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
16-bit Auto-Reload	02H	0AH
16-bit Capture	03H	0BH

NOTES:

1. Capture/Reload occurs only on Timer/Counter overflow.

2. Capture/Reload occurs on Timer/Counter overflow and a 1 to 0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generating mode.

รูปที่ 1.14 แสดงตารางการกำหนดรูปแบบการทำงานในโหมด Counter ของ Timer2


1. จากรูปที่ 1.14 โหมด 0 คือการนับสัญญาณนาฬิกาภายนอกขนาด 16 บิต โดยรีจิสเตอร์ TL2 จะทำหน้าที่นับค่า 8 บิตล่าง ส่วน TH2 จะทำหน้าที่นับค่า 8 บิตบน ที่ถูกกำหนดโดยรีจิสเตอร์ RCAP2H และ RCAP2L เมื่อ TL2 และ TH2 นับจนเกิดโอเวอร์โฟลว์ หรือ TL2 และ TH2 วนมานับค่า 0 จะทำให้เกิดเหตุการณ์ที่เรียกว่า ออโต้รีโหลด คือ TL2 และ TH2 จะเริ่มต้นนับค่าใหม่จากค่าเริ่มต้นที่ RCAP2H และ RCAP2L

2. จากรูปที่ 1.14 โหมด 1 คือการสุ่มนับสัญญาณนาฬิกาภายนอกขนาด 16 บิต โดยรีจิสเตอร์ TL2 จะทำหน้าที่นับ 8 บิตล่าง ส่วน TH2 จะทำหน้าที่นับ 8 บิตบน ที่เรียกว่าการ Capture สัญญาณ เพื่อคำนวณกลับเป็นค่าความถี่ของสัญญาณ

ส่วนการควบคุมการนับสามารถกระทำได้ 2 รูปแบบคือ กระทำด้วยรูปแบบกระบวนการทางฮาร์ดแวร์ หรือ กระทำด้วยรูปแบบกระบวนการทางซอฟต์แวร์ ซึ่งกำหนดได้ด้วยบิต EXEN2 ของรีจิสเตอร์ T2CON ถ้าบิต EXEN2 = ‘0’ เป็นการนับที่ถูกควบคุมด้วยกระบวนการทางซอฟต์แวร์ เป็นต้น

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer2 ของ AT89C51ED2 ทำงานในลักษณะ Counter โหมด 0 คือนับสัญญาณนาฬิกาที่ขา P1.0 และสามารถนับค่าได้สูงสุด 16 บิตแบบ Auto Reload

```
void init_timer2_counter16bit_mode0(){
    T2CON |= 0x02;
    RCAP2H = 0xff;
    RCAP2L = 0x9c;
    TF2 = 0;
    TH2 = RCAP2H;
    TL2 = RCAP2L;
    TR2 = 1;
}
```

	ใบเนื้อหา	หน้าที่ 13
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

3.2 การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ PIC16F887

ไมโครคอนโทรลเลอร์เบอร์ PIC16F887 ของตระกูล PIC16F มีโมดูล Timer/Counter อยู่ภายในจำนวน 3 ตัว โดยแต่ละตัวทำงานแยกขาดออกจากกัน และมีความละเอียดในการนับได้สูงสุด 16 บิตเพียง 1 ตัว และยังสามารถกำหนดให้ทำงานในลักษณะของ Counter ได้ 2 โมดูลคือ Timer0 และ Timer1 ดังนี้

3.2.1 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 0 ใน PIC16F887

โมดูล Timer/Counter 0 ใน PIC16F887 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้โดยกำหนดที่บิต TOCS ให้มีค่าเป็นลอจิก ‘1’ เพื่อทำการนับค่าสัญญาณที่เข้ามาที่ขา T0CKI (RA4) ส่วนจะนับช่วงขอบขาขึ้น หรือขอบขาลงจะกำหนดได้จากบิต TOSE ของรีจิสเตอร์ OPTION_REG ดังรายละเอียดในรูปที่ 1.4

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ PIC16F887 ทำงานในลักษณะ Counter เพื่อนับสัญญาณนาฬิกาที่ขา T0CKI และสามารถนับค่าได้สูงสุด 8 บิต


```
void init_timer0_counter8bit(){
    TRISAbits.TRISA4 = 1;
    OPTION_REGbits.TOCS = 1;
    OPTION_REGbits.TOSE = 1;
    INTCONbits.T0IF = 0;
    TMR0 = 0;
}
```

3.2.2 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 1 ใน PIC16F887

โมดูล Timer/Counter 1 ใน PIC16F887 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้โดยกำหนดที่บิต TMR1CS ให้มีค่าเป็นลอจิก ‘1’ เพื่อทำการนับค่าสัญญาณที่เข้ามาที่ขา T1CKI (RC0) ส่วนการปิดเปิดการทำงานของโมดูลจะกำหนดที่บิต TMR1ON ของรีจิสเตอร์ T1CON ดังรายละเอียดในรูปที่ 1.6

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ PIC16F887 ทำงานในลักษณะ Counter เพื่อนับสัญญาณนาฬิกาที่ขา T1CKI และสามารถนับค่าได้สูงสุด 16 บิต

```
void init_timer1_counter16bit(){
    TRISCbits.TRISC0 = 1;
    T1CONbits.TMR1CS = 1;
    PIR1bits.TMR1IF = 0;
    TMR1 = 0;
    T1CONbits.TMR1ON = 1;
}
```


	ใบเนื้อหา		หน้าที่ 14
	ชื่อวิชา	ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย		การใช้งานโมดูล Timer/Counter

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

3.3 การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ ATMEGA32

ไมโครคอนโทรลเลอร์เบอร์ ATMEGA32 ของตระกูล AVR มีโมดูล Timer/Counter อยู่ภายในจำนวน 3 ตัว โดยแต่ละตัวทำงานแยกขาดออกจากกัน และมีความละเอียดในการนับได้สูงสุด 16 บิตเพียง 1 ตัว และยังสามารถกำหนดให้ทำงานในลักษณะของ Counter ได้ 2 โมดูลคือ Timer0 และ Timer1 ดังนี้

3.3.1 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 0 ใน ATMEGA32

โมดูล Timer/Counter 0 ใน ATMEGA32 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้โดยกำหนดที่บิต CS02:CS00 เพื่อทำการนับค่าสัญญาณที่เข้ามาที่ขา T0 (PB0) ส่วนจะนับช่วงขอบขาขึ้น หรือขอบขาลง จะกำหนดได้จากบิต CS02:CS00 ของรีจิสเตอร์ TCCR0 ดังรายละเอียดในรูปที่ 1.8

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ ATMEGA32 ทำงานในลักษณะ Counter เพื่อนับสัญญาณนาฬิกาที่ขา T0 และสามารถนับค่าได้สูงสุด 8 บิต


```
void init_timer0_counter8bit(){
    DDRB &= ~(PB0<<1);
    TCCR0 |= (1<<CS02) | (1<<CS01);
    TCNT0 = 0;
}
```

3.3.2 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 1 ใน ATMEGA32

โมดูล Timer/Counter 1 ใน ATMEGA32 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้โดยกำหนดที่บิต CS12:CS10 เพื่อทำการนับค่าสัญญาณที่เข้ามาที่ขา T1 (PB1) ส่วนจะนับช่วงขอบขาขึ้น หรือขอบขาลง จะกำหนดได้จากบิต CS12:CS10 ของรีจิสเตอร์ TCCR1B ดังรายละเอียดในรูปที่ 1.10

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ ATMEGA32 ทำงานในลักษณะ Counter เพื่อนับสัญญาณนาฬิกาที่ขา T1 และสามารถนับค่าได้สูงสุด 16 บิต

```
void init_timer1_counter16bit(){
    DDRB &= ~(PB1<<1);
    TCCR1B |= (1<<CS12) | (1<<CS11);
    TCNT1 = 0;
}
```

	ใบเนื้อหา		หน้าที่ 15
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

4. การทำงานของโมดูล Timer/Counter ในลักษณะ Timer

การทำงานของโมดูล Timer/Counter ในลักษณะ Timer คือการกำหนดคุณสมบัติของรีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter นั้น ๆ ให้ทำงานโดยใช้สัญญาณอินพุตจากแหล่งกำเนิดภายใน ซึ่งก็คือคาบเวลาของการประมวลผลของ CPU ภายในไมโครคอนโทรลเลอร์ หรือที่เรียกว่าคาบเวลาแมชชีนไซเคิล โดยคาบเวลานี้จะขึ้นอยู่กับชนิดของไมโครคอนโทรลเลอร์ และค่า XTAL ที่ต่ออยู่ภายนอกด้วย

4.1 การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ AT89C51ED2

โดยทั่วไปคาบเวลาของแมชชีนไซเคิลของไมโครคอนโทรลเลอร์ตระกูล MCS-51 หาได้จากสมการ $T = 1/(XTAL/12)$ เพื่อใช้ในการคำนวณหาเวลาของการนับหนึ่งค่าจะใช้เวลากี่วินาที โดยไมโครคอนโทรลเลอร์ AT89C51ED2 มีโมดูล Timer/Counter อยู่ภายในทั้งหมด 3 ตัว และแต่ละตัวสามารถกำหนดให้ทำงานในรูปแบบ Timer ได้ทั้งหมดและมีความละเอียดของการนับที่สามารถกำหนดได้ และสามารถนับได้ละเอียดสูงสุด 16 บิต โดย Timer ต่าง ๆ สามารถกำหนดให้ทำงานในลักษณะ Timer ด้วยการกำหนดรีจิสเตอร์ที่เกี่ยวข้องดังนี้

4.1.1 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 0 ใน AT89C51ED2

โมดูล Timer/Counter 0 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้ 4 รูปแบบแสดงดังรูปที่ 1.15 ด้วยการกำหนดที่รีจิสเตอร์ TMOD ส่วนเวลาของการนับหนึ่งค่าจะใช้เวลาเท่าไรสามารถหาได้จากสมการ $T = 1/(XTAL/12)$

MODE	TIMER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	00H	08H
1	16-bit Timer	01H	09H
2	8-bit Auto-Reload	02H	0AH
3	two 8-bit Timers	03H	0BH


NOTES:


1. The Timer is turned ON/OFF by setting/clearing bit TR0 in the software.

2. The Timer is turned ON/OFF by the 1 to 0 transition on $\overline{INT0}$ (P3.2) when TR0 = 1 (hardware control).

รูปที่ 1.15 แสดงตารางการกำหนดรูปแบบการทำงานในโหมด Timer ของ Timer0

1. จากรูปที่ 1.15 โหมด 0 คือการนับสัญญาณนาฬิกาภายในขนาด 13 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับ 5 บิตล่าง ส่วน TH0 จะทำหน้าที่นับ 8 บิตบน หรืออาจกล่าวได้ว่าการนับแบบนี้เป็นการนับแบบ MOD32
2. จากรูปที่ 1.15 โหมด 1 คือการนับสัญญาณนาฬิกาภายในขนาด 16 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับ 8 บิตล่าง ส่วน TH0 จะทำหน้าที่นับ 8 บิตบน
3. จากรูปที่ 1.15 โหมด 2 คือการนับสัญญาณนาฬิกาภายในขนาด 8 บิต โดยรีจิสเตอร์ TLO จะทำหน้าที่นับค่า 8 บิตที่ถูกกำหนดโดยรีจิสเตอร์ TH0 และเมื่อ TLO นับจนเกิดโอเวอร์โฟลว์ หรือ TLO วนมานับค่า 0 จะทำให้เกิดเหตุการณ์ที่เรียกว่าอู่อั้วรีโหลด คือ TLO จะเริ่มต้นนับค่าใหม่จากค่าเริ่มต้นที่ TH0

	ใบเนื้อหา	หน้าที่ 16
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	
ชื่อเรื่อง การใช้งานโมดูล Timer/Counter		
<p>4. จากรูปที่ 1.15 โหมด 3 คือการนับสัญญาณนาฬิกาภายในขนาด 8 บิต โดยรีจิสเตอร์ TL0,TH0 จะทำหน้าที่นับค่า 8 บิตพร้อมกัน</p> <p>ส่วนการควบคุมการนับสามารถกระทำได้ 2 รูปแบบคือ กระทำด้วยรูปแบบกระบวนการทางฮาร์ดแวร์ หรือ กระทำด้วยรูปแบบกระบวนการทางซอฟต์แวร์ ซึ่งกำหนดได้ด้วยบิต GATE ของรีจิสเตอร์ TMOD ถ้าบิต GATE = ‘0’ เป็นการนับที่ถูกควบคุมด้วยกระบวนการทางซอฟต์แวร์ เป็นต้น</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ AT89C51ED2 ทำงานในลักษณะ Timer โหมด 1 และประยุกต์ใช้ Timer0 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p> <pre>void init_timer0_timer16bit_mode1(){ TMOD = (TMOD & 0xf0) 0x01; TR0 = 0; } void delay1S(){ //XTAL = 12MHz , T = 1/(12MHz/12) = 1uS char count; for(count=20;count>0;count--){ // T = 50mS x 20 = 1S TH0 = 0x3c; //Timer0 Count = 65536 – 15536 = 50000uS TL0 = 0xb0; TF0 = 0; TR0 = 1; while(TF0 == 0); TR0 = 0; } }</pre> <p>4.1.2 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 1 ใน AT89C51ED2</p> <p>โมดูล Timer/Counter 1 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้ 3 รูปแบบแสดงดังรูปที่ 1.16 ด้วยการกำหนดที่รีจิสเตอร์ TMOD ส่วนเวลาของการนับหนึ่งค่าจะใช้เวลาเท่าไรสามารถหาได้จากสมการ $T = 1/(XTAL/12)$</p>		

	ใบเนื้อหา		หน้าที่ 17
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

MODE	TIMER 1 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	00H	80H
1	16-bit Timer	10H	90H
2	8-bit Auto-Reload	20H	A0H
3	does not run	30H	B0H

NOTES:
1. The Timer is turned ON/OFF by setting/clearing bit TR0 in the software.
2. The Timer is turned ON/OFF by the 1 to 0 transition on INT0 (P3.2) when TR0 = 1 (hardware control).

รูปที่ 1.16 แสดงตารางการกำหนดรูปแบบการทำงานในโหมด Timer ของ Timer1

1. จากรูปที่ 1.16 โหมด 0 คือการนับสัญญาณนาฬิกาภายในขนาด 13 บิต โดยรีจิสเตอร์ TL1 จะทำหน้าที่นับ 5 บิตล่าง ส่วน TH1 จะทำหน้าที่นับ 8 บิตบน หรืออาจกล่าวได้ว่าการนับแบบนี้เป็นการนับแบบ MOD32


2. จากรูปที่ 1.16 โหมด 1 คือการนับสัญญาณนาฬิกาภายในขนาด 16 บิต โดยรีจิสเตอร์ TL1 จะทำหน้าที่นับ 8 บิตล่าง ส่วน TH1 จะทำหน้าที่นับ 8 บิตบน

3. จากรูปที่ 1.16 โหมด 2 คือการนับสัญญาณนาฬิกาภายในขนาด 8 บิต โดยรีจิสเตอร์ TL1 จะทำหน้าที่นับค่า 8 บิตที่ถูกกำหนดโดยรีจิสเตอร์ TH1 และเมื่อ TL1 นับจนเกิดโอเวอร์โฟลว์ หรือ TL1 วนมานับค่า 0 จะทำให้เกิดเหตุการณ์ที่เรียกว่าอู่อั้วรีโหลด คือ TL1 จะเริ่มต้นนับค่าใหม่จากค่าเริ่มต้นที่ TH1

ส่วนการควบคุมการนับสามารถกระทำได้ 2 รูปแบบคือ กระทำด้วยรูปแบบกระบวนการทางฮาร์ดแวร์ หรือ กระทำด้วยรูปแบบกระบวนการทางซอฟต์แวร์ ซึ่งกำหนดได้ด้วยบิต GATE ของรีจิสเตอร์ TMOD ถ้าบิต GATE = ‘0’ เป็นการนับที่ถูกควบคุมด้วยกระบวนการทางซอฟต์แวร์ เป็นต้น

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ AT89C51ED2 ทำงานในลักษณะ Timer โหมด 1 และประยุกต์ใช้ Timer1 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที

```
void init_timer1_timer16bit_mode1(){  
    TMOD = (TMOD & 0x0f) | 0x10;  
    TR1 = 0;  
}
```

	ใบเนื้อหา		หน้าที่ 18
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter			
<pre>void delay1S(){ //XTAL = 12MHz , T = 1/(12MHz/12) = 1uS char count; for(count=20;count>0;count--){ // T = 50mS x 20 = 1S TH1 = 0x3c; //Timer1 Count = 65536 – 15536 = 50000uS TL1 = 0xb0; TF1 = 0; TR1 = 1; while(TF1 == 0); TR1 = 0; } }</pre>			

4.1.3 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 2 ใน AT89C51ED2

โมดูล Timer/Counter 2 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้ 5 รูปแบบแสดงดังรูปที่ 1.17 ด้วยการกำหนดที่รีจิสเตอร์ T2CON ส่วนเวลาของการนับหนึ่งค่าจะใช้เวลาเท่าไรสามารถหาได้จากสมการ $T = 1/(XTAL/12)$ โดยโหมดที่ 2 ถึง 4 ใช้สำหรับสร้างฐานเวลาให้แก่โมดูลการสื่อสาร UART

MODE	T2CON	
	INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
16-bit Auto-Reload	00H	08H
16-bit Capture	01H	09H
BAUD rate generator receive & transmit same baud rate	34H	36H
receive only	24H	26H
transmit only	14H	16H


NOTES:

1. Capture/Reload occurs only on Timer/Counter overflow.

2. Capture/Reload occurs on Timer/Counter overflow and a 1 to 0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generating mode.

รูปที่ 1.17 แสดงตารางการกำหนดรูปแบบการทำงานในโหมด Timer ของ Timer2


1. จากรูปที่ 1.17 โหมด 0 คือการนับสัญญาณนาฬิกาภายในขนาด 16 บิต โดยรีจิสเตอร์ TL2 จะทำหน้าที่นับค่า 8 บิตล่าง ส่วน TH2 จะทำหน้าที่นับค่า 8 บิตบน ที่ถูกกำหนดโดยรีจิสเตอร์ RCAP2H และ RCAP2L เมื่อ TL2 และ TH2 นับจนเกิดโอเวอร์โฟลว์ หรือ TL2 และ TH2 วนมานับค่า 0 จะทำให้เกิดเหตุการณ์ที่เรียกว่าอโต้รีโหลด คือ TL2 และ TH2 จะเริ่มต้นนับค่าใหม่จากค่าเริ่มต้นที่ RCAP2H และ RCAP2L

	ใบเนื้อหา	หน้าที่ 19
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	


ชื่อเรื่อง การใช้งานโมดูล Timer/Counter
<p>2. จากรูปที่ 1.17 โหมด 1 คือการสุ่มนับสัญญาณนาฬิกาภายในขนาด 16 บิต โดยรีจิสเตอร์ TL2 จะทำหน้าที่นับ 8 บิตล่าง ส่วน TH2 จะทำหน้าที่นับ 8 บิตบน ที่เรียกว่าการ Capture สัญญาณ เพื่อคำนวณกลับเป็นค่าความถี่ของสัญญาณ</p> <p>ส่วนการควบคุมการนับสามารถกระทำได้ 2 รูปแบบคือ กระทำด้วยรูปแบบกระบวนการทางฮาร์ดแวร์ หรือ กระทำด้วยรูปแบบกระบวนการทางซอฟต์แวร์ ซึ่งกำหนดได้ด้วยบิต EXEN2 ของรีจิสเตอร์ T2CON ถ้าบิต EXEN2 = '0' เป็นการนับที่ถูกควบคุมด้วยกระบวนการทางซอฟต์แวร์ เป็นต้น</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer2 ของ AT89C51ED2 ทำงานในลักษณะ Timer โหมด 0 และประยุกต์ใช้ Timer2 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p> <pre>void init_timer2_timer16bit_mode0(){ T2CON = 0; TR2 = 0; } void delay1S(){ //XTAL = 12MHz , T = 1/(12MHz/12) = 1uS char count; for(count=20;count>0;count--){ // T = 50mS x 20 = 1S TH2 = 0x3c; //Timer2 Count = 65536 - 15536 = 50000uS TL2 = 0xb0; TF2 = 0; TR2 = 1; while(TF2 == 0); TR2 = 0; } }</pre>

4.2 การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ PIC16F887


โดยทั่วไปคาบเวลาของแมชชีนไทม์ของไมโครคอนโทรลเลอร์ตระกูล PIC16F หาได้จากสมการ $T = 1/(XTAL/4)$ เพื่อใช้ในการคำนวณหาเวลาของการนับหนึ่งค่าจะใช้เวลากี่วินาที โดยไมโครคอนโทรลเลอร์ PIC16F887 มีโมดูล Timer/Counter อยู่ภายในทั้งหมด 3 ตัว และแต่ละตัวสามารถกำหนดให้ทำงานในรูปแบบ Timer ได้ทั้งหมดและมีความละเอียดของการนับที่สามารถกำหนดได้ และสามารถนับได้ละเอียดสูงสุด 16 บิต คือ Timer1 ส่วน Timer0 และ Timer 2 สามารถนับได้ละเอียดสูงสุด 8 บิต โดย Timer ต่าง ๆ สามารถกำหนดให้ทำงานในลักษณะ Timer ด้วยการกำหนดรีจิสเตอร์ที่เกี่ยวข้องดังนี้

	ใบเนื้อหา	หน้าที่ 20
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	


ชื่อเรื่อง การใช้งานโมดูล Timer/Counter
<p>4.2.1 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 0 ใน PIC16F887</p> <p>โมดูล Timer/Counter 0 ใน PIC16F887 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้โดยกำหนดที่บิต T0CS ให้มีค่าเป็นลอจิก ‘0’ เพื่อทำการนับค่าแมชชีนไซเคิลภายในไมโครคอนโทรลเลอร์ และยังสามารถกำหนดค่า Prescaler ได้ตามข้อมูลการใช้งานรีจิสเตอร์ OPTION_REG ดังรายละเอียดในรูปที่ 1.4</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ PIC16F887 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer0 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p> <pre>void init_timer0_timer8bit(){ OPTION_REG &= 0xf0; OPTION_REG = 0x07; //Prescaler 1:256 OPTION_REGbits.T0CS = 0; INTCONbits.T0IF = 0; TMR0 = 0; } void delay1S(){ //XTAL = 20MHz , T = 1/((20MHz/4)/256) = 51.2uS char count; for(count=76;count>0;count--){ // T = (51.2uS x 256) x 76 = 1S TMR0 = 0; INTCONbits.T0IF = 0; while(INTCONbits.T0IF == 0); } }</pre>
<p>4.2.2 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 1 ใน PIC16F887</p> <p>โมดูล Timer/Counter 1 ใน PIC16F887 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้โดยกำหนดที่บิต TMR1CS ให้มีค่าเป็นลอจิก ‘0’ เพื่อทำการนับค่าแมชชีนไซเคิลภายในไมโครคอนโทรลเลอร์ และยังสามารถกำหนดค่า Prescaler ได้ตามข้อมูลการใช้งานรีจิสเตอร์ T1CON ดังรายละเอียดในรูปที่ 1.6</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ PIC16F887 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer1 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p>

	ใบเนื้อหา		หน้าที่ 21
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		


ชื่อเรื่อง การใช้งานโมดูล Timer/Counter			
<pre>void init_timer1_timer16bit(){ T1CONbits.T1CKPS1 = 1; //Prescaler 1:8 T1CONbits.T1CKPS0 = 1; T1CONbits.TMR1CS = 0; T1CONbits.TMR1ON = 0; PIR1bits.TMR1IF = 0; TMR1 = 0; } void delay1S(){ //XTAL = 20MHz , T = 1/((20MHz/4)/8) = 1.6uS char count; for(count=25;count>0;count--){ // T = (1.6uS x 25000) x 25= 1S TMR1 = 40536; PIR1bits.TMR1IF = 0; T1CONbits.TMR1ON = 1; while(PIR1bits.TMR1IF == 0); T1CONbits.TMR1ON = 0; } }</pre>			
4.2.3 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 2 ใน PIC16F887			
โมดูล Timer/Counter 2 ใน PIC16F887 ถูกออกแบบให้ทำงานในลักษณะ Timer เพื่อทำการนับค่าแมชชีนไซเคิลภายในไมโครคอนโทรลเลอร์ และยังสามารถกำหนดค่า Prescaler , Postscaler ได้ตามข้อมูลการใช้งานรีจิสเตอร์ T2CON ดังรายละเอียดในรูปที่ 1.7			
ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer2 ของ PIC16F887 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer2 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที			
<pre>void init_timer2_timer8bit(){ T2CON = 0x7b; //Prescaler 1:16 Postscaler 1:16 T2CONbits.TMR2ON = 0; PIR1bits.TMR2IF = 0; TMR2 = 0; }</pre>			

	ใบเนื้อหา	หน้าที่ 22
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter
<pre>void delay1S(){ //XTAL = 20MHz , T = 1/((20MHz/4)/16/16) = 51.2uS char count; for(count=76;count>0;count--){ // // T = (51.2uS x 256) x 76 = 1S TMR2 = 0; PIR1bits.TMR2IF = 0; T2CONbits.TMR2ON = 1; while(PIR1bits.TMR2IF == 0); T2CONbits.TMR2ON = 0; } }</pre>
<h3>4.3 การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ ATMEGA32</h3> <p>โดยทั่วไปคาบเวลาของแมชชีนไจกิลของไมโครคอนโทรลเลอร์ตระกูล AVR หาได้จากสมการ $T = 1/XTAL$ เพื่อใช้ในการคำนวณหาเวลาของการนับหนึ่งค่าจะใช้เวลากี่วินาที โดยไมโครคอนโทรลเลอร์ ATMEGA32 มีโมดูล Timer/Counter อยู่ภายในทั้งหมด 3 ตัว และแต่ละตัวสามารถกำหนดให้ทำงานในรูปแบบ Timer ได้ทั้งหมด และมีความละเอียดของการนับที่สามารถกำหนดได้ และสามารถนับได้ละเอียดสูงสุด 16 บิต คือ Timer1 ส่วน Timer0 และ Timer 2 สามารถนับได้ละเอียดสูงสุด 8 บิต โดย Timer ต่าง ๆ สามารถกำหนดให้ทำงานในลักษณะ Timer ด้วยการกำหนดรีจิสเตอร์ที่เกี่ยวข้องดังนี้</p> <h4>4.3.1 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 0 ใน ATMEGA32</h4> <p>โมดูล Timer/Counter 0 ใน ATMEGA32 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้ เพื่อทำการนับค่าแมชชีนไจกิลภายในไมโครคอนโทรลเลอร์ และยังสามารถกำหนดค่า Prescaler ได้จากบิต CS02:CS00 ของรีจิสเตอร์ TCCR0 ดังรายละเอียดในรูปที่ 1.8</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ ATMEGA32 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer0 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p> <pre>void init_timer0_timer8bit(){ TCCR0 = (1<<CS02) (1<<CS00); // Prescaler 1:1024 TCNT0 = 0; }</pre>

	ใบเนื้อหา	หน้าที่ 23
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter
<pre>void delay1S(){ //XTAL = 16MHz , T = 1/(16MHz/1024) = 64uS char count; for(count=61;count>0;count--){ // T = (64uS x 256) x 61 = 1S if((TIFR & (1<<TOV0)) != 0) TIFR = (1<<TOV0); TCNT0 = 0; while((TIFR & (1<<TOV0)) == 0); } }</pre>
4.3.2 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 1 ใน ATMEGA32
<p>โมดูล Timer/Counter 1 ใน ATMEGA32 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้ เพื่อทำการนับค่าแมชชีนไซเกิลภายในไมโครคอนโทรลเลอร์ และยังสามารถกำหนดค่า Prescaler ได้จากบิต CS12:CS10 ของรีจิสเตอร์ TCCR1B ดังรายละเอียดในรูปที่ 1.10</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ ATMEGA32 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer1 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p> <pre>void init_timer1_timer16bit(){ TCCR1B = (1<<CS12) (1<<CS10); // Prescaler 1:1024 } void delay1S(){ //XTAL = 16MHz , T = 1/(16MHz/1024) = 64uS if((TIFR & (1<<TOV1)) != 0) TIFR = (1<<TOV1); TCNT1 = 49911; // T = 64uS x 15625 = 1S while((TIFR & (1<<TOV1)) == 0); }</pre>
4.3.3 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 2 ใน ATMEGA32
<p>โมดูล Timer/Counter 2 ใน ATMEGA32 สามารถกำหนดให้ทำงานในลักษณะ Timer ได้ เพื่อทำการนับค่าแมชชีนไซเกิลภายในไมโครคอนโทรลเลอร์ และยังสามารถกำหนดค่า Prescaler ได้จากบิต CS22:CS20 ของรีจิสเตอร์ TCCR2 ดังรายละเอียดในรูปที่ 1.11</p> <p>ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer2 ของ ATMEGA32 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer2 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที</p> <pre>void init_timer2_timer8bit(){ TCCR2 = (1<<CS22) (1<<CS21) (1<<CS20); // Prescaler 1:1024 }</pre>

	ใบเนื้อหา		หน้าที่ 24
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004		หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter		

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter

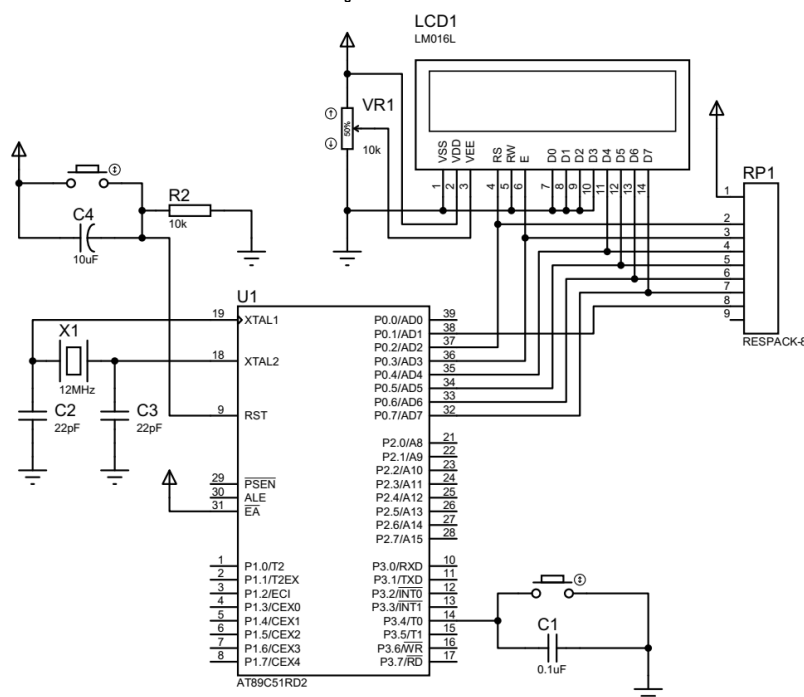
```

void delay1S(){ //XTAL = 16MHz , T = 1/(16MHz/1024) = 64uS
    char count;
    for(count=61;count>0;count--){ // T = (64uS x 256) x 61 = 1S
        if((TIFR & (1<<TOV2)) != 0) TIFR |= (1<<TOV2);
        TCNT2 = 0;
        while((TIFR & (1<<TOV2)) == 0);
    }
}


```

5. การประยุกต์ใช้งานโมดูล Timer/Counter เพื่อสร้างวงจรรนับ

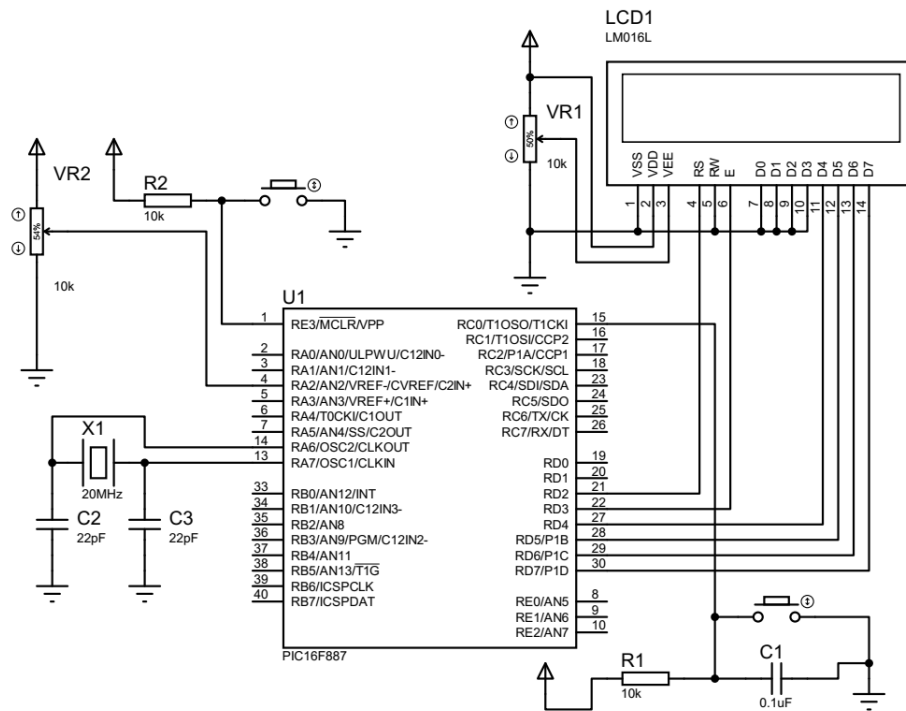
การประยุกต์ใช้งานโมดูล Timer/Counter เพื่อสร้างวงจรรนับ สามารถทำได้โดยใช้คุณสมบัติของโมดูล Timer/Counter เชื่อมต่อกับสัญญาณนาฬิกาหรือสัญญาณพัลส์ภายนอก และทำการกำหนดให้โมดูล Timer/Counter ทำงานในลักษณะของ Counter ถ้าใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C51ED2 สามารถเลือกใช้งานโมดูล Timer/Counter เพื่อทำงานในลักษณะ Counter ได้ทั้งหมด 3 โมดูล แต่ถ้าใช้ไมโครคอนโทรลเลอร์เบอร์ PIC16F877 และ ATMEGA32 สามารถเลือกใช้งานโมดูล Timer/Counter เพื่อทำงานในลักษณะ Counter ได้ทั้งหมด 2 โมดูล ส่วนตัวอย่างวงจรแสดงการใช้งานโมดูล Timer/Counter เพื่อสร้างวงจรรนับจะแสดงดังรูปด้านล่าง



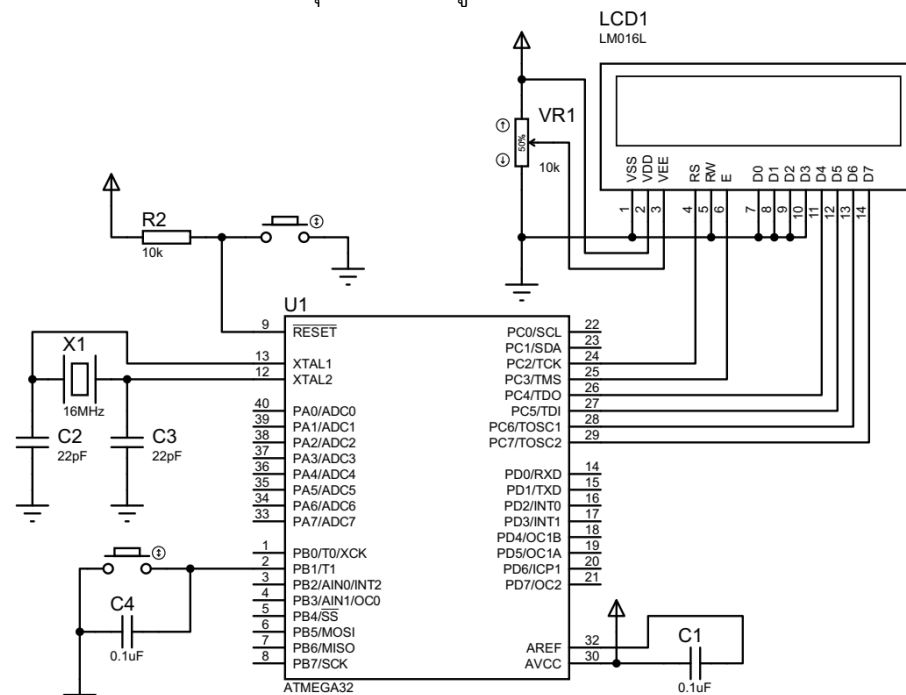
รูปที่ 1.18 แสดงตัวอย่างการประยุกต์ใช้งานโมดูล Timer0 ของ AT89C51RD2 เพื่อสร้างวงจรรนับ

	ใบเนื้อหา	หน้าที่ 25
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	

ชื่อเรื่อง การใช้งานโมดูล Timer/Counter



รูปที่ 1.19 แสดงตัวอย่างการประยุกต์ใช้งานโมดูล Timer1 ของ PIC16F887 เพื่อสร้างวงจรรนับ



รูปที่ 1.20 แสดงตัวอย่างการประยุกต์ใช้งานโมดูล Timer1 ของ ATMEGA32 เพื่อสร้างวงจรรนับ

	แบบฝึกหัด	หน้าที่ 1
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	
ชื่อเรื่อง การใช้งานโมดูล Timer/Counter		
<p><u>คำสั่ง</u> จงตอบคำถามต่อไปนี้ให้ถูกต้อง</p> <ol style="list-style-type: none"> ให้อธิบายหน้าที่การทำงานของโมดูล Timer/Counter ในลักษณะการทำงานเป็น Counter ให้อธิบายหน้าที่การทำงานของโมดูล Timer/Counter ในลักษณะการทำงานเป็น Timer ในไมโครคอนโทรลเลอร์ AT89C51ED2 มีโมดูล Timer/Counter กี่โมดูล และมีความสามารถอย่างไร ในไมโครคอนโทรลเลอร์ PIC16F887 มีโมดูล Timer/Counter กี่โมดูล และมีความสามารถอย่างไร ในไมโครคอนโทรลเลอร์ ATMEGA32 มีโมดูล Timer/Counter กี่โมดูล และมีความสามารถอย่างไร รีจิสเตอร์ T2CON ของไมโครคอนโทรลเลอร์ AT89C51ED2 มีความสำคัญอย่างไร รีจิสเตอร์ OPTION_REG ของไมโครคอนโทรลเลอร์ PIC16F887 มีความสำคัญอย่างไร 		

	แบบฝึกหัด	หน้าที่ 2
	ชื่อวิชา ดิจิทัลและไมโครคอนโทรลเลอร์ รหัสวิชา 30127-2004	หน่วยที่ 6
	ชื่อหน่วย การใช้งานโมดูล Timer/Counter	
ชื่อเรื่อง การใช้งานโมดูล Timer/Counter		
<p>8. รีจิสเตอร์ TCCR2 ของไมโครคอนโทรลเลอร์ ATMEGA32 มีความสำคัญอย่างไร</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>9. อธิบายวิธีการใช้งานบิต TOV0 ของรีจิสเตอร์ TIFR ของไมโครคอนโทรลเลอร์ ATMEGA32</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>10. อธิบายวิธีการนับคาบเวลาของแมชชีนไซเคิลโดยใช้ Timer2 ในไมโครคอนโทรลเลอร์ PIC16F887</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>11. ให้เขียนโปรแกรมเพื่อกำหนดค่าเริ่มต้นให้โมดูล Timer/Counter 1 ของ AT89C51ED2 ทำงานแบบ Timer 16 bit</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>12. ให้เขียนโปรแกรมเพื่อกำหนดค่าเริ่มต้นให้โมดูล Timer/Counter 0 ของ PIC16F887 ทำงานแบบ Timer</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>13. ให้เขียนโปรแกรมเพื่อกำหนดค่าเริ่มต้นให้โมดูล Timer/Counter 2 ของ ATMEGA32 ทำงานแบบ Counter</p> <p>.....</p> <p>.....</p> <p>.....</p>		