บทที่ 3
การออกแบบหุ่นยนต์
(Robot Designing)

การออกแบบหรือการวางแผนงานก่อนการลงมือปฏิบัติจริง ถือเป็นสิ่งสำคัญในกระบวนการผลิต เพราะถ้าไม่มีการออกแบบหรือการวางแผนงานที่ดีแล้ว จะทำให้เกิดปัญหาต่าง ๆ ตามมามากมาย ดังนั้นในการสร้างหุ่นยนต์นี้ จึงจำเป็นที่จะต้องออกแบบหุ่นยนต์และมีวิธีการทดลองที่เป็นขั้นตอนที่เหมาะสม โดยทำการวิเคราะห์ข้อมูลในทุก ๆ ด้าน ไม่ว่าจะเป็นทางด้านกลศาสตร์ ทางด้านวัสดุศาสตร์และทางด้านไฟฟ้า เพื่อให้หุ่นยนต์สามารถทำงานได้อย่างมีประสิทธิภาพ ไม่เกิดปัญหาในการสร้าง การประกอบหุ่นยนต์และการนำหุ่นยนต์ไปใช้งานจริง

สำหรับเนื้อหาของการออกแบบหุ่นยนต์นี้ สามารถแบ่งออกเป็นหัวข้อต่าง ๆ ได้ดังนี้

3.1 ลักษณะ การเคลื่อนที่และของการทำงานของหุ่นยนต์โดยรวม

3.2 การออกแบบทางด้านวัสดุศาสตร์

3.3 การออกแบบทางด้านกลศาสตร์

- 3.3.1 การออกแบบขั้นพื้นฐาน
  - (1) แบริ่งที่ใช้ในการจับยึดเพลาที่ใช้ในการขับเคลื่อน
  - (2) ระบบบังคับเลี้ยว
  - (3) ระบบขับเคลื่อน
- 3.3.2 การออกแบบเพลา
- 3.3.3 การออกแบบโครงสร้าง

3.4 การออกแบบทางด้านไฟฟ้า

- 3.4.1 การออกแบบการควบคุมมอเตอร์
- 3.4.2 การออกแบบการควบคุมหุ่นยนต์โดยใช้ PLC
- 3.4.3 การออกแบบการเดินสายไฟในหุ่นยนต์
- 3.4.4 การออกแบบเอ็นโคดเดอร์
- 3.4.5 การออกแบบกลไกในการแทงลูกโป่ง

3.5 การออกแบบโปรแกรมควบคุมการทำงานของหุ่นยนต์

3.5.1 ความหมายของรหัสต่าง ๆ ในโปรแกรม

3.5.2 โปรแกรมควบคุมการทำงาน

3.1 <u>ลักษณะ การเคลื่อนที่และของการทำงานของหุ่นยนต์โดยรวม</u>

โดยมีขั้นตอนการทำงานเคลื่อนที่แบ่งเป็น 10 ขั้นตอนดังนี้

ขั้นตอนที่ 1 : หลังจาก กดปุ่ม Main Switch และStart Switch เรียบร้อยแล้ว หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วเคลื่อนที่ไปยังตำแหน่งลูกโป่งลูกที่ 1 เพื่อทำการตรวจสอบสีลูกโป่งเป้าหมาย โดยที่มอเตอร์ขับเคลื่อนทั้งสองด้านหมุน Forward ไป 35 pulses (1 m)โดยมี Condition การทำงานเป็น 2 Condition ดังนี้

- ถ้าเป็นลูกโป่งสีขาวจะหยุดแทงก่อน โดยการให้มอเตอร์ชุดสายพานแทงลูกโป่ง Forward ไปแทง และReverseโดยใช้ Limit Switch เป็นตัวตัด
- แต่ถ้าเป็นลูกโป่งสีเขียวจะทำการข้ามไปยัง Step 2 โดยไม่หยุดแทงลูกโป่ง

ขั้นตอนที่ 2 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วมอเตอร์ขับเคลื่อนทั้งสองด้านหมุน Forward ไป 35 pulses (1 m) ไปหยุดตรงบริเวณเลี้ยวที่ 1

ขั้นตอนที่ 3 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วมอเตอร์ขับเคลื่อนด้านขวาหมุน Forward ส่วนมอเตอร์ด้านซ้ายหมุน Reverse เพื่อทำการเลี้ยวซ้าย ไป 9 pulses (90 องศา)

ขั้นตอนที่ 4 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วเคลื่อนที่ไปยังตำแหน่งเลี้ยวที่ 2 โดยที่มอเตอร์ขับเคลื่อนทั้งสองด้านหมุน Forward ไป pulses 70 (2 m)

ขั้นตอนที่ 5 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วมอเตอร์ขับเคลื่อนด้านขวาหมุน Forward ส่วนมอเตอร์ด้านซ้ายหมุน Reverse เพื่อทำการเลี้ยวซ้าย ไป 9 pulses (90 องศา)

- ถ้าเป็นลูกโป่งสีขาวจะหยุดแทงก่อน โดยการให้มอเตอร์ชุดสายพานแทงลูกโป่ง Forward ไปแทง และReverseโดยใช้ Limit Switch เป็นตัวตัด

-แต่ถ้าเป็นลูกโป่งสีเขียวจะทำการข้ามไปยัง Step 7 โดยไม่หยุดแทงลูกโป่ง

ขั้นตอนที่ 7 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วมอเตอร์ขับเคลื่อนทั้งสองด้านหมุน Forward ไป 35 pulses (1m) ไปหยุดตรงบริเวณเลี้ยวที่ 3

ขั้นตอนที่ 8 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วมอเตอร์ขับเคลื่อนด้านขวาหมุน Forward ส่วนมอเตอร์ด้านซ้ายหมุน Reverse เพื่อทำการเลี้ยวซ้าย ไป 9 pulses (90 องศา)

ขั้นตอนที่ 9 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วเคลื่อนที่ไปยังตำแหน่งเลี้ยวที่ 4 โดยที่มอเตอร์ขับเคลื่อนทั้งสองด้านหมุน Forward ไป pulses 70 (2 m)

ขั้นตอนที่ 10 : หุ่นยนต์จะหน่วงเวลาไว้ 1 วินาที แล้วมอเตอร์ขับเคลื่อนด้านขวาหมุน Forward ส่วนมอเตอร์ด้านซ้ายหมุน Reverse เพื่อทำการเลี้ยวขวา (90 องศา) โดยใช้ Timer เป็นตัว Reset

ซึ่งจากการทำงานทั้ง 10 ขั้นตอนจะพบว่า มีการทำงานเป็น 2 ลักษณะการทำงาน ดังนี้

(1) การเคลื่อนที่ไปตามเส้น แบ่งเป็น 2 ลักษณะ

(1.1) การเดินตรง

(1.2) การเลี้ยว -โดยการใช้ Encoder อย่างง่าย เป็นตัว Reset (เลี้ยวซ้าย)

-โดยการตั้งเวลาการทำงาน (เลี้ยวขวา)

(2) การตรวจสีลูกโป่งเป้าหมาย และการทำลายลูกโป่งเป้าหมาย

(2.1) ตรวจพบลูกโป่งสีเขียว ไม่ทำลายลูกโป่ง แต่ข้ามไปยัง Step การทำงานขั้นต่อไป

(2.2) ตรวจพบลูกโป่งสีขาว แล้วมอเตอร์ทำลายลูกโป่งทำงาน พร้อมกับกลับมาอยู่ในสภาวะเริ่มต้นอีกครั้ง(พร้อมทำลายใหม่อีกครั้ง)

3.2 <u>การออกแบบทางด้านวัสดุศาสตร์</u>

ในการออกแบบหุ่นยนต์เดินอัตโนมัติในครั้งนี้ เราจะให้ความสำคัญต่อน้ำหนักตัวหุ่นโดยจะทำให้อยู่ในระหว่าง 10 - 15 Kg ซึ่งไม่ได้ออกแบบให้นำมาใช้งานในการรับน้ำหนัก ดังนั้นวัสดุที่นำมาใช้เป็นส่วนประกอบ ได้ทำการออกแบบดังนี้

1.โครงสร้าง : เราจะใช้ "อะลูมิเนียม" เป็นโครงสร้างของตัวหุ่น เนื่องจากว่าอะลูมิเนียมจัดเป็นโลหะที่มีน้ำหนักเบา มีความทนทานต่อการเป็นสนิม มีความแข็งแรงอยู่ในเกณฑ์ปานกลาง แต่มีความเหนียวสูง สามารถนำไปใช้งานแทนเหล็กแต่มีข้อจำกัดคือ มีราคาแพง

2.อุปกรณ์ในการจับยึด : ใช้สลักเกลียวเหล็กขนาด M3x0.5 ยาว 25 มม. เนื่องจากมีขนาดที่สามาหมุนจับยึดได้ง่าย เกลียวและหัวหมุดไม่เสียหายง่าย สามารถใช้ทำงานได้หลายจุด

3.ล้อ : ล้อให้ใช้การทดลองเป็นล้อยาง เนื่องจากมีแรงเสียดทานกับพื้นทดสอบดี มีความแข็งแรง สามารถรับน้ำหนักเพียงพอ สามารถนำไปตัดกลึงได้ง่าย หาซื้อง่ายมีราคาถูก

4.มอเตอร์ : เป็นมอเตอร์เก่า 24 VDC มีเฟืองทด สามารถให้แรงบิดได้สูงในความเร็วรอบต่ำ ที่สามารถใช้งานได้ความเร็วรอบขณะไม่มีภาระประมาณ 90 รอบ/นาที (12 VDC) ในขณะที่มีภาระ จะมีความเร็วรอบประมาณ 40 รอบ/นาที (12 VDC)

5.เพลา : เพลาที่ใช้การส่งถ่ายกำลังไปยังล้อ ใช้สแตนเลส เนื่องจากไม่เกิดสนิมสามารถทำเกลียวในการจับยึดได้ดี ไม่ทำให้เกลียวเสียหาย มีความแข็งแรงสูง

3.3 การออกแบบทางด้านกลศาสตร์

3.3.1 การออกแบบขั้นพื้นฐาน

ในการออกแบบหุ่นยนต์เดินอัตโนมัติในครั้งนี้ เราจะให้ความสนใจเกี่ยวกับระบบขับเคลื่อนและสิ่งที่เกี่ยวข้องดังต่อไปนี้ คือ

(1) แบริ่งที่ใช้ในการจับยึดเพลาที่ใช้ในการขับเคลื่อน (Bearing support position)

จุดประสงค์เพื่อป้องกันไม่ให้เกิดโมเมนต์ดัดกับเพลา เหมือนการงัดกับตะปู และในการออกแบบไม่ควรใช้ Casing house bearing เนื่องจากมีน้ำหนักมากซึ่งเราจะใช้ Thrust Bearing ในการจับยึดล้อกับตัวโครงรถ

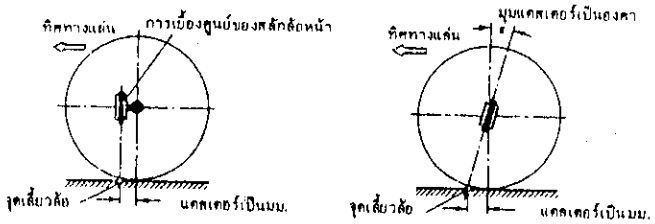
(2) ระบบบังคับเลี้ยว (Steering System)

ในการเคลื่อนที่ของหุ่นยนต์ต้องมีความเสถียรภาพในการควบคุม ไม่ว่าจะเป็นการเคลื่อนที่ในทางตรงหรือการเลี้ยวซึ่งในการที่มอเตอร์ทำการส่งกำลังในการบังคับเลี้ยว ล้อต้องหมุนกลับล้อตั้งตรงได้เองซึ่งในการทำงานลักษณะนี้ต้องได้รับการควบคุมจากหลายๆ ปัจจัยเช่นการออกแบบระบบกันสะเทือน การยืดหยุ่นของล้อ และที่สำคัญคือมุมล้อหน้า

การจัดตั้งมุมล้อหน้า (Wheel alignment angularity) มีผลต่อการควบคุมการเลี้ยวได้ง่าย ทรงตัวได้ดี และส่งผลต่อมอเตอร์ที่ทำการส่งกำลังให้ทำการส่งกำลังน้อยที่สุด ซึ่งมุมต่างๆ ของ ล้อหน้า ที่ทำการพิจารณาได้แก่

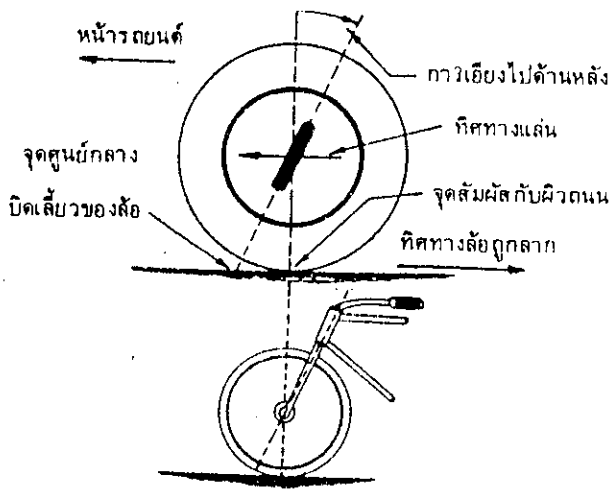
- ออกแบบโดยใช้มุมแคสเตอร์ (Caster angle)

มุมแคสเตอร์ (Caster angle) คือมุมที่เกิดจากการที่สลักล้อหน้าเอียงไปจากแนวดิ่งเมื่อมองจากด้านข้างของหุ่น หรือ เป็นมุมระหว่าง เส้นผ่าศูนย์กลาง ของล้อในแนวดิ่งตัดกันกับเส้นผ่าศูนย์กลางของแกนหันเลี้ยวล้อเป็นมุมของการเอียงของแกนหันเลี้ยวล้อออกจากแนวดิ่ง ซึ่งต่างจากมุมเอียงของแกนหันล้อเลี้ยว ดังแสดงในรูปที่ 3.1



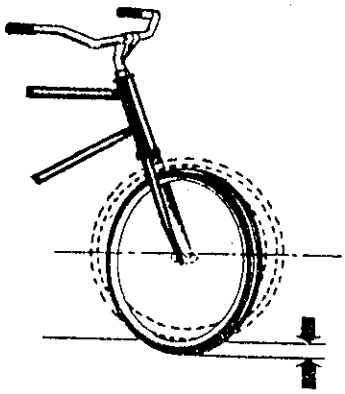
รูปที่ 3.1 แสดงมุมแคสเตอร์ที่เกิดจากการเยื้องศูนย์ของสลักล้อหน้า
และการเอียงของสลักล้อหน้า สาเหตุที่ต้องมีมุมแคสเตอร์

1.เพื่อให้หุ่นมีเสถียรภาพในการวิ่งทางตรง กล่าวคือมีทิศทางคงตัว มุมแคสเตอร์บวกมีแนวโน้มให้หุ่นสามารถเคลื่อนที่ไปข้างหน้า เพราะจุดตัดของเส้นศูนย์กลางเลี้ยวล้อกับพื้นผิวของถนนนำหน้า จุดสัมผัสของยางบนพื้นถนน



รูปที่ 3.2 แสดง การเปรียบเทียบการเอียงของแกนหันเลี้ยวของรถยนต์
กับการเอียงของล้อหน้า รถจักรยาน

2.ทำให้ล้อตีกลับตั้งตรงภายหลังจากการทำการเลี้ยว ถ้ามุมแคสเตอร์เป็นบวก จะเห็นได้จากในรถจักรยาน ขณะทำการเลี้ยวจะเห็นได้ว่า ตะเกียบล้อหน้าถูกยกสูงขึ้นเป็นการยกตัวรถดังนั้นถ้าปล่อยมือออกจากแฮนด์รถล้อจะตีกลับคืนสู่ทิศทางเดิม เพราะแรงโน้มถ่วงของโลก หรือน้ำหนักตัวรถ ที่กดให้รถต่ำลง

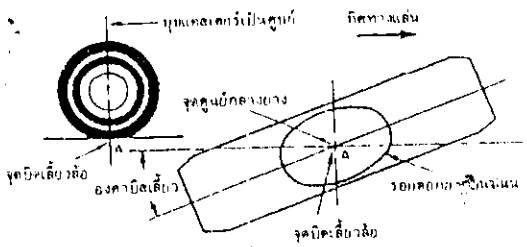


รูปที่ 3.3 แสดงการกดต่ำลงของล้อหน้ารถจักรยานขณะหันเลี้ยว

3.ชดเชยสภาพการเอียงของถนน ที่รถแล่นบนถนนผิวเอียงจะทำให้รถดึงไปด้านต่ำ การตั้งมุมแคสเตอร์จะช่วยแก้อาการดึงได้ โดยการตั้งให้มุมแคสเตอร์ของล้อหน้าด้านซ้ายให้มากกว่าในด้านขวา ประมาณ1-2องศา แต่ในการทำRobot ในครั้งนี้เราจะไม่ทำการพิจารณาหัวข้อนี้ร่วมด้วยเนื่องพื้นที่ที่ใช้ ในการทดลองไม่มีการลาดเอียง

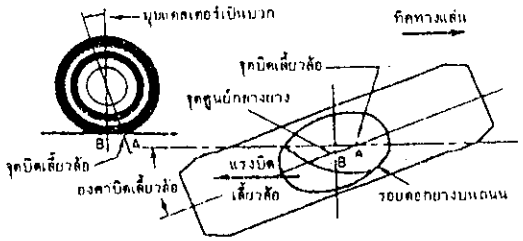
ผลของมุมแคสเตอร์เมื่อทำการเลี้ยว

1.มุมแคสเตอร์ศูนย์ (Zero Caster) เมื่อสลักล้อหน้าตั้งอยู่ในแนวดิ่ง หรือตั้งฉากกบพื้น จุดนำในการเลี้ยวจะทับกับจุดตกของน้ำหนัก ลักษณะนี้จะไม่ส่งผลถึงตัวรถเลยในด้านแนวโน้มที่จะทำการเลี้ยว ดังนั้นล้อไม่มีเสถียรภาพในการกำหนดทิศทาง ดังนั้นแรงที่ใช้ในการบังคับเลี้ยวจะน้อยมาก



รูปที่ 3.4 แสดง ขณะที่แคสเตอร์เป็นศูนย์จุดหันเลี้ยวจะทับกับจุดศูนย์กลางยาง

2.มุมแคสเตอร์บวก (Positive Caster) เป็นมุมที่ปลายแกนหันเลี้ยวเอียงไปด้านหน้า และจุดศูนย์กลางของรอยดอกยางบนพื้นจะอยู่หลังจุดหันเลี้ยวล้อ (จุดลาก) แต่ไม่ได้อยู่ในทิศทางการเคลื่อนที่ และเคลื่อนตามหลังจุดหันเลี้ยวล้อทำให้รักษาสภาพการเคลื่อนที่ได้ให้ตรงตลอดเวลา แต่ขณะหันเลี้ยวต้องอาศัยแรงที่ใช้ในการเลี้ยวมาก แต่พอผ่านการเลี้ยวไปแล้วจะสามารถคืนตัวกลับมาได้ง่าย ไม่ทำให้รถส่ายไป-มา

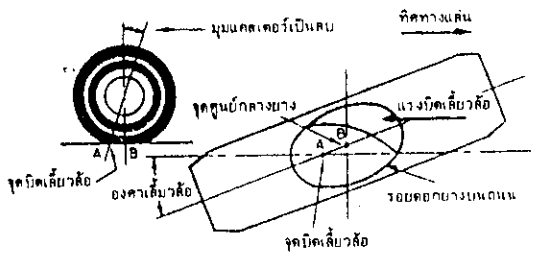


รูปที่ 3.5 แสดงขณะเคลื่อนที่ไปข้างหน้าเมื่อมุมแคสเตอร์มีค่าเป็นบวก
จะทำให้จุดหันเลี้ยวของล้ออยู่ข้างหน้าจุดศูนย์กลางยาง

3.มุมแคสเตอร์เป็นลบ (Negative Caster) ลักษณะเป็นมุมที่ปลายแกนหันเลี้ยวเอียงไปด้านหน้า และศูนย์กลางรอยดอกยางบนพื้นจะอยู่หน้าจุดหันเลี้ยวล้อ จุดศูนย์กลางรอยดอกยางจะไม่อยู่ในแนวเส้นทิศทางการเคลื่อนที่ ทำให้ล้อมีแนวโน้มจะเฉออกนอกทิศทางตลอดเวลา ทั้งนี้เพราะจุดศูนย์กลาง การหันเลี้ยวจะอยู่หลังจุดศูนย์กลางรอยดอกยาง ทำให้จุดศูนย์กลางรอยดอกยางพยายามที่จะเคลื่อนที่ไปข้างหลังจุดศูนย์กลางการหันเลี้ยว แคสเตอร์ลบจะติดตั้งกับรถที่มีเสถียรภาพการเคลื่อนที่ทางตรงมาก นั้นคือล้อจะตีกลับทางตรงเร็วเกินไป

ดังนั้นรถที่ขับเคลื่อนล้อหน้าจึงมักจะใช้มุมแคสเตอร์แบบนี้ เพราะว่าจุดที่ยางสัมผัสกับถนนเป็นจุดที่ล้อหมุนตะกุยให้เกิดการเคลื่อนที่ หรือทำหน้าที่เป็นจุดลาก และจุดหันเลี้ยวล้อเป็นจุด

ตาม ทำให้ล้อเกาะพื้นน้อยลงในตอนเลี้ยวโค้ง ตัวรถด้านนอกโค้งสูงขึ้น เป็นการต้านไม่ให้รถโคลงตัวขณะเลี้ยว



รูปที่ 3.6 แสดง ขณะที่เคลื่อนที่ไป เมื่อมุมแคสเตอร์เป็นลบ
จุดหันเลี้ยวงล้อจะอยู่หลังจุดศูนย์กลางยาง

ผลกระทบจากการใช้มุมแคสเตอร์ไม่ถูกต้อง

1. ถ้ามุมแคสเตอร์ของล้อทั้ง 2 ล้อไม่เท่ากันจะทำให้เคลื่อนที่กินไปยังด้านที่มีมุมแคสเตอร์บวกน้อยกว่า
2. มีค่าเป็นบวกมากเกินไป จะทำให้ต้องใช้แรงในการบังคับเลี้ยวสูง ล้อเต้นมากขณะวิ่ง
3. มีค่าเป็นลบมากเกินไป จะทำให้ทิศทางไม่คงตัว ทำให้เคลื่อนที่ส่ายไป-มา

(3) ระบบขับเคลื่อน

ระบบขับเคลื่อนที่ใช้ในหุ่นยนต์จะมีด้วยกัน 3 แบบดังนี้คือ

1. **Hydraulic Drive Systems** : ซึ่งเป็นการใช้พลังงานจากของเหลวเป็นตัวขับเคลื่อน เช่น น้ำมัน หรือของเหลวอื่นๆ มักจะใช้หุ่นยนต์ประเภทแขนกล

2. **Pneumatic Drive Systems** : ซึ่งเป็นการใช้พลังงานจากลมเป็นตัวขับเคลื่อน โดยมักจะใช้กับหุ่นยนต์ประเภทแขนกล หรือประเภทที่ไม่ต้องการการเคลื่อนที่ของรากฐานไป

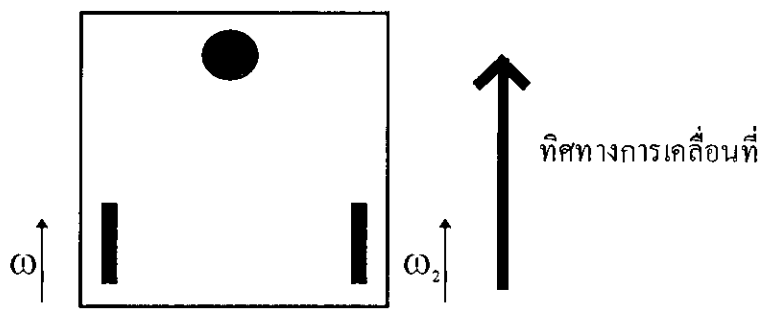
3. **Electric Drive Systems** : เป็นการใช้พลังงานไฟฟ้าเป็นตัวส่งกำลัง มักใช้กับแขนกลและใช้มากกับหุ่นยนต์ที่มีการเคลื่อนที่ได้ ซึ่งระบบนี้ก็แบ่งย่อยออกเป็น

- DC Motor
- AC Motor

- Servo System

ซึ่งระบบที่เราทำการศึกษาคือระบบ การใช้ **"DC Motor"** ในการควบคุมโดยใช้มอเตอร์เป็นตัวส่งกำลัง และใช้ล้อในการเคลื่อนที่ ซึ่งในการควบคุมการทำงานโดยเรามีหลักการทำงานดังนี้คือ

(1) การเดินตรง และการถอยหลัง



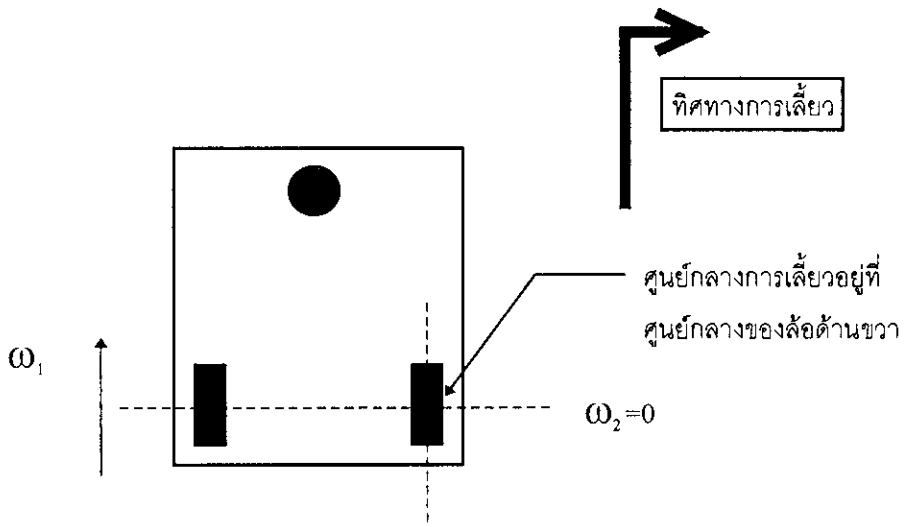
รูปที่ 3.7 แสดงทิศทางการเคลื่อนที่

ซึ่งมีหลักการทำงานคือมอเตอร์ 2 ตัวหมุนไปในทางเดียวกันโดยมีความเร็วเท่ากัน ซึ่งในรูปเป็นการเคลื่อนที่ไปข้างหน้า ซึ่งในทำนองเดียวกันการเคลื่อนที่ถอยหลังก็อาศัยหลักการเดียวกัน นี้ด้วยเพียงแต่กลับทิศทางเท่านั้น ($\omega_1 = \omega_2$)

(2) การบังคับเลี้ยว

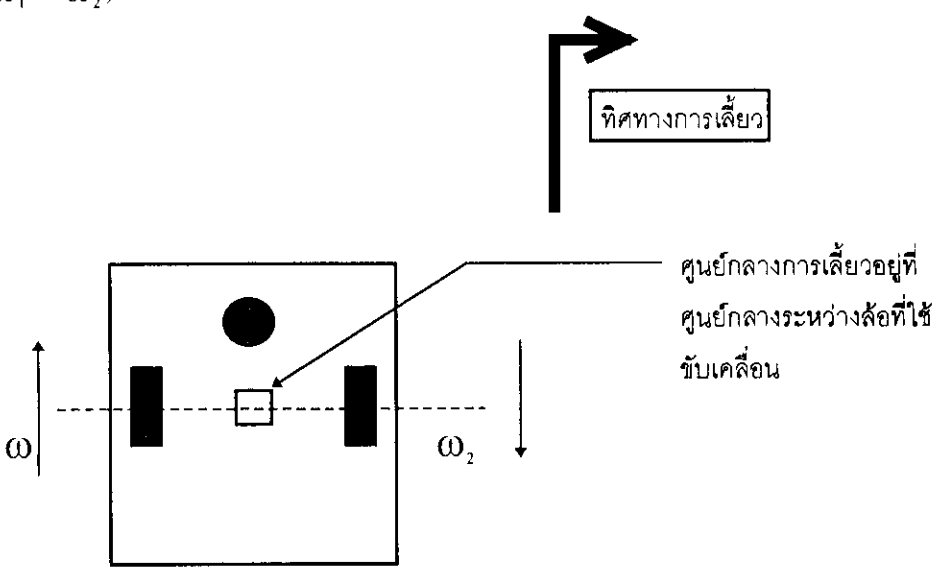
มีหลักการ คือ การหมุนของมอเตอร์ 2 ตัวทำงานแตกต่างกัน มีด้วยกัน 2 แบบด้วยกัน คือ

2.1 หมุนไปในทิศเดียวกันแต่ความเร็วแตกต่างกัน ซึ่งในกรณีนี้เราทำการพิจารณาการเลี้ยวขวาโดยเราทำการกำหนด $\omega_1 > \omega_2$ ซึ่งค่า ω_2 อาจจะเท่ากับ 0 ซึ่งจะง่ายต่อการพิจารณา



รูปที่ 3.8 แสดงการเลี้ยวขวาโดยการสั่งล้อขวาหยุดและหมุนล้อซ้ายใน

2.2 การหมุนของมอเตอร์ไปในทิศตรงกันข้ามกัน แต่คงระดับความเร็วเท่ากัน เป็นวิธีการบังคับเลี้ยวที่น่าจะเสถียรมากที่สุด และต้องการใช้พื้นที่ในการใช้ในการเลี้ยวโค้งน้อยอีกด้วยเนื่องจากรัศมีการเลี้ยวจะอยู่ที่ตรงจุดกึ่งกลางระหว่างล้อที่ใช้ในการขับเคลื่อน ($\omega_1 = \omega_2$)



รูปที่ 3.9 แสดงการเลี้ยวขวาโดยการหมุนล้อทั้งสองล้อ

แต่วิธีการเลี้ยวแบบนี้ต้องอาศัยการควบคุมมอเตอร์ ที่ต้องแน่นอนมากยิ่งขึ้นเหมาะสำหรับการเลี้ยวในที่ที่มีพื้นที่จำกัด และใช้เวลาในการทำการเลี้ยวน้อยแบบแรก แต่ก็มีข้อจำกัดที่การควบคุมมอเตอร์ ต้องมีความระมัดระวังสูงกว่า

3.3.2 <u>การออกแบบเพลา</u>

การออกแบบเพลาโดยพิจารณาถึงความแข็งแรง สามารถแบ่งเป็น 3 วิธีคือ

1. การออกแบบโดยวิธีสถิตย์

1.1 การออกแบบตามทฤษฎีความเค้นเฉือนสูงสุด

1.2 การออกแบบตามทฤษฎีพลังงานของการเปลี่ยนรูป

1.3 การออกแบบตามทฤษฎีความเค้นสูงสุด

1.4 การออกแบบตามมาตรฐาน ASME CODE

2 การออกแบบตามวิธีการล้า

2.1 การออกแบบตามทฤษฎีกู๊ดแมน

2.2 การออกแบบตามทฤษฎีโซเดอร์เบอร์ก

2.3 ทฤษฎีการล้าสะสมเชิงเส้นของ Minor Palmgren

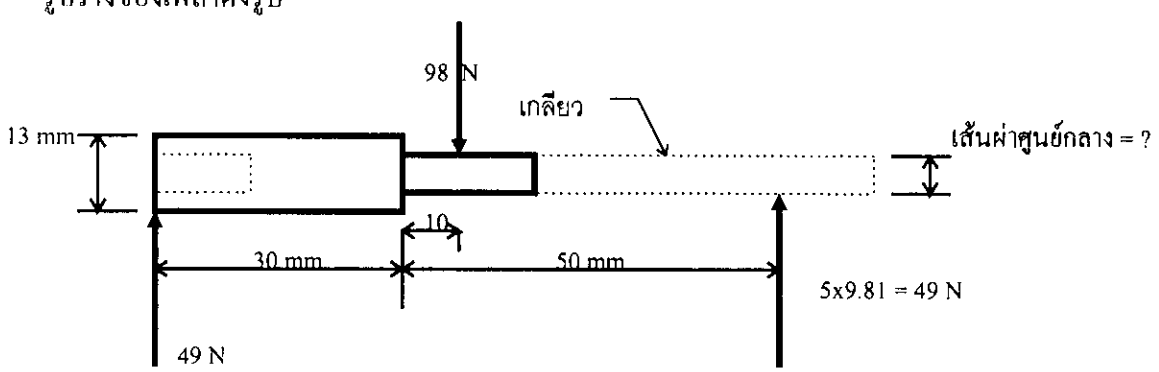
3. การออกแบบตามวิธีกลศาสตร์ของการแตกหัก

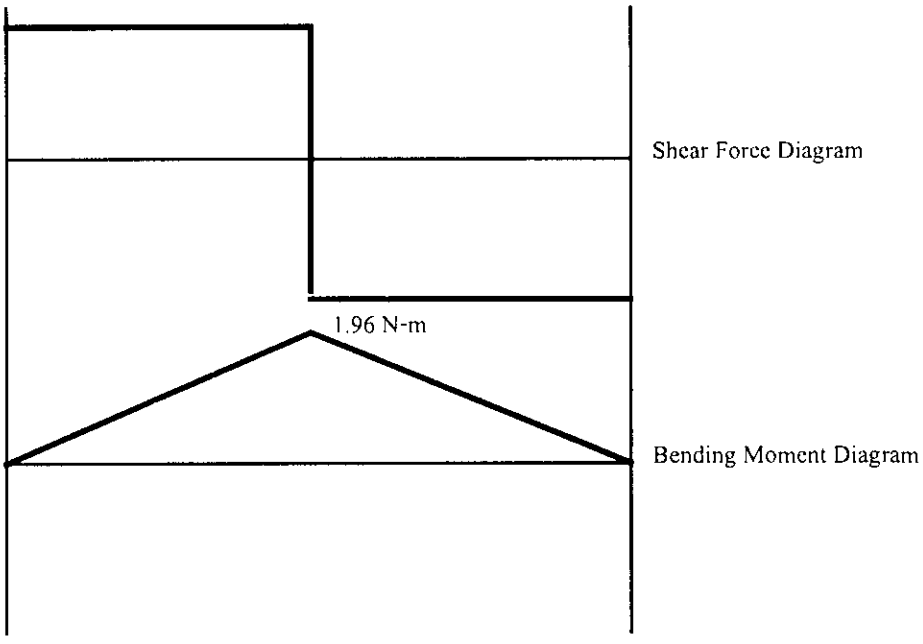
เพลาที่ใช้ในการขับล้อให้หุ่นยนต์ทำงาน ต้องมีความแข็งแรงสามารถรับแรงเฉือนโมเมนต์คัดได้ดี ไม่เกิดสนิมที่อาจเกิดการทำงานที่บกพร่อง สามารถรับแรงได้แม้เป็นชิ้นงานที่ ไม่หนามากนัก จึงเลือกใช้สแตนเลส (Stainless steel 403) ซึ่งสามารถจัดหาได้ง่าย

คุณสมบัติทางกลของวัสดุที่ใช้ในการทำเพลา

σ_{yt} = 35 ksi

รูปร่างของเพลาดังรูป





รูปที่ 3.10 แสดงการวิเคราะห์แรงในเพลา

น้ำหนักของรถสูงสุด 10 Kg.

แรงบิดที่เพลาได้รับ T = 0.51 N-m

โมเมนต์ดัดสูงสุด M = 1.96 N-m

หาขนาดของเส้นผ่าศูนย์กลางของเพลาโดยใช้มาตรฐาน ASME CODE

$$D^3 = \frac{16}{\pi \tau_d}[(K_m M)^2 + (K_s T)^2]^{0.5}$$

$$\tau_d = \frac{0.5\sigma_{yt}}{2N} = \frac{0.5 \times 37000}{2 \times 2} = 4625 \text{ psi}$$

N คือ ค่าความปลอดภัย = 2

K_m	=	1.5	ภาระกระตุกเล็กน้อย
K_s	=	1	ภาระกระตุกเล็กน้อย
M	=	17.3	in-lb
T	=	4.3	in-lb

D = 0.307 in

= 7.8 mm.

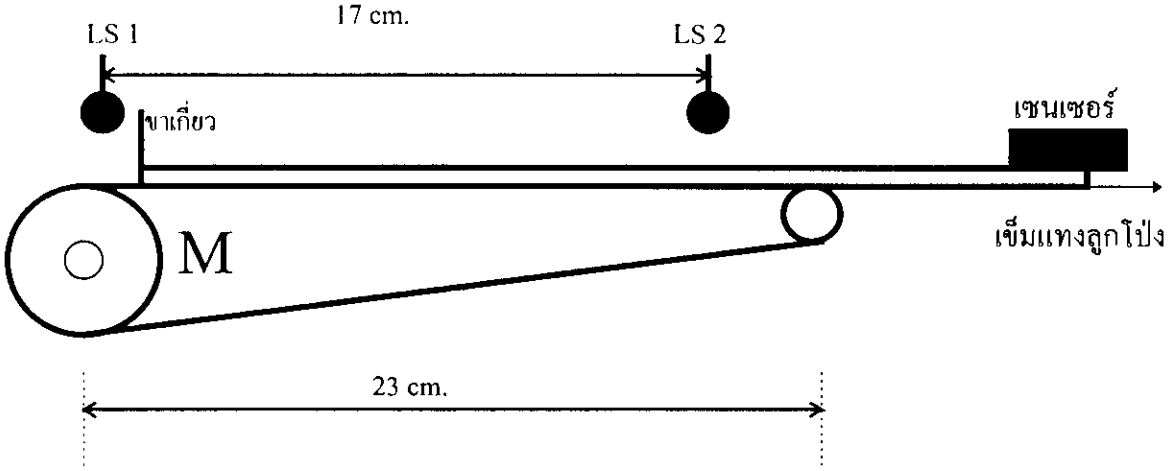
เนื่องจากไม่สามารถจัดหาตลับลูกปืนขนาด 7.8 มม. ได้ แต่มีตลับลูกปืนขนาด 8 มม. จึงเลือกใช้เพลาที่มีขนาด 8 มม.

3.3.3 การออกแบบโครงสร้าง

โครงสร้างหลักเป็นส่วนสำคัญ เพื่อรับน้ำหนักของหุ่นยนต์ทั้งหมด วัสดุที่ใช้ทำโครงสร้าง คือ อะลูมิเนียม เนื่องจากสามารถตัดกลึงได้ง่าย หาซื้อง่าย มีความเค้นดึงสูงสุด 83 MPa ความเค้นยืดหยุ่น 31 MPa(No. 1100-0 ความแข็งแรงต่ำสุดของอะลูมิเนียม ตารางที่ ก10 การออกแบบเครื่องจักรกล 1 อาจารย์จำลอง ลิ้มตระกูล) น้ำหนักเบาเพียง 1 ใน 3 ของเหล็ก การเลือกโดยอาศัยคุณสมบัติของอะลูมิเนียม ที่สามารถดัดแปลงเป็นชิ้นส่วนอื่น ๆ ได้ขนาดของอะลูมิเนียมที่ใช้คือ 25x25x2 mm.

3.4.5 การออกแบบกลไกในการแทงลูกโป่ง

กลไกการแทงลูกโป่งเป็นกลไกที่มีความสำคัญ เพื่อทำให้ลูกโป่งสีขาวแตก โดยปลายของกลไกแทงลุกโป่งจะติดด้วยเข็มแหลมที่สามารถให้ทำให้ลูกโป่งแตกอย่างง่ายดาย พร้อมกันนี้สามารถติดปลายเซ็นเซอร์ เพื่อให้เซ็นเซอร์สามารถตรวจสอบลูกโป่งได้ดี



รูปที่ 3.11 แสดงกลไกในการแทงลูกโป่ง

การทำงานของกลไกในการแทงลูกโป่งมีขั้นตอนดังนี้

1. เซ็นเซอร์ตรวจสอบสีของลูกโป่งเป็นสีขาว
2. PLC สั่งมอเตอร์หมุนตามเข็มนาฬิกาทำให้เข็มลื่อนออกไปแทงลูกโป่ง จนกระทั่งขาเกี่ยวถึงลิมิตสวิทย์ตัวที่ 2
3. PLC สั่งมอเตอร์หยุดหมุน หน่วงเวลา 1 วินาที่
4. PLC สั่งให้มอเตอร์ทวนเข็มนาฬิกา เข็มเลื่อนเข้าจนขาเกี่ยวถึงลิมิตตัวที่ 1 มอเตอร์หยุดหมุนและจบการทำงาน

3.4 การออกแบบทางด้านไฟฟ้า

3.4.1 การออกแบบการควบคุมมอเตอร์ (Motor Control System)

ชนิดของมอเตอร์ (Types Of Motor)

มอเตอร์ที่มักจะใช้ในส่วนการขับเคลื่อน และApplications อื่นๆ Mobile Robot เป็นมอเตอร์ไฟฟ้ากระแสตรง(DC Motor) มีดังต่อไปนี้คือ

1.1 Servo Motor

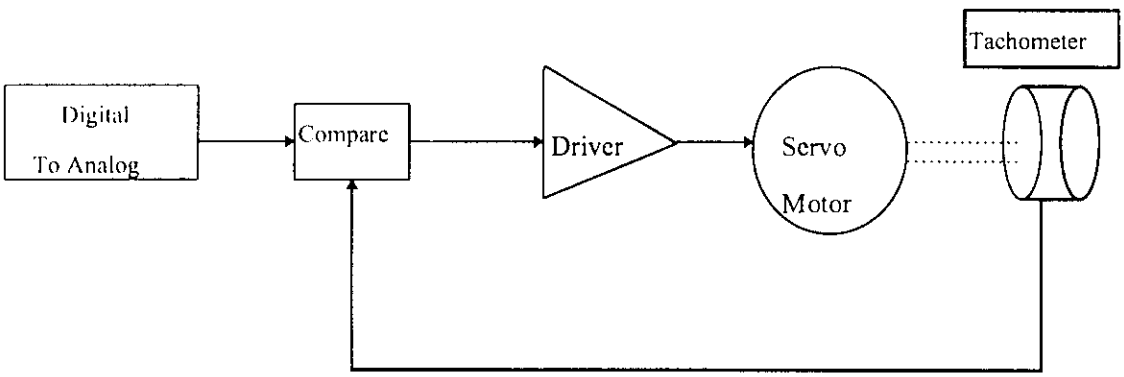
1.2 Stepping Motor

1.3 DC motor

1.1 Servo Motor

เป็นมอเตอร์ที่มีลักษณะการทำงานคล้าย กับ DC Motor ธรรมดาทั่วไป แต่มีการป้อนสัญญาณเพื่อนำไปเปรียบเทียบเพื่อให้ได้ตำแหน่งที่ต้องการ Servo Motor ที่นิยมใช้ได้แก่

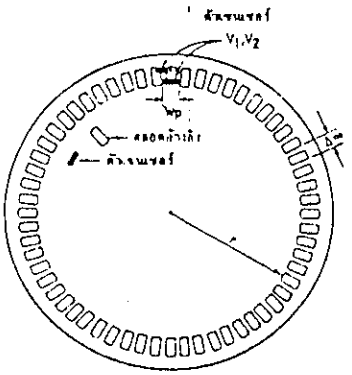
1. แบบใช้ Tachometers ซึ่งลักษณะของ Tachometers มีลักษณะคล้ายกับเครื่องกำเนิดกระแสไฟฟ้า ซึ่งต่อเข้าโดยตรงกับแกนของมอเตอร์ มีหน่วยเป็น มิลลิโวลต์ / รอบซึ่งมีแผนผังการทำงานดังนี้



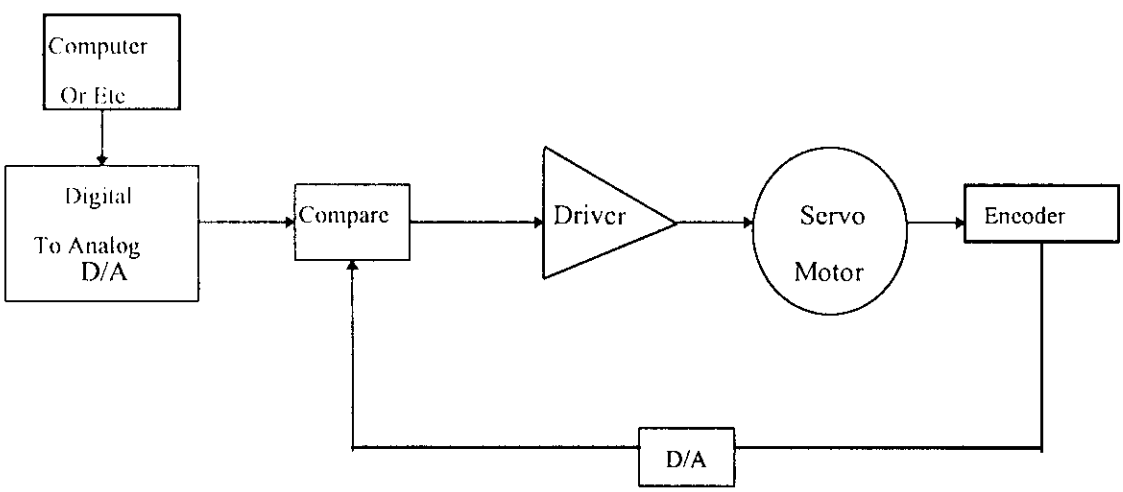
รูปที่ 3.12 แผนผังการทำงานของ Servo Motor และ Tachometer

ซึ่งผลจาก Output ของ Tachometer จะป้อนกลับเข้าวงจรเปรียบเทียบ เพื่อทำการขับมอเตอร์อีกทีหนึ่ง

2. แบบใช้ Encoder ลักษณะของ Encoder จะเป็นแผ่นจานและมีช่องสำหรับตัดผ่านตัว Sensor อีกทีหนึ่ง ตัวEncoder จะเชื่อมต่อโดยตรงกับแกนของมอเตอร์ ซึ่งจะใช้ Sensor ทำหน้าที่เช็คตำแหน่งของ Motor เพื่อป้อนกลับเข้าวงจรเปรียบเทียบ เช่นเดียวกับแบบแรกแต่ผลการป้อนกลับจะเป็นข้อมูลแบบดิจิตอล ซึ่งต้องมีการแปลงไปเป็น Analog อีกทีหนึ่ง



รูปที่ 3.13 แสดงลักษณะแผ่นจานของ Encoder

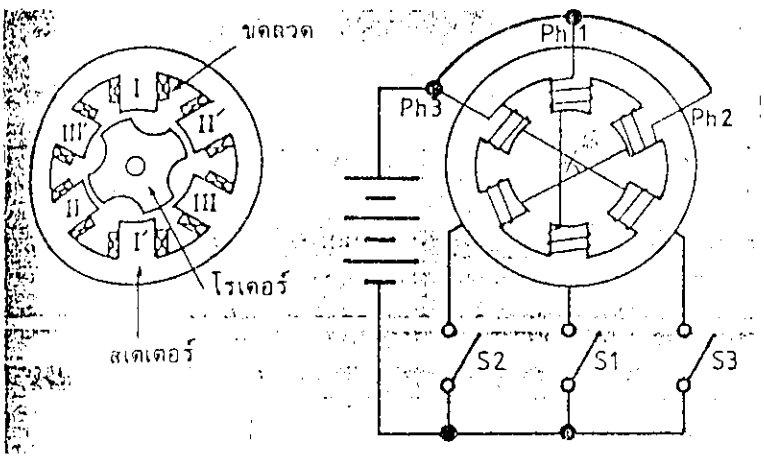


รูปที่ 3.14 แสดงแผนผังการทำงานของ Servo Motor แบบใช้Encoder

1.2 Stepping Motor

Stepping Motor เป็นมอเตอร์ที่ทำงานโดยการอาศัย Pulse ป้อนเข้าไปในขดลวดแต่ละขด เพื่อให้เกิดสนามแม่เหล็กในแต่ละขด โดยที่เราสามารถควบคุมได้ทั้งตำแหน่ง และความเร็วตามจำนวน Pulse ที่ป้อนเข้าสู่ขดลวดโดยไม่จำเป็นต้องอาศัยการปิ้นกลับเข้าช่วยแต่อย่างไร สามารถแบ่งได้ดังนี้คือ

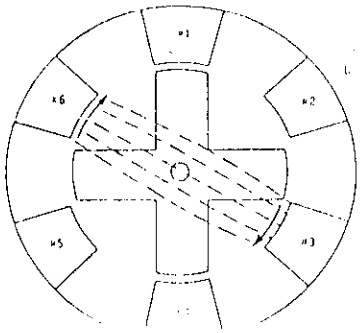
1. Variable Reluctance (V.R) ซึ่งเป็นพื้นฐานในการทำมอเตอร์ชนิดอื่นๆ มีลักษณะดังรูป 3.15



รูปที่ 3.15 พื้นฐานในการออกแบบ Stepping motor

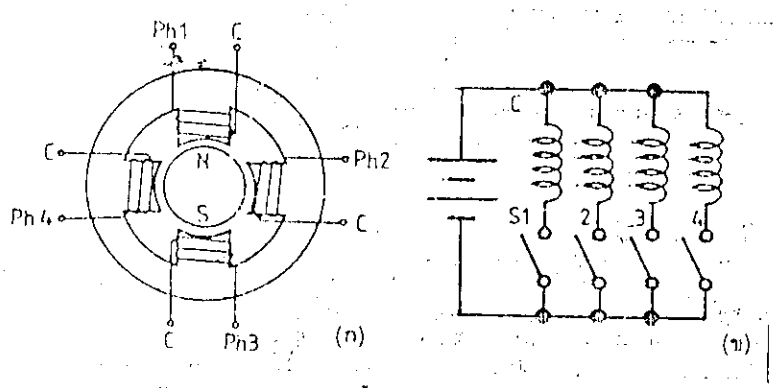
รูปข้างบนจะเป็นตัวอย่างของ Stepping Motor แบบ V.R. 3 Phase มีขั้วเหนือ และใต้อยู่ตรงข้ามกัน 3 คู่ โดยพันขดลวดอนุกรมในแต่ละคู่ ตัว Rotor และStator ทำจากเหล็กอ่อนผสม ซิลิกอน มีค่าความซึบซับของแม่เหล็กสูงทำให้เส้นแรงแม่เหล็กไหลผ่านมาก

สมมติการทำงานเริ่มจาก Phase Iทำให้เส้นแรงแม่เหล็กเป็นดังรูปคือ Rotor จะพยายามวางตำแหน่งตัวเองให้อยู่ในทิศทางที่ทำให้เกิดความต้านทานแม่เหล็กน้อยที่สุด และเมื่อเริ่มกระตุ้น Phase 2 เส้นแรงแม่เหล็กไม่ได้อยู่ในแนวทางเดินที่สะดวก ทำให้ค่าความต้านทานแม่เหล็กมีค่าสูงขึ้น Rotor ก็พยายามปรับตัวเอง โดยการหมุนไปในทิศทวนเข็มนาฬิกา แรงบิดที่ใช้หมุนเกิดจากแรงของเส้นแรงแม่เหล็ก แล้วไปหยุดที่ตำแหน่งความต้านทานน้อยที่สุด คือจะหมุนไป 1 Step คือ 30 องศา ลักษณะการเคลื่อนที่ของ Stator จะมีทิศทางตรงข้ามกับทิศทางการเคลื่อนที่ของ Rotor



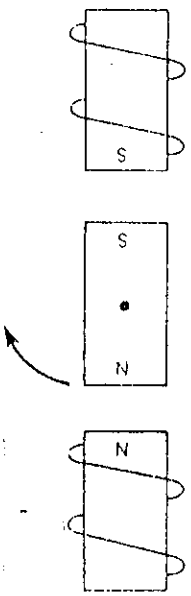
รูปที่ 3.16 แสดงลักษณะการเคลื่อนที่ของ Stator จะมีทิศทางตรงข้าม
กับทิศทางการเคลื่อนที่ของ Rotor

2. Stepping Motor ชนิด Permanent (P.M.) มีความแตกต่างกัน คือ Rotor จะเป็นตัวแม่เหล็กถาวร และจะต่างกันคือ ในStator แต่ละขั้ว จะมีขดลวดพันแยกจากกัน มีลักษณะดังรูปและแบ่งได้ดังนี้



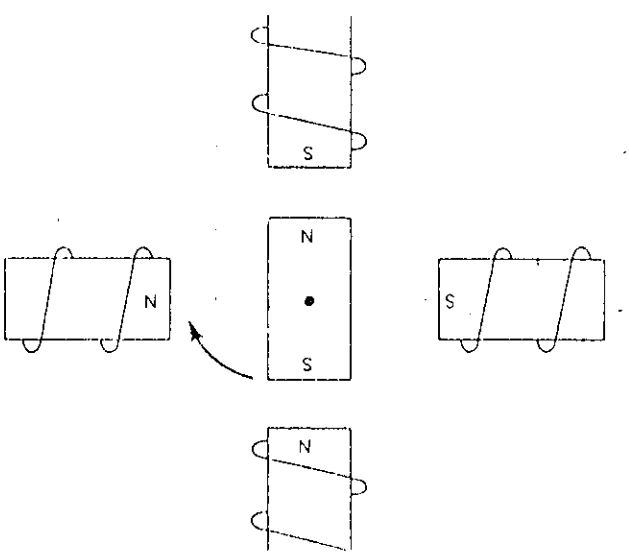
(ก) ภาพหน้าตัดของ PM สเต็ปปิ้งมอเตอร์แบบ 4 เฟส
(ข) วงจรกระตุ้นเฟสพื้นฐาน สำหรับ PM มอเตอร์ 4 เฟส

รูปที่ 3.17 แสดงหน้าตัดของ PM Stepping Motor และวงจรกระตุ้น Phase พื้นฐาน



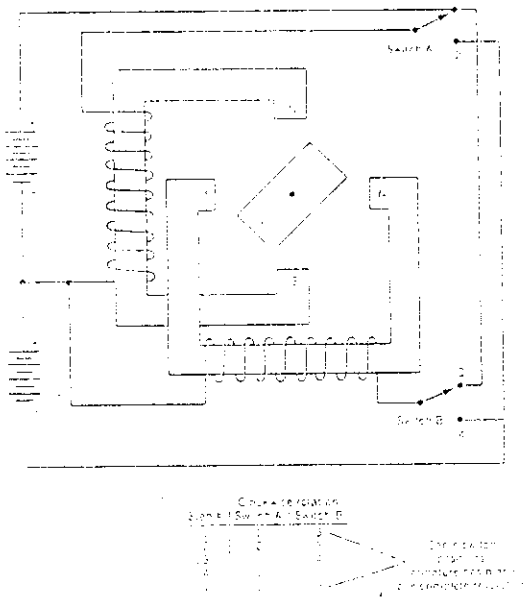
รูปที่ 3.18 แสดงลักษณะของ PM Stepping แบบง่าย

เมื่อสเตเตอร์ ทำงานคือมีกระแสไฟป้อนเข้า (ON) ขั้วแม่เหล็กของ Stator จะดูดกับขั้ว S ของ Rotor และขั้ว N ของ Rotor จะดูดกับขั้ว S ของ Stator ทำให้ Rotorหมุนตามทิศเข็มนาฬิกา และเนื่องจากความละเอียด และความแม่นยำยังไม่พอ จึงเพิ่ม Stator เข้าไปอีก 2 ขั้ว กลายเป็น 4 ขั้ว เมื่อสนามแม่เหล็กเป็นไปตามรูป Rotor จะวางตัวอยู่ในแนวระหว่างขั้ว ของ Stator นั้นคือขั้ว S ของ Rotor จะวางตัวระหว่างขั้ว N ของStator และขั้ว N ของRotor จะวางตัวอยู่ ระหว่างขั้ว S ของ Stator ดังแสดงในรูปที่ 3.19



รูปที่ 3.19 การวาง Rotor จะวางตัวอยู่ ระหว่างขั้ว S ของ Stator

เมื่อขั้วแม่เหล็กเปลี่ยนไปดังรูป 3.19 จะสังเกตเห็นว่า เมื่อสนามของแม่เหล็กบน Stator วิ่งวนไปในทิศทางใด ตัว Rotor ก็จะวิ่งไปในทิศทางนั้น ซึ่งจากรูปจะมีทิศทางการหมุนไปตามทิศตามเข็มนาฬิกา ในรูปที่ 3.20 เป็นการแสดง Switch Sequence Control ของมอเตอร์ในทิศตามเข็มนาฬิกา



รูปที่ 3.20 เป็นการแสดง Switch Sequence Control

- แบบ Unipolar มีลักษณะคล้ายคลึงกับแบบ Bipolar โดยมีความแตกต่างกันคือ Unipolar จะมี Tap กลางที่ขดลวด Stator และสามารถใช้แหล่งจ่ายไฟเดี่ยวได้ ซึ่งแบบ Bipolar จำเป็นจะต้องใช้แหล่งจ่ายไฟคู่ หรือต่อในลักษณะ Bridge แต่แบบ Unipolar จะมีความเข้มของสนามแม่เหล็กน้อยกว่า เนื่องจากการแบ่งขดลวดไปใช้ในการเกิดสนามแม่เหล็ก ทำให้มีค่า Torque น้อยกว่า Bipolar

การคำนวณมุมของ Stepping Motor

ในการคำนวณมุมที่เปลี่ยนไป 1 Step (θ)

$$S = 360 / \theta = m \times n$$

m = จำนวน Phase

n = จำนวนฟันของ Rotor

s = จำนวน Step ในการหมุน Rotor ไปใน 1 รอบ

ซึ่งจากการศึกษาเราสามารถสรุปข้อดี ข้อเสีย ของ Servo Motor และStepping Motor ได้ดังนี้

ตารางที่ 3.1 แสดงการเปรียบเทียบข้อเสีย ของ Servo Motor และStepping Motor

Servo Motor	**Stepping Motor**
1. จำเป็นที่จะต้องทำการแปลงสัญญาณดิจิตอลเป็นอนาลอกเสียก่อน ทั้งในส่วนป้อนกลับ และการขับมอเตอร์	1. สามารถควบคุมได้ โดยตรงจากสัญญาณดิจิตอล
2. ต้องอาศัยการป้อนกลับทำให้ยุ่งยากต่อการควบคุม และออกแบบวงจร (Close Loop)	2. ไม่จำเป็นต้องมีการป้อนกลับของสัญญาณสามารถที่จะ Control แบบ Open Loopได้โดยตรง
3. เกิดการสึกหรอเนื่องจากแปรงถ่าน	3. ไม่เกิดการสึกหรอ หรือเกิดน้อยมาก
4. มีราคาสูงทั้งตัวมอเตอร์ และวงจรควบคุม	4. สามารถเข้าถึงตำแหน่งได้อย่างรวดเร็ว
5. ต้องอาศัยการ เบรก ในการหยุดเนื่อง จากมี Torque สูง	5. ออกแบบวงจรควบคุมที่ง่ายกว่า
6. เหมาะสำหรับไว้ควบคุมในลักษณะ เส้นทางที่มีการเคลื่อนที่ต่อเนื่อง	6. Torque ทีเกิดมีขนาดน้อยกว่าเมื่อมีขนาด และน้ำหนักใกล้เคียงกัน
7. มีค่า Torque ที่สูงกว่า โดยเฉพาะเมื่อความเร็วสูงขึ้น	7. เหมาะสำหรับการเคลื่อนไม่ต่อเนื่อง

1.3 DC motor

ในการทดลองในครั้งนี้ได้ทำการศึกษาการใช้มอเตอร์หลายชนิด จึงตัดสินใจเลือกใช้มอเตอร์ไฟฟ้ากระแสตรง เนื่องจากเหตุผลดังนี้

1. มอเตอร์ไฟฟ้ากระแสตรงสามารถจัดซื้อได้ง่ายและมีราคาถูก

2. การใช้งานของมอเตอร์ไฟฟ้ากระแสตรง สามารถใช้งานได้ง่ายกว่า สามารถใช้งานแล้วให้ความเที่ยงตรงสูง การบำรุงรักษาน้อย

3. มอเตอร์ไฟฟ้ากระแสตรงสามารถใช้ไฟฟ้าได้หลายระดับแรงดัน เช่น มอเตอร์ที่ใช้ในการทดลองสามารถใช้งานได้ตั้งแต่ 6 - 30 V. จึงสามารถใช้งานได้หลายระดับแรงดันไฟฟ้า

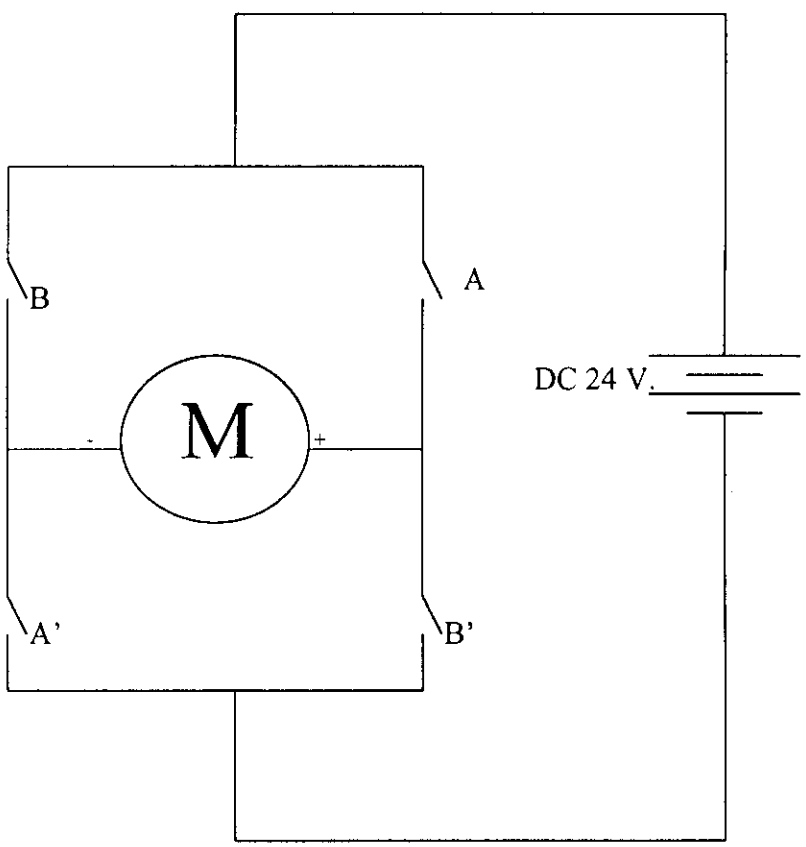
4. เนื่องจากสามารถใช้งานได้หลายระดับแรงดันไฟฟ้า ในทำนองเดียวกันจึงสามารถใช้งานได้หลายระดับความเร็ว

5. แหล่งพลังงานเป็นไฟฟ้ากระแสตรง ทำให้ไม่จำเป็นต้องมี Inverter ในการเปลี่ยนพลังงานเป็นไฟฟ้ากระแสตรงเป็นไฟฟ้ากระแสสลับ

6. มอเตอร์กระแสตรงมีขนาดเล็กกว่า เนื่องจากการใช้งานโดยทั่วไป มอเตอร์กระแสสลับจะมีขนาดใหญ่ ทำให้ไม่สามารถที่ประยุกต์เข้าทำงานในการทดลองนี้ได้

3. แรงบิดในการเริ่มต้นของมอเตอร์กระแสตรงมากกว่า

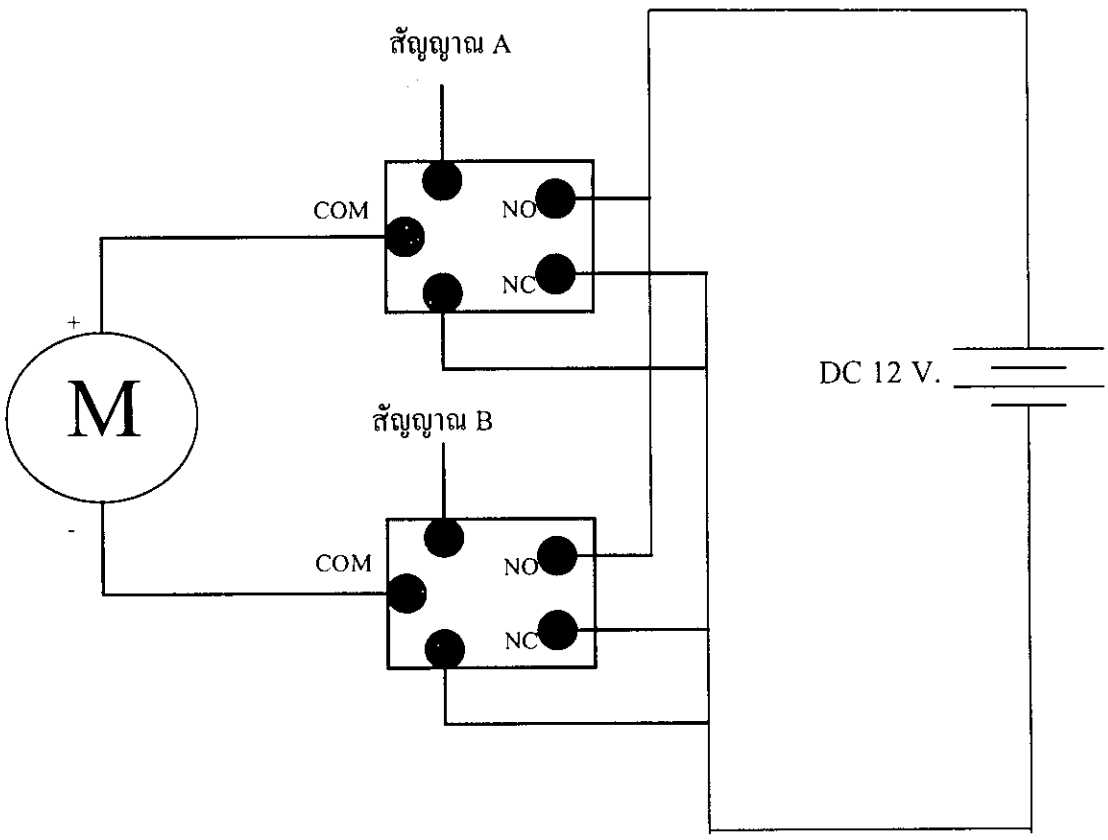
ด้วยเหตุผลที่กว่าในข้างต้น จึงเลือกใช้มอเตอร์ไฟฟ้ากระแสตรงใช้ในการทดลอง



รูปที่ 3.21 วงจรในการกลับทิศทางของมอเตอร์ไฟฟ้ากระแสตรง

ในขณะที่มอเดอร์หมุนตามเข็มนาฬิกา โดยการปิดสวิทซ์ A และ A' และเปิดสวิทซ์ B และ B' ขั้วบวกของแบตเตอรี่ต่อกับขั้วบวกของมอเตอร์ ทำให้มอเตอร์หมุนตามเข็มนาฬิกา ถ้าต้องการให้มอเตอร์หมุนทวนเข็มนาฬิกา โดยการเปิดสวิทซ์ A และ A' และปิดสวิทซ์ B และ B'

ในการทำงานจริงเลือกใช้ Rclay ในการปิด - เปิดวงจรเพื่อทำการกลับทิศทางของมอเตอร์



รูปที่ 3.22 การกลับทิศทางของมอเตอร์โดยใช้รีเลย์

การทำงานของวงจรรีเลย์มีความคล้ายคลึงกับ วงจรที่อยู่เริ่มบริดจ์ คือ สัญญาณ Aและสัญญาณ B จะไม่พร้อมกัน ถ้าสัญญาณ A ทำงานทำให้รีเลย์ทำงาน ทำให้ไฟฟ้าสามารถไหลเข้าไปในมอเตอร์ ผ่านมอเตอร์ไปเข้าขั้วลบที่รีเลย์อีกด้าน มอเตอร์จึงสามารถทำงานได้ในทิศทางตามเข็มนาฬิกา ถ้าสัญญาณเข้าที่ B มอเตอร์ก็จะทำงานในทิศทางทวนเข็มนาฬิกา

การควบคุมความเร็วของมอเตอร์ให้มีความเร็วตามที่ต้องการใช้งาน และสามารถเปลี่ยนค่าตามต้องการ ดังความสัมพันธ์ดังนี้

$$E = K\phi\omega$$

เมื่อ E = ความต่างศักดิ์ไฟฟ้า (Volt)

K = ค่าคงที่ของมอเตอร์

ϕ = เส้นแรงแม่เหล็ก(Waber)

ω = ความเร็วมอเตอร์(Rad/sec)

นอกจากการควบคุมความเร็วของมอเตอร์แล้ว มอเตอร์กระแสตรงยังสามารถควบคุมแรงบิดที่ใช้ในการทำงาน ดังความสัมพันธ์ดังนี้

$$T = K\phi I$$

เมื่อ T = แรงบิด(N-m)

I = กระแส(A.)

กำลังที่ใช้ในการขับเคลื่อน

ส่วนที่ 1 : ใช้ในการเปลี่ยนแปลงพลังงานจลน์ของการเคลื่อนที่ของมวลรถ m ในแบบต่างๆ ซึ่งมีพลังงานจลน์ (T)

T = T(ระบบการเลื่อนไหล) + T(ระบบหมุนรอบแกนคงที่)

$T = \frac{1}{2}(mv^2) + \frac{1}{2}(I_0\omega^2)$

โดยที่ v = ความเร็วรถ(m/s)

I_0 = โมเมนต์ความเฉื่อยรอบจุดศูนย์กลางมวล ($kg\text{-}m^2$)

ω = ความเร็วเชิงมุม(rad/s)

ซึ่งกำลังขับเคลื่อน $P = dT/dt$

$$= \quad mv_a + I_0\omega^2\alpha$$

$$= \quad (mv_a + I_0v^2/r)\, a$$

ส่วนที่ 2 : ใช้ในการเปลี่ยนแปลงพลังงานศักย์ เนื่องจากมีการเปลี่ยนแปลงของระดับในแนวดิ่ง

$$P = mgv\sin\theta$$

โดยที่ θ คือมุมลาดเอียง

ส่วนที่ 3 : การสูญเสียเนื่องจากแรงต้านการเคลื่อนที่ (R)

แรงต้านการเคลื่อนที่(R) = แรงต้านอากาศ (R_a) + แรงต้านการหมุนของล้อ (R_r)

- แรงต้านอากาศจะขึ้นอยู่กับอัตราความเร็วของรถ และพื้นที่หน้าตัด ซึ่งมีความสัมพันธ์ กันดังนี้

$$R_a = K_a A v^2$$

K_a = สัมประสิทธิ์แรงต้านอากาศ

v = แรงต้านของอากาศ (km/hr)

A = พื้นที่หน้าตัดของด้านรับลม (m^2)

ซึ่งค่า K_a ที่ใช้โดยทั่วไปมีดังนี้

K_a = 0.023 สำหรับรถแข่ง

K_a = 0.031 สำหรับรถทั่วไป

K_a = 0.045 สำหรับรถบรรทุก

- แรงต้านการหมุนของล้อ จะขึ้นอยู่กับ สภาพผิวของถนน สภาพยาง ขนาดของยาง น้ำหนัก และความดันลม ซึ่ง

$$R_r = K_r mg$$

K_r = สัมประสิทธิ์แรงต้านการหมุนของล้อ

ซึ่งค่า K_r ที่ใช้โดยทั่วไปมีดังนี้

$K_r = 0.0095$ สำหรับถนนที่เรียบมาก

$K_r = 0.18$ สำหรับถนนที่เป็นทราย

$K_r = 0.15$ สำหรับค่าเฉลี่ยที่ใช้กันโดยทั่วไป

เนื่องจากอัตราเร็วของรถจะมีผลต่อแรงต้านการหมุนของล้อ ดังนั้นค่า K_r ที่นิยมใช้อีกอย่างหนึ่งคือ $K_r = 0.015 + 0.0016V$

*** ดังนั้น กำลังขับเคลื่อนทั้ง 3 ส่วน

$$P = \{mv + (v/r^2)\}a + \{mg\sin\theta + R_a + R_r\}v$$

และ สำหรับแรงขับเคลื่อนที่ความเร็วใดๆ

$$F = P/v = \{m + (1/r^2)\}a + \{mg\sin\theta + R_a + R_r\}$$

แต่ในการออกแบบระบบให้มีความเร็วในการเคลื่อนที่คงที่ ดังนั้น $a = 0$

ดังนั้น $F = mg\sin\theta + R_a + R_r$ เมื่อ F คือแรงขับเคลื่อนที่ล้อ

และ แรงบิดที่แต่ละล้อมีค่า $T_w = F /$ จำนวนมอเตอร์

ในการออกแบบ กำหนดค่าต่างๆ ไว้ดังนี้

น้ำหนักรวมทั้งหมด $m = 10$ kg

$g = 9.81$ m/s

รัศมีของล้อ $r = 100$ mm (0.1 m)

ความเร็วสูงสุดที่ใช้ในการพิจารณา $v = 0.5$ m/s

ความเอียงของพื้นทดสอบสมมติ $\theta = 5^\circ$

พื้นที่หน้าตัดรับแรงลม คิดเพียง 75% (0.75 ของ 0.25x0.15 m)

พื้นทดสอบมีความลื่นมาก $K_r = 0.0095$

สำหรับกรณีทั่วไป $K_a = 0.031$

มอเตอร์ที่ใช้ในการขับเคลื่อน ใช้ 2 ตัว

พิจจารณา ค่า R_r = $K_r mg$

= $0.0095 \times 98.1 = 0.976$ N

ค่า R_a = $K_a Av^2$

= $0.031 \times 0.75(0.25 \times 0.15)(0.5 \times 60)^2 = 0.785$ N

$mg\sin\theta$ = $10 \times 9.81 \times \sin 5 = 8.55$ N

และจากที่ F = $mg\sin\theta + R_a + R_r$

ดังนั้น F = $8.55 + 0.785 + 0.976 = 10.311$N

ดังนั้น แรงขับเคลื่อนทั้งหมด มีค่า 10.311 N

และกำลังขับเคลื่อนที่ล้อ ของมอเตอร์แต่ละตัวมีค่า $T_w = FxR$ / จำนวนมอเตอร์

ดังนั้น T_w = $10.311 \times 0.1/2 = 0.516$ Nm

จาก ค่ากำลังไฟฟ้า = ค่ากำลังทางกล

ดังนั้น ค่ากำลังทางกล (P) = $\{mv + (v/r^2)\}a + \{mg\sin\theta + R_a + R_r\}v$

= Fv

= 10.311×0.5

= 5.156 watt

ค่ากำลังทางไฟฟ้า = IV ซึ่งใช้ค่าความต่างศักย์ในการทดสอบของมอเตอร์แต่ละตัว ใช้จาก แหล่งจ่าย 12V

ดังนั้น I = $(5.156/2) / 12$

= 0.215 A หรือ 215 mA

*** การออกแบบเลือกใช้มอเตอร์ ที่มี $T_w = 0.516$ Nm

และ กระแส 215 mA

3.4.2 การออกแบบการควบคุมหุ่นยนต์โดยใช้ PLC

(Programmable Logic Control)

การควบคุมการทำงานของหุ่นยนต์ จำเป็นต้องมีชุดประมวลผลกลาง (Central unit Processing) เพื่อใช้ในการประมวลผล มีหน่วยความจำสำหรับการเก็บชุดคำสั่งในการทำงาน อุปกรณ์ที่นิยมนำมาควบคุมการทำงานของหุ่นยนต์คือ PLC (Programmable Logic Control) และบอร์ดคอมพิวเตอร์เชิงเดี่ยว (Single Board Microprocessor) ในการศึกษาโครงงานนี้จะใช้ PLC ในการควบคุมเท่านั้น

หลักการทำงานของ PLC

PLC (Programmable Logic Control) เป็นอุปกรณ์ควบคุมการทำงานของเครื่องจักรหรือกระบวนการต่าง ๆ โดยภายในมี Microprocessor เป็นสมองสั่งการทำงาน สามารถต่ออินพุตและเอาท์พุตได้ทันที เราสามารถควบคุมได้ด้วยการป้อนโปรแกรมคำสั่งเข้าไปใน PLC โดยโปรแกรมนี้จะทำหน้าที่เหมือนวงจรรีเลย์ PLC จะสร้างอุปกรณ์ควบคุมต่าง ๆ ขึ้นมาเอง เช่น รีเลย์ ตัวตั้งเวลา ตัวนับ อุปกรณ์เหล่านี้จะอยู่ในรูปของซอฟแวร์ ไม่มีตัวตน แต่มีการทำงานตรงกับของจริง คุณสมบัติของPLC คือ

- ทำให้ขนาดของระบบเล็กลง ภายใน PLC จะใช้อุปกรณ์ทางอิเลคทรอนิคส์และซอฟแวร์แทน รีเลย์ ตัวนับ ตัวตั้งเวลา และองค์ประกอบอื่น ๆ อีกมากมาย ซึ่งจำนวนอุปกรณ์ต่าง ๆ เหล่านี้ไม่ขึ้นอยู่กับขนาดของ PLC

- ใช้โปรแกรมแทนการเดินสาย วงจรรีเลย์ต้องการการเดินสายระหว่างรีเลย์และอุปกรณ์ต่าง ๆ เพื่อประกอบวงจรควบคุม แต่ PLC ใช้โปรแกรมในรูปของวงจร ในหน่วยความจำจึงไม่มีการเดินสายระหว่างอุปกรณ์จริงให้ยุ่งยาก

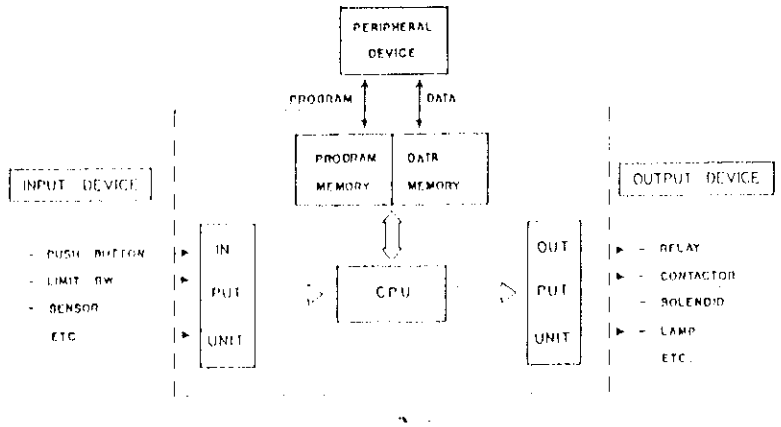
- เปลี่ยนแปลงและขยายวงจรง่าย โดยการแก้ไขผ่านทางโปรแกรม การทำงานก็จะแตกต่างไป การขยายระบบก็สามารถทำได้ง่าย โดยการต่อขยายอินพุตและเอาท์พุต

- ลดเวลาในการออกแบบและสร้าง อุปกรณ์ PLC มีมาตรฐานสามารถประกอบเข้าไปในตู้ได้อย่างรวดเร็ว การทำการออกแบบวงจรได้และการโปรแกรมทำได้โดยง่าย และสามารถทดสอบก่อนการใช้งานจริงอีกด้วย

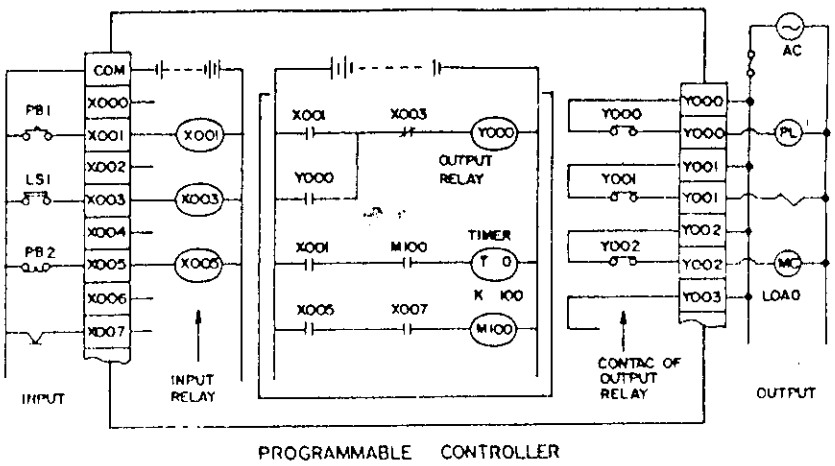
- PLCมีเสถียรภาพกว่ารีเลย์ ภายใน PLCเป็น Solid state วงจรไม่มีการเดินสายแบบรีเลย์จึงไม่มีปัญหาเรื่องสายขาด หน้าสัมผัสหลวม จึงสามารถทำงานโดยความเชื่อถือของระบบสูง

- มีหน่วยอินพุต-เอาท์พุตหลายแบบ มีทั้งแบบอนาลอก(Analog)และดิจิตอล(Digital)

- สามารถติดต่อกับอุปกรณ์ภายนอกได้ เช่น Computer , Printer



รูปที่ 3.23 แสดงโครงสร้างของ PLC

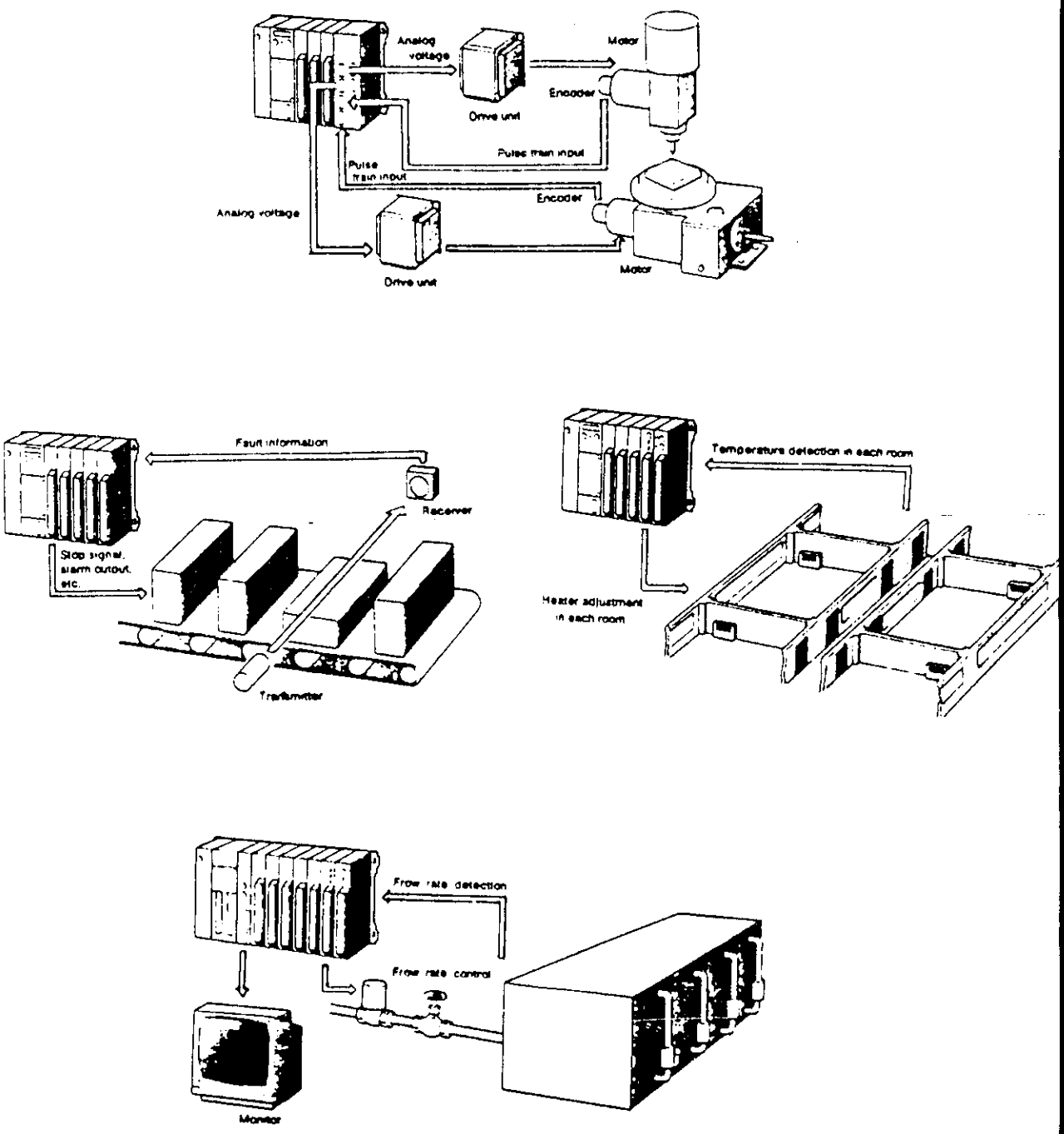


รูปที่ 3.24 การใช้โปรแกรมเดินสายภายใน
(โปรแกรมนี้เขียนขึ้นในคอมพิวเตอร์โดยใช้โปรแกรมโมนิเตอร์จากบริษัทผู้ผลิต PLC)

ตัวอย่างการประยุกต์ใช้งาน PLC

- โรงงานไฟฟ้า
- การฉีดพลาสติก
- อุตสาหกรรมเคมี
- การหล่อแบบต่อเนื่อง

- การบรรจุหีบห่อ
- การควบคุมเตาไฟฟ้า
- อุตสาหกรรมการประกอบรถยนต์
- การตรวจสอบผลิตภัณฑ์ เป็นต้น



รูปที่ 3.25 ตัวอย่างการใช้งานของ PLC ในอุตสาหกรรม

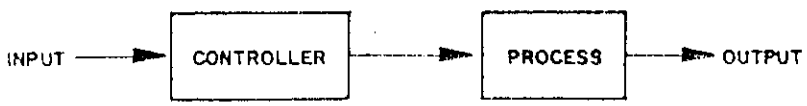
ประเภทของการควบคุม

การจำแนกประเภทของการควบคุมมีหลายกฎเกณฑ์ อาจแบ่งออกได้ตามลักษณะงาน แต่หลักในการการแบ่งที่นิยมคือ การใช้สัญญาณป้อนกลับเป็นเกณฑ์ โดยแบ่งได้เป็น 2 ประเภทคือ

1. แบบ Open Loop
2. แบบ Close Loop

1.แบบ Open Loop

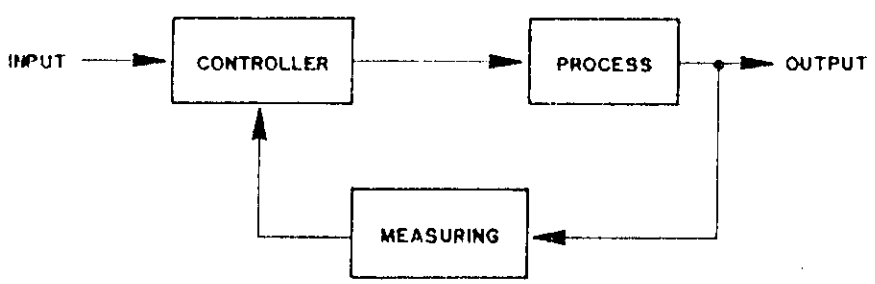
เป็นการควบคุมที่เอาท์พุตของระบบไม่มีผลต่อการควบคุมเลย นั่นคือ เอาท์พุตของระบบไม่มีการถูกวัดหรือป้อนกลับมาเปรียบเทียบกับอินพุต ตัวอย่างการควบคุมแบบนี้ได้แก่ การควบคุมการปิดเปิดสัญญาณไฟโฆษณา



รูปที่ 3.26 ระบบการควบคุมแบบ Open Loop

2.แบบ Close Loop

มีความตรงกันข้ามกับแบบOpen loop คือสัญญาณเอาท์พุตมีผลต่อการควบคุม คือมีระบบควบคุมแบบป้อนสัญญาณกลับ (Feedback Control) การควบคุมแบบนี้สามารถพบได้ง่าย เช่น การควบคุมกระบวนการของตู้เย็น



รูปที่ 3.27 ระบบการควบคุมแบบ Close Loop

ในการควบคุมหุ่นยนต์ในครั้งนี้ส่วนมากจะเป็นการควบคุมแบบ Close Loop ในบางจุดก็ทีการควบคุมแบบ Open loop เช่น กลไกในการเจาะลูกโป่งจะไม่มีการตรวจสอบว่าลูกโป่งที่ได้ทำการเจาะแตกหรือไม่

ประเภทเซนเซอร์ที่ใช้งาน

แบ่งออกเป็น 2 ประเภท ดังนี้

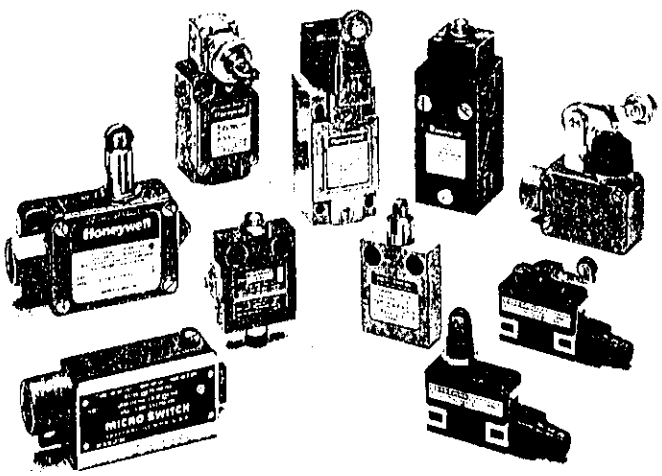
1.เซนเซอร์ภายใน (Internal sensors)

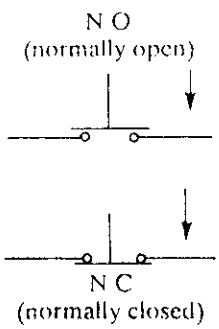
2.เซนเซอร์ภายนอก (External sensors)

1.เซนเซอร์ภายใน (Internal sensors)

เซนเซอร์ชนิดนี้ส่วนมากจะยึดติดบนแกนของมอเตอร์ ข้อต่อของแขน หรือในชิ้นส่วนที่ต้องการวัดระยะในการเปลี่ยนแปลงตำแหน่ง เซนเซอร์สามารถแบ่งได้หลายชนิด แต่ที่นิยมใช้มาก เช่น

1.1 ลิมิตสวิทซ์(The limit switch) เป็นเซนเซอร์พื้นฐาน มีหลักการทำงานที่เข้าใจง่าย คือมีลักษณะคล้ายกับสวิทซ์ไฟฟ้าธรรมดา มี 2 แบบคือ 1) แบบปกติเปิด(Normally open) ,2)แบบปกติปิด(Normally close) มีลักษณะดังรูป 3.28

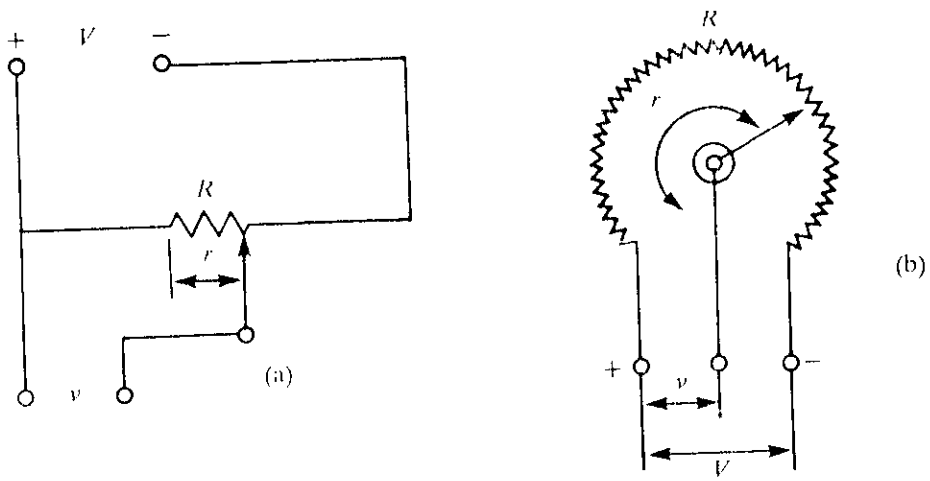




รูปที่ 3.28 แสดงลักษณะของ limit switch แบบต่าง ๆ

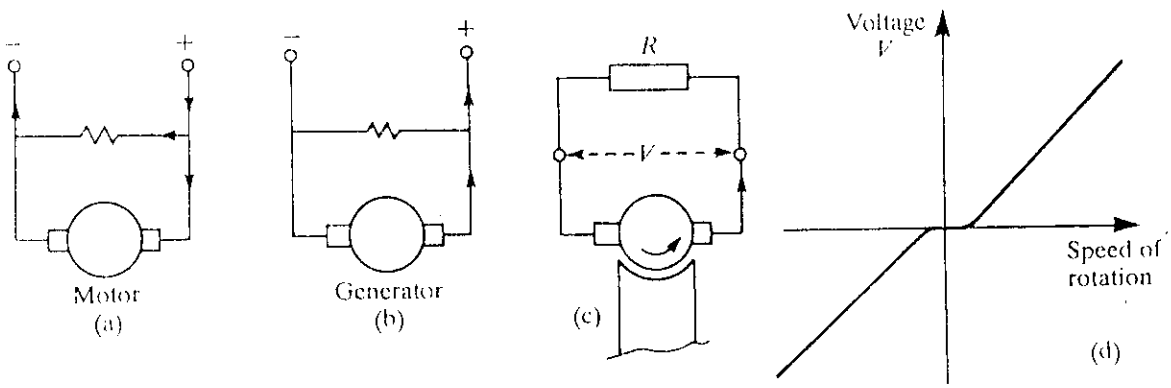
1.2 โปเทนทิโอมิเตอร์ (Potentiometer) เป็นเครื่องมือวัดสัญญาณไฟฟ้าป้อนกลับของการเปลี่ยนมุมของแขนกล โดยสัญญาณที่ป้อนกลับในรูปความต่างศักย์ไฟฟ้า (v) ดังรูปที่ 3.29

$$v = V*r/R$$



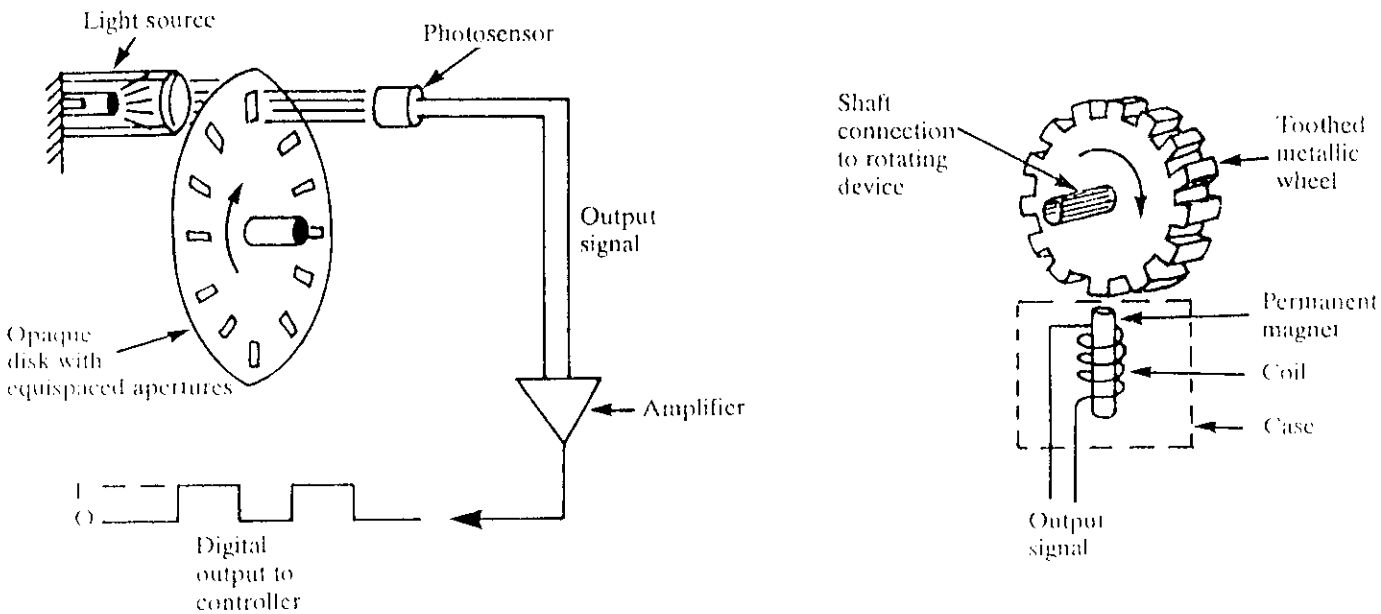
รูปที่ 3.29 แสดงลักษณะของโปเทนทิโอมิเตอร์

1.3 ทาโคเจนเนอเรเตอร์ (Tachogenerater) เป็นเครื่องมือวัดความเร็วที่ให้สัญญาณออกมาเป็นสัญญาณแบบอนาลอค ลักษณะการทำงานคล้ายกับโปเทนทิโอมิเตอร์ โดยจะให้ความต่างศักย์สูง เมื่อความเร็วสูงขึ้น ดังรูป 3.30



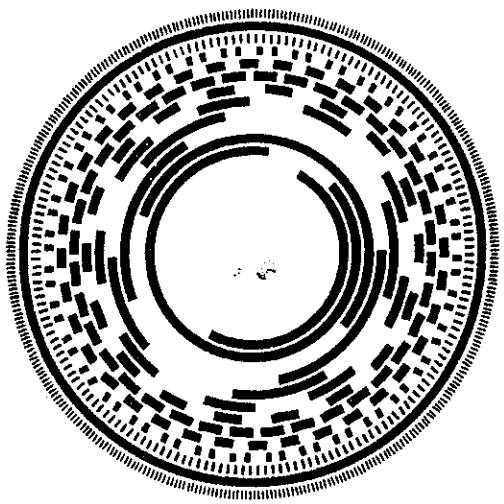
รูปที่ 3.30 หลักการทำงานของ ทาโคเจนเนอเรเตอร์

1.4 ดิจิตอลทาโคมิเตอร์ (Digital Tachometer) มี 2 แบบคือแบบใช้การเหนี่ยวนำ และการใช้แสง การทำงานโดยการนับสัญญาณสูง-ต่ำ (1-0) ที่ได้รับเทียบกับเวลา เพื่อใช้วัดความเร็วในการหมุนของเพลาต่าง ๆ ลักษณะของดิจิตอลทาโคมิเตอร์ ดังรูป 3.31

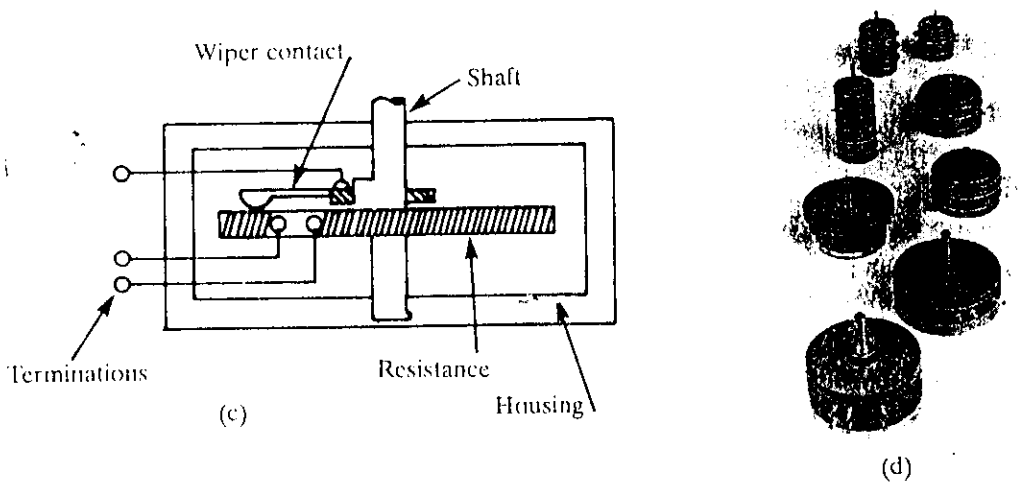


รูปที่ 3.31 ลักษณะการทำงานของทาโคมิเตอร์แบบเหนี่ยวนำและแบบแสง

1.5 เอ็นโคดเดอร์(Encoders) มีลักษณะคล้ายทาโคมิเตอร์ โดยการส่งสัญญาณเป็นสัญญาณสูง-ต่ำ(1-0) ในจานของเอ็นโคดเดอร์จะแบ่งเป็นช่อง ๆ มีประมาณ 2-3 แถว การนับจะส่งสัญญาณเป็นจังหวะ สัญญาณนี้จะนำไปประมวลผลเพื่อวัดความเร็วหรือจำนวนรอบการหมุนของเพลาในหุ่นยนต์



รูปที่ 3.32 จานนับของเอนโคดเดอร์



รูปที่ 3.33 แสดงโครงสร้างของเอนโคดเดอร์

2.เซนเซอร์ภายนอก (External sensors)

เป็นการเลียนแบบการรับรู้ความรู้สึกของมนุษย์ คือ หู ตา จมูก รสชาติและสัมผัส (รสชาติและกลิ่นยังไม่มีการคิดค้น) เซนเซอร์ที่คิดค้นขึ้นจึงมีมากมายตามความต้องการในการใช้งาน เราสามารถแบ่งกลุ่มเซนเซอร์ภายนอก เป็น 3 กลุ่มคือ

2.1 เซนเซอร์ที่ใช้ในงานด้านแสง(Photaic sensors)

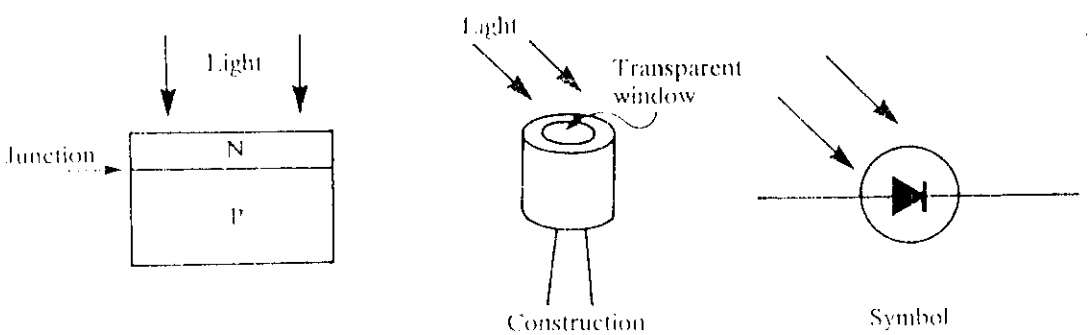
2.2 เซนเซอร์ที่ใช้ในการวัดแรง(Force sensors)

2.3 อื่น ๆ (Other sensors)

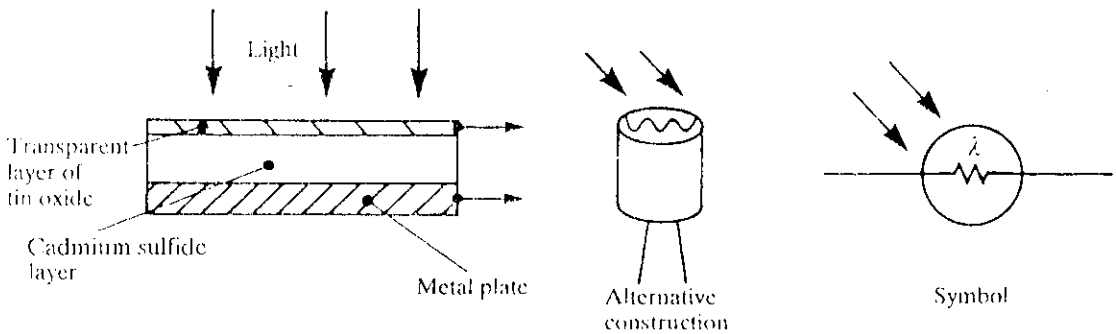
2.1 เซนเซอร์ที่ใช้ในงานด้านแสง (Photaic sensors)

เป็นเซนเซอร์ที่ใช้ในการตรวจสอบคลื่นไฟฟ้า ซึ่งคลื่นที่นิยมในการตรวจสอบส่วนมากจะเป็นแสงขาวและคลื่นอินฟาเรด เซนเซอร์ประเภทนี้ ได้แก่

2.1.1 ซิงเกิลโฟโต้เซลล์(Singlc photocell) เป็นพื้นฐานในการตรวจสอบด้านแสง หลักการทำงานคือ เมื่อ แสงหรือคลื่นอินฟาเรดที่มาตกกระทบเซนเซอร์จะเป็นผลทำให้เกิดการเปลี่ยนแปลงทางไฟฟ้า ทำให้สัญญาณที่ส่งไปประมวลผลเปลี่ยนไป

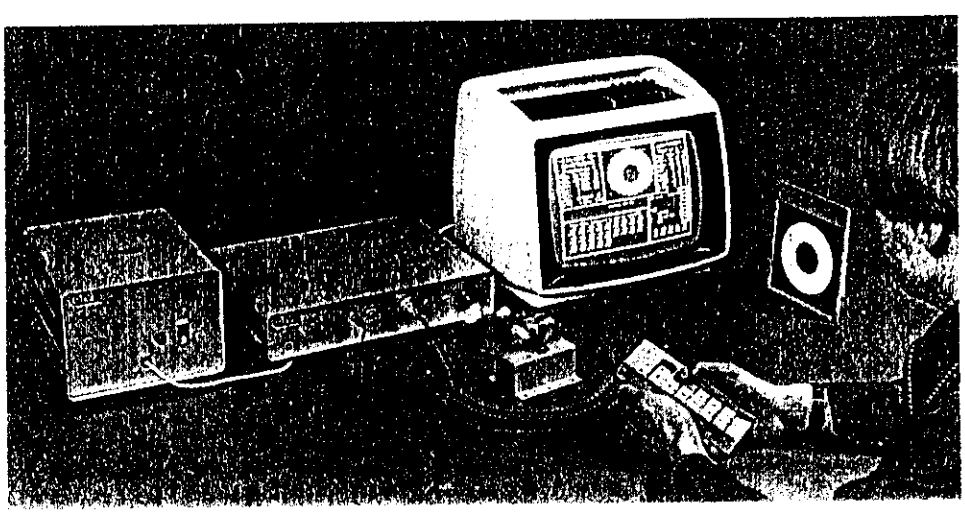


รูปที่ 3.34 เซนเซอร์ชนิดแคดเมี่ยมซัลไฟด์(Cadmuim sulfide)



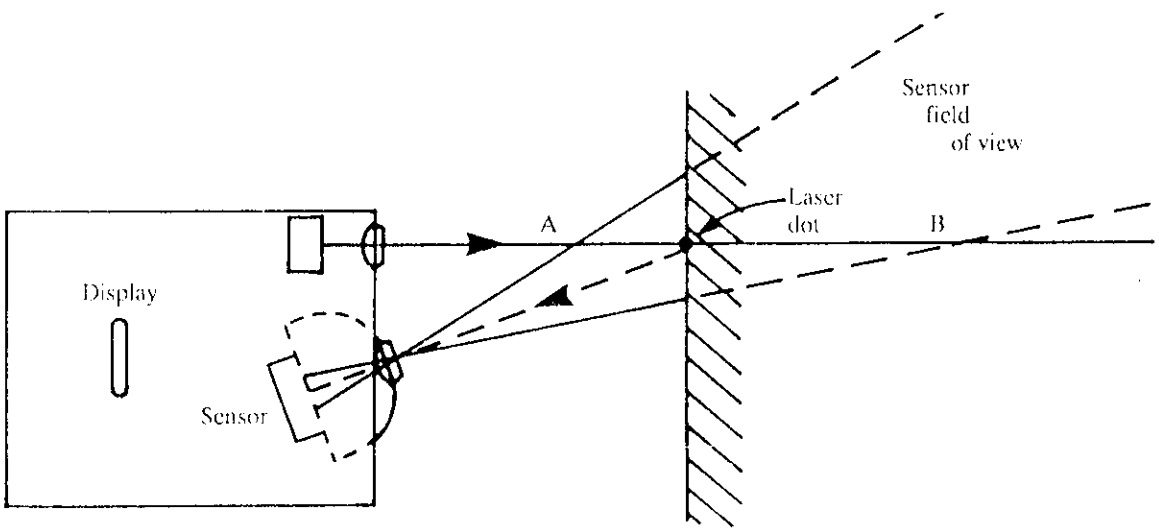
รูปที่ 3.35 โฟโต้ไดโอดเซนเซอร์ (Photodiode sensors)

2.1.2 แบบภาพ(Vision) เป็นเซนเซอร์ ที่ตรวจสอบโดยการรับภาพซึ่งได้รับจากกล้อง เข้ามาแยกระดับของสีต่าง ๆ และสามารถแสดงผลในอุปกรณ์แสดงผลชนิดต่าง ๆ เช่น จอโทรทัศน์ การตรวจสอบภาพแบบนี้จะให้ความละเอียดสูง แต่ค่าใช้จ่ายในการตรวจสอบย่อมสูงตามไปด้วย



รูปที่ 3.36 แสดงส่วนประกอบของระบบการตรวจสอบแบบภาพ(Vision)

2.1.3 เลเซอร์ (Laser sensing) การใช้เซนเซอร์เลเซอร์ส่วนมากจะใช้ในการวัดระยะทาง การตรวจสอบขนาดของชิ้นงาน หลักการทำงานมีตัวส่งแสงและตัวรับแสง นำสัญญาณที่ได้รับจากตัวรับแสงไปประมวลผลเพื่อทำการวิเคราะห์ตามที่ต้องการ



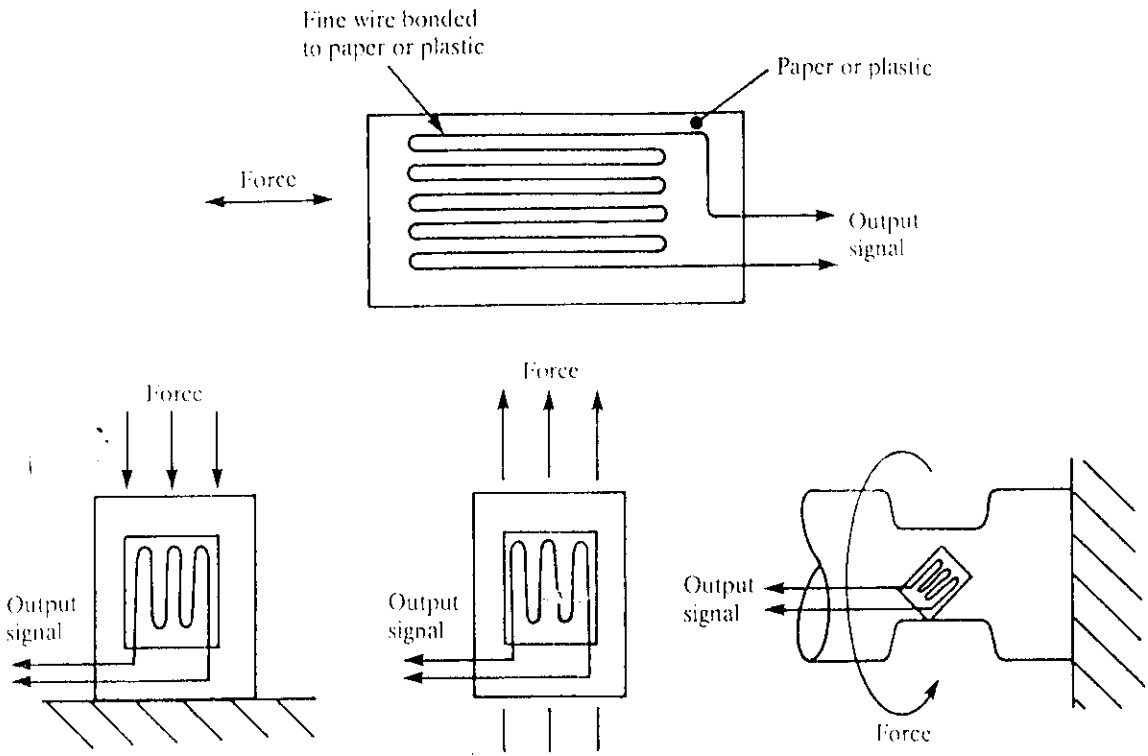
รูปที่ 3.37 แสดงการตรวจสอบระยะทางของเลเซอร์เซนเซอร์

2.2 การตรวจสอบแรง (Force sensing)

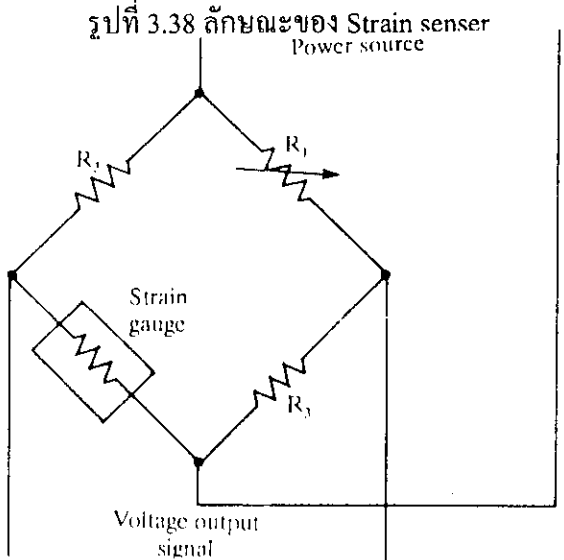
2.2.1 สเตรนเซนเซอร์ (Strain senser) ใช้ในการวัดความเครียดของวัสดุ ความสัมพันธ์คือ

Strain = ความยาวที่เพิ่มขึ้น / ความยาวเดิม

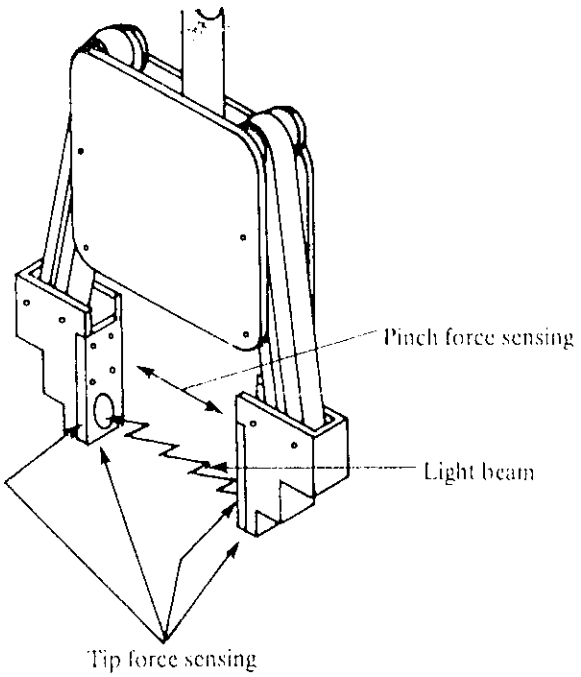
การทำงานโดยการเปลี่ยนแปลงความต่างศักย์ทางด้านไฟฟ้า และนำสัญญาณนี้ไปประมวลผล



รูปที่ 3.38 ลักษณะของ Strain senser



รูปที่ 3.39 วงจรที่ใช้ในการเปลี่ยนแปลงสัญญาณ ใน Strain sensor

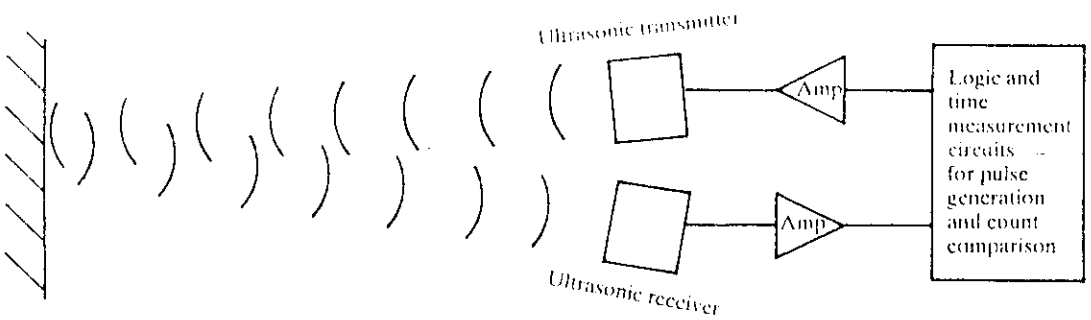


รูปที่ 3.40 การนำเอาStrain sensorไปใช้งาน

2.2.2 ทอร์กเซนเซอร์(Torque sensor) ใช้ในการวัดขนาดของแรงบิด ของวัตถุที่ต้องการทดสอบ

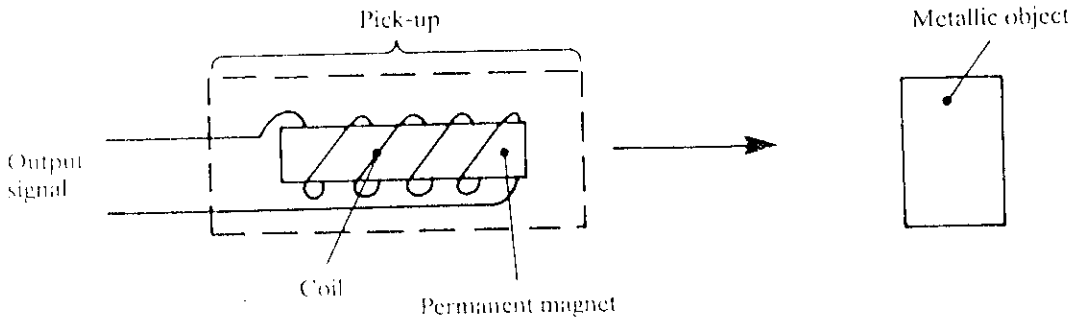
3. อื่น ๆ (Other sensor)

3.1 เซนเซอร์คลื่นเสียง (Acoustic sensor ,Ultrasonic sensor) โดยการส่งคลื่นเสียงจากตัวส่ง (Transmitter) ที่มีความถี่ 40 kHz จากตัวส่งไปสู่ตัวรับ (Receiver) ใช้ในการวัดระยะทาง การคาดคะเนขนาดของวัตถุ



รูปที่ 3.41 แสดงการทำงานของUltrasonic sensor

3.2 เซนเซอร์เหนี่ยวนำ (Magnetic sensor) ใช้ในการตรวจสอบวัตถุที่เป็นโลหะ โดยการเคลื่อนที่ของแม่เหล็กถาวรที่อยู่ในเซนเซอร์ ทำให้เกิดการส่งสัญญาณไฟฟ้า



รูปที่ 3.42 เซนเซอร์เหนี่ยวนำ

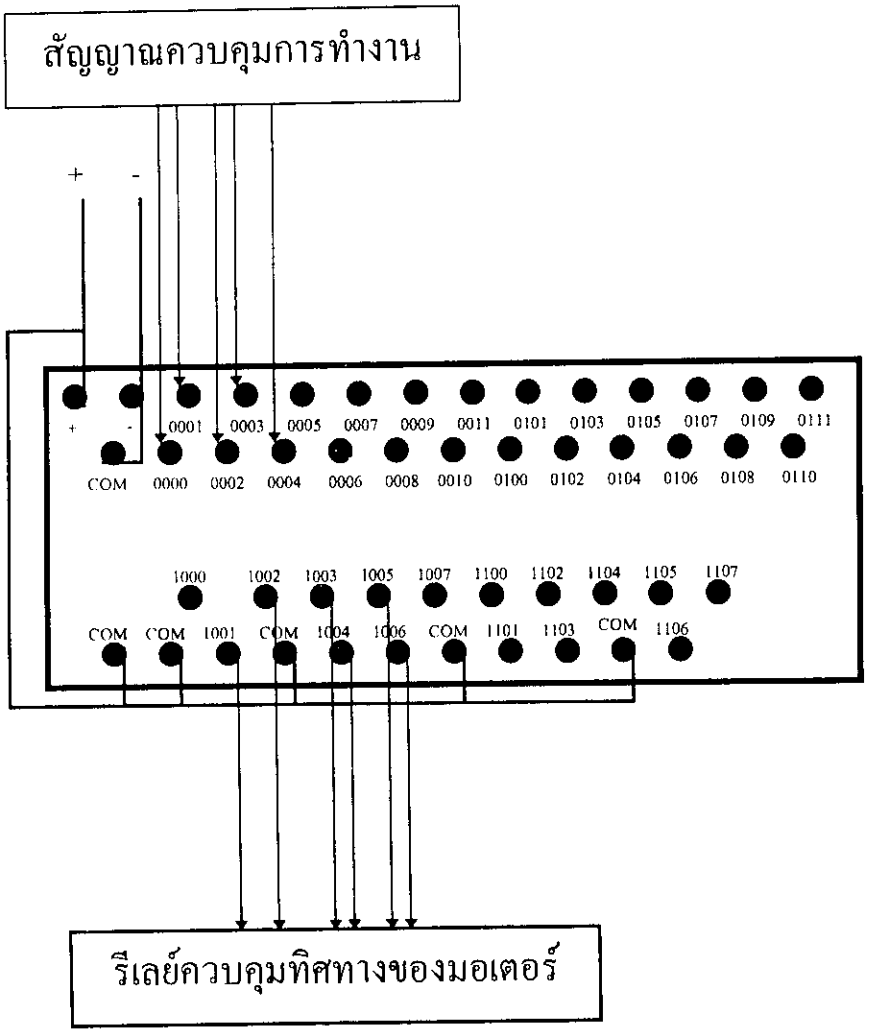
3.3 เซนเซอร์คาปาซิทีฟ (Capacitive sensor) เป็นเซนเซอร์ที่ใช้ในการตรวจสอบอโลหะ มีความสามารถในการตรวจสอบได้ในระยะใกล้ ๆ

เซนเซอร์ที่นิยมใช้โดยทั่วไปมีมากมาย การเลือกใช้เซนเซอร์ควรเลือกใช้เหมาะกับงาน ทั้งขนาดและประสิทธิภาพ เพื่อความเหมาะสมในการทำงาน จะต้องศึกษาการทำงานของเซนเซอร์ให้เข้าใจ ในการศึกษาโครงงานในครั้งนี้ได้เลือกใช้เซนเซอร์แบบแสงในการตรวจสอบเส้นทางและการตรวจสอบสีของลูกโป่ง

3.4.3 การออกแบบการเดินสายไฟในหุ่นยนต์

สายไฟฟ้าที่ต่อภายในหุ่นยนต์ประกอบด้วยสายไฟ 2 ประเภทคือ

1. สายไฟฟ้าสำหรับนำสัญญาณ สายไฟฟ้าชนิดนี้จะมีกระแสที่ไหลในสายต่ำมาก โดยปกติประมาณ 24 V. 50 mA. เช่น สายไฟฟ้า input - output ของ PLC เป็นต้น
2. สายไฟฟ้าสำหรับนำไฟฟ้าเป็นสายนำพลังงานให้มอเตอร์ทำงาน เป็นสายที่ต่อออกจากกล่องรีเลย์ มีกระแสที่ไหลอยู่ในสายนี้ 100-300 mA. เช่น สายไฟฟ้าที่ใช้ในการขับมอเตอร์ ขนาด 12 V. 300 mA. เป็นต้น



รูปที่ 3.43 แสดงการต่อสายไปกับ PLC

สัญญาณด้าน Input

0000 = สัญญาณวัดจำนวนรอบจากเอ็นโคดเดอร์

0001 = สัญญาณจากปุ่มสตาร์ท

0002 = สัญญาณลิมิตสวิทซ์ 1

0003 = สัญญาณลิมิตสวิทซ์ 2

0004 = เซนเซอร์

สัญญาณด้าน Output

1001 = สัญญาณควบคุมรีเลย์เพื่อให้มอเตอร์ด้านขวาหมุนไปด้านหน้า

1002 = สัญญาณควบคุมรีเลย์เพื่อให้มอเตอร์ด้านขวาหมุนไปด้านหลัง

1003 = สัญญาณควบคุมรีเลย์เพื่อให้มอเตอร์ด้านซ้ายหมุนไปด้านหน้า

1004 = สัญญาณควบคุมรีเลย์เพื่อให้มอเตอร์ด้านซ้ายหมุนไปด้านหลัง

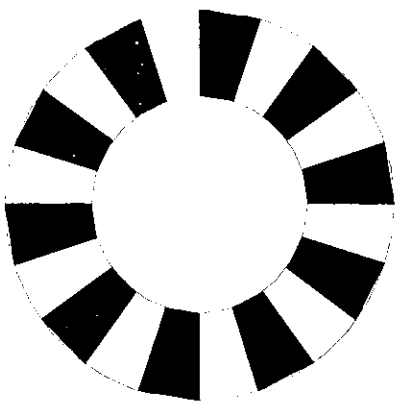
1005 = สัญญาณควบคุมรีเลย์เพื่อให้มอเตอร์แทงลูกโป่งหมุนออก

1006 = สัญญาณควบคุมรีเลย์เพื่อให้มอเตอร์แทงลูกโป่งหมุนเข้า

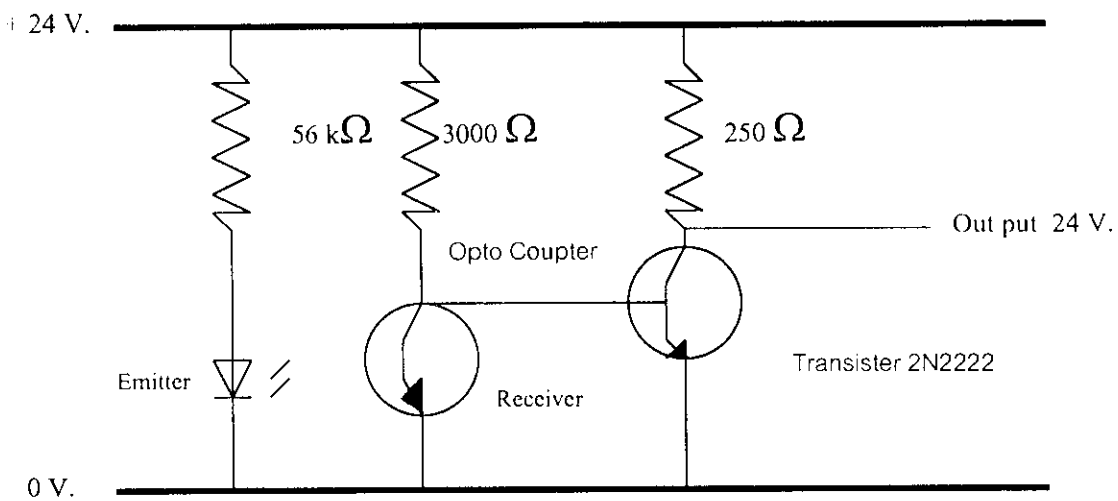
3.4.4 <u>การออกแบบเอ็นโคดเดอร์</u>

การมีลักษณะคล้ายทาโคมิเตอร์ โดยการส่งสัญญาณเป็นสัญญาณสูง-ต่ำ (1-0) ในจานของเอ็นโคดเดอร์จะแบ่งเป็นช่อง ๆ มีประมาณ 2-3 แถว การนับจะส่งสัญญาณเป็นจังหวะ สัญญาณนี้จะนำไปประมวลผลเพื่อวัดความเร็วหรือจำนวนรอบการหมุนของเพลาในหุ่นยนต์

การทดลองในครั้งนี้ได้ทำการสร้างเอ็นโคดเดอร์ขึ้นมา เพื่อทำการวัดระยะทางในการเดินทางของหุ่นยนต์เพื่อให้เกิดความแม่นยำในการเดินทางของหุ่นยนต์ รูปแบบที่ทำการสร้างดังรูป



รูปที่ 3.44 ลักษณะรูปร่างของโครงสร้าง Encoder



รูปที่ 3.45 วงจรของ Encoder

3.5 โปรแกรมควบคุมการทำงานของหุ่นยนต์

การโปรแกรมการทำงานโดยใช้ Syswin Support Software ซึ่งเป็นโปรแกรมที่ใช้ในเขียนชุดคำสั่งการทำงานและใช้ถ่ายคำสั่งนั้นลงไปใน PLC (ของ OMROM Model CPM1-A) ใช้กับไฟฟ้า 24 VDC มีขนาด 24 Input 16 Output ซึ่งเครื่องคอมพิวเตอร์ที่เหมาะกับการใช้งานควรมีขีดความสามารถอย่างน้อย ดังนี้คือ

- 50 MHz 80486 , running in cnhanced mode
- Ram 8 Mbytes
- เนื้อที่ว่างใน Hard disk อย่างน้อย 10 Mbytes
- ระบบแสดงผล VGA หรือ (800x600 SVGA)
- Microsoft Windows3.1 หรือสูงกว่า
- Mouse

ส่วนแรกของโปรแกรมเป็นการแสดงรหัสต่าง ๆ ที่ใช้ในโปรแกรม ซึ่งสัญลักษณ์เหล่านี้เป็นสิ่งจำเป็นที่ต้องทราบ เพราะจะทำให้ทราบการทำงานของโปรแกรมมากยิ่งขึ้น

ส่วนที่สองเป็นความสัมพันธ์ของรหัสต่างในโปรแกรม รหัสต่าง ๆ เมื่อใส่เงื่อนไขให้มีความสัมพันธ์กันอย่างเหมาะสม จะทำให้โปรแกรมสามารถทำงานได้ตามต้องการ โปรแกรมจะประกอบด้วย Ladder Diagram และ Boolean ผลการทำงานของสองภาษาจะเหมือนกัน

3.5.1 สัญลักษณ์ของรหัสต่าง ๆ

สัญลักษณ์ของรหัสต่าง ๆ (address) เพื่อให้ทราบความหมายของรหัสที่จะทำให้ทราบความสัมพันธ์ของรหัสเหล่านี้ ในบางครั้งได้ยกรหัสต่าง ๆ เข้าไปช่วยในการกำหนดเงื่อนไข เพื่อให้โปรแกรมสมบูรณ์ยิ่งขึ้น

Symbol List : Address AGV1 SWP 3/13/98 12:51:06 Page 1

000.00	Encoder
000.01	PB-start
000.02	Limit-sw1
000.03	Limit-sw2
000.04	Sensor
005.00	Cordinn forw
005.01	Forward1
005.02	Forward2
005.03	Turn1
005.06	Reset-cnt17
005.07	Turn#3
005.08	Forw6
005.10	Reset-cnt15
005.11	Turn2
005.12	Reset-cnt16
005.13	Forward4
005.14	Reset-cnt17
005.15	Forw5
010.01	R-forw1
010.02	M-rev
010.03	L-forw
010.04	L-rev
010.05	B-forw
010.06	B-rev
010.07	M-R-forw
010.08	Reset-cnt1
010.09	Reset-cnt2
011.00	M-L-rev
011.01	R-forw
011.02	Motor-R
011.03	Motor-L
011.04	R-forw2
011.05	L-forw2
011.06	M-forw
200.00	Start
200.01	Forward3
200.02	Reset-cnt11
200.03	Reset-cnt3
200.05	Reset-cnt19
200.07	Turn-R
200.08	Turn
200.09	Nnndetecting
200.10	Detecting
200.11	Nondetect

200.12	Detect
200.13	Confirmforw
TIM004	Delay-start
TIM005	Delay-forw2
TIM006	Delay-turn1
TIM010	Delay-forw3
TIM020	Delay-turn2
TIM021	Delay-forw4
TIM022	Delay-forw5
TIM023	Delay-turn3
TIM024	Delay-forw6
TIM030	Delay
TIM031	Reset-Turn
TIM032	Delay-rev
TIM033	Delay-b2
CNT001	Forw1m
CNT002	Forw2-1m
CNT003	turn-1
CNT011	forw-2m-1
CNT015	Turn-2
CNT016	Forw-1m-3
CNT017	Forw-1m-4
CNT018	Turn-3
CNT019	Forw-2m-2

B-forw	010.05
B-rev	010.06
Confirmforw	200.13
Confirm-forw	005.00
Delay	TIM030
Delay-b2	TIM033
Delay-forw2	TIM005
Delay-forw3	TIM010
Delay-forw4	TIM021
Delay-forw5	TIM022
Delay-forw6	TIM024
Delay-rev	TIM032
Delay-start	TIM004
Delay-turn1	TIM006
Delay-turn2	TIM020
Delay-turn3	TIM023
Detect	200.12
Detecting	200.10
Encoder	000.00
Forw1m	CNT001
Forw-1m-3	CNT016
Forw-1m-4	CNT017
Forw2-1m	CNT002
forw-2m-1	CNT011
Forw-2m-2	CNT019
Forw5	005.15
Forw6	005.09
Forward1	005.01
Forward2	005.02
Forward3	200.01
Forward4	005.13
L-forw	010.03
L-forw2	011.05
Limit-sw1	000.02
Limit-sw2	000.03
L-rev	010.04
M-forw	011.06
M-L-rev	011.00
Motor-L	011.03
Motor-R	011.02
M-rev	010.02
M-R-forw	010.07
Nondetect	200.11
Nondetecting	200.09

TB-start	000.01
Reset-cnt1	010.08
Reset-cnt11	200.02
Reset-cnt15	005.10
Reset-cnt16	005.12
Reset-cnt17	005.14
Reset-cnt19	200.05
Reset-cnt2	010.09
Reset-cnt3	200.03
Reset-Turn	TIM031
R-forw	011.01
R-forw1	010.01
R-forw2	011.04
Rset-cnt17	005.06
Sensor	000.04
Start	200.00
Turn	200.08
Turn#3	005.07
Turn1	005.03
Turn-1	CNT003
Turn2	005.11
Turn-2	CNT015
Turn-3	CNT018
Turn-R	200.07

3.5.2 โปรแกรมควบคุมการทำงาน

โปรแกรมการทำงานมีภาษาแลดเดอร์(Ladder Diagram) และภาษาบูลลีน(Boolean) แสดงความสัมพันธ์ของรหัสต่าง ๆ ดังแสดงไว้หน้าถัดไป

ภาษาแลดเดอร์(Ladder Diagram) เป็นภาษาที่ประกอบด้วยสัญลักษณ์คล้ายกับหน้าสัมผัส มีลักษณะคล้ายวงจรรีเลย์ สามารถเข้าใจและเขียนง่าย แต่เครื่องมือที่ใช้สำหรับเขียนและถ่ายเท โปรแกรมต้องมีความสามารถสูง ๆ สัญลักษณ์ต่าง ๆ สามารถดูได้จากภาคผนวก ก

ภาษาบูลลีน(Boolean) หรือบางครั้งเรียกภาษานิวโมนิค(Mnemonic) เป็นภาษาพื้นฐานของ PLC สามารถป้อนโปรแกรมได้ง่าย นิยมใช้เครื่องโปรแกรมขนาดเล็ก(Handy Programmer)

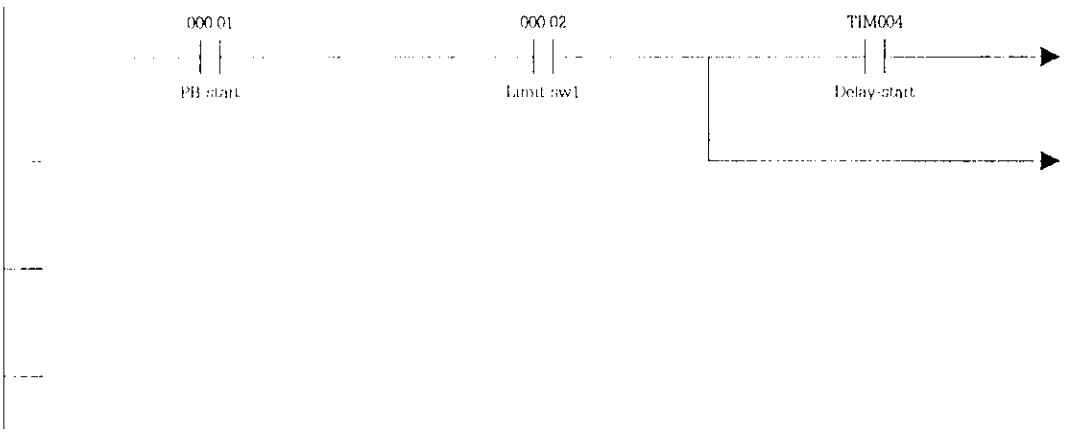
Ladder Diagram - 1:1 AGV1.SWP 3/13/98 12:28:07 Page 1

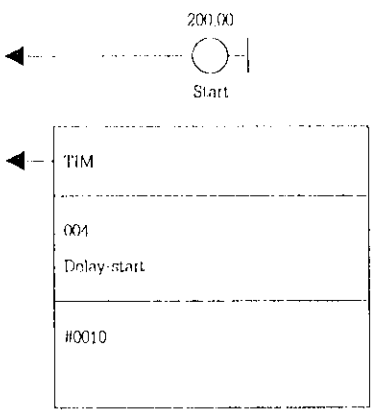
Main 1 - AGVs

This program demonstrates that the robot walks along the guideline
for detecting the target ballon and bursting it.

Network 1 - Start-condition

Push main switch and then push start switch to start system.





00000 LD 000.01 PB start

Ladder Diagram - 1:1 AGV1.SWP 3/13/98 12:28:07 Page 2

00001	AND	000.02	Limit-sw1
00002	OUT	TR0	TR0
00003	AND	TIM004	Delay-start
00004	OUT	200.00	Start
00005	LD	TR0	TR0
00006	TIM	004 #0010	Delay-start #0010

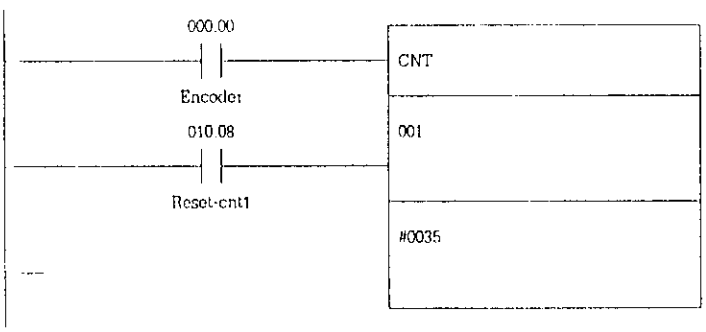
000.01	PB-start
000.02	Limit-sw1
TIM004.F	Delay-start
200.00	Start
TIM004	Delay-start

000.01 PB-start
1:1 1:11
000.02 Limit-sw1
1:1 1:6 1:7 1:9
1:30 1:31 1:33 1:35
TIM004.F Delay-start
Not Used
200.00 Start
*1:1 1:4
TIM004 Delay-start
*1:1

Network 2 - Counter1

Forward 1m (35 pulses)

00007



00007	LD	000.00	Encoder
00008	LD	010.08	Reset-cnt1

Ladder Diagram - 1:2 AGV1.SWP 3/13/98 12:28:07 Page 3

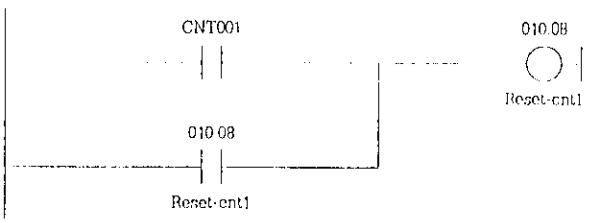
00009 CNT 001 #0035 001 #0035

000.00 Encoder
010.08 Reset-cnt1
CNT001

000.00 Encoder
1:2 1:13 1:16 1:20
1:24 1:28 1:37 1:41
1:44
010.08 Reset-cnt1
1:2 *1:3 1:4 1:6
1:7
CNT001
*1:2 1:3

Network 3 - Reset counter1

00010



00010 LD CNT001 CNT001
00011 OR 010.08 Reset-cnt1
00012 OUT 010.08 Reset-cnt1

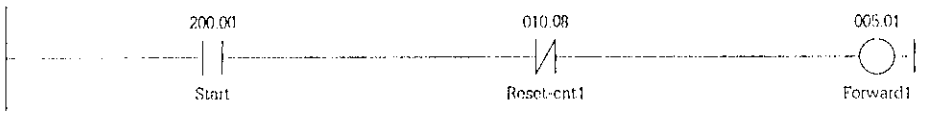
CNT001.F
010.08 Reset-cnt1

CNT001.F
Not Used
010.08 Reset-cnt1
1:2 *1:3 1:4 1:6
1:7

Ladder Diagram - 1:4 AGV1.SWP 3/13/98 12:20:07 Page 4

Network 4 - Forward 1m

00013



00013	LD	200.00	Start
00014	AND NOT	010.08	Reset-cnt1
00015	OUT	005.01	Forward1

200.00	Start
010.08	Reset-cnt1
005.01	Forward1

200.00 Start

*1:1 1:4

010.08 Reset-cnt1

1:2 *1:3 1:4 1:6

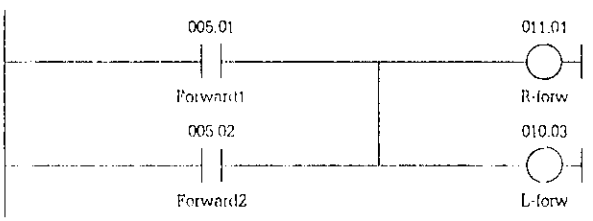
1:7

005.01 Forward1

*1:4 1:5

Network 5 - Motor Forward

00016



00016	LD	005.01	Forward1
00017	OR	005.02	Forward2
00018	OUT	011.01	R-forw
00019	OUT	010.03	L-forw

005.01	Forward1
005.02	Forward2
011.01	R-forw

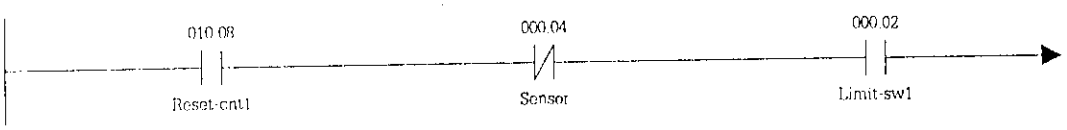
Ladder Diagram - 1:5 AGV1.SWP 3/13/98 12:28:07 Page 5

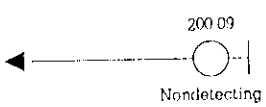
```
010.03                  L-forw

005.01                  Forward1
        *1:4                    1:5
005.02                  Forward2
        1:5                     *1:12           1:13
011.01                  R-forw
        *1:5
010.03                  L-forw
        *1:5
```

Network 6 - Nondetecting

Detecting the green balloon






```
00020   LD          010.08          Reset-cnt1
00021   AND NOT     000.04          Sensor
00022   AND         000.02          Limit-sw1
00023   OUT         200.09          Nondetecting
```

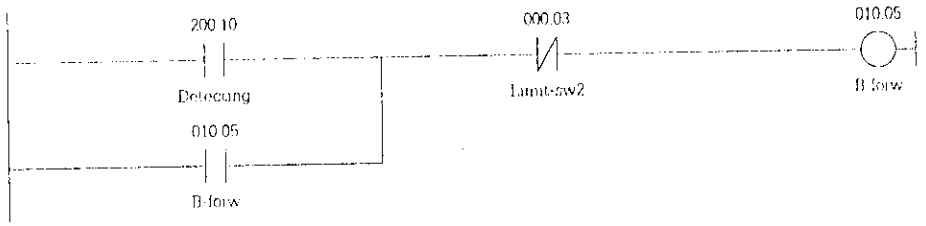
```
010.08                  Reset-cnt1
000.04                  Sensor
000.02                  Limit-sw1
200.09                  Nondetecting

010.08                  Reset-cnt1
        1:2                     *1:3            1:4             1:6
        1:7
000.04                  Sensor
```

Ladder Diagram - 1:7 AGV1.SWP 3/13/98 12:28:07 Page 7

200.10 Detecting
*1:7 1:8

Network 8 : B-motor forward



00028	LD	200.10	Detecting
00029	OR	010.05	B-forw
00030	AND NOT	000.03	Limit-sw2
00031	OUT	010.05	B-forw

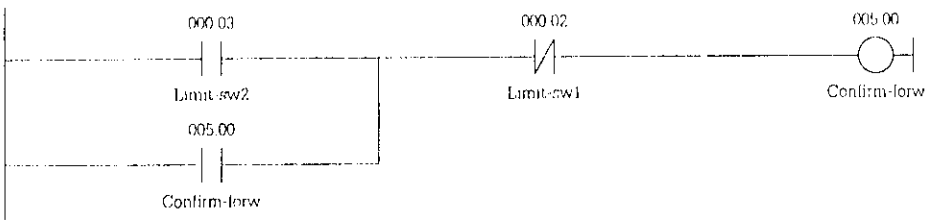
200.10	Detecting
010.05	B-forw
000.03	Limit-sw2

200.10 Detecting
*1:7 1:8
010.05 B-forw
*1:8 *1:32
000.03 Limit-sw2
1:8 1:9 1:32 1:33

Ladder Diagram - 1:9 AGV1.SWP 3/13/98 12:28:07 Page 8

Network 9 - Confirm forward

00012

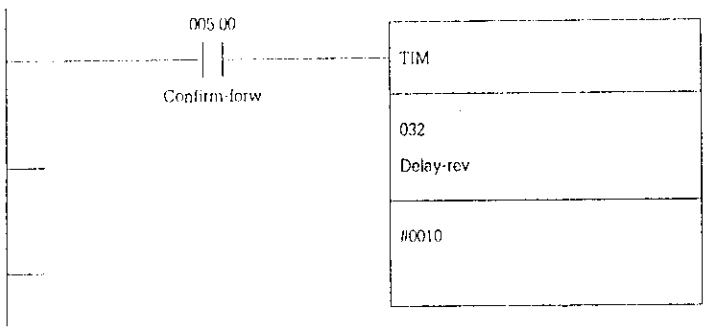


00032	LD	000.03	Limit-sw2
00033	OR	005.00	Confirm-forw
00034	AND NOT	000.02	Limit-sw1
00035	OUT	005.00	Confirm-forw

000.03		Limit-sw2		
005.00		Confirm-forw		
000.02		Limit-sw1		
000.03		Limit-sw2		
1:8	1:9	1:32	1:33	
005.00		Confirm-forw		
*1:9	1:10			
000.02		Limit-sw1		
1:1	1:6	1:7	1:9	
1:30	1:31	1:33	1:35	

Ladder Diagram - 1:10 AGV1.SWP 3/13/98 12:28:07 Page 9

Network 10 - Delay



00036	LD	005.00	Confirm-forw
00037	TIM	032 #0010	Delay-rev #0010

005.00	Confirm-forw
TIM032	Delay-rev

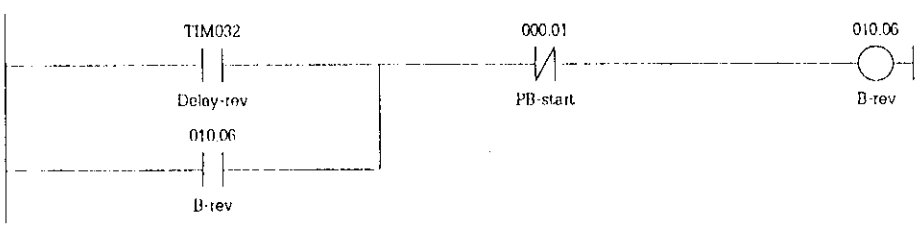
005.00 Confirm-forw

*1:9 1:10

TIM032 Delay-rev

*1:10 1:11

Network 11 - B-motor reverse



00038	LD	TIM032	Delay-rev
00039	OR	010.06	B-rev
00040	AND NOT	000.01	PB-start
00041	OUT	010.06	B-rev

TIM032.F	Delay-rev
010.06	B-rev

Ladder Diagram - 1:11 AGV1.SWP 3/13/98 12:28:07 Page 10

000.01 PB-start

TIM032.F Delay-rev

Not Used

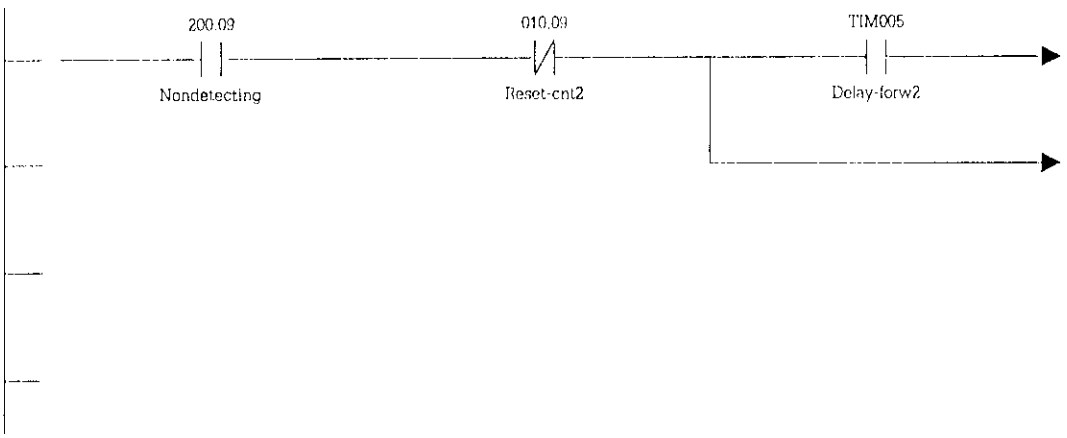
010.06 B-rev

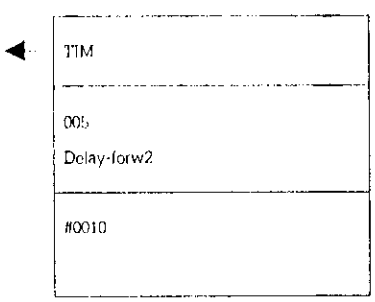
*1:11 *1:35

000.01 PB-start

1:1 1:11

Network 12 - Forward2





Ladder Diagram - 1:12 AGV1.SWP 3/13/98 12:28:07 Page 11

00042	LD	200.09	Nondetecting
00043	AND NOT	010.09	Reset-cnt2
00044	OUT	TR0	TR0
00045	AND	TIM005	Delay-forw2
00046	OUT	005.02	Forward2
00047	LD	TR0	TR0
00048	TIM	005 #0010	Delay-forw2 #0010

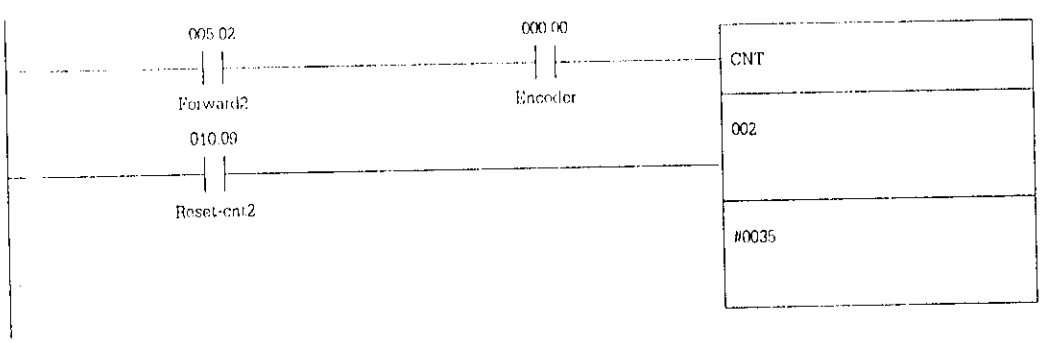
200.09	Nondetecting
010.09	Reset-cnt2
TIM005.F	Delay-forw2
005.02	Forward2
TIM005	Delay-forw2

200.09		Nondetecting			
	*1:6		1:12		
010.09		Reset-cnt2			
	1:12		1:13	*1:14	1:15
TIM005.F		Delay-forw2			
Not Used					
005.02		Forward2			
	1:5		*1:12	1:13	
TIM005		Delay-forw2			
	*1:12				

Network 13 - Counter2

Forward 1m (35 pulses)

00049



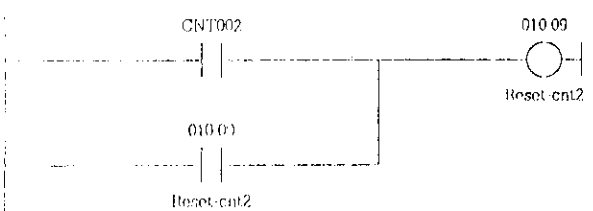
00049	LD	005.02	Forward2
00050	AND	000.00	Encoder

00051	LD	010.09	Reset-cnt2
00052	CNT	002 #0035	002 #0035

002.02 Forward2
000.00 Encoder
010.09 Reset-cnt2
CNT002

002.02 Forward2
1:5 *1:12 1:13
000.00 Encoder
1:2 1:13 1:16 1:20
1:24 1:28 1:37 1:41
1:44
010.09 Reset-cnt2
1:12 1:13 *1:14 1:15
CNT002
*1:13 1:14

Network 14 - Reset counter2



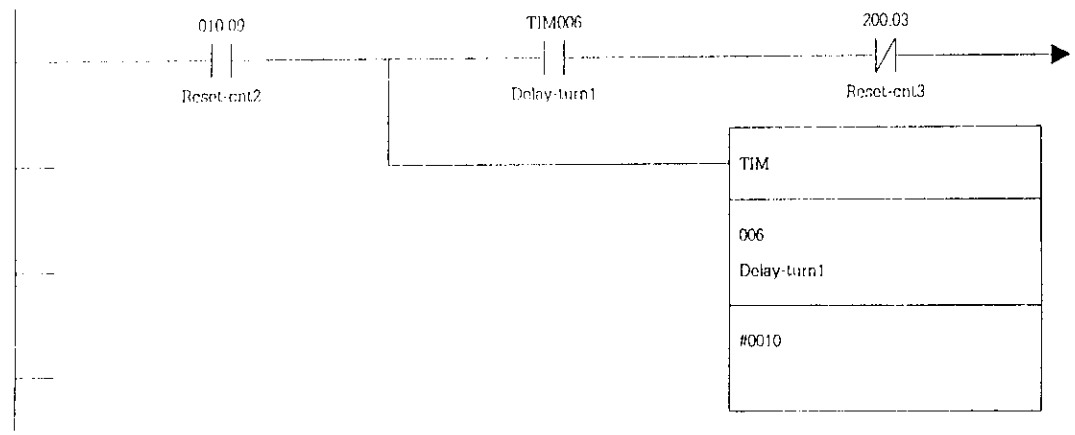
00053	LD	CNT002	CNT002
00054	OR	010.09	Reset-cnt2
00055	OUT	010.09	Reset-cnt2

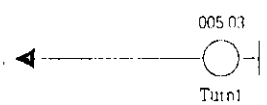
CNT002.F
010.09 Reset-cnt2

CNT002.F
Not Used
010.09 Reset-cnt2
1:12 1:13 *1:14 1:15

Ladder Diagram - 1:14 AGV1.SWP 3/13/98 12:28:07 Page 13

Network 15 - Turn left #1





00056	LD	010.09	Reset-cnt2
00057	OUT	TR0	TR0
00058	AND	TIM006	Delay-turn1
00059	AND NOT	200.03	Reset-cnt3
00060	OUT	005.03	Turn1
00061	LD	TR0	TR0
00062	TIM	006 #0010	Delay-turn1 #0010

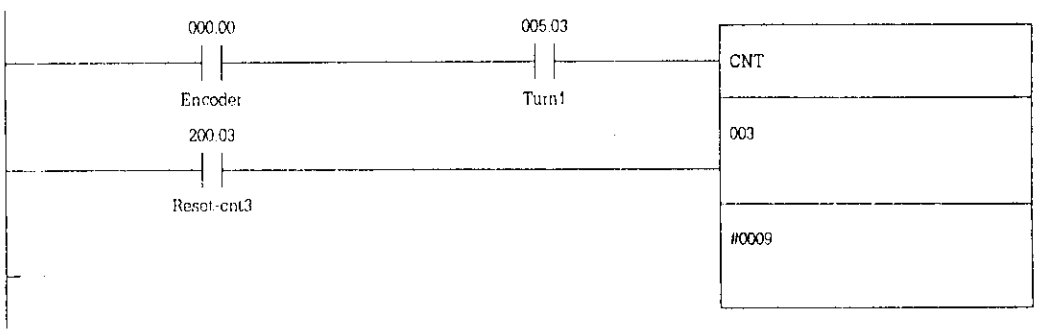
010.09	Reset-cnt2
TIM006 F	Delay-turn1

200.03 Reset-cnt3
005.03 Turn1
TIM006 Delay-turn1

010.09 Reset-cnt2
1:12 1:13 *1:14 1:15
TIM006 F Delay-turn1
Not Used
200.03 Reset-cnt3
1:15 1:16 *1:17 1:19
005.03 Turn1
*1:15 1:16 1:18
TIM006 Delay-turn1
*1:15

Network 16 - conter3

Turn left (9 pulses)



00063	LD	000.00	Encoder
00064	AND	005.03	Turn1
00065	LD	200.03	Reset-cnt3
00066	CNT	003 #0009	003 #0009

000.00 Encoder
005.03 Turn1
200.03 Reset-cnt3
CNT003

000.00 Encoder
1:2 1:13 1:16 1:20

Ladder Diagram - 1:16 AGV1.SWP 3/13/98 12:28:07 Page 15

1:24 1:28 1:37 1:41

1:44

005.03 Turn1

*1:15 1:16 1:18

200.03 Reset-cnt3

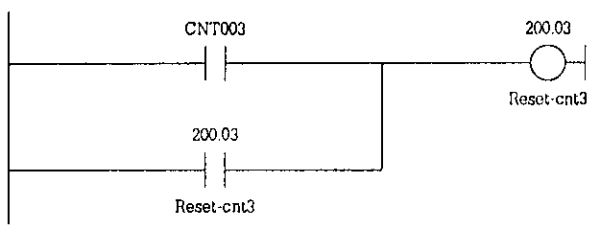
1:15 1:16 *1:17 1:19

CNT003

*1:16 1:17

Network 17 - Reset counter3

00067



00067	LD	CNT003	CNT003
00068	OR	200.03	Reset-cnt3
00069	OUT	200.03	Reset-cnt3

CNT003.F

200.03 Reset-cnt3

CNT003.F

Not Used

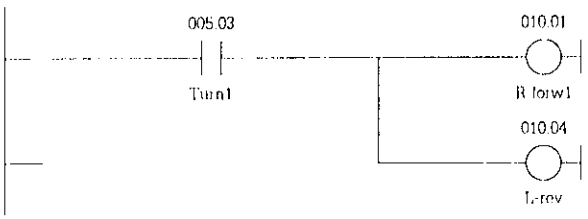
200.03 Reset-cnt3

1:15 1:16 *1:17 1:19

Ladder Diagram - 1.18 AGV1 SWP 3/13/98 12:28:07 Page 16

Network 18 - Turn left1

00070



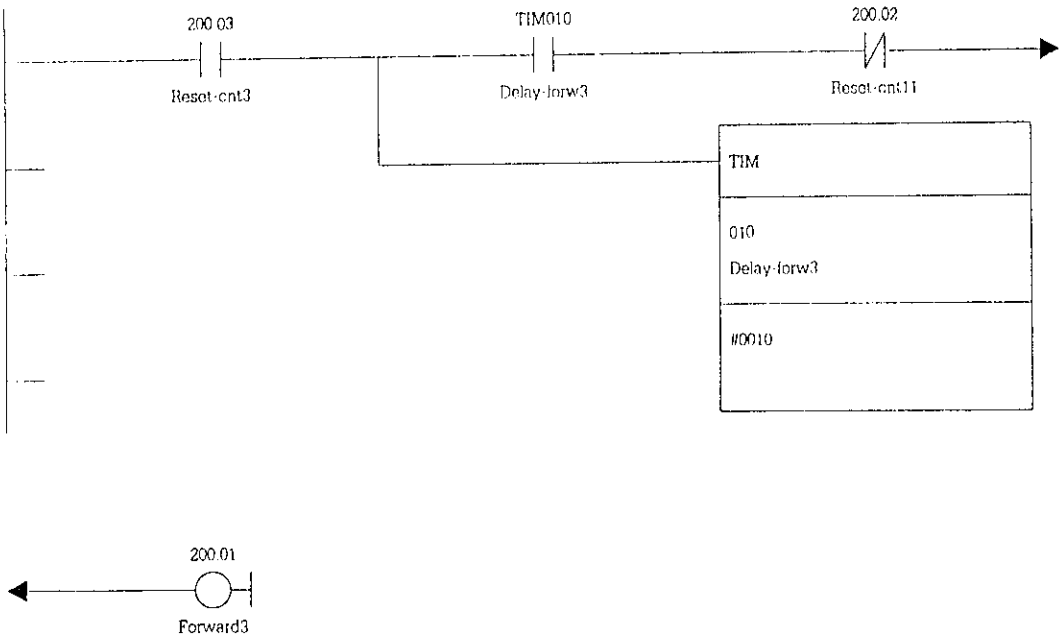
00070	LD	005.03	Turn1
00071	OUT	010.01	R-forw1
00072	OUT	010.04	L-rev

005.03 Turn1
010.01 R-forw1
010.04 L-rev

005.03 Turn1
*1:15 1:16 1:18
010.01 R-forw1
*1:18
010.04 L-rev
*1:18

Ladder Diagram - 1:19 AGV1.SWP 3/13/98 12:28:07 Page 17

Network 19 - Forward3



00073	LD	200.03	Reset-cnt3
00074	OUT	TR0	TR0
00075	AND	TIM010	Delay-forw3
00076	AND NOT	200.02	Reset-cnt11
00077	OUT	200.01	Forward3
00078	LD	TR0	TR0
00079	TIM	010 #0010	Delay-forw3 #0010

200.03	Reset-cnt3
TIM010.F	Delay-forw3

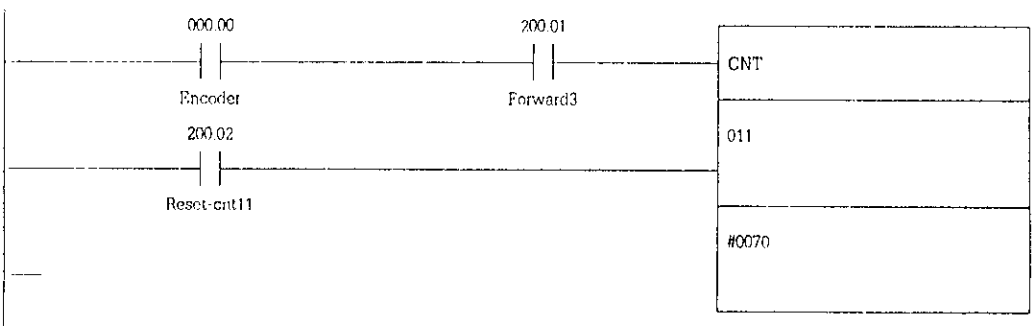
Ladder Diagram - 1:19 AGV1.SWP 3/13/98 12:28:07 Page 18

200.02		Reset-cnt11	
200.01		Forward3	
TIM010		Delay-forw3	
200.03		Reset-cnt3	
1:15	1:16	*1:17	1:19
TIM010.F		Delay-forw3	
Not Used			
200.02		Reset-cnt11	
1:19	1:20	*1:21	1:23
200.01		Forward3	
*1:19	1:20	1:22	
TIM010		Delay-forw3	
*1:19			

Network 20 - Counter11

Forward 2m (70 pulses)

00080



00080	LD	000.00	Encoder
00081	AND	200.01	Forward3
00082	LD	200.02	Reset-cnt11
00083	CNT	011 #0070	011 #0070

000.00	Encoder
200.01	Forward3
200.02	Reset-cnt11
CNT011	

000.00		Encoder	
1:2	1:13	1:16	1:20

Ladder Diagram - 1:20 AGV1.SWP 3/13/98 12:28:07 Page 19

1:24 1:28 1:37 1:41

1:44

200.01 Forward3

*1:19 1:20 1:22

200.02 Reset-cnt11

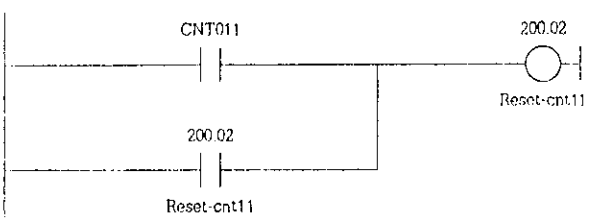
1:19 1:20 *1:21 1:23

CNT011

*1:20 1:21

Network 21 - Reset counter11

00084



00084	LD	CNT011	CNT011
00085	OR	200.02	Reset-cnt11
00086	OUT	200.02	Reset-cnt11

CNT011.F

200.02 Reset-cnt11

CNT011.F

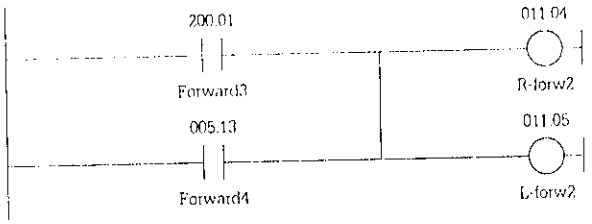
Not Used

200.02 Reset-cnt11

1:19 1:20 *1:21 1:23

Ladder Diagram - 1:22 AGV1.SWP 3/13/98 12:28:07 Page 20

Network 22 - Motor forward



00097	LD	200.01	Forward3
00098	OR	005.13	Forward4
00099	OUT	011.04	R-forw2
00090	OUT	011.05	L-forw2

200.01		Forward3
005.13		Forward4
011.04		R-forw2
011.05		L-forw2

200.01 Forward3

*1:19 1:20 1:22

005.13 Forward4

1:22 *1:27 1:28

011.01 R-forw2

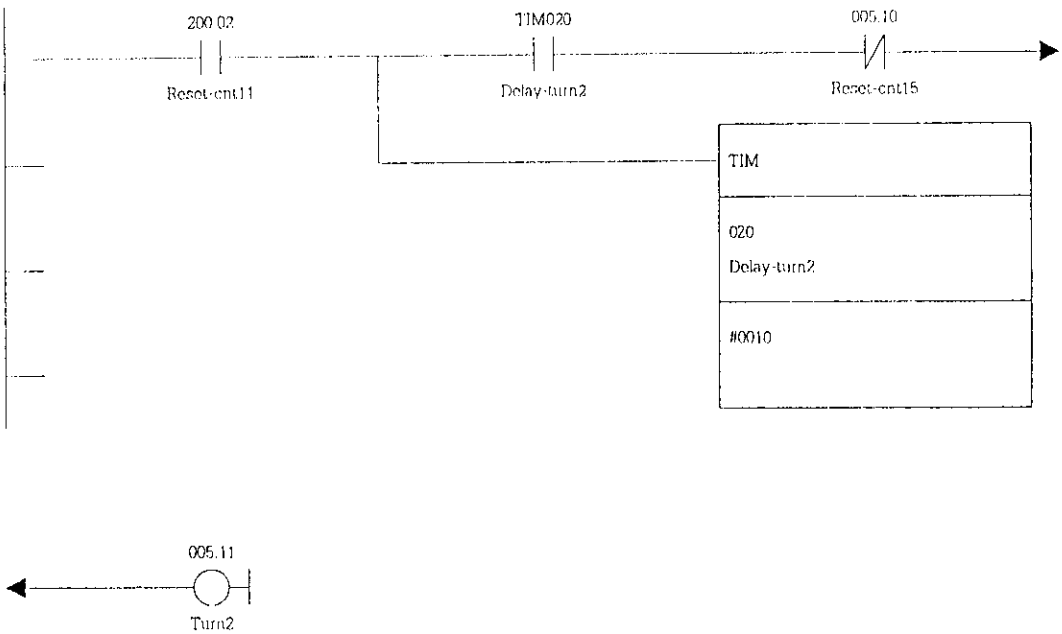
*1:22

011.05 L-forw2

*1:22

Ladder Diagram - 1:23 AGV1.SWP 3/13/98 12:28:07 Page 21

Network 23 - Turn left #2



00091	LD	200.02	Reset-cnt11
00092	OUT	TR0	TR0
00093	AND	TIM020	Delay-turn2
00094	AND NOT	005.10	Reset-cnt15
00095	OUT	005.11	Turn2
00096	LD	TR0	TR0
00097	TIM	020 #0010	Delay-turn2 #0010

200.02	Reset-cnt11
TIM020.F	Delay-turn2

Ladder Diagram - 1:23 AGV1.SWP 3/13/98 12:28:07 Page 22

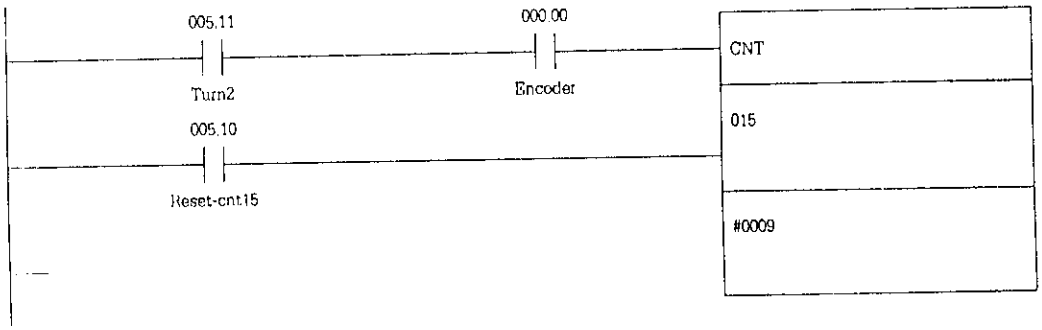
005.10 Reset-cnt15
005.11 Turn2
TIM020 Delay-turn2

200.02 Reset-cnt11
1.19 1:20 *1:21 1:23
TIM020.F Delay-turn2
Not Used
005.10 Reset-cnt15
1:23 1:24 *1:25 1:27
005.11 Turn2
*1:23 1:24 1:26
TIM020 Delay-turn2
*1:21

Network 24 - Counter 15

Turn left (9 pulses)

00098



00098 LD 005.11 Turn2
00099 AND 000.00 Encoder
00100 LD 005.10 Reset-cnt15
00101 CNT 015 #0009 015 #0009

005.11 Turn2
000.00 Encoder
005.10 Reset-cnt15
CNT015

005.11 Turn2
*1:23 1:24 1:26

Ladder Diagram - 1:24 AGV1.SWP 3/13/98 12:28:07 Page 23

000.00 Encoder

1:2 1:13 1:16 1:20

1:24 1:28 1:37 1:41

1:44

005.10 Reset-cnt15

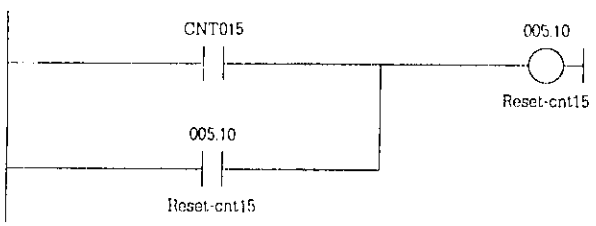
1:23 1:24 *1:25 1:27

CNT015

*1:24 1:25

Network 25 - Reset counter15

00102



00102 LD CNT015 CNT015

00103 OR 005.10 Reset-cnt15

00104 OUT 005.10 Reset-cnt15

CNT015.F

005.10 Reset-cnt15

CNT016.F

Not Used

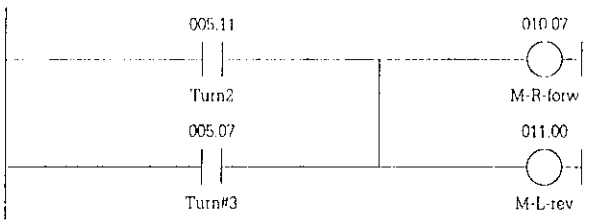
005.10 Reset-cnt15

1:23 1:24 *1:25 1:27

Ladder Diagram - 1:26 AGV1.SWP 3/13/98 12:28:07 Page 24

Network 26 - Motor turn left

00105



00105	LD	005.11	Turn2
00106	OR	005.07	Turn#3
00107	OUT	010.07	M-R-forw
00108	OUT	011.00	M-L-rev

005.11	Turn2
005.07	Turn#3
010.07	M-R-forw
011.00	M-L-rev

005.11 Turn2

*1:23 1.24 1:26

005.07 Turn#3

1:26 *1:40 1:41

010.07 M-R-forw

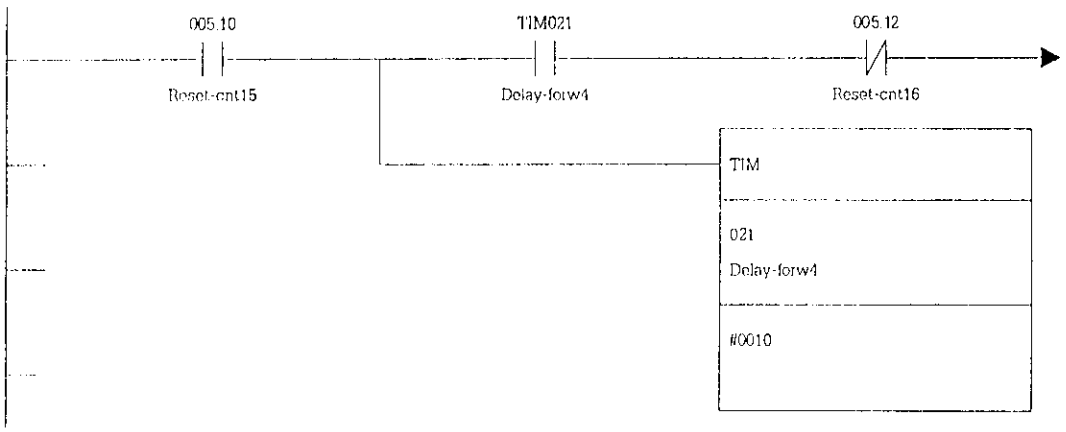
*1:26

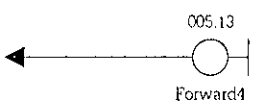
011.00 M-L-rev

*1:26

Ladder Diagram - 1:27 AGV1.SWP 3/13/98 12:28:07 Page 25

Network 27 - Forward #3





00109	LD	005.10	Reset-cnt15
00110	OUT	TR0	TR0
00111	AND	TIM021	Delay-forw4
00112	AND NOT	005.12	Reset-cnt16
00113	OUT	005.13	Forward4
00114	LD	TR0	TR0
00115	TIM	021 #0010	Delay-forw4 #0010

005.10	Reset-cnt15
TIM021 F	Delay-forw4

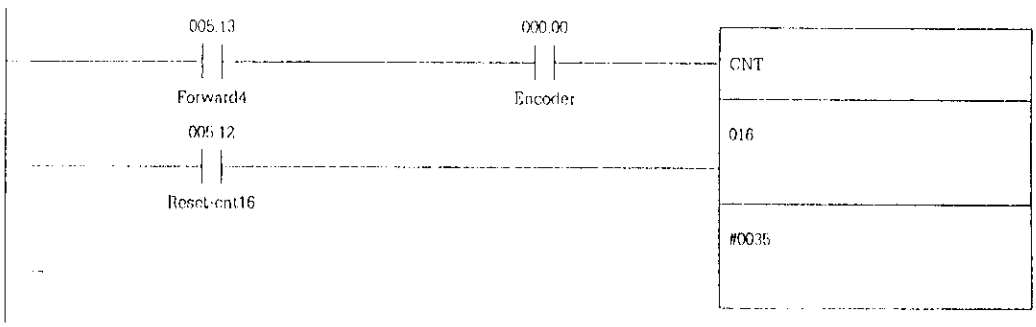
005.12 Reset-cnt16
005.13 Forward4
TIM021 Delay-forw4

005.10 Reset-cnt15
1:23 1:24 *1:25 1:27
TIM021 F Delay-forw4
Not Used
005.12 Reset-cnt16
1:27 1:28 *1:29 1:30
1:31
005.13 Forward4
1:22 *1:27 1:28
TIM021 Delay-forw4
*1:27

Network 28 - Counter16

Forward 1m (35 pulses)

00116



00116	LD	005.13	Forward4
00117	AND	000.00	Encoder
00118	LD	005.12	Reset-cnt16
00119	CNT	016 #0035	016 #0035

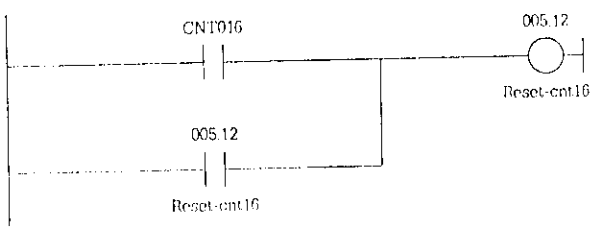
005.13 Forward4
000.00 Encoder
005.12 Reset-cnt16
CNT016

005.13 Forward4

Ladder Diagram - 1:28 AGV1.SWP 3/13/98 12:28:07 Page 27

```
          1:22          *1:27                    1:28
000.00                  Encoder
          1:2           1:13          1:16                1:20
          1:24          1:28          1:37                1:41
          1:44
005.12                  Reset-cnt16
          1:27          1:28          *1:29               1:30
          1:31
CNT016
          *1:28         1:29
```

Network 29 - Reset counter



00120	LD	CNT016	CNT016
00121	OR	005.12	Reset-cnt16
00122	OUT	005.12	Reset-cnt16

CNT016 F
005.12 Reset-cnt16

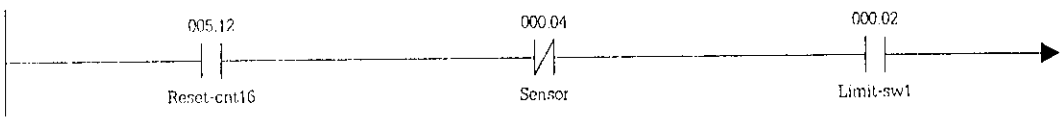
CNT016 F
Not Used
005.12 Reset-cnt16

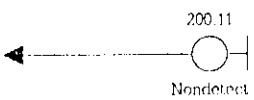
```
          1:27          1:28          *1:29               1:30
          1:31
```

Ladder Diagram - 1:30 AGV1.SWP 3/13/98 12:28:07 Page 28

Network 30 - Nondetecting

Robot detects the green balloon





00123	LD	005.12		Reset-cnt16
00124	AND NOT	000.04		Sensor
00125	AND	000.02		Limit-sw1
00126	OUT	200.11		Nondetect

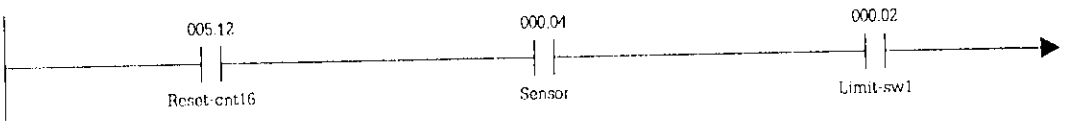
005.12	Reset-cnt16
000.04	Sensor
000.02	Limit-sw1
200.11	Nondetect

005.12		Reset-cnt16		
	1:27	1:28	*1.29	1:30
	1:31			
000.04		Sensor		
	1:6	1:7	1:30	1:31
000.02		Limit-sw1		
	1:1	1:6	1:7	1:9
	1:30	1:31	1:33	1:35
200.11		Nondetect		
	*1:30	1:36		

Ladder Diagram - 1:31 AGV1.SWP 3/13/98 12:28:07 Page 29

Network 31 - Detecting

Robot detects the target balloon(white balloon).



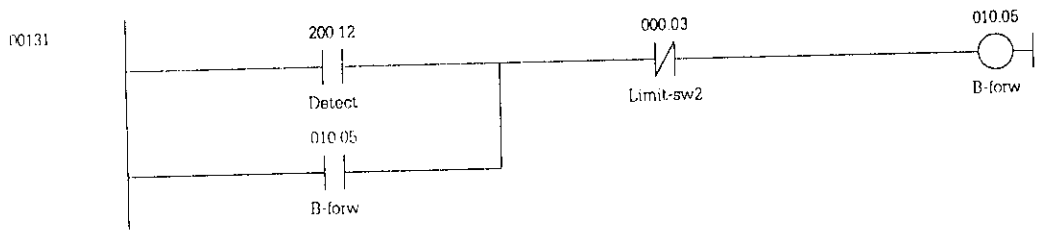
00127	LD	005.12		Reset-cnt16
00128	AND	000.04		Sensor
00129	AND	000.02		Limit-sw1
00130	OUT	200.12		Detect

005.12	Reset-cnt16
000.04	Sensor
000.02	Limit-sw1
200.12	Detect

005.12		Reset-cnt16		
	1:27	1:28	*1:29	1:30
	1:31			
000.04		Sensor		
	1:6	1:7	1:30	1:31
000.02		Limit-sw1		
	1:1	1:6	1:7	1:9
	1:30	1:31	1:33	1:35
200.12		Detect		
	*1:31	1:32		

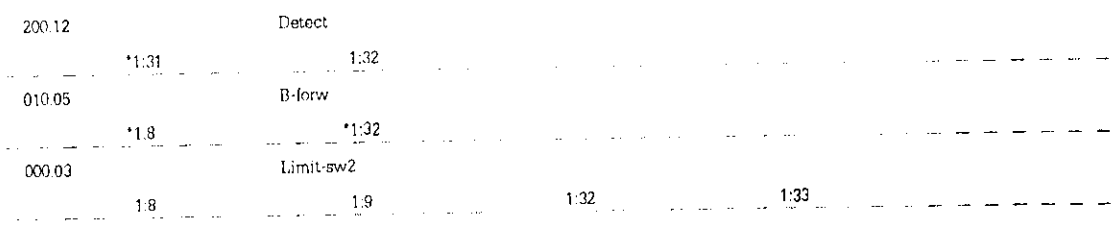
Ladder Diagram - 1:32 AGV1.SWP 3/13/98 12:28:07 Page 30

Network 32 - B-forward

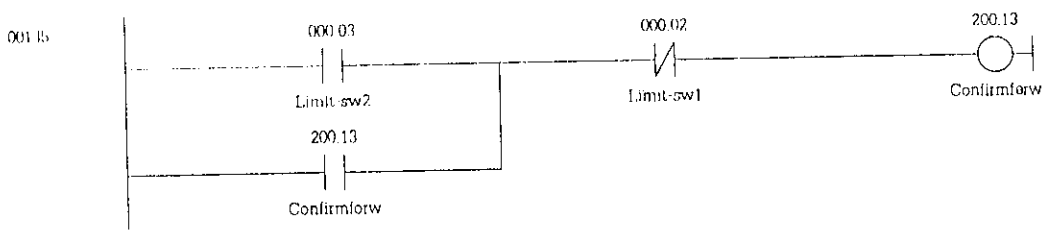


00131	LD	200.12	Detect
00132	OR	010.05	B-forw
00133	AND NOT	000.03	Limit-sw2
00134	OUT	010.05	B-forw

200.12	Detect
010.05	B-forw
000.03	Limit-sw2



Network 33 - Confirm forward



00135	LD	000.03	Limit-sw2
00136	OR	200.13	Confirmforw
00137	AND NOT	000.02	Limit-sw1
00138	OUT	200.13	Confirmforw

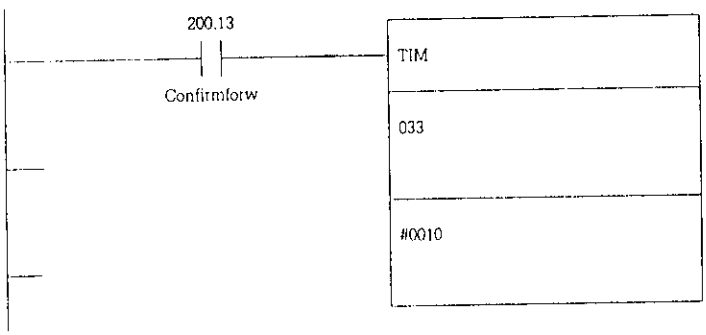
Ladder Diagram - 1:33 AGV1.SWP 3/13/98 12:28:07 Page 31

000.03 Limit-sw2
200.13 Confirmforw
000.02 Limit-sw1

000.03 Limit-sw2
1:8 1:9 1:32 1:33
200.13 Cnnfirmforw
*1:33 1:34
000.02 Limit-sw1
1:1 1:6 1:7 1:9
1:30 1:31 1:33 1:35

Network 34 - Delay

00139



00139 LD 200.13 Confirmforw
00140 TIM 033 #0010 033 #0010

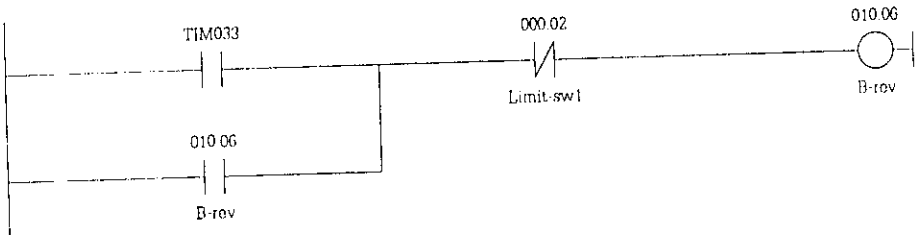
200.13 Confirmforw
TIM033

200.13 Confirmforw
*1:33 1:34
TIM033
*1:34 1:35

Ladder Diagram - 1:35 AGV1.SWP 3/13/98 12:28:07 Page 32

Network 35 - B-reverse

00141



00141	LD	TIM033	TIM033
00142	OR	010.06	B-rev
00143	AND NOT	000.02	Limit-sw1
00144	OUT	010.06	B-rev

TIM033.F

010.06 B-rev

000.02 Limit-sw1

TIM033.F

Not Used

010.06 B-rev

*1:11 *1:35

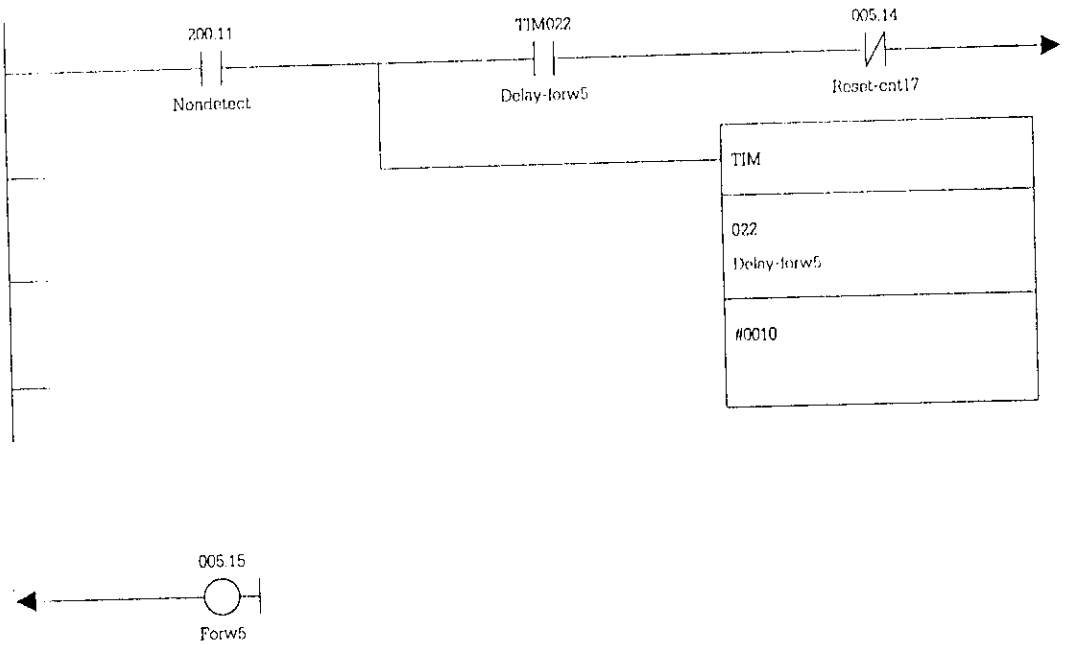
000.02 Limit-sw1

1:1 1:6 1:7 1:9

1:30 1:31 1:33 1:35

Ladder Diagram - 1:36 AGV1.SWP 3/13/98 12:28:07 Page 33

Network 36 - Forward #5



00145	LD	200.11	Nondetect
00146	OUT	TR0	TR0
00147	AND	TIM022	Delay-forw5
00148	AND NOT	005.14	Reset-cnt17
00149	OUT	005.15	Forw5
00150	LD	TR0	TR0
00151	TIM	022 #0010	Delay-forw5 #0010

200.11	Nondetect
TIM022.F	Delay-forw5

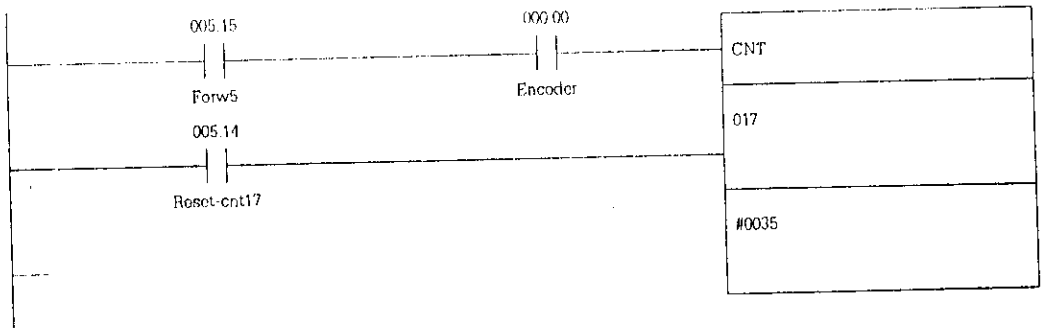
Ladder Diagram - 1:36 AGV1.SWP 3/13/98 12:28:07 Page 34

005.14 Reset-cnt17
005.15 Forw5
TIM022 Delay-forw5

200.11 Nondetect
*1:30 1:36
TIM022 F Delay-forw5
Not Used
005.14 Reset-cnt17
1:36 1:37 *1:38 1:40
005.15 Forw5
*1:36 1:37 1:39
TIM022 Delay-forw5
*1:36

Network 37 - Counter

Forward 1m (35 pulses)



00152 LD 005.15 Forw5
00153 AND 000.00 Encoder
00154 LD 005.14 Reset-cnt17
00155 CNT 017 #0035 017 #0035

005.15 Forw5
000.00 Encoder
005.14 Reset-cnt17
CNT017

005.15 Forw5
*1:36 1:37 1:39

000.00 Encodar

1:2 1:13 1:16 1:20

1:24 1:28 1:37 1:41

1:44

005.14 Reset-cnt17

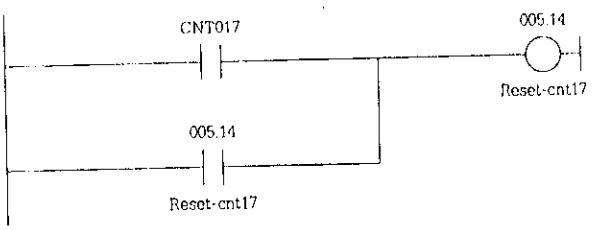
1:36 1:37 *1:38 1:40

CNT017

*1:37 1:38

Network 38 - Reset counter

00156



00156	LD	CNT017	CNT017
00157	OR	005.14	Reset-cnt17
00158	OUT	005.14	Reset-cnt17

CNT017.F

005.14 Reset-cnt17

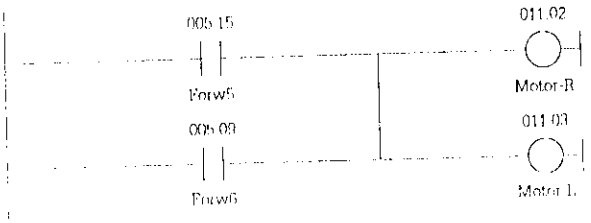
CNT017.F

Not Used

005.14 Reset-cnt17

1:36 1:37 *1:38 1:40

Network 39 : Forw #6



00159	LD	005.15	Forw5
00160	OR	005.09	Forw6
00161	OUT	011.02	Motor-R
00162	OUT	011.03	Motor-L

005.15	Forw5
005.09	Forw6
011.02	Motor-R
011.03	Motor-L

005.15 Forw5

*1:36 1:37 1:39

005.09 Forw6

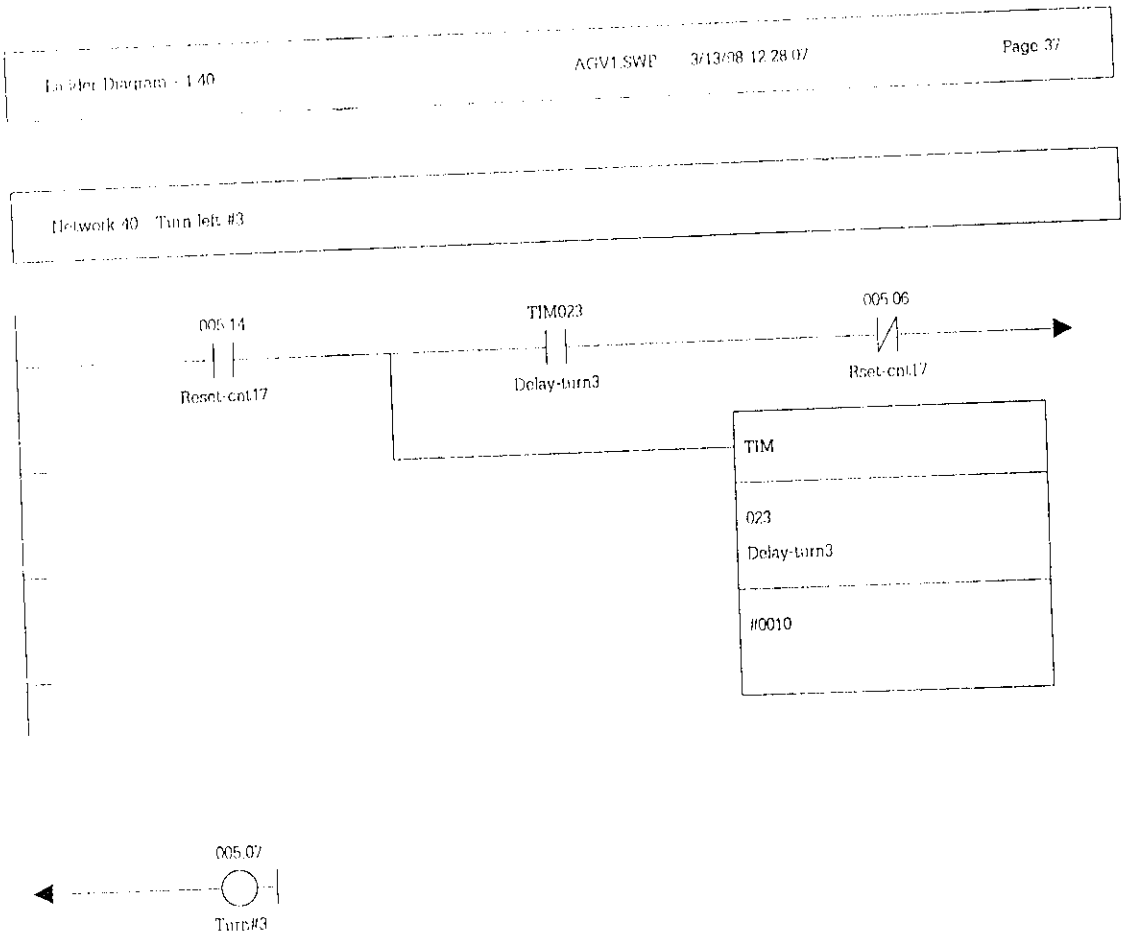
1:39 *1:43 1:44

011.02 Motor-R

*1:39

011.03 Motor-L

*1:39



00163	LD	005.14	Reset-cnt17
00164	OUT	TR0	TR0
00165	AND	TIM023	Delay-turn3
00166	AND NOT	005.06	Rset-cnt17
00167	OUT	005.07	Turn#3
00168	LD	TR0	TR0
00169	TIM	023 #0010	Delay-turn3 #0010

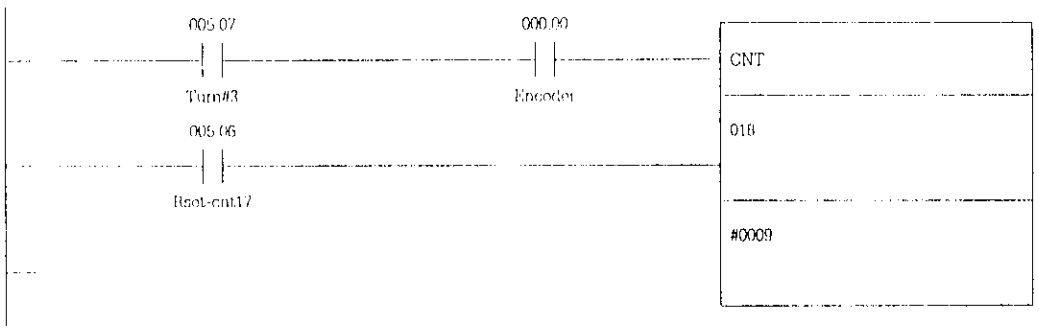
005.14	Reset-cnt17
TIM023.UP	Delay-turn3

005.06 Rset-cnt17
005.07 Turn#3
TIM023 Delay-turn3

005.14 Rset-cnt17
1:36 1:37 *1:38 1:40
TIM023 F Delay-turn3
Set Used
005.06 Rset-cnt17
1:40 1:41 *1:42 1:43
005.07 Turn#3
1:26 *1:40 1:41
TIM023 Delay-turn3
*1:40

Network 41 - Counter #18

Turn cnt 9 pulses



00170	LD	005.07	Turn#3
00171	AND	000.00	Encoder
00172	LD	005.06	Rset-cnt17
00173	CNT	018 #0009	018 #0009

005.07 Turn#3
000.00 Encoder
005.06 Rset-cnt17
CNT018

005.07 Turn#3
1:26 *1:40 1:41

0001[illegible] Encoder

1:2 1:13 1:16 1:20

1:24 1:28 1:37 1:41

1:44

005.06 Rset-cnt17

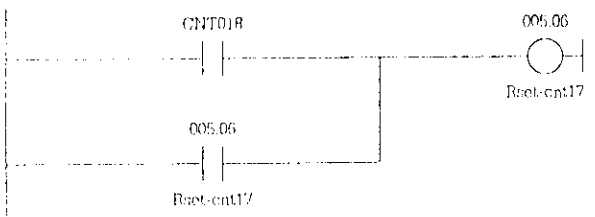
1:40 1:41 *1:42 1:43

CNT018

*1:41 1:42

Network 42 - Reset cnt18

00174



00174	LD	CNT018	CNT018
00175	OR	005.06	Rset-cnt17
00176	OUT	005.06	Rset-cnt17

CNT018 F

005.06 Rset-cnt17

CNT018 F

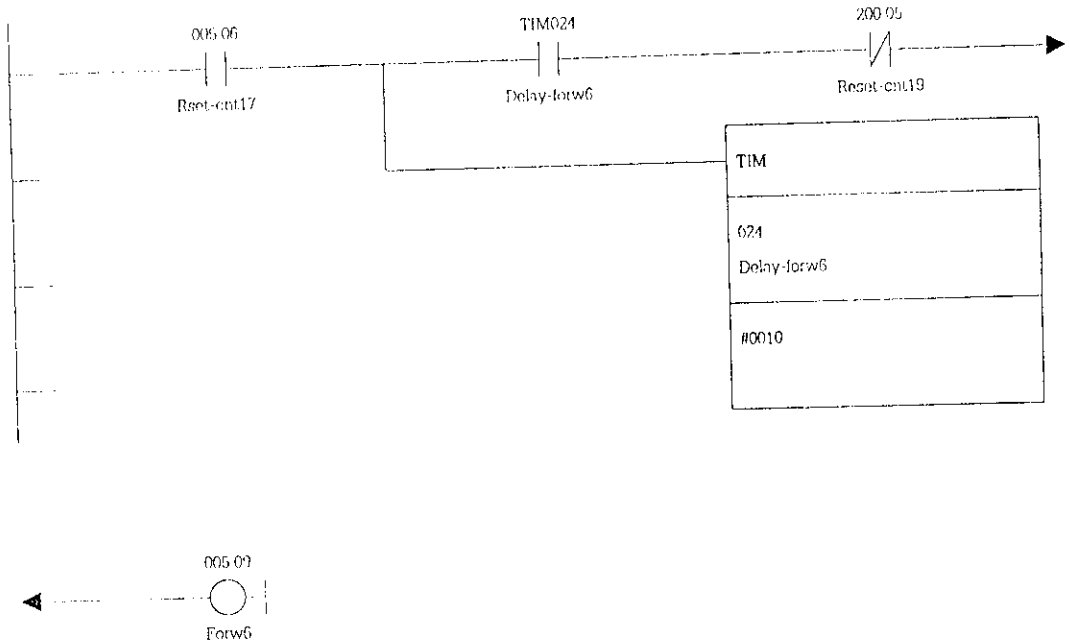
Not Used

005.06 Rset-cnt17

1:40 1:41 *1:42 1:43

Ladder Diagram - 1:43 AGV1.SWP 3/13/98 12:28:07 Page 40

Network 43 Forw #6



00177	LD	005.06	Rset-cnt17
00178	OUT	TR0	TR0
00179	AND	TIM024	Delay-forw6
00180	AND NOT	200.05	Reset-cnt19
00181	OUT	005.09	Forw6
00182	LD	TR0	TR0
00183	TIM	024 #0010	Delay-forw6 #0010

005.06	Rset-cnt17
TIM024 F	Delay-forw6

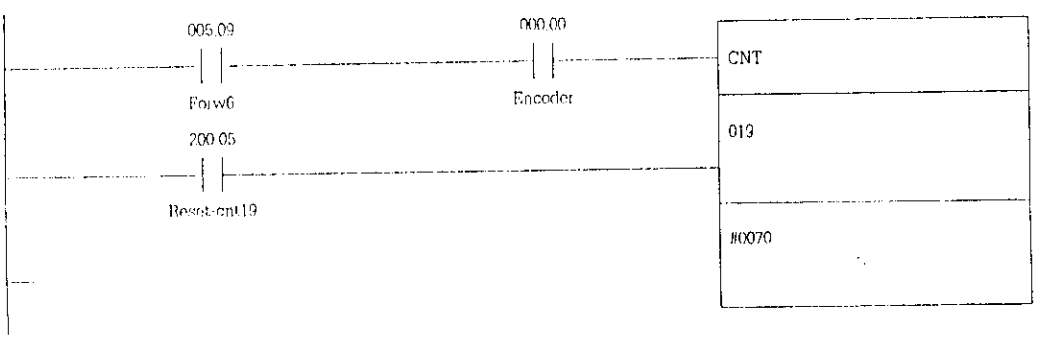
Ladder Diagram - 1:43 AGV1.SWP 3/13/98 12.28:07 Page 41

200.05 Reset-cnt19
005.09 Forw6
TIM024 Delay-forw6

005.06 Reset-cnt17
1:40 1:41 *1:42 1:43
TIM024.F Delay-forw6
Not Used
200.05 Reset-cnt19
1:43 1:44 *1:45 1:46
005.09 Forw6
1:39 *1:43 1:44
TIM024 Delay-forw6
*1:43

Network 44 - Counter #19

00184



00184 LD 005.09 Forw6
00185 AND 000.00 Encoder
00186 LD 200.05 Reset-cnt19
00187 CNT 019 #0070 019 #0070

005.09 Forw6
000.00 Encoder
200.05 Reset-cnt19
CNT019

005.09 Forw6
1:39 *1:43 1:44
000.00 Encoder
1:2 1:13 1:16 1:20

1:24 1:28 1:37 1:41

1:44

200.05 Reset-cnt19

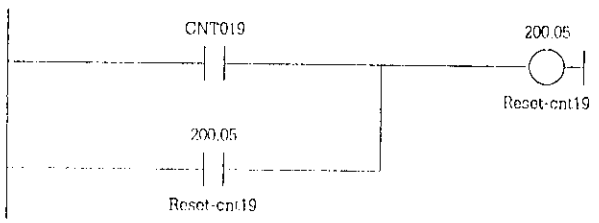
1:43 1:44 *1:45 1:46

CNT019

*1:44 1:45

Network 45 Reset cnt19

00188



00188	LD	CNT019	CNT019
00189	OR	200.05	Reset-cnt19
00190	OUT	200.05	Reset-cnt19

CNT019 F

200.05 Reset-cnt19

CNT019 F

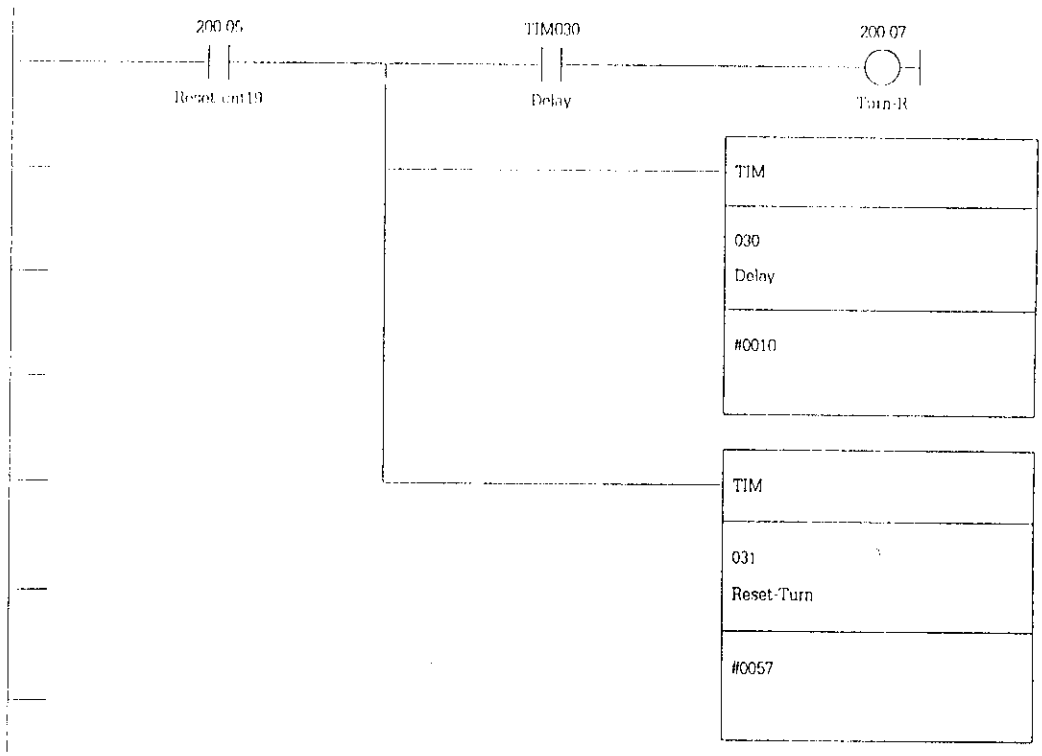
Not Used

200.05 Reset-cnt19

1:43 1:44 *1:45 1:46

Ladder Diagram - 1:46 AGV1.SWP 3/13/99 12:28:07 Page 43

Network 46 - Turn right



00191	LD	200.05	Reset-cnt19
00192	OUT	TR0	TR0
00193	AND	TIM030	Delay
00194	OUT	200.07	Turn-R
00195	LD	TR0	TR0
00196	TIM	030 #0010	Delay #0010
00197	TIM	031 #0057	Reset-Turn #0057

200.05	Reset-cnt19
TIM030.F	Delay
200.07	Turn-R
TIM030	Delay
TIM031	Reset-Turn

200.05 Reset-cnt19

1:43 1:44 1:45 1:46

TIM030.F Delay

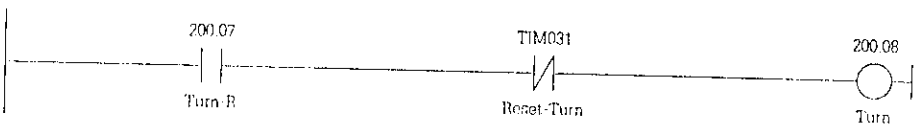
Not Used

200.07 Turn-R
*1:46 1:47
TIM030 Delay
*1:46
TIM031 Reset-Turn
*1:46 1:47

Network 47 - Turn right

Turn right (uses timer to reset turning)

00198



00198	LD	200.07	Turn-R
00199	AND NOT	TIM031	Reset-Turn
00200	OUT	200.08	Turn

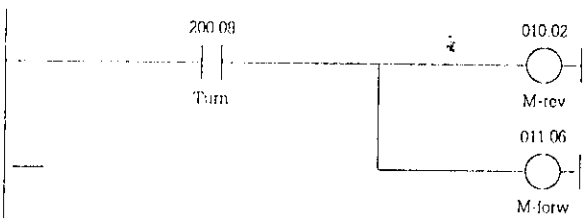
200.07 Turn-R
TIM031.F Reset-Turn
200.08 Turn

200.07 Turn-R
*1:46 1:47
TIM031.F Reset-Turn
Not Used
200.08 Turn
*1:47 1:48

Ladder Diagram - 1:48 AGV1.SWP 3/13/98 12:28:07 Page 45

Network 48 - Turn right

00201



00201	LD	200.08	Turn
00202	OUT	010.02	M-rev
00203	OUT	011.06	M-forw

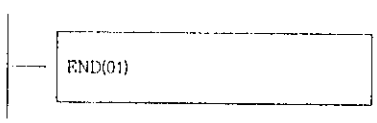
200.08		Turn
010.02		M-rev
011.06		M-forw

200.08		Turn
	*1:47	1:48
010.02		M-rev
	*1:48	
011.06		M-forw
	*1:48	

Network 49 - End of programe

Push PB_start again and main switch to stop

00204



00204 END