

การใช้งานโมดูล Timer/Counter

รหัสวิชา 30127-2004 (2-3-3) ดิจิทัลและไมโครคอนโทรลเลอร์

Digital And Microcontroller

1

การใช้งานโมดูล Timer/Counter

การใช้งานโมดูล Timer/Counter

- คุณสมบัติของโมดูล Timer/Counter
- รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter
 - รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ AT89C51ED2
 - รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ PIC16F887
 - รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ ATMEGA32
- การทำงานของโมดูล Timer/Counter ในลักษณะ Counter
 - การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ AT89C51ED2
 - การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ PIC16F887
 - การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ ATMEGA32
- การทำงานของโมดูล Timer/Counter ในลักษณะ Timer
 - การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ AT89C51ED2
 - การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ PIC16F887
 - การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ ATMEGA32
- การประยุกต์ใช้งานโมดูล Timer/Counter เพื่อสร้างวงจรนับ

Digital And Microcontroller

2

การใช้งานโมดูล Timer/Counter

1. คุณสมบัติของโมดูล Timer/Counter

โมดูล Timer/Counter คือโมดูลพิเศษที่มีคุณสมบัติในการทำหน้าที่ 2 ลักษณะ คือ หน้าที่ในการทำงานรูปแบบ Timer เพื่อนับสัญญาณนาฬิกาภายในของไมโครคอนโทรลเลอร์ที่เกิดจากคาบเวลาของการประมวลผล และทำหน้าที่ในรูปแบบ Counter ที่เป็นการนับสัญญาณพัลส์จากภายนอกผ่านขาสัญญาณที่เชื่อมต่อกับโมดูล Timer/Counter นั้น ๆ

ความละเอียดในการนับของโมดูล Timer/Counter สามารถแบ่งความละเอียดในการนับของ Timer/Counter ได้เป็น 2 กลุ่ม คือ กลุ่ม Timer/Counter ที่มีความละเอียดในการนับ 8 บิต และกลุ่ม Timer/Counter ที่มีความละเอียดในการนับ 16 บิต

คาบเวลาของแรมซิงไพล์ของไมโครคอนโทรลเลอร์ตระกูล MCS-51 หาได้จากสมการ $T = 1/(XTAL/12)$ ตระกูล PIC16F หาได้จากสมการ $T = 1/(XTAL/4)$ และตระกูล AVR หาได้จากสมการ $T = 1/XTAL$

Digital And Microcontroller

3

การใช้งานโมดูล Timer/Counter

ไมโครคอนโทรลเลอร์เบอร์ AT89C51ED2 มีโมดูล Timer/Counter อยู่จำนวน 3 ตัวได้แก่ Timer0 , Timer1 และ Timer2 โดย Timer/Counter ทั้ง 3 ตัวมีความละเอียดในการนับได้สูงสุด 16 บิต

ไมโครคอนโทรลเลอร์เบอร์ PIC16F887 มีโมดูล 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

ไมโครคอนโทรลเลอร์เบอร์ ATMEGA32 มีโมดูล 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

Digital And Microcontroller

4

การใช้งานโมดูล Timer/Counter

2. รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter

2.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ AT89C51ED2

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของไมโครคอนโทรลเลอร์ AT89C51ED2 แยกออกตามโมดูล Timer/Counter

2.1.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 0 ของ AT89C51ED2

TCON คือรีจิสเตอร์ที่ควบคุมการทำงานของ Timer0 และ Timer1 โดยจะใช้บิต TR0 ในการเปิดการทำงานโมดูล Timer0 และบิต TF0 เพื่อตรวจสอบสถานะการนับเกินของโมดูล Timer0

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.

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TCON

Digital And Microcontroller

5

การใช้งานโมดูล Timer/Counter

TMOD คือรีจิสเตอร์ที่ทำหน้าที่ในการกำหนดโหมดการทำงานของ Timer0 และ Timer1 โดยจะแบ่งออกเป็น 2 กลุ่ม คือ 4 บิตบน ทำหน้าที่ในการกำหนดโหมดการทำงานของ Timer1 ส่วน 4 บิตล่าง ทำหน้าที่ในการกำหนดโหมดการทำงานของ Timer0

GATE	C/T	M1	M0	GATE	C/T	M1	M0
------	-----	----	----	------	-----	----	----

TIMER 1 When TR1 (in TCON) is set and GATE = 1, TIMER/COUNTERs will run only while INTx pin is high (hardware control). When GATE = 0, TIMER/COUNTERs will run only while TR1 = 1 (software control).

C/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	13-bit Timer (MCS-48 compatible)
0	1	16-bit Timer/Counter
1	0	8-bit Auto-Reload Timer/Counter
1	1	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. (Timer 1) Timer/Counter 1 stopped.

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TMOD

TH0 และ TL0 คือรีจิสเตอร์ที่ทำหน้าที่เก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 0 โดย TH0 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TL0 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง

Digital And Microcontroller

6

การใช้งานโมดูล Timer/Counter

2.1.2 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 1 ของ AT89C51ED2

TCON และ TMOD เป็นรีจิสเตอร์ที่มีหน้าที่ควบคุม และกำหนดโหมดการทำงานของ Timer1 ซึ่งมีรายละเอียดหน้าที่การทำงานของรีจิสเตอร์เช่นเดียวกับหัวข้อ 2.1.1

TH1 และ TL1 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของ Timer/Counter 1 โดย TH1 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตบน และ TL1 ทำหน้าที่ในการเก็บผลลัพธ์ 8 บิตล่าง

2.1.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 2 ของ AT89C51ED2

T2CON คือรีจิสเตอร์ที่ทำหน้าที่ควบคุม และเลือกโหมดการทำงานของ Timer/Counter 2 โดยบิต TF2 แสดงสถานะการนับเกินของโมดูล Timer2, บิต TR2 ทำหน้าที่ในการเปิดการทำงานของโมดูล Timer2 และบิต C/T ทำหน้าที่ในการเลือกโหมดการทำงานของ Timer2

การใช้งานโมดูล Timer/Counter

TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RE2
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/RE2	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.						

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ T2CON

TH2 และ TL2 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์การนับของโมดูล Timer/Counter 2 โดย TH2 ทำหน้าที่เก็บผลลัพธ์ 8 บิตบน และ TL2 ทำหน้าที่เก็บผลลัพธ์ 8 บิตล่าง

การใช้งานโมดูล Timer/Counter

2.2 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ PIC16F887

2.2.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 0 ของ PIC16F887

OPTION_REG คือรีจิสเตอร์พิเศษของไมโครคอนโทรลเลอร์ PIC16F887 ที่มีบิตในการกำหนดคุณสมบัติของโมดูล Timer0 ได้แก่บิต TOCS, TOSE, PSA และ PS2:PS0

RW-1	RW-1	RW-1	RW-1	RW-1	RW-1	RW-1	RW-1
RBP0	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
bit 7							

bit 5 **TOCS:** TMR0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (F_{OSC}/4)

bit 4 **TOSE:** 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

การใช้งานโมดูล Timer/Counter

TMR0 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 0 มีความละเอียดในการนับขนาด 8 บิต

INTCON คือรีจิสเตอร์ที่ทำหน้าที่ควบคุมการอินเตอร์รัพต์ของ PIC16F887 และมีส่วนที่เกี่ยวข้องกับการทำงานของ Timer0 คือบิต TOIF ซึ่งมีหน้าที่แสดงสถานะการนับเกินของโมดูล Timer0 และบิต TOIE ซึ่งเป็นที่ยอมให้เกิดการอินเตอร์รัพต์จาก Timer0

RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-x
GIE	PEIE	TOIE	INTE	RBFIE ⁽¹⁾	TOIF ⁽⁰⁾	INTF	RBIF
bit 7							

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ 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 คือบิตที่ควบคุมการเปิดการทำงานของโมดูล Timer1, บิต TMR1CS คือบิตที่ทำหน้าที่เลือกแหล่งกำเนิดสัญญาณนาฬิกาเพื่อการนับ, บิต T1CKPS1:T1CKPS0 คือบิตที่ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ของโมดูล Timer1

การใช้งานโมดูล Timer/Counter

RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0
T1GINT ⁽¹⁾	TMR1GE ⁽⁰⁾	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNCR	TMR1CS	TMR1ON
bit 7							

bit 7 **T1GINT:** 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⁽⁰⁾
#TMR1ON = 0
#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 **T1SYNCR:** Timer1 External Clock Input Synchronization Control bit
#TMR1CS = 0
1 = Do not synchronize external clock input
0 = Synchronize external clock input
#TMR1CS = 1
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 (F_{OSC}/4)

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

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ T1CON

การใช้งานโมดูล Timer/Counter

PIR1 คือรีจิสเตอร์พิเศษภายในไมโครคอนโทรลเลอร์ PIC16F887 ซึ่งมีบิตที่เกี่ยวข้องกับการทำงานของ Timer1 คือบิต TMR1IF คือเมื่อ Timer1 เกิดเหตุการณ์การนับเกินจะทำการนับมีค่าเป็นลอจิก '1'

2.2.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 2 ของ PIC16F887

TMR2 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 2 มีความละเอียดในการนับขนาด 8 บิต

T2CON คือรีจิสเตอร์ที่ควบคุมการทำงานของโมดูล Timer/Counter 2 โดยมีบิตที่สำคัญได้แก่ บิต TMR2ON คือบิตที่ควบคุมการเปิดการทำงานของโมดูล Timer1, บิต T2CKPS1:T2CKPS0 คือบิตที่ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ของโมดูล Timer2, บิต TOUTPS3:TOUTPS0 คือบิตที่ทำหน้าที่ในการกำหนดค่าโพสเกลเลอร์ของโมดูล Timer2

การใช้งานโมดูล Timer/Counter

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	TZCKPS1	TZCKPS0
bit 7							bit 0

Unimplemented: Read as '0'

TOUTPS<3:0>: Timer2 Output Postscaler Select bits

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

bit 2 **TMR2ON**: Timer2 On bit
 1 = Timer2 is on
 0 = Timer2 is off

bit 1-0 **TZCKPS<1:0>**: Timer2 Clock Prescale Select bits
 00 = Prescaler is 1
 01 = Prescaler is 4
 1x = Prescaler is 16

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TZCON

Digital And Microcontroller

13

การใช้งานโมดูล Timer/Counter

2.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter ของ ATMEGA32

2.2.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 0 ของ ATMEGA32

TCR0 คือรีจิสเตอร์ขนาด 8 บิตที่ทำหน้าที่ในการกำหนดโหมดการทำงานของโมดูล Timer0 ให้ทำงานในลักษณะต่าง ๆ และกำหนดค่า Prescale ของ Timer0 โดยบิตที่เกี่ยวข้องกับการกำหนดให้ Timer0 ทำงานในรูปแบบ Timer/Counter คือบิต CS02:CS00 ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ให้แก่ Timer0

7	6	5	4	3	2	1	0
FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk _{IO/8} (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)
1	0	1	clk _{IO/1024} (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TCCR0

Digital And Microcontroller

14

การใช้งานโมดูล Timer/Counter

TCNT0 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 0 มีความละเอียดในการนับขนาด 8 บิต

TIFR คือรีจิสเตอร์ที่ทำหน้าที่แสดงสถานะการทำงานของโมดูล Timer/Counter ซึ่งมีบิตที่เกี่ยวข้องกับการทำงานของ Timer0 คือบิต TOV0 จะทำหน้าที่แสดงสถานะการนับเกินของโมดูล Timer0

7	6	5	4	3	2	1	0
OCF2	TOV2	ICF1	OCF1A	OCF1B	TOW1	OCF0	TOV0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TIFR

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

Digital And Microcontroller

15

การใช้งานโมดูล Timer/Counter

TCCR1B คือรีจิสเตอร์ที่มีหน้าที่ในการกำหนดโหมดในการทำงานของโมดูล Timer1 เลือกแหล่งกำเนิดสัญญาณนาฬิกาเป็นต้น โดยรีจิสเตอร์นี้จะมีบิตที่เกี่ยวข้องกับการทำงานของ Timer1 คือ บิต ICNC1 ทำหน้าที่ในการกำหนดการกรองสัญญาณรบกวนในขณะที่ทำงานในลักษณะแคปเจอร์ สัญญาณอินพุต, บิต ICES1 ทำหน้าที่ในการเลือกขอบขาสัญญาณที่ต้องการจะทำการแคปเจอร์ และบิต CS12:CS10 ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ให้แก่ Timer1

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 Canceller

Setting this bit (to one) activates the Input Capture Noise Canceller. When the Noise Canceller 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 Canceller 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.

Digital And Microcontroller

16

การใช้งานโมดูล Timer/Counter

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.

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TCCR1B

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

Digital And Microcontroller

17

การใช้งานโมดูล Timer/Counter

2.2.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของโมดูล Timer/Counter 2 ของ ATMEGA32

TCCR2 คือรีจิสเตอร์ที่ทำหน้าที่กำหนดโหมดการทำงานของโมดูล Timer2 ให้ทำงานในลักษณะต่าง ๆ และกำหนดค่า Prescale ของ Timer2 โดยบิตที่คือบิต CS22:CS20 ทำหน้าที่ในการกำหนดค่าปริสเกลเลอร์ให้แก่ Timer2

7	6	5	4	3	2	1	0
FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20
R/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 _{IO/8} (No prescaling)
0	1	0	clk _{IO/8} (From prescaler)
0	1	1	clk _{IO/32} (From prescaler)
1	0	0	clk _{IO/64} (From prescaler)
1	0	1	clk _{IO/128} (From prescaler)
1	1	0	clk _{IO/256} (From prescaler)
1	1	1	clk _{IO/1024} (From prescaler)

รูปแสดงข้อมูลบิตภายในรีจิสเตอร์ TCCR2

Digital And Microcontroller

18

การใช้งานโมดูล Timer/Counter

TCNT2 คือรีจิสเตอร์ที่ทำหน้าที่ในการเก็บผลลัพธ์ของการนับของโมดูล Timer/Counter 2 มีความละเอียดในการนับขนาด 8 บิต

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

การใช้งานโมดูล Timer/Counter

3. การทำงานของโมดูล Timer/Counter ในลักษณะ Counter

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

3.1.1 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 0 ใน AT89C51ED2
โมดูล Timer/Counter 0 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 4 รูปแบบด้วยการกำหนดที่รีจิสเตอร์ TMOD ส่วนสัญญาณนาฬิกาภายนอกจะถูกต่อที่ขา P3.4

MODE	COUNTER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	0AH	0CH
1	16-bit Timer	0BH	0DH
2	8-bit Auto-Reload	0EH	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).

การใช้งานโมดูล Timer/Counter

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;
    TL0 = 0;
    TR0 = 1;
}
```

การใช้งานโมดูล Timer/Counter

3.1.2 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 1 ใน AT89C51ED2

โมดูล Timer/Counter 1 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 3 รูปแบบ 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).

การใช้งานโมดูล Timer/Counter

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;
}
```

การใช้งานโมดูล Timer/Counter

3.1.3 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 2 ใน AT89C51ED2

โมดูล Timer/Counter 2 ใน AT89C51ED2 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 2 รูปแบบ 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.

การใช้งานโมดูล Timer/Counter

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;
}
```

Digital And Microcontroller

25

การใช้งานโมดูล Timer/Counter

3.2 การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ PIC16F887

ไมโครคอนโทรลเลอร์ PIC16F887 มีโมดูล Timer/Counter ที่สามารถกำหนดให้ทำงานในลักษณะ Counter ได้ 2 โมดูลคือ Timer0 และ Timer1

3.2.1 การทำงานในลักษณะ Counter ของโมดูล Timer/Counter 0 ใน PIC16F887

โมดูล Timer/Counter 0 ใน PIC16F887 สามารถกำหนดให้ทำงานในลักษณะ Counter ได้โดยกำหนดที่บิต TOCS ให้มีค่าเป็นลอจิก '1' เพื่อทำการนับค่าสัญญาณที่เข้ามาที่ขา TOCKI (RA4) ส่วนจะนับช่วงขอบขาขึ้น หรือขอบขาลงจะกำหนดได้จากบิต TOSE ของรีจิสเตอร์ OPTION_REG

ตัวอย่าง การเขียนฟังก์ชันเพื่อให้โมดูล Timer0 ของ PIC16F887 ทำงานในลักษณะ Counter เพื่อนับสัญญาณนาฬิกาที่ขา TOCKI และสามารถนับค่าได้สูงสุด 8 บิต

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

Digital And Microcontroller

26

การใช้งานโมดูล Timer/Counter

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) ส่วนการเปิดการทำงานของโมดูลจะกำหนดที่บิต TMR1ON ของรีจิสเตอร์ T1CON

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;
}
```

Digital And Microcontroller

27

การใช้งานโมดูล Timer/Counter

3.3 การทำงานของโมดูล Timer/Counter ในลักษณะ Counter ของ ATMEGA32

ไมโครคอนโทรลเลอร์เบอร์ ATMEGA32 ของตระกูล AVR มีโมดูล Timer/Counter ที่สามารถกำหนดให้ทำงานในลักษณะของ Counter ได้ 2 โมดูลคือ Timer0 และ Timer1

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 หรือขอบขาลงจะกำหนดได้จากบิต CS02:CS00 ของรีจิสเตอร์ TCCR0

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;
}
```

Digital And Microcontroller

28

การใช้งานโมดูล Timer/Counter

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 หรือขอบขาลงจะกำหนดได้จากบิต CS12:CS10 ของรีจิสเตอร์ TCCR1B

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;
}
```

Digital And Microcontroller

29

การใช้งานโมดูล Timer/Counter

4. การทำงานของโมดูล Timer/Counter ในลักษณะ Timer

การทำงานของโมดูล Timer/Counter ในลักษณะ Timer คือการให้โมดูล Timer/Counter ของไมโครคอนโทรลเลอร์นับสัญญาณอินพุตจากแหล่งกำเนิดภายใน ซึ่งก็คือคาบเวลาของการประมวลผลของ CPU หรือที่เรียกว่าคาบเวลาแมชชีนไซเคิล โดยคาบเวลานี้จะขึ้นอยู่กับชนิดของไมโครคอนโทรลเลอร์ และค่า XTAL ที่ต่ออยู่ภายนอก

4.1 การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ AT89C51ED2

โดยทั่วไปคาบเวลาของแมชชีนไซเคิลของไมโครคอนโทรลเลอร์ตระกูล MCS-51 หาได้จากสมการ $T = 1/(XTAL/12)$ เพื่อใช้หาคำนวนหาเวลาของการนับหนึ่งค่าจะใช้เวลาที่วินาที โดยไมโครคอนโทรลเลอร์ AT89C51ED2 มีโมดูล Timer/Counter อยู่ภายในทั้งหมด 3 ตัว และแต่ละตัวสามารถกำหนดให้ทำงานในรูปแบบ Timer ได้ทั้งหมดและมีความละเอียดของการนับได้ละเอียดสูงสุด 16 บิต

Digital And Microcontroller

30

การใช้งานโมดูล Timer/Counter

4.1.1 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 0 ใน AT89C51ED2

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ AT89C51ED2 ทำงานในลักษณะ Timer โหมด 1 และประยุกต์ใช้ Timer0 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที

```
void init_timer0_timer16bit_mode10()
{
    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;
    }
}
```

Digital And Microcontroller

31

การใช้งานโมดูล Timer/Counter

4.1.2 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 1 ใน AT89C51ED2

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

```
void init_timer1_timer16bit_mode10()
{
    TMOD = (TMOD & 0xf0) | 0x10;
    TR1 = 0;
}

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;
    }
}
```

Digital And Microcontroller

32

การใช้งานโมดูล Timer/Counter

4.1.3 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 2 ใน AT89C51ED2

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer2 ของ AT89C51ED2 ทำงานในลักษณะ Timer โหมด 0 และประยุกต์ใช้ Timer2 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที

```
void init_timer2_timer16bit_mode00()
{
    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;
    }
}
```

Digital And Microcontroller

33

การใช้งานโมดูล Timer/Counter

4.2 การทำงานของโมดูล Timer/Counter ในลักษณะ Timer ของ PIC16F887

โดยทั่วไปคาบเวลาของแมชชีนไจเกิลของไมโครคอนโทรลเลอร์ตระกูล PIC16F หาได้จากสมการ $T = 1/(XTAL/4)$ เพื่อใช้ในด้านหาค่าเวลาของการนับหนึ่งค่าจะใช้เวลากี่วินาที โดยไมโครคอนโทรลเลอร์ PIC16F887 มีโมดูล Timer/Counter อยู่ภายในทั้งหมด 3 ตัว แต่ละตัวสามารถกำหนดให้ทำงานในรูปแบบ Timer ได้ทั้งหมดและมีความละเอียดของการนับที่สามารถกำหนดได้ และสามารถนับได้ละเอียดสูงสุด 16 บิต คือ Timer1 ส่วน Timer0 และ Timer 2 สามารถนับได้ละเอียดสูงสุด 8 บิต

Digital And Microcontroller

34

การใช้งานโมดูล Timer/Counter

4.2.1 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 0 ใน PIC16F887

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer0 ของ PIC16F887 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer0 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที

```
void init_timer0_timer8bit()
{
    OPTION_REG &= 0xf0;
    OPTION_REG |= 0x07; //Prescaler 1:256
    OPTION_REGbits.TOCS = 0;
    INTCONbits.TOIF = 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.TOIF = 0;
        while(INTCONbits.TOIF == 0);
    }
}
```

Digital And Microcontroller

35

การใช้งานโมดูล Timer/Counter

4.2.2 การทำงานในลักษณะ Timer ของโมดูล Timer/Counter 1 ใน PIC16F887

ตัวอย่าง การเขียนฟังก์ชันเพื่อควบคุมให้โมดูล Timer1 ของ PIC16F887 ทำงานในลักษณะ Timer และประยุกต์ใช้ Timer1 สร้างฟังก์ชัน หน่วงเวลา 1 วินาที

```
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;
    }
}
```

Digital And Microcontroller

36

