

ระบบควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง

DC. MOTOR SPEED CONTROL SYSTEM



โดย

นายวิรัตน์ ทองจันทร์แก้ว

นายวิโรจน์ สระมุณี

นายพงษ์ศักดิ์ สุนทร

เลขหมู่.....
เลขทะเบียน..... 42629
วัน, เดือน, ปี- 5 ส.ย. 2545

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า 4
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง

DC. MOTOR SPEED CONTROL SYSTEM

โดย

นายวิรัตน์ ทองจันทร์แก้ว รหัสประจำตัว 41013231

นายวิโรจน์ สระมุณี รหัสประจำตัว 41013232

นายพงษ์ศักดิ์ สุนทร รหัสประจำตัว 41013256

อาจารย์ที่ปรึกษา

ผศ.พิชัย คุศิริวานิชกร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ ระบบควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง
DC. MOTOR SPEED CONTROL SYSTEM

ชื่อนักศึกษา นายวิรัตน์ ทองจันทร์แก้ว รหัสประจำตัว 41013231
นายวิโรจน์ ธรรมฤณี รหัสประจำตัว 41013232
นายพงษ์ศักดิ์ สุนทร รหัสประจำตัว 41013256

โครงการได้รับการตรวจสอบแล้วพร้อมที่จะทำการสอบได้



(ผศ. พิชัย กุศิริวานิชกร)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2543

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง ระบบควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง

DC. MOTOR SPEED CONTROL SYSTEM

ผู้จัดทำ

1. นายวิรัตน์ ทองจันทร์แก้ว
2. นายวิโรจน์ ธรรมฤดี
3. นายพงษ์ศักดิ์ สุนทร



..... อาจารย์ที่ปรึกษา
(ผศ.พิชัย คุณศิริวานิชกร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง
DC. MOTOR SPEED CONTROL SYSTEM

นายวิรัตน์ ทองจันทร์แก้ว
นายวิโรจน์ สระมุณี
นายพงษ์ศักดิ์ สุนทร
ผศ.พิชัย คูศิริวานิชกร (อาจารย์ที่ปรึกษา)
ปีการศึกษา 2543

บทคัดย่อ

วิทยานิพนธ์นี้เป็นการศึกษาเกี่ยวกับการออกแบบส่วนควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง โดยใช้เทคนิคของเฟสล็อกคูลและได้นำไปประยุกต์ใช้งานกับเครื่องวิ่งออกกำลังกาย ด้วยการใส่คอมพิวเตอรืในการควบคุมความเร็วของเครื่องวิ่ง พร้อมทั้งแสดงผลความเร็วในการวิ่งและระยะทางที่วิ่ง

นอกจากนี้ได้มีการเพิ่มเติม วงจรวัดอัตราการเต้นของหัวใจเข้าไป เพื่อใช้ในการวัดอัตราการเต้นของหัวใจในขณะที่วิ่ง โดยมีการแสดงผลและบันทึกผลอัตราการเต้นของหัวใจในช่วงเวลาต่างๆขณะทำการวิ่ง และสามารถเรียกดูอัตราการเต้นของหัวใจที่วิ่งในวันต่างๆในฐานข้อมูลเพื่อเปรียบเทียบกันได้

DC. MOTOR SPEED CONTROL SYSTEM

Mr. Wirat Tongchankaew

Mr. Wirot Saramunee

Mr. Pongsak Suntorn

Mr. Pichai Kusiriwanichkon (Advisor)

2000

Abstract

In this project , the speed control system of a DC. Motor using Phase Locked Loop (PLL) technique is studied and designed. The aim of its application is used for the running track.

In the system , we used computer to control the speed of the running track and display the speed and the distance in the run.

In addition , we included the heart rate measurement circuit to measure the heart rate of the runner and we provided database to collect and compare the heart rate in various time.

กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จได้ด้วยดี คณะผู้จัดทำขอขอบพระคุณ อาจารย์ที่ปรึกษา อาจารย์พิชัย
คูศิริวานิชกร เป็นอย่างสูง ที่ได้ช่วยกรุณาให้คำแนะนำให้กำลังใจและวางขอบเขตการทำงาน

ขอขอบคุณ คุณปรีชา ช่วยเต้า ศิษย์เก่าภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ ลาด
กระบัง ที่กรุณาช่วยจัดหาอุปกรณ์ตรวจจับ (sensor) ทางแสง ซึ่งใช้ในทางการแพทย์ เป็นตัวตรวจ
จับอัตราการเต้นของหัวใจและค่าอื่นๆที่เกี่ยวข้อง

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้จัดทำขอขอบแต่ผู้สนใจทุกท่าน

.....
(นายวิรัตน์ ทองจันทร์แก้ว)

.....
(นายวิโรจน์ สระมณี)

.....
(นายพงษ์ศักดิ์ สุนทร)

ผู้จัดทำ

วันที่...../...../.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและความเป็นมาของปริยญาณีพนธ์	1
1.2 ขอบเขตของปริยญาณีพนธ์	1
1.3 เนื้อหาโดยสังเขป	3
บทที่ 2 ทฤษฎีการควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูล	4
2.1 การทำงานของระบบ	4
2.2 ส่วนประกอบของระบบควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูล	4
2.2.1 Phase comparator	4
2.2.2 Low Pass Filter	5
2.2.3 Amplifier	5
2.2.4 Motor	5
2.2.5 Encoder	6
2.3 บล็อกไดอะแกรมของระบบ	6
2.4 การวิเคราะห์เสถียรภาพ โดยใช้รูลอเคลต	7
2.4.1 การพล็อตทางเดินของราก	8
2.4.2 วิธีการพล็อตทางเดินของราก	9
บทที่ 3 วงจรควบคุมความเร็วมอเตอร์	11
3.1 บล็อกการทำงานของระบบควบคุมความเร็วมอเตอร์ด้วยเทคนิคเฟสล็อกคูล	11
3.1.1 Phase comparator	11
3.1.2 Low Pass Filter	12
3.1.3 Amplifier	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
3.1.4 วงจร Chopper	13
3.1.5 Motor	13
3.1.6 Encoder	14
3.2 การออกแบบและวิเคราะห์เสถียรภาพของระบบด้วยเทคนิครูลอัส	16
3.3 แสดงวงจรควบคุมความเร็วมอเตอร์	22
3.4 การทดลองและผลการทดลอง	23
บทที่ 4 วงจรวัดอัตราการเดินของหัวใจ	25
4.1 การออกแบบ	26
4.1.1 วงจร sensor และวงจรขยายที่ 1	26
4.1.2 Low Pass Filter (LPF)	26
4.1.3 วงจรขยายที่ 2 และขมิททริกเกอร์	30
4.2 วงจรวัดอัตราการเดินของหัวใจ	30
4.3 การทดลองและผลการทดลอง	31
บทที่ 5 ภาคควบคุมและแสดงผล	32
5.1 การควบคุมและการแสดงผล	32
5.2 วงจรควบคุมโดยใช้ไมโครคอนโทรลเลอร์ตระกูล 89C2051	33
5.3 การทำงานของโปรแกรม	34
5.3.1 การทำงานของโปรแกรมวัดคาบเวลาอัตราการเดินของหัวใจ และนับพัลส์จากเซ็นเซอร์	35
5.3.2 การทำงานของโปรแกรมหลัก	38
บทที่ 6 การใช้โปรแกรมคอมพิวเตอร์ในการควบคุมและแสดงผล	40
6.1 วิธีการใช้งานโปรแกรมในการวิ่งทดสอบ	41
6.1.1 เปิดเข้าโปรแกรม	41
6.1.2 กำหนดความเร็ว และกำหนดเวลาหรือระยะทางในการวิ่ง	43
6.1.3 เริ่มวิ่ง	43
6.1.4 เมื่อจะหยุดวิ่ง	44
6.2 การดูข้อมูลการวิ่ง	45
6.2.1 การดูกราฟ	45
6.2.2 การแก้ไข หรือเพิ่มเติมข้อมูลประวัติของผู้ป่วย	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
6.2.3 การคู่มือประวัติผู้ป่วยทั้งหมดในรูปแบบของตาราง	47
บทที่ 7 สรุป ปัญหา แนวทางแก้ไขและพัฒนา	49
ภาคผนวก ก ไมโครคอนโทรลเลอร์ 89C2051 และพอร์ตอนุกรม RS232	
ภาคผนวก ข วงจร	
ภาคผนวก ค โปรแกรม Assembly	
ภาคผนวก ง รายละเอียด และคุณสมบัติของอุปกรณ์	
บรรณานุกรม	



สารบัญตาราง

ตารางที่ 4.1 Normalizer Butterworth FPF Loss Function

หน้า

27



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 1.1 บล็อกแสดงการทำงานของเครื่องวัดกำลังกาย	2
รูปที่ 2.1 บล็อกการทำงานของเฟสล็อกคูลูป	4
รูปที่ 2.2 บล็อกไดอะแกรม ระบบควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูลูป	6
รูปที่ 2.3 ตัวอย่างวิธีการพล็อตทางเดินของราก	10
รูปที่ 3.1 บล็อกการทำงานของวงจรถวลความเร็วมอเตอร์	11
รูปที่ 3.2 วงจร Low Pass Filter	12
รูปที่ 3.3 วงจรขยายปรับเกนได้	13
รูปที่ 3.4 แสดงความเร็วเชิงมุมของมอเตอร์ที่เวลาต่างๆเมื่อป้อนแรงดัน 36 V	13
รูปที่ 3.5 วงจรเซ็น โคคเคอร์แบบ optical tachometer generator	15
รูปที่ 3.6 แสดงบล็อกไดอะแกรมของระบบ	16
รูปที่ 3.7 ทางเดินของรากเมื่อ τ_2 มีค่าน้อยกว่า T_m	17
รูปที่ 3.8 แสดงเส้นทางเดินของรากกรณี τ_2 มีค่ามากกว่า T_m เล็กน้อย	18
รูปที่ 3.9 แสดงเส้นทางเดินของรากกรณี τ_2 มีค่ามากกว่า T_m มากๆ	18
รูปที่ 3.10 แสดงทางเดินของรากของระบบ	19
รูปที่ 3.11 แสดงรากของสมการและความสัมพันธ์ระหว่าง α , ξ , ω_n และ ω	20
รูปที่ 3.12 แสดง maximum over shoot ที่ค่า damping ratio (ξ) ค่าต่างๆ	21
รูปที่ 3.13 แสดงวงจรถวลความเร็วมอเตอร์	22
รูปที่ 3.14 แสดงสัญญาณเอาต์พุทของเฟสคอมพาราเตอร์ (TP1)	23
รูปที่ 3.15 รูปคลื่นแสดงสัญญาณจากเซ็น โคคเคอร์ (TP3) และสัญญาณควบคุม (TP2)	24
รูปที่ 3.16 แสดงสัญญาณเอาต์พุท (TP4) ในช่วงเริ่มต้นการทำงานของมอเตอร์	24
รูปที่ 4.1 แสดงการวางตำแหน่ง ตัวกำเนิดแสง ตัวรับแสง แบบส่งผ่านและสะท้อนกลับ	25
รูปที่ 4.2 บล็อกการทำงานของวงจรวัดการเดินของหัวใจ	25
รูปที่ 4.3 วงจร sensor และวงจขยายภาคที่ 1	26
รูปที่ 4.4 Normalized Fourth Order LPF Circuit ($\omega_c = 1 \text{ Rad./Sec.}$)	27
รูปที่ 4.5 วงจรขยายที่ 2 และซิมิทริกเกอร์	30
รูปที่ 4.6 แสดงสัญญาณเอาต์พุทของ Low Pass Filter (TP1)	31
รูปที่ 4.7 แสดงสัญญาณเอาต์พุทของวงจขยายที่ 2 (TP2)	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 5.1 แสดงการสั่งงานเพื่อควบคุมความเร็วมอเตอร์	32
รูปที่ 5.2 แสดงการวัดอัตราการเต้นของหัวใจ , ความเร็วของเครื่องวิ่ง และระยะทางที่วิ่ง	33
รูปที่ 5.3 วงจรนับ/จับเวลา 0 เมื่อทำงานในโหมด 3	34
รูปที่ 5.4 โฟลว์ชาร์ทแสดงการทำงานของโปรแกรมบริการอินเทอร์เน็ต INTO	36
รูปที่ 5.5 โฟลว์ชาร์ทแสดงการทำงานของโปรแกรมบริการอินเทอร์เน็ต INT1	36
รูปที่ 5.6 โฟลว์ชาร์ทแสดงการทำงานของโปรแกรมบริการอินเทอร์เน็ต Time TH0	37
รูปที่ 5.7 โฟลว์ชาร์ทแสดงการทำงานของโปรแกรมหลัก	39
รูปที่ 6.1 หน้าจอหลักของโปรแกรม	40
รูปที่ 6.2 หน้าจอลงทะเบียนผู้ป่วยใหม่	41
รูปที่ 6.3 ข้อความเตือนเมื่อกรอกรหัสผู้ป่วยซ้ำกับผู้ป่วยเก่า	41
รูปที่ 6.4 ข้อความแจ้งแก่ผู้ลงทะเบียนเมื่อบันทึกข้อมูลเรียบร้อยแล้ว	42
รูปที่ 6.5 หน้าจอเข้าโปรแกรม (ผู้ป่วยเก่า)	42
รูปที่ 6.6 ข้อความแจ้งกรณีกรอกรหัสไม่ถูกต้อง	42
รูปที่ 6.7 กรอบกำหนดความเร็ว และเวลาหรือระยะทาง	43
รูปที่ 6.8 กรอบแสดงผลขณะกำลังวิ่ง	44
รูปที่ 6.9 หน้าจอข้อมูลการวิ่งของผู้ป่วย	45
รูปที่ 6.10 หน้าจอแสดงกราฟอัตราการเต้นของหัวใจ	46
รูปที่ 6.11 หน้าจอเพิ่มลบ แก้ไข ค้นหาข้อมูลของผู้ป่วย	47
รูปที่ 6.12 แสดงประวัติส่วนตัวของผู้ป่วยทั้งหมด	48

บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของปรีญญาณีพนธ์

ในการที่จะเปลี่ยนพลังงานไฟฟ้ามาเป็นพลังงานกลเพื่อให้ทำงานช่วยอำนวยความสะดวกแก่เราในด้านต่าง ๆ นั้นจะต้องใช้มอเตอร์ อุปกรณ์อำนวยความสะดวกต่างๆ ที่เราสัมผัสอยู่เป็นประจำไม่ว่าจะเป็นเครื่องใช้ไฟฟ้าภายในบ้าน เช่น เครื่องซักผ้า พัดลม หรือในงานวิศวกรรม เช่น เครื่องเจาะแผ่นปรีนท์ ตลอดจนแขนกล หุ่นยนต์ หรือเครื่องจักรกลที่ทำงานอัตโนมัติในโรงงานอุตสาหกรรมต่างๆ ล้วนแล้วแต่ใช้มอเตอร์ทั้งสิ้น นั่นคือถ้าเราสามารถควบคุมการทำงานของมอเตอร์คือควบคุมความเร็วรอบ ทิศทาง ตำแหน่งของมอเตอร์ได้แล้ว เราสามารถที่จะนำมอเตอร์ไปใช้งานต่างๆ ได้มากมาย สำหรับปรีญญาณีพนธ์นี้ เป็นการออกแบบระบบควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง โดยได้นำไปประยุกต์ใช้งานในการควบคุมความเร็วของเครื่องวิ่งออกกำลังกาย ซึ่งออกแบบระบบโดยการควบคุมเครื่องวิ่งออกกำลังกายผ่านทางคอมพิวเตอร์ และได้เพิ่มวงจรวัดอัตราการเดินของหัวใจ เพื่อวัดอัตราการเดินของหัวใจขณะวิ่งด้วย

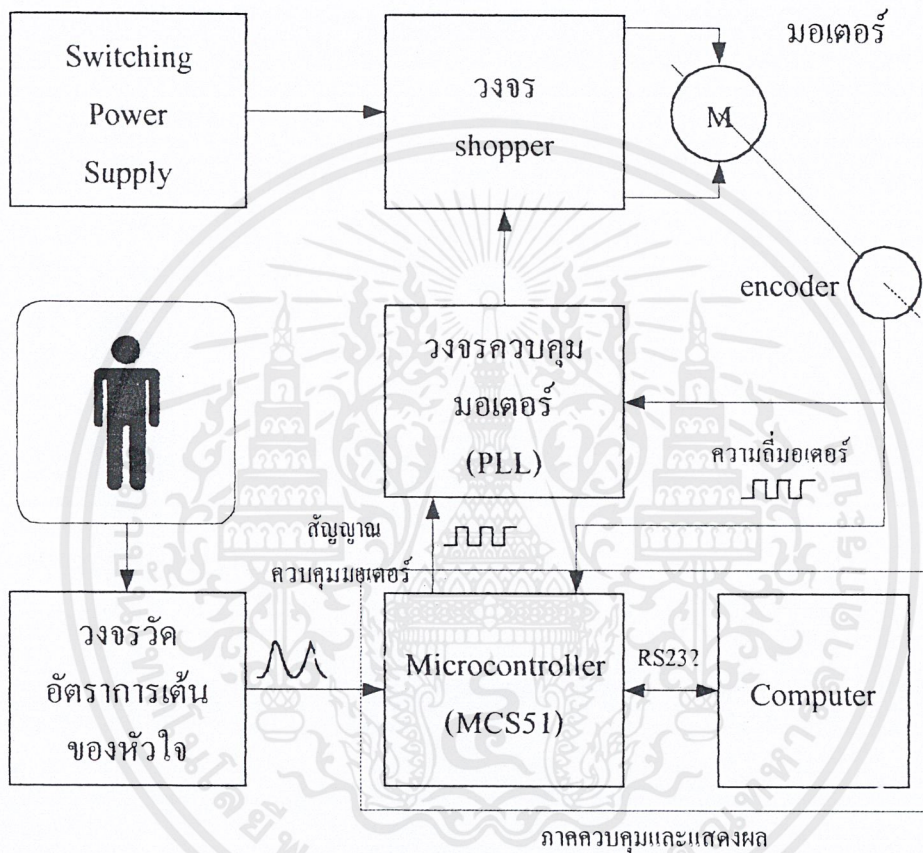
1.2 ขอบเขตของปรีญญาณีพนธ์

จากบล็อกการทำงานของเครื่องวิ่งออกกำลังกายแสดงดังรูปที่ 1.1 คอมพิวเตอร์จะเป็นตัวควบคุมความเร็วของเครื่องวิ่งและแสดงผล โดยค่าความเร็วที่ถูกกำหนดจากคอมพิวเตอร์จะถูกส่งมายัง Micro controller (MCS51) เมื่อ Micro controller รับค่าความเร็วจากคอมพิวเตอร์แล้ว ก็จะมีการสร้างสัญญาณพัลส์ ส่งไปยังวงจรควบคุมความเร็วมอเตอร์ โดยวงจรควบคุมความเร็วมอเตอร์จะใช้เทคนิคของ Phase Locked Loop (PLL) ซึ่งวงจรจะควบคุมให้ความถี่มอเตอร์ (ความถี่ที่ได้จากการถอดรหัสความเร็วมอเตอร์ ความถี่นี้จะได้กับการแปรผันตรงกับความเร็วมอเตอร์) เท่ากับความถี่ของสัญญาณควบคุม วงจรควบคุมความเร็วมอเตอร์ซึ่งใช้เทคนิคของ Phase Locked Loop นี้ จะทำการสร้างแรงดันเอาท์พุท ซึ่งแรงดันนี้จะแปรผันตรงกับความต่างเฟสระหว่างความถี่มอเตอร์และความถี่ควบคุมส่งไปยังวงจร Chopper เพื่อควบคุมความกว้างของพัลส์ที่จะจ่ายไปขับมอเตอร์ ทำให้มอเตอร์มีความเร็วตามที่ต้องการ

ในส่วนของการแสดงผล ไมโครคอนโทรลเลอร์จะวัดคาบเวลาของสัญญาณการเดินของหัวใจในหนึ่งลูกคลื่น ส่งไปยังคอมพิวเตอร์ เมื่อคอมพิวเตอร์ได้รับคาบเวลานี้แล้ว ก็จะมีการแปลงเป็นอัตราการเดินของหัวใจ แล้วแสดงผล สำหรับการวัดความเร็วและระยะทางในการวิ่ง จะวัดจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่มอเตอร์ซึ่งแปรผันตรงกับความเร็วมอเตอร์ ซึ่งก็หมายความว่าแปรผันตรงกับความเร็วของเครื่องวัดด้วย ความถี่ของมอเตอร์จะถูกวัดโดยไมโครคอนโทรลเลอร์ แล้วส่งไปยังคอมพิวเตอร์ คอมพิวเตอร์จะทำการแปลงให้เป็นความเร็วของเครื่องวัดและระยะทางในการวัด แล้วแสดงผล



รูปที่ 1.1 บล็อกแสดงการทำงานของเครื่องวัดออกกำลังกาย

สำหรับปฏิยานิพนธ์นี้จะทำเฉพาะในส่วนของวงจรควบคุมมอเตอร์ , วงจรวัดอัตราการเต้นของหัวใจ , Micro controller (MCS51) และ โปรแกรมคอมพิวเตอร์ที่ใช้ในการควบคุมเท่านั้น ดังนั้นในส่วน of Switching Power Supply และ วงจร Chopper จะไม่มีรายละเอียดในปฏิยานิพนธ์นี้ซึ่งถ้าสนใจสามารถหาอ่านได้ในปฏิยานิพนธ์เรื่อง Switching Power Supply for DC Motor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 เนื้อหาโดยสังเขป

รายละเอียดของปฏิญานิพนธ์จะแบ่งเป็นบทๆดังนี้ โดย

บทที่ 1 จะกล่าวถึงความสำคัญและความเป็นมาของปฏิญานิพนธ์ ตลอดจนขอบเขตของการศึกษาทดลอง

บทที่ 2 จะกล่าวถึงทฤษฎีการควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูลูป อธิบายถึงส่วนประกอบของระบบ และการวิเคราะห์เสถียรภาพของระบบ ซึ่งใช้ในการออกแบบ

บทที่ 3 จะเป็นส่วนของการออกแบบวงจรควบคุมความเร็วมอเตอร์ ซึ่งใช้เทคนิค เฟสล็อกคูลูป กล่าวถึงการวิเคราะห์เสถียรภาพโดยการใช้รูสโลคัส และการทดลองวัดผลแสดงสัญญาณของระบบเฟสล็อกคูลูปในการขับมอเตอร์

บทที่ 4 จะกล่าวถึงทฤษฎีและการออกแบบวงจรการวัดอัตราการเดินของหัวใจ พร้อมทั้งแสดงผลการตรวจวัดสัญญาณ

บทที่ 5 จะกล่าวถึงภาคควบคุมและแสดงผล โดยประกอบด้วยส่วนของวงจรที่ใช้ไมโครคอนโทรลเลอร์ 89C2051 และส่วนของโปรแกรมควบคุมการทำงาน

บทที่ 6 จะกล่าวถึงการใช้งานโปรแกรมซึ่งเขียนด้วยโปรแกรม Visual Basic

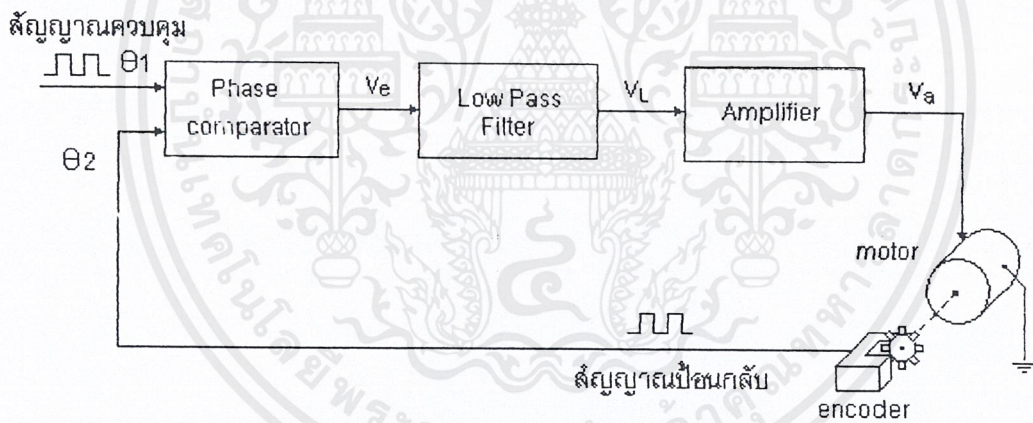
บทที่ 7 เป็นการสรุปผลจากการทดลองใช้งาน ปัญหาที่พบ รวมไปถึงแนวทางการพัฒนา

บทที่ 2

ทฤษฎีการควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูล

2.1 การทำงานของระบบ

บล็อกการทำงานของระบบแสดงดังรูปที่ 2.1 ความเร็วของมอเตอร์จะถูกแปลงเป็นสัญญาณพัลส์ด้วย Encoder เพื่อเป็นสัญญาณป้อนกลับไปยัง Phase comparator ซึ่งจะทำหน้าที่เปรียบเทียบเฟสของสัญญาณควบคุมและสัญญาณป้อนกลับ และจะให้เอาต์พุตเป็นสัญญาณเออร์เรอร์ที่แปรผันตรงกับความต่างเฟสของสัญญาณทั้งสอง จากนั้นสัญญาณเออร์เรอร์จะถูกกรองความถี่ต่ำผ่านด้วย Low Pass Filter ส่งไปยัง Amplifier เพื่อขยายสัญญาณเออร์เรอร์ให้สูงขึ้น สัญญาณที่ขยายด้วย Amplifier จะถูกส่งไปขับมอเตอร์ เพื่อให้ความเร็วมอเตอร์สอดคล้องกับสัญญาณควบคุม หรือสัญญาณป้อนกลับเท่ากับสัญญาณควบคุม



รูปที่ 2.1 บล็อกการทำงานของเฟสล็อกคูล

2.2 ส่วนประกอบของระบบควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูล

2.2.1 Phase comparator

เฟสคอมพาราเตอร์ เป็นตัววัดเฟสความต่างเฟสระหว่างสัญญาณบังคับ และสัญญาณป้อนกลับซึ่งจะให้สัญญาณที่แปรผันตรงกับความต่างเฟสของสัญญาณทั้งสอง โดยสัญญาณเออร์เรอร์จะเป็นไปตามสมการ

$$V_c = k_p(\theta_1 - \theta_2) \quad (2.1)$$

โดย V_c คือ สัญญาณเออร์เรอร์จากความต่างเฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

k_p คือ อัตราการขยายของคอมพาราเตอร์

θ_1 คือ เฟสของสัญญาณควบคุม

θ_2 คือ เฟสของสัญญาณป้อนกลับ

2.2.2 Low Pass Filter

Low Pass Filter จะทำหน้าที่กรองสัญญาณความถี่สูงจากเอาต์พุตของ Phase comparator โดย Low Pass Filter ที่ใช้ในระบบนี้จะมี Transfer function เป็น

$$F(S) = \frac{1 + S\tau_2}{S\tau_1} \quad (2.2)$$

2.2.3 Amplifier

Amplifier ทำหน้าที่เพิ่มกำลังของสัญญาณเออร์เรอร์เพื่อใช้สำหรับขับเคลื่อนมอเตอร์ โดย Transfer function ของ Amplifier คือ k_a เนื่องจาก pole ของ amplifier สูงกว่า pole ของมอเตอร์มาก ดังนั้น Amplifier จึงมี gain เท่ากับ k_a ตลอดช่วงความถี่ของระบบ

2.2.4 Motor

ความเร็วเชิงมุม (angular speed) $\omega(t)$ ของมอเตอร์ เมื่อป้อนด้วยแรงดันขนาด U_f กำหนดโดยสมการ

$$\omega(t) = U_f k_m \left(1 - e^{-t/T_m} \right) \quad (2.3)$$

โดย k_m คือ proportional gain

T_m คือ ค่า time constant เชิงกลของมอเตอร์

เมื่อนำสมการที่ 1 มาทำ Laplace transform จะได้

$$\Omega(S) = U_f(S) \frac{k_m}{1 + ST_m} \quad (2.4)$$

มุมเฟส Φ ของมอเตอร์ ได้จาก อินทิเกรต ความเร็วเชิงมุม ω เทียบกับเวลาจะได้ Laplace transform $\Phi(S)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\Phi(S) = \frac{k_m}{S(1+ST_m)} U_f(S) \quad (2.5)$$

จากสมการ 2.5 เราจะได้ Transfer function ของมอเตอร์

$$H_m(S) = \frac{\Phi(S)}{U_f(S)} = \frac{k_m}{S(1+ST_m)} \quad (2.6)$$

2.2.5 Encoder

เอ็นโคเดออร์เป็นอุปกรณ์กำเนิดพัลส์เพื่อเป็นสัญญาณป้อนกลับ โดยความถี่ของพัลส์ ที่สร้างขึ้นจากเอ็นโคเดออร์ จะแปรผันตรงกับความเร็วมอเตอร์ เอ็นโคเดออร์แบบนี้เรียกว่า optical tachometer generator จำนวนพัลส์ของเอ็นโคเดออร์ที่สร้างขึ้นมาจะขึ้นกับจำนวนซี่ของฟันจาน ซึ่งเขียนเป็นสมการได้เป็น

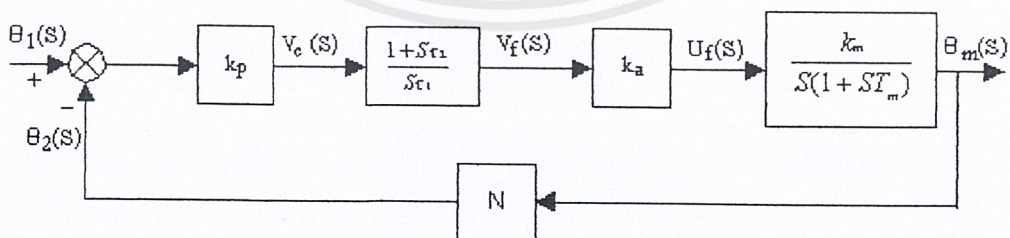
$$\theta_2 = N\theta_m \quad (2.7)$$

เมื่อ N คือ จำนวนซี่ของฟันจาน

θ_2 คือ มุมเฟสของสัญญาณป้อนกลับ

θ_m คือ มุมเฟสของมอเตอร์

2.3 บล็อกไดอะแกรมของระบบ



รูปที่ 2.2 บล็อก ไดอะแกรม ระบบควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกคูล

บล็อกไดอะแกรมของระบบแสดงดังรูป 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากบล็อกไดอะแกรมเราได้ $G(S)H(S)$ ซึ่งเป็น open loop gain

$$G(S)H(S) = \frac{k_p k_a k_m N}{\tau_1} * \frac{1 + S\tau_2}{S^2(1 + ST_m)}$$

$$G(S)H(S) = k \frac{1 + S\tau_2}{S^2(1 + ST_m)} \quad (2.8)$$

เมื่อ $k = \frac{k_p k_a k_m N}{\tau_1}$

ในการออกแบบและวิเคราะห์เสถียรภาพของระบบ เราสามารถทำได้โดยการใช้ทฤษฎีต่างๆ เช่น Root Locus , Routh Hurwitz , Bode Plot

2.4 การวิเคราะห์เสถียรภาพโดยใช้รูทโลคัส

พิจารณาระบบเปิด Single Input Single Output (SISO) ซึ่งมีทรานสเฟอร์ฟังก์ชันของระบบเป็น

$$T(S) = \frac{kG(S)}{1 + kG(S)H(S)} \quad (2.9)$$

จากทรานสเฟอร์ฟังก์ชันของระบบ k เป็นค่าคงที่ ถ้าเราสมมติ

$$H(S) = 1$$

$$G(S) = \frac{1}{S(S+2)}$$

นำค่าแทนในสมการ 2.9 เราจะได้

$$T(S) = \frac{k}{S^2 + 2S + k} \quad (2.10)$$

จากทรานสเฟอร์ฟังก์ชันในสมการที่ 2.10 จะเห็นว่าโพลของระบบเป็นฟังก์ชันของ k นั่นคือ โพลของระบบจะเปลี่ยนแปลงตามค่า k นั้นเอง

ดังนั้นเทคนิคของรูทโพลคือ การพล็อตทางเดินของราก เมื่อค่า k เปลี่ยนแปลงไป โดยค่า k จะเริ่มตั้งแต่ $-\infty$ จนถึง $+\infty$ จากนั้นก็จะพิจารณารากของสมการที่ค่า k ต่างๆ ซึ่งทางเดินของรากก็คือทางเดินของโพลในระบบ ถ้าหากโพลของระบบอยู่ทางด้านขวาของแกนจินตภาพ ก็จะทำให้ระบบไม่มีเสถียรภาพ ดังนั้นหากค่า k ตัวใดที่ทำให้รากของสมการอยู่ทางด้านขวาของแกนจินตภาพ ค่า k ตัวนั้นก็จะเป็ค่า k ที่ทำให้ระบบไม่มีเสถียรภาพ ซึ่งเราจำเป็นต้องหลีกเลี่ยง

2.4.1 การพล็อตทางเดินของราก

จากทรานสเฟอร์ฟังก์ชันสมการที่ 2.9 เราพิจารณาเฉพาะตัวส่วนซึ่งเมื่อทำให้มีค่าเท่ากับศูนย์ เราก็สามารถหารากสมการของโพลได้

$$1 + kG(S)H(S) = 0 \quad (2.11)$$

สมการที่ 2.11 เรียกว่าสมการคุณลักษณะ (Characteristic Equation) หรือเขียนได้เป็น

$$G(S)H(S) = \frac{-1}{k} \quad (2.12)$$

จากสมการ 2.12 เราจะได้

$$|G(S)H(S)| = \frac{1}{|k|} \quad (2.13)$$

$$\angle G(S)H(S) = r * 180^\circ \quad (2.14)$$

โดย $0 < k < \infty$; $r = 1, 3, 5, \dots$

ปกติ $G(S)H(S)$ จะอยู่ในรูปเศษส่วนดังนี้

$$G(S)H(S) = \frac{(S + Z_1)(S + Z_2) \dots (S + Z_m)}{(S + P_1)(S + P_2) \dots (S + P_n)}$$

เมื่อนำไปแทนค่าในสมการ 2.13 และ 2.14 จะได้

$$|G(S)H(S)| = \frac{|S + Z_1| |S + Z_2| \dots |S + Z_m|}{|S + P_1| |S + P_2| \dots |S + P_n|} = \frac{1}{|k|} \quad (2.15)$$

$$\angle G(S)H(S) = \left(\sum_{i=1}^m \angle S + Z_i \right) \left(\sum_{j=1}^n \angle S + P_j \right) = r * 180^\circ \quad (2.16)$$

โดย $0 < k < \infty$; $r = 1, 3, 5, 7$

จากสมการที่ 2.16 ที่ S ใดๆที่ทำให้ผลรวมของมุมเป็นไปตามสมการ ที่จุด S นั้นจะเป็นเส้นทางที่รากสมการตัดผ่าน และจุด S นั้นก็จะเป็นรากของสมการด้วย โดยเราสามารถหาค่า k ได้ด้วยการนำค่า S ที่ได้แทนในสมการ 2.15 ก็จะได้ค่า k ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 วิธีการพล็อตทางเดินของราก

การสร้างทางเดินของรากมีกฎเกณฑ์ต่างๆที่ช่วยในการสร้างทางเดินของราก แต่ก็ไม่สามารถพล็อตได้อย่างถูกต้อง วิธีการพล็อตในที่นี้จะแสดงเฉพาะค่า k เป็นบวกเท่านั้น เนื่องจากค่า k เป็นลบมักทำให้ระบบไม่มีเสถียรภาพ จึงไม่ขอแสดงไว้ในที่นี้

- จุดของราก เมื่อค่า $k=0$ และ $k=\infty$

ที่ $k=0$ รากของสมการก็คือ โพลต่างๆของ $G(S)H(S)$ ส่วนเมื่อ $k=\infty$ รากของสมการหรือตำแหน่งของโพลจะ ไปอยู่ที่ตำแหน่งของซีโรที่อินฟินิตี้

- มุมของเส้นกำกับ (Angle of Asymptotes)

เส้นกำกับหรือเส้นทางเดินของรากจะวิ่งเข้าหาหรือวิ่งไปตามเส้นกำกับนี้ เมื่อค่า $k \rightarrow \infty$ โดยมุมของเส้นกำกับสามารถหาได้โดย

$$\theta_i = \frac{r * 180^\circ}{|n - m|} \quad (2.17)$$

เมื่อ

$$r = 1, 3, 5, \dots$$

$$n = \text{จำนวนโพลของ } G(S)H(S)$$

$$m = \text{จำนวนซีโรของ } G(S)H(S)$$

- จุดตัดของเส้นกำกับ (Intersect of the Asymptotes)

เป็นจุดที่เส้นกำกับตัดกับแกนจริง หาได้โดย

$$\sigma = \frac{\sum \text{finite poles of } G(S)H(S) - \sum \text{finite zeros of } G(S)H(S)}{n - m} \quad (2.18)$$

$$n = \text{จำนวนโพลของ } G(S)H(S)$$

$$m = \text{จำนวนซีโรของ } G(S)H(S)$$

- จุดที่ทางเดินของรากชนกัน (Breakaway Point)

เป็นจุดที่ทางเดินของรากวิ่งมาชนกัน ซึ่งตำแหน่งนี้จะเกิดรากซ้ำขึ้น จุดนี้หาได้จากค่า S

เมื่อ

$$\frac{d}{ds} G(S)H(S) = 0 \quad (2.19)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง พล็อตทางเดินของราก โดยค่า k เริ่มจาก 0 ถึง ∞ เมื่อ

$$G(S)H(S) = \frac{S+4}{S(S+2)}$$

- เมื่อ $k = 0$ รากสมการอยู่ที่ 0, -2

$k = \infty$ รากสมการอยู่ที่ -4, ∞

- มุมของเส้นกำกับ (Angle of Asymptotes)

$$\theta_i = \frac{r * 180^\circ}{|2 - 1|} = 180^\circ$$

- จุดตัดของเส้นกำกับ (Intersect of the Asymptotes)

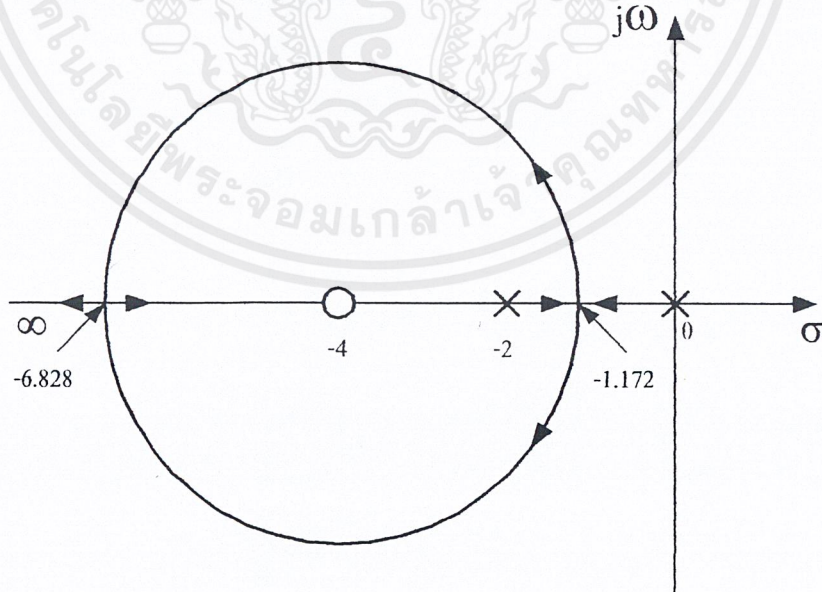
เนื่องจาก $\theta_i = 180^\circ$ จึงไม่มีจุดตัดแกนจริง

- จุดที่ทางเดินของรากชนกัน (Breakaway Point)

$$\frac{d}{ds} \frac{S+4}{S(S+2)} = 0$$

$$S^2 + 8S + 8 = 0$$

$$S = -1.172, -6.828$$



รูปที่ 2.3 ตัวอย่างวิธีการพล็อตทางเดินของราก

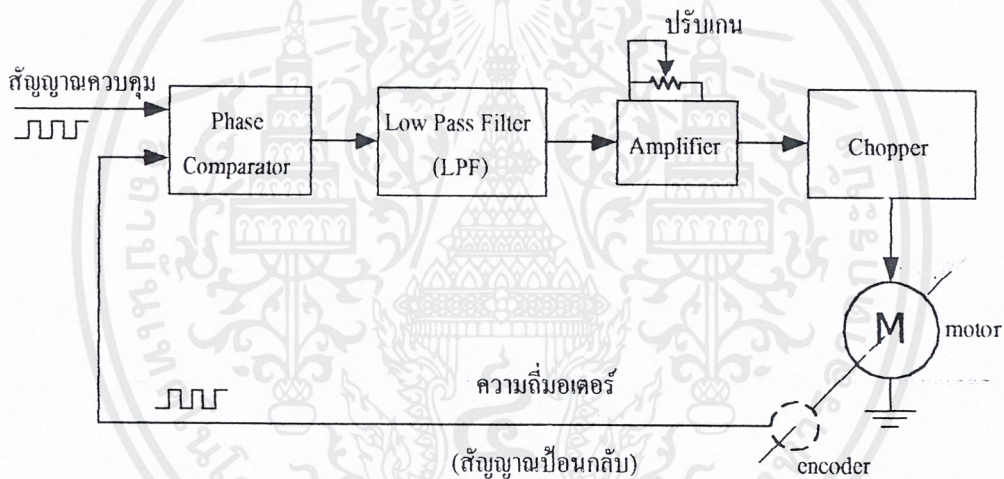
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วงจรควบคุมความเร็วมอเตอร์

3.1 บทคัดย่อการทำงานของระบบควบคุมความเร็วมอเตอร์ด้วยเทคนิคเฟสล็อกกลูป

วงจรควบคุมความเร็วมอเตอร์ใช้เทคนิคของ Phase Locked Loop แสดงบล็อกการทำงานของรูปที่ 3.1 หลักการของการควบคุมความเร็วมอเตอร์ด้วยเฟสล็อกกลูป ได้อธิบายไว้แล้วในทฤษฎีบทที่ 2 ต่อไปจะอธิบายการออกแบบ และหน้าที่การทำงานของแต่ละบล็อก



รูปที่ 3.1 บล็อกการทำงานของวงจรควบคุมความเร็วมอเตอร์

3.1.1 Phase comparator ทำหน้าที่เปรียบเทียบเฟส ระหว่างสัญญาณควบคุมและสัญญาณป้อนกลับ โดยจะให้สัญญาณเอาต์พุตที่เป็นแรงดันซึ่งแปรผันตรงกับความต่างเฟสระหว่างสัญญาณควบคุมและสัญญาณป้อนกลับ ในโครงงานนี้ใช้ ไอซี Phase comparator เบอร์ 4046 โดยความสัมพันธ์ระหว่างแรงดันเอาต์พุต (V_e) กับความต่างเฟสระหว่าง เฟสสัญญาณควบคุม (θ_1) และสัญญาณป้อนกลับ (θ_2) เขียนได้เป็น

$$V_e = k_p(\theta_1 - \theta_2) \quad (3.1)$$

โดย k_p คือเกนของ Phase comparator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

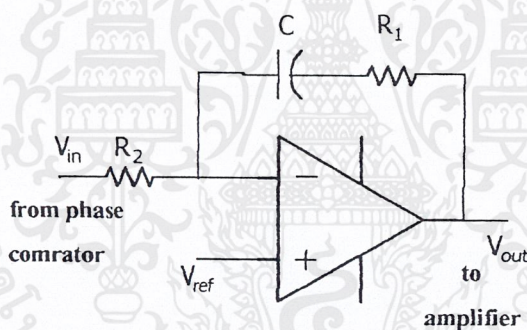
ในการออกแบบใช้ Phase comparator 2 (PC2) ภายใน IC 4046 ซึ่งมีค่าเกนของ Phase comparator (k_p) เป็น

$$k_p = \frac{V_{DD}}{4\pi} \quad (3.2)$$

โดยแหล่งจ่ายไฟที่ใช้มีค่า 5 V ดังนั้นค่าของ $k_p = 5/4\pi = 0.398 \text{ v/Rad}$.

3.1.2 Low Pass Filter (LPF) ทำหน้าที่ในการกรองความถี่ต่ำผ่าน เป็นตัวชดเชยเฟสเพื่อให้ระบบมีเสถียรภาพ โดยวงจร Low Pass Filter ที่เลือกใช้แสดงดังรูปที่ 3.2 ความสัมพันธ์ระหว่างแรงดันเอาต์พุต (V_{out}) กับแรงดันอินพุต (V_{in}) เขียนได้เป็น

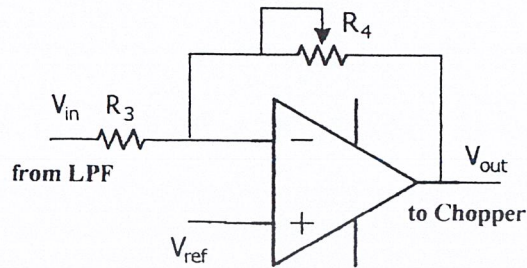
$$\frac{V_{out}}{V_{in}} = \frac{1 + sR_1C}{sR_2C} \quad (3.3)$$



รูปที่ 3.2 วงจร Low Pass Filter

3.1.3 Amplifier เป็นวงจรที่ทำหน้าที่ขยายแรงดันความต่างเฟส ที่ผ่านการกรองจากวงจร Low Pass Filter ให้แรงขึ้นเพื่อจะส่งไปยังอินพุตของวงจร Chopper โดยวงจรแสดงดังรูปที่ 3.3 ซึ่งสามารถปรับอัตราการขยายของวงจรได้ที่ R_4 การออกแบบให้วงจรในส่วนนี้สามารถปรับอัตราการขยายได้ ก็เนื่องมาจากในระบบควบคุมแบบลูปปิด (Close Loop System) ค่าอัตราการขยายของระบบจะมีผลต่อตำแหน่งของโพลในระบบ นั่นคือการเปลี่ยนอัตราการขยายของระบบจะส่งผลให้พารามิเตอร์ เช่น Overshoot, Setting time เปลี่ยนแปลงตามไปด้วย ดังนั้นวงจรในส่วนนี้จะถูกใช้ในการปรับอัตราการขยายของระบบ เพื่อให้ได้ค่าพารามิเตอร์ที่เหมาะสม จากวงจรในรูปที่ 3.3 ได้ความสัมพันธ์ระหว่างแรงดันเอาต์พุต (V_{out}) กับแรงดันอินพุต (V_{in}) เป็น

$$\frac{V_{out}}{V_{in}} = \frac{R_4}{R_3} \quad (3.4)$$

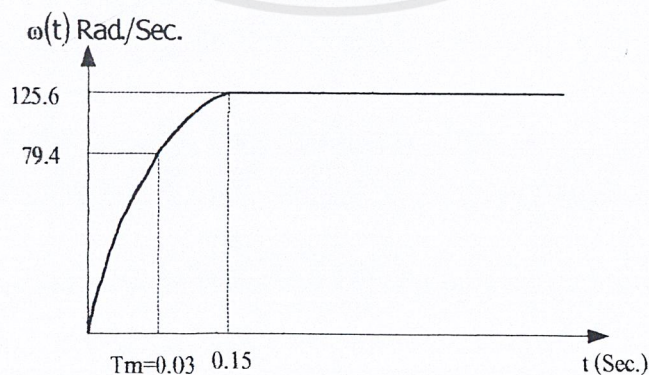


รูปที่ 3.3 วงจรขยายปรับเกนได้

3.1.4 วงจร Chopper ดังที่กล่าวมาแล้วในบทนำว่า ปริมาณอินพุตนี้ไม่ได้ทำในส่วนขอ วงจร Chopper แต่จำเป็นต้องใช้ ทรานสเฟอ์ฟังก์ชันของวงจร Chopper มาใช้ในการออกแบบ ซึ่งก็คืออัตราส่วนของแรงดันเฉลี่ยเอาต์พุตต่อแรงดันอินพุต ซึ่งก็คืออัตราขยายของวงจร Chopper นั้นเอง โดยมีค่าเท่ากับ 9 เท่า (ค่านี้ได้จากปริมาณอินพุตเรื่อง Switching Power Supply for DC.Motor)

3.1.5 Motor มอเตอร์ที่ใช้เป็น DC มอเตอร์ 36 V, 280 W, ความเร็วสูงสุด 1200 rpm. จากการทดลองวัดความเร็วของมอเตอร์ที่เวลาต่างๆ เมื่อจ่ายแรงดันเท่ากับ 36 V แสดงดังกราฟรูปที่ 3.4 และสามารถเขียนสมการความเร็วเชิงมุมของมอเตอร์ได้เป็น

$$\omega(t) = u_f k_m \left(1 - e^{-t/T_m} \right) \quad (3.5)$$



รูปที่ 3.4 แสดงความเร็วเชิงมุมของมอเตอร์ที่เวลาต่างๆเมื่อป้อนแรงดัน 36 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ 3.5 k_m คืออัตราส่วนระหว่างความเร็วเชิงมุมของมอเตอร์ในขณะที่ความเร็วคงที่ ต่อแรงดันที่จ่ายให้มอเตอร์ เขียนสมการได้เป็น

$$k_m = \frac{\omega}{u_f} \quad (3.6)$$

จากการทดลองเมื่อป้อนแรงดัน 36 V วัดความเร็วเชิงมุมในขณะที่ความเร็วคงที่ได้ 125.6 Rad./Sec. ดังนั้น k_m ของมอเตอร์มีค่าเท่ากับ

$$k_m = \frac{125.6}{36} = 3.49 \text{ Rad / Sec. / V} \quad (3.7)$$

ส่วน T_m คือ Time constant ของมอเตอร์ ซึ่งเป็นเวลาที่มอเตอร์มีความเร็ว 63.2 % ของความเร็วในสภาวะคงที่ จากการทดลองวัดได้ 30 ms.

เราสามารถหาทรานสเฟอ์ฟังก์ชันของมอเตอร์โดยการทำ Laplace transform กับสมการที่ 3.5 ซึ่งได้ $\omega(s)$ เป็น

$$\omega(s) = u_f(s) \frac{k_m}{s + \frac{1}{T_m}} \quad (3.8)$$

มุมเฟส (θ_m) ของมอเตอร์ ได้จาก อินทิเกรท ความเร็วเชิงมุม (ω) เทียบกับเวลา (t) จะได้ Laplace transform $\theta_m(s)$

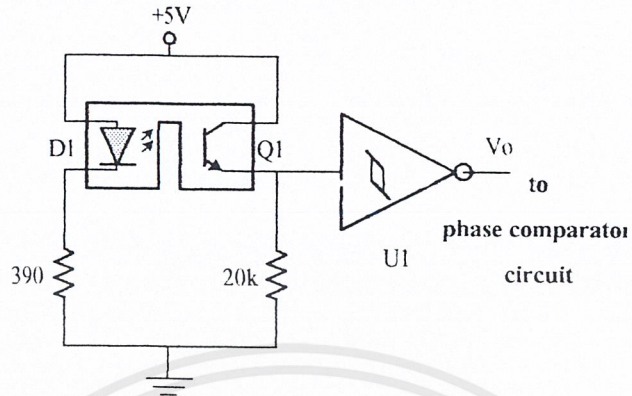
$$\theta_m(s) = \frac{k_m}{s \left(s + \frac{1}{T_m} \right)} u_f(s) \quad (3.9)$$

จากสมการ 3.9 จะได้ทรานสเฟอ์ฟังก์ชันของมอเตอร์เป็นอัตราส่วนระหว่างมุมเฟส (θ_m) ต่อแรงดันอินพุท (u_f) เป็น

$$\frac{\theta_m(s)}{u_f(s)} = \frac{k_m}{s \left(s + \frac{1}{T_m} \right)} \quad (3.10)$$

3.1.6 เอ็นโคเดอร์ (Encoder) ทำหน้าที่ถอดรหัสความเร็วของมอเตอร์ออกมาเป็นความถี่พัลส์ โดยค่าความถี่แปรผันตรงกับความเร็วมอเตอร์ เอ็นโคเดอร์ที่ใช้เป็นแบบ optical tachometer generator ซึ่งวงจรการทำงานแสดง ดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 วงจรเซ็นโคดเดอร์แบบ optical tachometer generator

จากวงจร transistor จะหยุดทำงานเมื่อฟันของจานเข้ามาบังแสงของ LED และจะทำงานเมื่อแสงจาก LED ผ่านร่องของจานมายัง transistor สัญญาณที่ได้ถูกส่งเข้า Schmitt-trigger ซึ่งจะช่วยให้เวลาได้ขึ้นและไต่ลงของสัญญาณน้อยลง

จำนวนพัลส์ที่ encoder สร้างขึ้นมาจะขึ้นกับจำนวนซี่ของฟันจาน สามารถเขียนสมการความสัมพันธ์ระหว่างเฟสของเซ็นโคดเดอร์กับเฟสของมอเตอร์ได้เป็น

$$\theta_2 = N\theta_m \quad (3.11)$$

เมื่อ N คือ จำนวนซี่ของฟันจาน

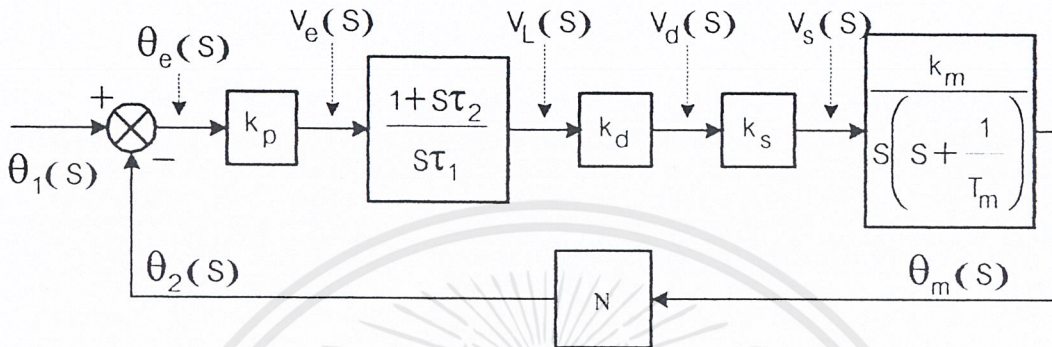
θ_2 คือ มุมเฟสของเซ็นโคดเดอร์

θ_m คือ มุมเฟสของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบและวิเคราะห์เสถียรภาพของระบบด้วยเทคนิคครุทโลคัส

จากบล็อกการทำงานในรูปที่ 3.1 เมื่อแทนค่าทรานสเฟอร์ฟังก์ชันแต่ละบล็อก จะได้ บล็อกไดอะแกรมของระบบดังรูปที่ 3.6



รูปที่ 3.6 แสดงบล็อกไดอะแกรมของระบบ

โดย	k_p	คือ อัตราการขยายของ Phase Comparator (V/Rad)
	k_d	คือ อัตราการขยายวงจรถ่ายปรับเกนได้ (V/V)
	k_s	คือ อัตราการขยายของวงจร Chopper (V/V)
	k_m	คือ Proportional gain ของมอเตอร์ (Rad/V)
	T_m	คือ Time constant เชิงกลของมอเตอร์ (Sec)
	N	คือ จำนวนซี่ของฟันจานเอ็น โคคเคอร์

จากบล็อกไดอะแกรมรูปที่ 3.6 ได้ทรานสเฟอร์ฟังก์ชัน $G(S)H(S)$ เป็น

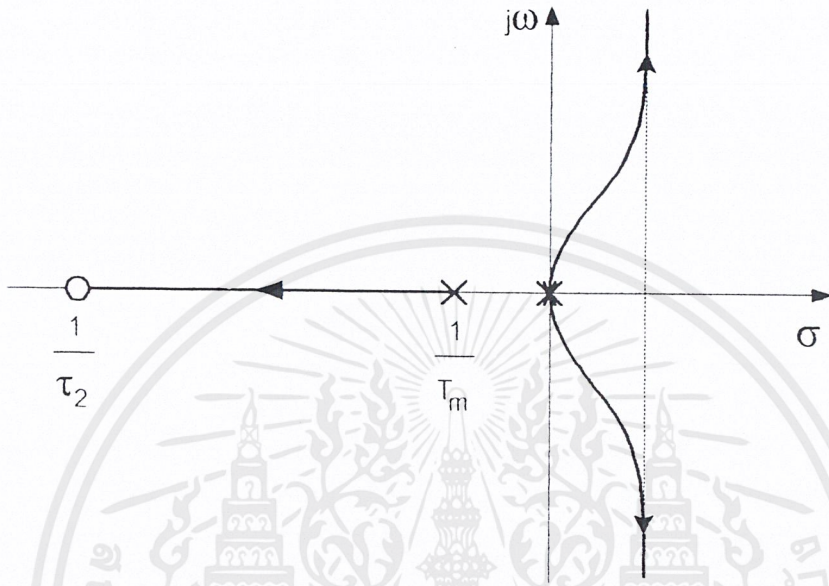
$$G(S)H(S) = \frac{k_p k_d k_s k_m N}{\tau_1} * \frac{1 + S\tau_2}{S^2 \left(S + \frac{1}{T_m} \right)} \quad (3.12)$$

$$G(S)H(S) = k \frac{1 + S\tau_2}{S^2 \left(S + \frac{1}{T_m} \right)} \quad (3.13)$$

โดย
$$k = \frac{k_p k_d k_s k_m N}{\tau_1} \quad (3.14)$$

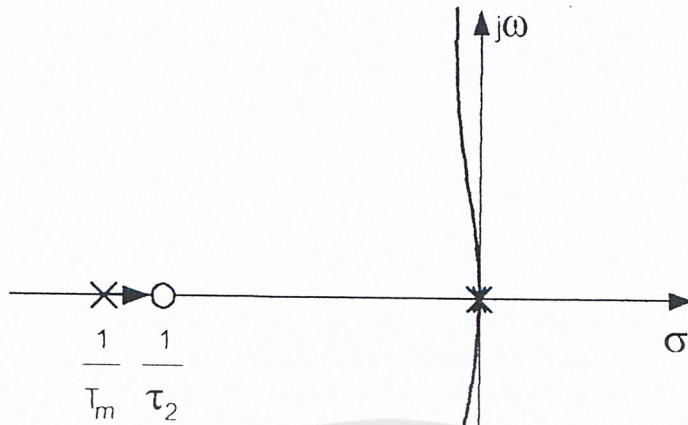
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ 3.13 ถ้าเราสมมติให้ τ_2 มีค่าน้อยกว่า T_m แล้วนำ $G(S)H(S)$ ไปทำการพล็อต ครุทโลคัสจะได้ดังรูปที่ 3.3 จากรูปที่ 3.7 จะเห็นว่ารากของสมการอยู่ด้านขวาของแกนจินตภาพ นั่นคือถ้า τ_2 มีค่าน้อยกว่า T_m ระบบจะไม่มีเสถียรภาพ (Unstable)

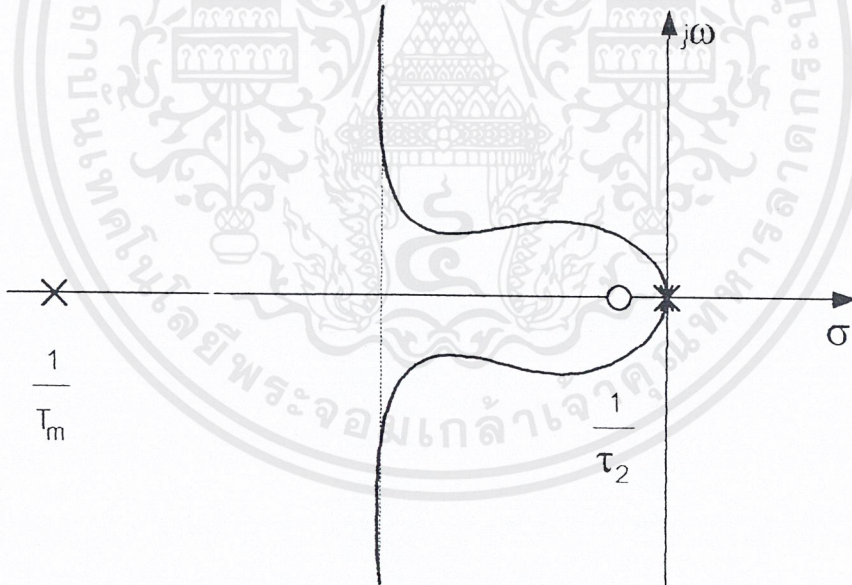


รูปที่ 3.7 ทางเดินของรากเมื่อ τ_2 มีค่าน้อยกว่า T_m

พิจารณาเมื่อ τ_2 มีค่ามากกว่า T_m โดยจะพิจารณา 2 กรณี คือกรณีที่ τ_2 มีค่าใกล้เคียง T_m หรือมากกว่า T_m เล็กน้อย และกรณีที่ τ_2 มีค่ามากกว่า T_m มากๆ นำค่าทั้งสองกรณีไปทำการพล็อตครุทโลคัส จะได้ดังรูปที่ 3.8 และ 3.9 ตามลำดับ จากทางเดินของรากของรูปทั้งสองจะเห็นได้ว่า กรณีที่ τ_2 มีค่ามากกว่า T_m เล็กน้อย ระบบจะเร็ว โอเวอร์ชูทของระบบจะสูง ส่วนกรณีที่ τ_2 มีค่ามากกว่า T_m มากๆ ระบบจะช้า โอเวอร์ชูทจะต่ำ ดังนั้นในการออกแบบจะต้องเลือก τ_2 ที่เหมาะสมเพื่อให้ระบบมีโอเวอร์ชูทไม่มากเกินไป และความเร็วของระบบจะต้องไม่ช้าจนเกินไป



รูปที่ 3.8 แสดงเส้นทางเดินของรากกรณี τ_2 มีค่ามากกว่า T_m เล็กน้อย



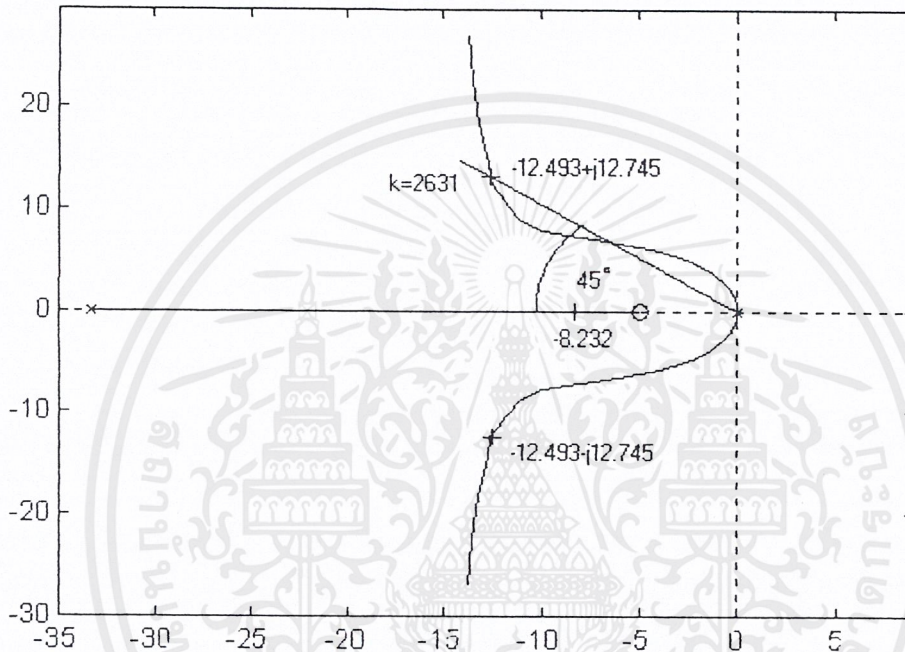
รูปที่ 3.9 แสดงเส้นทางเดินของรากกรณี τ_2 มีค่ามากกว่า T_m มากๆ

ในระบบที่ออกแบบ มอเตอร์ที่ใช้วัดค่า T_m ได้ 30 mSec. กำหนดค่า τ_2 ไว้ที่ 0.2 Sec.
แทนค่าในสมการ 3.13 ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$G(s)H(s) = \frac{k(1 + 0.2s)}{s^2(1 + 0.03s)} \quad (3.15)$$

นำค่า $G(s)H(s)$ จากสมการ 3.15 มาพล็อตครุฑโกลด์สโดยใช้โปรแกรม MathLab ได้เส้นทางเดินของราก ดังรูปที่ 3.10



รูปที่ 3.10 แสดงทางเดินของรากของระบบ

จากรูปที่ 3.10 โพลของระบบจะมี 3 ตัว เป็นโพลบนแกนจริง 1 ตัว และมีโพลที่เป็นเชิงซ้อน 2 ตัวเป็นคู่ conjugate กัน โดยเราจะพิจารณาโพลคู่นี้ในรูปของผลตอบสนองของสมการ Second Order ซึ่งมีรูปทั่วไปเป็น

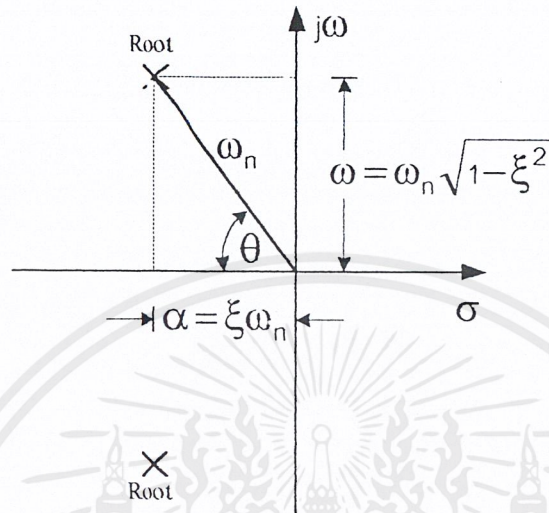
$$T(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (3.16)$$

รากสมการมี 2 ค่า คือ

$$\begin{aligned} S_1, S_2 &= -\xi\omega_n \pm j\omega_n\sqrt{1 - \xi^2} \\ &= -\alpha \pm j\omega \end{aligned} \quad (3.17)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่ารากสมการ S_1, S_2 กำหนดลงบนระนาบเชิงซ้อน เราได้ความสัมพันธ์ของ α, ξ, ω_n และ ω แสดงดังรูปที่ 3.11



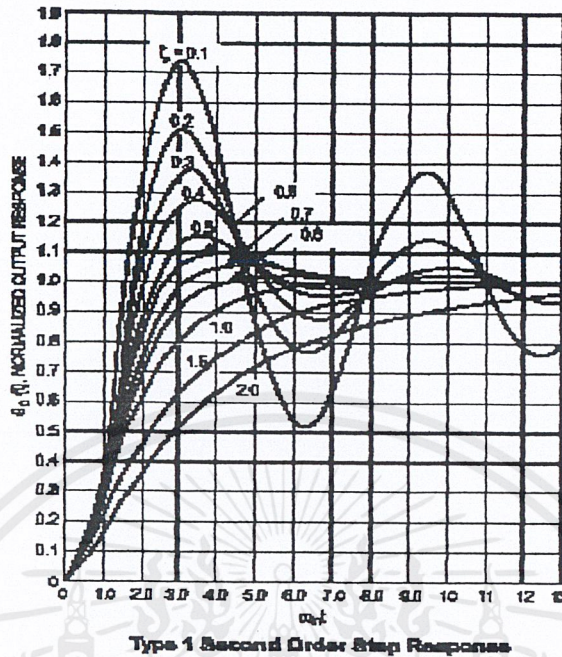
รูปที่ 3.11 แสดงรากของสมการและความสัมพันธ์ระหว่าง α, ξ, ω_n และ ω

จากรูปที่ 3.11 เราได้

$$\xi = \cos \theta \quad (3.18)$$

โดย ξ ถูกเรียกว่า Damping Ratio ซึ่งเป็นตัวกำหนด Over Shoot ของระบบ ซึ่งเปอร์เซ็นต์ Over Shoot แสดงดังสมการ

$$\text{Percent maximum overshoot} = 100e^{-\frac{\pi \xi}{\sqrt{1-\xi^2}}} \quad (3.19)$$



รูปที่ 3.12 แสดง maximum overshoot ที่ค่า damping ratio (ζ) ค่าต่างๆ

กลับไปดูรูปที่ 3.10 อีกครั้งหนึ่ง โดยเรากำหนด damping ratio (ζ) = 0.7 เราจะได้

$$\theta = \cos^{-1} 0.7 = 45^\circ \quad (3.20)$$

เมื่อได้ค่ามุมแล้ว ก็ทำการลากเส้นที่ทำมุมกับแกนจริงลบ 45° ไปตัดเส้นทางของราก จุดตัดที่ได้นั้นก็คือตำแหน่งของโพลที่ทำให้ระบบมีค่า damping ratio (ζ) เท่ากับ 0.7 ซึ่งจะได้

$$\text{Percent maximum overshoot} = 100e^{-0.7\pi/\sqrt{1-0.7^2}} = 4.6\% \quad (3.21)$$

จากนั้นทำการหาค่า k ที่จุดตัดดังกล่าว ได้ k เท่ากับ 2631 เท่า โดยตำแหน่งของโพลอยู่ที่ $-12.493 \pm j12.745$, -8.232
จากสมการที่ 3.14

$$k = \frac{k_p k_d k_s k_m N}{\tau_1}$$

ในวงจรที่ออกแบบ จะออกแบบให้สามารถปรับค่า k_d ได้ ทำการแทนค่าหา k_d โดย $k_p = 0.398V/Rad$, $k_s = 9$, $k_m = 3.49Rad/Sec/V$, $N = 60$, $\tau_1 = 2$

$$k_d = \frac{2631 \times 2}{0.398 \times 9 \times 3.49 \times 60} = 7$$

จากค่าที่ออกแบบได้ $\tau_1 = 2$ $\tau_2 = 0.2$ นำค่านี้ไปกำหนดค่า R_1, R_2 และ C ของวงจร Low Pass Filter ในรูปที่ 3.2 โดย

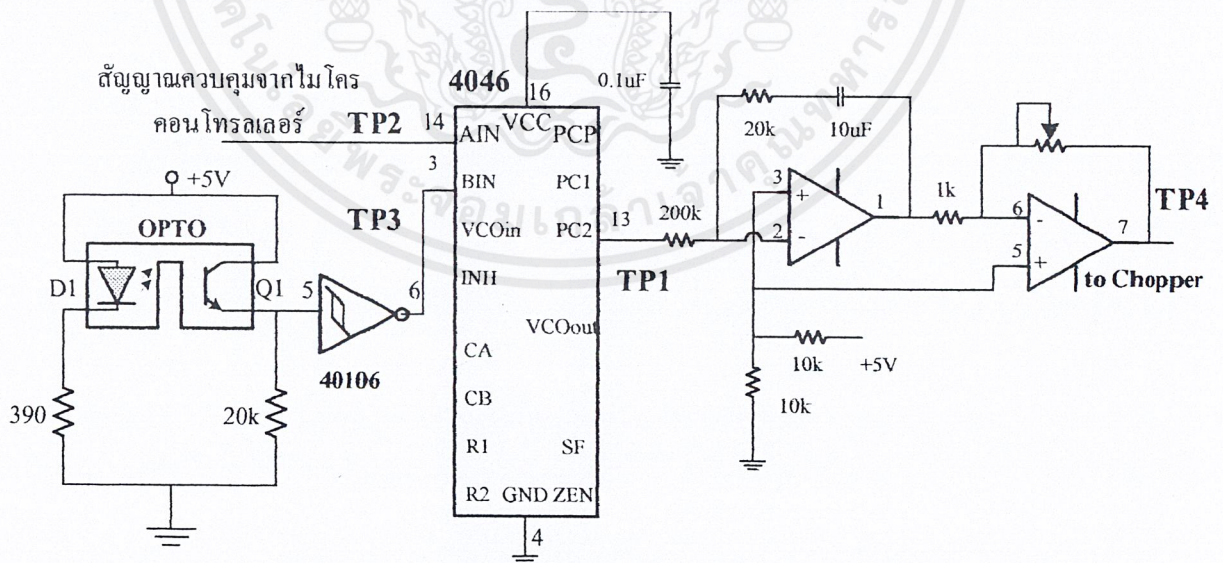
$$\tau_1 = R_2 C \quad (3.23)$$

$$\tau_2 = R_1 C \quad (3.24)$$

กำหนดค่า $R_2 = 200 k\Omega$ คำนวณค่า C จากสมการ (3.23) ได้เท่ากับ $10 \mu F$ นำค่า C ที่คำนวณได้แทนในสมการ (3.24) หาค่า R_1 ได้เท่ากับ $20 k\Omega$

4.3 แสดงวงจรควบคุมความเร็วมอเตอร์

วงจรควบคุมความเร็วมอเตอร์แบบสมบรูณ์แสดงดังรูปที่ 3.13 โดยสัญญาณควบคุมความเร็วจะถูกส่งมาจากไมโครคอนโทรลเลอร์ และเอาต์พุตของวงจรจะถูกต่อไปยังอินพุตของวงจร Chopper

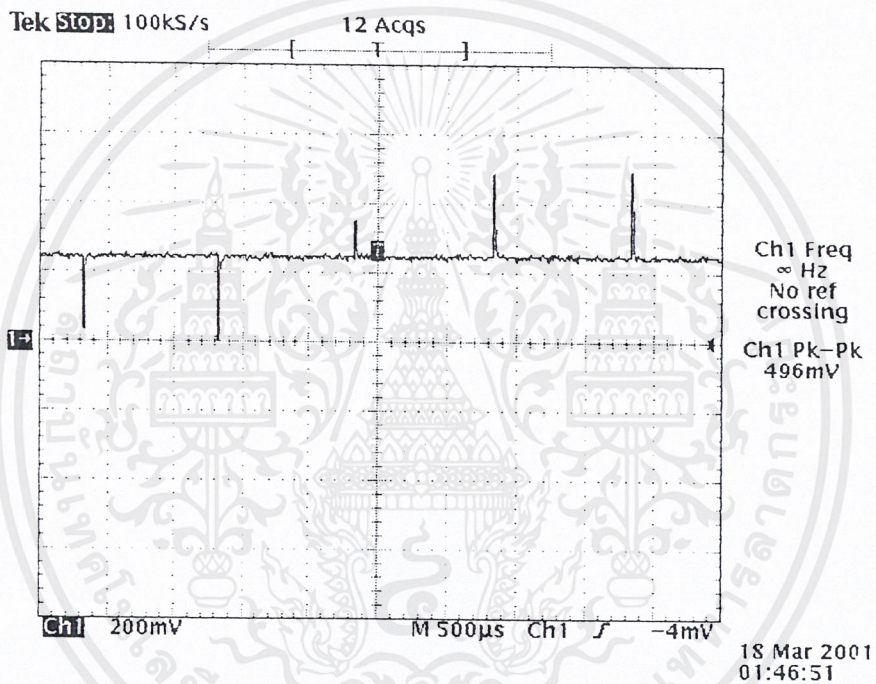


รูปที่ 3.13 แสดงวงจรควบคุมความเร็วมอเตอร์

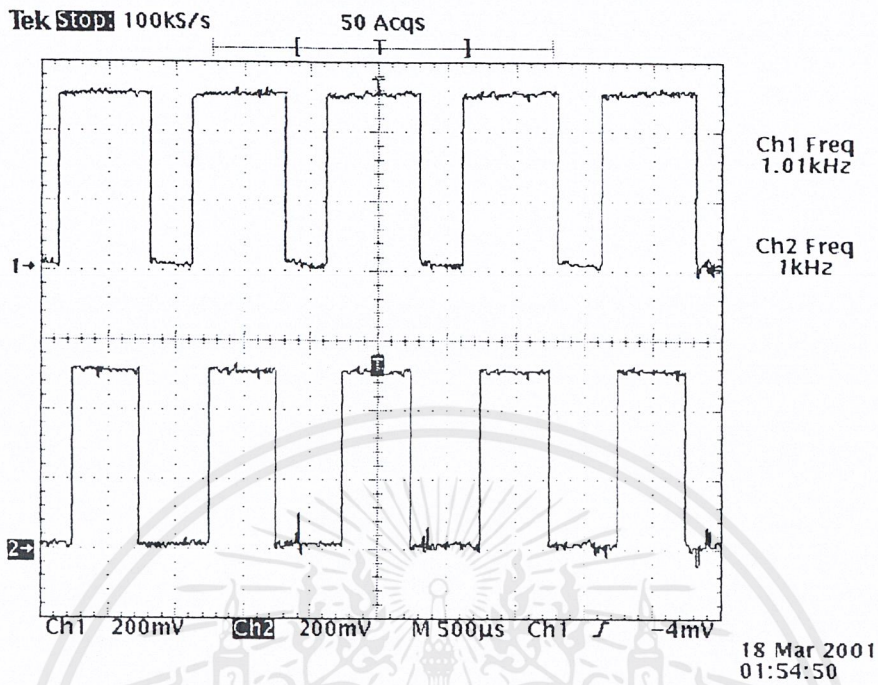
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การทดลองและผลการทดลอง

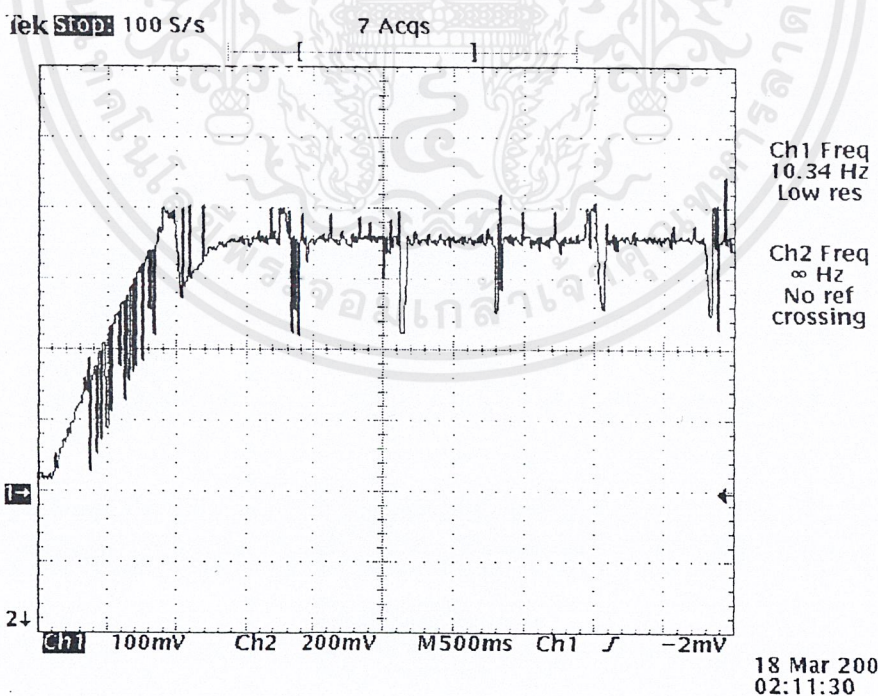
ทำการทดลองโดยการป้อนสัญญาณควบคุม 1KHz แล้ววัดสัญญาณเอาต์พุตของ Phase Comparator (TP1) ได้สัญญาณแสดงดังรูปที่ 3.14 และทำการเปรียบเทียบสัญญาณควบคุม (TP2) กับสัญญาณของเอ็น โคคเคอร์ (TP3) ได้สัญญาณแสดงดังรูปที่ 3.15 สำหรับรูปที่ 3.16 เป็นสัญญาณที่เอาต์พุตของวงจร (TP4) ในช่วงเริ่มต้นการทำงาน จากผลการทดลองรูปที่ 3.16 หาค่า Overshoot ของระบบได้ประมาณ 11 เปอร์เซ็นต์



รูปที่ 3.14 แสดงสัญญาณเอาต์พุตของเฟสคอมพาราเคอร์ (TP1) เมื่อป้อนสัญญาณควบคุม 1 KHz (วัด โดยใช้โพรบ $\times 10$)



รูปที่ 3.15 รูปบนแสดงสัญญาณจากเอ็น โคคเคอร์ (TP3) รูปล่างแสดงสัญญาณควบคุม (TP2) เมื่อป้อนสัญญาณควบคุม 1 KHz (วัดโดยใช้โพรบ $\times 10$)



รูปที่ 3.16 แสดงสัญญาณเอาท์พุท (TP4) ในช่วงเริ่มต้นการทำงานของมอเตอร์

เมื่อป้อนสัญญาณควบคุม 1 KHz (วัดโดยใช้โพรบ $\times 10$)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

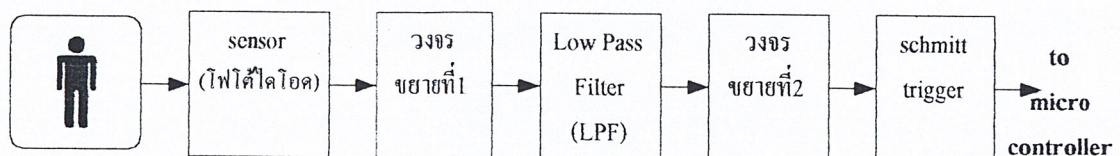
บทที่ 4

วงจรวัดอัตราการเต้นของหัวใจ

ในการวัดอัตราการเต้นของหัวใจ ใช้หลักการตรวจจับการเปลี่ยนแปลงความหนาแน่นของเลือดที่ปลายนิ้ว ซึ่งจะเปลี่ยนแปลงตามจังหวะการสูบฉีดเลือดของหัวใจ โดยการใช้แสงส่งผ่านเข้าไปในเส้นเลือดและใช้โฟโตไดโอดเป็นตัวรับแสง ซึ่งการวางตำแหน่งของตัว Sensor ทำได้ 2 แบบ ดังรูปที่ 4.1 เมื่อความหนาแน่นของเลือดเปลี่ยนแปลงจะทำให้ความเข้มของแสงเปลี่ยนแปลงด้วย ส่งผลให้กระแสที่ไหลผ่านโฟโตไดโอดเกิดการเปลี่ยนแปลง และเกิดการกระเพื่อมของสัญญาณตามการเต้นของหัวใจ บล็อกการทำงานของวงจรแสดงในรูปที่ 4.2 โดยสัญญาณจาก sensor ซึ่งเป็นโฟโตไดโอด จะถูกส่งเข้าไปขยายยังวงจรขยายที่ 1 จากนั้นจะถูกกรองความถี่สูงออกให้เหลือเฉพาะสัญญาณที่เกิดจากการเต้นของหัวใจด้วย Low Pass Filter (LPF) สัญญาณที่ผ่านการกรองแล้วจะถูกส่งไปยังวงจรขยายที่ 2 เพื่อขยายให้สัญญาณแรงขึ้น แล้วส่งต่อไปยังวงจร Schmitt trigger (Schmitt trigger) เพื่อทำให้เป็นสัญญาณสี่เหลี่ยม



รูปที่ 4.1 แสดงการวางตำแหน่ง ตัวกำเนิดแสง ตัวรับแสง แบบส่งผ่านและสะท้อนกลับ



รูปที่ 4.2 บล็อกการทำงานของวงจรวัดการเต้นของหัวใจ

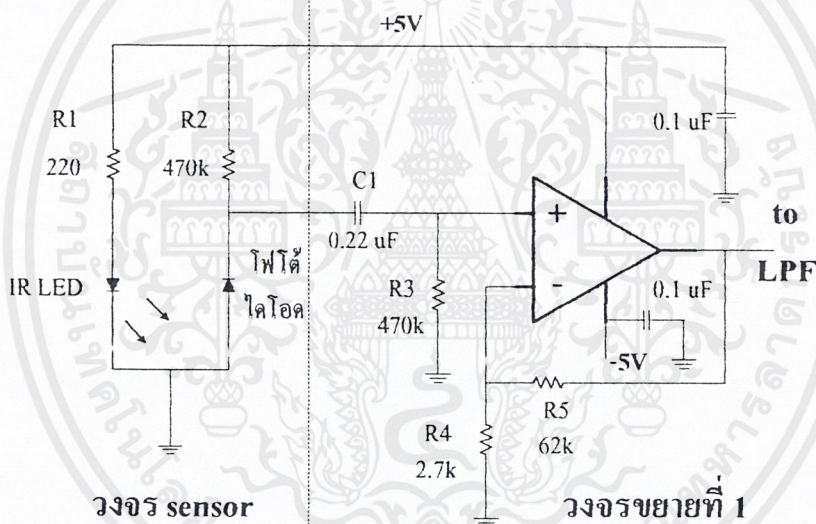
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 การออกแบบ

4.1.1 วงจร sensor และวงจรขยายที่ 1

พิจารณาวงจรรูปที่ 4.3 โฟโตไดโอดทำหน้าที่รับแสง ซึ่งแสงถูกส่งผ่านเข้าไปในเส้นเลือด เมื่อความหนาแน่นของเลือดเปลี่ยนแปลงจะทำให้ความเข้มของแสงเปลี่ยนแปลงด้วย เมื่อแสงที่มากกระทบโฟโตไดโอดเปลี่ยนแปลง ทำให้เกิดการกระเพื่อมของสัญญาณขึ้น สัญญาณที่ได้จะถูกส่งเข้าไปขยายยังวงจรขยายที่ 1 โดย C1 และ R3 ทำหน้าที่เป็น High Pass Filter เพื่อป้องกันไฟ DC ไม่ให้ผ่านเข้าไปยังวงจรขยาย ซึ่งมีความถี่คutoffอยู่ที่ 1.5 Hz ซึ่ง R4 และ R5 เป็นตัวกำหนดอัตราขยายของวงจรโดย

$$\text{อัตราการขยายของวงจรขยายที่ 1} = \frac{R_5}{R_4} = \frac{62k}{2.7k} = 23 \text{ เท่า}$$



รูปที่ 4.3 วงจร sensor และวงจรขยายภาคที่ 1

4.1.2 Low Pass Filter (LPF)

วงจร Low Pass Filter เป็นวงจรที่ทำหน้าที่กรองความถี่สูงออกไป เพื่อให้เหลือเฉพาะความถี่ต่ำที่เป็นความถี่ซึ่งเกิดจากการเต้นของหัวใจ โดย Filter ที่เลือกใช้เป็น Butterworth Filter ใช้ order 4 จากตาราง Normalized Butterworth LPF Loss Function ที่ n เท่ากับ 4 นำมาเขียน Transfer Function ของ LPF order 4 ความถี่คutoff (ω_c) อยู่ที่ 1 Rad./Sec. ได้เป็น

$$T(S) = \frac{k}{(S^2 + 0.76537S + 1)(S^2 + 1.84776S + 1)} \quad (4.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

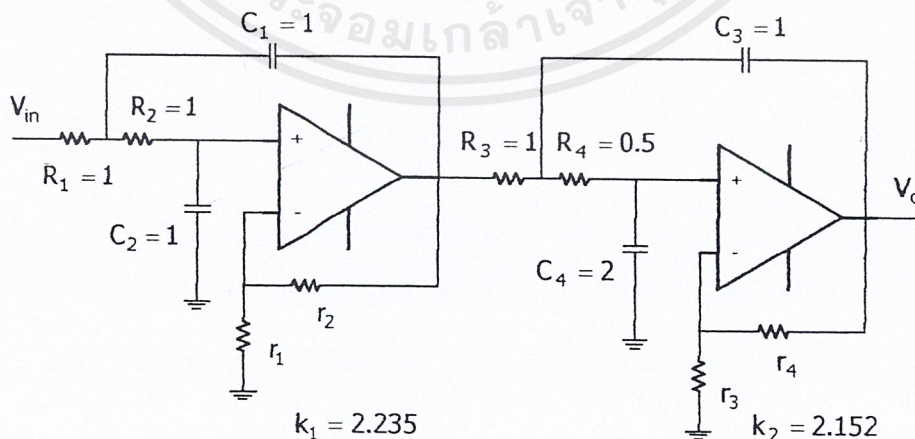
ตารางที่ 4.1 Normalizer Butterworth FPF Loss Function

n	Normalized Function ($\omega_p = 1$)
1	$(S + 1)$
2	$(S^2 + 1.414S + 1)$
3	$(S + 1)(S^2 + S + 1)$
4	$(S^2 + 0.76537S + 1)(S^2 + 1.84776S + 1)$
5	$(S + 1)(S^2 + 0.61803S + 1)(S^2 + 1.61803S + 1)$

การออกแบบวงจร ทำการออกแบบโดยวิธีเทียบสัมประสิทธิ์ วงจรที่ใช้แสดงดังรูปที่ 4.4 ซึ่งหา Transfer Function ได้เป็น

$$T(S) = \frac{V_o(S)}{V_{in}(S)} = \frac{k_1/R_1R_2C_1C_2}{S^2 + \left(\frac{1}{R_1C_2} + \frac{1}{R_2C_1} + \frac{1-k_1}{R_2C_2}\right)S + \frac{1}{R_1R_2C_1C_2}} \cdot \frac{k_2/R_3R_4C_3C_4}{S^2 + \left(\frac{1}{R_3C_3} + \frac{1}{R_4C_3} + \frac{1-k_2}{R_4C_4}\right)S + \frac{1}{R_3R_4C_3C_4}} \quad (4.2)$$

โดย $k_1 = \frac{r_1 + r_2}{r_1}$; $k_2 = \frac{r_3 + r_4}{r_3}$



รูปที่ 4.4 Normalized Forth Order LPF Circuit ($\omega_c = 1$ Rad./Sec.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการเทียบสัมประสิทธิ์ระหว่างสมการที่ (4.1) และ (4.2) จะได้

$$\frac{1}{R_1 R_2 C_1 C_2} = 1 \quad (4.3)$$

$$\frac{1}{R_1 C_2} + \frac{1}{R_2 C_1} + \frac{1+k_1}{R_2 C_2} = 0.76537 \quad (4.4)$$

$$\frac{1}{R_3 R_4 C_3 C_4} = 1 \quad (4.5)$$

$$\frac{1}{R_3 C_3} + \frac{1}{R_4 C_3} + \frac{1+k_2}{R_4 C_4} = 1.84776 \quad (4.6)$$

จากสมการที่ (4.3) กำหนด $R_1 = R_2 = C_1 = C_2 = 1$ นำค่า R_1, R_2, C_1, C_2 แทนลงในสมการที่ (4.4) เพื่อหาค่า k_1 จะได้

$$k_1 = 3 - 0.76537 = 2.235$$

กำหนด $C_3 = R_3 = 1, C_4 = 2$ แทนในสมการที่ (4.5) หา R_4 ได้เท่ากับ 0.5 นำค่า R_3, R_4, C_3, C_4 แทนในสมการที่ (4.6) เพื่อหาค่า k_2 จะได้

$$k_2 = 4 - 1.84776 = 2.152$$

นำค่า R และ C ที่คำนวณได้ไว้ในวงจรดังแสดงในรูปที่ 4.4 จะได้อัตรา Normalized มีความถี่คัทออฟ (ω_c) อยู่ที่ 1 Rad/Sec. ค่า R และ C ที่คำนวณได้ ซึ่งแสดงในวงจรรูปที่ 4.4 ยังไม่สามารถใช้งานได้ในทางปฏิบัติ จะต้องนำไปทำการ scaling ก่อน โดยจะทำการ scaling ทั้งความถี่และอิมพีแดนซ์ โดยกำหนดความถี่คัทออฟ (ω_c) เท่ากับ 5 Hz และกำหนด C_1, C_2, C_3 ซึ่งมีค่าเท่ากัน ให้มีค่าเท่ากับ 0.1 μ F. ดังนั้นเราจะได้

$$k_f = \frac{\omega_c'}{\omega_c} = \frac{2\pi(5)}{1} = 31.4$$

โดย ω_c' : ความถี่คัทออฟใหม่ที่ผ่านการ scaling แล้ว
 ω_c : ความถี่คัทออฟเดิม
 k_f : ค่าในการ scale ความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$k_i = \frac{C}{k_f C'} = \frac{1}{31.4 \times 0.1 \times 10^{-6}} = 318.5 \times 10^3$$

โดย C' : ค่าคาปาซิเตอร์ใหม่ ที่ผ่านการ scaling แล้ว
 C : ค่าคาปาซิเตอร์เดิม
 k_i : ค่าในการ scale อิมพีแดนซ์

$$R' = k_i R \quad (4.7)$$

โดย R' : ค่ารีซิสแตนซ์ใหม่ ที่ผ่านการ scaling แล้ว
 R : ค่ารีซิสแตนซ์เดิม

ดังนั้นจากสมการที่ 4.7 เราจะได้ $R_1 = R_2 = R_3 = 318.5 \text{ k}\Omega, R_4 = 159.2 \text{ k}\Omega$

$$C' = \frac{C}{k_f k_i} \quad (4.8)$$

จากสมการที่ (4.8) จะได้

$$C_4 = \frac{2}{(31.4)(318.5 \times 10^3)} = 0.2 \text{ }\mu\text{F.}$$

สำหรับค่า r_1, r_2, r_3 และ r_4 เป็นตัวกำหนดค่าเกณฑ์ของวงจร โดย

$$k_1 = \frac{r_2}{r_1} + 1 \quad (4.9)$$

$$k_2 = \frac{r_4}{r_3} + 1 \quad (4.10)$$

กำหนดค่า r_2 เท่ากับ $140 \text{ k}\Omega$ หาค่า r_1 จากสมการ (4.9) จะได้

$$r_1 = \frac{r_2}{k_1 - 1} = \frac{140 \times 10^3}{2.235 - 1} = 113 \text{ k}\Omega$$

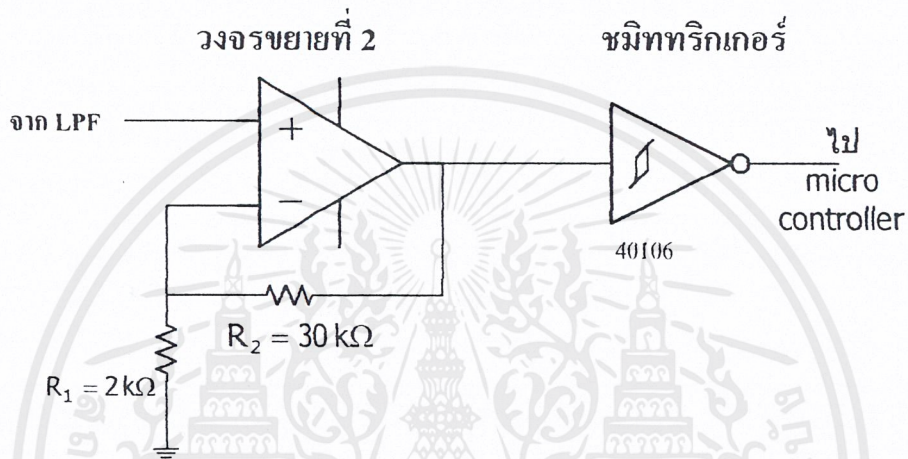
กำหนดค่า r_3 เท่ากับ $34 \text{ k}\Omega$ หาค่า r_4 จากสมการ (4.10) จะได้

$$r_4 = (k_2 - 1)r_3 = (2.152 - 1) \times 34 \times 10^3 = 39.2 \text{ k}\Omega$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 วงจรขยายที่ 2 และขมิททริกเกอร์

วงจรขยายที่ 2 เป็นวงจรที่ทำหน้าที่ขยายสัญญาณที่ผ่านการกรองแล้วจากวงจร Low Pass Filter (LPF) ให้มีขนาดสูงขึ้นก่อนจะส่งเข้าขมิททริกเกอร์ เพื่อทำให้เป็นสัญญาณรูปสี่เหลี่ยม การที่ต้องทำให้เป็นสัญญาณรูปสี่เหลี่ยมเนื่องจากเราต้องการใช้ ไมโครคอนโทรลเลอร์ ในการนับคาบเวลาของสัญญาณที่เข้ามา ดังนั้นสัญญาณจะต้องอยู่ในลักษณะ High และ Low



รูปที่ 4.5 วงจรขยายที่ 2 และขมิททริกเกอร์

พิจารณาวงจรในรูป 4.5 วงจรขยายที่ 2 ออกแบบโดยใช้ Op-Amp เป็นวงจรขยายแบบไม่กลับเฟส โดยอัตราขยายเป็นไปตามสมการ

$$\frac{V_{\text{out}}}{V_{\text{in}}} = \frac{R_2}{R_1} + 1$$

ดังนั้นจากวงจรที่ออกแบบมีอัตราขยาย (AV)

$$AV = \frac{V_{\text{out}}}{V_{\text{in}}} = \frac{30\text{ k}\Omega}{2\text{ k}\Omega} + 1 = 16 \text{ เท่า}$$

สำหรับขมิททริกเกอร์ใช้ IC NOT Gate Schmitt trigger เบอร์ 40106

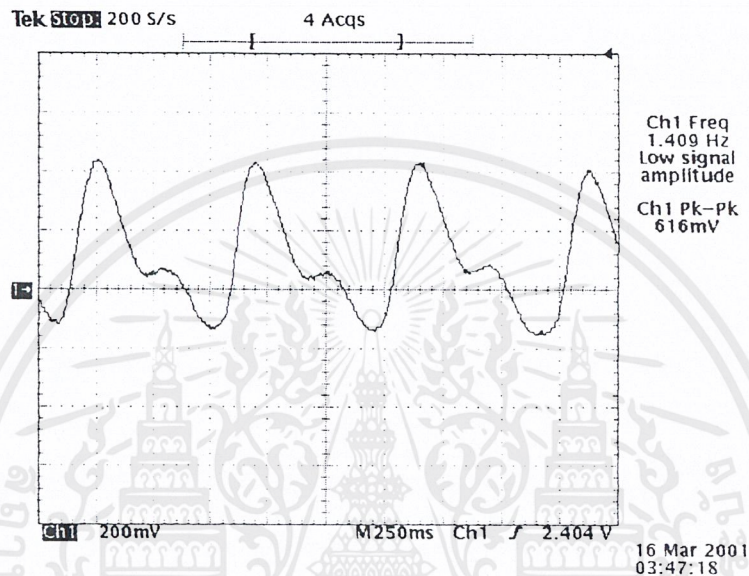
4.2 วงจรวัดอัตราการเต้นของหัวใจ

วงจรสมบรูณ์ของวงจรวัดอัตราการเต้นของหัวใจแสดงดังรูปในภาคผนวกที่ ข.1 สัญญาณเอาท์พุทของ วงจรจะถูกต่อ ไปยังไมโครคอนโทรลเลอร์ เพื่อทำการวัดอัตราการเต้นของหัวใจ

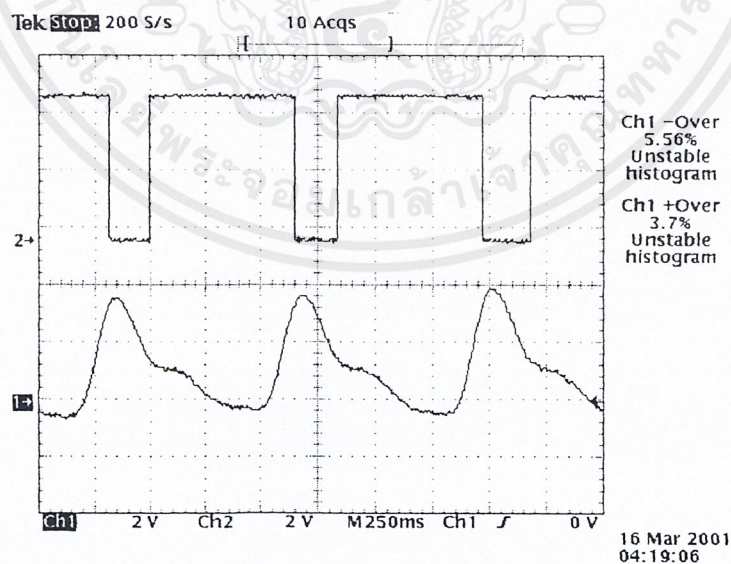
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองและผลการทดลอง

จากวงจรวัดอัตราการเต้นของหัวใจ รูปในภาคผนวกที่ ได้ทำการทดลองและวัดสัญญาณที่จุดต่างๆ ได้แก่ เอาท์พุทของ LPF (TP1), เอาท์พุทของวงจรขยายที่ 2 (TP2) และเอาท์พุทของซิมิทริกเกอร์ (TP3) ได้รูปของสัญญาณ ดังแสดงในรูปที่ 4.6 และ 4.7



รูปที่ 4.6 แสดงสัญญาณเอาท์พุทของ Low Pass Filter (TP1)



รูปที่ 4.7 แสดงสัญญาณเอาท์พุทของซิมิทริกเกอร์ (รูปบน)
และเอาท์พุทของวงจรขยายที่ 2 (รูปล่าง)

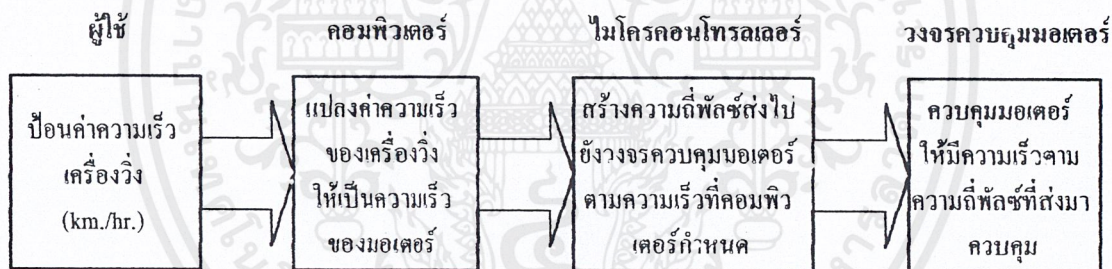
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ภาคควบคุมและแสดงผล

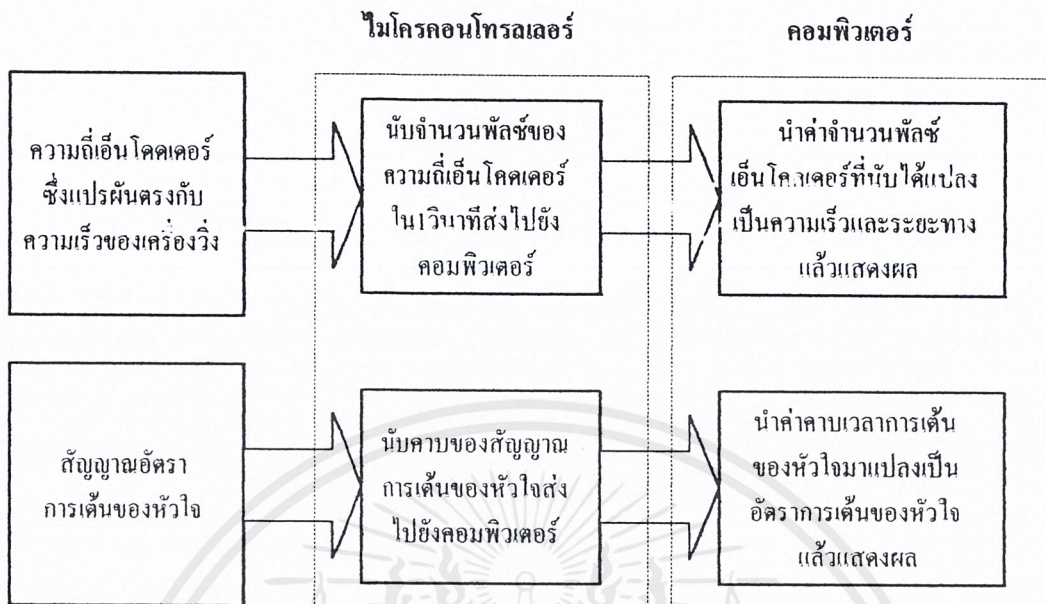
5.1 การควบคุมและการแสดงผล

ในการควบคุมและการแสดงผลเราได้ใช้คอมพิวเตอร์ทำงานร่วมกับไมโครคอนโทรลเลอร์ MCS51 โดยคอมพิวเตอร์และไมโครคอนโทรลเลอร์จะติดต่อกันผ่านทางพอร์ทอนุกรม RS232 ใน ส่วนของการควบคุมก็คือการควบคุมความเร็วของมอเตอร์ คอมพิวเตอร์จะทำหน้าที่รับค่าความเร็ว จากการวิ่งของผู้ใช้ ซึ่งมีหน่วยเป็นกิโลเมตรต่อชั่วโมง มาคำนวณให้เป็นความเร็วของมอเตอร์ จากนั้นจะส่งค่าความเร็วของมอเตอร์ที่คำนวณได้ไปยังไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์ได้รับค่าความเร็วของมอเตอร์ จะทำการสร้างสัญญาณความถี่พัลส์ส่งไปยังวงจรควบคุมความเร็วมอเตอร์ เพื่อควบคุมให้มอเตอร์มีความเร็วตามที่ต้องการ (ดูรูปที่ 5.1 ประกอบ)



รูปที่ 5.1 แสดงการทำงานของระบบเพื่อควบคุมความเร็วมอเตอร์

สำหรับการแสดงผลซึ่งประกอบด้วย การแสดงผลอัตราการเดินของหัวใจ ความเร็วของ เครื่องวิ่ง และระยะทางที่วิ่ง ไมโครคอนโทรลเลอร์จะทำหน้าที่วัดคาบและแสดงสัญญาณการเดิน ของหัวใจในหนึ่งลูกคลื่น แล้วส่งไปยังคอมพิวเตอร์ เมื่อคอมพิวเตอร์ได้รับจะทำการแปลงคาบเวลา ที่ได้เป็นอัตราการเดินของหัวใจ แล้วแสดงผลที่หน้าจอ ในส่วนของการวัดความเร็วและระยะทางที่ วิ่ง จะใช้หลักการวัดความเร็วเฉลี่ยใน 1 วินาที โดยทำการนับลูกคลื่นจากความถี่เซ็นเซอร์ ซึ่ง แปรผันตรงกับความเร็วของเครื่องวิ่งใน 1 วินาที แล้วส่งไปยังคอมพิวเตอร์ เพื่อให้คอมพิวเตอร์ คำนวณเป็นความเร็วของเครื่องวิ่งในหน่วยของกิโลเมตร และระยะทางที่วิ่งในหน่วยของเมตร



รูปที่ 5.2 แสดงการวัดอัตรการเดินของหัวใจ , ความเร็วของเครื่องวึ่ง และระยะทางที่วึ่ง

5.2 วงจรควบคุมโดยใช้ไมโครคอนโทรลเลอร์ตระกูล 89C2051

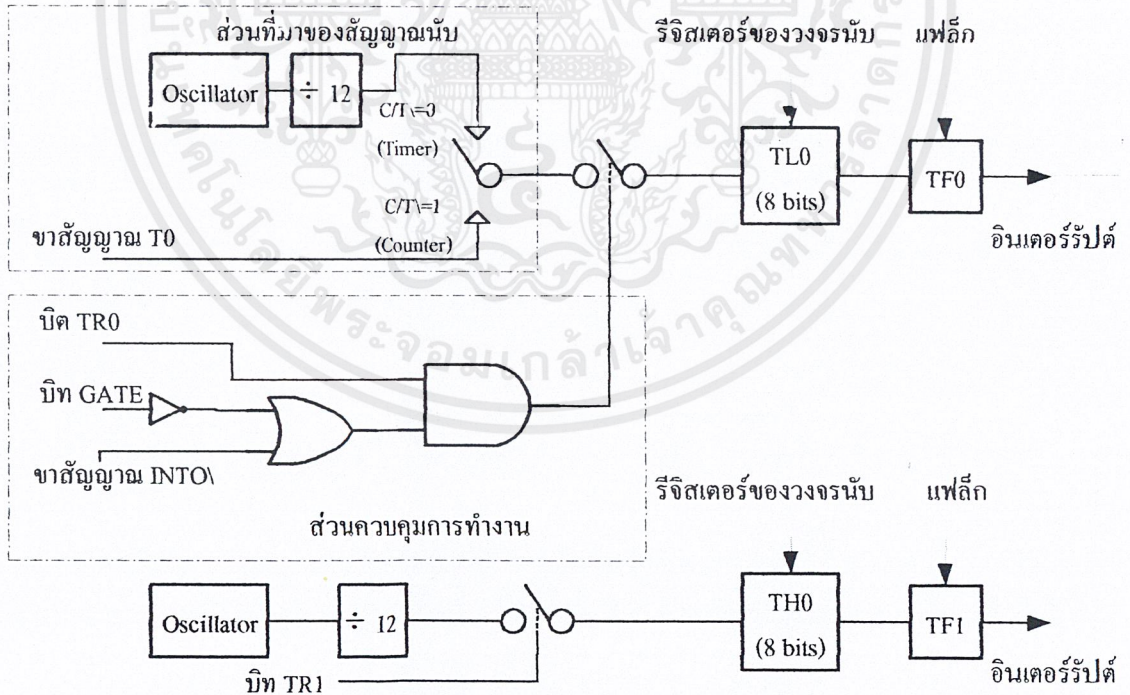
วงจรควบคุมออกแบบโดยใช้ไมโครคอนโทรลเลอร์ตระกูล MCS51 เบอร์ 89C2051 แสดงดังรูปในภาคผนวกที่ ข.2 สัญญาณการเดินของหัวใจและความถี่เอ็นโคเดอร์ถูกต่อกับขา INTO และ INT1 ตามลำดับ การต่อลักษณะนี้ก็เพื่อที่จะเขียนโปรแกรมการวัดในลักษณะของการอินเตอร์รัปต์ SW1 เป็นสวิทช์ควบคุมภายนอกเพื่อสั่งงานให้เครื่องวึ่งทำงานและหยุดทำงาน ส่วน LED1 และ LED2 ใช้แสดงผลสภาวะการทำงานและหยุดทำงานของเครื่องวึ่ง ขาคอลเลคเตอร์ของ Q1 จะถูกต่อไปยังขาอินพุทของวงจร Shopper เพื่อควบคุมการ start และ stop ของมอเตอร์โดยเมื่อต้องการจะหยุดมอเตอร์ก็จะส่งค่า High ไปยัง P1.0 เพื่อให้ Q1 ทำงานช้อตวงจร Shopper ลงกราวนด์

5.3 การทำงานของโปรแกรม

การออกแบบโปรแกรมในการวัดคาบเวลาสัญญาณการเดินของหัวใจ การนับจำนวนพัลส์ของความถี่เอ็นโคเดอร์ การสร้างความถี่พัลส์เพื่อควบคุมความเร็วของมอเตอร์ และการรับค่าจากคอมพิวเตอร์ จะใช้หลักการของการอินเทอร์รัปต์ทั้งหมด การออกแบบโปรแกรมในลักษณะนี้ จะทำให้ไมโครคอนโทรลเลอร์ทำงานได้เร็ว เนื่องจากไม่ต้องเสียเวลาในการตรวจเช็คทางฮาร์ดแวร์

ภายในไมโครคอนโทรลเลอร์ 89C2051 มี Timer 16 บิต 2 ตัว คือ Timer 0 และ Timer 1 โดยจะใช้ Timer 1 ในการกำหนดอัตราบอร์คในการรับส่งทางพอร์ทอนุกรม ดังนั้นจะเหลือ Timer 0 เพียงตัวเดียว ซึ่งจะใช้เป็นฐานเวลาในการนับคาบเวลาสัญญาณการเดินของหัวใจ คั้งเวลา 1 วินาที เป็นเวลาในการนับพัลส์ และเป็นฐานเวลาในการสร้างความถี่พัลส์ส่งไปควบคุมมอเตอร์

เราทำการกำหนดให้ Timer 0 ทำงานใน Mode 3 ซึ่งจะทำให้ Timer 0 แยกเป็น Timer 8 บิต 2 ตัว ที่ทำงานอิสระต่อกันดังแสดงในรูปที่ 5.3 โดยเมื่อ Timer TL0 เกิด Over Flow จะทำให้อินเทอร์รัปต์แฟล็ก TF0 เซ็ต และถ้าหากมีการ อินาเมิลอินเทอร์รัปต์ โปรแกรมจะกระโดดไปยังตำแหน่ง 000BH ของหน่วยความจำโปรแกรม เช่นเดียวกับ Timer TH0 ถ้าเกิด Over Flow ขึ้น จะทำให้ TF1 เซ็ต และโปรแกรมจะกระโดดไปทำยังตำแหน่ง 001BH ถ้าหากมีการ อินาเมิลอินเทอร์รัปต์



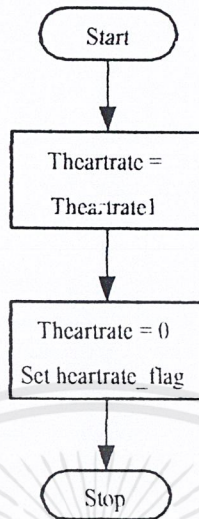
รูปที่ 5.3 วงจรนับ/จับเวลา 0 เมื่อทำงานในโหมด 3

5.3.1 การทำงานของโปรแกรมวัดคาบเวลาอัตราการเต้นของหัวใจ และนับพัลส์จากเอ็นโคดเดอร์

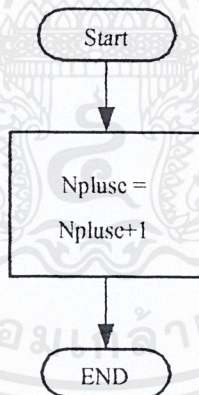
การทำงานของโปรแกรมแสดงคังโพล์วซาร์ทโปรแกรมบริการอินเทอร์รัปต์ รูปที่ 5.4 , 5.5 และ 5.6 โปรแกรมบริการอินเทอร์รัปต์ INTO จะทำงานเมื่อตรวจพบขอบขาของสัญญาณการเต้นของหัวใจ Thecartrate เป็นตัวแปรในการนับคาบเวลา จะเห็นว่าถูกเคลียร์เป็นศูนย์ ทุกขอบขาของสัญญาณการเต้นของหัวใจ และThecartrate จะถูกเพิ่มค่าขึ้นหนึ่ง ทุกๆครั้งที่ Timer TH0 นับเวลาได้ 256 uS. (ดูโพล์วซาร์ทโปรแกรมอินเทอร์รัปต์ Timer TH0 รูปที่ 5.6) คังนั้นเมื่อขอบขาของสัญญาณการเต้นของหัวใจปรากฏอีกครั้งหนึ่ง ค่าของ Thecartrate ก็คือค่าของสัญญาณที่นับได้ ในโปรแกรมบริการอินเทอร์รัปต์ INTO จะส่งค่า Thecartrate ไปเก็บยัง Thecartrate 1 และเคลียร์ Thecartrate เป็นศูนย์ เพื่อเริ่มต้นการนับใหม่ แล้ว Set heartrate_flag โดย heartrate_flag จะเป็น flag ที่แจ้งไปยังโปรแกรมหลัก ให้รู้ว่าคาบเวลาได้ถูกนับเสร็จแล้ว ซึ่งรายละเอียดของโปรแกรมหลักจะขออธิบายในหัวข้อต่อไป

โปรแกรมบริการอินเทอร์รัปต์ Timer TH0 จะทำงานทุกๆ 256 uS. ซึ่งถูกใช้เป็นฐานเวลาในการนับ การที่ใช้ฐานเวลานี้เนื่องจากว่า คริสตอลที่ใช้มีความถี่ 12 MHz เวลา 256 uS. Timer TH0 จะนับได้ 256 ครั้งพอดี คังนั้นค่าเริ่มต้นในการนับของ Timer TH0 ก็คือ 00H ทำให้ไม่จำเป็นต้องกำหนดค่าเริ่มต้นของ Timer TH0 ใหม่

นอกจาก Timer TH0 จะทำหน้าที่เป็นฐานเวลาในการนับคาบสัญญาณการเต้นของหัวใจแล้ว ยังทำหน้าที่จับเวลา 1 วินาที เพื่อใช้เป็นช่วงเวลาในการนับจำนวนพัลส์ของเอ็นโคดเดอร์ด้วย โดย Ntime เป็นตัวแปรในการนับ และจะครบ 1 วินาทีเมื่อ Ntime นับได้ 3906 ครั้ง ดูโพล์วซาร์ทบริการอินเทอร์รัปต์ Timer TH0 รูปที่ 5.6 เมื่อ Ntime นับครบ 3906 ครั้ง (เท่ากับ 1 วินาที) Ntime จะถูกเคลียร์เพื่อเริ่มนับใหม่ แล้วส่งค่า Npluse ไปเก็บยัง Npluse1 ค่าของ Npluse ก็คือจำนวนพัลส์จากเอ็นโคดเดอร์ที่นับได้ใน 1 วินาทีนั่นเอง (ดูโพล์วซาร์ทโปรแกรมบริการอินเทอร์รัปต์ INT1 รูปที่ 5.5 Npluse จะมีค่าเพิ่มขึ้นหนึ่งทุกๆครั้งที่ตรวจพบขอบขาของสัญญาณเอ็นโคดเดอร์) จากนั้นค่าของ Npluse จะถูกเคลียร์เป็นศูนย์ เพื่อเริ่มต้นนับใหม่ และบิต pluse_flag จะถูกเซท เพื่อแจ้งไปยังโปรแกรมหลักว่า จำนวนพัลส์ได้นับเสร็จแล้ว (รายละเอียดโปรแกรมหลักจะกล่าวไว้ในหัวข้อต่อไป)

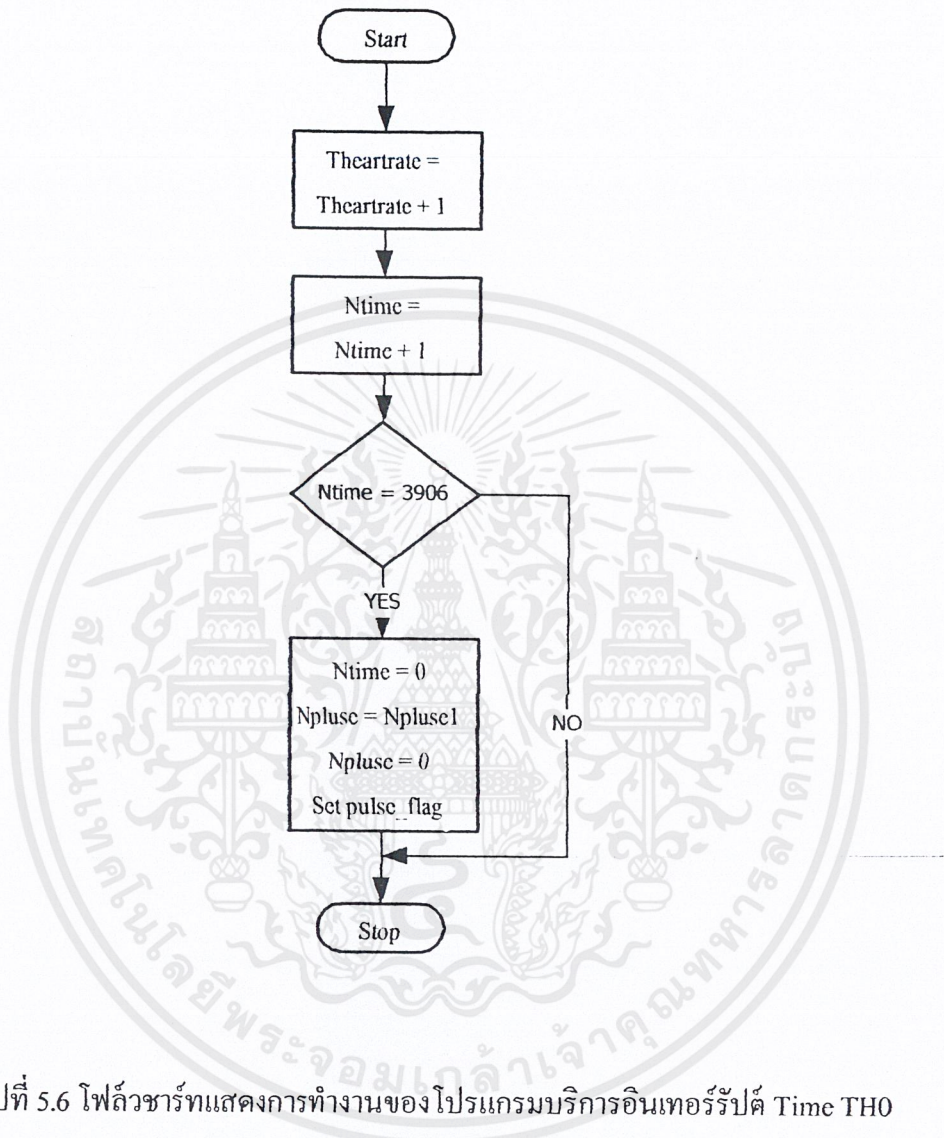


รูปที่ 5.4 โฟลิวชาร์ตแสดงการทำงานของโปรแกรมบริการอินเตอร์รัปต์ INT0
(ทำงานเมื่อตรวจพบขอบขาลงของสัญญาณการเค้นของหัวใจ)



รูปที่ 5.5 โฟลิวชาร์ตแสดงการทำงานของโปรแกรมบริการอินเตอร์รัปต์ INT1
(ทำงานเมื่อตรวจพบขอบขาลงของสัญญาณเอ็น โคคเคอร์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

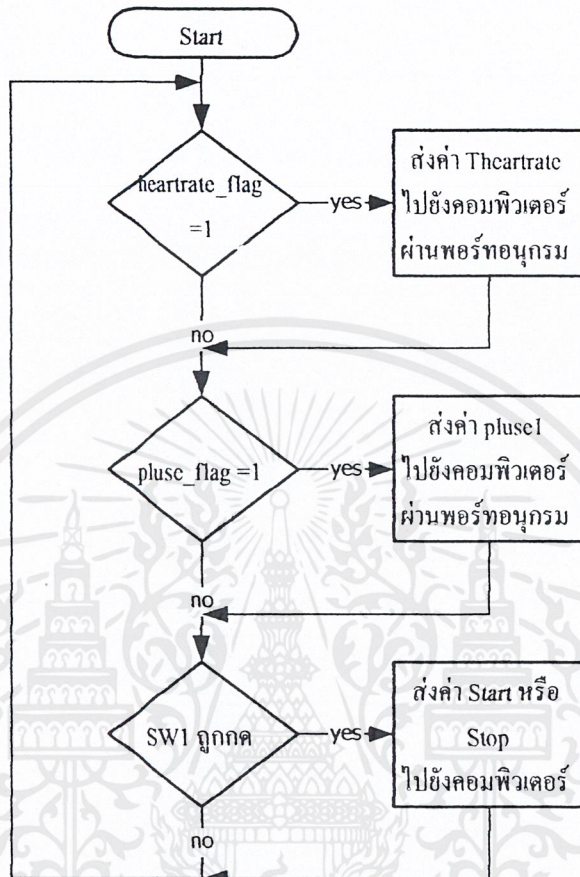


รูปที่ 5.6 ไฟล์ซาร์ทแสดงการทำงานของ โปรแกรมบริการอินเทอร์รัปต์ Time TH0
(ทำงานทุกๆเวลา 256 μ S.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.2 การทำงานของโปรแกรมหลัก

โปรแกรมหลักจะทำหน้าที่ในการส่งคาบเวลาของสัญญาณการเต้นของหัวใจ จำนวนพัลส์ เอ็น โคคเคอร์ที่นับได้ใน 1 วินาทีและส่งค่าควบคุมการทำงานและหยุดการทำงานของเครื่องวัด เมื่อมีการกดสวิทช์ (SW1) โดยการทำงานของโปรแกรมจะเริ่มขึ้นจากการตรวจสอบ heart rate flag (ดูรูป 5.7) ดังที่กล่าวไปแล้วว่า heart rate จะเป็น 1 เมื่อโปรแกรมบริการอินเทอร์เน็ต ได้รับคาบเวลาสัญญาณการเต้นของหัวใจ เสร็จแล้ว และจะส่งค่าการนับมาเก็บไว้ใน Theartrate 1 ดังนั้น เมื่อโปรแกรมหลักพบว่า heart rate เป็น "1" ก็จะส่งค่าของ Theartrate 1 ไปยังคอมพิวเตอร์ แล้วทำการเคลียร์ heart rate flag เพื่อให้ heart rate flag ถูกเซ็ตขึ้นใหม่อีกเมื่อคาบเวลาต่อไปนับเสร็จ จากนั้นโปรแกรมหลักจะไปตรวจสอบ plus flag ซึ่งถ้าเป็น "1" แสดงว่า การนับพัลส์สัญญาณ เอ็น โคคเคอร์เสร็จแล้ว และค่าจำนวนพัลส์ที่นับได้ถูกเก็บใน Nplus 1 โปรแกรมหลักก็จะทำการส่งค่า Nplus 1 ให้กับคอมพิวเตอร์แล้วทำการตรวจสอบการกดสวิทช์(SW1) ซึ่งเป็นสวิทช์ทำหน้าที่ 2 อย่าง คือ ควบคุมให้เครื่องทำงานและหยุดการทำงาน โดยเมื่อโปรแกรมตรวจพบว่ามีสวิทช์ SW1 ถูกกดจะทำการตรวจสอบสถานะของเครื่องขณะนั้นว่าอยู่ในสถานะใดแล้วจะส่งค่าไปควบคุมให้อยู่ในสถานะตรงข้าม เช่น ถ้าทำงานอยู่ก็จะส่งค่าไปควบคุมให้หยุดทำงาน



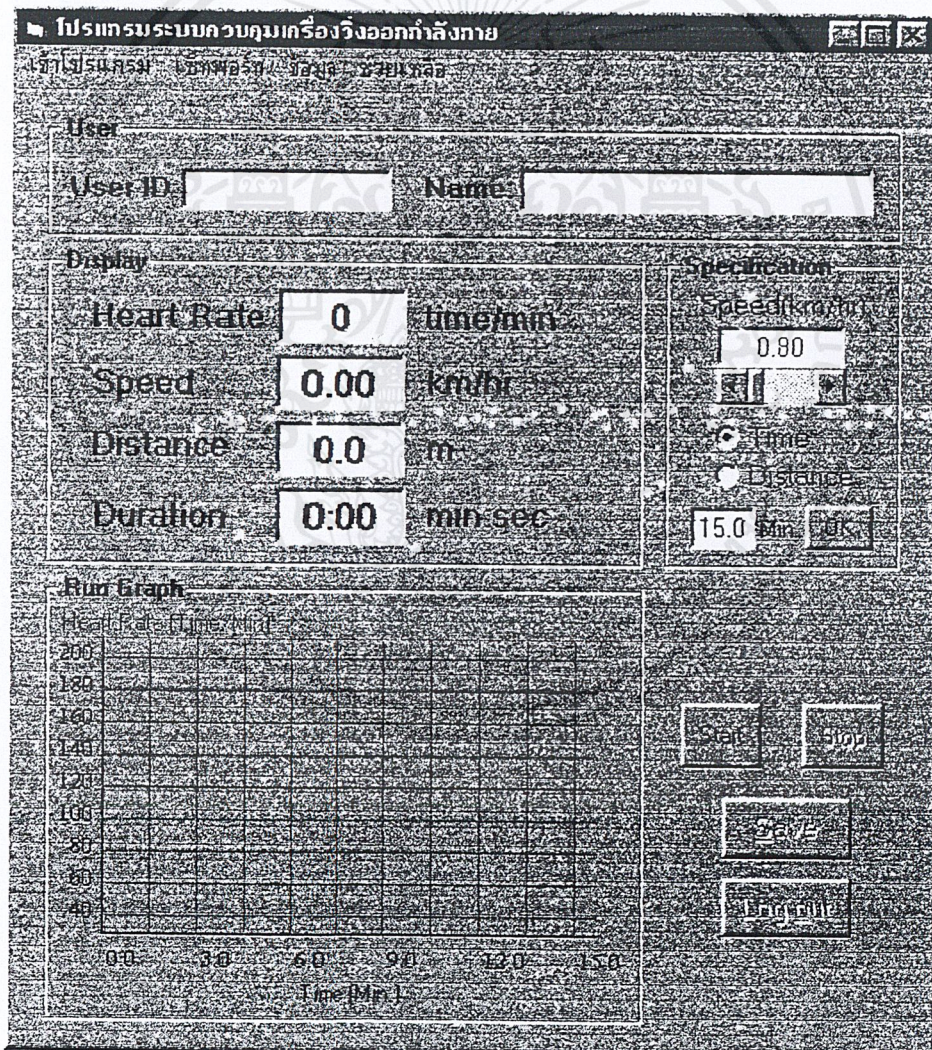
รูปที่ 5.7 ไฟล์ซาร์ทแสดงการทำงานของโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การใช้โปรแกรมคอมพิวเตอร์ในการควบคุมและแสดงผล

โปรแกรมที่เขียนขึ้น สามารถแสดงกราฟอัตราการเต้นของหัวใจ สามารถกำหนดความเร็วในการวิ่ง และเลือกกำหนดระยะทางหรือเวลาในการวิ่งได้ และมีฐานข้อมูลที่สามารถบันทึกเก็บข้อมูลประวัติของผู้ป่วยและสถิติการวิ่งของผู้ป่วยว่าวิ่งวันไหน เริ่มเวลาไหน ถึงเวลาไหน รวมทั้งบันทึกอัตราการเต้นของหัวใจ และเรียกดูได้ในรูปแบบของกราฟ มีหน้าจอหลักของโปรแกรมดังนี้



รูปที่ 6.1 หน้าจอหลักของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1 วิธีการใช้งานโปรแกรมในการวิงทดสอบ

6.1.1 เปิดเข้าโปรแกรม วิธีการเปิดเข้าโปรแกรม เราจะให้ผู้ปวยที่จะวิงมี 3 ประเภท คือ

1) ผู้ปวยใหม่ หมายถึง ผู้ปวยที่ยังไม่เคยได้ลงทะเบียนบันทึกประวัติไว้

เข้าโปรแกรมโดยเลือกเมนู “เข้าโปรแกรม” เมนูย่อย “ลงทะเบียน(ผู้ปวยใหม่)” จะปรากฏหน้าจอให้กรอกข้อมูลดังรูปที่ 6.2

รูปที่ 6.2 หน้าจอลงทะเบียนผู้ปวยใหม่

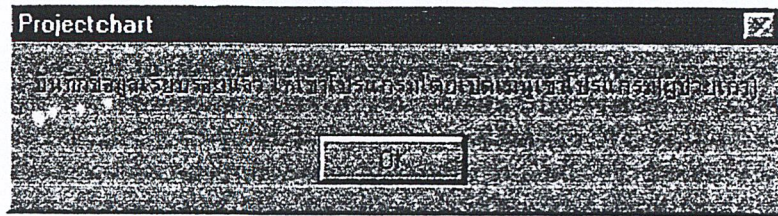
หากรหัสผู้ปวยที่เรากรอกซ้ำกับรหัสผู้ปวยเก่าที่มีอยู่แล้วก็จะปรากฏข้อความเตือนให้เรากรอกใหม่ดังรูปที่ 6.3



รูปที่ 6.3 ข้อความเตือนเมื่อกรอกรหัสผู้ปวยซ้ำกับผู้ปวยเก่า

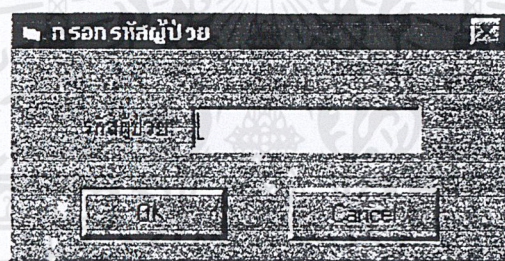
เมื่อบันทึกข้อมูลเรียบร้อยแล้ว นั่นคือผู้ปวยดังกล่าว จะเป็นผู้ปวยเก่าที่มีรหัสประจำตัวผู้ปวย และจะมีข้อความบอกว่า บันทึกข้อมูลแล้ว ดังรูปที่ 6.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า และไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 ข้อความแจ้งแก่ผู้ลงทะเบียนเมื่อบันทึกข้อมูลเรียบร้อยแล้ว

2) ผู้ป่วยเก่า หมายถึง ผู้ที่เคยลงทะเบียนหรือมีข้อมูลอยู่ในฐานข้อมูลแล้ว ให้เข้าโปรแกรม โดยเลือกเมนู “เข้าโปรแกรม” เมนูย่อย “เข้าโปรแกรม(ผู้ป่วยเก่า)” จะปรากฏหน้าจอให้กรอกรหัสผู้ป่วยดังรูปที่ 6.5



รูปที่ 6.5 หน้าจอเข้าโปรแกรม (ผู้ป่วยเก่า)

หากกรอกรหัสไม่ถูกต้อง จะมีข้อความบอกรหัสไม่ถูกต้องดังรูป และเมื่อเรากดปุ่ม “OK” ก็จะปรากฏหน้าจอให้เรากกรอกรหัสผู้ป่วยใหม่อีกครั้ง



รูปที่ 6.6 ข้อความแจ้งกรณีกรอกรหัสไม่ถูกต้อง

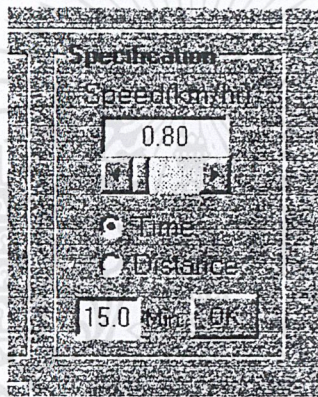
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกรอกรหัสถูกต้อง ก็จะเข้าสู่หน้าจอหลัก จะปรากฏชื่อและรหัสผู้ป่วย ตอนนี้เราจะสามารถใช้โปรแกรม และสามารถบันทึกข้อมูลได้

3) ผู้ป่วยที่วิ่งโดยไม่ต้องลงทะเบียน จะเปิดโปรแกรมปกติ แต่ไม่สามารถบันทึกข้อมูลได้ จะสังเกตเห็นว่าปุ่ม “Save” ไม่สามารถทำงานได้

6.1.2 กำหนดความเร็ว และกำหนดเวลาหรือระยะทางในการวิ่ง

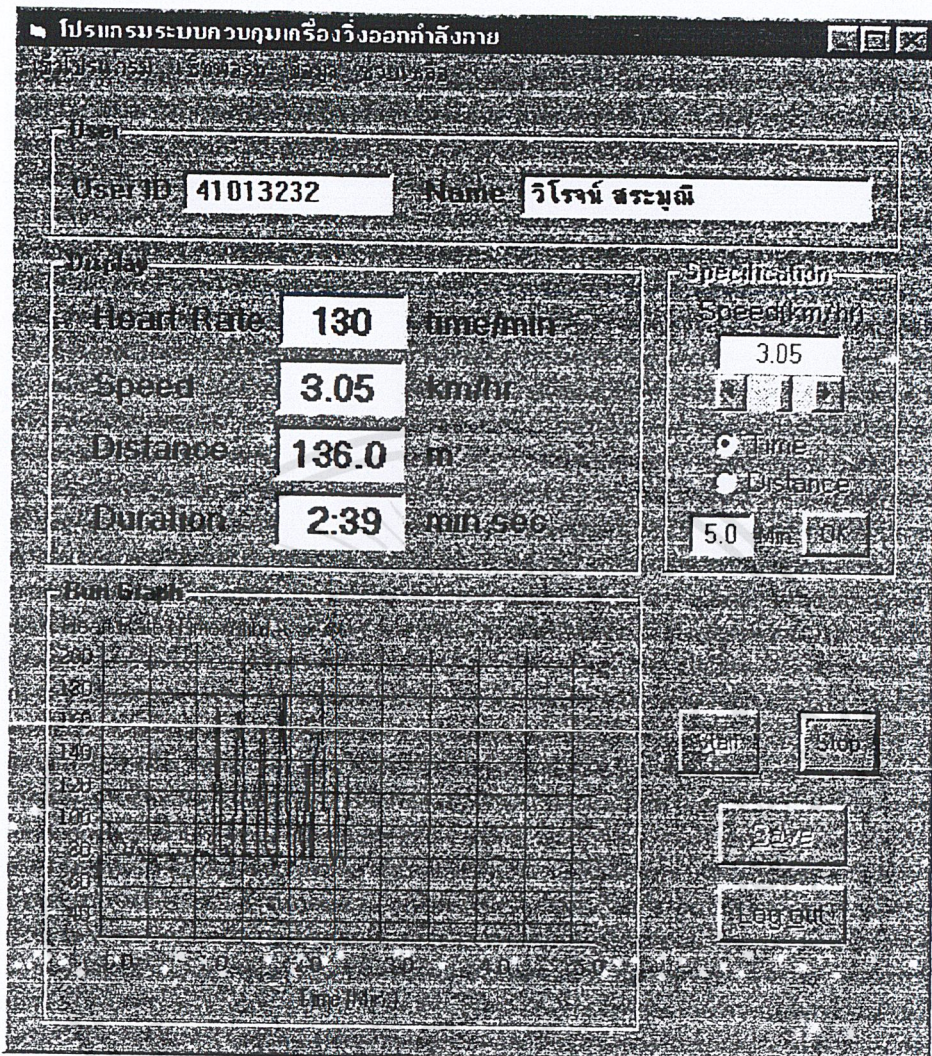
โดยกำหนดที่กรอบ Specification ดังรูปที่ 6.7



รูปที่ 6.7 กรอบกำหนดความเร็ว และเวลาหรือระยะทาง

6.1.3 เริ่มวิ่ง

ในกรณีที่จะวัดอัตราการเต้นของหัวใจพร้อมกันไปด้วย ทำได้โดยการกด Sensor เข้าที่ปลายนิ้ว ทำการ Turn ON ให้กับชุดจ่ายไฟ แล้วกดปุ่ม “Start” ในโปรแกรมหรือสวิทช์ “Start/Stop” ที่ชุดฮาร์ดแวร์ควบคุม ชุดเครื่องวิ่งก็จะเริ่มทำงาน ขณะที่กำลังวิ่งจะแสดงอัตราการเต้นหัวใจ ความเร็วในการวิ่ง ระยะทางที่วิ่งได้ เวลาที่ใช้ไปในการวิ่ง และแสดงกราฟอัตราการเต้นของหัวใจ ที่จอ monitor ดังรูปที่ 6.8



รูปที่ 6.8 กรอบแสดงผลขณะกำลังวิ่ง

จะสังเกตเห็นว่าขณะกำลังวิ่ง กรอบกำหนดค่าความเร็วและเมนูเข้าโปรแกรมจะไม่สามารถทำงานได้ จนกว่าจะกดปุ่ม “Stop” คือหยุดวิ่ง

6.1.4 เมื่อจะหยุดวิ่ง

ก็กดปุ่ม “Stop” และสามารถกดปุ่ม “Save” บันทึกข้อมูลได้ แต่กรณีที่เข้าโปรแกรมโดยไม่กรอกรหัสผู้ปวย จะไม่สามารถบันทึกข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 การดูข้อมูลการวิ่ง

ทำได้โดยเปิดเมนู “ข้อมูล” เมนูย่อย “ข้อมูลการวิ่งของผู้ป่วย” จะปรากฏหน้าจอบันทึกการวิ่งของผู้ป่วย ที่แสดงข้อมูลผู้ป่วยและตารางบันทึกสถิติการวิ่ง ดังรูป

The screenshot shows a software window titled "บันทึกการวิ่งของผู้ป่วย" (Patient Running Record). It contains the following elements:

- Form Fields:**
 - รหัสผู้ป่วย (Patient ID): 41013232
 - ชื่อ (Name): วิโรจน์
 - นามสกุล (Surname): สรรพวุฒิ
 - เพศ (Gender): ชาย (Male)
 - อายุ (Age): 24
 - ตำแหน่ง (Position): ปกติ (Normal)
- Buttons:**
 - ปุ่ม "Print" and "New Page" are located in a box on the right side.
 - Buttons "Show Graphs", "Refresh", and "Close" are at the bottom.
- Table:**

ตารางบันทึกการวิ่งของผู้ป่วย (Patient Running Record Table)

Date	Started	End Time
17/3/44	7:37:06	7:37:11
18/3/44	2:54:39	2:54:44
18/3/44	2:56:14	2:56:20
18/3/44	2:59:35	2:59:41
18/3/44	3:12:01	3:17:11
- Footer:** Record 5

รูปที่ 6.9 หน้าจอข้อมูลการวิ่งของผู้ป่วย

6.2.1 การดูกราฟ

จากหน้าจอรูปที่ 6.9 หากต้องการดูกราฟอัตราการเต้นหัวใจจากการวัดผลครั้งใด ก็ให้เลือกที่ตารางรายการตามวันและเวลาที่ต้องการดู แล้วกดปุ่ม “Show Graph” จะแสดงหน้าจอกราฟอัตราการเต้นของหัวใจดังรูปที่ 6.10



รูปที่ 6.10 หน้าจอแสดงกราฟอัตราการเต้นของหัวใจ

6.2.2 การแก้ไข หรือเพิ่มเติมข้อมูลประวัติของผู้ป่วย

ทำได้โดยให้เปิดเมนู “ข้อมูล” เมนูย่อย “เพิ่ม ลบ แก้ไข ค้นหาประวัติของผู้ป่วย” ก็จะปรากฏหน้าจอดังรูปที่ 6.11 และยังสามารถดูข้อมูลเฉพาะผู้ป่วยคนใดที่เราทราบชื่อหรือรหัสได้ โดยเรากำหนดรหัสผู้ป่วยหรือชื่อ

เพิ่ม ลบ แก้ไข ค้นหา ข้อมูลส่วนตัวของผู้ป่วย

รหัสผู้ป่วย: 41013232

ชื่อ: วิโรจน์

นามสกุล: สระมนี

เพศ: ชาย

อายุ: 24

ที่อยู่: ปกติ

Buttons: Add, Edit, Refresh, Update, Close

ค้นหาข้อมูล: Find Now, New Search

Record: 4

รูปที่ 6.11 หน้าจอ เพิ่มลบ แก้ไข ค้นหาข้อมูลของผู้ป่วย

6.2.3 การดูข้อมูลประวัติผู้ป่วยทั้งหมดในรูปแบบของตาราง

ทำได้โดยให้เปิดเมนู “ข้อมูล” เมนูย่อย “ประวัติส่วนตัวของผู้ป่วยทั้งหมด” ก็จะปรากฏหน้าจอดังรูปที่ 6.12

รหัสผู้ป่วย	ชื่อผู้ป่วย	ชื่อโรค	เพศ	อายุ	ชื่อแพทย์
41013156	กรรชิต	ขุนทอง	ชาย	24	เป็นโรคความดัน
41013228	วิเชียร	มาเรียน	ชาย	24	ปกติ
41013231	วิรัตน์	ทองจันทร์แก้ว	ชาย	24	อ. รัตภูมิ
41013232	วิโรจน์	สระมณี	ชาย	24	ปกติ
41013256	พงศ์ศักดิ์	สุนทร	ชาย	24	ปกติ
41013263	เมตรี	เพชรเกลี้ยง	ชาย	24	จ. พังงา
41013264	รณฤทธิ	รอตดำ	ชาย	24	จ. พัทลุง
41013266	วุฒิชัย	จอกทอง	ชาย	24	จ. นครศรีธรรมราช
41013276	อดิสิทธิ์	อภิวัดน์การญ	ชาย	24	ปกติ

รูปที่ 6.12 แสดงประวัติส่วนตัวของผู้ป่วยทั้งหมด

จะเห็นว่าโปรแกรมที่เขียนขึ้นนี้ จะใช้กับโรงพยาบาลหรือหมอที่ต้องการให้ผู้ป่วยวิ่งเพื่อบำบัดโรคหรือออกกำลังกาย โดยหมอจะบันทึกข้อมูลประวัติส่วนตัวของผู้ป่วยไว้ และจะให้ผู้ป่วยวิ่งและบันทึกสถิติ วันเวลา และกราฟอัตราการเต้นของหัวใจในการวิ่งแต่ละครั้งไว้เพื่อเปรียบเทียบการเปลี่ยนแปลงหรือพัฒนาการของผู้ป่วยได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

สรุป ปัญหา แนวทางแก้ไขและพัฒนา

จากการทดสอบการทำงานของเครื่องวิ่งออกกำลังกาย ระบบสามารถควบคุมความเร็วของเครื่องวิ่งได้ และสามารถปรับความเร็วในการวิ่งได้อยู่ในช่วง 0.5 กิโลเมตรต่อชั่วโมง ถึง 4.5 กิโลเมตรต่อชั่วโมง

ปัญหาที่พบก็คือในส่วนของวงจรวัดอัตราการเต้นของหัวใจ ซึ่งเมื่อตัว Sensor เกิดการเคลื่อนไหวขณะวิ่งจะทำให้เกิดสัญญาณรบกวนเข้ามาในวงจร ทำให้ไม่สามารถวัดอัตราการเต้นของหัวใจได้อย่างถูกต้องแน่นอน แต่พอจะสามารถมองภาพรวมหาค่าเฉลี่ยได้

จากปัญหาที่พบจะเห็นได้ว่า การวัดอัตราการเต้นของหัวใจด้วยวิธีการนี้ จะมีความไวต่อการเคลื่อนไหว ดังนั้นในการพัฒนาและปรับปรุง ควรใช้วิธีการวัดอัตราการเต้นของหัวใจวิธีอื่นที่การเคลื่อนไหวของร่างกายไม่มีผลต่อการทำงานของวงจร

ในส่วนของการเขียน โปรแกรมสามารถเพิ่มเติมขีดความสามารถต่อไปได้ เพื่อให้มีรูปแบบเป็นมาตรฐานสามารถใช้งานง่ายขึ้น

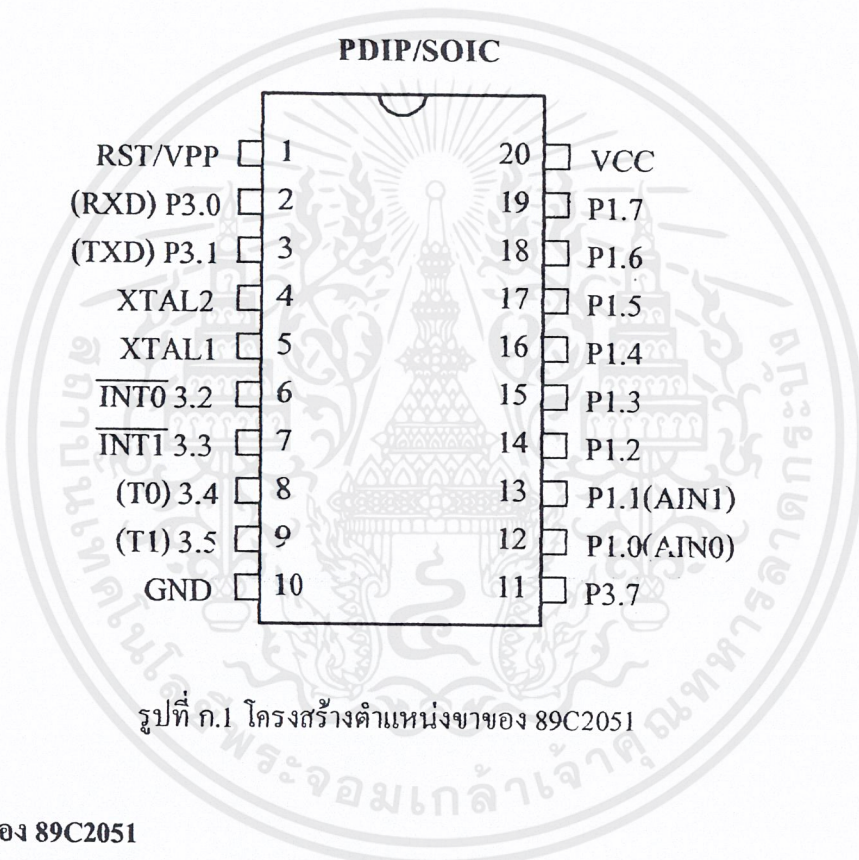


ภาคผนวก ก
ไมโครคอนโทรลเลอร์ 89C2051 และพอร์ตอนุกรม RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทฤษฎีไมโครคอนโทรลเลอร์ 89C2051 และพอร์ตอนุกรม RS232

ไมโครคอนโทรลเลอร์ที่ใช้ในโครงการนี้ เป็นไมโครคอนโทรลเลอร์ตระกูล MCS51 เบอร์ 89C2051 ซึ่ง 89C2051 เป็นไมโครคอนโทรลเลอร์ 8 บิต มีจำนวนขาทั้งหมด 20 ขา โดยมีโครงสร้างของตำแหน่งขาแสดงดังรูปที่ ก.1



1. คุณสมบัติของ 89C2051

- มีหน่วยความจำแบบแฟลช (Flash Memory) จำนวน 2 Kbyte ซึ่งสามารถโปรแกรมได้ 1000 ครั้ง
- ทำงานในช่วงแรงดันไฟเลี้ยง 2.7 V ถึง 6 V
- ทำงานที่ความถี่ออสซิลเลท 0 Hz ถึง 24 MHz
- มีแรม (RAM) ภายในจำนวน 128 byte
- มีพอร์ต อินพุท/เอาต์พุท 15 เส้น
- มี Timer/Counter 16 bit 2 ตัว
- รับสัญญาณอินเทอร์รัปต์ได้ 6 แหล่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มี Analog Comparator อยู่ภายใน
- มีโหมดประหยัดการทำงาน 2 โหมด คือ Low power Idle Mode และ Power down Mode

2. หน้าที่ของขาต่างๆ

- Vcc ขาจ่ายไฟเลี้ยง
- Gnd กราวนด์
- Port 1 เป็นพอร์ต 2 ทิศทาง (I/O Port) ขนาด 8 บิต P1.2 ถึง P1.7 มีการพูลอัพภายใน (internal pull ups) P1.0 และ P1.1 ไม่มีพูลอัพภายใน P1.0 และ P1.1 จะเป็นขาอินพุทของ Analog comparator โดย P1.0 (AIN0) เป็นขาอินพุทบวก (positive input) P1.1 (AIN1) เป็นขาอินพุทลบ (negative input) วงจรเอาต์พุทของ Port 1 สามารถซิงค์กระแสได้ 20 mA. จึงสามารถขับ LED ได้โดยตรง ถ้าต้องการใช้ Port 1 เป็นอินพุทพอร์ต เราจะต้องเขียนข้อมูลเป็น Logic 1 ไปยัง Port 1 ก่อน นอกจากนี้แล้ว Port 1 ยังเป็นพอร์ตที่รับข้อมูล โปรแกรมที่จะ โปรแกรมไปยังหน่วยความจำแฟลช (Flash Memory) ด้วย
- Port 3 ขา P3.0 ถึง P3.5 และ P3.7 เป็นพอร์ต 2 ทิศทาง (I/O Port) มีการพูลอัพภายใน (internal pull up) สำหรับ P3.6 เป็นบิตที่แสดงเอาต์พุทของ Analog comparator โดยไม่มีขาต่อออกมาภายนอก วงจรภาคเอาต์พุทของ Port 3 สามารถซิงค์กระแสได้ 20 mA. และสามารถขับ LED ได้โดยตรง เช่นเดียวกับ Port 1 เมื่อ Logic 1 ถูกเขียนไปยัง Port 3 เอาต์พุทของ Port 3 จะเป็น High เนื่องจากการพูลอัพภายใน และสามารถทำหน้าที่เป็นอินพุทพอร์ตได้ นอกจากนี้แล้ว Port 3 ยังทำหน้าที่อื่นอีก ดังแสดงในตารางที่ ก.1

ตารางที่ ก.1 แสดงหน้าที่ของขาต่างๆในพอร์ต 3

Port pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{\text{INT0}}$ (external interrupt 0)
P3.3	$\overline{\text{INT1}}$ (external interrupt 1)
P3.4	T0 (timer 0 external interrupt)
P3.5	T1 (timer 1 external interrupt)

- RST เป็นขาริเซทของ 89C2051 โดยจะถูกริเซทเมื่อ ขา RST เป็น High อยู่เป็นเวลามากกว่า 2 machine cycles โดย 1 machine cycles เท่ากับ 12 Clock ของความถี่ออสซิลเลเตอร์

3. หน่วยความจำสำหรับข้อมูลภายใน 89C2051 ช่วง 00H ถึง 7FH

สามารถแบ่งออกได้เป็น 3 กลุ่มคือ

3.1 Register bank 0 ถึง 3 อยู่ในตำแหน่งหน่วยความจำช่วงตำแหน่งที่ 00H ถึง 1FH หน่วยความจำนี้จะแบ่งออกเป็น 4 ชุด ชุดละ 8 ไบต์แต่ละชุดเรียกว่า BANK แต่ละไบต์ใน IBANK จะมีชื่อของรีจิสเตอร์ ว่า R0,R1,R2,R3,R4,R5,R6 และ R7 รีจิสเตอร์เหล่านี้เหล่านี้สามารถเรียกใช้งานในระหว่างทำโปรแกรมได้อย่างสะดวก และรีจิสเตอร์เหล่านี้จะมีชื่อซ้ำกันในทุก BANK การใช้งานจึงต้องจึงต้องเรียกใช้งานที่ละ BANK เท่านั้น โดยการกำหนดในรีจิสเตอร์ PSW เมื่อมีการ Reset การทำงานของ 8051 จะเริ่มการใช้งานของรีจิสเตอร์ R0 ถึง R7 ที่ BANK 0 ซึ่งรีจิสเตอร์ R0 ถึง R7 ในแต่ละ BANK ในการอ้างอิงในหน่วยความจำสำหรับข้อมูลภายใน 89C2051 ดังในตาราง

ตารางที่ ก.2 แสดงการอ้างอิงในหน่วยความจำสำหรับข้อมูลภายใน 89C2051 ของ Reg. Ro-R7

รีจิสเตอร์	ตำแหน่งหน่วยความจำ			
	BANK 0	BANK 1	BANK 2	BANK 3
R0	0	8	10	18
R1	1	9	11	19
R2	2	A	12	1A
R3	3	B	13	1B
R4	4	C	14	1C
R5	5	D	15	1D
R6	6	E	16	1E
R7	7	F	17	1F

3.2 Bit Address Area เป็นหน่วยความจำในช่วงตำแหน่ง 20H ถึง 2FH หน่วยความจำแต่ละบิตในช่วงของหน่วยความจำดังกล่าวจะสามารถตรวจสอบหรือตั้งค่าเป็น 1 หรือ 0 ได้โดยการโปรแกรมภาษาเครื่องแต่ละบิตของข้อมูลในหน่วยความจำช่วงนี้จะมีค่าตำแหน่งดังใน Memory Map

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Scratched pod Area เป็นช่วงของหน่วยความจำตำแหน่ง 30H ถึง 7FH หน่วยความจำช่วงนี้จะใช้สำหรับเก็บข้อมูลทั่วไป ถ้ารีจิสเตอร์ Stack pointer ซึ่งอ้างหน่วยความจำช่วงนี้จะต้องระวังไม่ให้เกิดการเขียนทับของข้อมูลอื่นจะทำให้การทำงานของโปรแกรมผิดพลาดได้

Byte	(MSB)				(LSB)				
7FH	Scratch Psd Area								
30H									
2FH									
2EH	7F	7E	7D	7C	7B	7A	79	78	
2DH	77	76	75	74	73	72	71	70	
2CH	6F	6E	6D	6C	6B	6A	69	68	
2BH	67	66	65	64	63	62	61	61	
2AH	5F	5E	5D	5C	5B	5A	59	58	
29H	57	56	55	54	53	52	51	50	
28H	4F	4E	4D	4C	4B	4A	49	49	
27H	47	46	45	44	43	42	41	40	
26H	3F	3E	3D	3C	3B	3A	39	38	
25H	37	36	35	34	33	32	31	30	
24H	2F	2E	2D	2C	2B	2A	29	28	
23H	27	26	25	24	23	22	21	20	
22H	1F	1E	1D	1C	1B	1A	19	18	
21H	17	16	15	14	13	12	11	10	
20H	0F	0E	0D	0C	0B	0A	09	08	
1FH	07	06	05	04	03	02	01	00	
18H	R0 - R7								รีจิสเตอร์ แบงค์ 3
17H	R0 - R7								รีจิสเตอร์ แบงค์ 2
10H	R0 - R7								รีจิสเตอร์ แบงค์ 1
0FH	R0 - R7								รีจิสเตอร์ แบงค์ 0
08H	R0 - R7								
07H	R0 - R7								
00H	R0 - R7								

รูปที่ ก.2 128 ไบต์ของ Ram ที่เข้าถึงข้อมูลแบบทางตรงและทางอ้อม

มีรีจิสเตอร์อีกกลุ่มหนึ่งที่เรียกว่า Special Function Register (SFR) เป็นรีจิสเตอร์ใช้เฉพาะงาน แต่ละตำแหน่งของ SFR อาจไม่ใช่เป็นหน่วยความจำแต่อาจใช้เป็น Counter Register , Shift register ซึ่งการอ้างอิงข้อมูลในตำแหน่งนั้น 89C2051 จะมองว่าเป็นหน่วยความจำตำแหน่งหนึ่ง จึงเรียกรวมองข้อมูลแต่ละตำแหน่งว่า Memory Map I/O ดังแสดงตำแหน่งดังนี้

ตารางที่ ก.3 แสดงรีจิสเตอร์ใช้งานพิเศษ (SFR)

สัญลักษณ์	ชื่อ	ตำแหน่ง
*ACC	Accumulator	OEOH
*B	B Register	OFOH
*PSW	Program Status Word	ODOH
SP	Stack Pointer	81H
DPTR	Data Pointer	
DPL	Low Byte	82H
DPH	High Byte	83H
*P0	Port 0	80H
*P1	Port 1	90H
*P2	Port 2	OAOH
*P3	Port 3	OBOH
*IP	Interrupt Priority Control	OB8H
*IE	Interrupt Enable	OA8H
TMOD	Timer/Counter Mode Control	89H
*TCON	Timer/Counter Control	88H
TH0	Timer/Counter 0 High Byte	8CH
TLO	Timer/Counter 0 Low Byte	8AH
TH1	Timer/Counter 1 High Byte	8DH
TL1	Timer/Counter 1 Low Byte	8BH
*SCON	Serial Control	98H
SBUF	Serial Data Buffer	99H
PCON	Power Control	87H

* เป็นรีจิสเตอร์ที่สามารถอ้างอิงแอดเดรสแบบบิตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความรู้เบื้องต้นเกี่ยวกับมาตรฐาน RS 232

1. พอร์ต RS 232 C

โดยปกติไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นอนุกรม เรียกชื่อกันว่า RS 232C อยู่ในตัวมันเองอยู่แล้ว หลายเครื่อง ไม่มีมากับเครื่อง อย่างเช่น IBM PC จำเป็นที่จะต้องมีการ์ดที่เรียกว่า อะซิงโครนัสอะแดปเตอร์ (Asynchronous Communication Adapter) มาเสียบใส่

พอร์ต RS 232 นี้ทำหน้าที่รับและส่งข้อมูลในแบบอนุกรมเรียกว่า Universal Asynchronous Adapter เหตุที่มีชื่อเรียกว่า RS 232 ก็เนื่องมาจากสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์ของอเมริกา หรือ EIA ได้กำหนดมาตรฐานของอุปกรณ์การสื่อสารแบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS 232 ความจริงมาตรฐานของการส่งข้อมูลแบบอนุกรมมีหลายมาตรฐาน แต่ที่นิยมกันมากที่สุดสำหรับไมโครคอมพิวเตอร์ก็คือ RS 232

หน้าที่สำคัญของการสื่อสารแบบอะซิงโครนัสก็คือ
รับสัญญาณ

1. เปลี่ยนสัญญาณเข้ามาแบบอนุกรมให้เป็นแบบขนาน
2. ตรวจสอบความผิดพลาดของสัญญาณที่รับ
3. ตัด stop bit และ parity
4. ส่งสัญญาณให้ซีพียูรู้ว่ารับสัญญาณไว้แล้ว

ส่งสัญญาณ

1. เปลี่ยนสัญญาณแบบขนานจากซีพียูค่อยทยอยส่งออกเป็นแบบอนุกรม
2. เพิ่ม stop bit และ parity
3. เพิ่มสัญญาณควบคุมโมเด็มที่ต่อเชื่อม (ถ้ามี) พอร์ตสื่อสารข้อมูลอนุกรม RS 232

2. คุณสมบัติของ RS 232C

อัตราการรับส่งข้อมูล	: 0 – 20000 บิตต่อวินาที
ระดับแรงดันเอาต์พุตสูงสุดในสถานะ ไม่มีโหลด	: -25 โวลต์ (ลอจิก “1”) +25 โวลต์ (ลอจิก “0”)
ระดับแรงดันเอาต์พุตสำหรับ โหลด 3-7 กิโลโอห์ม	: ลอจิก “1” -15 โวลต์ (7 กิโลโอห์ม) -5 โวลต์ (3 กิโลโอห์ม) : ลอจิก “0” +15 โวลต์ (7 กิโลโอห์ม) +5 โวลต์ (3 กิโลโอห์ม)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

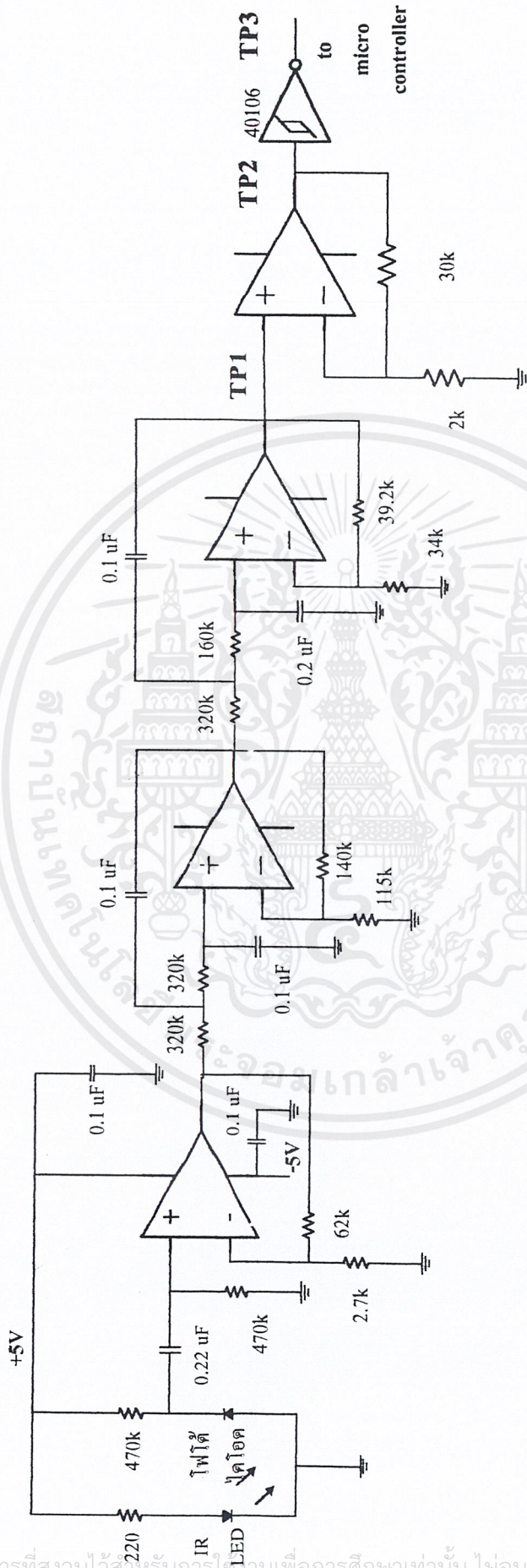
กระแสเอาต์พุท เมื่อลัดวงจร	: สูงสุด 500 มิลลิแอมป์
เอาต์พุทอิมพีแดนซ์เมื่อไม่จ่ายไฟเลี้ยง	: ต่ำสุด 300 โอห์ม
สล็อตเรททางเอาต์พุทสูงสุด	: 30 โวลต์ต่อไมโครวินาที
ความต้านทานอินพุทของภาครับ	: สูงสุด 7 กิโลโอห์ม
	: ต่ำสุด 3 กิโลโอห์ม
ค่าความจุอินพุทของภาครับ	: สูงสุด 2,500 พิโกฟารัด
ย่านแรงดันอินพุทของภาครับ	: -25 โวลต์ ถึง +25 โวลต์

3. มาตรฐาน RS 232C

มาตรฐาน RS 232C ได้จัดพิมพ์ขึ้นเมื่อปี ค.ศ. 1969 โดยสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์แห่งสหรัฐอเมริกา RS ย่อมาจาก Recommended Standard ส่วน 232 เป็นหมายเลขของบอกรของมาตรฐานตัวนี้ C เป็นหมายเลขท้ายสุดของมาตรฐานตัวนี้ จุดประสงค์ของมาตรฐานตัวนี้ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งปลายทาง (Data Terminal Equipment :DTE) กับอุปกรณ์สื่อสารข้อมูล (Data communication Equipment : DCE) สำหรับผู้ใช้ไมโครคอมพิวเตอร์ DTE ก็หมายถึงตัวไมโครคอมพิวเตอร์ และ DCE ก็หมายถึง โมเด็ม อุปกรณ์อื่นๆเช่น เครื่องพิมพ์ที่รับสัญญาณแบบอนุกรมอาจเป็นไปได้ทั้ง DTE และ DCE ขึ้นอยู่กับผู้ผลิต

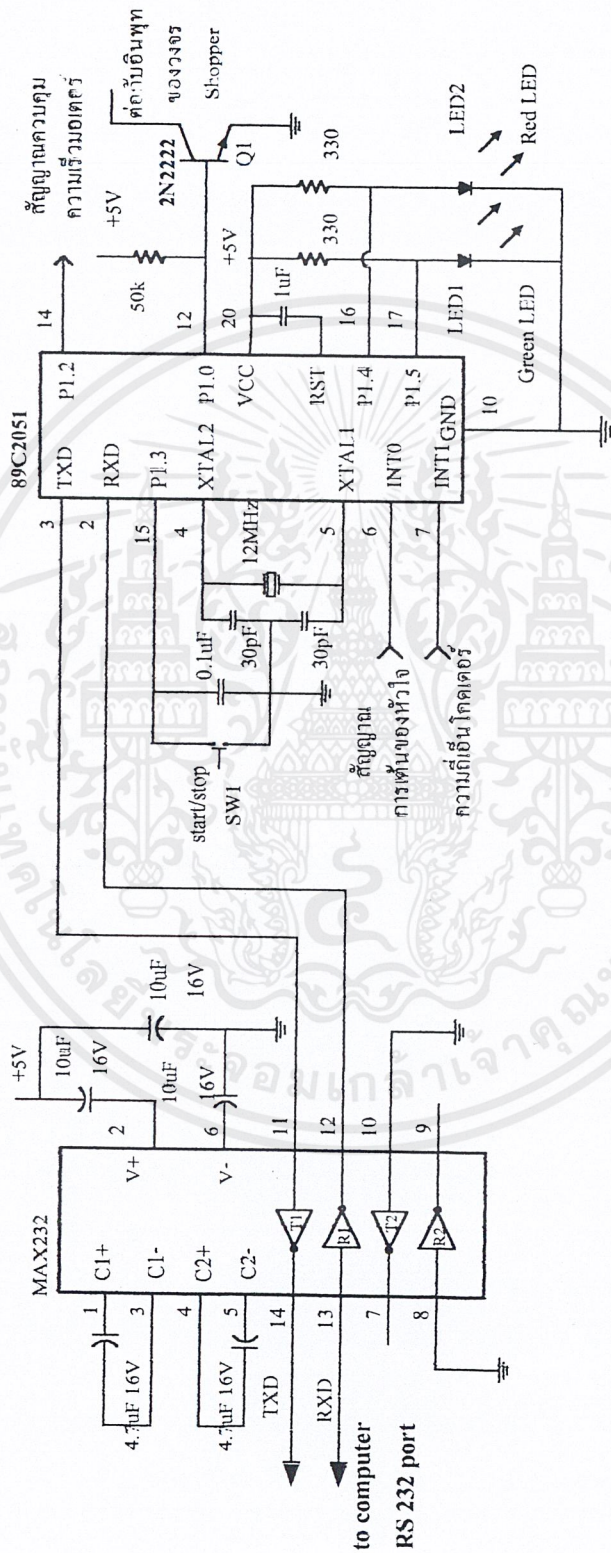


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



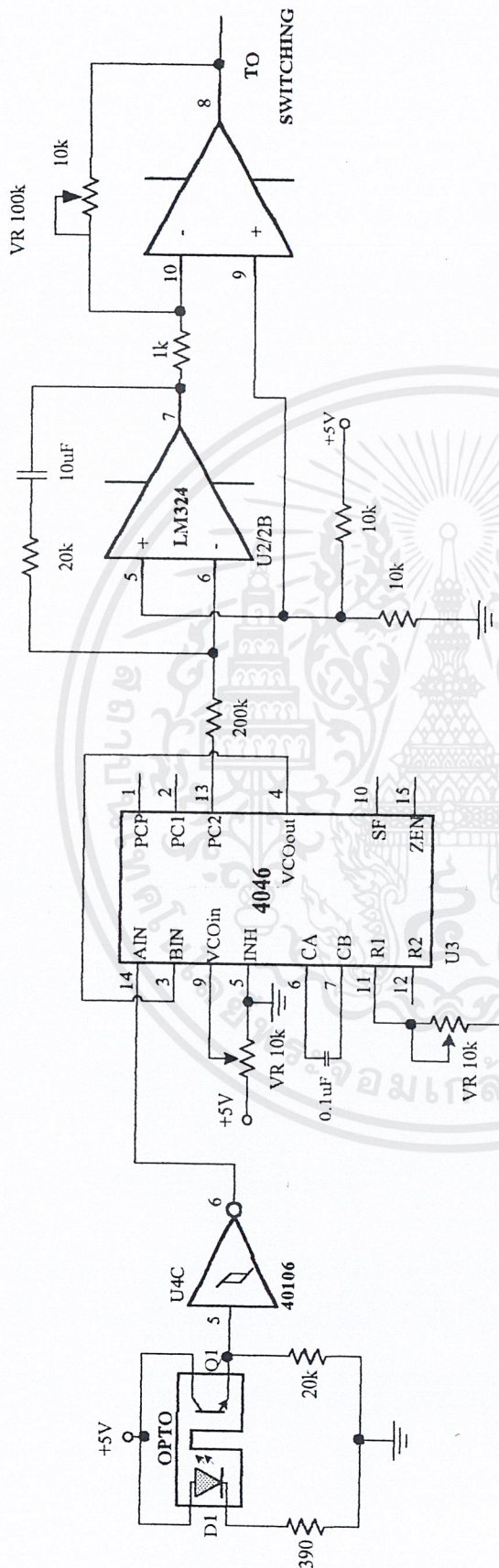
รูปที่ ข.1 วงจรวัดอัตราการเต้นของหัวใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



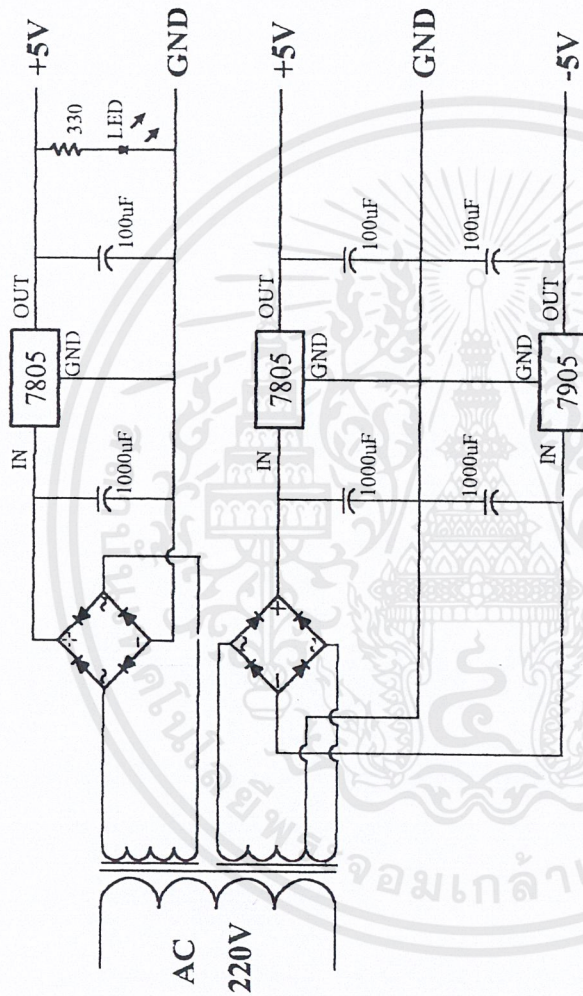
รูปที่ ข.2 วงจรควบคุมโดยใช้ไมโครคอนโทรลเลอร์ 89C2051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



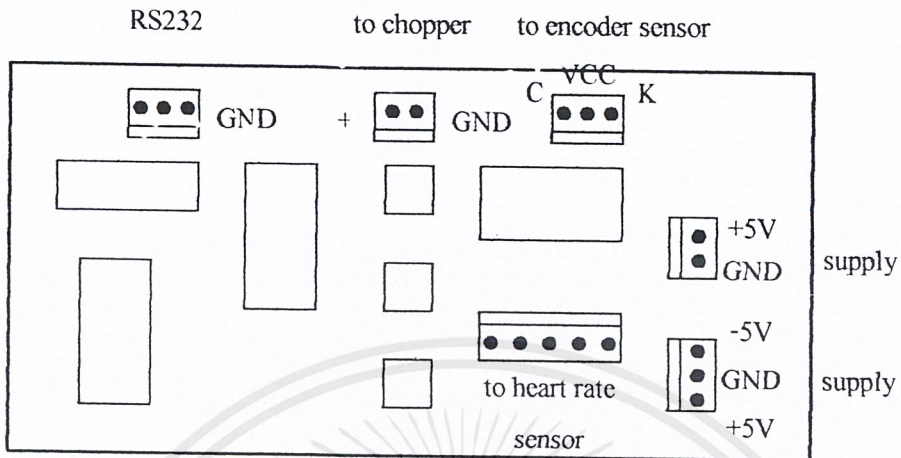
รูปที่ ๓.3 วงจรควบคุมความเร็วมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

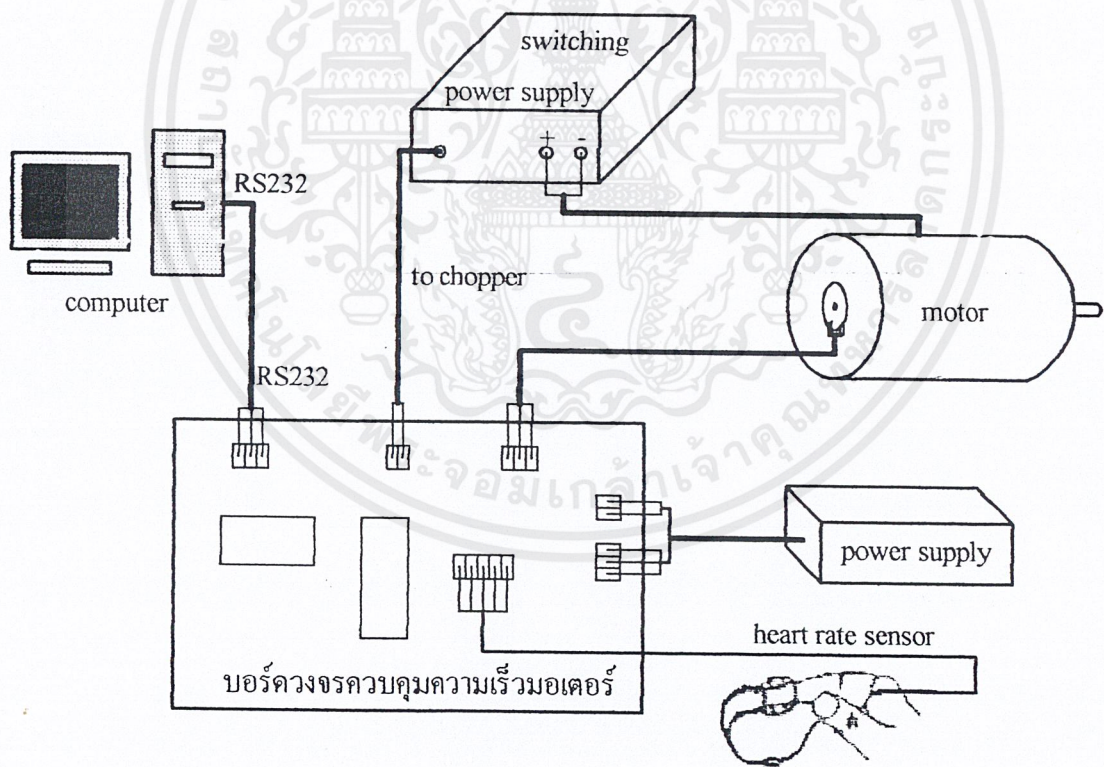


รูปที่ ข.4 วงจรภาคจ่ายไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.5 แสดงจุดต่อต่างๆของบอร์ดวงจรควบคุมความเร็วมอเตอร์



รูปที่ ข.6 แสดงการต่อระบบทั้งหมดของการควบคุมความเร็วมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค

โปรแกรม Assembly

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมแอสเซมบลีควบคุมการทำงานไมโครคอนโทรลเลอร์ 89C2051

```
H2 EQU 30H
H1 EQU 31H
H0 EQU 32H
B3 EQU 33H
B2 EQU 34H
B1 EQU 35H
B0 EQU 36H
AS1 EQU 37H
AS0 EQU 38H
TT1 EQU 39H
TT0 EQU 3AH
FF1 EQU 3BH
FF0 EQU 3CH
INT0_FLAG EQU 0H
INT1_FLAG EQU 1H
XTI EQU 2H
SR_FLAG EQU 3H
F_FLAG EQU 4H
STSP_FLAG EQU 5H
SW_FLAG EQU 6H
GREENBIT P1.5
RED BIT P1.4
```

ORG 0

```
SJMP MAIN
```

ORG 03H

```
AJMP INT0_ROUTINE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ORG 0BH
    AJMP TIMER0_ROUTINE

ORG 13H
    AJMP INT1_ROUTINE

ORG 1BH
    AJMP TIMER1_ROUTINE

ORG 23H
    AJMP SR_ROUTINE

ORG 40H

MAIN:
    MOV SP,#50H
    MOV TMOD,#23H           ;timer1 mode2,timer0 mode 3
    MOV PCON,#80H          ;setb smod
    MOV SCON,#52H
    MOV TH1,#-13
    MOV TL1,#-13           ; determine baudrate 4800
    SETB TR1               ; start timer1
    SETB IT0               ; int0 transition interrupt
    SETB IT1               ; int1 transition interrupt
    MOV IE,#0FFH          ; enable all interrupt
    SETB PT0               ; timer0 ,high priority
    CLR GREEN
    SETB TR0
    CLR ET0                ;disable timer0
    CLR INT0_FLAG
    CLR INT1_FLAG
    CLR SR_FLAG
    SETB SW_FLAG
    CLR STSP_FLAG
    MOV A,#0AH

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ACALL SRWR
MOV A,#0DH
ACALL SRWR

LLFF3:  JNB INT0_FLAG,LLFF
        MOV H1,TT1
        MOV H0,TT0
        ACALL HEX_TO_BCD
        MOV A,#T           ;start of block data
        ACALL SRWR        ;send start flag to pc
        MOV R0,#B2
LL1:    MOV A,@R0
        ACALL HEX_TO_ASCII
        MOV A,AS1
        ACALL SRWR
        MOV A,AS0
        ACALL SRWR
        INC R0
        CJNE R0,#(B0+1),LL1
        CLR INT0_FLAG

LLFF:   JNB INT1_FLAG,LLFF1
        MOV H1,FF1
        MOV H0,FF0
        ACALL HEX_TO_BCD
        MOV A,#F'
        ACALL SRWR
        MOV R0,#B2
LL2:    MOV A,@R0
        ACALL HEX_TO_ASCII

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV A,AS1
ACALL SRWR
MOV A,AS0
ACALL SRWR
INC R0
CJNE R0,#(B0+1),LL2
CLR INT1_FLAG
```

```
LLFF1: JNB SR_FLAG,LLFF2
      SETB RS1 ;determine register bank 2
      MOV AS1,40H
      MOV AS0,41H
      ACALL ASCII_TO_HEX
      MOV R2,A
      MOV AS1,42H
      MOV AS0,43H
      ACALL ASCII_TC_HEX
      MOV R3,A
      SETB F_FLAG
      CLR SR_FLAG
      CLR RS1
```

```
LLFF2: JB P1.3,SWO
      JB SW_FLAG,PASWO
      AJMP LLFF3
```

```
PASWO: CLR SW_FLAG
      JNB STSP_FLAG,STA
      CLR STSP_FLAG
      CLR GREEN
      SETB RED
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV A,#<'
ACALL SRWR
AJMP LLFF3
STA: SETB STSP_FLAG
CLR RED
SETB GREEN
MOV A,#>'
ACALL SRWR
AJMP LLFF3
SWO: SETB SW_FLAG
AJMP LLFF3
;-----timer TL0 interrupt service routine-----
;use register bank2
;reg :r2,r3,r4,r5,r6,r7
TIMER0_ROUTINE:
PUSH PSW
SETB RS1
DJNZ R4,EELL1
CLR TR0
MOV TL0,R5
SETB TR0
POP PSW
RETI
EELL1: CJNE R4,#0FFH,IE0_EXIT
CPL P1.2
JNB F_FLAG,EELL2
CLR F_FLAG
MOV R6,12H
MOV R7,13H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

EELL2:      MOV R4,16H
            MOV R5,17H
IE0_EXIT:   POP PSW
            RETI

;-----timerTH0 interrupt routine-----
;use register bank 1
;r1,r0:period of heart rate
;r3,r2:counter 1 s
TIMER1_ROUTINE:  PUSH PSW
                SETB RSO                ;determinc register bank 1
                INC R0
                CJNE R0,#0,LLN1
                INC R1
LLN1:         INC R2
                CJNE R2,#0,LLN2
                INC R3
LLN2:         CJNE R3,#0FH,LLN3
                CJNE R2,#42H,LLN3
                MOV FF1,R5
                MOV FF0,R4                ;FF1,FF0 = frequency of motor
                MOV R2,#0
                MOV R3,#0
                MOV R4,#0
                MOV R5,#0
                SETB INT1_FLAG
LLN3:         POP PSW
                RETI

;-----INT0 intrrupt service routine-----
;use register bank 1
;REG A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

INT0_ROUTINE: CLR IE0
                PUSH PSW
                SETB RS0
                MOV TT1,R1
                MOV TT0,R0
                MOV R0,#0
                MOV R1,#0
                SETB INT0_FLAG
                POP PSW
                RETI

```

```

;-----INT1 interrupt service routine-----
;use register bank1
;r5,r4:count pluse
INT1_ROUTINE: CLR IE1
                PUSH PSW
                SETB RS0                ;determine register bank 1
                INC R4
                CJNE R4,#0,LLM
                INC R5
LLM:           POP PSW
                RETI

```

```

;-----serial interrupt service routine-----
;use register bank2
;reg:r0
SR_ROUTINE:   JBC TI,PPL1
                CLR RI
                PUSH PSW
                SETB RS1
                PUSH ACC
                MOV A,SBUF

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CJNE A,#>',SR1 ;when mcs51 recive '>', it will enable timer t10
CLR P1.0
SETB ET0
SETB STSP_FLAG
CLR RED
SETB GREEN
SJMP SR_EXIT
SR1: CJNE A,#<',SR2 ;when mcs51 recive '<' it will disable timer t10
SETB P1.0
CLR ET0
CLR STSP_FLAG
CLR GREEN
SETB RED
SJMP SR_EXIT
SR2: CJNE A,#'M',SRR1 ;M is start of block character ,Tcontrol
MOV R0,#40H
SJMP SR_EXIT
SRR1: CJNE A,#'V',SRR2 ;V is end of block character, Tcontrol
SETB SR_FLAG
SJMP SR_EXIT
SRR2: MOV @R0,A
INC R0
SR_EXIT: POP ACC
POP PSW
RETI
PPL1: SETB XT1
RETI
;-----hex to bcd-----
;input H1,H0
;output B2,B1,B0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

;registers A,R0,R1,R2

HEX_TO_BCD: CLR A

MOV B2,A

MOV B1,A

MOV B0,A

MOV R0,#16

ADL1: MOV A,H0

RLC A

MOV H0,A

MOV A,H1

RLC A

MOV H1,A

MOV R1,#B0

MOV R2,#3

ADL: MOV A,@R1

ADDC A,ACC

DA A

MOV @R1,A

DEC R1

DJNZ R2,ADL

DJNZ R0,ADL1

RET

;-----Hex to ascii-----

;input A

;output AS1,AS0

;reg A

HEX_TO_ASCII: PUSH ACC

ANL A,#0F0H

SWAP A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ACALL F_ASCII
MOV AS1,A
POP ACC
ANL A,#0FH
ACALL F_ASCII
MOV AS0,A
RET

```

```

F_ASCII:  INC A
          MOVC A,@A+PC
          RET
          DB '0123456789'
          DB 'ABCDEF'

```

-----ascii to hex-----

;input AS1,AS0

;output a

;reg a

```

ASCII_TO_HEX: MOV A,AS1

```

```

CLR C

```

```

SUBB A,#30H

```

```

ACALL F_HEX

```

```

SWAP A

```

```

MOV AS1,A

```

```

MOV A,AS0

```

```

CLR C

```

```

SUBB A,#30H

```

```

ACALL F_HEX

```

```

ORL A,AS1

```

```

RET

```

```

F_HEX:  INC A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOVC A,@A+PC
RET
DB 0,1,2,3,4,5,6,7,8,9
DB 0,0,0,0,0,0,0
DB 0AH,0BH,0CH,0DH,0EH,0FH
```

-----send data to rs port-----

;input a

```
SRWR:   JNB XTI,$
        CLR XTI
        MOV SBUF,A
        RET
```

-----rec data from rs port-----

;output a

```
SRRD:   JNB RI,$
        CLR RI
        MOV A,SBUF
        RET
```

END



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCL4046B
SCL4446B



CMOS PHASE-LOCKED LOOPS

FEATURES

- ◆ Very low power consumption – 70 μ W (typ) @ $f_o = 10$ kHz, 5Vdc
- ◆ Operating frequency range (no offset) – Up to 3MHz (typ) @ 10Vdc (SCL4046B) Up to 4MHz (typ) @ 10Vdc (SCL4446B)
- ◆ Low frequency drift – 0.04%/°C (typ) @ 10Vdc
- ◆ Choice of two phase comparators:
 1. Exclusive-OR network
 2. Edge-controlled memory network with phase-pulse output for lock indication
- ◆ VCO Inhibit control for ON-OFF keying and ultra-low standby power consumption
- ◆ High VCO linearity 1% (typ)
- ◆ Source-follower output of VCO control input (Demodulator Output)
- ◆ Zener Diode to assist Supply Regulation
- ◆ Balanced Output Drive Current Specifications

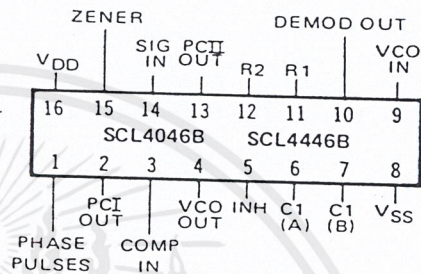
APPLICATIONS

- ◆ FM demodulator and modulator
- ◆ Frequency synthesis and multiplication
- ◆ Frequency discriminator
- ◆ Data synchronization
- ◆ Voltage-to-frequency conversion
- ◆ Tone decoding
- ◆ FSK-Modems
- ◆ Signal conditioning

DESCRIPTION

The SCL4046B and SCL4446B phase-locked loops contain two phase comparators, a voltage-controlled oscillator (VCO), source follower, and zener diode. The comparators have two common inputs. The Signal input can be used directly coupled to large voltage signals, or indirectly coupled (with a series capacitor) to small voltage signals. The self bias circuit adjusts small voltage signals in the linear region of the amplifier. Phase comparator I (an exclusive-OR gate) provides a digital error signal PCI_{OUT} , and maintains 90° phase shift at the center frequency between Signal and Comparator inputs (both at 50% duty cycle). Phase comparator II (with leading edge sensing logic) provides digital error signals PCI_{OUT} and Phase Pulses, and maintains a 0° phase shift between input signals (duty cycle is immaterial). The linear VCO produces an output signal VCO_{OUT} whose frequency is determined by the voltage of input VCO_{IN} and the capacitor and resistors connected to pins C1A, C1B, R1, and R2. The source follower output, Demod Out, with an external resistor is used where the VCO_{IN} signal is needed but no loading can be tolerated. The inhibit input Inh , when high, disables the VCO and source follower to minimize standby power consumption. The zener diode can be used to assist in power supply regulation.

CONNECTION DIAGRAM (all packages)



Add suffix for package:

- C 16-pin Cerdip
- D 16-pin Ceramic
- E 16-pin Epoxy
- F 16-pin Flat
- H Chip

RECOMMENDED OPERATING CONDITIONS

For maximum reliability:

DC Supply Voltage	$V_{DD} - V_{SS}$	3 to 15	Vdc
Operating Temperature	T_A	-55 to +125	°C
C, D, F, H Device		-40 to +85	°C

BLOCK DIAGRAM

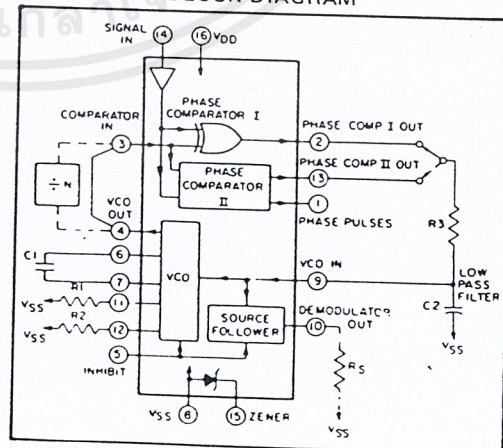


Fig. 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VCO SECTION

The VCO requires one external capacitor (C1) and one to two external resistors (R1 or R1 and R2). Resistor R1 and capacitor C1 determine the frequency range of the VCO and resistor R2 enables the VCO to have a frequency offset if required. The high input impedance ($10^{12}\Omega$) of the VCO simplifies the design of low-pass filters by permitting the designer a wide choice of resistor-to-capacitor ratios. In order not to load the low-pass filter, a source-follower output of the VCO input voltage is provided at terminal 10 (DEMODULA-

TOR OUTPUT). If this terminal is used, a load resistor (R_L) of $50k\Omega$ or more should be connected from this terminal to V_{SS} . If unused, this terminal should be left open. The VCO can be connected directly or through frequency dividers to the comparator input of the phase comparators. A full CMOS logic swing is available at the output of the VCO. A logic 0 on the INHIBIT input "enables" the VCO and the source follower, while a logic 1 "turns off" both to minimize stand-by power consumption.

PHASE COMPARATORS

The phase-comparator signal input (terminal 14) can be direct-coupled provided the signal swing is within CMOS logic levels [logic "0" $\leq 30\%$ ($V_{DD}-V_{SS}$), logic "1" $\geq 70\%$ ($V_{DD}-V_{SS}$)]. For smaller swings the signal must be capacitively coupled to the self-biasing amplifier at the signal input.

Phase comparator I is an exclusive-OR network; it operates analogously to an over-driven balanced mixer. To maximize the lock range, the signal and comparator-input frequencies must have a 50% duty cycle. With no signal or noise on the signal input, this phase comparator has an average output voltage equal to $V_{DD}/2$. The low-pass filter connected to the output of phase comparator I supplies the averaged voltage to the VCO input, and causes the VCO to oscillate at the center frequency (f_0).

The frequency range of input signals on which the PLL will lock, if it was initially out of lock, is defined as the frequency capture range ($2f_c$).

The frequency range of input signals on which the loop will stay locked if it was initially in lock is defined as the frequency lock range ($2f_L$). The capture range can not exceed the lock range.

With phase comparator I, the range of frequencies over which the PLL can acquire lock (capture range) is dependent on the low-pass-filter characteristics, and can be made as large as the lock range. Phase-comparator I enables a PLL system to remain in lock in spite of high amounts of noise in the input signal.

One characteristic of this type of phase comparator is that it may lock onto input frequencies that are close to harmonics of the VCO center-frequency. A second characteristic is that the phase angle between the signal and the comparator input varies between 0° and 180° , and is 90° at the center frequency. Figure 2 shows the (typical) triangular phase-to-output response characteristic of phase-comparator I. Typical waveforms for a CMOS phase-locked-loop employing phase comparator I in locked condition is shown in Figure 3.

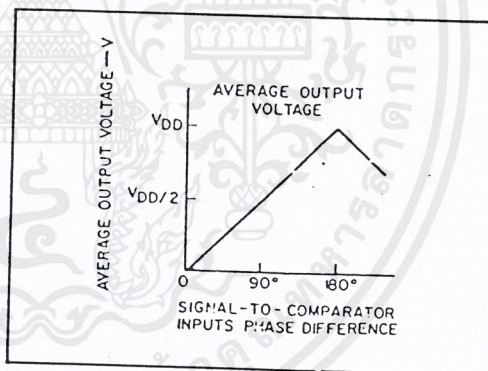


Fig. 2 — Phase-comparator I characteristics at low-pass filter output.

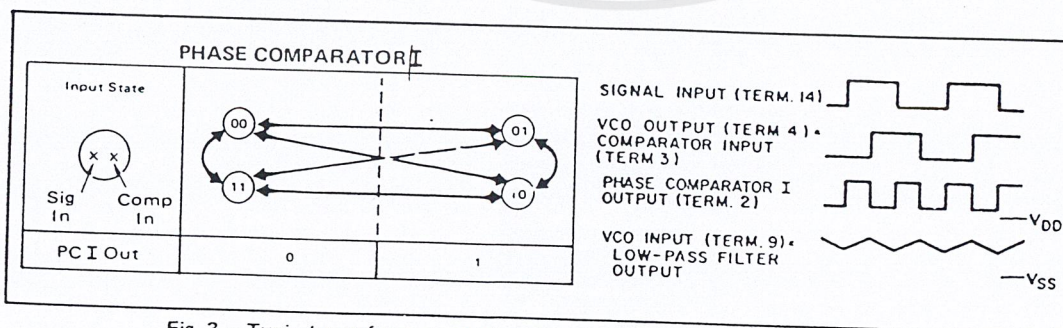


Fig. 3 — Typical waveforms employing phase comparator I in locked condition

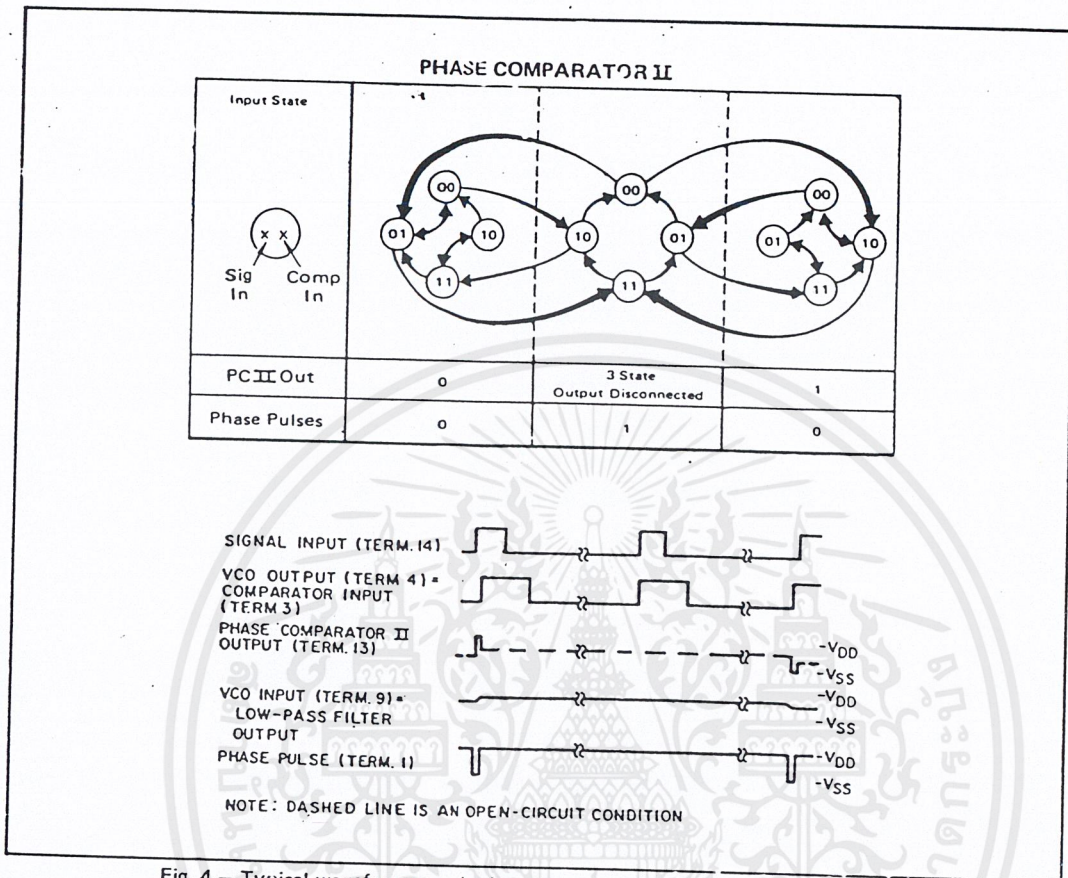


Fig. 4 — Typical waveforms employing phase comparator II in locked condition.

Phase-comparator II is an edge-controlled digital memory network. It consists of several flip-flop stages, control gating, and a three state output circuit comprising p- and n-type drivers having a common output node. When the p-MOS or n-MOS drivers are ON, they pull the output up to V_{DD} or down to V_{SS} , respectively. This type of phase comparator acts only on the positive edges of the signal and comparator inputs. The duty cycles of the signal and comparator inputs are not important since positive transitions control the PLL system utilizing this type of comparator. If the signal lags the comparator input in phase, the n-type output driver is maintained ON for a time corresponding to the phase difference. If the comparator input lags the signal in phase, the p-type output driver is maintained ON for a time corresponding to the phase difference. Subsequently, the capacitor voltage of the low-pass filter connected to this phase comparator is adjusted until the signal and comparator inputs are equal in both phase and frequency. At this stable point, both p- and n-type output

drivers remain OFF. Thus, the phase comparator output becomes an open circuit and holds the voltage on the capacitor of the low-pass filter constant. Moreover, the signal at the "phase pulses" output is a high level which can be used for indicating a locked condition. Thus, for phase comparator II, no phase difference exists between signal and comparator input over the full VCO frequency range. Moreover, the power dissipation due to the low-pass filter is reduced when this type of phase comparator is used because both the p- and n-type output drivers are OFF for most of the signal input cycle.

It should be noted that the PLL lock range for this type of phase comparator is equal to the capture range, independent of the low-pass filter. With no signal present at the signal input, the VCO is adjusted to its lowest frequency for phase comparator II. Figure 4 shows typical waveforms for a CMOS PLL employing phase comparator II in a locked condition.

DESIGN INFORMATION

This information is a guide for approximating the values of external components for the SCL4046B and SCL4446B in a Phase-Locked Loop system. The selected external components must be within the following ranges:

$$R1, R2 \geq 2k\Omega, R_S \geq 10k\Omega$$

$$C1 \geq 15pF$$

In addition to the given design information refer to Figure 5 for $R1$, $R2$, and $C1$ component selections

CHARACTERISTICS	USING PHASE COMPARATOR I		USING PHASE COMPARATOR II	
	VCO WITHOUT OFFSET $R_2 = \infty$	VCO WITH OFFSET	VCO WITHOUT OFFSET $R_2 = \infty$	VCO WITH OFFSET
VCO Frequency				
For No Signal Input	VCO in PLL system will adjust to center frequency, f_0		VCO in PLL system will adjust to lowest operating frequency, f_{min}	
Frequency Lock Range, $2f_L$	$2f_L = \text{full VCO frequency range}$ $2f_L = f_{max} - f_{min}$			
Frequency Capture Range, $2f_C$	$2f_C \approx \frac{1}{\pi} \sqrt{\frac{2\pi f_L}{R1}}$		$f_C = f_L$	
Loop Filter Component Selection	<p style="text-align: center;">For $2f_C$, see Ref.</p>			
Phase Angle between Signal and Comparator	90° at center frequency (f_0), approximating C^0 and 180° at ends of lock range ($2f_L$)		Always 0° in lock	
Locks on Harmonics of Center Frequency	Yes		No	
Signal Input Noise Rejection	High		Low	
VCO Component Selection	<ul style="list-style-type: none"> - Given: f_0 - Use f_0 with Fig.5a to determine $R1$ and $C1$ 	<ul style="list-style-type: none"> - Given: f_0 and f_L - Calculate f_{min} from the equation $f_{min} = f_0 - f_L$ - Use f_{min} with Fig. 5b to determine $R2$ and $C1$ - Calculate $\frac{f_{max}}{f_{min}}$ from the equation $\frac{f_{max}}{f_{min}} = \frac{f_0 + f_L}{f_0 - f_L}$ - Use $\frac{f_{max}}{f_{min}}$ with Fig.5c to determine ratio $R2/R1$ to obtain $R1$ 	<ul style="list-style-type: none"> - Given: f_{max} - Calculate f_0 from the equation $f_0 = \frac{f_{max}}{2}$ - Use f_0 with Fig.5a to determine $R1$ and $C1$ 	<ul style="list-style-type: none"> - Given: f_{min} & f_{max} - Use f_{min} with Fig.5b to determine $R2$ and $C1$ - Calculate $\frac{f_{max}}{f_{min}}$ - Use $\frac{f_{max}}{f_{min}}$ with Fig.5c to determine ratio $R2/R1$ to obtain $R1$

REF. G. S. Moschytz, "Miniaturized RC Filters Using Phase-Locked Loop", BSTJ, May, 1965.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS

PARAMETER	V _{DD} (Vdc)	CONDITIONS	T _{LOW} ²		+25°C			T _{HIGH} ²		Units
			Min.	Max.	Min.	Typ.	Max.	Min.	Max.	
QUIESCENT DEVICE CURRENT	I _{DD}	Inhibit = V _{DD} Signal Input = V _{DD}	-	5	-	0.05	5	-	150	μA _{dc}
			-	10	-	0.01	10	-	300	
			-	20	-	0.2	20	-	600	
TOTAL POWER DISSIPATION	P _T	Inh = V _{SS} , VCO _{IN} = $\frac{V_{DD}}{2}$ f _o = 10k Hz, ² C _L = 15pF, R ₁ = 1MΩ, R ₂ = R _S = ∞	-	-	-	0.07	-	-	-	mW
			-	-	-	0.6	-	-	-	
			-	-	-	2.4	-	-	-	

- NOTES: ¹ Remaining Static Electrical Characteristics are listed under "SCL4000B Series Family Specifications".
² T_{LOW} = -55°C for C, D, F, H device.
= -40°C for E device.
T_{HIGH} = +125°C for C, D, F, H device.
= + 85°C for E device.
³ VCO output (pin 4) and Phase Comparator Outputs (pins 2 and 13) have been designed for balanced output drive current specifications. Consult Family Specifications.

PARAMETER	CONDITIONS	V _{DD}	25°C			UNIT			
			Min.	Typ.	Max.				
VCO SECTION									
MAXIMUM OPERATING FREQUENCY SCL4046B	f _{max} R ₂ = ∞ VCO _{IN} = V _{DD}	R1 C1 10k 50pF	5	0.5	0.8	-	MHz		
			10	1.0	1.5	-			
			15	1.3	1.9	-			
		5k 50pF	5	0.6	1.0	-	MHz		
			10	1.4	2.1	-			
			15	1.8	2.7	-			
		2k 50pF	5	-	1.3	-	MHz		
			10	-	2.9	-			
			15	-	3.9	-			
		SCL4446B	R ₂ = ∞ VCO _{IN} = V _{DD}	R1 C1 10k 50pF	5	0.7	1.0	-	MHz
					10	1.3	2.0	-	
					15	1.9	2.8	-	
5k 50pF	5			0.9	1.3	-	MHz		
	10			1.9	2.9	-			
	15			2.6	3.9	-			
2k 50pF	5			-	1.8	-	MHz		
	10			-	3.9	-			
	15			-	5.4	-			
LINEARITY	R ₂ = ∞ VCO _{IN} = 2.5±0.3V, R ₁ ≥ 10kΩ VCO _{IN} = 5.0±2.5V, R ₁ ≥ 400kΩ VCO _{IN} = 7.5±5.0V, R ₁ ≥ 1MΩ			5	-	1	-	%	
				10	-	1	-		
				15	-	1	-		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS (Continued)

PARAMETER	CONDITIONS	V _{DD}	+25°C			UNIT
			Min.	Typ.	Max.	
VCO SECTION (Continued)						
TEMPERATURE-FREQUENCY STABILITY No Offset	R ₂ = ∞	5	—	0.12-0.24	—	%/ ^o C
		10	—	0.04-0.08	—	
		15	—	0.015-0.03	—	
With Offset	R ₂ ≤ 10X R ₁	5	—	0.06-0.12	—	%/ ^o C
		10	—	0.05-0.1	—	
		15	—	0.03-0.06	—	
INPUT RESISTANCE (VCO _{IN})	R _{IN}	5, 10, 15	—	10 ⁶	—	MΩ
OUTPUT DUTY CYCLE	All valid input combinations and voltages		—	50	—	%
OUTPUT TRANSITION TIME	t _{TLH} , t _{THL} C _L = 50pF	5	—	100	200	ns
		10	—	50	100	
		15	—	40	80	
PHASE COMPARATORS						
INPUT RESISTANCE Signal Input	R _{IN}	5	1	3	—	MΩ
		10	0.2	0.7	—	
		15	0.1	0.3	—	
Comparator Input	R _{IN}	5, 10, 15	—	10 ⁶	—	MΩ
AC-COUPLED INPUT SENSITIVITY Signal Input	V _{IN}	5	—	200	400	mV
		10	—	400	800	
		15	—	700	1400	
OUTPUT TRANSITION TIME PCI, PCI Outputs	t _{TLH} , t _{THL} C _L = 50pF	5	—	100	200	ns
		10	—	50	100	
		15	—	40	80	
Phase Pulses Output	t _{TLH} , t _{THL}	5	—	130	260	ns
		10	—	65	130	
		15	—	50	100	
DEMODULATOR OUTPUT						
OFFSET VOLTAGE	VCO _{IN} - V _{DEM} R _S ≥ 50kΩ	5 10 15	— — —	1.4 1.6 1.8	2.2 2.2 2.2	V _{dc}
LINEARITY	R _S ≥ 50kΩ VCO _{IN} = 2.5±0.3V VCO _{IN} = 5.0±2.5V VCO _{IN} = 7.5±5.0V	5 10 15	— — —	0.1 0.6 0.8	— — —	%
ZENER DIODE						
ZENER VOLTAGE	V _Z I _Z = 50μA	—	6.3	7.0	7.7	V
DYNAMIC RESISTANCE	R _Z I _Z = 1mA	—	—	100	—	Ω

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCL4046B, SCL4446B

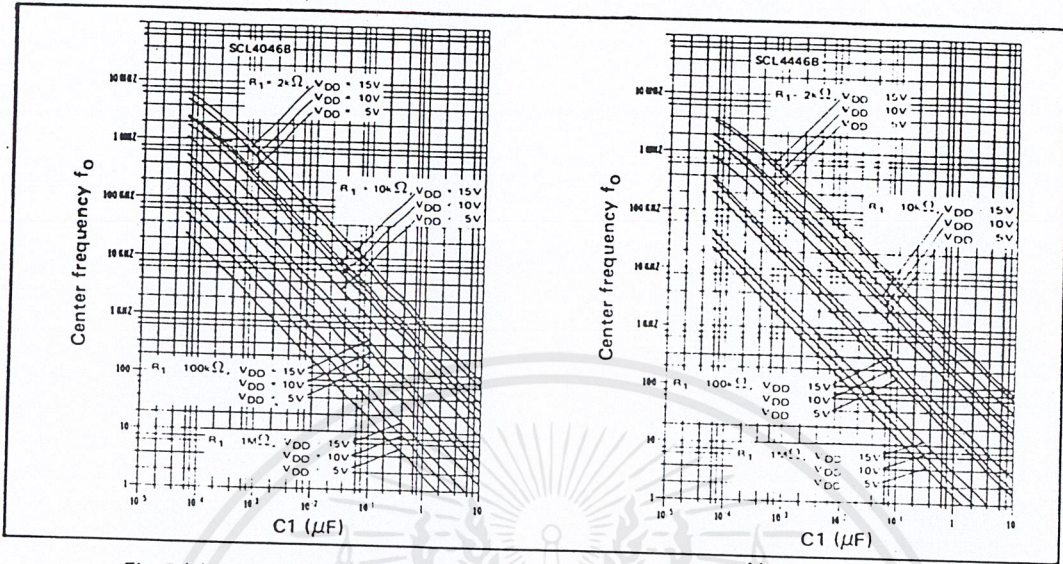


Fig. 5 (a) Typical center frequency (f_0) vs C_1 ($R_2 = \infty$, $V_{COIN} = \frac{V_{DD}}{2}$, $T_A = 25^\circ\text{C}$)

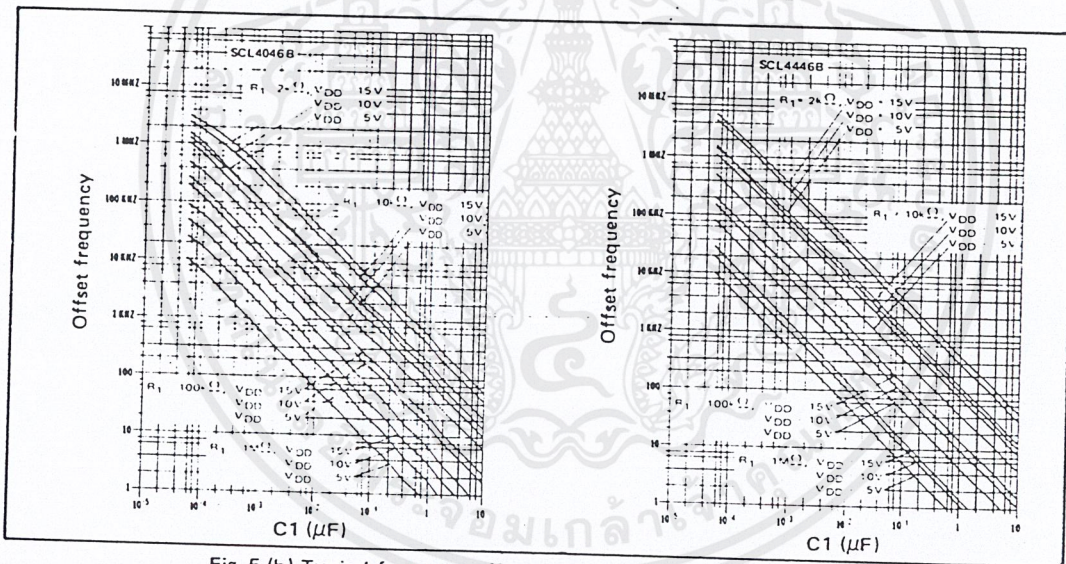


Fig. 5 (b) Typical frequency offset vs C_1 ($V_{COIN} = V_{SS}$, $T_A = 25^\circ\text{C}$)

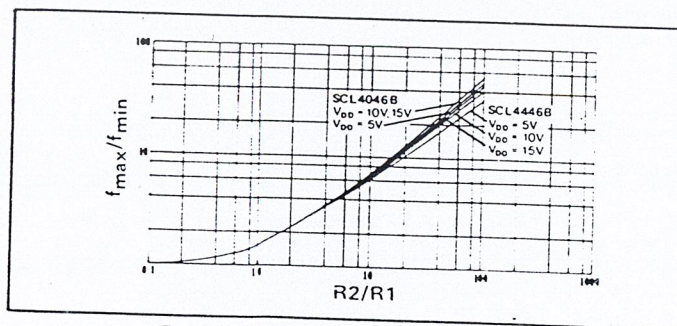


Fig. 5 (c) Typical f_{max}/f_{min} vs R_2/R_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

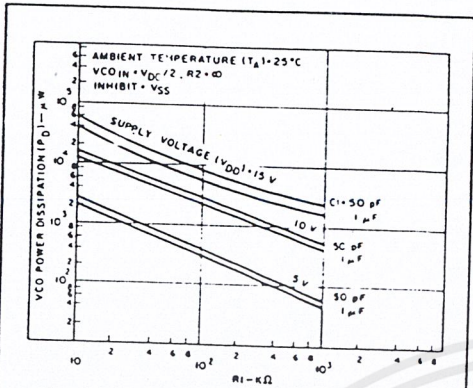


Fig. 6 (a) - Typical VCO power dissipation at center frequency vs R1.

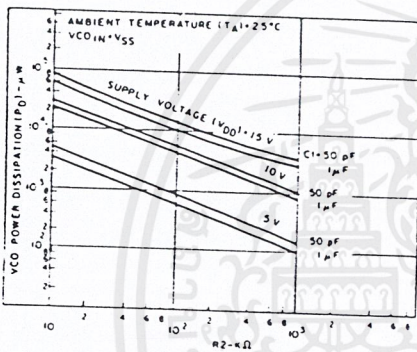


Fig. 6 (b) - Typical VCO power dissipation at f_{min} vs R2.

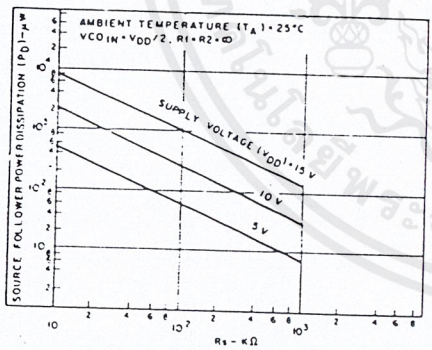


Fig. 6 (c) - Typical source follower power dissipation vs R_S .

NOTE: To obtain approximate total power dissipation of PLL system for no-signal input

$$P_D (\text{Total}) = P_D (f_o) + P_D (f_{MIN}) + P_D (R_S) \\ \text{— Phase Comparator I}$$

$$P_D (\text{Total}) = P_D (f_{MIN}) \\ \text{— Phase Comparator II}$$

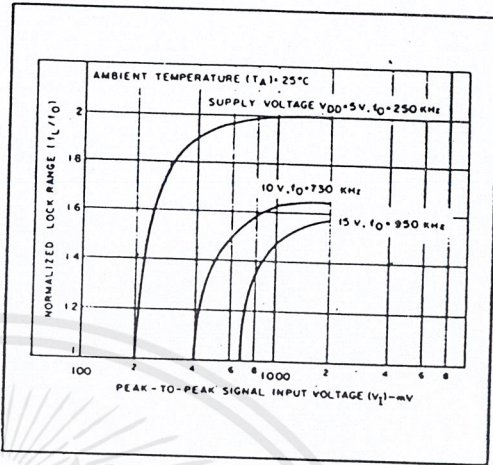


Fig. 7 - Typical lock range vs signal input amplitude

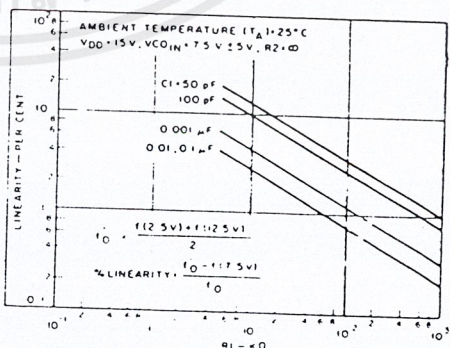
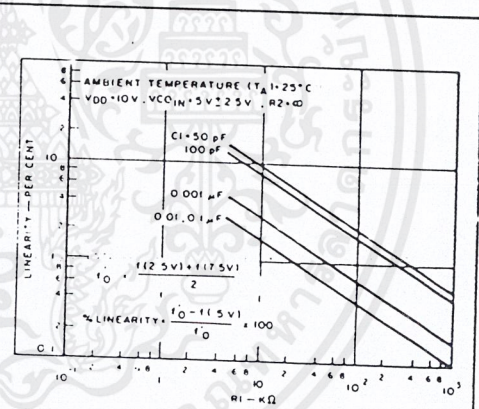


Fig. 8(a, b) - Typical VCO linearity vs R1 and C1

บรรณานุกรม

กิตติ ภัคดีวิวัฒนะกุล และจำลอง ครุอุตสาหะ. **Visual Basic 6 ฉบับโปรแกรมเมอร์**. พิมพ์ครั้งที่ 7.

กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์. 2543

ฟูศักดิ์ ชีวสุวิทย์. **ระบบควบคุมป้อนกลับ**. กรุงเทพฯ : สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณ

ทหารลาดกระบัง. 2539

โยธิน เปรมปราวณิชต์. **ระบบเซอร์โวและอิเล็กทรอนิกส์คอนโทรลมอเตอร์**. กรุงเทพฯ : สถาบัน

เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2539

Benjamin C. Kuo. **Automatic Control Systems**. 7th Ed. Prentice Hall. 1997

Lawrence P. Huelsman. **Active and Passive Analog Filter Design**. McGraw-Hill. 1993

Roland E. Best. **Phase-locked loop : theory, design, and application**. McGraw-Hill. 1993

