



## ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมเครื่องกล)

### ปริญญา

วิศวกรรมเครื่องกล

วิศวกรรมเครื่องกล

สาขา

ภาควิชา

เรื่อง

การรักษาสมดุลของหุ่นยนต์สองล้อด้วยตัวคุมค่ากำลังสองเชิงเส้น  
และตัวควบคุมโมเดลพรีดิกทีฟ

Self-Balancing of Two-Wheeled Mobile Robot with Linear Quadratic Regulator  
and Model Predictive Controller

นามผู้วิจัย

นายพงศกร กวีสุนทรเสนาะ

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

( รองศาสตราจารย์วิท นัตร์รัตนกุลชัย, Ph.D. )

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

( ผู้ช่วยศาสตราจารย์ทวีเดช ศิริธนาพิพัฒน์, Ph.D. )

หัวหน้าภาควิชา

( รองศาสตราจารย์ชวลิต กิตติชัยการ, Ph.D. )

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

( รองศาสตราจารย์กัญญา ชีระกุล, D.Agr. )

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... เดือน ..... พ.ศ. ....

วิทยานิพนธ์

เรื่อง

การรักษาสมดุลของหุ่นยนต์สองล้อด้วยตัวคุมค่ากำลังสองเชิงเส้น และตัวควบคุมโมเดลพรีดิกทีฟ

Self-Balancing of Two-Wheeled Mobile Robot with Linear Quadratic Regulator and  
Model Predictive Controller

โดย

นายพงศกร กวีสุนทรเสนาะ

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมเครื่องกล)

พ.ศ. 2557

ลิขสิทธิ์ มหาวิทยาลัยเกษตรศาสตร์

พงศกร กวีสุนทรเสนาะ 2557: การรักษาสมดุลของหุ่นยนต์สองล้อด้วยตัวคุมค่ากำลังสอง  
เชิงเส้น และตัวควบคุมโมเดลพรีดิกทีฟ ปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
(วิศวกรรมเครื่องกล) สาขาวิศวกรรมเครื่องกล ภาควิชาวิศวกรรมเครื่องกล อาจารย์ที่  
ปรึกษาวิทยานิพนธ์หลัก: รองศาสตราจารย์วิทิต ฉัตรรัตนกุลชัย, Ph.D. 80 หน้า

นักวิจัยหลายท่านได้ให้ความสนใจเกี่ยวกับหุ่นยนต์สองล้อเนื่องจากเป็นระบบที่ควบคุมได้  
ยากมีสมบัติเฟสไม่ต่ำที่สุด, มีหลายอินพุตหลายเอาต์พุตและมีอินพุตน้อยกว่าเอาต์พุต ซึ่งเหมาะ  
สำหรับงานวิจัยที่ต้องการแสดงถึงประสิทธิภาพของระบบควบคุมที่นำมาใช้ นอกจากนี้หุ่นยนต์  
สองล้อมีความสามารถในการเคลื่อนตัวที่คล่องแคล่วคล้ายคลึงกับมนุษย์ ซึ่งมีหลายงานวิจัยที่แสดง  
ถึงประสิทธิภาพในการเคลื่อนตัวของหุ่นยนต์ และจากความคล่องแคล่วของหุ่นยนต์นี้สามารถ  
นำไปประยุกต์ใช้งานจริงแทนมนุษย์ได้

งานวิจัยนี้ได้นำตัวคุมค่ากำลังสองเชิงเส้นและตัวควบคุมโมเดลพรีดิกทีฟที่มีลักษณะที่  
แตกต่างกัน โดยตัวควบคุมทั้งสองชนิดต้องอาศัยแบบจำลองคณิตศาสตร์ที่แม่นยำเพื่อใช้ในการ  
จัดการกับความไม่แน่นอนของระบบและรักษาสมดุลของหุ่นยนต์ ซึ่งในการวัดมุมเอียงของหุ่นยนต์  
สองล้อถือเป็นปัญหาสำคัญในการรักษาสมดุลของหุ่นยนต์สองล้อ ในงานวิจัยนี้ใช้เซนเซอร์ร่วมกัน  
ระหว่างเซนเซอร์วัดความเร็วและเซนเซอร์วัดความเร็วเชิงมุม

จากผลการจำลองแสดงถึงเสถียรภาพในการรักษาสมดุลหุ่นยนต์สองล้อของตัวคุมค่ากำลัง  
สองเชิงเส้นและตัวควบคุมโมเดลพรีดิกทีฟที่มีลักษณะที่แตกต่างกัน ได้แสดงถึงคุณลักษณะเฉพาะ  
ของตัวควบคุมโมเดลพรีดิกทีฟในเรื่องข้อจำกัดของอินพุตและเอาต์พุต ซึ่งสามารถนำไป  
ประยุกต์ใช้กับงานในโรงงานอุตสาหกรรมที่มีการใช้พลังงานมากได้ ในส่วนของผลการทดลองได้  
แสดงประสิทธิภาพของตัวกรองความถี่ Complementary ที่สามารถจัดการกับปัญหา noise และ  
drift จากเซนเซอร์ทั้งสองชนิด

ลายมือชื่อนิสิต

ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

Pongskorn Kaveesontornsano 2014: Self-Balancing of Two-Wheeled Mobile Robot with Linear Quadratic Regulator and Model Predictive Controller. Master of Engineering (Mechanical Engineering), Major Field: Mechanical Engineering, Department of Mechanical Engineering. Thesis Advisor: Associate Professor Withit Chatlatanagulchai, Ph.D. 80 pages.

Many researchers have paid attention on two-wheeled robot because it is a system that is difficult to control due to its non-minimum phase property, its having multiple inputs and outputs, and the number of inputs is fewer than the number of output. The two-wheeled robot is suitable for research that needs to prove the efficiency of the applying control system. Besides, the two-wheeled robot is able to move swiftly similar to human. There exist many research works showing the performance in robot movement. From this agility of the robot movement, the two-wheeled robot can be applied to real work in place of humans.

This research applies the linear quadratic regulator and the model predictive control techniques to controlling the robot. Each technique has different characteristics. The linear quadratic regulator depends on accurate mathematical model to handle the system uncertainty and to balance the robot. Tilt angle measurement of the robot is a major problem in balancing the two-wheeled robot. In this research, an accelerometer and a gyroscope are used together.

Simulation results show the stability in balancing the two-wheeled robot of the linear quadratic regulator and the model predictive controller. The two types of controller are different. The results show specific characteristic of each controller. The model predictive controller incorporates constraints on inputs and outputs. It can be applied to industrial work that consumes high energy. Experimental results show the efficiency of the complementary filter that can handle noise and sensor drifts for both sensors.

---

Student's signature

---

Thesis Advisor's signature

\_\_ / \_\_ / \_\_

## กิตติกรรมประกาศ

ข้าพเจ้าขอกราบขอบพระคุณ รศ.ดร. วิทิต ฉัตรรัตนกุลชัย ประธานกรรมการที่ปรึกษา  
วิทยานิพนธ์ ที่ให้โอกาสข้าพเจ้าเข้ามาสู่สาขาวิชาการควบคุม ได้มอบความรู้พื้นฐานการควบคุม  
จนกระทั่งระดับสูง ได้ให้คำปรึกษาทางด้านทฤษฎีและการปฏิบัติตลอดจนประสบการณ์ต่างๆ และ  
ขอกราบขอบพระคุณ ผศ.ดร. ทวีเดช ศิริธนาพิพัฒน์ กรรมการที่ปรึกษาวิชาเอก ที่ให้ความรู้และ  
คำแนะนำต่างๆมาโดยตลอด จนกระทั่งวิทยานิพนธ์นี้เสร็จสมบูรณ์

ขอขอบพระคุณมหาวิทยาลัยเกษตรศาสตร์, ห้องปฏิบัติการควบคุมหุ่นยนต์และการ  
สั่นสะเทือน (Control of Robot and Vibration Laboratory, CRVLAB) ที่เอื้อเฟื้อสถานที่และ  
อุปกรณ์การทำวิจัย อาจารย์คณะวิศวกรรมศาสตร์ทุกท่านที่ได้ให้คำแนะนำและวิชาความรู้ต่างๆ  
ขอขอบพระคุณ รศ.ดร. อนุชา หิรัญวัฒน์ ที่ให้คำแนะนำในการเรียนระดับปริญญาโท ขอขอบคุณรุ่นพี่  
และเพื่อนๆ ใน CRVLAB ที่ให้คำปรึกษาและคำแนะนำเมื่อเกิดปัญหาในการทำงานวิจัยนี้  
นอกจากนี้ขอขอบคุณความรู้ต่างๆจากนักวิจัยทุกท่านที่เกี่ยวข้องที่ให้ความรู้ในงานประชุมวิชาการ,  
บทความวิชาการและวารสารวิชาการ

สุดท้ายนี้ ข้าพเจ้าขอกล่าวคำขอบพระคุณแต่บิดาและมารดาที่เลี้ยงดูข้าพเจ้า คอยอบรมสั่ง  
สอนและให้คำแนะนำในการดำเนินชีวิตที่ดีมาโดยตลอด ขอขอบคุณ พี่ๆ เพื่อนๆ น้องๆ และทุกคน  
ที่เกี่ยวข้องที่คอยชี้แนะและให้กำลังใจ จนกระทั่งวิทยานิพนธ์เล่มนี้เสร็จสมบูรณ์

พงศกร กวีสุนทรเสนาะ

กรกฎาคม 2557

## สารบัญ

	หน้า
สารบัญ	(1)
สารบัญตาราง	(2)
สารบัญภาพ	(3)
คำอธิบายสัญลักษณ์และอักษรย่อ	(6)
คำนำ	1
วัตถุประสงค์	3
การตรวจเอกสาร	4
อุปกรณ์และวิธีการ	13
อุปกรณ์	13
วิธีการ	19
ผลและวิจารณ์	46
สรุปและข้อเสนอแนะ	58
สรุป	58
ข้อเสนอแนะ	59
เอกสารและสิ่งอ้างอิง	60
ภาคผนวก	64
ภาคผนวก ก การหาค่าพารามิเตอร์ด้วยโปรแกรม Solidworks	65
ภาคผนวก ข การหาแบบจำลองทางคณิตศาสตร์ของหุ่นยนต์สองล้อและอัตรา ขยายของตัวคุมค่ากำลังสองเชิงเส้นด้วยโค้ดโปรแกรม MATLAB	67
ภาคผนวก ค การจำลองผลการทดลองด้วยตัวคุมค่ากำลังสองเชิงเส้น	71
ภาคผนวก ง การหาผลการจำลองด้วยตัวคุมโมเดลพรีดิกทีฟ	73
ภาคผนวก จ โค้ดโปรแกรม MATLAB ของผลการทดลอง	76
ประวัติการศึกษา และการทำงาน	80

## สารบัญตาราง

ตารางที่

หน้า

1 ค่าพารามิเตอร์ทั่วไปของหุ่นยนต์

20



## สารบัญภาพ

ภาพที่		หน้า
1	(ซ้าย) หุ่นยนต์สำรวจดาวอังคาร (Curiosity) (กลาง) หุ่นยนต์เสิร์ฟอาหาร (Hajime Robot) (ขวา) หุ่นยนต์เดินสองขาเสมือนมนุษย์ (Humanoid robots)	4
2	(ซ้าย) หุ่นยนต์ล้อเดียว (Unicycle Robot) (กลาง) Lego NXTbike-GS (Self-Balancing Two Wheeled Robot)	5
3	(ซ้าย) หุ่นยนต์สองล้อที่สามารถทรงตัวด้วยตัวเอง (Quasimoro) (กลาง) หุ่นยนต์สองล้อที่สามารถเปลี่ยนจุดศูนย์กลางมวล (ขวา) หุ่นยนต์สองล้อที่มีแขนสองข้าง	7
4	การเคลื่อนที่ลงบันไดของหุ่นยนต์สองล้อ	9
5	หลักการ Complementary filter	11
6	หุ่นยนต์สองล้อและส่วนประกอบต่างๆ	13
7	อุปกรณ์เอนโคเดอร์เซนเซอร์	14
8	อุปกรณ์เซนเซอร์วัดความเร่ง	14
9	อุปกรณ์เซนเซอร์วัดความเร็วเชิงมุม (Rate Gyroscope)	15
10	อุปกรณ์วงจรขับเคลื่อนมอเตอร์กระแสตรง	15
11	มอเตอร์ไฟฟ้ากระแสตรง (DC Motor)	16
12	อุปกรณ์ไมโครคอนโทรลเลอร์ (Arduino Mega 2560 R3)	16
13	อุปกรณ์แบตเตอรี่ 12 V	17
14	ตัวอย่างการใช้งานโปรแกรม MATLAB	17
15	ตัวอย่างการใช้โปรแกรม Solidworks	18
16	โครงสร้างของหุ่นยนต์สองล้อจากโปรแกรม Solidworks	19
17	หุ่นยนต์สองล้อบนแกนพิกัด x,y,z	20
18	การวัดมุมของเซนเซอร์วัดความเร่ง	29
19	ตัวกรองสัญญาณแบบดิจิทัล (Digital Filter)	30
20	บล็อกไดอะแกรมการวัดค่าจากเซนเซอร์ทั้งสามชนิด	31
21	บล็อกไดอะแกรมในการควบคุมป้อนกลับแบบรักษาเสถียรภาพ	32
22	พื้นฐานของตัวควบคุมโมเดลพรีดิกทีฟ	36
23	เอาต์พุตของตัวคุมกำลังสองเชิงเส้น	46

## สารบัญญภาพ (ต่อ)

ภาพที่		หน้า
24	สัญญาณควบคุมของตัวคุมค่ากำลังสองเชิงเส้น	47
25	ตัวควบคุมโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด (MPC without Constraints)	48
26	สัญญาณควบคุมของโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด (MPC without constraints)	48
27	ตัวควบคุมโมเดลพรีดิกทีฟที่มีข้อจำกัด (MPC with Constraints ( $\pm 0.01$ Pitch))	49
28	สัญญาณควบคุมของตัวควบคุม โมเดลพรีดิกทีฟที่มีข้อจำกัด (MPC with constraints ( $\pm 0.01$ Pitch))	49
29	ตัวควบคุมโมเดลพรีดิกทีฟ (MPC ; $H_p = 24$ )	50
30	สัญญาณควบคุมของตัวควบคุม โมเดลพรีดิกทีฟ (MPC ; $H_p = 24$ )	50
31	เปรียบเทียบผลการจำลองระหว่างตัวคุมค่ากำลังสองเชิงเส้นและตัวควบคุม โมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด	51
32	เปรียบเทียบผลการจำลองระหว่างตัวคุมค่ากำลังสองเชิงเส้นและตัวควบคุม โมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด ( $\pm 0.01$ Pitch)	52
33	ภาพรวมของการวัดมุมเอียงด้วยหลักการ Complementary Filter	53
34	เปรียบเทียบผลระหว่างเซนเซอร์วัดความเร่งและ Complementary Filter	53
35	เปรียบเทียบผลของมุมจากเซนเซอร์ทั้งสองและ Complementary Filter ที่ความถี่สูง	54
36	เปรียบเทียบผลระหว่างเซนเซอร์วัดความเร็วเชิงมุมและ Complementary Filter	54
37	ค่าเอาต์พุตของมุม Pitch จากการรักษาสมดุลของหุ่นยนต์สองล้อ	56
38	สัญญาณควบคุม PWM จากการรักษาสมดุลของหุ่นยนต์สองล้อ	56

## สารบัญภาพ (ต่อ)

ภาพผนวกที่	หน้า
ก1 การหาค่าพารามิเตอร์ด้วยโปรแกรม Solidworks	66
ค1 การจำลองผลการทดลองด้วยตัวคุมค่ากำลังสองเชิงเส้น	72
ง1 การหาผลการจำลองด้วยตัวควบคุมโมเดลพรีดิกทีฟ	74
ง2 การกำหนดค่าพารามิเตอร์ของตัวควบคุมโมเดลพรีดิกทีฟ	75
จ1 การรับค่าจากเซนเซอร์วัดความเร่งและเซนเซอร์วัดความเร็วเชิงมุม	77
จ2 การรับค่าจากเอนโค้ดเดอร์เซนเซอร์ด้วย S-function และอัตราขยาย	77
จ3 การควบคุมมอเตอร์ซ้ายและมอเตอร์ขวาด้วย PWM	78
จ4 การวัดมุมเอียงด้วยหลักการ complementary filter (Subsystem)	78
จ5 การวัดตำแหน่งและมุมเอียงของเอนโคคเดอร์เซนเซอร์ (Subsystem1 และ Subsystem2)	79
จ6 การใช้ S-function ในการวัดค่าจากเอนโค้ดเดอร์เซนเซอร์	79

## คำอธิบายสัญลักษณ์ และอักษรย่อ

### คำอธิบายสัญลักษณ์

$\theta$	=	มุมเฉลี่ยของล้อซ้ายและล้อขวา ( $rad$ )
$\theta_l$	=	มุมของล้อซ้าย ( $rad$ )
$\theta_r$	=	มุมของล้อขวา ( $rad$ )
$\psi$	=	มุมเอียงของหุ่นยนต์สองล้อ ( $rad$ )
$\phi$	=	มุมเลี้ยวของหุ่นยนต์สองล้อ ( $rad$ )
$\tau$	=	เวลาคงที่ของ Complementary Filter (time constant)
$\varepsilon$	=	ค่าความผิดพลาดของตัวควบคุมโมเดลพรีดิกทีฟ
$\Phi$	=	สัมประสิทธิ์อัตราขยายของโมเดลพรีดิกทีฟ
$\in$	=	สัญญาณรบกวนสีขาว
$a$	=	ค่าสัมประสิทธิ์ของตัวกรองความถี่
$D$	=	ระยะความลึกของตัวถัง ( $m$ )
$F$	=	สัมประสิทธิ์อัตราขยายของโมเดลพรีดิกทีฟ
$F_\theta$	=	แรงภายนอกที่กระทำในระบบพิกัด $\theta$ ( $N.m$ )
$F_\psi$	=	แรงภายนอกที่กระทำในระบบพิกัด $\psi$ ( $N.m$ )
$F_\phi$	=	แรงภายนอกที่กระทำในระบบพิกัด $\phi$ ( $N.m$ )
$F_l$	=	แรงภายนอกที่กระทำกับล้อซ้าย ( $N.m$ )
$F_r$	=	แรงภายนอกที่กระทำกับล้อขวา ( $N.m$ )
$f_m$	=	ค่าสัมประสิทธิ์แรงเสียดทานระหว่างตัวถังและมอเตอร์
$f_w$	=	ค่าสัมประสิทธิ์แรงเสียดทานระหว่างล้อและพื้น
$g$	=	แรงโน้มถ่วงโลก ( $m/sec^2$ )
$H$	=	ระยะความสูงของตัวถัง ( $m$ )
$H_p$	=	ขอบเขตการทำนายสเตทในอนาคต
$H_c$	=	ขอบเขตการทำนายอินพุตในอนาคต
$i$	=	อันดับของระบบ
$i_{l,r}$	=	กระแสของมอเตอร์ซ้ายหรือมอเตอร์ขวา
$J$	=	ครรชนี่สมรรถนะ (Cost function)
$J_{by}$	=	โมเมนต์ความเฉื่อยตัวถังแกน $y_0$ ( $kgm^2$ )

## คำอธิบายสัญลักษณ์ และอักษรย่อ (ต่อ)

### คำอธิบายสัญลักษณ์

$J_{bz}$	=	โมเมนต์ความเฉื่อยตัวถังแกน $z_0$ ( $kgm^2$ )
$J_m$	=	โมเมนต์ความเฉื่อยของมอเตอร์ ( $Nm / A$ )
$J_w$	=	โมเมนต์ความเฉื่อยของล้อ ( $kgm^2$ )
$K$	=	อัตราขยายของตัวควบคุม
$K_b$	=	ค่าคงที่แรงดันไฟฟ้าย้อนกลับมอเตอร์
$K_t$	=	ค่าคงที่แรงบิดมอเตอร์
$k$	=	เวลาปัจจุบัน
$L$	=	ระยะจากแกนล้อถึงศูนย์กลาง ( $m$ )
$L_g$	=	ค่าลากรางจ์
$L_m$	=	ค่าของตัวเหนี่ยวนำ
$M$	=	มวลตัวถัง ( $kg$ )
$m$	=	มวลล้อ ( $kg$ )
$n$	=	อัตราทดมอเตอร์
$Q$	=	เมตริกซ์ถ่วงน้ำหนักของสเตท
$R$	=	รัศมีล้อ ( $m$ ), เมตริกซ์ถ่วงน้ำหนักของอินพุต
$R_m$	=	ค่าความต้านทานของมอเตอร์
$r$	=	สัญญาณอ้างอิงที่ต้องการตามรอยสัญญาณ
$s$	=	จุดที่ต้องการตามรอยสัญญาณ
$T$	=	พลังงานจลน์รวมของระบบ ( $kgm^2 / s^2$ )
$T_1$	=	พลังงานจลน์รวมจากการเคลื่อนที่เชิงเส้น ( $kgm^2 / s^2$ )
$T_2$	=	พลังงานจลน์รวมจากการเคลื่อนที่แบบหมุน ( $kgm^2 / s^2$ )
$T_{ref}$	=	ความเร็วในการตอบสนอง
$T_s$	=	ระยะห่างของช่วงเวลา
$q_i$	=	ระบบพิกัด
$\Delta U$	=	เวกเตอร์ของผลต่างอินพุต
$u$	=	อินพุต ( $V$ )
$V$	=	พลังงานศักย์รวมของระบบ ( $kgm^2 / s^2$ )

## คำอธิบายสัญลักษณ์ และอักษรย่อ (ต่อ)

### คำอธิบายสัญลักษณ์

$V_{ref}$	=	แรงดันที่ใช้ปฏิบัติการทำงานของไมโครคอนโทรลเลอร์
$v_{l,r}$	=	แรงดันของมอเตอร์ซ้ายหรือมอเตอร์ขวา
$W$	=	ความกว้างของฐานล้อ ( $m$ )
$w$	=	สัญญาณอินพุตของสิ่งรบกวนภายนอกในระบบ
$x$	=	ค่าสเตตของระบบ
$y$	=	เอาต์พุตของระบบ

### คำอธิบายอักษรย่อ

LQR	=	Linear Quadratic Regulator
MIMO	=	Multi-input Multi-output
MPC	=	Model Predictive Control
PWM	=	Pulse Width Modulation
NMP	=	Non-minimum Phase

# การรักษาสมดุลของหุ่นยนต์สองล้อด้วยตัวคุมค่ากำลังสองเชิงเส้น และตัวควบคุม โมเดลพรีดิกทีฟ

## Self-Balancing of Two-Wheeled Mobile Robot with Linear Quadratic Regulator and Model Predictive Controller

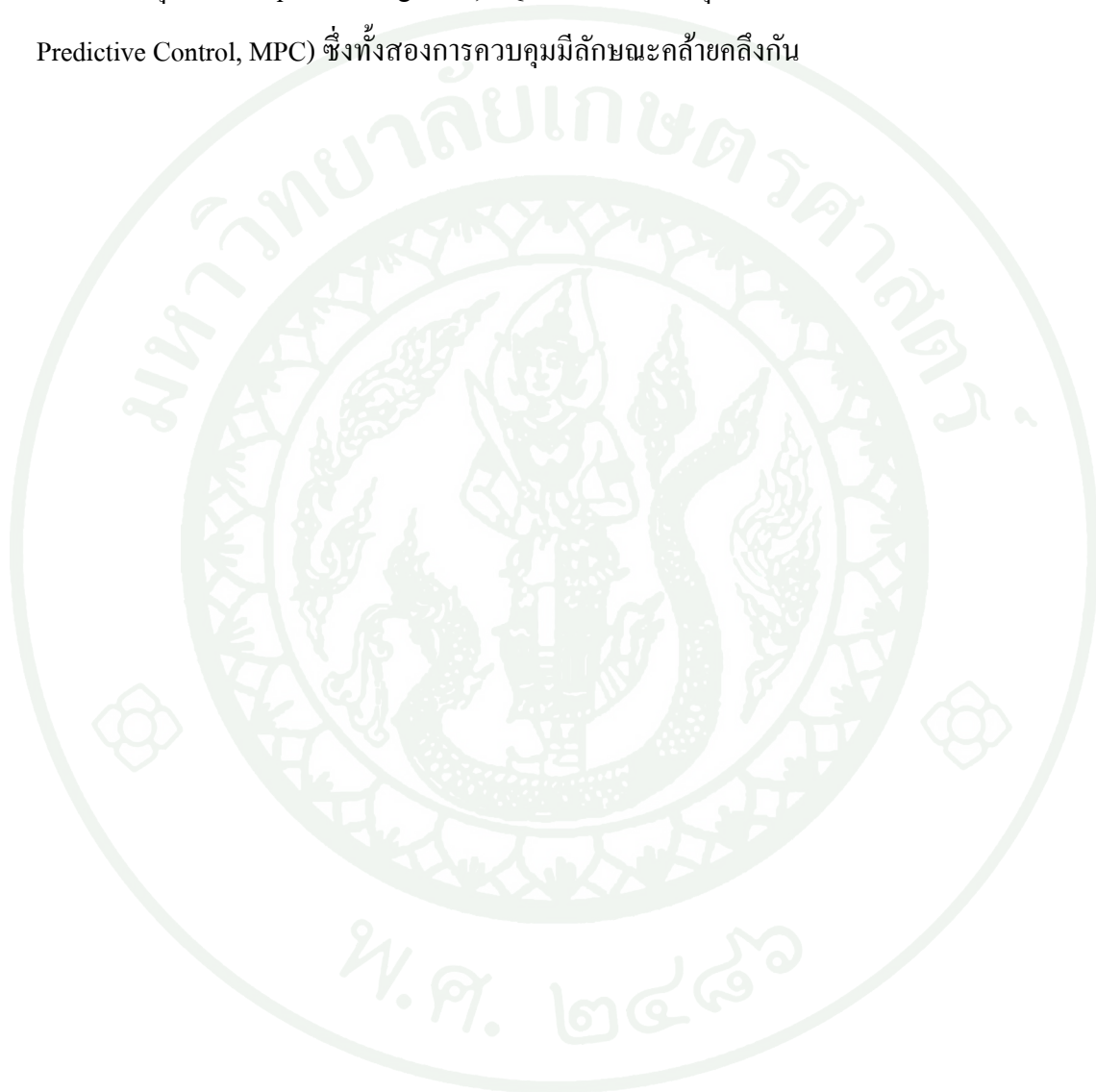
### คำนำ

หุ่นยนต์สองล้อเป็นหุ่นยนต์ที่ได้รับความสนใจอย่างแพร่หลายทางด้านการควบคุมซึ่งเราจะพบงานวิจัยที่เกี่ยวข้องเป็นจำนวนเพิ่มมากขึ้นในแต่ละปี โดยเป็นการควบคุมการรักษาสมดุลของหุ่นยนต์ซึ่งมีความยากเนื่องจากเป็นระบบที่มีความซับซ้อน เป็นระบบที่มีหลายสัญญาณป้อนเข้าและหลายสัญญาณป้อนออก (Multi-input Multi output, MIMO) โดยมีสัญญาณป้อนเข้าน้อยกว่าสัญญาณป้อนออก มีจำนวนแอกชูเอเตอร์ (actuators) น้อยกว่าจำนวนองศาอิสระ (degree of freedom) จึงทำให้เป็นระบบอันเดอร์แอกชูเอต (under-actuated) อีกทั้งยังเป็นระบบไม่ใช่มุมเฟสน้อยสุด (non-minimum phase system, NMP) และระบบไม่เป็นเชิงเส้น (nonlinear system)

จากความซับซ้อนในการควบคุมข้างต้นจึงมีงานวิจัยมากมายที่ต้องการนำเสนอเทคนิคการควบคุมหลากหลายวิธีเข้าไปจัดการกับระบบดังกล่าว ซึ่งมีสามตัวแปรหลักที่มักจะนำมาเปรียบเทียบกัน อย่างแรกคือมุมเอียง (tilt angle) ของหุ่นยนต์สองล้อหรืออาจเรียกได้ว่าการสั่นไหวของหุ่นยนต์สองล้อ (oscillations) สองคือการควบคุมความเร็วของหุ่นยนต์สองล้อ (velocity) สามคือตำแหน่งที่หุ่นยนต์สองล้อเคลื่อนที่ไป (translational)

หุ่นยนต์สองล้อเป็นหุ่นยนต์ที่มีความคล่องแคล่วเป็นหุ่นยนต์ที่สามารถนำไปประยุกต์ใช้ได้กับงานหลายรูปแบบที่เห็นเด่นชัดคือ เซกเวย์ (Segway) ซึ่งอาศัยหลักการการเปลี่ยนแปลงของจุดศูนย์กลางมวล (center of mass) นั่นคือเมื่อผู้ใช้ต้องการไปข้างหน้าหรือข้างหลังก็จะทำการโน้มตัวไปในทิศทางที่ต้องการ ซึ่งการโน้มตัวจะทำให้จุดศูนย์กลางมวลเปลี่ยนหุ่นยนต์เซกเวย์ก็จะวิ่งไปในทิศทางที่โน้มตัวเพื่อทำการรักษาสมดุล

งานวิจัยนี้ต้องการจะรักษาสมดุลของหุ่นยนต์สองล้อซึ่งการควบคุมมีลักษณะเดียวกับ ลูกตุ้มนาฬิกากลับหัวโดยเคลื่อนที่ด้วยล้อที่ขนานกันสองล้อที่เป็นอิสระต่อกัน ใช้เซนเซอร์สามประเภทคือ Accelerometer Gyroscope และ Encoder ต้องการนำตัวควบคุมที่เรียกว่า ตัวคุมค่ากำลัง สองน้อยที่สุด (linear quadratic regulator, LQR) และการควบคุมโมเดลพรีดิกทีฟ (model Predictive Control, MPC) ซึ่งทั้งสองการควบคุมมีลักษณะคล้ายคลึงกัน



## วัตถุประสงค์

1. ออกแบบการวัดมุมเอียงด้วยหลักการ Complementary filter โดยใช้เซนเซอร์ประเภท Accelerometer และ Gyroscope เพื่อใช้เป็นสัญญาณป้อนกลับสำหรับหุ่นยนต์สองล้อ
2. ออกแบบระบบควบคุมของหุ่นยนต์สองล้อที่สามารถรักษาสมดุลและรักษาเสถียรภาพของระบบได้ โดยใช้การออกแบบด้วยตัวควบคุมกำลังสองเชิงเส้นและตัวควบคุม โมเดลพรีดิกทีฟ

## ขอบเขตงานวิจัย

1. ศึกษาการวัดมุมเอียงที่รอบจุดศูนย์ของหุ่นยนต์สองล้อและการจัดการกับ noise และ drift
2. ปัจจัยหลักในการวิจัยคือ การติดตามสัญญาณอ้างอิงที่กำหนดด้วยการใช้ตัวคุมค่ากำลังสองเชิงเส้น และตัวควบคุม โมเดลพรีดิกทีฟ

## การตรวจเอกสาร

### 1. หุ่นยนต์สองล้อ

ในปัจจุบันนี้มีจำนวนหุ่นยนต์เพิ่มมากขึ้นอยู่ทั่วไปทุกหนทุกแห่งและนำไปใช้ในงานที่หลากหลาย อาทิเช่น หุ่นยนต์สำรวจบนต่างดาว, หุ่นยนต์ให้ความช่วยเหลือ, หุ่นยนต์เสิร์ฟอาหาร ดังภาพที่ 1 ในขณะที่หุ่นยนต์แบบใช้ขาเดิน (Legged Robots) สามารถเคลื่อนที่ข้ามผ่านสิ่งกีดขวางต่างๆ ได้ดีกว่าหุ่นยนต์ที่ใช้ล้อในการขับเคลื่อน (Wheeled Robots) โครงสร้างทางกลของหุ่นยนต์แบบใช้ขาเดินจะมีความซับซ้อนในการออกแบบหุ่นยนต์และมีจำนวนองศาอิสระ (degrees of freedom) ที่มากกว่าหุ่นยนต์ที่ใช้ล้อในการขับเคลื่อน แต่หุ่นยนต์ที่ใช้ล้อในการขับเคลื่อนจะมีประสิทธิภาพการเคลื่อนที่ได้เร็วกว่าและมีโครงสร้างทางกลที่ง่ายกว่า อย่างไรก็ตามหุ่นยนต์ที่ใช้ล้อในการขับเคลื่อนนั้นควบคุมได้ง่ายกว่าหุ่นยนต์แบบใช้ขาเดิน



ภาพที่ 1 (ซ้าย) หุ่นยนต์สำรวจดาวอังคาร (Curiosity) (กลาง) หุ่นยนต์เสิร์ฟอาหาร (Hajime Robot) (ขวา) หุ่นยนต์เดินสองขาเสมือนมนุษย์ (Humanoid robots)

ที่มา: วิกีพีเดีย. (2556)

ซึ่งงานวิจัยนี้สนใจเกี่ยวกับหุ่นยนต์ที่ใช้ล้อในการขับเคลื่อนจำนวนสองล้อ สำหรับหุ่นยนต์ที่มีจำนวนล้อมากกว่าสองนั้นต้องการเงื่อนไขให้กับหุ่นยนต์ยกตัวอย่างเช่น ระบบกันสะเทือน (suspension system) พื้นที่ที่หุ่นยนต์เคลื่อนที่ ซึ่งในหุ่นยนต์สองล้อสามารถเคลื่อนที่ได้คล่องแคล่ว เช่น การหมุนตัวเองอยู่กับที่ ทำให้สามารถเลี้ยวผ่านมุมที่แคบได้และการเปลี่ยนแปลง

ตำแหน่งจุดศูนย์กลางของหุ่นยนต์ที่มีประโยชน์สำหรับการขึ้นลงทางลาดชัน เป็นต้น และหุ่นยนต์สองล้อยังมีลักษณะการทรงตัวที่ใกล้เคียงมนุษย์มาก

หุ่นยนต์ที่ใช้ล้อในการขับเคลื่อนนั้นใช้พลังงานในการขับเคลื่อนน้อยและเคลื่อนที่ได้รวดเร็ว มีโครงสร้างทางกลที่ง่ายทำให้ควบคุมเสถียรภาพของระบบได้ง่ายเช่นกัน อย่างไรก็ตาม หุ่นยนต์ที่ใช้ล้อในการขับเคลื่อนนั้นก็ยากที่จะเคลื่อนที่ไปบนพื้นที่ที่มีความขรุขระหรือสภาพพื้นที่ที่ไม่เรียบสม่ำเสมอ โครงสร้างของหุ่นยนต์ที่ใช้ล้อในการขับเคลื่อนที่นิยมใช้กันมากในปัจจุบันอาจแบ่งออกเป็น 4 ประเภท โดย Siciliano and Khatib (2008) ตามจำนวนล้อของหุ่นยนต์ ประเภทแรก หุ่นยนต์ล้อเดียวเป็นหุ่นยนต์ควบคุมได้ยากมีความไม่แน่นอนสูง (unstable) ยากต่อการรักษาสมดุลของหุ่นยนต์ ซึ่งต้องทำการรักษาสมดุลสองแกนคือแกนพิท (pitch) และแกนโรล (roll) ในแกนพิทมีการควบคุมคล้ายคลึงกับลูกตุ้มนาฬิกากลับหัวเหมือนกับหุ่นยนต์สองล้อ ส่วนแกนโรล Lee *et al.* (2013) ได้ใช้แผ่นจานกลม (Disk) มารักษาสมดุล ดังภาพที่ 2 (ซ้าย) การรักษาสมดุลทั้งสองแกนมีส่วนทำให้หุ่นยนต์ล้อเดียวนั้นยังนำไปประยุกต์ใช้กับงานจริงได้ยาก



ภาพที่ 2 (ซ้าย) หุ่นยนต์ล้อเดียว (Unicycle Robot) (กลาง) Lego NXTbike-GS (ขวา) NXTway-GS (Self-Balanceing Two Wheeled Robot)

ที่มา: Lee *et al.* (2013); Yomamoto *et al.* (2008)

ประเภทที่สองคือหุ่นยนต์สองล้อ โดยทั่วไปจะแบ่งออกเป็นสองลักษณะ ลักษณะแรกจะมีลักษณะเหมือนกับรถจักรยาน (bicycle-type robot) ดังภาพที่ 2 (กลาง) หุ่นยนต์สองล้อลักษณะนี้ การรักษาสมดุลของหุ่นยนต์จะแปรผันตรงกับความเร็วของหุ่นยนต์ ดังนั้นกรณีที่หุ่นยนต์ลักษณะนี้

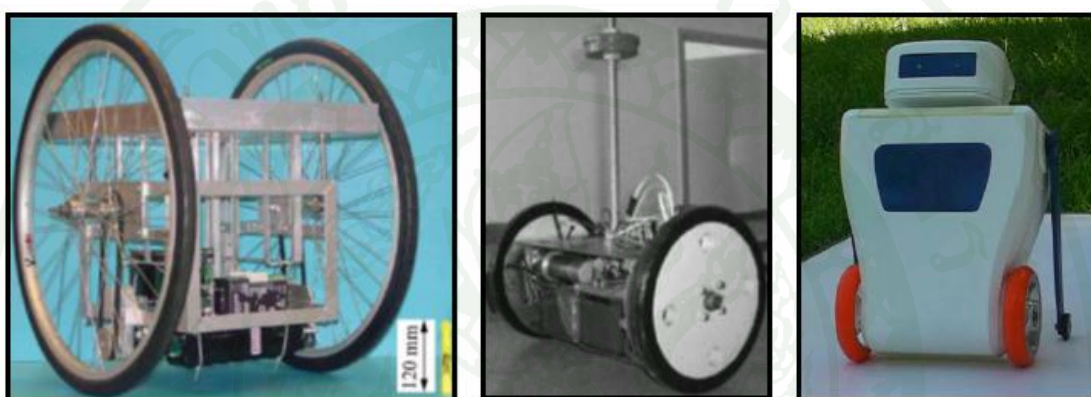
ต้องการจะรักษาตำแหน่งเดิมก็จะทำให้การรักษาสมดุลของหุ่นยนต์นั้นทำได้ยาก ส่วนหุ่นยนต์สองล้อลักษณะที่สองซึ่งเราสนใจในงานวิจัยนี้ดังภาพที่ 2 (ขวา) จะสามารถรักษาสมดุลของหุ่นยนต์ที่ตำแหน่งเดิมได้ดี ดังลักษณะการเคลื่อนที่ดังที่กล่าวไปข้างต้นทำให้หุ่นยนต์สองล้อมีข้อได้เปรียบหลายกรณี

ประเภทที่สามหุ่นยนต์สามล้อเป็นหุ่นยนต์ที่ไม่ต้องทำการรักษาสมดุล มีการออกแบบหุ่นยนต์ประเภทนี้หลายลักษณะยกตัวอย่างเช่น หุ่นยนต์สามล้อที่ขับเคลื่อนสองล้อและล้อที่สามเป็นลูกล้อเลื่อน (caster wheel) หุ่นยนต์สามล้อลักษณะนี้การเคลื่อนที่คล้ายคลึงกับหุ่นยนต์สองล้อ แต่ต้องการขนาดของหุ่นยนต์ที่กว้างกว่า หากต้องการเคลื่อนที่บนพื้นที่เรียบไม่สม่ำเสมอ ซึ่งในแต่ละลักษณะของหุ่นยนต์สามล้อจะมีข้อดีและข้อเสียตามการออกแบบของหุ่นยนต์สามล้อ ประเภทที่ดีคือหุ่นยนต์สี่ล้อได้มีการออกแบบหุ่นยนต์ประเภทนี้หลายลักษณะเช่นกันลักษณะที่นิยมกันคือโครงสร้างเหมือนรถยนต์ซึ่งมีเสถียรภาพในขณะที่เคลื่อนที่ด้วยความเร็วสูง แต่การหมุนของล้อหน้าทั้งสองไม่สามารถหมุนถึง 90 องศาทำให้การเลี้ยวหุ่นยนต์ต้องการพื้นที่มาก

การออกแบบโครงสร้างของหุ่นยนต์สองล้อมีความสำคัญต่อการรักษาสมดุลซึ่งมีความสัมพันธ์กับตำแหน่งของจุดศูนย์กลางของมวล (Center of mass) Ferdinando *et al.* (2011) ได้กล่าวว่าการให้จุดศูนย์กลางมวลของหุ่นยนต์นั้นอยู่สูงกว่าจุดหมุนก็จะทำให้เป็นหุ่นยนต์ที่รักษาสมดุลได้ยาก ถ้าให้จุดศูนย์กลางมวลอยู่ต่ำกว่าจุดหมุนก็จะทำให้เป็นหุ่นยนต์ที่รักษาสมดุลได้ด้วยตัวเองแต่จะทำให้การตอบสนองของมอเตอร์ช้าลง และได้ทำการปรับเปลี่ยนจุดศูนย์กลางมวลด้วยหุ่นยนต์ที่ทำจาก LEGO ทำให้ง่ายในการปรับเปลี่ยนและได้ทำการทดลองใส่โพลดที่มีน้ำหนักไว้ต่ำกว่าจุดหมุน ซึ่งการผลการทดลองที่ทำการควบคุมด้วย PID controller แสดงให้เห็นว่าการใส่โพลดน้ำหนักที่ต่ำกว่าจุดหมุนช่วยเพิ่มประสิทธิภาพในการรักษาสมดุล

Brkic and Kovacic (2009) ได้กล่าวถึงหลักการรักษาสมดุลของหุ่นยนต์สองล้อว่าในหลักเบื้องต้นนั้นขึ้นอยู่กับตำแหน่งของจุดศูนย์กลางของมวล (Center of mass) ซึ่งการย้ายจุดศูนย์กลางมวลให้อยู่สูงกว่าจุดที่หุ่นยนต์สองล้อรักษาสมดุลจะทำให้เกิดปัญหาในการรักษาสมดุลของหุ่นยนต์เพิ่มมากขึ้น และได้ให้ความสำคัญกับตำแหน่งของจุดศูนย์กลางมวลของหุ่นยนต์สองล้อจึงสร้างหุ่นยนต์สองล้อที่สามารถปรับตำแหน่งของจุดศูนย์กลางมวลดังภาพที่ 3 (กลาง) เพื่อทำให้ง่ายต่อ

การหาขอบเขตของการรักษาสมดุลของหุ่นยนต์สองล้อ ซึ่งสามารถเคลื่อนย้ายจุดศูนย์กลางมวลให้อยู่สูงกว่าจุดหมุนหรือต่ำกว่าจุดหมุนทำให้ส่งผลต่อระบบกลายเป็นระบบที่มีความเสถียรหรือไม่มีความเสถียรได้ตามต้องการ ซึ่งเป็นการปรับโครงสร้างของหุ่นยนต์เท่านั้น และได้ทำการแยกคิดตัวแปรหลักแบ่งออกเป็นสามส่วนคือ มุมเอียง ความเร็ว และตำแหน่งของหุ่นยนต์ โดยใช้ตัวควบคุมแบบสัดส่วนบวกอินทิกรัล (PI control) ในส่วนของมุมเอียงและความเร็ว และใช้ขั้นตอนการแสวงหาเส้นทาง (pure pursuit algorithm) สำหรับตำแหน่งของหุ่นยนต์สองล้อ



ภาพที่ 3 (ซ้าย) หุ่นยนต์สองล้อที่สามารถทรงตัวด้วยตัวเอง (Quasimoro) (กลาง) หุ่นยนต์สองล้อที่สามารถเปลี่ยนจุดศูนย์กลางมวล (ขวา) หุ่นยนต์สองล้อที่มีแขนสองข้าง

ที่มา: Oryschuk *et al.* (2009); Brkic and Kovacic (2009); Feng *et al.* (2011)

หุ่นยนต์สองล้อได้ถูกคาดหวังว่าจะสามารถปฏิบัติงานได้ในหลากหลายรูปแบบอาทิเช่น ทำงานเกี่ยวกับการขนส่งภายในบ้านหรือโรงพยาบาล ซึ่งปัจจัยหลักที่จะดำเนินการได้สำเร็จคือ หุ่นยนต์สองล้อต้องเคลื่อนที่ไปยังตำแหน่งที่ถูกกำหนดไว้ได้แม่นยำและสามารถหลบสิ่งกีดขวางภายในพื้นที่ปฏิบัติงานนั้นได้ Deng *et al.* (2010) ได้อธิบายเกี่ยวกับการหลบสิ่งกีดขวางของหุ่นยนต์สองล้อแบ่งออกเป็นสองหัวข้อหลักคือ การวางแผนในการเคลื่อนที่ (motion planning) และตัวควบคุมป้อนกลับ (feedback control) ได้มีนักวิจัยหลายท่านทำการศึกษาวิจัยเกี่ยวกับการวางแผนในการเคลื่อนที่โดย Ge and Cui (2000) ได้ใช้หลักในการกำหนดพื้นที่ที่หุ่นยนต์จะเคลื่อนที่ไปให้ถึงเป้าหมายโดยไม่ชนกับสิ่งกีดขวาง ข้อดีของหลักการนี้คือมีสมการคณิตศาสตร์ (mathematical) ที่ง่ายและไม่ซับซ้อน ในส่วนของตัวควบคุมป้อนกลับก็มีนักวิจัยหลายท่านได้ทำการศึกษาและวิจัยไว้

โดย Urakubo *et al.* (2004) อธิบายถึงหลักการว่าเป็นการสร้างฟังก์ชันจำลองขึ้นมาแทนสิ่งกีดขวาง และใช้ตัวควบคุมป้อนกลับในการสั่งให้หุ่นยนต์สองล้อเคลื่อนที่ไปยังจุดที่ต้องการโดยไม่ชนกับสิ่งกีดขวาง Deng *et al.* (2010) ได้นำทั้งสองหัวข้อมาผสมผสานกันจึงทำให้มีประสิทธิภาพในหลบสิ่งกีดขวางและเคลื่อนที่ไปยังจุดหมายได้แม่นยำมากขึ้น

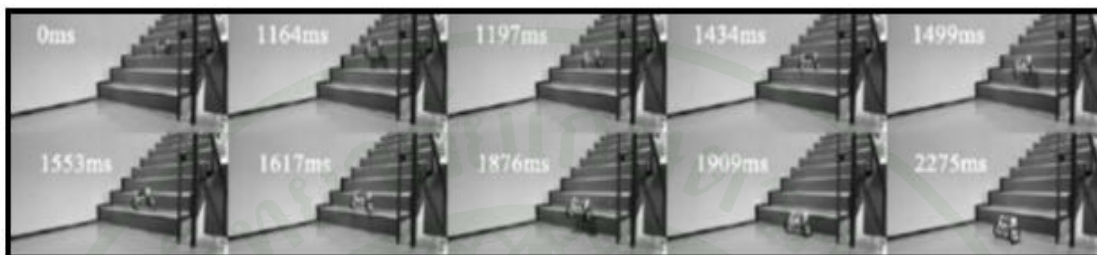
ในกรณีที่หุ่นยนต์สองล้อนั้นแบตเตอรี่กำลังจะหมดหรือมีแรงกระทำภายนอกที่รุนแรงก็จะทำให้หุ่นยนต์สองล้อไม่สามารถรักษาสถิตและล้มลง Feng *et al.* (2011) ได้ออกแบบหุ่นยนต์สองล้อที่มีแขนสองข้างดังภาพที่ 3 (ขวา) ซึ่งแขนทั้งสองสามารถช่วยในเรื่องการรักษาสถิตในกรณีที่ มีแรงกระทำภายนอกที่รุนแรง และช่วยประหยัดพลังงานของแบตเตอรี่ในกรณีที่ไม่ได้ใช้งานหรือ ตั้งอยู่ในงานแสดงหุ่นยนต์ ได้ใช้หลักการ Kalman filter ในการวัดมุมเอียงของหุ่นยนต์ซึ่งมีความแม่นยำและสามารถจัดการกับปัญหา noise หรือ drift ของเซนเซอร์วัดความเร่งและเซนเซอร์วัดความเร็วเชิงมุมได้ดี และทำการควบคุมตำแหน่งและการรักษาสถิตด้วยวิธีการวางตำแหน่งโพล (pole placement)

## 2. ตัวคุมค่ากำลังสองเชิงเส้น

Oryschuk *et al.* (2009) ได้ออกแบบหุ่นยนต์สองล้อที่สามารถรักษาสถิตของหุ่นยนต์ได้ ด้วยตัวหุ่นยนต์เอง (passive stabilization) ดังภาพที่ 3 (ซ้าย) โดยมีจุดศูนย์กลางมวลต่ำกว่าจุดหมุนของหุ่นยนต์สองล้อ ในการทดลองแสดงประสิทธิภาพในการรักษาเสถียรภาพของหุ่นยนต์ในขณะที่ ใส่น้ำหนัก (payload) ให้กับหุ่นยนต์สองล้อตั้งแต่ 0 นิวตันถึง 60 นิวตัน ออกแบบตัวควบคุมโดยใช้ ตัวคุมค่ากำลังสองเชิงเส้น (LQR) ทำการจูนเมตริกซ์ถ่วงน้ำหนัก (Q,R) ด้วยวิธีลองผิดลองถูก (trial and error) ทำให้หุ่นยนต์สองล้อสามารถรักษาเสถียรภาพขณะขึ้นเนินเป็นมุม 15 องศา และสามารถ ควบคุมความเร็วของหุ่นยนต์สองล้อ

สภาพแวดล้อมรอบตัวเราในปัจจุบันบนพื้นที่ที่เราอาศัยหรือเคลื่อนที่ไปนั้นไม่ใช่พื้นดินที่ เรียบสม่ำเสมอไปในทุกๆสภาพแวดล้อม อาทิเช่น บันได ระดับพื้นต่าง ขอบถนน ดังนั้นการนำ หุ่นยนต์สองล้อไปใช้จึงต้องมีการปรับปรุงในส่วนนี้ด้วยเช่นกัน Yap and Hashimoto (2012) ได้ทำ การควบคุมหุ่นยนต์สองล้อให้เคลื่อนที่ลงบันได ดังภาพที่ 3 ได้แบ่งโมเดลเป็น 3 ขั้นตอนคือ ช่วงที่ หุ่นยนต์ลอย ช่วงที่หุ่นยนต์สัมผัสพื้น และช่วง โพรเจกไทล์ (Projectile) ในส่วนของการควบคุมได้

ใช้ตัวควบคุมด้วยตัวคุมค่ากำลังสองเชิงเส้น (LQR) ซึ่งแบ่งออกเป็นสี่ส่วนคือ ก่อนลอย ขณะลอย ตกกระทบ และสมมูลของหุ่นยนต์ ได้ทำการควบคุมอินพุต (input) ที่ก่อให้เกิดการตอบสนองของ ทอร์ก (torque) ในระหว่างที่หุ่นยนต์สองล้อกำลังลอยซึ่งเป็นการเตรียมพร้อมกับการตกกระทบพื้น



ภาพที่ 4 การเคลื่อนที่ลงบันไดของหุ่นยนต์สองล้อ

ที่มา: Yap and Hashimoto (2012)

### 3. ตัวควบคุมโมเดลพรีดิกทีฟ

Ekaputri *et al.* (2012) ได้กล่าวถึงข้อดีของตัวควบคุม โมเดลพรีดิกทีฟคือสามารถทำงานได้อย่างมีประสิทธิภาพภายใต้การกำหนดข้อจำกัดให้แก่อินพุตและเอาต์พุต และข้อเสียคือมีความซับซ้อนและเพิ่มภาระในการคำนวณที่มากให้แก่ไมโครคอนโทรลเลอร์ ได้ใช้โปรแกรม MATLAB ในการคำนวณหาค่าของ horizon ที่เหมาะสมที่สุดได้เท่ากับ 9 horizons ซึ่งจากผลการจำลองแสดงให้เห็นว่าใช้เวลาในการตอบสนองได้เร็วที่สุด และได้ให้ความสำคัญกับค่าของ sampling time เนื่องจากในการทดลองจริงนั้นจะมีค่าอัตราหน่วงที่มาจากตัวขับเคลื่อน (actuator), เซนเซอร์ หรือจากการแปลงแบบจำลองทางคณิตศาสตร์แบบไม่เชิงเส้น (nonlinear) ให้เป็นแบบจำลองทางคณิตศาสตร์แบบเชิงเส้น (linear) ดังนั้นการเลือกค่าของ horizon ที่ถูกต้องจึงมีความสำคัญต่อการออกแบบตัวควบคุม โมเดลพรีดิกทีฟ

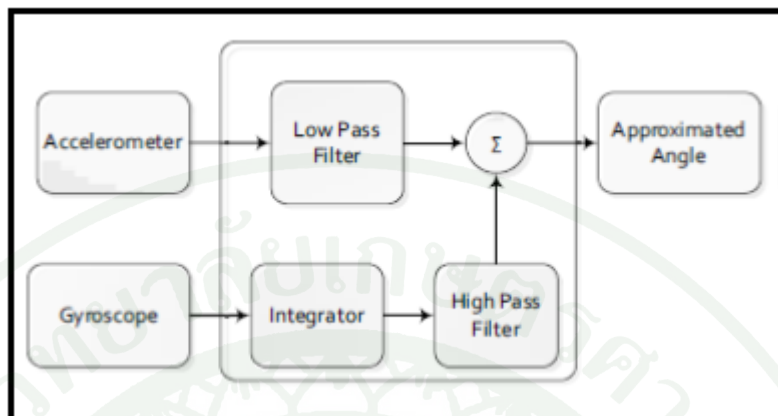
ในตัวควบคุมระดับสูงเช่น adaptive control หรือ on-line-based optimal control นั้นมักจะมีการคำนวณสูงและซับซ้อนทำให้ไมโครคอนโทรลเลอร์ประมวลผลได้ช้าลง ซึ่งปัจจุบันไมโครคอนโทรลเลอร์ได้มีการพัฒนาในเรื่องของความเร็วในการประมวลผลอย่างต่อเนื่องเช่นกัน อย่างไรก็ตามการเลือกปรับค่าพารามิเตอร์ต่างๆในตัวควบคุม โมเดลพรีดิกทีฟนั้นจึงมีความสำคัญต่อความเร็วในการประมวลผลของไมโครคอนโทรลเลอร์ Lin *et al.* (2012) ได้แสดงถึงการปรับ

ค่าพารามิเตอร์ของตัวควบคุมโมเดลพรีดิกทีฟ โดยการปรับค่าของ Prediction horizon ( $H_p$ ) ให้สูงขึ้นจะช่วยเพิ่มเสถียรภาพให้แก่เอาต์พุตและความเร็วในช่วง transience และการปรับค่าของ Control horizon ( $H_c$ ) ให้สูงขึ้นก็จะช่วยเพิ่มเสถียรภาพของเอาต์พุตเช่นกันแต่ความเร็วในช่วง transience จะช้าลง ซึ่งการปรับค่าพารามิเตอร์ทั้งสองตัวนี้มีผลโดยตรงต่อความเร็วในการประมวลผลของไมโครคอนโทรลเลอร์

งานวิจัยที่เกี่ยวกับความเร็วในการติดตามสัญญาณอ้างอิงนั้นมียกข้อจำกัดซึ่งในตัวควบคุมโมเดลพรีดิกทีฟมีความรวดเร็วในการติดตามของเอาต์พุตในช่วง transience ก่อนข้างคืออยู่แล้ว Zeng *et al.* (2013) ได้นำทฤษฎีของ Laguerre polynomials มาใช้เพื่อจัดการกับความเร็วของการติดตามสัญญาณอ้างอิง และช่วยลดภาระการคำนวณให้แก่ไมโครคอนโทรลเลอร์ ได้กำหนดข้อจำกัด (constraints) ให้แก่ความเร็วและความเร่งแก่หุ่นยนต์เพื่อป้องกันไม่ให้หุ่นยนต์ล้ม ในผลการทดลองจะแสดงให้เห็นถึงการใช้ค่า  $N = 1$  แทนใน  $N_c$  (Control horizon) = 9 ซึ่งประสิทธิภาพของเอาต์พุตที่ได้ไม่แตกต่างกันมากนักแต่ในส่วนของ  $N = 1$  จะลดภาระการคำนวณได้ถึง 7.48 ms

ในงานอุตสาหกรรมทั่วไปนั้นใช้ตัวควบคุมแบบสัดส่วนบวกอินทิกรัล (PI control) กับมอเตอร์ไฟฟ้ากระแสสลับชนิดสามเฟส (permanent magnet synchronous motor) ซึ่งไม่มีการควบคุมในส่วนข้อจำกัด (constraints) ของกระแสที่ป้อนให้กับมอเตอร์ Chai *et al.* (2013) ได้ใช้การควบคุมโมเดลพรีดิกทีฟ (MPC) นำมาใช้กับมอเตอร์ไฟฟ้ากระแสสลับชนิดสามเฟส สามารถควบคุมข้อจำกัดของกระแสและแรงดันที่ป้อนให้กับมอเตอร์ได้ มีผลให้ลดปริมาณการใช้พลังงานในโรงงานอุตสาหกรรมต่างๆ ได้ และสามารถลดสัญญาณรบกวน (noise) ที่เกิดขึ้น โดยการจูนค่า gain matrix ของกระแสแล้วความเร็วของมอเตอร์ ซึ่งงานวิจัยนี้ได้ใช้การควบคุมโมเดลพรีดิกทีฟกับหุ่นยนต์สองล้อในส่วนของการ simulation

#### 4. การวัดมุมเอียงของหุ่นยนต์



ภาพที่ 5 หลักการ Complementary filter

ที่มา: Wasif *et al.* (2013)

Cheng and Oelmann *et al.* (2010) อธิบายถึงหลักการเลือกใช้เซนเซอร์ (sensor) สิ่งที่ต้องคำนึงถึงคือ ความซับซ้อนในการใช้งาน ความแม่นยำของเซนเซอร์ ราคา ความน่าเชื่อถือ และอายุการใช้งานของเซนเซอร์ ได้ใช้ accelerometer และ gyroscope sensor ในการหามุมที่หมุนไปของแขนกลของหุ่นยนต์ โดยแบ่งออกเป็น 4 วิธี วิธีที่หนึ่งใช้ accelerometer จำนวน  $2n-1$  ต่อจำนวน  $n$ -joint-angle วิธีที่สองและวิธีที่สามใช้ accelerometer จำนวน  $n$  ตัว และ gyroscope sensor จำนวน  $n$  ตัว ซึ่งทั้งสองวิธีแตกต่างกันที่การรับค่าจาก gyroscope sensor โดยวิธีที่สองจะ integrate ค่าที่วัดจากเซนเซอร์ ส่วนวิธีที่สามจะ differentiation ค่าที่วัดจากเซนเซอร์ วิธีที่สี่ใช้ accelerometers จำนวน  $2n$  ตัว จากผลการทดลองแสดงให้เห็นว่าวิธีที่สามและวิธีสี่มีความแม่นยำในการวัดมุมเอียงของแขนกลดีกว่าสองวิธีแรก กล่าวโดยสรุปแล้วการติดตั้งเซนเซอร์ใกล้กับจุดหมุนของแขนกลจะช่วยทำให้มีความแม่นยำในการวัดมุมเอียงได้มากยิ่งขึ้น และในแต่ละวิธีมีข้อดีข้อเสียแตกต่างกันไปตามการใช้งาน ซึ่งในงานวิจัยนี้ใช้เซนเซอร์ accelerometer 1 ตัว และ gyroscope sensor 1 ตัว โดยติดตั้งเซนเซอร์ให้อยู่ใกล้กับจุดหมุน (pivot point) ของหุ่นยนต์สองล้อมากที่สุด

การวัดมุมเอียงของหุ่นยนต์สองล้อถือเป็นปัจจัยหลักที่สำคัญในการรักษาสมดุลหุ่นยนต์สองล้อของงานวิจัยนี้ Wasif *et al.* (2013) ใช้เซนเซอร์ร่วมกันระหว่าง accelerometer และ

gyroscope sensor ในการวัดมุมเอียงและใช้หลักการที่เรียกว่า complementary filter ภาพที่ 5 ซึ่งสามารถใช้งานและเข้าใจได้ง่ายกว่าหลักการของ Kalman filter ที่มีความซับซ้อนสูง อีกทั้งยังช่วยลดภาระการคำนวณอย่างมากให้กับไมโครคอนโทรลเลอร์ เซนเซอร์ทั้งสองชนิดสามารถวัดมุมเอียงได้แต่จะไม่มีความแม่นยำเมื่อไม่ได้ใช้ร่วมกัน ได้ให้ความสำคัญของค่า filter constant ในการออกแบบด้วยหลักการ complementary filter ซึ่งได้ทำการทดลองให้ค่าของ filter constant ที่ 0.75 ถึง 0.90 จะเห็นถึงความแตกต่างของความแม่นยำและอัตราหน่วงในการวัดมุม ดังนั้นการเลือกค่าของ filter constant ให้เหมาะสมกับค่า sampling time ของไมโครคอนโทรลเลอร์ก็จะช่วยเพิ่มประสิทธิภาพในการวัดมุมเอียงมากยิ่งขึ้น

## อุปกรณ์และวิธีการ

### อุปกรณ์

1. หุ่นยนต์สองล้อ เป็นหุ่นยนต์ขนาด 15.5 x 28 x 28.9 เซนติเมตร น้ำหนักรวม 5.08 กิโลกรัม วัสดุโครงทำจากอลูมิเนียม ส่วนประกอบที่สำคัญภายในหุ่นยนต์สองล้อประกอบด้วย บอร์ดไมโครคอนโทรลเลอร์ (Arduino ATmega 2560) , วงจรขับมอเตอร์กระแสตรง (Motor Driver) , เซนเซอร์สองชนิด, มอเตอร์กระแสตรง 2 ชุด โดยต่อตรงกับล้อทั้งสองที่ทำงานอย่างอิสระต่อกัน และแบตเตอรี่ 12V



ภาพที่ 6 หุ่นยนต์สองล้อและส่วนประกอบต่างๆ

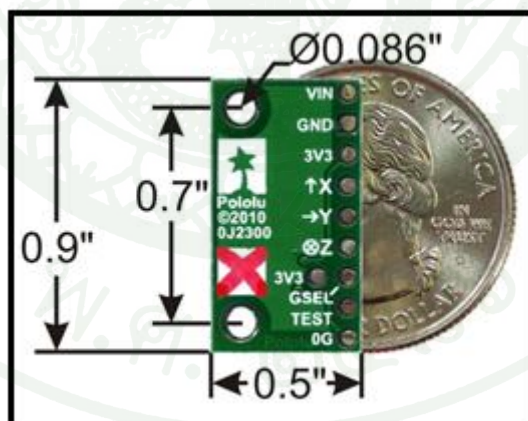
2. เซนเซอร์ ประกอบด้วย

2.1 เอนโคเดอร์เซนเซอร์ (Encoder) อุปกรณ์วัดตำแหน่งที่ทำงานโดยอาศัยการยิงลำแสงผ่านแถบมืด-สว่างและส่งออกเป็นสัญญาณดิจิทัล การอ่านค่าอัตราการเปลี่ยนแปลงของระดับสัญญาณถูกนำมาคำนวณเป็นอัตราเร็วของการเปลี่ยนแปลงตำแหน่ง งานวิจัยนี้ใช้เอนโคเดอร์ประเภท Rotary Encoder HKT3006-301C-500B มีความละเอียด 500 ครั้งต่อรอบ (CPR) โดยมี 2 ชุด ติดอยู่กับมอเตอร์และใช้สำหรับการวัดตำแหน่งที่หมุนไปของล้อทั้งสองข้าง



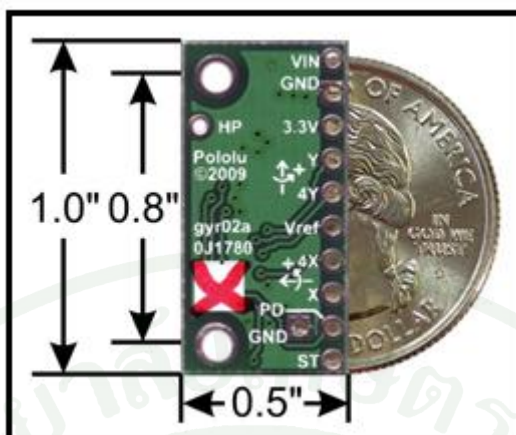
ภาพที่ 7 อุปกรณ์เอนโคเดอร์เซ็นเซอร์

2.2 เซนเซอร์วัดความเร่ง (Accelerometer) คือเซนเซอร์วัดความเร่งประเภท MMA7341LC 3-Axis  $\pm 3g$  (440 mV/g) ในงานวิจัยนี้ใช้สำหรับวัดมุมเอียงของหุ่นยนต์สองล้อในแกน Pitch ซึ่งใช้หลักแรงโน้มถ่วงของโลก โดยทำการแปลงสัญญาณ analog ที่อ่านได้จาก accelerometer ให้อยู่ในรูปของแรง “g” (หรือความเร่ง) น้ำหนัก 0.7 กรัม รายละเอียดเพิ่มเติมที่ [www.pololu.com](http://www.pololu.com)



ภาพที่ 8 อุปกรณ์เซนเซอร์วัดความเร่ง

2.3 เซนเซอร์วัดความเร็วเชิงมุม เรียกว่า Rate Gyroscope เป็นอุปกรณ์วัดความเร็วเชิงมุมที่มีการตอบสนองที่ดีในช่วงความถี่สูง โดยใช้ประเภท LPR550AL Dual-Axis  $\pm 500^\circ/s$  ในแกน Pitch และ Roll หรือ XY น้ำหนัก 0.7 กรัม รายละเอียดเพิ่มเติมที่ [www.pololu.com](http://www.pololu.com)



ภาพที่ 9 อุปกรณ์เซนเซอร์วัดความเร็วเชิงมุม Rate Gyroscope

3. วงจรขับมอเตอร์กระแสตรง (Motor Drive Module) ทำหน้าที่ขยายกระแสไฟฟ้า โดยอุปกรณ์นี้จะรับสัญญาณป้อนเข้าคือกระแสไฟฟ้าที่มีปริมาณและแรงดันต่ำ และจ่ายกระแสไฟฟ้าที่มีปริมาณและแรงดันสูงกว่า ซึ่งงานวิจัยนี้ใช้อุปกรณ์รุ่น L298N Dual Full-bridge motor drive module สามารถควบคุมมอเตอร์ได้ 2 ตัวแบบอิสระ รายละเอียดเพิ่มเติม [www.micontechlab.com](http://www.micontechlab.com)



ภาพที่ 10 อุปกรณ์วงจรขับมอเตอร์กระแสตรง

4. มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) เป็นมอเตอร์ชนิด ZYTD-38SRZ-R รับแรงดันได้ระหว่าง 12-24 V rpm : 250 เป็น DC Motor ที่ใส่เฟืองทดรอบเข้าไปเพื่อให้ความเร็วรอบลดลงตามการนำไปใช้งาน



ภาพที่ 11 อุปกรณ์มอเตอร์ไฟฟ้ากระแสตรง

5. ไมโครคอนโทรลเลอร์ประเภทอาดูยอีโน้ (Arduino Mega 2560 R3) คือตัวควบคุมของระบบ รับส่งข้อมูลที่ความเร็ว 16 MHZ ขนาดหน่วยความจำ 256 KB เป็นไมโครคอนโทรลเลอร์ที่มีราคาถูกเนื่องจากระบบเป็นแบบ Open Source ไม่มีลิขสิทธิ์ในการนำไปใช้งานเชิงพาณิชย์ และซอฟต์แวร์ Arduino IDE ที่ใช้ในการพัฒนาสามารถดาวน์โหลดได้ฟรี ซึ่งงานวิจัยนี้ใช้ไมโครคอนโทรลเลอร์ร่วมกับโปรแกรม Matlab 2013a ผ่านทาง Arduino Support from Simulink รายละเอียดเพิ่มเติม [www.arduino.cc/](http://www.arduino.cc/)



ภาพที่ 12 อุปกรณ์ไมโครคอนโทรลเลอร์ (Arduino Mega 2560 R3)

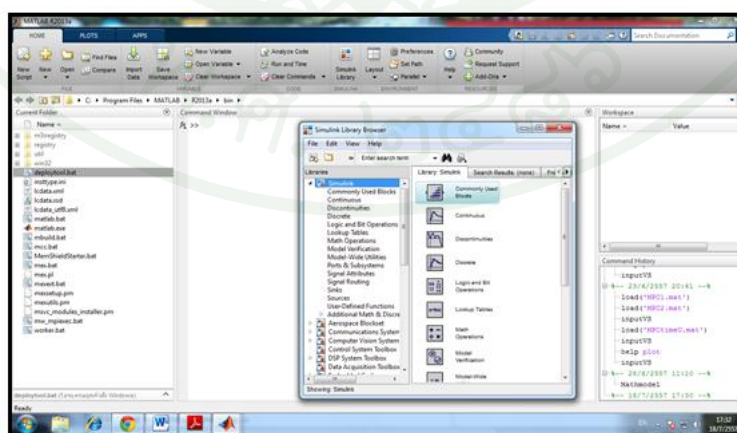
6. แบตเตอรี่ 12V 3000 mAh ใช้สำหรับจ่ายไฟให้แก่ ไมโครคอนโทรลเลอร์, มอเตอร์ กระแสตรงและวงจรขับมอเตอร์กระแสตรง (Power VCC)



ภาพที่ 13 อุปกรณ์แบตเตอรี่ 12 V

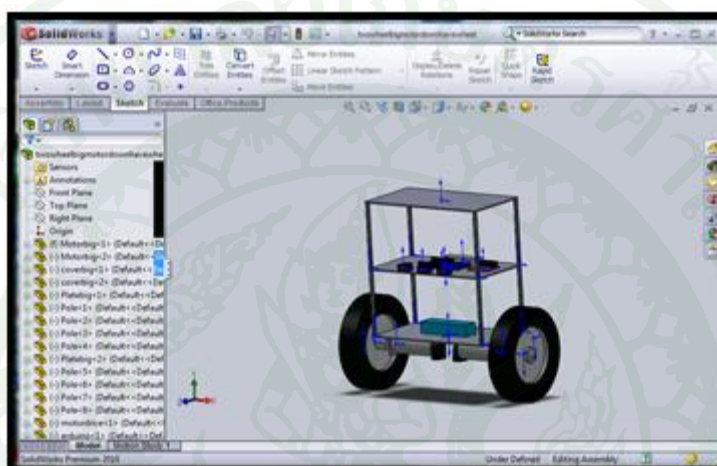
7. เครื่องคอมพิวเตอร์ ใช้ในการเขียนโค้ดซอฟต์แวร์ Simulink สำหรับ flash ลงไปยังบอร์ด ไมโครคอนโทรลเลอร์ และการใช้โปรแกรม Solidworks

8. โปรแกรม MATLAB เป็นโปรแกรมที่สามารถคำนวณสมการคณิตศาสตร์ที่ซับซ้อนได้ รวดเร็วและแม่นยำ มีฟังก์ชันสำเร็จรูปในแต่ละสาขาวิชาและฟังก์ชันพื้นฐานจำนวนมาก อีกทั้งยังสามารถแสดงผลเชิงกราฟิก ซึ่งเป็นโปรแกรมที่ใช้งานง่ายและทำงานได้หลากหลายรูปแบบ และใช้ในส่วนสำคัญของงานวิจัยนี้คือการจำลองผลการทดลองด้วยซอฟต์แวร์ Simulink



ภาพที่ 14 ตัวอย่างการใช้งานโปรแกรม MATLAB

9. โปรแกรม Solid works เป็น โปรแกรมที่ออกแบบชิ้นงานในระบบ 3 มิติ จะมีเครื่องมือช่วยในการสร้างชิ้นส่วนต่างๆ โดยการนำชิ้นส่วนมาประกอบกันก็จะทำให้สามารถจำลองชิ้นส่วนได้เหมือนจริง และสามารถสร้างงาน Drawing ได้โดยอัตโนมัติ ซึ่งในงานวิจัยนี้ใช้สำหรับจำลองหุ่นยนต์สองล้อ เพื่อใช้สำหรับหาค่าพารามิเตอร์ต่างๆ ได้โดยง่าย

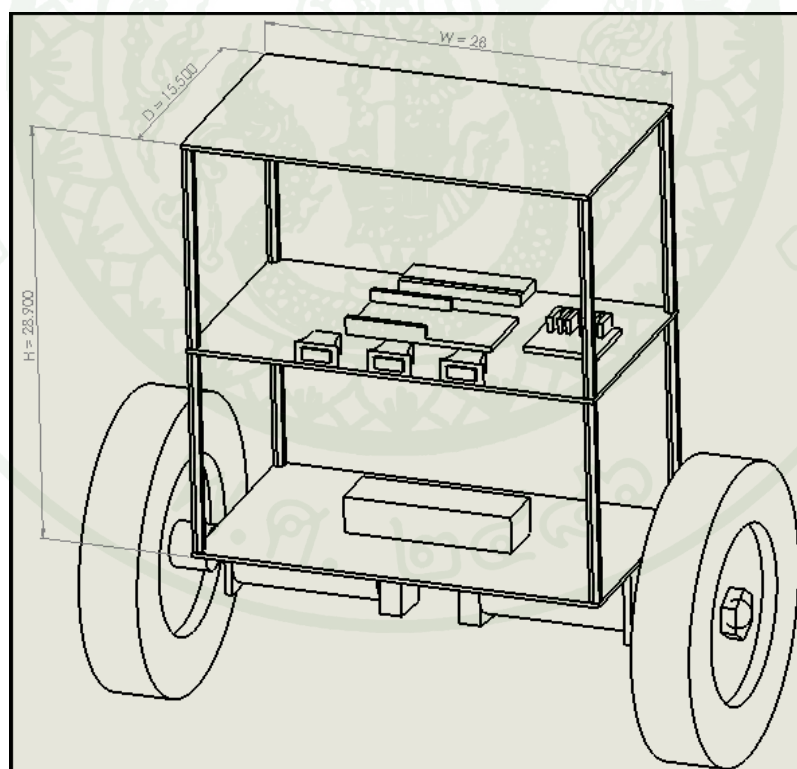


ภาพที่ 15 ตัวอย่างการใช้งานโปรแกรม Solidworks

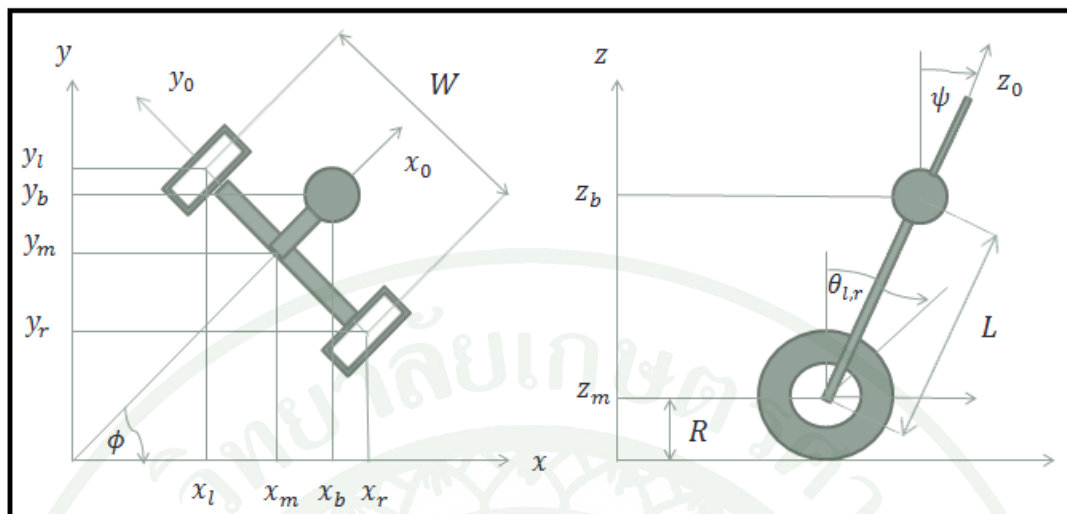
## วิธีการ

### การหาแบบจำลองทางคณิตศาสตร์

การจำลองระบบด้วยสมการคณิตศาสตร์ (Mathematical Model) มีความสำคัญต่อการออกแบบตัวควบคุม ต้องมีความแม่นยำใกล้เคียงกับระบบจริง ในบางระบบนั้นการหาสมการการเคลื่อนที่เชิงกลทำได้ค่อนข้างยากอาจต้องใช้วิธี system identification เพื่อให้ได้แบบจำลองทางคณิตศาสตร์มา ซึ่งการหาแบบจำลองที่มีประสิทธิภาพมีหลายวิธีหาได้จากสมการการเคลื่อนที่เชิงกล ยกตัวอย่างเช่น กฎข้อที่สองของนิวตัน (Grasser et al., 2002) ทฤษฎีของเคน (Kim et al., 2005) และงานวิจัยนี้นำสมการลากรางจ์มาใช้ในการหาแบบจำลองทางคณิตศาสตร์ซึ่งมีความเหมาะสมกับสำหรับระบบที่มีความยุ่งยากและซับซ้อน โดยจะคำนวณจากสมการพลังงานศักย์และพลังงานจลน์ในรูปของตัวแปรพิกัดหุ่นยนต์สองล้ออยู่กำจัดให้อยู่สามระบบพิกัดคือ  $(\theta, \psi, \phi)$



ภาพที่ 16 โครงสร้างของหุ่นยนต์สองล้อจากโปรแกรม Solidworks



ภาพที่ 17 หุ่นยนต์สองข้อบนแกนพิกัด x,y,z

ตารางที่ 1 ค่าพารามิเตอร์ทั่วไปของหุ่นยนต์

พารามิเตอร์	ความหมาย	ขนาด (หน่วย)
$g$	แรงโน้มถ่วงโลก	$9.81 (m/sec^2)$
$m$	มวลล้อ	$0.990 (kg)$
$R$	รัศมีล้อ	$0.10 (m)$
$J_w = mR^2 / 2$	โมเมนต์ความเฉื่อยของล้อ	$0.005 (kgm^2)$
$M$	มวลตัวถัง	$3.1 (kg)$
$W$	ระยะความกว้างฐานล้อ	$0.36 (m)$
$D$	ระยะความลึกของตัวถัง	$0.155 (m)$
$H$	ระยะความสูงของตัวถัง	$0.289 (m)$
$L$	ระยะจากแกนล้อถึงศูนย์ถ่วง	$0.1359 (m)$
$J_{by} = ML^2 / 3$	โมเมนต์ความเฉื่อยตัวถังแกน $y_0$	$0.001 (kgm^2)$
$J_{bz}$	โมเมนต์ความเฉื่อยตัวถังแกน $z_0$	$0.0039 (kgm^2)$
$J_m$	โมเมนต์ความเฉื่อยของมอเตอร์	$1 \times 10^{-5} (Nm/A)$
$R_m$	ค่าความต้านทานของมอเตอร์	$6.69 (\Omega)$

ตารางที่ 1 (ต่อ) ค่าพารามิเตอร์ทั่วไปของหุ่นยนต์

$K_b$	ค่าคงที่แรงดันไฟฟ้าย้อนกลับมอเตอร์	0.468 (V sec/ rad)
$K_t$	ค่าคงที่แรงบิดมอเตอร์	0.317 (Nm/ A)
$n$	อัตราทดมอเตอร์	1
$f_m$	ค่าสัมประสิทธิ์แรงเสียดทานระหว่างตัวถังและมอเตอร์	0.0022
$f_w$	ค่าสัมประสิทธิ์แรงเสียดทานระหว่างล้อและพื้น	0

ค่าพารามิเตอร์ที่หาได้ยาก ( $R_m, K_b, K_t, J_m, n, f_m, f_w$ ) จึงเลือกใช้ค่าที่ใกล้เคียงกับงานวิจัยอื่น (Yomamoto, 2008) กำหนดค่าได้ดังตารางที่ 1

สมการลากรางจ์เกิดจากผลต่างระหว่างพลังงานจลน์และพลังงานศักย์ภายในระบบรูปแบบดังนี้

$$L_g = T - V \quad \text{และ} \quad \frac{d}{dt} \left( \frac{\partial L_g}{\partial \dot{q}_i} \right) - \frac{\partial L_g}{\partial q_i} = Q_i \quad (1)$$

โดยที่  $T$  = พลังงานจลน์รวมของระบบ

$V$  = พลังงานศักย์รวมของระบบ

$q_i$  = ระบบพิกัด

$Q_i$  = แรงในระบบพิกัดที่กำลังสนใจ

$i$  = อันดับของระบบ

กำหนดให้

$\theta$  = มุมเอียงของล้อซ้ายและล้อขวา

$\psi$  = มุมเอียงของหุ่นยนต์สองล้อ

$\phi$  = มุมเอียงของหุ่นยนต์สองล้อ

ที่เวลาเริ่มต้น  $t = 0$  กำหนดให้ค่าพารามิเตอร์ของหุ่นยนต์สองล้อให้อยู่บนระบบพิกัดตั้งแสดงในภาพที่ 17 ดังนั้นสามารถเขียนค่าพารามิเตอร์ได้ดังนี้

$$(\theta, \phi) = \left( \frac{1}{2}(\theta_l + \theta_r) \quad \frac{R}{W}(\theta_r - \theta_l) \right) \quad (2)$$

$$(x_m, y_m, z_m) = (R\theta \cos \phi, R\theta \sin \phi, R) \quad (3)$$

$$(x_l, y_l, z_l) = \left( x_m - \frac{W}{2} \sin \phi, y_m + \frac{W}{2} \cos \phi, z_m \right) \quad (4)$$

$$(x_r, y_r, z_r) = \left( x_m + \frac{W}{2} \sin \phi, y_m - \frac{W}{2} \cos \phi, z_m \right) \quad (5)$$

$$(x_b, y_b, z_b) = (x_m + L \sin \psi \cos \phi, y_m + L \sin \psi \sin \phi, z_m + L \cos \psi) \quad (6)$$

พลังงานจลน์จากการเคลื่อนที่เชิงเส้น ( $T_1$ ), พลังงานจลน์จากการเคลื่อนที่แบบหมุน ( $T_2$ ) และพลังงานศักย์ ( $V$ ) หาได้จากสมการ (7), (8) และ (9)

$$T_1 = \frac{1}{2} m_w (\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2} m_w (\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \frac{1}{2} m_b (\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \quad (7)$$

$$T_2 = \frac{1}{2} J_w \dot{\theta}_l^2 + \frac{1}{2} J_w \dot{\theta}_r^2 + \frac{1}{2} J_{by} \dot{\psi}^2 + \frac{1}{2} J_{bz} \dot{\phi}^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_r - \dot{\psi})^2 \quad (8)$$

$$V = mgz_l + mgz_r + mgz_b \quad (9)$$

ดังนั้นค่าลากรางจ์ ( $L$ ) คือ

$$L_g = T_1 + T_2 - V \quad (10)$$

สมการลากรางจ์สามารถหาได้จากสมการ

$$\frac{d}{dt} \left( \frac{\partial L_g}{\partial \dot{\theta}} \right) - \frac{\partial L_g}{\partial \theta} = F_\theta \quad (11)$$

$$\frac{d}{dt} \left( \frac{\partial L_g}{\partial \dot{\psi}} \right) - \frac{\partial L_g}{\partial \psi} = F_\psi \quad (12)$$

$$\frac{d}{dt} \left( \frac{\partial L_g}{\partial \dot{\phi}} \right) - \frac{\partial L_g}{\partial \phi} = F_\phi \quad (13)$$

จากสมการลากรางจ์ (11), (12) และ (13) แทนค่าจากสมการที่ (10) จัดรูปใหม่จะได้ดังนี้

$$\begin{aligned} & \left[ (2m + M)R^2 + 2J_w + 2n^2 J_m \right] \ddot{\theta} + (MLR \cos \psi - 2n^2 J_m) \ddot{\psi} \\ & - MLR \dot{\psi}^2 \sin \psi - \left[ (2m + M)R^2 \theta + MLR \sin \psi \right] \dot{\phi}^2 = F_\theta \end{aligned} \quad (14)$$

$$\begin{aligned} & (MLR \cos \psi - 2n^2 J_m) \ddot{\theta} + (ML^2 + J_{bz} + 2n^2 J_m) \ddot{\psi} - MgL \sin \psi \\ & - (MLR \theta + ML^2 \sin \psi) \dot{\phi}^2 \cos \psi = F_\psi \end{aligned} \quad (15)$$

$$\begin{aligned} & \left[ \frac{1}{2} mW^2 + J_{bz} + \frac{W^2}{2R^2} (J_w + n^2 J_m) + (2m + M)R^2 \theta^2 + 2MLR \theta \sin \psi \right] \ddot{\phi} \\ & + 2 \left[ (2m + M)R^2 \theta \dot{\theta} + ML^2 \dot{\psi} \sin \psi \cos \psi + MLR (\dot{\theta} \sin \psi + \theta \dot{\psi} \cos \psi) \right] \dot{\phi} = F_\phi \end{aligned} \quad (16)$$

พิจารณาในส่วนของค่าแรงบิดมอเตอร์และความฝืดหนืดสามารถเขียนอยู่ในรูปของแรงได้ดังนี้

$$(F_\theta, F_\psi, F_\phi) = \left( \frac{1}{2} (F_l + F_r), F_\psi, \frac{R}{W} (F_r - F_l) \right) \quad (17)$$

$$F_l = nK_l i_l + f_m (\dot{\psi} - \dot{\theta}_l) - f_w \dot{\theta}_l \quad (18)$$

$$F_r = nK_r i_r + f_m (\dot{\psi} - \dot{\theta}_r) - f_w \dot{\theta}_r \quad (19)$$

$$F_\psi = -nK_l i_l - nK_r i_r + f_m (\dot{\psi} - \dot{\theta}_l) - f_m (\dot{\psi} - \dot{\theta}_r) \quad (20)$$

โดย  $i_{l,r}$  คือกระแสของมอเตอร์

เนื่องจากไม่สามารถใช้ค่ากระแสของมอเตอร์ได้โดยตรงจึงต้องกำหนดความสัมพันธ์ระหว่าง  $i_{l,r}$  และ  $v_{l,r}$  ดังนี้

$$L_m i_{l,r} = v_{l,r} + K_b (\dot{\psi} - \dot{\theta}_{l,r}) - R_m i_{l,r} \quad (21)$$

สามารถประมาณค่าของตัวเหนี่ยวนำ ( $L_m$ ) = 0 ดังนั้นจะได้

$$i_{l,r} = \frac{v_{l,r} + K_b (\dot{\psi} - \dot{\theta}_{l,r})}{R_m} \quad (22)$$

จากสมการที่ (22) สามารถเขียนอยู่ในรูปของแรงได้ดังนี้

$$F_\theta = \frac{\alpha}{2} (v_l + v_r) - (\beta + f_w) \dot{\theta} + \beta \dot{\psi} \quad (23)$$

$$F_\psi = -\alpha (v_l + v_r) + 2\beta \dot{\theta} - 2\beta \dot{\psi} \quad (24)$$

$$F_\phi = \frac{R}{W} \alpha (v_r - v_l) - \left( \beta + \frac{W}{R} f_w \right) \dot{\phi} \quad (25)$$

$$\alpha = \frac{nK_t}{R_m}, \beta = \frac{nK_t K_b}{R_m} + f_m \quad (26)$$

ในสมการ (14), (15) และ (16) นั้นเป็นระบบไม่เชิงเส้น (nonlinear system) ตัวควบคุมแบบกำลังสองเชิงและโมเดลพรีดิกทีฟของงานวิจัยนี้ยังคงอยู่ภายใต้ทฤษฎีเชิงเส้น ดังนั้นจึงต้องประมาณระบบดังกล่าวเป็นเชิงเส้นโดยการกำหนดให้ค่า  $\psi$  มีค่าน้อยๆ จึงสามารถประมาณค่าเทอมที่ไม่เป็นเชิงเส้นเช่น  $\sin \psi = \psi$ ,  $\cos \psi = 1$  และ  $\psi^2 = 0$  จะได้สามสมการพลศาสตร์เชิงเส้นสำหรับการออกแบบระบบควบคุมเชิงเส้นคือสมการ

$$\left[ (2m + M)R^2 + 2J_w + 2n^2 J_m \right] \ddot{\theta} + (MLR - 2n^2 J_m) \dot{\psi} = F_\theta \quad (27)$$

$$(MLR - 2n^2 J_m) \ddot{\theta} + (ML^2 + J_{by} + 2n^2 J_m) \dot{\psi} - MgL\psi = F_\psi \quad (28)$$

$$\left[ \frac{1}{2} mW^2 + J_{bz} + \frac{W^2}{2R^2} (J_w + n^2 J_m) \right] \ddot{\phi} = F_\phi \quad (29)$$

จากสมการ (27), (28) และ (29) เขียนให้อยู่ในรูปอย่างง่ายได้ดังนี้

$$E \begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + F \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} + G \begin{bmatrix} \theta \\ \psi \end{bmatrix} = H \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (30)$$

$$\text{โดย } E = \begin{bmatrix} (2m+M)R^2 + 2J_w + 2n^2J_m & MLR - 2n^2J_m \\ MIR - 2n^2J_m & ML^2 + J_{by} + 2n^2J_m \end{bmatrix}$$

$$F = \begin{bmatrix} \beta + f_w & -\beta \\ -2\beta & 2\beta \end{bmatrix}$$

$$G = \begin{bmatrix} 0 & 0 \\ 0 & -MgL \end{bmatrix}$$

$$H = \begin{bmatrix} \frac{\alpha}{2} & \frac{\alpha}{2} \\ -\alpha & -\alpha \end{bmatrix}$$

$$\text{และ } K\ddot{\phi} + I\dot{\phi} = J(v_r - v_l) \quad (31)$$

$$\text{โดย } I = \beta + \frac{W}{R} f_w$$

$$J = \frac{R}{W} \alpha$$

$$K = \frac{1}{2} mW^2 + J_{bz} + \frac{W^2}{2R^2} (J_w + n^2J_m)$$

สามารถเขียนให้อยู่ในรูปปริภูมิสเตรตได้คือ

$$\dot{x} = Ax + Bu = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} x + \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} u \quad (32)$$

$$\text{เมื่อ } x = [\theta, \psi, \dot{\theta}, \dot{\psi}, \phi, \dot{\phi}]^T \text{ และ } u = [v_l, v_r]^T$$

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & A_1(3,2) & A_1(3,3) & A_1(3,4) \\ 0 & A_1(4,2) & A_1(4,3) & A_1(4,4) \end{bmatrix}$$

$$A_1(3,2) = -MgLE(1,2) / \det(E)$$

$$A_1(4,2) = MgLE(1,1) / \det(E)$$

$$A_1(3,3) = -[(\beta + f_w)E(2,2) + 2\beta E(1,2)] / \det(E)$$

$$A_1(4,3) = [(\beta + f_w)E(1,2) + 2\beta E(1,1)] / \det(E)$$

$$A_1(3,4) = \beta(E(2,2) + 2E(1,2)) / \det(E)$$

$$A_1(4,4) = -\beta(E(1,2) + 2E(1,1)) / \det(E)$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ B_1(3) & B_1(3) \\ B_1(4) & B_1(4) \end{bmatrix}$$

$$B_1(3) = \alpha(E(2,2) / 2 + E(1,2)) / \det(E)$$

$$B_1(4) = -\alpha(E(1,2) / 2 + E(1,1)) / \det(E)$$

$$\det(E) = E(1,1)E(2,2) - E(1,2)^2$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{I}{K} \end{bmatrix}$$

$$B_2 = \begin{bmatrix} 0 & 0 \\ -\frac{J}{K} & -\frac{J}{K} \end{bmatrix}$$

โดยที่เอาท์พุทของระบบคือ

$$y = Cx$$

(33)

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

## การออกแบบระบบควบคุม

ระบบควบคุมที่ใช้ในงานวิจัยนี้ประกอบด้วย 4 ส่วนคือ คุณสมบัติความควบคุมได้และสังเกตได้, การออกแบบการวัดมุมเอียงด้วยหลักการ Complementary filter, การออกแบบตัวคุมค่ากำลังสองเชิงเส้น, การออกแบบตัวควบคุมโมเดลพรีดิกทีฟ (กิตติพงษ์, 2551; วโรดม, 2551; Colton, 2007; Ogata, 1997; Maciejowski, 2002; Wang, 2009.)

### 1. คุณสมบัติความควบคุมได้และสังเกตได้ (Controllability and Observability)

สำหรับการออกแบบระบบควบคุมในงานวิจัยนี้จะสอดคล้องและเป็นไปตามทฤษฎีเมื่อระบบที่ต้องการควบคุมนั้นมีคุณสมบัติความสามารถควบคุมได้และสังเกตได้

#### 1.1 คุณสมบัติความควบคุมได้ (Controllability)

นิยาม ความควบคุมได้ของสเตต

ระบบพลวัต  $\dot{x} = Ax + Bu$  จะควบคุมสเตตได้ ถ้าหากเลือกสเตตเริ่มต้น  $x(0) = x_0$  ใดๆและสเตตสุดท้าย  $x(t_1) = x_1$  โดย  $t_1 > 0$  แล้วสามารถหาอินพุต  $u(t)$  ที่ขับสเตตจาก  $x_0$  ไปหา  $x_1$

(A,B) จะสามารถควบคุมสเตตได้ ถ้าและเพียงแต่ถ้า เมทริกซ์ความควบคุมได้

$$C = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B] \quad (34)$$

มีค่าลำดับชั้น (rank) เท่ากับ  $n$  โดยที่  $n$  คือจำนวนของสเตต

ซึ่งสามารถพิจารณาได้ด้วยโปรแกรม MATLAB ด้วยคำสั่ง `rank(ctrb(A,B))`

#### 1.2 คุณสมบัติความสังเกตได้ (Observability)

นิยาม ความสังเกตได้ของสเตต

ระบบพลวัต  $\dot{x} = Ax + Bu, y = Cx + Du$  จะสังเกตสเตตได้ ถ้าหาก  $t_1 > 0$  ใดๆสามารถรู้ค่าสเตตเริ่มต้น  $x(0) = x_0$  ได้จากค่าของอินพุต  $u(t)$  และเอาต์พุต  $y(t)$  ในช่วงเวลา  $[0, t_1]$

(A,C) สามารถสังเกตสเตตได้ถ้าและเพียงแต่ถ้า เมทริกซ์ความสังเกตได้ถ้า

$$O = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad (35)$$

มีค่าลำดับชั้น (rank) เท่ากับ  $n$  โดยที่  $n$  คือจำนวนของสเตต  
ซึ่งสามารถพิจารณาได้ด้วยโปรแกรม MATLAB ด้วยคำสั่ง `rank(observ(A,C))`

## 2. ออกแบบการวัดมุมเอียงด้วยหลักการ Complementary filter

การออกแบบการวัดมุมเอียงโดยใช้การทำงานร่วมกันระหว่างเซนเซอร์วัดความเร่ง (Accelerometer) และเซนเซอร์วัดความเร็วเชิงมุม (Rate Gyroscope) โดยส่วนใหญ่แล้วคนทั่วไปมักจะคุ้นเคยกับหลักการของ Kalman filter ซึ่งมีความซับซ้อนและเข้าใจได้ยากกว่าหลักการ complementary filter

### 2.1 การอ่านค่าจากเซนเซอร์

$X_{acc} = (\text{ชนิดของข้อมูล}) * (\text{ค่าที่อ่านได้จากเซนเซอร์วัดความเร่ง} - \text{ระยะห่าง}) * \text{อัตราขยาย}$

$Gyro = (\text{ชนิดของข้อมูล}) * (\text{ค่าที่อ่านได้จากเซนเซอร์วัดความเร็วเชิงมุม} - \text{ระยะห่าง}) * \text{อัตราขยาย}$

-ชนิดของข้อมูล (float) คือการแปลงค่า ADC (analog to digital converter) ที่อ่านจากบอร์ด Arduino ให้อยู่ในหน่วยโวลต์ (voltage)

$$Volts = ADC * V_{ref} / 1023$$

โดยที่  $V_{ref} = 5V$  และ 10-bit ADC =  $2^{10} - 1 = 1023$

-ระยะห่าง (offset) สำหรับเซนเซอร์วัดความเร่งหาระยะห่างได้โดยการวางเซนเซอร์ให้อยู่ในแนวราบ (horizontal) แล้วอ่านค่าเฉลี่ยที่วัดได้ หรือสามารถหาค่าได้จาก

datasheet ของเซนเซอร์วัดความเร่งโดยใช้ค่า Zero g ในที่นี้เท่ากับ 1.65 V ทำการแปลงเป็นค่า ADC ได้เท่ากับ 337.59 และสำหรับเซนเซอร์วัดความเร็วเชิงมุมหาระยะห่างได้โดยการวางเซนเซอร์ให้ไม่มีการขยับเขยื้อนแล้วจึงอ่านค่าเฉลี่ยที่วัดได้

-อัตราขยาย (gain) หาได้จาก datasheet หรือจากการทดลอง ในที่นี้คือ เซนเซอร์วัดความเร่ง =  $440mV / g$  และ เซนเซอร์วัดความเร็วเชิงมุม =  $2mV / ^\circ / s$

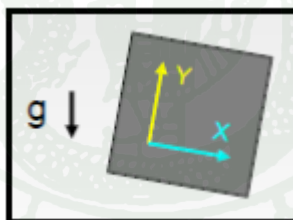
สรุปในงานวิจัยนี้อ่านค่าจากเซนเซอร์ทั้งสองชนิดดังนี้

$$X\_acc = (5 / 1023) * (X\_accADC - offset) * (1 / 0.44)$$

$$Gyro = (5 / 1023) * (GyroADC - offset) * (1 / 0.002)$$

## 2.2 การวัดมุมจากเซนเซอร์

-การวัดมุมของเซนเซอร์วัดความเร่ง สามารถหาได้จากแกน  $x$  เพียงแกนเดียว เนื่องจากในแกน  $y$  มีการเปลี่ยนแปลงน้อยมาก ซึ่งมีข้อดีคือช่วยลดเวลาการประมวลผลและความไม่เป็นเชิงเส้นได้ดีกว่าใช้ทั้งสองแกน ( $x, y$ )



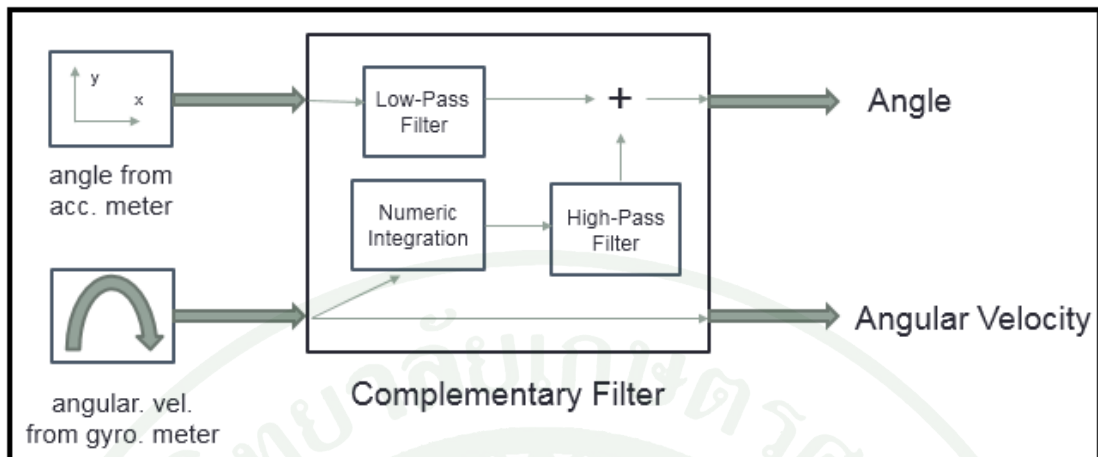
ภาพที่ 18 การวัดมุมของเซนเซอร์วัดความเร่ง

$$X - axis \text{ read} : (1g) \times \sin(\theta)$$

$$Angle = (X\_acc * \arcsin \theta) * 180 / \pi \quad (36)$$

-การวัดมุมของเซนเซอร์วัดความเร็วเชิงมุมด้วย Integration

$$Angle = gyro * dt \quad (37)$$



ภาพที่ 19 ตัวกรองสัญญาณแบบดิจิทัล (Digital Filter)

### 2.3 ตัวกรองสัญญาณแบบดิจิทัล (Digital Filter)

-ตัวกรองความถี่ต่ำกว่าผ่านได้ (Low-Pass Filter) สำหรับเซนเซอร์มีความแม่นยำในการวัดมุมเพียงช่วงความถี่ต่ำ ถ้ามีแรงกระทำภายนอกเพียงเล็กน้อยจะเกิดสัญญาณรบกวนในการวัดมุมเพียงของเซนเซอร์ได้

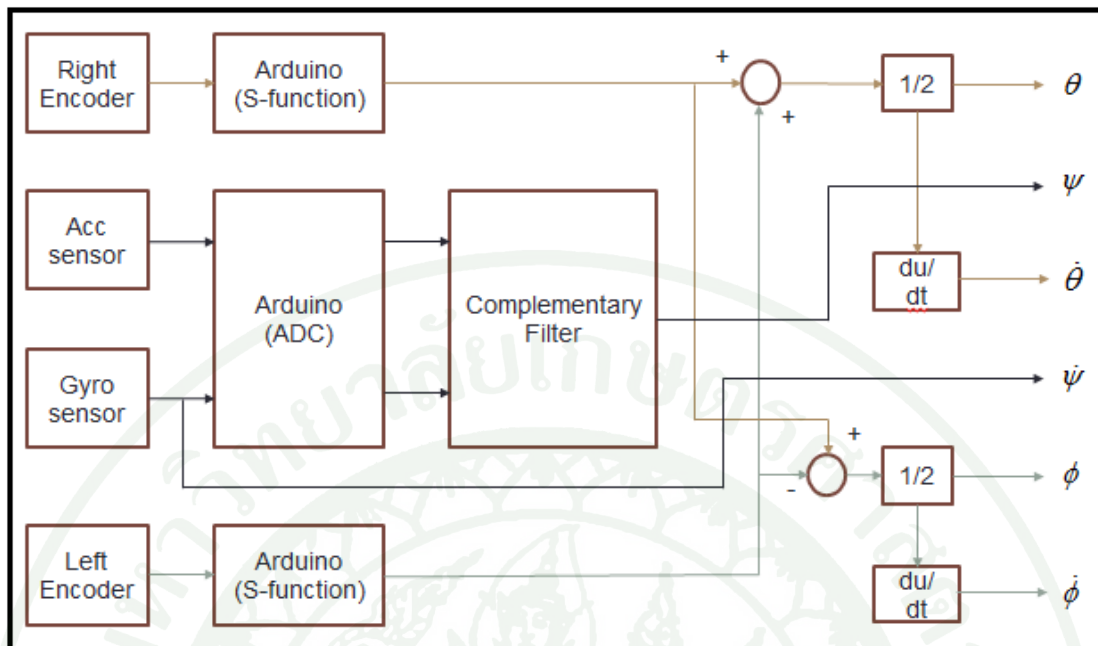
-ตัวกรองความถี่สูงกว่าผ่านได้ (High-Pass Filter) มีจุดเด่นคือสามารถตัดปัญหาการ drift ได้ดี ใช้สำหรับเซนเซอร์วัดความเร็วเชิงมุมและมีความแม่นยำในการวัดมุมเพียงช่วงความถี่สูง นำตัวกรองความถี่ทั้งสองชนิดมาใช้ร่วมกันเขียนเป็นสมการได้ดังนี้

$$angle = a * (angle + gyro * dt) + (1 - a) * (x\_acc) \quad (38)$$

โดยที่

$$a = \frac{\tau}{\tau + dt}$$

ในกรณีที่ค่าของเซนเซอร์วัดความเร็วเชิงมุมมีค่า drift ประมาณ 2 องศาต่อวินาที (กรณีที่ drift ค่อนข้างมาก) ควรลด ค่า  $\tau$  (time constant) ให้น้อยกว่า 1 เพื่อให้แน่ใจว่ามุมที่วัดได้จะไม่ drift เกิน 2 องศา แต่การลดค่า  $\tau$  ก็จะทำให้สัญญาณรบกวน (noise) จากเซนเซอร์วัดความเร่งผ่านเข้ามา ดังนั้นการเลือกค่า  $\tau$  จึงมีความสำคัญต่อตัวกรองความถี่มาก



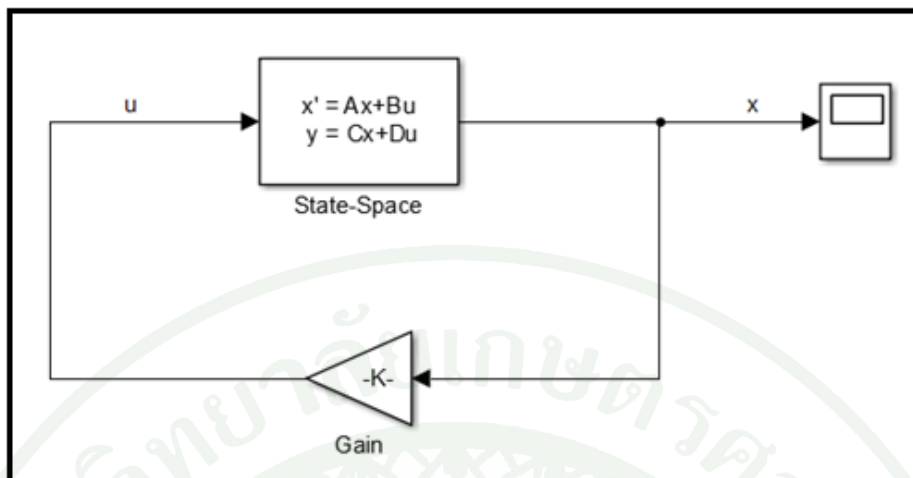
ภาพที่ 20 บล็อกไดอะแกรมการวัดค่าจากเซนเซอร์ทั้งสามชนิด

### 3. ตัวคุมค่ากำลังสองเชิงเส้น (Linear Quadratic Regulator, LQR)

ในวิธีการวางตำแหน่งโพลนั้นมีความเหมาะสมสำหรับระบบที่มีอันดับไม่สูงมาก หรือระบบที่มีลักษณะเด่นเป็นพลวัตอันดับสองและข้อจำกัดเรื่อง ระดับความเป็นอิสระ (degree of freedom) ตัวคุมค่ากำลังสองเชิงเส้นคือการเลือกอัตราขยายของตัวป้อนกลับสเตต ( $K$ ) ที่เหมาะสมที่สุด โดยการออกแบบสัญญาณควบคุมซึ่งทำให้ค่า (Cost function,  $J$ ) มีค่าน้อยที่สุด (minimization) กำหนดให้ (Cost function,  $J$ ) คือ

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (39)$$

โดยที่  $Q$  เป็นเมตริกซ์สมมาตรที่เป็นบวกแน่นอน (positive-definite) และเป็นเมตริกซ์ถ่วงน้ำหนักให้ความสำคัญในการควบคุมของสเตตแต่ละตัว  
 $R$  เป็นเมตริกซ์สมมาตรที่เป็นกึ่งบวกแน่นอน (semi-definite) และเป็นเมตริกซ์ถ่วงน้ำหนักให้แก่สัญญาณควบคุม



ภาพที่ 21 บล็อกไดอะแกรมในการควบคุมป้อนกลับแบบรักษาเสถียรภาพ

กำหนดสมการปริภูมิสถานะเชิงเส้นคือ

$$\dot{x} = Ax + Bu \quad (40)$$

$$y = Cx$$

$$u(t) = -Kx(t) \quad (41)$$

เมื่อ  $x = [\theta \ \psi \ \dot{\theta} \ \dot{\psi} \ \phi \ \dot{\phi}]$ ,  $u = [v_l \ v_r]$

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \end{bmatrix}$$

แทนค่า  $u$  (41) ลงในสมการ (40) จะได้

$$\dot{x} = Ax - BKx = (A - BK)x \quad (42)$$

กำหนดให้เมตริกซ์  $A - BK$  มีเสถียรภาพหรือมีค่าเฉพาะจริงเป็นค่าจริงเครื่องหมายลบทั้งหมด เมื่อแทนค่า  $u(t)$  ในสมการ Cost function (39) จะได้

$$\begin{aligned}
 J &= \int_0^{\infty} (x^T Q x + x^T K^T R K x) dt \\
 &= \int_0^{\infty} x^T (Q + K^T R K) dt
 \end{aligned} \tag{43}$$

กำหนดให้

$$x^T (Q + K^T R K) x = -\frac{d}{dt} (x^T P x)$$

โดยที่  $P$  เป็นเมตริกซ์สมมาตรที่เป็นบวกแน่นอน (positive-definite) เราจะได้

$$\begin{aligned}
 x^T (Q + K^T R K) x &= -\dot{x}^T P x - x^T P \dot{x} = -x^T [(A - BK)^T P + P(A - BK)] x \\
 (A - BK)^T P + P(A - BK) &= -(Q + K^T R K)
 \end{aligned} \tag{44}$$

สมการ (44) แสดงให้เห็นว่า  $A - BK$  เป็นเมตริกซ์ที่มีเสถียรภาพ ถ้าระบบมีเสถียรภาพแล้ว จะมีค่า  $P$  ที่เป็นเมตริกซ์สมมาตรที่เป็นบวกแน่นอนได้เพียงค่าเดียว ดังนั้นค่า  $P$  อื่นๆ ที่ได้จากการแก้สมการ (44) จะไม่เป็นเมตริกซ์สมมาตรที่เป็นบวกแน่นอน

จากสมการ Cost function จะได้

$$J = \int_0^{\infty} x^T (Q + K^T R K) x dt = -x^T(\infty) P x(\infty) + x^T(0) P x(0) \tag{45}$$

ค่า eigenvalues ของ  $A - BK$  ถูกกำหนดให้เป็นลบทั้งหมด หรือ  $x(\infty) \rightarrow 0$

ดังนั้นเราจะได้

$$J = x^T(0) P x(0)$$

ดังนั้น Cost function สามารถหาได้จากเทอมของ  $x(0)$  และเมตริกซ์  $P$

เพื่อหาคำตอบ กำหนดให้

$$R = T^T T$$

เมื่อ  $R$  เป็นเมตริกซ์สมมาตรที่เป็นบวกแน่นอน และ  $T$  ไม่เป็นเมตริกซ์หนึ่ง จากสมการ (44) สามารถเขียนได้ดังนี้

$$(A^T - K^T B^T)P + P(A - BK) + Q + K^T T^T TK = 0$$

$$A^T P + PA + [TK - (T^T)^{-1} B^T P]^T [TK - (T^T)^{-1} B^T P] - PBR^{-1} B^T P + Q = 0 \quad (46)$$

โดยพยายามทำให้ค่า  $J$  ต่ำสุดโดยการปรับค่า  $K$  หรือต้องการค่าต่ำสุดของ

$$x^T [TK - (T^T)^{-1} B^T P]^T [TK - (T^T)^{-1} B^T P] x$$

เนื่องจากเป็นพจน์ที่มีค่าเป็นกึ่งบวกแน่นอนดังนั้นค่าต่ำสุดเป็นศูนย์ และจะเป็นเช่นนั้นเมื่อ

$$TK = (T^T)^{-1} B^T P$$

$$\text{ดังนั้น} \quad K = T^{-1} (T^T)^{-1} B^T P = R^{-1} B^T P \quad (47)$$

สมการ (47) เป็นสมการที่ให้ค่า  $K$  ที่เหมาะสมที่สุดสำหรับระบบดังกล่าว ดังนั้นเราสามารถคำนวณหาสัญญาณควบคุมที่เหมาะสมที่สุดคือ

$$u(t) = -Kx(t) = -R^{-1} B^T P x(t) \quad (48)$$

ซึ่งเมตริกซ์  $P$  ในสมการ (48) ต้องสอดคล้องกับสมการ (44) หรือที่ลดรูปลงเป็น Reduced-matrix Riccati Equation เขียนเป็นสมการดังนี้

$$A^T P + PA - PBR^{-1} B^T P + Q = 0 \quad (49)$$

เมื่อสามารถหาค่าเมตริกซ์  $P$  ซึ่งเป็นคำตอบของสมการ Reduced-matrix Riccati Equation และ  $P$  เป็นเมตริกซ์สมมาตรที่เป็นบวกแน่นอน ก็จะแสดงได้ว่าระบบนั้นมีเสถียรภาพ จากนั้นจึงนำค่า  $P$  แทนค่าในสมการ (47) ก็จะได้ค่า  $K$  ที่เหมาะสมที่สุด ซึ่งในทางปฏิบัติการคำนวณหาค่า

$K$  เราสามารถหาได้ใช้โปรแกรม Matlab โดยการป้อนคำสั่งคือ  $K = lqr(A, B, Q, R)$  โดยที่  $A, B$  คือพลาเนตจากสมการ (32) และค่า  $Q$  และ  $R$  เป็นเมตริกซ์ถ่วงน้ำหนักของ Cost function,  $J$  ซึ่งหาได้โดยการลองผิดลองถูกจากการทดลอง

ในส่วนของผลการจำลองในงานวิจัยนี้จะกำหนดให้ติดตามสัญญาณดังนี้คือ

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (50)$$

$$u = Ke \quad (51)$$

$$\begin{aligned} \text{เมื่อ } x &= [\theta \quad \psi \quad \dot{\theta} \quad \dot{\psi} \quad \phi \quad \dot{\phi}]^T, u = [v_l \quad v_r]^T \\ r &= [\theta_{ref} \quad \psi_{ref} \quad 0 \quad 0 \quad \phi_{ref} \quad 0]^T \\ e &= r - y \\ K &= \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \end{bmatrix} \end{aligned}$$

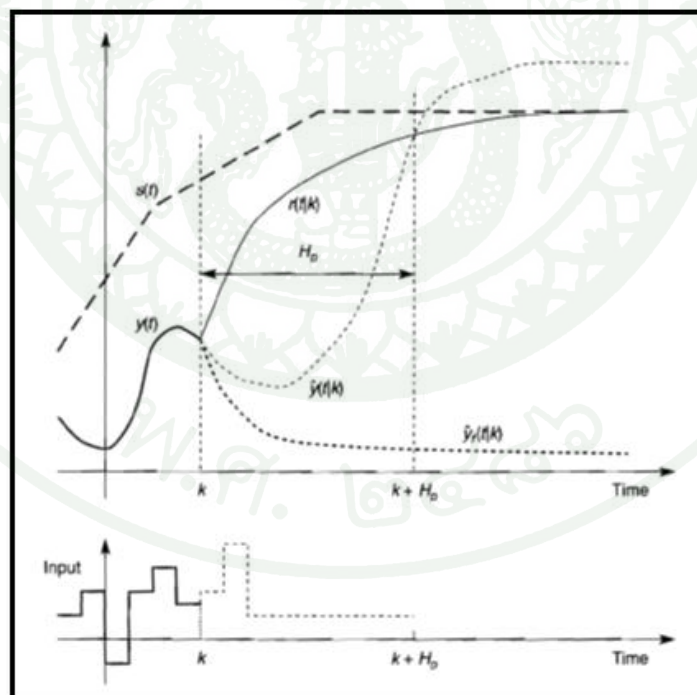
การออกแบบตัวขยาย  $K$  ให้สมการ Cost function ที่มีลักษณะคล้ายกับสมการ (43) ให้มีค่าต่ำสุด โดยสมการ Cost function สำหรับระบบนี้คือ

$$J = \int_0^{\infty} (x^T Q x + e^T K^T R K e) dt \quad (52)$$

#### 4. การออกแบบตัวควบคุม โมเดลพรีดิกทีฟ (Model predictive control)

ตัวควบคุม โมเดลพรีดิกทีฟได้รับความสนใจจากวงการอุตสาหกรรมและนักวิชาการต่างๆ ซึ่งจุดเด่นที่สำคัญคือการกำหนดข้อจำกัด (constraint) ให้กับ อินพุต (input), เอาต์พุต (output) และการเปลี่ยนแปลงของอินพุต นอกจากนี้ยังสามารถจัดการกับสิ่งรบกวนภายนอก (disturbance) และความไม่แน่นอน (uncertainty) ซึ่งมีความเหมาะสมที่จะนำมาใช้กับหุ่นยนต์สองล้อที่มีสิ่งรบกวนภายนอกและความไม่แน่นอนสูง

4.1 หลักการพื้นฐานของการออกแบบตัวควบคุมโมเดลพรีดิกทีฟอธิบายดังต่อไปนี้  
 ทีมงานวิศวกรกลุ่มหนึ่งต้องการออกแบบและควบคุมหุ่นยนต์สองล้อโดยใช้ตัว  
 ควบคุมโมเดลพรีดิกทีฟ ซึ่งมีกฎในการดำเนินงานคือจะต้องวางแผนงานที่จะทำล่วงหน้าไป 8 ชม.  
 และจะใช้แผนงานที่วางไว้เฉพาะชม.แรกเท่านั้น จะใช้กฎนี้จนทำการควบคุมหุ่นยนต์สองล้อสำเร็จ  
 ทีมงานวิศวกรได้เริ่มทำงานเวลา 9 โมงเช้าได้เริ่มวางแผนล่วงหน้าไป 8 ชม. อาทิเช่นพิจารณาเกี่ยวกับ  
 อุปกรณ์ต่างๆที่ใช้สร้างหุ่นยนต์สองล้อ การหาแบบจำลองทางคณิตศาสตร์ การออกแบบการวัดมุม  
 เอียงของหุ่นยนต์ ตัวควบคุมที่จะนำมาใช้งาน หลักการการรักษาสมดุลของหุ่นยนต์สองล้อ ซึ่งเริ่ม  
 ดำเนินงานต่างๆนี้ให้สำเร็จทันเวลา 5 โมงเย็น จากนั้นทีมงานวิศวกรได้เริ่มดำเนินงานตามแผนที่  
 วางไว้เฉพาะช่วงเวลา 9 โมง ถึง 10 โมงเช้า เมื่อถึงเวลา 10 โมงเช้าทีมงานได้ตรวจเช็คสิ่งที่ทำไปใน  
 ชม.แรกว่าทำงานส่วนใดเรียบร้อยและส่วนใดที่ยังทำไม่สำเร็จ แล้วจึงนำข้อมูลที่อัปเดตจากชั่วโมง  
 แรกมาวางแผนใหม่โดยวางแผนล่วงหน้าไป 8 ชม. (ระยะเวลาที่วางแผนล่วงหน้าเท่าเดิมคือ 8 ชม.)  
 โดยแผนที่วางไว้ว่างานจะสำเร็จก่อน 6 โมงเย็น จะใช้เฉพาะช่วงเวลา 10 โมง ถึง 11 โมง ซึ่งจะใช้  
 กฎนี้ไปจนกระทั่งทำการควบคุมหุ่นยนต์สองล้อสำเร็จ



ภาพที่ 22 พื้นฐานของตัวควบคุมโมเดลพรีดิกทีฟ

ที่มา : Maciejowski, (2002)

$s(t)$	จุดที่ต้องการตามรอยสัญญาณ (set point trajectory)
$r(t/k)$	สัญญาณอ้างอิงที่ต้องการตามรอยสัญญาณ (reference trajectory) เริ่มต้นจากเอาต์พุตปัจจุบัน $y(k)$ และสมมุติให้เข้าสู่จุดที่ต้องการตามรอยสัญญาณแบบ exponentially
$y(t)$	เอาต์พุตก่อนหน้า
$\hat{y}(t/k)$	เอาต์พุตในอนาคตที่ทำนาย ณ เวลาปัจจุบัน
$k$	เวลาปัจจุบันตั้งตัวอย่างนี้เริ่มต้นที่เวลา 9 โมง ดังนั้น $k = 9$
$H_p$	Prediction horizon ตั้งตัวอย่างนี้ $H_p = 8$
$H_c$	Control horizon

ค่าความผิดพลาดที่เวลาปัจจุบันคือ

$$\varepsilon(k) = s(k) - y(k) \quad (53)$$

เมื่อสัญญาณอ้างอิงที่ต้องการตามรอยสัญญาณเลือกค่าความผิดพลาดที่  $i$  สเต็ปก่อนหน้าจะได้

$$\varepsilon(k+i) = e^{-iT_s/T_{ref}} \varepsilon(k) \quad (54)$$

$$\varepsilon(k+i) = \lambda^i \varepsilon(k) \quad (55)$$

โดยที่  $T_{ref}$  ความเร็วในการตอบสนอง

$T_s$  sampling interval

และ  $\lambda = e^{-T_s/T_{ref}}$

สัญญาณอ้างอิงที่ต้องการตามรอยสัญญาณกำหนดเป็น

$$r(k+i/k) = s(k+i) - \varepsilon(k+i) \quad (56)$$

$$r(k+i/k) = s(k+i) - e^{-iT_s/T_{ref}} \varepsilon(k) \quad (57)$$

#### 4.2 ตัวควบคุมโมเดลพรีดิกทีฟสำหรับระบบหนึ่งอินพุตและหนึ่งเอาต์พุต (SISO)

กำหนดสมการปริภูมิสถานะเชิงเส้นคือ

$$x_m(k+1) = A_m x_m(k) + B_m u(k) \quad (58)$$

$$y(k) = C_m x_m(k) + D_m u(k) \quad (59)$$

โดยที่  $x_m$  คือเวกเตอร์ของแต่ละสเตทมีขนาด  $n_1$

$u$  คืออินพุตที่ให้แก่ระบบ

$y$  คือเอาต์พุตของระบบ

เนื่องจากสมมุติให้อินพุตไม่มีผลกระทบต่อเอาต์พุตจึงกำหนดให้  $D_m = 0$

หาผลต่างเวลาของสมการที่ (58) เราจะได้

$$x_m(k+1) - x_m(k) = A_m (x_m(k) - x_m(k-1)) + B_m (u(k) - u(k-1))$$

$$\Delta x_m(k+1) = x_m(k+1) - x_m(k)$$

$$\Delta x_m(k) = x_m(k) - x_m(k-1)$$

และผลต่างเวลาของอินพุตคือ

$$\Delta u(k) = u(k) - u(k-1)$$

ดังนั้นเขียนผลต่างของปริภูมิสเตทได้ดังนี้

$$\Delta x_m(k+1) = A_m \Delta x_m(k) + B_m \Delta u(k) \quad (60)$$

ขั้นตอนต่อไปคือการสร้างความสัมพันธ์ระหว่างผลต่างของสเตทและเอาต์พุตโดยการกำหนดสเตทเวกเตอร์ใหม่ดังนี้

$$x(k) = \begin{bmatrix} \Delta x_m(k)^T & y(k) \end{bmatrix}^T$$

ผลต่างเวลาของเอาต์พุตคือ

$$\begin{aligned} y(k+1) - y(k) &= C_m (x_m(k+1) - x_m(k)) \\ y(k+1) - y(k) &= C_m \Delta x_m(k+1) \\ y(k+1) - y(k) &= C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k) \end{aligned} \quad (61)$$

จากสมการที่ (60) และ (61) เขียนปริภูมิสถานะใหม่คือ

$$\begin{aligned} \begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix} &= \begin{bmatrix} A_m & o_m^T \\ C_m A_m & 1 \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} \Delta u(k) \\ y(k) &= \begin{bmatrix} o_m & 1 \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \end{aligned} \quad (62)$$

โดยที่  $o_m = [0 \ 0 \ \dots \ 0]$  และ  $(A, B, C)$  คือส่วนขยายของโมเดล (augmented model) ซึ่งจะนำไปใช้ต่อในส่วนถัดไปของการออกแบบตัวควบคุมโมเดลพรีดิกทีฟ

#### 4.3 การทำนายล่วงหน้าของสเตตและเอาต์พุต

สมมุติให้เวลาปัจจุบัน  $k_i, k_i > 0$  และข้อมูลของสเตตที่เวลาปัจจุบันสามารถวัดได้

กำหนดอินพุตในอนาคตคือ

$$\Delta u(k_i), \Delta u(k_i + 1), \dots, \Delta u(k_i + H_c - 1)$$

และกำหนดสเตตในอนาคต

$$x(k_i + 1/k_i), x(k_i + 2/k_i), \dots, x(k_i + m/k_i), \dots, x(k_i + H_p/k_i)$$

โดยที่  $x(k_i + m/k_i)$  คือสเตตที่ทำนายที่เวลา  $k_i + m$  โดยใช้ข้อมูลจากเวลาปัจจุบัน

จากสมการที่ (62) เมตริกซ์  $(A, B, C)$  สามารถเขียนให้อยู่ในรูปของการทำนายสเตตดังนี้

$$\begin{aligned}
x(k_i + 1 / k_i) &= Ax(k_i) + B\Delta u(k_i) \\
x(k_i + 2 / k_i) &= Ax(k_i + 1 / k_i) + B\Delta u(k_i + 1) \\
x(k_i + 3 / k_i) &= A^2x(k_i) + AB\Delta u(k_i) + B\Delta u(k_i + 1) \\
&\vdots \\
x(k_i + H_p / k_i) &= A^{H_p}x(k_i) + A^{H_p-1}B\Delta u(k_i) + A^{H_p-2}B\Delta u(k_i + 1) \\
&\quad + \dots + A^{H_p-H_c}B\Delta u(k_i + H_c - 1)
\end{aligned} \tag{63}$$

สามารถเขียนสมการการทำนายของเอาต์พุตคือ

$$\begin{aligned}
y(k_i + 1 / k_i) &= CAx(k_i) + CB\Delta u(k_i) \\
y(k_i + 2 / k_i) &= CA^2x(k_i) + CAB\Delta u(k_i) + CB\Delta u(k_i + 1) \\
y(k_i + 3 / k_i) &= CA^3x(k_i) + CA^2B\Delta u(k_i) + CAB\Delta u(k_i + 1) \\
&\quad + CB\Delta u(k_i + 2) \\
&\vdots \\
y(k_i + H_p / k_i) &= CA^{H_p}x(k_i) + CA^{H_p-1}B\Delta u(k_i) + CA^{H_p-2}B\Delta u(k_i + 1) \\
&\quad + \dots + CA^{H_p-H_c}B\Delta u(k_i + H_c - 1)
\end{aligned} \tag{64}$$

สเตตทุกตัวที่ทำนายสามารถเขียนอยู่ในรูปของสเตตปัจจุบัน  $x(k_i)$  และอินพุตในอนาคต  $\Delta u(k_i + j)$  ซึ่ง  $j = 0, 1, \dots, H_c - 1$

กำหนดเวกเตอร์

$$\begin{aligned}
Y &= [y(k_i + 1 / k_i) \quad y(k_i + 2 / k_i) \quad y(k_i + 3 / k_i) \quad \dots \quad y(k_i + H_p / k_i)]^T \\
\Delta U &= [\Delta u(k_i) \quad \Delta u(k_i + 1) \quad \Delta u(k_i + 2) \quad \dots \quad \Delta u(k_i + H_c - 1)]^T
\end{aligned}$$

โดยที่  $Y$  มีขนาด  $H_p$  และ  $\Delta U$  มีขนาด  $H_c$

จัดรูปสมการที่ (63) และ (64) ได้ดังนี้

$$Y = Fx(k_i) + \Phi\Delta U \tag{65}$$

ซึ่งกำหนดให้

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{H_p} \end{bmatrix}; \Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ CA^{H_p-1}B & CA^{H_p-2}B & CA^{H_p-3}B & \dots & CA^{H_p-H_c}B \end{bmatrix}$$

#### 4.4 ข้อจำกัด (Constraints) ของอินพุตและเอาต์พุต

ในส่วนนี้จะกำหนดข้อจำกัดให้แก่การเปลี่ยนแปลงอินพุตเริ่มที่เวลาปัจจุบัน  $k_i$

$$\Delta u^{\min} \leq \Delta u(k_i) \leq \Delta u^{\max}$$

ข้อจำกัดของการเปลี่ยนแปลงอินพุตล่วงหน้า 3 sample time เขียนได้ดังนี้

$$\Delta u^{\min} \leq \Delta u(k_i) \leq \Delta u^{\max}$$

$$\Delta u^{\min} \leq \Delta u(k_i + 1) \leq \Delta u^{\max}$$

$$\Delta u^{\min} \leq \Delta u(k_i + 2) \leq \Delta u^{\max}$$

ค่าของ sample time ที่จะทำนายล่วงหน้ามีความสำคัญเพราะมีความสัมพันธ์เกี่ยวกับภาระการคำนวณของไมโครคอนโทรลเลอร์ ดังนั้นไม่ควรเลือกค่าของ  $H_c$  ที่สูงมาก ซึ่งเราสามารถเขียนอยู่ในรูปอย่างง่ายดังนี้

$$\Delta U^{\min} \leq \Delta U \leq \Delta U^{\max}$$

เขียนได้สองอสมการดังนี้

$$-\Delta U \leq -\Delta U^{\min} \tag{66}$$

$$\Delta U \leq \Delta U^{\max} \tag{67}$$

เขียนในรูปเมตริกซ์

$$\begin{bmatrix} -I \\ I \end{bmatrix} \Delta U \leq \begin{bmatrix} -\Delta U^{\min} \\ \Delta U^{\max} \end{bmatrix} \quad (68)$$

เมื่อเขียนอยู่ในรูปของการทำนaylor ล่วงหน้าในทุก sample time และทุกข้อจำกัดข้างในเทอมของ  $\Delta U$  ได้ดังนี้

$$\begin{bmatrix} u(k_i) \\ u(k_i+1) \\ u(k_i+2) \\ \vdots \\ u(k_i+N_c-1) \end{bmatrix} = \begin{bmatrix} I \\ I \\ I \\ \vdots \\ I \end{bmatrix} u(k_i-1) + \begin{bmatrix} I & 0 & 0 & \dots & 0 \\ I & I & 0 & \dots & 0 \\ I & I & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & I & \dots & I & I \end{bmatrix} \begin{bmatrix} u(k_i) \\ u(k_i+1) \\ u(k_i+2) \\ \vdots \\ u(k_i+N_c-1) \end{bmatrix} \quad (69)$$

และเขียนให้สั้นลงในรูปของเมทริกซ์ด้วยการแทนค่า  $C_1, C_2$  ได้ดังนี้

$$-(C_1 u(k_i-1) + C_2 \Delta U) \leq -U^{\min} \quad (70)$$

$$(C_1 u(k_i-1) + C_2 \Delta U) \leq U^{\max} \quad (71)$$

โดยที่  $U^{\min}$  และ  $U^{\max}$  มีขนาดเวกเตอร์  $H_c$  ของ  $u^{\min}$  และ  $u^{\max}$  ตามลำดับ ดังนั้นสามารถเขียนข้อจำกัดของเอาต์พุตในเทอมของ  $\Delta U$  ได้ดังนี้

$$Y^{\min} \leq Fx(k_i) + \Phi \Delta U \leq Y^{\max} \quad (72)$$

#### 4.5 ตัวควบคุมโมเดลพรีดิกทีฟสำหรับระบบหลายอินพุตและหลายเอาต์พุต (MIMO)

สำหรับระบบ MIMO สิ่ง que เพิ่มเติมเข้ามาคือสัญญาณอินพุตของสิ่งรบกวนภายนอก ระบบ ( $w(k)$ ) และ สมมุติให้อินพุตจำนวน  $m$  และ เอาต์พุตจำนวน  $q$

กำหนดสมการปริภูมิสถานะเชิงเส้นสำหรับระบบ MIMO

$$\begin{aligned} x_m(k+1) &= A_m x_m(k) + B_m u(k) + B_d w(k) \\ y(k) &= C_m x_m(k) \end{aligned} \quad (73)$$

โดยที่  $x_m$  คือสเตตเวกเตอร์,  $y$  คือเอาต์พุต (output),  $u$  คือสัญญาณอินพุต (input)

$w(k)$  คือสัญญาณอินพุตของสิ่งรบกวนภายนอกของระบบ (input disturbance)

ลำดับของสัญญาณสิ่งรบกวนภายนอกของระบบแสดงได้ดังสมการ

$$w(k) - w(k-1) = \epsilon(k) \quad (74)$$

จากสมการที่ (73) เขียนในรูปผลต่างสมการคือ

$$x_m(k) = A_m x_m(k-1) + B_m u(k-1) + B_d w(k-1) \quad (75)$$

กำหนดให้  $\Delta x_m(k) = x_m(k) - x_m(k-1)$  และ  $\Delta u(k) = u(k) - u(k-1)$  จากสมการ (74) และ (75) จะได้ดังนี้

$$\Delta x_m(k+1) = A_m \Delta x_m(k) + B_m \Delta u(k) + B_d \epsilon(k) \quad (76)$$

ดังนั้นความสัมพันธ์ระหว่างเอาต์พุตกับสเตตคือ

$$\Delta y(k+1) = C_m \Delta x_m(k+1) = C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k) + C_m B_d \epsilon(k)$$

โดยที่  $\Delta y(k+1) = y(k+1) - y(k)$

เราสามารถเขียนสมการปริภูมิสถานะเชิงเส้นสำหรับระบบ MIMO ได้ดังนี้

$$\begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} A_m & o_m^T \\ C_m A_m & I_{q \times q} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} \Delta u(k) + \begin{bmatrix} B_d \\ C_m B_d \end{bmatrix} \epsilon(k)$$

$$y(k) = \begin{bmatrix} o_m & I_{q \times q} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}$$

โดยที่  $I_{q \times q}$  คือเมตริกซ์เอกลักษณ์ที่มีมิติ  $q \times q$  เมื่อ  $q$  คือจำนวนของเอาต์พุต,  $m$  คือจำนวนของอินพุต และ  $o_m$  คือเมตริกซ์ศูนย์ที่มีมิติ  $q \times n_1$  ในขณะที่เมตริกซ์  $A_m, B_m$  และ  $C_m$  มีมิติเท่ากับ  $n_1 \times n_1, n_1 \times m$  และ  $q \times n_1$

#### 4.6 การหาสัญญาณควบคุมที่เหมาะสมที่สุด (Optimization)

การทำให้เอาต์พุตที่เกิดจากการทำนายเข้าใกล้สัญญาณอ้างอิง (Set-point) จึงต้องการหาค่า  $\Delta u$  ที่เหมาะสมและทำให้  $J$  (Costfunction) มีค่าน้อยที่สุด กำหนดเวกเตอร์สัญญาณอ้างอิง (Set-potion) ดังนี้คือ

$$R_s^T = \overbrace{[1 \ 1 \ \dots \ 1]}^{N_p} r(k_i),$$

กำหนด Cost Function J คือ

$$J = (R_s - Y)^T (R_s - Y) + \Delta U^T \bar{R} \Delta U, \quad (77)$$

โดยที่เทอมแรกคือการลดค่าความผิดพลาดให้น้อยที่สุดระหว่างการทำนายจากเอาต์พุต (predicted output) และสัญญาณอ้างอิง (Set-point)  $\bar{R}$  คือเมตริกซ์ทแยงมุม (diagonal matrix)

$$\bar{R} = r_w I_{N_c \times N_c} \quad (78)$$

เมื่อ  $r_w$  เป็นตัวแปรที่ใช้ในการปรับสมรรถนะของระบบปิดและมีค่ามากกว่าหรือเท่ากับศูนย์ เพื่อหาสัญญาณควบคุมที่เหมาะสมโดยใช้สมการที่ (65) จะได้

$$J = (R_s - Fx(k_i))^T (R_s - Fx(k_i)) - 2\Delta U^T \Phi^T (R_s - Fx(k_i)) + \Delta U^T (\Phi^T \Phi + \bar{R}) \Delta U. \quad (79)$$

จากอนุพันธ์อันดับหนึ่งสมการ (79) จะได้สมการ

$$\frac{\partial J}{\partial \Delta U} = 0,$$

เราจะได้  $\Delta U$  ที่เหมาะสมดังนี้

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (R_s - Fx(k_i)), \quad (80)$$

เมตริกซ์  $(\Phi^T \Phi + \bar{R})^{-1}$  คือ Hessian matrix และ  $R_s$  คือ

$$R_s = \overbrace{[1 \ 1 \ 1 \ \dots \ 1]}^{N_p} r(k_i) = \bar{R}_s r(k_i),$$

$$\text{ซึ่ง } \bar{R}_s = \overbrace{[1 \ 1 \ 1 \ \dots \ 1]^T}^{N_p}$$

ดังนั้นการหาสัญญาณควบคุมที่เหมาะสมที่สุดเชื่อมโยงถึงสัญญาณอ้างอิง (Set-point)  $r(k_i)$  และตัวแปรสแตท  $x(k_i)$  เขียนสมการได้ดังนี้

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (\bar{R}_s r(k_i) - Fx(k_i)). \quad (81)$$

ในการคำนวณเชิงตัวเลขงานวิจัยนี้ใช้ Model Predictive Control Toolbox ของโปรแกรม MATLAB ในการคำนวณ คำสั่งโดยการออกแบบดูได้ในภาคผนวก ง. ซึ่ง Costfunction คือ

$$J = \sum_{i=0}^{p-1} [y(k+i+1/k) - r(k+i+1)]^T Q [y(k+i+1/k) - r(k+i+1)] \\ + \Delta u(k+i/k)^T R_{\Delta u} \Delta u(k+i/k) + [u(k+i/k) - u(k+i)]^T R_u [u(k+i/k) - u(k+i)]$$

โดยที่  $Q, R_{\Delta u}$  และ  $R_u$  เป็นเมทริกซ์ถ่วงน้ำหนักของเอาต์พุต, การเปลี่ยนแปลงของอินพุตและอินพุตตามลำดับ

## ผลและวิจารณ์

### ผลการจำลองด้วยโปรแกรม MATLAB

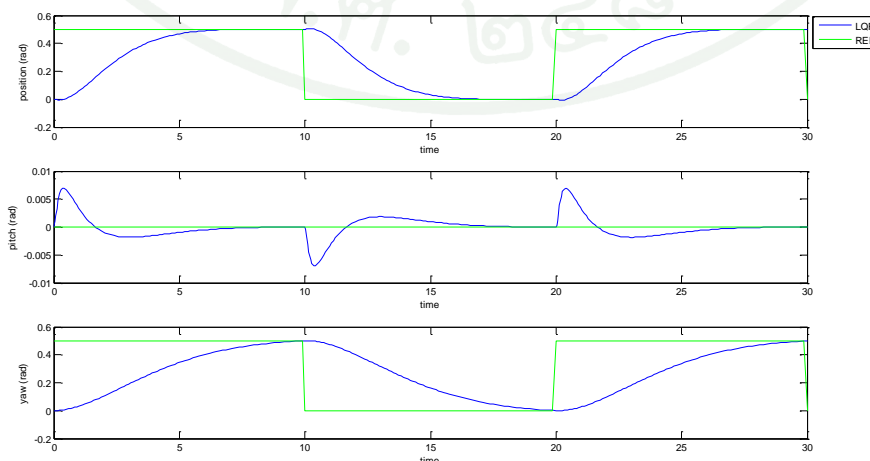
ในส่วนของผลและวิจารณ์นี้ มุ่งเน้นเกี่ยวกับประสิทธิภาพในการรักษาเสถียรภาพของตัวควบคุมค่ากำลังสองเชิงเส้นและการออกแบบตัวควบคุมโมเดลพรีดิกทีฟ ซึ่งจะนำตัวควบคุมทั้งสองมาเปรียบเทียบกัน แสดงถึงคุณสมบัติเฉพาะของตัวควบคุมโมเดลพรีดิกทีฟในเรื่องของข้อจำกัด (constraints) โดยจะใช้โปรแกรม MATLAB ในการจำลองผล

#### 1. ตัวควบคุมกำลังสองเชิงเส้น (Linear Quadratic Regulator, LQR)

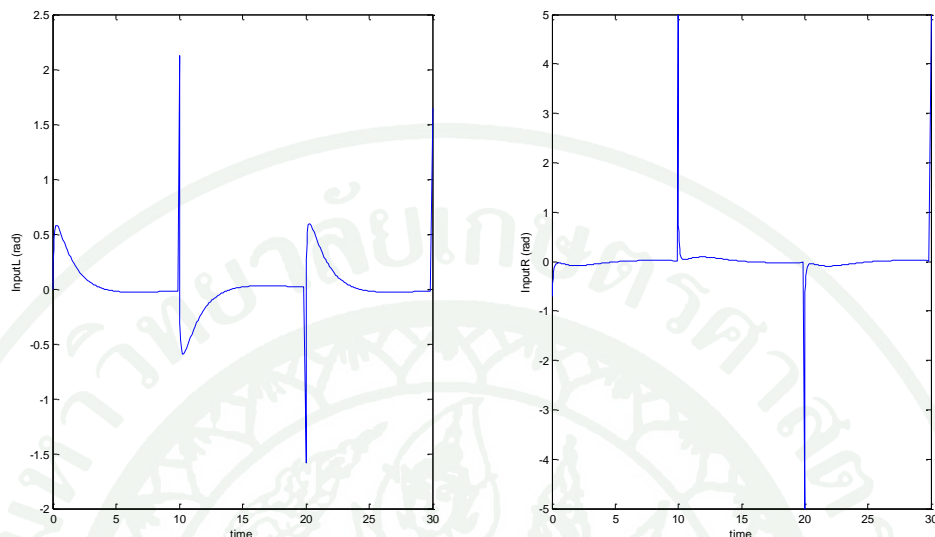
กำหนดให้ เมทริกซ์ถ่วงน้ำหนัก ( $Q, R$ ) และอัตราขยาย ( $K$ ) คือ

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$K = \begin{bmatrix} -0.7071 & -72.3187 & -7.8973 & -1.1923 & -0.7071 & -1.6314 \\ -0.7071 & -72.3187 & -7.8973 & -1.1923 & 0.7071 & 1.6314 \end{bmatrix}$$



ภาพที่ 23 เอาต์พุตของตัวคุมค่ากำลังสองเชิงเส้น



ภาพที่ 24 สัญญาณควบคุมของตัวคุมค่ากำลังสองเชิงเส้น

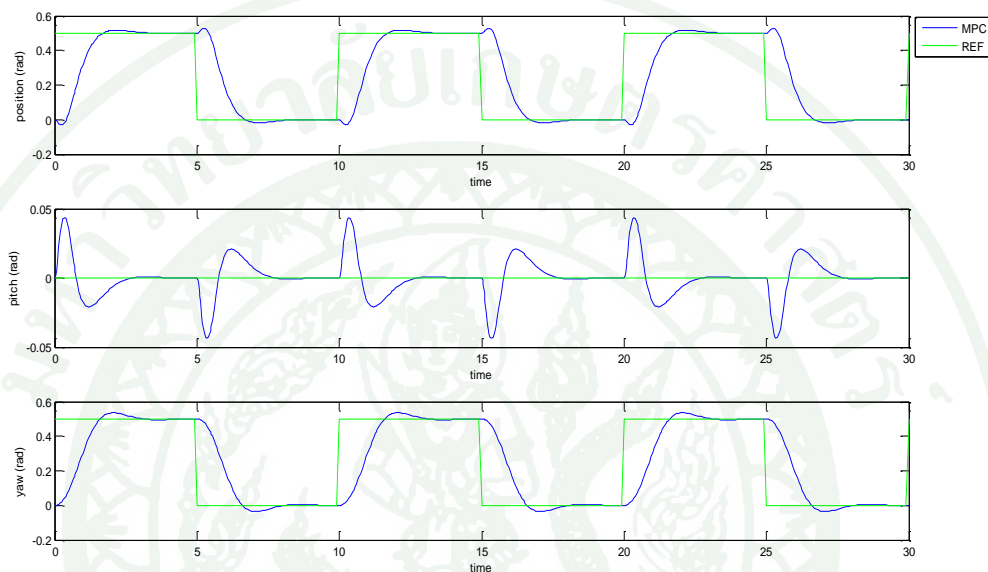
กำหนดให้ตำแหน่งเริ่มต้นที่ตำแหน่ง position ( $\theta$ ) = 0.5 rad และตำแหน่งมุมเลี้ยว yaw ( $\phi$ ) = 0.5 rad ซึ่งตัวคุมค่ากำลังสองเชิงเส้นสามารถติดตามตำแหน่งอ้างอิง (Reference) ที่เป็นสัญญาณแบบ Pulse Generator (เส้นสีเขียว) ในส่วนของตำแหน่ง (position) นั้นติดตามได้ดีกว่าในส่วนของมุมเลี้ยว (yaw) ในการเลือกค่าของเมตริกซ์ถ่วงน้ำหนัก ( $Q, R$ ) จะกำหนดให้เป็นเมตริกซ์ทแยงมุมที่เป็น 1 ทั้งหมดเพื่อทดสอบเสถียรภาพของระบบและภายหลังจะนำไปเปรียบเทียบกับตัวควบคุมโมเดลพรีดิกทีฟที่กำหนดเมตริกซ์ถ่วงน้ำหนักแบบเดียวกัน และจากภาพที่ 24 สัญญาณควบคุมของตัวคุมค่ากำลังสองเชิงเส้นนั้นเปลี่ยนแปลงอย่างรวดเร็วซึ่งอาจทำให้ตัวขับเคลื่อน (actuator) ไม่สามารถสั่งการทำงานได้ทัน ซึ่งในส่วนของตัวควบคุมโมเดลพรีดิกทีฟ จะมีความราบเรียบมากกว่า

## 2. ตัวควบคุมโมเดลพรีดิกทีฟ

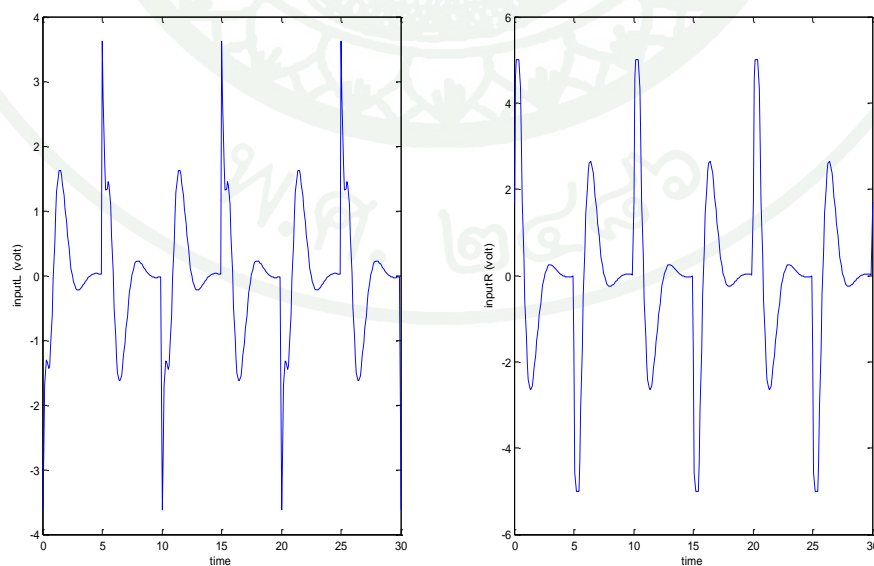
กำหนดค่าพารามิเตอร์ในการออกแบบตัวควบคุมดังนี้คือ Control Interval Time = 0.1, Prediction Horizon ( $H_p$ ) = 10, Control Horizon ( $H_c$ ) = 2, Input Weight (R) = 0.1, Output Weight (Q) = 1 และกำหนดให้ติดตามสัญญาณแบบ Pulse Generator เหมือนกับตัวคุมค่ากำลังสอง

เชิงเส้นที่ตำแหน่ง position ( $\theta$ ) = 0.5 rad และตำแหน่งมุมเลี้ยว yaw ( $\phi$ ) = 0.5 rad ซึ่งจะนำมาเปรียบเทียบกันภายหลังโดยแบ่งการออกแบบเป็น 3 กรณีดังนี้

### 2.1 ตัวควบคุมโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด (MPC without Constraints)

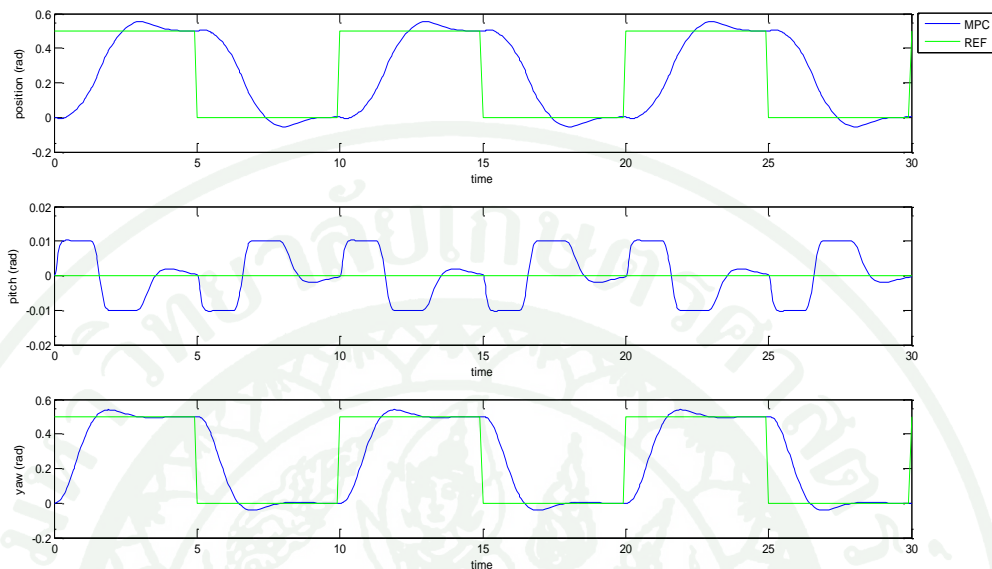


ภาพที่ 25 ตัวควบคุมโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด (MPC without Constraints)

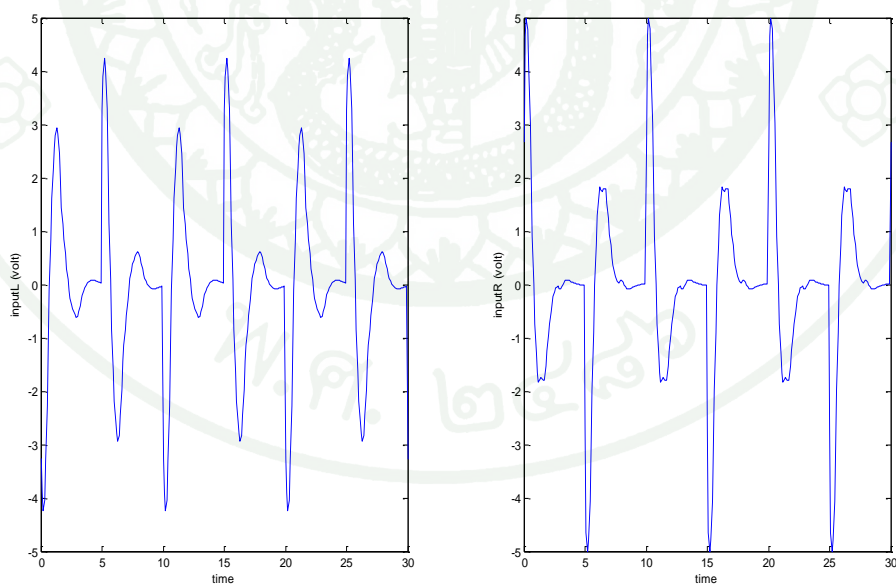


ภาพที่ 26 สัญญาณควบคุมของโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด (MPC without constraints)

## 2.2 กรณีกำหนดข้อจำกัดให้แก่มุมพิตซ์ที่ $\pm 0.01$ rad

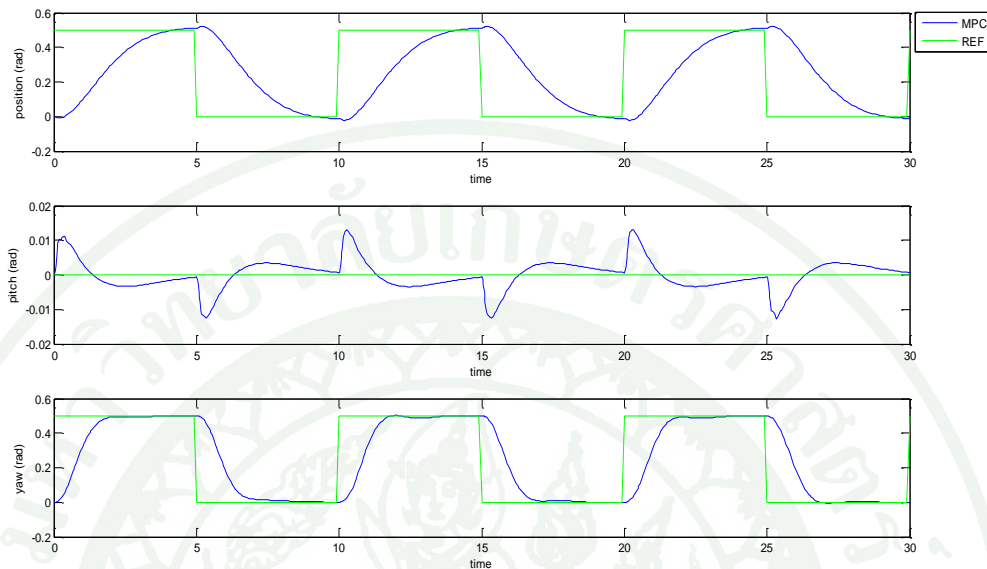


ภาพที่ 27 ตัวควบคุมโมเดลพรีดิกทีฟที่มีข้อจำกัด (MPC with Constraints ( $\pm 0.01$  Pitch))

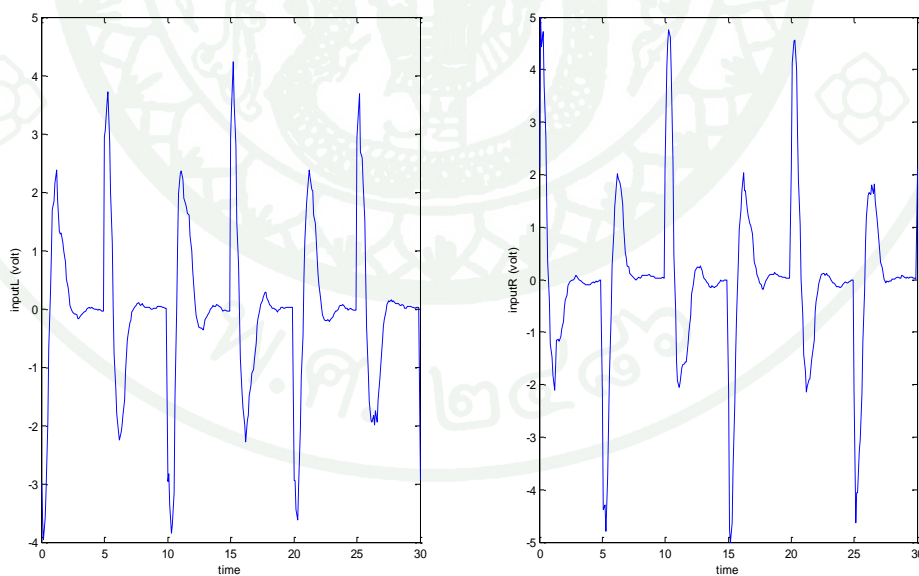


ภาพที่ 28 สัญญาณควบคุมของตัวควบคุมโมเดลพรีดิกทีฟที่มีข้อจำกัด (MPC with constraints ( $\pm 0.01$  Pitch))

### 2.3 กรณีกำหนดให้ ( $H_p = 24$ )



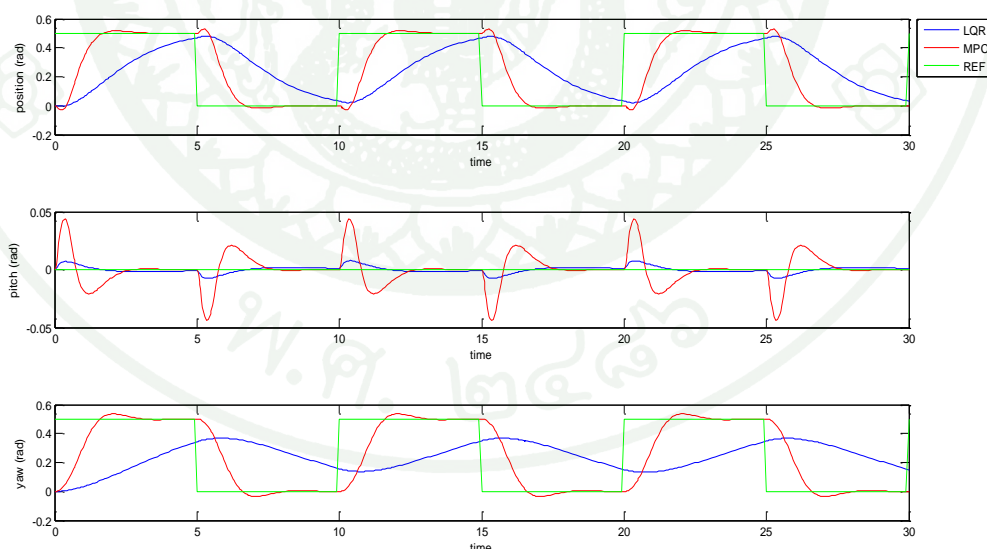
ภาพที่ 29 ตัวควบคุมโมเดลพรีดิกทีฟ (MPC ;  $H_p = 24$ )



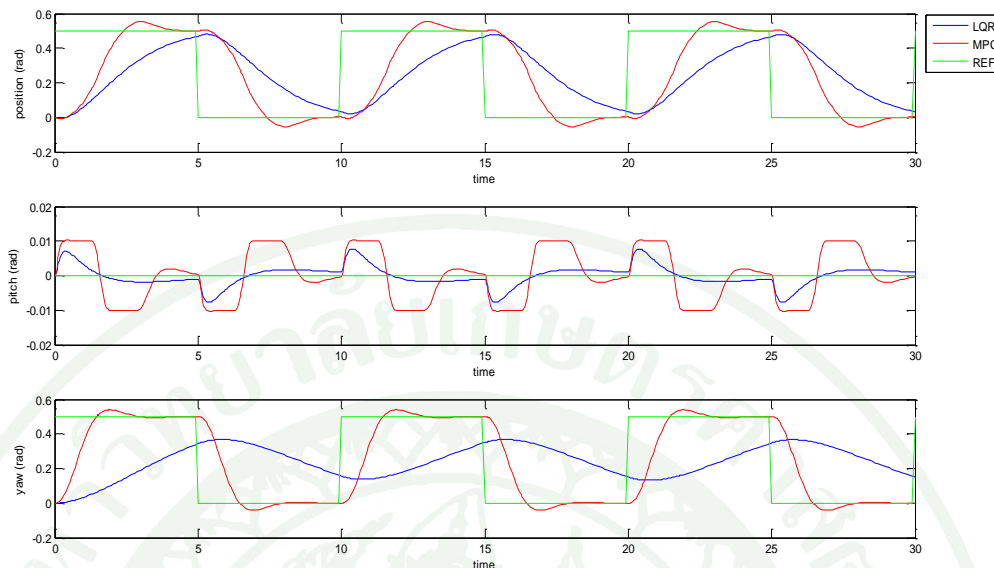
ภาพที่ 30 สัญญาณควบคุมของตัวควบคุมโมเดลพรีดิกทีฟ (MPC ;  $H_p = 24$ )

ภาพที่ 25 ในกรณีของตัวควบคุมโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัดค่าของสัญญาณควบคุมมีความราบเรียบมากกว่าในส่วนของตัวควบคุมกำลังสองเชิงเส้นและสามารถรักษาเสถียรภาพในการติดตามของเอาต์พุตทั้งสามได้ดี ภาพที่ 27 แสดงถึงจุดเด่นของตัวควบคุมโมเดลพรีดิกทีฟนี้คือการกำหนดข้อจำกัดให้อาต์พุต ซึ่งในงานวิจัยกำหนดข้อจำกัดให้แก่มุมพิชท์ที่  $\pm 0.01$  และในส่วนของสัญญาณควบคุมก็มีความราบเรียบเช่นเดียวกับกรณีไม่มีข้อจำกัด ภาพที่ 29 การกำหนดให้  $H_p = 24$  ให้มีค่าสูง ทำให้เสถียรภาพของเอาต์พุตทั้งสามนั้นดีขึ้น แต่จะทำให้ช่วง Transient ช้าลง ซึ่งทำให้การติดตามตำแหน่ง (position) เข้าสู่จุดอ้างอิงได้ช้ากว่ากรณีกำหนดให้  $H_p = 10$  ดังนั้นการเลือกค่าของ Prediction horizon ( $H_p$ ) ให้ตรงกับความต้องการจึงมีความสำคัญต่อการออกแบบระบบควบคุม

### 3. เปรียบเทียบผลการจำลองระหว่างตัวควบคุมกำลังสองเชิงเส้นและ ตัวควบคุม โมเดลพรีดิกทีฟ



ภาพที่ 31 เปรียบเทียบผลการจำลองระหว่างตัวควบคุมกำลังสองเชิงเส้นและตัวควบคุมโมเดลพรีดิกทีฟที่ไม่มีข้อจำกัด

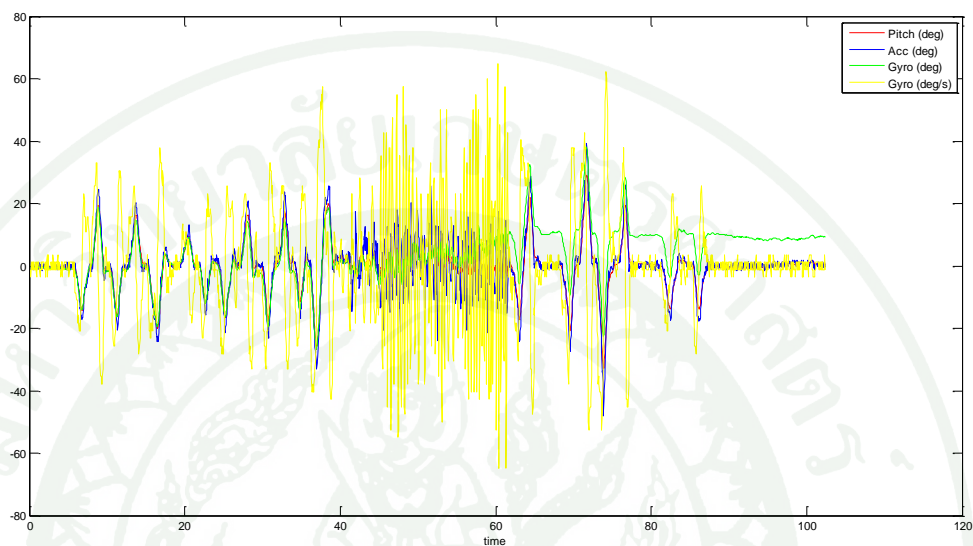


**ภาพที่ 32** เปรียบเทียบผลการจำลองระหว่างตัวคุมค่ากำลังสองเชิงเส้นและตัวควบคุมโมเดลพรีดิกทีฟที่มีข้อจำกัด ( $\pm 0.01$  Pitch)

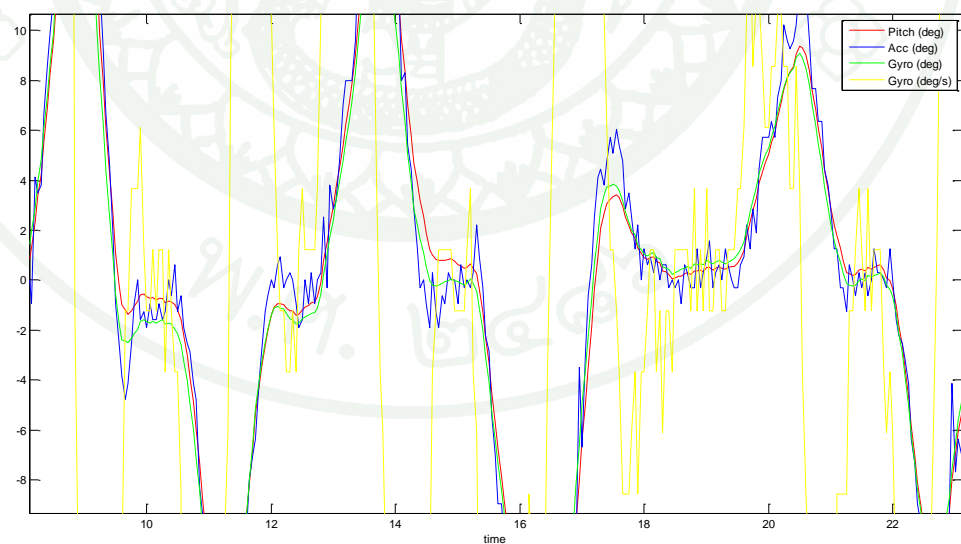
จากผลการจำลองภาพ 31 การติดตามตำแหน่งของตำแหน่ง (position) และการติดตามมุมการเลี้ยว (yaw) ตัวควบคุมโมเดลพรีดิกทีฟที่ติดตามได้รวดเร็วกว่าตัวคุมค่ากำลังสองเชิงเส้น ซึ่งเกิดค่าโอเวอร์ชูตเพียงเล็กน้อย แต่ในส่วนองมุมพิตช์ (pitch) นั้นตัวคุมค่ากำลังสองเชิงเส้นมีประสิทธิภาพที่ดีกว่า การกำหนดข้อจำกัด (Constraints) ให้กับตัวควบคุมโมเดลพรีดิกทีฟ ในส่วนองมุมพิตช์ (pitch) ที่  $\pm 0.01$  องศา ดังภาพที่ 32 ซึ่งเป็นการลดค่าความเคลื่อนไหวองมุมพิตช์ให้มีความใกล้เคียงกับตัวคุมค่ากำลังสองเชิงเส้น ซึ่งในส่วนองการติดตามตำแหน่งของตำแหน่ง (position) และการติดตามมุมการเลี้ยว (yaw) ตัวควบคุมโมเดลพรีดิกทีฟก็ยังสามารถติดตามได้รวดเร็วกว่าตัวคุมค่ากำลังสองเชิงเส้น

## ผลการทดลอง

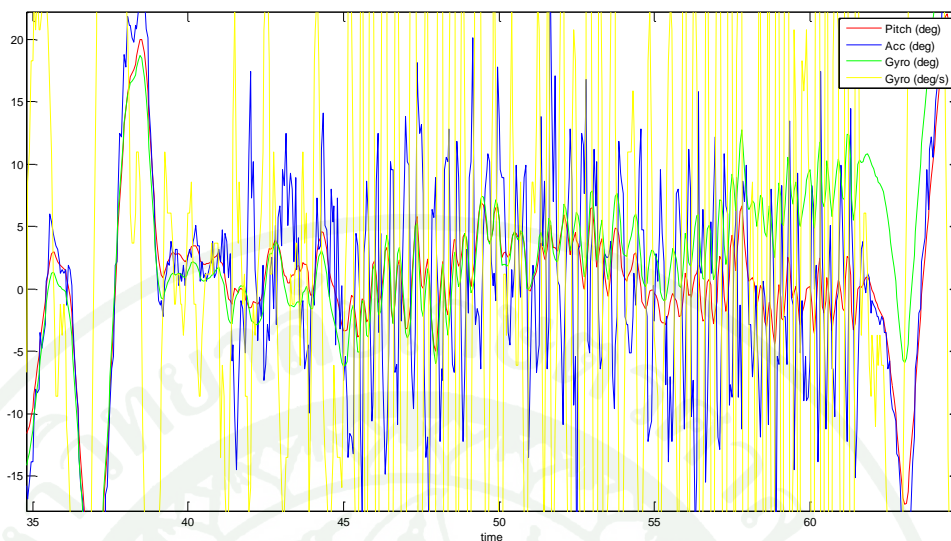
## 1. การวัดมุมเอียง (pitch) ของหุ่นยนต์สองล้อด้วยหลักการ Complementary Filter



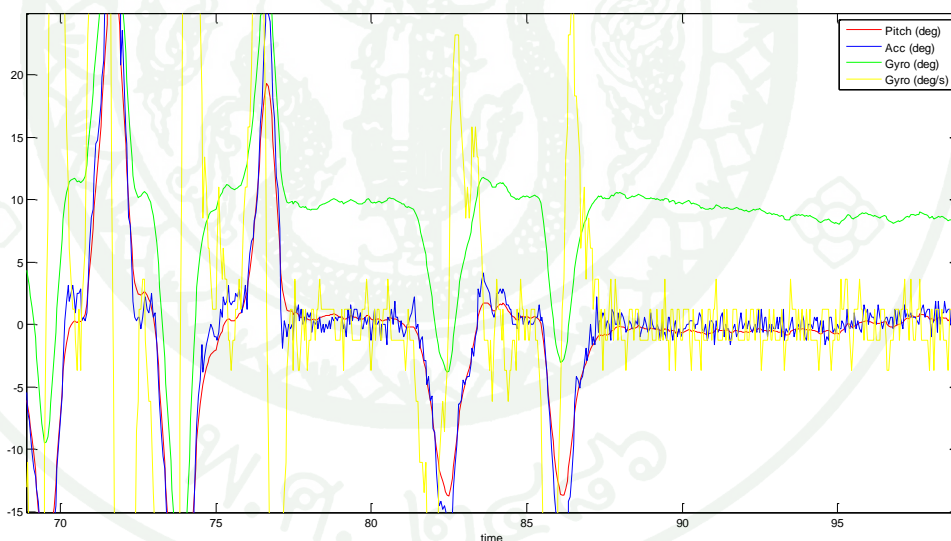
ภาพที่ 33 ภาพรวมของการวัดมุมเอียงด้วยหลักการ Complementary Filter



ภาพที่ 34 เปรียบเทียบผลระหว่างเซนเซอร์วัดความเร่งและ Complementary Filter



ภาพที่ 35 เปรียบเทียบผลของมุมจากเซนเซอร์ทั้งสองและ Complementary Filter ที่ความถี่สูง



ภาพที่ 36 เปรียบเทียบผลระหว่างเซนเซอร์วัดความเร็วเชิงมุมและ Complementary Filter

กำหนดค่าของ Sampling time 0.05 และ  $a = 0.96$  การกำหนดค่า  $a$  ได้อธิบายไว้ในส่วนของการออกแบบการควบคุม ในภาพที่ 34 ได้ขยับหุ่นยนต์ไปที่มุม  $\pm 20^\circ$  ที่เวลา 5 วินาทีถึง 35 วินาที ทำการเปรียบเทียบค่าของมุมที่วัดได้จากเซนเซอร์วัดความเร่งกับค่ามุมที่ผ่านตัวกรองความถี่ Complementary จะเห็นได้ว่าเส้นสีน้ำเงินจะมีการสั่นที่มากกว่าเส้นสีแดง ซึ่งในช่วงแรกค่ามุมที่วัด

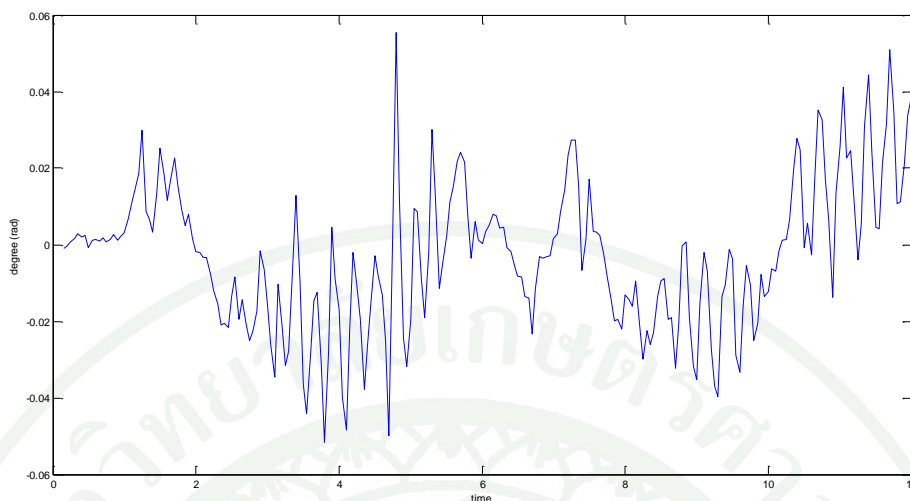
จากเซนเซอร์วัดความเร็วเชิงมุมในเส้นสีเขียว ได้ผลใกล้เคียงกับค่ามุมที่ผ่านตัวกรองความถี่ Complementary ในภาพที่ 35 ได้ทำการขยับหุ่นยนต์สองล้อให้สั้นด้วยความถี่สูงที่เวลา 40 วินาทีถึง 60 วินาที จะเห็นว่าค่าความเร็วเชิงมุมเส้นสีเหลืองและค่าของมุมที่วัดได้จากเซนเซอร์วัดความเร็วเชิงมุมในเส้นสีน้ำเงินมีการสั่นที่สูงมาก แต่ในส่วนของค่ามุมที่ผ่านตัวกรองความถี่ Complementary เส้นสีแดง มีการสั่นที่น้อยกว่ามาก ในภาพที่ 36 ค่าของมุมที่วัดจากเซนเซอร์วัดความเร็วเชิงมุมในเส้นสีเขียวที่เวลา 70 วินาที นั้น Drift ออกถึง 10 องศา ซึ่งในส่วนของเส้นสีน้ำเงินและสีแดงยังคงอยู่ที่ 0 องศา

## 2. การรักษาสมดุลของหุ่นยนต์สองล้อ

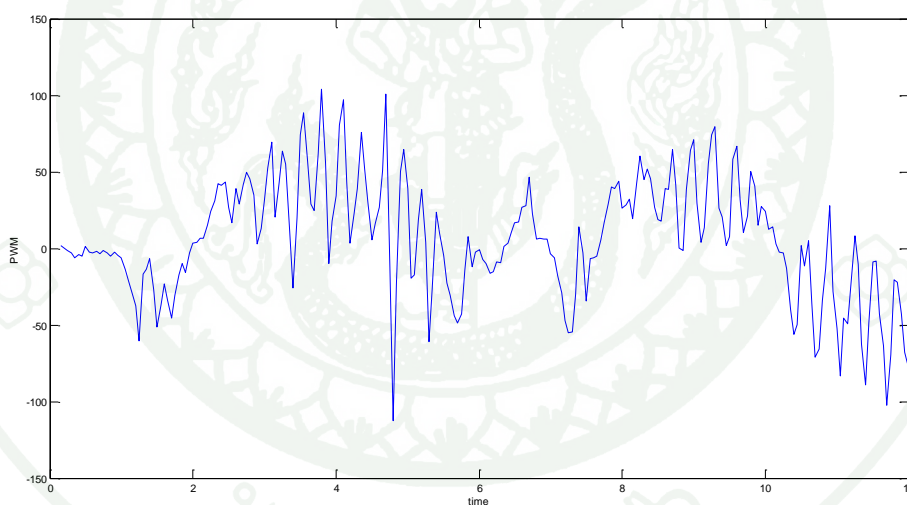
ในส่วนของผลการทดลองนี้ได้ใช้ support packages arduino mega 2560 ผ่านทางซอฟต์แวร์ Simulink ของโปรแกรม MATLAB ซึ่งไมโครคอนโทรลเลอร์สามารถวัดค่าตำแหน่งจากเอนโคเดอร์เซนเซอร์ได้โดยผ่าน S-Functions (ภาคผนวกที่ จ6) แต่เมื่อวัดตำแหน่งด้วยความเร็วที่สูงทำให้ไมโครคอนโทรลเลอร์นั้นวัดตำแหน่งไม่ทันและมีอาการค้าง ซึ่งอาจมีผลมาจากค่าความละเอียดของเอนโคเดอร์เซนเซอร์ที่มากหรืออาจเกิดจากการใช้ support packages ของโปรแกรม ซึ่งการใช้โปรแกรม IDE ของไมโครคอนโทรลเลอร์ arduino โดยตรงอาจจะวัดตำแหน่งที่ความเร็วสูงได้ทัน ดังนั้นงานวิจัยในอนาคตจะแก้ปัญหาในส่วนนี้ต่อไป

การทดลองนี้ได้ใช้ตัวคุมค่ากำลังสองเชิงในการกำหนดอัตราขยายแต่เนื่องจากไม่สามารถรับค่าจากเซนเซอร์เอนโคเดอร์ได้ทันจึงกำหนดให้อัตราขยายของมุมเฉลี่ยของล้อซ้ายและล้อขวา = 0 และอัตราขยายของมุมเลี้ยว = 0 และมีผลกระทบให้อัตราขยายของมุมพิทคลงไปมากเขียนได้ดังนี้คือ

$$K = \begin{bmatrix} 0 & -35.0 & 0 & -0.5 & 0 & 0 \\ 0 & -35.0 & 0 & -0.5 & 0 & 0 \end{bmatrix}$$



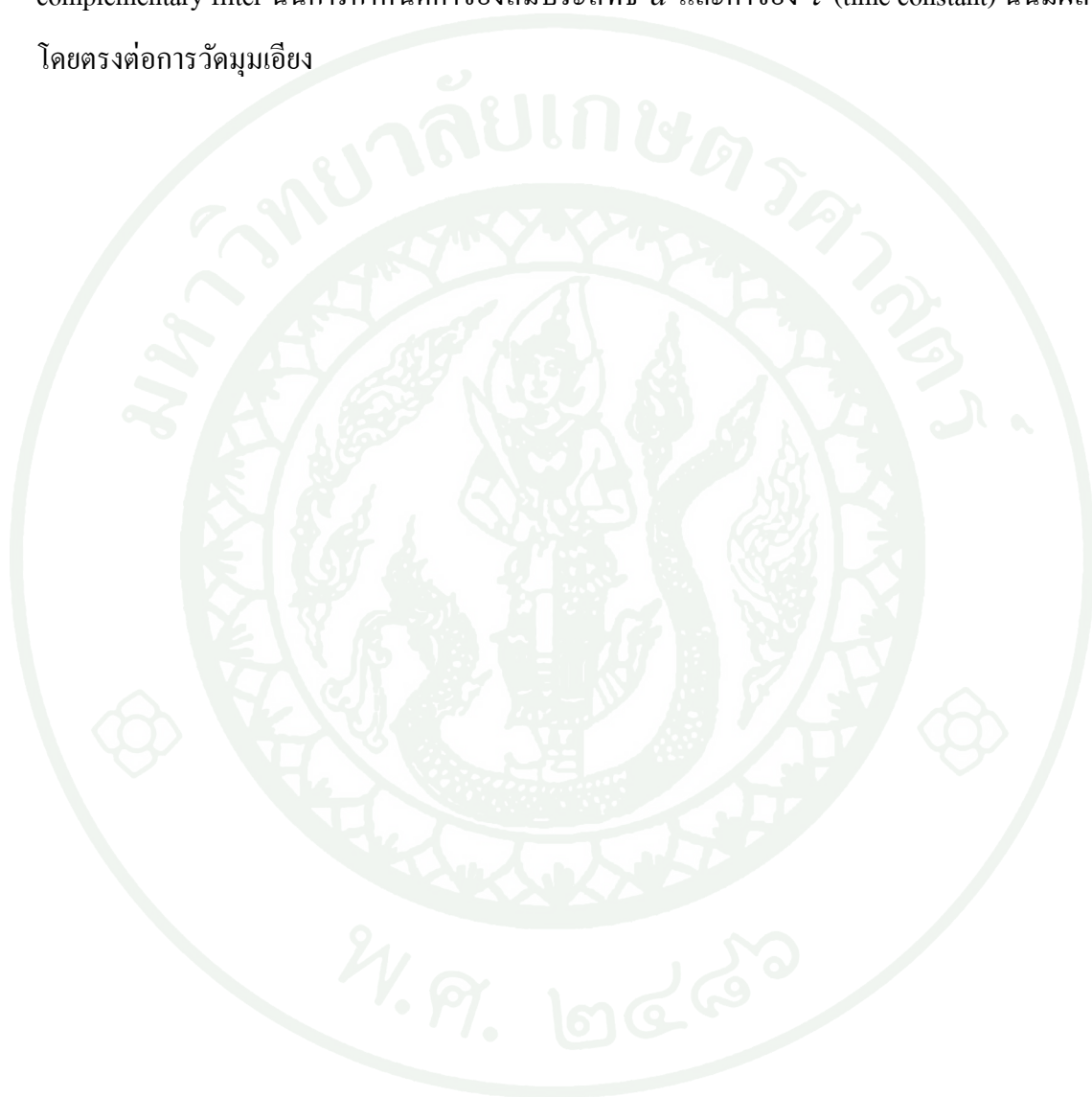
ภาพที่ 37 ค่าเอาต์พุตของมุม Pitch จากการรักษาสมดุลของหุ่นยนต์สองล้อ



ภาพที่ 38 สัญญาณควบคุม PWM จากการรักษาสมดุลของหุ่นยนต์สองล้อ

จากผลการทดลองในภาพที่ 37 จะเห็นได้ว่าค่าเอาต์พุตของมุม pitch นั้นรักษาเสถียรภาพที่มุม  $\pm 0.05$  (rad) อย่างไรก็ตามค่ามุมของมุม pitch นั้นค่อนข้างจะสั่นและมีการ drift ออกจากจุดศูนย์ เนื่องจากการวัดมุมเอียงที่ไม่แม่นยำ และการวัดมุมเอียงที่ไม่แม่นยำนั้นมีผลมาจากค่าของ  $dt$  (sample time) ที่ 0.05 ทำให้ต้องลดค่าสัมประสิทธิ์  $\alpha$  น้อยกว่า 0.98 มาก การกำหนดค่าน้อยกว่า

0.98 ก็จะทำให้ค่า noise จากเซนเซอร์วัดความเร่งเข้ามาได้มาก และถ้าทำการกำหนดค่าสัมประสิทธิ์  $a$  ให้เท่ากับ 0.98 ก็จะทำให้ค่ามุมที่วัดได้ drift ออกจากจุดศูนย์ ดังนั้นค่า sample time ควรน้อยกว่ากว่าหรือเท่ากับ 0.01 ก็จะทำให้ลดค่า noise และค่า drift ลงได้มาก ซึ่งโดยหลักการของ complementary filter นั้นการกำหนดค่าของสัมประสิทธิ์  $a$  และค่าของ  $\tau$  (time constant) นั้นมีผลโดยตรงต่อการวัดมุมเอียง



## สรุปและข้อเสนอแนะ

### สรุป

งานวิจัยนี้ได้ทำการรักษาสมมูลของหุ่นยนต์สองล้อด้วยตัวคุมค่ากำลังสองเชิงเส้น และตัวควบคุมโมเดลพรีดิกทีฟ ซึ่งได้แสดงถึงเสถียรภาพของตัวควบคุมทั้งสองดังผลการจำลอง การใช้ตัวคุมค่ากำลังสองเชิงเส้นนั้นทำให้สามารถหาอัตราขยายได้ง่ายเมื่อเปรียบเทียบกับตัวควบคุมอื่นๆ และยังมีเหมาะสมต่ออินพุตและเอาต์พุตของระบบ สามารถปรับค่าเมตริกซ์ถ่วงน้ำหนักให้ตรงกับการควบคุมที่ต้องการได้ จัดการปัญหาเรื่องระบบหลายอินพุตและเอาต์พุต การจัดสัญญาณรบกวน ความไว ความทนทานและอีกหลายปัญหาดังที่กล่าวข้างต้น ในส่วนของตัวควบคุมโมเดลพรีดิกทีฟสามารถจัดการกับปัญหาข้างต้นได้ดีเมื่อเปรียบเทียบกับตัวคุมค่ากำลังสองเชิงเส้นอีกทั้งยังมีความเร็วในการติดตามสัญญาณได้ดีกว่าในช่วงของ Transient และได้แสดงถึงคุณสมบัติพิเศษคือการกำหนดข้อจำกัดให้แก่เอาต์พุตของหุ่นยนต์สามารถนำไปประยุกต์ใช้ในงานอุตสาหกรรมที่ต้องการลดการใช้พลังงานได้

สรุปการวัดค่ามุมจากเซนเซอร์วัดความเร่งเพียงอย่างเดียวนั้นจะพบปัญหาในเรื่องของการสั่นที่มากเมื่อมีความถี่สูง และไม่สามารถบอกค่ามุมที่ต้องการได้ในขณะที่หุ่นยนต์สองล้อเคลื่อนที่ด้วยอัตราเร่งระดับหนึ่ง กล่าวคือไม่สามารถบอกความแตกต่างของมุมขณะที่หุ่นยนต์เคลื่อนที่และหยุดนิ่งได้ แต่จะวัดค่ามุมได้ดีในช่วงความถี่ต่ำหรือความเร็วต่ำๆ ส่วนการวัดมุมจากเซนเซอร์วัดความเร็วเชิงมุมเพียงอย่างเดียวก็จะพบปัญหาในเรื่องการ Drift ออกจากค่ามุมที่ต้องการ กล่าวคือเมื่อมีการเปลี่ยนแปลงที่ช้าก็จะทำให้ค่ามุม Drift ออกจากค่ามุมที่ต้องการมาก แต่สามารถวัดค่ามุมได้แม่นยำเมื่อความถี่สูงหรือความเร็วมาก ดังนั้นด้วยหลักการ Complementary Filter เป็นการนำข้อดีของเซนเซอร์ทั้งสองชนิดมาใช้งานร่วมกัน และจัดการข้อเสียของเซนเซอร์ทั้งสองด้วยตัวกรองความถี่สูงและตัวกรองความถี่ต่ำ

### ข้อเสนอแนะ

ตัวคุ่มค่ากำลังสองเชิงเส้นสามารถหาอัตราขยายได้ง่ายแต่การเลือกเมตริกซ์ถ่วงน้ำหนัก  $(Q, R)$  ยังต้องใช้วิธีการลองผิดลองถูกซึ่งในตอนนี้อาจทำให้เวลาค่อนข้างมาก ซึ่งในงานวิจัยทั่วไปนั้นจะให้เมตริกซ์ถ่วงน้ำหนักแก่เอาต์พุตมากกว่าและสำหรับหุ่นยนต์สองล้อมักจะให้น้ำหนักแก่ค่ามุม Pitch เพื่อรักษาเสถียรภาพของระบบ นอกจากนี้ความแม่นยำของตัวคุ่มค่ากำลังสองเชิงเส้นนี้ต้องอาศัยความแม่นยำของแบบจำลองทางคณิตศาสตร์ของหุ่นยนต์สองล้อ ซึ่งมีความสัมพันธ์กับการหาค่าพารามิเตอร์ของตัวหุ่นยนต์ที่แม่นยำเช่นกัน ดังนั้นค่าพารามิเตอร์บางตัวที่หาได้ค่อนข้างยากก็จะทำให้มีผลกระทบต่อความคลาดเคลื่อนของแบบจำลองทางคณิตศาสตร์

ในการกำหนดข้อจำกัดให้แก่ตัวควบคุม โมเดลพรีดิกทีฟเมื่อกำหนดให้ข้อจำกัดมีขนาดที่แคบเกินไปก็จะทำให้ไม่สามารถรักษาเสถียรภาพของระบบได้ ดังนั้นจำเป็นต้องมีการจำลองหรือทำการทดลองในการเลือกข้อจำกัดของค่าอินพุตและเอาต์พุตของระบบ การเลือกค่าพารามิเตอร์ที่สำคัญของตัวควบคุม โมเดลพรีดิกทีฟที่มีผลกระทบต่อเสถียรภาพของระบบ ดังนั้นจำเป็นต้องมีความเข้าใจว่าการเพิ่มค่าหรือการลดค่าของพารามิเตอร์ตัวใดแล้วจะส่งผลกระทบต่อระบบอย่างไร สำหรับการนำไปใช้งานจริงการเลือกค่าของ  $H_c$  นั้นส่งผลกระทบต่อการประมวลผลของไมโครคอนโทรลเลอร์ว่ามีความเร็วเพียงพอหรือไม่

การวัดมุมเอียงของหุ่นยนต์สองล้อด้วยการเขียนโค้ดโปรแกรมลงในซอฟต์แวร์ Simulink ของโปรแกรม MATLAB นั้นมีความเหมาะสมสำหรับการศึกษาและทำความเข้าใจในตัวควบคุมทั้งสองชนิด แต่อาจทำให้การประมวลผลของไมโครคอนโทรลเลอร์ช้าลงหรือเกิด Bug ที่ไม่พึงประสงค์ ดังนั้นหากผู้วิจัยมีความรู้ความเข้าใจทางด้านภาษา C++ ในการเขียนโค้ดสำหรับ Arduino โดยตรงก็จะทำให้ลดปัญหาของการประมวลผลและ Bug ได้ การใช้โปรแกรม LabVIEW และใช้ Hardware ของ NI ก็อาจเป็นอีกทางเลือกหนึ่งที่ดีแต่ต้องแลกกับราคาที่ค่อนข้างแพงกว่า ไมโครคอนโทรลเลอร์ (Arduino) หลายเท่าตัว

## เอกสารและสิ่งอ้างอิง

- กิตติพงษ์ เยาวาจา. 2551. การควบคุมที่เหมาะสมที่สุดของหุ่นยนต์สองล้อด้วยการควบคุมเชิง  
แบบดัดวิธีสูง.มหาวิทยาลัยเกษตรศาสตร์, กรุงเทพฯ.
- วโรตม ตูจินดา. 2550. การวิเคราะห์และออกแบบระบบควบคุมทางวิศวกรรม.  
มหาวิทยาลัยเกษตรศาสตร์, กรุงเทพมหานคร
- วิกิพีเดีย สารานุกรมเสรี. 2556. หมวดหมู่เทคโนโลยีและหุ่นยนต์. แหล่งที่มา:  
th.wikipedia.org/wiki/หุ่นยนต์, 2 กันยายน 2557.
- Brkic, K. and Z. Kovacic. 2009. Decoupled Control and Path Tracking of a Two-wheeled  
Self-balancing Mobile Robot. **IEEE International Symposium on Industrial  
Electronics (ISIE 2009)**, 5-8 July 2009, Seoul Olympic Parktel, Seoul, Korea.
- Chai, S., L.Wang and E. Rogers. 2013. Model predictive control of a permanent magnet  
synchronous motor with experimental validation. **Control Engineering Practice**21  
(2013) pp. 1584–1593
- Cheng, P. and B. Oelmann. 2010. Joint-Angle Measurement Using Accelerometers and  
Gyroscopes—A Survey. **IEEE Transactions on Instrumentation and Measurement**,  
VOL. 59, NO. 2, FEBRUARY 2010.
- Colton, S. 2007. The Balance Filter. **Submitted as a Chief Delphi white paper**. 25 June 2007.
- Deng, M., A. Inoue., K. Sekiguchi and L.Jiang. 2010. Two-wheeled mobile robot motion control  
in dynamic environments. **Robotics and Computer-Integrated Manufacturing**. No. 26  
(2010) pp. 268–272

- Ekaputri, C. and A. Rohman. 2012. Implementation Model Predictive Control (MPC) Algorithm-3 for Inverted Pendulum. **IEEE Control and System Graduate Research Colloquium (ICSGRC 2012).**
- Ferdinando H., H. Khoswanto and S. Tjokro. 2011. Design and Evaluation of Two-wheeled Balancing Robot Chassis. **International Symposium on Innovations in Intelligent Systems and Applications (INISTA).** 15-18 June 2011.
- Feng, T., T. Liu and X. Wang. 2011. Modeling and Implementation of Two-wheel Self-balancing Robot Equipped With Supporting Arms. **IEEE Conference on Industrial Electronics and Applications (ICIEA).** No. 6, 2011.
- Ge, S. S. and Y. J. Cui. 2000. New Potential Functions for Mobile Robot Path Planning. **IEEE Transactions on Robotics and Automation,** VOL. 16, No. 5, OCTOBER 2000.
- Grasser, F., A. D. Arrigo, S. Colombi and A. Ruffer. 2002. JOE: A Mobile, Inverted Pendulum. **IEEE Transactions on Industrial Electronics,** 49(1).
- Kim, Y. H., S. H. Kim and Y. K. Kwak. 2005. Dynamic Analysis of a Nonholonomic Two-wheeled Inverted Pendulum Robot, pp. 505-513. **Proceedings of the 8th on Artificial Life and Robotics,** 21(3).
- Lee, J., S. Han and J. Lee. 2013. Decoupled Dynamic Control for Pitch and Roll Axes of the Unicycle Robot. **IEEE Transactions on Industrial Electronics,** VOL. 60, NO.9, September 2013.
- Lin, C. and Y. Liu. 2012. Precision Tracking Control and Constraint Handling of Mechatronic Servo Systems Using Model Predictive Control. **IEEE/ASME Transactions on Mechatronics,** VOL 17, NO.4, AUGUST 2012.

Maciejowski, J.M. 2002. Predictive Control with Constraints. **Published by Pearson Education under The Prentice Hall imprint.** June 2001.

Ogata, K. 1997. **Modern Control Engineering.** 3rd Edition. Prentice-Hall. New York.

Oryschuk, P., A. Salerno., A. M. Al-Husseini and J. Angeles. 2009. Experimental Validation of an Underactuated Two-Wheeled Mobile Robot. **IEEE/ASME TRANSACTIONS ON MECHATRONICS, VOL. 14, NO. 2, APRIL 2009.**

Siciliano, B. and O. Khatib. 2008. **Springer Handbook of Robotics.** 2008 edition. (pp.391-409) Publisher Springer.

Urakubo, T., K. Okuma and Y. Tada. 2004. Feedback Control of a Two Wheeled Mobile Robot with Obstacle Avoidance Using Potential Functions. **IEEE/RSJ International Conference on Intelligent Robots and Systems** 28 September - 2 October 2004, Sendai, Japan.

Wang, L. 2009. Model Predictive Control System Design and Implementation Using MATLAB. **Advances in Industrial Control ISSN 1430-9491.** Springer-Verlag London Limited, 2009.

Wasif, A. and D. Raza., *et al.* 2013. Design and Implementation of a Two wheel Self Balancing Robot with a Two Level Adaptive Control. **IEEE International Conference on Control and Automation (ICCA), 2013.**

Yap, H. and S. Hashimoto. 2012. Development of a stair traversing two wheeled robot. **IEEE/RSJ International Conference on Intelligent Robots and Systems,** 7-12 October, 2012. Vilamoura, Algarve, Portugal.

Yomamoto, Y. 2008. **NXTway-GS (Self-Balancing Two-Wheeled Robot) Controller Design.**  
<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=19147>

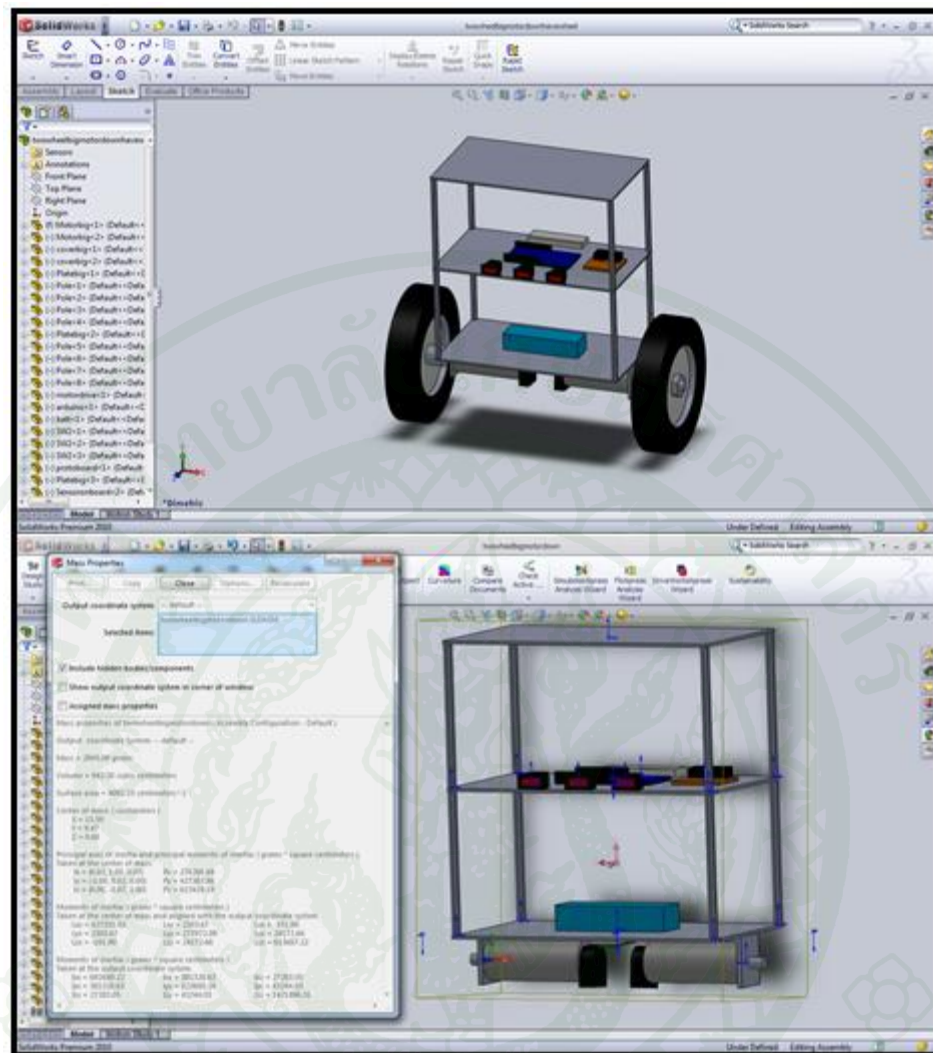
Zeng, Z. and H. Lu., and Z. Zhiqiang. 2013. High-speed Trajectory Tracking Based on Model Predictive Control for omni-directional Mobile Robots. **Control and Decision Conference (CCDC)**, No. 25, 2013. Chinese.







ภาคผนวก ก  
การหาค่าพารามิเตอร์ด้วยโปรแกรม Solidworks



ภาพผนวกที่ ก1 การหาค่าพารามิเตอร์ด้วยโปรแกรม Solidworks

จากการสร้างแบบจำลองของหุ่นยนต์ได้ค่าพารามิเตอร์ที่สำคัญคือ

$L$  ระยะจากแกนล้อถึงศูนย์กลาง  $0.1359 (m)$



ภาคผนวก ข  
การหาแบบจำลองทางคณิตศาสตร์ของหุ่นยนต์สองล้อและอัตราขยายของตัวคุมค่ากำลังสองเชิง  
เส้นด้วยโค้ดโปรแกรม MATLAB

## ค่าพารามิเตอร์ของหุ่นยนต์สองล้อ

%parameters of two wheel robot

$g = 9.81;$  %Gravity acceleration (m/sec<sup>2</sup>)  
 $m = 0.990;$  %Wheel weight (kg)  
 $R = 0.1;$  %Wheel radius (m)  
 $J_w = (m \cdot R^2)/2;$  %Wheel inertia moment (kgm<sup>2</sup>)  
 $M = 3.1;$  %Body weight (kg)  
 $W = 0.36;$  %Body width (m)  
 $D = 0.155;$  %Body depth (m)  
 $H = 0.339 ;$  %Body height (m)  
 $L = 0.1359 ;$  %Distance of the center of mass from the wheel axle (m)  
 $J_p = (M \cdot L^2)/3;$  %Body pitch inertia moment (kgm<sup>2</sup>)  
 $J_y = M \cdot (W^2 + D^2)/12;$  %Body yaw inertia moment (kgm<sup>2</sup>)  
 $J_m = 1 \cdot 10^{-5};$  %DC motor inertia moment  
 $R_m = 3.4;$  %DC motor resistance  
 $K_b = 0.468;$  %DC motor back EMF constant  
 $K_t = 0.317 ;$  %DC motor torque constant  
 $n = 1 ;$  %Gear ratio  
 $f_m = 0.0022;$  %Friction coefficient between body and DC motor  
 $f_w = 0 ;$  %Friction coefficient between wheel and floor

%We use the values described in reference [2] for  $R_m, K_b, K_t$

%We use the values that seems to be appropriate for  $J_m, n, f_m, f_w$ , because it is difficult to measure

## การหาแบบจำลองทางคณิตศาสตร์

$$\text{Beta} = ((n \cdot K_t \cdot K_b) / R_m) + f_m;$$

$$\text{Lfa} = (n \cdot K_t) / R_m;$$

$$E = [((2*m+M)*R^2+(2*Jw)+(2*n^2*Jm)) (M*L*R-2*n^2*Jm);$$

$$(M*L*R-(2*n^2*Jm)) (M*L^2+Jp+(2*n^2*Jm))];$$

$$E11 = ((2*m+M)*R^2+(2*Jw)+(2*n^2*Jm));$$

$$E12 = (M*L*R-2*n^2*Jm);$$

$$E21 = (M*L*R-(2*n^2*Jm));$$

$$E22 = (M*L^2+Jp+(2*n^2*Jm));$$

$$F = [(Beta+fw) -Beta;$$

$$-2*Beta 2*Beta];$$

$$G = [0 0;$$

$$0 -M*g*L];$$

$$H = [Lfa/2 Lfa/2;$$

$$-Lfa -Lfa];$$

$$I = [Beta + (W/R)*fw];$$

$$J = [(R/W)*Lfa];$$

$$K = [1/2*m*W^2 + Jp + (W^2/(2*R^2))*(Jw + n^2*Jm)];$$

$$A32 = [(-M*g*L*E12)/det(E)];$$

$$A42 = [(M*g*L*E11)/det(E)];$$

$$A33 = [-((Beta+fw)*E22 + 2*Beta*E12)/det(E)];$$

$$A43 = [((Beta+fw)*E12 + 2*Beta*E11)/det(E)];$$

$$A34 = [Beta*(E22 + 2*E12)/det(E)];$$

$$A44 = [-Beta*(E12 + 2*E11)/det(E)];$$

$$B3 = [Lfa*((E22/2) + E12)/det(E)];$$

$$B4 = [-Lfa*((E12/2) + E11)/det(E)];$$

$$A1 = [0 0 1 0;$$

$$0 0 0 1;$$

$$0 A32 A33 A34;$$

$$0 A42 A43 A44];$$

```

B1 = [0 0;
      0 0;
      B3 B3;
      B4 B4];
A2 = [0 1;
      0 -1/K];
B2 = [0 0;
      -J/K J/K];

A_bar = [A1, zeros(4, 2);
         zeros(2,4), A2]
B_bar = [B1;B2]
C_bar = [ 1 0 0 0 0 0;
         0 1 0 0 0 0;
         0 0 0 0 1 0]
D_bar = [zeros(3,2)]
Q_bar = [1 0 0 0 0 0;
         0 1 0 0 0 0;
         0 0 1 0 0 0;
         0 0 0 1 0 0;
         0 0 0 0 1 0;
         0 0 0 0 0 1]
R_bar = [1 0;
         0 1]

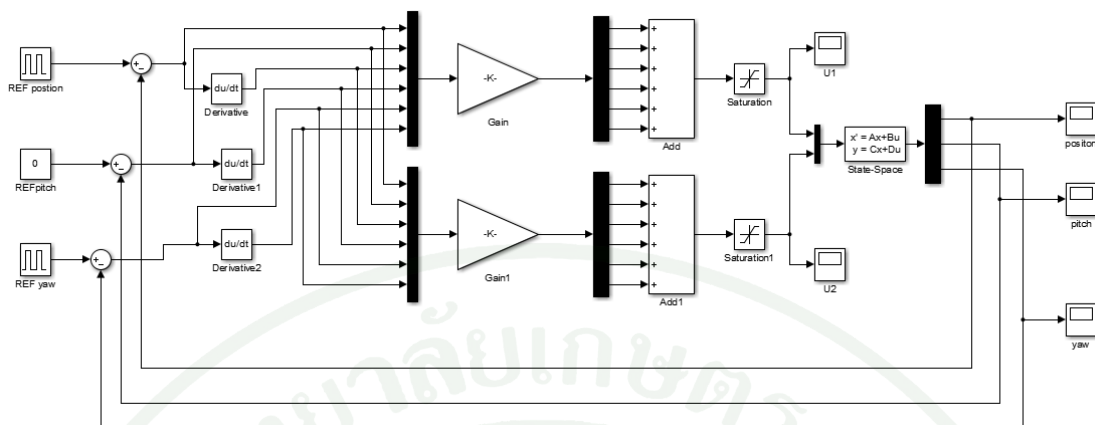
twowheel = ss(A_bar,B_bar,C_bar,D_bar);

```

อัตราขยายของตัวคุมค่ากำลังสองเชิงเส้น

```
K = lqr(A_bar,B_bar,Q_bar,R_bar)
```





ภาพผนวกที่ ค1 การจำลองผลการทดลองด้วยตัวคุมค่ากำลังสองเชิงเส้น

กำหนดค่าอัตราขยายและ plant ดังนี้

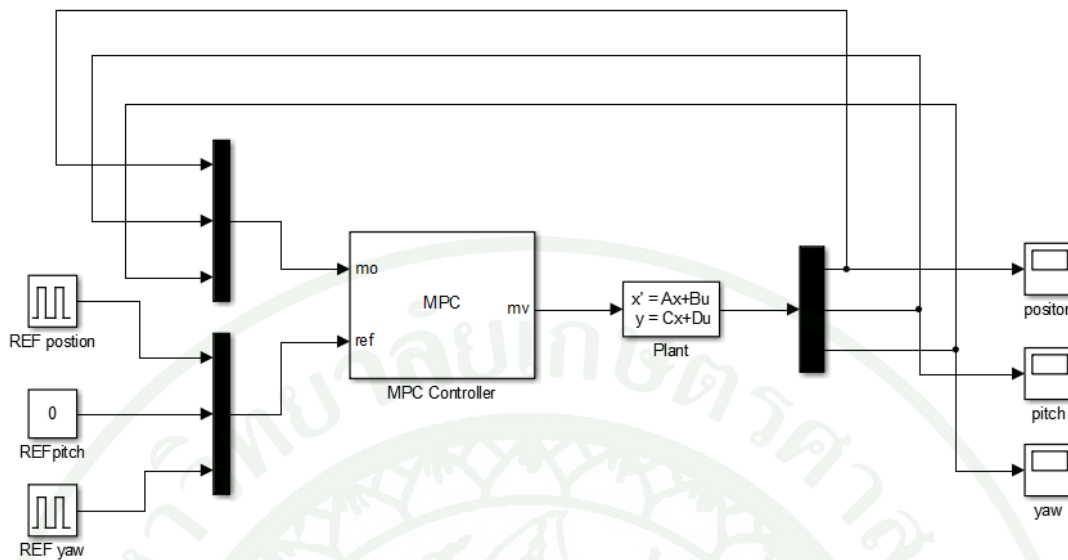
$$K = \begin{bmatrix} -0.7071 & -72.3187 & -7.8973 & -1.1923 & -0.7071 & -1.6314 \\ -0.7071 & -72.3187 & -7.8973 & -1.1923 & 0.7071 & 1.6314 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -60.7802 & -1.3670 & 1.3670 & 0 & 0 \\ 0 & 87.6433 & 1.3923 & -1.3923 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -0.2113 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1.3287 & 1.3287 \\ -1.3533 & -1.3533 \\ 0 & 0 \\ -0.1141 & 0.1141 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$



ภาคผนวก ง  
การหาผลการจำลองด้วยตัวควบคุมโมเดลพรีดิกทีฟ



ภาพผนวกที่ 1 การหาผลการจำลองด้วยตัวควบคุมโมเดลพรีดิกทีฟ

กำหนดค่าพารามิเตอร์ และ plant ดังนี้

Control Interval Time = 0.1, Prediction Horizon ( $H_p$ ) = 10, Control Horizon ( $H_c$ ) = 2

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -60.7802 & -1.3670 & 1.3670 & 0 & 0 \\ 0 & 87.6433 & 1.3923 & -1.3923 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -0.2113 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1.3287 & 1.3287 \\ -1.3533 & -1.3533 \\ 0 & 0 \\ -0.1141 & 0.1141 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The image shows two screenshots of the MPC Design Task software interface. The top screenshot displays the 'Model and Horizons' tab, where the 'Plant model' is set to 'twowheel'. The 'Horizons' section includes input fields for 'Control interval (time units)' (0.1), 'Prediction horizon (intervals)' (10), and 'Control horizon (intervals)' (2). There is a 'Blocking' checkbox and a 'Blocking allocation within prediction horizon' dropdown set to 'Beginning'. The 'Number of moves computed per step' is 3, and the 'Custom move allocation vector' is [2 3 5]. A 'Help' button is visible at the bottom.

The bottom screenshot shows the 'Constraints' tab. It contains two tables: 'Constraints on manipulated variables' and 'Constraints on output variables'.

Name	Units	Minimum	Maximum	Max Down Rate	Max Up Rate
In1		-5	5		
In2		-5	5		

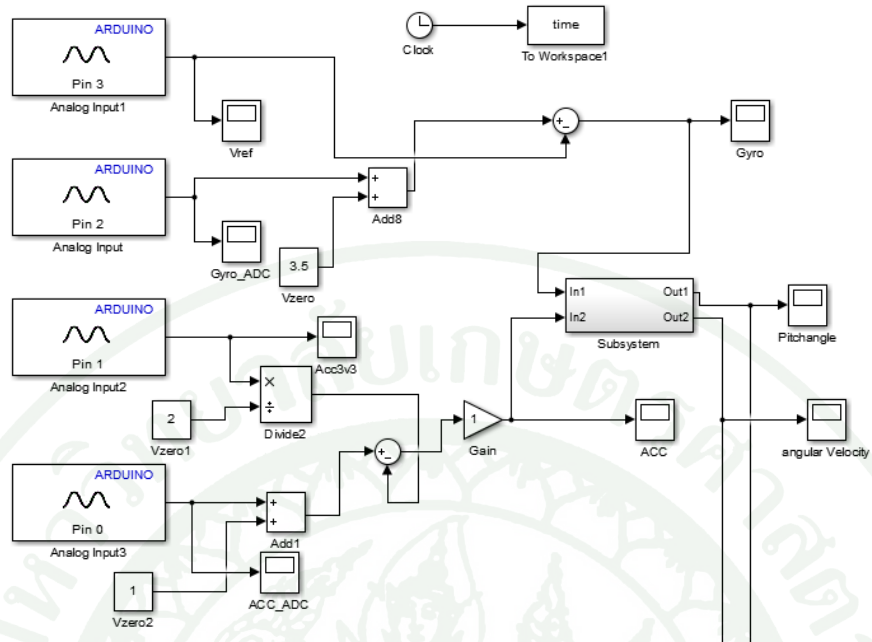
  

Name	Units	Minimum	Maximum
Out1			
Out2			
Out3			

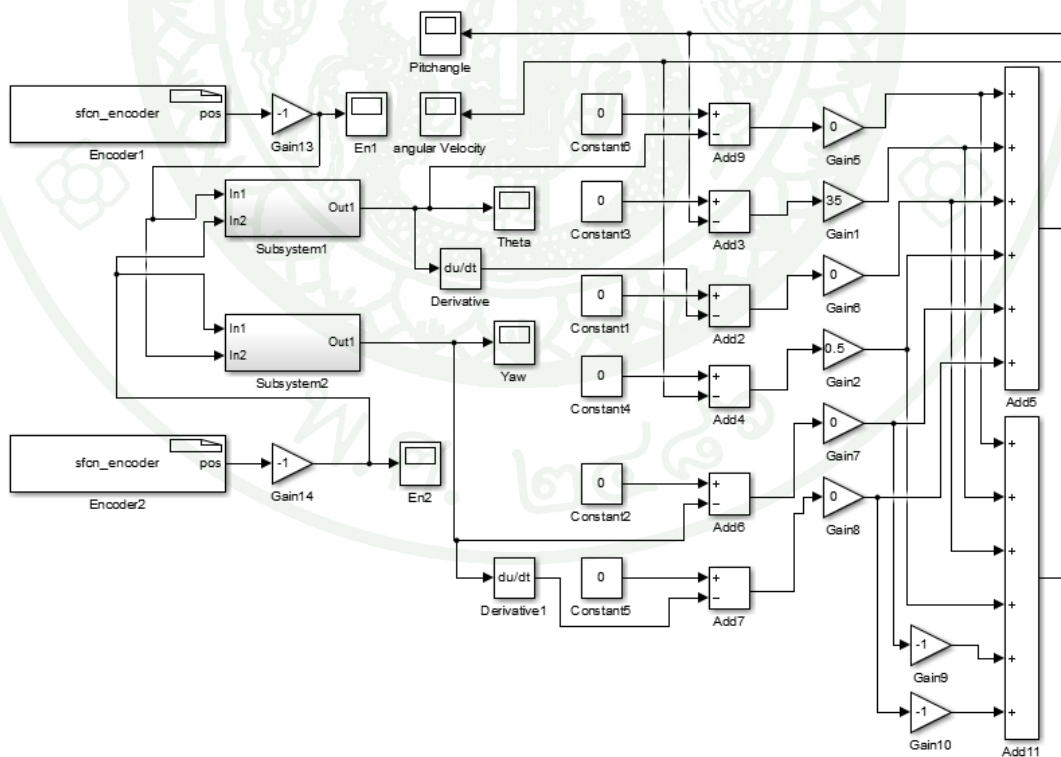
At the bottom of the constraints section, there are 'Constraint Softening' and 'Help' buttons.

ภาพผนวกที่ ๖2 การกำหนดค่าพารามิเตอร์ของตัวควบคุมโมเดลพรีดิกทีฟ

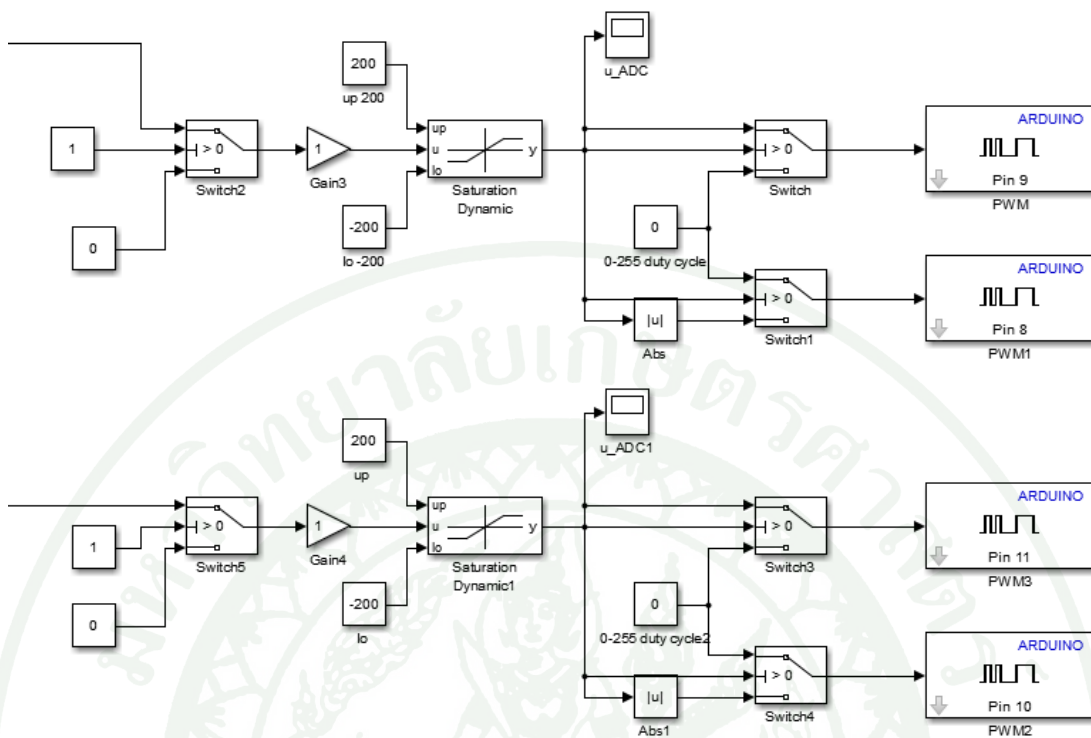




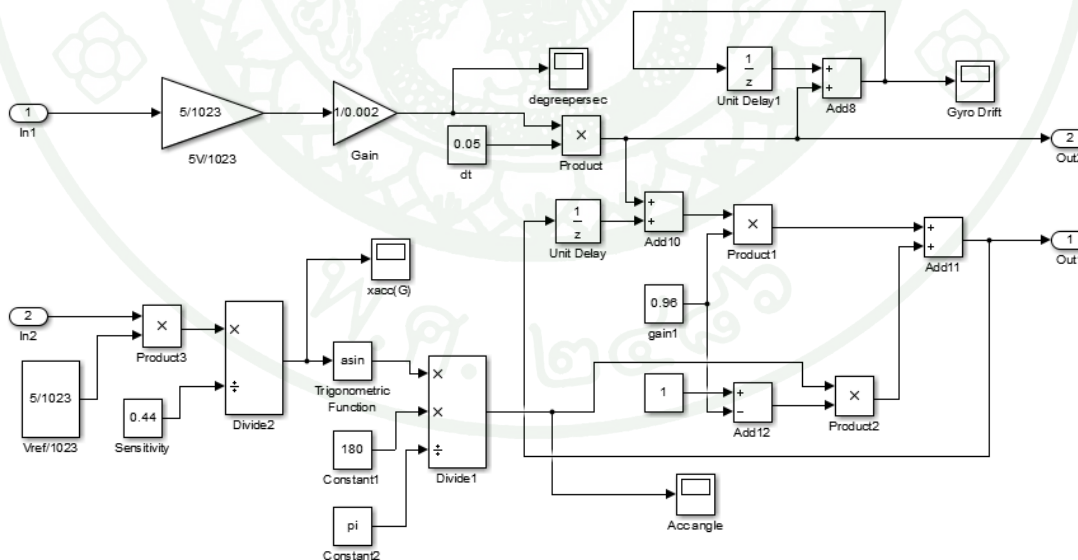
ภาพผนวกที่ จ1 การรับค่าจากเซนเซอร์วัดความเร่งและเซนเซอร์วัดความเร็วเชิงมุม



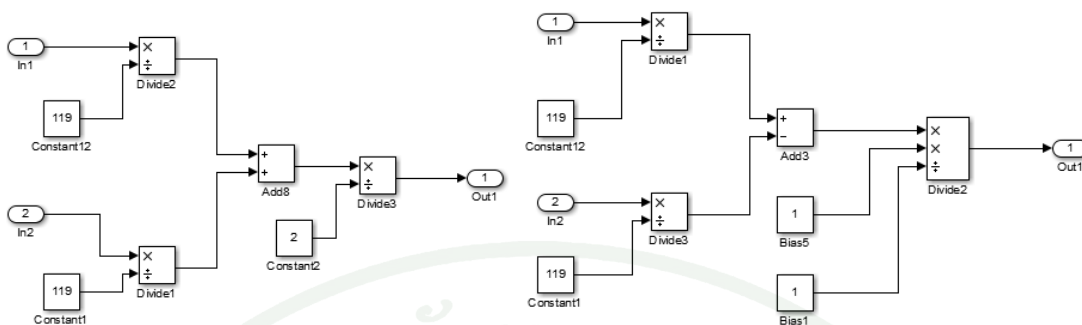
ภาพผนวกที่ จ2 การรับค่าจากเอนโค้ดเดอร์เซนเซอร์ด้วย S-function และอัตราขยาย



ภาพผนวกที่ จ3 การควบคุมมอเตอร์ช้าและมอเตอร์เร็วด้วย PWM



ภาพผนวกที่ จ4 การวัดมุมเอียงด้วยหลักการ complementary filter (Subsystem)



ภาพผนวกที่ ๖5 การวัดตำแหน่งและมุมเฉลี่ยของเอนโคเดอร์เซนเซอร์ (Subsystem1 และ Subsystem2)

Parameters

S-function name:  Build

S-function parameters

Name	Data type	Value
enc	uint8	1
pinA	uint8	18
pinB	uint8	19

Port/Parameter

- Input Ports
- Output Ports
- pos
- Parameters
  - enc
  - pinA
  - pinB

Initialization | Data Properties | Libraries | **Outputs** | Continuous Derivatives | Discrete Update | Build Info

Enter any library/object or source files used by the S-function. Then, specify any necessary include files or enter the external function declarations. These functions can be called in the Outputs, Derivatives and Update methods.

Library/Object/Source files (one per line)

Include files and external function declarations

Includes:

```
# ifndef MATLAB_MEX_FILE
# include <Arduino.h>

typedef struct { int pinA; int pinB; int pinC; } Encoder;
volatile Encoder Enc[3] = {{0,0,0,0}, {0,0,0,0}, {0,0,0,0}};

/* auxiliary function to handle encoder */
void func(double a);

External function declarations:
/* extern double func(double a); */
```

ภาพผนวกที่ ๖6 การใช้ S-function ในการวัดค่าจากเอนโคเดอร์เซนเซอร์

## ประวัติการศึกษาและการทำงาน

ชื่อ – นามสกุล	นายพงศกร กวีสุนทรเสนาะ
วัน เดือน ปี ที่เกิด	วันที่ 10 สิงหาคม 2532
สถานที่เกิด	จังหวัดกรุงเทพมหานคร
ประวัติการศึกษา	ปริญญาวิศวกรรมศาสตรบัณฑิต (วิศวกรรมขนถ่ายวัสดุ) มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ พ.ศ. 2553
ตำแหน่งหน้าที่การงานปัจจุบัน	-
สถานที่ทำงานปัจจุบัน	-
ผลงานดีเด่นและรางวัลทางวิชาการ	-
ทุนการศึกษาที่ได้รับ	-