ภาคผนวก ก

บอร์ด CP-JR6811

1. ส่วนประกอบของบอร์ด CP-JR6811

บอร์ด CP-JR6811 ถูกพัฒนาขึ้นโดยเอาคุณสมบัติต่างๆมารวมกันไว้ภายในบอร์ดเดียว สามารถใช้งานเป็นชุดจำลองการทำงาน(EMULATOR) เพื่อใช้ในขั้นตอนของการพัฒนาโปรแกรมและยังสามารถทำการเขียนข้อมูลเข้าในตัวซีพียู เพื่อนำไปใช้เป็นบอร์ดคอนโทรลได้ทันที ทำให้สะดวกและประหยัดมากขึ้น โดยไม่ต้องใช้เครื่องมือในการโปรแกรมซีพียูมาใช้ในการพัฒนาโปรแกรมอีกซึ่งมีคุณสมบัติทางเทคนิคดังนี้

1. CPU 68SEC811E2
2. 2 KBYTE EEPROM WITH BLOCK PROTECT & SECURITY
3. 256 BYTE RAM
4. 16 BITS TIMER SYSTEM
   - 4 OUTPUT COMPARE CHANNEL
   - 3 INPUT CAPTURE CHANNEL
   - 1 INPUT CAPTURE OR OUTPUT CAMPARE(SOFTWARE SELECT)
5. 8 BIT PULSE ACCUMULATOR
6. REAL TIME INTERRUPT CIRCUIT
7. COMPUTER OPERATING PROPERIL(COP) WATCHDOG SYSTEM
8. SYNCHRONOUS SERIAL PERIPHERAL INTERFACE (SPI)
9. ASYNCHRONOUS SERIAL COMMUNICATIONS INTERFACE (SCI)
   - RS232 DRIVER ONBOARD
   - RS485 DRIVER (OPTION)
   - RS422 DRIVER (OPTION)
10. 8 CHANNEL 8 BITS ANALOG TO DIGITAL (A/D) CONVERTER

11. 38 GENERAL PURPOSE I/O PIN
    - 16 BIDIRECTIONAL INPUT/OUTPUT PIN
    - 11 INPUT ONLY PIN
    - 11 OUTPUT ONLY PIN
12. DOTMATRIX LCD CONNECTOR INTERFACE
13. 34 PIN I/O BUS (ETT STANDARD) INTERFACE
14. SOFTWARE FOR DOWLOAD & PROGRAM CPU & SECURITY PROTECT
15. POWER SUPPLY WITH 7805 REGULATOR ON BOARD
16. PCB SIZE 5.5 CM X 9 CM
17. LOW POWER (กินกระแส 15 มิลิแอมป์ ที่แรงดัน + 5 โวลต์ ในขณะที่ไม่มีโหลด)
18. 2 MODE SELECT (DOWN LOAD AND RUN)

2. โครงสร้างและการทำงานของบอร์ด CP-JR6811

บอร์ด CP-JR6811 เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ถูกพัฒนาขึ้นโดยการรวบรวมเอาคุณสมบัติต่างๆที่จำเป็นไว้ภายในบอร์ดเดียว ซึ่งมีขนาดเล็กกระทัดรัดสามารถนำมาใช้ทดแทนวงจรอิเล็กทรอนิกส์ต่างๆได้อย่างมีประสิทธิภาพ มีความยืดหยุ่นในการใช้งานสามารถทำการปรับปรุงแก้ไขเปลี่ยนแปลงการทำงานได้ไม่ยากนัก โดยมีส่วนประกอบโครงสร้างภายนอกดังภาพประกอบที่ 20 และมีโครงสร้างภายในและขาสัญญาณต่างๆดังภาพประกอบที่ 21 สำหรับโครงสร้างภายในและขาสัญญาณต่างของซีพียู 68SEC811E2 สามารถดูได้จากภาพประกอบที่ 22

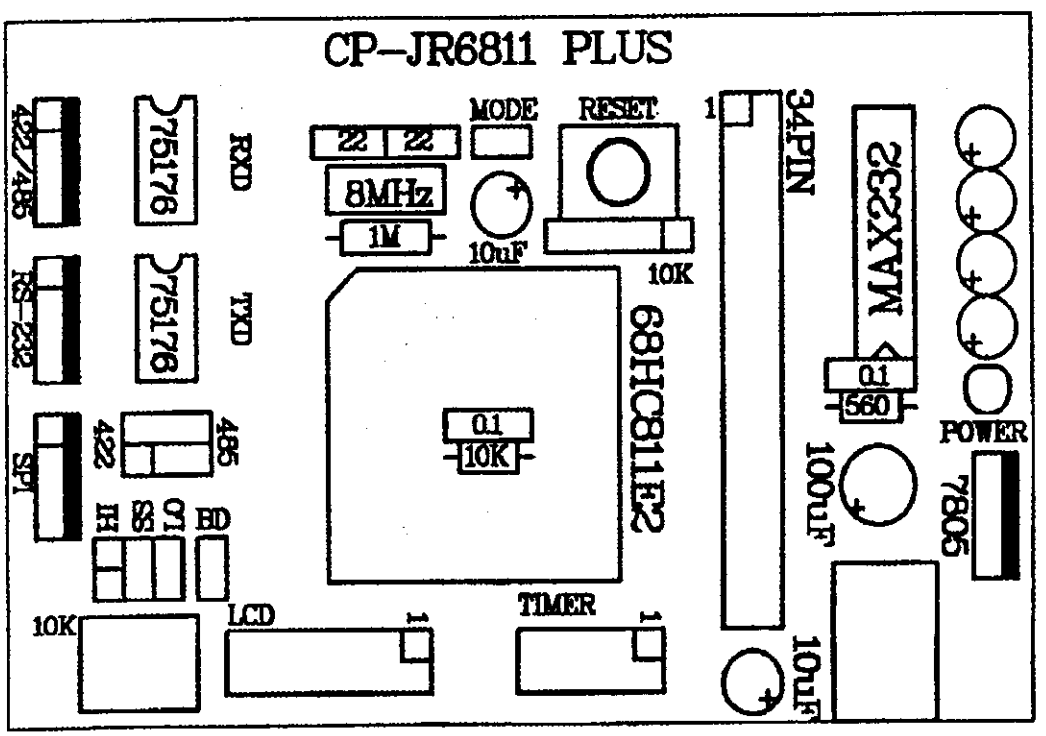
การทำงานของบอร์ดสามารถกำหนดได้ 2 โหมดคือ

1. โหมดดาวน์โหลด(DOWNLOAD MODE)ข้อมูลและเขียนข้อมูลใส่ในตัวซีพียู

เมื่อซีพียูถูกกำหนดให้ทำงานในโหมดนี้แล้ว หลังจากสิ้นสุดการรีเซ็ต ซีพียูจะรอรับข้อมูลจากพอร์ตสื่อสารอนุกรมและนำข้อมูลนั้นเขียนเข้าไปภายในตัวซีพียู หลังจากนั้นก็จะกระโดดไปทำงานตามคำสั่งของโปรแกรมที่ดาวน์โหลดเข้าไปโดยอัตโนมัติ การเลือกโหมดนี้ทำได้โดยการ SHORT JP-MODE แล้วกดสวิตช์รีเซ็ต จากนั้นจึงโหลดโปรแกรมที่ต้องการใช้งานลงในซีพียูโดยการเลือกโหมดจะมีผลสมบูรณ์ก็ต่อเมื่อเราทำการรีเซ็ตบอร์ดแล้วเท่านั้น การเลือกโหมดหลังการรีเซตแล้วจะไม่มีผลต่อการเปลี่ยนแปลงโหมดการทำงานแต่อย่างใด

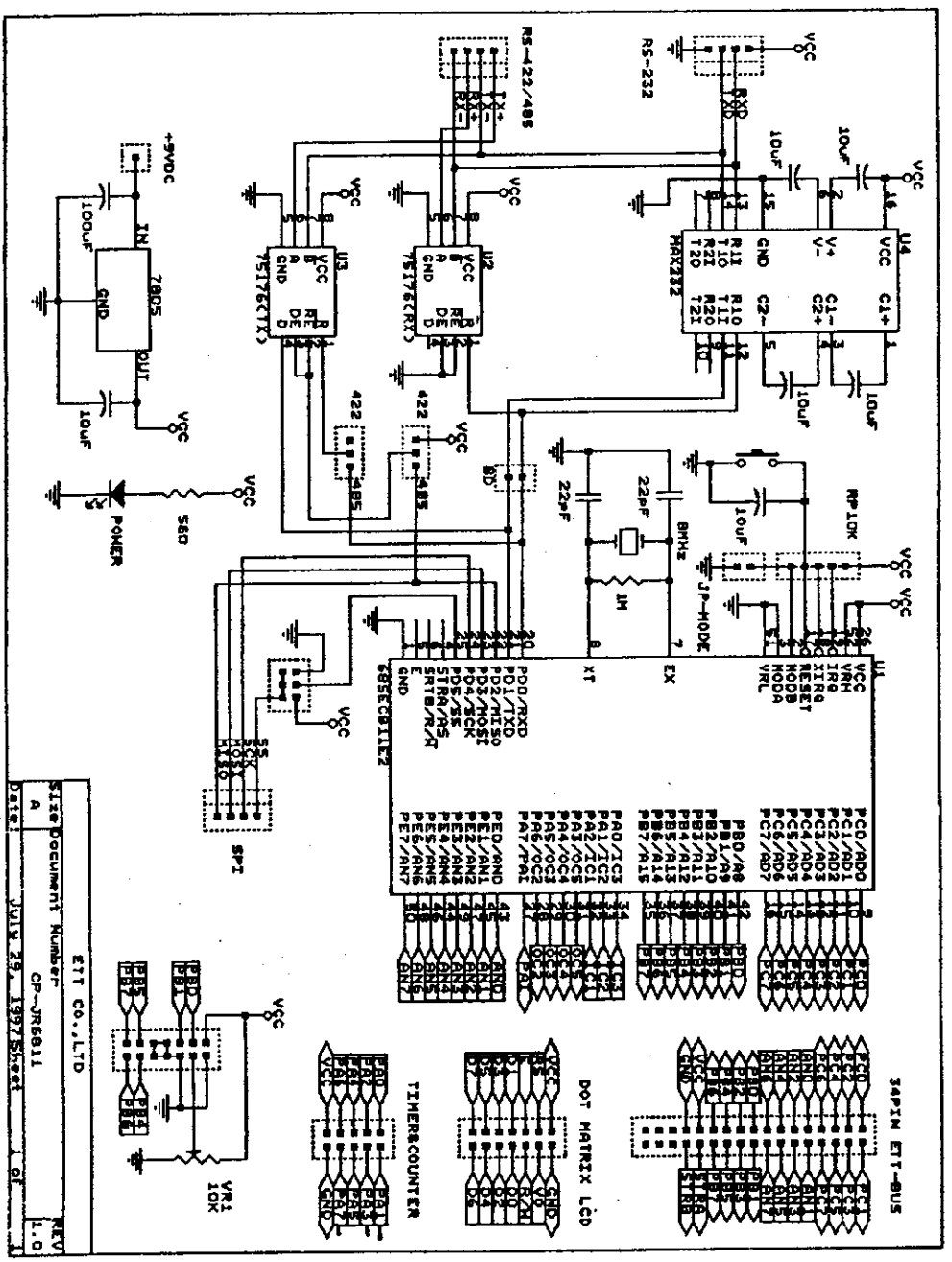
2. โหมดรัน(RUN MODE)

เมื่อซีพียูถูกกำหนดให้ทำงานในโหมดนี้ หลังจากรีเซ็ตแล้วซีพียูจะกระโดดไปทำงานตามคำสั่งของโปรแกรมที่เราทำการดาวน์โหลดไว้แล้วทันที การเลือกโหมดนี้ทำได้โดยการ OPEN JUMPER JP-MODE ออก แล้วกดสวิตซ์รีเซ็ตที่บอร์ดหลังจากรีเซ็ตแล้วโปรแกรมที่ดาวน์โหลดลงในซีพียูก็จะเริ่มทำงานตามที่ได้โปรแกรมไว้



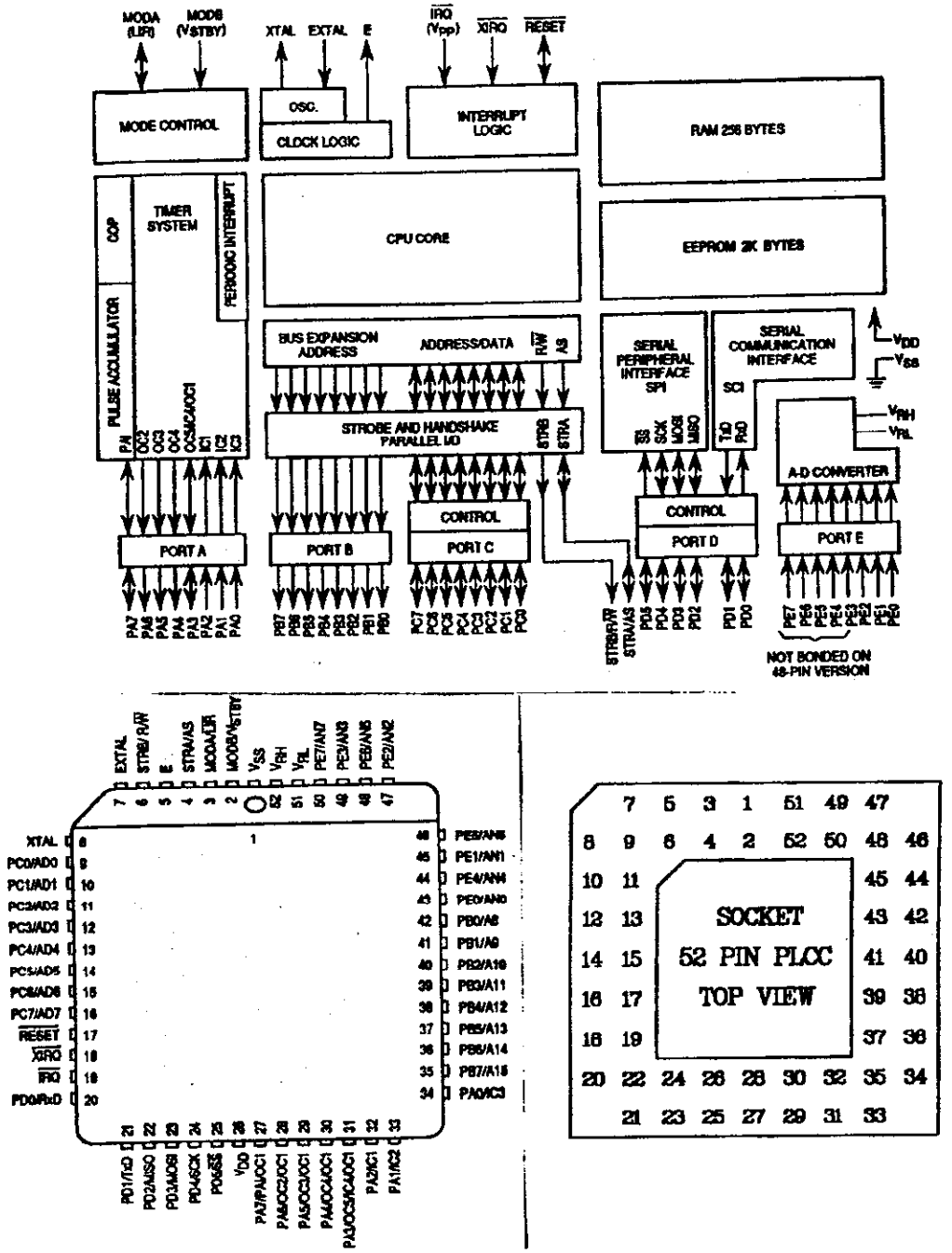
ภาพประกอบ 20 แสดงโครงสร้างของ บอร์ด CP-JR6811

(ที่มา: CP-JR6811 USER'S MANUAL, 2539)



ภาพประกอบ 21 แสดงโครงสร้างภายในและขาสัญญาณต่างๆ ของบอร์ด CP-JR6811

(ที่มา: CP-JR6811 USER'S MANUAL, 2539)



ภาพประกอบ 22 แสดงโครงสร้างภายในและขาสัญญาณต่างๆ ของ ซีพียู 68SEC811E2

(ที่มา: CP-JR6811 USER'S MANUAL, 2539)

3. รีจิสเตอร์ต่างๆ ของ บอร์ด CP-JR6811 ในส่วนที่เกี่ยวข้องกับตัวตั้งเวลาโปรแกรมได้

3.1 รีจิสเตอร์ตัวนับเวลา (Timer Counter register : TCNT) เป็นรีจิสเตอร์ขนาด 16 บิต มีแอดเดรสอยู่ที่ $100E-$100F ใช้เก็บข้อมูลที่ได้จากการนับของตัวนับ สามารถเลือกความถี่ของสัญญาณนาฬิกาได้ 4 ระดับจากปรีสเกลเลอร์คือ หารด้วย 1, 4, 8, หรือ 16 โดยกำหนดสถานะของบิต PR1 และ PR0 ในรีจิสเตอร์ TMSK2 ดังแสดงในตาราง 3

ตาราง 3 การกำหนดสถานที่บิต PR0 และ PR1 เพื่อกำหนดตัวหาร

PR1	PR0	ตัวหาร
0	0	1
0	1	4
1	0	8
1	1	16

(ที่มา: เรียนรู้และใช้งานไมโครคอนโทรลเลอร์68HC11)

ในการอ่านค่าของการนับสามารถอ่านได้โดยใช้ซอฟร์แวร์ เนื่องจากจังหวะของการอ่านค่าจะมีเฟสตรงข้ามกับสัญญาณนาฬิกา E ทำให้ไม่มีผลต่อการทำงานของตัวนับ

3.2 รีจิสเตอร์ควบคุม

ประกอบด้วย รีจิสเตอร์ TCTL1 , TCTL2, TMSK1 และ TMSK2 ซึ่งมีรายละเอียดของความหมายในแต่ละบิตดังนี้

3.2.1 รีจิสเตอร์ควบคุมตัวตั้งเวลา 1 (Timer Control register 1 : TCTL1)

เป็นรีจิสเตอร์ขนาด 8 บิต อยู่ที่แอดเดรส $1020 ใช้ในการควบคุมการทำงานของส่วนเอาต์พุตเปรียบเทียบความหมาย และหน้าที่ในแต่ละบิตมีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5

OM2, OM3, OM4, OM5 เป็นบิตกำหนดโหมดเอาต์พุต

OL2, OL3, OL4,0L5 เป็นบิตกำหนดระดับเอาต์พุต

3.2.2 รีจิสเตอร์ควบคุมตัวตั้งเวลา 2 (Timer Control register 2 : TCTL2)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ $1021 ใช้ควบคุมขอบขาของสัญญาณที่ต้องการตรวจจับที่อินพุตของส่วนอินพุตแคปเจอร์ ความหมายของบิตต่างๆมีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A

ซึ่งหลังจากซีพียูถูกรีเซตในครั้งแรกรีจิสเตอร์เหล่านี้จะถูกเคลียร์ให้มีค่าเป็นศูนย์ทั้งหมด โดยรีจิสเตอร์ TCTL2 จะถูกแบ่งออกเป็น 4 คู่เพื่อควบคุมวงจรอินพุต 4 วงจร โดยรหัสการตรวจสอบสัญญาณอินพุตจะแสดงไว้ดังตารางที่ 4

ตาราง 4 แสดงรหัสการตรวจสอบสัญญาณอินพุต

EDGXB	EDGXA	ลักษณะการทำงาน
0	0	ไม่มีการทำงานในส่วนของอินพุต
O	1	ตรวจสอบสัญญาณที่ช่วงขอบขาขึ้น
1	0	ตรวจสอบสัญญาณที่ช่วงขอบขาลง
1	1	ตรวจสอบสัญญาณในทุกๆขอบขาของสัญญาณ

(ที่มา: เรียนรู้และใช้งานไมโครคอนโทรลเลอร์68HC11)

3.2.3 รีจิสเตอร์อินเตอร์รัปต์มาส์กของตัวตั้งเวลา 1 (Timer Interrupt Mask register 1 : TMSK1)

เป็นรีจิสเตอร์ 8 บิต มีแอดเดรสอยู่ที่ $1022 ใช้ควบคุมการอีนาเบิลการอินเตอร์รัปต์ของอินพุตแคปเจอร์ และส่วนเปรียบเทียบเอาต์พุต ความหมายของแต่ละบิตมีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
OC1I	OC2I	OC3I	OC4I	OC5I	IC1I	IC2I	IC3I

OCxI (Output Compare x Interrupt) บิตนี้จะเซตเมื่อแฟล็ก OCxF ถูกเซต เมื่อบิตนี้เซตแสดงว่ามีการร้องขอการอินเตอร์รัปต์ ณ ที่บิตของเอาต์พุตนั้นๆ

ICxI (Input Compare x Interrupt) บิตนี้จะเซตเมื่อแฟล็ก ICxF ถูกเซต เมื่อบิตนี้เซตแสดงว่ามีการร้องขอการอินเตอร์รัปต์ ณ ที่บิตของเอาต์พุตนั้นๆ

3.2.4 รีจิสเตอร์อินเตอร์รัปต์มาส์กของตัวตั้งเวลา 2 (Timer Interrupt Mask register 2 : TMSK2)

รีจิสเตอร์ที่ใช้ควบคุมลำดับของการร้องขอการอินเตอร์รัปต์ทางฮาร์ดแวร์ที่มาจากสถานะของบิตในรีจิสเตอร์ TFLG2 และใน 2 บิตต่ำ (คือ บิต 0 และ บิต 1)ยังใช้บิตที่ทำหน้าที่กำหนดการหารสัญญาณนาฬิกา E หรือเรียกว่าปรีสเกลเลอร์ ความหมายแต่ละบิตมีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
TOI	RTII	PAOVI	PAII	0	0	PR1	PR2

TOI (Timer Overflow Interrupt Enable) เมื่อบิตนี้เป็น "0" ส่วนอินเตอร์รัปต์ของ TOF จะถูกดิสเอเบิล ถ้าบิตนี้เป็น "1" จะเกิดการร้องขอการอินเตอร์รัปต์ ถ้าบิต TOF เป็น "1" ด้วย

RTII (RTI Interrupt Enable) เมื่อบิตนี้เป็น "0" การอินเตอร์รัปต์ของ RTIF จะถูกดิสเอเบล ถ้าเป็น "1" จะเกิดการร้องขอการอินเตอร์รัปต์ ถ้าหากบิต RTIF เป็น "1" ด้วย

PAOVI (Pulse Accumulator Overflow Interrupt Enable) ถ้าเป็น "0" การอินเตอร์รัปต์ของ PAOVF ในรีจิสเตอร์ TFLG2 จะถูกดิสเอเบิล แต่ถ้าหากเป็น "1" จะเกิดการร้องขอการอินเตอร์รัปต์ ถ้าหากบิต PAOVF เป็น "1" ด้วย

PAII (Pulse Accumulator Input Interrupt Enable) ถ้าเป็น "0" การอินเตอร์รัปต์ของ PAIF จะถูกดิสเอเบิล แต่ถ้าเป็น "1" จะเกิดการร้องขอการอินเตอร์รัปต์เมื่อบิต PAIF ถูกเซต

บิต 3 และ บิต 2 ไม่ใช้งานกำหนดให้เป็น "0"

PR 1 และ PR 2 (Timer Prescaler Selects) ทั้ง 2 บิตนี้ใช้ในการกำหนดตัวหารสัญญาณนาฬิกา สามารถที่จะอ่านข้อมูลหรือสถานะของบิตนี้ได้ตลอดเวลาแต่ในการเขียนข้อมูลมาที่ 2 บิตนี้ จะเขียนได้ในจังหวะที่เป็นการอินิเชียลเท่านั้น เพราะการเขียนข้อมูลมาที่ 2 บิต จะถูกดิสเอเบิลไว้หลังจากเขียนข้อมูลในตัวแรกไปแล้ว หรือหลังจากที่รีเซตไปแล้วนาน 64 ไซเกิลของสัญญาณนาฬิกา E

3.3 รีจิสเตอร์แสดงสถานะ

ประกอบด้วย รีจีสเตอร์ TFLG1 และ TFLG2 ดังมีรายละเอียดดังต่อไปนี้

3.3.1 รีจิสเตอร์อินเตอร์รัปต์แฟล็กของตัวตั้งเวลา 1 (Timer Interrupt Flag register 1 : TFLG1)

รีจิสเตอร์ TFLG1 ใช้ในการแสดงเหตุการณ์ที่เกิดขึ้นในระบบตั้งเวลานี้ และทำงานร่วมกับรีจิสเตอร์ TMSK1 เพื่อก่อให้เกิดการอินเตอร์รัปต์ขึ้นในระบบ แต่ละบิตของรีจิสเตอร์ TFLG1 จะทำงานสัมพันธ์สอดคล้องกับบิตต่างๆ ในรีจิสเตอร์ TMSK1 ในตำแหน่งเดียวกัน ถ้าบิตในรีจิสเตอร์ TMSK1 เซตแล้วในช่วงเวลานั้นพบว่า แฟล็กหรือบิตที่สัมพันธ์กัน ณ ตำแหน่งเดียวกันนั้นเซตอยู่เหมือนกัน ก็จะทำให้เกิดการร้องขอการอินเตอร์รัปต์ขึ้นทันที ความหมายของแต่ละบิตมีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
OC1F	OC2F	OC3F	OC4F	OC5F	IC1F	IC2F	IC3F

OCxF (Output Compare x Flag) บิตแฟล็กนี้จะถูกเซตเมื่อตัวนับค่าเวลามีค่าตรงกับค่าในรีจิสเตอร์เปรียบเทียบเอาต์พุต การเขียนข้อมูล "0" มาที่บิตนี้ไม่สามารถจะเคลียร์บิตนี้ได้ การเคลียร์ทำได้โดยการเขียนข้อมูล "1" มายังบิตนี้เท่านั้น

ICxF (Input Capture x Flag) บิตแฟล็กนี้จะเซตเมื่อสามารถตรวจจับขอบขาของสัญญาณอินพุตที่ต้องการได้ในอินพุต ICx การเคลียร์แฟล็กนี้ทำได้โดยการเขียนข้อมูล "1" มาที่บิตนี้ ถ้าเขียนข้อมูล "0" มาบิตนี้จะไม่เคลียร์

3.3.2 รีจิสเตอร์อินเตอร์รัปต์แฟล็กของตัวตั้งเวลา 2 (Timer Interrupt Flag register 2 : TFLG2)

รีจิสเตอร์ TFLG2 ใช้ในการแสดงเหตุการณ์ที่เกิดขึ้นในระบบตั้งเวลานี้ และทำงานร่วมกับรีจิสเตอร์ TMSK2 เพื่อก่อให้เกิดการอินเตอร์รัปต์ขึ้นในระบบ แต่ละบิตของรีจิสเตอร์ TFLG2 จะทำงานสัมพันธ์สอดคล้องกับบิตต่างๆ ในรีจิสเตอร์ TMSK2 ในตำแหน่งเดียวกันถ้าหากบิตอินาเบิลในรีจิสเตอร์ TMSK2 ถูกเซต แล้วตรวจสอบบิตแฟล็กพบว่าเซตอยู่เช่นเดียวกันก็จะเกิดการร้องขอการอินเตอร์รัปต์ขึ้น ความหมายของแต่ละบิตมีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
TOF	RTIF	PAOVF	PAIF	0	0	0	0

TOF (Timer Overflow) บิตนี้สามารถเคลียร์ได้โดยการรีเซต บิตนี้จะเซตเมื่อตัวนับอิสระ 16 บิต ทำการนับครบรอบแล้วเปลี่ยนค่าจาก $FFFF เป็น $0000 การเคลียร์บิตนี้ทำได้โดยการเขียนข้อมูล "1" มายังบิตนี้

RTIF (Real time Interrupt Flag) บิตนี้จะเซตในทุกจังหวะของขอบขาขึ้นของสัญญาณ ณ จุดที่ต้องการตรวจจับ บิตนี้จะเคลียร์โดยการเขียนข้อมูล "1" มายังบิตนี้

PAOVF (Pulse Accumulater Overflow Interrupt Flag) บิตนี้จะเซตเมื่อตัวนับในพัลส์แอกคิวมูเลเตอร์เปลี่ยนค่าจาก $FF เป็น $00

PAIF (Pulse Accumulater Input Edge Interupt Flag) บิตนี้จะเซตเมื่อสามารถตรวจจับขอบขาสัญญาณที่ต้องการได้ที่ขา PAI

บิต 3 – บิต 0 ไม่ได้ใช้งานกำหนดให้เป็น "0" ทั้งหมด

3.4 รีจิสเตอร์เก็บข้อมูลในระบบตั้งเวลา

ประกอบด้วยรีจิสเตอร์ขนาด 16 บิต 9 ตัวคือ รีจิสเตอร์ TCNT TIC1-TIC3 และ TOC1-TOC5 โดยรีจิสเตอร์ TCNT เก็บข้อมูลที่ต้องการนับของตัวนับอิสระขนาด 16 บิต รีจิสเตอร์ TIC1-TIC3 มีขนาด 16 บิต จะทำหน้าที่เก็บข้อมูลของอินพุตแคปเจอร์มีแอดเดรสอยู่ที่ $1010-$1015 ส่วนรีจิสเตอร์ TOC1-TOC5 เก็บข้อมูลของส่วนเปรียบเทียบเอาต์พุตมีขนาด 16 บิต และแอดเดรสอยู่ที่ $1016-$101F

ส่วนอินพุตแคปเจอร์(Input capture)

หน้าที่ของส่วนนี้คือ ทำการบันทึกหรือเก็บค่าเมื่ออินพุตเกิดการเปลี่ยนแปลงตรงตามเงื่อนไขที่กำหนดไว้ของระบบ โดยตรวจสอบจากขอบขาของสัญญาณอินพุต ซึ่งสามารถตรวจจับได้ทั้งขอบขาขึ้น ขอบขาลง หรือใช้ตรวจจับทั้งสองขอบขา จากความสามารถนี้เองทำให้สามารถนำไปใช้วัดระยะเวลาระหว่างสัญญาณพัลส์ได้โดยใช้ซอฟต์แวร์มาช่วยในการคำนวณ ก็จะสามารถนำระบบไมโครคอนโทรลเลอร์นี้จับคาบเวลา ความถี่ หรือปริมาณอื่นๆที่สัมพันธ์กับช่วงเวลา เช่น วัดความเร็ว ตรวจจับการเคลื่อนที่ และระยะของการเคลื่อนที่ โดยค่าของช่วงเวลาจะถูกเก็บไว้ในรีจิสเตอร์ TNCT