



การจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชันเซนเซอร์

A Simulation of Electric Car's Motor Control with Fusion Sensor

ภูริศักดิ์ ศรีขจร
Purisak Srikajon

สหัสวรรษ สังขนิษฐ
Sahatsawat Sangkanit

ปริญญาโทนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชันเซนเซอร์

A Simulation of Electric Car's Motor Control with Fusion Sensor

โดย



ปริญญาโทนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2565

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชันเซนเซอร์

A Simulation of Electric Car's Motor Control with Fusion Sensor

ผู้จัดทำ นายภูริศักดิ์ ศรีขจร รหัสนักศึกษา 62010731

นายสหัสวรรษ สังขนิษฐ รหัสนักศึกษา 62010927

ปริญญาานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว



ผศ.ดร. ยุทธนา คิดใจเดียว

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการ	การจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชั่น เซนเซอร์	
นักศึกษา	นายภูริศักดิ์ ศรีขจร	รหัสนักศึกษา 62010731
	นายสหัสวรรษ สังขนิษฐ	รหัสนักศึกษา 62010927
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์	
ปีการศึกษา	2565	
อาจารย์ที่ปรึกษาโครงการ	ผศ.ดร. ยุทธนา คิดใจเดียว	

บทคัดย่อ

โครงการฉบับนี้นำเสนอเกี่ยวกับการจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชั่นเซนเซอร์ ด้วยการใช้งานโปรแกรม MATLAB และ Simulink โดยมอเตอร์รถยนต์ไฟฟ้าที่ผู้จัดทำโครงการเลือกมาใช้ในการจำลองคือ PMSM : Permanent Magnet Synchronous Motor ซึ่งก็คือมอเตอร์ซิงโครนัสชนิดแม่เหล็กถาวร โดยจุดประสงค์หลักในการจำลองการควบคุมมอเตอร์คือสามารถควบคุมความเร็วในการหมุนของมอเตอร์ได้ (RPM) ซึ่งผู้จัดทำโครงการจะใช้ตัวควบคุมพีไอในการควบคุมการหมุนของมอเตอร์ PMSM นี้ อีกทั้งโครงการนี้ยังประกอบไปด้วยวิธีการสร้างระบบการจำลองการทำงานของมอเตอร์รถยนต์ไฟฟ้า และได้มีการอธิบายถึงความสัมพันธ์ของระบบการควบคุมการทำงานของมอเตอร์รถยนต์ไฟฟ้าไว้อีกด้วย

คำสำคัญ : มอเตอร์ซิงโครนัสชนิดแม่เหล็กถาวร, ระบบควบคุมการทำงาน, ตัวควบคุมพีไอ, บล็อกการทำงาน, ฟิวชั่นเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	A Simulation of Electric Car's Motor Control with Fusion Sensor		
Student	Mr. Purisak	Srikajon	Student ID 62010731
	Mr. Sahatsawat	Sangkanit	Student ID 62010927
Degree	Bachelor of Engineering		
Program	Electronics Engineering		
Year	2022		
Project Advisor	Asst. Prof. Yuttana Kitjaidure, Ph.D		

ABSTRACT

This project presents a simulation of electric car's motor control by fusion sensor. By using MATLAB and Simulink programs, the electric car's motor that we use in the simulation is PMSM: Permanent Magnet Synchronous Motor, which is a permanent magnet synchronous motor. The main purpose of simulating motor control can control the rotational speed of the motor (RPM). We will use the PI controller. In controlling the rotation of this PMSM motor, this project also consists of a method for creating a simulation system for the operation of an electric car's motor. and has explained the relations of the control system of the electric car's motor as well.

Keywords: Permanent Magnet Synchronous Motor, Control system, PI Controller, Block diagram, Fusion Sensors

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการงานจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชันเซนเซอร์ ในครั้งนี้สำเร็จได้ด้วยดีเพราะได้รับความอนุเคราะห์และความเมตตาจากท่านอาจารย์ยุทธนา คิติใจเดียว อาจารย์ที่ปรึกษาที่คอยกำหนดแนวทาง คอยชี้แนะสิ่งต่างๆ และได้ให้ความอนุเคราะห์แก่ผู้จัดทำ โครงการรวมอีกทั้งอาจารย์ทุกๆท่านที่ให้ความรู้และสั่งสอนมาจนถึงปัจจุบัน ขอขอบพระคุณที่ให้การสนับสนุนด้วยดีตลอดมา ขอขอบพระคุณ คุณพ่อคุณแม่ ผู้ปกครอง และผู้อุปการะคุณทุกท่านที่ให้การสนับสนุนความช่วยเหลือเป็นอย่างดี

ภูริศักดิ์ ศรีขจร
สหสวรรค์ สังขนิษฐ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูป.....	VII
สารบัญตาราง.....	XI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	1
1.5 ระยะเวลาทำการวิจัยและแผนการดำเนินงานโครงการ.....	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	3
2.1 Permanent-Magnet Synchronous Motor.....	3
2.2 ฟิวชั่นเซนเซอร์ (Fusion Sensor).....	4
2.2.1 เซนเซอร์วัดกระแส (Current Sensor).....	5
2.2.2 เซนเซอร์วัดแรงดัน (Voltage Sensor).....	5
2.2.3 เซนเซอร์วัดแรงบิด (Torque Sensor).....	6
2.2.4 เซนเซอร์วัดตำแหน่งภายในมอเตอร์ (Encoder).....	12
2.3 Field Oriented Control Theory.....	16
2.3.1 ข้อดีของ FOC.....	16

2.3.2 เทคนิคของ Field Oriented Control ที่เกี่ยวข้องกับ Three reference frame และการแปลง Reference frame จากแบบหนึ่งเป็นอีกแบบหนึ่ง.....	17
2.4 Transformation	18
2.4.1 การแปลง Clarke Transform.....	19
2.4.2 การแปลง Park Transform	19
2.4.3 การแปลง Inverse Park Transform.....	20
2.4.4 การแปลง Inverse Clarke Transform	20
2.5 Sinusoidal Voltage และ SVPWM Generation	21
2.6 ตัวควบคุมพีไอดี (PID Controller).....	22
2.6.1 คุณสมบัติของตัวควบคุมแบบพีไอดี.....	23
2.6.2 จุดประสงค์ในการปรับจูนตัวควบคุม.....	25
2.6.3 วิธีการปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี.....	26
บทที่ 3 วิธีดำเนินงาน	30
3.1 เครื่องมืออุปกรณ์และข้อมูล	30
3.1.1 MATLAB & Simulink.....	30
3.1.2 Simulink-Toolboxes.....	30
3.2 ขั้นตอนการดำเนินงาน	31
3.2.1 ภาพรวมของระบบการจำลองการทำงานของมอเตอร์รถยนต์ไฟฟ้าด้วยฟิวชั่น เซนเซอร์.....	31
3.2.2 บล็อก TestSignals.....	31
3.2.3 บล็อก Controller	32
3.2.4 บล็อก Plant	40
3.2.5 บล็อก Evaluation	45
3.2.5 การใช้งาน PI Controller.....	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 ผลการดำเนินงาน	48
4.1 ผลจากการจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชันเซนเซอร์.....	48
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ	51
5.1 สรุปผลการทดลอง.....	51
5.2 วิจัยารณ์ผลการทดลองและข้อเสนอแนะ.....	51
บรรณานุกรม.....	52



สารบัญรูป

รูปที่	หน้า
2.1 PMSM และโทโพโลยีบริดจ์อินเวอร์เตอร์.....	3
2.2 การสร้างแรงบิดในมอเตอร์แบบ Sinusoidal และ Trapezoidal.....	4
2.3 ตัวอย่างลักษณะของเซนเซอร์วัดกระแส.....	5
2.4 ตัวอย่างลักษณะของเซนเซอร์วัดแรงดันสำหรับยานยนต์.....	5
2.5 การวัดแรงบิดแบบ Moment Arm.....	6
2.6 การวัดแรงบิดโดยอาศัยหลักการของ Torsion Bar.....	6
2.7 เซนเซอร์วัดแรงบิดแบบ Strain-gauged Shaft.....	7
2.8 ตัวอย่างเซนเซอร์วัดแรงบิดแบบ Stain-gauged Shaft.....	7
2.9 ลักษณะของเซนเซอร์วัดแรงบิดแบบ Clamp-on Collar.....	8
2.10 ตัวอย่างเซนเซอร์วัดแรงบิดแบบ Clamp-on Collar ชนิดไร้สาย (Wireless).....	8
2.11 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Optical.....	9
2.12 เซนเซอร์วัดแรงบิดแบบ Tooth Wheel.....	9
2.13 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Magnetostrictive.....	10
2.14 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Laser.....	10
2.15 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Capacitive.....	11
2.16 ตัวอย่าง Linear Encoder.....	12
2.17 โครงสร้างของ Absolute Linear Encoder.....	13
2.18 โครงสร้างของ Increment Linear Encoder.....	14
2.19 ตัวอย่าง Rotary Encoder.....	14
2.20 โครงสร้างของเอ็นโคเดอร์แบบหมุน ชนิด Absolute Rotary Encoder.....	15
2.21 โครงสร้างของเอ็นโคเดอร์แบบหมุน ชนิด Increment Rotary Encoder.....	15
2.22 การแปลงและ Reference frame.....	17

2.23 การรวมกันของปริมาณ Vector จากการแปลงทั้งหมด	18
2.24 Forward Transformations ของ Clake และ Park	18
2.25 คลื่น Sine ของ PWM	21
2.26 รูปคลื่นของ SVPWM	22
2.27 ระบบควบคุมป้อนกลับ	23
2.28 ผลของการปรับอัตราขยายที่มีผลตอบสนองเชิงเวลาในการเข้าสู่สภาวะอยู่ตัว	26
2.29 ผลตอบสนองกระบวนการเมื่อใช้ตัวควบคุม PI	27
3.1 ระบบควบคุมมอเตอร์รถยนต์ไฟฟ้า (PMSM)	31
3.2 บล็อก TestSignals	31
3.3 องค์ประกอบภายในบล็อก TestSignals	32
3.4 บล็อก Controller	32
3.5 บล็อก TestChannel	33
3.6 องค์ประกอบภายในบล็อก TestChannel	33
3.7 บล็อก SensorPeripherals	33
3.8 องค์ประกอบภายในบล็อก SensorPeripherals	33
3.9 บล็อก FOC	34
3.10 องค์ประกอบภายในบล็อก FOC	34
3.11 บล็อก thm2the	34
3.12 องค์ประกอบภายในบล็อก thm2the	35
3.13 บล็อก VelocityController	35
3.14 องค์ประกอบภายในบล็อก VelocityController	35
3.15 บล็อก abc2dq	36
3.16 องค์ประกอบภายในบล็อก abc2dq	36
3.17 บล็อก CurrentController	37

3.18	องค์ประกอบภายในบล็อก CurrentController	37
3.19	บล็อก dq2abc.....	38
3.20	องค์ประกอบภายในบล็อก dq2abc	38
3.21	บล็อก DrivePeripherals.....	39
3.22	องค์ประกอบภายในบล็อก DriverPeripherals.....	39
3.23	บล็อก Plant	40
3.24	องค์ประกอบภายในบล็อก Plant	40
3.25	บล็อก Gate Driver	41
3.26	องค์ประกอบภายในบล็อก Gate Driver	41
3.27	บล็อก ElecSensors.....	42
3.28	องค์ประกอบภายในบล็อก ElecSensors	42
3.29	บล็อก MechSensor	43
3.30	องค์ประกอบภายในบล็อก MechSensor	43
3.31	บล็อก MechLoad	44
3.32	องค์ประกอบภายในบล็อก MechLoad.....	44
3.33	บล็อก Evaluation	45
3.34	องค์ประกอบภายในบล็อก Evaluation.....	45
3.35	บล็อก PIDController ที่ใช้สำหรับปรับแต่งสัญญาณ PI	46
3.36	การเลือกวิธีที่ใช้ในการปรับแต่งสัญญาณ PI.....	46
3.37	หน้าต่างโปรแกรม PID Tuner	47
3.38	ส่วนที่ใช้ในการปรับแต่งค่าสัญญาณภายใน PID Tuner	47
4.1	กราฟเปรียบเทียบความเร็วรอบ (RPM) ก่อนปรับค่า PI	49
4.2	กราฟความเร็วรอบ, แรงบิด, แรงดัน และกระแส ก่อนปรับค่า PI.....	49
4.3	กราฟเปรียบเทียบความเร็วรอบ (RPM) หลังปรับค่า PI.....	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 กราฟความเร็วรอบ, แรงบิด, แรงดัน และกระแส หลังปรับค่า PI..... 50



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

รูปที่	หน้า
1.1 ระยะเวลาการดำเนินงาน	2
2.1 สูตรวิธีการปรับจูนพารามิเตอร์ตัวควบคุม PID จากวิธี Open Loop	28
2.2 สูตรวิธีการปรับจูนพารามิเตอร์ตัวควบคุม PID จากวิธี Close Loop	28



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบันเทคโนโลยียานยนต์ไฟฟ้ากำลังได้รับความนิยมอย่างแพร่หลายเป็นอย่างมาก ไม่ว่าจะเป็นตลาดยานยนต์ทั้งในประเทศและต่างประเทศ ซึ่งในปัจจุบันนี้เป็นยุคเปลี่ยนผ่านของเทคโนโลยียานยนต์ จากยานยนต์ที่ใช้น้ำมันเป็นยานยนต์ที่ใช้พลังงานไฟฟ้า โดยผู้จัดทำโครงการได้เล็งเห็นถึงจุดนี้ จึงเกิดความสนใจและมีความประสงค์ที่จะร่วมเป็นส่วนหนึ่งของยุคสมัยที่มีการเปลี่ยนแปลงของเทคโนโลยียานยนต์ที่ใช้น้ำมัน ไปเป็นพลังงานไฟฟ้าและผู้จัดทำโครงการได้มีความสนใจในเรื่องของมอเตอร์รถยนต์ไฟฟ้า ซึ่งเป็นสิ่งที่สำคัญที่สุดของยานยนต์ไฟฟ้า เพื่อที่จะให้ยานยนต์ไฟฟ้านั้นสามารถขับเคลื่อนได้ซึ่งสิ่งนั้นก็คือมอเตอร์ไฟฟ้า ซึ่งเปรียบเสมือนเป็นเครื่องยนต์ของยานยนต์ที่ใช้น้ำมัน และด้วยเหตุนี้จึงเกิดเป็นโครงการจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยพีวซัน เซ็นเซอร์ขึ้น

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

- 1.2.1 เพื่อศึกษาโครงสร้างของมอเตอร์รถยนต์ไฟฟ้า (PMSM)
- 1.2.2 เพื่อศึกษาระบบการควบคุมการทำงานของมอเตอร์รถยนต์ไฟฟ้า (PMSM)
- 1.2.3 เพื่อศึกษาการใช้งาน MATLAB และ Simulink สำหรับงานออกแบบระบบควบคุมมอเตอร์รถยนต์ไฟฟ้า (PMSM)

1.3 ขอบเขตของโครงการ

- 1.3.1 สามารถออกแบบระบบของมอเตอร์รถยนต์ไฟฟ้าได้ (PMSM)
- 1.3.2 สามารถควบคุมมอเตอร์รถยนต์ไฟฟ้า (PMSM) ได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.4.1 เกิดความรู้ความเข้าใจในโครงสร้างของมอเตอร์รถยนต์ไฟฟ้า (PMSM)
- 1.4.2 เกิดทักษะในการออกแบบระบบควบคุมมอเตอร์รถยนต์ไฟฟ้า (PMSM) ด้วยโปรแกรม MATLAB และ Simulink
- 1.4.3 สามารถนำทักษะและความรู้จากการทำโครงการในครั้งนี้ไปใช้ในการทำงานได้ในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ระยะเวลาทำการวิจัยและแผนการดำเนินงานโครงการ

ขั้นตอนการดำเนินงาน	สัปดาห์ที่													
	1	2	3	4	5	6	7	8	9	10	11	12	13	
1. ศึกษาทฤษฎี		←							→	๒๖ เมษายน ๒๕๖๒	←			→
2. เขียนโปรแกรม		←							→		←			→
3. แก้ไขโปรแกรม		←							→		←			→
4. เขียนเล่มโครงการ											←			→

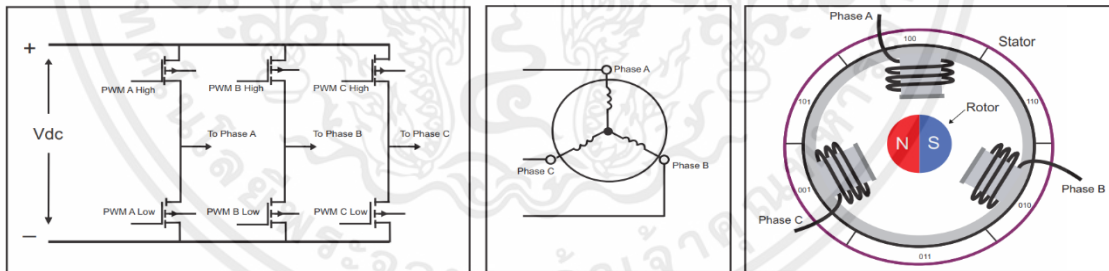
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Permanent-Magnet Synchronous Motor

มอเตอร์ไฟฟ้าชนิดซิงโครนัสแบบใช้แม่เหล็กถาวร (PMSM) เป็นมอเตอร์ไฟฟ้าที่มีเฟสทั้ง 3 เฟสเป็นขดลวดสเตเตอร์และแม่เหล็กถาวรเป็นโรเตอร์ สนามแม่เหล็กของช่องว่างภายในมอเตอร์นั้น มีค่าคงที่เนื่องมาจากแม่เหล็กถาวรภายในมอเตอร์ โดยมอเตอร์กระแสตรงแบบเดิมจะสับเปลี่ยนตัวเองให้ใช้ตัวสับเปลี่ยนแบบเชิงกล แต่ในขณะที่ PMSM ต้องการการสับเปลี่ยนแบบอิเล็กทรอนิกส์ เพื่อควบคุมทิศทางของกระแสภายในขดลวด และเนื่องจากมอเตอร์ PMSM มีขดลวดพันรอบสเตเตอร์อยู่ จึงต้องมีการสับเปลี่ยนกับภายนอกโดยใช้วงจรสวิตชิงจากภายนอกและใช้โทโพโลยีอินเวอร์เตอร์ 3 เฟส เพื่อจุดประสงค์นี้



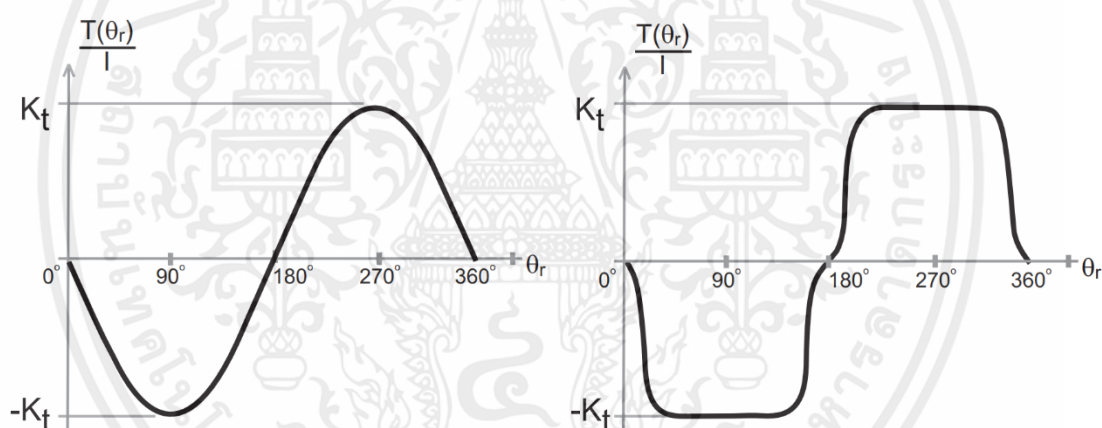
รูปที่ 2.1 PMSM และโทโพโลยีบริดจ์อินเวอร์เตอร์

แรงบิดเกิดขึ้นเนื่องจากการทำงานร่วมกันของสนามแม่เหล็กทั้งสอง จึงทำให้มอเตอร์หมุน โดยในมอเตอร์แม่เหล็กถาวรนั้น สนามแม่เหล็กสนามหนึ่งถูกสร้างขึ้นโดยแม่เหล็กถาวร และอีกสนามหนึ่งนั้นจะถูกสร้างขึ้นโดยขดลวดสเตเตอร์ โดยแรงบิดสูงสุดเกิดขึ้นเมื่อเวกเตอร์แม่เหล็กของโรเตอร์ทำมุม 90 องศากับเวกเตอร์แม่เหล็กของสเตเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเทคนิคของ Block Commutation จะมีเพียงสองขดลวดเท่านั้นที่ทำงานอยู่ ณ เวลาใดเวลาหนึ่ง และขดลวดที่สามยังคงไม่ทำงาน ซึ่งสิ่งนี้จะทำให้เกิดการเยื้องศูนย์ระหว่างสนามแม่เหล็กของสเตเตอร์และตัวโรเตอร์ และด้วยเหตุนี้เองจึงทำให้เกิดริปเปิ้ลของแรงบิด ที่จะทำให้เกิดสัญญาณรบกวน, การสั่นสะเทือนและการสึกหรอ ให้แก่ระบบมากขึ้น

ในเทคนิคของ Sinusoidal Commutation นั้น ขดลวดทั้งสามจะทำงานพร้อมกัน ณ เวลาใดเวลาหนึ่ง ซึ่งกระแสที่ผ่านขดลวดจะเป็นแบบ Sinusoidal และแต่ละเฟสจะมีเฟสต่างกัน 120 องศา ตามลำดับ โดยแรงบิดที่เกิดขึ้นจะมีความนิ่ง กล่าวคือมีริปเปิ้ลของแรงบิดที่น้อยมากๆ



รูปที่ 2.2 การสร้างแรงบิดในมอเตอร์แบบ Sinusoidal และ Trapezoidal

2.2 ฟิวชั่นเซนเซอร์ (Fusion Sensor)

ฟิวชั่นเซนเซอร์เป็นการนำเซนเซอร์หลายๆชนิด นำมาทำงานร่วมกันโดยมีจุดประสงค์ที่ไปในทิศทางเดียวกันยกตัวอย่างเช่น เราต้องการตำแหน่งของวัตถุชนิดหนึ่ง และเพื่อความแม่นยำที่มากยิ่งขึ้น เราก็อาจจะใช้เซนเซอร์หลายๆชนิด ทำให้ตำแหน่งของวัตถุที่เราต้องการนั้นมีความแม่นยำมากยิ่งขึ้น และที่ยกตัวอย่างขึ้นมาี้ คือตัวอย่างนิยามของคำว่าฟิวชั่นเซนเซอร์ โดยจะยกตัวอย่างเซนเซอร์ที่ได้นำมาประกอบรวมกันเป็นฟิวชั่นเซนเซอร์สำหรับโครงการนี้ต่อไป

2.2.1 เซนเซอร์วัดกระแส (Current Sensor)



รูปที่ 2.3 ตัวอย่างลักษณะของเซนเซอร์วัดกระแส

เซนเซอร์วัดกระแส (Current Sensor) คืออุปกรณ์ที่ตรวจจับและแปลงกระแสเป็นแรงดันเอาต์พุตที่วัดได้ง่าย ซึ่งจะเป็นสัดส่วนกับกระแสที่ไหลผ่านทิศทางการวัด และเซนเซอร์วัดกระแสนี้ ก็จะแปลงจากกระแสเป็นแรงดันไฟฟ้า โดยมีการทำงานคือภายในเซนเซอร์จะมีตัวต้านทานวางไว้ระหว่างเส้นทางที่กระแสไฟฟ้าไหลผ่าน กระแสจะถูกเปลี่ยนเป็นแรงดันไฟฟ้าในแบบเชิงเส้น

2.2.2 เซนเซอร์วัดแรงดัน (Voltage Sensor)



รูปที่ 2.4 ตัวอย่างลักษณะของเซนเซอร์วัดแรงดันสำหรับยานยนต์

เซนเซอร์วัดแรงดัน (Voltage Sensor) คือ เซนเซอร์ที่ใช้ในการวัดแรงดันไฟฟ้า ซึ่งมีการนำไปใช้งานที่หลากหลาย โดยเซนเซอร์วัดแรงดันนี้จะมีด้วยกัน 3 ประเภทคือ Low Voltage (LV) – 0 ถึง 49 Volts, Medium Voltage (MV) – 50 ถึง 1000 Volts และประเภทสุดท้ายคือ High Voltage (HV) – 1000 ถึง 4160 Volts โดยอ้างอิงข้อมูลจาก NEC voltage standard และผู้จัดทำก็ได้นำเซนเซอร์วัดแรงดันไฟฟ้านี้ เข้ามาใช้ในโครงการนี้ด้วย โดยเซนเซอร์วัดแรงดันไฟฟ้านี้จะทำหน้าที่ในการแปลงแรงดันไฟฟ้าให้เป็นสัญญาณอนาล็อก

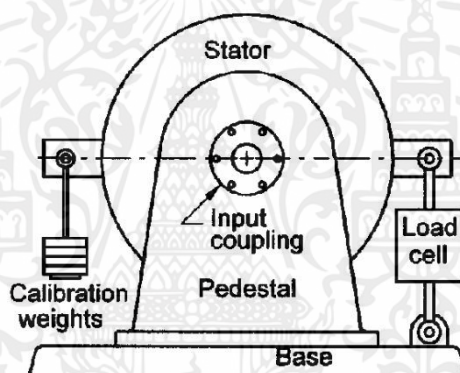
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 เซ็นเซอร์วัดแรงบิด (Torque Sensor)

เซ็นเซอร์วัดแรงบิด (Torque Sensor) ใช้เพื่อตรวจวัดการตรวจวัดแรงบิดบนส่วนประกอบทางกลต่างๆ ที่หมุนหรือไม่หมุนและเซ็นเซอร์วัดแรงบิดจะแปลงการเปลี่ยนแปลงทางกายภาพของแรงบิดให้เป็นสัญญาณไฟฟ้า โดยแรงบิดนี้สามารถวัดได้หลากหลายวิธี ดังนี้

1) Moment Arm

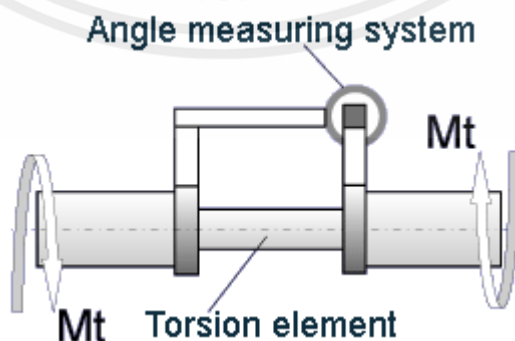
เป็นวิธีการวัดแรงบิดที่ง่ายที่สุด โดยมีหลักการทำงานโดยใช้กฎข้อที่สามของนิวตัน ที่กล่าวว่าทุก ๆ แรงกิริยาจะมีแรงปฏิกิริยาที่ขนาดเท่ากันเสมอ (ซึ่งมีทิศทางตรงกันข้าม) ดังนั้นหากเราจะวัดแรงบิด เช่น แรงบิดของมอเตอร์ หากเราสามารถวัดได้ว่าจะต้องใช้แรงเท่าไรที่จะทำให้มอเตอร์ไม่หมุน (หรืออยู่กับที่) จากแรงบิดที่มากระทำที่มอเตอร์หรือแรงบิดที่มอเตอร์ออกแรงนั้นคือแรงบิด ดังรูปที่ 2.5



รูปที่ 2.5 การวัดแรงบิดแบบ Moment Arm

2) Strain-gaged Shaft

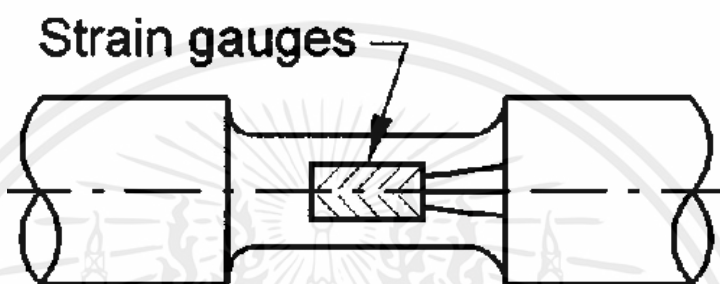
การวัดแรงบิดแบบนี้ทำงานโดยอาศัยหลักของคานบิดเมื่อเกิดแรงบิดมากระทำที่ Torsion Bar นั้น ดังรูปที่ 2.6



รูปที่ 2.6 การวัดแรงบิดโดยอาศัยหลักการของ Torsion Bar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการวัดแรงบิดด้วย Strain-gauged Shaft จะวัดแรงบิดโดยการนำเอา Strain Gage ยึดไว้ที่ตรงกลางของ Torsion Bar การบิดตัวของ Torsion Bar จะทำให้ Strain Gage บิดไปด้วย ดังนั้นความต้านทานของ Strain Gage ก็จะไปตามแรงบิดที่มากระทำกับ Torsion Bar นี้ ลักษณะของ Strain-gauged Shaft แสดงดังรูปที่ 2.7 และรูปที่ 2.8 แสดงตัวอย่างของเซนเซอร์วัดแรงบิด



รูปที่ 2.7 เซนเซอร์วัดแรงบิดแบบ Strain-gauged Shaft



รูปที่ 2.8 ตัวอย่างเซนเซอร์วัดแรงบิดแบบ Strain-gauged Shaft

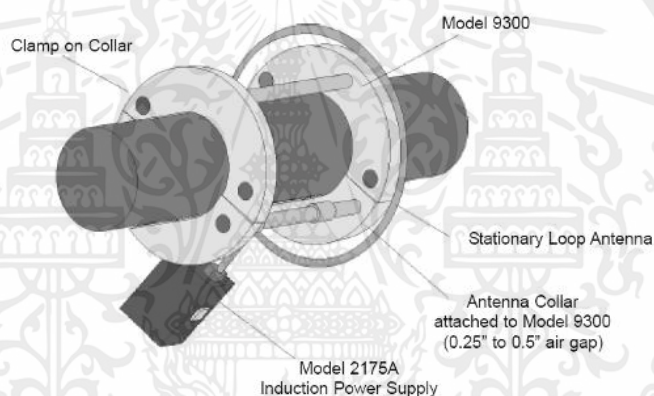
3) Clamp-on Collar

การวัดแรงบิดแบบนี้มีข้อดีตรงที่ว่าไม่ต้องตัดต่อเพลลาหรือแกนที่ต้องการจะวัดแรงบิด โดยวิธีการที่จะวัดแรงบิดแบบนี้จะทำการยึดอุปกรณ์ที่เรียกว่า Clamp-on Collar เข้าไปที่เพลลาหรือแกนที่ต้องการจะวัดแรงบิด และ Clamp-on Collar นี้จะบิดมากหรือน้อยตามแรงบิดที่มากระทำกับแกนนั้น ๆ วิธีนี้ให้ความสะดวกในการติดตั้งเป็นอย่างมาก โดยในรูปที่ 2.9 และ 2.10 แสดงโครงสร้างและการใช้งานของเซนเซอร์วัดแรงบิดแบบ Clamp-on Collar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 ลักษณะของเซนเซอร์วัดแรงบิดแบบ Clamp-on Collar

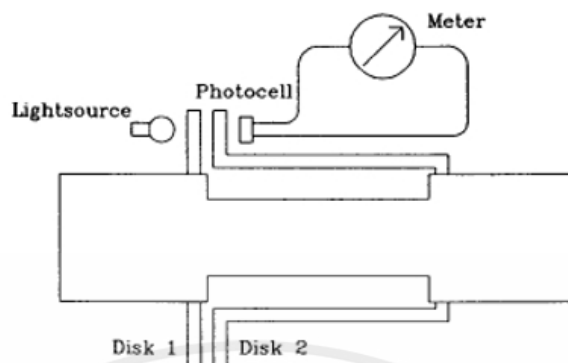


รูปที่ 2.10 ตัวอย่างเซนเซอร์วัดแรงบิดแบบ Clamp-on Collar ชนิดไร้สาย (Wireless)

4) Optical

หลักการวัดแรงบิดโดยใช้แสงหรือ Optical ก็ขึ้นอยู่กับหลักพื้นฐานของ Torsion Bar เช่นเดียวกัน แต่ได้มีการพัฒนาการวัดแรงบิดโดยใช้หลักการของแสงออกไปในหลายรูปแบบ วิธีที่ง่ายที่สุดแสดงดังรูปที่ 2.11 โดยการใช้จาน 2 แผ่น แผ่นที่ 1 ยึดอยู่ที่ด้านซ้ายของ Torsion Bar และแผ่นที่ 2 ยึดอยู่ที่ด้านขวามือของ Torsion Bar งานทั้งสองแผ่นนี้จะถูกเจาะรูหรือที่เราเรียกว่า Slit Disk ในตำแหน่งที่ตรงกัน และมีแหล่งกำเนิดแสงหรือ Light source ทางด้านจานแผ่นที่ 1 มีตัวรับแสงหรือ Photocell ทางด้านจานแผ่นที่ 2 ถ้าหาก Photocell ได้รับแสงจาก Light source จะทำให้เกิดแรงดันที่เอาต์พุตของ Photocell ดังนั้นการหมุนของเพลลาหรือจาน 1 รอบย่อมทำให้เกิดพัลส์หนึ่งลูก โดยพัลส์นี้เกิดจากการที่แสงจาก Light source สามารถผ่านจานทั้งสองไปปรากฏยัง Photocell ได้

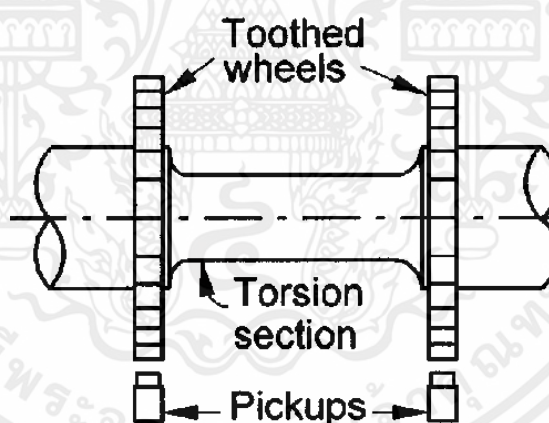
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Optical

5) Toothed Wheel

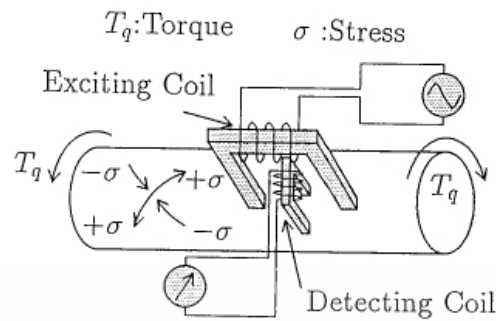
เซนเซอร์แบบนี้ใช้หลักการของคานบิดเช่นเดียวกัน และมีความคล้ายคลึงกันมากๆ ต่างกันที่ว่าวิธีนี้จะใช้วิธีตรวจจับตำแหน่งของซี่ฟันโลหะ หากเกิดแรงบิดมากกระทำที่ Torsion Bar จะทำให้เกิดความต่างเฟสของซี่ฟัน ซึ่งความต่างเฟสนี้ก็จะแปรผันตรงกับแรงบิดที่มากกระทำกับ Torsion Bar นั่นเอง โครงสร้างของเซนเซอร์วัดแรงบิดแบบนี้แสดงดังรูปที่ 2.12



รูปที่ 2.12 เซนเซอร์วัดแรงบิดแบบ Tooth Wheel

6) Magnetostrictive

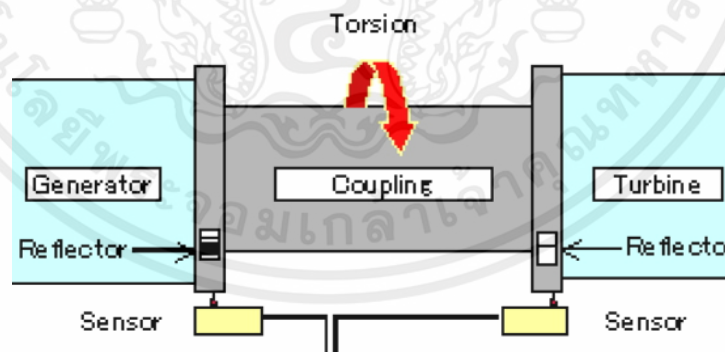
การตรวจจับแรงบิดจากการบิดของแกนหรือเพลลาที่ทำจากวัสดุ Ferromagnetic สามารถใช้หลักการของคุณสมบัติของแม่เหล็กได้เช่นกัน โดยอาศัยหลักการที่ว่า การบิดตัวของแกนที่ทำจากวัสดุ Ferromagnetic ภายใต้สนามแม่เหล็กจะมีผลต่อคุณสมบัติทางแม่เหล็กของแกน โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Magnetostrictive ดังรูปที่ 2.13



รูปที่ 2.13 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Magnetostrictive

7) Laser

เซนเซอร์วัดแรงบิดแบบ Laser นี้ แท้จริงแล้วจะใช้หลักการคล้าย ๆ กับ Toothed Wheel มาก โดยจะใช้แสงเลเซอร์ในการวัดมุมต่างเฟสของ Torsion Bar ทั้งสอง โครงสร้างของเซนเซอร์วัดแรงบิดแบบนี้แสดงดังรูปที่ 2.14 และที่ปลายแต่ละด้านของ Torsion Bar จะมีตัวสะท้อนแสงเลเซอร์ติดอยู่ ดังนั้นแรงบิดที่มากกระทำ Torsion Bar จะทำให้เกิดความต่างเฟสของสัญญาณที่ตรวจจับได้โดย Laser Sensor และเนื่องจาก Laser มีความแคบของลำแสงมาก จึงสามารถตรวจจับความเปลี่ยนแปลงเล็ก ๆ ได้ดี โดยมีบริษัทในประเทศญี่ปุ่นนำเอาหลักการนี้ไปใช้วัดแรงบิดของ Power Plant (วัดแรงบิดระหว่าง Turbine และ Generator) เพราะไม่ต้องการจะตัดแปลงหรือเปลี่ยนระบบ และนอกจากนี้เซนเซอร์แบบ Laser ยังใช้วัดแรงบิดที่เพลลาที่หมุนด้วยความเร็วสูง ๆ ได้ดี

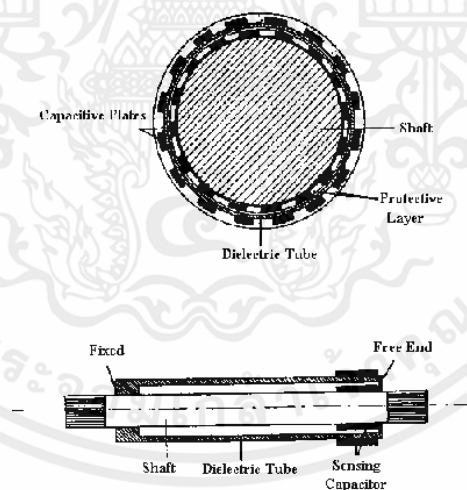


รูปที่ 2.14 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Laser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. Capacitive

เซนเซอร์วัดแรงบิดแบบ Capacitive นี้ พัฒนาขึ้นมาโดยใช้หลักการวัดความจุ โดยมีโครงสร้างดังรูปที่ 2.15 เซนเซอร์วัดแรงบิดแบบนี้จะประกอบไปด้วยซีโลหะสองชุดที่วางในตำแหน่งสลับฟันปลา ชุดแรกจะอยู่ด้านนอกและชุดที่สองจะอยู่ด้านใน โดยจะมีไดอิเล็กทริกที่เป็นท่อทรงกระบอก (Dielectric Tube) กั้นอยู่ตรงกึ่งกลางและซีโลหะทั้งสองชุดจะทำหน้าที่เป็นแผ่นเพลตของตัวเก็บประจุ (Capacitor Plate) ซึ่งจะมีค่าแปรผันตามการตำแหน่งการซ้อนหรือเหลื่อมกันของซีโลหะหรือแผ่นเพลตทั้งสองชุด ดังนั้นหากเกิดแรงบิดมากกระทำที่ Torsion Bar นี้ จะทำให้การเหลื่อมกันของโลหะเปลี่ยนไป (อาจจะมากหรือน้อยลง ตามแต่ว่าแรงบิดนั้นจะเป็นในแนวตามเข็มนาฬิกาหรือทวนเข็มนาฬิกา) ดังนั้นถ้าหากดูตามรูปจะเห็นได้ว่า หากแรงบิดที่มากกระทำ Torsion Bar มากขึ้นในแนวตามเข็มนาฬิกา ดังนั้นซีโลหะชุดด้านในจะเลื่อนตามเข็มนาฬิกา และการซ้อนกันของซีโลหะทั้งสองชุดจะน้อยลง ซึ่งจะทำให้ค่าตัวเก็บประจุมีค่าน้อยลง และหากแรงบิดมากกระทำในแนวทวนเข็มนาฬิกา การซ้อนกันของซีโลหะจะมากขึ้น ทำให้ค่าตัวเก็บประจุมีค่ามากขึ้น ดังนั้นหากเราทราบถึงสมการความสัมพันธ์ของแรงบิดต่อมุมที่มีผลต่อค่าตัวประจุ เราก็จะทราบค่าแรงบิดได้โดยง่ายจากการวัดค่าตัวเก็บประจุนั่นเอง



รูปที่ 2.15 โครงสร้างของเซนเซอร์วัดแรงบิดแบบ Capacitive

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 เซนเซอร์วัดตำแหน่งภายในมอเตอร์ (Encoder)



รูปที่ 2.16 ตัวอย่าง Linear Encoder

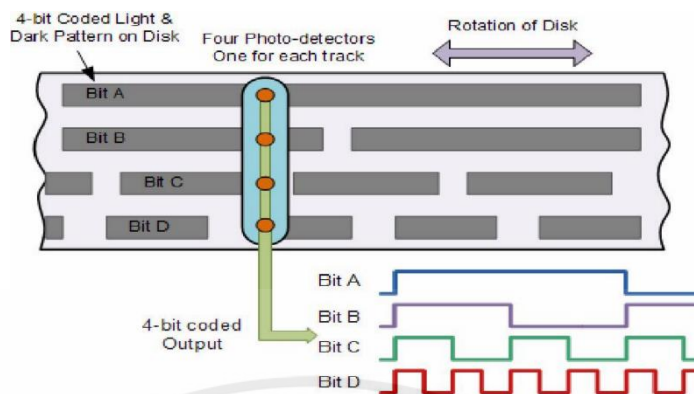
เอ็นโคเดอ (Encoder) เป็นเซนเซอร์ชนิดหนึ่ง ที่สามารถนำมาประยุกต์ใช้ในการวัดตำแหน่งหรือระยะขจัด (Position or Displacement Sensor) มีด้วยกัน 2 ลักษณะ คือ เป็นแบบเชิงเส้น (Linear Encoder) และเป็นแบบหมุน (Rotary Encoder) และในแต่ละลักษณะยังแบ่งออกเป็นแบบที่สัญญาณเอาต์พุตที่เกิดจากการวัดอยู่ในรูปแบบของรหัสเชิงตรรกะ (Digital coding signal) ซึ่งจะเรียกเอ็นโคเดอในรูปแบบนี้ว่า “Absolute Encoder” ส่วนอีกลักษณะหนึ่งจะเป็นแบบที่ให้สัญญาณเอาต์พุตที่เกิดจากการวัดอยู่ในรูปแบบของพัลส์ (Pulse Signal) ซึ่งมักจะถูกเรียกว่า “Incremental Encoder”

1) เอ็นโคเดอแบบเชิงเส้น (Linear Encoder)

เอ็นโคเดอแบบเชิงเส้นใช้วัดตำแหน่งหรือระยะทางในแนวที่เป็นเส้นตรง สัญญาณเอาต์พุตที่เกิดจากการวัดมีสองรูปแบบ คือ Absolute Linear และ Increment Linear Encoder

Absolute Linear Encoder

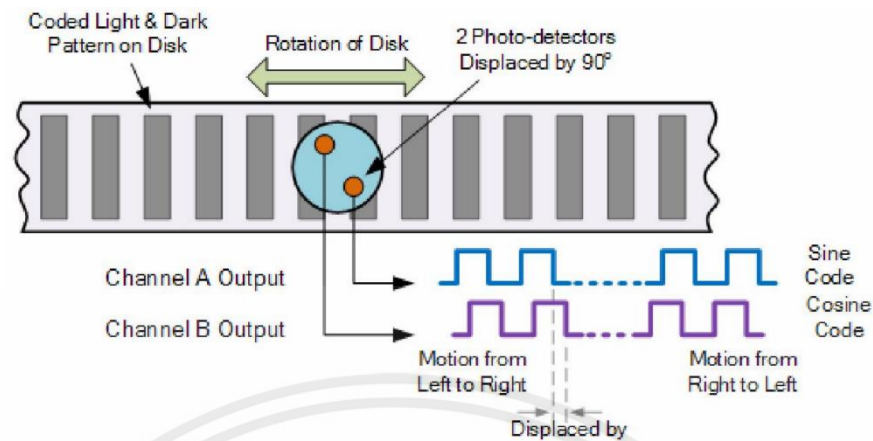
หลักการทำงานของเอ็นโคเดอแบบนี้ดังแสดงในรูปที่ 2.17 โดยจำนวนแถบรหัส (Digital bits coding) จะถูกกำหนดให้อยู่ในแนวเชิงเส้นของการเคลื่อนที่ โดยมีหัวอ่านที่สร้างขึ้นจากเซนเซอร์แสง (Photo-detector) เท่ากับจำนวนแถบรหัส และเคลื่อนที่ไปในแนวเส้นตรงของการเคลื่อนที่ ทำการอ่านรหัสในขณะที่เคลื่อนที่ไป ข้อดีของเอ็นโคเดอแบบนี้ ก็คือเมื่อเปิดเครื่อง หัวอ่านจะรับรู้ถึงตำแหน่งที่ทำการวัดอยู่ในขณะนั้นได้โดยทันทีแต่มีข้อเสียที่ ค่าความละเอียดที่ถูกต้องและแม่นยำ จะขึ้นอยู่กับจำนวนแถบรหัส และสิ้นเปลืองจำนวนเซนเซอร์ที่ต้องใช้ในการอ่านรหัสของทุกแถบ ซึ่งทำได้ยากและมีราคาแพงและยากต่อการบำรุงรักษา



รูปที่ 2.17 โครงสร้างของ Absolute Linear Encoder

Increment Linear Encoder

หลักการทำงานของเอ็นโคเดอร์แบบ Increment Linear Encoder ดังรูปที่ 2.18 แถบเส้นตรงเล็กๆ จะถูกวางในแนวตั้งฉากกับทิศทางการเคลื่อนที่ โดยมีหัวอ่านสร้างขึ้นจากเซนเซอร์แสง 2 ตัว วางทำมุมกัน ทำให้ในขณะที่หัวอ่านเคลื่อนที่ไป จะทำให้สัญญาณที่ออกมาทางเอาต์พุตในรูปแบบของพัลส์ (Pulse) จะมีเฟสเอียงกัน 90 องศา ซึ่งมีไว้สำหรับตรวจสอบทิศทางการเคลื่อนที่ของหัวอ่าน ดังนั้นความละเอียดและแม่นยำ จึงขึ้นอยู่กับจำนวนเส้นตรงเล็กๆ ที่ถูกวางในแนวตั้งฉากกับทิศทางการเคลื่อนที่ ซึ่งมักจะบอกในรูปแบบของจำนวนของจำนวนพัลส์ที่ได้ต่อระยะการเคลื่อนที่ เช่น 25 Pulse/10 mm. เอ็นโคเดอร์แบบนี้จะใช้หลักการนับจำนวนของแถบเส้นตรงเล็กๆ ดังนั้นการสร้างเอ็นโคเดอร์แบบนี้จึงทำได้ง่าย ราคาถูกกว่าเอ็นโคเดอร์แบบเชิงเส้น และต้องการบำรุงรักษาน้อยกว่า แต่ข้อเสียของเอ็นโคเดอร์ประเภทนี้ก็คือ ต้องใช้วงจรนับความเร็วสูง (High Speed Counter) ทำการนับแถบเส้นตรงเล็กๆ ถ้าไฟดับขณะเคลื่อนที่ ค่าการรับจะถูกกลับไป ทำให้ไม่สามารถที่จะระบุตำแหน่งที่อยู่ขณะนั้นได้ ดังนั้นเมื่อเริ่มต้น หัวอ่านจะถูกส่งให้ส่งกลับมาเริ่มต้นที่ตำแหน่งศูนย์ (Zero) ของการเคลื่อนที่ทุกครั้ง เพื่อทำการรีเซ็ต (Reset) ค่าของการนับก่อนทุกครั้งไป นอกจากนี้ ยังเกิดค่าความคลาดเคลื่อนสะสมที่เกิดจากการนับขึ้น-นับลง ในขณะที่เคลื่อนที่ ไป-มาได้อีกด้วย



รูปที่ 2.18 โครงสร้างของ Increment Linear Encoder

2) เอ็นโคเดอร์แบบหมุน (Rotary Encoder)



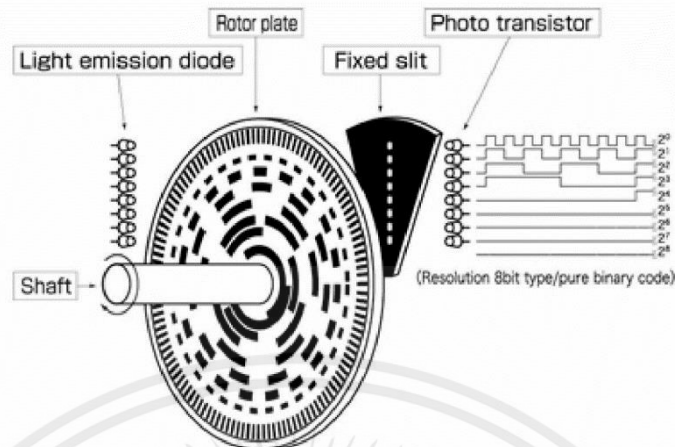
รูปที่ 2.19 ตัวอย่าง Rotary Encoder

ใช้วัดตำแหน่งมุมที่เกิดจากการหมุนที่แกนเพลลาของเอ็นโคเดอร์ สัญญาณเอาต์พุตที่เกิดจากการวัด มีสองรูปแบบ คือ Absolute Rotary Encoder และ Increment Rotary Encoder

Absolute Rotary Encoder

หลักการทำงานของเอ็นโคเดอร์ในรูปที่ 2.20 โดยจำนวนแอมบรหัส (Digital bits coding) จะถูกกำหนดให้รอบซ้อนกันในแนวของการหมุน โดยมีหัวอ่านที่สร้างขึ้นจากเซนเซอร์แสง (Photo-detector) เท่ากับจำนวนแอมบรหัส ทำการอ่านรหัสในขณะที่เคลื่อนที่ไป โดยหลักการทำงานจะคล้ายคลึงกันกับ Absolute Linear Encoder โดยเปลี่ยนจากทิศทางการเคลื่อนที่เชิงเส้นเป็นการหมุน

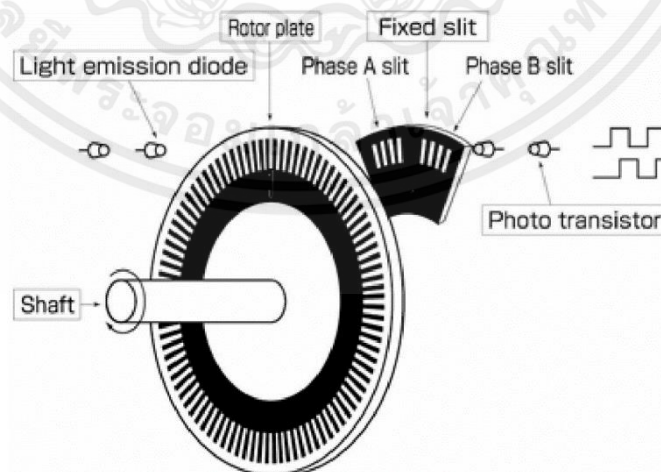
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 โครงสร้างของเอ็นโคเดอร์แบบหมุน ชนิด Absolute Rotary Encoder

Increment Rotary Encoder

หลักการทำงานของเอ็นโคเดอร์แบบ Increment Rotary Encoder คือ แฉกเส้นตรงเล็กๆ จะถูกวางในแนวตั้งฉากกับทิศทางการหมุน โดยมีหัวอ่านสร้างขึ้นจากเซนเซอร์แสง 2 ตัว วางทำมุมกัน ทำให้ในขณะที่จานหมุน (Rotor plate) หมุนไป จะทำให้สัญญาณที่ออกมาทางเอาต์พุตในรูปแบบของพัลส์ (Pulse) มีเฟสเอียงกัน 90 องศา ซึ่งมีไว้สำหรับตรวจสอบทิศทางการหมุนที่เพลาของเอ็นโคเดอร์ ดังนั้นความละเอียดและแม่นยำ จึงขึ้นอยู่กับจำนวนเส้นตรงเล็กๆ ที่ถูกวางในแนวตั้งฉากกับทิศทางการหมุน ซึ่งมักจะบอกในรูปแบบของจำนวนพัลส์ที่ได้ต่อระยะการเคลื่อนที่ เช่น 100 Pulse/rev โดยหลักการทำงานจะคล้ายคลึงกันกับ Incremental Linear Encoder โดยเปลี่ยนจากทิศทางการเคลื่อนที่เป็นเชิงเส้นเป็นการหมุน



รูปที่ 2.21 โครงสร้างของเอ็นโคเดอร์แบบหมุน ชนิด Increment Rotary Encoder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Field Oriented Control Theory

ในมอเตอร์กระแสตรง กระแสที่ผลิตฟลักซ์และแรงบิดนั้น จะเป็นมุมฉากและเราสามารถควบคุมได้อย่างอิสระ อีกทั้งกระแสเหล่านี้จะทำให้เกิดแรงเคลื่อนแม่เหล็ก (Magneto motive force : MMF) ที่เป็นมุมฉากเช่นกัน โดยแรงบิดที่เกิดขึ้นสามารถหาได้จากสมการ

$$T_e = K_a \phi (I_f) I_a$$

เมื่อ

$\phi (I_f)$ คือ ฟลักซ์

I_a คือ กระแสภายในขดลวดสเตเตอร์

ดังนั้น ฟลักซ์จะขึ้นอยู่กับกระแสภายในขดลวดของสนามแม่เหล็กเท่านั้น หากฟลักซ์มีค่าคงที่ แรงบิดจะถูกควบคุมกระแสจากภายในขดลวดสเตเตอร์ ด้วยเหตุนี้เอง จึงกล่าวได้ว่ามอเตอร์กระแสตรงนั้น มีอิสระในการควบคุมแรงบิดและฟลักซ์ แต่ในมอเตอร์กระแสสลับสนามแม่เหล็กของโรเตอร์และสเตเตอร์จะไม่ทำมุมฉากต่อสิ่งใด โดยกระแสเดียวที่สามารถควบคุมมันได้คือกระแสที่อยู่ในสเตเตอร์และ Field Oriented Control เป็นเทคนิคที่ใช้ในการแยกส่วนการควบคุมของแรงบิดและฟลักซ์ โดยสามารถแยกส่วนด้วยการแปลงปริมาณกระแสที่อยู่ในสเตเตอร์ (กระแสเฟส) จาก Stationary Reference frame ไปเป็นส่วนประกอบของ Torque producing และ Flux producing ใน Rotating reference frame

2.3.1 ข้อดีของ FOC

- การแปลงโมเดล AC ที่ซับซ้อนและมีการเชื่อมต่อให้เป็นระบบเชิงเส้น
- การควบคุมแรงบิดและฟลักซ์ได้อย่างอิสระ คล้ายกับมอเตอร์กระแสตรง
- มีการตอบสนองไดนามิกที่รวดเร็วและประสิทธิภาพของ Transient และ steady state ที่ดี
- ประสิทธิภาพสูง

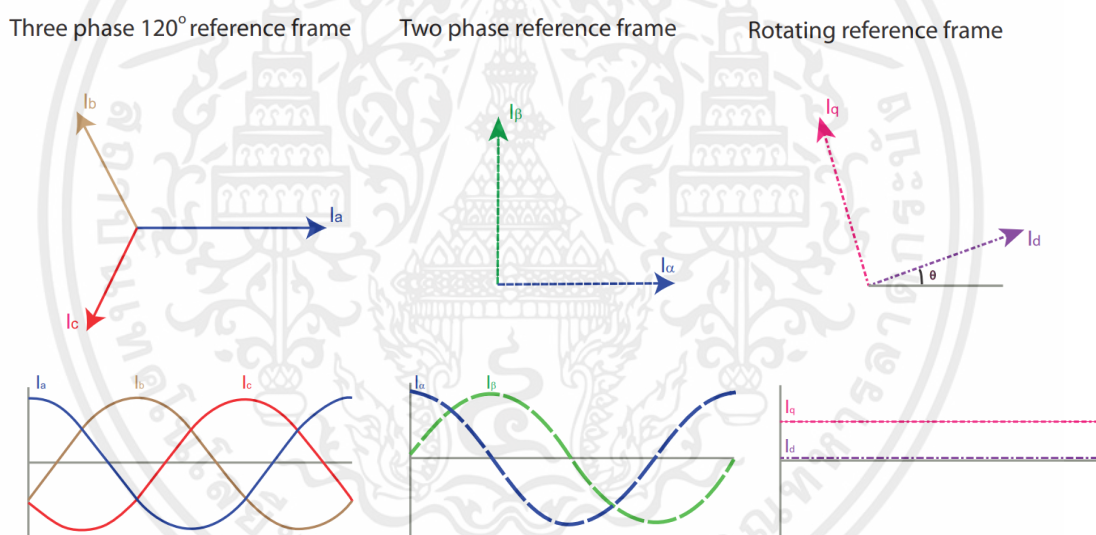
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 เทคนิคของ Field Oriented Control ที่เกี่ยวข้องกับ Three reference frame และการแปลง Reference frame จากแบบหนึ่งเป็นอีกแบบหนึ่ง

2.3.2.1 Stator reference frame (a,b,c) ซึ่ง a,b,c นี้ จะเป็นระนาบร่วม ทำมุมห่างกัน 120 องศา กับมุมอื่นๆ ตามลำดับ

2.3.2.2 Orthogonal reference frame (ab) ในระนาบเดียวกันกับ Stator reference frame ซึ่งมุมระหว่างแกนทั้งสองคือ 90 องศา แทนที่จะเป็น 120 องศา โดยแกน a อยู่ในแนวเดียวกับแกนในเฟรมที่สอง

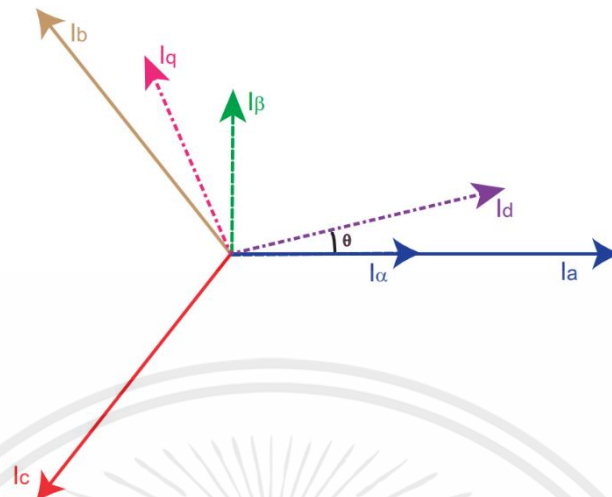
2.3.2.3 Rotor reference frame (dq) ซึ่งแกน d จะอยู่ใกล้กับโพล N และ S หรือ Flux vector ของ Rotor และแกน q จะทำมุม 90 องศา กับแกน d



รูปที่ 2.22 การแปลงและ Reference frame

จากรูปที่ 2.22 เป็นการแปลงเพื่อใช้สำหรับ Decoupling ของกระแส stator สู่อุปกรณ์สร้าง Torque producing (I_q) และ Flux producing (I_d)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

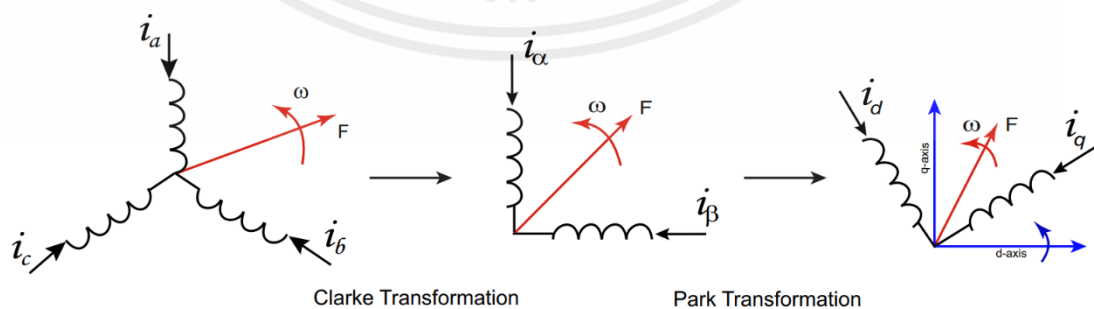


รูปที่ 2.23 การรวมกันของปริมาณ Vector จากการแปลง

จากรูปที่ 2.23 จะเป็นการรวมกันของปริมาณ Vector จากการแปลงที่กล่าวมาทั้งหมดให้อยู่ใน Reference frame เดียว โดยเมื่อส่วนประกอบของ Torque producing และ Flux producing จะถูกควบคุมด้วยตัวควบคุม PI โดยจะได้เอาต์พุตซึ่งเป็นแรงดันไฟฟ้านั้น จะถูกแปลงกลับ (inverse transformation) เป็น Stator reference frame

2.4 Transformation

ใน Field Oriented Control นั้น ส่วนประกอบ i_q และ i_d จะเป็น Reference สำหรับ Rotating reference frame โดยเมื่อทำการวัดกระแส Stator ที่ใช้สำหรับแปลงจาก three phase ที่แปรผันตามเวลาของ Stator reference frame ไปเป็น 2 แกน Rotating-dq Rotor reference frame ซึ่งกระบวนการนี้มี 2 ขั้นตอน ดังรูปที่ 2.24



รูปที่ 2.24 Forward Transformations ของ Clarke และ Park

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 2.24 เป็นการแปลงจาก Three Phase ที่ 120 องศา Reference frame เป็น ลักษณะ 2 แกน Orthogonal reference frame หรือรู้จักกันในอีกชื่อหนึ่งว่า Clarke transform และ ในทำนองเดียวกันนั้น การแปลงจาก Orthogonal reference frame แบบ 2 แกน Rotating reference frame แบบ 2 แกน ก็รู้จักกันในชื่อ Park transform

2.4.1 การแปลง Clarke Transform

ในการวัดกระแสที่มอเตอร์จะแปลงจาก Three phase reference frame เป็น 2 แกน Orthogonal reference frame ซึ่งการแปลงนั้นจะแสดงได้จากสมการดังนี้

$$I_\alpha = \frac{2}{3}(I_a) - \frac{1}{3}(I_b - I_c)$$

$$I_\beta = \frac{2}{\sqrt{3}}(I_b - I_c)$$

เมื่อ I_a, I_b และ I_c คือ ปริมาณ Three-phase

I_α, I_β คือ ปริมาณ Stationary orthogonal reference frame

ถ้า I_α ถูกแทนที่ด้วย I_a และ $I_a + I_b + I_c = 0$ จะทำให้ I_a, I_b และ I_c สามารถแปลงเป็น I_α, I_β ได้ โดยจะได้สมการดังนี้

$$I_\alpha = I_a$$

$$I_\beta = \frac{I_a + 2I_b}{\sqrt{3}}$$

เมื่อ $I_a + I_b + I_c = 0$

2.4.2 การแปลง Park Transform

สำหรับปริมาณของ Orthogonal stationary reference frame แบบ 2 แกน จะถูกแปลงเป็น ปริมาณของ Rotating reference frame ซึ่งการแปลงนั้นจะแสดงได้จากสมการดังนี้

$$I_d = I_\alpha \cos \theta + I_\beta \sin \theta$$

$$I_q = I_\beta \cos \theta - I_\alpha \sin \theta$$

เมื่อ I_d, I_q คือ ปริมาณ Rotating reference frame

I_α, I_β คือ ปริมาณ Orthogonal stationary reference frame

θ คือ มุมในการหมุน

2.4.3 การแปลง Inverse Park Transform

ในตอนนี้เอาต์พุตของตัวควบคุม PI นั้น จะควบคุมด้วยส่วนประกอบของแรงดันไฟฟ้าภายใน Rotating reference frame ด้วยเหตุนี้ Inverse ของกระบวนการก่อนหน้านี้จะถูกนำกลับมาใช้ประโยชน์อีกครั้งเพื่อให้ได้มาซึ่งคลื่นแรงดันอ้างอิง ใน Stationary reference frame และในส่วนแรกนั้น ปริมาณภายใน Rotating reference frame จะถูกแปลงเป็น Orthogonal stationary reference frame แบบ 2 แกน โดยใช้ Inverse Park transform ซึ่งการแปลง Inverse Park transform นั้นจะแสดงได้จากสมการดังนี้

$$\begin{aligned}V_\alpha &= V_d \cos \theta - V_q \sin \theta \\V_\beta &= V_q \cos \theta + V_d \sin \theta\end{aligned}$$

เมื่อ V_α, V_β คือ ปริมาณ Orthogonal stationary reference frame

V_d, V_q คือ ปริมาณ Rotating reference frame

2.4.4 การแปลง Inverse Clarke Transform

การแปลงจาก Orthogonal stationary reference frame แบบ 2 แกน เป็น Three phase-stator stationary reference frame จะทำได้โดยใช้ Inverse Clarke transform ซึ่งการแปลง Inverse Clarke transform นั้นจะแสดงได้จากสมการดังนี้

$$\begin{aligned}V_a &= V_\alpha \\V_b &= \frac{-V_\alpha + \sqrt{3} \cdot V_\beta}{2} \\V_c &= \frac{-V_\alpha - \sqrt{3} \cdot V_\beta}{2}\end{aligned}$$

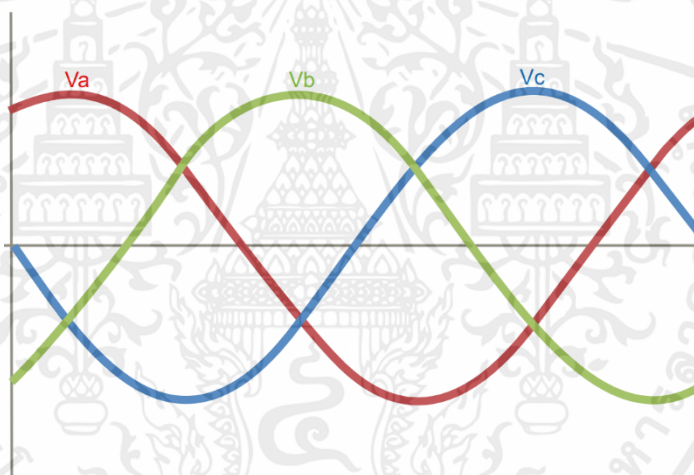
เมื่อ V_a, V_b และ V_c คือ ปริมาณของ Three-phase

V_α, V_β คือ ปริมาณ Stationary orthogonal reference frame

2.5 Sinusoidal Voltage และ SVPWM Generation

เอาต์พุตของการแปลง Inverse Clarke transform จะทำให้ค่า Duty cycle ของช่องต่างๆ ใน Pulse-width modulation (PWM) สอดคล้องกันกับ แรงดัน Three-phase สำหรับการให้ Sinusoidal ทำงานด้วยแรงดันเฟสนั้น ค่า Duty cycle เหล่านี้จะต้องถูกใช้งานโดยตรง ซึ่งมันมีหลากหลายวิธีการที่เราใช้งานกันตามปกติ ที่จะสามารถดำเนินการใช้งาน SVPWM อัลกอริทึมได้ โดยวิธีการอย่างง่ายที่เทียบเท่ากับการ Modulate แบบธรรมดา ที่ใช้กันอยู่ในปัจจุบัน

และในวิธีการนี้ ค่าเฉลี่ยของค่าต่ำสุดและสูงนับช่วงเวลาของของแรงดัน Three-phase ทั้งหมด จะถูกคำนวณเป็นแรงดัน Offset ดังนั้น แรงดัน Offset ช่วงเวลานี้ จะถูกขจัดออกไปจากแรงดัน Three-phase ช่วงเวลา



รูปที่ 2.25 คลื่น Sine ของ PWM

สำหรับ SVPWM (คลื่น Sine กับ 3rd harmonic injection) ซึ่งมีขั้นตอนดังนี้

$$V_{\text{off}} = \frac{[\min(V_a, V_b, V_c) + \max(V_a, V_b, V_c)]}{2}$$

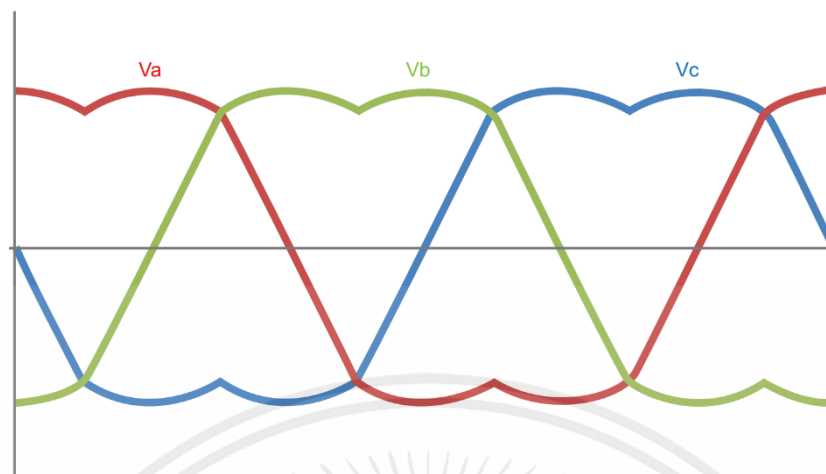
$$V_{a_{\text{new}}} = V_a - V_{\text{off}}$$

$$V_{b_{\text{new}}} = V_b - V_{\text{off}}$$

$$V_{c_{\text{new}}} = V_c - V_{\text{off}}$$

และในรูปที่ 2.26 จะแสดงรูปคลื่นสุดท้ายของแรงดันไฟฟ้า SVPWM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 รูปคลื่นของ SVPWM

2.6 ตัวควบคุมพีไอดี (PID Controller)

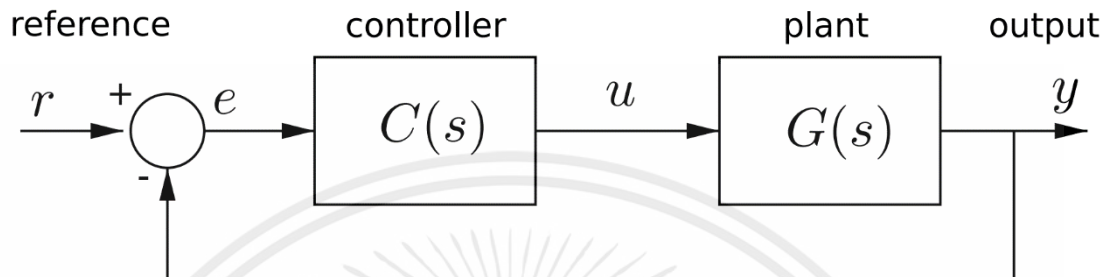
สำหรับในเรื่องของการควบคุมพีไอดีนั้น เราจะพิจารณาลักษณะสมบัติต่างๆ 3 อย่างดังนี้ 1. เสถียรภาพ (Stability) 2. ผลตอบสนองชั่วคราว (Transient response) และ 3. ความคลาดเคลื่อนในสถานะอยู่ตัว (Steady state error)

สิ่งสำคัญอันดับแรกของระบบควบคุมก็คือเสถียรภาพ เพราะถ้าระบบไม่มีเสถียรภาพเราก็ไม่สามารถควบคุมระบบได้ สำหรับลักษณะผลตอบสนองนั้น เรามักต้องการให้ได้ผลตอบสนองที่เร็ว นั่นคือ ช่วงเวลาขึ้น (Rise time) ช่วงเวลาเข้าที่ (Settling time) ส่วนพุ่งเกิน (Overshoot) ที่ยอมรับได้ ส่วนความคลาดเคลื่อนในสถานะอยู่ตัวนั้นต้องให้มีค่าเป็นศูนย์ หรือต่ำที่สุดเท่าที่สามารถทำได้ ลักษณะอื่นๆ นอกเหนือจากนี้จะระบุเฉพาะในแต่ละระบบ

ในระบบควบคุมที่พลานต์ (Plant) ได้ถูกกำหนดแล้ว เราไม่สามารถเปลี่ยนแปลงค่าพารามิเตอร์ภายในเพื่อปรับปรุงระบบเพื่อให้ทำงานได้ตามต้องการ ดังนั้นในขั้นแรก เราอาจปรับค่าอัตราขยายของวงรอบการควบคุม ในเชิงปฏิบัติมักพบว่า การปรับอัตราขยายอย่างเดียวไม่เพียงพอที่จะทำให้ระบบมีลักษณะตามต้องการได้ เช่นถ้าเราเพิ่มอัตราขยายจะทำให้ความคลาดเคลื่อนในสถานะอยู่ตัวลดลง แต่ก็อาจมีผลทำให้ตอบสนองเลวลง การแก้ไขในกรณีเช่นนี้ทำได้โดยออกแบบตัวควบคุม (Controller) เสริมเข้าไปในระบบ

2.6.1 คุณสมบัติของตัวควบคุมแบบพีไอดี

ตัวควบคุมในระบบควบคุมมีหลายชนิด แต่ที่ใช้ในระบบควบคุมกระบวนการส่วนใหญ่ เป็นตัวควบคุมแบบ PID



รูปที่ 2.27 ระบบควบคุมป้อนกลับ

เมื่อพิจารณา รูปที่ 2.27 เรากำหนดให้ $R(s)$ คือสัญญาณอ้างอิง $Y(s)$ คือ สัญญาณคลาดเคลื่อนออกจากค่าอ้างอิง

$$E(s) = R(s) - Y(s)$$

และ $U(s)$ คือ สัญญาณควบคุมที่ออกจากตัวควบคุม เมื่อใช้ในตัวควบคุม PID เราจะได้ว่า

$$C(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

ดังนั้น สัญญาณควบคุมสามารถบรรยายในโดเมนเวลาได้ดังนี้

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right)$$

โดยตัวควบคุม PID ประกอบไปด้วยเทคนิคการควบคุมพื้นฐาน 3 แบบ คือ แบบสัดส่วน (Proportional หรือ P) แบบอินทิกรัล (Integral หรือ I) และแบบอนุพันธ์ (Derivative หรือ D) แต่ละแบบนั้น สามารถนำมาประกอบกันเพื่อให้ได้ตัวควบคุมที่ต้องการ ตัวควบคุมมีพารามิเตอร์ 3 ตัว คือ ค่าอัตราขยายแบบสัดส่วน (K_p) ค่า Integral time (T_i) และ derivative time (T_d) รายละเอียดของการควบคุมพื้นฐานในแต่ละแบบมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1.1 Proportional Action (P) เป็นเทคนิคที่ง่ายที่สุด หลักการคือการนำสัญญาณควบคุมจากตัวควบคุมป้อนเข้ากระบวนการ มีค่าเป็นสัดส่วนกับความผิดพลาด (error = setpoint - output) ที่เกิดขึ้น หรือกล่าวอีกนัยหนึ่ง Proportional Action ทำหน้าที่เป็นอัตราขยายเขียนแทนด้วยสัญลักษณ์ K_p

การควบคุมเชิงสัดส่วนนี้สามารถควบคุมระบบได้ดีระดับหนึ่ง เหมาะสมกับกระบวนการที่ต้องการผลตอบสนองรวดเร็วในขณะเดียวกันก็ยอมให้เกิดความคลาดเคลื่อนขนาดคงที่ค่าหนึ่ง ถ้ามีการเปลี่ยนแปลงสภาวะการทำงานหรือพารามิเตอร์บางตัวในระบบ ก็อาจทำให้เกิดปัญหาขึ้นได้ เช่น ความคลาดเคลื่อนในสภาวะอยู่ตัว (Steady state error) หรือที่เรียกว่า “Offset” ตัวควบคุมแบบ P ไม่สามารถแก้ไขให้หมดไปได้ แต่สามารถลดผลได้ โดยเพิ่มอัตราขยายของตัวควบคุมเพื่อเพิ่มขนาดของสัญญาณควบคุม ทำให้ความคลาดเคลื่อนมีค่าน้อยลง อย่างไรก็ตาม หากเพิ่มอัตราขยายของตัวควบคุมมากเกินไปก็อาจจะส่งผลให้ผลการตอบสนองมีการแกว่งได้ เนื่องจากระบบจะมีความอ่อนไหวต่อการเปลี่ยนแปลง

2.6.1.2 Integral Action (I) สัญญาณควบคุมแบบ PI สามารถอธิบายได้ด้วยสมการ

$$U(s) = K_p \left(1 + \frac{1}{T_i s} \right) E(s) = \left(K_p + \frac{K_i}{s} \right) E(s)$$

เมื่อ T_i คือ Integral time (วินาที) วิธีนี้จะสามารถแก้ความคลาดเคลื่อนในสภาวะอยู่ตัวได้ เนื่องจากเสมือนเพิ่มขั้วที่จุดกำเนิดให้กับระบบวงเปิด แต่ก็ทำให้ระบบมีเสถียรภาพสัมพัทธ์ลดลง โดยทั่วไปแล้วระบบที่ใช้ตัวควบคุมแบบ PI จะมีช่วงเวลาการแกว่งนานกว่าช่วงเวลาที่ใช้ตัวควบคุมแบบสัดส่วน 50% หรือก็คือ

$$T_{PI} = 1.5T_p$$

เมื่อ T_{PI} คือ ช่วงเวลาในการแกว่งของระบบที่ใช้ตัวควบคุมแบบ T_{PI} และ T_p คือ ช่วงเวลาในการของระบบที่ใช้ตัวควบคุมแบบ P

อย่างไรก็ตาม สำหรับระบบที่มีค่าคงตัวของเวลา (Time constant) น้อยปัญหานี้จะไม่มีผลมากนักแต่สำหรับระบบที่มีค่าคงตัวของเวลามาก ปัญหานี้จะใหญ่มากและอาจทำให้ระบบเข้าสู่จุดวิกฤติที่ไม่สามารถยอมรับได้ เช่น ระบบควบคุมดับ

2.6.1.3 Derivative Action (D) การควบคุมแบบสัดส่วนและอินทิกรัลต่างก็มีข้อจำกัด และอาจทำให้เกิดปัญหาต่อการควบคุมกระบวนการเนื่องจากการแกว่งมาก อย่างไรก็ตาม ปัญหาดังกล่าวสามารถแก้ไขได้ด้วยการเพิ่มตัวอนุพันธ์ สัญญาณควบคุมของตัวควบคุมแบบ PID สามารถอธิบายได้ดังสมการ

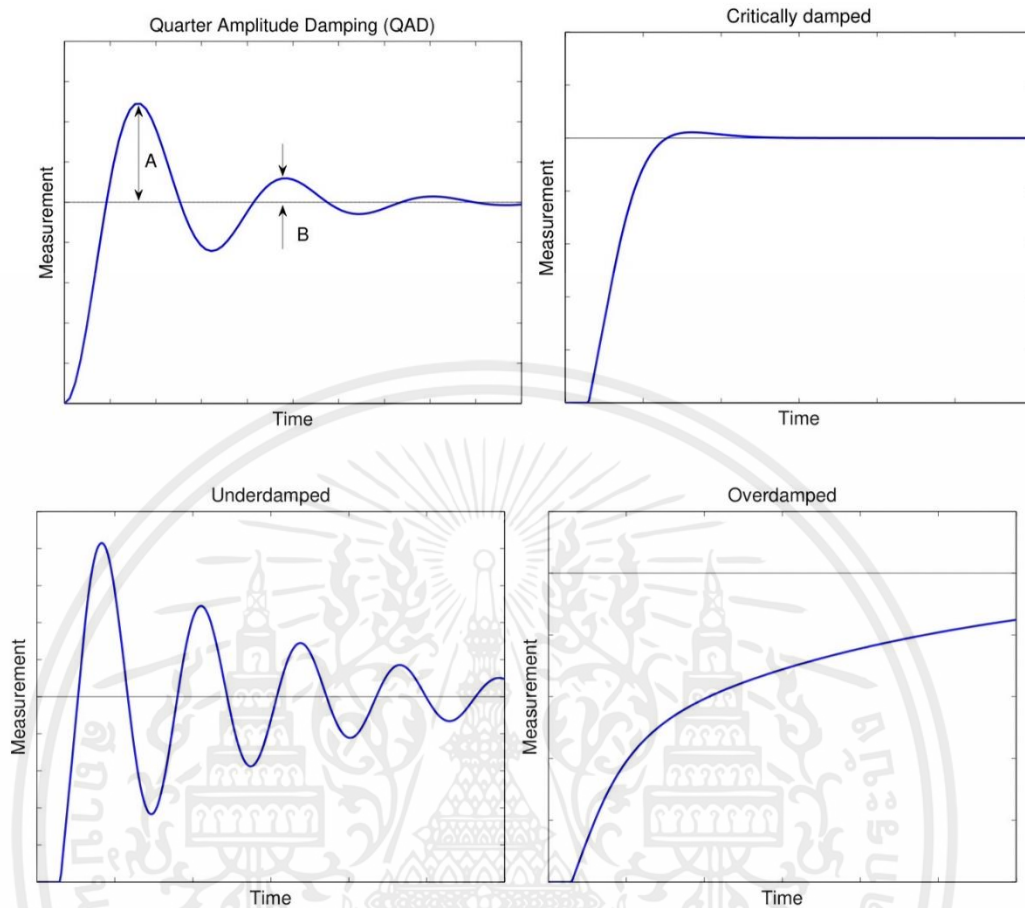
$$U(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) E(s) = \left(K_p + \frac{K_i}{s} + K_d s \right) E(s)$$

ในทฤษฎีระบบควบคุม Derivative action ทำหน้าที่เสมือนกับการเพิ่มศูนย์ (Zero) หนึ่งตัวให้กับระบบ ทำให้ระบบมีเสถียรภาพสัมพัทธ์ดีขึ้น อย่างไรก็ตาม การเพิ่ม K_d มากเกินไปก็อาจจะทำให้ระบบมีการตอบสนองที่ช้า

2.6.2 จุดประสงค์ในการปรับจูนตัวควบคุม

หลังจากการวัดคาบเวลาทางธรรมชาติแล้ว การปรับจูนตัวควบคุม (Controller tuning) ก็สามารทำได้ด้วยวิธี Ziegler-Nichols หรือ Shinskey หรือวิธีอื่นๆ วัตถุประสงค์ก็เพื่อจะหาตัวควบคุมที่ทำให้ผลรวมของความคลาดเคลื่อนในช่วงเวลาทำงานมีค่าต่ำสุด ในทางปฏิบัติการปรับจูนที่เหมาะสมวิธีหนึ่งคือ การปรับจูนให้ได้ผลตอบที่มีลักษณะการหน่วงแอมพลิจูดหนึ่งในสี่ (Quarter amplitude damping, QAD) ในรูปที่ 2.28 ที่แสดงลักษณะสมบัติของ QAD สัดส่วนของส่วนที่พุ่งเกินที่สอง B ต่อส่วนที่พุ่งเกินที่หนึ่ง A คิดเป็น 1 : 4 หรือ ผลตอบสนองลักษณะนี้เกี่ยวข้องโดยตรงกับผลตอบที่มีความคลาดเคลื่อนต่ำสุด

ในบางกระบวนการไม่อาจยอมรับการแกว่งของสัญญาณได้ เพราะอาจทำให้อุปกรณ์ชำรุดเสียหาย ดังนั้น จึงต้องเลือกการปรับจูนตัวควบคุมให้ได้ผลตอบสนองในลักษณะอื่น เช่น ผลตอบสนองแบบหน่วงวิกฤต (Critical damped) ดังแสดงในรูปที่ 2.28 ถ้าหากเพิ่มอัตราขยายแก่ตัวควบคุมอีกเพียงเล็กน้อย ผลตอบสนองก็จะแกว่งรอบๆค่าอ้างอิงหลายครั้ง ตัวควบคุมที่ให้ผลตอบสนองแบบหน่วงวิกฤตจะให้ค่าคลาดเคลื่อนที่มีขนาดใหญ่กว่าตัวควบคุมที่ให้ผลตอบแบบ QAD แต่ยอมรับได้และตรงกับความต้องการ อย่างไรก็ตาม หากเพิ่มค่าอัตราขยายของตัวควบคุมให้สูงขึ้นอีก ระบบก็จะเกิดการแกว่งอย่างต่อเนื่อง ผลตอบลักษณะนี้เรียกว่าหน่วงขาด (Underdamped) และจะมีความคลาดเคลื่อน (วัดเทียบกับค่าอ้างอิง) เล็กน้อย ในทางกลับกันหากลดอัตราขยายของตัวควบคุมลง การรบกวนจะส่งผลต่อผลตอบสนองทำให้การแกว่งลดลง แต่ทำให้ความคลาดเคลื่อนมากขึ้น ผลตอบในลักษณะนี้เรียกว่าหน่วงเกิน (Overdamped)



รูปที่ 2.28 ผลของการปรับอัตราขยายที่มีผลตอบสนองเชิงเวลาในการเข้าสู่สถานะอยู่ตัว

2.6.3 วิธีการปรับค่าพารามิเตอร์ของตัวควบคุมพีไอดี

ตัวควบคุม PID สามารถเขียนได้ในรูปต่อไปนี้

$$C(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

โดยที่ K_p คือค่าอัตราขยายสัดส่วน (Proportional gain) T_i คือเวลารีเซ็ต (Reset time) มีหน่วยเป็นวินาที และ T_d คือ อัตราอนุพันธ์ (Derivative rate) มีหน่วยเป็นวินาที ลักษณะสมบัติและการปรับจูนตัวควบคุมแบบ P, PI และ PID มีรายละเอียดการปรับจูนด้วยวิธีต่างๆ เช่น Ziegler-Nichols, Shinskey หรือ Cohen-Coon

สูตร Ziegler-Nichols นั้น ให้ผลตอบสนองที่ดีที่สุดสำหรับกระบวนการที่มีค่าคงตัวทางเวลาที่ยาวกว่าค่าประวิงเวลา และการควบคุมระบบของเหลว อย่างไรก็ตามวิธีนี้อาจให้ผลที่ไม่ดีมากนักกับการควบคุมการไหลหรือความดันของของเหลว หรือการควบคุมวงรอบอื่นที่ต้องการปรับตัว

อย่างรวดเร็ว การปรับค่าด้วยสูตร Cohen-Coon นั้นจะให้ผลโดยรวมที่ดีสำหรับกระบวนการแบบ Self-regulating

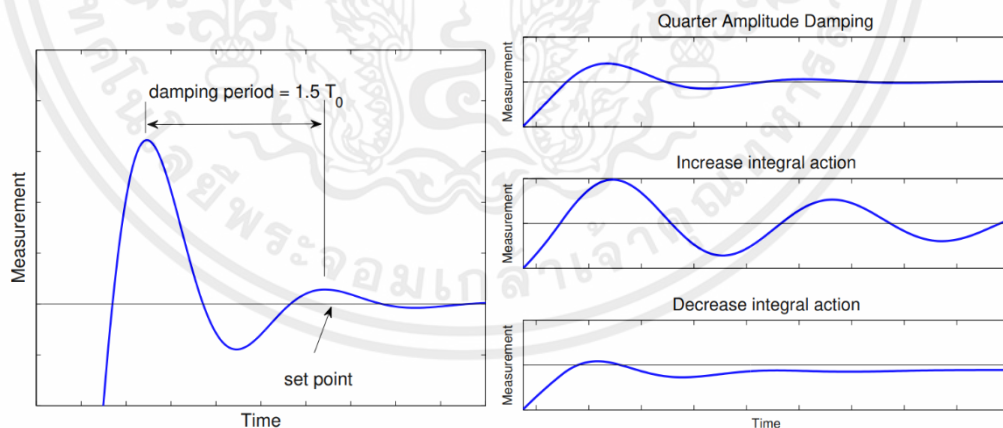
2.6.2.1 Proportional-only mode (P control)

ตัวควบคุมแบบสัดส่วน (P control) จะได้จากการกำหนดค่า $T_i = \infty$ และ $T_d = 0$ และเลือกค่า K_p ที่เหมาะสม ตัวควบคุมแบบนี้เหมาะในการใช้งานกับกระบวนการที่ต้องการผลตอบสนองที่รวดเร็ว ในขณะที่เดียวกันก็ยอมให้เกิดความคลาดเคลื่อนขนาดคงที่ค่าหนึ่ง ตัวควบคุมแบบสัดส่วนจะมีพารามิเตอร์ที่ต้องปรับจูนเพียงตัวเดียว ส่วนวิธีการปรับจูนพารามิเตอร์ สามารถเริ่มใช้จากสูตรดังตารางที่ 2.1 หรือ ตารางที่ 2.2 เพื่อเป็นค่าเริ่มต้น

หากไม่ต้องการให้ QAD การลด K_p จะทำให้เกิดการหน่วงวิกฤติ และหากยังคงลด K_p ต่อไปอีก ก็จะทำให้เกิดการหน่วงเกินและในทางตรงกันข้าม การเพิ่ม K_p จากการตั้งโดยวิธี QAD จะทำให้เกิดการหน่วงนั้นขาด

2.6.2.2 Proportional-plus-integral mode (PI control)

ตัวควบคุมแบบสัดส่วนและอินทิกรัล (หรือ PI) จะได้จากการกำหนดค่า $T_d = 0$ และเลือกค่า K_p และ T_i ที่เหมาะสม มีข้อได้เปรียบกว่าตัวควบคุมเชิงสัดส่วนอย่างเดียวคือ ให้ผลตอบสนองที่เร็วและให้ค่าความคลาดเคลื่อนที่สภาวะอยู่ตัวเป็นศูนย์ วิธีการปรับจูนพารามิเตอร์ของตัวควบคุมแบบ PI จะยุ่งยากขึ้นเนื่องจากมีพารามิเตอร์ที่ต้องเลือกจำนวน 2 ตัว



(a) ผลตอบสนองของตัวควบคุม PI

(b) ผลตอบสนองเมื่อเปลี่ยนค่าพารามิเตอร์ T_i

ที่ให้ค่า $T_{PI} = 1.5T_0$

รูปที่ 2.29 ผลตอบสนองกระบวนการเมื่อใช้ตัวควบคุม PI

Shinsky เสนอค่า Damped period ที่เหมาะสมกับตัวควบคุมแบบ PI คือเท่ากับ $1.5T_0$ โดยประมาณ สำหรับกระบวนการที่ไม่สามารถหาค่า Natural period T_0 จากวิธี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Close Loop ได้โดยง่ายนั้น ก็สามารถใช้ค่า τ และค่า L ที่หาได้จากวิธี Open circuit จากนั้นก็สามารถหาค่า T_0 ในรูปที่ 2.29a ได้แสดงผลตอบสนองของตัวควบคุม PI ที่คาบเท่ากับ $1.5T_0$ โดยมี Quarter amplitude damping (QAD) (ในรูปที่ 2.29b ให้กับตัวควบคุมทำให้ค่า damped period ของผลตอบสนองกระบวนการเนื่องจากตัวควบคุม PI และ $T_{PI} = 1.5T_0$) การเพิ่มตัวอินทิกรัล ดังรูปที่ 2.29b ให้กับตัวควบคุมทำให้ค่า damped period เพิ่มขึ้นและการแกว่งรอบค่าสุดท้ายนานขึ้น การลดผลจากตัวอินทิกรัลลง จะทำให้กระบวนการใช้ระยะเวลาขึ้นกว่าที่จะทำให้ผลตอบสนองกลับเข้าสู่ค่าที่กำหนด การปรับจูนอาศัยค่า T_0 หรือค่า τ และค่า L หรือใช้ทั้งสองอย่าง

ตารางที่ 2.1 : ตารางสรุปวิธีการปรับจูนพารามิเตอร์ตัวควบคุม PID จากวิธี Open Loop

ชนิดควบคุม	ค่าพารามิเตอร์	
	Ziegler-Nichols	Shinsky
P	$K_p = \frac{1}{RL}$	$K_p = \frac{1}{RL}$
PI	$K_p = \frac{0.9}{RL}$ $T_i = \frac{L}{0.3}$	
PID	$K_p = \frac{1.2}{RL}$ $T_i = 2L$ $T_d = 0.5L$	

ตารางที่ 2.2 : ตารางสรุปวิธีการปรับจูนพารามิเตอร์ตัวควบคุม PID จากวิธี Close Loop

ชนิดตัวควบคุม	ค่าพารามิเตอร์		
	Ziegler-Nichols	Cohen-Coon	Shinsky
P	$K_p = 0.5K_0$	$K_p = \frac{1}{RL} \left(1 + \frac{L}{3\tau} \right)$	$K_p = 0.5K_0$
PI	$K_p = 0.45K_0$ $T_i = T_0/1.2$	$K_p = \frac{1}{RL} \left(0.9 + \frac{L}{12\tau} \right)$ $T_i = L \left(\frac{30 + 3L/\tau}{9 + 20/L\tau} \right)$	$K_p = 0.5K_0$ $K_p = 0.43T_0$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PID	$K_p = 0.6K_0$	$K_p = \frac{1}{RL} \left(\frac{4}{3} + \frac{L}{4\tau} \right)$	$K_p = K_0/4$
	$T_i = 0.5T_0$	$T_i = L \left(\frac{32 + \frac{6L}{\tau}}{13 + \frac{8L}{\tau}} \right)$	$T_i = 0.5T_0$
	$T_d = T_0/8$	$T_d = \frac{4L}{11 + 2L/\tau}$	$T_d = 0.12T_0$

2.6.2.3 Proportional, Integral and derivative mode (PID)

ตัวควบคุมแบบ PID จะใช้กับกระบวนการที่มีการตอบสนองช้า และมีคาบเวลาที่ค่อนข้างนาน เหมาะสำหรับการประยุกต์ใช้กับกระบวนการควบคุมอุณหภูมิ เมื่อค่าอุณหภูมิที่วัดได้เริ่มเปลี่ยนแปลง heat rate อาจต้องเปลี่ยนแปลงอย่างรวดเร็ว Derivative action จะลดคาบเวลาของผลตอบสนองให้สั้นลงเมื่อเกิดการรบกวน (Upset) อย่างไรก็ตาม ตัวควบคุม PID ไม่สามารถใช้กับระบบที่มีสัญญาณรบกวนการวัดหรือการเปลี่ยนแปลงแบบฉับพลันและการปรับจูนค่าพารามิเตอร์ด้วยวิธีต่างๆ สามารถสรุปได้ในตารางที่ 2.1 และตารางที่ 2.2

บทที่ 3

วิธีดำเนินงาน

3.1 เครื่องมืออุปกรณ์และข้อมูล

3.1.1 MATLAB & Simulink

โปรแกรมหลักคือโปรแกรม MATLAB เป็นโปรแกรมที่ใช้ในการจัดการข้อมูลในลักษณะที่เป็น Matrix โดยจุดเด่นคือใช้ในการพัฒนาหรือทดลองอัลกอริทึมทางคณิตศาสตร์ การแสดงข้อมูลให้ออกมาในลักษณะที่เป็นกราฟฟิกส์และใช้ในการจำลองการทำงานของระบบต่างๆ และถัดมาจะเป็นสิ่งที่เรียกว่า Simulink โดย Simulink คือเครื่องมือย่อยที่อยู่ภายในโปรแกรม MATLAB ใช้สำหรับการ Simulation และการออกแบบ Model-Based โดยอธิบายให้เข้าใจง่ายคือการใช้ Block diagram ในการสร้างระบบขึ้นมาโดยเราสามารถทำให้ในหนึ่ง Block ประกอบไปด้วยหลายๆ System ย่อยภายในนั้น และภายใน System หนึ่งนั้น ก็ประกอบไปด้วยหลากหลาย Component รวมกันอยู่

3.1.2 Simulink-Toolboxes

ภายในโปรแกรม MATLAB นั้น จะมีสิ่งอำนวยความสะดวกที่เรียกว่า Toolboxes อยู่ ซึ่งมันคือฟังก์ชันการทำงานที่จะเขียนขึ้นตามแต่ลักษณะหรือจุดประสงค์ของมัน โดยจะเขียนขึ้นจากผู้เชี่ยวชาญในด้านนั้นๆ ยกตัวอย่างเช่น Image Processing หรือ Neural Network หรือจะเป็นอย่าง Motor Control ที่ใช้สำหรับโครงงานนี้เช่นเดียวกัน โดยภายในโครงงานนี้จะมีการนำ Toolboxes มาใช้งาน โดยสามารถจำแนก Toolboxes สำหรับการนำไปใช้งานได้ดังนี้

Electrical และ Mechanical Modeling ประกอบไปด้วย

- Simscape และ Simscape Electrical

การแปลงพิกัดสำหรับการควบคุม Vector ประกอบไปด้วย

- Simulink Control Design และ Control System Toolbox

การปรับหรือจูนค่าพารามิเตอร์ต่างๆในส่วนของ การควบคุม PWM ประกอบด้วย

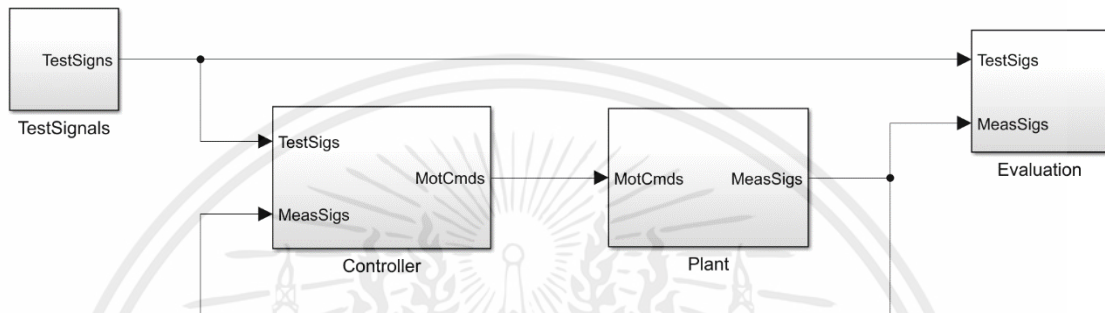
- Simulink Design Optimization และ Optimization Toolbox

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการดำเนินงาน

3.2.1 ภาพรวมของระบบการจำลองการทำงานของมอเตอร์รถยนต์ไฟฟ้าด้วยพีวชั้น เซนเซอร์

เป็นการนำบล็อกการทำงานต่างๆ ที่ได้กล่าวมาทั้งหมด นำมาเชื่อมต่อการทำงานเข้าด้วยกันจึงเกิดเป็นระบบใหญ่ที่ใช้ในการจำลองควบคุมมอเตอร์รถยนต์ไฟฟ้า



รูปที่ 3.1 ระบบควบคุมมอเตอร์รถยนต์ไฟฟ้า (PMSM)

3.2.2 บล็อก TestSignals

ขั้นตอนนี้เป็นขั้นตอนแรกสุด เริ่มจากการที่ผู้จัดทำโครงงานสร้างบล็อกที่ชื่อว่า TestSignals โดยบล็อกนี้มีหน้าที่ในการสร้างสัญญาณเริ่มต้นที่ใช้ในการประมวลผลต่อไปในส่วนต่างๆ และสัญญาณ (ความเร็ว) ที่สร้างออกมาจากบล็อกนี้นั้น เราสามารถกำหนดได้ตามที่ต้องการ



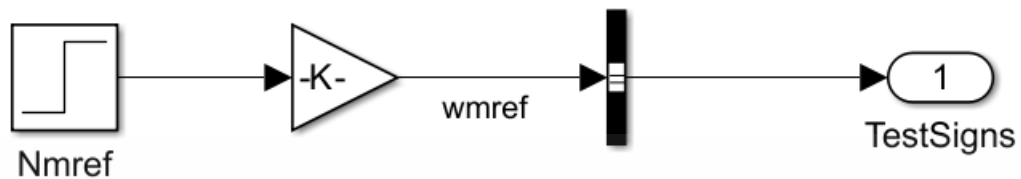
รูปที่ 3.2 บล็อก TestSignals

3.2.2.1 องค์ประกอบภายในบล็อก TestSignals

โดยภายในบล็อกนี้เราจะสร้างสัญญาณที่เป็นความเร็วได้โดยการ สร้างตัว กำหนดสัญญาณความเร็ว (หน่วยเป็นรอบต่อนาที : RPM) จากนั้นจะทำการแปลงสัญญาณความเร็วที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

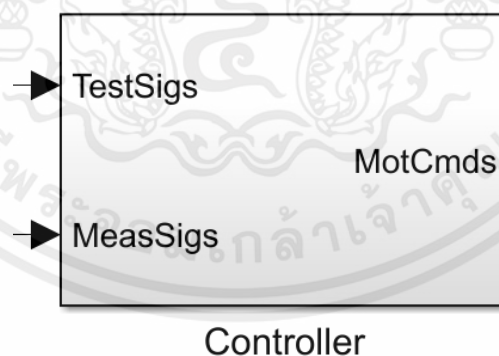
เป็นรอบให้เป็นความเร็วเชิงมุมโดยจะใช้ชื่อว่า wmref (หมายถึงความเร็วเชิงมุมที่เป็นลักษณะทางกล สำหรับใช้ในการอ้างอิง) และเข้าไปที่ Bus Creator เพื่อการนำไปใช้งาน (Select Bus) ต่อไป



รูปที่ 3.3 องค์ประกอบภายในบล็อก TestSignals

3.2.3 บล็อก Controller

บล็อกนี้เป็นบล็อกที่ทำหน้าที่ในการจำแนกสัญญาณต่างๆ และทำหน้าที่ในการควบคุมมอเตอร์รถยนต์ไฟฟ้า อีกทั้งบล็อกนี้ เป็นปัจจัยหลักสำคัญที่ใช้ในการควบคุมมอเตอร์รถยนต์ไฟฟ้า โดยภายในบล็อกจะประกอบไปด้วยหลากหลายบล็อกย่อยเพื่อใช้ในการควบคุม

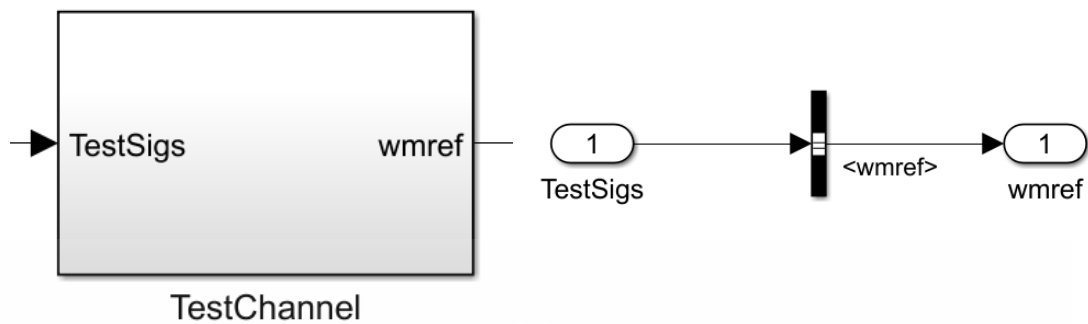


รูปที่ 3.4 บล็อก Controller

3.2.3.1 บล็อก TestChannel และองค์ประกอบภายในบล็อก

บล็อกนี้จะทำหน้าที่ในการนำสัญญาณ wmref ที่ได้จาก TestSignals เข้ามาใช้งานและจัดระเบียบสัญญาณจากนั้นจะนำสัญญาณนี้ เข้าไปที่บล็อก (FOC : Field Oriented Control) ต่อไป

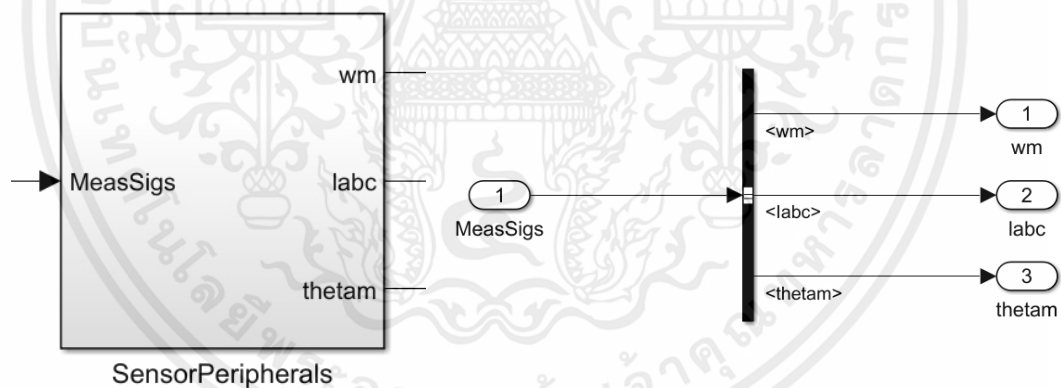
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 บล็อก TestChannel รูปที่ 3.6 องค์ประกอบภายในบล็อก TestChannel

3.2.3.2 บล็อก SensorPeripherals และองค์ประกอบภายในบล็อก

บล็อกนี้จะทำหน้าที่ในการจำแนกสัญญาณออกมาโดยการใช้ Bus Selector เพื่อจำแนกสัญญาณออกมา 3 ชนิด นั่นคือ wm (สัญญาณความเร็วเชิงมุมที่เป็นลักษณะเชิงกลที่เป็นค่าจริง), labc คือสัญญาณที่เป็นกระแสไฟฟ้าสำหรับจ่ายให้กับมอเตอร์ และ สัญญาณสุดท้ายคือ thetam ซึ่งเป็นค่าของมุมของมอเตอร์ว่าหมุนไปกี่องศาแล้ว และเมื่อทำการจำแนกแยกสัญญาณออกมทั้งหมดแล้วก็จะส่งต่อไปที่บล็อก FOC ต่อไป



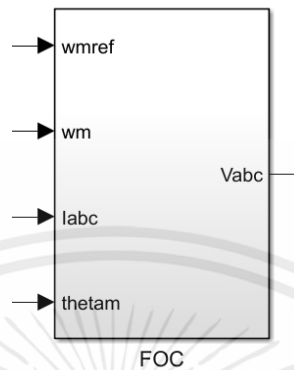
รูปที่ 3.7 บล็อก SensorPeripherals รูปที่ 3.8 องค์ประกอบภายในบล็อก SensorPeripherals

3.2.3.3 บล็อก FOC (Field-Oriented Control)

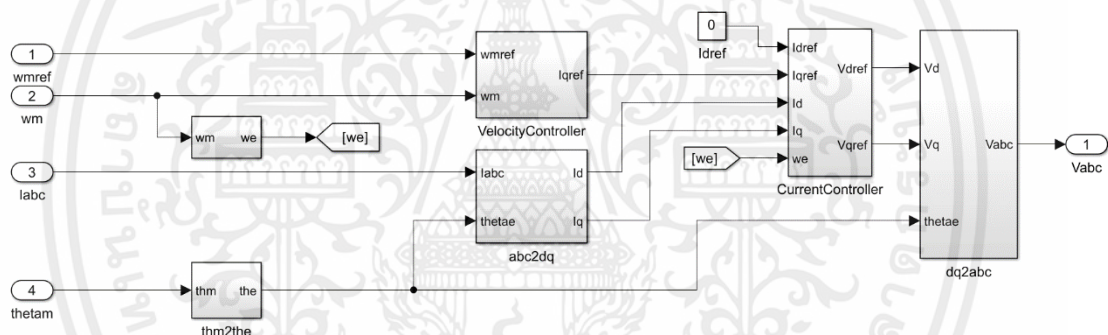
เป็นบล็อกหลักที่ใช้ในการควบคุมมอเตอร์ โดยบล็อก FOC นี้จะทำหน้าที่ในการควบคุม Gate Driver (จะอยู่ในส่วน Plant) และ Gate Driver นี้จะส่งสัญญาณไปที่ตัว Inverter (ในบล็อกจะใช้ชื่อว่า Converter) โดยตัว Inverter จะทำหน้าที่แปลงไฟฟ้ากระแสตรงให้เป็นไฟฟ้ากระแสสลับสามเฟส (DC to AC 3-Phase) โดยจะควบคุมมอเตอร์ด้วยการจ่ายกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเตเตอร์ ที่มีส่วนประกอบ 2 ส่วนคือ ส่วนที่สร้างฟลักซ์แม่เหล็กและส่วนที่สร้างแรงบิด โดยทั้งหมด จะถูกควบคุมอยู่บนแกนอ้างอิง ที่หมุนไปพร้อมกับฟลักซ์เวกเตอร์ของโรเตอร์



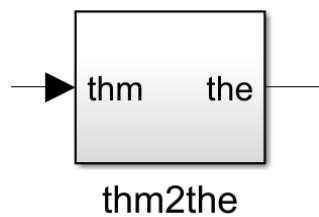
รูปที่ 3.9 บล็อก FOC



รูปที่ 3.10 องค์ประกอบภายในบล็อก FOC

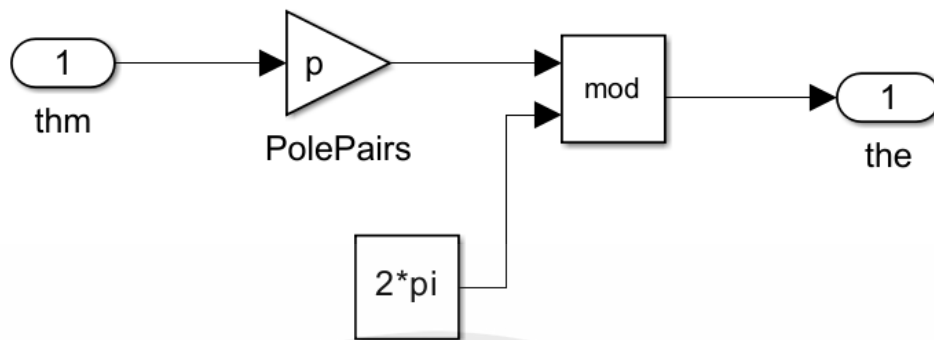
3.2.3.3.1 บล็อก thm2the และองค์ประกอบภายในบล็อก

เป็นบล็อกที่ทำหน้าที่เปลี่ยนมุมเชิงกลเป็นมุมเชิงไฟฟ้า โดยภายในบล็อกนี้จะมีการนำจำนวน Pole ที่ใช้ในการคำนวณด้วย โดย mod นั้นจะทำหน้าที่ในการคำนวณและให้ผลลัพธ์ออกมาเป็นค่ามุมของ Rotor ที่อยู่ในมอเตอร์ หรือภายในมุม 360 องศา



รูปที่ 3.11 บล็อก thm2the

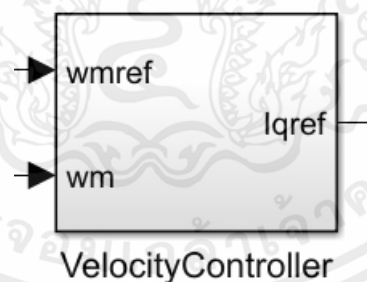
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



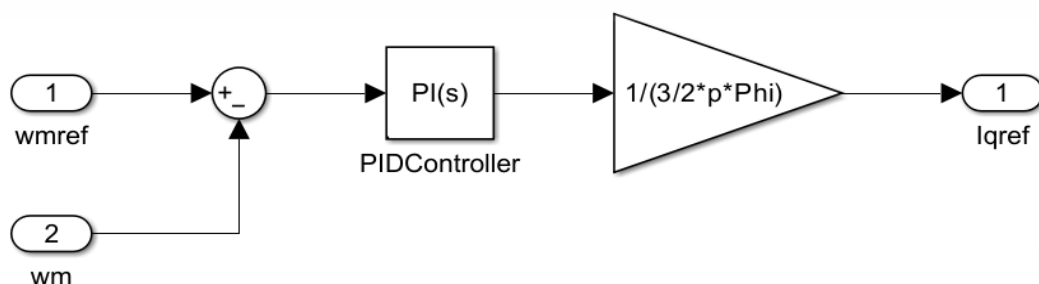
รูปที่ 3.12 องค์ประกอบภายในบล็อก thm2the

3.2.3.3.2 บล็อก VelocityController และองค์ประกอบภายในบล็อก

เป็นบล็อกที่ทำหน้าที่ในการควบคุมความเร็วของมอเตอร์ โดยจะใช้สัญญาณ 2 ชนิดเข้ามาเปรียบเทียบเพื่อหาความต่างกันของความเร็วเชิงมุมทั้งสอง และนำค่าที่ได้ออกมานั้น เข้าไปที่ PI Controller เพื่อให้ได้สัญญาณที่มีความเสถียรภาพและสมบูรณ์ตามที่เรารต้องการ จากนั้นจึงสัญญาณที่ได้ไปแปลงเป็นกระแส I ที่พิกัด q (I_{qref}) เพื่อที่นำสัญญาณที่ได้ออกมานี้ ไปใช้ในการควบคุมกระแสไฟฟ้าของระบบต่อไป (Current Controller)



รูปที่ 3.13 บล็อก VelocityController

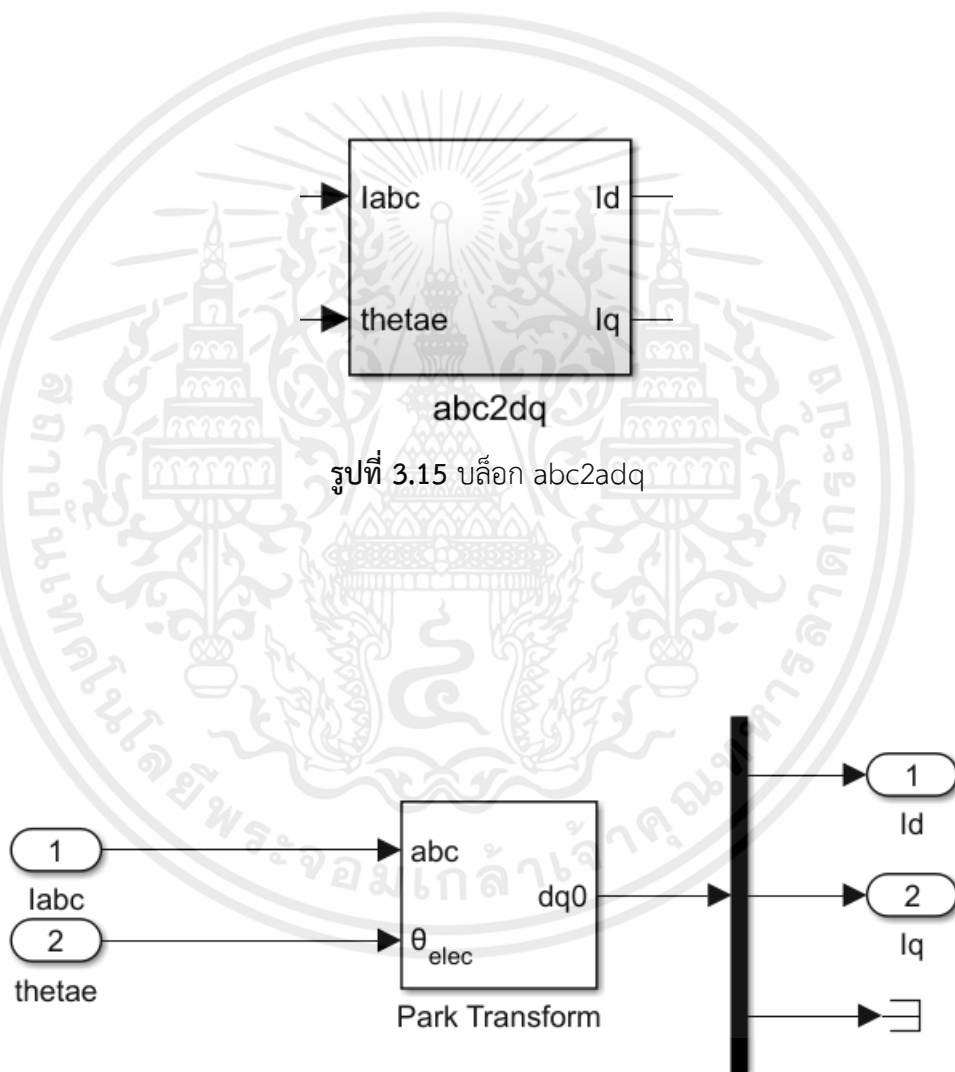


รูปที่ 3.14 องค์ประกอบภายในบล็อก VelocityController

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.3.3 บล็อก abc2dq และองค์ประกอบภายในบล็อก

เป็นบล็อกที่จะนำกระแสที่ใช้เลี้ยงมอเตอร์จากบล็อก SensorPeripherals มาใช้ร่วมกับ θ_{elec} เพื่อแปลงพิกัดของสัญญาณที่เป็นกระแสไฟฟ้าสำหรับจ่ายให้กับมอเตอร์ (i_{abc}) เป็น I_d และ I_q และจะเห็นได้ว่าในช่อง output ตัวที่สาม สิ่งนั้นเรียกว่า Terminator ซึ่งเป็น Component ที่มีหน้าที่ในการอุดสัญญาณ output ที่ไม่ต้องการได้ ซึ่งไม่ได้มีการทำงานพิเศษอะไร



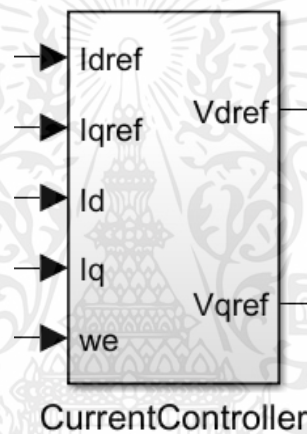
รูปที่ 3.15 บล็อก abc2adq

รูปที่ 3.16 องค์ประกอบภายในบล็อก abc2adq

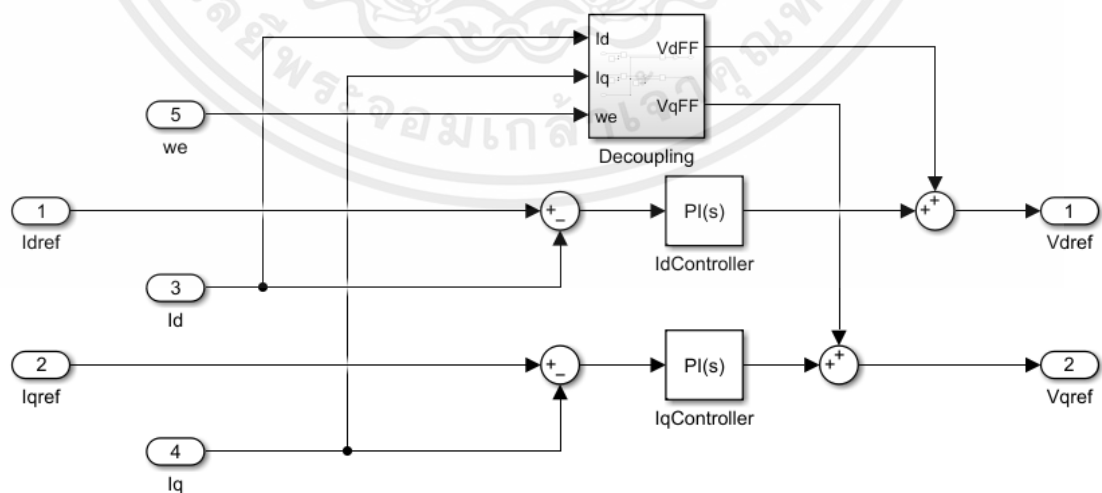
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.3.4 บล็อก CurrentController และองค์ประกอบภายในบล็อก

เป็นบล็อกที่ใช้ในการควบคุมกระแสไฟฟ้าได้โดยการนำ I_{dref} และ I_{qref} (กระแส d และ q อ้างอิง) มาเปรียบเทียบกับ I_d และ I_q (กระแส d และ q ที่ได้ออกมาจากการจำลองจริง) และหลังจากที่นำมาเปรียบเทียบกันแล้ว เราจึงนำค่าสัญญาณที่ได้ออกมานั้น เข้าไปที่ PI Controller เพื่อทำการปรับจนให้ได้สัญญาณที่มีความเสถียรภาพและสมบูรณ์ตามที่ต้องการ และหลังจากที่เราปรับจูนจาก PI Controller แล้วนั้น เราจะนำสัญญาณที่ได้ออกมาไปทำการควบคุมร่วมกับค่าแรงดัน Feed Forward ที่ได้จากการ Decoupling เพื่อสร้างสัญญาณ V_{dref} และ V_{qref} เพื่อนำไปใช้กับ Inverse Park Transform ต่อไป



รูปที่ 3.17 บล็อก CurrentController

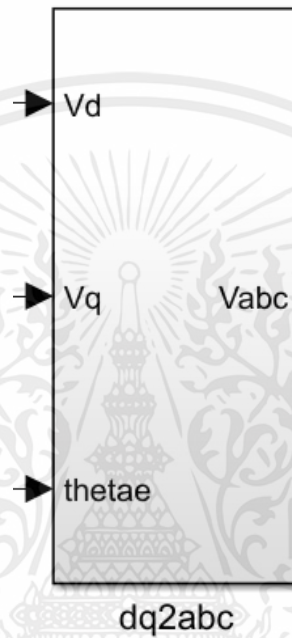


รูปที่ 3.18 องค์ประกอบภายในบล็อก CurrentController

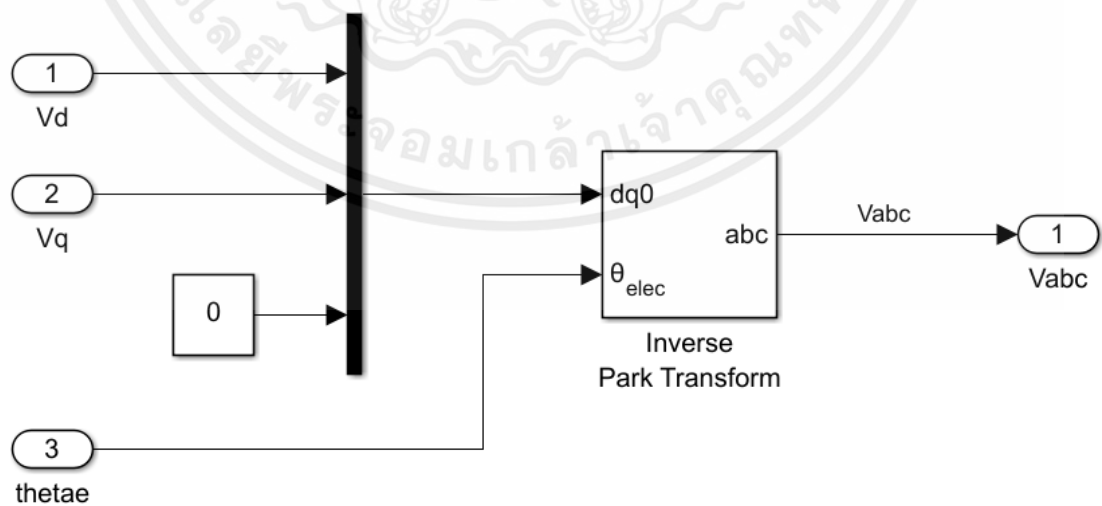
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.3.5 บล็อก dq2abc และองค์ประกอบภายในบล็อก

เป็นบล็อกที่ใช้ในการแปลงแรงดันไฟฟ้าจากพิกัด d และ q ไปเป็น พิกัด V_{abc} (แรงดันไฟฟ้า) ด้วย θ_{e} ที่เป็น Feedback กลับมาด้วยวิธี Inverse Park Transform



รูปที่ 3.19 บล็อก dq2abc

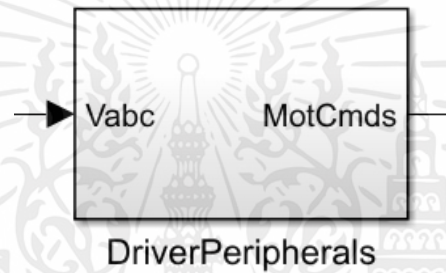


รูปที่ 3.20 องค์ประกอบภายในบล็อก dq2abc

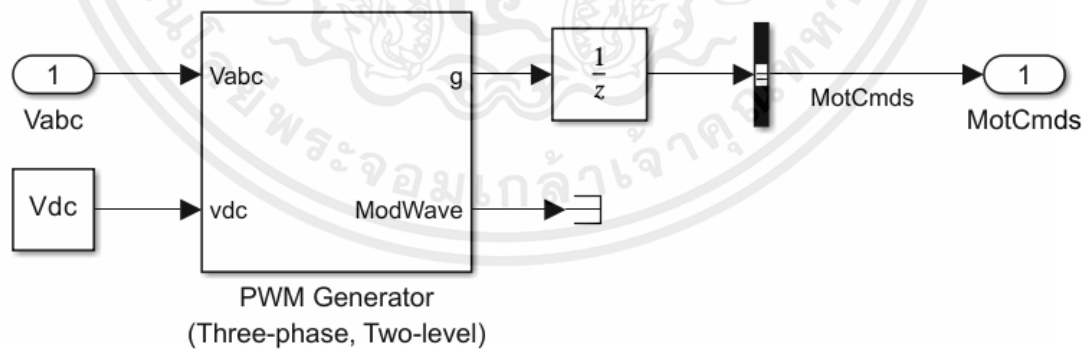
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.4 บล็อก DriverPeripherals

เป็นบล็อกที่ทำหน้าที่ในการสร้างสัญญาณ PWM ซึ่งสัญญาณนี้จะทำหน้าที่ในการส่งสัญญาณ Pulse เพื่อเปิด-ปิด สวิตช์แบบ Sinusoidal หรือในรูปแบบคลื่น Sine wave ตามที่มอเตอร์ต้องการ โดยบล็อกนี้จะประกอบไปด้วย Vabc และ Vdc ซึ่ง Vabc นี้จะได้มาจากบล็อกของ FOC และ Vabc นี้ เราสามารถกำหนดค่าได้จากในโปรแกรม MATLAB ได้เลย



รูปที่ 3.21 บล็อก DriverPeripherals

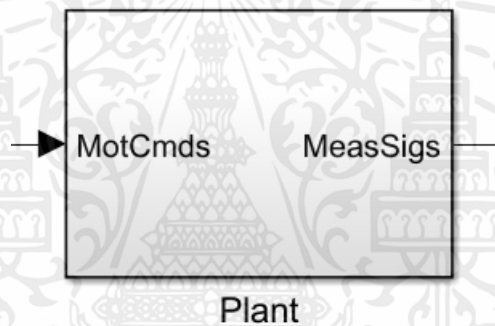


รูปที่ 3.22 องค์ประกอบภายในบล็อก DriverPeripherals

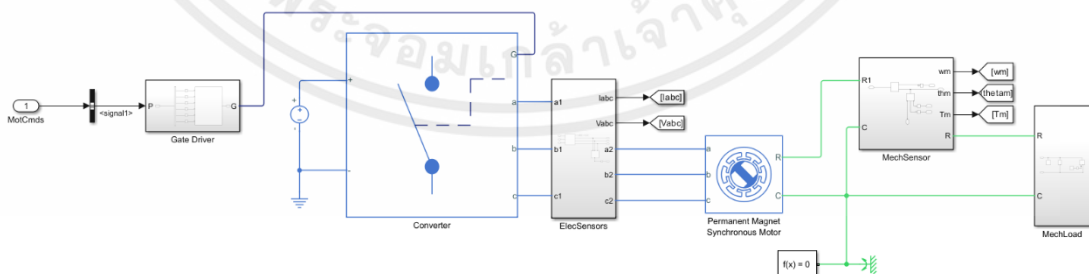
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 บล็อก Plant

บล็อกนี้เป็นบล็อกที่มีเซ็นเซอร์ต่างๆ ที่ใช้ในการวัดค่า เพื่อให้ได้ Data และส่งข้อมูลต่างๆ กลับไปที่บล็อก Controller เพื่อทำการควบคุมมอเตอร์อีกครั้ง โดยบล็อกนี้จะประกอบไปด้วย Inverter (Converter) ที่ทำหน้าที่ในการแปลงไฟฟ้ากระแสตรงเป็นไฟฟ้ากระแสสลับแบบสามเฟส (DC to AC-3 Phase) และบล็อกที่เรียกว่า ElecSensors ซึ่งจะมี Current Sensor และ Voltage Sensor อยู่ภายในอีกทั้งยังมีเซ็นเซอร์อีกสองตัว นั่นคือ Torque Sensor และ Rotational Motion Sensor และสุดท้ายคือหัวใจหลักของโครงการในครั้งนี้ ซึ่งนั่นก็คือมอเตอร์ถยนต์ไฟฟ้าที่ผู้จัดทำโครงการใช้ในการ Simulation ในครั้งนี้ PMSM : Permanent Magnet Synchronous Motor



รูปที่ 3.23 บล็อก Plant

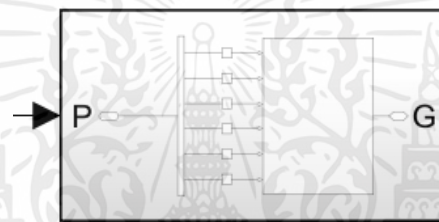


รูปที่ 3.24 องค์ประกอบภายในบล็อก Plant

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

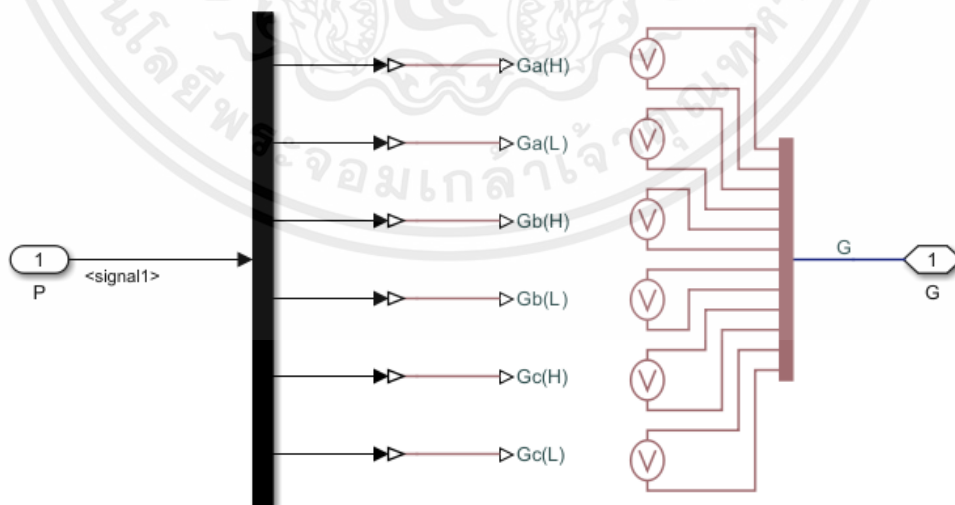
3.2.4.1 บล็อก Gate Driver

บล็อกนี้เป็นบล็อกที่ทำงานได้โดยการที่จะได้รับสัญญาณ PWM จากในส่วนของ Controller ส่งมาที่ตัว Gate Driver เพื่อให้ตัว Gate Driver นี้ทำหน้าที่ควบคุมการเปิดปิดของ สวิตซ์ทั้ง 6 ตัว โดยจะแบ่งสัญญาณออกเป็น 3 เฟส และในแต่ละเฟส จะมีสวิตซ์โลจิก High และ Low อีกทั้งสัญญาณที่ออกจาก Gate Driver นั้น จะทำหน้าที่เปิด-ปิดสวิตซ์ทั้ง 6 ตัว โดยที่แต่ละเฟส จะมีค่าความต่างเฟสอยู่ที่ 120 องศา และแรงดันไฟฟ้าจะอยู่ในลักษณะของรูปคลื่นไซน์



Gate Driver

รูปที่ 3.25 บล็อก Gate Driver

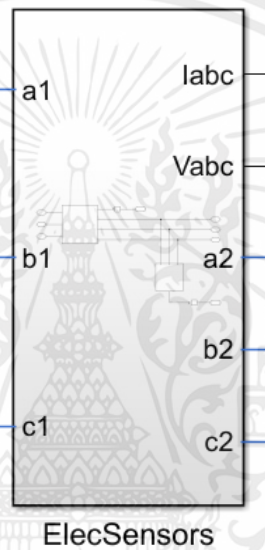


รูปที่ 3.26 องค์ประกอบภายในบล็อก Gate Driver

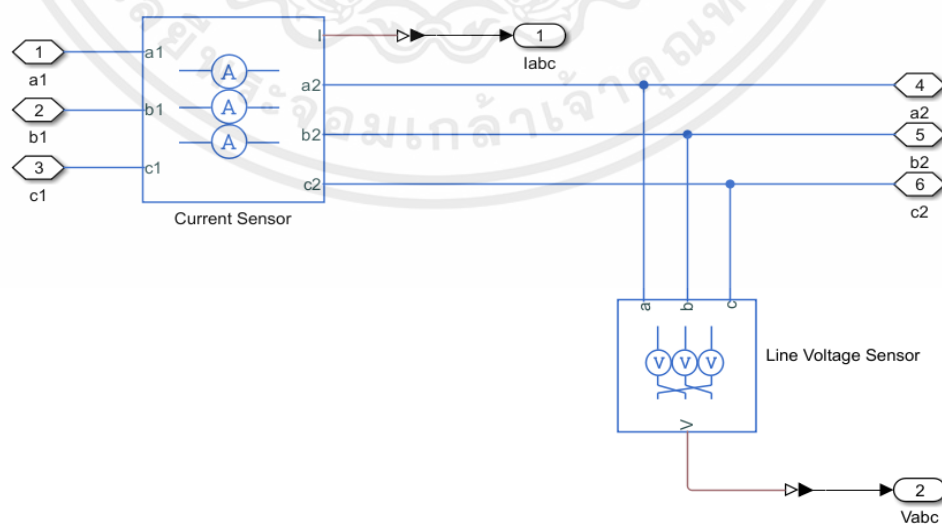
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4.2 บล็อก ElecSensors

บล็อกนี้เป็นบล็อกที่มีการใส่เซนเซอร์เอาไว้อสองตัวคือ Current Sensor และ Voltage Sensor ซึ่งเซนเซอร์ทั้งสองชนิดนี้ทำหน้าที่ตรวจวัดค่ากระแสไฟฟ้าและแรงดันไฟฟ้า โดยกระแสไฟฟ้าจะเป็นค่า Feedback นำกลับไปคำนวณในส่วนของบล็อก Controller และในส่วนของค่าแรงดันไฟฟ้าที่วัดออกมาจากเซนเซอร์นั้น จะเป็นค่าที่ใช้ในการเช็คค่าในส่วนองแรงดันไฟฟ้าที่ใช้ นั้นมีความผิดปกติหรือไม่ (นำ Vabc หรือแรงดันไฟฟ้าสามเฟสนี้ เข้าไปต่อกับ Scope โดยตรง)



รูปที่ 3.27 บล็อก ElecSensors

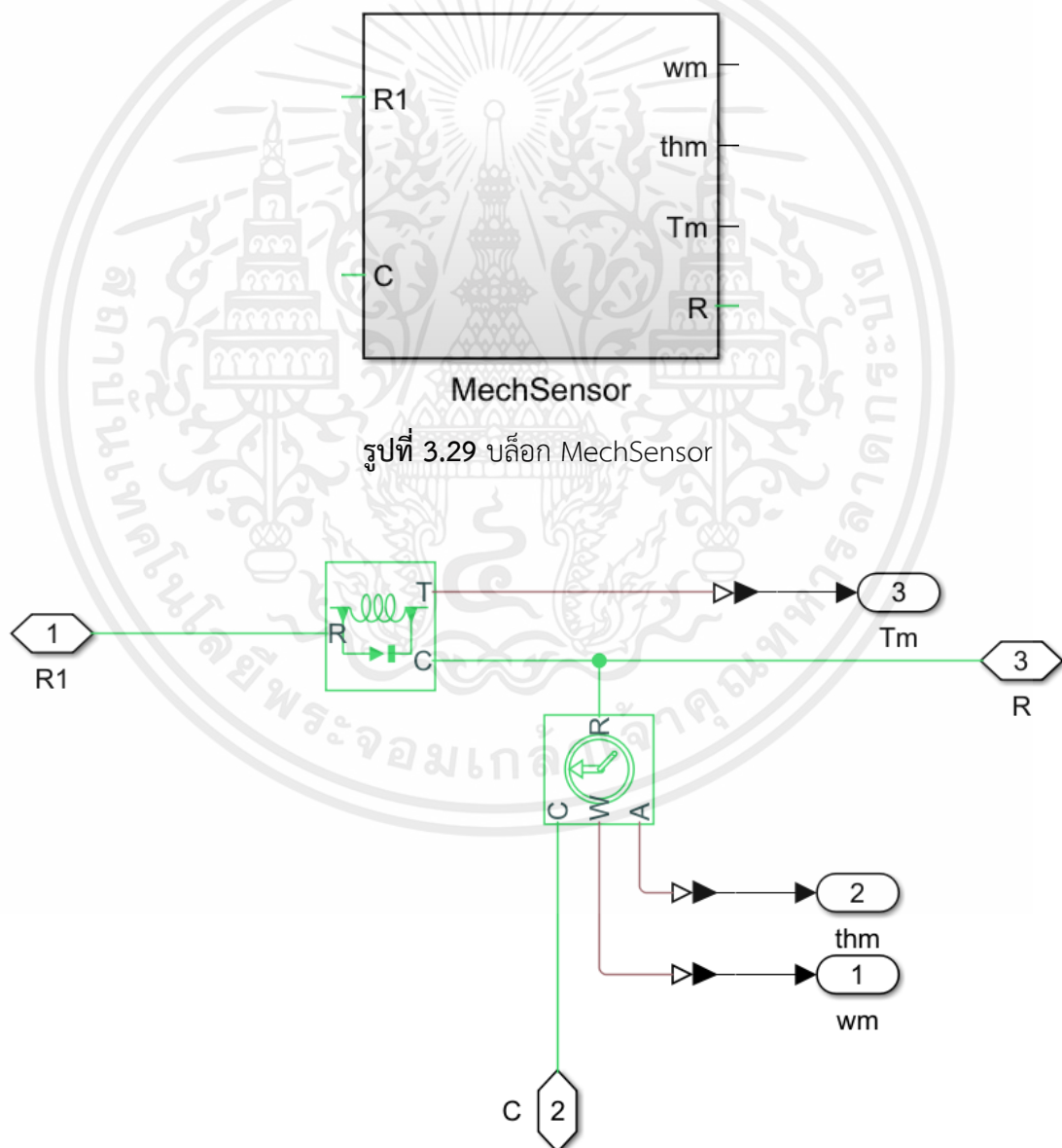


รูปที่ 3.28 องค์ประกอบภายในบล็อก ElecSensors

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4.3 บล็อก MechSensor

บล็อกนี้เป็นบล็อกมีการใส่เซนเซอร์เอาไว้ด้วยกันสองชนิดคือ Torque Sensor ที่เป็น Ideal และ Rotational Motion Sensor ที่เป็น Ideal เช่นเดียวกัน โดย Torque Sensor จะรับค่าจาก PMSM ที่เป็นค่า Torque ที่เกิดขึ้นภายใน Motor จากนั้นจึงแปลงออกมาเป็นตัวเลขแล้วจึงส่งค่าออกไปที่ Scope โดยตรง และ Rotational Motion Sensor จะทำหน้าที่ในการวัดความเร็วเชิงมุม (w) ในการหมุนของมอเตอร์และมุม θ ของมอเตอร์ที่ใช้ในการกำหนด ตำแหน่งการหมุนของมอเตอร์

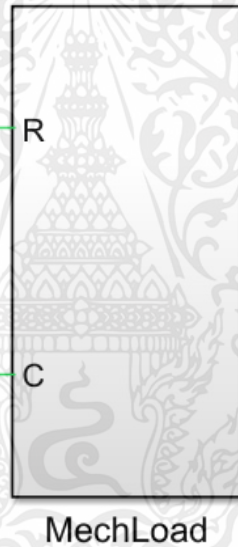


รูปที่ 3.30 องค์ประกอบภายในบล็อก MechSensor

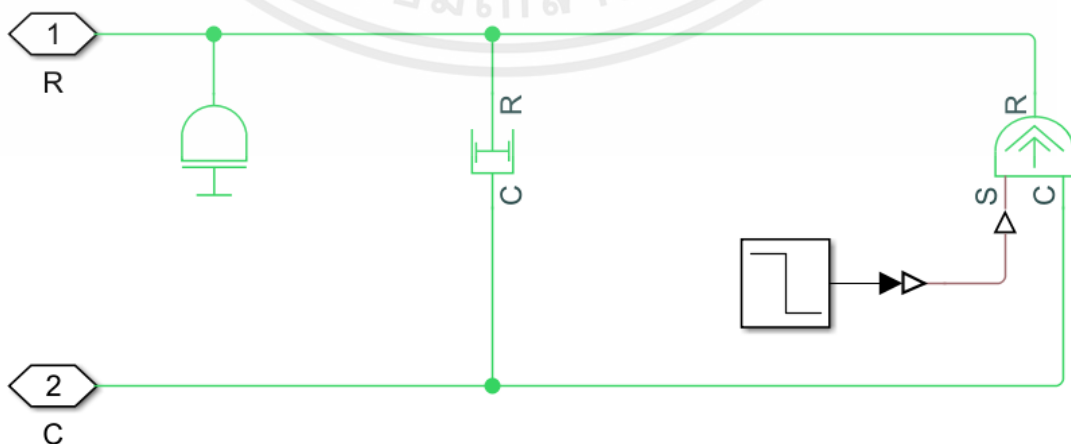
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4.4 บล็อก MechLoad

บล็อกนี้เป็นบล็อกใช้ในการจำลองโพลดภาระที่เกิดขึ้น เพื่อให้เห็นถึงความแตกต่างกันในสถานการณ์ต่างๆ อย่างชัดเจนยิ่งขึ้น ในเวลาที่โพลดเพิ่มขึ้น มอเตอร์นั้นจะมีการเปลี่ยนแปลงอย่างไร ซึ่งการที่เราจะทำอย่างนั้นได้ MechLoad คือสิ่งที่จำเป็นสำหรับการทำเทคนิคนี้ โดยภายในบล็อก MechLoad นี้จะประกอบไปด้วย Inertia ซึ่งจะทำหน้าที่ร่วมกับ Rotational Damper คือเป็นตัวที่ให้ค่าความเฉื่อยและความหน่วงให้กับมอเตอร์ และสิ่งสุดท้ายภายในของ MechLoad ก็คือ Ideal Torque Source ซึ่งจะเป็นจุดกำเนิดค่า Torque ซึ่งเราสามารถกำหนดหรือสร้างค่า Torque ได้จากการใส่ Source เข้ากับ Torque โดยตรงเพื่อให้ได้ค่า Torque ที่ต้องการหรือเหมาะสมได้



รูปที่ 3.31 บล็อก MechLoad

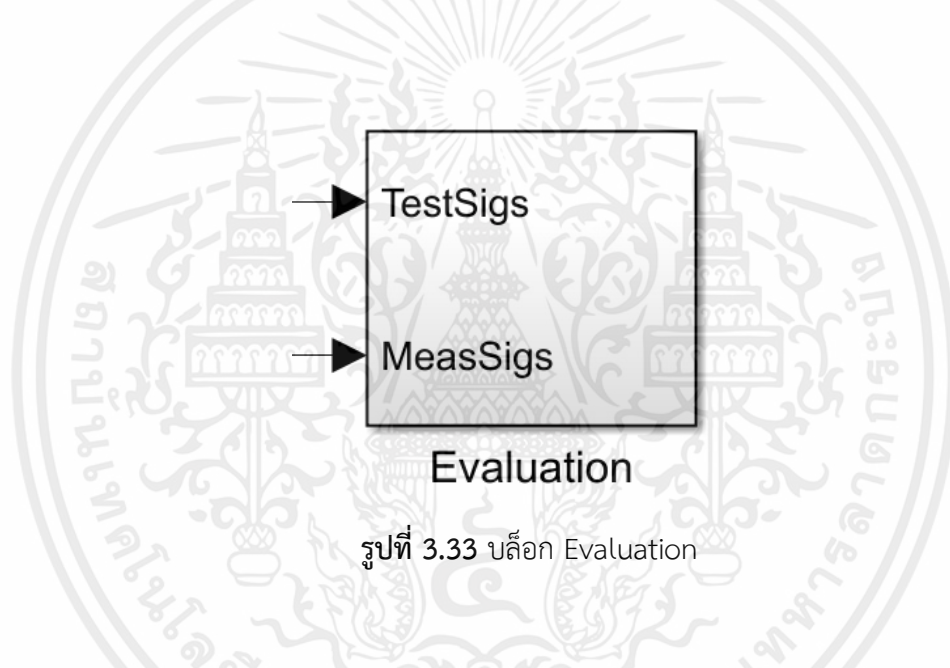


รูปที่ 3.32 องค์ประกอบภายในบล็อก MechLoad

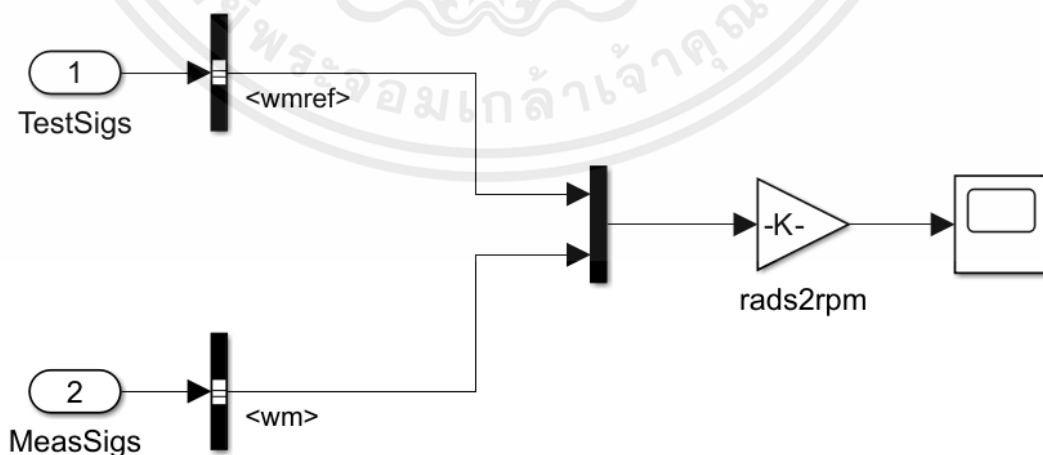
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 บล็อก Evaluation

บล็อกนี้จะทำหน้าที่ในการรับสัญญาณเอาต์พุตมาทำการเปรียบเทียบให้เห็นภาพ โดยการใช้ Scope โดย Input ที่เข้ามาจะประกอบไปด้วย wref หรือก็คือความเร็วเชิงมุมทางกลที่ใช้อ้างอิง ที่เราได้กำหนดค่าไว้แล้วในส่วนของ TestSigs และ MeasSigs ซึ่งเราจะเลือกสัญญาณโดยการใช้ Bus Selector เลือกสัญญาณความเร็วเชิงมุมทางกลที่เกิดจากการจำลองการหมุนจริงๆ มาเปรียบเทียบกับสัญญาณอ้างอิง โดยเราจะนำสัญญาณทั้งสองชนิดนี้มาแปลงจากสัญญาณเชิงมุม (ความเร็ว) ให้เป็นความเร็วต่อรอบ (RPM) จากนั้นจึงนำสัญญาณที่ได้ทั้งสองต่อเข้ากับ Scope เพื่อใช้ในการเปรียบเทียบหรือตรวจสอบผลลัพธ์ว่าเป็นไปตามที่ต้องการหรือไม่



รูปที่ 3.33 บล็อก Evaluation



รูปที่ 3.34 องค์ประกอบภายในบล็อก Evaluation

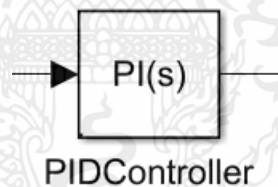
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 การใช้งาน PI Controller

ในส่วนของการปรับแต่งสัญญาณเอาต์พุตที่เป็นความเร็วรอบของมอเตอร์ (RPM) ให้มีความเสถียรภาพ (Stable) และค่าความผิดพลาด (Error) น้อยลง จึงได้ใช้ PI Controller ในการปรับแต่งสัญญาณ เนื่องจากต้องการเอาต์พุตที่มีความเสถียรภาพและสามารถปรับแต่งได้ง่ายที่สุด จึงทำการปรับแต่งค่า P (Proportional) เพื่อที่จะทำการลด Rise-time หรือลดช่วงเวลา Response-time ให้ระบบสามารถทำงานได้ไวยิ่งขึ้นและใช้การปรับแต่งค่า I (Integral) เพื่อกำจัดค่า Steady state error ออกไป เพื่อให้ได้เอาต์พุตเท่ากับค่าที่เราต้องการ เพียงเท่านี้ก็เพียงพอแล้วในการปรับแต่งสัญญาณเพื่อให้เอาต์พุตของระบบที่ออกมาเกิดเสถียรภาพและพร้อมใช้งาน

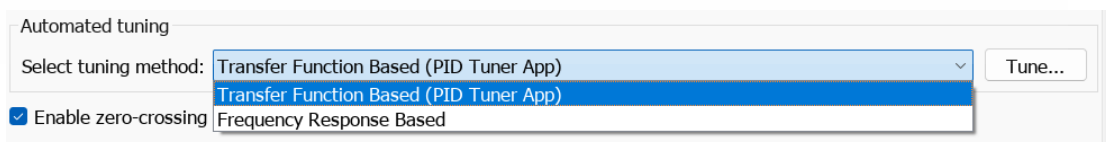
ในการจะใช้งาน PI Controller เราจะใช้ในการปรับแต่งสัญญาณในส่วนของ VelocityController เพื่อให้ความเร็วรอบของระบบ (มอเตอร์ PMSM) มีเสถียรภาพมากที่สุด โดยในการจำลองการทำงานของมอเตอร์ PMSM ผู้จัดทำได้ใช้เครื่องมือสำเร็จรูปในการปรับแต่งสัญญาณ (PID Tuner App) โดยมีขั้นตอนดังนี้ (หมายเหตุ : สำหรับ Simulink ที่ลง Simulink Design Optimization และ Optimization Toolbox เท่านั้น)

ขั้นตอนที่ 1 : เลือกบล็อก PID Controller



รูปที่ 3.35 บล็อก PIDController ที่ใช้สำหรับปรับแต่งสัญญาณ PI

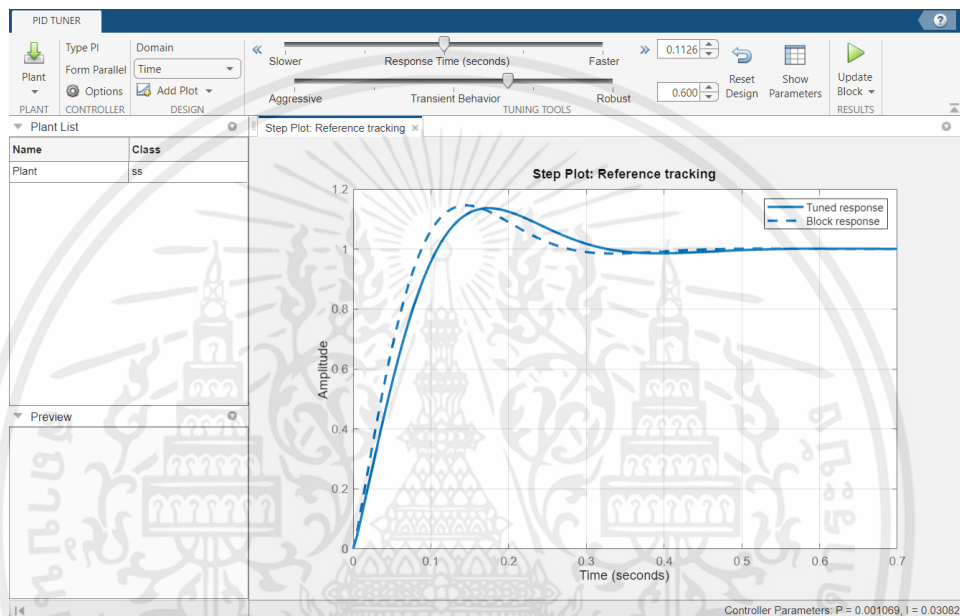
ขั้นตอนที่ 2 : คลิกเข้าไปใน PIDController แล้วจะพบหน้าต่างขึ้นมา ในส่วนของ Automated tuning เราจะเลือก Tuning method เป็น Transfer Function Based (PID Tuner App) จากนั้นก็ให้กด OK แล้วคลิกเข้าไปที่ปุ่ม Tune



รูปที่ 3.36 การเลือกวิธีที่ใช้ในการปรับแต่งสัญญาณ PI

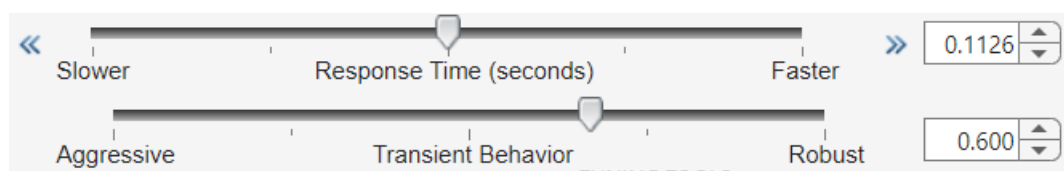
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 : โดยหลังจากที่เราคลิกเข้าไปที่ปุ่ม Tune แล้วจะพบกับหน้าต่างสำหรับการปรับแต่งสัญญาณในรูปที่ 3.37 และเมื่อเราเข้ามาที่หน้าต่างนี้แล้วนั้น เราจะเห็นกราฟที่เป็น เส้นประสีฟ้า ซึ่งเส้นประหมายถึง สัญญาณจริงในขณะนี้ (สัญญาณก่อนมีการปรับแต่ง) และเส้นทึบสีฟ้าหมายถึง สัญญาณที่เราจะปรับแต่งออกมาได้ถ้าหากเรากด Update Block (ลักษณะของสัญญาณที่ได้รับการปรับแต่งแล้วในปัจจุบัน)



รูปที่ 3.37 หน้าต่างโปรแกรม PID Tuner

ขั้นตอนที่ 4 : วิธีสำหรับการปรับแต่งสัญญาณคือ เราจะเห็น Response Time และ Transient Behavior ด้านบนของกราฟ โดยเราจะเห็นเป็นหลอดที่สามารถลากปุ่มที่อยู่ภายในหลอดไปทางซ้ายและขวาได้ตามต้องการ ซึ่งนั่นจะเป็นการใช้ในการปรับแต่งสัญญาณ โดยเราสามารถค่อยๆ ขยับปุ่มภายในหลอดทีละนิด เพื่อสังเกตความเปลี่ยนแปลงของสัญญาณได้ หรืออีกวิธีที่ใช้ในการปรับค่าของสัญญาณคือ จะมีช่องที่อยู่ด้านขวาของหลอดสำหรับปรับค่าสัญญาณอยู่ ซึ่งช่องนั้นก็สามารใช้ในการปรับสัญญาณได้เช่นเดียวกัน โดยเราสามารถปรับแต่งสัญญาณให้ได้ตามที่เรต้องการได้โดยวิธีที่กล่าวมาข้างต้น และสำหรับโครงการครั้งนี้ ผู้จัดทำจะใช้ค่า Response Time = 0.0227 seconds และค่า Transient behavior = 0.846



รูปที่ 3.38 ส่วนที่ใช้ในการปรับแต่งค่าสัญญาณภายใน PID Tuner

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

4.1 ผลจากการจำลองการควบคุมมอเตอร์ของรถยนต์ไฟฟ้าด้วยฟิวชันเซนเซอร์

ผลการทดลองนี้ แสดงถึงค่า ความเร็วรอบต่อนาที (RPM), แรงบิด (Torque), แรงดันไฟฟ้าสามเฟส (3-Phase Voltage) และกระแสไฟฟ้าสามเฟส (3-Phase Current)

โดยผลลัพธ์ที่ได้นี้ เป็นผลลัพธ์จากการควบคุมมอเตอร์ (PMSM) ให้ได้ความเร็วตามที่ผู้จัดทำโครงการต้องการและมีเงื่อนไขในการจำลองดังนี้

ที่เวลา $T = 0.0$ ถึง 0.5 วินาที, ความเร็วรอบ = 0 RPM

ที่เวลา $T = 0.5$ ถึง 1.0 วินาที, ความเร็วรอบ = 1000 RPM

ที่เวลา $T = 1.0$ ถึง 1.5 วินาที, ความเร็วรอบ = 3000 RPM

ที่เวลา $T = 1.5$ ถึง 2.5 วินาที, ความเร็วรอบ = 6000 RPM

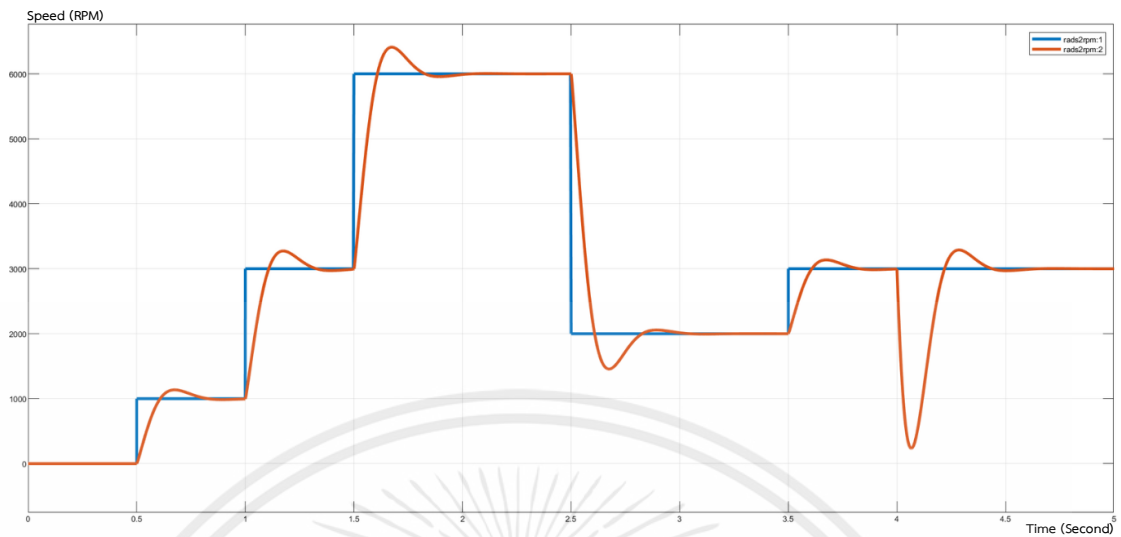
ที่เวลา $T = 2.5$ ถึง 3.5 วินาที, ความเร็วรอบ = 3000 RPM

ที่เวลา $T = 3.5$ ถึง 4.0 วินาที, ความเร็วรอบ = 4000 RPM

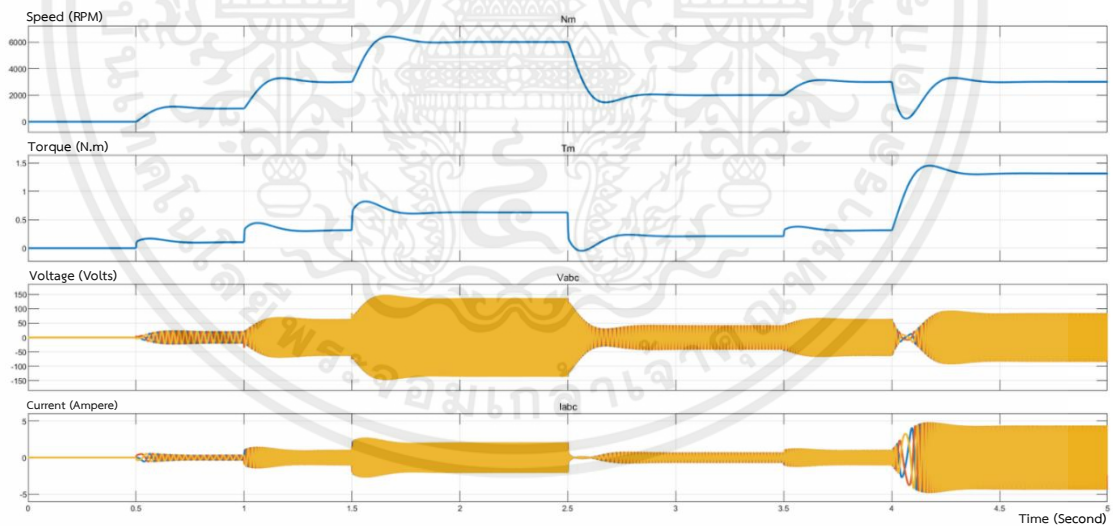
ที่เวลา $T = 4.0$ ถึง 5.0 วินาที, ความเร็วรอบ = 4000 RPM และมีโหลด = 1 Nm

ซึ่งทางผู้จัดทำโครงการได้ใช้ PI-Controller เพื่อให้ผลลัพธ์ที่ได้ออกมานั้นเกิดเสถียรภาพมากที่สุด โดยค่า P (Proportional) = 0.008838 และค่า I (Integral) = 0.0934 สำหรับการปรับแต่งสัญญาณ (PI Controller)

- เส้นกราฟสีแดง คือ ค่าความเร็ว (RPM) ของมอเตอร์ที่ได้วัดได้จากการจำลอง
- เส้นกราฟสีน้ำเงิน คือ ค่าความเร็ว (RPM) ของมอเตอร์ที่เราต้องการหรือเป็นค่าในอุดมคติ

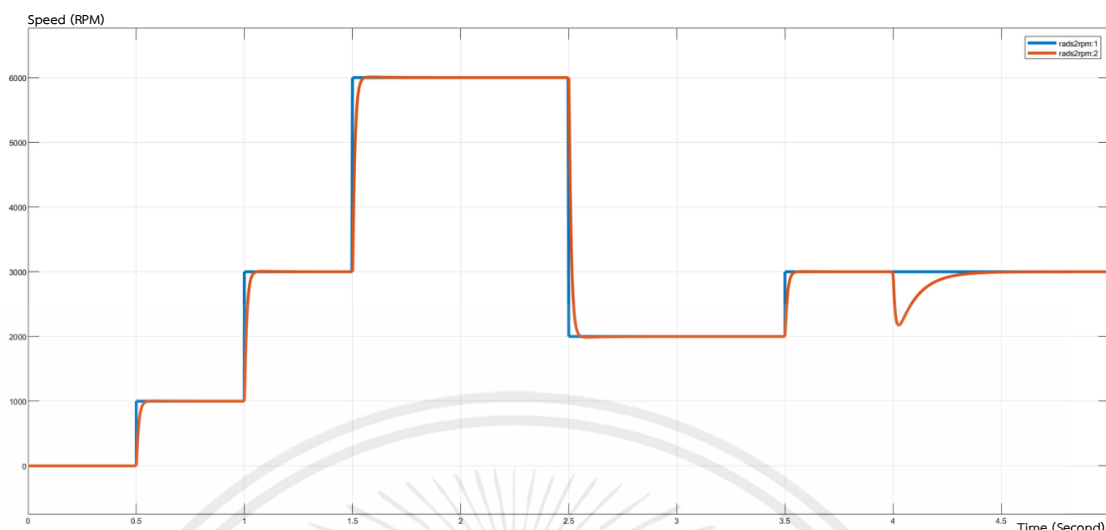


รูปที่ 4.1 กราฟเปรียบเทียบความเร็วรอบ (RPM) ก่อนปรับค่า PI

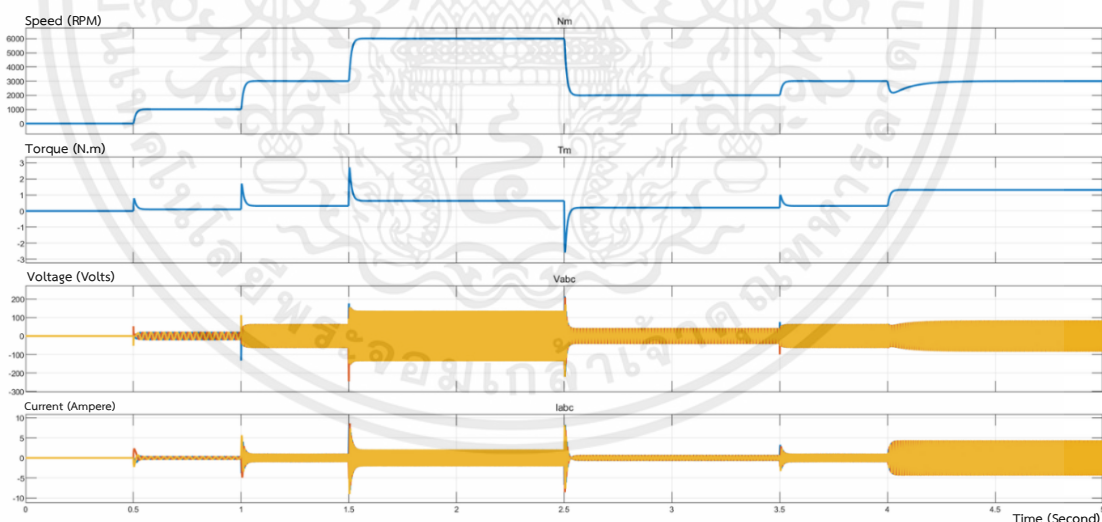


รูปที่ 4.2 กราฟความเร็วรอบ, แรงบิด, แรงดันไฟฟ้า และกระแสไฟฟ้า ก่อนปรับค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 กราฟเปรียบเทียบความเร็วรอบ (RPM) หลังปรับค่า PI



รูปที่ 4.4 กราฟความเร็วรอบ, แรงบิด, แรงดัน และกระแส หลังปรับค่า PI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

จากการทดลองโดยอ้างอิงจากรูปที่ 4.1 ถึง 4.4 จะเห็นได้ว่า ความเร็วของมอเตอร์รถยนต์ไฟฟ้าก่อนที่จะมีการปรับค่าโดยการใช้ PI Controller นั้น มีความไม่เสถียรภาพเป็นอย่างมาก ซึ่งส่งผลให้ค่าแรงบิด (Torque) ที่เกิดขึ้นนั้น ไม่สอดคล้องกับความเร็วของมอเตอร์ที่ต้องการเป็นอย่างมาก ซึ่งส่งผลให้ความเร่งของมอเตอร์ไม่เร็วพอที่จะใช้งานได้อย่างมีประสิทธิภาพ แต่หลังจากที่ได้มีการปรับค่าโดยการใช้งาน PI Controller แล้วนั้น ส่งผลให้ค่าความเร็วของมอเตอร์เกิดเสถียรภาพ (ใกล้เคียงกับค่าความเร็วในอุดมคติ) และแรงบิดที่เกิดขึ้นนั้น ส่งผลให้ความเร่งของมอเตอร์นั้น มีค่ามากเพียงพอที่จะทำให้ความเร็วรอบให้ถึงจุดที่ต้องการภายในเวลาอันสั้น

5.2 วิจารณ์ผลการทดลองและข้อเสนอแนะ

เนื่องจากการทดลองที่ได้ปฏิบัติอยู่นั้น เป็นการจำลองการทำงานของมอเตอร์รถยนต์ไฟฟ้า ซึ่งไม่ได้มีการใช้งานมอเตอร์จริงๆ จึงอาจจะมีตัวแปรบางอย่างที่แตกต่างกัน เช่น อุณหภูมิ, โหลด, ภาระ, ความต่อเนื่องในการทำงานและอาจมีตัวแปรอื่นๆ ที่ผู้จัดทำโครงงานไม่ทราบ ด้วยเหตุที่กล่าวมานี้ จึงทำให้การทดลองนี้ ยังไม่สมบูรณ์เท่าที่ควร อีกทั้งผู้จัดทำโครงงาน ไม่สามารถหาข้อมูลของมอเตอร์รถยนต์ไฟฟ้าที่ใช้งานกันอยู่ในปัจจุบันได้อย่างครบถ้วน เนื่องจากข้อมูลหรือค่าพารามิเตอร์หลายๆอย่างที่เป็นตัวแปรที่ทำให้เกิดความแตกต่างกันของมอเตอร์ในรถยนต์แต่ละยี่ห้อเป็นความลับทางการค้าของบริษัทต่างๆ ทางผู้จัดทำเลยทำได้แค่อ้างอิงจากข้อมูลบางส่วน หรือเท่าที่จะหาได้เท่านั้น และอีกหนึ่งปัญหาสำคัญคือ โครงงานนี้รวบรวมข้อมูลเพื่อใช้ในการทดลองค่อนข้างยาก และมีความซับซ้อนของเนื้อหาที่ใช้ในการนำมาประยุกต์รวมกันค่อนข้างมาก จึงทำให้เกิดความล่าช้าในการทำโครงงานชิ้นนี้

ข้อเสนอแนะคือ โครงงานชิ้นนี้สามารถนำไปพัฒนาต่อได้โดยการเพิ่มส่วนของตัวรถ หรือจะเพิ่มสถานการณ์ต่างๆเข้ามาร่วมใช้ในการทดสอบหรือจำลองการทำงานของมอเตอร์ หรือจะเป็นการเพิ่มจำนวนมอเตอร์ เพื่อใช้อ้างอิงให้ใกล้เคียงกับของจริงมากยิ่งขึ้น และสามารถพัฒนาได้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] Microsemi: Park, Inverse Park and Clarke, Inverse Clarke Transformations MSS Software Implementation User Guide
- [2] Microsemi: Field Oriented Control of Permanent Magnet Synchronous Motors User's Guide
- [3] MathWorks: Three-Phase PMSM Drive
- [4] MathWorks: Field-Oriented Control | Develop field-oriented control algorithms using simulation
- [5] Wengliang Liu and Pengfei Xu (2019) : Design of PMSM Speed Control system based on Simulink Model
- [6] ศ.ดร.เดวิด บรรณเจตพงศ์ชัย, ผศ.ดร.มานพ วงศ์สายสุวรรณ, ผศ.ดร.สุชิน อรุณสวัสดิ์วงศ์ และ รศ.ดร.จิตโกมุท ส่งศิริ (2021) : Control and Instrumentation Laboratory