บทที่ 4
ขั้นตอนและผลการทดสอบการทำงาน

4.1 การทำแผงวงจรควบคุม

ในส่วนของแผงวงจรควบคุม จะประกอบด้วยอุปกรณ์ย่อยทั้งหมด 22 จำนวน และการออกแบบลายวงจรบนแผ่นวงจรพิมพ์ที่จะทำเป็นแผงวงจรควบคุม จะใช้โปรแกรม Altium Designer 6 ในการออกแบบ ซึ่งมีรายละเอียดดังต่อไปนี้

4.1.1 อุปกรณ์ย่อยต่างๆที่ใช้บนแผงวงจรควบคุม

ตารางที่ 4.1 รายการอุปกรณ์ย่อยต่างๆที่ใช้บนแผงวงจรควบคุม

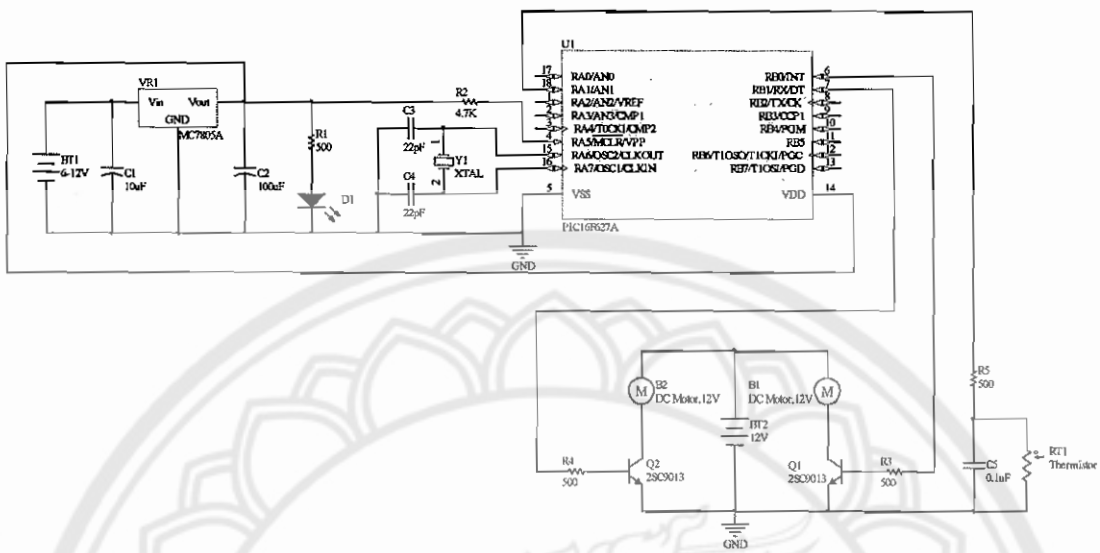
รายการอุปกรณ์	จำนวน
1. ไอซีซ็อคเก็ต (IC Socket) 18 ขา (U1*)	1
2. ไมโครคอนโทรลเลอร์ตระกูล PIC เบอร์ PIC16F627A (U1)	1
3. ไอซีเร็กกูเลเตอร์ 5 V เบอร์ MC7805A (VR1)	1
4. ตัวต้านทาน 1/4 W ขนาด 500 Ω (R1, R3, R4, R5)	4
5. ตัวต้านทาน 1/4 W ขนาด 4.7 kΩ (R2)	1
6. ตัวเก็บประจุแบบอิเล็กทรอไลต์ ขนาด 10 μF (C1)	1
7. ตัวเก็นประจุแบบอิเล็กทรอไลต์ ขนาด 100 μF (C2)	1
8. ตัวเก็บประจุแบบเซรามิค ขนาด 22 pF (C3, C4)	2
9. ตัวเก็บประจุแบบเซรามิค ขนาด 0.1 μF (C5)	1
10. เทอร์มิสเตอร์ เบอร์ TTC-104 (RT1)	1
11. คริสตอล (X-TAL) 4.0 MHz (Y1)	1
12. แอลอีดี (LED) สีเขียว ขนาดเส้นผ่านศูนย์กลาง 3 mm (D1)	1
13. ทรานซิสเตอร์แบบ NPN เบอร์ 2SC9013 (Q1, Q2)	2
14. เทอร์มินอลบล็อก (Terminal block) (BT1, BT2, B1, B2)	4

หมายเหตุ : ไอซีซ็อคเก็ต (U1*) คืออุปกรณ์สำหรับใส่หรือยึดตัว PIC เอาไว้บนแผงวงจรควบคุม

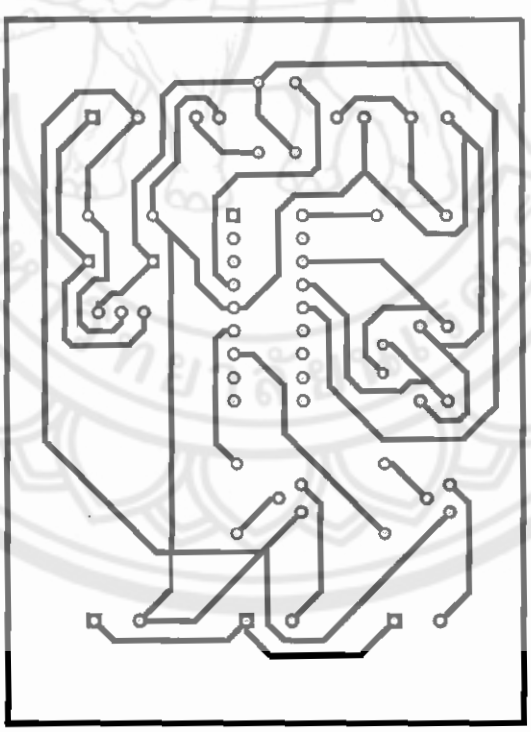
4.1.2 การออกแบบลายวงจรบนแผ่นวงจรพิมพ์

การออกแบบลายวงจรบนแผ่นวงจรพิมพ์จะใช้โปรแกรม Altium Designer 6 ในการออกแบบ ซึ่งในที่นี้จะให้โปรแกรมทำการออกแบบลายวงจรให้โดยอัตโนมัติ แต่อันดับแรกก่อนการออกแบบลายวงจรดังกล่าว จะต้องทำการออกแบบหรือเขียนวงจรไฟฟ้าของแผงวงจรควบคุม ซึ่งจะออกแบบวงจรได้ดังรูปที่ 4.1 เพื่อให้โปรแกรมตรวจสอบและจำลองการทำงานของวงจร ถ้า

วงจรไม่มีข้อผิดพลาด โปรแกรมก็จะทำการออกแบบลายวงจรโดยอัตโนมัติให้ทันที ซึ่งจะได้ลายวงจรดังรูปที่ 4.2



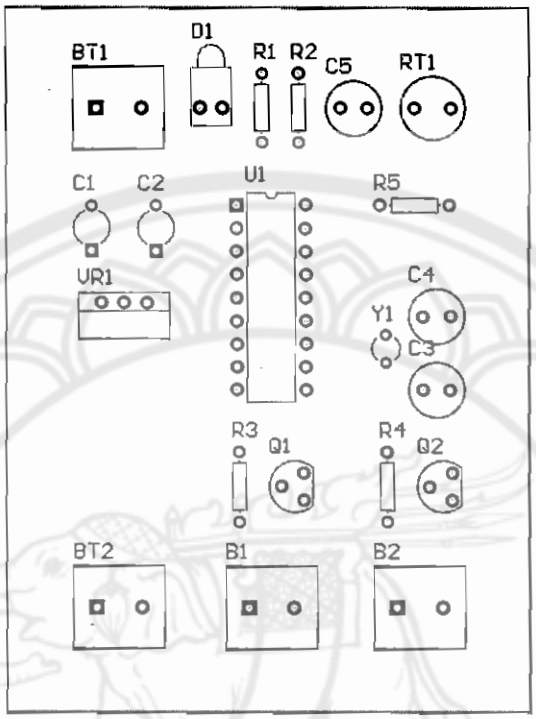
รูปที่ 4.1 วงจรที่ออกแบบได้เพื่อเตรียมที่จะสร้างลายวงจร



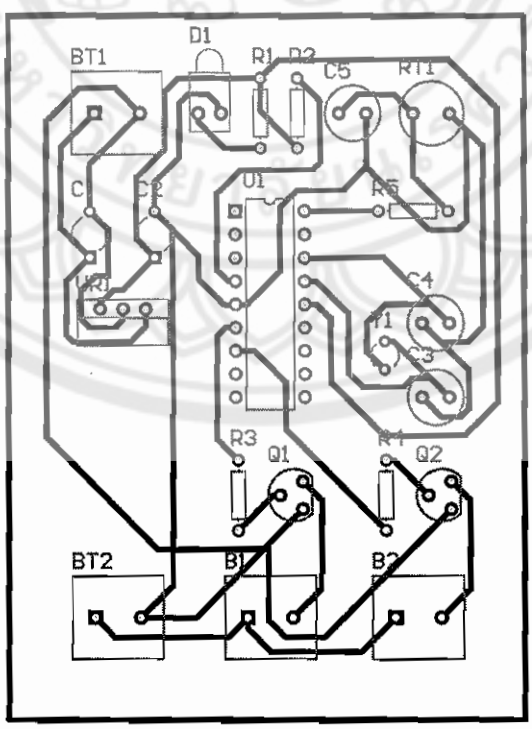
รูปที่ 4.2 ลายวงจรที่ได้จากโปรแกรม Altium Designer 6

จากลายวงจรดังรูปที่ 4.2 จะมีขนาดกรอบสี่เหลี่ยมกว้าง 57.4 mm และยาว 77.4 mm ซึ่งเป็นขนาดจริงจากการที่โปรแกรมกำหนดค่าให้ เมื่อเราใส่ค่าขนาดกว้างและยาวลงไปในโปรแกรมเป็น

60 mm และ 80 mm ตามลำดับ ซึ่งเป็นขนาดจริงของแผงวงจรควบคุมที่เราจะทำ เนื่องจากพื้นที่ของลายวงจรจะต้องมีขอบเขตน้อยกว่าพื้นที่ของแผ่นวงจรพิมพ์หรือแผงวงจรควบคุม สำหรับตำแหน่งต่างๆของอุปกรณ์ย่อยแต่ละตัวจะแสดงได้ดังรูปที่ 4.3 หรือ 4.4



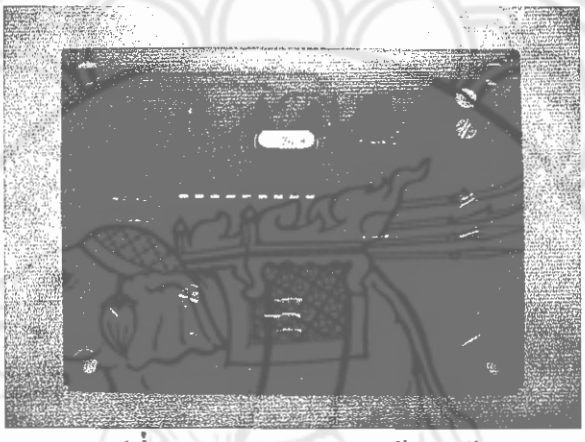
รูปที่ 4.3 ตำแหน่งของอุปกรณ์ย่อยแต่ละตัวบนแผงวงจรควบคุม



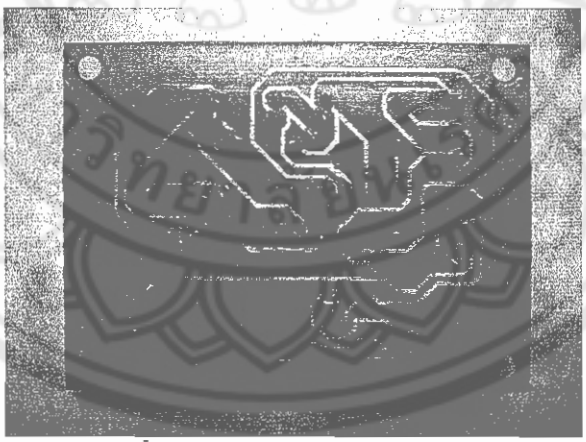
รูปที่ 4.4 ลายวงจรและตำแหน่งของอุปกรณ์ย่อยแต่ละตัวบนแผงวงจรควบคุม

เมื่อมองรูปที่ 4.4 เสมือนกับว่ากระดาษแผ่นนี้เป็นแผงวงจรควบคุมจริงๆ สำหรับลายวงจรที่เห็นนั้น จะอยู่ด้านหลังของกระดาษ ส่วนอุปกรณ์ย่อยต่างๆนั้น จะอยู่ด้านหน้าของกระดาษ เนื่องจากลายวงจรที่ได้จากโปรแกรมนี้ เราจะต้องนำไปปริ้นบนแผ่นใส แล้วนำมาเข้าสู่กระบวนการสร้างลายวงจรหรือลายทองแดงบนแผ่นวงจรพิมพ์ต่อไป

เมื่อสร้างลายวงจรหรือลายทองแดงบนแผ่นวงจรพิมพ์แล้ว ก็จะทำการเจาะรูที่ตำแหน่งขาต่างๆของอุปกรณ์ย่อยด้วยสว่านขนาดเล็ก จากนั้นจึงทำการบัดกรีเพื่อให้อุปกรณ์ย่อยแต่ละตัวเชื่อมต่อกับแผ่นวงจรพิมพ์ตามลายวงจรบนแผ่นวงจรพิมพ์นั้น เมื่อบัดกรีเสร็จเรียบร้อยแล้วก็จะได้แผงวงจรควบคุมออกมาดังรูปที่ 4.5 และ 4.6



รูปที่ 4.5 แผงวงจรควบคุมด้านหน้า



รูปที่ 4.6 แผงวงจรควบคุมด้านหลัง

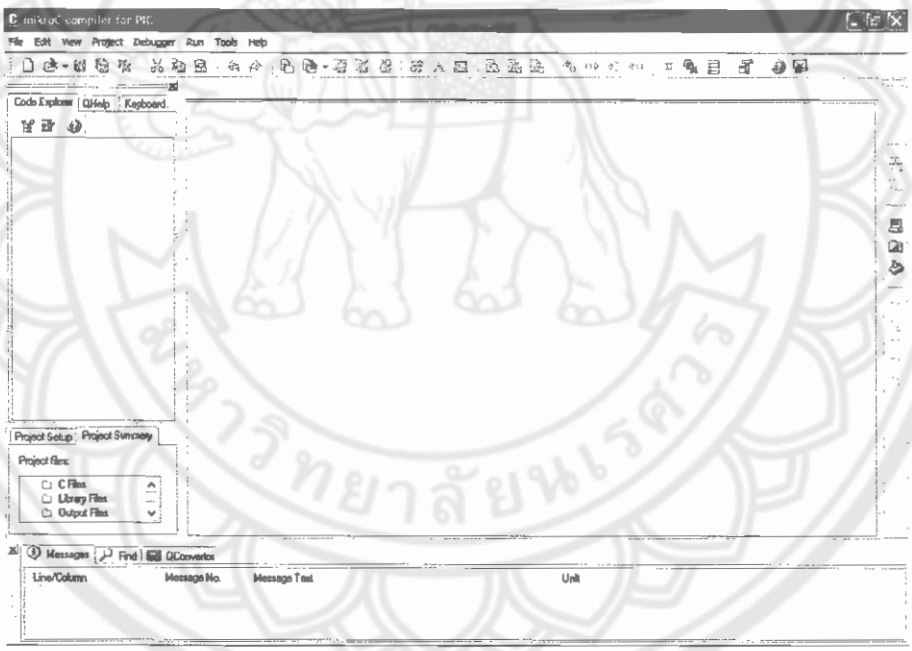
4.2 การเขียนโปรแกรมภาษาซีและการป้อนโปรแกรมเข้าสู่ตัว PIC

การเขียนโปรแกรมภาษาซีนั้น จะใช้โปรแกรม mikroC ในการเขียน เมื่อเขียนโปรแกรมเสร็จแล้ว ก็จะทำการตรวจสอบหรือคอมไพล์ (Compile) โปรแกรมที่เราเขียนไว้ว่าถูกต้องหรือไม่ ถ้าโปรแกรมถูกต้องหรือไม่มีข้อผิดพลาด การตรวจสอบโปรแกรมก็จะผ่าน และโปรแกรมจะสร้างไฟล์ตัวหนึ่งขึ้นมา เพื่อที่จะใช้ไฟล์นี้ในการโหลดหรือป้อนเข้าไปให้กับ PIC ไฟล์นี้จะเป็นไฟล์ที่มีนามสกุลเป็น ".hex" เมื่อได้ไฟล์นี้จากโปรแกรม mikroC แล้ว ก็จะเข้าสู่กระบวนการป้อนไฟล์ที่เป็นเหมือนตัวแทนของโปรแกรมนี้เข้าสู่ตัว PIC ต่อไป ซึ่งการป้อนไฟล์เข้าสู่ตัว PIC นี้จะใช้โปรแกรม WINPIC และใช้อุปกรณ์เพิ่มเติมคือ เครื่องโปรแกรมไมโครคอนโทรลเลอร์ ซึ่งรายละเอียดต่างๆดังกล่าว จะอธิบายได้ดังต่อไปนี้

4.2.1 การเขียนโปรแกรมภาษาซี

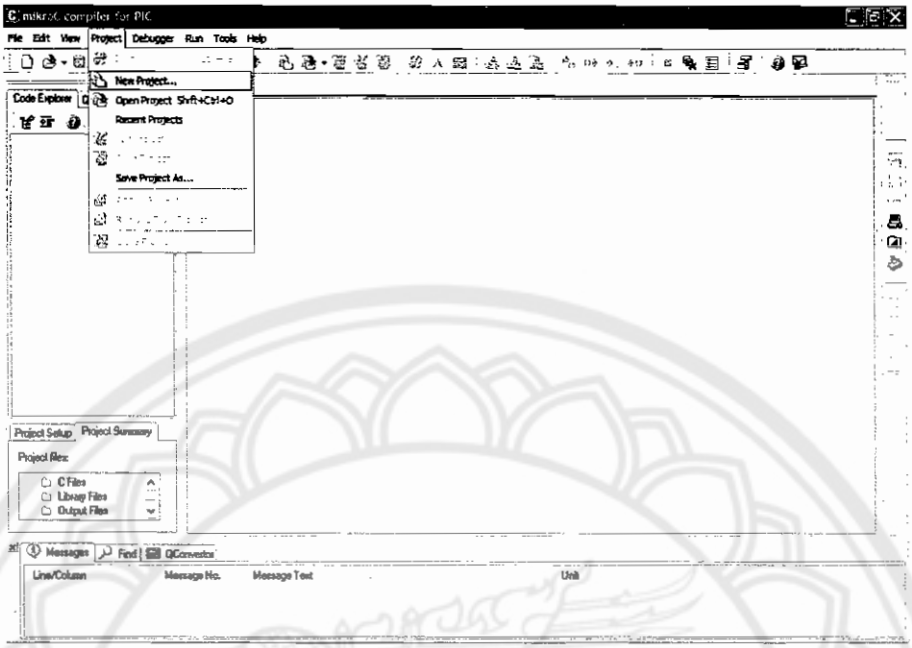
การเขียนโปรแกรมภาษาซีโดยใช้โปรแกรม mikroC จะมีขั้นตอนดังนี้

1. เปิดโปรแกรม mikroC ขึ้นมา จะปรากฏหน้าต่างแรกของโปรแกรมดังรูปที่ 4.7



รูปที่ 4.7 รูปประกอบขั้นตอนที่ 1 ของการเขียนโปรแกรมภาษาซี

2. คลิกที่เมนู Project > New Project เพื่อทำการสร้างโปรเจคใหม่



รูปที่ 4.8 รูปประกอบขั้นตอนที่ 2 ของการเขียนโปรแกรมภาษาซี

3. จากนั้นจะปรากฏหน้าต่าง New Project ขึ้นมา แล้วทำการตั้งค่าดังนี้

ที่ Project Name ตั้งชื่อเป็น My_Project

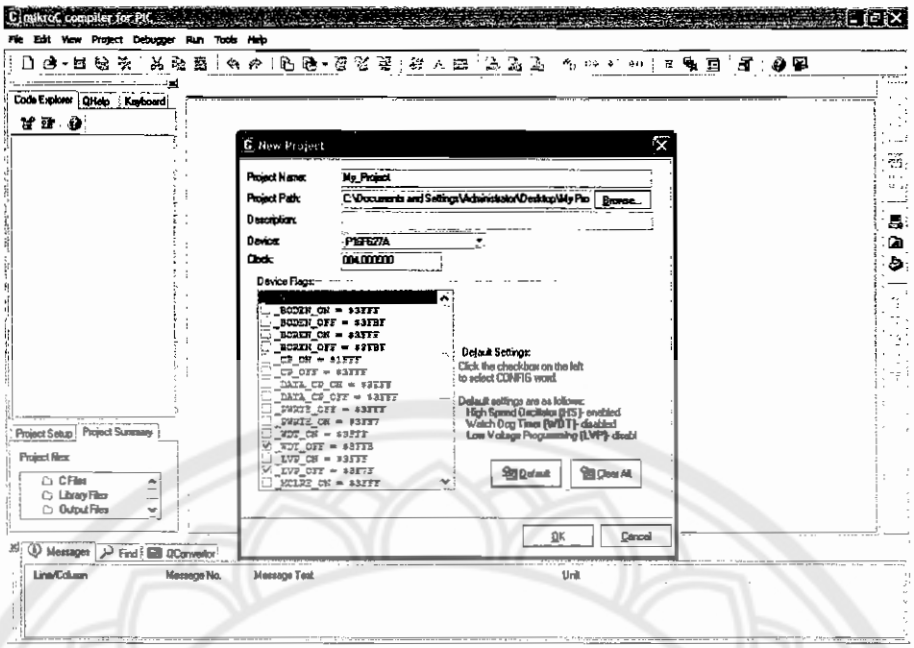
ที่ Project Path เก็บไฟล์โปรเจคไว้ที่ตำแหน่งโฟลเดอร์ My Project ที่เราสร้างขึ้นมาไว้แล้ว

ที่ Description เว้นว่างเอาไว้

ที่ Device เลือกไมโครคอนโทรลเลอร์ PIC เบอร์ PIC16F627A

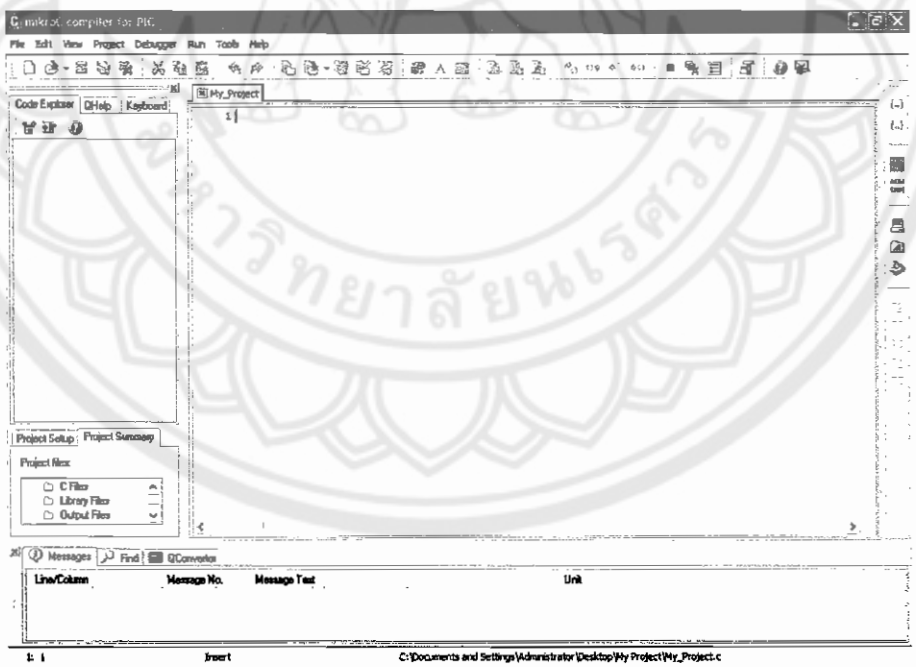
ที่ Clock กำหนดความถี่ของสัญญาณนาฬิกาเป็น 4.00 MHz

เสร็จแล้วคลิกที่ Default แล้วคลิกที่ OK



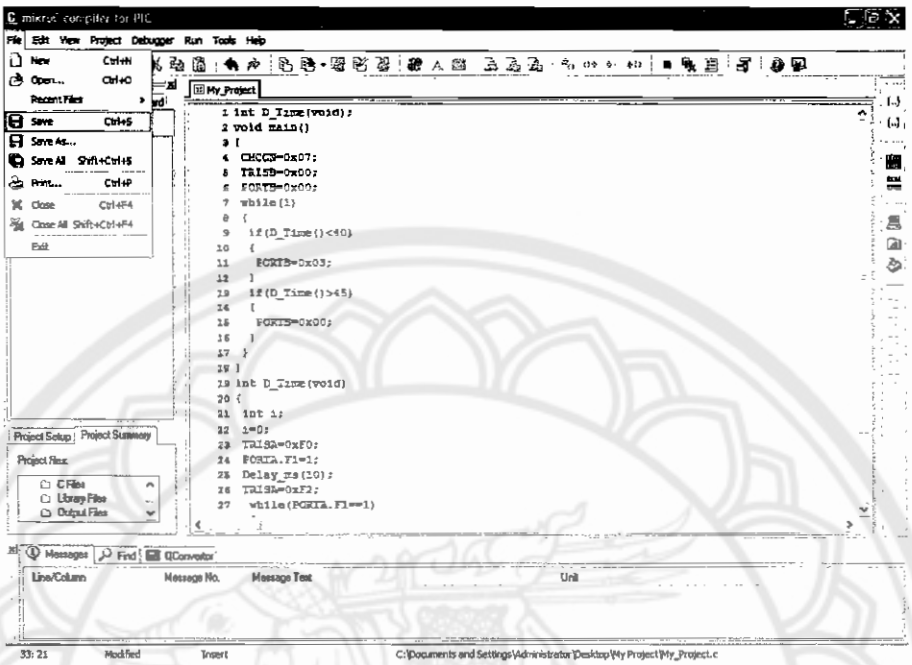
รูปที่ 4.9 รูปประกอบขั้นตอนที่ 3 ของการเขียนโปรแกรมภาษาซี

4. จากนั้นก็จะเข้าสู่หน้าต่างแรกของการเขียนโปรแกรมภาษาซี แล้วเขียนโปรแกรมภาษาซีจนเสร็จ



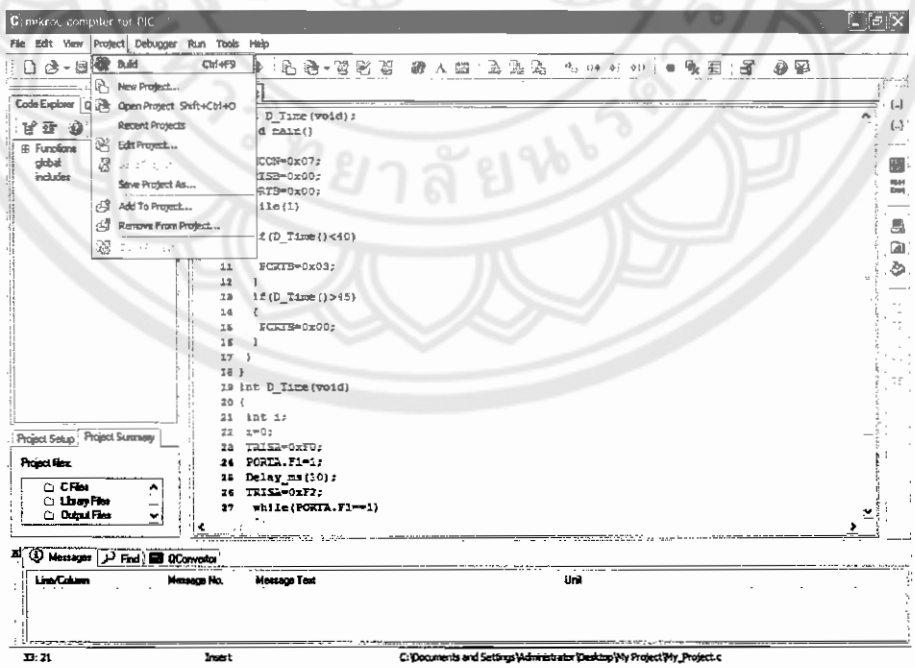
รูปที่ 4.10 รูปประกอบขั้นตอนที่ 4 ของการเขียนโปรแกรมภาษาซี

5. เมื่อเขียนโปรแกรมภาษาซีเสร็จแล้ว คลิกที่เมนู File > Save โปรแกรม mikroC ก็จะทำการเก็บบันทึกโปรแกรมภาษาซีที่เราเขียนไว้ที่ตำแหน่งโฟลเดอร์ My Project โดยอัตโนมัติ



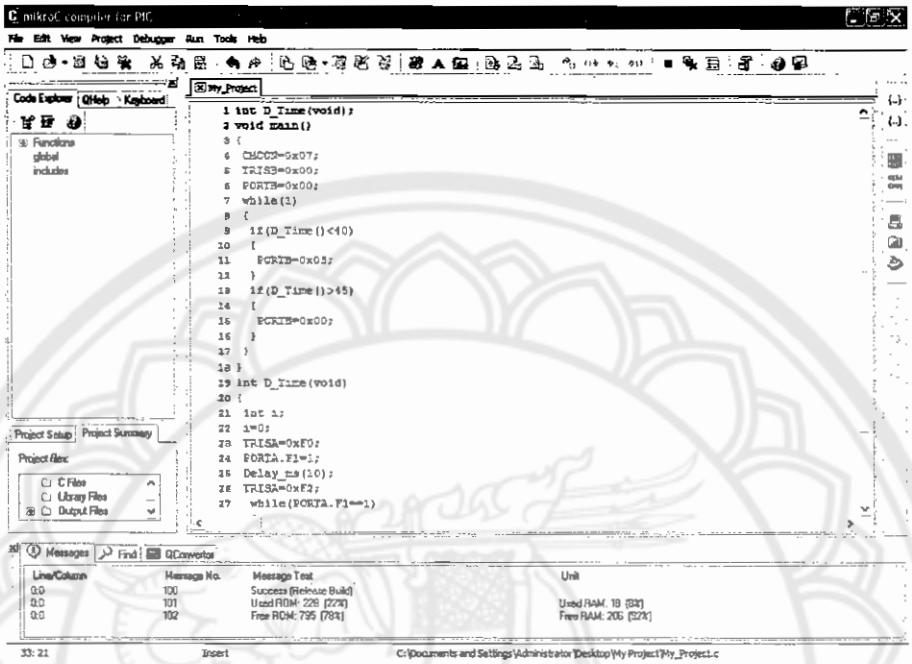
รูปที่ 4.11 รูปประกอบขั้นตอนที่ 5 ของการเขียนโปรแกรมภาษาซี

6. คลิกที่เมนู Project > Build เพื่อทำการคอมไพล์โปรแกรมภาษาซีที่เราเขียนไว้



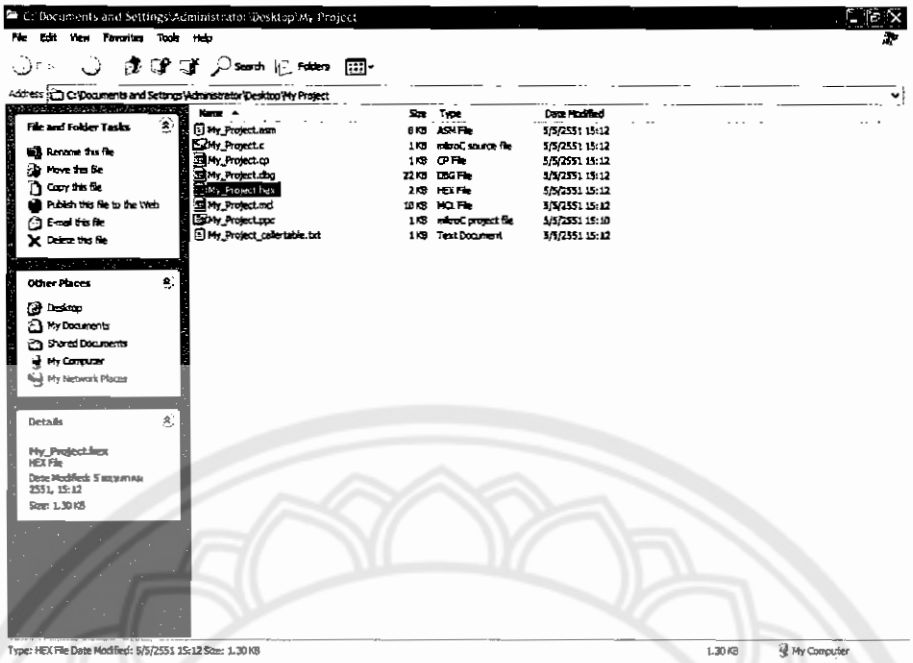
รูปที่ 4.12 รูปประกอบขั้นตอนที่ 6 ของการเขียนโปรแกรมภาษาซี

7. จากนั้นจะปรากฏข้อความที่หน้าต่าง Message ด้านล่างว่า Success นั่นคือ โปรแกรมภาษาซีที่เราเขียนไว้ถูกต้อง ไม่มีข้อผิดพลาด และโปรแกรม mikroC ก็จะทำการสร้างไฟล์ที่มีนามสกุลเป็น ".hex" ขึ้นมาจากโปรแกรมภาษาซีที่เราเขียนไว้



รูปที่ 4.13 รูปประกอบขั้นตอนที่ 7 ของการเขียนโปรแกรมภาษาซี

8. ไฟล์ที่โปรแกรม mikroC สร้างขึ้นมาในที่นี้ก็คือไฟล์ My Project.hex ซึ่งจะถูกสร้างเก็บไว้ที่โฟลเดอร์ My Project ดังรูปที่ 4.14



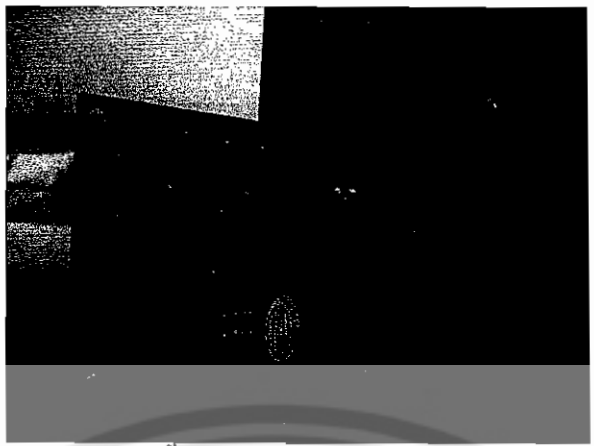
รูปที่ 4.14 รูปประกอบขั้นตอนที่ 8 ของการเขียนโปรแกรมภาษาซี

4.2.2 การป้อนโปรแกรมเข้าสู่ตัว PIC

การป้อนโปรแกรมเข้าสู่ตัว PIC โดยใช้โปรแกรม WINPIC จะมีขั้นตอนดังนี้

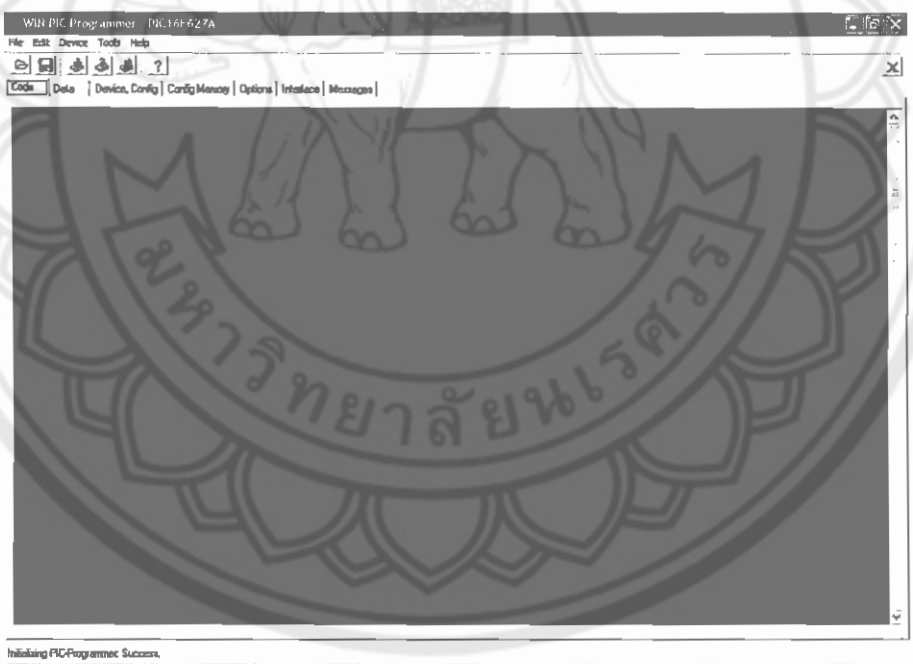
1. นำเครื่องตัว PIC มาเสียบหรือเชื่อมต่อกับเครื่องโปรแกรมไมโครคอน โทรลเลอร์ดังรูปที่ 4.15 แล้วนำไปเชื่อมต่อกับพอร์ตอนุกรม หรือ COM PORT ของคอมพิวเตอร์ดังรูปที่ 4.16

รูปที่ 4.15 รูปประกอบขั้นตอนที่ 1 ของการป้อนโปรแกรมเข้าสู่ตัว PIC



รูปที่ 4.16 รูปประกอบขั้นตอนที่ 1 ของการป้อนโปรแกรมเข้าสู่ตัว PIC

2. เปิดโปรแกรม WINPIC ขึ้นมา จะปรากฏหน้าต่างแรกของโปรแกรมดังรูปที่ 4.17 จะสังเกตได้ว่าด้านล่างของหน้าต่างโปรแกรมจะปรากฏข้อความว่า "Initialising PIC-Programmer: Success" ซึ่งหมายความว่า เครื่องโปรแกรมไมโครคอนโทรลเลอร์ได้ถูกเชื่อมต่อกับคอมพิวเตอร์อย่างถูกต้องแล้ว

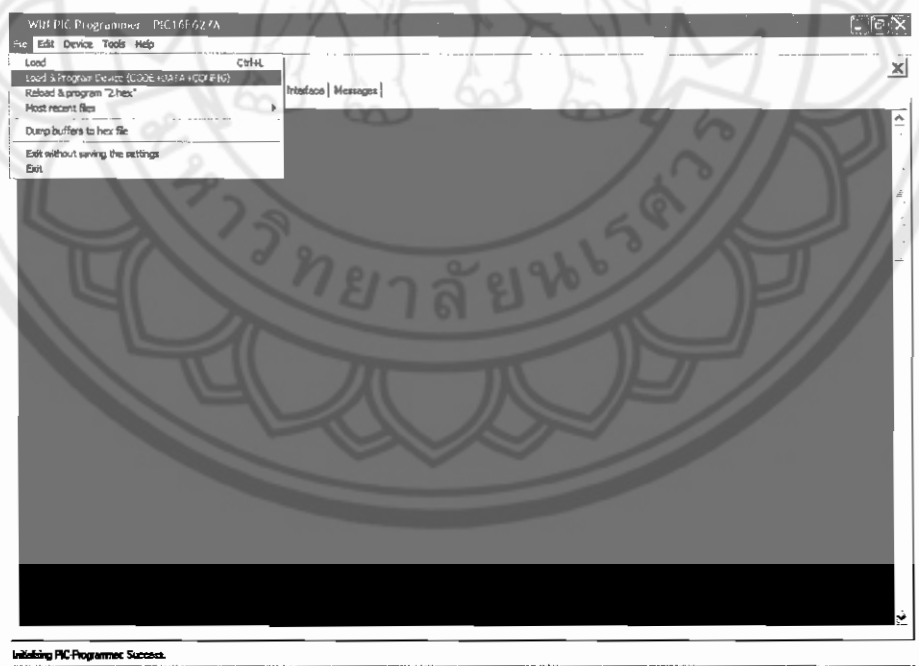


รูปที่ 4.17 รูปประกอบขั้นตอนที่ 2 ของการป้อนโปรแกรมเข้าสู่ตัว PIC

3. แต่ก่อนหน้าขั้นตอนตั้งแต่แรก เราจะทำการตั้งค่าเบอร์ของ PIC ที่จะใช้ในการป้อนโปรแกรม ซึ่งเมื่อเข้าโปรแกรม WINPIC แล้ว คลิกเลือกที่ Device, Config ที่ Part ให้เลือกเบอร์ของ PIC เป็น PIC16F627A

รูปที่ 4.18 รูปประกอบขั้นตอนที่ 3 ของการป้อนโปรแกรมเข้าสู่ตัว PIC

4. คลิกที่เมนู File > Load & Program Device (CODE+DATA+CONFIG) เพื่อเตรียมทำการโหลดโปรแกรมหรือไฟล์ My Project.hex เข้าสู่ตัว PIC

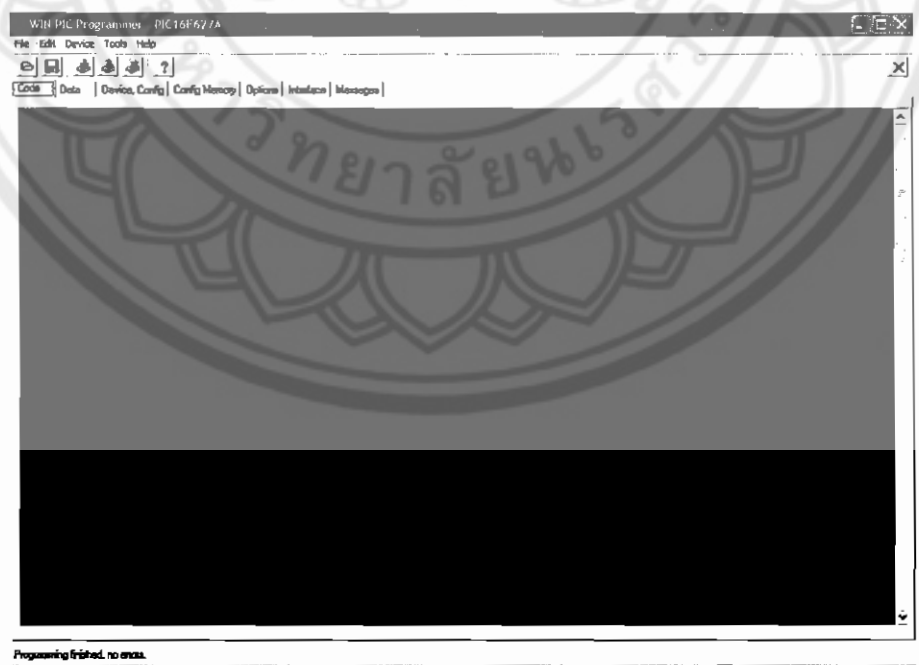


รูปที่ 4.19 รูปประกอบขั้นตอนที่ 4 ของการป้อนโปรแกรมเข้าสู่ตัว PIC

5. จากนั้นจะปรากฏหน้าต่าง Load Hex File ขึ้นมา แล้วไปที่ตำแหน่งโฟลเดอร์ My Project เพื่อทำการเลือกไฟล์ My Project.hex แล้วคลิกที่ Open เพื่อเริ่มทำการป้อนไฟล์เข้าสู่ตัว PIC

รูปที่ 4.20 รูปประกอบขั้นตอนที่ 5 ของการป้อนโปรแกรมเข้าสู่ตัว PIC

6. เมื่อเสร็จแล้วจะปรากฏข้อความด้านล่างว่า Initialising PIC-Programmer: Success.

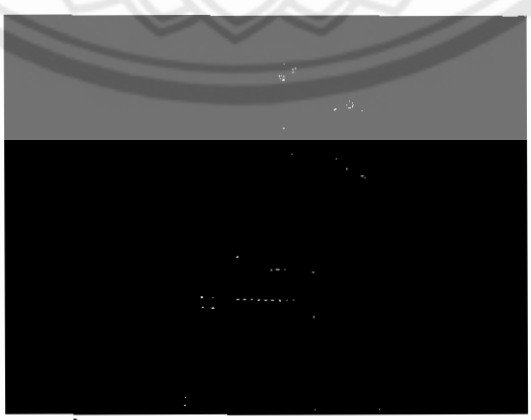


รูปที่ 4.21 รูปประกอบขั้นตอนที่ 6 ของการป้อนโปรแกรมเข้าสู่ตัว PIC

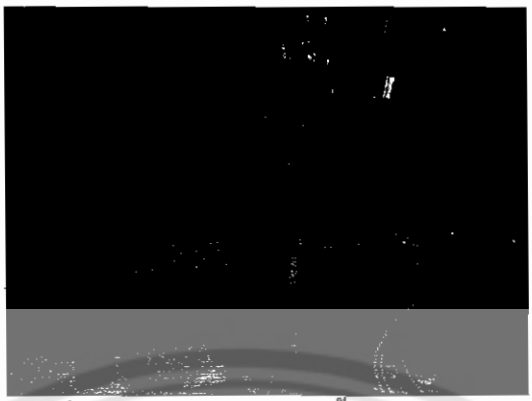
7. จากนั้น ทำการถอดเครื่องโปรแกรมไมโครคอนโทรลเลอร์ออกจากพอร์ตอนุกรมของคอมพิวเตอร์ แล้วดึงตัว PIC ออกจากเครื่องโปรแกรมไมโครคอนโทรลเลอร์ แล้วนำตัว PIC นี้ไปเสียบหรือเชื่อมต่อกับไอซีซ็อคเก็ต บนแผงวงจรควบคุม เพื่อเตรียมนำไปทดสอบการทำงานต่อไป

4.3 ผลการทดสอบการทำงานของแผงวงจรควบคุม

เนื่องจากก่อนที่จะมีการทำแผงวงจรควบคุม ได้มีการทดสอบการทำงานของวงจรบนโปรโตบอร์ด (Protoboard) แล้วว่า วงจรมีการทำงานเป็นไปตามที่เรากำหนดไว้ นั่นคือ เมื่ออุณหภูมิแวดล้อมสูงขึ้นถึงจุดๆหนึ่งตามที่เราได้ตั้งค่าไว้ในโปรแกรม มอเตอร์ทั้ง 2 ตัวก็จะทำงาน และเมื่ออุณหภูมิแวดล้อมลดลงถึงจุดๆหนึ่งตามที่เราได้ตั้งค่าไว้ในโปรแกรม มอเตอร์ทั้ง 2 ตัวก็จะหยุดทำงาน ซึ่งเราจะใช้ไดร์เป่าผมเป็นตัวสร้างอากาศร้อนขึ้นมาเพื่อใช้ทดสอบการทำงานของแผงวงจรควบคุม และเมื่อมีการทำแผงวงจรควบคุมเสร็จแล้ว ก็จะทำการทดสอบการทำงานของแผงวงจรควบคุมโดยการนำแผงวงจรควบคุมไปยึดกับ DC motor ขนาด 12 V ตัวใดตัวหนึ่งที่เป็นพัดลมระบายความร้อนของคอมพิวเตอร์ดังรูปที่ 4.22 แล้วทำการเชื่อมต่อวงจรคือ ที่เทอร์มินอลบล็อก BT_1 จะใช้สายไฟเพื่อเชื่อมต่อกับแบตเตอร์รี่ขนาด 6 V ที่เทอร์มินอลบล็อก BT_2 จะใช้สายไฟเพื่อเชื่อมต่อกับแบตเตอร์รี่ขนาด 12 V ที่เทอร์มินอลบล็อก B_1 จะทำการเชื่อมต่อกับมอเตอร์ตัวหนึ่ง ซึ่งในที่นี้จะให้เป็นมอเตอร์ที่ถูกยึดติดอยู่กับแผงวงจรควบคุมตามรูปที่ 4.22 และที่เทอร์มินอลบล็อก B_2 จะทำการเชื่อมต่อกับมอเตอร์อีกตัวหนึ่ง ดังรูปที่ 4.23 เมื่อทำการเชื่อมต่อวงจรเสร็จแล้ว ก็จะเริ่มทำการทดสอบการทำงานของแผงวงจรควบคุมโดยการเปิดสวิตช์ของแบตเตอร์รี่ทั้ง 2 ส่วนนั่นคือ BT_1และ BT_2 จากนั้นก็จะทำการสร้างอากาศร้อนโดยใช้ไดร์เป่าผม ซึ่งจะได้ว่าผลการทดสอบเหมือนเดิมนั่นคือ เมื่ออุณหภูมิแวดล้อมสูงขึ้นถึงจุดๆหนึ่งตามที่เราได้ตั้งค่าไว้ในโปรแกรม มอเตอร์ทั้ง 2 ตัวก็จะทำงาน และเมื่ออุณหภูมิแวดล้อมลดลงถึงจุดๆหนึ่งตามที่เราได้ตั้งค่าไว้ในโปรแกรม มอเตอร์ทั้ง 2 ตัวก็จะหยุดทำงาน ซึ่งในการทดสอบจะมีการทดสอบทั้งหมด 4 สภาวะย่อยๆด้วยกันในเวลา 1 วันหรือ 24 ชั่วโมง รายละเอียดและผลการทดสอบจะแสดงดังตารางที่ 4.2



รูปที่ 4.22 การเชื่อมต่อระหว่างมอเตอร์ตัวแรกกับแผงวงจรควบคุม



รูปที่ 4.23 การเชื่อมต่อระหว่างมอเตอร์ทั้ง 2 ตัวกับแผงวงจรควบคุม

4.4 ผลการประหยัดพลังงาน

การทดสอบการทำงานของแผงวงจรควบคุม จะมีการทดสอบอยู่ทั้งหมด 4 สภาวะ สภาวะแรกคือ สภาวะปกติ นั่นคือเป็นสภาวะที่มีการทำงานของมอเตอร์ทั้ง 2 ตัวอย่างต่อเนื่องตลอด 24 ชั่วโมง ส่วนอีกสามสภาวะนั้นจะเป็นสภาวะที่มีการใช้งานแผงวงจรควบคุม โดยที่แต่ละสภาวะจะแบ่งตามช่วงอุณหภูมิหรือช่วงของค่าตัวแปรในโปรแกรมภาษาซีที่จะทำให้มอเตอร์ทั้ง 2 ตัวทำงานและหยุดทำงาน

สภาวะที่สองจะมีการตั้งค่าที่โปรแกรมภาษาซีเพื่อกำหนดให้ระบบควบคุมมอเตอร์ทั้ง 2 ตัวให้เริ่มทำงานเมื่ออุณหภูมิ (T_2) มากกว่า 35 °C และหยุดทำงานเมื่ออุณหภูมิ (T_1) น้อยกว่า 32 °C โดยการกำหนดค่าให้กับตัวแปร n_1 เป็น 44 และ n_2 เป็น 48 ตามลำดับ ซึ่งจะได้ผลการประหยัดพลังงานเท่ากับ 4.45 % สภาวะที่สามจะทำการกำหนดให้ระบบควบคุมมอเตอร์ทั้ง 2 ตัวให้เริ่มทำงานเมื่ออุณหภูมิ (T_2) มากกว่า 40 °C และหยุดทำงานเมื่ออุณหภูมิ (T_1) น้อยกว่า 32 °C โดยการกำหนดค่าให้กับตัวแปร n_1 เป็น 40 และ n_2 เป็น 48 ตามลำดับ ซึ่งจะได้ผลการประหยัดพลังงานเท่ากับ 24.45 % และสภาวะที่สี่จะทำการกำหนดให้ระบบควบคุมมอเตอร์ทั้ง 2 ตัวให้เริ่มทำงานเมื่ออุณหภูมิ (T_2) มากกว่า 40 °C และหยุดทำงานเมื่ออุณหภูมิ (T_1) น้อยกว่า 35 °C โดยการกำหนดค่าให้กับตัวแปร n_1 เป็น 40 และ n_2 เป็น 45 ตามลำดับ ซึ่งจะได้ผลการประหยัดพลังงานสูงสุดเท่ากับ 36.11 % ดังตารางที่ 4.2

ตารางที่ 4.2 ผลการทดสอบการทำงาน

สภาวะปกติ		สภาวะที่มีการใช้งานแผงวงจรควบคุม			
พลังงานในรูป W-h ของมอเตอร์ทั้ง 2 ตัว (Wh_1, W-h)	3.6*		สภาวะที่ 1	สภาวะที่ 2	สภาวะที่ 3
		T_1 (°C)	32	32	35
		T_2 (°C)	35	40	40
พลังงานที่ใช้ในรูป W ของทั้งระบบใน 24 ชั่วโมง (W_1, W) $W_1=Wh_1\times24$	86.4	n_2	48	48	45
		n_1	44	40	40
		พลังงานในรูป W-h ของมอเตอร์ทั้ง 2 ตัว (Wh_1, W-h)	3.6*		
		พลังงานในรูป W-h ของ PIC (Wh_2, W-h)	0.8**		
		ระยะเวลาที่มอเตอร์ทั้ง 2 ตัวทำงานใน 1 ชั่วโมง (m, นาที)	44	32	25
		พลังงานที่ใช้ในรูป W ของทั้งระบบใน 24 ชั่วโมง (W_2, W) $W_2=[Wh_1\times(m/60)\times24]+(Wh_2\times24)$	82.56	65.28	55.20
		เปอร์เซ็นต์การประหยัดพลังงานเทียบกับสภาวะปกติ (%) $\%=[(W_1-W_2)/W_1]\times100$	4.45	24.45	36.11

หมายเหตุ :

1. "3.6*" เนื่องจาก DC motor 1 ตัวที่ใช้จะมีกำลัง 1.8 W ดังนั้นเมื่อใช้ 2 ตัว จะได้กำลังเป็น 3.6 W
2. "0.8**" ในที่นี้จะใช้กำลังวัตต์สูงสุดสำหรับการทำงานของ PIC16F627A นั่นคือ 0.8 W
3. ตัวแปร n_1 คือค่าจำนวนเต็มบวกที่จะต้องกำหนดลงไปในโปรแกรมภาษาซีในบรรทัดที่ 9 ตามโค้ดในหน้าที่ 41 บทที่ 3 เพื่อที่จะกำหนดช่วงอุณหภูมิ (T_2) ที่ทำให้มอเตอร์ทั้ง 2 ตัวทำงาน
4. ตัวแปร n_2 คือค่าจำนวนเต็มบวกที่จะต้องกำหนดลงไปในโปรแกรมภาษาซีในบรรทัดที่ 13 ตามโค้ดในหน้าที่ 41 บทที่ 3 เพื่อที่จะกำหนดช่วงอุณหภูมิ (T_1) ที่ทำให้มอเตอร์ทั้ง 2 ตัวหยุดทำงาน

4.5 วิเคราะห์ผลการทดสอบการทำงาน

เราสามารถวิเคราะห์และสรุปผลการทำงานของแผงวงจรควบคุมได้ว่า เมื่อทำการจ่ายไฟเลี้ยงให้กับ PIC และมอเตอร์ทั้ง 2 ตัวแล้ว แผงวงจรควบคุมจะทำหน้าที่ควบคุมการทำงานของมอเตอร์ทั้ง 2 ตัวนั้น นั่นคือ เมื่ออุณหภูมิแวดล้อมสูงขึ้นถึงจุดๆหนึ่งตามที่เราได้ตั้งค่าไว้ในโปรแกรมภาษาซี จุดนั้นกำหนดให้เป็นค่าจำนวนเต็มบวก n_1 มอเตอร์ทั้ง 2 ตัวก็จะทำงาน และเมื่ออุณหภูมิแวดล้อมลดลงถึงจุดๆหนึ่งตามที่เราได้ตั้งค่าไว้ในโปรแกรมภาษาซี จุดนั้นกำหนดให้เป็นค่าจำนวนเต็มบวก n_2 มอเตอร์ทั้ง 2 ตัวก็จะหยุดทำงาน โดยหลักการหลักการทำงานนี้จะอาศัยความสัมพันธ์ระหว่างค่าความต้านทานของเทอร์มิสเตอร์กับเวลาที่ใช้ในการคายประจุของตัวเก็บประจุ (C_5) ในวงจร ซึ่งจะเป็นช่วงเวลาที่สั้นมากๆ และในโปรแกรมภาษาซีจะมีการนับค่าเก็บไว้ที่ตัวแปรหนึ่งในขณะที่มีการคายประจุของตัวเก็บประจุดังกล่าว ดังนั้นความสัมพันธ์จะได้ว่า

ค่าตัวแปร $\propto$ เวลาที่ใช้ในการคายประจุ $\propto$ ค่าความต้านทานของเทอร์มิสเตอร์ $\propto$ 1/อุณหภูมิ

จากความสัมพันธ์ดังกล่าวจะอธิบายได้ว่า เมื่ออุณหภูมิแวดล้อมสูงขึ้น ค่าความต้านทานของเทอร์มิสเตอร์จะน้อยลง ทำให้เวลาที่ใช้ในการเก็บประจุน้อยลง ดังนั้นจึงทำให้ตัวแปรที่นับค่าได้มีค่าน้อยลงด้วย และเมื่ออุณหภูมิแวดล้อมต่ำลง ค่าความต้านทานของเทอร์มิสเตอร์จะมากขึ้น ทำให้เวลาที่ใช้ในการคายประจุมากขึ้น ดังนั้นจึงทำให้ตัวแปรที่นับค่าได้มีค่ามากขึ้นด้วย ค่าของตัวแปรดังกล่าวนี้จะถูกหารด้วย 10 แล้วเก็บไว้ที่ตัวแปรเดิมเพื่อให้มีค่าน้อยลง แล้วค่าของตัวแปรที่ได้ใหม่นี้ก็จะถูกนำไปกำหนดช่วงเวลาที่จะทำให้มอเตอร์ทั้ง 2 ตัวทำงานและหยุดทำงานในโปรแกรมภาษาซี นั่นคือถ้าตัวแปรดังกล่าวมีค่าน้อยกว่า n_1 มอเตอร์ทั้ง 2 ตัวก็จะทำงาน และถ้าตัวแปรดังกล่าวมีค่ามากกว่า n_2 มอเตอร์ทั้ง 2 ตัวก็จะหยุดทำงาน เมื่อ n_1 และ n_2 คือค่าของตัวแปรที่เราจะต้องกำหนดในโปรแกรมภาษาซี โดยที่ n_2 จะต้องมีค่ามากกว่า n_1 และนอกจากนี้เรายังสามารถเปลี่ยนค่า n_1 และ n_2 ได้เพื่อเปลี่ยนแปลงการทำงานของแผงวงจรควบคุม นั่นคือถ้าเราเพิ่มค่า n_1 มอเตอร์ทั้ง 2 ตัวก็จะเริ่มทำงานในระดับอุณหภูมิที่ต่ำลง เมื่อลดค่า n_1 มอเตอร์ทั้ง 2 ตัวก็จะเริ่มทำงานในระดับอุณหภูมิที่สูงขึ้น เมื่อเพิ่มค่า n_2 มอเตอร์ทั้ง 2 ตัวก็จะหยุดทำงานในระดับอุณหภูมิที่ต่ำลง และเมื่อลดค่า n_2 มอเตอร์ทั้ง 2 ตัวก็จะหยุดทำงานในระดับอุณหภูมิที่สูงขึ้น เมื่อนำแผงวงจรควบคุมนี้ไปใช้กับพัดลมระบายความร้อนของคอมพิวเตอร์ จะช่วยให้ประหยัดพลังงานไฟฟ้าได้มากขึ้น เนื่องจากพัดลมระบายความร้อนจะไม่ได้ทำงานอยู่ตลอดเวลา ถึงแม้ว่าจะต้องจ่ายไฟเลี้ยงให้กับ PIC ก็ตาม

จากตารางที่ 2 จะพบว่าผลการทดสอบในสภาวะที่สองจะมีการประหยัดพลังงาน 4.45 % สภาวะที่สามจะมีการประหยักพลังงาน 24.45 % และสภาวะที่สี่จะมีการประหยัดพลังงานสูงสุด 36.11 % ซึ่งพบว่าเมื่อมีการตั้งค่าอุณหภูมิ T_1 และ T_2 ให้ห่างกัน จะส่งผลให้มีการประหยัดพลังงานมากขึ้น เนื่องจากระยะเวลาในการทำงานของมอเตอร์จะต่างกัน