



โปรแกรมส่วนต่อประสานกราฟิก
สำหรับระบบควบคุม

MATLAB GRAPHICAL USER INTERFACE

FOR CONTROL SYSTEM

นายนรรค ประทุม รหัส 50361392

ชื่อสมุดคณะวิศวกรรมศาสตร์
หน้าที่รับ.....	1.1 ผู้เชี่ยวชาญ
.....	2555
เลขทะเบียน.....	15729439
เลขเรียกหนังสือ.....	NS.
มหาวิทยาลัยนเรศวร ว.2662	

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2553



ใบรับรองปริญญาบัณฑิต

ชื่อหัวข้อโครงการ ໂປຣແກນສ່ວນຕ່ອປະສານກາຟິກກັບຜູ້ໃຊ້ສໍາຫຼັບຮະບນຄວນຄຸມ
ผู้ดำเนินโครงการ นายนเรศ ประทุม รหัส 50361392
ที่ปรึกษาโครงการ ดร.มุตติชา สงวนจันทร์
สาขาวิชา วิศวกรรมไฟฟ้า
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2553

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยแม่ฟ้าหลวง อนุมัติให้ปริญญาบัณฑิตบันนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

ที่ปรึกษาโครงการ

(ดร.มุตติชา สงวนจันทร์)

กรรมการ

(ดร.นิพัทธ์ จันทร์มนิธิ)

กรรมการ

(ดร.ศุภวรรณ พลพิทักษ์ชัย)

ชื่อหัวข้อโครงการ	โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับระบบควบคุม
ผู้ดำเนินโครงการ	นาชนเรศ ประทุม รหัส 50361392
ที่ปรึกษาโครงการ	คร.นุติชา สงฆ์จันทร์
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2553

บทคัดย่อ

ในการควบคุมอุปกรณ์ต่างๆ ระบบควบคุมมีเป็นส่วนที่สำคัญ อีกทั้งการหาผลตอบสนองที่เกี่ยวข้องกับระบบควบคุมนั้นหาได้ด้วยวิธีการที่บุ่งหากซับซ้อนและคำตอบที่ได้มานั้นไม่สามารถทราบได้ว่าถูกต้องหรือไม่ หากมีการนำไปใช้งานทำให้ระบบควบคุมผิดพลาดไปและเกิดความเสียหายต่ออุปกรณ์ได้ เพื่อลดความผิดพลาดดังกล่าว จึงมีความคิดที่ว่า ควรจะมีโปรแกรมที่นำมาตรวจสอบคำตอบของผลตอบสนองที่นำมาได้ว่าถูกต้องหรือไม่

โครงการนี้เป็นการสร้างโปรแกรมที่เกี่ยวกับการหาผลตอบสนองของระบบ ที่ทำให้ผู้ใช้สามารถใช้งานได้ง่ายขึ้น ลักษณะเหมือนการใช้งานโปรแกรมทั่วไป โดยจะวิเคราะห์ผลตอบสนองของระบบโดยอุปกรณ์ในรูปของกราฟทั้งในทางเวลาและทางความถี่

Project title MATLAB Graphical User Interface for Control System

Name Mr.Nares Pratoom ID. 50361392

Project advisor Miss. Mutita Songjun, Ph.D.

Major Electrical Engineering

Department Electrical and Computer Engineering

Academic year 2010

Abstract

Control system is the important part to control equipment and the response of control systems is compute very difficult. This project is creating a program to compute the response of control systems from the transfer function or state space. This program computes the response of control systems in time domain and frequency domain that allow user easily to use same use general program. The output of this program is the graphs response of control systems.

กิตติกรรมประกาศ

โครงการนี้สำเร็จสุล่อง ได้ด้วยดี ด้วยการช่วยเหลือและสนับสนุนจากหลายท่านด้วยกัน ซึ่งผู้เขียนขอขอบพระคุณดังต่อไปนี้

ขอขอบคุณ ดร.มุติตา สงจันทร์ อาจารย์ที่ปรึกษาโครงการ ใน การให้ความรู้ แนะนำ การออกแบบโปรแกรม และ แนะนำหนังสือที่เกี่ยวข้องในการค้นหาความรู้ ตลอดจนเวลาให้ คำแนะนำที่เป็นประโยชน์ ผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์ที่ดี และขอกราบขอบพระคุณ อีกทั้งสูง

ขอขอบคุณภาควิชาศัลปกรรม ไฟฟ้า และคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัย นเรศวร ที่ให้ความช่วยเหลือเรื่องเงินทุนในการจัดทำโครงการ อีกทั้งบัณฑิตวิชาความรู้ และอบรมสั่งสอนให้ผู้จัดทำเป็นคนดีของลังกawi

ขอขอบคุณเพื่อนๆทุกคนที่เคยให้ความช่วยเหลือ ให้กำลังใจ และให้คำปรึกษาในการทำ โครงการนี้จนสุดล่อง ได้อย่างดี

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ผู้ให้กำเนิดและเลี้ยงดูจนทำให้ผู้จัดทำมีวันนี้

คุณค่า และประโยชน์อันพิเศษมาก โครงการนี้ ทางผู้จัดทำขอขอบคุณทุกท่านไว ณ โอกาสนี้

นเรศ ประทุม

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์.....	ก
บทคัดย่อภาษาไทย.....	ข
บทคัดย่อภาษาอังกฤษ.....	ก
กิตติกรรมประกาศ	ก
สารบัญ.....	ก
สารบัญตาราง.....	ช
สารบัญรูป.....	ฉ

บทที่ 1 บทนำ.....	1
-------------------	---

1.1 ที่มาและความสำคัญของ โครงการ	1
1.2 วัตถุประสงค์ของ โครงการ	1
1.3 ขอบเขตของ โครงการ.....	2
1.4 ข้อตอนและแผนการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับจาก โครงการ	3
1.6 งบประมาณ.....	3

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	4
---	---

2.1 ฟังก์ชันถ่ายโอน (Transfer Function)	5
2.2 การหาฟังก์ชันถ่ายโอนของระบบควบคุมจากภาพล็อกไดอะแกรม	6
2.3 ปริภูมิสถานะ (State Space)	9
2.4 ความสัมพันธ์ระหว่างฟังก์ชันถ่ายโอน กับ สมการปริภูมิสถานะ	10
2.5 ผลตอบสนองของระบบ	11
2.5.1 ระบบควบคุมอันดับหนึ่ง (First-Order System)	12
2.5.2 ระบบควบคุมอันดับสอง (Second -Order Systems)	16
2.5.3 คุณลักษณะของผลตอบสนองชั่วขณะของระบบอันดับสอง	18
2.6 หาเสถียรภาพของระบบควบคุมเชิงเส้น	20

สารบัญ (ต่อ)

หน้า

2.6.1 เร้าท์ไฮอร์วิช (Routh-Hurwitz Criterion).....	20
2.7 การเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI)	35
2.7.1 หลักการเขียนโปรแกรมด้วย GUIDE	36
2.7.2 หลักการเขียนโปรแกรมโดยไม่ใช้ GUIDE	37
2.8 การหาผลตอบสนองทางความถี่ (เพิ่มเติม)	39
2.8.1 ไนคุวิสต์ (Nyquist plot).....	40
2.8.2 โบเด (Bode plot)	40
2.8.3 แผนภูมินิโกลส์ (Nichols plot)	41
 บทที่ 3 วิธีการดำเนินโครงการ.....	42
3.1 ฐานแบบการทำงานของโปรแกรม	42
3.2 การเขียนโปรแกรม	43
3.2.1 การเขียนโปรแกรมในหน้าต่างหลัก (หน้าต่างที่ 1).....	43
3.2.2 การเขียนโปรแกรมในหน้าต่างฟังก์ชันถ่ายโอน (หน้าต่างที่ 2)	49
3.2.3 การเขียนโปรแกรมในหน้าต่างปริภูมิสถานะ (หน้าต่างที่ 3).....	50
3.2.4 การเขียนโปรแกรมในหน้าต่างแสดงเร้าท์ไฮอร์วิช (หน้าต่างที่ 4)	51
3.2.5 การเขียนโปรแกรมในหน้าต่างแสดงช่วงเวลา (หน้าต่างที่ 5).....	51
3.2.6 การเขียนโปรแกรมในหน้าต่างแสดงค่าปริภูมิสถานะ (หน้าต่างที่ 6)	52
3.2.7 การเขียนโปรแกรมในหน้าต่างแสดงค่าปริภูมิสถานะ (หน้าต่างที่ 7)	53
3.2.8 การแปลงโปรแกรม (Compile) เพื่อนำไปใช้โดยไม่ต้องติดตั้งโปรแกรม MATLAB โดยใช้ deploytool.....	54
 บทที่ 4 ผลการดำเนินโครงการ	57
4.1 การเริ่มต้นการใช้งานโปรแกรม.....	57
4.2 โปรแกรมระบบควบคุมโดยการป้อนฟังก์ชันถ่ายโอนจากผู้ใช้	57

สารบัญ (ต่อ)

หน้า

4.3 โปรแกรมระบบควบคุมโครงการป้อนปริมาณสถานะจากผู้ใช้	61
4.4 การทำงานของหน้าต่างหลัก.....	64
4.5 การทดสอบความถูกต้องของโปรแกรม	70
บทที่ 5 สรุปผลการดำเนินโครงการ	78
5.1 ผลการดำเนินโครงการ	78
5.2 ปัญหาที่พบขณะปฏิบัติโครงการ	79
5.3 ข้อเสนอแนะ	79
เอกสารอ้างอิง	80
ประวัติผู้ดำเนินโครงการ.....	81

สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางการลดรูปเลือกໄດอจะแกรม	9
2.2 ตารางแสดงข้อกำหนดของพุนนาມ.....	26



สารบัญรูป

รูปที่	หน้า
2.1 แสดงบล็อกไดอะแกรมแบบมีส่วนป้อนกลับ	6
2.2 แสดงบล็อกไดอะแกรมเมื่อมีสัญญาณรบกวน	8
2.3 บล็อกไดอะแกรมแสดงระบบอันดับหนึ่ง	12
2.4 ภาพครุภูบล็อกไดอะแกรมระบบอันดับหนึ่ง	12
2.5 กราฟลักษณะการตอบสนองของระบบควบคุมอันดับหนึ่ง มีลักษณะของกราฟเป็นแบบเอ็กซ์บอร์ดที่มีความชันของกราฟที่ $t = 0$ คือ $\frac{1}{T}$	13
2.6 กราฟลักษณะการตอบสนองของระบบควบคุมอันดับหนึ่ง ต่อสัญญาณถ่วง	15
2.7 ภาพผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณอินพัลส์	16
2.8 กราฟแสดงผลตอบสนองของระบบอันดับสอง	19
2.9 บล็อกไดอะแกรมของระบบควบคุม	27
2.10 เวกเตอร์แสดงจำนวนโพลและเซิร์โร	28
2.11 แสดงทิศทางของโพลและเซิร์โร	29
2.12 แสดงทิศทางของโพลและเซิร์โร	30
2.13 แสดงทิศทางของโพลและเซิร์โร	30
2.14 แสดงเส้นทางเดินของราก	32
2.15 รูปแสดงเส้นทางของราก	34
2.16 ตัวอย่าง Nyquist plot	40
2.17 ตัวอย่าง Bode plot	41
2.18 ตัวอย่าง Nichols plot	41
3.1 ผังงานสำหรับขั้นตอนการทำงานของโปรแกรม	42
3.2 การเรียกใช้ GUIDE ของ GUI	43
3.3 หน้าต่างของโปรแกรม GUIDE เริ่มต้น	44
3.4 หน้าตาโปรแกรมหลักที่ออกแบบ	45
3.5 คำสั่งเรียกใช้การแก้ไขคุณสมบัติ	45
3.6 การแก้ไขคุณสมบัติของแต่ละวัตถุ	46
3.7 การแก้ไขฟังก์ชันเรียกกลับ	48
3.8 แก้ไขคำสั่งการทำงานต่างๆ ของปุ่มที่สร้างขึ้น	48

สารบัญรูป (ต่อ)

ข้อที่	หน้า
3.9 หน้าต่างของฟังก์ชันถ่ายโอน	49
3.10 หน้าต่างของปริภูมิสถานะ	50
3.11 หน้าต่างแสดงการหาสедีบรภาพของเร้าท์เชอร์วิช	51
3.12 หน้าต่างแสดงการซ่วยเหลือ	52
3.13 หน้าต่างแสดงปริภูมิสถานะ	53
3.14 หน้าต่างแสดงปริภูมิสถานะ	53
3.15 วิธีการตั้งค่าตัวแปรโปรแกรม	54
3.16 ค่าตัวแปรโปรแกรม	55
3.17 การแปลงโปรแกรมให้ทำงานบนพื้นฐาน Window	55
3.18 การเพิ่มไฟล์ต่างๆของโปรแกรม	56
3.19 ไฟล์ที่ได้จากการแปลงโปรแกรม	56
4.1 หน้าต่างหลักของโปรแกรมเมื่อถูกเรียกขึ้นมา	58
4.2 หน้าต่างของฟังก์ชันถ่ายโอนเมื่อถูกเรียกขึ้นมา	59
4.3 แสดงผลเมื่อมีการใส่ค่าฟังก์ชันถ่ายโอน	59
4.4 หน้าต่างช่วยเหลือสำหรับการใช้งานการป้อนฟังก์ชันถ่ายโอน	60
4.5 หน้าต่างตัวอย่างการป้อนค่าฟังก์ชันถ่ายโอน	60
4.6 หน้าต่างของปริภูมิสถานะเมื่อถูกเรียกขึ้นมา	62
4.7 หน้าต่างหลักหลังป้อนค่าปริภูมิสถานะ	62
4.8 หน้าต่างช่วยเหลือสำหรับการใช้งานหน้าต่างป้อนค่าปริภูมิสถานะ	63
4.9 หน้าต่างตัวอย่างการป้อนค่าปริภูมิสถานะ	63
4.10 โปรแกรมแสดงกราฟของผลตอบสนองเมื่อปุ่ม Unit Step ถูกกด	64
4.11 โปรแกรมแสดงกราฟของผลตอบสนองเมื่อปุ่ม Impulse ถูกกด	65
4.12 โปรแกรมแสดงกราฟของผลตอบสนองเมื่อปุ่ม Unit Ramp ถูกกด	65
4.13 โปรแกรมแสดงเส้นทางเดินของรากเมื่อปุ่ม Root locus ถูกกด	66
4.14 โปรแกรมแสดงตารางของเร้าท์เชอร์วิชและจำนวนโพล เมื่อปุ่ม Routh Hurwitz ถูกกด	66

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.15 โปรแกรมแสดงผลตอบสนองทางความถี่ของโบดีเมื่อปุ่ม Bode ถูกกด	67
4.16 โปรแกรมแสดงผลตอบสนองทางความถี่ของนิโคลส์เมื่อปุ่ม Nichols ถูกกด	67
4.17 โปรแกรมแสดงผลตอบสนองทางความถี่ของไนคิวิสท์เมื่อปุ่ม Nyquist ถูกกด	68
4.18 โปรแกรมแสดงผลการแปลงฟังก์ชันถ่ายโอนเป็นปริภูมิสถานะ	69
4.19 โปรแกรมหลังจากกดปุ่ม Clear all	69
4.20 กราฟผลตอบสนองของสัญญาณอินพัลส์ที่ได้จากโปรแกรม	70
4.21 ตัวอย่างสำหรับเปรียบเทียบผลตอบสนองของสัญญาณอินพัลส์	70
4.22 กราฟผลตอบสนองของสัญญาณระดับที่ได้จากโปรแกรม	71
4.23 ตัวอย่างสำหรับเปรียบเทียบผลตอบสนองของสัญญาณระดับ	71
4.24 กราฟแสดงผลเส้นทางเดินของรถที่ได้จากโปรแกรม	72
4.25 ตัวอย่างสำหรับเปรียบเทียบเส้นทางเดินของรถของฟังก์ชันถ่ายโอน	72
4.26 กราฟแสดงผลตอบสนองทางความถี่ด้วยวิธีโนเมเดของฟังก์ชันถ่ายโอน	73
4.27 ตัวอย่างเปรียบเทียบผลตอบสนองทางความถี่ของฟังก์ชันถ่ายโอนด้วยวิธีโนเมเด	73
4.28 กราฟแสดงผลตอบสนองทางความถี่ด้วยวิธีนิโคลส์ของฟังก์ชันถ่ายโอน	74
4.29 ตัวอย่างเปรียบเทียบผลตอบสนองทางความถี่ของฟังก์ชันถ่ายโอนด้วยวิธีนิโคลส์	74
4.30 กราฟแสดงผลตอบสนองทางความถี่ด้วยวิธีไนคิวิสท์ของฟังก์ชันถ่ายโอน	75
4.31 ตัวอย่างเปรียบเทียบผลตอบสนองทางความถี่ของฟังก์ชันถ่ายโอนด้วยวิธีไนคิวิสท์	75
4.32 ภาพแสดงการป้อนค่าปริภูมิสถานะ	76
4.33 ภาพแสดงหน้าต่างโปรแกรมหน้าหลักการแปลงปริภูมิสถานะไปเป็นฟังก์ชันถ่ายโอน	76
4.34 ภาพแสดงการแปลงปริภูมิสถานะไปเป็นฟังก์ชันถ่ายโอน	77

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

สืบเนื่องจากในระบบควบคุม (Control System) การหาค่าของฟังก์ชันต่างๆ อาทิเช่น ฟังก์ชันถ่ายโอน (Transfer Function) ของระบบ และการหาตัวพารามิเตอร์ต่างๆ ของระบบ รวมไปถึงการหาผลเฉลยหรือผลตอบสนองของระบบทางด้านเวลาและทางด้านความถี่ไม่ว่าจะเป็นระบบอันดับหนึ่งหรือระบบอันดับสองนั้น มีข้อดีอย่างหนึ่งคือสามารถใช้วิธีการทำที่บุญชากและซับซ้อนหากมีการคำนวณด้วยมือ และอาจจะทำให้ค่าที่ได้มานาเกิดความผิดพลาดขึ้นได้ ซึ่งถ้าเราจำเป็นต้องนำผลลัพธ์ที่ได้จากการคำนวณไปทำการวิเคราะห์ต่อ ก็จะทำให้ได้ผลของการวิเคราะห์ที่ผิดพลาด และทำให้การนำไปใช้ไม่เกิดประโยชน์ ดังนั้นเพื่อที่จะทำให้ได้ผลลัพธ์ที่ถูกต้องและแม่นยำและเพิ่มความสะดวกในการคำนวณ วิธีที่น่าสนใจที่สุดคือการสร้างโปรแกรมขึ้นมา เพื่อกำหนดและแสดงผลตอบสนองของระบบที่ผู้ใช้ต้องการและหาคำตอบที่จำเป็นออกมานะ

ดังที่ได้กล่าวมาแล้ว ปริญญา妮พนธ์บันนี้จึงได้เสนอวิธีการเขียนโปรแกรมดังกล่าว โดยใช้การเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphical User Interface, GUI) ของโปรแกรมแม่ทแเรน (MATLAB) รับค่าจากผู้ใช้และคำนวณหาค่าอອกมา โดยปริญญา妮พนธ์นี้ มุ่งเน้นไปในทางการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ ในการเขียนโปรแกรมนี้ ขึ้นมา

1.2 วัตถุประสงค์ของโครงการ

- 1) เพื่อใช้โปรแกรมที่ได้เขียนขึ้นมา นำไปตรวจสอบคำตอบผลตอบสนองของระบบควบคุมที่ได้จากการคำนวณด้วยมือเพื่อให้มีความถูกต้องแม่นยำ
- 2) เพื่อศึกษาการเขียนโปรแกรมแม่ทแเรนในส่วนของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ ให้มีความชำนาญมากขึ้น
- 3) เพื่อนำความรู้ที่ได้ไปประยุกต์ใช้กับระบบอื่นๆ

1.3 ขอบเขตของโครงการ

- 1) ศึกษาหาคุณลักษณะของระบบควบคุมในส่วนต่างๆที่จำเป็นต้องใช้ในการคำนวณ
 - 2) ศึกษาการเขียนโปรแกรมส่วนประสาณต่อประสาณกราฟิกกับผู้ใช้ของโปรแกรมแม่ทัพเดิบ
 - 3) แสดงการคำนวณหาค่าของโปรแกรมที่เขียน ได้อย่างถูกต้องแม่นยำ
 - 4) ศึกษาประโยชน์และความเป็นไปได้ในการนำความรู้โปรแกรมแม่ทัพเดิบ ในส่วนของโปรแกรมส่วนต่อประสาณกราฟิกกับผู้ใช้ ไปใช้ในงานด้านอื่นๆ

1.4 รับผิดชอบและแผนการดำเนินงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงการ

- 1) มีความรู้ความชำนาญในการใช้โปรแกรม MATLAB ได้ดีขึ้น
- 2) เข้าใจและทราบถึงทฤษฎีในระบบควบคุมให้แน่นอนมากขึ้น
- 3) นำความรู้จากการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ไปพัฒนาในงานค้านอ่นๆ

1.6 งบประมาณ

1) ค่าหนังสือ	400 บาท
2) ค่าจัดทำเอกสาร	400 บาท
3) ค่าพิมพ์เอกสาร	200 บาท
รวมทั้งสิ้น (หนึ่งพันบาทถ้วน)	<u>1000</u> บาท

หมายเหตุ : ถ้าเฉลี่ยหุกรายการ

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

เนื่องจากในการเขียนโปรแกรม ด้วยการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI) จำเป็นต้องมีความรู้ในเรื่องของระบบควบคุม (Control system) พื้นฐาน เช่น ฟังก์ชันถ่ายโอน (Transfer function), ปริภูมิสถานะ (State Space) เป็นต้น และยังจำเป็นต้องมีความรู้เรื่องการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ด้วย ดังนั้นในบทนี้จะอธิบายเรื่องระบบควบคุม และการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ขั้นพื้นฐาน ดังนี้

- 1) ฟังก์ชันถ่ายโอน
- 2) การหาฟังก์ชันถ่ายโอนของระบบควบคุมจากภาพเดล็อกໄโคะแกรน
- 3) ปริภูมิสถานะ
- 4) ความสัมพันธ์ระหว่างฟังก์ชันถ่ายโอน กับ สมการปริภูมิสถานะ
- 5) ผลตอบสนองของระบบ
 - ก) ผลตอบสนองของระบบควบคุมอันดับหนึ่ง
 - ผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณระดับ
 - ผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณผลัด
 - ผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณอินพัลส์
 - ข) ผลตอบสนองของระบบควบคุมอันดับสอง
 - กรณีระบบที่มีความหน่วงน้อย
 - ระบบที่มีความหน่วงมาก
 - ค) กรณีระบบที่มีความหน่วงมาก
 - คุณลักษณะของผลตอบสนองชั่วขณะของระบบอันดับสอง
 - ค่าความคลาดเคลื่อนเชิงสถิติ

6) หาสติบราชการของระบบความคุณเชิงเส้น

ก) เร้าท์เซอร์วิช

ข) เส้นทางเดินของราก (Root locus)

7) การเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI)

ก) หลักการเขียนโปรแกรมด้วยตัวช่วยพัฒนาส่วนต่อประสานกราฟิกกับผู้ใช้ (GUIDE)

ข) หลักการเขียนโปรแกรมโดยไม่ใช้ตัวช่วยพัฒนาส่วนต่อประสานกราฟิกกับผู้ใช้ (GUIDE)

2.1 ฟังก์ชันถ่ายโอน (Transfer Function)

นิยามของฟังก์ชันถ่ายโอน : ฟังก์ชันถ่ายโอนเป็นอัตราส่วนของสมการทางค้านเอาต์พุต และสมการทางค้านอินพุตที่ถูกแปลงโดยลากป้ายแล้ว ภายใต้เงื่อนไขค่าเริ่มต้นเป็นศูนย์ (Initial Condition = 0)

เมื่อพิจารณาสมการเชิงอนุพันธ์ของระบบแบบไม่แปรเปลี่ยนตามเวลา

$$a_0 y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} \dot{y} + a_n y = b_0 x^{(m)} + b_1 x^{(m-1)} + \dots + b_{m-1} \dot{x} + b_m x \quad (2.1)$$

โดย $n \geq m$

กำหนดให้ y เป็นเอาต์พุตของระบบ และ x เป็นอินพุตของระบบ ฟังก์ชันถ่ายโอนของระบบสามารถหาได้โดยการแปลงลากป้ายสมการทั้งสองข้าง ภายใต้เงื่อนไขเริ่มต้นเป็นศูนย์

$$\mathcal{L} [a_0 y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} \dot{y} + a_n y] = \mathcal{L} [b_0 x^{(m)} + b_1 x^{(m-1)} + \dots + b_{m-1} \dot{x} + b_m x] \quad (2.2)$$

ทำให้ได้

$$Y(s) [a_0 s^{(n)} + a_1 s^{(n-1)} + \dots + a_{n-1} s + a_n] = X(s) [b_0 s^{(m)} + b_1 s^{(m-1)} + \dots + b_{m-1} s + b_m] \quad (2.3)$$

หากความสัมพันธ์ของสมการเอาต์พุตกับสมการอินพุตได้

$$\text{ฟังก์ชันถ่ายโอน} = G(s) = \frac{\mathcal{L}(\text{output})}{\mathcal{L}(\text{input})}$$

$$\frac{Y(s)}{X(s)} = \frac{[b_0 s^{(m)} + b_1 s^{(m-1)} + \dots + b_{m-1} s + b_m]}{[a_0 s^{(n)} + a_1 s^{(n-1)} + \dots + a_{n-1} s + a_n]} \quad (2.4)$$

ขั้นตอนการหาฟังก์ชันถ่ายโอน

เขียนสมการเชิงอนุพันธ์แทนอุปกรณ์ในระบบ

แปลงมาเป็นสมการเชิงอนุพันธ์ภายในได้เงื่อนไขค่าเริ่มต้นเป็นศูนย์

คำนวณหาอัตราส่วนระหว่างเอกสารพุตกับอินพุต โดยรูปแบบทั่วไปของฟังก์ชันถ่ายโอน

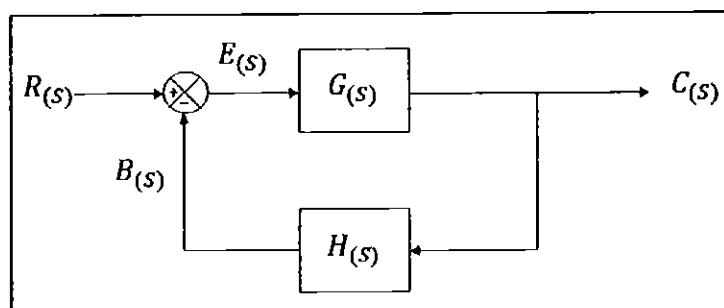
คือ

$$G(s) = \frac{Y(s)}{X(s)} \quad (2.5)$$

2.2 การหาฟังก์ชันถ่ายโอนของระบบควบคุมจากภาพลีอคไอօสแกรม

ปกติการวิเคราะห์ระบบส่วนใหญ่จะพิจารณาระบบที่แท้จริง ซึ่งค่อนข้างจะยุ่งยากและ слับซับซ้อน ทำให้การอธิบายโดยตรงด้วยสมการอนุพันธ์เป็นเรื่องที่เข้าใจยาก วิธีการหนึ่งที่จะ อธิบายให้เข้าใจได้ง่ายขึ้น คือ การเขียนภาพลีอคແທນວงจรที่ยุ่งยากก่อน จากนั้นนำแต่ละบล็อก มาประกอบกันเพื่อหาฟังก์ชันถ่ายโอนของระบบรวม ข้อได้เปรียบของการนำภาพ ลีอคไอօสแกรมมาแทนระบบ คือ กรณีที่ระบบควบคุมมีองค์ประกอบต่างกัน และมีความ เป็นอิสระต่อกัน เราสามารถที่จะหาฟังก์ชันถ่ายโอนของแต่ละภาพลีอคก่อน แล้วนำรวมกัน เพื่อหาฟังก์ชันถ่ายโอนรวมของทั้งระบบได้

ส่วนประกอบของภาพลีอคไอօสแกรมมีลักษณะดังรูปที่ 2.1



รูปที่ 2.1 แสดงบล็อกไอօสแกรมแบบมีส่วนป้อนกลับ

โดยมีองค์ประกอบต่างๆดังนี้

$R(s)$ แทน สัญญาณทางขาเข้า (Input Signal)

$C(s)$ แทน สัญญาณทางขาออก (Output Signal)

$E(s)$ แทน สัญญาณผลต่าง (Error Signal)

$G(s)$ แทน พิมพ์ชันถ่ายโอนไปหน้า (Feed forward transfer function)

$H(s)$ แทน พิมพ์ชันถ่ายโอนข้อนกลับ (Feedback transfer function)

$B(s)$ แทน สัญญาณป้อนกลับ (Feedback Signal)

ระบบป้อนกลับจะนำเอาสัญญาณทางขาเข้า $R(s)$ ผลการเปรียบเทียบจะเรียกว่า สัญญาณผลต่าง (Error signal) ของระบบ และ พิมพ์ชันถ่ายโอนข้อนกลับ $H(s)$ นอกจากจะทำหน้าที่แปลงสัญญาณทางออกแล้วยังทำหน้าที่ขยายสัญญาณนี้อีกด้วย

พิสูจน์หาพิมพ์ชันถ่ายโอนแบบปิด (Close loop transfer function) ได้ดังนี้

$$C(s) = G(s)E(s) \quad (2.6)$$

$$E(s) = R(s) - B(s) \quad (2.7)$$

$$B(s) = H(s)C(s) \quad (2.8)$$

$$\text{แทน (8) ลงใน (7) ได้ } E(s) = R(s) - H(s)C(s) \quad (2.9)$$

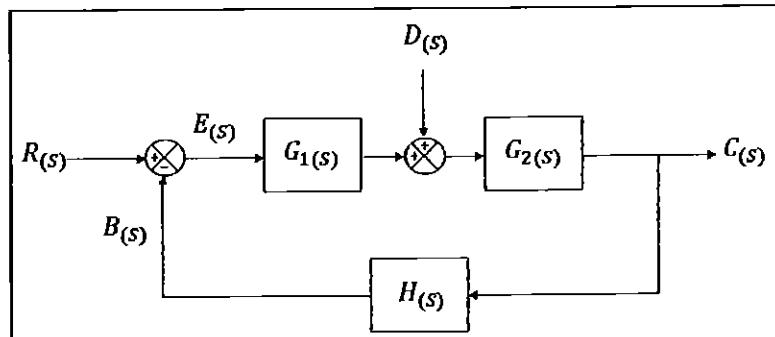
$$\text{แทน (9) ลงใน (6) ได้ } C(s) = G(s)R(s) - G(s)H(s)C(s) \quad (2.10)$$

$$\text{จะได้พิมพ์ชันถ่ายโอนแบบปิดคือ } \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \quad (2.11)$$

โดยที่ $1 + G(s)H(s) = 0$ เรียกว่า สมการคุณลักษณะ (Characteristic equation)

พิมพ์ชันถ่ายโอนแบบปิดที่มีสัญญาณรบกวน (Disturbance) มาเกี่ยวข้อง

ภาพนัดออกໄดอะแกรมจะมีลักษณะดังรูปที่ 2.2



รูปที่ 2.2 แสดงบล็อกໄดอะแกรมเมื่อมีสัญญาณรบกวน

ตัว $D(s)$ ก็อ สัญญาณรบกวน(Disturbance)

ในการหาฟังก์ชันถ่ายโอนในลักษณะนี้จะใช้วิธีแยกคิดทีละสัญญาณแล้วนำมานำกันซึ่งมีวิธีทำดังนี้

กำหนดให้ $D(s)$ เป็นศูนย์ แล้วคิดเฉพาะสัญญาณขาเข้าของ $R(s)$ จะได้ฟังก์ชันถ่ายโอนของสัญญาณขาเข้า เป็น

$$\frac{C_R(s)}{R(s)} = \frac{G_1(s)G_2(s)}{1 + G_1(s)G_2(s)H(s)}$$

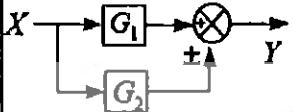
และกำหนดให้ $R(s)$ เป็นศูนย์ แล้วคิดเฉพาะสัญญาณรบกวน $D(s)$ จะได้ฟังก์ชันถ่ายโอนของสัญญาณรบกวน เป็น

$$\frac{C_D(s)}{D(s)} = \frac{G_2(s)}{1 + G_1(s)G_2(s)H(s)} \quad (2.12)$$

จะได้ฟังก์ชันถ่ายโอนรวมจาก $C(s) = C_R(s) + C_D(s)$

$$C(s) = \frac{G_1(s)G_2(s)}{1 + G_1(s)G_2(s)H(s)} [G_1(s)R(s) + D(s)] \quad (2.13)$$

ตารางที่ 2.1 ตารางการลดรูปล็อกไฮอะแกรน

รูปเดิม	ลดรูปแล้ว	สมการ
$X \rightarrow [G_1] \rightarrow [G_2] \rightarrow Y$	$X \rightarrow [G_1 G_2] \rightarrow Y$	$Y = (G_1 G_2)X$
	$X \rightarrow [G_1 \pm G_2] \rightarrow Y$	$Y = (G_1 \pm G_2)X$
	$u \rightarrow [G] \rightarrow y$ $u \leftarrow [1/G] \leftarrow y$	$y = Gu$ $u = \frac{1}{G}y$
	$u \rightarrow [G] \rightarrow y$ $y \leftarrow [G] \leftarrow u$	$y = Gu$
	$u_1 \rightarrow [G] \rightarrow y$ $u_2 \rightarrow [G] \rightarrow y$	$e_2 = G(u_1 - u_2)$
	$u_1 \rightarrow [G] \rightarrow y$ $u_2 \rightarrow [G] \rightarrow y$ $u \rightarrow [1/G] \rightarrow u_2$ $u \rightarrow [G_2] \rightarrow y$ $u \rightarrow [1/G_2] \rightarrow G_1$ $u \rightarrow [G_1] \rightarrow y$	$y = Gu_1 - u_2$ $y = (G_1 - G_2)u$

2.3 ปริภูมิสถานะ (State Space)

1) State หรือ สถานะ คือ เขตที่เล็กที่สุดของตัวแปร โดยตัวแปรจะถูกเรียกว่า ตัวแปรสถานะ ที่อธิบายระบบ $t \geq t_0$

2) State Variable หรือ ตัวแปรสถานะ คือ ตัวแปรที่อยู่ในเขตที่เล็กที่สุดที่ใช้อธิบายระบบ เช่น ถ้ามีตัวแปรสเก็ตบอร์ดอย n ตัว คือ $x_1, x_2, x_3, \dots, x_n$ ที่สามารถอธิบายระบบได้สมบูรณ์ ดังนั้น ตัวแปร n คือ ตัวแปรสถานะ

3) State Vector หรือ เวกเตอร์สถานะ คือ ถ้ามีตัวแปรสถานะจำนวน n ตัวที่สามารถอธิบายระบบได้อย่างสมบูรณ์ ตัวแปรสถานะจำนวน n ตัวนี้จะถูกพิจารณาเป็นส่วนประกอบของ เวกเตอร์ X และเวกเตอร์ X นี้จะถูกเรียกว่า เวกเตอร์สถานะ

$$X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}_{n \times 1} \quad \text{เวกเตอร์สถานะ}$$

4).State Space คือ ปริภูมิที่ประกอบไปด้วยมิติ n มิติ ที่มีแกนพิกัด $x_1, x_2, x_3, \dots, x_n$ สมการสถานะและสมการเอาต์พุต คือ

$$\dot{x}(t) = f(x, u, t) \quad (2.14)$$

$$y(t) = g(x, u, t) \quad (2.15)$$

โดยที่ n คือ อินพุต , $\dot{x}(t)$ คือ สมการสถานะ, $y(t)$ คือ สมการเอาต์พุต

ในระบบเชิงเส้น รูปแบบทั่วไปคือ

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (2.16)$$

$$y(t) = C(t)x(t) + D(t)u(t) \quad (2.17)$$

โดยที่ $A(t)$ คือ เมตริกซ์สถานะ

$B(t)$ คือ เมตริกซ์อินพุต

$C(t)$ คือ เมตริกซ์เอาต์พุต

$D(t)$ คือ เมตริกซ์ส่งผ่านโดยตรง

2.4 ความสัมพันธ์ระหว่างฟังก์ชันถ่ายโอน กับ สมการปริภูมิสถานะ

$$\text{พิจารณาฟังก์ชันถ่ายโอน โดยให้เป็นดังสมการ } \frac{Y(s)}{U(s)} = G(s) \quad (2.18)$$

สมการปริภูมิสถานะ คือ

$$\dot{x} = Ax + Bu \quad (2.19)$$

$$y = Cx + Du \quad (2.20)$$

แปลงลาปต้าชสมการปริภูมิสถานะ ได้เป็น

$$sX(s) - x(0) = AX(s) + BU(s) \quad (2.21)$$

$$Y(s) = CX(s) + DU(s) \quad (2.22)$$

$$\text{กำหนดให้ } x(0) = 0 \text{ จะได้ } sX(s) - AX(s) = BU(s) \quad (2.23)$$

$$(sI - A)X(s) = BU(s)$$

คูณด้วย $(sI - A)^{-1}$ ทั้งสองข้างของสมการ จะได้

$$X(s) = (sI - A)^{-1} BU(s) \quad (2.24)$$

$$Y(s) = [C(sI - A)^{-1} B + D]U(s) \quad (2.25)$$

เปรียบเทียบกับสมการ $Y(s) = U(s)G(s)$ จะได้

$$G(s) = C(sI - A)^{-1} B + D$$

(2.26)

นี้คือฟังก์ชันถ่ายโอนในเทอมของ A, B, C , และ D

สุดท้าย $G(s)$ สามารถเขียนอยู่ในรูป

$$G(s) = \frac{Q(s)}{(sI - A)} \quad (2.27)$$

2.5 ผลตอบสนองของระบบ

การออกแบบระบบควบคุมนั้นจำเป็นต้องตรวจสอบผลตอบสนอง (Response) ของระบบต่อสัญญาณป้อนเข้า เพื่อทดสอบคุณสมบัติของระบบและผลตอบสนอง โดยป้อนสัญญาณที่รูปปั่นป่วน หรือสัญญาณร่องรอย ให้กับระบบแล้ว จึงพิจารณาจากผลตอบสนองของระบบในโดเมนเวลา (Time Domain) เพื่อหาผลตอบสนองของระบบ (Transient Response) ความคลาดเคลื่อนในสถานะคงที่ (Steady-state error) และเสถียรภาพของระบบ (Stability) ใน การศึกษาถึงผลตอบสนอง ของระบบควบคุม นิยมใช้สัญญาณแบบหนึ่งหน่วย (Step Function), แบบพัลส์ (Pulse Function), แบบลาก (Ramp Function) ในการป้อนเพื่อศึกษาพฤติกรรมของระบบต่อสัญญาณที่ป้อนเข้า

โดยทั่วไปผลตอบสนองต่อเวลาของระบบความคุณเมื่อการพิจารณาออกเป็น 2 ส่วนคือ

1. ผลตอบสนองชั่วขณะ (Transient Response) จะพิจารณาลักษณะการเปลี่ยนแปลงของระบบความคุณทันทีหลังจากสัญญาณอินพุตเริ่มเปลี่ยนแปลงจนกระทั่งเข้าสู่จุด平衡นิ่ง
2. ผลตอบสนองสภาวะคงที่ (Steady-State Response) จะพิจารณาลักษณะการเปลี่ยนแปลงของระบบความคุณ หลังจากระบบทั้งหมดเข้าสู่สภาวะสมดุลใหม่หรือการเปลี่ยนแปลงทั้งหมดหยุดลง

ถ้าให้ $C(t)$ เป็นผลตอบสนองทางเวลา เราจะได้สมการเป็น $C(t) = C_i(t) + C_{ss}(t)$

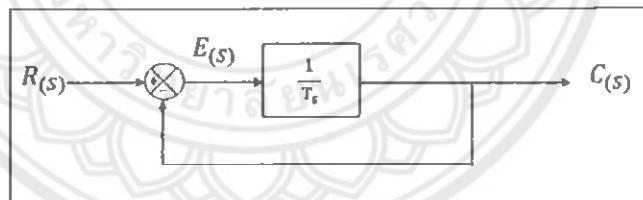
โดย $C_i(t)$ เท่ากับ ผลตอบสนองชั่วขณะ และ $C_{ss}(t)$ เท่ากับ ผลตอบสนองสภาวะคงที่

ผลตอบสนองชั่วขณะนี้จะมีค่าเข้าสู่จุด平衡 เมื่อเวลาเข้าสู่อนันต์ ($t \rightarrow \infty$) ดังสมการ

$$\lim_{t \rightarrow \infty} C_i(t) = 0 \quad (2.28)$$

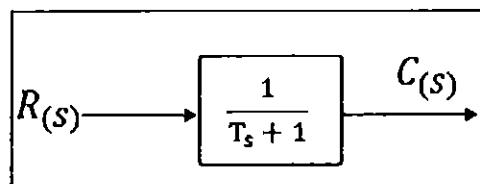
2.5.1 ระบบความคุณอันดับหนึ่ง (First-Order System)

พิจารณาบล็อกไซโอดะแกรมของระบบอันดับหนึ่ง ดังรูปที่ 2.3



รูปที่ 2.3 บล็อกไซโอดะแกรมแสดงระบบอันดับหนึ่ง

ลักษณะเป็น



รูปที่ 2.4 ภาพลักษณะบล็อกไซโอดะแกรมระบบอันดับหนึ่ง

$$\text{จะได้ฟังก์ชันถ่ายโอนคือ } \frac{C(s)}{R(s)} = \frac{1}{T_s + 1} \quad (2.29)$$

$$\text{จะได้พังก์ชันถ่ายโอนคือ } \frac{C(s)}{R(s)} = \frac{1}{T_s + 1} \quad (2.29)$$

ในการวิเคราะห์ผลตอบสนองของระบบนี้ จะใช้สัญญาณสัญญาณในการทดสอบ คือ สัญญาณระดับ (Unit-step) สัญญาณลาด (Unit-ramp) สัญญาณอินพัลส์ (Impulse) เป็นสัญญาณขั้นพุ่งและกำหนดให้ค่าเริ่มต้นเป็นศูนย์

1). ผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณระดับ

(Unit-step response of first-order system)

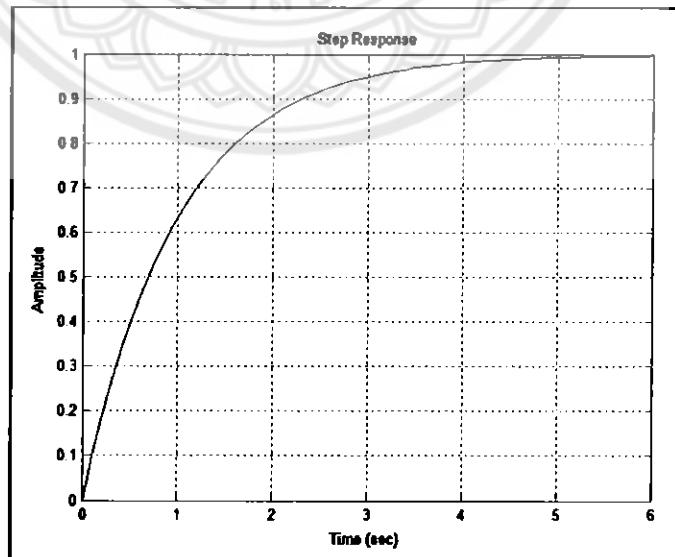
$$\text{เมื่อทำการแปลงลาปลาชสัญญาณอันดับหนึ่ง จะได้ } R(s) = \frac{1}{s} \quad (2.30)$$

$$\text{เปลี่ยนผลตอบสนองของสัญญาณเอาต์พุตได้เป็น } C(s) = \frac{1}{T_s + 1} \cdot \frac{1}{s} \quad (2.31)$$

$$\text{เมื่อทำการแยกเศษส่วนบໍอย (Partial fraction) จะได้ } C(s) = \frac{1}{s} - \frac{1}{T_s + 1} \quad (2.32)$$

$$\text{แปลงลาปลาชผกผัน จะได้ } c(t) = 1 - e^{-\frac{t}{T}} , t \geq 0 \quad (2.33)$$

จะเห็นว่าค่าเริ่มต้นของสัญญาณเอาต์พุต $c(t)$ มีค่าเป็นศูนย์และค่าสุดท้ายมีค่าเท่ากับหนึ่งและที่ $t = T$ จะได้ $c(t) = 1 - e^{-1} = 0.632$ จะได้ค่าของ $c(t)$ มีค่าเท่ากับ 0.632 หรือ 63.2% ของการเปลี่ยนแปลงทั้งหมด



รูปที่ 2.5 กราฟลักษณะการตอบสนองของระบบควบคุมอันดับหนึ่ง มีลักษณะของกราฟเป็นแบบเอ็กโพเนนเชียลที่มีความชันของกราฟที่ $t = 0$ คือ $\frac{1}{T}$

$$\frac{dc}{dt} \Big|_{t=0} = \frac{1}{T} \bullet e^{\frac{-t}{T}} \Big|_{t=0} = \frac{1}{T} \quad (2.34)$$

$$\text{สัญญาณความคลาดเคลื่อน } e_{(1)} \text{ หาได้จาก } e_{(1)} = r_{(1)} - c_{(1)} = e^{\frac{-t}{T}} \quad (2.35)$$

เมื่อ t เข้าสู่อนันต์ แล้ว ค่า $e^{\frac{-t}{T}}$ จะเข้าสู่ศูนย์ทำให้ค่าความคลาดเคลื่อน มีค่าเท่ากับศูนย์

2).ผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณลาก

(Unit-ramp Response of first-Order system)

$$\text{เมื่อทำการแปลงลาปลาชสัญญาณลาก จะได้ } R(s) = \frac{1}{s^2} \quad (2.36)$$

$$\text{เมื่อผลตอบสนองของสัญญาณเอาต์พุต ได้เป็น } C(s) = \frac{1}{T_s + 1} \bullet \frac{1}{s^2} \quad (2.37)$$

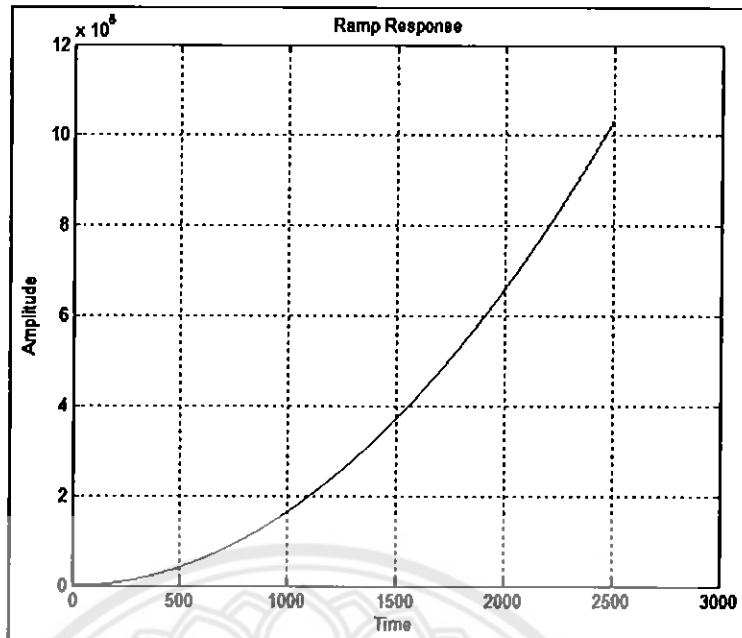
$$\text{เมื่อทำการแยกเศษส่วนบ่อขึ้น(Partial fraction) จะได้ } C(s) = \frac{1}{s^2} - \frac{T}{s} + \frac{T^2}{T_s + 1} \quad (2.38)$$

$$\text{แปลงลาปลาชผกผัน จะได้ } c(t) = t - T \left(1 - e^{\frac{-t}{T}} \right), t \geq 0 \quad (2.39)$$

$$\text{สัญญาณความคลาดเคลื่อน } e_{(1)} \text{ หาได้จาก } e_{(1)} = r_{(1)} - c_{(1)} = T \left(1 - e^{\frac{-t}{T}} \right) \quad (2.40)$$

เมื่อ t เข้าสู่อนันต์ แล้ว ค่า $e^{\frac{-t}{T}}$ จะเข้าสู่ศูนย์ทำให้ค่าความคลาดเคลื่อน มีค่าเท่ากับ T

$$e_{(\infty)} = T$$



รูปที่ 2.6 กราฟลักษณะการตอบสนองของระบบความคุณอันดับหนึ่ง ต่อสัญญาณลาด

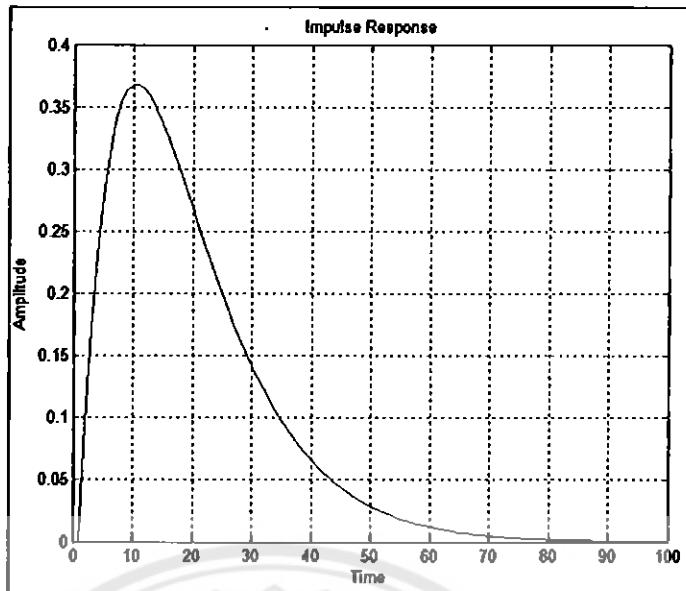
3). ผลตอบสนองของระบบความคุณอันดับหนึ่งต่อสัญญาณอินพุตส์

(Impulse Response of first-Order system)

เมื่อทำการแปลงลาปลาชสัญญาณลาด จะได้ $R(s) = 1$ (2.41)

เขียนผลตอบสนองของสัญญาณเอาต์พุตได้เป็น $C(s) = \frac{1}{T_s + 1}$ (2.42)

แปลงลาปลาชผกผัน จะได้ $c(t) = \frac{1}{T} \cdot e^{\frac{-t}{T}}, t \geq 0$ (2.43)



รูปที่ 2.7 ภาพผลตอบสนองของระบบควบคุมอันดับหนึ่งต่อสัญญาณอินพัลส์

2.5.2 ระบบควบคุมอันดับสอง (Second -Order Systems)

ระบบควบคุมอันดับสอง คือ ระบบควบคุมที่ตัวแปรออก สามารถแสดงได้โดยสมการอนุพันธ์กำลังสองหรือสามารถถอดรูบایได้ค่วยฟังก์ชันถ่ายโอนแบบปิด (Closed-loop Transfer-Function , $\frac{C(s)}{R(s)}$)

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (2.44)$$

โดย ω_n คือ ความเร็วเชิงมุนในการแก่วิ่งตามธรรมชาติ (Undamped natural frequency)

ζ คือ อัตราการหน่วงของระบบ (damping ratio)

การวิเคราะห์คุณสมบัติของระบบควบคุมอันดับสอง จะพิจารณาในเทอมของตัวแปรสองตัว คือ ω_n และ ζ

- กรณี $0 < \zeta < 1$ ตำแหน่งของโพล จะเป็นค่าเชิงซ้อนสังขคกัน มีตำแหน่งอยู่ทางซ้ายบนรูปนากอนเอกสารนี้ระบบควบคุมจะอยู่ในสภาพความหน่วงน้อย (Under damped) และผลตอบสนองช้าๆจะเกิดการแก่วิ่ง (oscillatory)

- กรณี $\zeta = 1$ เป็นระบบควบคุมที่ความหน่วงมีค่าิกฤติ (Critical damped)

- กรณี $\zeta = 1$ เป็นระบบควบคุมที่ความหน่วงมีค่าวิกฤติ (Critical damped)

- กรณี $\zeta > 1$ เป็นระบบควบคุมที่มีความหน่วงมาก (Over damped)

ทั้งสองกรณีผลตอบสนองช้าๆ และไม่เกิดการแกว่ง

พิจารณาผลตอบสนองของระบบควบคุมต่อสัญญาณอันดับหนึ่ง (Unit-step) แบ่งเป็น 3

กรณี คือ

1). กรณี $0 < \zeta < 1$ เป็นระบบที่มีความหน่วงน้อย

จากสมการฟังก์ชันถ่ายโอนแบบปิด

$$\frac{C_{(s)}}{R_{(s)}} = \frac{\omega_n^2}{(s + \zeta\omega_n + j\omega_d)(s + \zeta\omega_n - j\omega_d)} \quad (2.45)$$

โดยที่ ω_d คือความเร็วเชิงมุนในการแกว่งของระบบ มีความหน่วงน้อย มีค่าเท่ากับ $\omega_n\sqrt{1-\zeta^2}$ และผลตอบสนองของระบบเมื่อสัญญาณอินพุตเป็นฟังก์ชันอันดับหนึ่ง มีค่า $R(s)$ เท่ากับ $\frac{1}{s}$

$$\text{จะได้ } \frac{C_{(s)}}{R_{(s)}} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \cdot \frac{1}{s} \quad (2.46)$$

เมื่อแยกตัวประกอบจะได้

$$C_s = \frac{1}{s} - \frac{s + \zeta\omega_n}{(s + \zeta\omega_n)^2 + \omega_d^2} - \frac{\zeta}{\sqrt{1-\zeta^2}} \cdot \frac{\omega_n \cdot \sqrt{1-\zeta^2}}{(s + \zeta\omega_n)^2 + \omega_d^2} \quad (2.47)$$

ทำการแปลงตามปลาซิกมัน จะได้

$$C_t = L^{-1}[C_s] = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}} \cdot \sin\left(\omega_d t + \tan^{-1} \frac{\sqrt{1-\zeta^2}}{\zeta}\right) \quad (2.48)$$

สัญญาณความคลาดเคลื่อน $e_{(t)}$ ของระบบ คือ $e_{(t)} = r_{(t)} - c_{(t)}$

$$e_{(t)} = e^{-\zeta\omega_n t} \left[\cos \omega_d t + \frac{\zeta}{\sqrt{1-\zeta^2}} \cdot \sin \omega_d t \right], t \geq 0 \quad (2.49)$$

จะเห็นได้ว่าความถี่ในการแกว่งของผลตอบสนองช้าๆ จะเท่ากับ ω_d และจะเปลี่ยนแปลง

ตามอัตราการหน่วง ζ

2). กรณี $\zeta = 1$ เป็นระบบที่มีความหน่วงมีค่าวิกฤติ

เป็นระบบควบคุมที่มีความหน่วงเป็นค่าวิกฤติ โดย โพลทั้งสองจะเป็นค่าจริง และมีค่าเท่ากันหรือเกือบเท่ากัน สัญญาณเอาต์พุตแสดงการตอบสนองของระบบมีค่า

$$C_s = \frac{\omega_n^2}{(S + \omega_n)^2} \cdot \frac{1}{S} \quad (2.50)$$

เมื่อทำการแปลงถ้าปลาซึ่กันจะได้

$$C_t = L^{-1}[C_s] = 1 - e^{-\omega_n t} (1 + \omega_n t) \quad , t \geq 0 \quad (2.51)$$

3). กรณี $\zeta > 1$ เป็นระบบที่มีความหน่วงมาก

โพลทั้งสองจะเป็นค่าจริงและมีค่าต่างกัน ระบบจะมีคุณสมบัติเป็นระบบควบคุมอันดับหนึ่ง 2 ระบบประกอบกัน สัญญาณเอาต์พุตจะมีค่า

$$C_{(s)} = \frac{\omega_n^2}{(s + \zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1})(s + \zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1})} \cdot \frac{1}{S} \quad (2.52)$$

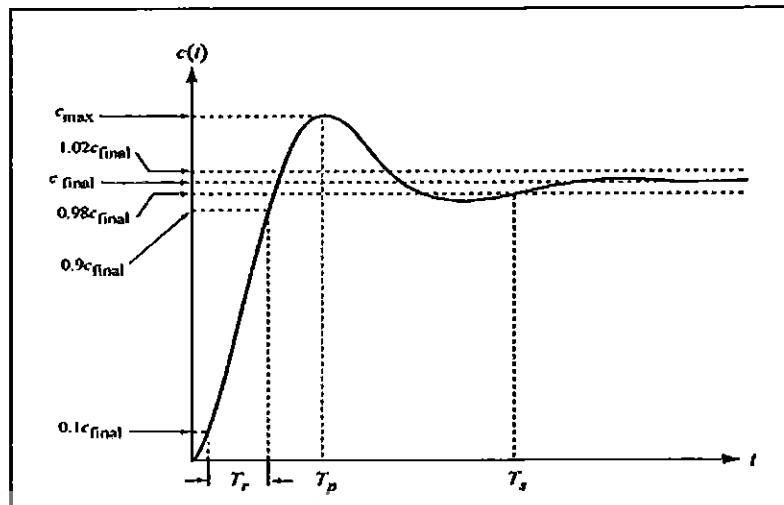
เมื่อแปลงถ้าปลาซึ่กันจะได้

$$C_t = L^{-1}[C_s] = 1 + \frac{\omega_n}{2\sqrt{\zeta^2 - 1}} \left[\frac{e^{-P_1 t}}{P_1} - \frac{e^{-P_2 t}}{P_2} \right] \quad , t \geq 0 \quad (2.53)$$

โดยที่ $P_1 = \omega_n (\zeta + \sqrt{\zeta^2 - 1})$ และ $P_2 = \omega_n (\zeta - \sqrt{\zeta^2 - 1})$

ดังนั้นผลตอบสนอง C_t จึงประกอบด้วยเทอมของเชิงเส้นเชิงสอดคล้อง

2.5.3 คุณลักษณะของผลตอบสนองชั้นแรกของระบบอันดับสอง



รูปที่ 2.8 กราฟแสดงผลตอบสนองของระบบอันดับสอง

ที่มา : <http://www.flickr.com/photos/mitopencourseware/3027390131/>

ลักษณะการตอบสนองทั่วไปของระบบอันดับสอง เมื่อได้รับสัญญาณอินพุตมาตรฐานที่เป็นสัญญาณแบบอันดับหนึ่ง สามารถตรวจสอบคุณสมบัติบางประการของระบบควบคุม โดยลักษณะการตอบสนองชี้ว่าจะของระบบอันดับสองการวิเคราะห์จากค่าต่างๆดังนี้

1) เวลาหน่วง (Delay Time, t_d)

t_d คือ ระยะเวลาที่ระบบควบคุมใช้ในการตอบสนอง จนมีค่าเท่ากับครึ่งหนึ่งของค่าสุดท้ายครั้งแรก หาได้ด้วยการแทนค่า ด้วย $C(t_r) = 1$ ได้

$$C(t_r) = 1 = 1 - e^{-\zeta \omega_n t_r} \left(\cos \omega_d t_r + \frac{\zeta}{\sqrt{1-\zeta^2}} \cdot \sin \omega_d t_r \right) \quad (2.54)$$

เมื่อ $e^{-\zeta \omega_n t_r} \neq 0$ จะได้

$$\cos \omega_d t_r + \frac{\zeta}{\sqrt{1-\zeta^2}} \cdot \sin \omega_d t_r = 0 \quad (2.55)$$

$$\text{หรือ } \tan \omega_d t_r = -\frac{\zeta}{\sqrt{1-\zeta^2}} = -\frac{\omega_d}{\sigma} \quad (2.56)$$

2) เวลาเร่ง (Rise Time, t_r)

$$t_r = \frac{1}{\omega_d} \tan^{-1} \frac{\omega_d}{\delta} = \frac{\pi - \beta}{\omega_d} \quad (2.57)$$

$$\text{โดยที่} \quad \beta = \tan^{-1} \frac{\omega_d}{\sigma} \quad (2.58)$$

3) เวลาสูงสุด (Peak Time, t_p)

t_p คือ เวลาที่ระบบความคุณมีการตอบสนองมีค่าสูงสุด โดยเวลาสูงสุดของการตอบสนองสูงสุดครั้งแรกเกิดขึ้นที่ $\omega_d t_p = \pi$

$$\text{ดังนั้น} \quad t_p = \frac{\pi}{\omega_d} \quad (2.59)$$

4) การตอบสนองสูงสุด (Maximum Overshoot, M_p) คือ ค่าสูงสุดของสัญญาณ酵ตพุตของระบบ เกิดขึ้นที่เวลาสูงสุด ที่ $t = t_p = \frac{\pi}{\omega_d}$

$$M_p = C(t_p) - 1 = e^{-\left(\zeta/\sqrt{1-\zeta^2}\right)\pi} \quad (2.60)$$

หรือแสดงในรูปของเปอร์เซ็นต์การตอบสนองสูงสุด (Maximum Percent Overshoot) ได้เป็น

$$\%M_p = e^{-(\sigma/\omega_d)\pi} \times 100\% \quad (2.61)$$

5) เวลาสู่สมดุล (Settling Time, t_s) คือ ระยะเวลาที่ระบบความคุณใช้ในการตอบสนองจนมีค่าเข้าใกล้ค่าสุดท้าย และต่างจากค่าสุดท้ายไม่เกิน $\pm 2\%$ หรือ $\pm 5\%$ ดังนี้

$$t_s = 4T = \frac{4}{\sigma} = \frac{4}{\zeta\omega_n} \text{ สำหรับกรณี } \pm 2\% \quad (2.62)$$

$$t_s = 3T = \frac{3}{\sigma} = \frac{3}{\zeta\omega_n} \text{ สำหรับกรณี } \pm 5\% \quad (2.63)$$

2.6 หาสถิติรากของระบบความคุณเชิงเส้น

2.6.1 เร้าท์ไฮร์วิช (Routh-Hurwitz Criterion)

A. Hurwitz และ E.J. Routh ได้เสนอวิธีการพิจารณาความเสถียรของระบบซึ่งพิจารณาจากสมการเฉพาะของระบบ โดยไม่ต้องหารากของสมการ ซึ่งได้รับความนิยมอย่างแพร่หลาย อย่างไรก็ตามเนื่องจากปัจจุบัน คอมพิวเตอร์เข้ามามีบทบาทมากในงานวิศวกรรม และโปรแกรมต่าง ๆ มี ประสิทธิภาพมากขึ้น การหารากของสมการเป็นไปได้ง่ายขึ้น ดังนั้นในการศึกษาระบบความคุณขั้นพื้นฐานวิธีเร้าท์ไฮร์วิชนี้ อาจจะลดความสำคัญลงอย่างไรก็ตาม ในด้านการออกแบบระบบที่ซับซ้อนที่มีเร้าท์ไฮร์วิชนี้ก็ยังมีความสำคัญอยู่มาก

พิจารณาสมการเลขพา $q(s)$ ซึ่งสามารถเขียนทั่วไปได้ในรูป

$$q(s) = a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 = 0 \quad (2.64)$$

เพื่อที่จะหาความเสถียรของระบบเราจะต้องพิจารณาว่ารากของสมการนี้ มีค่าอยู่ในค้านขวามีของ ระนาบເສທ່ຽມไม่ สามารถเขียนให้อยู่ในรูป

$$q(s) = a_n (s - \gamma_1)(s - \gamma_2) \dots (s - \gamma_n) = 0 \quad (2.65)$$

$$\begin{aligned} q(s) &= a_n s^n - a_n (r_1 + r_2 + \dots + r_n) s^{n-1} + a_n (r_1 r_2 + r_1 r_3 + \dots) s^{n-2} \\ &\quad - a_n (r_1 r_2 r_3 + r_1 r_2 r_4 + \dots) s^{n-3} + \dots + a_n (-1)^n \gamma_1 \gamma_2 \gamma_3 \dots \gamma_n = 0 \end{aligned} \quad (2.66)$$

เมื่อ γ_i เป็นรากของสมการเลขพา ถ้าเราคุณสมการนี้เข้าด้วยกันเราจะได้

หรือกล่าวอีกรูปนี้ เราจะได้ว่าถ้าหากว่าต้องการให้รากทั้งหมดนั้นอยู่ด้านซ้ายมือของ ระนาบເສ และจำเป็นที่สัมประสิทธิ์แต่ละตัวต้องไม่เป็นศูนย์ เพื่อให้ระบบเสถียรข้อกำหนดนี้ถือว่าเป็นสิ่งจำเป็นแต่ยังไม่เพียงพอที่จะกำหนดคได้ว่าระบบจะเสถียร นั่นก็อเรขากราฟ ได้ทันทีว่า ระบบจะไม่เสถียร หากไม่เป็นไปตามข้อกำหนดนี้ แต่ถ้าเป็นไปตามข้อกำหนดนี้เราก็ยังจะไม่สามารถกำหนดอย่างแน่นอน ได้ว่าระบบนี้จะเสถียร

$$\begin{aligned} q(s) &= a_n s^n - a_n (\text{ผลรวมของรากทั้งหมด}) s^{n-1} \\ &\quad + a_n (\text{ผลรวมของผลคูณของรากที่เลือกครั้งละสองราก}) s^{n-2} \\ &\quad - a_n (\text{ผลรวมของผลคูณของรากที่เลือกครั้งละสามราก}) s^{n-3} + \dots \\ &\quad + a_n (-1)^n (\text{ผลคูณของรากทั้งหมด}) = 0 \end{aligned}$$

บังจะด้วยมีการพิจารณาต่อไปอีกเพื่อหาความเสถียรของระบบ

ยกตัวอย่างเช่นระบบที่มีสมการเลขพาเป็น

$$q(s) = (s+2)(s^2 - s - 4) = s^3 + s^2 + 2s + 8 \quad (2.67)$$

ซึ่งเป็นระบบ ที่มีสัมประสิทธิ์ทุกตัวมีเครื่องหมายเป็นบวกเหมือนกัน แต่ระบบนี้จะมีค่าเลขพาค่าหนึ่งเป็นลบ ซึ่งระบบจะเป็นระบบไม่เสถียร

ทฤษฎีเร้าท์ເອຣົວິຈະเป็นข้อกำหนดที่จำเป็นและเพียงพอที่จะทำให้กำหนดได้ว่าระบบนี้เป็นระบบเสถียรหรือไม่ วิธีการนี้ในตอนแรกที่มีการนำเสนอจะพิจารณาในรูปของຄิเทอร์ມີແນນທ แต่ในการนำมาใช้ในภาษาหลังจะเปลี่ยนลักษณะอุปกรณ์ในรูปของແຕວລຳດັບ (array) ມາກກວ່າ

ทฤษฎีเร้าท์ไฮอร์วิช จะมีพื้นฐานจากการจัดลำดับสัมประสิทธิ์ของสมการเฉพาะ

$$a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 = 0 \quad (2.68)$$

ให้อยู่ในรูป

$$\begin{array}{c|cccc} s^n & a_n & a_{n-2} & a_{n-4} & \dots \\ \hline s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & \dots \\ s^{n-2} & b_{n-1} & b_{n-3} & b_{n-5} & \dots \\ s^{n-3} & c_{n-1} & c_{n-3} & c_{n-5} & \dots \\ \vdots & \vdots & & & \\ s^0 & h_{n-1} & & & \end{array} \quad (2.69)$$

ตัวแปรต่อ ๆ ไป เราสามารถเขียนได้ในรูป

$$\begin{array}{c|cccc} s^n & a_n & a_{n-2} & a_{n-4} & \dots \\ \hline s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & \dots \\ s^{n-2} & b_{n-1} & b_{n-3} & b_{n-5} & \dots \\ s^{n-3} & c_{n-1} & c_{n-3} & c_{n-5} & \dots \\ \vdots & \vdots & & & \\ s^0 & h_{n-1} & & & \end{array} \quad (2.70)$$

$$\text{เมื่อ } b_{n-1} = \frac{-1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-2} \\ a_{n-1} & a_{n-3} \end{vmatrix}, b_{n-3} = \frac{-1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-4} \\ a_{n-1} & a_{n-5} \end{vmatrix}$$

$$\text{และ } c_{n-1} = \frac{-1}{b_{n-1}} \begin{vmatrix} a_n & a_{n-2} \\ b_{n-1} & b_{n-3} \end{vmatrix} \text{ เช่นนี้ไปเรื่อย ๆ}$$

แนวทางการคำนวณลำดับทั้งหมด จะใช้คิเทอร์มันเน็ต ที่มีพื้นฐานหรือใช้รูปแบบหาสมการที่หาค่าต่างๆ ทฤษฎีเร้าท์ไฮอร์วิช กำหนดไว้ว่าจำนวนราก $q(s)$ ที่มีส่วนจริงเป็นบวกนั้นจะเท่ากับจำนวนการเปลี่ยนแปลงครั้งหนาบ ในแต่ตัวตั้งแคลว์แรกในลำดับใน แคลคูลัสของเร้าท์ซึ่งข้อกำหนดนี้เป็นข้อกำหนดที่จำเป็นและเพียงพอที่จะกำหนดสภาวะเสถียรของระบบ

ในการพิจารณาแต่ตัวตั้งแคลว์แรกนี้จะต้องแยกพิจารณากรณีบ่อกออกเป็น 4 กรณี ด้วยกัน และจำเป็น ต้องใช้วิธีการพิจารณา แต่ละกรณีแตกต่างกันออกไป ทั้งนี้ต้องมีการปรับปรุงวิธีการพิจารณาให้เหมาะสมกับแต่ละกรณีด้วย โดยกรณีทั้ง 4 นี้ ก็คือ

กรณีที่ 1 ไม่มีค่าในแนวตั้ง แคลว์แรก เป็นศูนย์เลย

ตัวอย่าง ระบบอันดับสอง สมการเฉพาะในรูปพหุนามของระบบอันดับสองจะเป็น

$$q(s) = a_2 s^2 + a_1 s^1 + a_0 \quad (2.71)$$

เขียน แคลคูลัสของเร้าท์ ให้เป็น

$$\begin{array}{c|cc} s^2 & a_2 & a_0 \\ \hline s^1 & a_1 & 0 \\ s^0 & b_1 & 0 \end{array} \quad (2.72)$$

$$\text{โดย } b_1 = \frac{-1}{a_1} \begin{vmatrix} a_2 & a_0 \\ a_1 & 0 \end{vmatrix} = \frac{a_1 a_0 - (0)a_2}{a_1} = a_0$$

ดังนั้นระบบอันดับสองจะพิจารณาได้ง่าย ๆ ว่าระบบนี้จะเสถียรถ้าหากว่า a_2, a_1 และ a_0 มีเครื่องหมายเป็นบวกทั้งหมด หรือเป็นลบทั้งหมด

ตัวอย่าง ระบบอันดับสาม สมการเฉพาะในรูปพหุนามของระบบอันดับสามจะเป็น

$$q(s) = a_3 s^3 + a_2 s^2 + a_1 s + a_0 \quad (2.73)$$

จะได้แคลคูลัสของเร้าท์เป็น

$$\begin{array}{c|cc} s^3 & a_3 & a_1 \\ s^2 & a_2 & a_0 \\ s^1 & b_1 & 0 \\ s^0 & c_1 & 0 \end{array} \quad (2.74)$$

$$\text{โดย } b_1 = \frac{a_2 a_1 - a_0 a_3}{a_2} \text{ และ } c_1 = \frac{b_1 a_0}{b_1} = 0$$

ในการที่ระบบอันดับสามตามตัวอย่างนี้จะเสถียรได้ ก็คือสัมประสิทธิ์ทั้งหมดต้องเป็นบวกและ $a_2 a_1 > a_0 a_3$ ส่วนกรณีที่ $a_2 a_1 = a_0 a_3$ จะทำให้ระบบเป็นระบบเสถียรวิกฤติและจะมีรากคู่หนึ่งที่อยู่บนแกนจินตภพภายในรูปสามเหลี่ยมของระบบเสถียรวิกฤตนี้จะเข้ากับกรณีที่ 3 เพราะมีค่าในแผลตั้งแคลแครก ค่าหนึ่งเป็นศูนย์ เมื่อ $a_2 a_1 = a_0 a_3$ จะกล่าวถึงค่าไป

ตัวอย่างสุดท้ายของกรณี พิจารณาสมการเฉพาะที่ไม่มีค่าเป็นศูนย์ในแผลตั้งแคลแครกเริ่มจากพิจารณาพหุนาม

$$q(s) = (s - 1 + j\sqrt{7})(s - 1 - j\sqrt{7})(s + 3) = s^3 + s^2 + 2s + 24 \quad (2.75)$$

หากพิจารณาจากรากจะเห็นว่าระบบนี้ไม่เสถียรอย่างไรก็ตามหากเริ่มพิจารณาจากสมการในรูปพหุนามในขั้นแรกนี้ จะเห็นว่าสัมประสิทธิ์เป็นบวกทั้งหมดดังนั้นจึงเข้าข่ายที่จะเป็นระบบเสถียร ได้ขั้นตอนไปจึงต้องพิจารณาแคลคูลัสของเร้าท์ให้

$$\begin{array}{c|cc} s^3 & 1 & 2 \\ s^2 & 1 & 24 \\ s^1 & -22 & 0 \\ s^0 & 24 & 0 \end{array}$$

จะเห็นว่า มีการเปลี่ยนแปลงเครื่องหมายสองครั้งในแคลคูลัสแรก เมื่อทราบอยู่ในขั้นตอนอยู่แล้วว่า รากของสมการเฉพาะนี้จะมี 2 ค่า และที่วางอยู่ในด้านขวา มือของระบบเอส ดังนั้น แคลคูลัสของเร้าท์ จะต้องสามารถบอกได้ว่าระบบนี้เป็นระบบที่ไม่เสถียร

กรณีที่ 2 มีค่าในแนวตั้ง แคลคูลัสเป็นศูนย์หนึ่งค่า แต่ค่าในแคนอนอนมีค่าในแคลคูลัสเป็นศูนย์ นั้นไม่มีค่าอื่นเป็นศูนย์ น ในการพิจารณาต้องแทนค่า ϵ ด้วยเลขจำนวนน้อยเช่น ϵ ซึ่งยอนให้มีค่าเข้าสู่ศูนย์จากนั้นจึงทำการคำนวณหาค่าในแคลคูลัสอื่น ๆ ต่อไป ยกตัวอย่างเช่น พิจารณาสมการเฉพาะในรูปของพหุนาม

$$q(s) = s^5 + 4s^4 + 2s^3 + 4s^2 + 11s + 10 = 0 \quad (2.76)$$

$$\text{ให้ } \text{แคลคูลัสของเร้าท์ เป็นลักษณะดังนี้} \text{ ก็อ } \text{ ในการหา } b_1 \text{ เราจะได้ } b_1 = \frac{-1}{2} \begin{vmatrix} 1 & 2 \\ 2 & 4 \end{vmatrix} = 0$$

แต่ถ้าหากว่ากำหนดค่า b_1 เป็นศูนย์ไป การหาค่าอื่นๆ จะทำไม่ได้ เพราะค่า b_1 นี้จะเป็นค่าที่ต้องนำไปหารในการหาแคนอนอนอันดับต่อไป ดังนั้นเราจะกำหนดให้ $b_1 = \epsilon$ ซึ่ง ϵ เป็นค่าเล็กๆ ค่าหนึ่งแล้วจึงใส่ลิมิตของค่านี้ให้เข้าสู่ศูนย์ในภายหลัง จากนั้นขั้นตอนก็ดำเนินไปเรื่อยๆ ก็อ

s^5	1	2	11	
s^4	2	4	10	
s^3	ϵ	6	0	
s^2	c_1	10	0	
s^1	d_1	0	0	
s^0	10	0	0	

$$\text{หรือ } b_2 = \frac{-1}{2} \begin{vmatrix} 1 & 11 \\ 2 & 10 \end{vmatrix} = 6 \quad , \quad c_1 = \frac{-1}{\epsilon} \begin{vmatrix} 2 & 4 \\ \epsilon & 6 \end{vmatrix} = \frac{-12}{\epsilon} \quad ,$$

$$d_1 = \frac{-1}{c_1} \begin{vmatrix} 2\epsilon & 6 \\ c_1 & 10 \end{vmatrix} = \frac{6c_1 - 10\epsilon}{c_1}$$

$$\text{จากนั้นเป็นการใส่ลิมิตให้ } \epsilon \rightarrow 0 \text{ จะได้ } c_1 = \frac{-12}{\epsilon} \rightarrow -\infty \quad \text{ และ}$$

$$d_1 = \frac{6c_1 - 10\epsilon}{c_1} \rightarrow 6$$

จะเห็นว่าจริงๆ แล้ว C_1 เป็นค่าจำนวนลบซึ่งมีตัวเลขสูงมาก ดังนั้นถ้าคำดับของเร้าที่จึงมีการเปลี่ยนเครื่องหมายสองครั้งทำให้ระบบไม่เสถียร และมีรากอยู่บนด้านขวาของระนาบเอส จำนวน 2 ราก

กรณีที่ 3 มีค่าในแคลว์ดี้ ถ้าแรกเป็นศูนย์หนึ่งค่า และค่าในแคลวอนที่มีค่าในแคลวแรกเป็นศูนย์หนึ่นนี้มีค่า อื่นเป็นศูนย์ด้วย กรณีนี้จะเกิดขึ้นเมื่อมีแคลวอนแคลวหนึ่งมีค่าเป็นศูนย์หนึ่ง สภาพนี้จะเกิดมีเมื่อพหุนามมีรากที่ สมการกันรอบจุดศูนย์บนระนาบอสตั้งนี้จะเกิดขึ้นในกรณีที่มีตัวประกอบเช่น $(q+a)(q-a)$ หรือ $(s-jb)(s+jb)$ เป็นต้น เมื่อเกิดกรณีนี้ขึ้นจำเป็นที่จะต้องใช้ตัวช่วยโพลิโนเมียล $U(s)$ เพื่อใช้กับค่าต่อไปใน ถ้าคำดับของเร้าที่สำหรับ ตัวช่วยโพลิโนเมียล นี้จะมีอันดับเป็นเลขคู่ Steven โดยแสดงถึงจำนวนของรากที่สมมาตรกันเพื่อแสดงถึงวิธีการนี้ พิจารณาสมการเฉพาะอันดับสาม

$$q(s) = s^3 + 2s^2 + 4s + K \quad (2.77)$$

เมื่อ K เป็นตัวปรับอัตราขยายของระบบ จะได้ ถ้าคำดับของเร้าที่เป็น

$$\begin{array}{c|cc} s^3 & 1 & 4 \\ s^2 & 2 & K \\ s^1 & 8-K & 0 \\ \hline s^0 & 2 & 0 \\ & K & 0 \end{array}$$

สำหรับระบบเสถียรต้องการ $0 < K < 8$ เมื่อ $K = 8$ จะมีสองรากบนแกนจินตภาพ และระบบจะอยู่ในสภาพเสถียรวิกฤติ และจะได้แคลวอนของ s^1 นี้เป็นศูนย์ทั้งแกน เป็นสมการของ แคลวอนที่เป็นศูนย์อยู่ถัดมา สมการของแคลวแนวก่อหน้าแคลวอนที่เป็นศูนย์นี้ในกรณีที่อ แคลวอน s^2 จะได้พหุนาม ในกรณีนี้เป็น $U(s) = 2s^2 + Ks$ แต่เนื่องจาก $K = 8$ เราจะได้ $U(s) = 2s^2 + 8$ หรือ $U(s) = 2(s+j2)(s-j2)$

เพื่อแสดงให้เห็นว่า โพลิโนเมียลตัวช่วย $U(s)$ ในความเป็นจริงแล้วเป็นตัวประกอบของพหุนามเฉพาะ

$$\text{ให้หาร } q(s) \text{ ด้วย } U(s) \text{ เราจะได้ } \frac{1}{2}s+1$$

เมื่อ $K = 8$ ตัวประกอบของพหุนามเฉพาะจะเป็น

$$q(s) = (s+2)(s+j2)(s-j2) \quad (2.78)$$

1572 9439

✓.

96266 ✓

2553

กรณี 4 เมื่อมองกับในกรณี 3 และมีรากซ้ำในแกนจินตภาพ ถ้าบนแกนจินตภาพ j^b มีรากของสมการเฉพาะอยู่ ระบบจะไม่เป็นถึงเสถียรหรือไม่เสถียร แต่จะเป็นระบบเสถียรวิกฤติ ในกรณี

นี่การตอบสนองจะแก่วงตัวในลักษณะของสัญญาณรูปไซค์ แต่ถ้าบันแคนจินตภาพ j^b นี่มีรากของสมการซ้ำอยู่การตอบสนองของระบบจะไม่เสถียร เนื่องจากจะมีรูปสมการเป็น $t(\sin(\omega t + \phi))$ อย่างไรก็ตามวิธีของเรานี้เรอร์วิชจะไม่สามารถที่จะตรวจสอบได้ พิจารณาระบบที่มีสมการเฉพาะเป็น

$$q(s) = (s+1)(s+j)(s-j)(s+j)(s-j) = s^5 + s^4 + 2s^3 + 2s^2 + 1 \quad (2.79)$$

ซึ่งได้ ถ้าลำดับของเรานี้เป็น

$$\begin{array}{c|ccc} s^5 & 1 & 2 & 1 \\ s^4 & 1 & 2 & 1 \\ s^3 & \varepsilon & \varepsilon & 0 \\ s^2 & 1 & 1 & \\ s^1 & \varepsilon & 0 & \\ s^0 & 1 & & \end{array}$$

เมื่อ $\varepsilon \rightarrow 0$ จะคุยกลายกันว่าถ้าตั้งแต่แรกไม่มีการเปลี่ยนเครื่องหมาย และระบบจะเป็นระบบเสถียรวิกฤติ แต่ในความเป็นจริงทราบว่าระบบนี้ไม่เสถียร อย่างไรก็ตามหากพิจารณาเพิ่มเติมเราจะพบว่า โพลิโนเมียลตัวช่วยที่acco s^2 คือ $(s^2 + 1)$ และ โพลิโนเมียลตัวช่วยที่acco s^4 คือ $(s^4 + 2s^2 + 1) = (s^2 + 1)^2$

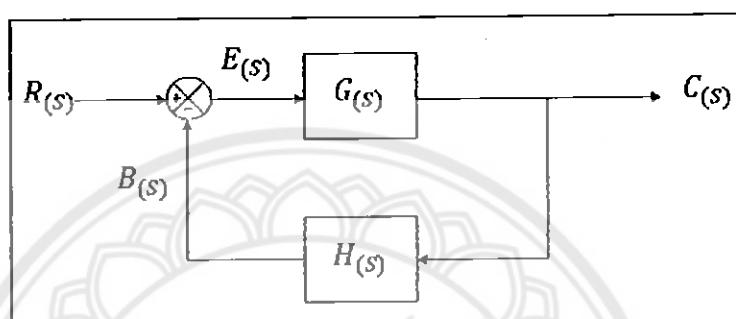
นั่นคือจะพบว่ามีรากซ้ำกัน 2 ราก ที่ $\pm j$ ดังนั้นระบบนี้จะไม่เสถียร เพื่อความสะดวกในการพิจารณาระบบว่าเสถียรหรือไม่ คำวิธีการเรานี้เรอร์วิชได้รวมรวมข้อกำหนดของพหุนามอันดับ 2 ถึง 6 เพื่อจะให้ระบบเสถียรไว้ในตารางที่ 3

ตารางที่ 2.2 ตารางแสดงข้อกำหนดของพหุนาม

	สมการเฉพาะ	ข้อกำหนด
1	$s^2 + bs + 1 = 0$	$b > 0$
2	$s^3 + bs^2 + cs + 1 = 0$	$bc - 1 > 0$
3	$s^4 + bs^3 + cs^2 + as + 1 = 0$	$bcd - d^2 - b^2 > 0$
4	$s^5 + bs^4 + cs^3 + ds^2 + as + 1 = 0$	$bcd + b - d^2 - b^2c > 0$

2.6.2 เส้นทางเดินของราก (Root locus)

ตำแหน่งของรากสมการลักษณะ (Characteristic Equation) มีผลกระทำกับลักษณะการตอบสนองของระบบความคุณเป็นอย่างมาก โดยในระบบความคุณหนึ่งๆ ตำแหน่งของรากสมการลักษณะอาจเปลี่ยนแปลงไปได้โดยการเปลี่ยนพารามิเตอร์ (Parameter) ของระบบความคุณ ในขั้นตอนนี้จะพิจารณาผลจากการเปลี่ยนแปลงของพารามิเตอร์ในระบบความคุณซึ่งแสดงไว้ในรูปที่ 2.9



รูปที่ 2.9 บล็อกไซด์แกรนของระบบความคุณ

$$\text{โดยที่} \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \quad (2.80)$$

การเปลี่ยนแปลงของพารามิเตอร์ดังกล่าวข้างต้น อาจเกิดขึ้นในด้าน $G(s)$ หรือ $H(s)$ ในลักษณะที่ทำให้สามารถเขียนสมการลักษณะของระบบความคุณได้ในรูปของ

$$0 = 1 + G(s)H(s) \quad (2.81)$$

$$0 = 1 + \frac{K(s + z_1)(s + z_2) \dots (s + z_m)}{(s + p_1)(s + p_2) \dots (s + p_n)} \quad (2.82)$$

โดย K คือพารามิเตอร์ของระบบที่เปลี่ยนแปลงได้โดยที่ $0 < K < \infty$, z_i คือจุดของ $G(s)H(s)$ และ p_i คือโพลของ $G(s)H(s)$ G เป็นไข่ของขนาด (Condition of Magnitude) สำหรับจุดบนทางเดินรากสมการลักษณะคือ

$$\frac{(s + p_1)(s + p_2) \dots (s + p_n)}{(s + z_1)(s + z_2) \dots (s + z_m)} = -K \quad (2.83)$$

โดยที่ $s - q$ คือขนาดของเวลาเดอร์ของจำนวนเชิงซ้อน $s - q$ ในระบบเอกสารซึ่งจากกฎของจำนวนเชิงซ้อน พบร่วม

$$\frac{(s+p_1)(s+p_2)\dots(s+p_n)}{(s+z_1)(s+z_2)\dots(s+z_m)} = K \quad (2.84)$$

นั่นคือ

$$\frac{(s+p_1)(s+p_2)\dots(s+p_n)}{(s+z_1)(s+z_2)\dots(s+z_m)} = K \quad (2.85)$$

เงื่อนไขของมุม (Condition of Angle) สำหรับจุดบนทางเดินรากสมการลักษณะคือ

$$\angle \left[\frac{(s+p_1)(s+p_2)\dots(s+p_n)}{(s+z_1)(s+z_2)\dots(s+z_m)} \right] = \angle [-K] \quad (2.86)$$

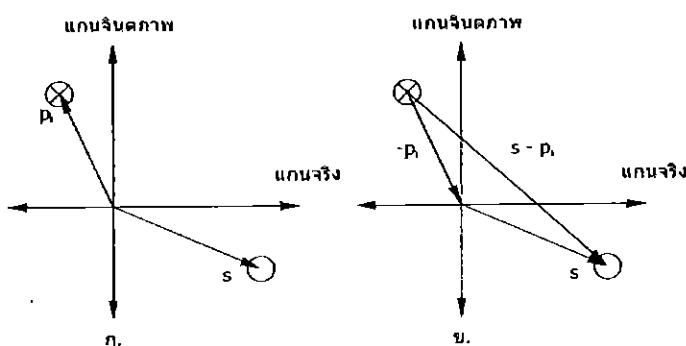
โดยที่ $\angle(s-q)$ คือมุมของเวกเตอร์ของจำนวนเชิงซ้อน $s-q$ ในระบบ笛卡特 ซึ่งเนื่องจากมุมของจำนวนจริงที่มีค่าเป็นลบคือ $-180^\circ \pm k360^\circ$, $k=0,1,\dots$ และจากกฎของจำนวนเชิงซ้อน เราพบว่า

$$\frac{\angle[(s+p_1)(s+p_2)\dots(s+p_n)]}{\angle[(s+z_1)(s+z_2)\dots(s+z_m)]} = -180^\circ \pm k360^\circ, k=0,1,\dots \quad (2.87)$$

นั่นคือ

$$\begin{aligned} \angle(s+p_1) + \angle(s+p_2) + \dots + \angle(s+p_n) - \angle(s+z_1) \\ - \angle(s+z_2) - \dots - \angle(s+z_m) = -180^\circ \pm k360^\circ, k=0,1,\dots \end{aligned} \quad (2.88)$$

จะสังเกตเห็นได้ว่ารูปแบบของเวกเตอร์ $(s-p_i)$ และ $(s-z_j)$ ปรากฏขึ้นบ่อยมากในห้องสมการของขนาดและมุม จนมีความจำเป็นต้องเข้าใจถึงลักษณะทางเรขาคณิตของเวกเตอร์เหล่านี้เพื่อใช้ในการเขียนทางเดินของรากสมการลักษณะเมื่อ K มีค่าเพิ่มขึ้นจากศูนย์จนเข้าสู่อนันต์ คั่นนั่นพิจารณาในขณะนี้เวกเตอร์ของจำนวนเชิงซ้อน s และ p_i ในรูป 2.10 ด.



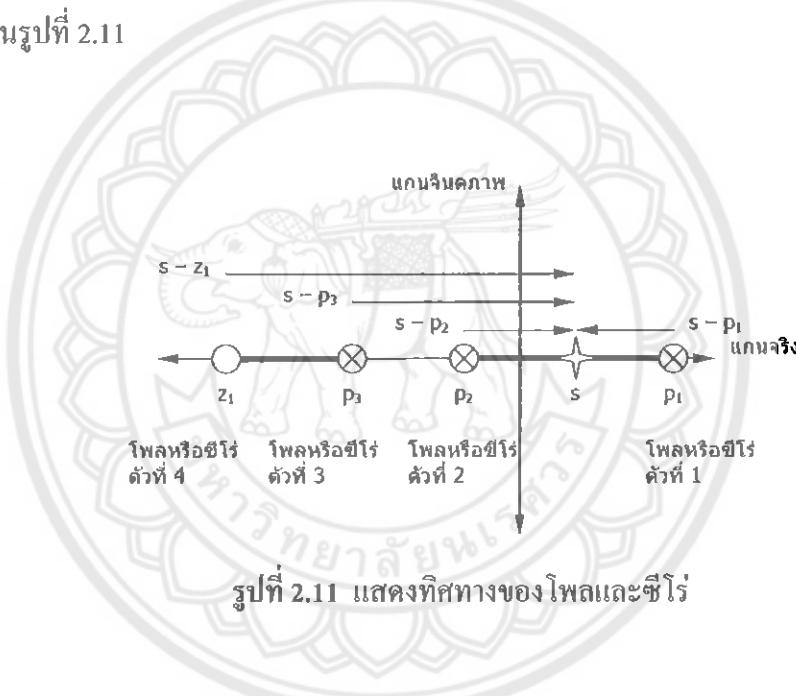
รูปที่ 2.10 เวกเตอร์แสดงจำนวนโพลและซีโร่

ในการหาเวคเตอร์ $s - pi$ เราทราบว่า $s - pi = s + (-pi)$ โดยเวคเตอร์ s และ $-pi$ แสดงไว้ในรูปที่ 2.8 ข. จากกฎของเวคเตอร์ เราทราบด้วยว่า $s + (-pi)$ กือเวคเตอร์ที่มีจุดเริ่มต้นที่หางของเวคเตอร์ $-pi$ และมีจุดปลายที่หัวของเวคเตอร์ s ดังแสดงไว้ในรูปที่ 2.10 ข.

คุณสมบัติของจุดบนทางเดินรากสมการลักษณะ

เนื่องจากทางของสมการลักษณะจะต้องเป็นไปตามเงื่อนไขของขนาดและมุม พนว่า คุณสมบัติของจุดใดๆบนระนาบเชิงช้อนระนาบอสที่จะเป็นทางของสมการลักษณะได้กือ

1) ถ้า $G(s)H(s)$ มีโพลหรือซีโร่ที่เป็นจำนวนจริง จะมีรากสมการเป็นจำนวนจริงอยู่ทางซ้ายของโพลหรือซีโร่ ตัวคี่ของ $G(s)H(s)$ ที่นับจากทางซ้ายและอยู่บนแกนจำนวนจริง ดังแสดงในรูปที่ 2.11



รูปที่ 2.11 แสดงทิศทางของโพลและซีโร่

ในรูปที่ 2.11 พิจารณาบนความคุณในรูปที่ 2.9 โดยสมมุติให้ $G(s)H(s)$ มีโพลอยู่สามตัว และมีซีโร่ออยู่หนึ่งตัว โดยแทนโพลด้วยสัญลักษณ์ \otimes แทนซีโร่ด้วยสัญลักษณ์ \star และแทนจุดที่เราต้องการทดสอบว่าจะเป็นรากสมการหรือไม่ด้วยสัญลักษณ์ ส่วนเวคเตอร์ $s - p_i$ และ $s - z_i$ ถูกเลื่อนขึ้นตามแนวแกนจำนวนเชิงช้อนเพื่อให้เห็นได้ชัดเจน ในที่นี่ต้องการที่จะแสดงให้เห็นว่ารากของสมการลักษณะจะอยู่ทางซ้ายของโพล หรือซีโร่ ตัวที่หนึ่ง (เริ่มนับจากทางซ้าย) ซึ่งแทนด้วยเส้นหนาอยู่ทางซ้ายของโพลหรือซีโร่ตัวที่สาม ซึ่งแทนด้วยเส้นหนาไม่อยู่ทางซ้ายของโพลหรือซีโร่ตัวที่สอง และไม่อยู่ทางซ้ายของโพลหรือซีโร่ตัวที่สี่ คุณสมบัติในข้อนี้ได้จากการพิจารณาเงื่อนไขของมุม โดยสำหรับสถานการณ์ในรูปที่ 2.11 พนว่าจุดจะเป็นรากของสมการก็ต่อเมื่อ

$$\begin{aligned} & \angle(s+p_1) + \angle(s+p_2) + \dots + \angle(s+p_n) - \angle(s+z_1) \\ & - \angle(s+z_2) - \dots - \angle(s+z_m) = -180^\circ \pm k360^\circ, k=0,1,\dots \end{aligned} \quad (2.89)$$

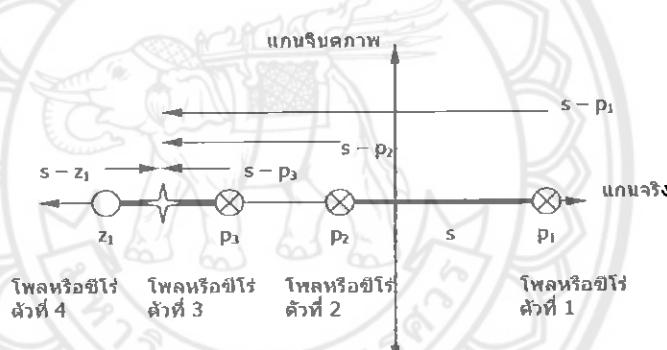
ด้านขวาแทนค่ามุมของเวกเตอร์ทั้งสี่ในรูปที่ 2.10 พบว่า

$$-180^\circ + 0^\circ + 0^\circ - 0^\circ = -180^\circ, k=0 \quad (2.90)$$

นั่นคือเงื่อนไขของมุมเป็นจริงที่ค่า $k=0$ ในลักษณะเดียวกัน ด้านพิจารณาจุดที่อยู่ระหว่าง z_1 และ p_i ดังแสดงในรูปที่ 2.12 จะพบว่าจุดดังกล่าวทำให้เงื่อนไขของมุมเป็นจริงได้เช่นกัน เมื่อจากพบว่าหลังจากแทนค่ามุมของเวกเตอร์ $s-p_i$ และ $s-z_i$ แล้ว ผลลัพธ์คือ

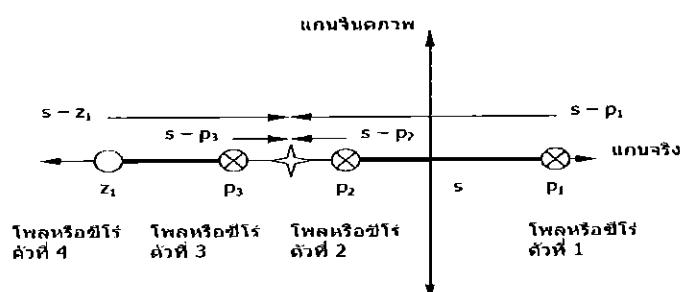
$$-180^\circ - 180^\circ - 180^\circ - 0^\circ = -180^\circ - (1)360^\circ, k=1 \quad (2.91)$$

นั่นคือเงื่อนไขของมุมเป็นจริงที่ค่า $k=0$



รูปที่ 2.12 แสดงทิศทางของโพลงและซีโร่

สำหรับจุดที่อยู่ทางซ้ายของโพลงหรือซีโร่ตัวที่สอง เงื่อนไขของมุมจะไม่เป็นจริงสำหรับค่า k ใดๆ ที่เป็นจำนวนจริงโดยจากรูปที่ 2.13



รูปที่ 2.13 แสดงทิศทางของโพลงและซีโร่

การแทนค่ามุมของเวคเตอร์ $s - p_i$ และ $s - z_i$ ลงในสมการของมุมให้ผลลัพธ์คือ

$$-180^\circ - 180^\circ + 0^\circ - 0^\circ \neq -180^\circ - (1)360^\circ, k = 1 \quad (2.92)$$

ในลักษณะเดียวกันเราจะได้สำหรับจุดที่อยู่ทางซ้ายของโพลหรือซีโร่ตัวที่สองว่า

$$-180^\circ - 180^\circ - 180^\circ - 180^\circ \neq -180^\circ - (1)360^\circ, k = 1 \quad (2.93)$$

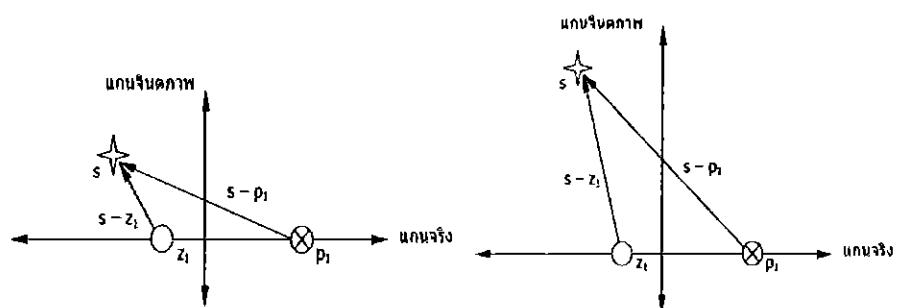
ซึ่งไม่อาจเป็นจริงสำหรับค่าใดๆ ของ k ที่เป็นจำนวนจริง

2) มุมของเวคเตอร์ของรากสมการ s ที่มีขนาดเข้าสู่อนันต์จะมีค่า

$$\angle S = \frac{180^\circ \pm k360^\circ}{n-m} \quad (2.94)$$

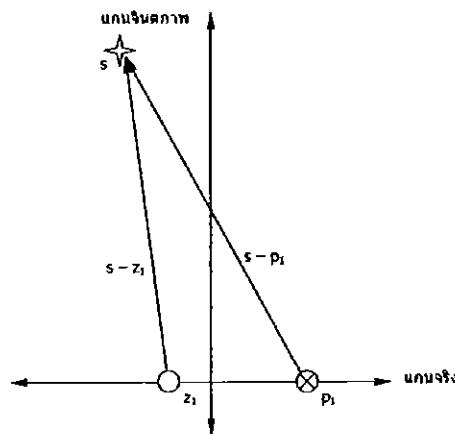
โดยที่ n และ m คือจำนวนโพลและซีโร่ของ $G(s)H(s)$ ตามลำดับ สำหรับที่มาของคุณสมบัติข้อนี้อาจนำไปใช้จากการพิจารณาปีที่ 2.12 ซึ่งใช้สัญลักษณ์ชี้ที่เดียวกันในรูปที่ 2.13 ในรูปที่ 2.14 สมมุติให้ $G(s)H(s)$ มีโพลหนึ่งตัวและซีโร่หนึ่งตัวตามตำแหน่งที่แสดงอยู่ ในรูปที่ 2.14 ก พิจารณาสถานการณ์ที่ตำแหน่งของรากสมการอยู่ใกล้กับจุดกำนิด ซึ่งหมายความว่าขนาดของเวคเตอร์ของรากสมการมีค่าน้อย จะสังเกตได้ว่าความแตกต่างระหว่างมุมของเวคเตอร์ $s - p_i$ และเวคเตอร์ $s - z_i$ นั้นมีค่านานัก

ในรูปที่ 2.14 ข เราจะเห็นได้ว่าความแตกต่างระหว่างมุมของเวคเตอร์ $s - p_i$ และเวคเตอร์ $s - z_i$ นั้นมีค่าน้อยลง และความแตกต่างของมุมนี้ยังน้อยลงมากขึ้นเมื่อ s อยู่ไกลจากจุดกำนิดมากขึ้นดังแสดงในรูปที่ 2.14 ค ในลักษณะนี้เราจึงทราบว่าเมื่อรากสมการ s อยู่ที่อนันต์ ความตั้มพันธ์ระหว่างมุมของเวคเตอร์ $s - p_i$ และเวคเตอร์ $s - z_i$ ก็คือ $\angle(s - p_i) \approx \angle(s - z_i)$



ก.

ข.



ค.

รูปที่ 2.14 แสดงเส้นทางเดินของราก

สำหรับระบบที่ $G(s)H(s)$ มี n โพล และ m ชีโว่น์ความสัมพันธ์ที่ได้ถือ

$$\angle(s + p_1) \approx \angle(s + p_2) \approx \dots \approx \angle(s + p_n) \approx \angle(s + z_1) \approx \dots \approx \angle(s + z_m) \quad (2.95)$$

นั่นคือ นุ่มนวลของแคลคูลัสที่ลากจากโพลหรือชีโว่ตัวใดๆ ของ $G(s)H(s)$ ไปที่ราก สมการที่อยู่ที่อนันต์จะมีค่าไก่กันมากจนเราสามารถถือได้ว่ามีค่าเท่ากัน เพื่อความสะดวกเราจึง ประมาณค่าของนุ่มนวลดังกล่าวด้วยนุ่มนวลของตัวรากสมการเอง เมื่อรากสมการมีขนาดเป็นอนันต์ เราจึง ให้ว่า $\angle S = \frac{180^\circ \pm k360^\circ}{n-m}$ โดยเราเรียกนุ่มนวลของรากสมการ ณ เมื่อขนาดของ s มีค่าเข้าสู่อนันต์ ที่หาได้ว่านุ่มนวลเข้า (Angle of Asymptote) เราเรียกเส้นตรงที่มีความยาวเป็นอนันต์และทำ นุ่มนวล กับ แกนจำนวนจริงเป็นนุ่มนวลเข้าว่าเส้นสู่เข้า (Line of Asymptote)

3) จุดตัดแกนจำนวนจริงของเส้นสู่เข้า (Asymptote Crossing) หาได้จากการพิจารณา สมการลักษณะในรูป $\frac{(s + p_1)(s + p_2) \dots (s + p_n)}{(s + z_1)(s + z_2) \dots (s + z_m)} = -K$ (2.96)

$$\text{จุดตัดแกนจำนวนจริงของเส้นสู่เข้า } \sigma_C \text{ หาได้จากสมการ } \sigma_C = \frac{\sum_{i=1}^n P_i - \sum_{i=1}^m Z_i}{n-m} \quad (2.97)$$

4) เมื่อทางเดินของรากสมการอยู่บนแกนจำนวนจริงดังเช่นในข้อ 1. และอยู่ระหว่างโพ สองตัว หรือชีโว่สองตัว ทางเดินของรากสมการดังกล่าวจะมีจุดซึ่งทางเดินของรากสมการที่เป็น จำนวนเชิงซ้อนพุ่งออก (Break-away point) หรือ พุ่งเข้า (Break-in Point) ตามลำดับ โดยที่จุด ดังกล่าวค่า K จะมีค่ามากที่สุดหรือน้อยที่สุด สำหรับทางเดินรากสมการที่อยู่ระหว่างโพสองตัว

หรือซีโร่สองตัวนั้นตามลำดับ สำหรับการหาจุดที่ทางเดินรากสมการพุ่งเข้าหรือพุ่งออกนั้น ทำได้โดยการเขียนสมการลักษณะในรูปของ $K = Q(s)$

$$\text{จากนั้นทำการหา } dK / ds \text{ และหารากของสมการ} \quad \frac{dK}{ds} = 0$$

โดยคำ ตอบที่ได้อาจมีได้มากนaby แต่คำตอบซึ่งอยู่บนส่วนของทางเดินรากสมการระหว่างโพลสองตัว หรือซีโร่สองตัวคือจุดที่รากสมการพุ่งเข้าหรือพุ่งออกตามลำดับ

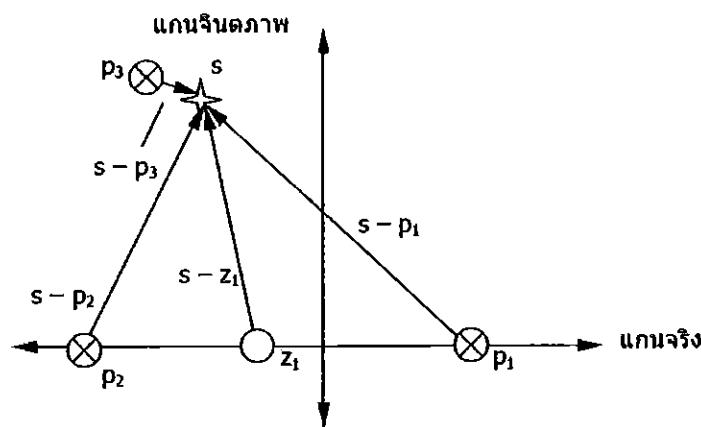
5) มุมที่โพลพุ่งเข้า (Angle of Arrival) และ มุมที่โพลพุ่งออก (Angle of Departure) จากโพลหรือซีโร่ของ $G(s)H(s)$ ที่เป็นจำนวนเชิงซ้อน หาได้จากการวิเคราะห์ดังต่อไปนี้ โดยวิเคราะห์ที่ได้จากการบบสมมุติอันหนึ่ง บนส่วนหนึ่งของระนาบจำนวนเชิงซ้อนโดยจุดมุ่งหมายของเราก็อ การหามุมที่ทางเดินรากสมการพุ่งออกจาก โพล p_3 ซึ่งเป็นจำนวนเชิงซ้อน ในรูปที่ 2.15 จุด S เป็นจุดบนทางเดินรากสมการที่อยู่ใกล้กับโพล p_3 โดยหาก s เป็นจุดบนทางเดินรากสมการจริงแล้ว เส้น直ของมุมจะต้องเป็นจริง นั่นคือ

$$\begin{aligned} & \angle(s+p_1) + \angle(s+p_2) + \dots + \angle(s+p_n) - \angle(s+z_1) \\ & - \angle(s+z_2) - \dots - \angle(s+z_m) = -180^\circ \pm k360^\circ, k=0,1,\dots \end{aligned} \quad (2.98)$$

โดยหากว่าปีที่ 5-7 เราพบว่า

$$\angle(s+p_1) + \angle(s+p_2) + \angle(s+p_3) - \angle(s+z_1) = -180^\circ \pm k360^\circ \quad (2.99)$$

เนื่องจากทางเดินรากสมการพุ่งออกจากโพล p_3 แน่นอนว่า จุด s จะต้องอยู่ใกล้กับโพล p_3 เป็นมาก ดังนั้นจึงสามารถแทน S ด้วย p_3 ซึ่งทำให้เงื่อนไขของมุมกลายเป็นโดยเราไม่แทน $s=p_3$ ใน $\angle(s-p_3)$ เนื่องจาก $\angle(s-p_3)$ เป็นมุมของทางเดินรากสมการที่พุ่งออกจากโพล p_3 ซึ่งเป็นสิ่งที่ต้องการหาและสามารถหาได้โดยการเดินรากสมการที่พุ่งออกจากโพล p_3 สำหรับระบบควบคุมที่สมมุติขึ้นจากสมการข้างต้นได้



รูปที่ 2.15 รูปแสดงเส้นทางของراك

โดยวิธีการนี้จะเห็นได้ว่า ความสัมพันธ์ในรูปแบบทั่วไปที่สามารถนำไปใช้ในการหานุนพุ่งเข้าหรือมุนมุ่งออกได้ ได้แก่

สำหรับการหานุน $\angle(s - p_T)$ ที่ทางเดินรากสมการพุ่งเข้าหาโพลเชิงซ้อน pT

$$\begin{aligned} & \angle(p_T - p_1) + \dots + \angle(s - p_T) + \dots + \angle(p_T - p_n) - \angle(p_T - z_1) \\ & - \dots - \angle(p_T - z_m) = -180^\circ \end{aligned} \quad (2.100)$$

สำหรับการหานุน $\angle(s - z_T)$ ที่ทางเดินรากสมการพุ่งเข้าหาซีโร่เชิงซ้อน zT

$$\begin{aligned} & \angle(z_T - p_1) + \dots + \angle(z_T - p_n) - \angle(z_T - z_1) \\ & - \dots - \angle(s - z_T) - \dots - \angle(z_T - z_m) = -180^\circ \end{aligned} \quad (2.101)$$

สรุปขั้นตอนการหาเส้นทางเดินของรากของสมการคุณลักษณะ มีดังนี้

- 1) เขียนฟังก์ชันถ่ายโอน
- 2) หาตำแหน่งโพลและซีโร่
- 3) หาจำนวนของโพล และซีโร่เพื่อหาเส้นทางเดินของรากที่แยกจากกัน
- 4) หานุนและจุดตัดกันของเส้นและซินโพล
- 5) หาจุดแยกออกจากแกนจิริ
- 6) หานุนจากโพล และมุมเข้าหาซีโร่ บนระนาบเอส

7) หาจุดตัดแกนจินตภาพของทางเดินของรถ

2.7 การเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI)

ส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphical User Interface, GUI) เป็นส่วนประกอบของโปรแกรมที่อยู่ระหว่างส่วนโปรแกรมหลักกับผู้ใช้ จะช่วยเพิ่มความสมบูรณ์ และความสะดวกให้แก่ผู้ใช้งานมากขึ้น

วัตถุประเกท ส่วนต่อประสานกราฟิกกับผู้ใช้ในโปรแกรมเมทแล็บ (MATLAB) ถือเป็นวัตถุชนิดหนึ่งภายใต้หน้าต่าง (Figure) ซึ่งสามารถเขียนได้ 2 รูปแบบ คือ

1) รูปแบบที่ 1 ใช้เครื่องมือช่วยในการพัฒนาโปรแกรมของโปรแกรมเมทแล็บ (Graphical User Interface Development Environment (GUIDE)) แบบนี้จะเขียนได้ยากกว่า และมีลักษณะคล้ายการเขียนโปรแกรมภาษาอื่นๆ เช่น Visual Basic เป็นต้น ซึ่งการเขียนลักษณะนี้จะได้ไฟล์จำนวน 2 ไฟล์ คือ *.m เพื่อเก็บโปรแกรม และ *.pg เพื่อเป็นหน้าต่างของส่วนต่อประสาน

2) รูปแบบที่ 2 ใช้ชุดคำสั่ง หรือ พังก์ชันในการสร้าง โดยไม่ใช้เครื่องมือช่วยในการพัฒนาโปรแกรมของโปรแกรมเมทแล็บ แบบนี้จะมีความอิสระและยืดหยุ่น ได้มากกว่า เราสามารถทำงานด้วย *.m เพียงไฟล์เดียวก็ได้

วัตถุต่างๆ ที่เราสามารถสร้างขึ้นได้ในส่วนต่อประสานกราฟิกที่เป็นวัตถุส่วนติดต่อ กับผู้ใช้ (User Interface Object) เดียวกัน แต่ต่างกันที่คุณสมบัติ Style โดย Style ที่เราได้สร้างประกอบไปด้วย

1) Edit Text เหนาะสำหรับให้ผู้ใช้ป้อนข้อมูล

2) Static Text เหนาะสำหรับใช้ในการแสดงข้อความ คำอธิบาย ชื่อ ฯลฯ ในลักษณะตาราง โดยผู้ใช้ไม่สามารถเปลี่ยนแปลง แต่สามารถเปลี่ยนข้อความที่แสดงได้ด้วยการเขียนโปรแกรม

3) Push Button เหนาะสำหรับให้ผู้ใช้กดเพื่อเป็นการตอบตกลง ยกเลิก หรือใช้ในการบอกรอเริ่มต้นหรือสิ้นสุดเหตุการณ์ใดๆ

4) Pop-up Menu เหนาะสำหรับเป็นรายการให้ผู้ใช้เป็นตัวเลือก แบบเลือกได้ตัวเลือกเดียว

5) Slider Bar เหนาะสำหรับให้ผู้ใช้ปรับค่าแบบเพิ่มหรือลด ขึ้นหรือลง ซ้ายหรือขวา เป็นต้น

6) Radio Button เหนาฯสำหรับเป็นรายการให้ผู้ใช้เป็นตัวเลือกแบบเลือกได้ตัวเดียวเดียว ซึ่งผู้ใช้สามารถเห็นตัวเลือกได้ทุกตัว โดยนิยมบรรจุ Radio Button ที่ต้องการให้เลือกข้อใดข้อหนึ่งไว้ภายใน Button Group (Panel) เดียวกัน

7) Check Box เหนาฯสำหรับให้ผู้ใช้เลือกเพื่อเปิดหรือปิด เอาหรือไม่เอา ใช่หรือไม่ใช่ เป็นต้น

8) Axes เหนาฯสำหรับใช้ในการแสดงกราฟิก เช่น กราฟ ภาพ หรือ วิดีโอ ฯลฯ

9) List Box เหนาฯสำหรับเป็นรายการให้ผู้ใช้เป็นตัวเลือก แบบเลือกได้ตัวเดียว หรือ หลายตัวเลือกพร้อมกัน

2.7.1 หลักการเขียนโปรแกรมด้วย GUIDE

การเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ มีหลักการสำคัญอยู่ที่ตัวแปร “Handles” เพื่อใช้เป็นตัวอ้างอิง หรือชี้ไปที่วัตถุแต่ละวัตถุ ค่าคุณสมบัติ (Property) ต่างๆของแต่ละวัตถุ และฟังก์ชันเรียกกลับ (Callback Function) ซึ่งมีหลักการเดียวกับการแก้ไขหน้าต่าง Figure

ตัวแปร Handles เป็นตัวแปรที่ใช้เก็บเลขจำนวนหนึ่งที่ใช้เป็นตัวอ้างอิงถึงวัตถุแต่ละวัตถุ คล้ายกับตัวชี้ดำเนินการในภาษา C ถ้าเราสร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ด้วยคำสั่ง เครื่องมือช่วยในการพัฒนา โปรแกรมของโปรแกรมเม็ทแล้ว ตัวแปร handles จะถูกสร้างให่องค์สำหรับทุกๆวัตถุที่สร้างขึ้น และบรรจุไว้ในตัวแปรชื่อ handles แล้วใช้ tag ในการเรียกใช้ เช่น handles.axes ใช้เรียกใช้ในการแสดง กราฟิก เป็นต้น

ฟังก์ชันเรียกกลับ (Callback Function)

ฟังก์ชันเรียกกลับเป็นฟังก์ชันที่ถูกเรียกใช้งานเพื่อตอบสนองเหตุการณ์แต่ละเหตุการณ์ เช่น การกดปุ่ม การลากเมาส์ผ่าน การคลิก เป็นต้น การสร้างฟังก์ชันเรียกกลับด้วยเครื่องมือช่วยในการพัฒนา โปรแกรมของ โปรแกรมเม็ทแล้ว ทำได้โดยคลิกขวาบนวัตถุที่ต้องการแล้วเลือก ดู ฟังก์ชันเรียกกลับ (View Callback) ในฟังก์ชันที่เราต้องการ

ตัวอย่าง pushbutton1_Callback(hObject, eventdata, handles) ตัวแปรที่สำคัญคือ

hObject เป็นตัวแปร handle หรือตัวชี้ของวัตถุที่เราใช้เป็นตัวเรียกฟังก์ชันเรียกกลับ นั้นๆ โดยในที่นี่มีค่าเดียวกับ handles.pushbutton1 นั้นเอง

`eventdata` ตัวแปรนี้จะถูกใช้ในบางฟังก์ชันเรียกกลับบางฟังก์ชันเท่านั้น เพื่อเก็บข้อมูล
บางอย่าง เช่น แปลงพิมพ์ที่ถูกกด หรือตัวเลือกที่ถูกเลือก เป็นต้น

`handles` เป็นตัวแปรที่สำคัญที่สุดในการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับ
ผู้ใช้ ซึ่งใช้ในการเก็บ `handles` ของวัตถุต่างๆ รวมกันทั้งหมด

การสร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้แบบหลายหน้าต่าง

ในกรณีที่เราต้องการหน้าต่างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ ในการทำงาน
มากกว่า 1 หน้าต่าง และใช้งานร่วมกัน มีการส่ง ผ่านค่าระหว่างหน้าต่าง โดยหน้าต่างเหล่านี้
สามารถใช้งานได้ 2 วิธีคือ แบบ Modeless และแบบ Modal โดยมีความแตกต่างกันคือ

1) แบบ Modeless ผู้ใช้สามารถใช้งานหน้าต่างอื่นๆ ได้อิสระ ทำได้โดยการเลือก
คุณสมบัติ `WindowStyle` ของหน้าต่างเป็น `normal` ซึ่งเป็นค่าปกติ

2) แบบ Modal ผู้ใช้ไม่สามารถทำงานที่หน้าต่างอื่นๆ ของโปรแกรมได้ นอกจาก
หน้าต่างนี้เท่านั้น ทำได้โดยเลือกคุณสมบัติ `WindowStyle` ของ หน้าต่าง เป็น `modal`

2.7.2 หลักการเขียนโปรแกรมโดยไม่ใช้ GUIDE

ในการสร้างโปรแกรม ส่วนต่อประสานกราฟิกกับผู้ใช้ด้วย เครื่องมือช่วยในการพัฒนา
โปรแกรมของโปรแกรมเม็ทเดิมเราจำเป็นต้องทราบว่าโปรแกรม จะมีลักษณะใดล่วงหน้า เพราะ
เราต้องสร้างวัตถุต่างๆ เอาไว้ก่อนในหน้าต่างโปรแกรม แต่ในบางครั้งเราไม่สามารถสรุปล่วงหน้าได้
ว่า โปรแกรมของเราจะมีหน้าตาอย่างไร เราจึงไม่สามารถสร้างไว้ก่อนด้วยเครื่องมือช่วยในการ
พัฒนาโปรแกรมของโปรแกรมเม็ทเดิมได้ ดังนั้น ในกรณีนี้เราจำเป็นต้องเขียนโปรแกรมในการ
สร้างวัตถุต่างๆ ขึ้นเองด้วยฟังก์ชัน `uicontrol`

ตัวอย่าง `handle=uicontrol(parent,'ชื่อคุณสมบัติ','ค่าคุณสมบัติ',...)`

โดยมีคุณสมบัติหลักๆ ที่ควรกำหนด ดังนี้

1) Tag หมายถึง ชื่อของวัตถุนั้นๆ

2) Style หมายถึง รูปแบบของวัตถุนั้นๆ กำหนดได้โดยใช้สตริง ดังนี้

'checkbox' ก็อปปี้แล็อกแบบ Check Box

'edit' ก็อปปี้องรับข้อความ หรือ Edit Text

'flame'	คือ รูปสี่เหลี่ยมที่ช่วยในการมอง หรือจับกู้นวัตถุต่างๆเข้าด้วยกัน เช่น uipanel หรือ uitbuttomgroup
'listbox'	คือ รายการเลือก หรือ List Box
'popupmenu'	คือ รายการเลือกแบบ Drop Down หรือ Pop-up Menu
'pushbutton'	คือ ปุ่มกดแบบปกติ หรือ Push Button
'radiobutton'	คือ ปุ่มเดือดแบบ Radio Button
'slider'	คือ แตบเลื่อน หรือ Slider Bar
'text'	คือ ข้อความแบบไม่เปลี่ยนแปลง หรือ Static Text
'togglebutton'	คือ ปุ่มกดแบบเปิดหรือปิดคล้าย Push Button แต่เวลากดปุ่มจะถ้างไว้ในสถานะนั้นจะกว่าจะกดอีกรัง หมายความว่ารับสั่งเป็นปุ่มใน แดบเครื่องมือ (Toolbar)
3) Position	หมายถึง ตำแหน่งและขนาด ซึ่งเป็นอาร์เรย์ที่มีสมาชิก 4 ตัว ได้แก่ พิกัด x ,พิกัด y, ความกว้าง (width) และความสูง (height) ตามลำดับ
4) String	หมายถึง ข้อความหรือตัวอักษรที่ปรากฏบนวัตถุนั้นๆให้ผู้ใช้เห็น
5) Callback	หมายถึง ชื่อฟังก์ชันเรียกกลับของวัตถุนั้นๆ บทสนทนา (Dialog) พิเศษ
1) บทสนทนา สำหรับแสดงข้อความทั่วไป	ใช้คำสั่ง messagebox('ข้อความ')
2) บทสนทนา สำหรับแสดงข้อความเตือน	ใช้คำสั่งwaredig('ข้อความ')
3) บทสนทนา สำหรับแสดงข้อความผิดพลาด	ใช้คำสั่งerrordig('ข้อความ')
4) บทสนทนา สำหรับแสดงข้อความช่วยเหลือ	ใช้คำสั่งhelpdig('ข้อความ')
5) บทสนทนา สำหรับป้อนข้อมูลเข้า	ใช้คำสั่งanswer=inputdig({'คำถาม 1', 'คำถาม 2',})
6) บทสนทนา สำหรับป้อนໄດเร็คทอรี่	ใช้คำสั่งdir=uigetdir
7) บทสนทนา สำหรับป้อนแฟ้มข้อมูล	ใช้คำสั่ง[filename,pathname]=uigetfile

8) บทสนทนา สำหรับเปิดแฟ้มข้อมูล ใช้คำสั่ง `uiopen`

9) บทสนทนา สำหรับเลือกสี ใช้คำสั่ง `uisetcolor`

2.8 การหาผลตอบสนองทางความถี่ (เพิ่มเติม)

ผลตอบสนองในโค้ดเมนความถี่ หมายถึง ผลตอบสนองในสถานะอยู่ตัวของระบบ
เนื่องจากสัญญาณป้อนเข้าแบบไนซ์ วิธีโค้ดเมนความถี่เป็นวิธีการดึงเดินแบบหนึ่งที่มีประโยชน์มาก
ในการวิเคราะห์และออกแบบระบบควบคุม ขั้นตอนวิธีการคือ ให้สัญญาณเข้าที่ความถี่ต่างๆ
ภายในพิสัยที่กำหนดแล้วศึกษาผลตอบสนองที่ได้

ประโยชน์ของการศึกษาวิธีหาผลตอบสนองในโค้ดเมนความถี่มีดังนี้

1) การตรวจสอบหาสตูดิโอภาพของระบบปิด ไม่จำเป็นต้องคุ้ดແเน่ง เพลลงของระบบปิด
หรือแก้สมการคุณลักษณะเพื่อหาตำแหน่งรากของสมการ

2) สามารถตรวจสอบผลตอบสนองต่อความถี่ของระบบ ได้ง่าย โดยการป้อนสัญญาณ
อินพุตเป็นสัญญาณไนซ์จากเครื่องกำเนิดสัญญาณแล้ว วัดผลการตอบสนองต่อความถี่นั้นของ
ระบบ

3) ระบบควบคุมบางชนิดที่มีความชับช้อง อาจจะใช้วิธีการตรวจสอบโดยการทดสอบ
ผล

ตอบสนองต่อความถี่ เพื่อหาฟังก์ชันด้วย โอน(Transfer function)

4) การหาผลตอบสนองต่อความถี่สามารถ นำมายังเคราะห์ฟังก์ชันที่ไม่ได้อยู่ในรูป
อัตรา

ส่วนได้ เช่น ระบบที่เคลื่อนที่ช้า (transport lag)

5) วิธีหาผลตอบสนองต่อความถี่สามารถใช้กับระบบที่มีลักษณะที่ไม่แน่นอน

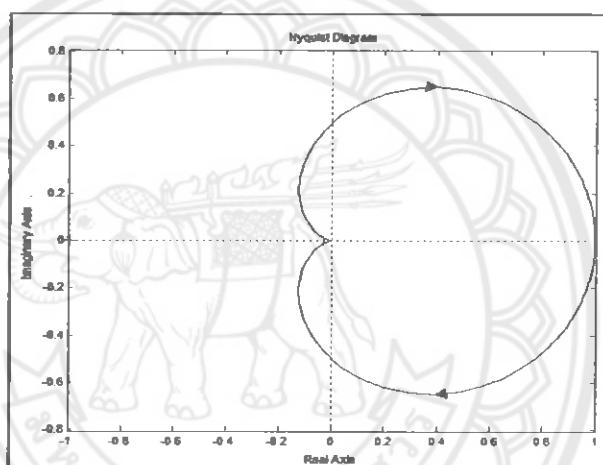
6) การออกแบบและวิเคราะห์ผลตอบสนองต่อความถี่สามารถใช้กับระบบที่เป็นไม่เป็น
เชิงเส้นได้

หลักสำคัญของการตอบสนองเชิงความถี่ (Concept of Frequency Response)

เมื่อให้สัญญาณอินพุตเป็นรูปคลื่น ไซน์เข้าไปยังระบบเชิงเส้น (Linear system) ผลการตอบสนองค้านเอาท์พุตที่สภาวะคงตัวจะเป็นสัญญาณ ไซน์ที่มีความถี่เดียวกับอินพุตแต่มีขนาด (Amplitude, Gain) และมุม (Phase) ที่แตกต่างจากสัญญาณอินพุต

2.8.1 ไนคิสต์ (Nyquist plot)

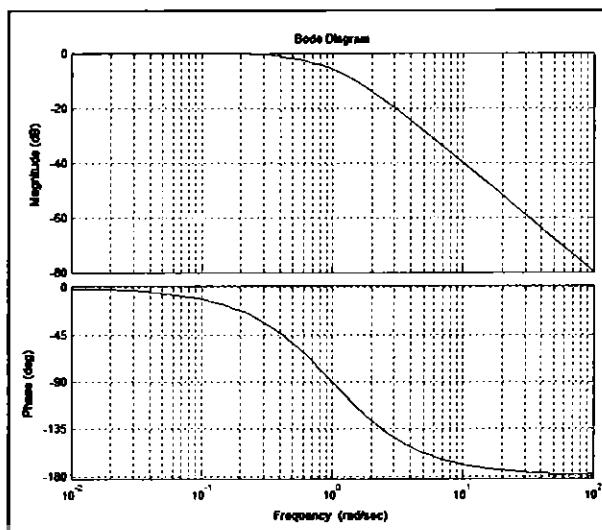
เป็นการพล็อตระหว่างขนาดและมุมของ $G(s)$ บนแกนจินตภาพ ($G(s) - plane$) เมื่อเปลี่ยนจาก $0 \rightarrow \infty$ เป็นการหาความสัมพันธ์ระหว่างความเสถียรของระบบปิด (close loop system) และการตอบสนองต่อความถี่ของระบบเปิด ซึ่งทำให้เราสามารถหาความเสถียรของระบบปิดได้เมื่อเราทราบการตอบสนองต่อความถี่ของระบบเปิด ดังตัวอย่างรูปที่ 2.15



รูปที่ 2.16 ตัวอย่าง Nyquist plot

2.8.2 โบเด (Bode plot)

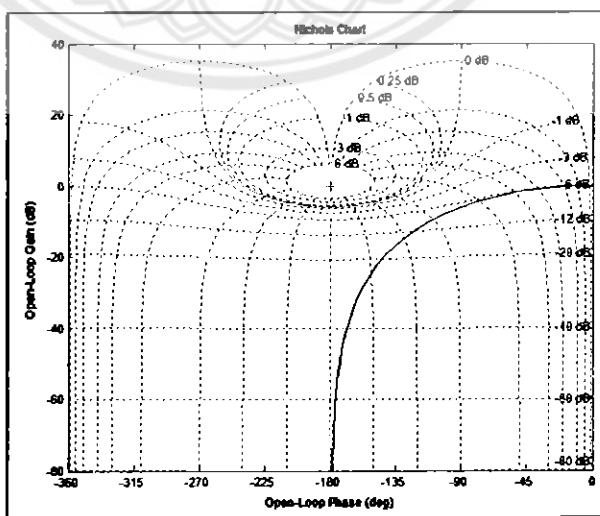
ประกอบด้วยกราฟสองชุดเป็นการแยกพล็อตขนาด (Magnitude or gain) ในสเกล log และมุม (phase) กับความถี่ในสเกล log โดยแกนตั้งเป็นขนาดมีหน่วยเป็นเดซิเบล ($\text{dB} = 20\log |G(s)|$) และมุมเพลสเมินหน่วยเป็นองศา หรือเรเดียน และแกนนอนเป็น $\log \omega$ ดังตัวอย่างรูปที่ 2.17



รูปที่ 2.17 ตัวอย่าง Bode plot

2.8.3 แผนภูมินิโคลส์ (Nichols plot)

แผนภูมินิโคลส์ ช่วยในการหาผลตอบสนองในโดเมนความถี่ของระบบวงปิด จากแผนภาพใบเดียวโดยตรง อัตราขยายเป็นเดซิเบล คือแกนตั้งของแผนภูมินิโคลส์และเฟสเป็นองศา คือ แกนนอนของแผนภูมินิโคลส์ ดังนั้นการพล็อตบนแผนภูมินิโคลส์มีความถี่เป็นพารามิเตอร์ที่ แปรผันตามเส้นโค้ง เส้นโค้งของผลตอบสนองในโดเมนความถี่ของระบบวงปิดอ่านได้จากจุดตัด ของเส้นโค้งกับค่าองศาของขนาดและเฟสคงตัวที่แต่ละความถี่ ดังตัวอย่างรูปที่ 2.17



รูปที่ 2.18 ตัวอย่าง Nichols plot

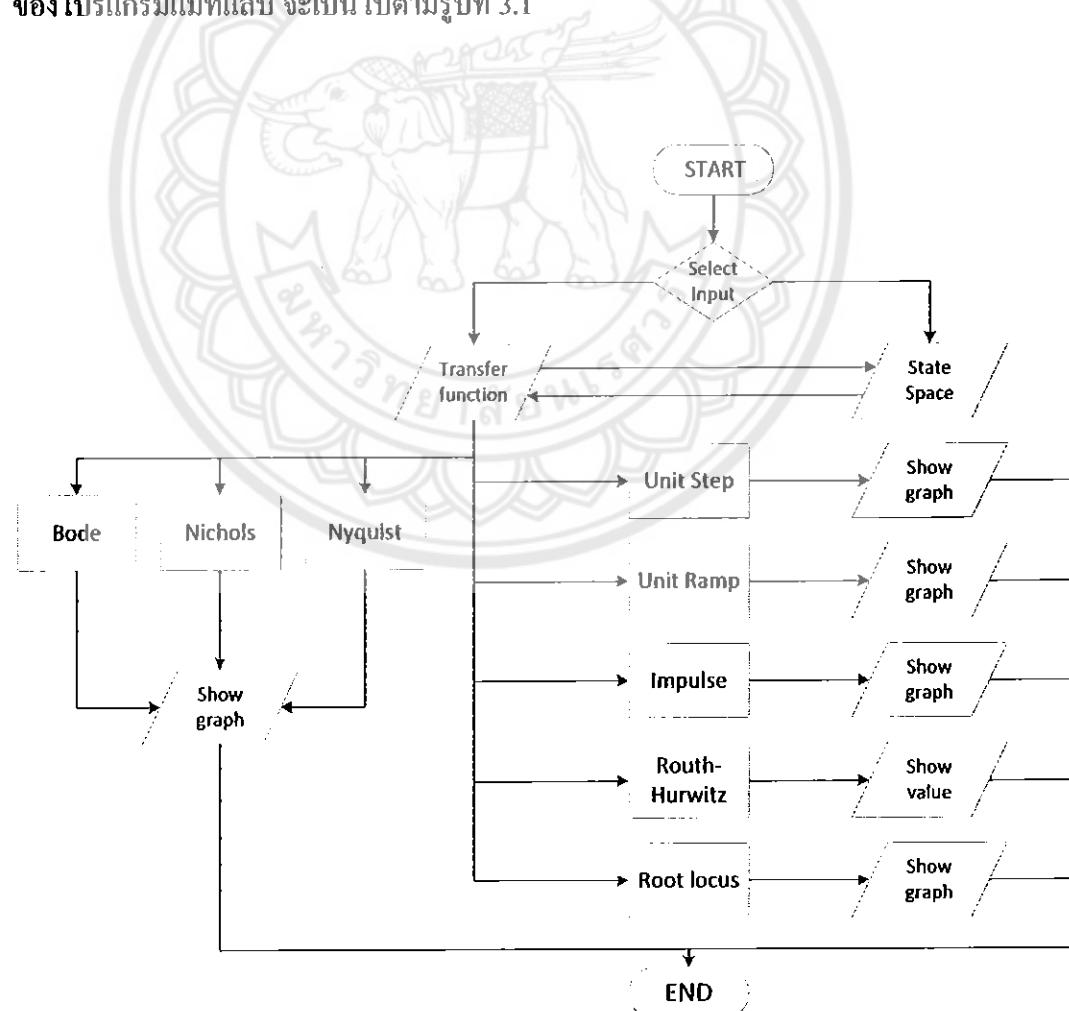
บทที่ 3

วิธีการดำเนินโครงการ

ในบทนี้จะอธิบายถึงวิธีการเขียนโปรแกรมโดยใช้ส่วนต่อประสานกราฟิกกับผู้ใช้ โดยใช้ไกด์ (GUIDE) ซึ่งจะได้ไฟล์แต่ละหน้าต่างออกมา 2 ไฟล์ โปรแกรมจะสร้างขึ้นมา 7 หน้าต่าง โดย 3 หน้าต่างหลักในการป้อนข้อมูลและ 4 หน้าต่างย่อยในการแสดงผล วิธีการเขียนโปรแกรมจะอธิบายได้ดังนี้

3.1 รูปแบบการทำงานของโปรแกรม

สำหรับผู้ใช้งานของโปรแกรมระบบควบคุมที่เขียนจากส่วนต่อประสานกราฟิกกับผู้ใช้ของโปรแกรมเมื่อแล็บ จะเป็นไปตามรูปที่ 3.1



รูปที่ 3.1 ผู้ใช้งานสำหรับขั้นตอนการทำงานของโปรแกรม

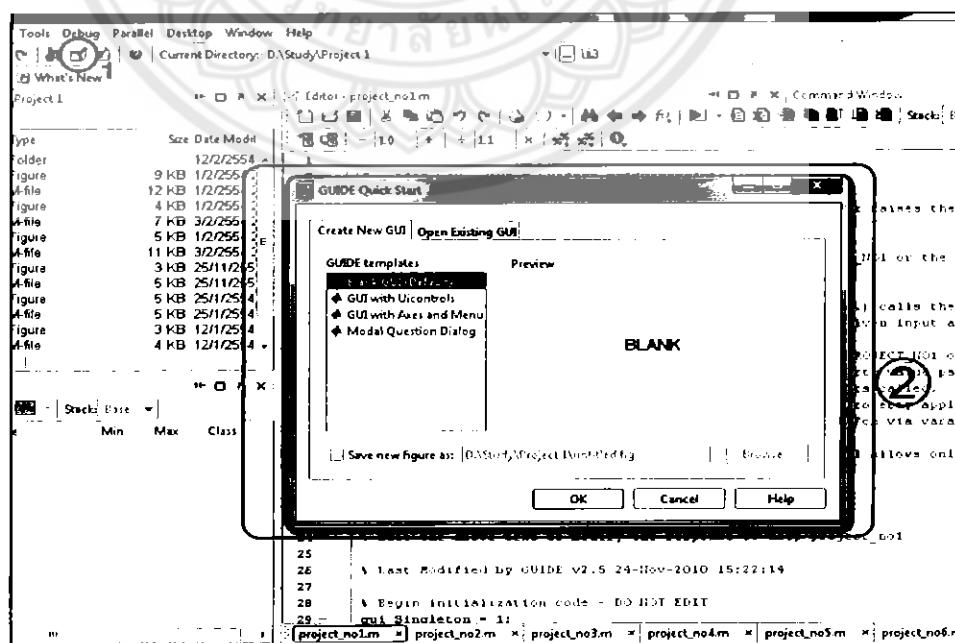
การทำงานของโปรแกรม เมื่อเริ่มต้นโปรแกรมขึ้นมา ขั้นตอนแรกผู้ใช้ต้องทำการเลือก ว่าต้องการป้อนค่า เป็นฟังก์ชันถ่ายโอนของระบบ หรือ ปริภูมิสถานะ โดยทั้งคู่สามารถแปลง กลับไปกลับมาได้ เมื่อผู้ใช้ทำการป้อนค่าเสร็จแล้ว โปรแกรมจะใช้ฟังก์ชันถ่ายโอนในการนำมานำ คำนวณ โดยผู้ใช้สามารถเลือกการแสดงผลที่ผู้ใช้ต้องการ โดยการเลือกที่ปุ่มคำสั่งนั้นๆ ใน โปรแกรม จากนั้น โปรแกรมจะคำนวณและแสดงค่าที่ผู้ใช้ต้องการออกมานา

3.2 การเขียนโปรแกรม

สำหรับการเขียน โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ในระบบควบคุมนี้ จะใช้ หน้าต่าง โปรแกรม จำนวน 7 หน้าต่าง โดยมีหน้าต่างหลักจำนวน 1 หน้าต่าง เพื่อแสดงค่าผลลัพธ์ ที่ผู้ใช้ต้องการ, หน้าต่างสำหรับป้อนค่าฟังก์ชันถ่ายโอนและปริภูมิสถานะจำนวน 2 หน้าต่าง และ หน้าต่างอื่นๆ ที่จำเป็นต้องใช้ในการใช้งานโปรแกรมอีก 4 หน้าต่าง โดยมีขั้นตอนวิธีการเขียน โปรแกรมโดยใช้ส่วนต่อประสานกราฟิกกับผู้ใช้ด้วยไกด์ ของโปรแกรมแมทแล็บ ดังนี้

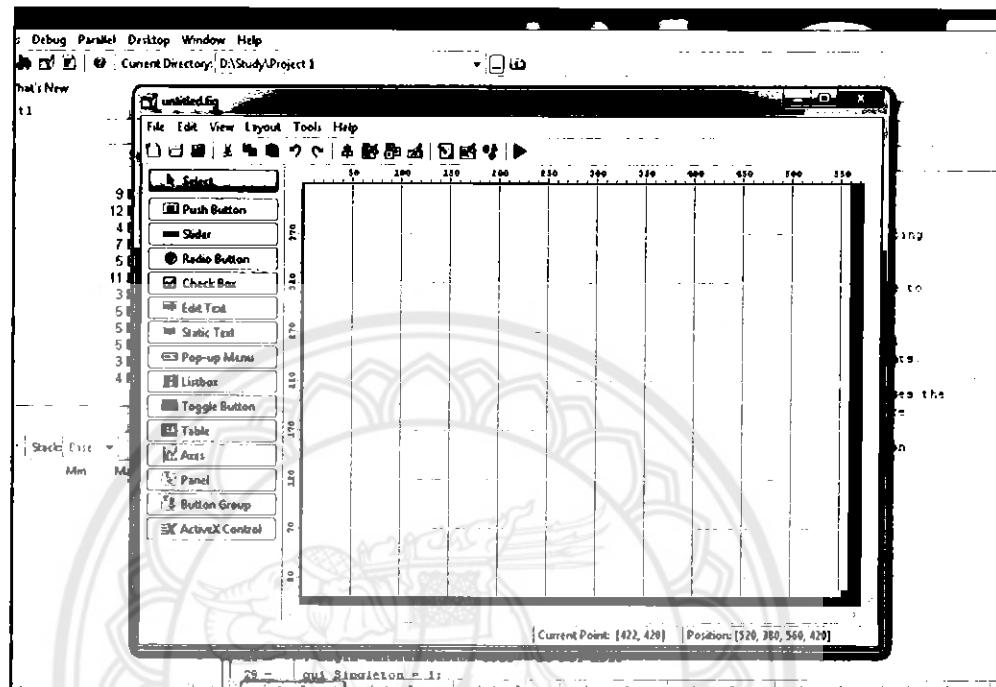
3.2.1 การเขียนโปรแกรมในหน้าต่างหลัก (หน้าต่างที่ 1)

1) เมื่อเปิด โปรแกรมแมทแล็บขึ้นมาแล้ว ให้คลิกที่ไอคอนไกด์ (GUIDE ) ดังในรูป ที่ 3.2 หมายเลข 1 จากนั้นจะได้หน้าต่างของการเริ่มต้นการใช้โปรแกรม (GUIDE Quick Start) ดัง ในรูปที่ 3.2 หมายเลข 2 จากนั้นให้คลิกที่ปุ่มตกลง (OK) เพื่อไปยังขั้นตอนต่อไป



รูปที่ 3.2 การเรียกใช้ GUIDE ของ GUI

หลังจากคลิกที่ปุ่มตกลง (OK) จะได้หน้าต่างใหม่ คือหน้าต่างที่จะไว้สร้างหน้าตาของโปรแกรม หน้าต่างใหม่จะมีลักษณะที่ว่างเปล่า ดังรูปที่ 3.3



รูปที่ 3.3 หน้าต่างของโปรแกรม GUIDE เริ่มต้น

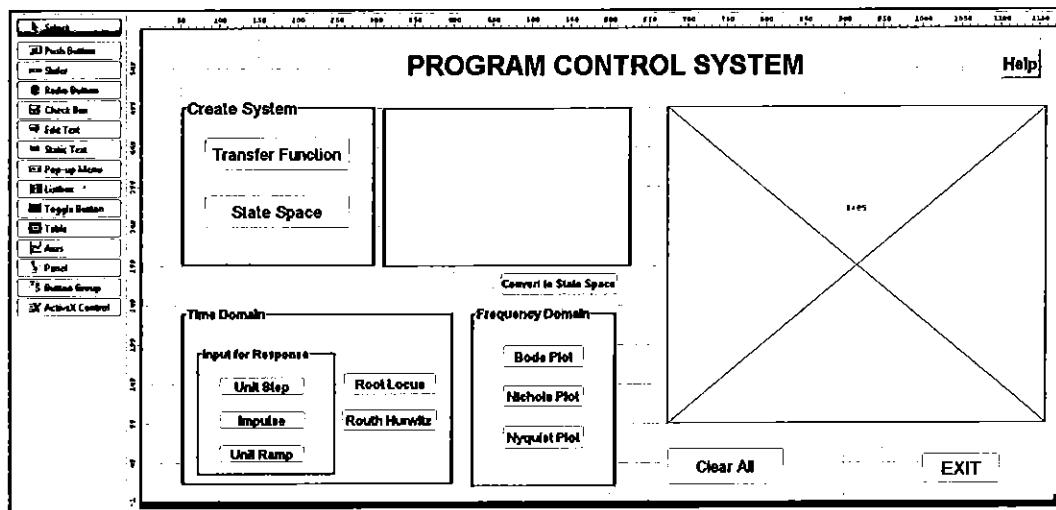
2) ออกแบบหน้าตาโปรแกรมและวางแผนวัตถุต่างๆดัง รูปที่ 3.4 เพื่อเป็นหน้าโปรแกรมที่ไว้แสดงผลต่างๆ ที่ต้องการ สำหรับการแก้ไขคุณสมบัติของแต่ละวัตถุทำได้โดย

คลิกขวาที่ปุ่มที่ต้องการแก้ไข

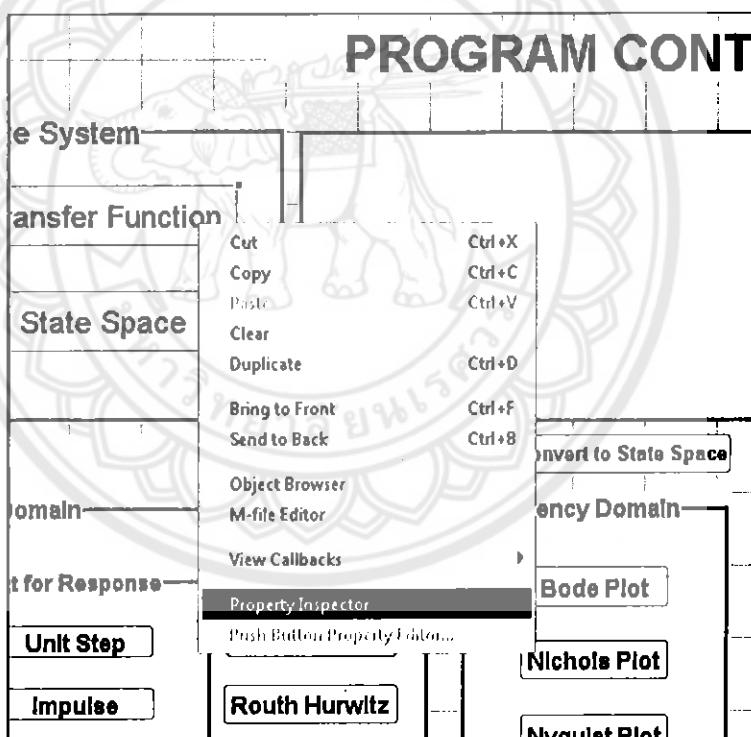
คลิกที่คำสั่งตรวจสอบคุณสมบัติ (Property Inspector) ดังรูปที่ 3.5

เมื่อคลิกที่คำสั่งตรวจสอบคุณสมบัติแล้วจะได้หน้าต่างแก้ไขคุณสมบัติ ดังรูปที่ 3.6

แก้ไขคุณสมบัติของปุ่มนั้นๆตามต้องการ เช่นชื่อ ขนาดตัวอักษร เป็นต้น โดยที่สำคัญที่สุด คือ ชื่อของวัตถุ (Tag) ที่ไว้เรียกใช้งานวัตถุนั้น



รูปที่ 3.4 หน้าตาโปรแกรมหลักที่ออกแบบ



รูปที่ 3.5 คำสั่งเรียกใช้การแก้ไขคุณสมบัติ

FontAngle	normal
FontName	MS Sans Serif
FontSize	12.0
FontUnits	points
FontWeight	bold
ForegroundColor	
HandleVisibility	on
HitTest	on
HorizontalAlignment	center
Interruptible	on
KeyPressFcn	
ListboxTop	1.0
Max	1.0
Min	0.0
Position	[5 11.462 21 2.308]
SelectionHighlight	on
SliderStep	[0.01 0.1]
String	
Style	pushbutton
Tag	bode

รูปที่ 3.6 การแก้ไขคุณสมบัติของแต่ละวัตถุ

3) บันทึกหน้าต่างโปรแกรมที่ได้สร้างไว้ (ชื่อของหน้าต่างมีความสำคัญมาก เพราะจะต้องใช้ชื่อในการเรียกโปรแกรมขึ้นมาใช้)

4) แก้ไขคำสั่งการทำงานของแต่ละวัตถุ เรียกว่า การแก้ไขฟังก์ชันเรียกกลับ (Callback) เป็นการเขียนคำสั่งเพิ่มลงไประเพื่อให้โปรแกรมนั้นทำงานตามที่ต้องการ หลังจากวัตถุนั้นถูกเรียกใช้ โดยมีวิธีการทำดังนี้

คลิกขวาที่วัตถุที่ต้องการแก้ไข

เลือกคำสั่งคูฟังก์ชันเรียกกลับ (View Callbacks) จากนั้นเลือกฟังก์ชันเรียกกลับ (Callback) ตั้งรูปที่ 3.7 โปรแกรมสร้างโค้ดของฟังก์ชันนี้ขึ้นโดยอัตโนมัติ ดังรูปที่ 3.8 ในไฟล์นามสกุล .m ที่มีชื่อเดียวกันกับที่ได้ตั้งไว้ โดยให้แต่ละวัตถุมีหน้าที่ ดังนี้

- ปุ่มฟังก์ชันถ่ายโอน(Transfer Function) ทำหน้าที่เปิดหน้าต่างฟังก์ชันถ่ายโอน (หน้าต่างที่ 2) เพื่อรับค่าฟังก์ชันถ่ายโอนจากผู้ใช้ สร้างโดยใช้ ปุ่มกด (Push Button)

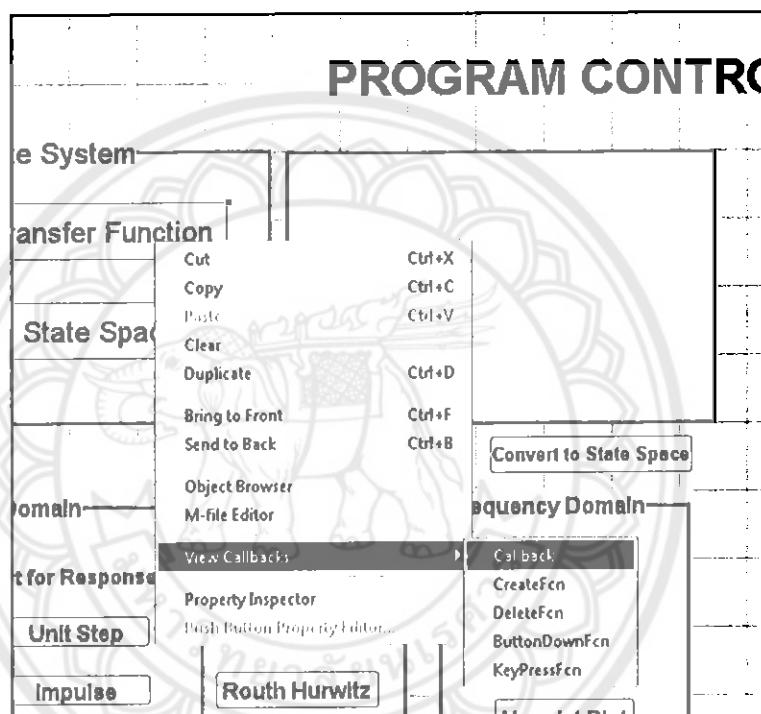
- ปุ่มปริภูมิสถานะ(State Space) ทำหน้าที่เปิดหน้าต่างปริภูมิสถานะ (หน้าต่างที่ 3) เพื่อรับค่าปริภูมิสถานะจากผู้ใช้ สร้างโดยใช้ ปุ่มกด (Push Button)

- ปุ่มแปลงค่าจากฟังก์ชันถ่ายโอนเป็นปริภูมิสถานะ (Convert to state space) ทำหน้าที่แปลงค่าจากฟังก์ชันถ่ายโอนเป็นปริภูมิสถานะและส่งค่าไปยังหน้าต่างที่ 7 เพื่อแสดงผลลัพธ์ สร้างโดยใช้ปุ่มกด (Push Button)
- กราฟแสดงผล ทำหน้าที่แสดงกราฟที่ได้จากการคำนวณ สร้างโดยใช้วัสดุแสดงผลกราฟิก (Axis)
- ปุ่มผลตอบสนองของระบบต่อสัญญาณระดับ (Unitstep response) ทำหน้าที่วัดรูปกราฟผลตอบสนองของระบบต่อสัญญาณระดับจากฟังก์ชันถ่ายโอนหรือปริภูมิสถานะสร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มผลตอบสนองของระบบต่อสัญญาณอินพัลส์ (Impulse response) ทำหน้าที่วัดรูปกราฟผลตอบสนองของระบบต่อสัญญาณอินพัลส์จากฟังก์ชันถ่ายโอนหรือปริภูมิสถานะ สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มผลตอบสนองของระบบต่อสัญญาณลาด (Unitramp response) ทำหน้าที่วัดรูปกราฟผลตอบสนองของระบบต่อสัญญาณลาดจากฟังก์ชันถ่ายโอนหรือปริภูมิสถานะ สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มเส้นทางเดินของราก (Root locus) ทำหน้าที่วัดรูปเส้นทางเดินของรากจากฟังก์ชันถ่ายโอนหรือปริภูมิสถานะ สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มเร้าท์เชอร์วิช (Routh-Hurwitz) ทำหน้าที่แสดงการหาความเสถียรของระบบจากสมการลักษณะเฉพาะของฟังก์ชันถ่ายโอน สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มวัดกราฟเสถียรภาพของโบดี (Bode plot) ทำหน้าที่วัดรูปกราฟจากฟังก์ชันโบดี สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มวัดกราฟเสถียรภาพของไนคิสต์ (Nyquist plot) ทำหน้าที่วัดกราฟจากฟังก์ชันไนคิสต์ สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มวัดกราฟเสถียรภาพของนิโคลส์ (Nichols plot) ทำหน้าที่วัดกราฟจากฟังก์ชันนิโคลส์ สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงวิธีการใช้งานโปรแกรม สร้างโดยใช้ปุ่มกด (Push Button)

- ปุ่มล้างข้อมูลทุกอย่าง (Clear all) ทำหน้าที่ลบข้อมูลที่ผู้ใช้ได้ป้อนค่าเข้าไปสร้างโดยใช้ปุ่มกด (Push Button)

- ปุ่มออก (EXIT) ทำหน้าที่ออกจากโปรแกรมและปิดโปรแกรมลง สร้างโดยใช้ปุ่มกด (Push Button)

บันทึกการแก้ไข



รูปที่ 3.7 การแก้ไขฟังก์ชันเรียกกลับ

```

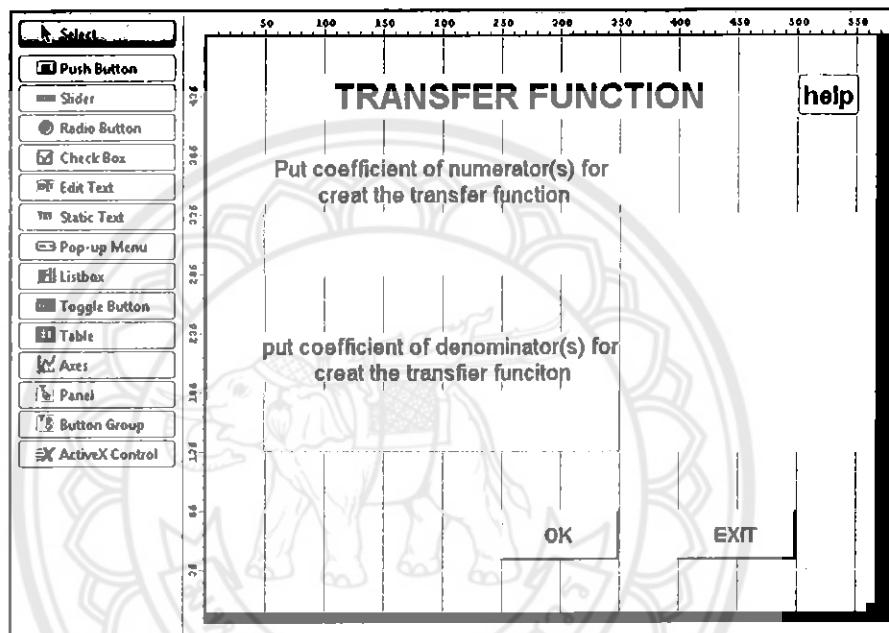
80
81 % --- Executes on button press in transfer.
82 %|function transfer_Callback(hObject, eventdata, handles)
83 %|% hObject    handle to transfer (see GCBO)
84 %|% eventdata reserved - to be defined in a future version of MATLAB
85 %|% handles    structure with handles and user data (see GUIDATA)
86 - project_no2;
87
88 % --- Executes on button press in state.
89 %|function state_Callback(hObject, eventdata, handles)
90 %|% hObject    handle to state (see GCBO)
91 %|% eventdata reserved - to be defined in a future version of MATLAB
92 %|% handles    structure with handles and user data (see GUIDATA)
93 - project_no3;
94

```

รูปที่ 3.8 แก้ไขคำสั่งการทำงานต่างๆของปุ่มที่สร้างขึ้น

3.2.2 การเขียนโปรแกรมในหน้าต่างฟังก์ชันถ่ายโอน (หน้าต่างที่ 2)

- 1) ทำตามขั้นตอนที่ 1 ของหน้าต่างที่ 1 หัวข้อ 3.2.1
- 2) ออกแบบหน้าต่างโปรแกรมและแก้ไขคุณสมบัติต่างๆของเครื่องมือได้ดังรูปที่ 3.9



รูปที่ 3.9 หน้าต่างของฟังก์ชันถ่ายโอน

3) แก้ไขฟังก์ชันเรียกกลับของแต่ละปุ่มโดยทำตามขั้นตอนที่ 3.2.1.4 และโดยให้แต่ละปุ่มทำหน้าที่ ดังนี้

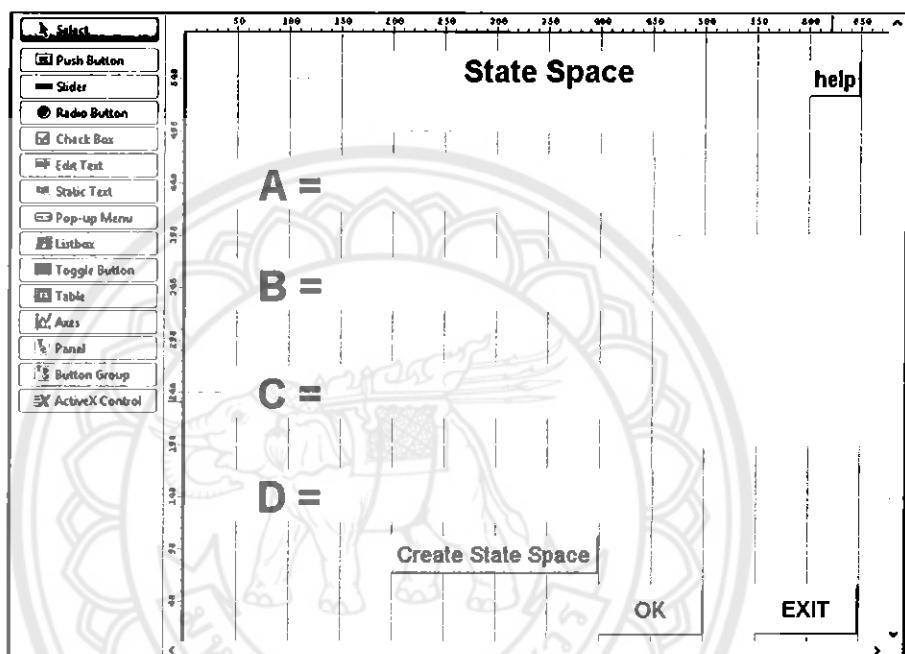
- ซ่องรับค่าจากผู้ใช้ ทำหน้าที่รับค่าสัมประสิทธิ์ของฟังก์ชันถ่ายโอนเพื่อนำไปใช้ในการคำนวณ สร้างโดยใช้วัตถุที่สามารถแก้ไขข้อความได้ (Edit text)
- ปุ่มตกลง (OK) จะทำหน้าที่นำค่าจาก Edit text มาคำนวณเป็นฟังก์ชันถ่ายโอนและส่งค่าไปที่หน้าต่างหลัก (หน้าต่างที่ 1) สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มออก (EXIT) ใช้สำหรับออกจากหน้าต่างโปรแกรม สร้างโดยใช้ปุ่มกด (Push Button)

- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงวิธีการป้อนค่าฟังก์ชันถ่ายโอนและการใช้งานโปรแกรม สร้างโดยใช้ปุ่มกด (Push Button)

3.2.3 การเขียนโปรแกรมในหน้าต่างปริภูมิสถานะ (หน้าต่างที่ 3)

1) ตามขั้นตอนที่ 1 ของหน้าต่างที่ 1 หัวข้อ 3.2.1

2) ออกแบบหน้าต่างโปรแกรม และแก้ไขคุณสมบัติต่างๆของเครื่องมือได้ดังรูปที่ 3.10



รูปที่ 3.10 หน้าต่างของปริภูมิสถานะ

3) แก้ไขฟังก์ชันเรียกกลับของแต่ละปุ่มโดยตามขั้นตอนที่ 3.2.1.4 และโดยให้แต่ละปุ่มทำหน้าที่ดังนี้

- ช่องรับค่าจากผู้ใช้ ทำหน้าที่รับค่าเมมตริกจากผู้ใช้สร้างโดยใช้วัตถุที่สามารถแก้ไขข้อความได้ (Edit text)

- ปุ่มสร้างปริภูมิสถานะ (Create State Space) ทำหน้าที่สร้างปริภูมิสถานะและแสดงให้ผู้ใช้ทราบ สร้างโดยใช้ปุ่มกด (Push Button)

- ปุ่มตกลง (OK) จะทำหน้าที่แปลงค่าจากปริภูมิสถานะเป็นฟังก์ชันถ่ายโอนและส่งค่าไปยังหน้าต่างหลักของโปรแกรมเพื่อใช้ในการคำนวณ สร้างโดยใช้ปุ่มกด (Push Button)

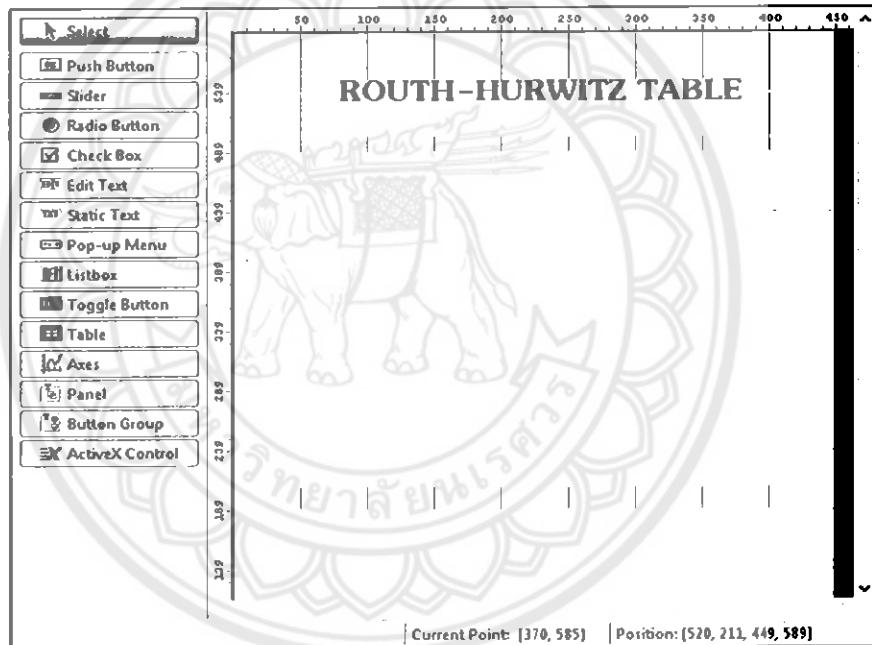
- ปุ่มออก (EXIT) ใช้สำหรับออกจากหน้าต่างโปรแกรม สร้างโดยใช้ปุ่มกด (Push Button)

- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงวิธีการป้อนค่าฟังก์ชันถ่ายโอนและการใช้งานโปรแกรม สร้างโดยใช้ปุ่มกด (Push Button)

3.2.4 การเขียนโปรแกรมในหน้าต่างแสดงเร้าท์ເຊອຣົວິຟ (หน้าต่างที่ 4)

1) ตามขั้นตอนที่ 1 ของหน้าต่างที่ 1 หัวข้อ 3.2.1

2) ออกแบบหน้าต่างโปรแกรมและแก้ไขคุณสมบัติต่างๆของเครื่องมือได้ดัง รูปที่ 3.11



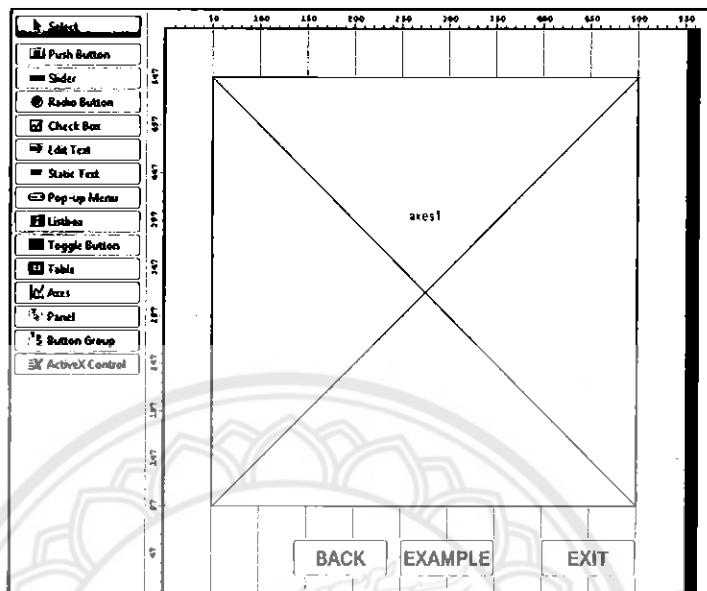
รูปที่ 3.11 หน้าต่างแสดงการหาเสถียรภาพของเร้าท์ເຊອຣົວິຟ

3).แก้ไขคำสั่งของโปรแกรมให้หน้าต่างนี้แสดงการหาเสถียรภาพของเร้าท์ເຊອຣົວິຟ โดยรับค่ามาจากหน้าต่างโปรแกรมหลัก (หน้าต่างที่ 1) หลังจากปุ่มเร้าท์ເຊອຣົວິຟ (Routh-Hurwitz) ถูกกด สร้างโดยใช้วัตถุแสดงผลที่ไม่สามารถแก้ไขได้ (Static Text)

3.2.5 การเขียนโปรแกรมในหน้าต่างแสดงช่วยเหลือ (หน้าต่างที่ 5)

1) ตามขั้นตอนที่ 1 ของหน้าต่างที่ 1 หัวข้อ 3.2.1

2) ออกรูปแบบหน้าต่างโปรแกรมและแก้ไขคุณสมบัติต่างๆของเครื่องมือได้ ดังรูปที่ 3.12



รูปที่ 3.12 หน้าต่างแสดงการช่วยเหลือ

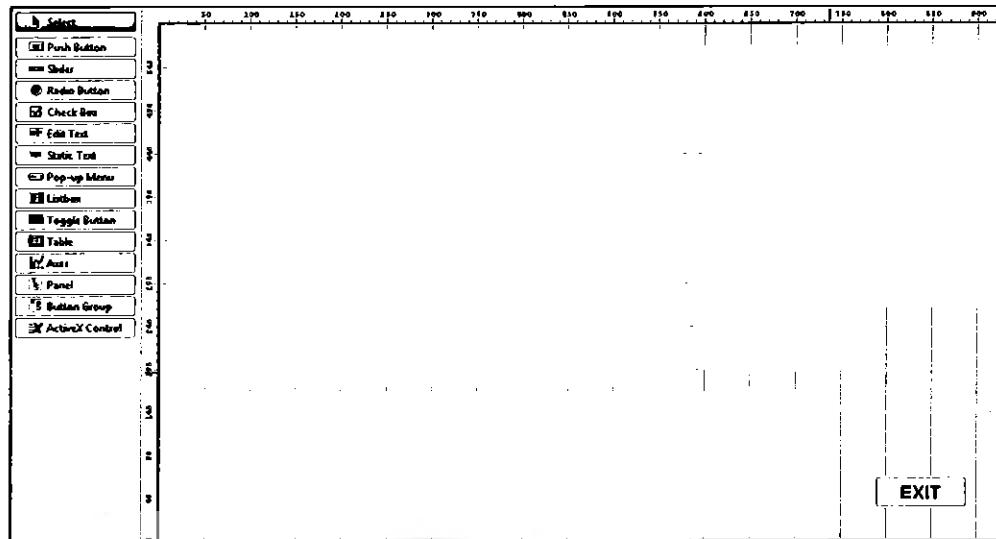
3) แก้ไขคำสั่งของโปรแกรมให้หน้าต่างนี้แสดงคำอธิบายโปรแกรม และช่วยเหลือผู้ใช้จากปุ่มช่วยเหลือ (Help) จากหน้าต่างที่ 1, 2 และ 3 โดยรับค่าจากหน้าต่างดังกล่าว แล้วจึงเปิดภาพแสดงคำอธิบายโปรแกรมตามหน้าต่างดังกล่าว โดยมีปุ่มที่สำคัญเพิ่มเติมดังนี้

- ปุ่มตัวอย่าง (Example) ทำหน้าที่แสดงตัวอย่างการใช้งานโปรแกรม สร้างโดยใช้ปุ่มกด (Push Button)
- ปุ่มย้อนกลับ (Back) ทำหน้าที่ย้อนกลับไปที่คำอธิบายการใช้งาน สร้างโดยใช้ปุ่มกด (Push Button)

3.2.6 การเขียนโปรแกรมในหน้าต่างแสดงค่าปริภูมิสถานะ (หน้าต่างที่ 6)

1) ตามขั้นตอนที่ 1 ของหน้าต่างที่ 1 หัวข้อ 3.2.1

2) ออกรูปแบบหน้าต่างโปรแกรมและแก้ไขคุณสมบัติต่างๆของเครื่องมือได้ ดังรูปที่ 3.13



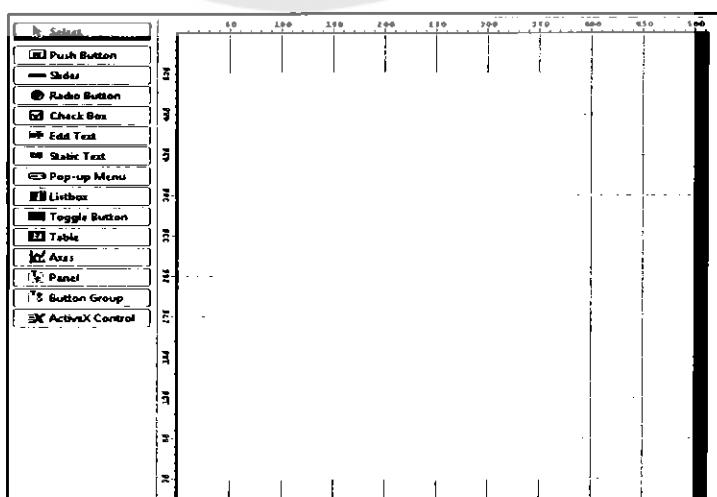
รูปที่ 3.13 หน้าต่างแสดงปริภูมิสถานะ

3) แก้ไขคำสั่งของโปรแกรมให้หน้าต่างนี้แสดงปริภูมิสถานะที่รับค่าจากผู้ใช้หลังจากปุ่มสร้างปริภูมิสถานะ (Create State Space) ในหน้าต่างที่ 3 ถูกกด

3.2.7 การเขียนโปรแกรมในหน้าต่างแสดงค่าปริภูมิสถานะ (หน้าต่างที่ 7)

1) ตามขั้นตอนที่ 1 ของหน้าต่างที่ 1 หัวข้อ 3.2.1

2) ออกแบบหน้าต่างโปรแกรม และแก้ไขคุณสมบัติต่างๆของเครื่องมือได้ ดังรูปที่ 3.14 สร้างโดยใช้ วัตถุแสดงผลที่ไม่สามารถแก้ไขได้ (Static Text) ในการแสดงข้อความโดยที่ผู้ใช้ไม่สามารถแก้ไขได้



รูปที่ 3.14 หน้าต่างแสดงปริภูมิสถานะ

3) แก้ไขคำสั่งของโปรแกรมให้หน้าต่างนี้แสดงปริภูมิสถานะที่ส่งค่ามาจากหน้าต่างที่ 1 หลังจากปุ่มแปลงไปเป็นปริภูมิสถานะ (Convert to State Space) ในหน้าต่างที่ 1 ลูกกอก เพื่อแปลงค่าจากพังก์ชันถ่ายโอนไปเป็นปริภูมิสถานะ สร้างโดยใช้วัตถุแสดงผลที่ไม่สามารถแก้ไขได้ (Static Text)

3.2.8 การแปลงโปรแกรม (Compile) เพื่อนำไปใช้โดยไม่ต้องติดตั้งโปรแกรม MATLAB โดยใช้ deploytool

1) การตั้งค่าตัวแปลงโปรแกรม โดยพิมพ์ mbuild -setup ในหน้าต่างคำสั่ง (Command Window) จากนั้นเลือกตัวแปรโปรแกรมเป็น MS Visual C++ ตามขั้นตอนจนถึงเสร็จสมบูรณ์ (Done) ดังรูปที่ 3.15

```

>> mbuild -setup
Please choose your compiler for building standalone MATLAB applications:

Would you like mbuild to locate installed compilers [y]/n? y

Select a compiler:
[1] Lcc-win32 C 2.4.1 in C:\PROGRA~1\MATLAB\R2008a\sys\lcc
[2] Microsoft Visual C++ 2008 in C:\Program Files\Microsoft Visual Studio 9.0

[0] None

Compiler: 2

Please verify your choices:
Compiler: Microsoft Visual C++ 2008
Location: C:\Program Files\Microsoft Visual Studio 9.0

Are these correct [y]/n? y

Warning: Applications/components generated using Microsoft Visual Studio
2008 require that the Microsoft Visual Studio 2008 run-time
libraries be available on the computer used for deployment.
To redistribute your applications/components, be sure that the
deployment machine has these run-time libraries.

Trying to update options file: C:\Users\BangPan9K\AppData\Roaming\MathWorks\MATLAB\R2008a\compopts.bat
From template:          C:\PROGRA~1\MATLAB\R2008a\bin\win32\mbuildopts\msvc9compp.bat

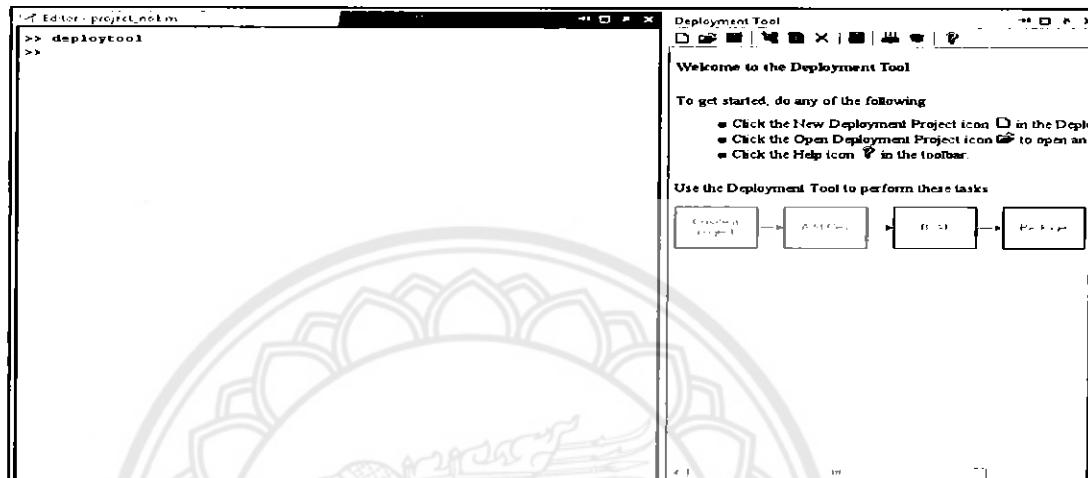
Done . .

```

รูปที่ 3.15 วิธีการตั้งค่าตัวแปรโปรแกรม

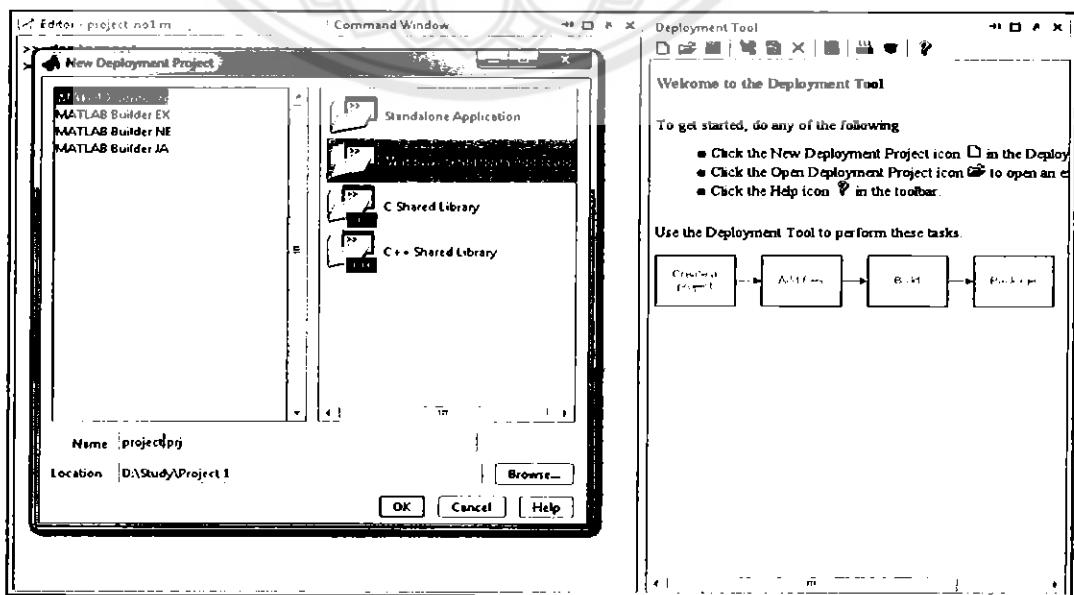
2) ขั้นตอนการแปลงโปรแกรมโดยคำสั่ง deploytool

- พิมพ์คำสั่ง deploytool ในหน้าต่างคำสั่ง Command Window และโปรแกรมการแปลงโปรแกรมจะแสดงขึ้นมาดังรูปที่ 3.16



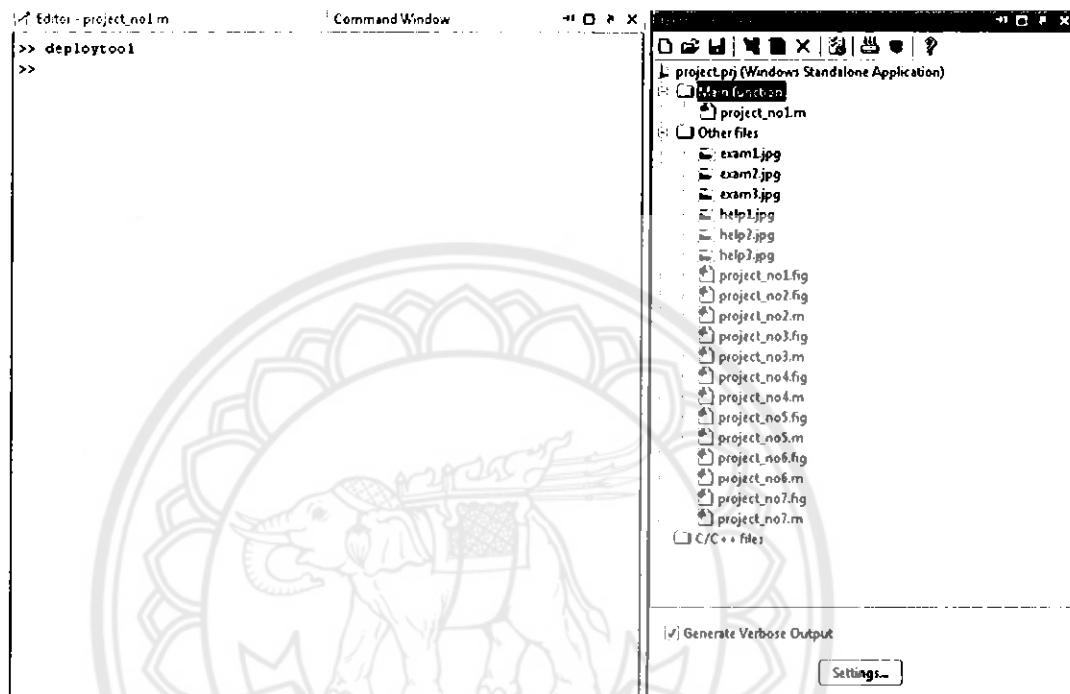
รูปที่ 3.16 ค่าตัวแปร โปรแกรม

- คลิก เพื่อเริ่มต้นการแปลงโปรแกรม จากนั้นให้ตั้งชื่อ โปรแกรม เลือก Window Standalone Application ดังรูปที่ 3.17



รูปที่ 3.17 การแปลงโปรแกรมให้ทำงานบนพื้นฐาน Window

- เลือก Main function คลิกที่ (Add File) เพื่อเลือกหน้าต่างแรกหลังจากที่โปรแกรมถูกเปิดขึ้นมา แล้วเลือก other files จากนั้นให้เพิ่มไฟล์ที่จำเป็นของโปรแกรมเข่น ไฟล์รูปภาพ ดังรูปที่ 3.18



รูปที่ 3.18 การเพิ่มไฟล์ต่างๆ ของโปรแกรม

- จากนั้นให้คลิกที่ Build the project เพื่อเริ่มการแปลงโปรแกรม จากนั้นรอจนสำเร็จจะได้ไฟล์ที่มีนามสกุลไฟล์ .EXE เพื่อนำไปใช้ ดังรูปที่ 3.19

<input type="checkbox"/>	build	03/02/54 10:10 PM	Text Document 15 KB
<input type="checkbox"/>	mccExcludedFiles	03/02/54 10:09 PM	Text Document 1,093 KB
<input checked="" type="checkbox"/>	project_control	03/02/54 10:10 PM	Application 4,184 KB
<input checked="" type="checkbox"/>	project_control_main	03/02/54 10:10 PM	C Source 5 KB
<input checked="" type="checkbox"/>	project_control_mcc_component_data	03/02/54 10:10 PM	C Source 9 KB
<input type="checkbox"/>	readme	03/02/54 10:10 PM	Text Document 10 KB

รูปที่ 3.19 ไฟล์ที่ได้จากการแปลงโปรแกรม

บทที่ 4

ผลการดำเนินโครงการ

ในบทนี้จะแสดงผลการทดลอง เมื่อเราทดลองใส่ฟังก์ชันถ่ายโอนของระบบ หรือปริญมิ-สถานะของระบบในโปรแกรมระบบควบคุม ซึ่งสามารถดูค่ากราฟผลตอบสนองของระบบเมื่อป้อนสัญญาณแบบสัญญาณระดับ สัญญาณลาก และสัญญาณอินพัลส์ ว่าค่ากราฟเส้นทางเดินของรากในระบบนั้นๆ และยังสามารถหาสตีเบอร์ภาพจากเร้าที่酵อร์วิช หาผลตอบสนองทางความถี่ได้อีกด้วย

4.1 การเริ่มต้นการใช้งานโปรแกรม

สำหรับการใช้งานโปรแกรมระบบควบคุมที่ได้เขียนขึ้นมา สำหรับเครื่องคอมพิวเตอร์ ที่ต้องการใช้โปรแกรมระบบควบคุมโดยที่ไม่มีโปรแกรมแม่ทั้งหมด (MATLAB) ติดตั้งอยู่ จำเป็นต้องมีการติดตั้งโปรแกรมแปลภาษาของแม่ทั้งหมด (MCRInstaller) ก่อนที่จะมีการเรียกใช้งานโปรแกรมที่ผ่านการแปลให้ใช้งานบนพื้นฐานวินโดว์ (Window) แล้ว โดยวิธีการติดตั้งโปรแกรมแปลภาษาของแม่ทั้งหมด (MCRInstaller) เมื่อกับการติดตั้งโปรแกรมทั่วไป แต่สำหรับเครื่องคอมพิวเตอร์ที่มีโปรแกรมแม่ทั้งหมด ติดตั้งอยู่แล้วสามารถเรียกใช้งานโปรแกรมผ่านโปรแกรมแม่ทั้งหมด “ได้โดยใช้ไฟล์ .mg และ .m” ที่ได้เขียนขึ้นมา ใช้งานผ่านโดยโปรแกรมแม่ทั้งหมด หรือ ใช้งานโปรแกรมที่ผ่านการแปลให้ใช้งานบนพื้นฐานวินโดว์ แล้วก็ได้เช่นกัน

4.2 โปรแกรมระบบควบคุมโดยการป้อนฟังก์ชันถ่ายโอนจากผู้ใช้

รายละเอียดของโปรแกรมและขั้นตอนการทำงานของโปรแกรม

1) ในขั้นตอนแรกเมื่อทำการเปิดโปรแกรมขึ้นมา จะพบกับหน้าต่างหลักของโปรแกรม ดังรูปที่ 4.1

2) คลิกที่ปุ่มฟังก์ชันถ่ายโอน (Transfer Function) โปรแกรมในหน้าต่างที่ 2 ของโปรแกรมจะปรากฏขึ้นมาเพื่อรับค่าฟังก์ชันถ่ายโอน ดังรูปที่ 4.2 โดยมีรายละเอียดของโปรแกรมดังนี้

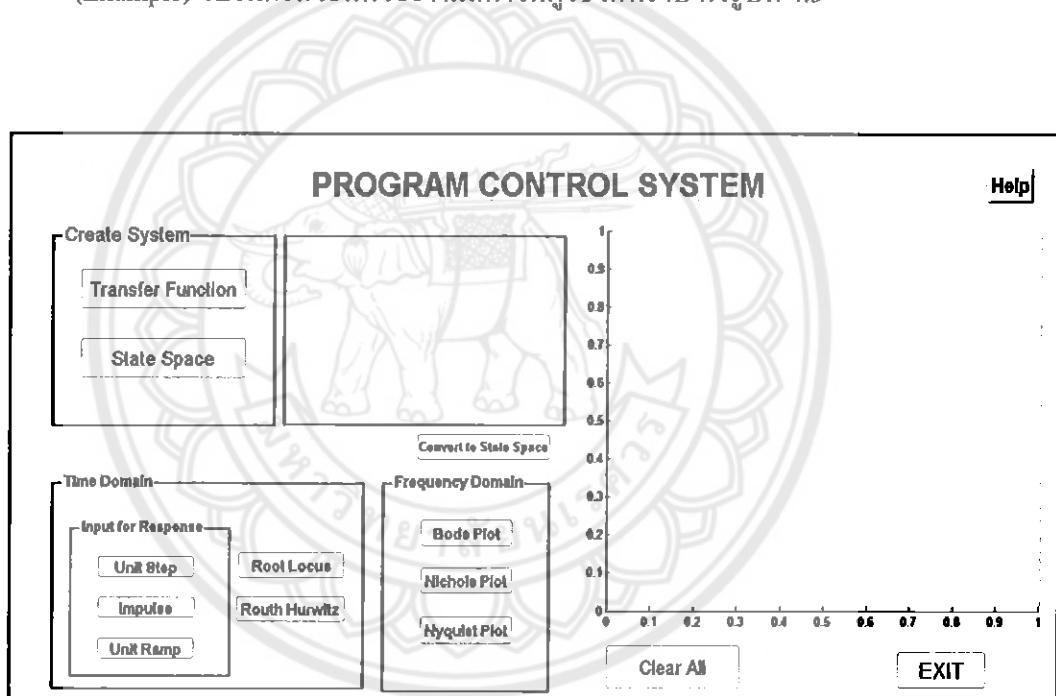
- ช่องใส่ค่าสัมประสิทธิ์ของตัวเศษ (coefficient of numerator) สำหรับป้อนค่าสัมประสิทธิ์ของตัวเศษของฟังก์ชันถ่ายโอน เมื่อนำตัวชี้ตำแหน่ง (cursor) ไปวางที่ช่องป้อนค่า จะมีคำอธิบายการใช้งานเบื้องต้นขึ้นมาให้ผู้ใช้ได้ทราบ

- ช่องใส่ค่าสัมประสิทธิ์ของตัวส่วน (coefficient of denominator) สำหรับป้อนค่าสัมประสิทธิ์ของตัวส่วนของพังก์ชันถ่ายโอน เมื่อนำตัวชี้ตำแหน่ง (cursor) ไปวางที่ช่องป้อนค่า จะมีคำอธิบายการใช้งานเบื้องต้นขึ้นมาให้ผู้ใช้ได้ทราบ

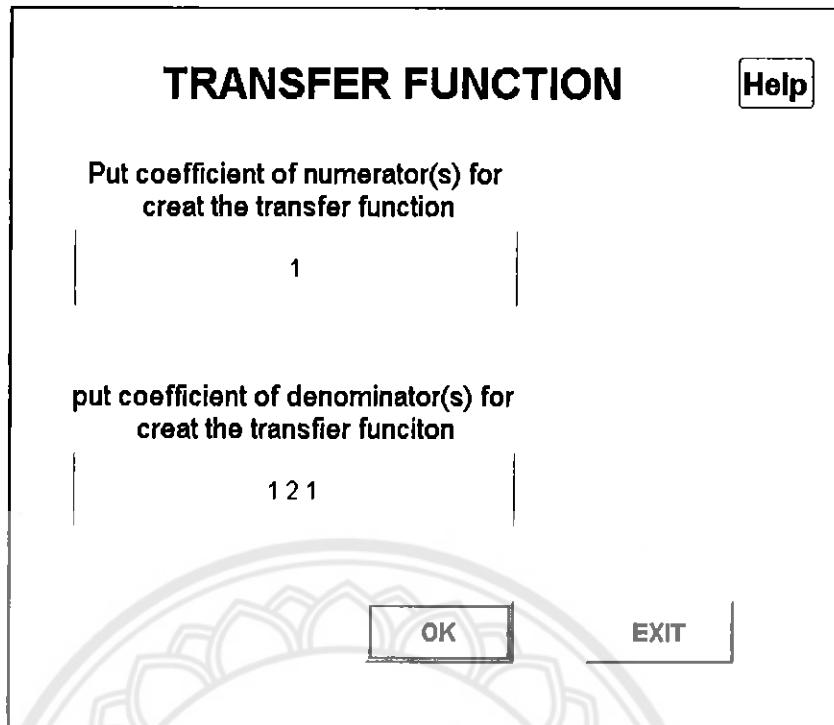
- ปุ่มตกลง (OK) เมื่อถูกคลิกจะส่งค่าพังก์ชันถ่ายโอนไปที่หน้าต่างหลัก รูปที่ 4.1 และแสดงผลที่ช่องแสดงผลพังก์ชันถ่ายโอน ดังรูปที่ 4.3

- ปุ่มออก (EXIT) เมื่อถูกคลิกจะปิดและออกจากหน้าต่างนี้

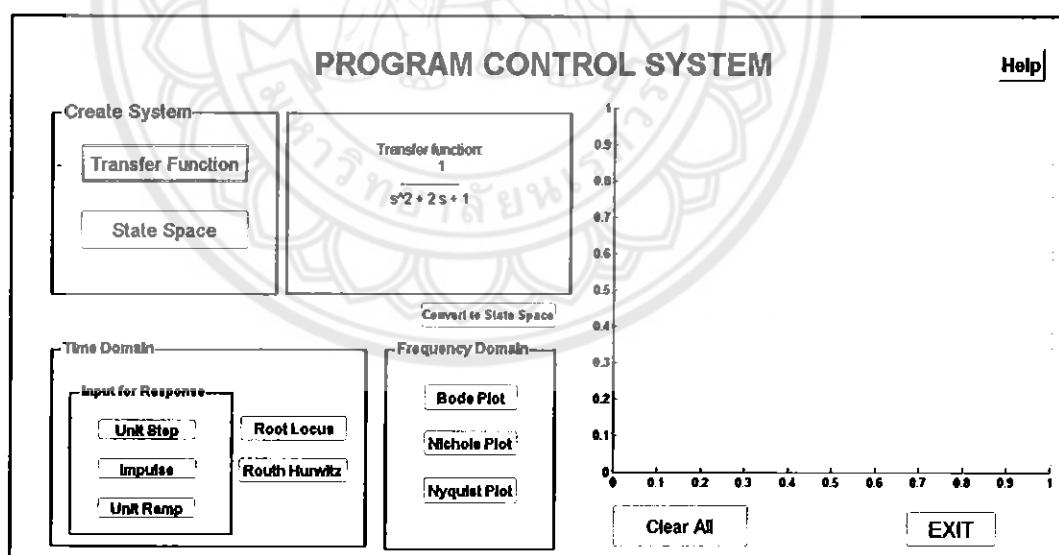
- ปุ่มช่วยเหลือ (Help) เมื่อถูกคลิกจะแสดงหน้าต่างช่วยเหลือในการใช้งานโปรแกรมให้ผู้ใช้ได้ทราบ ดังรูปที่ 4.4 และในหน้าช่วยเหลือ เมื่อผู้ใช้คลิกที่ปุ่มตัวอย่าง (Example) โปรแกรมจะมีตัวอย่างแสดงให้ผู้ใช้ได้ทราบ ดังรูปที่ 4.5



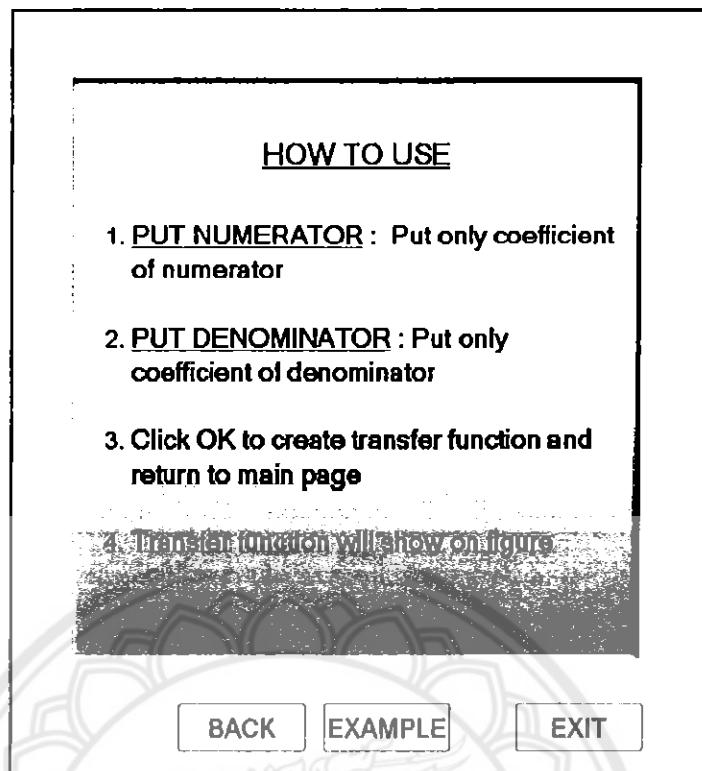
รูปที่ 4.1 หน้าต่างหลักของโปรแกรมเมื่อถูกเรียกขึ้นมา



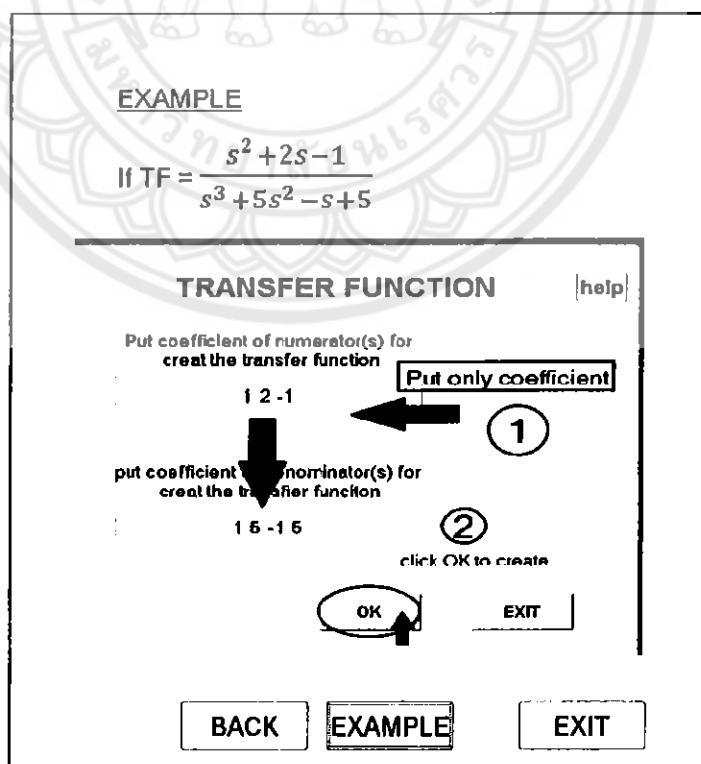
รูปที่ 4.2 หน้าต่างของฟังก์ชันถ่ายโอนเมื่อูกเรียกขึ้นมา



รูปที่ 4.3 แสดงผลเมื่อมีการใส่ค่าฟังก์ชันถ่ายโอน



รูปที่ 4.4 หน้าต่างช่วยเหลือสำหรับการใช้งานการป้อนฟังก์ชันด้วยโอน



รูปที่ 4.5 หน้าต่างตัวอย่างการป้อนค่าฟังก์ชันด้วยโอน

4.3 โปรแกรมระบบควบคุมโดยการป้อนปริภูมิสถานะจากผู้ใช้

รายละเอียดของโปรแกรมและขั้นตอนการทำงานของโปรแกรม

1) ในขั้นตอนแรกเมื่อทำการเปิดโปรแกรมขึ้นมา จะพนักงานห้ามตั้งหลักของโปรแกรม ดังรูปที่ 4.1

2) คลิกที่ปุ่มปริภูมิสถานะ (State Space) โปรแกรมในหน้าต่างที่ 3 ของโปรแกรมจะปรากฏขึ้นมาเพื่อรับค่าปริภูมิสถานะ ดังรูปที่ 4.6 โดยมีรายละเอียดของโปรแกรมดังนี้

- ช่องใส่ค่าแมตริกซ์สถานะ (A) สำหรับป้อนแมตริกขนาด $n \times n$ โดยขนาดไม่เกินแมตริก 5×5 และเมื่อนำตัวชี้ตำแหน่ง (cursor) ไปวางที่ช่อง จะมีคำอธิบายการใช้งานเบื้องต้นขึ้นมาให้ผู้ใช้ได้ดู
- ช่องใส่ค่าแมตริกซ์อินพุต (B) สำหรับป้อนแมตริกขนาด $n \times p$ และเมื่อนำตัวชี้ตำแหน่ง (cursor) ไปวางที่ช่อง จะมีคำอธิบายการใช้งานเบื้องต้นขึ้นมาให้ผู้ใช้ได้ดู
- ช่องใส่ค่าแมตริกซ์เอาต์พุต (C) สำหรับป้อนแมตริกขนาด $q \times n$ และเมื่อนำตัวชี้ตำแหน่ง (cursor) ไปวางที่ช่อง จะมีคำอธิบายการใช้งานเบื้องต้นขึ้นมาให้ผู้ใช้ได้ดู
- ช่องใส่ค่าแมตริกซ์ส่งผ่าน โดยตรง (D) สำหรับป้อนแมตริกขนาด $q \times p$ และเมื่อนำตัวชี้ตำแหน่ง (cursor) ไปวางที่ช่อง จะมีคำอธิบายการใช้งานเบื้องต้นขึ้นมาให้ผู้ใช้ได้ดู
- ปุ่มสร้างปริภูมิสถานะ (Create State Space) เมื่อถูกคลิกจะเรียกหน้าต่างโปรแกรมแสดงการสร้างปริภูมิสถานะว่าถูกต้องหรือไม่
- ปุ่มคลอก (OK) เมื่อถูกคลิกจะส่งค่าไปยังหน้าต่างหลัก โดยจะทำการแปลงจากปริภูมิสถานะไปเป็นฟังก์ชันถ้าป้อนก่อน เพื่อให้ง่ายต่อการคำนวณในหน้าต่างหลักจะได้ผล ดังรูปที่ 4.7
- ปุ่มออก (EXIT) เมื่อถูกคลิกจะปิดและออกจากหน้าต่างนี้

State Space

A = $\begin{bmatrix} -2 & -1 \\ 1 & 0 \end{bmatrix}$

B = $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$

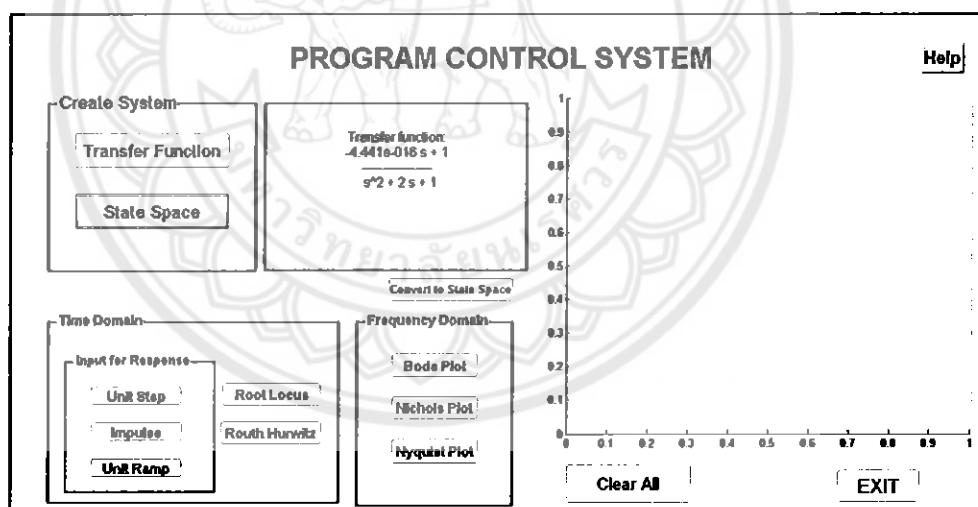
C = $\begin{bmatrix} 0 & 1 \end{bmatrix}$

D = $\begin{bmatrix} 0 \end{bmatrix}$

[Create State Space](#)

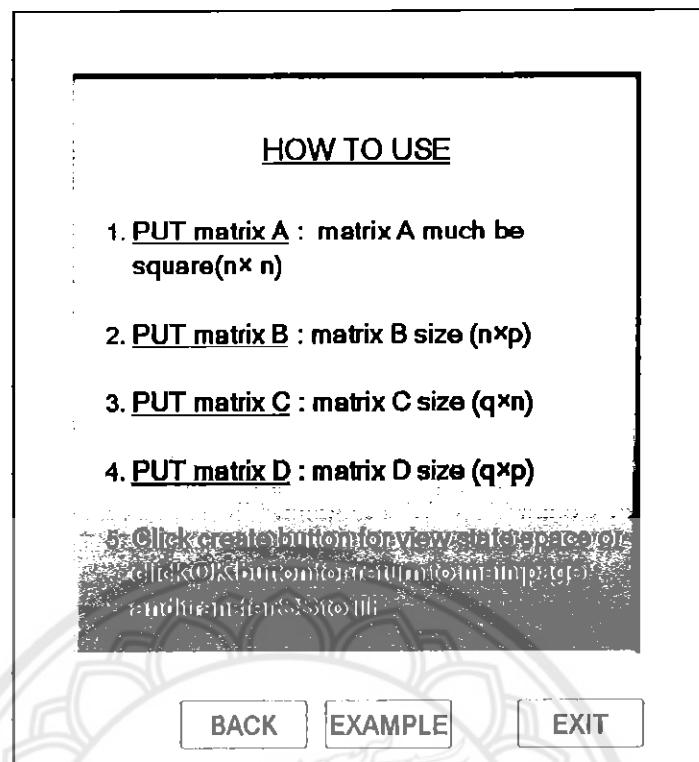
[OK](#) [EXIT](#)

รูปที่ 4.6 หน้าต่างของปริภูมิสถานะเมื่อถูกเรียกขึ้นมา

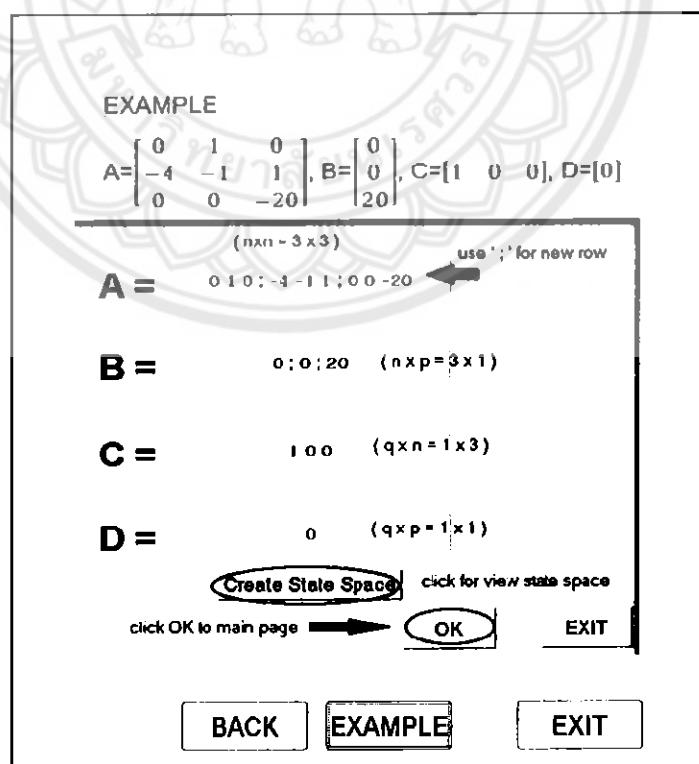


รูปที่ 4.7 หน้าต่างหลักหลังป้อนค่าปริภูมิสถานะ

- ปุ่มช่วยเหลือ (Help) เมื่อคลิกจะแสดงหน้าต่างช่วยเหลือในการใช้งานโปรแกรมให้ผู้ใช้ได้ทราบ ดังรูปที่ 4.8 และในหน้าช่วยเหลือ เมื่อผู้ใช้คลิกที่ปุ่มตัวอย่าง (Example) โปรแกรมจะมีตัวอย่างแสดงให้ผู้ใช้ได้ทราบ ดังรูปที่ 4.9



รูปที่ 4.8 หน้าต่างช่วยเหลือสำหรับการใช้งานหน้าค่าป้อนค่าปริภูมิสถานะ

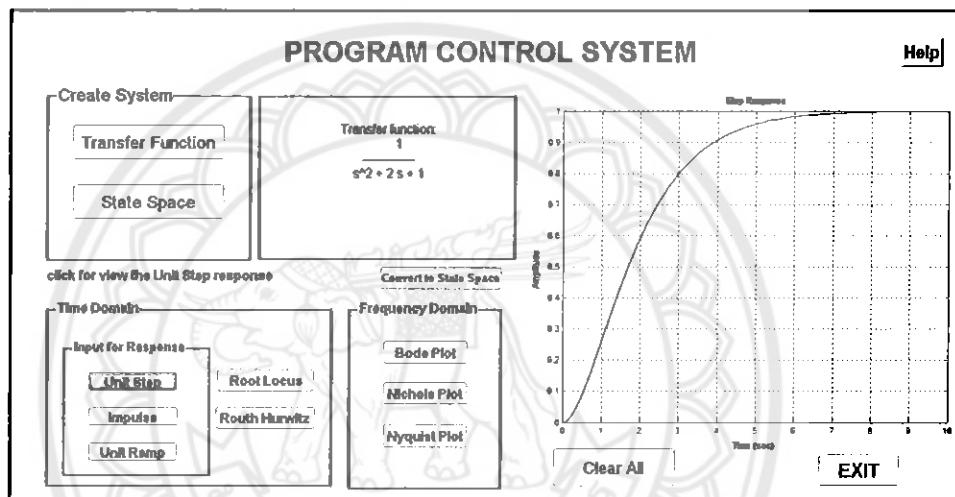


รูปที่ 4.9 หน้าต่างตัวอย่างการป้อนค่าปริภูมิสถานะ

4.4 การทำงานของหน้าต่างหลัก

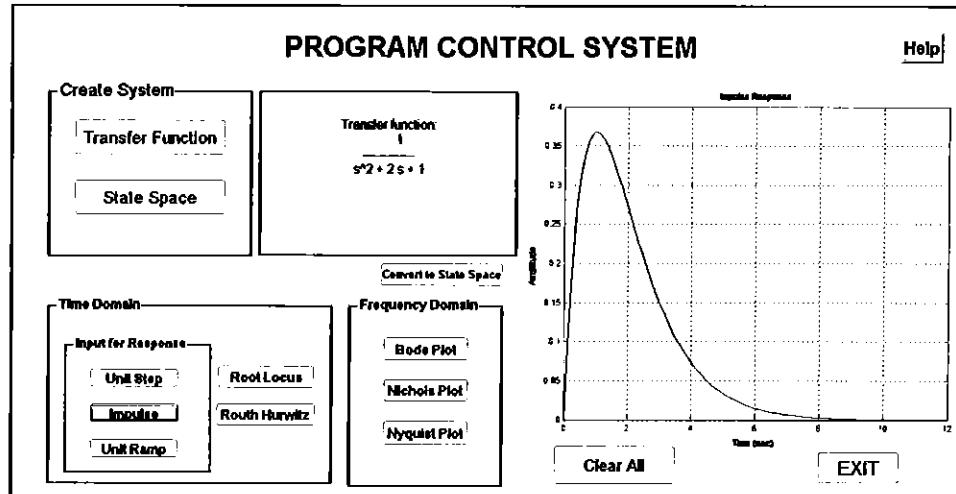
เมื่อหน้าต่างหลักได้รับค่ามาจากหน้าต่างป้อนฟังก์ชันถ่ายโอนแล้วจะมีการแสดงผลดังรูปที่ 4.3 และรูปที่ 4.7 เมื่อมีการกดปุ่มต่างๆ ในหน้าต่างหลัก จะได้ผลดังนี้

1) เมื่อปุ่มผลตอบสนองต่อสัญญาณระดับ (Unit Step) ถูกกดกราฟผลตอบสนองของสัญญาณระดับของฟังก์ชันถ่ายโอนหรือปริภูมิสถานะจะถูกแสดงที่จอแสดงผลกราฟิกทางขวาของโปรแกรม ดังรูปที่ 4.10



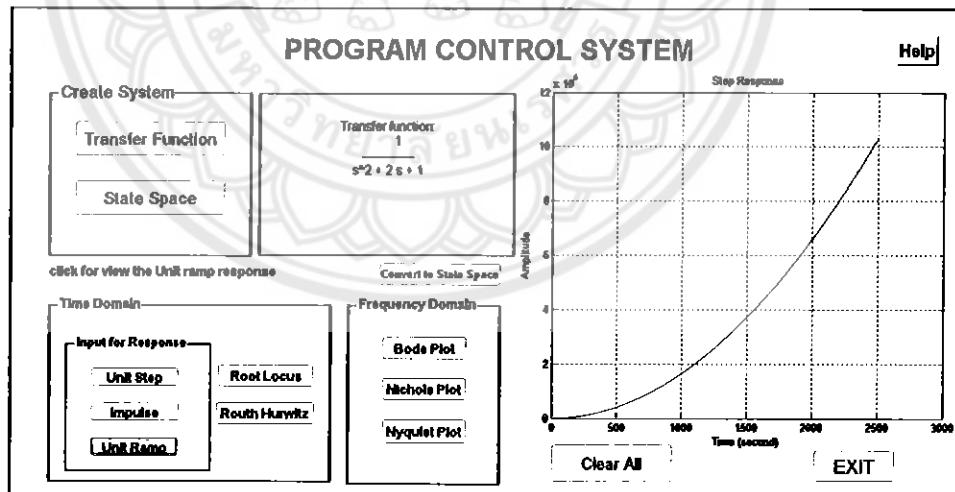
รูปที่ 4.10 โปรแกรมแสดงกราฟของผลตอบสนองเมื่อปุ่ม Unit Step ถูกกด

2) เมื่อปุ่มผลตอบสนองต่อสัญญาณอินพัลส์ (Impulse) ถูกกด กราฟผลตอบสนองของสัญญาณอินพัลส์ของฟังก์ชันถ่ายโอนหรือปริภูมิสถานะจะถูกแสดงที่จอแสดงผลกราฟิกทางขวาของโปรแกรม ดังรูปที่ 4.11



รูปที่ 4.11 โปรแกรมแสดงกราฟของผลตอบสนองเมื่อปั๊ม Impulse ถูกกด

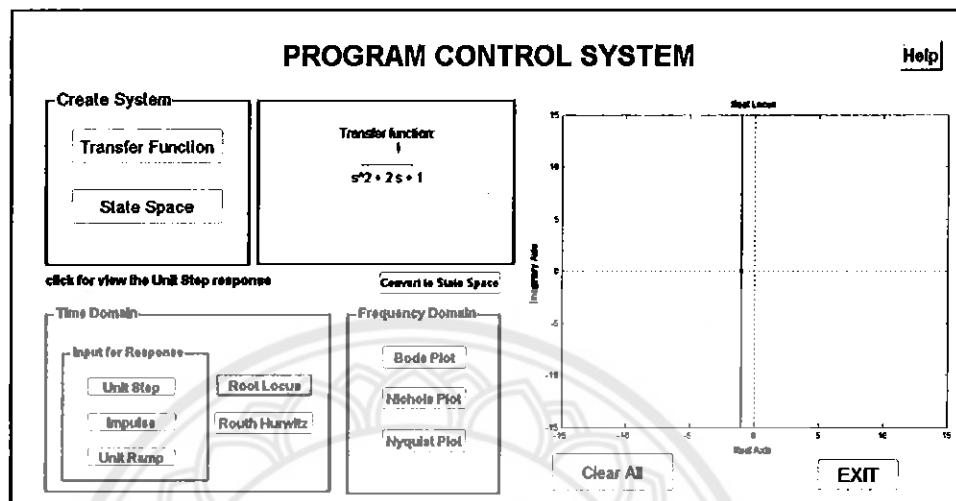
3) เมื่อปั๊มผลตอบสนองต่อสัญญาณลาด (Unit Ramp) ถูกกด กราฟผลตอบสนองของสัญญาณลาดของฟังก์ชันถ่ายโอนหรือปริภูมิสถานะจะถูกแสดงที่จอกล่องเพื่อแสดงผลกราฟทางขวาของโปรแกรม ดังรูปที่ 4.12



รูปที่ 4.12 โปรแกรมแสดงกราฟของผลตอบสนองเมื่อปั๊ม Unit Ramp ถูกกด

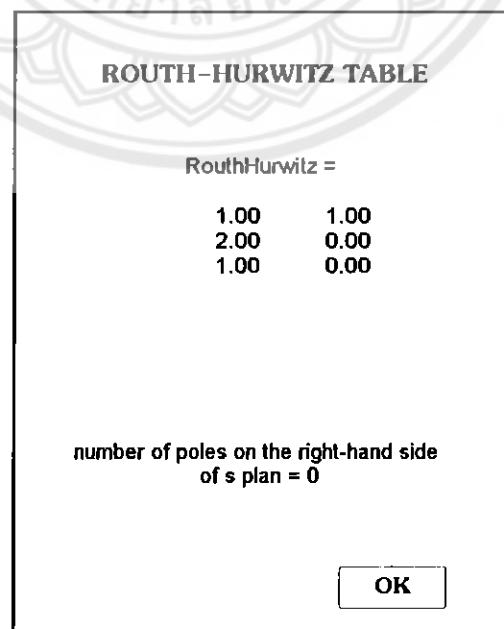
4) เมื่อปั๊มแสดงเส้นทางเดินของราก (Root locus) ถูกกด กราฟแสดงเส้นทางเดินของรากของฟังก์ชันถ่ายโอนหรือปริภูมิสถานะ จะถูกแสดงที่จอกล่องเพื่อแสดงผลกราฟทางขวาของโปรแกรม ดังรูปที่ 4.13 โดยสามารถขยายกราฟเข้า (zoom in) เพื่อให้สามารถมองเห็นได้

ชัดเจนมากยิ่งขึ้นในกรณีที่ระบบมีจำนวนวนมาก โดยเส้นทางเดินของรากจะมีสีที่แตกต่างกันทำให้พิจารณาได้ง่าย



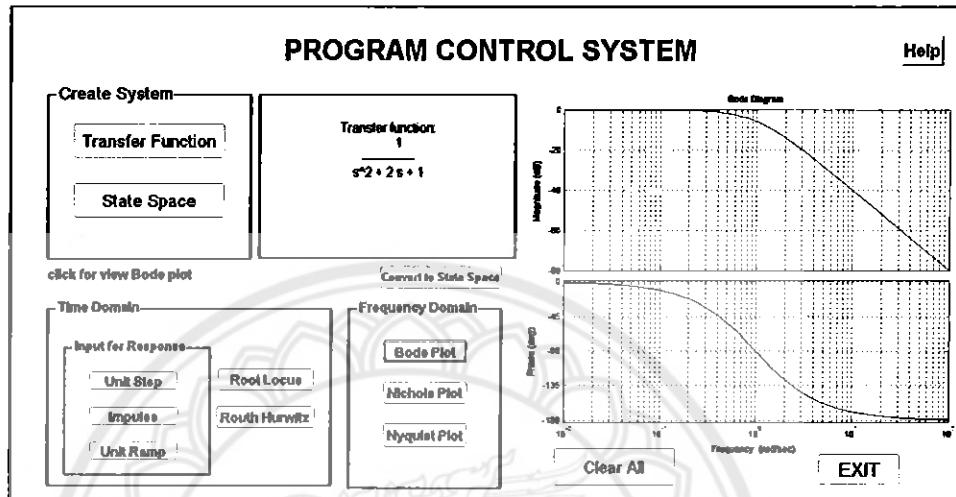
รูปที่ 4.13 โปรแกรมแสดงเส้นทางเดินของรากเมื่อปุ่ม Root locus ถูกกด

5) เมื่อปุ่มเร้าท์เฮอร์วิช (Routh Hurwitz) ถูกกด ตารางแสดงการหาค่าเสถียรภาพของเร้าท์เฮอร์วิช จะถูกแสดงในหน้าต่างที่ 4 ของโปรแกรม ดังรูปที่ 4.14 โดยด้านล่างของหน้าต่างจะมี การแสดงจำนวนโพลทางขวาของระบบแอลซ ด้วย ปุ่มตกลง (OK) สำหรับออกจากหน้าต่างนี้



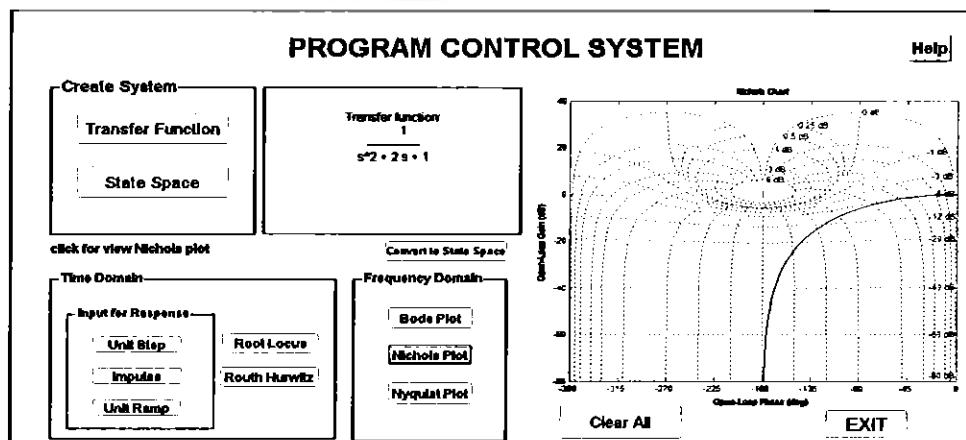
รูปที่ 4.14 โปรแกรมแสดงตารางของเร้าท์เฮอร์วิชและจำนวนโพล เมื่อปุ่ม Routh Hurwitz ถูกกด

6) เมื่อปุ่มผลตอบสนองทางความถี่ของโบเด (Bode) ถูกกด กราฟผลตอบสนองทางความถี่ที่แสดงโดยทฤษฎีโนดีของฟังก์ชันถ่ายโอน หรือปริภูมิสถานะ จะถูกแสดงที่จอแสดงผล กราฟพิกทั้งสองของโปรแกรม ดังรูปที่ 4.15 และยังสามารถขยายรูปกราฟได้



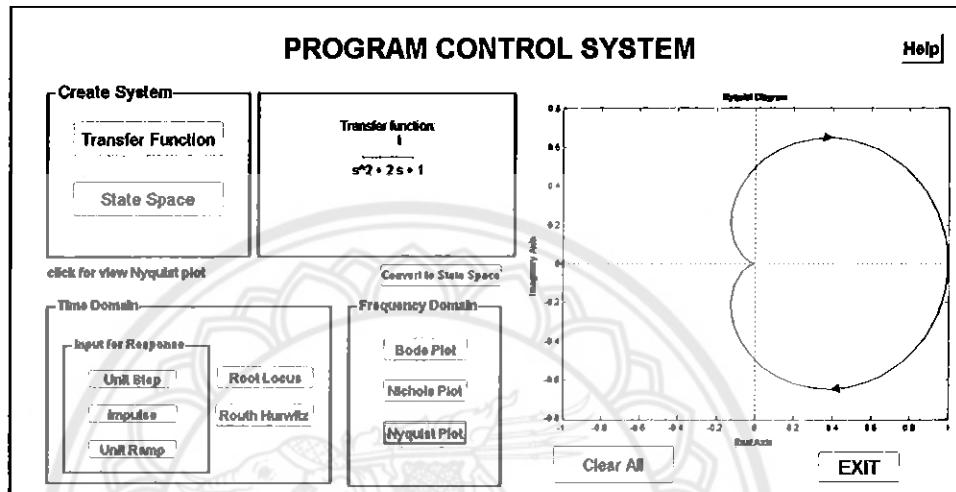
รูปที่ 4.15 โปรแกรมแสดงผลตอบสนองทางความถี่ของโนดีเมื่อปุ่ม Bode ถูกกด

7) เมื่อปุ่มผลตอบสนองทางความถี่ของนิโคลส์ (Nichols) ถูกกด กราฟผลตอบสนองทางความถี่ที่วัดโดยทฤษฎีนิโคลส์ของฟังก์ชันถ่ายโอน หรือปริภูมิสถานะ จะถูกแสดงที่จอแสดงผลกราฟพิกทั้งสองของโปรแกรม ดังรูปที่ 4.16 โดยสามารถขยายรูปกราฟเข้าไปได้ทำให้คุ้มครองข้อมูลนี้



รูปที่ 4.16 โปรแกรมแสดงผลตอบสนองทางความถี่ของนิโคลส์เมื่อปุ่ม Nichols ถูกกด

8) เมื่อปุ่มผลตอบสนองทางความถี่ของไนคิสท์ (Nyquist) ถูกกด กราฟผลตอบสนองทางความถี่ที่วัดโดยทฤษฎีในไนคิสท์ของฟังก์ชันถ่ายโอน หรือปริภูมิสถานะ จะถูกแสดงที่จอแสดงผลกราฟทางขวาของโปรแกรมดังรูปที่ 4.17 โดยสามารถขยายรูปกราฟเข้าไปได้ทำให้คุกราฟชัดเจนยิ่งขึ้น



รูปที่ 4.17 โปรแกรมแสดงผลตอบสนองทางความถี่ของไนคิสท์เมื่อปุ่ม Nyquist ถูกกด

9) เมื่อปุ่มแปลงฟังก์ชันถ่ายโอนเป็นปริภูมิสถานะ (Convert to State Space) ถูกกด หน้าต่างที่ 6 และ 7 ของโปรแกรมจะถูกเรียกขึ้นมาเพื่อแสดงผลการแปลงจากฟังก์ชันถ่ายโอนเป็นปริภูมิสถานะดังรูปที่ 4.18 โดยบอกแยกเป็นเมตริก A, B, C, และ D

```
a =
x1 x2
x1 -2 -1
x2  1  0

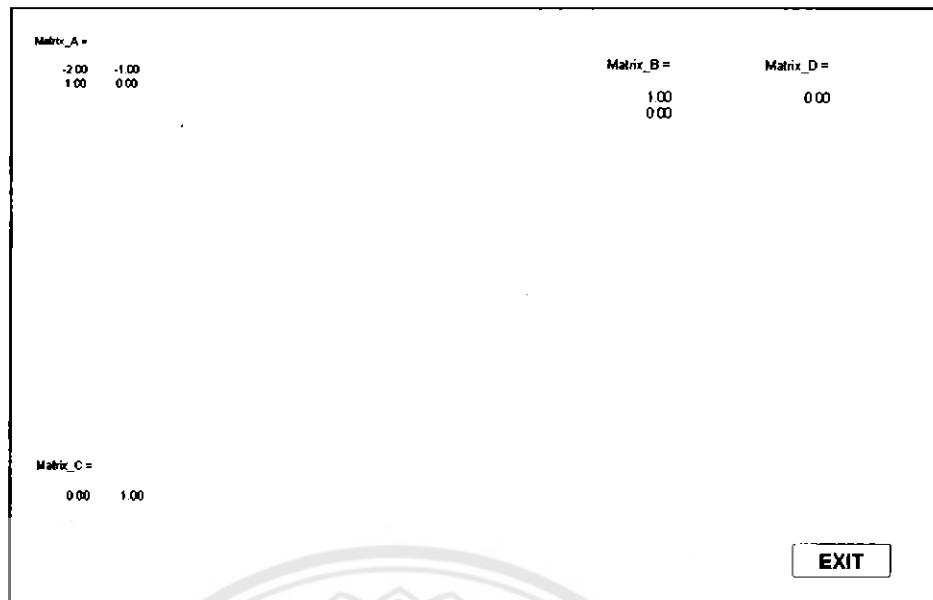
b =
u1
x1  1
x2  0

c =
x1 x2
y1  0  1

d =
u1
y1  0

Continuous-time model.
```

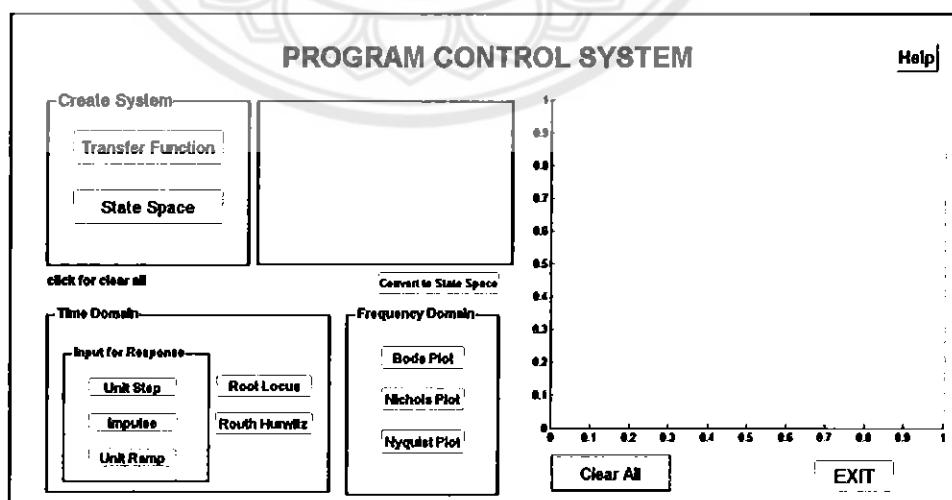
(ก)



(ว)

รูปที่ 4.18 โปรแกรมแสดงผลการแปลงฟังก์ชันถ่ายโอนเป็นปริภูมิสถานะ

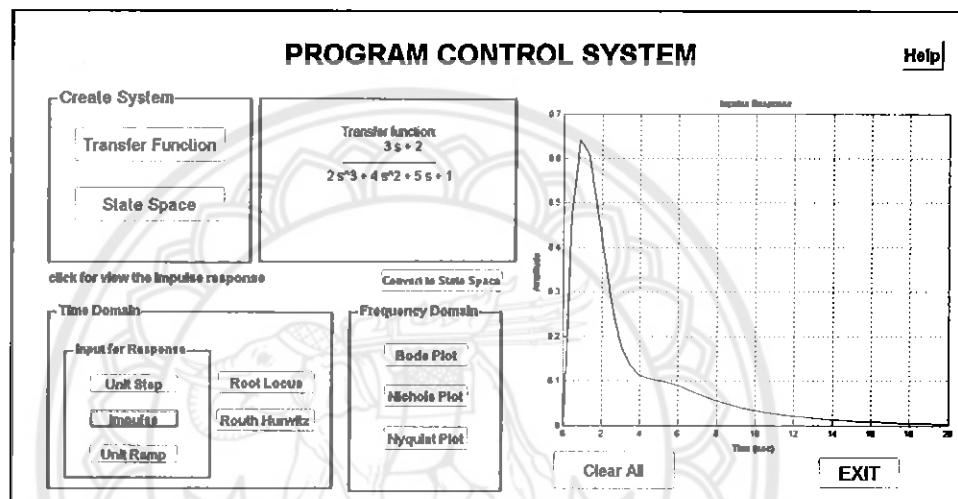
10) เมื่อปุ่มล้างข้อมูลทุกอย่าง (Clear all) ถูกกด ข้อมูลที่เกี่ยวกับการคำนวณทุกอย่าง กราฟแสดงผลต่างๆจะถูกลบทั้งหมด ทำให้หน้าต่างโปรแกรมว่างเปล่าดังเดิมเพื่อรับข้อมูลใหม่ ดังรูปที่ 4.19



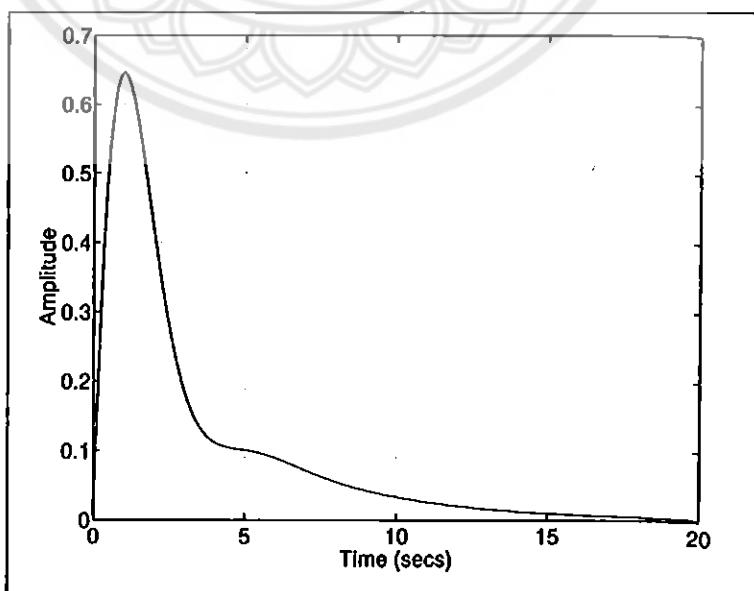
รูปที่ 4.19 โปรแกรมหลังจากกดคู่ปุ่ม Clear all

4.5 การทดสอบความถูกต้องของโปรแกรม

ทดลองใส่ค่าฟังก์ชันดัชนีโอนเท่ากับ $G(s) = \frac{3s+2}{2s^3 + 4s^2 + 5s + 1}$ ในโปรแกรมที่เขียนขึ้น จะได้ผลตอบสนองต่อสัญญาณอินพลัส ดังภาพที่ 4.20 ซึ่งเมื่อเปรียบเทียบกับกราฟที่ใช้ในการเปรียบเทียบจากหนังสือ Feedback Control Problems ของ Frederick/Chow ดังรูปที่ 4.21 ผลปรากฏว่าลักษณะกราฟที่ได้ออกมาเหมือนกัน

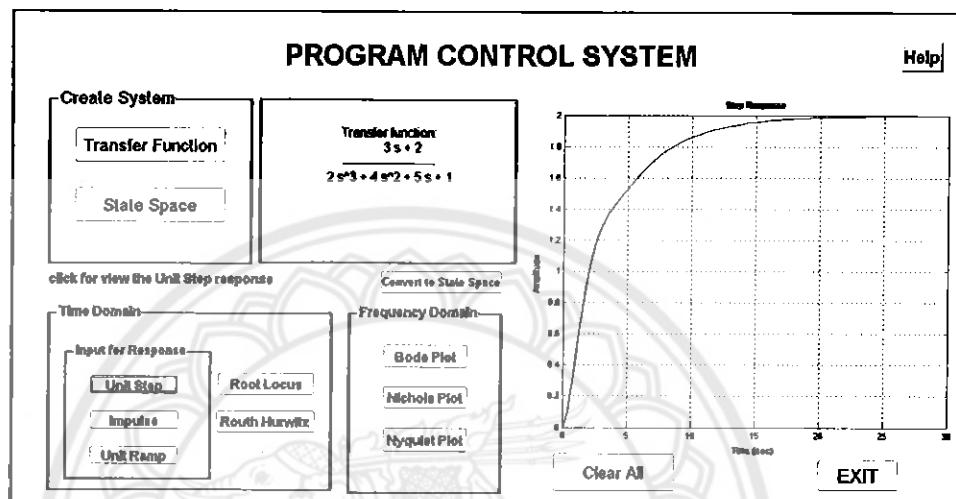


รูปที่ 4.20 กราฟผลตอบสนองของสัญญาณอินพลัสที่ได้จากโปรแกรม

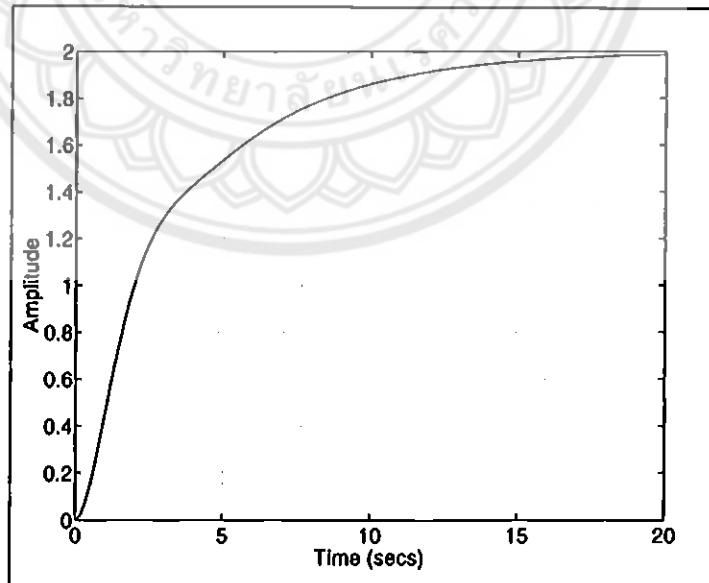


รูปที่ 4.21 ตัวอย่างสำหรับเปรียบเทียบผลตอบสนองของสัญญาณอินพลัส

คำฟังก์ชันถ่ายโอนเดิม พิจารณาผลตอบสนองต่อสัญญาณระดับ ของโปรแกรม ดังรูปที่ 4.22 เปรียบเทียบกับกราฟที่ใช้ในการเปรียบเทียบจากหนังสือ Feedback Control Problems ของ Frederick/Chow ดังรูปที่ 4.23 ผลปรากฏว่าลักษณะกราฟที่ได้ออกมาเหมือนกัน แสดงว่า โปรแกรมสามารถทำงานได้ถูกต้อง

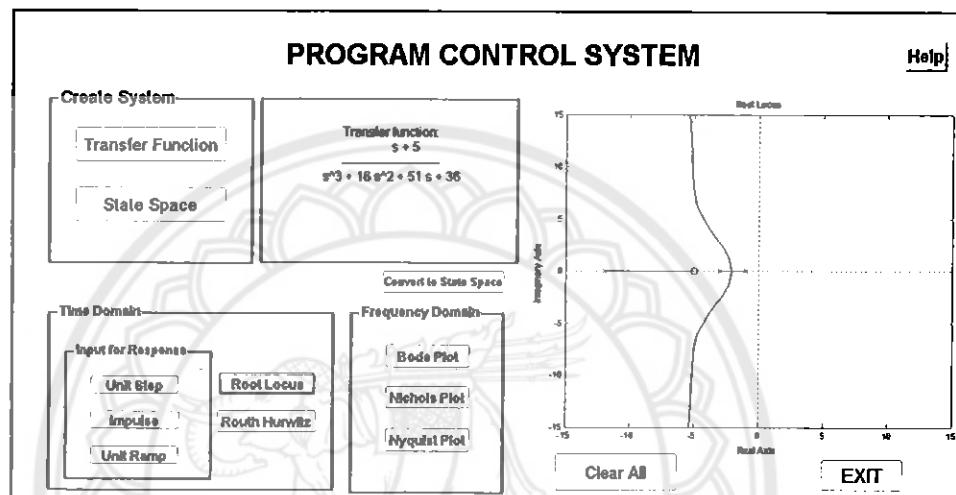


รูปที่ 4.22 กราฟผลตอบสนองของสัญญาณระดับที่ได้จากโปรแกรม

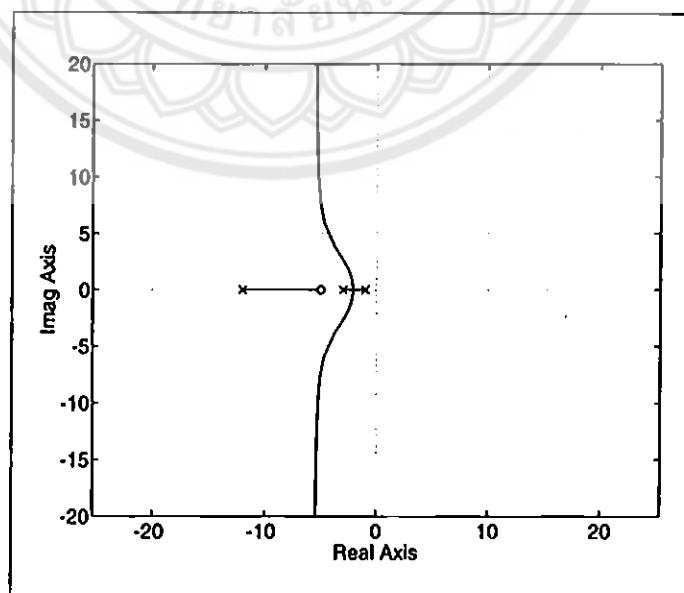


รูปที่ 4.23 ตัวอย่างสำหรับเปรียบเทียบผลตอบสนองของสัญญาณระดับ

การทดสอบโปรแกรมในส่วนเส้นทางเดินของราก โดยทดลองใส่ฟังก์ชันถ่ายโอนเข้าไปในโปรแกรมเป็น $G(s) = \frac{s+5}{s^3 + 16s^2 + 51s + 36}$ โปรแกรมจะแสดงเส้นทางเดินของรากของฟังก์ชันถ่ายโอนดังรูปที่ 4.24 เมื่อเปรียบเทียบกับเส้นทางเดินของรากที่นำมาเปรียบเทียบจากหนังสือ Feedback Control Problems ของ Frederick/Chow ดังรูปที่ 4.25 ผลปรากฏว่าโปรแกรมสามารถแสดงเส้นทางเดินของรากของฟังก์ชันถ่ายโอนอุปกรณ์ได้ถูกต้อง



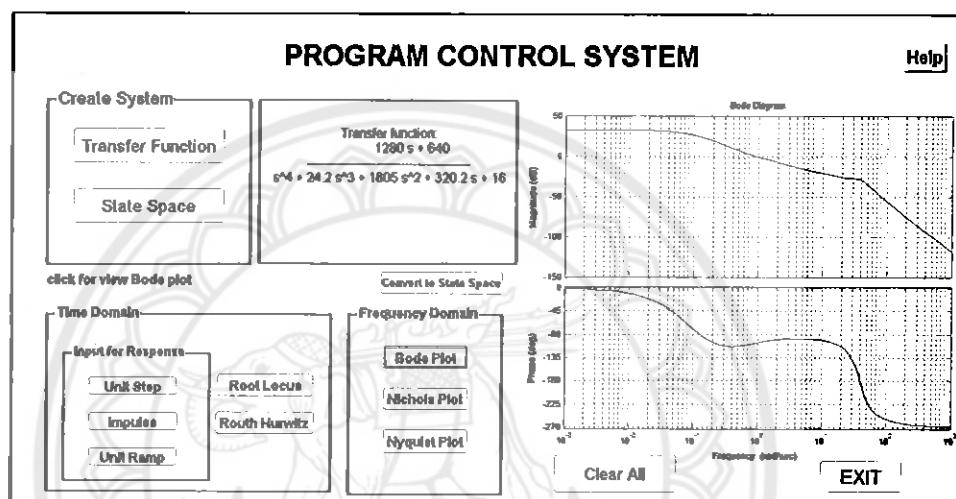
รูปที่ 4.24 กราฟแสดงผลเส้นทางเดินของรากที่ได้จากโปรแกรม



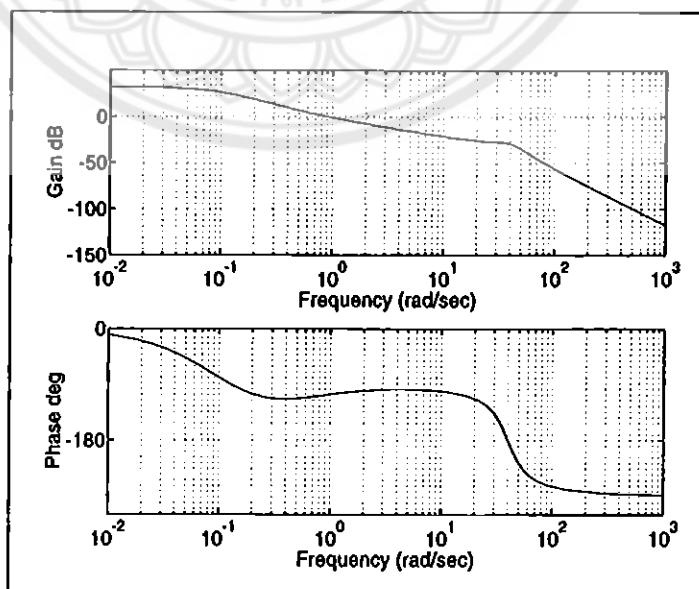
รูปที่ 4.25 ตัวอย่างสำหรับเปรียบเทียบเส้นทางเดินของรากของฟังก์ชันถ่ายโอน

การทดสอบโปรแกรมในส่วนของผลตอบสนองทางความถี่ จะใช้ฟังก์ชันถ่ายโอนเท่ากับ $G(s) = \frac{1280s + 640}{s^4 + 24.2s^3 + 1604.81s^2 + 320.24s + 16}$ ในการทดสอบ โดยผลตอบสนองทางความถี่เปรียบเทียบจากหนังสือ Feedback Control Problems ของ Frederick/Chow

พิจารณาผลตอบสนองทางความถี่จากฟังก์ชันถ่ายโอนค่าวิธีโนเบที่ได้จากโปรแกรมดังรูปที่ 4.26 จะพบว่ามีลักษณะเหมือนกับกราฟตัวอย่างเบรียบเทียบดังรูปที่ 4.27

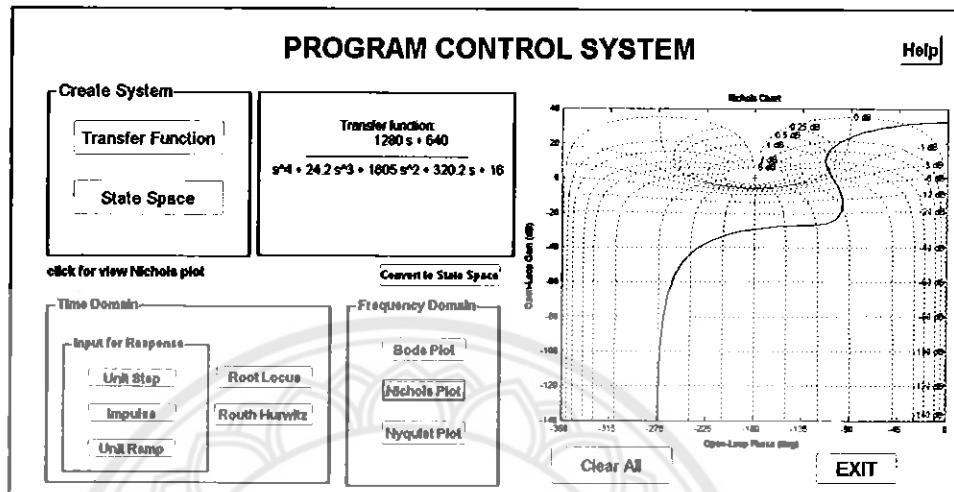


รูปที่ 4.26 กราฟแสดงผลตอบสนองทางความถี่ค่าวิธีโนเบทของฟังก์ชันถ่ายโอน

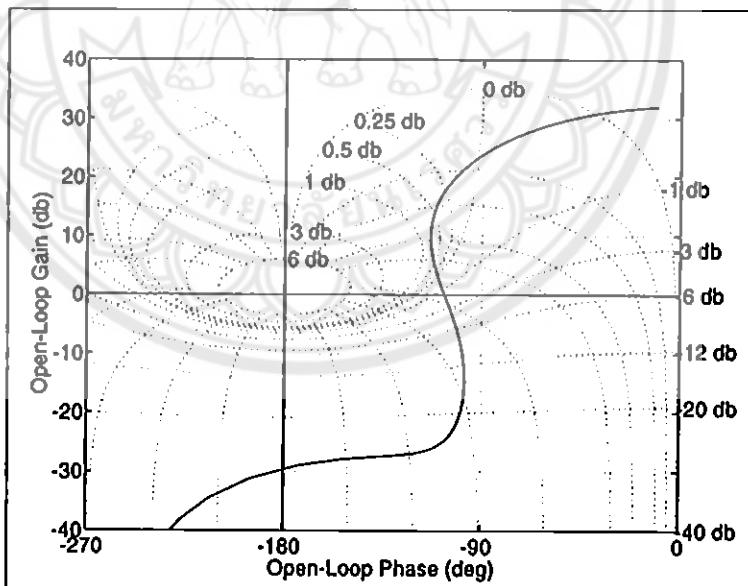


รูปที่ 4.27 ตัวอย่างเบรียบเทียบผลตอบสนองทางความถี่ของฟังก์ชันถ่ายโอนค่าวิธีโนเบ

พิจารณาผลตอบสนองทางความถี่จากฟังก์ชันถ่ายโอนด้วยวิธีนิโคลส์ที่ได้จากโปรแกรมคั่งรูปที่ 4.28 จะพบว่ามีลักษณะเหมือนกับกราฟตัวอย่างเบรีบเนที่บินคั่งรูปที่ 4.29

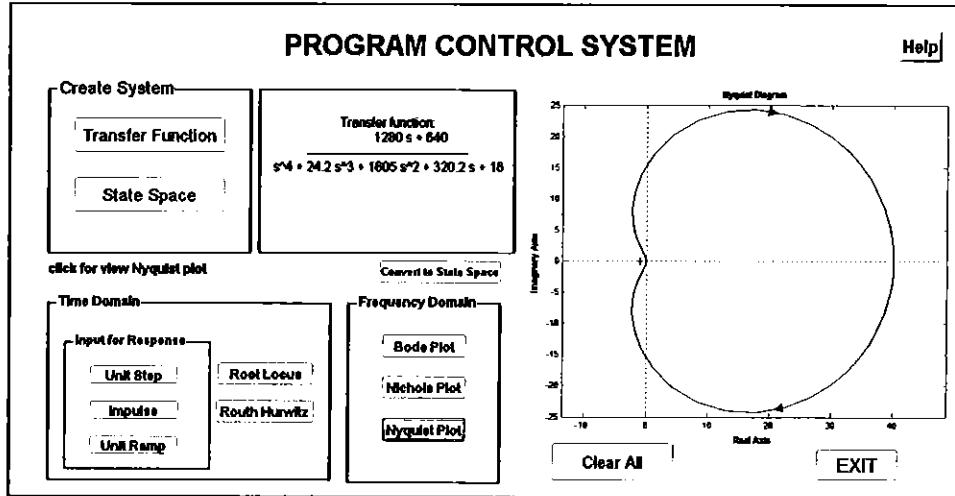


รูปที่ 4.28 กราฟแสดงผลตอบสนองทางความถี่ด้วยวิธีนิโคลส์ของฟังก์ชันถ่ายโอน

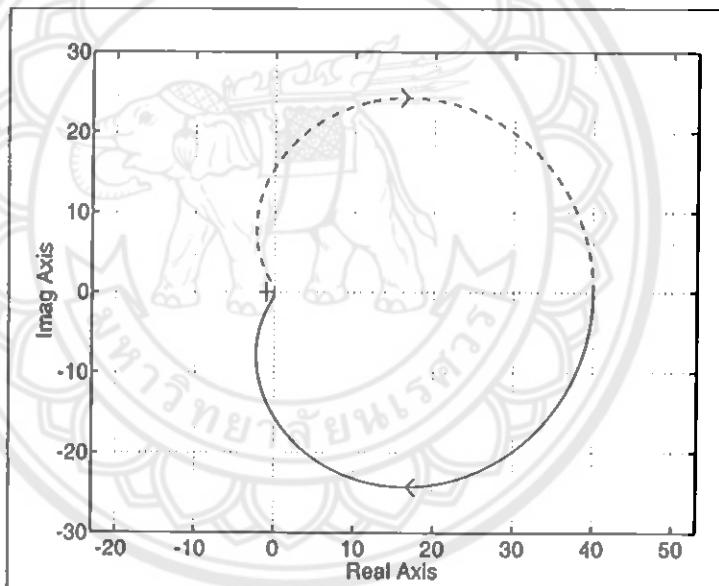


รูปที่ 4.29 ตัวอย่างเบรีบเนที่บินที่แสดงผลตอบสนองทางความถี่ของฟังก์ชันถ่ายโอนด้วยวิธีนิโคลส์

พิจารณาผลตอบสนองทางความถี่จากฟังก์ชันถ่ายโอนด้วยวิธีในคิวิสท์ที่ได้จากโปรแกรมคั่งรูปที่ 4.30 จะพบว่ามีลักษณะเหมือนกับกราฟตัวอย่างเบรีบเนที่บินคั่งรูปที่ 4.31



รูปที่ 4.30 กราฟแสดงผลตอบสนองทางความถี่ด้วยวิธีในคณิตศาสตร์ของฟังก์ชันค่าข้างใน

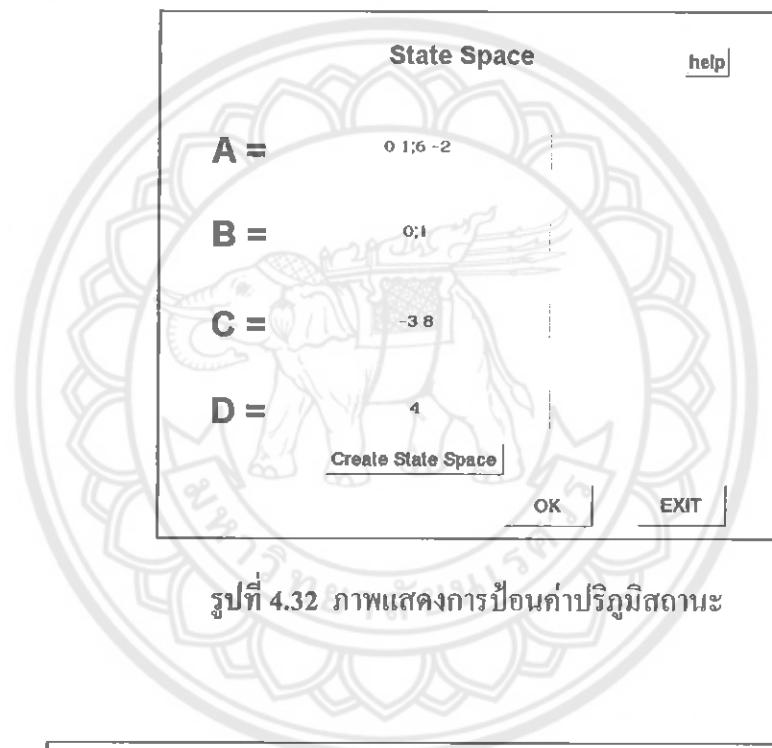


รูปที่ 4.31 ตัวอย่างเปรียบเทียบผลตอบสนองทางความถี่ของฟังก์ชันค่าข้างในด้วยวิธีในคณิตศาสตร์

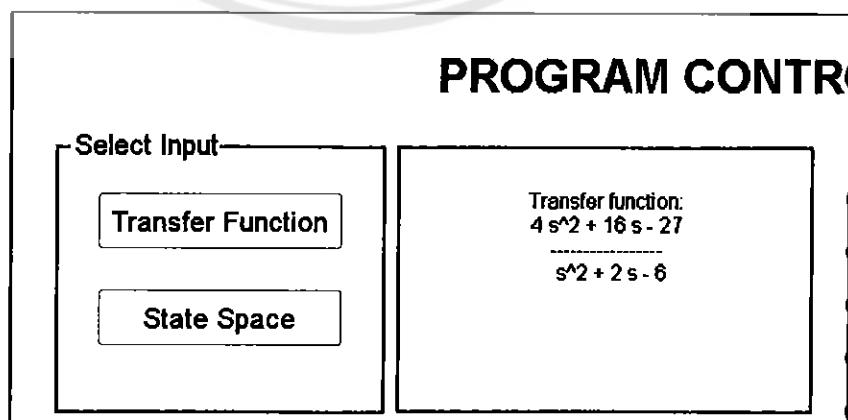
จากการเปรียบเทียบที่ผ่านมาจะพบว่า โปรแกรมที่ได้จัดทำขึ้นมีความสามารถในการหาค่าผลตอบสนองทางความถี่ ด้วยวิธีของ โนเบนิโคลส์ หรือ ในคณิตศาสตร์ ได้อย่างถูกต้อง

โดยโปรแกรมที่ได้จัดทำนี้สามารถรองรับระบบควบคุมที่มีอันดับไม่เกิน 10 เนื่องจากปัญหาในการแสดงผลเมมทริกต่างๆ ของหน้าต่าง โปรแกรม ถ้าผู้ใช้ใส่อันดับเกิน 10 โปรแกรมจะมีการเตือนผู้ใช้ให้ทราบโดยทันที

สำหรับการแปลงปริภูมิสถานะเป็นฟังก์ชันถ่ายโอน โดยทั่วไปให้ค่าเมตริกต่างๆดังนี้
 $A = \begin{bmatrix} 0 & 1 \\ 6 & -2 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $C = \begin{bmatrix} -3 & 8 \end{bmatrix}$ และ $D = 4$ ดังรูปที่ 4.32 เมื่อคำนวณตามตกลงแล้วจะได้
ค่าฟังก์ชันถ่ายโอน เป็น $G(s) = \frac{4s^2 + 16s - 27}{s^2 + 2s - 6}$ ไปแสดงในหน้าต่างหลักดังรูปที่ 4.33 เมื่อ
เปรียบเทียบกับตัวอย่างการแปลงปริภูมิสถานะเป็นฟังก์ชันถ่ายโอนดังรูปที่ 4.34 จะพบว่าการ
แปลงปริภูมิสถานะเป็นฟังก์ชันถ่ายโอนของโปรแกรมที่ได้เขียนขึ้นมีความถูกต้อง โดยสามารถใส่
เมตริกสถานะสูงสุดขนาดไม่เกิน 5×5



รูปที่ 4.32 ภาพแสดงการป้อนค่าปริภูมิสถานะ



รูปที่ 4.33 ภาพแสดงหน้าต่างโปรแกรมหน้าหลักการแปลงปริภูมิสถานะไปเป็นฟังก์ชันถ่ายโอน

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 6 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = [-3 \quad 8] \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + [4]u(t)$$

$$A = \begin{bmatrix} 0 & 1 \\ 6 & -2 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = [-3 \quad 8], d = 4$$

$$\text{Transfer Function} = C(SI - A)^{-1}B + d = \frac{4S^2 + 16S - 27}{S^2 + 2S - 6}$$

รูปที่ 4.34 ภาพแสดงการแปลงปริภูมิสถานะไปเป็นฟังก์ชันถ่ายโอน

ที่มา: site.iugaza.edu.ps/taisha/files/2010/03/Lab3.pdf

จากผลการทดลองที่ได้กล่าวมา จากการทดลองการใช้งานโปรแกรม และการทดสอบความถูกต้อง พบร่วมกับโปรแกรมสามารถทำงานได้ดีและมีความถูกต้อง โดยสามารถใส่ค่าฟังก์ชันถ่ายโอนได้อันดับไม่เกิน 10 เนื่องจากข้อจำกัดของโปรแกรมและการแสดงผลเมตริกนี้ของการส่งค่าผ่านหลายหน้าค้างทำให้ค่าอนhangยก ในส่วนของปริภูมิสถานะจะใส่ค่าเมตริกสถานะ (A) ได้สูงสุดขนาดไม่เกิน 5×5 โปรแกรมสามารถแปลงค่าปริภูมิสถานะเป็นฟังก์ชันถ่ายโอนเพื่อมาใช้ในการคำนวณในหน้าต่างหลัก ได้อย่างถูกต้อง

บทที่ 5

สรุปผลการดำเนินโครงการ

5.1 ผลการดำเนินโครงการ

โครงการนี้เป็นการเขียนโปรแกรมเพื่อใช้สำหรับระบบควบคุมเบื้องต้น เขียนโปรแกรมโดยใช้ส่วนต่อประสานกราฟิกกับผู้ใช้งานโปรแกรมแม่ทัพเดิม (MATLAB) ซึ่งเป็นการหาผลตอบสนองของระบบควบคุม โดยการป้อนฟังก์ชันถ่ายโอนหรือปริภูมิสถานะ เข้าไปที่โปรแกรม โดยโปรแกรมจะแสดงผลตอบสนองของระบบทั้งในทางเวลา และทางความถี่ โดยโปรแกรมสามารถแสดงผลตอบสนองของระบบที่ระบบควบคุมไม่เกินอันดับ 10 เนื่องจากข้อจำกัดในการแสดงผล โดยผลตอบสนองของระบบจะแบ่งได้ ดังนี้

1) การวิเคราะห์ผลตอบสนองในทางเวลา โดยใช้

- ผลตอบสนองของระบบต่อสัญญาณระดับ
- ผลตอบสนองของระบบต่อสัญญาณอินพุตส์
- ผลตอบสนองของระบบต่อสัญญาณล่าด้วย
- วิเคราะห์เส้นทางเดินของรถ
- การวิเคราะห์โดยใช้ทฤษฎีเร้าท์ไฮร์วิช

2) การวิเคราะห์ผลตอบสนองในทางความถี่ โดยใช้

- การวิเคราะห์ผลตอบสนองโดยใช้ทฤษฎีโนมี
- การวิเคราะห์ผลตอบสนองโดยใช้ทฤษฎีนิโกลส์
- การวิเคราะห์ผลตอบสนองโดยใช้ทฤษฎีไนกวิสท์

โปรแกรมที่เขียนขึ้นจากโครงการนี้ สามารถที่จะคำนวณ และสามารถแสดงกราฟที่เกี่ยวข้องกับการวิเคราะห์ผลตอบสนองของระบบได้ถูกต้อง โปรแกรมสามารถทำงานได้ดีและถูกต้อง ทั้งผลตอบสนองของระบบในทางเวลาและผลตอบสนองของระบบในทางความถี่ เพื่อให้ผู้ใช้ประกอบการศึกษาระบบควบคุมนี้ ได้ตรวจสอบความถูกต้องของคำตอบ จากการคำนวณที่

ทำให้ได้จากการคำนวณด้วยตนเอง ทำให้ผู้ใช้ประกอบการศึกษาระบบควบคุมสามารถทราบได้ว่า คำตอบจาก การคำนวณถูกต้องหรือผิดพลาดอย่างไร และจะได้แก้ไขให้ถูกต้องต่อไป

5.2 ปัญหาที่พบขณะปฏิบัติโครงการ

- 1) การแสดงผลในส่วนของเมตริก และปริภูมิสถานะ โปรแกรมแม่ทัพเลื่อนมีการแสดงผล ที่ผู้ใช้ดูยาก ทำให้ใช้เวลานานในการจัดการกับการแสดงผล
- 2) การส่งค่าผ่านหลายหน้าต่างทำได้ยาก ทำให้ใช้เวลา กับการทดลองหาวิธีส่งและรับค่า ของแต่ละหน้าต่าง
- 3) การทดสอบความถูกต้อง ในส่วนของของ โปรแกรม ในการระบบที่มีอันดับสูงๆ มาทดสอบความถูกต้องของ โปรแกรมนี้ หายาก

5.3 ข้อเสนอแนะ

- 1) ถ้าผู้สนใจจะนำ โปรแกรม ไปประกอบในการศึกษาระบบควบคุม ควรจะศึกษาวิธีการ ป้อนข้อมูลของ โปรแกรม แม่ทัพเลื่อน ก่อน เพื่อให้ง่ายในการใช้งาน
- 2) ถ้าผู้สนใจจะนำ โปรแกรม ไปพัฒนาต่อให้มีฟังก์ชันมากขึ้น ดีขึ้น ควรทำให้ผู้ใช้ สามารถใช้งานได้ง่ายขึ้น
- 3) ในการพัฒนา โปรแกรม ด้วย โปรแกรม แม่ทัพเลื่อน สามารถใช้ความช่วยเหลือ จากคัว ช่วงของ โปรแกรม (help) หรือ <http://www.mathworks.com/products/matlab/> เพื่อใช้ในการหา ข้อมูลมาประกอบการพัฒนา

เอกสารอ้างอิง

- [1] Dean K. Frederick and Joe H. Chow.(1995).**Feedback Control Problems Using MATLAB and The Control System Toolbox.** Boston: PWS Publishing Company.
- [2] ดร.ปริญญา สงวนสัตย์.(2553).**คู่มือ MATLAB ฉบับสมบูรณ์.**นนทบุรี:บริษัท ไอคิวซี พ्रีเมียร์ จำกัด
- [3] สุวัตถิ์ กลั่นความคื้.(2539).**ระบบควบคุม.กรุงเทพมหานคร:สำนักพิมพ์จุฬาลงกรณ์ มหาวิทยาลัย**
- [4] John Dorsey.(2002). **Continuous and discrete control systems.** Boston : McGraw-Hill
- [5] The MathWorks, Inc. (2010). **MATLAB-Documentation** Retrieved February 25, 2011, from <http://www.mathworks.com/help/techdoc>

ประวัติผู้ดำเนินโครงการ



ชื่อ นายนเรศ ประทุม
ภูมิลำเนา 81/1 หมู่ 2 ต.สารจิตร อ.ศรีสัชนาลัย จ.สุโขทัย
ประวัติการศึกษา
– จบมัธยมศึกษาจาก โรงเรียนสารรค์อนันต์วิทยา
– ปัจจุบันกำลังศึกษาระดับปริญญาตรีชั้นปีที่ 4
สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์
มหาวิทยาลัยนเรศวร

E-mail: pktech22k@gmail.com

