การนำ FUZZY LOGIC มาประยุกต์ใช้งานในเครื่องทำน้ำอุ่น

(Fuzzy Logic Application In Electric Shower)

โดย

นาย ไพรสิทธิ์ ศรีวัฒนานุกูลกิจ

นาย วิบูลย์ ปิยวัฒนเมธา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033216

ปริญญานิพนธ์ปีการศึกษา 2536

ภาควิชา **อิเล็กทรอนิคส์**

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การนำ FUZZY LOGIC มาประยุกต์ใช้งานในเครื่องทำน้ำอุ่น

ผู้จัดทำ

1. นาย ไพรสิทธิ์ ศรีวัฒนานุกูลกิจ 33100275
2. นาย วิบูลย์ ปิยวัฒนเมธา 33100349

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

_____ อาจารย์ที่ปรึกษา

(อาจารย์ ขนิษฐา แซ่ตั้ง)

_____ อาจารย์ที่ปรึกษา

(ผศ. ดร. กิตติ ไพฑูรย์วัฒนกิจ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำ Fuzzy Logic มาประยุกต์ใช้งานในเครื่องทำน้ำอุ่น
(Fuzzy Locgic Application In Electric Shower)

โดย นาย ไพรสิทธิ์ ศรีวัฒนานุกูลกิจ 33100275
นาย วิบูลย์ ปิยวัฒนเมธา 33100349
อาจารย์ที่ปรึกษา ผศ.ดร. กิตติ ไพฑูรย์วัฒนกิจ
อาจารย์ ขนิษฐา แซ่ตั้ง

ปีการศึกษา 2536

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้เป็นการศึกษาทฤษฎี Fuzzy Logic โดยศึกษาเกี่ยวกับความเป็นมาทางทฤษฎี Fuzzy Logic และการนำการคำนวณแบบ Fuzzy Logic มาประยุกต์ใช้งานกับเครื่องทำน้ำอุ่นที่ใช้อยู่ทั่วๆไปตามบ้าน โดยส่วนควบคุมจะใช้ไมโครคอนโทรลเลอร์เบอร์ 8031 ตระกูล MCS-51 ของ Intel มาเป็นตัวควบคุมอุณหภูมิโดยจะมีการเลือกฟังค์ชั่นการทำงานต่างๆ โครงงานนี้จะเน้นให้เห็นความสำคัญ ของการผสมผสานระหว่างทฤษฎี Fuzzy Logic กับงานด้านอิเลคทรอนิคส์ และอาจจะใช้เป็นแนวทางการศึกษาแก่ผู้ที่สนใจที่จะนำเอาการควบคุมแบบ Fuzzy Logic ไปควบคุมอุปกรณ์ชนิดอื่นต่อไป

Abstract

This thesis presents basics of fuzzy logic theory what fuzzy logic is and where it comes from. In addition it will cover some detials in mathematics of fuzzy logic which will provide the trace to the application of fuzzy logic theory in electronic control system. In this project, we have applied fuzzy theory in electric showers used in almot every household to provide us the most optimal demand in controlling water temperature.

We have chosen MCS-51 microcontroller from Intel Corparation to control the electric shower by using liquid crystal to display functions of this system. The scope of this project is mainly enunsiated in the harmony of fuzzy logic theory and electronic control system.

Finally, it is hoped that this thesis will become a source for anyone interested in the above matters.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 Concepts of Fuzzy Logic	
2.1 Fuzzy Logic คืออะไร	2
2.2 Fuzzy Logic ทำงานอย่างไร	2
2.3 ข้อได้เปรียบของทฤษฎี Fuzzy Logic	4
2.4 Fuzzy Model	4
บทที่ 3 Fuzzy Logic In Controlling Electric Shower	
3.1 HARDWARE	8
3.2 SOFTWARE	20
3 .3 TIMER/COUNTER	29
3.4 INTERRUPTS	34
3.5 FLOW CHART	41
3.6 PROGRAM ASSEMBLY LANGUAGE	46
บทที่ 4 ผลการทดลอง	79
บทที่ 5 สรุปและวิจารณ์	80
กิตติกรรมประกาศ	81
เอกสารอ้างอิง	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

Fuzzy logic เป็นศาสตร์ที่มีมากว่า 30 ปีแล้วแต่พึ่งจะมาเป็นที่นิยมกันอย่างแพร่หลายเมื่อไม่นานมานี้ สิ่งที่ทำให้การควบคุมด้วยระบบ Fuzzy Logic เป็นที่สนใจของชาวโลกคือ ในปี 1987 ที่ประเทศญี่ปุ่น ได้ประสบความสำเร็จในการควบคุมการจราจรทางรถไฟด้วยทฤษฎี Fuzzy Logic ที่ Sendai นับจากจุดนั้นมาบริษัทและมหาวิทยาลัยหลายแห่งในประเทศญี่ปุ่น ได้มีการศึกษาอย่างจริงจังในศาสตร์แขนงนี้ จนสามารถทำให้นำทฤษฎีนี้ มาใช้ในอุปกรณ์เครื่องใช้ไฟฟ้าต่างๆมากมาย เช่น ใช้ในเครื่องปรับอากาศ กล้องวีดีโอ หม้อหุงข้าว และกล้องถ่ายภาพ เป็นต้นและในปัจจุบันนี้ยังได้นำไปประยุกต์ใช้ใน Expert System เช่น ควบคุมอุณหภูมิของหอกลั่นน้ำมัน ในโรงกลั่น การควบคุมระบบ Automatic Transmission ในรถยนต์เป็นต้น ซึ่งการควบคุมเหล่านี้ สามารถทำให้ได้ผลลัพธ์ใกล้เคียงผลลัพธ์ซึ่งเกิดจากการตัดสินใจของมนุษย์ทำให้ผลการควบคุมออกมาเป็นที่น่าพอใจเป็นอย่างยิ่ง

วงจรควบคุมการทำงานของเครื่องทำน้ำอุ่นที่สร้างขึ้นนี้ ใช้หลักการทำงานของ Fuzzy Logic มาประยุกต์ในการควบคุมการทำงานของเครื่องทำน้ำอุ่น นอกจากนี้ทฤษฎีนี้ยังสามารถนำไปใช้งานได้อีกในงานหลายๆ ด้าน ทางผู้จัดทำหวังว่าการทำโครงงานชิ้นนี้จะเป็นประโยชน์ต่อผู้ที่สนใจทฤษฎีนี้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. CONCEPTS OF FUZZY LOGIC

2.1 คำว่า "Fuzzy" คืออะไร?

คำนี้มาจากคำว่า "Fuzz" ซึ่งหมายถึงขนอ่อนที่ปกคลุมตัวลูกไก่ คำว่า "Fuzzy" ในภาษาอังกฤษมีความหมายว่า"ไม่ชัดเจน มัว" คำนี้ตรงกับภาษาฝรั่งเศสว่า "Flou" ส่วนญี่ปุ่นนั้นใช้คำว่า "Aimai" อย่างไรก็ดี ในเชิงวิชาการและเทคโนโลยี คำว่า "Fuzzy" นี้เป็นศัพท์เทคนิคโดยเฉพาะ และหมายความถึงสภาพแห่งความกำกวมหรือคลุมเครือที่มีฐานมาจากสัญชาติญาณมนุษย์มากเสียยิ่งกว่าจากเรื่องของ Probability (ความเป็นไปได้หรือความน่าจะเป็น))

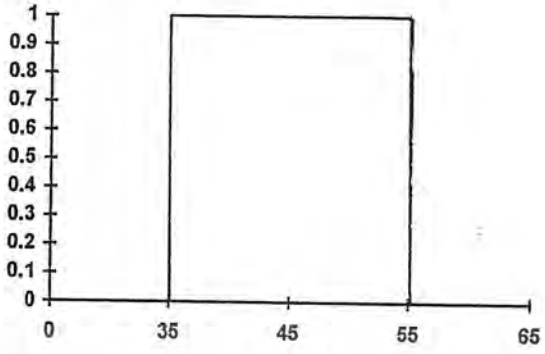
ประมาณ 30 ปีมาแล้วที่ศาสตราจารย์ ลอตฟี เอ ซาเดห์ (L.A. Zadeh) ได้คิดค้น Fuzzy Sets เพื่อต้องการปรับแนวความคิดเรื่องขอบเขตของ Fuzzy เข้ากับวิทยาศาสตร์ ทฤษฎี Fuzzy จึงรับการสร้างขึ้นบนรากฐานของ Fuzzy Sets นี้และแล้ววิศวกรรมศาสตร์แขนงใหม่ภายใต้ชื่อว่า "Fuzzy Engineering" ก็เกิดขึ้น แม้ Fuzzy Sets อาจจะฟังดูว่าเกี่ยวกับคณิตศาสตร์มากไปหน่อยก็ตาม แต่แนวความคิดพื้นฐานสามารถอธิบายได้โดยไม่ยากเลยแม้แต่น้อย

2.2 ทฤษฎี Fuzzy ทำงานอย่างไร?

Fuzzy Sets เป็นทฤษฎีทางคณิตศาสตร์ที่ใช้เป็นเครื่องมือในการตัดสินใจ ซึ่งในการตัดสินใจที่ใช้ในระบบ นี้จะเลียนแบบการตัดสินใจของมนุษย์ คือจะเป็นการประมาณ (Approximation) จากปัจจัยหรือตัวแปรต่างๆของระบบเพื่อทำการสรุปหาเหตุผล ดังนั้นฟัซซี่ลอจิกจึงไม่ใช่ลอจิกที่มีเพียง 0 กับ 1 หรือเราเรียกว่า Crisp Sets ดังแสดงในรูปที่ 2.2.1 เท่านั้น แต่เป็นลอจิกที่แทนความหมายของตัวแปร โดยบอกถึงระดับความเป็นสมาชิก (Membership Function, $0 \le \mu \le 1$) ของตัวแปรนั้นๆ ดังแสดงในรูปที่ 2.2.2

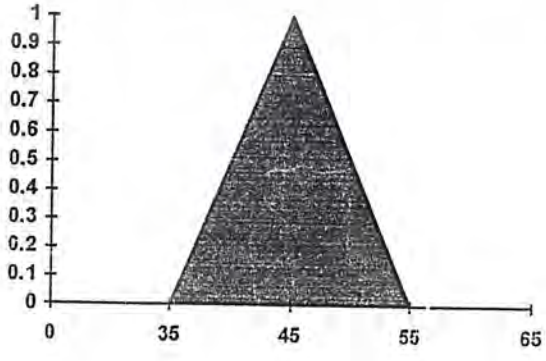
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.2.1 แสดงกราฟ Crisp Sets ของช่วงอายุวัยกลางคน



เราลองมาดูตัวอย่างจากการนึกมโนภาพจากคำว่า "วัยกลางคน" ดู คำดังกล่าวนี้ชวนให้นึกถึงภาพ ๆ หนึ่งขึ้นมาในสมอง แต่ก็เป็นภาพที่มีขอบเขตคลุมเครือ คือไม่อาจทราบค่าที่แน่นอนได้ จึงไม่อาจจะใช้ระบบไบนารี่ในเครื่องคอมพิวเตอร์ธรรมดามาแก้ปัญหานี้ได้ จุดนี้เองที่ ทฤษฎี Fuzzy เข้ามาช่วยในการคำนวณหาขอบเขต โดยสมมติว่าเราสรุปว่า วัยกลางคนหมายถึงอายุ 45 ปี แต่คนอายุ 35 หรือ 55 ก็ไม่อาจจะจัดอยู่ข่าย "ไม่ใช่วัยกลางคนได้" ดังนั้นความหมายของคำว่า "วัยกลางคน" นั้นค่อนข้างจะแตกต่างกันภายในกรอบแห่งขอบเขตนั้น ในทางตรงกันข้าม ผู้ที่มีอายุน้อยกว่า 30 หรือมากกว่า 60 ก็เรียกได้ว่า "ไม่อยู่ในข่ายวัยกลางคนแน่" ความคิดนี้แสดงออกมาได้ตามกราฟรูปที่ 2.2.2 หรือตามฟังชั่นที่เรียกว่า "ฟังชั่นความเป็นสมาชิก" (Membership Function) ที่มีค่าเกรดระหว่าง 0 และ 1 โดย Fuzzy Sets นี้ก็แสดงออกในรูปของฟังชั่นความเป็นสมาชิกนี้ อย่างไรก็ดี สังเกตุด้วยว่า ค่าเกรดระหว่างฟังชั่นความเป็นสมาชิก สามารถจะแปรเปลี่ยนได้ระหว่าง 0 และ 1 นี่เองทำให้การแสดงออกมาเป็นรูปธรรมที่เป็นตัวเลขได้ดังแสดงในรูปที่ 2.2.2

รูปที่ 2.2.2 แสดงกราฟ Fuzzy Sets ของช่วงอายุวัยกลางคน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.2.1 คนอายุ 34 และ 56 จะไม่ถูกจัดเป็น "วัยกลางคน" แต่ใครก็ตามที่ตอนนี้ ที่ยังคิดว่าตัวเองหนุ่มด้วยอายุแค่ 34 ก็จะย่างเข้า "วัยกลางคน" ทันทีเมื่อถึงวันเกิดในปีหน้า ลักษณะที่ค่อนข้างจะไม่ยืดหยุ่นนี้ เกิดขึ้นก็เป็นเพราะ การกำหนดค่าที่ไม่มีความยืดหยุ่น แนวความคิดที่มีการกำหนดค่าชัดเจนในรูป 0 หรือ 1 เช่นเดียวกับระบบไบนารี่ที่ใช้กันในคอมพิวเตอร์ธรรมดาทั่วไปนั้น ทำงานโดยกำหนดช่วงตายตัว (Fixed Range) โดยที่เราได้กำหนดไว้ว่า 'วัยกลางคน'คือช่วงระหว่าง 35 ถึง 55 ปี ลักษณะเช่นนี้เรียกว่า Crisp Sets ซึ่งจะมีลักษณะที่ตรงกันข้ามกับ Fuzzy Sets อย่างสิ้นเชิง

2.3 ข้อได้เปรียบของ Fuzzy Logic เหนือ Traditional Logic

Fuzzy logic theory มีข้อได้เปรียบเหนือ Traditional Logic หรือลอจิกแบบทั่วไปอยู่ 2 ประการ คือ

ประการแรก Fuzzy Sets สามารถให้คำจำกัดความของคุณภาพทางภาษา (Qualitatively Using Linguistic Terms) ได้ดี เช่น คำว่า ค่อนข้าง หรือ เกือบจะ ' ประกอบกับคำแสดงคุณลักษณะทั่วๆไป เช่น สูง,เตี้ย,ดี,เลว,ร้อน,เย็น และอื่นๆ และยังสามารถให้ค่าความสำคัญของแต่ละสมาชิกใน Fuzzy Sets ได้ (Degrees of Membership)

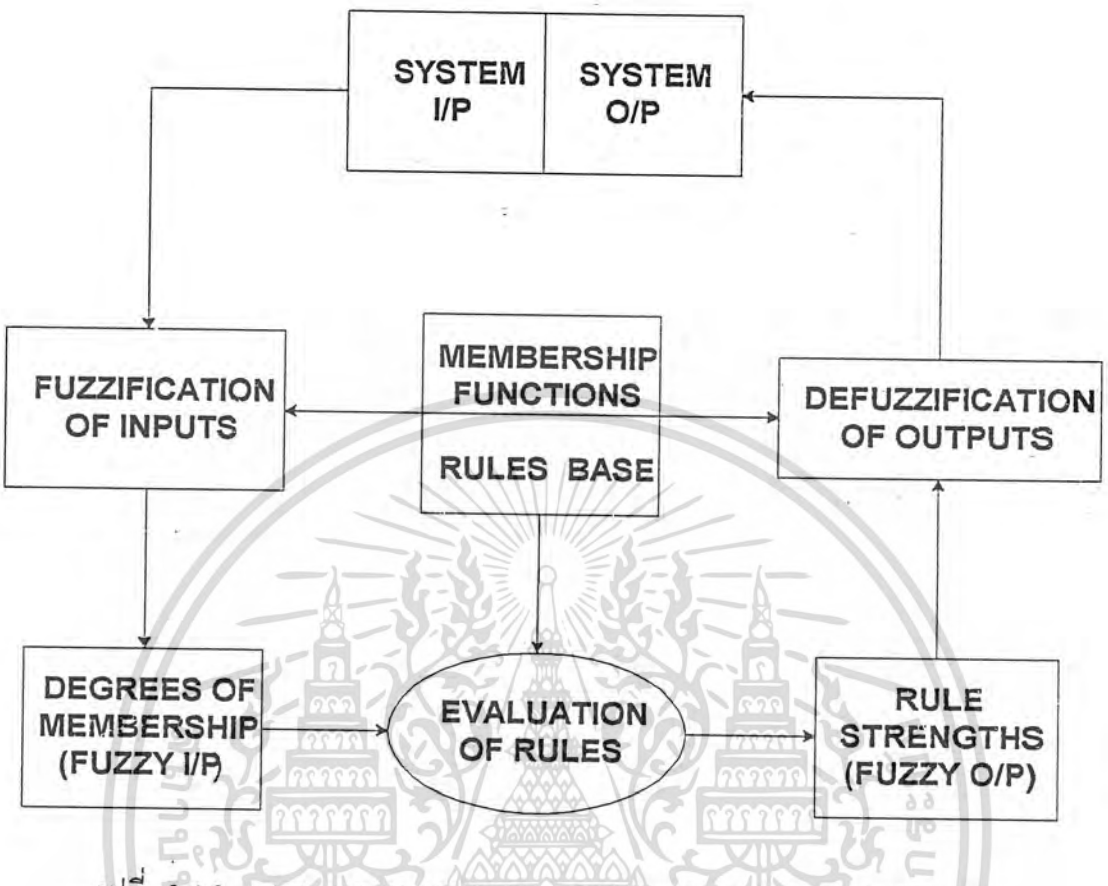
ประการที่ 2 เอาท์พุทที่ได้จาก Fuzzy System จะมีการเปลี่ยนแปลงแบบค่อยเป็นค่อยไปอย่างต่อเนื่อง ถึงแม้ว่า อินพุทของ Fuzzy System จะเปลี่ยนไปอย่างทันทีทันใดก็ตาม

2.4 Fuzzy Model

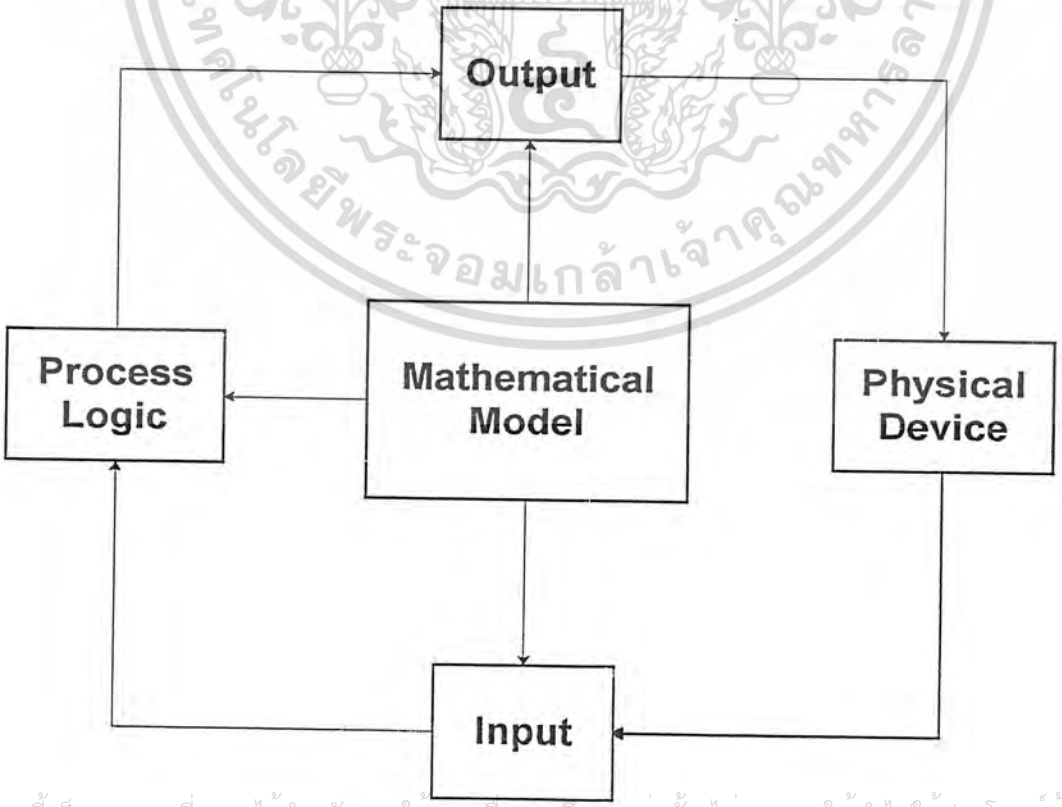
องค์ประกอบของระบบควบคุมแบบฟัซซี่ลอจิก และระบบควบคุมที่ใช้กันอยู่ในปัจจุบัน จะมีลักษณะแตกต่างกันไม่มากดังแสดงในรูปที่ 2.4.1 และ 2.4.2 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.4.1 แสดงระบบควบคุมโดยใช้ทฤษฎีฟัซซี่ลอจิก



รูปที่ 2.4.2 แสดงระบบควบคุมแบบ PID (Proportional Integral Derivative)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนหลักๆที่ได้เพิ่มเข้ามาใช้สำหระบระบบควบคุมที่ใช้ระบบฟัซซี่ลอจิก คือ

2.4.1 Fuzzification of Inputs คือขั้นตอนที่ทำหน้าที่แปลงข้อมูลที่เข้ามา ไปเป็นอินพุทของระบบฟัซซี่ ซึ่งจะอยู่ในรูปของค่าความเป็นสมาชิก (Degree of Membership) ในฟัซซี่เซต ดังที่ได้กล่าวมาแล้วในตอนต้น

2.4.2 Rule Evaluation คือขั้นตอนการตีความหรือวิเคราะห์ซึ่งใช้เทคนิคของ 'and Minimum หรือ 'or' โดยใช้ Maximum ของตัวแปรส่วนเงื่อนไข (Condition) ในการคำนวณผลลัพธ์รวมจากกฎ IF/THEN RULES ของ Fuzzy System ซึ่งโดยปกติจะใช้เทคนิคของกฎ Minimum หรือกฎ Smallest เพื่อที่จะทำให้ความแข็งแกร่งของกฎที่ได้เลือกอยู่นั้นขึ้นกับค่าของตัวแปรส่วนเงื่อนไข (Condition) ที่มีค่าความเป็นสมาชิกน้อยที่สุด

โดยลักษณะของ IF/THEN Rules มีดังนี้

If เงื่อนไข Then ผลลัพธ์

หรือ If เงื่อนไข (1) และ เงื่อนไข (2) Then ผลลัพธ์

2.4.3 Defuzzification of Outputs คือขั้นตอนที่ทำการเปรียบเทียบ Fuzzy Output ทั้งหมดเข้ากับผลลัพธ์รวมซึ่งในยุคแรกๆของระบบ Fuzzy จะใช้ Fuzzy Output ที่มีค่ามากที่สุดในระบบเป็นผลลัพธ์ของระบบ (Maximum Defuzzification) ซึ่งในปัจจุบันวิธีการนี้ไม่ได้นิยม เพราะว่าจะมีการละทิ้งผลลัพธ์ของกฎบางกฎ จึงเปลี่ยนมาใช้วิธีหา Centroid (Center of Gravity) เพราะสามารถถ่วงน้ำหนักของ Fuzzy Outputs ได้อย่างทั่วถึง โดยวิธี Centroid ทำได้โดย

$$I = \frac{\sum_{i=1}^{n} (u_n \times U_n)}{\sum_{i=1}^{n} u_n}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย

I = RESULT OF CENTROID METHOD

u_n = DEGREE OF MEMBERSHIP

Un = STATUS FROM IF/THEN RULES

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การควบคุมเครื่องทำน้ำอุ่นโดยใช้ Fuzzy Theory

3.1 ขั้นตอนการออกแบบ HARDWARE ของระบบ

3.1.1 โครงสร้างทางสถาปัตยกรรม MCS-51

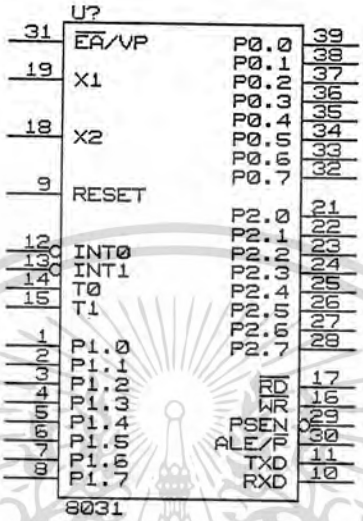
ลักษณะโดยทั่วไปของ MCS-51 จะประกอบด้วย

1. ใช้ HMOS และ CHMOS เทคโนโลยีในการสร้างและทำงานด้วยแหล่งจ่ายไฟขนาด 5 V. เพียงแหล่งเดียว
2. CPU มีขนาดคำ 8 บิต
3. มีวงจรออสซิเลเตอร์ และวงจรนาฬิกาบน Chip
4. ชุดแบงค์รีจิสเตอร์มี 4 ชุด
5. มีตัวจับเวลา/ตัวนับ ขนาด 16 บิต 2 ชุด
6. มีพอร์ตไอโอแบบขนาน 2 ทิศทางจำนวน 4 พอร์ตๆละ 8 บิต
7. พอร์ทอนุกรมสามารถที่จะรับส่งแบบ Full Duplex ที่ความเร็วสูง
8. 1 วัฏจักรคำสั่งกินเวลา 1 ไมโครวินาที ด้วยการใช้คริสทัล 12 เม็กกะเฮริตซ์
9. แอดเดรสข้อมูลภายนอกได้ 64 กิโลไบต์
10. แอดเดรสโปรแกรมภายนอกได้ 64 กิโลไบต์
11. สามารถกำหนดเลขที่อยู่ขนาดไบต์หรือบิตได้โดยตรง
12. มีซอฟแวร์บิตแฟลกสำหรับผู้ใช้ที่จะกำหนดเองได้ถึง 128 ตำแหน่งบิต
13. โครงสร้างอินเตอร์รัพต์จะติดตั้งได้ถึง 5 แหล่ง
14. ตัวโพรเซสเซอร์สามารถใช้งานแบบบูลีน
15. มีคำสั่งคูณ และหารทางฮาร์ดแวร์ที่ทำได้ภายใน 4 ไมโครวินาที
16. ตัวเลขทางคณิตศาสตร์ ใช้ได้ทั้งระบบไบนารี่และระบบเดซิมัล
17. การใช้พื้นที่สแตกสำหรับโปรแกรมย่อยต่างๆ ทำได้กว้างกว่า MCS-48
18. ชุดคำสั่งของ MCS-51 จะมีความสามารถสูงกว่า MCS-48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.1 การจัดขาลักษณะภายนอกของ MCS-51

รูปที่ 3.1.1 แสดงลักษณะการจัดขาของ MCS-51



ลักษณะขาตามสถาปัตยกรรมของ MCS-51 อยู่ 4 กลุ่ม คือ

1. กลุ่มขารับแหล่งจ่ายไฟฟ้า และระบบสัญญาณนาฬิกา
2. กลุ่มขาแอดเดรสและข้อมูล
3. กลุ่มขาควบคุม
4. กลุ่มขาพอร์ตแบบขนานและแบบอนุกรม

ขาบางขาจะหน้าที่ได้สองหน้าที่ขึ้นอยู่กับการติดตั้งด้วยซอฟแวร์หรือฮาร์ดแวร์ เช่น ขาที่ 32-39 จะทำหน้าที่ได้เป็นกลุ่มขาแอดเดรสและข้อมูลหรือจะทำหน้าที่เป็นกลุ่มขาพอร์ตแบบขนานรายละเอียดหน้าที่แต่ละขามีดังนี้

ขา Vss (ขา 20) เป็นขาสำหรับต่อลงดิน

ขา Vcc (ขา 40) เป็นขาที่ต่อแรงดันไฟ 5 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ 033219

ขา PORT0 (ขา 32-39) ทำหน้าที่เป็นพอร์ทไอโอ 8 บิต การเขียน 1 ไปที่พอร์ตนี้จะเป็นการปล่อยลอย ทำให้มันมีหน้าที่เป็นอินพุท พอร์ต 0 จะทำหน้าที่เป็นมัลติเพลกซ์ ด้วยสัญญาณแอดเดรสไบต์ต่ำกับบัสข้อมูล

ขา PORT 1 (ขา 1-8) เป็นพอร์ตไอโอ 8 บิต เมื่อถูกเขียนค่า '1' ด้วยโปรแกรมมันจะมีสถานะสูง การให้สถานะเช่นนี้จะเป็นการ Initial การใช้งานพอร์ตเป็น Input 'ขณะที่ทำหน้าที่เป็นอินพุต การให้สัญญาณลงต่ำจะเป็นการจ่ายกระแสออกเนื่องจากพูลอัพภายใน

ขา PORT 2 (ขา 21-28) เป็นพอร์ตไอโอ 8 บิต ด้วยการพูลอัพภายใน พอร์ต 2 ทำหน้าที่เป็นบัฟเฟอร์เอาท์พุต อีกหน้าที่หนึ่งจะเป็นตัวส่งแอดเดรสไบต์สูง

ขา PORT 3 (ขา 10-17) เป็นพอร์ตไอโอ 8 บิตแบบพูลอัพภายใน แล้วยังมีอีกหน้าที่หนึ่งตามตารางที่ 3.1.1 ข้างล่างด้วย

ตารางที่ 3.1.1 แสดงหน้าที่พิเศษของ PORT 3

ขาพอร์ต	ขา	การทำงานตามฟังค์ชั่นพิเศษ
P3.0	10	RxD พอร์ตอนุกรมอินพุต
P3.1	11	TxD พอร์ตอนุกรมเอาท์พุต
P3.2	12	INT0 อินเตอร์รัพต์ภายนอกตัวที่ 1
P3.3	13	INT1 อินเตอร์รัพต์ภายนอกตัวที่ 2
P3.4	14	T0 สัญญาณกระตุ้นเข้าที่ตัวจับเวลา/ตัวนับ 0
P3.5	15	T1 สัญญาณกระตุ้นเข้าที่ตัวจับเวลา/ตัวนับ 1
P3.6	16	WR สัญญาณควบคุมการเขียน
P3.7	17	RD สัญญาณควบคุมการอ่าน

การที่จะให้ทำงานตามฟังค์ชั่นข้างบนได้ จะต้องติดตั้งโปรแกรม ด้วยการส่งค่าหนึ่งไปแลตช์ไว้ก่อนที่ให้ทำงานตามฟังค์ชั่นข้างบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา RST (ขา 9) จะต้องสถานะสูงเป็นเวลาอย่างน้อย 2 วัฏจักรระหว่างที่ออสซิลเลเตอร์ทำงานขณะที่ต้องรีเซตทั้งระบบงาน โดยจะต่อตัวรีจิสเตอร์พูลดาวน์ จากขา RST ลงดิน และเพื่อให้ตัวชิปรีเซตได้โดยอัตโนมัติ ขณะเปิดไฟจะใช้คาปาซิเตอร์ ต่อคร่อมระหว่างขา RST กับขา Vcc

ขา ALE/PROG (ขา 30) เป็นขาแอดเดรสแลทช์อีนาเบิ้ล (Addres Latch Enable) ด้วยการส่งพัลซ์ออกไปใช้สำหรับแลทช์ค่าแอดเดรสต่ำจากพอร์ต 0 ในระหว่างการเข้าถึงข้อมูลจากหน่วยความจำภายใน ALE จะถูกส่งสัญญาณนาฬิกาออกมา ในอัตราความเร็วคงที่ ที่ 1/8 ของความถี่ออสซิลเลเตอร์ตลอดเวลา

ขา PSEN (ขา 29) Program Storage Enable เป็นสโตรบอ่านข้อมูลจากโปรแกรมหน่วยความจำภายนอก เมื่อชิปทำงานด้วยโปรแกรมภายนอก

ขา EA/Vpp (ขา 31) External Accesมีสถานะสูง ตัวซีพียูในชิปจะทำงานตามโปรแกรมที่อยู่ในหน่วยความจำภายใน

ขา XTAL 1 (ขา 19) ใช้เป็นตัวอินพุตเข้าสู่ตัวออสซิลเลเตอร์ขยายแบบ Invert

ขา XTAL 2 (ขา 18) ใช้เป็นตัวเอาท์พุตจากตัวออสซิลเลเตอร์ขยายแบบ Invert

3.1.1.2 การจัดการหน่วยความจำ โหมดการกำหนดเลขที่อยู่

สถาปัตยกรรมของ MCS-51 ได้แบ่งหน่วยความจำมาให้บนชิปพร้อมทั้งสามารถที่จะขยายหน่ายความจำภายนอกได้ โดยทั่วไปการกำหนดเลขที่อยู่ของเบอร์นี้ จะมีคำสั่งเพื่อช่วยให้ทำงานได้รวดเร็ว และสามารถใช้คำสั่งได้อย่างมีประสิทธิภาพ

3.1.1.3 การจัดการหน่วยความจำ

MCS-51 แบ่งตามพื้นฐานหน่วยความจำของการกำหนดเลขที่อยู่แอดเดรส ได้เป็น 3 ส่วนที่ประกอบด้วยเนื้อที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) 64 กิโลไบต์หน่วยความจำโปรแกรม
2) 64 กิโลไบต์ หน่วยความจำข้อมูลภายนอก
3) 256 ไบต์เป็นหน่วยความจำข้อมูลภายใน

3.1.1.4 เนื้อที่หน่วยความจำโปรแกรม

ประกอบด้วย ส่วนภายในและภายนอกชิป ถ้าขา EA มีสถานะสูง MCS - 51 จะบริการโปรแกรมภายใน โดยโปรแกรมจะมีความยาวไม่เกิน 4 K และถ้า EA มีสถานะต่ำจะแฟทซ์ข้อมูลภายนอกทุกกรณี ตัวนับโปรแกรม 16 บิตจะเป็นตัวกำหนดเลขที่อยู่โปรแกรม

3.1.1.5 เนื้อที่หน่วยความจำข้อมูล

ประกอบด้วยความจำข้อมูลภายในและภายนอก หน่วยความจำข้อมูลภายนอก จะเข้าถึงได้ด้วยการใช้คำ สั่ง MOVX

การเข้าถึงของข้อมูลในรูปของคำสั่งไมโครคอนโทรลเลอร์จะแบ่งเป็นลักษณะงานได้ดังนี้

1) การกำหนดตำแหน่งที่อยู่โดยรีจิสเตอร์
2) การกำหนดตำแหน่งที่อยู่โดยตรง
3) การกำหนดตำแหน่งที่อยู่โดยอ้อมด้วยรีจิสเตอร์
4) การกำหนดตำแหน่งที่อยู่โดยทันที

3.1.1.6 โหมดการกำหนดเลขที่อยู่ข้อมูล

คำสั่งภาษาแอสเซมบลี้ MCS-51 จะประกอบด้วยนิวมอนิก รหัสการทำงานหนึ่งไบต์ และตัวโอเปอแรนด์อีก ตั้งแต่ 0-3 ไบต์ โดยการใช้ค่าสัญลักษณ์ต่างๆที่ถูกแบ่งด้วย (,) คำสั่งที่ใช้โอเปอแรนด์สองสัญลักษณ์จะให้ตัวแรกเป็นตัวรับการถ่ายเทข้อมูล ส่วนตัวหลังจะเป็นตัวส่งข้อมูล คำสั่งต่างๆส่วนใหญ่นั้น จะทำงานโดยใช้แอคคิวมูเลเตอร์ เป็นตัวหลักในการส่งข้อมูล ในโอเปอร์แรนด์ และจะเป็นตัวรับผลลัพธ์ที่เกิดขึ้นหลังการทำงานคำสั่งนั้นๆ ด้วยการให้ตัว A เป็นตัวกำหนดการส่งข้อมูลหรือรับการถ่ายเทข้อมูลในโอเปอแรนด์ฟิลด์ตัวอย่างเช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADD A, {ตัวส่งข้อมูล}

คำสั่งนี้จะเป็นการบวกเอาข้อมูลที่อยู่ในตัวส่งข้อมูลกับข้อมูลในแอคคูมิวเลเตอร์ และผลรวมจะเก็บกลับที่แอกคูมิวเลเตอร์

การใช้งานในลักษณะ {ตัวส่งข้อมูล} ในคำสั่งตัวอย่างนี้ สามารถที่จะเลือกใช้ได้ตามลักษณะการทำงานเลขที่อยู่ได้ 4 โหมดด้วยกัน คือ

1. การกำหนดเลขที่อยู่รีจิสเตอร์ ด้วยการใช้ข้อมูลในเรจิสเตอร์ทำงานในแบงค์ที่ถูกเลือกในขณะนั้น

2. การกำหนดเลขที่อยู่โดยตรง ด้วยการใช้เลขที่อยู่ตามตำแหน่ง RAM ภายในพอร์ตไอโอหรือกลุ่มรีจิสเตอร์ SFR (Special Function Registers)

3. การกำหนดเลขที่อยู่รีจิสเตอร์โดยอ้อม ด้วยการใช้ค่าในรีจิสเตอร์ทำงาน เป็นตัวชี้ตำแหน่ง ค่าข้อมูลของ RAM ในชิป

4. การกำหนดเลขที่อยู่รีจิสเตอร์โดยทันทีด้วยการใช้ข้อมูลขนาด 1 ไบต์คงที่ภายในคำสั่ง

โหมด 3 โหมดแรกเป็นการเข้าถึง RAM ภายในและวงจรรีจิสเตอร์ต่างๆ ภายในตามโครงสร้างทางสถาปัตยกรรมของ MCS-51 นอกจากจะใช้เป็นตัวส่งข้อมูลในโอเปอร์แรนด์ฟิลด์แล้ว ยังใช้เป็นตัวรับการถ่ายเทข้อมูลในโอเปอแรนด์ฟิลด์อีกด้วย ส่วนโหมดที่ 4 เนื่องจากโอเปอแรนด์ฟิลด์เป็นข้อมูลคงที่ จึงต้องเป็นตัวส่งข้อมูลคงที่อย่างเดียวเท่านั้น

3.1.1.7 การกำหนดที่อยู่แบบรีจิสเตอร์

จะให้ข้อมูลเข้าถึง โดยการใช้รีจิสเตอร์กลุ่มที่ถูกเลือกจาก การติดตั้งการใช้งานในกลุ่มรีจิสเตอร์แบงค์นั้นๆ โดยแต่ละแบงค์จะมี 8 รีจิสเตอร์ และใช้บิตต่ำสุดสามบิตแรกของรหัสคำสั่งเป็นตัวกำหนดเลือกใช้รีจิสเตอร์ใช้งานตัวใดตัวหนึ่งใน 8 ตัว ทั้งสามบิตนี้จะรวมอยู่ในรหัสออปโค้ดกับโอเปอร์แรนด์ ตำแหน่งรวมอยู่ในรูปแบบตำแหน่งสั้นๆ เพียงหนึ่งไบต์ โดยมีรูปแบบออปโค้ดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OPCODE	n	n	n

เช่น MOV A, R0 ; ย้ายค่าใน R0 ไปไว้ที่แอคคูมิวเลเตอร์

3.1.1.8 การกำหนดเลขที่อยู่โดยตรง

เป็นวิธีเดียวที่จะเข้าถึงข้อมูลทางฮาร์ดแวร์รีจิสเตอร์เช่น กลุ่มของ SFR และสามารถกำหนดเลขที่อยู่ โดยตรงบริเวณตำแหน่งต่างๆของแรมภายในจำนวน 128 ไบต์ ด้วยการใช้ไบต์โอเปอแรนด์ตัวต่อจากออป โค๊ดคำสั่ง เป็นตัวกำหนดตำแหน่งที่ถูกใช้ โดยมีรูปแบบออปโค๊ดดังนี้

OPCODE	DIRECT ADDRESS

เช่น MOV A, 40H ; เป็นการย้ายข้อมูลที่ตำแหน่ง 40H ไปไว้ที่ ACC

3.1.1.9 การกำหนดเลขที่อยู่รีจิสเตอร์โดยอ้อม

การกำหนดเลขที่อยู่โดยอ้อมใน MCS-51 ด้วยการใช้ค่าข้อมูลที่อยู่ใน R0 หรือ R1 ในแบงค์ที่ถูกติดตั้งให้ทำ งานเท่านั้นจะเป็นค่าดัชนี หรือ ตัวชี้ตำแหน่งข้อมูลภายใน 256 ไบต์ แบ่งเป็นบล็อกต่ำ จำนวน 128 ไบต์ ของแรมภายในที่สามารถกำหนดตัวชี้ตำแหน่งด้วยค่า R0 และ R1 ด้วยการเลือกการทำงานนี้ โดยมีรูป แบบออปโค๊ดดังนี้

OPCODE	i

เช่น MOV A, @R0 ; ย้ายค่าข้อมูลในหน่วยความจำที่ถูกชี้ด้วย R0

3.1.1.10 การกำหนดเลขที่อยู่โดยทันที

เมื่อแหล่งกำเนิดตัวโอเปอร์แรนด์เป็นค่าคงที่มากกว่าที่จะเป็นตัวแปร ดังนั้นตัวคงที่ที่ใช้ในขณะนั้นจะเป็นส่วน หนึ่งของรหัสคำสั่งรวมอยู่ในรูปแบบคำสั่งดังนี้

OPCODE	DATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น MOV A, #12

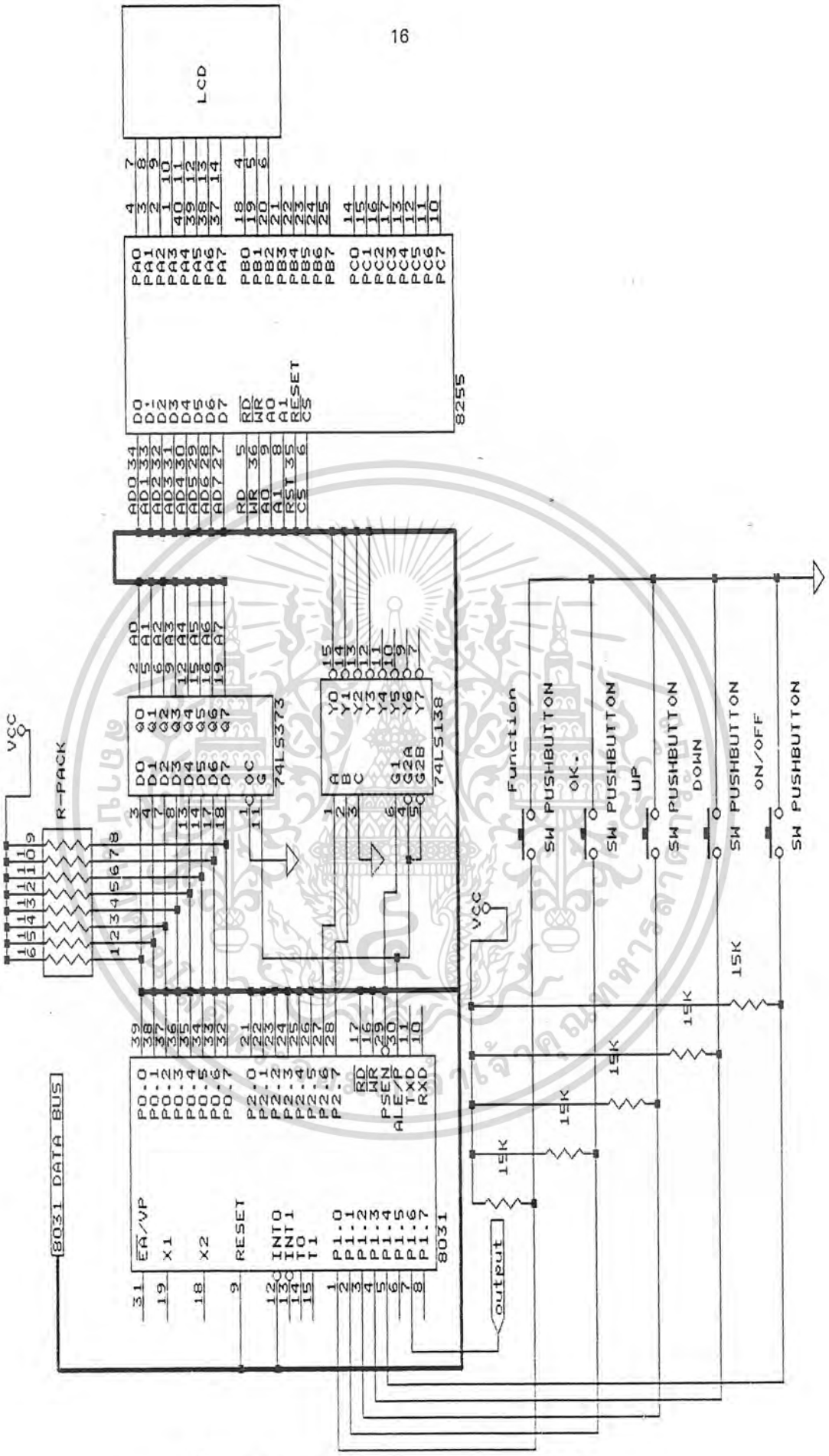
ADD A, #34

3.1.2 ในส่วนการติดต่อกับผู้ใช้ จะประกอบด้วย ส่วนการแสดงผล และส่วนสวิทซ์เพื่อรับคำสั่งจากผู้ใช้ ดังแสดงในรูปที่ 3.1.2 ส่วนการแสดงผลจะใช้แผงแสดงผลแบบผลึกเหลว (Liquid Crystal Display) ตัว LCD นี้จะต่ออยู่กับ PORT A ของ 8255 โดยใช้ตัวควบคุมการแสดงผลของ (Hitachi) เพื่อใช้แสดงผลการวัดอุณหภูมิ และ สถานะของระบบในขณะนั้น และส่วนสวิทซ์จะมีอยู่ 5 ตัวโดยจะทำหน้าที่ดังนี้คือ ON/OFF, FUNCTION, INCREMENT, DECREMENT และ O.K.

3.1.3 ในวงจรการวัดอุณหภูมิ ใช้ I.C. LM3911 เป็นตัววัดอุณภูมิ โดยจะมีการเปลียนแปลงเท่ากับ -10 mV/Celcius แล้วทำการปรับหรือ calibrate ค่า resistor 4 ค่าที่ต่อในวงจรจนทำให้การแสดงผลอุณหภูมิถูกต้องที่สุด ความละเอียดในการวัด อุณหภูมิของวงจรนี้เท่ากับ 0.1 Celcius เอาท์พุตที่ได้จากวงจรนี้นำไปเข้า Pin No. 5 ของวงจร Analog to digital แบบ 8 บิท(ADC 0804) ซึ่งสามารถแสดงค่าอุณหภูมิทั้งหมดได้ 256 ค่าที่แตกต่างกัน เพราะฉะนั้น Span Temperature จะมีค่าเท่ากับ 25.6 Celcius ดังนั้นช่วงอุณหภูมิที่แสดงได้ จะอยู่ระหว่าง 23 ถึง 48.6 Celcius โดยกำหนดให้ T_o = 23 Celcius นำเอาท์พุทที่ได้จาก Analog to Digital มาต่อเข้ากับ Data Bus ของ Microcontroller (Port 0 ทั้งหมด) ได้โดยตรง รูปวงจรทั้งหมดของส่วนนี้แสดงอยู่ในรูปที่ 3.1.3

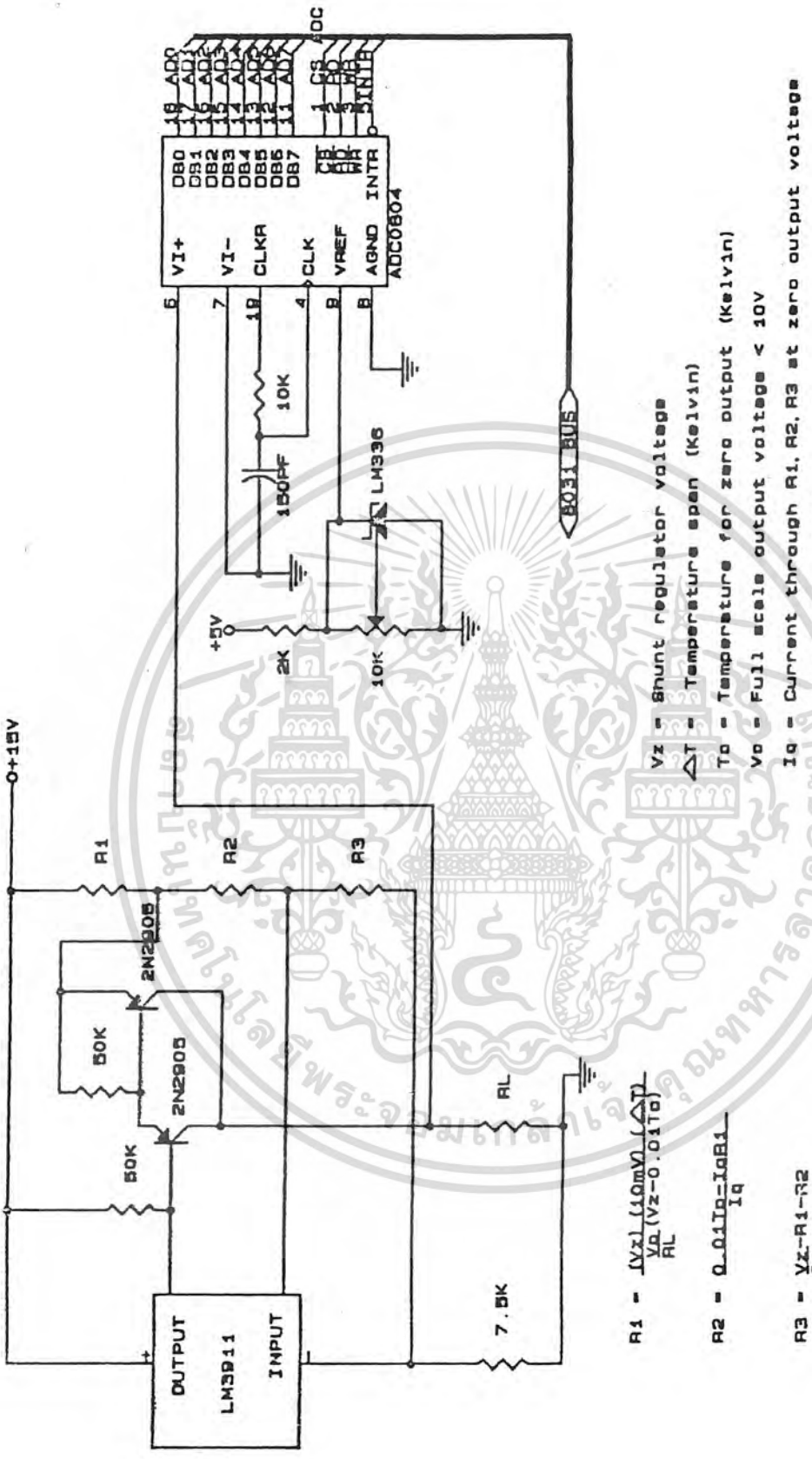
3.1.4 ในการควบคุมตัว Heater ของเครื่องทำน้ำอุ่น เราจะใช้ Microtroller ทำงานร่วมกับ Optoisolator Triac MOC 3043 ต่อเข้ากับขา P1.5 โดยควบคุมการ ON และ OFF ของ heater การใช้ Optoisolator นี้ มีจุดประสงค์เพื่อที่จะแยกกราวด์ โดยสามารถที่จะป้องกันความเสียหายและสัญญาณรบกวน อันเกิดจากเครื่องน้ำอุ่น ซึ่งจะทำให้การควบคุมผิดพลาดได้ Optoiolator MOC 3043 ใช้ควบคุมการ ON/OFF ของ Triac ในเครื่องทำน้ำอุ่นอีกทีหนึ่ง โดยใน I.C. นี้จะมีวงจร ZERO CROOSING ทำหน้าที่ตรวจจรือ detect ช่วง Crossing Zero ของไฟกระแสสลับที่เป็นไฟเลี้ยงของเครื่องทำน้ำอุ่นนี้ รูปที่ 3.1.4 แสดง block diagram ภายในของ MOC 3043

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1.2 แสดงวงจร LCD ส่วนติดต่อกับผู้ใช้

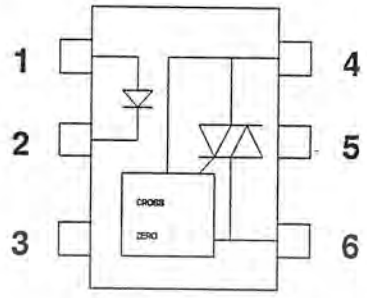
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1.3 แสดงวงจรส่วนวัดอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

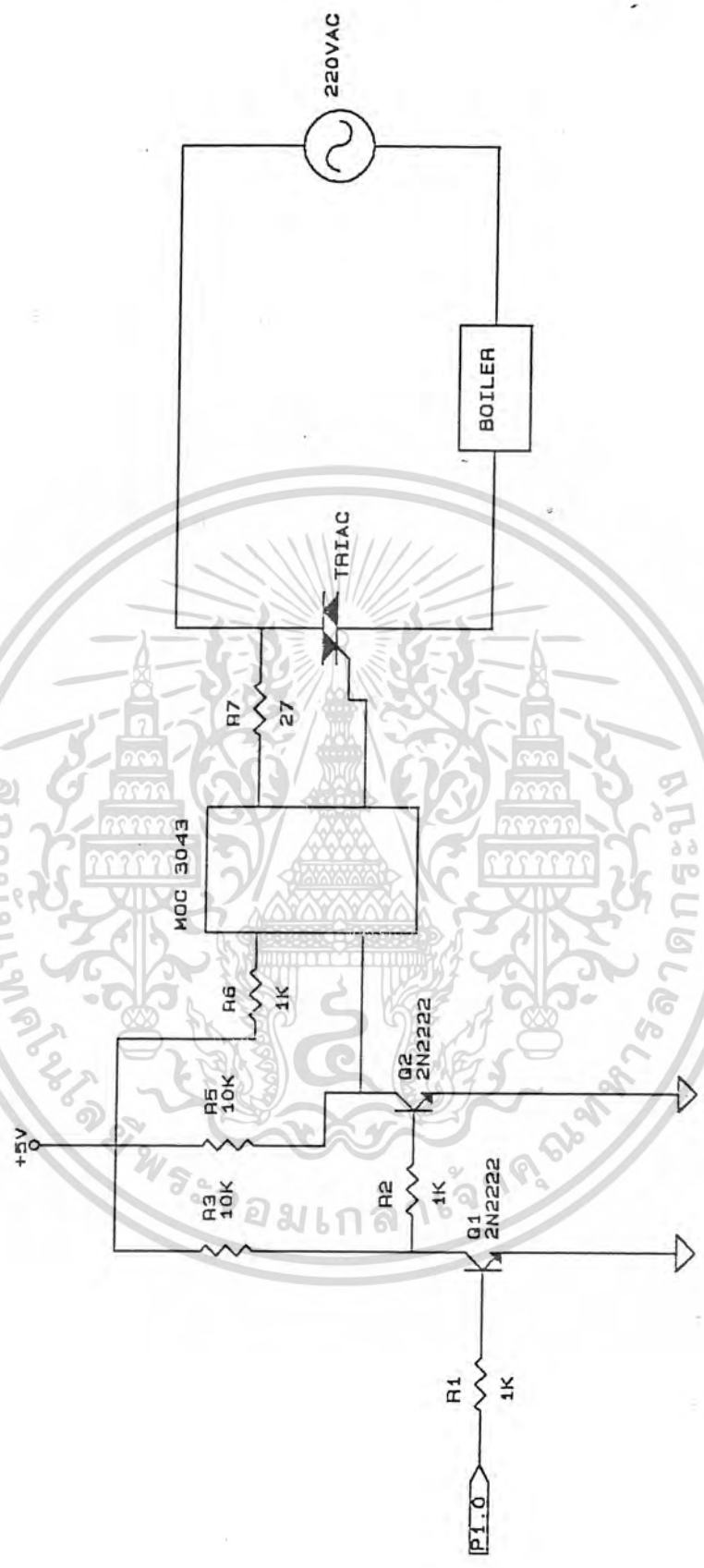
<u>รูปที่ 3.1.4</u> แสดง Block Diagram ภายในของ MOC 3043



สัญญาณที่ได้จาก Zero Crossing Circuit นี้จะนำไปเป็นสัญญาณ Trig ให้แก่ Triac ภายใน MOC สัญญาณทริกที่ได้นี้ ได้มาจากการทำงานร่วมกับสัญญาณซึ่งมาจากการทำงานร่วมกับสัญญาณซึ่งออกมาจากพอร์ตๆหนึ่งของ Microcontroller MCS - 31 โดยที่ถ้าเราต้องการให้ Heater ทำงานเราจะส่งระดับแรงดัน 0 volt ออกไป และถ้าเราต้องการให้ Heater หยุดทำงานเราจะส่งระดับแรงดัน 5 volt ออกไปแทน รูปที่ 3.1.5 แสดงวงจรส่วนควบคุมการทำงานของ Heater

ลักษณะการควบคุมการ ON/OFF ของ Heater จะใช้เทคนิค MARK AND SPACE คือ เป็นการควบคุมโดยเป็นสัดส่วนของเวลา (Time - Proportional Control) ซึ่งจะให้ผลการควบคุมที่ดีกว่าโดยอาศัยวิธี Phase-Angle-Firing คือ วิธี MARK AND SPACE จะสามารถลด Noise ซึ่งเกิดจากการ Trig ของวิธี Phase-Angle-Fring ที่ Peak ของ Line Voltage ได้อย่างมาก ทำให้การทำงานของระบบไม่ถูกรบกวน และข้อสำคัญคือ ช่วยยืดอายุการทำงาน Heater และ ตัว Triac เองได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1.5 แสดงวงจรส่วนควบคุม HEATER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2. ขั้นตอนการออกแบบ Software

เรานำทฤษฎี Fuzzy Logic ของศาสตราจารย์ Lotfi Zadeh มาประยุกต์ใช้โดยทำได้ดังนี้

3.2.1 Fuzzification of Inputs

เราจะนำ Input Temperature ที่ได้มาหาค่า Degree of Membership โดยเราจะแปลง Input Termperature เป็น Input ของ Fuzzy Logic System ได้ 2 ค่าคือ

1.1) CHANGE ERROR TEMPERATURE (ΔE) = E(t) - E(t-1)

1.2) ERROR TEMPERATURE E(t) = SET TEMPERATURE - T(t)

โดยที่

SET TEMPERATURE	~	อุณหภูมิที่ตั้งไว้ ณ. เครื่องทำน้ำอุ่น
T(t)	~	อุณหภูมิที่วัดได้ในขณะนั้น
E(t)	~	ค่า Error ปัจจุบัน
E(t-1)	~	ค่า Error ก่อนหน้านั้น
ΔE	~	ค่า Change Error Temperature

โดยจากตัวแปรทางอินพุททั้งสองตัวนี้ E(t) และ E(t-1) จะต้องนำมาผ่านขั้นตอนของการ FUZZIFICATION ให้อยู่ในรูปของ FUZZY INPUT ได้โดยการเทียบค่าตัวแปรทางอินพุทกับตารางที่ 3.2.1 ซึ่งเป็น QUANTIZED TABLE ที่กำหนดขึ้นมา เมื่อได้ค่า FUZZY INPUT แล้วจากตารางที่ 3.2.1 จึงนำมาเปรียบเทียบกันกับกราฟความเป็นสมาชิก(รูปกราฟที่ 3.2.1 และ 3.2.2) ซึ่งเกิดจากการกำหนดขึ้นมาก่อน แล้วค่อยๆทำการปรับกราฟจนได้กราฟที่มีความสอดคล้องกับความเป็นจริงมากที่สุด โดยที่ ERROR และ CHANGE ERROR จะใช้กราฟความเป็นสมาชิกคนละกราฟกัน ซึ่งในแต่ละกราฟจะมีจำนวนเทอมหรือ LABEL เท่ากับ 5 LABEL ดังแสดงในกราฟที่ 3.2.1 และ 3.2.2

โดยที่

LP = LARGE POSITIVE

SP = SMALL POSITIVE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Z = ZERO

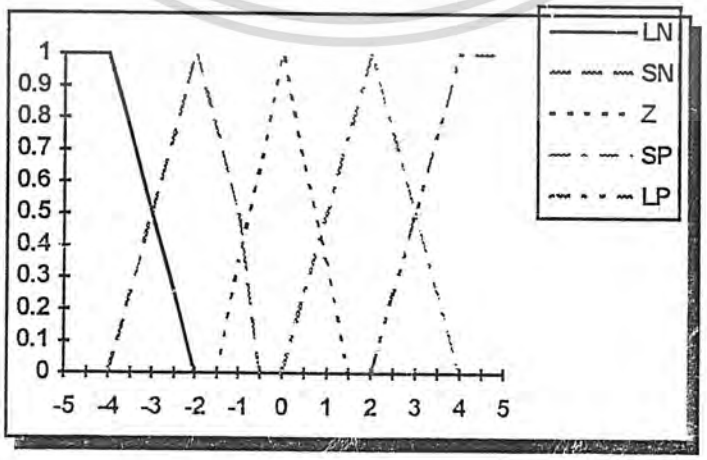
SN = SMALL NEGATIVE

LN = LARGE NEGATIVE

ตารางที่ 3.2.1 QUANTIZED TABLE

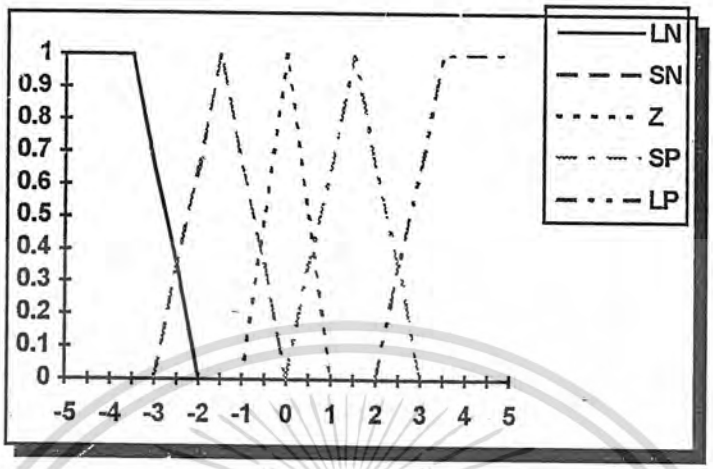
ERROR TEMPERATURE	CHANGE ERROR TEMPERATURE	QUANTIZED VALUE
>= 2.5	>= 1.0	+ 5
2.0 ~ 2.4	0.8 ~ 0.9	+ 4
1.5 ~ 1.9	0.6 ~ 0.7	+ 3
1.0 ~'1.4	0.4 ~ 0.5	+ 2
0.5 ~ 0.9	0.2 ~ 0.3	+ 1
- 0.4 ~ 0.4	- 0.1 ~ 0.1	0
- 0.9 ~ - 0.5	- 0.3 ~ - 0.2	- 1
- 1.4 ~ - 1.0	- 0.5 ~ - 0.4	- 2
- 1.9 ~ - 1.5	- 0.7 ~ - 0.6	- 3
- 2.4 ~ -2.0	- 0.9 ~ - 0.8	- 4
<= -2.5	<= -1	- 5

รูปกราฟที่ 3.2.1 กราฟความเป็นสมาชิก (MEMBERSHIP GRAPH)ของ ERROR TEMP.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปกราฟที่ 3.2.2 กราฟความเป็นสมาชิกของ CHANGE ERROR TEMP.



3.2.2 RULE EVALUATION

การคำนวณค่าตามกฎการที่ตั้งไว้ โดยอาศัยค่าความเป็นสมาชิกของตัวแปรที่ได้จากการเปรียบเทียบค่า FUZZY INPUT กับกราฟความเป็นสมาชิก(กราฟที่ 3.2.1) เป็นตัวกำหนดความสำคัญของกฎแต่ละกฎ โดยอาศัยเทคนิคของค่ากฎ SMALLEST ระหว่างตัวแปร FUZZY INPUT ทั้ง 2 ตัวในส่วนเงื่อนไขคือ ERROR และ CHANGE ERROR ตาม FUZZY IF/THEN RULES ดังตารางที่ 3.2.2

ตารางที่ 3.2.2 แสดง FUZZY IF/THEN RULES TABLE

IF / THEN RULES	ERROR TEMPERATURE	CHANGE ERROR TEMPERATURE	BOILER STATUS
1	LP	ANY VALUE	H
2	SP	LP	C
3	SP	SP	E
4	SP	Z	F
5	SP	SN	E
6	SP	LN	H
7	Z	LP	A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8	Z	SP	B
9	Z	Z	B
10	Z	SN	A
11	Z	LN	D
12	SN	LP	C
13	SN	SP	B
14	SN	Z	C
15	SN	SN	A
16	SN	LN	B
17	LN	ANY VALUE	A

โดยที่

LP = LARGE POSITIVE

SP = SMALL POSITIVE

Z = ZERO

SN = SMALL NEGATIVE

LN = LARGE NEGATIVE

จากตารางที่ 3.2.2 นั้น ค่า STATUS เป็นผลลัพธ์ (Consequence) ที่เกิดจากแต่ละ LABEL ในส่วนเงื่อนไขของแต่ละตัว มีค่าตามตารางที่ 3.2.3

ตารางที่ 3.2.3 ตารางแสดงค่า STATUS

STATUS (Temperature)	ASSIGNED VALUE
A (LOWEST)	1
B (LOW)	2
C (INTERMEDIATE LOW)	3
D (INTERMEDIATE)	4
E (HIGH)	5
F (INTERMEDIATE HIGH)	6
G (VERY HIGH)	7
H (HIGHEST)	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 Defuzzification of Outputs

เป็นขั้นตอนที่นำเอาผลลัพธ์ที่ได้จากขั้นตอน RULES EVALUATION เพื่อหาค่าของ FUZZY OUTPUT ซึ่งเมื่อทำการเปรียบเทียบตามกฎต่างๆแล้ว จะเห็นได้ว่าถ้ามีการคำนวณตามกฎการควบคุม แล้วนำเพียงกฎใดกฎหนึ่งมาใช้จะทำให้ผลการควบคุมระบบผิดพลาดได้ จึงได้นำค่าเอาท์พุทของกฎต่างๆ มาหาค่าเอาท์พุทที่จะใช้ในการควบคุมระบบ โดยใช้การคำนวณแบบ CENTER OF GRAVITY คำนวณผลลัพธ์ออกมา โดยค่าของจุดศูนย์ถ่วงสำหรับตัวแปรเอาท์พุท HEATER ได้แสดงดังตารางที่ 3.2.3

หลังจากการที่ได้ทำการคำนวณตาม RULE EVALUATION และทำการหาค่า CENTER OF GRAVITY แล้วสามารถเขียนตารางของ FUZZY OUTPUT สำหรับค่า ERROR และ CHANGE ERROR ต่างๆได้ดังตารางที่ 3.2.4

ตารางที่ 3.2.4 ตารางที่แสดงค่า OUTPUTS จากการคำนวณโดย FUZZY RULES

Ch.Err. / Error	-5	-4	-3	-2	-1	0	1	2	3	4	5
-5	1	1	1	1	1	1	1	1	1	1	1
-4	1	1	1	1	1	1	1	1	1	1	1
-3	1.5	1.5	1.5	1	1	2	1.5	1.5	2	2	2
-2	2	2	2	1	1	3	2	2	3	3	3
-1	2.82	2.82	2.82	1	1	2.58	2	2	2.35	2.35	2.35
0	4	4	4	1	1	2	2	2	1	1	1
1	6.35	6.35	6.35	3.35	3.35	4.35	3.76	3.76	2.17	2.17	2.17
2	8	8	8	5	5	6	5	5	3	3	3
3	8	8	8	6.5	6.5	7	6.5	6.5	5.5	5.5	5.5
4	8	8	8	8	8	8	8	8	8	8	8
5	8	8	8	8	8	8	8	8	8	8	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่ได้ทำ DEFUZZIFICATION แล้วก็จะได้ผลลัพธ์ของการคำนวณในรูปของ FUZZY OUTPUT ซึ่งค่าของ FUZZY OUTPUT นี้ยังไม่สามารถนำไปใช้ในการควบคุมเครื่องทำน้ำอุ่นได้โดยตรงจึงต้องนำมาผ่านขบวนการที่จะทำการเปลี่ยนให้ค่าของ FUZZY OUTPUT นั้นสามารถนำไปใช้ในการควบคุมเครื่องทำน้ำอุ่นได้ซึ่งวิธีการก็คือจะนำค่า FUZZY OUTPUT ไปเปรียบเทียบกับตารางเอาท์พุท(ตารางที่ 3.2.5) ซึ่งเป็นขบวนการกลับกันกับการหาค่าของ FUZZY INPUT เพื่อหาสถานะการทำงานของเครื่องทำน้ำอุ่น เมื่อได้ทำการเปลี่ยนค่า FUZZY OUTPUT ให้เป็นสถานะการทำงานของ HEATER เรียบร้อยแล้วก็จะได้สถานะการทำงานของ HEATER ที่ ERROR TEMPERATURE และ CHANGE ERROR TEMPERATURE ต่างๆ ดังแสดงในตารางที่ 3.2.6

ตารางที่ 3.2.5 แสดงค่าความสัมพันธ์ ของ FUZZY OUTPUT กับ HEATER STATUS

FUZZY OUTPUT	HEATER STATUS
0.0 - 1.4	A
1.5 - 2.4	B
2.5 - 3.4	C
3.5 - 4.4	D
4.5 - 5.4	E
5.5 - 6.4	F
6.5 - 7.4	G
7.5 - 8.0	H

ตารางที่ 3.2.6 ตารางแสดง FUZZY OUTPUTS ที่นำไปใช้งานจริง

Ch.Err. / Error	-5	-4	-3	-2	-1	0	1	2	3	4	5
-5	A	A	A	A	A	A	A	A	A	A	A
-4	A	A	A	A	A	A	A	A	A	A	A
-3	B	B	B	A	A	B	B	B	B	B	B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-2	B	B	B	A	A	C	B	B	C	C	C
-1	C	C	C	A	A	C	B	B	B	B	B
0	D	D	D	A	A	B	B	B	A	A	A
1	F	F	F	C	C	D	D	D	B	B	B
2	H	H	H	E	E	F	E	E	C	C	C
3	H	H	H	G	G	G	G	G	F	F	F
4	H	H	H	H	H	H	H	H	H	H	H
5	H	H	H	H	H	H	H	H	H	H	H

จากขั้นตอนต่างๆที่กล่าวมาเราจะทำการทดลองหาค่า Fuzzy Output 1 ค่า โดยในตอนแรกเราจะสมมุติค่าต่างๆดังต่อไปนี้

SET TEMPERATURE = 31.9 องศา

E(t) = 30.0 องศา

E(t-1) = 29.5 องศา

ดังนั้นนำค่าเหล่านี้เข้าสู่ขั้นตอน FUZZIFICATION METHOD แปลงเป็น Fuzzy Input จะได้ค่า

CHANGE ERROR TEMPERATURE = 0.5 องศา

ERROR TEMPERATURE = 1.9 องศา

จากค่าที่ได้นำไปเทียบกับตารางที่ 3.2.1 จะได้

QUANTIZED CHANGE ERROR TEMP. = +2

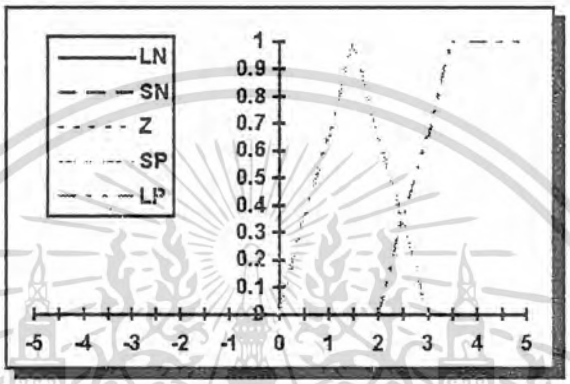
QUANTIZED ERROR TEMP. = +3

เมื่อนำค่า Quantized Change Error Temp. = +2 ไปหาค่า Membership จากกราฟรูป 3.2.2 จะพบว่าค่า +2 ไปตัดกราฟ Small Positive ที่ค่า Membership = 0.65 ดังแสดงในกราฟรูป 3.2.3 และในทำนองเดียวกันเมื่อนำค่า Quantized Error Temp. = +3 ไปหาค่า

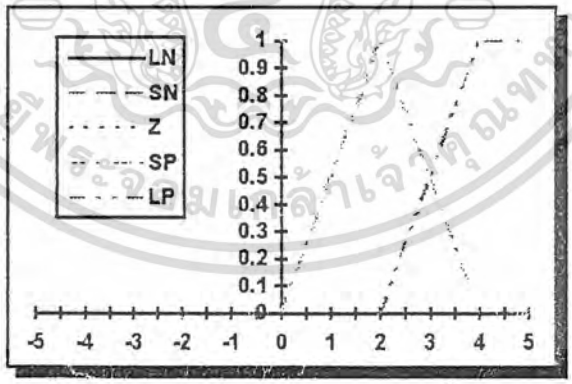
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

membership = 0.5 และไปตัดกราฟ large positive ที่ค่า membership = 0.5 เช่นเดียวกันดังแสดงในกราฟรูป 3.2.4

รูปกราฟที่ 3.2.3 กราฟความเป็นสมาชิกของ CHANGE ERROR TEMP.



รูปกราฟที่ 3.2.4 กราฟความเป็นสมาชิก (MEMBERSHIP GRAPH) ของ ERROR TEMP.



จากนั้นนำค่า membership ที่ได้เหล่านั้นเข้าสู่ขั้นตอน RULES EVALUATION โดยนำไปเปรียบเทียบหากฎที่จะตกกฎจากตารางที่ 3.2.2 จะเห็นว่าจะตกกฎที่ 1 ซึ่งให้ค่าเอาท์พุทเป็น H(=8) และตกกฎที่ 3 ซึ่งให้ค่าเอาท์พุทเป็น E(=5) และใช้เทคนิค SMALLEST ในแต่ละกฎที่ได้จะทำให้ได้ค่า degree of membership ของกฎที่ 1 เป็น 0.5 และของกฎที่ 2 เป็น 0.5 เช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าที่ได้จากขั้นตอน RULES EVALUATION นำเข้าสู่ขั้นตอน DEFUZZIFICATION OF OUTPUTS โดยใช้การคำนวณแบบ CENTER OF GRAVITY เพื่อคำนวณหาค่าเฉลี่ยของเอาท์พุทออกมา โดยมีสูตรการคำนวณว่า

$$I = \frac{\sum_{i=1}^{n} (u_n \times Un)}{\sum_{i=1}^{n} u_n}$$

โดยที่

I = RESULT OF CENTROID METHOD

u_n = DEGREE OF MEMBERSHIP

Un = STATUS FROM IF/THEN RULES

ดังนั้นค่า I จะเท่ากับ

$$I = \frac{(0.5 \times 8) + (0.5 \times 5)}{(0.5 + 0.5)} = 6.5$$

ซึ่งเป็นค่าที่ตรงกับในตารางที่ 3.2.4 และเมื่อนำค่าที่คำนวณได้ไปเทียบกับตารางที่ 3.2.5 จะได้ค่าของ FUZZY OUTPUT คือ F ซึ่งตรงกับค่าในตารางที่ 3.2.6 ที่ ERROR TEMPERATURE = +3 และ CHANGE ERROR TEMPERATURE = +2 และในการหาค่า FUZZY OUTPUT ที่อุณหภูมิอื่นๆก็ทำได้ในทำนองเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ตัวจับเวลา/ตัวนับ (Timer/Counter)

MCS-51 มี ตัวจับเวลา 16 บิต ตัวนับ 2 ตัว คือ Timer/Counter 0 และ Timer/Counter 1 ขณะที่แต่ละตัวจับเวลา/ตัวนับสามารถที่จะติดตั้งให้ทำงานได้เป็นตัวจับเวลาหรือตัวนับก็ได้

ตัวจับเวลา/ตัวนับ 0 และตัวจับเวลา/ตัวนับ 1 แต่ละตัวจะถูกติดตั้งให้ทำงานเป็นตัวจับเวลาหรือเป็นตัวนับ ได้ด้วยการเซ็ตหรือเคลียร์บิตที่ตัวควบคุมในรีจิสเตอร์ TMOD ในกลุ่ม SFR

ในฟังค์ชั่นตัวจับเวลา ตัวรีจิสเตอร์จะเพิ่มค่าทุกๆ วัฏจักรแมชชีน ดังนั้น ตัวเลขในรีจิสเตอร์จะเป็นจำนวนของวัฏจักรแมชชีน เนื่องจากแต่ละวัฏจักรแมชชีน ประกอบด้วย 12 คาบของออสซิลเลเตอร์ อัตราการนับแต่ละครั้ง จะกินเวลาเป็น 1/12 ของความถี่ออสซิลเลเตอร์

ในฟังค์ชั่นตัวนับรีจิสเตอร์จะเพิ่มค่าทุกครั้งที่มีการเปลี่ยนแปลงสถานะ จาก 1 เป็น 0 ที่เข้ามาที่ขา T0 หรือ T1 ในฟังค์ชั่นนี้สัญญาณที่เข้ามาจะถูก Sampling ทุกช่วง S5P2 ของทุกวัฏจักรแมชชีน โดยถ้า Sampling สัญญาณเข้าเป็นระดับสูงในวัฏจักรหนึ่ง ดังนั้นถ้าในวัฏจักรตัวต่อมาของสัญญาณเข้ามาเป็นระดับต่ำ รีจิสเตอร์จะนับเพิ่มหนึ่งค่า โดยที่ค่าใหม่ของตัวนับ จะปรากฏที่รีจิสเตอร์ช่วง S3P1 ของวัฏจักร ซึ่งค่าหนึ่งที่รับเข้าไป จะใช้ช่วง 2 วัฏจักรแมชชีน (เท่ากับ 24 คาบ) ในการรับค่าช่วงการเปลี่ยน 1 เป็น 0 ดังนั้น ค่าสูงสุดในการนับจะมีอัตรา 1/24 ของความถี่ออสซิลเลเตอร์ และสัญญาณอินพุทที่นับนั้นจะไม่มีช่วงระยะห่างที่แน่นอนของ Duty Cycle แต่จะถูกนับ เมื่อระดับแรงดันที่ถูกแซมปลิ้งในแต่ละครั้งจะต้องมีช่วงคงที่อย่างน้อย 1 วัฏจักรแมชชีน ก่อนที่จะเปลี่ยนค่าระดับใหม่ ในการเลือกการทำงานระหว่างตัวนับกับตัวจับเวลา จะเลือกได้ 4 โหมด คือ โหมด 0,1 และ 2 เลือกได้ทั้งสองตัวของ Timer/Counter ส่วนโหมด 3 จะทำงานแตกต่างออกไป

3.3.1 โหมด 0

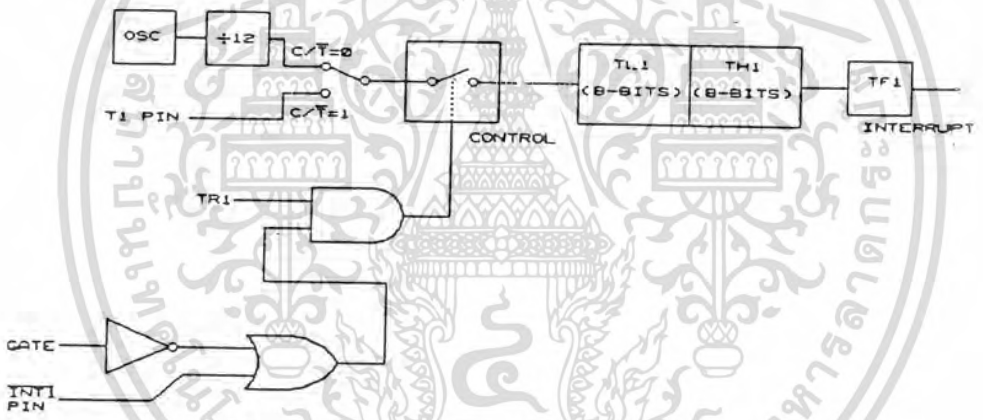
การใช้ตัวจับเวลา/ตัวนับ 0 หรือ 1 ให้อยู่ในโหมด 0 จะทำงานคล้ายกับของ MCS-48 โดยตัวจับเวลาของ MCS-48 มีขนาด 8 บิต มีตัว Prescaler เป็นตัวหาร 12 รูปที่ 3.3.1 แสดงการทำงานในโหมด 0 ของตัวจับเวลา/ตัวนับ 1 ในโหมดนี้ รีจิสเตอร์ตัวจับเวลาถูกกำหนดให้มี 13 บิต ด้วยการนับขึ้น เมื่อเป็น 1 หมดทุกบิต จะกลับมาที่ 0 ทุกบิตใหม่ เมื่อกลับเป็น 0 ทุกบิต จะเป็นการเกิด overflow ไปทดให้แฟลกอินเตอร์รัพต์ TF1 ปรับเป็น 1 การควบคุมให้เริ่มนับตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุทจะควบคุมด้วยการอีนาเบิ้ล TR1 = 1 GATE = 0 INT1 = 1 การปรับ GATE = 1 เป็นการติดตั้งตัวนับให้นับด้วยสัญญาณภายนอกที่เข้ามาที่ขา INT1 TR1 จะเป็นการควบคุมในรีจิสเตอร์ TMOD ของ SFR

รีจิสเตอร์ตัวนับจะมี 13 บิต ประกอบด้วย TH1 8 บิต และ TL1 5 บิต อันดับต่ำส่วนอีก 3 บิตที่เหลือในอันดับสูงของ TL1 จะไม่ใช้การเซตแฟลก TR1 ให้ทำงาน จะไม่ได้เคลียร์ค่าในรีจิสเตอร์ของ TH1 และ TL1 การทำงานในโหมด 0 ในตัวจับเวลา/ตัวนับ 0 จะทำงานเหมือนกับตัวจับเวลา/ตัวนับ 1 โดยใช้ TR0 และ INT0 รวมกับสัญญาณควบคุมต่างๆ ในรูป 3.3.1 มีความแตกต่างในการควบคุม คือ บิตของ GATE ทั้งสอง ตัวหนึ่งจะแทนตัวจับเวลา/ตัวนับ 1 (TMOD.7) และอีกตัวจะแทน ตัวจับเวลา/ตัวนับ0 (TMOD.3)

รูปที่ 3.3.1 แสดงการทำงานในโหมด 0 ของตัวจับเวลา/ตัวนับ 1 ขนาด 13 บิต



3.3.2 โหมด 1

โหมด 1 ทำงานเหมือนโหมด 0 ต่างกันแต่เฉพาะการใช้รีจิสเตอร์ ตัวจับเวลา/ตัวนับจะทำงานนับด้วย ขนาด 16 บิต โดยไม่มี Prescaler คือความถี่ 1/12 ของออสซิลเลเตอร์ เป็นความถี่ที่เข้ามาถูกหารด้วย ค่า 16 บิต ในรีจิสเตอร์ตัวนับ

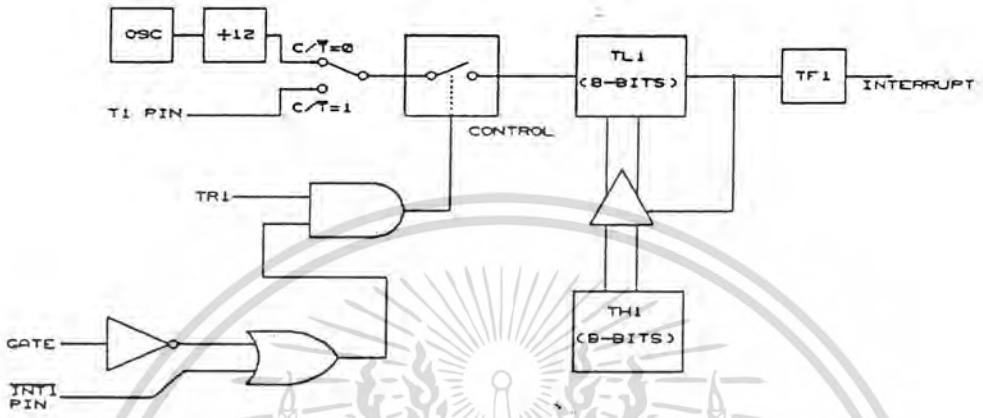
3.3.3 โหมด 2

มีการทำงานโดยการกำหนดให้ตัวนับขนาด 8 บิต ของ TL1 และจะโหลดใหม่โดยอัตโนมัติทุกครั้ง เมื่อมี การ Overflow จาก TL1 ดังรูปที่ 3.3.2 ไม่เพียงแต่ TF1 จะปรับเป็น 1 แต่ TL1 จะถูกโหลดโดยอัตโนมัติจากค่าที่ตั้งไว้ใน TH1 ซึ่งค่าที่ตั้งไว้ใน TH1 ซึ่งค่าใน TH1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถตั้งค่าได้ด้วยซอฟแวร์ คือการใช้คำ สั่ง MOV และบรรจุเข้าใหม่ที่ TL1 ทุกครั้งที่เกิด Overflow TH0 และ TF0 จะเป็นตัวร่วมในการทำงานใน โหมดนี้

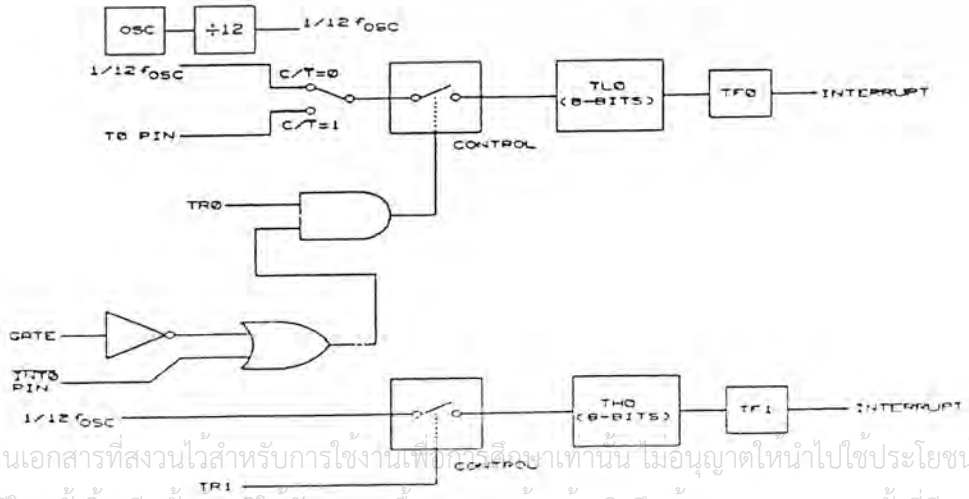
รูปที่ 3.3.2 ตัวจับเวลา/ตัวนับ 1 ทำงานในโหมด 2 แบบโหลดใหม่ขนาด 8 บิต



3.3.4 โหมด 3

ถ้าใช้ตัวจับเวลา/ตัวนับ ในโหมด 3 มีการทำงานเป็นตัวนับ มีผลเช่นเดียวกับการตั้ง TR1 = 0 และใช้ตัว จับเวลา/ตัวนับ ในโหมด 3 จัดการให้ TL0 และ TH0 เป็นตัวนับสองตัวแรกที่แยกออกจากกัน วงจรควบ คุมสำหรับโหมด 3 ที่ใช้ตัวจับเวลา/ตัวนับเวลา 0 แสดงในรูป 3.3.3 TL0 ใช้ตัวจับเวลา/ตัวนับ 0 ร่วม กับบิตควบคุม C/T, GATE, TR0 INT0 และ TF0 ตัว TH0 จะถูกล็อกการทำงานในฟังค์ชั่นตัวจับเวลา และใช้บิตแฟลก TR1 และ TF1เข้าร่วมการทำงานในโหมด 3 ดังนั้นตัว TH0 ในโหมดนี้จะควบคุมการอินเตอร์รัพต์ของตัวจับเวลา 1 เป็นกลุ่มจับเวลาและขนาด 8 บิตสองตัว

รูปที่ 3.3.3 ใช้ตัวจับเวลา/ตัวนับ 1 ในโหมด 3 เป็นตัวกลุ่มนับขนาด 8 บิต 2 ตัว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3.1 TMOD : Timer/Counter Mode Control Register

GATE	C/T	M1	M0	GATE	C/T	M1	MO

GATE : ควบคุมเกต เมื่อเซ็ตเป็น 1 จะเป็นอีนาเบิ้ล ตัวจับเวลา/ตัวนับ ขณะที่ขา INTx มีสถานะสูง และขาควบคุม TRx ใน TCON จะถูกเซตเป็น 1 เมื่อตัวนับภายในถูก เคลียร์ให้อีนาเบิ้ล เมื่อไรก็ตามที่บิตควบคุม TRX ถูกเซตเป็น 1

C/T : เลือกการทำงานแบบตัวจับเวลาหรือตัวนับ ถ้าเป็น 0 จะเลือกการทำงานเป็นตัวจับ เวลา ถ้าเป็น 1 จะเป็นการทำงานแบบตัวนับ และรับสัญญาณเข้าที่ขา TX

M1,M0 การทำงาน

0 0 : ทำงานแบบตัวจับเวลาของ MCS-48 ใช้ TLx เป็นตัวป้อนบิตอีก 5 บิต

0 1 : การใช้ตัวจับเวลา/ตัวนับ ขนาด16 บิตจะใช้ THx และ TLx เป็นตัวนับไม่มี Prescaler

1 0 : การโหลดขนาด 8 บิตโดยอัตโนมัติที่ตัวนับและตัวจับเวลา โดยใช้ THx เก็บค่าที่ตั้งไว้และจะถ่ายเข้าไปที่ TLx ใหม่ทุกครั้งที่เกิด Overflow คือ TLx ถูกนับเป็น 0 หมด

ตารางที่ 3.3.2 TCON : TIMER/COUNTER CONTROL REGISTER

IF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1 : TCON.7 ; ตัวจับเวลา 1 แฟลกเป็น 1 เมื่อเกิด Overflow ถูกเซ็ตเป็น 1 ด้วยฮาร์ดแวร์ทางสัญญาณ เมื่อตัวจับเวลา/ตัวนับ Overflow และจะเคลียร์ตัวเองเมื่ออินเตอร์รัพต์ไปแล้ว

TR1 : TCON.6 ; ตัวจับเวลา 1 เป็นตัวควบคุมบิตให้เริ่มการทำงานจะเซตหรือเคลียร์ด้วยซอฟแวร์ที่จะมาทำให้ ตัวจับเวลา/ตัวนับ 1 เริ่มหรือหยุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TF0 : TCON.5 ; ตัวจับเวลา 0 แฟลกเป็น 1 เมื่อเกิด Overflow ถูกเซตเป็นหนึ่งด้วยฮาร์ดแวร์ทางสัญญาณ เมื่อตัวจับเวลา/ตัวนับ overflow เคลียร์ตัวเองเมื่อเข้าอินเทอร์รัพต์ไปแล้ว

TR0 : TCON.4 ; ตัวจับเวลา 0 เป็นตัวควบคุมบิตให้เริ่มทำงาน จะเซตหรือเคลียร์ด้วยซอฟแวร์ที่จะมาทำให้ตัวจับเวลา/ตัวนับ เริ่มหรือหยุดการทำงาน

IE1 : TCON.3 ; เป็นแฟลกของขอบสัญญาณอินเทอร์รัพต์ 1 จะเซตด้วยฮาร์ดแวร์เมื่อสัญญาณขอการอินเทอร์รัพต์ปรากฏเข้าที่ขา INT1 และเคลียร์เมื่อการทำงานอินเทอร์รัพต์สิ้นสุดลง

IT1 : TCON.2 ; รูปแบบการควบคุมบิตของการอินเตอร์รัพต์ 1 จะเซตหรือเคลียร์ได้ด้วยซอฟแวร์ที่จะเป็นตัวกำหนดให้มีการกระตุ้นอินเตอร์รัพต์จากภายนอกที่เป็นขอบขาลงโดยถ้า IT1 = 1 จะควบคุมอินเตอร์รัพต์แบบขอบขาลง และถ้า IT = 0 จะควบคุมอินเทอร์รัพต์แบบระดับแรงดันต่ำ

IE0 : TCON.1 ; เป็นแฟลกขอบสัญญาณอินเตอร์รัพต์ 0 เซตด้วยฮาร์ดแวร์เมื่อสัญญาณขอบอินเทอร์รัพต์ปรากฏเข้าที่ขา INT0 และเคลียร์เมื่อการทำงานอินเตอร์รัพต์สิ้นสุด

IT0 : TCON.0 ; รูปแบบการควบคุมบิต ของการอินเตอร์รัพต์ จะเซตหรือเคลียร์ด้วยซอฟแวร์ที่จะเป็นตัวกำหนดให้มีการกระตุ้นอินเตอร์รัพต์จากภายนอกที่แบบขอบขาลงหรือเป็นแบบระดับต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การอินเตอร์รัพต์

3.4.1 ความสามารถในการอินเตอร์รัพต์

โดยทั่วไปความสามารถในการอินเตอร์รัพต์ เป็นการทำงานชนิดหนึ่งของซีพียู ที่จำเป็นจะต้องศึกษาถึงความสามารถและเทคนิคการทำงาน การอินเตอร์รัพต์ของซิงเกิ้ลชิปมีความสัมพันธ์กันอย่างใกล้ชิดระหว่างอุปกรณ์ต่อพ่วงกับระบบ การทำงานของอุปกรณ์ต่อพ่วงเหล่านี้มีระบบฮาร์ดแวร์ ที่ช่วยให้การส่งสัญญาณ Real Time กับมอนิเตอร์ได้อย่างต่อเนื่อง โดยปราศจากการรบกวนต่อสัญญาณการทำงานของซีพียู ตัวอย่างเช่น ขณะที่มีการรับสัญญาณอนุกรมจากซีอาร์ทีตัวหนึ่ง ก็จะมีการส่งสัญญาณไปยังอุปกรณ์ตัวอื่น และตัวจับเวลา/ตัวนับ ก็จะนับพัลส์การเปลี่ยนแปลงที่เข้ามาอย่างรวดเร็วไปพร้อมกันด้วย ในขณะที่ตัวจับเวลา/ตัวนับอีกตัวหนึ่งก็กำลังวัดความกว้างของพัลส์ที่เข้ามา

ซีพียูตัวนี้รู้ได้อย่างไรว่าเมื่อไรถึงจะมีการรับและส่งสัญญาณอนุกรมซีอาร์ทีหรือให้ตัวจับเวลา/ตัวนับ มีการนับจำนวนและวัดความกว้างของพัลซ์ว่ามีการสิ้นสุดลงเมื่อไร ตัวโปรแกรม MCS-51 สามารถที่จะะเลือกการโปรแกรมได้ 3 วิธีด้วยกันคือพิจารณาการโปรแกรมตัวรีจิสเตอร์ TCON และ SCON ที่ประกอบด้วยสถานะบิตที่ถูกเซตทางฮาร์ดแวร์ เมื่อตัวจับเวลาตัวหนึ่งเกิด Overflow หรือเมื่อการรับส่งข้อมูลที่พอร์ตอนุกรมสิ้นสุดลง

เทคนิคการโปรแกรมวิธีแรก คือ โดยการอ่านสถานะ ของรีจิสเตอร์ควบคุม เข้าไปยังแอกคูมิวเลเตอร์ แล้วทดสอบสถานะบิตตามลักษณะการทำงานนั้นๆ แล้วทำการกระโดดไปยังโปรแกรมย่อย ตามผลที่เกิดขึ้นนั้นๆลักษณะการทดสอบร่วมกันครั้งละหลายลักษณะงานเช่นนี้เปรียบเสมือนตัวโปรแกรมใช้ระบบไมโครโปรเซสเซอร์หลายตัวควบคุมชิปอุปกรณ์ต่อพ่วงต่างๆ ซึ่งผู้โปรแกรมต้องทำความเข้าใจลึกซึ้งกับระบบ และจังหวะที่เกิดขึ้นในแต่ละงาน

วิธีที่สอง MCS-51 สามารถที่จะทำงานด้วยการกระโดดไปตามสถานะของการควบคุมหรือสถานะบิตของรีจิสเตอร์ควบคุมงานหรือการรับสัญญาณที่เข้ามาตามขาอินพุตแต่ละบิตด้วยการใช้คำสั่งเพียงคำสั่งเดียว ดังนั้นงาน 4 อย่างก็สามารถใช้คำสั่งตรวจสอบได้ภายใน 4 คำสั่งซึ่งจะใช้เวลาประมาณ 8 ไมโครวินาที

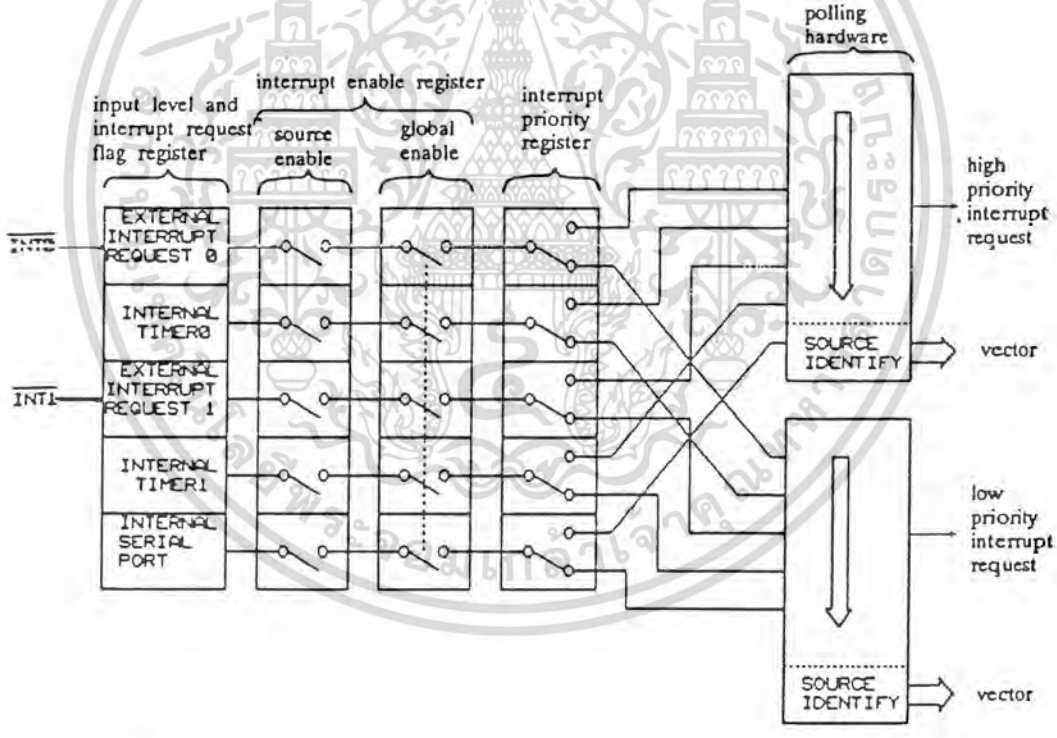
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่วิธีทั้งสองที่กล่าวมา ซีพียูต้องทำการตรวจสอบบิตสถานะต่างๆอยู่ตลอดเวลาซึ่งทำให้เสียเวลา

วิธีที่สาม ซีพียูจะทำหน้าที่หลักของตัวเอง และใช้เวลาเพียงบางส่วนเพื่อจัดการกับอุปกรณ์ต่อพ่วงที่ขออินเตอร์รัพต์เข้ามา เมื่อสิ้นสุดงานบริการนั้นแล้วก็จะกลับมาทำงานของตัวเอง วิธีนี้จะทำให้ซีพียูมีเวลามากขึ้น พร้อมที่จะทำงานอื่นได้ตลอดเวลา ทำให้รู้สึกว่าตัวไมโครโพรสเซอร์ทำงานพร้อมกันได้หลายงานในเวลาเดียวกัน การใช้วิธีที่สามจะเป็นวิธีที่ดีที่สุดในการใช้งานในลักษณะนี้ด้วยการเกิดอินเตอร์รัพต์จากทางฮาร์ดแวร์

แผนภูมิทางฮาร์ดแวร์ของการเกิดอินเตอร์รัพต์ชนิดต่างๆ ของ MCS-51 ตามรูปที่ 3.4.1 จะมีรายละเอียดดังนี้

รูปที่ 3.4.1 แสดง hardware interrupt



แบบการอินเตอร์รัพต์จากภายนอก INT0 และ INT1 สามารถจะใช้สัญญาณทั้งแบบขอบขาลงหรือระดับแรงดันต่ำขึ้นอยู่กับการติดตั้งบิต IT0 และ IT1 ของรีจิสเตอร์ TCON แฟลกที่เกี่ยวข้องกับการอินเตอร์รัพต์จากภายนอกคือ IE0 และ IE1 ในรีจิสเตอร์ TCON เมื่อเกิดสัญญาณอินเตอร์รัพต์จากภายนอก แฟลกดังกล่าวก็จะถูกทำให้เป็น 0 ด้วยฮาร์ดแวร์เป็นการป้องกันไม่ให้เกิดอินเตอร์รัพต์ซ้อนกันและเป็นการบอกว่าและเป็นการบอกว่าเป็นการอินเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รัพต์ที่เกิดจากสัญญาณอินเทอร์รัพต์แบบขอบขาลงถ้าเป็นการอินเตอร์รัพต์ด้วยสัญญาณระดับต่ำก็จะเป็นการร้องขอแฟลกให้เป็น 0 ตลอดเวลาที่สัญญาณเข้าที่ขาอินเตอร์รัพต์ยังคงเป็น 0

การอินเตอร์รัพต์ตัวจับเวลา 0 และตัวจับเวลา 1 จะทำงานด้วยค่าต่างๆ ของแฟลก TF0 และ TF1 ที่เซ็ตเป็น 1 เมื่อค่าในรีจิสเตอร์ของตัวจับเวลา/ตัวนับถูกเพิ่มจาก 1 หมดทุกบิตกลายเป็น 0 หมดทุกบิต หลังจากเกิดอินเตอร์รัพต์ตัวจับเวลาแฟลก ดังกล่าวจะถูกเคลียร์ให้เป็น 0 เมื่อฮาร์ดแวร์บนชิปเข้าทำงานในโปรแกรมบริการของการอินเตอร์รัพต์ตัวจับเวลาแล้ว

การอินเตอร์รัพต์สามารถที่จะถูกกำหนด ให้ทำงานหรือยกเลิกได้ ในขณะใดขณะหนึ่งของการทำงานด้วยการติดตั้งด้วยซอฟแวร์ แหล่งกำเนิดอินเตอร์รัพต์แต่ละชนิดสามารถที่จะอีนาเบิ้ล หรือดิสเอเบิ้ลด้วยการเซตหรือเคลียร์ค่าบิต ภายในตัวรีจิสเตอร์ IE บิต EA ภายในรีจิสเตอร์ IE จะเป็นตัวควบคุมการอินเตอร์รัพต์ทุกชนิดให้เริ่มทำงานตามตารางที่ 3.4.1

ตารางที่ 3.4.1 รีจิสเตอร์การอินเตอร์รัพต์อีนาเบิ้ล : IE

EA	-	ET2	ES	ET1	EX1	ET0	EX0

ลักษณะสัญญลักษณ์ ตำแหน่งบิตและฟังค์ชั่น

EA : IE.7 ; จะดิสเอเบิ้ลอินเทอร์รัพต์ทั้งหมด ถ้า EA = 0 จะไม่มีการอินเตอร์รัพต์ในการตอบรับ ถ้า EA = 1 สามารถที่จะอินเตอร์รัพต์ได้ โดยแต่ละแหล่งอินเตอร์รัพต์มีอิสระในการเซตหรือเคลียร์ให้อีนาเบิ้ลแต่ละบิตก่อนได้

- : IE.6 ; สำรอง

ET2 : IE.5 ; จะอีนาเบิ้ลหรือดิสเอเบิ้ลอินเตอร์รัพต์ Overflow ของตัวจับเวลา 2 ถ้า ET = 0 การอินเตอร์รัพต์ตัวจับเวลา 2 จะดิสเอเบิ้ล

ES : IE.4 ; จะอีนาเบิ้ลหรือดิสเอเบิ้ลอินเตอร์รัพต์ของพอร์ตอนุกรม ถ้า ES = 0 การอินเทอร์รัพต์พอร์ตอนุกรมจะดิสเอเบิ้ล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า ET = 0 การอินเตอร์รัพต์ตัวจับเวลา 1 จะดิสเอเบิ้ล

EX1 : IE.2 ; จะอีนาเบิ้ลหรือดิสเอเบิ้ลอินเตอร์รัพต์จากภายนอก 1 ถ้า EX1 = 0 การอินเตอร์รัพต์จากภายนอก 1 จะดิสเอเบิ้ล

ET0 : IE.1 ; จะอีนาเบิ้ลหรือดิสเอเบิ้ลการอินเตอร์รัพต์ Overflow ของตัวจับเวลา 0 ถ้า ET0 = 0 การอินเตอร์รัพต์ตัวจับเวลา 0 จะดิสเอเบิ้ล

EX0 : IE.0 ; จะอีนาเบิ้ลหรือดิสเอเบิ้ลอินเตอร์รัพต์จากภายนอก 0 ถ้า EX0 = 0 การอินเตอร์รัพต์จากภายนอก 0 จะดิสเอเบิ้ล

3.4.2 โครงสร้างลำดับความสำคัญการอินเตอร์รัพต์

แต่ละแหล่งอินเตอร์รัพต์สามารถที่จะโปรแกรมให้มีระดับไพโอริตี้สูงหรือต่ำได้ด้วยการเซตหรือเคลียร์ค่าบิตต่างๆใน IP ของ SFR ตามตารางที่ 3.4.2 เป็นรายละเอียดของ IP โดยที่ตัวแฟลกอินเตอร์รัพต์ทุกตัวสามารถที่จะเซตหรือเคลียร์ ได้ด้วยซอฟแวร์ซึ่งจะมีผลเช่นเดียวกับผลที่เกิดจากฮาร์ดแวร์การอินเตอร์รัพต์ความสำคัญต่ำสามารถถูกอินเตอร์รัพต์ที่มีความสำคัญสูงกว่า แต่จะไม่สามารถถูกอินเตอร์รัพต์จากตัวอื่น ที่มีความสำคัญต่ำ

ตารางที่ 3.4.2 แสดงรายละเอียดของ IP

-	-	PT2	PS	PT1	PX1	PT0	PX0

ลักษณะสัญญลักษณ์ ตำแหน่งบิตและฟังค์ชั่น

- ; IP.7 ; สำรอง

- ; IP.6 ; สำรอง

PT2 ; IP.5 ; กำหนดระดับความสำคัญการอินเตอร์รัพต์ตัวจับเวลา 2 ถ้า PT2 = 1 เป็นการโปรแกรมให้มีระดับการอินเตอร์รัพต์ความสำคัญสูงกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PT2 ; IP.5 ; กำหนดระดับความสำคัญการอินเตอร์รัพต์ตัวจับเวลา 2 ถ้า PT2 = 1 เป็นการโปรแกรมให้มีระดับการอินเตอร์รัพต์ความสำคัญสูงกว่า

PS ; IP.4 ; การโปรแกรมให้มีระดับการอินเตอร์รัพต์พอร์ทอนุกรม ถ้า PS = 1 เป็นการโปรแกรมให้มีระดับการอินเตอร์รัพต์ความสำคัญสูงกว่า

PT1 ; IP.3 ; กำหนดระดับความสำคัญอินเตอร์รัพต์ตัวจับเวลา 1

PX1 ; IP.2 ; กำหนดระดับความสำคัญภายนอก 1 (INT1)

PT0 ; IP.1 ; กำหนดระดับความสำคัญอินเตอร์รัพต์ตัวจับเวลา 0

PX0 ; IP.0 ; กำหนดระดับความสำคัญอินเตอร์รัพต์ภายนอก 0 (INT0)

PX0 ; IP.0 ; ในเหตุการณ์ที่มีการร้องขอของระดับความสำคัญเดียวกันถูกรับเข้ามาพร้อมกัน การหาลำดับการใช้ร่วมกันก่อนหลังภายในเมื่อการร้องขอได้รับบริการ ดังนั้นระดับความสำคัญภายในแต่ละอัน จะมีการติดตั้งระดับตามโครงสร้างความสำคัญการอินเตอร์รัพต์ที่มีการใช้ร่วมกันก่อนหลังดังตารางที่ 3.4.3

ตารางที่ 3.4.3 แสดงแหล่งที่มาการ interrupt

แหล่งที่มาการอินเตอร์รัพต์	แหล่งที่มาการอินเตอร์รัพต์
ลำดับความสำคัญภายใน การอินเตอร์รัพต์ 0 จากภายนอก (IE0)	(สูงสุด) 1
การเกิด Overflow ของตัวจับเวลา/ตัวนับ 0 (TF0)	2
การอินเตอร์รัพต์ 1 จากภายนอก (IE1)	3
การเกิด Overflow ของตัวจับเวลา/ตัวนับ 1 (TF1)	4
พอร์ตอนุกรม (RI+TI)	5
การเกิด Overflow ของตัวจับเวลา/ตัวนับ 2 การเกิด Overflow ของตัวจับเวลา 2/ขอบขาลงบน T2EX	(ต่ำสุด) 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเตอร์รัพต์แฟลกจะถูกแซมปลิ้งทุกช่วงวัฏจักรแมชชีนช่วง S5P2 ตัวที่ถูกแซมปลิ้งจะเข้าได้ในช่วงวัฏจักรแมชชีนตัวต่อมา ถ้าแฟลกตัวหนึ่งถูกเซตในช่วง S5P2 ของช่วงวัฏจักรตัวต่อมา ในช่วงวัฏจักรการตรวจพบนี้ระบบการอินเตอร์รัพต์จะเริ่มการทำงานด้วยการทำคำสั่ง LCALL เพื่อที่จะนำเข้าสู่การบริการอินเตอร์รัพต์นั้นๆ โดยสภาวะทางฮาร์ดแวร์ คำสั่ง LCALL จะไม่ถูกกันออกเมื่อเกิดเหตุการณ์ต่อไปนี้

1. การอินเตอร์รัพต์ของระดับความสำคัญที่เท่ากันหรือสูงกว่าได้ดำเนินงานไปแล้ว

2. วัฏจักรที่กำลังดำเนินงานอยู่ซึ่งไม่ใช่เป็นวัฏจักรสุดท้ายในการทำงานตามคำสั่งที่กำลังทำอยู่หรือการทำคำสั่งขณะนั้นจะต้องสิ้นสุดลงสมบูรณ์ การอินเตอร์รัพต์ที่ร้องขอเข้ามาจึงจะได้รับการตอบสนอง

3. คำสั่งในขณะนั้นเป็น RETI หรือการเขียนโปรแกรมเข้ารีจิสเตอร์ IE หรือ IP ของ SFR หรือ การร้องขอการอินเตอร์รัพต์ จะยังไม่ได้การตอบรับหลังคำสั่ง RETI หรือหลังการอ่านและเขียนข้อมูลเข้ารีจิสเตอร์ IE หรือ IP และจะได้รับการตอบรับได้จนกว่าจะทำคำสั่งอย่างน้อยหนึ่งคำสั่งต่อมาของโปรแกรมหลักไปแล้ว ถ้าไม่เกิดกรณีใดจากข้างบนนี้ปรากฏผลของการใช้อินเตอร์รัพต์ จะทำงานช่วงวัฏจักรแมชชีนลูกต่อมา

3.4.3 การตอบสนองการอินเตอร์รัพต์โปรโตคอล

การทำงานอินเตอร์รัพต์จะเริ่มการทำงานจาก Vector แอดเดรสที่ให้มาดังแสดงในตารางที่ 3.4.4 จนกระทั่งถึงคำสั่ง RETI ซึ่งอยู่ในโปรแกรมย่อยการบริการอินเตอร์รัพต์ต่างๆดังกล่าวข้างบน คำสั่ง RETI จะเคลียร์แอกตีฟระดับความสำคัญ ของการอินเตอร์รัพต์ของฟลิบฟลอบที่ถูกเซต เมื่อได้การร้องขอ การอินเตอร์รัพต์ในตอนแรก

ตารางที่ 3.4.4 แสดงตำแหน่ง address ของการ interrupt

ตำแหน่งอินเตอร์รัพต์	ตำแหน่ง Vector แอดเดรส
EXTERNAL INTERRUPT 0 (IE0)	0003H
TIMER 0 OVERFLOW (TF0)	000BH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EXTERNAL INTERRUPT 1 (IE0)	0013H
TIMER 1 OVERFLOW (TF1)	001BH
SERIAL PORT (RI+TI)	0023H
TIMER 2	002BH

3.4.4 การอินเตอร์รัพต์จากภายนอก

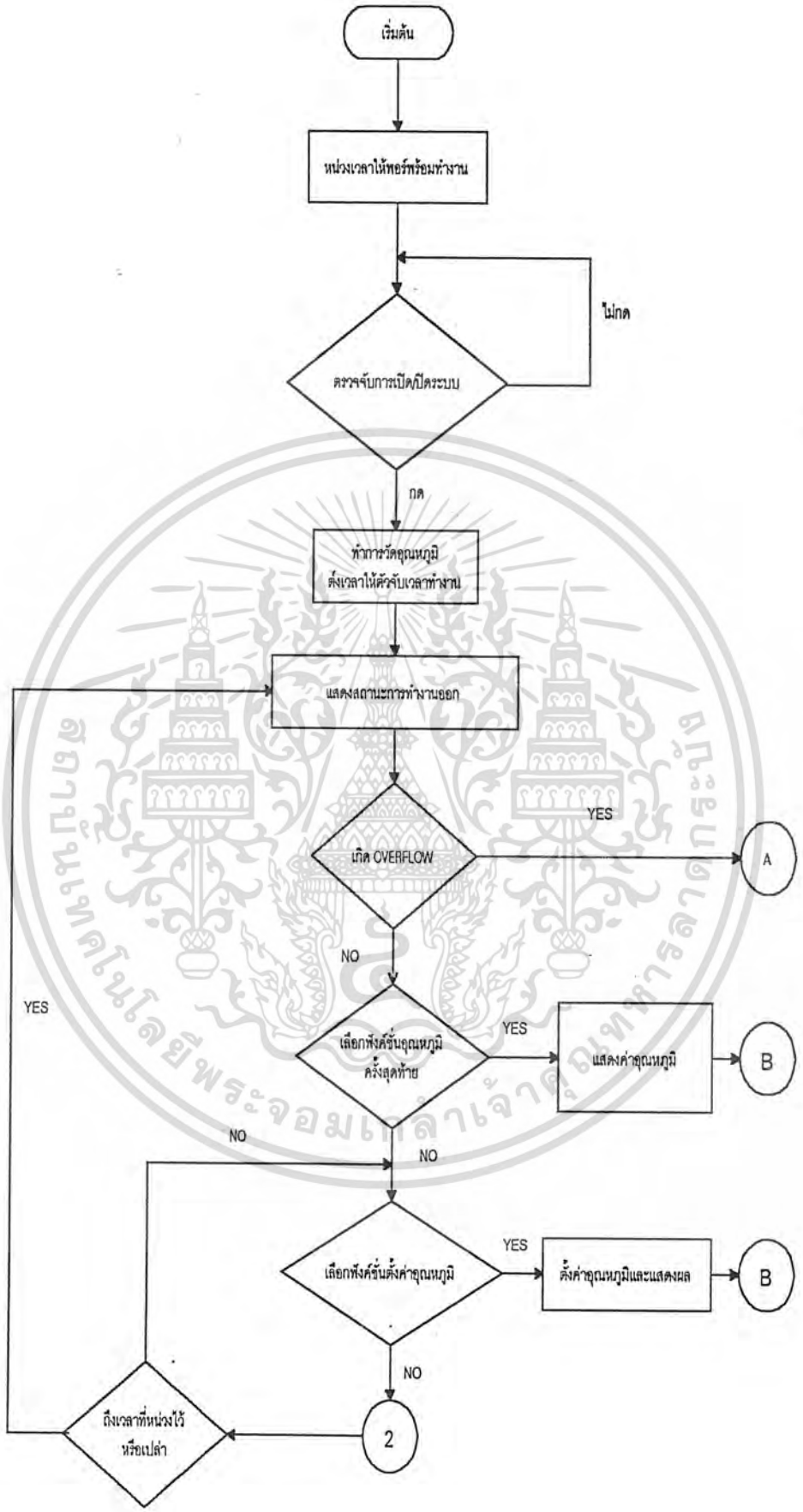
แหล่งกำเนิดภายนอกสามารถที่จะถูกโปรแกรมเลือกระดับการอินเตอร์รัพต์หรือช่วงการเปลี่ยนแแปลงด้วยการเซตหรือเคลียร์บิตที่ IT1 หรือ IT0 ในรีจิสเตอร์ TCON ถ้า ITx = 0 การอินเตอร์รัพต์ภายนอก x จะถูกกระตุ้นด้วยการกระตุ้นที่ระดับต่ำที่ขา INTx แต่ถ้า ITx = 1 การอินเตอร์รัพต์ภายนอก x เป็นการกระตุ้นใช้ขอบสัญญาณขาลง ในโหมดนี้ ถ้าตัวอย่างของสัญญาณของขา INTx แสดงถึงระดับสูงในวัฏจักรสูงในวัฏจักรลูกหนึ่งและต่ำในวัฏจักรอีกลูกหนึ่งแฟลกการร้องขออินเตอร์รัพต์ IEx ในรีจิสเตอร์ TCON จะถูกเซต ดังนั้น แฟลกบิต IEx จะเป็นการแสดงถึงการร้องขออินเตอร์รัพต์ เพราะสัญญาณที่ขาการอินเตอร์รัพต์ จะถูกสุ่มตัวอย่างหนึ่งครั้ง ในแต่ละวัฏจักรแมชชีน สัญญาณที่เข้าจะต้องรักษาระดับแรงดันสูงหรือต่ำอย่างน้อยภายใน ช่วง 12 คาบของความถี่ออสซิลเลเตอร์เพื่อให้มั่นใจว่าในการสุ่มตัวอย่างที่รับเข้าไปได้ค่าที่แน่นอนถ้าการอินเตอร์รัพต์ภายนอกถูกแอคทีฟเปลี่ยนแปลง แหล่งสัญญาณภายนอกจะต้องรักษาค่าการร้องขอสถานะสูง เป็นเวลาอย่างน้อยหนึ่งลูก และรักษาค่าสถานะต่ำอีกอย่างน้อยเป็นเวลาหนึ่งวัฏจักร เพื่อให้แน่ใจว่าการเปลี่ยนแปลงค่าจะสามารถทำให้แฟลกการร้องขออินเตอร์รัพต์ของ IEx ถูกเซต ค่าใน IEx จะถูกเคลียร์โดยอัตโนมัติด้วยซีพียูเมื่อโปรแกรมการบริการอินเตอร์รัพต์ถูกเรียกใช้

ถ้าสัญญาณอินเตอร์รัพต์ภายนอกอยู่ในระดับแอกทีฟ แหล่งภายนอกจะต้องเก็บการแอกทีฟการร้องขอไว้จนกว่าสัญญาณการร้องขออินเตอร์รัพต์จะถูกสร้างขึ้นมาเรียบร้อยแล้ว แล้วมันจะต้องกลับมารับแอกทีพการร้องขอใหม่ ก่อนที่การทำงานบริการอินเตอร์รัพต์เดิม จะสิ้นสุดลง หรือการอินเตอร์รัพต์อีกลูกหนึ่งจะถูกสร้างขึ้นมาใหม่

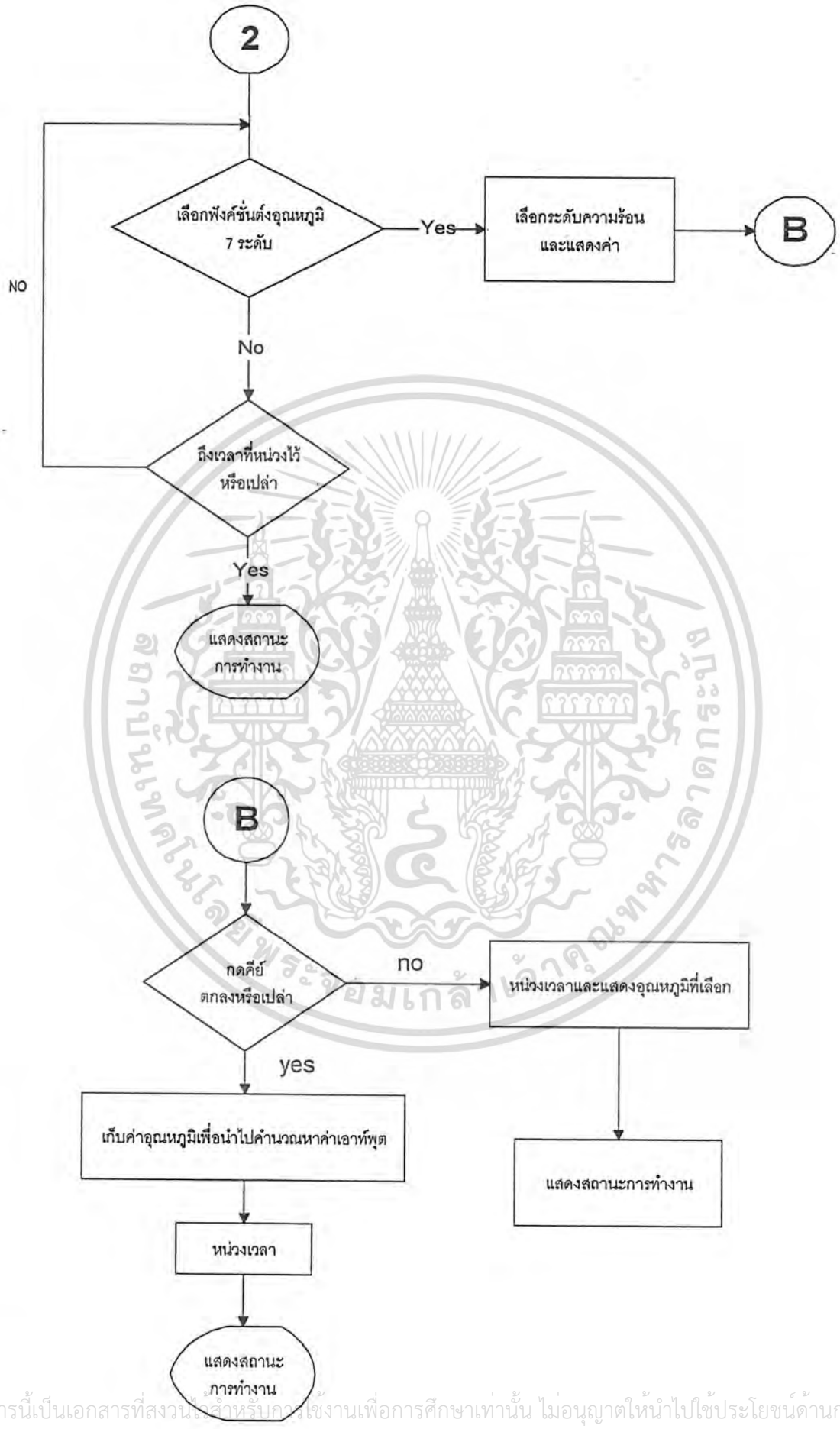
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOW CHART

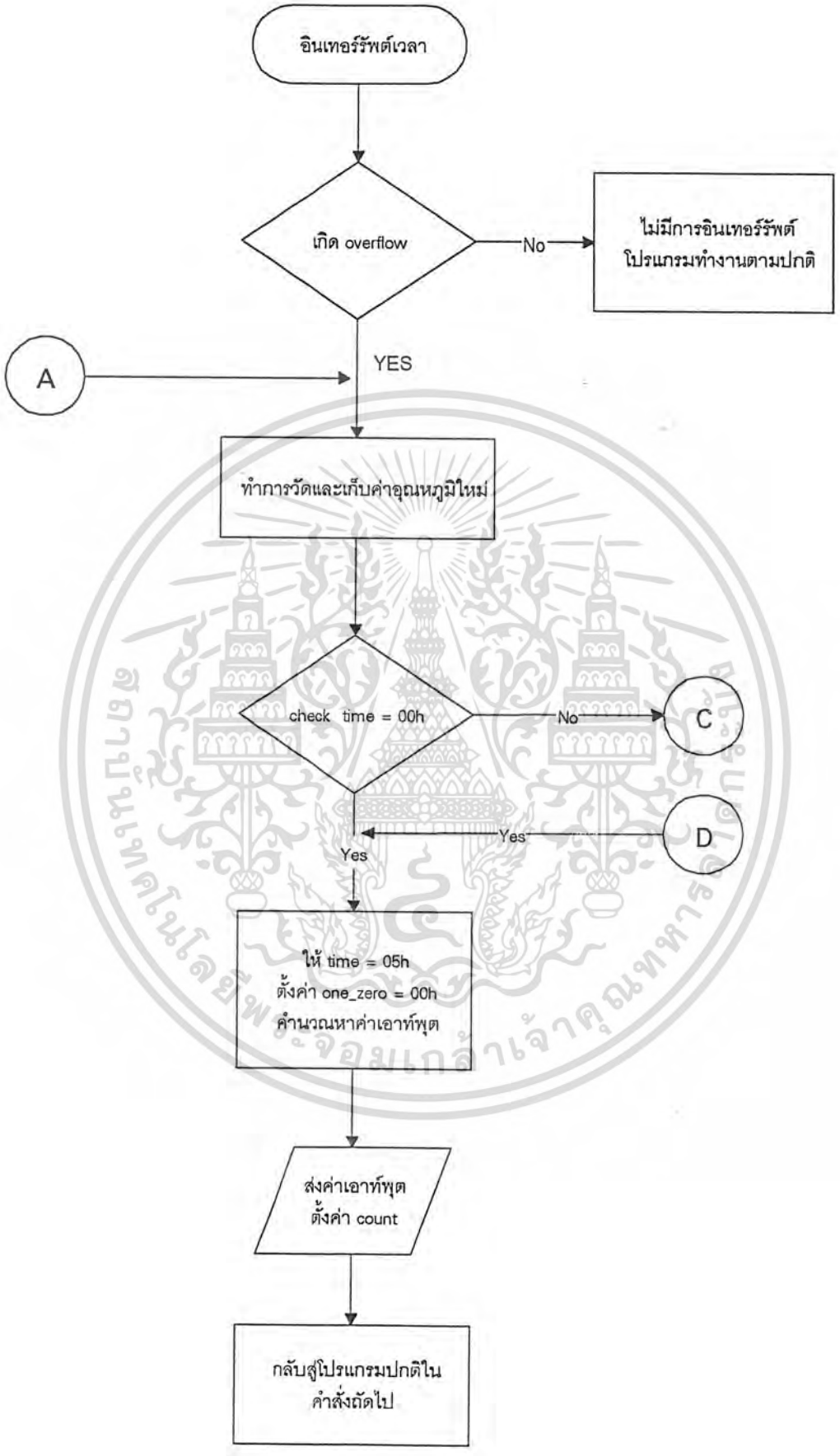
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



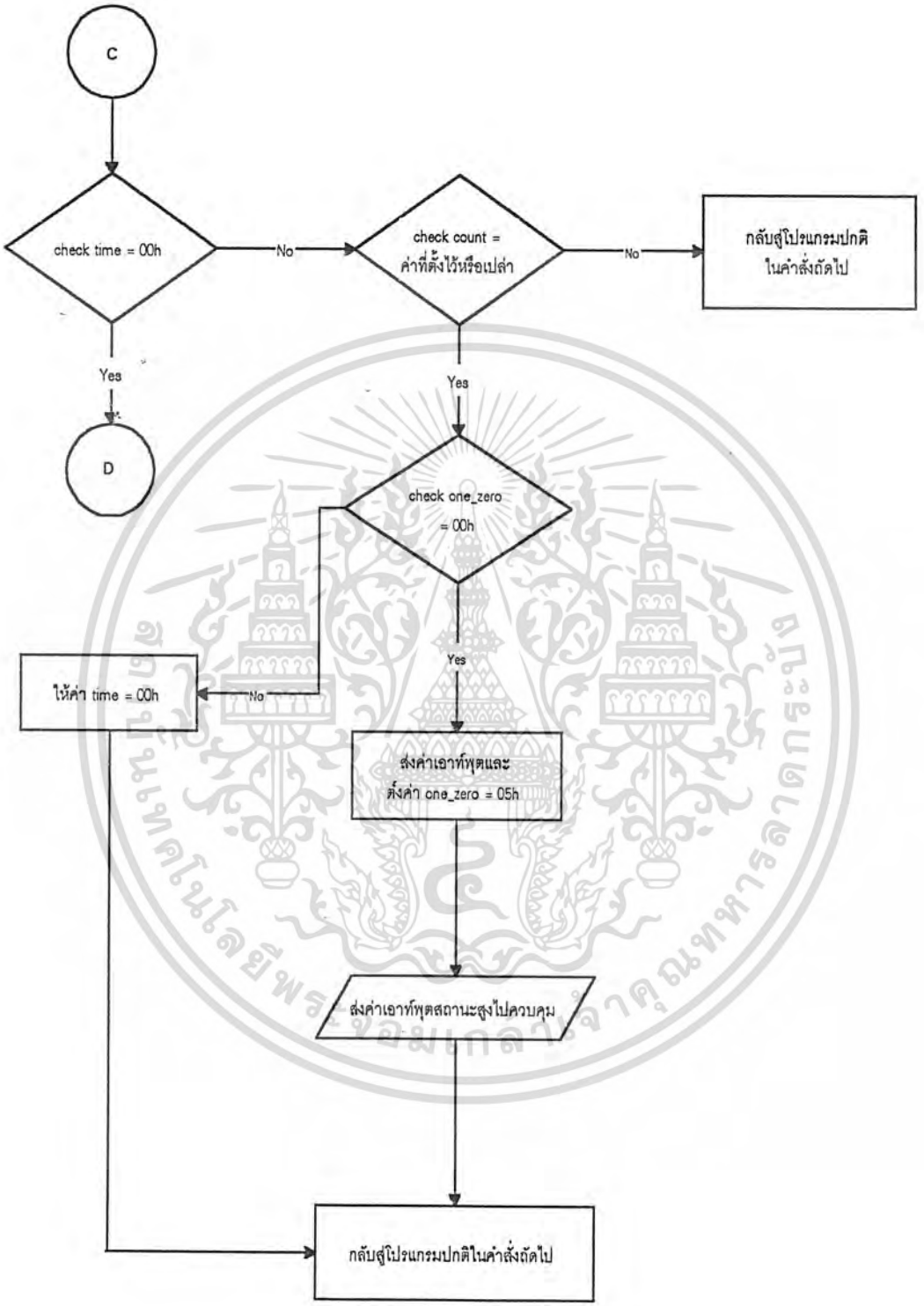
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Assembly Language

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายโครงสร้างของโปรแกรม Assembly

โปรแกรมนี้ถูกออกแบบให้มีฟังค์ชั่นเลือกใช้งานได้ 3 ฟังค์ชั่น คือ

1 โปรแกรมเก็บค่าอุณหภูมิครั้งสุดท้าย
2 โปรแกรมตั้งค่าอุณหภูมิตั้งแต่ 23.0 - 48.4 องศาเซลเซียส
3 โปรแกรมตั้งระดับความร้อนได้ 7 ระดับ

โดยมีคีย์ใช้งานอยู่ 5 คีย์ คือ

1. คีย์ ON/OFF
2. คีย์ FUNCTION
3. คีย์ INCREASE
4. คีย์ DECREASE
5. คีย์ OKAY

การทำงานเริ่มแรก เมื่อทำการเปิดระบบจะมีการหน่วงเวลาให้พอร์ตของ 8255 พร้อมที่จะใช้งาน และกำหนดค่าเริ่มต้นที่จำเป็นต่อการควบคุม เช่น กำหนดสถานะของ 8255 กำหนดรูปแบบการแสดงผลของจอ LCD กำหนดค่าเริ่มต้นให้กับตัวแปรต่างๆ หลังจากนั้น CPU จะสั่งให้อ่านค่าอุณหภูมิจาก ADC มาเก็บไว้ที่ E(t-1), E(t) พร้อมทั้งนำค่าที่อ่านได้แสดงออกจอ LCD ด้วยโปรแกรมย่อย Smile ต่อจากนั้นจะเข้าสู่โปรแกรม Check Key เพื่อตรวจสอบการกด Key ของผู้ใช้งานเมื่อผู้ใช้งานเลือกระดับความร้อนตามที่ตัวเองต้องการ โปรแกรม Check Key จะตรวจสอบการกดคีย์ OKAY ถ้าตกลง CPU ก็จะนำเอาค่าอุณหภูมิที่เลือกนั้นไปเก็บไว้ที่ Tset พร้อมกับนำค่าที่ได้ไปคำนวณหาค่าเอาท์พุต

ในส่วนของการแสดงผลจะมีโปรแกรมที่ใช้ในการเขียนตัวอักษรออกแสดงทางจอ LCD คือ โปรแกรมย่อย Write Screen การจับเวลาจะใช้ตัว Timer/Counter 0 โดยตัวโปรแกรมจะอยู่ในส่วน TIMER การรับค่าอุณหภูมิจากตัว ADC ใช้วิธีการขออินเตอร์รัพต์ INT0 ในการคำนวณค่าเอาท์พุต CPU จะนำเอาค่า E(t-1), E(t) และ Tset ไปใช้ในโปรแกรมย่อย Quant_Error ค่าที่ได้จากการคำนวณจะนำไปเปรียบเทียบกับตาราง Fuzzy Output เพื่อกำหนดค่าในการควบคุมการเปิด ปิดของ Heater (Central 1 - Central 8) ซิกิลเวอร์ที่ใช้เป็นตัวจับเวลาการ เปิด ปิด ของ Heater ก็คือ Timer/Counter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
prta1     equ     4000h
prtb1     equ     4001h
prtc1     equ     4002h
ctlw1     equ     4003h
ad_cs     equ     0c000h
keep      equ     30h
code      equ     31h  ; stack code for move code-
byte
mark      equ     32h  ; position to write
sub_mk    equ     33h  ; mark address of dd_ram
wd_len    equ     34h  ; word length
tem       equ     35h  ; temperature
decup     equ     36h  ; check inc dec
a_keep    equ     37h
nuvo      equ     38h  ; value of function
bird equ          39h  ; dph of current
heart     equ     40h  ; dpl of current
t_water   equ     41h  ; water temperature
counter   equ     42h
paluse    equ     43h
tony equ          44h
high equ          45h  ;dph of set temp
low  equ          46h  ;dpl of set temp
t_keep    equ     47h  ; value show temp
shigh     equ     49h  ;store temp dph
slow      equ     4ah  ;store temp dpl
acc_save  equ     4bh
mall_1    equ     4ch
mall_2    equ     4dh
bata      equ     4eh
pata equ          4fh
mali equ          50h
st_fast   equ     51h  ;step fast  inc
st_ft     equ     52h  ;step fast  dec
superh    equ     53h  ;old temp
superl    equ     54h  ;old temp
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
rose        equ         55h
error       equ         56h  ;quanti of error
ch_error    equ    57h  ;quanti of change_error
tt          equ         58h  ;temp of T(t)
tt_1        equ         59h  ;temp of T(t-1)
tset        equ         5ah  ;temp set
t_old       equ         5bh  ;temp old
value_3     equ         5ch  ;value of func3
one_zero    equ         5dh  ;check one or zero
sour        equ         5eh  ;check couter
count       equ         5fh  ;counter
util        equ         60h  ;output

            org     0000h
            ajmp    boot

            org     0003h
            ljmp    temp_int

            org     000bh
            ljmp    time_int

            org     0100h
boot:       mov     r7,#0ffh      ; delay for reset 8255 ports

dly1:       mov     r6,#0ffh
dly2:       djnz    r6,dly2
            djnz    r7,dly1
prog:       mov     dptr,#ctlw1   ; initial portA & portB for LCD
            mov     a,#80h
            movx    @dptr,a
            lcall   init_lcd
            mov     sp,#65h
            mov     ip,#01h       ;set priority int
default:    mov     dptr,#temper2
            mov     shigh,dph
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     slow,dpl
                mov     t_old,#87h
                mov     tset,#87h
                mov     util,#00h
wr_form:        setb    p1.6
                clr     ea
                mov     superh,shigh
                mov     superl,slow
                mov     t_old,a_keep      ;นำค่าอุณหภูมิที่ใช้ค่าสุดท้าย
                mov     mark,#00h         ;,มาเก็บค่าใน t_old
                mov     wd_len,#10h
                mov     dptr,#mscrn       ;display system off
                lcall   wr_scrn
power_on:       jb      p1.4,$
                jnb     p1.4,$
                mov     dptr,#onp         ;display on
                mov     mark,#0ch
                mov     wd_len,#03h
                acall   wr_scrn
                lcall   ov_delay
                mov     ie,#83h           ;เซตค่าอินเตอร์รัพต์
                lcall   meassure          ;อ่านค่าอุณหภูมิ
                mov     tt_1,t_water      ;store T(t-1)
initial:        lcall   timer             ;เริ่มจับเวลา
                mov     nuvo,#00h
                mov     dptr,#temper2
                mov     slow,dpl
                mov     shigh,dph
;********** display current temperature ***************
smile:          lcall   meassure                ;แสดงค่าอุณหภูมิปัจจุบัน
                mov     mark,#00h
                mov     wd_len,#10h
                mov     dptr,#current
                lcall   wr_scrn
                mov     mark,#0ah
                mov     code,#01h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    mov     r3,#04h
                    lcall   john            ;เรียกโปรแกรมย่อยแสดงค่า
                                            ;อุณหภูมิปัจจุบัน
;********************* check key ************************
chk_key:            jb      p1.4,start1
                    jnb     p1.4,$
                    ajmp    power_off
                    mov     paluse,#00h
start1:             jb      p1.0,exit1      ;if p1.0 high exit check key
                    jnb     p1.0,$
start:              mov     r4,#03h
start2:             dec     r4
                    mov     a,r4
                    jz      start
                    cjne    r4,#02h,func2
                    ajmp    function
func2:              cjne    r4,#01h,start
                    ajmp    temp
exit1:              ajmp    smile
power_off:          ljmp    wr_form                 ; ปิดการทำงาน
;       ******************* loop key ******************* p1.4 key  on/off
                                                        ;p1.0 key  function
                                                        ;p1.1 key  ok
loop2:              jb      p1.2,loop2_1            ;p1.2 key  inc
                    jnb     p1.2,f_inc1             ;p1.3 key  dec
f_inc1:             mov     r7,#00h
f_inc:              mov     r6,#00h                 ; โปรแกรมย่อย ตรวจสอบ
                    nop                             ;การกดคีย์
                    nop
                    djnz    r6,$
                    mov     r6,#00h
                    nop
                    nop
                    djnz    r6,$
                    jb      p1.2,step_inc
                    djnz    r7,f_inc
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    ljmp    fast_inc
step_inc:           mov     st_fast,#00h
                    ljmp    increase
loop2_1:            jb      p1.3,loop2_2
                    jnb     p1.3,f_dec1
f_dec1:             mov     r7,#00h
f_dec:              mov     r6,#00h
                    nop
                    nop
                    djnz    r6,$
                    mov     r6,#00h
                    nop
                    nop
                    djnz    r6,$
                    jb      p1.3,step_dec
                    djnz    r7,f_dec
                    ljmp    fast_dec
step_dec:           mov     st_ft,#00h
                    ljmp    decrease
loop2_2:            jb      p1.0,loop2_3    ;function
                    jnb     p1.0,$
                    mov     a,decup
                    cjne    a,#01h,start2
                    ajmp    chk_key
loop2_3:            jb      p1.1,loop2_4
                    jnb     p1.1,$
                    ajmp    okay
loop2_4:            jb      p1.4,loop2
                    jnb     p1.4,$
                    ljmp    wr_form
exit;               ret
;*************************** inc dec ****************************
fast_inc:           mov     st_fast,#05h        ; โปรแกรมย่อย แสดงค่า
increase:           mov     decup,#01h          ; เลือกค่าอุณหภูมิของ
                    mov     a,nuvo              ; ฟังค์ชั่นset_temperature
                    cjne    a,#01h,inc
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ajmp    loop2
inc:            mov     a,st_fast
                cjne    a,#05h,step_inc2
chk_fast:       jb      p1.2,loop2
                lcall   inc_set
                ajmp    chk_fast
step_inc2:      acall   inc_set
                ljmp    loop2
in c_set:       mov     dptr,#set_temp
                mov     mark,#00h
                mov     wd_len,#10h
                mov     code,#00h
                lcall   wr_scrn
                mov     dph,high
                mov     dpl,low
                mov     r1,#04h
peter:          inc     dptr
                djnz    r1,peter
                mov     a,#01h
                movc    a,@a+dptr
                mov     a_keep,a
                lcall   tmax
                inc     dptr
                mov     mark,#0ah
                mov     code,#01h
                mov     r3,#04h
                lcall   wr_l
                mov     high,dph
                mov     low,dpl
                ret
fast_dec:       mov     st_ft,#05h
decrease:       mov     decup,#01h          ; โปรแกรมย่อย การลดค่า
                mov     a,nuvo              ;อุณหภูมิ
                cjne    a,#01h,dec
                ajmp    loop2
dec:            mov     a,st_ft
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                cjne    a,#05h,step_dec2
chk_fast2:      jb      p1.3,t_fast
                acall   dec_set
                ajmp    chk_fast2
step_dec2:      acall   dec_set
t_fast:         ajmp    loop2
dec_set:        mov     dptr,#set_temp
                mov     mark,#00h
                mov     code,#00h
                mov     wd_len,#10h
                lcall   wr_scrn
                mov     dph,high
                mov     dpl,low
                mov     a,dpl
                clr     c
                subb    a,#06h
                mov     dpl,a
                jc      pai
                ajmp    sit
pai:            dec     dph
sit:            mov     a,#01h
                movc    a,@a+dptr
                mov     a_keep,a
                lcall   tmin
                inc     dptr
                mov     mark,#0ah
                mov     code,#01h
                mov     r3,#04h
                call    wr_l
                mov     high,dph
                mov     low,dpl
                ret
okay:           mov     shigh,high      ; โปรแกรมย่อยเก็บค่า
                mov     slow,low        ; อุณหภูมิที่เลือก
                mov     paluse,#01h
                mov     decup,#00h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     a,nuvo
                cjne    a,#02h,s_ok
                mov     tset,a_keep
                mov     tset,t_old
displayo:       mov     nuvo,#00h           ;. แสดงค่าที่เลือกออก
                mov     dptr,#ok            ; จอ LCD
                mov     mark,#00h
                mov     wd_len,#0Ah
                lcall   wr_scrn
                mov     r5,#00h
sub_ok:         mov     mark,#0ah
                mov     code,#01h
                mov     r3,#04h
                mov     dph,shigh
                mov     dpl,slow
                lcall   wr_l
                lcall   delay2
                inc     r5
                mov     a,#06h
                subb    a,r5
                jz      end_ok
                mov     dptr,#bnk
                mov     mark,#0ah
                mov     r3,#04h
                mov     code,#01h
                lcall   wr_l
                lcall   delay2
                ajmp    sub_ok
end_ok:         ljmp    smile               ; แสดงค่าอุณหภูมิปัจจุบัน

tmax:           mov     a,a_keep            ; โปรแกรมย่อยเช็คค่า
                cjne    a,#0ffh,no_limit    ; อุณหภูมิค่าสูงสุดและ
                mov     tony,#01h           ; ค่าต่ำสุดที่เลือกได้
                ljmp    over1
tmin:           mov     a,a_keep
                cjne    a,#00h,no_limit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    mov     tony,#00h
                    ljmp    over1
no_limit:           ret
;******************  Function old temperature ***********
function:           mov     decup,#00h
                    mov     nuvo,#01h
                    mov     dptr,#old_temp
                    mov     mark,#00h
                    mov     wd_len,#10h
                    lcall   wr_scrn
                    mov     high,superh
                    mov     low, superl
                    mov     dph,high
                    mov     dpl,low
                    mov     mark,#0ah
                    mov     code,#01h
                    mov     r3,#04h
                    lcall   wr_l
                    ajmp    loop2
;***************  function set_tempterature *************
temp:               mov     decup,#00h          ; ฟังค์
                    mov     nuvo,#02h
                    mov     mark,#00h
                    mov     high,shigh
                    mov     low,slow
                    mov     dptr,#set_temp      ;display set_temp
                    mov     code,#00h
                    mov     wd_len,#10h
                    lcall   wr_scrn
                    mov     r5,#00h
light:              mov     mark,#0ah           ;display temperature
                    mov     dph,high
                    mov     dpl,low
                    mov     r3,#04h
                    mov     code,#01h           ;wait
                    lcall   wr_l
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
stone:          jb      p1.1,metal
                jnb     p1.1,$
                ajmp    okay
metal:          jb      p1.2,fire
                jnb     p1.2,$
                ajmp    temp
fire:           jb      p1.3,rock
                jnb     p1.3,$
                ajmp    temp
rock:           jb      p1.4,inca
                jnb     p1.4,$
                ajmp    wr_form
inca:           ajmp    stone
over2:          mov     dptr,#st_min
                mov     wd_len,#11h
                lcall   wr_scrn
                ajmp    stone
ov_delay:       mov     r7,#00h         ; delay for warnning statement
ov0:            mov     r6,#00h
ov1:            mov     r5,#02h
ov2:            nop
                nop
                djnz    r5,ov2
                djnz    r6,ov1
                djnz    r7,ov0
                ret
;           ***************** initial LCD routines ******************
;             #Pa0 - Pa7 : D0 - D7  (data read/write LCD)
;             #Pb2       : E        (enable signal pluse)
;             #Pb1       : R/W      (read/write)
;             #Pb0       : RS       (register selection)

init_lcd:       mov     a,#00h
                mov     dptr,#prtb1
                movx    @dptr,a         ; e = r/w = rs = low state
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     a,#38h          ;  function set
        dec     dpl                     ; format 0 0 1 DL N F . .
                movx    @dptr,a         ; DL = 1 transfer data 8 bit format
                                        ; N  = 1 show >= 2 line
                                        ; F  = 0 cursor 5x7 dot format
                lcall   epluse          ; enable signal pluse
                lcall   dl_lcd          ; delay > 4.1 ms
                mov     a,#0ch          ; display on/off control
                mov     dptr,#prta1     ; format 0 0 0 0 1 D C B
                movx    @dptr,a         ; D = 1  display on
                lcall   epluse          ; C = 1  cursor on
                                        ; B = 1  blink of cursor position
                mov     a,#06h          ; entry mode set
                mov     dptr,#prta1     ; format 0 0 0 0 0 1 I/D S
                                        ; I/D = 1 increment DD RAM
address+1
                                        ; S shifts cursor and data
                                        ; S = 1 stay cursor , shift data
                                        ; S = 0 stay data , shift cursor
                movx    @dptr,a
                lcall   epluse

        clr_lcd:mov     a,#00h
                mov     dptr,#prtb1
                movx    @dptr,a
                mov     a,#01h          ; clear all display
                mov     dptr,#prta1     ; format 0 0 0 0 0 0 0 1
                movx    @dptr,a
                lcall   epluse
                ret

;   ************** generate enable pluse ******************

    epluse:     mov     dptr,#prtb1
                movx    a,@dptr         ; ckeck status of 3 signal control
                orl     a,#04h          ; enable signal pulse to high
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    movx    @dptr,a
                    mov     r6,#05h         ; delay pulse
dl_epl:             mov     r7,#00h
                    djnz    r7,$
                    djnz    r6,dl_epl
                    anl     a,#0fbh         ; clear signal pulse to low
                    movx    @dptr,a
                    ret

dl_lcd:             mov     r6,#0ah         ; LCD delay
dl_lcd1:            mov     r7,#00h
                    nop
                    nop
                    djnz    r7,$
                    djnz    r6,dl_lcd1
                    ret
;       ************** set dd_ram routines **************     โปรแกรมย่อยกำหนดตำแหน่ง
set_dd:             mov     dptr,#prtb1     ; dd_ram
                    mov     a,#00h
                    movx    @dptr,a
                    mov     a,sub_mk
                    orl     a,#80h
                    mov     dptr,#prta1
                    movx    @dptr,a
                    lcall   epluse
                    ret

;       ************** write screen ***************     โปรแกรมย่อยเขียนตัวอักษร
wr_scrn:            mov     code,#00h       ; ออกหน้าจอ LCD
                    mov     r3,wd_len
wr_l:               mov     a,code
                    movc    a,@a+dptr
                    mov     keep,a          ; read data
                    mov     a,mark
                    push    dph
                    push    dpl
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     dptr,#order
                movc    a,@a+dptr       ; read add' of dd-ram from order
                mov     sub_mk,a
                lcall   set_dd
                lcall   wr_char
                mov     a,code
                inc     a               ; inc add' of data
                mov     code,a
                mov     a,mark
                inc     a               ; inc pos' of dd-ram add'
                mov     mark,a
                pop     dpl
                pop     dph
                djnz    r3,wr_l
                ret
wr_char:        mov     dptr,#prtb1     ; write each character
                mov     a,#01h
                movx    @dptr,a
                mov     dptr,#prta1
                mov     a,keep
                movx    @dptr,a
                lcall   epluse
                ret

blink1:         mov     r6,#0aah        ; โปรแกรมย่อยให้ตัวอักษรกระพริบ
sub_b1:         mov     r7,#0aah
rcv1:           jb      p1.1,bl1
                jnb     p1.1,$
                ljmp    stop_temp
bl1:            jb      p1.2,bl2
                jnb     p1.2,$
                ljmp    stop_temp
bl2:            jb      p1.3,bl3
                jnb     p1.3,$
                ljmp    stop_temp
bl3:            jb      p1.4,bl4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    jnb     p1.4,$
                    ljmp    wr_form
bl4:                jb      p1.0,bl5
                    jnb     p1.0,$
                    ajmp    function_3
bl5:                djnz    r7,rcv1
                    djnz    r6,sub_b1
                    ret
;################### 7 level ###################
; ฟังค์ชั่นตั้งค่าอุณหภูมิ 7 ระดับ
function_3:         mov     nuvo,#03h
                    mov     r5,#00h
f_3:                mov     dptr,#func_3
                    mov     mark,#00h
                    mov     wd_len,#10h
                    lcall   wr_scrn
                    lcall   delay2
                    jnb     p1.0,out_func_3
                    jnb     p1.2,func3_1
                    jnb     p1.3,func3_1
                    jnb     p1.4,out_3_2
                    mov     dptr,#bnk
                    mov     mark,#09h
                    mov     wd_len,#01h
                    lcall   wr_scrn
                    lcall   delay2
                    inc     r5
                    mov     a,#06h
                    subb    a,r5
                    jz      out_func_3
                    ajmp    f_3
out_func_3:         ljmp    smile
out_3_2:            ljmp    wr_form
func3_1:            mov     r5,#04h
                    ajmp    level_4
find_l:             cjne    r5,#04h,ff_2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ajmp    level_4
ff_2:           cjne    r5,#05h,ff_3
                ajmp    level_5
ff_3:           cjne    r5,#06h,ff_4
                ajmp    level_6
ff_4:           cjne    r5,#07h,ff_5
                ajmp    level_7
ff_5:           cjne    r5,#01h,ff_6
                ajmp    level_1
ff_6:           cjne    r5,#02h,ff_7
                ajmp    level_2
ff_7:           cjne    r5,#03h,ff_8
                ajmp    level_3
ff_8:           cjne    r5,#00h,ff_9
                mov     r5,#07h
                ajmp    level_7
ff_9:           cjne    r5,#08h,ff_10
                mov     r5,#01h
                ajmp    level_1
ff_10:          ljmp    func_3
scan:           jb      p1.2,sc1        ; โปรแกรมย่อยเช็คการเลือก
                jnb     p1.2,$          ; ระดับความร้อน
                ajmp    inc_3
sc1:            jb      p1.3,sc2
                jnb     p1.3,$
                ajmp    dec_3
sc2:            jb      p1.1,sc3
                jnb     p1.1,$
                ajmp    okay_3
sc3:            jb      p1.4,scan
                jnb     p1.4,$
                ljmp    wr_form
inc_3:          inc     r5
                ajmp    find_l
dec_3:          dec     r5
                ajmp    find_l
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
level_4:    mov   dptr,#medium      ;แสดงระดับความร้อนที่
            mov   value_3,#04h      ; เลือกออกจอ LCD
            ajmp  write
level_5:    mov   dptr,#hi_lev
            mov   value_3,#05h
            ajmp  write
level_6:    mov   dptr,#vhi_lev
            mov   value_3,#06h
            ajmp  write
level_7:    mov   dptr,#highest
            mov   value_3,#07h
            ajmp  write
level_1:    mov   dptr,#lowest
            mov   value_3,#01h
            ajmp  write
level_2:    mov   dptr,#vl_lev
            mov   value_3,#02h
            ajmp  write
level_3:    mov   dptr,#l_lev
            mov   value_3,#03h
write:      mov   mark,#00h
            mov   wd_len,#10h
            lcall wr_scrn
            ajmp  scan
okay_3:     mov   a,value_3         ; กำหนดค่าอุณหภูมิของน้ำที่ระดับ
            cjne  a,#01h,oo_1       ; ความร้อนต่างกัน
            mov   tset,#46h         ;30 c
            ljmp  smile
oo_1:       cjne  a,#02h,oo_2
            mov   tset,#64h         ;33 c
            ljmp  smile
oo_2:       cjne  a,#03h,oo_3
            mov   tset,#78h         ;35 c
            ljmp  smile
oo_3:       cjne  a,#04h,oo_4
            mov   tset,#8ch         ;37 c
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ljmp    smile
oo_4:           cjne    a,#05h,oo_5
                mov     tset,#0aah      ;40 c
                ljmp    smile
oo_5:           cjne    a,#06h,oo_6
                mov     tset,#0beh      ;42 c
                jmp     smile
oo_6:           mov     tset,#0f1h      ;47 c

;************************ temp interrupt ***********************
meassure:       push    dph     ; โปรแกรมย่อยวัดค่าอุณหภูมิ
                push    dpl
                mov     dptr,#ad_cs     ;ad_cs fron adc
                movx    a,@dptr
                movx    @dptr,a
                setb    ex0             ;int 0 enable
                clr     c
                jnc     $
                pop     dpl
                pop     dph
                ret
temp_int:       mov     dptr,#ad_cs     ; โปรแกรมย่อยรับค่าอุณหภูมิจาก
                movx    a,@dptr         ; ADC
                mov     t_water,a
                clr     ex0
                setb    c
                reti
;************************ timer ******************************
timer:          mov     counter,#00h    ; ตั้งค่าเวลาการค่าอุณหภูมิ
                mov     count,#00h      ; จาก ADC
                mov     util,#00h
                mov     tl0,#00h
                mov     th0,#00h
                mov     tmod,#01h
                setb    tcon.4
                ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
time_int:       mov     acc_save,a
                mov     a,util
                cjne    a,#05h,toy
                ajmp    time_2
toy:            inc     counter
                mov     a,counter
                cjne    a,#01h,t_int      ;time int
                mov     counter,#00h
                clr     tcon.4
                lcall   meassure
                ljmp    quant_error       ; ส่งค่าอุณหภูมิที่อ่านได้ไปเ
t_int:          mov     a,acc_save        ; คำนวณหาเอาท์พุต
                reti
time_2:         inc     count
                mov     a,count
                cjne    a,sour,t2_int
                clr     tcon.4
                mov     count,#00h
                mov     a,one_zero
                cjne    a,#00h,tim_21
                ljmp    zero_2
tim_21:         ljmp    one_2
t2_int:     mov a,one_zero
                cjne    a,#00h,t2_int_1
                ljmp    zero1_2
t2_int_1:       ljmp    one1_2
;***********  display temperature  from ADC ********************
john:           mov     t_keep,t_water    ;t_water from adc
                mov     dptr,#temper
rd_set:         mov     a,#00h
                movc    a,@a+dptr
                cjne    a,t_keep,serch
                ajmp    end_ff
serch:          mov     r7,#05h
newstr:         inc     dptr
                djnz    r7,newstr
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ajmp    rd_set
end_ff:         mov     bird,dph
                mov     heart,dpl
                mov     mark,#0ah
                mov     code,#01h
                mov     r3,#04h
                lcall   wr_l
                lcall   delay3
                mov     dptr,#spot
                mov     mark,#08h
                mov     r3,#01h
                mov     code,#00h
                lcall   wr_l
                lcall   ov_delay
                ret
delay2:     mov r7,#0ffh            ; โปรแกรมหน่วงเวลา
del1:           mov     r6,#0ffh
del2:           jnb     p1.4,sd2
                jnb     p1.2,sd2
                jnb     p1.3,sd2
                jnb     p1.0,sd2
                nop
                djnz    r6,del2
                djnz    r7,del1
sd2:            ret
delay3:         mov     r7,#6ah         ; โปรแกรมหน่วงเวลา
del3_1:         mov     r6,#0aah
del3_2:         mov     r5,#06h
del3_3:         nop
                nop
                djnz    r5,del3_3
                djnz    r6,del3_2
                djnz    r7,del3_1
                ret
much:           mov     tt,#0ffh
                ajmp    mm
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;#################### fuzzy output ######################
quant_error:    mov     util,#05h       ; คำนวนหาค่า error
                mov     tt,t_water
                mov     a,tt
                clr     c
                add     a,#46h
                jc      much
                mov     tt,a
mm:             mov     a,tset
                clr     c
                cjne    a,tt,not_equal
                mov     error,#1fh
                jmp     end_p
not_equal:      jc      minus_0         ; check ผลต่างระหว่างค่า tset กับ
                ajmp    plus            ; t(t) ว่าเป็น บวก หรือ ลบ
minus_0:        ljmp    minus
plus:           mov     a,tset
                clr     c
                subb    a,tt
                cjne    a,#19h,plus_1
                mov     error,#00h
                ljmp    end_p
plus_1:         jc      plus_2          ; เมื่อ error เป็นค่าบวกกำหนด
                mov     error,#00h      ; ค่า quantize
                ljmp    end_p
plus_2:         cjne    a,#14h,plus_3
                mov     error,#02h
                ljmp    end_p
plus_3:         jc      plus_4
                mov     error,#02h
                ljmp    end_p
plus_4:         cjne    a,#0fh,plus_5
                mov     error,#04h
                ljmp    end_p
plus_5:         jc      plus_6
                mov     error,#04h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ljmp    end_p
plus_6:         cjne    a,#0ah,plus_7
                mov     error,#0eh
                ljmp    end_p
plus_7:         jc      plus_8
                mov     error,#0eh
                ljmp    end_p
plus_8:         cjne    a,#05h,plus_9
                mov     error,#0fh
                ljmp    end_p
plus_9:         jc      plus_10
                mov     error,#0fh
                ljmp    end_p
plus_10:        mov     error,#1fh
                ljmp    end_p
minus:          mov     a,tt            ; เมื่อค่า error เป็น -  กำหนดค่า
                clr     c               ; quantize
                subb    a,tset
                cjne    a,#19h,minus_1
                mov     error,#0ffh
                ljmp    end_p
minus_1:        jc      minus_2
                mov     error,#0ffh
                ljmp    end_p
minus_2:        cjne    a,#14h,minus_3
                mov     error,#0feh
                ljmp    end_p
minus_3:        jc      minus_4
                mov     error,#0feh
                ajmp    end_p
minus_4:        cjne    a,#0fh,minus_5
                mov     error,#0fdh
                ljmp    end_p
minus_5:        jc      minus_6
                mov     error,#0fdh
                ljmp    end_p
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
minus_6:        cjne    a,#0ah,minus_7
                mov     error,#6fh
                ljmp    end_p
minus_7:        jc      minus_8
                mov     error,#6fh
                ajmp    end_p
minus_8:        cjne    a,#05h,minus_9
                mov     error,#7eh
                ajmp    end_p
minus_9:        jc      minus_10
                mov     error,#7eh
                ajmp    end_p
minus_10:       mov     error,#1fh
end_p:          nop
;############## quanti of change_error #############
quanti_ch:      mov     a,tt_1          ; คำนวณหาค่า change_error
                clr     c
                cjne    a,tt,n_equal
                mov     ch_error,#1fh
                ljmp    end_cha
n_equal:        jc      negative_0      ; check ค่า change_error เป็น +,-
                ajmp    positive
negative_0:     ljmp    negative
positive:       mov     a,tt_1          ; กำหนดค่า quantize เมื่อ
                clr     c               ; change มีค่าเป็น +
                subb    a,tt
                cjne    a,#09h,pos_1
                mov     ch_error,#02h
                ljmp    end_cha
pos_1:          jc      pos_2
                mov     ch_error,#00h
                ljmp    end_cha
pos_2:          cjne    a,#07h,pos_3
                mov     ch_error,#04h
                ljmp    end_cha
pos_3:          jc      pos_4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     ch_error,#02h
                ljmp    end_cha
pos_4:          cjne    a,#05h,pos_5
                mov     ch_error,#0eh
                ljmp    end_cha
pos_5:          jc      pos_6
                mov     ch_error,#04h
                ljmp    end_cha
pos_6:          cjne    a,#03h,pos_7
                mov     ch_error,#0fh
                ljmp    end_cha
pos_7:          jc      pos_8
                mov     ch_error,#0eh
                ljmp    end_cha
pos_8:          cjne    a,#01h,pos_9
                mov     ch_error,#1fh
                jmp     end_cha
pos_9:          jc      end_cha
                mov     ch_error,#0fh
                ljmp    end_cha
negative:       mov     a,tt            ; กำหนดค่า quantize เมื่อ
                clr     c               ; เมื่อ change_error มีค่า -
                subb    a,tt_1
                cjne    a,#09h,nega_1
                mov     ch_error,#0feh
                jmp     end_cha
nega_1:         jc      nega_2
                mov     ch_error,#0ffh
                ljmp    end_cha
nega_2:         cjne    a,#07h,nega_3
                mov     ch_error,#0fdh
                ljmp    end_cha
nega_3:         jc      nega_4
                mov     ch_error,#0feh
                ljmp    end_cha
nega_4:         cjne    a,#05h,nega_5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     ch_error,#6fh
                ljmp    end_cha
nega_5:         jc      nega_6
                mov     ch_error,#0fdh
                ljmp    end_cha
nega_6:         cjne    a,#03h,nega_7
                mov     ch_error,#7eh
                ajmp    end_cha
nega_7:         jc      nega_8
                mov     ch_error,#6fh
                ajmp    end_cha
nega_8:         cjne    a,#01h,nega_9
                mov     ch_error,#1fh
                ajmp    end_cha
nega_9:         jc      end_cha
                mov     ch_error,#7eh
end_cha:        mov     tt_1,tt
look_up:        mov     a,error         ; กำหนดค่าเอาท์พุต
                cjne    a,#00h,q_1      ; ตามกฎต่างๆ ตามค่าที่ได้
                ljmp    central_8       ; จากการคำนวณ
q_1:            cjne    a,#02h,q_2
                ljmp    central_8
q_2:            cjne    a,#04h,q_3
                ljmp    central_8
q_3:            cjne    a,#0ffh,q_4
                ljmp    central_1
q_4:            cjne    a,#0feh,q_5
                ljmp    central_1
q_5:            cjne    a,#0fdh,q_6
                ljmp    central_1
q_6:            cjne    a,#0eh,q_7
                mov     a,ch_error
                jb      acc.7,r_1
                jb      acc.6,r_2
                jb      acc.4,r_1
                jb      acc.3,r_2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                jb      acc.0,r_5
r_1:            ajmp    central_8
r_2:            ajmp    central_5
r_5:            acall   central_3

q_7:            cjne    a,#0fh,q_8
                mov     a,ch_error
                jb      acc.7,s_1
                jb      acc.6,s_2
                jb      acc.4,s_3
                jb      acc.3,s_2
                jb      acc.0,s_5
s_1:            ljmp    central_8
s_2:            ljmp    central_5
s_3:            ljmp    central_6
s_5:            ljmp    central_3
q_8:            cjne    a,#1fh,q_9
                mov     a,ch_error
                jb      acc.7,u_4
                jb      acc.6,u_2
                jb      acc.4,u_1
                jb      acc.3,u_1
                jb      acc.0,u_2

q_9:            mov     a,ch_error
                jb      acc.7,u_1
                jb      acc.6,u_2
                jb      acc.4,u_3
                jb      acc.3,u_1
                jb      acc.0,u_3

u_1:            ljmp    central_2
u_2:            jmp     central_1
u_3:            ajmp    central_3
u_4:            ajmp    central_4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
central_8:      mov     util,#05h       ; กำหนดค่าเอาพุต พร้อมทั้งส่ง
                mov     mall_1,#01h     ; ไปควบคุณระบบ
                mov     pata,#0ffh
                mov     bata,#0feh
                mov     mall_2,#05h
                mov     rose,#06h
                mov     mali,#00h
c_8:            ajmp    output

central_6:      mov     util,#05h
                mov     mall_1,#02h
                mov     pata,#63h
                mov     bata,#0c0h
                mov     mall_2,#04h
                mov     rose,#15h
                mov     mali,#0a0h
                ajmp    output

central_5:      mov     mall_1,#02h
                mov     pata,#15h
                mov     bata,#0a0h
                mov     mall_2,#04h
                mov     rose,#3ch
                mov     mali,#0b0h
                ajmp    output

central_4:      mov     mall_1,#04h
                mov     pata,#63h
                mov     bata,#0c0h
                mov     mall_2,#04h
                mov     rose,#63h
                mov     mali,#0c0h
                ajmp    output

central_3:      mov     mall_1,#04h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     pata,#3ch
                mov     bata,#0b0h
                mov     mall_2,#02h
                mov     rose,#15h
                mov     mali,#0a0h
                ajmp    output

central_2:      mov     mall_1,#04h
                mov     pata,#15h
                mov     bata,#0a0h
                mov     mall_2,#02h
                mov     rose,#63h
                mov     mali,#0c0h
                ajmp    output

central_1:      mov     mall_1,#05h
                mov     pata,#25h
                mov     bata,#40h
                mov     mall_2,#01h
                mov     rose,#63h
                mov     mali,#0c0h
                ajmp    output
                                        ; ส่งค่าเอาท์พุตไปควบคุมระบบ
output:         clr     p1.6            ; clr   ie.0 clr int0
                ajmp    zero
out_2:          setb    p1.6
                ajmp    one
out_3:          mov     counter,#00h
                mov     tl0,#00h
                mov     th0,#00h
                setb    tcon.4
                mov     util,#00h
                reti

one:            mov     one_zero,#01h
                mov     sour,mall_1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
one1:           mov     tmod,#01h
one1_2:         mov     th0,pata
                mov     tl0,bata
                setb    tcon.4
                reti
one_2:          ajmp    out_3

zero:           mov     one_zero,#00h
                mov     sour,mall_2
zero1:          mov     tmod,#01h
zero1_2:        mov     th0,rose
                mov     tl0,mali
                setb    tcon.4
                setb    ie.1            ; ea
                reti
zero_2:         ajmp    out_2

;       ***************** screen's data ********************

order:          db      00h,01h,02h,03h,04h,05h,06h,07h,40h,41h,42h,43h,44h,45h,46h,47h
mscrn:          db      "  SYSTEM    OFF "
onp:            db      "ON "
current:        db      "Current      ",0dfh,"c"
showtim1:       db      " function 1    "
old_temp:       db      "Old_temp.    ",0dfh,"c"
set_temp:       db      "Set_temp.    ",0dfh,"c"
func_3:         db      "Set_temp 7 level"
st_max:         db      "Max Temp. 48.3",0dfh,"c"
st_min:         db      "Min Temp. 23.0",0dfh,"c"
hi_lev:         db      "QUITE HIGH TEMP."
vhi_lev:        db      "  HIGH  TEMP.  "
l_lev:          db      " QUITE LOW TEMP."
vl_lev:         db      "   LOW TEMP.   "
medium:         db      "  MEDIUM TEMP.  "
highest:        db      " HIGHEST  TEMP. "
lowest:         db      " LOWEST TEMP.   "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bnk:        db   '    '
spot:       db   3ah
ok:         db   ' O.K.TEMP.'
out_range:  db   '**Out of range**'
temper:     db   00h,'23.0',01h,'23.1',02h,'23.2',03h,'23.3',04h,'23.4'
            db   05h,'23.5',06h,'23.6',07h,'23.7',08h,'23.8',09h,'23.9'
            db   0ah,'24.0',0bh,'24.1',0ch,'24.2',0dh,'24.3',0eh,'24.4'
            db   0fh,'24.5',10h,'24.6',11h,'24.7',12h,'24.8',13h,'24.9'
            db   14h,'25.0',15h,'25.1',16h,'25.2',17h,'25.3',18h,'25.4'
            db   19h,'25.5',1ah,'25.6',1bh,'25.7',1ch,'25.8',1dh,'25.9'
            db   1eh,'26.0',1fh,'26.1',20h,'26.2',21h,'26.3',22h,'26.4'
            db   23h,'26.5',24h,'26.6',25h,'26.7',26h,'26.8',27h,'26.9'
            db   28h,'27.0',29h,'27.1',2ah,'27.2',2bh,'27.3',2ch,'27.4'
            db   2dh,'27.5',2eh,'27.6',2fh,'27.7',30h,'27.8',31h,'27.9'
            db   32h,'28.0',33h,'28.1',34h,'28.2',35h,'28.3',36h,'28.4'
            db   37h,'28.5',38h,'28.6',39h,'28.7',3ah,'28.8',3bh,'28.9'
            db   3ch,'29.0',3dh,'29.1',3eh,'29.2',3fh,'29.3',40h,'29.4'
            db   41h,'29.5',42h,'29.6',43h,'29.7',44h,'29.8',45h,'29.9'
            db   46h,'30.0',47h,'30.1',48h,'30.2',49h,'30.3',4ah,'30.4'
            db   4bh,'30.5',4ch,'30.6',4dh,'30.7',4eh,'30.8',4fh,'30.9'
            db   50h,'31.0',51h,'31.1',52h,'31.2' 53h,'31.3',54h,'31.4'
            db   55h,'31.5',56h,'31.6',57h,'31.7',58h,'31.8',59h,'31.9'
temper2:    db   5ah,'32.0',5bh,'32.1',5ch,'32.2',5dh,'32.3',5eh,'32.4'
            db   5fh,'32.5',60h,'32.6',61h,'32.7',62h,'32.8',63h,'32.9'
            db   64h,'33.0',65h,'33.1',66h,'33.2',67h,'33.3',68h,'33.4'
            db   69h,'33.5',6ah,'33.6',6bh,'33.7',6ch,'33.8',6dh,'33.9'
            db   6eh,'34.0',6fh,'34.1',70h,'34.2',71h,'34.3',72h,'34.4'
            db   73h,'34.5',74h,'34.6',75h,'34.7',76h,'34.8',77h,'34.9'
            db   78h,'35.0',79h,'35.1',7ah,'35.2',7bh,'35.3',7ch,'35.4'
            db   7dh,'35.5',7eh,'35.6',7fh,'35.7',80h,'35.8',81h,'35.9'
            db   82h,'36.0',83h,'36.1',84h,'36.2',85h,'36.3',86h,'36.4'
            db   87h,'36.5',88h,'36.6',89h,'36.7',8ah,'36.8',8bh,'36.9'
            db   8ch,'37.0',8dh,'37.1',8eh,'37.2',8fh,'37.3',90h,'37.4'
            db   91h,'37.5',92h,'37.6',93h,'37.7',94h,'37.8',95h,'37.9'
            db   96h,'38.0',97h,'38.1',98h,'38.2',99h,'38.3',9ah,'38.4'
            db   9bh,'38.5',9ch,'38.6',9dh,'38.7',9eh,'38.8',9fh,'38.9'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
db    9bh,"38.5",9ch,"38.6",9dh,"38.7",9eh,"38.8",9fh,"38.9"
db    0a0h,"39.0",0a1h,"39.1",0a2h,"39.2",0a3h,"39.3"
db    0a4h,"39.4",0a5h,"39.5",0a6h,"39.6",0a7h,"39.7"
db    0a8h,"39.8",0a9h,"39.9",0aah,"40.0",0abh,"40.1"
db    0ach,"40.2",0adh,"40.3",0aeh,"40.4",0afh,"40.5"
db    0b0h,"40.6",0b1h,"40.7",0b2h,"40.8",0b3h,"40.9"
db    0b4h,"41.0",0b5h,"41.1",0b6h,"41.2",0b7h,"41.3"
db    0b8h,"41.4",0b9h,"41.5",0bah,"41.6",0bbh,"41.7"
db    0bch,"41.8",0bdh,"41.9",0beh,"42.0",0bfh,"42.1"
db    0c0h,"42.2",0c1h,"42.3",0c2h,"42.4",0c3h,"42.5"
db    0c4h,"42.6",0c5h,"42.7",0c6h,"42.8",0c7h,"42.9"
db    0c8h,"43.0",0c9h,"43.1",0cah,"43.2",0cbh,"43.3"
db    0cch,"43.4",0cdh,"43.5",0ceh,"43.6",0cfh,"43.7"
db    0d0h,"43.8",0d1h,"43.9",0d2h,"44.0",0d3h,"44.1"
db    0d4h,"44.2",0d5h,"44.3",0d6h,"44.4",0d7h,"44.5"
db    0d8h,"44.6",0d9h,"44.7",0dah,"44.8",0dbh,"44.9"
db    0dch,"45.0",0ddh,"45.1" 0deh,"45.2",0dfh,"45.3"
db    0e0h,"45.4",0e1h,"45.5",0e2h,"45.6",0e3h,"45.7"
db    0e4h,"45.8",0e5h,"45.9",0e6h,"46.0",0e7h,"46.1"
db    0e8h,"46.2",0e9h,"46.3",0eah,"46.4",0ebh,"46.5"
db    0ech,"46.6",0edh,"46.7",0efh,"46.8",0f0h,"46.9"
db    0f1h,"47.0",0f2h,"47.1",0f3h,"47.2",0f4h,"47.3"
db    0f5h,"47.4",0f6h,"47.5",0f7h,"47.6",0f8h,"47.7"
db    0f9h,"47.8",0fah,"47.9",0fbh,"48.0",0fch,"48.1"
db    0fdh,"48.2",0feh,"48.3",0ffh,"48.4"
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

อุณหภูมิของน้ำตั้งไว้ที่ 40 องศาเซลเซียส
อุณหภูมิของน้ำตอนเริ่มต้น 23 องศาเซลเซียส

เวลา (วินาที)	อุณหภูมิของน้ำที่วัดได้ (องศาเซลเซียส)	สถานะตามกฎพัชชี
0	23	8
15	36	8
30	38	6
45	41	3
60	40	2
75	38	4
90	39	2
105	39	1
120	41	2
135	39	1
150	41	3

ค่าอุณหภูมิโดยเฉลี่ยขณะทำการทดลอง 39.2 องศาเซลเซียส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปและวิจารณ์

หลังจากที่ได้ทำการทดลองและศึกษา Project นี้อยู่เป็นเวลานาน ทำให้สามารถสรุปได้ว่าเครื่องทำน้ำอุ่นที่ควบคุมด้วยระบบฟัซซี่ลอจิก ให้ผลการควบคุมอุณหภูมิเป็นที่น่าพอใจ เมื่อเทียบกับเครื่องทำน้ำอุ่นโดยทั่วไป โดยที่อุณหภูมิของเครื่องทำน้ำอุ่นที่ควบคุมด้วยระบบฟัซซี่นั้นจะมีอุณหภูมิค่อนข้างที่จะคงที่ เมื่อเทียบกับระบบ ที่ควบคุมอุณหภูมิน้ำด้วยวิธีอื่นๆ การเปลี่ยนแปลงของอุณหภูมิน้ำที่ควบคุมด้วยระบบ Fuzzy นี้จะเปลี่ยนแปลงไปค่อนข้างน้อย เมื่อปัจจัยที่มีผลต่อการเปลี่ยนแปลงอุณหภูมิน้ำเปลี่ยนไป

แต่ในการควบคุมอุณหภูมิของน้ำให้ได้ดีที่สุดนั้น ยังขึ้นอยู่กับปัจจัยอีกหลายอย่างดังที่ได้กล่าวไว้ข้างต้น เช่น แรงดันของน้ำและอุณหภูมิของน้ำที่เข้ามา ปัญหาที่พบในระหว่างการศึกษา Project นี้ที่ทำให้อุณหภูมิของน้ำไม่เป็นไปตามที่เราตั้งไว้คือ อุณหภูมิของน้ำจะขึ้นกับแรงดันของน้ำที่เข้ามายังเครื่องทำน้ำอุ่น ซึ่งเมื่อแรงดันน้ำไม่คงที่ จะทำให้การควบคุมอุณหภูมินั้นยาก อีกปัญหาหนึ่งที่พบคือ ถ้ามีแรงดันน้ำสูงจนเกินความสามารถของตัว Heater ที่จะทำให้น้ำร้อนได้ทันกับที่ต้องการ นอกจากนี้ยังพบว่าคุณภาพของตัว Heater ซึ่งชำรุดง่ายและมีราคาค่อนข้างแพงจึงทำให้ทางผู้จัดทำไม่สามารถดัดแปลงค่าที่ใช้ในการควบคุมต่างๆได้ตามชอบใจ เช่นระยะเวลาในการ ON/OFF ของตัว Heater

ถ้าต้องการให้การควบคุมอุณหภูมิมีประสิทธิภาพมากกว่านี้ จะต้องมีอุปกรณ์ช่วยรักษาระดับแรงดันน้ำให้สม่ำเสมอเช่น มีถังพักน้ำและปั๊มน้ำ รวมทั้งมี Heater ที่มีความทนทานสามารถให้ความร้อนได้สูงอย่างรวดเร็ว ทันกับปริมาณน้ำที่เปลี่ยนแปลงไป นอกจากนั้นเรายังสามารถปรับปรุงกราฟความเป็นสมาชิก (Membership Graph) ซึ่งเป็นตัวกำหนดเอาท์พุตที่ออกจากวงจรควบคุม ให้เหมาะสมกับปัจจัยต่างๆก็จะทำให้การควบคุมอุณหภูมิของน้ำมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

เป็นภาระหน้าที่อย่างแท้จริงของวิศวกรที่จะต้องนำเอาความรู้ทางวิทยาศาสตร์มาประยุกต์ เพื่อนำมาใช้งานได้จริง คณะผู้จัดซึ่งตระหนักถึงความรับผิดชอบข้อนี้ จึงมีความตั้งใจเป็นอย่างสูงอีกทั้งใช้แรงกายและแรงใจเป็นอย่างมาก ประกอบกับความร่วมมือจากบุคคลหลายฝ่ายจนคณะผู้จัดทำอดที่จะนำมากล่าวถึง ณ. ที่นี้ไม่ได้ บุคคลเหล่านี้คือ

ผศ.ดร. กิตติ ให้คำปรึกษาทางทฤษฎี Fuzzy Theory

อาจารย์ ขนิษฐา ให้คำแนะนำตลอดจนที่หลับนอน

ผศ. พลผดุง ให้คำแนะนำและคำติชม ตลอดจนเทคนิคต่างๆ

พี่ภพ พี่เซนต์ พี่เล็ก และพี่รักษ์ ความรู้และความเมตตาของที่พวกพี่ๆมอบให้จะยังเป็นประโยชน์ต่อน้องๆและชาวโลกอีกมากมาย

ขอขอบคุณเจ้าหน้าที่ธุรการภาคและบุคคลที่เกี่ยวข้องทุกท่าน และขอขอบคุณอีกครั้งสำหรับทุกสิ่งทุกอย่างที่ทำให้ Project นี้ลุล่วงไปได้

จากใจ คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. Philips Company Limited, "Signetics Microcontroller Users' Guide".

2. Greg Viot, "Concepts to Constructs", AI EXPERT, November, 1993, pp. 26-33.

3. Keneth J. Ayala, "The 8051 Microcontroller Architecture, Programming, and Application", West Publishing Company Limited.

4. Lotfi A. Zadeh, "The Calculus of Fuzzy If/Then Rules", AI EXPERT, March, 1992, pp. 23-27.

5. Y. F. Li and C. C. Lau, "Development of Fuzzy Algorithm for Servo Systems", IEEE Control System Magazine, Vol. 9, No. 6, pp. 65-72, 1989.

6. Rodger Knaus, "Piecewise Linear Fuzzy Sets", AI EXPERT, March, 1992, pp. 16-19.

7. Omron Electronics Corporations, "Clearly Fuzzy".

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้