บอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนาน

STEPPING MOTOR CONTROL BOARD VIA PARALLEL PORT

โดย
นางสาวนพวรรณ์ ตัญเจริญสุขจิต
นางสาวนฤมล ใจสงเคราะห์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

บอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนาน

STEPPING MOTOR CONTROL BOARD VIA PARALLEL PORT

โดย
นางสาวนพวรรณ์ ตัญเจริญสุขจิต
นางสาวนฤมล ใจสงเคราะห์

อาจารย์ที่ปรึกษา
ผู้ช่วยศาสตราจารย์ วรพงศ์ ตั้งศรีรัตน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง บอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนาน

(STEPPING MOTOR CONTROL BOARD VIA PARALLEL PORT)

ผู้จัดทำ

1. นางสาวนพวรรณ์ ตัญเจริญสุขจิต
2. นางสาวนฤมล ใจสงเคราะห์

....อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์ วรพงศ์ ตั้งศรีรัตน์)

บอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนาน

นางสาวนพวรรณ์ ตัญเจริญสุขจิต
นางสาวนฤมล ใจสงเคราะห์
ผู้ช่วยศาสตราจารย์ วรพงศ์ ตั้งศรีรัตน์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2544

บทคัดย่อ

โครงงานนี้นำเสนอเทคนิคการออกแบบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนานของเครื่องคอมพิวเตอร์และการนำไปประยุกต์ใช้ในการควบคุมมู่ลี่อัตโนมัติ โดยมีจุดมุ่งหมายในการออกแบบสร้างบอร์ดเชื่อมต่อกับคอมพิวเตอร์ผ่านทางพอร์ตขนานสำหรับควบคุมสเต็ปปิ้งมอเตอร์ให้มีการทำงานอย่างอิสระโดยการเขียนโปรแกรมควบคุม บอร์ดที่นำเสนอสามารถควบคุมสเต็ปปิ้งมอเตอร์ได้พร้อมกัน 2 ตัวและสามารถต่อขยายบอรดควบคุมได้สูงสุด 4 บอร์ด นอกจากนี้ได้เสนอการประยุกต์ใช้งานในการนำไปควบคุมมู่ลี่เพื่อให้มีการปรับปริมาณแสงที่เข้ามาในสำนักงานหรือบ้านพักได้อย่างอัตโนมัติ

STEPPING MOTOR CONTROL BOARD VIA PARALLEL PORT

Noppawan Tunjaraunsugjit

Narumon Jaisongkror

Assistant Professor Worapong Tangsrirat Advisor

2001

Abstract

This project proposes a design technique for implementing a stepping motor control board via parallel port. The goal of this project is to propose the design and implement parallel interfacing board for independently stepping motors through software programming. The proposed interfacing board can be controlled two stepping motors simultaneously, and can also be extended to maximum of 4 boards. As an application example, an automationally controllable blind is also included.

สารบัญ

สารบัญ(ต่อ)

สารบัญรูปภาพ

สารบัญรูปภาพ(ต่อ)

หน้า

สารบัญตาราง

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

ในปัจจุบันการควบคุมการทำงานแบบอัตโนมัติเข้ามามีบทบาทสำคัญในกระบวนการผลิตและอุตสาหกรรมต่าง ๆ อย่างมากมาย องค์ประกอบสำคัญในระบบควบคุมแบบนี้ คือ คอมพิวเตอร์ โดยต้องยอมรับว่าศักยภาพของคอมพิวเตอร์นั้นมีประโยชน์มากมายต่อการทำงานต่าง ๆ และสามารถคอมพิวเตอร์ถูกนำมาใช้ในงานหลาย ๆ ด้านและงานที่น่าสนใจมากอย่างหนึ่ง คือ เครื่องจักรกลทั้งหลายที่ปัจจุบันถูกนำไปใช้อย่างแพร่หลาย เนื่องจากมีประสิทธิภาพและมีความแม่นยำสูง ในการใช้งานเครื่องจักรที่ต้องการกำหนดตำแหน่ง ทิศทางหรือความเร็วที่แน่นอน ซึ่งมอเตอร์ที่มักนิยมนำมาใช้ในงานควบคุม คือ สเต็ปปิ้งมอเตอร์ ซึ่งถ้าหากสามารถออกแบบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ที่มีประสิทธิภาพก็จะมีความคล่องตัวสูงได้แล้ว สามารถประยุกต์การใช้ควบคุมการทำงานของสเต็ปปิ้งมอเตอร์ได้ง่ายและมีประสิทธิภาพมากขึ้น ปริญญานิพนธ์ฉบับนี้แสดงถึงการประยุกต์ใช้งานบอร์ดควบคุมสเต็ปปิ้งมอเตอร์เพื่อควบคุมการทำงานของมูลี่อัตโนมัติเนื่องจากเล็งเห็นว่ามีประโยชน์ สามารถนำมาใช้งานได้จริงและมีส่วนช่วยในการประหยัดไฟฟ้า

1.2 วัตถุประสงค์ของการดำเนินการ

1. เพื่อศึกษาการควบคุมสเต็ปปิ้งมอเตอร์โดยผ่านทางพอร์ตขนานของเครื่องคอมพิวเตอร์ส่วนบุคคล (PC)
2. เพื่อแสดงการควบคุมการรับส่งค่าได้จริงกับบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางหน้าจอคอมพิวเตอร์
3. สามารถนำบอร์ดควบคุมไปประยุกต์ใช้งานในการควบคุมการทำงานของสเต็ปปิ้งมอเตอร์ในระบบควบคุมต่าง ๆ ผ่านทางคอมพิวเตอร์ได้

1.3 ขอบเขตการทำงาน

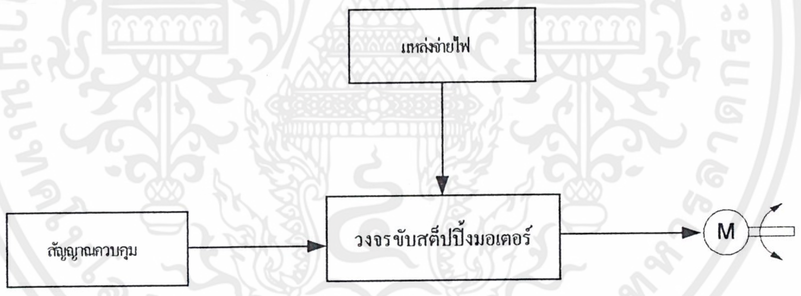
1. สร้างบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ในการควบคุมการทำงานผ่านทางพอร์ตขนานของเครื่องคอมพิวเตอร์ได้
2. สามารถควบคุมสเต็ปปิ้งมอเตอร์ได้ทั้งแบบไบโพลาร์และยูนิโพลาร์
3. สามารถขับสเต็ปปิ้งมอเตอร์ได้ 3 รูปแบบ คือ แบบหนึ่งเฟส , 2 เฟส และครึ่งสเต็ป
4. สามารถควบคุมสเต็ปปิ้งมอเตอร์พร้อมกัน 2 ตัวได้ในบอร์ดเดียวกัน

5. สามารถต่อขยายบอร์ดควบคุมได้ 4 บอร์ด ทำให้สามารถควบคุมมอเตอร์ได้ทั้งหมด 8 ตัว
6. ออกแบบวงจรและแสดงการทำงานเพื่อควบคุมมู่ลี่อัตโนมัติได้
7. สามารถเขียนโปรแกรมควบคุมและแสดงผลผ่านหน้าจอคอมพิวเตอร์ได้

บทที่ 2
หลักการทำงานและทฤษฎีพื้นฐาน

2.1 สเต็ปปิ้งมอเตอร์

สเต็ปปิ้งมอเตอร์เป็นอุปกรณ์ที่เปลี่ยนพลังงานไฟฟ้าเป็นพลังงานกล โดยพลังงานทางไฟฟ้าที่ป้อนให้จะมีลักษณะเป็นไบนารีโวลต์เตจและเอาต์พุตมีลักษณะเป็นการเคลื่อนที่เชิงมุม คือ เมื่อป้อนสัญญาณพัลซ์ด้วยความถี่และลำดับที่เหมาะสมให้กับขดลวดสเตเตอร์ มอเตอร์จะหมุนเป็นจังหวะตามพัลซ์ที่ป้อนเข้ามา ซึ่งต่างจากมอเตอร์ทั่วไปที่จะหมุนทันทีและตลอดเวลาเมื่อป้อนแรงดันไฟฟ้า ข้อดีของสเต็ปปิ้งมอเตอร์ คือ สามารถกำหนดตำแหน่งของการหมุนด้วยตัวเลข (องศาหรือระยะทาง) ได้อย่างละเอียด โดยใช้คอมพิวเตอร์เป็นเครื่องกำหนดและจัดเก็บตัวเลขโดยไม่ค่าตำแหน่งผิดพลาด (Position Error) สะสม นอกจากนี้ยังไม่จำเป็นต้องมีการป้อนกลับของตำแหน่งหรือความเร็ว



รูปที่ 2.1 บล็อคไดอะแกรมแสดงการควบคุมสเต็ปปิ้งมอเตอร์

2.1.1 โครงสร้างและหลักการทำงาน

โครงสร้างของขั้วแม่เหล็กบนสเตเตอร์ (Stator) ทำมาจากแผ่นเหล็กวงแหวนที่มีซี่ยื่นออกมาประกอบกันเป็นชั้น ๆ โดยที่แต่ละซี่นั้นมีขดลวดพันสวมอยู่ เมื่อมีการป้อนกระแสผ่านขดลวดสเตเตอร์หลายขดที่ต่ออนุกรมกันซึ่งเรียกว่าเฟส จะทำให้เกิดสนามแม่เหล็กไฟฟ้าในแอร์แกป (air gap) และแรงบิด (Torque) เกิดแรงผลักต่อโรเตอร์ทำให้มอเตอร์หมุนได้ โรเตอร์จะหมุนด้วยมุมที่คงที่ (Step Angle) ตำแหน่งมุมของโรเตอร์สามารถควบคุมได้โดยใช้วิธีสวิทช์ซิ่งเรียงกันไปเป็นลำดับ และถ้าเพิ่มจำนวนของขั้วแม่เหล็กมากขึ้นจะเพิ่มจำนวนของสเต็ปต่อวงจรรอบมากขึ้นตามด้วย

แท่งแม่เหล็กถาวรติดอยู่กับเพลาและหมุนได้อิสระเหมือนอาร์เมเจอร์ มีขั้วแม่เหล็ก 2 ขั้ว ซึ่งเป็นส่วนหนึ่งของโครงโลหะที่เป็นสเตเตอร์ ตำแหน่งแกนของอาร์เมเจอร์แม่เหล็กคือ a-á ซึ่งต่างไปจากแกนขั้วของแม่เหล็กไฟฟ้าเล็กน้อยเป็นมุม θ แรงแม่เหล็กที่เกิดจากการดึงดูดของขั้วแม่เหล็กที่ต่างกันทำให้เกิดแรงบิดขึ้น

แรงแม่เหล็กที่เกิดจากการดึงดูดของขั้วแม่เหล็กที่ต่างกันทำให้เกิดแรงตั้งฉากกับแกน a- á

$$F_n = F\cos\theta \tag{2.1}$$

แรงบิดรวม

$$T = F_n.R \tag{2.2}$$

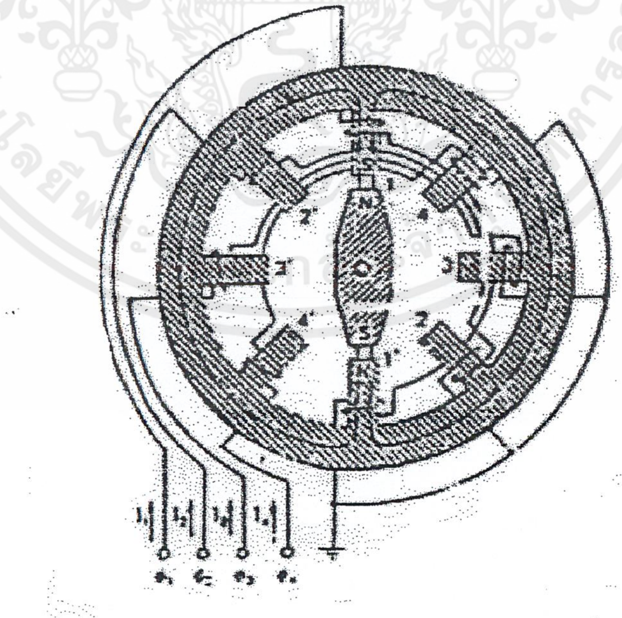
แรงบิดนี้ทำให้อาร์เมเจอร์หมุนไปในทิศทางตามเข็มนาฬิกาจนกว่าแกนของอาเมเจอร์ a-á จะอยู่ในแนวเดียวกับแกนขั้วของสเตเตอร์

รูปที่ 2.2 แรงดึงดูดที่ทำให้เกิดแรงบิดที่หมุนอาร์เมเจอร์ให้ไปอยู่ในตำแหน่งสมดุลย์

ถ้าหากมีคู่ขั้วแม่เหล็กไฟฟ้าหลาย ๆ คู่ขั้วรอบ ๆ สเตเตอร์และหากขั้วเหล่านั้นถูกกระตุ้นด้วยกระแสพัลซ์ในรูปแบบที่เรียงลำดับกันไป อาร์เมเจอร์ก็จะหมุนในรูปลักษณะของสเต็ปที่เป็นไปตามการหมุนของสนามแม่เหล็กที่เกิดจากการสวิตซ์ที่เรียงลำดับของขดลวดขั้วแม่เหล็กไฟฟ้าของสเตเตอร์

2.1.2 ชนิดของสเต็ปปิ้งมอเตอร์

1. สเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร (Permanent Magnet : PM) มีโครงสร้างของโรเตอร์เรียบ ไม่มีซี่ขั้วแม่เหล็กแต่เป็นแม่เหล็กถาวรแทน ควบคุมโดยการป้อนกระแสเข้าที่ขดลวดที่สเตเตอร์ ทำให้เกิดแรงยึดเหนี่ยวระหว่างแม่เหล็กถาวรที่โรเตอร์และขดลวดที่สเตเตอร์แม้จะเลิกจ่ายไฟแล้วก็ตาม จากรูปที่ 2.3 เป็นสเต็ปปิ้งมอเตอร์แบบ 4 เฟส แต่ละเฟสเป็นขดลวดอยู่บน 2 ขั้วของสเตเตอร์ ดังนั้นจึงมีสเตเตอร์ 8 ขั้ว โรเตอร์ทำจากแม่เหล็กถาวรและอยู่ในแนวของขั้วสเตเตอร์ 1 และ 1′ และหยุดที่อยู่ที่ตำแหน่งนี้ได้ด้วยกระแส I_1 ที่ไหลอยู่ในเฟส I ขดลวดของเฟส ϕ_1, ϕ_4, ϕ_3 และ ϕ_2 จะได้รับพลังงานด้วยกระแสพัลซ์ที่สอดคล้องกับ I_1, I_4, I_3 และ I_2 แต่ละสเต็ปโรเตอร์จะหมุนไปตามทิศทางตามเข็มนาฬิกา 45° (360/8) เมื่อขั้วของเหนือของโรเตอร์ (แม่เหล็กถาวร) หมุนไปถึงขั้วของสเตเตอร์หมายเลข 2 ลำดับการขับขดลวดเฟสของสต็ปปิ้งมอเตอร์คือ 1-4-3-2 จะต้องกระทำเหมือนเดิม (เพื่อให้มอเตอร์หมุนไปตามเข็มนาฬิกาอีก 180°) ถ้าต้องการให้หมุนกลับทิศทางใน 180° ที่เหลือต้องป้อนกระแสกลับทิศทางเพื่อให้เกิดการเหนี่ยวนำเป็นขั้วใต้ที่ขั้วสเตเตอร์ 1′, 4′, 3′และ 2′ ตามลำดับ



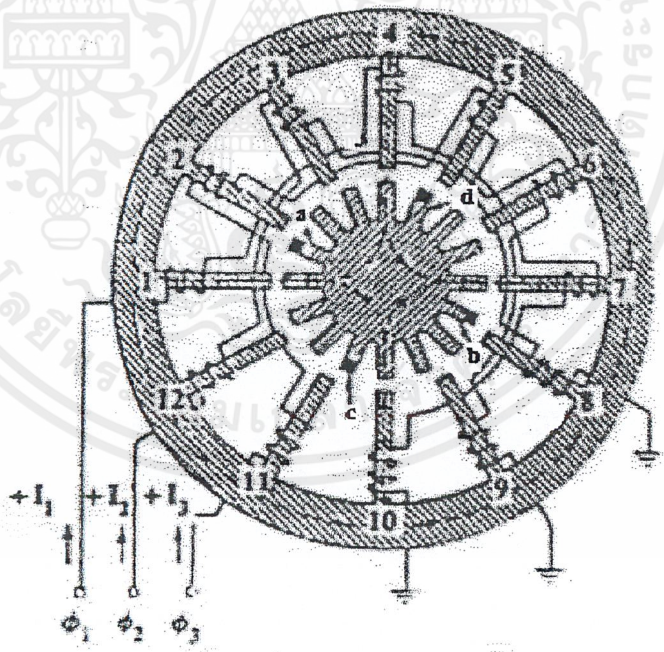
รูปที่ 2.3 โครงสร้างของสเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร

2. สเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แปรค่า(Variable-Reluctance : VR) โรเตอร์ ทำจากเหล็กอ่อน เมื่อมีการหมุนจะไม่เกิดปรากฏการณ์ทางแม่เหล็ก จึงหมุนได้โดยไม่ติดขัด แบ่งเป็นชนิดสแต็กเดียวและหลายสแต็ก สแต็กในที่นี้หมายถึงเฟสซึ่งประกอบด้วยโรเตอร์ที่เป็นซี่ฟันและโครงสร้างของสเตเตอร์อยู่ภายนอก สเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แปรค่าได้สแต็กเดียวจะมีโรเตอร์เดียวเมื่อเทียบกับสเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แปรค่าหลายสแต็ก

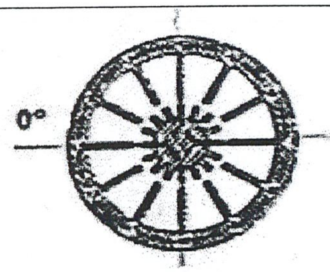
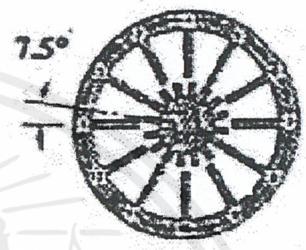
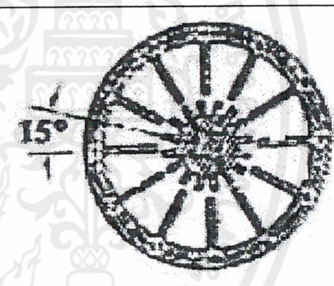
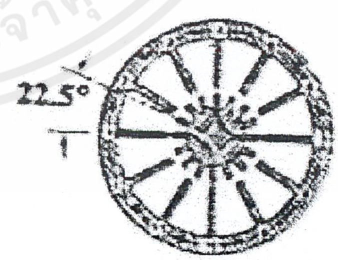
สต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แปรค่าได้สแต็กเดียวสามารถแสดงได้ดังรูปที่ 2.4 จากรูปเป็นชนิดมี 3 เฟส แต่ละเฟสใช้ขดลวดพันบน 4 ขั้วหรือซี่ฟันของสเตเตอร์ เฟสที่ 1 พันอยู่ที่ 1, 4, 7 และ 10 ของสเตเตอร์ ดังนั้นสเตเตอร์จะมี 12 ซี่ฟัน และ ณ ที่นี้กำหนดมีโรเตอร์ 16 ซี่ฟัน ขั้วของสเตเตอร์ที่อยู่ตรงกันข้ามจะพันด้วยขดลวดลักษณะที่ต่างกันเพื่อให้มีความสมดุลย์ระหว่างเส้นแรงแม่เหล็กเข้าและออกจากโรเตอร์ สมมุติว่ากระแส I_1 ป้อนให้กับเฟสที่ 1 ดังแสดงในรูปที่ 2.4 และโรเตอร์ทั้ง 4 ซี่ฟันจะอยู่ในแนวซี่ฟันที่ 1, 4, 7 และ 10 ของสเตเตอร์ เส้นแรงแม่เหล็กจะเข้าสู่โรเตอร์จากสเตเตอร์ซี่ฟันที่ 4 และ 10 และออกจากโรเตอร์ไปยังซี่ฟันของสเตเตอร์ที่ 1 และ 7 ซึ่งเป็นทางเดินของเส้นแรงแหล็กที่ครบวงจรโดยผ่านโครงร่างของสเตเตอร์ ปลายซี่ฟันของสเตเตอร์ที่ 4 จะถูกเหนี่ยวนำเป็นขั้วเหนือและปลายของซี่ฟันโรเตอร์ซึ่งอยู่ในแนวเดียวกับซี่ฟันที่ 4 ของสเตเตอร์จะเป็นเส้นทางผ่านเข้าไปยังโรเตอร์ของเส้นแรงแม่เหล็กและเหนี่ยวนำให้ปลายของซี่ฟันของโรเตอร์นั้นเป็นขั้วใต้ การทำให้เกิดลักษณะเป็นแม่เหล็กนี้จะทำให้มีเส้นแรงแม่เหล็กอย่างต่อเนื่องผ่านแอร์แกปที่เหลืออีก 3 คู่ก็เกิดลักษณะของแม่เหล็กในทำนองเดียวกัน

ในสภาวะต่อไปจะให้โรเตอร์หมุนไปหนึ่งสเต็ปในทิศทางตามเข็มนาฬิกา จึงต้องจ่ายพลังงานให้กับเฟสที่ 3 ที่มีขดลวดพันอยู่บนซี่ฟันที่ 2, 5, 8, 11 ของสเตเตอร์ด้วยกระแส I_3 หลังจากหยุดจ่ายกระแส I_1 แล้ว เส้นแรงแม่เหล็กจะหาทางเดินที่ต่างไปจากเดิมเพื่อทำให้วงจรแม่เหล็กครบวงจร ในทำนองเดียวกันเส้นแรงแม่เหล็กในวงจรไฟฟ้าจะหาเส้นแรงแม่เหล็กในวงจรแม่เหล็กก็จะหาเส้นทางเดินที่ค่ารีลัคแต้นซ์ต่ำสุด (แอร์แกประหว่างซี่ฟันจะทำให้เกิดค่ารีลัคแต้นซ์ต่อเส้นแรงแม่เหล็ก แอร์แกปกว้างมากค่ารีลัคแต้นซ์ก็จะมีค่ามาก) ด้วยเหตุผลดังกล่าวเส้นแรงแม่เหล็กจะออกจากขั้วที่ 2 และ 8 ของสเตเตอร์ซึ่งถูกเหนี่ยวนำให้เป็นขั้วเหนือ และเส้นแรงแม่เหล็กนี้จะกระโดดผ่านแอร์แกปไปยังซี่ฟันของโรเตอร์ที่ใกล้ที่สุด ซี่ฟัน a และ b ของโรเตอร์เป็นโรเตอร์ที่อยู่ใกล้ที่สุดและจะถูกเหนี่ยวนำให้เป็นขั้วใต้ เส้นแงแม่เหล็กจะออกจาก

ซี่ฟัน d และ e ของโรเตอร์ผ่านแอร์แกปเข้าสู่ซี่ฟันที่ 5 และ 11 ของสเตเตอร์ ดังนั้น ส่วนที่เหลือของวงจรแม่เหล็กจะสมบูรณ์โดยผ่านโครงร่างของสเตเตอร์ ในระหว่างเวลานั้นแรงของแม่เหล็กหรือแรงดึงดูดจะเกิดขึ้นระหว่างซี่ฟันที่ 2 ของสเตเตอร์ (ถูกเหนี่ยวนำเป็นขั้วเหนือ) และซี่ฟัน a ของโรเตอร์ (ถูกเหนี่ยวนำเป็นขั้วใต้) แรงดึงดูดจะเกิดขึ้นระหว่างคู่ขั้ว (11,e), (8,6) และ (5,d) ด้วย ผลที่เกิดขึ้นจะทำให้เกิดแรงบิดกระทำต่อโรเตอร์ ทำให้เกิดการหมุนไปจนกระทั่งซี่ฟัน a, d, b และ e ของโรเตอร์อยู่ในแนวเดียวกับซี่ฟัน 2, 5, 8 และ 11 ของสเตเตอร์ตามลำดับ ขณะเวลาดังกล่าแอร์แกประหว่างซี่ฟันตามลำดับจะมีค่าน้อยที่สุด ผลลัพธ์ของค่ารีลัคแต้นซ์จะมีค่าต่ำสุด และเส้นแรงแม่เหล็กจะมีค่าสูงสุดผ่านวงจรแม่เหล็กที่ตำแน่งเป็นนี้เป็นตำแหน่งที่สมดุลย์ของการขับเฟส 3 ในกระบวนการที่กล่าวมาแล้วโรเตอร์จะเคลื่อนที่ในทิศตามเข็มนาฬิกาหนึ่งสเต็ปเป็นมุม 7.5°



รูปที่ 2.4 สเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แบบแปรค่าได้สแต็กเดียว

การเรียงลำดับเฟส	ตำแหน่งของโรเตอร์และเส้นแรงแม่เหล็ก
ตำแหน่งโรเตอร์เริ่มต้น : - ϕ_1 ได้รับพลังงาน -ซี่ฟันของโรเตอร์จะอยู่ในแนวซี่ฟันที่ 1, 4, 7 และ 10 ของสเตเตอร์	
สเต็ปที่ 1: เฟส ϕ_1 ได้รับพลังงาน -ซี่ฟันของโรเตอร์จะอยู่ในแนวซี่ฟันที่ 2, 5, 8 และ 11 ของสเตเตอร์ -โรเตอร์จะเคลื่อนที่ไปในทิศทางตาม เข็มนาฬิกาเป็นมุม 7.5°(1/3 ช่วงห่าง ระหว่างซี่ฟันของโรเตอร์)	
สเต็ปที่ 2 : เฟส ϕ_2 ได้รับพลังงาน -ซี่ฟันของโรเตอร์จะอยู่ในแนวซี่ฟันที่ 3, 6, 9 และ 12 ของสเตเตอร์ -โรเตอร์จะเคลื่อนที่ไปในทิศทางตาม เข็มนาฬิกา รวมเป็นมุม 15°	
สเต็ปที่ 3 : เฟส ϕ_2 ได้รับพลังงาน -ซี่ฟันของโรเตอร์จะอยู่ในแนวซี่ฟันที่ 1, 4, 7 และ 10 ของสเตเตอร์ -โรเตอร์จะเคลื่อนที่ไปในทิศทางตาม เข็มนาฬิกา รวมเป็นมุม 22.5°	

ตารางที่ 2.1 ลำดับการทำงานของสเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์สแต็กเดียว

พารามิเตอร์ต่างๆของสเต็ปปิ้งมอเตอร์

1. ความห่างระหว่างปลายซี่ฟันของโรเตอร์และสเตเตอร์

$$P_r = \frac{360}{N_r} \tag{2.3}$$

และ $$P_s = \frac{360}{N_s} \tag{2.4}$$

2. มุมสเต็ป

$$\theta_s = \frac{P_r}{N_p} = \frac{360}{N_r N_p} \tag{2.5}$$

หรือ $$\theta_s = \left|P_r - P_s\right| \tag{2.6}$$

3. อัตราการสเต็ป

$$R_s = \frac{360}{\theta_s} = N_r N_p \tag{2.7}$$

4. ความเร็วของสเต็ปปิ้งมอเตอร์

$$\omega = \frac{60f}{R_s} = \frac{60f}{N_s N_r} = \frac{\theta_s}{6} \text{ rpm} \tag{2.8}$$

5. จำนวนโพลของสเตเตอร์ต่อเฟส

$$X = \frac{N_s}{N_p} \tag{2.9}$$

หรือ $$X = \frac{R_s}{N_p(N_p \pm 1)} = \frac{N_r}{(N_p \pm 1)} \tag{2.10}$$

โดยที่ กำหนดสัญลักษณ์ต่าง ๆ ดังนี้

N_r = จำนวนซี่ฟันของโรเตอร์

N_s = จำนวนซี่ฟันของสเตเตอร์

N_p = จำนวนเฟส

P_r = ความห่างระหว่างปลายซี่ฟันของโรเตอร์ (องศา)

P_s = ความห่างระหว่างปลายซี่ฟันของสเตเตอร์ (องศา)

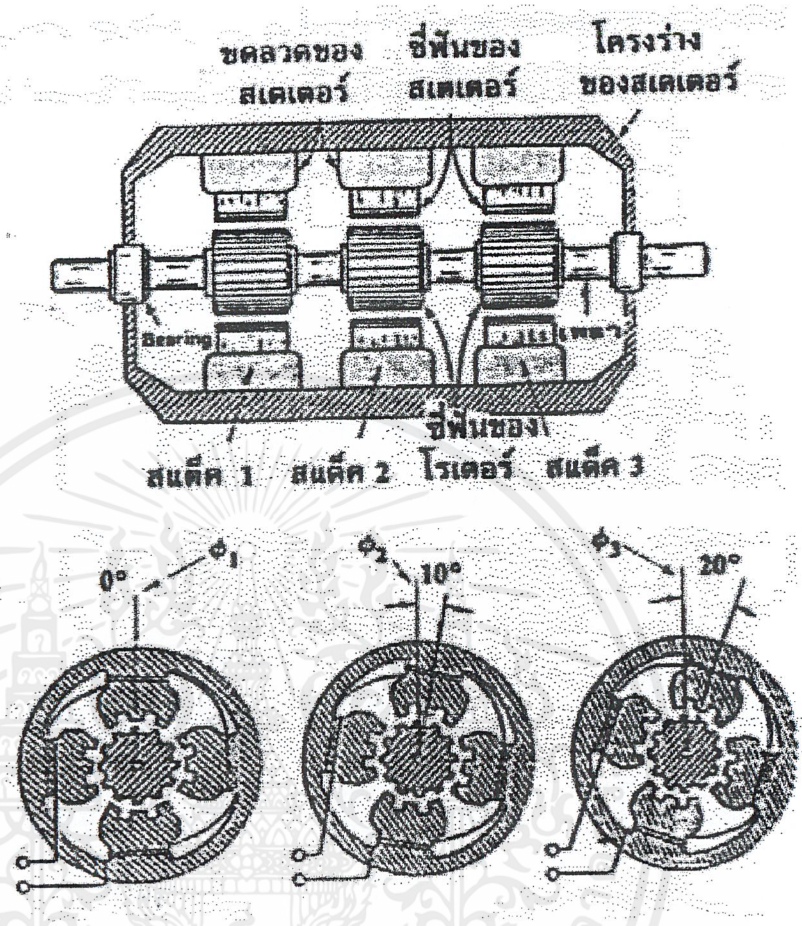
θ_s = มุมสเต็ป (องศา)

R_s = อัตราการสเต็ปหรือความเร็วในการสเต็ป

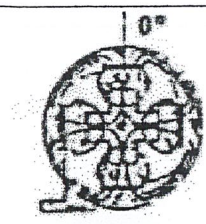
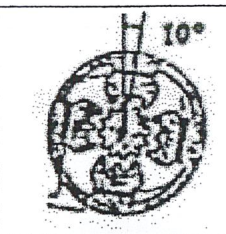
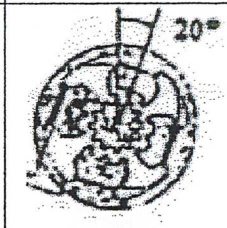
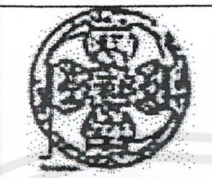
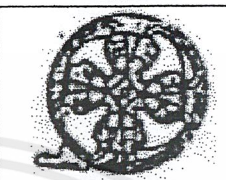
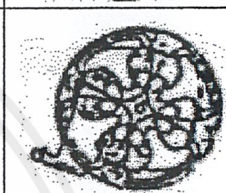
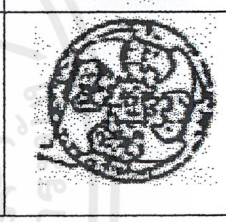
X = จำนวนซี่ฟันของสเตเตอร์ต่อเฟส

สเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แปรค่าหลายสแต็กแสดงดังรูปที่ 2.5 เป็นสเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แปรค่าได้ 3 สแต็ก สเตเตอร์ขแงแต่ละสแต็กประกอบด้วย 4 โพล และแต่ละโพลจะมีซี่ฟัน 3 ซี่ ในแต่ละสแต็กจำนวนซี่ฟันของโรเตอร์และสเตเตอร์จะมีจำนวนเท่ากันซึ่งต่างกับชนิดสแต็กเดียว คือ จำนวนซี่ฟันองโรเตอร์และสเตเตอร์จะเท่ากันไม่ได้ ถ้าเท่ากันจะไม่ทำงาน จากรูปจะเห็นว่าแต่ละสแต็กจะมีตำแหน่งของสเตเตอร์แตกต่างจากตำแหน่งของสเตเตอร์ในสแต็กถัดไปเท่ากับ 10° ส่วนซี่ฟันของโรเตอร์ทั้ง 3 อันจะประกอบอยู่บนแกนเดียวกันและได้ระบการปรับแต่งให้อยู่ในแนวเดียวกันอย่างสมบูรณ์

สเต็ปปิ้งมอเตอร์ชนิด 3 สแต็กถึงแม้ว่าโรเตอร์ทั้ง 3 อันจะติดอยู่บนเพลาอันเดียวกันแต่มีวงจรแม่เหล็กที่แยกกันดังนี้ คือ ถ้าเฟสที่ 1 ถูกขับด้วยกระแสเป็นเฟสเริ่มต้นให้ซี่ฟันของโรเตอร์-สเตเตอร์อยู่ในแนวเดียวกัน ส่วนซี่ฟันของโรเตอร์และสเตเตอร์ในสแต็กที่ 2 ในขณะนั้นจะมีตำแหน่งต่างกัน 10° และซี่ฟันของโรเตอร์และสเตเตอร์ในสแต็คที่ 3 จะมีตำแหน่งต่างกัน 20° ต่อจากนั้นหยุดจ่ายกระแสในสแต็กที่ 1 และป้อนกระแสให้กับสแต็กที่ 2 โรเตอร์จะหมุนไปอีก 10° ซึ่งจะทำให้ซี่ฟันของโรเตอร์และสเตเตอร์ในสแต็กที่ 2 อยู่ในแนวเดียวกัน ในขณะนี้ซี่ฟันของโรเตอร์และสเตเตอร์ในสแต็กที่ 3 จะมีตำแหน่งต่างกัน 10° จากนั้นหยุดจ่ายกระแสในให้สแต็กที่ 2 และป้อนกระแสให้กับสแต็กที่ 3 โรเตอร์จะหมุนไปอีก 10° ซึ่งจะทำให้ซี่ฟันของโรเตอร์และและสเตเตอร์ในสแต็กที่ 3 อยู่ในแนวเดียวกัน ส่วนซี่ฟันของโรเตอร์และสเตเตอร์ในสแต็กที่ 1 จะมีตำแหน่งต่างกัน 10°

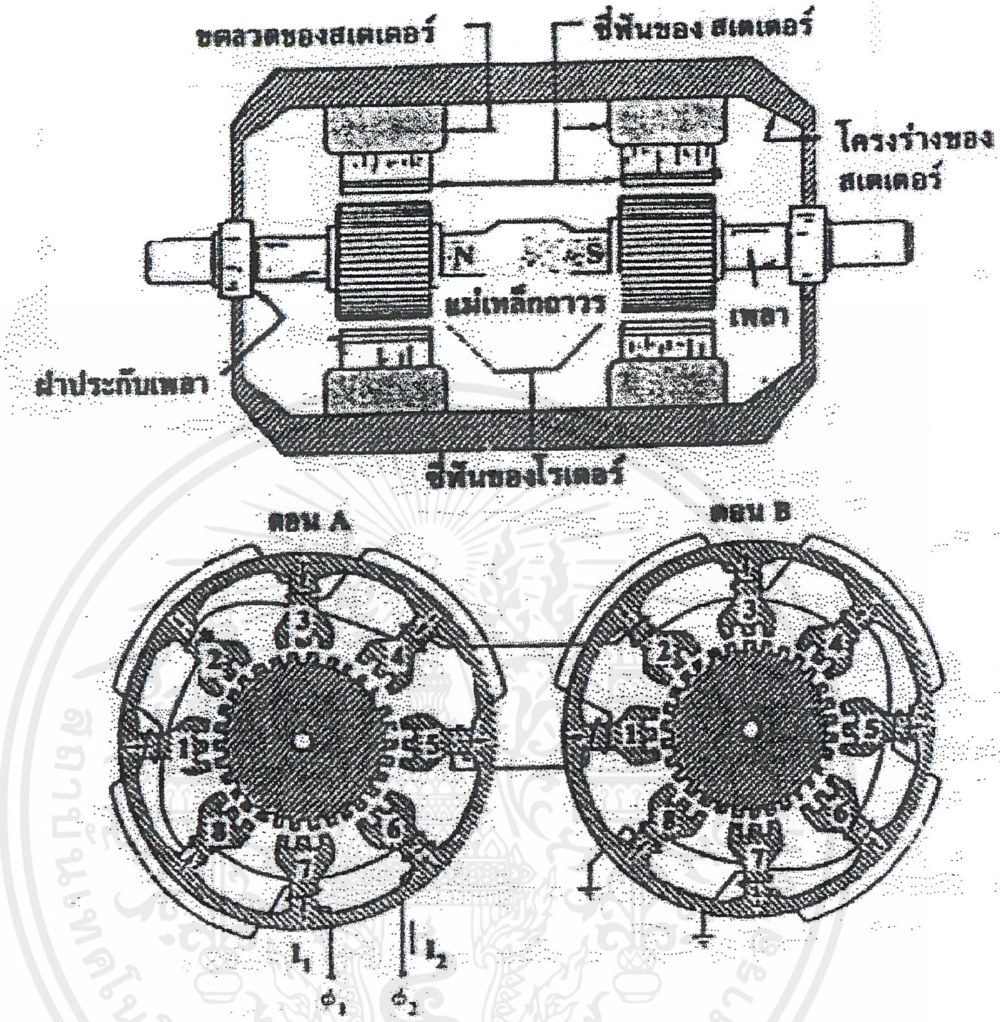


รูปที่ 2.5 สเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์แบบแปรค่าได้หลายสแต็ก

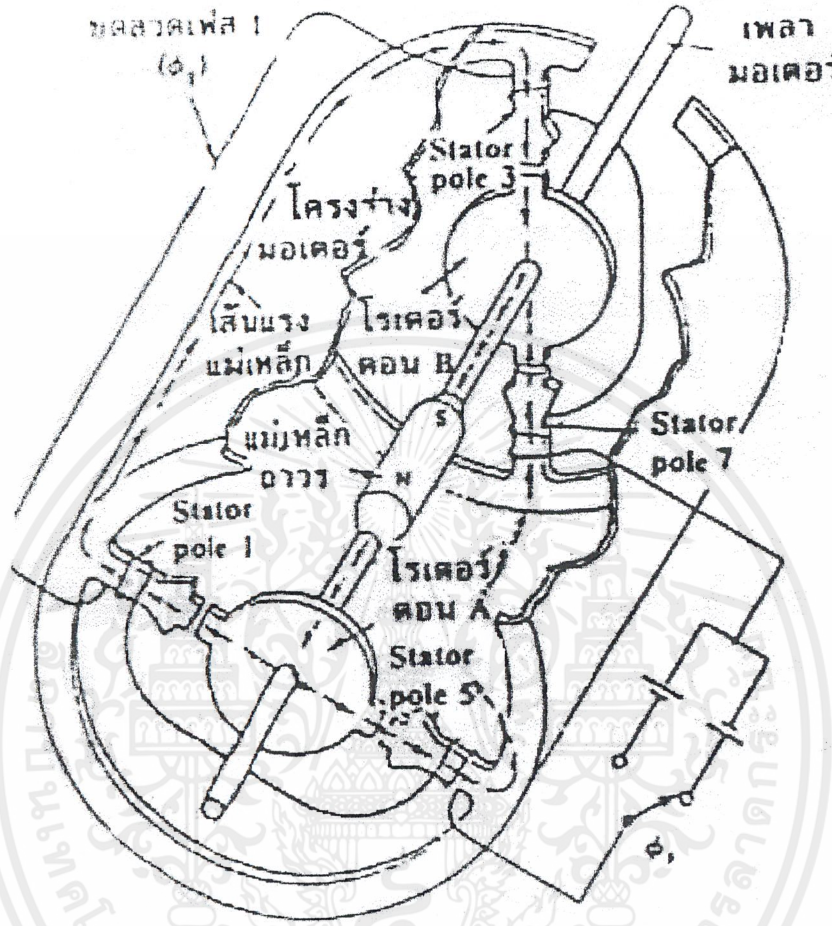
	สเต็กที่ 1	สเต็กที่ 2	สเต็กที่ 3
ตำแหน่งเริ่มต้นของโรเตอร์: -ϕ_1 ได้รับพลังงาน			
สเต็ปที่ 1: เฟส ϕ_2 ได้รับพลังงาน			
สเต็ปที่ 2: เฟส ϕ_3 ได้รับพลังงาน			
สเต็ปที่ 3: เฟส ϕ_3 ได้รับพลังงาน			

ตารางที่ 2.2 ลำดับการทำงานของของสเต็ปปิ้งมอเตอร์แบบรีลัคแต้นซ์ 3 สเต็ก

3. สเต็ปปิ้งมอเตอร์แบบไฮบริด (Hybrid) นิยมใช้งานที่ต้องต่อกับคอมพิวเตอร์ มีโครงสร้างโรเตอร์แบบแม่เหล็กถาวรและสเตเตอร์แบบรีลัคแต้นซ์แปรค่าได้ โครงสร้างของไฮบริดแสดงได้ดังรูปที่ 2.6 โครงสร้างประกอบด้วย 2 ตอนกับแกนแม่เหล็กอยู่ระหว่าง 2 ตอน แต่ละตอนประกอบด้วยซี่ฟันของโรเตอร์และโพลของสแตอร์ที่มีซี่ฟันเช่นกันแล้พันด้วยขดลวด การทำงานของไฮบริดเริ่มจากขณะที่เฟส ϕ_1 ได้รับพลังงานโดยการป้อนกระแส I_1 ซี่ฟันของโรเตอร์ในตอน A จะอยู่ในแนวเดียวกับซี่ฟันของโพลที่ 1 และโพลที่ 5 ส่วนของตอน B จะอยู่ในแนวเดียวกับซี่ฟันของโพลที่ 3 และโพลที่ 7 ดังรูปที่ 2.7 เพื่อให้เพลาของมอเตอร์หมุนไปหนึ่งสเต็ปในทิศทางตามเข็มนาฬิกา จะต้องหยุดป้อนกระแส I_1 และ ป้อนกระแส I_2 ให้กับเฟส ϕ_2



รูปที่ 2.6 สเต็ปปิ้งมอเตอร์แบบไฮบริด



รูปที่ 2.7 วงจรแม่เหล็กของสเต็ปปิ้งมอเตอร์แบบไฮบริด

ชนิดมอเตอร์ / คุณสมบัติ	แบบ VR	แบบ PM	แบบ Hybrid
ความถูกต้อง	ต่ำเมื่อมีสเต็ปการหมุนสูง	มากสุดเมื่อเทียบกับชนิดอื่น	สูงแม้สเต็ปการหมุนจะสูง
แรงยึดเหนี่ยว	ต่ำ	สูง	สูง
แรงบิดดึงและแรงผลัก	ไม่ดี	ดี	ดี

ตารางที่ 2.3 การเปรียบเทียบคุณสมบัติของมอเตอร์แต่ละชนิด

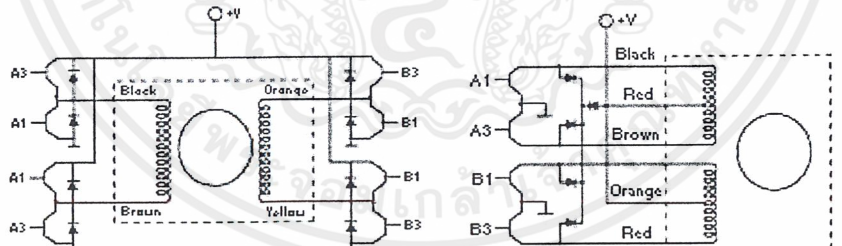
นอกจากชนิดของสเต็ปปิ้งมอเตอร์ที่ได้กล่าวมาแล้ว ในปัจจุบันได้มีการออกแบบสเต็ปปิ้งมอเตอร์สำหรับการใช้งานเฉพาะอีกหลายรูปแบบ

2.1.3 การพันขดลวดของสเต็ปปิ้งมอเตอร์

การพันขดลวดบนสเตเตอร์ของสเต็ปปิ้งมอเตอร์ มีด้วยกัน 2 วิธี คือ

1. ไบโพลาร์ (Bipolar) จะมีการพันขดลวดในแต่ละขั้วแม่เหล็กของสเตเตอร์ โดยขั้วแม่เหล็กที่เกิดขึ้นที่สเตเตอร์จะถูกกำหนดโดยทิศทางของการไหลของกระแสไฟฟ้า ซึ่งสามารถทำให้เกิดขั้วแม่เหล็กในทิศทางตรงกันข้ามได้ เพียงการกลับทิศทางของการไหลในกระแสไฟฟ้า
2. ยูนิโพลาร์ (Unipolar) มีการพันขดลวด 2 ขด บนแต่ละขั้วแม่เหล็กของสเตเตอร์ แต่ละขดแบ่งเป็น 2 เฟส (มอเตอร์ชนิดนี้จึงมีทั้งหมด 4 เฟส) ทำให้และขดลวดเกิดขั้วแม่เหล็กในทิศทางตรงกันข้าม การกลับทิศทางขั้วแม่เหล็กทำได้โดยใช้วงจรสวิทซึ่งให้กระแสไฟฟ้าไหลสลับจากขดลวดหนึ่งไปยังอีกขดลวดหนึ่งเท่านั้น

การพันทั้งสองแบบมีคุณสมบัติที่ต่างกัน คือ แบบยูนิโพลาร์จะทำให้เกิดแรงบิดน้อยกว่าแบบไบโพลาร์ ซึ่งสามารถสังเกตความแตกต่างของมอเตอร์ทั้งสองชนิดได้จากสายไฟที่ต่อมาจากสเต็ปปิ้งมอเตอร์ คือ แบบไบโพลาร์จะมี 4 สาย (2 เฟส) ส่วนแบบยูนิโพลาร์จะมี 5 หรือ 6 สาย (4 เฟสหรือมากกว่า) ทั้งนี้วงจรขับก็ต่างกันเพราะไบโพลาร์มี 2 เฟส จึงต้องมีการป้องกันโวลต์เตจย้อนกลับ



รูปที่ 2.8 การพันขดลวด
(ก) ไบโพลาร์ (ข) ยูนิโพลาร์

2.1.4 การกระตุ้นสเต็ปปิ้งมอเตอร์

การควบคุมและสั่งงานให้สเต็ปปิ้งมอเตอร์ทำงานสามารถทำได้โดยการจ่ายกำลังไปยังขดลวดในแต่ละขดบนสเตเตอร์ โดยการป้อนจะทำในลักษณะเป็นลำดับในลูปที่ถูกต้อง มี 3 รูปแบบ ได้แก่

1. การกระตุ้นแบบเฟสเดียว (One Phase Excitation)

ทำการกระตุ้นขดลวดทีละขดในเวลาหนึ่งไล่เรียงกันไป ทิศทางการหมุนของมอเตอร์ขึ้นกับทิศทางการป้อนกำลังไฟฟ้า สามารถเขียนขั้นตอนการกระตุ้นเป็นตารางออกมาได้ดังนี้

สเต็ปที่	เฟส 1	เฟส 2	เฟส 3	เฟส 4
1	ON			
2		ON		
3			ON	
4				ON
5	ON			
6		ON		
7			ON	
8				ON

ตารางที่ 2.4 การกระตุ้นสเต็ปปิ้งมอเตอร์แบบเฟสเดียว

2. การกระตุ้นแบบ 2 เฟส (Two Phase Excitation)

คล้ายกับการกระตุ้นแบบแรกแต่ต่างกันตรงที่จะกระตุ้นทีละ 2 ขด ที่อยู่ใกล้กันในเวลาเดียวกัน และจะเรียงลำดับกันไปเช่นเดียวกับแบบเฟสเดียวแต่จะให้แรงบิดที่มากกว่า แต่ข้อเสียของแบบนี้คือต้องให้กำลังไฟที่มากกว่าเป็น 2 เท่าของแบบเฟสเดียว สามารถเขียนลำดับการกระตุ้นของขดลวดแบบ 2 เฟส ได้ดังนี้

สเต็ปที่	เฟส 1	เฟส 2	เฟส 3	เฟส 4
1	ON	ON		
2		ON	ON	
3			ON	ON
4	ON			ON
5	ON	ON		
6		ON	ON	
7			ON	ON
8				ON

ตารางที่ 2.5 การกระตุ้นสเต็ปปิ้งมอเตอร์แบบ 2 เฟส

3. การกระตุ้นแบบครึ่งสเต็ป(Half Step Excitation)

เป็นการผสมผสานของการกระตุ้นระหว่างแบบหนึ่งเฟสกับ 2 เฟส เพื่อให้จำนวนรอบของสเต็ปมากขึ้นเป็น 2 เท่า ได้แรงบิดมาก เนื่องจากมีช่วงสเต็ปที่มีระยะสั้นลงและมีแรงดึงจากขดลวด 2 ขดที่ถูกกระตุ้นพร้อมกัน และเป็นผลให้ค่าตำแหน่งความถูกต้องมากขึ้นไปด้วย แต่มีข้อเสียตรงที่ต้องจ่ายไฟเป็น 2 เท่าแบบหนึ่งเฟส สามารถลำดับการทำงานของแบบครึ่งสเต็ปได้ดังนี้

สเต็ปที่	เฟส 1	เฟส 2	เฟส 3	เฟส 4
1	ON			
2	ON	ON		
3		ON		
4		ON	ON	
5			ON	
6			ON	ON
7				ON
8	ON			ON

ตารางที่ 2.6 การกระตุ้นสเต็ปปิ้งมอเตอร์แบบครึ่งสเต็ป

2.2 พอร์ตขนาน

ระบบคอมพิวเตอร์ประกอบด้วยพื้นฐานสำคัญ คือ หน่วยความจำ หน่วยรับและส่งข้อมูล หน่วยประมวลผลกลาง โดยแต่ละส่วนทำงานเกี่ยวโยงสัมพันธ์กัน โดยมีการติดต่อเชื่อมโยงจากส่วนหนึ่งไปยังอีกส่วนหนึ่ง ในการอินเตอร์เฟสจึงต้องพิจารณาถึงการทำงานส่วนใหญ่ของเครื่องคอมพิวเตอร์ คือ การรับส่งข้อมูลจากส่วนหนึ่งไปยังอีกส่วนหนึ่ง ซึ่งจะเกี่ยวข้องกับการทำงานด้านอื่นๆ ที่มีกระบวนการด้านคณิตศาสตร์และทางลอจิกเกิดจากการทำงานของหน่วยประมวลผลกลางร่วมกับอุปกรณ์อื่น ๆ บนแผ่นวงจรเดียวกัน เช่น การติดต่อกับหน่วยแสดงผล จอมอนิเตอร์หรืออุปกรณ์ภายนอกต่าง ๆ

2.2.1 การอินเตอร์เฟสแบบขนาน

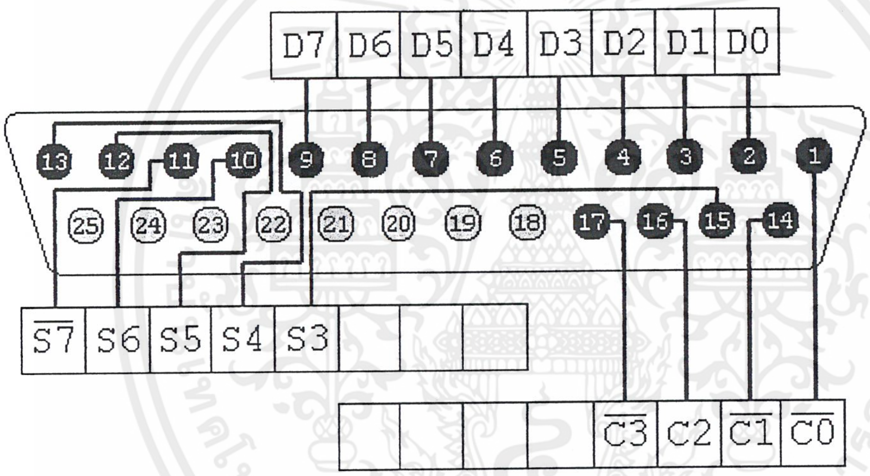
พอร์ตขนานมีการอินเตอร์เฟสแบบขนาน คือ การรับส่งข้อมูลที่ข้อมูลทุก ๆ บิตจะถูกส่งออกไปพร้อม ๆ กันครั้งเดียว โดยแต่ละบิตจะส่งในสายส่งแต่ละเส้น จึงทำให้มีอัตราการถ่ายทอดข้อมูลสูงกว่าแบบอนุกรมมาก (เร็วกว่าประมาณ 8-10 เท่า) จึงสามารถทำการประยุกต์ใช้งานได้ดีในการติดต่อข้อมูลกับอุปกรณ์ภายนอกให้ระดับแรงดันที่ใช้กับอุปกรณ์ TTL ที่สัญญาณลอจิกเป็น"1"เท่ากับ +5 โวลท์ และลอจิกเป็น"0"เท่ากับ 0 โวลท์ แต่ก็มีข้อเสีย คือ ไม่สามารถทำงานในระยะทางที่ไกล ๆ ระหว่างอุปกรณ์ที่ติดต่อกันจะเกิดความผิดพลาดของข้อมูลขึ้นง่ายเนื่องจากระดับแรงดันไม่สม่ำเสมออีกทั้งสิ้นเปลืองค่าใช้จ่ายในการต้องใช้สายสัญญาณหลายเส้น พอร์ตนี้เป็นได้ทั้งอินพุตและเอาต์พุตแต่โดยปกติมักจะใช้เป็นเอาต์พุตเพื่อให้ข้อมูลกับเครื่องพิมพ์

2.2.2 ข้อมูลของรีจิสเตอร์

พอร์ตขนานมีรีจิสเตอร์ควบคุมการทำงานอยู่ 3 ตัว ดังนี้

1. พอร์ตเอาต์พุตสำหรับสัญญาณข้อมูล 8 เส้น มีรีจิสเตอร์ดาต้า (Data Register) ควบคุม
2. พอร์ตอินพุตสำหรับการอ่านค่าสถานะต่าง ๆ จากภายนอก มีอยู่ 8 เส้น มีรีจิสเตอร์สเตตัส (Status Register) ควบคุม
3. พอร์ตเอาต์พุตสำหรับส่งสัญญาณควบคุมไปยังอุปกรณ์ภายนอก มีอยู่ด้วยกัน 4 เส้น ใช้รีจิสเตอร์คอนโทรล (Control Register) ในการควบคุม

2.2.3 ขาของพอร์ตขนาน



รูปที่ 2.9 ตำแหน่งขาของพอร์ตขนาน

DB-25	รีจิสเตอร์	ทิศทาง	บิต	ขาสัญญาณ	หน้าที่การทำงาน
1	Control	out	$\overline{\text{C0}}$	$\overline{\text{STROBE}}$	ส่งค่าออกบอกว่าที่ขาดาต้ามีข้อมูลแล้ว
2-9	Data	out	D1-D8	DATA1-DATA8	รับข้อมูลเอาต์พุต บางครั้งใช้เป็นขาส่งข้อมูลได้
10	Status	In	S6	nACK	เป็นพัลส์จากเครื่องพิมพ์เพื่อบอกว่าได้รับข้อมูลแล้ว
11	Status	In	$\overline{\text{S7}}$	$\overline{\text{BUSY}}$	เครื่องพิมพ์แจ้งว่ายังไม่พร้อมรับข้อมูล
12	Status	In	S5	PE	แจ้งกระดาษหมด
13	Status	In	S4	SELECT	แจ้งว่าเครื่องพิมพ์ต่ออยู่
14	Control	out	$\overline{\text{C1}}$	$\overline{\text{AUTO FEED}}$	สั่งเครื่องพิมพ์ให้เลื่อนบันทัด
15	Status	In	S3	$\overline{\text{ERROR}}$	สัญญาณจากเครื่องพิมพ์แสดงข้อผิดพลาดของการพิมพ์
16	Control	out	C2	$\overline{\text{INIT}}$	รีเซตเครื่องพิมพ์
17	Control	out	$\overline{\text{C3}}$	$\overline{\text{SELECT-IN}}$	ส่งสัญญาณการเลือกเครื่อง
18-25				GND	กราวด์

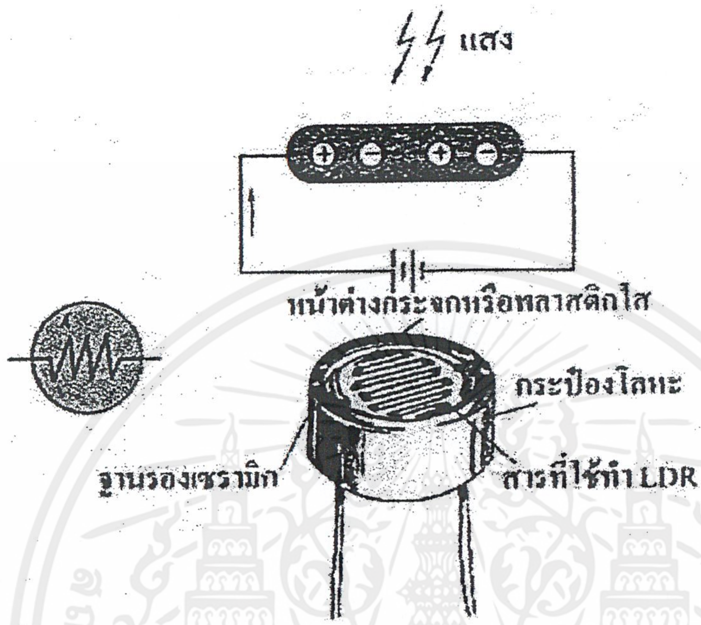
ตารางที่ 2.7 สัญญาณทั้งหมดของพอร์ตขนาน

2.3 ตัวต้านทานไวแสง

อุปกรณ์ตัวรับแสงจะทำหน้าที่ในการเปลี่ยนแปลงค่าของตัวมันซึ่งอยู่ในรูปแบบทางไฟฟ้าให้แปรกับค่าของพลังงานแสงที่ได้รับ โดยอุปกรณ์ตัวรับแสงจะประกอบด้วยสารกึ่งตัวนำซึ่งอาจนำมาต่อเชื่อมให้เกิดเป็นรอยต่อหรือเป็นเนื้อสารกึ่งตัวนำอย่างเดียวก็ได้ ซึ่งคุณสมบัติที่ได้ก็จะต่างกันในแต่ละชนิด ตัวอย่างของอุปกรณ์ตัวรับแสงได้แก่ ตัวต้านทานไวแสง โฟโต้ไดโอด โฟโต้ทรานซิสเตอร์และโซล่าร์เซลล์ ในปริญญานิพนธ์นี้เลือกใช้ตัวต้านทานไวแสงจึงจะกล่าวเฉพาะตัวต้านทานไวแสงเท่านั้น

ตัวต้านทานไวแสงหรือแอลดีอาร์ (Light Dependent Resister:LDR) เป็นตัวต้านทานที่แปรค่าได้กับแสง โดยเมื่อป้อนพลังงานอย่างเพียงพอให้กับวาเลนซ์อิเล็คตรอน (Valence Electron) ของสารกึ่งตัวนำจะทำให้เกิดการแตกตัวของโฮล (Hole) และอิเล็คตรอนเกิดขึ้นมากมาย พลังงานที่จะ

ทำให้โฮลและอิเล็คตรอนแตกตัวนี้จะต้องเป็นพลังงานที่มาจากภายนอก เช่น แสง ความร้อน ปริมาณของพลังงานที่ทำให้เกิดการแตกตัวของอิเล็คตรอน-โฮล (Electron-Hole) จะขึ้นอยู่กับชนิดของสารนั้นๆ



รูปที่ 2.10 (ก) หลักการทำงานของแอลดีอาร์
(ข) โครงสร้างการทำงานของแอลดีอาร์แคดเมียมซัลไฟล์
(ค) ความต้านทานของแอลดีอาร์ที่เปลี่ยนแปลงตามความสว่างของแสง

หลักการทำงานของ แอลดีอาร์ แสดงได้ดังรูปที่ 2.10 (ก) แรงดันที่ตกคร่อมสารกึ่งตัวนำซึ่งมีค่าความนำไฟฟ้า (σ) จะมีผลให้กระแส (I) ไหล และเมื่อได้รับแสงมากขึ้นค่าความนำก็จะเพิ่มขึ้น($\Delta\sigma$) และจะเป็นสัดส่วนกับการแตกตัวของอิเล็คตรอน-โฮลที่เกิดขึ้นเนื่องมาจากแสง (Δn_I) ทำให้กระแสไหลเพิ่มขึ้น (ΔI) เขียนเป็นสมการ ได้ดังนี้

$$\sigma = KI/v \qquad (2.11)$$

K เป็นค่าคงที่ขึ้นกับขนาดตัวรับแสงที่ใช้

$$\sigma + \Delta\sigma = K((I+\Delta I)/v) \qquad (2.12)$$

และ

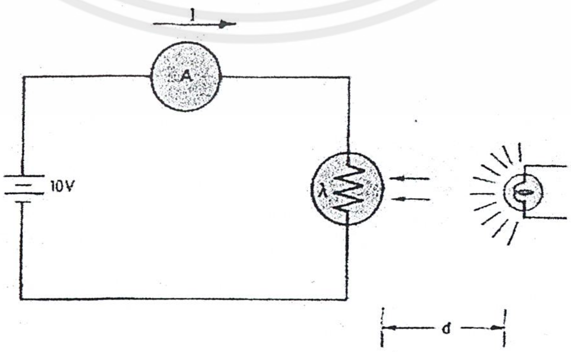
$$\Delta\sigma = K\Delta I/v \quad (2.13)$$

และเนื่องมาจากค่ากระแสที่เปลี่ยนแปลง (ΔI) เป็นสัดส่วนกับจำนวนการแตกตัวของอิเล็คตรอน-โฮลที่เกิดขึ้นเนื่องมาจากพลังงานแสงที่ได้รับ (Δn_I) ดังนั้นค่าความนำไฟฟ้าของแอลดีอาร์จึงเป็นสัดส่วนโดยตรงกับพลังงานแสงที่ได้รับหรือค่าความเข้มแสงนั่นเอง

แอลดีอาร์ส่วนใหญ่ทำมาจากสารกึ่งตัวนำจำพวกแคดเมียมซัลไฟด์หรือแคดเมียมเซเลไนด์ ซึ่งให้ผลตอบสนองต่อแสงได้ดีที่ความยาวคลื่นประมาณ 4,000 – 10,000 ^{o}A ซึ่งเป็นแถบแสงที่ตามองเห็นพอดี โครงสร้างของแอลดีอาร์ที่ทำมาจากสารกึ่งตัวนำจำพวกแคดเมียมซัลไฟด์แสดงได้ดังรูปที่ 2.10 (ข) ซึ่งจะถูกห่อหุ้มไว้ด้วยวัตถุที่แสงส่องผ่านได้ ในที่มีความต้านทาน $\approx 10\ M\Omega$ ที่อุณหภูมิห้อง สำหรับเส้นผ่าศูนย์กลาง 1.4 เซนติเมตร ลักษณะการทำงานของแอลดีอาร์จะเห็นได้ว่าเมื่อแสงมีความเข้มมากๆ มาตกกระทบก็จะทำให้ตัวมันสามารถนำกระแสไฟฟ้าได้ดีขึ้นเป็นผลให้ค่าความต้านทานในตัวมันลดลง ดังแสดงในรูปที่ 2.10 (ค)

ข้อดีของแอลดีอาร์ คือ ราคาถูก มีขนาดเล็ก มีการเปลี่ยนแปลงค่าได้กว้าง และมีอัตราขยายที่สูง แต่ก็ยังมีข้อเสียเมื่อเปรียบเทียบกับอุปกรณ์รับแสงตัวอื่นๆ คือ มีผลตอบสนองต่อความถี่ของแสงได้แคบ โดยแอลดีอาร์จะมีค่าผลตอบสนองเชิงเวลา (Time Response) ที่ช้า เพราะต้องขึ้นกับเวลาในการรวมตัวของประจุและเป็นสิ่งที่สำคัญสำหรับการพิจารณาเพื่อนำมาใช้ในการ moduration ซึ่งค่าความเข้มแสงจะเปลี่ยนแปลงอย่างรวดเร็ว โดยปรกติแล้วจะใช้ rise time = 75 ms และ fall time = 350 ms

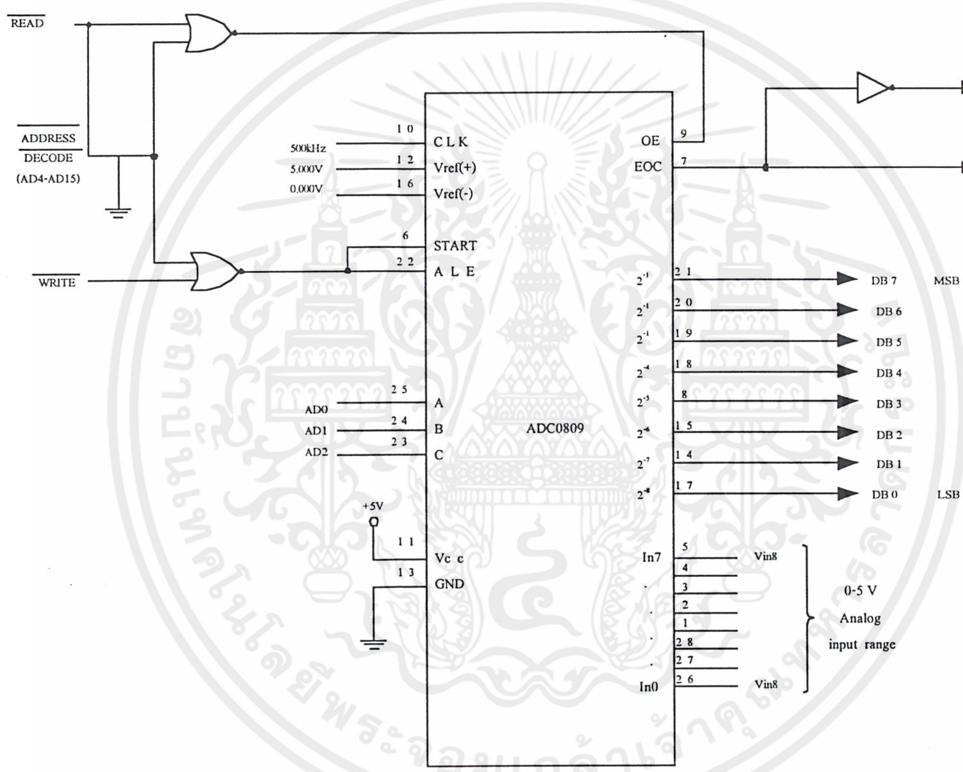
ตัวอย่างการใช้งานของแอลดีอาร์ที่เห็นได้ง่ายๆก็คือใช้เป็นตัววัดความเข้มของแสง โดยต่อเป็นวงจรเบื้องต้น ดังรูปที่ 2.11



รูปที่ 2.11 ตัวอย่างวงจรวัดความเข้มแสง

2.4 วงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล

รูปแบบสัญญาณไฟฟ้าพบในชีวิตประจำวันนั้นมักจะอยู่ในรูปของสัญญาณที่ต่อเนื่องหรือที่เรียกว่าสัญญาณอนาลอก (Analog Signal) ซึ่งเดิมการประมวลผลในงานต่าง ๆ จะนำสัญญาณแบบอนาลอกมาใช้ แต่เมื่อเทคนิคและอุปกรณ์การประมวลผลสัญญาณทางดิจิตอลได้รับการพัฒนาขึ้นมา เนื่องจากความสามารถในการประมวลเก็บข้อมูล การสื่อสารและการนำเสนอกระทำได้ง่ายและมีประสิทธิภาพมากกว่า ดังนั้นการเปลี่ยนรูปสัญญาณจึงได้มีการพัฒนาขึ้น



รูปที่ 2.12 วงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล

หลักการทำงานของวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลสามารถอธิบายได้ว่า

ADC0809 สามารถรับค่าอินพุตได้ 8 ค่า แต่จะรับได้ทีละค่า ขาที่กำหนดตำแหน่งการรับอินพุตที่ขาเข้ามานั้นกำหนดจากขา A B และ C ดังนี้

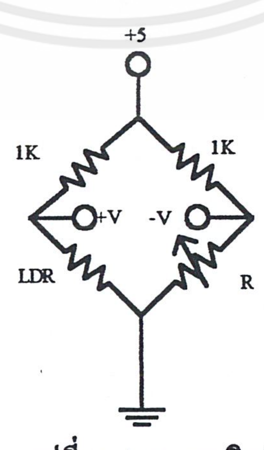
ขาอินพุต	ขาที่ใช้ในการรับข้อมูล		
	C	B	A
IN0	0	0	0
IN1	0	0	1
IN2	0	1	0
IN3	0	1	1
IN4	1	0	0
IN5	1	0	1
IN6	1	1	0
IN7	1	1	1

ตารางที่ 2.8 การกำหนดขาอินพุตของ ADC0809

วงจรจะทำงานตามสัญญาณนาฬิกา ซึ่งในวงจรนี้ทำการจ่ายสัญญาณนาฬิกาที่ความถี่ 500 KHz ค่าอินพุตอยู่ในช่วง 0 – 5 โวลต์และทำการส่งค่าดิจิตอลออกไปเมื่อขา WRITE = 0

2.5 วงจรบริดจ์

เป็นวงจรที่ใช้เปรียบเทียบแรงดันที่ถูกส่งออกมาจากแอลดีอาร์ก่อนจะส่งต่อไปยังวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล



รูปที่ 2.13 วงจรบริดจ์

บทที่ 3
บอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนาน และการนำไปประยุกต์ใช้งานในการควบคุมมู่ลี่อัตโนมัติ

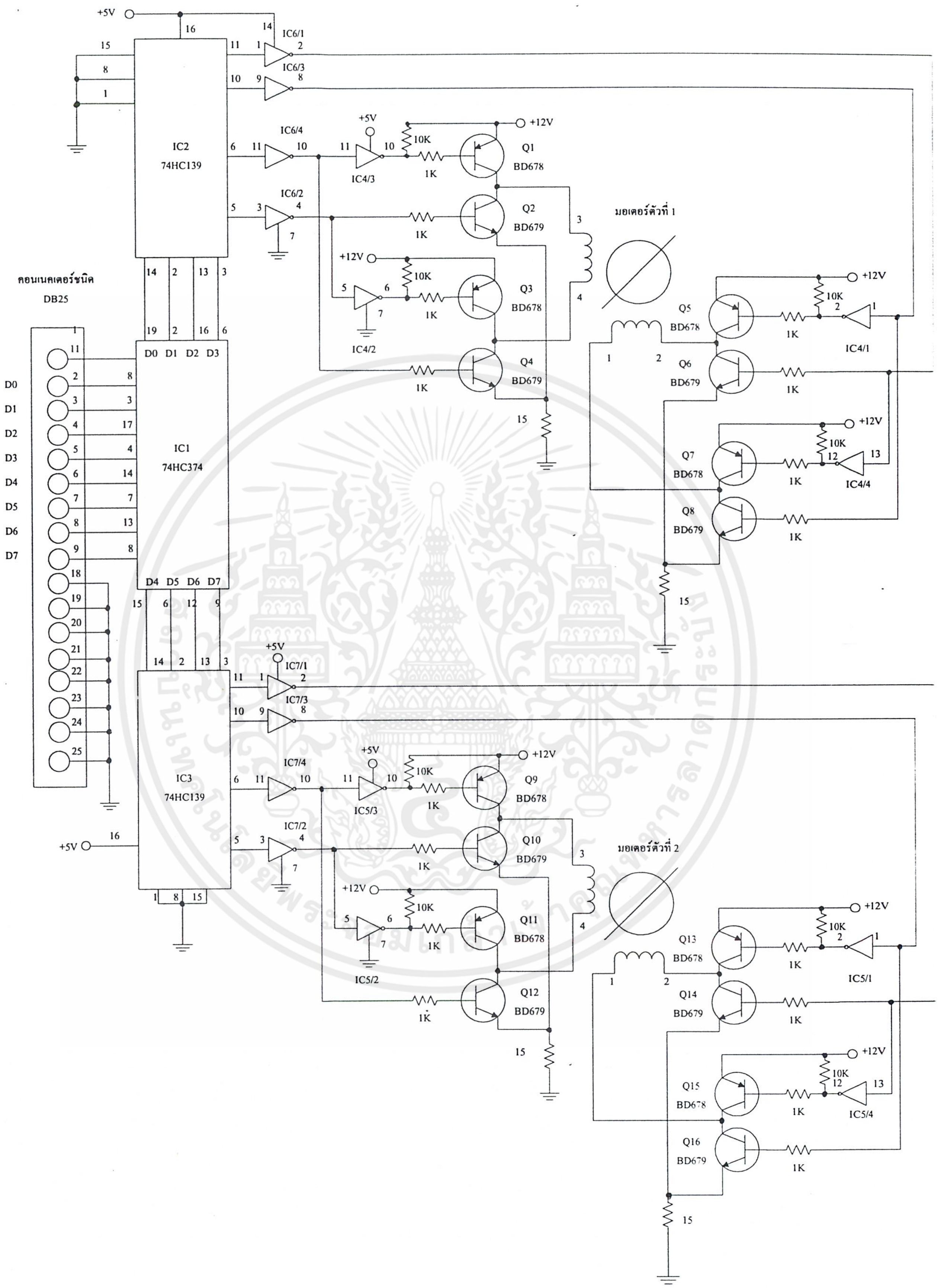
3.1 วงจรขับสเต็ปปิ้งมอเตอร์

จากรูปที่ 3.1 สามารถอธิบายการทำงานของวงจรขับสเต็ปปิ้งมอเตอร์ซึ่งมีรูปถ่ายจริงดังรูปที่ 3.2และรูปที่ 3.3ได้ว่า IC1 เป็นดี-ฟลิปฟลอป (D Flip-Flop) ซึ่งทำงานร่วมกับสัญญาณนาฬิกาของคอมพิวเตอร์ที่ขา 11 เพื่อควบคุมแลตช์อีนาเบิล (Latch Enable) และทำหน้าที่เป็นตัวเชื่อมต่อระหว่างพอร์ตขนานและบอร์ดควบคุม เมื่อคอมพิวเตอร์ส่งข้อมูลขนาด 8 บิตผ่านทางพอร์ตขนาน ขา 11 จะมีลอจิกเป็น “1” เอาท์พุตของ IC1 จึงทำงาน (ขา 19,2,16,5,15,6,12,9) จากนั้นข้อมูลจะส่งไปยัง IC2 และ IC3 ซึ่งเป็นไอซีถอดกระแส (Decoder) ทำหน้าที่ถอดรหัสข้อมูลก่อนส่งต่อไปยัง IC6 เพื่อแอกทีฟลอจิกเป็น “1” สำหรับป้อนให้วงจรขับบริดจ์

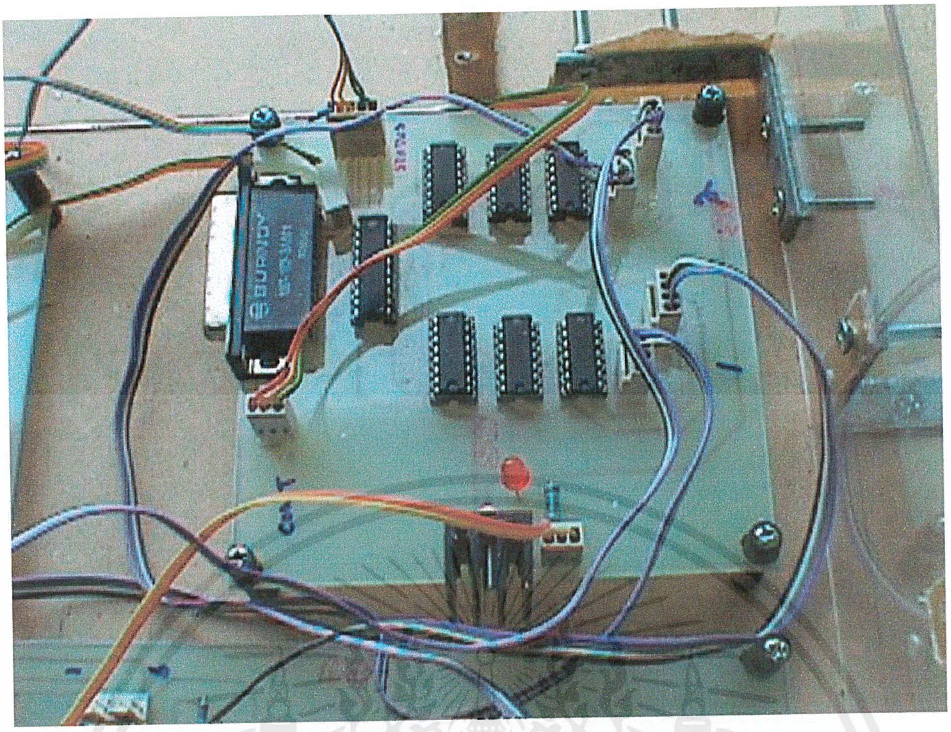
IC6 จะทำหน้าที่ควบคุมเฉพาะทรานซิสเตอร์ชนิด NPN ในวงจรบริดจ์ IC4 ควบคุมทรานซิสเตอร์ชนิดพีเอ็นพี (PNP) หน้าที่ของ IC4 มี 2 ประการ คือ เปลี่ยนระดับแรงดันสัญญาณควบคุมจากเอาต์พุตของ IC6 จาก 5 โวลต์เป็น 12 โวลต์ในวงจรบริดจ์และอีกประการคือเป็นตัวกลับสถานะของสัญญาณอีกครั้งเพื่อไบแอสให้กับทรานซิสเตอร์แบบพีเอ็นพี

ส่วนวงจรขับแบบบริดจ์ประกอบด้วย Q1, Q2, Q3, Q4 และ Q1, Q2 ต่อทำงานแบบคอมพลีเมนตารีสวิตซ์ ดังนั้นเมื่อ Q1 ทำงาน Q2 จะไม่ทำงาน เช่นเดียวกัน เมื่อ Q3 ทำงาน Q4 จะไม่ทำงาน เพื่อกำหนดทิศทางการไหลของกระแสไฟฟ้าที่ป้อนเข้าสู่ขดลวดของสเต็ปปิ้งมอเตอร์หรือกำหนดขั้วไฟฟ้าที่ขดลวด ส่วน Q5-Q8 ก็เป็นวงจรบริดจ์ที่ทำงานเหมือนกับวงจรบริดจ์ของ Q1-Q4 ซึ่ง Q1-Q8 ควบคุมสเต็ปปิ้งมอเตอร์ได้ 1 ตัว

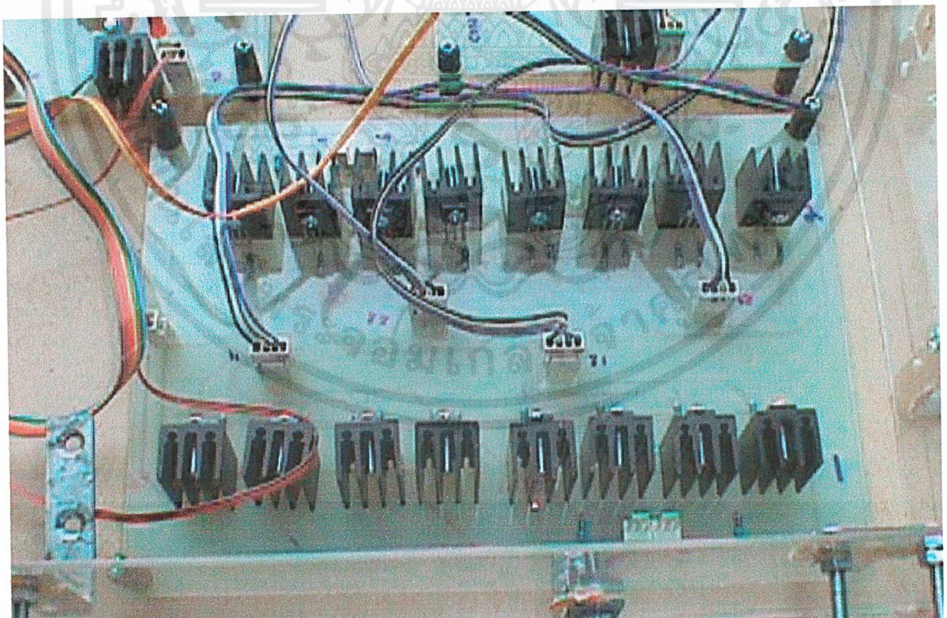
วงจรที่ประกอบด้วย IC6, IC5, IC7 และ Q9-Q16 ก็มีลักษณะการทำงานเหมือนกับวงจรที่กล่าวมาข้างต้นโดยจะใช้ในการควบคุมสเต็ปปิ้งมอเตอร์ตัวที่ 2



รูปที่ 3.1 วงจรขับสเต็ปปิ้งมอเตอร์



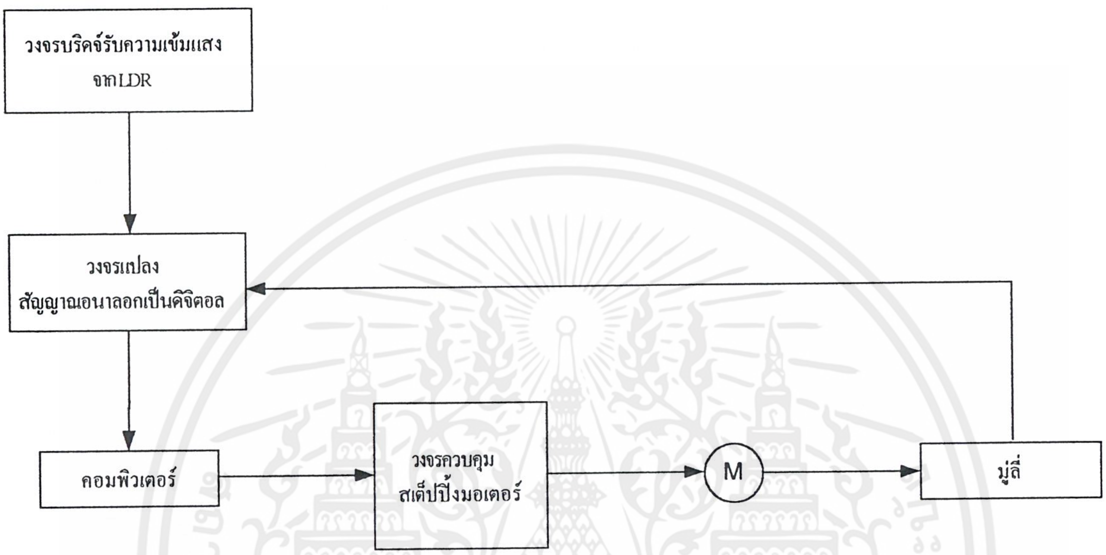
รูปที่ 3.2 ส่วนรับ-ส่งจากพอร์ตขนานและส่วนจัดสัญญาณให้กับวงจรขับสเต็ปปิ้งมอเตอร์



รูปที่ 3.3 วงจรขับสเต็ปปิ้งมอเตอร์

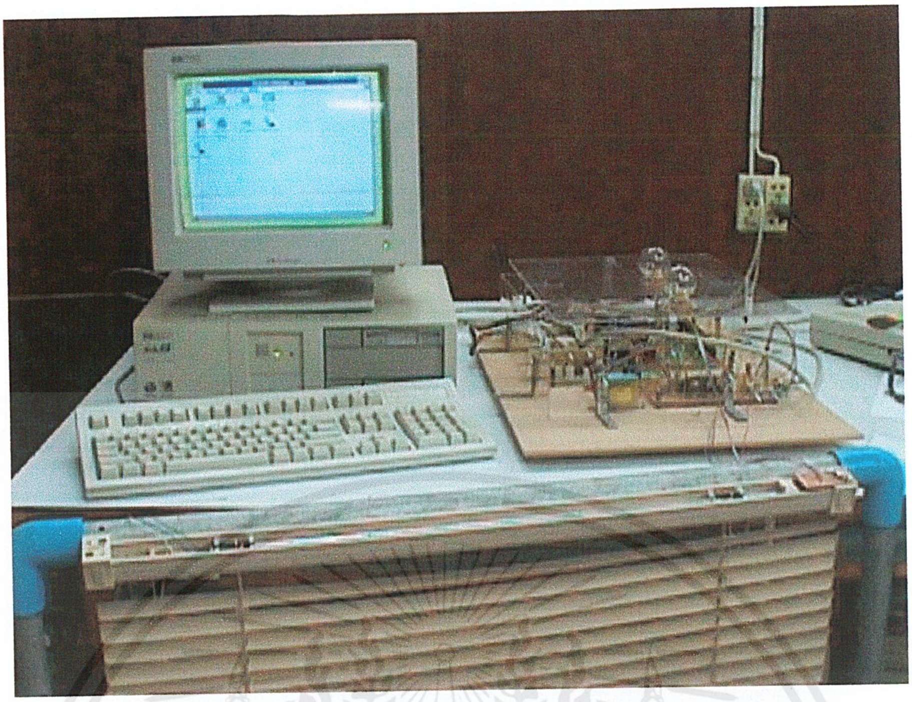
3.2 การประยุกต์ใช้งาน

3.2.1 หลักการทำงานของมู่ลี่อัตโนมัติ

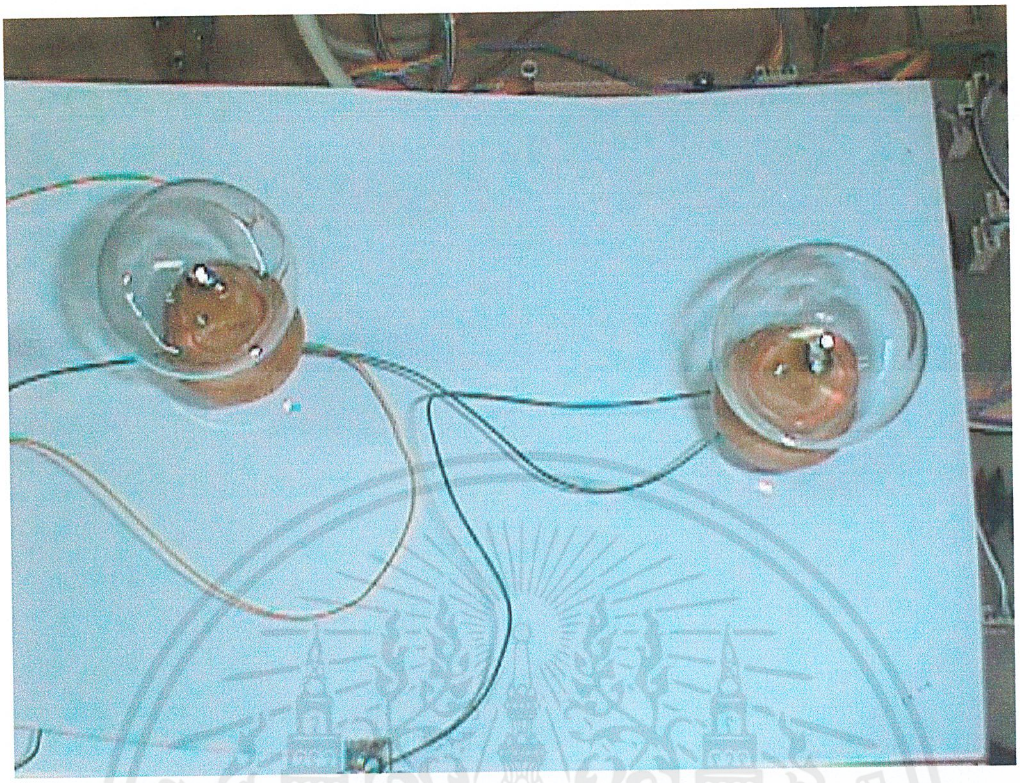


รูปที่ 3.4 บล็อกไดอะแกรมการทำงานของวงจรควบคุมมู่ลี่อัตโนมัติ

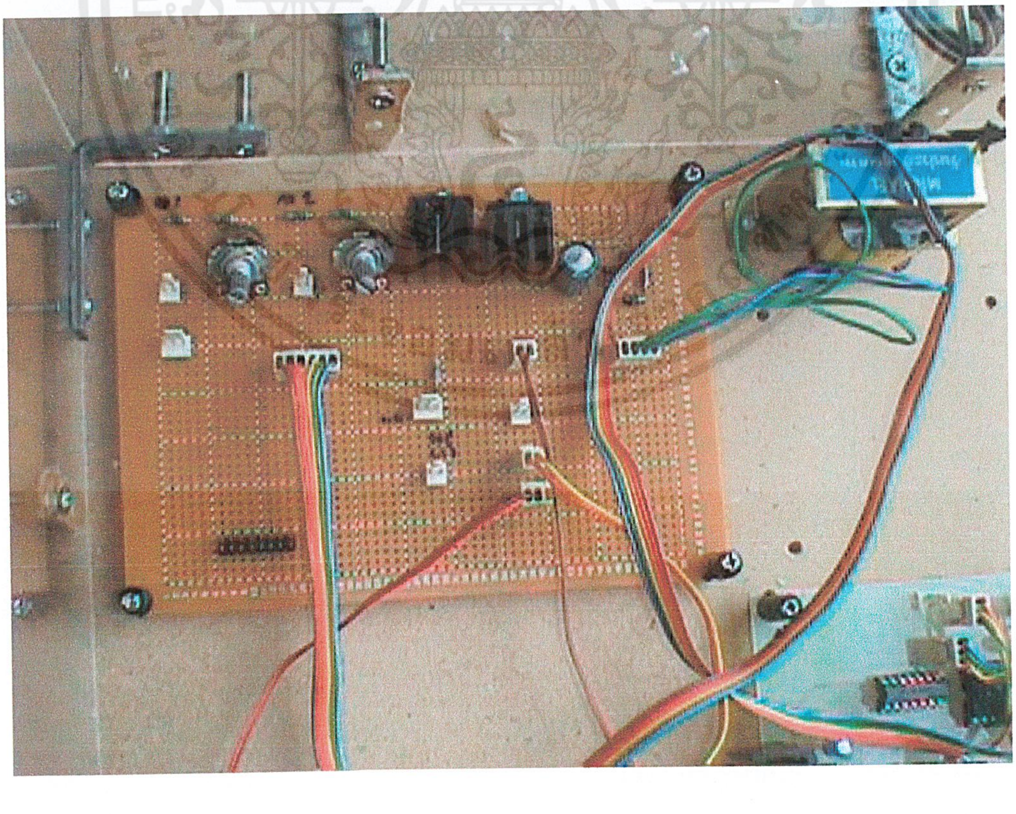
จากรูปสามารถอธิบายการทำงานของการคุมคุมมู่ลี่แบบอัตโนมัติซึ่งมีรูปถ่ายจริงของการทำงานของมู่ลี่อัตโนมัติดังรูปที่ 3.5 และมีรูปถ่ายจริงของวงจรควบคุมการทำงานของมู่ลี่อัตโนมัติดังรูปที่ 3.10 (ก)และ(ข)ได้ว่า เริ่มต้นการทดลองเมื่อวงจรบริจด์รับค่าแรงดันที่แปรตามปริมาณแสงจากแอลดีอาร์ซึ่งมีภาพถ่ายวงจรบริดจ์และแอลดีอาร์ดังรูปที่ 3.7 และ 3.6 ตามลำดับ ระดับแรงดันที่ได้มานี้จะถูกส่งเข้าไปยังวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลขนาด 8 บิตซึ่งมีรูปถ่ายจริงดังรูปที่ 3.8 ซึ่งจะมีความแตกต่างของระดับสัญญาณ 256 ระดับ เนื่องจาก ADC0809 ที่ซึ่งใช้เป็นตัวแปลงสัญญาณสามารถรับอินพุทได้ทั้งหมด 8 ค่าแต่รับได้ทีละค่า จึงต้องทำการกำหนดว่าจะรับค่าแรงดันมาจากช่องสัญญาณตัวใดก่อนโดยการกำหนดจากขาคอนโทรลของพอร์ตขนานขาที่ 14 และ 16 ให้เป็นตัวเลือกตำแหน่งการรับสัญญาณ โดยกำหนดให้ ขา 14 = A และ ขา 16 = B ซึ่งมีการเลือกดังนี้ ถ้า AB = 00 ให้รับอินพุตที่ ขา IN0 ถ้า AB = 01 ให้รับอินพุตที่ ขา IN1 ข้อมูลที่ถูกป้อนเข้านี้จะถูกเปลี่ยนจากข้อมูลแบบอนาลอกเป็นข้อมูลแบบดิจิตอล โดยที่ขา IN0 เป็นอินพุทจากแอลดีอาร์ และ IN1 เป็นอินพุทจากโพเทนชิโอห์มมิเตอร์ซึ่งทำหน้าที่เป็นตัวควบคุมป้อนกลับของตัวมู่ลี่ซึ่งมีรูปถ่ายจริงของโครงสร้างมู่ลี่ดังรูปที่ 3.9



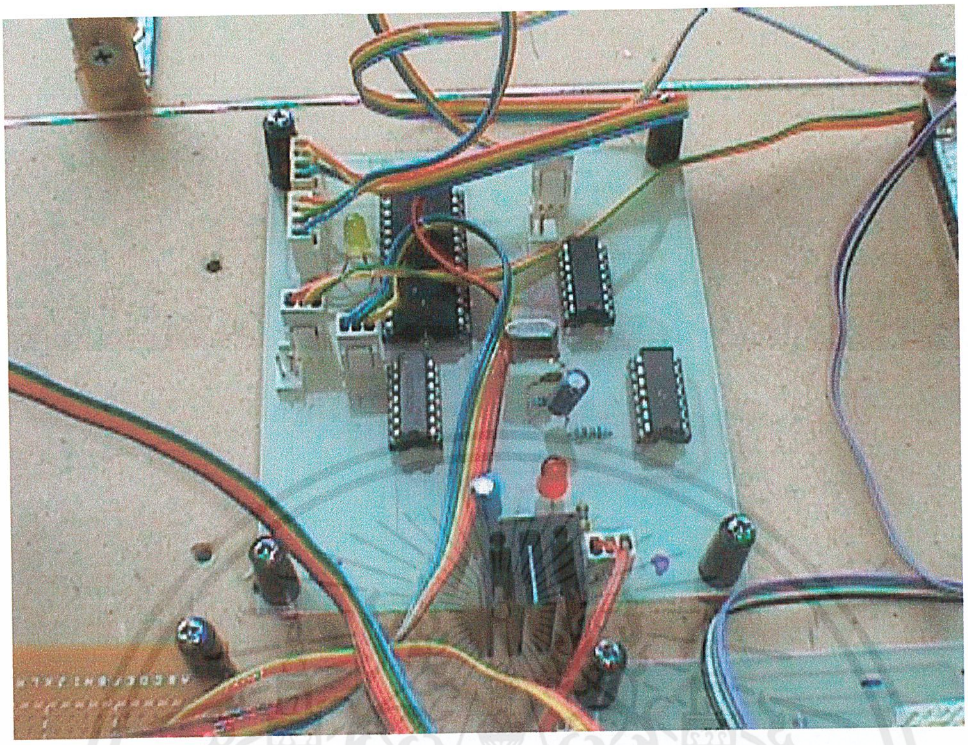
รูปที่ 3.5 ภาพการทำงานของมู่ลี่อัตโนมัติ



รูปที่ 3.6 ตัวต้านทานไวแสงที่ใช้ในบอร์ด

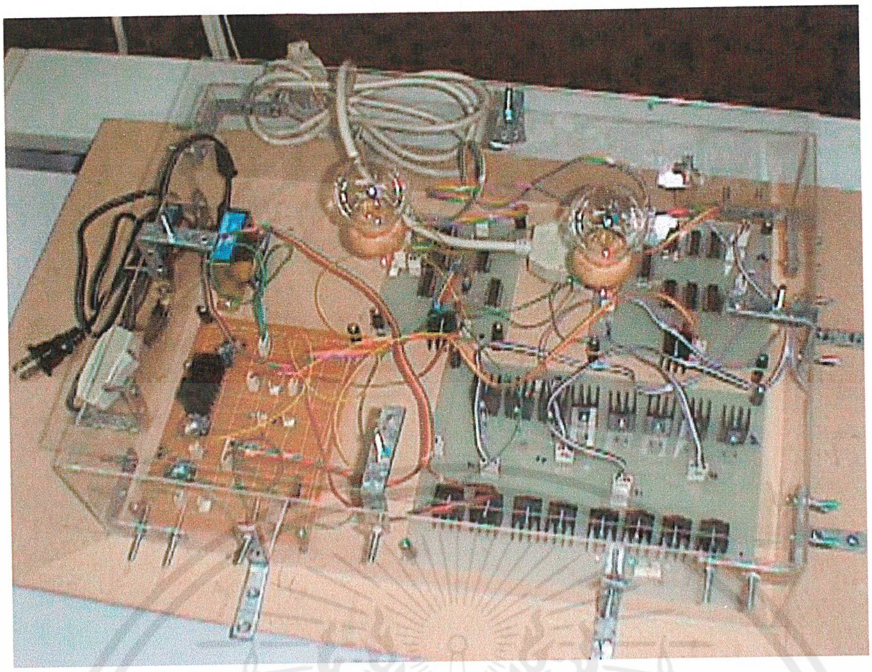


รูปที่ 3.7 วงจรบริดจ์และวงจรวงจรเปลี่ยนระดับแรงดัน

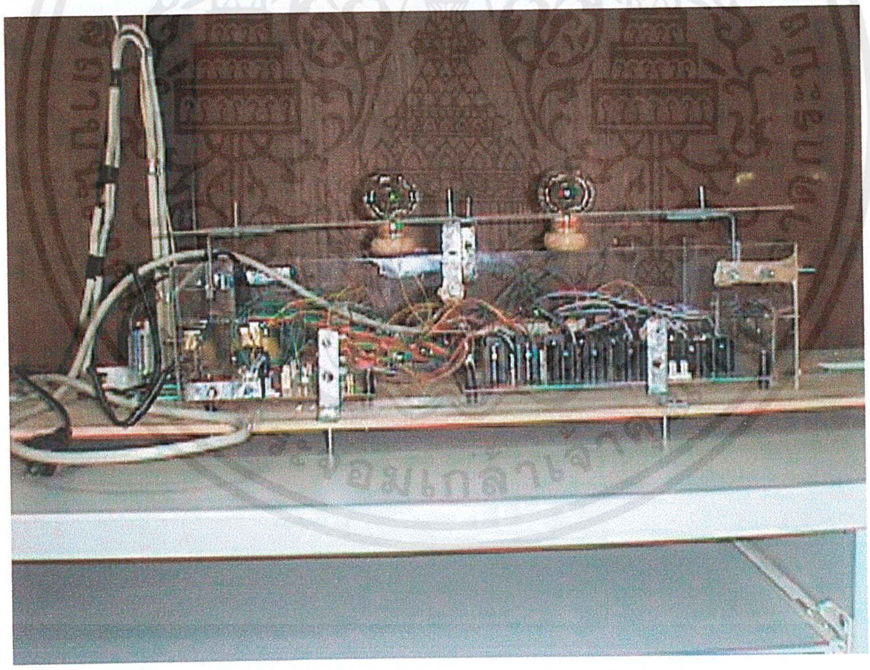


รูปที่ 3.8 วงจรแปลงสัญญาณข้อมูลอนาล็อกเป็นดิจิตอล

รูปที่ 3.9 ตัวโครงสร้างของมู่ลี่อัตโนมัติ



(ก)



(ข)

รูปที่ 3.10 วงจรควบคุมการทำงานของมู่ลี่อัตโนมัติ

(ก) ภาพถ่ายด้านบน

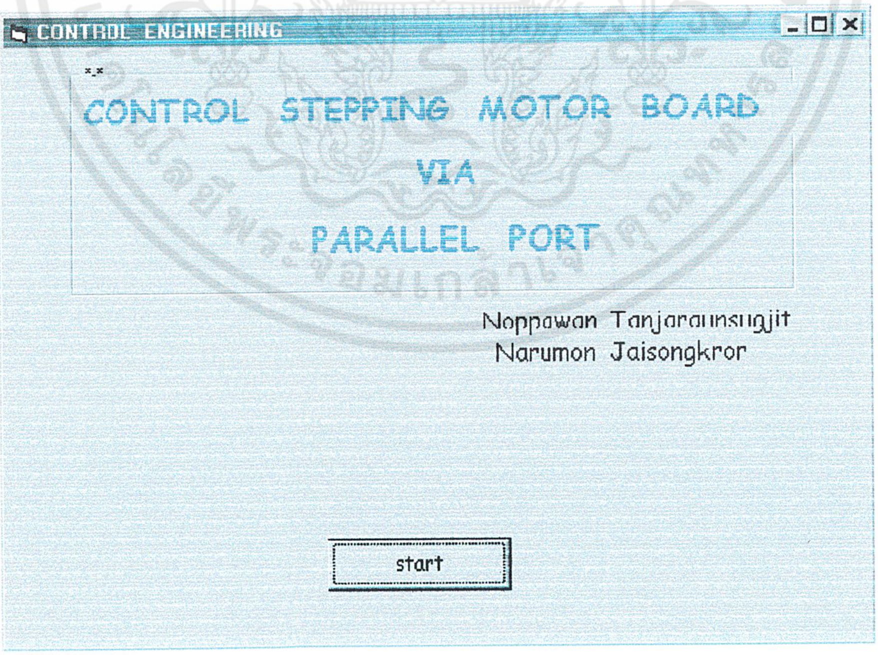
(ข) ภาพถ่ายด้านข้าง

การทำงานของตัวเปลี่ยนข้อมูลจากอนาลอกเป็นข้อมูลดิจิตอล ต้องกำหนดขาคอนโทรล ของพอร์ตขนานขาที่ 17 ให้ ขาที่ 17= WR ซึ่งในตอนเริ่มต้นการทำงานต้องกำหนดให้ WR = 0 ก่อน จากนั้นตัวเปลี่ยนข้อมูลจากอนาลอกเป็นข้อมูลดิจิตอลจะส่งค่าดิจิตอลที่แปลงแล้วออกมาซึ่ง ข้อมูลที่ถูกแปลงออกมาจะเป็นลักษณะข้อมูลขนาน 8 บิต แต่เนื่องจากที่พอร์ตขนานมีขาอินพุต เพียง 5 ขา ฉะนั้นเราจึงต้องนำข้อมูลขนานจึงถูกนำมาใช้เพียง 5 บิต หรือ 32 ระดับ ซึ่งสัญญาณ 5 บิตนี้จะใช้เป็นตัวกำหนดการทำงานของวงจรขับสเต็ปปิ้งมอเตอร์ต่อไป

โดยนำค่าที่ได้จากขา IN0 ค่าในช่วงที่กำหนดเมื่อนำมาเปรียบเทียบกันแล้วนำผลที่ได้มา ควบคุมให้สเต็ปปิ้งมอเตอร์ทำงาน เมื่อมอเตอร์ทำงานเสร็จแล้วจึงทำการรับค่าอินพุตที่ ขา IN1 นำ มาเปรียบเทียบกับค่าที่คำนวณจากการเคลื่อนที่ของมอเตอร์เพื่อให้มอเตอร์ขยับไปในขนาดและทิศ ทางที่ต้องการและจะทำเช่นนี้ต่อไปเรื่อยๆจนกว่าจะมีคำสั่งให้หยุดการทำงาน

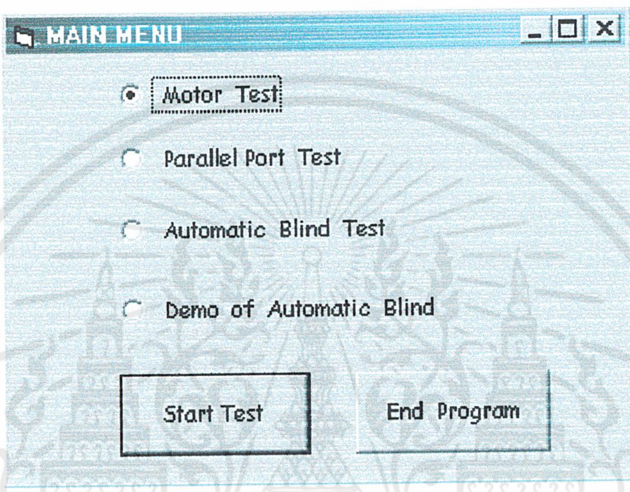
3.2.2 โปรแกรมควบคุมการทำงานของมู่ลี่อัตโนมัติ

ในส่วนของการอินเตอร์เฟสจะใช้โปรแกรมวิชวลเบสิก โดยในการทำงานจะแบ่งเป็นส่วน ทดสอบต่าง ๆ เพื่อให้สามารถเข้าใจการทำงานของตัวโครงงานได้ง่ายและสามารถนำไปประยุกต์ ใช้งานได้อย่างสะดวก

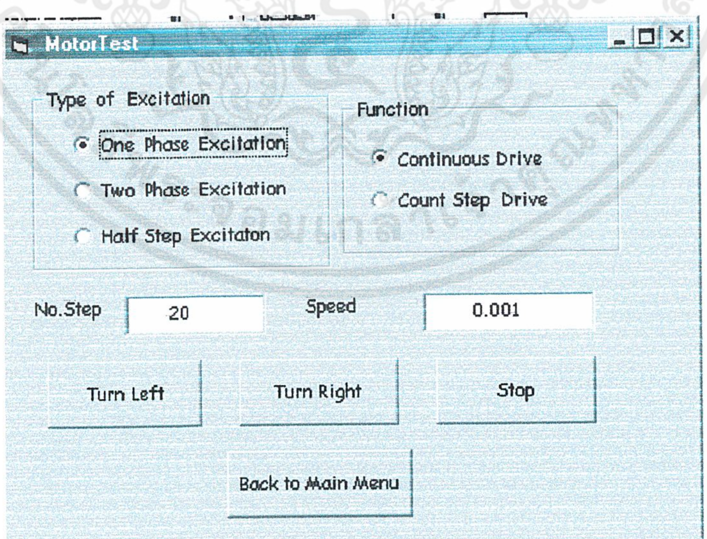


รูปที่ 3.11 หน้าจอแรกของการอินเตอร์เฟส

จากรูปที่ 3.11 แสดงการติดต่อกับโปรแกรมเพื่อทำการทดสอบการทำงานต่าง ๆ ซึ่งเมื่อเข้าไปในส่วนของการเลือกการทดสอบแล้วดังรูปที่ 3.12 จะเห็นว่ามีฟังก์ชันให้เลือกใช้ในการทดสอบการทำงานอยู่ 3 ส่วนและส่วนของการจำลองแสดงการทำงานของมูลี่อัตโนมัติ ใน 3 ส่วนแรกนั้นจะใช้เพื่อทดสอบส่วนต่าง ๆ ของโครงงานได้ ได้แก่ การทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ การทดสอบการรับส่งค่าของพอร์ตขนานและการทดสอบการควบคุมมูลี่อัตโนมัติ

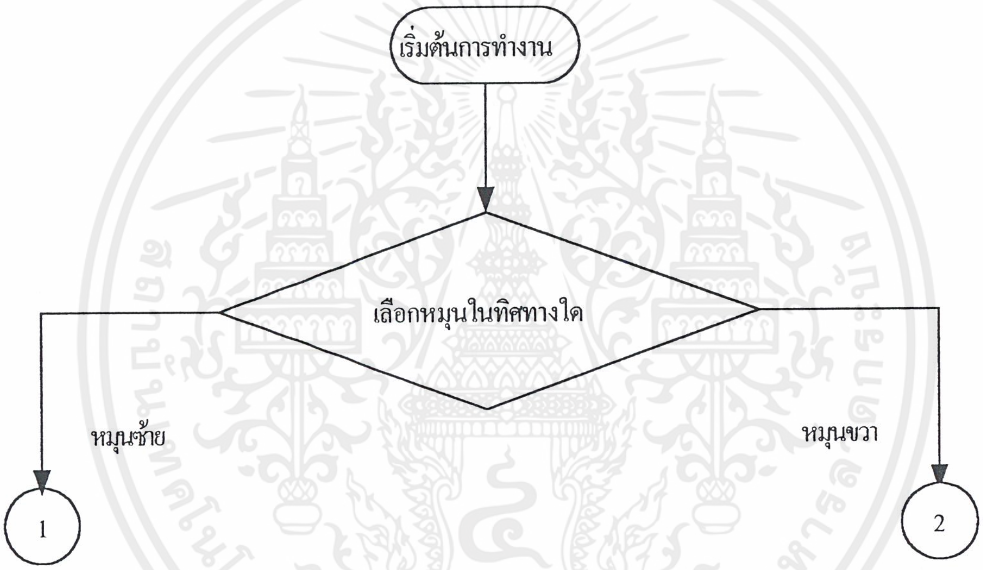


รูปที่ 3.12 หน้าจอหลักในการเลือกการทำงาน

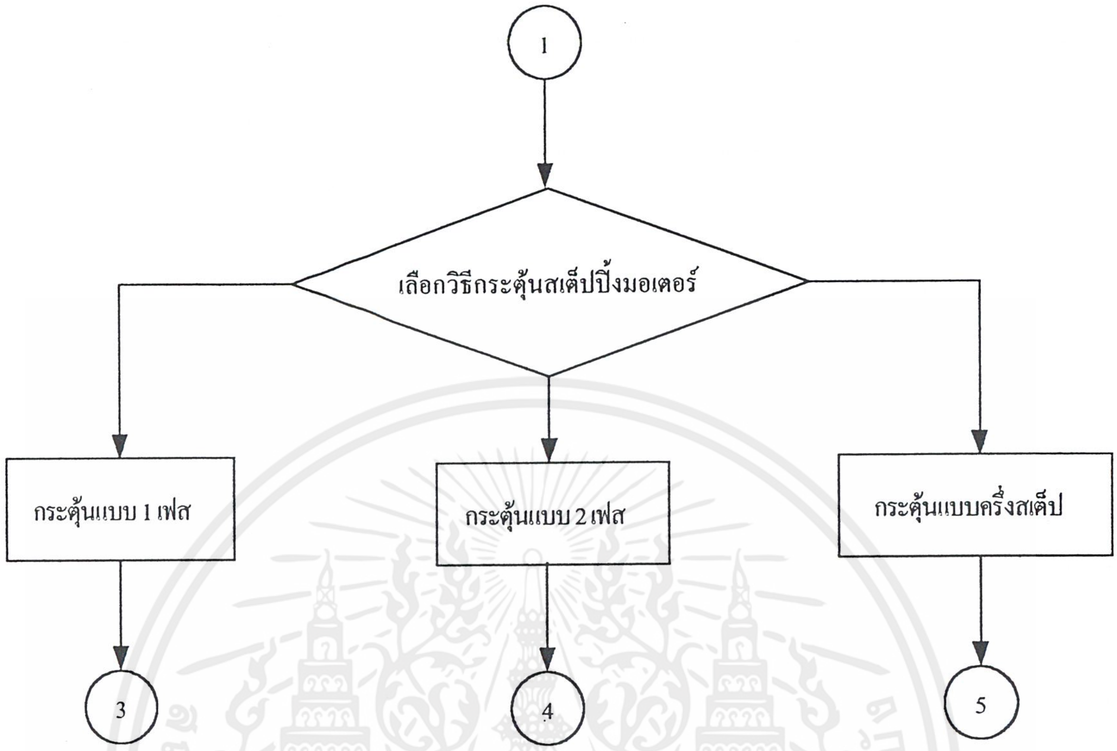


รูปที่ 3.13 หน้าจอในโปรแกรมทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์

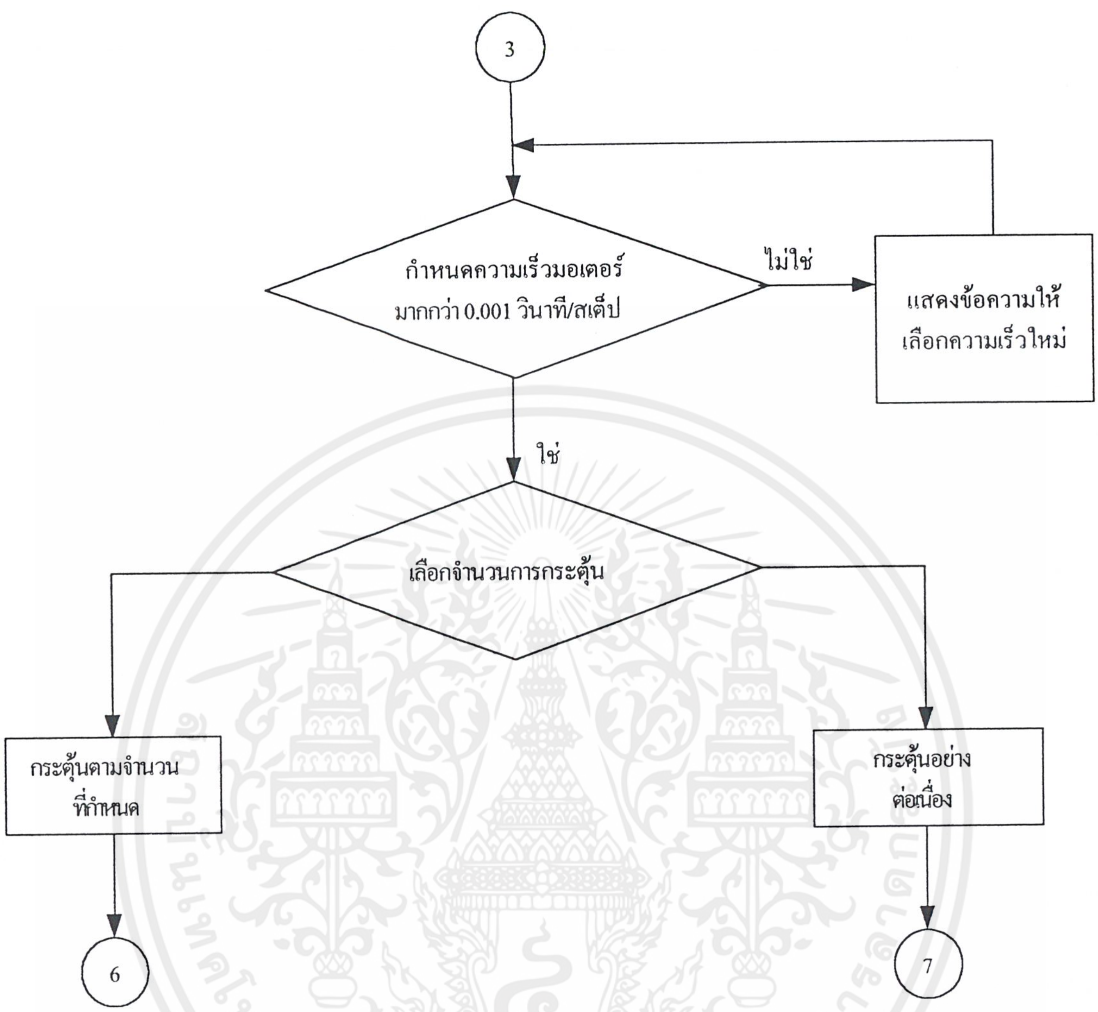
จากรูปที่ 3.13 แสดงหน้าจอการอินเตอร์เฟสของการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ โดยมีฟังก์ชันการทำงานให้เลือกว่าต้องการกระตุ้นมอเตอร์แบบใดและต้องการให้หมุนไปจำนวนรอบเท่าไรหรือต้องให้หมุนจนกว่าจะมีคำสั่งให้หยุดทำงาน นอกจากนี้ยังสามารถกำหนดความเร็วของมอเตอร์ได้ด้วย เพื่อหาค่าที่เหมาะสมในการสั่งงานกับสเต็ปปิ้งมอเตอร์ ในส่วนของโปรแกรมการสั่งงานสามารถอธิบายได้ดังรูปที่ 3.14



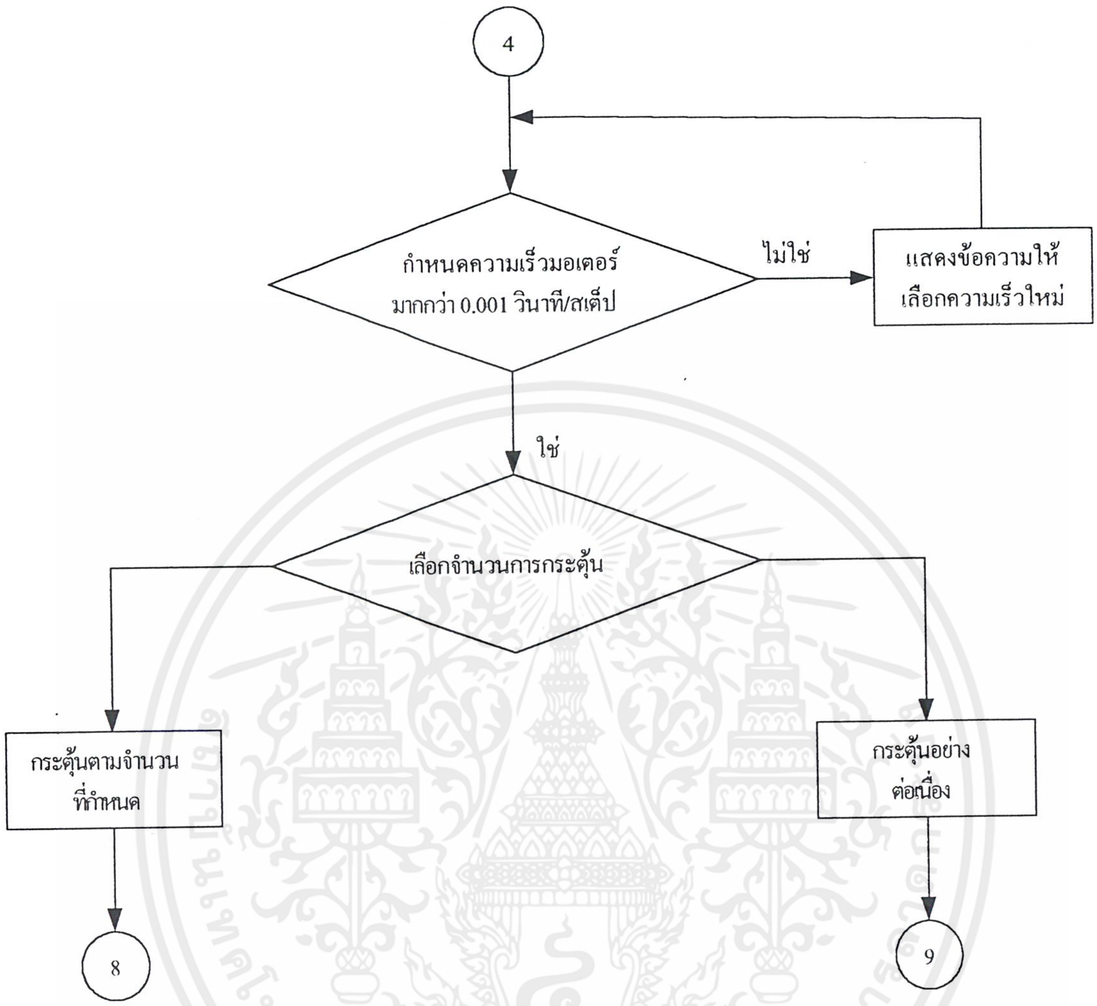
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์



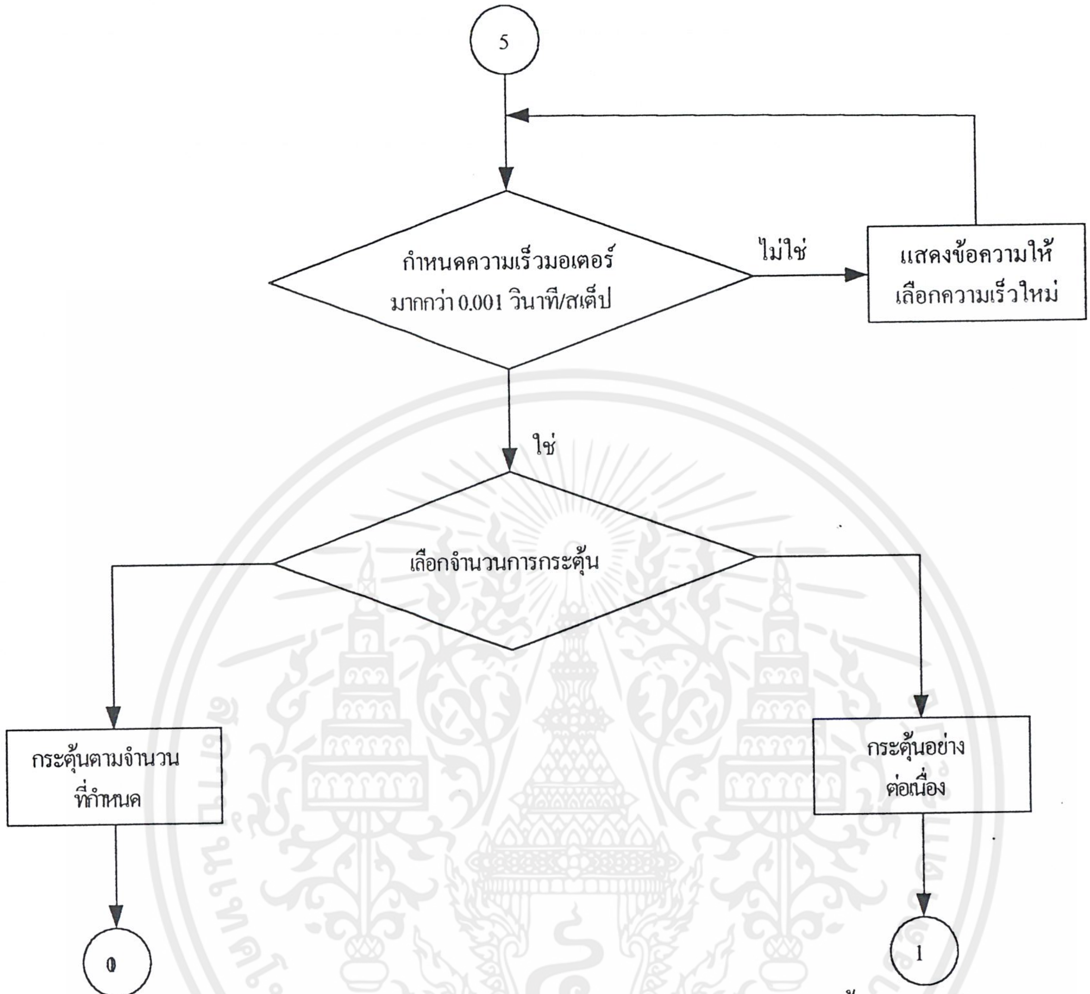
รูปที่ 3.14 โฟล์ชาร์ทแสดง โปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



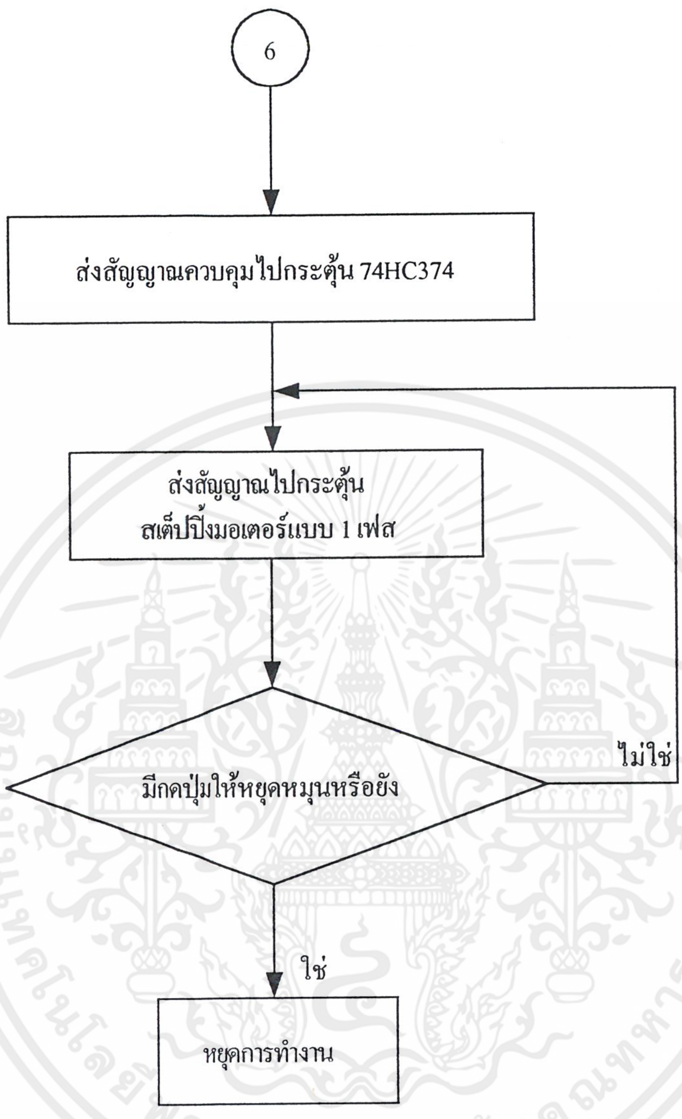
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



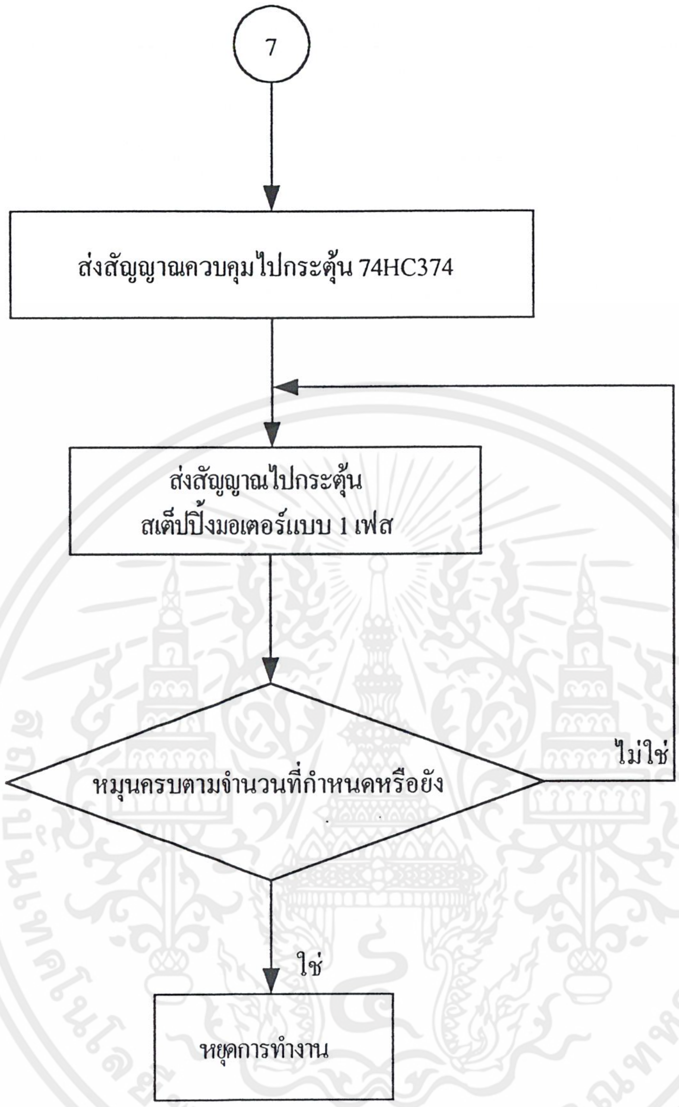
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



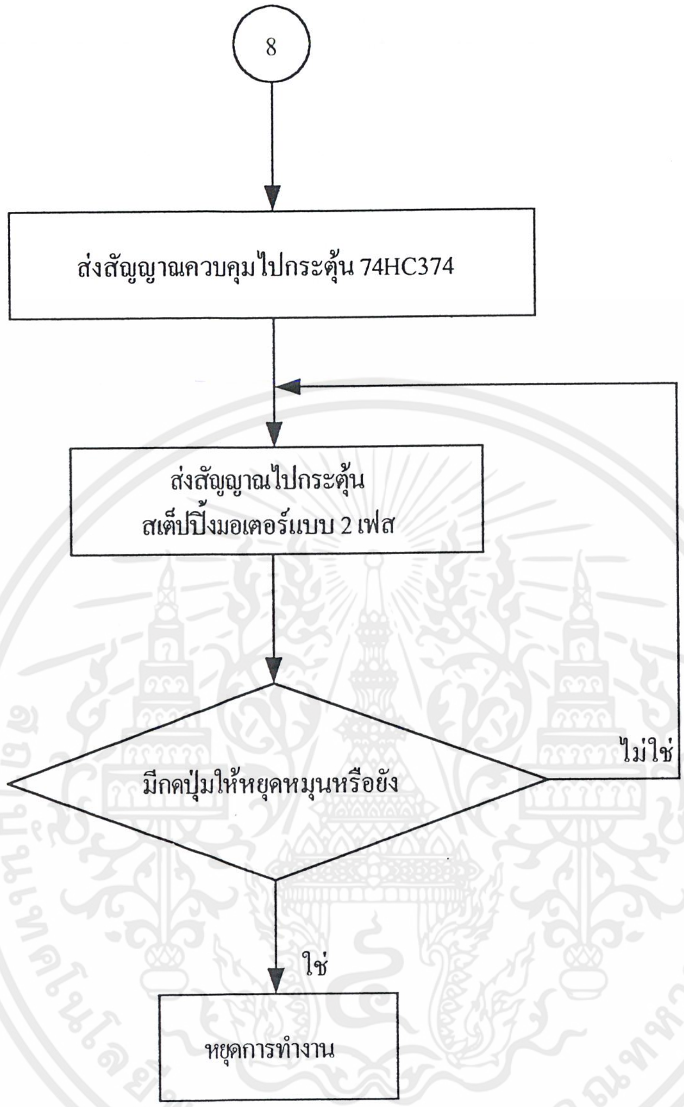
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



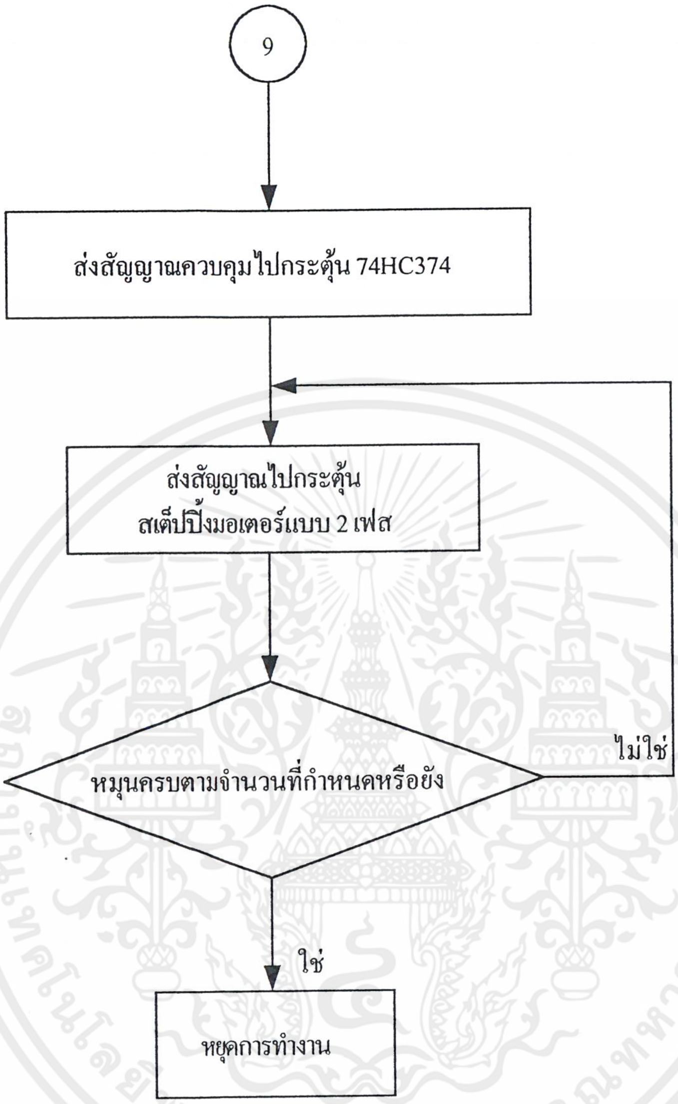
รูปที่ 3.14 โฟล์ชาร์ทแสดง โปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



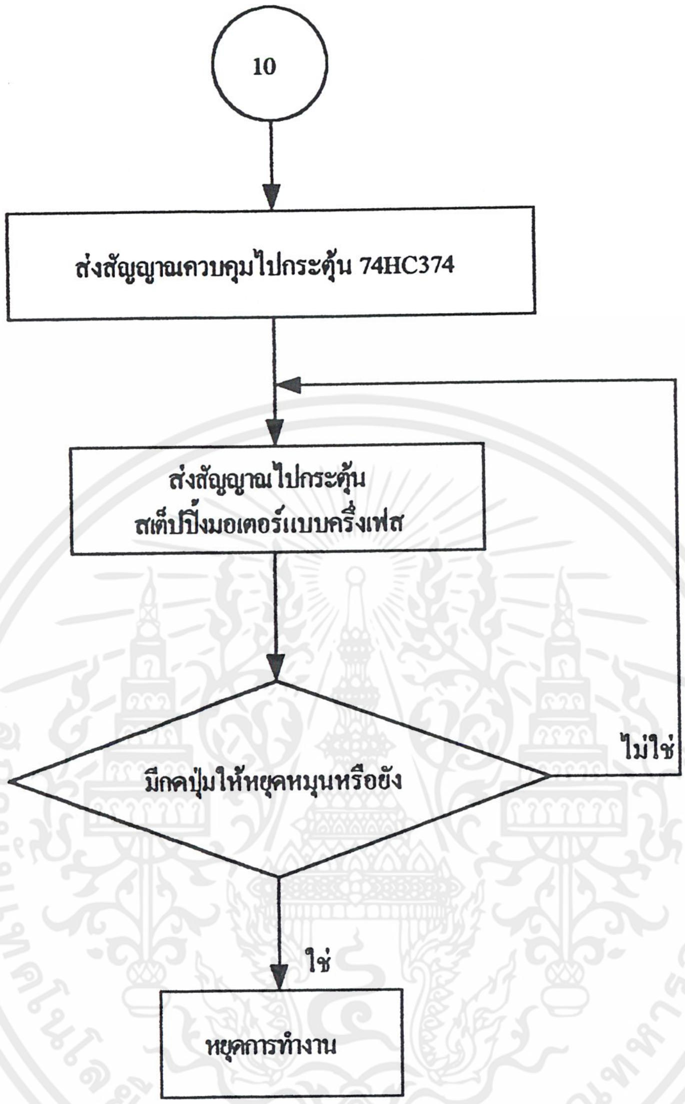
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



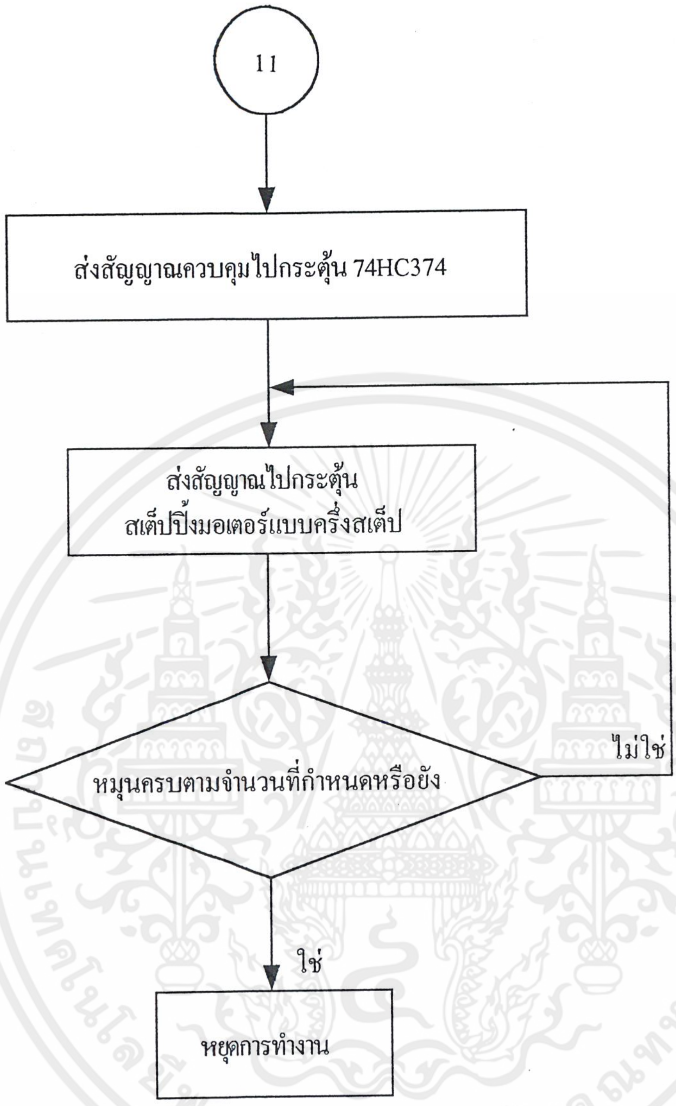
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



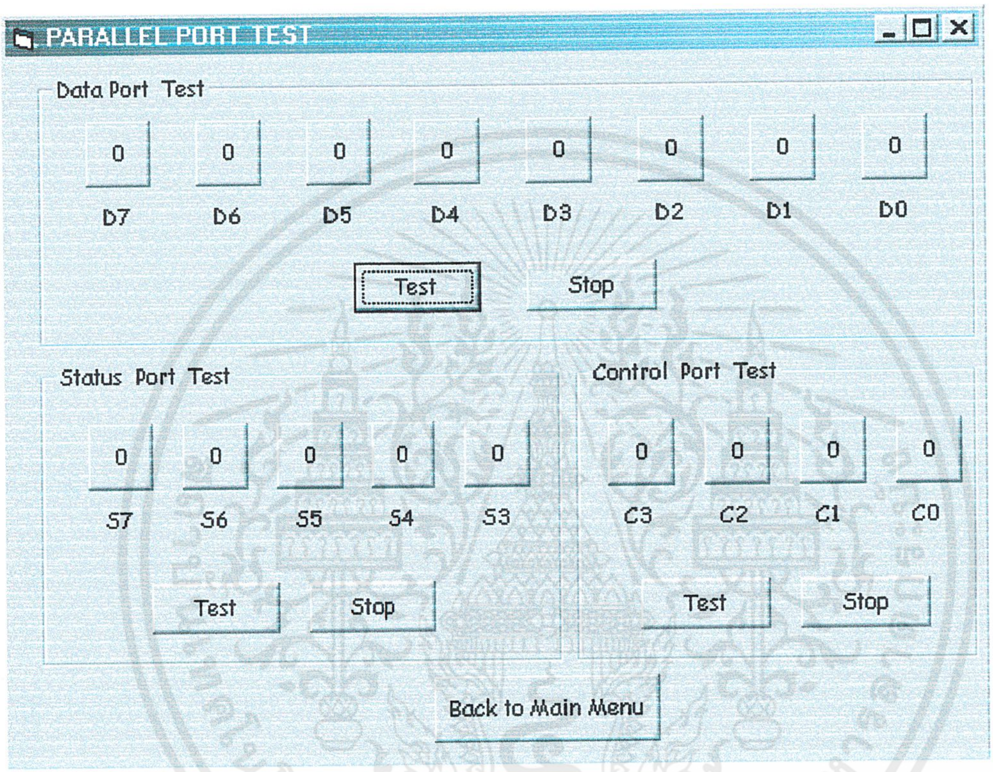
รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)



รูปที่ 3.14 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ (ต่อ)

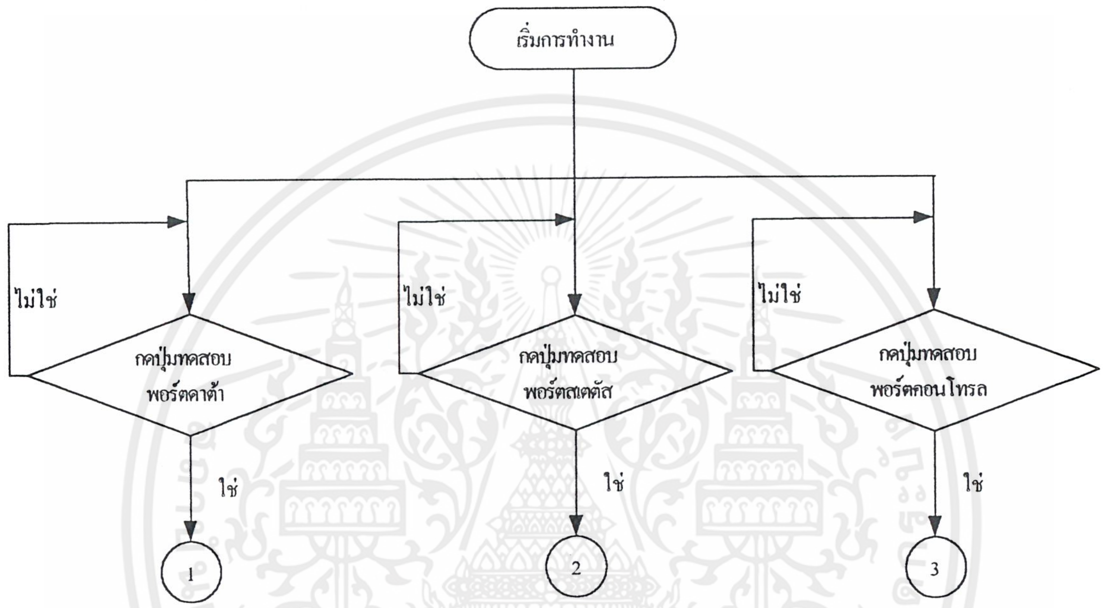
โฟล์ชาร์ทที่แสดงเป็นส่วนการของการสั่งมอเตอร์ให้หมุนไปทางซ้าย ส่วนการหมุนขวาการมีลักษณะขั้นตอนการทำงานที่เหมือนกัน

ในส่วนของฟังก์ชันทดสอบพอร์ตขนานนั้นจะเป็นส่วนที่ใช้เช็คความถูกต้องของการรับส่งค่าเพื่อให้เข้าใจการทำงานของพอร์ตขนาน นอกจากนี้ยังสามารถใช้ในการเช็คการทำงานของมูลี่อัตโนมัติได้ด้วยว่ามีความถูกต้องตรงตามคำสั่งหรือไม่ หน้าจอการอินเตอร์เฟสสามารถแสดงได้ดังรูปที่ 3.15

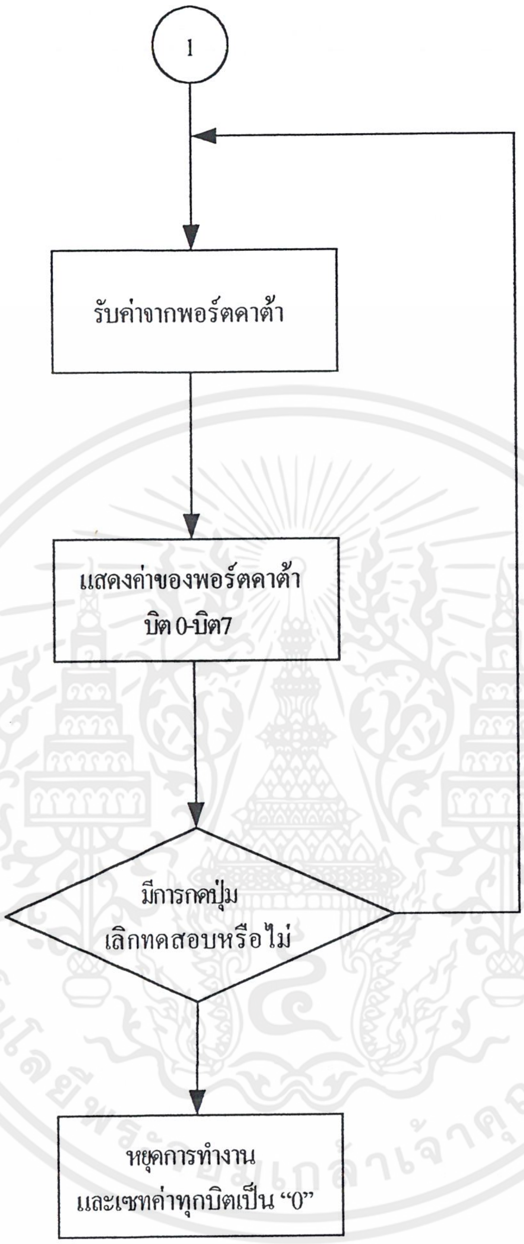


รูปที่ 3.15 หน้าจอการทดสอบพอร์ตขนาน

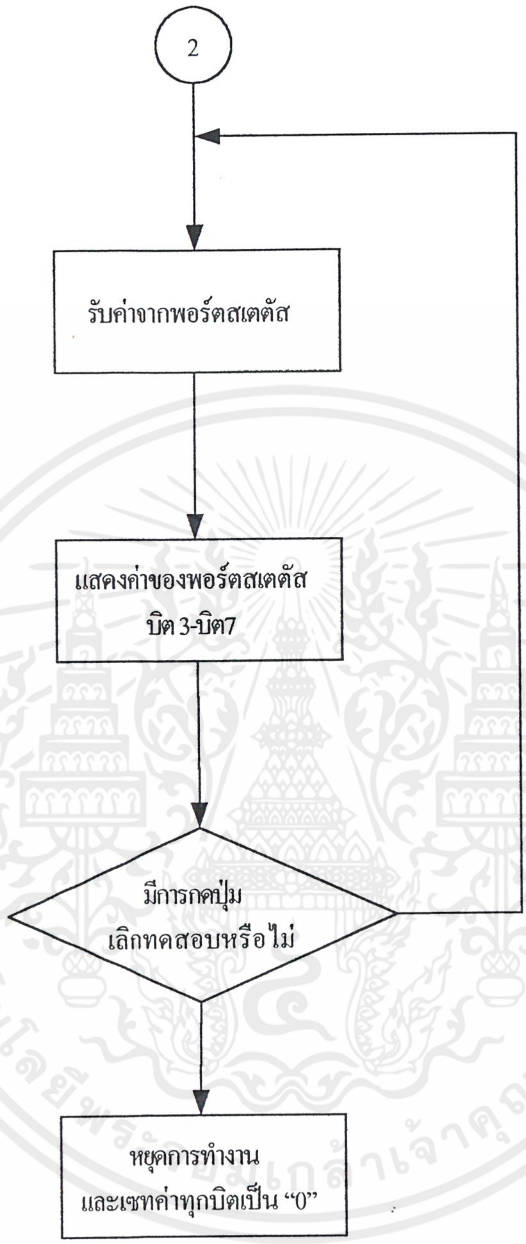
โปรแกรมการทำงานนั้นเป็นการสั่งงานอย่างง่าย ๆ ให้รับค่าจากพอร์ตมาแสดงค่าว่าสถานะของพอร์ตด้าต้า พอร์ตสเตตัสและพอร์ตคอนโทรลมีการรับส่งค่าเป็นอย่างไร โดยสามารถเขียนแสดงอธิบายได้ด้วยรูปที่ 3.16 .



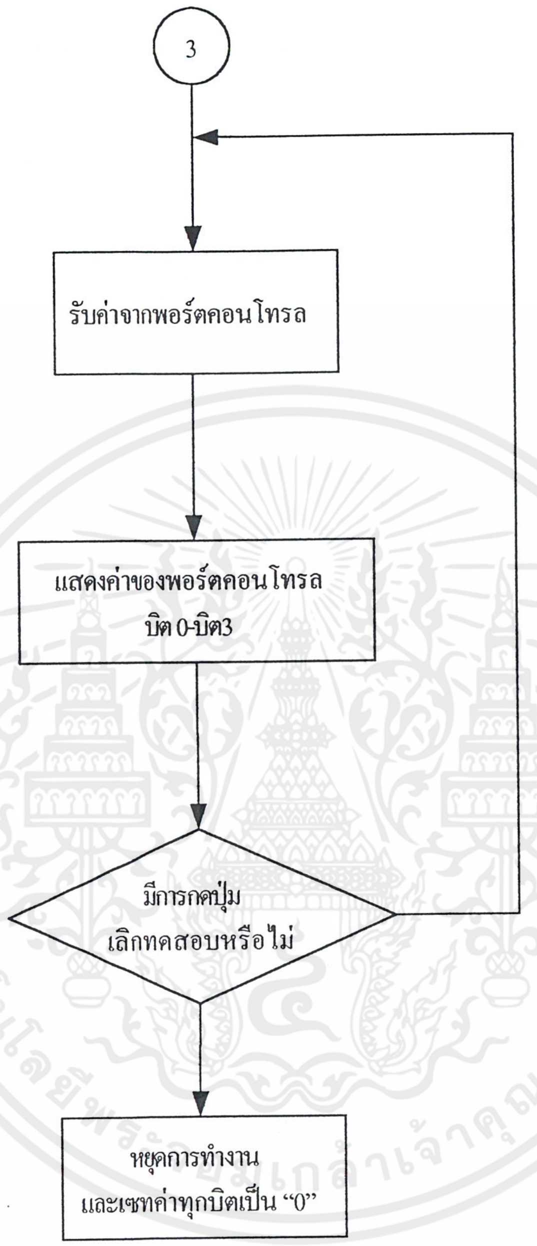
รูปที่ 3.16 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบพอร์ตขนาน



รูปที่ 3.16 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบพอร์ตขนาน (ต่อ)

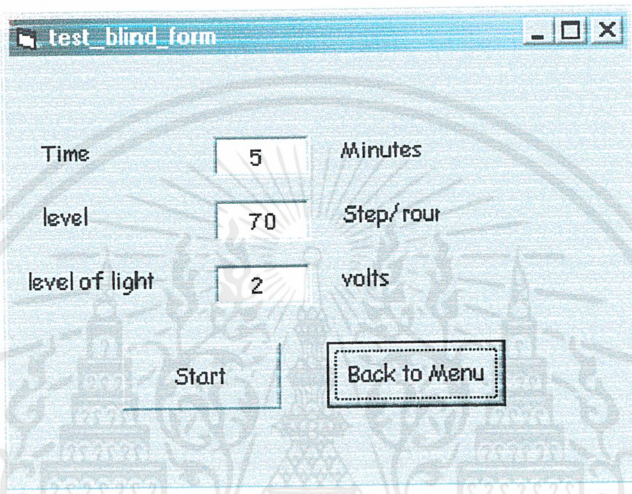


รูปที่ 3.16 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบพอร์ตขนาน (ต่อ)



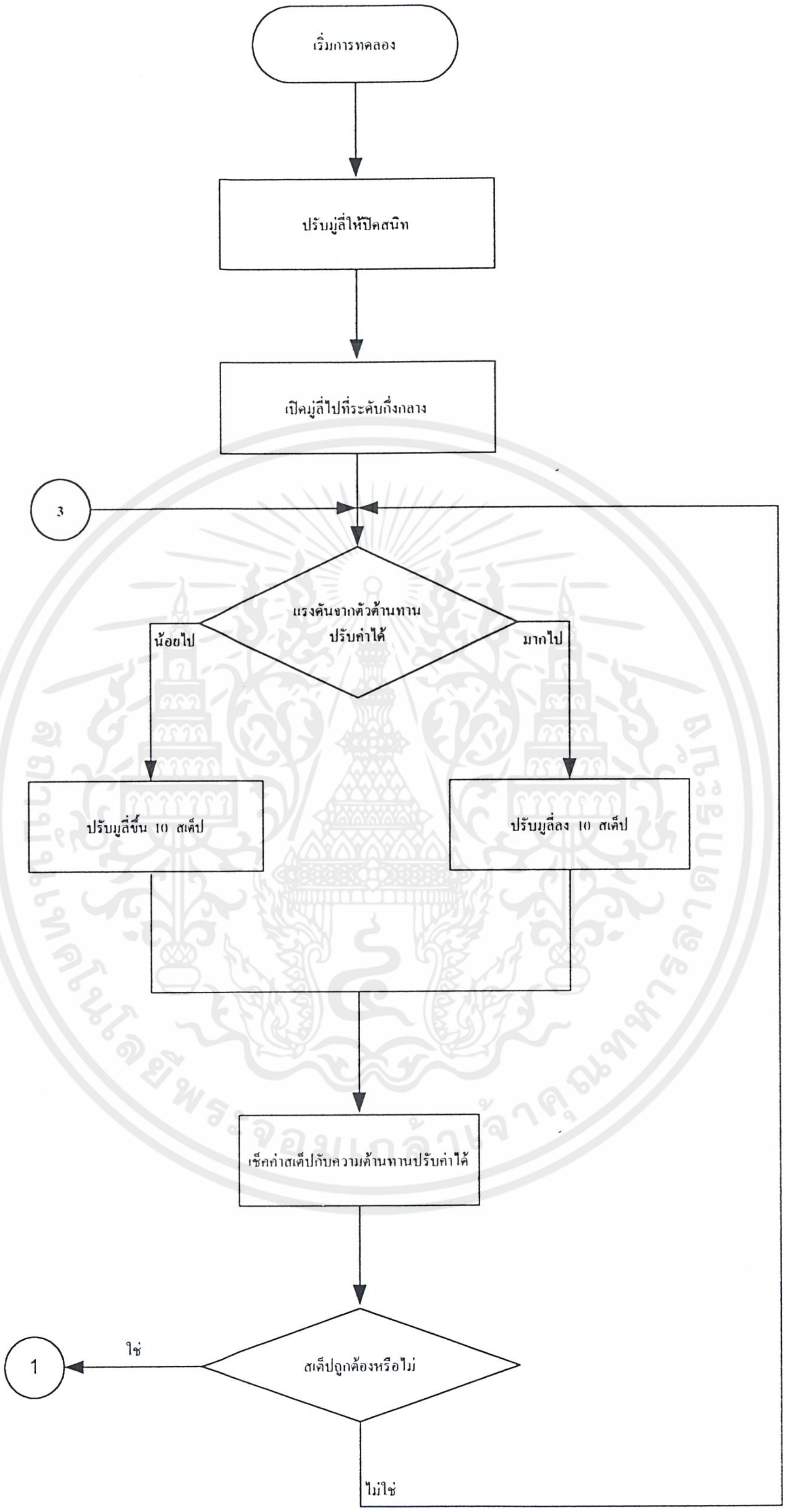
รูปที่ 3.16 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบพอร์ตขนาน (ต่อ)

ในส่วนของการทดสอบบอร์ดขนานสามารถแสดงหน้าจอการอินเตอร์เฟสได้ตามรูปที่ 3.17 หากทำการป้อนข้อมูลให้กับตัวโปรแกรมก็สามารถจะตรวจได้ว่ามู่ลี่สามารถทำงานได้อย่างไรบ้าง ส่วนนี้จะมีประโยชน์หากมีการนำมู่ลี่อัตโนมัติไปใช้งานจริงเพราะในแต่ละสถานที่ใช้ในช่วงเวลาที่ต่างกันและต้องการระดับความเข้มแสงที่แตกต่างกัน ดังนั้นการควบคุมมู่ลี่จึงแตกต่างกันไปในแต่ละสถานที่

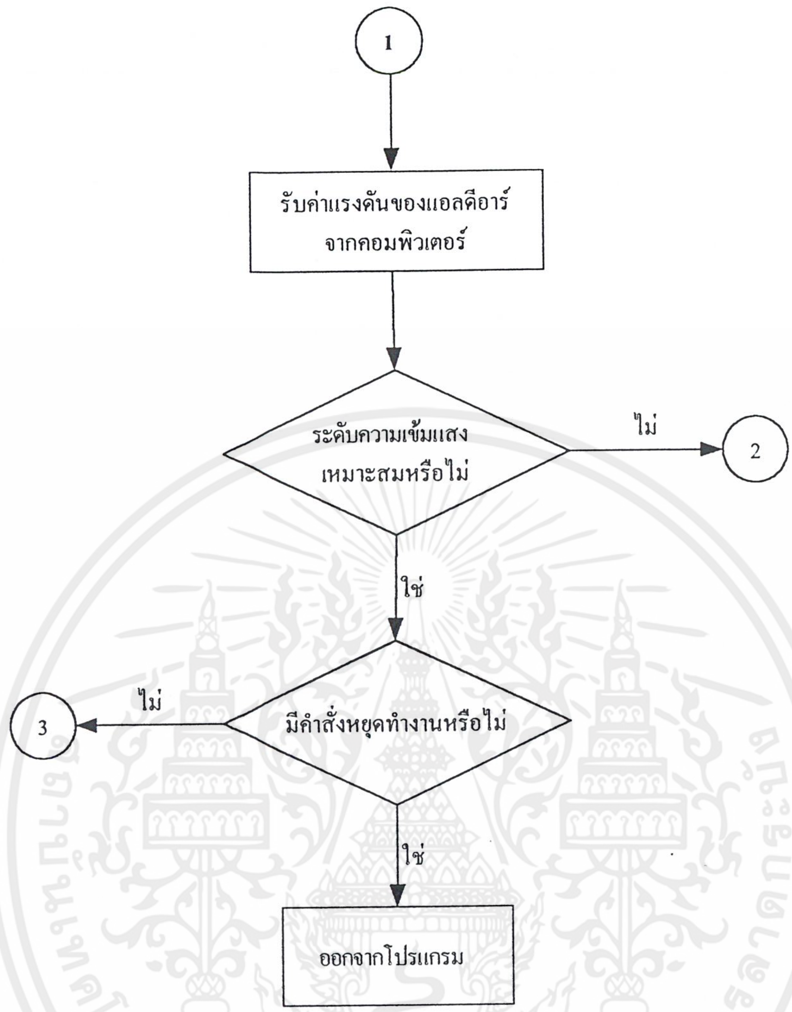


รูปที่ 3.17 หน้าจอการทดสอบมู่ลี่อัตโนมัติ

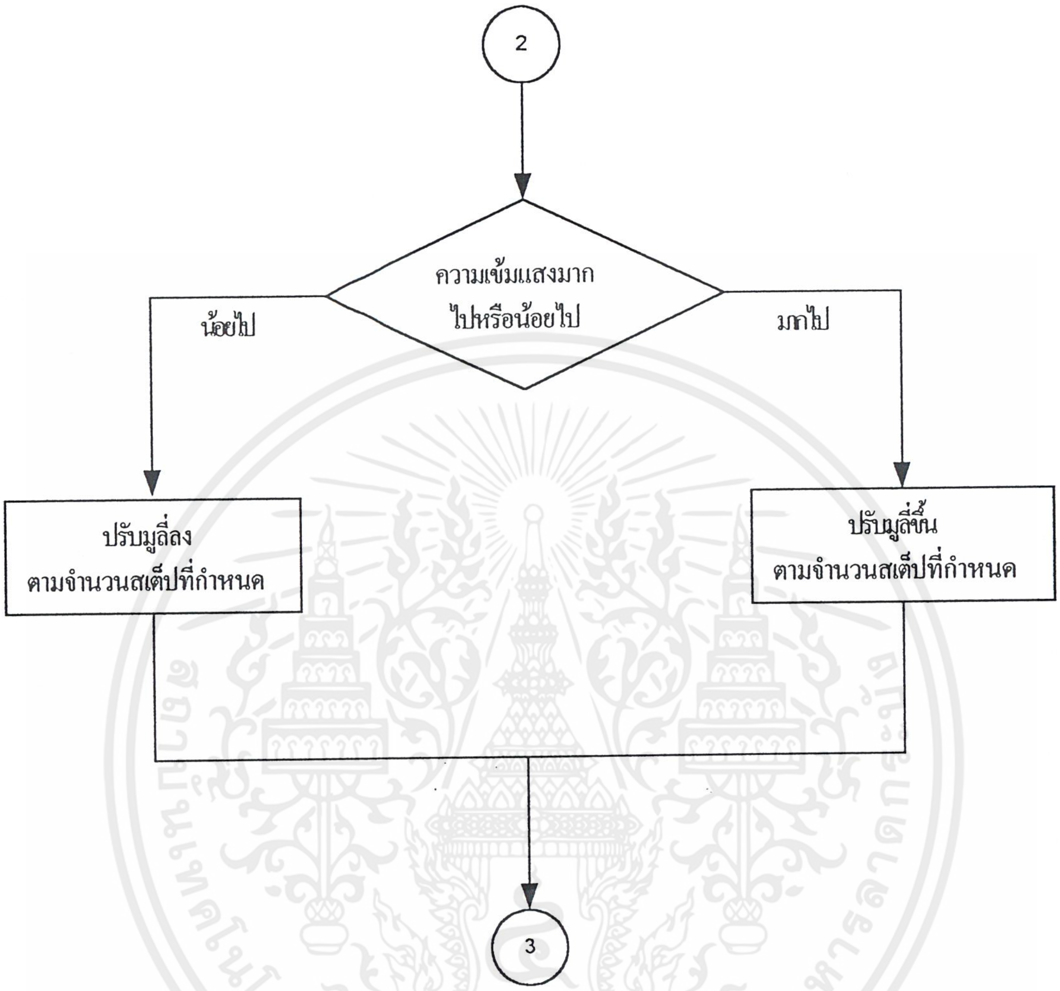
โปรแกรมการทดสอบมู่ลี่อัตโนมัติสามารแสดงได้ดังรูปที่ 3.18



รูปที่ 3.18 โฟล์ชาร์ทแสดงโปรแกรมการทดสอบมู่ลี่อัตโนมัติ

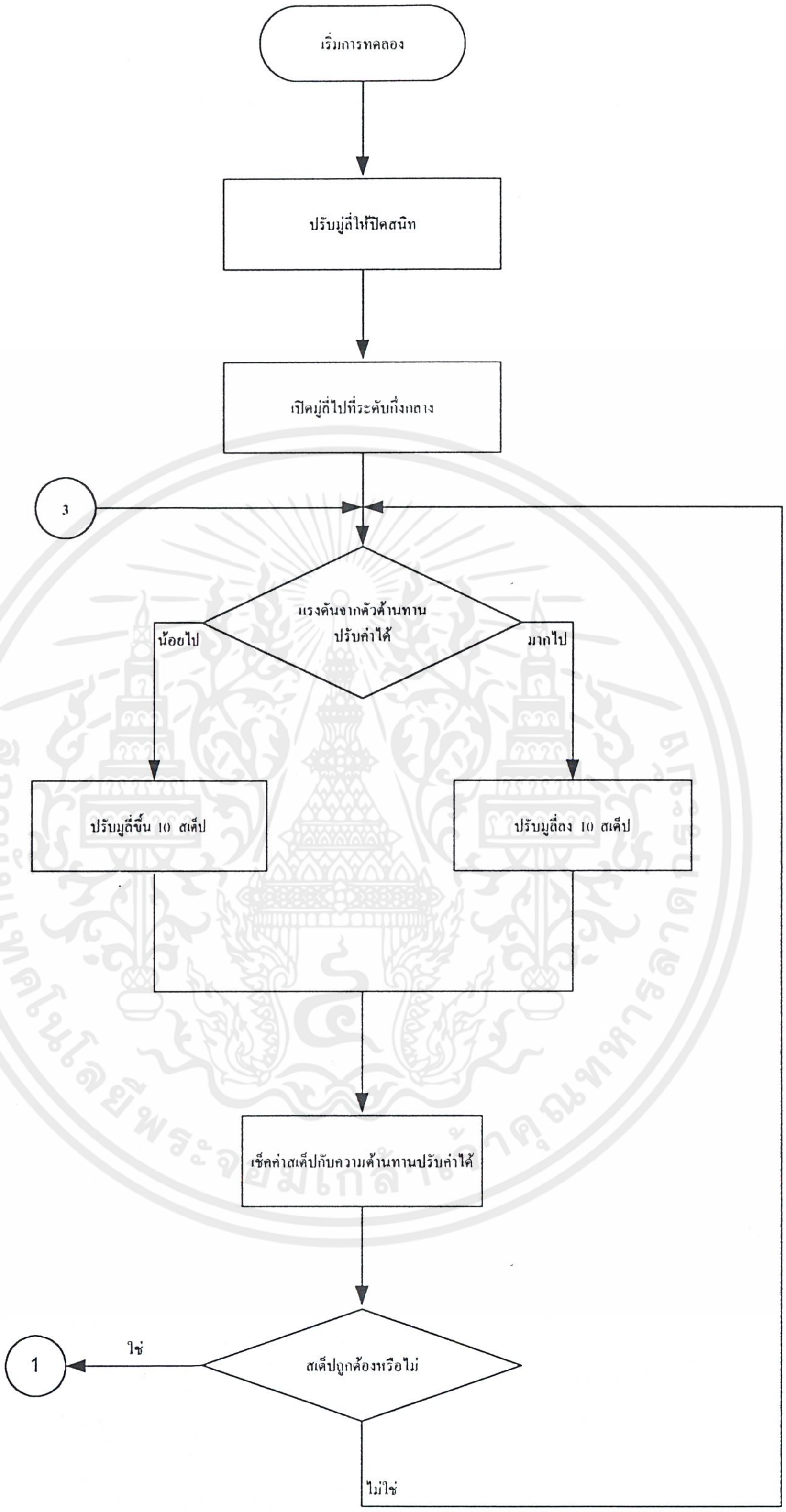


รูปที่ 3.18 โฟล์ชาร์ทแสดงโปรแกรมควบคุมมูลี่อัตโนมัติ (ต่อ)

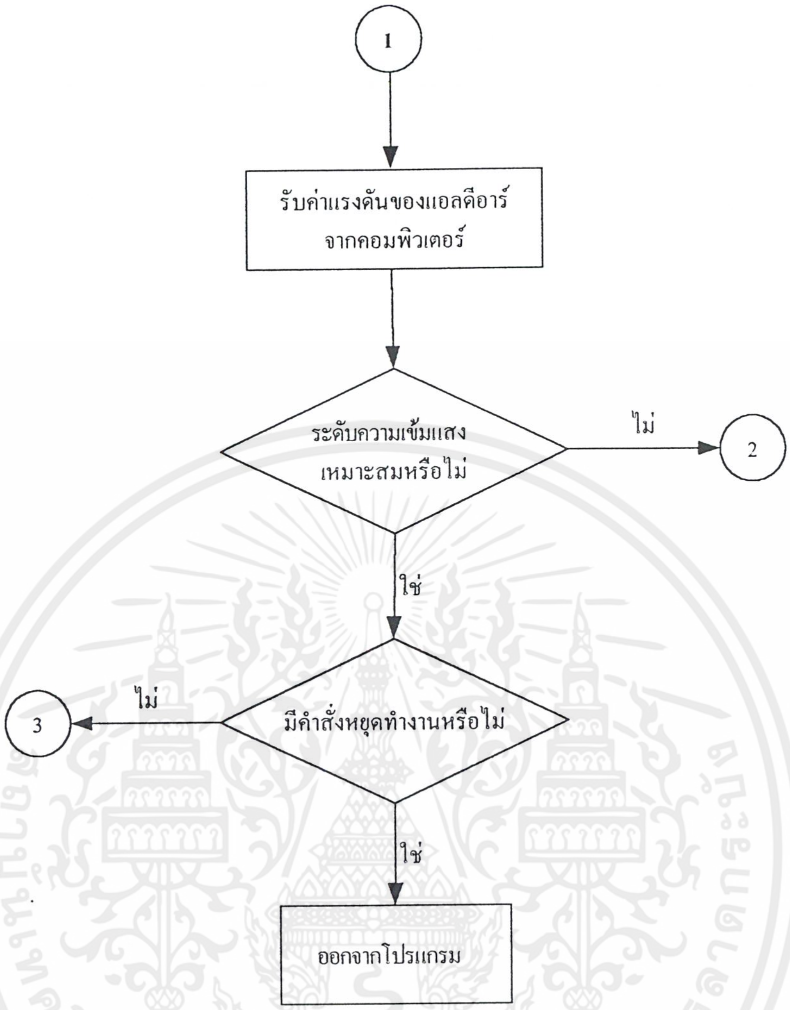


รูปที่ 3.18 โฟล์ชาร์ทแสดงโปรแกรมควบคุมมู่ลี่อัตโนมัติ (ต่อ)

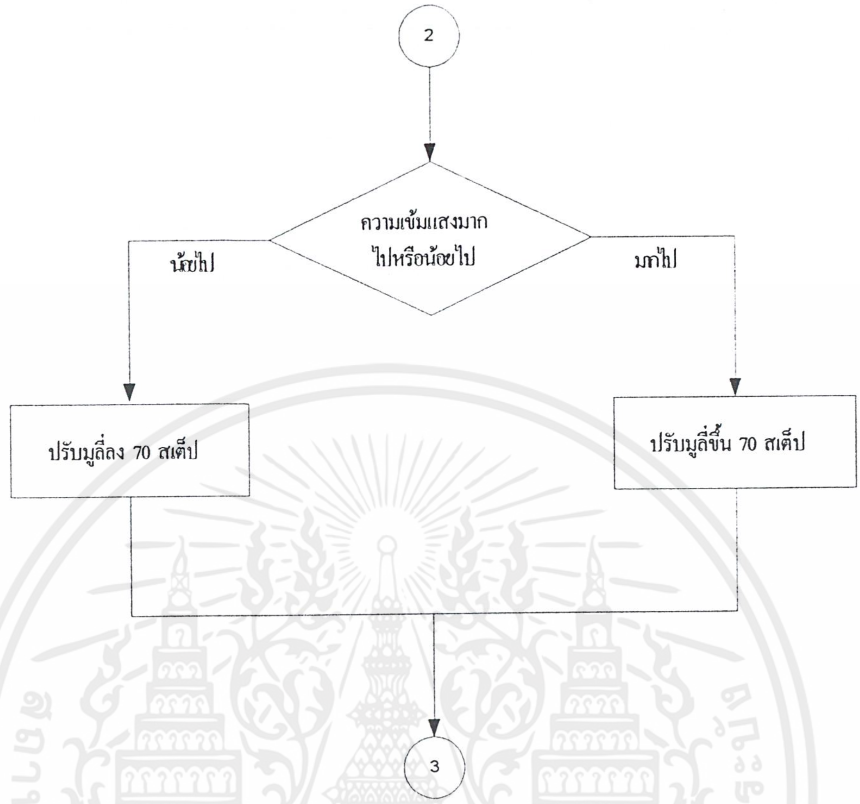
ในส่วนของการจำลองการทำงานมู่ลี่อัตโนมัติก็จะมีขั้นตอนการทำงานที่เหมือนกับในส่วนของการทดสอบมู่ลี่อัตโนมัติแต่จะแตกต่างในส่วนของการกำหนดจำนวนสเต็ปการหมุนของมอเตอร์เพื่อใช้ในการควบคุมปริมาณแสงที่ผ่านเข้ามา ดังแสดงได้ดังรูปที่ 3.19



รูปที่ 3.19 โฟล์ชาร์ทแสดงโปรแกรมการจำลองการทำงานของมู่ลี่อัตโนมัติ

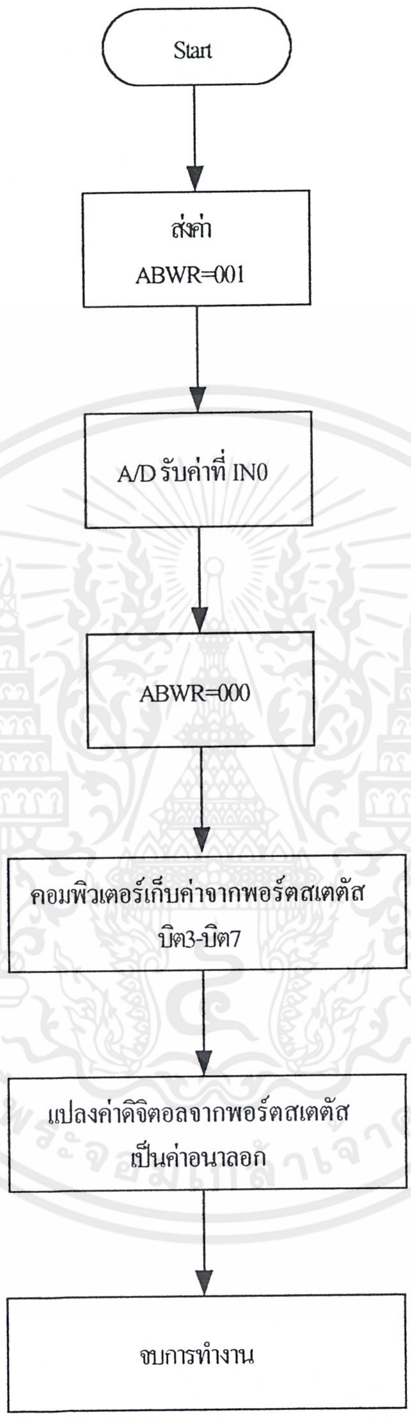


รูปที่ 3.19 โฟล์ชาร์ทแสดงโปรแกรมการจำลองการทำงานของมูลี่อัตโนมัติ (ต่อ)

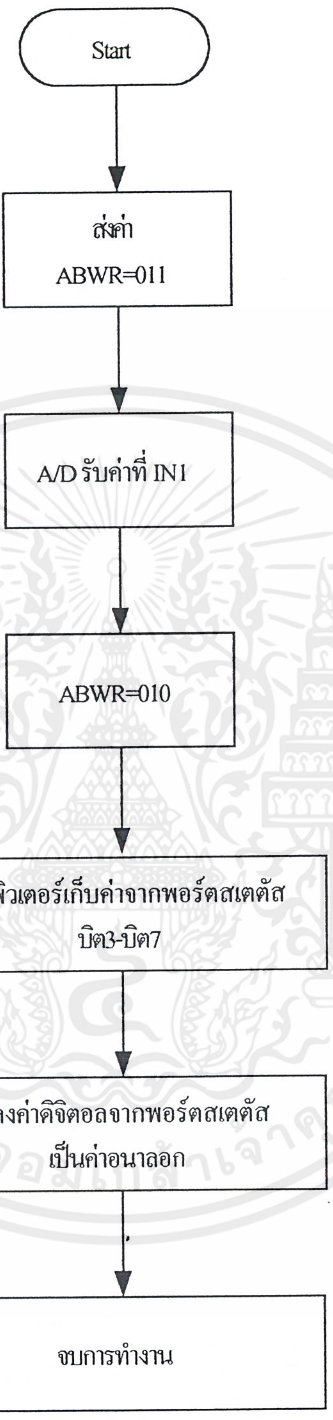


รูปที่ 3.19 โฟล์ชาร์ทแสดงโปรแกรมการจำลองการทำงานของมูลี่อัตโนมัติ (ต่อ)

การคำนวณค่าแรงดันไฟฟ้าที่ได้รับจากวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลทั้งจากการทดสอบมูลี่อัตโนมัติและการจำลองการทำงานมีขั้นตอนการทำงานที่เหมือนกันและสามารถแสดงได้ดังรูปที่ 3.20 และ 3.21



รูปที่ 3.20 การคำนวณแรงดันไฟฟ้าจากแอลดีอาร์



รูปที่ 3.21 การคำนวณแรงดันไฟฟ้าจากโพเทนชิโอห์มมิเตอร์

บทที่ 4

ผลการทดลอง

4.1 ผลการทดสองการส่งค่าของวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล

โดยการป้อนค่าทางแหล่งจ่ายไฟแล้ววัดสัญญาณที่ออกจากADC0809

แรงดันที่จ่ายเข้า	ผลการแปลงสัญญาณอนาลอกเป็นดิจิตอล								แรงดันทางทฤษฎี	ค่าความผิดพลาด
	บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0		
0.000	0	0	0	0	0	0	0	0	0.0000	0.00%
0.039	0	1	0	0	0	0	0	0	0.0390	0.00%
0.056	1	1	0	0	0	0	0	0	0.0586	4.42%
0.081	0	0	1	0	0	0	0	0	0.0781	3.68%
0.092	1	0	1	0	0	0	0	0	0.0977	5.79%
0.120	0	1	1	0	0	0	0	0	0.1172	2.41%
0.142	1	1	1	0	0	0	0	0	0.1367	3.88%
0.158	0	0	0	1	0	0	0	0	0.1563	1.12%
0.173	1	0	0	1	0	0	0	0	0.1758	1.58%
0.198	0	1	0	1	0	0	0	0	0.1953	1.38%
0.223	1	1	0	1	0	0	0	0	0.2148	3.82%
0.236	0	0	1	1	0	0	0	0	0.2343	0.73%
0.251	1	0	1	1	0	0	0	0	0.2539	1.14%
0.291	1	1	1	1	0	0	0	0	0.2929	0.65%
0.322	1	0	0	0	1	0	0	0	0.3320	3.01%
0.600	1	1	1	1	1	0	0	0	0.6054	0.89%
0.661	0	1	0	0	0	1	0	0	0.6640	0.45%
0.760	1	1	1	0	0	1	0	0	0.7617	0.22%
0.890	1	0	1	1	0	1	0	0	0.8789	1.26%

ตารางที่ 4.1 ผลการทดลองส่งค่าอนาลอกเป็นดิจิตอล

แรงดันที่จ่ายเข้า	ผลการแปลงสัญญาณอนาลอกเป็นดิจิตอล								แรงดันทางทฤษฎี	ค่าความผิดพลาด
	บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0		
1.180	1	0	1	1	1	1	0	0	1.1914	0.96%
1.245	1	0	0	0	0	0	1	0	1.2695	1.93%
1.350	0	1	1	0	0	0	1	0	1.3671	1.25%
1.470	1	0	1	1	0	0	1	0	1.5039	2.25%
1.580	0	1	0	0	1	0	1	0	1.6015	1.3%
1.780	1	0	1	1	1	0	1	0	1.8164	2.00%
1.850	1	0	0	0	0	1	1	0	1.8945	2.35%
1.930	0	0	1	0	0	1	1	0	1.9531	1.18%
2.030	1	0	0	1	0	1	1	0	2.0507	1.01%
2.539	0	0	1	0	0	0	0	1	2.5781	1.52%
2.806	0	1	0	0	1	0	0	1	2.8515	1.60%
2.937	0	0	0	1	1	0	0	1	2.9688	1.07%
2.969	1	1	0	1	1	0	0	1	3.0273	1.93%
3.020	1	0	1	1	1	0	0	1	3.0664	1.51%
3.120	1	1	0	0	0	1	0	1	3.1835	1.99%
3.330	1	0	1	1	0	1	0	1	3.3789	1.45%
3.398	1	0	0	0	1	1	0	1	3.4570	1.71%
3.400	1	0	0	0	1	1	0	1	3.4570	1.65%
3.496	0	1	1	0	1	1	0	1	3.5546	1.65%
3.630	1	0	1	1	1	1	0	1	3.6914	1.66%
3.847	1	0	0	1	0	0	1	1	3.9257	2.00%
3.850	0	0	0	1	0	0	1	1	3.9063	1.44%
4.020	1	0	0	0	1	0	1	1	4.0820	1.52%
4.030	0	1	0	0	1	0	1	1	4.1015	1.74%
4.080	0	0	1	0	1	0	1	1	4.1406	1.46%

ตารางที่ 4.1 ผลการทดลองส่งค่าอนาลอกเป็นดิจิตอล (ต่อ)

แรงดันที่จ่ายเข้า	ผลการแปลงสัญญาณอนาลอกเป็นดิจิตอล								แรงดันทางทฤษฎี	ค่าความผิดพลาด
	บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0		
4.350	0	1	0	0	0	1	1	1	4.4140	1.45%
4.470	0	0	0	1	0	1	1	1	4.5313	1.35%
4.550	1	0	1	1	0	1	1	1	4.6093	1.29%
4.710	1	0	1	0	1	1	1	1	4.7851	1.57%
4.770	1	0	0	1	1	1	1	1	4.8632	1.92%
4.930	1	1	1	1	1	1	1	1	4.9804	1.01%

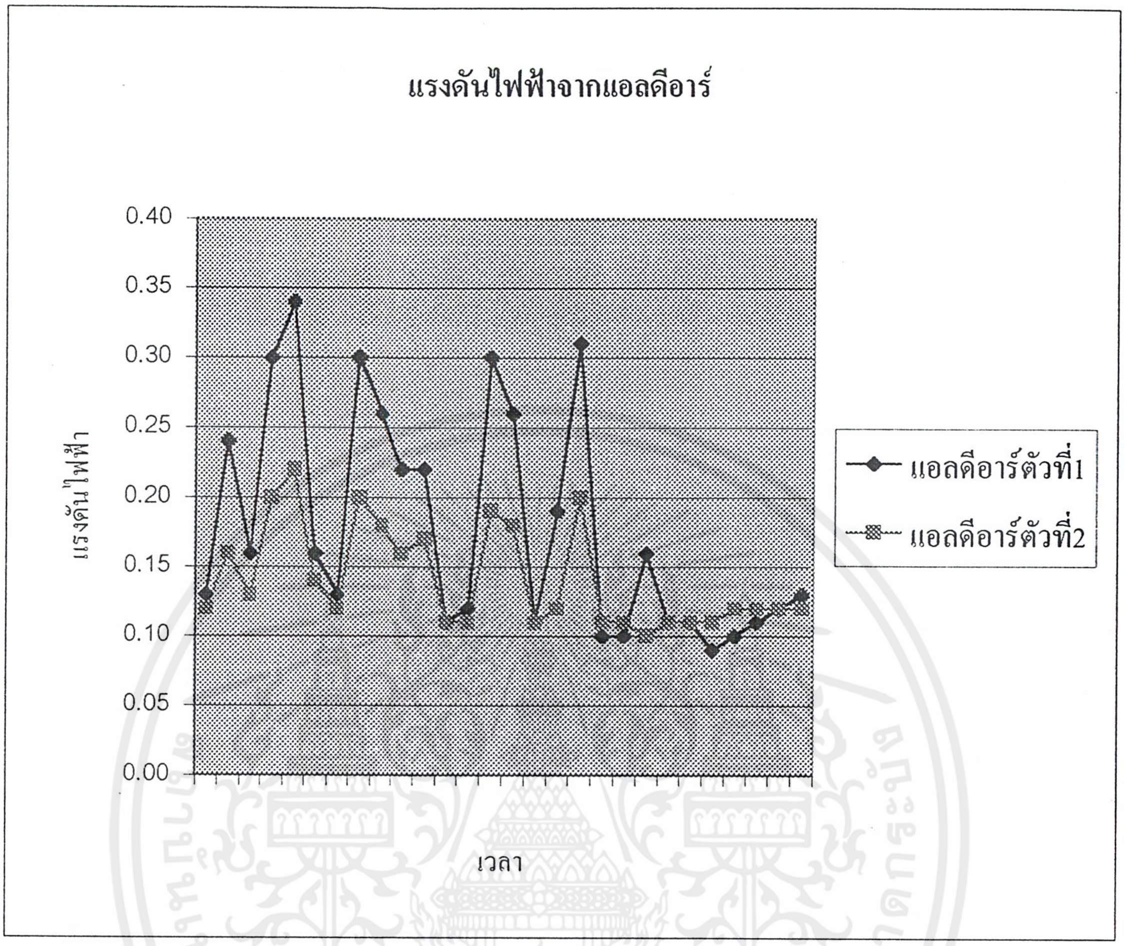
ตารางที่ 4.1 ผลการทดลองส่งค่าอนาลอกเป็นดิจิตอล (ต่อ)

4.2 ผลการทดลองการวัดค่าแรงดันไฟฟ้าจากแอลดีอาร์ โดยการวัดค่าของแอลดีอาร์จากวงจรบริดจ์

เวลา	แอลดีอาร์ตัวที่1	แอลดีอาร์ตัวที่2
10:20	0.13	0.12
10:30	0.24	0.16
10:40	0.16	0.13
10:50	0.30	0.2
11:00	0.34	0.22
11:10	0.16	0.14
11:20	0.13	0.12
11:30	0.30	0.2
11:40	0.26	0.18
11:50	0.22	0.16
12:00	0.22	0.17
12:10	0.11	0.11
12:20	0.12	0.11
12:30	0.30	0.19

เวลา	แอลดีอาร์ตัวที่1	แอลดีอาร์ตัวที่ 2
12:40	0.26	0.18
12:50	0.11	0.11
13:00	0.19	0.12
13:10	0.31	0.2
13:20	0.10	0.11
13:30	0.10	0.11
13:40	0.16	0.1
13:50	0.11	0.11
14:00	0.11	0.11
14:10	0.09	0.11
14:20	0.10	0.12
14:30	0.11	0.12
14:40	0.12	0.12
14:50	0.13	0.12

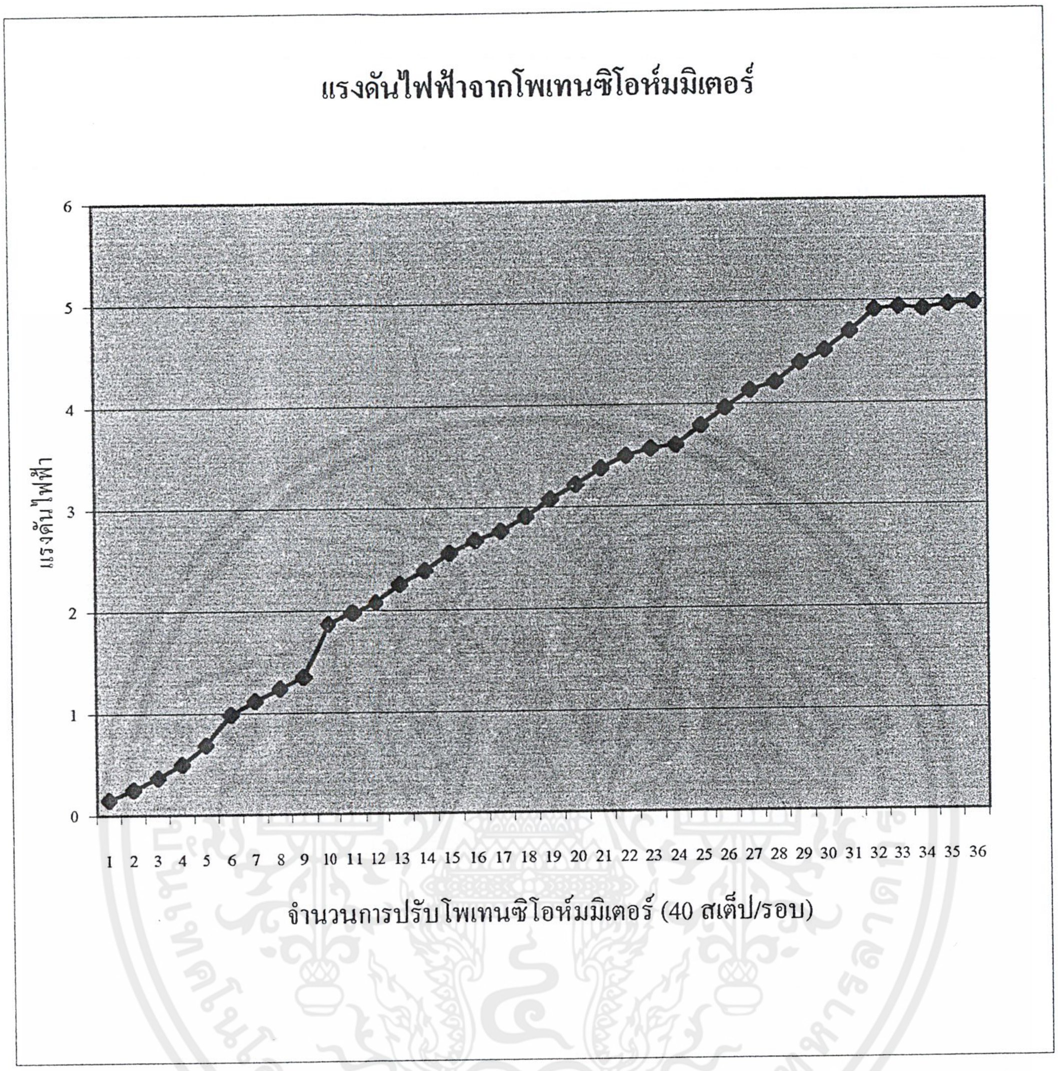
ตารางที่ 4.2 ผลการทดลองวัดค่าแรงดันไฟฟ้าจากแอลดีอาร์



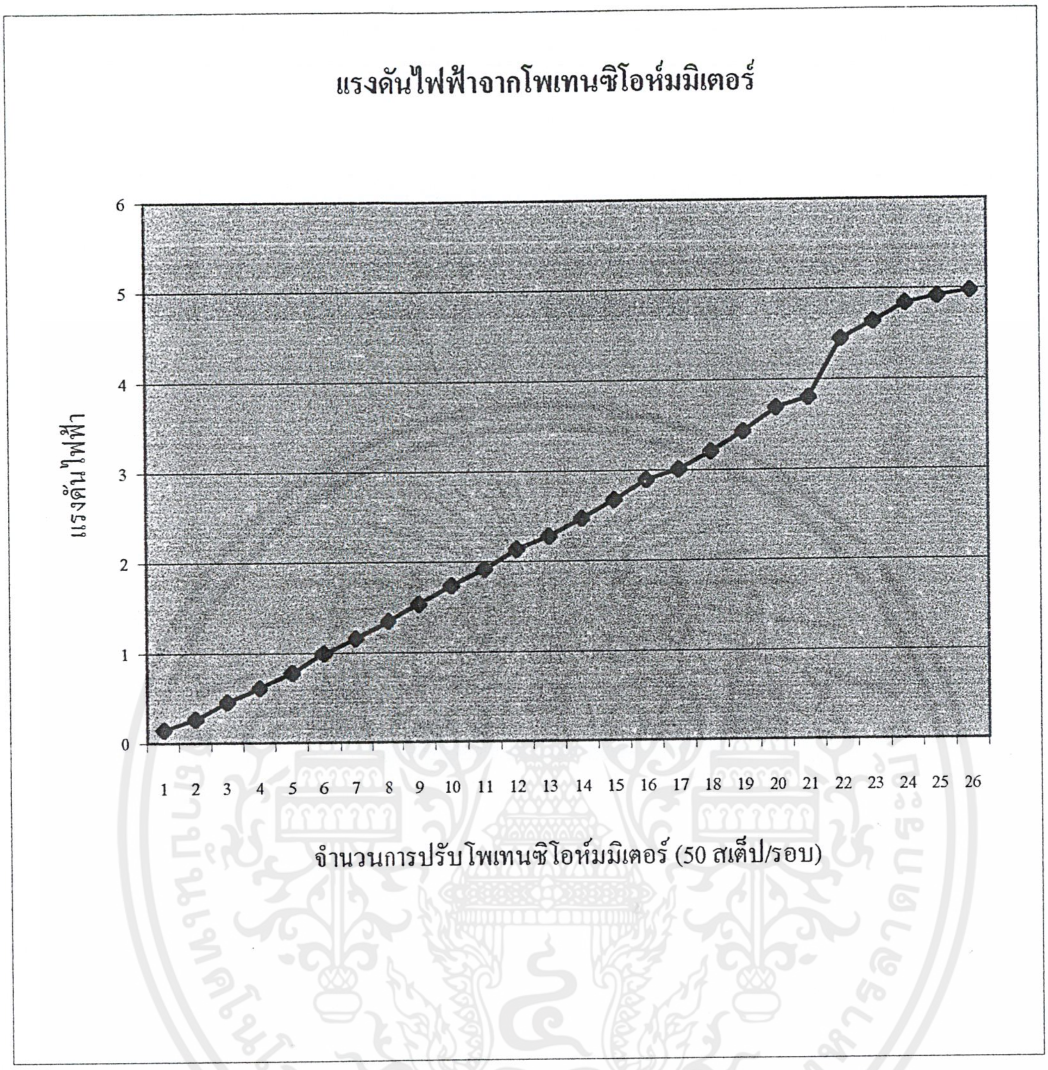
รูปที่ 4.1 ผลการทดลองวัดค่าแรงดันจากแอลดีอาร์

4.3 ผลการทดลองการวัดค่าแรงดันไฟฟ้าจากโพเทนซิโอห์มมิเตอร์

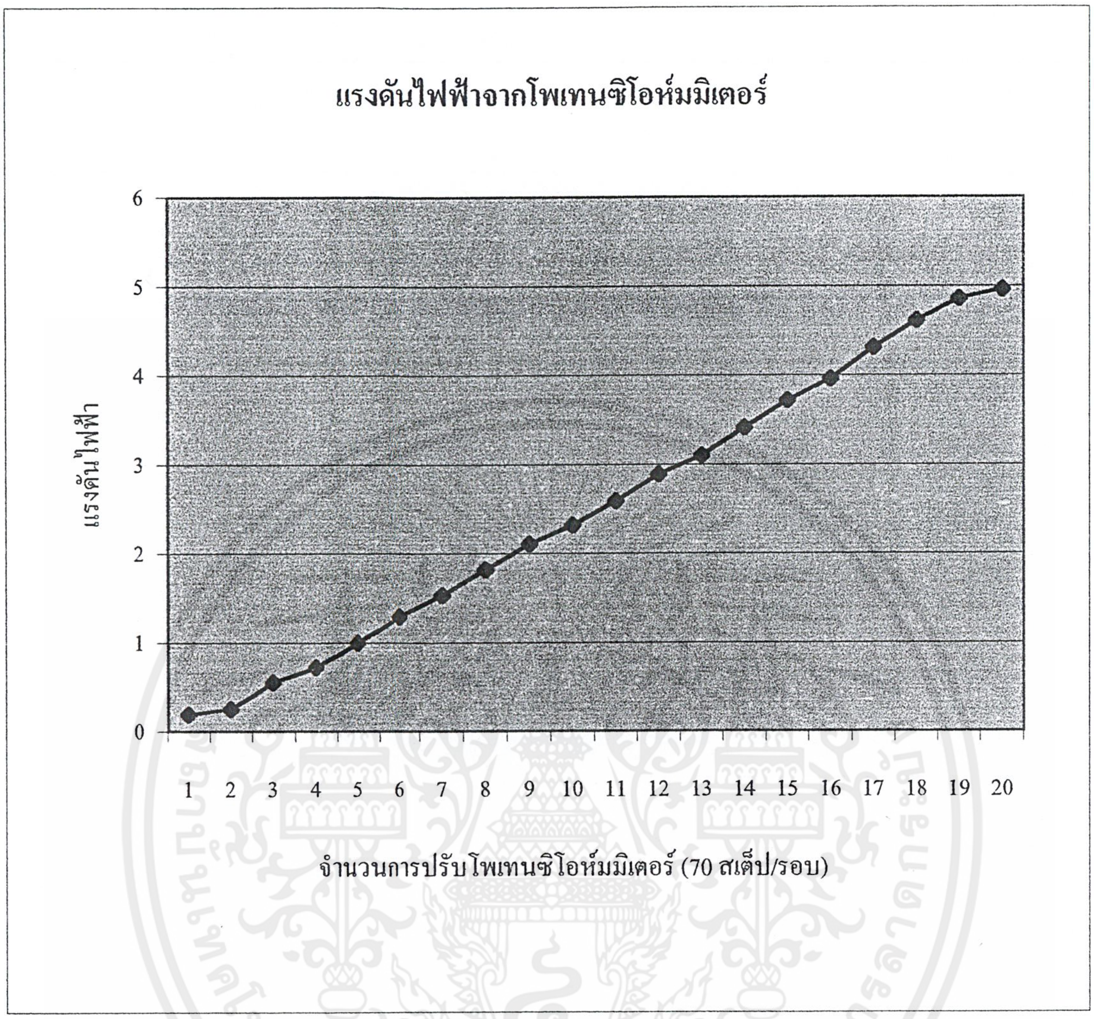
จากการทดลองพบว่าหากไม่ได้ทำการต่อโพเทนซิโอห์มมิเตอร์เข้ากับก้านปรับระดับขอมูลี่ มูลี่จะสามารถปรับระดับได้ประมาณ 1 องศาต่อการปรับสเต็ปปิ้งมอเตอร์ 1 สเต็ป แต่เมื่อทำการต่อโพเทนซิโอห์มมิเตอร์เข้าไปเพื่อทำเป็นตัวป้อนกลับให้กับระบบจะทำให้เกิดแรงเฉื่อยขึ้น แรงเฉื่อยนี้จะทำให้สเต็ปปิ้งมอเตอร์เกิดความผิดพลาดในปรับมูลี่ การทำงานจึงเป็นไปอย่างไม่ต่อเนื่อง และจากการทดลองจะพบว่าหาเลือกการปรับมูลี่ต่อครั้งในจำนวนที่แตกต่างกันจะให้ผลต่างกัน คือเมื่อกำหนดจำนวนการกระตุ้นสเต็ปปิ้งมอเตอร์ต่อการปรับมูลี่ 1 ครั้งด้วยสเต็ปการปรับที่มากขึ้นจะทำให้การทำงานเป็นไปอย่างต่อเนื่องมากขึ้นดังแสดงในรูปที่ 4.2 4.3 และ 4.4



รูปที่ 4.2 ผลการทดลองวัดค่าแรงดันจากโพเทนซิโอห์มมิเตอร์
ด้วยอัตราการปรับมอเตอร์ 40 ครั้ง/สเต็ป



รูปที่ 4.3 ผลการทดลองวัดค่าแรงดันจากโพเทนซิโอห์มมิเตอร์
ด้วยอัตราการปรับมอเตอร์ 50 ครั้ง/สเต็ป



รูปที่ 4.4 ผลการทดลองวัดค่าแรงดันจากโพเทนซิโอห์มมิเตอร์
ด้วยอัตราการปรับมอเตอร์ 70 ครั้ง/สเต็ป

ในส่วนของการจำลองการทำงานของมูลี่อัตโนมัติเลือกใช้การอัตราการปรับสเต็ปปิ้งมอเตอร์ 70 สเต็ป/ครั้ง สาเหตุที่เลือกค่านี้เนื่องจากค่าที่ทำให้เกิดความต่อเนื่องและสามารถปรับระดับแสงเข้าในแต่ละครั้งได้ในปริมาณที่ไม่มากจนเกินไป โดยมุมในการปรับมูลี่แต่ละครั้งประมาณ 9 องศา และเลือกปรับได้ 10 ระดับ (มุมในการปรับมูลี่ทั้งหมด 90 องศา)

บทที่ 5
บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง

โครงงานนี้มีจุดประสงค์หลักในการสร้างบอร์ดควบคุมสเต็ปปิ้งมอเตอร์ผ่านทางพอร์ตขนานของเครื่องคอมพิวเตอร์และแสดงการประยุกต์ใช้โดยสร้างมูลี่อัตโนมัติขึ้นมาเป็นแบบจำลอง ในส่วนของแบบจำลองการทำงานนี้ต้องทำการแปลงสัญญาณอนาลอกที่ได้จากแอลดีอาร์มาเป็นสัญญาณดิจิตอลก่อนจากนั้นจึงทำการป้อนข้อมูลให้คอมพิวเตอร์ประมวลผลการทำงานเพื่อควบคุมสเต็ปปิ้งมอเตอร์ให้สามารถปรับมูลี่ได้ตามต้องการและทำการป้อนกลับข้อมูลเพื่อตรวจสอบการทำงานของระบบโดยใช้โพเทนซิโอห์มมิเตอร์

จากการทดลองพบว่าสามารถนำบอร์ดควบคุมสเต็ปปิ้งมอเตอร์มาประยุกต์ใช้เป็นตัวควบคุมมูลี่อัตโนมัติได้ แต่ยังเกิดปัญหาในการทดลองบางส่วน หากทำการพัฒนาต่อไปจะสามารถนำไปใช้ในชีวิตประจำวันได้เพื่อเป็นส่วนช่วยในการประหยัดพลังงานไฟฟ้า

5.2 ปัญหาที่เกิดขึ้นในการทดลอง

1. คุณสมบัติของมอเตอร์

   การเลือกสเต็ปปิ้งมอเตอร์ให้เหมาะสมกับงานนั้นมีความสำคัญมากเพราะถ้าหากมอเตอร์มีแรงบิดสูงสุดต่ำกว่าโหลดของมอเตอร์จะทำให้เกิดปัญหา เช่น มอเตอร์สั่นอยู่กับที่ หยุดหมุน หมุนตามแนวแรงของโหลด เป็นต้น ความเร็วของสเต็ปปิ้งมอเตอร์จะขึ้นอยู่กับความถี่ของพัลซ์ที่ส่งออกจากเครื่องคอมพิวเตอร์ อย่างไรก็ตาม มอเตอร์ก็มีขีดจำกัดในด้านความเร็วซึ่งเป็นความเร็วที่เหมาะสม โดยจะมีอยู่แค่ช่วงเดียว ซึ่งถ้าหากความเร็วของมอเตอร์ต่ำเกินไปจะสั่น เนื่องจากแรงเฉื่อยของมอเตอร์และหากความเร็วของมอเตอร์เกินความสามารถในการรับโหลด มอเตอร์จะสั่นหยุดหมุนเช่นกันเนื่องจากการตอบสนองของมอเตอร์ช้ากว่าความถี่ของพัลซ์ที่ส่งออกไป

   คุณสมบัติของมอเตอร์มีความสำคัญดังกล่าวข้างต้น แต่ในการเลือกซื้อมอเต์อ์ตามที่ต้องการนั้นยาก สเต็ปปิ้งมอเตอร์ที่ใช้เป็นแบบที่ซื้อมาแล้วค่อยหาคุณสมบัติพื้นฐาน จากการทดสอบและตามบริษัทที่ผลิตตามอินเตอร์เนตซึ่งไม่สามารถรู้คุณสมบัติที่ครบถ้วน เนื่องจากบางมอเตอร์บางตัวก็เลิกผลิตไปแล้ว

2. ปัญหาจากวงจรสร้าวสัญญาณนาฬิกา สัญญาณที่ได้ยังไม่เรียบและมีสัญญาณรบกวนมาก ซึ่ง ADC0809 ไม่สามารถทำงานได้ หากสัญญาณนาฬิกามีสัญญาณรบกวนมากอาจทำให้

ADC0809 เสียได้ การแก้ไข คือ การนำตัวเก็บประจุมาต่อที่แหล่งจ่ายไฟ ทำให้สัญญาณที่ได้มีความเรียบมากขึ้นและนำตัวเก็บประจุมาต่อที่ไอซีแต่ละตัวเพื่อกันสัญญาณรบกวน เมื่อจ่ายไฟให้กับวงจรสร้างสัญญาณนาฬิกาแล้ว หากอยู่ใกล้เครื่องใช้ไฟฟ้าที่ส่งคลื่นได้ อาจทำให้เครื่องใช้ไฟฟ้านั้นทำงานได้ไม่ดี เพราะเนื่องจากสัญญาณนาฬิกาที่ใช้มีความถี่สูง 500 กิโลเฮิร์ต และถ้าวงจรนี้ไปอยู่ในทีที่มีสัญญาณรบกวนมากก็จะทำให้ผลของสัญญาณนาฬิกาออกมาไม่เป็นไปตามต้องการและส่งผลให้วงจรแปลงข้อมูลอนาลอกเป็นดิจิตอลใช้งานไม่ได้

3. การวัดค่าแรงดันจากแอลดีอาร์ยังมีค่าผิดพลาด ทั้งนี้เนื่องมาจากคุณสมบัติของมิเตอร์ที่ใช้ในการวัดและตัวผู้วัดเอง นอกจากนี้ค่าแรงดันที่วัดได้จากคอมพิวเตอร์ก็ยังมีควมผิดพลาดเพราะรับสัญญาณดิจิตอลเข้ามาแค่ 32 ระดับ (5 บิต) หากต้องการนำไปประยุกต์ใช้ในงานที่ละเอียดจะทำให้เกิดค่าความผิดพลาดในการทำงานได้มาก

ภาคผนวก

ภาคผนวก ก

คู่มือไอซี

FAIRCHILD
SEMICONDUCTOR™

August 1986
Revised March 2000

DM74LS165
8-Bit Parallel In/Serial Output Shift Registers

General Description

This device is an 8-bit serial shift register which shifts data in the direction of Q_A toward Q_H when clocked. Parallel-in access is made available by eight individual direct data inputs, which are enabled by a low level at the shift/load input. These registers also feature gated clock inputs and complementary outputs from the eighth bit.

Clocking is accomplished through a 2-input NOR gate, permitting one input to be used as a clock-inhibit function. Holding either of the clock inputs HIGH inhibits clocking, and holding either clock input LOW with the load input HIGH enables the other clock input. The clock-inhibit input should be changed to the high level only while the clock input is HIGH. Parallel loading is inhibited as long as the load input is HIGH. Data at the parallel inputs are loaded directly into the register on a HIGH-to-LOW transition of the shift/load input, regardless of the logic levels on the clock, clock inhibit, or serial inputs.

Features

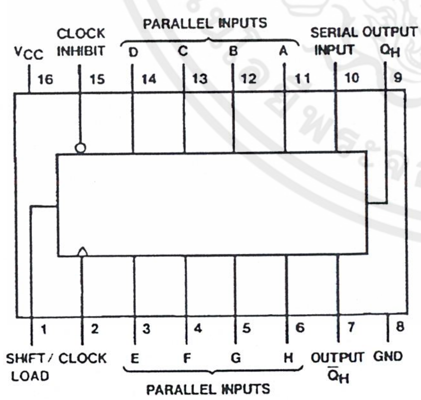
- Complementary outputs
- Direct overriding (data) inputs
- Gated clock inputs
- Parallel-to-serial data conversion
- Typical frequency 35 MHz
- Typical power dissipation 105 mW

Ordering Code:

Order Number	Package Number	Package Description
DM74LS165M	M16A	16-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-012, 0.150 Narrow
DM74LS165WM	M16B	16-Lead Small Outline Intergrated Circuit (SOIC), JEDEC MS-013, 0.300 Wide
DM74LS165N	N16E	16-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter "X" to the ordering code.

Connection Diagram

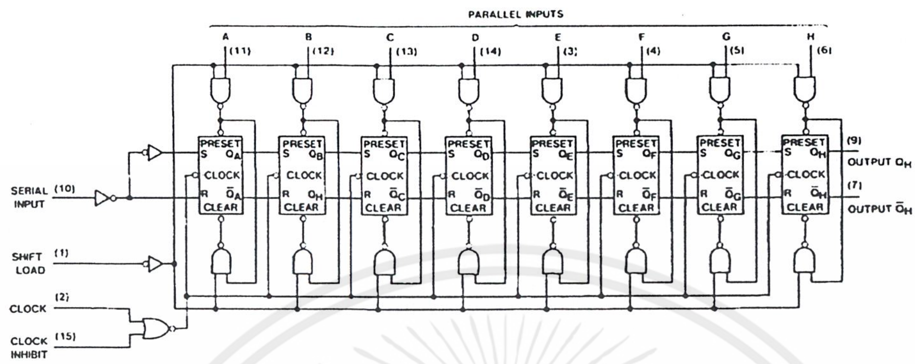


Function Table

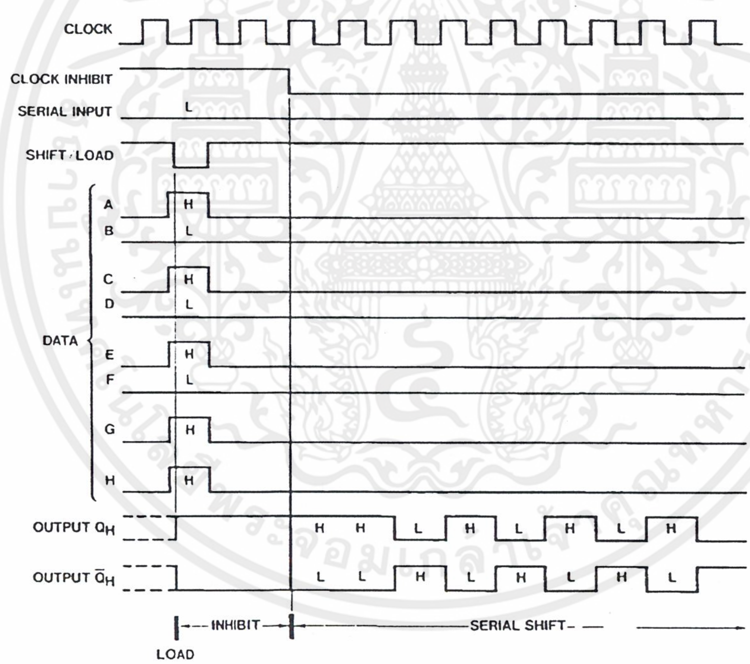
Inputs					Internal Outputs		Output
Shift/ Load	Clock Inhibit	Clock	Serial	Parallel A...H	Q_A	Q_B	Q_H
L	X	X	X	a...h	a	b	h
H	L	L	X	X	Q_{A0}	Q_{B0}	Q_{H0}
H	L	↑	H	X	H	Q_{An}	Q_{Gn}
H	L	↑	L	X	L	Q_{An}	Q_{Gn}
H	H	X	X	X	Q_{A0}	Q_{B0}	Q_{H0}

H = HIGH Level (steady state)
L = LOW Level (steady state)
X = Don't Care (any input, including transitions)
↑ = Transition from LOW-to-HIGH level
a...h = The level of steady-state input at inputs A through H, respectively.
Q_{A0}, Q_{B0}, Q_{H0} = The level of Q_A, Q_B, or Q_H, respectively, before the indicated steady-state input conditions were established.
Q_{An}, Q_{Gn} = The level of Q_A or Q_G, respectively, before the most recent ↑ transition of the clock.

Logic Diagram



Timing Diagram



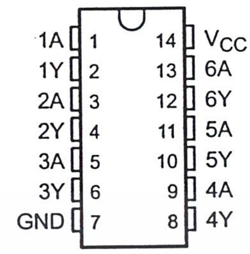
Typical Shift, Load, and Inhibit Sequences

SN5406, SN5416, SN7406, SN7416 HEX INVERTER BUFFERS/DRIVERS WITH OPEN-COLLECTOR HIGH-VOLTAGE OUTPUTS

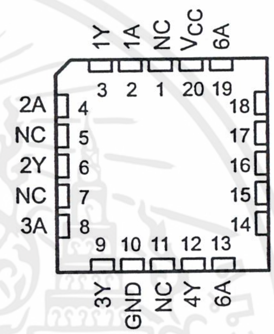
SDLS031A – DECEMBER 1983 – REVISED DECEMBER 2001

- **Convert TTL Voltage Levels to MOS Levels**
- **High Sink-Current Capability**
- **Input Clamping Diodes Simplify System Design**
- **Open-Collector Drivers for Indicator Lamps and Relays**
- **Inputs Fully Compatible With Most TTL Circuits**

SN5406, SN5416 . . . J OR W PACKAGE
SN7406 . . . D, N, OR NS PACKAGE
SN7416 . . . D OR N PACKAGE
(TOP VIEW)



SN5406 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

These TTL hex inverter buffers/drivers feature high-voltage open-collector outputs for interfacing with high-level circuits (such as MOS) or for driving high-current loads (such as lamps or relays), and also are characterized for use as inverter buffers for driving TTL inputs. The SN5406 and SN7406 have minimum breakdown voltages of 30 V. The SN5416 and SN7416 have minimum breakdown voltages of 15 V. The maximum sink current is 30 mA for the SN5406 and SN5416, and 40 mA for the SN7406 and SN7416.

ORDERING INFORMATION

T_A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	SOIC – D	Tube	SN7406D	7406
		Tape and reel	SN7406DR	
		Tube	SN7416D	7416
		Tape and reel	SN7416DR	
	PDIP – N	Tube	SN7406N	SN7406N
			SN7416N	SN7416N
	SOP – NS	Tape and reel	SN7406NSR	SN7406
–55°C to 125°C	CDIP – J	Tube	SNJ5406J	SNJ5406J
		Tube	SNJ5416J	SNJ5416J
	CDIP – W	Tube	SNJ5406W	SNJ5406W
		Tube	SNJ5416W	SNJ5416W
	LCCC – FK	Tube	SNJ5406FK	SNJ5406FK

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.

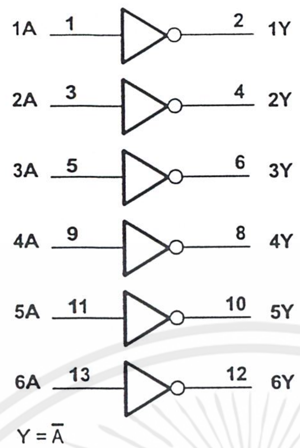
Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

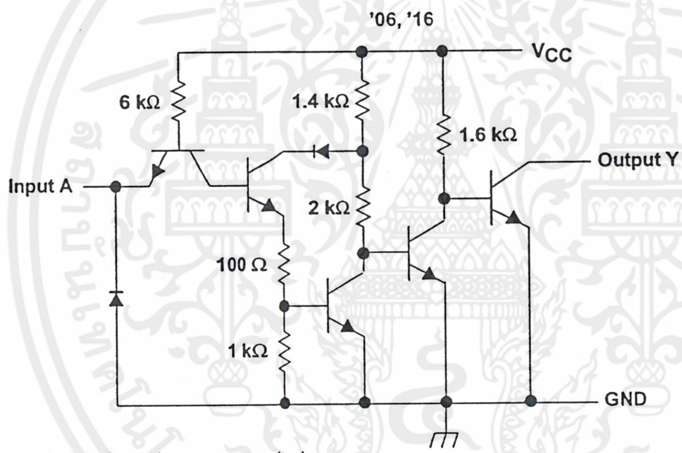
SN5406, SN5416, SN7406, SN7416
HEX INVERTER BUFFERS/DRIVERS
WITH OPEN-COLLECTOR HIGH-VOLTAGE OUTPUTS

SDLS031A – DECEMBER 1983 – REVISED DECEMBER 2001

logic diagram (positive logic)



schematic (each buffer/driver)



Resistor values shown are nominal.

absolute maximum ratings over operating free-air temperature (unless otherwise noted)†

Supply voltage, V_{CC} (see Note 1) .... 7 V
Input voltage, V_I (see Note 1) .... 5.5 V
Output voltage, V_O (see Notes 1 and 2): SN5406, SN7406 .... 30 V
SN5416, SN7416 .... 15 V
Package thermal impedance, θ_{JA} (see Note 3): D package .... 86°C/W
N package .... 80°C/W
NS package .... 76°C/W
Storage temperature range, T_{stg} .... –65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. Voltage values are with respect to network ground terminal.
2. This is the maximum voltage which should be applied to any output when it is in the off state.
3. The package thermal impedance is calculated in accordance with JESD 51-7.

Octal D-type flip-flop; positive edge-trigger; 3-state

74HC/HCT374

FEATURES

- 3-state non-inverting outputs for bus oriented applications
- 8-bit positive, edge-triggered register
- Common 3-state output enable input
- Independent register and 3-state buffer operation
- Output capability: bus driver
- I_{CC} category: MSI

GENERAL DESCRIPTION

The 74HC/HCT374 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). They are specified in compliance with JEDEC standard no. 7A.

The 74HC/HCT374 are octal D-type flip-flops featuring separate D-type inputs for each flip-flop and 3-state outputs for bus oriented applications. A clock (CP) and an output enable ($\overline{OE}$) input are common to all flip-flops.

The 8 flip-flops will store the state of their individual D-inputs that meet the set-up and hold times requirements on the LOW-to-HIGH CP transition.

When $\overline{OE}$ is LOW, the contents of the 8 flip-flops are available at the outputs. When $\overline{OE}$ is HIGH, the outputs go to the high impedance OFF-state. Operation of the $\overline{OE}$ input does not affect the state of the flip-flops.

The "374" is functionally identical to the "534", but has non-inverting outputs.

QUICK REFERENCE DATA

GND = 0 V; T_{amb} = 25 °C; $t_r = t_f$ = 6 ns

SYMBOL	PARAMETER	CONDITIONS	TYPICAL		UNIT
			HC	HCT	
t_{PHL}/t_{PLH}	propagation delay CP to Q_n	C_L = 15 pF; V_{CC} = 5 V	15	13	ns
f_{max}	maximum clock frequency		77	48	MHz
C_I	input capacitance		3.5	3.5	pF
C_{PD}	power dissipation capacitance per flip-flop	notes 1 and 2	17	17	pF

Notes

1. C_{PD} is used to determine the dynamic power dissipation (P_D in μW):
   $P_D = C_{PD} \times V_{CC}^2 \times f_i + \sum (C_L \times V_{CC}^2 \times f_o)$ where:
   f_i = input frequency in MHz
   f_o = output frequency in MHz
   $\sum (C_L \times V_{CC}^2 \times f_o)$ = sum of outputs
   C_L = output load capacitance in pF
   V_{CC} = supply voltage in V
2. For HC the condition is V_I = GND to V_{CC}
   For HCT the condition is V_I = GND to V_{CC} – 1.5 V

ORDERING INFORMATION

See *"74HC/HCT/HCU/HCMOS Logic Package Information"*.

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1	$\overline{OE}$	3-state output enable input (active LOW)
2, 5, 6, 9, 12, 15, 16, 19	Q_0 to Q_7	3-state flip-flop outputs
3, 4, 7, 8, 13, 14, 17, 18	D_0 to D_7	data inputs
10	GND	ground (0 V)
11	CP	clock input (LOW-to-HIGH, edge-triggered)
20	V_{CC}	positive supply voltage

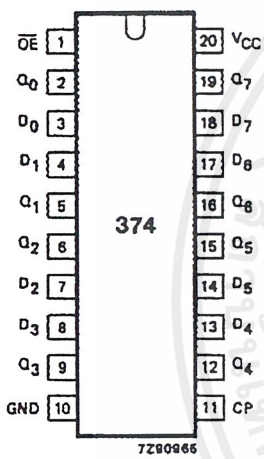


Fig.1 Pin configuration.

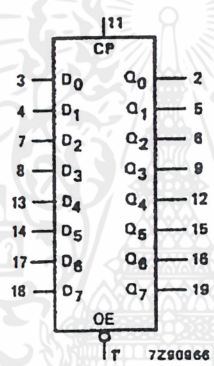


Fig.2 Logic symbol.

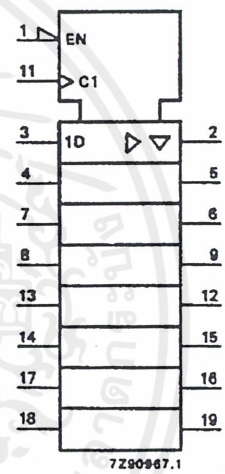


Fig.3 IEC logic symbol.

Octal D-type flip-flop; positive edge-trigger; 3-state

74HC/HCT374

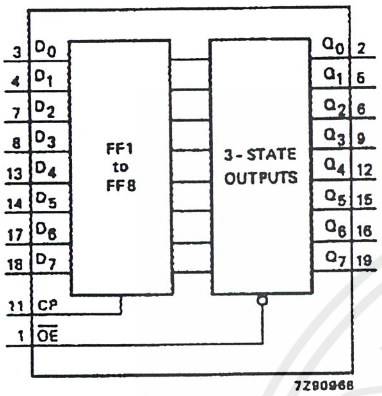


Fig.4 Functional diagram.

FUNCTION TABLE

OPERATING MODES	INPUTS			INTERNAL FLIP-FLOPS	OUTPUTS
	$\overline{OE}$	CP	D_n		Q_0 to Q_7
load and read register	L L	↑ ↑	l h	L H	L H
load register and disable outputs	H H	↑ ↑	l h	L H	Z Z

Notes

1. H = HIGH voltage level
   h = HIGH voltage level one set-up time prior to the LOW-to-HIGH CP transition
   L = LOW voltage level
   l = LOW voltage level one set-up time prior to the LOW-to-HIGH CP transition
   Z = high impedance OFF-state
   ↑ = LOW-to-HIGH CP transition

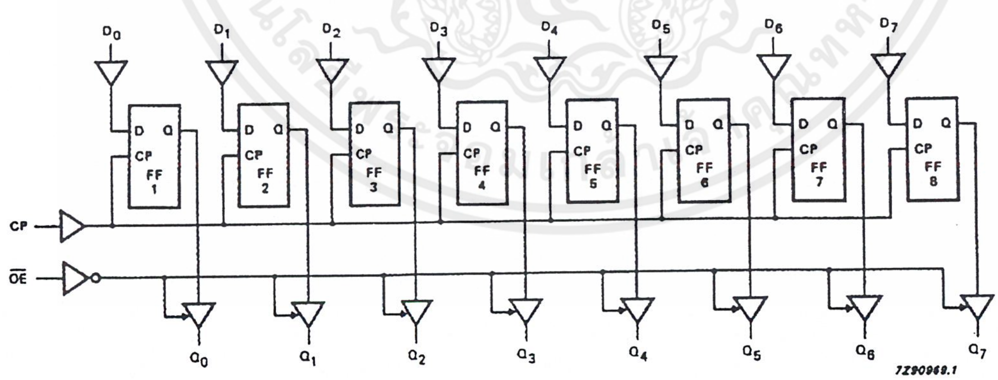


Fig.5 Logic diagram.

Dual 2-to-4 line decoder/demultiplexer 74HC/HCT139

FEATURES

- Demultiplexing capability
- Two independent 2-to-4 decoders
- Multifunction capability
- Active LOW mutually exclusive outputs
- Output capability: standard
- I_{CC} category: MSI

GENERAL DESCRIPTION

The 74HC/HCT139 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). It is specified in compliance with JEDEC standard no. 7A.

The 74HC/HCT139 are high-speed, dual 2-to-4 line decoder/multiplexers. This device has two independent decoders, each accepting two binary weighted inputs (nA_0 and nA_1) and providing four mutually exclusive active LOW outputs ($n\overline{Y}_0$ to $n\overline{Y}3$). Each decoder has an active LOW enable input ($n\overline{E}$).
When $n\overline{E}$ is HIGH, every output is forced HIGH. The enable can be used as the data input for a 1-to-4 demultiplexer application.
The "139" is identical to the HEF4556 of the HE4000B family.

QUICK REFERENCE DATA

GND = 0 V; T_{amb} = 25 °C; $t_r = t_f$ = 6 ns

SYMBOL	PARAMETER	CONDITIONS	TYPICAL		UNIT
			HC	HCT	
t_{PHL}/t_{PLH}	propagation delay	C_L = 15 pF; V_{CC} = 5 V			
	nA_n to $n\overline{Y}_n$		11	13	ns
	$n\overline{E}_3$ to $n\overline{Y}_n$		10	13	ns
C_I	input capacitance		3.5	3.5	pF
C_{PD}	power dissipation capacitance per multiplexer	notes 1 and 2	42	44	pF

Notes

1. C_{PD} is used to determine the dynamic power dissipation (P_D in μW):

   $P_D = C_{PD} \times V_{CC}^2 \times f_i + \sum (C_L \times V_{CC}^2 \times f_o)$ where:

   f_i = input frequency in MHz

   f_o = output frequency in MHz

   $\sum (C_L \times V_{CC}^2 \times f_o)$ = sum of outputs

   C_L = output load capacitance in pF

   V_{CC} = supply voltage in V
2. For HC the condition is V_I = GND to V_{CC}
   For HCT the condition is V_I = GND to V_{CC} – 1.5 V

APPLICATIONS

- Memory decoding or data-routing
- Code conversion

ORDERING INFORMATION

See *"74HC/HCT/HCU/HCMOS Logic Package Information"*.

Dual 2-to-4 line decoder/demultiplexer

74HC/HCT139

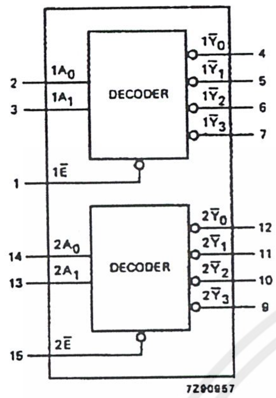


Fig.4 Functional diagram.

FUNCTION TABLE

INPUTS			OUTPUTS			
$n\overline{E}$	nA_0	nA_1	$n\overline{Y}_0$	$n\overline{Y}_1$	$n\overline{Y}_2$	$n\overline{Y}_3$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L

Notes

1. H = HIGH voltage level
   L = LOW voltage level
   X = don't care

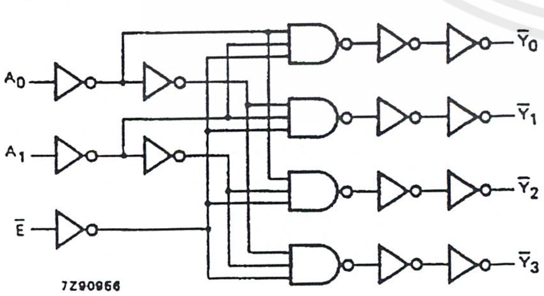


Fig.5 Logic diagram (one decoder/demultiplexer).

Dual 2-to-4 line decoder/demultiplexer — 74HC/HCT139

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1, 15	$1\overline{E}$, $2\overline{E}$	enable inputs (active LOW)
2, 3	$1A_0$, $1A_1$	address inputs
4, 5, 6, 7	$1\overline{Y}_0$ to $1\overline{Y}_3$	outputs (active LOW)
8	GND	ground (0 V)
12, 11, 10, 9	$2\overline{Y}_0$ to $2\overline{Y}_3$	outputs (active LOW)
14, 13	$2A_0$, $2A_1$	address inputs
16	V_{CC}	positive supply voltage

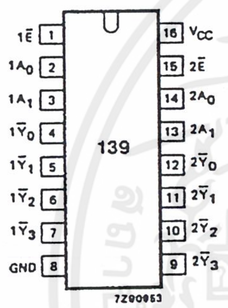


Fig.1 Pin configuration.

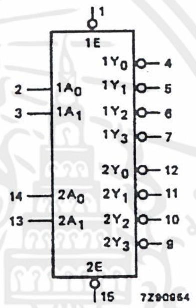


Fig.2 Logic symbol.

(a) (b)

Fig.3 IEC logic symbol.

Hex inverter — 74HC/HCT04

FEATURES

- Output capability: standard
- I_{CC} category: SSI

GENERAL DESCRIPTION

The 74HC/HCT04 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). They are specified in compliance with JEDEC standard no. 7A. The 74HC/HCT04 provide six inverting buffers.

QUICK REFERENCE DATA

GND = 0 V; T_{amb} = 25 °C; $t_r = t_f$ = 6 ns

SYMBOL	PARAMETER	CONDITIONS	TYPICAL		UNIT
			HC	HCT	
t_{PHL}/t_{PLH}	propagation delay nA to nY	C_L = 15 pF; V_{CC} = 5 V	7	8	ns
C_I	input capacitance		3.5	3.5	pF
C_{PD}	power dissipation capacitance per gate	notes 1 and 2	21	24	pF

Notes

1. C_{PD} is used to determine the dynamic power dissipation (P_D in µW):

   $P_D = C_{PD} \times V_{CC}^2 \times f_i + \Sigma (C_L \times V_{CC}^2 \times f_o)$ where:

   f_i = input frequency in MHz

   f_o = output frequency in MHz

   $\Sigma (C_L \times V_{CC}^2 \times f_o)$ = sum of outputs

   C_L = output load capacitance in pF

   V_{CC} = supply voltage in V

2. For HC the condition is V_I = GND to V_{CC}
   For HCT the condition is V_I = GND to $V_{CC} - 1.5$ V

ORDERING INFORMATION

See *"74HC/HCT/HCU/HCMOS Logic Package Information"*.

Hex inverter

74HC/HCT04

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1, 3, 5, 9, 11, 13	1A to 6A	data inputs
2, 4, 6, 8, 10, 12	1Y to 6Y	data outputs
7	GND	ground (0 V)
14	V_{CC}	positive supply voltage

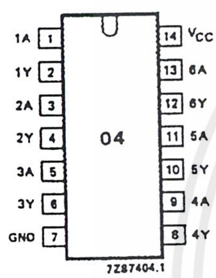


Fig.1 Pin configuration.

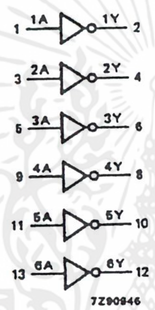


Fig.2 Logic symbol.

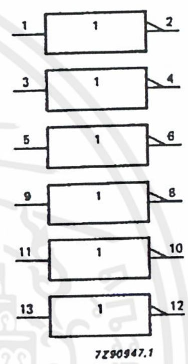


Fig.3 IEC logic symbol.

FUNCTION TABLE

INPUT	OUTPUT
nA	nY
L	H
H	L

Notes

1. H = HIGH voltage level
   L = LOW voltage level

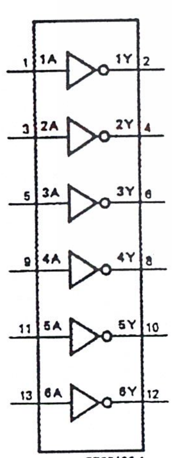


Fig.4 Functional diagram.

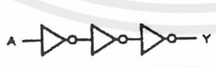


Fig.5 Logic diagram (one inverter).

National Semiconductor

May 2000

LM78XX
Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expanded to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

Features

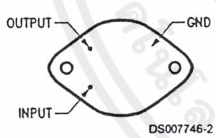
- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

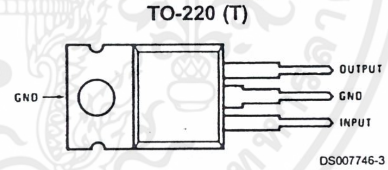
Connection Diagrams

Metal Can Package
TO-3 (K)
Aluminum



Bottom View
Order Number LM7805CK,
LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)



Top View
Order Number LM7805CT,
LM7812CT or LM7815CT
See NS Package Number T03B

Absolute Maximum Ratings (Note 3)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Input Voltage	
(V_O = 5V, 12V and 15V)	35V
Internal Power Dissipation (Note 1)	Internally Limited
Operating Temperature Range (T_A)	0°C to +70°C
Maximum Junction Temperature	
(K Package)	150°C
(T Package)	150°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature (Soldering, 10 sec.)	
TO-3 Package K	300°C
TO-220 Package T	230°C

Electrical Characteristics LM78XXC (Note 2)

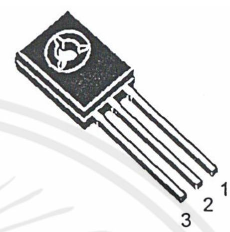
0°C ≤ T_J ≤ 125°C unless otherwise noted.

Output Voltage				5V			12V			15V			Units
Input Voltage (unless otherwise noted)				10V			19V			23V			
Symbol	Parameter	Conditions		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
V_O	Output Voltage	Tj = 25°C, 5 mA ≤ I_O ≤ 1A		4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V
		P_D ≤ 15W, 5 mA ≤ I_O ≤ 1A		4.75		5.25	11.4		12.6	14.25		15.75	V
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7.5 ≤ V_{IN} ≤ 20)			(14.5 ≤ V_{IN} ≤ 27)			(17.5 ≤ V_{IN} ≤ 30)			V
ΔV_O	Line Regulation	I_O = 500 mA	Tj = 25°C		3	50		4	120		4	150	mV
			ΔV_{IN}	(7 ≤ V_{IN} ≤ 25)			14.5 ≤ V_{IN} ≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
			0°C ≤ Tj ≤ +125°C			50			120			150	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 20)			(15 ≤ V_{IN} ≤ 27)			(18.5 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 1A	Tj = 25°C			50			120			150	mV
			ΔV_{IN}	(7.5 ≤ V_{IN} ≤ 20)			(14.6 ≤ V_{IN} ≤ 27)			(17.7 ≤ V_{IN} ≤ 30)			V
			0°C ≤ Tj ≤ +125°C			25			60			75	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 12)			(16 ≤ V_{IN} ≤ 22)			(20 ≤ V_{IN} ≤ 26)			V
ΔV_O	Load Regulation	Tj = 25°C	5 mA ≤ I_O ≤ 1.5A		10	50		12	120		12	150	mV
			250 mA ≤ I_O ≤ 750 mA			25			60			75	mV
		5 mA ≤ I_O ≤ 1A, 0°C ≤ Tj ≤ +125°C				50			120			150	mV
I_Q	Quiescent Current	I_O ≤ 1A	Tj = 25°C			8			8			8	mA
			0°C ≤ Tj ≤ +125°C			8.5			8.5			8.5	mA
ΔI_Q	Quiescent Current Change	5 mA ≤ I_O ≤ 1A				0.5			0.5			0.5	mA
		Tj = 25°C, I_O ≤ 1A				1.0			1.0			1.0	mA
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7.5 ≤ V_{IN} ≤ 20)			(14.8 ≤ V_{IN}≤ 27)			(17.9 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 500 mA, 0°C ≤ Tj ≤ +125°C				1.0			1.0			1.0	mA
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7 ≤ V_{IN} ≤ 25)			(14.5 ≤ V_{IN}≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
V_N	Output Noise Voltage	T_A =25°C, 10 Hz ≤ f ≤ 100 kHz			40			75			90		μV
$\frac{\Delta V_{IN}}{\Delta V_{OUT}}$	Ripple Rejection	f = 120 Hz	I_O ≤ 1A, Tj = 25°C or	62	80		55	72		54	70		dB
			I_O ≤ 500 mA 0°C ≤ Tj ≤ +125°C	62			55			54			dB
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(8 ≤ V_{IN} ≤ 18)			(15 ≤ V_{IN} ≤ 25)			(18.5 ≤ V_{IN} ≤ 28.5)			V
R_O	Dropout Voltage	Tj = 25°C, I_{OUT} = 1A			2.0			2.0			2.0		V
	Output Resistance	f = 1 kHz			8			18			19		mΩ

BD677/A/679/A/681
BD678/A/680/A/682

COMPLEMENTARY SILICON POWER DARLINGTON TRANSISTORS

- STMicroelectronics PREFERRED SALESTYPES
- COMPLEMENTARY PNP - NPN DEVICES
- MONOLITHIC DARLINGTON CONFIGURATION
- INTEGRATED ANTIPARALLEL COLLECTOR-EMITTER DIODE



SOT-32

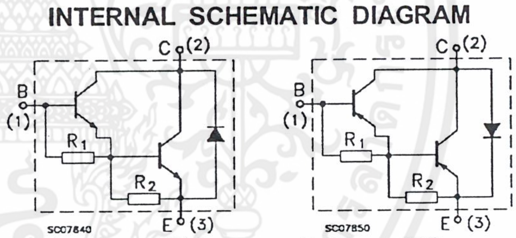
APPLICATION
- LINEAR AND SWITCHING INDUSTRIAL EQUIPMENT

DESCRIPTION
The BD677, BD677A, BD679, BD679A and BD681 are silicon epitaxial-base NPN power transistors in monolithic Darlington configuration mounted in Jedec SOT-32 plastic package.

They are intended for use in medium power linar and switching applications

The complementary PNP types are BD678, BD678A, BD680, BD680A and BD682 respectively.

INTERNAL SCHEMATIC DIAGRAM



R_1 Typ.= 7K Ω R_2 Typ.= 230

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter		Value			Unit
		NPN	BD677/A	BD679/A	BD681	
		PNP	BD678/A	BD680/A	BD682	
V_{CBO}	Collector-Base Voltage ($I_E = 0$)		60	80	100	V
V_{CEO}	Collector-Emitter Voltage ($I_B = 0$)		60	80	100	V
V_{EBO}	Emitter-Base Voltage ($I_C = 0$)			5		V
I_C	Collector Current			4		A
I_{CM}	Collector Peak Current			6		A
I_B	Base Current			0.1		A
P_{tot}	Total Dissipation at $T_c \leq 25$ °C			40		W
T_{stg}	Storage Temperature			-65 to 150		°C
T_j	Max. Operating Junction Temperature			150		°C

For PNP types voltage and current values are negative.

THERMAL DATA

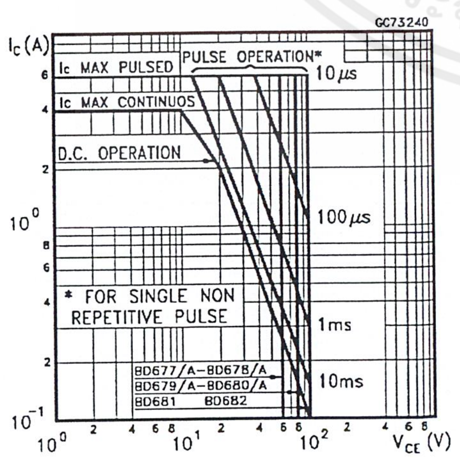
$R_{thj-case}$	Thermal Resistance Junction-case	Max	3.12	°C/W
$R_{thj-amb}$	Thermal Resistance Junction-ambient	Max	100	°C/W

ELECTRICAL CHARACTERISTICS (T_{case} = 25 °C unless otherwise specified)

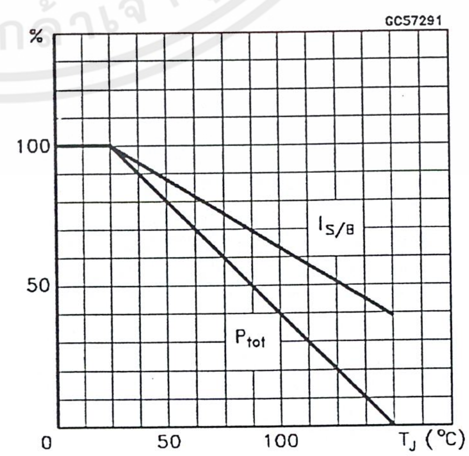
Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I_{CBO}	Collector Cut-off Current (I_E = 0)	V_{CE} = rated V_{CBO} V_{CE} = rated V_{CBO} T_C = 100 °C			0.2 2	mA mA
I_{CEO}	Collector Cut-off Current (I_B = 0)	V_{CE} = half rated V_{CEO}			0.5	mA
I_{EBO}	Emitter Cut-off Current (I_C = 0)	V_{EB} = 5 V			2	mA
$V_{CEO(sus)}$*	Collector-Emitter Sustaining Voltage	I_C = 50 mA **for BD677/677A/678/678A** **for BD679/679A/680/680A** **for BD681/682**	 60 80 100			 V V V
$V_{CE(sat)}$*	Collector-Emitter Saturation Voltage	**for BD677/678/679/680/681/682** I_C = 1.5 A I_B = 30 mA **for BD677A/678A/679A/680A** I_C = 2 A I_B = 40 mA			2.5 2.8	V V
V_{BE}*	Base-Emitter Voltage	**for BD677/678/679/680/681/682** I_C = 1.5 A V_{CE} = 3 V **for BD677A/678A/679A/680A** I_C = 2 A V_{CE} = 3 V			2.5 2.5	V V
h_{FE}*	DC Current Gain	**for BD677/678/679/680/681/682** I_C = 1.5 A V_{CE} = 3 V **for BD677A/678A/679A/680A** I_C = 2 A V_{CE} = 3 V	750 750			
h_{fe}	Small Signal Current Gain	I_C = 1.5 A V_{CE} = 3 V f = 1MHz	1			

* Pulsed: Pulse duration = 300 ms, duty cycle 1.5 %

Safe Operating Areas



Derating Curve



October 1999

National Semiconductor

ADC0808/ADC0809
8-Bit µP Compatible A/D Converters with 8-Channel Multiplexer

General Description

The ADC0808, ADC0809 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 8-channel multiplexer and microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 8-channel multiplexer can directly access any of 8-single-ended analog signals.

The device eliminates the need for external zero and full-scale adjustments. Easy interfacing to microprocessors is provided by the latched and decoded multiplexer address inputs and latched TTL TRI-STATE outputs.

The design of the ADC0808, ADC0809 has been optimized by incorporating the most desirable aspects of several A/D conversion techniques. The ADC0808, ADC0809 offers high speed, high accuracy, minimal temperature dependence, excellent long-term accuracy and repeatability, and consumes minimal power. These features make this device ideally suited to applications from process and machine control to consumer and automotive applications. For 16-channel multiplexer with common output (sample/hold port) see ADC0816 data sheet. (See AN-247 for more information.)

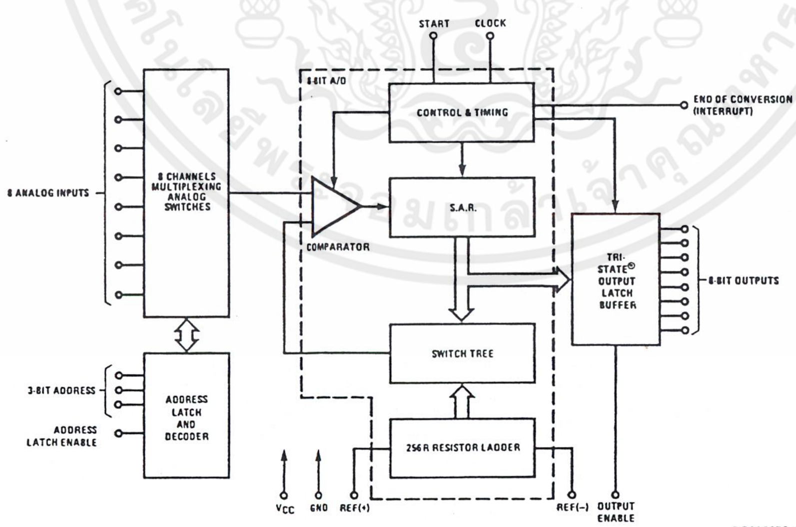
Features

- Easy interface to all microprocessors
- Operates ratiometrically or with 5 V_{DC} or analog span adjusted voltage reference
- No zero or full-scale adjust required
- 8-channel multiplexer with address logic
- 0V to 5V input range with single 5V power supply
- Outputs meet TTL voltage level specifications
- Standard hermetic or molded 28-pin DIP package
- 28-pin molded chip carrier package
- ADC0808 equivalent to MM74C949
- ADC0809 equivalent to MM74C949-1

Key Specifications

Resolution	8 Bits
Total Unadjusted Error	±½ LSB and ±1 LSB
Single Supply	5 V_{DC}
Low Power	15 mW
Conversion Time	100 µs

Block Diagram

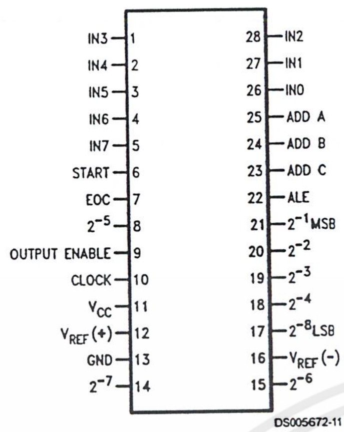


DS005672-1

See Ordering Information

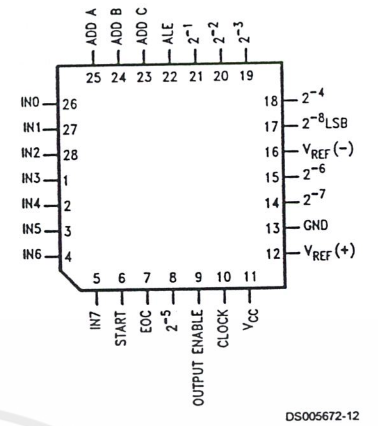
Connection Diagrams

Dual-In-Line Package



Order Number ADC0808CCN or ADC0809CCN
See NS Package J28A or N28A

Molded Chip Carrier Package



Order Number ADC0808CCV or ADC0809CCV
See NS Package V28A

Ordering Information

TEMPERATURE RANGE		−40˚C to +85˚C			−55˚C to +125˚C
Error	±½ LSB Unadjusted	ADC0808CCN	ADC0808CCV	ADC0808CCJ	ADC0808CJ
	±1 LSB Unadjusted	ADC0809CCN	ADC0809CCV		
Package Outline		N28A Molded DIP	V28A Molded Chip Carrier	J28A Ceramic DIP	J28A Ceramic DIP

Absolute Maximum Ratings (Notes 2, 1)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage (V_{CC}) (Note 3)	6.5V
Voltage at Any Pin	−0.3V to (V_{CC}+0.3V)
Except Control Inputs	
Voltage at Control Inputs	−0.3V to +15V
(START, OE, CLOCK, ALE, ADD A, ADD B, ADD C)	
Storage Temperature Range	−65˚C to +150˚C
Package Dissipation at T_A=25˚C	875 mW
Lead Temp. (Soldering, 10 seconds)	
Dual-In-Line Package (plastic)	260˚C
Dual-In-Line Package (ceramic)	300˚C
Molded Chip Carrier Package	
Vapor Phase (60 seconds)	215˚C
Infrared (15 seconds)	220˚C
ESD Susceptibility (Note 8)	400V

Operating Conditions (Notes 1, 2)

Temperature Range (Note 1)	$T_{MIN} \le T_A \le T_{MAX}$
ADC0808CCN,ADC0809CCN	−40˚C≤T_A≤+85˚C
ADC0808CCV, ADC0809CCV	−40˚C ≤ T_A ≤ +85˚C
Range of V_{CC} (Note 1)	4.5 V_{DC} to 6.0 V_{DC}

Electrical Characteristics

Converter Specifications: V_{CC}=5 V_{DC}=V_{REF+}, $V_{REF(-)}$=GND, $T_{MIN} \le T_A \le T_{MAX}$ and f_{CLK}=640 kHz unless otherwise stated.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	ADC0808					
	Total Unadjusted Error	25˚C			±½	LSB
	(Note 5)	T_{MIN} to T_{MAX}			±¾	LSB
	ADC0809					
	Total Unadjusted Error	0˚C to 70˚C			±1	LSB
	(Note 5)	T_{MIN} to T_{MAX}			±1¼	LSB
	Input Resistance	From Ref(+) to Ref(−)	1.0	2.5		kΩ
	Analog Input Voltage Range	(Note 4) V(+) or V(−)	GND−0.10		V_{CC}+0.10	V_{DC}
$V_{REF(+)}$	Voltage, Top of Ladder	Measured at Ref(+)		V_{CC}	V_{CC}+0.1	V
$\frac{V_{REF(+)}+V_{REF(-)}}{2}$	Voltage, Center of Ladder		V_{CC}/2-0.1	V_{CC}/2	V_{CC}/2+0.1	V
$V_{REF(-)}$	Voltage, Bottom of Ladder	Measured at Ref(−)	−0.1	0		V
I_{IN}	Comparator Input Current	f_c=640 kHz, (Note 6)	−2	±0.5	2	μA

Electrical Characteristics

Digital Levels and DC Specifications: ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV, 4.75≤V_{CC}≤5.25V, −40˚C≤T_A≤+85˚C unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Units
ANALOG MULTIPLEXER						
$I_{OFF(+)}$	OFF Channel Leakage Current	V_{CC}=5V, V_{IN}=5V,				
		T_A=25˚C		10	200	nA
		T_{MIN} to T_{MAX}			1.0	μA
$I_{OFF(-)}$	OFF Channel Leakage Current	V_{CC}=5V, V_{IN}=0,				
		T_A=25˚C	−200	−10		nA
		T_{MIN} to T_{MAX}	−1.0			μA
CONTROL INPUTS						
$V_{IN(1)}$	Logical "1" Input Voltage		V_{CC}−1.5			V
$V_{IN(0)}$	Logical "0" Input Voltage				1.5	V
$I_{IN(1)}$	Logical "1" Input Current (The Control Inputs)	V_{IN}=15V			1.0	μA
$I_{IN(0)}$	Logical "0" Input Current (The Control Inputs)	V_{IN}=0	−1.0			μA
I_{CC}	Supply Current	f_{CLK}=640 kHz		0.3	3.0	mA

Electrical Characteristics (Continued)

Digital Levels and DC Specifications: ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV, $4.75 \le V_{CC} \le 5.25V$, $-40^{\circ}C \le T_A \le +85^{\circ}C$ unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Units
DATA OUTPUTS AND EOC (INTERRUPT)						
$V_{OUT(1)}$	Logical "1" Output Voltage	V_{CC} = 4.75V I_{OUT} = −360µA I_{OUT} = −10µA		 2.4 4.5		 V(min) V(min)
$V_{OUT(0)}$	Logical "0" Output Voltage	I_O=1.6 mA			0.45	V
$V_{OUT(0)}$	Logical "0" Output Voltage EOC	I_O=1.2 mA			0.45	V
I_{OUT}	TRI-STATE Output Current	V_O=5V V_O=0	 −3		3	µA µA

Electrical Characteristics

Timing Specifications $V_{CC}=V_{REF(+)}=5V$, $V_{REF(-)}$=GND, $t_r=t_f$=20 ns and $T_A=25^{\circ}C$ unless otherwise noted.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
t_{WS}	Minimum Start Pulse Width	(*Figure 5*)		100	200	ns
t_{WALE}	Minimum ALE Pulse Width	(*Figure 5*)		100	200	ns
t_s	Minimum Address Set-Up Time	(*Figure 5*)		25	50	ns
t_H	Minimum Address Hold Time	(*Figure 5*)		25	50	ns
t_D	Analog MUX Delay Time From ALE	R_S=0Ω (*Figure 5*)		1	2.5	µs
t_{H1}, t_{H0}	OE Control to Q Logic State	C_L=50 pF, R_L=10k (*Figure 8*)		125	250	ns
t_{1H}, t_{0H}	OE Control to Hi-Z	C_L=10 pF, R_L=10k (*Figure 8*)		125	250	ns
t_c	Conversion Time	f_c=640 kHz, (*Figure 5*) (Note 7)	90	100	116	µs
f_c	Clock Frequency		10	640	1280	kHz
t_{EOC}	EOC Delay Time	(*Figure 5*)	0		8+2 µS	Clock Periods
C_{IN}	Input Capacitance	At Control Inputs		10	15	pF
C_{OUT}	TRI-STATE Output Capacitance	At TRI-STATE Outputs		10	15	pF

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to GND, unless othewise specified.

Note 3: A zener diode exists, internally, from V_{CC} to GND and has a typical breakdown voltage of 7 V_{DC}.

Note 4: Two on-chip diodes are tied to each analog input which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the V_{CC}n supply. The spec allows 100 mV forward bias of either diode. This means that as long as the analog V_{IN} does not exceed the supply voltage by more than 100 mV, the output code will be correct. To achieve an absolute $0V_{DC}$ to $5V_{DC}$ input voltage range will therefore require a minimum supply voltage of 4.900 V_{DC} over temperature variations, initial tolerance and loading.

Note 5: Total unadjusted error includes offset, full-scale, linearity, and multiplexer errors. See *Figure 3*. None of these A/Ds requires a zero or full-scale adjust. However, if an all zero code is desired for an analog input other than 0.0V, or if a narrow full-scale span exists (for example: 0.5V to 4.5V full-scale) the reference voltages can be adjusted to achieve this. See *Figure 13*.

Note 6: Comparator input current is a bias current into or out of the chopper stabilized comparator. The bias current varies directly with clock frequency and has little temperature dependence (*Figure 6*). See paragraph 4.0.

Note 7: The outputs of the data register are updated one clock cycle before the rising edge of EOC.

Note 8: Human body model, 100 pF discharged through a 1.5 kΩ resistor.

Functional Description

Multiplexer. The device contains an 8-channel single-ended analog signal multiplexer. A particular input channel is selected by using the address decoder. *Table 1* shows the input states for the address lines to select any channel. The address is latched into the decoder on the low-to-high transition of the address latch enable signal.

TABLE 1.

SELECTED ANALOG CHANNEL	ADDRESS LINE		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

CONVERTER CHARACTERISTICS

The Converter

The heart of this single chip data acquisition system is its 8-bit analog-to-digital converter. The converter is designed to give fast, accurate, and repeatable conversions over a wide range of temperatures. The converter is partitioned into 3 major sections: the 256R ladder network, the successive approximation register, and the comparator. The converter's digital outputs are positive true.

The 256R ladder network approach (*Figure 1*) was chosen over the conventional R/2R ladder because of its inherent monotonicity, which guarantees no missing digital codes. Monotonicity is particularly important in closed loop feedback control systems. A non-monotonic relationship can cause oscillations that will be catastrophic for the system. Additionally, the 256R network does not cause load variations on the reference voltage.

The bottom resistor and the top resistor of the ladder network in *Figure 1* are not the same value as the remainder of the network. The difference in these resistors causes the output characteristic to be symmetrical with the zero and full-scale points of the transfer curve. The first output transition occurs when the analog signal has reached +½ LSB and succeeding output transitions occur every 1 LSB later up to full-scale.

The successive approximation register (SAR) performs 8 iterations to approximate the input voltage. For any SAR type converter, n-iterations are required for an n-bit converter. *Figure 2* shows a typical example of a 3-bit converter. In the ADC0808, ADC0809, the approximation technique is extended to 8 bits using the 256R network.

The A/D converter's successive approximation register (SAR) is reset on the positive edge of the start conversion (SC) pulse. The conversion is begun on the falling edge of the start conversion pulse. A conversion in process will be interrupted by receipt of a new start conversion pulse. Continuous conversion may be accomplished by tying the end-of-conversion (EOC) output to the SC input. If used in this mode, an external start conversion pulse should be applied after power up. End-of-conversion will go low between 0 and 8 clock pulses after the rising edge of start conversion.

The most important section of the A/D converter is the comparator. It is this section which is responsible for the ultimate accuracy of the entire converter. It is also the comparator drift which has the greatest influence on the repeatability of the device. A chopper-stabilized comparator provides the most effective method of satisfying all the converter requirements.

The chopper-stabilized comparator converts the DC input signal into an AC signal. This signal is then fed through a high gain AC amplifier and has the DC level restored. This technique limits the drift component of the amplifier since the drift is a DC component which is not passed by the AC amplifier. This makes the entire A/D converter extremely insensitive to temperature, long term drift and input offset errors.

Figure 4 shows a typical error curve for the ADC0808 as measured using the procedures outlined in AN-179.

Functional Description (Continued)

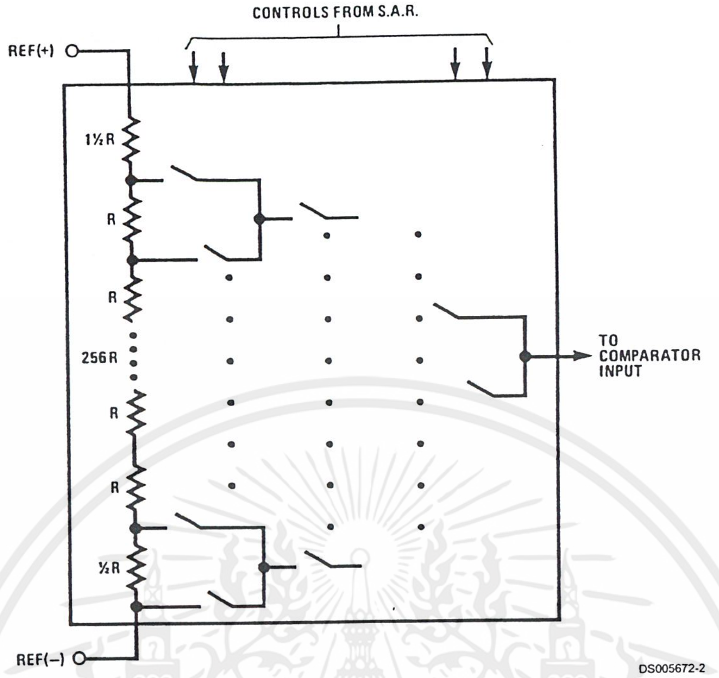


FIGURE 1. Resistor Ladder and Switch Tree

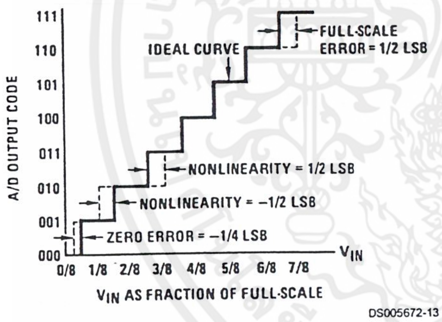


FIGURE 2. 3-Bit A/D Transfer Curve

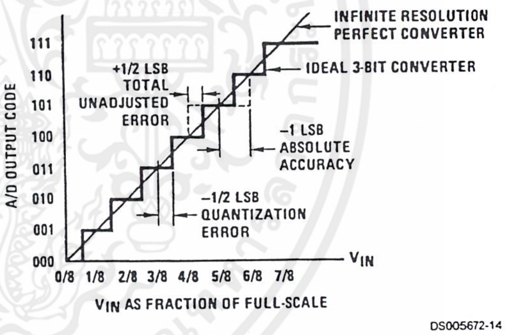


FIGURE 3. 3-Bit A/D Absolute Accuracy Curve

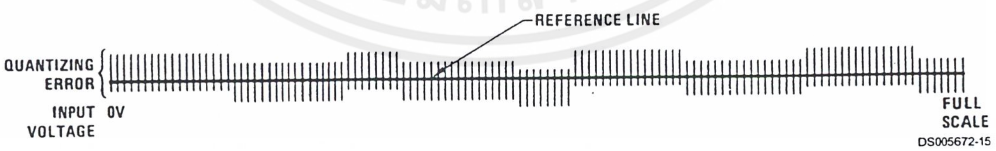


FIGURE 4. Typical Error Curve

Timing Diagram

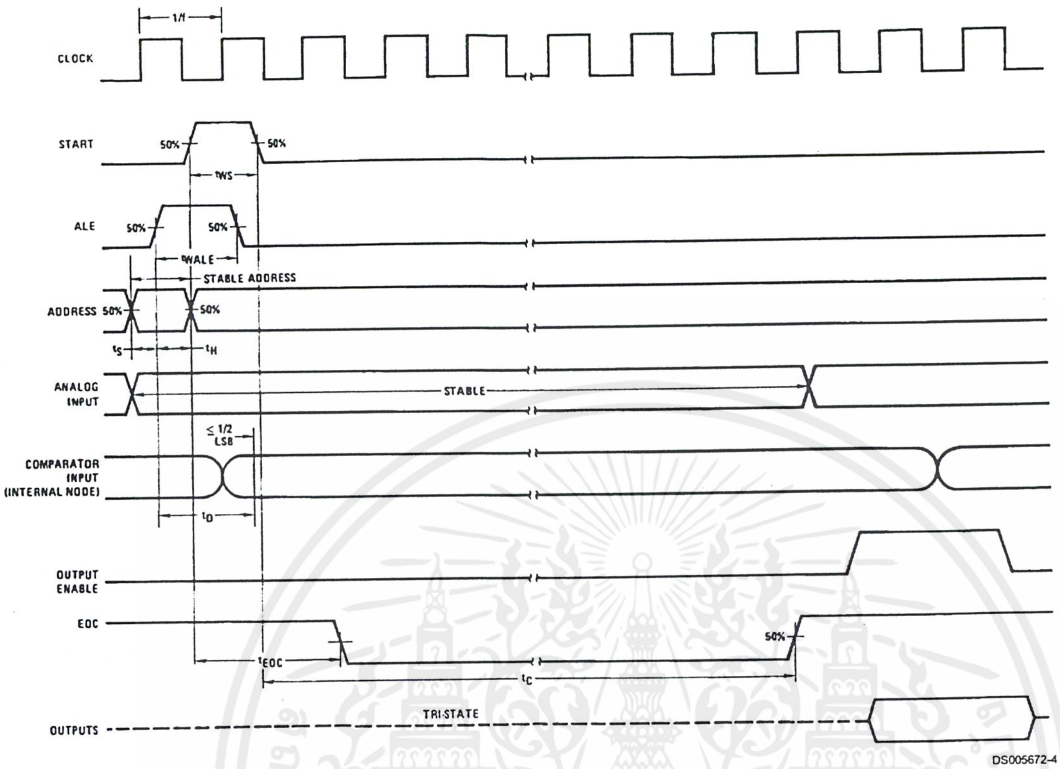


FIGURE 5.

Applications Information (Continued)

4.0 ANALOG COMPARATOR INPUTS

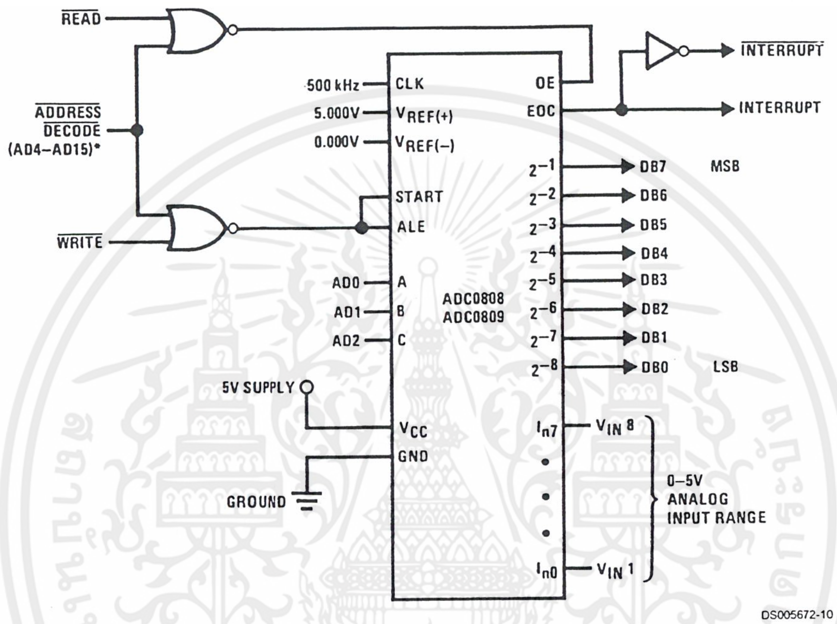
The dynamic comparator input current is caused by the periodic switching of on-chip stray capacitances. These are connected alternately to the output of the resistor ladder/switch tree network and to the comparator input as part of the operation of the chopper stabilized comparator.

The average value of the comparator input current varies directly with clock frequency and with V_{IN} as shown in *Figure 6*.

If no filter capacitors are used at the analog inputs and the signal source impedances are low, the comparator input current should not introduce converter errors, as the transient created by the capacitance discharge will die out before the comparator output is strobed.

If input filter capacitors are desired for noise reduction and signal conditioning they will tend to average out the dynamic comparator input current. It will then take on the characteristics of a DC bias current whose effect can be predicted conventionally.

Typical Application



*Address latches needed for 8085 and SC/MP interfacing the ADC0808 to a microprocessor

TABLE 2. Microprocessor Interface Table

PROCESSOR	$\overline{\text{READ}}$	$\overline{\text{WRITE}}$	INTERRUPT (COMMENT)
8080	$\overline{\text{MEMR}}$	$\overline{\text{MEMW}}$	INTR (Thru RST Circuit)
8085	$\overline{\text{RD}}$	$\overline{\text{WR}}$	INTR (Thru RST Circuit)
Z-80	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{INT}}$ (Thru RST Circuit, Mode 0)
SC/MP	NRDS	NWDS	SA (Thru Sense A)
6800	VMA•ϕ2•R/W	VMA•ϕ•$\overline{\text{R/W}}$	$\overline{\text{IRQA}}$ or $\overline{\text{IRQB}}$ (Thru PIA)

MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

MC54/74HC393

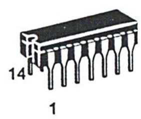
Dual 4-Stage Binary Ripple Counter
High–Performance Silicon–Gate CMOS

The MC54/74HC393 is identical in pinout to the LS393. The device inputs are compatible with standard CMOS outputs; with pullup resistors, they are compatible with LSTTL outputs.

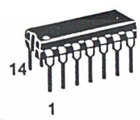
This device consists of two independent 4–bit binary ripple counters with parallel outputs from each counter stage. A ÷ 256 counter can be obtained by cascading the two binary counters.

Internal flip–flops are triggered by high–to–low transitions of the clock input. Reset for the counters is asynchronous and active–high. State changes of the Q outputs do not occur simultaneously because of internal ripple delays. Therefore, decoded output signals are subject to decoding spikes and should not be used as clocks or as strobes except when gated with the Clock of the HC393.

- Output Drive Capability: 10 LSTTL Loads
- Outputs Directly Interface to CMOS, NMOS, and TTL
- Operating Voltage Range: 2 to 6 V
- Low Input Current: 1 µA
- High Noise Immunity Characteristic of CMOS Devices
- In Compliance with the Requirements Defined by JEDEC Standard No. 7A
- Chip Complexity: 236 FETs or 59 Equivalent Gates



J SUFFIX
CERAMIC PACKAGE
CASE 632–08



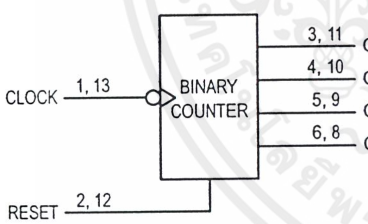
N SUFFIX
PLASTIC PACKAGE
CASE 646–06

D SUFFIX
SOIC PACKAGE
CASE 751A–03

ORDERING INFORMATION

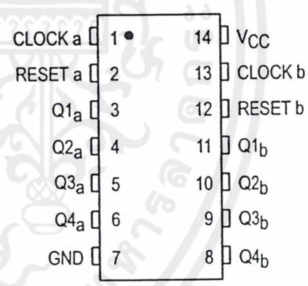
MC54HCXXXJ	Ceramic
MC74HCXXXN	Plastic
MC74HCXXXD	SOIC

LOGIC DIAGRAM



PIN 14 = V_{CC}
PIN 7 = GND

PIN ASSIGNMENT

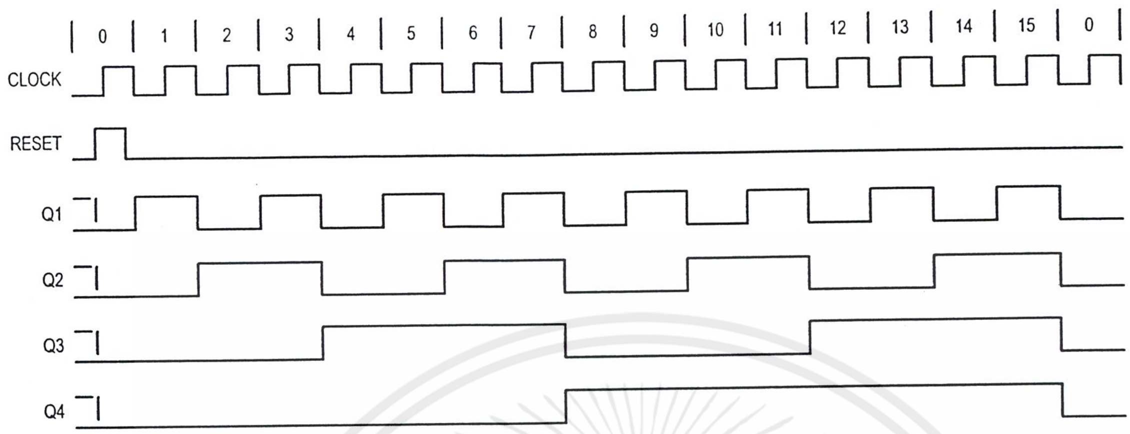


FUNCTION TABLE

Inputs		
Clock	**Reset**	**Outputs**
X	H	L
H	L	No Change
L	L	No Change
⎯⌐ (rising edge)	L	No Change
¬⎯ (falling edge)	L	Advance to Next State

MOTOROLA

TIMING DIAGRAM



COUNT SEQUENCE

Count	Outputs			
	Q4	Q3	Q2	Q1
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

FEATURES

- Synchronous counting and loading
- Two count enable inputs for n-bit cascading
- Positive-edge triggered clock
- Asynchronous reset
- Output capability: standard
- I_{CC} category: MSI

GENERAL DESCRIPTION

The 74HC/HCT161 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). They are specified in compliance with JEDEC standard no. 7A.

The 74HC/HCT161 are synchronous presettable binary counters which feature an internal look-ahead carry and can be used for high-speed counting.
Synchronous operation is provided by having all flip-flops clocked simultaneously on the positive-going edge of the clock (CP).
The outputs (Q_0 to Q_3) of the counters may be preset to a HIGH or LOW level. A LOW level at the parallel enable input ($\overline{PE}$) disables the counting action and causes the data at the data inputs (D_0 to D_3) to be loaded into the counter on the positive-going edge of the clock (providing that the set-up and hold time requirements for $\overline{PE}$ are met). Preset takes place regardless of the levels at count enable inputs (CEP and CET).

A LOW level at the master reset input ($\overline{MR}$) sets all four outputs of the flip-flops (Q_0 to Q_3) to LOW level regardless of the levels at CP, $\overline{PE}$, CET and CEP inputs (thus providing an asynchronous clear function).

The look-ahead carry simplifies serial cascading of the counters. Both count enable inputs (CEP and CET) must be HIGH to count. The CET input is fed forward to enable the terminal count output (TC). The TC output thus enabled will produce a HIGH output pulse of a duration approximately equal to a HIGH level output of Q_0. This pulse can be used to enable the next cascaded stage.

The maximum clock frequency for the cascaded counters is determined by the CP to TC propagation delay and CEP to CP set-up time, according to the following formula:

$$f_{max} = \frac{1}{t_{P(max)}\ (\text{CP to TC}) + t_{SU}\ (\text{CEP to CP})}$$

QUICK REFERENCE DATA

GND = 0 V; T_{amb} = 25 °C; $t_r = t_f$ = 6 ns

SYMBOL	PARAMETER	CONDITIONS	TYPICAL		UNIT
			HC	HCT	
t_{PHL}/t_{PLH}	propagation delay	C_L = 15 pF; V_{CC} = 5 V			
	CP to Q_n		19	20	ns
	CP to TC		21	24	ns
	$\overline{MR}$ to Q_n		20	25	ns
	$\overline{MR}$ to TC		20	26	ns
	CET to TC		10	14	ns
f_{max}	maximum clock frequency		44	45	MHz
C_I	input capacitance		3.5	3.5	pF
C_{PD}	power dissipation capacitance per package	notes 1 and 2	33	35	pF

Notes

1. C_{PD} is used to determine the dynamic power dissipation (P_D in μW):

   $P_D = C_{PD} \times V_{CC}^2 \times f_i + \sum (C_L \times V_{CC}^2 \times f_o)$ where:

   f_i = input frequency in MHz

   f_o = output frequency in MHz

   $\sum (C_L \times V_{CC}^2 \times f_o)$ = sum of outputs

   C_L = output load capacitance in pF

   V_{CC} = supply voltage in V

2. For HC the condition is V_I = GND to V_{CC}

   For HCT the condition is V_I = GND to V_{CC} − 1.5 V

Presettable synchronous 4-bit binary counter; asynchronous reset

74HC/HCT161

ORDERING INFORMATION

See *"74HC/HCT/HCU/HCMOS Logic Package Information"*.

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1	$\overline{MR}$	asynchronous master reset (active LOW)
2	CP	clock input (LOW-to-HIGH, edge-triggered)
3, 4, 5, 6	D_0 to D_3	data inputs
7	CEP	count enable input
8	GND	ground (0 V)
9	$\overline{PE}$	parallel enable input (active LOW)
10	CET	count enable carry input
14, 13, 12, 11	Q_0 to Q_3	flip-flop outputs
15	TC	terminal count output
16	V_{CC}	positive supply voltage

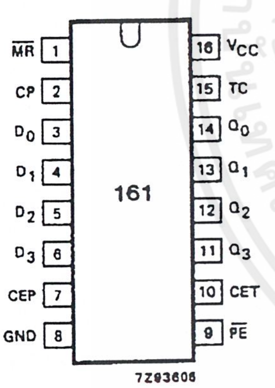


Fig.1 Pin configuration.

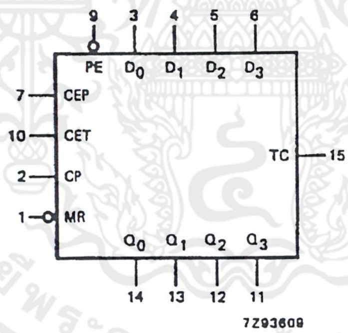


Fig.2 Logic symbol.

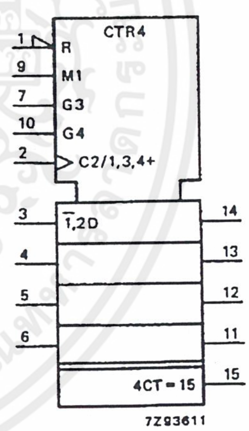


Fig.3 IEC logic symbol.

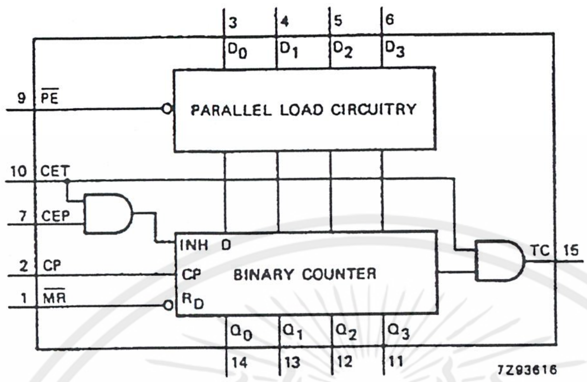


Fig.4 Functional diagram.

FUNCTION TABLE

OPERATING MODE	INPUTS						OUTPUTS	
	$\overline{MR}$	CP	CEP	CET	$\overline{PE}$	D_n	Q_n	TC
reset (clear)	L	X	X	X	X	X	L	L
parallel load	H H	↑ ↑	X X	X X	l l	l h	L H	L (1)
count	H	↑	h	h	h	X	count	(1)
hold (do nothing)	H H	X X	l X	X l	h h	X X	q_n q_n	(1) L

Note

1. The TC output is HIGH when CET is HIGH and the counter is at terminal count (HHHH).
   H = HIGH voltage level
   h = HIGH voltage level one set-up time prior to the LOW-to-HIGH CP transition
   L = LOW voltage level
   l = LOW voltage level one set-up time prior to the LOW-to-HIGH CP transition
   q = lower case letters indicate the state of the referenced output one set-up time prior to the LOW-to-HIGH CP transition
   X = don't care
   ↑ = LOW-to-HIGH CP transition

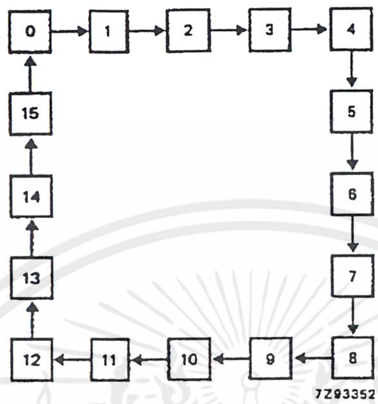


Fig.5 State diagram.

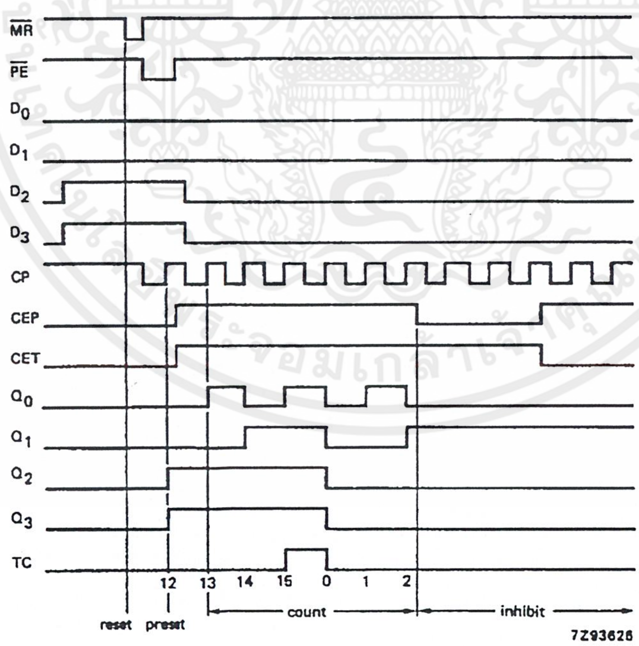
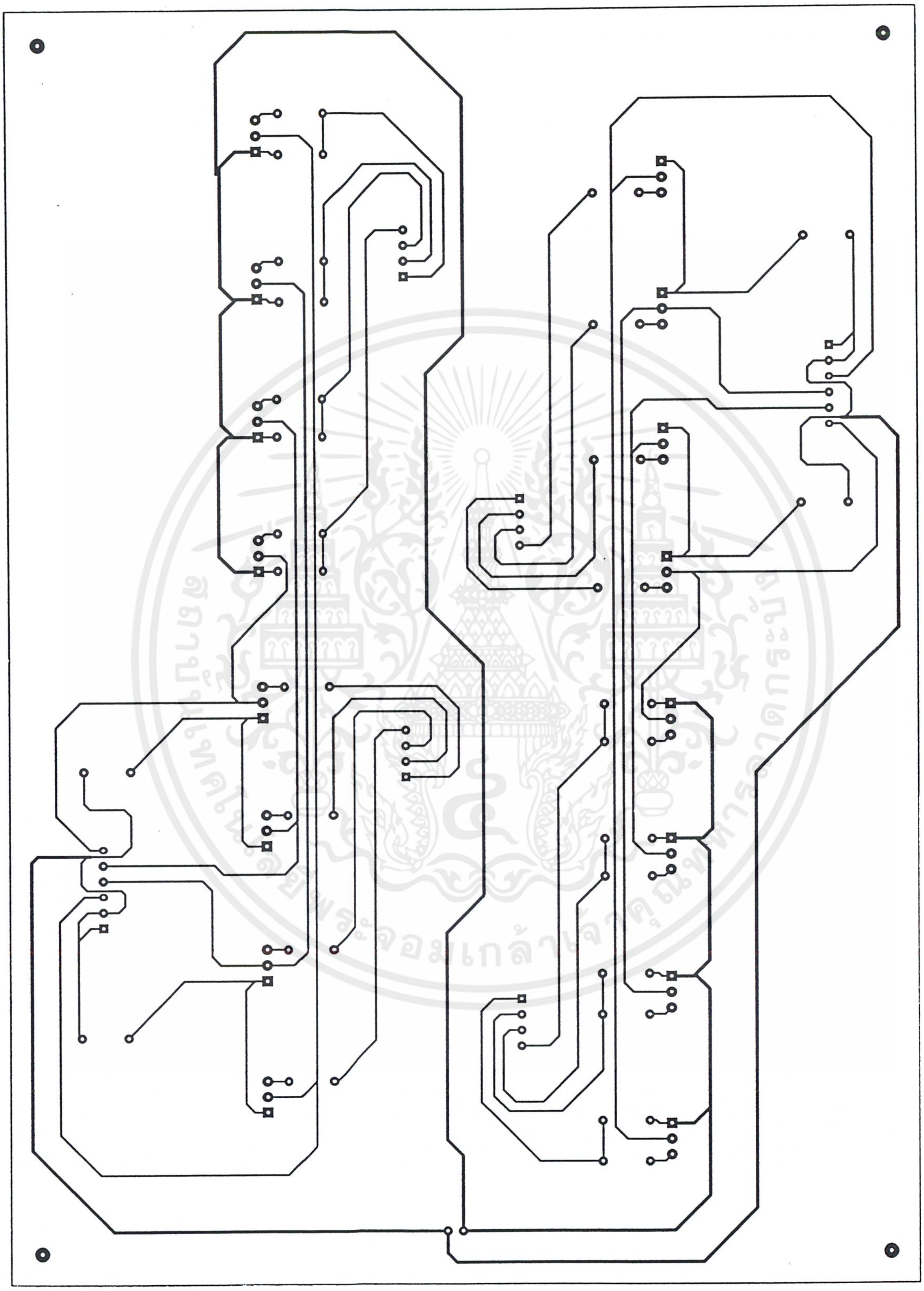


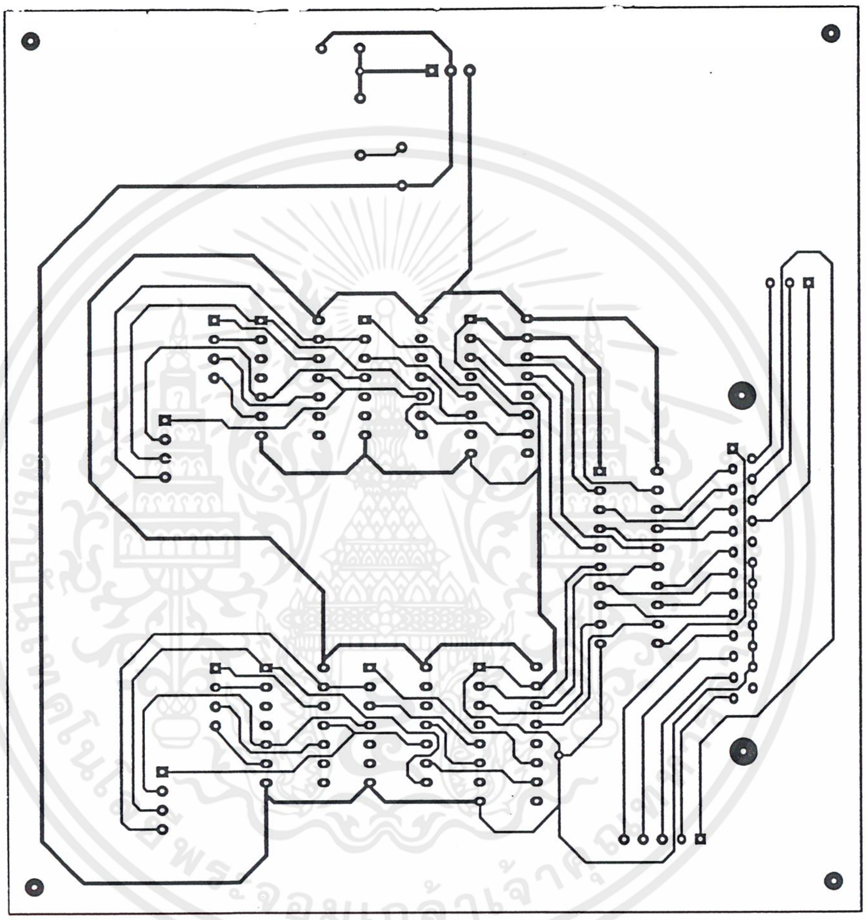
Fig.6 Typical timing sequence: reset outputs to zero; preset to binary twelve; count to thirteen, fourteen, fifteen, zero, one and two; inhibit.

ภาคผนวก ข

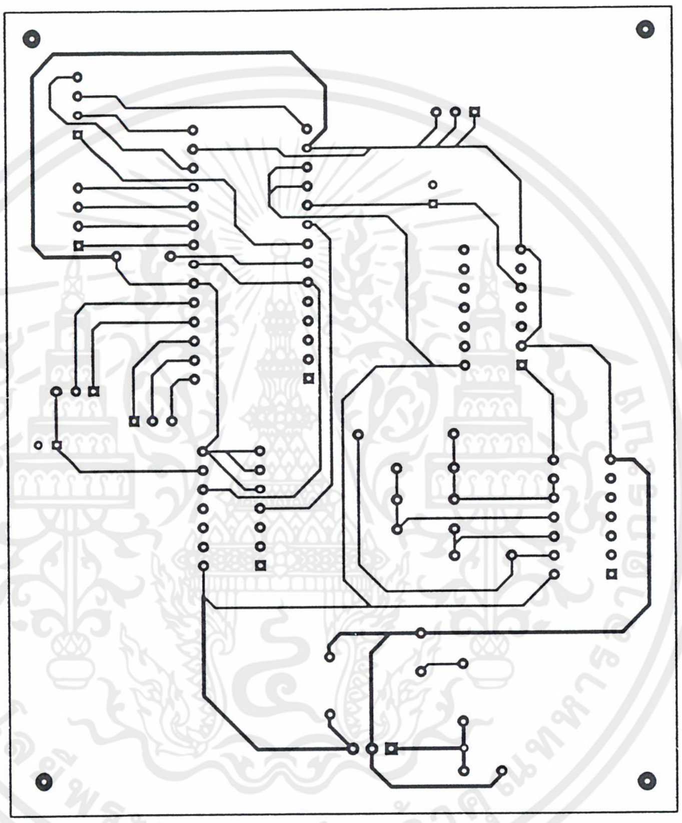
ลายทองแดงของวงจร



รูปที่ ข-1 แผ่นวงจรพิมพ์ของบอร์ดควบคุมของสเต็ปปิ้งมอเตอร์ 1



รูปที่ ข-2 แผ่นวงจรพิมพ์ของบอร์ดควบคุมของสเต็ปปิ้งมอเตอร์ 2



รูปที่ ข-3 แผ่นวงจรพิมพ์ของบอร์ดควบคุมของสเต็ปปิ้งมอเตอร์ 3

ภาคผนวก ค

โปรแกรมควบคุมการทำงาน

โปรแกรมการควบคุมมอเตอร์

```
Private Sub Comm_back1_Click()
main_menu_form.Show
Unload test1_motor_form
End Sub
Sub latch()
Call delay
Out &H37A, &HF
delay
Out &H37A, &H0
End Sub
Sub delay()             'frequency of computer
valspeed = Val(Text2.Text)
    times = Timer
    Do
    DoEvents
    Loop Until Timer >= times + valspeed
    End Sub
Private Sub comm_left_Click()
Lefts = True
Rights = False
Stops = False
valstep = Val(Text1.Text)
valspeed = Val(Text2.Text)
valcount = 0
nub% = 0
Dim step_out1(4) As Integer
Dim step_out2(4) As Integer
Dim step_out3(9) As Integer
        step_out1(1) = &H1
```

```
        step_out1(2) = &H2
        step_out1(3) = &H4
        step_out1(4) = &H8

        step_out2(1) = &H9
        step_out2(2) = &H3
        step_out2(3) = &H6
        step_out2(4) = &HC

        step_out3(1) = &H9
        step_out3(2) = &H1
        step_out3(3) = &H3
        step_out3(4) = &H2
        step_out3(5) = &H6
        step_out3(6) = &H4
        step_out3(7) = &HC
        step_out3(8) = &H8
        step_out3(9) = &H0

If Val(Text2) >= 0.001 Then

'------------1 phase --continuous drive---------------
   If Option1phase.Value = True Then
         If Option_con = True Then
         i% = 1
              Do
              DoEvents
              m% = i% - 4 * (nub% \ 4)
              Out &H37A, &H0
              Out &H378, step_out1(m%)
```

```
            Call latch
            i% = i% + 1
            nub% = nub% + 1
            Loop Until Stops = True
        End If

'------------1 phase --count step drive---------------
        If Option_count = True Then
            For i% = 1 To valstep
            m% = i% - 4 * (nub \ 4)
            Out &H37A, &H0
            Out &H378, step_out1(m%)
            Call latch
            nub% = nub% + 1
            Next i%
         End If

    End If  'end of choose 1 phase

'------------2 phase --continuous drive---------------
    If Option2phase.Value = True Then
        If Option_con = True Then
            i% = 1
            Do
            DoEvents
            m% = i% - 4 * (nub% \ 4)
            Out &H37A, &H0
            Out &H378, step_out2(m%)
            Call latch
            i% = i% + 1
```

```
            nub% = nub% + 1
            Loop Until Stops = True
    End If

'------------2 phase --count step  drive---------------
    If Option_count = True Then
          For i% = 1 To valstep
          m% = i% - 4 * (nub \ 4)
          Out &H37A, &H0
          Out &H378, step_out2(m%)
          Call latch
          nub% = nub% + 1
          Next i%
     End If
  End If  'end of choose 2 phase

'------------half step--continuous drive--------------
  If Option_h_step.Value = True Then
    If Option_con = True Then
    i% = 1
            Do
            DoEvents
            m% = i% - 9 * (nub% \ 9)
      Debug.Print m%
            Out &H37A, &H0
            Out &H378, step_out3(m%)
            Call latch
            i% = i% + 1
      Debug.Print i%
            nub% = nub% + 1
```

```
        Debug.Print nub%
            Loop Until Stops = True
          End If

'------------half step--count step  drive---------------
      If Option_count = True Then
          For i% = 1 To valstep
          m% = i% - 9 * (nub \ 9)
          Out &H37A, &H0
          Out &H378, step_out3(m%)
          Call latch
          nub% = nub% + 1
          Next i%
      End If

   End If  'end of choose 2 phase

Else: MsgBox " motor can't drive ,value not over 0.001"
End If

End Sub
Private Sub CommRight_Click()
Lefts = False
Rights = True
Stops = False
valstep = Val(Text1.Text)
valspeed = Val(Text2.Text)
valcount = 0
nub% = 0
Dim step_out1(4) As Integer
```

```
Dim step_out2(4) As Integer
Dim step_out3(9) As Integer
        step_out1(1) = &H8
        step_out1(2) = &H4
        step_out1(3) = &H2
        step_out1(4) = &H1

        step_out2(1) = &HC
        step_out2(2) = &H6
        step_out2(3) = &H3
        step_out2(4) = &H6

        step_out3(1) = &H8
        step_out3(2) = &HC
        step_out3(3) = &H4
        step_out3(4) = &H6
        step_out3(5) = &H2
        step_out3(6) = &H3
        step_out3(7) = &H1
        step_out3(8) = &H9
        step_out3(9) = &H0

If Val(Text2) >= 0.001 Then
'------------1 phase --continuous drive---------------
   If Option1phase.Value = True Then
     If Option_con = True Then
          i% = 1
           Do
           DoEvents
           m% = i% - 4 * (nub% \ 4)
```

```
            Out &H37A, &H0
            Out &H378, step_out1(m%)
            Call latch
            i% = i% + 1
            nub% = nub% + 1
            Loop Until Stops = True
    End If
'------------1 phase --count step---------------
    If Option_count = True Then
          For i% = 1 To valstep
          m% = i% - 4 * (nub \ 4)
          Out &H37A, &H0
          Out &H378, step_out1(m%)
          Call latch
          nub% = nub% + 1
           Next i%
    End If

  End If  'end of choose 1  phase

'------------2 phase --continuous drive---------------
  If Option2phase.Value = True Then
    If Option_con = True Then
    i% = 1
            Do
            DoEvents
            m% = i% - 4 * (nub% \ 4)
            Out &H37A, &H0
            Out &H378, step_out2(m%)
            Call latch
```

```
            i% = i% + 1
            nub% = nub% + 1
            Loop Until Stops = True
      End If
'------------2 phase --count step  drive---------------
      If Option_count = True Then
          For i% = 1 To valstep
          m% = i% - 4 * (nub \ 4)
          Out &H37A, &H0
          Out &H378, step_out2(m%)
          Call latch
          nub% = nub% + 1
          Next i%
       End If
   End If  'end of choose 2 phase
'------------half step--continuous drive---------------
   If Option_h_step.Value = True Then
      If Option_con = True Then
      i% = 1
            Do
            DoEvents
            m% = i% - 9 * (nub% \ 9)
        Debug.Print m%
            Out &H37A, &H0
            Out &H378, step_out3(m%)
            Call latch
            i% = i% + 1
        Debug.Print i%
            nub% = nub% + 1
        Debug.Print nub%
```

```
                Loop Until Stops = True
            End If

'------------half step--count step  drive---------------
        If Option_count = True Then
              For i% = 1 To valstep
              m% = i% - 9 * (nub \ 9)
              Out &H37A, &H0
              Out &H378, step_out3(m%)
              Call latch
              nub% = nub% + 1
              Next i%
        End If
    End If  'end of choose 2 phase
Else: MsgBox " motor can't drive ,value not over 0.001"
End If

End Sub
Private Sub Comm_stop_Click()
    'Dim a As Integer
    Stops = True
    Do
    DoEvents
    Out &H378, &H0
    Call latch
    Loop Until (Lefts = True) Or (Rights = True)
End Sub
```

โปรแกรมทดสอบพอร์ตขนาน

```
Private Sub Com_back2_Click()
  main_menu_form.Show
  Unload test1_port_form
End Sub
Sub delay()
    times = Timer
    Do
    DoEvents
    Loop Until Timer >= times + 0.01
    End Sub
Sub delay2()
    times = Timer
    Do
    DoEvents
    Loop Until Timer >= times + 0.001
    End Sub
Private Sub Com_test_con_Click()
Dim com_com(4) As Integer
starts_control = True
stops_control = False
  Do
  DoEvents
    For m% = 0 To 3
    Com_con(m%).Caption = (&H37A And (2 ^ m%)) / (2 ^ m%)
    Call delay
    Next m%
  Loop Until stops_control = True
End Sub
```

```
Private Sub stop_control_Click()
starts_control = False
stops_control = True
  Do
  DoEvents
    For m% = 0 To 3
    Com_con(m%).Caption = (&H37A And 0)
    Call delay
    Next m%
  Loop Until starts_control = True
End Sub
Private Sub Com_test_status_Click()
starts_status = True
stops_status = False
  Do
  DoEvents
    For i% = 0 To 4
    Com_status(i%).Caption = (Inp(&H379) And &HF8 And (2 ^ (i + 3))) / (2 ^ (i + 3))
    Call delay
    Next i%
  Call delay
  Loop Until stops_status = True
End Sub
Private Sub stop_status_Click()
stops_status = True
starts_status = False
  Do
  DoEvents
    For j = 0 To 4
    Com_status(j).Caption = Inp(&H379) And &HF8 And 0
```

```
        Call delay
        Next j
    Loop Until starts_status = True
End Sub
Private Sub Com_test_data_Click()
starts_data = True
stops_data = False
    Do
    DoEvents
        For i% = 0 To 7
        com_data(i%).Caption = ((&H378) And (2 ^ i)) / (2 ^ i)
        Call delay2
        Next i
    Loop Until stops_data = True
End Sub
Private Sub stop_data_Click()
stops_data = True
starts_data = False
    Do
    DoEvents
      For i% = 0 To 7
      com_data(i%).Caption = ((&H378) And 0)
      Call delay
      Next i
    Loop Until starts_data = True
End Sub
```

โปรแกรมทดสอบมูลี่อัตโนมัติ

```
Private Sub com_back_Click()
main_menu_form.Show
Unload test1_blind_form
End Sub
Public Sub latch_motor()
Call delay_speed
Out &H37A, &HF
delay_speed
Out &H37A, &H0
End Sub
Public Sub delay_speed()
Const valspeed = 0.001
    times = Timer
    Do
    DoEvents
    Loop Until Timer >= times + valspeed
    End Sub
Public Sub delay_time()
times = Timer
    Do
    DoEvents
    Loop Until Timer >= times + 1
    End Sub
Public Sub step_down()
  valstep = step_text.Text
  valtime = time_text.Text
  vallight = light_text.Text
```

```
        nub% = 0
        Dim step_out1(4) As Integer
        step_out1(1) = &H1
        step_out1(2) = &H2
        step_out1(3) = &H4
        step_out1(4) = &H8
                Do
                DoEvents
                m% = i% - 4 * (nub% \ 4)
                Out &H37A, &H0
                Out &H378, step_out1(m%)
                latch_motor
                i% = i% + 1
                nub% = nub% + 1
                Loop Until nub = valstep
End Sub
Public Sub step_up()
    valstep = step_text.Text
        nub% = 0
        Dim step_out1(4) As Integer
        step_out1(1) = &H8
        step_out1(2) = &H4
        step_out1(3) = &H2
        step_out1(4) = &H1
                Do
                DoEvents
                m% = i% - 4 * (nub% \ 4)
                Out &H37A, &H0
                Out &H378, step_out1(m%)
                latch_motor
```

```
            i% = i% + 1
            nub% = nub% + 1
            Loop Until nub = valstep
End Sub
Function get_data() As Single
Dim num(8) As Integer
  Out &H37A, &H3      '0011 wr=1 ,input0=00
  Out &H37A, &HB      '1011 wr=0 ,input0=00
  Data = (Inp(&H379) And &HF8 Xor &H80)
  Out &H379, &H3
    For i = 7 To 0 Step -1
      If Data >= (2 ^ i) Then
      num(i) = Data \ (2 ^ i)
      Data = Data - (2 ^ i)
      Else: num(i) = 0
      End If
    '  Debug.Print num(i)
    Next i
    Sum_data = (num(3) * 0.03125 + num(4) * 0.0625 + num(5) * 0.125 + num(6) * 0.25 +
num(7) * 0.5) * 5
    'Debug.Print sum
    get_data = Sum_data
  End Function
Function get_resistance() As Single
Dim num(8) As Integer
  Out &H37A, &H7      '0011 wr=1 ,input2=10
  Out &H37A, &HF      '1011 wr=0 ,input2=10
  Data = (Inp(&H379) And &HF8 Xor &H80)
  Out &H379, &H7
    For i = 7 To 0 Step -1
```

```
        If Data >= (2 ^ i) Then
        num(i) = Data \ (2 ^ i)
        Data = Data - (2 ^ i)
        Else: num(i) = 0
        End If
      '  Debug.Print num(i)
      Next i
      Sum_resis = (num(3) * 0.03125 + num(4) * 0.0625 + num(5) * 0.125 + num(6) * 0.25 +
num(7) * 0.5) * 5
      'Debug.Print sum
      get_resistance = Sum_resis
   End Function
Private Sub com_start_Click()
valstep = step_text.Text
valtime = (time_text.Text)
vallight = light_text.Text
Const step_value = 0.0034

   For g = 1 To 10
   step_up
   Next g

     Do
     DoEvents
     start_resis = get_resistance
     resis_ideal0 = 2.6
     resis_error0 = (resis_ideal0 - step_start) / resis_ideal0 * 100

        If resis_ideal0 > start_resis Then
        step_up
```

```
        Else: If resis_ideal0 < start_resis Then step_down
        End If

    Loop Until resis_error0 < 5

    Do
  DoEvents
  times = Timer
        Do
        DoEvents
        ldr_value = get_data
        ldr_error = Abs(ldr_value - vallight) * 100 / vallight
           If ldr_value > vallight Then
           step_down
           Else: If ldr_value < vallight Then Call step_up
           End If
        Loop Until ldr_error <= 5
    delay_time
    Do
    DoEvents
     resis_value = get_resistance
     resis_ideal = step_value * valstep
  Debug.Print step_value
     resis_error = (resis_ideal - resis_value) / resis_value * 100

        If resis_ideal > resis_value Then
        step_up
        Else: step_down
        End If
```

```
   Loop Until resis_error < 5
   Loop Until Timer >= times + valtime
End Sub
```

โปรแกรมการจำลองการทำงานของมู่ลี่อัตโนมัติ

```
Public Sub latch_motor()
Call delay_speed
Out &H37A, &HF
delay_speed
Out &H37A, &H0
End Sub
Public Sub delay_speed()
Const valspeed = 0.001
      times = Timer
      Do
      DoEvents
      Loop Until Timer >= times + valspeed
      End Sub
Public Sub delay_time()
times = Timer
      Do
      DoEvents
      Loop Until Timer >= times + 1
      End Sub
Public Sub step_down()
   valstep = 70
   vallight = 2
   Const step_min = 0.12
   nub% = 0
      Dim step_out1(4) As Integer
```

```
        step_out1(1) = &H1
        step_out1(2) = &H2
        step_out1(3) = &H4
        step_out1(4) = &H8
            check_motor_step = get_resistance
            If check_motor_low >= step_min Then
              Do
              DoEvents
              m% = i% - 4 * (nub% \ 4)
              Out &H37A, &H0
              Out &H378, step_out1(m%)
              latch_motor
              i% = i% + 1
              nub% = nub% + 1
              Loop Until nub = valstep
            End If
End Sub
Public Sub step_up()
  valstep = 70
  Const step_max = 2.6
  nub% = 0
    Dim step_out1(4) As Integer
    step_out1(1) = &H8
    step_out1(2) = &H4
    step_out1(3) = &H2
    step_out1(4) = &H1
        check_motor_step = get_resistance
        If check_motor_high <= step_max Then
          Do
          DoEvents
```

```
            m% = i% - 4 * (nub% \ 4)
            Out &H37A, &H0
            Out &H378, step_out1(m%)
            latch_motor
            i% = i% + 1
            nub% = nub% + 1
            Loop Until nub = valstep
          End If
End Sub
Function get_data() As Single
Dim num(8) As Integer
   Out &H37A, &H3      '0011 wr=1 ,input0=00
   Out &H37A, &HB      '1011 wr=0 ,input0=00
   Data = (Inp(&H379) And &HF8 Xor &H80)
   Out &H379, &H3
      For i = 7 To 0 Step -1
         If Data >= (2 ^ i) Then
         num(i) = Data \ (2 ^ i)
         Data = Data - (2 ^ i)
         Else: num(i) = 0
         End If
      '  Debug.Print num(i)
      Next i
      Sum_data = (num(3) * 0.03125 + num(4) * 0.0625 + num(5) * 0.125 + num(6) * 0.25 +
num(7) * 0.5) * 5
      'Debug.Print sum
      get_data = Sum_data
   End Function
Function get_resistance() As Single
Dim num(8) As Integer
```

```
  Out &H37A, &H7      '0011 wr=1 ,input2=10
  Out &H37A, &HF      '1011 wr=0 ,input2=10
  Data = (Inp(&H379) And &HF8 Xor &H80)
  Out &H379, &H7
    For i = 7 To 0 Step -1
      If Data >= (2 ^ i) Then
      num(i) = Data \ (2 ^ i)
      Data = Data - (2 ^ i)
      Else: num(i) = 0
      End If
    '  Debug.Print num(i)
    Next i
    Sum_resis = (num(3) * 0.03125 + num(4) * 0.0625 + num(5) * 0.125 + num(6) * 0.25 +
num(7) * 0.5) * 5
    'Debug.Print sum
    get_resistance = Sum_resis
  End Function
Private Sub start_demo_Click()
Const valstep = 70
Const vallight = 2
Const step_value = 0.0034
stops_demo = False
start_demo = True

  For g = 1 To 10
  step_up
  Next g

    Do
    DoEvents
```

```
start_resis = get_resistance
resis_ideal0 = 2.6
resis_error0 = (resis_ideal0 - step_start) / resis_ideal0 * 100

    If resis_ideal0 > start_resis Then
    step_up
    Else: If resis_ideal0 < start_resis Then step_down
    End If

Loop Until resis_error0 < 5

Do
DoEvents
times = Timer
    resis_value = get_resistance
    Do
    DoEvents
    ldr_value = get_data
    ldr_error = Abs(ldr_value - vallight) * 100 / vallight
        If ldr_value > vallight Then
        step_down
        Else: If ldr_value < vallight Then step_up
        End If
    Loop Until ldr_error <= 5
 delay_time
 Do
 DoEvents
  resis_value = get_resistance
  resis_ideal = step_value * valstep
Debug.Print step_value
```

```
        resis_error = (resis_ideal - resis_value) / resis_value * 100

            If resis_ideal > resis_value Then
            step_up
            Else: If resis_ideal < resis_value Then step_down
            End If

    Loop Until resis_error < 5
    Loop Until stops_demo = True
End Sub
Private Sub stop_demo_Click()
stops_demo = True
start_demo = False
main_menu_form.Show
Unload demo1_form

End Sub
```

เอกสารอ้างอิง

1. กฤษดา ใจเย็น, "เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตขนาน" , บริษัทอินโนเวตีฟ เอ็กเพอรีเมนจ์ จำกัด, 122 หน้า
2. โยธิน เปรมปราณีรัชต์, "ระบบเซอร์โวและอิเล็กทรอนิกส์คอนโทรลมอเตอร์" , เจแปน อินเตอร์เนชั่นแนล คอร์ปเปอเรชั่น เอเจนซี่, 504 หน้า, 2533
3. เศกสิทธ์ คำชมภู, "บอร์ดควบคุมสเต็ปเปอร์มอเตอร์", วารสารเซมิคอนดัคเตอร์อิเล็กทรอนิกส์ ", ฉบับที่ 169, 2540, หน้า 45-49
4. ชัยวัฒน์ ลิ้มพรจิตรวิไล, "MC-2 บอร์ดควบคุมสเต็ปเปอร์มอเตอร์ผ่านคอมพิวเตอร์", วารสารเซมิคอนดัคเตอร์อิเล็กทรอนิกส์ ", ฉบับที่ 175, 2540, หน้า 201-211
5. ไกรวุฒิ โรจน์ประเสริฐสุด, " บอร์ดควบคุมสเต็ปเปอร์ด้วยเครื่องพีซี", วารสารเซมิคอนดัคเตอร์อิเล็กทรอนิกส์ ", ฉบับที่ 146, 2538, หน้า 25-38

เว็บไซด์อ้างอิง

1. http://www.thaiio.com
2. http://www.ams2000.com
3. http://lineengineer.com
4. http://anaheimautomation.com

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้ด้วยความช่วยเหลือจากอาจารย์และบุคคลหลายท่าน

ขอขอบพระคุณอาจารย์วรพงศ์ ตั้งศรีรัตน์ ที่ช่วยให้คำปรึกษาและเอื้อเฟื้อเอกสารประกอบการทำปริญญานิพนธ์

ขอขอบพระคุณอาจารย์ภาควิชาวิศวกรรมระบบควบคุมทุกท่านที่ให้การประสิทธิ์ประสาทวิชา

ขอขอบคุณ ซูโม่ที่เอื้อเฟื้อรถ เล็กเชอร์ และมากมายนับไม่ถ้วน ผึ้งที่ช่วยเคี่ยวเข็น นุ้ยที่ให้ยืมอุปกรณ์ จอยและพี่สมบูรณ์ผู้ให้ความรู้ทางด้านอิเล็กทรอนิกส์ เดียวที่หาสแกนเนอร์และอำนวยความสะดวกสารพัด และเพื่อนภาคอีกมากมายที่ให้ความช่วยเหลือ

ท้ายที่สุดขอขอบพระคุณคุณพ่อ คุณแม่และครอบครัวของเราทั้งสองคนที่ให้ชีวิต กำลังใจและกำลังทรัพย์

นพวรรณ์ ตัญเจริญสุขจิต

นฤมล ใจสงเคราะห์