สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

แบบจำลองการขนส่งในไลน์การผลิต

Model of Transportation in Assembly Line

โดย

นายสัจจพันธ์ สัจจารักษ์

นายสุทัศน์ ธานีรัตน์

นายสุรคุณ ขิระทาน

ว.วท.
ส548ป
2551

เลขหมู่.....
เลขทะเบียน 103031
วัน,เดือน,ปี 24 ส.ค. 2552

b 12094031
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2551

ภาควิชาวิศวกรรมระบบควบคุม สาขาวิชาวิศวกรรมแมคคาทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แบบจำลองการขนส่งในไลน์การผลิต

Model of Transportation in Assembly Line

ผู้จัดทำ 1. นายสัจจพันธ์ สัจจารักษ์ 48010949

2. นายสุทัศน์ ธานีรัตน์ 48010987

3. นายสุรคุณ ขิระทาน 48011025

.....อาจารย์ที่ปรึกษา

(ผศ.ดร.คงศักดิ์ อนันตหิรัญรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบจำลองการขนส่งในไลน์การผลิต

โดย

นายสัจจพันธ์ สัจจารักษ์

นายสุทัศน์ ธานีรัตน์

นายสุรคุณ ขิระทาน

อาจารย์ที่ปรึกษา

ผศ.ดร.ดงศักดิ์ อนันตหิรัญรัตน์

ปีการศึกษา 2551

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการศึกษารูปแบบการขนส่งในไลน์การผลิตซึ่งมีลักษณะเป็นการขนส่งในไลน์การผลิตที่มีการขนส่งชิ้นส่วนการประกอบให้แก่สายการผลิต โดยปกติโรงงานที่ได้ไปศึกษาจะใช้คนในการเข็นรถเพื่อส่งชิ้นส่วนประกอบ ทำให้ได้ประสิทธิภาพต่ำและเป็นการสิ้นเปลืองบุคลากรที่ทำงานตรงส่วนนี้ แบบจำลองการขนส่งในไลน์การผลิตนี้จะเป็นแบบระบบอัตโนมัติที่ออกแบบระบบให้ใช้หุ่นยนต์รถขนส่งเป็นรถที่รับชิ้นงานจากเครนอัตโนมัติและสามารถวิ่งเข้าไปในไลน์การผลิตเพื่อส่งชิ้นส่วนประกอบโดยใช้คนควบคุมที่จุดติดตั้งเครนอัตโนมัติเพียงคนเดียว การทำงานทั้งในส่วนของเครนอัตฌนมัติและรถขนส่งจะใช้ไมโครคอนโทรลเลอร์ MCS-51 ควบคุมและมีการติดต่องระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเอตร์เพื่อแสดงผลการทำงานผ่านทางโปรแกรม Visual Basic 6.0 ทำให้มีความสะดวกและสามารถนำระบบนี้ไปพัฒนาใช้ในระบบอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MODEL OF TRANSPORTATION IN ASSEMBLY LINE

By

Mr.Satjapan Satjarak

Mr.Sutat Thaneerat

Mr.Surakun Kiratan

Advisor

Asst.Prof.Kongsak Anuntahirunrat

Academic Year 2008

ABSTRACT

This thesis is study case model transportation in assembly line . From normal case have worker for carry parts to assembly line so this case is low efficiency and consumer the worker. This model designed for automation system. It have automatic car for carry part automatic crane to assembly line. The system control by one people at automatic crane. This system use microcontroller MCS-51 and have interface between with hardware by Visual Basic 6.0 program. The one is easy to work and succeed to develop for other industry.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้และตัวโครงงาน สามารถสำเร็จลงได้ด้วยดีก็เพราะได้รับความกรุณาจากอาจารย์ ดร.คงศักดิ์ อนันตหิรัญรัตน์ ซึ่งเป็นอาจารย์ที่ปรึกษาและยังคอยให้คำปรึกษารวมทั้งให้คำแนะนำซึ่งเป็นประโยชน์อย่างยิ่งในการทำปริญญานิพนธ์นี้ และต้องขอขอบพระคุณอาจารย์ในภาควิชาวิศวกรรมระบบควบคุมทุกท่านที่ได้ประสิทธิประสาทวิชาความรู้แก่ผู้จัดทำ ซึ่งผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูงมานะโอกาสนี้ด้วย

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ของผู้จัดทำที่ได้ให้การสนับสนุนทุกอย่างและเป็นผู้ให้กำลังใจแก่ผู้จัดทำมาโดยตลอด

ขอขอบคุณ บริษัทไทยแอโรว์จำกัด ที่ทำให้ผู้จัดทำได้แนวความคิดที่จะทำโครงงานชิ้นนี้ขึ้นมา และขอขอบคุณเพื่อนทุกคนที่คอยช่วยเหลือในการเรียนและเป็นที่ปรึกษา จนทำให้ผู้จัดทำได้ทำงานนี้สำเร็จลงได้ และขอพระคุณทุกท่านที่คอยช่วยเหลือและให้คำแนะนำ

ผู้จัดทำ

นายสัจจพันธ์ สัจจารักษ์

นายสุทัศน์ ธานีรัตน์

นายสุระคุณ ขิระทาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ ---------- II

บทคัดย่อภาษาอังกฤษ ---------- III

กิตติกรรมประกาศ ---------- IV

สารบัญ ---------- V

สารบัญรูป ---------- V II

สารบัญตาราง ---------- IX

บทที่ 1 บทนำ ---------- 1

1.1 วัตถุประสงค์ของโครงงาน ---------- 2

1.2 ขอบเขตโครงงาน ---------- 2

1.3 ตารางแผนการทำงาน ---------- 4

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง ---------- 5

2.1 Microprocessor 8051 ---------- 5

2.1.1 โครงสร้างภายในไมโครคอนโทรลเลอร์เบอร์ 8051 ---------- 5

2.1.2 ตัวตั้งเวลาและตัวนับ (Timer/Counter) ---------- 6

2.1.3 8051 กับการ interrupt วงจรนับ/จับเวลา ---------- 6

2.1.4 ส่วนควบคุมการทำงานของวงจรนับ/จับเวลา ---------- 8

2.1.5 การ Initial ค่า Baud rate ---------- 8

2.1.6 การทำงานเป็นตัวนับสัญญาณ ---------- 9

2.1.7 การสื่อสารข้อมูลแบบอนุกรม (SERIAL PORT COMMUNICATION) --- 9

2.1.8 การจัดการข้อมูลอนุกรมของ 8051 ---------- 11

2.1.9 กระบวนการรับและส่งข้อมูลอนุกรมของ 8051 ---------- 13

2.1.10 การเชื่อมต่อแบบมาตรฐาน RS-232C ---------- 13

2.2 มอเตอร์กระแสตรง ---------- 15

2.2.1 หลักการทำงานของมอเตอร์กระแสตรง ---------- 15

2.2.2 คุณสมบัติของมอเตอร์กระแสตรง ---------- 15

2.2.3 โมเดลคณิตศาสตร์ของดีซีมอเตอร์ ---------- 16

2.2.4 การควบคุมตำแหน่งของมอเตอร์ ---------- 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 3 การออกแบบและการสร้าง ---------- **23**

3.1 แบบโครงสร้างและชิ้นส่วนประกอบ ---------- 23

3.2 อุปกรณ์และส่วนประกอบในการทำแบบจำลองการขนส่งในไลน์การผลิต ---------- 25

3.2.1 ชั้นวางกล่องชิ้นส่วนประกอบ ---------- 25

3.2.2 เครนอัตโนมัติ ---------- 26

3.3 แผงวงจรควบคุม ---------- 29

3.4 การใช้ Visual Basic ในการอินเตอร์เฟส ---------- 31

3. 5 รถขนส่งชิ้นส่วนอัตโนมัติ ---------- 37

3.6 อินฟราเรดเซนเซอร์ ---------- 38

3.7 แนวทางการออกแบบและการทำงานของตัวรถขนส่งชิ้นส่วน ---------- 42

3.8 การทำงานงานของอินฟราเรดเซนเซอร์ ---------- 43

บทที่ 4 การทดลอง ---------- **49**

4.1 การคำนวณระยะการเคลื่อนที่ ---------- 49

บทที่ 5 สรุปผลการทดลองและบทวิจารณ์ ---------- **55**

5.1 สรุปผลการทดลอง ---------- 55

5.2 แนวทางการพัฒนา ---------- 55

5.3 ปัญหาในการทำงาน ---------- 56

ภาคผนวก ---------- **57**

ก โปรแกรมควบคุมเครนอัตโนมัติ ---------- 58

ข โปรแกรมรถขนส่งรถขนส่ง ---------- 79

ค โปรแกรมVishual Basic 6.0 ---------- 87

เอกสารอ้างอิง ---------- **92**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1.1 แผนผังการทำงานของแบบจำลองการขนส่งในไลน์การผลิต ---------- 3

รูปที่ 2.1 รูปแบบสัญญาณไฟฟ้าของข้อมูล ซึ่งเป็นการส่งข้อมูลแบบอนุกรม ---------- 10
ด้วยอัตราเร็ว 2400 บิต/วินาที

รูปที่ 2.2 แผนภาพสัญญาณเวลาของข้อมูลแบบอนุกรมจำนวน 8 บิต ---------- 11
พร้อมด้วยบิตเริ่มต้นบิตพาริตี้ และ บิตสุดท้าย

รูปที่ 2.3 แผนภาพแสดงการทำงานของวงจรส่วนการรับและส่งข้อมูลอนุกรมของ 8051 ----- 12

รูปที่ 2.4 ไอซีเบอร์MAX232ซึ่งเป็นวงจรเชื่อมต่อแบบRS-232C ---------- 13
โดยการใช้ไฟเลี้ยง +5 เพียงชุดเดียว

รูปที่ 2.5 การเชื่อมต่อขาสัญญาณ ---------- 14

รูปที่ 2.6 วงจรภายในของมอเตอร์กระแสตรง ---------- 15

รูปที่ 2.7 แสดงอินพุตและเอาต์พุตของโมเดลทางคณิตศาสตร์ของมอเตอร์ ---------- 16

รูปที่ 2.8 แสดงโมเดลของดีซีมอเตอร์แบบฟิลด์แยกกระตุ้น ---------- 17

รูปที่ 2.9 แสดงถึงแรงบิดต่างๆที่เกิดขึ้นต่อโหลดของมอเตอร์ ---------- 18

รูปที่ 2.10 แสดงบล็อกไดอะแกรมของดีซีมอเตอร์โมเดล ---------- 19

รูปที่ 2.11 แสดง Rotary Encoder ---------- 22

รูปที่ 2.12 แสดงการสร้าง Pulse ของ Rotary Encoder ---------- 22

รูปที่ 3.1 แบบโครงสร้างชั้นวางชิ้นส่วนประกอบ ---------- 23

รูปที่ 3.2 แบบโครงสร้างแนวแกน X และ Y ---------- 23

รูปที่ 3.3 แบบโครงสร้างฐานตัวจับ ---------- 24

รูปที่ 3.4 แบบโครงสร้างตัวรถขนส่งชิ้นส่วนประกอบ ---------- 24

รูปที่ 3.5 แบบจำลองโครงสร้างทังหมดของแบบจำลอง ---------- 24

รูปที่ 3.6 (ก) ภาพด้านหน้าแสดงขนาดชั้นรับส่งชิ้นส่วนประกอบ ---------- 25

(ข) ภาพด้านข้างแสดงขนาดชั้นรับส่งชิ้นส่วนประกอบ ---------- 25

รูปที่ 3.7 แสดงโครงสร้างของชั้นรับส่งชิ้นส่วนประกอบ ---------- 26

รูปที่ 3.8 ชุดเคลื่อนที่ในแนวแกน X ---------- 27

รูปที่ 3.9 ชุดการเคลื่อนที่แนวแกน Y ---------- 28

รูปที่ 3.10 ชุดการเคลื่อนที่แนวแกน Z ---------- 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

หน้า

รูปที่ 3.11 ภาพแสดงชั้นวางกล่องชิ้นส่วนประกอบ และเครนอัตโนมัติ ---------- 29
รูปที่ 3.12 แผงวงจรควบคุม ---------- 29
รูปที่ 3.13 วงจรควบคุมเครนอัตโนมัติ ---------- 30
รูปที่ 3.14 ฟอร์ม 1 เพื่อการอินเตอร์เฟสบนคอมพิวเตอร์ ---------- 31
รูปที่ 3.15 ฟอร์ม 2 เพื่อการยืนยันรายการและไลน์การผลิตที่จะส่ง ---------- 32
รูปที่ 3.16 ฟอร์ม 2 เพื่อการแสดงความผิดพลาดของรายการที่สั่ง ---------- 32
รูปที่ 3.17 แผนผังความคิดการทำงานของโปรแกรม Visual Basic ---------- 34
รูปที่ 3.18 แผนผังความคิดการทำงานของเครนอัตโนมัติ ---------- 36
รูปที่ 3.19 วงจร Micro-controller, วงจร Drive Motor และ วงจรลดแรงดัน LM2940 --------- 38
รูปที่ 3.20 วงจรตรวจจับแสงสะท้อนอินฟราเรดที่ใช้สำหรับเดินตามเส้น ---------- 39
รูปที่ 3.21 วงจรตรวจจับแสงสะท้อนอินฟราเรดที่ใช้สำหรับตรวจจับของที่วางบนรถ -------- 40
รูปที่ 3.22 วงจรรับสัญญาณให้เคลื่อนที่จากจุดรับของและวงจรรับสัญญาณให้หยุดจาก ------ 40
ไลน์การผลิต
รูปที่ 3.23 ลายวงจร sensor เดินตามเส้น ---------- 41
รูปที่ 3.24 ลายวงจร sensor ตรวจเช็คกล่อง ---------- 41
รูปที่ 3.25 ลายวงจร sensor รับคำสั่ง ---------- 41
รูปที่ 3.26 ลายวงจร sensor สำหรับหยุด ---------- 41
รูปที่ 3.37 โครงสร้างและส่วนประกอบของตัวรถขนส่งชิ้นส่วนประกอบ ---------- 42
รูปที่ 3.28 การรับส่งสัญญาณของอินฟราเรดเซนเซอร์ป้อนคำสั่ง ---------- 43
รูปที่ 3.29 แสดงการรับส่งสัญญาณของอินฟราเรดเซนเซอร์ตรวจเช็คของ ---------- 43
รูปที่ 3.30 แสดงการรับส่งสัญญาณของ sensor ที่ใช้หยุด ---------- 44
รูปที่ 3.31 แสดงการวางตำแหน่งของเซนเซอร์ตรวจจับเส้น ---------- 44
รูปที่ 3.32 แสดงลักษณะเส้นทางที่ใช้ในการตรวจจับ ---------- 45
รูปที่ 3.33 แสดงลักษณะเส้นทางที่เซนเซอร์ที่ต้องเจอ ---------- 46
รูปที่ 3.34 แผนผังความคิดการทำงานของรถขนส่งอัตโนมัติ ---------- 48
รูปที่ 4.1 แสดงตำแหน่งชั้นวางชิ้นส่วนประกอบ ---------- 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 1.1 แสดงช่วงเวลาการทำงาน ---------- 4
ตารางที่ 1.2 แสดงStep เวลาการทำงาน ---------- 4
ตารางที่ 2.1 แสดงโหมดการทำงานของพอร์ตอนุกรม ---------- 12
ตารางที่ 4.1 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y ครั้งที่ 1 - 50
ตารางที่ 4.2 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y ครั้งที่ 2 - 51
ตารางที่ 4.3 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y ครั้งที่ 3 - 52
ตารางที่ 4.4 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y (ค่าเฉลี่ย) - 53
ตารางที่ 4.6 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ --- 54
ครั้งที่ 1
ตารางที่ 4.7 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ ---- 54
ครั้งที่ 2
ตารางที่ 4.8 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ ---- 54
ครั้งที่ 3
ตารางที่ 4.9 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ ---- 54
โดยเฉลี่ย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 กล่าวนำ

ในปัจจุบัน การคอนโทรลมอเตอร์กระแสตรงหรือดีซีมอเตอร์(DC Motor) จะพบได้โดยทั่วไป โดยเฉพาะในงานอุตสาหกรรมสมัยใหม่ส่วนมาก ด้วยความเจริญก้าวหน้าของเทคโนโลยีทางไมโครอิเลคทรอนิคส์ (Microelectronic) และ ไมโครโปรเซสเซอร์ (Microprocessor) ทำให้การออกแบบระบบคอนโทรลมอเตอร์กระแสตรงและการวิเคราะห์ปัญหาเป็นสิ่งที่น่าสนใจและมีความสำคัญมากระบบการคอนโทรลแบบดั้งเดิมมีพลังจักรกลที่สำคัญคือพวกมอเตอร์ไฟฟ้าต่างๆไฮโดรลิกและพวกเบรกและครัช(clutch) เป็นต้น แต่ด้วยความเจริญของเทคโนโลยีการสร้างแม่เหล็กถาวรที่มีคุณภาพสูง ทำให้มอเตอร์กระแสตรงกลายเป็นพลังจักรกลที่สำคัญในระบบคอนโทรลเกือบทุกชนิดในปัจจุบัน ซึ่งพบเห็นได้ตั้งแต่การใช้มอเตอร์กระแสตรงในเครื่องใช้ไฟฟ้าภายในบ้านโดยปกติ การใช้ในรถยนต์ การใช้ในเครื่องคอมพิวเตอร์ การใช้ในหุ่นยนต์ตลอดถึงเครื่องจักรกลที่ทำงานแบบอัตโนมัติในอุตสาหกรรมต่างๆ ฯลฯ บวกกับความก้าวหน้าของไมโครโปรเซสเซอร์ สามารถนำมาประยุกต์ใช้กับการคอนโทรลมอเตอร์กระแสตรงได้ จึงทำให้ได้รับความนิยมอย่างมากในปัจจุบัน

มอเตอร์กระแสตรงมีผลการทำงานที่ไม่มีตำแหน่งหยุดที่แน่นอน ดังนั้นระบบคอนโทรลมอเตอร์กระแสตรงจึงเป็นแบบวงรอบปิด ตำแหน่งเอาท์พุตหรือความเร็วเอาท์พุทจะถูกป้อนกลับไปเปรียบเทียบกับอินพุตอ้างอิงเพื่อให้ได้ผลการทำงานตามที่ต้องการ เช่นการบบังคับแขนหุ่นยนต์ เมื่อจับวัตถุขึ้นมาชิ้นหนึ่ง เราจะควบคุมให้แขนหุ่นยนต์เคลื่อนที่ไปอีกจุดหนึ่งแล้ววางวัตถุลงหรือใช้ในการส่งของจากที่หนึ่งไปยังอีกที่หนึ่ง เป็นต้นซึ่งโครงงานนี้จะเป็นการออกแบบและสร้างระบบควบคุมตำแหน่งของดีซีมอเตอร์ทีควบคุมตำแหน่งของเครนอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์ตระกูล 8051 โดยตำแหน่งที่ต้องการจะถูกป้อนเข้าไปให้กับไมโครคอนโทรเลอร์ทางพอร์ตอนุกรม โดยจะใช้โปรแกรมอินเตอร์เฟส(Visual Basic) เป็นตัวติดต่อรับส่งค่า ตรงส่วนนี้จะเป็นจุดที่เริ่มต้นของการขนส่งชิ้นส่วนประกอบ เพื่อยกชิ้นส่วนจากชั้นวางเพื่อส่งให้รถวิ่งเข้าไปในสายพานการผลิตตามที่โปรแกรมที่ป้อนคำสั่งไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของโครงงาน

จากประสบการณ์การฝึกงานของกลุ่มผู้ทำโครงงานนี้ ได้มีโอกาสเข้ารับการฝึกงานที่โรงงานประกอบชุดสายไฟรถยนต์ ซึ่งมีลักษณะในการประกอบโดยการแบ่งเป็นลายการผลิต ซึ่งแต่ละลายการผลิตก็จะทำการประกอบชุดสายไฟรถยนต์ตามใบสั่ง และเนื่องจากระยะทางจากที่จัดเก็บชิ้นส่วนประกอบกับลายการผลิต มีระยะที่ไกลพอสมควร จึงจำเป็นที่จะต้องมีการขนส่งชิ้นส่วนประกอบต่างๆ จากที่จัดเก็บชิ้นส่วนโดยอาศัยรถเข็นในการส่ง รถเข็นแต่ละคันจะมีผู้เข็นหนึ่งคน และเนื่องจากชิ้นส่วนที่จำเป็นต้องขนส่งในมีจำนวนมาก จึงทำให้โรงงานนั้นต้องจ้างพนักงานเข็นรถเป็นจำนวนหลายสิบคน ซึ่งเป็นการสิ้นเปลืองแรงงาน ด้วยเหตุนี้จึงทำให้ผู้จัดทำโครงงานได้เกิดความคิดที่จะทดลองออกแบบแบบจำลองการขนส่งชิ้นส่วนประกอบต่างๆ ขึ้นมา

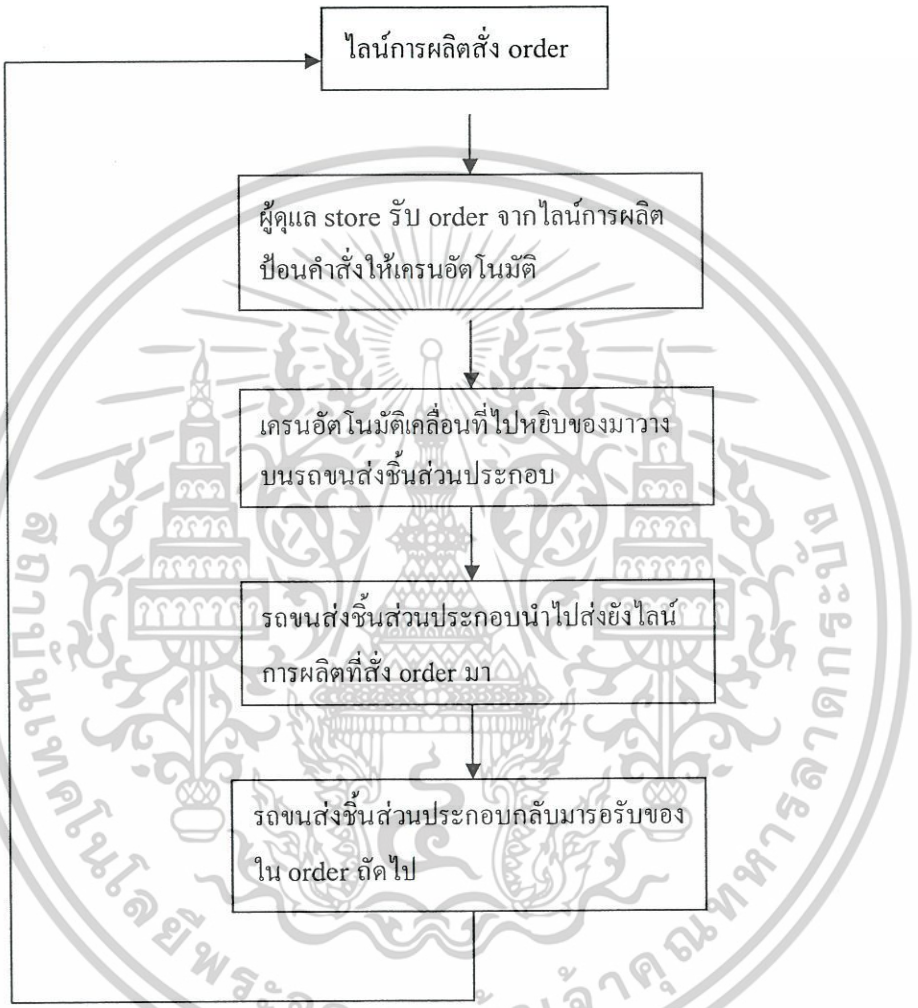
แบบจำลองระบบการขนส่งในไลน์การผลิต สามารถทำงานได้อย่างเป็นระบบง่ายต่อการปรับปรุงแก้ไขเปลี่ยนแปลง การขนส่งจะประกอบด้วยการสั่งการผ่านหน้าจอคอมพิวเตอร์ในการเลือกชิ้นส่วนประกอบที่ต้องการ ซึ่งติดตั้งอยู่บริเวณที่จัดเก็บชิ้นส่วนประกอบ การขนส่งจะกระทำโดยรถขนส่งที่ออกแบบมาให้รับคำสั่งผ่านหน้าจอคอมพิวเตอร์และทำการส่งชิ้นส่วนประกอบโดยอัตโนมัติ โดยขั้นตอนทั้งหมาดนี้จะอาศัยพนักงานเพียงคนเดียว จึงทำให้การขนส่งสามารถทำได้อย่างรวดเร็ว ประหยัดการลงทุน และบุคลากร ซึ่งทางโรงงานสามารถคืนทุนได้อย่างรวดเร็ว ยิ่งในกรณีที่มีการทำงาน 3 กะยิ่งคืนทุนได้เร็วขึ้น และจะเป็นการนำระบบการทำงานที่ควบคุมด้วยคอมพิวเตอร์มาใช้งานทำให้ได้มาตรฐานซึ่งจะเป็นสิ่งที่ทำให้ลูกค้าเกิดความพอใจมากยิ่งขึ้น

1.3 ขอบเขตโครงงาน

การทำแบบจำลองการขนส่งในไลน์การผลิต ที่ใช้ศึกษาในการขนส่งวัสดุอุปกรณ์หรือชิ้นส่วนที่ใช้ ในการประกอบผลิตภัณฑ์ในไลน์การผลิต (รูปแบบของที่ใช้ศึกษาในการขนส่งจะมีลักษณะเป็นกล่องทรงสี่เหลี่ยม)โดยมีรูปแบบโครงสร้างประกอบไปด้วย ชั้นวางกล่องชิ้นส่วนประกอบ เครนอัตโนมัติ และรถขนส่งชิ้นส่วนประกอบ โดยชั้นวางกล่องชิ้นส่วนประกอบจะมีหน้าที่เป็นชั้นวางใน store ที่มีการเก็บเป็นช่องๆ เพื่อความสะดวกในการใช้เครนอัตโนมัติในการขนส่ง ซึ่งเครนอัตโนมัติจะทำหน้าที่ในการรับคำสั่งจากผู้ควบคุมดูแล store ที่จะป้อนคำสั่งให้เครนอัตโนมัติเคลื่อนที่เข้าไปหยิบกล่องชิ้นส่วนประกอบออกจากชั้นวางกล่องชิ้นส่วนประกอบ เพื่อนำออกมาวางบนรถขนส่งในไลน์การผลิตโดยอัตโนมัติ พร้อมทั้งแสดงผลของการทำงานออกมาให้ผู้ควบคุมดูแล store เห็นว่ามีการหยิบของตรงกับการป้อนคำสั่งหรือไม่ และจำนวนของที่เหลืออยู่ใน store เท่าไร ส่วนของตัวรถขนส่งชิ้นส่วนประกอบจะทำหน้าที่ในการขนส่งชิ้นส่วนประกอบจากจุดรับของจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

store ไปส่งยังไลน์การผลิตต่างๆที่ได้รับคำสั่งมาจากจุดรับของที่ store โดยการเคลื่อนที่ตามเส้นที่ได้ตั้งไว้อัตโนมัติ พอส่งของเสร็จแล้วก็จะเคลื่อนที่กลับมายังจุดรับของที่ store เองโดยอัตโนมัติเพื่อรอรับคำสั่งในการขนส่งต่อไป โดยมีรูปแบบการทำงานดังนี้



รูปที่ 1.1 แผนผังการทำงานของแบบจำลองการขนส่งในไลน์การผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ตารางแผนการทำงาน

ตารางที่ 1.1 แสดงช่วงเวลาการทำงาน

Mount – year step	7 - 51 (ก.ค.)	8 – 51 (ส.ค.)	9 – 51 (ก.ย.)	10 – 51 (ต.ค.)	11 – 51 (พ.ย.)	12 – 51 (ธ.ค.)	1 – 51 (ม.ค.)	2 – 51 (ก .พ.)
Step1.								
Step 2.								
Step 3.								
Step 4.								
Step 5.								
Step 6.								

ตารางที่ 1.2 แสดงStep เวลาการทำงาน

Retail	
Step 1	ค้นคว้าข้อมูล
Step 2	ออกแบบโครงสร้าง
Step 3	จัดหาอุปกรณ์
Step 4	สร้างโครงสร้างและทดสอบ
Step 5	ทำการโปรแกรมและทดสอบ
Step 6	ทดสอบ, แก้ไขปรับปรุง และ สรุปผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1 Microprocessor 8051

ในไมโครคอนโทรลเลอร์เบอร์ 8051 ประกอบด้วยพอร์ตอินพุตเอาท์พุตแบบ 2 ทิศทาง (Bidirectional) จำนวน 4 พอร์ต แต่ละพอร์ตมีอุปกรณ์แลทช์ข้อมูล และเอาท์พุตไดรเวอร์ประกอบอยู่ทางด้านเอาท์พุต และทางด้านอินพุตจะมีบัฟเฟอร์ (พอร์ตทั้ง 4 เป็นรีจีสเตอร์พิเศษชื่อ P0, P1, P2 และ P3) เราสามารถใช้งานแต่ละพอร์ตเป็นอินพุต หรือเอาท์พุตได้ตามต้องการ แต่ละบิตของพอร์ตสามารถเชื่อมต่อกับสัญญาณ TTL ได้โดยตรง

ในการติดต่อกับหน่วยความจำภายนอกพอร์ต P0 และ P2 จะใช้สำหรับกำหนดตำแหน่งของหน่วยความจำภายนอก โดยพอร์ต P0 จะทำงานในลักษณะของมัลติเพล็ก คือเป็นทั้งพอร์ตตำแหน่งและ พอร์ตข้อมูล โดย P0 จะเป็นตำแหน่งของหน่วยความจำด้านต่ำ (Low byte) และ P2 จะเป็นตำแหน่งของหน่วยความจำด้านสูง (High byte)

สัญญาณต่างๆ ของไมโครคอนโทรลเลอร์เบอร์ 8051 สามารถจำแนกตามการทำงานเป็น 3 กลุ่มคือ

กลุ่มสัญญาณตำแหน่ง : เป็นตัวกำหนดตำแหน่งของหน่วยความจำ

กลุ่มสัญญาณควบคุม : เป็นสัญญาณควบคุมการทำงานของไมโครคอนโทรลเลอร์

กลุ่มสัญญาณข้อมูล : เป็นทางผ่านของข้อมูลระหว่างไมโครคอนโทรลเลอร์กับหน่วยความจำ

2.1.1 โครงสร้างภายในไมโครคอนโทรลเลอร์เบอร์ 8051

- หน่วยความจำภายในสำหรับข้อมูลขนาด 128 ไบต์
- หน่วยความจำสำหรับเก็บโปรแกรมขนาด 4 กิโลไบต์
- อุปกรณ์ควบคุมอินเตอร์รัพท์
- ตัวตั้งเวลาและตัวนับขนาด 16 บิต 2 ชุด
- พอร์ตควบคุมการสื่อสารอนุกรมแบบ Full Duplex ซึ่งสามารถรับและส่งข้อมูลพร้อมกันได้
- พอร์ตขนานสำหรับติดต่อกับอุปกรณ์ภายนอกจำนวน 4 พอร์ตๆ ละ 8 บิต
- วงจรผลิตสัญญาณนาฬิกาภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ตัวตั้งเวลาและตัวนับ (Timer/Counter)

ไมโครคอนโทรลเลอร์เบอร์ 8051 ประกอบด้วยรีจีสเตอร์ ตัวตั้งเวลา/ตัวนับ ขนาด 16 บิต จำนวน 2 ชุด คือ Timer0 และ Timer1 โดย Timer ทั้งหมดสามารถกำหนดให้ทำงานในลักษณะ ของตัวนับหรือตัวจับเวลาก็ได้

การทำงานในโหมดของตัวตั้งเวลา ค่าของรีจีสเตอร์จะเพิ่มขึ้นทุกๆแมชชีนไซเคิล โดย 1 แมชชีนไซเคิล ประกอบด้วยสัญญาณนาฬิกา 12 ลูก ดังนั้นอัตราการจับเวลาจะเป็น 1/12 เท่าของความถี่สัญญาณนาฬิกา ของระบบ ค่าสูงสุดที่ตั้งได้คือ 2 ยกกำลัง 16

การทำงานในโหมดการนับ ค่าของการนับจะเพิ่มขึ้น เมื่อสัญญาณเข้ามาที่ขา T0 และ T1 เปลี่ยนจาก 1 เป็น 0 ความเร็วในการนับสูงสุดคือ 1/24 เท่าของสัญญาณนาฬิกา โดยสัญญาณที่เข้ามาที่ขา T0 หรือ T1 จะมี Duty Cycle เท่าใดก็ได้

2.1.3 8051 กับการ interrupt วงจรนับ/จับเวลา

<u>วงจรนับ/จับเวลา</u>

8051 ประกอบด้วย register ขนาด 16 bit จำนวน 2 ตัว คือ T0 (Timer0) และ T1 (Timer1) ซึ่งสามารถนำไปใช้งานได้อย่างอิสระ โดยสามารถควบคุมให้ทำหน้าที่เป็นตัวจับเวลา)Timer) เพื่อนับจำนวน plus สัญญาณนาฬิกาภายใน หรือควบคุมให้ทำหน้าที่เป็นตัวนับ)Counter) เพื่อนับจำนวน plus ของระบบ ได้ ภายใน register แต่ละตัวยังสามารถแยกออกได้เป็น register ขนาด 8 bit คือ TH0, TL0, TH1 และ TL1 โดยการทำงานของ register ทั้ง 2 ตัวนี้มีผลมาจากการกำหนดค่าของ bit ที่อยู่ภายใน TMOD (Timer mode control register) และ TCON (Timer/Counter control register), bit ต่างๆภายใน register TMOD, bit ต่างๆภายใน register TCON

<u>การ Interrupt วงจรนับตรวจ/จับเวลา</u>

จากกระบวนการทำงานของวงจรนับ/จับเวลาของ 8051 จำเป็นต้องกำหนดค่าเริ่มต้นให้กับ register T0 หรือ T1 ค่านี้เป็นค่าจำนวน plus ภายในที่จะต้องนับหรือค่าของจำนวน plus ภายนอก ที่เข้ามาทางขาสัญญาณสัญญาณ T0 หรือ T1 ค่าตัวเลขภายใน register นี้จะต้องลดให้มีค่าที่น้อยกว่าค่าที่ต้องการอยู่หนึ่งค่า ทั้งนี้เนื่องจากการทำงานของ register จะเพิ่มค่าจากที่กำหนดไปเรื่อยๆ จนถึงค่าสูงสุดของ register และกลับไปเป็นค่า 0 เมื่อมีการเกิด Overflow เกิดขึ้น ทำให้เกิดการกำหนดค่า flag เพื่อแจ้งให้ CPU ได้รับทราบ ดังนั้นโปรแกรมทั่วไปจึงมักใช้สภาวะของ flag นี้)TF0 และ TF1) ซึ่งเป็น bit อยู่ภายใน register TCON เพื่อตรวจสอบว่ากระบวนการนั้นได้เสร็จสิ้นลงแล้ว หรือใช้เพื่อทำการ interrupt program ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 ส่วนควบคุมการทำงานของวงจรนับ/จับเวลา

ซึ่งประกอบด้วยส่วนของการกำหนดที่มาของสัญญาณ) Timer) หรือ) Counter) และ bit หรือขาสัญญาณสำหรับการหยุดหรือทำงานของวงจรนับ

การทำงานเป็นตัวจับเวลาต้องมีการกำหนดค่าให้อยู่ในสภาวะดังตารางก่อน

การกำหนดค่า bit	Register	ความหมาย
C/T = 0	TMOD	กำหนดให้เป็นการจับเวลา
TR1 หรือ TR0 = 1	TCON	กำหนดสภาวะให้มีการทำงาน
GATE = 0 หรือระดับขาสัญญาณที่ขา INT0 หรือ INT1 เป็นสภาวะlogic สูง	TMOD	กำหนดสภาวะควบคุมการทำงาน

8051 Timer/Counter

TMOD and TCON Registers

Timer/Counter 0				Timer/Counter 1			
C/T#	M1	M0	GATE#	C/T#	M1	M0	GATE#

TMOD

Timer/Counter 0				Timer/Counter 1			
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TCON

Baud Rate

$$\text{Baud Rate} = \frac{2^{SMOD} \times f}{384 \times (256 - TH1)}$$

Baud Rate	f	SMOD	TH1
62.5K	12.000	1	FFh
19.2K	11.059	1	FDh
9.6K	11.059	0	FDh
4.8K	11.059	0	FAh
2.4K	11.059	0	F4h
1.2K	11.059	0	E8h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การ Initial ค่า Baud rate

Set baud rate to 9600 for 11.0592 Crystal

```
mov TMOD, #20h ; set timer 1 for auto reload
mov TCON, #41h ; run counter 1 edge trig
mov TH1, #0FDh ; 9600 baud
mov SCON, #50h ; 8-bit data mode 1
```

Sending and Receiving Subroutines

```
sndchr:                 getchr:
    clr TI                  jnb RI, getchr
    mov SBUF, A             mov A, SBUF
    txloop: jnb TI, txloop  clr RI
    ret                     ret
```

- การจับเวลาใน Mode 0: การทำงานใน mode 0 วงจรนับจับเวลาจะทำหน้าที่เป็นตัวนับขนาด 13 bit(โดยใช้ register TH0 หรือ TH1 เป็นตัวนับขนาด 8 bit และ register TL0 หรือ TL1 มีขนาด 5 bit) ตามลักษณะของ แผนภาพในรูป
- การจับเวลาใน Mode 1: การทำงานใน mode 1 มีความคล้ายคลึงใน mode 0 มาก แตกต่างกันที่ mode 1 เป็นตัวนับขนาด 16 bit เต็ม ดังรูป
- การจับเวลาใน Mode 2: การทำงานใน mode 2 ของวงจรนับ/จับเวลาแตกต่างกันออกไปเพียงใช้ register TL0 (TL1) เป็น ตัวนับขนาด 8 bit ส่วน register TH0 (TH1) เก็บค่าเริ่มต้นของการนับไว้
- การจับเวลาใน mode 3 : การทำงานใน mode 3 จะสามารถใช้ได้เฉพาะกับ Timer 0 เท่านั้น หากว่านำไปกำหนดให้กับ Timer 1 จะทำให้หยุดการทำงานไป เมื่อ Timer 0 ได้รับการกำหนดทำงานใน mode 3 จะมีผลทำให้ register ของมันแยกการทำงานเป็นอิสระ โดย register TL0 จะถูกควบคุมจาก bit ภายใน register TCON และ ขาสัญญาณ INT0 ดังแสดงในรูป และเมื่อมีการ Overflow เกิดขึ้น จากค่า 0FFH เป็น 00H ก็ จะมีผลให้ flag TF0 มีการเปลี่ยนแปลงเกิดขึ้น สำหรับ register TH0 จะถูกกำหนดให้ทำงานในแบบ ของตัวจับเวลาภายใต้การควบคุมของ bit TR1 ใน register TCON เท่านั้น และหากเกิด Overflow จะมีผลเฉพาะต่อ flag TF1 ในส่วน Timer 1 ขณะเมื่อ Timer 0 ถูกกำหนดให้ทำงาน mode 3 ก็ยังจะสามารถทำงานใน mode อื่นๆ ที่ไม่ใช้ mode 3 ได้ เช่นเดิม ยกเว้นจะไม่มีการ interrupt เกิดขึ้นเท่านั้น) เนื่องจาก flag TF1 ถูก ใช้โดย Timer 0 ไปแล้ว (รูปการทำงานใน mode 3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 การทำงานเป็นตัวนับสัญญาณ

การใช้งานในลักษณะตัวนับ(Counter) โดยหลักแล้วจะเหมือนกับลักษณะการทำงานเป็น ตัวจับเวลา)Timer) ดังได้กล่าวในหัวข้อที่ผ่านมา ข้อแตกต่างประการเดียวคือ แทนที่จะนับplusสัญญาณภายในและผ่านวงจรหาร 12 มาเป็นการนับ plus สัญญาณทางขาสัญญาณ T0 (P3.4) ให้กับ Timer0 หรือขาสัญญาณ T1 (P3.5) ให้กับ Timer1 เท่านั้น นอกจากนี้ก่อนการเริ่มต้นใช้งานจะ ต้องกำหนดค่าของ bit C/T ภายใน register TCON ให้มีค่าเป็น 1 เสียก่อน

<u>วงจรนับ/จับเวลา2(Timer2)</u>

Mode การทำงานของ Timer2 ประกอบด้วย

Capture mode: สามารถเลือกใช้งานได้ 2 ลักษณะ ด้วยการกำหนดให้กับ bit EXEN2 ของ register T2CON ดังนี้ 1. เมื่อกำหนด bit EXEN2 เป็น 0 Timer 2 ยังทำงานเป็นวงจรนับ/ตรวจจับเวลา เมื่อมีการ overflow ขึ้น bit ใน register TF2 จะถูกเซต และสามารถนำไปสร้างการ interrupt ขึ้นได้ 2. เมื่อกำหนดค่า bit EXEN2 เป็น 1 การทำงานจะครอบคลุมการทำงานลักษณะข้างต้น แต่จะเพิ่มเมื่อมีการเปลี่ยนแปลงระดับ สัญญาณทางขาสัญญาณ T2EX จาก logic สูง ไปเป็น logic ต่ำ จะมีผลทำให้ค่าข้อมูลภายใน register ของ Timer2 คือ TL2 และ TH2 ถูกนำไปใส่)Capture) ให้กับ register RCAP2L และ RCAP2H ซึ่งเป็น register หน้าที่พิเศษ หรือ SFR ที่มีใน Microcontroller เบอร์ 8052 เท่านั้นนอกจากนี้จะมีผลทำให้ bit EXF2 ภายใน register T2CON มีค่าเป็น 1 สามารถนำไปใช้งานในการinterrupt ได้เช่นกัน

Auto-reload mode: สามารถทำงานได้ 2 ลักษณะเช่นเดียวกัน

Baud rate Generator: ของ Timer 2 จะมีความแตกต่างจาก Timer 0 และ Timer 1 โดยวงจรรับ และการส่ง สามารถเป็นค่าที่ต่างกันได้ ขึ้นอยู่กับการกำหนดค่าให้กับ bit TCLK และ RCLK ของ Timer 2 ดังแสดงให้เห็นจากภาพการทำงาน

การทำงานของโหมดนี้คล้ายคลึงกับการทำงานใน Auto-reload mode กล่าวคือค่าใน register TH2 เปลี่ยนแปลงจากค่า 0FFH ไปเป็นค่า 0 หรือที่เรียกว่า overflow จะมีผลให้มีการโหลดข้อมูลขนาด 16 bit จาก register RCAP2H และ RCAP2L ซึ่งมีการเตรียมค่าล่วงหน้าแล้วโดยอัตโนมัติ

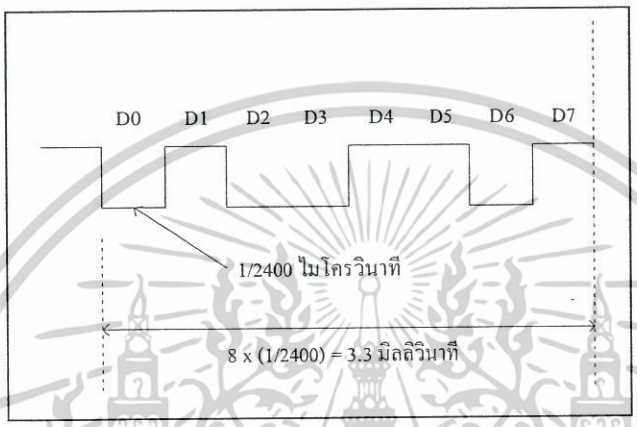
2.1.7 การสื่อสารข้อมูลแบบอนุกรม (SERIAL PORT COMMUNICATION)

การสื่อสารข้อมูลอนุกรมเป็นการรับหรือส่งข้อมูลในลักษณะของบิตหรือกลุ่มของบิต คราวละหนึ่งบิตเป็นลำดับเรื่อยไปจนสิ้นสุด

<u>จังหวะเวลาของการสื่อสารข้อมูลอนุกรม</u>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปมักจะระบุกันในหน่วยของจำนวนบิตข้อมูลภายในเวลาหนึ่งวินาที เรียกว่า อัตราบอด ตามค่ามาตรฐานเหล่านี้ ได้แก่ 110, 150, 300, 1200, 2400, 4800, และ 9600 จากรูปที่ 20 แสดงให้เห็นลักษณะของรูปแบบสัญญาณข้อมูลอนุกรมที่ปรากฏในสายส่งสัญญาณ ข้อมูลทั้ง 8 บิตนี้หากว่าถูกส่งออกมาด้วยอัตรา 2400 บอดจะใช้เวลาในการส่งข้อมูลหนึ่งบิตมีค่าเท่ากับ 1/2400 หรือ 416 ไมโครวินาที และเวลาในการส่งข้อมูลทั้ง 8 บิตมีค่าเท่ากับ 8x416 หรือ 3,328 ไมโครวินาที



รูปที่ 2.1 รูปแบบสัญญาณไฟฟ้าของข้อมูล ซึ่งเป็นการส่งข้อมูลแบบอนุกรมด้วยอัตราเร็ว 2400 บิต/วินาที (สังเกตว่าจะเริ่มส่งจากบิต D0 ซึ่งเป็นบิตนัยสำคัญต่ำออกมาก่อนเป็นลำดับแรก)

<u>รูปแบบของข้อมูลอนุกรม</u>

วิธีการที่จะทำให้ข้อมูลสื่อสารอนุกรมมีความถูกต้องมากยิ่งขึ้น จะใช้การเพิ่มเติมบิตข้อมูลบางอย่างร่วมไปกับการส่งข้อมูลจริง ได้แก่

1. บิตเริ่มต้น (Start Bit)

มีหน้าที่สำหรับการบ่งบอกให้วงจรฮาร์ดแวร์ทางด้านรับทราบถึง ตำแหน่งจุดเริ่มต้นของบิตข้อมูลกลุ่มใหม่ เพื่อที่จะทำการปรับจังหวะของสัญญาณการรับข้อมูลให้ตรงกัน ดังนั้นบิตเริ่มต้นนี้จึงจะถูกเพิ่มเข้าไปก่อนระดับลอจิกของสภาวะของสายสื่อสาร ขณะเมื่อไม่มีการส่งข้อมูล

2. บิตแสดงภาวะความเป็นเลขคู่หรือเลขคี่ (Parity Bit)

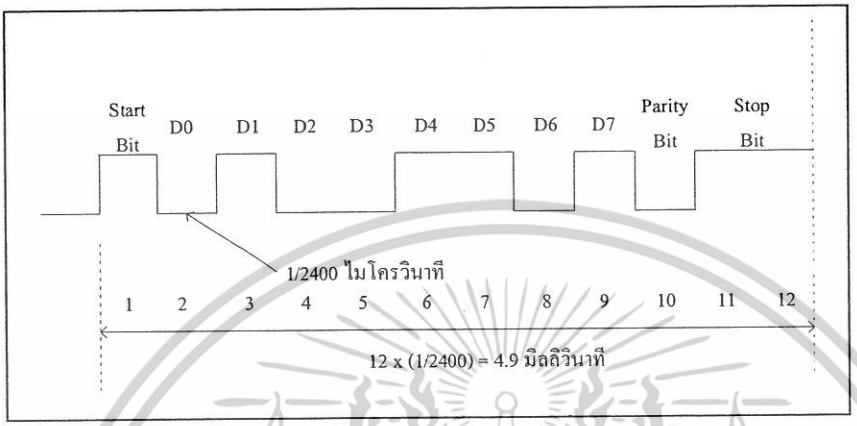
มีหน้าที่เพื่อตรวจสอบความถูกต้องของข้อมูล โดยทั่วไปมักเรียกว่า บิตพาริตี้และจะนำไปแทรกต่อท้ายบิตข้อมูล ค่าของบิตนี้ขึ้นอยู่กับจำนวนค่าของบิตข้อมูลที่เป็น 1 ซึ่งจะเป็นได้สองลักษณะ คือ พาริตี้คู่ (Even Parity) หรือ พาริตี้คี่ (Odd Parity)

3. บิตสุดท้าย (Stop Bit)

เป็นบิตที่เพิ่มเติมขึ้นเพื่อระบุถึงขอบเขตการสิ้นสุดของกลุ่มบิตข้อมูลบิตสุดท้ายนี้อาจจะมีจำนวนมากกว่าหนึ่งบิตได้ คือ 1.5 บิต และ 2 บิต ดังนั้นกรณีของการส่งข้อมูล 8 บิตพร้อมบิตที่เพิ่มเติมเข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยสมบูรณ์ คือ บิตเริ่มต้น บิตพาริตี้ และ บิตสุดท้าย รวมทั้งสิ้น 12 บิต ตามแผนภาพสัญญาณเวลาในรูปที่ 2 หากข้อมูลถูกส่งออกไปด้วยอัตราเร็ว 2400 บอด เวลาโดยรวมในการส่งข้อมูลหนึ่งไบต์จะมีค่าเป็น 12 x 416 ไมโครวินาที หรือ 4.99 มิลลิวินาที

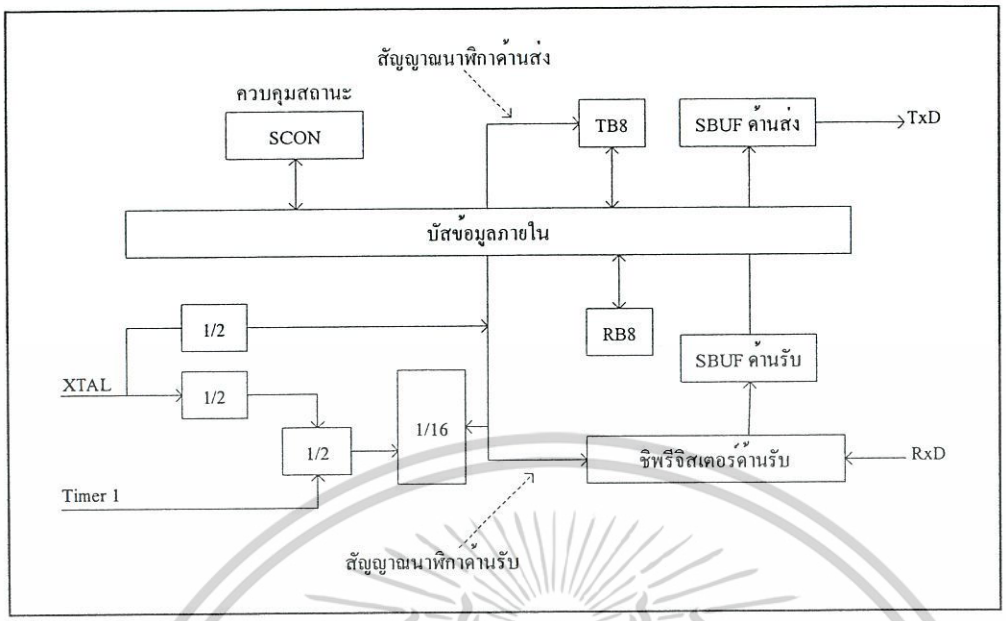


รูปที่ 2.2 แผนภาพสัญญาณเวลาของข้อมูลแบบอนุกรมจำนวน 8 บิต พร้อมด้วยบิตเริ่มต้นบิตพาริตี้ และ บิตสุดท้าย

2.1.8 การจัดการข้อมูลอนุกรมของ 8051

พอร์ตอนุกรมของ 8051 มีโครงสร้างการทำงานในแบบที่เรียกว่า ฟูลดูเพล็กซ์ (Full Duplex) ซึ่งหมายถึงความสามารถในการรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกัน โดยทางด้านวงจรของตัวส่ง (Transmitter) ประกอบด้วยรีจิสเตอร์ SBUF ทำหน้าที่เก็บข้อมูลที่จะส่งออกการใช้คำสั่งเขียนหรือโอนย้ายข้อมูลมายังรีจิสเตอร์นี้ จะเป็นการส่งข้อมูลนั้นออกไปยังพอร์ตอนุกรมทางขาสัญญาณ TxD โดยอัตโนมัติส่วนวงจรด้านตัวรับ (Reciever) ประกอบด้วยรีจิสเตอร์ SBUF เช่นเดียวกัน แต่ทำหน้าที่เก็บข้อมูลที่นำมาจากส่วนวงจรเลื่อนบิตหรือชิพรีจิสเตอร์ (Shift Register) ของวงจรจัดการข้อมูลอนุกรมภายใน สัญญาณข้อมูลอนุกรมที่รับเข้าจะผ่านมาทางขาสัญญาณ RxD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แผนภาพแสดงการทำงานของวงจรส่วนการรับและส่งข้อมูลอนุกรมของ 8051

พอร์ตอนุกรมของ 8051 สามารถโปรแกรมให้ทำหน้าที่ในรูปแบบต่างๆ กันสี่แบบ โดยการกำหนดค่าบิต SM0 กับ SM1 ซึ่งอยู่ภายในรีจิสเตอร์ควบคุมและบอกสถานะ SCON

ซึ่งโหมดการทำงานทั้ง 4 แบบของพอร์ตอนุกรม มีดังนี้

โหมดการทำงาน	คำอธิบาย
โหมด 0	เป็นการขยายพอร์ตอินพุตเอาต์พุต โดยทำงานร่วมกับไอซีชิพรีจิสเตอร์ภายนอกประเภททีทีแอลหรือซีมอส
โหมด 1	ใช้สำหรับการเชื่อมต่ออนุกรมแบบ UART (Universal Asynchronous receiver/ Transmitter) โดยการใช้กลุ่มข้อมูลแบบ 10 บิตและสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้
โหมด2	ใช้สำหรับการเชื่อมต่ออนุกรมแบบ UART โดยการใช้กลุ่มข้อมูลแบบ 11 บิต และกำหนดอัตราความเร็วในการส่งข้อมูลคงที่
โหมด 3	ใช้สำหรับการเชื่อมต่ออนุกรมแบบ UART โดยการใช้กลุ่มข้อมูลแบบ 11 บิต และสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้

ตารางที่ 2.1 แสดงโหมดการทำงานของพอร์ตอนุกรม

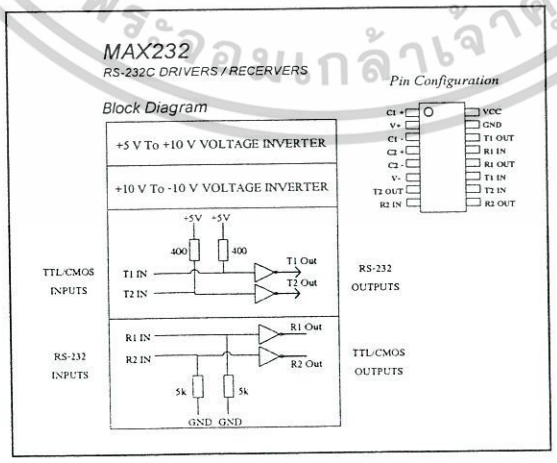
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.9 กระบวนการรับและส่งข้อมูลอนุกรมของ 8051

การส่งข้อมูลออกทางพอร์ตอนุกรมของ 8051 จะเริ่มต้นขึ้น ภายหลังเมื่อมีการเขียนข้อมูลลงในรีจิสเตอร์ SBUF ข้อมูลนี้จะถูกจัดการด้วยวิธีการทางด้านฮาร์ดแวร์ในการเลื่อนบิตและส่งสัญญาณออกไปภายนอกโดยอัตโนมัติ เมื่อข้อมูลเหล่านี้ได้ส่งออกครบถ้วนแล้ว จึงจะทำการกำหนดค่าของแฟล็ก TI ให้เป็น 1 เพื่อแจ้งให้ทราบว่าขณะนี้รีจิสเตอร์ SBUF ว่าง และพร้อมที่จะส่งข้อมูลไบต์ต่อไปแล้ว ในกรณีที่ผู้ใช้เขียนข้อมูลใหม่ลงในรีจิสเตอร์ SBUF โดยไม่รอให้แฟล็ก TI มีค่าเป็น 1 ก่อนจะมีผลทำให้ข้อมูลที่ส่งออกไปผิดพลาดได้ สำหรับการรับข้อมูลจากพอร์ตอนุกรมจะต้องเริ่มต้นโดยการกำหนดค่าบิต REN ให้มีค่าเป็น 1 ก่อน หลังจากนั้นเมื่อมีบิตของข้อมูลถูกส่งเข้ามาจากภายนอกระบบฮาร์ดแวร์ของ 8051 จึงจะทำการเลื่อนบิตเหล่านี้เข้ามาโดยอัตโนมัติ และ เมื่อบิตสุดท้ายถูกเลื่อนเข้ามาเรียบร้อยแล้ว ข้อมูลนั้นจะถูกย้ายมาเก็บไว้ยังรีจิสเตอร์ SBUF และทำการกำหนดให้แฟล็ก RI ให้มีค่าเป็น 1 ซึ่งมีผลทำให้เกิดการอินเตอร์รัปต์โปรแกรมขึ้น

2.1.10 การเชื่อมต่อแบบมาตรฐาน RS-232C

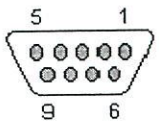
ในการเชื่อมต่อแบบอนุกรมเข้ากับอุปกรณ์คอมพิวเตอร์ต่างๆ เช่น คอมพิวเตอร์ เทเลกซ์ หรือ โทรพิมพ์ เป็นต้น มักจะกำหนดใช้การเชื่อมต่อตามมาตรฐาน RS-232C ทั้งนี้เพื่อให้มีการใช้งานเส้นสัญญาณ หรือรูปแบบของตัวเชื่อมต่อที่สอดคล้องกัน จะได้ลดปัญหาการเข้ากันไม่ได้ระหว่างสัญญาณของอุปกรณ์ที่มาเชื่อมต่อกันทั้งสองด้านให้น้อยลง เนื่องจากระดับโวลเตจที่ใช้และการแทนความหมายของระดับลอจิกตามมาตรฐานนี้ แตกต่างไปจากที่ใช้งานกันในระบบดิจิตอลทั่วไปโดยระดับสัญญาณของ RS-232C เป็นแบบไบโพลาร์ ระดับโวลเตจทางด้านลบช่วง -3V ถึง -20V แทนค่าลอจิก 1 และโวลเตจทางด้านบวกช่วง +3V ถึง +20V แทนค่าลอจิก 0 ดังนั้นจะเห็นได้ว่ามีความจำเป็นต้องเพิ่มเติมอุปกรณ์หรือวงจรพิเศษเข้าไป เพื่อเปลี่ยนระดับโวลเตจจากระบบ 0V ถึง +5V จากขาสัญญาณของ 8051 เป็นระดับโวลเตจที่สูงกว่าค่า +3.0V หรือต่ำกว่า -3.0V



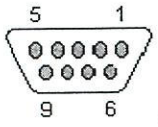
รูปที่ 2.4 ไอซีเบอร์MAX232ซึ่งเป็นวงจรเชื่อมต่อแบบRS-232Cโดยการใช้ไฟเลี้ยง +5 เพียงชุดเดียว วงจรการเชื่อมต่อระหว่าง 8051 กับ PC ทาง Serial Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อสัญญาณ Serial Port



(To Computer 1).



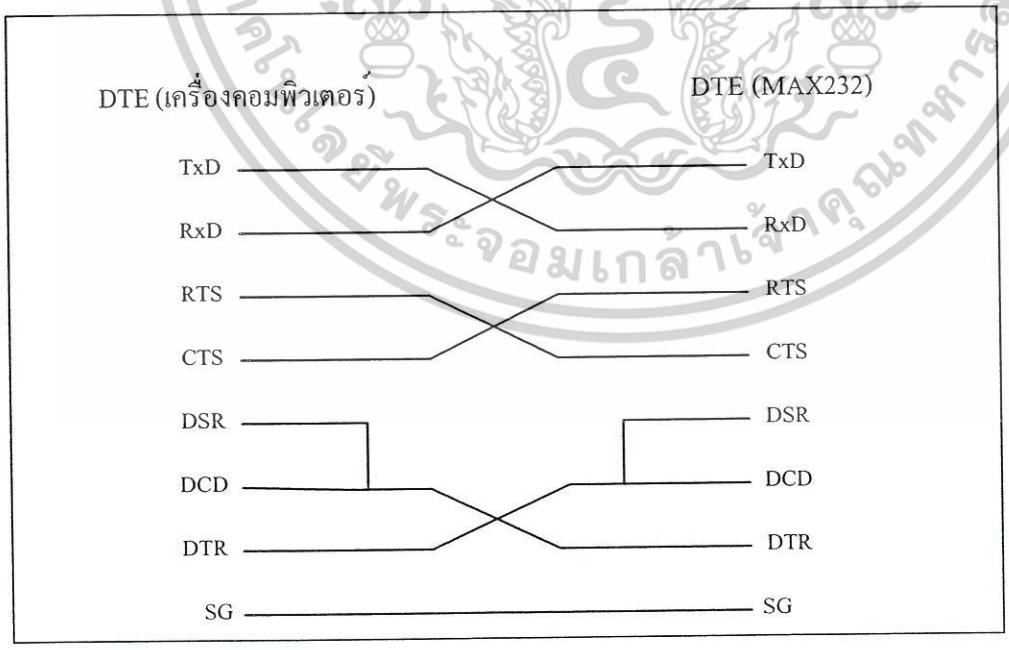
(To Computer 2).

9 PIN D-SUB FEMALE to Computer 1.

9 PIN D-SUB FEMALE to Computer 2.

	D-Sub 1	D-Sub 2	
Receive Data	2	3	Transmit Data
Transmit Data	3	2	Receive Data
Data Terminal Ready	4	6+1	Data Set Ready + Carrier Detect
System Ground	5	5	System Ground
Data Set Ready + Carrier Detect	6+1	4	Data Terminal Ready
Request to Send	7	8	Clear to Send
Clear to Send	8	7	Request to Send

การต่อสายอินเตอร์เฟส RS232 ระหว่างเครื่องคอมพิวเตอร์ กับ MAX232



รูปที่ 2.5 การเชื่อมต่อขาสัญญาณ

2.2 มอเตอร์กระแสตรง

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

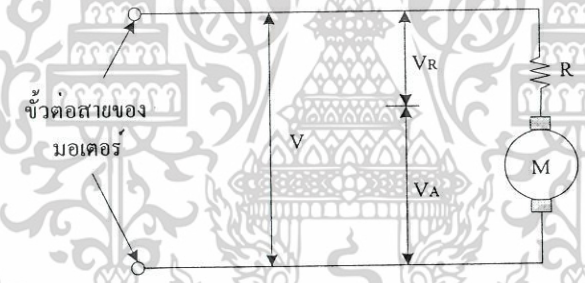
2.2.1 หลักการทำงานของมอเตอร์กระแสตรง

เมื่อมีการผ่านกระแสไฟฟ้าเข้าไปยังขดลวดในสนามแม่เหล็กจะทำให้เกิดแรงแม่เหล็ก ซึ่งมีสัดส่วนของแรงขึ้นกับกระแสแรงของสนามแม่เหล็ก โดยแรงจะเกิดขึ้นเป็นมุมฉากกับกระแสและสนามแม่เหล็ก ขณะที่ทิศทางของแรงกลับตรงกันข้ามกัน ถ้าหากกระแสของสนามแม่เหล็กไหลย้อนกลับจะทำให้เกิดการเปลี่ยนแปลงของกระแส และ สนามแม่เหล็กเป็นผลทำให้ทิศทางของแรงเปลี่ยนไป ด้วยคุณสมบัตินี้ทำให้มอเตอร์กระแสตรงกลับทิศทางการหมุนได้

สนามแม่เหล็กของมอเตอร์ส่วนหนึ่งเกิดขึ้นจากแม่เหล็กถาวร ซึ่งจะถูกยึดติดกับแผ่นเหล็กหรือ เหล็กกล้า โดยปกติส่วนนี้จะเป็นส่วนที่ยึดอยู่กับที่ และ ขดลวดเหนี่ยวนำจะพันอยู่กับส่วนที่เป็นแกนหมุนของมอเตอร์

2.2.2 คุณสมบัติของมอเตอร์กระแสตรง

ในการอธิบายคุณสมบัติของมอเตอร์กระแสตรงให้ละเอียดนั้นต้องพิจารณาแรงดันที่ป้อนและความต้านทานของโรเตอร์ด้วย วงจรภายในของมอเตอร์เขียนได้ดังรูปที่ 3



รูปที่ 2.6 วงจรภายในของมอเตอร์กระแสตรง

โดยสมมติให้ทุ่นโรเตอร์ไม่มีความต้านทานอยู่เลย อนุกรมกับความต้านทานซึ่งในที่นี้ก็คือความต้านทานของขดลวดนั่นเอง แรงดันที่ขั้วต่อสายของมอเตอร์ก็คือผลบวกระหว่างแรงดันที่ทุ่นโรเตอร์ (V_A) และ แรงดันตกคร่อมความต้านทานขดลวด (V_R)

แรงดัน V_A ถูกเรียกว่า แรงเคลื่อนเหนี่ยวนำป้อนกลับ (BACK EMF) ซึ่งเกิดขึ้นในโรเตอร์ขณะที่หมุนแรงดันที่เกิดขึ้นนี้เป็นไปตากกฎของการเหนี่ยวนำแม่เหล็กไฟฟ้า จากการเคลื่อนที่ของตัวนำในสนามแม่เหล็ก สัมพันธ์กับแรงเคลื่อนเหนี่ยวนำแม่เหล็กและความเร็วในการเคลื่อนที่ของตัวนำ แรงดันที่เกิดขึ้นจะมีขั้วตรงกันข้ามกับแรงดันที่ป้อนให้กับมอเตอร์และแปรผันตรงกับความเร็วในการหมุน ผลบวกของแรงดันที่ทุ่นโรเตอร์ (V_A) และแรงดันตกคร่อมขดลวด (V_R) ต้องเท่ากับแรงดันที่ป้อนให้กับมอเตอร์ (V)

$$V = V_A + V_R \qquad (V)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพิจารณาตั้งแต่มอเตอร์หยุดนิ่ง ความเร็วมีค่าเป็นศูนย์ ดังนั้น $V_A = 0, V_R = V$ กระแสที่ไหลในมอเตอร์หาได้จาก

$$I = V_R / R \qquad (A)$$

เมื่อมอเตอร์เริ่มหมุนจะมีความเร็ว และ V_A เพิ่มขึ้นเป็นเส้นตรงตามความเร็ว V_R ซึ่งมีค่าเท่ากับความแตกต่างระหว่าง V_A และ V จะเริ่มลดลงกระแส I ก็จะเริ่มลดลงเช่นกันขณะที่มอเตอร์ยังมีความเร่งอยู่ความเร็วจะเพิ่มขึ้น แรงบิดจะลดลงจนกว่าจะถึงจุดซึ่งแรงบิดจองมอเตอร์รับภาระโหลดได้สมดุลพอดี ขณะที่มอเตอร์ไม่มีโหลดและหมุนอย่างอิสระจะมีเพียงค่าความฝืดของแบริ่งและแรงต้านอากาศทำให้ V_A เกือบเท่ากับค่า V

2.2.3 โมเดลคณิตศาสตร์ของดีซีมอเตอร์

ดีซีมอเตอร์ที่ใช้ร่วมกับ ดีซีแอมปลิไฟล์ ทั้งในระบบการบังคับตำแหน่งและการบังคับความเร็ว มักจะได้รับการประยุกต์ใช้เป็นส่วนประกอบสร้างกำลังงานในระบบการนำร่องและระบบบังคับต่างๆและเนื่องจากวิทยาการเกี่ยวกับสารแม่เหล็กและการขยายด้วย โซลิดสเตททำให้ดีซีมอเตอร์แบบแม่เหล็กถาวรได้รับความนิยมใช้เป็นส่วนประกอบการขับเคลื่อนในระบบการบังคับแบบปิดลูฟต่างๆมากขึ้น การออกแบบและการชดเชยระบบดังกล่าวได้อย่างเหมาะสมจะต้องใช้โมเดลทางคณิตศาสตร์ของส่วนประกอบทั้งหมดในระบบ

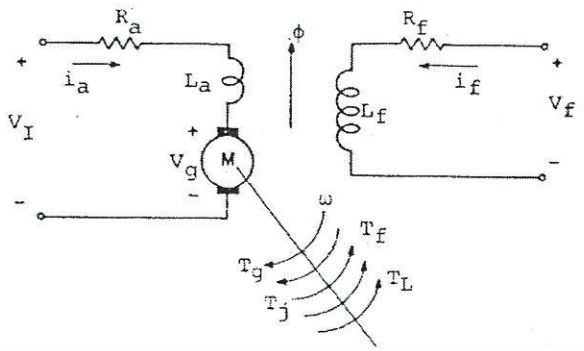
รูปที่ 2.7 แสดงอินพุตและเอาต์พุตของโมเดลทางคณิตศาสตร์ของมอเตอร์

โมเดลอิเล็คโทรแมคคานิคอล

ส่วนสำคัญของดีซีมอเตอร์แบบฟิลด์แยกกระตุ้นมีโมเดลดังแสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.8 แสดงโมเดลของดีซีมอเตอร์แบบฟิลด์แยกกระตุ้น

R_a: ความต้านทานของอาร์เมเจอร์

L_a: อินดัคแต้นซ์ของอาร์เมเจอร์

V_g: โวลท์เต็จกำเนิดในอาร์เมเจอร์(โวลท์เต็จย้อนกลับ)

R_f: ความต้านทานของฟิลด์

L_f: อินดัคแต้นซ์ของฟิลด์

ϕ : ช่องว่างอากาศของเส้นแรงสนามแม่เหล็ก

ω : ความเร็วของเพลาอาร์เมเจอร์

T_g: แรงบิดที่พัฒนาขึ้นในมอเตอร์

T_f: แรงบิดเสียดทานของมอเตอร์

T_J: แรงเฉื่อยของมอเตอร์

T_L: แรงบิดโหลดบนเพลาของมอเตอร์

ขั้นแรกเราจะหาสมการพื้นฐานโมเดลของดีซีมอเตอร์ได้จากลูฟของอาร์เมเจอร์

$$V_i(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + V_g(t) \qquad \text{----- (1)}$$

เทอมโวลท์เต็จ $V_g(t)$ ในสมการ (1) คือโวลท์เต็จย้อนกลับของมอเตอร์ ซึ่งเกิดขึ้นเมื่อเส้นลวดตัวนำของอาร์มาเจอร์หมุนตัดเส้นแรงแม่เหล็กซึ่งเกิดขึ้นในกระแสของฟิลด์ (i_f) ตามกฎของฟาราเดย์ลูฟของเส้นลวดตัวนำหมุนในฟิลด์แม่เหล็กคงที่จะมีการเหนี่ยวนำโวลท์เต็จขึ้นในขดลวดนั้น

$$V_i(t) = \frac{d\lambda(t)}{dt} \qquad \text{----- (2)}$$

เมื่อ $\lambda(t)$ คือเส้นแรงแม่เหล็กที่ลิงเคจ(linkages) ไปยังขดลวดและ t คือเวลาในการหมุนของค็อมนิวเทเตอร์ของมอเตอร์ การควบคุมวงจรของแต่ละส่วนของตัวนำในโรเตอร์จะเกิดโวลท์เต็จขึ้นในส่วนของตัวนำนั้นตามสมการ (2) เมื่อ $\frac{d\lambda(t)}{dt}$ จะเป็นสัดส่วนต่อเส้นแรงแม่เหล็กในช่องว่างอากาศและความเร็วเชิงมุม $\omega(t)$ เราจะได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

103031

$$V_g(t) = K\phi(t)\omega(t) \qquad \text{----- (3)}$$

สมมติให้กระแสของฟิลด์มีค่าคงที่และไม่คิดถึงส่วนการเปลี่ยนแปลงในเส้นแรงฟิลด์ เนื่องจากอาร์เมเจอร์รีแอคชั่นเส้นแรงฟิลด์ก็จะมีค่าคงที่ดังน้นสมการ (3) ก็จะเป็น

$$V_g(t) = K_e\omega(t) \qquad \text{---- (4)}$$

เมื่อเราสมมติให้เส้นแรงของฟิลมีค่าคงที่ แรงบิดของแม่เหล็กไฟฟ้าซึ่งเกิดขึ้นแก่โรเตอร์ของมอเตอร์จะเป็นสัดส่วนกับกระแสของอาร์มาเจอร์

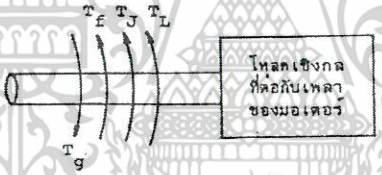
$$T_g(t) = K_t\, i_a(t) \qquad \text{---- (5)}$$

เมื่อ K_t คือ ค่าคงที่ของแรงบิดของมอเตอร์

กำลังงานเชิงกลที่เกิดขึ้นในโรเตอร์คือผลคูณของแรงบิดที่เกิดขึ้นและความเร็วเชิงมุม

$$P_g(t) = T_g(t)\,\omega(t) \qquad \text{---- (6)}$$

กำลังงานเชิงกลที่เกิดขึ้นในโรเตอร์ ทั้งหมดนี้จะจ่ายไปยังโหลดที่ต่ออยู่กับเพลาของมอเตอร์ แต่กำลังงานนี้บางส่วนจะสูญเสียไปในมอเตอร์ การสูญเสียจากแรงเสียดทาน หมายถึงความหน่วงเนื่องจากลมที่มีต่อโรเตอร์ แรงเสียดทานตัวรองรับโรเตอร์ กระแสที่ไหลวนในเหล็กของโรเตอร์และฮีสเทรีซิส โดยแรงบิดต่างๆแสดงดังนี้



รูปที่2.9 แสดงถึงแรงบิดต่างๆที่เกิดขึ้นต่อโหลดของมอเตอร์

$T_g(t)$: แรงบิดของมอเตอร์

$T_f(t)$: แรงบิดที่ต้องชนะการสูญเสียเนื่องจากการเสียดทาน

$T_J(t)$: แรงบิดเพื่อใช้เพิ่มอัตราเร่งแก่ความเฉื่อยของโหลด

$T_L(t)$: แรงบิดโหลด

ในช่วงเวลาใดๆก็ตาม แรงบิดของมอเตอร์จะต้องเท่ากับและมีทิศทางตรงข้ามกับผลรวมของแรงบิด $T_f(t)$ $T_J(t)$ และ $T_L(t)$ ดังนั้น

$$T_g(t) = T_f(t) + T_L(t) + J\frac{d\omega(t)}{dt} \qquad \text{----- (7)}$$

เมื่อ J คือผลรวมของโมเม้นท์แรงเฉื่อยของโรเตอร์และโหลดที่ต่ออยู่ที่เพาของมอเตอร์

ผลรวมของแรงบิดเสียดทานที่ประกอบกันขึ้นที่เพลาของมอเตอร์ ซึ่งเป็นลิเนียร์ฟังก์ชันกับความเร็วเชิงมุมของโรเตอร์เรียกว่า ส่วนประกอบของวิสคอสฟริคชั่นและมักจะอยู่ในเทอมที่แยกออกจากฟริคชั่นอื่นๆ ซึ่งแสดงได้ด้วยสมการต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T_g(t) = T_f(t) + T_L(t) + J\frac{d\omega(t)}{dt} + B\omega(t) \qquad \text{----- (8)}$$

เมื่อ B คือสัมประสิทธิ์ของวิสคอสฟริคชั่นของมอเตอร์และโหลดที่ต่ออยู่กับเพลาของมอเตอร์ $T_f(t)$ คือผลรวมของฟริคชั่นของโหลดและมอเตอร์ทั้งหมด มีแรงต้านของลมและการสูญเสียกำลังในเหล็กของเพลามอเตอร์ยกเว้นวิสคอสฟริคชั่น

สมการ (1) (4) (5) และ (8) เป็นชุดสมการพื้นฐานของดีซีมอเตอร์โมเดลและสมการเหล่านี้เราสามารถจะหาทรานสเฟอร์ฟังก์ชั่นของดีซีมอเตอร์ได้ โดยใส่ลาปลาสทรานสฟอร์มทั้งสองข้างของชุดสมการพื้นฐานได้เป็น

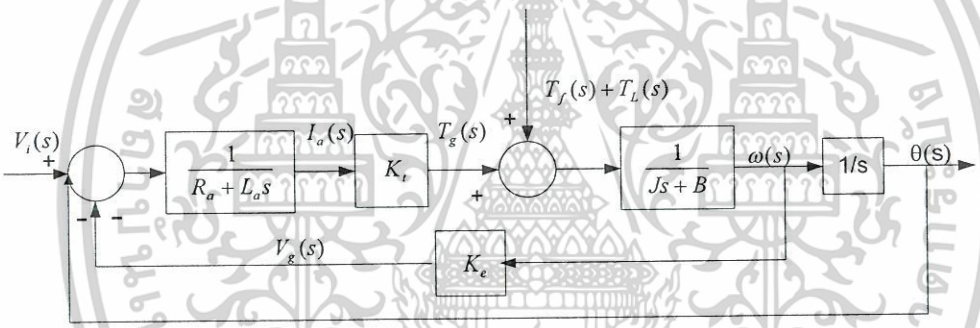
$$V_i(s) - V_g(s) = (R_a + sL_a)\, I_a(s) \qquad \text{----- (9)}$$

$$V_g(s) = K_e\omega(s) \qquad \text{----- (10)}$$

$$T_g(s) = K_t\, I_a(s) \qquad \text{----- (11)}$$

$$T_g(s) - T_f(s) - T_L(s) = (B + sJ)\,\omega(s) \qquad \text{----- (12)}$$

สามารถเขียนเป็นบล็อกไดอะแกรมที่แสดงสมการพื้นฐานเหล่านี้ได้ดังนี้



รูปที่ 2.10 แสดงบล็อกไดอะแกรมของดีซีมอเตอร์โมเดล

ข้อสังเกต

สมมติว่าโวลท์เต็จที่ป้อนให้กับวงจรอาร์มาเจอร์ของมอเตอร์มีค่าคงที่ ดังนั้นมอเตอร์จะหมุนด้วยความเร็วคงที่คือทำงานอยู่ที่สภาวะสงบนิ่งด้วยโหลดที่คงที่ กำลังงานเชิงกลที่เกิดขึ้นโดยโรเตอร์จะหาได้จากสมการ (6) จะได้

$$P_g = T_g\omega = K_t I_a \omega \qquad \text{---- (13)}$$

เมื่อทุกเทอมในสมการสุดท้ายมีค่าคงที่ เนื่องจากมอเตอร์ทำงานอยู่ที่สภาวะสงบนิ่ง กำลังไฟฟ้าที่ถูกดูดกลืนโดยอาร์เมเจอร์ต้องเท่ากับ

$$P = V_g I_a = K_e \omega\, I_a \qquad \text{---- (14)}$$

ดังนั้นเราจะได้ว่ากำลังงานเชิงกลที่เกิดขึ้นต้องเท่ากับกำลังงานไฟฟ้าที่ถูกดูดกลืนใน โรเตอร์ คือสรุปได้ว่า $K_e = K_t$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทรานสเฟอร์ฟังก์ชั่นของดีซีมอเตอร์

บล็อกไดอะแกรมของรูปที่ 5 แสดงถึงระบบที่มีสองอินพุท และมีเอาต์พุทเป็นความเร็วเชิงมุม $\omega(s)$ และการเคลื่อนที่แบบเชิงมุม $\theta(s)$ จากรูปที่ 5 ความเร็วเอาท์พุทของระบบเขียนได้เป็น

$$\omega(s) = G_1(s)V_i(s) + G_2(s)[T_f(s) + T_L(s)] \quad \text{---- (15)}$$

เมื่อ

$$G_1(s) = \left.\frac{\omega(s)}{V_i(s)}\right|_{T_f(s)+T_L(s)=0} \quad \text{---- (16)}$$

$$G_2(s) = \left.\frac{\omega(s)}{T_f(s) + T_L(s)}\right|_{V_i(s)=0} \quad \text{---- (17)}$$

$G_1(s)$ คือทรานสเฟอร์ฟังก์ชั่นระหว่างโวลท์เต็จและความเร็ว

$$G_1(s) = \frac{\omega(s)}{V_i(s)} = \frac{K_t}{(L_a s + R_a)(Js + B) + K_t K_e}$$

$$= \frac{K_m}{\alpha S^2 + \beta s + 1} \quad \text{----- (18)}$$

เมื่อ

$$K_m = \frac{K_t}{R_a B + K_t K_e}$$

$$\alpha = \frac{L_a J}{R_a B + K_t K_e}$$

$$\beta = \frac{R_a J + L_a B}{R_a B + K_t K_e}$$

สมการ (18) เป็นโวลท์เต็จทรานสเฟอร์ฟังก์ชั่นของดีซีมอเตอร์ในเมื่อสมมติว่า T_f และ T_L มีค่าเป็นศูนย์ สมการ (18) สามารถเขียนใหม่ได้เป็น

$$G_1(s) = \frac{K_t}{R_a B(1 + \tau_e s)(1 + \tau_m s) + K_t K_e}$$

เมื่อ $\tau_e = \frac{L_a}{R_a}$ = ไทม์คอนสแต้นท์ทางไฟฟ้า

$\tau_m = \frac{J}{B}$ = ไทม์คอนสแต้นท์ทางเชิงกล

ถ้าอินดัคแต้นท์ของอาร์มาเจอร์มีค่าน้อย ไทม์คอนสแต้นท์ทางไฟฟ้าสามารถตัดทิ้งได้และได้สมการเป็น

$$G_V(s) = \frac{\omega(s)}{V_i(s)} = \frac{K_t}{R_a(Js + B) + K_t K_e}$$

$$= \frac{K_m}{\tau s + 1} \quad \text{----- (19)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ $\tau = \dfrac{R_a J}{R_a B + K_t K_e}$

ในสมการ (19) ค่าคงที่ K_m อาจเรียกได้ว่าเป็นค่าคงที่ของมอเตอร์ ทรานสเฟอร์ ฟังก์ชั่น แรงบิดโหลด $G_2(s)$ หาได้เป็น

$$G_2(s) = \frac{\omega(s)}{T_f(s) + T_L(s)} = \frac{-\dfrac{1}{Js + B}}{1 + \dfrac{K_t K_e}{(Js + B)(L_a s + R_a)}}$$

$$= \frac{-\dfrac{R_a}{K_t} K_m \left[\dfrac{L_a}{R_a} s + 1\right]}{\alpha s^2 + \beta s + 1} \qquad \text{----- (20)}$$

ซึ่งถ้าอินดัคแต้นท์ของอาร์มาเจอร์ไม่นำมาคิด จะทำให้ได้สมการ ดังนี้

$$G_L(s) = \frac{\omega(s)}{T_f(s) + T_L(s)} = \frac{-\dfrac{R_a}{K_t} K_m}{\tau s + 1}$$

ซึ่งจากสมการที่ (15) เมื่อให้ค่าของ ว่า T_f และ T_L มีค่าเป็นศูนย์จะทำให้ค่าทรานสเฟอร์ฟังก์ชั่น มีค่าดังนี้

$$\boxed{\omega(s) = G_v(s) V_i(s) = \frac{K_m}{\tau s + 1} V_i(s)}$$

และจากรูปที่ 4.5 ตำแหน่งเอาท์พุทของระบบเป็นดังนี้

โมเดลคณิตศาสตร์ในการควบคุมตำแหน่งของมอเตอร์ จาก transfer function ในหัวข้อที่ 3.3 โมเดลคณิตศาสตร์ของดีซีมอเตอร์ ซึ่งจะได้ model ของการควบคุมความเร็วของมอเตอร์ในรูปของสมการอันดับหนึ่งเป็นดังนี้ คือ

$$\boxed{\omega(s) = G_v(s) V_i(s) = \frac{K_m}{\tau s + 1} V_i(s)}$$

จากสมการข้างต้นดังกล่าวนั้นจะเห็นถึงความสัมพันธ์ระหว่างความเร็วเชิงมุม (output) และ ค่าแรงดันที่ป้อน (input)

และในการควบคุมตำแหน่งจะมีการผ่านตัว Integrator (1/s) ทำให้ได้เอาท์พุตคือ $\Theta(s)$ ซึ่งจากรูปที่ 5. และ transfer function ที่ได้จากหัวข้อที่ 4.2 ข้างต้นนั้นเมื่อทำการผ่าน Integrator (1/s) เข้าไปจะทำให้ได้ค่าของ output เป็นมุมในการเคลื่อนที่ของมอเตอร์ ซึ่งเขียนในรูปของสมการได้ดังนี้

$$\boxed{\theta(s) = \frac{\omega(s)}{s} = \left[\frac{1}{s}\right]\left[\frac{K_m V_i(s)}{\tau s + 1}\right]}$$

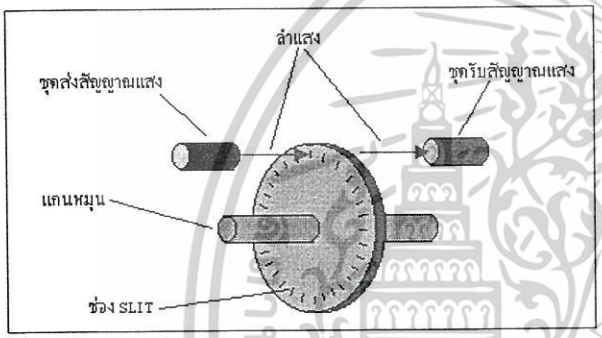
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 การควบคุมตำแหน่งของมอเตอร์

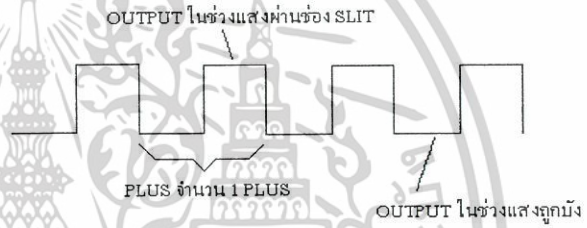
ในโครงงานนี้ได้นำเอา Opto Sensor มาประยุกต์เป็นเอ็นโค้ดเดอร์ซึ่งได้ออกแบบใบเอ็นโค้ดเดอร์เองโดยเมื่อมอเตอร์หมุนไปหนึ่งรอบก็จะมีสัญญาณพัลซ์ออกมา4ลูก แต่ข้อเสียคือจะไม่สามารถรู้ทิศทางการหมุนได้ ดังนั้นการควบคุมทิศทางจะขึ้นอยู่กับโปรแกรมที่ควบคุม โดยหลักการทำงานของ Encoder มีดังนี้

เอ็นโค้ดเดอร์

Rotary Encoder มีลักษณะเป็นแผ่นกลมมีแกนอยู่ตรงกลาง และ ที่แผ่นกลม จะมีช่องเล็ก ที่แสงสามารถ ส่องผ่านได้ เป็นจำนวนมากเราเรียกช่องนี้ว่า ช่อง slit ซึ่งที่ด้านหนึ่งของ แผ่นกลม นี้ จะมีตัวส่งแสง infrared ไปยังตัวรับสัญญาณแสง infrared ซึ่งจะอยู่ในด้านตรงกันข้าม ดังรูปที่ 11



รูปที่ 2.11 แสดง Rotary Encoder



รูปที่ 2.12 แสดงการสร้าง Pulse ของ Rotary Encoder

เมื่อหมุนแกนหมุนทำให้แผ่นกลมหมุนไปตัดลำแสง Infrared ดังนั้นชุดรับแสง infrared จึงมีแสงมากระทบเป็นช่วงๆ (เป็น pulse) ตามจังหวะที่แสงผ่านช่อง slit จึงทำให้ สัญญาณ out put ของชุดรับแสง infrared มีลักษณะเป็น pulse ดังรูปที่ 4.7

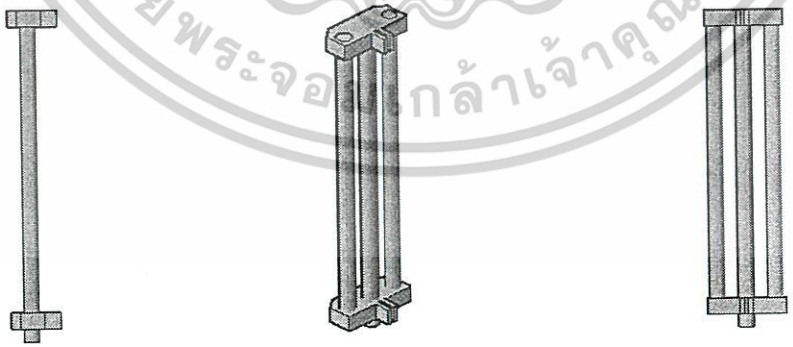
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

3.1 แบบโครงสร้างและชิ้นส่วนประกอบ

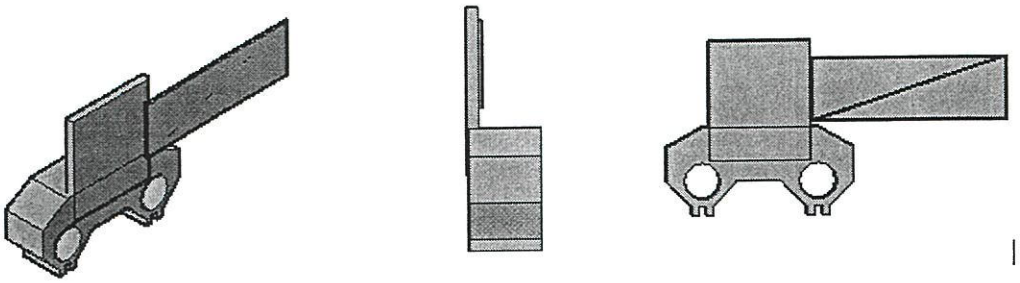
ทำการออกแบบส่วนประกอบต่างๆของแบบจำลองการขนส่งในไลน์การผลิต ซึ่งประกอบด้วย ชั้นวางกล่องชิ้นส่วนประกอบ รางเลื่อนแกน x y z รถขนส่งชิ้นส่วนประกอบ รวมไปถึงภาพรวมทั้งหมดของแบบจำลอง

รูปที่ 3.1 แบบโครงสร้างชั้นวางชิ้นส่วนประกอบ

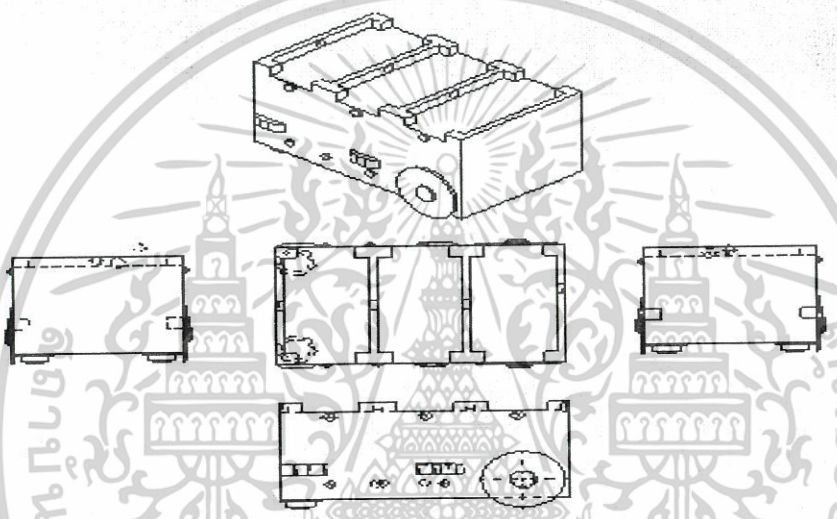


รูปที่ 3.2 แบบโครงสร้างแนวแกน X และ Y

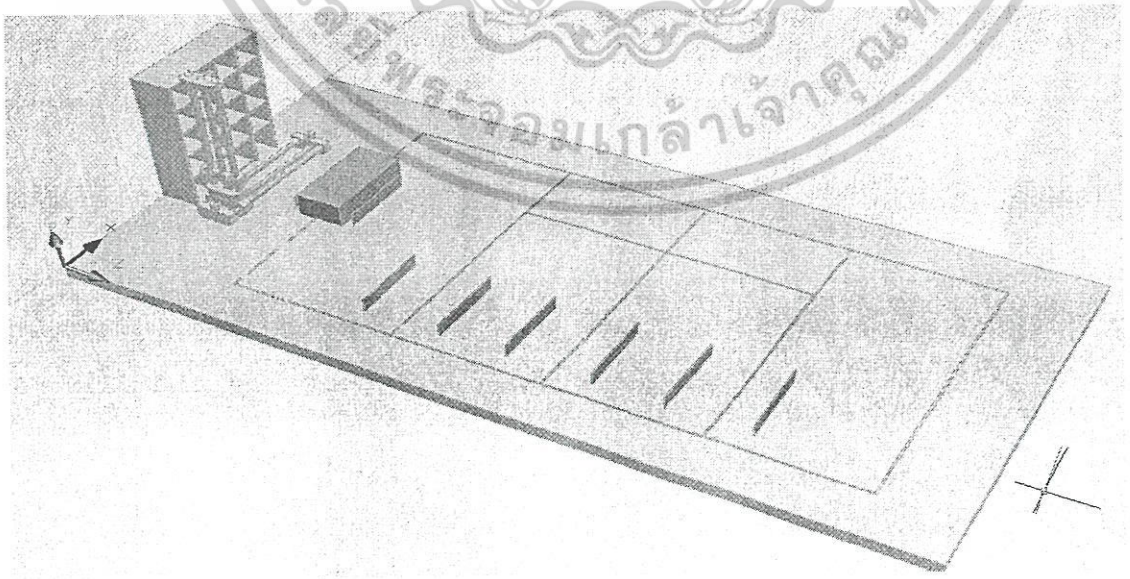
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แบบโครงสร้างฐานตัวจับ



รูปที่ 3.4 แบบโครงสร้างตัวรถขนส่งชิ้นส่วนประกอบ



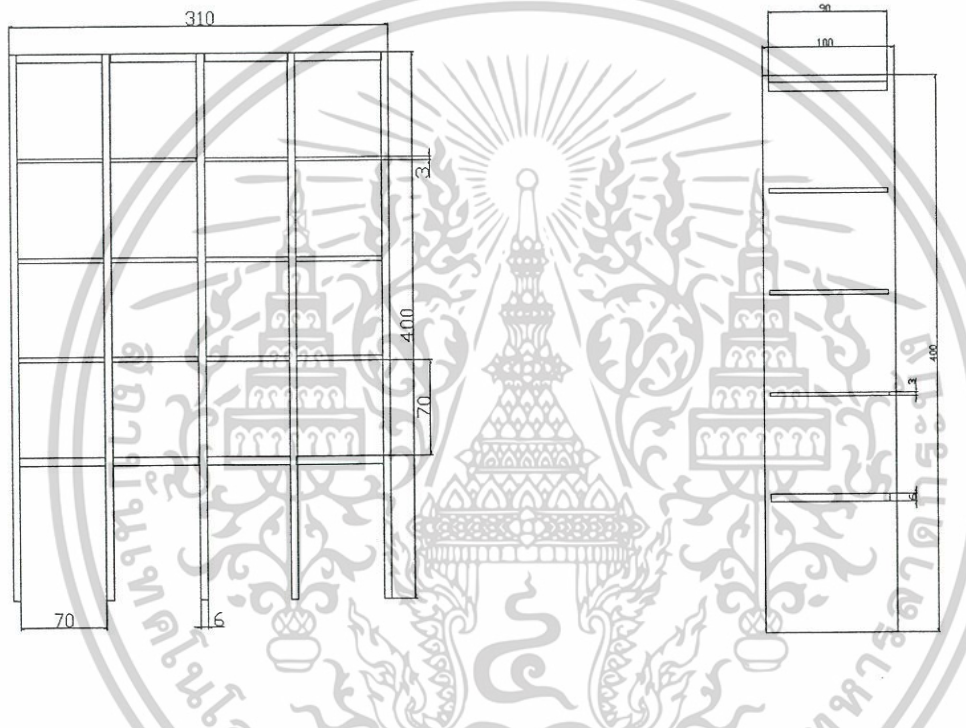
รูปที่ 3.5 แบบจำลองโครงสร้างทั้งหมดของแบบจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 อุปกรณ์และส่วนประกอบในการทำแบบจำลองการขนส่งในไลน์การผลิต

3.2.1 ชั้นวางกล่องชิ้นส่วนประกอบ

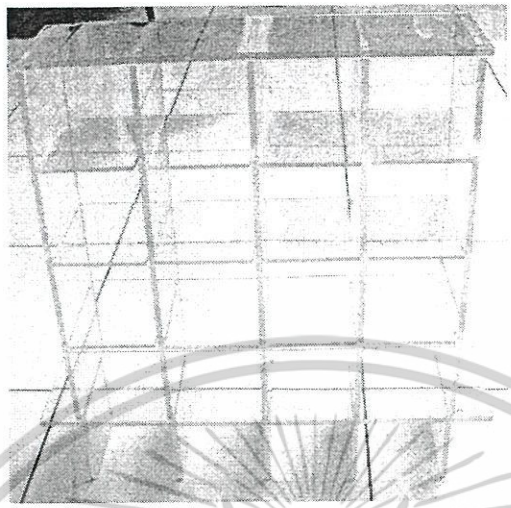
โครงสร้างของสถานีรับส่งชิ้นส่วนประกอบ เป็นช่องสำหรับใช้เก็บกล่องชิ้นส่วนประกอบต่างๆ โดยมีขนาด 31×40×10 cm ซึ่งทำมาจากแผ่นอะคิลิกใสที่มีความหนา 6 มิลลิเมตร และ 3 มิลลิเมตร แบ่งออกเป็น 4 ชั้น แต่ละชั้นมี 4 ช่อง รวมทั้งหมด 16 ช่อง ใช้เก็บกล่องชิ้นส่วนประกอบขนาด 6×6×10 มิลลิเมต



(ก) (ข)

รูปที่ 3.6 (ก) ภาพด้านหน้าแสดงขนาดชั้นรับส่งชิ้นส่วนประกอบ
(ข)ภาพด้านข้างแสดงขนาดชั้นรับส่งชิ้นส่วนประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แสดงโครงสร้างของชั้นรับส่งชิ้นส่วนประกอบ

3.2.2 เครนอัตโนมัติ

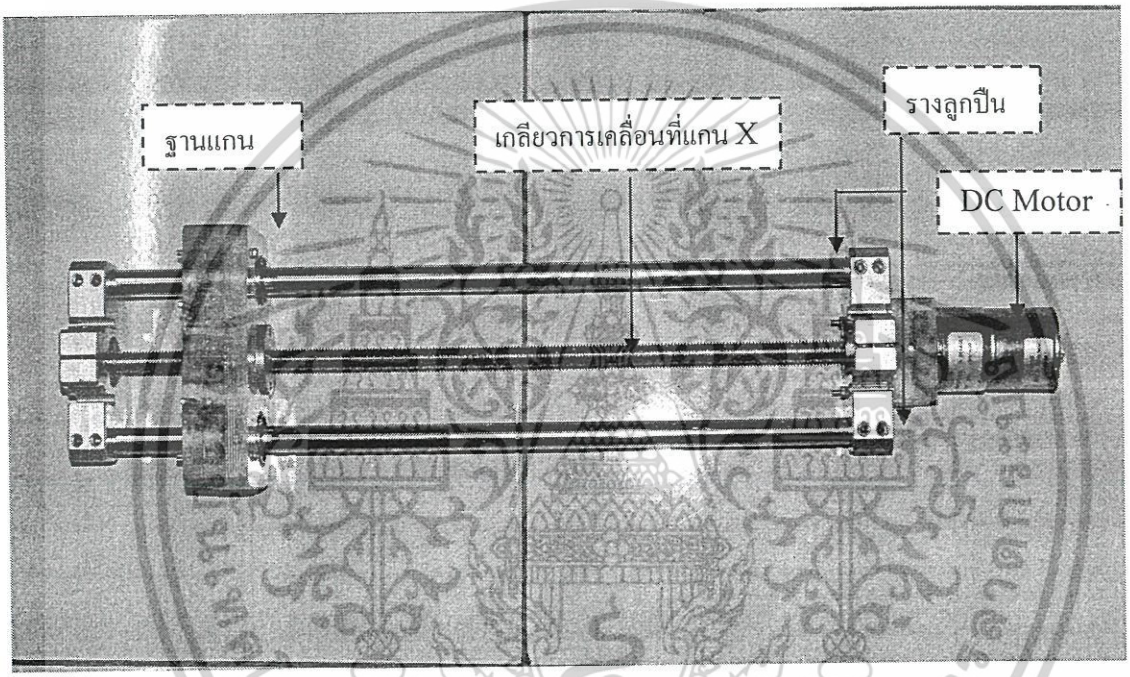
การทำงานของเครนอัตโนมัติ ส่วนใหญ่จะมีการใช้งานคล้ายคลึงกับคลังสินค้า กล่าวคือ ในการทำงานแบบปกติในโรงงาน เมื่อต้องมีการเบิกของจาก store จะต้องมีคนอย่างน้อยหนึ่งคนที่ทำหน้าที่ในการเบิกจ่ายวัสดุอุปกรณ์ในไลน์การผลิตเพื่อนำไปส่งยังไลน์ต่างๆ แต่ store ที่ใช้เครนอัตโนมัตินี้ จะทำหน้าที่ในการขนของจาก store ออกมาและส่งให้กับไลน์การผลิตต่อไป โดยจะรับคำสั่งจากผู้ควบคุม store ซึ่งจะป้อนคำสั่งในการเบิกจ่ายวัสดุอุปกรณ์ที่ store ที่จะเก็บวัสดุอุปกรณ์ในลักษณะของคลังสินค้า และโปรแกรมจะสั่งให้เครนอัตโนมัติไปยกของนำมาส่งยังจุดที่รับส่งวัสดุอุปกรณ์ ซึ่งจะหยิบวัสดุอุปกรณ์ออกมาจากชั้นวางใดๆก็ได้ โดยไม่เกิดความสับสน หรือเสียเวลาในการค้นหา เพราะโปรแกรมการทำงานจะแจ้งให้ผู้คุมทราบว่าอะไรอยู่ที่ใด และการเคลื่อนที่ของเครนจะถูกควบคุมให้มีการทำงานได้ตุ้มค่าเวลาที่สุด ซึ่งจะเป็นการสะดวกและประหยัดเวลาในการทำงานกับพื้นที่ในการจัดเก็บวัสดุอุปกรณ์ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของเครนอัตโนมัติ

ชุดเคลื่อนที่ในแนวแกน X

เป็นโครงสร้างของการเคลื่อนที่ไปตามรางในแนวราบ เพื่อนำตัวยกไปยังช่องเก็บวัสดุอุปกรณ์ที่ต้องการ โดยจะใช้ DC Motor เป็นตัวขับเคลื่อนเกลียว เพื่อให้ตัวฐานของแกน Y เคลื่อนที่ไปยังตำแหน่งที่ต้องการ

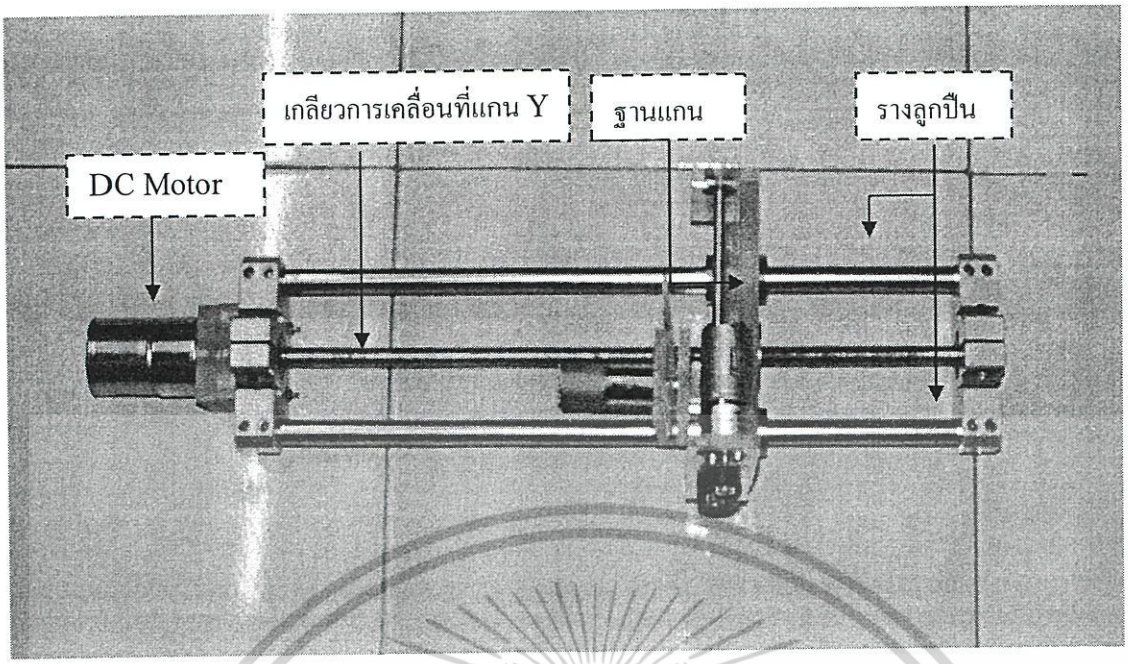


รูปที่ 3.8 ชุดเคลื่อนที่ในแนวแกน X

ชุดเคลื่อนที่ในแนวแกน Y

เป็นโครงสร้างของการเคลื่อนที่ที่ยกตัวขึ้นลงในแนวดิ่ง เพื่อนำตัวยกไปยังช่องเก็บวัสดุอุปกรณ์ที่ต้องการ โดยจะใช้ DC Motor เป็นตัวขับเคลื่อนเกลียว เพื่อให้ตัวฐานของแกน Z เคลื่อนที่ไปยังตำแหน่งที่ต้องการ

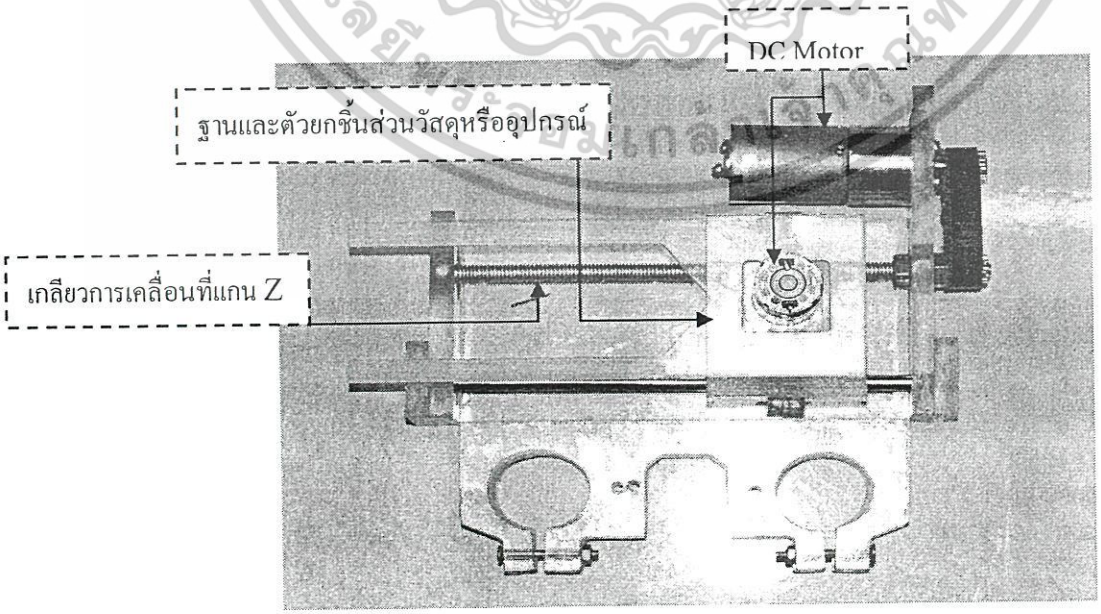
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ชุดการเคลื่อนที่แนวแกน Y

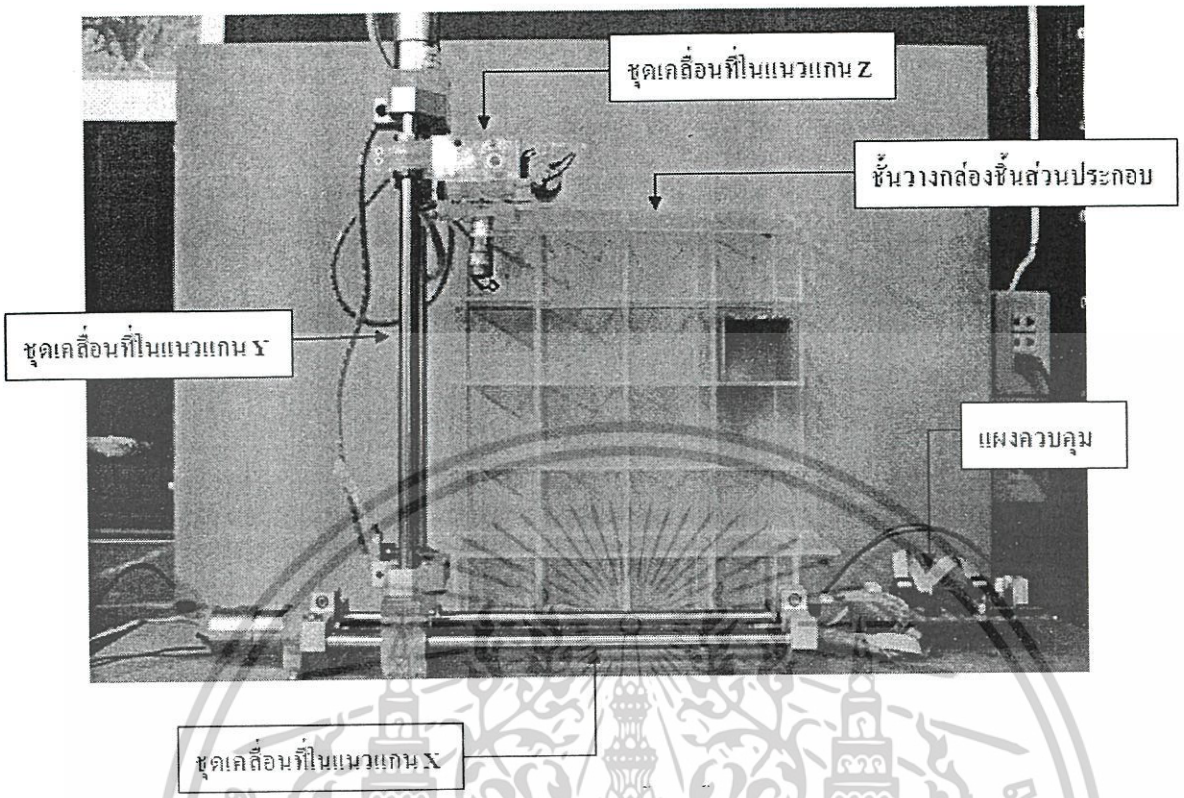
ชุดการเคลื่อนที่แนวแกน Z

เป็นโครงสร้างของการเคลื่อนที่ในแนวราบ เพื่อการนำชิ้นส่วนวัสดุหรืออุปกรณ์ในไลน์การผลิตออกจาก ชั้นวางของใน store ซึ่งสามารถเคลื่อนที่เข้าออกและออกได้ในแนวแกน แล้วยังสามารถหมุน 180 องศา เพื่อนำมาวางบนรถรับส่งในไลน์การผลิต โดยจะใช้ DC Motor เป็นตัวขับเคลื่อนเกลียว เพื่อให้ฐานของตัวยกเคลื่อนที่เข้าและออกในแนวแกน และยังใช้ในการขับเคลื่อนการหมุน 180 องศา ของตัวยกอีกด้วย



รูป 3.10 ชุดการเคลื่อนที่แนวแกน Z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.11 ภาพแสดงชั้นวางกล่องชิ้นส่วนประกอบ และเครนอัตโนมัติ

3.3 แผงวงจรควบคุม

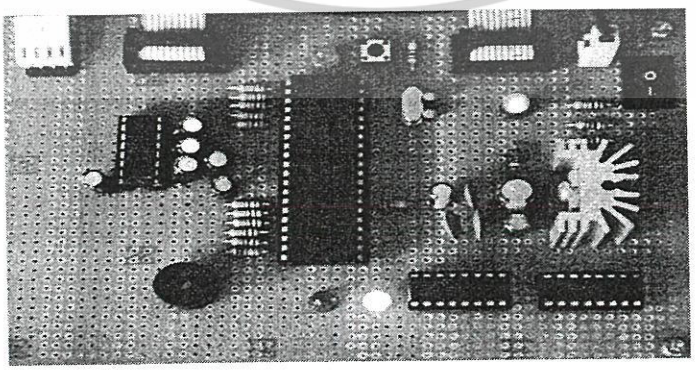
แผงวงจรควบคุมการทำงานประกอบไปด้วย

วงจรไมโครคอนโทรลเลอร์ MCS-51

วงจรปรับแรงดัน (LM 2940)

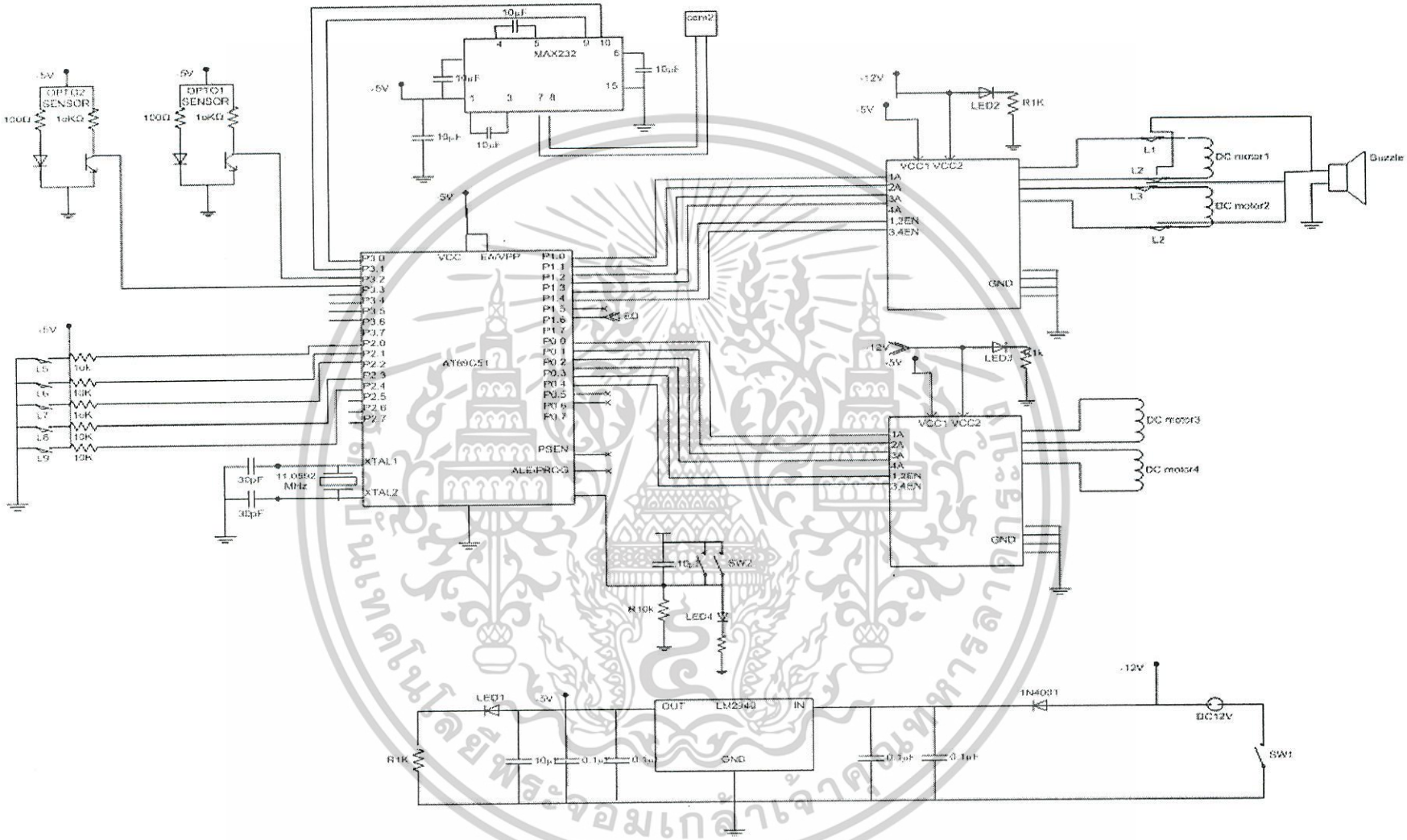
วงจรไดร์เวอร์ขับมอเตอร์ (L 293D)

วงจรการเชื่อมต่อแบบอนุกรม (MAX 232)



รูปที่ 3.12 แผงวงจรควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

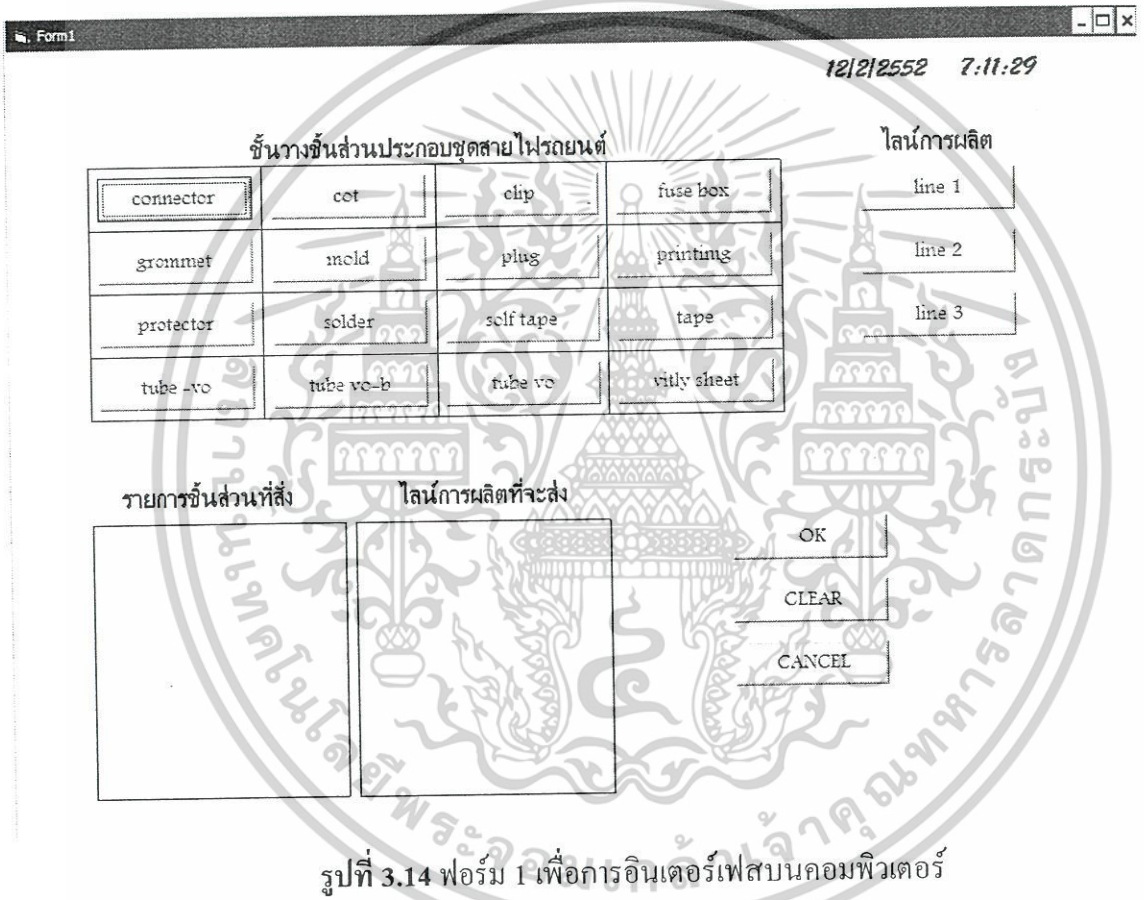


รูปที่ 3.13 วงจรควบคุมเครนอัตโนมัติ

3.4 การใช้ Visual Basic ในการอินเตอร์เฟส

ในส่วนของ Visual Basic ได้ทำการเขียนโปรแกรมขึ้นมาทั้งหมด 3 ฟอร์ม

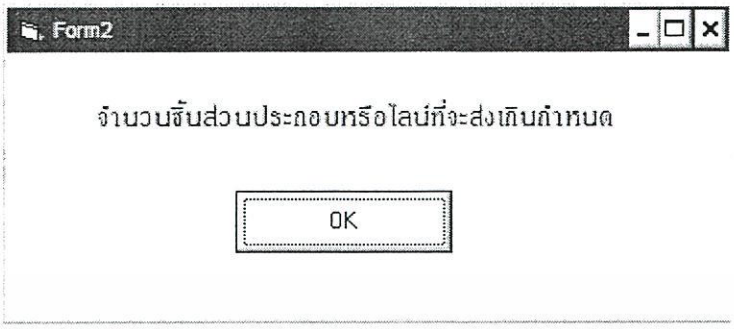
ฟอร์ม 1 หรือฟอร์มหลักได้ทำการออกแบบไวดังนี้ ซึ่งประกอบด้วยส่วนต่างๆ คือ ปุ่มเลือกชิ้นส่วนประกอบที่จะส่ง ปุ่มเลือกไลน์การผลิตที่จะส่ง กล่องข้อความ 2 กล่องเพื่อแสดงรายการชิ้นส่วนประกอบที่เลือก และไลน์การผลิตที่จะส่ง ปุ่ม OK กดเพื่อยืนยันข้อมูล ปุ่ม CLEAR เพื่อทำการล้างข้อมูลรายการและไลน์ที่ได้เลือกไป(ในกรณีกดผิด) ปุ่ม CANCEL กดเพื่อออกจากโปรแกรม



รูปที่ 3.14 ฟอร์ม 1 เพื่อการอินเตอร์เฟสบนคอมพิวเตอร์

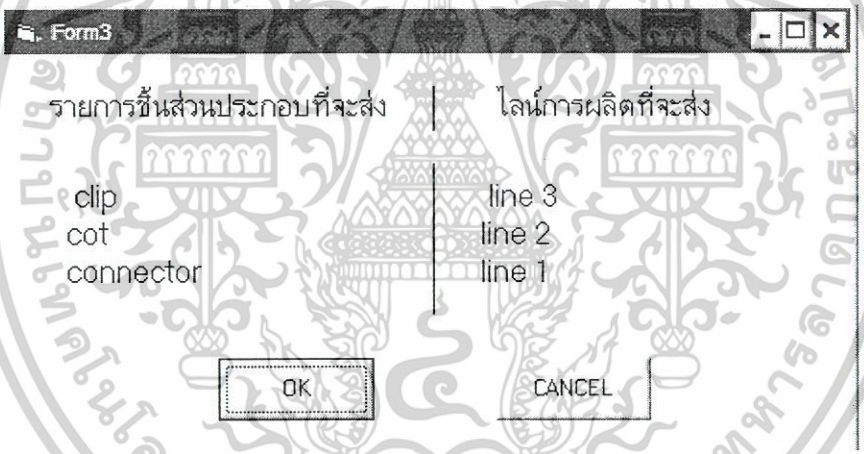
ฟอร์ม 2 เป็นฟอร์มที่เขียนขึ้นเพื่อแสดงว่า รายการชิ้นส่วนประกอบและไลน์การผลิตที่จะส่งนั้นมีความผิดพลาดซึ่งเกิดมาจาก รายการชิ้นส่วนประกอบที่สั่งมีจำนวนเกิน 3 รายการ และไลน์การผลิตเกิน 3 กด OK เพื่อออกจากฟอร์มนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 ฟอร์ม 2 เพื่อการแสดงความผิดพลาดของรายการที่สั่ง

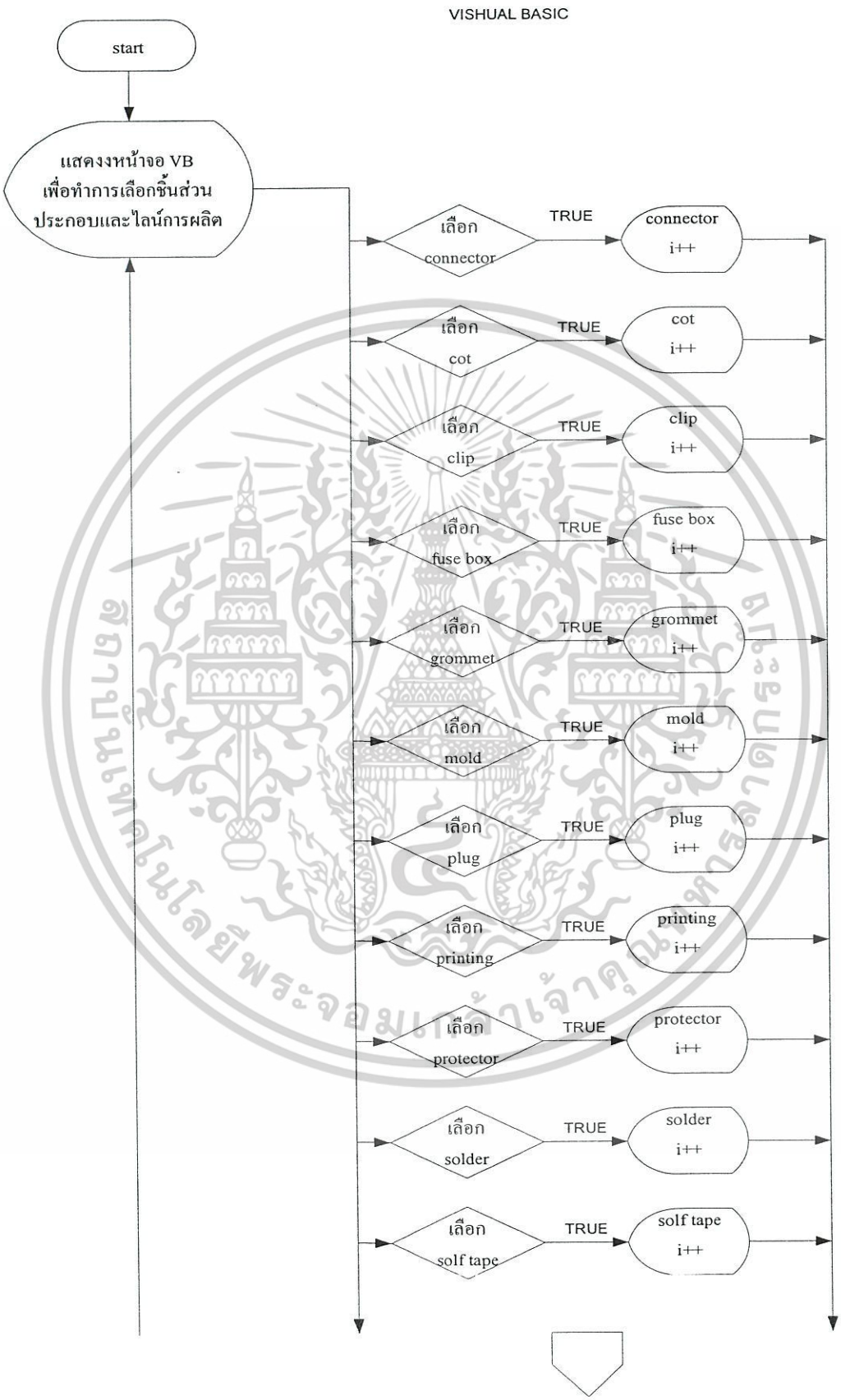
ฟอร์ม 3 เป็นฟอร์มที่เขียนขึ้นเพื่อยืนยันรายการชิ้นส่วนประกอบและไลน์การผลิตที่จะส่ง และจะทำการส่งข้อมูลผ่าน serial port เมื่อกดปุ่มฯษ และกดปุ่ม CANCEL เพื่อออกจากฟอร์มนี้



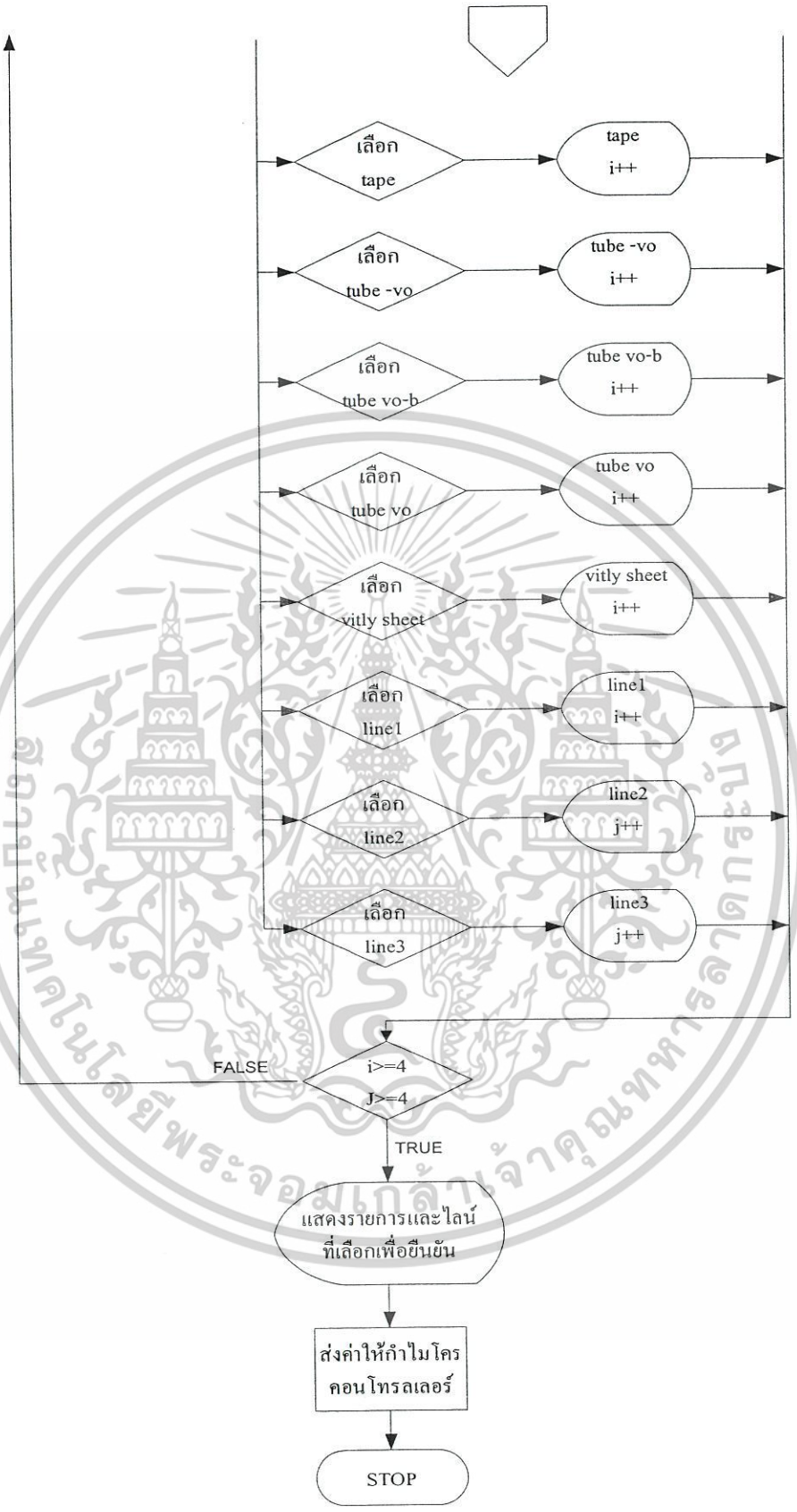
รูปที่ 3.16 ฟอร์ม 2 เพื่อการยืนยันรายการและไลน์การผลิตที่จะส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนผังการทำงานของโปรแกรม Vishual Basiccและ Mcrocontroller



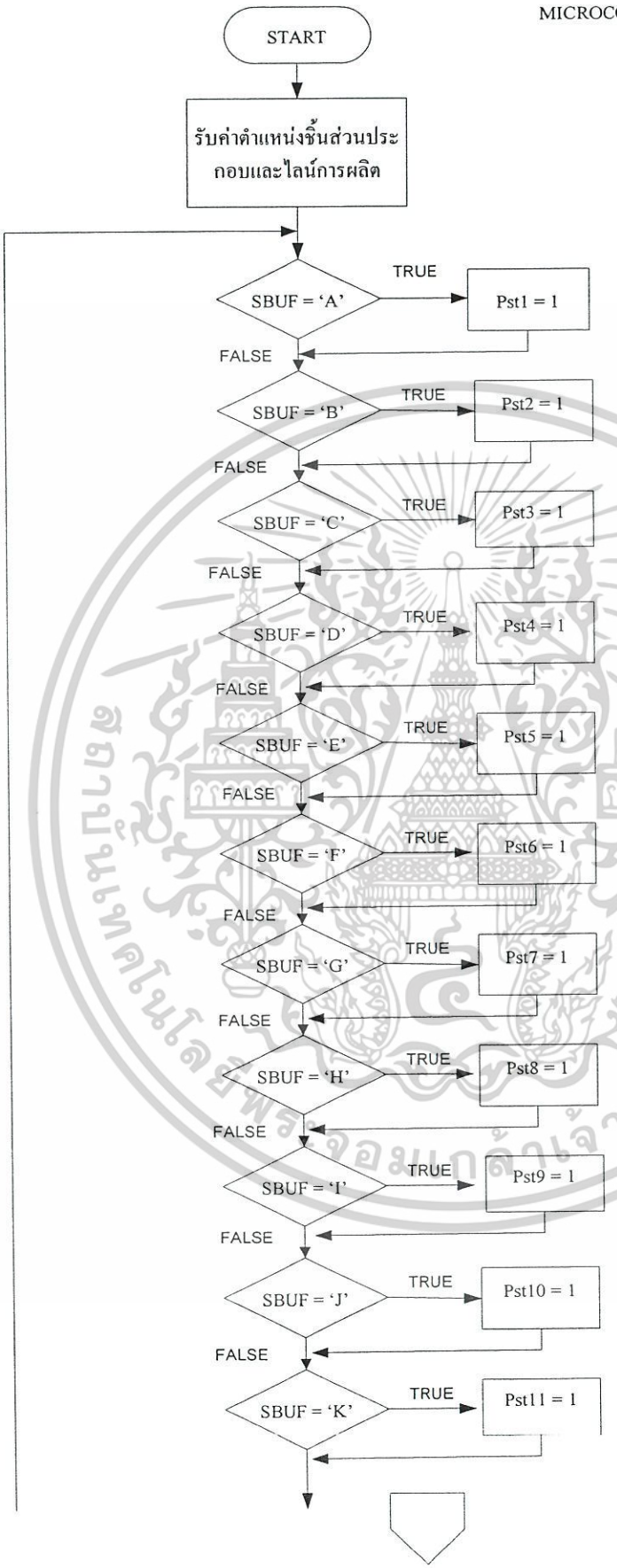
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



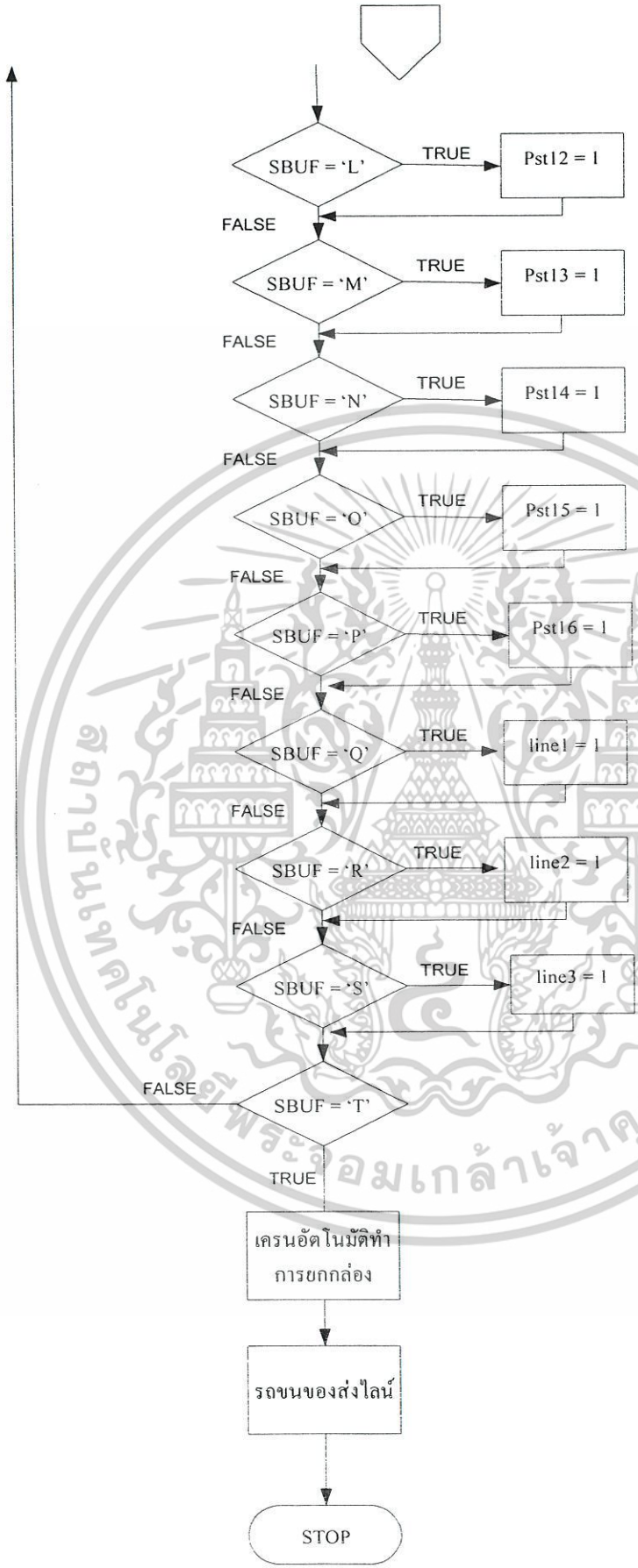
รูปที่ 3.17 โฟลว์ชาร์ตการทำงานของโปรแกรม Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MICROCONTROLLER



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 โฟลว์ชาร์ตการทำงานของเครนอัตโนมัติ

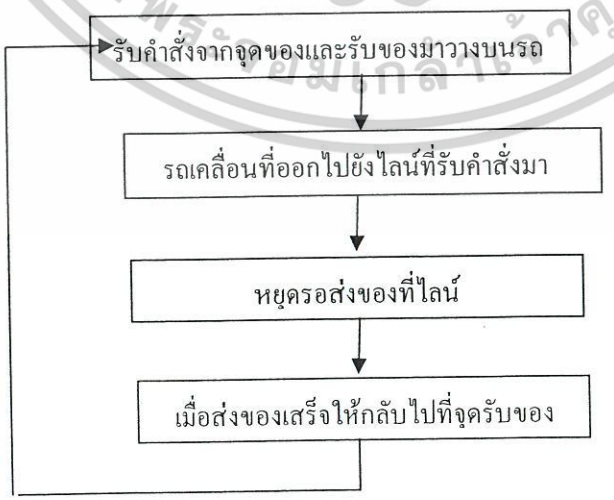
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. 5 รถขนส่งชิ้นส่วนอัตโนมัติ

รถขนส่งชิ้นส่วนอัตโนมัติ จะเป็นส่วนของตัวขนส่งชิ้นส่วนไปยังไลน์การผลิต โดยมีโครงสร้างที่ทำมาจากอะคิลิค ในลักษณะเหมือนกับกล่องสี่เหลี่ยมผืนผ้า โดยมีความกว้างประมาณ 126mm ความยาวประมาณ 206mm และความสูงประมาณ 100mm รถส่งชิ้นส่วนจะขับเคลื่อนด้วย DC motor ขนาด 12v สองตัวซึ่งจะติดตั้งอยู่ที่ท้ายของตัวรถ ส่วนล้อหน้าใช้ล้อบอลซึ่งมีความสามารถเคลื่อนที่ด้าย 360 องศา และด้านข้างจะมีโลเลอร์ที่ช่วยในการเคลื่อนที่เมื่อรถขนส่งชิ้นส่วนเคลื่อนเข้าสู่ไลน์หรือเคลื่อนเข้าตรงจุดรับของ การเคลื่อนที่ของตัวรถจากจุดรับของไปยังไลน์ ใช้การเคลื่อนที่แบบเดินตามเส้นโดยจะมี sensor ที่ใช้ตรวจจับเส้นอยู่ข้างใต้ของตัวรถ และยังมี sensor ด้านข้างอีกทั้งสองข้าง โดยข้างหนึ่งเอาไว้รับคำสั่งจากจุดรับของว่าให้ไปส่งของที่ไลน์การผลิตไลน์ไหน ส่วนอีกข้างจะใช้สำหรับการหยุดของตัวรถในแต่ละไลน์ โดย sensor ที่ใช้เป็น sensor รับส่งแบบอินฟราเรด ซึ่งการควบคุมระบบของตัวรถทั้งหมดจะใช้ Micro-controller เป็นตัวควบคุมการทำงาน โดยมีขอบเขตการทำงานของตัวรถขนส่งดังนี้

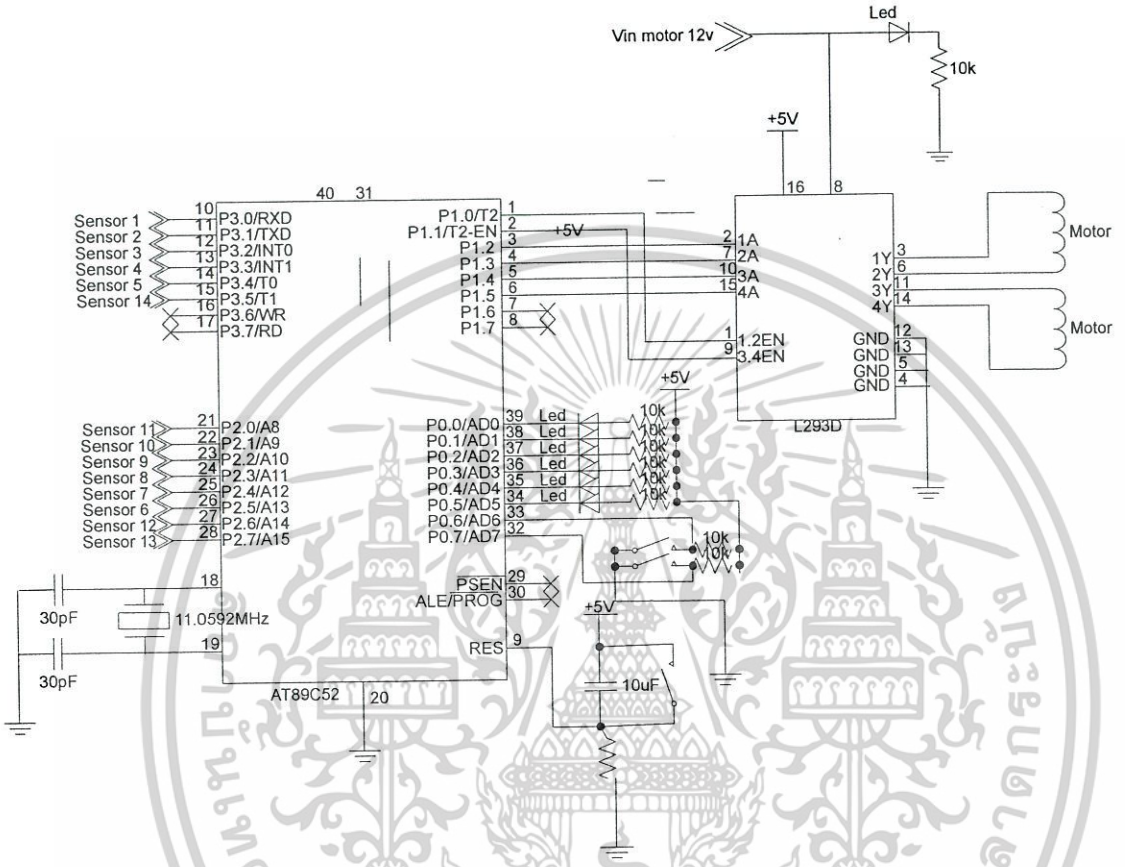
- เป็นขนส่งอัตโนมัติ ที่เคลื่อนที่ด้วย DC motor จำนวนสองตัว
- ตรวจจับเส้นทางโดยใช้ อินฟราเรดเซนเซอร์จำนวน 4 ตัว
- ตรวจจับของที่นำมาวางบนตัวรถขนส่ง โดยใช้อินฟราเรดเซนเซอร์จำนวน 3 ตัว
- รับคำสั่งให้เคลื่อนที่ไปส่งของตามไลน์และรับคำสั่งให้หยุดในแต่ละไลน์ โดยใช้ อินฟราเรดเซนเซอร์จำนวน 6 ตัว
- ตัวรถขนส่งต้องสามารถรับของและรับคำสั่งจากจุดรับของ ซึ่งจะนำไปส่งยังไลน์การผลิตที่ได้รับคำสั่งมาได้

บล็อกแสดงการทำงานของรถขนส่งชิ้นส่วน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจร MCS51 ที่ใช้ควบคุมรถขนส่งชิ้นส่วนกับลายวงจรขับ DC มอเตอร์



รูปที่ 3.19 วงจร Micro-controller, วงจร Drive Motor และ วงจรลดแรงดัน LM2940

Micro-controller เป็นบล็อกหลักสำหรับควบคุมการทำงานทั้งหมด โดยจะเก็บโปรแกรมควบคุมการทำงานไว้ในหน่วยความจำภายใน เนื่องจากตัวรถขนส่งมีขนาดเล็กไม่ใหญ่มากดังนั้น งานควบคุมจึงใช้เพียง Micro-controller MCS51 ได้

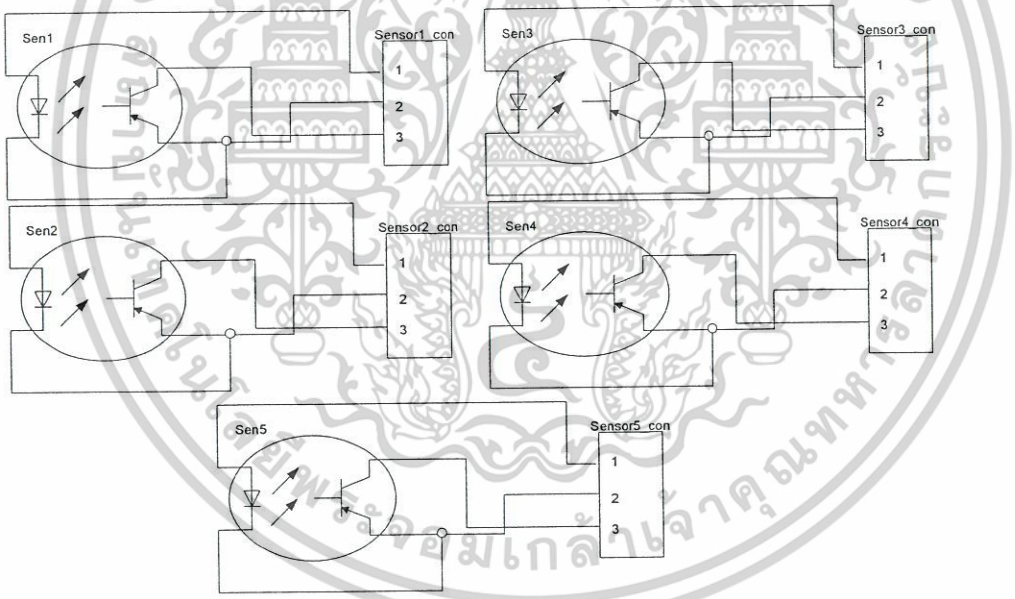
3.6 อินฟราเรดเซนเซอร์

อินฟราเรดเซนเซอร์ ทำหน้าที่ตรวจจับเส้นทางของรถขนส่ง ตรวจจับของที่นำมาวางบนรถขนส่งทำหน้าที่รับสัญญาณคำสั่งในการเลือกไลน์ที่จะส่งและรับสัญญาณหยุดที่ไลน์นั้นๆ ประกอบด้วยวงจรสองส่วนได้แก่ วงจรตรวจจับแสงสะท้อนอินฟราเรด และวงจรเปรียบเทียบแรงดัน ดังแสดงรายละเอียดต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

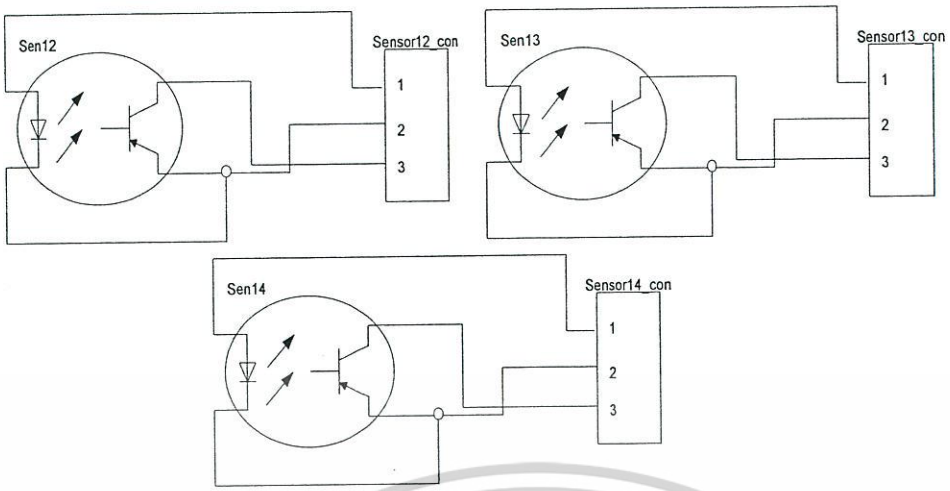
วงจรตรวจจับแสงสะท้อนอินฟราเรด ประกอบด้วย LED อินฟราเรด ทำหน้าที่ส่งแสงอินฟราเรดไปกระทบพื้นและโฟโต้ทรานซิสเตอร์(Photo transistor) จะทำหน้าที่รับแสงอินฟราเรดที่สะท้อนกลับมา ถ้าไม่มีแสงอินฟราเรดมากระทบที่โฟโต้ทรานซิสเตอร์ เอาต์พุตจะมีค่าแรงดันต่ำ

ในกรณีที่ แสงอินฟราเรดโฟโต้ทรานซิสเตอร์ จะยอมให้กระแสไหลผ่านมากหรือน้อยขึ้นอยู่กับ ความเข้มของแสงที่ได้รับ โดยแรงดันเอาต์พุตจะอยู่ที่ 0.5-5v สำหรับในโปรเจ็คนี้จะใช้วงจรตรวจจับทั้งหมด 14 ชุดด้วยกัน แบ่งเป็น วงจรตรวจจับเส้นที่ติดตั้งไว้ใต้ท้องของตัวรถขนส่ง 5 ชุด วงจรตรวจจับเมื่อมีของมาวางบนตัวรถที่ติดตั้งไว้ด้านบนของตัวรถขนส่ง 3 ชุด และวงจรป้อนคำสั่งให้รถวิ่งกับวงจรสั่งให้รถหยุด ส่วนของสองวงจรนี้จะติดตั้งตัวรับกับตัวส่งของ sensor ไว้แยกกันโดยตัวส่งสัญญาณจะติดตั้งไว้ที่จุดรับของและที่รับของตามไลน์การผลิต ในส่วนของตัวรับสัญญาณจะติดตั้งไว้ที่ด้านข้างของตัวรถ ซึ่งจะแยกกันระหว่างตัวส่งสัญญาณคำสั่งให้รถวิ่งกับตัวส่งสัญญาณให้รถหยุดไว้คนละด้านกัน ทั้งหมด 6 ชุด โดยแต่ละชุดจะมีคอนเน็กเตอร์ (connector) เชื่อมต่อไปยังวงจรเปรียบเทียบแรงดัน วงจรที่ได้แสดงดังรูป

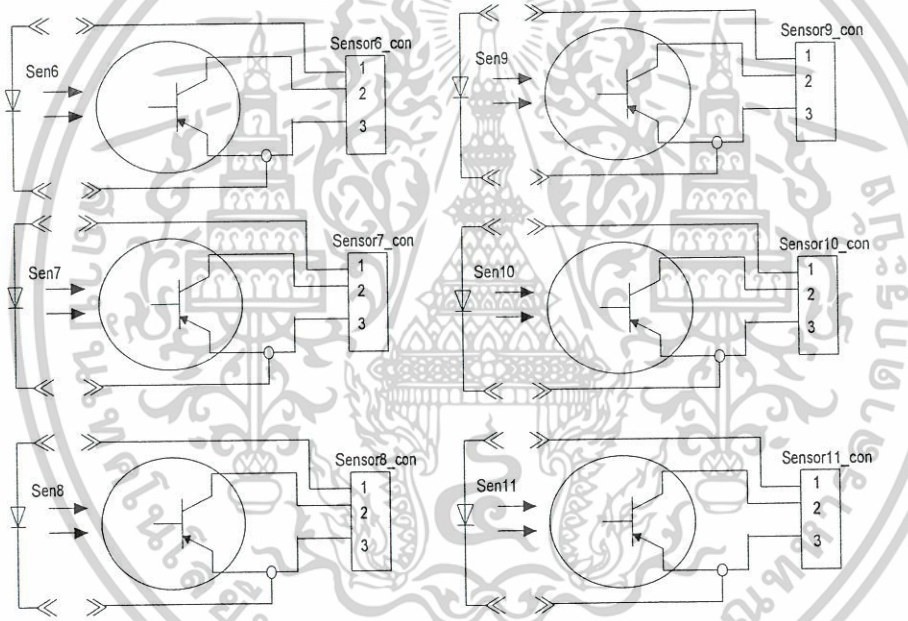


รูปที่ 3.20 วงจรตรวจจับแสงสะท้อนอินฟราเรดที่ใช้สำหรับเดินตามเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



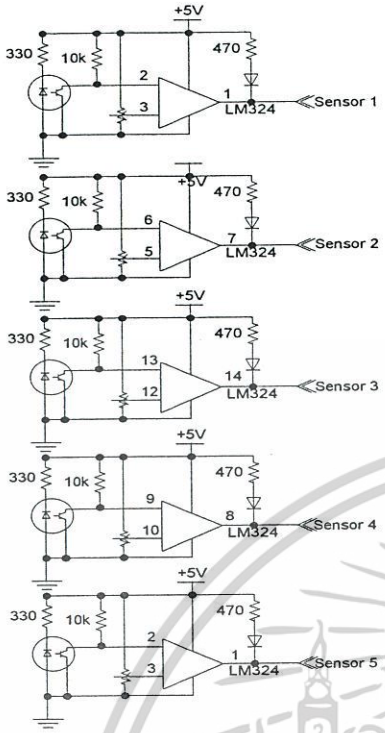
รูปที่ 3.21 วงจรตรวจจับแสงสะท้อนอินฟราเรดที่ใช้สำหรับตรวจจับของที่วางบนรถ



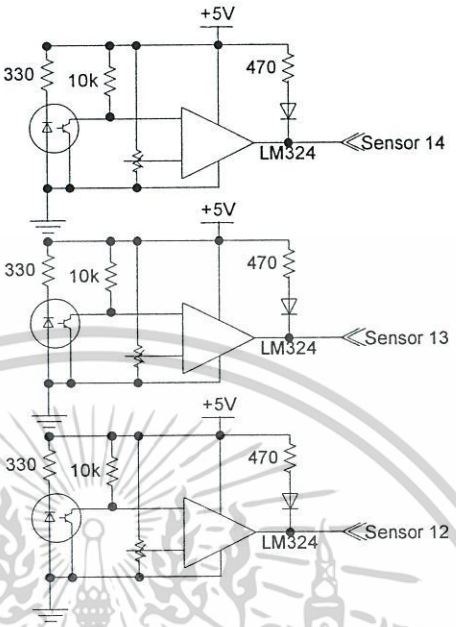
รูปที่ 3.22 วงจรรับสัญญาณให้เคลื่อนที่จากจุดรับของและวงจรรับสัญญาณให้หยุดจากไลน์การผลิต

วงจรเปรียบเทียบแรงดัน เนื่องจากแรงดันที่ได้รับจากวงจรตรวจจับแสงสะท้อนและรับสัญญาณแสง ไม่เพียงพอที่จะกำหนดเป็นลอจิก 0 หรือ 1 ที่จะส่งไปยัง Micro-controller ได้จึงต้องมีการปรับแรงดัน โดยนำเอาต์พุตที่ได้จากวงจรตรวจจับแสงสะท้อนมาเข้าออปแอมป์ LM324 เพื่อเปรียบเทียบแรงดัน โดยแรงดันที่มาเปรียบเทียบจะเป็น 0 โวลต์และ 5 โวลต์และเอาต์พุตที่ได้จากออปแอมป์จะส่งต่อไปยัง Micro-controller โดยมีลายวงจรดังนี้

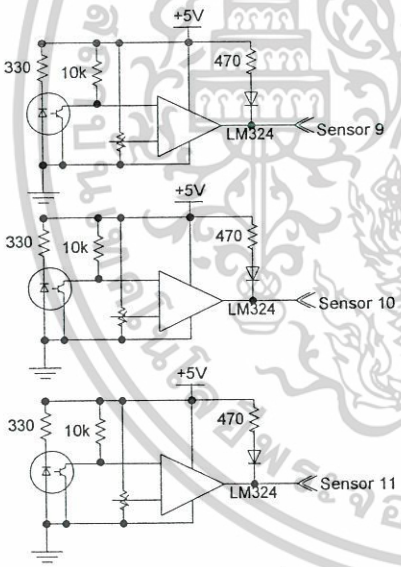
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



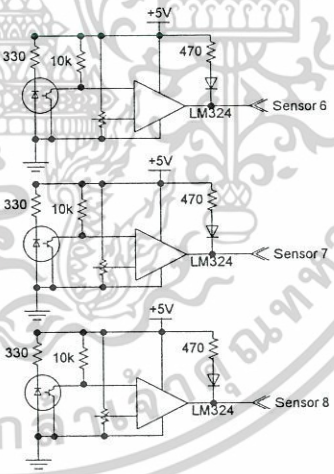
รูปที่ 3.23 ลายวงจร sensor เดินตามเส้น



รูปที่ 3.24 ลายวงจร sensor ตรวจเช็คกล่อง



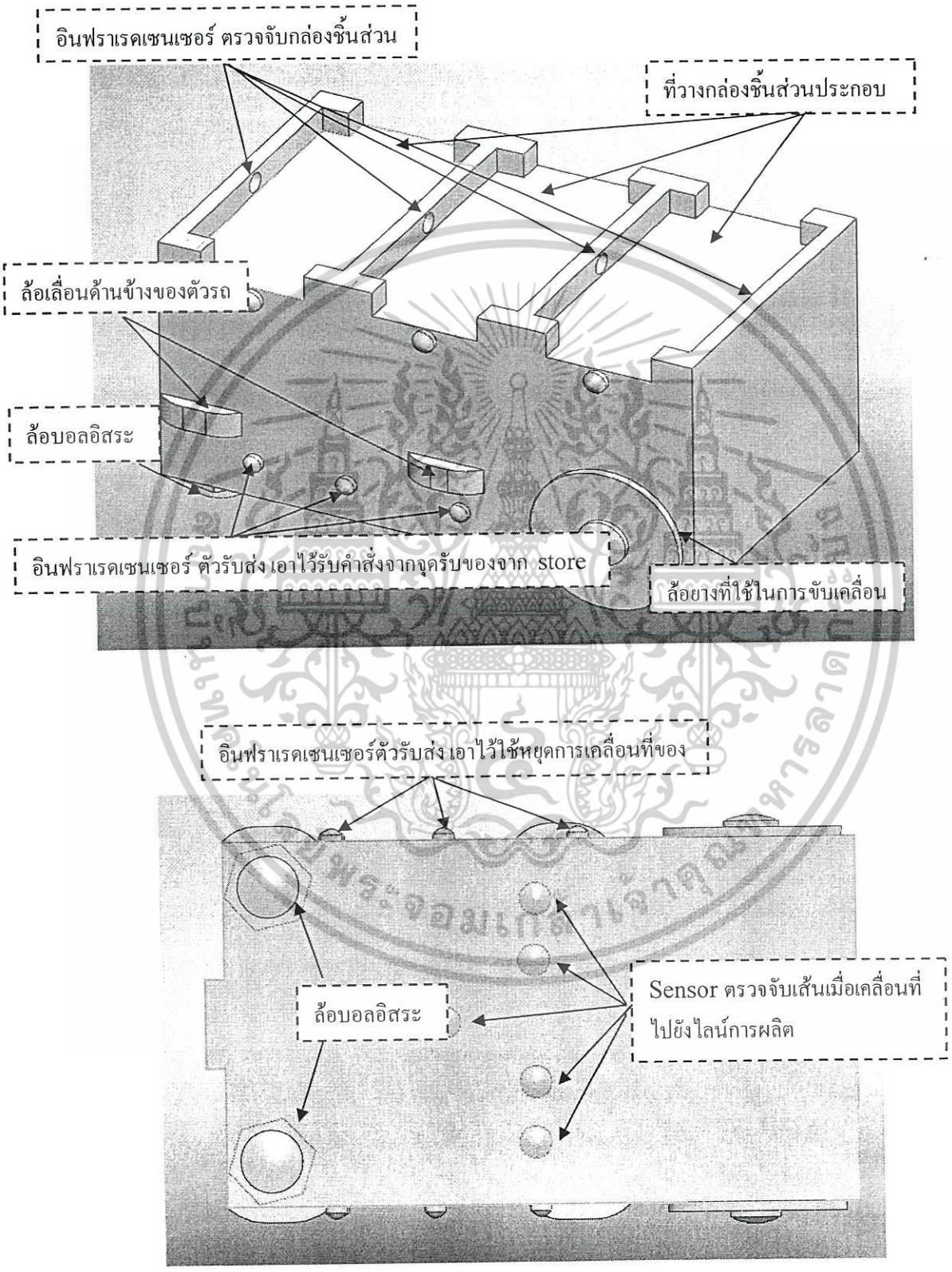
รูปที่ 3.25 ลายวงจร sensor รับคำสั่ง



รูปที่ 3.26 ลายวงจร sensor สำหรับหยุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

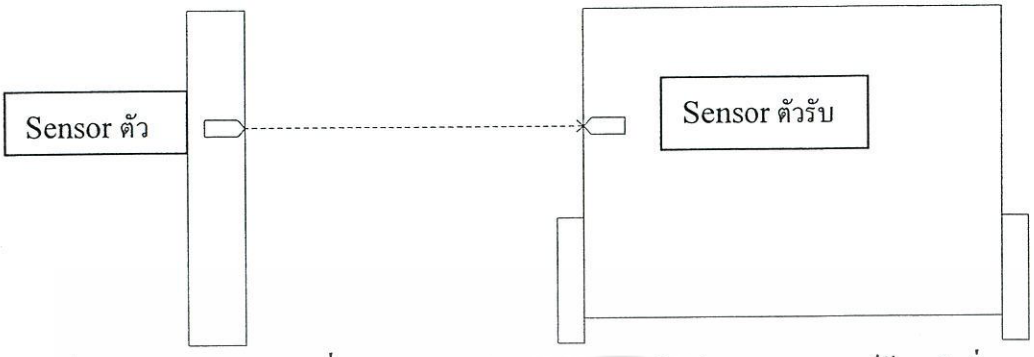
3.7 แนวทางการออกแบบและการทำงานของตัวรถขนส่งชิ้นส่วน



รูปที่3.27 โครงสร้างและส่วนประกอบของตัวรถขนส่งชิ้นส่วนประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

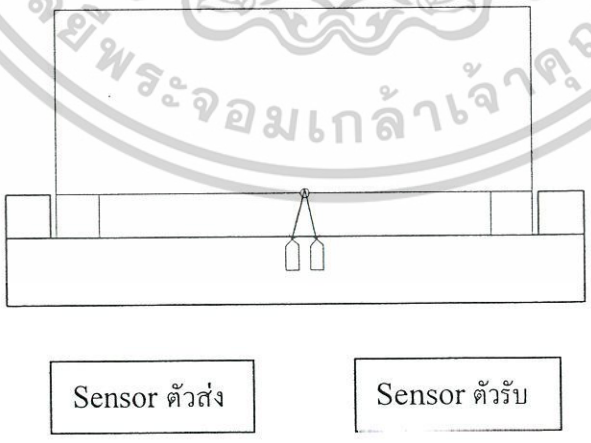
3.8 การทำงานงานของอินฟราเรดเซนเซอร์



รูปที่3.28 การรับส่งสัญญาณของอินฟราเรดเซนเซอร์ป้อนคำสั่ง

อินฟราเรดเซนเซอร์ที่ใช้ในการรับคำสั่งเลือกไลน์และตรวจจับของที่วางบนรถขนส่ง เมื่อมีการป้อนคำสั่งในการเลือกไลน์ที่จะให้นำของไปส่ง จะมีหลักการทำงานดังรูปคือ อินฟราเรดเซนเซอร์ตัวส่งที่จุดรับของจะส่งสัญญาณอินฟราเรดมาที่อินฟราเรดเซนเซอร์ตัวรับที่อยู่ด้านข้างของตัวรถ เมื่ออินฟราเรดเซนเซอร์ตัวรับได้รับสัญญาณจะมีค่าลอจิกเป็น 1 หรือ on และจะส่ง output เป็น 1 ไปที่แผง Micro-controller แล้ว Microcontroller จะโปรแกรมให้รถขนส่งชิ้นส่วนวิ่งตามคำสั่งไปยังไลน์นั้นๆ แต่ถ้าไม่มีการรับส่งสัญญาณอินฟราเรด จะมีค่าลอจิกเป็น 0 หรือ off

การที่รถขนส่งชิ้นส่วนจะเคลื่อนที่ได้นั้น จะต้องมีการวางของบนตัวรถและรับการป้อนคำสั่งจากข้างต้นก่อน โดยจะใช้อินฟราเรดเซนเซอร์ ที่อยู่ด้านบนของตัวรถเป็นตัวตรวจเช็คว่ามีกล่องวางอยู่บนรถหรือเปล่า ถ้าไม่มีของวางอยู่บนตัวรถ อินฟราเรดเซนเซอร์ก็จะไม่มีการตรวจจับแสงแล้วจะให้ค่าลอจิกออกมาเป็น 0 แต่ถ้ามีการวางของบนตัวรถ อินฟราเรดเซนเซอร์จะมีการตรวจจับแสงแล้วส่งค่าลอจิกเป็น 1 รอการรับคำสั่งในการเลือกจุดส่ง เพื่อทำการเคลื่อนที่ไปยังจุดส่งที่ต้องการต่อไป

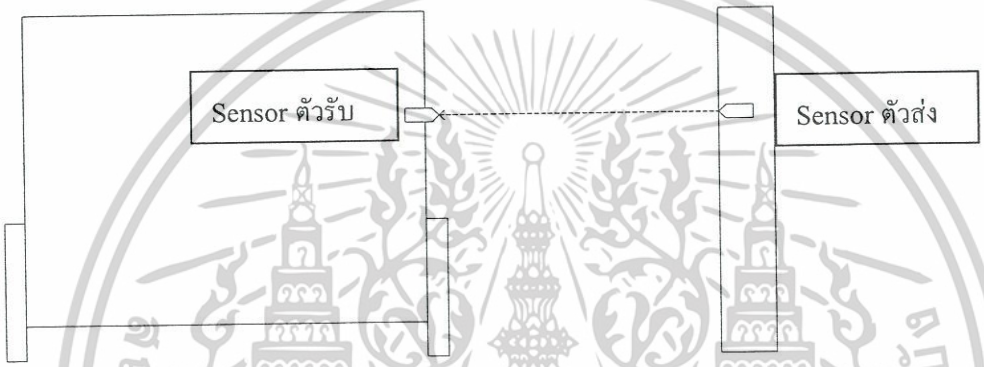


รูปที่3.29 แสดงการรับส่งสัญญาณของอินฟราเรดเซนเซอร์ตรวจเช็คของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินฟราเรดเซนเซอร์ที่ใช้คำสั่งให้หยุดในแต่ละไลน์

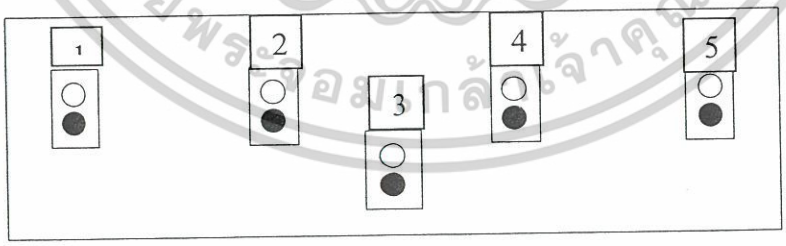
การหยุดของรถขนส่งเมื่อวิ่งไปถึงไลน์ที่ต้องการส่ง จะใช้อินฟราเรดเซนเซอร์ ส่งสัญญาณมาให้อินฟราเรดเซนเซอร์ตัวรับสัญญาณที่อยู่ข้างตัวรถขนส่ง ซึ่งมีลักษณะการทำงานเดียวกันกับการป้อนคำสั่งเลือกไลน์ที่จะส่ง คือ เมื่อ อินฟราเรดเซนเซอร์ ตัวรับสัญญาณยังไม่ได้รับสัญญาณจากอินฟราเรดเซนเซอร์ตัวส่งสัญญาณ จะมีค่าลอกจิกเป็น 0 แต่เมื่อรถขนส่งเคลื่อนที่เข้าไปยังไลน์ที่ต้องการส่ง ที่จุดส่งของจะมีการส่งสัญญาณมายัง อินฟราเรดเซนเซอร์ตัวรับที่อยู่ด้านข้างของตัวรถ ทำให้อินฟราเรดเซนเซอร์ตัวรับมีค่าลอจิกเป็น 1 และส่งค่าไปยัง Micro-controller ที่ตั้งโปรแกรมไว้ให้มีการหยุดของตัวรถขนส่งเมื่อมีการทำงานของอินฟราเรดเซนเซอร์ดังรูป



รูปที่3.30 แสดงการรับส่งสัญญาณของ sensor ที่ใช้หยุด

อินฟราเรดเซนเซอร์ที่ใช้ในการเดินตามเส้น

อินฟราเรดเซนเซอร์ที่ใช้มีทั้งหมด 5 ตัว โดยมีการวางรูปแบบดังรูป เพื่อใช้ในการตรวจจับเส้นสีดำ ซึ่งเส้นสีดำมี



รูปที่3.31 แสดงการวางตำแหน่งของเซนเซอร์ตรวจจับเส้น

มีขนาดความกว้างประมานครึ่งนิ้ว (ถ้าเซนเซอร์อยู่ในตำแหน่งเส้นสีดำ ค่าที่อ่านได้จะเป็น on แต่ถ้าเซนเซอร์อยู่ตำแหน่งนอกแถบสีดำ ค่าที่อ่านได้จะเป็น off) ในการควบคุมรถขนชิ้นส่วนให้เดินตามเส้น จะแบ่งลักษณะการตรวจจับออกเป็น 2 ลักษณะคือ เซนเซอร์ 2,3 และ 4 จะใช้ในการตรวจจับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นในการเดินตามเส้น ส่วนเซนเซอร์ 1 กับ 5 จะใช้ในการตรวจจับเส้นเพื่อทำการเลี้ยวเมื่อเจอทางแยกโดยลักษณะของเส้นที่เดินตามมีดังนี้

รูปที่3.32 แสดงลักษณะเส้นทางที่ใช้ในการตรวจจับ

ลักษณะการทำงานของเซนเซอร์ 2, 3 และ 4

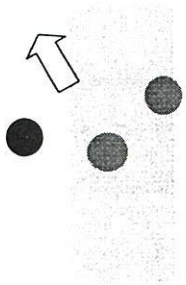
เมื่อรถขนส่งเดินตามเส้น จะมีลักษณะตำแหน่งของเซนเซอร์ทั้งสามตัวที่ใช้ตรวจจับเส้น ในการเดินเส้น และค่าที่อ่านได้ดังต่อไปนี้

รถขนส่งชิ้นส่วนเดินตรง

จากรูปตำแหน่งของตัวรถขนส่งชิ้นส่วนที่วางอยู่บนเส้นจะทำให้หุ่นเดินตามเส้นทาง โดยค่าเอาต์พุตที่ได้จากเซนเซอร์จะเป็นไปตามตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถขนส่งชิ้นส่วนเดินเอียงซ้าย



sensor2	sensor3	sensor4
0	1	1

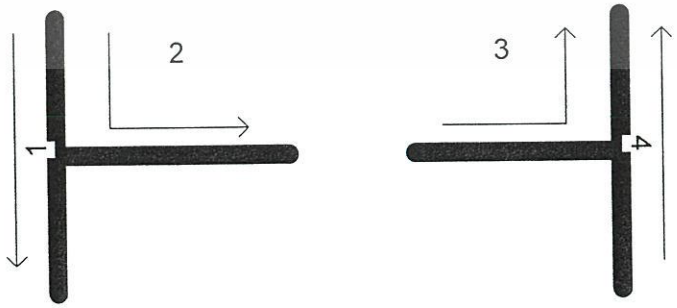
จากตำแหน่งของตัวรถขนส่งชิ้นส่วนที่วางอยู่บนเส้นจะทำให้ตัวรถขนส่งเดินเอียงซ้าย โดยค่าเอาต์พุตที่ได้จะเปลี่ยนที่ เซนเซอร์ตัวที่ 4 จะเปลี่ยนจาก 0 เป็น 1 เมื่อเทียบกับการเดินตรง

รถขนส่งชิ้นส่วนเดินเอียงขวา

sensor2	sensor3	sensor4
1	1	0

จากตำแหน่งของตัวรถขนส่งชิ้นส่วนที่วางอยู่บนเส้นจะทำให้ตัวรถขนส่งเดินเอียงขวา โดยค่าเอาต์พุตที่ได้จะเปลี่ยนที่ เซนเซอร์ตัวที่ 1 จะเปลี่ยนจาก 0 เป็น 1 เมื่อเทียบกับการเดินตรง

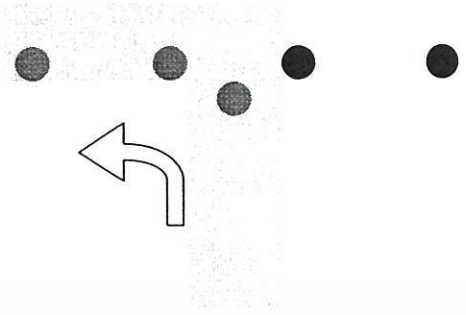
ลักษณะการทำงานของเซนเซอร์ 1 กับ 5



รูปที่3.33 แสดงลักษณะเส้นทางที่เซนเซอร์ที่ต้องเจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถขนส่งชิ้นส่วนเดินเลี้ยวซ้าย



sensor1	sensor2	sensor3	sensor4	sensor5
1	1	1	0	0

จากตำแหน่งของตัวรถขนส่งชิ้นส่วนที่วางอยู่บนเส้น จะทำให้ตัวรถขนส่งเดินเลี้ยวซ้าย เนื่องจาก เซนเซอร์ตัวที่ 1 มีค่าลอจิกเป็น 1 และ เซนเซอร์ตัวที่ 5 มีค่าลอจิกเป็น 0 ส่วนเซนเซอร์ตัวอื่นจะไม่นำมาใช้ในการเลือกว่าจะเลี้ยวซ้ายหรือเลี้ยวขวา โดยค่าเอาต์พุตที่ได้เป็นไปตามตาราง

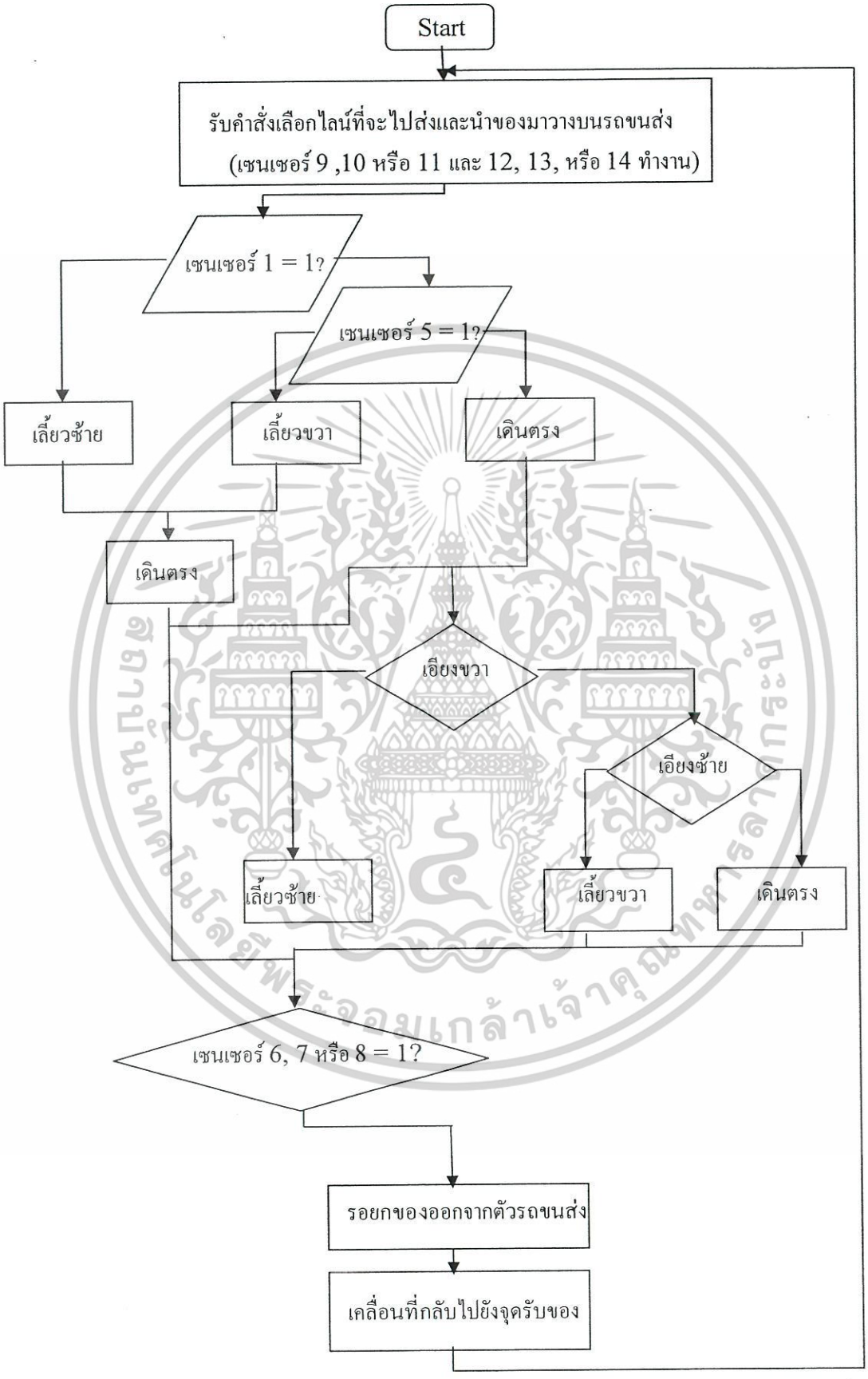
รถขนส่งชิ้นส่วนเดินเลี้ยวขวา

sensor1	sensor2	sensor3	sensor4	sensor5
0	0	1	1	1

จากตำแหน่งของตัวรถขนส่งชิ้นส่วนที่วางอยู่บนเส้นจะทำให้ตัวรถขนส่งเดินเลี้ยวขวา เนื่องจาก เซนเซอร์ตัวที่ 5 มีค่าลอจิกเป็น 1 และ เซนเซอร์ตัวที่ 1 มีค่าลอจิกเป็น 0 ส่วนเซนเซอร์ตัวอื่นจะไม่นำมาใช้ในการเลือกว่าจะเลี้ยวซ้ายหรือเลี้ยวขวา โดยค่าเอาต์พุตที่ได้เป็นไปตามตาราง

ลักษณะการควบคุมการทำงานของตัวรถขนส่งที่เขียนเป็นโฟลว์ชาร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.34 โฟลว์ชาร์ตการทำงานของรถขนส่งอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลอง

4.1 การคำนวณระยะการเคลื่อนที่

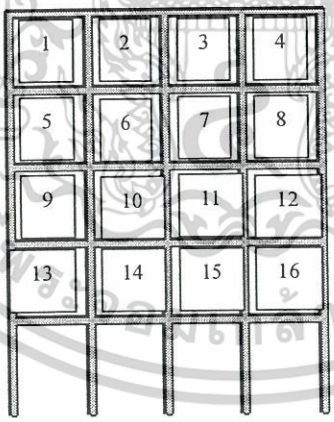
เนื่องจากที่สถานีของเครนอัตโนมัติ ต้องมีการควบคุมตำแหน่งการหยุดของชุดการเคลื่อนที่แกน X และแกน Y เพื่อให้ตรงกับำแหน่งกับช่องที่บรรจุชิ้นส่วนไว้ดังนั้นจากความสำพันธ์ระหว่าง L ระยะการเคลื่อนที่ = P ระยะพิท x N จำนวนรอบ (ระยะพิทของลีดสกรูที่ใช้เท่ากับ 1.5 มม.)

ทำให้ได้จำนวนรอบของลีดสกรู โดยในแต่ละรอบของการเคลื่อนที่ของลีดสกรูจะทำให้แถบดำตัดผ่านออปโต้เซนเซอร์ ซึ่งจะได้สัญญาณพัลซ์ออกมา 4 ลูก เพื่อส่งให้กับไมโครคอนโทรลเลอร์และสามารถควบคุมตำแหน่งได้ตามตางรางข้างล่าง

ตัวอย่างการคำนวณระยะการเคลื่อนที่

ตำแหน่งที่ 1 ของแกน X L = 225 = 1.5 x N

N = 150 รอบ และจำนวนพัลซ์ = 600



รูปที่ 4.1 แสดงตำแหน่งชั้นวางชิ้นส่วนประกอบ

กำหนดให้ ณ ตำแหน่งชั้นวางของหมายเลข 16 มีพิกัดของแกน X และY เท่ากับ (0,0)

พิกัด X,Y(มม.) เป็นพิกัดที่คำนวณได้

พิกัด X,Y(มม.)จริง เป็นพิกัดที่แกนXและY เคลื่อนที่ได้จริง

Pulse X,Y เป็นจำนวนพัลซ์ที่ได้จากการคำนวณ

เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าผิดพลาด เกิดจากค่าความคลาดเคลื่อนของพิกัด X,Y ที่เคลื่อนที่ได้จริงเปรียบเทียบกับพิกัด X,Y ที่ได้จากการคำนวณ

ตารางที่ 4.1 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y ครั้งที่ 1

ตำแหน่ง	พิกัด X,Y(มม.)	Pulse X,Y	พิกัด X,Y(มม.)จริง	ค่าผิดพลาด
1	225 , 225	600 , 600	225.532 , 225.017	+0.532 , +0.170
2	150 , 225	400 , 600	150.574 , 224.562	+0.574 , -0.438
3	75 , 225	200 , 600	74.873 , 225.250	-0.127 , +0.025
4	0 , 225	0 , 600	0 , 225.037	0 , +0.370
5	225 ,150	600 , 400	226.043 , 150.038	+1.043 , +0.038
6	150 , 150	400 , 400	150.105 , 151.010	+0.105 , +1.010
7	75 , 150	200 , 400	75.070 , 151.150	+0.070 , +1.150
8	0 , 150	0 , 400	0 , 150.196	0 , +0.960
9	225 , 75	600 , 200	226.028 , 74.864	+1.028 , -0.136
10	150 , 75	400 , 200	150.155 , 75.040	+0.155 , +0.400
11	75 , 75	200 , 200	75.020 , 75.018	+0.020 , +0.018
12	0 , 75	0 , 200	0 , 75.595	0 , +0.595
13	225 , 0	600 , 0	224.552 , 0	-0.448 , 0
14	150 , 0	400 , 0	151.173 , 0	+1.173 , 0
15	75 , 0	200 , 0	75.037 , 0	+0.037 , 0
16	0 , 0	0 , 0	0 , 0	0 , 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y ครั้งที่ 2

ตำแหน่ง	พิกัด X,Y(มม.)	Pulse X,Y	พิกัด X,Y(มม.)จริง	ค่าผิดพลาด
1	225 , 225	600 , 600	225.695 , 225.588	+0.695 , +0.588
2	150 , 225	400 , 600	150.390, 224.855	+0.390 , -0.145
3	75 , 225	200 , 600	74.424, 225.455	-0.576 , +0.455
4	0 , 225	0 , 600	0 , 225.900	0 , +0.900
5	225 ,150	600 , 400	226.055, 150.566	+1.055 , +0.566
6	150 , 150	400 , 400	150.885, 151.103	+0.885 , +1.103
7	75 , 150	200 , 400	75.104, 150.996	+0.104 , +1.996
8	0 , 150	0 , 400	0 , 150.221	0 , +0.221
9	225 , 75	600 , 200	226.096, 74.459	+1.096 , -0.551
10	150 , 75	400 , 200	150.155 , 75.040	+0.155 , +0.40
11	75 , 75	200 , 200	75.109, 75.015	+0.109 , +0.0.15
12	0 , 75	0 , 200	0 , 75.980	0 , +0.980
13	225 , 0	600 , 0	224.634, 0	-0.366 , 0
14	150 , 0	400 , 0	151.677, 0	+1.677 , 0
15	75 , 0	200 , 0	75.254, 0	+0.254 , 0
16	0 , 0	0 , 0	0 , 0	0 , 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y ครั้งที่ 3

ตำแหน่ง	พิกัด X,Y(มม.)	Pulse X,Y	พิกัด X,Y(มม.)จริง	ค่าผิดพลาด
1	225 , 225	600 , 600	225.856 , 225.965	+0.856 , +0.965
2	150 , 225	400 , 600	149.788, 225.054	-0.212 , -0.054
3	75 , 225	200 , 600	74.100, 225.995	-0.900 , +0.995
4	0 , 225	0 , 600	0 , 226.030	0 , +1.030
5	225 ,150	600 , 400	226.104, 150.100	+1.104 , +0.100
6	150 , 150	400 , 400	150.012, 151.113	+0.012 , +1.113
7	75 , 150	200 , 400	74.886, 150.034	-0.114 , +1.034
8	0 , 150	0 , 400	0 , 150.600	0 , +0.600
9	225 , 75	600 , 200	225.545, 74.565	+0.545 , -0.435
10	150 , 75	400 , 200	150.345, 75.098	+0.345 , +0.098
11	75 , 75	200 , 200	75.234, 75.341	+0.234 , +0.341
12	0 , 75	0 , 200	0 , 75.012	0 , +0.012
13	225 , 0	600 , 0	224.998, 0	-0.002 , 0
14	150 , 0	400 , 0	151.155, 0	+1.155 , 0
15	75 , 0	200 , 0	75.364, 0	+0.364 , 0
16	0 , 0	0 , 0	0 , 0	0 , 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 การแสดงผลการทดลองตำแหน่งการเคลื่อนที่ของแกน X และ แกน Y (ค่าเฉลี่ย)

ตำแหน่ง	พิกัด X,Y(มม.)	Pulse X,Y	พิกัด X,Y(มม.)จริง	ค่าผิดพลาด
1	225 , 225	600 , 600	225.694 , 225.523	+0.694 , 0.523
2	150 , 225	400 , 600	150.250 , 224.823	+0.250 , -0.177
3	75 , 225	200 , 600	74.465 , 225.563	-0.54, +0.563
4	0 , 225	0 , 600	0 , 225.655	0 , +0.655
5	225 ,150	600 , 400	226.067 , 150.233	+1.067 , +0.223
6	150 , 150	400 , 400	150.333 , 151.082	+0.333 , +1.028
7	75 , 150	200 , 400	75.020 , 150.726	+0.020 , +0.726
8	0 , 150	0 , 400	0 , 150.339	0 , +0.339
9	225 , 75	600 , 200	225.889 , 74.629	+0.889
10	150 , 75	400 , 200	150.218 , 75.059	+0.218 , +0.059
11	75 , 75	200 , 200	75.121, 75.124	+0.121 , +0.124
12	0 , 75	0 , 200	0 , 75.529	0 , +0.529
13	225 , 0	600 , 0	224.728 , 0	-0.272 , 0
14	150 , 0	400 , 0	151.351 , 0	+1.351 , 0
15	75 , 0	200 , 0	75.218	+0.218
16	0 , 0	0 , 0	0 , 0	0 , 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ครั้งที่ 1

ไลน์การผลิตที่	เวลารถขาไป (วินาที)	เวลารอส่งของ(คิดเป็นเวลาหน่วงเมื่อส่งของเสร็จ)/(วินาที)	เวลาขากลับ (วินาที)
1	4.95	5	6.45
2	10.68	5	12.65

ตารางที่ 4.6 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ครั้งที่ 2

ไลน์การผลิตที่	เวลารถขาไป (วินาที)	เวลารอส่งของ(คิดเป็นเวลาหน่วงเมื่อส่งของเสร็จ)/(วินาที)	เวลาขากลับ (วินาที)
1	5.21	5	6.80
2	10.25	5	12.20

ตารางที่ 4.7 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์ครั้งที่ 3

ไลน์การผลิตที่	เวลารถขาไป (วินาที)	เวลารอส่งของ(คิดเป็นเวลาหน่วงเมื่อส่งของเสร็จ)/(วินาที)	เวลาขากลับ (วินาที)
1	4.75	5	6.37
2	11.02	5	12.90

ตารางที่ 4.8 การแสดงผลการทดลองการเคลื่อนที่ของตัวรถไปยังไลน์การผลิตแต่ละไลน์โดยเฉลี่ย

ไลน์การผลิตที่	เวลารถขาไป (วินาที)	เวลารอส่งของ(คิดเป็นเวลาหน่วงเมื่อส่งของเสร็จ)/(วินาที)	เวลาขากลับ (วินาที)
1	4.97	5	6.54
2	10.65	5	12.58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและบทวิจารณ์

5.1 สรุปผลการทดลอง

ปริญญานิพนธ์ฉบับนี้ได้นำเสนอและศึกษาเรื่องแบบจำลองที่ใช้ขนส่งในไลน์การผลิต การศึกษาในที่นี้ได้ออกแบบแบบจำลองระบบซึ่งสามารถควบคุมการทำงานผ่านทางคอมพิวเตอร์ ซึ่งเป็นการติดต่อผ่านทางพอร์ทอนุกรม โดยในส่วนของการติดต่อนี้ได้ทดลองเขียนโปรแกรมเพื่อรับค่าจากโปรแกรมvisual basic 6.0เพื่อใช้เป็นการควบคุมการทำงานซึ่งผลที่ได้คือสามารถควบคุมการทำงานของตัวคอนโทรลเลอร์ได้ โดยโปรแกรมในส่วนของตัวMCS 51 ซึ่งเป็นตัวโปรแกรมหลักที่ใช้ควบคุมในส่วนของเครนอัตโนมัติ หลังจากที่ได้ทำการทดลองเขียนโปรแกรมหลายครั้งจนได้โปรแกรมที่สมบูรณ์พบว่าระบบสามารถทำงานตามที่ได้ออกแบบและกำหนดกระบวนการไว้

ในส่วนของรถที่ใช้ในการขนส่งกล่องบรรจุชิ้นส่วน เมื่อออกแบบโครงสร้างรถเป็นที่เรียบร้อยแล้วก็ทำการสร้างตัวรถขึ้นมา จากนั้นก็สร้างวงจร ไมโครคอนโทรลเลอร์ เพื่อใช้ควบคุมกระบวนการทำงาน โดยเมื่อทำการเขียนโปรแกรมและทดลองให้รับกล่องบรรจุชิ้นส่วนจากชุดเครนอัตโนมัติ พบว่ารถขนส่งนั้นสามารถวิ่งไปตามเส้นที่กำหนดไว้ได้แต่จะมีปัญหาเมื่อรถวิ่งผ่านตรงทางแยก เนื่องจากตัวเซนเซอร์ตรวจจับทางแยกได้ไม่ค่อยดีนัก จึงได้ทำการเปลี่ยนทางแยกให้เป็นเส้นโค้งเพื่อให้รถสามารถเลี้ยวได้ง่ายขึ้น เมื่อทำการแก้ไขแล้วจึงทำให้ระบบทั้งระบบสามารถทำงานได้สัมพันธ์กันได้โดยใช้ผู้ควบคุมเพียงคนเดียว

5.2 แนวทางการพัฒนา

การนำหลักการที่ได้จากการทำโครงงานชิ้นนี้ สามารถนำไปประยุกต์ใช้ได้ในอุตสาหกรรมหลายๆอย่าง ที่เป็นระบบอัตโนมัติ ทั้งในส่วนของที่เป็นเครนอัตโนมัติ และ ในส่วนของรถขนส่งอัตโนมัติ นอกจากนี้สามารถนำไปต่อยอดเพื่อปรับปรุงประสิทธิภาพการทำงานให้สูงขึ้น ตรงส่วนของชุดเครนอัตโนมัติอาจเปลี่ยนมอเตอร์ที่ใช้ขับลีดสกรูที่เป็นระบบเซอร์โว เพื่อให้ได้ตำแหน่งที่แม่นยำขึ้น และสามารถควบคุมความเร็วในการเคลื่อนที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ปัญหาในการทำงาน

1 เนื่องจากวัสดุที่ใช้ในการออกแบบไว้ในตอนแรกนั้น เมื่อมีการสอบถามราคาจากตัวแทนจำหน่าย ราคาวัสดุนั้นมีราคาแพงจึงต้องเสียเวลาในการหาวัสดุอื่นเพื่อใช้ทดแทน เพื่อให้ใกล้เคียงกับวัสดุที่ได้ออกแบบไว้ในตอนแรก

2 การประกอบชิ้นส่วนของสถานีอัต โนมัติจะไม่ได้ระดับอันเป็นผลมาจากการตัดแผ่นอะคิลิกไม่ได้มุมองศาที่แน่นอน แต่ก็สามารถใช้งานได้

3 เกลียวที่ใช้เป็นเป็นลีดสกรู เมื่อทำการกลึงมาแล้วชิ้นงานที่ได้มีเกลียวที่ไม่ค่อยสม่ำเสมอทำให้เวลาเกลียวหมุนการเคลื่อนที่ของแกนจะไม่ค่อยราบเรียบ

4 เครื่องมือในการทำงานในห้องปฏิบัติการของภาควิชามีเครื่องในการทำงานไม่ค่อยเพียงพอทำให้ต้องเสียเวลาในการหาเครื่องมือมาใช้เอง รวมทั้งเครื่องมือการทำงานบางเครื่องอยู่ในสภาพที่ไม่สมบูรณ์ทำให้งานออกมาไม่ได้ตามแบบที่ได้วางไว้

5 การนับสัญญาณพัลซ์ของคอนโทรลเลอร์บางครั้งมีความคลาดเคลื่อน

6 โปรแกรมที่เขียนขึ้นบางครั้งเกิดการerrorขึ้นที่ตัวคอนโทรลเลอร์ทำให้เกิดการทำงานที่ผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก. โปรแกรมควบคุมเครนอัตโนมัติ

```
#include<reg51.h>
unsigned int one,two,three,duty_time; // name of counter
unsigned char pst1,pst2,pst3,pst4,pst5,pst6,pst7,pst8,pst9,pst10,pst11,pst12,pst13,pst14,pst15,pst16,command ,inChar;
sbit        sensor1 = P3^2; // x
sbit        sensor2 = P3^3; // y

sbit        en1 = P1^0;   //          x
sbit        en2 = P1^1;             // y
sbit        en3 = P0^0;             // z
sbit        en4 = P0^1;             // 180 degree
sbit        motor1_f = P1^2;        // x
sbit        motor1_b = P1^3;
sbit  motor2_f = P1^5;  // y
sbit  motor2_b = P1^4;
sbit        motor3_f = P0^2;        // z
sbit        motor3_b = P0^3;
sbit        motor4_f = P0^4;        // 180 degree
sbit        motor4_b = P0^5;
sbit        limit1 = P2^1;
sbit        limit2 = P2^0;
sbit        limit3 = P2^2;
sbit        limit4 = P2^3;
sbit        limit5 = P2^4;
////////////////////////////////////
//          FUNCTION
////////////////////////////////////
void init_motor(void)
{
  en1 = 1;
  en2 = 1;
            en3 = 1;
            en4 = 1;
}
void unit_motor(void)
{
  en1 = 0;
  en2 = 0;
            en3 = 0;
            en4 = 0;
}

void motor1_forward (void)
            {
                        motor1_f = 1;
                        motor1_b = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
void motor1_reward (void)
        {
                motor1_f = 0;
                motor1_b = 1;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
void motor1_stop (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
// motor2
void motor2_forward (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 1;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
 void motor2_reward (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                motor2_b = 1;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
void motor2_stop (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
// motor3
void motor3_forward (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 1;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
 void motor3_reward (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 1;
                motor4_f = 0;
                motor4_b = 0;
        }
  void motor3  stop (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }

// motor4
void motor4_forward (void)
        {

                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 1;
                motor4_b = 0;
        }
void motor4_reward (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 1;
        }
void motor4_stop (void)
        {
                motor1_f = 0;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 0;
                motor3_f = 0;
                motor3_b = 0;
                motor4_f = 0;
                motor4_b = 0;
        }
void delay(unsigned int x)
{
        unsigned int time=50000;
        while(x--)
        while(time--);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void delay1 (unsigned char tick)
{
        unsigned char i,j;
        for(i=0;i<tick;i++)
        for(j=0;j<200;j++);
}
void delay2(unsigned int x)
{
        unsigned int time=25000;
        while(x--)
        while(time--);
}

//////////////////////
// counter
//////////////////////
void check_xorderone(void)
{

 sensor1=1;
        one=0;
        while(one<=600)
        {
                if(sensor1 == 0 )
                {
                        delay1(10);
                        if(sensor1 == 0 )
                        {
                                while(sensor1==1);
                                one++;
                                while(sensor1==0);
                        }
                }
        }
 }
 void check_yorderone(void)
 {

        sensor2=1;
        two=0;
        while(two<=575)
        {
                if(sensor2 == 0 )
                {
                        delay1(10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        if(sensor2 == 0 )
                        {
                                while(sensor2==1);
                                two++;
                                while(sensor2==0);
                        }
                }
        }
}
//////////////////////////
void check_xordertwo(void)
{

        sensor1=1;
        one=0;
        while(one<=400)
        {
                if(sensor1 == 0 )
                {
                        delay1(10);
                        if(sensor1 == 0 )
                        {
                                while(sensor1==1);
                                one++;
                                while(sensor1==0);
                        }
                }
        }
}
void check_yordertwo(void)
{

        sensor2=1;
        two=0;
        while(two<=375)
        {
                if(sensor2 == 0 )
                {
                        delay1(10);
                        if(sensor2 == 0 )
                        {
                                while(sensor2==1);
                                two++;
                                while(sensor2==0);
                        }
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
}
//////////////////////////
void check_xorderthree(void)
{

        sensor1=1;
        one=0;
        while(one<=200)
        {
                if(sensor1 == 0 )
                {
                        delay1(10);
                        if(sensor1 == 0 )
                        {
                                while(sensor1==1);
                                one++;
                                while(sensor1==0);
                        }
                }
        }
}
void check_yorderthree(void)
{

        sensor2=1;
        two=0;
        while(two<=175)
        {
                if(sensor2 == 0 )
                {
                        delay1(10);
                        if(sensor2 == 0 )
                        {
                                while(sensor2==1);
                                two++;
                                while(sensor2==0);
                        }
                }
        }
}

////////////////////////////// HOME

void check_xhome(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        limit5=1;
        while(limit5==1)
        {
                motor1_reward();
                delay2(1);
        }
        motor1_stop();


}
void check_yhomeone(void)
{

        sensor2=1;
        two=0;
        while(two<=575)
        {
                if(sensor2 == 0 )
                {
                        delay1(10);
                        if(sensor2 == 0 )
                        {
                                while(sensor2==1);
                                two++;
                                while(sensor2==0);
                        }
                }
        }
}
 void check_yhometwo(void)
{

        sensor2=1;
        two=0;
        while(two<=370)
        {
                if(sensor2 == 0 )
                {
                        delay1(10);
                        if(sensor2 == 0 )
                        {
                                while(sensor2==1);
                                two++;
                                while(sensor2==0);
                        }
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
}
void check_yhomethree(void)
{

        sensor2=1;
        two=0;
        while(two<=175)
        {
                if(sensor2 == 0 )
                {
                        delay1(10);
                        if(sensor2 == 0 )
                        {
                                while(sensor2==1);
                                two++;
                                while(sensor2==0);
                        }
                }
        }
}
////////////////////////////
// DOWNLOAD AND UPLOAD
////////////////////////////

void upload(void)                                       //UPLOAD
{
        while(limit1==1)
                {
                        motor3_forward();
                }
        motor3_stop();
        motor2_forward();
        delay2(2);
        motor2_stop();
        delay(3);
        while(limit2==1)
                {
                        motor3_reward();
                }
        motor3_stop();

}
void download(void)                                     //DOWNLOAD
{
        while(limit4==1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                        motor4_reward();
                }
        motor4_stop();
        delay(2);
        motor2_reward();
        delay2(2);
        motor2_stop();
        delay(2);
        while(limit1==1)
         {
                        motor3_forward();
         }
        motor3_stop();
        delay(2);
        motor4_forward();
        delay(4);
        motor4_stop();
        delay(2);
        while(limit2==1)
          {
                motor3_reward();
          }
        motor3_stop();
        delay(2);
        while(limit3==1)
          {
                        motor4_forward();
                }
        motor4_stop();
        delay(2);
}
////////////////////////////
// position
////////////////////////////

void positionone(void)                          //ROW 1
{

                        delay(2);
                        motor1_forward();
                        check_xorderone();
                        motor1_stop();
                        delay(2);
                        motor2_forward();
                        check_yorderone();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomeone();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positiontwo(void)
{
                    delay(2);
                    motor1_forward();
                    check_xordertwo();
                    motor1_stop();
                    delay(2);
                    motor2_forward();
                    check_yorderone();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomeone();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positionthree(void)
{
                    delay(2);
                    motor1_forward();
                    check_xorderthree();
                    motor1_stop();
                    delay(2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    motor2_forward();
                    check_yorderone();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomeone();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positionfour(void)
{

                    delay(2);
                    motor2_forward();
                    check_yorderone();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomeone();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}
//////////////////////
//////////////////////
void positionfive(void)                    //ROW 2
{

                    delay(2);
                    motor1_forward();
                    check_xorderone();
                    motor1_stop();
                    delay(2);
                    motor2_forward();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    check_yordertwo();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhometwo();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positionsix(void)
{
                    delay(2);
                    motor1_forward();
                    check_xordertwo();
                    motor1_stop();
                    delay(2);
                    motor2_forward();
                    check_yordertwo();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhometwo();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
          download();
}

void positionseven(void)
{
                    delay(2);
                    motor1_forward();
                    check_xorderthree();
                    motor1_stop();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                delay(2);
                motor2_forward();
                check_yordertwo();
                motor2_stop();
                delay(2);
                upload();
                delay(2);
                motor2_reward();
                check_yhometwo();
                motor2_stop();
                delay(2);
                check_xhome();
                motor1_stop();
                delay(2);
                download();
}

void positioneight(void)
{

                delay(2);
                motor2_forward();
                check_yordertwo();
                motor2_stop();
                delay(2);
                upload();
                delay(2);
                motor2_reward();
                check_yhometwo();
                motor2_stop();
                delay(2);
                check_xhome();
                motor1_stop();
                delay(2);
                download();

}
//////////////////////
//////////////////////
void positionnine(void)                                    //ROW3
{

                delay(2);
                motor1_forward();
                check_xorderone();
                motor1_stop();
                delay(2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    motor2_forward();
                    check_yorderthree();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomethree();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positionten(void)
{

                    delay(2);
                    motor1_forward();
                    check_xordertwo();
                    motor1_stop();
                    delay(2);
                    motor2_forward();
                    check_yorderthree();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomethree();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positioneleven(void)
{

                    delay(2);
                    motor1_forward();
                    check_xorderthree();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    motor1_stop();
                    delay(2);
                    motor2_forward();
                    check_yorderthree();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomethree();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}

void positiontwelve(void)
{
                    delay(2);
                    motor2_forward();
                    check_yorderthree();
                    motor2_stop();
                    delay(2);
                    upload();
                    delay(2);
                    motor2_reward();
                    check_yhomethree();
                    motor2_stop();
                    delay(2);
                    check_xhome();
                    motor1_stop();
                    delay(2);
                    download();
}
//////////////////////////
//////////////////////////
void positionthirteen(void)                    //ROW 4
{
                    delay(2);
                    motor1_forward();
                    check_xorderone();
                    motor1_stop();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                delay(2);
        upload();
                delay(2);
                check_xhome();
                motor1_stop();
                delay(2);
                download();
}

void positionfourteen(void)
{
                delay(2);
                motor1_forward();
                check_xordertwo();
                motor1_stop();
                delay(2);
                upload();
                delay(2);
                check_xhome();
                motor1_stop();
                delay(2);
                download();
}

void positionfifteen(void)
{
                delay(2);
                motor1_forward();
                check_xorderthree();
                motor1_stop();
                delay(2);
                upload();
                delay(2);
                check_xhome();
                motor1_stop();
                delay(2);
                download();
}

void positiontsixteen(void)
{
                delay(2);
                upload();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                delay(2);
                download();
}
//Function receive
void main (void)
{
        while(1)
  {
                unsigned char dat = 0;
                init_motor();
                TMOD = 0x21;
                SCON = 0x50;
                TH1 = 0xFD;
                RI=0;
                TI=0;
                TR1=1;
                pst1=0;
                pst2=0;
                pst3=0;
                pst4=0;
                pst5=0;
                pst6=0;
                pst7=0;
                pst8=0;
                pst9=0;
                pst10=0;
                pst11=0;
                pst12=0;
                pst13=0;
                pst14=0;
                pst15=0;
                pst16=0;
                while(dat!='T')
        {
                        dat=0;
                        while(~RI);
                        RI=0;
                        dat=SBUF;
                        if(dat=='U')
                        {
                                pst1=0;
                                pst2=0;
                                pst3=0;
                                pst4=0;
                                pst5=0;
                                pst6=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                pst7=0;
                pst8=0;
                pst9=0;
                pst10=0;
                pst11=0;
                pst12=0;
                pst13=0;
                pst14=0;
                pst15=0;
                pst16=0;
        }
        if(dat=='A')
        {
                pst1=1;
        }
        if(dat=='B')
        {
                pst2=1;
        }
        if(dat=='C')
        {
                pst3=1;
        }
        if(dat=='D')
        {
                pst4=1;
        }
        if(dat=='E')
        {
                pst5=1;
        }
        if(dat=='F')
        {
                pst6=1;
        }
        if(dat=='G')
        {
                pst7=1;
        }
        if(dat=='H')
        {
                pst8=1;
        }
        if(dat=='I')
        {
                pst9=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
        if(dat=='J')
        {
                pst10=1;
        }
        if(dat=='K')
        {
                pst11=1;
        }
        if(dat=='L')
        {
                pst12=1;
        }
        if(dat=='M')
        {
                pst13=1;
        }
        if(dat=='N')
        {
                pst14=1;
        }
        if(dat=='O')
        {
                pst15=1;
        }
        if(dat=='P')
        {
                pst16=1;
        }
}
if(pst1==1)
{
        positionone();
}

if(pst2==1)
{
        positiontwo();
}
if(pst3==1)
{
        positionthree();
}
if(pst4==1)
{
        positionfour();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
if(pst5==1)
{
        positionfive();
}
if(pst6==1)
{
        positionsix();
}
if(pst7==1)
{
        positionseven();
}
if(pst8==1)
{
        positioneight();
}
if(pst9==1)
{
        positionnine();
}
if(pst10==1)
{
        positionten();
}
if(pst11==1)
{
        positioneleven();
}
if(pst12==1)
{
        positiontwelve();
}
if(pst13==1)
{
        positionthirteen();
}
if(pst14==1)
{
        positionfourteen();
}
if(pst15==1)
{
        positionfifteen();
}
if(pst16==1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                        positiontsixteen();
                }
                dat='0';
                unit_motor();
                }
}
```

ข. โปรแกรมควบคุมรถขนส่งชิ้นส่วนอัตโนมัติ

```
#include<reg51.h>

sbit        sensor1 = P3^0;
sbit        sensor2 = P3^1;
sbit        sensor3 = P3^2;
sbit        sensor4 = P3^3;
sbit        sensor5 = P3^4;
sbit        sensor14 = P3^5;
sbit        sensor6 = P2^5;
sbit        sensor7 = P2^4;
sbit        sensor8 = P2^3;
sbit        sensor9 = P2^2;
sbit        sensor10 = P2^1;
sbit        sensor11 = P2^0;
sbit        sensor12 = P2^6;
sbit        sensor13 = P2^7;

sbit        check1 = P0^0;
sbit        check2 = P0^1;
sbit        check3 = P0^2;
sbit        check4 = P0^3;
sbit        check5 = P0^4;
sbit        check6 = P0^5;
sbit        check7 = P0^6;
sbit        check8 = P0^7;

sbit        en1 = P1^0;
sbit        en2 = P1^1;
sbit        motor1_f = P1^2;
sbit        motor1_b = P1^3;
sbit  motor2_b = P1^4;
sbit  motor2_f = P1^5;
sbit        led = P1^7;

int         i,j,k,l,t,m,e,s,b,a,c;
////////////////////////////////////////////////////////////
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
///                    Test motor
//////////////////////////////////////////////////////////
void testmotor1(void)
        {                       //en1 = 1;
                                //en2 = 0;
                                motor1_f = 1;
                                motor1_b = 0;
                                motor2_f = 0;
                                motor2_b = 1;
        }

void testmotor2(void)
        {
                                //en1 = 0;
                                //en2 = 1;
                                motor1_f = 0;
                                motor1_b = 1;
                                motor2_f = 1;
                                motor2_f = 0;
        }
//////////////////////////////////////////////////////////
///    program delay
//////////////////////////////////////////////////////////

void delay(unsigned int x)
{
        unsigned int time = 25000;//50000
        while(x--)
        while(time--);
}

void delay1(unsigned int x)
{
        unsigned int time = 10000;//50000
        while(x--)
        while(time--);
}
//////////////////////////////////////////////////////////
///                    stop
//////////////////////////////////////////////////////////

void stop(void)
        {
                        en1 = 0;
                        en2 = 0;
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void start(void)
        {
                en1 = 1;
                en2 = 2;
        }

//////////////////////////////////////////////////////
///             forward
//////////////////////////////////////////////////////
void forward(void)
        {
                //en1 = 1;
                //en2 = 1;
                motor1_f = 1;
                motor1_b = 0;
                motor2_f = 1;
                motor2_b = 0;
        }

//////////////////////////////////////////////////////
///   turnleft && turnright
//////////////////////////////////////////////////////
void turnl(void)
        {

                motor1_f = 0;
                motor1_b = 1;
                motor2_f = 1;
                motor2_b = 0;

        }

void turnr(void)
                {
                motor1_f = 1;
                motor1_b = 0;
                motor2_f = 0;
                motor2_b = 1;

        }

//////////////////////////////////////////////////////
///       track line
//////////////////////////////////////////////////////

void track(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
//if((sensor2 == 1)&&(sensor3 == 1)&&(sensor4 == 1))
//                      {
//                              stop();
//                      }
        if((sensor2 == 0)&&(sensor4 == 0))
                      {
                              forward();
                      }
        else if((sensor2 == 1)&&(sensor4 == 0))
                      {
                              //en1 = 1;
                              //en2 = 1;
                              motor1_f = 1;
                              motor1_b = 0;
                              motor2_f = 0;
                              motor2_b = 0;
                      }
        else if((sensor2 == 0)&&(sensor4 == 1))
                      {
                              //en1 = 1;
                              //en2 = 1;
                              motor1_f = 0;
                              motor1_b = 0;
                              motor2_f = 1;
                              motor2_b = 0;
                      }
}
////////////////////////////////////////////////////////
///                   line 1  &&  line 2
////////////////////////////////////////////////////////
void line1(void)
        {
                      if(sensor10 == 1)
                              {
                                      i = 1;
                                      check1 = 0;
                                      delay(4);
                              }
                      while(i == 1)
                        {
                              track();
                              if(sensor5 == 1)
                                      {
                                              j = 1;
                                              check3 = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    stop();
                    delay(2);
                    i--;
                }
    }
if(j == 1)
        {
                check4 = 0;
                forward();
                delay(2);
                turnr();
                k = 1;
                j--;
        }
while(k == 1)
        {
                track();
                if(sensor7 == 1)
                        {
                                check2 = 0;
                                stop();
                                l = 1;
                                k++;
                        }
        }
while(l == 1)
        {
                stop();
                if((sensor12 == 0)||(sensor13 == 0)||(sensor14 == 0))
                        {
                                l--;
                                b = 1;
                                delay(4);
                        }
        }
while(b == 1)
        {
                track();
                if(sensor5 == 1)
                        {
                                forward();
                                delay(2);
                                turnr();
                        }
                if(sensor6 == 1)
                        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        stop();
                                        b--;
                                    }
        }

void line2(void)
        {
            if(sensor11 == 1)
                {
                    i = 1;
                    check1 = 0;
                    delay(4);
                }
            while(i == 1)
              {
                track();
                if(sensor5 == 1)
                    {
                        j = 1;
                        check3 = 0;
                        stop();
                        delay(2);
                        i--;
                    }
              }
            if(j == 1)
                {
                    check4 = 0;
                    forward();
                    delay(2);
                    turnr();
                    k = 1;
                    j--;
                }
             while(k == 1)
                {
                    track();
                    if(sensor8 == 1)
                        {
                            check2 = 0;
                            stop();
                            l = 1;
                            k++;
                        }
                }
             while(l == 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                        stop();
                        if((sensor12 == 0)||(sensor13 == 0)||(sensor14 == 0))
                                {
                                        l--;
                                        b = 1;
                                        delay(4);
                                }
                }
        while(b == 1)
                {
                        track();
                        if(sensor5 == 1)
                                {
                                        forward();
                                        delay(2);
                                        turnr();
                                }
                        if(sensor6 == 1)
                                {
                                        stop();
                                        b--;
                                }
        }

//////////////////////////////////////////////////////////
///                     test check
//////////////////////////////////////////////////////////
void check(void)
        {
                check1 = 0;
                check2 = 0;
                check3 = 0;
                check4 = 0;
                check5 = 0;
                check6 = 0;
                check7 = 0;
                check8 = 0;
        }

//////////////////////////////////////////////////////////
///                     main
//////////////////////////////////////////////////////////

void main(void)
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
en1 = 1;
en2 = 1;
i = 0;
j = 0;
k = 0;
l = 0;
b = 0;
a = 0;
c = 0;

if((sensor10 == 1)&&((sensor12 == 1)||(sensor13 == 1)||(sensor14 == 1)))
        {
                c = 1;
        }
while(c == 1)
        {
                line1();
        }
if((sensor11 == 1)&&((sensor12 == 1)||(sensor13 == 1)||(sensor14 == 1)))
        {
                a = 1;
        }
while(a == 1)
        {
                line2();
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค. โปรแกรม visual basis 6.0

FORM 1

```
Private Sub Command1_Click()
MSComm1.Output = Chr(&H41)
counter(1) = counter(1) + 1
Label6.Caption = " connector " & vbCrLf & Label6.Caption
counteremer(1) = counteremer(1) + 1
If counteremer(1) >= 3 Then
Label9.Caption = "connector"
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command2_Click()
MSComm1.Output = Chr(&H42)
counter(2) = counter(2) + 1
Label6.Caption = " cot       " & vbCrLf & Label6.Caption
counteremer(2) = counteremer(2) + 1
If counteremer(2) >= 3 Then
Label9.Caption = "cot"
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command3_Click()
MSComm1.Output = Chr(&H43)
counter(3) = counter(3) + 1
Label6.Caption = " clip      " & vbCrLf & Label6.Caption
counteremer(3) = counteremer(3) + 1
If counteremer(3) >= 3 Then
Label9.Caption = "cot"
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command4_Click()
MSComm1.Output = Chr(&H44)
counter(4) = counter(4) + 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Label6.Caption = "    " & vbCrLf & Label6.Caption
counteremer(4) = counteremer(4) + 1
If counteremer(4) >= 3 Then
Label9.Caption = "fuse box"
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command5_Click()
MSComm1.Output = Chr(&H45)
counter(5) = counter(5) + 1
Label6.Caption = " grommet    " & vbCrLf & Label6.Caption
counteremer(5) = counteremer(5) + 1
If counteremer(5) >= 3 Then
Label9.Caption = "grommet"
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command6_Click()
MSComm1.Output = Chr(&H46)
counter(6) = counter(6) + 1
Label6.Caption = " mold      " & vbCrLf & Label6.Caption
counteremer(6) = counteremer(6) + 1
If counteremer(6) >= 3 Then
Label9.Caption = "mold "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command7_Click()
MSComm1.Output = Chr(&H47)
counter(7) = counter(7) + 1
Label6.Caption = " plug      " & vbCrLf & Label6.Caption
counteremer(7) = counteremer(7) + 1
If counteremer(7) >= 3 Then
Label9.Caption = "plug  "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command8_Click()
MSComm1.Output = Chr(&H48)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
counter(8) = counter(8) + 1
Label6.Caption = " printing   " & vbCrLf & Label6.Caption
counteremer(8) = counteremer(8) + 1
If counteremer(8) >= 3 Then
Label9.Caption = "printing  "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command9_Click()
MSComm1.Output = Chr(&H49)
counter(9) = counter(9) + 1
Label6.Caption = " protector  " & vbCrLf & Label6.Caption
counteremer(9) = counteremer(9) + 1
If counteremer(9) >= 3 Then
Label9.Caption = "protector  "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command10_Click()
MSComm1.Output = Chr(&H4A)
counter(10) = counter(10) + 1
Label6.Caption = " solder    " & vbCrLf & Label6.Caption
counteremer(10) = counteremer(10) + 1
If counteremer(10) >= 3 Then
Label9.Caption = "solder  "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command11_Click()
MSComm1.Output = Chr(&H4B)
counter(11) = counter(11) + 1
Label6.Caption = " solf tape  " & vbCrLf & Label6.Caption
counteremer(11) = counteremer(11) + 1
If counteremer(11) >= 3 Then
Label9.Caption = "solf tape   "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command12_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MSComm1.Output = Chr(&H4C)
counter(12) = counter(12) + 1
Label6.Caption = " tape      " & vbCrLf & Label6.Caption
counteremer(12) = counteremer(12) + 1
If counteremer(12) >= 3 Then
Label9.Caption = "tape    "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command13_Click()
MSComm1.Output = Chr(&H4D)
counter(13) = counter(13) + 1
Label6.Caption = " tube -vo   " & vbCrLf & Label6.Caption
counteremer(13) = counteremer(13) + 1
If counteremer(13) >= 3 Then
Label9.Caption = "tube -vo "
Shape1.BackColor = vbRed
End If
End Sub

Private Sub Command14_Click()
MSComm1.Output = Chr(&H4E)
counter(14) = counter(14) + 1
Label6.Caption = " tube vo-b  " & vbCrLf & Label6.Caption
counteremer(14) = counteremer(14) + 1
If counteremer(14) >= 3 Then
Label9.Caption = "tube -vo "
Shape1 = vbRed
End If
End Sub

Private Sub Command15_Click()
MSComm1.Output = Chr(&H4F)
counter(15) = counter(15) + 1
Label6.Caption = " tube vo    " & vbCrLf & Label6.Caption
counteremer(15) = counteremer(15) + 1
If counteremer(15) >= 3 Then
Label9.Caption = "tube vo "
Shape1.BackColor = vbRed
End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command16_Click()
MSComm1.Output = Chr(&H50)
counter(16) = counter(16) + 1
Label6.Caption = " vitly sheet " & vbCrLf & Label6.Caption
counteremer(16) = counteremer(16) + 1
If counteremer(16) >= 3 Then
Label9.Caption = "vitly sheet "
Shape1.BackColor = vbRed
End If
End Sub
```

FORM 2

```
Private Sub Command1_Click()
Call Form1.Command22_Click
Unload Me
End Sub
```

FORM 3

```
Private Sub Command1_Click()
Form1.MSComm1.Output = Chr(&H54)
Unload Me
End Sub

Private Sub Command2_Click()
Unload Me
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] ฉัททวุฒิ พืชผลมพิชิต ,คู่มือเรียน Visual Basic 6 ,บริษัทโปรวิชัน จำกัด,2550

[2] ศุภชัย บุศราทิจ, Micro-c51 , เมษายน 2543

[3] ธีรบูลย์ หล่อวิเชียรรุ่ง,นคร ภักดีชาติ ชัยยวัฒน์ ลิ้มพรจิตรวิไล ปฏิบัติการไมโครคอนโทรเลอร์ MCS-51

[4] นิรุธ อำนวยศิลป์. คู่มือการเขียนโปรแกรม Microsoft Visual C++ Version 6.0 ฉบับเพื่อการใช้งานจริง. กรุงเทพนคร : ซัคเซส มีเดีย.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้