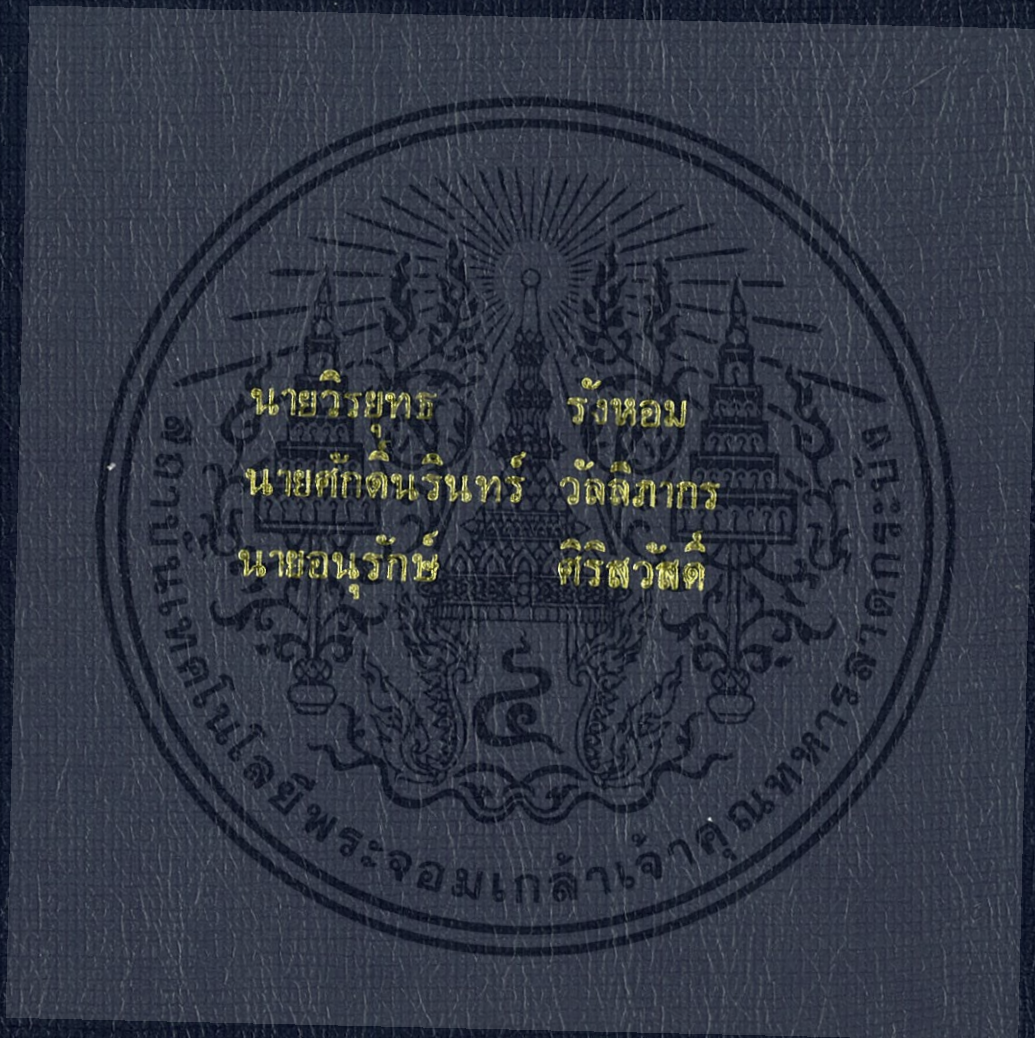


การประยุกต์ใช้โครงข่ายประสาทเทียมควบคุมยานขนส่ง
โดยใช้เครื่องควบคุมที่โปรแกรมได้

APPLICATION OF NEURAL NETWORK FOR AGV USING PLC



ปริญญาวิทยาศาสตรบัณฑิต สาขาวิศวกรรมศาสตร์
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
บการศศึกษา 2545

การประยุกต์ใช้ โครงข่ายประสาทเทียมควบคุมยานขนส่ง โดยใช้เครื่องควบคุม
ที่โปรแกรมได้

APPLICATION OF NEURAL NETWORK FOR AGV USING PLC



นายวิรัช รังหอม
นายศักดิ์นรินทร์ วัลติภากร
นายอนุรักษ์ ศิริสวัสดิ์

รฟ.
จ ๒๔๕/๑
๒๕๕๕

เลขหมู่.....

เลขทะเบียน...50249...

วัน,เดือน,ปี 28 เม.ย. 2547

b. ๐๐๑๖๖๓๕๖๗
i.

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในเพียงการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา หรือทำซ้ำโดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา 2545

APPLICATION OF NEURAL NETWORK FOR AGV USING PLC



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
DEPARTMENT OF INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING**

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
2002
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาานิพนธ์

หัวข้อปริญญาานิพนธ์ การประยุกต์ใช้ โครงข่ายประสาทเทียมควบคุมยานขนส่ง โดยใช้เครื่อง
ควบคุมที่โปรแกรมได้

APPLICATION OF NEURAL NETWORK FOR AGV USING PLC

นักศึกษาผู้จัดทำ นายวิรุทธ รังหอม รหัสประจำตัว 43015587
นายศักดิ์รินทร์ วัลติภากร รหัสประจำตัว 43015590
นายอนุรัตน์ ศิริสวัสดิ์ รหัสประจำตัว 43015599

ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2545

อาจารย์ผู้ควบคุมปริญญาานิพนธ์	ลายมือชื่อ
ผศ.ทวิพล ชือสัตย์	

วัน เดือน ปี ที่สอบ วันอังคารที่ 22 เมษายน พ.ศ. 2546

สถานที่สอบ ณ ห้องสอบปริญญาานิพนธ์ ภาควิชาวิศวกรรมการวัดคุม

ภาควิชารับรองแล้ว



(ผศ.ประสิทธิ์ จุลเสวีวงศ์)

หัวหน้าภาควิชาวิศวกรรมการวัดคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ การประยุกต์ใช้ โครงข่ายประสาทเทียมควบคุมยานขนส่ง โดยใช้เครื่อง
ควบคุมที่โปรแกรมได้

APPLICATION OF NEURAL NETWORK FOR AGV USING PLC

นักศึกษาผู้จัดทำ นายวิรุทธ รังหอม
นายศักดิ์รินทร์ วัลติภากร
นายอนรรักษ์ ศิริสวัสดิ์
อาจารย์ที่ปรึกษา ผศ.ทวีพล ชื้อสัคย์
ปีการศึกษา 2545

บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอ การประยุกต์ใช้โครงข่ายระบบประสาทเทียม สำหรับยานขนส่ง
อัตโนมัติ เพื่อการระบุตำแหน่งในการหยุด เพื่อส่งของ โดยที่ยานขนส่งอัตโนมัตินี้ จะทำการ
วิ่งไปตามพื้นที่ต่างๆ ของโรงงานตามเส้นทางที่อยู่บนพื้น และหยุดเพื่อส่งของ ณ.ตำแหน่งที่ระบุไว้
การควบคุมการเคลื่อนที่ของยานขนส่งนั้น อาศัยเครื่องควบคุมแบบครกเกที่โปรแกรมได้ ร่วมกับ
คอมพิวเตอร์

ซึ่งมีซอฟต์แวร์ที่พัฒนาขึ้นด้วย Visual Basic ตามรูปแบบของโครงข่ายประสาทเทียม
(Neural Network) โดยตำแหน่งต่างๆ จะถูกฝึกฝนโดยเก็บข้อมูลจากเซนเซอร์ที่อยู่ด้านล่างของ
ยานขนส่ง ซึ่งมีลักษณะเป็นรถยกของ(fork lift)

Thesis Title Application of Neural Network for AGV using PLC
Authors Mr. Weerayoot Ranghom
 Mr. Saknarin Wanlipagone
 Mr. Anurak Sirisawad
Thesis Advisor Asst.Prof. Taweepol Suesut
Year 2002

ABSTRACT

This thesis presents, the application of Neural Network for Automated Guide Vehicle (AGV) in order to recognize and identify the dropping position for take off or take on the goods. The AGV can be moved follow the guide line on the floor by the photo sensors and the programmable Logic Controller(PLC).

The PLC has cooperated with the computer that has the Neural Network algorithm software which developed by visual Basic.

Each of dropping position on the factory have the difference patterns are trained through the matrix photo sensors on the AGV that designed look like the fork lift.

สารบัญตาราง

ตารางที่	หน้า
2.1 ความแตกต่างระหว่างคอมพิวเตอร์ทั่วไปกับระบบประสาทเทียม.....	6
2.2 แสดงการปรับค่าถ่วงน้ำหนักสำหรับกฎการสอนแบบต่างๆ.....	14
4.1 การทดลองตัว Infarad ที่ระยะความห่างต่างๆ.....	46



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
3.5 แสดงวงจรของ sensor	43
3.6 แสดงวงจรของ sensor ที่จะนำไปต่อเข้ากับ PLC.....	43
3.7 แสดงการจัดวาง sensor ในการตรวจจับเส้นทางและแยก.....	44
3.8 โครงสร้างทางไฟฟ้าของมอเตอร์ไฟฟ้ากระแสตรงแบบอนุกรม.....	45



บทที่ 1

บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย

เมื่อเราพิจารณาถึงมนุษย์และสัตว์จะเห็นว่าสามารถรู้จักภาพและวิเคราะห์แยกแยะเสียง ได้ดีกว่าคอมพิวเตอร์ที่ทันสมัยที่สุด แม้จะเป็นคอมพิวเตอร์ที่ถูกสร้างขึ้นมาเพื่องานทางด้านนี้ โดยเฉพาะก็ตาม และนักวิจัยในสาขานี้ได้คาดหวังกันว่า ระบบโครงข่ายประสาทเทียม (Artificial Neural Network Systems : ANNS) จะเป็นคลื่นลูกใหม่ที่ใช้ในการประมวลผล ข่าวสาร สำหรับใช้แทนหรือสนับสนุนการคำนวณแบบเดิม โครงข่ายประสาทเทียมจะช่วยเสริม ความสามารถของคอมพิวเตอร์สำหรับการประมวลผลในระบบใหญ่ ๆ ซึ่งระบบนี้สามารถเรียนรู้ และตัดสินใจสำหรับกระบวนการของระบบนั้น ๆ ได้

ปัจจุบันอุตสาหกรรมประเภทต่างๆ มีการแข่งขันอย่างเสรี ทำให้ผู้ผลิตต้องมีการพัฒนา ระบบการผลิตในหลายจุด การจัดการวัสดุ เป็นจุดหนึ่งที่จะต้องพัฒนาให้ดีขึ้น เพื่อช่วยให้ต้นทุน การผลิตลดลง รวมถึงการพัฒนากระบวนการผลิตให้มีคุณภาพ และมีประสิทธิภาพสูงขึ้น ซึ่งระบบ จัดการวัสดุนั้นจะ ประกอบด้วยระบบการขนส่งวัสดุ ระบบการเก็บวัสดุและการควบคุมระบบ การพัฒนาระบบจำเป็นต้องมีการนำเอาระบบอัตโนมัติต่างๆ มาใช้ เช่น PLC , DNC , CNC , หุ่นยนต์ , สายพานลำเลียง , เทรนอัตโนมัติ , AGV เป็นต้น ซึ่ง AGV นี้จะมีการควบคุมสั่งการโดย ใช้แนวคิดของ Artificial Neural Network Systems (ANNS)

1.2 วัตถุประสงค์ของปริญญานิพนธ์

เพื่อควบคุมรถโฟร์คลิฟท์(AGV) ให้สามารถวิ่งไปตามเส้นทางที่กำหนดไว้ โดยวิ่งไปตาม แถบสีที่ติดอยู่บนพื้น เพื่อจะยกวัสดุหรือสิ่งของไปส่งยังจุดที่ต้องการได้ และตรวจสอบจุดที่ กำหนดไว้ระหว่างทางที่ตัวรถวิ่งผ่าน โดยเราจะกำหนดให้จุดที่เราต้องการตรวจสอบเป็นจุดส่งของ หรือทางแยก โดยการตรวจสอบนั้นเราจะใช้ Computer เป็นตัวตรวจสอบและตัดสินใจ ซึ่งจะใช้ ซึ่ง โปรแกรมที่ใช้ในการติดต่อกับ Computer นั้นเราใช้โปรแกรม Visual Basic ในการติดต่อกับ PLC โดยผ่านพอร์ตอนุกรม RS-232 ซึ่ง PLC นี้เป็นตัวควบคุมรถโฟร์คลิฟท์

1.3 ขอบเขตของปริญญานิพนธ์

ควบคุมรถโฟร์คลิฟท์(AGV) ให้สามารถวิ่งไปตามเส้นทางที่กำหนดไว้ โดยวิ่งไปตาม แถบสีที่ติดอยู่บนพื้น และตรวจสอบจุดที่กำหนดไว้ระหว่างทางที่ตัวหุ่นยนต์วิ่งผ่าน โดยเราจะ กำหนดให้จุดที่เราต้องการตรวจสอบเป็นจุดส่งของ หรือทางแยก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนการศึกษา

ศึกษาแนวคิดและวิธีการของ Artificial Neural Network Systems (ANNS) ออกแบบและสร้างรถที่จะใช้ในการควบคุม ศึกษาการควบคุมมอเตอร์ไฟฟ้ากระแสตรงว่าสามารถทำอะไรได้บ้าง ศึกษาการเขียนโปรแกรม Visual Basic ที่ใช้ในการติดต่อกับอุปกรณ์ภายนอกผ่านพอร์ตทอนุกรม RS-232 ซึ่งใช้ควบคุมและสั่งการ AGV ศึกษาเครื่อง PLC OMRON รุ่น (SRM1-CO2-V2) และศึกษาถึงคุณสมบัติและประสิทธิภาพของ sensor ที่ใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ความรู้เบื้องต้นเกี่ยวกับทฤษฎี และหลักการ

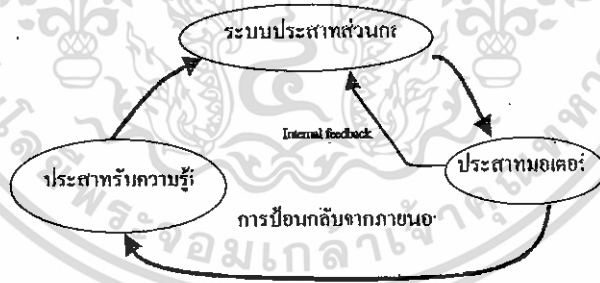
2.1 โครงข่ายประสาทเทียม (Neural Networks)

2.1.1 ระบบเซลล์ประสาททางชีววิทยา

ระบบประสาทมีหน้าที่ควบคุมรักษาสมดุลภายในร่างกายให้คงที่ ระบบประสาทของมนุษย์เป็นระบบที่มีโครงสร้างที่ซับซ้อนจะประกอบด้วยเซลล์ประสาท (Neural) ประมาณ 10 เซลล์ และเซลล์ประสาทแต่ละเซลล์จะเชื่อมต่อกับเซลล์ประสาทตัวอื่นอีกประมาณ 10 ล้าน เซลล์ ระบบประสาทมียู่ด้วยกัน 3 ส่วน แสดงในภาพที่ 2.1 คือ

1. ระบบประสาทส่วนกลาง (Central Nervous System)
2. ระบบประสาทรับความรู้สึก (Sensory Systems)
3. ระบบประสาทมอเตอร์ (Motor Systems)

ระบบประสาทส่วนกลางจะทำหน้าที่วิเคราะห์ข้อมูลต่างๆ ที่รับมาจากระบบประสาทรับความรู้สึก ที่ประกอบด้วยเซลล์ประสาทรับความรู้สึก (Sensory Receptor) ซึ่งจะรับรู้ถึงสัญญาณการเปลี่ยนแปลงของสิ่งแวดล้อมรอบตัว แล้วส่งให้ประสาทส่วนกลางทำการประมวลผล เมื่อประมวลผลเสร็จแล้ว ก็จะส่งสัญญาณไปควบคุมระบบประสาทมอเตอร์ให้ทำงาน (เป็นการแสดงผลของ อวัยวะต่างๆ)

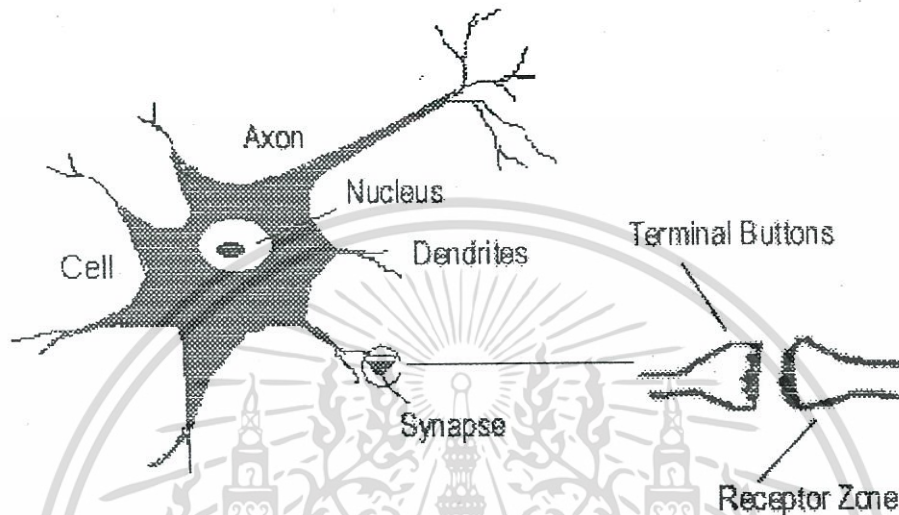


ภาพที่ 2.1 การสื่อสารข้อมูลในระบบประสาท

2.1.2 การสื่อสารข้อมูลในระบบประสาท

เซลล์ประสาทแต่ละเซลล์ประกอบด้วย ตัวเซลล์ (Cell Body) และแขนงที่แยกออกจากตัวเซลล์ คือ แอกซอน (Axon) และ เดนไดรต์ (Dendrite) คือส่วนที่รับสัญญาณจากเซลล์อื่น ดังภาพที่ 2.2 เซลล์ประสาทแต่ละเซลล์จะมีแอกซอนเพียงหนึ่งอัน ปลายของแอกซอนจะแยกเป็นแขนง หลายเส้นและ

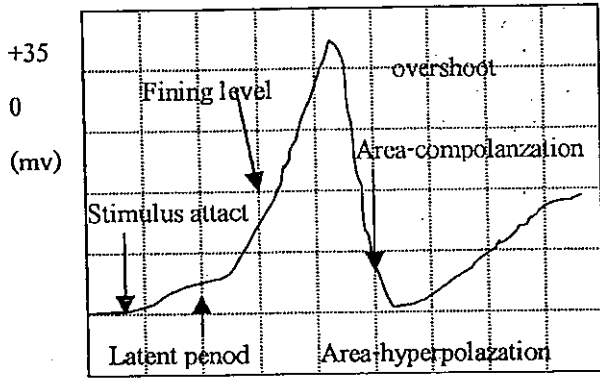
ไปสิ้นสุดที่ Terminal Buttons และจะมีตัวรับกระแสประสาทที่เรียกว่า Receptor Zone ซึ่งต่ออยู่กับ
 เคนไดรต์ของเซลล์ประสาทตัวอื่น กระแสประสาทที่ส่งจากเซลล์หนึ่งไปยังอีกเซลล์จะต้องผ่านรอยต่อ
 ระหว่าง Terminal Buttons และ Receptor Zone ซึ่งเรียกว่า ซินแนปส์ (Synapse) กระแสประสาท
 (ข้อมูลที่อยู่ในรูปของการเปลี่ยนแปลงทางไฟฟ้าเคมี) จะไปกระตุ้นให้ตัวเซลล์ส่งกระแสประสาทไปยัง
 เซลล์ประสาทตัวอื่น ซึ่งมีเคนไดรต์ทำหน้าที่รับกระแสประสาทจากเซลล์อื่นเข้าสู่ตัวเซลล์



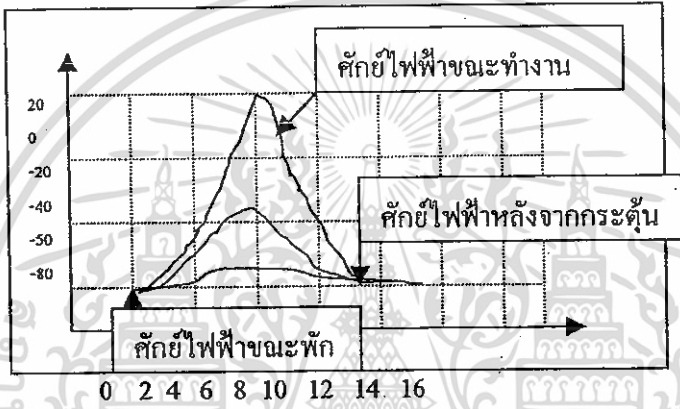
ภาพที่ 2.2 โครงสร้างของเซลล์ประสาทและการส่งสัญญาณกระแสประสาท

คุณสมบัติของเซลล์ประสาทโดยทั่วไปคือ จะมีความไวต่อการถูกกระตุ้น เมื่อเซลล์ประสาทถูก
 กระตุ้นจะทำให้เกิดการเปลี่ยนแปลงสมมูลย์ของไอออนบริเวณ Receptor Zone ซึ่งจะทำให้เกิดความ
 ต่างศักย์ขึ้น และเมื่อความต่างศักย์ของจุดนี้สูงกว่าค่าเทรชโฮลด์ก็ทำให้เกิดการส่งสัญญาณกระแส
 ประสาทไปยังเซลล์อื่น

ความต่างศักย์ของเยื่อหุ้มเซลล์ประสาทขณะพักประมาณ -60 ถึง -90 mV ถ้ามีการกระตุ้นเซลล์
 ประสาทจะมีการแก้โพลาไรซ์ (Depolarization) ของเยื่อหุ้มเซลล์ และหลังจากเกิดการแก้โพลาไรซ์จน
 ความต่างศักย์เพิ่มขึ้นเป็น 15 mV แล้ว จะทำให้อัตราการแก้โพลาเพิ่มขึ้นอย่างรวดเร็ว ซึ่งจุดนี้จะเรียก
 ว่า Firing Level แสดงดังรูปที่ 2.3 เมื่อความต่างศักย์เพิ่มขึ้นจนถึงจุดสูงสุด (35 mV) จะทำให้
 อัตราการแก้โพลาไรซ์ลดลงอย่างรวดเร็ว (70%) และจะค่อย ๆ ลดลงกลับเข้าสู่ภาวะปกติ หลังจาก
 เข้าสู่ภาวะปกติแล้วความต่างศักย์จะลดลงต่ำกว่าระดับปกติซึ่งจะเรียกช่วงนี้ว่า After Hyperpolarization
 ลำดับของการเปลี่ยนแปลงความต่างศักย์ทั้งหมดนี้เรียกว่า ศักย์ไฟฟ้าขณะทำงาน (After Potential)



ภาพที่ 2.3 แสดงลักษณะการเกิดศักย์ไฟฟ้าขณะทำงาน



ภาพที่ 2.4 แสดงศักย์ไฟฟ้าขณะทำงานที่เกิดจากการรวมกันของซินแนปส์

2.1.3 โครงข่ายประสาทเทียมคืออะไร

อันดับแรก เมื่อเราพูดถึง โครงข่ายประสาทเทียมเรามักจะพูดถึง ANNs (Artificial Neural Network System) เพราะว่าเราจะเข้าใจได้ง่ายกว่า โครงข่ายประสาทเทียมทางชีววิทยาจะมีความสลับซับซ้อนมากกว่าที่อยู่ในรูปทางคณิตศาสตร์ รูปแบบ ANNs ที่อยู่ใน Model ทางคณิตศาสตร์ ANNs คือ การประมวลผลทางข้อมูลที่มีรูปแบบเดียวกันกับระบบเชิงชีววิทยา เช่น สมอง หัวใจ หลักของ ขบวนการ คือ ระบบการประมวลผลของข้อมูลรูปแบบใหม่ ซึ่งจำลองการประมวลผลของข้อมูลอย่างต่อเนื่อง ร่วมกับของข้อมูลจำนวนมากที่ใช้งานในการแก้ไขปัญหเฉพาะอย่าง ANNs เหมือนกับคนคือ จะต้องมีการเรียนรู้ เพื่อให้เข้าใจง่ายขึ้น โครงข่ายระบบประสาทเทียม ก็คือ โปรแกรมคอมพิวเตอร์ที่เลียนแบบเครือข่ายระบบประสาทของมนุษย์ เพื่อประมวลผล ระบบเกี่ยวกับประสาทหรือระบบประสาทจำลองไม่เหมือนกับโปรแกรม โปรแกรมใช้การมีเหตุผล ที่จะใช้กฎเกณฑ์ที่รู้กับสถานการณ์ต่างๆ สถานการณ์ใหม่ๆแต่ละอันอาจจะต้องการกฎเกณฑ์อื่นๆเพิ่มเข้าไปในโปรแกรม สำหรับสถาน

การันตีใหม่แต่ละอัน ระบบประสาทปลอมจะถูกฝึกฝนจากประสบการณ์ หลังจากระบบประสาทจำลอง ถูกฝึกฝนจากสถานการณ์ที่กำหนดแล้ว มันสามารถทำงานตามสถานการณ์ที่กำหนดขึ้นได้

ตารางที่ 2.1 ความแตกต่างระหว่างคอมพิวเตอร์ทั่วไปกับระบบประสาทเทียม

คอมพิวเตอร์ทั่วไป	เครือข่ายระบบประสาท
การใช้เหตุผลพิจารณาจากหลักทั่วไปต่อ input และ output	การใช้เหตุผลเฉพาะ ต่อ input และ output โดยที่เราสร้างกฎเกณฑ์ขึ้น.
มีศูนย์กลางการคำนวณที่เดียว, เกิดขึ้นพร้อมกัน, และเป็นลำดับ.	มีศูนย์กลางการคำนวณร่วมกัน, เกิดขึ้นพร้อมๆกัน
หน่วยความจำเป็นชุดๆ, เก็บความหมาย, และตำแหน่งที่อยู่	หน่วยความจำถูกแจกจ่าย, และตำแหน่งที่อยู่จะอยู่ภายใน
ไม่ใช่ความผิดพลาด ที่มีลักษณะกว้าง และไม่ใช่ความผิดพลาดของงาน.	เป็นความผิดพลาด ที่มีลักษณะกว้าง, มากเกินไป, และการแซร์ความรับผิดชอบ
ความเร็ว ได้จากการวัดในหนึ่งล้านของวินาที	ความช้า ได้จากการวัดในหนึ่งพันของวินาที
ถูกต้องกับความจริง	ไม่ถูกต้องกับความจริง
สิ่งที่เกี่ยวข้องกับการหยุดนิ่งไม่เคยเคลื่อนที่	สิ่งที่เกี่ยวข้องกับการเคลื่อนที่
ใช้ได้ถ้ามีกฎที่จำกัดและมีข้อมูลที่แน่นอน	ทำงานได้แม้ไม่รู้กฎ, กฎซับซ้อน, หรือแค่บอกข้อมูลบางส่วน.

2.1.4 นิยามอื่นๆที่เกี่ยวกับระบบโครงข่ายประสาทเทียม

ANS (Artificial neural system) หรือ Neural networks มีชื่อเรียกอื่นๆ อีก เช่น

- Parallel distributed processing models
- Connectivist / Connectionism models
- Adaptive system
- Self-Organizing systems
- Neurocomputing
- Neuromorphic system

การศึกษาเกี่ยวกับโครงข่ายประสาทเทียมของ DARPA (1988, AFCEA International Press, P.60) ซึ่งกล่าวไว้ว่า โครงข่ายประสาทเทียมคือระบบที่มีการรวมกันแบบง่าย ๆ ของส่วนประมวลผลย่อยๆ หลายๆ ตัวทำงานขนานกัน ซึ่งถูกกำหนดหน้าที่โดยโครงสร้างของเครือข่าย ความเหนียวแน่นของการเชื่อมต่อและจุดที่ทำหน้าที่ในการคำนวณและประมวลผล

Haykin, s (1994), โครงข่ายประสาทเทียม คือการประมวลผลแบบกระจายรวม ซึ่งมีคุณสมบัติในการจดจำประสบการณ์ หรือความรู้ที่ได้แล้วนำมาเก็บไว้เพื่อเรียกใช้จากครั้งต่อไป เหมือนกับใน 2 หัวข้อดังนี้

1. Knowledge จะถูกสะสมจากการเรียนรู้โดยระบบโครงข่าย
2. Interneuron การเชื่อมต่อกันภายในรู้จักนำหน้าที่ใช้ในการเก็บข้อมูล

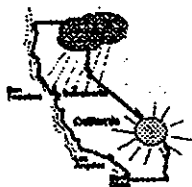
ANNs ได้ถูกใช้งานอย่างกว้างขวางในปัจจุบันซึ่งมีประโยชน์มากมาย เช่น การแก้ไขปัญหาที่มีความสลับซับซ้อนต่อองศาการวิเคราะห์แนวโน้มของข้อมูล

2.1.5 ทำไมจึงใช้ระบบโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมมีความสามารถที่จะแก้ไขปัญหาที่มีความสลับซับซ้อน มีความสามารถในการจำแนกรูปแบบและตรวจสอบแนวโน้มที่มีความสลับซับซ้อนซึ่งทำได้ยาก เช่น



สามารถแยกแยะลายมือ และจำนวนตัวเลขบนเช็คได้



สามารถแยกแยะสภาพอากาศ หรือพยากรณ์อากาศได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สามารถจดจำ และแยกแยะใบหน้าของคนได้



สามารถเข้าใจ ,จดจำ และแยกแยะ เสียง และรูปภาพได้

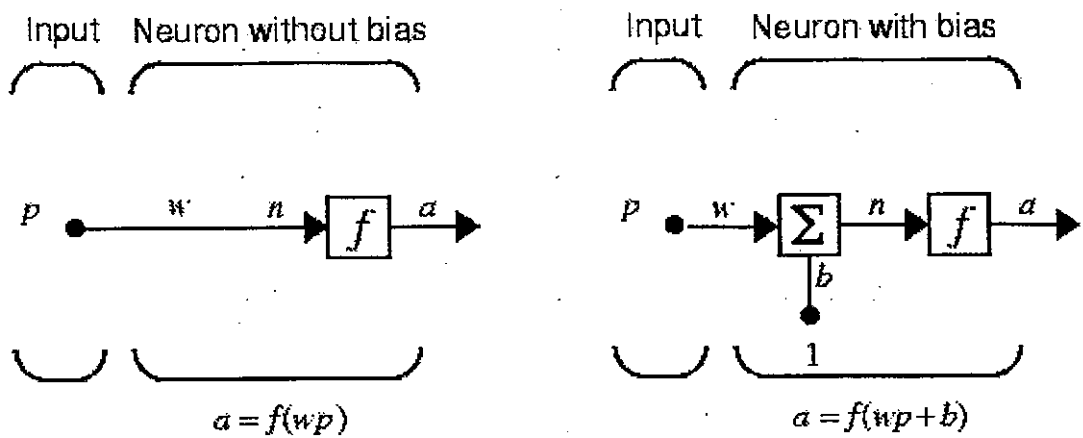
โครงข่ายประสาทเทียมที่ได้รับการฝึกฝนสามารถเป็นผู้เชี่ยวชาญในข้อมูลนั้นๆ ซึ่งทำการวิเคราะห์อยู่ ความเชี่ยวชาญนี้สามารถที่จะใช้ในการบ่งชี้ ความเหมาะสมของคำตอบของคำถามว่าจะมีอะไรเกิดขึ้น

- Adaptive Learning : ความสามารถที่จะเรียนรู้ได้ว่าจะทำอะไรกับข้อมูลหรืองานโดยเริ่มต้นบนพื้นฐานของข้อมูลที่ได้รับการฝึกฝนมาหรือเกิดจากประสบการณ์ที่ได้พบมา
- Self- Organisation : ANN สามารถสร้างองค์กร หรือตัวแทนของข้อมูลที่เป็นของตัวเองได้ซึ่งจะรับไว้ขณะเกิดการเรียนรู้
- Real Time Operation : กระบวนการคำนวณ ANN อาจจะดำเนินการแบบขนานกัน และ Hardware หรืออุปกรณ์พิเศษที่ได้ถูกออกแบบและสร้างขึ้นที่จะให้ประโยชน์ในเรื่องการคำนวณ
- Fault Tolerance via Redundant information Coding: คือการแบ่งออกเป็นสัดส่วนของโครงข่ายประสาทเทียม ซึ่งจะนำไปสู่การตอบสนองทางด้านประสิทธิภาพอย่างไรก็ตามความสามารถของโครงข่ายบางตัวอาจสร้างขึ้นใหม่ได้เมื่อเกิดการเสียหายขึ้น

2.1.6 แบบจำลองโครงข่ายเซลล์ประสาท

ในหัวข้อที่แล้วเราได้กล่าวถึงการทำงานของระบบประสาทโดยทั่วไป ของสัตว์เลี้ยงลูกด้วยนม ในหัวข้อนี้จะมาพิจารณากันถึงระบบการทำงานของแบบจำลองระบบประสาทที่จะใช้ในการประมวลผลโดยเครื่องคอมพิวเตอร์เพื่อนำไปใช้ควบคุม (รักษาสมดุล) ระบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.5 แบบจำลองพื้นฐานโครงข่ายประสาทเทียม แบบไม่มีการถ่วงน้ำหนัก และมีการถ่วงน้ำหนัก

input p ถูกส่งไปคูณกับน้ำหนัก w , จากการคูณ w กับ p wp คือเหตุผลเท่านั้นของการย้ายฟังก์ชัน f เซลล์ประสาทที่เหมาะสมควรมี b ที่เป็นตัวถ่วง ฟังก์ชันการโยกย้าย input n คือการรวม wp และการถ่วง b คือการรวมเหตุผลของฟังก์ชันการโยกย้าย f ตอนนี้ f คือฟังก์ชันการโยกย้าย สิ่งที่ทำให้เหตุผล n และ ตัวอย่างต่างๆของการโยกย้ายฟังก์ชันในส่วนถัดไป บันทึกว่า w และ b ทั้งสองปรับตัวแปรของเซลล์ประสาท ตัวแปรสามารถถูกปรับเพื่อให้เครือข่ายมีพฤติกรรมต่างๆ ดังเช่น เราสามารถฝึกฝนเครือข่ายให้ทำงานที่เจาะจงโดยการปรับน้ำหนักหรือการถ่วง, หรือบางทีตัวเครือข่ายเองจะปรับตัวแปรเหล่านี้เพื่อให้ได้ผลที่ต้องการ

เซลล์ประสาทมีข้อกำหนดสำหรับการถ่วง, และการถ่วงถูกใช้ในจำนวนมากมายและในจำนวนมากจะสมมุติขึ้น อย่างไรก็ตาม, อาจจะไม่มีการถ่วงในเซลล์ประสาทก็ได้ถ้าต้องการ

แบบจำลองที่จะกล่าวมาเป็นแบบแรกในที่นี้ถูกเสนอโดย McCulloch และ pitts ในปี ค.ศ. 1943 ซึ่งระบบของเซลล์ประสาทแสดงในภาพที่ 2.6 (ก) อินพุต x (สำหรับ $I = 1, 2, 3, \dots, n$) จะมีค่าเป็น $\{0,1\}$ ซึ่งจะขึ้นอยู่กับสัญญาณอินพุตสัญญาณจากเซลล์อื่นในขณะนั้นว่าจะมีหรือไม่มีสัญญาณ ส่วนสัญญาณที่จะส่งต่อไปยังเซลล์ถัดไป ซึ่งเป็นเซลล์แสดงผล (เซลล์ของผลลัพธ์จะแทนด้วย O) และ Firing Level ของแบบจำลองนี้ถูกจำลองโดย

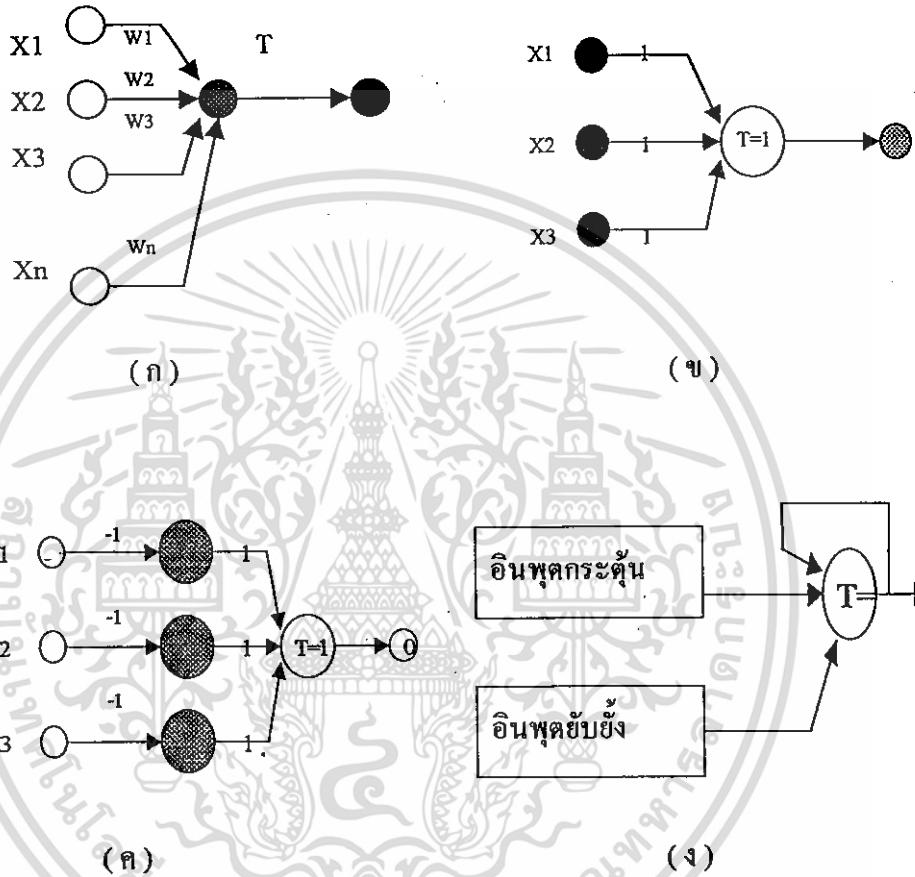
$$O^{k+1} = \begin{cases} 1: & \text{ถ้า } \sum_{i=1}^n w_i x_i^k \geq T \\ 0: & \sum_{i=1}^n w_i x_i^k < T \end{cases} \quad (2.1)$$

โดยที่ $k = 0, 1, 2, \dots$ เป็นช่วงเวลาที่ต่อเนื่อง w เป็นค่าถ่วงน้ำหนักที่เชื่อมต่อกับอินพุตที่

I ซึ่งถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$W = +1$ แสดงถึงสถานะการกระตุ้นของซินแนปส์ และถ้า $W = -1$ ซินแนปส์จะมีการยับยั้งการส่งผ่านสัญญาณและ T เป็นค่าความต่างศักย์เทรชโฮลด์ หรือขีดเริ่มเปลี่ยน ซึ่งถ้าค่าผลรวมของผลคูณระหว่างค่าถ่วงน้ำหนักกับสัญญาณอินพุตจะต้องมากกว่า T จึงจะมีสัญญาณผ่านไปยังเซลล์อื่นได้

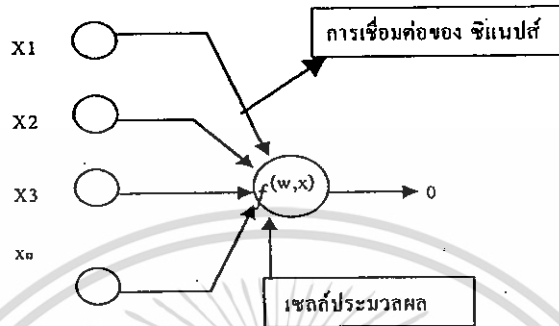


ภาพที่ 2.6 แบบจำลองเซลล์ประสาทของ McCulloch-Pitts และโครงข่ายลอจิกพื้นฐาน (ก) โดอะแกรมแบบจำลอง (ข) เกต NOR (ค) เกต NAND (ง) เซลล์หน่วยความทรงจำ

แบบจำลองของ McCulloch-Pitts แม้จะเป็นโครงข่ายแบบง่าย ๆ แต่แบบจำลองนี้สามารถใช้แทนตัวดำเนินการทางลอจิกได้ คือตัวดำเนินการ NOT, OR, และ AND เมื่อเรากำหนดค่าถ่วงน้ำหนักและค่าเทรชโฮลด์ที่เหมาะสมกับโครงข่าย ในภาพที่ 2.6 (ข) หรือ 2.6 (ค) เป็นแบบจำลองที่ใช้ดำเนินซึ่งเป็น Gates ที่ใช้ในพีชคณิตบูลีนคือ NOR หรือ NAND ซึ่งมีอินพุตสามตัว และในภาพที่ 2.6 (ง) เป็นเซลล์ที่แทนหน่วยความทรงจำซึ่งเป็นโครงข่ายที่มีการป้อนกลับจากเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ANN ที่จะกล่าวถึงต่อไปนี้เป็นแบบจำลองที่ใช้เป็นต้นแบบ ในการศึกษาแบบจำลอง และระบบการเรียนรู้ชนิดต่าง ๆ ที่กล่าวต่อไปซึ่งโครงข่ายนี้จะคล้ายกับแบบจำลองของ McCulloch-Pitts ส่วนที่ต่างกันคือค่าตัวแปรต่างๆ ที่ใช้ในแบบจำลอง เนื่องจากค่าตัวแปรในแบบจำลอง ANN เป็นเลขจำนวนจริงและค่าถ่วงน้ำหนักที่ได้จากการเรียนรู้ซึ่งแบบจำลองนี้แสดงในภาพที่ 2.7



ภาพที่ 2.7 แบบจำลองโครงข่ายเซลล์ประสาทเทียม

จากภาพที่ 2.7 แสดงโครงข่ายการเชื่อมต่อของแบบจำลองเซลล์ประสาทที่สามารถสอนให้โครงข่ายตัดสินใจได้โดยมี X เป็นสัญญาณอินพุต และ W เป็นค่าถ่วงน้ำหนักที่ได้จากการสอนโครงข่าย และแต่ละโหนดในโครงข่ายจะใช้แทนเซลล์ประสาทแต่ละเซลล์ ซึ่งบางครั้งจะเรียกว่าหน่วยประมวลผลพื้นฐาน (Process Element Unit) และมีซิแนปส์ซึ่งจะเชื่อมต่อโหนดเพื่อใช้ในการส่งผ่านสัญญาณ การกระตุ้นหรือยับยั้งสัญญาณจะขึ้นอยู่กับค่าถ่วงน้ำหนัก W และสำหรับสัญญาณเอาต์พุตสามารถคำนวณได้ดังนี้

$$o = f(W \cdot X) \quad (2.2)$$

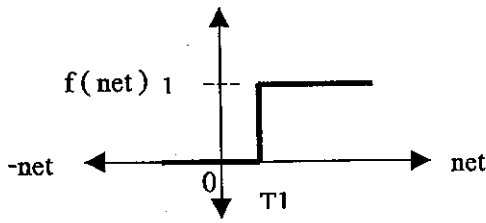
โดยที่ W เป็นเวกเตอร์ของค่าถ่วงน้ำหนักซึ่งสามารถกำหนดได้ดังนี้

$$W = [w_1, w_2, w_3, \dots, w_n] \quad (2.3)$$

และ X เป็นเวกเตอร์อินพุต $X = [x_1, x_2, x_3, \dots, x_n]$ เมื่อ t เป็นตัวดำเนินการทรานสโพสค์ของเมตริกซ์ฟังก์ชันกำหนดสัญญาณเอาต์พุตในสมการที่ (2.2) ถูกเรียกว่าฟังก์ชันการเร่งเร็ว หรือ แอกติเวชันฟังก์ชัน (Activation Function) [5] ซึ่งมีคุณสมบัติคล้ายกับกราฟของศักย์ไฟฟ้าขณะทำงานในภาพที่ 2.8 Activation Function มีด้วยกันสองชนิดคือ ชนิดที่เป็นเชิงเส้นและชนิดไม่เป็นเชิงเส้นดังแสดงในภาพที่ 2.8

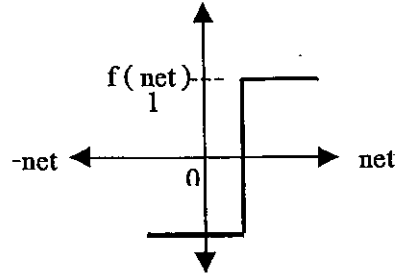
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดให้ $net = W'x = \sum_{i=1}^n w_i x_i$ และ T_1, T_2 เป็นค่าทรชโฮลด์



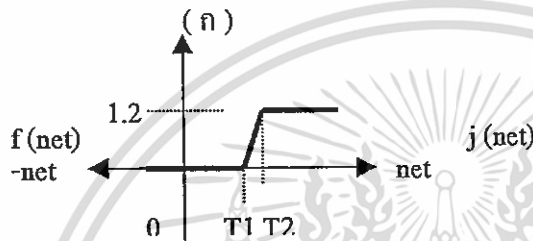
0; ถ้า $net < T_1$

$$f(net) = \begin{cases} 0; & \text{ถ้า } net < T_1 \\ 1; & \text{ถ้า } net \geq T_1 \end{cases}$$



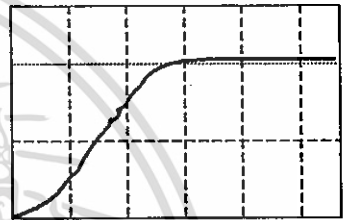
0; ถ้า $net < T_1$

$$f(net) = \begin{cases} -1; & \text{ถ้า } net < T_1 \\ 1; & \text{ถ้า } net \geq T_1 \end{cases}$$



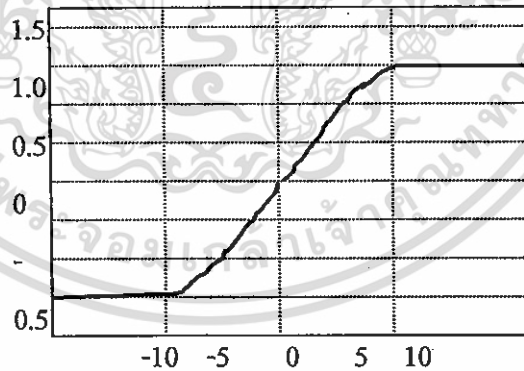
$$f(net) = \begin{cases} 0; & \text{ถ้า } net < T_1 \\ \frac{1}{T_2 - T_1}(T_2 - net); & \text{ถ้า } T_1 \leq net < T_2 \\ 1; & \text{ถ้า } net \geq T_2 \end{cases}$$

(ค)



$$f(net) = \frac{1}{1 + e^{-net}}$$

(ง)



$$f(net) = \frac{2}{1 + e^{-net}} - 1$$

(จ)

ภาพที่ 2.8 Activation Function แบบชนิดต่างๆ (ก) Threshold Logic (ข) Bipolar Binary

(ค) Linear Threshold (ง) Sigmoid (จ) Bipolar Continuous

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ Activation Function ในภาพที่ 2.8 ที่เป็นแบบไม่ต่อเนื่อง ภาพที่ 2.8 (ก-ค) และแบบต่อเนื่อง ภาพที่ 2.8 (ง-จ) ค่าที่ได้จากฟังก์ชันเหล่านี้มีสองลักษณะ [5] [6] คือ สองสภาพขั้วหรือ ไบโพลาร์ (Bipolar) ซึ่งโดเมนของ $f(\text{net})$ จะอยู่ในช่วงปิด $[-1,1]$ และหนึ่งสภาพขั้วหรือ ยูนิโพลาร์ (Unipolar) ซึ่งโดเมนของ $f(\text{net})$ จะอยู่ในช่วงปิด $[0,1]$ การที่จะกำหนดว่า Activation Function ใดคุณสมบัติที่สุดคงเป็นไปได้ เนื่องจาก ANNs ไม่มีโครงข่ายที่แน่นอน นั่นคือโครงข่ายหนึ่งจะใช้ได้กับปัญหาใดปัญหาหนึ่งเท่านั้น ดังนั้นการเลือก Activation Function ใดจะต้องพิจารณาให้เหมาะสมกับปัญหานั้น ๆ

2.1.7 การเรียนรู้ของโครงข่าย

การสอนโครงข่าย คือการหาค่าถ่วงน้ำหนักที่เหมาะสมให้โครงข่ายนั้นๆ ซึ่งประสิทธิภาพในการสอนจะขึ้นอยู่กับค่าถ่วงน้ำหนักที่หาได้

วิธีการสอนมี 2 แบบ

- การสอนแบบแนะนำ (Supervised Learning)

การสอนด้วยวิธีนี้จะกำหนด input และ output ที่ต้องการ เมื่อป้อน input ให้โครงข่าย โครงข่ายจะทำการประมวลผล ผลที่ได้คือ คำตอบ และค่าถ่วงน้ำหนัก แล้วนำผลที่ได้มาคำนวณหาค่าความผิดพลาดห่างจากคำตอบที่ต้องการมาน้อยเพียงใด ถ้ายังมีค่าความผิดพลาดห่างจากคำตอบที่ต้องการอยู่ ก็ต้องทำการสอนต่อไปจนกว่าจะมีค่าใกล้เคียงกับคำตอบที่ต้องการ และค่าถ่วงน้ำหนักที่ได้จากการสอนนี้จะเปรียบเหมือนฟังก์ชันที่ใช้ในการประมวลผล

- การสอนแบบไม่มีการแนะนำ (Unsupervised Learning)

การสอนด้วยวิธีนี้ภายในโครงข่ายจะมี output อยู่หลายโหนด โดยแต่ละโหนดจะแทนกลุ่มของข้อมูลที่มีคุณสมบัติเหมือนกัน เมื่อป้อน input เข้าโครงข่าย โครงข่ายจะทำการคำนวณโดยใช้ค่าถ่วงน้ำหนักเป็นตัวแยกความแตกต่างของ input ไปเก็บไว้ในโหนด output ของโครงข่าย ซึ่งไม่สามารถระบุได้ว่า output โหนดใดเป็นของข้อมูลกลุ่มใด ต้องทำการกำหนดเอง ซึ่งต่างจากการสอนซึ่งนำที่โครงข่ายสามารถระบุกลุ่มของเอาต์พุตได้แน่นอน

การสอนโครงข่ายเซลล์ประสาทจำลอง เป็นการหาฟังก์ชันการแปลง และฟังก์ชันการแปลงที่ได้จะมีคุณสมบัติไม่เป็นเชิงเส้น ซึ่งฟังก์ชันการแปลงของ ANN ในที่นี้คือเซตของค่าความถ่วงน้ำหนักของโครงข่าย ดังนั้นฟังก์ชันการแปลงจะมีศักยภาพมาน้อยเพียงใดนั้นจะขึ้นอยู่กับค่าถ่วงน้ำหนักของโครงข่ายนั้นๆ ว่ามีเสถียรภาพมาน้อยเพียงใดนั้นขึ้นอยู่กับค่าถ่วงน้ำหนักของโครงข่ายนั้น ๆ ว่ามีเสถียรภาพมาน้อยเพียงใด และค่าถ่วงน้ำหนักคำนวณได้จากการสอนโครงข่าย ซึ่งการสอนโครงข่ายมีหลายแบบด้วยกันเช่น กฎการสอนของ Hebb, กฎการสอนแบบ การสอนโดยใช้สหสัมพันธ์, กฎการสอนแบบ Winner - Take - All, และกฎการสอนแบบ Outstar ของ Grossberg ในตารางที่ 2 ได้สรุปกฎการสอนโครงข่ายของวิธีต่าง ๆ ไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 แสดงการปรับค่าถ่วงน้ำหนักสำหรับกฎการสอนแบบต่างๆ

กฎการสอน	การปรับค่าถ่วงน้ำหนัก	ค่าเริ่มต้นของ ค่าถ่วงน้ำหนัก	การ สอน	Activation Function
Hebb	$\Delta w_{ij} = cf(W_i, X)x_j$	0	U	Any
Preceptron	$\Delta w_{ij} = c(d_i - f(W_i, X))x_j$	จำนวนจริงใด ๆ	S	Binary Bipolar of Binary Unipolar
Delta	$\Delta x_i = c(d_i - o_i)f(W_i, X)X$	จำนวนจริงใด ๆ	S	ฟังก์ชันต่อเนื่อง
Widrow- Hoff	$\Delta w_{ij} = c(d_i - (W_i, X))x_j$	จำนวนจริงใด ๆ	S	Any
Correlation	$\Delta x_{ij} = cd_i x_j$	0	S	Any
Winner- Take-All	$\Delta w_{mj} = c(x_j - x_{mj})$ m : โหนดที่เป็น Winner	Random Normalized	U	ฟังก์ชันต่อเนื่อง
Outstat	$\Delta w_{ij} = c(d_i - w_{ij})$	0	S	ฟังก์ชันต่อเนื่อง

c คือเป็นค่าคงที่ของอัตราการเรียนรู้

s คือการสอนแบบแนะนำ

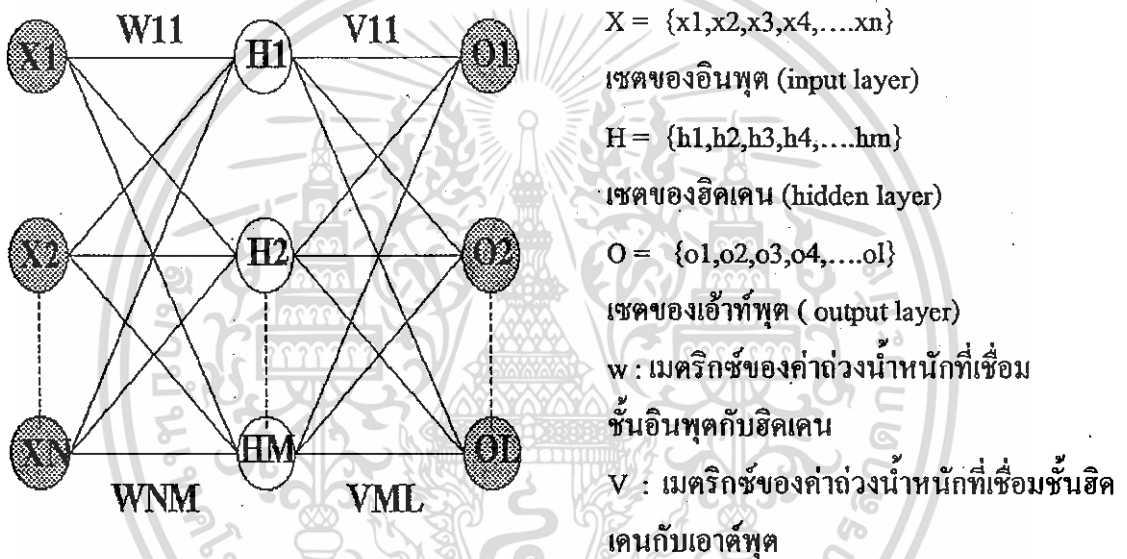
u คือการสอนแบบไม่แนะนำ

Δw_{ij} คือค่าถ่วงน้ำหนักที่ถูกปรับค่าและมีการเชื่อมต่อระหว่างอินพุตโหนดที่ i และเอาต์พุตโหนดที่ j

2.1.8 การแพร่กระจายกลับ

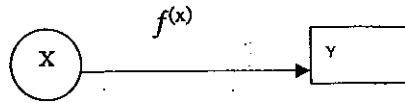
การแพร่กระจายกลับหรือแบคโพรพาเกชัน (Back-Propagation) เป็นขั้นตอนที่ใช้สอนโครงข่ายแบบ Multilayer Perceptron ซึ่งเป็นแบบจำลองโครงข่ายเซลล์ประสาทที่มีการเชื่อมโยงกันเป็นโครงข่ายแบบเป็นชั้น ๆ ดังในภาพที่ 2.9 ซึ่งโครงข่ายนี้มีการเชื่อมโยงกัน 3 ชั้น ซึ่งประกอบด้วยชั้นของอินพุตซึ่งมีเซลล์ประสาทอยู่ N โหนด ถัดมาเป็นชั้นของฮิดเดนหรือชั้นภายใน (Hidden Layer) ซึ่งประกอบด้วยโหนดต่าง ๆ จำนวน M โหนด และสุดท้ายคือชั้นของเอาต์พุตซึ่งมีโหนดต่าง ๆ อยู่ L โหนด โครงข่ายแบบ Multilayer Perceptron ในภาพที่ 2.9 แต่ละโหนดในชั้นเดียวกัน จะไม่มีการเชื่อมโยงกัน จะมีเฉพาะระหว่างชั้นเท่านั้น และการเชื่อมเหมือนในภาพที่ 2.9 อาจจะมีจำนวนชั้นมากกว่านี้ก็ได้ซึ่งอาจมีสี่ชั้น โดยการเพิ่มชั้นฮิดเดนเข้าไปอีกหนึ่งชั้น หรือถ้าต้องการจำนวนชั้นมากกว่านี้ก็สามารถทำได้โดยการเพิ่มชั้นของฮิดเดน

ซึ่งโครงข่ายนี้มีการเชื่อมโยงกัน 3 ชั้น ซึ่งประกอบด้วยชั้นของอินพุตซึ่งมีเซลล์ประสาทอยู่ N โหนด ถัดมาเป็นชั้นของฮิดเดนหรือชั้นภายใน (Hidden Layer) ซึ่งประกอบด้วยโหนดต่าง ๆ จำนวน M โหนด และสุดท้ายคือชั้นของเอาต์พุตซึ่งมีโหนดต่าง ๆ อยู่ L โหนด โครงข่ายแบบในภาพที่ 2.9 แต่ละโหนดในชั้นเดียวกันจะไม่มีมีการเชื่อมโยงกันจะมีเฉพาะระหว่างชั้นเท่านั้น และการเชื่อมโยงนี้จะต่อถึงกันทุกโหนด โครงข่ายแบบ Multilayer Perceptron ไม่จำเป็นต้องมีสามชั้นเหมือนในภาพที่ 2.9 อาจจะมีจำนวนชั้นมากกว่านี้ก็ได้ซึ่งอาจมีสี่ชั้น โดยการเพิ่มชั้นฮิดเดนเข้าไปอีกหนึ่งชั้น หรือถ้าต้องการจำนวนชั้นมากกว่านี้ก็สามารถทำได้โดยการเพิ่มชั้นของฮิดเดน



ภาพที่ 2.9 แสดงโครงข่าย Multilayer Perceptron ที่มี 3 ชั้น

ชั้นฮิดเดน เป็นตัวเพิ่มความสามารถให้แก่โครงข่ายเซลล์ประสาท โครงข่ายแบบ Multilayer Perceptron ถ้าไม่มีชั้นฮิดเดนก็จะกลายเป็นโครงข่ายแบบ Perceptron อย่างเช่น โครงข่ายแบบ Perceptron ที่เสนอโดย Rosenblatt ซึ่ง Minsky และ Papert ได้แสดงให้เห็นว่าโครงข่ายเซลล์ประสาท จำลองนี้เป็นโครงข่ายที่มีความสัมพันธ์ระหว่างอินพุตกับเอาต์พุตในลักษณะเชิงเส้นซึ่งไม่สามารถจะแก้ปัญหาเอ็กคลูซีฟ ออร์ (Exclusive OR : XOR) ได้ในกรณีที่ไม่ มีชั้นฮิดเดนความสัมพันธ์ของฟังก์ชันการแปลงดังกล่าวแสดงในตัวอย่างข้างล่างนี้

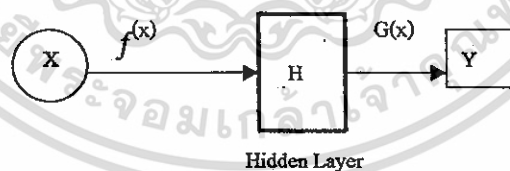


ภาพที่ 2.10 แสดงความสัมพันธ์ระหว่างตัวแปร X และ Y

จากภาพที่ 2.10 เมื่อ X เป็นตัวแปรอิสระ Y เป็นตัวแปรตามและ $f(X)$ เป็นฟังก์ชันการแปลงจาก X ไปยัง Y ในกรณีที่ X เป็นการเป็นอิสระต่อกันแบบเชิงเส้น (Linearly Independent) เราสามารถคำนวณฟังก์ชันการแปลง $f(X)$ ได้ ตัวอย่างเช่น ถ้า $\alpha_1 = \{0,1\}$ และ $\alpha_2 = \{1,0\}$ เมื่อ $X = [\alpha_1, \alpha_2]$ และ $Y = [1,1]$ ดังนั้นเราสามารถคำนวณฟังก์ชันการแปลงได้ ดังนี้

$$X \cdot f(X) = Y \rightarrow \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

เมื่อฟังก์ชันการแปลง $f(X) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ แต่ถ้าตัวแปร X ไม่เป็นอิสระต่อกันแบบเชิงเส้น (Linearly Dependent) เราไม่สามารถจะกำหนดฟังก์ชันการแปลง $f(X)$ ได้เช่น ถ้า $\alpha_1 = \{0,0\}$ และ $\alpha_2 = \{1,1\}$ เมื่อ $X = [\alpha_1, \alpha_2]$ และ $Y = [1,1]$ จะได้ว่า $X = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$ ซึ่งในกรณีนี้เราไม่สามารถคำนวณฟังก์ชัน $f(X)$ ที่สอดคล้องกับ Y ได้ และในกรณีนี้จะเหมือนกับปัญหา XOR แต่ถ้าเพิ่มชั้นฮิดเดนเข้าไปในภาพที่ 2.11 จะทำให้สามารถแก้ปัญหาที่เป็น Linearly Dependent นี้ได้



ภาพที่ 2.11 แสดงความสัมพันธ์ระหว่างตัวแปร X กับ Y โดยมีชั้นฮิดเดนเป็นตัวเชื่อม

ในการหาความสัมพันธ์จาก X ไปยัง Y โดยใช้ชั้นของฮิดเดนในทาง ANN จะทำให้ฟังก์ชัน $f(X)$ และ $g(X)$ ไม่เป็นเชิงเส้น ซึ่งฟังก์ชัน $f(X)$ และ $g(X)$ ในที่นี้ก็คือค่าถ่วงน้ำหนักที่ได้จากการสอนโครงข่ายนั่นเอง การที่จะกำหนดชั้นของฮิดเดนว่าในโครงข่ายหนึ่ง ๆ ควรมีฮิดเดนกี่ชั้น และแต่ละชั้นประกอบด้วยกี่โหนดนั้นไม่มีกฎเกณฑ์ที่แน่นอน ดังนั้นการกำหนดจำนวนชั้นและจำนวนโหนดของแต่ละชั้นของฮิดเดนสามารถทำได้โดยการทดลอง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.9 วิธีการของกฎเดลต้าเอนกประสม

กฎเดลต้า (Delta Rule) ถูกพัฒนาขึ้นมาเพื่อใช้สอนโครงข่ายเซลล์ประสาทจำลอง ซึ่งครั้งแรกใช้ในการสอนโครงข่ายเซลล์ประสาทจำลอง Perceptron ซึ่งจะเป็นการสอน Preceptron แบบต่อเนื่อง และต่อมาได้พัฒนาการสอนเดลต้าให้ใช้ได้กับโครงข่ายที่มีการเชื่อมต่อกันหลายชั้น จึงเรียกกฎเดลต้าที่ถูกพัฒนาขึ้นใหม่นี้ว่า กฎเดลต้าเอนกประสม (Generalized Delta Rule : GDR).

ขั้นตอนการสอนโครงข่าย Multilayer Perceptron จะเรียกว่า ขั้นตอนการสอนแบบค่าความผิดพลาดแพร่กระจายกลับหรือ Error Back-propagation ซึ่งเป็นการแพร่กระจายของค่าความผิดพลาดที่เกิดขึ้นในชั้นเอาต์พุตที่ต้องการกับเอาต์พุตที่คำนวณได้ โดยคำนวณย้อนกลับจากชั้นเอาต์พุตผ่านชั้นฮิดเดนตลอดมาจนถึงชั้นอินพุตเพื่อทำการปรับค่าถ่วงน้ำหนักจากภาพที่ 2.11 เมื่อทำการสอนโครงข่ายนี้ ซึ่งมีการสอนแบบชี้นำคั้งนั้นในการสอนจะต้องป้อนเซตของข้อมูลที่จะใช้สอนของอินพุต กำหนดให้ P เป็นจำนวนเซตทั้งหมดที่ใช้ในการสอน โครงข่ายคั้งนั้นเซตของอินพุตและเอาต์พุตที่ต้องการ จะมีอยู่ P เซต, ถ้าให้ D เป็นเซตของเอาต์พุตที่ต้องการ จะได้ว่า $D_p = \{d_1, d_2, d_3, \dots, d_L\}$ เมื่อ $P = 1, 2, 3, \dots, P$

W เป็นเมตริกซ์ของค่าถ่วงน้ำหนักที่เชื่อมโยงระหว่างชั้นของอินพุตกับชั้นฮิดเดนซึ่ง W มีขนาด M แถวและ N คอลัมน์

V เป็นเมตริกซ์ของค่าถ่วงน้ำหนักที่เชื่อมโยงระหว่างชั้นฮิดเดนไปยังชั้นเอาต์พุต ซึ่งมีขนาด L แถว และ N คอลัมน์

$$W = \begin{bmatrix} W_{11} & W_{12} & W_{13} & \dots & W_{1n} \\ W_{21} & W_{22} & W_{23} & \dots & W_{2n} \\ W_{31} & W_{32} & W_{33} & \dots & W_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ W_{M1} & W_{M2} & W_{M3} & \dots & W_{Mn} \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$V = \begin{bmatrix} v_{11} & v_{12} & v_{13} & \dots & v_{1m} \\ v_{21} & v_{22} & v_{23} & \dots & v_{2m} \\ v_{31} & v_{32} & v_{33} & \dots & v_{3m} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots \\ v_{L1} & v_{L2} & v_{L3} & \dots & v_{Lm} \end{bmatrix}$$

และ $f(\text{net})$ เป็น โคๆ ดังที่แสดงในภาพที่ 2.11

เมื่อทำการ ป้อนเซตของข้อมูลที่ใช้สอนเข้าสู่โครงข่าย จะสามารถคำนวณโหนดของฮิดเดนที่ m ได้ ดังนี้

$$\text{net}_m = \sum_{n=1}^N W_{mn} X_n \quad (2.4)$$

$$h_m = f(\text{net}_m)$$

และสามารถคำนวณเอาต์พุตโหนดที่ 1 ได้ดังนี้

$$\text{net}_t = \sum_{m=1}^M v_{1m} k_m \quad (2.5)$$

$$O_t = f(\text{net}_t)$$

เมื่อคำนวณชั้นของเอาต์พุตครบทุกโหนดแล้ว ขั้นต่อไปเป็นการปรับค่าถ่วงน้ำหนักของโครงข่าย โดยใช้ผลรวมของค่าความคลาดเคลื่อนกำลังสอง ระหว่างค่าของผลลัพธ์ที่ได้จากโครงข่ายกับค่าเอาต์พุตที่ต้องการของแพทเทิร์นที่ P ซึ่งสามารถคำนวณได้ดังนี้

$$E_p = \frac{1}{2} \sum_{i=1}^I (O_{pi} - d_{pi})^2 \quad (2.6)$$

ดังนั้นในการปรับค่าถ่วงน้ำหนักที่เชื่อมต่อระหว่างชั้นเอาต์พุตกับชั้นฮิดเดนสำหรับแพทเทิร์นที่ P (H_p) สามารถคำนวณได้ดังนี้

$$\Delta V = -\eta \frac{\partial E_p}{\partial V} \quad (2.7)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ $\delta_p^v = (Dp - Op)Op(1 - Op)$ ซึ่งเป็นความคลาดเคลื่อนภายในชั้นของเอาท์พุท

α เป็นค่าโมเมนตัม

η เป็นค่าอัตราการเรียนรู้ (Learning Rate) ซึ่งเป็นค่าคงที่ และการปรับค่าถ่วงน้ำหนักที่เชื่อมต่อระหว่างชั้นฮิดเดนกับชั้น อินพุทสามารถคำนวณได้ดังนี้

$$\Delta W = \alpha V + \eta \delta_p^w X_p \quad (2.8)$$

โดยที่ $\delta_p^w = Hp(1 - Hp)(\delta_p^v V)$ ซึ่งเป็นค่าความคลาดเคลื่อนภายในชั้นฮิดเดน สำหรับ α และ η จะมีคุณสมบัติเหมือนกับสมการที่ (2.8)

ในกรณีที่โครงข่ายมี 1 ชั้นสามารถปรับค่าถ่วงน้ำหนักโดยวิธีค่าความผิดพลาดแพร่กระจายกลับได้ดังนี้

$$\Delta W' = \alpha W' + \eta \delta_p' X_p^{i-1} \quad (2.9)$$

เมื่อ W' เป็นเมตริกซ์ค่าถ่วงน้ำหนักที่อยู่ระหว่างชั้นที่ i กับ $i-1$ ($i = 1, 2, 3, \dots$), X_p^{i-1} เป็นเซตของอินพุทแพทเทิร์นที่ P จากชั้นที่ $i-1$ และ δ_p' แวกเตอร์ของความคลาดเคลื่อนสำหรับชั้นที่ i สำหรับชั้นเอาท์พุทหรือชั้นที่ 1 สามารถคำนวณได้ดังนี้

$$\delta_p' = (o_p - o_p') o_p' (1 - o_p') \quad (2.10)$$

โดยที่ o_p^i คือชั้นของเอาท์พุท และสำหรับความคลาดเคลื่อนของชั้นที่ i ใดๆ โดยที่ $i \neq 1$ สามารถคำนวณได้ดังนี้

$$\delta_p^i = o_p^i (1 - o_p^i) (\delta_p^{i-1} W^{i-1}) \quad (2.11)$$

โดยที่ o_p^i คือเอาท์พุทชั้นที่ i แพทเทิร์นที่ P และสำหรับ δ_p^{i-1} และ W^{i-1} เป็นค่าความคลาดเคลื่อนและเมตริกซ์ค่าถ่วงน้ำหนักของชั้นถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.10 ปัจจัยที่ทำให้ประสิทธิภาพการสอนโครงข่ายด้วยวิธี Back-propagation เพิ่มขึ้นมีดังนี้

- การกำหนดค่าเริ่มต้นให้กับเมตริกซ์ถ่วงน้ำหนัก

ก่อนที่จะทำการสอนโครงข่าย Multilayer Perceptron โดยวิธี Error Back-Propagation จำเป็นต้องกำหนดค่าเริ่มต้นให้กับเมตริกซ์ค่าถ่วงน้ำหนักที่เชื่อมโยงระหว่างชั้นทุกชั้น โดยค่านี้จะเป็นตัวเลขจำนวนจริงที่มีค่าน้อย ๆ ที่ได้มาจากการสุ่ม ค่าเริ่มต้นของเมตริกซ์เชื่อมโยงจะมีผลต่อเวลาที่ใช้ในการสอนและอาจจะส่งผลถึงค่าถ่วงน้ำหนักหลังจากที่ได้สอน โครงข่ายไปแล้วว่าจะมีศักยภาพมากน้อยเพียงใด ในการแปลงข้อมูลอินพุตไปสู่เอาต์พุต ในอัลกอริทึมข้างล่างนี้ แสดงขั้นตอนการกำหนดค่าเริ่มต้นให้กับเมตริกซ์ค่าถ่วงน้ำหนักอัลกอริทึมข้างล่างนี้ ขั้นตอนการกำหนดค่าเริ่มต้นให้กับเมตริกซ์ค่าถ่วงน้ำหนัก

กำหนดให้ ROW เป็นจำนวนแถวของเมตริกซ์ถ่วงน้ำหนัก
COL เป็นจำนวนคอลัมน์ของเมตริกซ์ถ่วงน้ำหนัก
W เป็นเมตริกซ์ของค่าถ่วงน้ำหนักที่เชื่อมโยงระหว่างชั้นใด ๆ

```
Void Initial_Weights (Mutrix W, int ROW, int COL)
```

```
For (int I = 0 : i < ROW : I++)
```

```
For (int j = 0: j < COL : j++)
```

```
W[i][j] = 1 - random (101) / 53.0;
```

- อัตราเร็วในการเรียนรู้ (Learn Rate : η)

โดยทั่วไปแล้วค่านี้จะเป็นค่าคงที่ที่มีค่าอยู่ระหว่าง 0.05 - 0.25 แต่มีงานวิจัยบางส่วนที่ค่าอัตราเร็วในการเรียนรู้มีการปรับค่าได้ เช่น กำหนดให้อัตราเร็วในการเรียนรู้ผันแปรตามจำนวนรอบที่สอน อัตราเร็วในการเรียนรู้ผันแปรตามค่าความผิดพลาดเฉลี่ยของระบบ และค่าอัตราเร็วในการเรียนรู้ผันแปรตามค่าความแปรปรวนของแต่ละแพทเทิร์น

- โมเมนตัม (Momentum : α)

สำหรับกฎการสอบแบบเจเนอราลไลซ์ลดค่า โดยทั่วไปค่าโมเมนตัมอาจจะใช้หรือไม่ใช้ก็ได้ แต่ค่านี้จะมีคุณสมบัติช่วงป้องกันการแกว่ง (Oscillate) ของระบบ โดยค่านี้จะสัมพันธ์กับค่าอัตราเร็วในการเรียนรู้ คือถ้าอัตราเร็วในการเรียนรู้มีค่ามากแต่ค่าโมเมนตัมมี

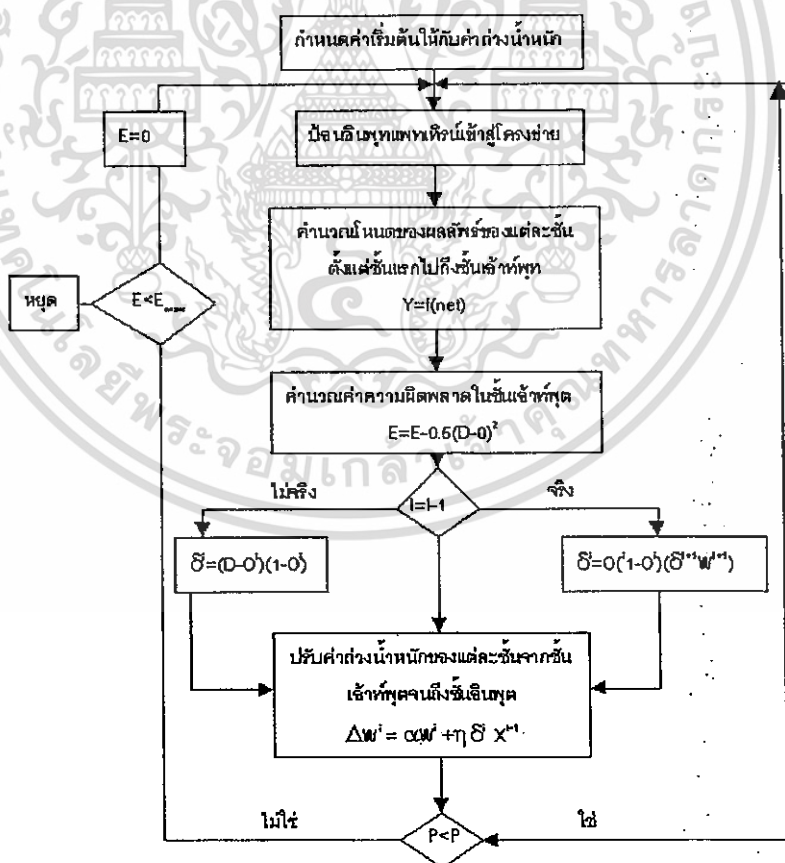
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าน้อยจะทำให้ระบบโครงข่ายเกิดการแกว่ง ส่วนกรณีอื่น ๆ โครงข่ายจะไม่เกิดการแกว่งแต่จะมีผลต่อเวลาที่ใช้ในการสอน

- ไบแอส (Bias)

โหนดไบแอสสำหรับโครงข่ายโดยทั่วไปอาจจะมีหรือไม่มีก็ได้ ในกรณีของโครงข่ายที่ใช้ในระบบที่เป็นReal-time แล้วจะไม่ค่อยมีการใช้โหนดไบแอสเนื่องจากจะเสียเวลาในการคำนวณ และถ้ามีการออกแบบโครงข่ายอย่างเหมาะสมแล้วเทอมไบแอสไม่จำเป็นต้องมีแต่ถ้าโครงข่าย มีจำนวนโหนดไม่เหมาะสมคือมีจำนวนโหนดน้อยไป การเพิ่มโหนดไบแอสเข้าไปจะทำให้โครงข่ายสามารถจดจำได้ดีขึ้น โหนดไบแอสจะถูกเพิ่มเข้าไปเป็นโหนด ๆ หนึ่งในแต่ละชั้น ยกเว้นในชั้นของผลลัพธ์และโหนดไบแอสที่เพิ่มเข้าไปมักจะเป็นค่าคงที่ ผลของการเพิ่มโหนดไบแอสเข้าไปในโครงข่ายจะทำให้ประสิทธิภาพการสอนดีขึ้น

2.1.11 ขั้นตอนการสอนโครงข่ายประสาทเทียม



ภาพที่ 2.12 แผนภูมิการสอนแบบ Error Back-propagation ที่มี 1 ชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสอนโครงข่าย Multilayer Perception ด้วยวิธี Back-Propagation เป็นโครงข่ายแบบไม่มีไบแอส หรือการถ่วงน้ำหนัก และในภาพที่ 2.12 เป็นแผนภูมิของขั้นตอนการสอนของ อัลกอริทึมนี้โดยที่ E_{max} เป็นค่าความผิดพลาดที่ต่ำสุดที่พอจะยอมรับได้ ซึ่งในการสอนโครงข่ายในรอบหนึ่ง ๆ (Epoch) ก็คือการสอนชุดของแพทเทิร์นทั้งหมด P แพทเทิร์น ถ้าค่าความผิดพลาดของระบบ E ยังมีค่ามากกว่าค่าความผิดพลาดที่ตั้งไว้ E_{max} ก็จะนำข้อมูลชุดเดิมมาทำการสอนใหม่จนกว่าค่าความผิดพลาด E จะน้อยกว่าค่า E_{max}

2.1.12 การสอนโครงข่ายประสาทเทียมโดยการใช้อัลกอริทึมของ MARQUARDT - LEVENBERG

อัลกอริทึมการเรียนรู้แบบการแพร่กลับ (back-propagation) เป็นโมเดลของโครงข่ายประสาทที่ได้รับ ความนิยมมากที่สุด งานวิจัยต่าง ๆ ถูกทำเพื่อสนับสนุน โครงสร้างนี้ เพื่อที่จะหาวิธีเร่งความสามารถในการจดจำของโครงข่าย โดยทั่วไปมีวิธีที่นิยมทำกัน 2 วิธี คือ

1. การพัฒนาเทคนิค *al hoe*[2] — [5] วิธีนี้จะใช้การปรับเปลี่ยนอัตราการเรียนรู้ การใช้ค่าโมเมนต์ และการปรับสเกล (rescaling) อีกกรณีหนึ่ง คือการหาวิธีทางคณิตศาสตร์ที่เหมาะสมที่ให้ค่าที่ดีที่สุด

2. การใช้ *conjugate gradient* หรือ *Quasi-Newton* วิธีของ Newton มีประสิทธิภาพมากแต่จะ ต้องใช้เวลาในการคำนวณมากขึ้นอยู่กับขนาดของโครงข่าย และยังจำกัดที่ขนาดของหน่วยความจำ

สำหรับวิธีทางคณิตศาสตร์อื่นๆ คือการประยุกต์ใช้โครงข่ายประสาทเทียมเป็นแบบไม่เป็นเชิงเส้น *least-squares* วิธีการออปติ้ม โดยทั่วไป ถูกออกแบบให้ทำงานกับ ฟังก์ชันเป็นแบบเชิงเส้น อย่างไรก็ตามเมื่อรู้เป้าหมายของโครงข่ายก็มีความเป็นไปได้ที่จะออกแบบอัลกอริทึมที่มีประสิทธิภาพมากกว่าได้ วิธีการประยุกต์แบบไม่เชิงเส้นกับโครงข่ายประสาทเทียมเน้นหนักบนการประยุกต์แบบ ซีควนเชียล (sequential) เมื่อค่า *weight* ถูกปรับเปลี่ยนหลังจาก presentation ของคู่ อินพุท/เอาต์พุท วิธีการนี้มีประโยชน์ เมื่อเป็นการสอนแบบ *on-line* แต่วิธีนี้ต้องการการประเมินค่าหลายแบบ สำหรับอัลกอริทึมมาตรฐาน

สำหรับโครงข่ายขนาดใหญ่ซึ่งต้องการหน่วยความจำขนาดใหญ่ในการคำนวณ และเป็นอุปสรรคอย่างยิ่งต่อเครื่องคอมพิวเตอร์ที่ใช้ในการคำนวณ (โดยเฉพาะอย่างยิ่ง วิธี *Quasi-Newton*) อย่างไรก็ตาม สำหรับโครงข่ายซึ่งประกอบด้วยค่าถ่วงน้ำหนัก ซึ่งมีขนาดไม่เกินหลักร้อยวิธีนี้จะมี ประสิทธิภาพมากกว่า เมื่อเปรียบเทียบกับเทคนิค *conjugate gradient*

อัลกอริทึมการแพร่กลับ (BACK-PROPAGATION ALGORITHM)

เมื่อพิจารณาโครงข่ายหลายชั้น (multilayer feed forward network) เช่น โครงข่าย 3 ชั้น ดังภาพที่ 2.13 ค่า net อินพุต จนถึง i ในชั้น $K+1$ คือ

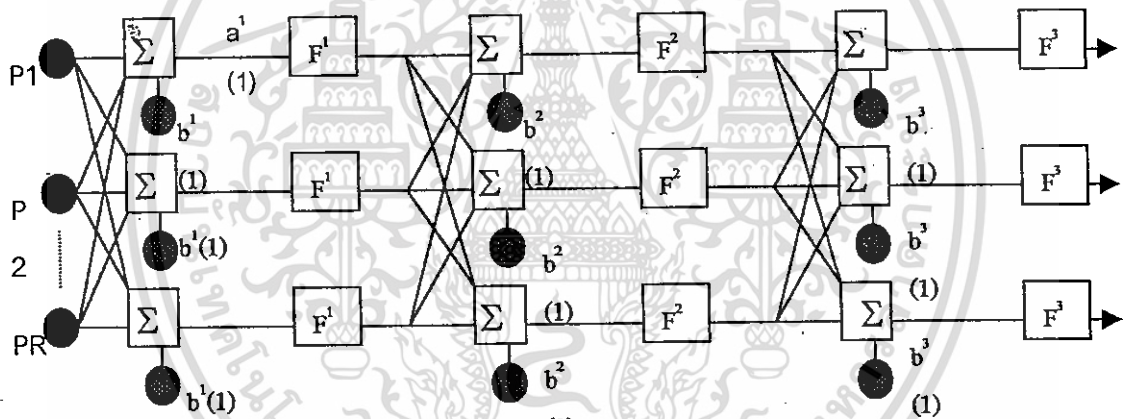
$$n^{k+1}(i) = \sum_{j=1}^k W^{k+1}(i, j) a^k(j) + b^{k+1}(i) \quad (2.12)$$

ค่าเอาต์พุตของหน่วย i จะเป็น

$$a^{k+1}(i) = f^{k+1}(n^{k+1}(i)) \quad (2.13)$$

$$a^0 = \underline{p} \quad (2.14)$$

$$a^{k+1}(i) = f^{k+1}(W^{k+1} \underline{a}^k + \underline{b}^{k+1}), K = 0, 1, \dots, M-1 \quad (2.15)$$



ภาพที่ 2.13 โครงข่ายประสาทเทียมแบบ Back-Propagation

ความสัมพันธ์ของคู่อินพุต / เอาต์พุต

$$\{ (p_1, t_1), (p_2, t_2), \dots, (p_q, t_q) \} \quad (2.16)$$

ดัชนีประสิทธิภาพ ของโครงข่ายคือ

$$V = \frac{1}{2} \sum_{q=1}^Q (\underline{t}_q - \underline{a}_q^M)^T (\underline{t}_q - \underline{a}_q^M) = \frac{1}{2} \sum_{q=1}^Q \underline{e}_q^T \underline{e}_q \quad (2.17)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ a_q^M คือ เอาท์พุทของโครงข่าย

เมื่อ q th อินพุท , p_q

$e_q = t_q - a_q^M$ คือค่า เออเรอร์ สำหรับ q th อินพุทสำหรับโครงข่ายแพร่กลับแบบมาตรฐาน เราใช้ an approximate steepest descent rule the performance index กำหนดให้เป็น

$$\hat{V} = \frac{1}{2} \underline{e}_q^T \underline{e}_q \quad (2.18)$$

เมื่อผลรวมของ squares ถูกแทนที่โดย square errors สำหรับคู่ของ อินพุท / เอาท์พุท แบบเดี่ยว The approximate steepest (gradient) descent algorithm คือ

$$\Delta W^k(i, j) = -\alpha \frac{\partial \hat{V}}{\partial w^k(i, j)} \quad (2.19)$$

$$\Delta b^k(i) = -\alpha \frac{\partial \hat{V}}{\partial b^k(i)} \quad (2.20)$$

เมื่อ α คืออัตราการเรียนรู้ (learning rate) ถูกใช้ในการเปลี่ยนหน่วย i ในชั้น k

$$\delta^k(i) = \frac{\partial \hat{V}}{\partial n^k(i)} \quad (2.21)$$

จากการใช้สมการ (2.19) (2.20) และ (2.21) จะได้

$$\frac{\partial \hat{V}}{\partial w^k(i, j)} = \frac{\partial \hat{V}}{\partial n^k(i)} \frac{\partial n^k(i)}{\partial w^k(i, j)} = \delta^k(i) a^{k-1}(j) \quad (2.22)$$

$$\frac{\partial \hat{V}}{\partial b^k(i)} = \frac{\partial \hat{V}}{\partial n^k(i)} \frac{\partial n^k(i)}{\partial b^k(i)} = \delta^k(i) \quad (2.23)$$

$$\underline{\delta}^k = F^k(\underline{n}^k) W^{k+1T} \underline{\delta}^{k+1} \quad (2.24)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ

$$F^k(\underline{n}^k) = \begin{bmatrix} f^k(n^k(1)) \\ 0 \end{bmatrix} \quad (2.25)$$

และ

$$f^k(n) = \frac{df^k(n)}{dn} \quad (2.26)$$

This recurrence relation ถูก อินนิเชียนที่ขั้นสุดท้าย

$$\underline{\delta}^M = -F^M(\underline{n}^M)(t_q - \underline{a}_q) \quad (2.27)$$

การสอนโครงข่ายเริ่มจาก การแพร่จากอินพุตโดยใช้สมการ (2.12) และการแพร่กลับโดยใช้สมการ (2.24) และ (2.25) สุดท้ายทำการปรับเปลี่ยน weight และค่า ออฟเซตโดยใช้สมการ (2.19), (2.20), (2.22) และ (2.26) ตามลำดับ

MARQUARDT - LEVENBERG MODIFICATION

Marquardt - Levenberg ถูก approximation คล้ายวิธีของ Newton สมมุติว่าเรามีฟังก์ชัน ซึ่งเราต้องการให้ minimize with respect to the parameter vector \underline{x} ดังนั้นวิธีของ Newton จะเป็น

$$\Delta \underline{x} = -[\nabla^2 V(\underline{x})]^{-1} \nabla V(\underline{x}) \quad (2.28)$$

เมื่อ $\nabla^2 V(\underline{x})$ คือ เมตริกซ์ Hessian และ $\nabla V(\underline{x})$ คือ gradient ถ้าเรากำหนดให้ $V(\underline{x})$ คือ ผลรวมของ square function

$$V(\underline{x}) = \sum_{i=1}^N e_i^2(\underline{x}) \quad (2.29)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นแสดงได้ว่า

$$\nabla V(\underline{x}) = J^T(\underline{x})\underline{e}(\underline{x}) \quad (2.30)$$

$$\nabla^2 V(\underline{x}) = J^T(\underline{x})J(\underline{x}) + S(\underline{x}) \quad (2.31)$$

เมื่อ $J(\underline{x})$ คือ เมทริกซ์ จาคอบีเยน

$$J(\underline{x}) = \begin{bmatrix} \frac{\partial e_1(\underline{x})}{\partial x_1} \\ \frac{\partial e_2(\underline{x})}{\partial x_1} \\ \vdots \\ \frac{\partial e_N(\underline{x})}{\partial x_1} \end{bmatrix} \quad (2.32)$$

และ

$$S(\underline{x}) = \sum_{i=1}^N e_i(\underline{x}) \nabla^2 e_i(\underline{x}) \quad (2.33)$$

สำหรับวิธี Gauss-Newton ถ้า $S(\underline{x}) \approx 0$ และ ปรับสมการที่ (28) จะเป็น

$$\Delta \underline{x} = [J^T(\underline{x})J(\underline{x})]^{-1} J^T(\underline{x})\underline{e}(\underline{x}) \quad (2.34)$$

การปรับวิธีของ Gauss-Newton โดยใช้การปรับของ Marquardt-Levenberg

$$\Delta \underline{x} = [J^T(\underline{x})J(\underline{x}) + \mu I]^{-1} J^T(\underline{x})\underline{e}(\underline{x}) \quad (2.35)$$

ตัวแปร μ ถูกคูณด้วยค่าตัวแปร (β) whenever a step ซึ่งจะเพิ่มผลของ $V(\underline{x})$

เมื่อลด $V(\underline{x})$ μ ถูกหารด้วย β สังเกตว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ μ มีค่ามาก อัลกอริทึม becomes steepest descent (with step $\frac{1}{\mu}$) ในขณะที่ μ มีค่าน้อย อัลกอริทึมจะกลายเป็นวิธีของ Gauss - Newton

The key step ของอัลกอริทึมนี้ คือ การคำนวณของเมตริกซ์จาโคเบียน สำหรับโครงข่ายประสาทเทียมในเทอมของจาโคเบียน สามารถคำนวณโดยการปรับเปลี่ยนอัลกอริทึมการแพร่กลับแบบธรรมดา คำนวณที่ใช้ประสิทธิภาพ (performance index) สำหรับการ mapping ถูกกำหนดโดยสมการที่ (2.36)

$$\underline{x} = [w^1(1,1)w^1(1,2).....w^1(S_1,R)b^1(1).....b^1(S_1)w^2(1,1).....b^M(SM)] \quad (2.36)$$

และ $N = Q \times SM$ การแพร่กระจายกลับแบบมาตรฐานสามารถคำนวณได้จากเทอม

$$\frac{\partial \hat{V}}{\partial w^k(j,j)} = \frac{\partial \sum_{m=1}^{SM} e_q^2(m)}{\partial w^k(j,j)} \quad (2.37)$$

สำหรับส่วนย่อยใน เมตริกซ์จาโคเบียน ที่ใช้วิธี Marquardt คำนวณได้จาก

$$\frac{\partial e_q(m)}{\partial w^k(j,j)} \quad (2.38)$$

เทอมนี้สามารถคำนวณ โดยการใช้ค่าการแพร่กลับมาตรฐานที่ เลขอร์สุดท้าย

$$\Delta M = -F^M (\underline{\eta}^M) \quad (2.39)$$

แต่ละคอลัมน์ของเมตริกซ์ในสมการที่ (2.26) คือค่า sensitivity เวกเตอร์ ซึ่ง back propagation through the network to produce one row of the Jacobian

สรุปวิธี Marquardt

1. ที่อินพุตแต่ละโหนดคำนวณหาความสัมพันธ์กับเอาต์พุต โดยการใช้สมการ (2.30) และ (2.31) และหาค่า error จาก $(e_q = t_q - a_q^M)$ และคำนวณหาค่า sum square error ของอินพุตแต่ละโหนด

2. คำนวณเมตริกซ์จาโคเบียน โดยการใช้สมการ (2.32) , (2.33) , (2.34) , (2.35) และ (2.36)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แก๊สมถการ (2.39) หาค่า Δx

4. คำนวณค่า sum square error โดยการให้ $x + \Delta x$ ถ้าผลรวมของ square error น้อยกว่า ค่าคำนวณในขั้นที่ 1 ให้ลดค่า μ โดย β กำหนดให้ $x = x + \Delta x$ และกลับไปขั้น (step) 1 ถ้าผลรวมของ square error ไม่ลดให้เพิ่มค่า μ โดย β และไปขั้น (step) 3.

5. คำนวณจนค่า square อยู่เข้าสู่ค่าที่ตั้งไว้ (2.39)

2.2 ยานขนส่งแบบนำร่องอัตโนมัติ (Automatic Guided Vehicle) หรือ AGV

ยานขนส่งนำร่องอัตโนมัติ หรือ AGV จัดเป็นหุ่นยนต์เคลื่อนที่ชนิดหนึ่ง ซึ่งมีใช้อย่างแพร่หลายในอุตสาหกรรมมากกว่า 30 ปี AGV เป็นยานพาหนะที่ไม่ต้องใช้คนขับ สามารถวิ่งไปตามเส้นทางที่กำหนดได้เองโดยอัตโนมัติ สามารถใช้แทนยานพาหนะ เช่น รถโฟล์คลิฟท์, รถบรรทุก, รถแทรกเตอร์ เป็นต้น หรืออาจใช้แทนระบบสายพานลำเลียง AGV มีหน้าที่ขนส่งวัตถุดิบ ชิ้นส่วนผลิตภัณฑ์ หรือสินค้าจากสถานที่หนึ่งไปยังอีกสถานที่หนึ่ง AGV มีใช้มากในโรงงานแบบอัตโนมัติ AGV มักใช้ในโรงงานแบบยืดหยุ่น FMS เพื่อป้อนงานแก่เครื่องจักรประเภท CNC, DNC AGV เป็นส่วนหนึ่งในระบบ AMDC หรือในโรงเก็บวัสดุ ปัจจุบันนี้เนื่องจากระบบอัตโนมัติเข้ามามีบทบาทในโรงงานอย่างแพร่หลาย จึงมักจะพบว่ามีการใช้ AGV อยู่มาก AGV จึงกลายเป็นองค์ประกอบหนึ่งที่สำคัญในระบบโรงงาน FMS การขนส่งวัสดุมีตั้งแต่การนำวัสดุจาก โรงเก็บของไปยังพื้นที่ปฏิบัติงานหรือขนส่งระหว่างพื้นที่ปฏิบัติงาน จากจุดหนึ่งไปยังอีกจุดหนึ่ง รวมไปถึงการนำผลผลิตไปเก็บยังโกดังเก็บสินค้า ในโรงงานที่ใช้ระบบอัตโนมัติทั้งหมดนั้น ระบบขนส่ง จะถูกควบคุมการทำงานจากคอมพิวเตอร์ส่วนกลางโดยสามารถขนวัสดุตามคำสั่งลักษณะต่างๆ ไปยังตำแหน่งใดก็ได้ ไม่ว่าจะเป็นหน่วยเครื่องจักร สถานีประกอบชิ้นส่วน หรือ โรงเก็บของใน โรงงานประเภทประกอบชิ้นส่วน ระบบขนส่ง เช่น AGV สายพานลำเลียง ถือได้ว่าเป็นเครื่องมือในการจัดการขั้นแรก สำหรับนำชิ้นงานจากโรงเก็บของไปยังอุปกรณ์จัดการขั้นต่อไป เช่น แขนหุ่นยนต์เป็นต้น เพื่อให้แขนหุ่นยนต์หยิบชิ้นงานป้อนเครื่องจักรต่อไป AGV ในปัจจุบัน ได้รับการออกแบบเพื่อให้สามารถทำงานภายใต้การควบคุมจากคอมพิวเตอร์ส่วนกลาง

โดยสามารถโปรแกรมให้ AGV วิ่งไปตามเส้นทางและหยุดยังตำแหน่งเป้าหมายได้อย่างถูกต้องแม่นยำ ในปัจจุบัน AGV ส่วนใหญ่จะถูกกำหนดให้เคลื่อนที่ไปตามทางเดินนำร่อง ซึ่งติดตั้งตามพื้นโรงงาน ทางเดินนำร่องที่นิยมใช้เช่น การฝังสายตัวนำใต้พื้น การใช้แถบสีติดบนพื้น เป็นต้น โครงสร้างภายนอกสำหรับ AGV ส่วนมากจะทำเป็นแท่นไว้วางภาชนะที่ใส่วัตถุดิบ เช่น ถาด ถัง หรือชั้นวางของ เป็นต้น ในการติดตั้งระบบ AGV นั้นสิ่งที่ควรคำนึงถึงคือ ต้องไม่ไปกีดขวางทางเข้าสู่เครื่องจักรเพื่อซ่อมแซมหรือบำรุงรักษา ในทางตรงกันข้ามควรเป็นการเพิ่มความสะดวกให้กับงาน ปัจจุบันนี้เทคโนโลยีทาง AGV ก้าวหน้าไปมาก มีการวิจัยและพัฒนา AGV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบเคลื่อนที่อิสระกันอย่างกว้างขวาง AGV แบบนี้ไม่ต้องติดตั้งทางเดินนำร่องบนพื้น แต่จะใช้ระบบนำทางซึ่งอยู่บนตัว AGV ควบคุมตำแหน่งการเคลื่อนที่ เทคนิคสำหรับ AGV แบบเคลื่อนที่อิสระนั้นมีหลายอย่าง บ่อยครั้งที่มีการนำเทคนิคต่างๆ มาผสมผสานกันอย่างไรก็ตาม โรงงานอัตโนมัติในปัจจุบันส่วนมากก็ยังคงใช้ AGV แบบมีทางเดินนำร่องอยู่ โรงงานหลายแห่งมีการติดตั้งแขนหุ่นยนต์บน AGV ทำให้ AGV มีลักษณะเป็นหุ่นยนต์เคลื่อนที่สมบูรณ์แบบ วิธีนี้ทำให้สามารถใช้หุ่นยนต์เพียงตัวเดียวกับเครื่องจักรหลายตัวเป็นการเพิ่มประสิทธิภาพ ในการใช้งานแขนหุ่นยนต์ และลดค่าใช้จ่าย

AGV ส่วนมากจะใช้แบตเตอรี่รีชาร์จที่ถ่วงกรวดและขับเคลื่อนด้วยมอเตอร์กระแสตรง ในกรณีที่ทิศทางการขนส่งวัสดุมีความแน่นอน ปริมาณการขนส่งมีจำนวนมากตลอดจนการขนส่งเป็นไปอย่างต่อเนื่องตลอดเวลา ระบบ AGV จะเสียเปรียบระบบสายพานลำเลียง แต่ข้อดีของ AGV เหนือสายพานลำเลียงคือ ไม่เกาะเกาะการติดตั้งทางเดินนำร่องมีความยืดหยุ่นกว่าระบบสายพานลำเลียง เพราะสามารถดัดแปลงแก้ไขได้ง่าย นอกจากนี้ในโรงงานอุตสาหกรรมแล้ว AGV ยังสามารถนำไปประยุกต์ใช้ในงานด้านอื่นๆ อีกมาก เช่น การขนส่งสัมภาระไปยังห้องต่างๆ ของแขกที่มาพักในโรงแรม การรับส่งเอกสารในสำนักงาน โดย AGV ทำหน้าที่เป็นรถส่งสารแทนการใช้พนักงาน การบริการอาหารตามโรงแรมและภัตตาคาร หรือการขนส่งวัสดุในสถานที่ ที่เป็นความลับทางราชการและอื่นๆ

2.2.1 การขับเคลื่อนและการเลี้ยวของ AGV

AGV ส่วนใหญ่ใช้พลังงานในการขับเคลื่อนจากแบตเตอรี่รีชาร์จที่ถ่วงกรวด และขับเคลื่อนโดยโช้มอเตอร์กระแสตรงซึ่งมีควบคุมทิศทางการหมุนความเร็วและตำแหน่งโดยใช้คอมพิวเตอร์ซึ่งติดตั้งอยู่บน AGV สำหรับเทคนิคการเลี้ยวนั้นจะขึ้นอยู่กับโครงสร้างระบบล้อ ระบบล้อตามปกติที่ใช้กันอาจจะเป็นระบบ 3 ล้อ หรือ 4 ล้อ ก็ได้ AGV บางตัวอาจใช้ถึง 6 ล้อ ทั้งนี้ขึ้นอยู่กับผู้ผลิต เมื่อเปรียบเทียบระบบ 4 ล้อ กับระบบ 3 ล้อ แล้วระบบ 4 ล้อ จะมีเสถียรภาพดีกว่า กำลังลากจูงดีกว่า ช่องว่างทางกลน้อยกว่า แต่ระบบ 3 ล้อ ก็มีข้อดีตรงที่ระบบเลี้ยวไม่ยุ่งยากซับซ้อน โครงรถมีน้ำหนักเบากว่า ซึ่งทำให้ประหยัดพลังงานจากแบตเตอรี่ได้มากกว่า

2.3 การติดต่อสื่อสารกับ PLC

PLC มาจากคำเต็มว่า Programmable Logic Controller หรือ PC เป็นอุปกรณ์ที่ได้รับการคิดค้น พัฒนาและผลิตขึ้นเพื่อประโยชน์ ในการควบคุมการทำงานทางไฟฟ้าและอิเล็กทรอนิกส์ โดยผู้ใช้งานสามารถจัดการให้ PLC ทำงานตามขั้นตอนและรูปแบบที่ต้องการโดยการ ป้อนโปรแกรมคำสั่งงานแทนการเดินทางวงจรด้วยสายไฟฟ้าในระบบ Relay หรือ Hard Wired Controller PLC มีความสามารถในการทำงานเช่นเดียวกับระบบ Relay แต่มีคุณสมบัติพิเศษที่เหนือเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กว่าระบบ Relay หลายประการ อาทิ เช่น ความสามารถในการปรับปรุง เปลี่ยนแปลง แก้ไขวงจร หรือทำวงจรใหม่ได้สะดวกกว่า นอกจากนี้ PLC ยังมี Function พื้นฐานจำนวนมากที่มี ความสำคัญ และมีความจำเป็นต้องใช้ในระบบควบคุมทั่วไป เช่น Time/Counter/Real Time Clock/การคำนวณ ทางคณิตศาสตร์ และอื่นๆ นอกจากนี้ในปัจจุบัน PLC ยังมีขีดความสามารถใช้งานควบคุมด้าน Analog อย่างมีประสิทธิภาพอีกด้วย ในส่วนของการซ่อมบำรุง สามารถ ทำการซ่อมบำรุง PLC ได้ ง่ายกว่าระบบ Relay โดยสามารถตรวจสอบหาข้อบกพร่องต่างๆ และแก้ไขได้อย่างรวดเร็ว

ในขณะที่ขีดความสามารถของ PLC ได้รับการพัฒนาขึ้นไปในระดับที่สูง ทั้งด้านศักยภาพ ของการนำไปใช้งานด้านความสะดวกต่อผู้ใช้งาน กล่าวคือ PLC สามารถใช้งานควบคุมทาง ไฟฟ้าหรืออิเล็กทรอนิกส์ได้เกือบทุกประเภท บุคลากรที่ปฏิบัติหน้าที่ทางเทคนิคทั่วไปไม่ว่าจะเป็น ทางเครื่องกล หรือไฟฟ้า จะมีความรู้ด้านวงจรไฟฟ้าหรือไม่ก็ตาม หากได้รับการศึกษาฝึกฝน ที่ถูกวิธี ก็จะสามารถเขียนโปรแกรมและใช้งาน PLC ได้เป็นอย่างดี ถึงจะใช้เวลาที่ฝึกเพียงสั้นๆ ก็ ตาม

2.3.1 โครงสร้างของ PLC

โครงสร้างของ PLC นั้นจะประกอบไปด้วย หน่วยประมวลผลกลาง หน่วยความจำ หน่วย รับข้อมูล หน่วยส่งข้อมูล และหน่วยป้อนโปรแกรม โดยส่วนประกอบทั้งหมดของ PLC นั้นจะรวม อยู่ภายในเครื่องเดียวกัน แต่ถ้าเป็นขนาดใหญ่ๆ สามารถถอดแยกส่วนประกอบได้ หน่วยความจำ ของ PLC ประกอบไปด้วย หน่วยความจำชนิด RAM , ROM หน่วยความจำชนิด RAM ทำหน้าที่ เก็บโปรแกรมของผู้ใช้งาน และข้อมูลสำหรับการปฏิบัติงานของ PLC ส่วน ROM ทำหน้าที่ เก็บโปรแกรมสำหรับการปฏิบัติงานของ PLC ตามที่ผู้ใช้โปรแกรม โดยเราสามารถแบ่ง ประเภทของหน่วยความจำดังนี้

- RAM (Random Access Memory) หน่วยความจำประเภทนี้จะมีแอสเซมบลีโค้ดๆ ต่อไว้ เพื่อ ใช้เก็บข้อมูลเมื่อเกิด ไฟดับ การอ่านและการเขียนโปรแกรมลงใน RAM ทำได้ง่ายมาก จึงเหมาะกับการ ใช้งาน ในระยะทดลองเครื่องที่มีการเปลี่ยนแปลงแก้ไขโปรแกรมบ่อยๆ

- EPROM (Erasable Programmable Read Only Memory) หน่วยความจำชนิด EPROM นี้ จะต้องใช้เครื่องมือพิเศษในการเขียน โปรแกรม การลบโปรแกรมทำได้โดยใช้แสงอัลตราไวโอเลต หรือตากแดดร้อนๆ นานๆ มีข้อดีตรงที่โปรแกรมจะไม่สูญหายแม้ไฟดับ จึงเหมาะ สำหรับการใ้ งานที่ไม่ต้องการ โปรแกรมบ่อยๆ

- EEPROM (Electrical Erasable Programmable Read Only Memory) หน่วยความจำชนิด นี้ไม่จำเป็นต้องใช้เครื่องมือพิเศษในการเขียน และลบโปรแกรม โดยใช้วิธีการทางไฟฟ้าเหมือนกับ RAM

นอกจากนี้ก็ไม่จำเป็นต้องมีแบตเตอรี่สำรองไฟเมื่อไฟดับ ราคาจึงจะแพงกว่า แต่คุณสมบัติที่ดีของ RAM และ EPROM

2.3.2 ส่วนประกอบของ PLC

PLC แบ่งออก 3 ส่วนด้วยกันคือ

2.1 ส่วนที่เป็นหน่วยประมวลผลกลาง (Control Processing Unit : CPU)

2.2 ส่วนที่เป็นอินพุต/เอาต์พุต (Input Output : I/O)

2.3 ส่วนที่เป็นอุปกรณ์การโปรแกรม (Programming Device)

CPU เป็นส่วนมันสมองของระบบ ภายใน CPU จะประกอบไปด้วยวงจร Logic Gate ชนิดต่างๆ หลายชนิด และมี Microprocessor-based ใช้สำหรับแทนอุปกรณ์จำพวกรีเลย์ (Relay) เคาน์เตอร์ (Counter) ไทเมอร์ (Timer) แลชีควนเซอร์ (Sequencers) เพื่อให้ผู้ใช้ได้ออกแบบใช้วงจรรีเลย์ แลคเคอร์ ลอจิก เข้าไปได้ CPU จะยอมรับ (Read) อินพุตค่า Z(Input Data) จากอุปกรณ์ให้สัญญาณ (Sensing Device) ต่างๆ จากนั้นจะปฏิบัติการและเก็บข้อมูล โดยใช้โปรแกรมจากหน่วยความจำ และส่งข้อมูลที่เหมาะสมถูกต้องไปยังอุปกรณ์ควบคุม (Control Device) แหล่งจ่ายไฟฟ้ากระแสตรง (DC Current) โวลต์ต่ำ (Low Level Voltage) ซึ่งใช้โดยโปรเซสเซอร์ (Processor) และไอโอ โมดูล (I/O Modules) และแหล่งจ่ายไฟนี้จะเก็บไว้ที่ CPU หรือแยก ออกไปติดตั้งที่จุดอื่นก็ได้ขึ้นอยู่กับผู้ผลิตแต่ราย

2.3.3 การติดต่อสื่อสารระบบ PLC

ในการติดต่อสื่อสารของ PLC ไม่ว่าจะเป็นการระบบควบคุมขนาดใหญ่ที่ประกอบไปด้วยคอมพิวเตอร์ควบคุมหลักของ PLC หรืออุปกรณ์อื่น เพื่อต้องการให้มีประสิทธิภาพและความเร็วในการทำงานที่สูงขึ้น สามารถทำงานเป็นปกติภายใต้สภาวะแวดล้อมของโรงงานอุตสาหกรรมต่างๆ เพื่อหน้าที่ในการสั่งงานและการกระจายข้อมูลที่แสดงสภาพการควบคุมและปฏิบัติงานของคนไปยังอุปกรณ์อื่นในระบบควบคุม ระบบการติดต่อที่ใช้ในการสื่อสารของ PLC หรือเส้นทาง การสื่อสารข้อมูลความเร็วสูง (Data highways) มีลักษณะที่แตกต่างจากระบบการติดต่อสื่อสารของคอมพิวเตอร์ ทั่วไป แต่ในปัจจุบันสามารถที่จะคิดแปลงมาใช้ในการติดต่อสื่อสารของ PLC การติดต่อที่ใช้โดยทั่วไปประมาณ 9600 Bit ขึ้นอยู่กับระยะทางการติดต่อสื่อสารด้วย

2.3.4 การติดต่อสื่อสารข้อมูลระหว่าง PLC มี 2 ระบบในการใช้ติดต่อสื่อสารคือ

1. Master slave system จะประกอบไปด้วยคอมพิวเตอร์ควบคุมหลัก ทำหน้าที่ในการจัดการดูแลสื่อสารข้อมูลทั้งหมดในระบบโครงข่าย อุปกรณ์ทุกจุดในระบบโครงข่ายไม่สามารถติดต่อเอกสารแบบเอกสารที่ส่งงานในเวลาสำหรับการทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

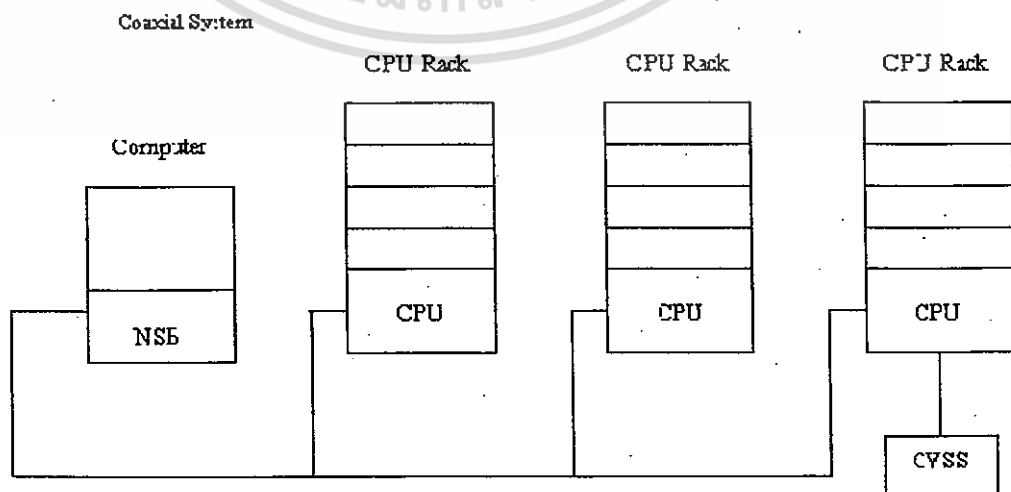
กันได้โดยตรง การรับส่งข้อมูล หรือส่งข้อมูลทำให้ระบบ Master slave system มีการ ตรวจสอบ ข้อมูล มีการติดต่อสื่อสารและมีการตรวจสอบที่สะดวกมากขึ้น

2. Peer to peer systems คือการติดต่อกันระหว่าง PLC กับ PLC โดยไม่ต้องมีคอมพิวเตอร์ เข้าช่วย อุปกรณ์ทั้งหมดสามารถที่จะดู โครงข่าย เพื่อจะติดต่อกับอุปกรณ์อื่นได้เช่นกัน

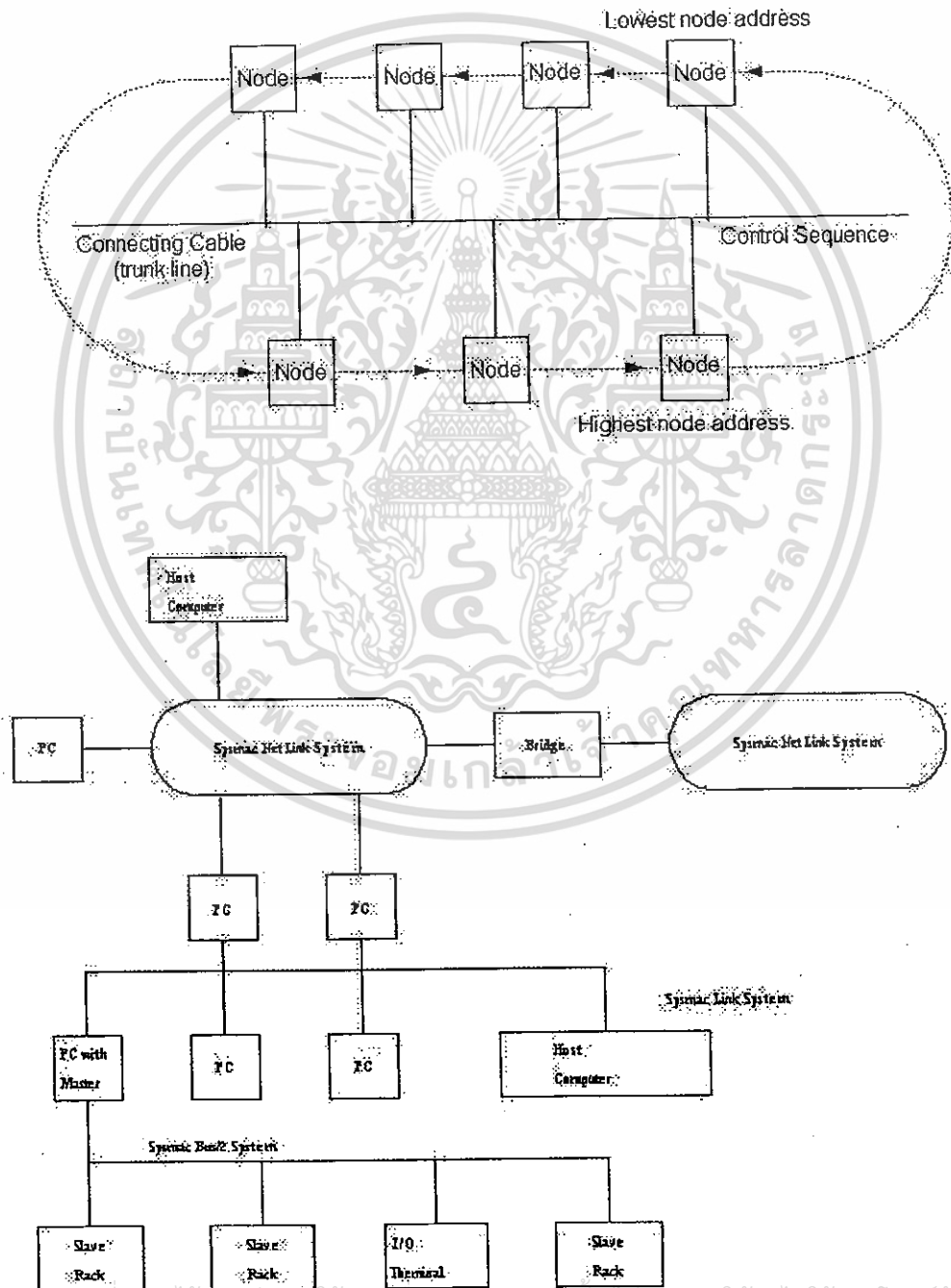
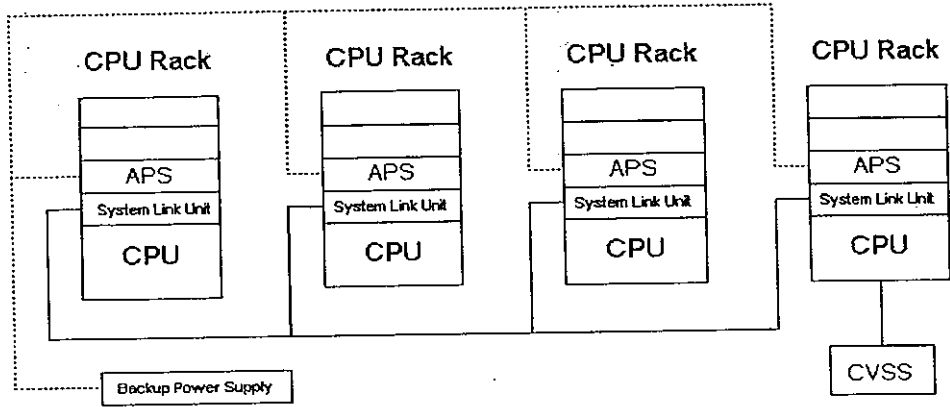
2.3.5 อุปกรณ์สื่อสารข้อมูลในระบบโครงข่ายของ PLC

อุปกรณ์ที่เป็นสื่อสัญญาณ ในการติดต่อสื่อสารในระบบโครงข่ายของ PLC จะประกอบไปด้วยสายสัญญาณ และอุปกรณ์เชื่อมเข้ากับ ระบบโครงข่าย เพื่อในการติดต่อ ในโครงข่าย ตัวนำสัญญาณ หรือสายสัญญาณนี้ต้องมีคุณสมบัติที่ดี สายสัญญาณที่ใช้กันอยู่มีด้วยกันหลายแบบ ในการใช้งาน ในการส่งสัญญาณก็ต้องขึ้นอยู่กับคุณสมบัติของสายสัญญาณด้วย ระบบโครงข่ายอุตสาหกรรม คือระบบโครงข่ายธุรกิจ หรือระบบ LAN (Local Area Network) ที่พัฒนาขึ้น เพื่อใช้กับสื่อสารข้อมูล ของโรงงานอุตสาหกรรม ที่มีสัญญาณรบกวนและต้องการความเร็วในการส่งถ่ายข้อมูลสูงระบบโครงข่ายคือลักษณะการเชื่อมต่อและการติดต่อแลกเปลี่ยนข้อมูลระหว่างจุดเชื่อมต่อต่างๆ ในระบบ โครงข่ายแบ่งออกเป็น

1. โครงข่ายกระจาย (Star Network)
2. โครงข่ายบัสรวม (Common Bus Network)
3. โครงข่ายวงแหวน (Ring Network)
4. โครงข่ายวงแหวนร่วม (Star-Shaped Ring Network)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

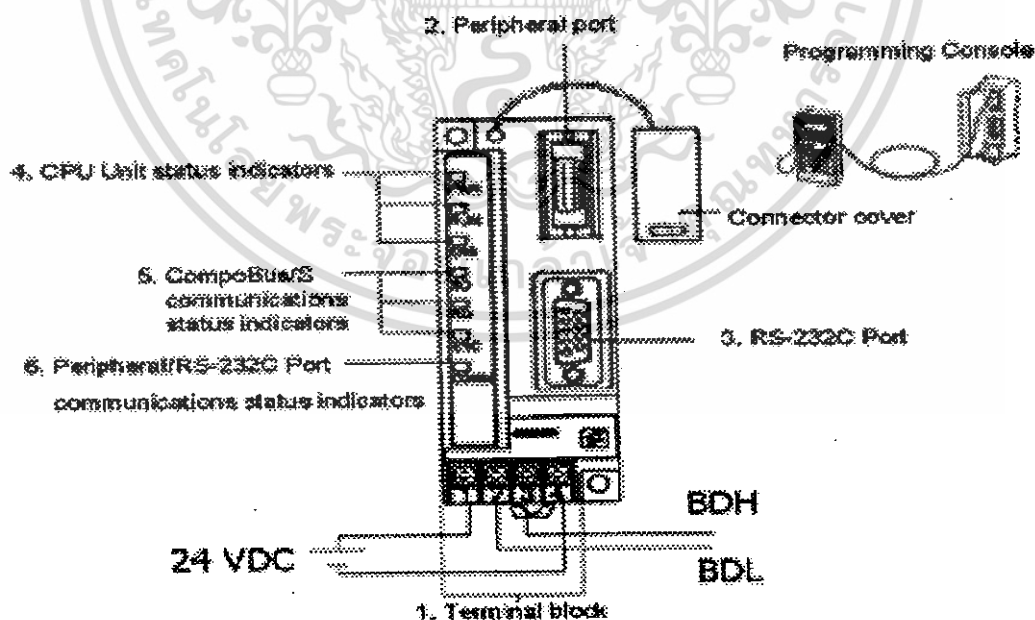
2.3.6 สายเชื่อมต่อระบบโครงข่าย

1. สายคู่ เหมาะสำหรับการติดต่อสื่อสารระหว่างจุดต่อจุด ระยะทางไม่เกิน 4,000 ฟุต อัตราการรับส่งข้อมูลไม่เกิน 250 กิโลบิต/วินาที และจะ ต้องไม่มีสัญญาณรบกวน สายคู่จะมีราคาที่ถูกกว่าสายชนิดอื่น

2. สายโคแอกเชียล สามารถที่จะแบ่งตามความถี่ออกได้อีก คือสายโคแอกเชียลความถี่เดียว และสายโคแอกเชียล หลายความถี่ สายโคแอกเชียล ความถี่เดียวเหมาะสำหรับการติดต่อสื่อสารระหว่างจุดต่อจุดที่มีระยะทางไม่ไกลนัก ส่วนสายสายโคแอกเชียลหลายความถี่ เหมาะสำหรับระยะทาง ที่ไกลๆ การติดต่อสื่อสารที่มีข้อมูลในการติดต่อสื่อสารจำนวนมากๆ สามารถที่จะรับข้อมูลสูงสุด ถึง 150 ล้านบิต/วินาที

3. ใยแสง (Fiber optic) เหมาะสำหรับระบบ โครงข่ายที่มีการรับส่งข้อมูลขนาดใหญ่ ระยะทางไกล และมีสัญญาณรบกวนมากๆ ราคาค่อนข้างแพงกว่าสายชนิดอื่น

2.3.7 PLC (SRM1-CO2-V2)



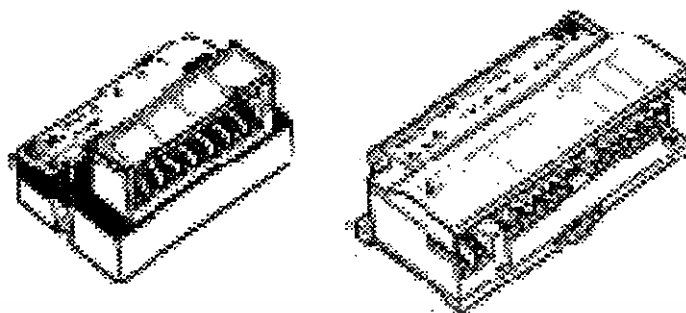
ภาพที่ 2.14 PLC (SRM1-CO2-V2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.Terminal block	เป็นจุดต่อไฟ 24 VDC ที่ช่องที่ 1 กับ 4 ส่วนช่องที่ 2 กับ 3 จะเป็นช่องส่งสัญญาณข้อมูล ซึ่งเป็นแบบ Compobus System	
2.Peripheral Port	เป็นจุดเชื่อมต่อกับอุปกรณ์ที่ใช้ทำการ โปรแกรม(Programming Console) โดยใช้สาย RS-422	
3.RS-232C Port	เป็นจุดที่ใช้เชื่อมต่อกับคอมพิวเตอร์ โดยใช้สาย RS-232C	
4.CPU Unit Status Indicators	เป็นไฟบอกสถานะของ PLC	
PWR (ไฟสีเขียว)	ไฟติด	เมื่อมีการจ่ายไฟให้กับ PLC
	ไฟดับ	เมื่อ ไม่มีการจ่ายไฟให้กับ PLC
RUN (ไฟสีเขียว)	ไฟติด	เมื่ออยู่ใน Mode Run หรือ Mode Monitor
	ไฟดับ	ไฟดับ เมื่ออยู่ใน Mode Program
ERR(ไฟสีแดง)	ไฟติด	เมื่อ PLC อยู่ในสภาวะผิดปกติ
	ไฟดับ	เมื่อ PLC อยู่ในสภาวะปกติ
5.Compobus/S Communications Status Indications	เป็นไฟบอกสถานะการรับส่งข้อมูลของ Compobus	
SD (ไฟสีเขียว)	ไฟติด	เมื่อมีการส่งข้อมูล
	ไฟดับ	เมื่อ ไม่มีการส่งข้อมูล
RD (ไฟสีเขียว)	ไฟติด	เมื่อ ได้รับข้อมูล
	ไฟดับ	เมื่อ ไม่ได้รับข้อมูล
ERC (ไฟสีแดง)	ไฟติด	เมื่อการรับส่งข้อมูลอยู่ในสภาวะผิดปกติ
	ไฟดับ	เมื่อการรับส่งข้อมูลอยู่ในสภาวะปกติ
6.COMM (ไฟสีเขียว)	เป็นไฟบอกสถานะการรับส่งข้อมูลผ่านสายRS-232C	
	ไฟกระพริบ	เมื่อมีการรับหรือส่งข้อมูล ผ่าน สาย RS-232C
	ไฟดับ	เมื่อ ไม่มีการรับหรือส่งข้อมูลผ่านสายRS-232C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Remote I/O



16 Input

16 Output

Name	Number of words or bits	Word addresses
Input bits	8 words	IR 000 to 007
Output bits	8 words	IR 010 to 017
Work bits	44 words	IR 008 and IR 009 IR 018 and IR 019 IR 200 to IR 239

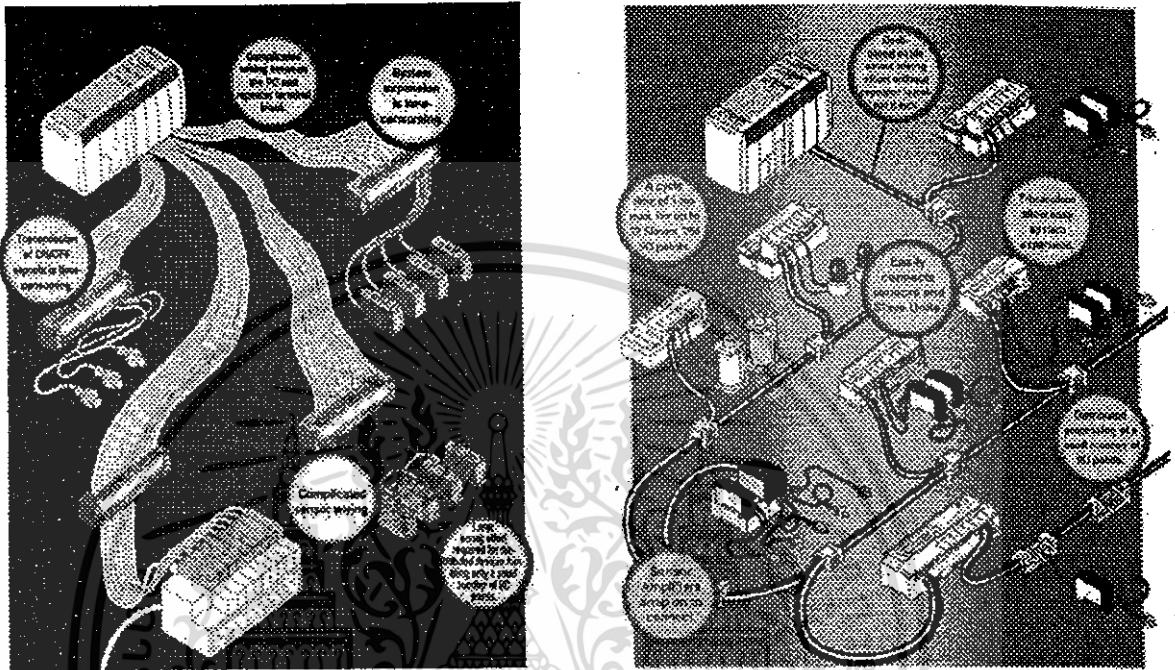
I/O	Word address	Bits	
		15 to 08	07 to 00
Inputs	000	IN 1	IN0
	001	IN 3	IN 2
	002	IN 5	IN 4
	003	IN 7	IN 6
	004	IN 9	IN 8
	005	IN 11	IN 10
	006	IN 13	IN 12
	007	IN 15	IN 14
Output	010	OUT 1	OUT0
	011	OUT 3	OUT 2
	012	OUT 5	OUT 4
	013	OUT 7	OUT 6
	014	OUT 9	OUT 8
	015	OUT 11	OUT 10
	016	OUT 13	OUT 12
	017	OUT 15	OUT 14

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

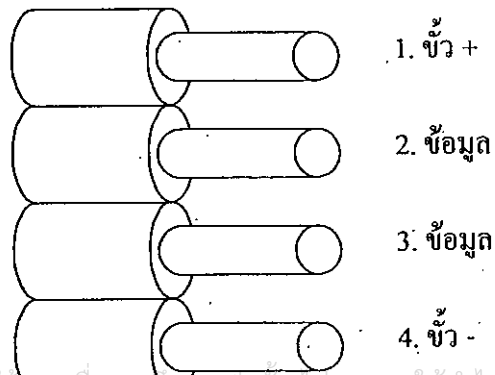
Remote I/O สามารถต่อสายที่ใช้ในการรับส่งสัญญาณข้อมูลไปได้ไกลถึง 100 m มี Input ให้ใช้ถึง 16 Input และ Output ให้ใช้ถึง 16 Output

2.3.8 Compobus System



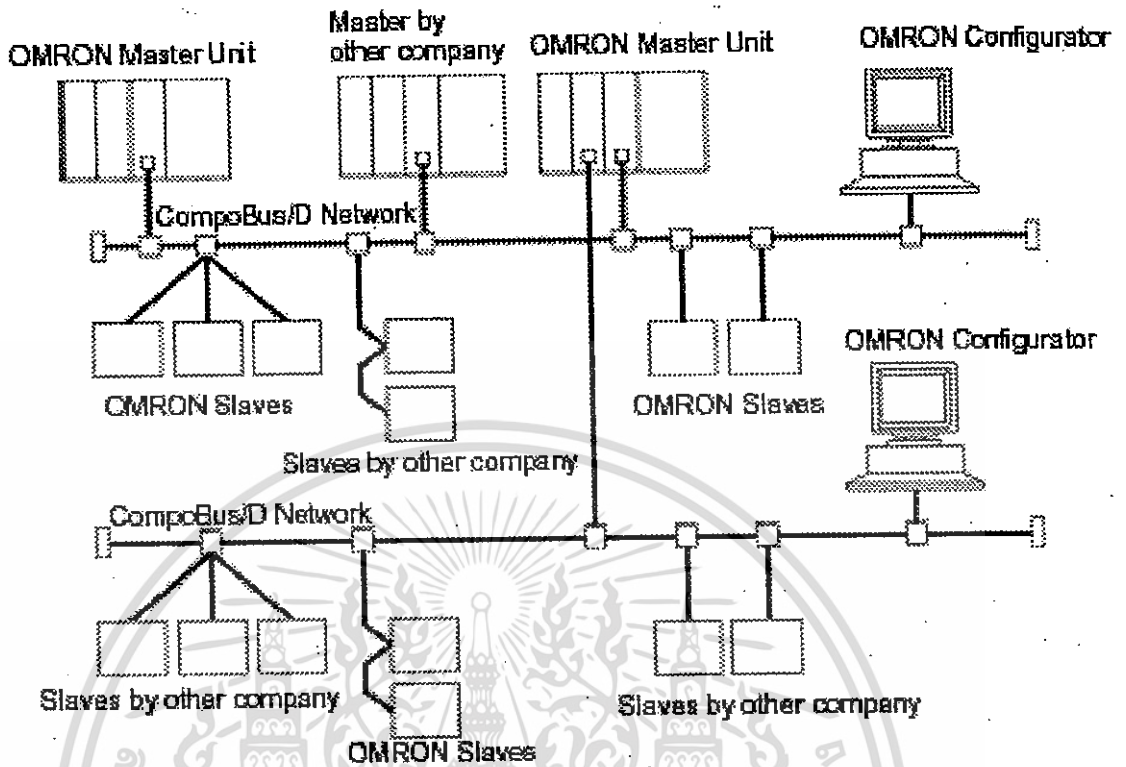
ภาพที่ 2.15 การเปรียบเทียบการเชื่อมต่อระหว่างแบบธรรมดากับการเชื่อมต่อแบบ Compobus System

ประโยชน์ของ Compobus System คือช่วยให้การขยายจุดเชื่อมต่อการรับส่งข้อมูลระหว่างอุปกรณ์ต่างๆ สามารถทำได้ง่ายและสะดวกขึ้น โดยใช้ตัวเชื่อมต่อดังรูปที่ 3 หรือ 4 ซึ่งขนาดของสายที่ใช้จะมีขนาด 0.75 mm² จำนวน 4 สาย โดยสายที่ 1 และ 4 จะเป็นสายสำหรับจ่ายไฟเป็น ขั้ว + และ - ตามลำดับ ส่วนสายที่ 2 และ 3 จะเป็นสายสำหรับ รับส่งข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

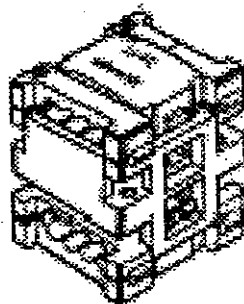
2.3.9 Compobus Network



ภาพที่ 2.16 ตัวอย่างระบบเครือข่ายที่ใช้การเชื่อมต่อแบบ Compobus System

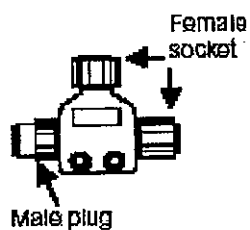
การเชื่อมต่อแบบ Compobus System สามารถเชื่อมต่อกับอุปกรณ์ต่างๆ เช่น ตัว Remote I/O ตัว PLC ที่เป็นตัวลูก และ PLC ที่เป็นตัวแม่เหมือนกัน โดยการสื่อสารรับส่งข้อมูลของ อุปกรณ์ Input และ Output สามารถทำได้พร้อมกับการสื่อสารรับส่งข้อมูลระหว่างตัว PLC ด้วยกัน

2.3.10 Branch Connector หรือ Extension Connector

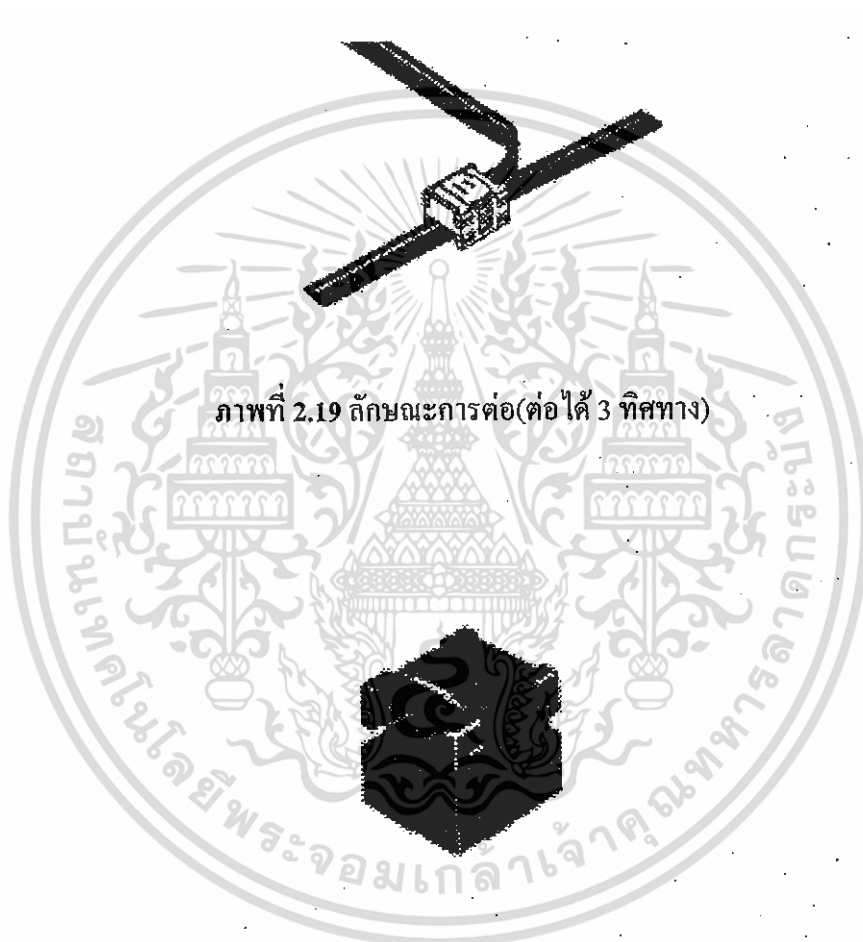


ภาพที่ 2.17 ตัวเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

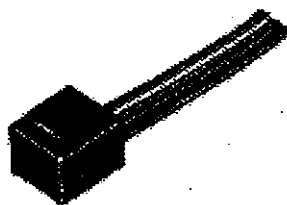


ภาพที่ 2.18 ตัวเชื่อมต่อแบบ Socket



ภาพที่ 2.19 ลักษณะการต่อ(ต่อได้ 3 ทิศทาง)

ภาพที่ 2.20 ตัวเสียบสายเมื่อสิ้นสุดการต่อ (Connector Terminator)



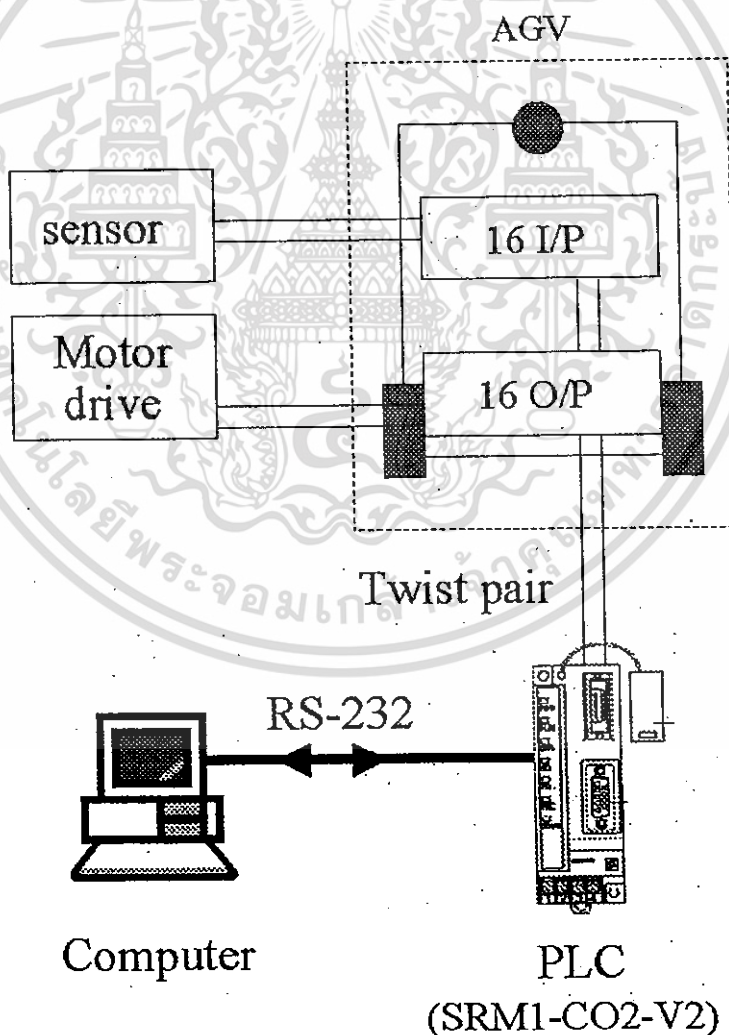
ภาพที่ 2.21 ลักษณะการต่อสายเมื่อสิ้นสุดการต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบและการสร้าง

3.1 โครงสร้างของโครงการ

หลักการดำเนินงานของโครงการนี้ คือการใช้ Computer ซึ่งใช้โปรแกรม Visual Basic เขียนโปรแกรม ควบคุมและสั่งการ ตัวหุ่นยนต์ (AGV) ผ่าน PLC โดยใช้พอร์ตท่อนุกรม RS-232 โดย PLC จะต่อเข้ากับ 16 I/P และ 16 O/P และอุปกรณ์ SENSOR จะต่อเข้ากับ 16 I/P ส่วน Motor จะต่อเข้ากับ 16 O/P ซึ่งสามารถควบคุมให้วิ่งไปยังจุดที่กำหนดไว้ โดยวิ่งตามแถบสีบนพื้น โดย AGV จะเลือกเส้นทางที่ใช้เวลาน้อยที่สุด หลังจากที่ได้ทำการเรียนรู้เส้นทางทุกเส้นทางที่จะไปยังจุดที่กำหนดไว้แล้ว เพื่อยกวัสดุหรือสิ่งของไปไว้ยังจุดเดิมหรือจุดอื่นๆ ที่ต้องการได้

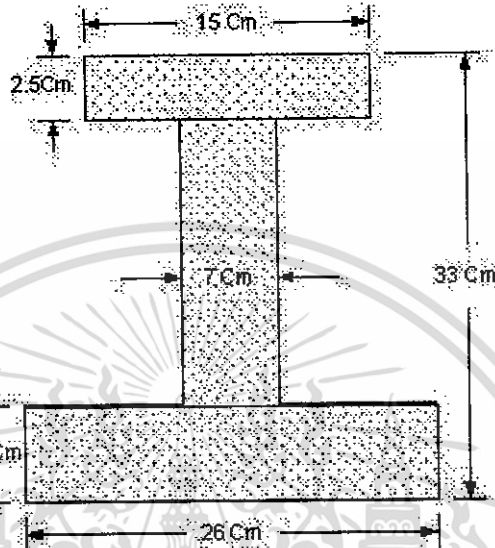


ภาพที่ 3.1 แสดงการเชื่อมต่อระหว่างอุปกรณ์ที่ใช้ในโครงการ

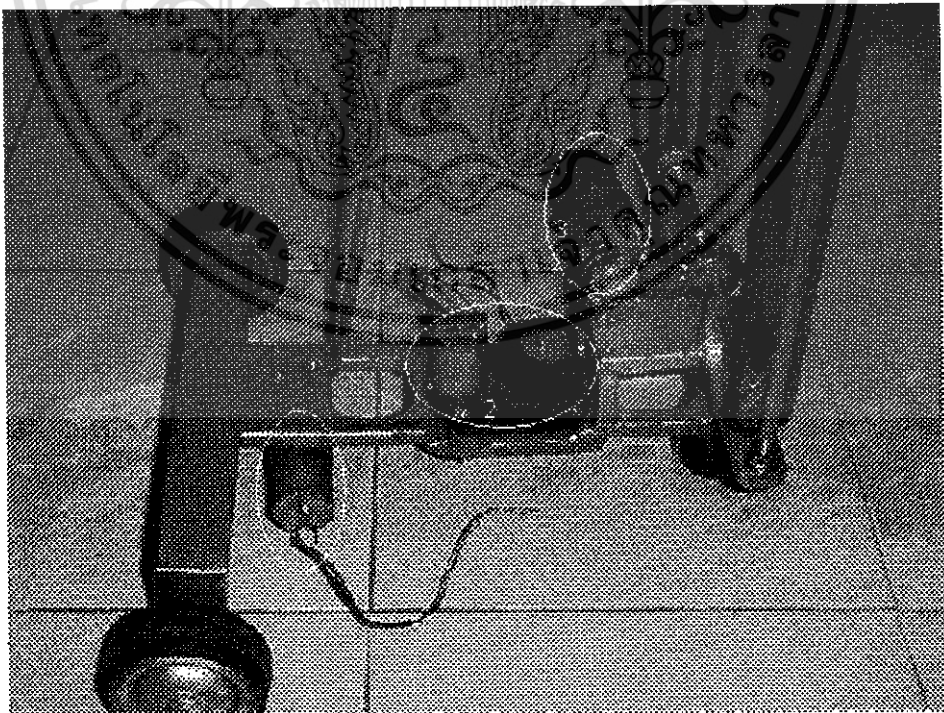
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบตัวรอด

การออกแบบตัวรอดที่ใช้ในโครงการ ตัวโครงสร้างจะทำมาจากเหล็ก เพื่อที่จะมีความแข็งแรงและทนทาน ซึ่งจะเป็นดังรูปที่ 3.2 ข้างล่างนี้ ซึ่งจะมีความยาวด้านหน้า 15 เซนติเมตร กว้าง 2.5 เซนติเมตร ด้านหลัง 26 เซนติเมตร กว้าง 3.5 เซนติเมตร ความยาวของตัวรอด 33 เซนติเมตร กว้าง 7 เซนติเมตร

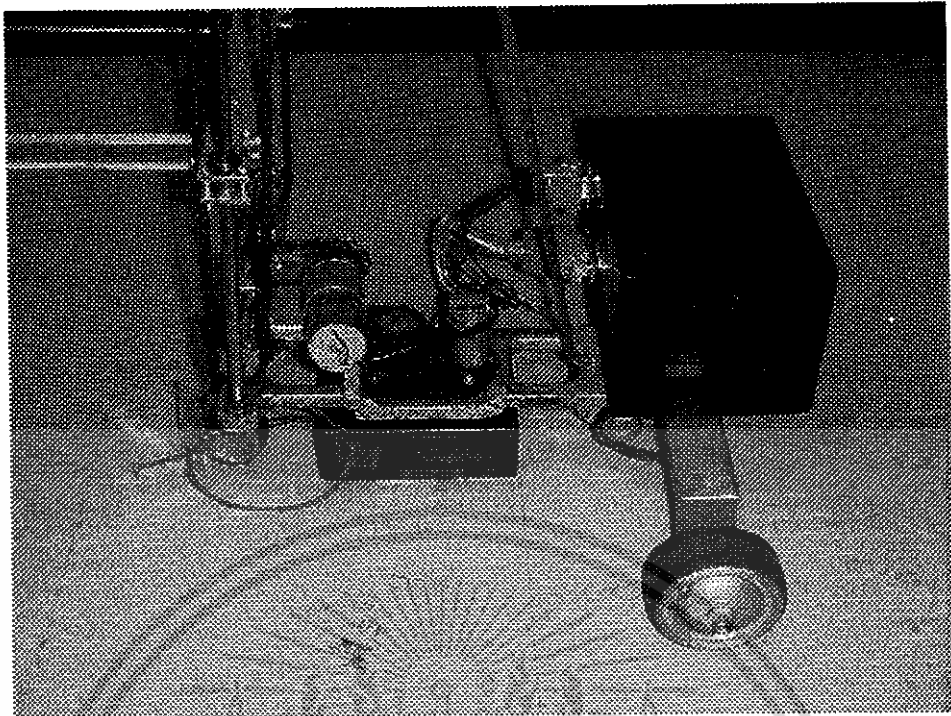


ภาพที่ 3.2 แสดงตัวรอดด้าน Top View



ภาพที่ 3.3 โครงสร้างและการวางมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)



(ข)

ภาพที่ 3.4 แสดงลักษณะของตัวรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

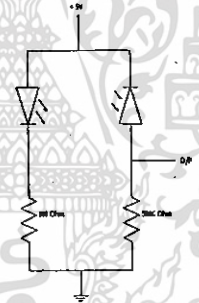
ลักษณะการทำงานของมอเตอร์

มอเตอร์ ใช้มอเตอร์ไฟฟ้ากระแสตรง 24 โวลต์ 3 คิว ซึ่งมีหน้าที่ต่างกันดังนี้
 ตัวที่ 1 ทำหน้าที่ขับเคลื่อน ซึ่งเป็นการทำงานขับเคลื่อนล้อหลัง
 ตัวที่ 2 ทำหน้าที่ควบคุมการเลี้ยว โดยใช้ล้อหน้า
 ตัวที่ 3 ทำหน้าที่ควบคุมชุดยกที่ใช้ในการยกของ

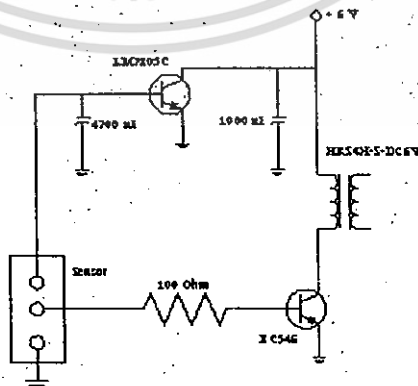
3.3 การควบคุม AGV

3.3.1 sensor ในการตรวจสอบเส้นทาง

Sensor ที่ใช้ในการตรวจจับเส้นทางของ AGV เราจะใช้ตัว Infrared ที่มีลักษณะที่เป็นชิปเดียวกัน โดยใช้หลักการของโฟโตไดโอด ที่ตัวหนึ่งจะทำหน้าที่ปล่อยรังสี อินฟราเรดออกมาเมื่อมีการกระทบกับพื้นผิวของวัตถุ ก็จะมีการสะท้อนของรังสีอินฟราเรดที่ถูกส่งออกไปยังตัวรับ และจะทำการเปลี่ยนความเข้มแสงให้เป็นแรงดันไฟฟ้า ในการตรวจสอบเส้นทางโดยอาศัยการสะท้อนของแสงกับแถบสีขาวในการทำงานของตัว Infrared ดังจะแสดงในภาพที่ 3.5



ภาพที่ 3.5 แสดงวงจรของ sensor



ภาพที่ 3.6 แสดงวงจรของ sensor ที่จะนำไปต่อเข้ากับ PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะการทำงานของวงจรที่ใช้ในตรวจสอบเส้นทางในการเคลื่อนที่ของตัวAGV โดยเมื่อเราจ่ายไฟฟ้ากระแสตรง 5 โวลต์ เข้าสู่วงจร และเมื่อไม่มีแถบสีขาวเข้ามาปิดกั้นขวางทางของลำแสงที่ส่งออกมาจากตัวส่งของ Infrared และตัวรับก็จะยังคงไม่ทำงานก็จะเหมือนกับเปิดวงจรที่ตัวรับและแรงดันที่ตร้อมความต้านทานก็จะยัง คงเป็น 0 V และเมื่อมีแถบสีขาวเข้ามาขวางทางตัวส่งก็จะทำให้เกิดการสะท้อนของแถบสีขาวไปยังตัวรับของ Infrared และจะทำให้ตัวรับทำงานก็จะทำให้เกิดแรงดันตกคร่อมที่ตัวความต้านทานหรือมีแรงดันที่ Output 5V นั้นเอง

ดังนั้นเราก็จะใช้หลักการนี้ในการตรวจสอบเส้นทางของตัวAGV โดยเราจะใช้ เซ็นต์เซอร์ ตัวที่ 1 ในการตรวจจับเส้นทาง และตัวที่ 2-5 ใช้ในการตรวจสอบจุดส่งของ หรือ ทางแยก และเราจะนำเอา Output ของ ตัวอินฟราเรดเข้าไปต่อกับตัว PLC. เพื่อให้ PLC ตรวจสอบสถานะของตัว AGV ว่าอยู่ในสถานะใดและจะส่งค่าที่ได้ไปยังคอมพิวเตอร์เพื่อให้คอมพิวเตอร์เป็นตัวตัดสินใจในการทำงานและส่งค่าในการตัดสินใจไปยัง PLC เพื่อไปสั่งการให้ กับตัวมอเตอร์อีกทีหนึ่ง



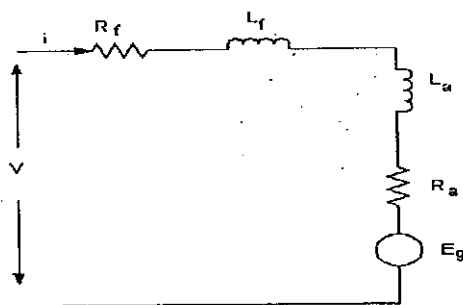
ภาพที่ 3.7 แสดงการจัดวาง sensor ในการตรวจจับเส้นทางและแยก

3.3.2 การควบคุมการเคลื่อน

ในการเคลื่อนของตัว AGV เราใช้มอเตอร์ไฟฟ้ากระแสตรงเพื่อใช้ในการเคลื่อนมอเตอร์ไฟฟ้ากระแสตรงแบบอนุกรม(Series)

มอเตอร์ไฟฟ้ากระแสตรงแบบอนุกรม ขดลวดอาร์เมเจอร์และขดลวดฟิลด์ จะต่ออนุกรมกัน และต่อเข้ากับแหล่งจ่ายเพียงชุดเดียวดังภาพที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.8 โครงสร้างทางไฟฟ้าของมอเตอร์ไฟฟ้ากระแสตรงแบบอนุกรม

จากวงจรไฟฟ้าในภาพที่ 3.7 เราจะได้สมการความเร็วของมอเตอร์ที่ใช้ในการควบคุมการเคลื่อนของตัว AGV ดังนี้คือ

$$\omega = V - \frac{R_f + R_a I}{K_v I} \quad (3.1)$$

- ω คือ อัตราความเร็วเชิงมุม
 V คือ ค่าแรงดันที่จ่ายให้กับตัวมอเตอร์ (โวลต์)
 I คือ กระแสของตัวมอเตอร์ (แอมแปร์)
 K_v คือ ค่าคงที่ของอัตราเร็ว (โวลต์-นาทึ/แอมแปร์-รอบ)
 R_f ค่าความต้านทานของขดฟิลด์ (โอห์ม)
 R_a คือค่าความต้านทานของขดอาร์มเจอร์ (โอห์ม)

หรือความเร็วของมอเตอร์เป็นดังสมการที่ 3.2

$$N = \frac{60}{2\pi} \left(V - \frac{R_f + R_a I}{K_v I} \right) \quad (3.2)$$

3.3.3 การควบคุมความเร็วของมอเตอร์

มอเตอร์ไฟฟ้ากระแสตรงสามารถควบคุมความเร็วได้โดยการควบคุมแรงดัน และ กระแสไฟฟ้าที่ป้อนให้กับมอเตอร์

การปรับลดแรงดันที่จ่ายให้กับตัวมอเตอร์ให้มีค่าลดลงจะส่งผลทำให้ค่าความเร็วของตัวมอเตอร์ลดลงดังสมการข้างต้นและเมื่อเรานำตัวต้านทานที่ปรับค่าได้ต่ออนุกรมเข้ากับวงจร โดยการ

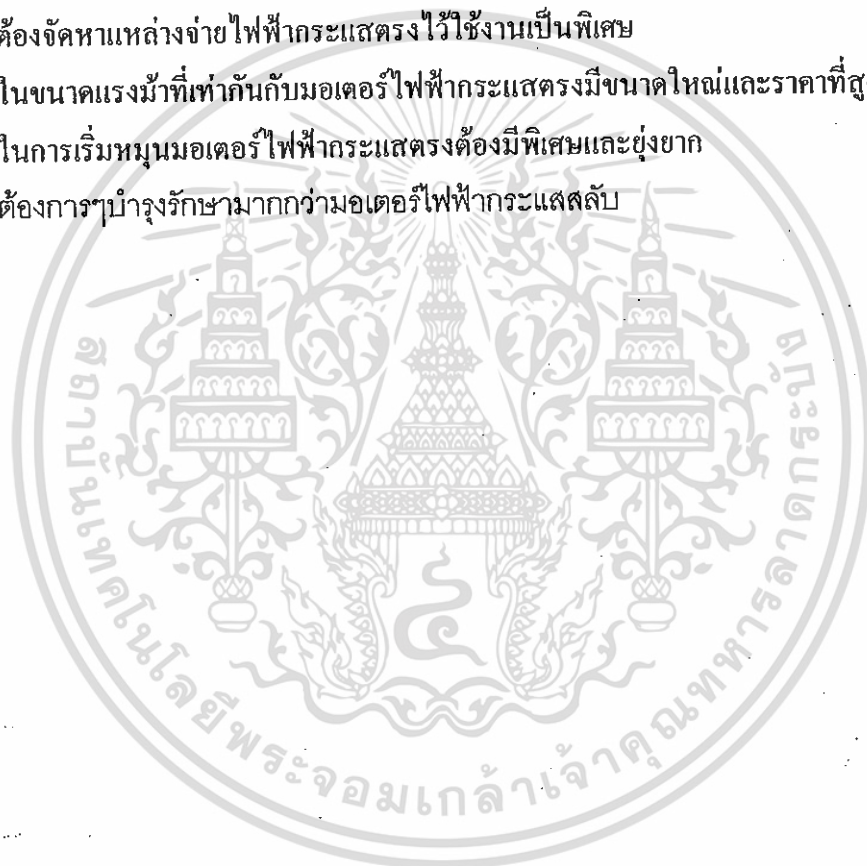
เพิ่มค่าแรงดันให้สูงขึ้นและเมื่อเราจ่ายแรงดันคงที่ให้กับวงจรที่มีค่าความต้านทานในวงจรเพิ่มขึ้นก็จะทำให้ค่ากระแสของวงจรลดลงซึ่งจะส่งผลให้ค่าความเร็วของตัวมอเตอร์มีค่าความเร็วลดลง

ข้อดีของมอเตอร์ไฟฟ้ากระแสตรง

- 1 มีพิสัยของความเร็วกว้างมาก ซึ่งสามารถเพิ่มความเร็วให้สูงหรือต่ำกว่าความเร็วปกติได้
- 2 มีแรงบิดขณะเริ่มหมุนสูงซึ่งเหมาะกับงานยกของ นุด ลาก และขับเคลื่อน
- 3 มีวิธีการควบคุมความเร็วที่ง่าย

ข้อเสียของมอเตอร์ไฟฟ้ากระแสตรง

- 1 ต้องจัดหาแหล่งจ่ายไฟฟ้ากระแสตรงไว้ใช้งานเป็นพิเศษ
- 2 ในขนาดแรงม้าที่เท่ากับมอเตอร์ไฟฟ้ากระแสตรงมีขนาดใหญ่และราคาที่สูงกว่า
- 3 ในการเริ่มหมุนมอเตอร์ไฟฟ้ากระแสตรงต้องมีพิเศษและยุ่งยาก
- 4 ต้องการบำรุงรักษามากกว่ามอเตอร์ไฟฟ้ากระแสสลับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

การหาระยะห่างที่เหมาะสม ระหว่างตัว อินฟาราด กับ พื้น เป็นสิ่งจำเป็นในการใช้ควบคุม ตัว AGV เพราะจะส่งผลถึงความถูกต้องและความสามารถของตัว AGV เอง ดังนั้นเราต้องมีการ ตรวจสอบและหาค่า ที่จะใช้ให้เหมาะสมมากที่สุด

จากภาพที่ 3.5 และ 3.6 เราจึงนำมาตรวจสอบที่ระยะต่างๆโดยใช้แถบสีขาว และ ดำ ในการ ตรวจสอบ ซึ่งระยะและค่าที่แรงดันที่ได้ในระยะต่างๆเป็น ไปดังตารางที่ 4.1

ตารางที่ 4.1 การทดลองตัว Infrared ที่ระยะความห่างต่างๆ

ระยะห่าง (Cm)	สีขาว(V)	สีดำ(V)
0.5	4.93	0.34
1.0	3.5-1.7	0.16
1.5	1.9	0.09
2.0	1.39	0.08

สรุปผลการทดลอง

จากการทดลองตัว อินฟาราด จะพบว่าที่ระยะห่างต่างๆของตัวอินฟาราด กับแถบสีที่ใช้ใน การควบคุมเส้นทางเดินจะได้ว่าที่ระยะห่าง 0.5 เซนติเมตร จะได้ค่าแรงดันค่อนข้างจะคงที่ และ เที่ยงตรง มากที่สุดเพื่อที่จะใช้ในการเป็น อินพุท ให้กับตัว AGV

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 ข้อจำกัดของโครงการ

จากโครงการที่ได้ทำการศึกษา เกี่ยวกับ การควบคุมการทำงานของ AGV โดยใช้โปรแกรม Visual Basic ในการควบคุม และ ตั้งการผ่าน PLC ผ่านพอร์ตอนุกรม ซึ่งมีข้อจำกัด ทั้ง ขนาดของ ตัว AGV การจัดวางอุปกรณ์ การเคลื่อนของตัว AGV ซึ่งต้องทำการแก้ไขในหลายจุด

5.2 แนวทางในการพัฒนา

จากการได้ศึกษาลักษณะการทำงานของ ตัว AGV นี้ เราสามารถนำมาพัฒนาในส่วนของ การควบคุมการเคลื่อน โดยการใช้ Step Motor ซึ่งสามารถควบคุมการเคลื่อนได้ดี โดยอาจจะควบคุม การเคลื่อนเป็นองศาในการเคลื่อนแต่ละที ซึ่งสามารถควบคุมได้ง่าย และแม่นยำกว่า รวมไปถึงการใช้ Sensor ให้มีระยะที่ไกลขึ้น เพื่อที่จะสามารถนำ AGV นี้ไปใช้ในบริเวณที่มีพื้นผิวที่ขรุขระได้

5.3 บทสรุป

ในปัจจุบันหุ่นยนต์ได้เข้ามามีบทบาทอย่างมากในงานอุตสาหกรรม ดังนั้นการที่เรา สามารถพัฒนาหุ่นยนต์ขึ้นมาใช้ในโรงงาน หรือ ในสภาพที่คนไม่สามารถเข้าไปปฏิบัติงานได้ โดย ให้หุ่นยนต์ทำงานแทน นอกจากจะเป็นประโยชน์แล้ว ยังลดความเสี่ยงต่ออันตราย จะเกิดกับคน รวมทั้งลดค่าใช้จ่ายต่างๆในระยะยาว ซึ่งสามารถประหยัดกว่าที่จะใช้คนทำงานอีกด้วย

บรรณานุกรม

นาย ไกรสร อัญชลีวรพันธ์ , วิทยานิพนธ์ ยานนำส่งอัตโนมัติ , ปริญญาโทภาควิชาวิศวกรรมการควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วุฒิชัย ปอตระกูล , ศุภฤกษ์ ตปะนียกุล , สมศักดิ์ เอื้อประชาชนนท์ , “ปริญญาทิพนธ์ หุ่นยนต์สำรวจเคลื่อนที่ควบคุมการทำงานด้วยไมโครคอมพิวเตอร์ (Surveying Mobile Robotic Controlled By Microcomputer) ” , ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

คอนสัน ปงผาบ , “การเขียนโปรแกรมในงานควบคุม” , สำนักพิมพ์ ส.ส.ท.

A Practical Guide to Neural Nets

CompoBus/S SRM1(-V2) Master Control Units Operation Manual

<http://www.biocompsystems.com>