

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การบันทึกข้อมูลและการแสดงผลสำหรับอินเวอร์เตอร์ขับเคลื่อนมอเตอร์เหนี่ยวนำ 3 เฟส

DATA ACQUISITION AND MONITORING OF 3 PHASE

INVERTER FED INDUCTION MOTOR



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....

เลขทะเบียน..... 49887

วันเดือนปี..... 2 มี.ย. 2547

b.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกข้อมูลและการแสดงผลสำหรับอินเวอร์เตอร์ขับเคลื่อนมอเตอร์เหนี่ยวนำ 3 เฟส

DATA ACQUISITION AND MONITORING OF 3 PHASE
INVERTER FED INDUCTION MOTOR



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2545

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การบันทึกข้อมูลและการแสดงผลสำหรับอินเวอร์เตอร์ขับเคลื่อนมอเตอร์เหนี่ยวนำ 3 เฟส

DATA ACQUISITION AND MONITORING OF 3 PHASE INVERTER FED

INDUCTION MOTOR

ผู้จัดทำ

- | | | | |
|-----------------|----------------|--------------|----------|
| 1. นายไพโรจน์ | พินิกรัมย์ | รหัสประจำตัว | 43015131 |
| 2. นายรุ่งเรือง | ทรงทรัพย์วัฒนา | รหัสประจำตัว | 43015178 |
| 3. นายสุรชาติ | องค์อินทร์ | รหัสประจำตัว | 43015189 |
| 4. นายแอน | เข็มจันทร์ | รหัสประจำตัว | 43015152 |



[Handwritten Signature]

อาจารย์ที่ปรึกษา

(ผศ.ดร. วิจิตร กิณเรศ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกข้อมูลและการแสดงผลสำหรับอินเวอร์เตอร์ขับเคลื่อนมอเตอร์เหนี่ยวนำ 3 เฟส

นายไพโรจน์ พินิกรมย์
 นายรุ่งเรือง ทรงทรัพย์วัฒนา
 นายสุรชาติ องค์อินทร์
 นายแอน เข็มจันทร์
 ผศ.ดร.วิจิตร กิณเรศ อาจารย์ที่ปรึกษา
 ปีการศึกษา 2545

บทคัดย่อ

การบันทึกข้อมูลและการแสดงผลสำหรับอินเวอร์เตอร์ขับเคลื่อนมอเตอร์เหนี่ยวนำ 3 เฟส เป็นโครงการที่ทำการศึกษาเกี่ยวกับการนำสัญญาณของกระแสและสัญญาณแรงดัน ซึ่งสัญญาณทั้งสองนี้จะรับมาจากมอเตอร์เหนี่ยวนำ 3 เฟส ซึ่งรูปคลื่นแรงดันและกระแสที่วัดได้จะถูกลดทอนให้มีค่าต่ำลงโดยอาศัยการแบ่งแรงดันและตัวตรวจจับกระแส (LEM) ตามลำดับ เพื่อใช้ป้อนให้กับวงจรเปลี่ยนสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลและใช้วงจรป้องกันแรงดันเกินเพื่อป้องกันความเสียหายที่จะเกิดกับคอมพิวเตอร์ แล้วทำการเขียนโปรแกรมติดต่อกับสัญญาณทั้งสอง โดยใช้ภาษา C++ เพื่อทำการติดต่อกับสัญญาณดิจิทัลที่ได้จากวงจรแปลงสัญญาณแล้วนำผลลัพธ์ที่ได้ไปแสดงที่หน้าจอคอมพิวเตอร์ สัญญาณที่แสดงนั้นจะมีสัญญาณของกระแสและสัญญาณแรงดันซึ่งเป็นสัญญาณแบบอนาล็อกนำสัญญาณของกระแสและสัญญาณแรงดันซึ่งเป็นสัญญาณแบบอนาล็อกไปประมวลผลเพื่อหาค่าของแรงบิดแม่เหล็ก และกำลังไฟฟ้าของมอเตอร์ และแสดงสัญญาณแรงบิดและกำลังไฟฟ้าทางจอคอมพิวเตอร์

**DATA ACQUISITION AND MONITORING OF 3 PHASE
INVERTER FED INDUCTION MOTOR**

Pairoj Pinuekram

Rungruang Songsupwattana

Surachart Ong-in

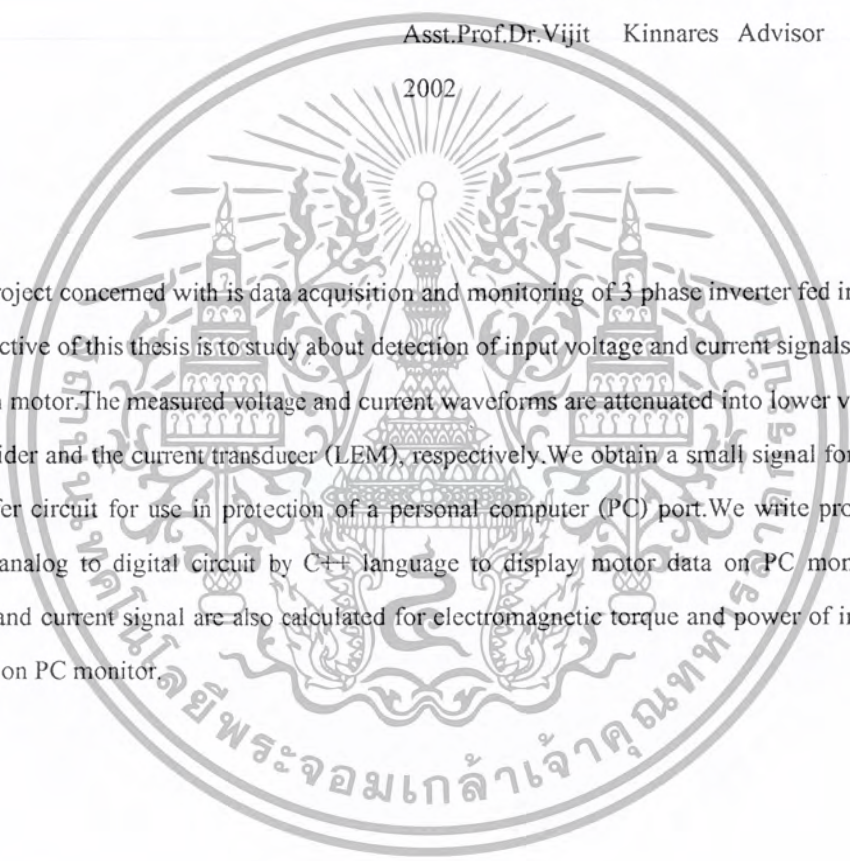
Ant Khemchan

Asst.Prof.Dr.Vijit Kinnares Advisor

2002

ABSTRACT

The project concerned with is data acquisition and monitoring of 3 phase inverter fed induction motor. The objective of this thesis is to study about detection of input voltage and current signals of three phase induction motor. The measured voltage and current waveforms are attenuated into lower values by the voltage divider and the current transducer (LEM), respectively. We obtain a small signal for use via (A/D) and buffer circuit for use in protection of a personal computer (PC) port. We write program to interface with analog to digital circuit by C++ language to display motor data on PC monitor. The voltage signal and current signal are also calculated for electromagnetic torque and power of induction motor showing on PC monitor.



สารบัญ

| | หน้า |
|---|------|
| บทคัดย่อ | I |
| ABSTRACT | II |
| สารบัญ | III |
| สารบัญรูป | IV |
| สารบัญตาราง | VIII |
| บทที่ 1 บทนำ | 1 |
| บทที่ 2 ทฤษฎีที่ใช้ในโครงการ | 2 |
| 2.1 ทฤษฎีของมอเตอร์เหนี่ยวนำ | 2 |
| 2.2 การวิเคราะห์หาค่าแรงบิดของมอเตอร์เหนี่ยวนำ | 18 |
| 2.3 ทฤษฎีการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล | 20 |
| 2.4 ทฤษฎีการติดต่อกับคอมพิวเตอร์ | 26 |
| บทที่ 3 การคำนวณ การสร้าง การออกแบบ | 41 |
| 3.1 การตรวจจับแรงดัน | 41 |
| 3.2 การตรวจจับกระแส | 41 |
| 3.3 วงจรการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล | 43 |
| 3.4 ลักษณะของ ET-PC 8255 | 44 |
| บทที่ 4 ผลการทดลองและการทดสอบ | 49 |
| 4.1 การเปรียบเทียบสัญญาณระหว่างการจำลองโดยใช้โปรแกรม ORCAD กับวัดสัญญาณจริง | 49 |
| 4.2 ทดลอง ADC0820 8-bit โดยใช้ Agilent 54622D Mixed-Signal Oscilloscopes | 56 |
| 4.3 ทดสอบสัญญาณก่อนเข้า ADC0820 | 57 |
| 4.4 แสดงบล็อกไดอะแกรมที่ใช้ในการจำลองในโปรแกรม Matlab | 59 |
| 4.5 เปรียบเทียบสัญญาณระหว่างการจำลองในโปรแกรม Matlab กับสัญญาณที่ได้จากโครงการ | 61 |
| บทที่ 5 บทวิจารณ์และสรุป | 77 |
| ภาคผนวก ก โปรแกรมการคำนวณและแสดงผล | |
| ภาคผนวก ข ข้อมูลเกี่ยวกับอุปกรณ์ที่ใช้ในโครงการ | |
| กิตติกรรมประกาศ | |
| เอกสารอ้างอิง | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

| | หน้า |
|---|------|
| รูปที่ 2.1 หลักการทำงานของมอเตอร์เหนี่ยวนำ 3 เฟส | 3 |
| รูปที่ 2.2 วงจรสมมูลย์ทางไฟฟ้าต่อเฟสของโรเตอร์ 2 รูป | 4 |
| รูปที่ 2.3 วงจรสมมูลย์ทางไฟฟ้าต่อเฟสของสเตเตอร์ | 5 |
| รูปที่ 2.4 วงจรสมมูลย์ทางไฟฟ้าต่อเฟสของมอเตอร์เหนี่ยวนำ เมื่อพิจารณาต้านสเตเตอร์ | 6 |
| รูปที่ 2.5 วงจรสมมูลย์ทางไฟฟ้าต่อเฟสของมอเตอร์เหนี่ยวนำเมื่อพิจารณาค่า core loss คงที่และนำไปติดกับการสูญเสียทางกล | 6 |
| รูปที่ 2.6 วงจรสมมูลย์ทางไฟฟ้าต่อเฟสของมอเตอร์เหนี่ยวนำที่ใช้พิจารณาค่ากำลัง ไฟฟ้า | 7 |
| รูปที่ 2.7 ขั้นตอนการพิจารณาค่ากำลังไฟฟ้าของมอเตอร์เหนี่ยวนำ | 8 |
| รูปที่ 2.8 กราฟคุณลักษณะของมอเตอร์เหนี่ยวนำ 3 เฟส | 8 |
| รูปที่ 2.9 วงจรทดสอบมอเตอร์เหนี่ยวนำในลักษณะ No-load | 9 |
| รูปที่ 2.10 วงจรทดสอบมอเตอร์เหนี่ยวนำในลักษณะ Blocked-rotor | 10 |
| รูปที่ 2.11 การควบคุมความเร็วมอเตอร์เหนี่ยวนำโดยเปลี่ยนความถี่ของ ไฟฟ้า | 11 |
| รูปที่ 2.12 ความสัมพันธ์ของค่าแรงบิดและความเร็วของมอเตอร์ โดยการ เปลี่ยนค่าความถี่ และแรงดันไฟฟ้าของระบบไฟฟ้า | 12 |
| รูปที่ 2.13 กราฟความสัมพันธ์ของแรงบิดและความเร็วเมื่อ ปรับความเร็วมอเตอร์โดยการเปลี่ยนค่าสลิป | 12 |
| รูปที่ 2.14 แสดงบล็อก ไดอะแกรมของระบบอินเวอร์เตอร์เพื่อการขับนำมอเตอร์เหนี่ยวนำ | 13 |
| รูปที่ 2.15 ชนิดของ variable-frequency converters | 14 |
| รูปที่ 2.16 โครงสร้างและรูปคลื่น PWM-VSI | 15 |
| รูปที่ 2.17 ลักษณะสัญญาณและฮาร์มอนิกส์ที่เกิดขึ้นของ SQUARE-WAVE VSI | 16 |
| รูปที่ 2.18 วงจรและลักษณะสัญญาณในจุดมคติของCSI | 17 |
| รูปที่ 2.19 บล็อก ไดอะแกรมการหาค่าฟลัก | 18 |
| รูปที่ 2.20 บล็อก ไดอะแกรมหาค่าแรงบิด | 19 |
| รูปที่ 2.21 ระบบเปลี่ยนปริมาณทางธรรมชาติเป็นปริมาณดิจิทัล | 20 |
| รูปที่ 2.22 แสดงวิธีพื้นฐานของ ADC | 21 |
| รูปที่ 2.23 แสดงทรานเฟอร์ฟังก์ชันของคอมพาราเตอร์ | 21 |

สารบัญรูป (ต่อ)

| | หน้า |
|---|------|
| รูปที่ 2.24 วิธีประมาณค่าสัญญาณ | 22 |
| รูปที่ 2.25 การสุ่มสัญญาณ | 24 |
| รูปที่ 2.26 แสดงวงจรพื้นฐานของ S/H | 25 |
| รูปที่ 2.27 แสดงรูปคลื่นเอาท์พุทของ S/H | 25 |
| รูปที่ 2.28 แสดงบล็อกไดอะแกรมของระบบคอมพิวเตอร์ | 26 |
| รูปที่ 2.29 แสดงการอินเตอร์เฟซคอมพิวเตอร์ | 27 |
| รูปที่ 2.30 แสดงระบบการรับส่งข้อมูลแบบขนาน | 28 |
| รูปที่ 2.31 บล็อกไดอะแกรม โครงสร้างของพอร์ตนาน | 29 |
| รูปที่ 2.32 ขาสัญญาณของ Connector DB25 | 29 |
| รูปที่ 2.33 การจัดเรียงสัญญาณของพอร์ตนาน | 30 |
| รูปที่ 2.34 สัญญาณเวลาในการติดต่อกับเครื่องพิมพ์ | 32 |
| รูปที่ 2.35 แสดงแผนภาพการติดต่อใน Nibble Mode | 34 |
| รูปที่ 2.36 แสดงแผนภาพการติดต่อใน Byte Mode | 35 |
| รูปที่ 2.37 แผนภาพเวลาแสดงการทำงานใน EPP Mode (วงรอบการเขียน) | 37 |
| รูปที่ 2.38 แผนภาพเวลาแสดงการทำงานใน ECP โหมด | 38 |
| รูปที่ 2.39 แผนภาพเวลาแสดงการทำงานในโหมด ECP | 39 |
| รูปที่ 3.1 วงจรตรวจจับแรงดัน | 41 |
| รูปที่ 3.2 วงจรตรวจจับกระแส | 42 |
| รูปที่ 3.3 โครงสร้างภายนอกของ LEM | 42 |
| รูปที่ 3.4 ลักษณะของ ADC0820 | 43 |
| รูปที่ 3.5 การต่อ ADC0820 | 43 |
| รูปที่ 3.6 ลักษณะของ ET-PC 8255 | 44 |
| รูปที่ 3.7 ลักษณะของ IC 8255 | 44 |
| รูปที่ 3.8 แสดงรายวงจรตรวจจับแรงดัน | 45 |
| รูปที่ 3.9 แสดงรายวงจรตรวจจับกระแส | 45 |
| รูปที่ 3.10 แสดงรายวงจร ADC0820 | 46 |
| รูปที่ 3.11 แสดงรูปโครงงานที่สร้างขึ้น | 46 |
| รูปที่ 3.12 แสดงรูปโครงงานที่สร้างขึ้นกับอินเวอร์เตอร์ที่ใช้ | 47 |
| รูปที่ 3.13 แสดงรูปลักษณะ โครงงานทั้งหมด | 47 |
| รูปที่ 3.14 แสดง โครงสร้างของโครงงาน | 48 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

| | หน้า |
|--|------|
| รูปที่ 4.1 แสดงวงจรจำลองตรวจจับกระแสชาयน์ | 49 |
| รูปที่ 4.2 แสดงรูปคลื่นของวงจรจำลองตรวจจับกระแสชาयน์ | 49 |
| รูปที่ 4.3 แสดงรูปคลื่นของวงจรตรวจจับกระแสชาयน์ | 50 |
| รูปที่ 4.4 แสดงวงจรจำลองตรวจจับกระแสที่ดับบีวเอ็ม(PWM) | 51 |
| รูปที่ 4.5 แสดงรูปคลื่นของวงจรจำลองตรวจจับกระแสที่ดับบีวเอ็ม(PWM) | 51 |
| รูปที่ 4.6 แสดงรูปคลื่นของวงจรตรวจจับกระแส PWM และสัญญาณที่ออกจาก A/D | 52 |
| รูปที่ 4.7 แสดงวงจรจำลองตรวจจับแรงดันชาयน์ | 52 |
| รูปที่ 4.8 แสดงรูปคลื่นของวงจรจำลองตรวจจับแรงดันชาयน์ | 53 |
| รูปที่ 4.9 แสดงรูปคลื่นของวงจรตรวจจับแรงดันชาयน์ | 54 |
| รูปที่ 4.10 แสดงวงจรจำลองตรวจจับแรงดันที่ดับบีวเอ็ม(PWM) | 54 |
| รูปที่ 4.11 แสดงรูปคลื่นของวงจรจำลองตรวจจับแรงดันดับบีวเอ็ม(PWM) | 55 |
| รูปที่ 4.12 แสดงรูปคลื่นของวงจรตรวจจับแรงดันดับบีวเอ็ม(PWM) และสัญญาณที่ออกจาก A/D | 55 |
| รูปที่ 4.13 แสดงรูปคลื่นสัญญาณของ ADC0820 ที่ 2.46 V | 56 |
| รูปที่ 4.14 แสดงรูปคลื่นสัญญาณของ ADC0820 ที่ 2.01 V | 56 |
| รูปที่ 4.15 สัญญาณแรงดันชาयน์ก่อนเข้า ADC0820 | 57 |
| รูปที่ 4.16 สัญญาณกระแสชาयน์ก่อนเข้า ADC0820 | 57 |
| รูปที่ 4.17 สัญญาณแรงดันที่ดับบีวเอ็ม(PWM)ก่อนเข้า ADC0820 | 58 |
| รูปที่ 4.18 สัญญาณกระแสที่ดับบีวเอ็ม(PWM)ก่อนเข้า ADC0820 | 58 |
| รูปที่ 4.19 แสดงบล็อกไดอะแกรมของมอเตอร์ที่ใช้สัญญาณชาयน์ | 59 |
| รูปที่ 4.20 แสดงบล็อกไดอะแกรมของมอเตอร์ที่ใช้สัญญาณที่ดับบีวเอ็ม(PWM) | 59 |
| รูปที่ 4.21 แสดงบล็อกไดอะแกรมการหาค่าฟลักซ์(ψ_d)และฟลักซ์คว(ψ_q) | 60 |
| รูปที่ 4.22 แสดงบล็อกไดอะแกรมการหาค่ากำลังไฟฟ้า | 60 |
| รูปที่ 4.23 แสดงบล็อกไดอะแกรมการหาค่าแรงบิด | 61 |
| รูปที่ 4.24 แสดงสัญญาณแรงดันABCที่ได้ทางคอมพิวเตอร์ | 61 |
| รูปที่ 4.25 แสดงสัญญาณแรงดันABCที่ได้จากการจำลองใน Matlab | 62 |
| รูปที่ 4.26 แสดงสัญญาณกระแสABCที่ได้ทางคอมพิวเตอร์ | 62 |
| รูปที่ 4.27 แสดงสัญญาณแรงดัน V_d, V_q ที่ได้ทางคอมพิวเตอร์ | 63 |
| รูปที่ 4.28 แสดงสัญญาณแรงดัน V_d, V_q ที่ได้จากการจำลองใน Matlab | 63 |
| รูปที่ 4.29 แสดงสัญญาณกระแส I_d, I_q ที่ได้ทางคอมพิวเตอร์ | 64 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

| | หน้า |
|--|------|
| รูปที่ 4.30 แสดงสัญญาณกระแส I_d, I_q ที่ได้จากการจำลองใน Matlab | 64 |
| รูปที่ 4.31 แสดงสัญญาณฟลักซ์คิวิ (ψ_d, ψ_q) ที่ได้ทางคอมพิวเตอร์ | 65 |
| รูปที่ 4.32 แสดงสัญญาณฟลักซ์คิวิ (ψ_d, ψ_q) ที่ได้จากการจำลองใน Matlab | 65 |
| รูปที่ 4.33 แสดงสัญญาณกำลังไฟฟ้าที่ได้ทางคอมพิวเตอร์(ขณะไม่มีโหลด) | 66 |
| รูปที่ 4.34 แสดงสัญญาณกำลังไฟฟ้าที่ได้ทางคอมพิวเตอร์(ขณะมีโหลด 26 นิวตัน) | 66 |
| รูปที่ 4.35 แสดงสัญญาณกำลังไฟฟ้าที่ได้จากการจำลองใน Matlab | 67 |
| รูปที่ 4.36 แสดงสัญญาณแรงบิดที่ได้ทางคอมพิวเตอร์(ขณะไม่มีโหลด) | 68 |
| รูปที่ 4.37 แสดงสัญญาณแรงบิดที่ได้ทางคอมพิวเตอร์(ขณะมีโหลด 26 นิวตัน) | 68 |
| รูปที่ 4.38 แสดงสัญญาณแรงบิดที่ได้จากการจำลองใน Matlab | 69 |
| รูปที่ 4.39 แสดงสัญญาณแรงดันพีดับบีวเอ็ม(PWM)ที่ได้ทางคอมพิวเตอร์ | 70 |
| รูปที่ 4.40 แสดงสัญญาณแรงดันพีดับบีวเอ็ม(PWM)ที่ได้จากการจำลองใน Matlab | 70 |
| รูปที่ 4.41 แสดงสัญญาณกระแสพีดับบีวเอ็ม(PWM)ที่ได้ทางคอมพิวเตอร์ | 71 |
| รูปที่ 4.42 แสดงสัญญาณกระแสพีดับบีวเอ็ม(PWM)ที่ได้จากการจำลองใน Matlab | 71 |
| รูปที่ 4.43 แสดงสัญญาณแรงดัน (V_d, V_q) พีดับบีวเอ็มที่ได้ทางคอมพิวเตอร์ | 72 |
| รูปที่ 4.44 แสดงสัญญาณแรงดัน (V_d, V_q) พีดับบีวเอ็มที่ได้จากการจำลองใน Matlab | 72 |
| รูปที่ 4.45 แสดงสัญญาณกระแส (I_d, I_q) พีดับบีวเอ็มที่ได้ทางคอมพิวเตอร์ | 73 |
| รูปที่ 4.46 แสดงสัญญาณกระแส (I_d, I_q) พีดับบีวเอ็มที่ได้จากการจำลองใน Matlab | 73 |
| รูปที่ 4.47 แสดงสัญญาณฟลักซ์คิวิ (ψ_d, ψ_q) ที่ได้ทางคอมพิวเตอร์ | 74 |
| รูปที่ 4.48 แสดงสัญญาณฟลักซ์คิวิ (ψ_d, ψ_q) ที่ได้จากการจำลองใน Matlab | 74 |
| รูปที่ 4.49 แสดงสัญญาณกำลังไฟฟ้าพีดับบีวเอ็ม(PWM) ที่ได้ทางคอมพิวเตอร์ | 75 |
| รูปที่ 4.50 แสดงสัญญาณกำลังไฟฟ้าพีดับบีวเอ็ม(PWM) ที่ได้จากการจำลองใน Matlab | 75 |
| รูปที่ 4.51 แสดงสัญญาณแรงบิดพีดับบีวเอ็ม(PWM) ที่ได้ทางคอมพิวเตอร์ | 76 |
| รูปที่ 4.52 แสดงสัญญาณแรงบิดพีดับบีวเอ็ม(PWM) ที่ได้จากการจำลองใน Matlab | 76 |

สารบัญตาราง

| | หน้า |
|---|------|
| ตารางที่ 2.1 รายละเอียดขาสัญญาณ Connector ของพอร์ทขนาน | 30 |
| ตารางที่ 2.2 รีจิสเตอร์ข้อมูล (Data) ตำแหน่ง 378H (LPT1),278H(LPT2) | 31 |
| ตารางที่ 2.3 สัญญาณใน Nibble Mode และหน้าที่การทำงาน | 33 |
| ตารางที่ 2.4 แสดงสัญญาณใน Byte Mode โหมดและหน้าที่การทำงาน | 34 |
| ตารางที่ 2.5 แสดงสัญญาณเปรียบเทียบใน EPP โหมด | 36 |
| ตารางที่ 2.6 แสดงสัญญาณเปรียบเทียบใน ECP โหมด | 38 |



บทที่ 1

บทนำ

การบันทึกข้อมูลและการแสดงผลสำหรับอินเวอร์เตอร์ขับเคลื่อนมอเตอร์เหนี่ยวนำ 3 เฟส เป็นโครงการที่ทำการศึกษเกี่ยวกับนำสัญญาณของกระแสและสัญญาณของแรงดันจากมอเตอร์เหนี่ยวนำมาวิเคราะห์และประมวลผล เพื่อหาค่าต่างๆ ที่เกี่ยวข้องทางวงจรของมอเตอร์และหาค่าผลของมอเตอร์ที่เกิดขึ้น โดยในโครงการนี้จะทำการประมวลผลโดยนำสัญญาณกระแสและแรงดัน มาหาค่าแรงบิดของมอเตอร์และค่ากำลังไฟฟ้าของมอเตอร์ โดยอาศัยหลักการ ของการ ตรวจจับสัญญาณกระแสและแรงดันด้วยวงจรตรวจจับสัญญาณจากนั้นนำมาประมวลผลเพื่อหาค่า

โดยในการหาค่าแรงบิดนั้นจะอาศัยหลักการของการตรวจจับสัญญาณ โดยสัญญาณที่ได้จะมีค่าเป็น 3 เฟสจากนั้นจะใช้สมการเปลี่ยนค่าของกระแสและแรงดันจาก 3 เฟสให้มีค่าเป็นค่ากระแสและแรงดันที่มีค่า 2 แกน ที่มีมุมต่างเฟสกัน 90 องศา นำไปวิเคราะห์เพื่อหาค่าของแรงบิดที่เกิดขึ้น การหาค่าของกำลังไฟฟ้าโดยนำกระแสและแรงดัน ไปวิเคราะห์และประมวลผลเพื่อหาผลของกำลังไฟฟ้าที่เกิดขึ้นกับมอเตอร์โดยอาศัยสมการทางไฟฟ้าในการพิจารณา จากสัญญาณของกระแสและสัญญาณของแรงดันเราสามารถตรวจจับได้โดยอาศัยวงจรทางอิเล็กทรอนิกส์ที่ออกแบบไว้ แรงบิดและกำลังไฟฟ้าเราสามารถหาโดยอาศัยสมการทางวิศวกรรมไฟฟ้ามาวิเคราะห์และประมวลผลได้ ข้อมูลทั้งหมดเราจะนำมาแสดงค่าโดยจะแสดงที่หน้าจอคอมพิวเตอร์ โดยอาศัยการเขียน โปรแกรมภาษา C++ เพื่อรับค่าสัญญาณกระแสและแรงดันที่เป็นแบบดิจิทัลเพื่อนำมาประมวลผลและแสดงผลที่จอคอมพิวเตอร์

โครงการนี้เป็นชิ้นงานต้นแบบที่ใช้หลักการวิเคราะห์สัญญาณและประมวลผลสัญญาณโดยอาศัยคอมพิวเตอร์และสามารถแสดงผลได้ที่จอคอมพิวเตอร์ เพื่อให้เกิดความสะดวกในการวัดค่าต่างๆ ที่เกิดขึ้นกับอินดักชันมอเตอร์โดยไม่ต้องเข้าไปวัดค่าโดยตรง

วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาคูณลักษณะของสัญญาณกระแสและสัญญาณแรงดันของอินดักชันมอเตอร์ 3 เฟส
2. วิเคราะห์สัญญาณกระแสและสัญญาณแรงดันของอินดักชันมอเตอร์ เพื่อนำมาวิเคราะห์และประมวลผลหาค่า แรงบิดและกำลังไฟฟ้าของอินดักชันมอเตอร์
3. เพื่อศึกษาการนำสัญญาณกระแสและสัญญาณแรงดันนำไปประมวลผลเพื่อให้แสดงที่หน้าจอคอมพิวเตอร์
4. เพื่อให้เกิดความสะดวกในการวัดสัญญาณ โดยไม่ต้องเข้าไปวัดเพียงแต่อยู่ที่หน้าจอคอมพิวเตอร์ก็สามารถวัดสัญญาณที่จุดต่างๆ ได้
5. เพื่อเป็นแนวทางในการศึกษาการนำสัญญาณต่างๆ มาประมวลผลและประยุกต์ใช้ในการควบคุมอุปกรณ์ระยะไกล หรือที่มีอันตราย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีที่เกี่ยวข้องกับโครงการงาน

2.1 ทฤษฎีของมอเตอร์เหนี่ยวนำ

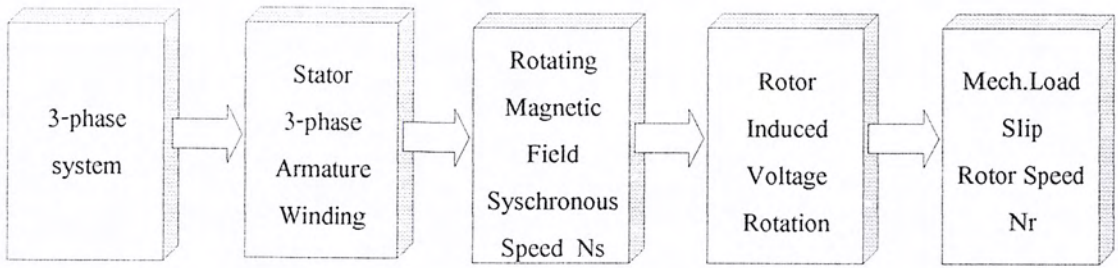
มอเตอร์ที่เกิดจากการเหนี่ยวนำนี้อาจเป็นแบบเฟสเดียว หรือหลายเฟสก็ได้โดยปกติแล้วมอเตอร์ที่เกิดจากการเหนี่ยวนำชนิดหลายเฟส (Poly phase induction motor) มักจะหมายถึงแบบ 3 เฟส ซึ่งแบ่งออกได้เป็น 2 ชนิดคือ

1.แบบที่มีตัวหมุนคล้ายกรงกระรอก (squirrel-cage rotor) ซึ่งเป็นแบบที่ไม่ต้องการใช้วงแหวน (slip ring) ติดอยู่ที่ตัวหมุน

2.แบบที่มีตัวหมุนมีขดลวดพันอยู่รอบๆ (wound rotor) โดยที่ขั้วแม่เหล็กจะเกิดขึ้นบนตัวหมุนนี้จะต้องมีจำนวนขั้วเท่ากับที่มีอยู่บนตัวอยู่กับที่ (stator) แบบนี้ต้องใช้วงแหวนติดไว้ที่บนตัวหมุน เพื่อนำปลายของขดลวดที่พันอยู่บนตัวหมุนนั้น ต่อออกมายังวงจรรภายนอก หรือเรียกอีกชื่อหนึ่งว่า “Slip ring”

2.1.1 หลักการทำงาน (Principle of Operation)

มอเตอร์เหนี่ยวนำนี้ ไม่มีวงจรไฟฟ้าต่อเชื่อมกันระหว่างตัวอยู่กับที่กับตัวหมุน กระแสที่เกิดขึ้นในตัวหมุน ที่ไหลอยู่ภายในแท่งตัวนำที่ฝังอยู่รอบๆ ตัวหมุนนั้น เกิดจากแรงเคลื่อนไฟฟ้าที่เกิดขึ้นภายในตัวหมุนนั้น แรงเคลื่อนไฟฟ้าที่เกิดขึ้นนี้เกิดจากสนามแม่เหล็กหมุน ที่มาจากตัวอยู่กับที่ ขนาดของแรงเคลื่อนขึ้นอยู่กับ อัตราการ เปลี่ยนแปลงของเส้นแรงแม่เหล็กที่เกิดขึ้นบนตัวนำของ (rotor) ปฏิกิริยาการต่างๆ ที่เกิดขึ้นนี้มีหลักการคล้ายคลึงกับหม้อแปลงที่มีส่วนที่สองสามารถเคลื่อนที่ได้ สนามแม่เหล็กพัทธ์ที่เกิดขึ้นบนตัวอยู่กับที่นี้ จะหมุนคล้ายกับการหมุนของแม่เหล็กถาวรด้วยความเร็วเท่ากับ (synchronous) เมื่อมีสนามแม่เหล็กวิ่งผ่านตัวนำบน rotor ทำให้เกิดการเหนี่ยวนำแรงเคลื่อนไฟฟ้าขึ้นที่ตัวนำใน rotor แรงเคลื่อนนี้จะทำให้เกิดกระแสไหลในตัวนำ ซึ่งทำให้เกิดแรงบิด (Torque) ขึ้นที่ตัวหมุน ในทิศทางเดียวกับการเคลื่อนที่ของสนามแม่เหล็กหมุน ทิศทางของแรงเคลื่อนไฟฟ้านั้นหาได้โดยใช้กฎมือขวาของเฟรมมิ่ง ส่วนทิศทางการเคลื่อนที่หาได้จากกฎมือซ้ายของเฟรมมิ่ง



รูปที่ 2.1 หลักการทำงานของมอเตอร์เหนี่ยวนำ 3 เฟส

เมื่อขดลวด stator ได้รับแรงดันไฟฟ้า 3 เฟส จะสร้างสนามแม่เหล็กหมุนขึ้นมา เคลื่อนที่หมุนไปรอบช่องอากาศด้วยความเร็วเท่ากับ synchronous speed N_s ดังสมการ

$$\text{Synchronous speed } N_s = \frac{120f}{p} \text{ rpm} \quad (2-1)$$

Slip คือ อัตราส่วนของความเร็วที่เกิดจากความแตกต่างของความเร็วที่ rotor กับความเร็วที่ synchronous คือความเร็วที่ synchronous ของสนามแม่เหล็กหมุนที่เกิดขึ้นในเครื่องจักร

$$S = \frac{N_s - N_r}{N_s} \quad (2-2)$$

เมื่อ

$$S = \text{Slip}$$

$$N_s = \text{Synchronous speed (rpm)}$$

$$N_r = \text{rotor speed (rpm)}$$

$$\text{Slip speed } sN_s = N_s - N_r \text{ (rpm)} \quad (2-3)$$

$$\text{Rotor speed } N_r = (1-S)N_s \text{ (rpm)} \quad (2-4)$$

$$\text{Rotor frequency } f_r = sf \text{ (Hz)} \quad (2-5)$$

ขณะเริ่มหมุน (starting) rotor หยุดนิ่งอยู่กับที่ $s = 1$

rotor frequency = stator frequency

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 วงจรสมมูลย์ของมอเตอร์เหนี่ยวนำ(Three-Phase Induction Motor Equivalent Circuit)

เราจะสร้างแบบจำลองที่แทนคุณลักษณะของเครื่องจักรไฟฟ้าเหนี่ยวนำ ในรูปของวงจรไฟฟ้า โดยอาศัยทฤษฎีพื้นฐานทางแม่เหล็กไฟฟ้าและวงจรไฟฟ้าสร้างขึ้นมา

Rotor Equipvalent Circuit (per phase)

จากสมการที่ 2-5 จะเห็นได้ว่าค่าความถี่ของ rotor จะมีค่าเปลี่ยนไปตามค่า slip ดังนั้นค่าพารามิเตอร์ต่างๆทางไฟฟ้าที่ rotor จะขึ้นอยู่กับค่า slip

เมื่อมอเตอร์ได้รับ โหลดจะทำให้เกิดกระแสไหลภายในตัวหมุน และทำให้ความเร็วของตัวหมุนช้าลงไป เมื่อเป็นเช่นนั้นย่อมทำให้ค่า slip เพิ่มขึ้นด้วย นั่นคือทำให้แรงเคลื่อนไฟฟ้าที่ตัวหมุนเพิ่มขึ้น เป็น sE_2 ส่วนค่า leakage reactance X_2 ก็เปลี่ยนเป็น sX_2 ส่วนค่า R_2 ไม่ขึ้นกับการเปลี่ยนแปลงของ slip

$$X_{2s} = sX_2 \quad (2-6)$$

$$E_{2s} = sE_2 \quad (2-7)$$

เมื่อ X_{2s} = ค่า leakage reactance ของ rotor ที่ค่า slip ใดๆ

X_2 = ค่า leakage reactance ของ rotor ขณะหยุดนิ่ง

E_{2s} = ค่า induced voltage ของ rotor ที่ค่า slip ใดๆ

E_2 = ค่า induced voltage ของ rotor ขณะหยุดนิ่ง

วงจรสมมูลย์ทางไฟฟ้า per phase ของ rotor สามารถแสดงได้ดังนี้



รูปที่ 2.2 วงจรสมมูลย์ทางไฟฟ้าต่อเฟส ของโรเตอร์ 2 รูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

R_2 = ค่า resistance ของ rotor

จากรูป rotor current

$$I_2 = \frac{sE_2}{\sqrt{R_2^2 + (sX_2)^2}} \quad (2-8)$$

จัดรูปสมการที่ 2-6 ใหม่ จะได้

$$I_2 = \frac{E_2}{\sqrt{(R_2/s)^2 + (X_2)^2}} \quad (2-9)$$

แสดงเป็นวงจรสมมูลย์ได้ดังรูป ข)

∴ ความต้านทานของวงจรคือ

$$\begin{aligned} \frac{R_2}{s} &= R_2 + \frac{R_2}{s} - R_2 \\ &= R_2 + R_2 \frac{(1-s)}{s} \end{aligned} \quad (2-10)$$

ซึ่งประกอบด้วยค่าความต้านทาน 2 ส่วนคือ

1. ส่วนแรกเป็นส่วนของความต้านทานของ rotor คือ R_2 แทนการสูญเสียของเส้นลวดใน rotor (Cu loss)
2. ส่วนที่สองคือค่า $\frac{R_2(1-s)}{s}$ คือ load ความต้านทาน R_L ที่สมมูลย์ load ทางไฟฟ้าแทน load ทางกลของมอเตอร์นั่นเอง

Stator Equivalent Circuit (per phase)

เนื่องจากมอเตอร์เหนี่ยวนำอาศัยหลักการในการทำงานเช่นเดียวกับหม้อแปลง มอเตอร์มีส่วนที่เคลื่อนที่ได้ rotor ดังนั้นจึงเขียนวงจรสมมูลย์ได้เช่นเดียวกับหม้อแปลง

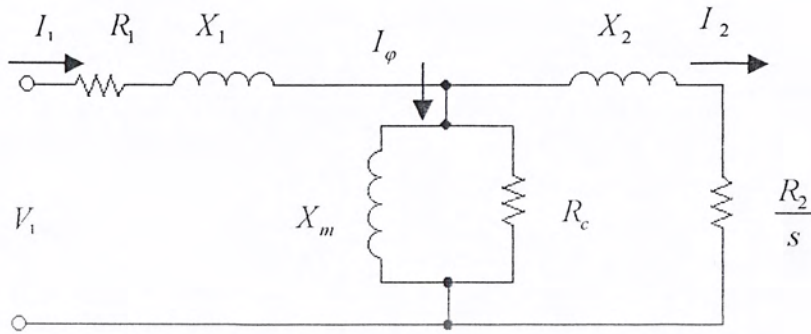
ต่างกันตรงที่



รูปที่ 2.3 วงจรสมมูลย์ทางไฟฟ้าต่อเฟสของสเตเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการพิจารณารวมวงจรสมมูลระหว่าง rotor และ stator ของ induction motor ดังรูป กำหนดให้ค่า R_2 และ X_2 เป็นค่าที่พิจารณาอยู่ด้าน stator แล้ว



รูปที่ 2.4 วงจรสมมูลทางไฟฟ้าต่อเฟสของมอเตอร์เหนี่ยวนำเมื่อพิจารณาด้านสเตเตอร์

กระแสกระตุ้น I_ϕ สามารถแยกออกได้สองส่วนคือ กระแสส่วนที่สูญเสียในแกนเหล็ก I_c ซึ่งจะมีเฟสเดียวกับ E_1 และกระแสส่วนสร้างแม่เหล็ก I_m ซึ่งจะล้าหลัง E_1 เป็นมุม 90° ในวงจรสมมูล กระแสกระตุ้น I_ϕ สามารถแสดงไว้ในส่วนวงจรรวมที่ประกอบด้วยความต้านทาน R_c และค่าแมกนิตีซิ่ง รีแอกแตนซ์ X_m

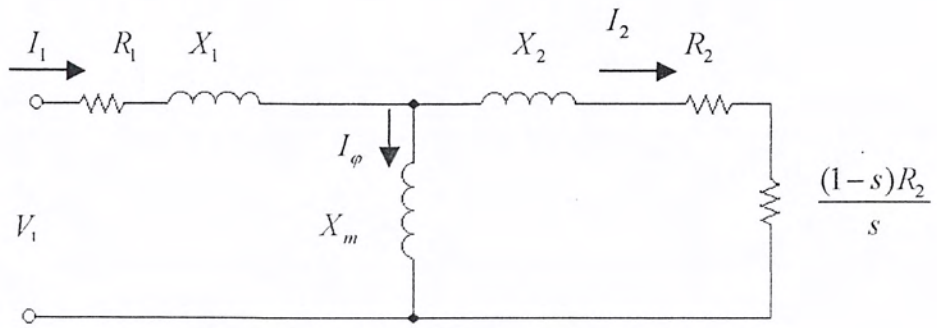
2.1.3 กำลังและแรงบิด (Power and Torque Characteristics)



รูปที่ 2.5 วงจรสมมูลทางไฟฟ้าต่อเฟสของมอเตอร์เหนี่ยวนำเมื่อพิจารณาค่า core loss คงที่และนำไปคิดรวมกับการสูญเสียทางกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจากสมการที่ 2-8 สามารถเขียนวงจรสมมูลใหม่ได้ดังนี้



รูปที่ 2.6 วงจรสมมูลทางไฟฟ้าต่อเฟสของมอเตอร์เหนี่ยวนำที่ใช้พิจารณาค่ากำลังไฟฟ้า

สมการแรงบิดและกำลังไฟฟ้าของมอเตอร์เหนี่ยวนำสามารถพิสูจน์ได้จากวงจรสมมูล

กำลังสูญเสียเนื่องจากความต้านทานใน stator (Stator copper loss)

$$P_{cu,1} = 3I_1^2 R_1 \quad (2-11)$$

กำลังสูญเสียเนื่องจากความต้านทานใน rotor (Rotor copper loss)

$$P_{cu,2} = 3I_2^2 R_2 \quad (2-12)$$

กำลังไฟฟ้าที่ป้อน (Input power)

$$P_i = 3V_1 I_1 \cos \phi_1 \quad (2-13)$$

Rotor input (power across air gap)

$$P_g = 3I_2^2 \frac{R_2}{s} \quad (2-14)$$

Rotor output (electromagnetic power)

$$P_e = P_g - P_{cu,2} = (1-s)P_g \quad (2-15)$$

Electromagnetic torque

$$T_e = \frac{P_e}{\omega_r} = \frac{P_g}{\omega_s} \quad (2-16)$$

Output power (mechanical power)

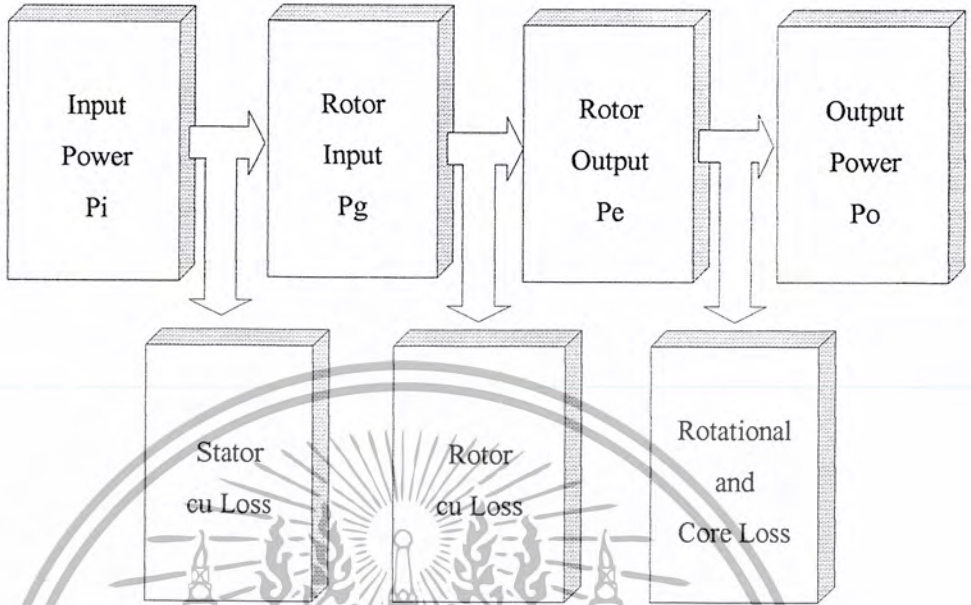
$$P_o = P_e - P_r \quad (2-17)$$

Output torque

$$T_o = \frac{P_o}{\omega_r} \quad (2-18)$$

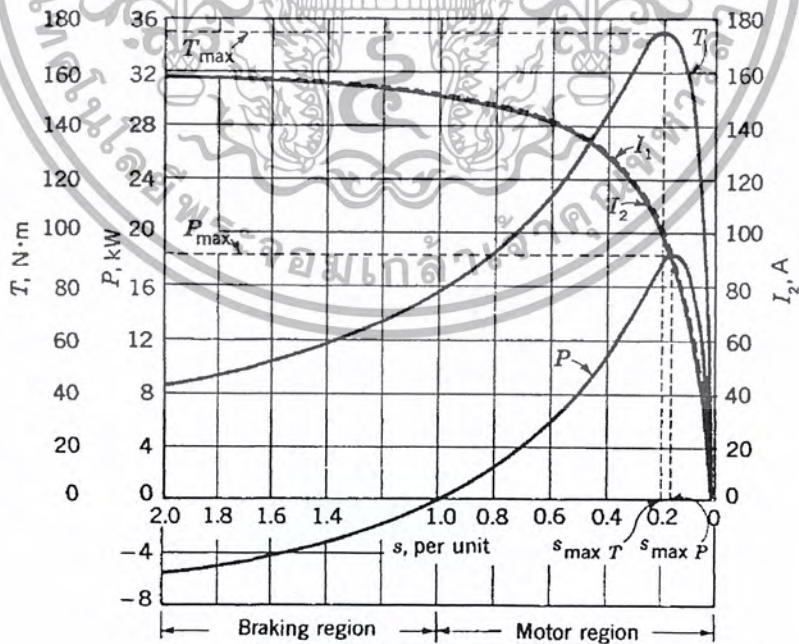
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการพิจารณาค่ากำลังไฟฟ้าของมอเตอร์เหนี่ยวนำ (induction motor)



รูปที่ 2.7 ขั้นตอนการพิจารณาค่ากำลังไฟฟ้าของมอเตอร์เหนี่ยวนำ

เราสามารถหาค่าต่างๆเหล่านี้ที่ค่า slip ต่างๆ ได้โดยการแทนค่า slip แล้วคำนวณตามขั้นตอนข้างต้นเมื่อแทนค่า slip ให้มีค่าเท่ากับ 0 ถึง 1.0 จะได้กราฟดังรูป



รูปที่ 2.8 กราฟคุณลักษณะของมอเตอร์เหนี่ยวนำ 3 เฟส

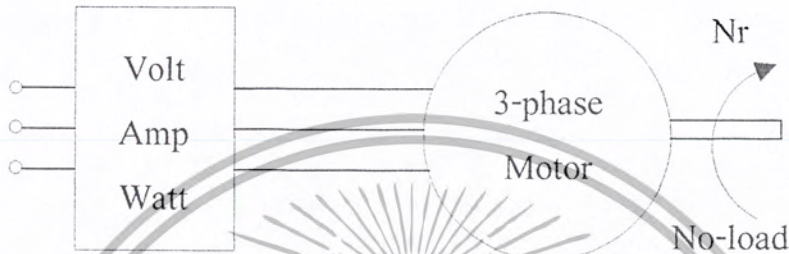
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 การทดสอบมอเตอร์ (Three-Phase Induction Motor Tests)

การทดสอบมอเตอร์เหนี่ยวนำมี 2 วิธี คือ

1. No-Load Test

ทดสอบที่พิกัดแรงดันไฟฟ้าขณะที่ motor อยู่ในสภาพ no-load แล้วบันทึกค่า V,A,W แล้วพิจารณาเป็นค่าต่อเฟส V_o, I_o, P_o



รูปที่ 2.9 วงจรทดสอบมอเตอร์เหนี่ยวนำในลักษณะ No-load

จากการทดสอบสามารถหา ϕ_o, R_c, X_m ได้

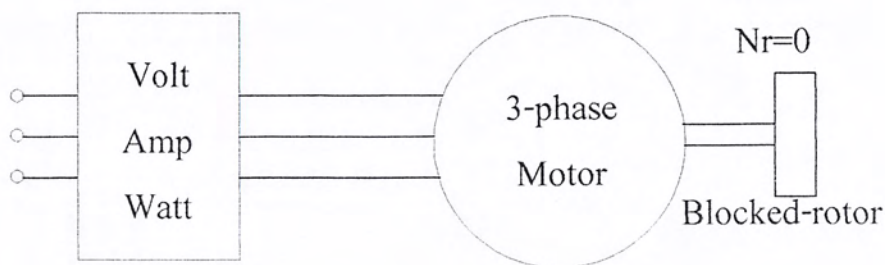
$$\phi_o = \cos^{-1} \frac{P_o}{V_o I_o} \quad (2-19)$$

$$R_c = \frac{V_o^2}{P_o} \quad (2-20)$$

$$X_m = \frac{V_o}{I_o \sin \phi_o} \quad (2-21)$$

2. Blocked-Rotor Test

ทดสอบที่พิกัดกระแสไฟฟ้าในขณะที่ motor อยู่ในสภาพ block-rotor แล้วบันทึกค่า V,A,W แล้วเปลี่ยนให้เป็นค่าต่อเฟส V_b, I_b, P_b



รูปที่ 2.10 วงจรทดสอบมอเตอร์เหนี่ยวนำในลักษณะ Blocked-rotor

จากการทดสอบจะได้

$$\phi_b = \cos^{-1} \frac{P_b}{V_b I_b} \tag{2-22}$$

$$Z_b = \frac{V_b}{I_b} \tag{2-23}$$

$$R_b = \frac{P_b}{I_b^2} \tag{2-24}$$

$$X_b = \sqrt{(Z_b)^2 - (R_b)^2} = \frac{V_b \sin \phi_b}{I_b} \tag{2-25}$$

$$R_b = R_1 + R_2 \tag{2-26}$$

ค่า R_1 (armature winding resistance per phase) สามารถหาได้จากวิธีการ VI-method (DC)

$$X_b = X_1 + X_2 \tag{2-27}$$

Maximum Torque Calculation

Maximum torque สามารถประมาณค่าได้ดังนี้

$$\frac{R_2}{s_{\max,T}} = \sqrt{R_1^2 + (X_1 + X_2)^2} \tag{2-28}$$

ดังนั้นค่า maximum torque จะเกิดขึ้นที่ค่า slip เท่ากับ

$$s_{\max,T} = \frac{R_2}{\sqrt{R_1^2 + (X_1 + X_2)^2}} \tag{2-29}$$

2.1.5 การควบคุมมอเตอร์เหนี่ยวนำ 3 เฟส (Operations of Three-Phase Induction Motors)

สภาวะเริ่มมอเตอร์ (Starting Conditions)

ขณะเริ่มหมุนมอเตอร์จะเกิดกระแสสูงกว่าปกติ พิจารณาเมื่อ slip = 1

$$I_2 = \frac{V_1}{\sqrt{(R_1 + R_2)^2 + (X_1 + X_2)^2}} \quad (2-30)$$

จะเห็นได้ว่าค่าแรงบิดขณะเริ่มหมุนจะพิจารณาได้จาก

$$T_e = \frac{P_g}{\omega_s} = \frac{3I_2^2 R_2}{\omega_s} = \frac{3R_2}{\omega_s} \cdot \frac{V_1^2}{(R_1 + R_2)^2 + (X_1 + X_2)^2} \quad (2-31)$$

ดังนั้นเราสามารถลดค่ากระแสขณะเริ่มหมุนได้โดยพิจารณาจากสมการ 2-30

1. ลดแรงดันไฟฟ้าที่จ่ายให้ motor (แรงบิดขณะเริ่มหมุนจะลดลง)

- Autotransformer

- Star-delta Start

2. เพิ่มค่าความต้านทานไฟฟ้าที่ rotor

การควบคุมความเร็ว (Speed Control)

ความเร็วของ motor ขณะจ่ายภาระทางกลสามารถเปลี่ยนแปลงได้โดย

1. เปลี่ยนค่าความเร็ว synchronous ของ motor

โดยใช้ความสัมพันธ์จากสมการ $N_s = \frac{120f}{P}$ ซึ่งจะเห็นว่าความเร็ว synchronous จะขึ้นอยู่กับ

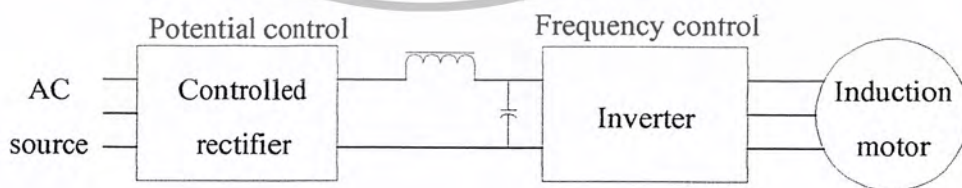
กับจำนวนขั้วแม่เหล็ก และความถี่ของระบบไฟฟ้า

ดังนั้นความเร็ว rotor

$$N_r = (1-s) \frac{120f}{P} \quad (2-32)$$

ซึ่งสามารถทำได้โดยใช้วิธีการ

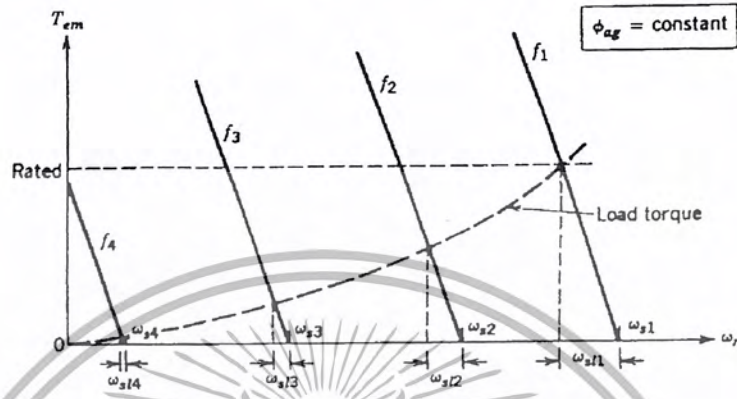
- Pole-changing
- Variable Frequency



รูปที่ 2.11 การควบคุมความเร็วมอเตอร์เหนี่ยวนำโดยเปลี่ยนความถี่ของไฟฟ้า

สำหรับวิธีการเปลี่ยนความถี่ ถ้าต้องการให้ค่าแรงบิดของมอเตอร์ไม่เปลี่ยนแปลง จะต้องทำให้ค่าความหนาแน่นเส้นแรงแม่เหล็กที่ความถี่ต่างๆมีค่าไม่เปลี่ยนแปลง ซึ่งสามารถจะกระทำได้โดยการทำให้อัตราส่วนแรงดันไฟฟ้าต่อความถี่ (V/f) มีค่าคงที่

$$\frac{V}{f} = k_m B_m \tag{2-32}$$

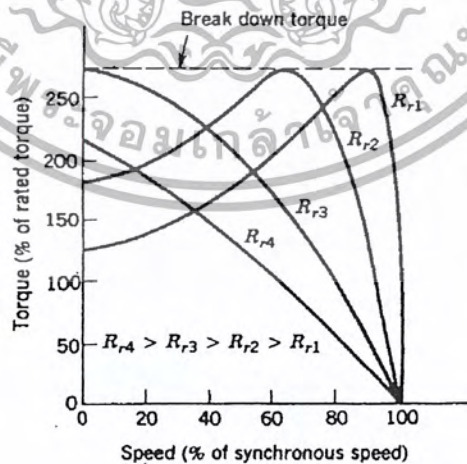


รูปที่ 2.12 ความสัมพันธ์ของค่าแรงบิดและความเร็วของมอเตอร์โดยการเปลี่ยนค่าความถี่ และแรงดันไฟฟ้าของระบบไฟฟ้า

2. การเปลี่ยนค่าสลิป (slip)

การควบคุมสลิปของมอเตอร์จะทำได้โดยการควบคุม torque-speed curve ของมอเตอร์ซึ่งทำได้ 2 วิธีคือ

- Variable Stator Voltage
- Variable Rotor Resistance



รูปที่ 2.13 กราฟความสัมพันธ์ของแรงบิดและความเร็ว

เมื่อปรับความเร็วมอเตอร์โดยการเปลี่ยนค่าสลิป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

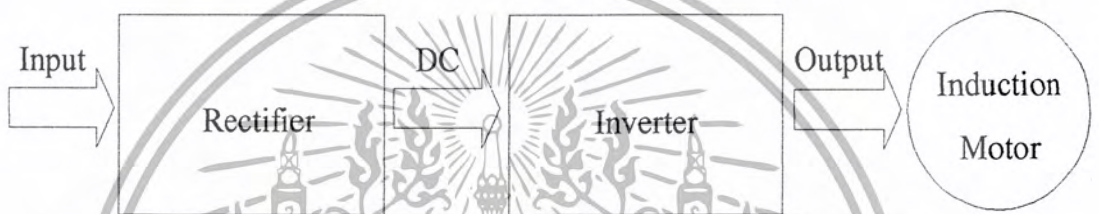
2.1.6 อินเวอร์เตอร์เพื่อการขับเคลื่อนมอเตอร์เหนี่ยวนำ

อินเวอร์เตอร์ที่ใช้ในการควบคุมมอเตอร์เหนี่ยวนำ จะต้องสามารถเปลี่ยนแปลงแรงดันและความถี่ได้ซึ่งจะทำหน้าที่เชื่อม โยระหว่างระบบจ่ายไฟและมอเตอร์ ดังนั้นอินเวอร์เตอร์ที่ดีต้องมีคุณสมบัติพื้นฐานที่สอดคล้องกับความต้องการของมอเตอร์

สามารถที่จะปรับความถี่ได้ เป็นสัดส่วนกับความเร็วรอบที่ต้องการ

สามารถปรับแรงดันเอาต์พุตเพื่อรักษาอัตราส่วนแรงดันต่อความถี่ (V/f) ให้คงที่ตลอดเพื่อให้แรงบิดคงที่

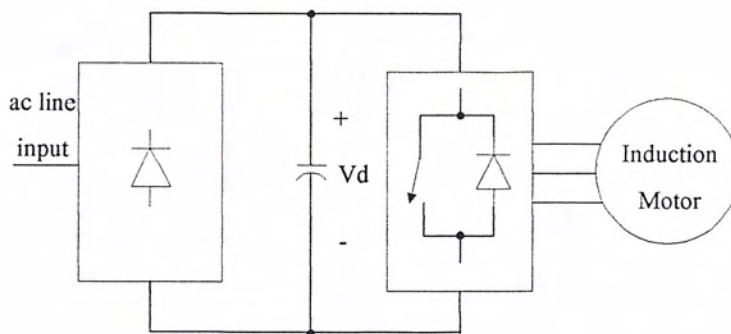
สามารถจ่ายกระแสได้เต็มพิกัดที่ความถี่ใดๆซึ่งอยู่ในช่วง torque คงที่



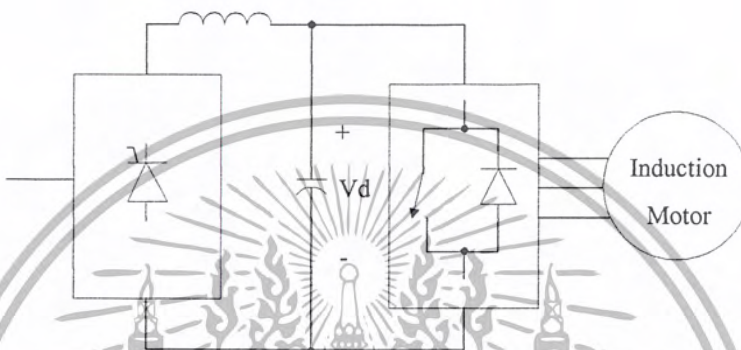
รูปที่ 2.14 แสดงบล็อกไดอะแกรมของระบบอินเวอร์เตอร์เพื่อการขับนำมอเตอร์เหนี่ยวนำ

ชนิดอินเวอร์เตอร์สำหรับ Induction Motor

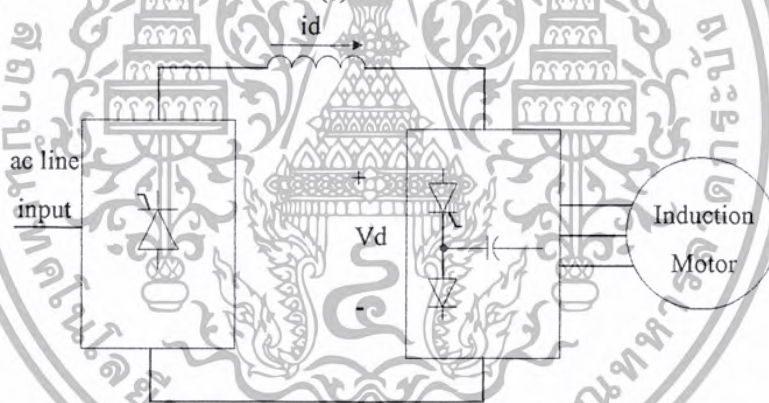
- PWM VSI
- Six step VSI หรือ Quasi VSI
- Current source inverter CSI



(a)



(b)



(c)

รูปที่ 2.15 ชนิดของ *variable-frequency converters*

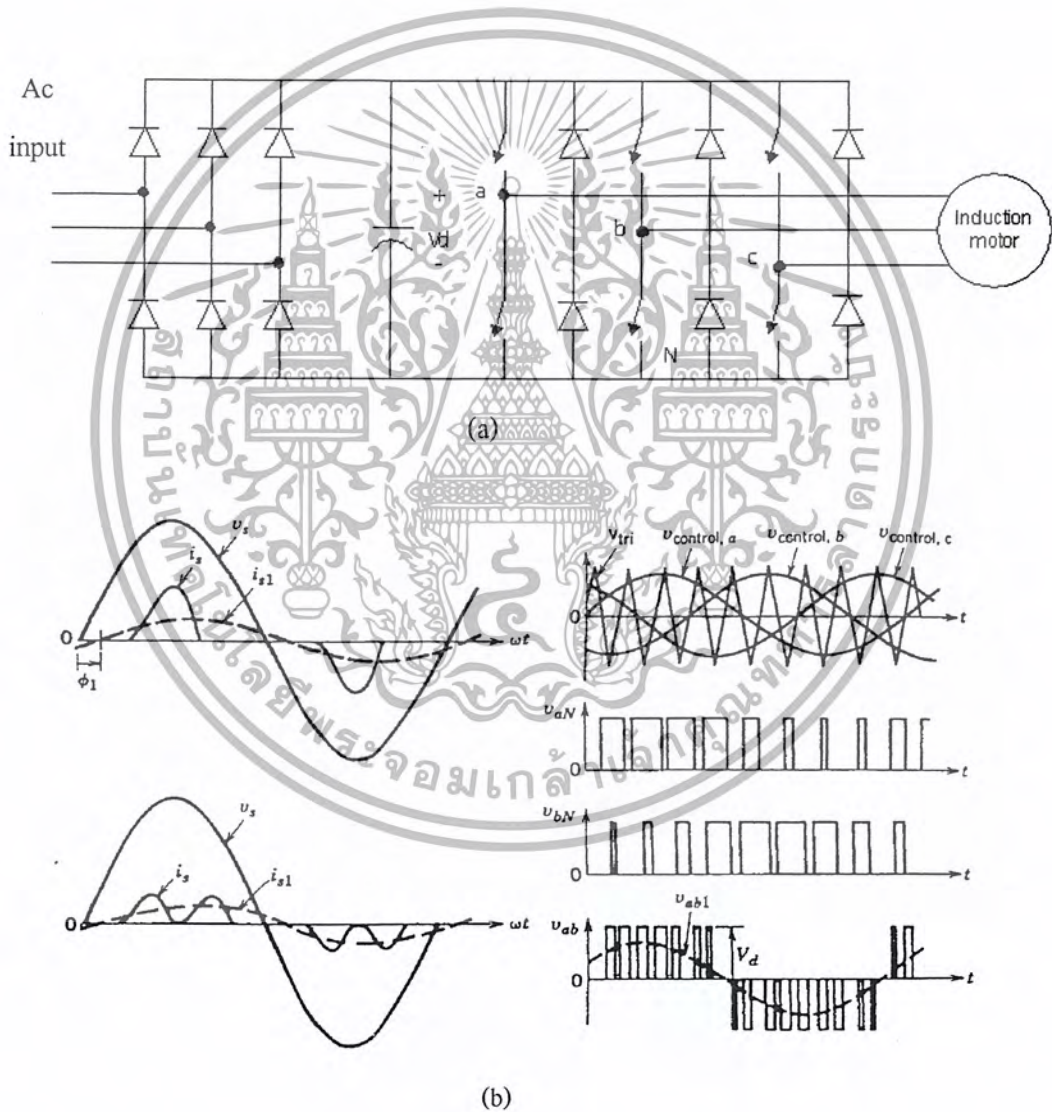
- (a) PWM-VSI กับ ไดโอดเรกติไฟเออร์
- (b) SQUARE-WAVE VSI กับเรกติไฟเออร์แบบควบคุมได้
- (c) CSI กับเรกติไฟเออร์แบบควบคุมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PWM-VSI

โครงสร้างของ PWM-VSI ในส่วนของแรงดันดีซีลิงค์สามารถใช้เรกติไฟเออร์ที่เป็นไดโอด ทำให้แรงดันของดีซีลิงค์มีค่าคงที่ซึ่งไม่สามารถควบคุมได้ ทำให้ค่า power factor ทางด้านอินพุตดีขึ้น ดังนั้น ความถี่และแรงดันเอาต์พุตของแรงดันมูลฐานสามารถควบคุมโดยใช้เทคนิคของ PWM เทคนิค PWM-VSI นี้จะให้แรงดันเอาต์พุตที่มีองค์ประกอบฮาร์มอนิก (harmonic) อันดับค่าแต่จำนวนครั้งของการ switching สูงขึ้นดังนั้นจึงมีปัญหาการกระเพื่อมของแรงบิด

แม้ว่า PWM-VSI สามารถลดการสูญเสียของมอเตอร์เนื่องจากผลของฮาร์มอนิก (harmonic) ก็ตามแต่ประสิทธิภาพของอินเวอร์เตอร์ค่อนข้างจะต่ำเพราะผลของจำนวน switching สูง

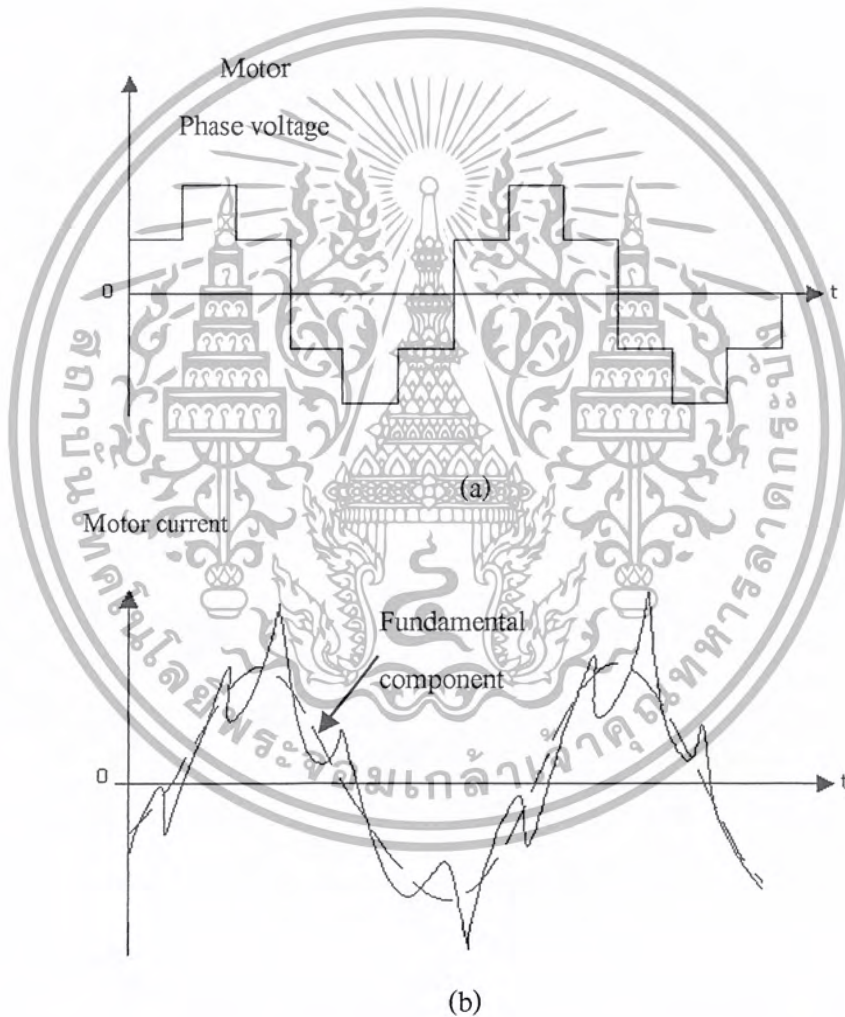


รูปที่ 2.16 PWM-VSI (a) โครงสร้างของ PWM-VSI (b) รูปคลื่นของ PWM-VSI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQUARE-WAVE VSI

สำหรับ SQUARE-WAVE VSI สามารถควบคุมการจ่ายแรงดันเอาต์พุตที่ปรับเปลี่ยนความถี่ได้ การควบคุมแรงดันสี่เหลี่ยมโดยใช้เรกติไฟเออร์เป็นอุปกรณ์ตัดต่อทำให้ค่าเพาเวอร์แฟคเตอร์ (power factor) ทางด้านอินพุตมีค่าต่ำ เราสามารถแก้ไขได้โดยเปลี่ยนมาใช้ (dc/dc) converter ที่เป็นชอปเปอร์มาควบคุม ทำให้ค่าเพาเวอร์แฟคเตอร์ (power factor) ดีขึ้น อินเวอร์เตอร์ชนิดนี้เหมาะสำหรับมอเตอร์ขนาดกำลังต่ำไปจนถึงขนาดปานกลาง และจะใช้ได้ดีในช่วงความถี่อินเวอร์เตอร์สูง แต่อย่างไรก็ตามลักษณะการควบคุมเพื่อที่จะให้ (V/f) คงที่ จะต้องควบคุมวงจรกำลัง 2 ส่วนคือในส่วนของสี่เหลี่ยมและอินเวอร์เตอร์ทำให้ระบบซับซ้อนจึงทำให้เสถียรภาพของระบบต่ำลง



รูปที่ 2.17 (a) ลักษณะสัญญาณของ SQUARE-WAVE VSI (b) ผลของฮาร์โมนิกส์ที่เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CURRENT SOURCE INVERTER (CSI)

CSI สามารถควบคุมได้โดยการควบคุมเฟสของ ac line และมีอินดักเตอร์ขนาดใหญ่ต่ออนุกรมอยู่แต่ไม่มีตัวเก็บประจุต่อขนานอยู่ด้วยเพื่อสร้างแหล่งจ่ายกระแสให้กับอินเวอร์เตอร์ซึ่งเป็นข้อแตกต่างของโครงสร้างทางวงจรของ VSI สำหรับ CSI มีข้อดีซึ่งสามารถสรุปได้ดังนี้ วงจรกำลังจะใหญ่และไว้วางใจได้ดี ป้องกันการลัดวงจรได้ดี แต่ก็มีข้อเสียคือช่วงความถี่ของอินเวอร์เตอร์ปรับได้ค่อนข้างต่ำและไม่สามารถทำงานได้ที่สภาวะไร้อโหลด



รูปที่ 2.18 (a) วงจรของ CSI (b) ลักษณะสัญญาณในอุดมคติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การวิเคราะห์หาค่าแรงบิดของมอเตอร์เหนี่ยวนำ

ในการหาค่าแรงบิดของมอเตอร์เหนี่ยวนำ เราสามารถใช้โปรแกรม MATLAB ในการวิเคราะห์ เพื่อหาค่าของแรงบิด โดยอาศัยสมการทางวิศวกรรม โดยเราจะเปลี่ยนสมการของกระแสและสมการของแรงดันจากระบบ 3 เฟส 3 แกนเป็น 2 แกน โดยอาศัยสมการ

$$u_D = \frac{1}{3}(u_{BA} - u_{AC}) \quad (2-33)$$

$$u_Q = -\left(\frac{1}{\sqrt{3}}\right)(u_{AC} + u_{BA}) \quad (2-34)$$

$$i_D = i_A \quad (2-35)$$

$$i_Q = \left(\frac{1}{\sqrt{3}}\right)(i_A + 2i_B) \quad (2-36)$$

จากนั้นจะนำสมการกระแสคูณกับค่าความต้านทานที่ได้จากมอเตอร์เหนี่ยวนำ



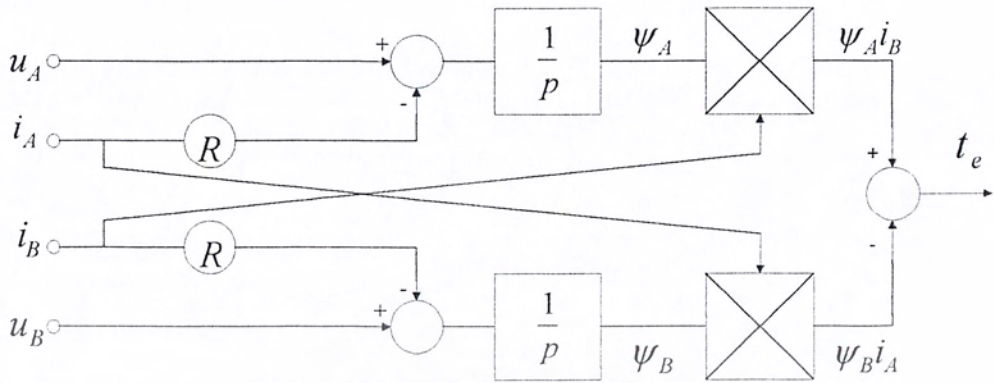
รูปที่ 2.19 บล็อกไดอะแกรมการหาค่าฟลัก

$$\psi_{sd} = \int (V_{sd} - I_{sd}R) dt \quad (2-37)$$

$$\psi_{sq} = \int (V_{sq} - I_{sq}R) dt \quad (2-38)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถหาค่าแรงบิดได้



รูปที่ 2.20 บล็อกไดอะแกรมหาค่าแรงบิด

$$t_e = \psi_{sd} I_{sq} - \psi_{sq} I_{sd} \tag{2-39}$$

การวิเคราะห์เพื่อหาค่ากำลังไฟฟ้า

$$P = \frac{3}{2} (V_{sd} I_{sd} + V_{sq} I_{sq}) \tag{2-40}$$

$$Q = \frac{3}{2} (V_{sd} I_{sq} - V_{sq} I_{sd}) \tag{2-41}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

ปริมาณทางธรรมชาติส่วนใหญ่ จะเป็นปริมาณอนาลอกในทางไฟฟ้าจะเรียกว่า สัญญาณอนาลอก (analog signal) ซึ่งเป็นสัญญาณที่มีค่าต่อเนื่อง ได้แก่ กระแสไฟฟ้า แรงดันไฟฟ้า อุณหภูมิ ความเร็ว หรือเวลา ถ้าหากเราต้องการให้คอมพิวเตอร์ประมวลผลสัญญาณเหล่านั้น เราจะต้องเปลี่ยนสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัล โดยใช้อุปกรณ์ที่เรียกว่า Analog-to-Digital Converter :ADC

2.3.1 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

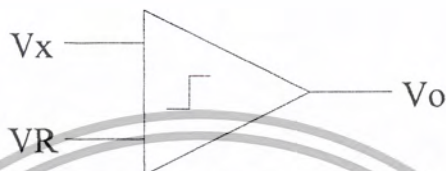
ในการวัดปริมาณต่างๆทางธรรมชาติ ปริมาณเหล่านั้นจะต้องถูกเปลี่ยนให้อยู่ในรูปของสัญญาณไฟฟ้าเสียก่อน โดยมีอุปกรณ์ที่เรียกว่าทรานสดิวเซอร์ (Transducer) เช่น ตัววัดอุณหภูมิจะทำหน้าที่เปลี่ยนอุณหภูมิให้เป็นแรงดันไฟฟ้า การวัดน้ำหนักทำได้โดยใช้ load cell เปลี่ยนค่าน้ำหนักให้เป็นแรงดันไฟฟ้า เป็นต้น จากนั้นจะนำเอาที่พื้ที่ได้คือ อิเล็กตรอนจากร ADC เพื่อเปลี่ยนให้เป็นสัญญาณดิจิทัล เพื่อนำไปประมวลผลโดยระบบจะเป็นดังรูป



รูปที่ 2.21 ระบบเปลี่ยนปริมาณทางธรรมชาติเป็นปริมาณดิจิทัล

จากรูป จะเห็นว่าระหว่างเอาที่พื้ของทรานสดิวเซอร์กับอินพุทของวงจร ADC จะมีวงจร signal conditioning วงจรนี้จะมีหรือไม่มีก็ได้ขึ้นอยู่กับระบบที่ออกแบบขึ้น ถ้าหากเอาที่พื้ที่ได้จากทรานสดิวเซอร์ มีสัญญาณอ่อนเกินไป อาจใช้วงจรนี้ขยายสัญญาณให้แรงขึ้น หรือถ้าหากสัญญาณที่ได้จากเอาที่พื้ของทรานสดิวเซอร์มีรูปร่างไม่เหมาะสม อาจใช้วงจรนี้แต่งรูปสัญญาณให้ดูดีขึ้น

วิธีการแปลงสัญญาณอนาลอกเป็นดิจิตอลแบบง่าย ๆ แสดง ในรูปที่ 2.20 แรงดันอินพุตที่ไม่ทราบค่า V_x จะต่อเข้ากับขาอินพุตขาหนึ่งของอนาลอกคอมพาราเตอร์ และแรงดันอ้างอิงที่ขนาดแปรตามเวลา V_R ต่อเข้ากับอินพุตอีกขาหนึ่งของคอมพาราเตอร์ ลักษณะของทรานเฟอร์ฟังก์ชันของคอมพาราเตอร์ แสดงในรูปที่ 2.21 ถ้าแรงดันอินพุต V_1 มีค่ามากกว่าอินพุต V_2 แล้วแรงดันเอาต์พุตจะเป็นลอจิก 1 ถ้าอินพุต V_1 มากกว่า V_2 แล้วเอาต์พุตจะเป็น 0



รูปที่ 2.22 แสดงวิธีพื้นฐานของ ADC

$$V_o = 1, V_1 > V_2$$

$$V_o = 0, V_1 < V_2$$

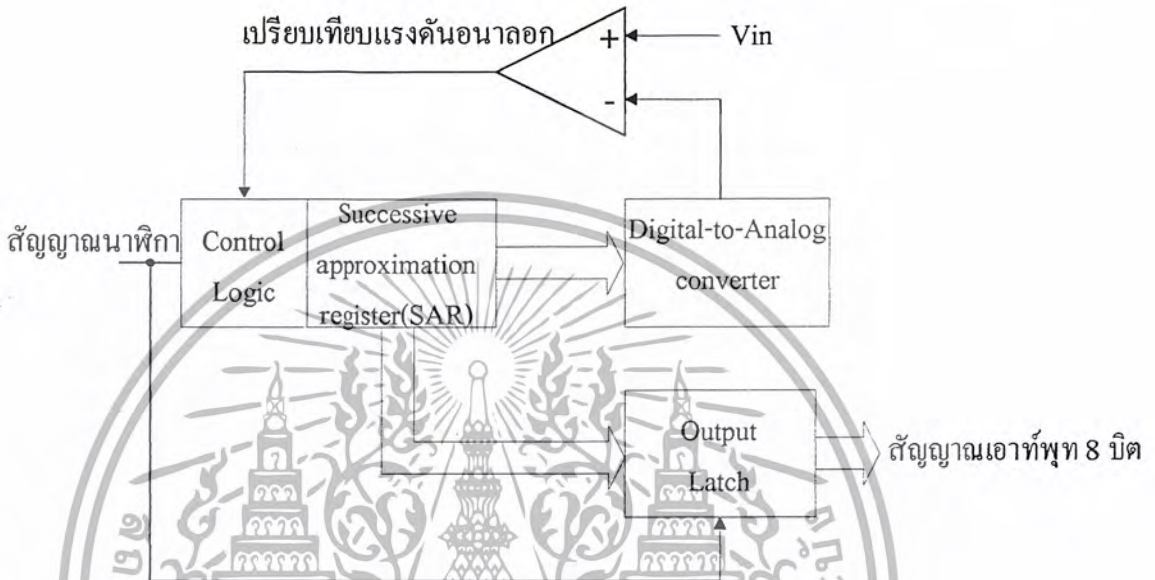


รูปที่ 2.23 แสดงทรานเฟอร์ฟังก์ชันของคอมพาราเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 วงจร ADC แบบ Successive-Approximation

ในระบบแปลงสัญญาณอนาลอกเป็นดิจิทัลมีวิธีการในการแปลงสัญญาณอยู่หลายวิธี แต่ละวิธีมีข้อดีข้อเสียแตกต่างกันไป บางวิธีจะต้องใช้เวลาในการแปลงสัญญาณนาน แต่วิธีที่นิยมใช้กันคือวิธีการประมาณค่าสัญญาณ (Successive-Approximation) โดยมีไดอะแกรมการทำงานของระบบดังรูป



รูปที่ 2.24 วิธีประมาณค่าสัญญาณ

จะเห็นว่าโครงสร้างภายในประกอบด้วยรีจิสเตอร์เก็บข้อมูล SAR วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก และออปแอมป์สำหรับเปรียบเทียบสัญญาณ วิธีนี้เป็นวิธีการแปลงสัญญาณอนาลอกเป็นดิจิทัลขนาด n บิต จะใช้เวลาในการทำงาน n ขั้นตอน เมื่อมีสัญญาณอนาลอกเข้ามาทางอินพุทเริ่มแรกจะให้ค่าในรีจิสเตอร์ SAR เป็นลอจิก "0" ทุกบิต ต่อมาให้บิตสูงสุดเป็นลอจิก "1" (1000 0000) ซึ่งเป็นการเริ่มประมาณค่าและส่งค่านี้เป็นสัญญาณอนาลอกแล้วนำไปเปรียบเทียบกับสัญญาณที่เข้ามา ถ้าค่าที่ประมาณนี้มีค่ามากกว่าสัญญาณอนาลอกที่เข้ามาก็จะให้บิตสูงสุดเป็นลอจิก "0" แต่ถ้าค่าที่ประมาณมีค่าน้อยกว่าสัญญาณที่เข้ามาก็จะให้บิตสูงสุดนี้เป็นลอจิก "1" ต่อไปต่อมาเมื่อมีสัญญาณนาฬิกาเข้าไปอีกหนึ่งลูกก็จะประมาณค่าในบิตถัดลงมาจากบิตสูงสุด และทำการเปรียบเทียบสัญญาณที่เข้ามาทางอินพุทอีกและทำเช่นนี้ซ้ำๆจนประมาณค่าได้ทุกบิต สำหรับความเร็วในการแปลงสัญญาณจะขึ้นอยู่กับความเร็วของสัญญาณนาฬิกาที่ป้อนให้กับวงจรด้วย

2.3.3 การแซมปลิงสัญญาณ

ในการแปลงสัญญาณอนาลอกเป็นรหัสดิจิทัลอนันต์ ADC จะต้องใช้เวลาช่วงหนึ่งในการจัดการซึ่งเวลาดังกล่าวขึ้นอยู่กับหลายๆแฟคเตอร์ เช่น ความละเอียดของการเปลี่ยนสัญญาณ (จำนวนดิจิทัลบิต) เทคนิคของการเปลี่ยนแปลงสัญญาณ และความเร็วในการทำงานของอุปกรณ์ร่วมอื่นๆ การกำหนดความเร็วของการแปลงสัญญาณขึ้นอยู่กับการประยุกต์ใช้งานเฉพาะอย่างและความแม่นยำที่ต้องการ ช่วงเวลาในการแปลงสัญญาณบางครั้งเรียกว่า Aperture time ซึ่งความหมายโดยทั่วไปหมายถึงช่วงเวลาที่เกิดความไม่แน่นอนของการวัดและผลก็คือความผิดพลาด (Error) ค่าที่วัดได้

Sampling and hold และ Aperture time error

วงจร Sampling and hold จะทำการสุ่ม (Sampling) สัญญาณอินพุต และนำสัญญาณที่สุ่มนั้นมาเก็บ (hold) ไว้ในช่วงเวลาหนึ่งได้ ซึ่งส่วนใหญ่จะใช้การประจุแรงดันนั้นไว้ในตัวเก็บประจุที่รั่วไหลต่ำ Aperture time ของ Sampling & Hold คือเวลาดังแต่เริ่มสุ่มสัญญาณเก็บประจุค่าแรงดันจนถึงค่าที่สุ่ม ซึ่งสำหรับ Sampling and hold และ Aperture time ขึ้นอยู่กับแบนวิดท์ และ Switching time ของอุปกรณ์แอคทีฟที่ใช้ในวงจร ซึ่งหาและสร้างได้ง่ายและราคาถูกกว่าการสร้าง ADC ความเร็วสูง

ในการสุ่มสัญญาณอนาลอกจะถูกสุ่มเป็นระยะๆคงที่ตามรูปที่ 2.23 การสุ่มจะเป็นการตัดต่อสัญญาณอนาลอก ในช่วงเวลาอันสั้นด้วยสวิทซ์ที่ทำงานด้วยความเร็วสูง ผลของการสุ่มสัญญาณด้วยความเร็วจะเสมือนกับการคูณขบวนสัญญาณพัลส์แคบๆ กับสัญญาณอนาลอก ซึ่งจะได้เป็นสัญญาณที่มอดูเลทระหว่างขบวนพัลส์สัญญาณอนาลอก โดยเสมือนว่าสัญญาณอนาลอกเข้ามาบนขบวนพัลส์ ดังแสดงในรูปที่ 2.23ค ถ้าหากสัญญาณอนาลอกที่ถูกสุ่ม ถูกสุ่มจนกว่าสัญญาณค่าใหม่ถูกสุ่มเข้ามาซึ่งจะได้ลักษณะของเอาท์พุทแสดงในรูปที่ 2.23ง

มีปัญหาที่ว่าอัตราการสุ่มสัญญาณนั้นควรมีขนาดเท่าไร จึงจะไม่ทำให้ข้อมูลสูญหายไป เมื่อสัญญาณนั้นถูกเปลี่ยนกลับมาเป็นเช่นเดิม คำตอบคือขึ้นอยู่กับสัญญาณอนาลอก ซึ่งทฤษฎีของการสุ่มตัวอย่างกล่าวไว้ว่า “ถ้าสัญญาณต่อเนื่องซึ่งมีความถี่และฮาร์โมนิกไม่เกิน f_c จะถูกสุ่มด้วยอัตราการสุ่มไม่น้อยกว่า $2f_c$ แล้วสัญญาณดังกล่าวจะสามารถเปลี่ยนกลับมาได้อย่างเดิมโดยไม่สูญเสียรายละเอียดหรือผิดเพี้ยนไป”

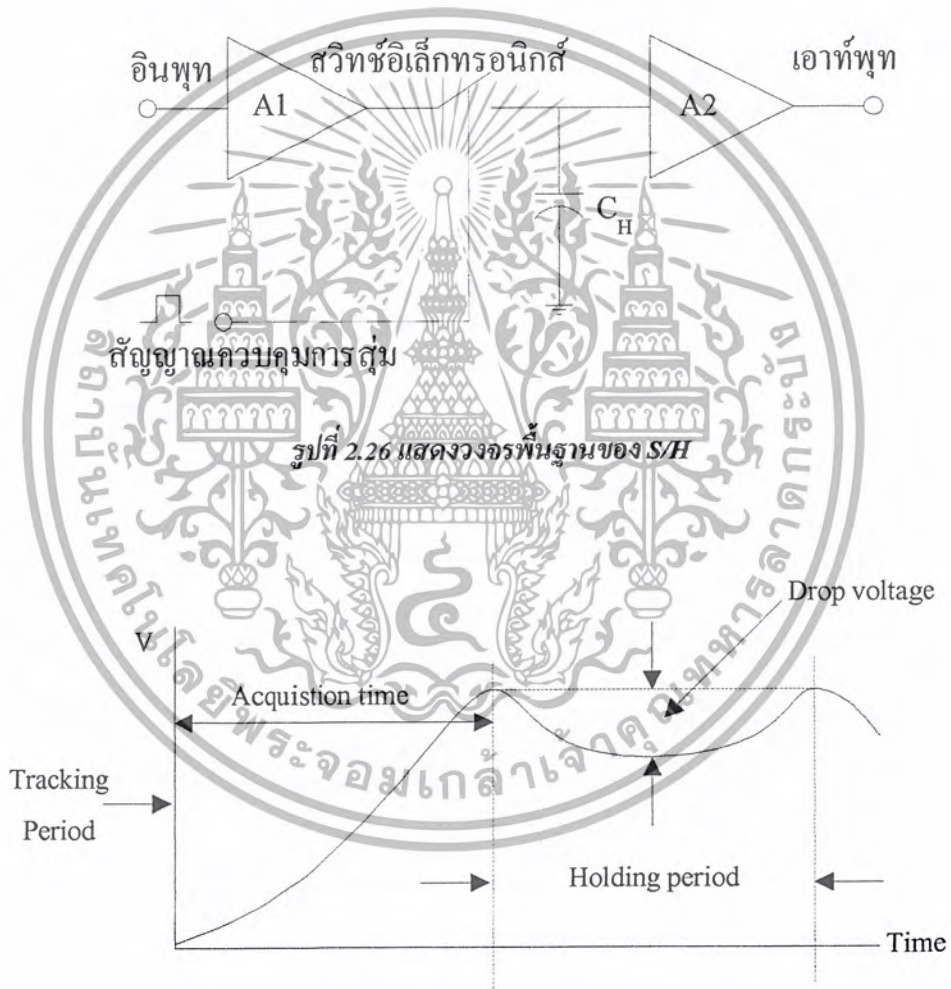


รูปที่ 2.25 การสุ่มสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรแซมปลิงแอนด์โฮลด์ (S/H)

ความจริงแล้ววงจร S/H มีได้มีใช้เฉพาะกับ ADC เท่านั้นแต่ก็ยังใช้กันในระบบ Data distribution , Sampling scope , DVM , Reconstruction filter และ อนุภาคคอมพิวเตอร์เป็นต้น วงจร S/H โดยพื้นฐานแล้วเป็นอุปกรณ์หรือวงจรเก็บแรงดัน (Voltage memory) ซึ่งใช้อุปกรณ์ร่วมที่สำคัญคือตัวเก็บประจุในรูปที่ 2.24 แสดงวงจรพื้นฐานของ S/H อิเล็กทรอนิกส์จะต่อกับสัญญาณแรงดันเข้าตัวเก็บประจุซึ่งสวิทช์นี้ควบคุมจาก Sampling pulse ช่วงการตัดต่อสวิทช์และเวลาในการประจุแรงดันจนถึงค่าที่ Sampling มา นั้นเรียกว่า Aperture time ของ S/H จากลักษณะการทำงานดังกล่าววงจร S/H จะมีจุดต่อสัญญาณเข้าออก 3 จุดด้วยกันคือ สัญญาณอนาล็อกอินพุต, สัญญาณ Sampling และเอาต์พุต

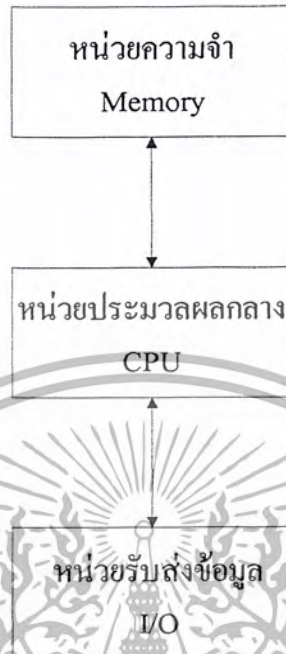


รูปที่ 2.27 แสดงรูปคลื่นเอาต์พุตของ S/H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ทฤษฎีการติดต่อกับคอมพิวเตอร์

2.4.1 โครงสร้างโดยทั่วไปของระบบคอมพิวเตอร์



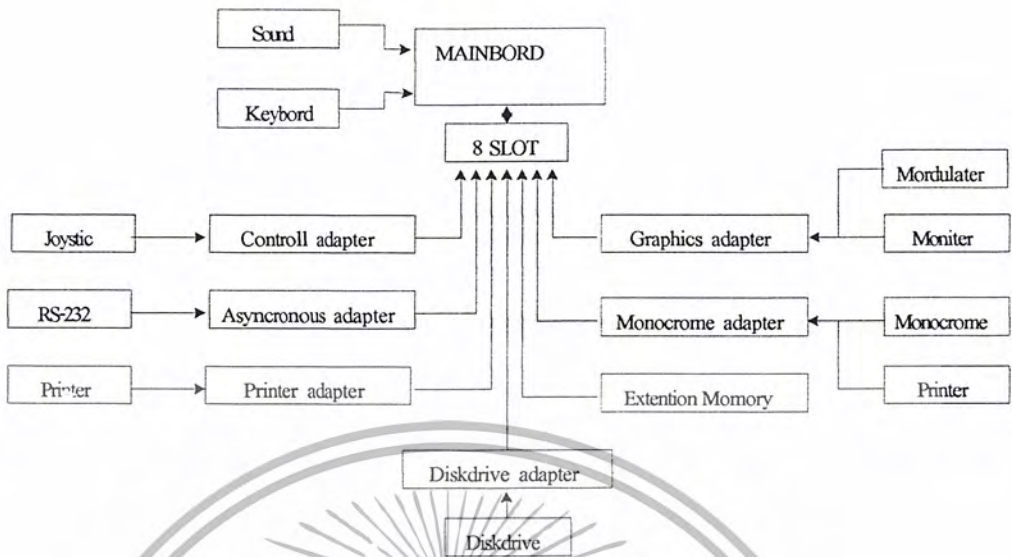
รูปที่ 2.28 แสดงบล็อกไดอะแกรมของระบบคอมพิวเตอร์

ระบบคอมพิวเตอร์ จะประกอบด้วยส่วนประกอบใหญ่ ดังต่อไปนี้

- หน่วยความจำ (Memory Unit)
- หน่วยประมวลผลกลาง (Central Processing Unit)
- หน่วยรับส่งข้อมูล (Input/Output Unit)

ซึ่งได้แสดงดังบล็อกไดอะแกรม ดังรูป จะเห็นว่าแต่ละส่วนทำงานเกี่ยวโยงสัมพันธ์กัน โดยมีการติดต่อเชื่อมโยงจากส่วนหนึ่ง ไปยังอีกส่วนหนึ่ง แสดงให้เห็นว่ามีการเคลื่อนย้ายข้อมูลนั้นบางส่วนจะเป็นแบบทิศทางเดียว บางส่วนจะเป็นแบบสองทิศทาง

ในการพิจารณาการอินเตอร์เฟสจะพิจารณาถึงการทำงานเพียงบางส่วนของส่วนใหญ่ว่าของเครื่องคอมพิวเตอร์ คือการรับและส่งข้อมูลจากส่วนหนึ่ง ไปยังอีกส่วนหนึ่ง ซึ่งจะเกี่ยวข้องกับการทำงานด้านอื่นๆ ที่มีกระบวนการกระทำทางคณิตศาสตร์และทางลอจิกเกิดจากการทำงานของหน่วยประมวลผลกลางร่วมกับอุปกรณ์อื่นๆ บนแผ่นวงจรเดียวกัน เช่น ติดต่อกับหน่วยความจำรอม (ROM : Read Only Memory) , แรม (RAM, Random Access Memory) หน่วยแสดงผล , จอภาพ (Monitor) หรืออุปกรณ์ภายนอกต่างๆ ที่มาทำการ อินเตอร์เฟส ดังรูป



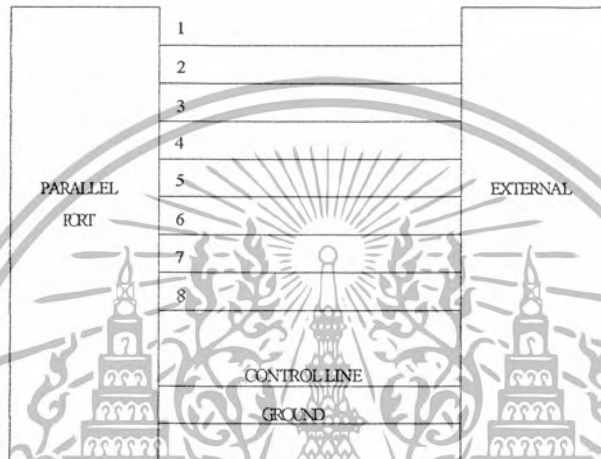
รูปที่ 2.29 แสดงการอินเทอร์เฟซคอมพิวเตอร์

ข้อมูลที่จะโอนย้ายทุกข้อมูลจะต้องมีแหล่งส่งข้อมูลและแหล่งรับข้อมูล ซึ่งในขบวนการต่างๆ จะมีหลักสำคัญอยู่ว่าข้อมูลนั้นจะเป็นตำแหน่ง(Address) หรือข้อมูล(Data) จะส่งจากจุดไหนไปยังจุดไหน เช่นส่งจากหน่วยประมวลผลกลางไปยังหน่วยความจำหรืออุปกรณ์อินพุต/เอาต์พุต และจะส่งเมื่อใด ขบวนการเหล่านี้ โดยทั่วไปจะต้องมีสัญญาณตรวจสอบความพร้อมของอุปกรณ์ที่จะส่งหรือรับข้อมูลก่อนเสมอ ซึ่งจุดส่งข้อมูลและจุดรับข้อมูลจะต้องมีสัญญาณการตรวจสอบความพร้อมเสมอ เพื่อให้ข้อมูลมีการใช้งานอย่างเป็นระเบียบ สำหรับการรับและการส่งข้อมูลไปยังอุปกรณ์ภายนอกโดยข้อมูลที่โอนย้ายไปมานั้นจะอยู่ในลักษณะของเลขฐานสอง ตัวเลขแต่ละตัวจะแทนเป็นบิต อาจจะเป็นแบบ 8 บิต หรือ 16 บิต ก็ขึ้นอยู่กับระบบการรับส่งข้อมูลของระบบนั้นๆ วงจรอินเทอร์เฟซนี้ โดยปกติจะมีหน้าที่สำคัญ 4 ประการ

- บัฟเฟอร์ (Buffer)
- ถอดรหัสข้อมูล(Data Decoding)
- ถอดรหัสคำสั่ง(Command decoding)
- ควบคุมและกำหนดเวลา (Timing Control)

2.4.2 การอินเตอร์เฟสแบบขนาน

การอินเตอร์เฟสแบบขนาน คือการรับส่งข้อมูลที่ข้อมูลหลายๆ บิต จะถูกส่งออกไปพร้อมๆกัน ครั้งเดียวโดยแต่ละบิตจะส่งในสายส่งแต่ละเส้น โดยจะทำการส่งไปพร้อมๆกัน ทั้ง 8 บิต ครั้งเดียว ข้อดีของการรับส่งข้อมูลแบบขนานที่เห็นได้เด่นชัด คือ ความเร็วในการส่งถ่ายข้อมูลที่รวดเร็วกว่าการส่งแบบอนุกรม แต่ระยะทางที่ส่งจะสั้นกว่าการส่งข้อมูลแบบอนุกรม นอกจากสายบัสข้อมูลแล้วยังมีสายส่งข้อมูลอย่างน้อย 2 เส้น ร่วมกันอีก คือ Data Ready และ Ground



รูปที่ 2.30 แสดงระบบการรับส่งข้อมูลแบบขนาน

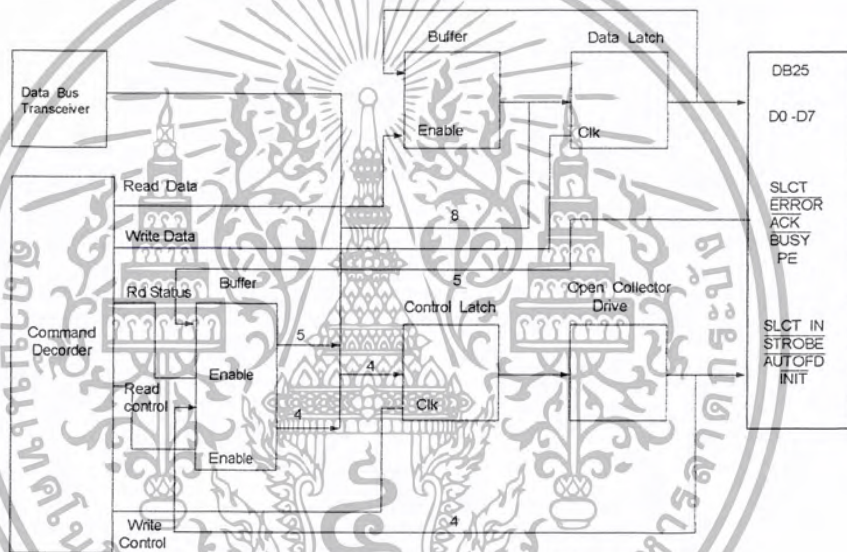
ความจำเป็นของสายกราวด์นั้น มีไว้เพื่อใช้เป็นจุดอ้างอิงของแรงดันไฟฟ้า เพื่อแสดงสถานะทางลอจิกของข้อมูล โดยที่ระดับแรงดันไฟฟ้าของข้อมูลจะมีเพียงลอจิก “0” กับ “1” เท่านั้น สำหรับสาย Data Ready นั้นจะเป็นคำบอกถึงความพร้อมของตัวส่งข้อมูลว่าขณะนี้ข้อมูลพร้อมที่จะส่งแล้ว ข้อสังเกตอีกข้อหนึ่ง คือ ในการส่งข้อมูลแบบขนานนี้ควรมีสัญญาณควบคุม เพื่อใช้ควบคุมการปฏิบัติงานให้ถูกต้อง

2.4.3 พอร์ทขนาน(PARALELL PORT)

พอร์ทขนาน โดยมากจะใช้สำหรับการต่อกับเครื่องพิมพ์(Printer) แต่ก็มีการทำงานหลายอย่างที่ได้ต่อเชื่อมกับพอร์ทขนาน เพื่อนำมาใช้งาน อย่างเช่น การส่งข้อมูลไฟล์ ระหว่างคอมพิวเตอร์ การส่งสัญญาณไปควบคุมการทำงานของอุปกรณ์ภายนอก การรับสัญญาณจากภายนอกเพื่อนำมาประมวลผล และอุปกรณ์ป้องกันของฮาร์ดแวร์ (Hardlock) เป็นต้น ดังนั้นจะเห็นได้ว่าการทำงาน ที่เกิดจากการเชื่อมต่อของพอร์ทขนานมีส่วนที่น่าสนใจและน่าศึกษาอย่างมาก

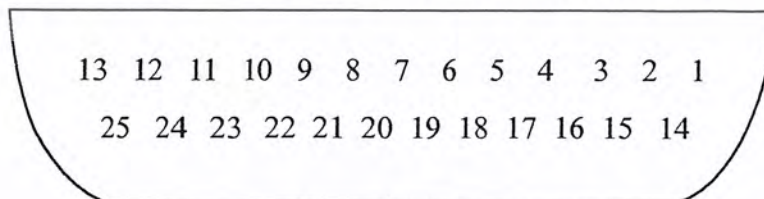
โครงสร้างของพอร์ทขนาน

โครงสร้างของพอร์ทขนานนั้น เป็น โครงสร้างที่ถูกออกแบบมาเพื่อทำงานร่วมกับ เครื่องพิมพ์ โดยเฉพาะดังนั้นสัญญาณต่างๆ จะเป็นสัญญาณที่มีความเกี่ยวข้องกับการติดต่อกับเครื่องพิมพ์ พอร์ทขนานบนเครื่องคอมพิวเตอร์จะมีหลายชุด โดยจะใช้สัญลักษณ์ LTP1,LTP2 เป็นต้น



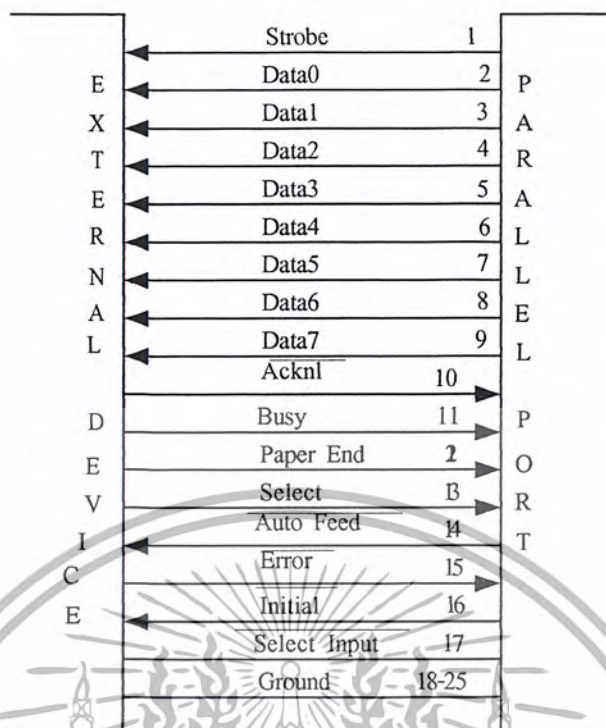
รูปที่ 2.31 บล็อกไดอะแกรมโครงสร้างของพอร์ทขนาน

ขาสัญญาณและรายละเอียดขาสัญญาณ Connector ของพอร์ทขนาน



รูปที่ 2.32 ขาสัญญาณของ Connector DB25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.33 การจัดเรียงสัญญาณของพอร์ตขนาน

| ขาสัญญาณ | สัญลักษณ์ | ทิศทางจากพอร์ต | ความหมาย |
|----------|-----------|----------------|--|
| 1 | -STROBE | OUT | สัญญาณเป็น Low เพื่อส่งข้อมูลให้เครื่องพิมพ์ |
| 2-9 | D0 - D7 | OUT | สัญญาณข้อมูล |
| 10 | -ACKNL | IN | สัญญาณ Low แสดงว่าเครื่องพิมพ์ได้รับข้อมูลแล้ว |
| 11 | BUSY | IN | สัญญาณ High แสดงว่าเครื่องพิมพ์ไม่พร้อม สำหรับรับข้อมูล |
| 12 | PE | IN | สัญญาณ High แสดงว่าเครื่องพิมพ์กระดาษหมด |
| 13 | SLCT | IN | สัญญาณ High แสดงว่าเครื่องพิมพ์อยู่ในสภาวะ Select |
| 14 | -AUTOFD | OUT | สัญญาณ Low เพื่อให้เครื่องพิมพ์เลื่อนกระดาษขึ้นหลังจากพิมพ์เสร็จ |
| 15 | -ERROR | IN | สัญญาณ Low แสดงการผิดพลาดจาก Paper End, Off line |
| 16 | -INIT | OUT | สัญญาณ Low เพื่อให้เครื่องพิมพ์เริ่มดำเนินการทำงานใหม่ |
| 17 | -SLCTIN | OUT | สัญญาณ Low เพื่อให้เครื่องพิมพ์รับข้อมูล โดยปกติอาจจะมีการ กำหนดไว้แล้วที่เครื่องพิมพ์ |
| 18 - 25 | GND | - | กราวด์ |

ตารางที่ 2.1 รายละเอียดขาสัญญาณ Connector ของพอร์ตขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจัดวางตำแหน่งต่างๆที่ควบคุมการทำงานของพอร์ตขนาน

การจัดวางตำแหน่งของพอร์ตขนาน

| LPT | ตำแหน่ง | ความหมาย |
|-----|---------|----------|
| 1 | 378H | Data |
| | 379H | Status |
| | 37AH | Control |
| 2 | 278H | Data |
| | 279H | Status |
| | 27AH | Control |

ตารางที่ 2.2 รีจิสเตอร์ข้อมูล (Data) ตำแหน่ง 378H (LPT1), 278H (LPT2)

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
|----|----|----|----|----|----|----|----|

เมื่อต้องการส่งข้อมูลออกพอร์ตขนาน ไปยังภายนอก ก็ใช้คำสั่ง OUT ออกที่แอดเดรสนี้โดยตรงคุณสมบัติของขาสัญญาณ D0 - D7 นี้จะสามารถจ่ายกระแส(Source) ได้ 2.6 mA และดึงกระแส(Sink) ได้ 24mA ดังนั้นจึงควรระมัดระวังในเรื่องของการเชื่อมต่อ กับอุปกรณ์ภายนอกกับขาสัญญาณเหล่านี้ โดยไม่ให้ Short ลงกราวด์ เพราะจะทำให้วงจรของพอร์ตขนานเสียหายได้

รีจิสเตอร์แสดงสถานะ(Status) ตำแหน่ง 379H(LPT1), 279H(LPT2)

| | | | | | | | |
|-------|-------|-------|-------|-------|---|---|---|
| ขา 11 | ขา 10 | ขา 12 | ขา 13 | ขา 15 | - | - | - |
|-------|-------|-------|-------|-------|---|---|---|

รีจิสเตอร์นี้จะแสดงสถานะการทำงานของพอร์ต โดยบิตที่ 7 (ขา 11) เพียงบิตเดียวที่จะแสดงสถานะตรงข้ามกับสัญญาณจริง ที่ได้รับเข้ามา

รีจิสเตอร์ควบคุม(Control) ตำแหน่ง 37AH(LPT1), 27AH(LPT2)

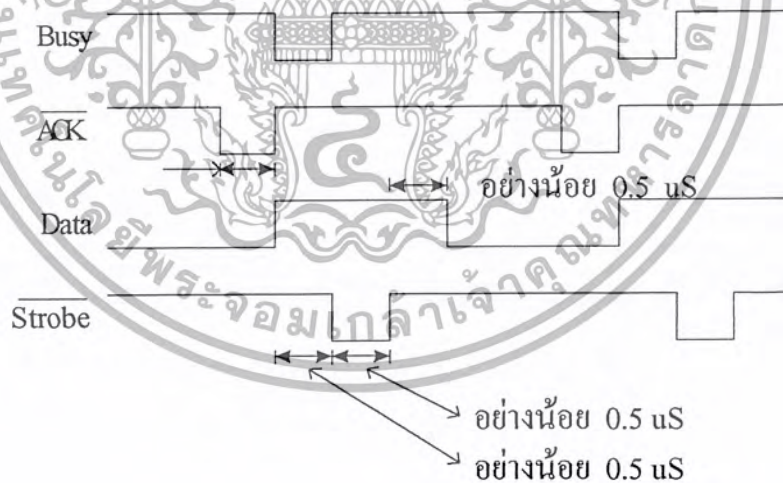
| | | | | | | | |
|---|---|---|---------------|-------|-------|-------|------|
| - | - | - | IRQ Enable | ขา 17 | ขา 16 | ขา 14 | ขา 1 |
|---|---|---|---------------|-------|-------|-------|------|

รีจิสเตอร์ตำแหน่งนี้จะทำหน้าที่ส่งสัญญาณควบคุมออกไปภายนอก โครงสร้างทางฮาร์ดแวร์ของพอร์ตตำแหน่งนี้เป็น Open Collector โดยมี R pull up 4.7K ซึ่งสามารถดันกระแสได้ประมาณ 7 mA ข้อมูลทั้ง 4 บิตล่างจะตรงข้ามกับสัญญาณจริง ที่ส่งไปยังภายนอกยกเว้น บิตที่ 2 (ขา 16 สัญญาณ Initial)ที่ข้อมูลจะตรงกับสัญญาณที่ส่งออกไปภายนอก เช่นกำหนดบิต 1 (Strob) เป็น 1 แต่สัญญาณที่ส่งออกไปภายนอก จะเป็น 0 และบิต IRQ Enable ทำหน้าที่สร้างสัญญาณ อินเทอร์พท์ CPU เมื่อมีการส่งสัญญาณตอบกลับมา(สัญญาณเป็น Low ที่ ขา 10ACKL)

พอร์ตขนานของ PC ในปัจจุบันสามารถที่จะรับและส่งข้อมูลได้ แบ่งตามสถานะจะได้ 3 กลุ่ม ก็คือ พอร์ตส่งข้อมูล เอาท์พุทขนาด 8 บิต พอร์ตควบคุมทำหน้าที่เอาท์พุท ขนาด 4 บิต พอร์ตสถานะ ทำหน้าที่อินพุทขนาด 5บิต และถ้าแบ่งตาม โหมดตามมาตรฐาน IEEE 1284-1994 Standard จะแบ่งได้

2.4.4 โหมดตามมาตรฐานของ IEEE1284-1994 Standard

1. Compatibility Mode หรือ Centronics Mode เป็นโหมคมาตรฐานที่พอร์ตขนานโดยทั่วไปใช้ในการส่งข้อมูลสู่ Printer โหมคนี้ข้อมูลจะถูกวางลงใน ระบบบัสข้อมูลสถานะของ Printer จะถูกเช็คถ้าไม่มีข้อผิดพลาดหรือไม่มีสัญญาณ Busy ส่วนของซอร์พแวร์ ก็จะเป็นตัวสร้างสัญญาณ Strob ส่งให้กับ Printer ดังรูป



รูปที่ 2.34 สัญญาณเวลาในการติดต่อกับเครื่องพิมพ์

การส่งข้อมูลออกเครื่องพิมพ์มาตรฐาน

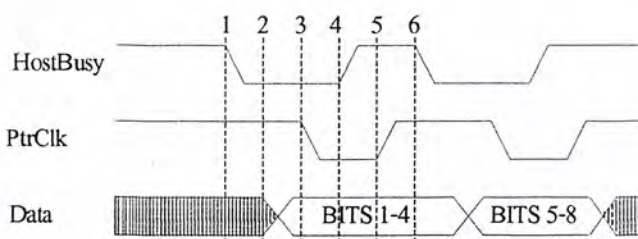
1. รอจนกว่าสัญญาณ Busy จะเป็น Low
2. ส่งข้อมูลไปที่ขา D0- D7 พร้อมหน่วงเวลา 0.5 uS
3. ส่งสัญญาณ Strobe เป็น Low เป็นเวลาประมาณ 0.5 uS และค้างสถานะของข้อมูล D0- D7 ไว้อีก 0.5 uS
4. เครื่องพิมพ์จะทำการส่งสัญญาณ ACK มาเป็น Low อย่างน้อย 5 uS
5. เมื่อสัญญาณ ACK กลับมาเป็น High เครื่องพิมพ์จะให้สัญญาณ Busy เป็น Low

2. Nibble Mode เป็นโหมดของการติดต่อที่ทางพอร์ทขนาน ที่อุปกรณ์ต่อพ่วง (Peripheral) สามารถที่จะส่งข้อมูลย้อนกลับมาให้กับเครื่อง PC ได้ โดยวิธีส่งทีละ 4 บิต (Nibble) แต่การส่ง 2 วงรอบของการส่งข้อมูล โดยใช้พอร์ทที่แสดง สถานะเป็นตัวส่ง 4 พอร์ทและใช้สัญญาณ nACK เป็นตัวระบุวงรอบของการส่งข้อมูลโดยสัญญาณจะเป็นดังตาราง และรูปดังนี้

| ชื่อสัญญาณมาตรฐาน | ชื่อสัญญาณใน Nibble Mode | In /Out | ลักษณะการทำงานของสัญญาณใน nibble Mode |
|-------------------|--------------------------|----------|---|
| nTROBE | NSTROBE | Out | ไม่ใช้ในโหมดนี้ |
| NAUTOFEED | HostBusy | Out | สัญญาณระบุการส่งข้อมูลถ้าเป็น Low ตัว Host(PC) พร้อมรับข้อมูล 4 บิตถ้าเป็น High ตัว Host(PC) ได้รับข้อมูลแล้ว 4 บิต |
| nSELECTIN | 1248Active | Out | จะเป็น High เมื่ออยู่ในโหมดการส่งข้อมูลตามมาตรฐาน 1248 |
| nINIT | NINIT | Out | ไม่ใช้ในโหมดนี้ |
| nACK | PtrClk | In | เป็น Low เมื่อมีการวางข้อมูลแล้วเป็น High เมื่อตอบสนองสัญญาณที่ขา HostBusy เป็นขอขาขึ้น |
| BUSY | PtrBusy | In | ใช้เป็นสัญญาณข้อมูลบิต 3 และ บิต 7 |
| PaperEnd | AckDataReq | In | ใช้เป็นสัญญาณข้อมูลบิต 2 และ บิต 6 |
| SELECT | Xflag | In | ใช้เป็นสัญญาณข้อมูลบิต 1 และ บิต 5 |
| nERROR | NDataAvail | In | ใช้เป็นสัญญาณข้อมูลบิต 0 และ บิต 4 |
| DATA(8:1) | Not Used | Not Used | ไม่ใช้ในโหมดนี้ |

ตารางที่ 2.3 สัญญาณใน Nibble Mode และหน้าที่การทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35 แสดงแผนภาพการติดต่อใน Nibble Mode

การอ่านข้อมูลเข้าพอร์ทของเครื่องพิมพ์

1. สัญญาณจากตัวพอร์ทขนานส่งยืนยันที่จะสามารถรับข้อมูลได้ Host Busy ตกเป็น Low
2. อุปกรณ์ต่อพ่วงเริ่มวางข้อมูล 4 บิตแรก
3. สัญญาณจากอุปกรณ์ต่อพ่วงยืนยันข้อมูลถูกต้องเมื่อ PtrClk เริ่มเป็นขอบขาสูง
4. สัญญาณ HostBusy เป็นขอบขาขึ้นแสดงการรับข้อมูลทั้ง 4 บิต เข้าและแสดงสถานะว่ายังไม่พร้อมสำหรับอีก 4 บิต
5. อุปกรณ์ต่อพ่วงส่งสัญญาณ PtrClk เป็นขอบขาขึ้นเพื่อบอกกับตัวพอร์ทขนาน
6. เริ่มต้นกระทำข้อ 1-4 ใหม่อีกครั้งเพื่อดึงข้อมูล บิต 5-8

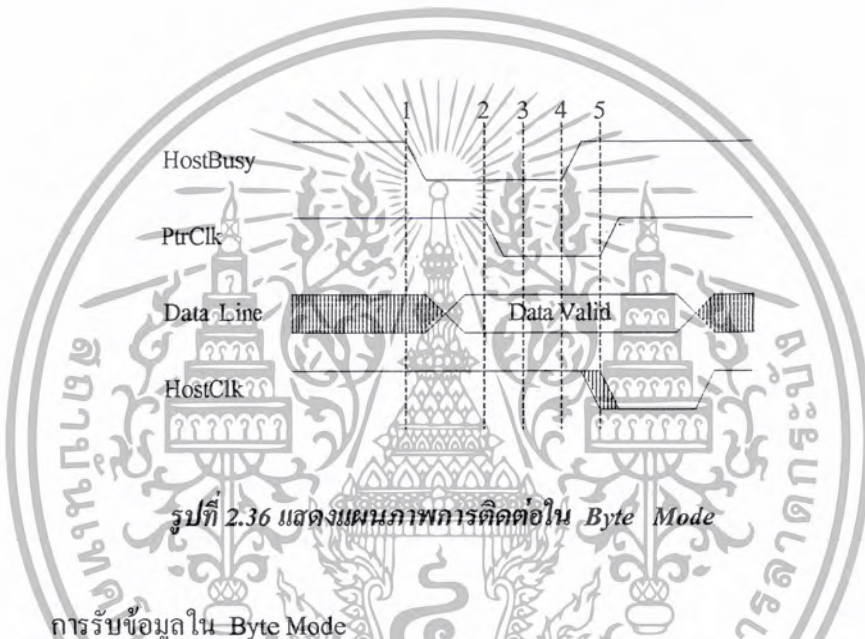
3. Byte Mode ในโหมดนี้จะเป็นการส่งข้อมูลให้กับพอร์ทขนานในแบบเดียวกับ Nibble Mode เพียงแต่สามารถที่จะส่งข้อมูลได้ 1 ไบท์ ในหนึ่งวงรอบของการส่งข้อมูลโดยใช้สาย Data Bus ในการส่ง ซึ่งการส่งข้อมูลเข้าสู่พอร์ทขนานแบบนี้ อาจจะคล้ายกับคอมพิวเตอร์ใหม่คนละทิศทาง ดังนั้นการเรียกการส่งข้อมูลเข้าสู่พอร์ทขนาน โหมดนี้จะเรียกว่า Enhance Bi-direction อาจจะมีการเข้าใจผิดและเรียกการส่งข้อมูลแบบนี้ว่าเป็นแบบ Enhance Parallel Port (EPP) ดังตารางแสดงสัญญาณใน Byte โหมด

| ชื่อสัญญาณมาตรฐาน | ชื่อสัญญาณใน Byte Mode | In / Out | ลักษณะการทำงานของสัญญาณใน Byte Mode |
|-------------------|------------------------|----------|--|
| nTROBE | HostClk | Out | จะเป็น Low เมื่อสิ้นสุดการส่งข้อมูล 1 ไบท์เข้าสู่พอร์ทขนาน |
| nAUTOFEED | HostBusy | Out | สัญญาณระบุงการส่งข้อมูล ถ้าเป็น Low ตัว Host พร้อมรับข้อมูล 8 บิตแล้ว ถ้าเป็น High ตัว Host ได้รับข้อมูล 8 บิตแล้ว |
| nSELECTIN | 1248Active | Out | จะเป็น High เมื่ออยู่ในโหมดการส่งข้อมูลนี้ตามมาตรฐาน 1248 |
| nINIT | NINIT | Out | ไม่ใช้งาน โหมดนี้ |
| nACK | PtrClk | In | เป็น Low เมื่อมีการวางข้อมูลแล้วเป็น High เมื่อตอบสนองสัญญาณที่ขา Host Busy เป็นขอบขาขึ้น |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|-----------|------------|--------------|---|
| BUSY | PtrBusy | In | ตัวส่งสัญญาณ Busy |
| PaperEnd | AckDataReq | In | แสดงตามมาเมื่อมีการส่งข้อมูลแล้ว |
| SELECT | Xflag | In | ไม่มีการใช้งานใน Byte Mode |
| NERROR | NDataAvail | In | เป็น Low เมื่ออุปกรณ์ต่อพ่วง มีการวางข้อมูลแล้ว |
| DATA(8:1) | Data 8 บิต | Bi-direction | ใช้ในการส่งข้อมูลระหว่างพอร์ท และอุปกรณ์ต่อพ่วง |

ตารางที่ 2.4 แสดงสัญญาณใน Byte Mode โหมดและหน้าที่การทำงาน



รูปที่ 2.36 แสดงแผนภาพการติดต่อใน Byte Mode

การรับข้อมูลใน Byte Mode

1. เมื่อสัญญาณ Host Busy ตกเป็น Low แสดงการยืนยันว่าสามารถรับข้อมูลได้จาก Host
2. อุปกรณ์ต่อพ่วงตอบรับ โดยเริ่มการวางข้อมูลที่ถูกต้อง
3. อุปกรณ์ต่อพ่วง ตอบรับความพร้อม โดยให้ PtrClk เริ่มเป็น Low
4. พอร์ทขนานส่ง Host Busy เป็นขอขาขึ้นแสดงการตอบรับ การรับข้อมูลเข้า
5. อุปกรณ์ต่อพ่วงส่งสัญญาณ PtrClk เป็นขอขาขึ้นเพื่อบอกกับพอร์ทขนานและสัญญาณHostClk แสดงการตอบรับ โดยเริ่มเป็นขอขาลง
6. เริ่มวงจรรอบของการส่งข้อมูลในวงจรต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Enhanced Parallel Port (EPP) Mode โหมดนี้ได้ถูกพัฒนาโดย Intel , Xircom และ Zenith data System เพื่อเพิ่มและขยายความสามารถ จาก SPP ให้สามารถส่งข้อมูลได้จำนวนมากและส่งสัญญาณควบคุมแบบ 2 ทิศทางได้ ถ้าแบ่งการส่งข้อมูลเป็นแบบของการส่ง เป็นแบบโปรโตคอล 4 แบบ คือ

- 4.1 วงรอบการเขียนข้อมูล
- 4.2 วงรอบของการอ่านข้อมูล
- 4.3 วงรอบของการเขียน แอคเครส
- 4.4 วงรอบของการอ่าน แอคเครส

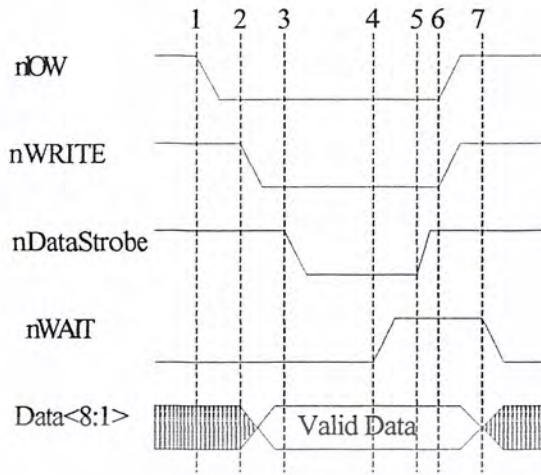
วงรอบของการจัดการจะมี 2 ส่วนหลัก ก็คือ คำสั่งกับ แอคเครส โดยการติดต่อระหว่าง อุปกรณ์ต่อพ่วง

เพื่อใช้งานเฉพาะอย่างผู้ใช้ก็จะสามารถที่จะใช้วงรอบของแอคเครสเป็นสัญญาณควบคุมสัญญาณตามต้องการได้ การส่งข้อมูลในโหมดนี้รองรับการส่งได้ถึง 500 K ถึง 2 M ไบต์ต่อวินาทีทำให้สามารถกล่าวได้ว่าพอร์ทขนานใน โหมดนี้มีความสามารถเทียบได้กับการ์ด บนสล็อตขยาย ISA ในการรับส่งข้อมูลนี้จะใช้ลักษณะของสัญญาณเป็นคู่ในการติดต่อสื่อสาร อย่างคู่ของสัญญาณในการติดต่อและตอบรับของสัญญาณ nWAIT และ nDataStrobe ในการเชื่อมซึ่งกันและกัน

| Spp Signal | EPP Signal Name | IN/ OUT | ลักษณะการทำงานใน EPP Mode |
|------------|-----------------|---------|-------------------------------------|
| NSTROBE | NWRITE | Out | Low = เขียน, High = อ่าน |
| NAUTOFEED | NDATASTROBE | Out | Low = เขียน, High = อ่าน |
| Nselection | nADDRSTB | Out | Low = เขียน, อ่าน, แอคเครส |
| NINIT | nRESET | Out | Low = อุปกรณ์ต่อพ่วงรีเซต |
| NACK | nINIT | In | อุปกรณ์ต่อพ่วงใช้ตอบกลับ(INTR) |
| BUSY | nWAIT | In | Low = เริ่มวงรอบ , High = เสร็จสิ้น |
| D0 – D7 | A/D | Bi-Di | แอคเครส และค่าแบบ 2 ทาง |
| PE | ผู้ใช้กำหนด | In | ผู้ใช้กำหนด |
| Select | ผู้ใช้กำหนด | In | ผู้ใช้กำหนด |
| NERROR | ผู้ใช้กำหนด | In | ผู้ใช้กำหนด |

ตารางที่ 2.5 แสดงสัญญาณเปรียบเทียบใน EPP โหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.37 แผนภาพเวลาแสดงการทำงานใน EPP Mode (วงจรการเขียน)

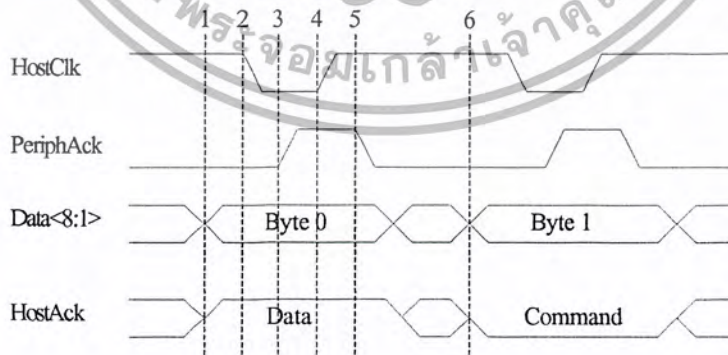
1. โปรแกรมติดต่อกันโดยส่งออกมาเป็นสัญญาณ I/O Write(nOW)
2. สัญญาณ nWRITE ยืนยันการส่งข้อมูลออกมาเป็นขอขาลงมีข้อมูลใน D0-D7
3. สัญญาณ nDataStrobe ส่งยืนยันการวางข้อมูลขณะที่สัญญาณ nWAIT เป็น Low
4. พอร์ตหรือการตอบรับจากอุปกรณ์ต่อพ่วง nWAIT และแสดงการรับข้อมูลเป็นขอขาขึ้น
5. สัญญาณ nDataStrobe ยืนยันการเสร็จวงจรรอบของการเขียน
6. ครบวงจรของ ISA
7. nWAIT เป็น Low แสดงการเริ่มวงจรรอบของการเขียนต่อไป ของข้อมูลชุดใหม่

5. ECP Mode โหมดนี้ได้ถูกพัฒนาต่อมา โดยการร่วมมือระหว่างบริษัทไมโครซอฟท์ และ Hewlett Packard เพื่อให้เหมาะสมกับการใช้งานของ Printer และ Scanner คุณภาพสูงซึ่งการติดต่อจะใช้โปรโตคอลโครงสร้างคล้ายกับแบบของ EPP แต่เพิ่มความสามารถ การติดต่อแบบ DMA ขึ้นมาได้ อีกทั้งประสิทธิภาพของข้อมูลในโหมดนี้ข้อมูลจะเป็นลักษณะการบีบอัดข้อมูล โดยใช้วิธี RLE (Run Length Encoding) ในอัตราส่วน 64:1 ส่งผ่านข้อมูลแบบ FIFO (First In First Out) ทั้งทางพอร์ตและอุปกรณ์ต่อพ่วง การต่อพ่วงอุปกรณ์ในพอร์ตขนานพอร์ตเดียว กับอุปกรณ์หลายตัว เช่น FAX ,MODEM PRINTER ในโหมดนี้ยังสามารถมีการติดต่อในขณะที่ เครื่องพิมพ์อยู่ในสถานะ Busy ก็จะสามารถที่จะรับข้อมูลจากโมเด็มได้ซึ่งเป็นข้อดีเมื่อเปรียบเทียบกับ การทำงานใน Standard Parallel Port (SPP) โหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| SPP Signal | ECP Moad Name | IN/OUT | ลักษณะการทำงานใน ECP Mode |
|------------|-----------------|---------|--|
| NSTROBE | HostClk | Out | ใช้งานคู่กับ PeriphAck แสดงในช่วงการส่งข้อมูลในทิศทาง Forward |
| NAUTOFEED | HostAck | Out | ใช้งานคู่กับ PeriphAck แสดงในช่วงการส่งข้อมูลในทิศทาง Reward |
| NSelection | 1284Active | Out | จะเป็น High เมื่อพอร์ทอยู่ในสถานะส่งข้อมูล |
| NINIT | nReverseRequest | Out | จะเป็น Low เมื่อมีการส่งผ่านข้อมูลในทิศทาง Reverse |
| NACK | PeriphClk | In | ใช้คู่กับ HostAck ในการส่งผ่านข้อมูลในทิศทาง Reverse |
| BUSY | PerphAck | In | ใช้งานคู่กับ HostClk ในการส่งผ่านทิศทาง Forverse และ Reverse เมื่อเป็นคำสั่งควบคุม |
| PE | nAckReverse | In | เป็น Low เมื่อมีการร้องขอส่งผ่านแบบ Reverse |
| SELECT | Xflag | In | ใช้แสดงสถานะของ Flag ที่เพิ่มขึ้น |
| NERROR | nPeriphrequest | In | จะเป็น High เมื่อพอร์ทอยู่ในสถานะส่งผ่านข้อมูลแบบ Reverse |
| D0 - D7 | D0 - D7 | Bi - Di | ใช้ในการส่งผ่านข้อมูลทั้งแบบ Forverse และ Reverse |

ตารางที่ 2.6 แสดงสัญญาณเปรียบเทียบใน ECP โหมด

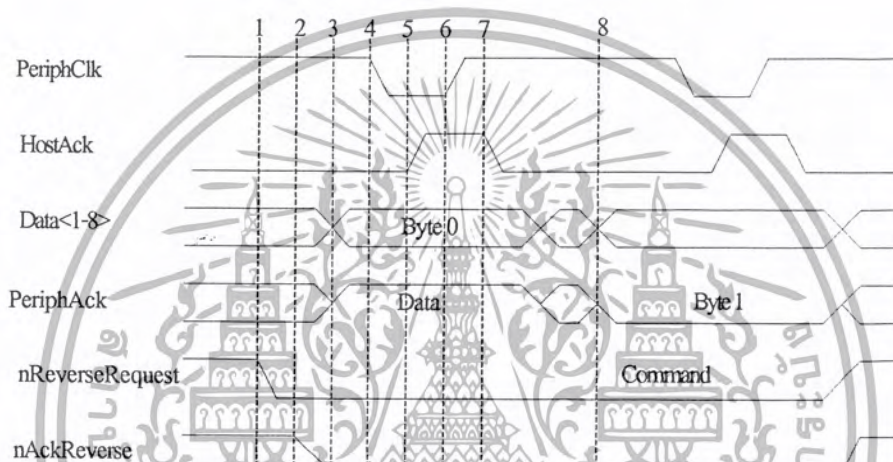


รูปที่ 2.38 แผนภาพเวลาแสดงการทำงานใน ECP โหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Host เริ่มทำการวางข้อมูล สัญญาณ Host Ack เริ่มมีการเปลี่ยนเป็น High
2. Host ยืนยันการวางข้อมูลที่ต้องการ โดยจะเป็น LOW
3. อุปกรณ์ต่อพ่วงเริ่มรับรู้การวางข้อมูล และตอบรับกลับมาโดยแสดง PeriphAck เป็น High
4. HostClk เริ่มเป็นขอขาขึ้นแสดงเป็น Clk ให้กับอุปกรณ์ต่อพ่วงเพื่อกำกับชุดข้อมูล
5. อุปกรณ์ต่อพ่วงกำหนดให้ PeriphAck เริ่มเป็น Low เพื่อพร้อมรับไปท์ต่อไป
6. วงรอบของการส่งเริ่มใหม่แต่ HostClk เป็น Low แสดงถึงชุดข้อมูลที่จะส่ง เป็น คำสั่งควบคุม

คุม



รูปที่ 2.39 แผนภาพเวลาแสดงการทำงานในโหมด ECP

1. Host ร้องขอการส่งข้อมูลแบบ Reverse โดยให้สัญญาณ nReverseRequest เป็น Low
2. อุปกรณ์ต่อพ่วงตอบรับ โดยกำหนด nAckReverse เริ่มเป็น Low
3. อุปกรณ์ต่อพ่วงเริ่มวางข้อมูล โดยแสดงช่วงของข้อมูลด้วยสัญญาณ PeriphAck เป็น High
4. อุปกรณ์ต่อพ่วงยืนยันให้ PeriphClk เป็น Low เพื่อบอกถึงข้อมูลที่ ถูกต้องอยู่ในสายสัญญาณ
5. Host รับรู้การวางข้อมูล โดยให้ HostAck เป็น High
6. Peripheral กำหนดให้ PeriphClk เป็น High ซึ่งขอขาสัญญาณนี้จะเป็น Clk ให้กับ Host

7. Host กำหนดให้ HostAck เป็น Low เพื่อแสดงความพร้อมสำหรับ ไบท์ต่อไป
8. เริ่มวงรอบการส่งข้อมูลให้แก่วงรอบใหม่นี้จะเป็นชุดคำสั่ง เพราะ PeriphAck เป็น Low

การส่งผ่านข้อมูลในโหมคนี้จะสามารถอ้างเข้าถึงโดยใช้ตำแหน่งของ I/O ซึ่งสามารถติดต่อโดยใช้ A0 – A9

ในการติดต่อ แต่ในปัจจุบันโหมคนี้จะใช้การขยายพอร์ทในการติดต่อกับ รีจิสเตอร์ที่ควบคุม การกำหนดโหมครีจิสเตอร์ที่ส่งผ่านข้อมูล โดยใช้หน่วยความจำแบบ Multiple pages โดยการบวก 400h เพื่อสร้างเป็นตำแหน่งใหม่ ในการเข้าถึงในโหมคนี้



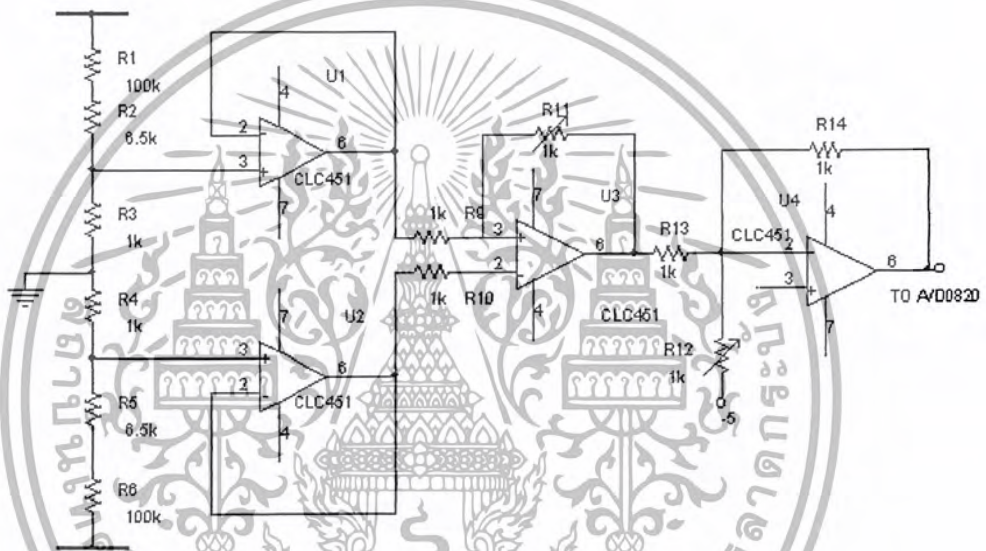
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การคำนวณ การสร้าง การออกแบบ

3.1 การตรวจจับแรงดัน (Voltage detector)

ใช้วงจรตรวจจับแรงดัน เพื่อลดระดับจากแรงดันสูงให้เป็นแรงดันต่ำสำหรับใช้ควบคุมการทำงานของวงจร โดยออกแบบให้ลดระดับแรงดันจาก 380 V ให้เหลือระดับแรงดัน 5 V เพื่อสะดวกสำหรับการทำงานแบบลอจิก วิธีการลดทอนแรงดันได้ใช้ความต้านทานมาเป็นส่วนแบ่งของแรงดัน (Voltage divider) เมื่อได้แรงดันตามต้องการแล้วจะนำมาผ่านวงจรควบคุมเพื่อจัดเตรียมแรงดันให้ได้ตามต้องการและเพิ่มอิมพีแดนซ์ของวงจรให้สูงขึ้น ก่อนที่จะนำไปใช้งานควบคุมจริงต่อไป

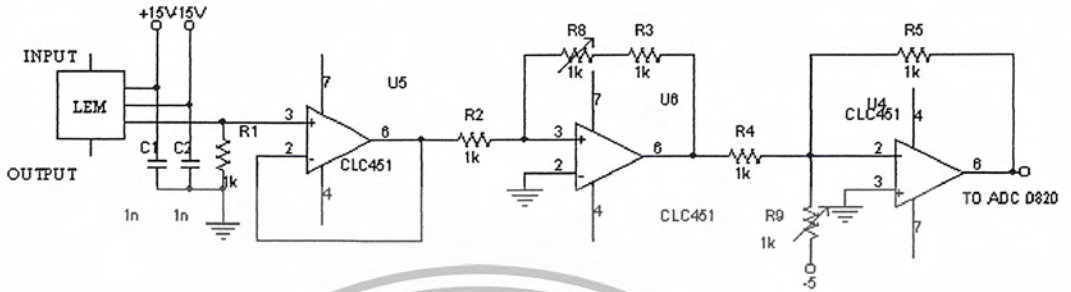


รูปที่ 3.1 วงจรตรวจจับแรงดัน

3.2 การตรวจจับกระแส (Current detector)

วงจรนี้ทำหน้าที่ตรวจจับกระแสเพื่อการควบคุม ในโครงงานนี้ได้ใช้ LEM ซึ่งเป็นอุปกรณ์ที่ใช้หลักการ hall-effect เป็นตัวตรวจจับกระแส เนื่องจากมีข้อดีกว่า CT (current transformer) ในด้านที่สามารถตรวจจับได้กระแสที่เป็นรูปคลื่นไฟฟ้ากระแสตรงและกระแสสลับ สามารถตอบสนองกับงานที่มีความถี่สูงได้ ความผิดพลาดของสัญญาณที่ได้มีค่าและมีให้เลือกใช้งานหลายขนาดวิธีการใช้งานง่าย ซึ่งในการควบคุมนี้ใช้ LEM ตรวจจับกระแสจากมอเตอร์ เพื่อที่จะนำมาใช้งานวงจรต่อไป โดยเลือกพิคคของ LEM 50 A รุ่น LA55-P สำหรับการใช้งาน LEM จะต้องมีการเซทย่านในการลดระดับของกระแสลงซึ่งในโครงงานนี้ใช้ 1:1000 นอกจากนี้ต้องมีวงจรสนับสนุนทำงาน

พร้อมกับ LEM ด้วย เช่น แหล่งจ่ายไฟขนาด +/-15 วงจรบัฟเฟอร์และวงจรปรับค่าออฟเซตสัญญาณ เพื่อที่จะทำให้สัญญาณที่ได้เหมาะสมกับวงจรที่ใช้งานหลังจาก LEM เพื่อช่วยให้การทำงานดียิ่งขึ้น



รูปที่ 3.2 วงจรตรวจอับกระแส

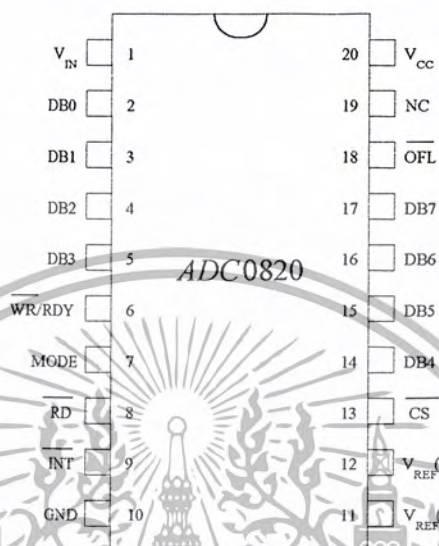


รูปที่ 3.3 โครงสร้างภายนอกของ LEM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วงจรการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

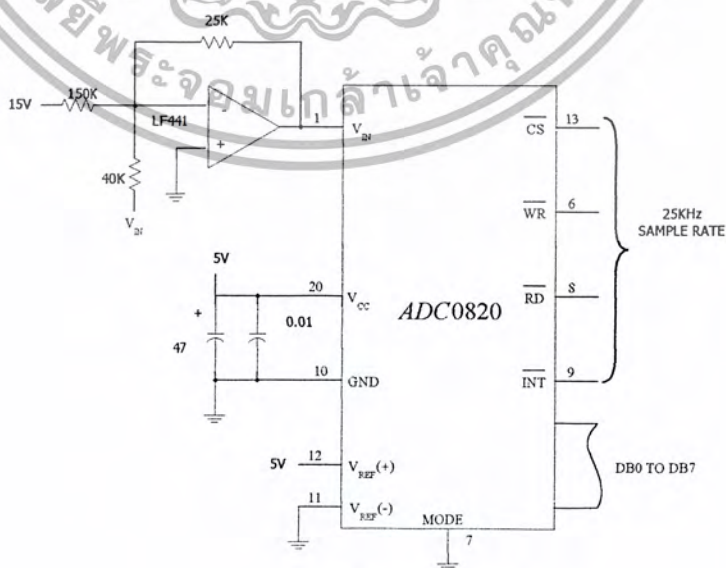
ในโครงการนี้ใช้ ADC0820 8-bit เป็นตัวรับจากอุปกรณ์ตรวจจับแล้วทำการเปลี่ยนจากสัญญาณอนาลอกเป็นสัญญาณดิจิทัล ซึ่งมีคุณลักษณะเช่น มี Conversion Time เท่ากับ $2.5\mu s$ สามารถใช้ได้ถึง 400 kHz มีความผิดพลาดและพลังงานต่ำ



รูปที่ 3.4 ลักษณะของ ADC0820

ลักษณะพื้นฐาน

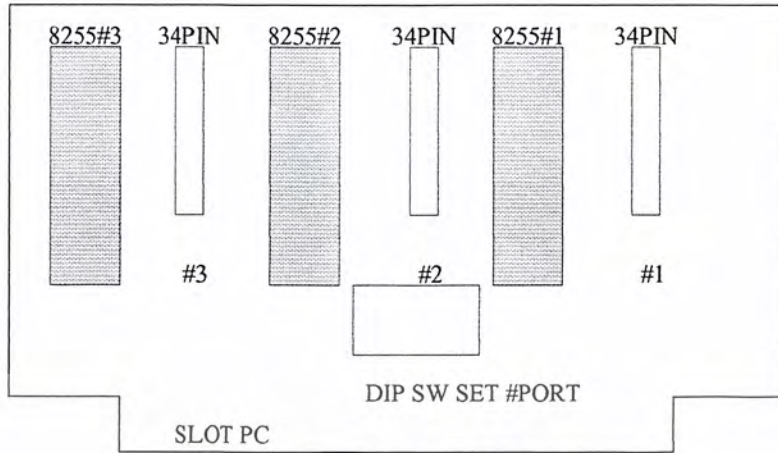
ADC0820 8-bit เป็นเบอร์ที่มีความเร็วสูงทำงานด้วยวิธี track/hold ใช้เทคนิคของวิธีการประมาณค่าสัญญาณ (Successive Approximation) ซึ่งง่ายต่อการ interface มีความเร็วในการแปลงสัญญาณ $2.5\mu s$ ADC0820 ต้องใช้ไฟ +5V และแรงดัน reference 1.23V สำหรับการแปลงสัญญาณจาก 0-2 V_{REF}



รูปที่ 3.5 การต่อ ADC0820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ลักษณะของ ET-PC 8255



รูปที่ 3.6 ลักษณะของ ET-PC 8255

ET-PC 8255 จะเป็นการคัดต่อขยายระบบเครื่อง PC ให้มีส่วนของอินพุต,เอาต์พุต พอร์ตใช้งานมากขึ้น โดยจะมีพอร์ตใช้งานเป็นอินพุต หรือเอาต์พุต จำนวน 9 พอร์ต หรือ 72 บิต I/O(TTL 0-5V)การทำงาน of ET-PC 8255 การ์ด ET-PC 8255 จะประกอบด้วย 2 ส่วนใหญ่ๆคือส่วน IC 8255 ซึ่งเป็น IC ทำหน้าที่เป็นอินพุต,เอาต์พุต พอร์ต และส่วนของวงจร IC DECODE (เลือกตำแหน่งของ PORT 8255)คือ IC 74LS688 และ 74LS139 และ DIP SW การใช้งาน IC 8255

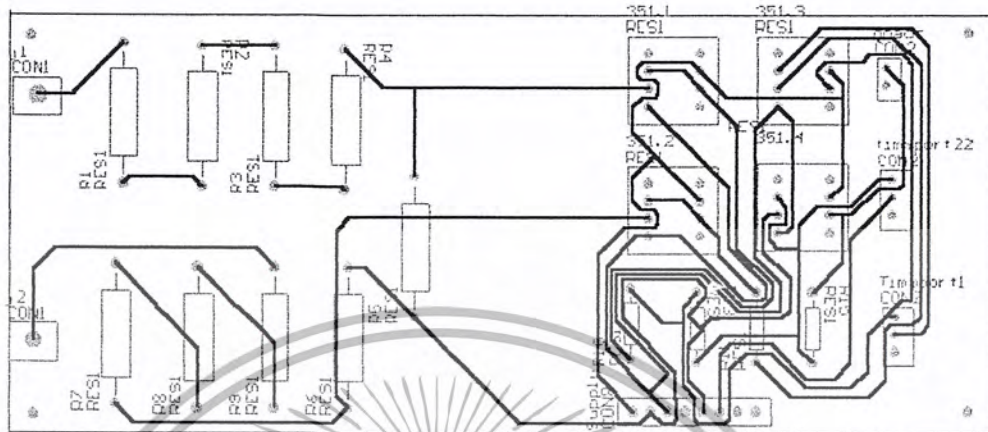
IC 8255 นี้จะเป็น IC ซึ่งประกอบด้วย พอร์ต ใช้งาน 3 พอร์ต และอีก 1พอร์ต ควบคุมก่อนที่เราจะใช้งาน 8255 เราจะต้องส่งข้อมูลไปให้ยังพอร์ต ควบคุมก่อนว่าจะให้ พอร์ต ทั้ง 3 พอร์ต ของ 8255 ที่เหลือนั้นทำหน้าที่อะไร เป็น อินพุต หรือเอาต์พุตพอร์ต เราจะต้องเป็นผู้กำหนด CONTROL CODE PORT ควบคุมดังรูป



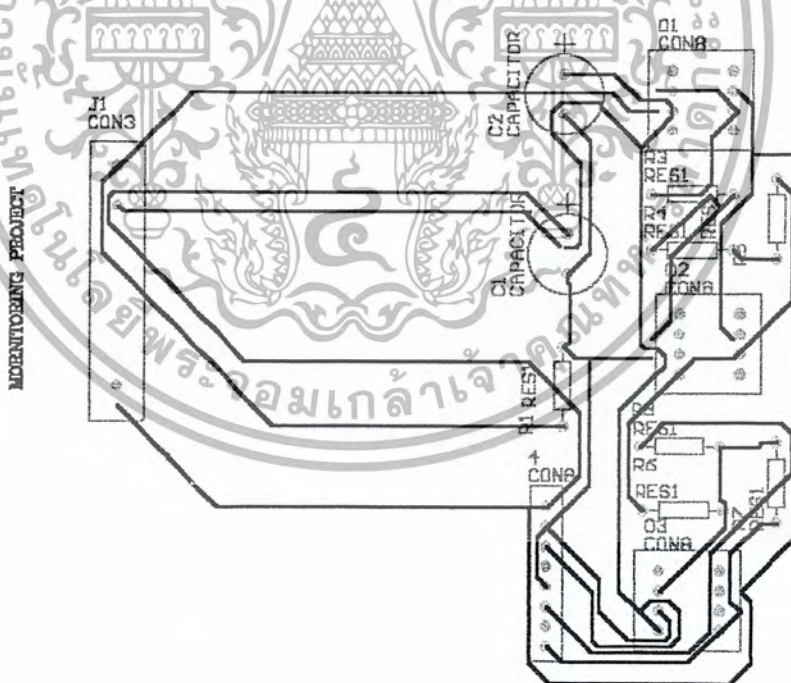
รูปที่ 3.7 ลักษณะของ IC 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้โปรแกรม Potel ออกแบบวงจรที่ใช้ใน โครงการงานซึ่งเราทำการออกแบบวงจรตรวจจ็ับแรงและ วงจรตรวจจ็ับกระแสอย่างละ 2 ชุด และวงจรเปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 4 ชุด

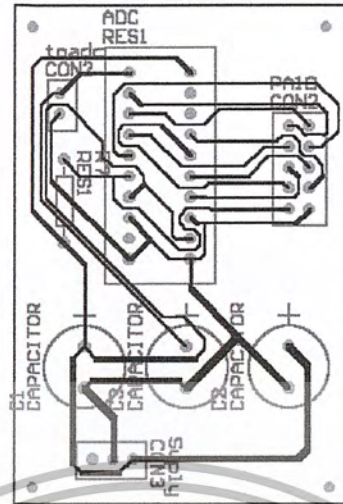


รูปที่ 3.8 แสดงรายวงจรถวจจ็ับแรงดัน



รูปที่ 3.9 แสดงรายวงจรถวจจ็ับกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

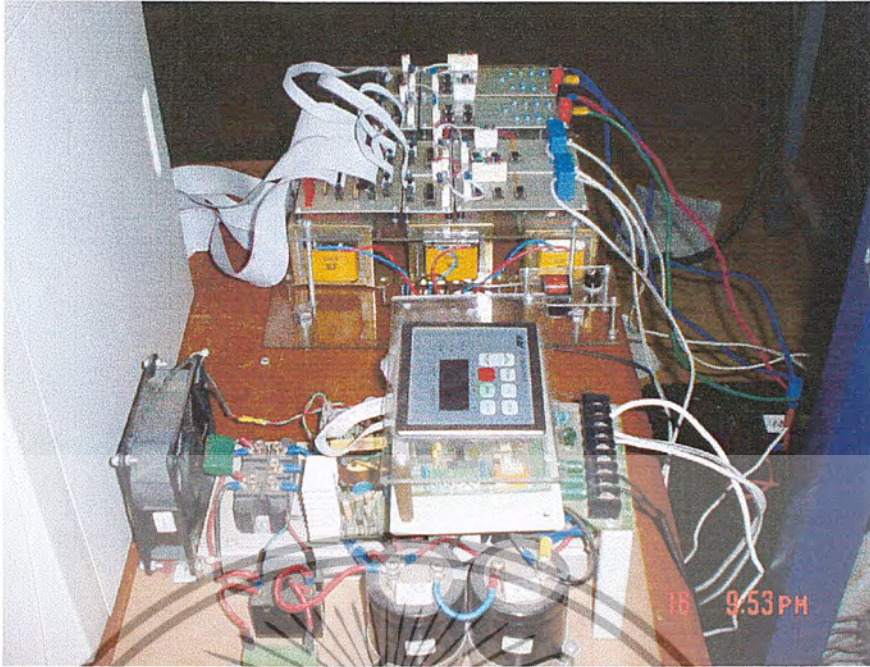


รูปที่ 3.10 แสดงรายการวงจร ADC0820

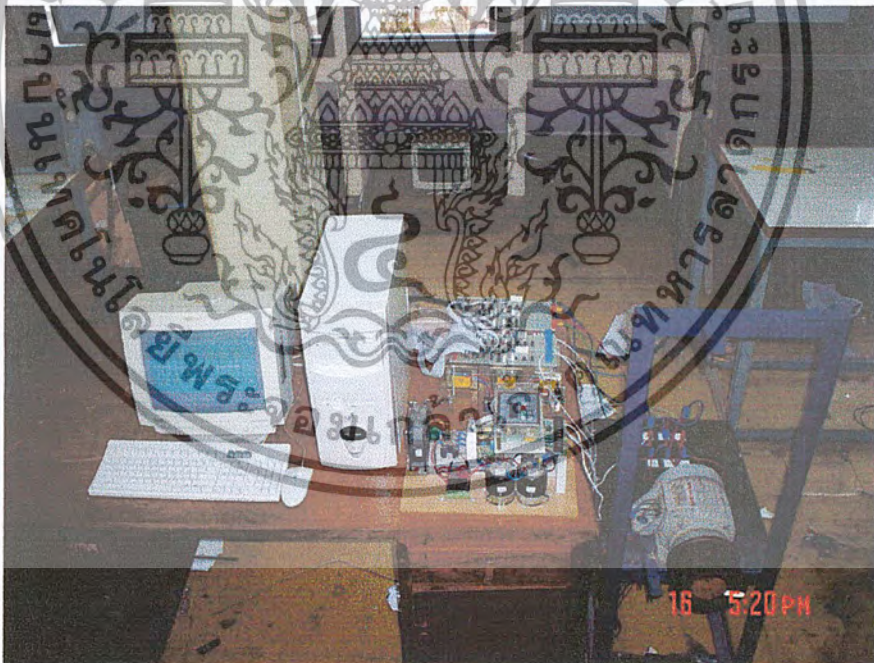


รูปที่ 3.11 แสดงรูปโครงการที่สร้างขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

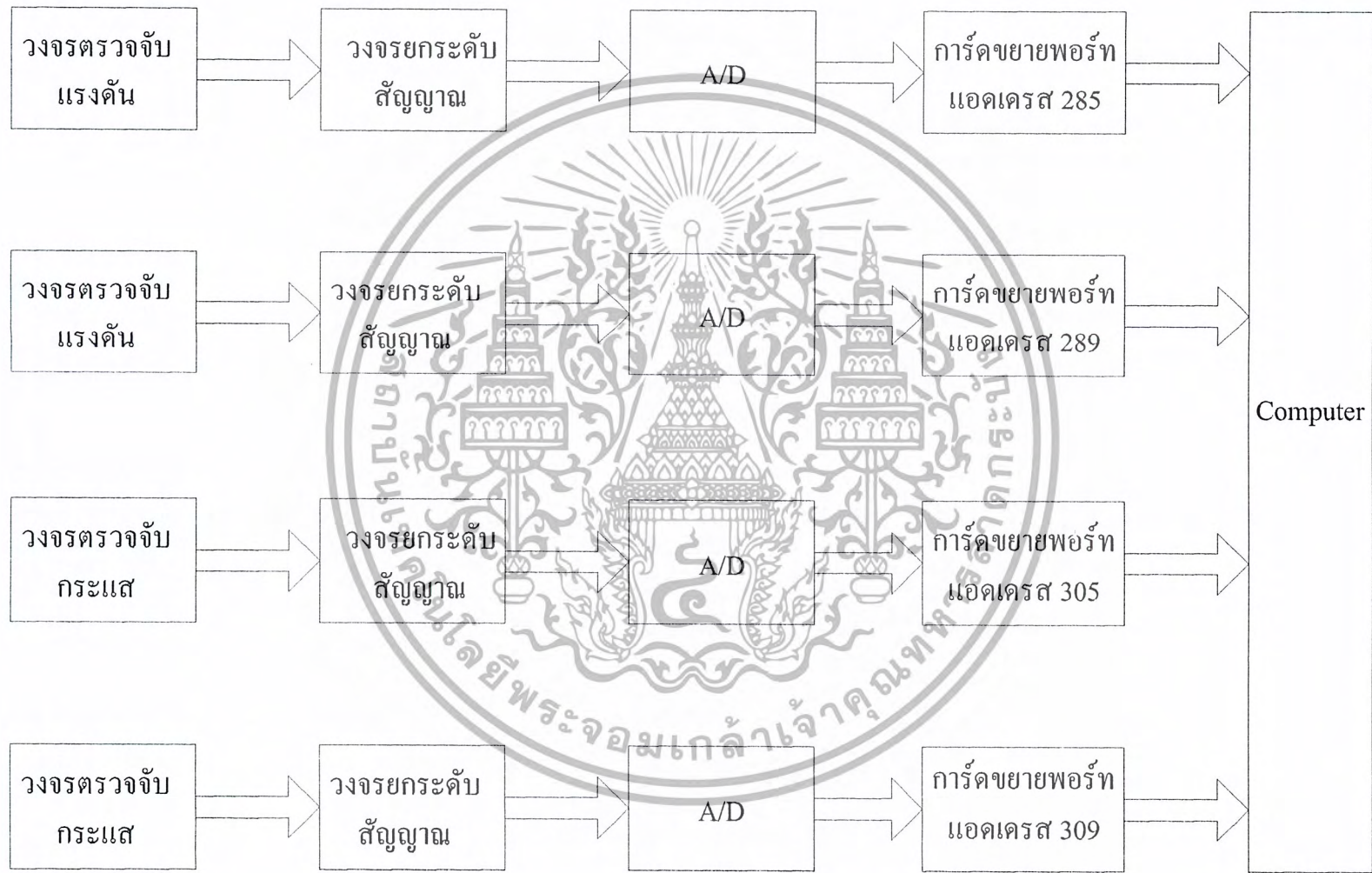


รูปที่ 3.12 แสดงรูปโครงงานที่สร้างขึ้นกับอินเวอร์เตอร์ที่ใช้



รูปที่ 3.13 แสดงรูปลักษณะโครงงานทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



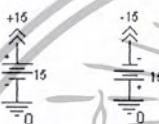
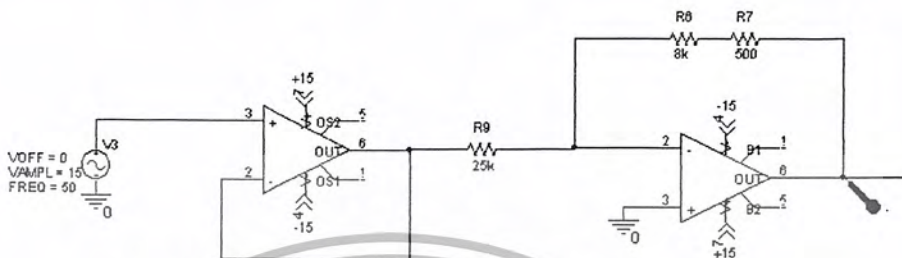
รูปที่ 3.14 แสดงโครงสร้างของโครงการ

บทที่ 4

ผลการทดลองและการทดสอบ

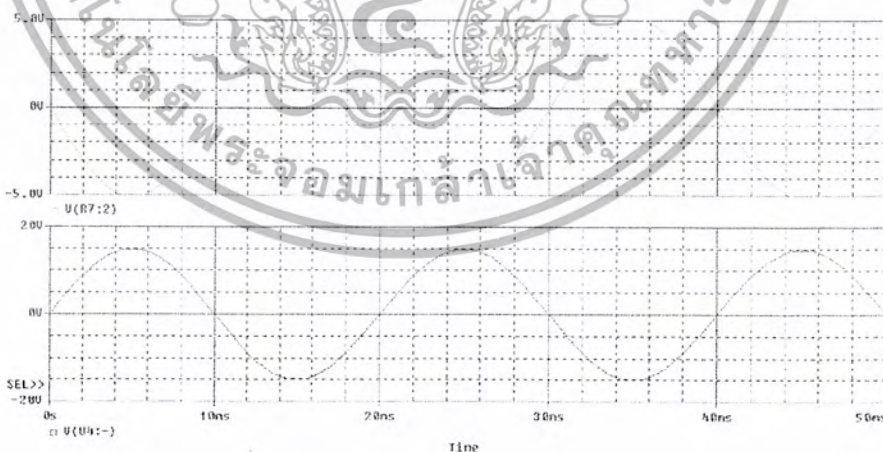
4.1 การเปรียบเทียบสัญญาณระหว่างการจำลองโดยใช้โปรแกรม ORCAD กับวัดสัญญาณจริง

4.1.1 จำลองการวัดกระแสโดยใช้สัญญาณอินพุตเป็นไซน์(Sinwave)



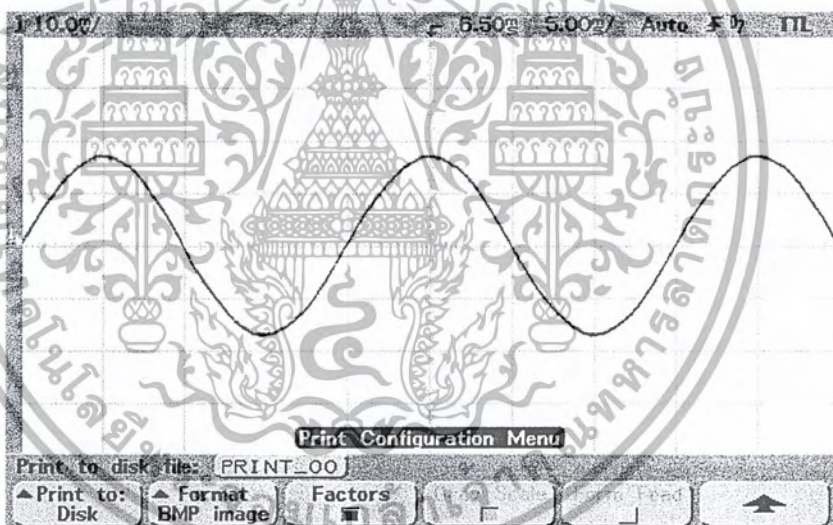
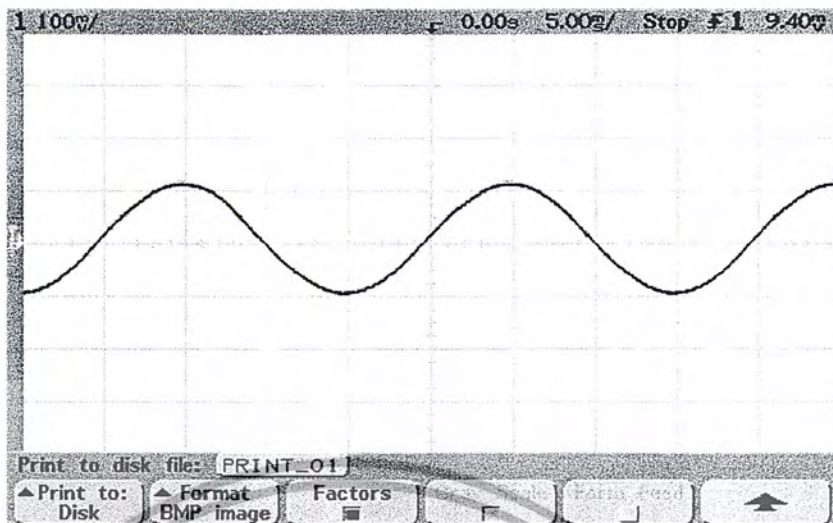
รูปที่ 4.1 แสดงวงจรจำลองตรวจจับกระแสไซน์

เปรียบเทียบสัญญาณระหว่างการจำลอง โดยใช้โปรแกรม ORCAD กับวัดสัญญาณจริงของกระแสไซน์



รูปที่ 4.2 แสดงรูปคลื่นของวงจรจำลองตรวจจับกระแสไซน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

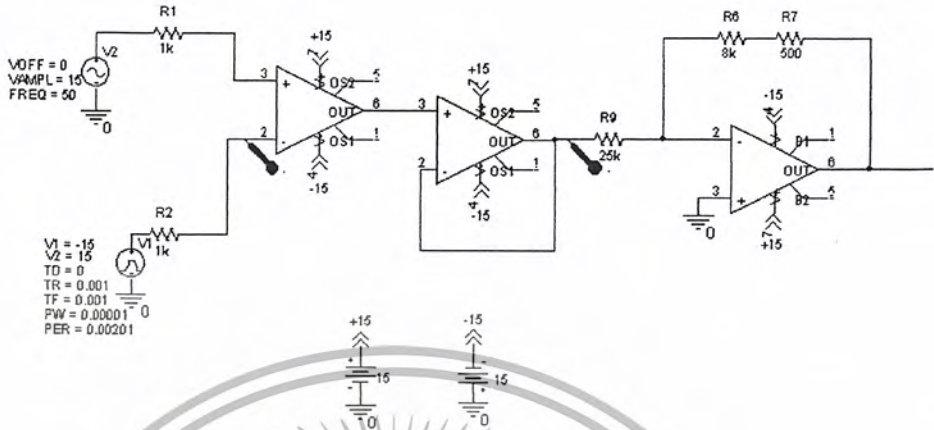


(b)

รูปที่ 4.3 (a),(b) แสดงรูปคลื่นของวงจรตรวจับกระแสชายน

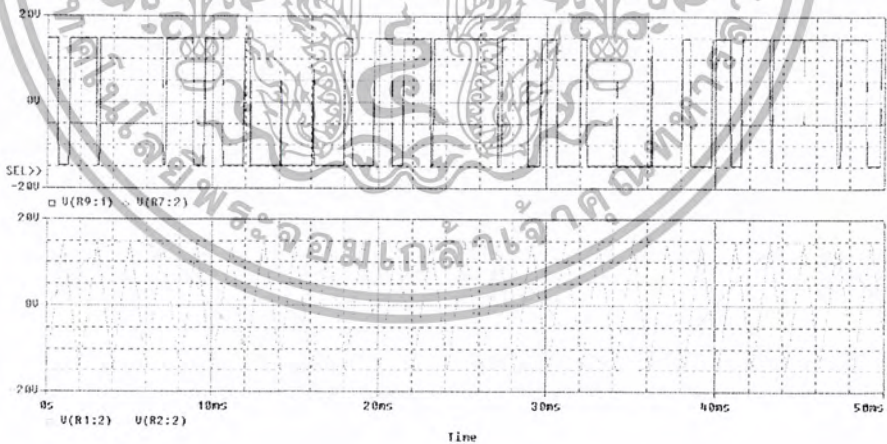
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 จำลองการวัดกระแสโดยใช้สัญญาณอินพุตเป็นพีคดับบีวเอ็ม (PWM)



รูปที่ 4.4 แสดงวงจรจำลองตรวจจ่ายกระแสพีคดับบีวเอ็ม (PWM)

เปรียบเทียบสัญญาณระหว่างการจำลอง โดยใช้โปรแกรม ORCAD กับวัดสัญญาณจริงของกระแสพีคดับบีวเอ็ม (PWM)

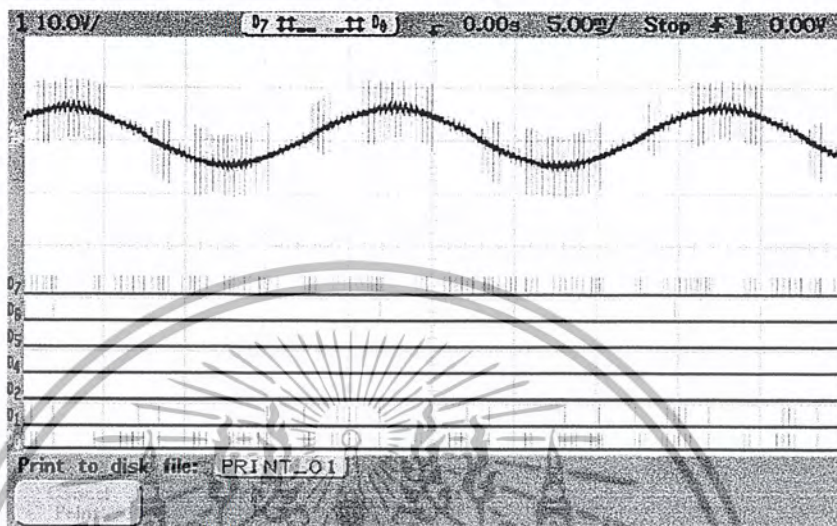


รูปที่ 4.5 แสดงรูปคลื่นของวงจรจำลองตรวจจ่ายกระแสพีคดับบีวเอ็ม (PWM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

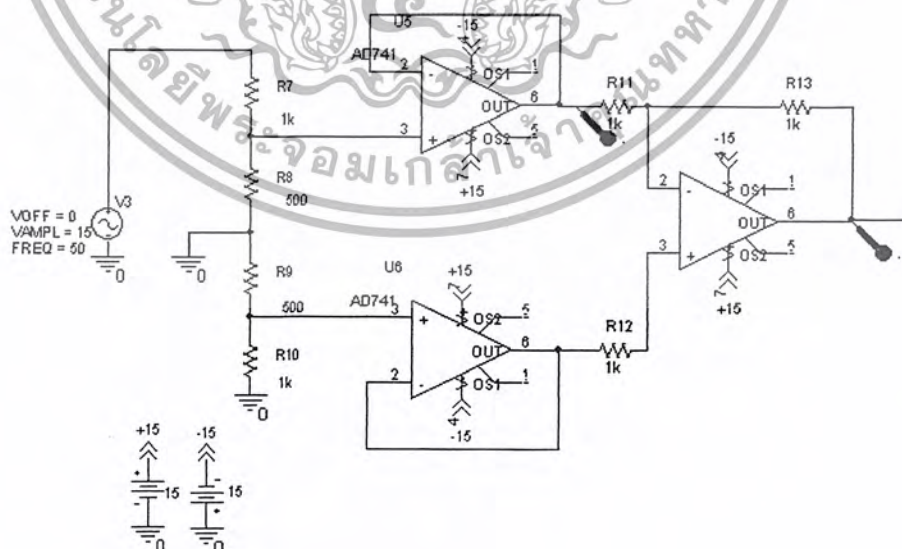
ทดสอบสัญญาณกระแสที่ดับบีวเอ็ม(PWM) ด้วยอินเวอร์เตอร์(INVERTER)โดยใช้ความถี่ fundamental 50 Hz

จากผลของการจำลองกับการทดลองจริงสัญญาณที่ได้มีความแตกต่างกันบ้างเนื่องจากการใช้โปรแกรม ORCAD ไม่สามารถสร้างสัญญาณทดสอบได้เหมือนของจริง



รูปที่ 4.6 แสดงรูปคลื่นของวงจรตรวจจับสนสัญญาณ PWM และสัญญาณที่ออกจาก A/D

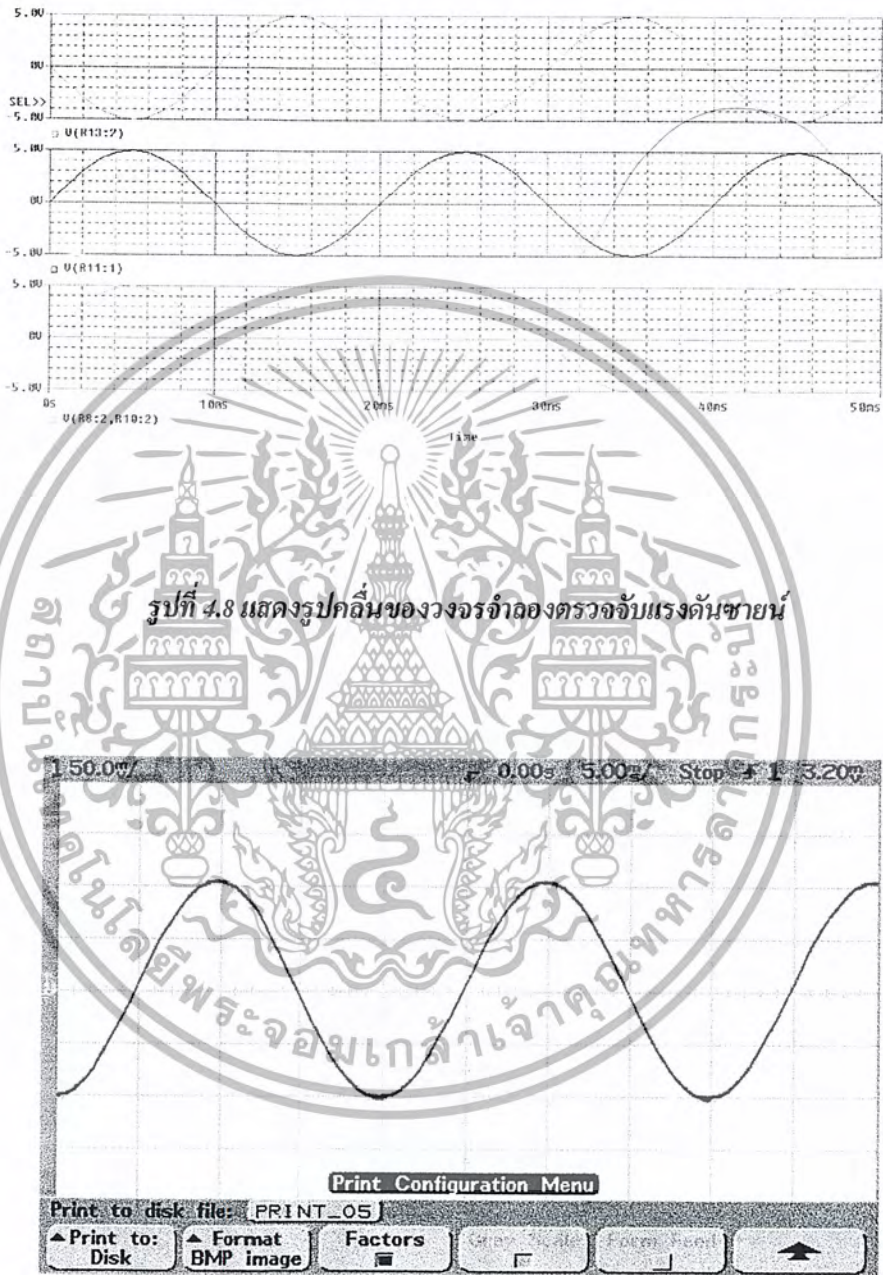
4.1.3 จำลองการวัดแรงดันโดยใช้สัญญาณอินพุตเป็นไซน์ (Sinwave)



รูปที่ 4.7 แสดงวงจรจำลองตรวจจับสนสัญญาณไซน์

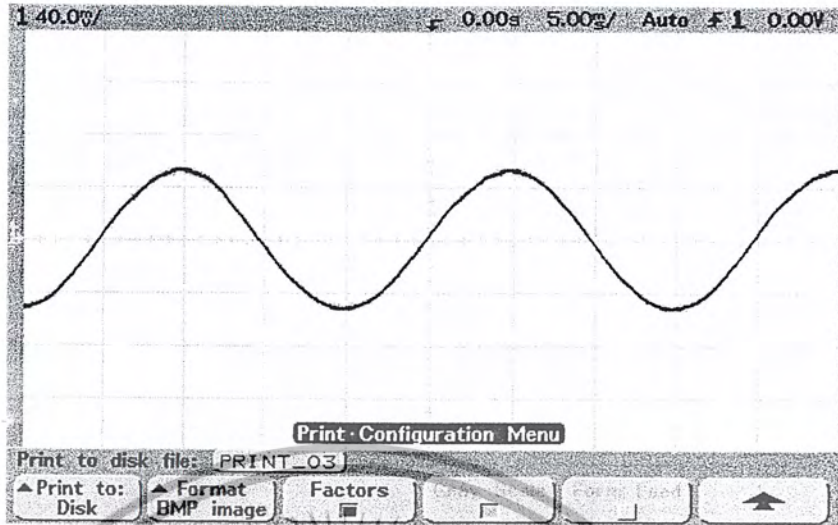
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบสัญญาณระหว่างการจำลองโดยใช้โปรแกรม ORCAD กับวัดสัญญาณจริงของแรงดันขายน



(a)

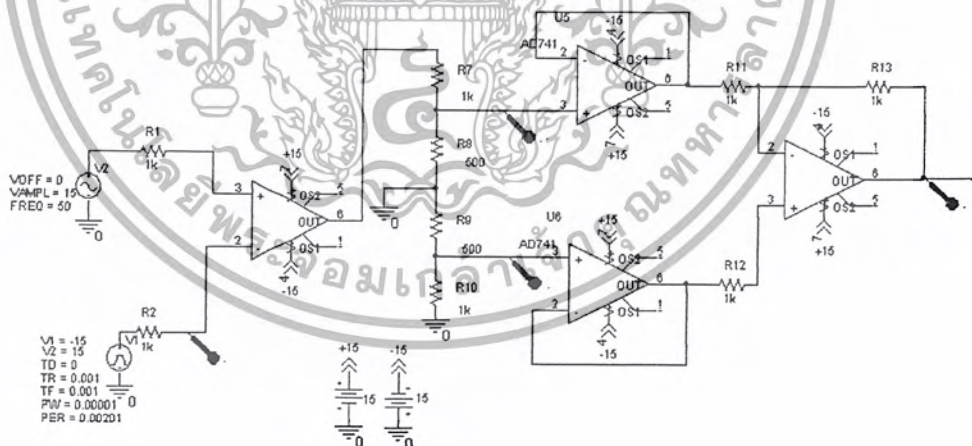
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b)

รูปที่ 4.9 (a),(b) แสดงรูปคลื่นของวงจรตรวจจับแรงดันชานน์

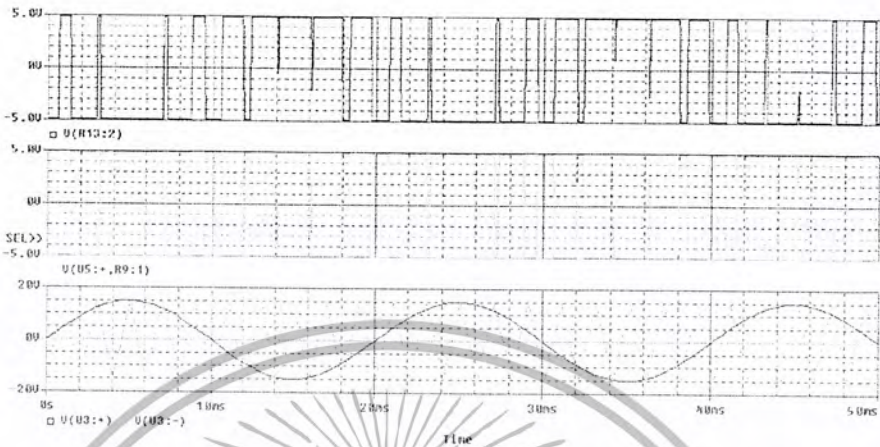
4.1.4 จำลองการวัดแรงดันโดยใช้สัญญาณอินพุตเป็นพัลส์บิวดัม (PWM)



รูปที่ 4.10 แสดงวงจรจำลองตรวจจับแรงดันที่ดับบิวดัม (PWM)

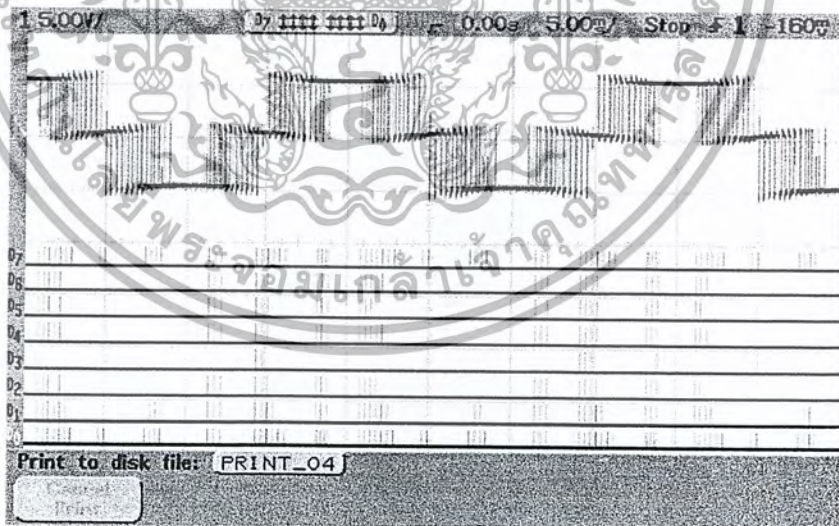
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบสัญญาณระหว่างการจำลองโดยใช้โปรแกรม ORCAD กับวัดสัญญาณจริงของแรงดันที่คัตบิวเอ็ม (PWM)



รูปที่ 4.11 แสดงรูปคลื่นของวงจรจำลองตรวจับแรงดันคัตบิวเอ็ม (PWM)

ทดสอบสัญญาณแรงดันคัตบิวเอ็ม (PWM) ด้วยอินเวอร์เตอร์ (INVERTER) โดยใช้ความถี่ fundamental 50 Hz

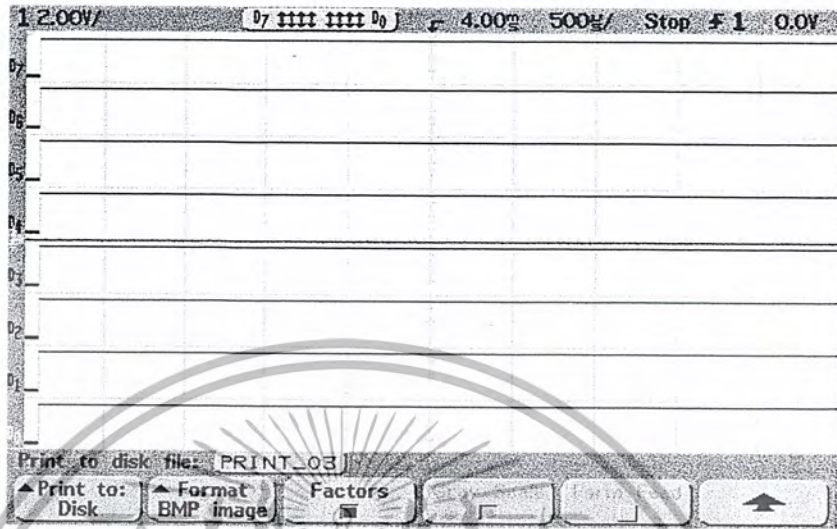


รูปที่ 4.12 แสดงรูปคลื่นของวงจรตรวจับแรงดันคัตบิวเอ็ม (PWM) และสัญญาณที่ออกจาก A/D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

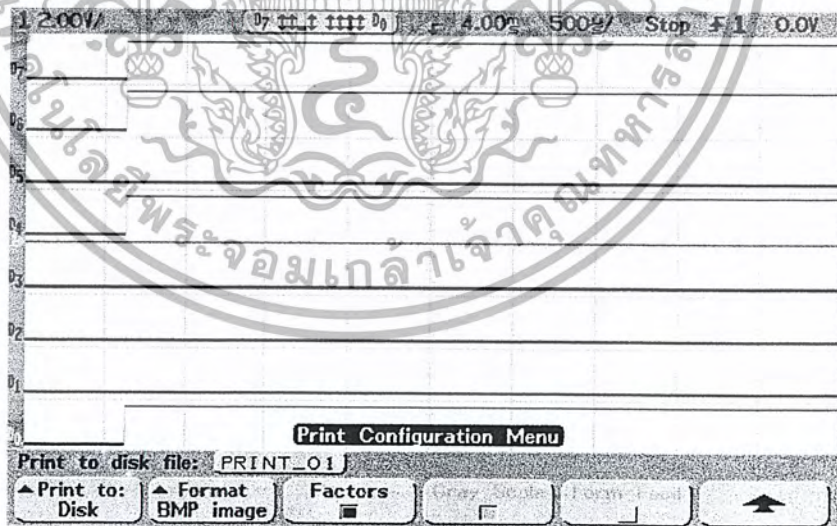
4.2 ทดลอง ADC0820 8-bit โดยใช้ Agilent 54622D Mixed-Signal Oscilloscopes

แรงดันคี่ซี (DC) 2.46 V มีลักษณะสัญญาณลอจิก 1111 1111



รูปที่ 4.13 แสดงรูปคลื่นสัญญาณของ ADC0820 ที่ 2.46 V

แรงดัน คี่ซี (DC) 2.01 V มีลักษณะสัญญาณลอจิก 1101 0001

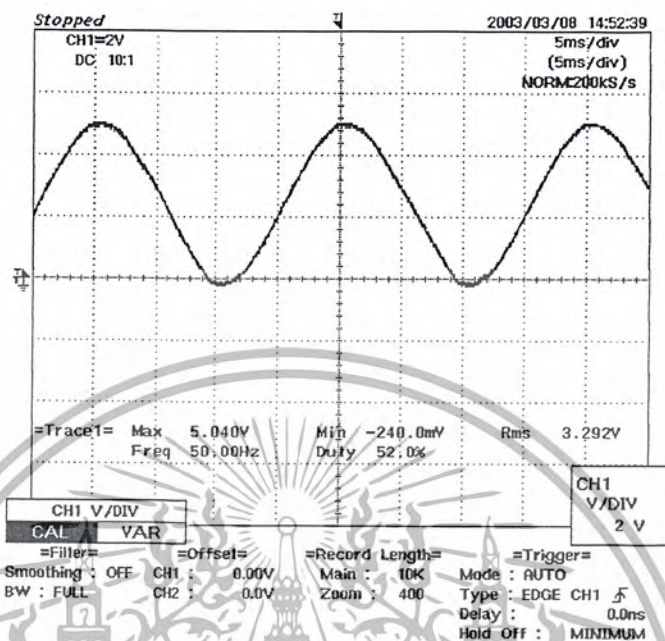


รูปที่ 4.14 แสดงรูปคลื่นสัญญาณของ ADC0820 ที่ 2.01 V

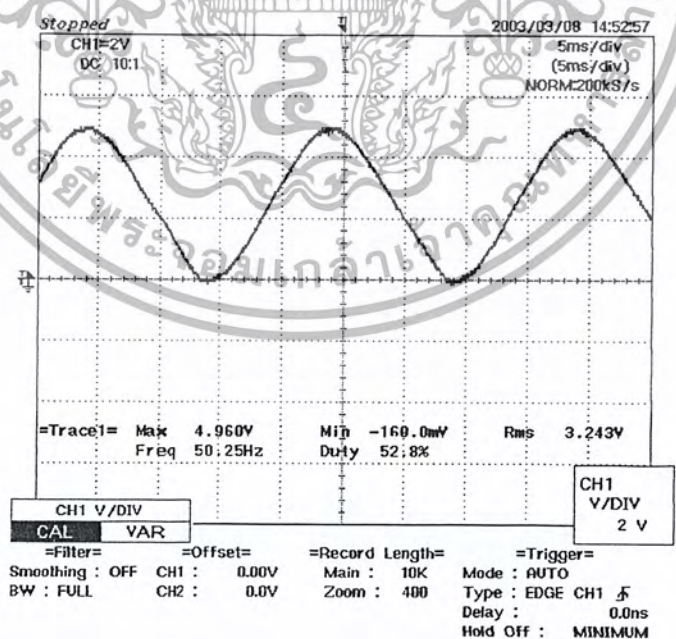
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ทดสอบสัญญาณก่อนเข้า ADC0820

4.3.1 ทดสอบด้วยสัญญาณชานัน



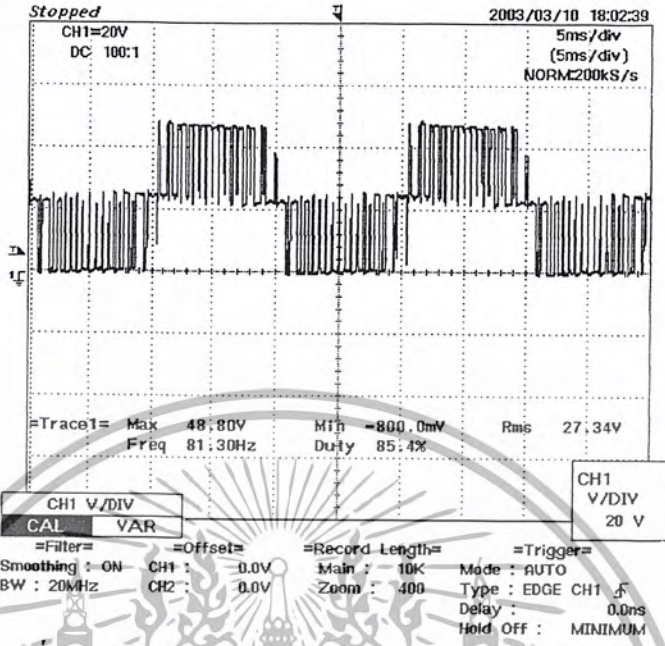
รูปที่ 4.15 สัญญาณแรงดันชานันก่อนเข้า ADC0820



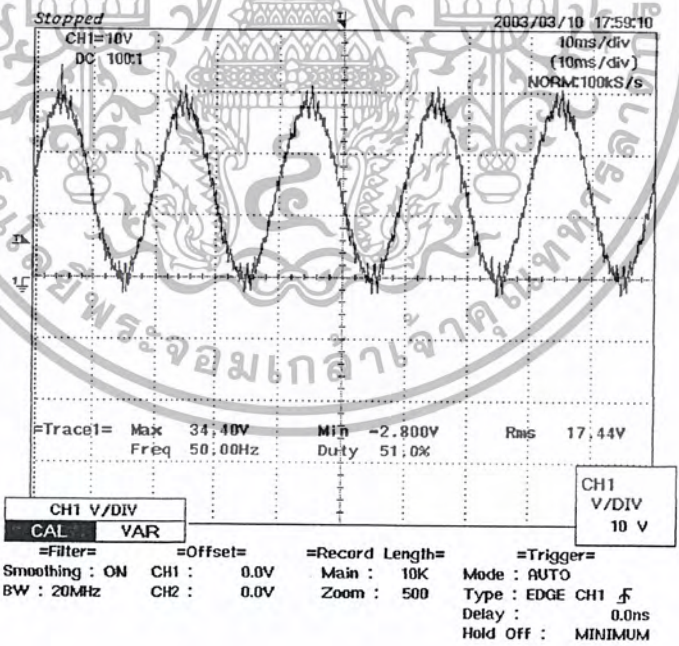
รูปที่ 4.16 สัญญาณกระแสชานันก่อนเข้า ADC0820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1 ทดสอบด้วยสัญญาณพีดับบีวเอ็ม (PWM)



รูปที่ 4.17 สัญญาณแรงดันที่ดับบีวเอ็ม(PWM)ก่อนเข้า ADC0820

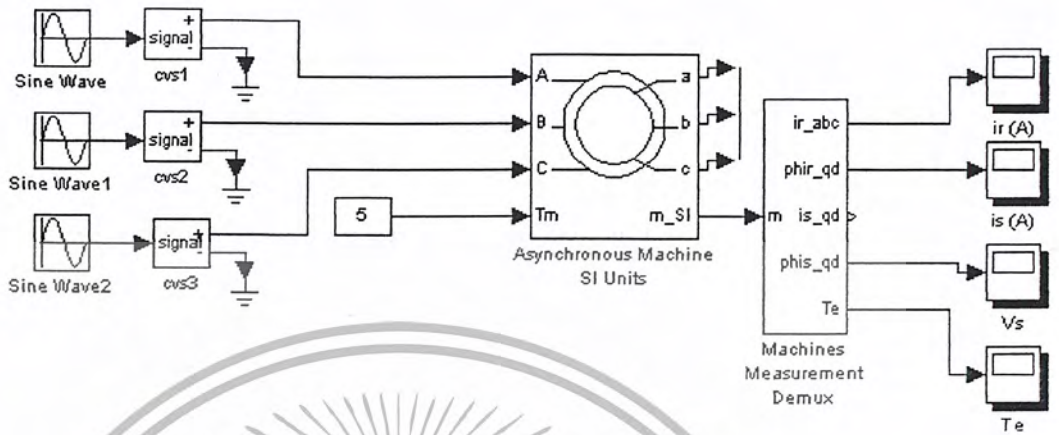


รูปที่ 4.18 สัญญาณกระแสที่ดับบีวเอ็ม(PWM)ก่อนเข้า ADC0820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

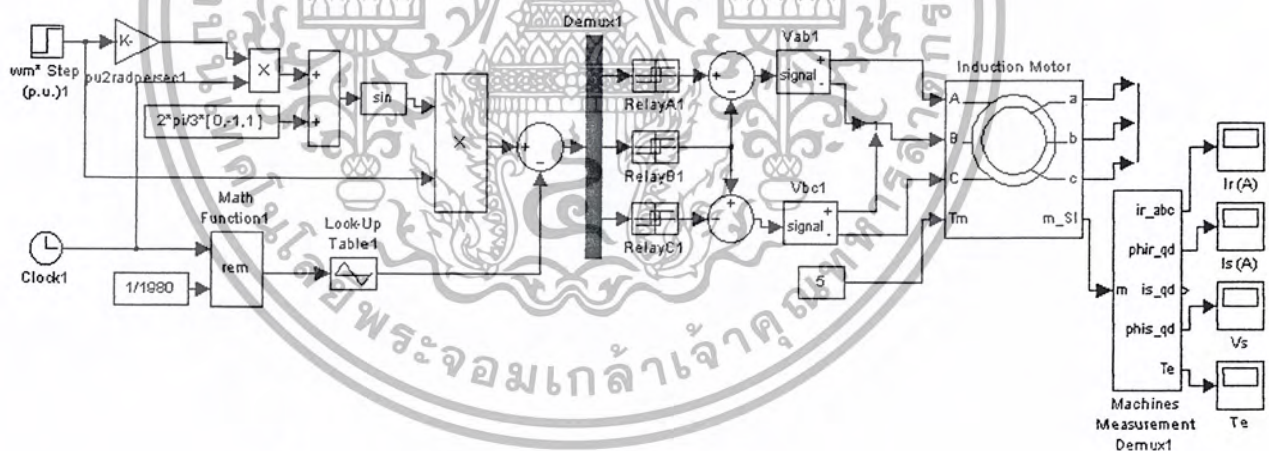
4.4 แสดงบล็อกไดอะแกรมที่ใช้ในการจำลองในโปรแกรม Matlab

สัญญาณไซน์ (SINEWAVE)



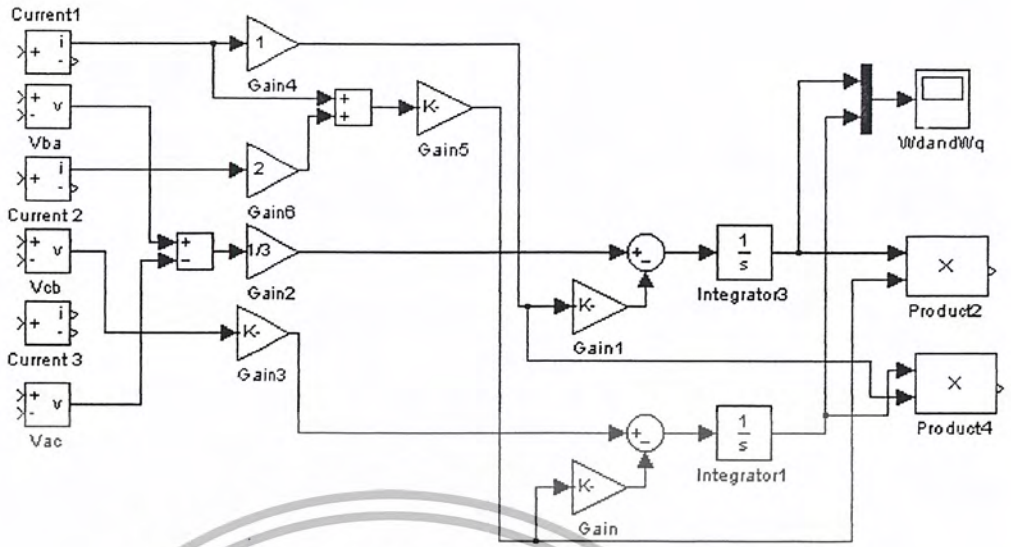
รูปที่ 4.19 แสดงบล็อกไดอะแกรมของมอเตอร์ที่ใช้สัญญาณไซน์

สัญญาณพีดีบีเอ็ม (PWM)



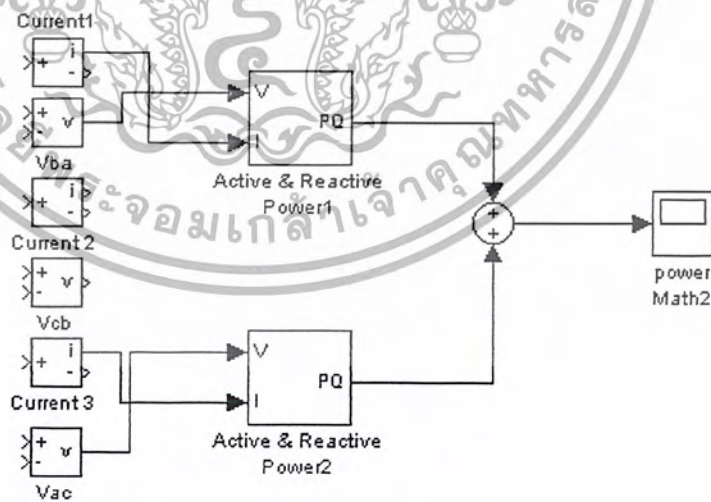
รูปที่ 4.20 แสดงบล็อกไดอะแกรมของมอเตอร์ที่ใช้สัญญาณพีดีบีเอ็ม(PWM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



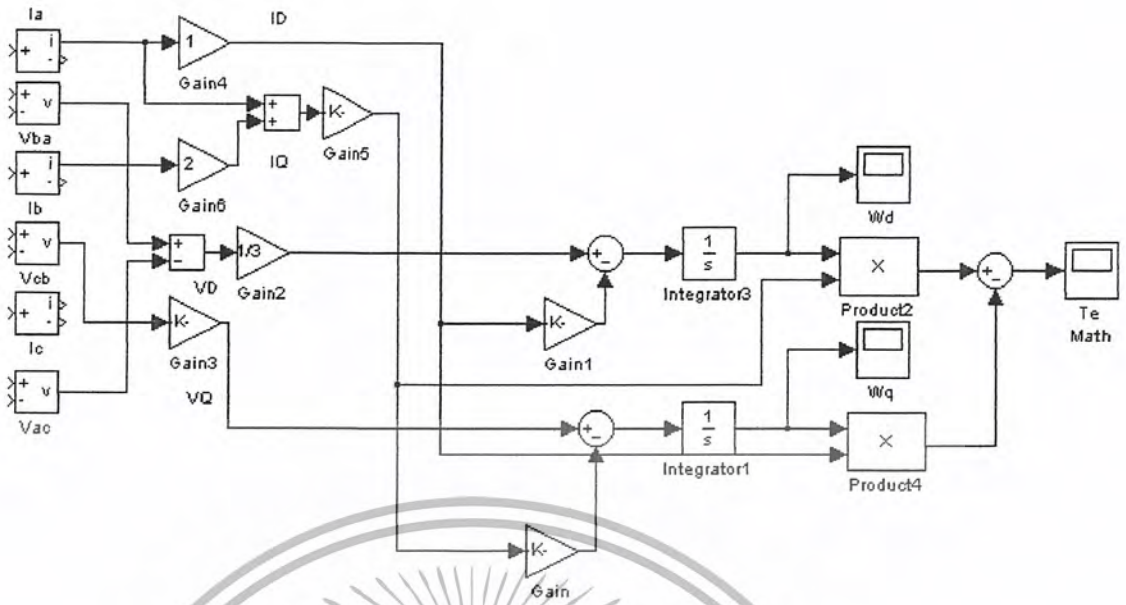
รูปที่ 4.21 แสดงบล็อกไดอะแกรมการหาค่าฟลักที (ψ_d) และฟลักทีว (ψ_q)

ซึ่งบล็อกไดอะแกรมที่สร้างมาเพื่อหาค่าฟลักทีกำลังไฟฟ้าและค่าแรงบิดของมอเตอร์นั้นได้มาจาก การเปลี่ยนแรงดันและกระแส 3 เฟส 3 แกน มาเป็น 2 แกน ซึ่งสมการ ได้แสดงไว้แล้วในบทที่ 2 ในหัวข้อ 2.2



รูปที่ 4.22 แสดงบล็อกไดอะแกรมการหาค่ากำลังไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

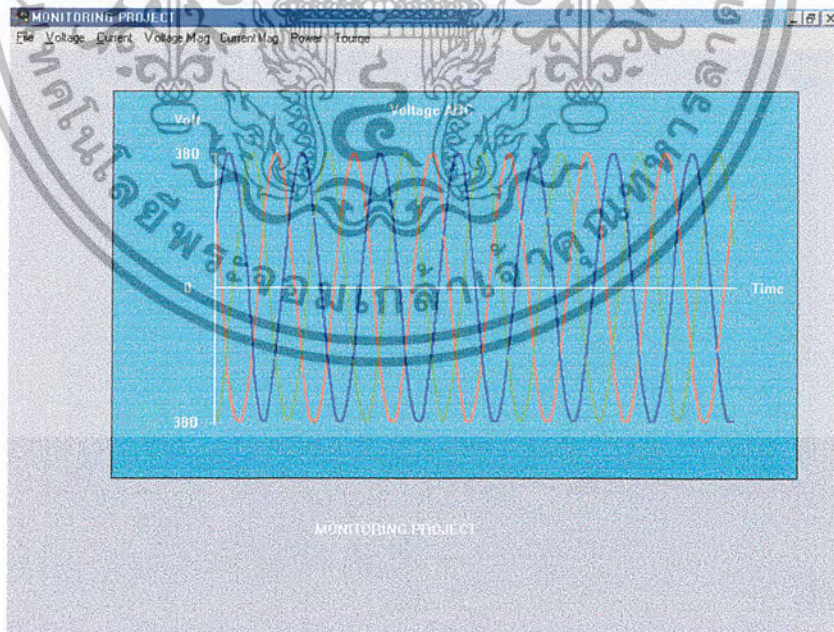


รูปที่ 4.23 แสดงบล็อกไดอะแกรมการหาค่าแรงบิด

4.5 เปรียบเทียบสัญญาณระหว่างการจำลองในโปรแกรม Matlab กับสัญญาณที่ได้จากโรงงาน

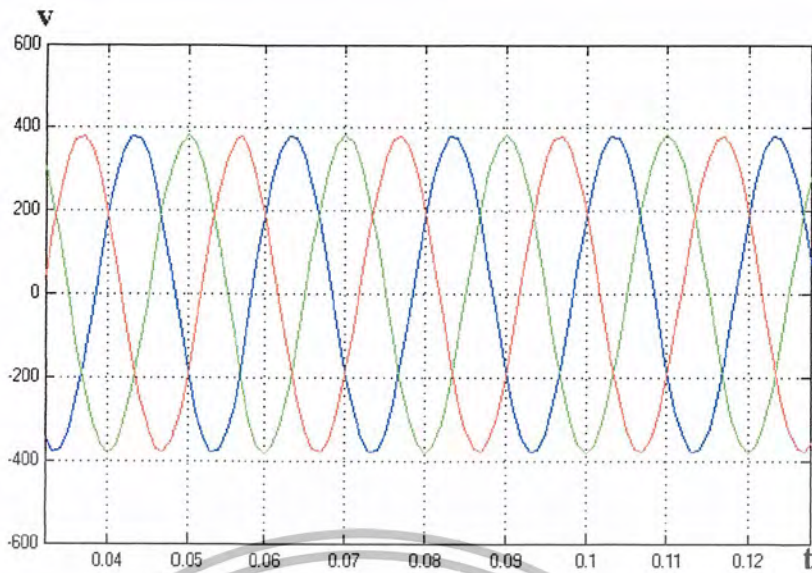
โดยแสดงทางจอกอมพิวเตอร์และสัญญาณจากสตอเรจ โค็บ ซึ่งการเปรียบเทียบจะทำการเปรียบเทียบทั้งสัญญาณขาเข้าและสัญญาณพีคดับบีวเอ็ม(PWM)

4.5.1 พิจารณาสัญญาณขาเข้า

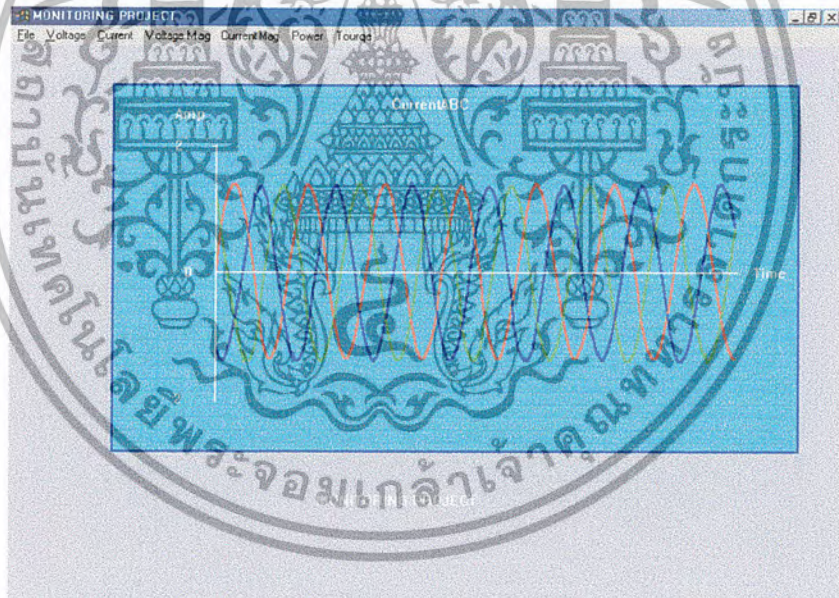


รูปที่ 4.24 แสดงสัญญาณแรงดัน ABC ที่ได้ทางคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

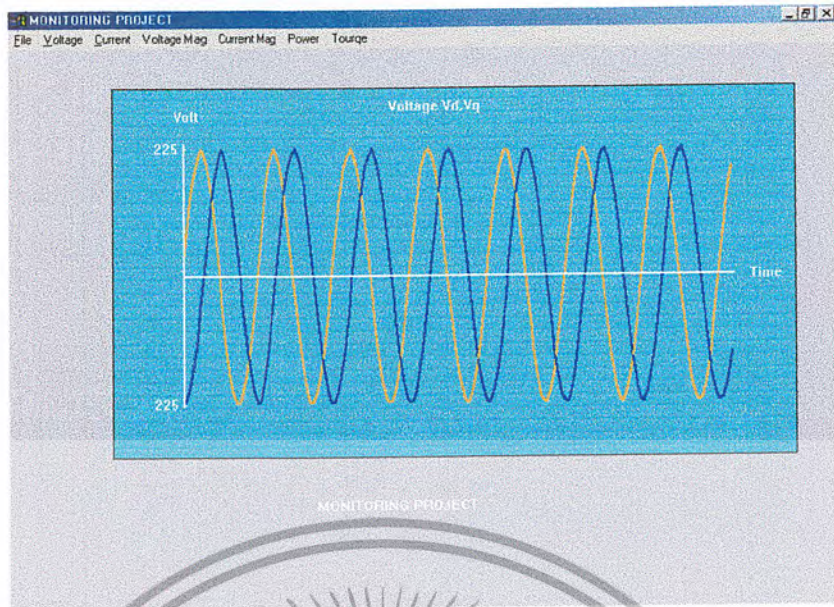


รูปที่ 4.25 แสดงสัญญาณแรงดัน ABC ที่ได้จากการจำลองใน Matlab

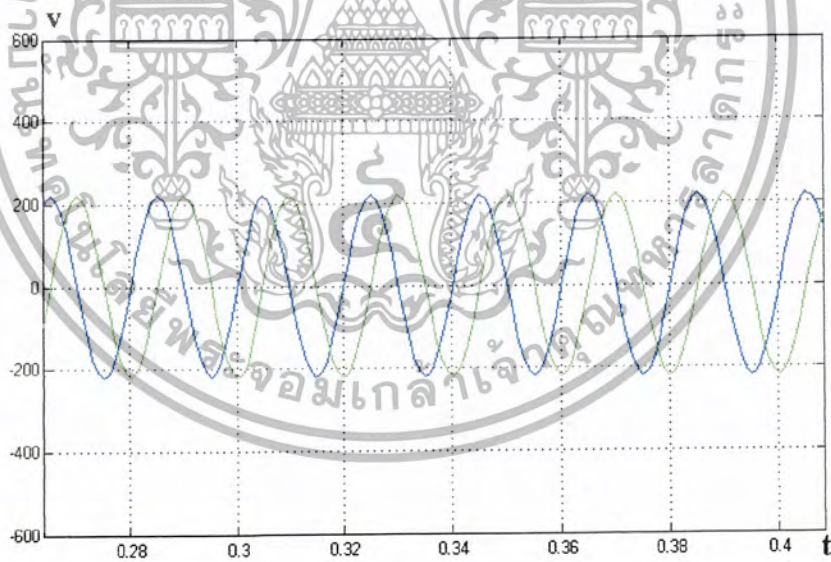


รูปที่ 4.26 แสดงสัญญาณกระแส ABC ที่ได้ทางคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

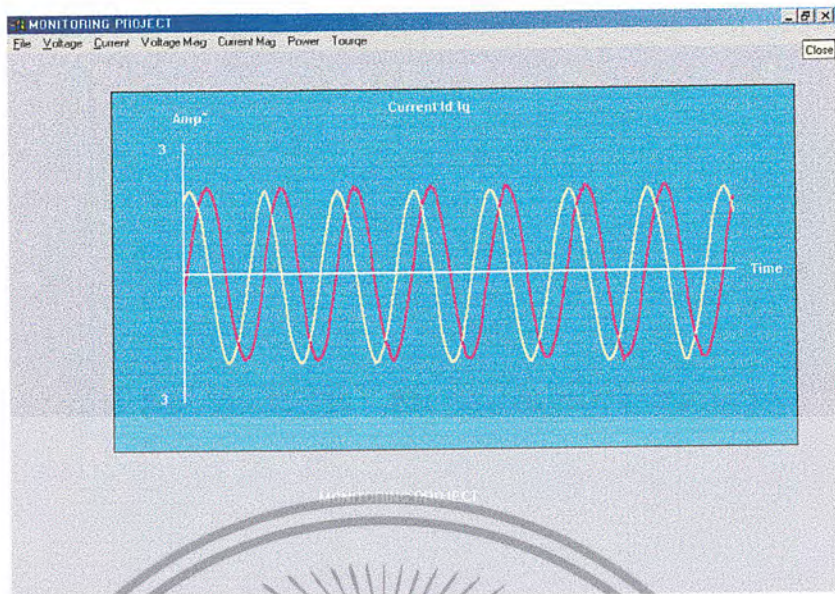


รูปที่ 4.27 แสดงสัญญาณแรงดัน V_d, V_q ที่ได้ทางคอมพิวเตอร์

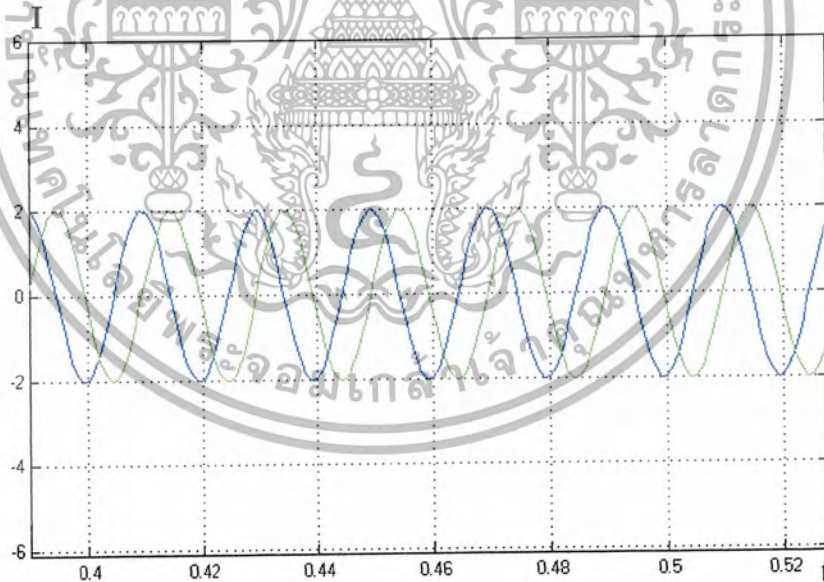


รูปที่ 4.28 แสดงสัญญาณแรงดัน V_d, V_q ที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

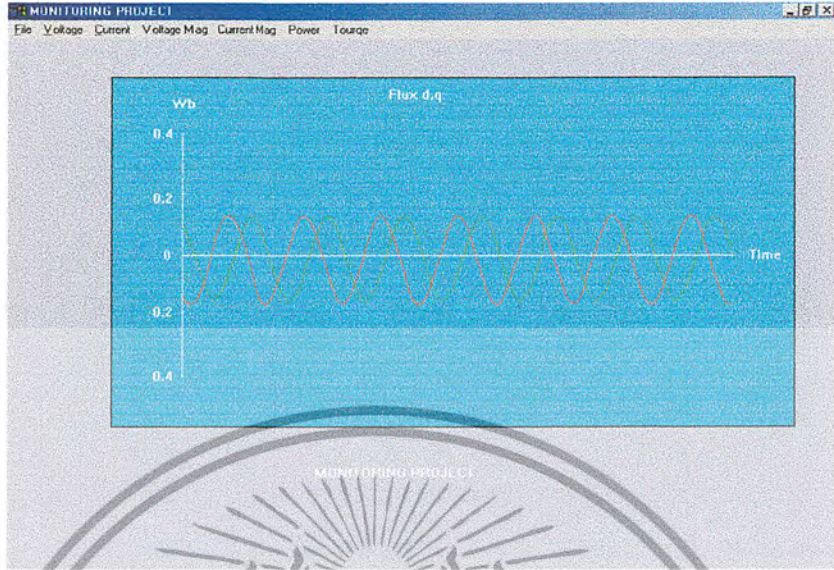


รูปที่ 4.29 แสดงสัญญาณกระแส I_d, I_q ที่ได้ทางคอมพิวเตอร์

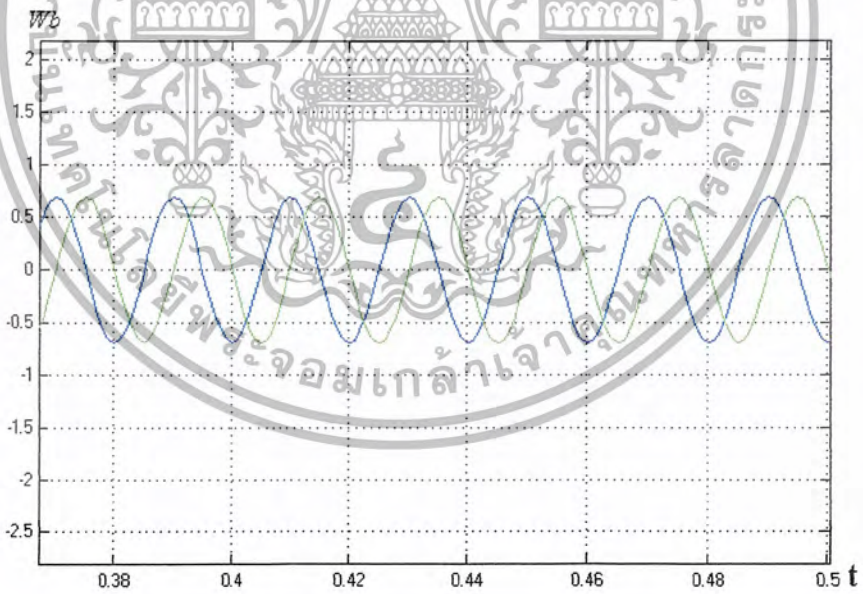


รูปที่ 4.30 แสดงสัญญาณกระแส I_d, I_q ที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



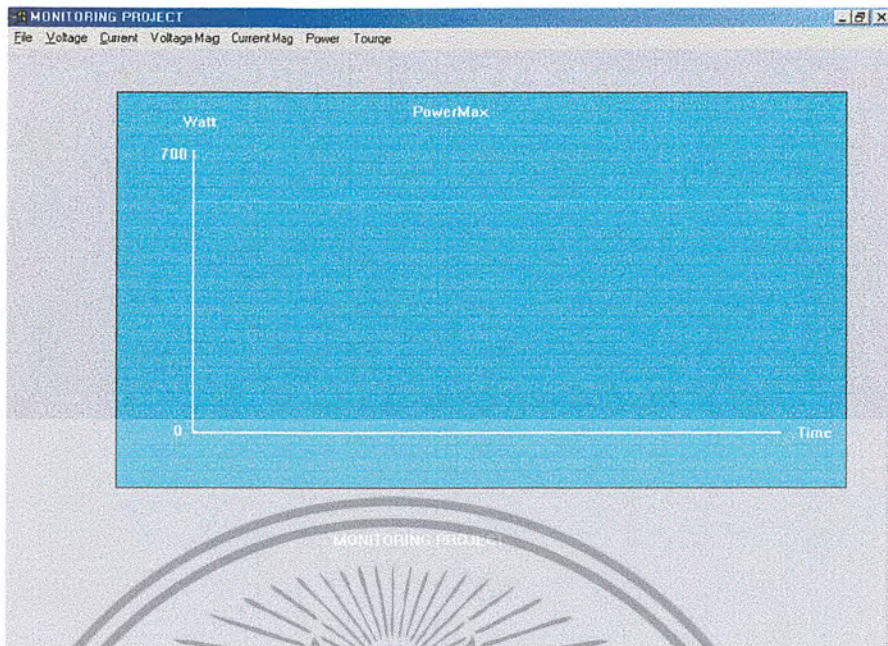
รูปที่ 4.31 แสดงสัญญาณฟลักต์ (ψ_d, ψ_q) ที่ได้ทางคอมพิวเตอร์



รูปที่ 4.32 แสดงสัญญาณฟลักต์ (ψ_d, ψ_q) ที่ได้จากการจำลองใน Matlab

ลักษณะสัญญาณของ (ψ_d, ψ_q) มีมุมต่างเฟสกัน 90 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

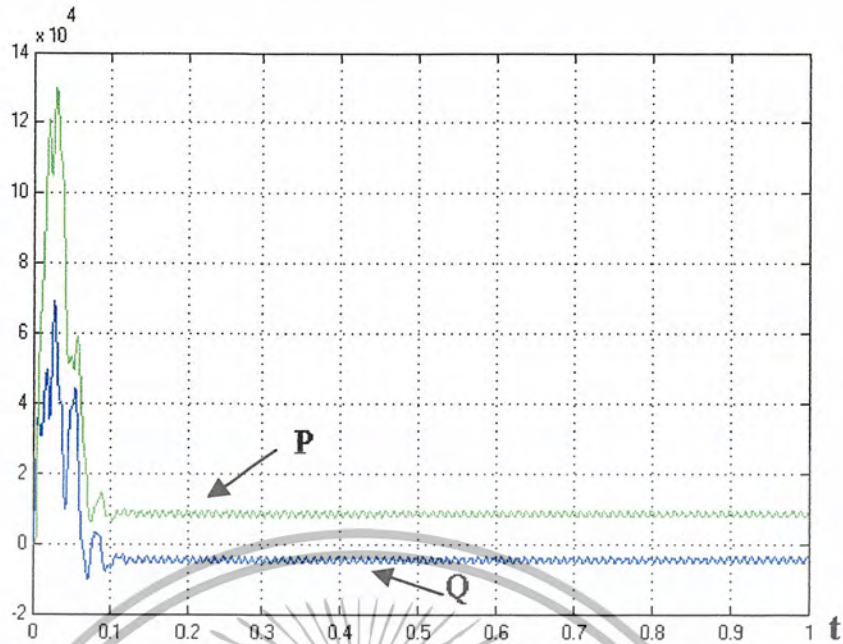


รูปที่ 4.33 แสดงสัญญาณกำลังไฟฟ้าที่ได้ทางคอมพิวเตอร์(ขณะไม่มีโหลด)



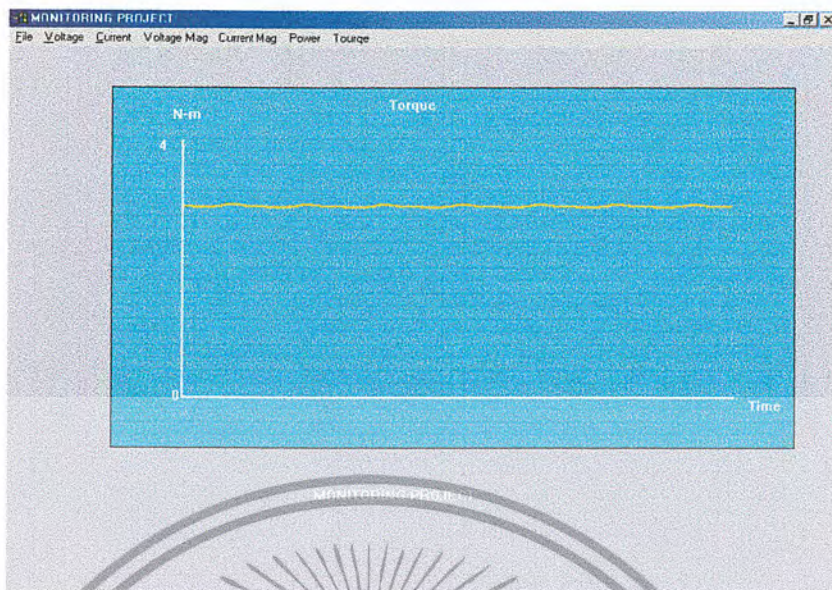
รูปที่ 4.34 แสดงสัญญาณกำลังไฟฟ้าที่ได้ทางคอมพิวเตอร์(ขณะมีโหลด 26 นิวตัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

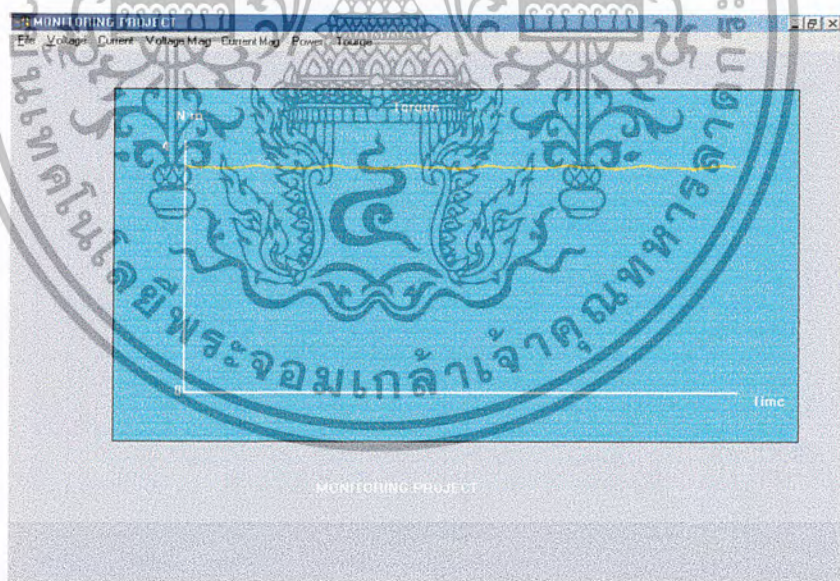


รูปที่ 4.35 แสดงสัญญาณกำลังไฟฟ้าที่ได้จากการจำลองใน Matlab

ค่ากำลังไฟฟ้าที่ได้จากการคำนวณโดยโปรแกรมคอมพิวเตอร์ ขณะมีโหลดและไม่มีโหลดจะเห็นได้ว่าจะเกิดการเปลี่ยนแปลงของกำลังไฟฟ้าอย่างเห็นได้ชัด กำลังไฟฟ้าจะมีค่าเพิ่มสูงขึ้น เมื่อเปรียบเทียบกับค่ากำลังไฟฟ้าที่วัดจากวัดมิเตอร์ขณะ No load วัดได้ 580 W ขณะมีโหลดวัดได้ 685 W ในขณะมีโหลดรูปคลื่นของกระแสและแรงดันมีการเปลี่ยนแปลงไม่ค่อยมากจึงสังเกตเห็นการเปลี่ยนแปลงที่น้อยมาก เนื่องจากค่าความละเอียดของ A/D แค่ 8 บิต ถ้าหากเพิ่มความละเอียดของ A/D โดยให้จำนวนบิตมากๆ ค่าที่ได้ก็จะมีความละเอียดมากขึ้นและ อัตราส่วนของการลดทอนของแรงดันและกระแสค่อนข้างมากถ้าเกิดการเปลี่ยนแปลงเพียงน้อยนิดจึงไม่เกิดการเปลี่ยนแปลงที่เห็นได้ชัดเจน

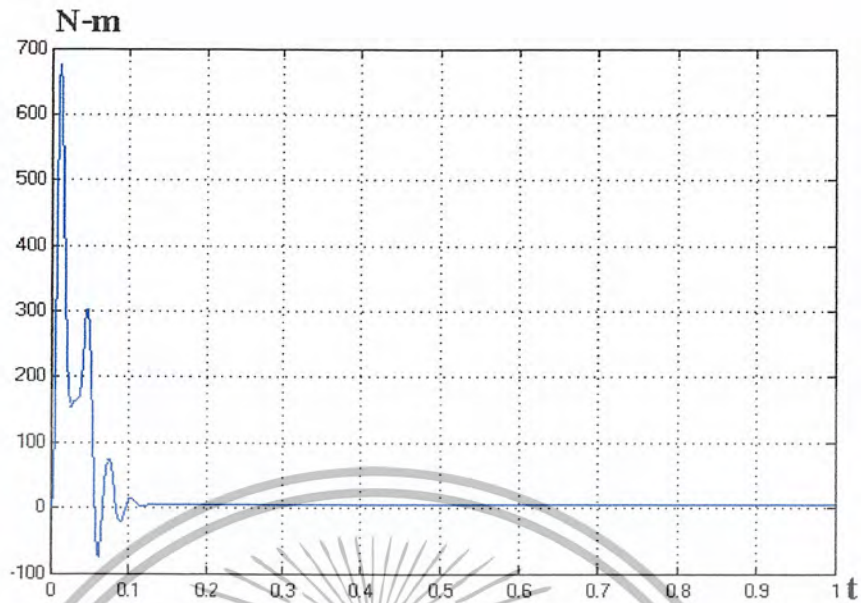


รูปที่ 4.36 แสดงสัญญาณแรงบิดที่ได้ทางคอมพิวเตอร์(ขณะไม่มีโหลด)



รูปที่ 4.37 แสดงสัญญาณแรงบิดที่ได้ทางคอมพิวเตอร์(ขณะมีโหลด 26 นิวตัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.38 แสดงสัญญาณแรงบิดที่ได้จากการจำลองใน Matlab

สัญญาณแรงบิด (Torque) ในรูปที่ 4.36 เป็นสัญญาณขณะไม่มีภาระโหลด รูปที่ 4.37 เป็นสัญญาณขณะมีภาระต่อโหลด ซึ่งได้จากการคำนวณของคอมพิวเตอร์ ซึ่งแรงบิดขณะไม่มีโหลดคือ 2.8 N.m และแรงบิดขณะมีโหลดคือ 3.8 N.m (ผลจากการทดลอง) ซึ่งมีค่าใกล้เคียงกัน

สัญญาณแรงบิดและกำลังไฟฟ้าที่ได้จากโครงการไม่สามารถจับสถานะเริ่มต้น(transient)ได้จะสามารถจับได้เฉพาะช่วงคงที่(steady state)ส่วนในการจำลองใน Matlab นั้นสามารถจับได้ตั้งแต่สถานะเริ่มต้น

ผลจากการวัดค่ากำลังไฟฟ้า(Power)และแรงบิด(Torque)ของ 3 เฟส Induction Motor โดยเครื่อง (FLUKE 41B POWER HARMONICS ANALYZER)

ขณะไม่มีภาระจ่ายภาระ โหลด

กระแสไฟฟ้า = 1.2 Amp

แรงดันไฟฟ้า = 390 Volt

กำลังไฟฟ้า 3 เฟส = 570 Watt

แรงบิด = 3 นิวตัน

ขณะทำการจ่ายภาระ โหลดที่ 25 นิวตัน

กระแสไฟฟ้า = 1.3 Amp

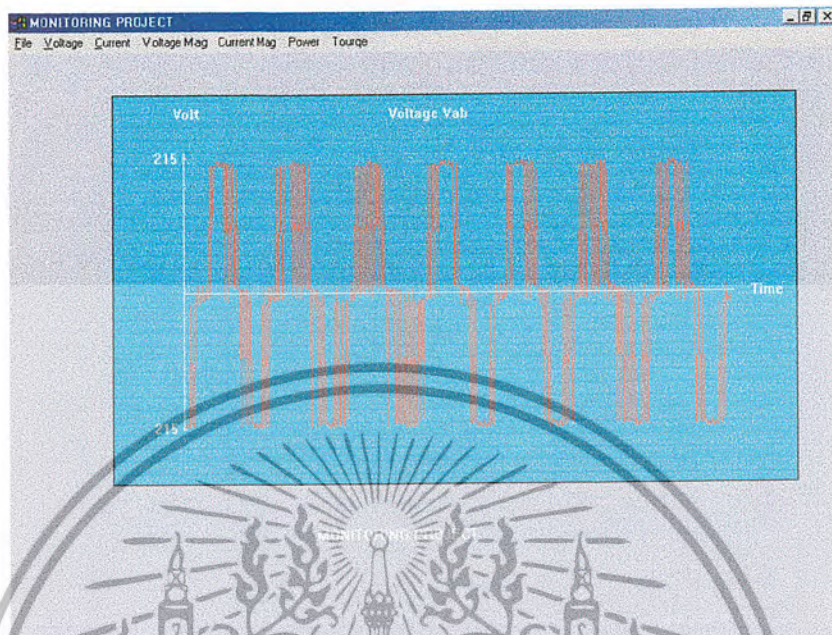
แรงดันไฟฟ้า = 387 Volt

กำลังไฟฟ้า 3 เฟส = 705 Watt

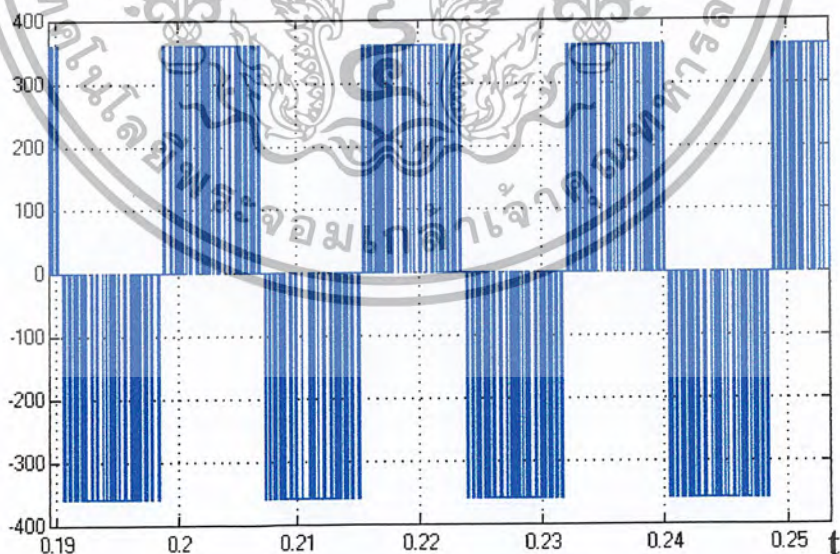
แรงบิด = 3.83 นิวตัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 พิจารณาสัญญาณพีดับิวเอ็ม (PWM)

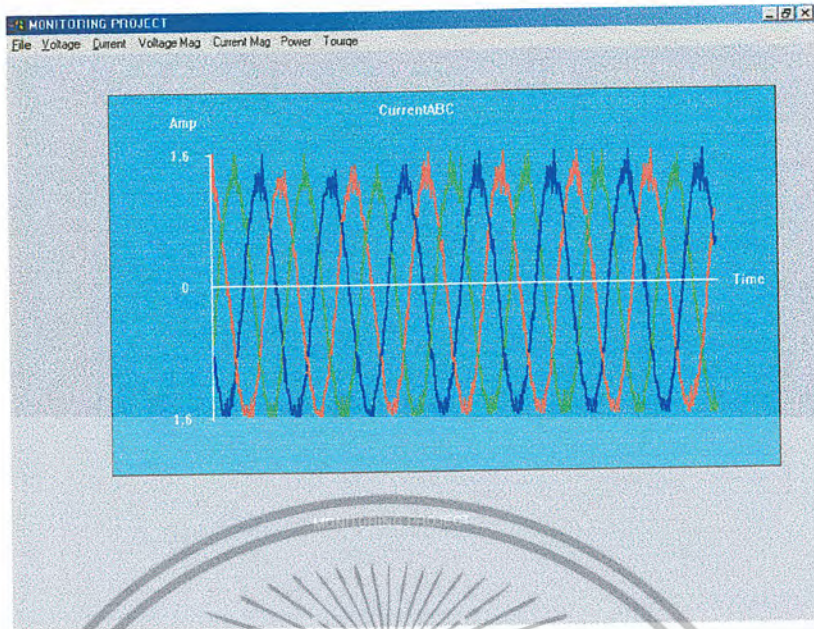


รูปที่ 4.39 แสดงสัญญาณแรงดันที่ดับิวเอ็ม(PWM)ที่ได้ทางคอมพิวเตอร์

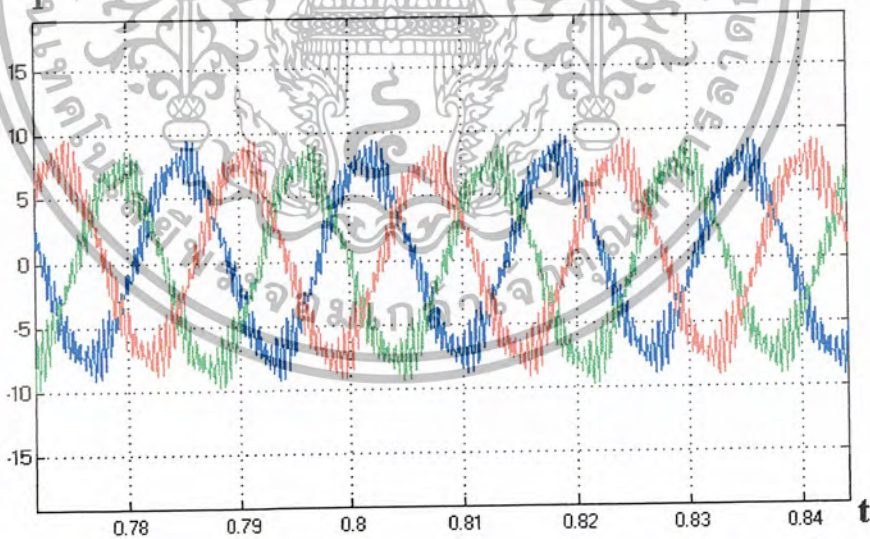


รูปที่ 4.40 แสดงสัญญาณแรงดันที่ดับิวเอ็ม(PWM)ที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

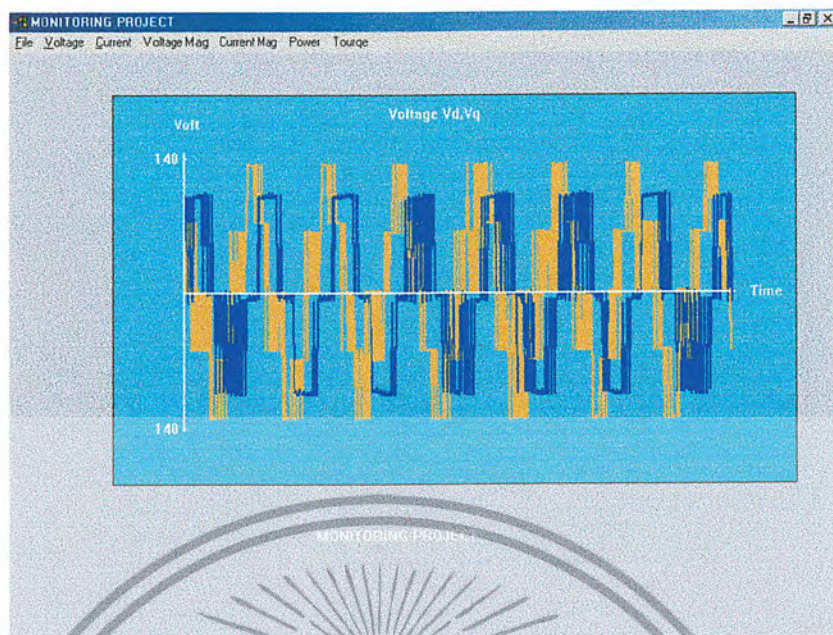


รูปที่ 4.41 แสดงสัญญาณกระแสที่ดับนิวเอม(PWM)ที่ได้ทางคอมพิวเตอร์

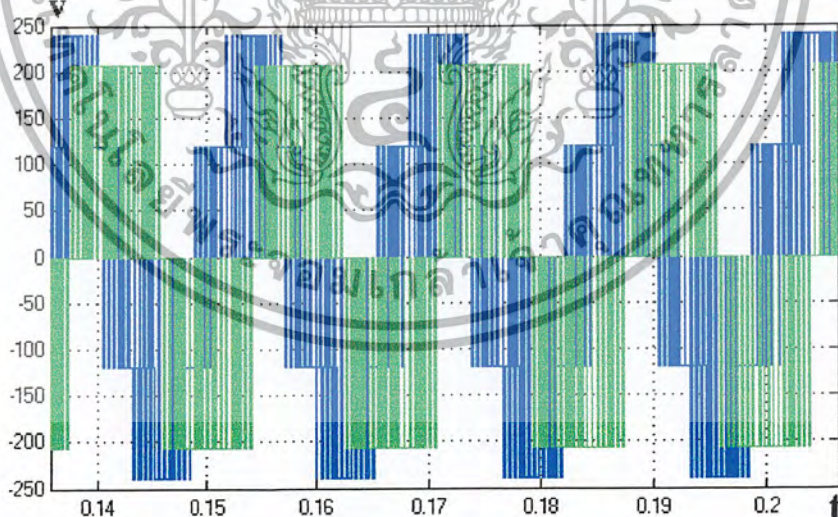


รูปที่ 4.42 แสดงสัญญาณกระแสที่ดับนิวเอม(PWM)ที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

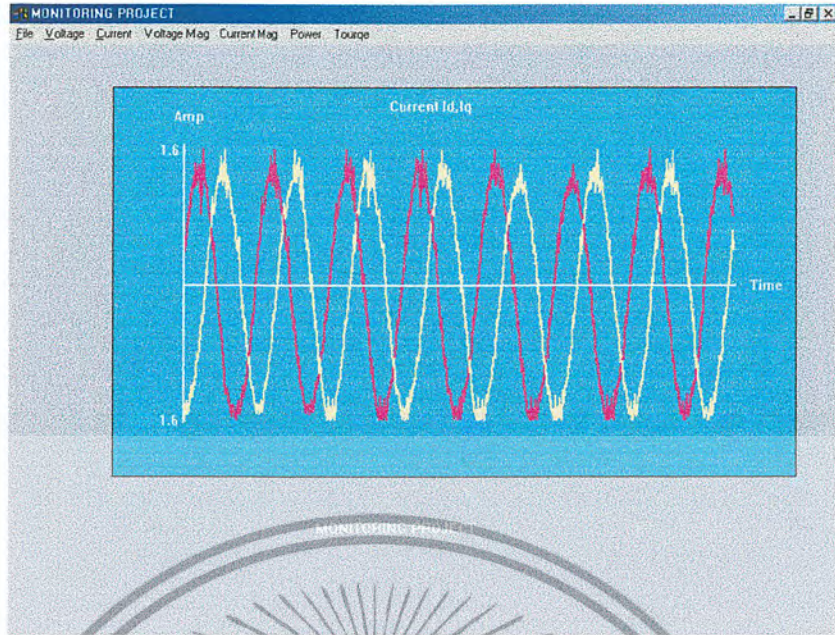


รูปที่ 4.43 แสดงสัญญาณแรงดัน (V_d, V_q) ที่ดับนิวเอมที่ได้ทางคอมพิวเตอร์

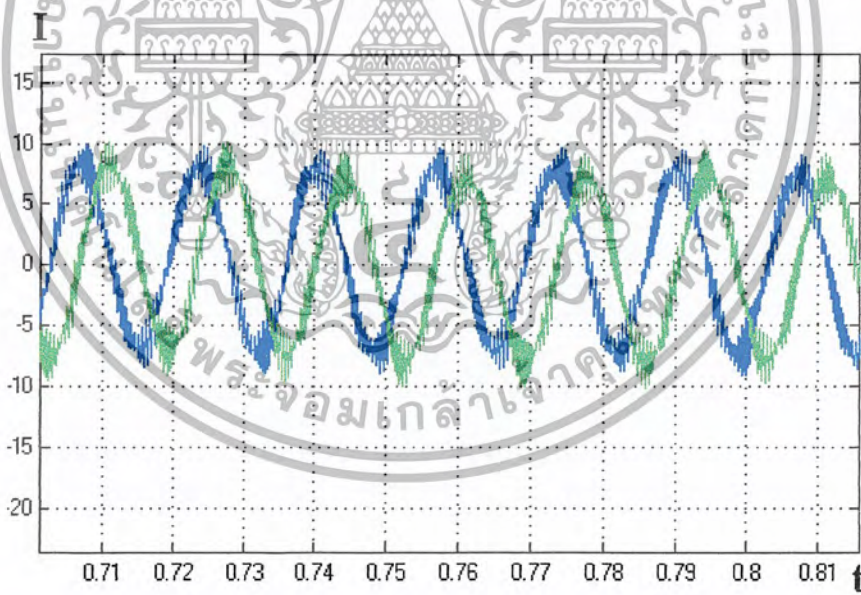


รูปที่ 4.44 แสดงสัญญาณแรงดัน (V_d, V_q) ที่ดับนิวเอมที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

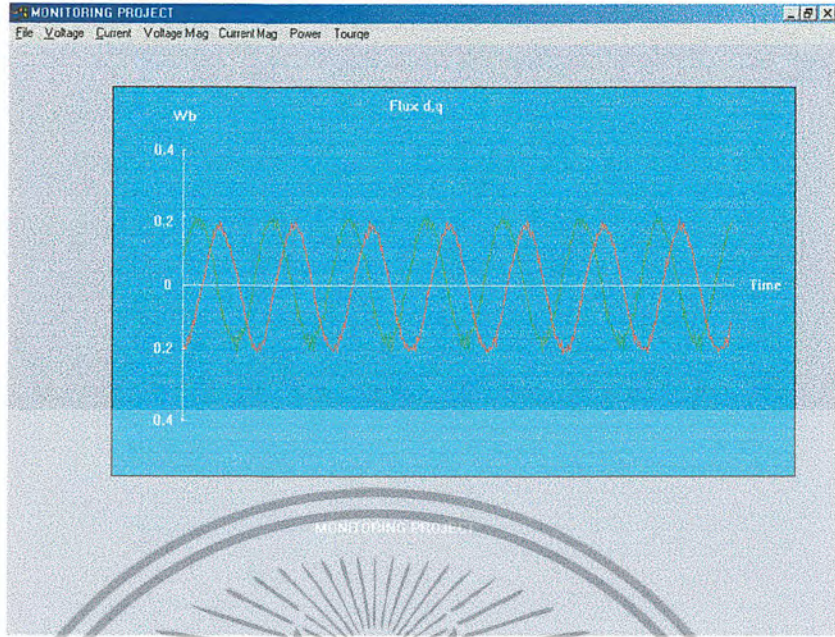


รูปที่ 4.45 แสดงสัญญาณกระแส (I_d, I_q) ที่ดับบีวเอ็มที่ได้ทางคอมพิวเตอร์

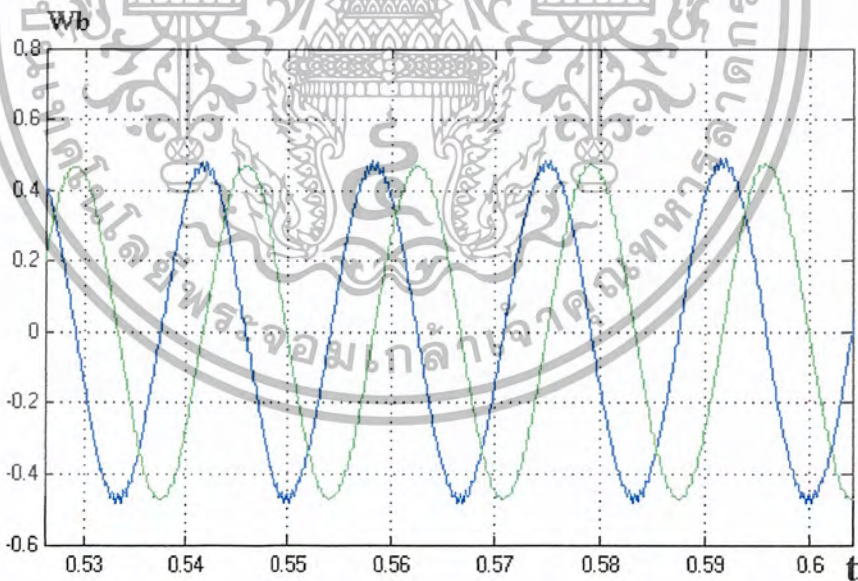


รูปที่ 4.46 แสดงสัญญาณกระแส (I_d, I_q) ที่ดับบีวเอ็มที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

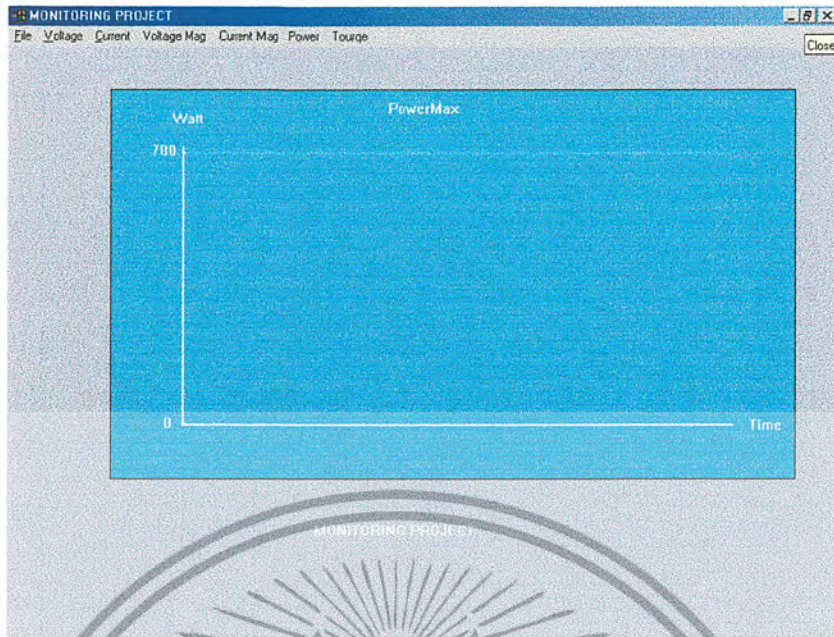


รูปที่ 4.47 แสดงสัญญาณฟลักซ์, คิว (ψ_D, ψ_Q) ที่ได้จากทางคอมพิวเตอร์

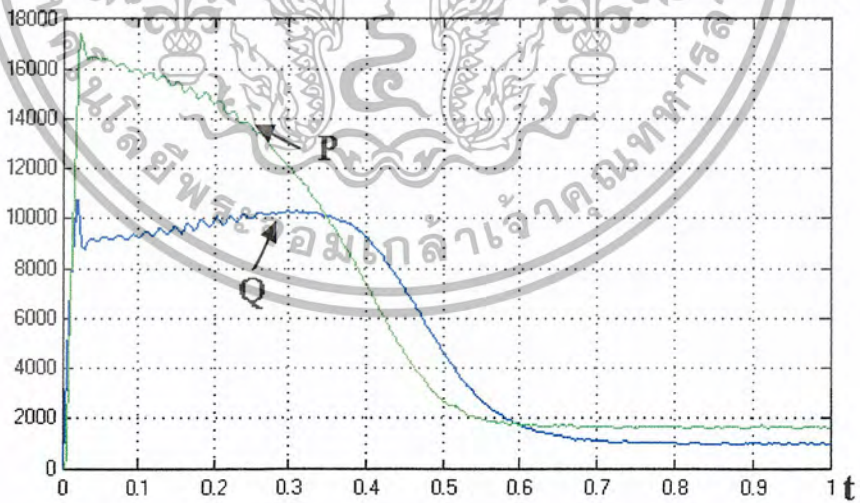


รูปที่ 4.48 แสดงสัญญาณฟลักซ์, คิว (ψ_D, ψ_Q) ที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

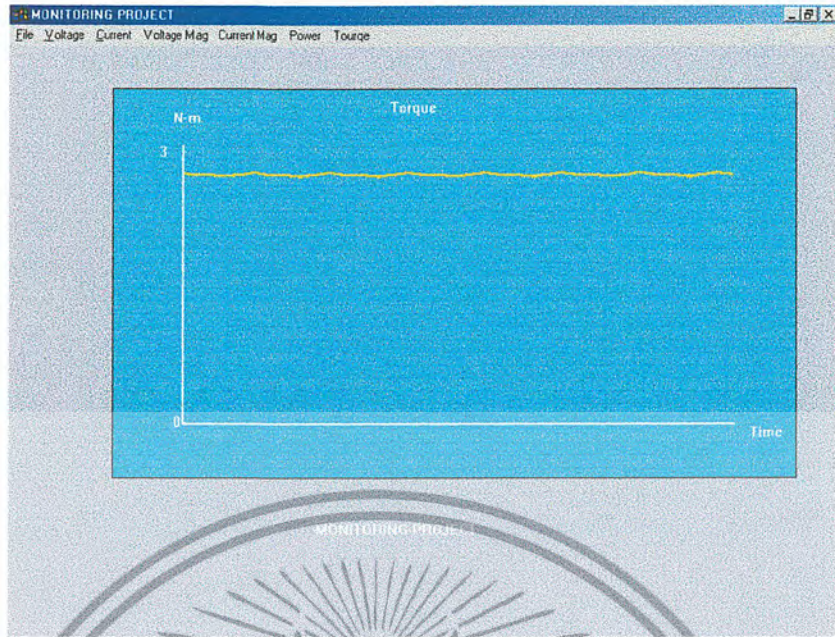


รูปที่ 4.49 แสดงสัญญาณกำลังไฟฟ้าที่ดับบิวเอ็ม(PWM) ที่ได้จากทางคอมพิวเตอร์

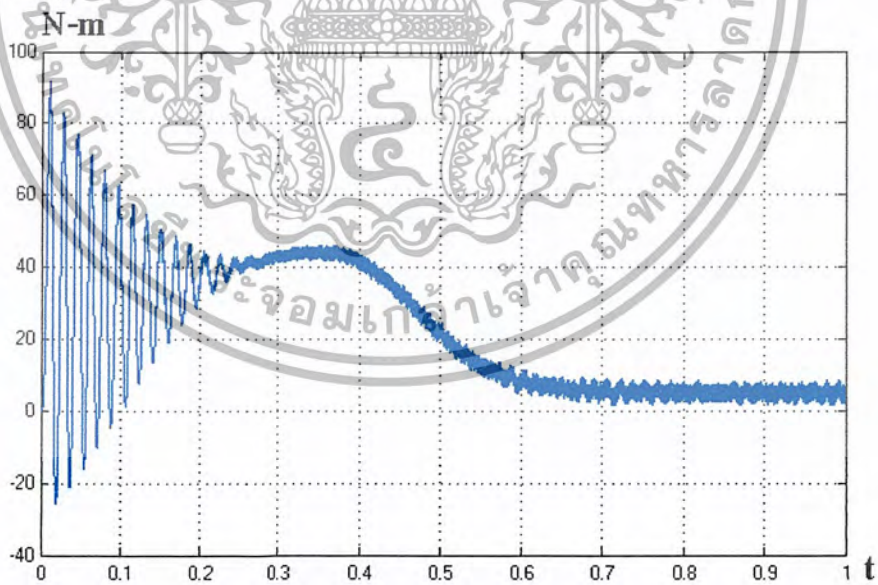


รูปที่ 4.50 แสดงสัญญาณกำลังไฟฟ้าที่ดับบิวเอ็ม(PWM) ที่ได้จากการจำลองใน Matlab

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.51 แสดงสัญญาณแรงบิดที่ดับบีวเอ็ม(PWM) ที่ได้จากทางคอมพิวเตอร์



รูปที่ 4.52 แสดงสัญญาณแรงบิดที่ดับบีวเอ็ม(PWM) ที่ได้จากการจำลองใน Matlab

สัญญาณแรงบิดและกำลังไฟฟ้าที่ได้จากโครงการไม่สามารถจับสถานะเริ่มต้น(transient)ได้จะสามารถจับได้เฉพาะช่วงคงที่(steady state)ส่วนในการจำลองใน Matlab นั้นสามารถจับได้ตั้งแต่สถานะเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์และสรุป

จากการทำโครงการสามารถแบ่งหมวดงานออกเป็น 4 ส่วน ซึ่งสามารถสรุปผลงานและความคืบหน้าของงานได้ดังต่อไปนี้

1. ส่วนของวงจรตรวจจับสัญญาณ

ในส่วนของวงจรตรวจจับสัญญาณสามารถแบ่งออกได้ 2 ส่วน

5.1.1 ส่วนตรวจจับสัญญาณแรงดัน ในส่วนของวงจรตรวจจับสัญญาณแรงดันจะใช้หลักการของวงจรแบ่งแรงดันโดยใช้ความต้านทาน มาต่ออนุกรมเพื่อแบ่งแรงดันจาก 380 และ 220 โวลต์ ลดทอนเหลือ 5 โวลต์ โดยเลือกใช้ค่าความต้านทานที่มีค่าความผิดพลาด 2.5% สัญญาณที่วัดจึงมีความถูกต้อง

5.1.2 ส่วนตรวจจับสัญญาณกระแส เพื่อนำสัญญาณไปประมวลผลจะเลือกใช้ LEM เป็นโมดูลซึ่งแปลงสัญญาณกระแสโดยอาศัยหลักการของปรากฏการณ์ฮอลล์ (Hall Effect) โดยการทำงานจะคล้ายกับหม้อแปลงกระแสทั่วไป สัญญาณกระแสที่ออกจาก LEM จะถูกแปลงเป็นสัญญาณแรงดันค่ามีค่าไม่เกิน 5 โวลต์ แต่สัญญาณที่ได้ยังไม่ถูกต้องมากนักเนื่องจากจะมีค่าออฟเซตกระแสดรบกวน เนื่องจากการป้อนไฟเลี้ยงให้กับ LEM จึงต้องเพิ่มวงจร Summing ที่รับค่าได้เพื่อใช้ตัดค่าออฟเซตกระแสดรบกวนออกก่อน

2. ส่วนของวงจร A/D

ในส่วนของวงจร A/D ใช้ A/D เบอร์ ADC0820 ซึ่งมีอัตราการ Sampling 400 KHz ซึ่งวงจร A/D นี้จะทำการแปลง สัญญาณจาก อนุภาคเป็นสัญญาณ ดิจิตอล ก่อนที่สัญญาณจะเข้า A/D จะมีวงจรระดับสัญญาณ ให้เป็น 5 โวลต์ สัญญาณดิจิตอลที่ได้จะถูกป้อนเข้า คอมพิวเตอร์

3. การ์ดขยายพอร์ต

เนื่องจากโครงการเป็นการรับ-ส่งข้อมูลด้วยพอร์ตขนาน ข้อมูลที่รับเข้ามีหลายแอดเดรส เนื่องจากแอดเดรสจากพอร์ตขนานมีไม่เพียงพอ จึงต้องใช้การ์ดขยายพอร์ตขนาน เพื่อเพิ่มแอดเดรสในการรับข้อมูลเข้าไปในโครงการ จะใช้การ์ดขยายพอร์ต PC-8255 ซึ่งมีคุณลักษณะ สามารถขยายพอร์ตได้ถึง 9 แอดเดรส ซึ่งเพียงพอสำหรับโครงการ แต่เนื่องจากเกิดการผิดพลาดในการรับสัญญาณจึงต้องใช้ การขยายพอร์ต 2 ตัว โดยใช้ตัวละ 2 แอดเดรส การควบคุมพอร์ตนั้น จะใช้หลักการสับ Diff Swith ซึ่งแล้วแต่ความต้องการให้ทำงานที่แอดเดรส ไหน ซึ่งในการสับ Diff Swith นั้น ตำแหน่งของแอดเดรสจะต้องไม่ทับกับแอดเดรสของ การ์ดต่างๆ ของ Computer จะทำให้การ์ดไม่สามารถรับส่งข้อมูลได้

4. ส่วนของโปรแกรม

ในส่วนของการเขียนโปรแกรมนั้นเราจะ ภาษา Visual C++ ในการเขียน ซึ่งในการเขียนนั้นจะแบ่งได้เป็น

- การเขียนโปรแกรมรับค่าสัญญาณดิจิทัล โดยการเขียนโปรแกรม จะรับค่ามาจากการ์ดขยายพอร์ต โดยอาศัยแอดเดรสจากการ์ดขยายพอร์ต ซึ่งสัญญาณที่รับมานั้นจะประกอบด้วย 4 สัญญาณด้วยกันซึ่งประกอบด้วย สัญญาณแรงดัน 2 สัญญาณและสัญญาณกระแส 2 สัญญาณ
- นำสัญญาณที่นำเข้ามา 2 สัญญาณ เพื่อคำนวณหาสัญญาณที่ 3
- เมื่อได้ครบทั้ง 3 สัญญาณแล้ว นำสัญญาณที่ รับเข้ามาได้เพื่อคำนวณหาค่าต่างๆโดยอาศัยสมการที่ได้อ้างถึงเพื่อคำนวณหาค่าต่างๆ ไม่ว่าจะ เป็นสัญญาณ แรงบิด , กำลังไฟฟ้า
- สัญญาณที่คำนวณ ได้จาก การเขียนโปรแกรมนำไปเปรียบเทียบกับสัญญาณที่เกิดจากการจำลอง ในโปรแกรม Matlab เพื่อตรวจสอบความถูกต้องของสัญญาณ

ข้อจำกัดของโครงการ

เนื่องจากในส่วนของการคำนวณโดยการเขียน โปรแกรมนี้จะพบว่า โครงการยังมีปัญหาในส่วนของค่าที่นำไปแสดงนั้น เป็นการพิจารณาที่ค่าเวลาใดเวลาหนึ่ง เมื่อมีการรัน โปรแกรมการรับค่าของโปรแกรมจะรับค่า ที่เวลานั้นเท่านั้น ไม่สามารถทำให้มีการรับค่าได้ตลอดเวลา ค่าที่แสดงจึงเป็นค่าขณะที่โปรแกรมทำงานเท่านั้น ไม่ว่าจะ เป็นขณะมีโหลดหรือ ไม่มีโหลด

การพัฒนา

- โดยใช้การรับค่ามาจาก A/D แล้วใช้ DSP (Digital Signal Processing) ในการรับค่าแล้วคำนวณหาค่าต่างๆ จากการเขียนโปรแกรมในตัว DSP แล้วป้อนข้อมูลต่างๆ ที่คำนวณได้ ไปแสดงผลที่คอมพิวเตอร์โดยตรงเลย โดยการส่งข้อมูลจะเป็นการส่งข้อมูลแบบอนุกรม
- การแสดงผลจะต้องพัฒนาให้สามารถแสดงผลรูปคลื่นที่เกิดขึ้นตลอดเวลาเมื่อมีการเปลี่ยนแปลงของโหลด
- ควรพัฒนาให้สามารถ รับโหลดได้ โดยแสดงให้เห็นการเปลี่ยนแปลงที่เกิดขึ้นขณะ ไม่มีโหลดและขณะรับโหลด
- A/D ควรใช้เบอร์ที่มีความละเอียดมากๆ จะทำให้ค่าที่จับได้มีความถูกต้องมากยิ่งขึ้น
- ในส่วนของโปรแกรมแสดงผล ควรทำให้สามารถเลือกค่าที่จะทำการแสดงได้ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

จากการวางแผนการดำเนินงานตลอดระยะเวลา 1 ปี ในช่วงภาคเรียนแรกมีการวางแผนไว้ว่าจะต้องเสร็จในส่วนของ HARDWARE ทั้งหมดตลอดระยะเวลาในการวางแผนไว้วันนั้นปรากฏว่าเสร็จสิ้นตามแผนการดำเนินงานแต่ไม่สมบูรณ์ 100 % การดำเนินงานของกลุ่มมีความล่าช้ากว่าเป้าหมายที่ได้วางไว้เล็กน้อย เนื่องจากปัญหาและอุปสรรคต่างๆ แต่อยู่ในขั้นที่ดี เพราะเจอกับอุปสรรคและปัญหาหมยมาย โดยพอจะสรุปปัญหาที่พบมีดังนี้

1. เนื่องจากความขาดแคลนของอุปกรณ์เครื่องมือวัด
2. ขาดอุปกรณ์ทางอิเล็กทรอนิกส์
3. เครื่องมือที่เกี่ยวข้องกับการทดลอง
4. ทฤษฎีที่ค้นคว้าไม่มีความสมบูรณ์เพียงพอ

จากปัญหาที่เกิดขึ้นนั้น เราได้ทำการแก้ไขไปพร้อมการทำงานในภาคเรียนที่สองแก้ปัญหาโดยการเพิ่มเวลาในการทำงานให้มากขึ้นกว่าเดิมและซื้ออุปกรณ์ที่หาซื้อได้ยากมาสำรองไว้ และปัญหาที่เกิดขึ้นในภาคเรียนที่สอง ส่วนใหญ่จะอยู่ในส่วนของโปรแกรมที่ใช้ในการรับค่าของข้อมูล เราก็ได้แก้ปัญหาที่เกิดขึ้น เพื่อให้งานบรรลุตามเป้าหมายที่เราได้วางแผนไว้





ภาคผนวก ก

โปรแกรมการคำนวณและแสดงผล

ทบวงศึกษาวัฒนธรรมและกีฬา

กระทรวงศึกษาธิการ

กระทรวงศึกษาธิการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <windows.h>
#include <windowsx.h>
#include <conio.h>
#include "multi2.h"
#include "resource.h"
#include <stdio.h>

//      int test=0,i,j,test1=0,zz=0,k;
//      char str[80],str1[80];
//      long mg = yd[600];

double r = 1.12;
double mg = 0.334;
double ls = 0.02;
int e,Draw,y1[600],y2[600],I1[600],I2[600];
int y3[600],y4[600],y5[600],y6[600],V1[600],V2[600],V3[600];
int y8[600],y12[600],y13[600],Vd[600],gb[9],gb2[8],Iq[600],y9[600],y10[600],Vq[600];
int DataVoltageAB[600][50],DataVoltageBC[600][50], DataCurrentA[600][50],
DataCurrentB[600][50],DataVoltageCA[600][50], DataCurrentC[600][50];
int Vq2[600][50],Vd2[600][50],Iq2[600][50],Id2[600][50];
int Power[600][50],Wmd[600][50],Wmq[600][50];
int P1[600];
int Wmd1[600],Wmq1[600];
int Wd[600],Wq[600];
int y14[600];
int Td[600];
int y15[600];
char DrawVoltage,t;

LRESULT CALLBACK WindowProc(HWND hwnd,UINT msg,WPARAM wparam,LPARAM
lparam)
{
    HDC hdc;
    HDC hdc1;
    HDC hdc2;
    HDC hdc3;
    HDC hdc4;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

HDC hdc5;
HDC hdc7;
HDC hdc8;
HDC hdc9;
HDC hdc10;
HDC hdc11;
HDC hdc15;
HDC hdc16;
HDC hdc17;
HDC hdc18;
HDC hdc19;
HDC hdc20;
HDC hdc21;
HDC hdc22;
HDC hdc23;
HDC hdc24;
HDC hdc25;
HDC hdc26;
HDC hdc27;
hdc=GetDC(hwnd);
switch(msg)
{
case WM_MOUSEMOVE:
{
//***** Control Port *****
_outpw(0x283,0x92);
_outpw(0x300,0x92);
//***** Detect Signal *****
for(int x =1; x <= 600; x++)
{
for(int y =1; y <= 50; y++)
{
_outpw(0x378,0x00);
Sleep(0.001);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

_outpw(0x378,0xFF);

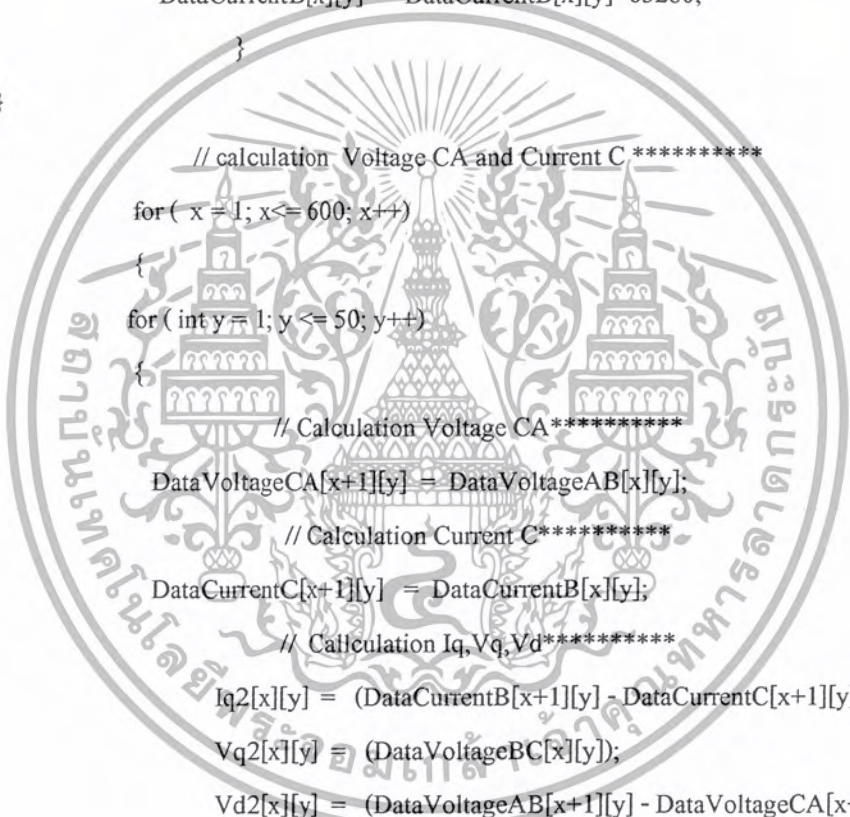
Sleep(0.001);

DataVoltageAB[x][y] = _inpw(0x285);
DataVoltageBC[x][y] = _inpw(0x289);
DataCurrentA[x][y] = _inpw(0x305);
DataCurrentB[x][y] = _inpw(0x309);
DataVoltageAB[x][y] = DataVoltageAB[x][y] -65280 ;
DataVoltageBC[x][y] = DataVoltageBC[x][y] -65280;
DataCurrentA[x][y] = DataCurrentA[x][y] - 65280;
DataCurrentB[x][y] = DataCurrentB[x][y] -65280;
}
}

// calculation Voltage CA and Current C *****
for ( x =1; x<= 600; x++)
{
for ( int y = 1; y <= 50; y++)
{
// Calculation Voltage CA *****
DataVoltageCA[x+1][y] = DataVoltageAB[x][y];
// Calculation Current C *****
DataCurrentC[x+1][y] = DataCurrentB[x][y];
// Calculation Iq,Vq,Vd *****
Iq2[x][y] = (DataCurrentB[x+1][y] - DataCurrentC[x+1][y]);
Vq2[x][y] = (DataVoltageBC[x][y]);
Vd2[x][y] = (DataVoltageAB[x+1][y] - DataVoltageCA[x+1][y]);
Wmd[x][y] = (Vd2[x][y] - (r*DataCurrentA[x][y]));
Wmq[x][y] = (Vq2[x][y] - (r*Iq2[x][y]));
Power[x][y] = DataVoltageAB[x][y]*DataCurrentA[x][y];
}
}

//***** Chang Id,Iq,Vd,Vq from 2D to 1D *****

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//***** Detect Signal *****
for (int i =100; i <= 699; i++)
{
    _outpw(0x378,0x00);
    Sleep(0);
    _outpw(0x378,0xFF);

    Sleep(0);

    int Vab = _inpw(0x285);
    int Vbc = _inpw(0x289);
    int Ia = _inpw(0x305);
    int Ib = _inpw(0x309);

    //***** voltage A *****
    V1[i] = Vab - 65280;
    if (V1[i] == 128)
    { y3[i] = 228; }
    if (V1[i] > 128)
    { int k = V1[i]-128;
      y3[i] = 228+k; }
    if (V1[i] < 128)
    { int k = 128-V1[i];
      y3[i] = 228-k; }

    //***** Voltage B *****
    V2[i] = Vbc - 65280;
    if (V2[i] == 128)
    { y4[i] = 228; }
    if (V2[i] > 128)
    { int k = V2[i]-128;
      y4[i] = 228+k; }
    if (V2[i] < 128)
    { int k = 128-V2[i];
      y4[i] = 228-k; }
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//***** Voltage C *****
//***** Current A *****

I1[i] = Ia - 65280;
if (I1[i] == 128)
    { y1[i] = 228; }
if (I1[i] > 128)
    { int k = I1[i]-128;
y1[i] = 228+k; }
if (I1[i] < 128)
    { int k = 128-I1[i];
y1[i] = 228-k; }

//***** Current B *****
I2[i] = Ib - 65280;
if (I2[i] == 128)
    { y2[i] = 228; }
if (I2[i] > 128)
    { int k = I2[i]-128;
y2[i] = 228+k; }
if (I2[i] < 128)
    { int k = 128-I2[i];
y2[i] = 228-k; }

//***** Calculation Voltage CA and Current C *****
for (int w = 1; w <= 10; w++)
{
for(int k = 699;k >= 102;k--)
    {
int l = y4[k-30];
y5[k] = l;
int w = y2[k-30];
y6[k] = w ;
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//***** Voltage Phase C Axis *****
if (y5[130] > y5[131])
for (int t = 1; t <=30; t++)
{
int s = y5[131-t];
y5[130-t] = (s+4);
}
if (y5[130] < y5[131])
for (int t = 1; t <=30; t++)
{
int s = y5[131-t];
y5[130-t] = (s-4);
}
if (y5[130] == y5[131])
for (int t = 1; t <=30; t++)
{
int s = y5[131-t];
y5[130-t] = (s-4);
}
//***** Current Phase C Axis *****
if (y6[130] > y6[131])
for (int t = 1; t <=30; t++)
{
int z = y6[131-t];
y6[130-t] = (z+4);
}
if (y6[130] < y6[131])
for (int t = 1; t <=30; t++)
{
int z = y6[131-t];
y6[130-t] = (z-4);
}

```

if ((y6[130] == y6[131]) && (y6[135] < y6[136]))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for ( int t = 1 ; t <=30 ; t++)
{
    int z = y6[131-t] ;
    y6[130-t]= (z+4);
}
if ((y6[130] == y6[131]) && ( y6[135] > y6[136]))
for ( int t = 1 ; t <=30 ; t++)
{
    int z = y6[131-t] ;
    y6[130-t]= (z-4);
}
//***** Calculation Vd *****
Vd[i] = (mg)*(V1[i]-V2[i]);
if (Vd[i] == 128)
    { y8[i] = 228; }
if (Vd[i] > 128)
    { int k = I2[i]-128;
      y8[i] = 228+k; }
if (Vd[i] < 128)
    { int k = 128-I2[i];
      y8[i] = 228-k; }
//***** Calculation Vq *****
Vq[i] = (mg)*(y5[i]-y3[i]);
if (Vq[i] == 128)
    { y10[i] = 228; }
if (Vq[i] > 128)
    { int k = I2[i]-128;
      y10[i] = 228+k; }
if (Vq[i] < 128)
    { int k = 128-I2[i];
      y10[i] = 228-k; }

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//***** Callculation Iq *****
Iq[i] = (mg)*(y2[i]-y6[i]);
if(Iq[i] == 128)
    { y9[i] = 228; }
if(Iq[i] > 128)
    { int k = I2[i]-128;
      y9[i] = 228+k; }
if(Iq[i] < 128)
    { int k = 128-I2[i];
      y9[i] = 228-k; }

// ***** End of Calculation *****
}

//***** Integral Signal Part *****
//***** Integral signal for Turqe Calculation *****
//***** Integral signal for Power Calculation *****

for( x = 1; x <= 600; x++)
{
for( int y = 1; y <= 50; y++)
{
    Power[x][y+1] = (Power[x][y+1] + Power[x][y]);
    Wmd[x][y+1] = (Wmd[x][y+1] + Wmd[x][y]);
    Wmq[x][y+1] = (Wmq[x][y+1] + Wmq[x][y]);
}

P1[x+100] = (Power[x][y+1]/50);
Wmd1[x+100] = (Wmd[x][y+1]/50);
Wmq1[x+100] = (Wmq[x][y+1]/50);
P1[x+100] = (P1[x+100])/500;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//***** Callculate Flux Wd,Wq *****
for ( int j = 100 ; j <= 700; j++)
{
    Wd[j] = (Wmd1[j] - (ls*I1[j]));
    Wq[j] = (Wmq1[j] - (ls*Iq[j]));
    Td[j] = ((Wd[j]*Iq[j])-(Wq[j]*I1[j]));
    Td[j] = (Td[j]/800);
    if (Wd[j] == 128)
        { y14[j] = 228; }
    if (Wd[j] > 128)
        { int k = Wd[j]-128;
          y14[j] = 228+k; }
    if (Wd[j] < 128)
        { int k = 128-Wd[j];
          y14[j] = 228-k; }
    if (Wq[j] == 128)
        { y13[j] = 228; }
    if (Wq[j] > 128)
        { int k = Wq[j]-128;
          y13[j] = 228+k; }
    if (Wq[j] < 128)
        { int k = 128-Wq[j];
          y13[j] = 228-k; }
    if (Td[j] == 128)
        { y15[j] = 228; }
    if (Td[j] > 128)
        { int k = Td[j]-128;
          y15[j] = 228+k; }
    if (Td[j] < 128)
        { int k = 128-Td[j];
          y15[j] = 228-k; }
}

```

}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//***** Show Power *****
for( j = 100; j <=700; j++)
    {
    if(P1[j] == 128)
        { y12[j] = 228; }
    if(P1[j] > 128)
        { int k = P1[j]-128;
          y12[i] = 228+k; }
    if(P1[j] < 128)
        { int k = 128-P1[j];
          y12[j] = 228-k; }
        }
    }break;
case WM_COMMAND:
    {
    if(wparam==IDM_V_VoltageA)        e = 1;
    if(wparam==IDM_V_VoltageB)        e = 2;
    if(wparam==IDM_V_VoltageC)        e = 3;
    if(wparam==IDM_V_VoltageABC)      e = 4;
    if(wparam==IDM_C_CurrentA)        e = 5;
    if(wparam==IDM_C_CurrentB)        e = 6;
    if(wparam==IDM_C_CurrentC)        e = 7;
    if(wparam==IDM_C_CurrentABC)      e = 8;
    if(wparam==IDM_Y_Vq)              e = 9;
    if(wparam==IDM_V_Vd)              e = 10;
    if(wparam==IDM_I_Id)              e = 11;
    if(wparam==IDM_I_Iq)              e = 12;
    if(wparam==IDM_P_power)           e = 13;
    if(wparam==IDM_T_Tourqe)          e = 14;
    } break;
case WM_CREATE:
    return(0);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case WM_PAINT :
{
    hdc = GetDC(hwnd);
    HBRUSH brush1 = CreateSolidBrush(RGB(0,200,200));
    SelectObject(hdc,brush1);
    Rectangle(hdc,50,40,760,410);
    // *****Voltage Show*****
    // *****Phase A*****

if (e == 1)
for (int a = 101; a<= 698; a++)
{
    hdc2 = GetDC(hwnd);
    SetBkMode(hdc2,TRANSPARENT);
    SetTextColor(hdc2,RGB(255,0,0));
    hdc1 = GetDC(hwnd);
    HPEN pen2 = CreatePen(PS_SOLID,2,RGB(255,0,0));
    SelectObject(hdc1,pen2);
    TextOut(hdc2,75,92,"380",3);
    TextOut(hdc2,75,348,"380",3);
    TextOut(hdc2,99,92,"-",1);
    TextOut(hdc2,99,348,"-",1);
    TextOut(hdc2,370,50,"VoltageA",8);
    TextOut(hdc2,90,60,"Volt",4);
    TextOut(hdc2,715,220,"Time",4);
    MoveToEx(hdc1,100,100,NULL);
    LineTo(hdc1,100,358);
    MoveToEx(hdc1,100,228,NULL);
    LineTo(hdc1,700,228);
    MoveToEx(hdc1,(a),y3[a],NULL); // y3
    LineTo(hdc1,(a+1),y3[a+1]);

Sleep(1);
DeleteObject(pen2);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ReleaseDC(hwnd,hdc1);
ReleaseDC(hwnd,hdc2);
}

//***** Phase B*****

if (e == 2)
for ( int b = 101;b<=698;b++)
{
hdc3 = GetDC(hwnd);
hdc4 = GetDC(hwnd);
SetBkMode(hdc3,TRANSPARENT);
SetTextColor(hdc3,RGB(0,255,0));
HPEN pen3 = CreatePen(PS_SOLID,2,RGB(0,255,0));
SelectObject(hdc3,pen3);
TextOut(hdc3,75,92,"380",3);
TextOut(hdc3,75,348,"380",3);
TextOut(hdc3,99,92,"-",1);
TextOut(hdc3,99,348,"-",1);
TextOut(hdc3,370,50,"VoltageB",8);
TextOut(hdc3,90,60,"Volt",4);
TextOut(hdc3,715,220,"Time",4);
MoveToEx(hdc4,100,100,NULL);
LineTo(hdc4,100,358);
MoveToEx(hdc4,100,228,NULL);
LineTo(hdc4,700,228);
MoveToEx(hdc4,(b),y4[b],NULL);
LineTo(hdc4,(b+1),y4[b+1]);
Sleep(1);
DeleteObject(pen3);
ReleaseDC(hwnd,hdc3);
ReleaseDC(hwnd,hdc4);
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// ***** Phase C *****

if (e == 3)
    for ( int c = 170; c <= 698; c++)
    {
        hdc5 = GetDC(hwnd);
        HPEN pen9 = CreatePen(PS_SOLID,2,RGB(0,0,255));
        SelectObject(hdc5,pen9);
        SetBkMode(hdc5,TRANSPARENT);
        SetTextColor(hdc5,RGB(0,0,255));
        TextOut(hdc5,75,92,"380",3);
        TextOut(hdc5,75,348,"380",3);
        TextOut(hdc5,99,92,"-",1);
        TextOut(hdc5,99,348,"-",1);
        TextOut(hdc5,370,50,"VoltageC",8);
        TextOut(hdc5,90,60,"Volt",4);
        TextOut(hdc5,715,220,"Time",4);
        MoveToEx(hdc5,100,100,NULL);
        LineTo(hdc5,100,358);
        MoveToEx(hdc5,100,228,NULL);
        LineTo(hdc5,700,228);
        MoveToEx(hdc5,(c),y5[c],NULL);
        LineTo(hdc5,(c+1),y5[c+1]);
        Sleep(1);
        DeleteObject(pen9);
        ReleaseDC(hwnd,hdc5);
    }

// *****Phase ABC*****

```

```

if (e == 4)
    for (int d = 101; d <= 697; d++)
    {
        hdc7 = GetDC(hwnd);
        SetBkMode(hdc7,TRANSPARENT);
        SetTextColor(hdc7,RGB(255,255,255));
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TextOut(hdc7,75,92,"380",3);
TextOut(hdc7,75,348,"380",3);
TextOut(hdc7,99,92,"-",1);
TextOut(hdc7,99,348,"-",1);
TextOut(hdc7,370,50,"VoltageABC",10);
TextOut(hdc7,300,450,"MORNITORING PROJECT",19);
TextOut(hdc7,90,60,"Volt",4);
TextOut(hdc7,715,220,"Time",4);
hdc8 = GetDC(hwnd);
HPEN pen4 = CreatePen(PS_SOLID,2,RGB(255,255,255));
SelectObject(hdc8,pen4);
MoveToEx(hdc8,100,100,NULL);
LineTo(hdc8,100,358);
MoveToEx(hdc8,100,228,NULL);
LineTo(hdc8,700,228);
hdc9 = GetDC(hwnd);
HPEN pen5 = CreatePen(PS_SOLID,2,RGB(255,0,0));
SelectObject(hdc9,pen5);
MoveToEx(hdc9,(d),y3[d],NULL);
LineTo(hdc9,(d+1),y3[d+1]);
Sleep(1);
hdc10 = GetDC(hwnd);
HPEN pen6 = CreatePen(PS_SOLID,2,RGB(0,255,0));
SelectObject(hdc10,pen6);
MoveToEx(hdc10,(d),y4[d],NULL);
LineTo(hdc10,(d+1),y4[d+1]);
Sleep(1);
hdc11 = GetDC(hwnd);
HPEN pen9 = CreatePen(PS_SOLID,2,RGB(0,0,255));
SelectObject(hdc11,pen9);
MoveToEx(hdc11,(d),y5[d],NULL);
LineTo(hdc11,(d+1),y5[d+1]);
Sleep(1);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DeleteObject(pen5);
ReleaseDC(hwnd,hdc11);
DeleteObject(pen6);
ReleaseDC(hwnd,hdc10);
DeleteObject(pen4);
ReleaseDC(hwnd,hdc9);
DeleteObject(pen9);
ReleaseDC(hwnd,hdc8);
ReleaseDC(hwnd,hdc7);
}

// *****Current Show*****
// *****Phase A*****
if((e==5)||(e==11))
for(int g = 101; g<= 698; g++)
{
hdc15 = GetDC(hwnd);
SetBkMode(hdc15,TRANSPARENT);
SetTextColor(hdc15,RGB(255,0,0));
hdc16 = GetDC(hwnd);
HPEN pen10 = CreatePen(PS_SOLID,2,RGB(255,0,0));
SelectObject(hdc16,pen10);
TextOut(hdc15,83,92,"2",1);
TextOut(hdc15,83,348,"2",1);
TextOut(hdc15,99,92,"-",1);
TextOut(hdc15,99,348,"-",1);
TextOut(hdc15,90,60,"Ampere",6);
TextOut(hdc15,715,220,"Time",4);
TextOut(hdc15,300,450,"MORNITORING PROJECT",19);

if ( e == 5)
TextOut(hdc15,370,50,"CurrentA",8);

if ( e ==11)
TextOut(hdc15,370,50,"CurrentId",9);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MoveToEx(hdc16,100,100,NULL);
LineTo(hdc16,100,358);
MoveToEx(hdc16,100,228,NULL);
LineTo(hdc16,700,228);
MoveToEx(hdc16,(g),y1[g],NULL);
LineTo(hdc16,(g+1),y1[g+1]);
Sleep(1);
DeleteObject(pen10);
ReleaseDC(hwnd,hdc16);
ReleaseDC(hwnd,hdc15);
}

// *****Phase B*****
if (e == 6)
for (int h = 101;h<=698;h++)
{
hdc17 = GetDC(hwnd);
hdc18 = GetDC(hwnd);
SetBkMode(hdc17,TRANSPARENT);
SetTextColor(hdc17,RGB(0,255,0));
HPEN pen3 = CreatePen(PS_SOLID,2,RGB(0,255,0));
SelectObject(hdc17,pen3);
TextOut(hdc17,83,92,"2",1);
TextOut(hdc17,83,348,"2",1);
TextOut(hdc17,99,92,"-",1);
TextOut(hdc17,99,348,"-",1);
TextOut(hdc17,370,50,"CurrentB",8);
TextOut(hdc17,300,450,"MORNITORING PROJECT",19);
TextOut(hdc17,90,60,"Ampere",6);
TextOut(hdc17,715,220,"Time",4);
MoveToEx(hdc18,100,100,NULL);
LineTo(hdc18,100,358);
MoveToEx(hdc18,100,228,NULL);
LineTo(hdc18,700,228);
MoveToEx(hdc18,(h),y2[h],NULL);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LineTo(hdc18,(h+1),y2[h+1]);
Sleep(1);
DeleteObject(pen3);
ReleaseDC(hwnd,hdc17);
ReleaseDC(hwnd,hdc18);
}

// *****Phase C *****

if (e == 7)
    for (int a = 101; a<= 698; a++)
    {
        hdc20 = GetDC(hwnd);
        SetBkMode(hdc20,TRANSPARENT);
        SetTextColor(hdc20,RGB(0,0,255));
        hdc19 = GetDC(hwnd);
        HPEN pen2 = CreatePen(PS_SOLID,2,RGB(0,0,255));
        SelectObject(hdc19,pen2);
        TextOut(hdc20,75,92,"380",3);
        TextOut(hdc20,75,348,"380",3);
        TextOut(hdc20,99,92,"-",1);
        TextOut(hdc20,99,348,"-",1);
        TextOut(hdc20,370,50,"CurrentC",8);
        TextOut(hdc20,300,450,"MORNITORING PROJECT",19);
        TextOut(hdc20,90,60,"Amp",3);
        TextOut(hdc20,715,220,"Time",4);
        MoveToEx(hdc19,100,100,NULL);
        LineTo(hdc19,100,358);
        MoveToEx(hdc19,100,228,NULL);
        LineTo(hdc19,700,228);
        MoveToEx(hdc19,(a),y6[a],NULL);
        LineTo(hdc19,(a+1),y6[a+1]);
        Sleep(1);
        DeleteObject(pen2);
        ReleaseDC(hwnd,hdc19);
        ReleaseDC(hwnd,hdc20);
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

// *****Phase ABC*****

if(e == 8)
    for (int d = 101; d<= 697; d++)
    {
        hdc21 = GetDC(hwnd);
        SetBkMode(hdc21,TRANSPARENT);
        SetTextColor(hdc21,RGB(255,255,255));
        TextOut(hdc21,83,92,"2",1);
        TextOut(hdc21,83,348,"2",1);
        TextOut(hdc21,99,92,"-",1);
        TextOut(hdc21,99,348,"-",1);
        TextOut(hdc21,370,50,"CurrentABC",10);
        TextOut(hdc21,300,450,"MORNITORING PROJECT",19);
        TextOut(hdc21,90,60,"Ampare",6);
        TextOut(hdc21,715,220,"Time",4);
        hdc22 = GetDC(hwnd);
        HPEN pen4 = CreatePen(PS_SOLID,2,RGB(255,255,255));
        SelectObject(hdc22,pen4);
        MoveToEx(hdc22,100,100,NULL);
        LineTo(hdc22,100,358);
        MoveToEx(hdc22,100,228,NULL);
        LineTo(hdc22,700,228);
        hdc23 = GetDC(hwnd);
        HPEN pen5 = CreatePen(PS_SOLID,2,RGB(255,0,0));
        SelectObject(hdc23,pen5);
        MoveToEx(hdc23,(d),y1[d],NULL);
        LineTo(hdc23,(d+1),y1[d+1]);
        Sleep(1);
        hdc24 = GetDC(hwnd);
        HPEN pen6 = CreatePen(PS_SOLID,2,RGB(0,255,0));
        SelectObject(hdc24,pen6);
        MoveToEx(hdc24,(d),y2[d],NULL);
        LineTo(hdc24,(d+1),y2[d+1]);
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Sleep(1);
hdc25 = GetDC(hwnd);
HPEN pen9 = CreatePen(PS_SOLID,2,RGB(0,0,255));
SelectObject(hdc25,pen9);
MoveToEx(hdc25,(d),y6[d],NULL);
LineTo(hdc25,(d+1),y6[d+1]);
Sleep(1);
DeleteObject(pen5);
ReleaseDC(hwnd,hdc25);
DeleteObject(pen6);
ReleaseDC(hwnd,hdc24);
DeleteObject(pen4);
ReleaseDC(hwnd,hdc23);
DeleteObject(pen9);
ReleaseDC(hwnd,hdc22);
ReleaseDC(hwnd,hdc21);
}
if (e == 9)
{
for (int a = 101; a <= 697; a++)
{
hdc26 = GetDC(hwnd);
SetBkMode(hdc26,TRANSPARENT);
SetTextColor(hdc26,RGB(255,255,0));
hdc27 = GetDC(hwnd);
HPEN pen2 = CreatePen(PS_SOLID,3,RGB(255,255,0));
SelectObject(hdc27,pen2);
TextOut(hdc26,75,92,"380",3);
TextOut(hdc26,75,348,"380",3);
TextOut(hdc26,99,92,"-",1);
TextOut(hdc26,99,348,"-",1);
TextOut(hdc26,370,50,"VoltageVd",9);
TextOut(hdc26,300,450,"MORNITORING PROJECT",19);
TextOut(hdc26,90,60,"Volt",4);
TextOut(hdc26,715,220,"Time",4);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MoveToEx(hdc27,100,100,NULL);
LineTo(hdc27,100,358);
MoveToEx(hdc27,100,228,NULL);
LineTo(hdc27,700,228);
TextOut(hdc26,(a),y8[a],".",1);
Sleep(1);
DeleteObject(pen2);
ReleaseDC(hwnd,hdc27);
ReleaseDC(hwnd,hdc26);
}
if (e == 10)
    for (int a = 101; a <= 698; a++)
    {
        hdc26 = GetDC(hwnd);
        SetBkMode(hdc26,TRANSPARENT);
        SetTextColor(hdc26,RGB(255,255,255));
        hdc27 = GetDC(hwnd);
        HPEN pen2 = CreatePen(PS_SOLID,3,RGB(0,255,255));
        SelectObject(hdc27,pen2);
        TextOut(hdc26,75,92,"380",3);
        TextOut(hdc26,75,348,"380",3);
        TextOut(hdc26,99,92,"-",1);
        TextOut(hdc26,99,348,"-",1);
        TextOut(hdc26,370,50,"VoltageVq",9);
        TextOut(hdc26,300,450,"MORNITORING PROJECT",19);
        TextOut(hdc26,90,60,"Volt",4);
        TextOut(hdc26,715,220,"Time",4);
        MoveToEx(hdc27,100,100,NULL);
        LineTo(hdc27,100,358);
        MoveToEx(hdc27,100,228,NULL);
        LineTo(hdc27,700,228);
        TextOut(hdc26,(a),y10[a],".",1);
        Sleep(1);
        DeleteObject(pen2);
    }
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        ReleaseDC(hwnd,hdc27);
        ReleaseDC(hwnd,hdc26);
    }
    if (e == 12)
        for (int a = 101; a <= 698; a++)
        {
            hdc2 = GetDC(hwnd);
            SetBkMode(hdc2,TRANSPARENT);
            SetTextColor(hdc2,RGB(255,255,255));
            hdc1 = GetDC(hwnd);
            HPEN pen2 = CreatePen(PS_SOLID,3,RGB(255,0,255));
            SelectObject(hdc1,pen2);
            TextOut(hdc2,82,92,"2",1);
            TextOut(hdc2,82,348,"2",1);
            TextOut(hdc2,99,92,"-",1);
            TextOut(hdc2,99,348,"-",1);
            TextOut(hdc2,370,50,"CurrentIq",9);
            TextOut(hdc1,300,450,"MORNITORING PROJECT",19);
            TextOut(hdc2,90,60,"Amp",3);
            TextOut(hdc2,715,220,"Time",4);
            MoveToEx(hdc1,100,100,NULL);
            LineTo(hdc1,100,358);
            MoveToEx(hdc1,100,228,NULL);
            LineTo(hdc1,700,228);
            MoveToEx(hdc1,(a),y10[a],NULL);
            LineTo(hdc1,(a+1),y10[a+1]);

            Sleep(1);

            DeleteObject(pen2);

            ReleaseDC(hwnd,hdc1);

            ReleaseDC(hwnd,hdc2);
        }
    if (e == 13)
        for (int a = 101; a <= 698; a++)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        hdc2 = GetDC(hwnd);
        SetBkMode(hdc2,TRANSPARENT);
        SetTextColor(hdc2,RGB(255,255,255));

        hdc1 = GetDC(hwnd);
        HPEN pen2 = CreatePen(PS_SOLID,2,RGB(0,255,255));
        SelectObject(hdc1,pen2);
        TextOut(hdc2,70,92,"750",3);
        TextOut(hdc2,82,348,"0",1);
        TextOut(hdc2,99,92,"-",1);
        TextOut(hdc2,99,348,"-",1);
        TextOut(hdc2,370,50,"PowerMax",8);
        TextOut(hdc2,300,450,"MORNITORING PROJECT",19);
        TextOut(hdc2,90,60,"Watt",4);
        TextOut(hdc2,715,350,"Time",4);
        MoveToEx(hdc1,100,100,NULL);
        LineTo(hdc1,100,358);
        MoveToEx(hdc1,100,358,NULL);
        LineTo(hdc1,700,358);
        MoveToEx(hdc1,(a),y12[a],NULL);
        LineTo(hdc1,(a+1),y12[a+1]);
        Sleep(1);
        DeleteObject(pen2);
        ReleaseDC(hwnd,hdc1);
        ReleaseDC(hwnd,hdc2);
    }
    if (e == 14)
        for (int a = 101; a <= 697; a++)
    {
        hdc2 = GetDC(hwnd);
        SetBkMode(hdc2,TRANSPARENT);
        SetTextColor(hdc2,RGB(255,255,255));
        hdc1 = GetDC(hwnd);
        HPEN pen2 = CreatePen(PS_SOLID,2,RGB(255,255,0));
        SelectObject(hdc1,pen2);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

hdc3 = GetDC(hwnd);
HPEN pen3 = Createpen(PS_SOLID,2,RGB(255,255,255));
SelectObject(hdc3,pen3);
TextOut(hdc2,82,92,"5",1);
TextOut(hdc2,82,348,"0",1);
TextOut(hdc2,99,92,"-",1);
TextOut(hdc2,99,348,"-",1);
TextOut(hdc2,370,50,"Torque",6);
TextOut(hdc2,300,450,"MORNITORING PROJECT",19);
TextOut(hdc2,90,60,"N-m",3);
TextOut(hdc2,715,358,"Time",4);
MoveToEx(hdc3,100,95,NULL);
LineTo(hdc3,100,358);
MoveToEx(hdc3,100,358,NULL);
LineTo(hdc3,700,358);
MoveToEx(hdc1,(a),y15[a],NULL);
LineTo(hdc1,(a+1),y15[a+1]);
Sleep(1);
DeleteObject(pen2);
ReleaseDC(hwnd,hdc1);
ReleaseDC(hwnd,hdc2);
}

DeleteObject(brush1);
ReleaseDC(hwnd,hdc);

}break;

case WM_DESTROY :
{
PostQuitMessage(0);
}break;
}

return(DefWindowProc(hwnd,msg,wparam,lparam));

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int WINAPI WinMain(HINSTANCE hinstance,HINSTANCE hPrevInstance,LPSTR
                    lpCmdline,int nShowCmd)
{
    WNDCLASSEX winclass;
    winclass.cbSize = sizeof(WNDCLASSEX);
    winclass.style = CS_VREDRAW|CS_HREDRAW|CS_OWNDC|CS_DBLCLKS;
    winclass.lpfnWndProc = WindowProc;
    winclass.cbClsExtra = 0;
    winclass.cbWndExtra = 0;
    winclass.hInstance = hinstance;
    winclass.hbrBackground = (HBRUSH)GetStockObject(LTGRAY_BRUSH);
    winclass.lpszClassName = "MYWIN";
    winclass.lpszMenuName = MAKEINTRESOURCE(IDR_MENU1);
    winclass.hIcon = LoadIcon(hinstance,IDI_APPLICATION);
    winclass.hCursor = LoadCursor(hinstance, IDC_ARROW);
    winclass.hIconSm = LoadIcon(hinstance,IDI_APPLICATION);
    RegisterClassEx(&winclass);
    HWND hwnd;
    MSG msg;
    Hwn=CreateWindowEx(NULL,"MYWIN","MORNITORING
PROJECT",WS_OVERLAPPEDWINDOW,100,100,
400,400,NULL,NULL,hinstance,NULL);
    ShowWindow(hwnd,nShowCmd);
    UpdateWindow(hwnd);
    while (1)
    {
        if (PeekMessage(&msg,NULL,0,0,PM_NOREMOVE))
        {
            if (!GetMessage(&msg,NULL,0,0))
                return msg.wParam;
            TranslateMessage(&msg);
            DispatchMessage(&msg);
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}  
}  
return (msg.wParam);  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข

ข้อมูลเกี่ยวกับอุปกรณ์ที่ใช้ในโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADC0820

8-Bit High Speed μ P Compatible A/D Converter with Track/Hold Function

General Description

By using a half-flash conversion technique, the 8-bit ADC0820 CMOS A/D offers a 1.5 μ s conversion time and dissipates only 75 mW of power. The half-flash technique consists of 32 comparators, a most significant 4-bit ADC and a least significant 4-bit ADC.

The input to the ADC0820 is tracked and held by the input sampling circuitry eliminating the need for an external sample-and-hold for signals moving at less than 100 mV/ μ s.

For ease of interface to microprocessors, the ADC0820 has been designed to appear as a memory location or I/O port without the need for external interfacing logic.

Key Specifications

- Resolution
- Conversion Time
- Low Power
- Total Unadjusted Error

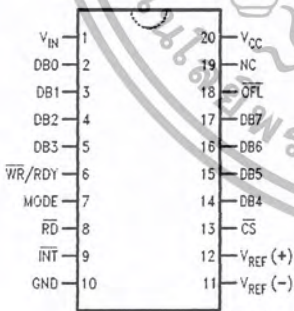
8 Bits
 2.5 μ s Max (RD Mode)
 1.5 μ s Max (WR-RD Mode)
 75 mW Max
 $\pm 1/2$ LSB and ± 1 LSB

Features

- Built-in track-and-hold function
- No missing codes
- No external clocking
- Single supply — 5 V_{CC}
- Easy interface to all microprocessors, or operates stand-alone
- Latched STR1-STATE output
- Logic inputs and outputs meet both MOS and T²L voltage level specifications
- Operates ratiometrically or with any reference value equal to or less than V_{CC}
- 0V to 5V analog input voltage range with single 5V supply
- No zero or full-scale adjust required
- Overflow output available for cascading
- 0.3" standard width 20-pin DIP
- 20-pin molded chip carrier package
- 20-pin small outline package
- 20-pin shrink small outline package (SSOP)

Connection and Functional Diagrams

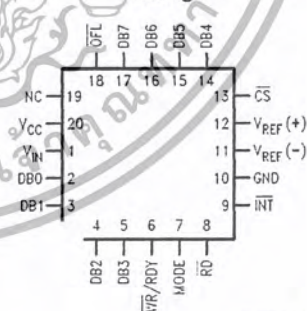
Dual-In-Line, Small Outline and SSOP Packages



Top View

DS005501-1

Molded Chip Carrier Package



DS005501-33

Connection and Functional Diagrams (Continued)

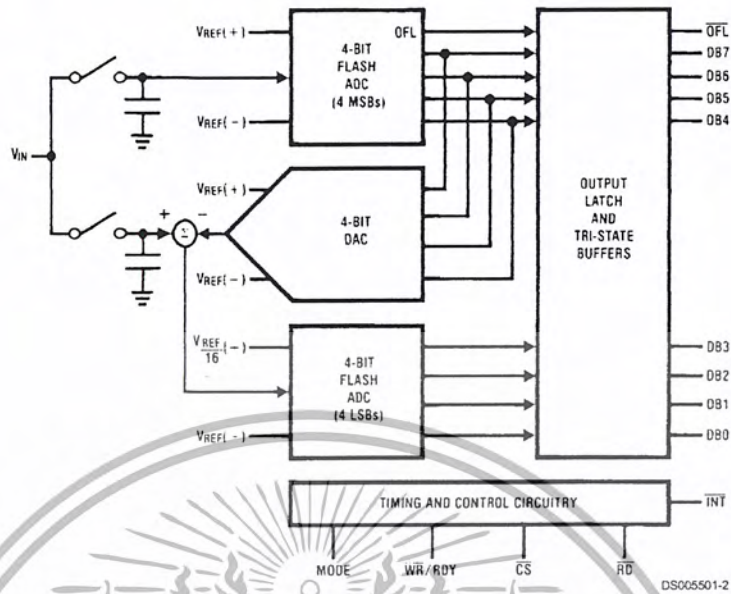


FIGURE 1.

Ordering Information

| Part Number | Total Unadjusted Error | Package | Temperature Range |
|-------------|------------------------|--------------------------------|-------------------|
| ADC0820BCV | ±½ LSB | V20A — Molded Chip Carrier | 0°C to +70°C |
| ADC0820BCWM | | M20B — Wide Body Small Outline | 0°C to +70°C |
| ADC0820BCN | | N20A — Molded DIP | 0°C to +70°C |
| ADC0820CCJ | ±1 LSB | J20A — Cerdip | -40°C to +85°C |
| ADC0820CCWM | | M20B — Wide Body Small Outline | 0°C to +70°C |
| ADC0820CIWM | | M20B — Wide Body Small Outline | -40°C to +85°C |
| ADC0820CCN | | N20A — Molded DIP | 0°C to +70°C |

Absolute Maximum Ratings (Notes 1, 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

| | |
|---|-------------------------|
| Supply Voltage (V_{CC}) | 10V |
| Logic Control Inputs | -0.2V to V_{CC} +0.2V |
| Voltage at Other Inputs and Output | -0.2V to V_{CC} +0.2V |
| Storage Temperature Range | -65°C to +150°C |
| Package Dissipation at $T_A = 25^\circ\text{C}$ | 875 mW |
| Input Current at Any Pin (Note 5) | 1 mA |
| Package Input Current (Note 5) | 4 mA |
| ESD Susceptibility (Note 9) | 1200V |
| Lead Temp. (Soldering, 10 sec.) | |
| Dual-In-Line Package (plastic) | 260°C |

| | |
|--------------------------------|-------|
| Dual-In-Line Package (ceramic) | 300°C |
| Surface Mount Package | |
| Vapor Phase (60 sec.) | 215°C |
| Infrared (15 sec.) | 220°C |

Operating Ratings (Notes 1, 2)

| | |
|--------------------------|---------------------------------|
| Temperature Range | $T_{MIN} \leq T_A \leq T_{MAX}$ |
| ADC0820CCJ | -40°C $\leq T_A \leq$ +85°C |
| ADC0820CIWM | -40°C $\leq T_A \leq$ +85°C |
| ADC0820BCN, ADC0820CCN | 0°C $\leq T_A \leq$ 70°C |
| ADC0820BCV | 0°C $\leq T_A \leq$ 70°C |
| ADC0820BCWM, ADC0820CCWM | 0°C $\leq T_A \leq$ 70°C |
| V_{CC} Range | 4.5V to 8V |

Converter Characteristics

The following specifications apply for RD mode (pin 7=0), $V_{CC}=5V$, $V_{REF(+)}=5V$, and $V_{REF(-)}=GND$ unless otherwise specified. **Boldface limits apply from T_{MIN} to T_{MAX} ; all other limits $T_A=T_I=25^\circ\text{C}$.**

| Parameter | Conditions | ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM | | | | | | Limit Units |
|--------------------------------------|---|---|-----------------------------|-----------------------------|---------------------------------|---------------------------------|--------------------------------|-------------|
| | | ADC0820CCJ | | | ADC0820CIWM | | | |
| | | Typ (Note 6) | Tested Limit (Note 7) | Design Limit (Note 8) | Typ (Note 6) | Tested Limit (Note 7) | Design Limit (Note 8) | |
| Resolution | | | 8 | | 8 | 8 | Bits | |
| Total Unadjusted Error (Note 3) | ADC0820BCN, BCWM ADC0820CCJ ADC0820CCN, CCWM, CIWM, ADC0820CCMSA | | ± 1 | | $\pm 1/2$ ± 1 ± 1 | $\pm 1/2$ ± 1 ± 1 | LSB LSB LSB | |
| Minimum Reference Resistance | | 2.3 | 1.00 | | 2.3 | 1.2 | k Ω | |
| Maximum Reference Resistance | | 2.3 | 6 | | 2.3 | 5.3 | k Ω | |
| Maximum $V_{REF(+)}$ Input Voltage | | | V_{CC} | | V_{CC} | V_{CC} | V | |
| Minimum $V_{REF(-)}$ Input Voltage | | | GND | | GND | GND | V | |
| Minimum $V_{REF(+)}$ Input Voltage | | | $V_{REF(-)}$ | | $V_{REF(-)}$ | $V_{REF(-)}$ | V | |
| Maximum $V_{REF(-)}$ Input Voltage | | | $V_{REF(+)}$ | | $V_{REF(+)}$ | $V_{REF(+)}$ | V | |
| Maximum V_{IN} Input Voltage | | | $V_{CC}+0.1$ | | $V_{CC}+0.1$ | $V_{CC}+0.1$ | V | |
| Minimum V_{IN} Input Voltage | | | GND-0.1 | | GND-0.1 | GND-0.1 | V | |
| Maximum Analog Input Leakage Current | $\overline{CS}=V_{CC}$ $V_{IN}=V_{CC}$ $V_{IN}=GND$ | | 3 -3 | | 0.3 -0.3 | 3 -3 | μA μA | |
| Power Supply Sensitivity | $V_{CC}=5V \pm 5\%$ | | $\pm 1/16$ $\pm 1/4$ | | $\pm 1/16$ $\pm 1/4$ | $\pm 1/4$ $\pm 1/4$ | LSB | |

DC Electrical Characteristics

The following specifications apply for $V_{CC}=5V$, unless otherwise specified. **Boldface** limits apply from T_{MIN} to T_{MAX} ; all other limits $T_A=T_J=25^\circ C$.

| Parameter | Conditions | | ADC0820CCJ | | | ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM | | | Limit Units |
|--|---|---|-----------------|-----------------------------|-----------------------------|---|-----------------------------|-----------------------------|-------------|
| | | | Typ (Note 6) | Tested Limit (Note 7) | Design Limit (Note 8) | Typ (Note 6) | Tested Limit (Note 7) | Design Limit (Note 8) | |
| $V_{IN(1)}$, Logical "1" Input Voltage | $V_{CC}=5.25V$ | \overline{CS} , \overline{WR} , \overline{RD} | | 2.0 | | | 2.0 | 2.0 | V |
| | | Mode | | 3.5 | | | 3.5 | 3.5 | V |
| $V_{IN(0)}$, Logical "0" Input Voltage | $V_{CC}=4.75V$ | \overline{CS} , \overline{WR} , \overline{RD} | | 0.8 | | | 0.8 | 0.8 | V |
| | | Mode | | 1.5 | | | 1.5 | 1.5 | V |
| $I_{IN(1)}$, Logical "1" Input Current | $V_{IN(1)}=5V$; \overline{CS} , \overline{RD} | | 0.005 | 1 | | 0.005 | | 1 | μA |
| | | $V_{IN(1)}=5V$; \overline{WR} | | 0.1 | 3 | | 0.1 | 3 | μA |
| | | $V_{IN(1)}=5V$; Mode | | 50 | 200 | | 50 | 200 | μA |
| $I_{IN(0)}$, Logical "0" Input Current | $V_{IN(0)}=0V$; \overline{CS} , \overline{RD} , \overline{WR} , Mode | | -0.005 | -1 | | -0.005 | | -1 | μA |
| $V_{OUT(1)}$, Logical "1" Output Voltage | $V_{CC}=4.75V$, $I_{OUT}=-360 \mu A$; DB0-DB7, \overline{OFL} , \overline{INT} | | | 2.4 | | | 2.8 | 2.4 | V |
| | $V_{CC}=4.75V$, $I_{OUT}=-10 \mu A$; DB0-DB7, \overline{OFL} , \overline{INT} | | | 4.5 | | | 4.6 | 4.5 | V |
| $V_{OUT(0)}$, Logical "0" Output Voltage | $V_{CC}=4.75V$, $I_{OUT}=1.6 mA$; DB0-DB7, \overline{OFL} , \overline{INT} , RDY | | | 0.4 | | | 0.34 | 0.4 | V |
| I_{OUT} , TRI-STATE Output Current | $V_{OUT}=5V$; DB0-DB7, RDY | | 0.1 | 3 | | 0.1 | 0.3 | 3 | μA |
| | $V_{OUT}=0V$; DB0-DB7, RDY | | -0.1 | -3 | | -0.1 | -0.3 | -3 | μA |
| I_{SOURCE} , Output Source Current | $V_{OUT}=0V$; DB0-DB7, \overline{OFL} , \overline{INT} | | -12 | -6 | | -12 | -7.2 | -6 | mA |
| | $V_{OUT}=5V$; DB0-DB7, \overline{OFL} , \overline{INT} , RDY | | -9 | -4.0 | | -9 | -5.3 | -4.0 | mA |
| I_{SINK} , Output Sink Current | $V_{OUT}=5V$; DB0-DB7, \overline{OFL} , \overline{INT} , RDY | | 14 | 7 | | 14 | 8.4 | 7 | mA |
| I_{CC} , Supply Current | $\overline{CS}=\overline{WR}=\overline{RD}=0$ | | 7.5 | 15 | | 7.5 | 13 | 15 | mA |

AC Electrical Characteristics

The following specifications apply for $V_{CC}=5V$, $t_r=t_f=20 ns$, $V_{REF(+)}=5V$, $V_{REF(-)}=0V$ and $T_A=25^\circ C$ unless otherwise specified.

| Parameter | Conditions | Typ (Note 6) | Tested Limit (Note 7) | Design Limit (Note 8) | Units |
|---|--|-----------------|-----------------------------|-----------------------------|---------|
| t_{CRD} , Conversion Time for RD Mode | Pin 7 = 0, Figure 2 | 1.6 | | 2.5 | μs |
| t_{ACC0} , Access Time (Delay from Falling Edge of \overline{RD} to Output Valid) | Pin 7 = 0, Figure 2 | $t_{CRD}+20$ | | $t_{CRD}+50$ | ns |
| t_{CWR-RD} , Conversion Time for WR-RD Mode | Pin 7 = V_{CC} ; $t_{WR} = 600 ns$, $t_{RD}=600 ns$; Figures 3, 4 | | | 1.52 | μs |
| t_{WR} , Write Time | Min | | 600 | | ns |
| | Max | | 50 | | μs |
| t_{RD} , Read Time | Min | | 600 | | ns |
| t_{ACC1} , Access Time (Delay from Falling Edge of \overline{RD} to Output Valid) | Pin 7 = V_{CC} , $t_{RD} < t_r$; Figure 3 $C_L=15 pF$ | 190 | | 280 | ns |
| | $C_L=100 pF$ | 210 | | 320 | ns |

AC Electrical Characteristics (Continued)

The following specifications apply for $V_{CC}=5V$, $t_r=t_f=20$ ns, $V_{REF(+)}=5V$, $V_{REF(-)}=0V$ and $T_A=25^\circ C$ unless otherwise specified.

| Parameter | Conditions | Typ (Note 6) | Tested Limit (Note 7) | Design Limit (Note 8) | Units |
|---|---|-----------------|-----------------------------|-----------------------------|------------|
| t_{ACC2} , Access Time (Delay from Falling Edge of \overline{RD} to Output Valid) | Pin 7 = V_{CC} , $t_{RD} > t_i$; Figure 4 $C_L = 15$ pF | 70 | | 120 | ns |
| | $C_L = 100$ pF | 90 | | 150 | ns |
| t_{ACC3} , Access Time (Delay from Rising Edge of RDY to Output Valid) | $R_{PULLUP} = 1k$ and $C_L = 15$ pF | 30 | | | ns |
| t_i , Internal Comparison Time | Pin 7 = V_{CC} ; Figures 4, 5 $C_L = 50$ pF | 800 | | 1300 | ns |
| t_{1H} , t_{0H} , TRI-STATE Control (Delay from Rising Edge of \overline{RD} to Hi-Z State) | $R_L = 1k$, $C_L = 10$ pF | 100 | | 200 | ns |
| t_{INTL} , Delay from Rising Edge of \overline{WR} to Falling Edge of \overline{INT} | Pin 7 = V_{CC} , $C_L = 50$ pF $t_{RD} > t_i$; Figure 4 | | | t_i | ns |
| | $t_{RD} < t_i$; Figure 3 | $t_{RD} + 200$ | | $t_{RD} + 290$ | ns |
| t_{INTH} , Delay from Rising Edge of \overline{RD} to Rising Edge of \overline{INT} | Figures 2, 3, 4 $C_L = 50$ pF | 125 | | 225 | ns |
| t_{INTHWR} , Delay from Rising Edge of \overline{WR} to Rising Edge of \overline{INT} | Figure 5, $C_L = 50$ pF | 175 | | 270 | ns |
| t_{RDY} , Delay from \overline{CS} to RDY | Figure 2, $C_L = 50$ pF, Pin 7 = 0 | 50 | | 100 | ns |
| t_{ID} , Delay from \overline{INT} to Output Valid | Figure 5 | 20 | | 50 | ns |
| t_{RI} , Delay from \overline{RD} to \overline{INT} | Pin 7 = V_{CC} , $t_{RD} < t_i$ Figure 3 | 200 | | 290 | ns |
| t_p , Delay from End of Conversion to Next Conversion | Figures 2, 3, 4, 5 (Note 4) See Graph | | | 500 | ns |
| Slew Rate, Tracking | | 0.1 | | | V/ μ s |
| C_{VIN} , Analog Input Capacitance | | 45 | | | pF |
| C_{OUT} , Logic Output Capacitance | | 5 | | | pF |
| C_{IN} , Logic Input Capacitance | | 5 | | | pF |

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to the GND pin, unless otherwise specified.

Note 3: Total unadjusted error includes offset, full-scale, and linearity errors.

Note 4: Accuracy may degrade if t_{WR} or t_{RD} is shorter than the minimum value specified. See Accuracy vs t_{WR} and Accuracy vs t_{RD} graphs.

Note 5: When the input voltage (V_{IN}) at any pin exceeds the power supply rails ($V_{IN} < V^-$ or $V_{IN} > V^+$), the absolute value of current at that pin should be limited to 1 mA or less. The 4 mA package input current limits the number of pins that can exceed the power supply boundaries with a 1 mA current limit to four.

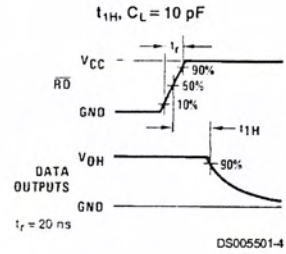
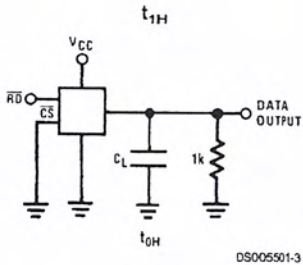
Note 6: Typicals are at $25^\circ C$ and represent most likely parametric norm.

Note 7: Tested limits are guaranteed to National's AOQL (Average Outgoing Quality Level).

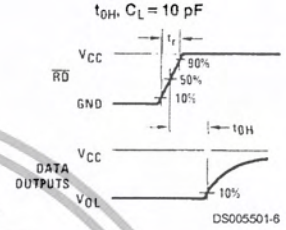
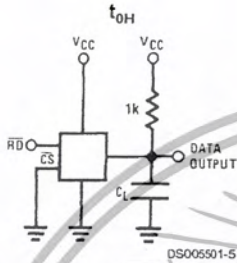
Note 8: Design limits are guaranteed but not 100% tested. These limits are not used to calculate outgoing quality levels.

Note 9: Human body model, 100 pF discharged through a 1.5 k Ω resistor.

TRI-STATE Test Circuits and Waveforms

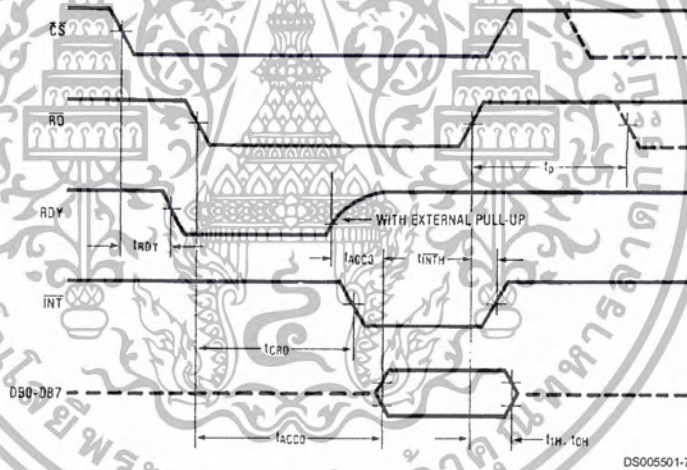


$t_r = 20 \text{ ns}$



$t_r = 20 \text{ ns}$

Timing Diagrams



Note: On power-up the state of $\overline{\text{INT}}$ can be high or low.

FIGURE 2. RD Mode (Pin 7 is Low)

Timing Diagrams (Continued)

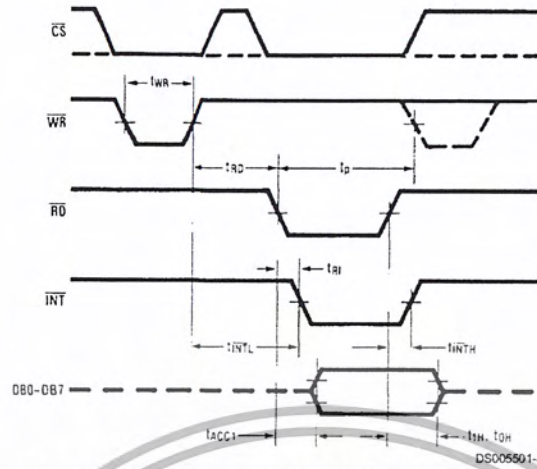


FIGURE 3. WR-RD Mode (Pin 7 is High and $t_{RD} < t_i$)

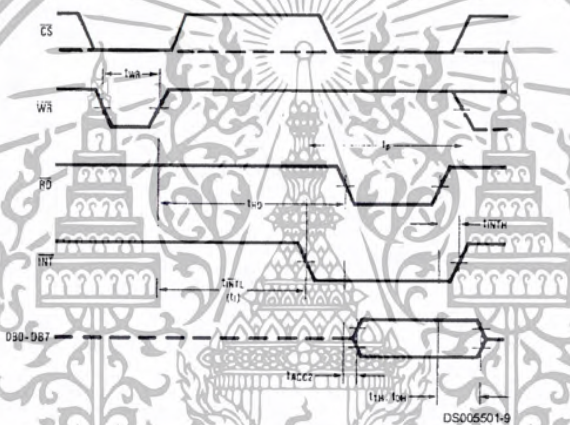


FIGURE 4. WR-RD Mode (Pin 7 is High and $t_{RD} > t_i$)

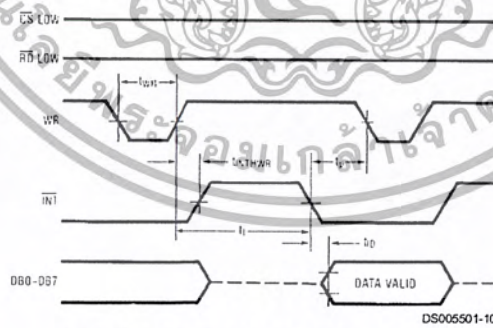
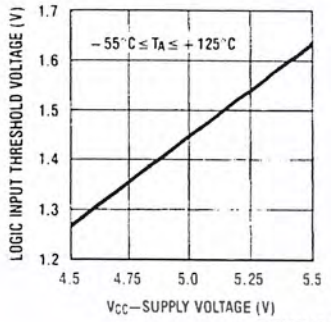


FIGURE 5. WR-RD Mode (Pin 7 is High)
Stand-Alone Operation

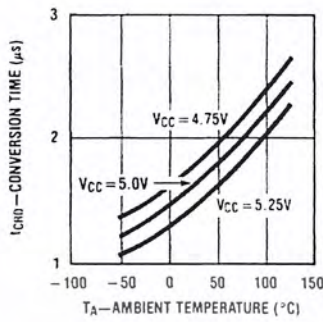
Typical Performance Characteristics

Logic Input Threshold Voltage vs Supply Voltage



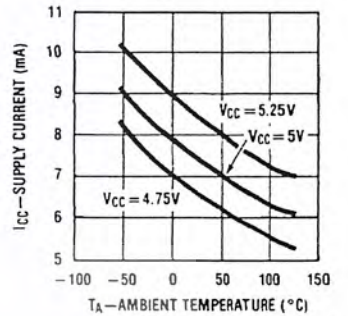
DS005501-34

Conversion Time (RD Mode) vs Temperature



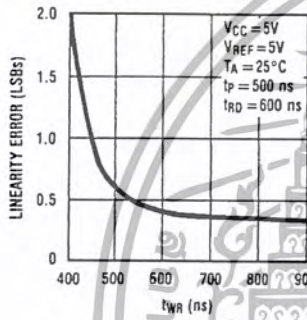
DS005501-35

Power Supply Current vs Temperature (not including reference ladder)



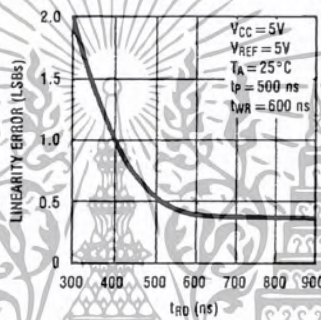
DS005501-36

Accuracy vs t_{WR}



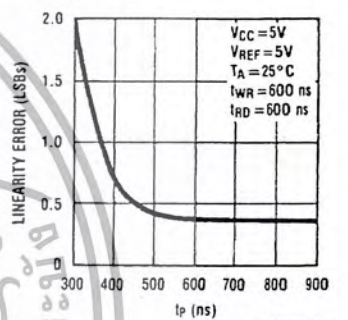
DS005501-37

Accuracy vs t_{RD}



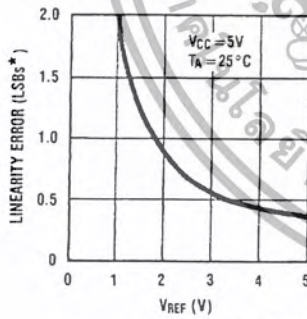
DS005501-38

Accuracy vs t_p



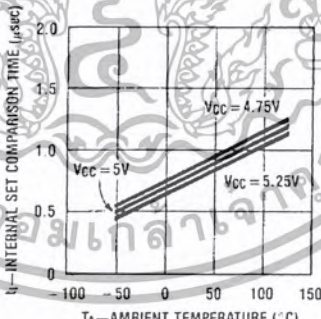
DS005501-39

Accuracy vs V_{REF}
[V_{REF} = V_{REF}(+) - V_{REF}(-)]



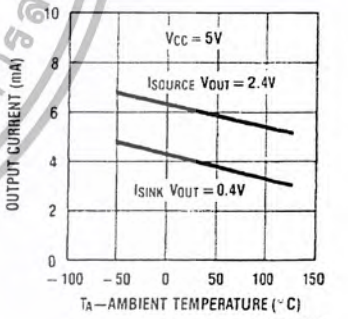
DS005501-40

t_i Internal Time Delay vs Temperature



DS005501-41

Output Current vs Temperature



DS005501-42

*1 LSB = $\frac{V_{REF}}{256}$

Description of Pin Functions

| Pin | Name | Function |
|-----|-------------------------|--|
| 1 | V _{IN} | Analog input; range = GND ≤ V _{IN} ≤ V _{CC} |
| 2 | DB0 | TRI-STATE data output—bit 0 (LSB) |
| 3 | DB1 | TRI-STATE data output—bit 1 |
| 4 | DB2 | TRI-STATE data output—bit 2 |
| 5 | DB3 | TRI-STATE data output—bit 3 |
| 6 | \overline{WR} /RDY | <p>WR-RD Mode</p> <p>\overline{WR}: With \overline{CS} low, the conversion is started on the falling edge of \overline{WR}. Approximately 800 ns (the preset internal time out, t_i) after the \overline{WR} rising edge, the result of the conversion will be strobed into the output latch, provided that \overline{RD} does not occur prior to this time out (see Figures 3, 4).</p> <p>RD Mode</p> <p>RDY: This is an open drain output (no internal pull-up device). RDY will go low after the falling edge of \overline{CS}; RDY will go TRI-STATE when the result of the conversion is strobed into the output latch. It is used to simplify the interface to a microprocessor system (see Figure 2).</p> |
| 7 | Mode | <p>Mode: Mode selection input—it is internally tied to GND through a 50 μA current source.</p> <p>RD Mode: When mode is low</p> <p>WR-RD Mode: When mode is high</p> |
| 8 | \overline{RD} | <p>WR-RD Mode</p> <p>With \overline{CS} low, the TRI-STATE data outputs (DB0-DB7) will be activated when \overline{RD} goes low (see Figure 5). \overline{RD} can also be used to increase the speed of the converter by reading data prior to the preset internal time out (t_i, ~800 ns). If this is done, the data result transferred to output latch is latched after the falling edge of the \overline{RD} (see Figures 3, 4).</p> <p>RD Mode</p> <p>With \overline{CS} low, the conversion will start with \overline{RD} going low, also \overline{RD} will enable the TRI-STATE data outputs at the completion of the conversion. RDY going TRI-STATE and \overline{INT} going low indicates the completion of the conversion (see Figure 2).</p> |
| 9 | \overline{INT} | <p>WR-RD Mode</p> <p>\overline{INT} going low indicates that the conversion is completed and the data result is in the output latch. \overline{INT} will go low, ~800 ns (the preset internal time out, t_i) after the rising edge of \overline{WR} (see Figure 4); or \overline{INT} will go low after the falling edge of \overline{RD}, if \overline{RD} goes low prior to the 800 ns time out (see Figure 3). \overline{INT} is reset by the rising edge of \overline{RD} or \overline{CS} (see Figures 3, 4).</p> <p>RD Mode</p> <p>\overline{INT} going low indicates that the conversion is completed and the data result is in the output latch. \overline{INT} is reset by the rising edge of \overline{RD} or \overline{CS} (see Figure 2).</p> |
| 10 | GND | Ground |
| 11 | V _{REF(-)} | The bottom of resistor ladder, voltage range: GND ≤ V _{REF(-)} ≤ V _{REF(+)} (Note 5) |
| 12 | V _{REF(+)} | The top of resistor ladder, voltage range: V _{REF(-)} ≤ V _{REF(+)} ≤ V _{CC} (Note 5) |
| 13 | \overline{CS} | \overline{CS} must be low in order for the \overline{RD} or \overline{WR} to be recognized by the converter. |
| 14 | DB4 | TRI-STATE data output—bit 4 |
| 15 | DB5 | TRI-STATE data output—bit 5 |
| 16 | DB6 | TRI-STATE data output—bit 6 |
| 17 | DB7 | TRI-STATE data output—bit 7 (MSB) |
| 18 | \overline{OFL} | Overflow output—If the analog input is higher than the V _{REF(+)} , \overline{OFL} will be low at the end of conversion. It can be used to cascade 2 or more devices to have more resolution (9, 10-bit). This output is always active and does not go into TRI-STATE as DB0-DB7 do. |
| 19 | NC | No connection |
| 20 | V _{CC} | Power supply voltage |

1.0 Functional Description

1.1 GENERAL OPERATION

The ADC0820 uses two 4-bit flash A/D converters to make an 8-bit measurement (Figure 1). Each flash ADC is made up of 15 comparators which compare the unknown input to a reference ladder to get a 4-bit result. To take a full 8-bit reading, one flash conversion is done to provide the 4 most significant data bits (via the MS flash ADC). Driven by the 4

MSBs, an internal DAC recreates an analog approximation of the input voltage. This analog signal is then subtracted from the input, and the difference voltage is converted by a second 4-bit flash ADC (the LS ADC), providing the 4 least significant bits of the output data word.

The internal DAC is actually a subsection of the MS flash converter. This is accomplished by using the same resistor

1.0 Functional Description (Continued)

ladder for the A/D as well as for generating the DAC signal. The DAC output is actually the tap on the resistor ladder which most closely approximates the analog input. In addition, the "sampled-data" comparators used in the ADC0820 provide the ability to compare the magnitudes of several analog signals simultaneously, without using input summing amplifiers. This is especially useful in the LS flash ADC, where the signal to be converted is an analog difference.

1.2 THE SAMPLED-DATA COMPARATOR

Each comparator in the ADC0820 consists of a CMOS inverter with a capacitively coupled input (Figures 6, 7). Analog switches connect the two comparator inputs to the input capacitor (C) and also connect the inverter's input and output. This device in effect now has one differential input pair. A comparison requires two cycles, one for zeroing the comparator, and another for making the comparison.

In the first cycle, one input switch and the inverter's feedback switch (Figure 6) are closed. In this interval, C is charged to the connected input (V1) less the inverter's bias voltage (V_B, approximately 1.2V). In the second cycle (Figure 7), these two switches are opened and the other (V2) input's switch is closed. The input capacitor now subtracts its stored voltage from the second input and the difference is amplified by the inverter's open loop gain. The inverter's input (V_{B'}) becomes

$$V_B - (V1 - V2) \frac{C}{C + C_S}$$

and the output will go high or low depending on the sign of V_{B'} - V_B.

The actual circuitry used in the ADC0820 is a simple but important expansion of the basic comparator described above. By adding a second capacitor and another set of switches to the input (Figure 8), the scheme can be expanded to make dual differential comparisons. In this circuit, the feedback switch and one input switch on each capacitor (Z switches) are closed in the zeroing cycle. A comparison is

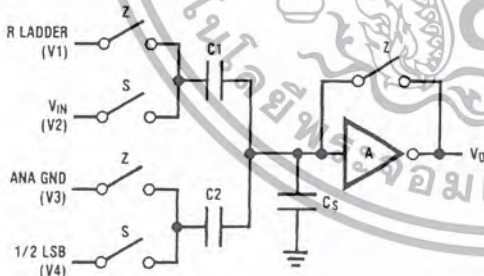


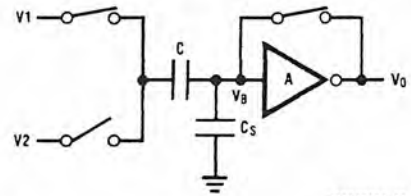
FIGURE 8. ADC0820 Comparator (from MS Flash ADC)

1.3 ARCHITECTURE

In the ADC0820, one bank of 15 comparators is used in each 4-bit flash A/D converter (Figure 12). The MS (most significant) flash ADC also has one additional comparator to detect input overrange. These two sets of comparators operate alternately, with one group in its zeroing cycle while the other is comparing.

When a typical conversion is started, the \overline{WR} line is brought low. At this instant the MS comparators go from zeroing to comparison mode (Figure 11). When \overline{WR} is returned high

then made by connecting the second input on each capacitor and opening all of the other switches (S switches). The change in voltage at the inverter's input, as a result of the change in charge on each input capacitor, will now depend on both input signal differences.

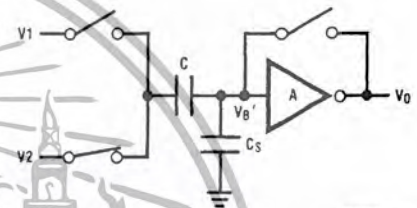


DS006501-12

- V_O = V_B
- V on C = V1 - V_B
- C_S = stray input node capacitor
- V_B = inverter input bias voltage

Zeroing Phase

FIGURE 6. Sampled-Data Comparator



DS006501-13

- V_{B'} - V_B = (V2 - V1) $\frac{C}{C + C_S}$
- V_O' = $\frac{-A}{C + C_S} [CV2 - CV1]$
- V_O' is dependent on V2 - V1

Compare Phase

FIGURE 7. Sampled-Data Comparator

$$V_O = \frac{-A}{C1 + C2 + C_S} [C1(V2 - V1) + C2(V4 - V3)]$$

$$= \frac{-A}{C1 + C2 + C_S} [\Delta Q_{C1} + \Delta Q_{C2}]$$

DS006501-14

after at least 600 ns, the output from the first set of comparators (the first flash) is decoded and latched. At this point the two 4-bit converters change modes and the LS (least significant) flash ADC enters its compare cycle. No less than 600 ns later, the \overline{RD} line may be pulled low to latch the lower 4 data bits and finish the 8-bit conversion. When \overline{RD} goes low, the flash A/Ds change state once again in preparation for the next conversion.

Figure 11 also outlines how the converter's interface timing relates to its analog input (V_{IN}). In \overline{WR} - \overline{RD} mode, V_{IN} is

1.0 Functional Description (Continued)

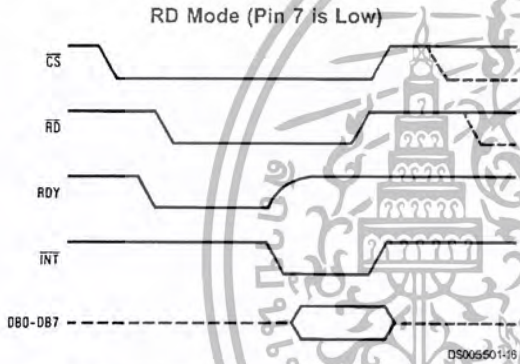
measured while \overline{WR} is low. In RD mode, sampling occurs during the first 800 ns of \overline{RD} . Because of the input connections to the ADC0820's LS and MS comparators, the converter has the ability to sample V_{IN} at one instant (Section 2.4), despite the fact that two separate 4-bit conversions are being done. More specifically, when \overline{WR} is low the MS flash is in compare mode (connected to V_{IN}), and the LS flash is in zero mode (also connected to V_{IN}). Therefore both flash ADCs sample V_{IN} at the same time.

1.4 DIGITAL INTERFACE

The ADC0820 has two basic interface modes which are selected by strapping the MODE pin high or low.

RD Mode

With the MODE pin grounded, the converter is set to Read mode. In this configuration, a complete conversion is done by pulling \overline{RD} low until output data appears. An \overline{INT} line is provided which goes low at the end of the conversion as well as a RDY output which can be used to signal a processor that the converter is busy or can also serve as a system Transfer Acknowledge signal.



When in RD mode, the comparator phases are internally triggered. At the falling edge of \overline{RD} , the MS flash converter goes from zero to compare mode and the LS ADC's comparators enter their zero cycle. After 800 ns, data from the MS flash is latched and the LS flash ADC enters compare mode. Following another 800 ns, the lower 4 bits are recovered.

WR then RD Mode

With the MODE pin tied high, the A/D will be set up for the WR-RD mode. Here, a conversion is started with the \overline{WR} input; however, there are two options for reading the output data which relate to interface timing. If an interrupt driven scheme is desired, the user can wait for \overline{INT} to go low before reading the conversion result (Figure 10). \overline{INT} will typically

go low 800 ns after \overline{WR} 's rising edge. However, if a shorter conversion time is desired, the processor need not wait for \overline{INT} and can exercise a read after only 600 ns (Figure 9). If this is done, \overline{INT} will immediately go low and data will appear at the outputs.

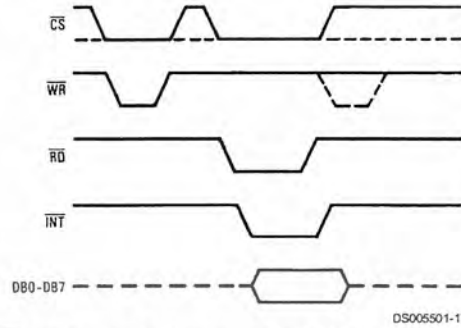


FIGURE 9. WR-RD Mode (Pin 7 is High and $t_{RD} < t_i$)

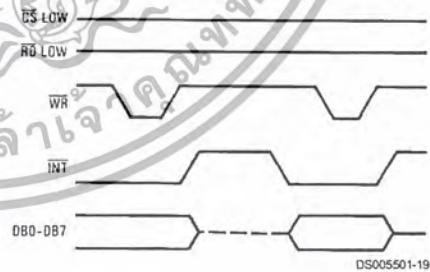


FIGURE 10. WR-RD Mode (Pin 7 is High and $t_{RD} > t_i$)

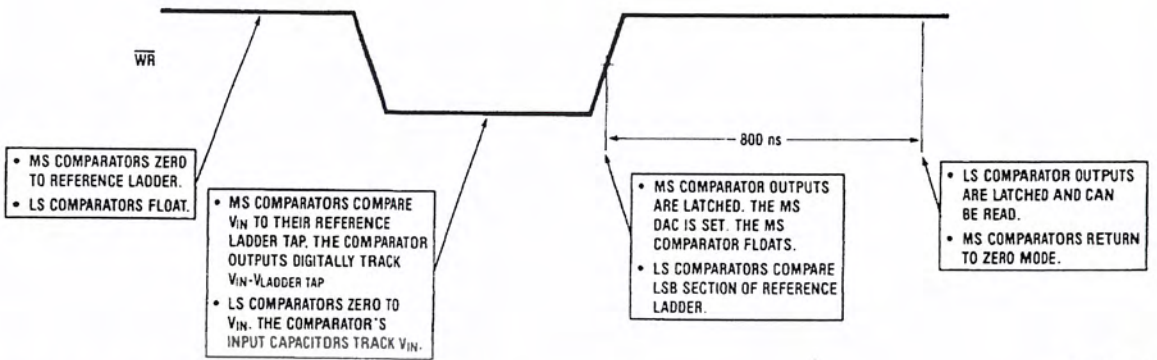
Stand-Alone

For stand-alone operation in WR-RD mode, \overline{CS} and \overline{RD} can be tied low and a conversion can be started with \overline{WR} . Data will be valid approximately 800 ns following \overline{WR} 's rising edge.

WR-RD Mode (Pin 7 is High) Stand-Alone Operation



1.0 Functional Description (Continued)



Note: MS means most significant
LS means least significant

DS005501-20

FIGURE 11. Operating Sequence (WR-RD Mode)

OTHER INTERFACE CONSIDERATIONS

In order to maintain conversion accuracy, \overline{WR} has a maximum width spec of 50 μ s. When the MS flash ADC's sampled-data comparators (Section 1.2) are in comparison mode (\overline{WR} is low), the input capacitors (C, Figure 8) must hold their charge. Switch leakage and inverter bias current can cause errors if the comparator is left in this phase for too long.

Since the MS flash ADC enters its zeroing phase at the end of a conversion (Section 1.3), a new conversion cannot be started until this phase is complete. The minimum spec for this time (t_P , Figures 2, 3, 4, 5) is 500 ns.



Detailed Block Diagram

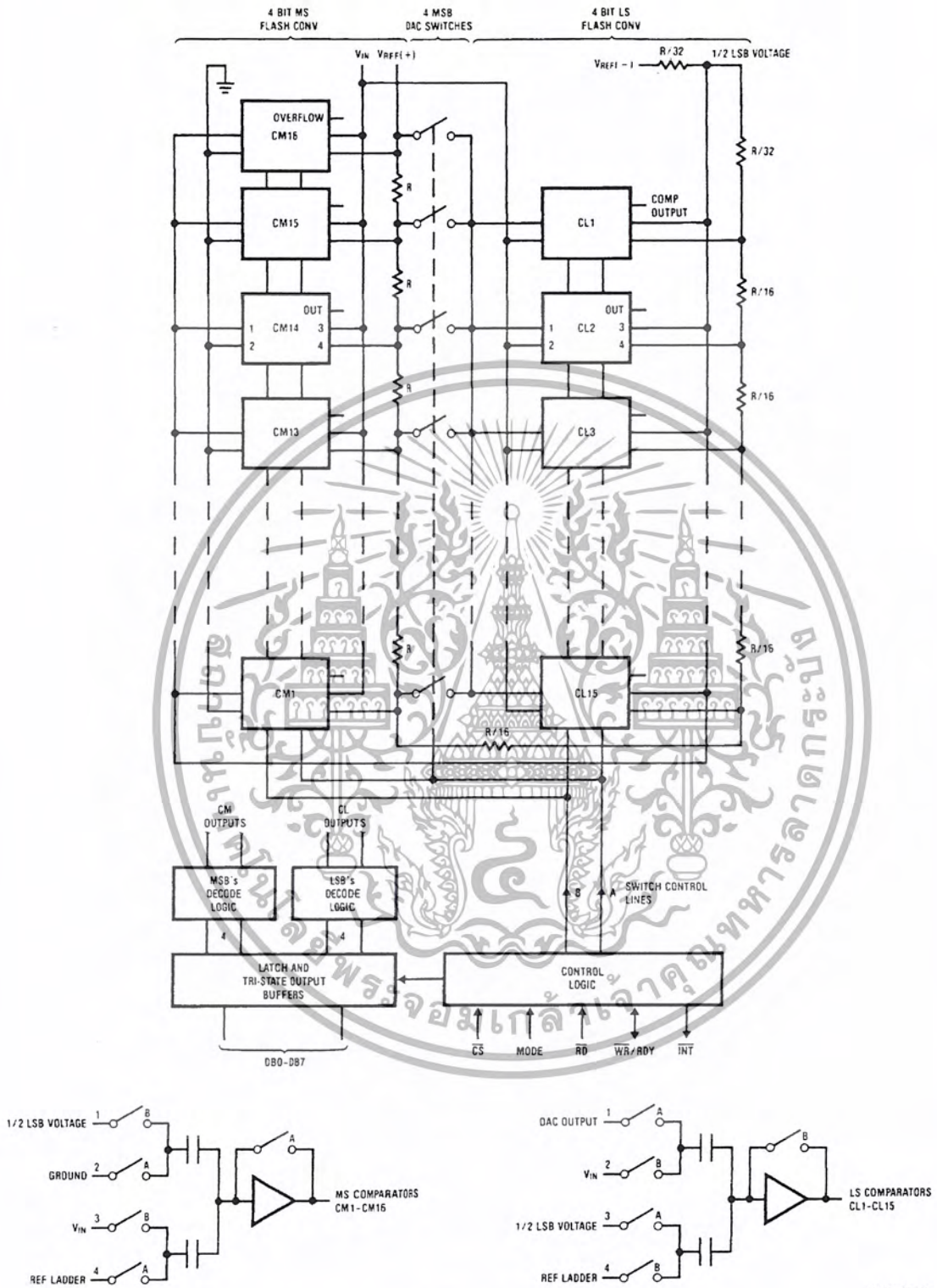


FIGURE 12.

DS005501-15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 Analog Considerations

2.1 REFERENCE AND INPUT

The two V_{REF} inputs of the ADC0820 are fully differential and define the zero to full-scale input range of the A to D converter. This allows the designer to easily vary the span of the analog input since this range will be equivalent to the voltage difference between $V_{IN}(+)$ and $V_{IN}(-)$. By reducing V_{REF} ($V_{REF}=V_{REF}(+)-V_{REF}(-)$) to less than 5V, the sensitivity of the converter can be increased (i.e., if $V_{REF}=2V$ then 1 LSB=7.8 mV). The input/reference arrangement also facilitates ratiometric operation and in many cases the chip power supply can be used for transducer power as well as the V_{REF} source.

This reference flexibility lets the input span not only be varied but also offset from zero. The voltage at $V_{REF}(-)$ sets the input level which produces a digital output of all zeroes. Though V_{IN} is not itself differential, the reference design affords nearly differential-input capability for most measurement applications. Figure 13 shows some of the configurations that are possible.

2.2 INPUT CURRENT

Due to the unique conversion techniques employed by the ADC0820, the analog input behaves somewhat differently than in conventional devices. The A/D's sampled-data comparators take varying amounts of input current depending on which cycle the conversion is in.

The equivalent input circuit of the ADC0820 is shown in Figure 14. When a conversion starts (\overline{WR} low, WR-RD mode), all input switches close, connecting V_{IN} to thirty-one 1 pF capacitors. Although the two 4-bit flash circuits are not both in their compare cycle at the same time, V_{IN} still sees all input capacitors at once. This is because the MS flash converter is connected to the input during its compare interval and the LS flash is connected to the input during its zeroing phase (Section 1.3). In other words, the LS ADC uses V_{IN} as its zero-phase input.

The input capacitors must charge to the input voltage through the on resistance of the analog switches (about 5 k Ω to 10 k Ω). In addition, about 12 pF of input stray capacitance must also be charged. For large source resistances, the analog input can be modeled as an RC network as shown in Figure 15. As R_S increases, it will take longer for the input capacitance to charge.

In RD mode, the input switches are closed for approximately 800 ns at the start of the conversion. In WR-RD mode, the time that the switches are closed to allow this charging is the time that \overline{WR} is low. Since other factors force this time to be at least 800 ns, input time constants of 100 ns can be accommodated without special consideration. Typical total input capacitance values of 45 pF allow R_S to be 1.5 k Ω without lengthening \overline{WR} to give V_{IN} more time to settle.

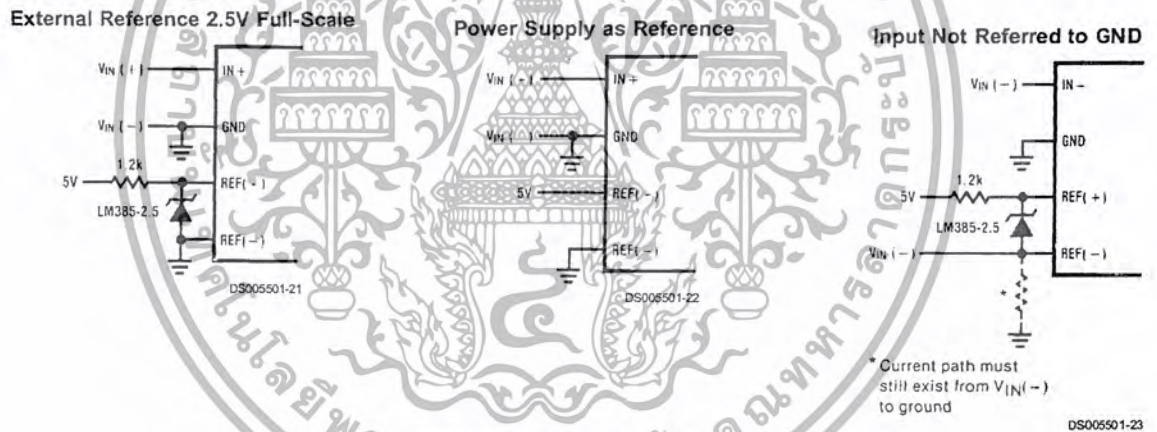


FIGURE 13. Analog Input Options

2.0 Analog Considerations (Continued)

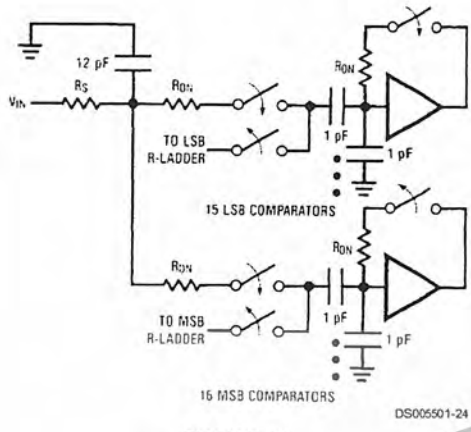


FIGURE 14.

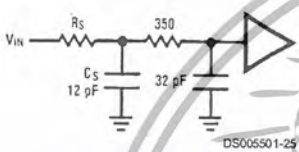


FIGURE 15.

2.3 INPUT FILTERING

It should be made clear that transients in the analog input signal, caused by charging current flowing into V_{IN} , will not degrade the A/D's performance in most cases. In effect the ADC0820 does not "look" at the input when these transients occur. The comparators' outputs are not latched while \overline{WR} is low, so at least 600 ns will be provided to charge the ADC's input capacitance. It is therefore not necessary to filter out these transients by putting an external cap on the V_{IN} terminal.

2.4 INHERENT SAMPLE-HOLD

Another benefit of the ADC0820's input mechanism is its ability to measure a variety of high speed signals without the help of an external sample-and-hold. In a conventional SAR type converter, regardless of its speed, the input must remain at least $\frac{1}{2}$ LSB stable throughout the conversion process if full accuracy is to be maintained. Consequently, for many high speed signals, this signal must be externally sampled, and held stationary during the conversion.

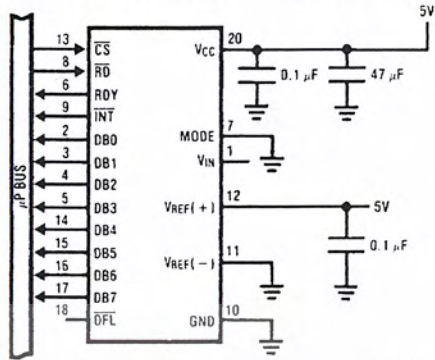
Sampled-data comparators, by nature of their input switching, already accomplish this function to a large degree (Section 1.2). Although the conversion time for the ADC0820 is 1.5 μ s, the time through which V_{IN} must be $\frac{1}{2}$ LSB stable is much smaller. Since the MS flash ADC uses V_{IN} as its "compare" input and the LS ADC uses V_{IN} as its "zero" input, the ADC0820 only "samples" V_{IN} when \overline{WR} is low (Sections 1.3 and 2.2). Even though the two flashes are not done simultaneously, the analog signal is measured at one instant. The value of V_{IN} approximately 100 ns after the rising edge of \overline{WR} (100 ns due to internal logic prop delay) will be the measured value.

Input signals with slew rates typically below 100 mV/ μ s can be converted without error. However, because of the input time constants, and charge injection through the opened comparator input switches, faster signals may cause errors. Still, the ADC0820's loss in accuracy for a given increase in signal slope is far less than what would be witnessed in a conventional successive approximation device. An SAR type converter with a conversion time as fast as 1 μ s would still not be able to measure a 5V 1 kHz sine wave without the aid of an external sample-and-hold. The ADC0820, with no such help, can typically measure 5V, 7 kHz waveforms.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

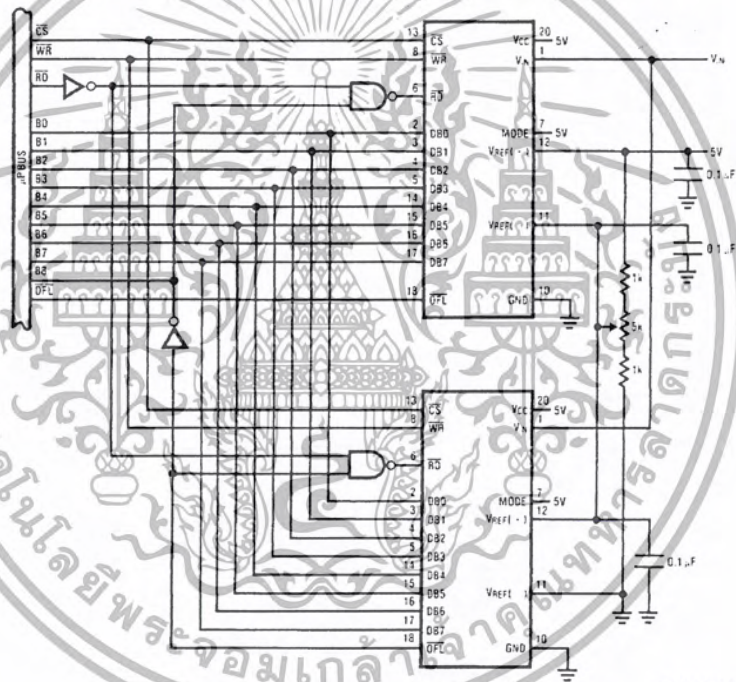
3.0 Typical Applications

8-Bit Resolution Configuration



DS005501-26

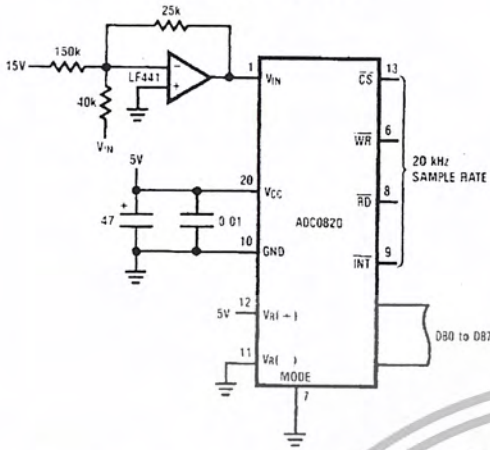
9-Bit Resolution Configuration



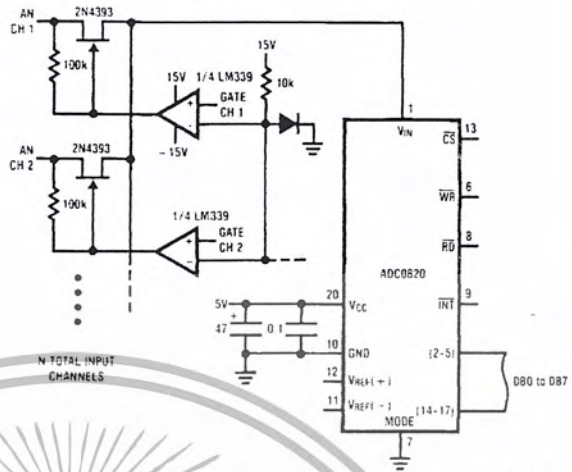
DS005501-27

3.0 Typical Applications (Continued)

Telecom A/D Converter

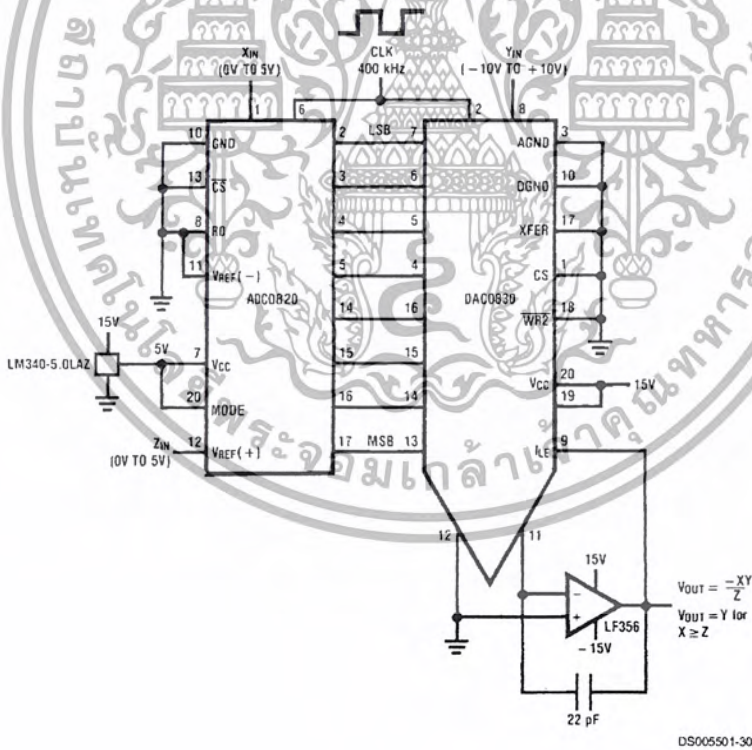


Multiple Input Channels



- $V_{IN} = 3 \text{ kHz max } \pm 4V_P$
- No track-and-hold needed
- Low power consumption

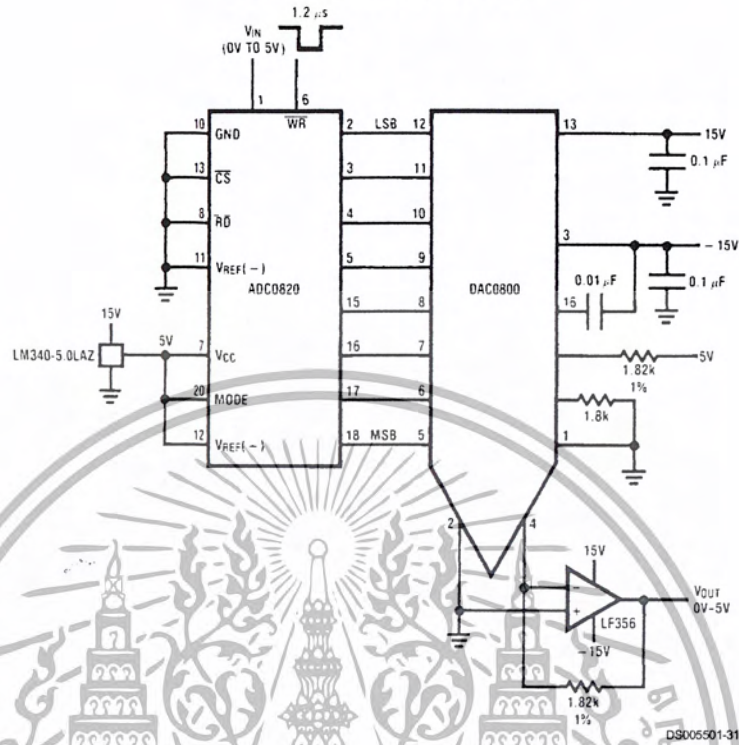
8-Bit 2-Quadrant Analog Multiplier



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

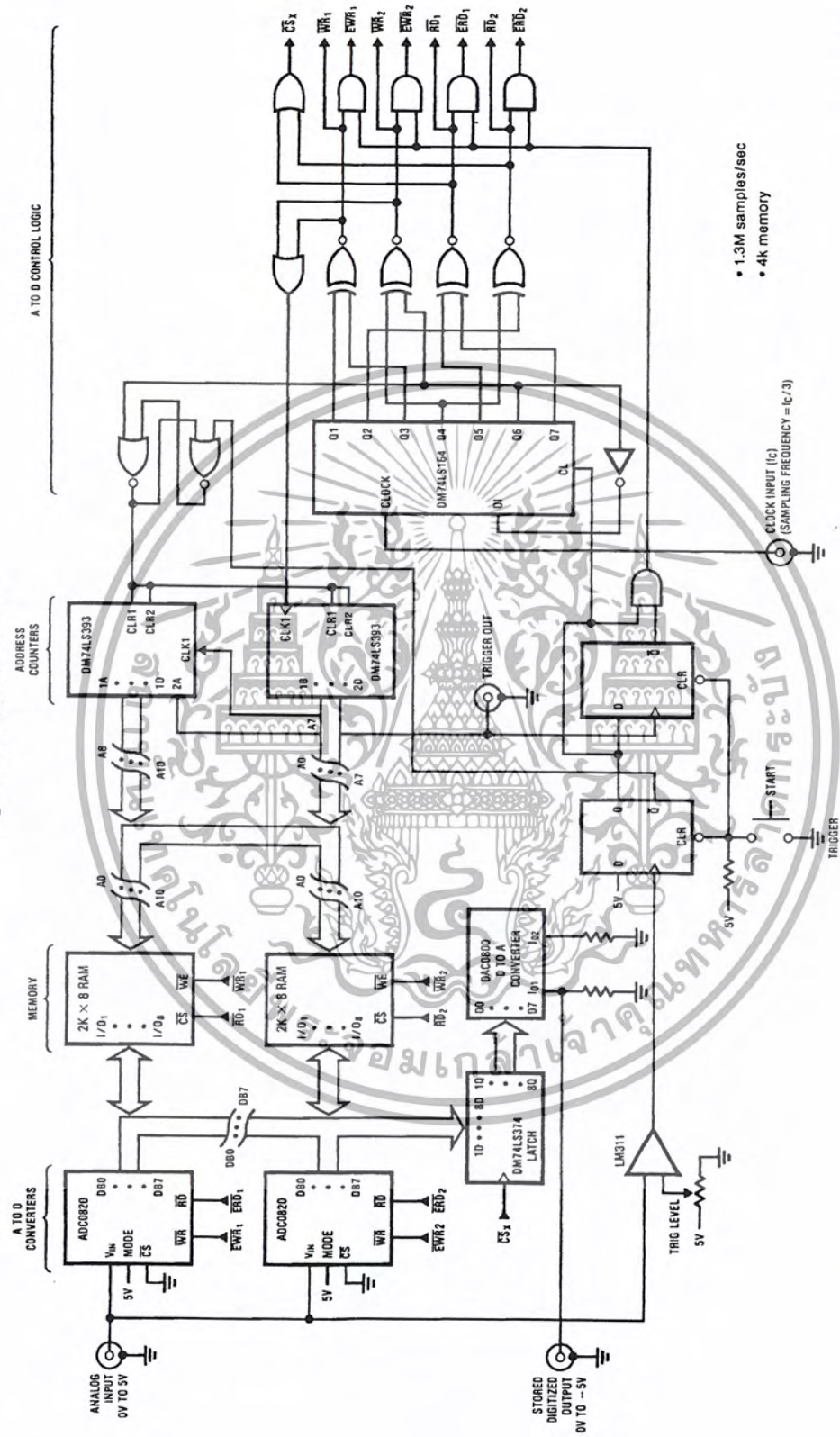
3.0 Typical Applications (Continued)

Fast Infinite Sample-and-Hold



3.0 Typical Applications (Continued)

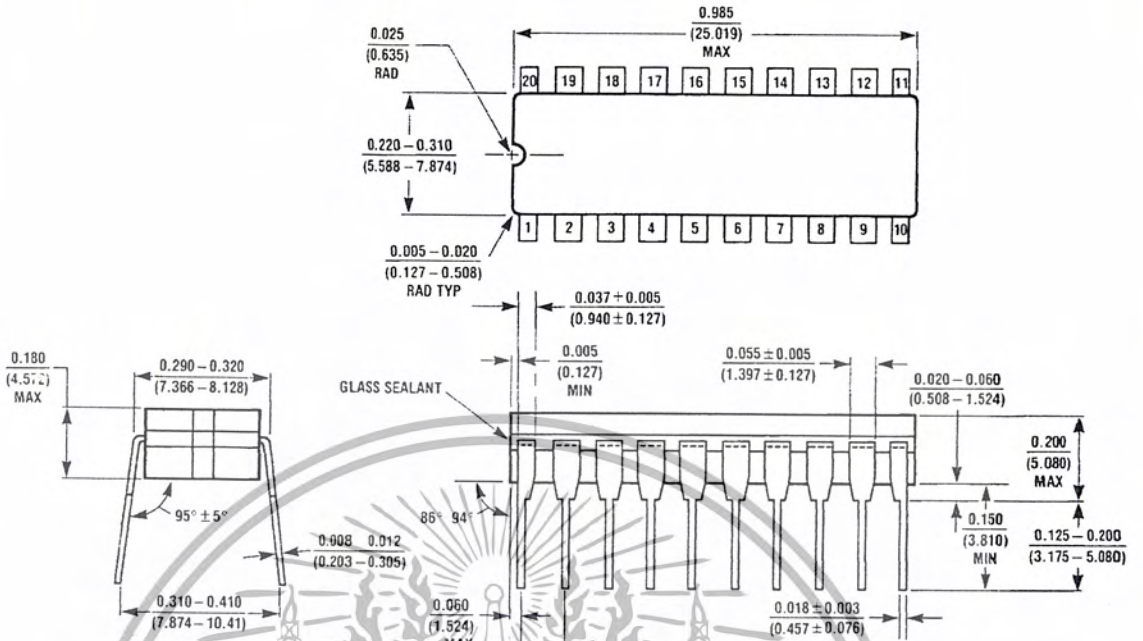
Digital Waveform Recorder



DS90C0501-32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted

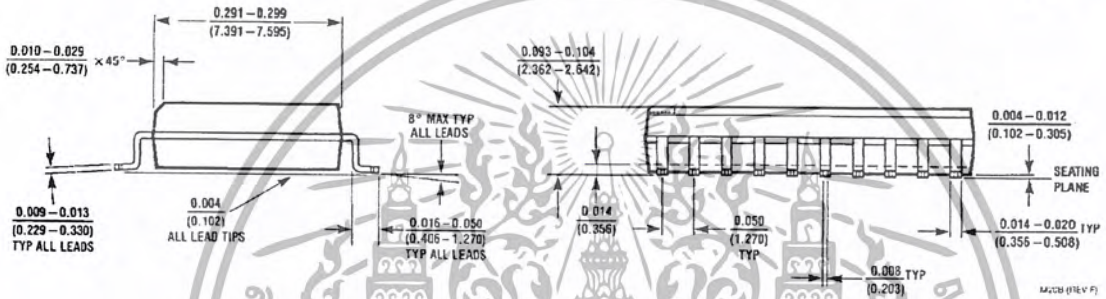
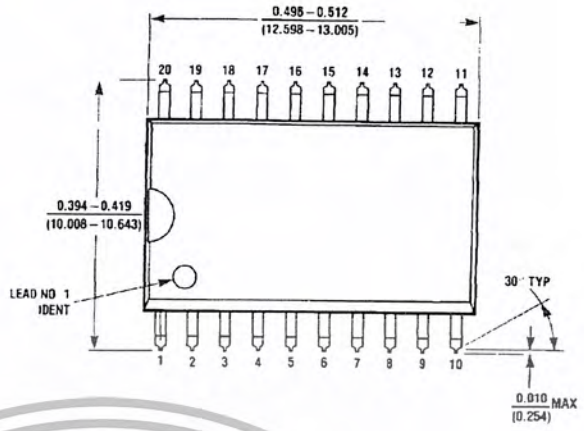


Hermetic Dual-In-Line Package (J)
 Order Number ADC0820CCJ
 NS Package Number J20A

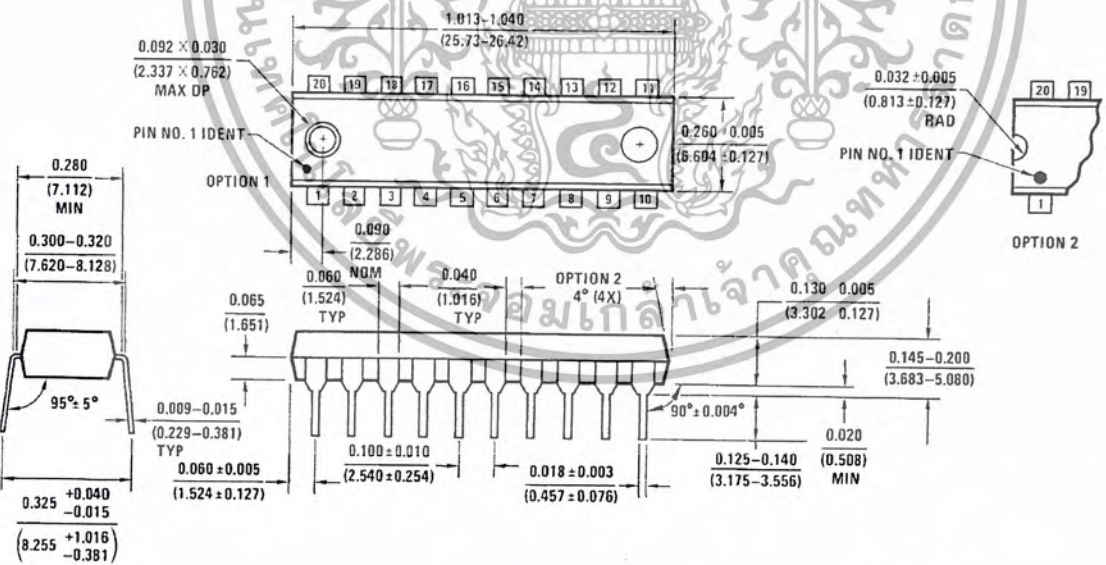
J20A (REV M)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



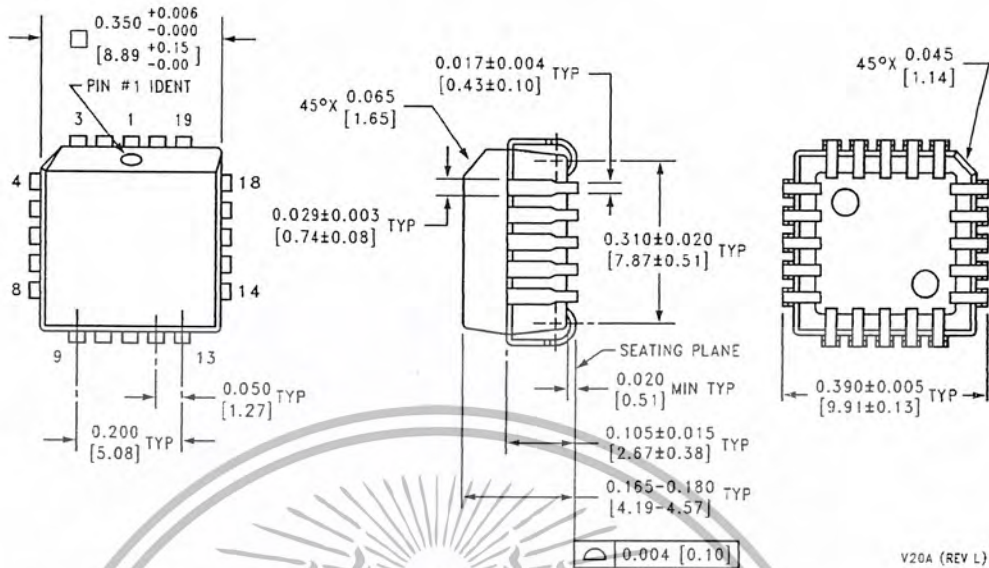
SO Package (M)
 Order Number ADC0820BCWM, ADC0820CCWM or ADC0820CIWM
 NS Package Number M20B



Molded Dual-In-Line Package (N)
 Order Number ADC0820BCN or ADC0820CCN
 NS Package Number N20A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)




Molded Chip Carrier Package (V)
 Order Number ADC0820BCV
 NS Package Number V20A

V20A (REV L)

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

 National Semiconductor Corporation
 Americas
 Email: support@nsc.com

National Semiconductor Europe
 Fax: +49 (0) 180-530 85 86
 Email: europe.support@nsc.com
 Deutsch Tel: +49 (0) 69 9508 6208
 English Tel: +44 (0) 870 24 0 2171
 Français Tel: +33 (0) 1 41 91 8790

National Semiconductor Asia Pacific Customer Response Group
 Tel: 65-2544466
 Fax: 65-2504466
 Email: ap.support@nsc.com

National Semiconductor Japan Ltd.
 Tel: 81-3-5639-7560
 Fax: 81-3-5639-7507

www.national.com

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Current Transducer LA 55-P/SP32

$I_{PN} = 50 \text{ A}$

For the electronic measurement of currents : DC, AC, pulsed..., with a galvanic isolation between the primary circuit (high power) and the secondary circuit (electronic circuit).



Electrical data

| | | | | | | | |
|-------------------------|---|------------------------------------|-----------------------------|--------------------------|-------------------|------------------|------------------|
| I_{PN} | Primary nominal r.m.s. current | 50 | A | | | | |
| I_P | Primary current, measuring range | 0 .. ± 70 | A | | | | |
| R_M | Measuring resistance @ | $T_A = 70^\circ\text{C}$ | | $T_A = 85^\circ\text{C}$ | | | |
| | | $R_{M \min}$ | $R_{M \max}$ | $R_{M \min}$ | $R_{M \max}$ | | |
| | | with $\pm 12 \text{ V}$ | @ $\pm 50 \text{ A}_{\max}$ | 10 | 100 | 60 | 95 |
| | | | @ $\pm 70 \text{ A}_{\max}$ | 10 | 50 | 60 ¹⁾ | 60 ¹⁾ |
| with $\pm 15 \text{ V}$ | @ $\pm 50 \text{ A}_{\max}$ | 50 | 160 | 135 | 155 | | |
| | @ $\pm 70 \text{ A}_{\max}$ | 50 | 90 | 135 ²⁾ | 135 ²⁾ | | |
| I_{SN} | Secondary nominal r.m.s. current | 50 | mA | | | | |
| K_N | Conversion ratio | 1 : 1000 | | | | | |
| V_C | Supply voltage ($\pm 5\%$) | $\pm 12 \dots 15$ | V | | | | |
| I_C | Current consumption | 10 (@ $\pm 15 \text{ V}$) + I_S | mA | | | | |
| V_d | R.m.s. voltage for AC isolation test, 50 Hz, 1 mn | 2.5 | kV | | | | |

Features

- Closed loop (compensated) current transducer using the Hall effect
- Printed circuit board mounting
- Insulated plastic case recognized according to UL 94-V0.

Special feature

- Overmoulded with an organic material
- Risk of deterioration when in contact with various chemical agents such as acids, alcohols and chlorinated solvents (see LEM Spec. n° CA010705)
- Good protection against severe humidity.

Accuracy - Dynamic performance data

| | | | | |
|--------------|---|---|------------|------------------|
| X | Accuracy @ $I_{PN}, T_A = 25^\circ\text{C}$ | @ $\pm 15 \text{ V} (\pm 5\%)$ | ± 0.65 | % |
| | | @ $\pm 12 \dots 15 \text{ V} (\pm 5\%)$ | ± 0.90 | % |
| ϵ_L | Linearity | | < 0.15 | % |
| I_O | Offset current @ $I_P = 0, T_A = 25^\circ\text{C}$ | Typ | Max | |
| I_{OM} | Residual current ³⁾ @ $I_P = 0$, after an overload of $3 \times I_{PN}$ | | ± 0.3 | mA |
| I_{OT} | Thermal drift of I_O | $0^\circ\text{C} \dots \neq 70^\circ\text{C}$ | ± 0.1 | ± 0.5 |
| | | $-25^\circ\text{C} \dots + 85^\circ\text{C}$ | ± 0.1 | ± 0.6 |
| t_{ra} | Reaction time @ 10 % of $I_{P \max}$ | | < 500 | ns |
| t_r | Response time @ 90 % of $I_{P \max}$ | | ≤ 1 | μs |
| di/dt | di/dt accurately followed | | > 200 | A/ μs |
| f | Frequency bandwidth (-1 dB) | | DC .. 200 | kHz |

Advantages

- Excellent accuracy
- Very good linearity
- Low temperature drift
- Optimized response time
- Wide frequency bandwidth
- No insertion losses
- High immunity to external interference
- Current overload capability.

Applications

- AC variable speed drives and servo motor drives
- Static converters for DC motor drives
- Battery supplied applications
- Uninterruptible Power Supplies (UPS)
- Switched Mode Power Supplies (SMPS)
- Power supplies for welding applications.

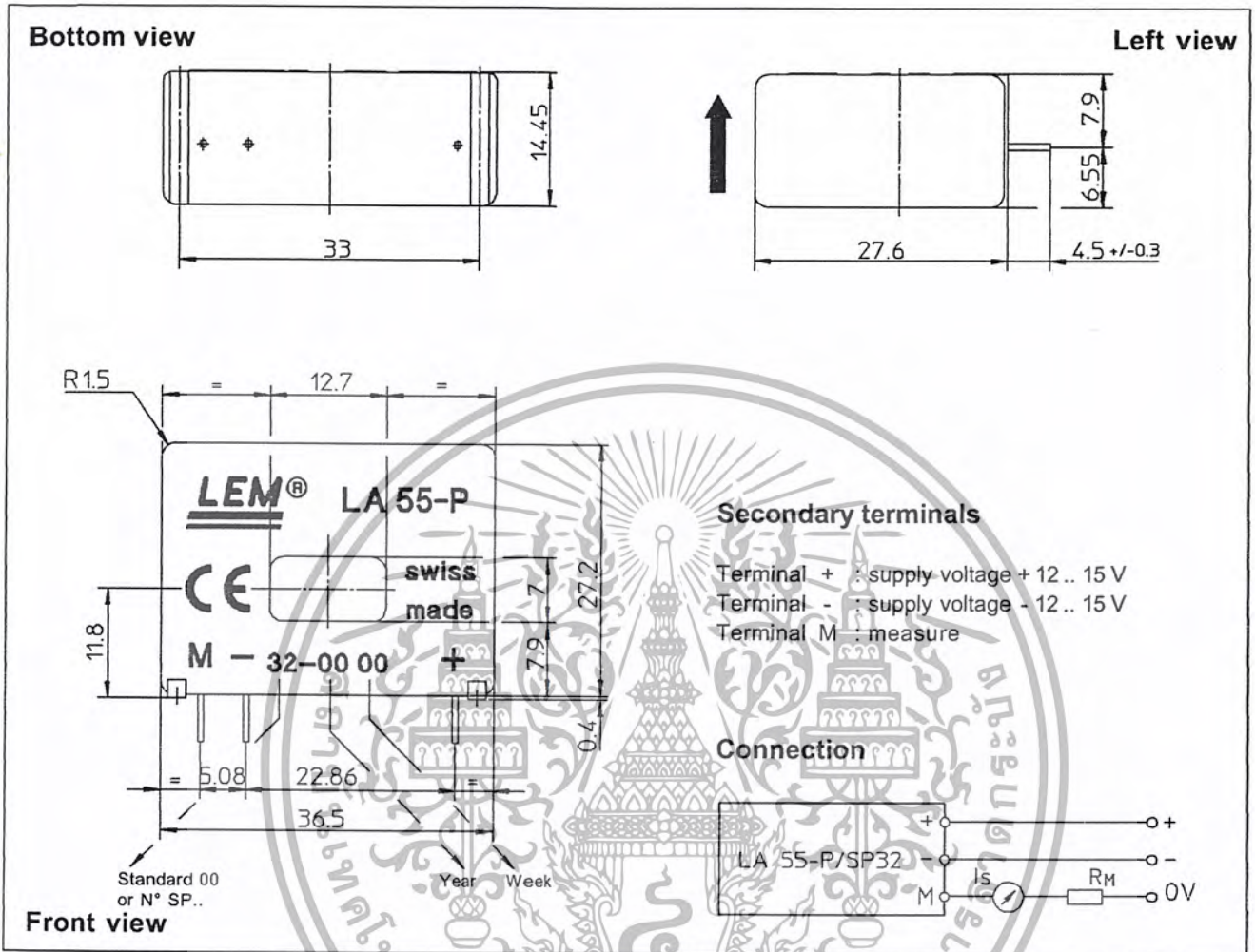
General data

| | | | |
|-------|-------------------------------|--------------------------|------------------|
| T_A | Ambient operating temperature | -25 .. +85 | $^\circ\text{C}$ |
| T_S | Ambient storage temperature | -40 .. +90 | $^\circ\text{C}$ |
| R_S | Secondary coil resistance @ | $T_A = 70^\circ\text{C}$ | 80 |
| | | $T_A = 85^\circ\text{C}$ | 85 |
| m | Mass Standards ⁴⁾ | 21 | g |
| | | EN 50178 | |

- Notes : ¹⁾ Measuring range limited to $\pm 60 \text{ A}$
²⁾ Measuring range limited to $\pm 55 \text{ A}_{\max}$
³⁾ Result of the coercive field of the magnetic circuit
⁴⁾ A list of corresponding tests is available

010723/2

Dimensions LA 55-P/SP32 (in mm. 1 mm = 0.0394 inch)



Mechanical characteristics

- General tolerance ± 0.2 mm
- Primary through-hole 12.7 x 7 mm
- Fastening & connection of secondary 3 pins
- Recommended PCB hole 0.63 x 0.56mm
- Recommended PCB hole 0.9 mm

Remarks

- I_s is positive when I_p flows in the direction of the arrow.
- Temperature of the primary conductor should not exceed 90°C.
- Dynamic performances (di/dt and response time) are best with a single bar completely filling the primary hole.
- In order to achieve the best magnetic coupling, the primary windings have to be wound over the top edge of the device.
- Avoid contact with chemical agents such as alcohol or chlorinated solvents which may cause corrosive damage to the transducer housing.

LEM reserves the right to carry out modifications on its transducers, in order to improve them, without previous notice.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณ บิดา มารดา ที่ให้กำเนิดและสนับสนุนเรื่องของทุนการศึกษาและ โครงการงานชิ้นนี้ ได้สำเร็จลุล่วงได้ด้วยดี โดยได้รับความช่วยเหลือสนับสนุนและอำนวยความสะดวกจากหลายท่าน ผู้จัดทำ ขอกล่าวขอบพระคุณ ผศ.ดร.วิจิตร กิมเรศ อาจารย์ที่ปรึกษาโครงการที่ช่วยให้คำปรึกษาและวางแนวทาง ในการทำโครงการ ขอขอบคุณที่ ป.โท ในภาควิชา ฟิสิกส์, ฟิสิกส์, ฟิสิกส์, ฟิสิกส์, ฟิสิกส์, ฟิสิกส์, ฟิสิกส์ ที่ช่วยให้ความรู้ และให้ยืมอุปกรณ์ที่ใช้ในการทำโครงการ ภาควิชาวิศวกรรมไฟฟ้าที่สนับสนุนเครื่องมือและอุปกรณ์ที่ใช้ ในการทดลอง รวมทั้งห้องสมุดที่เป็นแหล่งสนับสนุนด้านตำราและหนังสืออ้างอิงจึงขอกราบขอบพระคุณ ทุกท่านไว้ ณ โอกาสนี้ด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] นิรุช อำนวยศิลป์, “เขียนเกมส้อย่างมืออาชีพด้วย Visual C++ และ Direct X” , อินโฟเพรส, กรุงเทพฯ ฯ, 2545
- [2] นิรุช อำนวยศิลป์, “คู่มือการเขียนโปรแกรม Visual C++” , อินโฟเพรส (1989), กรุงเทพฯ ฯ, 2543
- [3] พ.อ.เจนวิทย์ เหลืองอร่าม , ปิยวิทย์ เหลืองอร่าม , “การเขียนโปรแกรมด้วย C/ C++” , ซีเอ็ดยูเคชั่น, กรุงเทพฯ ฯ, 2543
- [4] ยืน ภู่วรรณ, วัฒนา เชียงกุล , “ไมโครคอมพิวเตอร์” , ซีเอ็ดยูเคชั่น , 2527
- [5] รศ.ดร.มนัส สังวรศิลป์ , วรรัตน์ ภัทรอมรกุล, “คู่มือการใช้งาน MATLAB ฉบับสมบูรณ์” , อินโฟเพรส, กรุงเทพฯ, 2543
- [6] A.E.Fitzgerald, Charles Kingsley, Jr., Stephen D. Umans, “Electric Machinery”, fifth edition in SI units, McGraw-Hill Book Company(UK)Limited, 1992
- [7] Ned Mohan, Troe M. Undeland and William P. Robbins, “Power electronics Converter, Applications, and Design”, second edition, John Wiley & Sons, Inc., 1995
- [8] Peter Vas, “Parameter Estimation, Condition Monitoring, and Diagnosis of Electrical Machines”, Clarendon Press Oxford, 1993



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้