เรียนลัด

การควบคุมเซอร์โวมอเตอร์

สำหรับมือใหม่

OMRON

อรัมภบท

เครื่องจักรรุ่นใหม่ๆที่เราเห็นกันอยู่ในปัจจุบันนี้ ล้วนแล้วแต่ทำงานด้วยความเร็วสูงแทบทั้งสิ้น ไม่ว่าจะเป็นเครื่องจักรที่นำเข้าจากต่างประเทศ หรือเครื่องจักรที่ผลิตขึ้นใช้เองภายในประเทศก็ตาม เหตุผลหนึ่งคืออุตสาหกรรมปัจจุบันต้องแข่งขันกันในเรื่องของความเร็วในการผลิต เพื่อเพิ่มผลผลิตต่อหนึ่งหน่วยเวลาให้ทันต่อความต้องการของลูกค้า การพัฒนาเครื่องจักรเพื่อใช้กับงานที่ต้องการความเร็วสูงต้องหาอุปกรณ์ต้นกำลังที่ให้การตอบสนองต่อความเร็วสูง ควบคู่กับความถูกต้องแม่นยำมาใช้งาน อุปกรณ์ต้นกำลังที่ตอบสนองต่อความเร็วและมีความแม่นยำสูง ก็คือ เซอร์โวมอเตอร์ ซึ่งเป็นอุปกรณ์ที่ใช้ในการขับเคลื่อนเครื่องจักรอุตสาหกรรมปัจจุบัน ตัวอย่างเช่น เครื่องบรรจุภัณฑ์ความเร็วสูง , งานประกอบชิ้นส่วนอีเล็กทรอนิคส์ที่ต้องการความแม่นยำสูง , เครื่องจักรที่ใช้ในอุตสาหกรรมยานยนต์ , เครื่องจักรที่ใช้ในอุตสาหกรรมกระดาษ เป็นต้น จากคุณสมบัติเด่นในเรื่องของความเร็วและความแม่นยำ ทำให้เซอร์โวมอเตอร์ได้รับความสนใจและมีผู้หันมาศึกษาระบบการควบคุมเซอร์โวมอเตอร์กันมากขึ้น

“เซอร์โวมอเตอร์สำหรับมือใหม่” จะทำให้ท่านเห็นภาพรวมของการควบคุมเซอร์โวมอเตอร์ โดยนำเสนอในเชิงสรุปเนื้อหา ตัดประเด็นที่เข้าใจยากออกไป เน้นให้ข้อมูลที่ผู้เริ่มต้นศึกษาควรทราบ ในตอนท้าย จะแสดงตัวอย่างวิธีการควบคุมเซอร์โวมอเตอร์อย่างง่ายๆ เป็นลำดับขั้นตอนให้ปฏิบัติตามเพื่อให้เกิดความมั่นใจและสามารถนำความรู้ที่ได้ไปต่อยอด เพื่อควบคุมเซอร์โวมอเตอร์ในระบบที่มีความซับซ้อนมากขึ้นต่อไป

OMRON ELECTRONICS CO., LTD.

สารบัญ

1. ข้อมูลทั่วไปเกี่ยวกับเซอร์โวมอเตอร์ 1

ระบบควบคุมเซอร์โวมอเตอร์คืออะไร

ทำไมต้องใช้เซอร์โวมอเตอร์

เซอร์โวมอเตอร์มีกี่ประเภท

AC เซอร์โวมอเตอร์มีโครงสร้างอย่างไร

เอ็นโค้ดเดอร์ของเซอร์โวมอเตอร์มีกี่แบบ

คุณสมบัติที่ดีของเซอร์โวมอเตอร์มีอะไรบ้าง

ขนาดของเซอร์โวมอเตอร์ระบุขนาดเป็นอะไร

การควบคุมเซอร์โวมอเตอร์มีหลักการอย่างไร

ระบบควบคุมเซอร์โวมอเตอร์ประกอบด้วยอุปกรณ์อะไรบ้าง

คอนโทรลเลอร์ทำหน้าที่อะไร

ไดร์เวอร์ทำหน้าที่อะไร

ไดร์เวอร์มีกี่ประเภท

เซอร์โวมอเตอร์แบ่งตามประเภทการใช้งานได้กี่แบบ

2. การเลือกขนาดของเซอร์โวมอเตอร์และไดร์เวอร์ 7

3. การเลือกคอนโทรลเลอร์ 10

4 เทคนิคการควบคุมเซอร์โวมอเตอร์อย่างง่าย 11

ตัวอย่างที่ 1 การสั่งให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Relative Position (19)

ตัวอย่างที่ 2 การสั่งให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Absolute Position (36)

OMRON ELECTRONICS CO., LTD.

1. ข้อมูลทั่วไปเกี่ยวกับเซอร์โวมอเตอร์

ระบบควบคุมเซอร์โวมอเตอร์ คือ อะไร?

ระบบควบคุมเซอร์โวมอเตอร์ คือ ระบบควบคุมเครื่องจักรกลให้ทำงานตอบสนองต่อการเคลื่อนที่ ในเชิง , ความเร็ว, อัตราเร่ง, ตำแหน่ง และแรงบิด โดยการตอบสนองนั้นต้องมีความถูกต้องแม่นยำมากที่สุดในระยะเวลาที่สั้นที่สุด

ทำไมต้องใช้เซอร์โวมอเตอร์?

เครื่องจักรที่นำเซอร์โวมอเตอร์มาใช้เป็นอุปกรณ์ต้นกำลัง เนื่องจากสามารถเปลี่ยนแปลงความเร็วรอบได้อย่างรวดเร็วและมีคุณสมบัติของการเคลื่อนที่เชิงไดนามิกส์สูง ซึ่งมอเตอร์ทั่วไปไม่ว่าจะเป็นมอเตอร์ไฟฟ้ากระแสสลับหรือกระแสตรงไม่สามารถตอบสนองได้

เซอร์โวมอเตอร์มีกี่ประเภท?

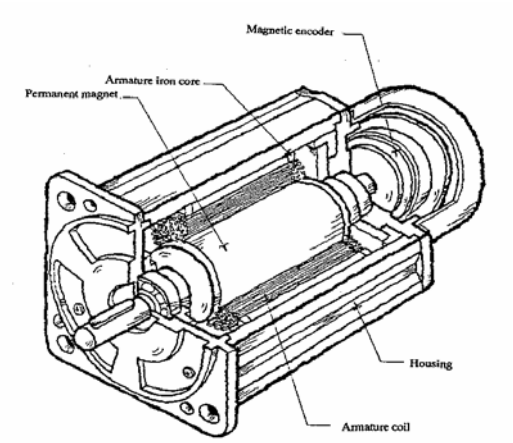
เซอร์โวมอเตอร์มีทั้งแบบที่ใช้ไฟกระแสสลับและไฟกระแสตรง ในช่วงหลายปีที่ผ่านมาการใช้งานเซอร์โวมอเตอร์แบบไฟกระแสตรงเป็นที่แพร่หลายมาก เนื่องจากการควบคุมกระแสไฟฟ้าสูงๆที่จ่ายให้กับมอเตอร์ต้องใช้เอสซีอาร์(SCRs)ในการควบคุม ต่อมาพัฒนาการของอุปกรณ์เพาเวอร์อีเลคทรอนิคส์ทำให้ทรานซิสเตอร์(Transistor) มีขีดความสามารถในการควบคุมกระแสสูงๆเพิ่มขึ้นและยังสามารถทำงานได้ที่ความถี่สูง ทำให้การใช้งานเซอร์โวมอเตอร์แบบไฟกระแสสลับเป็นที่นิยมในปัจจุบัน

AC เซอร์โวมอเตอร์มีโครงสร้างอย่างไร ?

เซอร์โวมอเตอร์ที่นิยมใช้กันมากในปัจจุบันประกอบด้วยโครงสร้าง 3 ส่วนคือ

1.) โรเตอร์ (Rotor)
   คือ ส่วนที่เคลื่อนที่(หมุน) ทำจากแม่เหล็กถาวร
2.) สเตเตอร์(Stator)
   คือ ส่วนที่อยู่กับที่มีขดลวดพันอยู่ในร่องสล็อตทำหน้าที่สร้างสนามแม่เหล็กหมุน เพื่อทำให้โรเตอร์เคลื่อนที่
3.) เอ็นโค้ดเดอร์ (Encoder)
   คือ ส่วนที่ต่ออยู่กับเพลาของมอเตอร์ ทำหน้าที่สร้างสัญญาณพัลส์ตามจังหวะการหมุนของมอเตอร์เพื่อเป็นสัญญาณป้อนกลับให้กับระบบ

OMRON ELECTRONICS CO., LTD.



ภาพแสดงลักษณะโครงสร้างของเซอร์โวมอเตอร์

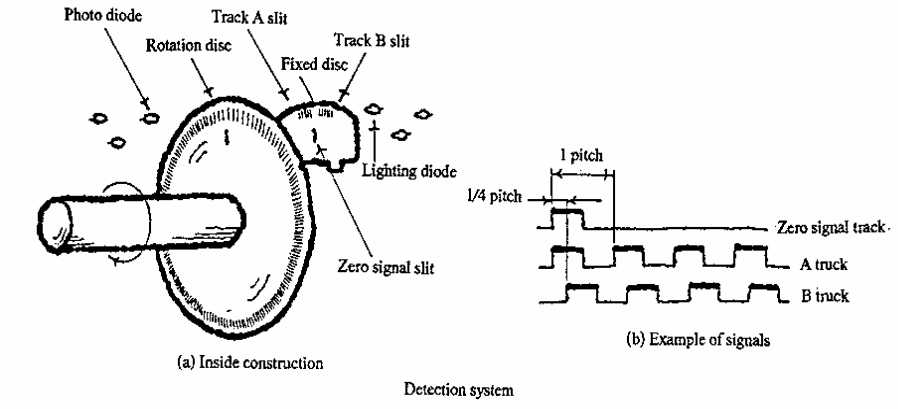
เอ็นโค้ดเดอร์ที่เซอร์โวมอเตอร์มีกี่แบบ?

มี 2 แบบ

1.) <u>Incremental Encoder</u> (มีสายสัญญาณ 3 เส้น คือเฟส A, เฟส B, เฟส Z)

หลักการของ Increment Encoder คือ ลำแสงจะถูกยิงจาก Lighting Diode ผ่าน Fixed Disc ไปยัง Rotation Disc ที่ ติดตั้งอยู่บนแกนเพลา โดยมี Photo Diode เป็นตัวรับแสง ลำแสงจะผ่านรูบน Fixed Disc และ Rotation Disc ตามจังหวะของการหมุนของมอเตอร์ ทำให้ได้สัญญาณไฟฟ้าออกมาจาก Photo Diode เนื่องจากรูของ A และ B บน Fixed Disc จะต่างเฟสกันอยู่ 90 องศา ถ้านับค่าพัลส์ที่ได้จากตัวเอ็นโค้ดเดอร์ ก็จะเป็นค่ามุมของการหมุน ส่วนเอาต์พุตเฟส A และ B ที่ต่างเฟสกันอยู่ 90 องศาจะเป็นตัวชี้ถึงทิศทางการหมุนของมอเตอร์ ส่วนเฟส Z หรือ Zero Signal เป็นตัวชี้จุด 0 องศาของการหมุน

ROI ELECTRONICS CO., LTD.



2.) Absolute Encoder (มีสายสัญญาณตามลักษณะของโค้ด เช่น Gray, Binary, BCD)

เอ็นโค้ดเดอร์อีกชนิดหนึ่งคือ Absolute Encoder สามารถเลือกเอาต์พุตได้หลายแบบเช่น Gray Code, Binary Code หรือ BCD Code การเลือกเซอร์โวมอเตอร์แบบ Absolute Encoder ไม่จำเป็นต้องเลือกโค้ดของเอาต์พุต ให้เลือกแต่เพียงความละเอียดที่ต้องการใช้งานเท่านั้น สิ่งที่แตกต่างจาก Incremental Encoder คือจำนวนสายสัญญาณเอาต์พุตที่มีมากกว่า ซึ่งขึ้นอยู่กับความละเอียดที่เลือกใช้ อีกประการหนึ่ง คือ ความหมายสัญญาณของ Absolute Encoder ณ เวลาหนึ่งจะให้ค่าออกมาเป็นค่าสมบูรณ์ไม่ใช่ค่าที่เปรียบเทียบจากจุดเริ่มต้นเหมือนกับแบบ Incremental Encoder ดังนั้นถ้าใช้เซอร์โวมอเตอร์ที่มีเอ็นโค้ดเดอร์แบบAbsolute Encoder ก็ไม่จำเป็นต้องค้นหาตำแหน่งเริ่มต้น (Origin search) ทุกครั้งที่เปิดเครื่องใหม่

คุณสมบัติของเซอร์โวมอเตอร์ที่ดีต้องมีอะไรบ้าง?

คุณสมบัติของเซอร์โวมอเตอร์ที่ดีต้องมีความสามารถดังต่อไปนี้

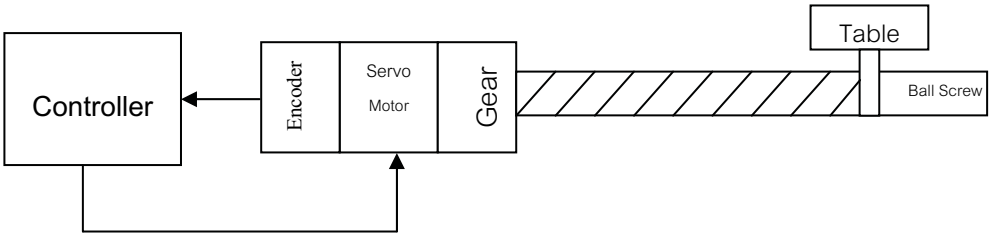
1.) สามารถทำงานที่ความเร็วต่างๆกันโดยไม่เกิดความร้อน
2.) สามารถที่จะรับแรงบิดของโหลดในขณะที่มอเตอร์หยุดนิ่ง
3.) สามารถทำงานที่ความเร็วรอบต่ำๆได้เป็นเวลานานโดยไม่เกิดความร้อน

ขนาดของเซอร์โวมอเตอร์ระบุหน่วยเป็นอะไร?

ขนาดของเซอร์โวมอเตอร์จะบอกเป็นวัตต์(Watt) ในท้องตลาดมีขนาดตั้งแต่ 30 W. จนถึง 15 KW.

การควบคุมเซอร์โวมอเตอร์มีหลักการอย่างไร?

ในระบบควบคุมเซอร์โวมอเตอร์ เราสามารถที่จะควบคุมพารามิเตอร์ทางกลต่างๆได้ เช่นการควบคุมตำแหน่ง , ควบคุมความเร็ว , ควบคุมแรงบิด หรือแม้แต่เส้นทางการเคลื่อนที่ก็สามารถทำได้ ทั้งนี้ขึ้นอยู่กับลักษณะการควบคุมที่เลือกใช้ โดยทั่วไปแล้วลักษณะการควบคุมเซอร์โวมอเตอร์ในอุตสาหกรรมมักจะเป็นแบบ SEMI-CLOSE LOOP (มีการป้อนกลับของสัญญาณเพื่อเปรียบเทียบ)



OMRON ELECTRONICS CO., LTD.

ระบบควบคุมเซอร์โวมอเตอร์ประกอบด้วยอุปกรณ์อะไรบ้าง?

ระบบควบคุมเซอร์โวมอเตอร์ประกอบด้วยส่วนสำคัญหลัก 3 ส่วน คือ

1.) คอนโทรลเลอร์ (Controller)

2.) ไดร์เวอร์ (Driver)

3.) เซอร์โวมอเตอร์และเอ็นโค้ดเดอร์ (Servo motor with Encoder)

คอนโทรลเลอร์ทำหน้าที่อะไร?

สร้างสัญญาณพัลส์ (Pulse Train Input), สัญญาณอนาลอก (Analog Input) จ่ายให้กับไดร์เวอร์เพื่อส่งผ่านไปยังเซอร์โวมอเตอร์ให้เคลื่อนที่(หมุน)ไปตามระยะทางหรือความเร็วที่ต้องการ

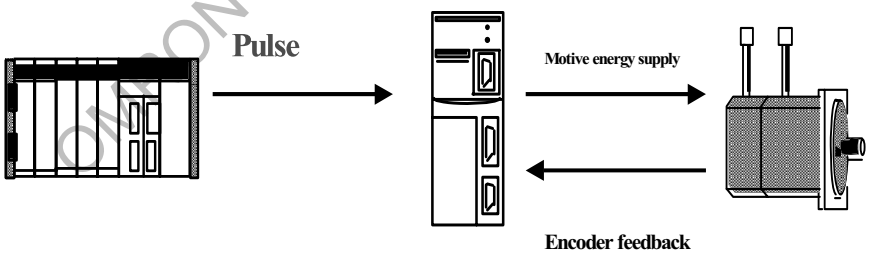
ไดร์เวอร์ทำหน้าที่อะไร?

ในการควบคุมเซอร์โวมอเตอร์จะประกอบไปด้วยพารามิเตอร์หลายอย่าง เช่น ความเร็ว , ตำแหน่ง , สัญญาณจากเอ็นโค้ดเดอร์ , การปรับตั้งค่าความผิดพลาด ฯ ซึ่งสัญญาณต่างๆเหล่านี้ไม่สามารถส่งไปยังเซอร์โวมอเตอร์ได้โดยตรง ต้องอาศัยเซอร์โวไดร์เวอร์เป็นตัวรวบรวมสัญญาณทั้งหมดเข้าด้วยกันเพื่อส่งไปควบคุมเซอร์โวมอเตอร์ให้ทำงานตามต้องการ

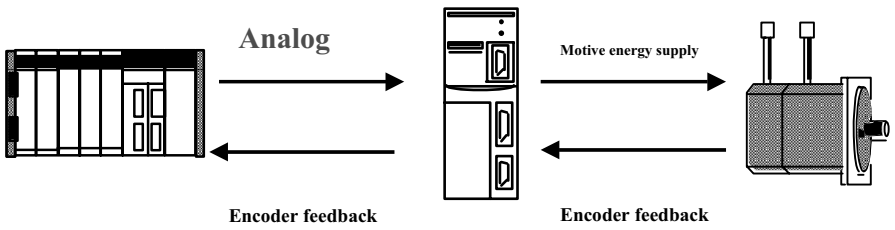
ไดร์เวอร์มีกี่ประเภท?

แบ่งเป็น 3 ประเภท ตามลักษณะการรับสัญญาณจากคอนโทลเลอร์ ได้แก่

1.) รับสัญญาณพัลส์ (Pulse Train Input Driver) เป็นการควบคุมตำแหน่ง,ความเร็ว

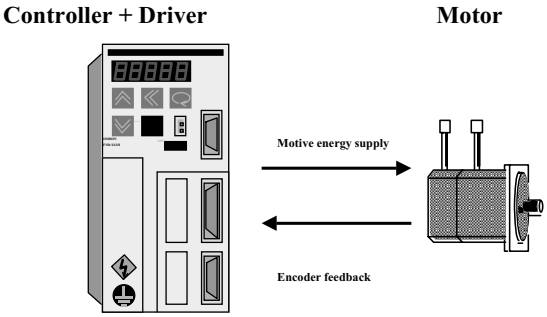


2.) รับสัญญาณอนาลอก (Analog Input Driver) เป็นการควบคุมความเร็วและแรงบิด



OMRON ELECTRONICS CO., LTD.

3.) ไม่รับสัญญาณจากภายนอกสามารถกำหนดตำแหน่งในตัว (Position Driver)



เซอร์โวมอเตอร์แบ่งตามประเภทการใช้งานได้กี่แบบ?

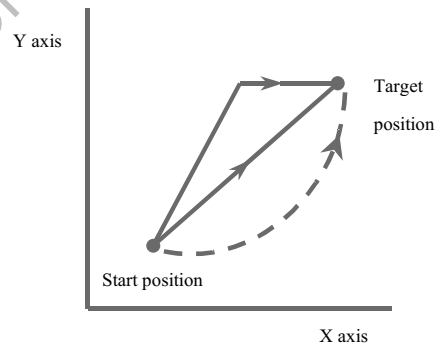
สามารถแบ่งตามประเภทการใช้งาน 3 แบบ คือ

1.) **ใช้ควบคุมตำแหน่ง** (Position Control)

แบ่งเป็น 2 ประเภท

1.1) PTP (Point to Point)

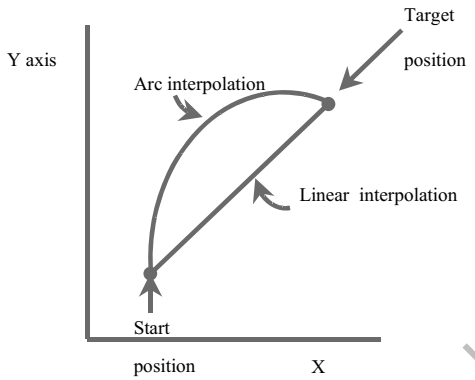
สนใจเฉพาะตำแหน่งเริ่มต้นและตำแหน่งเป้าหมายเท่านั้นไม่สนใจเส้นทางการเคลื่อนที่



OMRON ELECTRONICS CO., LTD.

1.2) CP (Continuous Path)

สนใจตำแหน่งเริ่มต้น เส้นทางการเคลื่อนที่ ตำแหน่งเป้าหมาย และต้องระบุว่าจะเคลื่อนที่เป็นแบบใด เช่น เส้นตรง (Linear) หรือแบบเส้นโค้ง (Arc)



การเคลื่อนที่แบบ PTP หรือ CP ต้องพิจารณาที่ตัวคอนโทรลเลอร์เป็นหลักว่ามีฟังก์ชันรองรับหรือไม่ โดยส่วนใหญ่แล้วคอนโทรลเลอร์จะรองรับ PTP เป็นฟังก์ชันพื้นฐาน

OMRON ELECTRONICS CO., LTD.

2.) **ใช้ควบคุมความเร็ว** (Speed Control)

ถ้าเป็นไดร์เวอร์แบบรับสัญญาณพัลส์(Pulse Train) จะใช้ความถี่ของสัญญาณพัลส์ในการควบคุมความเร็ว

ถ้าเป็นไดร์เวอร์แบบรับสัญญาณอนาลอก(Analog) ใช้ส่งสัญญาณแรงดันแบบอนาลอกไปควบคุมความเร็ว

3.) **ใช้การควบคุมแรงบิด** (Torque)

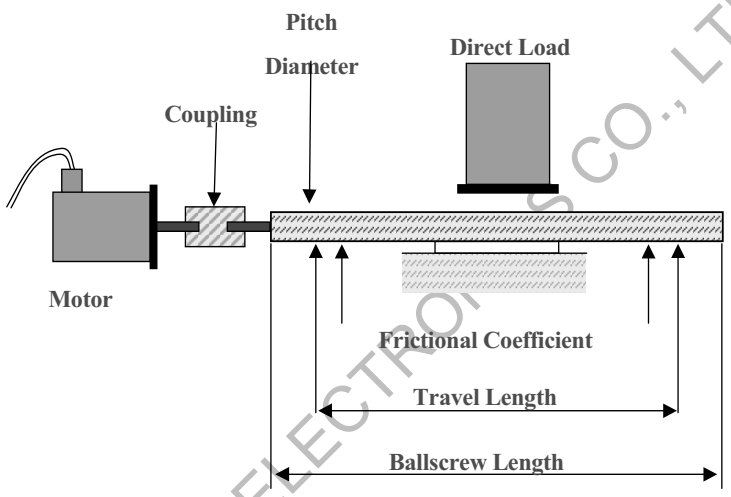
ถ้าเป็นไดร์เวอร์แบบรับสัญญาณ Pulse Train จะไม่สามารถควบคุมแรงบิดได้ต้องใช้ไดร์เวอร์แบบ Analog เท่านั้น

****

2. การเลือกขนาดของเซอร์โวมอเตอร์และไดร์เวอร์

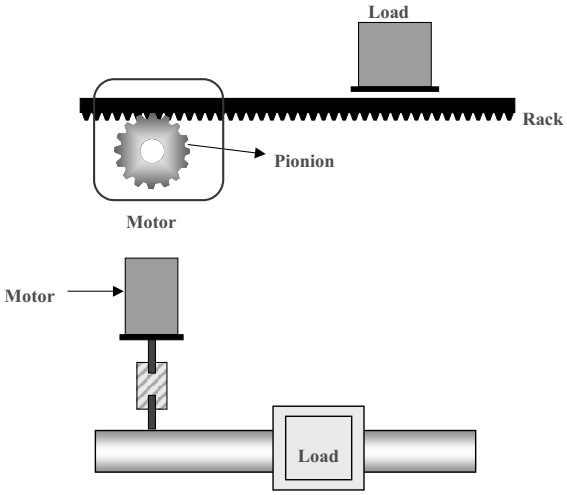
ขนาดของเซอร์โวมอเตอร์และเซอร์โวไดร์เวอร์จะบอกขนาดเป็นวัตต์ (Watt) การหาขนาดของเซอร์โวมอเตอร์ ผู้ใช้งานจำเป็นต้องรู้ระบบการทำงานทางกลด้วยว่าชิ้นส่วนทางกลที่จะนำเซอร์โวมอเตอร์ไปใช้งานเป็นอย่างไร เช่น มีขนาด , น้ำหนัก , ลักษณะรูปแบบการเคลื่อนที่เป็นอย่างไร เมื่อเราหาขนาดของเซอร์โวมอเตอร์ได้แล้ว ขนาดของเซอร์โวไดร์เวอร์ก็จะสามารถหาได้โดยอัตโนมัติ ปกติแล้วรูปแบบการทำงานหรือการเคลื่อนที่ทางกลที่พบบ่อยสามารถสรุปได้ 4 รูปแบบ คือ

1.) **ระบบการเคลื่อนที่แบบบอลสกูร** (BALL SCREW)

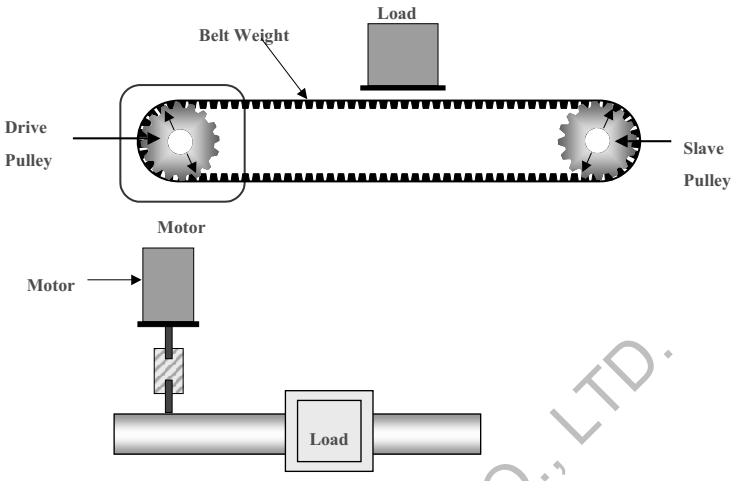


OMRON ELECTRONICS CO., LTD.

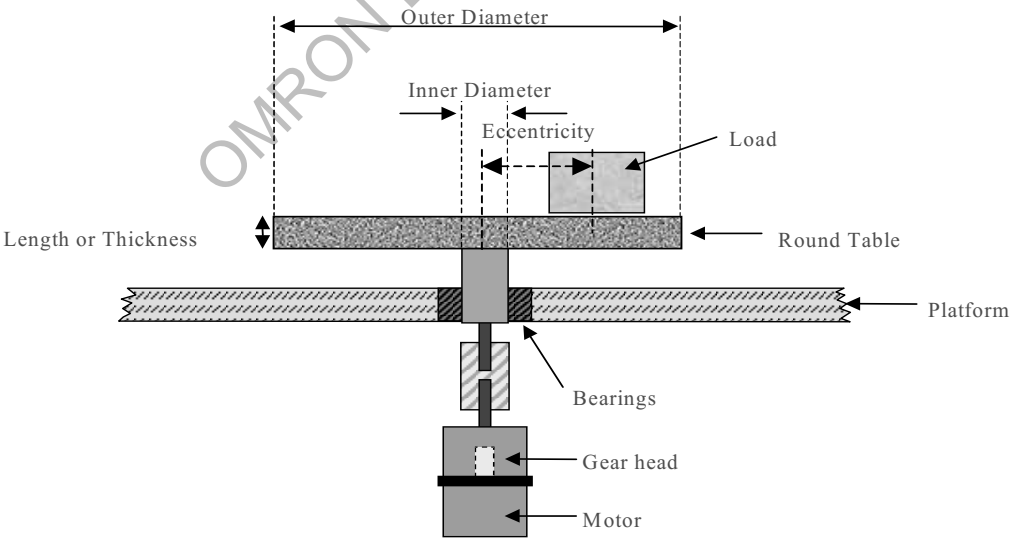
2.) **ระบบขับเคลื่อนแบบเฟืองสะพาน** (RACK & PINION)



3.) **ระบบขับเคลื่อนแบบสายพาน** (BELT DRIVE)



4.) **ระบบขับเคลื่อนแบบจานหมุน** (ROTARY TABLE)



OMRON ELECTRONICS CO., LTD.

จากรูปแบบการทำงานหรือการเคลื่อนที่ทางกลทั้ง 4 แบบ การหาขนาดของเซอร์โวมอเตอร์สามารถหาได้จากการคำนวณซึ่งต้องทราบค่าพารามิเตอร์ต่างๆ เช่น ถ้าเป็นระบบขับเคลื่อนแบบบอลสกูร เราต้องทราบ ระยะทางในการเคลื่อนที่ , ความยาวของบอลสกูร, ระยะพิชของบอลสกูร, ประสิทธิภาพของบอลสกูร , สัมประสิทธิ์แรงเสียดทานของบอลสกูร , ความเร็วในการเคลื่อนที่สูงสุด ,น้ำหนักของโหลด , เวลาในการทำงานต่อรอบ, และน้ำหนักรวมของแกนอื่นที่ตกลงบนแกนที่สนใจ จะเห็นได้ว่ามีค่าพารามิเตอร์ที่ใช้ในการคำนวณค่อนข้างมาก ซึ่งอาจจะไม่สะดวกสำหรับผู้ใช้งาน

ปัจจุบันผู้ผลิตเซอร์โวมอตอร์จะมีซอฟต์แวร์เพื่อใช้ในการคำนวณหาขนาดของเซอร์โวมอเตอร์เพื่ออำนวยความสะดวกกับผู้ใช้งาน ค่าพารามิเตอร์ต่างๆจะถูกป้อนเข้าเครื่องคอมพิวเตอร์เพื่อจำลองลักษณะการทำงานเหมือนสถานการณ์จริง หลังจากการประมวลผลซอฟต์แวร์จะระบุขนาดของเซอร์โวมอเตอร์และเซอร์โวไดเวอร์โดยอัตโนมัติ

ดังนั้นหน้าที่ของผู้ใช้งานคือการหาข้อมูลและป้อนค่าพารามิเตอร์ต่างๆให้ถูกต้อง เพื่อที่ซอฟต์แวร์จะได้คำนวณหาขนาดของเซอร์โวมอเตอร์อย่างถูกต้องมากที่สุด โดยปกติแล้วค่าที่ได้จากการคำนวณด้วยซอฟต์แวร์จะเป็นค่าที่รวม Safety Factor ไว้แล้ว

****

OMRON ELECTRONICS CO., LTD.

3. การเลือกคอนโทรลเลอร์

การควบคุมเซอร์โวมอเตอร์ให้เคลื่อนที่ มีองค์ประกอบหลัก 3 ส่วนที่สำคัญ คือ คอนโทรลเลอร์ ไดร์เวอร์และเซอร์โวมอเตอร์ เมื่อเราหาขนาดของเซอร์โวมอเตอร์และเซอร์โวไดร์เวอร์ได้แล้ว ส่วนต่อไปที่เราต้องพิจารณา คือ ชนิดของคอนโทรลเลอร์ ซึ่งมีหน้าที่สร้างสัญญาณลักษณะต่างๆให้กับไดร์เวอร์เพื่อส่งผ่านไปควบคุมเซอร์โวมอเตอร์ การพิจารณาเลือกคอนโทรลเลอร์ใช้งานจะต้องเลือกให้สอดคล้องกับชนิดของสัญญาณที่ไดร์เวอร์สามารถรับได้ ปกติแล้วชนิดของคอนโทรลเลอร์จะมีอยู่ 2 แบบ (ไม่รวมไดร์เวอร์ที่มีคอนโทรลเลอร์ในตัว) คือ

1.) คอนโทรลเลอร์ที่ให้สัญญาณพัลส์เอาต์พุต

2.) คอนโทรลเลอร์ที่ให้สัญญาณอนาลอกเอาต์พุต

ไม่ว่าจะเป็นคอนโทรลเลอร์แบบใดก็ตาม การพิจารณาเลือกใช้งาน ขึ้นอยู่กับวัตถุประสงค์ในการควบคุม เช่น ถ้าต้องการควบคุมตำแหน่งและความเร็ว เราควรเลือกใช้คอนโทรลเลอร์แบบที่ให้สัญญาณพัลส์เอาต์พุต แต่ถ้าเราต้องการควบคุมความเร็วและควบคุมแรงบิดก็ต้องเลือกใช้งานคอนโทรลเลอร์ที่ให้สัญญาณแบบอนาลอกเอาต์พุต

คอนโทรลเลอร์ที่ใช้สำหรับควบคุมเซอร์โวมอเตอร์ในงานอุตสาหกรรม ส่วนใหญ่แล้วมักนิยมใช้ พีแอลซี (PLC) มากกว่าคอนโทรลเลอร์ชนิดอื่น เนื่องจากพีแอลซีถูกออกแบบให้มีลักษณะเฉพาะสำหรับงานอุตสาหกรรม สามารถทนต่อการเปลี่ยนแปลงของสภาพแวดล้อมและมีความยืดหยุ่นดีกว่าคอนโทรลเลอร์ชนิดอื่น

OMRON ELECTRONICS CO., LTD.

หลักการพิจารณาเลือกคอนโทรลเลอร์ที่เป็นพีแอลซี

1.) จำนวนพัลส์เอาต์พุต (Transistor Output) จำนวนของพัลส์เอาต์พุตขึ้นอยู่กับผู้ผลิต เช่น ในพีแอลซีบางรุ่นอาจมีให้ 2 พัลส์เอาต์พุตหรือ 4 พัลส์เอาต์พุต ซึ่งหมายถึงจำนวนของเซอร์โวมอเตอร์ที่เราสามารถควบคุมได้ด้วยคอนโทรลเลอร์หนึ่งตัว

2.) ความเร็ว โดยปกติแล้วจะบอกมาเป็นหน่วยของความถี่(Hz.) การพิจารณาเลือกใช้ความถี่สูงหรือความถี่ต่ำขึ้นอยู่กับความละเอียดและความซับซ้อนของงาน งานลักษณะที่ต้องการควบคุมความเร็วสูงก็ต้องเลือกพัลส์เอาต์พุตประเภท High Speed เช่น 1 MHz. เป็นต้น

3.) ถ้าระบบต้องการควบคุมความเร็วและควบคุมแรงบิดต้องเลือกคอนโทรลเลอร์ชนิดที่ให้สัญญาณอนาลอกเอาต์พุต

****

4. เทคนิคการควบคุมเซอร์โวมอเตอร์อย่างง่าย

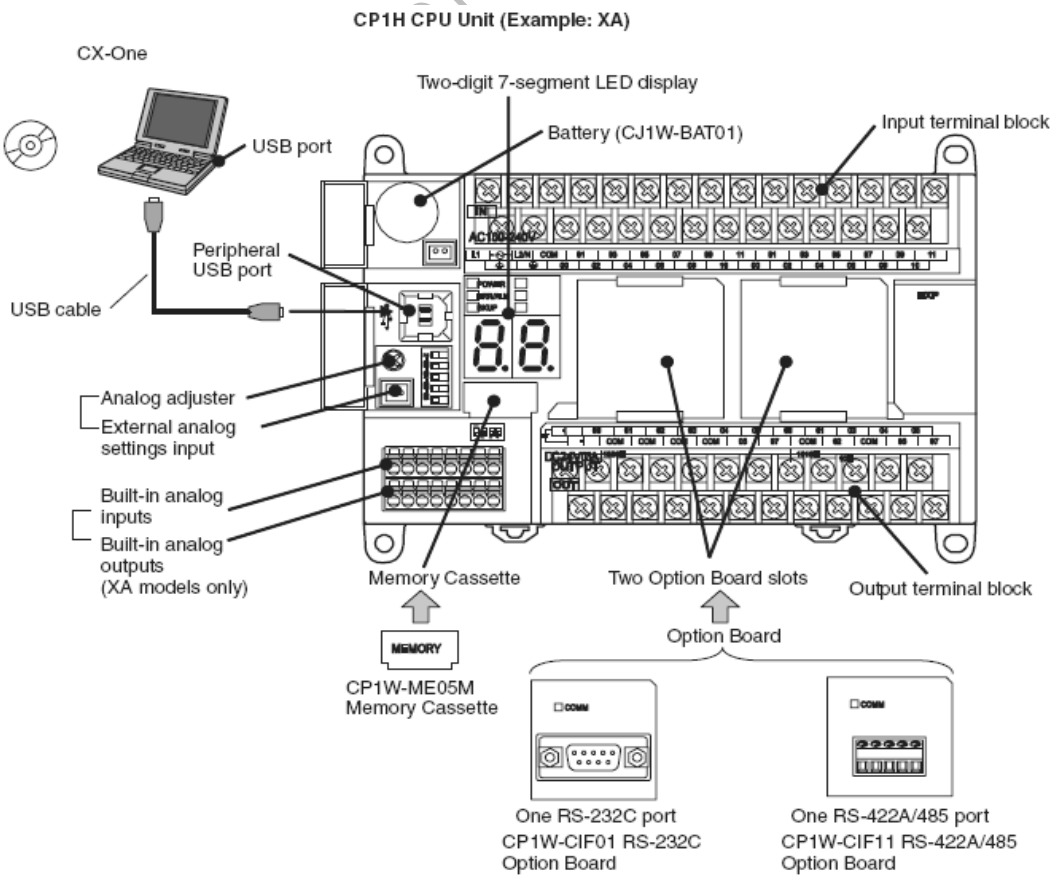
หลังจากที่ได้รู้จักกับเซอร์โวมอเตอร์มาพอสมควรแล้ว ต่อไปเราจะมาเรียนรู้วิธีการทำให้เซอร์โวมอเตอร์สามารถเคลื่อนที่ไปตามระยะทางและความเร็วที่เราต้องการ เป็นที่ทราบกันดีแล้วว่าระบบควบคุมเซอร์โวมอเตอร์ต้องประกอบด้วย 3 ส่วนสำคัญคือ คอนโทรลเลอร์, ไดร์เวอร์, และเซอร์โวมอเตอร์จึงจะสามารถทำงานได้ ในหัวข้อนี้จะเป็นตัวอย่างการนำเซอร์โวมอเตอร์มาใช้สำหรับงานควบคุมตำแหน่งและควบคุมความเร็ว โดยเลือกใช้คอนโทรลเลอร์เป็นพีแอลซีรุ่น CP1H-X/XA , ไดร์เวอร์และเซอร์โวมอเตอร์รุ่น SmartStep ขนาด 100 W.

ก่อนที่จะศึกษาการเขียนโปรแกรมควบคุมตามตัวอย่าง เราต้องทำความเข้าใจความหมายของขั้วต่อต่างๆของพีแอลซีและไดร์เวอร์ให้ถ่องแท้เสียก่อนเนื่องจากมีความสำคัญ หากต่อวงจรผิดอาจทำให้ระบบไม่ทำงานหรือเกิดการลัดวงจรขึ้นได้

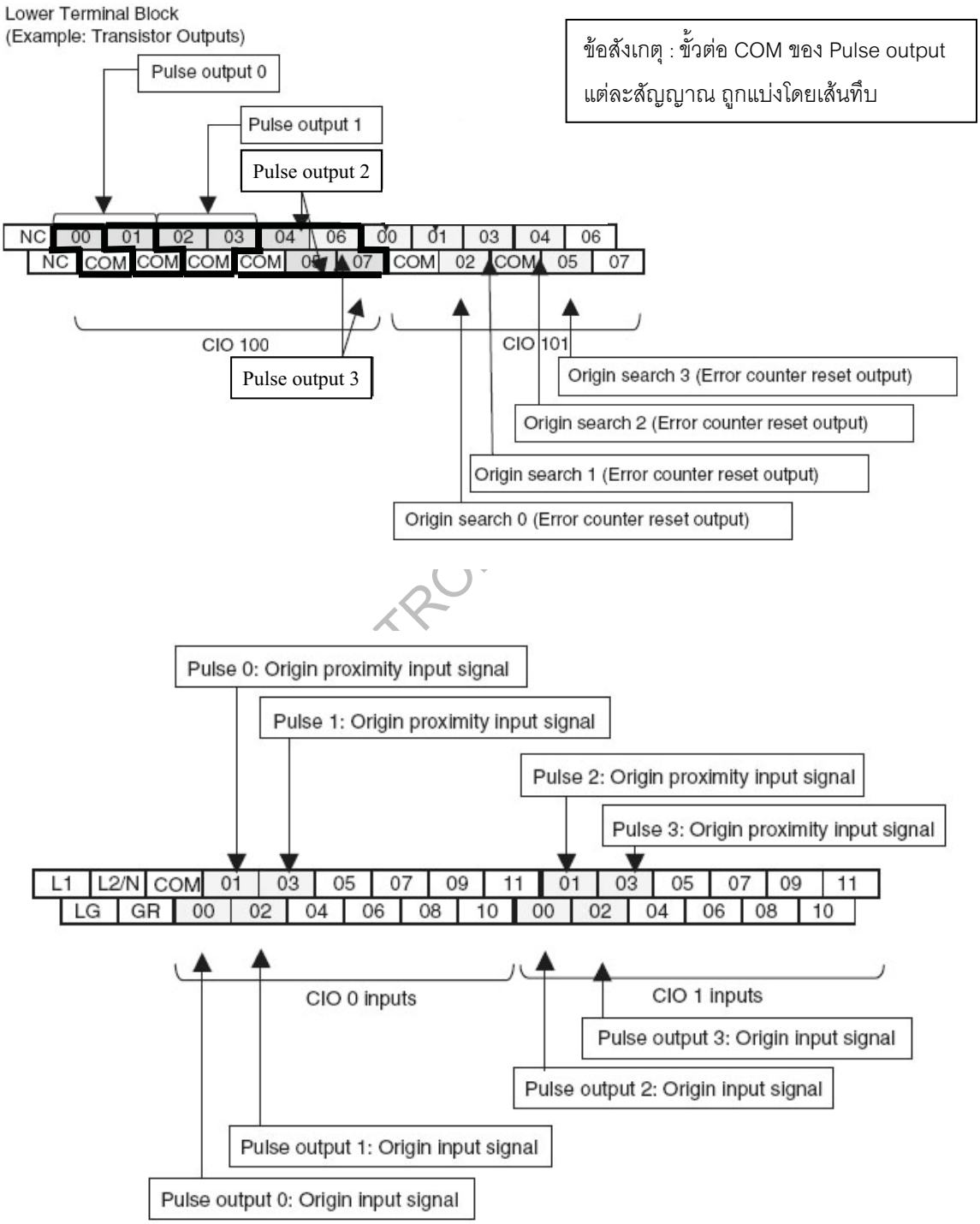
TRONICS CO., LTD.

1.) คอนโทรลเลอร์(PLC รุ่น CP1H-X/XA)

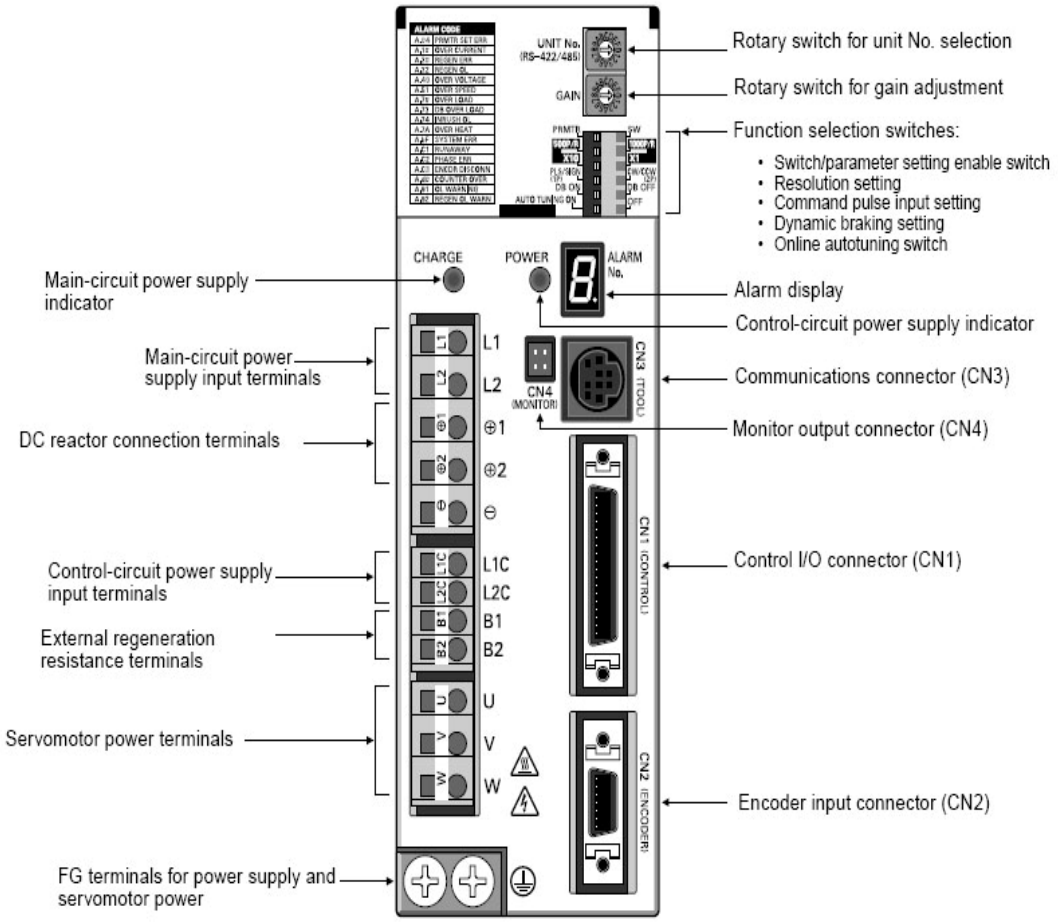
PLC รุ่น CP1H สามารถควบคุมเซอร์โวมอเตอร์ได้มากถึง 4 แกน ที่ความถี่ 100 KHz (Pulse output 0, Pulse output 1, Pulse output 2 , Pulse output 3)



รูปแสดงการต่อสัญญาณอินพุตและเอาท์พุตของ CP1H-X/XA



2.) **ไดร์เวอร์**(R7D-AP01H)



สัญญาณระหว่าง PLC กับไดร์เวอร์จะอยู่ที่พอร์ต CN1 เราเรียกว่า Control I/O Signal ซึ่งแบ่งเป็นสัญญาณควบคุมอินพุตและสัญญาณควบคุมเอาต์พุต

2.1) ความหมายของสัญญาณอินพุต (CN1 Control Inputs)

ขา	สัญญาณ	หน้าที่	คำอธิบาย
1	+ PULS/CW/A	Feed pulses, reverse pulses, or 90_ phase difference pulses (phase A)	รับสัญญาณพัลส์จาก PLC เพื่อให้เซอร์โวมอเตอร์เคลื่อนที่ไปตำแหน่งที่ต้องการ ลักษณะของสัญญาณแบ่งออกเป็น 3 แบบ คือ 1. Feed Pulse / Pulse Direction 2. ทิศทาง CW (ตามเข็ม) / CCW (ทวนเข็ม) 3. สัญญาณ A / B ที่ต่างเฟสกัน 90 องศา
2	- PULS/CW/A		
3	+ SIGN/CCW/B	Direction signal, forward pulses, or 90_ phase difference pulses (phase B)	
4	- SIGN/CCW/B		
5	+ ECRST	เคลียร์ค่าผิดพลาดของ counter ภายในไดรเวอร์	เปิดเพื่อหยุดคำสั่งและเคลียร์ค่าผิดพลาดสะสมในไดรเวอร์ (ต้องรับสัญญาณอย่างน้อย 20 µs)
6	- ECRST		
13	+ 24 Vin	รับไฟเลี้ยง 24 VDC	เป็นขั้วไฟ 24 VDC สำหรับขาอินพุต (ขา14, 18)
14	RUN	รอรับสัญญาณ Run	เพื่อให้เซอร์โวมอเตอร์อยู่ในสถานะพร้อมทำงาน
18	RESET	เคลียร์ Alarm	เพื่อเคลียร์ alarm ในไดรเวอร์

2.2) ความหมายของสัญญาณเอาต์พุต (CN1 Control Outputs)

ขา	สัญญาณ	หน้าที่	คำอธิบาย
32	Z	เอาท์พุตเฟส Z	สัญญาณเฟส z จากเอ็นโค้ดเดอร์ ส่งออกไปให้ PLC เพื่อช่วยในการหาตำแหน่ง Home
33	ZCOM		
34	ALM	เอาท์พุต Alarm	ให้สถานะ off เมื่อเซอร์โวไดรเวอร์เกิด Alarm
35	ALMCOM		
7	BKIR	เอาท์พุต Break Interlock	ส่งสัญญาณออกมาตัดวงจรเบรค ตอนเกิด Alarm
8	INP	บอกการเข้าถึงตำแหน่ง	ให้สถานะ on เมื่อมอเตอร์เข้าถึงตำแหน่งเป้าหมายแล้ว
10	OGND	ขั้ว Ground	ขั้ว Ground สำหรับขาเอาท์พุต (ขา7, 8)

OMRON ELECTRONICS CO., LTD.

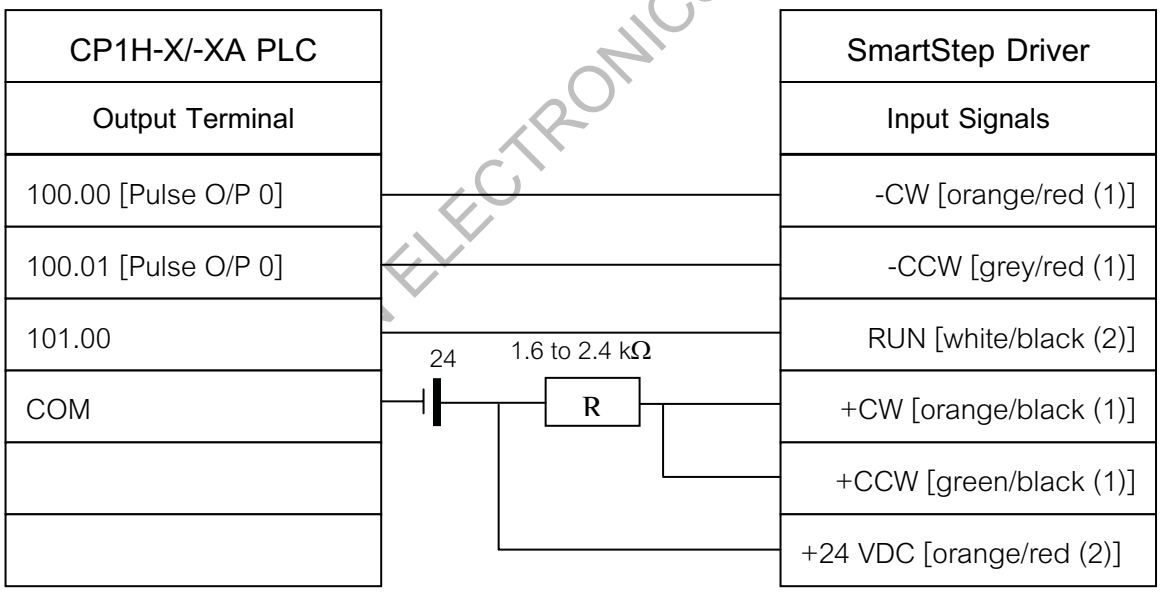
2.3) การต่อสัญญาณระหว่างพีแอลซีกับไดร์เวอร์ สำหรับการเคลื่อนที่แบบ Relative

จากรูปข้างล่างเป็นการต่อสายสัญญาณระหว่างพีแอลซีกับเซอร์โวไดร์เวอร์อย่างง่ายๆ เหมาะสำหรับงานที่ต้องการสั่งให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Relative Position หรือ Incremental

การเคลื่อนที่แบบ Relative Position เป็นอย่างไร?

การเคลื่อนแบบ Relative Position เป็นลักษณะการเคลื่อนที่ ที่ไม่จำเป็นต้องกำหนดตำแหน่งเริ่มต้น (Home Position) มักใช้กับงานที่ต้องการสั่งให้เซอร์โวมอเตอร์เคลื่อนที่ไปข้างหน้าให้ได้ระยะทางตามต้องการ ลักษณะของเครื่องจักรประเภทที่ใช้คำสั่ง Relative Position เช่น เครื่องฟีด , เครื่องตัดถุงพลาสติก , เครื่องตัดกระดาษ เป็นต้น

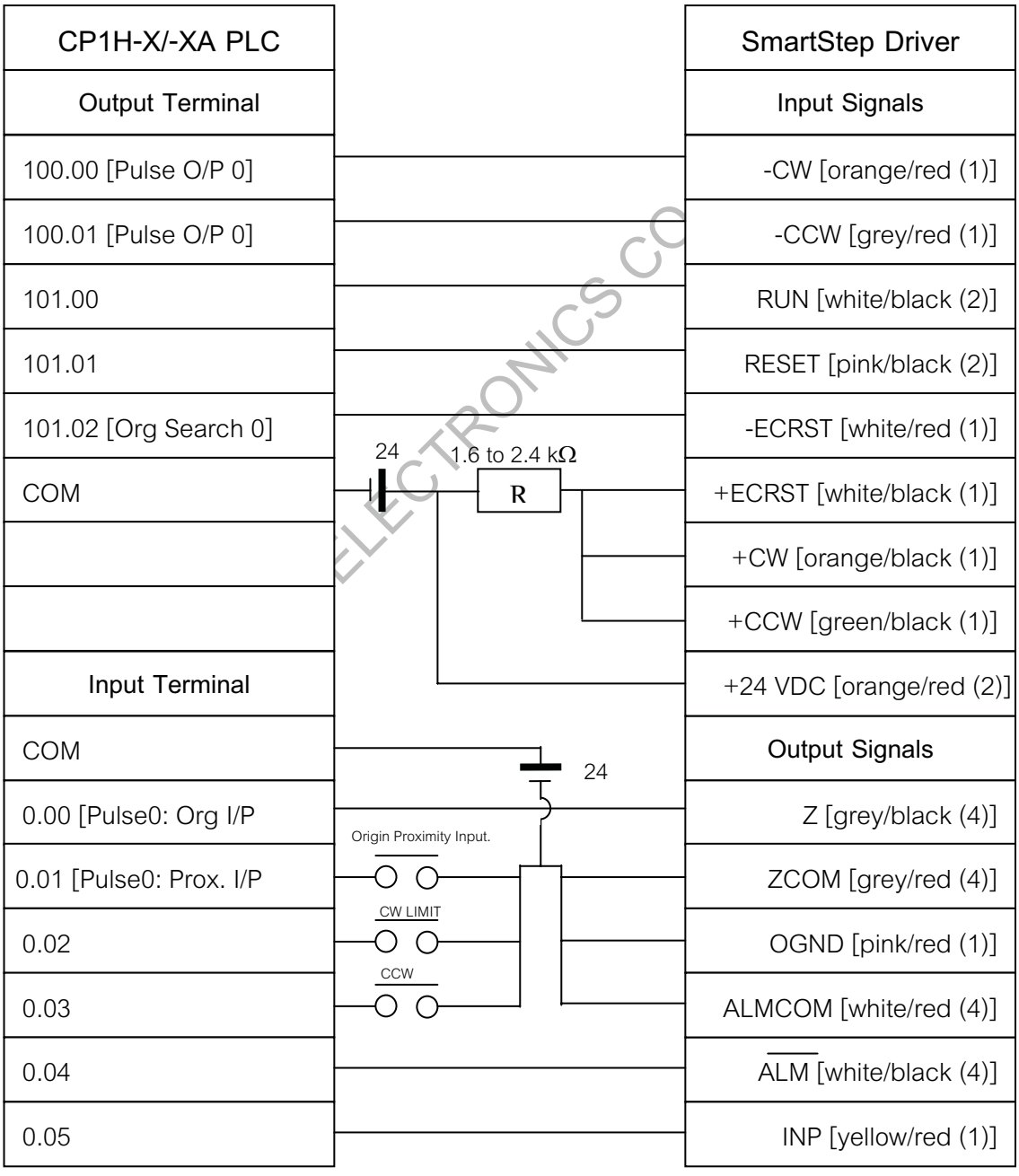
ELECTRONICS CO., LTD.



รูปแสดง การต่อสัญญาณระหว่างพีแอลซีกับไดร์เวอร์แบบ Relative Position

2.4) **การต่อสัญญาณระหว่างพีแอลซีกับไดร์เวอร์ สำหรับการเคลื่อนที่แบบ** Absolute

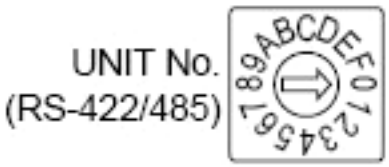
จากรูปข้างล่างเป็นการต่อสายสัญญาณระหว่างพีแอลซีและเซอร์โวไดร์เวอร์ สำหรับการเคลื่อนที่ ที่ต้องมีจุดเริ่มต้นของการเคลื่อนที่ (Home Position) สังเกตเห็นว่าจะมีการต่อสัญญาณ Phase Z, Origin Proximity Sensor รวมถึงการต่อ Limit Switch (CW&CCW Limit) ซึ่งมีไว้สำหรับป้องกันการเคลื่อนที่เลยขอบเขตที่กำหนด ลักษณะการเคลื่อนที่แบบ Absolute Position จะต้องทำคำสั่ง Origin search ในครั้งแรกก่อน แล้วจึงจะสามารถใช้คำสั่งการเคลื่อนที่แบบ Absolute ได้



รูปแสดงการต่อสัญญาณระหว่างพีแอลซีกับไดร์เวอร์แบบ Absolute Position

2.5) ความหมายของโรตารีสวิทซ์(ROTARY SWITCH) **และดิฟสวิทซ์**(DIP SWITCH) **ของไดร์เวอร์**

2.5.1) โรตารีสวิทซ์สำหรับตั้งค่า Unit No.

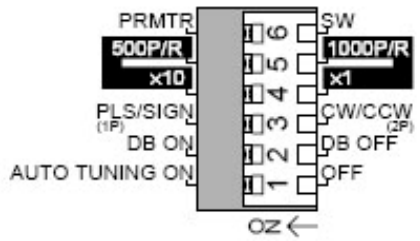


โรตารีสวิทซ์นี้ใช้สำหรับตั้งค่าUnit no. กรณีที่เราใช้การติดต่อสื่อสารแบบRS-422/485 เมื่อต้องการแลกเปลี่ยนค่าพารามิเตอร์ของไดร์เวอร์หลายๆแกนกับซอฟต์แวร์บนคอมพิวเตอร์สามารถเลือกตั้งค่าได้ระหว่าง 1 ถึง F (ห้ามตั้งค่าซ้ำกัน) สายสัญญาณจะรวมอยู่ในพอร์ท CN1

2.5.2) โรตารีสวิทซ์สำหรับตั้งค่า Gain

โรตารีสวิตซ์นี้มีไว้สำหรับปรับความไวผลตอบสนองของเซอร์โวมอเตอร์ ยิ่งตั้งค่ามากจะทำให้เซอร์โวมอเตอร์ตอบสนองได้เร็วขึ้น แต่ก็จะทำให้การเคลื่อนที่ราบรื่นลดลง ตรงกันข้ามกับการตั้งค่าน้อยๆ เซอร์โวมอเตอร์จะสามารถเคลื่อนที่ได้ราบรื่นกว่าแต่ตอบสนองช้าลง

2.5.3) ดิฟสวิทซ์



OMRON ELECTRONICS CO., LTD.

ความหมายของดิฟสวิทซ์ (Dip Switch)

ดิฟสวิทซ์ 6 :

ใช้เลือกการตั้งค่าระหว่างการใช้พารามิเตอร์ภายใน หรือตั้งค่าจากดิฟสวิทซ์ 1 ถึง 5 ถ้า ON ดิฟสวิทซ์ 6 หมายถึงการตั้งค่าพารามิเตอร์ภายในเซอร์โวไดรเวอร์ผ่านทางโปรแกรม CX-Drive หรือ Keypad

ดิฟสวิตซ์ 5 และ 4 :

ใช้ในการปรับตั้งค่าความละเอียดของการหมุนของเซอร์โวมอเตอร์โดยกำหนดเป็นจำนวนพัลส์ต่อการหมุนของเซอร์โวมอเตอร์ 1 รอบ

สวิตซ์	**ค่าความละเอียด**	
5	4	
OFF	OFF	1,000 pulse/revolution (0.36º/step)
OFF	ON	10,000 pulse/revolution (0.036º/step)
ON	OFF	500 pulse/revolution (0.72º/step)

ดิฟสวิทซ์ 3: เลือกประเภทของสัญญาณพัลส์

ดิฟสวิทซ์ 2: ตั้งค่าการใช้งาน Dynamic Brake.

ดิฟสวิทซ์ 1: ทำ Online Auto Tuning

OMRON ELECTRONICS CO., LTD.

****

<u>ตัวอย่างที่ 1</u> การสั่งให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Relative Position

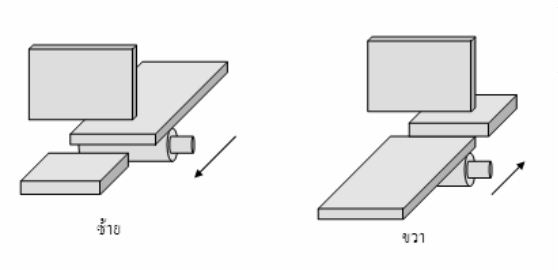
วัตถุประสงค์

1.) เข้าใจการต่อวงจรควบคุมระหว่างคอนโทรลเลอร์(PLC)กับไดร์เวอร์สำหรับการเคลื่อนที่แบบ Relative Position

2.) สามารถเขียนโปรแกรมควบคุมให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Relative Position

โจทย์จำลองลักษณะงานจริง

ต้องการตัดกระดาษที่มีความยาวสม่ำเสมอที่ 50 มม. โดยใช้เซอร์โวมอเตอร์ควบคุมโรลเลอร์ในการป้อนกระดาษเข้าสู่เครื่องตัด ด้วยความเร็วที่คงที่ สามารถเลือกทิศทางการป้อนชิ้นงานได้ทั้งซ้ายและขวา



ในตัวอย่างนี้เราจะจำลองสถานการณ์โดยใช้บอลสกูรแทนโรลเลอร์ป้อนกระดาษ โดยสมมติให้การเคลื่อนที่ของ Table ที่ติดตั้งอยู่บนบอลสกูรเปรียบเหมือนกับการป้อนกระดาษ ทุกครั้งที่เรากดสวิทช์ SW.1ให้ Table เลื่อนไปทางซ้าย 50 มม. และทุกครั้งที่กดสวิทช์ SW.2 ให้ Table เลื่อนไปทางขวา 50 มม.

OMRON ELECTRO , LTD.

ปฏิบัติตามขั้นตอน

1.) เตรียมอุปกรณ์

1.1) Controller ใช้ PLC รุ่น CP1H-X40DT (Open collector)

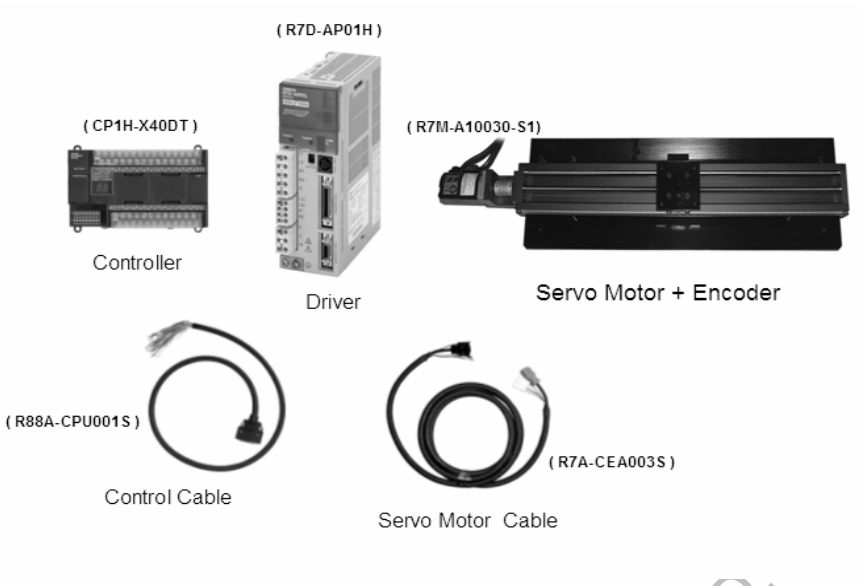
1.2) Driver ใช้รุ่น Smart Step R7D-AP01H

1.3) Servomotor+ Encoder ติดตั้งอยู่บน Ball Screw (R7M-AP01H)

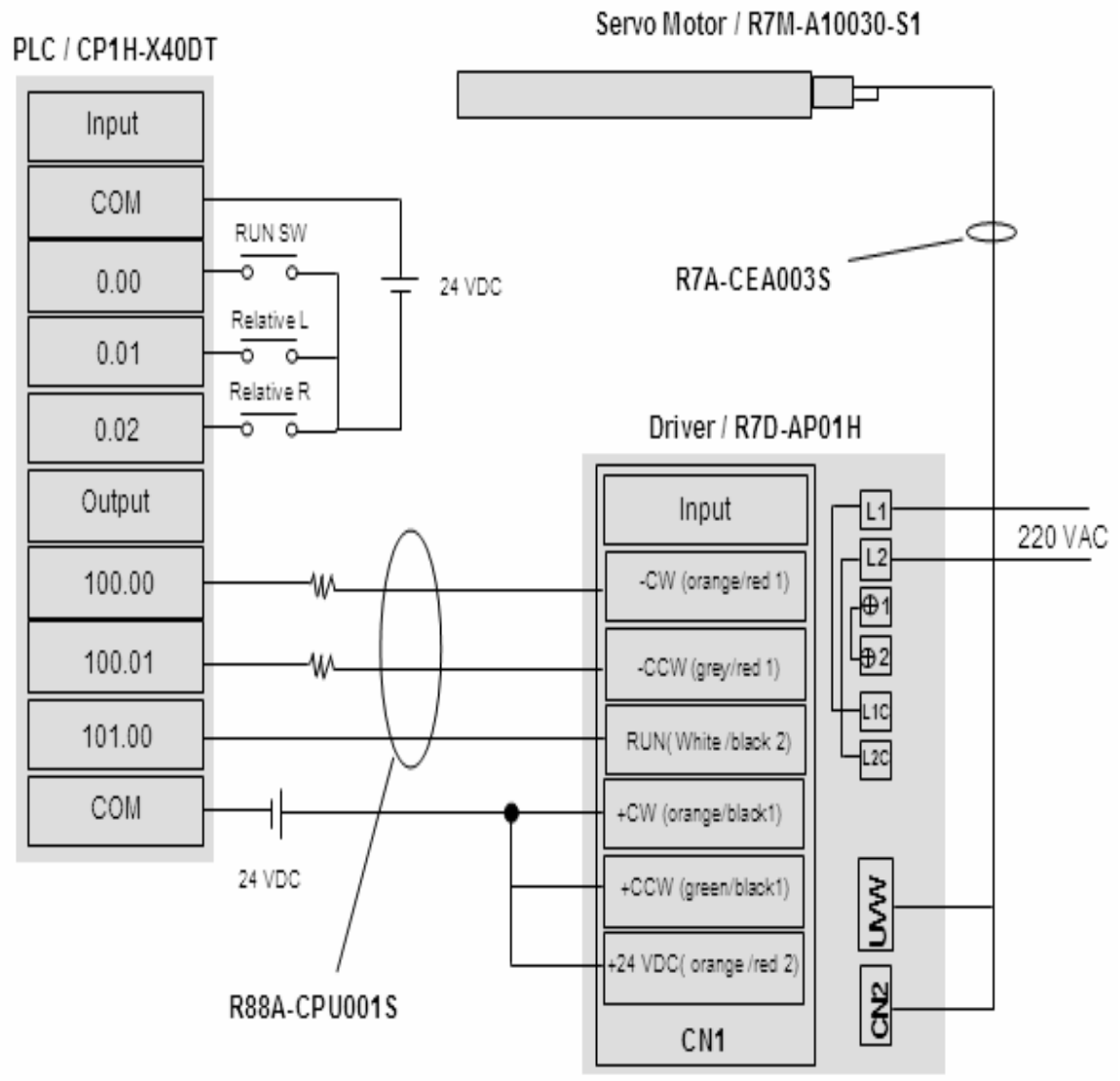
1.4) Control Cable ใช้เชื่อมต่อระหว่าง PLC กับ Driver (R88A-CPU001S)

1.5) Servo Motor Cable ใช้เชื่อมต่อระหว่างเซอร์โวมอเตอร์กับ Driver (R7A-CEA003S)

1.6) สายต่อวงจรและสวิทช์ ON/OFF



2.) **เดินสายตามวงจร**



3.) กำหนดความละเอียดในการหมุนของเซอร์โวมอเตอร์ต่อรอบ

ทำไมต้องกำหนดความละเอียดในการหมุนของเซอร์โวมอเตอร์?

การที่เซอร์โวมอเตอร์สามารถเคลื่อนที่ได้เกิดจากสัญญาณพัลส์ที่ส่งมาจากตัวคอนโทรลเลอร์ (PLC) จำนวนพัลส์ที่ส่งมาจากตัวคอนโทรลเลอร์ยิ่งสูง ความละเอียดในการเคลื่อนที่ของเซอร์โวมอเตอร์ก็จะยิ่งสูงตามไปด้วย(ขึ้นอยู่กับความละเอียดของชิ้นส่วนทางกลด้วย) เช่น สัญญาณจากคอนโทรลเลอร์จำนวน10,000 พัลส์จะให้ความละเอียดมากกว่า 5,000 พัลส์ เป็นต้น

จากโจทย์เราต้องการให้กระดาษเคลื่อนที่เป็นระยะทางเท่ากับ 50 มม. (Table เคลื่อนที่) ซึ่งในตัวอย่างนี้เราจำลองการเคลื่อนที่ของกระดาษด้วยบอลสกูร สิ่งแรกที่เราต้องทราบ คือ ถ้ามอเตอร์หมุนหนึ่งรอบจะทำให้ Table เคลื่อนที่ไปเป็นระยะทางเท่าไร เราสามารถคำนวณหาได้จากระยะพิช (Pitch) ของบอลสกูร ระยะพิชของบอลสกูรที่ใช้ในตัวอย่างนี้มีค่าเท่ากับ 2.5 มม. หมายความว่ามอเตอร์หมุน 1 รอบจะทำให้ Table เคลื่อนที่ไปเท่ากับ 2.5 มม. ดังนั้นถ้าเราต้องการให้ Table เคลื่อนที่ไปเท่ากับ 50 มม. แสดงว่ามอเตอร์ต้องหมุนเท่ากับ 20 รอบ (20 รอบ * 2.5 ระยะพิช = 50 มม. ระยะทาง)

สมมติว่าในตัวอย่างนี้เราต้องการกำหนดความละเอียดในการหมุนของเซอร์โวมอเตอร์ต่อรอบเท่ากับ 10,000 พัลส์ หมายความว่าเซอร์โวมอเตอร์ได้รับสัญญาณพัลส์จำนวน 10,000 พัลส์จากคอนโทรลเลอร์ทำให้เซอร์โวมอเตอร์หมุนไปเท่ากับ 1 รอบ และเคลื่อนที่ไปด้วยระยะทางเท่ากับ 2.5 มม.(10,000 **พัลส์** = 1 **รอบ** = 2.5 **มม**.)

ถ้าต้องการให้เซอร์โวมอเตอร์เคลื่อนที่ไปเท่ากับ50มม.ต้องป้อนสัญญาณพัลส์เท่ากับ 200,000 พัลส์ (10,000/2.5 * 50 = 200,000 **พัลส์**)

OMRON ELECTRONICS CO., LTD.

การกำหนดความละเอียดในการหมุนของเซอร์โวมอเตอร์ต่อรอบทำได้อย่างไร?

การกำหนดความละเอียดในการหมุนของเซอร์โวมอเตอร์ต่อรอบสามารถกำหนดได้โดยการตั้งค่าดิฟสวิทซ์เบอร์ 4, 5 ที่ตัวไดร์เวอร์ (แสดงตามขั้นตอนในหัวข้อ 3.1)

3.1) กำหนดความละเอียดในการหมุนของเซอร์โวมอเตอร์ต่อรอบด้วยดิฟสวิทซ์

3.1.1) ตั้งค่าโรตารีสวิทซ์ตามตาราง

UNIT NO.	0
GAIN	4

3.1.2) ตั้งดิฟสวิทซ์ตามตารางเมื่อต้องการความละเอียดเท่ากับ 10,000 พัลส์

DIP 1	OFF
DIP 2	OFF
DIP 3	OFF
DIP 4	ON
DIP 5	OFF
DIP 6	OFF

ความหมายของดิฟสวิทซ์เบอร์ 4 และ 5

สวิตซ์	**ค่าความละเอียด**	
5	4	
OFF	OFF	1,000 pulse/revolution (0.36º/step)
OFF	ON	10,000 pulse/revolution (0.036º/step)
ON	OFF	500 pulse/revolution (0.72º/step)

OMRON ELECTRONICS CO., LTD.

<u>หมายเหตุ</u> การกำหนดความละเอียดโดยใช้ดิฟสวิทซ์จะมีข้อจำกัดตรงที่ผู้ใช้งานไม่สามารถกำหนดความละเอียดได้เองตามความต้องการ ต้องเลือกตามความละเอียดที่กำหนดมาให้ เช่น 500,1000,5000 **หรือ** 10,000 **พัลส์ต่อรอบ**

ถ้าต้องการตั้งความละเอียดของจำนวนพัลส์ต่อรอบตามความต้องการสามารถทำได้หรือไม่?

ความละเอียดของจำนวนพัลส์ต่อรอบผู้ใช้งานสามารถกำหนดได้เองตามความต้องการโดยใช้วิธีตั้งค่า Gear Ratio

Gear Ratio **คืออะไร?**

Gear Ratio คือ การกำหนดความละเอียดในการเคลื่อนที่ของเซอร์โวมอเตอร์ ซึ่งเป็นการกำหนดความสัมพันธ์ระหว่างสัญญาณป้อนกลับที่รับมาจากเอ็นโค้ดเดอร์กับจำนวนพัลส์ที่ส่งมาจากคอนโทรลเลอร์(PLC) ทำให้เราสามารถทราบระยะทางการเคลื่อนที่ได้อย่างถูกต้องตรงกับค่าจริง

จำนวนพัลส์ของสัญญาณป้อนกลับคำนวณอย่างไร?

ระยะทางในการเคลื่อนที่ของ Table ที่ติดตั้งบนบอลสกูร สามารถทราบได้จากจำนวนพัลส์ที่เอ็นโค้ดเดอร์ส่งกลับมายังตัวไดร์เวอร์ จากคุณสมบัติของเอ็นโค้ดเดอร์ที่ใช้ในตัวอย่างนี้จะกำเนิดสัญญาณพัลส์เท่ากับ 2000 พัลส์ต่อการหมุนเซอร์โวมอเตอร์ 1 รอบ ดังนั้นถ้าเซอร์โวมอเตอร์หมุนเท่ากับ 20 รอบ ได้ระยะทาง 20 มม. จะทำให้เอ็นโค้ดเดอร์กำเนิดสัญญาณพัลส์เท่ากับ 40,000 พัลส์ (20 รอบ * 2,000 พัลส์)

ข้อควรจำ

ปกติแล้วไดร์เวอร์จะมองเห็นสัญญาณพัลส์จากเอ็นโค้ดเดอร์เป็น 4 เท่า จึงเท่ากับ 40,000 * 4 = 160,000 พัลส์ (ที่การหมุน 20 รอบได้ระยะทางเท่ากับ 50 มม.)

ในตัวอย่างนี้ต้องการส่งสัญญาณจำนวน10,000 พัลส์จากคอนโทรลเลอร์(PLC) เพื่อให้ Table เคลื่อนที่เป็นระยะทางเท่ากับ 50 มม. จะต้องกำหนด Gear Ratio เท่ากับเท่าไรสามารถคำนวณได้ดังนี้

จากรูปข้างล่าง แสดงความสัมพันธ์ระหว่างสัญญาณป้อนกลับจากเอ็นโค้ดเดอร์กับสัญญาณพัลส์จากคอนโทรลเลอร์ ซึ่งไดร์เวอร์จะมองเห็นจำนวนพัลส์จากเอ็นโค้ดเดอร์มีค่าเท่ากับ 160,000 พัลส์ (40000*4) เมื่อเซอร์โวมอเตอร์เคลื่อนที่ไปได้ระยะทาง 50 มม. เราจะกำหนดให้เป็น G1

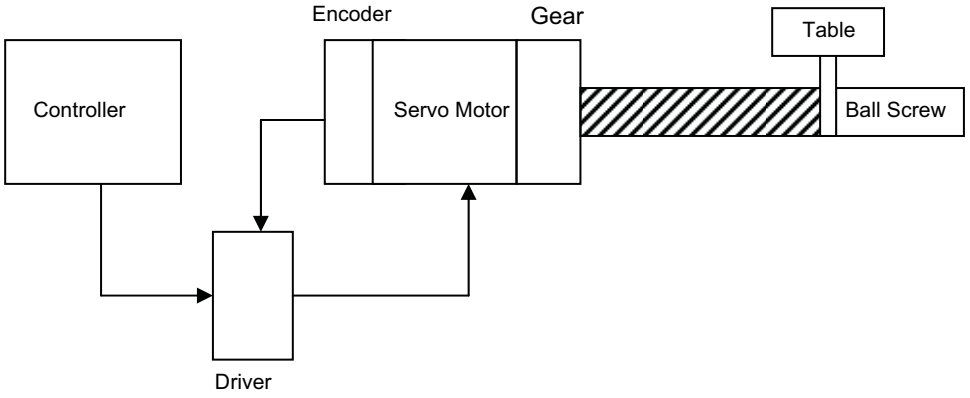
และสัญญาณพัลส์ที่รับมาจากคอนโทรลเลอร์เท่ากับ 10,000 พัลส์ (กำหนดเองจากซอฟต์แวร์) เพื่อทำให้เซอร์โวมอเตอร์เคลื่อนที่ไปเป็นระยะทาง 50 มม. เราจะกำหนดให้เป็น G2

Gain Ratio = G1/G2

= 160,000/10,000

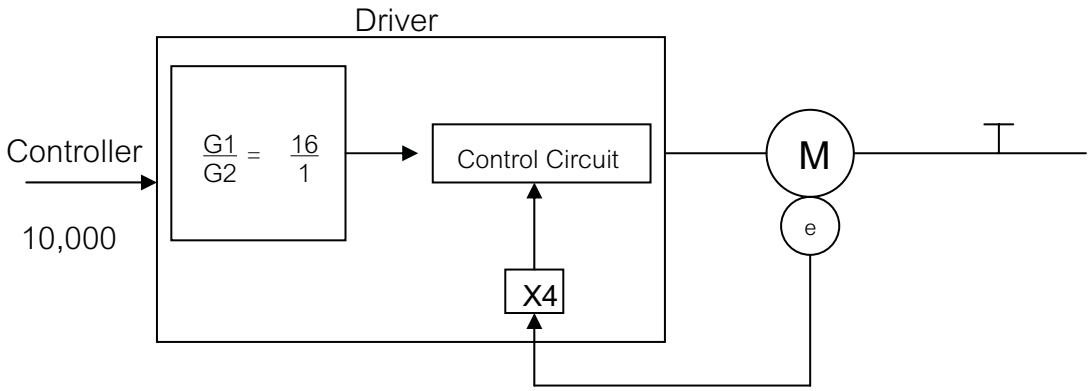
= 16/1

G2 = 10,000 พัลส์ ที่ระยะ 50 มม.



G1 = 160,000

OMRON ELECTRONICS CO., LTD.



รูปแสดงการกำหนด Gear Ratio

การตั้งค่า Gear Ratio ทำได้อย่างไร?

การกำหนดความละเอียดโดยใช้ Gear Ratio จะต้องใช้ซอฟต์แวร์ CX-Drive ในการกำหนดค่าเท่านั้น (แสดงตามขั้นตอนในหัวข้อ 3.2)

CX-Drive คืออะไร?

CX-Drive เป็นซอฟต์แวร์ที่ใช้สำหรับกำหนดค่าพารามิเตอร์ต่างๆของอินเวอร์เตอร์หรือไดร์เวอร์ของออมรอนอยู่ใน Package Software CX-One

3.2 การกำหนด Gain Ratio โดยใช้ซอฟต์แวร์ CX-Drive

3.2.1) ตั้งค่าโรตารีสวิทซ์ตามตาราง

UNIT NO.	0
GAIN	4

OMRON ELECTRONICS CO., LTD.

3.2.2) ตั้งค่าดิฟสวิทซ์ตามตาราง

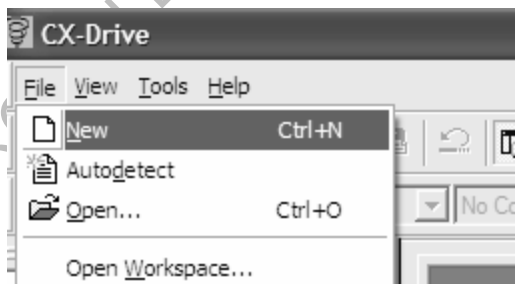
(On ดิฟสวิทซ์เบอร์ 6 เพราะเราต้องการกำหนดพารามิเตอร์ผ่านซอฟต์แวร์)

DIP 1	OFF
DIP 2	OFF
DIP 3	OFF
DIP 4	OFF
DIP 5	OFF
DIP 6	ON

3.2.3) ต่อเซอร์โวไดร์เวอร์กับคอมพิวเตอร์ที่ติดตั้งซอฟต์แวร์ CX-one

3.2.4) เปิด CX-one เลือกโปรแกรม CX-Drive

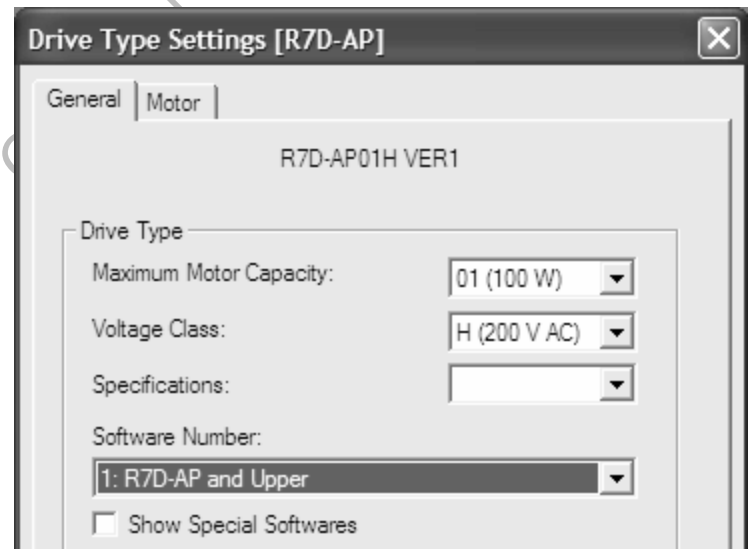
3.2.5) เลือก File /New



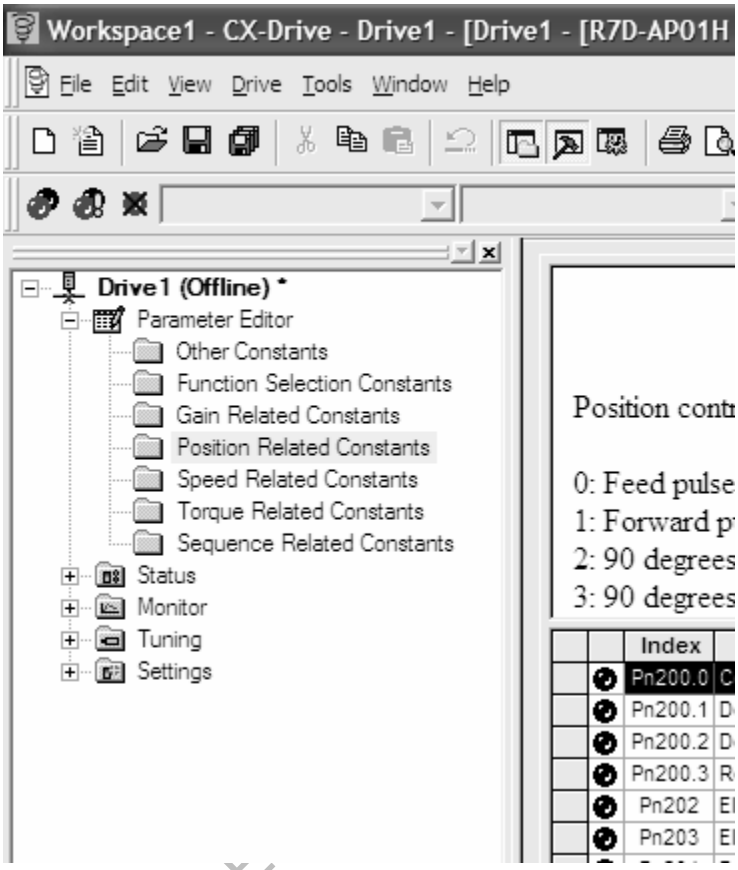
OMRON ELECTRONICS CO., LTD.

3.2.6) ป้อนค่าตามรูป / เลือก Settings (Drive Type)

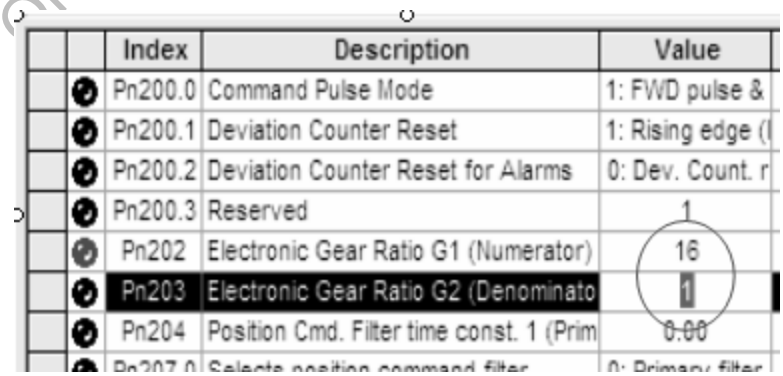
3.2.7) ป้อนค่าตามรูป / เลือก O.K. / O.K.



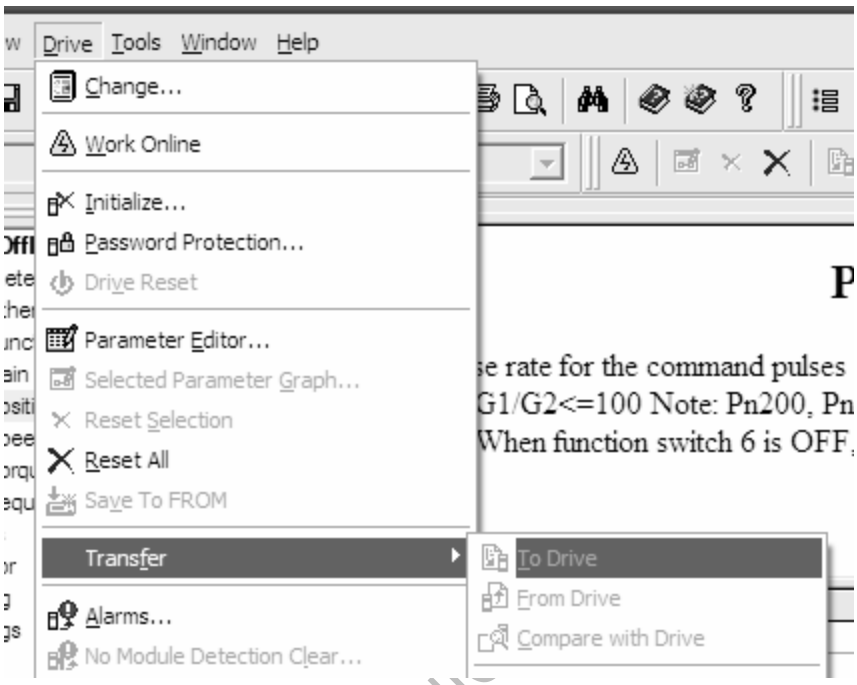
3.2.8) Double Click ที่ Drive1 /Parameter Edit /Position Related Constants



3.2.9) เปลี่ยนค่าใน Pn202 = 16 , Pn203 =1



3.2.10) เลือก Drive/Transfer/To Drive



3.2.11) ปิด Power ที่จ่ายให้กับเซอร์โวไดเวอร์แล้วเปิดใหม่เพื่อ Update ข้อมูล

<u>หมายเหตุ</u> ในตัวอย่างนี้สมมติว่าเราเลือกใช้การตั้งค่าความละเอียดผ่านทางซอฟต์แวร์ (ปฏิบัติตามหัวข้อ 3.2)

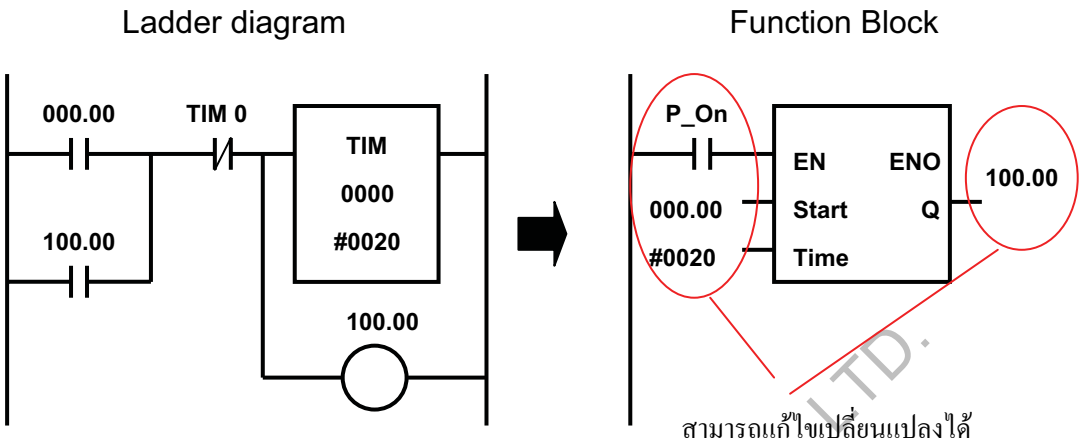
OMRON ELECTRONICS

4.) เขียนโปรแกรมควบคุมเซอร์โวมอเตอร์ที่พีแอลซีโดยใช้คำสั่ง Function Block

Function Block (FB) คือ อะไร?

Function Block เป็นโปรแกรมหรือฟังก์ชันที่ถูกสร้างและรวบรวมไว้เป็นคำสั่งเดียว สำหรับใช้ในการเขียนแลดเดอร์ไดอะแกรม (Ladder Diagram) โดยมีหน้าสัมผัสที่ทำหน้าที่เป็นตัวเริ่มการทำงาน อินพุต(Input)และเอาต์พุต (Output) สามารถแก้ไขข้อมูลผ่านทางค่าพารามิเตอร์ที่กำหนดภายใน Function Block

จากรูปเป็นตัวอย่าง Function Block ของวงจรนับเวลาใน Ladder Diagram ซึ่งเราสามารถกำหนดและแก้ไขค่าพารามิเตอร์ต่างๆในคำสั่งไทม์เมอร์(Timer) ที่อยู่ในรูปแบบของ Function Block ได้



รูปแสดงตัวอย่าง Function Block ของวงจรนับเวลา

OMRON ELECTRONICS CO., LTD.

OMRON FB Library คืออะไร ?

Omron FB Library เป็นการรวบรวมไฟล์ต่างๆของฟังก์ชันบล็อกที่ถูกสร้างไว้โดยบริษัทออมรอน ซึ่งไฟล์เหล่านี้มีไว้เพื่อช่วยในการเขียนโปรแกรมให้ง่ายขึ้นประกอบด้วยฟังก์ชันมาตรฐานต่างๆ เพื่ออำนวยความสะดวกให้กับผู้เขียนโปรแกรมที่ต้องการติดต่อระหว่าง PLC กับอุปกรณ์ต่างๆของออมรอน

OMRON FB Library มีประโยชน์อย่างไร?

ประโยชน์ของ Omron FB Library

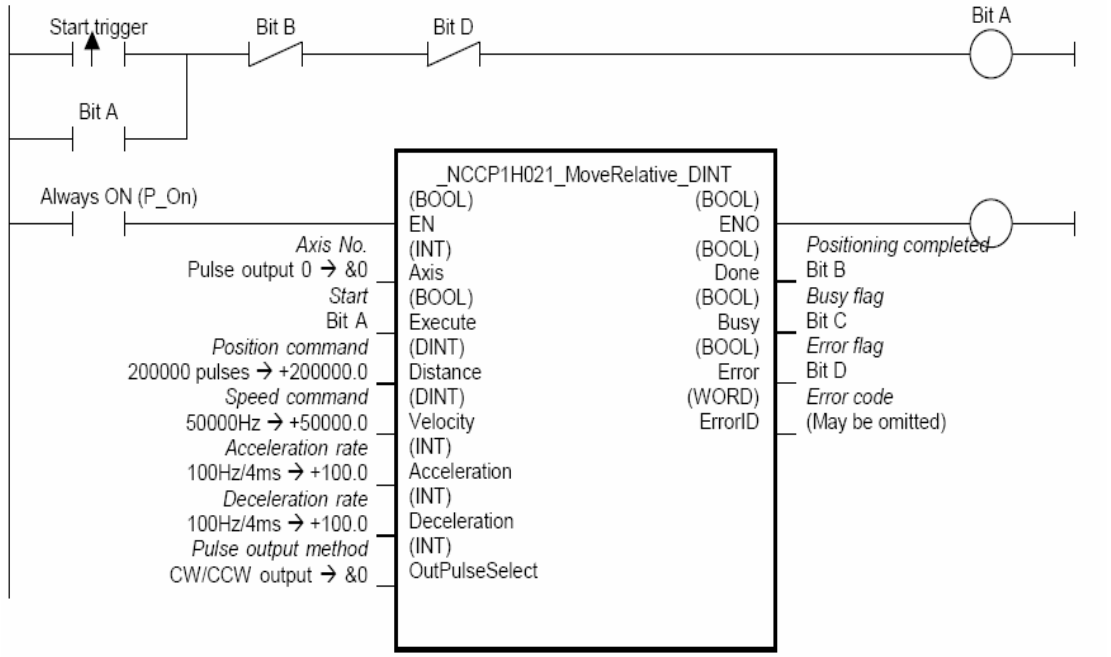
1.) ไม่จำเป็นต้องเขียน Ladder Diagram ในการใช้งานฟังก์ชันพื้นฐานระหว่าง PLC กับอุปกรณ์ภายนอกทำให้ลดเวลาในการเขียนโปรแกรม
2.) ใช้งานง่ายแค่โหลดไฟล์ Function Block มาเก็บไว้ที่โปรแกรมแล้วเรียกใช้งาน
3.) ไม่จำเป็นต้องทดสอบโปรแกรมเพราะผ่านการทดสอบมาแล้วทำให้ลดเวลาในการค้นหาจุดบกพร่องของโปรแกรม
4.) ง่ายต่อความเข้าใจเพราะ Function Block จะมีชื่อที่แสดงอย่างชัดเจน ในแต่ละรูปแบบการทำงานสามารถตั้งชื่อให้สอดคล้องกับกระบวนการได้ ในขณะที่กำลังทำงานสามารถที่จะตรวจสอบค่าได้ทันทีเพราะพารามิเตอร์ทั้งหมดจะถูกแสดงค่าไว้ช่วยลดเวลาในการแก้ไขปรับปรุง

คำสั่ง Move Relative

ในตัวอย่างนี้เราจะใช้คำสั่ง Move Relative ซึ่งเป็นคำสั่งที่ทำให้เซอร์โวมอเตอร์แกนที่เราต้องการควบคุมให้เคลื่อนที่ตามตำแหน่ง , ความเร็ว , อัตราเร่ง และอัตราหน่วงที่กำหนด โดยจะทำงานต่อจากตำแหน่งเดิมที่มีการเคลื่อนที่ (สามารถโหลดคำสั่งนี้ได้จาก Function Block Library)

หลักการทำงานของคำสั่ง FB: Move Relative

เมื่อ Start Bit A จะทำให้ให้ฟังก์ชัน Move Relative ทำงาน เซอร์โวมอเตอร์จะหมุนตามค่าตัวแปรที่ตั้งไว้ ในขณะที่ Bit C จะทำงาน(ON) จนกระทั่งเซอร์โวมอเตอร์หมุนถึงตำแหน่งที่เรากำหนดไว้ เมื่อถึงตำแหน่งที่กำหนดแล้ว Bit B จะทำงาน (ON) เพื่อตัดการทำงานของ Bit A ในกรณีที่เกิดความผิดปกติขึ้น (Error) Bit D จะทำงาน(On) เพื่อตัดการทำงานของ Bit A เช่นกัน



ความหมายของตัวแปรอินพุต

Name	Variable name	Data type	Default	Range	Description
EN	En	BOOL			1(ON): ใช้งาน FB
					0(OFF) : ไม่ใช้งาน FB
Axis No.	Axis	INT	&0	&0 - &3	& 0: Pulse Output 0
					& 1: Pulse Output 1
					& 2: Pulse Output 2
					& 3: Pulse Output 3
Start	Execute	BOOL	0(OFF)		สั่งให้มอเตอร์เคลื่อนที่
Position				-2,147,483,648	ระบุระยะทางที่จะให้มอเตอร์
Command	Distance	DINT	(+0)	To	เคลื่อนที่ไป (pulse)
				2,147,483,647	
Speed				&1	
Command	Velocity	DINT	&1	To	ระบุความเร็วที่ต้องการ (Hz)
				&1000000	
Acceleration				(+1.0)	
rate	Acceleration	REAL	(+ 1.0)	to	ระบุอัตราเร่งที่ต้องการ
				(+65535.0)	(Hz/4ms)
Deceleration				(+1.0)	
rate	Deceleration	REAL	(+ 1.0)	to	ระบุอัตราหน่วงที่ต้องการ
				(+65535.0)	(Hz/4ms)
Pulse ouput	Output Pulse	INT	&0	&0 to &1	&0:CW/CCW Output
method	Select				&1:Pulse + Direction

OMRON

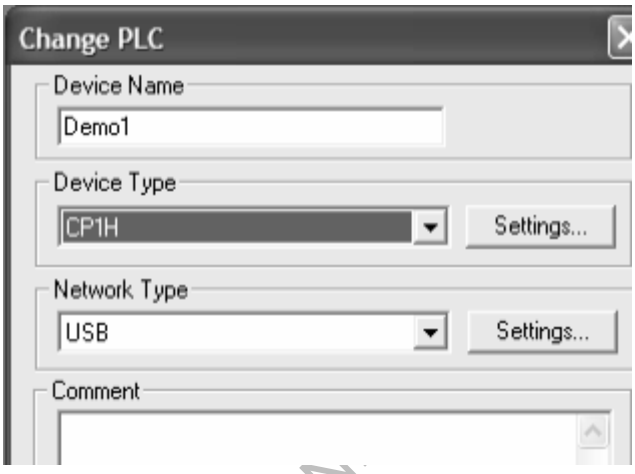
ความหมายของตัวแปรเอาท์พุต

Name	Variable name	Data type	Default	Range	Description
ENO	ENO	BOOL			1(ON):FB ทำงานปกติ
					0(OFF): FB ทำงานผิดปกติ
Positioning	Done	BOOL			1(ON):เมื่อมอเตอร์เคลื่อนที่
Completed					ไปยังตำแหน่งที่ระบุไว้แล้ว
Busy flag	Busy	BOOL			1(ON):เมื่ออยู่ในกระบวนการทำงาน
Error flag	Error	BOOL			1(ON):เมื่อเกิดความผิดพลาดในการทำงาน
Error Code	ErrorID	WORD			แสดง Error Code โดยสามารถเทียบได้
(May be omitted)					จากคู่มือการใช้งานหากนอกเหนือที่มี
					จะแสดงเป็น #0000

ขั้นตอนการเขียนโปรแกรมด้วย FB

4.1) เชื่อมต่อ PLC กับ คอมพิวเตอร์

4.2) เปิด CX-One เลือก CX-Programmer / File /New/ป้อนค่าตามรูป/Settings



OMRON ELECTRON

4.3) ป้อนค่า CPU Type / เลือก O.K. (ตามรุ่นของPLC ที่เลือกใช้งาน)

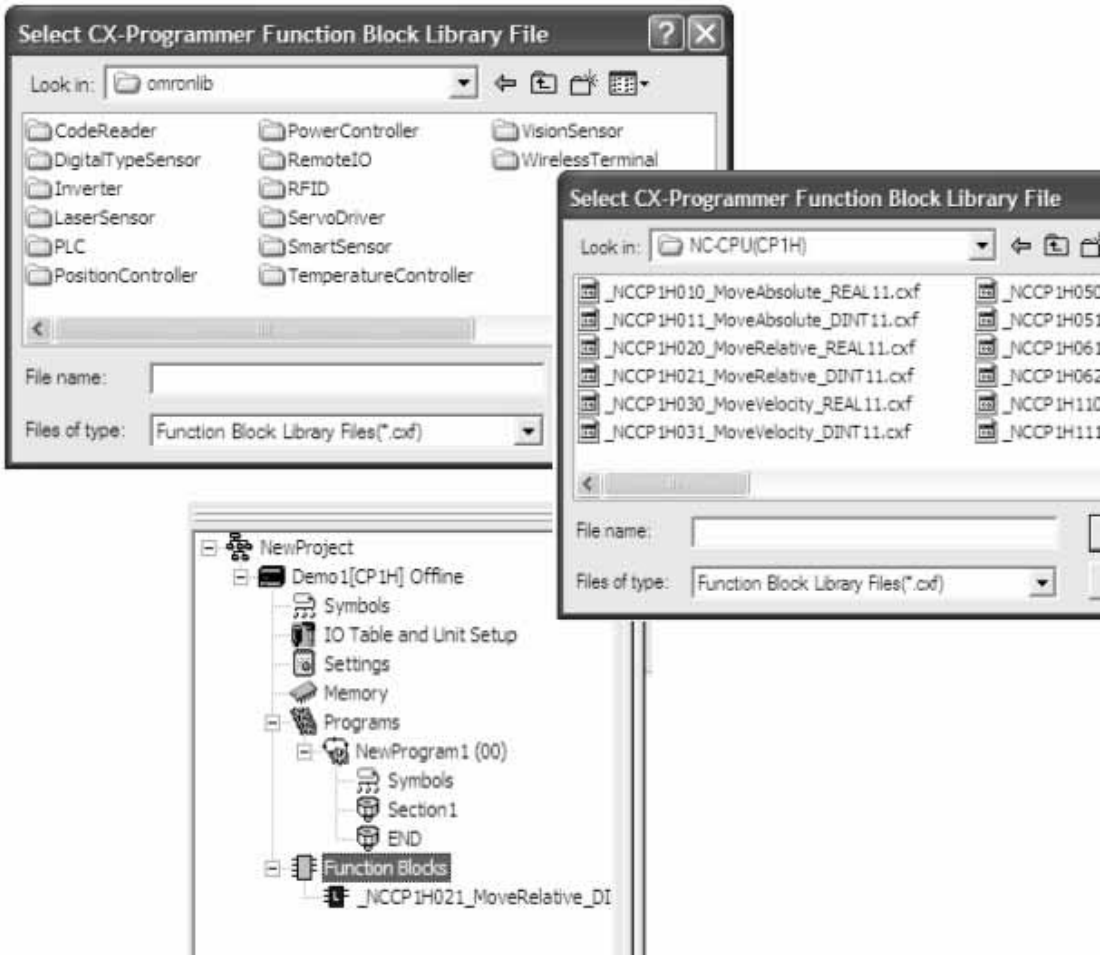
4.4) หลังจากเลือก O.K. แล้วหน้าจอจะปรากฏดังรูปซึ่งเป็นหน้าจอพร้อมใช้งาน

4.5) เริ่มต้นใช้งานโดยการคลิกขวาที่ไอคอน Function Block

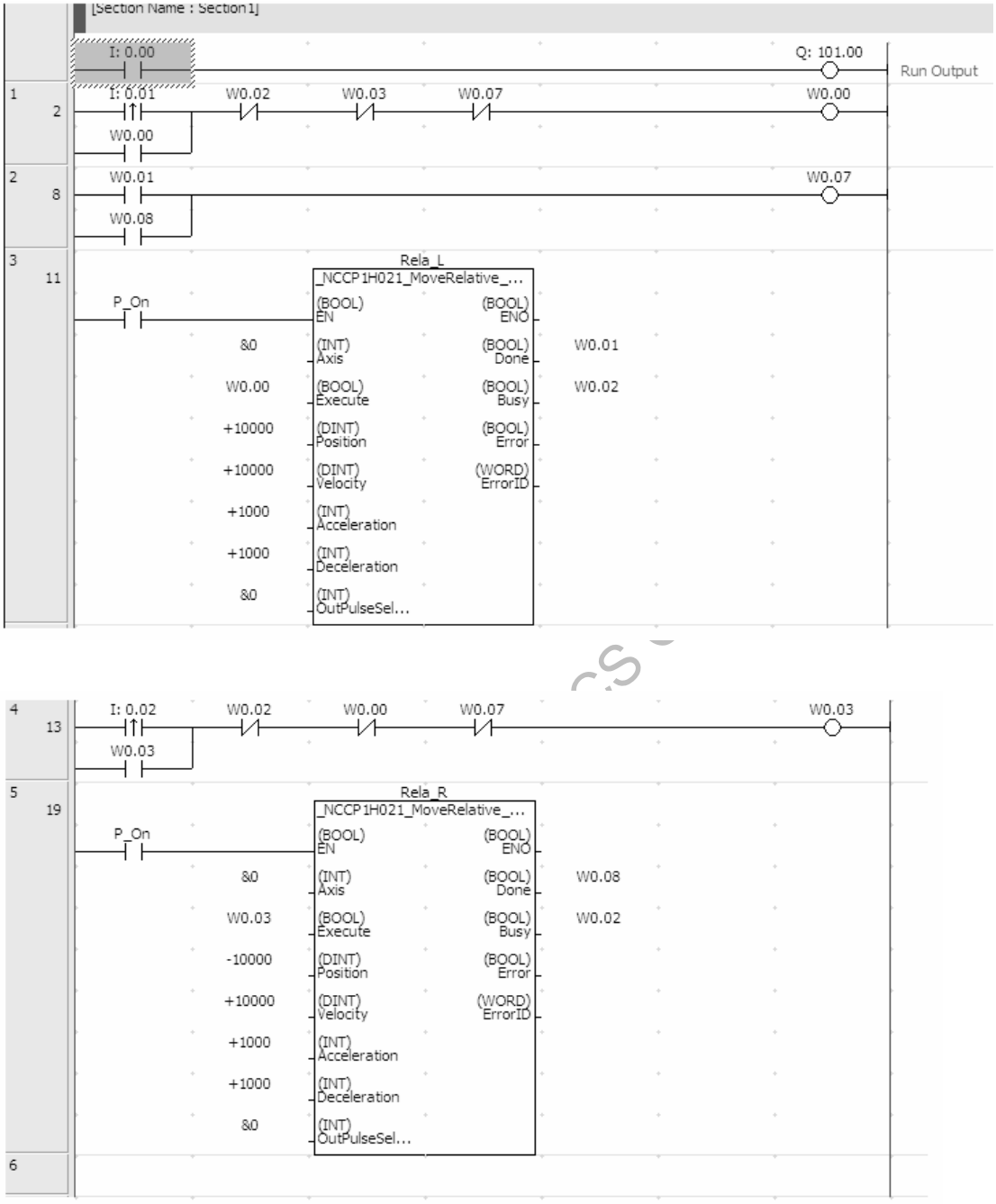
OMRON ELECTRONICS CO.

4.6) เลือก Insert Function Block / From file

4.7) เข้าไปที่โฟลเดอร์ FBL/omronlib /Position controller/ NC-CPU(CP1H)/_NCCP1H021_MoveRelative_DINT



4.8) เขียนโปรแกรมตามรูป



การทำงานของโปรแกรม

ฟังก์ชันบล็อกแรก เป็นการสั่งให้เซอร์โวมอเตอร์เคลื่อนที่ไปทางซ้าย จากวงจร Bit ที่สั่งให้ฟังก์ชันบล็อกทำงานคือ W0.00 ตำแหน่งในการเคลื่อนที่เราจะกำหนดที่ Position , ความเร็วในการเคลื่อนที่กำหนดที่ Velocity, อัตราเร่งและอัตราหน่วงกำหนดที่ Acceleration และ Deceleration เมื่อเซอร์โวมอเตอร์เคลื่อนที่ถึงตำแหน่งที่กำหนด Bit W0.01 จะทำงานเพื่อสั่งให้ Bit W0.07 ตัดการทำงานของฟังก์ชันบล็อกตัวแรก

ฟังก์ชันบล็อกตัวที่สอง เป็นการสั่งให้เซอร์โวมอเตอร์เคลื่อนที่ไปทางขวา จากวงจร Bit W0.03 ทำหน้าที่สั่งให้ฟังก์ชันบล็อกทำงานเมื่อเคลื่อนที่ถึงตำแหน่งที่กำหนดไว้แล้ว Bit W0.08 จะทำงานเพื่อสั่งให้ Bit W0.07 ตัดการทำงานของฟังก์ชันบล็อกตัวที่สอง

5.) ทดสอบโปรแกรม

จากโปรแกรมระยะทางในการเคลื่อนที่ มีค่าเท่ากับ +10000 และ –10000 ซึ่งหมายถึงจำนวนพัลส์เอาต์พุตที่ส่งออกไปเพื่อให้Table เคลื่อนที่ไปทางซ้ายและขวาเท่ากับ 50 มม.(เครื่องหมายบวกและลบคือทิศทางการเคลื่อนที่) เปรียบได้กับการป้อนกระดาษเข้าเครื่องตัดตามความยาวที่ต้องการทั้งซ้ายและขวาด้วยอัตราความเร็ว 10000 พัลส์ต่อวินาที

การทดสอบโปรแกรมหลังจากที่ดาวน์โหลดโปรแกรมลงในพีแอลซีแล้ว ให้เปลี่ยนโหมดการทำงานของพีแอลซีเป็นมอนิเตอร์(Monitor)หรือรัน(Run)โหมด หลังจากนั้นกด SW.0 เพื่อให้เซอร์โวมอเตอร์อยู่ในสถานะพร้อมใช้งาน (ล็อกโรเตอร์)

กด SW.1 Table จะเคลื่อนที่ไปทางซ้ายสังเกตความเร็วและระยะทางการเคลื่อนที่

กด SW.2 Table จะเคลื่อนที่ไปทางขวาสังเกตความเร็วและระยะทางการเคลื่อนที่

OMRON ELECTRONICS CO., LTD.

<u>หมายเหตุ</u>

กรณีที่ตั้งความละเอียดของจำนวนพัลส์ต่อรอบโดยใช้ดิฟสวิทซ์ ระยะทางการเคลื่อนที่(Position) ในคำสั่งฟังก์ชันบล็อกต้องมีค่าเท่ากับ 200,000 พัลส์จึงจะทำให้บอลสกูรเคลื่อนที่ไปเท่ากับ 50 มม. (10,000/2.5 * 50 = 200,000 พัลส์)

ในโปรแกรมเรากำหนดระยะทางการเคลื่อนที่เท่ากับ 10,000 พัลส์ สามารถทำให้บอลสกูรเคลื่อนที่ไปเป็นระยะทางเท่ากับ 50 มม. เพราะเรากำหนดความละเอียดด้วยวิธี Gear Ratio (10,000 * 16 = 160,0000 พัลส์)

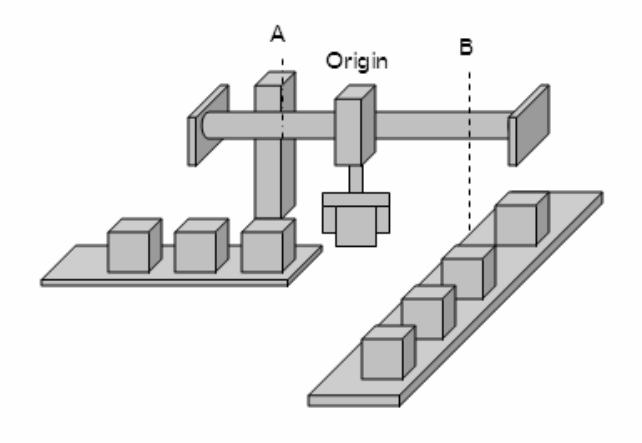
<u>ตัวอย่างที่ 2</u> การสั่งให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Absolute Position

วัตถุประสงค์

1.) เข้าใจการต่อวงจรควบคุมระหว่างคอนโทรลเลอร์(PLC)กับไดร์เวอร์สำหรับการเคลื่อนที่แบบ Absolute Position
2.) สามารถเขียนโปรแกรมควบคุมให้เซอร์โวมอเตอร์เคลื่อนที่แบบ Absolute Position

โจทย์จำลองลักษณะงานจริง

ต้องการให้แขนกลเคลื่อนที่ไปจับชิ้นงานที่ตำแหน่ง A เพื่อนำไปวางบนสายพานที่ตำแหน่ง B ในระบบต้องกำหนดให้มีจุดเริ่มต้น(Origin) และสามารถค้นหาจุดเริ่มต้นได้(Origin Search)



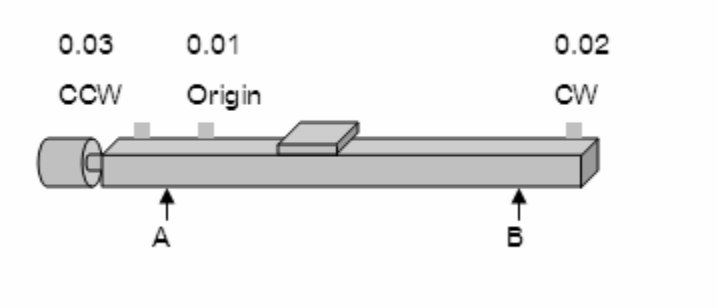
จากโจทย์เราจะจำลองการทำงานโดยใช้บอลสกูรแทนแขนกล

เงื่อนไข

1.) เมื่อกดสวิทช์ SW.1 ให้ Table เคลื่อนที่หาตำแหน่ง Origin

2.) เมื่อกดสวิทช์ SW.2 ให้ Table เคลื่อนไปที่จุด A (CCW) เปรียบเหมือนกับแขนกลเคลื่อนที่ไปจับชิ้นงาน

3.) เมื่อกดสวิทช์ SW.3 ให้ Table เคลื่อนที่ไปที่จุด B (CW) เปรียบเหมือนกับแขนกลเคลื่อนที่นำชิ้นงานไปวางที่ตำแหน่ง B



ONICS CO., LTD.

<u>ปฏิบัติตามขั้นตอน</u>

1.) เตรียมอุปกรณ์

1.1) Controller ใช้ PLC รุ่น CP1H-X40DT (Open collector)

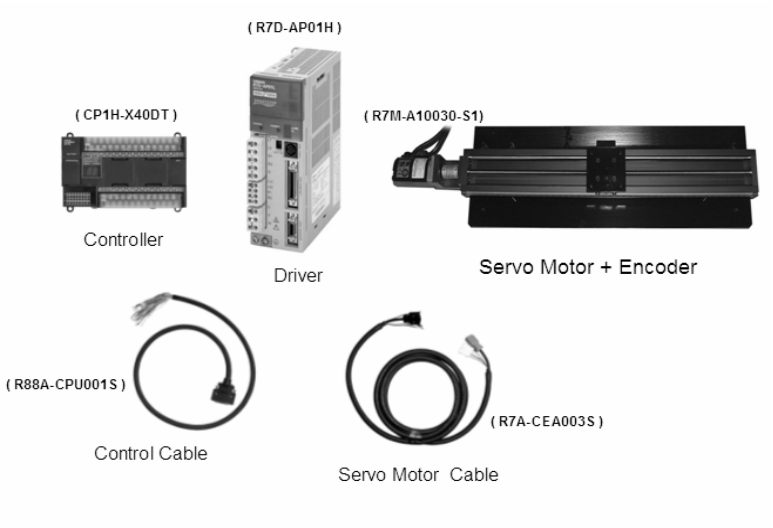
1.2) Driver ใช้รุ่น Smart Step R7D-AP01H

1.3) Servomotor+ Encoder ติดตั้งอยู่บน Ball Screw (R7M-AP01H)

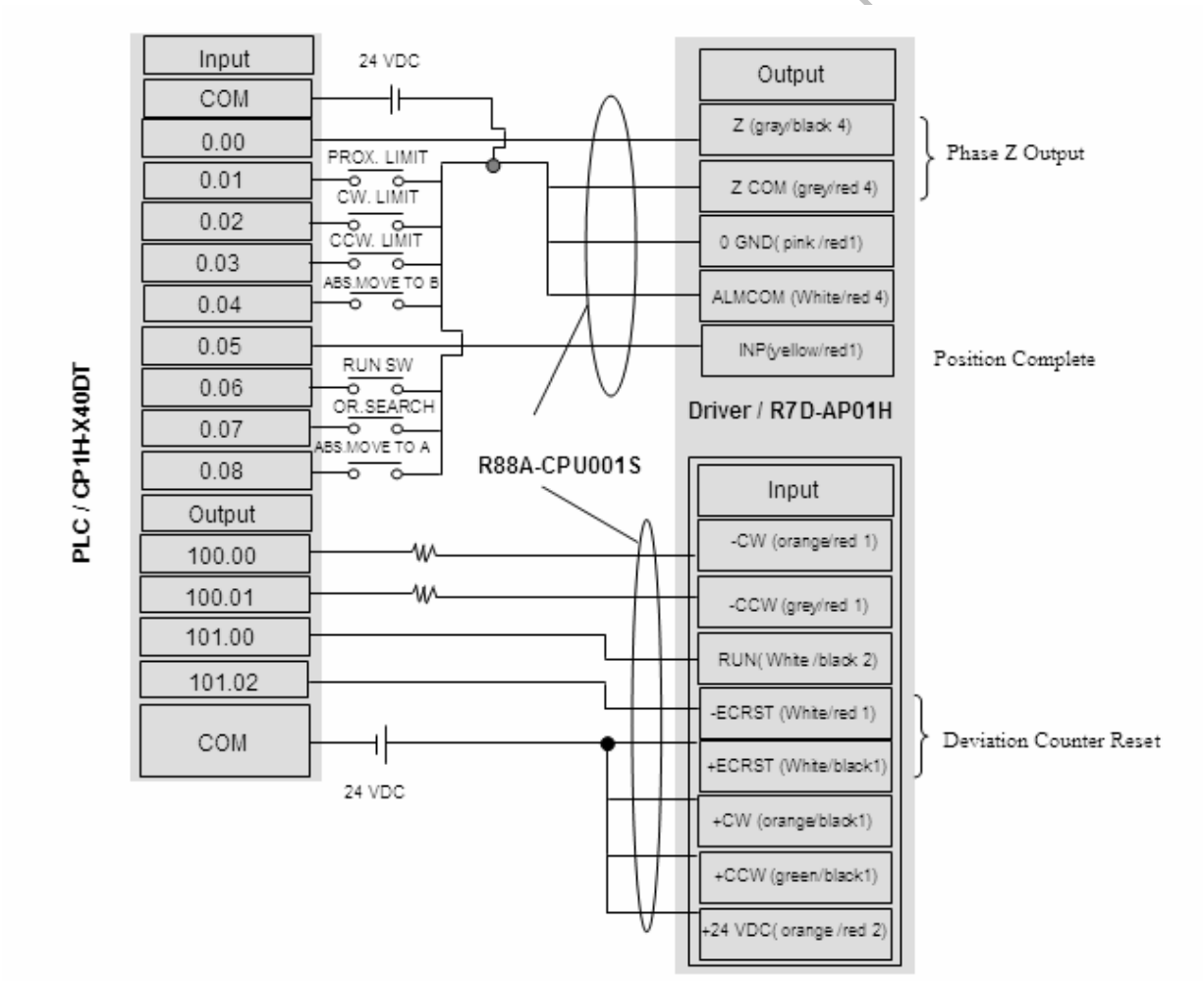
1.4) Control Cable ใช้เชื่อมต่อระหว่าง PLC กับ Driver (R88A-CPU001S)

1.5) Servo Motor Cable ใช้เชื่อมต่อระหว่างเซอร์โวมอเตอร์กับ Driver (R7A-CEA003S)

1.6) สายต่อวงจรและสวิทช์ ON/OFF



2.) เดินสายตามวงจร



หมายเหตุ : เซอร์โวมอเตอร์รุ่น Smart Step จำเป็นต้องต่อตัวความต้านทานเพิ่มเติมเพื่อจำกัดกระแสไม่ให้เกินกว่าค่าที่กำหนด (ปกติจะใช้ความต้านทานประมาณ 1.6 – 2.4 กิโลโอมห์

3.) กำหนดพารามิเตอร์ใน PLC Setting โดยใช้ซอฟต์แวร์ CX-Programmer

ขั้นตอน

3.1) เชื่อมต่อ PLC กับคอมพิวเตอร์ที่ติดตั้งซอฟต์แวร์ CX-Programmer

3.2) Double click ที่ Setting

ทำไมต้องกำหนดค่าในSetting?

เนื่องจากในตัวอย่างนี้เป็นการเคลื่อนที่ที่ต้องมีจุดเริ่มต้น (Home Position) จะสังเกตเห็นว่าในวงจรมีการต่อสัญญาณเฟส Z, Origin Proximity Sensor รวมถึงการต่อลิมิตสวิทช์ CW Limit และ CCW Limit (เพื่อป้องกันการเคลื่อนที่เลยขอบเขตที่กำหนด) เพิ่มเติมขึ้นมา ก่อนการเขียนโปรแกรมเราจะต้องกำหนดรายละเอียดของพัลส์เอาต์พุตที่พีแอลซีก่อน เพื่อกำหนดรูปแบบการค้นหาตำแหน่ง Origin โดยใช้ซอฟต์แวร์CX-Programmer ในการกำหนดค่าพารามิเตอร์ ซึ่งมีรายละเอียดที่ผู้ใช้งานต้องทำความเข้าใจดังนี้

ความหมายของพารามิเตอร์แต่ละตัวคืออะไร?

3.2.1) Bass Settings : การตั้งค่าพื้นฐาน

Undefined Origin : กำหนดใช้งานฟังก์ชัน Origin Search

- Hold : มีการใช้งานฟังก์ชัน Origin Search
- Undefined : ไม่มีการใช้งานฟังก์ชัน Origin Search

Limit Input Signal Operation: กำหนดใช้งานลิมิตสวิทช์ร่วมกับการทำงาน

- Search Only: ใช้งาน CW/CCW limit input เฉพาะเวลาทำ Origin Search เท่านั้น
- Always: ใช้งาน CW/CCW limit input ตลอดเวลาที่มีการใช้งานฟังก์ชัน Pulse Output

Limit Input Signal : ชนิดของลิมิตสวิทช์ที่ใช้งาน

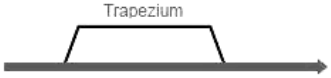
- NO: ปกติเปิด
- NC: ปกติปิด

Search/Return Initial Speed: ความเร็วเริ่มต้นของการทำ Origin search/return

- ให้กำหนดค่าที่ต้องการ (pps)

Speed Curve: ชนิดของการเคลื่อนที่ของเซอร์โวมอเตอร์

- Trapezium



- S – curve

OMRON ELECTRONICS CO., LTD.

3.2.2) **Define Origin Operation Setting** : การตั้งค่าพารามิเตอร์ของคำสั่ง Origin Search ต้องทำการคลิกที่ช่องสี่เหลี่ยมหน้า Use Define Origin Operation ก่อน จึงจะตั้งค่าต่างๆ ได้

Search Direction: ทิศทางการหาตำแหน่ง Origin

- CW: ตามเข็มนาฬิกา
- CCW: ทวนเข็มนาฬิกา

Detection Method: วิธีการจับสัญญาณจาก phase z

โดยปกติจะใช้งานควบคู่กับ Proximity เมื่อ Proximity สามารถรับสัญญาณได้ ก็จะสั่งให้รับสัญญาณจาก phase z (Origin input signal) ซึ่งสามารถกำหนดได้ 3 รูปแบบ ดังนี้

- Method 0: เมื่อ proximity แสดงสถานะเป็น OFF -> ON -> OFF แล้วค่อยตรวจจับ Origin input signal
- Method 1: เมื่อ proximity แสดงสถานะเป็น OFF -> ON แล้วค่อยตรวจจับ Origin input signal
- Method 2: ไม่ต้องอาศัย proximity สามารถตรวจจับ Origin input signal ได้โดยตรง

Search Operation: เป็นการทำ Origin search แบบควบคู่กับ limit switch มี 2 แบบคือ

- Inverse 1: เมื่อรับสัญญาณจาก limit switch ในขณะที่กำลังทำ Origin search ให้เปลี่ยนทิศทางการหมุน และทำ Origin search ต่อไป
- Inverse 2: เมื่อรับสัญญาณจาก limit switch ในขณะที่กำลังทำ Origin search ให้หยุดการทำงาน แล้วแสดงเป็น error

Operation Mode: เลือกโหมดการทำงานของ Servo driver

- Mode 0: ใช้ควบคุม stepping motor Driver ซึ่ง Mode 1 และ 2 ก็สามารถควบคุมได้
- Mode 1: ใช้ควบคุม servo motor โดยไม่จับสัญญาณตรวจสอบการเคลื่อนที่(INP)ทำให้ลดเวลาประมวลผลลง แต่ความแม่นยำก็จะลดลงด้วย
- Mode 2: ใช้ควบคุม servo motor โดยมีการจับสัญญาณตรวจสอบการเคลื่อนที่(INP)ใช้สำหรับการเคลื่อนที่ ที่ต้องการความแม่นยำสูง

Origin Input Signal: ชนิดของสัญญาณจาก phase z

- NC: ปกติปิด
- NO: ปกติเปิด

Proximity Input Signal: ชนิดของสัญญาณจาก proximity origin input

- NC: ปกติปิด
- NO: ปกติเปิด

OMRON ELECTRONICS CO.,LTD.

Search High Speed: ความเร็วสูงช่วงต้นในการเริ่มต้นหา origin

- ใส่ค่าตามต้องการ (pps)

Search Proximity Speed: ความเร็วต่ำในการเตรียมหยุดที่ตำแหน่ง origin

- ใส่ค่าตามต้องการ (pps)

Search Compensation Value: กำหนดค่าชดเชยที่ตำแหน่ง origin

- ใส่ค่าตามต้องการ

Search Acceleration Ratio: อัตราเร่งในการหา origin

- ใส่ค่าตามต้องการ (Hz/4ms)

Search Deceleration Ratio: อัตราหน่วงในการหา origin

- ใส่ค่าตามต้องการ (Hz/4ms)

Positioning Monitor Time: ระบุช่วงเวลาที่รอสัญญาณ Positioning Completed Signal ส่งมาให้จากเซอร์โวไดรเวอร์ หลังจากที่เซอร์โวมอเตอร์เข้าถึงตำแหน่งเป้าหมายเรียบร้อยแล้ว ถ้าสัญญาณไม่มาภายในเวลา จะเกิด Positioning Timeout Error (error code 0300)

- ใส่ค่าตามต้องการ (ms)

3.2.3) Origin Return: **การตั้งค่าพารามิเตอร์ของคำสั่ง** Origin Return

Speed: ความเร็วในการทำ origin return

- ใส่ค่าตามต้องการ (pps)

Acceleration Ratio: อัตราเร่งในการทำ origin return

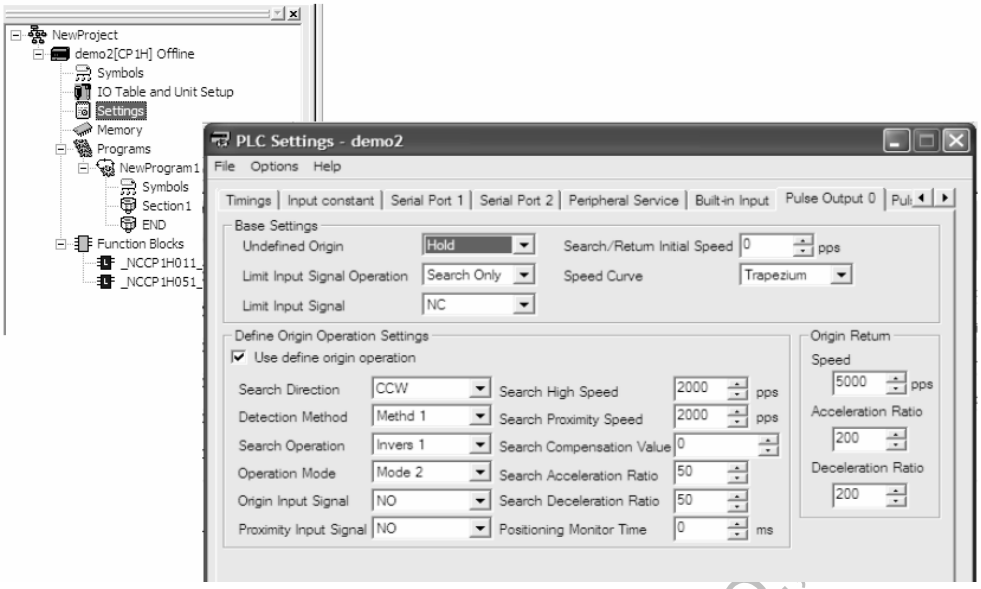
- ใส่ค่าตามต้องการ (Hz/4ms)

Deceleration Ratio: อัตราหน่วงในการทำ origin return

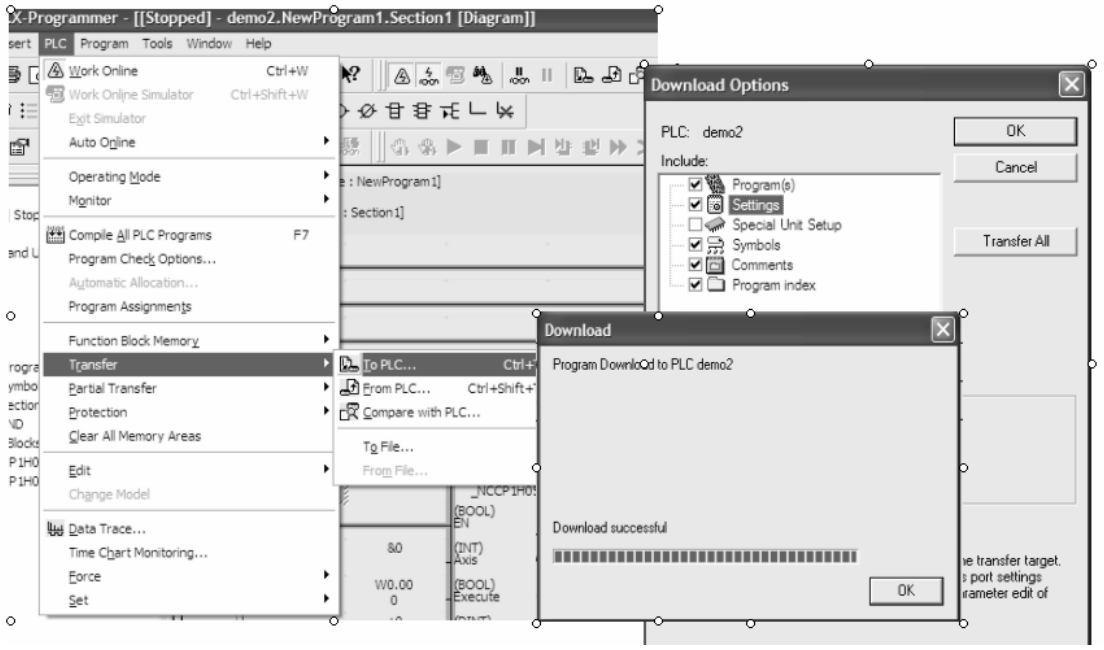
- ใส่ค่าตามต้องการ (Hz/4ms)

OMRON ELECTRONICS CO.,LTD.

เมื่อทำเข้าใจความหมายของพารามิเตอร์แต่ละตัวแล้วให้ป้อนค่าตามรูป



3.3) เลือก On-line /Transfer/To PLC/O.K.



3.4) ปิดไฟที่จ่ายให้กับ PLC แล้วเปิดใหม่เพื่อ Update ข้อมูล

4.) เขียนโปรแกรมควบคุมเซอร์โวมอเตอร์ที่พีแอลซีโดยใช้คำสั่ง Function Block

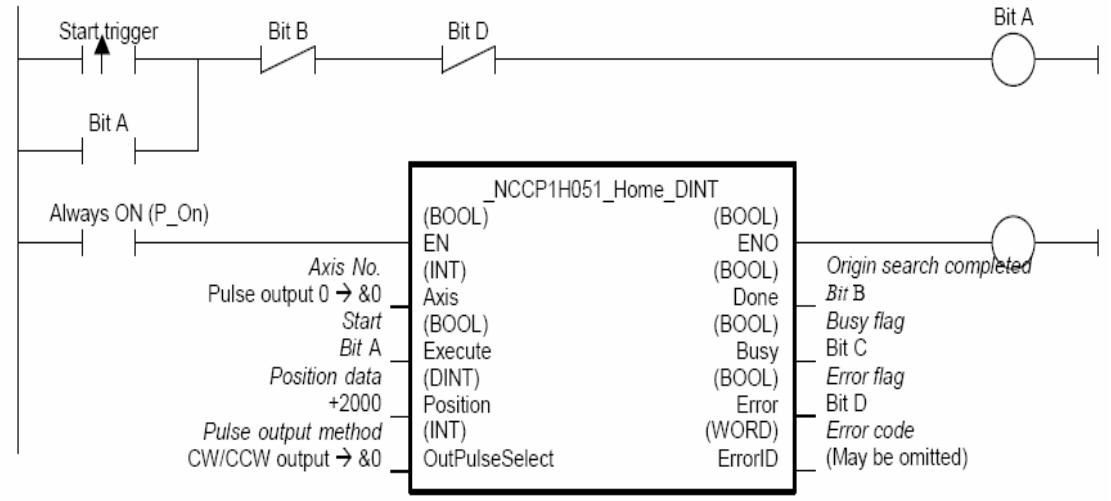
ในตัวอย่างนี้เราจะใช้คำสั่งฟังก์ชันบล็อก 2 คำสั่ง คือคำสั่ง Origin Search และคำสั่ง Move Absolute

คำสั่ง ORIGIN SEARCH มีหน้าที่อะไร?

คำสั่ง Origin Search มีหน้าที่สั่งให้เซอร์โวมอเตอร์กลับสู่ตำแหน่งจุดเริ่มต้น (Origin)

หลักการทำงาน

เมื่อ Start Bit A จะทำให้ฟังก์ชันนี้ทำงาน โดยเซอร์โวมอเตอร์จะเริ่มค้นหาตำแหน่ง Origin เมื่อพบจุด Origin แล้ว Bit B จะทำงาน (On) เพื่อตัดสัญญาณของ Bit A เซอร์โวมอเตอร์จะหยุด ณ ตำแหน่ง Origin หากเกิดความผิดปกติ(Error) Bit D จะทำงาน (On) เพื่อตัดการทำงานของ Bit A



ความหมายของตัวแปรอินพุต

Name	Variable name	Data type	Default	Range	Description
EN	EN	BOOL			1(ON):ใช้งาน FB
					0(OFF): ไม่ใช้งานFB
Axis No.	Axis	INT	&0	&0 - &3	&0:Pulse ouput 0
					&1:Pulse ouput 1
					&2:Pulse ouput 2
					&3:Pulse ouput 3
Start	Execute	BOOL	0(OFF)		ใช้งาน origin search
				-2,147,483,648	ระบุตำแหน่งที่เคลื่อนที่เป็นจุดเริ่มต้นใหม่
Position Data	Position	DINT	(+0)	to	(Pulse)
				2,147,483,648	
Pulse output	Out Pulse Select	INT	&0	&0 - &1	&0:CW/CCW output
Method					&1:pulse +direction output

ความหมายของตัวแปรเอาท์พุต

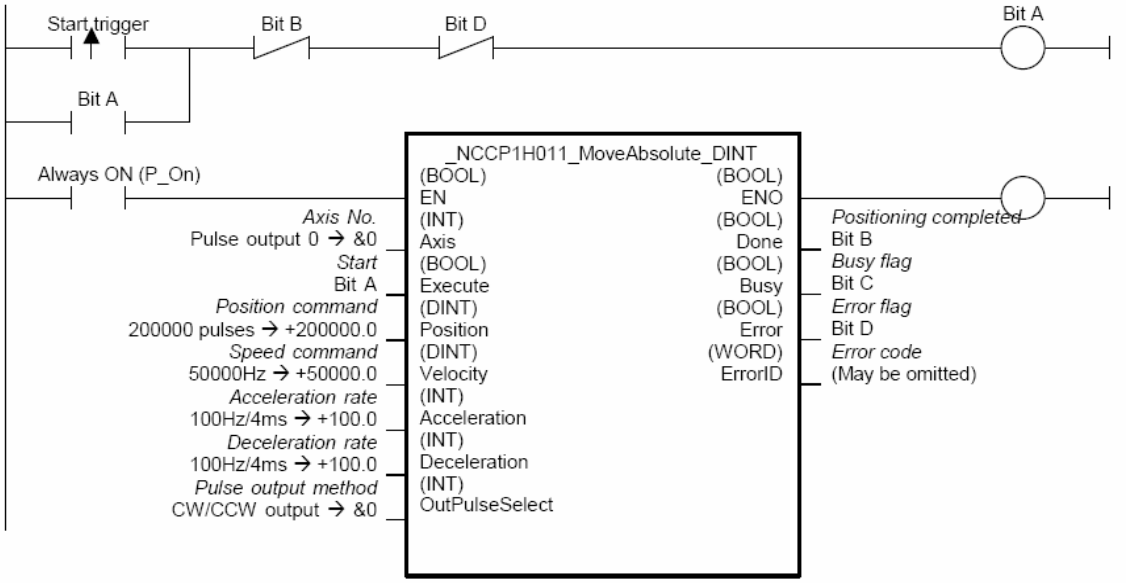
Name	Variable name	Data type	Default	Range	Description
ENO	ENO	BOOL			1(ON):FB ทำงานปกติ
					0(OFF): FB ทำงานผิดปกติ
Origin search	Done	BOOL			1(ON):เมื่อเซอร์โวมอเตอร์
completed					เข้าสู่ตำแหน่งเริ่มต้นแล้ว
Busy flag	Busy	BOOL			1(ON):เมื่ออยู่ในกระบวนการทำงาน
Error flag	Error	BOOL			1(ON):เมื่อเกิดความผิดพลาดในการทำงาน
Error code	ErrorID	WORD			แสดง Error Code โดยสามารถเทียบได้
(May be omitted)					จากคู่มือการใช้งานหากนอกเหนือที่มี
					จะแสดงเป็น #0000

คำสั่ง MOVE ABSOLUTE มีหน้าที่อะไร?

คำสั่ง Move Absolute ทำหน้าที่สั่งให้เซอร์โวมอเตอร์ในแกนที่ต้องการ หมุนตามตำแหน่ง, ความเร็ว, อัตราเร่ง, อัตราหน่วงที่กำหนด โดยมีเงื่อนไขก่อนใช้คำสั่ง Move Absolute ต้องทำคำสั่ง Origin Search ก่อน

หลักการทำงาน

เมื่อ Start Bit A จะทำให้ฟังก์ชันนี้ทำงาน เซอร์โวมอเตอร์จะหมุนตามตัวแปรที่กำหนดไว้ ในขณะที่ Bit C จะทำงาน (On) จนกระทั่งเซอร์โวมอเตอร์หมุนถึงตำแหน่งที่กำหนดไว้ Bit B จะทำงาน (On) เพื่อตัดสัญญาณ Bit A และหากเกิดความผิดปกติ(Error) Bit D จะทำงาน (On) เพื่อตัดการทำงานของ Bit A



ความหมายของตัวแปรอินพุต

Name	Variable name	Data type	Default	Range	Description
EN	EN	BOOL			1(ON):ใช้งาน FB
					0(OFF): ไม่ใช้งานFB
Axis No.	Axis	INT	&0	&0 - &3	&0:Pulse ouput 0
					&1:Pulse ouput 1
					&2:Pulse ouput 2
					&3:Pulse ouput 3
Start	Execute	BOOL	0(OFF)		สั่งให้เซอร์โวมอเตอร์เคลื่อนที่
				-2,147,483,648	ระบุตำแหน่งของ pulse ที่ต้องการให้เคลื่อนที่
Position Data	Position	DINT	(+0)	to	(Pulse)
command				2,147,483,648	
Speed command	Velocity	DINT	&1	&1 to	ระบุความเร็วที่ต้องการ (Hz)
				& 1000000	
Acceleration rate	Acceleration	REAL	(+1.0)	(+1.0) to	ระบุอัตราเร่งที่ต้องการ (Hz/4ms)
				(+65535.0)	
Deceleration rate	Deceleration	REAL	(+1.0)	(+1.0) to	ระบุอัตราหน่วงที่ต้องการ (Hz/4ms)
				(+65535.0)	
Pulse output	Out Pulse Select	INT	&0	&0 - &1	&0:CW/CCW output
Method					&1:pulse +direction output

ความหมายของตัวแปรเอาท์พุต

Name	Variable name	Data type	Default	Range	Description
ENO	ENO	BOOL			1(ON):FB ทำงานปกติ
					0(OFF): FB ทำงานผิดปกติ
Position	Done	BOOL			1(ON):เมื่อเซอร์โวมอเตอร์
completed					เคลื่อนที่ไปยังตำแหน่งที่ระบุแล้ว
Busy flag	Busy	BOOL			1(ON):เมื่ออยู่ในกระบวนการทำงาน
Error flag	Error	BOOL			1(ON):เมื่อเกิดความผิดพลาดในการทำงาน
Error code	ErrorID	WORD			แสดง Error Code โดยสามารถเทียบได้
(May be omitted)					จากคู่มือการใช้งานหากนอกเหนือที่มี
					จะแสดงเป็น #0000

บิตข้อมูลแสดงสถานะการเคลื่อนที่คืออะไร?

ข้อมูลการเคลื่อนที่ของเซอร์โวมอเตอร์จะถูกเก็บไว้ในหน่วยความจำประเภท Auxiliary Area ซึ่งมีประโยชน์ในการเขียนโปรแกรมเป็นอย่างมากในการรับรู้สถานะปัจจุบันของระบบว่าเคลื่อนที่เป็นอย่างไร ณ เวลานั้นๆ โดยการใช้บิต(Bit) หรือเวิร์ด(Word)ของ Auxiliary Area มาเป็น Flag Status ของโปรแกรม

ตารางแสดง Flag Status ที่ Auxiliary Area

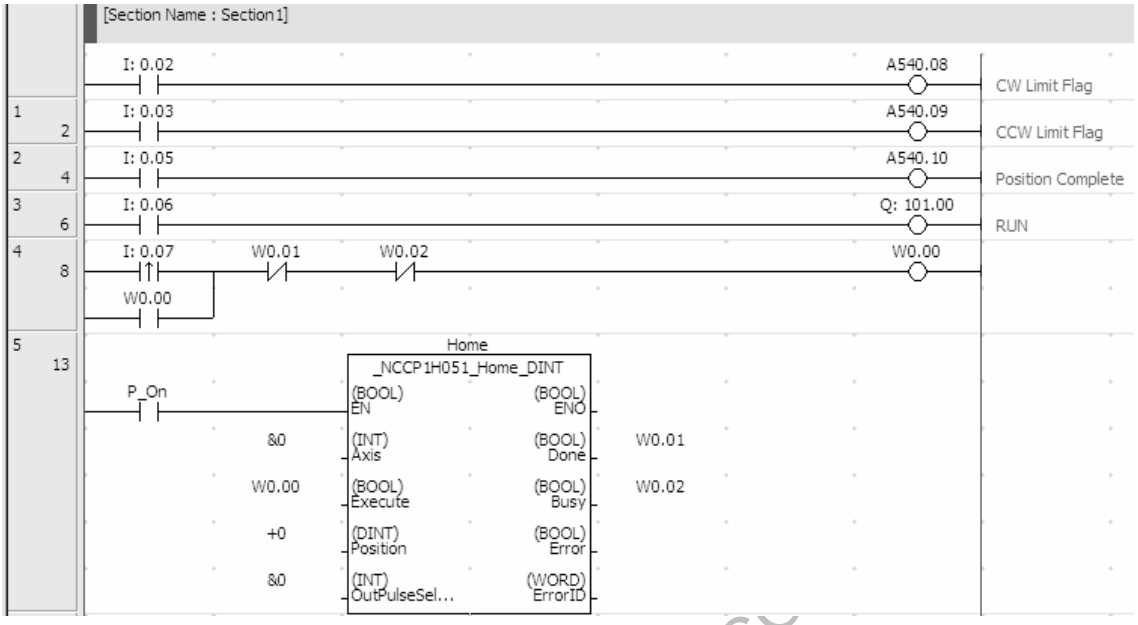
Function		Pulse Output Number			
		0	1	2	3
เก็บค่า PV ของ Pulse Output	Leftmost 4 digits	A277	A279	A323	A325
เก็บในช่วง 80000000 – 7FFFFFFF hex (-2,147,483,648 – 2,147,483,647)	Rightmost 4 digits	A276	A278	A322	A324
Reset Bits ค่า pulse output PV จะถูกรีเซตเมื่อบิตนี้เปลี่ยนสถานะจากปิดเป็นเปิด	0: Not cleared. 1: Clear PV.	A540.00	A541.00	A542.00	A543.00
CW Limit Input Signal Flags ใช้สัญญาณในการทำ origin search	On เมื่อได้รับสัญญาณจากภายนอก	A540.08	A541.08	A542.08	A540.08
CCW Limit Input Signal Flags ใช้สัญญาณในการทำ origin search	On เมื่อได้รับสัญญาณจากภายนอก	A540.09	A541.09	A542.09	A543.09
Positioning Complete Input Signal แสดงถึงการเข้าที่เรียบร้อยแล้วในการทำ ORG Search	On เมื่อได้รับสัญญาณจากภายนอก	A540.10	A541.10	A542.10	A543.10
Accel/Decel Flags	0: Constant speed	A280.00	A281.00	A326.00	A327.00

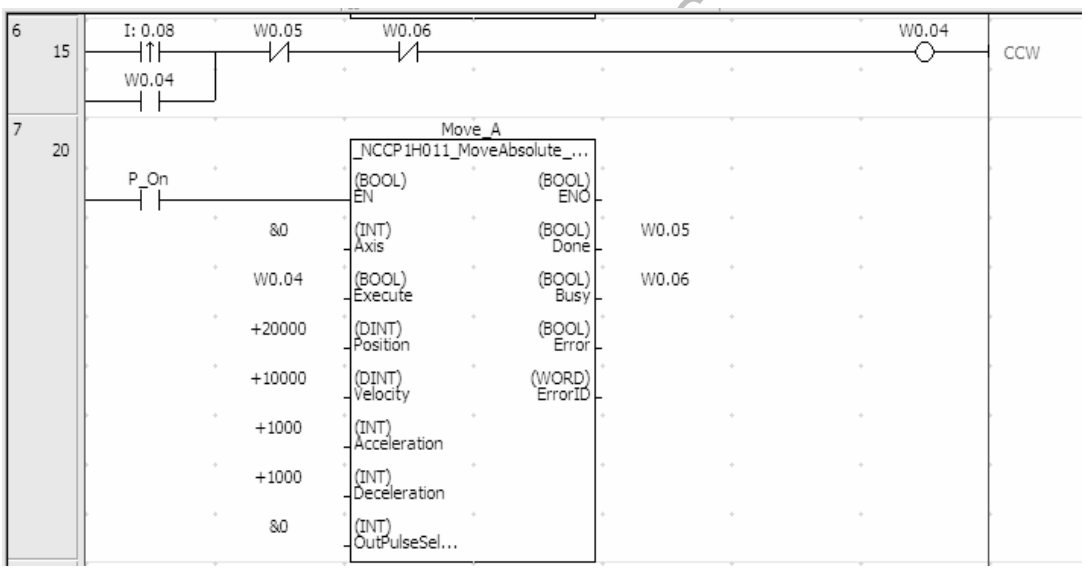
OMRON ELECTRONICS CO.,LTD.

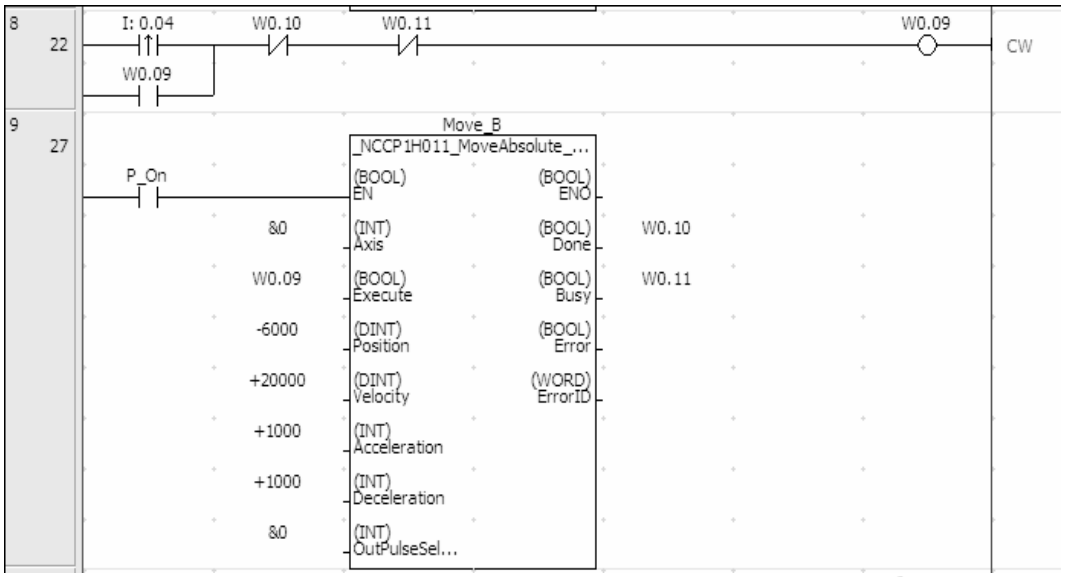
On เมื่อกำลังอยู่ในช่วงที่pulse output เปลี่ยนแปลงความถี่ จากชุดคำสั่ง ACC(888) หรือ PLS2(887)	1: Accelerating or Decelerating				
Overflow/Underflow Flags On เมื่อมี overflowหรือunderflow ในค่าPV	0: Normal 1: Overflow or Underflow	A280.01	A281.01	A326.01	A327.01
Output Amount Set Flags On เมื่อมีการตั้งค่าจำนวนpulse จากคำสั่ง PULS(886)	0: No setting 1: Setting made	A280.02	A281.02	A326.02	A327.02
Output Completed Flags On เมื่อมีการจ่ายpulseที่ตั้งไว้จากคำสั่ง PULS(886)หรือPLS2(887) เรียบร้อยแล้ว	0: Output not completed 1: Output completed	A280.03	A281.03	A326.03	A327.03
Output In-progress Flags On เมื่อเริ่มมีการจ่าย pulse output	0: Stopped 1: Outputting pulses	A280.04	A281.04	A326.04	A327.04
No-origin Flags On เมื่อยังไม่มีการกำหนด origin	0: Origin established 1: Origin not established	A280.05	A281.05	A326.05	A327.05
At-origin Flags On เมื่อค่าPVเป็น 0 (อยู่ที่ origin)	0: Not stopped at origin 1: Stopped at origin	A280.06	A281.06	A326.06	A327.06
Output Stopped Error Flags On เมื่อเกิด error ในขณะที่กำลังจ่าย pulse output เมื่อทำ ORG Search	0: No error 1: Stop error occurred	A280.07	A281.07	A326.07	A327.07
Stop Error Codes	-	A444	A445	A438	A439

OMRON ELECTRONICS CO., LTD.

ขั้นตอน เขียนโปรแกรมตามรูป







การทำงานของโปรแกรม

ฟังก์ชันบล็อกแรก เป็นการสั่งให้เซอร์โวมอเตอร์ค้นหาตำแหน่ง Home Position จากวงจร Bit ที่สั่งให้ฟังก์ชันบล็อกทำงานคือ W0.00 ตำแหน่งในการเคลื่อนที่เราจะกำหนดที่ Position , ส่วนความเร็วในการเคลื่อนที่กำหนดที่ Velocity, อัตราเร่งและอัตราหน่วงกำหนดที่ Acceleration และ Deceleration จะถูกกำหนดที่ Output Pulse Setting

เมื่อเซอร์โวมอเตอร์เคลื่อนที่ถึงตำแหน่งHome Position Bit W0.01 จะทำงานเพื่อสั่งให้ Bit W0.00 ตัดการทำงานของฟังก์ชันบล็อก

ฟังก์ชันบล็อกตัวที่สอง เป็นการสั่งให้เซอร์โวมอเตอร์เคลื่อนที่ไปหยิบชิ้นงานที่ตำแหน่ง A จากวงจร Bit W0.04 ทำหน้าที่สั่งให้ฟังก์ชันบล็อกทำงานเมื่อเคลื่อนที่ถึงตำแหน่งที่กำหนดไว้แล้ว Bit W 0.05 จะทำงาน เพื่อสั่งให้ Bit W0.04 ตัดการทำงานของฟังก์ชันบล็อก

ฟังก์ชันบล็อกสุดท้าย เป็นการสั่งให้เซอร์โวมอเตอร์เคลื่อนที่ไปหยิบชิ้นงานไปวางที่ตำแหน่ง B จากวงจร Bit W0.09 ทำหน้าที่สั่งให้ฟังก์ชันบล็อกทำงานเมื่อเคลื่อนที่ถึงตำแหน่งที่กำหนดไว้แล้ว Bit W 0.10 จะทำงาน เพื่อสั่งให้ Bit W0.09 ตัดการทำงานของฟังก์ชันบล็อก

5.) การทดสอบโปรแกรม

หลังจากดาวน์โหลดโปรแกรมเรียบร้อยแล้ว การทดสอบโปรแกรมต้องสั่งให้พีแอลซี อยู่ในโหมดมอนิเตอร์(Monitor) หรือรัน(Run) โหมด กดสวิทช์ SW.0 เพื่อให้เซอร์โวมอเตอร์อยู่ในตำแหน่งพร้อมทำงาน (ล็อกโรเตอร์) การสั่งให้ Table เคลื่อนที่หาตำแหน่ง Origin Search สามารถทำได้โดยกด SW.1 Table จะเคลื่อนที่ไปตามค่าพารามิเตอร์ที่ได้กำหนดไว้ในPLC Settings รอจนกระทั่งเซอร์โวมอเตอร์เข้าสู่จุด Home แล้วจึงทดสอบคำสั่ง Absolute โดยการกด SW2. Table จะต้องเคลื่อนไปที่ตำแหน่ง A ตามระยะที่กำหนดไว้ เมื่อกด SW.3 Table จะต้องเคลื่อนไปที่ตำแหน่ง B ตามระยะที่กำหนดไว้

OMRON ELECTRONICS CO., LTD.

หมายเหตุ เนื่องจากโปรแกรมนี้มีตำแหน่งHomeซึ่งเป็นการเคลื่อนที่ในแนวเส้นตรง จึงต้องมีการตรวจจับสถานะของอินพุท CW (limit 0.02) และ CCW (limit 0.03) ซึ่งสัญญาณทั้งสองนี้จะไปแสดงค่าที่ Bit A540. 08 และ A540.09 ซึ่งเป็น Bit ที่ช่วยในการทำงานของคำสั่ง Origin Search กรณีที่มอเตอร์เคลื่อนที่ไปในทิศทางตรงกันข้ามกับตำแหน่ง Home มอเตอร์จะหมุนกลับไปในทิศทางที่ถูกต้องโดยอัตโนมัติ เมื่อถึงตำแหน่งของ CW limit หรือ CCW limit

*****//****//*****

OMRON ELECTRONICS CO., LTD.

ส่งท้ายเซอร์โวมอเตอร์สำหรับมือใหม่

สำหรับมือใหม่ทุกท่านที่ได้อ่านรายละเอียดหรือมีโอกาสได้ทดลองปฏิบัติตามตัวอย่างจนจบเนื้อหาในคู่มือเล่มนี้แล้ว อยากให้มือใหม่ทุกท่านเข้าใจว่าเป็นเพียงแค่การเริ่มต้น ยังมีเรื่องของการควบคุมเซอร์โวมอเตอร์อีกมากมายให้ท่านได้ศึกษาเรียนรู้เพิ่มเติม เช่น การควบคุมเซอร์โวมอเตอร์แบบใช้อนาลอกอินพุต(Analog Input) หรือการใช้งานไดร์เวอร์ที่ทำงานเป็นแบบเครือข่าย (Network) โดยสั่งงานผ่านโปรโตคอลที่มีการสื่อสารความเร็วสูง ทำให้ผู้ใช้งานสามารถเข้าถึงพารามิเตอร์ภายในไดร์เวอร์เพื่อเปลี่ยนแปลงค่าต่างๆได้อย่างเหมาะสมกับโหลด ณ ขณะนั้น

ส่วนงานควบคุมตำแหน่งแบบ PTP (Point to Point) ก็มีคอนโทรลเลอร์แบบยูนิต ประเภทต่างๆให้พิจารณาเลือกใช้งานตามลักษณะงาน เช่น NC Unit เหมาะกับการควบคุมตำแหน่งแบบ Interpolation , MC Unit เหมาะสำหรับงาน ที่มีการเคลื่อนที่เป็นเส้นโค้งตาม Curve โดยใช้การโปรแกรมแบบ G-code หรือถ้าเป็นงาน Synchronize ก็เลือกใช้ MCH Unit ซึ่งจะทำให้เซอร์โวมอเตอร์หลายๆแกนในระบบเคลื่อนที่ประสานงานกันได้อย่างลงตัว เป็นต้น

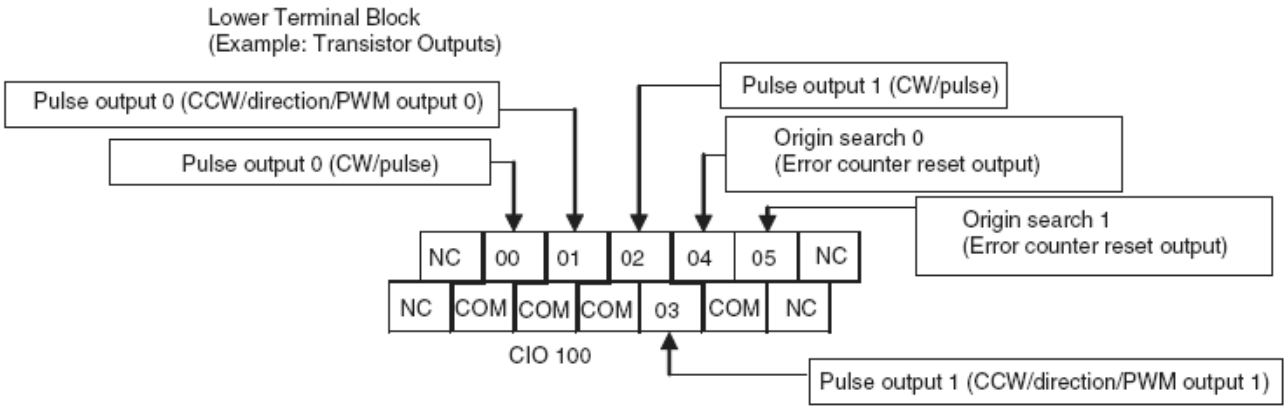
อย่างไรก็ตามสำหรับท่านที่ต้องการทราบข้อมูลเพิ่มเติมหรือต้องการเข้าฝึกอบรม เกี่ยวกับการควบคุมเซอร์โวมอเตอร์ สามารถติดต่อได้โดยตรงที่ฝ่ายการตลาดผลิตภัณฑ์ บริษัท ออมรอนอีเลคทรอนิคส์ จำกัด โทรศัพท์ 0-2942-6700

OMRON ELECTRONICS CO., LTD.

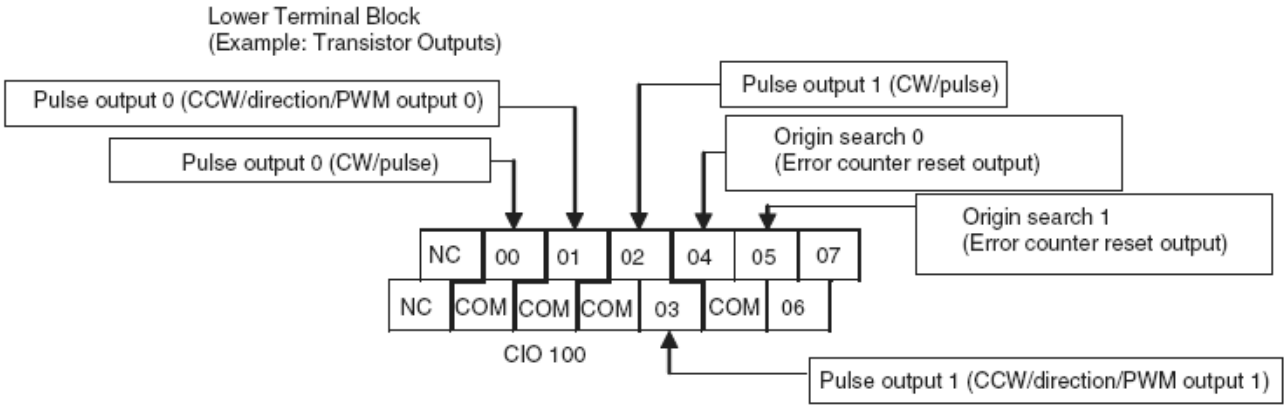
CP1L สามารถต่อเซอร์โวมอเตอร์ได้ 2 ชุด คือ output 0 และ output 1

การต่อสายด้านเอาท์พุต

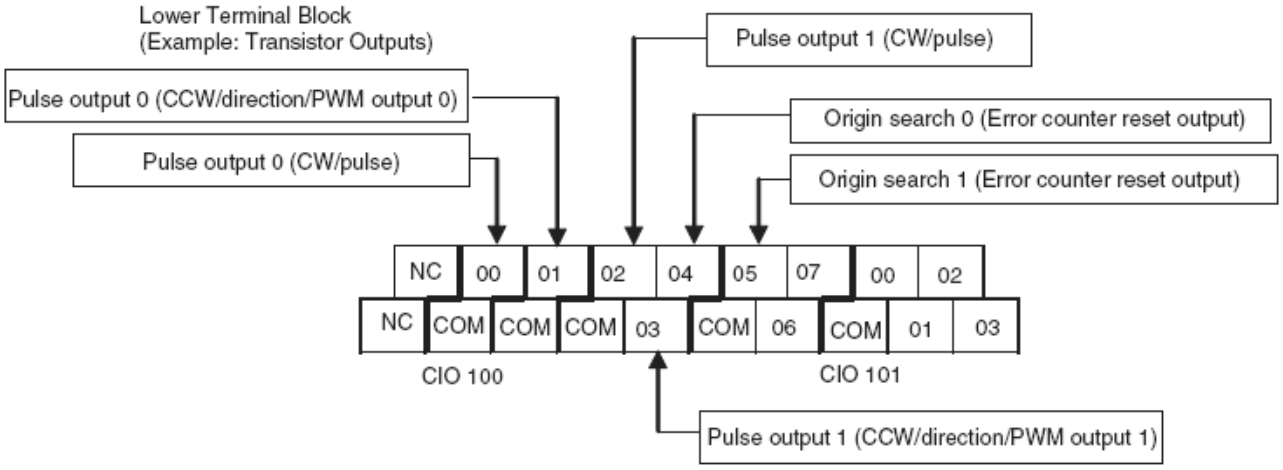
■ CPU Unit with 14 I/O Points



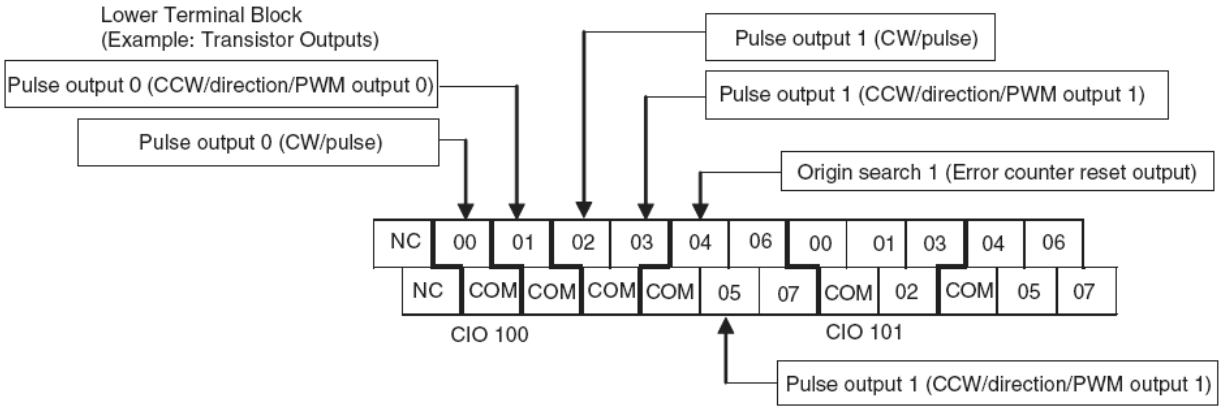
■ CPU Unit with 20 I/O Points



■ CPU Unit with 30 I/O Points

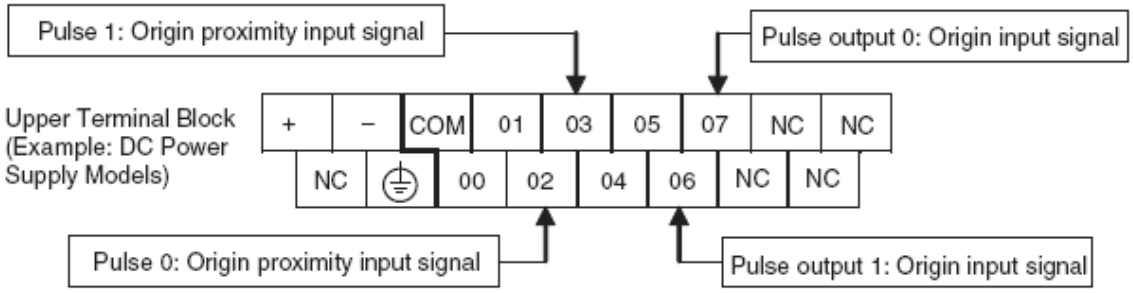


■ **CPU Unit with 40 I/O Points**

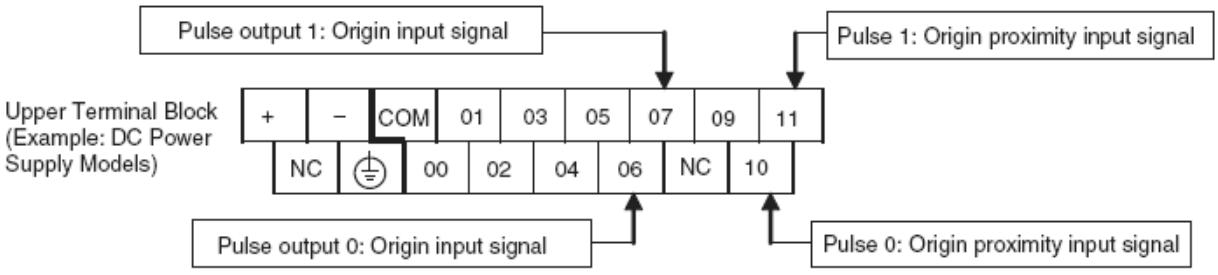


<u>การต่อสายด้านอินพุต</u>

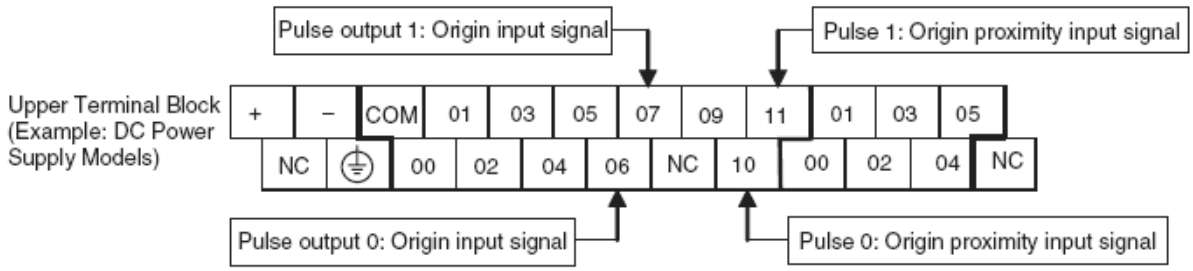
CPU Unit with 14 I/O Points

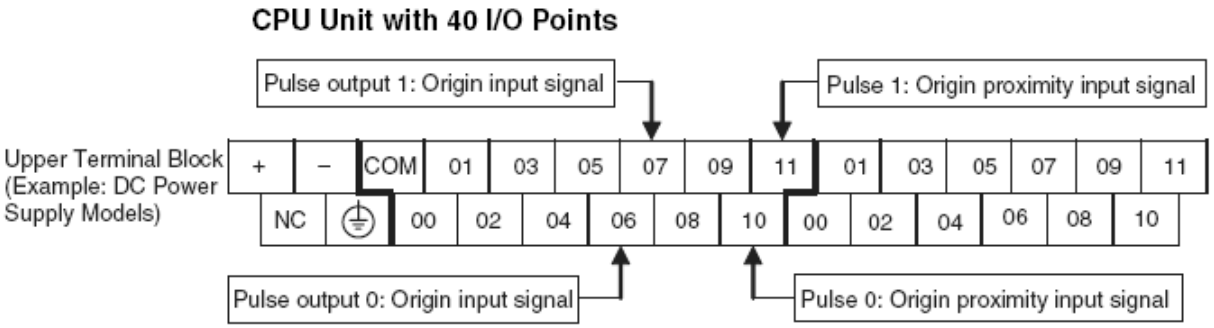


CPU Unit with 20 I/O Points



CPU Unit with 30 I/O Points

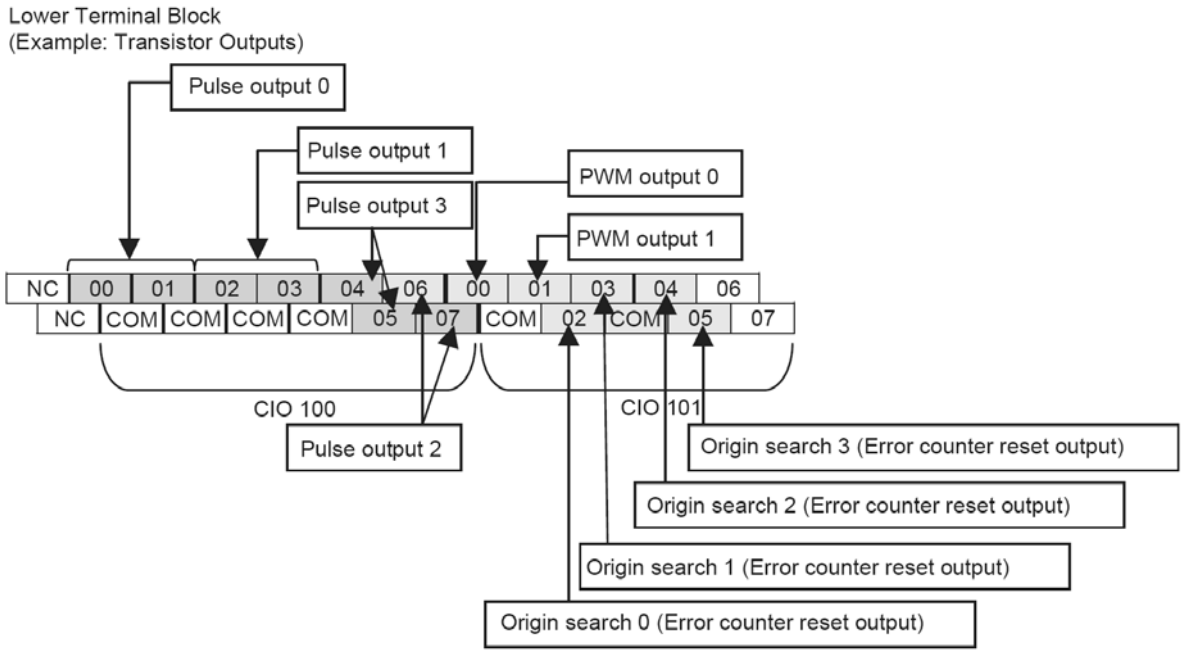




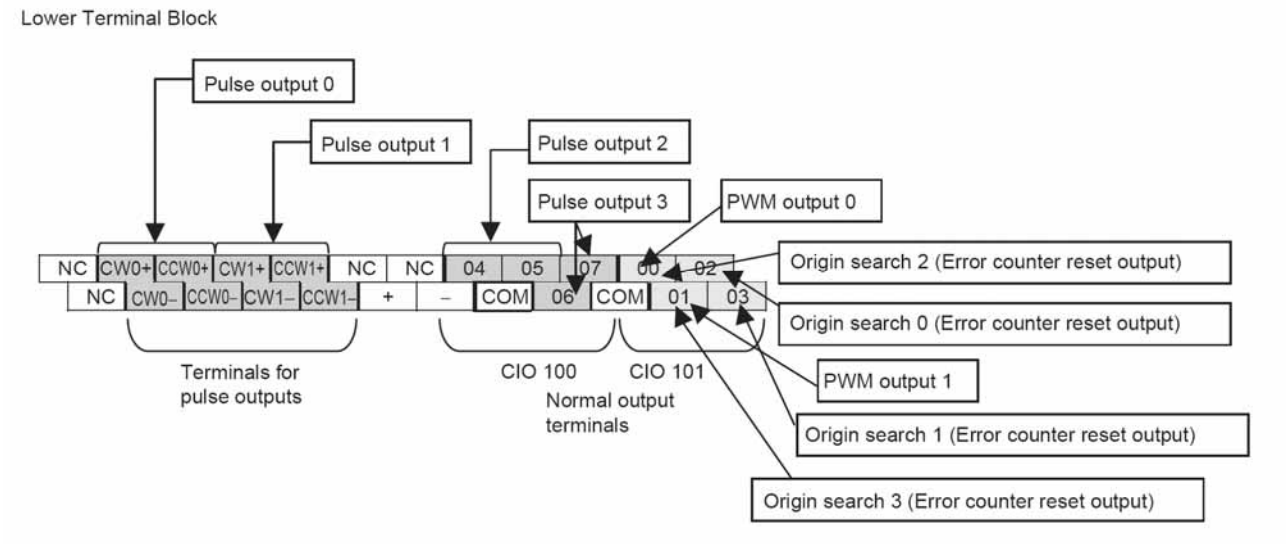
CP1H สามารถต่อเซอร์โวมอเตอร์ได้ 4 ชุด คือ output 0, output 1, output 2 และ output 3

<u>การต่อสายด้านเอาท์พุต</u>

X/XA CPU Unit

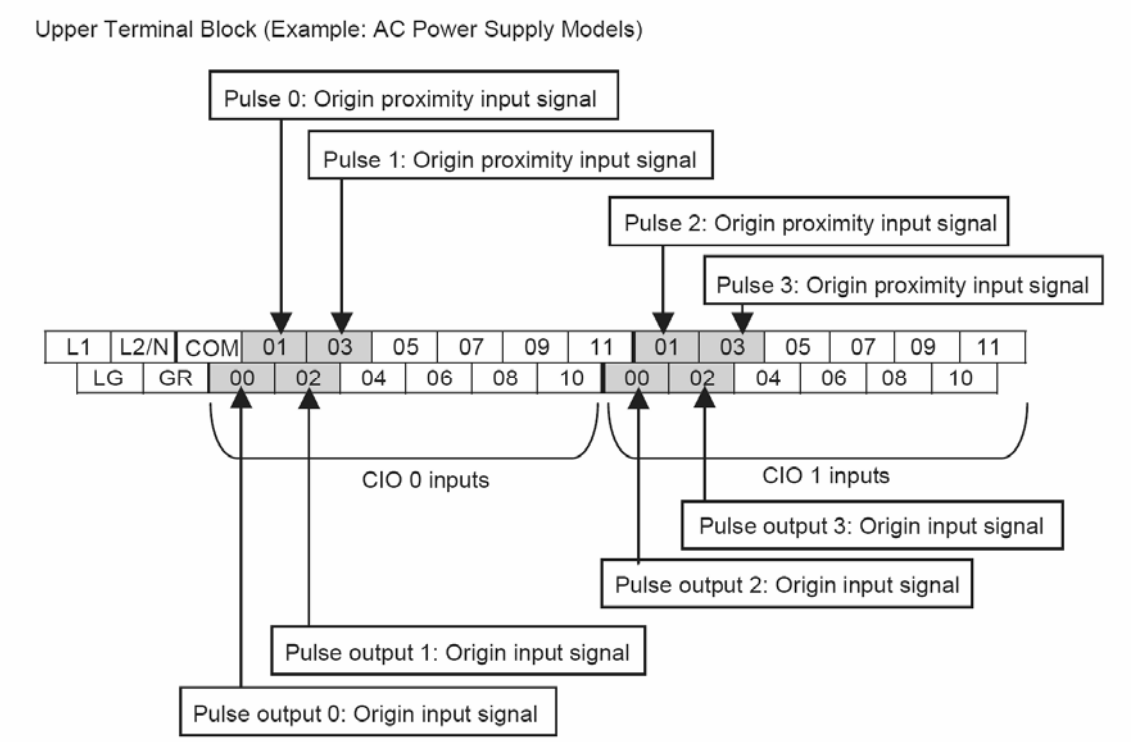


Y CPU Unit

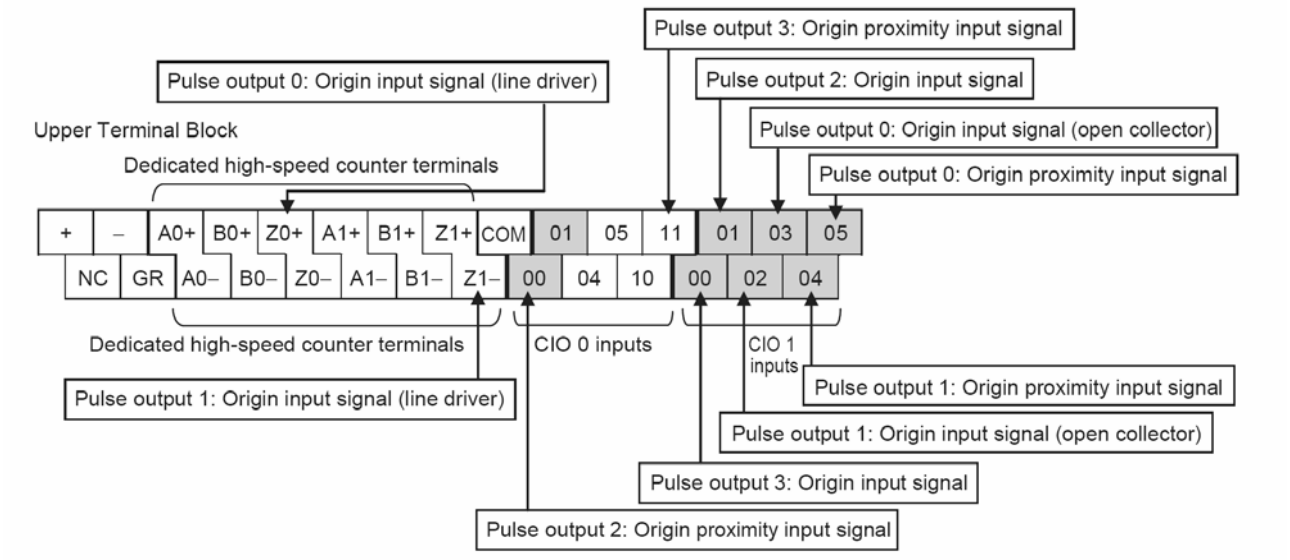


<u>การต่อสายด้านอินพุต</u>

X/XA CPU Unit



Y CPU Unit



OMRON ELECTRONICS CO., LTD

การควบคุมเซอร์โวมอเตอร์ สำหรับมือใหม่

บริการเกี่ยวกับเซอร์โวมอเตอร์:

- ข้อมูลทางเทคนิค
- หลักสูตรฝึกอบรม
- ชุดทดลองสำหรับการเรียนรู้

OMRON

PLCEASY
www.plceasy.com

บริษัท ออมรอน อีเลคทรอนิคส์ จำกัด

อาคารเอสเอ็มทาวเวอร์ ชั้น 20 ถ.พหลโยธิน แขวงลุมพินี เขตปทุมวัน กรุงเทพฯ 10330

Call Center: 0-2643-6700 Fax: 0-2637-0501

E-Mail: omron-th@eu.omron.com

Website: www.omron-ap.co.th, www.plceasy.com