



## รายงานการวิจัยฉบับสมบูรณ์

การออกแบบและสร้างหุ่นยนต์รถสองล้อ

**Design and Construct of Two-Wheel Mobile Robot**

ธัญช      ศรีพนม      หัวหน้าโครงการ  
มหนูศักดิ์      จานทอง      ผู้ร่วมวิจัย

คณะครุศาสตร์อุตสาหกรรม  
มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี  
งบประมาณประจำปี 2552

# มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

---

รายงานโครงการวิจัย

การออกแบบและสร้างหุ่นยนต์  
รถสองล้อ

**Design and Construct of Two-Wheel  
Mobile Robot**

ธวัช ศรีพนม

มนุศักดิ์ จานทอง

## ส่วนที่ 1 รายละเอียดโครงการ

### 1. ชื่อโครงการวิจัย การออกแบบและสร้างหุ่นยนต์รถสองล้อ

(Design and Construct of Two-Wheel Mobile Robot)

### 2. หน่วยงานหลักที่รับผิดชอบงานวิจัย และสถานที่ตั้งพร้อมทั้งชื่อหน่วยงาน และลักษณะของการร่วมงานวิจัยกับหน่วยงานอื่น (ถ้ามี)

#### หน่วยงานหลักที่รับผิดชอบ

สาขาวิชาครุศาสตร์เครื่องกล คณะครุศาสตร์อุตสาหกรรม มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี ต. คลองหก อ.ธัญบุรี จ.ปทุมธานี 12110 โทร. 02-549-4744-5

### 3. คณะผู้วิจัย บทบาทของนักวิจัยแต่ละคนในการทำวิจัย และสัดส่วนที่ทำการวิจัย (%)

ลำดับ	ชื่อ-สกุล	บทบาทนักวิจัย	สัดส่วนการทำวิจัย
1	นายธนัช ศรีพนม Mr. Tanut Sripanom	หัวหน้าโครงการและ ผู้วิจัยหลัก	50%
2	นายมนูศักดิ์ जानทอง Mr. Manusak Janthong	ผู้ร่วมวิจัย	50%

### 4. ประเภทของการวิจัย การพัฒนาทดลอง

### 5. สาขาวิชาการและกลุ่มวิชาที่ทำการวิจัย สาขาวิศวกรรมศาสตร์และอุตสาหกรรมวิจัย

## ส่วนที่ 2 เนื้อหาโครงการ

# บทคัดย่อ

โครงการวิจัยนี้เป็นการออกแบบโครงสร้างหุ่นยนต์รถสองล้อและตัวควบคุมหุ่นยนต์รถสองล้อให้สามารถทรงตัวอยู่ได้ โดยที่หุ่นยนต์ที่ออกแบบนี้เป็นหุ่นยนต์ต้นแบบซึ่งโครงสร้างที่ออกแบบจะเป็นโครงสร้างแบบอย่างง่าย ตัวถังของหุ่นยนต์เป็นรูปทรงสี่เหลี่ยมขนาด 45x40x30 ซม. ภายในตัวถังประกอบไปด้วย 1) ดีซีโซเวอร์มอเตอร์ขนาด 200 วัตต์และมีเกียร์ทดรอบขนาดอัตราทด 1:21 2) แผงวงจรขับเคลื่อนมอเตอร์ 3) ตัวประมวลผลที่ใช้ในการควบคุมการเคลื่อนที่และทรงตัวของหุ่นยนต์ 4) Accelerometer และ Gyroscope เป็นเซนเซอร์ช่วยในการวัดมุมเอียงของหุ่นยนต์เทียบกับพื้นโลก สำหรับตัวควบคุมการทรงตัวของหุ่นยนต์ได้ใช้หลักการการควบคุมแบบ Cascade Controller ชนิดพีไอดี (PID Controller) ในการทดลองควบคุมหุ่นยนต์รถสองล้อได้ใช้ Kalman filter เข้ามาช่วยในการประมาณค่ามุมเอียงที่ได้จาก Accelerometer และ Gyroscope จากการทดลองหุ่นยนต์สามารถทรงตัวอยู่ได้ตลอดช่วงเวลาที่กำหนด และยังสามารถทรงตัวอยู่ได้เมื่อมีการเอียงหุ่นยนต์ไปด้านหน้า หุ่นยนต์ก็จะเคลื่อนที่ไปด้านหน้าและถ้ามีการเอียงหุ่นยนต์ไปด้านหลัง หุ่นยนต์ก็จะเคลื่อนที่ไปด้านหลัง

# กิตติกรรมประกาศ

คณะผู้วิจัยขอขอบพระคุณมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรีและสำนักงานวิจัยแห่งชาติ (วช.) อย่างสูงที่ได้มอบทุนวิจัยสำหรับโครงการวิจัยนี้ และขอขอบพระคุณภาคีชาววิศวกรรมเครื่องกลที่อำนวยความสะดวกพร้อมทั้งสนับสนุนในด้านต่างๆ เพื่อดำเนินการโครงการวิจัย และท้ายที่สุดขอกล่าวคำขอบพระคุณไปยังคณาจารย์ทุกๆ ท่านที่ได้ให้คำแนะนำที่มีประโยชน์ทุกๆ ด้านแก่คณะผู้วิจัย

คณะผู้วิจัย

# สารบัญ

บทคัดย่อ.....	iii
กิตติกรรมประกาศ.....	iv
สารบัญ.....	v
สารบัญรูปภาพ.....	vi
สารบัญตาราง.....	ix
1. บทนำ.....	1
2. ทฤษฎีที่เกี่ยวข้อง.....	4
3. โครงสร้างของระบบ.....	14
4. แบบจำลองของระบบ.....	19
5. การทดลองและผลลัพธ์.....	27
6. บทสรุป.....	38
บรรณานุกรม.....	40
ภาคผนวก ก โค้ดโปรแกรมที่ใช้ในการควบคุม.....	42

# สารบัญรูปภาพ

2.1	Segway .....	5
2.2	ระบบ inverted pendulum-cart .....	5
2.3	YAIP – the inverted pendulum robot .....	6
2.4	แผนผังการประมาณมุมเอียง Attitude โดยใช้หลักการ Complementary filters .....	6
2.5	Gyrover .....	6
2.6	UniBot .....	7
2.7	Two Wheel Balancing Robot ของอดิศักดิ์ คงฉางและคณะ .....	7
2.8	ดัดเบิลยูแอล-16 อาร์ที .....	8
2.9	หุ่นยนต์สองล้อที่ออกแบบคล้ายรถเข็นคนป่วย .....	8
2.10	ระบบควบคุมแบบปิด .....	10
2.11	Diagram ระบบควบคุมแบบปิด .....	10
2.12	โครงสร้างของ State observer .....	12
2.13	ระบบควบคุมแบบป้อนกลับโดยใช้ตัวควบคุมแบบ PID .....	13
3.1	อุปกรณ์ภายในกล่องควบคุม .....	14
3.2	ตัวควบคุม CompactRIO และชุดขับเคลื่อนมอเตอร์ .....	15
3.3	Real-Time controller รุ่น NI cRIO-9012 .....	15
3.4	Reconfiguration chassis รุ่น NI cRIO-9102 .....	16
3.5	Analog Output Module รุ่น NI 9263 Module .....	16
3.6	Analog Input Module รุ่น NI 9205 Module .....	16

3.7	Digital I/O Module รุ่น NI 9401 Module .....	17
3.8	แผงวงจรเซนเซอร์ Accelerometer กับ Gyroscope .....	17
3.9	ดีซีเซอร์โวมอเตอร์พร้อมเกียร์ทด .....	18
3.10	ชุดขับดีซีมอเตอร์ .....	18
4.1	โมเดลของหุ่นยนต์สองล้อ .....	20
4.2	ด้านข้างและด้านบนของหุ่นยนต์สองล้อ .....	20
5.1	การทดสอบ Accelerometer และ Gyroscope .....	28
5.2	แสดงผลเปรียบเทียบค่ามุมเอียงที่ได้จาก Accelerometer กับ Kalman filter .....	29
5.3	ตัวควบคุมที่ออกแบบใน MATLAB/Simulink .....	30
5.4	Electrical Model .....	30
5.5	Torque Transformations .....	31
5.6	Mechanical Model .....	31
5.7	Kinematics C to Cr and Cl .....	32
5.8	กราฟมุมเอียงของหุ่นยนต์ที่ได้จากการ Simulation .....	32
5.9	กราฟมุมหมุนของวงล้อของหุ่นยนต์ที่ได้จากการ Simulation .....	33
5.10	กราฟแรงดันไฟฟ้าที่จ่ายให้กับมอเตอร์ของหุ่นยนต์ที่ได้จากการ Simulation .....	33
5.11	แผนผังการเชื่อมโยงสัญญาณต่างๆ ของอุปกรณ์ .....	34
5.12	กราฟมุมเอียงของหุ่นยนต์ที่ได้จากการทดลอง เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหน้า .....	35
5.13	กราฟ Control Signal ที่จ่ายให้กับ Servo Drive ของหุ่นยนต์ที่ได้ จากการทดลอง เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหน้า .....	35
5.14	กราฟมุมเอียงของหุ่นยนต์ที่ได้จากการทดลอง เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหลัง .....	36
5.14	กราฟ Control Signal ที่จ่ายให้กับ Servo Drive ของหุ่นยนต์ที่ได้ จากการทดลอง เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหลัง .....	36
ก.1	โค้ดทั้งหมดที่อยู่ในส่วนของ FPGA .....	43
ก.2	โค้ดวัดสัญญาณ PWM .....	44
ก.3	โค้ดวัดและสร้างสัญญาณ Analog .....	44
ก.4	โค้ดวัดสัญญาณจาก Encoder .....	44
ก.5	Front panel ของโค้ดในส่วนของ FPGA .....	45
ก.6	Front panel ของโค้ดในส่วนของ Target .....	45
ก.7	โค้ดทั้งหมดที่อยู่ในส่วนของ Target .....	46
ก.8	โค้ดหลักการควบคุมในส่วนของ Target .....	47
ก.9	โค้ดวัดค่า Angular velocity ในส่วนของ Target .....	48

ก.10 โค้ดวัดค่า Tilt angle ในส่วนของ Target .....	48
ก.11 โค้ด Kalman filter ในส่วนของ Target .....	48

# สารบัญตาราง

4.1 ตัวแปรต่างๆ .....	21
-----------------------	----

# บทที่ 1

## บทนำ

ในปัจจุบันนวัตกรรมล้ำสมัยของการควบคุมหุ่นยนต์รถสองล้อ ได้มีการพัฒนาจากหลักการ  
ทำงานของร่างกายมนุษย์ในการทรงตัว และการรักษาสมดุลจากการมองคนสู่สมองกล จาก  
กล้ามเนื้อสู่ฟันเฟือง และจากขาสู่อล้อ ทำให้การควบคุมการขับเคลื่อนที่ เป็นไปอย่างมีประสิทธิภาพ  
และปลอดภัยสูงสุด พร้อมทั้งตัวควบคุมที่ใช้ควบคุมการทรงตัวของรถสองล้อ ซึ่งเป็นแนวคิด  
พื้นฐานมาจากการทำงานของ Gyroscope ในมนุษย์ สำหรับควบคุมชุดอุปกรณ์ล้อและชุดวง  
แหวน ที่อยู่รวมกันในแกนหมุนอิสระ ประสานการทำงานร่วมกันอย่างลงตัวในทุกเสี้ยววินาที ให้วง  
ล้อสามารถรักษาทิศทางการหมุนและการทรงตัวได้อย่างสมดุลตลอดเวลา ความสามารถตอบสนอง  
ทุกการเคลื่อนไหวได้ตามความต้องการ ควบคุมทิศทางทรงตัวและความเร็วอย่างเหมาะสมใน  
ทุกสถานการณ์ ดังเช่นตัวอย่างรถสองล้อของเซกเวย์ (Segway) ถือว่าสุดยอดนวัตกรรมอัจฉริยะ  
สำหรับการเดินทางของมนุษย์ พร้อมเทคโนโลยีสมัยใหม่ สำหรับยนตรกรรมอัจฉริยะเอนกประสงค์  
เพื่อการเดินทาง ที่มีสีสันมากที่สุดในปัจจุบัน ด้วยการออกแบบให้สามารถใช้งานได้ทุก  
สภาพแวดล้อมและภูมิประเทศที่หลากหลาย ทั้งภายในอาคาร ทางเท้า ทางชันหรือพื้นที่ขรุขระ ได้  
อย่างสนุกสนานและปลอดภัยที่สุด สำหรับในประเทศไทยได้มีการนำเข้ายานพาหนะขับเคลื่อน  
ส่วนบุคคล 2 ล้อ หรือเซกเวย์ มาเพื่อจำหน่ายให้กับกลุ่มลูกค้าระดับสูง อาทิเช่น ผู้เล่นกอล์ฟ  
บริษัทผู้จำหน่ายรถยนต์ หรือโรงงานนำเสนอสินค้าต่าง ๆ แต่สำหรับในประเทศไทยยังไม่ได้มี  
การเรียนรู้ การพัฒนาและการออกแบบระบบควบคุมหุ่นยนต์รถสองล้อ หรือรวมถึงแนวทางใน

การออกแบบและพัฒนายานพาหนะขับเคลื่อนส่วนบุคคล 2 ล้อ และทางมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี ได้มีนโยบายเพื่อไปสู่การเป็นผู้นำทางด้านเทคโนโลยี

ด้วยเหตุนี้เองทางคณะผู้วิจัยมีความสนใจและเล็งเห็นความจำเป็นที่จะทำงานวิจัยชิ้นนี้เพื่อที่จะพัฒนาระบบควบคุมการทรงตัวของอินเวอร์สเพนดูลัม (Inverted pendulum) สำหรับมาประยุกต์ใช้ในการควบคุมการทรงตัวของหุ่นยนต์รถสองล้อ

### วัตถุประสงค์ของโครงการวิจัย

เพื่อออกแบบและสร้างหุ่นยนต์รถสองล้อพร้อมทั้งตัวควบคุมที่ใช้ควบคุมการทรงตัวของรถสองล้อ

### ขอบเขตของโครงการวิจัย

1. การวิจัยครั้งนี้ เป็นการออกแบบและสร้างหุ่นยนต์รถสองล้อขนาดเล็กพร้อมทั้งตัวควบคุมที่ใช้ควบคุมการทรงตัวของรถสองล้อ และในการออกแบบการทดลองเพื่อศึกษาองค์ประกอบที่เกี่ยวข้องต่อการควบคุมการทำงาน
2. เครื่องมือที่ใช้ในการวิจัยครั้งนี้ ประกอบด้วย
  - 2.1 โครงสร้างหุ่นยนต์รถสองล้อ มอเตอร์ และตัวตรวจรู้
  - 2.2 โมเดลทางพลศาสตร์ของสองล้อ
  - 2.3 ตัวควบคุมและชุดสัญญาณป้อนกลับจากตัวตรวจรู้ของรถสองล้อ
  - 2.4 โปรแกรมควบคุมและการอินเทอร์เฟสต่างๆในการควบคุมรถสองล้อ
3. หุ่นยนต์รถสองล้อขนาดเล็กควบคุมด้วยระบบอินเวอร์สเพนดูลัม (Inverted pendulum) ในการควบคุมการทรงตัว โดยกำหนดความสามารถในการรับน้ำหนักได้ไม่เกิน 100 กิโลกรัม และมีความเร็วในการเคลื่อนที่ได้สูงสุดไม่เกิน 10 กิโลเมตรต่อชั่วโมง

### ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ต้นแบบของหุ่นยนต์รถสองล้อที่สามารถทรงตัวได้ ซึ่งสามารถที่จะนำไปขยายผลเพื่อพัฒนาในเชิงธุรกิจกับภาคอุตสาหกรรมได้
2. พัฒนาทักษะในการออกแบบตัวควบคุมและการเชื่อมโยงอุปกรณ์ในงานควบคุมอัตโนมัติ
3. รู้ถึงระบบงานควบคุมอัตโนมัติ โครงสร้าง ตัวตรวจรู้ และการอินเทอร์เฟส
4. เป็นอุปกรณ์ประกอบการสอนในด้านการควบคุมอัตโนมัติ และทำให้ผู้สนใจได้รู้หลักการประยุกต์ใช้ระบบควบคุมอัตโนมัติ
5. สามารถนำเสนอผลงานให้กับมหาวิทยาลัยในรูปของการ Conference ในระดับมหาวิทยาลัย หรือ การเผยแพร่ความรู้ด้วยการตีพิมพ์ในเอกสารวิชาการ

### หน่วยงานที่จะนำผลการวิจัยไปใช้ประโยชน์

1. คณะครุศาสตร์อุตสาหกรรม มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
2. คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
3. คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
4. ศูนย์พัฒนาบุคลากรเพื่ออุตสาหกรรมและปิโตรเคมี มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
5. หน่วยงานของรัฐและเอกชนที่มีความสนใจ
6. นักวิจัย นักประดิษฐ์ ของสถาบันการศึกษาต่าง ๆ

# บทที่ 2

## ทฤษฎีที่เกี่ยวข้อง

### 2.1 วรรณกรรมที่เกี่ยวข้อง

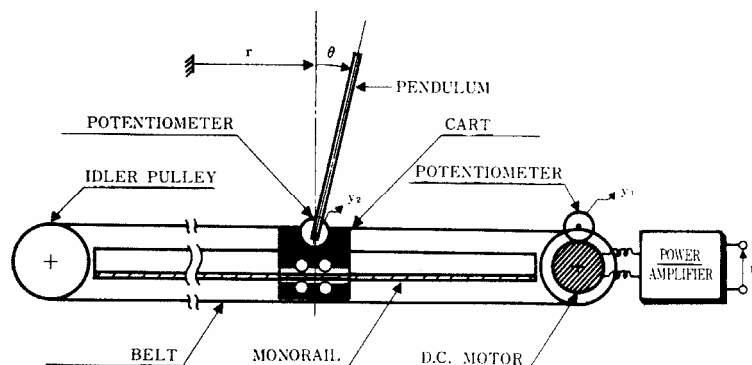
ในการศึกษาวิจัยเกี่ยวกับหุ่นยนต์อุตสาหกรรมและการควบคุม คณะผู้วิจัยได้ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้องเพื่อใช้เป็นข้อมูลพื้นฐานในการทำวิจัย โดยรวบรวมและเรียบเรียงสาระสำคัญไว้ดังต่อไปนี้

Segway [1] เป็นชื่อหนึ่งของยานพาหนะที่มีความพิเศษที่แตกต่างจากยานพาหนะอื่น ก็คือมีสองล้อที่แนวศูนย์กลางล้ออยู่ในแนวเดียวกัน (ซึ่งแตกต่างกับจักรยาน มีแนวศูนย์กลางล้อขนานกัน) และสามารถทรงตัวด้วยตัวเองได้โดยที่ไม่ล้ม ซึ่ง Segway จัดว่าเป็นรถไฟฟ้าชนิดหนึ่ง โดยมีมอเตอร์ไฟฟ้าเป็นตัวต้นกำลังสองตัว และมีเซนเซอร์วัดมุมเอียงของหุ่นยนต์เมื่อเทียบกับพื้นโลก



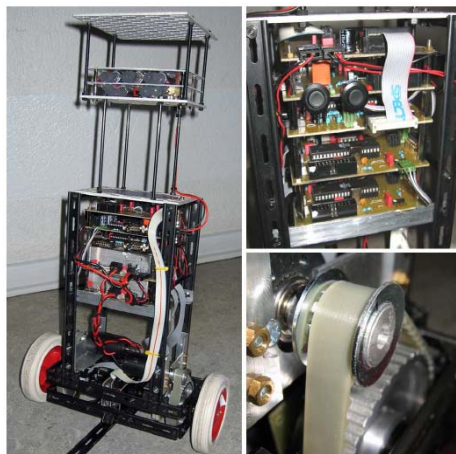
รูปที่ 2.1 Segway [1]

[5] ได้ออกแบบตัวควบคุมที่ใช้ควบคุมระบบอินเวอร์์เพนดูลัมและรถโดยที่ระบบแสดงในรูปที่ 2.2 ซึ่งตัวควบคุมที่ออกแบบก็เพื่อที่จะ 1. เล็งยให้ก้านเพนดูลัมตั้งตรง เมื่อก้านเพนดูลัมอยู่ในช่วงเชิงเส้น (Stabilizing) 2. ทำให้ก้านเพนดูลัมจากตำแหน่งที่ห้อยลงอยู่ด้านล่างแกว่งขึ้นไปอยู่ในช่วงที่เป็นเชิงเส้น (Swing-up) โดยได้ออกแบบตัวควบคุม 2 ชนิดด้วยกันคือ 1. ตัวควบคุมแบบ Feedforward 2. ตัวควบคุมแบบเชิงเส้น (Linear feedback controller) โดยใช้วิธีการ LQR



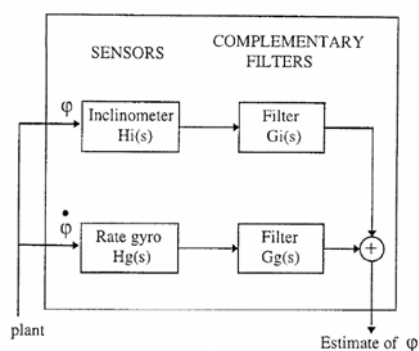
รูปที่ 2.2 ระบบ inverted pendulum-cart [5]

[2] ได้ออกแบบระบบควบคุมหุ่นยนต์รถสองล้อที่มีลักษณะเหมือนกับระบบลูกตุ้มผกผัน โดยใช้ไมโครโปรเซสเซอร์ Atmel MEGA16 เป็นตัวประมวลผล และใช้ Accelerometer ADX202 และ Rate gyro ADXRS300 เป็นอุปกรณ์วัดมุมเอียง ส่วนตำแหน่งและความเร็วของหุ่นได้ใช้ Encoder และตัวควบคุมได้ออกแบบด้วย State feedback โดยใช้ LQR และยังสามารถออกแบบตัวชดเชยแรงเสียดทานที่เกิดขึ้นเพิ่มเข้าไปอีกด้วย



รูปที่ 2.3 YAIP – the inverted pendulum robot [2]

การประมาณค่ามุม Attitude สำหรับ Helicopter ได้ถูกนำเสนอโดย [3] ซึ่งการประมาณค่ามุมจะต้องใช้ข้อมูลหรือสัญญาณที่ได้จาก Inclinometer และ Rate Gyro มาผ่าน Complementary filter โดยที่ Filter จะถูกหามาจาก Transfer function ของ Inclinometer และ Rate Gyro ซึ่งแผนผังการประมาณมุมแสดงไว้ในรูปที่ 3



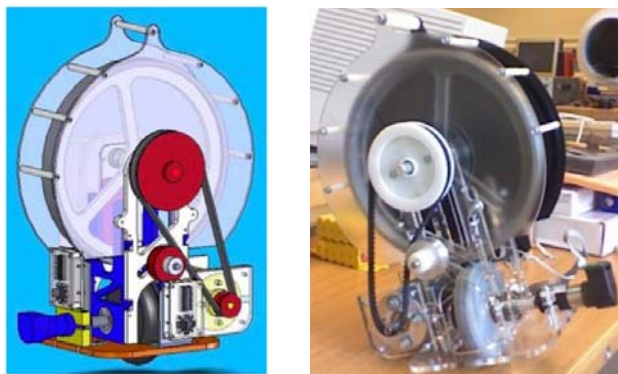
รูปที่ 2.4 แผนผังการประมาณมุมเอียง Attitude โดยใช้หลักการ Complementary filters [3]

Advanced Robotics Lab (ARL) [4] ได้ออกแบบและสร้างหุ่นยนต์ล้อเดี่ยว (Single wheel robot) และเรียกหุ่นยนต์ตัวนี้ว่า Gyrover แสดงในรูปที่ 4 ซึ่ง Gyrover สามารถวิ่งบนบกและในน้ำได้



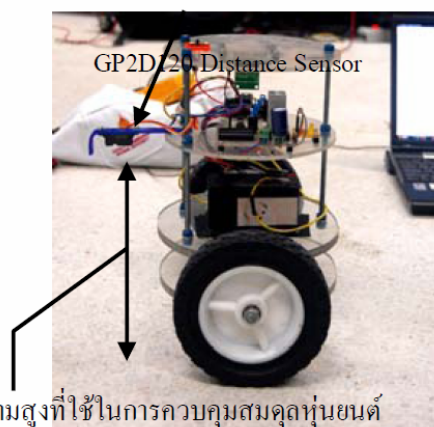
รูปที่ 2.5 Gyrover [8]

Jascha van Pommeren [5] ได้ออกแบบสร้างหุ่นยนต์ที่เรียกว่า Unibot ดังรูปที่ 6 ซึ่งหุ่นยนต์ตัวนี้เป็นหุ่นยนต์ล้อเดี่ยวที่นำเอาหลักการของ Wheeled inverted pendulum และ Inertia wheel pendulum มาประยุกต์ใช้



รูปที่ 2.6 UniBot [9]

อดิศักดิ์ คงฉางและคณะ [8] ได้พัฒนาระบบควบคุมหุ่นยนต์สองล้อให้สมดุลได้ด้วยตัวเอง โดยใช้ไมโครคอนโทรลเลอร์ตระกูล PIC จำนวน 3 ตัว สองตัวแรกใช้ในการควบคุมการหมุนของมอเตอร์แต่ละตัว ส่วนอีกตัวใช้ในการควบคุมหุ่นยนต์ให้ตั้งตัวอยู่ได้ ซึ่งในการวัดมุมเอียงของหุ่นยนต์นั้นใช้เซ็นเซอร์วัดระยะทางจากพื้นแทนการวัดมุมเอียงโดยตรง และใช้ตัวควบคุมแบบ State feedback



รูปที่ 2.7 Two Wheel Balancing Robot ของอดิศักดิ์ คงฉางและคณะ

"ขาหุ่นยนต์" รุ่น "ดับเบิลยูแอล-16 อาร์ที" หุ่นยนต์ดังกล่าวใช้เวลาพัฒนาร่วม 3 ปีเต็มแล้ว เกิดจากความร่วมมือระหว่างทีมงานของศาสตราจารย์อัสสุโอะ ทาคานิชิ อาจารย์คณะวิศวกรรมศาสตร์ มหาวิทยาลัยวาเซดะ กับ บริษัททะมะสุ เป้าหมายสูงสุดเพื่อใช้เป็น "ขาหุ่นยนต์" หรือ "ขากล" พา "คนชรา" และ "ผู้พิการ" เดินเหินไปไหนมาไหนสะดวกขึ้น แม้จะเคลื่อนที่ไปในระยะไม่ไกลนักก็ตามแทนที่จะนั่ง "รถเข็น" ก็ใช้ขากลเป็นเหมือน "ขาเทียมไฮเทค" แทน

โครงการนี้เริ่มต้นมาตั้งแต่ปี 2546 จากนั้นจึงค่อยๆ พัฒนาเรื่อยมาจนในที่สุดที่ทีมงานสามารถทำให้ขาหุ่นยนต์ก้าวเดินขึ้นลงบันไดได้อย่างปลอดภัย นอกจากนั้น ยังทรงตัวได้ในสภาวะที่เดินไปเหยียบสิ่งกีดขวางบนพื้นโหละกบบริษัทหะมะสุ ระบุว่า ดับเบิลยูแอล-16 อาร์ทีรี จะพร้อมนำออกมาทดลองใช้งานจริงๆ ในอีกประมาณ 1-2 ปีข้างหน้า อย่างไรก็ตาม มันยังมีข้อจำกัดตรงที่ยังต้องใช้ "มือ" ควบคุม "คันบังคับ" เพื่อสั่งให้ขากลยยับไปในทิศทางต่างๆ ซึ่งทำให้ผู้พิการทั้งแขนและขาใช้งานหุ่นรุ่นนี้ไม่ได้ แต่ในอนาคตอย่างน้อยๆ 5 ปีต่อจากนี้ เชื่อว่าทีมงานจะคิดค้นวิธีการควบคุมขากลแบบใหม่ที่ไม่ต้องใช้มือ หรือ แขนได้สำเร็จ



รูปที่ 2.8 ดับเบิลยูแอล-16 อาร์ทีรี

โตโยต้าได้นำเสนอ หุ่นยนต์สองล้อที่ออกแบบคล้ายรถเข็นคนป่วย สามารถเคลื่อนที่ได้ด้วยความเร็ว ๒๐ กม./ชม. สามารถเคลื่อนที่ไปตามพื้นที่ขรุขระหรือเป็นเนินขึ้นลงโดยทรงตัวเองได้อัตโนมัติ เมื่อเซ็นเซอร์ตรวจจับพบสิ่งกีดขวางอยู่ข้างหน้า มันจะวิ่งอ้อมไปอีกด้านทันที หุ่นยนต์แบบนี้สามารถพาผู้ป่วย และคนชราไปยังเตียงนอนได้โดยไม่ต้องมีคนคอยช่วย



รูปที่ 2.9 หุ่นยนต์สองล้อที่ออกแบบคล้ายรถเข็นคนป่วย

## 2.2 สมการลากรองจ์ (Lagrange's Equations)

สมการลากรองจ์ของการเคลื่อนที่เป็นวิธีการที่ถูกใช้ในการหาสมการการจำลองทางคณิตศาสตร์ของระบบทางกลศาสตร์ โดยที่สมการลากรองจ์ของการเคลื่อนที่สามารถเขียนอยู่ในรูปสมการได้ดังนี้

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i \quad (2.1)$$

โดยที่  $L$ ,  $q_i$  และ  $Q_i$  คือ Lagrangian function, generalized coordinates หรือ the state variables และ  $e$  generalized forces ตามลำดับ และ Lagrangian function เป็นผลต่างระหว่างพลังงานจลน์กับพลังงานศักย์ คือ

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q) \quad (2.2)$$

โดยที่  $T$  และ  $V$  เป็นพลังงานจลน์และพลังงานศักย์ของระบบตามลำดับ นอกจากนี้ Generalized force  $Q_i$  ประกอบไปด้วยแรงทั้งหมดที่กระทำต่อระบบในทิศทางของ  $i$ -th generalized coordinate ได้แก่

1. External forces
2. Forces due to friction

Generalized force  $Q_i$  สามารถเขียนเป็นสมการได้ดังนี้

$$Q_i = \sum_{j=1}^N \bar{F}_j \frac{\partial \bar{r}_j}{\partial q_i} \quad (2.3)$$

โดยที่  $N$ ,  $\bar{F}_j$  และ  $\bar{r}_j$  คือ จำนวนอนุภาคหรือวัตถุแตร่งในระบบ, เวกเตอร์ของแรงที่กระทำต่ออนุภาคหรือวัตถุแตร่งในระบบ และเวกเตอร์ตำแหน่งของอนุภาคหรือวัตถุแตร่งในระบบ ตามลำดับ

## 2.3 ระบบควบคุมแบบปิดหรือระบบควบคุมป้อนกลับ (Closed loop or feedback control system)

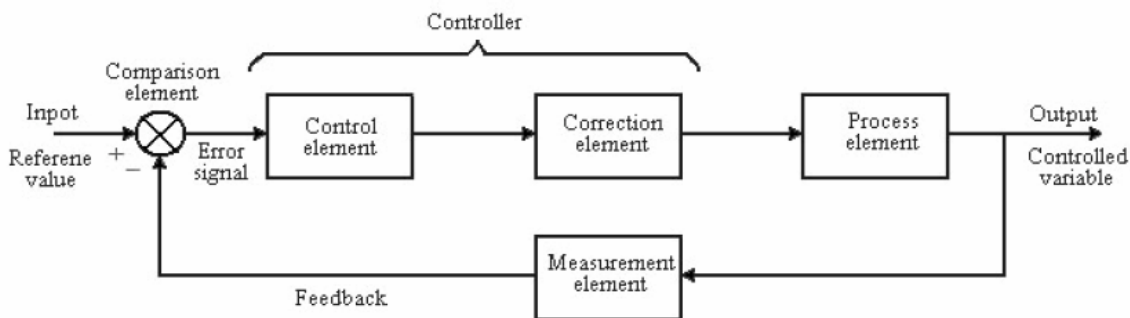
ระบบควบคุมแบบป้อนกลับจะแตกต่างจากระบบควบคุมแบบเปิด ก็คือ มีการนำเอาผลที่ได้จากกระบวนการกลับเข้ามาเป็นส่วนหนึ่งของข้อมูลที่จะส่งไปเป็นอินพุตที่จะให้กับระบบ การที่จะทราบค่าเอาต์พุตได้จะต้องมีการวัดข้อมูลของเอาต์พุต เมื่อทราบค่าเอาต์พุตแล้วมักจะนำค่าเอาต์พุตที่ได้ไปเปรียบเทียบกับเอาต์พุตที่ต้องการ จากนั้นความแตกต่างระหว่างเอาต์พุตที่ต้องการและเอาต์พุตที่แท้จริงจะได้รับการส่งต่อไปสู่อุปกรณ์ควบคุม แล้วส่งต่อเป็นอินพุตเข้าสู่ระบบเพื่อให้ความแตกต่างของเอาต์พุตที่ต้องการและเอาต์พุตที่แท้จริงลดลงเรื่อยๆ จนกระทั่งไม่มีความแตกต่างระหว่างค่าทั้ง

สอง ดังนั้นก็จะได้ว่าค่าเอาต์พุตของระบบเป็นไปตามต้องการ ระบบควบคุมแบบป้อนกลับแสดงในรูปที่ 2.10



รูปที่ 2.10 ระบบควบคุมแบบปิด

ตัวอย่างของระบบควบคุมแบบเปิดที่เห็นกันทั่วไปในชีวิตประจำวันได้แก่ พัดลมไฟฟ้าที่ใช้กันทั่วไป เราเลือกความแรงของพัดลม แล้วเราจะได้อาต์พุตออกมาโดยที่ไม่มีการตรวจสอบหรือตรวจวัดค่าใดๆ ในทำนองกลับกัน เครื่องปรับอากาศจะมีการตรวจสอบว่าอุณหภูมิของห้องเป็นเท่าใดเพื่อที่จะตรวจสอบว่าระบบปรับอากาศควรจะทำงานหรือไม่ ซึ่งเครื่องปรับอากาศจะเป็นตัวอย่างของระบบควบคุมแบบปิดหรือระบบควบคุมแบบป้อนกลับ



รูปที่ 2.11 Diagram ระบบควบคุมแบบปิด

ระบบควบคุมแบบปิดสามารถพิจารณาได้ว่าประกอบด้วยระบบย่อยที่ต่อวางกันตามรูปที่ 9 ในความเป็นจริงระบบย่อยเหล่านี้อาจไม่สามารถที่จะแยกแต่ละชิ้นส่วนออกมาเป็นส่วนๆ ได้ หรือแยกอุปกรณ์ใดอุปกรณ์หนึ่งอย่างซึ่งชัดเจนไปได้ว่า อุปกรณ์นั้นทำหน้าที่อย่างหนึ่งอย่างใดโดยเฉพาะ แต่อุปกรณ์ในความเป็นจริงเหล่านี้ สามารถแยกการทำงานออกเป็น ส่วน ต่าง ๆ ได้ตามที่แสดงในรูปที่ 2.11 โดยส่วนต่าง ๆ ในระบบควบคุมแบบปิดนี้จะประกอบด้วย

- 1) Comparison element ส่วนนี้จะทำหน้าที่เปรียบเทียบค่าตัวแปรที่ต้องการ ออกมา หรืออาจเรียกว่าค่ามาตรฐานของตัวแปรที่เราต้องการ เพื่อเปรียบเทียบกับค่าที่เราวัดค่าตัวแปรนั้นได้ในสภาพความเป็นจริง ซึ่งเป็นที่เป็นเอาต์พุตของระบบ ส่วนนี้จะให้สัญญาณหรือค่า ความผิดพลาดออกมา ซึ่งความผิดพลาดนี้จะบอกให้ทราบว่าขณะนี้ค่าตัวแปรที่ต้องการควบคุม นั้น มีค่าอยู่แตกต่าง จากค่าที่เราต้องการให้มันเป็นเท่าไร นั่นคือ

$$\text{ความผิดพลาด} = \text{ค่าสัญญาณอ้างอิง} - \text{ค่าสัญญาณที่วัดได้}$$

- 2) Control element เป็นส่วนที่ทำหน้าที่ตัดสินใจว่าจะต้องทำอะไร เมื่อได้รับ สัญญาณ ความผิดพลาด เรามักจะใช้คำว่า Controller เมื่อเราเรียกส่วนนี้ รวมถึง Correction element
- 3) Correction element ส่วนนี้มีหน้าที่กำหนดการเปลี่ยนแปลงของตัวแปร เพื่อที่จะ ลดค่า ความผิดพลาดให้น้อยลง เรามักเรียกอุปกรณ์ที่ทำหน้าที่ในส่วนนี้ว่า Actuator
- 4) Process element กระบวนการ หรือ Plant จะเป็นระบบซึ่งเราต้องการควบคุม ค่าตัวแปรตัวใดตัวหนึ่งหรือหลายตัว
- 5) Measure element ส่วนนี้จะเป็นส่วนของเครื่องมือวัด ซึ่งเครื่องมือวัดนี้จะให้ สัญญาณที่ แสดงถึงขนาดของตัวแปรที่เราต้องการที่จะควบคุม และเมื่อได้ค่าที่วัดแล้วก็จะมีการ ป้อนสัญญาณนั้นกลับเข้าสู่ส่วนเปรียบเทียบ (Comparison element) เพื่อให้ระบบ พิจารณาว่ามีความ ผิดพลาดเกิดขึ้นหรือไม่

## 2.4 State Feedback Controller

พิจารณา State space model ของระบบสามารถอธิบายได้ดังนี้

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (2.4)$$

โดยที่  $(A, B, C, D)$  เป็นเมตริกซ์ และ State feedback ถูกแทนค่าด้วยค่าคงที่ที่เมตริกซ์  $K$  และตัวแปร  $r$  เป็นค่าอ้างอิงขาเข้าจากภายนอกของระบบ ดังนั้น สัญญาณควบคุมจริงที่เข้าสู่ระบบก็คือ  $u = r - Kx$  และ Linear quadratic regulator (LQR) จะถูกใช้ในการออกแบบหาค่าควบคุม  $K$  เพื่อที่ทำให้ค่า Performance index มีค่าน้อยที่สุด

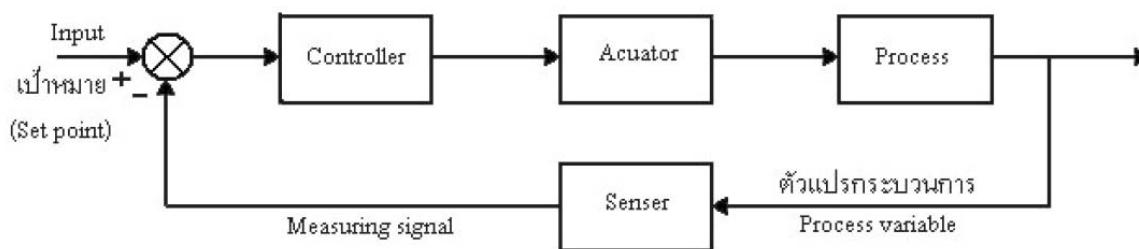
$$J = \int_0^{\infty} (X^T Q X + U^T R U) dt . \quad (2.5)$$

โดยที่  $Q$  เป็น positive semi-definite real matrix และ  $R$  เป็น positive definite real matrix จาก LQR จึงทำให้ทราบค่า  $K$  คือ

$$K = -R^{-1}B^T P \quad (2.6)$$

ส่วน  $P$  เป็น symmetrical solution matrix ในการออกแบบตัวควบคุมด้วย LQR ได้ถูกสมมุติว่าทราบ 1) โมเดลของระบบที่ต้องการควบคุมที่ถูกต้อง 2) State ทุกตัวมาจากการวัดในระบบ แต่





รูปที่ 2.13 ระบบควบคุมแบบป้อนกลับโดยใช้ตัวควบคุมแบบ PID [10]

จากรูปที่ 2.13 กำหนดให้ค่าผิดพลาด  $e(t)$  มีค่าเท่ากับความแตกต่างระหว่างค่าเอาต์พุต ที่วัดได้จริง ณ เวลา  $t$  ใด ๆ ตัวควบคุมแบบ PID จะกำหนดสัญญาณควบคุม  $u(t)$  ไปที่กระบวนการ เพื่อเปลี่ยนแปลงเอาต์พุตของกระบวนการในทิศทางที่ให้เอาต์พุตของกระบวนการเข้าหา หรือไปสู่ ค่าเอาต์พุตที่ต้องการด้วยปริมาณที่ได้จากการคำนวณในสมการ (2.8) ที่อยู่ในเทอมของค่าผิดพลาด

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de}{dt} e(t) \quad (2.8)$$

เมื่อ

- $K_p$  = เป็นค่าเกนสัดส่วนโดยตรง
- $K_I$  = เป็นค่าเกนปริพันธ์
- $K_D$  = เป็นค่าเกนอนุพันธ์ และเป็นเกนเวลาเริ่มต้น

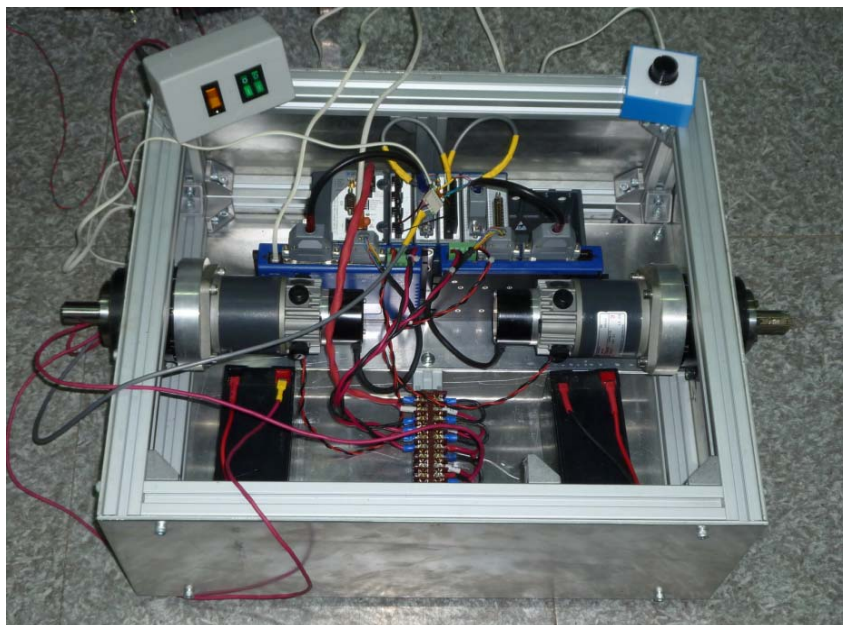
การปรับแต่งตัวควบคุมในที่นี้จะหมายถึงการปรับเปลี่ยนค่าและกำหนดค่าของเกน ต่าง ๆ ของตัวควบคุม PID นั้นเอง จะเป็นว่ามีทั้งหมด 3 ค่าด้วยกัน คือ  $K_p$ ,  $K_I$  และ  $K_D$  สามารถตีความได้ว่าปริมาณของสัญญาณควบคุมจะมีค่ามาจากผลรวมของสามเทอม ซึ่งประกอบด้วย

- 1) ค่าเกนสัดส่วนโดยตรง คูณกับ ค่าผิดพลาด
- 2) ค่าเกนปริพันธ์ คูณกับ ปริพันธ์ของค่าผิดพลาด
- 3) ค่าเกนอนุพันธ์ คูณกับ อนุพันธ์ของค่าผิดพลาด

# บทที่ 3

## โครงสร้างของระบบ

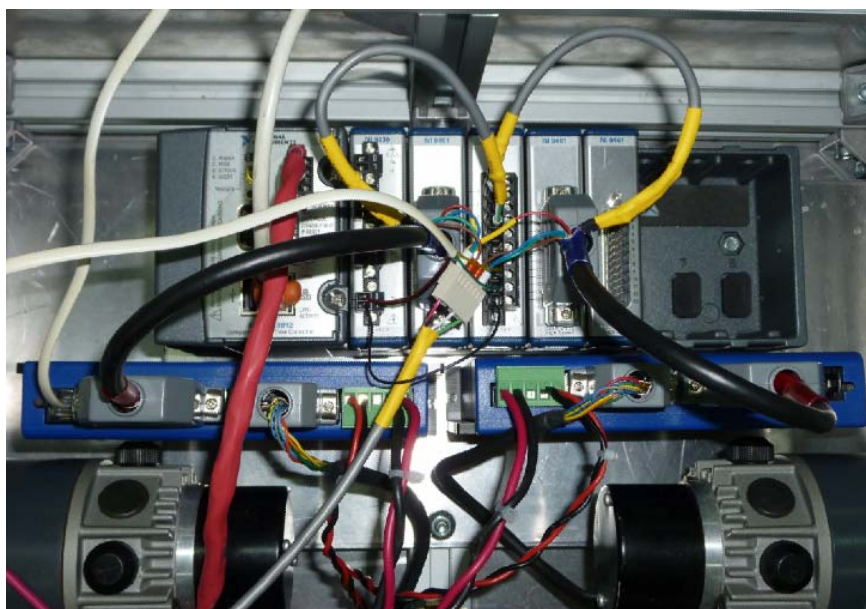
บทนี้จะเป็นการอธิบายถึงโครงสร้างหุ่นยนต์ที่ใช้ในการทดลองและอุปกรณ์ต่างในการควบคุมการเคลื่อนที่ของหุ่นยนต์ เพื่อให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการ ซึ่งอุปกรณ์หลักๆ ได้แก่



รูปที่ 3.1 อุปกรณ์ภายในกล่องควบคุม

### 3.1 ตัวควบคุม (Controller)

โครงการวิจัยนี้ได้นำเอา CompactRIO ของบริษัท National Instruments (NI) (ดังรูปที่ 3.2) มาประยุกต์ใช้ในการควบคุมการทรงตัวและการเคลื่อนที่ของหุ่นยนต์รถสองล้อ โดยชุด CompactRIO ที่ใช้ในโครงการนี้ประกอบไปด้วย



รูปที่ 3.2 ตัวควบคุม CompactRIO และชุดขับเคลื่อนมอเตอร์

1. Real-Time controller รุ่น NI cRIO-9012 Controller มีความเร็วในการประมวลผลอยู่ที่ 400 MHz และมีหน่วยความจำ DRAM 64 MB พร้อมทั้งหน่วยความจำสำหรับเก็บข้อมูล 128 MB มี Ethernet port ไว้สำหรับสื่อสาร 1 port



รูปที่ 3.3 Real-Time controller รุ่น NI cRIO-9012 [10]

2. Reconfiguration chasis รุ่น NI cRIO-9102 Chasis ตัวนี้จะมี 8 slot เพื่อที่จะใส่โมดูลในการควบคุม และมี FPGA รุ่น Virtex-II จำนวน 1 ล้านเกต



รูปที่ 3.4 Reconfiguration chassis รุ่น NI cRIO-9102 [10]

3. Analog Output Module รุ่น NI 9263 Module นี้สามารถส่งสัญญาณ Analog ได้ 4 ช่องสัญญาณในเวลาเดียวกัน และมีความละเอียด 16 Bits ด้วยความเร็วในการเก็บข้อมูล 100 kS/s



รูปที่ 3.5 Analog Output Module รุ่น NI 9263 Module [10]

4. Analog Input Module รุ่น NI 9205 Module นี้สามารถรับสัญญาณ Analog ได้ถึง 32 ช่องสัญญาณ และมีความละเอียด 16 Bits ด้วยความเร็วในการเก็บข้อมูล 250 kS/s



รูปที่ 3.6 Analog Input Module รุ่น NI 9205 Module [10]

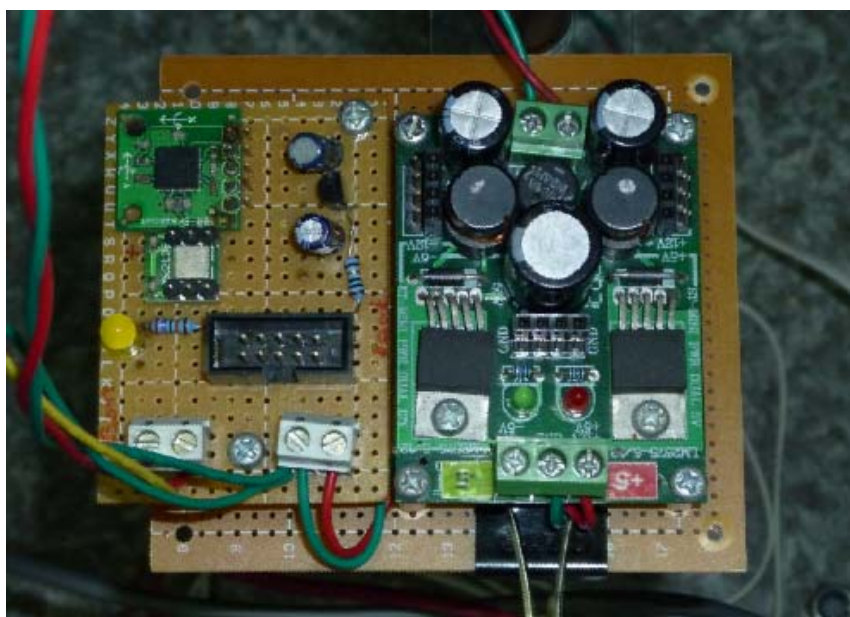
5. Digital I/O Module รุ่น NI 9401 Module นี้ใช้สำหรับรับและส่งสัญญาณ Digital จำนวน 8 ช่องสัญญาณ (5V/TTL) ด้วยความเร็วในการรับส่งข้อมูล 100 ns



รูปที่ 3.7 Digital I/O Module รุ่น NI 9401 Module [10]

### 3.2 เซนเซอร์ Accelerometer กับ Gyroscope

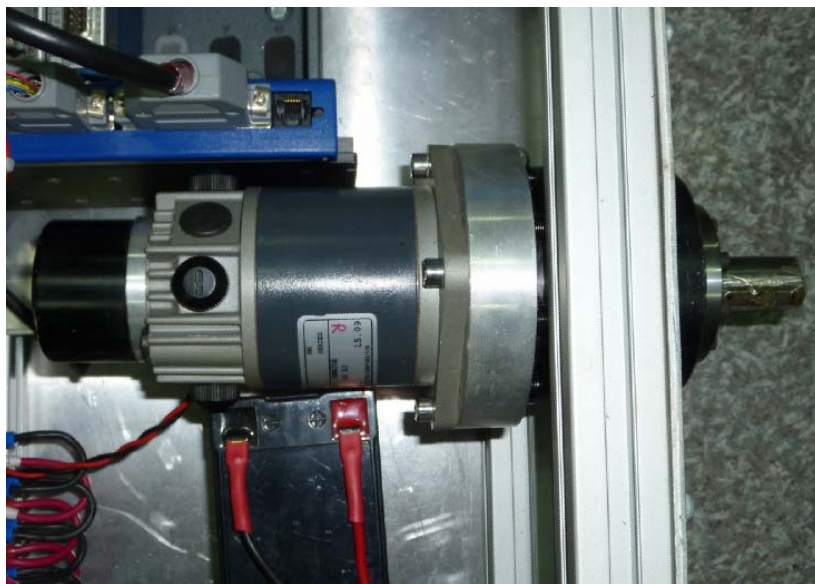
เซนเซอร์ Accelerometer กับ Gyroscope จะถูกนำมาใช้ในการวัดมุมเอียงของหุ่นยนต์รถสองล้อ โดยที่ Accelerometer ได้ใช้ในการวัดมุมเอียงของรถเมื่อเทียบกับพื้นโลก ซึ่งรุ่นที่ใช้คือ MEMSIC 2125 ส่วน Gyroscope รุ่น IDG300 ใช้ในการวัดความเร็วเชิงมุมของการเอียงของรถ ซึ่งเซนเซอร์ทั้งสองจะมีการใช้ตัวกรองสัญญาณที่ชื่อว่า Kalman filter เพื่อให้ได้ค่ามุมเอียงที่ถูกต้องมากขึ้น



รูปที่ 3.8 แผงวงจรเซนเซอร์ Accelerometer กับ Gyroscope

### 3.3 ดีซีเซอร์โวมอเตอร์

ตัวต้นกำลังที่ใช้ในการขับเคลื่อนได้ใช้ดีซีเซอร์โวมอเตอร์ 24 โวลต์จำนวน 2 ตัว เป็นของบริษัท Yasgawa รุ่น UGRMEM-04MA กำลัง 200 วัตต์ ให้แรงบิดต่อเนื่อง 0.706 Nm และมีเอนโคเดอร์ที่ใช้ในการวัดตำแหน่งในการหมุนของมอเตอร์ ซึ่งเป็นชนิด Incremental encoder ให้สัญญาณขาออกแบบ TTL ความละเอียด 2048 pulse/rev และยังมีหัวเกียร์ทดรอบ ชนิด Planetary gear ของบริษัท Harmonic Drive รุ่น HPG-20A อัตราทด 1:21



รูปที่ 3.9 ดีซีเซอร์โวมอเตอร์พร้อมเกียร์ทด

#### 3.4 ชุดขับดีซีมอเตอร์

ชุดขับดีซีมอเตอร์ที่ใช้ในการจ่ายไฟฟ้าให้กับมอเตอร์เป็นของบริษัท Copley Controls รุ่น Accelus ASP-090-36 ซึ่งสามารถจ่ายไฟดีซีได้สูงสุด 90 โวลต์ และจ่ายกระแสได้สูงสุด 36 แอมป์



รูปที่ 3.10 ชุดขับดีซีมอเตอร์

# บทที่ 4

## แบบจำลองของระบบ

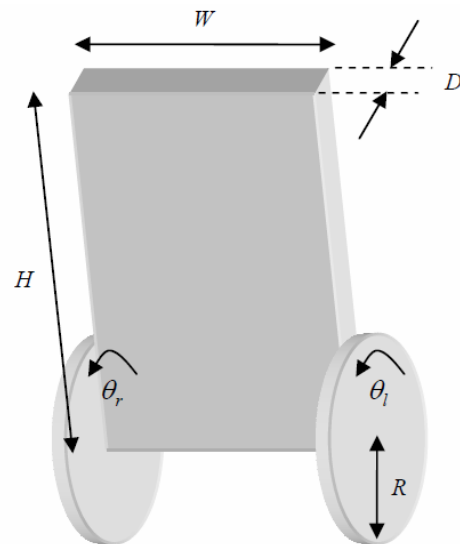
ในบทนี้จะเป็นการคำนวณหาสมการการเคลื่อนที่ของหุ่นยนต์สมดุสองล้อหรือหาแบบจำลองของระบบที่ต้องการควบคุมที่ใช้ในการทดลอง ในการคำนวณหาสมการการเคลื่อนที่ของระบบหุ่นยนต์นั้นสามารถแยกออกได้เป็น 2 ส่วนหลักๆ คือ

1. สมการการเคลื่อนที่ของตัวโครงสร้างหุ่นยนต์
2. สมการการเคลื่อนที่ของตัวต้นกำลังหรือมอเตอร์

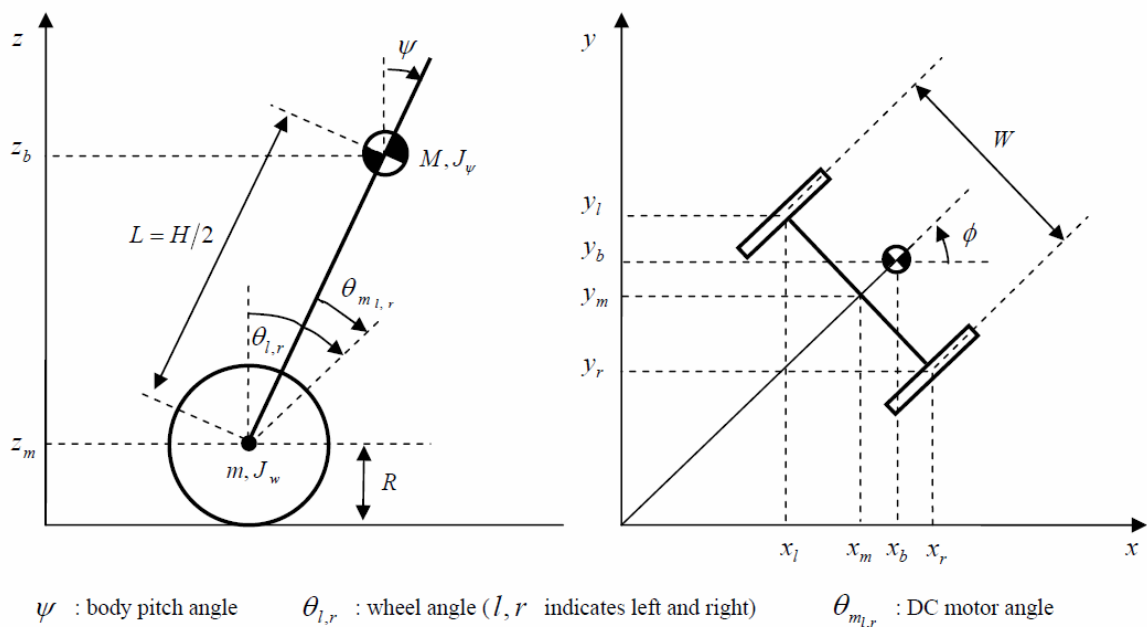
เมื่อได้สมการทั้งคู่แล้วก็นำมารวมกันเป็นสมการการเคลื่อนที่ของระบบหุ่นยนต์

### 4.1 สมการการเคลื่อนที่ของตัวโครงสร้างหุ่นยนต์

สำหรับสมการนี้คณะผู้วิจัยได้อ้างอิงจาก [9]



รูปที่ 4.1 โมเดลของหุ่นยนต์สองล้อ [9]



รูปที่ 4.2 ด้านข้างและด้านบนของหุ่นยนต์สองล้อ [9]

สมการลากรองจ์ของการเคลื่อนที่ (2.1) จะถูกนำมาใช้ในการหาแบบจำลองทางคณิตศาสตร์ของตัวโครงสร้างหุ่นยนต์ ซึ่งก่อนอื่นจะต้องคำนวณหาค่าพลังงานจลน์และศักย์ของตัวโครงสร้างหุ่นยนต์ได้ดังนี้

พลังงานจลน์

$$\begin{aligned}
T = & \frac{1}{2}m(\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2}m(\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \frac{1}{2}M(\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \\
& + \frac{1}{2}J_w \dot{\theta}_l^2 + \frac{1}{2}J_w \dot{\theta}_r^2 + \frac{1}{2}J_\psi \dot{\psi}^2 + \frac{1}{2}J_\phi \dot{\phi}^2 + \frac{1}{2}n^2 J_m (\dot{\theta}_l - \dot{\psi})^2 \\
& + \frac{1}{2}n^2 J_m (\dot{\theta}_r - \dot{\psi})^2
\end{aligned} \tag{4.1}$$

พลังงานศักย์

$$U = mgz_l + mgz_r + mgz_b \tag{4.2}$$

ตารางที่ 4.1 ตัวแปรต่างๆ

$m$	มวลของวงล้อ	5	kg
$M$	มวลของตัวหุ่นยนต์และคนขับ	70	kg
$R$	รัศมีของวงล้อ	0.2413	m
$W$	ความกว้างของตัวหุ่นยนต์และคนขับ	0.4	m
$D$	ความหนาของตัวหุ่นยนต์และคนขับ	0.25	m
$H$	ความสูงของตัวหุ่นยนต์และคนขับ	1.6	m
$L = H/2$	ระยะจากจุดศูนย์กลางมวลถึงแกนหมุนวงล้อ	0.8	m
$J_w = mR^2/2$	Inertia moment ของวงล้อ	0.1456	kg·m <sup>2</sup>
$J_\psi = ML^2/3$	Pitch inertia moment ของตัวหุ่นยนต์และคนขับ	14.933	kg·m <sup>2</sup>
$J_\phi = M(W^2 + D^2)/12$	Yaw inertia moment ของตัวหุ่นยนต์และคนขับ	1.2979	kg·m <sup>2</sup>
$J_m$	Inertia moment ของมอเตอร์และเกียร์	1.68x10 <sup>-4</sup>	kg·m <sup>2</sup>
$R_m$	ค่าความต้านทานของมอเตอร์	0.41	Ω
$L_m$	ค่าความเหนี่ยวนำของมอเตอร์	0.6	mH
$K_b$	ค่าคงที่ Back emf ของมอเตอร์	0.119	V·s/rad
$K_t$	ค่าคงที่แรงบิดของมอเตอร์	0.119	N·m/A
$n$	อัตราทดรอบของเกียร์	21:1	

โดยที่ค่าตัวแปรในสมการด้านบนมีค่าดังต่อไปนี้

$$[\theta \ \phi]^T = \left[ \frac{1}{2}(\theta_l + \theta_r) \quad \frac{R}{W}(\theta_r - \theta_l) \right]^T \tag{4.3}$$

$$\begin{bmatrix} x_m & y_m & z_m \end{bmatrix}^T = \left[ \int \dot{x}_m dt \quad \int \dot{y}_m dt \quad R \right]^T \quad (4.4)$$

$$\begin{bmatrix} \dot{x}_m & \dot{y}_m \end{bmatrix}^T = \begin{bmatrix} R\dot{\theta} \cos \phi & R\dot{\theta} \sin \phi \end{bmatrix}^T \quad (4.5)$$

$$\begin{bmatrix} x_l & y_l & z_l \end{bmatrix}^T = \begin{bmatrix} x_m - \frac{W}{2} \sin \phi & y_m + \frac{W}{2} \cos \phi & z_m \end{bmatrix}^T \quad (4.6)$$

$$\begin{bmatrix} x_r & y_r & z_r \end{bmatrix}^T = \begin{bmatrix} x_m + \frac{W}{2} \sin \phi & y_m - \frac{W}{2} \cos \phi & z_m \end{bmatrix}^T \quad (4.7)$$

$$\begin{bmatrix} x_b & y_b & z_b \end{bmatrix}^T = \begin{bmatrix} x_m + L \sin \psi \cos \phi & y_m + L \sin \psi \sin \phi & z_m + L \cos \psi \end{bmatrix}^T \quad (4.8)$$

ดังนั้น Lagrangian function (2.2) สามารถคำนวณหาได้ดังนี้

$$\begin{aligned} L = & \frac{1}{2} m (\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2} m (\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \frac{1}{2} M (\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \\ & + \frac{1}{2} J_w \dot{\theta}_l^2 + \frac{1}{2} J_w \dot{\theta}_r^2 + \frac{1}{2} J_\psi \dot{\psi}^2 + \frac{1}{2} J_\phi \dot{\phi}^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_l - \dot{\psi})^2 \\ & + \frac{1}{2} n^2 J_m (\dot{\theta}_r - \dot{\psi})^2 - mgz_l - mgz_r - mgz_b \end{aligned} \quad (4.9)$$

และสมการแบบจำลองของหุ่นยนต์ก็คือ

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = \tau_\theta \quad (4.10)$$

$$\begin{aligned} \left\{ (2m + M)R^2 + 2J_w + 2n^2 J_m \right\} \ddot{\theta} + \left\{ MLR \cos \psi - 2n^2 J_m \right\} \ddot{\psi} \\ - MLR \dot{\psi}^2 \sin \psi = \tau_\theta \end{aligned} \quad (4.11)$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\psi}} \right) - \frac{\partial L}{\partial \psi} = \tau_\psi \quad (4.12)$$

$$\begin{aligned} \left\{ MLR \cos \psi - 2n^2 J_m \right\} \ddot{\theta} + \left\{ ML^2 + J_\psi + 2n^2 J_m \right\} \ddot{\psi} - MgL \sin \psi \\ - ML^2 \dot{\phi}^2 \sin \psi \cos \psi = \tau_\psi \end{aligned} \quad (4.13)$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = \tau_\phi \quad (4.14)$$

$$\left\{ \frac{1}{2} m W^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2 J_m) + ML^2 \sin^2 \psi \right\} \ddot{\phi} + 2ML^2 \dot{\psi} \dot{\phi} \sin \psi \cos \psi = \tau_\phi \quad (4.15)$$

จะเห็นได้ว่าสมการแบบจำลองของหุ่นยนต์ (4.11) (4.13) (4.15) เป็นสมการไม่เชิงเส้น ดังนั้นจึงต้องทำการแปลงให้เป็นสมการเชิงเส้นได้ดังนี้

$$\{(2m+M)R^2 + 2J_w + 2n^2J_m\} \ddot{\theta} + \{MLR - 2n^2J_m\} \ddot{\psi} = \tau_\theta \quad (4.16)$$

$$\{MLR - 2n^2J_m\} \ddot{\theta} + \{ML^2 + J_\psi + 2n^2J_m\} \ddot{\psi} - MgL\psi = \tau_\psi \quad (4.17)$$

$$\left\{ \frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2J_m) \right\} \ddot{\phi} = \tau_\phi \quad (4.18)$$

โดยที่

$$\begin{bmatrix} \tau_\theta & \tau_\psi & \tau_\phi \end{bmatrix}^T = \begin{bmatrix} n\tau_l + n\tau_r & -n\tau_l - n\tau_r & \frac{W}{2R}(n\tau_r - n\tau_l) \end{bmatrix} \quad (4.19)$$

#### 4.2 สมการการเคลื่อนที่ของตัวต้นกำลังหรือมอเตอร์

ตัวต้นกำลังที่ใช้ในโครงการนี้เป็นดีซีเซอร์โวมอเตอร์ของ Yaskawa ซึ่งพารามิเตอร์ต่างๆ ที่จำเป็นในการโมเดลนั้นได้แสดงไว้ในตารางที่ 4.1 โดยทั่วไปแล้วสมการการเคลื่อนที่ของมอเตอร์จะแบ่งออกเป็นสองส่วน คือ 1. ส่วนของวงจรรีเลย์ทรอนิกส์ ซึ่งมีแรงดันไฟฟ้าเป็นตัวแปรขาเข้า และมีกระแสที่ไหลอยู่ในขดลวดมอเตอร์เป็นตัวแปรขาออก 2. ส่วนของทางกล มีแรงบิดที่ได้จะกระแสที่ไหลผ่านขดลวดมอเตอร์เป็นตัวแปรขาเข้า และมีความเร็วในการหมุนหรือตำแหน่งในการหมุนของมอเตอร์เป็นตัวแปรขาออก จากสมการการเคลื่อนที่ของตัวโครงสร้างหุ่นยนต์ เราจะเห็นได้ว่าตัวแปรขาเข้าเป็นแรงบิดทั้งหมด ดังนั้นเราจึงจำเป็นที่จะต้องโมเดลสมการการเคลื่อนที่ของตัวต้นกำลังหรือมอเตอร์ให้มีตัวแปรขาออกเป็นแรงบิดด้วยเช่นกัน ด้วยเหตุนี้สมการการเคลื่อนที่ของตัวต้นกำลังหรือมอเตอร์จะใช้เพียงส่วนแรกเพียงส่วนเดียว คือ ส่วนของวงจรรีเลย์ทรอนิกส์ เพราะมีตัวแปรขาออกเป็นกระแสที่สามารถแปลงเป็นแรงบิดได้ ซึ่งสมการการเคลื่อนที่ของมอเตอร์ในส่วนของวงจรรีเลย์ทรอนิกส์สามารถเขียนได้ดังนี้

$$L_m \frac{di_{l,r}}{dt} + R_m i_{l,r} - K_b (\dot{\psi} - \dot{\theta}_{l,r}) = v_{l,r} \quad (4.20)$$

จากสมการที่ (4.20) จะเห็นได้ค่า  $L$  มีค่าเพียง 0.00006 H เท่านั้น ซึ่งมีค่าน้อยมาก ดังนั้นเราจึงคิดว่าค่านี้มีค่าเท่ากับศูนย์ จึงทำให้เทอมแรกของสมการทางซ้ายมือหายไป เหลือเพียง

$$i_{l,r} = \frac{K_b (\dot{\psi} - \dot{\theta}_{l,r}) + v_{l,r}}{R_m} \quad (4.21)$$

ในสมการที่ (14.21) สมการการหาค่ากระแสที่ไหลอยู่ในขดลวดมอเตอร์ ดังนั้นถ้าต้องการแรงบิดก็ต้องนำเอาค่าคงที่แรงบิดของมอเตอร์  $K_t$  มาคูณได้ดังนี้

$$\tau_{l,r} = K_t i_{l,r} = \frac{K_t K_b (\dot{\psi} - \dot{\theta}_{l,r}) + K_t v_{l,r}}{R_m} \quad (4.22)$$

#### 4.3 สมการการเคลื่อนที่ของหุ่นยนต์สองล้อ

จากสมการที่ได้ (4.16) ถึง (4.19) เป็นสมการของโครงสร้างของหุ่นยนต์ ส่วนสมการที่ (4.22) เป็นสมการหาแรงบิดของมอเตอร์ เมื่อนำเอาสมการที่ (4.22) เข้าไปแทนในสมการที่ (4.16) ถึง (4.19) จะได้สมการรวมของทั้งระบบคือ

$$\begin{aligned} \left\{ (2m + M)R^2 + 2J_w + 2n^2 J_m \right\} \ddot{\theta} + \left\{ MLR - 2n^2 J_m \right\} \ddot{\psi} \\ = \frac{2nK_t K_b}{R_m} \dot{\psi} - \frac{2nK_t K_b}{R_m} \dot{\theta} + \frac{nK_t v_l}{R_m} + \frac{nK_t v_r}{R_m} \end{aligned} \quad (4.23)$$

$$\begin{aligned} \left\{ MLR - 2n^2 J_m \right\} \ddot{\theta} + \left\{ ML^2 + J_\psi + 2n^2 J_m \right\} \ddot{\psi} - MgL\psi \\ = \frac{-2nK_t K_b}{R_m} \dot{\psi} + \frac{2nK_t K_b}{R_m} \dot{\theta} - \frac{nK_t v_l}{R_m} - \frac{nK_t v_r}{R_m} \end{aligned} \quad (4.24)$$

$$\begin{aligned} \left\{ \frac{1}{2} mW^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2 J_m) \right\} \ddot{\phi} \\ = \frac{-W^2 K_t K_b}{2R_m R^2} \dot{\phi} + \frac{nWK_t}{2R_m R} v_r - \frac{nWK_t}{2R_m R} v_l \end{aligned} \quad (4.25)$$

จากนั้นนำสมการการเคลื่อนที่ทั้ง 3 (4.23) ถึง (4.25) เราก็จะนำมาเพียง 2 สมการแรกมาใช้ในการควบคุม ก็เนื่องมาจากสมการทั้งสองจะมีตัวแปรขาออกเป็นมุม  $\theta$  และ  $\psi$  ซึ่งเพียงพอสำหรับการควบคุม และสมการที่ (4.25) ไม่มีความเกี่ยวข้องกับสมการที่ (4.23) และ (4.24) เลย จากนั้นนำสมการที่ (4.23) และ (4.24) มาจัดรูปให้อยู่ใน State-space model จะได้ดังต่อไปนี้คือ

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (4.26)$$

โดยที่  $x = [\theta \ \dot{\theta} \ \psi \ \dot{\psi}]^T$  และ  $u = [v_r \ v_l]^T$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & A_{22} & A_{23} & A_{24} \\ 0 & 0 & 0 & 1 \\ 0 & A_{42} & A_{43} & A_{44} \end{bmatrix} \quad (4.27)$$

$$A_{22} = -gML(MLR - 2n^2 J_m) / E \quad -57.9812$$

$$A_{23} = -2 \frac{nK_t K_b}{R_m} (ML^2 + J_w + MLR) / E \quad -0.8391$$

$$A_{24} = 2 \frac{nK_t K_b}{R_m} (ML^2 + J_w + MLR) / E \quad 0.8391$$

$$A_{42} = gML((2m + M)R^2 + 2J_w + 2n^2 J_m) / E \quad 22.1148$$

$$A_{43} = 2 \frac{nK_t K_b}{R_m} (MLR + (2m + M)R^2 + 2J_w) / E \quad 0.2115$$

$$A_{44} = -2 \frac{nK_t K_b}{R_m} (MLR + (2m + M)R^2 + 2J_w) / E \quad -0.2115$$

$$E = 2J_w J_w + [2J_w + 2n^2 J_m + 2mR^2]ML^2 + [M + 2m]R^2 J_w + [2J_w + 4J_w + 2MR^2 + 4mR^2 + 4MLR]n^2 J_m = 126.627$$

$$B = \begin{bmatrix} 0 & 0 \\ B_{21} & B_{22} \\ 0 & 0 \\ B_{41} & B_{42} \end{bmatrix} \quad (4.28)$$

$$B_{21} = B_{22} = \frac{nK_t}{R_m} (ML^2 + J_w + MLR) / E \quad 2.8139$$

$$B_{41} = B_{42} = -\frac{nK_t}{R_m} (MLR + (2m + M)R^2 + 2J_w) / E \quad -0.8887$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (4.29)$$

และ

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (4.30)$$

และเมื่อแทนค่าตัวแปรต่างๆในสมการที่ (4.27) และ (4.28) ด้วยค่าจากตารางที่ 4.1 เราจะได้สมการแบบจำลองของหุ่นยนต์สองล้อคือ

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -57.9812 & -0.8391 & 0.8391 \\ 0 & 0 & 0 & 1 \\ 0 & 22.1148 & 0.2115 & -0.2115 \end{bmatrix} \text{ และ } B = \begin{bmatrix} 0 & 0 \\ 2.8139 & 2.8139 \\ 0 & 0 \\ -0.8887 & -0.8887 \end{bmatrix}$$

# บทที่ 5

## การทดลองและผลลัพธ์

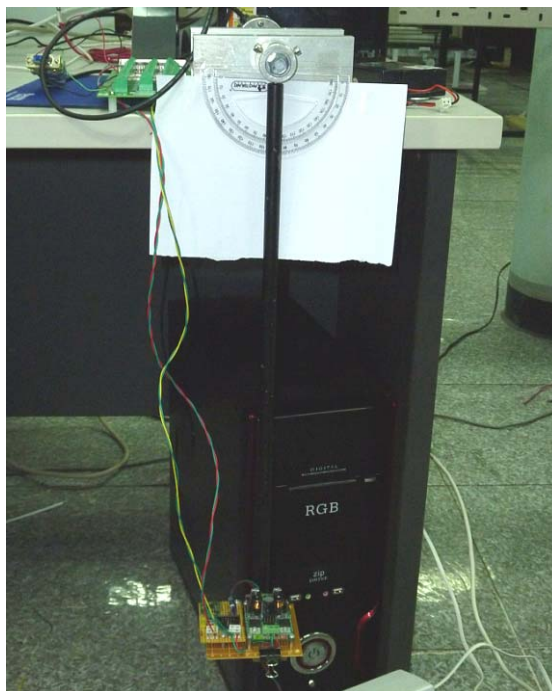
ในบทนี้จะเป็นการอธิบายถึงการทดลองและผลลัพธ์ที่ได้ ซึ่งจะกล่าวถึงขั้นตอนการออกแบบตัวควบคุมรวมอยู่ในบทนี้ด้วย การทดลองนั้นได้ถูกแบ่งออกเป็น 2 ส่วนหลักๆ คือ

1. การทดลองออกแบบตัวประมาณค่ามุมเอียงของตัวหุ่น
2. การทดสอบตัวควบคุมที่ได้ออกแบบ โดยนำมาทดสอบจริงกับระบบควบคุมหุ่นยนต์

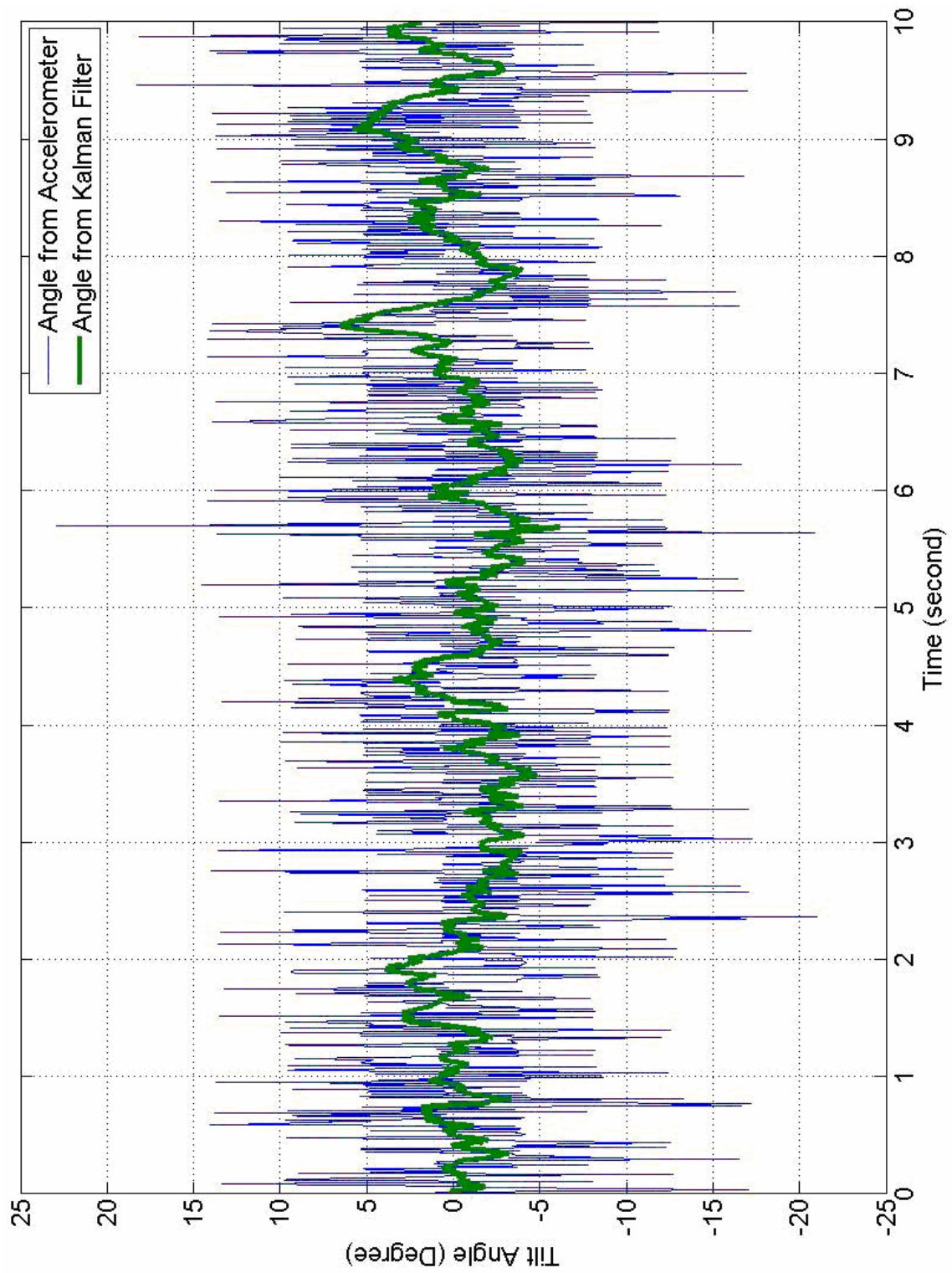
### 5.1 การทดลองออกแบบตัวประมาณค่ามุมเอียงของตัวหุ่น

ในการวัดมุมเอียงของหุ่นยนต์เมื่อเทียบกับพื้นโลกนั้นได้ใช้เซนเซอร์อยู่สองชนิดรวมเข้าด้วยกัน คือ Accelerometer กับ Gyroscope ดังแสดงไว้ในรูปที่ 3.8 โดยหลักการแล้ว Accelerometer จะถูกใช้วัดค่ามุมเอียงโดยตรง ส่วน Gyroscope จะถูกใช้วัดความเร็วเชิงมุมของการหมุนหรือการเคลื่อนที่รอบของแกนของมอเตอร์ ซึ่งถ้าหุ่นยนต์ไม่มีการเคลื่อนที่บนพื้นเลย มีเฉพาะการหมุนรอบแกนมอเตอร์ เราจะสามารถใช้ Accelerometer เพียงตัวเดียวในการวัดมุมเอียง แต่ทว่าหุ่นยนต์จะต้องเคลื่อนที่บนระนาบพื้นด้วย ดังนั้นเมื่อหุ่นยนต์เคลื่อนที่ก็จะทำให้ Accelerometer วัดมุมเอียงผิดพลาด เนื่องจากได้รับผลของความเร่งของการเคลื่อนบนระนาบเข้าไปด้วย ดังนั้นจึงจำเป็นต้องใช้ Gyroscope เข้าร่วมด้วย และจะต้องออกแบบตัวประมาณค่ามุมเอียง โดยใช้ข้อมูลที่ได้จาก Accelerometer กับ Gyroscope ในโครงการนี้ได้ใช้ Kalman filter มา

ช่วยในการประมาณค่ามุมเอียงหรือพุดอีกนัยหนึ่งคือ ใช้ Kalman filter มาช่วยกรองเอาค่าความเร่งออกจากค่าที่อ่านได้จาก Accelerometer



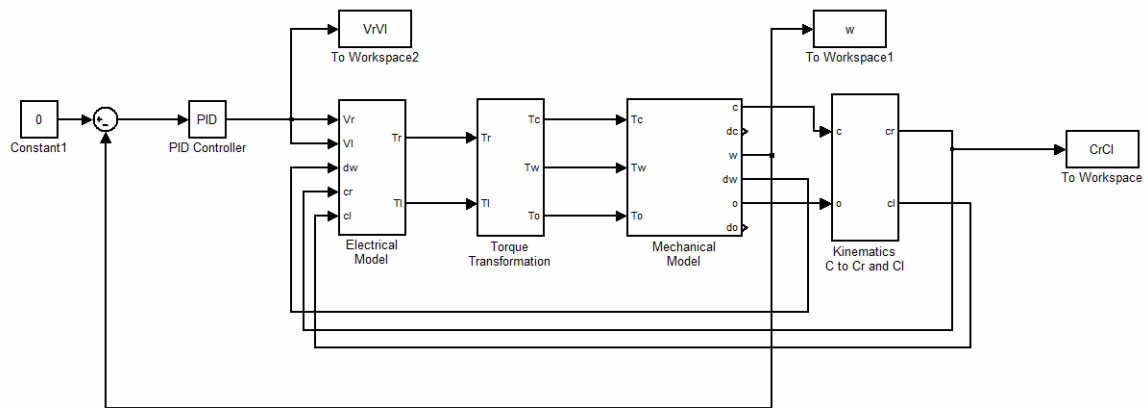
รูปที่ 5.1 การทดสอบ Accelerometer และ Gyroscope



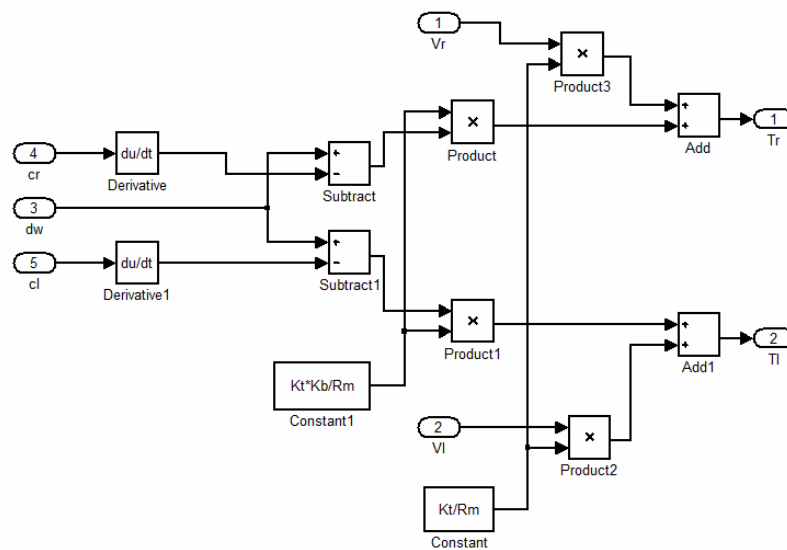
รูปที่ 5.2 แสดงผลเปรียบเทียบค่ามุมเอียงที่ได้จาก Accelerometer กับ Kalman filter

## 5.2 การทดลองออกแบบตัวควบคุมหุ่นยนต์

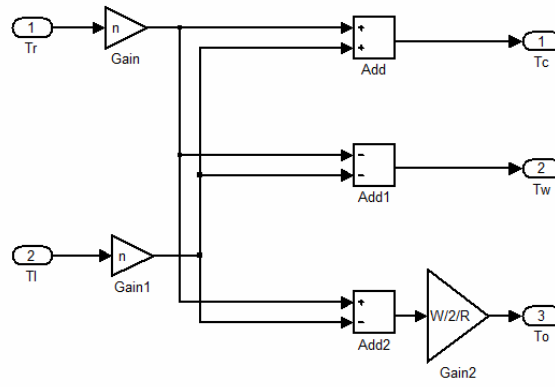
ในการออกแบบตัวควบคุมการทรงตัวของหุ่นยนต์นั้น ผู้วิจัยได้ทำการออกแบบและทดสอบตัวควบคุมผ่านด้วยซอฟต์แวร์ MATLAB/Simulink ก่อนที่จะนำเอาตัวควบคุมมาทดลองใช้กับตัวหุ่นยนต์จริง โดยที่ทฤษฎีตัวควบคุมที่เลือกใช้นั้นได้ใช้ ตัวควบคุมแบบพีไอดี (PID controller)



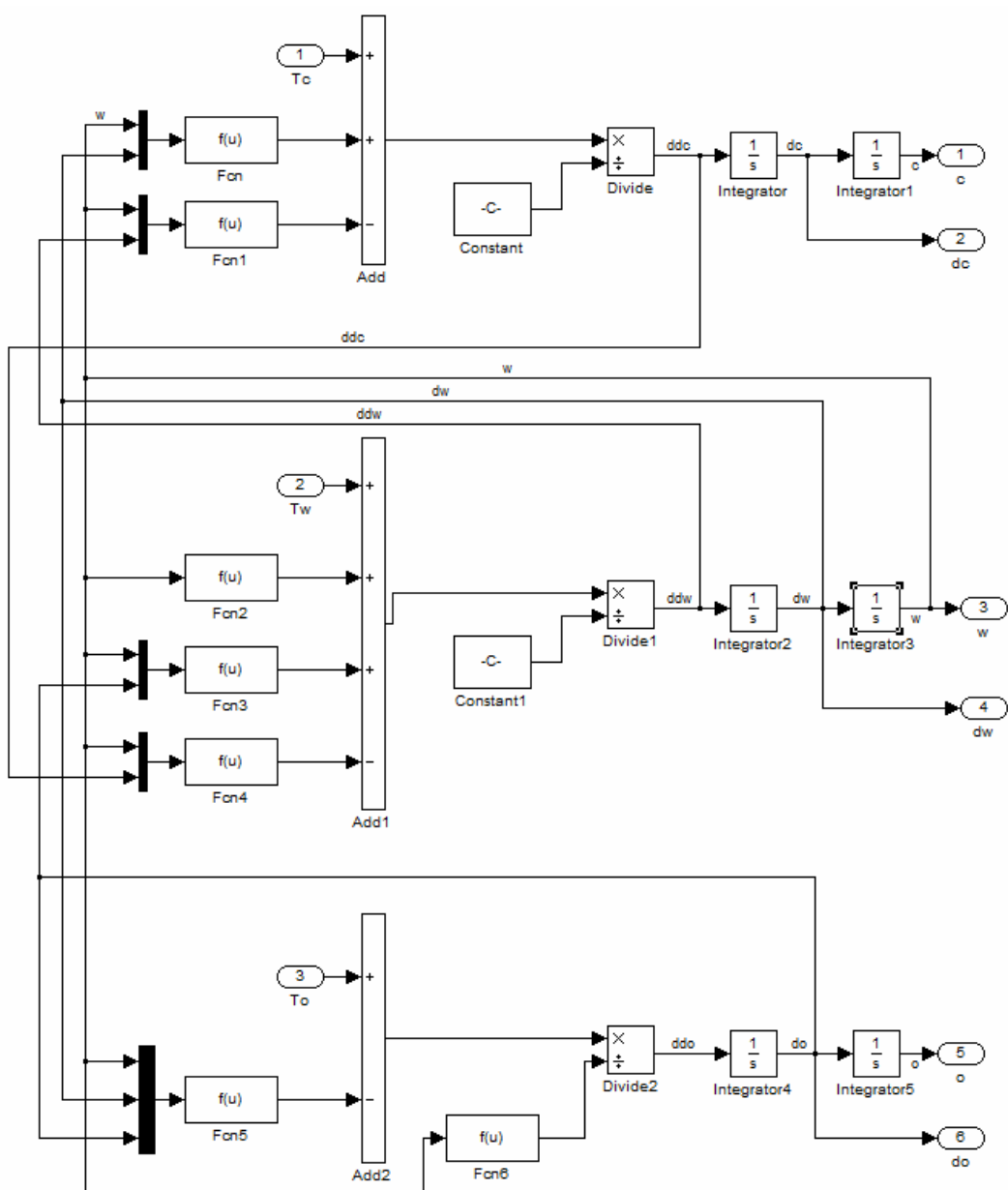
รูปที่ 5.3 ตัวควบคุมที่ออกแบบใน MATLAB/Simulink



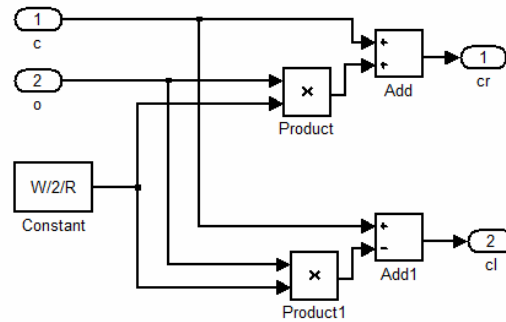
รูปที่ 5.4 Electrical Model



รูปที่ 5.5 Torque Transformations

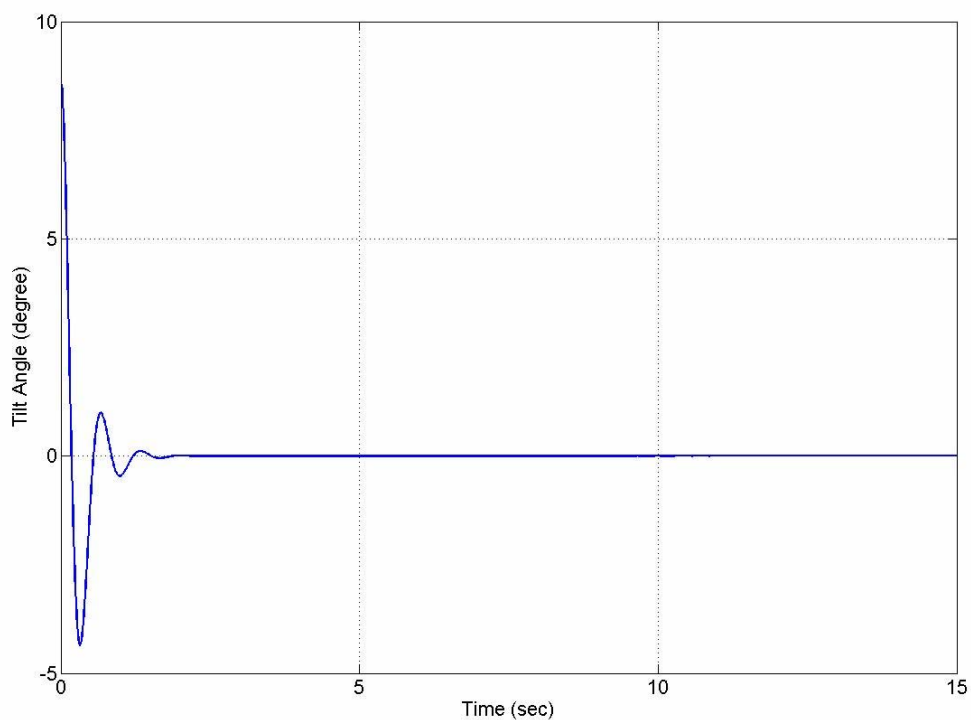


รูปที่ 5.6 Mechanical Model

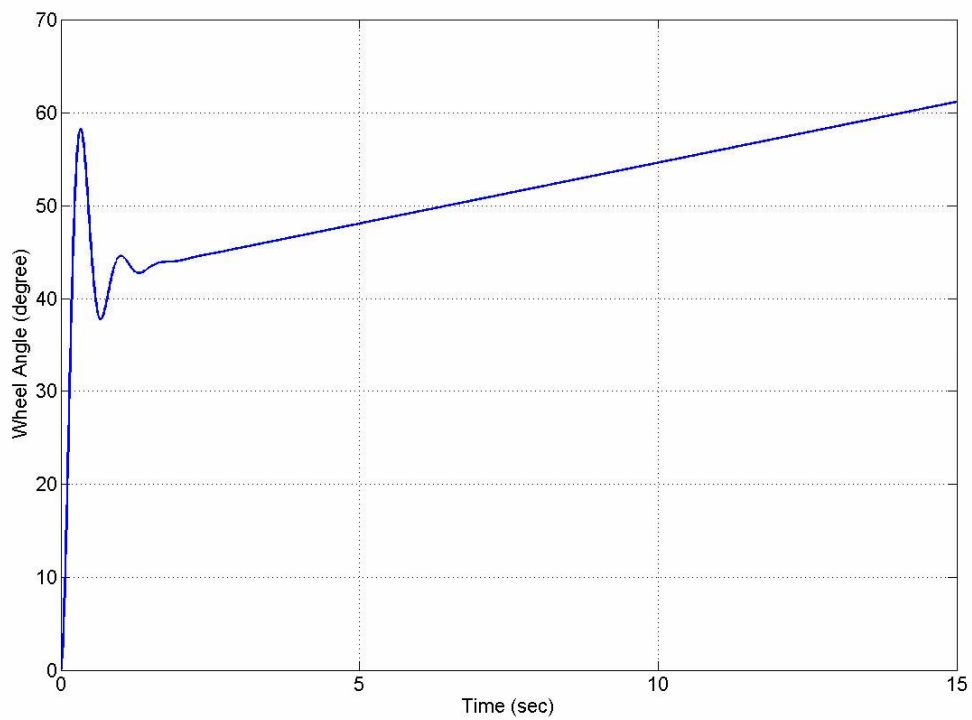


รูปที่ 5.7 Kinematics C to Cr and Cl

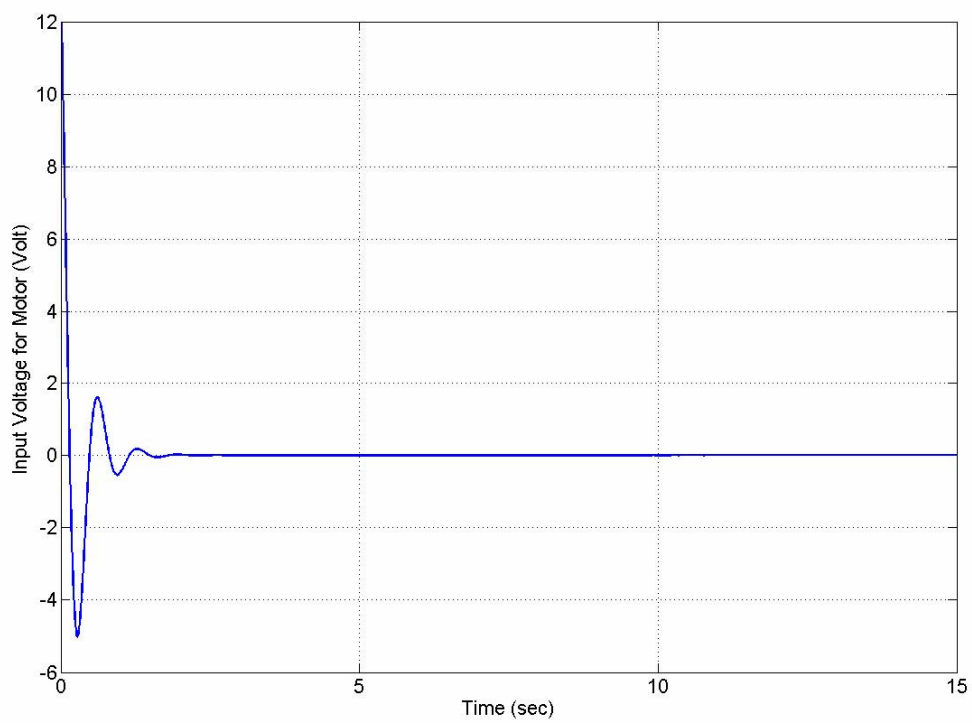
ในการ Simulation การควบคุมหุ่นยนต์ด้วยซอฟต์แวร์ MATLAB/Simulink นั้น ได้กำหนดให้ค่า  $K_p = -80$ ,  $K_I = -200$  และ  $K_D = -5$  ซึ่งผลการ Simulation ได้แสดงในรูปที่ 5.8 ถึง 5.10 โดยที่รูปที่ 5.8 ได้แสดงให้เห็นว่าตัวควบคุมที่ออกแบบนั้นสามารถควบคุมหุ่นยนต์ให้ทรงตัวได้ไม่ล้ม คือเอียงให้มุมเอียงที่ค่าเท่ากับ “ศูนย์” ได้ตลอด แต่ที่รูปที่ 5.9 ก็แสดงให้เห็นว่าหุ่นยนต์จะเคลื่อนที่ไปข้างหน้าตลอดเวลา ด้วยความเร็วในการหมุนของวงล้อประมาณ 8 องศาต่อวินาที



รูปที่ 5.8 กราฟมุมเอียงของหุ่นยนต์ที่ได้จากการ Simulation

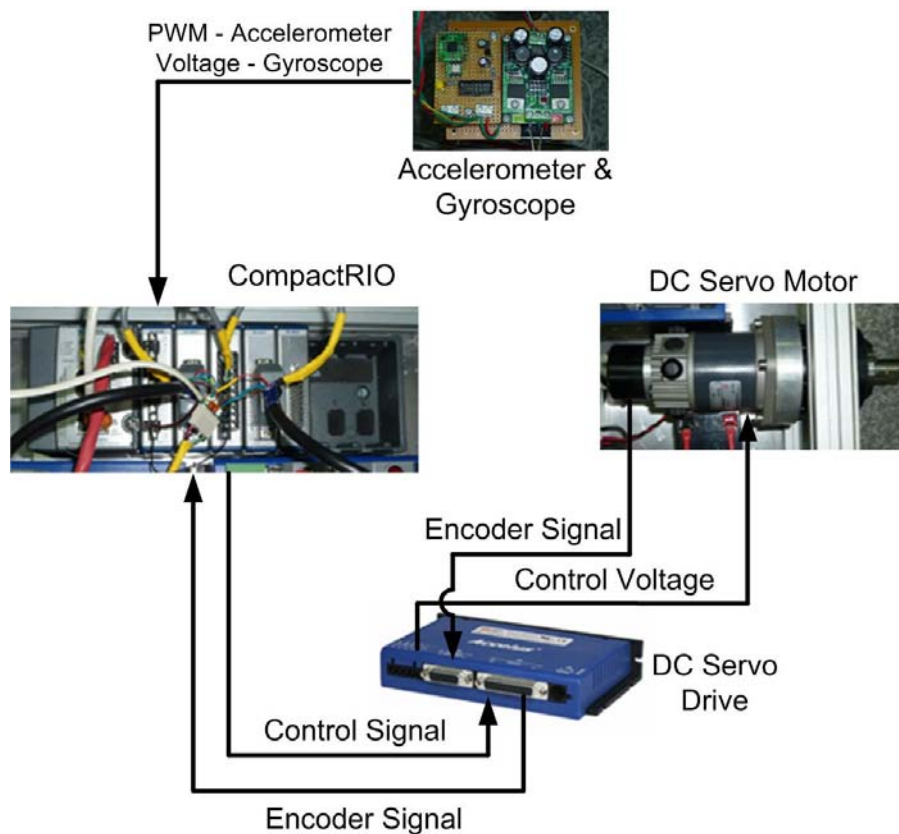


รูปที่ 5.9 กราฟมุมหมุนของวงล้อของหุ่นยนต์ที่ได้จากการ Simulation



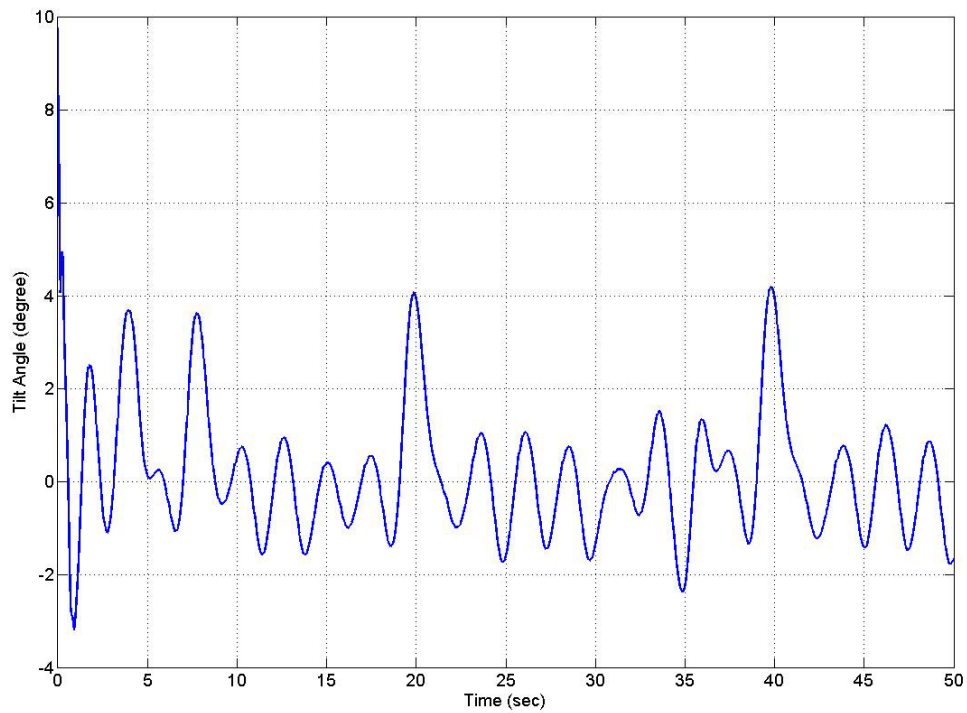
รูปที่ 5.10 กราฟแรงดันไฟฟ้าที่จ่ายให้กับมอเตอร์ของหุ่นยนต์ที่ได้จากการ Simulation

จากนั้นก็ได้นำเอาตัวควบคุมที่ได้ทดลอง Simulation ด้วยซอฟต์แวร์ MATLAB/Simulink มาทดลองกับหุ่นยนต์จริง โดยที่แผนผังการเชื่อมโยงสัญญาณต่างๆ ของอุปกรณ์ได้แสดงไว้ในรูปที่ 5.11

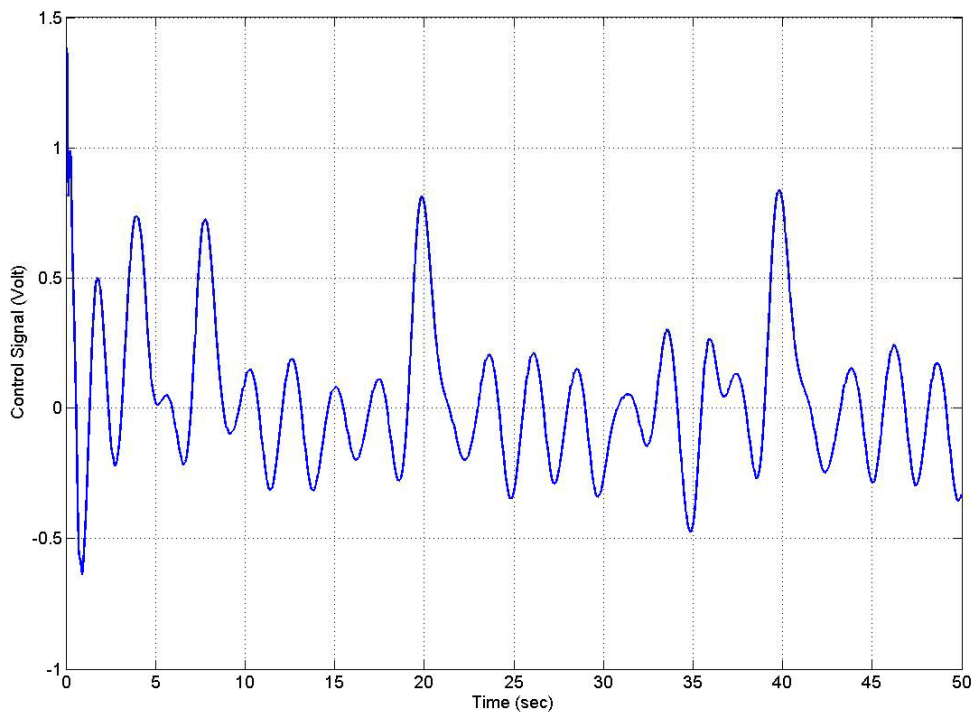


รูปที่ 5.11 แผนผังการเชื่อมโยงสัญญาณต่างๆ ของอุปกรณ์

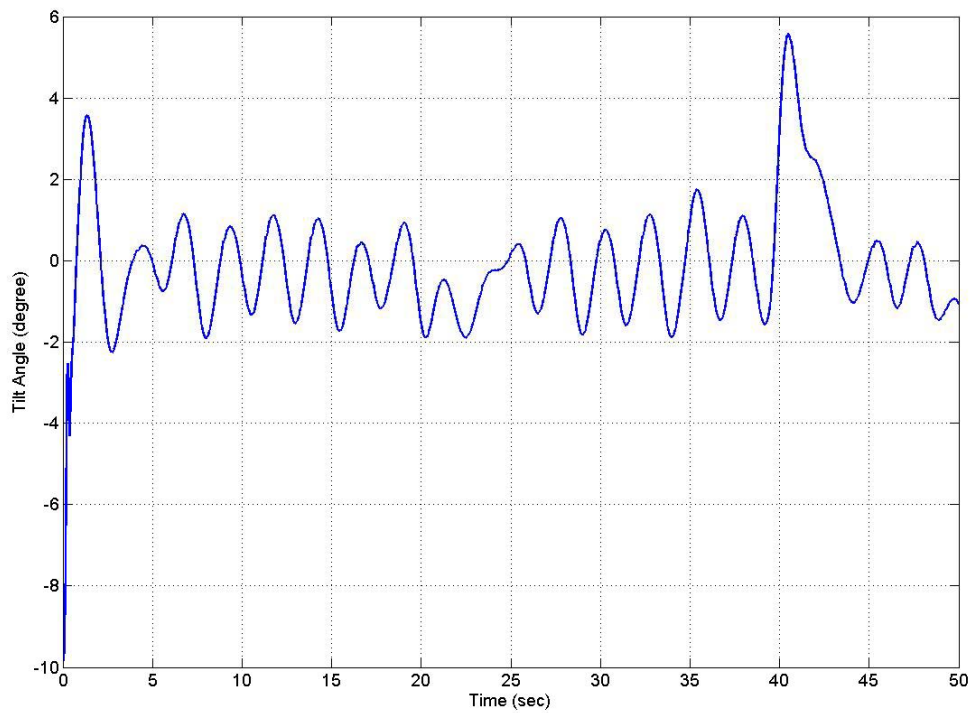
จากรูปที่ 5.11 เป็นแผนผังการส่งสัญญาณการวัดและควบคุมของหุ่นยนต์ จะอธิบายเริ่มจาก Accelerometer และ Gyroscope ซึ่งอุปกรณ์จะถูกใช้ในการวัดมุมเอียงของตัวหุ่นยนต์ โดยที่ Accelerometer จะมีสัญญาณขาออกเป็น PWM ส่วน Gyroscope จะมีสัญญาณขาออกเป็น Analog สัญญาณทั้งสองจะถูกส่งไปยังตัวควบคุม CompactRIO ใน CompactRIO จะมีโมดูล Timer/Counter ใ้รับสัญญาณ PWM และมีโมดูล Analog input ใ้รับสัญญาณ Analog จาก Gyroscope CompactRIO จะทำงานด้วย Realtime OS ของ LabVIEW และคณะผู้วิจัยได้พัฒนาโปรแกรมควบคุมการทรงตัวด้วยหลักการควบคุมของ PID Controller โดยใช้ซอฟต์แวร์ LabVIEW CompactRIO ยังมีก็ติดต่อกับ DC servo drive เพื่อใช้ในการควบคุมการเคลื่อนที่ของหุ่นยนต์ โดยสัญญาณที่ส่งออกไปเป็น Analog  $\pm 10$  โวลต์ และยังมีการป้อนกับตำแหน่งการหมุนของมอเตอร์จาก Encoder ที่อยู่ด้านท้ายของมอเตอร์กลับไปยัง CompactRIO



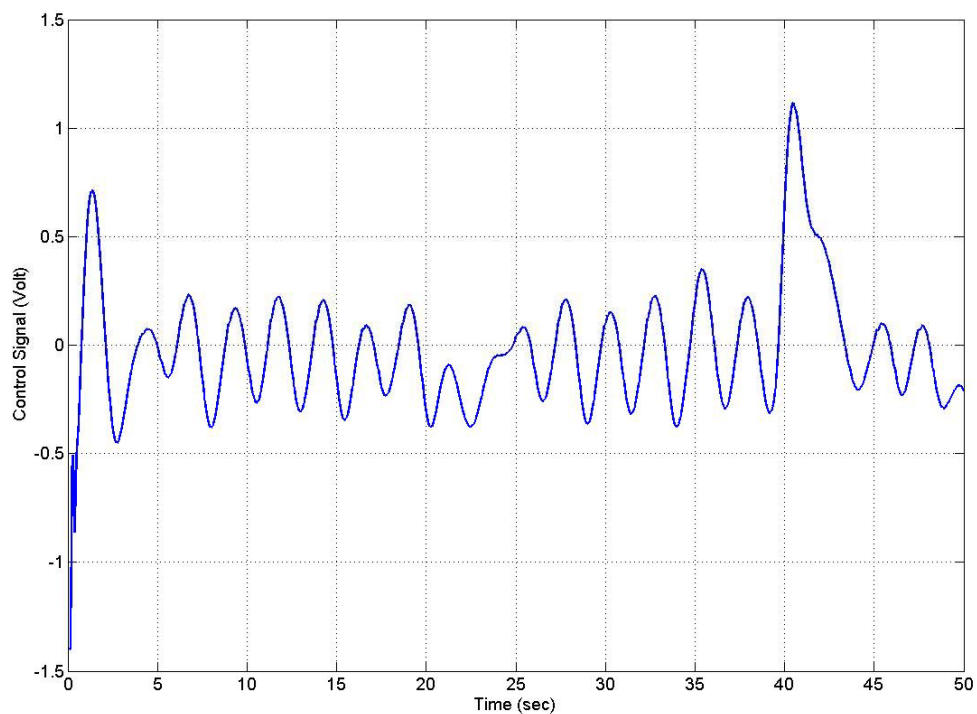
รูปที่ 5.12 กราฟมุมเอียงของหุ่นยนต์ที่ได้จากการทดลอง  
เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหน้า



รูปที่ 5.13 กราฟ Control Signal ที่จ่ายให้กับ Servo Drive ของหุ่นยนต์ที่ได้จากการทดลอง  
เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหน้า



รูปที่ 5.14 กราฟมุมเอียงของหุ่นยนต์ที่ได้จากการทดลอง  
เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหลัง



รูปที่ 5.15 กราฟ Control Signal ที่จ่ายให้กับ Servo Drive ของหุ่นยนต์ที่ได้จากการทดลอง  
เมื่อเริ่มต้นให้หุ่นยนต์เอียงไปด้านหลัง

ในการทดลองการควบคุมการทรงตัวของหุ่นยนต์ได้นำเอาตัวควบคุม PID Controller และ Kalman filter ที่ได้ออกแบบมาโปรแกรมด้วยซอฟต์แวร์ LabVIEW โดยที่โค้ดของโปรแกรมได้แสดงไว้ในภาคผนวก ก. ซึ่งผลทดลองได้แสดงไว้ในรูปที่ 5.12-15 โดยที่รูปที่ 5.12 และ 5.14 เป็นผลการทดลองที่ได้บันทึกค่ามุมเอียงของหุ่นยนต์ ถ้าหุ่นยนต์ไม่มีการเอียงหรือตั้งฉากกับพื้นโลก มุมเอียงนี้จะมีค่าเท่ากับศูนย์ นั่นหมายความว่าเป้าหมายของการควบคุมก็คือ การเลี้ยงให้หุ่นยนต์มีมุมเอียงอยู่ที่ประมาณ “ศูนย์” นั่นเอง จากกราฟก็แสดงให้เห็นว่าหุ่นยนต์สามารถเลี้ยงตัวเองให้มีมุมเอียงอยู่รอบ ๆ ค่าศูนย์ได้ตลอดระยะเวลาของการทดลอง ส่วนรูปที่ 5.13 และ 5.15 เป็นผลการทดลองที่ได้บันทึกค่าSignal ที่จ่ายให้กับ Servo Drive ของหุ่นยนต์

# บทที่ 6

## บทสรุป

โครงการวิจัยนี้เป็นการออกแบบและสร้างหุ่นยนต์รถสองล้อ โดยเป้าหมายหลักของโครงการวิจัยก็คือออกแบบตัวควบคุมหุ่นยนต์รถสองล้อให้สามารถทรงตัวอยู่ได้ โดยหุ่นยนต์ที่ออกแบบสร้างนี้เป็นหุ่นยนต์ที่มีโครงสร้างเป็นอลูมิเนียมและมีตัวถังของหุ่นยนต์เป็นรูปทรงสี่เหลี่ยมขนาด 45x40x30 ซม. ซึ่งภายในตัวถังจะบรรจุอุปกรณ์ต่างๆ ที่ใช้ในการควบคุมและอุปกรณ์ต้นกำลัง ประกอบไปด้วย 1) ดีซีโซเวอร์มอเตอร์ขนาด 200 วัตต์ ของ Yaskawa รุ่น UGRMEM-04MA จำนวน 2 ตัว โดยที่มอเตอร์แต่ละตัวก็จะมีเกียร์ทดรอบขนาดอัตราทด 1:21 ติดอยู่เพื่อเพิ่มแรงบิดของมอเตอร์ให้สูงขึ้น 2) แผงวงจรขับดีซีโซเวอร์มอเตอร์ของบริษัท Copley Controls รุ่น Accelus ASP-090-36 จำนวน 2 แผง เพื่อจ่ายไฟให้กับมอเตอร์แต่ละตัว 3) ตัวประมวลผลที่ใช้ในการควบคุมการเคลื่อนที่และทรงตัวของหุ่นยนต์ ผู้วิจัยได้เลือกนำเอาอุปกรณ์ควบคุมของบริษัท National Instruments (NI) คือ CompactRIO มาประยุกต์ใช้เป็นตัวควบคุมการเคลื่อนที่ของหุ่นยนต์ 4) Accelerometer และ Gyroscope เป็นเซนเซอร์ช่วยในการวัดมุมเอียงของหุ่นยนต์เทียบกับพื้นโลก

ในการควบคุมหุ่นยนต์รถสองล้อได้ใช้ซอฟต์แวร์ LabVIEW ในการประดิษฐ์โปรแกรมควบคุมการเคลื่อนที่ สำหรับตัวทฤษฎีการควบคุมนั้นผู้วิจัยได้เลือกใช้ตัวควบคุมแบบพีไอดี (PID controller) ซึ่งหลักการหาค่าที่หรือค่าพารามิเตอร์ของตัวควบคุมนั้น ผู้วิจัยได้ใช้วิธี Trial and error และในการวัดมุมเอียงหุ่นยนต์รถสองล้อได้ใช้ Accelerometer และ Gyroscope เป็น

เซนเซอร์ช่วยวัด ซึ่ง Accelerometer จะมีสัญญาณรบกวนค่อนข้างมากถ้าหุ่นยนต์เคลื่อนที่ เนื่องจากว่า Accelerometer จะวัดค่าความเร่งของหุ่นยนต์รวมเข้ามาเพิ่มกับค่ามุมเอียงของหุ่นยนต์ จึงทำให้ค่ามุมเอียงที่ได้จาก Accelerometer จะมีค่าความผิดพลาดค่อนข้างมาก ดังนั้นจึงได้ใช้ Kalman filter เข้ามาช่วยในการประมาณค่ามุมเอียงที่ได้จาก Accelerometer และ Gyroscope

การทดลองนั้นทางคณะผู้ทำโครงการวิจัยได้นำเอาตัวควบคุมที่ได้ออกแบบไว้มาทดลองกับ หุ่นยนต์จริง โดยเป้าหมายของการควบคุมก็คือ การเลี้ยงให้หุ่นยนต์มีมุมเอียงอยู่ที่ประมาณ “ศูนย์” ซึ่งได้ผลการทดลองแสดงให้เห็นว่า (ซึ่งได้แสดงไว้ในรูปที่ 5.12 และ 5.14) หุ่นยนต์สามารถเลี้ยงตัวเองให้มีมุมเอียงอยู่รอบๆ ค่าศูนย์ได้ตลอดระยะเวลาของการทดลอง

### ข้อเสนอแนะ

1. ในการวิจัยต่อไปควรจะทดลองออกแบบและสร้างหุ่นยนต์สองล้อแบบชนิดราคาถูก คือเลือกอุปกรณ์ต่างๆ ที่ราคาไม่แพงมาใช้สร้างหุ่นยนต์
2. เพื่อศึกษาทฤษฎีการควบคุม ควรจะมีการทดลองใช้ตัวควบคุมชนิดอื่น อาทิเช่น State Feedback, Nonlinear controller เป็นต้น

# บรรณานุกรม

- [1] Segway Inc. Segway homepage, <http://www.segway.com>, 2010.
- [2] Mori, S., Nishihara, H. & Furuta, K., "Control of unstable mechanical system –control of pendulum", *Int. J. Control*, vol. 23, no. 5, 1976.
- [3] Akesson, J., Blomdell, A. & Braun, R., "Design and Control of YAIP-an Inverted Pendulum on Two Wheels Robot", *IEEE Int. Conf. on Control Application*, Munich, Germany, Oct. 4-6, 2006.
- [4] Baerveldt, A.J. & Klang, R., "A Low-cost and Low-weight Attitude Estimation System for an Autonomous Helicopter", *IEEE Int. Conf. on Intelligent Engineering Systems (INES '97)*, Budapest, Hungary, Sept. 15-17, 1997.
- [5] The Chinese University of Hong Kong homepage, <http://arl.acae.cuhk.edu.hk/node/341>, 2008.
- [6] Jascha van Pommeren homepage, <http://sicl.ucsd.edu/jaschavp/Project.html>, 2008
- [7] Franklin, G. F., Powell, J. D. & Emami-Naeini, A., *Feedback Control of Dynamic Systems*, Addison Wesley, 2<sup>nd</sup>, 1991.
- [8] อติศักดิ์ คงฉาง, วรทิต อ่อนประเสริฐ และอิทธิโชค จักรไพวงศ์, "หุ่นยนต์สองล้อที่สามารถสมดุลด้วยตัวเอง", *Proceeding of the CRIT*, 2007.
- [9] Yamamoto, Y., "NXTway-GS Model-Based Design- Control of self-balancing two-wheeled robot built with LEGO Mindstorms NXT", rev. 1.4, 2009.
- [10] National Instruments homepage: <http://www.ni.com>, 2010.

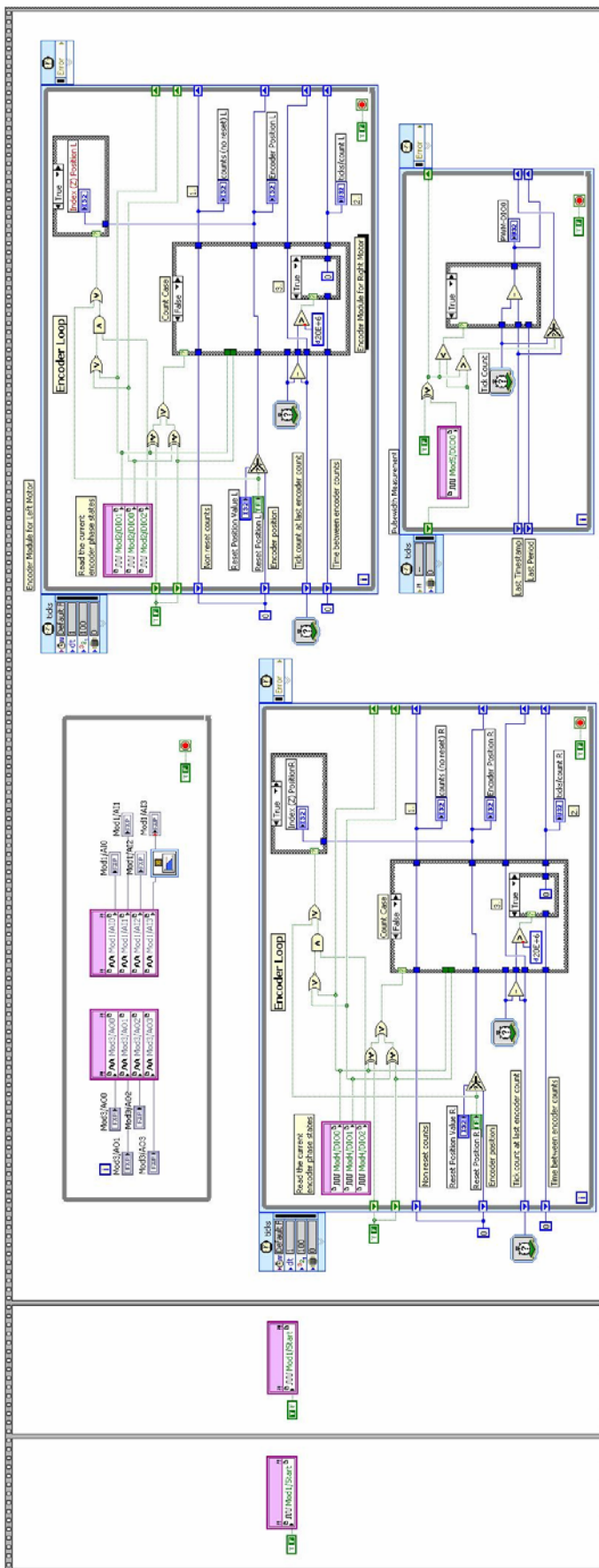
- [11] Fraden, J, A., *Handbook of Modern Sensors - Physic, Designs and Applications*, Springer-Verlag New York, 3<sup>rd</sup>, 2005.

ภาคผนวก

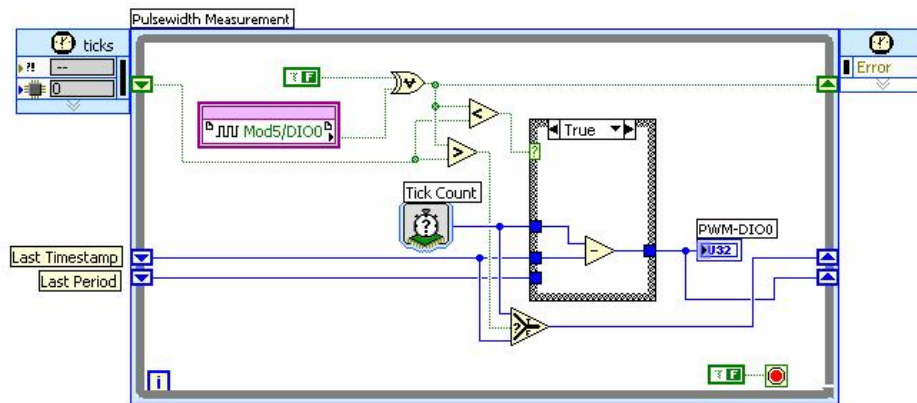
ก.

โค้ดโปรแกรมที่ใช้ใน

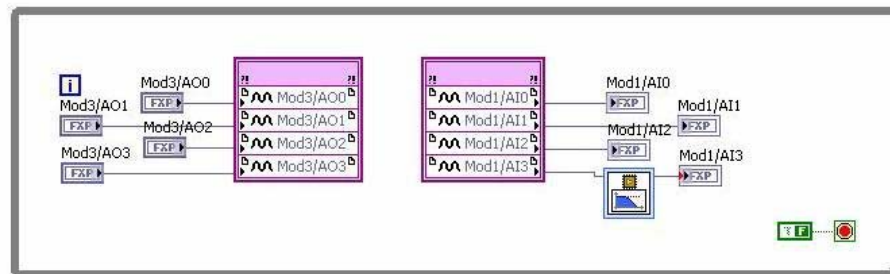
การควบคุม



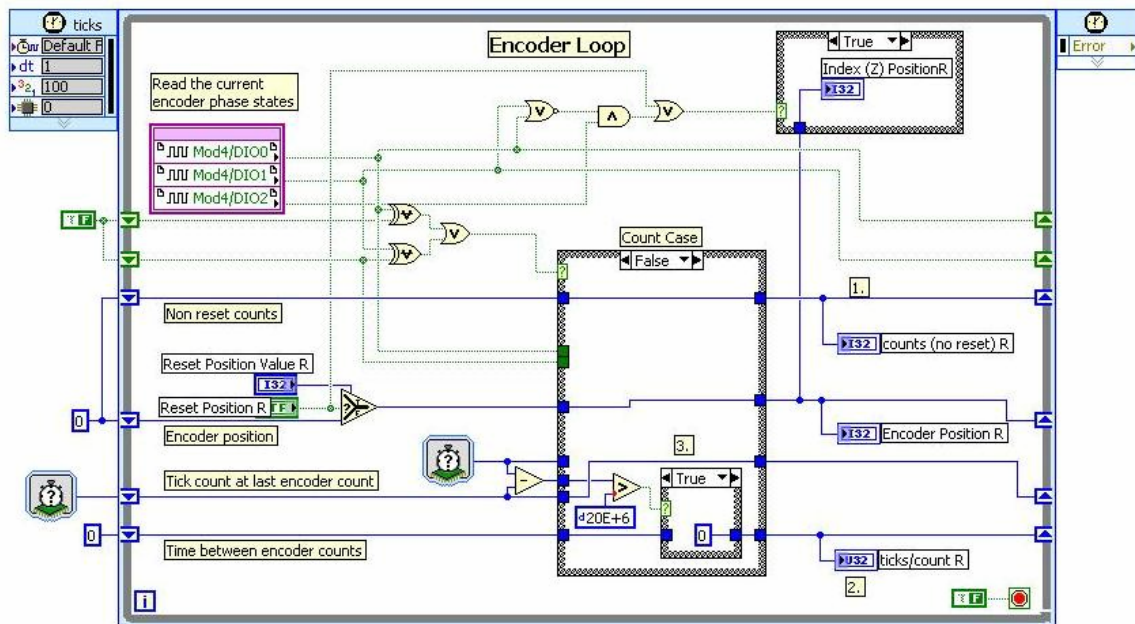
รูปที่ ก.1 โค้ดทั้งหมดที่อยู่ในส่วนของ FPGA



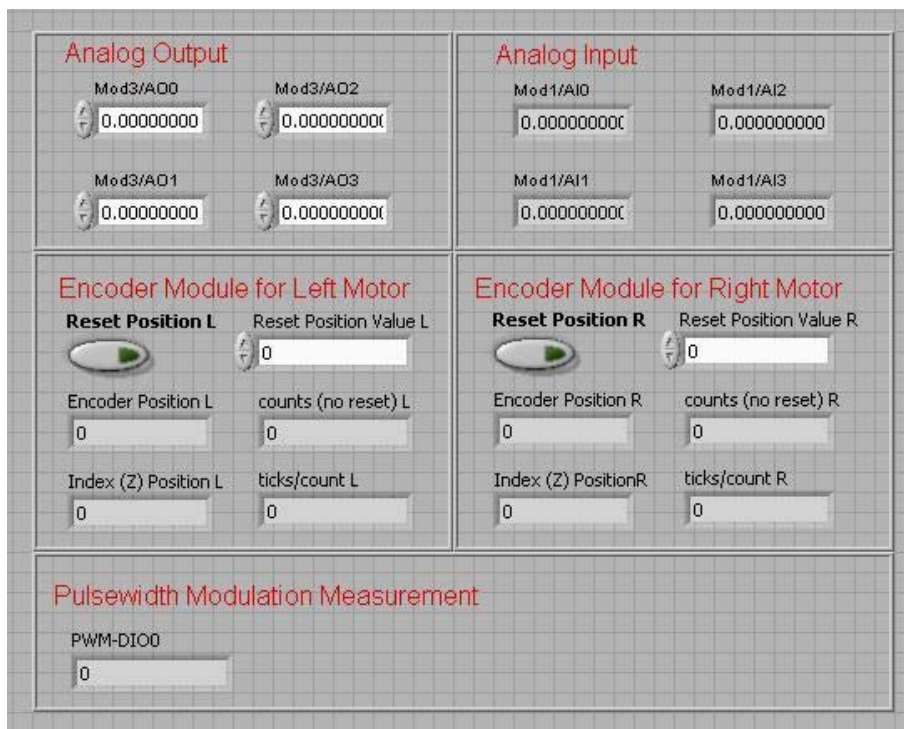
รูปที่ ก.2 โค้ดวัดสัญญาณ PWM



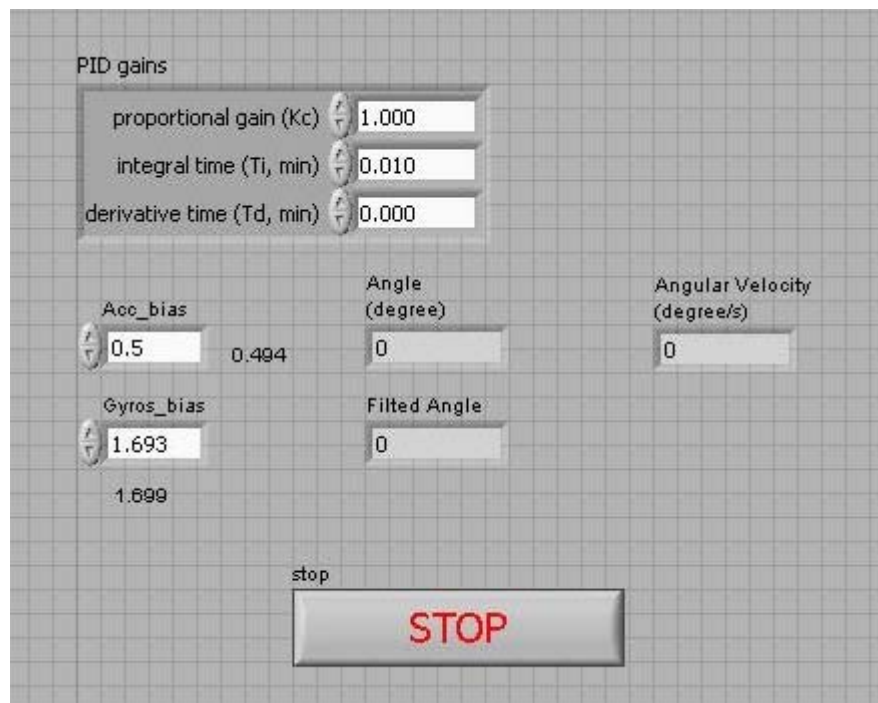
รูปที่ ก.3 โค้ดวัดและสร้างสัญญาณ Analog



รูปที่ ก.4 โค้ดวัดสัญญาณจาก Encoder



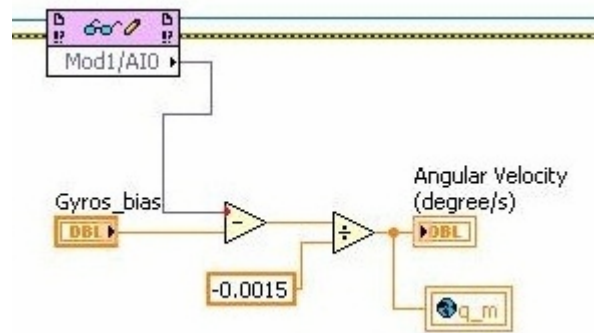
รูปที่ ก.5 Front panel ของโค้ดในส่วนของ FPGA



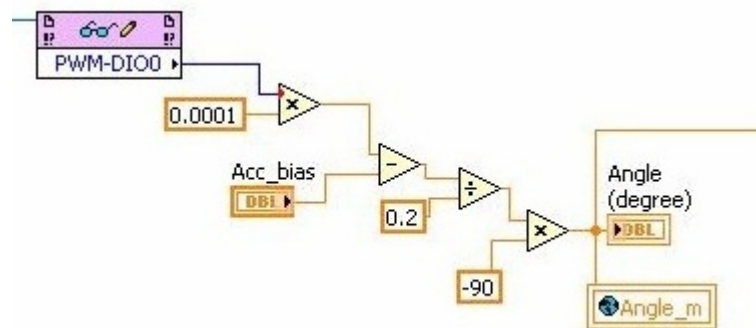
รูปที่ ก.6 Front panel ของโค้ดในส่วนของ Target







รูปที่ ก.9 โค้ดวัดค่า Angular velocity ในส่วนของ Target



รูปที่ ก.10 โค้ดวัดค่า Tilt angle ในส่วนของ Target

