สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เทคโนโลยีสารสนเทศในการแสดงผลข้อมูลผลการผลิตแบบ On Time

On Time Production Counter Using Information Technology

โดย

นาย อรรถพล รุ่งถาวรวงศ์

นาย เฉลิมวงศ์ โชติชัย

เลขหมู่.....
เลขทะเบียน 72836
วัน.เดือน,ปี 23 มิ.ย. 2550

b 11224311x
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทคโนโลยีสารสนเทศในการแสดงผลข้อมูลผลการผลิตแบบ On Time

On Time Production Counter Using Information Technology

โดย

นาย อรรถพล รุ่งถาวรวงศ์ รหัส 47015235
นาย เฉลิมวงศ์ โชติชัย รหัส 47015241

อาจารย์ที่ปรึกษา

ดร. กิติพล ชิตสกุล

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เทคโนโลยีสารสนเทศในการแสดงผลข้อมูลผลการผลิตแบบ On Time

On Time Production Counter Using Information Technology

ผู้จัดทำ

1. นาย อรรถพล รุ่งถาวรวงศ์ รหัส47015235
2. นาย เฉลิมวงศ์ โชติชัย รหัส47015241

.....อาจารย์ที่ปรึกษา

(ดร. กิตินล ชิตสกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทคโนโลยีสารสนเทศในการแสดงผลข้อมูลผลการผลิตแบบ On Time

นาย อรรถพล รุ่งถาวรวงศ์ รหัส 47015235
นาย เฉลิมวงศ์ โชติชัย รหัส 47015241
ดร. กิติพล ชิตสกุล อาจารย์ที่ปรึกษา
ปีการศึกษา 2549

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ได้อธิบายถึงการออกแบบ และ การสร้างเครื่องนับจำนวนผลการผลิต ซึ่งประกอบด้วยป้ายนับจำนวนผลการผลิต และ ป้ายแสดงเป้าหมายการผลิต ซึ่งป้ายนับจำนวนผลการผลิตจะแสดงจำนวนผลการผลิต ของแต่ละสายการผลิต (ในกรณีนี้มี 2 สายการผลิต) ส่วนป้ายแสดงเป้าหมายการผลิต มีการกำหนดเป้าหมายในการผลิตโดยการใช้คีย์แพ็ดเป็นตัวตั้งค่าเป้าหมาย การแสดงผลที่ป้ายแสดงเป้าหมายการผลิตจะเป็นการแสดงผลแบบนับลง โดยการเขียนโปรแกรมบนไมโครคอนโทรเลอร์ ให้ป้ายแสดงเป้าหมายการผลิตนับลง ตามผลรวมของจำนวนผลการผลิตทั้งสองสายการผลิตมารวมกัน นอกจากนี้ป้ายนับจำนวนการผลิตยังสามารถเชื่อมต่อไปยังไมโคร-คอมพิวเตอร์ผ่านทางพอร์ตอนุกรมเพื่อตั้งค่าเป้าหมายการผลิต และแสดงจำนวนการผลิตในแต่ละสายการผลิตและจำนวนโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

On Time Production Counter Using Information Technology

Mr. Attaphol Roongthawornwong ID 47015235
Mr. Chalermwong Chotichai ID 47015241
Dr. Kitiphol Chitsakul Advisor
Academic year 2006

Abstract

This thesis describes a design and implementation of "on time production counter". The project consists of two parts, production counter display boards and a target display board. The production counter display boards shows number of products from production lines (there is two production lines for this case). We can set production target up the target display board via a keypad. The target board shows the production target from the both production lines and counts down as production increases. All of the boards are controlled by microcontrollers. Moreover production counter display boards can interface to a microcomputer via serial ports for each line setting up and displaying number of products. The total products can also display on the monitor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการทำรายงานเล่มนี้ประสบความสำเร็จได้ด้วยดีเพราะได้รับคำปรึกษาและข้อเสนอแนะจากอาจารย์ที่ปรึกษา ดร. กิติพล ชิตสกุล ขอบคุณ คุณพินิต เตชะธัญญกุล ที่ให้คำปรึกษาและข้อมูลของโรงงาน ขอบคุณรุ่นพี่ปริญญาโททุก ๆ ท่าน และเพื่อน ๆ ที่ให้คำแนะนำ และ เอื้อเฟื้ออุปกรณ์การทดลอง และขอขอบพระคุณ คุณพ่อคุณแม่ที่คอยเป็นกำลังใจให้ตลอดมา

อรรถพล รุ่งถาวรวงศ์
เฉลิมวงศ์ โชติชัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 แสดงบอร์ด ET-DSP4 3

รูปที่ 2.2 แสดงโครงสร้างภายในของ LED 7-segment 4

รูปที่ 2.3 แสดงการต่อวงจรควบคุม LED 7-segment แบบรับกระแสเข้า 5

รูปที่ 2.4 แสดงการต่อวงจรควบคุม LED 7-segment แบบรับกระแสออก 5

รูปที่ 2.5 ตู้เติมลมอัตโนมัติ 7

รูปที่ 2.6 แสดงโครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51 9

รูปที่ 2.7 แสดงขาต่างๆของไมโครคอนโทรลเลอร์ MCS-51 10

รูปที่ 2.8 แสดงขาของไมโครคอนโทรลเลอร์ MSC-51 ที่ใช้ต่อกับ XTAL 13

รูปที่ 2.9 แสดงสัญญาณนาฬิกาและแมชชีนไซเคิล 13

รูปที่ 2.10 แสดงการต่อพอร์ตเข้ากับระบบบัสภายในของไมโครคอนโทรลเลอร์ MSC-51 14

รูปที่ 2.11 แสดงโครงสร้างของพอร์ตทั้ง 4 พอร์ตของไมโครคอนโทรลเลอร์ MSC-51 15

รูปที่ 2.12 แสดงการจัดหน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51 16

รูปที่ 2.13 แสดงขั้นตอนต่าง ๆ ในการอ่านข้อมูล 17

รูปที่ 2.14 แสดงโครงสร้างโดยรวมของไทเมอร์/เคาน์เตอร์ 21

รูปที่ 2.15 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 0 26

รูปที่ 2.16 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 1 27

รูปที่ 2.17 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 2 28

รูปที่ 2.18 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0 โหมด 3 29

รูปที่ 2.19 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 16 บิต รีโหลดค่าแบบอัตโนมัติ 36

รูปที่ 2.20 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 16 บิต รีโหลดค่าแบบอัตโนมัติ 36

รูปที่ 2.21 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2 โหมด 16 บิต แคปเจอร์ 37

รูปที่ 2.22 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมดกำหนดอัตราเร็วในการรับส่งข้อมูลข้อพอร์ตอนุกรม 38

รูปที่ 2.23 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมดการสร้างสัญญาณพัลส์ดิวตี้ไซเคิล 50% 39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่ 2.24 แสดงลักษณะของคีย์สวิตช์แบบเมทริกซ์ขนาด 4×3 41
รูปที่ 3.1 แสดงการสร้างโปรเจ็ค Counter 44
รูปที่ 3.2 แสดงการสร้างไฟล์ Counter.c 45
รูปที่ 3.3 แสดงไฟล์ Counter.c ที่ทำการเพิ่มไฟล์เข้าไปยังโปรเจ็ค 45
รูปที่ 3.4 แสดงการเขียนโปรแกรม 46
รูปที่ 3.5 แสดงการสร้างไฟล์นามสกุล.hex 46
รูปที่ 4.1 แสดงการแสดงผลของป้ายนับจำนวนผลการผลิต และ LCD 47
รูปที่ 4.2 แสดงผลของการใส่รหัสผ่าน 47
รูปที่ 4.3 แสดงผลของการกำหนดเป้าหมายในการผลิต 47
รูปที่ 4.4 แสดงการทำงานของป้ายนับจำนวนผลการผลิต 48
รูปที่ 4.5 แสดงหน้าต่างหลักของโปรแกรม On Time Production Counter
Using Information Technology 49
รูปที่ 4.6 กำหนดค่าเป้าหมายใน LINE1 และการแสดงผลของป้ายนับจำนวนการผลิต 50
รูปที่ 4.7 การนับค่าใน LINE1 และการแสดงผลของป้ายนับจำนวนการผลิต
เมื่อกดสวิตช์ที่อยู่บนชุดจำลองการทำงานของรีเลย์ 51
รูปที่ 4.8 กำหนดค่าเป้าหมายใน LINE2 และการแสดงผลของป้ายนับจำนวนการผลิต 52
รูปที่ 4.9 การนับค่าใน LINE2 และการแสดงผลของป้ายนับจำนวนการผลิต
เมื่อกดสวิตช์ที่อยู่บนชุดจำลองการทำงานของรีเลย์ 53
รูปที่ 4.10 กำหนดค่าเป้าหมายทั้งใน LINE1และLINE2
และการแสดงผลที่ป้ายนับจำนวนการผลิต 54
รูปที่ 4.11 การนับค่าใน LINE1 และLINE2 และการแสดงผลของป้ายนับจำนวนการผลิต
เมื่อกดสวิตช์ที่อยู่บนชุดจำลองการทำงานของรีเลย์ 54
รูปที่ 4.12 การกำหนดเป้าหมายใน LINE1และLINE2
และการแสดงผลของป้ายนับจำนวนการผลิต 56
รูปที่ 4.13 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิต
เมื่อจำลองให้เกิดไฟฟ้าดับที่ LINE1 56
รูปที่ 4.14 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อทำการส่งข้อมูลกลับ 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.15 แสดงผลบนจอมอนิเตอร์ และ ป้ายนับจำนวนการผลิต
เมื่อจำลองให้เกิดไฟฟ้าดับที่LINE2 57
รูปที่ 4.16 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อทำการส่งข้อมูลกลับ 58
รูปที่ 4.17 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิต
เมื่อจำลองให้ไฟฟ้าดับทั้งสองLINE 58
รูปที่ 4.18 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อทำการส่งข้อมูลกลับ 59
รูปที่ 4.19 แสดงการนับค่าใน LINE1 และ LINE2 และ การแสดงผล
ของป้ายนับจำนวนการผลิต หลังจากมีการส่งข้อมูลกลับ 59
รูปที่ 4.20 แสดงการยืนยันการออกจากโปรแกรม 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงตำแหน่งขาข้อมูลที่ต่อกับพอร์ต	6
ตารางที่ 2.2 แสดงข้อมูลที่ส่งให้ LED 7-segment แสดงเป็นเลขและตัวอักษรต่าง ๆ	6
ตารางที่ 2.3 แสดงรายละเอียดต่าง ๆ ของไมโครคอนโทรลเลอร์ MCS-51	10
ตารางที่ 2.4 แสดงหน้าที่พิเศษของพอร์ต 3	11
ตารางที่ 2.5 แสดงการกำหนดค่าให้ไทเบอร์/เคาน์เตอร์ 0 เป็นไทเบอร์	30
ตารางที่ 2.6 แสดงการกำหนดค่าให้ไทเบอร์/เคาน์เตอร์ 0 เป็นเคาน์เตอร์	31
ตารางที่ 2.7 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 1 เป็นไทเมอร์	31
ตารางที่ 2.8 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 1 เป็นเคาน์เตอร์	32
ตารางที่ 2.9 แสดงการกำหนดโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2	35
ตารางที่ 2.10 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ เป็นไทเมอร์ 2	40
ตารางที่ 2.11 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 2 เป็นเคาน์เตอร์	40
ตารางที่ 2.12 แสดงการเลือกรูปแบบการทำงานของไทเมอร์/เคาน์เตอร์ 2	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

เนื่องจากช่วงปิดเทอมภาคฤดูร้อน ปีการศึกษา 2548 นายอรรถพล รุ่งถาวรวงศ์ ได้ไปทำการฝึกงานที่บริษัท เดโชโมลด์แอนด์ไดคาสติ้ง จำกัด (Decho Mould & Die-Casting Co.,Ltd.) ตั้งอยู่เลขที่ 99 หมู่ที่ 8 ถนนวัดศรี-อ่อนนุช ตำบลบางโฉลง อำเภอบางพลี จังหวัดสมุทรปราการ ซึ่งดำเนินธุรกิจเกี่ยวกับการผลิตชิ้นส่วนรถจักรยานยนต์ และ เครื่องยนต์ ในการผลิตจะมีอยู่หลายสายการผลิต ซึ่งแบ่งออกเป็นส่วน ๆ เช่น สายการผลิตดุมล้อ (hub) รถจักรยานยนต์, สายการผลิตซี่ลวด (spoke) และ ส่วนของสายการประกอบวงล้อรถจักรยานยนต์ (มี 2 สายการผลิต) แต่ละวันบริษัท จะตั้งเป้าหมายจำนวนวงล้อรถจักรยานยนต์ ที่ต้องประกอบไว้ พบว่ามีปัญหาเกิดขึ้นในกระบวนการผลิต คือ การประกอบวงล้อที่เสร็จพร้อมส่งมอบให้ลูกค้าเกิดความล่าช้า ซึ่งยากต่อการตรวจสอบและควบคุม

ดังนั้นบริษัทจึงได้มีโครงการที่จะติดตั้งระบบสารสนเทศในการแสดงผลข้อมูลผลการผลิตแบบ On Time ซึ่งประกอบด้วยระบบนับจำนวนวงล้อที่ประกอบเสร็จแล้วในแต่ละสายการผลิต จากนั้นนำข้อมูลที่ได้มารวมกัน ซึ่งจะอาศัยไมโครคอนโทรเลอร์ในการประมวลผลพร้อมกับให้มีการแสดงผลที่ป้ายนับจำนวนการผลิต(ประกอบด้วยป้ายแสดงเป้าหมายการผลิตและป้ายนับจำนวนผลการผลิตของแต่ละสายการผลิต) แสดงให้พนักงานเห็นความคืบหน้าของการผลิตและเป็นการตรวจสอบว่าสายการผลิตใดผลิตได้ล่าช้าเพื่อทางบริษัทจะได้กระตุ้นพนักงานและตรวจสอบต่อไป

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อออกแบบและสร้างเครื่องนับจำนวนผลการผลิตแบบ On Time
2. เพื่อศึกษาผลของสัญญาณรบกวนในโรงงาน ที่มีต่อระบบเทคโนโลยีสารสนเทศในการแสดงผลข้อมูลการผลิตแบบ On Time และศึกษาแนวทางแก้ไข
3. เพื่อส่งเสริมความร่วมมือกันระหว่างสถาบันการศึกษา และ ภาคอุตสาหกรรม ในด้านวิชาการและการพัฒนาทางด้านเทคโนโลยีของภาคอุตสาหกรรม

1.3 แนวทางการทำโครงงาน

1. ออกแบบและสร้างเครื่องนับจำนวนผลการผลิต การเชื่อมต่อกับเครื่องมือในโรงงาน
2. พัฒนาซอฟต์แวร์ด้วยไมโครคอนโทรลเลอร์ในการประมวลผลที่ได้จากการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. พัฒนาชุดแสดงผลเพื่อควบคุมเป้าหมายการผลิตในสายการผลิตในแต่ละวัน
4. พัฒนาซอฟต์แวร์เชื่อมต่อเป็นระบบสารสนเทศในการแสดงผลข้อมูลการผลิตแบบ On Time
5. ทดสอบ แก้ไข ระบบและรวบรวมผลเพื่อเขียนรายงาน

1.4 ผลที่คาดว่าจะได้รับ

1. พัฒนาความสามารถของนักศึกษาในด้านการออกแบบ และ สร้างระบบเทคโนโลยีสารสนเทศใน การเพิ่มผลผลิตในสภาพที่เป็นจริงของโรงงาน
2. ช่วยให้โรงงานมีระบบ ตรวจสอบและกำหนดเป้าหมายการผลิตที่ชัดเจน
3. นักศึกษามีประสบการจริงในการวิเคราะห์และแก้ไขปัญหาต่างๆ ในการทำโครงงาน

1.5 โครงสร้างของรายงาน

รายงานนี้เป็นการนำเสนอผลงานที่ได้ปฏิบัติ เพื่อออกแบบป้ายแสดงจำนวนผลการผลิตในอุตสาหกรรมล้อรถจักรยานยนต์ โดยแบ่งนำเสนอเป็นบทตอนดังนี้

บทที่ 1 บทนำกล่าวถึงความเป็นมาของโครงงาน วัตถุประสงค์ และผลที่คาดว่าจะได้รับ

บทที่ 2 กล่าวถึงทฤษฎีที่เกี่ยวข้องในโครงงานซึ่งประกอบด้วยหัวข้อต่าง ๆ คือ ป้ายแสดงการผลิต, เครื่องจักรในอุตสาหกรรมการผลิตล้อจักยานยนต์, ไมโครคอนโทรลเลอร์ *MCS-51*,การต่อคีย์สวิตช์แบบเมทริกซ์หรือคีย์แพดกับไมโครคอนโทรลเลอร์

บทที่ 3 เป็นส่วนของการออกแบบและการสร้าง

บทที่ 4 กล่าวถึงการทดลองและผลการทดลอง ซึ่งแบ่งออกเป็น 2 ส่วน คือ
การทดลองและผลการทดลองในส่วนของการตั้งเป้าหมายทางคีย์แพ็ด
และการทดลองและผลการทดลองในส่วนของการติดต่อกับคอมพิวเตอร์

และบทที่ 5 บทสรุป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ป้ายแสดงการผลิต

ป้ายแสดงการผลิตเป็นป้ายที่ใช้ในการแสดงจำนวนผลการผลิต ซึ่งในที่นี้ได้ใช้บอร์ดแสดงผล ET-DSP4 ของบริษัท อีทีที จำกัด ในการแสดงผลการนับจำนวนการผลิต

2.1.1 บอร์ดแสดงผล ET-DSP4

บอร์ด ET-DSP4 โครงสร้างทางด้านฮาร์ดแวร์ที่มีตัวแสดงผลแบบ 7-Segment ขนาดใหญ่ (4.7*7 ซม.) จำนวน 4 หลักซึ่งสามารถแสดงผลได้ชัดเจนและมองเห็นได้ในระยะไกลๆ โดยในส่วนของวงจรควบคุมการแสดงผลของหน้าจอนั้นจะใช้ วงจร Latch แบบ 8-Bit Serial Shift To Parallel เบอร์ 74HC595 ต่ออนุกรมกันจำนวน 4 ตัว จึงทำให้การเขียนโปรแกรมเพื่อควบคุมการแสดงผลเป็นไปได้โดยง่ายไม่ซับซ้อน และใช้สัญญาณในการเชื่อมต่อน้อย

รูปที่ 2.1 แสดงบอร์ด ET-DSP4

การควบคุมการแสดงผล จะมีวงจรภาคแสดงผลโดยใช้ 74HC595 สำหรับควบคุมการแสดงผลของ 7-Segment โดยมีสัญญาณในการควบคุมการทำงานทั้งหมด 3 เส้นดังนี้คือ

- Serial-In เป็นสัญญาณข้อมูลเข้าที่ส่งสัญญาณควบคุมมาจาก CPU โดยเริ่มส่งจากข้อมูลที่มีนัยสำคัญสูงสุด ก่อนแล้วจำตามด้วยข้อมูลบิตอื่น ๆ ตามลำดับ ซึ่งสัญญาณข้อมูลแต่ละบิตจะถูกเลื่อนเข้าสู่ ชิพรีจิสเตอร์ของ 74HC595 โดยการควบคุมของสัญญาณนาฬิกา Shift Clock

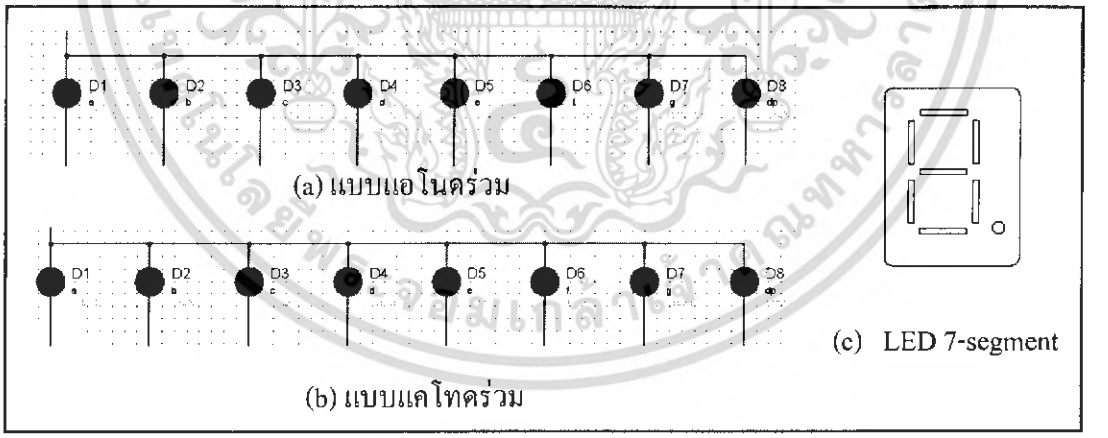
- Shift Clock เป็นสัญญาณนาฬิกาที่ส่งมาจาก CPU ใช้สำหรับเลื่อนบิตข้อมูลจากขา Serial-In เข้าสู่ชิพรีจิสเตอร์ภายในของ 74HC595 โดยข้อมูลจะถูกเลื่อนเข้าไปภายในชิพรีจิสเตอร์ในทุกๆ ช่วงขอบขาขึ้น (Rising Edge) ของสัญญาณนาฬิกา Shift Clock นี้เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Latch Clock เป็นสัญญาณ Latch ที่ส่งมาจาก CPU เพื่อสั่ง Latch ข้อมูลที่ถูกเลื่อนเข้ามาไว้ภายในชิพรีจิสเตอร์เรียบร้อยแล้วออกไปยังขาสัญญาณ เอาต์พุท โดยข้อมูลจะถูก Latch ในช่วงของสัญญาณ Latch Clock เป็นขอบขาขึ้น (Rising Edge)

2.1.2 LED 7-segment

LED 7-segment จะเกิดการนำหลอด LED จำนวน 7 หลอดมาต่อเรียงกันเพื่อให้แสดงผลเป็นตัวเลขได้และสามารถแสดงตัวอักษรภาษาอังกฤษได้บางตัว โครงสร้างของ LED 7-segment คือการนำเอาตัว LED แต่ละหลอดหรือแต่ละเซกเมนต์ทั้ง 7 หลอดต่อกัน เพื่อให้สามารถแสดงเป็นตัวเลขและตัวอักษรได้ โดย LED 7-segment 1 ชุดประกอบด้วย LED จำนวน 7 หลอดซึ่งมีชื่อประจำคือ a, b, c, d, c, f และg แต่ในปัจจุบันได้เพิ่มมาเป็น 8 ส่วน คือส่วนของจุดหรือ dp แต่ยังคงเรียกกันเป็น LED 7-segment เหมือนเดิม ถ้าหลอด LEDโดยเอาแอโนดมาต่อร่วมกันจะเรียกว่าแบบแอโนดร่วม (common anode seven-segment) ถ้าเป็นแบบแคโทดต่อร่วมกันจะเรียกว่าแบบแคโทดร่วม (common cathode seven-segment) ดังแสดงในรูปที่ 2.2 ถ้าหากต้องการให้หลอด สว่างเป็นเลขใด ก็ทำได้โดยให้หลอดแต่ละหลอดสว่างประกอบกันให้เป็นเลขนั้น ถ้าหากทุกหลอดสว่างพร้อมกันจะแสดงเป็นเลข 8

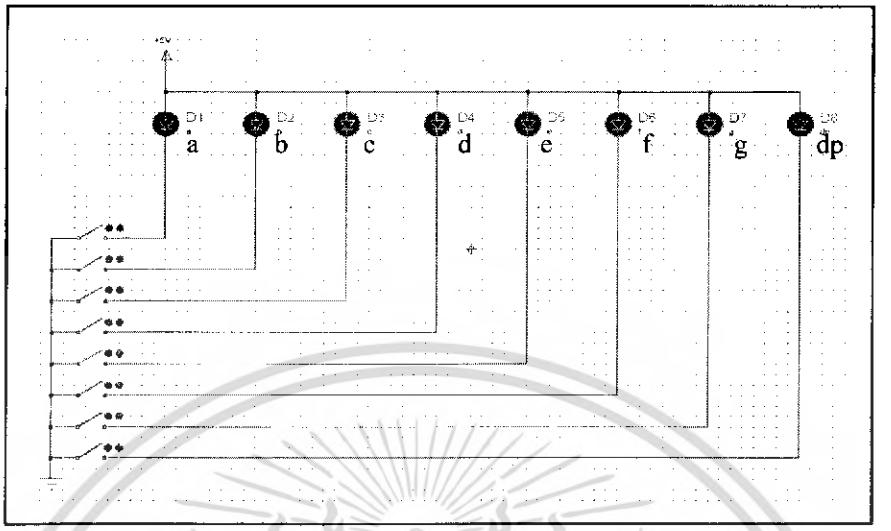


รูปที่ 2.2 แสดงโครงสร้างภายในของ LED 7-segment

การควบคุมการรับกระแสเข้า (Current Sink)

LED 7-segment ที่จะนำมาใช้กับการควบคุมแบบรับกระแสเข้า จะต้องเป็น LED 7-segment ที่มีโครงสร้างภายในลักษณะแบบแอโนดร่วม โดยจะต้องต่อขาร่วมกับไฟบวก และควบคุมการทำงานด้วยไฟลบหรือกราวด์จากขาข้อมูล (a, b, c, d, e, f ,g และ dp) ดังรูปที่ 2.3

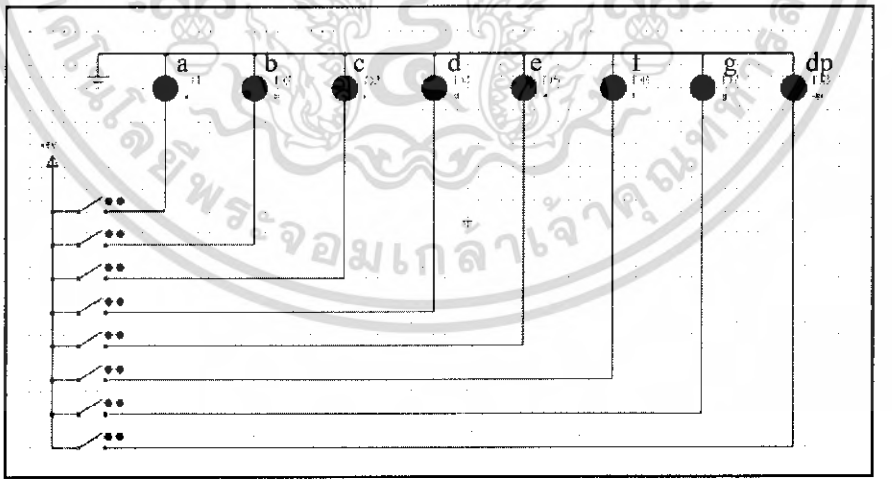
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงการต่อวงจรควบคุม LED 7-segment แบบรับกระแสเข้า

การควบคุมการรับกระแสออก (Current Source)

LED 7-segment ที่จะนำมาใช้กับการควบคุมแบบรับกระแสเข้า จะต้องเป็น LED 7-segment ที่มีโครงสร้างภายในลักษณะแบบแคโทดร่วม โดยจะต้องต่อขาร่วมกราวด์หรือไฟและควบคุมการทำงานด้วยไฟบวกที่ขาข้อมูล ดังรูปที่ 2.4



รูปที่ 2.4 แสดงการต่อวงจรควบคุม LED 7-segment แบบรับกระแสออก

ถ้าหากนำ LED 7-segment ไปต่อกับพอร์ตโดยให้บิตต่ำสุดต่อกับขาข้อมูล a ของ LED 7-segment บิตต่อไปต่อกับขาข้อมูล b เรียงไปเรื่อย ๆ จนถึงขาข้อมูล g ดังแสดงในตารางที่ 2.1 ถ้าหากต้องการแสดงผลเป็นเลขต่าง ๆ จะส่งข้อมูลออกไปยังพอร์ตดังตารางที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งบิต	ตำแหน่งขาข้อมูล
7	dp
6	g
5	f
4	e
3	d
2	c
1	b
0	a

ตารางที่ 2.1 แสดงตำแหน่งขาข้อมูลที่ต่อกับพอร์ต

แสดงผล	แอโนดร่วม	แคโทดร่วม
0	C0H	3FH
1	F9H	06H
2	A4H	5BH
3	B0H	4FH
4	99H	66H
5	92H	6DH
6	82H	7DH
7	F8H	07H
8	80H	1FH
9	98H	6FH
A	88H	77H
b	83H	7CH
C	C6H	39H
d	A1H	5EH
E	86H	79H
F	8EH	71H

ตารางที่ 2.2 แสดงข้อมูลที่ส่งให้ LED 7-segment แสดงเป็นเลขและตัวอักษรต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 เครื่องจักรในอุตสาหกรรมการผลิตล้อจักรยานยนต์

เครื่องจักรในอุตสาหกรรมการผลิตล้อจักรยานยนต์จะมีอยู่หลายส่วนแต่ในส่วนที่เกี่ยวข้องกับป้ายแสดงจำนวนผลการผลิตนั้นจะเป็นส่วนของเครื่องเติมลมอัตโนมัติ ซึ่งมีหลักการทำงานดังนี้

เมื่อทำการเปิดเครื่องที่เบรกเกอร์โดยดันไปที่ตำแหน่ง ON แล้วนั้น จะปรากฏไฟโชว์ที่ตัวเครื่อง โดยสีส้มและสีเขียวจะสว่าง หลังจากนั้นให้นำสายเติมลมไปต่อเข้ากับล้อจักรยานยนต์โดยดันให้แน่นเพื่อป้องกันการรั่วไหลของลม เพราะจะทำให้เกิดค่าที่ผิดเพี้ยนจากค่าที่ตั้งไป หลังจากนั้นให้ทำการเหยียบสวิทช์เท้า 1 ครั้งและปล่อยเท้าทันที หรือ กดสวิทช์มือที่ตัวเครื่องที่มีคำว่า start เครื่องจะทำการเติมลมทันทีโดยไฟสีส้มและสีเขียวจะดับและไฟสีแดงจะสว่าง โดยกระบวนการแรกจะเติมลมเพื่อขึ้นขอบยางตามเวลาที่เราได้ตั้งค่าไว้ที่ timer และเมื่อครบระยะเวลาดังกล่าวแล้วนั้นเครื่องจะทำการต่อลมในยางทิ้งจนถึงค่าที่ตั้งไว้และเติมไปที่ค่าที่เราต้องการทันที กระบวนการจะสิ้นสุดการทำงานโดยจะมีรีเลย์เป็นตัวตัดการทำงานของเครื่องไฟสีแดงจะดับและไฟสีส้มและสีเขียวจะติดพร้อมกันให้ทำการนำสายเติมลมออกทันที

รูปที่ 2.5 ตู้เติมลมอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 เป็นอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่งที่มีการรวมเอาหน่วยประมวลผลหรือซีพียู, หน่วยคำนวณทางคณิตศาสตร์และลอจิก, วงจรรับสัญญาณอินพุต, วงจรขับสัญญาณเอาต์พุต, หน่วยคำนวณ และวงจรกำเนิดสัญญาณนาฬิกาไว้ด้วยกันทำให้สามารถนำไปใช้งานแทนวงจรอีเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี ช่วยลดจำนวนอุปกรณ์, ความซับซ้อน และขนาดของระบบ ในขณะที่มีขีดความสามารถสูงขึ้น ภายใต้งบประมาณที่เหมาะสมในบางครั้งจึงอาจเรียกไมโครคอนโทรลเลอร์ว่า single chip microcomputer หรือไมโครคอนโทรลเลอร์ชิพเดี่ยวก็ได้ลักษณะงานที่เหมาะสมกับการนำไมโครคอนโทรลเลอร์ไปใช้งาน มักจะเป็นงานประยุกต์ที่เกี่ยวข้องกับการควบคุม หรือจัดการสัญญาณอินพุต/เอาต์พุตของวงจรอิเล็กทรอนิกส์และวงจรดิจิตอลต่างๆ เช่น ระบบแสดงผล หรือระบบเตือนภัย ระบบควบคุมภายในเครื่องใช้ไฟฟ้า เป็นต้น

2.3.1 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 ได้จัดให้มีส่วนประกอบภายในเพื่ออำนวยความสะดวกแก่ผู้ใช้ ใช้เทคโนโลยีในการผลิตแบบ NMOS และ PMOS ซึ่งภายในได้รวมวงจรต่าง ๆ ไว้อย่างครบถ้วนพร้อมทีจะใช้งานเมื่อจ่ายไฟเลี้ยง และ สัญญาณนาฬิกา ไมโครคอนโทรลเลอร์ MCS-51 ได้ถูกผลิตออกมาหลายเบอร์โดยบริษัทต่าง ๆ เช่น Atmel, Philips, Dalla และบริษัทอื่น ๆ ซึ่งไม่ว่าเบอร์อะไรก็ตามโครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51 จะคล้ายกันจะแตกต่างกันออกไปในส่วนของความสามารพิเศษในแต่ละเบอร์ ซึ่งโครงสร้างภายในขอไมโครคอนโทรลเลอร์ MCS-51 สามารถแสดงได้ในรูปที่ 2.6

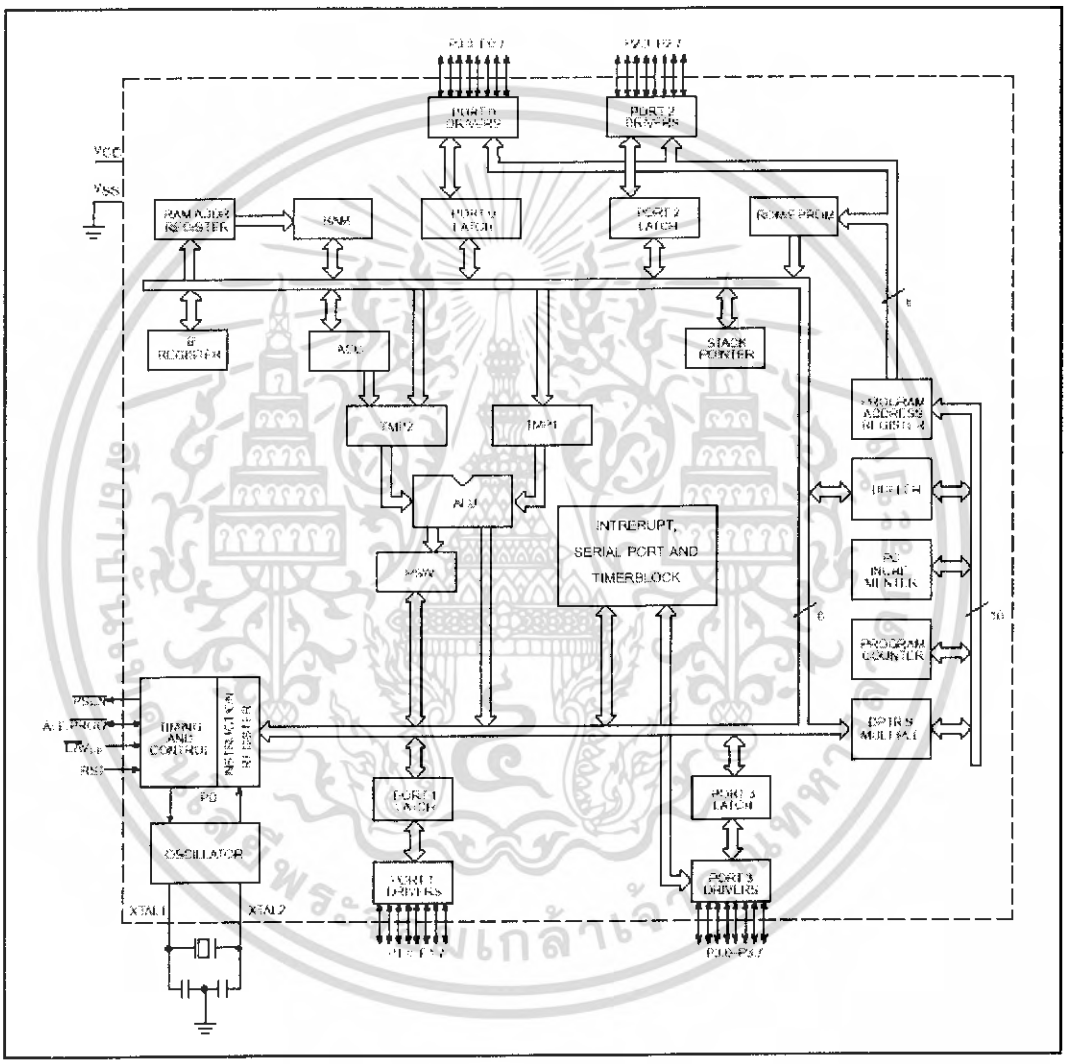
2.3.2 คุณสมบัติของไมโครคอนโทรลเลอร์ MCS-51

คุณสมบัติที่สำคัญของไมโครคอนโทรลเลอร์ MCS-51 มีดังนี้

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลช ซึ่งสามารลบและเขียนใหม่นับพันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทางสามารถใช้งานได้ทั้งอินพุตและเอาต์พุต
- มีพอร์ตอนุกรมที่สามารถรับส่งข้อมูลแบบฟูลดูเพล็กซ์ (Full Duplex)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีชุดTimer/Counter ขนาด 16 บิตอย่างน้อย 2 ชุด
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัพท์ได้ 6 แหล่ง
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ในภายในชิพ



รูปที่ 2.6 แสดงโครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

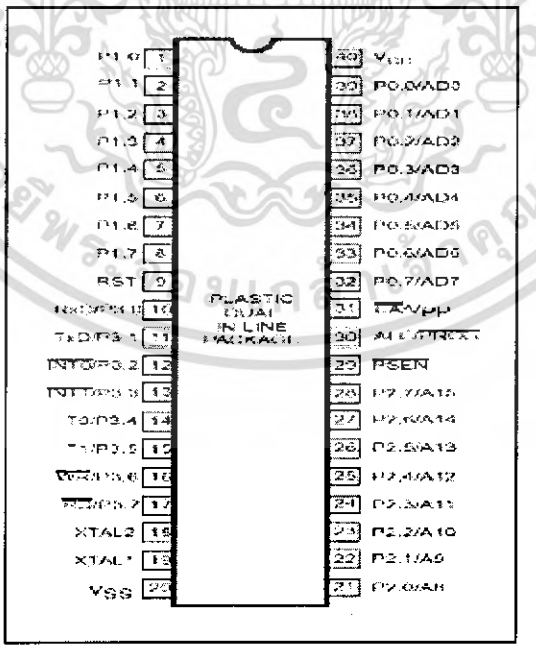
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 การจัดขาต่างๆของไมโครคอนโทรลเลอร์ MCS-51

ไอซี ไมโครคอนโทรลเลอร์ 8051 โครงสร้างไอซีเป็นแบบ DIP มีขาทั้งหมด 40 ขา โดยขาต่าง ๆ จะใช้เป็นขาพอร์ตอินพุต, เอาท์พุต, ขาสัญญาณควบคุม, ขาตำแหน่งหน่วยความจำ และขาข้อมูลดังรูปที่ 2.7

เบอร์	หน่วยความจำโปรแกรม	หน่วยความจำข้อมูล	Timer/Counter 16 บิต
8031	-	128 Bytes	2
8051	4K ROM	128 Bytes	2
8052	8K ROM	256 Bytes	3
AT89C51	4K FLASH	128 Bytes	2
AT89C52	8K FLASH	256 Bytes	3
AT89C1051	1K FLASH	64 Bytes	1
AT89C2051	2K FLASH	128 Bytes	2

ตารางที่ 2.3 แสดงรายละเอียดต่าง ๆ ของไมโครคอนโทรลเลอร์ MCS-51



รูปที่ 2.7 แสดงขาต่างๆของไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายของขาต่างๆมีดังนี้

พอร์ต 0 (P0.0-P0.7) ได้แก่ขาที่ 32-39 ของ MSC-51 เป็นพอร์ตแบบ 2 ทิศทางขนาด 8 บิต แต่ละขาสามารถใช้ทำหน้าที่ได้ทั้งเป็นพอร์ตอินพุตและเอาต์พุตทั่วไป สำหรับ Port 0 บิตที่ถูกส่งค่า '1' ออกมา จะทำให้ขานั้นเป็น Float ซึ่งสามารถใช้งานเป็น High impedance Input ได้นอกจากนี้ยังทำหน้าที่เป็น Address Bus ส่วนของ Low byte (PCL) สลับกับ Data Bus ในกรณีที่มีการใช้งาน External Memory ในการนี้ Logic '1' จะถูกสร้างจาก Strong Internal pull-up และท้ายสุด Port 0 จะใช้ส่งค่าที่มีอยู่ใน Internal program memory ออกมาเพื่อการตรวจสอบ (Program verification) แต่จะต้องต่อ External pull-up ให้ด้วย

พอร์ต 1 (P1.0-P1.7) ได้แก่ขาที่ 1-8 ของ MSC-51 เป็นพอร์ตแบบ 2 ทิศทางขนาด 8 บิต แต่ละขาสามารถใช้ทำหน้าที่ได้ทั้งเป็นพอร์ตอินพุต และ เอาต์พุตทั่วไปและสามารถอ้างทีละบิตได้ คือ P1.0, P1.1,...,P1.7 สำหรับ Port 1 บิต ที่ถูกส่งค่า '1' ออกมา จะทำให้ขานั้นเป็น High จาก Internal pull-up ซึ่งสามารถใช้งานเป็น Input ได้ และ เมื่อถูกป้อนด้วย Logic '0' นั้นมันจะให้กระแสออกมาได้ซึ่งกระแสจะไหลมาจาก Internal pull-upนั่นเอง

พอร์ต 2 (P2.0-P2.7) ได้แก่ขาที่ 21-28 ของ MSC-51 เป็นพอร์ตแบบ 2 ทิศทางขนาด 8 บิต แต่ละขาสามารถใช้ทำหน้าที่ได้ทั้งเป็นพอร์ตอินพุตและเอาต์พุตทั่วไป และ ใช้เป็นพอร์ตสำหรับติดต่อกับหน่วยความจำภายนอก คือทำหน้าที่เป็น Address Bus ส่วนของ High byte (PCH)

ขาพอร์ต	ชื่อ	หน้าที่พิเศษ
P3.0	RxD	ใช้รับข้อมูลทางพอร์ตอนุกรม
P3.1	TxD	ใช้ส่งข้อมูลทางพอร์ตอนุกรม
P3.2	$\overline{\text{INT0}}$	อินเตอร์รัพต์ภายนอก 0
P3.3	$\overline{\text{INT1}}$	อินเตอร์รัพต์ภายนอก 1
P3.4	T0	ตัวจับเวลา / ตัวนับ 0
P3.5	T1	ตัวจับเวลา / ตัวนับ 1
P3.6	$\overline{\text{WR}}$	สัญญาณเขียนข้อมูลหน่วยความจำภายนอก
P3.7	$\overline{\text{RD}}$	สัญญาณอ่านข้อมูลหน่วยความจำภายนอก

ตารางที่ 2.4 แสดงหน้าที่พิเศษของพอร์ต 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต 3 (P3.0-P3.7) ได้แก่ขาที่ 10-17 ของ MSC-51 เป็นพอร์ตแบบ 2 ทิศทางขนาด 8 บิต แต่ละขาสามารถใช้ทำหน้าที่ได้ทั้งเป็นพอร์ตอินพุตและเอาต์พุตทั่วไป และ มีหน้าที่พิเศษดังที่แสดงในตารางที่ 2.4 ดังนั้นเมื่อมีการใช้สัญญาณดังกล่าวจะทำให้การทำงานของ MSC-51 ผิดพลาดได้ ดังนั้นจึงไม่ควรเขียนข้อมูลไปที่พอร์ต 3

PSEN (Program Store Enable) ได้แก่ขาที่ 29 ของ MSC-51 เป็นขาที่ส่งสัญญาณออก ขานี้จะแอคทีฟเมื่อ MCS-51 ต้องการอ่าน Code โปรแกรมภายนอกโดยปกติถ้าหน่วยความจำภายนอกเป็น EPROM ขา PSEN จะต่อกับขา Output Enable (OE) ของ EPROM

ALE (Address Latch Enable)/PROG ได้แก่ขาที่ 30 ของ MSC-51 เป็นขาที่ใช้ในการควบคุมการแลตช์ของพอร์ต 0 เมื่อมีการใช้งานในการใช้หน่วยความจำภายนอก นอกจากนี้ขานี้ยังเป็นขาสำหรับรับพัลส์ของโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MSC-51 ในรุ่นที่มีหน่วยความจำแบบ EPROM

EA (External Access)/Vpp ได้แก่ขาที่ 31 ของ MSC-51 ใช้สำหรับเลือกการติดต่อหน่วย-ความจำโปรแกรมภายนอก หรือภายในตัวไมโครคอนโทรลเลอร์ MSC-51 ถ้าเป็นลอจิก "0" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ MSC-51 ติดต่อกับหน่วยความจำโปรแกรมภายนอก แต่ถ้าเป็นลอจิก '1' เป็นการเลือกให้ไมโครคอนโทรลเลอร์ MSC-51 ติดต่อกับหน่วยความจำโปรแกรมภายในนอกจากนี้ขานี้ ยังสามารถเป็นขาอินพุตสำหรับรับแรงดันไฟสูง สำหรับการโปรแกรมหน่วย-ความจำภายใน สำหรับไมโครคอนโทรลเลอร์ MSC-51 แบบแฟลช ต้องการแรงดันสำหรับการโปรแกรม +12 V

RST (Reset) ขา RST ได้แก่ขาที่ 9 ของ MSC-51 จะใช้ในการรีเซตการทำงานของ MCS-51 โดยจะให้ขานี้เป็นลอจิก '1' อย่างน้อย 2 Machine Cycles จึงจะรีเซตระบบได้

VCC ได้แก่ขาที่ 40 ของ MSC-51 ใช้สำหรับต่อไฟเลี้ยง + 5V

GND (VSS) ได้แก่ขาที่20 ของ MSC-51 เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

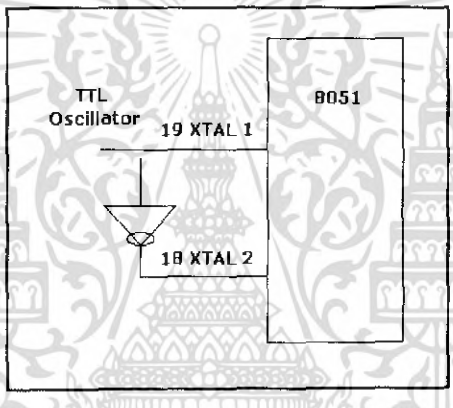
XTAL1 ได้แก่ขาที่ 18 ของMSC-51 เป็นขาอินพุตของวงจรออสซิลเลเตอร์แอมพลิไฟเออร์ และยังเป็นอินพุตของวงจรกำเนิดสัญญาณนาฬิกาภายใน

XTAL2 ได้แก่ขาที่19 ของMSC-51เป็นขาเอาต์พุตของวงจรออสซิลเลเตอร์แอมพลิไฟเออร์

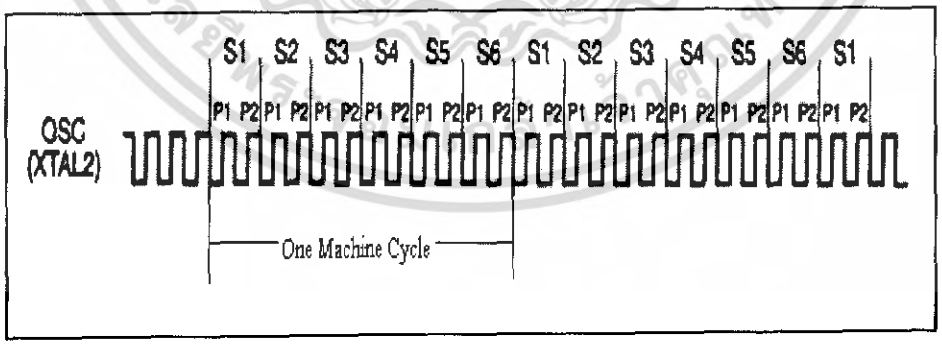
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 ฐานเวลาและความถี่สัญญาณนาฬิกาของไมโครคอนโทรลเลอร์ MCS-51

ตามที่ทราบมาแล้วว่าระบบคอมพิวเตอร์จะทำงานได้จะต้องมีฐานเวลาและสัญญาณนาฬิกาเป็นตัวควบคุมการทำงานสำหรับไมโครคอนโทรลเลอร์ MSC-51 จะมีวงจรออสซิลเลเตอร์อยู่ภายในการกำหนดจังหวะเวลาทำได้โดยต่อ Crystal เข้ากับขา XTAL1 และ XTAL2 โดยปกติมักจะใช้ Crystal ความถี่ 12 MHz กับตัวเก็บประจุหรืออาจใช้สัญญาณนาฬิกาจาก TTL Clock Source ต่อกับ XTAL1 และ XTAL2 ดังรูปที่ 2.8 เพื่อสร้างสัญญาณนาฬิกาในการกำหนดการทำงานของไมโครคอนโทรลเลอร์ MSC-51



รูปที่ 2.8 แสดงขาของไมโครคอนโทรลเลอร์ MSC-51 ที่ใช้ต่อกับ XTAL



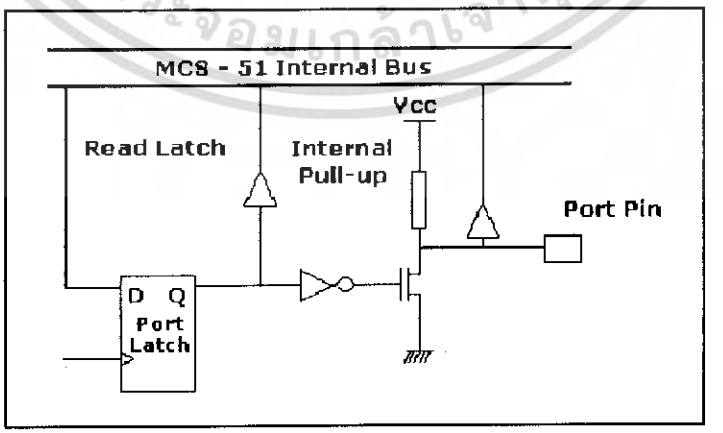
รูปที่ 2.9 แสดงสัญญาณนาฬิกา และ แมชชีนไซเคิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5 โครงสร้างและการทำงานของพอร์ต

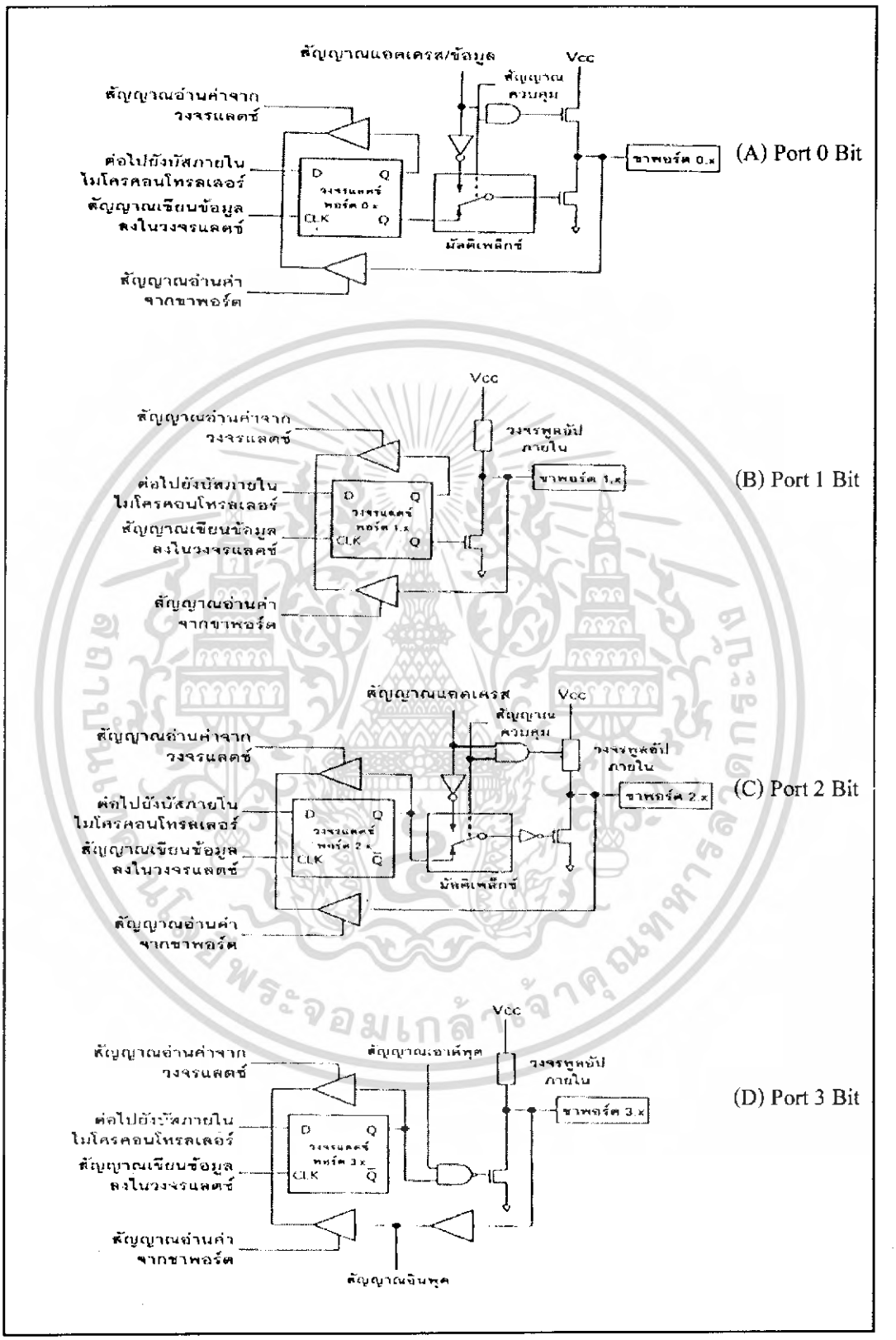
ไมโครคอนโทรลเลอร์ MSC-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต คือ พอร์ต 0 ถึงพอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง คือ สามารถเป็นได้ทั้งอินพุตสำหรับรับสัญญาณข้อมูลเข้าและเอาต์พุตสำหรับส่งสัญญาณข้อมูลออกทุกพอร์ตขอไมโครคอนโทรลเลอร์ MSC-51 แบบแฟลช จะมีวงจรแลตช์ และ วงจรขับตลอดจนบัฟเฟอร์อินพุตขาของพอร์ตจะแสดงโครงสร้างภายในได้ดังรูปที่ 2.11 โดยจะมีโครงสร้างเป็น Field-effect Transistor ต่ออยู่กับขาภายนอก และ มีความต้านทานต่อ Full-up อยู่สำหรับพอร์ต 1, 2 และ3 แต่ถ้าเป็นพอร์ต 0 จะไม่มีความต้านทานพูลอัพภายใน เพราะว่าไม่ได้ใช้เป็นขา Address Bus และ Data Bus จากรูปที่ 2.11 จะเห็นว่าพอร์ต 0 ไม่มีความต้านทานต่อ Full-up ในชิพ ดังนั้นถ้าหากต้องการใช้พอร์ตนี้เป็นพอร์ตอินพุต จะต้องต่อความต้านทานพูลอัพ ภายนอกให้กับพอร์ตนี้ทุกบิตด้วย และจากโครงสร้างของพอร์ต 0 จะเห็นว่ามีวงจรมัลติเพล็กอยู่ภายในเพื่อใช้ในการเลือกว่าจะใช้เป็นพอร์ต หรือ ใช้ในการติดต่อกับหน่วยความจำภายนอก เช่นเดียวกับพอร์ต 2 แต่พอร์ต 2 จะมีความต้านทานพูลอัพอยู่ภายในชิพถ้าจะต่อใช้งานเป็นพอร์ตอินพุตสามารถใช้ได้ทันที สำหรับพอร์ต 3 จะมีโครงสร้างคล้ายกับพอร์ต 1 แต่จะมีสัญญาณอินพุตเอาต์พุตพิเศษเพิ่มเข้ามา เนื่องจากพอร์ตนี้สามารถใช้งานเป็นหน้าที่พิเศษแต่ละบิตได้

พอร์ตนี้สามารถใช้เป็นอินพุตเอาต์พุตกับอุปกรณ์ภายนอกได้ ในการอ่านข้อมูลจากพอร์ตจะอ่านได้ 2 แบบ คือ Read Latch และ Read Pin โดย Read Latchหมายถึงการอ่านข้อมูลที่ถูก Latch เอาไว้เข้าสู่บัสภายในของ MCS-51 เช่นการทำคำสั่ง CPL P1.5 แต่ถ้าเป็นการ Read Pin จะเป็น การใช้พอร์ตเป็นอินพุต โดยการอ่านค่าจากขาของไอซีเข้าสู่บัสภายในโดยการอ่านแบบ Read Latch และ Read Pin จะมีสัญญาณมาควบคุมที่บัฟเฟอร์ดังรูปที่ 2.10



รูปที่ 2.10 แสดงการต่อพอร์ตเข้ากับระบบบัสภายในของไมโครคอนโทรลเลอร์ MSC-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

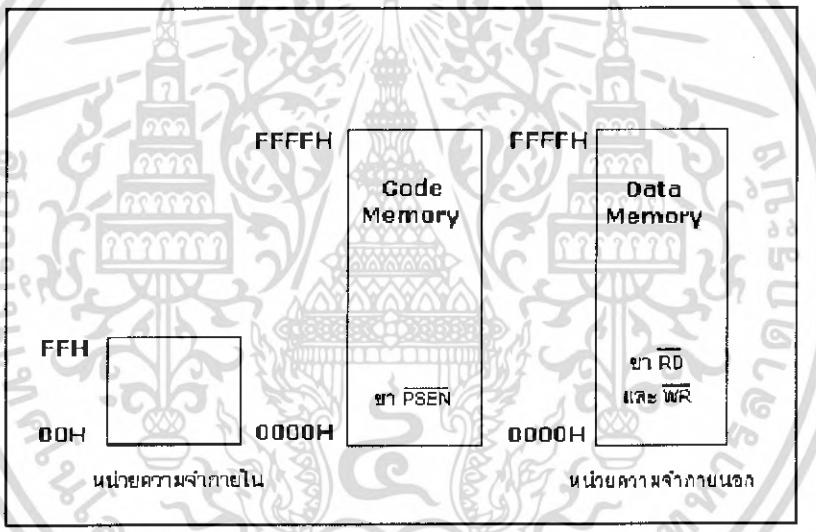


รูปที่ 2.11 แสดงโครงสร้างของพอร์ตทั้ง 4 พอร์ตของไมโครคอนโทรลเลอร์ MSC-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6 โครงสร้างหน่วยความจำ

หน่วยความจำสำหรับไมโครคอนโทรลเลอร์ MCS-51 จะมี 2 ชนิด คือ หน่วยความจำที่ใช้โปรแกรม (ROM) และ หน่วยความจำที่ใช้เก็บข้อมูลในการประมวลผล (RAM) ไมโครคอนโทรลเลอร์ MCS-51 บางเบอร์ เช่น 8051, 8052 จะมีหน่วยความจำโปรแกรมภายในชิพ และไมโครคอนโทรลเลอร์ MCS-51 ทุกเบอร์สามารถอ้างหน่วยความจำโปรแกรมภายนอกได้มากที่สุด 64 K และอ้างหน่วยความจำข้อมูลภายนอกได้มากที่สุด 64 K สำหรับหน่วยความจำ RAM ภายในจะประกอบไปด้วยพื้นที่ใช้งานทั่วไป, รีจีสเตอร์แบงค์, พื้นที่ใช้งานระดับบิต และ รีจีสเตอร์ฟังก์ชันพิเศษเราอาจเขียนไดอะแกรมของหน่วยความจำของ 8031 ได้ดังรูปที่ 2.12 โดยในรูปจะบอกด้วยว่าขาใดจะแอคทีฟ



รูปที่ 2.12 แสดงการจัดหน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51

ในไมโครคอนโทรลเลอร์ 8031 จะมีหน่วยความจำภายในตั้งแต่ตำแหน่ง 00H จนถึง FFH และสามารถอ้างหน่วยความจำโปรแกรมภายนอกได้ 64K ตำแหน่ง ถ้าอ่านข้อมูลจากหน่วยความจำโปรแกรมขา PSEN จะแอคทีฟ นอกจากนี้ 8031 สามารถอ้างหน่วยความจำข้อมูลภายนอกได้ 64K ตำแหน่งโดยการติดต่อกับหน่วยความจำนี้ ขา RD และ WR จะแอคทีฟ สำหรับหน่วยความจำข้อมูลภายในนั้นจะแบ่งออกได้ดังนี้

1. ชุดรีจีสเตอร์ 4 ชุด แต่ละชุดเรียกว่ารีจีสเตอร์แบงค์ที่ตำแหน่ง 00H ถึง 01H โดยแต่ละชุดประกอบด้วยรีจีสเตอร์ R0 ถึง R7

2.หน่วยความจำที่สามารถเข้าถึงข้อมูลระดับบิตได้ ตำแหน่ง 20H ถึง 2FH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

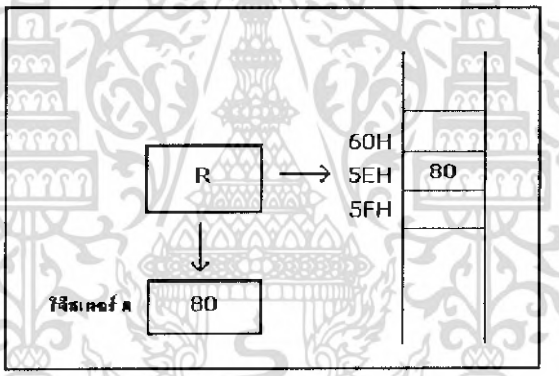
3. หน่วยความจำใช้งานทั่วไปตำแหน่ง 30H ถึง 7FH

4. รีจีสเตอร์ฟังก์ชันพิเศษ ตำแหน่ง 80H ถึง FFH

การอ้างตำแหน่งหน่วยความจำภายในจะอ้างได้ 2 แบบ คือ การอ้างไปที่ตำแหน่งของไบต์ (เขียนหมายเลขตำแหน่งด้านนอก) หรือ การอ้างไปที่ตำแหน่งของบิต (เขียนหมายเลขตำแหน่งด้านใน) โดยตำแหน่งของหน่วยความจำที่อ้างเป็นแบบบิตได้จะมีตำแหน่งบิตที่แน่นอน

หน่วยความจำใช้งานทั่วไป

จากรูปที่ 2.13 จะเห็นว่าใน 8031 จะมีหน่วยความจำ RAM สำหรับใช้งานทั่วไปจำนวน 80 ไบต์ ตั้งแต่ตำแหน่ง 30H ถึง 7FH ตำแหน่งเหล่านี้ สามารถอ้างตำแหน่งแบบ Direct Addressing Mode หรือ Indirect Addressing Mode ได้



รูปที่ 2.13 แสดงขั้นตอนต่าง ๆ ในการอ่านข้อมูล

Bit - Addressable RAM

ในไมโครคอนโทรลเลอร์MCS-51 จะมีหน่วยความจำที่สามารถอ้างข้อมูลในระดับบิตได้ ตั้งแต่ตำแหน่ง 20H ถึง 2FH รวม 16 ไบต์ โดยสามารถทางลอจิกได้ จำนวนบิตที่ใช้งานได้ทั้งหมดมีจำนวน 128 บิต (8 บิต x 16 ไบต์)

Register Banks

หน่วยความจำข้อมูลภายในที่เป็นชุดรีจีสเตอร์มีทั้งหมด 32 ตำแหน่ง โดยจะมี 4 ชุด แต่ละชุดมีรีจีสเตอร์ 8 ตัว คือ R0 ถึง R7 โดยชุดแรกจะอยู่ในตำแหน่ง 00H - 07H ถ้าไม่มีการเลือก Bank จะเป็นการติดต่อกับรีจีสเตอร์ Bank แรกเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
72836

2.3.7 รีจิสเตอร์ฟังก์ชั่นพิเศษ (Special Function Register : SFR)

รีจิสเตอร์ฟังก์ชั่นพิเศษ เป็นรีจิสเตอร์ที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 ซึ่งมีด้วยกัน 22 ตัวสำหรับเบอร์ AT89C51 และ28 ตัวในเบอร์ AT89C52 และ อนุกรม TA89Sxx ทั้งนี้เนื่องจากใน AT89C52 และ AT89Sxx มีจำนวนไทเมอร์/เคาน์เตอร์มากกว่า AT89C51

รีจิสเตอร์ SFR มีแอดเดรสอยู่ระหว่าง 80H-FFH ในพื้นที่ของหน่วยความจำข้อมูลส่วนบน สามารถเข้าถึงได้โดยตรง (Direct Addressing)

รีจิสเตอร์แสดงสถานะของโปรแกรม (Program Status Word : PSW)

เป็นรีจิสเตอร์ขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต จึงสามารถกำหนดค่าในแต่ละบิตของรีจิสเตอร์ตัวนี้ได้อย่างอิสระ มีแอดเดรสอยู่ที่ D0H ทำหน้าที่เก็บสถานะของการทำงานของโปรแกรมในขณะนั้นจะเรียกสถานะต่าง ๆ ของโปรแกรม แฟลก (Flag) เมื่อซีพียูกระทำคำสั่งทางคณิตศาสตร์ และ ลอตจิกแล้วเกิดการเปลี่ยนแปลงสถานะขึ้น ผลของการเปลี่ยนแปลงนั้นจะมาปรากฏที่บิตต่าง ๆ ของรีจิสเตอร์ PSW รายละเอียดของแต่ละบิตในรีจิสเตอร์ PSW

รีจิสเตอร์ B (B Register)

รีจีสเตอร์ B จะอยู่ตำแหน่ง F0H ของหน่วยความจำข้อมูลภายใน เป็นรีจีสเตอร์ที่สามารถใช้งานทั่วไปได้ โดยทั่วไปรีจีสเตอร์นี้จะใช้คูณหรือหารกับรีจีสเตอร์ Accumulator รีจีสเตอร์ Bนี้สามารถเข้าถึงข้อมูลระดับบิตได้ โดยตำแหน่งของบิตคือตำแหน่ง F0H ถึง F7H

ตัวชี้สแตค (Stack Pointer : SP)

ตัวชี้สแตคหรือรีจิสเตอร์ตัวชีสแตค มีขนาด 8 บิต มีแอดเดรสอยู่ที่ 81H ใช้ในการเก็บค่าตำแหน่งของตัวชี้สแตค ซึ่งสามารถเปลี่ยนแปลงได้ เมื่อซีพียูมีการกระโดยไปทำงานที่โปรแกรมย่อย หรือ กระโดดจากโปรแกรมย่อมแล้วกลับมายังโปรแกรมหลัก เมื่อมีการีเซตเกิดขึ้น (รีเซต : การกระทำที่ส่งผลให้ซีพียูต้องเริ่มต้นการทำงานใหม่ตั้งแต่ต้น) ค่าของรีจิสเตอร์ SP จะเท่ากับ 07H ดังนั้นแอดเดรสแรกของพื้นที่ที่สำรองไว้ทำหน้าที่เป็นสแตคจะเท่ากับ 08H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจีสเตอร์ Data Pointer (DPTR)

รีจีสเตอร์นี้สำหรับชี้ตำแหน่งรหัสโปรแกรมหรือข้อมูลในหน่วยความจำ โดยเป็นรีจีสเตอร์ขนาด 16 บิต ซึ่งประกอบด้วย รีจีสเตอร์ 2 ตัว คือ DPL ตำแหน่งที่ 82H โดยจะเก็บค่า 8 บิตต่ำ และ DPH ตำแหน่งที่ 83H โดยจะเก็บค่า 8 บิตสูง รีจีสเตอร์ทั้งสองตัวนี้จะรวมกันกลายเป็นรีจีสเตอร์ 16 บิต รีจิสเตอร์ DPTR นี้ใช้ในการเก็บค่าแอดเดรสของหน่วยความจำหรืออุปกรณ์ภายนอกที่ไมโครคอนโทรลเลอร์ต้องการติดต่อด้วย

รีจีสเตอร์พอร์ต (Port Registers)

ในไมโครคอนโทรลเลอร์ MCS-51 ค่าของพอร์ทจะหมายถึงค่าของหน่วยความจำด้วย หากต้องการส่งข้อมูลออกไปที่พอร์ต ก็เพียงแต่เขียนข้อมูลไปที่หน่วยความจำตำแหน่งที่พอร์ต นั้นอยู่ใน MCS-51 พอร์ต 0 จะอยู่ที่ตำแหน่ง 80H , พอร์ต 1 จะอยู่ที่ตำแหน่ง 90H, พอร์ท 2 จะอยู่ที่ตำแหน่ง A0H และพอร์ต 3 จะอยู่ที่ตำแหน่ง B0H พอร์ต 0,2 และ 3 โดยทั่วไปแล้วจะไม่ใช้ถ้าหากมีการติดต่อกับหน่วยความจำภายนอก หรือ ใช้เป็นพอร์ตพิเศษ (เช่น Interrupts, Serial Port, ฯลฯ) โดยปกติแล้วจะใช้พอร์ต 1 ในการติดต่อกับอุปกรณ์ภายนอก พอร์ตทุกพอร์ตสามารถอ้างข้อมูลในระดับบิตได้

รีจีสเตอร์เวลา (Timer Registers)

รีจีสเตอร์เวลา เป็นรีจิสเตอร์ขนาด 16 บิต แบ่งเป็นไบต์สูง และ ไบต์ต่ำสุดเช่นเดียวกับรีจิสเตอร์ DPTR รีจิสเตอร์ไทเมอร์ใช้ในการเก็บค่าของตัวนับหรือเคาน์เตอร์ (Counter) ภายในตัวของไมโครคอนโทรลเลอร์ เพื่อใช้ในการสร้างฐานเวลา,จับเวลา หรือนับจำนวนพัลส์สัญญาณนาฬิกาภายใน บางทีเรียกว่ารีจิสเตอร์ตัวนี้ว่ารีจิสเตอร์ไทเมอร์/เคาน์เตอร์

ในไมโครคอนโทรลเลอร์เบอร์ AT89C51 มีรีจิสเตอร์ไทเมอร์/เคาน์เตอร์ 2 ตัวแบ่งเป็น T0 หรือ Timer 0 และ T1 หรือ Timer 1 ในรีจิสเตอร์ยังแบ่งเป็นรีจิสเตอร์ไทเมอร์ไบต์ต่ำ (TL) และรีจิสเตอร์ไทเมอร์ไบต์สูง (TH) เหมือนกัน โดยรีจิสเตอร์ TLO มีแอดเดรสอยู่ที่ 8AH รีจิสเตอร์ TH0 มีแอดเดรสอยู่ที่ 8BH ในขณะที่ TL1 และ TH1 มีแอดเดรสอยู่ที่ 8CH และ 8DH สำหรับในเบอร์ AT89C52 และในอนุกรม AT89Sxx จะมีรีจิสเตอร์ไทเมอร์/เคาน์เตอร์ถึง 3 ตัว โดยมีรีจิสเตอร์ TL2 และ TH2 ซึ่งมีแอดเดรสอยู่ที่ 0CCH และ 0CDH เพิ่มเติม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจีสเตอร์พอร์ทอนุกรม (Serial Port Registers)

ไมโครคอนโทรลเลอร์ MCS-51 จะมีพอร์ตสื่อสารอนุกรม (Serial Port) อยู่ภายในชิพ ซึ่งสามารถจะรับและส่งข้อมูลได้โดยติดต่อผ่านรีจีสเตอร์ SBUF (Serial Data Buffer) ซึ่งอยู่ที่ตำแหน่ง 99H โดยถ้าต้องการส่งข้อมูลแบบอนุกรมให้เขียนข้อมูลไปที่รีจีสเตอร์นี้ ตัว Serial Port สามารถโปรแกรมให้ทำงานได้ 4 โหมด โดยโปรแกรมผ่านรีจีสเตอร์ SCON (Serial Port Control Register) ตำแหน่ง 98H

รีจีสเตอร์อินเตอร์รัพท์ (Interrupt Port Registers)

ไมโครคอนโทรลเลอร์ MCS-51 สามารถ Interrupt ได้ 5 ตำแหน่ง โดยมี 2 priority ตัว Interrupt นี้จะถูก Disable หลังจากระบบถูกรีเซต และจะ Enabled หลังจากที่เขียนข้อมูล ไปที่รีจีสเตอร์ IE หรือตำแหน่ง A8H ลำดับความสำคัญสามารถเซตได้ที่รีจีสเตอร์ IP หรือตำแหน่ง B8H

Power Control Register (PCON)

รีจีสเตอร์ PCON อยู่ที่ตำแหน่ง 87H ใช้หยุดการทำงานของ ไมโครคอนโทรลเลอร์ MCS-51 โดยจะหยุดจ่ายสัญญาณนาฬิกาให้ระบบ ทำให้ข้อมูลต่างๆภายในไมโครคอนโทรลเลอร์ MCS-51 ไม่มีการเปลี่ยนแปลง นอกจากนี้ยังลดพลังงานไฟฟ้าที่จ่ายให้ ไมโครคอนโทรลเลอร์ MCS-51 ลงด้วย

รีจิสเตอร์ควบคุม (Control Register)

PCON เป็นรีจิสเตอร์ที่เกี่ยวข้องกับ การกำหนดอัตราการรับส่งข้อมูลของวงจรสื่อสารอนุกรม และกำหนดการทำงานโหมดประหยัดพลังงานของไมคอนโทรลเลอร์ MCS-51 แบบแฟลช

SCON เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของวงจรสื่อสารอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

TCON และT2CON เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของตัวไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดย T2CON ใช้สำหรับไทเมอร์/เคาน์เตอร์ 2 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxx

TMOD และ T2MOD เป็นรีจิสเตอร์ที่ใช้กำหนดโหมด หรือ ลักษณะในการทำงานของไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชโดย T2MOD ใช้สำหรับไทเมอร์/เคาน์เตอร์ 2 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52

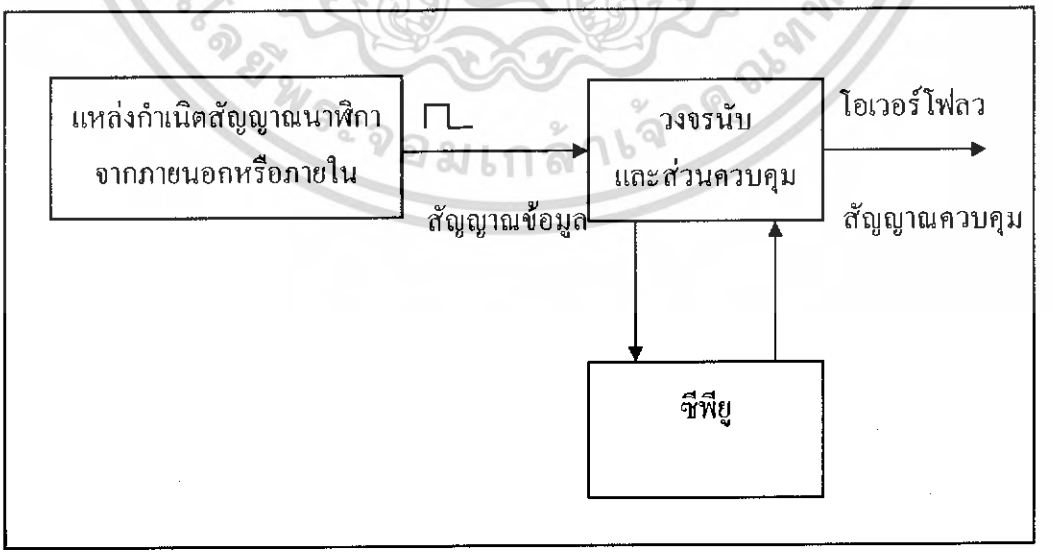
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IE และ IP เป็นรีจิสเตอร์ที่เกี่ยวข้องกับการตอบสนองการอินเตอร์รัพต์ (Interrupt : การขัดจังหวะการทำงานปกติของซีพียู) โดย IE เป็นรีจิสเตอร์สำหรับอีนาเบิลหรือใช้ในการกำหนดลักษณะของการตอบสนองการอินเตอร์รัพต์ในขณะที่ IP เป็น รีจิสเตอร์สำหรับกำหนดลำดับความสำคัญของการตอบสนองการอินเตอร์รัพต์ ว่าจะให้ซีพียูตอบสนองการเกิดอินเตอร์รัพต์ในลักษณะใดก่อนหลัง

2.3.8 ไทเมอร์/เคาน์เตอร์ (Timer/Counter)

ไทเมอร์/เคาน์เตอร์ เป็นวงจรภายในที่ช่วยเพิ่มความสามารถให้กับไมโครคอนโทรลเลอร์ เนื่องจากไทเมอร์/เคาน์เตอร์มีลักษณะการทำงานเป็นตัวนับเวลาหรือนับสัญญาณนาฬิกาซึ่งสามารถทำงานเกี่ยวกับฐานเวลา, การสร้างสัญญาณพัลส์, การเปรียบเทียบค่าของเวลา หรือ ค่าการนับของสัญญาณพัลส์ที่รับเข้ามาจากภายนอก และรวมไปถึงกับการควบคุมอัตราการรับส่งข้อมูลของพอร์ตอนุกรมด้วยไทเมอร์/เคาน์เตอร์ เป็นวงจรที่สามารถทำงานได้เอง โดยเป็นอิสระจากซีพียู แต่ซีพียูสามารถควบคุมการทำงาน, การอ่านและการเขียนข้อมูลต่าง ๆ ได้

ไมโครคอนโทรลเลอร์ในแต่ละเบอร์อาจจะมีไทเมอร์/เคาน์เตอร์จำนวนที่แตกต่างกัน หรือมีความสามารถที่แตกต่างกัน เช่น เบอร์ AT89C51 มีไทเมอร์/เคาน์เตอร์จำนวน 2 ตัว (T0 และ T1) ขณะที่เบอร์ AT89C52 หรือเบอร์ AT89C51AC2 มีไทเมอร์/เคาน์เตอร์จำนวน 3 ตัว(T0, T1 และ T2) ไทเมอร์/เคาน์เตอร์แต่ละตัวอาจจะมีความสามารถที่แตกต่างกัน ในรายละเอียดทางฮาร์ดแวร์ และการทำงานแต่อย่างไรก็ตามจะมีหลักการทำงานที่เหมือนกันคือถูกควบคุมจากซีพียูและใช้สัญญาณ



รูปที่ 2.14 แสดงโครงสร้างโดยรวมของไทเมอร์/เคาน์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นาฬิกามาเป็นสัญญาณอินพุทในกรณีที่ใช้งานเป็นไทเมอร์ หรือ รับสัญญาณอินพุทจากภายนอกในกรณีที่ใช้งานเป็นเคาน์เตอร์ นอกจากนี้แล้วไทเมอร์/เคาน์เตอร์ทุกตัวยังมีการทำงานที่เป็นอิสระต่อกันอีกด้วย

1. ไทเมอร์/เคาน์เตอร์ 0 และ ไทเมอร์/เคาน์เตอร์ 1

การทำงานในโหมดไทเมอร์ (Timer Mode) ของไทเมอร์/เคาน์เตอร์ 0 และ 1

เมื่อกำหนดให้มีการทำงานเป็นโหมดไทเมอร์หรือตัวตั้งเวลาวงจรนับหรือเคาน์เตอร์ภายในจะทำการนับขึ้นโดยใช้สัญญาณนาฬิกาภายใน ซึ่งสัญญาณนาฬิกาตัวนี้จะเกิดขึ้นทุก ๆ 1 แมชชิน-ไซเคิล (Machine cycle) นั่นหมายความว่าจะเกิดการนับขึ้น 1 ค่าในแต่ละแมชชินไซเคิลนั่นเอง โดยทั่วไปในไมโครคอนโทรลเลอร์โดยทุก ๆ 1 แมชชินไซเคิล จะใช้เวลา 12 คาบของสัญญาณนาฬิกาที่ป้อนให้ตัวไมโครคอนโทรลเลอร์ นั่นก็หมายความว่า จะเกิดการนับขึ้นทุก ๆ 1/12 ของความถี่สัญญาณนาฬิกา

การทำงานในโหมดเคาน์เตอร์ (Counter Mode) ของไทเมอร์/เคาน์เตอร์ 0 และ 1

เมื่อกำหนดให้มีการทำงานเป็นโหมดเคาน์เตอร์หรือตัวนับ วงจรนับหรือเคาน์เตอร์ภายในจะทำการนับขึ้นโดยใช้สัญญาณขอบขาลง (เปลี่ยนจาก '1' เป็น '0') จากภายนอกที่เข้ามาทางขา T0 (P3.4), T1 (P3.5) กระบวนการอ่านสัญญาณอินพุทเหล่านี้จะใช้เวลาทั้งสิ้น 2 แมชชินไซเคิล นั่นหมายความว่าอัตราการนับจะมีค่าเท่ากับ 1/24 (กรณี 1 แมชชินไซเคิลใช้เวลา 12 คาบ) ของความถี่สัญญาณนาฬิกา ดังนั้นสามารถใช้การทำงานในโหมดเคาน์เตอร์เพื่อนับสัญญาณนาฬิกาได้สูงสุดเท่ากับ ความถี่สัญญาณนาฬิกา/24 (กรณี 1 แมชชินไซเคิล ใช้เวลา 12 คาบ) เช่น ใช้สัญญาณนาฬิกา 12 MHz จะสามารถรับสัญญาณอินพุทที่มีความถี่สูงสุดได้ 12 MHz/24 = 500kHz (กรณี 1 แมชชินไซเคิลใช้เวลา 12 คาบ)

รีจิสเตอร์ของไทเมอร์/เคาน์เตอร์ 0 และ 1 (Timer/Counter0, 1 Register)

การทำงานให้ไทเมอร์/เคาน์เตอร์ ทำงานตามความต้องการนั้นจะต้องมีการกำหนดค่าต่าง ๆ ให้กับรีจิสเตอร์ที่เกี่ยวข้องกับการทำงานอย่างเหมาะสม ซึ่งรีจิสเตอร์เหล่านี้เป็นรีจิสเตอร์ฟังก์ชันพิเศษ (SFR) ซึ่งมีรายละเอียดดังนี้

- TL0 เป็นรีจิสเตอร์ไบต์ต่ำของไทเมอร์ 0 มีแอดเดรสอยู่ที่ 8AH
- TH0 เป็นรีจิสเตอร์ไบต์สูงของไทเมอร์ 0 มีแอดเดรสอยู่ที่ 8BH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- TL1 เป็นรีจิสเตอร์ไบต์ต่ำของไทเมอร์ 1 มีแอดเดรสอยู่ที่ 8CH

- TH1 เป็นรีจิสเตอร์ไบต์สูงของไทเมอร์ 1 มีแอดเดรสอยู่ที่ 8DH

รีจิสเตอร์ ทั้ง 4 ตัวนี้เป็นรีจิสเตอร์ขนาด 8 บิต ดังนั้นเมื่อนำ TL0 มารวมกันกับ TH0 จะทำให้ได้รีจิสเตอร์ของไทเมอร์ 0 ขนาด 16 บิต ในทำนองเดียวกันเมื่อนำ TL1 มารวมกันกับ TH1 จะทำให้ได้รีจิสเตอร์ของไทเมอร์ 1 ขนาด 16 บิต ซึ่งสามารถเก็บค่าได้สูงสุดถึง 65,536 ค่า (0-65,536 หรือ 0000H ถึง FFFFH)

รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

(Timer/Count0, 1 Control Register : TCON)

TCON เป็นรีจิสเตอร์ที่ทำหน้าที่ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ1 มีขนาด 8 บิต และ มีแอดเดรสอยู่ที่ 88H และสามารถเข้าถึงในระดับบิตได้ ในแต่ละบิตจะมีหน้าที่แตกต่างกันออกไปดังนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

T2MOD

- TF1 (Timer 1 overflow flag) จะเกิดการเซตด้วยกระบวนการทางฮาร์ดแวร์ คือเมื่อการนับเพิ่มขึ้นจนเกิดการโอเวอร์โฟลว (เปลี่ยนจาก FFH ไปเป็น 00H ในกรณีที่เป็นแบบ 8 บิต และเปลี่ยนจาก FFFFH ไปเป็น 0000H กรณีที่เป็นแบบ 16บิต) บิตนี้จะถูกเคลียร์โดยกระบวนการทางฮาร์ดแวร์เช่นกันโดยจะเคลียร์ เมื่อซีพียูกระโดดไปทำงานในโปรแกรมบริการการอินเตอร์รัพต์ของไทเมอร์ 1 โอเวอร์โฟลว

-TR1 (Timer 1 run control bit) ใช้ควบคุมการปิด/เปิด (Enable/Disable) การทำงานของไทเมอร์ 1 การเซตและการเคลียร์สามารถกระทำได้โดยกระบวนการทางซอฟต์แวร์

'0' ปิดการทำงานของไทเมอร์ 0 (disable)

'1' เปิดการทำงานของไทเมอร์ 0 (enable)

- TF0 (Timer 0 overflow flag) จะเกิดการเซตด้วยกระบวนการทางฮาร์ดแวร์ คือเมื่อการนับเพิ่มขึ้นจนเกิดการโอเวอร์โฟลว บิตนี้จะถูกเคลียร์ โดยกระบวนการทางฮาร์ดแวร์เช่นกัน โดยจะเคลียร์ เมื่อซีพียูกระโดดไปทำงานในโปรแกรมบริการการอินเตอร์รัพต์ของไทเมอร์ 0 โอเวอร์โฟลว

- TR0 (Timer 0 run control bit) ใช้ควบคุมการปิด/เปิด (Enable/Disable) การทำงานของไทเมอร์ 0 การเซตและการเคลียร์สามารถกระทำได้โดยกระบวนการทางซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'0' ปิดการทำงานของไทเมอร์ 0 (disable)

'1' เปิดการทำงานของไทเมอร์ 0 (enable)

- IE1 (External Interrupt1 edge flag) ใช้ในกระบวนการอินเตอร์รัพต์ คือ จะเกิดการเซตโดยกระบวนการทางฮาร์ดแวร์ นั่นคือตรวจพบว่ามีสัญญาณอินเตอร์รัพต์เข้ามาที่ขาอินพุทของอินเตอร์รัพต์ 1 (INT1: P3.3) บิตนี้จะเซต และ จะเคลียร์เมื่อ ซีพียูกระโดดไปทำงานในโปรแกรมบริการการอินเตอร์รัพต์ของเอ็กเทอร์นอลอินเตอร์รัพต์ 1

- IT1 (Interrupt 1 type control bit) ใช้ในกระบวนการอินเตอร์รัพต์ของเอ็กเทอร์นอลอินเตอร์รัพต์ 1 โดยเป็นตัวกำหนดว่าจะให้เป็นอินเตอร์รัพต์เกิดขึ้นเมื่อตรวจพบสัญญาณที่เข้ามาทางขาอินพุทของอินเตอร์รัพต์ 1 (INT1: P3.3) เป็นการตรวจสอบสัญญาณขอบขาลงหรือระดับลอจิก "0" การเซตและเคลียร์บิตนี้สามารถทำได้โดยกระบวนการทางซอฟต์แวร์

'0' จะเกิดการอินเตอร์รัพต์เมื่อตรวจพบสัญญาณขอบขาลง (falling edge)

'1' จะเกิดการอินเตอร์รัพต์เมื่อตรวจพบสัญญาณเป็นลอจิก "0" (low level triggered)

- IE0 (External Interrupt 0 edge flag) ใช้ในกระบวนการอินเตอร์รัพต์ คือ จะเกิดการเซตโดยกระบวนการทางฮาร์ดแวร์ นั่นคือตรวจพบว่ามีสัญญาณอินเตอร์รัพต์เข้ามาที่ขาอินพุทของอินเตอร์รัพต์ 0 (INT0: P3.2) บิตนี้จะเซต และ จะเคลียร์เมื่อ ซีพียูกระโดดไปทำงานในโปรแกรมบริการการอินเตอร์รัพต์ของเอ็กเทอร์นอลอินเตอร์รัพต์ 0

- IT0 (Interrupt 0 type control bit) ใช้ในกระบวนการอินเตอร์รัพต์ของเอ็กเทอร์นอลอินเตอร์รัพต์ 0 โดยเป็นตัวกำหนดว่าจะให้เป็นอินเตอร์รัพต์เกิดขึ้นเมื่อตรวจพบสัญญาณที่เข้ามาทางขาอินพุทของอินเตอร์รัพต์ 0 (INT0: P3.2) เป็นการตรวจสอบสัญญาณขอบขาลงหรือระดับลอจิก "0" การเซตและเคลียร์บิตนี้สามารถทำได้โดยกระบวนการทางซอฟต์แวร์

'0' จะเกิดการอินเตอร์รัพต์เมื่อตรวจพบสัญญาณขอบขาลง (falling edge)

'1' จะเกิดการอินเตอร์รัพต์เมื่อตรวจพบสัญญาณเป็นลอจิก "0" (low level triggered)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

(Timer/Count0, 1 Mode Control Register : TMOD)

TMOD เป็นรีจิสเตอร์ที่ทำหน้าที่เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 มีขนาด 8 บิตและมีแอดเตรสอยู่ที่ 89H ไม่สามารถเข้าถึงในระดับบิตได้ รีจิสเตอร์ตัวนี้จะแยกออกเป็น 2 ส่วน คือ บิตบนจะเอาไว้ใช้เลือกโหมดของไทเมอร์/เคาน์เตอร์ 1 และ 4 บิตล่างจะใช้เลือกโหมดของไทเมอร์/เคาน์เตอร์ 0 โดยในแต่ละบิตจะมีหน้าที่แตกต่างกันออกไปดังนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GATE	C/T'	M1	M0	GATE	C/T'	M1	M0
ไทเมอร์/เคาน์เตอร์ 1				ไทเมอร์/เคาน์เตอร์ 0			

TMOD

- GATE ทำหน้าที่เป็นตัวเลือกรูปแบบการทำงานของไทเมอร์/เคาน์เตอร์ ว่าจะให้มีการควบคุมการทำงานแบบซอฟต์แวร์หรือแบบฮาร์ดแวร์ดังนี้

'0' เลือกรูปแบบการควบคุมการทำงานเป็นแบบซอฟต์แวร์ คือไทเมอร์/เคาน์เตอร์ x จะทำงานเมื่อ TRx ในรีจิสเตอร์ TCON เป็น '1'

'1' เลือกรูปแบบการควบคุมการทำงานเป็นแบบซอฟต์แวร์ คือไทเมอร์/เคาน์เตอร์ x จะทำงานเมื่อ TRx ในรีจิสเตอร์ TCON เป็น '1' และ ที่ขาอินพุทของอินเตอร์รัพต์ INTx เป็นสถานะลอจิก '1'

- M1, M0 (Mode Selector bit) เป็นบิตที่ใช้เลือกโหมดการทำงานของไทเมอร์ซึ่งมีดังนี้

'00' เลือกโหมดการทำงานไทเมอร์/เคาน์เตอร์เป็นแบบ 13 บิต

'01' เลือกโหมดการทำงานไทเมอร์/เคาน์เตอร์เป็นแบบ 16 บิต

'10' เลือกโหมดการทำงานไทเมอร์/เคาน์เตอร์เป็นแบบ 8 บิต รีโหลดค่าอัตโนมัติ

'11' เลือกโหมดการทำงานไทเมอร์/เคาน์เตอร์ 0 (ไทเมอร์/เคาน์เตอร์1 ไม่ทำงาน) ให้ทำงานแบบแยกส่วน โดยแยกออกเป็นไทเมอร์/เคาน์เตอร์ 2 ตัว โดยตัวแรกจะใช้รีจิสเตอร์ TL0 จะถูกควบคุมการทำงานโดย TR0 ส่วนตัวที่สองจะใช้รีจิสเตอร์ TH0 จะถูกควบคุมการทำงานโดย TR1 (TR0 และ TR1 อยู่ในรีจิสเตอร์ TCON)

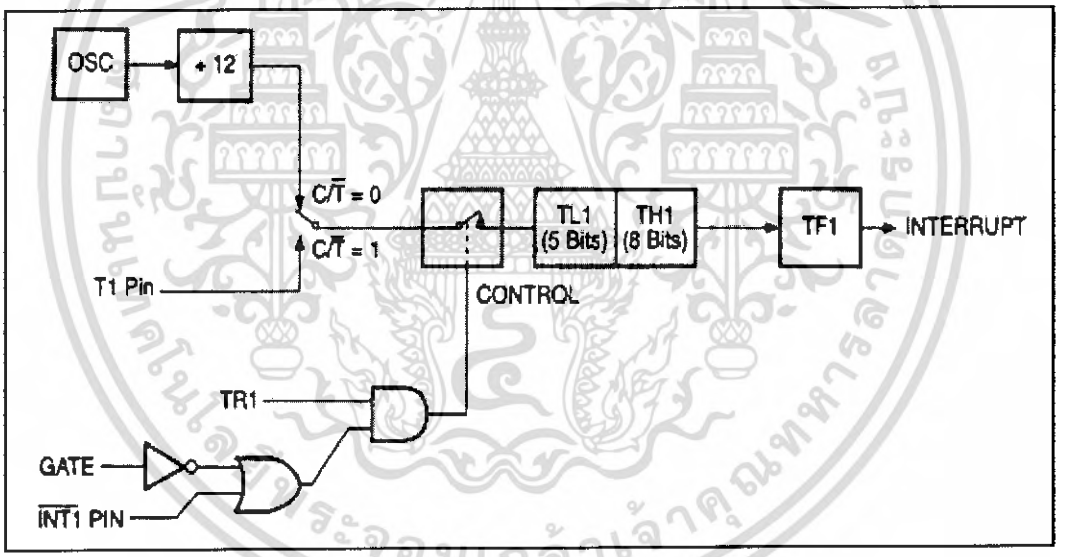
การทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 ในโหมดต่างๆ

ดังที่ได้กล่าวมาแล้วในข้างต้นว่าไทเมอร์/เคาน์เตอร์ 0 และ 1 มีอยู่ทั้งหมด 4 โหมดที่มีการทำงานแตกต่างกันออกไป ซึ่งสามารถควบคุมการทำงานและเลือกโหมดการทำงานได้โดยรีจิสเตอร์ TCON และรีจิสเตอร์ TMOD ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 0 (ไทเมอร์/เคาน์เตอร์ 13 บิต)

การทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 ในโหมด 0 จะใช้ TL0,1 5 บิต และ ใช้ TH0,1 ครบทั้ง 8 บิต รวมเป็น 13 บิต ซึ่งจะทำหน้าที่เป็นวงจรนับขึ้นขนาด 13 บิต โดยสามารถรับสัญญาณนาฬิกาได้จากสองแหล่งกำเนิด ซึ่งถูกควบคุมจากบิต C/T' ในรีจิสเตอร์ TCON กล่าวคือเมื่อ C/T' = '0' จะใช้สัญญาณนาฬิกาจากภายใน (*ความถี่ออสซิลเลเตอร์/12*) และเมื่อ C/T' = '1' จะใช้สัญญาณนาฬิกาจากภายนอกที่ที่รับเข้ามาทางขา T0,1 (P3.4, P3.5) แต่สัญญาณนาฬิกาที่เลือกนี้จะยังไม่สามารถส่งไปให้กับวงจรนับได้ ถ้าหากบิต TR0, 1 ในรีจิสเตอร์ TCON ยังคงมีค่าเท่ากับ '0' อยู่ การที่จะทำให้มีสัญญาณนาฬิกาส่งไปยังวงจรนับได้หรือให้ไทเมอร์/เคาน์เตอร์เริ่มนับนั้นสามารถทำได้โดยกำหนดให้บิต Gate ในรีจิสเตอร์ TMOD = '0' หรือให้ลอจิกที่ขา INT0,1'(P3.2, P3.3) = '1' ดอนนี้ไทเมอร์/เคาน์เตอร์พร้อมที่จะทำงานแล้ว และจะเริ่มทำงานหรือเริ่มนับทันทีเมื่อกำหนดให้ TR0, 1 ในรีจิสเตอร์ TCON = '1'



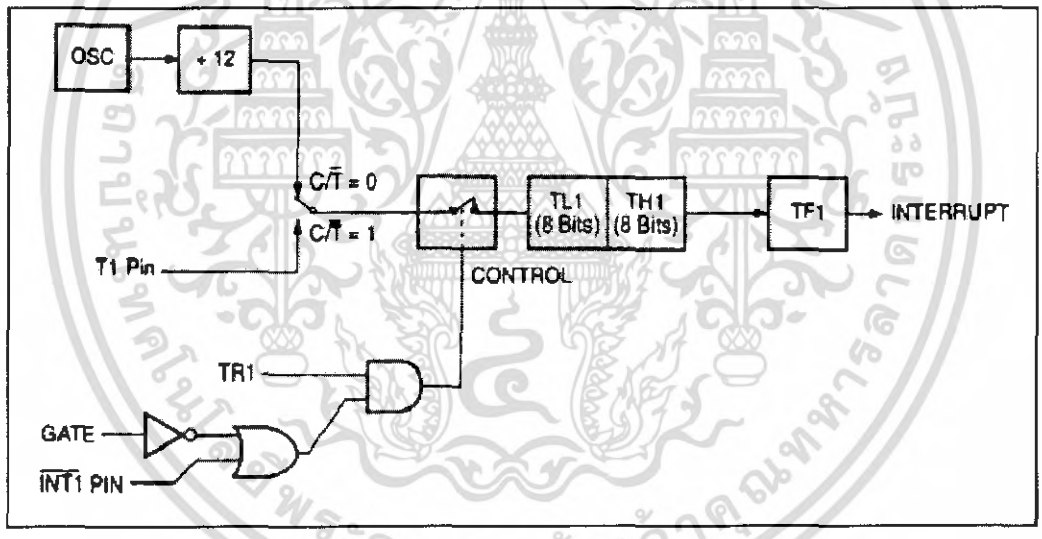
รูปที่ 2.15 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0,1 โหมด 0
(ไทเมอร์/เคาน์เตอร์13 บิต)

เมื่อไทเมอร์/เคาน์เตอร์เริ่มทำงานนับวงจรภายใน จะเริ่มนับขึ้นไปเรื่อย ๆ และ เมื่อนับถึงค่าสูงสุดแล้วจะวนกลับมาเริ่มที่ 0000H ใหม่อีกครั้งซึ่งจังหวะนี้เรียกว่าเกิดการโอเวอร์โฟลว (Over flow) สัญญาณโอเวอร์โฟลวนี้จะทำให้บิต TF0,1 ในรีจิสเตอร์ TCON เกิดการเซต และ จะถูกเคลียร์เมื่อซีพียูกระโดดไปทำยังโปรแกรมอินเตอร์รัพต์ 1, 3 ซึ่งเป็นตำแหน่งของการอินเตอร์รัพต์ของไทเมอร์/เคาน์เตอร์ 0,1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 1 (ไทเมอร์/เคาน์เตอร์ 16 บิต)

การทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 ในโหมด 1 จะใช้ TL0, 1 และ TH0, 1 ตัวละ 8 บิต รวมเป็น 16 บิต ซึ่งจะทำหน้าที่เป็นวงจรนับขึ้นขนาด 16 บิต โดยสามารถรับสัญญาณนาฬิกาได้จากสองแหล่งกำเนิด ซึ่งถูกควบคุมจากบิต C/T' ในรีจิสเตอร์ TCON กล่าวคือเมื่อ C/T' = '0' จะใช้สัญญาณนาฬิกาจากภายใน (*ความถี่ออสซิลเลเตอร์/12*) และเมื่อ C/T' = '1' จะใช้สัญญาณนาฬิกาจากภายนอกที่รับเข้ามาทางขา T0,1 (P3.4, P3.5) แต่สัญญาณนาฬิกาที่เลือกนี้จะยังไม่สามารถป้อนให้กับวงจรนับได้ ถ้าหากบิต TR0, 1 ในรีจิสเตอร์ TCON ยังคงมีค่าเท่ากับ '0' อยู่ การที่จะทำให้มีสัญญาณนาฬิกาป้อนไปยังวงจรนับได้ หรือ ทำให้ไทเมอร์/เคาน์เตอร์ทำงานนั้นสามารถทำได้โดยกำหนดให้บิต Gate ในรีจิสเตอร์ TMOD = '0' หรือกำหนดให้ลอจิกที่ขา INT0,1'(P3.2, P3.3) = '1' ดอนนี้ไทเมอร์/เคาน์เตอร์พร้อมที่จะทำงานแล้ว และ จะเริ่มทำงานได้เมื่อกำหนดให้ TR0,1 ในรีจิสเตอร์ TCON = '1'



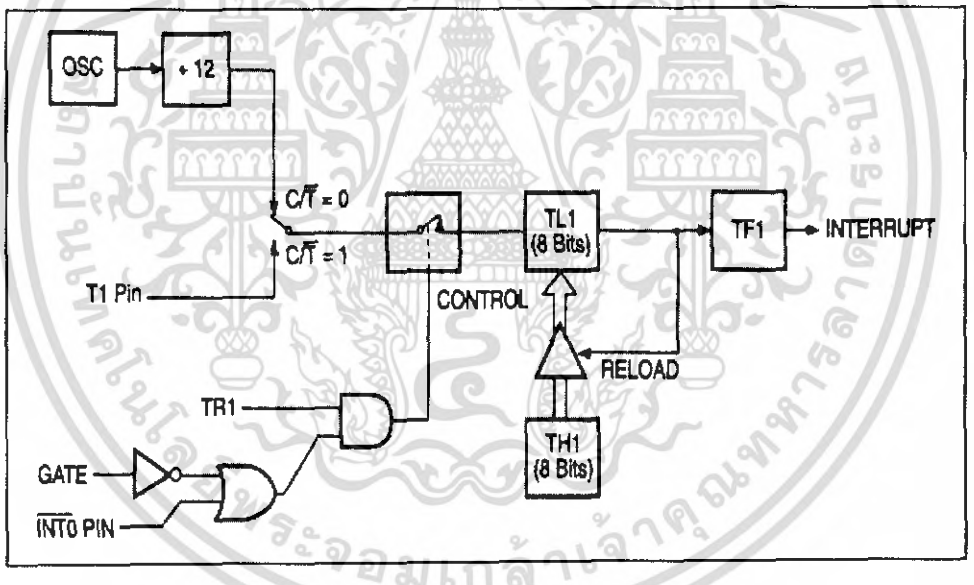
รูปที่ 2.16 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 1
(ไทเมอร์/เคาน์เตอร์ 16 บิต)

เมื่อไทเมอร์/เคาน์เตอร์เริ่มทำงานวงจรนับภายใน จะเริ่มนับขึ้นไปเรื่อย ๆ และ เมื่อนับถึงค่าสูงสุด (FFFFH) แล้วจะวนกลับมาเริ่มที่ 0000H ใหม่อีกครั้งซึ่งจังหวะนี้เรียกว่าเกิดการโอเวอร์ไฟลว (Over flow) สัญญาณโอเวอร์โฟลวนี้จะทำให้บิต TF0, 1 ในรีจิสเตอร์ TCON เกิดการเซต และจะถูกเคลียร์เมื่อซีพียูกระโดดไปทำยังโปรแกรมอินเตอร์รัพต์ 1, 3 ซึ่งเป็นตำแหน่งของการอินเตอร์รัพต์ของไทเมอร์/เคาน์เตอร์ 0, 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 2 (ไทเมอร์/เคาน์เตอร์ 8 บิต รีโหลดค่าอัตโนมัติ)

การทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 ในโหมด 2 จะใช้ TL0, 1 เป็นตัวเก็บค่าของการนับ ซึ่งเป็นวงจรนับขนาด 8 บิต และใช้ TH0,1 เป็นตัวเก็บค่ารีโหลด โดยวงจรนับสามารถรับสัญญาณนาฬิกาได้จากสองแหล่งกำเนิด ซึ่งถูกควบคุมจากบิต C/T' ในรีจิสเตอร์ TCON กล่าวคือ เมื่อ C/T' = '0' จะใช้สัญญาณนาฬิกาจากภายใน (*ความถี่ออสซิลเลเตอร์/12*) และเมื่อ C/T' = '1' จะใช้สัญญาณนาฬิกาจากภายนอกที่ที่รับเข้ามาทางขา T0, 1 (P3.4, P3.5) แต่สัญญาณนาฬิกาที่เลือกนี้จะยังไม่สามารถป้อนให้กับวงจรนับได้ ถ้าหากบิต TR0,1 ในรีจิสเตอร์ TCON ยังคงมีค่าเท่ากับ '0' อยู่ การที่จะทำให้มีสัญญาณนาฬิกาป้อนไปยังวงจรนับได้หรือทำให้ไทเมอร์/เคาน์เตอร์ทำงานนั้นสามารถทำได้โดยกำหนดให้บิต Gate ในรีจิสเตอร์ TMOD = '0' หรือให้ลอจิกที่ขา INT0, 1' (P3.2, P3.3) = '1' ตอนนี้ไทเมอร์/เคาน์เตอร์พร้อมที่จะทำงานแล้ว และจะเริ่มทำงานหรือเริ่มนับทันทีเมื่อกำหนดให้ TR0,1 ในรีจิสเตอร์ TCON = '1'



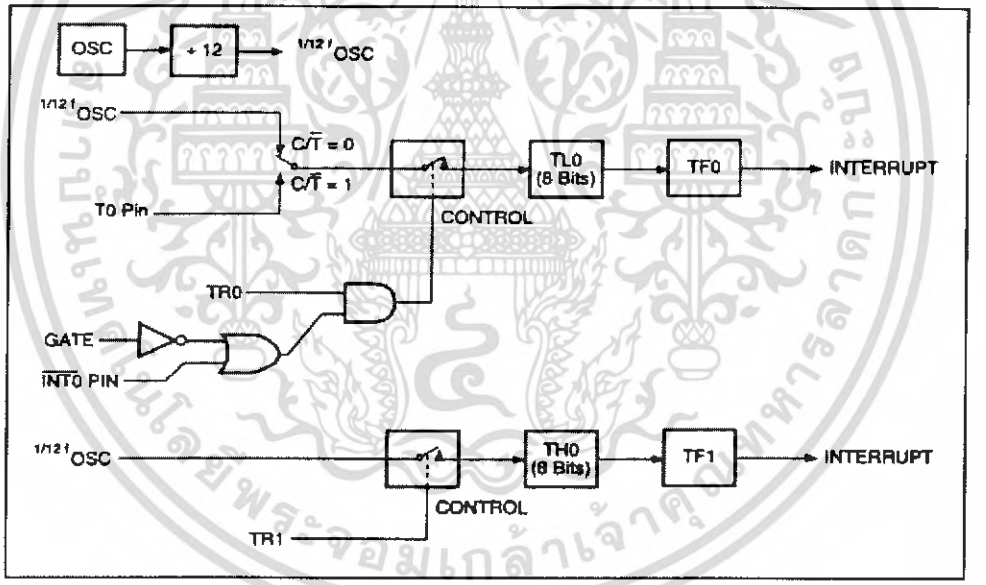
รูปที่ 2.17 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 2
(ไทเมอร์/เคาน์เตอร์ 8 บิตรีโหลดค่าอัตโนมัติ)

เมื่อไทเมอร์/เคาน์เตอร์เริ่มทำงานนับวงจรภายในจะเริ่มนับขึ้นไปเรื่อย ๆ และ เมื่อนับถึงค่าสูงสุด (FFH) แล้วจะวนกลับมาเริ่มที่ 00H ใหม่อีกครั้งซึ่งจังหวะนี้เรียกว่าเกิดการโอเวอร์โฟลว สัญญาณโอเวอร์โฟลวนี้จะทำให้เกิดการรีโหลด โดยจะทำการโหลดค่าภายใน TH0, 1 ไปยัง TL0, 1 โดยที่ค่าของ TH0, 1 ไม่เปลี่ยนแปลง ในขณะเดียวกันบิต TF0,1 ในรีจิสเตอร์ TCON เกิดการเซต และจะถูกเคลียร์เมื่อซีพียูกระโดดไปทำยังโปรแกรมอินเตอร์รัพต์ 1, 3 ซึ่งเป็นตำแหน่งของการอินเตอร์รัพต์ของไทเมอร์/เคาน์เตอร์ 0, 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์/เคาน์เตอร์ 0, 1 โหมด 3 (ไทเมอร์/เคาน์เตอร์ 8 บิต แบบแยกออกจากกัน)

การทำงานของไทเมอร์/เคาน์เตอร์ 0, 1 ในโหมด 3 จะแยกออกเป็นสองตัวและการทำงานจะเป็นอิสระต่อกัน ตัวแรกจะใช้ TL0 ซึ่งจะมีการทำงานเหมือนกันกับ การทำงานของไทเมอร์/เคาน์เตอร์ 0 ในโหมด 2 แต่จะไม่มีการรีโหลดค่า ส่วนตัวที่สองจะใช้ TH0 ซึ่งจะรับสัญญาณนาฬิกาภายในเพียงอย่างเดียว และ สามารถควบคุมการปิด/เปิดการทำงานได้จากบิต TR1 ในรีจิสเตอร์ TCON เมื่อนับถึงค่าสูงสุด(FFH) แล้ว (ทั้ง TL0 และ TH0) จะวนกลับมาเริ่มที่ 00H ใหม่อีกครั้งซึ่งจังหวะนี้เรียกว่าเกิดการโอเวอร์โฟลว สัญญาณโอเวอร์โฟลวที่ส่งออกจาก TL0 จะทำให้บิต TF0 เกิดการเซต และ สัญญาณโอเวอร์โฟลวที่ส่งออกจาก TH0 จะทำให้บิต TF1 เกิดการเซต ซึ่งทั้ง TF0 และ TF1 อยู่ในรีจิสเตอร์ TCON หากมีการเปิดใช้อินเตอร์รัพต์บิตนี้จะถูกเคลียร์โดยอัตโนมัติเมื่อซีพียูกระโดดไปทำงานยังโปรแกรมอินเตอร์รัพต์ 1 และอินเตอร์รัพต์ 3 ซึ่งเป็นตำแหน่งของการอินเตอร์รัพต์ของไทเมอร์/เคาน์เตอร์ 0 และ 1



รูปที่ 2.18 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 0 โหมด 3
(ไทเมอร์/เคาน์เตอร์ 8 บิต แบบแยกออกจากกัน)

การเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ ไทเมอร์/เคาน์เตอร์ 1

การเลือกโหมดการทำงานของ Timer 0 และ Timer 1 จำเป็นต้องกำหนดค่าให้กับรีจิสเตอร์ TMOD ให้ถูกต้อง ซึ่งรีจิสเตอร์เป็นรีจิสเตอร์ TMOD ขนาด 8 บิต โดยจะใช้ 4 บิตล่างในการเลือกโหมดการทำงานของ Timer 0 และใช้ 4 บิตบนในการเลือกโหมดการทำงานของ Timer 1 สำหรับรายละเอียดกำหนดค่าในรีจิสเตอร์ TMOD ที่ใช้เลือกโหมดการทำงานของ Timer 0 และ Timer 1 สามารถสรุปได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GATE	C/T'	M1	M0	GATE	C/T'	M1	M0
ไทเมอร์/เคาน์เตอร์ 1				ไทเมอร์/เคาน์เตอร์ 0			

T2MOD

การใช้งานไทเมอร์/เคาน์เตอร์ 0 เป็นไทเมอร์

- การควบคุมจากภายใน : ควบคุมการปิด/เปิดของไทเมอร์โดยการกำหนดบิตให้ TR0 ในรีจิสเตอร์ TCON เป็นลอจิก '0' และลอจิก '1' ตามลำดับ

- การควบคุมจากภายนอก : ควบคุมการปิด/เปิดของไทเมอร์โดยการกำหนดบิตให้สัญญาณลอจิกที่ขา INT0'(P3.2) เป็น '1' ในขณะที่บิต TR0 ในรีจิสเตอร์ TCON มีสถานะเป็นลอจิก '1'

โหมด	ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 0	ค่าในรีจิสเตอร์ TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทเมอร์ขนาด 13 บิต	0×?0	0×?8
1	ไทเมอร์ขนาด 16 บิต	0×?1	0×?9
2	ไทเมอร์ขนาด 8 บิต รีโหลดค่าอัตโนมัติ	0×?2	0×?A
3	ไทเมอร์ขนาด 8 บิต ทำงานแบบแยกส่วน	0×33	0×3B

ตารางที่ 2.5 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 0 เป็นไทเมอร์

การใช้งานไทเมอร์/เคาน์เตอร์ 0 เป็นเคาน์เตอร์

- การควบคุมจากภายใน : ควบคุมการปิด/เปิดของเคาน์เตอร์โดยการกำหนดบิตให้ TR0 ในรีจิสเตอร์ TCON เป็น ลอจิก '0' และลอจิก '1' ตามลำดับ

- การควบคุมจากภายนอก : ควบคุมการปิด/เปิดของเคาน์เตอร์โดยการกำหนดบิตให้สัญญาณลอจิกที่ขา INT0' (P3.2) เป็น '1' ในขณะที่บิต TR0 ในรีจิสเตอร์ TCON มีสถานะเป็นลอจิก '1'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด	ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 0	ค่าในรีจิสเตอร์ TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทเมอร์ขนาด 13 บิต	0X?4	0X?C
1	ไทเมอร์ขนาด 16 บิต	0X?5	0X?D
2	ไทเมอร์ขนาด 8 บิต รีโหลดค่าอัตโนมัติ	0X?6	0X?E
3	ไทเมอร์ขนาด 8 บิต ทำงานแบบแยกส่วน	0X37	0X3F

ตารางที่ 2.6 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 0 เป็นเคาน์เตอร์

การใช้งานไทเมอร์/เคาน์เตอร์ 1 เป็นไทเมอร์

- การควบคุมจากภายใน : ควบคุมการปิด/เปิดของไทเมอร์โดยการกำหนดบิตให้ TR1 ในรีจิสเตอร์ TCON เป็นลอจิก '0' และลอจิก '1' ตามลำดับ

- การควบคุมจากภายนอก : ควบคุมการปิด/เปิดของไทเมอร์โดยการกำหนดบิตให้สัญญาณลอจิกที่ขา INT1'(P3.3) เป็น '1' ในขณะที่บิต TR0 ในรีจิสเตอร์ TCON มีสถานะเป็นลอจิก '1'

โหมด	ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 0	ค่าในรีจิสเตอร์ TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทเมอร์ขนาด 13 บิต	0X0?	0X8?
1	ไทเมอร์ขนาด 16 บิต	0X1?	0X9?
2	ไทเมอร์ขนาด 8 บิต รีโหลดค่าอัตโนมัติ	0X2?	0XA?
3	หยุดทำงาน (Disable)	0X3?	0XB?

ตารางที่ 2.7 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 1 เป็นไทเมอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานไทเมอร์/เคาน์เตอร์ 1 เป็นเคาน์เตอร์

- การควบคุมจากภายใน : ควบคุมการปิด/เปิดของเคาน์เตอร์โดยการกำหนดบิตให้ TR1 ในรีจิสเตอร์ TCON เป็น ลอจิก '0' และลอจิก '1' ตามลำดับ

- การควบคุมจากภายนอก : ควบคุมการปิด/เปิดของเคาน์เตอร์โดยการกำหนดบิตให้สัญญาณลอจิกที่ขา INT1' (P3.3) เป็น '1' ในขณะที่บิต TR0 ในรีจิสเตอร์ TCON มีสถานะเป็นลอจิก '1'

โหมด	ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 0	ค่าในรีจิสเตอร์ TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทเมอร์ขนาด 13 บิต	0X4?	0XC?
1	ไทเมอร์ขนาด 16 บิต	0X5?	0XD?
2	ไทเมอร์ขนาด 8 บิต รีโหลดค่าอัตโนมัติ	0X6?	0XE?
3	หยุดทำงาน (Disable)	0X7?	0XF?

ตารางที่ 2.8 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 1 เป็นเคาน์เตอร์

2.ไทเมอร์/เคาน์เตอร์ 2

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในไมโครคอนโทรลเลอร์ MCS-51 มีพื้นฐานเหมือนกันกับไทเมอร์/เคาน์เตอร์ 0 และ 1 แต่มีความสามารถมากกว่าไทเมอร์/เคาน์เตอร์ 0 และไทเมอร์/เคาน์เตอร์ 1 หลายอย่าง เช่น สามารถทำงานในโหมด 16 บิตรีโหลดค่าอัตโนมัติ และทำงานในโหมดแคปเจอร์ได้เป็นต้น

รีจิสเตอร์ของไทเมอร์/เคาน์เตอร์ 2 (Timer/Counter 2 Register)

รีจิสเตอร์ของไทเมอร์/เคาน์เตอร์ 2 ทำหน้าที่ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 2 ซึ่งมีทั้งหมด 4 ตัวคือ RCAP2L, RCAP2H, TL2, TH2 ซึ่ง RCAP2L และ RCAP2H จะถูกใช้ในกรณีที่ไทเมอร์ 2 ทำงานในโหมด 16-bit capture 16-bit auto-reload

- RCAP2L ทำหน้าที่ในการเก็บค่าในการแคปเจอร์ไบต์ต่ำหรือเก็บค่ารีโหลดไบต์ต่ำ ทั้งนี้ขึ้นอยู่กับโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2 มีแอดเดรสอยู่ที่ CAH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- RCAP2H ทำหน้าที่ในการเก็บค่าในการแคปเจอร์ไบต์สูง หรือเก็บค่ารีโหลดไบต์สูง ทั้งนี้ขึ้นอยู่กับโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2 มีแอดเดรสอยู่ที่ CBH

- TL2 เป็นรีจิสเตอร์ไบต์ต่ำของไทเมอร์ 2 มีแอดเดรสอยู่ที่ CCH

- TH2 เป็นรีจิสเตอร์ไบต์สูงของไทเมอร์ 2 มีแอดเดรสอยู่ที่ CDH

รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 2

(Timer/Counter 2 Control Register : T2CON)

T2CON ทำหน้าที่สำหรับควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 2 มีแอดเดรสอยู่ที่ C8H รายละเอียดการทำงานของ T2CON มีดังนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2'	CP/RL2'

T2CON

- TF2 (Timer 2 overflow flag) จะเซตเมื่อไทเมอร์ 2 เกิดการนับเกิน (overflow) และสามารถเคลียร์ได้โดยกระบวนการทางซอฟต์แวร์ TF2 จะ ไม่เกิดการเซตถ้าหาก RCLK = '1' หรือ TCLK = '1'

- EXF2 (Timer 2 external flag) จะเซตเมื่อมีการแคปเจอร์รีโหลดเกิดขึ้น นั่นคือเมื่อ EXEN2 = '1' และมีสัญญาณขอบขาลงเกิดขึ้นที่ขา T2EX และถ้ามีการกำหนดให้ไทเมอร์ 2 มีการอินเตอร์รัพต์ และบิต EXF2 = '1' จะทำให้ซีพียูกระโดดไปทำงานในส่วนของโปรแกรมการบริการการอินเตอร์รัพต์ บิต EXF2 สามารถเคลียร์ได้โดยกระบวนการทางซอฟต์แวร์ สำหรับการใช้งานในโหมดเคาน์เตอร์ (DCEN ='1') บิต EXF2 นี้จะไม่ส่งผลให้ไทเมอร์/เคาน์เตอร์ 2 เกิดการอินเตอร์รัพต์

- RCLK (Receive clock enable) เมื่อบิตนี้ถูกเซตจะทำให้พอร์ตอนุกรมที่ทำงานในโหมด 1 และ โหมด 3 ใช้สัญญาณพัลส์ที่เกิดจากการโอเวอร์โฟลวของไทเมอร์ 2 สำหรับการควบคุมการรับข้อมูลเข้ามาทางพอร์ตอนุกรมแต่ละ RCLK = '0' จะใช้สัญญาณพัลส์ที่เกิดจากการโอเวอร์โฟลวของไทเมอร์ 1 สำหรับการควบคุมการส่งข้อมูล

- TCLK (*Transmit clock enable*) เมื่อบิตนี้ถูกเซตจะทำให้พอร์ตอนุกรมใช้สัญญาณพัลส์ที่เกิดจากการโอเวอร์โฟลวของไทเมอร์ 2 สำหรับการควบคุมการส่งข้อมูลออกไป แต่ถ้า TCLK = '0' จะใช้สัญญาณพัลส์ที่เกิดจากการโอเวอร์โฟลวของไทเมอร์ 1 สำหรับการควบคุมการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- EXEN2 (Timer 2 external enable) เมื่อบิตนี้ถูกเซตจะยอมให้มีการแคปเจอร์หรือการรีโหลดโดยรับสัญญาณขอบขาลงที่ขา T2EX แต่การแคปเจอร์หรือการรีโหลดนี้จะเกิดขึ้นได้ก็ต่อเมื่อไทเมอร์ 2 ไม่ได้ถูกใช้โดยพอร์ตอนุกรม (RCLK ='0' และ TCLK = "0") และถ้าบิต EXEN2 = '0' ไทเมอร์ 2 จะไม่ยอมตอบสนองต่อสัญญาณที่เข้ามายังขา T2EX

- TR2 (Start/Stop control for Timer 2) เมื่อบิต TR2 = '1' ไทเมอร์ 2 จะเริ่มทำงานและเมื่อบิต TR2 = '0' ไทเมอร์ 2 จะไม่ทำงาน

- C/T2' (Timer or Counter select for Timer 2) เมื่อ C/T2 = '0' จะทำงานในโหมดไทเมอร์ (รับสัญญาณนาฬิกาจากภายใน) แต่ถ้า C/T2 = '1' จะทำงานในโหมดเคาน์เตอร์ (รับสัญญาณนาฬิกาจากภายนอก)

- CP/RL2' (Capture/Reload select) เมื่อ CP/RL2 = '1' จะทำให้เกิดการแคปเจอร์ เมื่อมีสัญญาณขอบขาลงเข้ามาที่ขา T2EX แต่ถ้า CP/RL2 = '0' จะทำให้เกิดการรีโหลด เมื่อไทเมอร์ 2 เกิดการโอเวอร์โฟลว หรือเมื่อมีสัญญาณขอบขาลงเข้ามาที่ขา T2EX ในขณะที่ EXEN2 = '1' แต่ถ้า RCLK หรือ TCLK = '1' CP/RL2' จะไม่ตอบสนองต่อการแคปเจอร์และการรีโหลด

รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 2

(Timer/Counter 2 Mode Control Register: T2MOD)

T2MOD ทำหน้าที่สำหรับเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2 มีแอดเดรสอยู่ที่ C9H รายละเอียดการทำงานของ T2MOD มีดังนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	-	-	T2OE	DCEN

T2MOD

T2OE (Timer 2 Output Enable) ทำหน้าที่เปิด/ปิด (Enable/Disable) เอาต์พุตของไทเมอร์/เคาน์เตอร์ 2 โดยเมื่อ T2OE = '1' จะเป็นการ Enable และ เมื่อ T2OE = '0' จะเป็นการ Disable

DCEN (Timer 2 Direction Counter Enable) ทำหน้าที่เปิด/ปิด การทำงานของไทเมอร์/เคาน์เตอร์ 2 ให้สามารถทำงานในโหมดนับขึ้นนับลงได้ โดยเมื่อ DCEN = '1' จะเป็นการ Enable และเมื่อ DCEN = '0' จะเป็นการ Disable

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของไทเมอร์/เคาน์เตอร์ในโหมดต่าง ๆ

ไทเมอร์/เคาน์เตอร์ 2 มีโหมดการทำงาน 3 โหมด คือโหมด16 บิตแคปเจอร์, โหมด16 บิตรีโหลดอัตโนมัติ และโหมดกำหนดอัตราเร็วการรับส่งข้อมูลของพอร์ตอนุกรม (bund rate generator) โดยการเลือกโหมดต่าง ๆ เหล่านี้สามารถเลือกได้โดยการกำหนดค่ารีจิสเตอร์ T2CON ซึ่งมีรายละเอียดดังตารางที่ 2.9

RCLK + TCLK	CP/RL2'	TR2	MODE
0	0	1	16 บิตรีโหลดค่าแบบอัตโนมัติ
0	1	1	16 บิตแคปเจอร์
1	×	1	กำหนดอัตราเร็วการรับส่งข้อมูลของพอร์ตอนุกรม
×	×	0	หยุดการทำงาน (Disable)

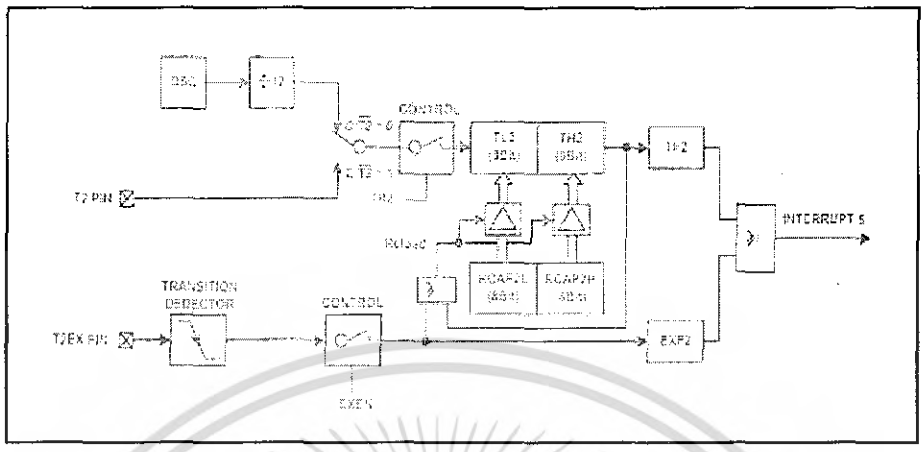
ตารางที่ 2.9 แสดงการกำหนดโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 16 บิตรีโหลดค่าแบบอัตโนมัติ (กรณี DCEN = '0')

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 0 กรณีที่ DCEN = '0' จะคล้ายกับการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 แต่กรณีรีโหลอดของไทเมอร์/เคาน์เตอร์ 2 นั้นจะทำการโหลดค่าแบบ 16 บิต โดยจะทำการโหลดค่าจากรีจิสเตอร์ RCAP2L และ RCAP2H ไปยังรีจิสเตอร์ T2L และ T2H การรีโหลดของไทเมอร์/เคาน์เตอร์จะเกิดขึ้นได้ 2 กรณี คือเมื่อบิต EXEN2 เป็น '0' จะรีโหลดเมื่อเกิดโอเวอร์โฟลว และในกรณีที่ EXEN2 เป็น '1' จะสามารถรีโหลดได้จากสัญญาณขอบขาลง เกิดขึ้นที่ขา T2EX ได้อีกด้วย

ผลจากการรีโหลดทั้งสองกรณีจะสามารถทำให้เกิดการอินเตอร์รัพต์ขึ้น หากมีการอีนาเบิลการอินเตอร์รัพต์จากไทเมอร์ 2 เอาไว้ (ET2 = '1' และ EA2 = '1') การอินเตอร์รัพต์จะทำให้ซีพียูกระโดดไปประมวลผลในตำแหน่งบริการการอินเดอร์รัพต์ ของไทเมอร์ 2 และ หลังจากที่ซีพียูประมวลผลคำสั่งในส่วนของการอินเตอร์รัพต์เสร็จสิ้น TF2 จะไม่เคลียร์เองแบบอัตโนมัติเหมือน TF0 และ TF1 ดังนั้นการเคลียร์บิต TF2 และ EXF2 ต้องทำโดยกระบวนการทางซอฟต์แวร์ สำหรับไดอะแกรมการทำงานในโหมด 16 บิต รีโหลดค่าแบบอัตโนมัติในกรณีที่ DCEN = '0' ซึ่งแสดงในรูปที่ 2.19

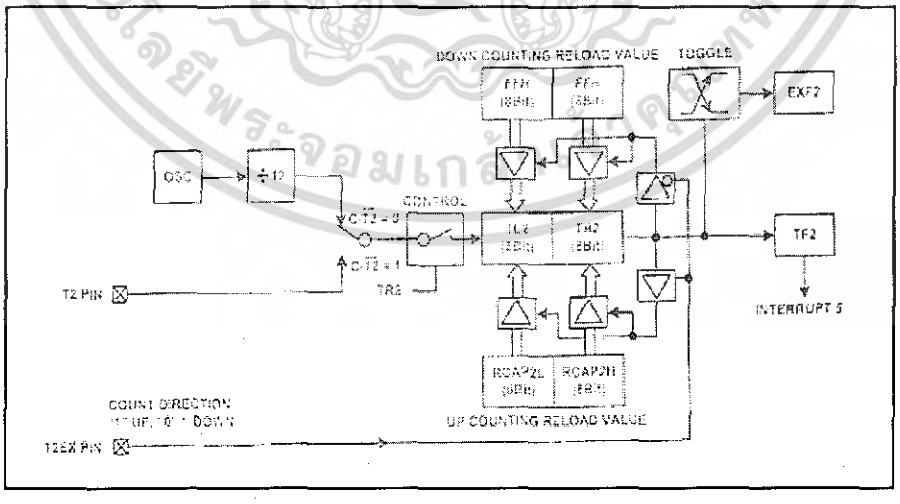
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2
ในโหมด 16 บิตรีโหลดค่าแบบอัตโนมัติ (กรณี DCEN = '0')

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 16 บิตรีโหลดค่าแบบอัตโนมัติ (กรณี DCEN = '1')

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 16 บิต รีโหลดค่าแบบอัตโนมัติกรณี DCEN = '1' จะเกิดการรีโหลดเมื่อไทเมอร์/เคาน์เตอร์นับจนเกิดการโอเวอร์โฟลว และการรีโหลดไม่ได้รีโหลดค่ามาจากรีจิสเตอร์ RCAP2L และ RCAP2H เท่านั้น แต่ยังสามารถรีโหลดค่ามาจากวงจรภายใน (FFFFH) การรีโหลดดังกล่าวสามารถเลือกได้จากการป้อนสัญญาณให้กับขา T2EX เมื่อ T2XE = '1' ไทเมอร์ 2 จะนับขึ้นและทำการรีโหลดจากรีจิสเตอร์ RCAP2L และ RCAP2H



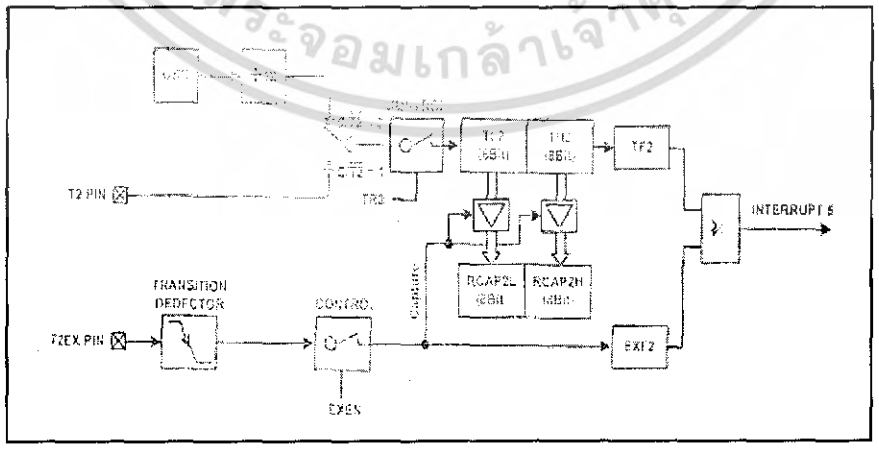
รูปที่ 2.20 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2
ในโหมด 16 บิตรีโหลดค่าแบบอัตโนมัติ (กรณี DCEN = '1')

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการรีโหลดเกิดขึ้นจะทำให้ TF2 และ TF1 เกิดการเซตและถ้าได้มีการอีนาเบิลการอินเตอร์รัพต์ไว้ (ET2 = '1' และ EA2 = '1') ซีพียูจะกระโดดไปประมวลผลในตำแหน่งบริการการอินเตอร์รัพต์ของไทเมอร์ 2 และหลังจากที่ซีพียูประมวลผลคำสั่งในส่วนของการอินเตอร์รัพต์เสร็จสิ้น TF2 จะไม่เคลียร์เองแบบอัตโนมัติเหมือน TF0 และ TF1 ดังนั้นการเคลียร์บิต TF2 และ EXF2 ต้องทำโดยกระบวนการทางซอฟต์แวร์ สำหรับไดอะแกรมการทำงานในโหมด 16 บิตรีโหลดค่าแบบอัตโนมัติกรณีที่ DCEN = '1' แสดงในรูปที่ 2.20

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 1 (16 บิต แคปเจอร์)

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมด 16 บิต แคปเจอร์คือจะนำค่าที่อยู่ใน TL2 และ TH2 เข้าไปเก็บไว้ในรีจิสเตอร์ RCAP2L และ RCAP2H โดยที่การนับของไทเมอร์/เคาน์เตอร์ 2 ยังคงทำการนับต่อไป การที่แคปเจอร์จะเกิดขึ้นได้ก็ต่อเมื่อมีบิต EXEN = '1' และมีสัญญาณขอบขาลง (เปลี่ยนจากลอจิก '1' เป็นลอจิก '0') ที่ขา T2EX (P1.1) การทำงานในโหมด 1 นี้สามารถสร้างอินเตอร์รัพต์ได้ 2 กรณี คือเมื่อมีการนับค่าจนเกิดโอเวอร์โฟลว (TF2 ='1') และเมื่อมีการแคปเจอร์ (EXF2 ='1') และ ถ้าได้มีการ Enable การอินเตอร์รัพต์ของไทเมอร์/เคาน์เตอร์ 2 เอาไว้ (ET2 = '1' และ EA = '1') ซีพียูจะกระโดดไปประมวลผลในตำแหน่งบริการการอินเตอร์รัพต์ของไทเมอร์ 2 ดังนั้นในโปรแกรมบริการการอินเตอร์รัพต์ ต้องทำการเขียนโปรแกรมเพื่อทำการตรวจสอบว่าการอินเตอร์รัพต์ครั้งนี้เกิดจากการโอเวอร์โฟลว หรือ เกิดจากการแคปเจอร์ซึ่งสามารถทำได้โดยการตรวจสอบที่บิต TF2 หรือ บิต EXF2 และหลังจากที่ซีพียูประมวลผลในส่วนของการอินเตอร์รัพต์เสร็จสิ้น TF2 และ EXF2 จะไม่เคลียร์เองโดยอัตโนมัติต้องทำการเขียนโปรแกรมเพื่อเคลียร์สองบิตนี้เอง หรือที่เรียกว่าเคลียร์โดยกระบวนการทางซอฟต์แวร์นั่นเอง



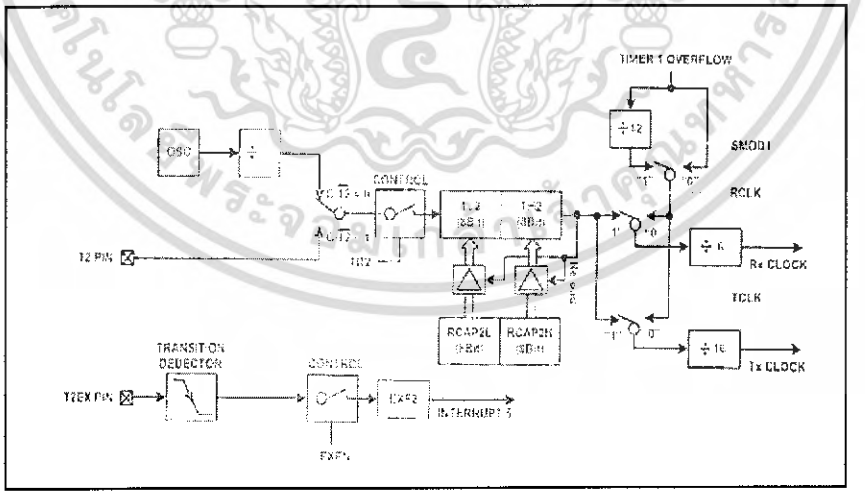
รูปที่ 2.21 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2 โหมด 16 บิต แคปเจอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของไทเมอร์/เคาน์เตอร์ 2 ในโหมดกำหนดอัตราเร็วในการรับส่งข้อมูลข้อพอร์ตอนุกรม

โดยปกติการกำหนดอัตราเร็วในการรับ/ส่งข้อมูล ของพอร์ตอนุกรมในไมโครคอนโทรลเลอร์ MCS-51 นั้นจะใช้ไทเมอร์ 1 แต่สามารถใช้ไทเมอร์/เคาน์เตอร์ 2 ในโหมด 2 มาทำหน้าที่เช่นนี้ได้เช่นกัน การกำหนดอัตราในการรับ/ส่งข้อมูลของพอร์ตอนุกรมนั้นจะใช้สัญญาณโอเวอร์โฟลวที่เกิดจากไทเมอร์/เคาน์เตอร์นั่นหมายความว่าถ้าความถี่ในการโอเวอร์โฟลวมีค่าสูงขึ้น อัตราเร็วในการรับ/ส่งข้อมูลของพอร์ตอนุกรมก็มีค่าสูงขึ้นตามไปด้วยเช่นกัน การใช้งานไทเมอร์/เคาน์เตอร์ 2 เป็นตัวกำหนดอัตราเร็วในการรับ/ส่งข้อมูลของพอร์ตอนุกรมสามารถทำได้โดยการให้บิต RCLK เป็น '1' เมื่อต้องการใช้งานไทเมอร์/เคาน์เตอร์ 2 เป็นตัวกำหนดอัตราเร็วในการรับข้อมูล, ให้บิต TCLK เป็น '1' เมื่อต้องการใช้งานไทเมอร์/เคาน์เตอร์ 2 เป็นตัวกำหนดอัตราเร็วในการส่งข้อมูล และสำหรับความเร็วในการรับ/ส่ง สามารถควบคุมได้โดยการกำหนดค่าให้กับรีจิสเตอร์ RCAP2L และ RCAP2H

เมื่อใช้ไทเมอร์/เคาน์เตอร์ 2 ในโหมดการกำหนดอัตราเร็วในการรับ/ส่งข้อมูลของพอร์ตอนุกรม และยังสามารถใช้ประโยชน์การอินเตอร์รัพต์ของไทเมอร์ 2 (INTERRUPT 5) ได้ตามปกติ แต่การอินเตอร์รัพต์ จะเกิดขึ้นในกรณีที่ให้บิต EXEN เป็น '1' และ มีสัญญาณขอบขาลงเข้ามาที่ขา T2EX แต่ในกรณีที่ไทเมอร์/เคาน์เตอร์ 2 เกิดการโอเวอร์โฟลวจะไม่มีการสร้างสัญญาณการอินเตอร์รัพต์ การทำงานของไทเมอร์/เคาน์เตอร์ 2 สามารถดูได้ในรูปที่ 2.22



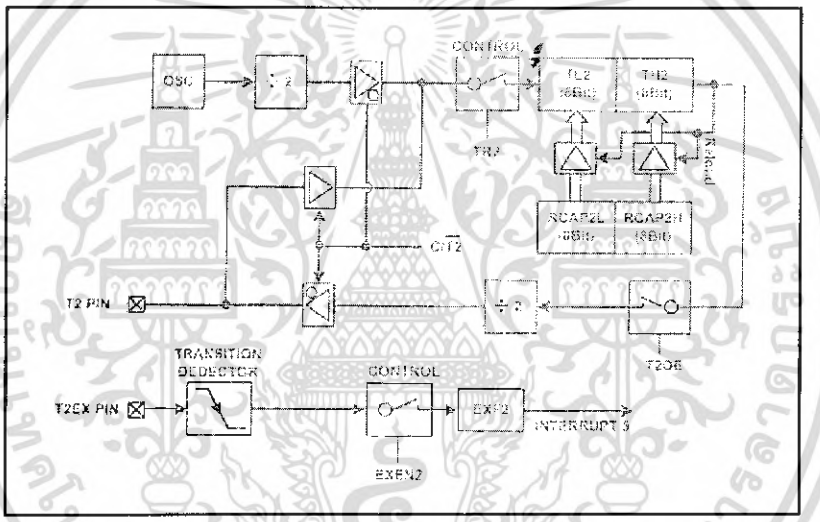
รูปที่ 2.22 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2
ในโหมดกำหนดอัตราเร็วในการรับส่งข้อมูลข้อพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของไทเมอร์/เคาน์เตอร์ 2 โหมดการสร้างสัญญาณพัลซ์ดิวตี้ไซเคิล 50% (C/T2' ="0")

การทำงานของไทเมอร์/เคาน์เตอร์ 2 โดยในโหมดการสร้างสัญญาณพัลส์ที่มีขนาดของดิวตี้ไซเคิล 50% ได้ โดยการกำหนดให้บิต C/T2' เป็นลอจิก '0' สัญญาณพัลส์ที่สร้างขึ้นจะส่งออกไปยังขา T2 (P1.0) เมื่อให้บิตT2OE เป็น '1' การเปลี่ยนแปลงความถี่ของสัญญาณพัลส์สามารถทำได้โดยการเปลี่ยนแปลงค่ารีโหลดใน รีจิสเตอร์ RCAP2L และ RCAP2H ซึ่งความถี่ที่ได้สามารถคำนวณได้ดังสูตรต่อไปนี้

$$\text{Clock-Out Frequency} = \frac{\text{Oscillator Frequency}}{4\times[65536-(\text{RCAP2H, RCAP2L})]}$$



รูปที่ 2.23 แสดงไดอะแกรมการทำงานของไทเมอร์/เคาน์เตอร์ 2
ในโหมดการสร้างสัญญาณพัลส์ดิวตี้ไซเคิล 50% (C/T2' ="0")

การเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2

การเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2 จำเป็นต้องกำหนดค่าให้กับรีจิสเตอร์ T2CON ให้ถูกต้อง ซึ่งรีจิสเตอร์ T2CON เป็นรีจิสเตอร์ที่ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ และมีรายละเอียดการกำหนดค่าในรีจิสเตอร์ T2CON สามารถสรุปได้ดังต่อไปนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2'	CP/RL2'

T2CON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานไทเมอร์/เคาน์เตอร์ 2 เป็นไทเมอร์

ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 2	ค่าในรีจิสเตอร์ TCON	
	การควบคุมจากภายใน	การควบคุมจากภายนอก
ไทเมอร์ขนาด 16 บิตรีโหลดค่าอัตโนมัติ	0X00	0X08
แคปเจอร์แบบ 16 บิต	0X01	0X09
กำหนด baud rate ในการรับและส่งเท่ากัน	0X34	0X36
กำหนด baud rate ในการรับเท่ากัน	0X24	0X26
กำหนด baud rate ในการส่งเท่ากัน	0X14	0X16

ตารางที่ 2.10 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 2 เป็นไทเมอร์

- การควบคุมจากภายใน : การรีโหลดและการแคปเจอร์จะเกิดขึ้นเมื่อไทเมอร์/เคาน์เตอร์ 2 เกิดการโอเวอร์โฟลว

- การควบคุมจากภายนอก : การรีโหลดและการแคปเจอร์จะเกิดขึ้นได้สองกรณี คือ เมื่อไทเมอร์/เคาน์เตอร์ 2 เกิดการโอเวอร์โฟลว และเมื่อมีสัญญาณขอบขาลงเข้ามาที่ขา T2EX (การรีโหลดและการแคปเจอร์จะไม่เกิดขึ้นถ้ามีการนำไทเมอร์/เคาน์เตอร์ 2 ไปใช้ในการควบคุมอัตราเร็วในการรับ/ส่งข้อมูลของพอร์ตอนุกรม)

การใช้งานไทเมอร์/เคาน์เตอร์ 2 เป็นเคาเตอร์

- การควบคุมจากภายใน : การรีโหลดและการแคปเจอร์จะเกิดขึ้นเมื่อไทเมอร์/เคาน์เตอร์ 2 เกิดการโอเวอร์โฟลว

ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 2	ค่าในรีจิสเตอร์ TCON	
	การควบคุมจากภายใน	การควบคุมจากภายนอก
ไทเมอร์ขนาด 16 บิตรีโหลดค่าอัตโนมัติ	0X02	0X0A
เคาเตอร์แบบแคปเจอร์แบบ 16 บิต	0X03	0X0B

ตารางที่ 2.11 แสดงการกำหนดค่าให้ไทเมอร์/เคาน์เตอร์ 2 เป็นเคาน์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การควบคุมจากภายนอก : การรีโหลดและการแคปเจอร์จะเกิดขึ้นได้สองกรณี คือ เมื่อไทเมอร์/เคาน์เตอร์ 2 เกิดการโอเวอร์โฟลว และเมื่อมีสัญญาณขอบขาลงเข้ามาที่ขา T2EX (การรีโหลดและการแคปเจอร์จะไม่เกิดขึ้นถ้ามีการนำไทเมอร์/เคาน์เตอร์ 2 ไปใช้ในการควบคุมอัตราเร็วในการรับ/ส่งข้อมูลของพอร์ตอนุกรม)

การเลือกรูปแบบการทำงานของไทเมอร์/เคาน์เตอร์ 2

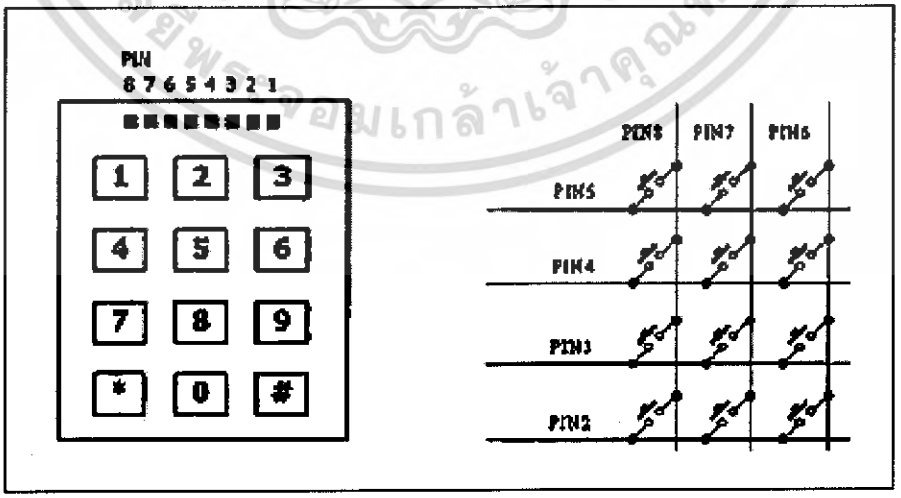
ฟังก์ชันของไทเมอร์/เคาน์เตอร์ 2	ค่าในรีจิสเตอร์ TMOD
สร้างสัญญาณพัลส์ส่งออกที่ขา T2 (P1.0)	T2OE = '1'
ไทเมอร์/เคาน์เตอร์ ทำงานแบบนับลง	DCEN = '1'

ตารางที่ 2.12 แสดงการเลือกรูปแบบการทำงานของไทเมอร์/เคาน์เตอร์ 2

2.4 การต่อคีย์สวิตช์แบบเมทริกซ์หรือคีย์แพดกับไมโครคอนโทรลเลอร์

โดยปกติการต่อคีย์สวิตช์เข้ากับพอร์ตของไมโครคอนโทรลเลอร์ คีย์สวิตช์หนึ่งตัวจะต้องต่อกับพอร์ตหนึ่งบิต ถ้าหากต้องการต่อคีย์สวิตช์เป็นจำนวนมากจะต้องใช้พอร์ตหลายพอร์ต แต่เราสามารถลดจำนวนพอร์ตได้โดยใช้การต่อคีย์สวิตช์แบบเมทริกซ์แทน

หลักการในการคำนวณเส้นสัญญาณที่ต้องใช้คือ คำนวณจากขนาดของ M×N โดย M คือ จำนวนแถว (row) และ N คือ จำนวนหลักหรือคอลัมน์ (column) โดยจำนวนเส้นสัญญาณที่ต้องใช้คือจำนวนของผลรวมของจำนวนแถวและจำนวนหลัก (M+N)



รูปที่ 2.24 แสดงลักษณะของคีย์สวิตช์แบบเมทริกซ์ขนาด 4×3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการของโปรแกรมในการตรวจสอบสภาพการกดสวิตช์ จะใช้หลักการ keyboard scan โดยเริ่มต้นจะต้องทำให้สภาวะลอจิกของทุกเส้นสัญญาณในแนวคอลัมน์เป็น 1 ก่อน จากนั้นจึงเริ่มต้นกำหนดค่าลอจิกของเส้นสัญญาณในแถวแรก (row 0) เป็นลอจิก 0 เมื่อได้อ่านค่าข้อมูลของสัญญาณในแนวคอลัมน์เข้ามาตรวจสอบ หากพบว่ามีเส้นสัญญาณใดมีค่าลอจิกต่ำ แสดงว่ามีการกดสวิตช์ ที่ต่ออยู่ระหว่างคอลัมน์นั้นกับเส้นสัญญาณในแถวแรก ทำให้ช่องสัญญาณทั้งสองเชื่อมต่อเข้าด้วยกัน หากว่าไม่มีเส้นสัญญาณใดเป็นลอจิกต่ำเลยแสดงว่าไม่มีการกดสวิตช์ใด ๆ ในแถวแรกเลย จากนั้นก็ดำเนินการซ้ำเช่นเดียวกันกับเส้นสัญญาณในแถวอื่น ๆ อีกในรอบถัดไปจนกระทั่งครบทุกแถว

ในการนำคีย์สวิตช์แบบเมทริกต์ไปใช้งานจะต้องเพิ่มโปรแกรมย่อย ในการหน่วงเวลาชักระยะหนึ่งก่อนที่จะกลับมาวนรอบการอ่านสภาวะสวิตช์ในรอบใหม่ ในการใช้งานคีย์สวิตช์แบบเมทริกต์สภาวะความเป็นจริงแล้วลักษณะสมบัติเชิงกลของมัน คือ หลังจากการกดสวิตช์หนึ่งครั้งก็จะมีการกระเพื่อมของหน้าสัมผัส (Contract denounce) ทำให้ลักษณะเหมือนเกิดการเปิดและปิดหน้าสัมผัสหลาย ๆ ครั้ง ซึ่งมีผลให้โปรแกรมดำเนินการผิดพลาด เนื่องจากมองเห็นว่ามีการกดสวิตช์หลายครั้ง ดังนั้นจึงต้องทำการหน่วงเวลาโปรแกรมชักระยะหนึ่ง เพื่อเป็นการลดปัญหาการกระเพื่อมโดยกระบวนการทางด้านซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

ในส่วนของการออกแบบและการสร้างแบ่งออกได้เป็น 2 ส่วน คือ การออกแบบในส่วนของฮาร์ดแวร์ และ การออกแบบในส่วนของซอร์ฟแวร์

3.1 การออกแบบในส่วนของฮาร์ดแวร์

เนื่องจากบริษัทบริษัท เดโชโมลด์แอนด์ไดคาสติ้ง จำกัด ได้กำหนดให้มีการใช้บอร์ดแสดงผล รุ่น ET-DSP4 ของบริษัท อีทีที จำกัด ดังนั้นจึงได้นำบอร์ดแสดงผล รุ่น ET-DSP4 มาประยุกต์ใช้งาน โดยการเขียนโปรแกรมควบคุมการทำงาน โดยได้เพิ่มในส่วนของอินพุทที่เป็นสวิตช์ซึ่งใช้แทนการทำงานของเซนเซอร์ที่เป็นรีเลย์ที่ใช้นับค่าของจำนวนผลการผลิต

3.2 การออกแบบในส่วนของซอร์ฟแวร์

เป็นการออกแบบในการเขียนโปรแกรมโดยใช้โปรมแกรม Keil uVision3 เพื่อใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ ในโปรมแกรม Keil uVision3 ประกอบด้วย Toolbar สำหรับเครื่องมือที่จำเป็นในโหมด Edit มีดังนี้

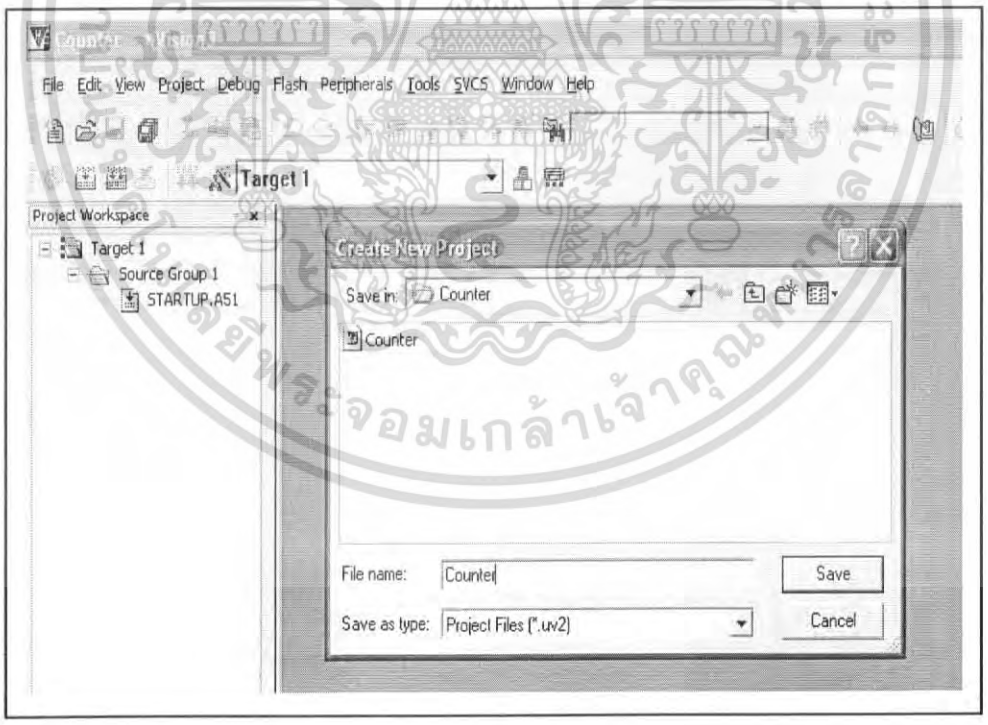
1. [New File] สร้างเอกสารใหม่ ซึ่งเป็นหน้าต่างที่ใช้สำหรับเขียนโปรแกรม
2. [Open] เปิดไฟล์ขึ้นมาเพื่อกระทำการต่างๆ เช่น แก้ไข หรือ คัดลอก
3. [Save] จัดเก็บไฟล์ที่อยู่ในปัจจุบัน
4. [Save all] จัดเก็บไฟล์ทั้งหมด
5. [Cut] ตัดข้อความที่ถูกเลือกไปเก็บไว้ยัง Clipboard
6. [Copy] คัดลอกข้อความถูกเลือก
7. [Paste] วางข้อความที่อยู่ใน Clipboard
8. [Undo] เปลี่ยนการแก้ไขกลับไปก่อนหน้า
9. [Redo] เปลี่ยนการแก้ไขกลับไปหลังจากนี้
10. [Toggle Bookmark] ใช้หรือยกเลิก Bookmark
11. [Next Bookmark] ให้ Cursor กระโดดไปยังตำแหน่งของ Bookmark ถัดไป
12. [Previous Bookmark] ให้ Cursor กระโดดไปยังตำแหน่งของ Bookmark ถัดก่อนหน้า
13. [Clear All Bookmark] ยกเลิก Bookmark ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14. [Source Browse Window] เปิดและปิดหน้าต่าง Source Browser Window
15. [Start/Stop Debug Session] สลับระหว่างโหมด Debug และโหมด Edit
16. [Project Window] เปิดและปิด หน้าต่าง Project Workspace
17. [Output Window] เปิดและปิด หน้าต่าง Output Window
18. [Build Target] สร้างไฟล์ต่างๆที่จำเป็นสำหรับโปรเจ็ค ซึ่งเป็นการ Compile ไฟล์ที่เขียนขึ้น
19. [Rebuild all Target Files] สร้างไฟล์ต่างๆที่จำเป็นสำหรับโปรเจ็ค ซึ่งเป็นการ Compile ไฟล์ที่เขียนขึ้นในโปรเจ็คทั้งหมด

การเขียนโปรแกรมโดยใช้โปรมแกรมเพื่อใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 มีขั้นตอนดังนี้

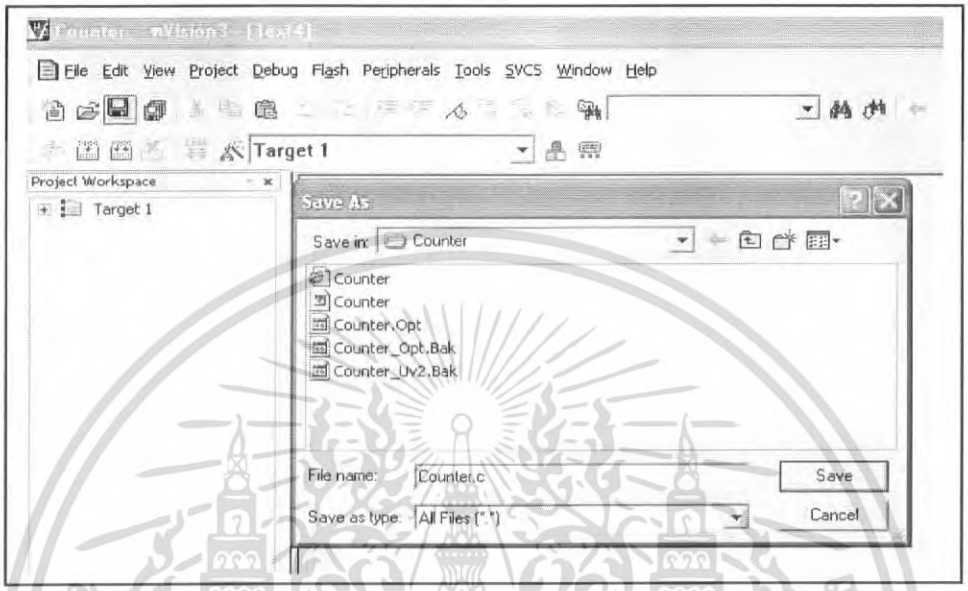
3.2.1. เริ่มสร้างโปรเจ็คใหม่โดยการคลิกที่ Project-> New Project และทำการ Save ชื่อโปรเจ็คเป็น Counter จะปรากฏหน้าต่างดังรูปที่ 3.1



รูปที่ 3.1 แสดงการสร้างโปรเจ็ค Counter

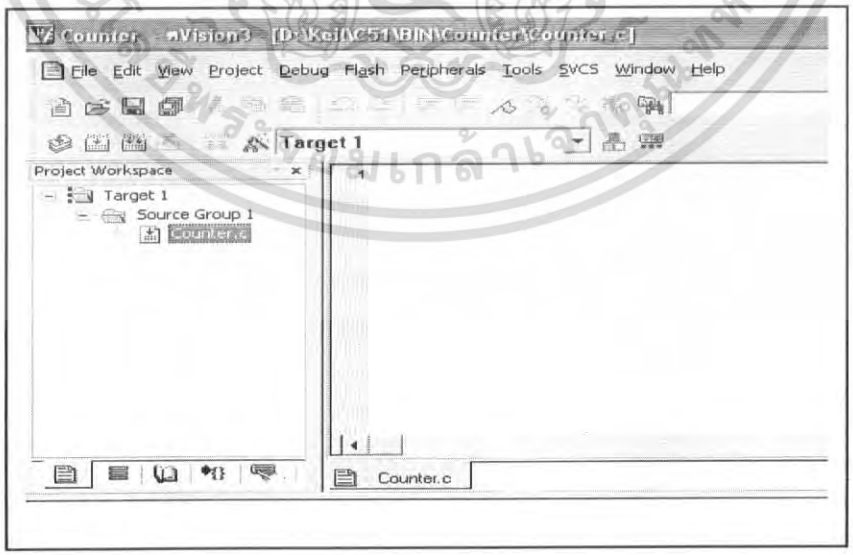
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2. ทำการเพิ่มไฟล์เข้าไปในยังโปรเจ็ค โดยไฟล์ที่จะใช้เพิ่มต้องเป็นไฟล์.c ดังนั้นให้ทำการสร้างไฟล์.c ก่อน และทำการ save เป็นไฟล์.c ดังรูปที่ 3.2



รูปที่ 3.2 แสดงการสร้างไฟล์ Counter.c

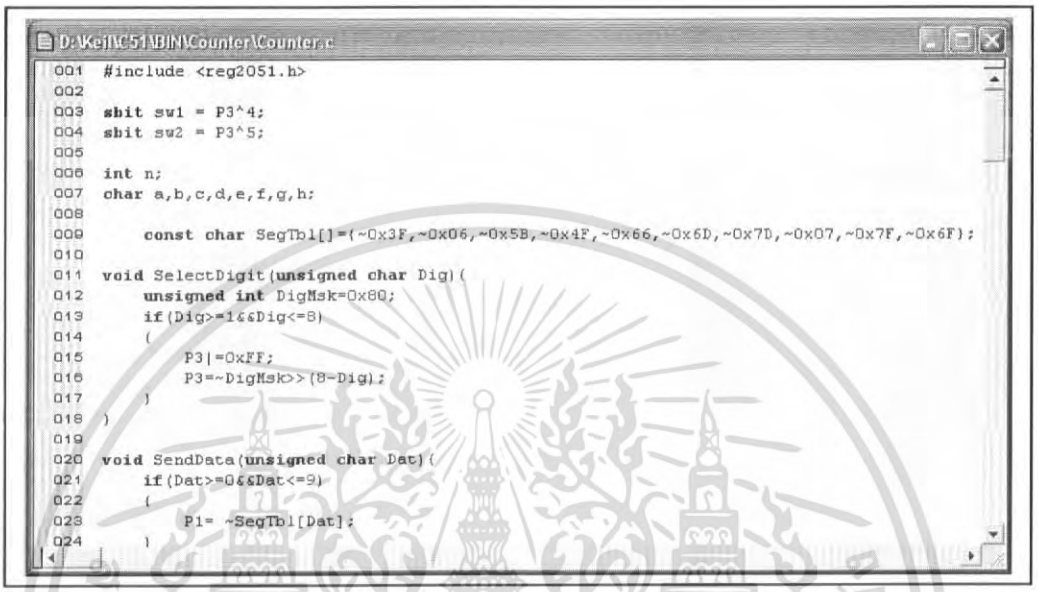
3.2.3 ทำการเพิ่มไฟล์เข้ามายังโปรเจ็คโดยการคลิกขวาที่ Source Group 1 แล้วเลือกที่ Add Files to Group 'Source Group' เมื่อทำการเพิ่มไฟล์เข้าไปยังโปรเจ็คเรียบร้อยแล้ว ซึ่งจะมีไฟล์ที่เราแอดเข้าไปปรากฏอยู่ ที่หน้าต่างProject Workspace ดังรูปที่ 3.3



รูปที่ 3.3 แสดงไฟล์ Counter.c ที่ทำการเพิ่มไฟล์เข้าไปยังโปรเจ็ค

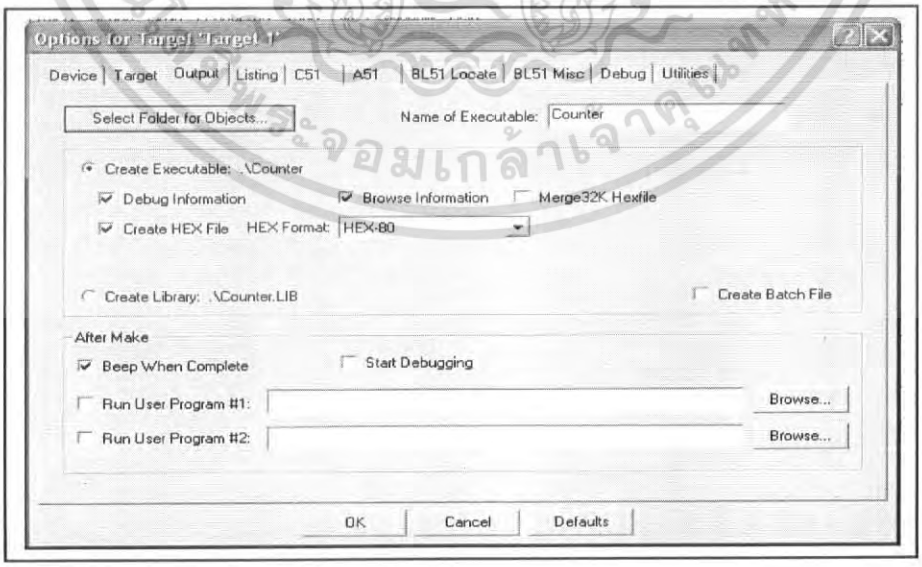
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 ทำการเขียน เขียนโปรแกรม โดยดับเบิลคลิกที่ไฟล์ Counter.cในรูปที่ 3.3 เพื่อเลือกหน้าต่างสำหรับเขียนโปรแกรมขึ้นมา ดังรูปที่ 3.4



รูปที่ 3.4 แสดงการเขียนโปรแกรม

3.2.5 ทำการ Build Target, Save ไฟล์ที่เขียนขึ้น และให้คอมไพล์เลอร์ทำการสร้างไฟล์นามสกุล.hex ขึ้นมาเพื่อใช้ทำการโปรแกรมลงในไมโครคอนโทรลเลอร์ ดังรูปที่ 3.5



รูปที่ 3.5 แสดงการสร้างไฟล์นามสกุล.hex

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

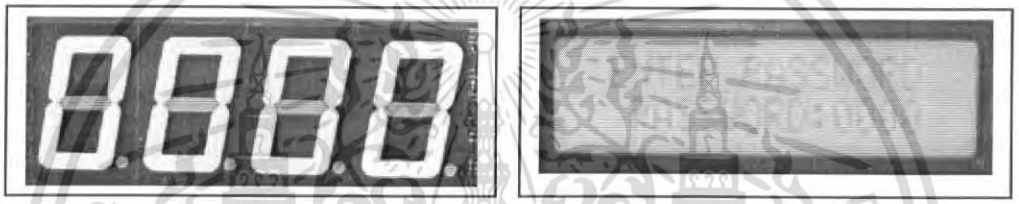
บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองและผลการทดลองในส่วนของการตั้งเป้าหมายทางคีย์แพ็ด

ในการทดลองในส่วนของการตั้งเป้าหมายทางคีย์แพ็ดนั้น เมื่อได้ทำการโปรแกรมไฟล์ลงไมโครคอนโทรลเลอร์ของบอร์ดแสดงผล ET-DSP4 แล้วทำการทดลองโปรแกรมปรากฏว่าได้ผลการทดลองตามการโปรแกรม ตามขั้นตอนดังนี้คือ

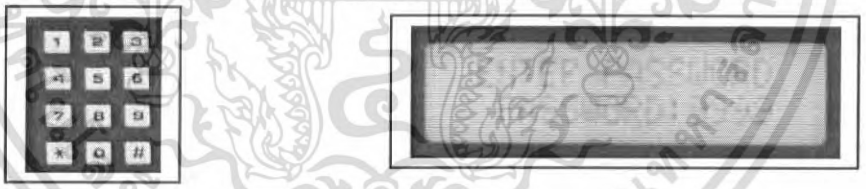
1. ป้ายนับจำนวนผลการผลิตจะแสดงค่าเป็น "0000" และที่หน้าจอ LCD จะแสดงข้อความให้ทำการใส่รหัสผ่าน (Password)



(a) การแสดงผลของป้ายนับจำนวนผลการผลิต (b) การแสดงผลของจอ LCD

รูปที่ 4.1 แสดงการแสดงผลของป้ายนับจำนวนผลการผลิต และ LCD

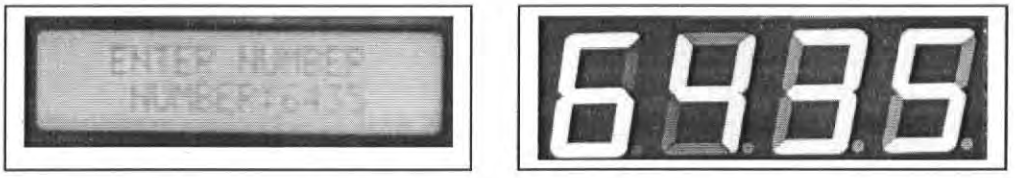
2. ใส่รหัสผ่าน (ในที่นี้คือ "9999") โดยใส่ค่า "9999" แล้วกดปุ่ม # บนคีย์แพ็ด



(a) ใส่รหัสผ่านด้วยคีย์แพ็ด (b) การแสดงผลบนจอ LCD

รูปที่ 4.2 แสดงผลของการใส่รหัสผ่าน

3. กำหนดเป้าหมายในการผลิต (ในที่นี้คือ "6435") โดยใส่ค่า "6435" แล้วกดปุ่ม #



(a) กำหนดเป้าหมายด้วยคีย์แพ็ด (b) การแสดงผลของป้ายนับจำนวนผลการผลิตหลังจากกดปุ่ม #

รูปที่ 4.3 แสดงผลของการกำหนดเป้าหมายในการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ทดลองนับค่าจำนวนผลการผลิต (จำลองด้วยการรับค่าจากการกดสวิทซ์) ป้ายนับจำนวนผลการผลิตจะแสดงผลโดยการนับลงที่ละ 1 ตามจำนวนครั้งของการกดสวิทซ์ และการเมื่อผลิตครบตามเป้าหมาย ป้ายนับจำนวนผลการผลิตก็จะแสดงค่า “0000”

รูปที่ 4.4 แสดงการทำงานของป้ายนับจำนวนผลการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองและผลการทดลองในส่วนของการติดต่อกับคอมพิวเตอร์

สำหรับการทดลองและผลการทดลองส่วนของการติดต่อกับคอมพิวเตอร์นั้นมีขั้นตอนการทดลองและผลการทดลองดังนี้คือ

4.2.1. การติดตั้งโปรแกรม On Time Production Counter Using Information Technology

โดยทำการติดตั้งโปรแกรมตามขั้นตอนที่อธิบายในคู่มือการติดตั้ง และ ใช้งานโปรแกรม On Time Production Counter Using Information Technology (ในภาคผนวก)

4.2.2. การใช้งานโปรแกรม On Time Production Counter Using Information Technology

ทำการเปิดสวิตซ์ และ รีเซทป้ายนับจำนวนผลการผลิต จากนั้นทำการเปิดโปรแกรม On Time Production Counter Using Information Technology ขึ้นมาเพื่อทำการทดลองกำหนดค่าเป้าหมาย และทดสอบการทำงานของโปรแกรมต่อไป

รูปที่ 4.5 แสดงหน้าต่างหลักของโปรแกรม On Time Production Counter Using Information Technology

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

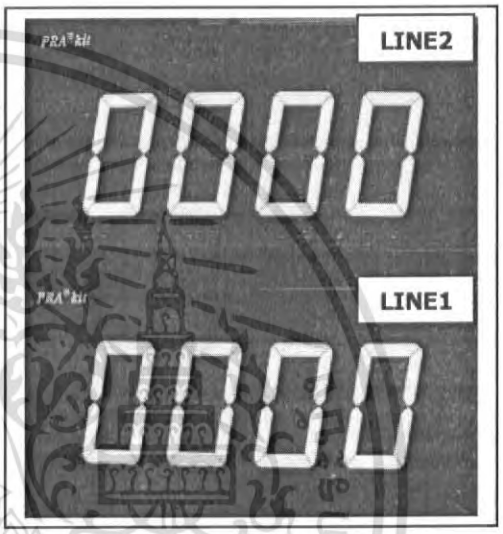
4.2.3. การทดลองใช้งานโปรแกรมโดยการกำหนดค่าเป้าหมาย

การทดลองนี้เป็นการทดสอบการทำงานของโปรแกรม ซึ่งแบ่งออกได้เป็น 3 กรณีดังนี้ คือ

1. กำหนดค่าเป้าหมายเฉพาะ LINE1 มีขั้นตอนในการทดลอง และผลการทดลองดังนี้ คือ

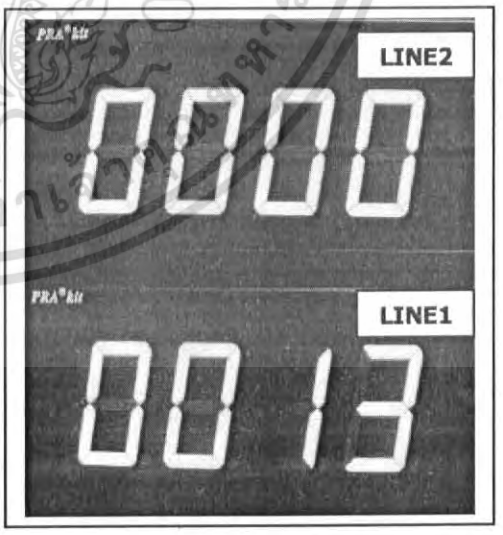
- กำหนดค่าเป้าหมายเฉพาะ LINE1 (การทดลองกำหนดเป็น 13) จากนั้นทำการส่งค่าเริ่มต้นของการนับไปยังป้ายนับจำนวนการผลิต โดยการกดปุ่ม Enter บนคีย์บอร์ดหรือคลิกที่ปุ่ม OK ในเฟรม LINE1

(a) เมื่อกำหนดค่าเป้าหมายเป็น “13” แล้วส่งค่า จะมีกล่องข้อความเตือนให้ใส่ตัวเลขครบ 4หลัก



(b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต

(c) เมื่อกำหนดค่าเป้าหมายเป็น “0013” แล้วส่งค่า



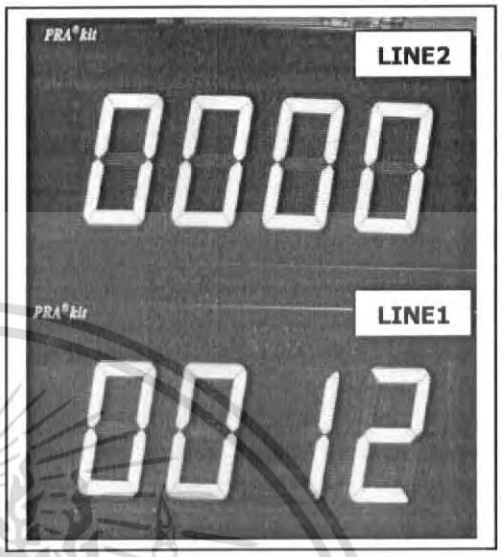
(d) การแสดงผลที่ป้ายนับจำนวนผลการผลิตหลังจากการส่งค่าจาก LINE1

รูปที่ 4.6 กำหนดค่าเป้าหมายใน LINE1 และการแสดงผลของป้ายนับจำนวนการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

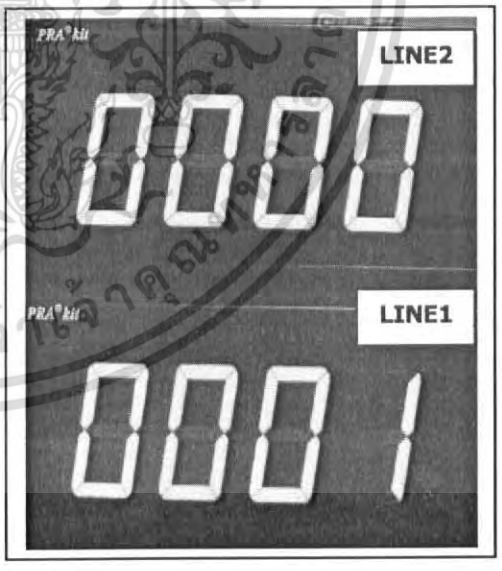
- ทดลองนับค่าของผลผลิต โดยการกดที่สวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์

(a) ในส่วนของ ACTUAL TARGETและTOTAL ค่าจะเพิ่มขึ้น 1 ค่า ส่วนใน DIFFERENCE TARGET ค่าจะลดลง 1 ค่า



(b) ป้ายนับจำนวนผลการผลิตจะนับลง 1 ค่า จาก “0013” เป็น “0012”

(c) การแสดงผลการนับค่าบนจอมอนิเตอร์



(d) ป้ายนับจำนวนผลการผลิตจะนับลงครั้งละ 1 ค่า

รูปที่ 4.7 การนับค่าใน LINE1 และการแสดงผลของป้ายนับจำนวนการผลิต เมื่อกดสวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

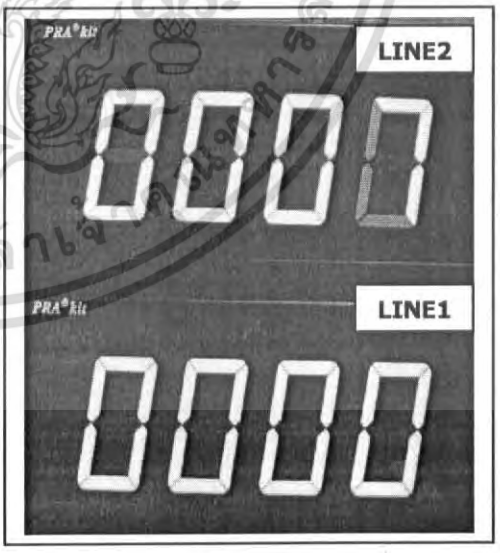
(e) เมื่อการนับเสร็จลง ที่ LINE1 จะมีเสียงดนตรี และ มีข้อความแสดงให้ทราบว่าการนับเสร็จแล้ว โดยข้อความจะกระพริบ 4 ครั้ง และ จะติดค้างไว้ จนกว่าจะมีการกำหนดเป้าหมายในการนับอีกครั้ง ในขณะที่การแสดงผลที่ป้ายนับจำนวนการผลิตจะแสดงค่าเป็น "0000"

รูปที่ 4.7 การนับค่าใน LINE1 และการแสดงผลของป้ายนับจำนวนการผลิต เมื่อกดสวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์ (ต่อ)

2. กำหนดค่าเป้าหมายเฉพาะ LINE2 มีขั้นตอนในการทดลอง และผลการทดลองดังนี้ คือ

- กำหนดค่าเป้าหมายเฉพาะ LINE2 (การทดลองกำหนดเป็น 7) จากนั้นทำการส่งค่าเริ่มต้นของการนับไปยังป้ายนับจำนวนการผลิต โดยการกดปุ่ม Enter บนคีย์บอร์ดหรือคลิกที่ปุ่ม OK ในเฟรม LINE2

(a) เมื่อกำหนดค่าเป้าหมายเป็น "0007" แล้วส่งค่า

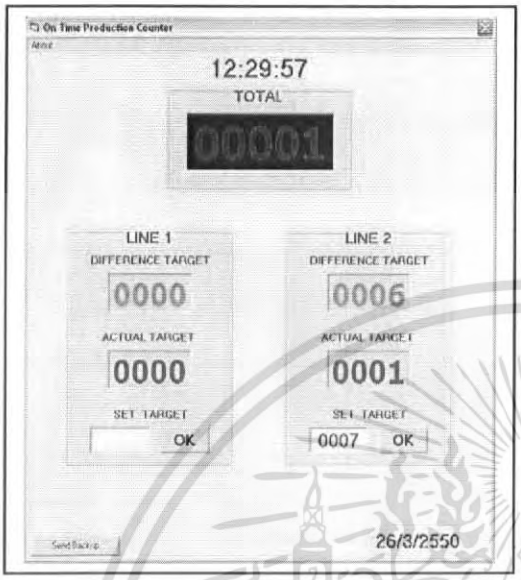


(b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต หลังจากการส่งค่าจาก LINE2

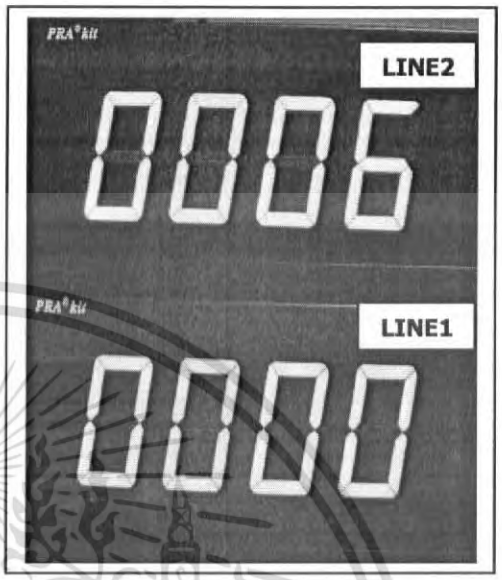
รูปที่ 4.8 กำหนดค่าเป้าหมายใน LINE2 และการแสดงผลของป้ายนับจำนวนการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทดลองนับค่าของผลผลิตโดยการกดที่สวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์

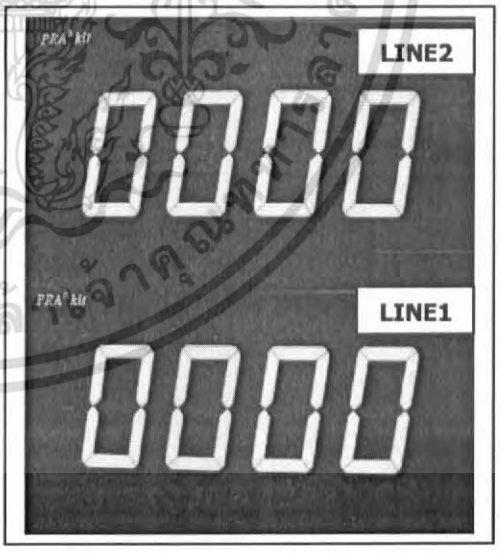


(a) ในส่วนของ ACTUAL TARGET และTOTAL ค่าจะเพิ่มขึ้นทีละ 1 ค่า ส่วนใน DIFFERENCE TARGET ค่าจะลดลง 1 ค่า



(b) ป้ายนับจำนวนผลการผลิตจะนับลง 1 ค่า จาก “0007” เป็น “0006”

(c) การแสดงผลบนจอมอนิเตอร์เมื่อการนับใน LINE2 เสร็จสิ้นลง



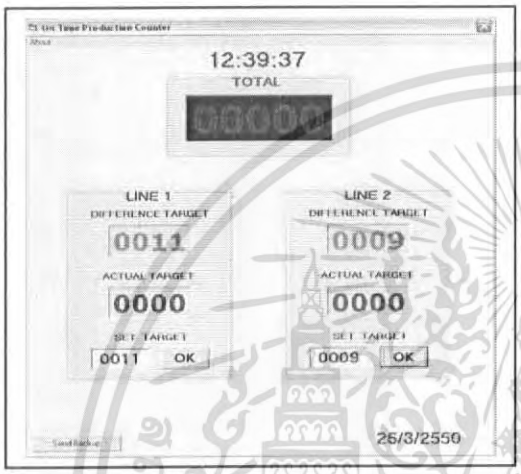
(d) ป้ายนับจำนวนผลการผลิตจะแสดงค่า “0000”

รูปที่ 4.9 การนับค่าใน LINE2 และการแสดงผลของป้ายนับจำนวนการผลิต เมื่อกดสวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์

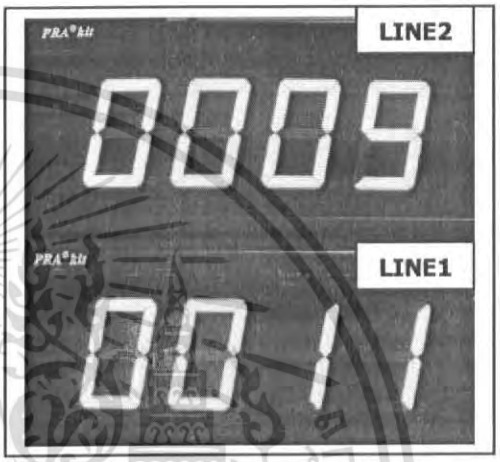
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กำหนดค่าเป้าหมายทั้ง LINE1 และ LINE2 มีขั้นตอนในการทดลองและผลการทดลองดังนี้ คือ

- กำหนดค่าเป้าหมายให้กับ LINE1 และ LINE2 (กำหนดค่าเป็น 11 และ 9 ตามลำดับ) จากนั้นทำการส่งค่า เริ่มต้นของการนับไปยังป้ายนับจำนวนการผลิต โดยการกดปุ่ม Enter บนคีย์บอร์ดหรือคลิกที่ปุ่ม OK ในเฟรม LINE1 และ LINE2



(a) กำหนดค่าเป้าหมาย LINE1 เป็น "0011"
กำหนดค่าเป้าหมาย LINE2 เป็น "0009"

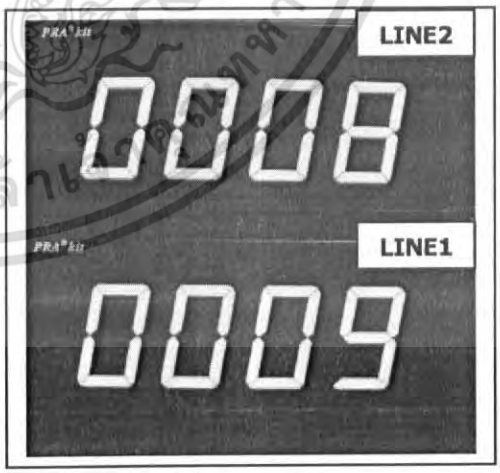


(b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต หลังจากการส่งค่าจาก LINE1 และLINE2

รูปที่ 4.10 กำหนดค่าเป้าหมายทั้งใน LINE1และLINE2 และการแสดงผลที่ป้ายนับจำนวนการผลิต

- ทดลองนับค่าของผลผลิตโดยการกดที่สวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์

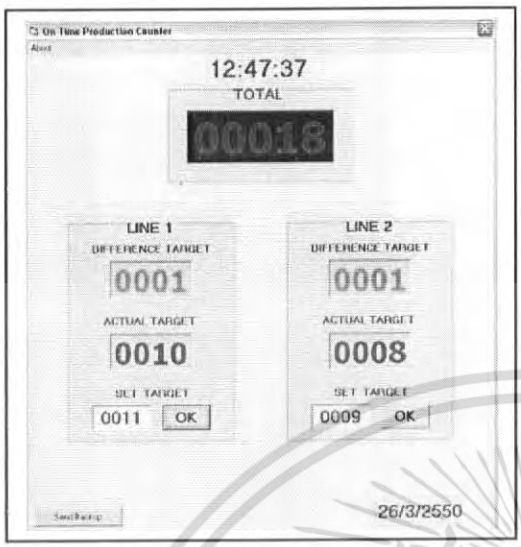
(a) การแสดงค่าที่ TOTAL จะมีค่าเท่ากับผลรวมของ ACTUAL TARGET ใน LINE1 และLINE2



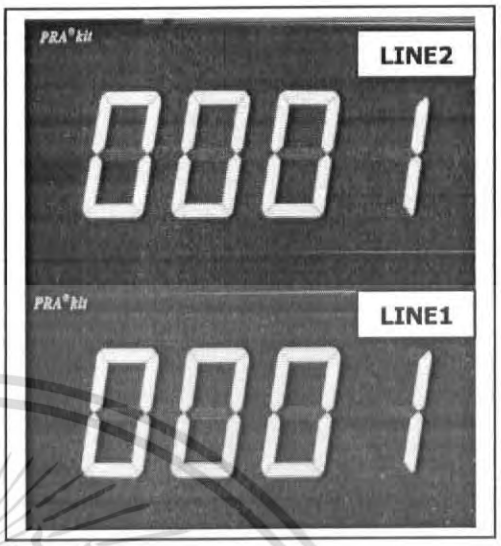
(b) การแสดงผลที่ป้ายนับจำนวนผลการผลิตของ LINE1 และLINE2 เมื่อมีการนับ

รูปที่ 4.11 การนับค่าใน LINE1 และLINE2 และการแสดงผลของป้ายนับจำนวนการผลิต เมื่อกดสวิตซ์ที่อยู่บนชุดจำลองการทำงานของรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

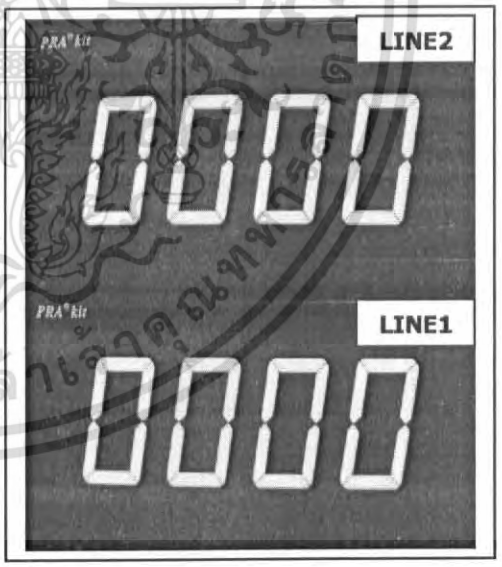


(c) การแสดงค่าที่ TOTAL จะมีค่าเท่ากับผลรวมของ ACTUAL TARGET ใน LINE1 และLINE2 โดย ACTUAL TARGET แต่ละ LINE จะเพิ่มทีละ 1 ค่า



(d) การแสดงผลที่ป้ายนับจำนวนผลการผลิตของ LINE1 และLINE2 เมื่อมีการนับ โดยจะลดลงทีละ 1 ค่า

(e) การแสดงเมื่อการนับเสร็จลง โดยจะมีข้อความแสดงให้ทราบว่าการนับเสร็จแล้ว แสดงในทั้งสอง LINE



(f) การแสดงผลที่ป้ายนับจำนวนผลการผลิตของ LINE1 และLINE2 เมื่อจบการนับทั้งสอง LINE จะแสดงค่าเป็น “0000”

รูปที่ 4.11 การนับค่าใน LINE1 และLINE2 และการแสดงผลของป้ายนับจำนวนการผลิต เมื่อกดสวิตช์ที่อยู่บนชุดจำลองการทำงานของรีเลย์ (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4. การทดลองส่งข้อมูลกลับ (Send Backup Data)

การทดลองนี้ได้จำลองให้เกิดไฟฟ้าดับที่ LINE ผลิต ขณะที่คอมพิวเตอร์ยังทำงานอยู่ (มีไฟฟ้าสำรอง) ซึ่งสามารถแบ่งการทดลองออกเป็น 3 กรณีดังนี้

1. ส่งข้อมูลกลับเมื่อไฟฟ้าดับเฉพาะ LINE1 มีขั้นตอนทดลองและผลการทดลองดังนี้ คือ

- ทำการกำหนดเป้าหมายและทดลองการนับทั้งสอง LINE

(a) กำหนดค่าเป้าหมาย LINE1 เป็น “0025” (b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต
กำหนดค่าเป้าหมาย LINE2 เป็น “0022” ของ LINE1 และLINE2

รูปที่ 4.12 การกำหนดเป้าหมายใน LINE1และLINE2 และการแสดงผลของป้ายนับจำนวนการผลิต

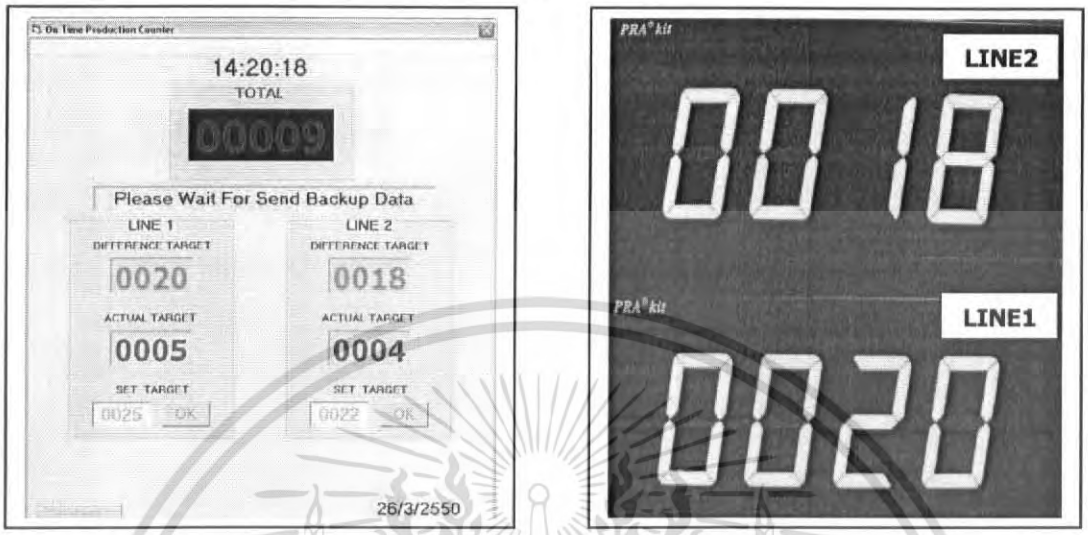
- จำลองให้เกิดไฟฟ้าดับที่ LINE1 (ปิดสวิตซ์ Power ป้ายนับจำนวนผลการผลิต LINE1)

(a) DIFFERENCE TARGET LINE1 เป็น “0020” (b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต
DIFFERENCE TARGET LINE2 เป็น “0018” ของ LINE1 ดับ และLINE2 เป็น “0018”

รูปที่ 4.13 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อจำลองให้เกิดไฟฟ้าดับที่ LINE1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทดลองการส่งข้อมูลกลับ โดยคลิกที่ปุ่ม Send Backup



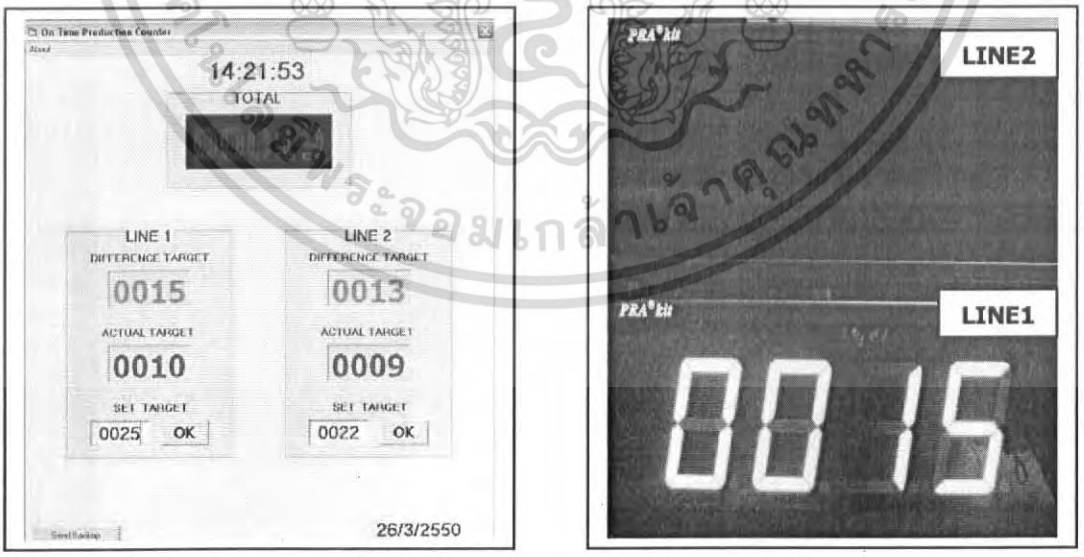
(a) จะมีข้อความแสดงให้ทราบว่ากำลังส่งข้อมูลกลับ (b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต LINE1 เป็น"0020"และLINE2 เป็น"0018"

รูปที่ 4.14 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อทำการส่งข้อมูลกลับ

- ทดลองนับค่าจำนวนผลการผลิต ผลการทดลองโปรแกรมทำงานได้ตามปกติ

2. ส่งข้อมูลกลับเมื่อไฟฟ้าดับเฉพาะ LINE2

- จำลองให้เกิดไฟฟ้าดับที่ LINE2 (ปิดสวิตซ์ Power ป้ายนับจำนวนผลการผลิต LINE2)

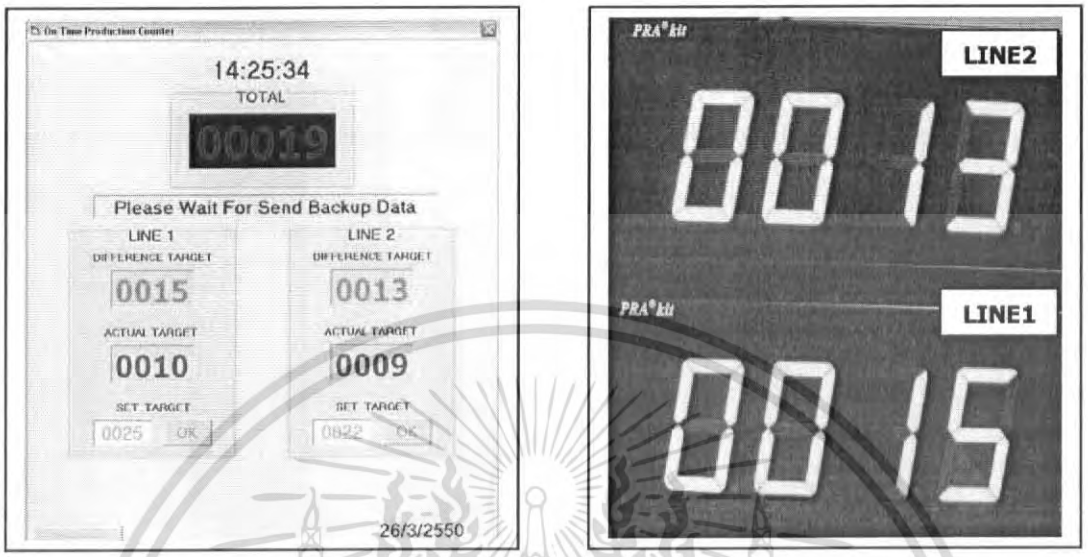


(a) DIFFERENCE TARGET LINE1 เป็น "0015" DIFFERENCE TARGET LINE2 เป็น "0013" (b) การแสดงผลที่ป้ายนับจำนวนผลการผลิตของ LINE1 เป็น "0015" และ LINE2 ดับ

รูปที่ 4.15 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อจำลองให้เกิดไฟฟ้าดับที่LINE2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทดลองการส่งข้อมูลกลับ โดยคลิกที่ปุ่ม Send Backup



(a) จะมีข้อความแสดงให้ทราบว่ากำลังส่งข้อมูลกลับ (b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต LINE1 เป็น“0015”และLINE2 เป็น“0013”

รูปที่ 4.16 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อทำการส่งข้อมูลกลับ

- ทดลองนับค่าจำนวนผลการผลิต ผลการทดลองโปรแกรมทำงานได้ตามปกติ

3. ส่งข้อมูลกลับเมื่อไฟฟ้าดับทั้ง LINE1 และ LINE2

- จำลองให้เกิดไฟฟ้าดับทั้งสอง LINE (ปิดสวิตซ์ Power ทั้งสอง LINE)

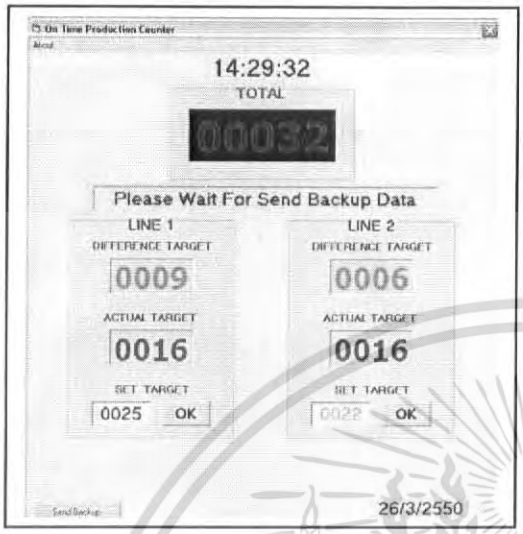
(a) DIFFERENCE TARGET LINE1 เป็น “0009” (b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต

DIFFERENCE TARGET LINE2 เป็น “0006” ของ LINE1 และ LINE2 ดับ

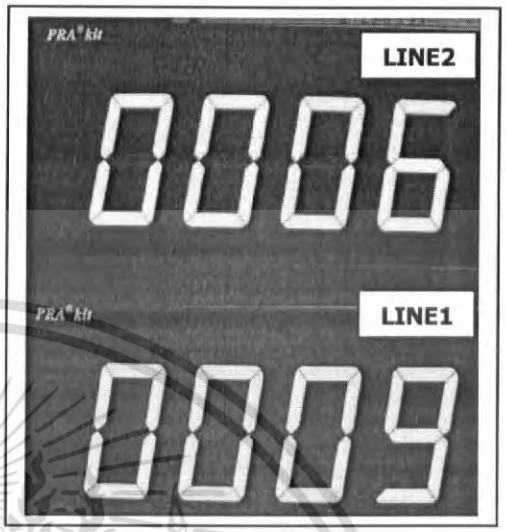
รูปที่ 4.17 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อจำลองให้ไฟฟ้าดับทั้งสองLINE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทดลองการส่งข้อมูลกลับ โดยคลิกที่ปุ่ม Send Backup



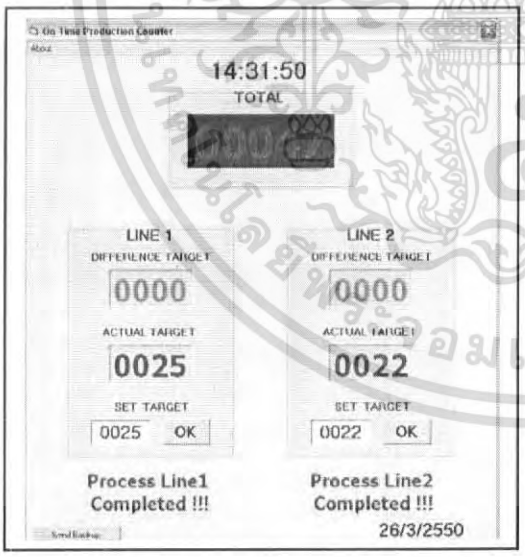
(a) จะมีข้อความแสดงให้ทราบว่ากำลังส่งข้อมูลกลับ



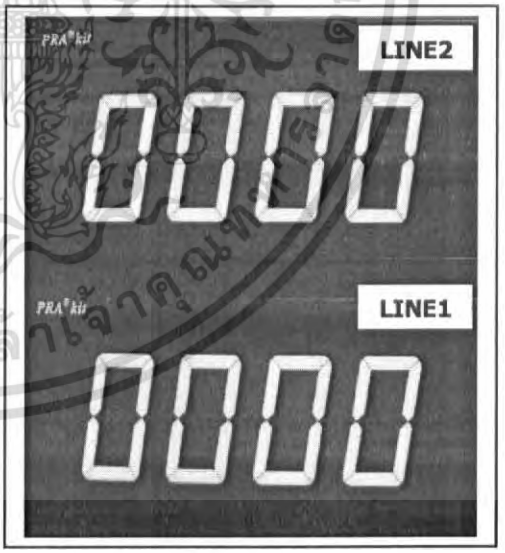
(b) การแสดงผลที่ป้ายนับจำนวนผลการผลิต LINE1 เป็น"0009"และLINE2 เป็น"0006"

รูปที่ 4.18 แสดงผลบนจอมอนิเตอร์และป้ายนับจำนวนการผลิตเมื่อทำการส่งข้อมูลกลับ

- ทดลองนับค่าจำนวนผลการผลิต ผลการทดลองโปรแกรมทำงานได้ตามปกติ



(e) การแสดงเมื่อการนับเสร็จลง โดยจะมีข้อความแสดงให้ทราบว่าการนับเสร็จแล้ว แสดงในทั้งสอง LINE



(f) การแสดงผลที่ป้ายนับจำนวนผลการผลิตของ LINE1 และLINE2 เมื่อจบการนับทั้งสอง LINE จะแสดงค่าเป็น "0000"

รูปที่ 4.19 แสดงการนับค่าใน LINE1 และLINE2 และการแสดงผลของป้ายนับจำนวนการผลิต หลังจากมีการส่งข้อมูลกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.5. การออกจากโปรแกรม

การทดลองออกจากโปรแกรมนั้นทำได้โดยการคลิกที่ปุ่ม จะมีกล่องข้อความเพื่อยืนยันการออกจากโปรแกรมแสดงขึ้นมา

รูปที่ 4.20 แสดงการยืนยันการออกจากโปรแกรม

สำหรับกล่องข้อความเพื่อยืนยันการออกจากโปรแกรมจะมีปุ่มให้เลือก คือ ปุ่ม OK และปุ่ม Cancel เมื่อทดลองคลิกที่ปุ่ม ได้ผลการทดลองดังนี้

- เมื่อคลิกที่ปุ่ม Cancel โปรแกรมจะทำงานตามปกติ
- เมื่อคลิกที่ปุ่ม OK จะออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 สรุปผลการทดลอง

5.1.1 สรุปผลการทดลองในส่วนของการตั้งเป้าหมายทางคีย์แพ็ด

จากการทดลอง เมื่อทำการป้อนรหัสผ่าน และ กำหนดเป้าหมายในการผลิตแล้วค่าจำนวนเป้าหมายในการผลิตจะถูกส่งไปยังยังป้ายนับจำนวนผลการผลิต และ เมื่อทำการจำลองการนับจำนวนผลการผลิตจากการนับจำนวนครั้งของการกดสวิตช์ เพื่อใช้แทนการนับที่ได้จากการทำงานของตู้เติมลมอัตโนมัติ ปรากฏว่าได้ผลการทดลองตามความต้องการจากการทำงานของโปรแกรม

5.1.2 สรุปผลการทดลองในส่วนของการติดต่อกับคอมพิวเตอร์

จากการทดลอง เมื่อทำการติดตั้งโปรแกรม On Time Production Counter Using Information Technology แล้วทำการรันโปรแกรมและทดสอบการทำงานของโปแกรม ได้ข้อสรุปดังนี้ คือ

1. การกำหนดค่าเป้าหมายต้องทำการกำหนดเป้าหมายให้ครบ 4 หลัก หากการกำหนดเป้าหมายไม่ครบ 4 หลัก โปรแกรมจะไม่สามารถส่งค่าเป้าหมายไปยังป้ายนับจำนวนผลการผลิตได้ และจะมีกล่องข้อความเตือนให้มีการกำหนดค่าครบ 4 หลัก
2. ก่อนที่จะทำการส่งข้อมูลกลับในกรณีที่ไฟฟ้าดับในส่วนของ LINE ผลิต จะต้องทำการรีเซทป้ายนับจำนวนการผลิตใน LINE ผลิต ที่ไฟฟ้าดับทุกครั้ง ไม่เช่นนั้นจะทำให้การส่งข้อมูลกลับไม่ประสบความสำเร็จ
3. ในระหว่างที่โปรแกรมกำลังทำการส่งข้อมูลกลับนั้น ห้ามไม่ให้มีการนับจำนวนผลการผลิต จนกว่าการส่งข้อมูลกลับจะเสร็จสมบูรณ์ เพราะถ้าหากมีการนับจำนวนผลการผลิตในระหว่างที่ทำการส่งข้อมูลกลับ จะทำให้โปรแกรมทำงานผิดพลาด

การทำงานของโปรแกรมในส่วนอื่น ๆ นั้นการทำงานเป็นไปตามโปรแกรมที่เขียนขึ้น ซึ่งจะพบว่า ในส่วนของการติดต่อกับคอมพิวเตอร์นั้น การใช้งานโปรแกรม On Time Production Counter Using Information Technology นั้นใช้งานได้ง่ายอีกทั้งยังทำให้ทราบถึงรายละเอียดเกี่ยวกับข้อมูลการผลิตในขณะนั้นได้อย่างง่ายดาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการทดลองโดยการกำหนดเป้าหมายในการผลิตด้วยคีย์แพ็ด และ ในขณะที่เปิดโปรแกรม On Time Production Counter Using Information Technology (ไม่มีการกำหนดค่าจากโปรแกรม) และเมื่อทดลองทำการนับค่าจำนวนผลการผลิต ปรากฏว่าการแสดงค่าที่ป้ายนับจำนวนการผลิตเป็นปกติ แต่ในส่วนของการแสดงผลที่จอมอนิเตอร์นั้นเกิดข้อผิดพลาดขึ้น ทั้งนี้เนื่องจากการกำหนดเป้าหมายในการผลิตด้วยคีย์แพ็ด ไม่ได้มีการส่งค่าไปยังคอมพิวเตอร์ และเมื่อมีการนับจำนวนการผลิตขึ้น จะมีการส่งค่าให้กับคอมพิวเตอร์ เพื่อไปแสดงผลที่จอมอนิเตอร์ จึงทำให้การทำงานของโปรแกรม On Time Production Counter Using Information Technology เกิดความผิดพลาดขึ้น

5.2 ปัญหาและอุปสรรคที่เกิดขึ้นขณะทำโครงงาน

การเขียนโปรแกรมเพื่อให้ไมโครคอนโทรลเลอร์ ทำงานร่วมกันคือไมโครคอนโทรลเลอร์ที่ใช้ในการกำหนดเป้าหมาย และ จะส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ของป้ายนับจำนวนผลการผลิต มีความยุ่งยากและซับซ้อน จึงทำให้โปรแกรมมีความผิดพลาดบ่อย ก่อนจะทำการแก้ไขจนสมบูรณ์

สำหรับในส่วนของการทดลอง และ ผลการทดลองในส่วนของการติดต่อกับคอมพิวเตอร์มีอุปสรรคในการทดลองคือเนื่องจากการทดลองต้องใช้พอร์ตอนุกรมจำนวน 2 พอร์ต แต่เครื่องคอมพิวเตอร์ที่ใช้ในการทดลองมีพอร์ตอนุกรมเพียงพอร์ตเดียว จึงทำให้ยากที่จะทำการทดลองการนับค่าจำนวนผลการผลิตได้พร้อมกัน 2 LINE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. กิตติ ภักดีวัฒนะกุล และ จำลอง ครูอุตสาหะ **Visual Basic 6 ฉบับโปรแกรมเมอร์.** ไทยเจริญการพิมพ์ กรุงเทพมหานคร 2544
2. ฉัททวุฒิ พืชผล และพิชิต สันติกุลานนท์ **คู่มือเรียน Visual Basic 6** โปรวิชั่น กรุงเทพมหานคร 2542
3. ชัยวัฒน์ ลิ้มพรจิตรวิไล และ วรพจน์ กรแก้ววัฒนกุล **เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51**. อินโนเวตีฟ เอ็กเพอริเมนต์ กรุงเทพมหานคร, 2542
4. ธีรวัฒน์ ประกอบผล, **การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี.** สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น) กรุงเทพมหานคร 2545
5. สันติ นุราช, อุกฤษฎ์ ตันทสุทธานนท์, **เรียนรู้ไมโครคอนโทรลเลอร์ MCS-51 ฉบับภาษา C** Micro Research Technology กรุงเทพมหานคร 25xx
6. อภิชาต ภู่พลับ, **เขียนโปรแกรม Hardware Interface ด้วย VB 6.** ไอดีซีอินโฟ ดิสทริบิวเตอร์เซ็นเตอร์ กรุงเทพมหานคร , 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการติดตั้งและใช้งานโปรแกรม

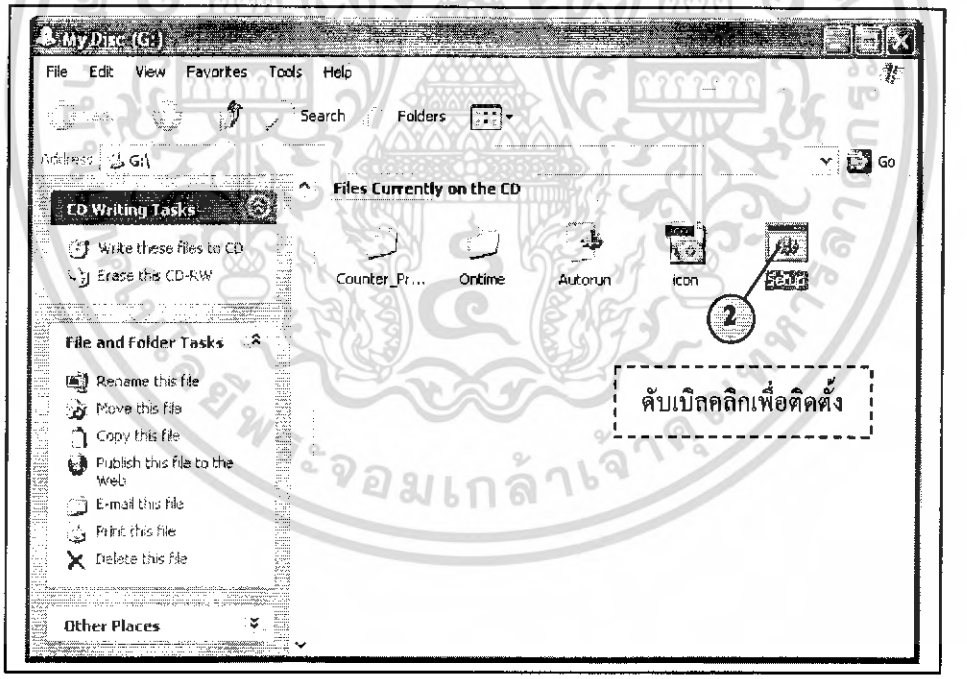
On Time Production Counter Using Information Technology

ขั้นตอนการติดตั้งโปรแกรม On Time Production Counter Using Information Technology

การติดตั้งโปรแกรม On Time Production Counter Using Information Technology นั้นจะมีลักษณะที่ไม่แตกต่างจากการติดตั้งโปรแกรมทั่วๆ ไปในวินโดว์ คือต้องการรันไฟล์เซตอัพเพื่อให้วินโดว์ดำเนินการกับโปรแกรม ซึ่งมีขั้นตอนการติดตั้งดังนี้

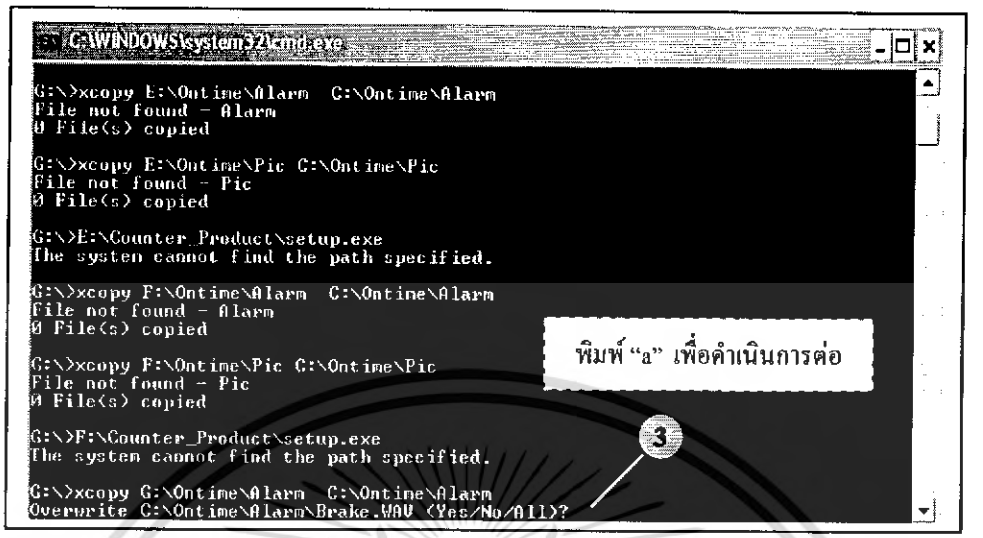
1. ใส่แผ่นซีดีโปรแกรม On Time Production Counter Using Information Technology ลงในไดรว์ซีดีรอม

2. รอให้โปรแกรมติดตั้งทำงานโดยอัตโนมัติ (Auto run) แต่ถ้าโปรแกรมอัตโนมัติไม่ทำงานก็ให้ดับเบิลคลิกที่ไฟล์ Setup.bat



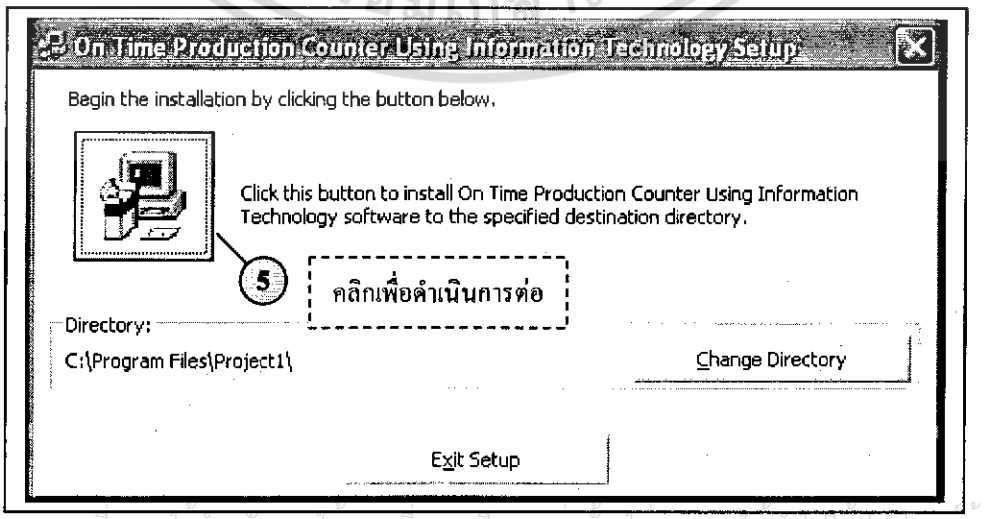
3. โปรแกรมจะเริ่มการติดตั้ง โดยวินโดว์จะทำการคัดลอกไฟล์ต่างๆ ที่ใช้ในโปรแกรม ซึ่งจะมีหน้าต่างปฏิบัติการเพื่อแสดงขั้นตอนการติดตั้ง โดยขั้นตอนนี้จะให้มีการยืนยันการคัดลอกไฟล์ โดยให้พิมพ์ "a" ที่คีย์บอร์ด เพื่อเป็นการคัดลอกไฟล์ทั้งหมด จากนั้นก็จะเข้าสู่การติดตั้งโปรแกรม และให้ดำเนินการตามขั้นตอนที่ 4.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



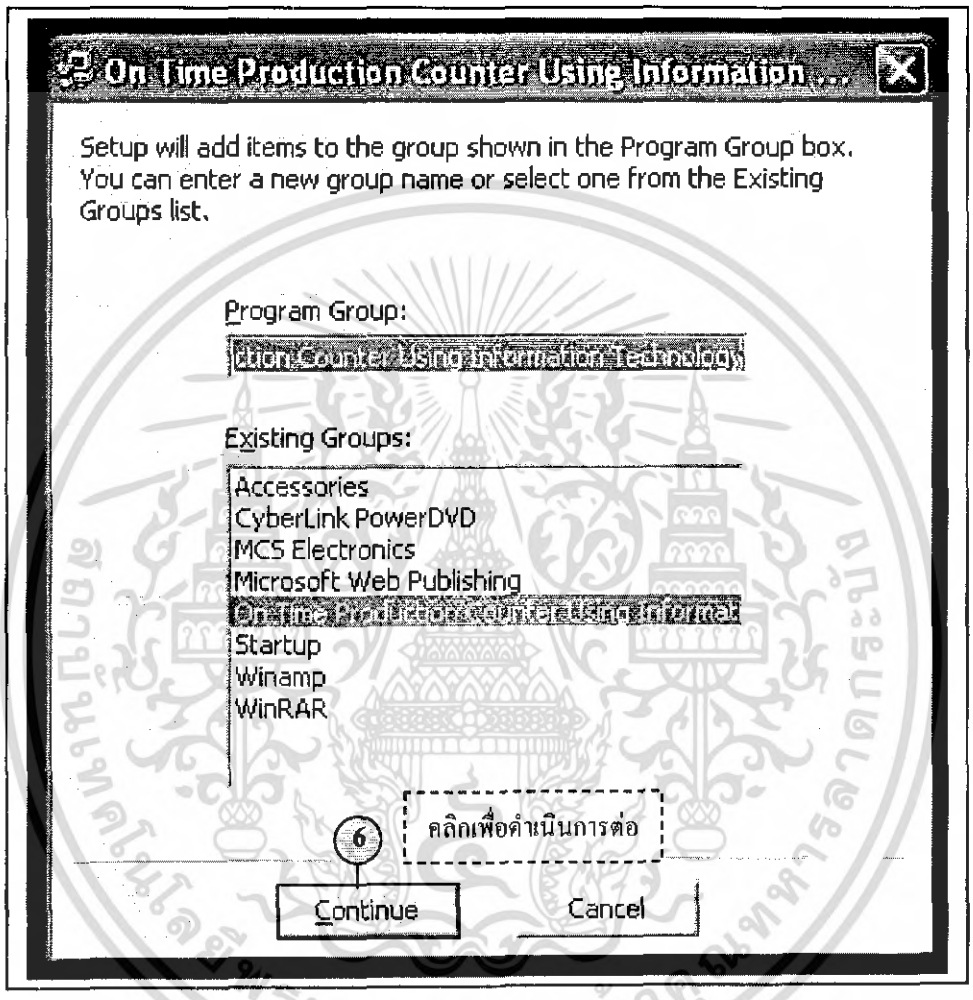
4. โปรแกรมจะเข้าสู่การติดตั้ง ให้คลิกที่ปุ่ม "OK" เพื่อทำการติดตั้งโปรแกรม

5. ทำการคลิกที่ปุ่มตังในรูปเพื่อทำการติดตั้งต่อไป

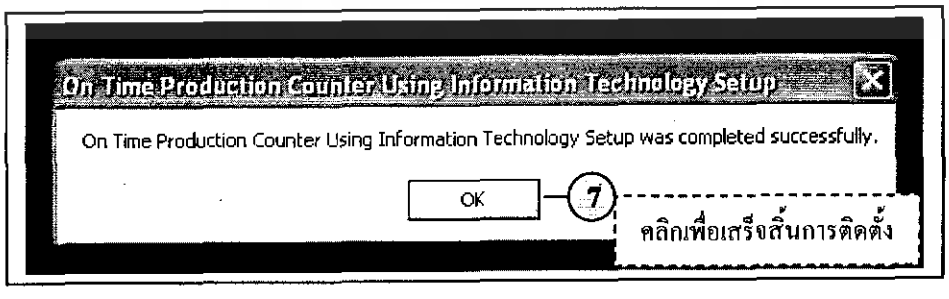


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ในการติดตั้งระบบจะให้มีการเลือกกลุ่มของโปรแกรม (Program Group) ซึ่งการติดตั้งได้เลือกเป็น On Time Production Counter Using Information Technology โดยอัตโนมัติแล้ว จากนั้นให้คลิกที่ปุ่ม "Continue"



7. โปรแกรมจะทำการติดตั้งจนเสร็จ เมื่อการติดตั้งเสร็จสมบูรณ์จะมีหน้าต่างแสดงว่าการติดตั้งเสร็จสิ้น ให้คลิกที่ปุ่ม "OK"

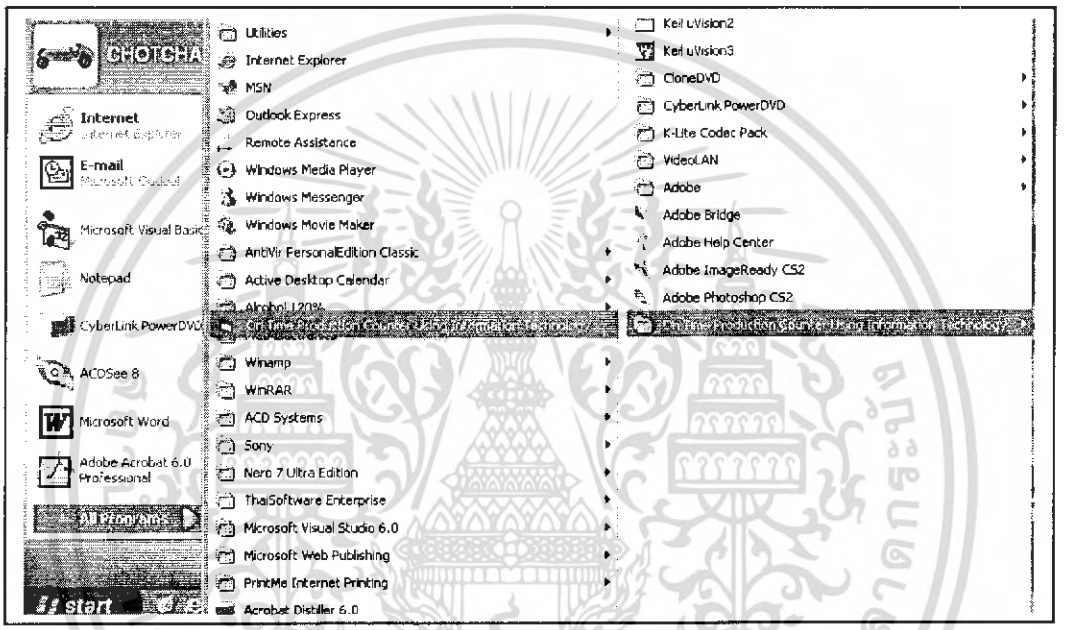


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

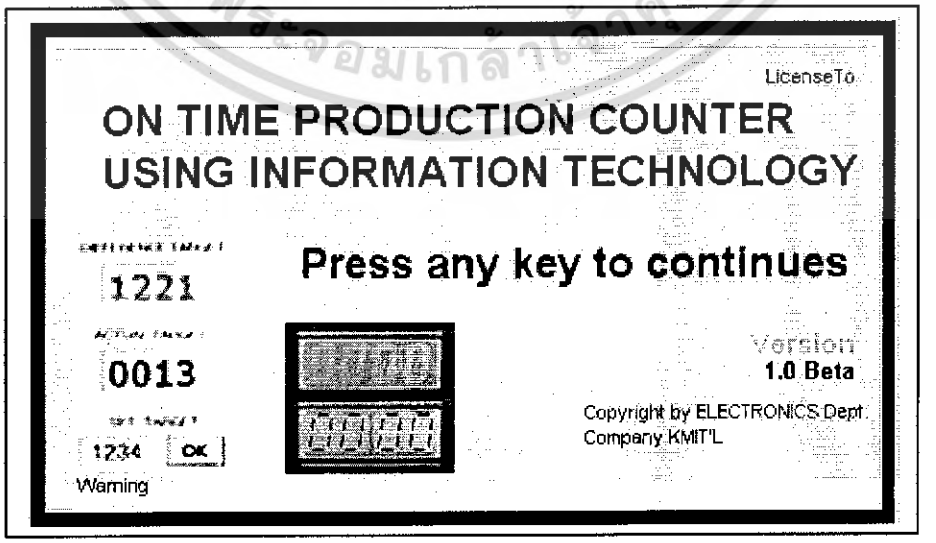
ขั้นตอนการใช้งานโปรแกรม On Time Production Counter Using Information Technology

การใช้งานโปรแกรม On Time Production Counter Using Information Technology ซึ่งมีขั้นตอนการใช้งานดังนี้

1. เปิดโปรแกรม On Time Production Counter Using Information Technology โดยคลิกที่ Start >> All Programs >> On Time Production Counter Using Information Technology >> On Time Production Counter Using Information Technology ดังรูป

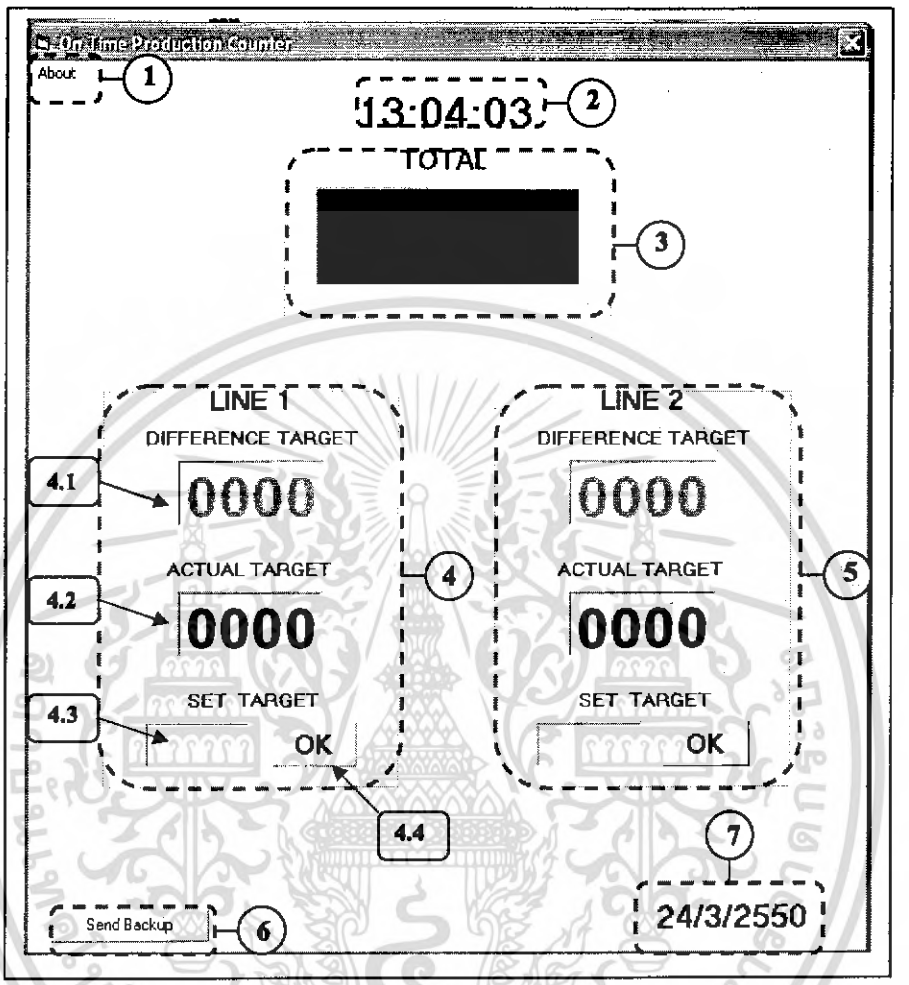


2. โปรแกรมจะเปิดหน้าต่างเพื่อแสดงการเข้าสู่โปรแกรม จากนั้นให้คลิกภายในกรอบของหน้าต่างหรือไม่ก็ทำการกดที่ปุ่มใดๆ ก็ได้บนคีย์บอร์ดเพื่อโปรแกรมจะทำงานต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. โปรแกรมจะแสดงหน้าต่างหลักขึ้นมา ซึ่งมีส่วนประกอบดังนี้



1) เป็นส่วนของเมนูบาร์ (Menubar) ประกอบด้วยเมนู About ซึ่งเมื่อคลิกที่เมนู About นี้ก็จะมีหน้าต่างที่แสดงรายละเอียดเกี่ยวกับผู้พัฒนาโปรแกรม

2) เป็นลาเบล (Label) แสดงเวลาในขณะนั้น

3) เป็นเฟรมที่ใช้เป็นส่วนในการแสดงผลผลิตทั้งหมดที่ผลิตได้ขณะนั้น โดยค่าที่แสดงจะเป็นค่าที่เกิดจากการนำค่าของผลผลิตใน LINE1 และ LINE2 มารวมกัน

4) ส่วนนี้เป็นเฟรมที่แสดงรายละเอียดต่างๆ ใน LINE1 ซึ่งประกอบด้วยส่วนต่างๆ ดังนี้

4.1) DIFFERENCE TARGET ส่วนนี้จะเป็นส่วนที่แสดงค่าส่วนต่างของเป้าหมายกับค่าผลผลิตที่ผลิตได้จริงๆ ของ LINE1 ในเวลาขณะนั้น นั่นก็คือค่าผลผลิตที่เหลือที่ต้องทำการผลิตให้เสร็จของ LINE1

4.2) ACTUAL TARGET ส่วนนี้จะเป็นส่วนที่แสดงค่าผลผลิตที่ผลิตได้จริงๆ ของ LINE1 ในเวลาขณะนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3) SET TARGET ในส่วนนี้เป็นกล่องข้อความสำหรับพิมพ์ค่าเป้าหมายที่ต้องการผลิตใน LINE1

4.4) OK ปุ่ม OK นี้เป็นปุ่มเพื่อยืนยันการกำหนดเป้าหมายของ LINE1 หลักจากที่กำหนดค่าเป้าหมายในข้อ 4.3 แล้ว เมื่อเราคลิกที่ปุ่ม OK หรือกดปุ่ม Enter ที่คีย์บอร์ด ค่าเป้าหมายก็จะถูกส่งไปยังป้ายนับจำนวนผลข้อมูลผลการผลิตใน LINE1 และจะแสดงผลเป็นค่าเริ่มต้นในส่วนของ DIFFERENCE TARGET นั่นเอง

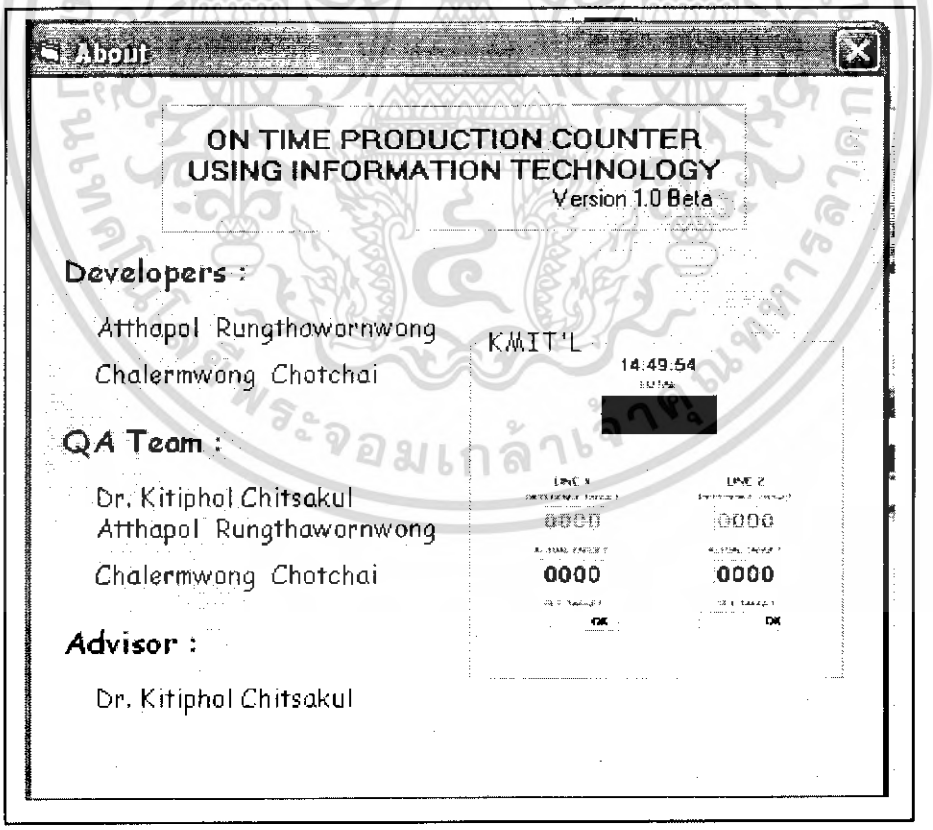
5) ส่วนนี้เป็นเฟรมที่แสดงรายละเอียดต่างๆ ใน LINE2 ซึ่งส่วนประกอบต่างๆ เหมือนกับ LINE1

6) เป็นลาเบล “Send Backup” สำหรับส่งข้อมูลกลับในกรณีที่เกิดไฟฟ้าดับในส่วนของ LINE ผลิต (คอมพิวเตอร์ยังทำงานอยู่)

7) เป็นลาเบล แสดงวัน, เดือน และปีพุทธศักราช ในขณะนั้น

4. ทดลองใช้งานส่วนประกอบด้วยต่างๆ ของโปรแกรม ดังนี้

4.1 คลิกที่เมนูบาร์ About จะมีหน้าต่างที่แสดงรายละเอียดเกี่ยวกับผู้พัฒนาโปรแกรม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

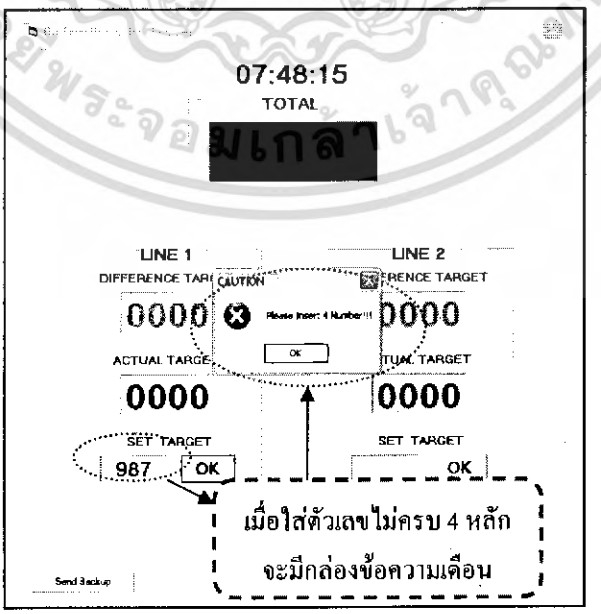
4.2 การกำหนดค่าเป้าหมายในการผลิตสำหรับการผลิตใน LINE1 และ LINE2

4.2.1 พิมพ์จำนวนเป้าหมายที่ต้องการผลิตในแต่ละ LINE ผลิต ด้วยตัวเลข 4 หลัก เช่น ต้องการที่จะผลิตจำนวน 987 ใน LINE1 ก็ให้พิมพ์ "0987" ในกล่องข้อความ SET TARGET ของ LINE1

4.2.2 จากนั้นให้คลิกที่ปุ่ม OK หรือไม่ก็กดที่ปุ่มใดๆ บนคีย์บอร์ต

4.2.3 จำนวนเป้าหมายที่ต้องการผลิตจะถูกส่งไปแสดงที่ป้ายนับจำนวนผลการผลิต และจะส่งค่ากลับมาแสดงในส่วนของ DIFFERENCE TARGET ให้พร้อมนับค่าจำนวนผลผลิต

4.2.4 กรณีทำการส่งค่าจำนวนเป้าหมาย โดยพิมพ์ตัวเลขไม่ครบ 4 หลัก จะมีกล่องข้อความเตือนให้ใส่ให้ครบ 4 หลัก จากนั้นให้ทำตามขั้นตอน 4.2.1 ถึง 4.2.3 ใหม่อีกครั้ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การส่งข้อมูลกลับกรณีที่เกิดไฟฟ้าดับ

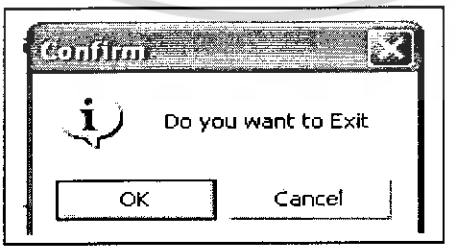
4.3.1 ทำการรีเซทป้ายนับจำนวนผลการผลิตของ LINE ผลิต ที่ไฟฟ้าดับ (ไฟฟ้าอาจดับเฉพาะ LINE1 , LINE2 หรืออาจดับทั้งสอง LINE)

4.3.2 ทำการคลิกที่ปุ่ม Send Backup

4.3.3 จะมีลาเบลข้อความแสดงให้ทราบว่ากำลังทำการส่งข้อมูลกลับอยู่ ("Pleasa Wait For Sent Backup Data") ให้รอจนกว่าข้อความจะหายไป จึงจะสามารถทำการนับจำนวนการผลิตต่อไปได้

4.4 การออกจากโปรแกรม

4.4.1 คลิกที่ปุ่ม จะมีกล่องยืนยันการออกจากโปรแกรมแสดงขึ้นมา



4.4.2 หากต้องการออกจากโปรแกรมให้คลิกที่ปุ่ม OK

4.4.3 หากต้องการกลับไปเพื่อให้โปรแกรมจะทำงานต่อ คลิกที่ปุ่ม Cancel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

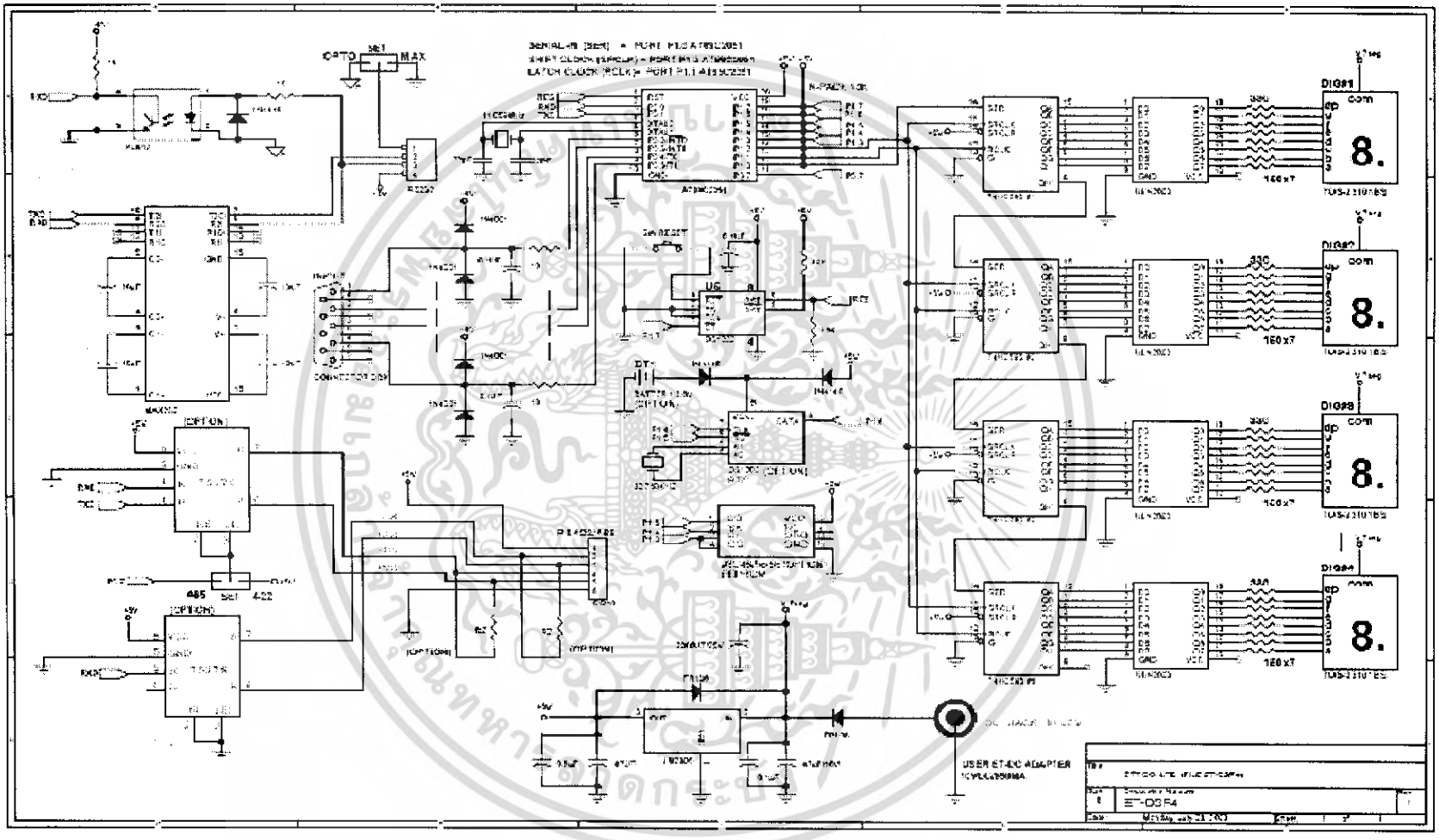
5. คุณสมบัติอื่น ๆ ของโปรแกรม On Time Production Counter Using Information Technology

- เมื่อลาเบลแสดงเวลาแสดงเวลาเป็น "12:00:00" จะมีเสียงดนตรีดังขึ้นมาพร้อมกับจะมีข้อความ "To Stop Work!!!" แสดงขึ้นมาให้ทราบว่าถึงเวลาพักเที่ยงแล้ว

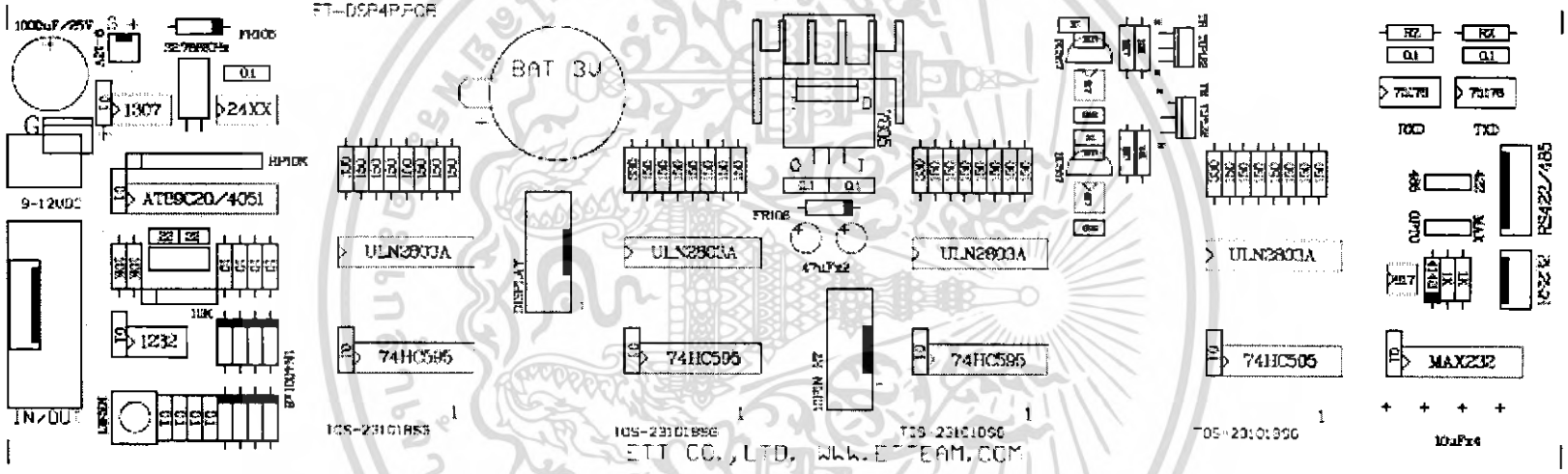
- เมื่อลาเบลแสดงเวลาแสดงเวลาเป็น "12:59:00" จะมีเสียงดนตรีดังขึ้นมาพร้อมกับจะมีข้อความ "To Stop Relax!!!" แสดงขึ้นมาให้ทราบว่าหมดเวลาพักเที่ยงแล้ว และข้อความจะหายไปเมื่อลาเบลแสดงเวลาแสดงเวลาเป็น "13:00:00"

- เมื่อการนับจำนวนการผลิตจบลง (ผลิตได้ครบตามเป้าหมายที่ตั้งไว้) ใน LINE ใด ก็จะมีเสียงดนตรีดังขึ้นมาพร้อมกับจะมีข้อความ "Process Linex Completed!!!" แสดงขึ้นมาให้ทราบว่า LINE นั้นได้ทำการผลิตเสร็จแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ลายวงจรของบอร์ดแสดงผล ET-DSP4



การวางอุปกรณ์ในบอร์ดแสดงผล ET-DSP4