

คู่มือปฏิบัติการวิศวกรรมไฟฟ้าและคอมพิวเตอร์

การควบคุมเซอร์โวมอเตอร์ด้วย PID

(PID for Servo Motor Control)

1. วัตถุประสงค์

1. เพื่อให้มีความรู้ความเข้าใจในระบบควบคุมแบบ PID
2. สามารถประยุกต์โปรแกรมคอมพิวเตอร์ เพื่อใช้ในการควบคุมระบบมอเตอร์

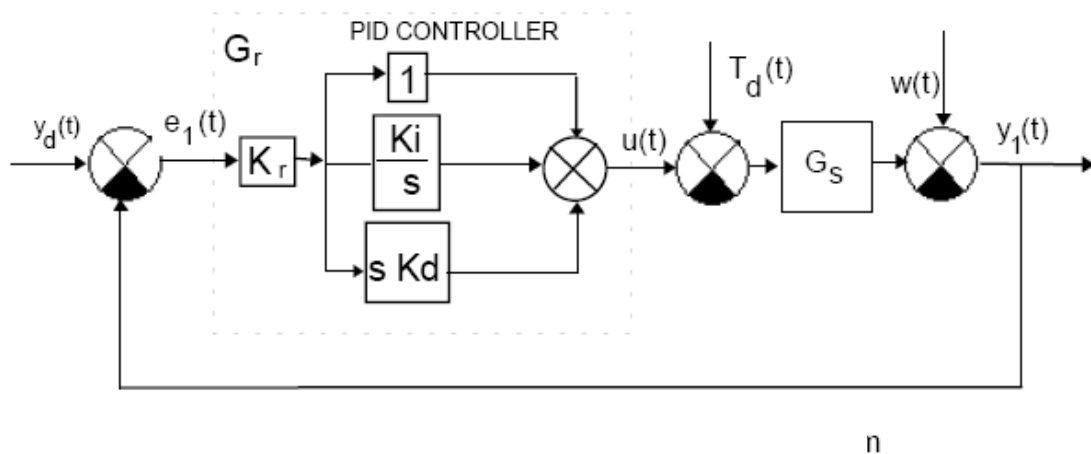
2. อุปกรณ์ที่ใช้ในการทดลอง

1. ชุดคอมพิวเตอร์และชุดอุปกรณ์เชื่อมต่อกับModular Servo System
2. โปรแกรมแมทแลบ 5.3

3. ทฤษฎีพื้นฐาน

3.1. การควบคุม PID แบบต่อเนื่อง (Continuous PID Controller)

กล่องแผนภาพ ของระบบควบคุม PID แบบต่อเนื่อง ของ servo motor ดังแสดงในรูปที่ 1



รูปที่ 1 กล่องแผนภาพของระบบควบคุม PID

โดยที่

- $y_d(t)$ คือ สัญญาณขาเข้า (input)
 $e(t)$ คือ ค่าความผิดพลาด (error)
 $y_1(t)$ คือ สัญญาณขาออก (output)
 $T_d(t)$ คือ สัญญาณรบกวนจากแรงบิดภายนอก
 $w(t)$ คือ สัญญาณรบกวนจากอุปกรณ์ตรวจจับ

ฟังก์ชันถ่ายโอน (Transfer function) ของการควบคุม PID คือ

$$G_r(s) = \frac{U(s)}{E_1(s)} = K_r \left(1 + \frac{K_i}{s} + sK_d \right)$$

ฟังก์ชันถ่ายโอน (Transfer function) ของมอเตอร์กระแสตรง (DC-Motor) คือ

$$G_s(s) = \frac{Y_1(s)}{U(s)} = \frac{K_s}{s(sT_s + 1)}$$

ฟังก์ชันถ่ายโอนของระบบ servo สามารถคำนวณได้จาก

$$G(s) = \frac{Y_1(s)}{Y_d(s)} = \frac{G_r(s)G_s(s)}{1 + G_r(s)G_s(s)}$$

ฟังก์ชันถ่ายโอนของค่าความผิดพลาดของระบบ servo คือ

$$G_e(s) = \frac{E_1(s)}{Y_d(s)} = \frac{1}{1 + G_s(s)G_r(s)}$$

ในสูตรของฟังก์ชันถ่ายโอน $G_r(s)$ ตัวแปรที่ผู้ใช้งานสามารถปรับแต่งได้คือ

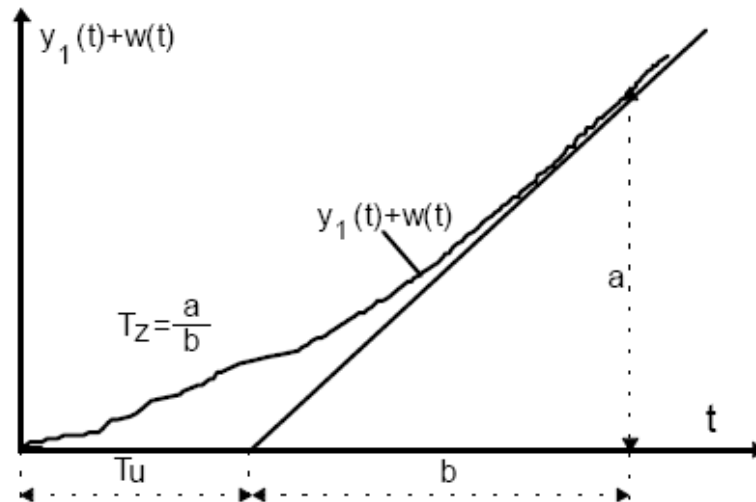
K_r - gain coefficient

K_i - integral coefficient

K_d - derivation coefficient

3.1.1. ระเบียบวิธี Ziegler-Nichols

รูปที่ 2 แสดงผลตอบสนองของฟังก์ชันแบบขั้นบันได (Step response) $y_1(t)$ (รวมสัญญาณรบกวน) ของมอเตอร์กระแสตรง (DC-motor)



รูปที่ 2 ผลตอบสนองของฟังก์ชันแบบขั้นบันไดของมอเตอร์กระแสตรงและตัวแปร
จาก Ziegler-Nichols

Ziegler-Nichols เป็นระเบียบวิธีสำหรับการปรับแต่งตัวแปรของการควบคุม PID สำหรับกระบวนการทดลองอย่างง่าย ซึ่งมีอยู่ด้วยกันสองวิธี คือ ระเบียบวิธี Ultimate-Sensitivity และ ระเบียบวิธี Transient-Response ซึ่งจะได้อธิบายในภายหลัง

สำหรับการใช้ระเบียบวิธี Ziegler-Nichols โมเดลอย่างง่ายสำหรับมอเตอร์กระแสตรง คือ

$$G_Z(s) = \frac{T_Z e^{-sT_u}}{s}$$

เมื่อ T_u คือ เวลาหน่วง (time delay)

T_Z คือ ค่าคงที่ Integration (ดูจากรูปที่ 2)

จากการจำลองข้างต้นเป็นการประยุกต์ใช้ระเบียบวิธี Ziegler-Nichols โดยกำหนดตัวแปรต่างๆ ดังนี้

❖ เมื่อสมมติให้ระบบมี Overshoot เป็น 2% ถึง 5%

การควบคุมแบบ P : $K_r = \frac{0.5}{T_Z T_u}$

$$\text{การควบคุมแบบ PI} \quad : \quad K_r = \frac{0.5}{T_z T_u} \quad K_i = \frac{1}{5T_u}$$

$$\text{การควบคุมแบบ PID} \quad : \quad K_r = \frac{0.65}{T_z T_u} \quad K_i = \frac{1}{5T_u} \quad K_d = 0.23T_u$$

❖ เมื่อสมมติให้ระบบมี Overshoot เท่ากับ 20 %

$$\text{การควบคุมแบบ P} \quad : \quad K_r = \frac{0.7}{T_z T_u}$$

$$\text{การควบคุมแบบ PI} \quad : \quad K_r = \frac{0.5}{T_z T_u} \quad K_i = \frac{1}{3T_u}$$

$$\text{การควบคุมแบบ PID} \quad : \quad K_r = \frac{1.1}{T_z T_u} \quad K_i = \frac{1}{2T_u} \quad K_d = 0.37T_u$$

โดยที่ T_z และ T_u สามารถวัดได้จาก step response ของมอเตอร์กระแสตรง (ดูได้จากรูปที่ 2)

3.1.2. ระเบียบวิธี Integral Square Error

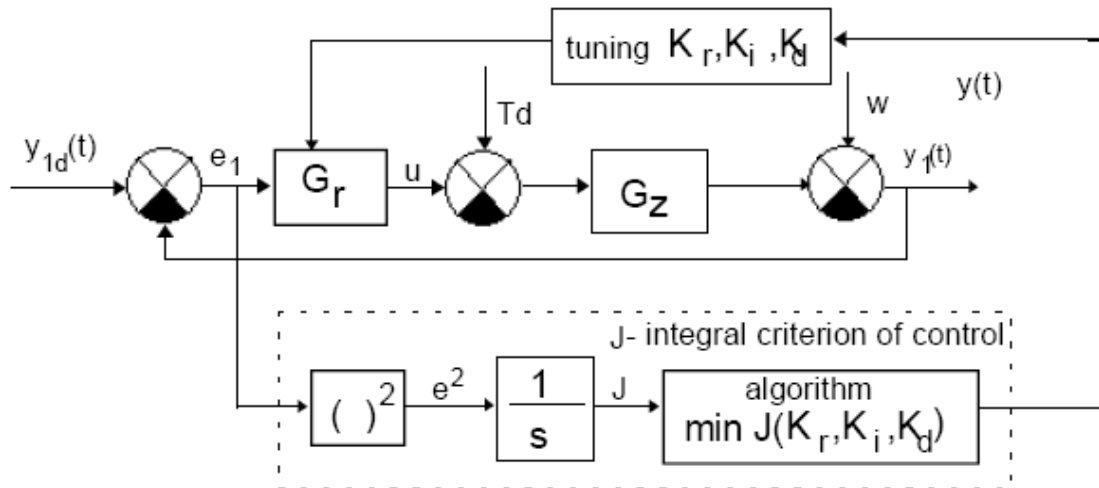
รูปที่ 3 แสดงแนวความคิดทั่วไปของระเบียบวิธี Integral Square การปรับแต่งตัวแปรจากรูปที่ 2 จะนำระเบียบวิธี Ziegler-Nichols มาประยุกต์ใช้ด้วยดังนี้

จากความสัมพันธ์

$$J(K_r, K_i, K_d) = \int_0^{\infty} e_1^2 dt$$

ในการวิเคราะห์ผลเฉลยสำหรับแก้ปัญหาเพื่อหาค่าที่เหมาะสมที่สุด (Optimization)

$$\min \int_0^{\infty} e_1^2 dt \rightarrow K_r, K_i, K_d \text{ สามารถหาค่าที่เหมาะสมได้}$$



รูปที่ 3 ระเบียบวิธี Integral Square Error

ในกรณีนี้ ค่าตัวแปรที่ดีที่สุดในการควบคุม

การควบคุมแบบ P : $K_r = \frac{1}{T_z T_u}$

การควบคุมแบบ PI : $K_r = \frac{1}{T_z T_u}$ $K_i = \frac{1}{4.3 T_u}$

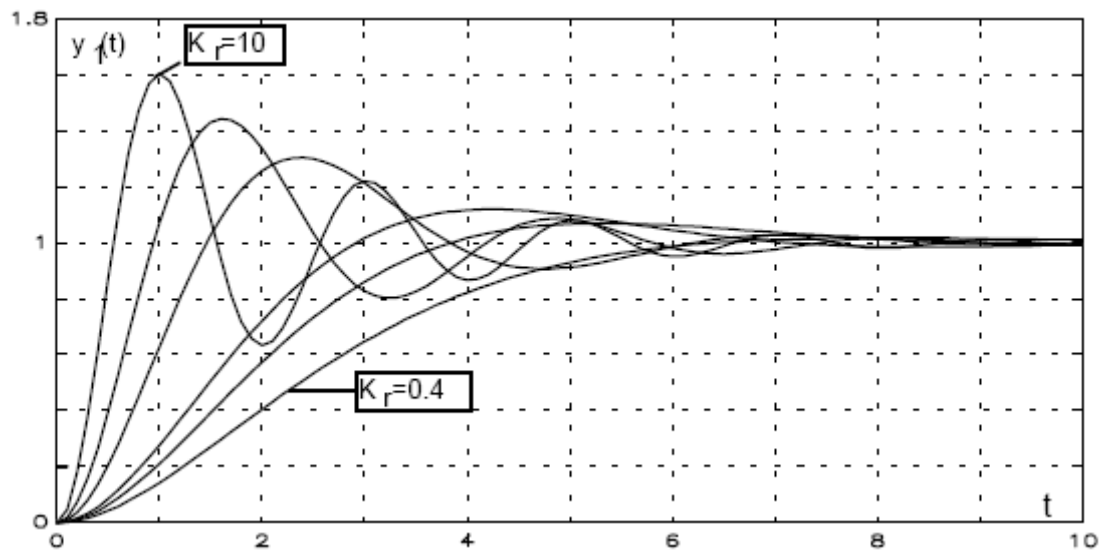
การควบคุมแบบ PID : $K_r = 1.3 \frac{1}{T_z T_u}$ $K_i = \frac{1}{1.6 T_u}$ $K_d = 0.5 T_u$

ถ้าแบบจำลองของ DC-motor ใช้ $G_s = \frac{K_s}{s(sT_s + 1)}$ ดังนั้นสามารถหาได้จากการคำนวณ

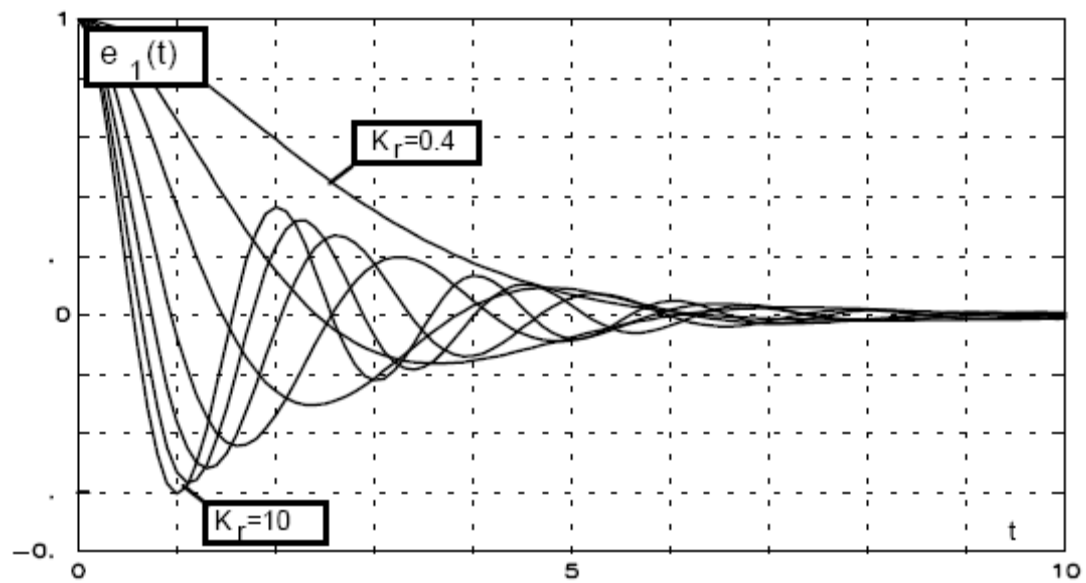
$$J(K_r, K_i, K_d) = \int_0^{\infty} e(t)^2 dt = \frac{(1 - K_s K_r T_s)}{2 K_i K_s K_r [T_s + T_i (K_r K_s K_d - 1)]}$$

ค่าตัวแปรสำหรับการควบคุม สามารถหาได้จากความสัมพันธ์ต่อไปนี้

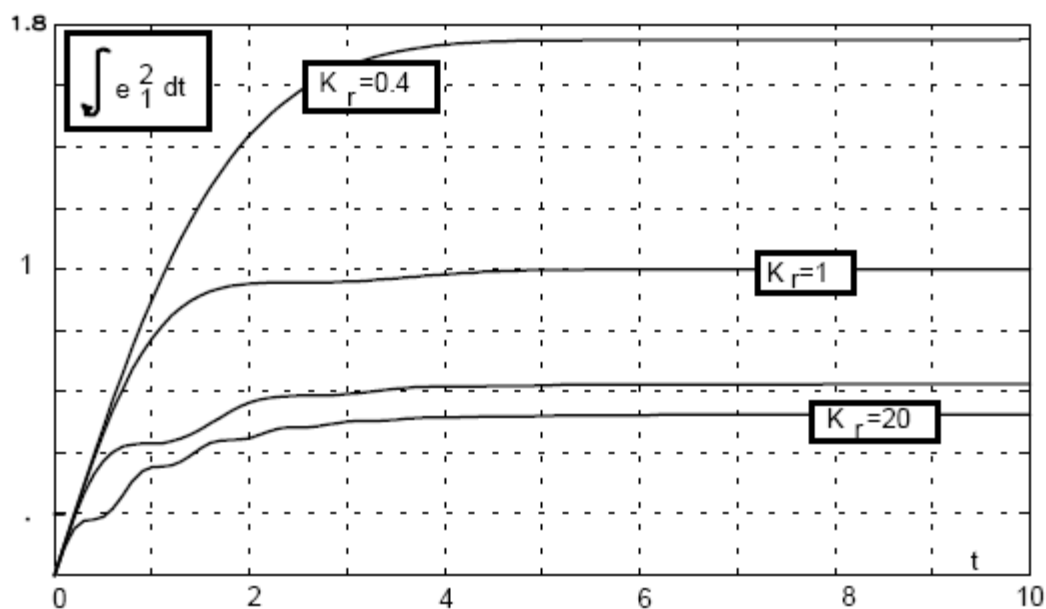
$$\frac{\partial J}{\partial K_r} = 0 \qquad \frac{\partial J}{\partial K_i} = 0 \qquad \frac{\partial J}{\partial K_d} = 0$$



รูปที่ 4 แสดงค่า (P) ควบคุมทาง output $y_1(t)$ โดยใช้ฟังก์ชันโอนย้ายมาตรฐานของ DC-motor ในการทดลอง ($T_s=1, K_s=1$)



รูปที่ 5 แสดงฟังก์ชันผิดพลาด $e(t)$ ของค่าเกน K_r ต่างๆ



รูปที่ ค่า Integral criterion [J] ที่เปลี่ยนแปลง จากค่าเกนค่าต่างๆ (P control)

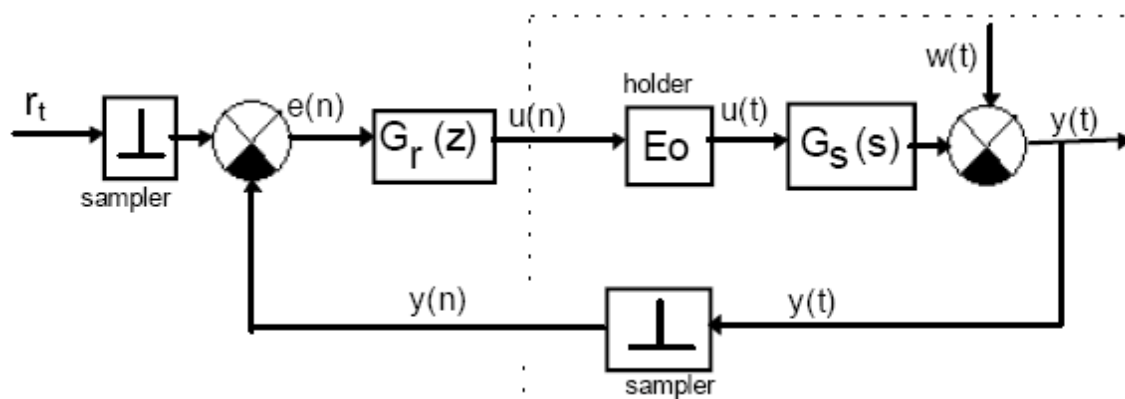
3.2. การควบคุม PID แบบดิจิตอล (Digital PID Controller)

ฟังก์ชันถ่ายโอนของระบบควบคุม PID แสดงอยู่ในรูปสมการอนุพันธ์และอินทิเกรต ซึ่งสามารถใช้การวิเคราะห์แบบไม่ต่อเนื่อง (Discrete) สำหรับใช้ประมาณค่าของการอนุพันธ์และการอินทิเกรต ได้

$$G_r(z) = K_r \left(1 + K_i \frac{z}{z-1} + K_d \frac{z-1}{z} \right)$$

โดยที่ $G_r(z)$ คือ ฟังก์ชันโอนย้ายของ PID controller.

แบบจำลองของ the digital control closed-loop system ในรูปที่ 7



รูปที่ 7 ระบบมอเตอร์กระแสตรงที่ใช้ระบบควบคุมแบบดิจิตอล

กฎ Ziegler-Nichols สามารถนำมาใช้ในการหาค่าเกณฑ์ของ digital PID controller โดยอาศัยการทดลองจากรูปที่ 2 (คำนวณภายใต้การสันนิษฐานที่ $T_u \geq 0.5T_o$)

การใช้วิธี Ziegler-Nichols สามารถหาค่าเกณฑ์ของระบบควบคุมแบบต่างๆ ได้ดังนี้

P digital controller

$$u_t = K_r(r_t - y_t)$$

$$K_r = \frac{1}{T_z(T_u + T_o)}$$

PI digital controller

$$u_t - u_{t-1} = K_r[(y_{t-1} - y_t)(r_{t-1} - r_t)]$$

$$K_r = \frac{1}{T_z(T_u + 0.5T_o)} - 0.5K_i$$

$$K_i = \frac{0.27T_o}{T_z(T_u + T_o)^2}$$

สำหรับ PID digital controller

$$u_t - u_{t-1} = K_r[(y_{t-1} - y_t) + K_i(r_t - y_t) + K_d(2y_{t-1} - y_t - y_{t-2})]$$

$$K_r = \frac{1}{T_z(T_u + T_z)} - 0.5K_i$$

$$K_i = \frac{0.6T_o}{T_z(T_u + 0.5T_o)^2}$$

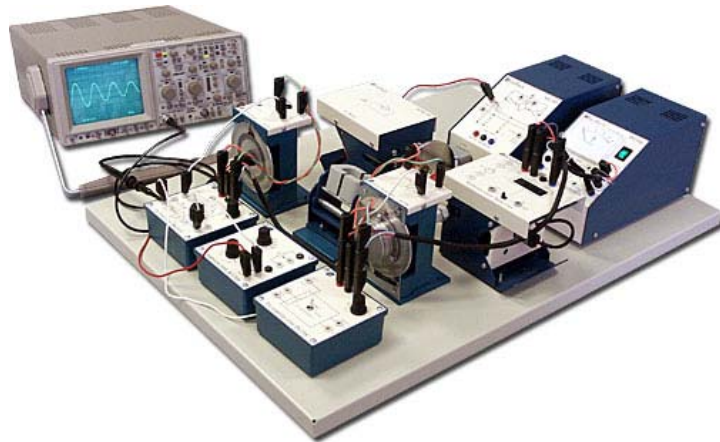
$$K_d = \frac{0.5}{T_z T_o} \quad \text{or} \quad K_d = \frac{0.6}{T_z T_o} \quad \text{if } (T_u * T_z) \text{ is an integer}$$

ข้อสังเกต : ค่าตัวแปรของ PID digital controller ขึ้นอยู่กับ Sampling time (T_o)

4. ขั้นตอนการทดลอง

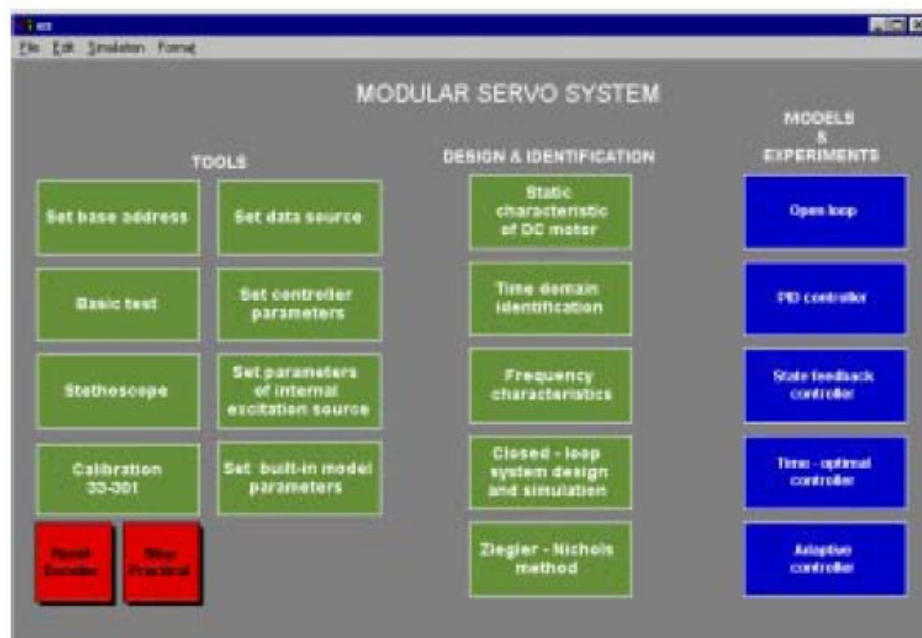
ตอนที่ 1 การระบุเอกลักษณ์ของระบบ และการหาค่าเกนของระบบควบคุม

1. ต่อชุดวงจรเซอร์โวมอเตอร์ ดังแสดงในรูปที่ 8 พร้อมเชื่อมต่อเข้ากับเครื่องคอมพิวเตอร์



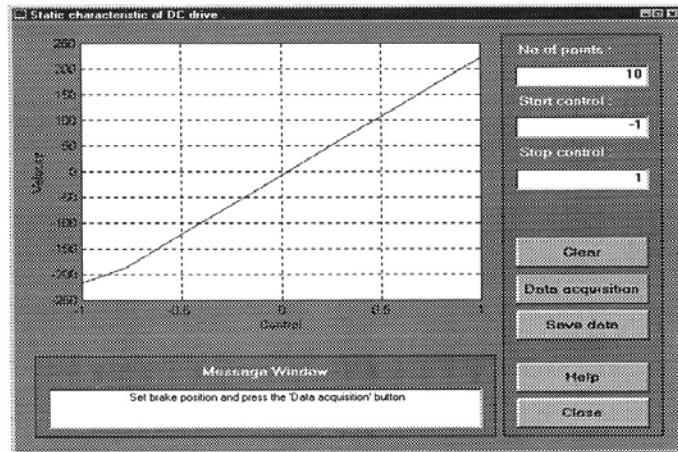
รูปที่ 8 ชุดวงจรเซอร์โวมอเตอร์

2. ปรับความเร็วมอเตอร์ไปที่ความเร็ว 1500 รอบ/นาที โดยปรับที่กล่อง Attenuator Unit AU150B
3. เปิดโปรแกรมเมทแล็บขึ้นมา โดยคลิกที่ Start → Program → MATLAB 5.3
4. บนหน้าต่างคำสั่ง (Command Windows) ใช้คำสั่ง es เพื่อเปิดหน้าต่าง Main Control Window ดังรูปที่ 9



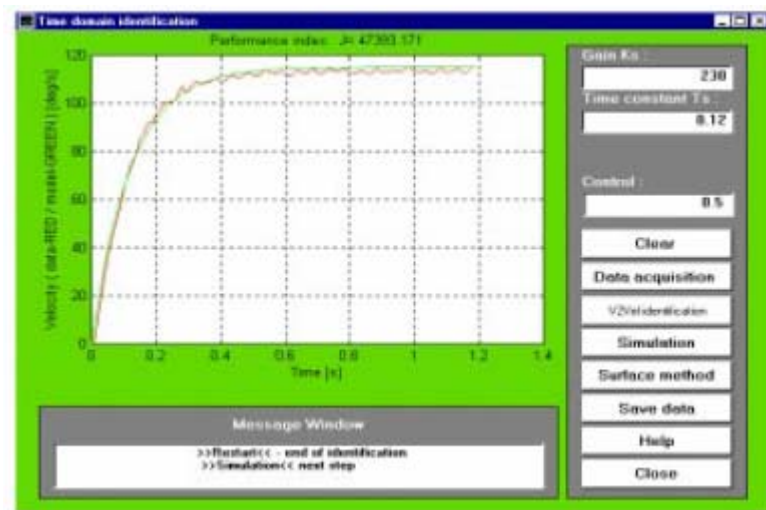
รูปที่ 9 Main Control Window

5. ทดสอบการควบคุมมอเตอร์ โดยดับเบิ้ลคลิกที่กล่อง **Static Characteristic of the DC motor** ซึ่งระบบจะทำงานเมื่อคลิกที่ปุ่ม **Data acquisition** สังเกตการเปลี่ยนแปลงที่เกิดขึ้นทั้งที่มอเตอร์ และที่หน้าจอคอมพิวเตอร์



รูปที่ 10 หน้าต่าง Static Characteristic of the DC motor

6. จากนั้นให้ดับเบิ้ลคลิกที่กล่องเครื่องมือ **Time domain identification** เพื่อระบุเอกลักษณ์ของระบบ ปรับค่า Control ไปที่ 1.0 จากนั้นให้คลิกที่ปุ่ม **Data acquisition** รอสักครู่จะปรากฏหน้าต่างที่มีกราฟเส้นสีแดงขึ้นมา จากนั้นให้ทำการปรับค่า **Gain (K_s)** และค่า **Time constant (T_s)** แล้วคลิกที่ปุ่ม **Simulation** จะมีกราฟสีแดงปรากฏขึ้น ให้ปรับค่าดังกล่าวไปเรื่อยๆจนกว่าลักษณะของกราฟทั้งสองเส้นจะใกล้เคียงกัน ดังรูปที่ 11



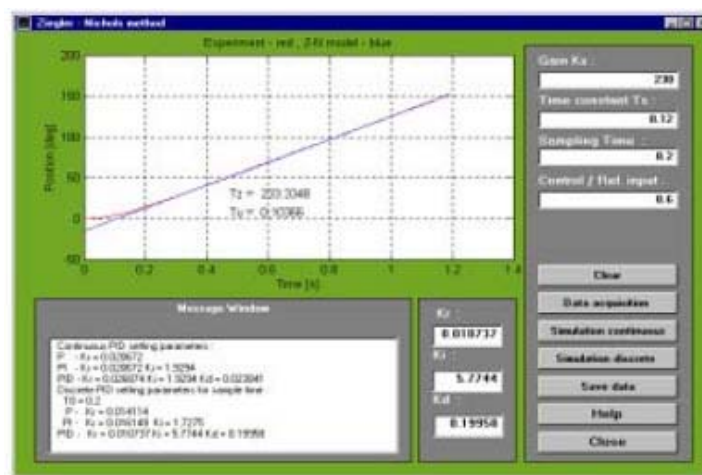
รูปที่ 11 หน้าต่างแสดงการหาโมเดลของระบบ

จากการทดสอบพบว่า มอเตอร์กระแสตรงที่ใช้ในการทดลองมีค่าต่างๆ เป็นดังนี้

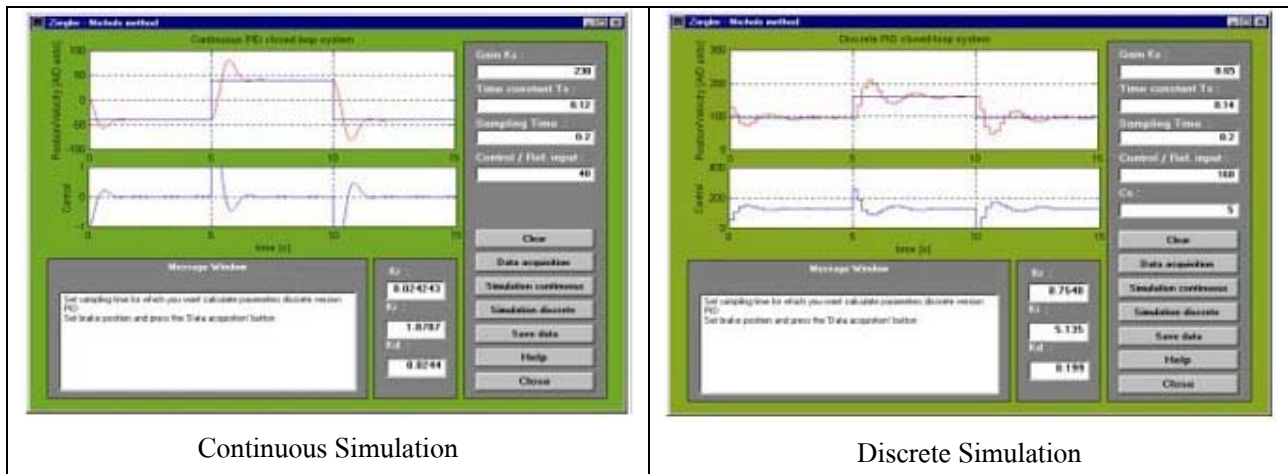
$K_s = \dots\dots\dots$ $T_s = \dots\dots\dots$

ฟังก์ชันโอนย้ายของระบบ คือ $\frac{1}{s} \cdot \frac{K_s}{(T_s \cdot s + 1)} = \dots\dots\dots$

7. ดับเบิลคลิกที่กล่องเครื่องมือ **Ziegler Nichols method** จะปรากฏหน้าต่างของกล่องเครื่องมือ Ziegler Nichols method ดังรูปที่ 9 แล้ววิธีการคือให้หาค่า Gain (K_s) และ ค่า Time constant (T_s) และป้อนค่าทั้งสองลงในช่อง จากนั้นคลิกที่ปุ่ม Data acquisition โปรแกรมจะเริ่มประมวลผลแล้ว จะเห็นผลตอบสนองขั้นต้นของระบบ และ straight line คือตัวอย่างของผลตอบสนองขั้นต้นของ Ziegler-Nichols จะมีค่าคงที่ 2 ตัวที่ถูกคำนวณออกมาจากกราฟ คือ T_z และ T_u (T_z คือ เส้นสัมผัสของ ตัวอย่างของผลตอบสนอง และ T_u คือระยะเวลาหนึ่งของ it[] จาก Ziegler-Nichols ค่าสัมประสิทธิ์ของ Continuous และ Discrete จะแยกกันคำนวณ ดูจาก Message Window



รูปที่ 10 หน้าต่าง Ziegler-Nichols method หลังจากการทดลองเสร็จสมบูรณ์



รูปที่ 11 Simulation PIDcontroller

บันทึกค่าต่างๆ ที่ได้จากการกระบวนกร Ziegler – Nichols เมื่อกำหนด Sampling Time = 0.01

- Continuous Simulation

ควบคุมแบบ P $K_r = \dots\dots\dots$

ควบคุมแบบ PI $K_r = \dots\dots\dots$ $K_i = \dots\dots\dots$

ควบคุมแบบ PID $K_r = \dots\dots\dots$ $K_i = \dots\dots\dots$ $K_d = \dots\dots\dots$

- Discrete Simulation

ควบคุมแบบ P $K_r = \dots\dots\dots$

ควบคุมแบบ PI $K_r = \dots\dots\dots$ $K_i = \dots\dots\dots$

ควบคุมแบบ PID $K_r = \dots\dots\dots$ $K_i = \dots\dots\dots$ $K_d = \dots\dots\dots$

ผลตอบสนองที่เกิดขึ้น มีลักษณะเป็นดังนี้

Continuous Simulation	Discrete Simulation

วิเคราะห์และสรุปผลการทดสอบตอนที่ 1

.....

.....

.....

.....

.....

.....

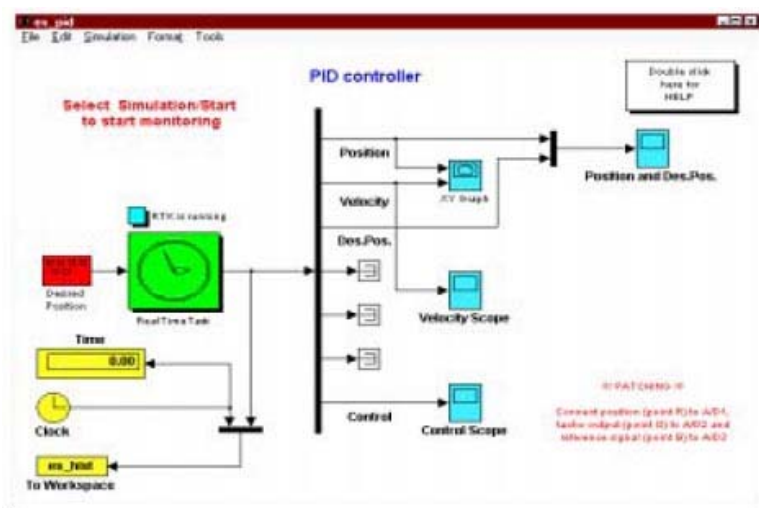
.....

ตอนที่ 2 การปรับแต่งค่าเกนของระบบควบคุม

ในความเป็นจริงนั้น ในโรงงานทั่วไปสายงานการผลิตจะไม่สามารถทำการปรับค่าพารามิเตอร์ K_p , K_i และ K_d ได้โดยตรง ต้องอาศัยการสร้างแบบจำลองบนคอมพิวเตอร์ เนื่องจากสายงานการผลิตต้องเดินเครื่องตลอดเวลาจึงต้องอาศัยหลักการของ Ziegler-Nichols method มาช่วยแก้ปัญหา

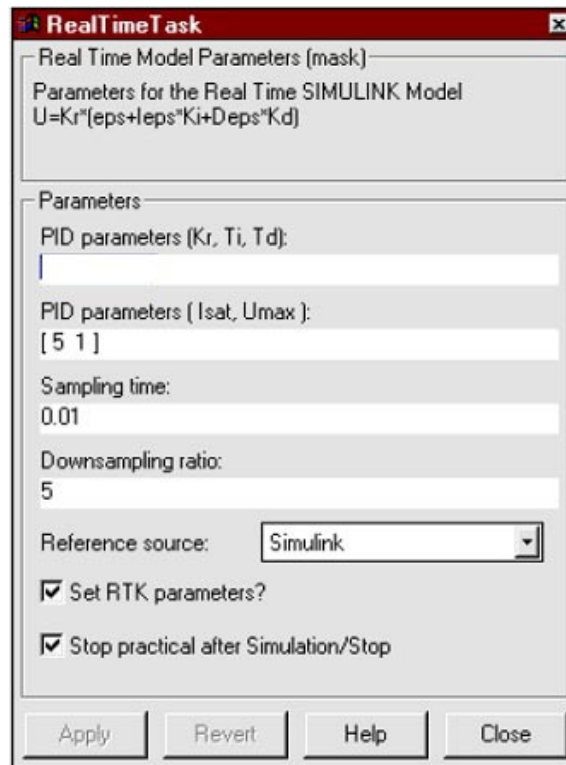
ในการทดลองนี้เราสามารถทำการปรับค่าพารามิเตอร์ต่างๆในกล่องเครื่องมือ PID Controller ได้เลย ซึ่งคล้ายกับการใช้หลักการของ Ziegler-Nichols เพราะชุดมอดูร์ได้ทำการต่อเข้ากับกล่องเครื่องมือ PID Controller เรียบร้อยแล้ว การประยุกต์และการตั้งค่าของระบบควบคุม PID สามารถทำโดยผู้ใช้ ซึ่งการปรับแต่งค่าเกนอาจต้องใช้การสังเกตผลตอบสนองที่เกิดขึ้นและอาจต้องใช้ประสบการณ์ในการช่วยหาค่าเกนที่เหมาะสมที่สุดในการควบคุมระบบ

1. กลับไปที่ Main Control Window และดับเบิลคลิกที่ปุ่ม PID Controller ดังรูปที่ 12



รูปที่ 12 หน้าต่าง PID Controller

2. ดับเบิลคลิกที่กล่องเครื่องมือ Real Time Task และลองปรับค่าพารามิเตอร์ K_p , K_i และ K_d ที่คำนวณได้จากตอนที่ 1 และตั้งค่า Sampling Time = 0.01 ดังรูปที่ 13



รูปที่ 13 Real Time Task Window

3. ดับเบิลคลิกที่กล่อง **Desire Position** เลือกลักษณะรูปแบบการควบคุมมอเตอร์ ให้มีการเปลี่ยนตำแหน่งในรูปแบบ **Square** ซึ่งมี Amplitude = 100 และ Frequency = 0.1 Hz
4. ก่อนทำการสตาร์ทที่ **Simulation** ให้เรียกหน้าต่างคำสั่ง (Command Windows) ขึ้นมาแล้วใช้คำสั่ง **Clear All** เพื่อลบคำสั่งที่ค้างอยู่ในตัวโปรแกรมก่อน
5. คลิกที่ปุ่ม Simulation → Start และดูผลของสัญญาณที่ได้ใน Position and Des. Pos.
6. ปรับค่าพารามิเตอร์ K_r , K_i และ K_d จนได้สัญญาณที่ใกล้เคียงหรือถูกต้องที่สุด ให้สังเกตการหมุนของมอเตอร์ และบันทึกผลการทดลอง

ควบคุมแบบ P $K_r = \dots\dots\dots$

ควบคุมแบบ PI $K_r = \dots\dots\dots$ $K_i = \dots\dots\dots$

ควบคุมแบบ PID $K_r = \dots\dots\dots$ $K_i = \dots\dots\dots$ $K_d = \dots\dots\dots$

ผลตอบสนองที่เกิดขึ้น มีลักษณะเป็นดังนี้

P Controller	PI Controller
PID Controller	

วิเคราะห์และสรุปผลการทดสอบตอนที่ 2

.....

.....

.....

.....

.....

.....

.....

คำถามท้ายการทดลอง

1. จงออกแบบตัวควบคุมแบบ PID ของระบบเซอร์โวมอเตอร์ โดยควบคุมมอเตอร์ให้มีลักษณะการทำงานเป็นสัญญาณฟันเลื่อย (Saw tooth) ขนาด Amplitude =100 และมีความถี่ 0.1 Hz

ควบคุมแบบ PID

$K_r = \dots\dots\dots$

$K_i = \dots\dots\dots$

$K_d = \dots\dots\dots$

ผลตอบสนองที่เกิดขึ้น มีลักษณะเป็นดังนี้

