บทที่ 3
ปัญหาและวิธีการแก้ปัญหา

ในบทนี้กล่าวถึงการพัฒนาระบบเขียนโปรแกรมที่สามารถจับต้องและโต้ตอบได้ โรโบ-บล็อก โดยนำทฤษฎีในบทที่ 2 มาประยุกต์ใช้ นั่นคือ โรโบ-บล็อกที่ถูกพัฒนาขึ้นมานี้ เป็นเครื่องมือ Tangible[5] ซึ่งเป็นเครื่องมือแบบกายภาพ Physical เพื่อใช้ในการเขียนโปรแกรมควบคุมหุ่นยนต์ซึ่งเป็น Physical เช่นกัน โดยอุปกรณ์ที่พัฒนาขึ้นมานี้ ถูกนำมาใช้ในการเรียนรู้การเขียนโปรแกรม โดยเน้นที่การดีบักโปรแกรมของเด็ก

โรโบ-บล็อก ประกอบไปด้วยบล็อกคำสั่งซึ่งแต่ละบล็อกแทนคำสั่งเฉพาะแต่ละคำสั่ง โดยบล็อกคำสั่งนี้สร้างด้วยอะคริลิก ภายในมีวงจรอิเล็กทรอนิกส์ที่สามารถส่งสัญญาณหากันผ่านทางขั้วแม่เหล็กที่ติดอยู่ทั้งส่วนหัวและส่วนท้ายของบล็อก การใช้ขั้วแม่เหล็กทำให้เด็กสามารถต่อและถอดบล็อกได้สะดวก โดยชุดคำสั่งทั้งหมดได้แสดงไว้ในตารางที่ 3.1

ตารางที่ 3.1 ชุดคำสั่งทั้งหมดของระบบโรโบ-บล็อก

รายการบล็อก	**คำอธิบาย**
<u>บล็อกการเคลื่อนที่</u>	
เดินหน้า/ถอยหลัง	สั่งให้รถหุ่นยนต์เคลื่อนไปข้างหน้า/ถอยหลัง
หมุนซ้าย/หมุนขวา	สั่งให้รถหุ่นยนต์หมุนไปทางซ้าย/ไปทางขวา โดยที่รถไม่ได้เคลื่อนที่
<u>บล็อกควบคุมปากกา</u>	
จรด/ยกปากกา	วางปากกาเพื่อให้เกิดเส้นตามทางที่รถวิ่ง/ยกปากกา
<u>บล็อกโครงสร้าง</u>	
วนซ้ำ	บล็อกนี้มีสองชิ้นทำงานร่วมกัน โดยจะใช้ประกบครอบกลุ่มบล็อกคำสั่งเพื่อให้วนซ้ำการทำงานของกลุ่มบล็อกนั้นๆ ตามจำนวนครั้งที่กำหนด
<u>บล็อกอื่นๆ</u>	
ส่งเสียง	สั่งให้หุ่นส่งเสียง

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

เมื่อนำบล็อกคำสั่งเหล่านี้มาประกอบกันเป็นโปรแกรมและประกอบเข้าบล็อกมาสเตอร์ซึ่งเป็นหน่วยประมวลผลโปรแกรม จึงสามารถควบคุมรถหุ่นยนต์ให้สามารถทำงานตามโปรแกรมที่ต่อไว้ได้

3.1 โครงสร้างการทำงานของโรโบ-บล็อก

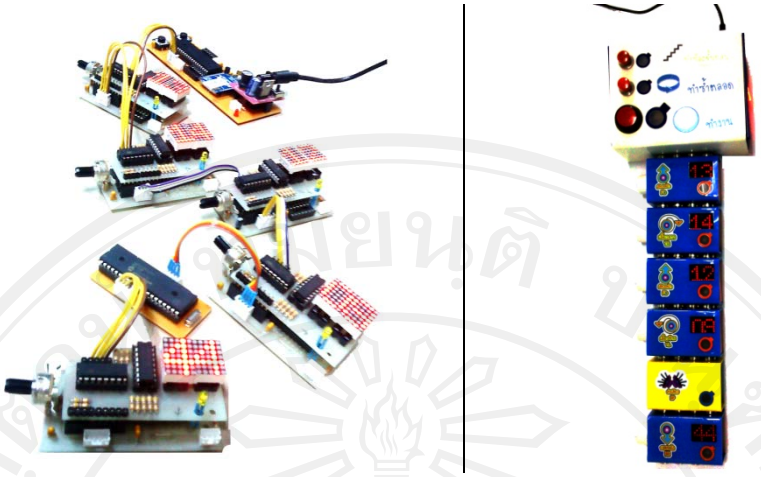
3.1.1 บล็อกคำสั่ง

ภายในบล็อกคำสั่งประกอบไปด้วยวงจรอิเล็กทรอนิกส์ซึ่งควบคุมด้วยไมโครคอนโทลเลอร์ตระกูล PIC รุ่น dsPIC30f4011 โดยทำหน้าที่จดจำคำสั่งของบล็อกนั้นๆ และติดต่อสื่อสารกับบล็อกที่อยู่รอบข้างผ่านทางช่องสัญญาณ RS-232 โดยการเชื่อมต่อระหว่างบล็อกจะมีสายทั้งหมด 4 เส้นทำหน้าที่เป็น (1) ไฟ +5V (2) ดิน (3) สัญญาณ Rx (4) สัญญาณ Tx โดยสายทั้ง 4 นี้เชื่อมโยงกันระหว่างบล็อกผ่านตัวสัมผัสกระดุมแม่เหล็ก บล็อกคำสั่งจะถูกแบ่งออกเป็น 2 กลุ่ม คือ (1) กลุ่มที่มีการทำงานตายตัว ได้แก่ บล็อกจรดปากกา ยกปากกาและส่งเสียง (2) กลุ่มที่สามารถปรับค่าพารามิเตอร์การทำงานได้ ซึ่งได้แก่ บล็อกการเคลื่อนที่ และบล็อกวนซ้ำ ซึ่งบล็อกประเภทนี้มีความพิเศษคือ จะมีปุ่มหมุนติดอยู่ด้านข้างเพื่อใช้ปรับค่าและมีหน้าจอแสดงผลชนิด Dot Matrix ขนาด 5x7 จุดจำนวน 2 ตัว

3.1.2 บล็อกมาสเตอร์

บล็อกนี้เป็นศูนย์กลางในการประมวลผลโปรแกรม โดยทำหน้าที่ตรวจสอบว่าบล็อกที่นำมาต่อกับบล็อกมาสเตอร์นั้นประกอบด้วยคำสั่งอะไรบ้าง ปุ่มคำสั่งต่างๆ ของบล็อกมาสเตอร์ประกอบไปด้วยปุ่ม “ทำงาน” ซึ่งจะสั่งให้โปรแกรมเริ่มทำงาน ปุ่ม “ทำซ้ำตลอด” ซึ่งจะวนทำคำสั่งทั้งหมดไปเรื่อยๆ และปุ่ม “ทำทีละขั้นตอน” ที่ช่วยให้เด็กสามารถสังเกตการทำงานของโปรแกรมได้ง่ายขึ้น โดยจะมีไฟระบุคำสั่งที่กำลังทำงานอยู่ด้วย เมื่อบล็อกมาสเตอร์ประมวลผลโปรแกรมเรียบร้อยแล้ว จะส่งคำสั่งไปยังรถหุ่นยนต์ผ่านทางอุปกรณ์ไร้สาย XBee ซึ่งทำงานตามมาตรฐาน ZigBee [6] และใช้ในวงจรอิเล็กทรอนิกส์ทั้งของบล็อกมาสเตอร์และรถหุ่นยนต์ ดังรูปที่ 3.1

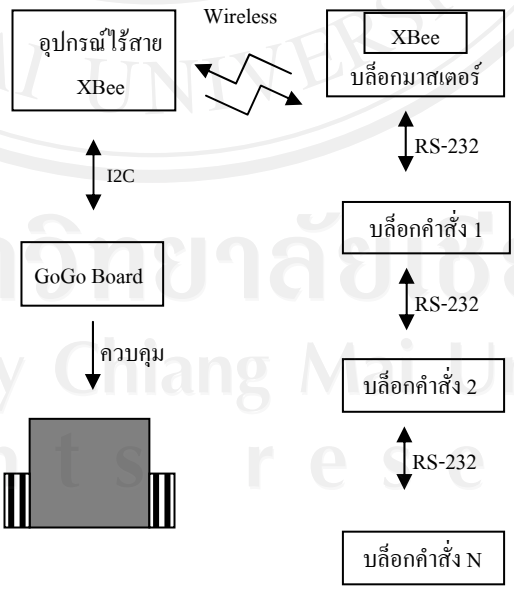
ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูปที่ 3.1 โรโบ-บล็อกที่นำมาต่อเป็นโปรแกรมโดยยังไม่ได้ประกอบกล่องอะคริลิก (ซ้าย) โรโบ-บล็อกหลังจากประกอบกล่องอะคริลิกแล้ว (ขวา)

3.1.3 รถหุ่นยนต์

รถหุ่นยนต์ประกอบจากชิ้นส่วน LEGO เป็นหลักแต่มีปากกาติดอยู่ส่วนหน้าของหุ่นยนต์ซึ่งใช้กลไกพิเศษเพื่อยกและจรดปากกา การควบคุมการเคลื่อนที่และการยกหรือการจรดปากกานั้นควบคุมโดยชุดหุ่นยนต์เพื่อการเรียนรู้สำหรับเด็กที่ชื่อ GoGo Board [19] ซึ่งเป็นเครื่องมือที่สามารถนำมาใช้งานได้ง่ายเพราะมีวงจรควบคุมมอเตอร์ในตัวและสนับสนุนการติดต่อสื่อสารแบบไร้สาย ระบบการทำงานทั้งหมดของโรโบ-บล็อกได้แสดงไว้ในรูปที่ 3.2



รูปที่ 3.2 แผนผังการติดต่อสื่อสารของโรโบ-บล็อก

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

3.2 การออกแบบวงจร

ในขั้นต้นโรโบ-บล็อกถูกพัฒนาขึ้นมาโดยใช้ไมโครคอนโทรเลอร์ตระกูล PIC เบอร์ 18F4550 แต่เนื่องจากโรโบ-บล็อกมีการติดต่อสื่อสารโดยใช้ RS-232 สองทิศทาง แต่ไมโครคอนโทรเลอร์เบอร์ 18F4550 มี RS-232 ที่เป็น Hardware เพียงด้านเดียว และถึงแม้จะสามารถใช้ RS-232 โดยการเขียน Software ขึ้นมาได้ แต่ RS-232 ที่เป็น Software นี้ก็ไม่มี Interrupt ซึ่งเป็นสิ่งจำเป็นในการตรวจสอบเมื่อมีข้อมูลเข้าสู่ไมโครคอนโทรเลอร์ ดังนั้นการใช้ไมโครคอนโทรเลอร์เบอร์ 18F4550 ในการควบคุมโรโบ-บล็อกจึงไม่เหมาะนัก จึงจำเป็นต้องเปลี่ยนไมโครคอนโทรเลอร์จากเบอร์ 18F4550 เป็นเบอร์ dsPIC30F4011 ซึ่งมี RS-232 สองชุด

วงจรของโรโบ-บล็อกประกอบด้วยวงจรของบล็อกต่างๆ 3 แบบ คือ วงจรบล็อกมาสเตอร์, วงจรบล็อกคำสั่งกลุ่มที่มีการทำงานตายตัว, วงจรบล็อกคำสั่งกลุ่มที่สามารถปรับค่าพารามิเตอร์การทำงานได้ โดยแต่ละแบบมีรายละเอียดดังนี้

3.2.1 วงจรบล็อกมาสเตอร์

บล็อกมาสเตอร์ถูกออกแบบมาให้มีลักษณะดังรูปที่ 3.3 โดยมีรายละเอียดคุณสมบัติดังนี้

- ด้านหน้าของบล็อกมีปุ่มทั้งหมด 3 ปุ่ม คือ ปุ่มสำหรับเลือกฟังก์ชั่นการทำงาน 2 ปุ่ม และปุ่มสำหรับสั่งให้รถหุ่นยนต์เริ่มทำงาน 1 ปุ่ม
- ด้านข้างของปุ่มแต่ละปุ่มมี LED แสดงสถานะการทำงานของฟังก์ชั่นและการทำงาน ของแต่ละปุ่มนั้นๆ
- ด้านล่างมีคอนเนคเตอร์(connector) สำหรับเชื่อมต่อ และรับ-ส่งข้อมูลกับบล็อกคำสั่งต่างๆ
- ด้านบนมีช่องเสียบอแดปเตอร์ เพื่อจ่ายไฟเลี้ยงให้กับบล็อกมาสเตอร์ และส่งต่อไปยังบล็อกคำสั่งอื่นๆผ่านคอนเนคเตอร์ที่อยู่ด้านล่าง
- สามารถส่งข้อมูลโปรแกรมต่างๆที่ได้ถูกต่อไว้ไปยังรถหุ่นยนต์ได้

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

รูปที่ 3.3 บล็อกมาสเตอร์ที่ถูกต่อเข้ากับบล็อกคำสั่ง

จากคุณสมบัติของบล็อกมาสเตอร์ที่ได้ออกแบบไว้ดังข้างต้น ทำให้ออกแบบวงจรการทำงานของบล็อกมาสเตอร์โดยมีรายละเอียดดังนี้

3.2.1.1 ปุ่มสำหรับเลือกฟังก์ชั่นการทำงาน และปุ่มสำหรับสั่งให้รถหุ่นยนต์เริ่มทำงาน

ส่วนนี้แสดงในส่วนของเลข 1 ในรูปที่ 3.4 ซึ่งส่วนนี้ประกอบด้วยตัวต้านทาน 3.3 กิโลโอห์ม ต่อลงกราวด์ เพื่อให้ขาไมโครคอนโทรเลอร์ที่ต่อกับปุ่มแต่ละปุ่มเป็น low เสมอ ส่วนอีกฝั่งหนึ่งของปุ่มจะต่อขึ้นไฟ เพื่อให้เวลากดปุ่มขาไมโครคอนโทรเลอร์จะมีค่าเป็น high และการใส่ตัวต้านทาน 3.3 กิโลโอห์มนี้ เพื่อให้เวลากดปุ่ม ไฟไม่ไหลลงสู่กราวด์โดยตรง (หากไฟไหลลงกราวด์โดยไม่ผ่านตัวต้านทาน จะทำให้ไฟฟ้าลัดวงจร)

3.2.1.2 LED แสดงสถานการณ์ทำงานของฟังก์ชั่นและการทำงาน

LED ในวงจรนี้มีทั้งหมด 3 ดวง ซึ่งแต่ละดวงแสดงสถานการณ์ทำงานของฟังก์ชั่นและการทำงานของโปรแกรม โดยวงจร LED อยู่ที่ส่วนหมายเลข 2 ของรูปที่ 3.4 จากรูปจะเห็นได้ว่า มีตัวต้านทาน 100 โอห์ม ต่อเข้ากับขาของไมโครคอนโทรเลอร์ และอีกฝั่งหนึ่งของตัวต้านทานต่อเข้ากับหลอด LED แล้วต่อลงกราวด์ ซึ่งวงจรลักษณะนี้เป็นวงจรปกติสำหรับหลอด LED แต่ค่าของตัวต้านทานจะแตกต่างกันไปตามประเภทของหลอด LED

3.2.1.3 คอนเนคเตอร์สำหรับเชื่อมต่อ และรับ-ส่งข้อมูลกับบล็อกคำสั่งต่างๆ

การรับ-ส่งข้อมูลในโรโบ-บล็อกนี้ ใช้การส่งข้อมูลตามมาตรฐานของ RS-232 ดังที่กล่าวมาในข้างต้น นั่นคือจะมีสายไฟสำหรับรับ TX, RX และกราวด์ และนอกจากนี้ยังได้เพิ่มไฟสำหรับส่งต่อไฟเลี้ยงไปยังบล็อกคำสั่งถัดไปด้วย

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

โดยคอนเนคเตอร์ที่ใช้ในโรโบ-บล็อกนี้ คือกระดุมแม่เหล็ก เนื่องจากผู้วิจัยได้ทดลองปรับเปลี่ยนคอนเนคเตอร์มาหลายรูปแบบ เช่น คอนเนคเตอร์แบบ AUX แต่เนื่องจากคอนเนคเตอร์แบบนี้สามารถต่อกับสายไฟได้เพียง 3 เส้น จึงไม่เพียงพอสำหรับการใช้ในงานนี้ อีกทั้งยังมีความยืดหยุ่นในการเคลื่อนที่ต่ำ คอนเนคเตอร์แบบ USB ที่สามารถส่งข้อมูลได้ถึง 4 ประเภทซึ่งตรงกับความต้องการของโรโบ-บล็อก แต่ก็ยังมีปัญหาด้านการต่อ-ถอดประกอบ ซึ่งต้องมีความแม่นยำในการต่อประกอบกันแต่ละครั้ง และยังต้องใช้แรงในการถอดออก จึงไม่เหมาะสมกับงานนี้ซึ่งมีผู้ใช้เป็นเด็ก ดังนั้นผู้วิจัยจึงได้ศึกษางานวิจัยอื่นๆที่มีการประกอบสิ่งของต่างๆและมีผู้ใช้งานเป็นเด็กเหมือนกัน พบว่า คอนเนคเตอร์ที่เหมาะสมที่สุดคือแม่เหล็ก ซึ่งสามารถประกอบและถอดได้ง่าย ไม่ต้องใช้แรงมาก ไม่ต้องการความแม่นยำสูง ผู้วิจัยจึงคิดนำกระดุมแม่เหล็กมาใช้ในงานนี้

โดยวงจรของคอนเนคเตอร์นี้อยู่ในส่วนที่ 3 ของรูปที่ 3.4 วงจรนี้จะต่อมาจากขา TX และ RX ของไมโครคอนโทรเลอร์โดยตรง เพื่อต่อเข้ากับขา RX และ TX ของไมโครคอนโทรเลอร์ของบล็อกคำสั่งต่างๆ โดยขา TX จะต่อเข้ากับขา RX ของบล็อกคำสั่ง และขา RX จะต่อเข้ากับขา TX

3.2.1.4 การแปลงไฟสำหรับเลี้ยงวงจร

ที่ด้านบนของบล็อกมาสเตอร์จะมีช่องสำหรับเสียบอแดปเตอร์ซึ่งทำการแปลงไฟกระแสสลับ 220 โวลต์ ให้เป็นกระแสตรง 12 โวลต์อยู่แล้ว แต่เนื่องจากไมโครคอนโทรเลอร์ที่ใช้ใช้ไฟ 5 โวลต์ ดังนั้นจึงจำเป็นต้องมีการแปลงไฟจาก 12 โวลต์ให้เป็น 5 โวลต์ ซึ่งวงจรในส่วนนี้คือหมายเลข 4 ในรูปที่ 3.4

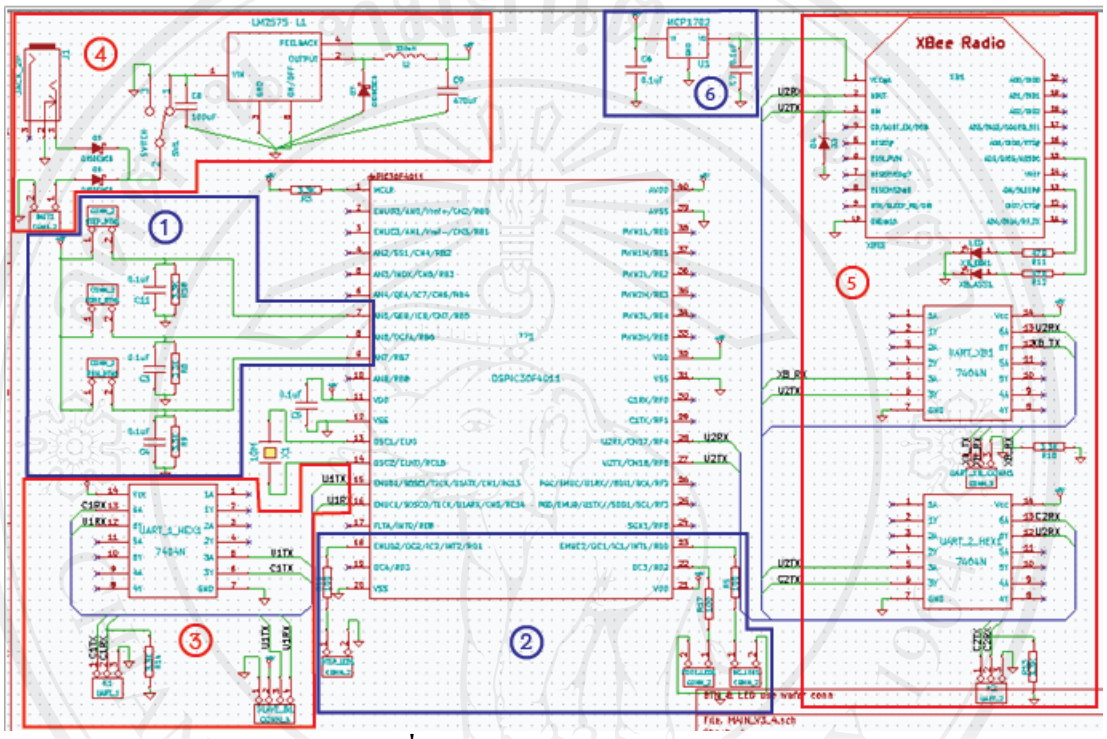
วงจรของบล็อกมาสเตอร์มีการแปลงไฟแบบ Switching ซึ่งการแปลงไฟแบบนี้มีข้อดีคือ วงจรมีขนาดเล็ก มีความร้อนไม่มากนักจึงไม่ต้องมีฮีสซิง(heat zing) เพื่อระบายความร้อนและไม่เปลืองพลังงานโดยป่าวประโยชน์ เนื่องจากไม่ได้ดึงกระแสไฟส่วนเกินลงกราวด์เหมือนวงจรแปลงกระไฟแบบ linear โดยในวงจรนี้ใช้ IC เบอร์ LM 2575-5 ในการแปลงกระแสไฟจาก 12 โวลต์ให้เหลือ 5 โวลต์

3.2.1.5 การส่งข้อมูลโปรแกรมไปยังรถหุ่นยนต์

โรโบ-บล็อกใช้ XBEE ในการส่งข้อมูลจากบล็อกมาสเตอร์ไปยังรถหุ่นยนต์ ดังที่ได้กล่าวมาข้างต้น ซึ่งแท้จริงแล้วสามารถต่อขา DOUT ของ XBEE เข้ากับขา RX ของไมโครคอนโทรเลอร์ได้โดยตรง เพื่อให้ไมโครคอนโทรเลอร์รับข้อมูลจาก XBEE และต่อขา DIN ของ XBEE เข้ากับขา TX ของไมโครคอนโทรเลอร์เข้ากับขา TX ของ XBEE ดังส่วนที่ 4 ของรูปที่ 3.4

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

แต่เนื่องจาก XBEE ใช้ไฟเลี้ยงเพียง 3.0-3.4 โวลต์ จึงจำเป็นต้องแปลงไฟจาก 5 โวลต์ ให้เหลือเพียง 3.3 โวลต์ ซึ่งในวงจรนี้ได้ใช้ IC เบอร์ MCP 1702 ในการแปลงไฟ ดังส่วนที่ 6 ของรูปที่ 3.4



รูปที่ 3.4 วงจรของบล็อกมาสเตอร์

3.2.2 วงจรบล็อกคำสั่งกลุ่มที่มีพารามิเตอร์ตายตัว

บล็อกคำสั่งกลุ่มที่มีพารามิเตอร์ตายตัว ได้แก่คำสั่ง ส่งเสียง จรดปากกา และยกปากกา โดยบล็อกคำสั่งในกลุ่มนี้ไม่จำเป็นต้องปรับพารามิเตอร์การทำงานจึงไม่มีลูกบิดที่บล็อก ซึ่งต่างจากบล็อกคำสั่งที่สามารถปรับพารามิเตอร์ได้

บล็อกคำสั่งกลุ่มที่มีพารามิเตอร์ตายตัว มีคุณสมบัติที่ได้ออกแบบไว้ดังรูปที่ 3.5 ซึ่งมีรายละเอียดดังนี้

- มีคอนเนคเตอร์สำหรับประกอบและรับ-ส่งข้อมูลกับบล็อกต่างๆ ที่ด้านบนและล่างของบล็อก
- LED ที่ด้านบนของบล็อก เพื่อใช้แสดงสถานะการทำงาน

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

รูปที่ 3.5 บล็อกส่งเสียง ซึ่งเป็นบล็อกคำสั่งที่ไม่ต้องปรับพารามิเตอร์

คุณสมบัติของบล็อกคำสั่งประเภทนี้มีไม่มากนัก ทำให้วงจรมีความซับซ้อนไม่มาก ดังรูปที่ 3.6 ซึ่งมีรายละเอียดดังนี้

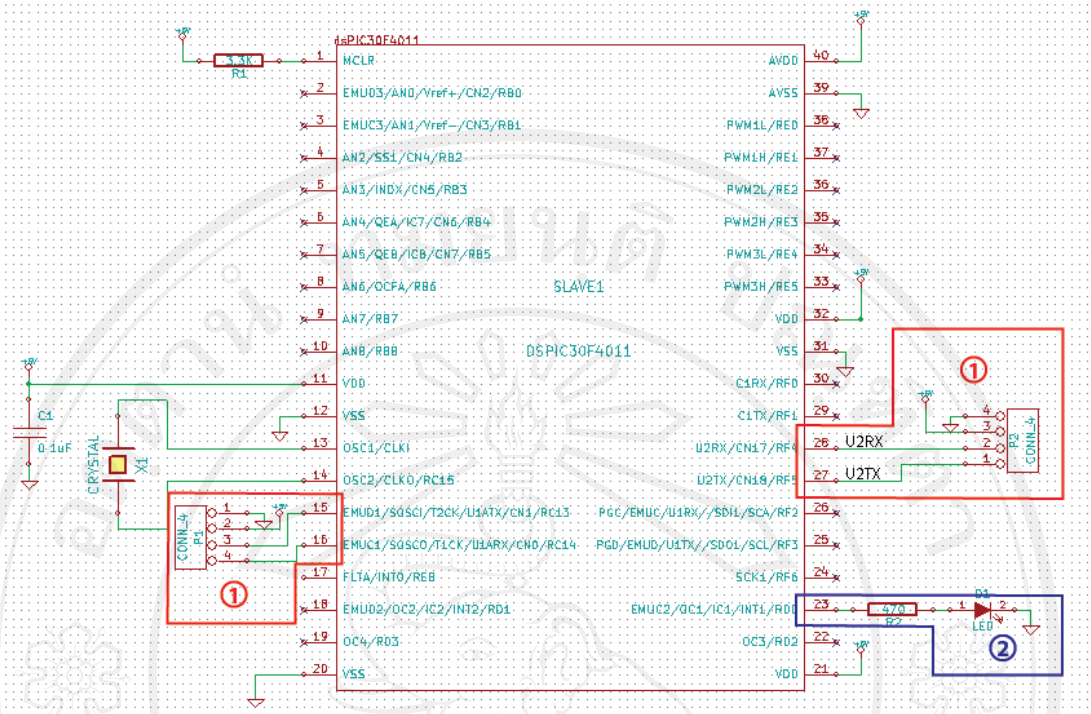
3.2.2.1 คอนเนคเตอร์สำหรับเชื่อมต่อ และรับ-ส่งข้อมูลกับบล็อกคำสั่งต่างๆ

คอนเนคเตอร์นี้มีความคล้ายคลึงกับคอนเนคเตอร์ที่ได้กล่าวถึงในบล็อกมาสเตอร์ข้างต้นแล้ว แต่ในบล็อกคำสั่งนี้จะมีคอนเนคเตอร์สองด้านด้วยกัน เพื่อใช้ต่อกับบล็อกต่างๆ ซึ่งคอนเนคเตอร์ทั้งสองต่างกันเพียงการเรียงสายข้อมูล นั่นคือ คอนเนคเตอร์ฝั่งด้านบนเรียงสาย TX, RX, กราวด์, ไฟ ส่วนคอนเนคเตอร์ฝั่งด้านล่างเรียง RX, TX, กราวด์, ไฟ เหตุที่ RX และ TX ต้องสลับกันเนื่องจาก ในการสื่อสารกันระหว่างไมโครคอนโทรเลอร์โดยตรง ขา RX ซึ่งเป็นขาที่ใช้รับข้อมูลต้องต่อเข้ากับขา TX ซึ่งใช้ในการส่งข้อมูลของไมโครคอนโทรเลอร์อีกตัว ในทางตรงกันข้าม ขา TX ซึ่งเป็นขาที่ใช้ในการส่งข้อมูล ก็จะต้องต่อเข้ากับขา RX ซึ่งใช้ในการรับข้อมูลของไมโครคอนโทรเลอร์อีกตัวหนึ่ง ดังวงจรส่วนที่ 1 ของรูปที่ 3.6

3.2.2.2 LED แสดงสถานะการทำงานของคำสั่ง

ในส่วนที่ 2 ของรูปที่ 3.6 เป็นวงจรของไฟ LED ซึ่งใช้ในการแสดงสถานะการทำงานของบล็อกคำสั่ง นั่นคือ หากคำสั่งของบล็อกนี้ทำงานอยู่ ไฟ LED ดวงนี้จะติดขึ้น โดยรายละเอียดของวงจรได้กล่าวไว้ในส่วนของบล็อกมาสเตอร์แล้ว

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูปที่ 3.6 วงจรของบล็อกคำสั่งที่มีพารามิเตอร์ตายตัว

3.2.3 วงจรบล็อกคำสั่งกลุ่มที่สามารถปรับพารามิเตอร์การทำงานได้

วงจรของบล็อกคำสั่งกลุ่มที่สามารถปรับพารามิเตอร์การทำงานได้เป็นวงจรที่มีความซับซ้อนมากที่สุด เนื่องจากวงจรบล็อกคำสั่งประเภทนี้ประกอบด้วยวงจรจำนวน 2 วงจรด้วยกัน คือ วงจรของบล็อกคำสั่งคล้ายกับวงจรบล็อกคำสั่งกลุ่มที่มีพารามิเตอร์ตายตัว แต่เนื่องจากสามารถปรับพารามิเตอร์ได้ จึงต้องมีวงจรสำหรับแสดงค่าของพารามิเตอร์นั้น ดังรูปที่ 3.7 ซึ่งวงจรบล็อกคำสั่งกลุ่มที่สามารถปรับพารามิเตอร์การทำงานได้ มีคุณสมบัติที่ออกแบบไว้ดังนี้

- มีคอนเนคเตอร์สำหรับประกอบและรับ-ส่งข้อมูลกับบล็อกต่างๆ ที่ด้านบนและล่างของบล็อก
- LED ที่ด้านบนของบล็อก เพื่อใช้แสดงสถานะการทำงาน
- ลูกบิดด้านซ้าย สำหรับปรับพารามิเตอร์
- หน้าจอด้านบน สำหรับแสดงตัวเลขพารามิเตอร์

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

รูปที่ 3.7 ภายในของบล็อกคำสั่งกลุ่มที่สามารถปรับพารามิเตอร์การทำงานได้ ประกอบด้วยแผ่นวงจร 2 แผ่น

จากคุณสมบัติที่ได้ออกแบบไว้ในข้างต้น จะเห็นได้ว่า คุณสมบัติการมีคอนเนคเตอร์สำหรับประกอบและรับ-ส่งข้อมูลกับบล็อกต่างๆ ที่ด้านบนและล่างของบล็อก และการมี LED ที่ด้านบนของบล็อก เพื่อใช้แสดงสถานะการทำงาน นั้นเป็นคุณสมบัติพื้นฐานที่บล็อกคำสั่งทุกบล็อกต้องมีอยู่แล้ว จากรูปที่ 3.8 ได้แสดงคุณสมบัติดังกล่าวไว้ในส่วนที่ 1 และ 2 ตามลำดับ และรายละเอียดของวงจรได้กล่าวไปแล้วในส่วนของวงจรบล็อกคำสั่งกลุ่มที่มีพารามิเตอร์ตายตัว

ดังนั้นจึงขอกล่าวถึงเฉพาะส่วนที่เป็นคุณสมบัติเฉพาะของวงจรบล็อกคำสั่งกลุ่มที่สามารถปรับพารามิเตอร์การทำงานได้นี้ นั่นคือ

3.2.3.1 วงจรการรับค่าพารามิเตอร์จากลูกบิด

วงจรนี้อยู่ในส่วนที่ 3 ของรูปที่ 3.8 ซึ่งเป็นวงจรรับค่าเซ็นเซอร์ อาศัยหลักการอ่านกระแสไฟของขาไมโครคอนโทรเลอร์ที่เปลี่ยนแปลงตามความต้านทานค่าต่างๆ โดยจากขาของไมโครคอนโทรเลอร์จะต่อด้วยตัวต้านทาน 33 กิโลโอห์ม ดึงขึ้นไฟ และต่อขนานกับเซ็นเซอร์ซึ่งในวงจรนี้คือตัวต้านทานปรับค่าได้ ที่เชื่อมต่อลงกราวด์ และมีตัวเก็บประจุ 1 uF ต่อขนานกับเซ็นเซอร์ในลักษณะของ low pass filter เพื่อให้การอ่านค่าเซ็นเซอร์ช้าลง โดยยิ่งตัวเก็บประจุมีค่ามากการอ่านค่ายิ่งช้า แต่หากตัวเก็บประจุมีค่าน้อย การอ่านค่าของเซ็นเซอร์จะสามารถอ่านได้เร็วขึ้น

จากวงจรนี้ หากตัวต้านทานปรับค่าได้มีค่าความต้านทานเป็น 0 โอห์ม จะทำให้กระแสไฟไหลลงกราวด์ได้เต็มที่ ทำให้มีกระแสไฟเข้ามายังขาของไมโครคอนโทรเลอร์น้อย แต่หากตัวต้านทานปรับค่าได้มีค่าความต้านทานสูง จะทำให้กระแสไฟไหลลงกราวด์ได้น้อย ส่งผลให้กระแสไฟเข้ามายังขาของไมโครคอนโทรเลอร์ได้มาก

จากหลักการดังกล่าว จึงคำนวณค่าเซ็นเซอร์ได้โดยใช้สูตร

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

$$RS = \frac{Rref \times OUTPUT}{1023 - OUTPUT}$$

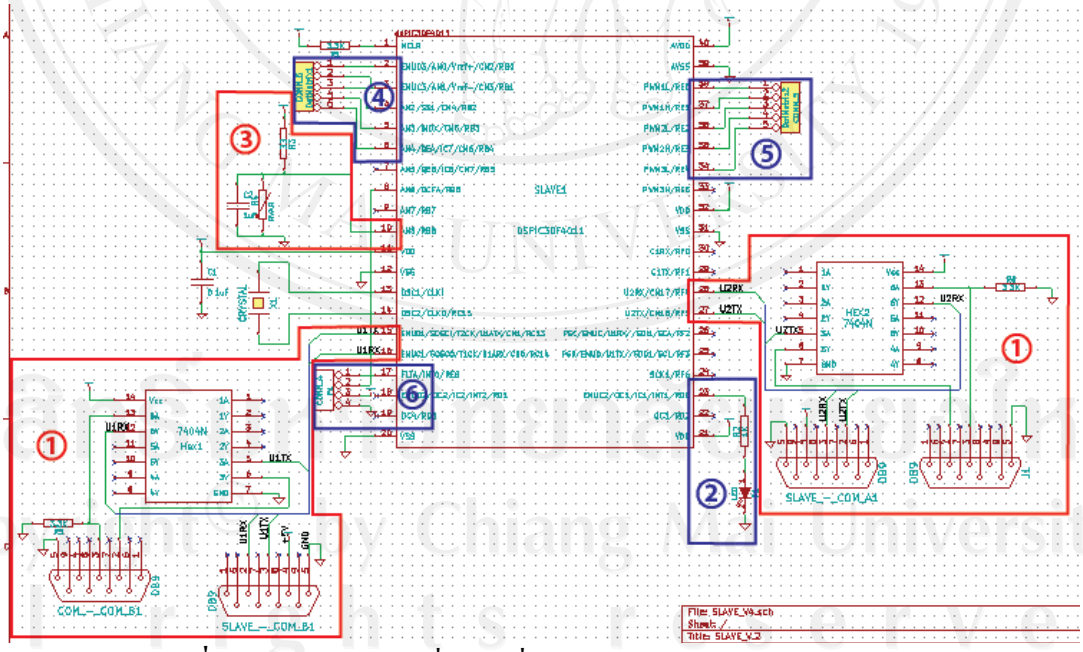
- RS คือ ค่าความต้านทานของเซ็นเซอร์
- R reference คือ ค่าของตัวต้านทานที่ดึงขึ้นไฟ นั่นคือ 33,000
- OUTPUT คือ ค่าที่อ่านได้จากค่าเซ็นเซอร์

จากนั้นจึงนำค่า RS ไปใช้ในการเขียนโปรแกรมเพื่อคำนวณค่าพารามิเตอร์ต่อไป

3.2.3.2 การส่งข้อมูลไปยังวงจรขับ LED Dot Matrix

เนื่องจากวงจรสำหรับขับ LED Dot Matrix ซึ่งเป็นจอแสดงค่าพารามิเตอร์ ซึ่งเป็นวงจรที่แยกจากวงจรของบล็อกคำสั่ง ดังนั้นในการเชื่อมต่อวงจรทั้งสองนี้จึงทำการเชื่อมต่อโดยผ่านคอนเนคเตอร์ ดังส่วนที่ 4, 5 และ 6 ของรูปที่ 3.8

- ส่วนที่ 4 เป็นส่วนที่ส่งไฟไปยัง LED Dot Matrix ตัวที่ 1 ซึ่งเป็นหลักสิบ
- ส่วนที่ 5 เป็นส่วนที่ส่งไฟไปยัง LED Dot Matrix ตัวที่ 2 ที่ใช้แสดงหลักหน่วย
- ส่วนที่ 6 เป็นส่วนที่ส่งพัลส์ ไปยัง IC ที่ขับ LED โดยตรง



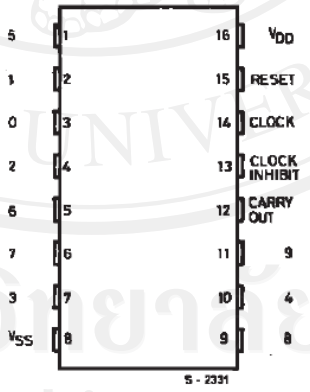
รูปที่ 3.8 วงจรบล็อกคำสั่งกลุ่มที่สามารถปรับพารามิเตอร์การทำงานได้

3.2.3.3 วงจรขับ LED Dot Matrix

วงจรขับ LED Dot Matrix เป็นวงจรสำหรับขับ LED Dot Matrix ทั้ง 2 ตัวให้แสดงตัวเลขตามที่ไมโครคอนโทรเลอร์ของวงจรบล็อกคำสั่งต้องการ โดยเชื่อมต่อกับวงจรบล็อกคำสั่งผ่านคอนเน็คเตอร์ โดยการนำวงจรขับ LED Dot Matrix เสียบลงบนวงจรของบล็อกคำสั่ง ดังรูปที่ 3.8

โดยส่วนที่ 1 ของรูปที่ 3.12 ซึ่งเป็นวงจรขับ LED นี้ จะเชื่อมต่อกับส่วนที่ 6 ของวงจรบล็อกคำสั่ง โดยส่วนนี้จะมีการเชื่อมต่อกันด้วยสายไฟ 4 เส้น คือ

- ไฟ สำหรับเลี้ยงวงจรขับ LED Dot Matrix
- กราวด์
- พัลล์สัญญาณนาฬิกา สำหรับสร้างให้กับ IC HCF4017B ซึ่งเป็น IC Counter โดยมีการทำงานคือ เมื่อได้รับพัลล์มาจากวงจรบล็อกคำสั่ง IC ตัวนี้จะส่งกระแสไฟผ่านขาหมายเลข 0 และเมื่อได้รับพัลล์อีกครั้ง จะเปลี่ยนการส่งกระแสจากขาหมายเลข 0 ให้เป็นขาหมายเลข 1 และเมื่อได้รับพัลล์อีก จะเปลี่ยนการส่งกระแสจากขาหมายเลข 1 เป็นหมายเลข 2 ดังรูปที่ 3.9 โดยทุกครั้งที่ได้รับพัลล์จะเปลี่ยนขาที่ส่งกระแสไปเรื่อยๆ จนครบทั้ง 10ขา นั่นคือ ถึงขาหมายเลข 9 ซึ่งพัลล์ที่จะได้รับครั้งต่อไป จะทำให้วนกลับเปลี่ยนไปส่งกระแสที่ขาหมายเลข 0 วนเช่นนี้ไปเรื่อยๆ



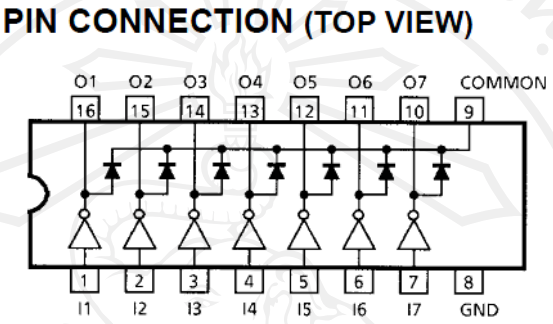
รูปที่ 3.9 ขาของ IC HCF4017B

- RESET เนื่องจาก IC เบอร์ HCF4017B มี Decode Output ถึง 10 ขา ในการใช้ปกติจะต้องขับไล่จาก Output ขาที่ 0 จนถึงขาที่ 9 แต่เนื่องจากโรโบ-บล็อกใช้ LED Dot Matrix ขนาด 5*7 ดังนั้นจึงใช้ Output เพียงแค่ 7 ขา คือ 1-7 ซึ่งจำเป็นต้องมีการ Reset เพื่อให้

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

กลับมาส่ง Output ผ่านขาที่ 0 ทันทีโดยไม่ต้องวนจนครบทั้ง 10 ขา การ Reset จะเกิดขึ้นทุกครั้งที่มีการส่งพัลส์เข้ามาที่ขา Reset

ขา Output ของ IC HCF4017B จะถูกเชื่อมต่อทรานซิสเตอร์อาร์เรย์ ดังส่วนที่ 2 ของรูปที่ 2.12 ซึ่งทรานซิสเตอร์อาร์เรย์ เบอร์ ULN2003 ที่ใช้นี้เมื่อขา IN หมายเลขใดได้รับไฟเข้าทางฝั่งขา Output หมายเลขนั้นๆก็จะสับลงกราวด์ ดังรูปที่ 3.10



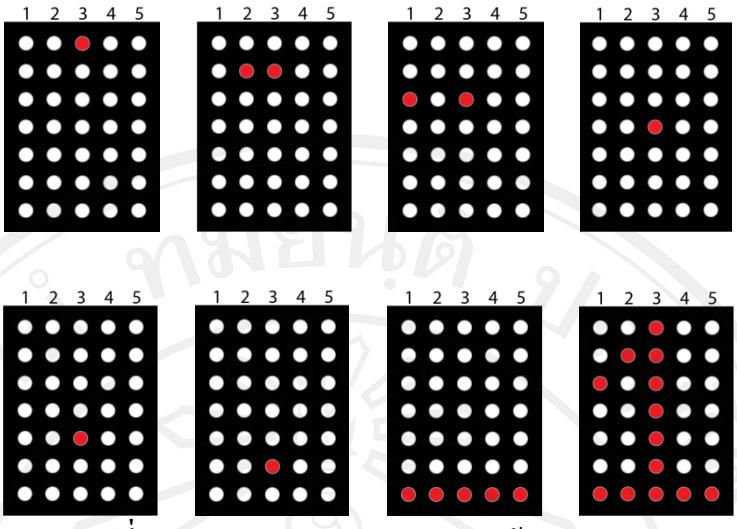
รูปที่ 3.10 การทำงานของทรานซิสเตอร์อาร์เรย์

ส่วนที่ 4 และส่วนที่ 5 ของรูปที่ 3.12 เป็น LED Dot Matrix ที่เป็นหน้าจอเพื่อแสดงค่าพารามิเตอร์ โดยส่วนที่ 4 จะเชื่อมต่อกับส่วนที่ 4 ของรูปที่ 3.8 และส่วนที่ 5 จะเชื่อมต่อกับส่วนที่ 5 ของรูปที่ 3.8 เช่นกัน

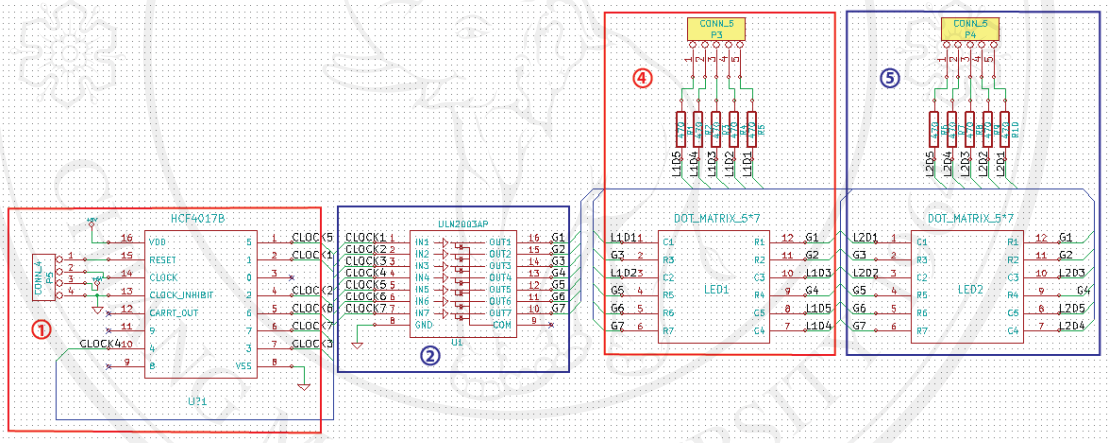
โดยการเชื่อมต่อนี้ ไมโครคอนโทรเลอร์จากบล็อกคำสั่งจะส่งกระแสไฟมายัง LED Dot Matrix ตัวละ 5 เส้น เพื่อจ่ายไฟให้กับ LED ดวงเล็กๆบน Dot Matrix ตามแถวแนวตั้ง จากตัวอย่างที่แสดงในรูปที่ 3.11 ที่ขับตัวเลข 1 จะเริ่มต้นจากไมโครคอนโทรเลอร์ส่งกระแสไฟมายังสายไฟเส้นที่ 3 เพื่อขับ LED ซึ่งในขณะนั้น แถวนอนแถวแรกได้ถูกสับลงกราวด์ดังส่วนที่ 1 จากนั้นไมโครคอนโทรเลอร์จะส่งไฟมายังสายที่ 2 และ 3 พร้อมกับที่แถวที่ 2 แนวนอนถูกเปลี่ยนเป็นกราวด์ดังที่เห็นในส่วนที่ 2 แล้วไมโครคอนโทรเลอร์เปลี่ยนส่งไฟแถวที่ 1 และ 3 พร้อมกับที่แถวที่ 3 แนวนอนถูกเปลี่ยนเป็นกราวด์ดังส่วนที่ 3 และเปลี่ยนเช่นนี้ไปเรื่อยๆ จนกระทั่งถึงส่วนที่ 7 ไมโครคอนโทรเลอร์ปล่อยกระแสไฟมาทุกสายพร้อมกับแถวที่ 7 ถูกเปลี่ยนให้ลงกราวด์ และจากนั้นจะกลับมาทำยังส่วนแรกใหม่

LED จะถูกขับดังที่กล่าวมาข้างต้นเป็นจังหวะที่สม่ำเสมอและรวดเร็ว จนทำให้มองเห็นเป็นเลข 1 ดังที่เห็นในส่วนที่ 3.11

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



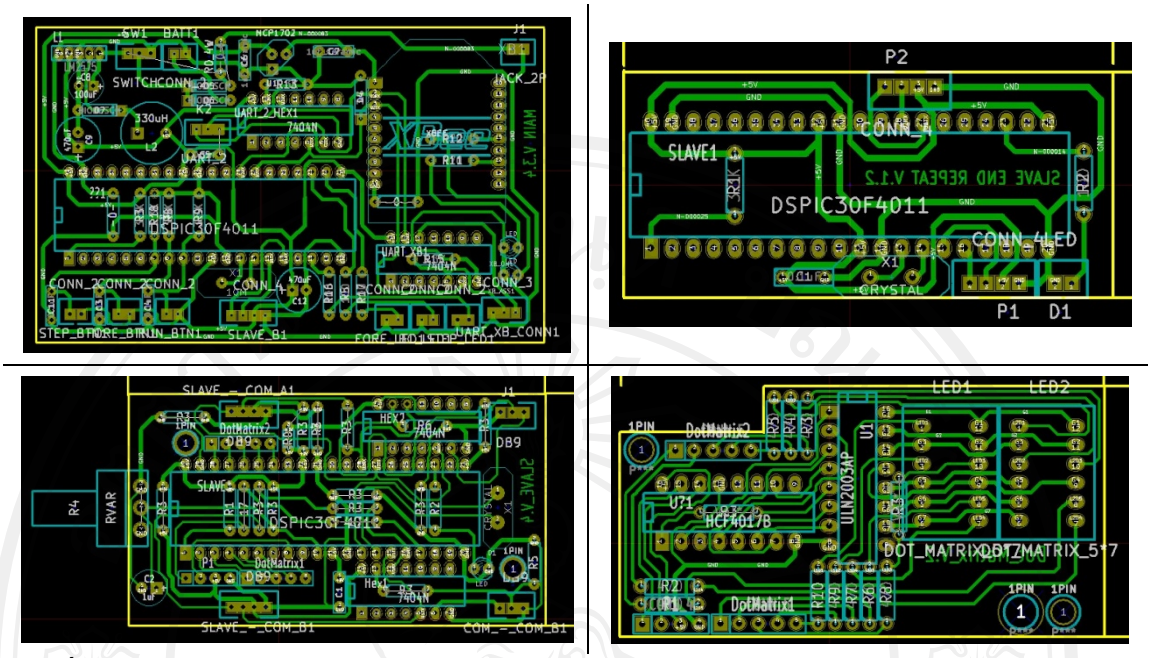
รูปที่ 3.11 การขับ LED Dot Matrix ให้แสดงเลข 1



รูปที่ 3.12 วงจรขับ LED Dot Matrix

จากวงจรที่ได้อธิบายมาในข้างต้น ได้ทำการออกแบบเป็นแผ่นวงจรพิมพ์ได้ดังรูปที่ 3.13

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูปที่ 3.13 วงจรบล็อกมาสเตอร์(บนซ้าย), วงจรบล็อกคำสั่งกลุ่มที่มีการทำงานตายตัว (บนขวา), วงจรบล็อกคำสั่งกลุ่มที่สามารถปรับค่าพารามิเตอร์การทำงานได้(ล่างซ้าย), และวงจรสำหรับแสดงค่าพารามิเตอร์การทำงาน(ล่างขวา)

3.3 โปรโตคอลการตรวจสอบโครงสร้างของโปรแกรม

ระบบการทำงานของโรโบ-บล็อกถูกพัฒนาขึ้นมาด้วยภาษา C รูปที่ 3.14 แสดงตัวอย่างลำดับการทำงานของโปรโตคอลที่ใช้ตรวจสอบโครงสร้างของบล็อกที่ถูกต่อไว้ ซึ่งลำดับขั้นตอนการทำงานโดยละเอียดมีดังต่อไปนี้

(1) เมื่อกดปุ่ม “ทำงาน” บนบล็อกมาสเตอร์ บล็อกมาสเตอร์จะส่งสัญญาณบรอดคาสไปยังบล็อกคำสั่งแรก เมื่อบล็อกคำสั่งได้รับสัญญาณนี้สิ่งแรกที่ต้องทำคือตรวจสอบดูว่ามันเป็นบล็อกสุดท้ายของโปรแกรมหรือไม่โดยการส่งต่อคำสั่งบรอดคาสไปยังบล็อกถัดไป ผลที่ได้จะมีสองกรณี

(1.1) ถ้าถัดจากตัวมันมีบล็อกต่ออยู่มันจะได้รับสัญญาณตอบรับกลับมา

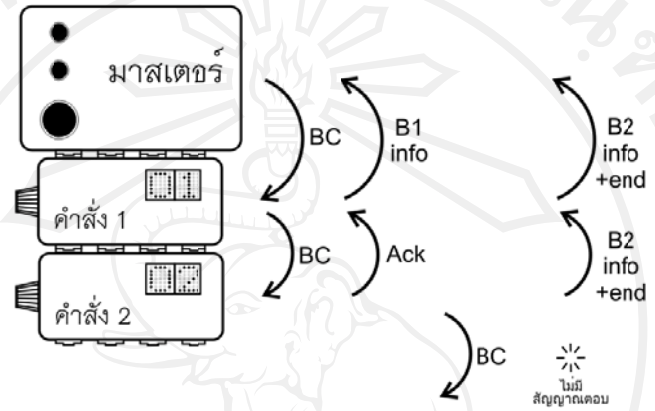
(1.2) ถ้าไม่มีสัญญาตอบกลับมาภายใน 50 มิลลิวินาทีก็แสดงว่าไม่มีบล็อกต่ออยู่และสรุปได้ว่าตัวมันเป็นบล็อกสุดท้าย

(2) บล็อกคำสั่งจะส่งข้อมูลเกี่ยวกับตัวมันเอง เช่น ชนิดคำสั่ง และพารามิเตอร์ ไปยังบล็อกมาสเตอร์โดยถ้าหากว่าผลจากข้อ (1) คือมันเป็นบล็อกสุดท้ายมันก็จะส่งคำสั่งจบโปรแกรมไปด้วย

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

(3) บล็อกคำสั่งอื่นๆ ที่ต่ออยู่ถัดลงไปก็จะอาศัยการทำงานตามข้อ (1) และ (2) ไปเรื่อยๆ จนบล็อกมาสเตอร์ได้รับข้อมูลโครงสร้างของโปรแกรมครบถ้วน

บล็อกมาสเตอร์จะเก็บข้อมูลทั้งหมดที่ได้รับจากบล็อกคำสั่งไว้ใน EEPROM เพื่อทำการประมวลผล และส่งไปยังรถหุ่นยนต์ต่อไป



รูปที่ 3.14 การทำงานของโรโบ-บล็อก BC คือสัญญาณบรอดคาส Ack คือสัญญาตอบกลับ Bx info คือข้อมูลของบล็อก x และ end คือจบโปรแกรม

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved