สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

T032793

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อ การพัฒนาตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

Implementation Advance of Speech-Coding by Using Standard Vector Quantization and FPGAs

ชื่อนักศึกษา

1. นายดาบชัย	พัวพานิช	รหัสประจำตัว	40031108
2. นายวรพงศ์	เลปนานนท์	รหัสประจำตัว	40031127
3. นายวิระศักดิ์	ตรีโอษฐ	รหัสประจำตัว	40031130
4. นายศิริชัย	จำรัสเลิศลักษณ์	รหัสประจำตัว	40031132

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต **สาขาวิชา** วิศวกรรมโทรคมนาคม

อาจารย์ที่ปรึกษา อาจารย์กิติพงศ์ มะโน

อาจารย์ที่ปรึกษาร่วม อาจารย์วรวิทย์ สมหา

คณะกรรมการสอบปริญญานิพนธ์		ลายมือชื่อ
1. อาจารย์กิติพงศ์	มะโน	.....
2. อาจารย์วรวิทย์	สมหา	.....
3. อาจารย์อำพล	ทองระอา	.....
4. อาจารย์สุระชัย	พิมพ์สาลี	.....
5. อาจารย์อมรชัย	ชัยชนะ	.....

วัน/เดือน/ปีที่สอบ วันเสาร์ที่ 1 พฤษภาคม พ.ศ. 2542 เวลา 19.00 น.

สถานที่สอบ ห้อง ค.310 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงชื่อ.....

(ผศ.ดร.ธีระพล เทพหัสดิน ณ อยุธยา)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่ 1 เดือน มิ.ย. พ.ศ. 42

เลขหมู่.....
เลขทะเบียน 32793
วัน, เดือน, ปี 10 มิ.ย. 2542

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

การพัฒนาตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

Implementation of Speech-Coding by using Standard Vector Quantization and FPGAs

นายดาบชัย พัวพานิช

นายวรพงศ์ เลปนานนท์

นายวิระศักดิ์ ตรีโอษฐ์

นายศิริชัย จำรัสเลิศลักษณ์

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง การพัฒนาตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

Implementation of Speech-Coding by using Standard Vector Quantization and FPGAs

วัตถุประสงค์

1.เพื่อศึกษาเกี่ยวกับตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

2.เพื่อออกแบบตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

3.เพื่อสร้างตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

4.เพื่อทดลองและทดสอบตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

ประโยชน์ที่คาดว่าจะได้รับ

1.มีความรู้ความเข้าใจเกี่ยวกับหลักการเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

2.ได้วงจรต้นแบบและวิธีการสร้างตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

3.ได้ตัวเข้ารหัสเสียงซึ่งออกแบบจากอัลกอริทึมของการเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

4.ได้ทราบถึงประสิทธิภาพของตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs	
ชื่อนักศึกษา	นายดาบชัย	พัวพานิช
	นายวรพงศ์	เลปนานนท์
	นายวิระศักดิ์	ตรีโอษฐ์
	นายศิริชัย	จำรัสเลิศลักษณ์
อาจารย์ที่ปรึกษา	อาจารย์กิติพงค์	มะโน
อาจารย์ที่ปรึกษาร่วม	อาจารย์วรวิทย์	สมหา
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต	
สาขาวิชา	วิศวกรรมโทรคมนาคม	
ปีการศึกษา	2541	

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอ การพัฒนาตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs ซึ่งได้ออกแบบวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและสร้างให้สามารถทำงานที่เวลาจริงได้ด้วยอุปกรณ์ FPGAs โดยใช้โปรแกรม View Draw ของบริษัท View logic ในการออกแบบวงจรและใช้โปรแกรมพัฒนาอุปกรณ์ FPGAs ของบริษัท Xilinx ในการกำหนดการทำงานของวงจรด้วยอุปกรณ์ FPGAs เบอร์ 4005E สำหรับวงจรเข้ารหัสเสียงและเบอร์ 3020A สำหรับวงจรถอดรหัสเสียง โดยผลจากการทดลองเข้ารหัสสัญญาณที่ความถี่เสียงปรากฏว่าทำให้อัตราการส่งข้อมูลลดลงและสามารถทำงานที่เวลาจริงได้ โดยที่คุณภาพของเสียงอยู่ในเกณฑ์ที่ยอมรับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Implementation of Speech-Coding by using Standard Vector Quantization and FPGAs
Students	Mr.Dabchai Puapanich
	Mr.Worapong Lepananon
	Mr.Wirasak Tree-ote
	Mr.Sirichai Jamraslerdluk
Advisor	Mr.Kitipong Mano
Co-Advisor	Mr.Worawit Somha
Education Level	Bachelor of Science in Industrial Education
Program in	Telecommunication Engineering
Acadamic Year	1998

ABSTRACT

This thesis present speech coding by using standard vector quantization and FPGAs the circuit of speech coding and speech decoding were design by using the standard vector quantization and were implement at the real time by FPGAs device. The View Draw software from View Logic company used to design the circuit and the advance of FPGAs device program from Xilinx company used to control the working of circuit with FPGAs number 4005E for the speech coding and FPGAs number 3020A for the speech decoding. The result showed that the advance of speech signal has bit rate decrease and can implement at the real time. Beside, the quality of speech is satisfied.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงไปได้ด้วยดีนั้น คณะผู้จัดทำขอขอบคุณท่านอาจารย์ที่ปรึกษาทั้งสองท่าน และคณาจารย์ภาควิชาครุศาสตร์วิศวกรรมทุกท่านที่ได้ให้ความอนุเคราะห์เครื่องมือ และอุปกรณ์ ให้คำแนะนำปรึกษา แนวความคิดต่างๆ รวมทั้งแนวทางการแก้ไขปัญหาในการจัดทำปริญญานิพนธ์ พร้อมทั้งให้โอกาสในการทำงานอย่างเต็มที่ สุดท้ายนี้ที่ระลึกถึงอย่างยิ่ง บิดา มารดา ผู้ให้การสนับสนุนด้านการศึกษาทุกท่าน และเพื่อนๆ ตลอดจนผู้เกี่ยวข้องทุกท่านที่ให้ความช่วยเหลือ ความห่วงใยและกำลังใจในการทำงาน จนทำให้โครงงานชิ้นนี้สำเร็จลุล่วงไปได้ด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์	1
1.2 ขีดความสามารถของโครงงาน	3
1.3 เนื้อหาโดยสังเขป	4
บทที่ 2 ทฤษฎีและหลักการ	5
2.1 กล่าวนำ	5
2.2 การควอนไตซ์เวกเตอร์แบบมาตรฐาน	5
2.2.1 หลักการควอนไตซ์เวกเตอร์แบบมาตรฐาน	5
2.2.2 ตัวอย่างหลักการควอนไตซ์เวกเตอร์แบบมาตรฐาน	8
2.2.3 การคำนวณหาค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน	9
2.3 ทฤษฎีการแปลงสัญญาณ	10
2.4 วงจรแปลงสัญญาณแอนะลอกและวงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอก	11
2.5 ทฤษฎีการสุ่มสัญญาณ	21
2.6 ทฤษฎีและหลักการของ FPGAs	22
2.6.1 โครงสร้างภายใน	22
2.6.2 ส่วนที่เป็นองค์ประกอบของลอจิก	25
2.6.3 ส่วนอินพุตและเอาต์พุต	25
2.6.4 รายละเอียดและการใช้งาน	25
2.7 การใช้ซอฟต์แวร์ของบริษัทวิวลอจิกและของบริษัทไซลิงค์	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.7.1 ขั้นตอนในการออกแบบและจำลองการทำงาน โดยใช้ซอฟต์แวร์ของบริษัทวิวลอจิก	33
2.7.2 ขั้นตอนในการทดสอบวงจรโดยใช้ซอฟต์แวร์ของบริษัทไซลิงค์	34
บทที่ 3 การออกแบบและการสร้าง	35
3.1 กล่าวนำ	35
3.2 หลักการออกแบบวงจร	35
3.3 ขั้นตอนการออกแบบวงจร	35
3.3.1 การออกแบบวงจรเข้ารหัสเสียง	36
3.3.2 การออกแบบวงจรถอดรหัสเสียง	37
3.3.3 การแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล	39
3.3.4 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก และวงจรกรองความถี่ต่ำ	39
3.3.5 วงจรปรีไมโครโฟน	40
3.3.6 วงจรขยายสัญญาณเสียง	41
3.3.7 วงจรแหล่งจ่ายไฟ	42
3.3.8 วงจรกำเนิดความถี่	43
3.4 การสร้างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียง	43
3.4.1 การสร้างวงจรเข้ารหัสเสียง	44
3.4.2 การสร้างวงจรถอดรหัสเสียง	47
3.5 การเชื่อมต่อระหว่างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียง	50
บทที่ 4 การทดลองและผลการทดลอง	52
4.1 กล่าวนำ	52
4.2 ขั้นตอนการทดสอบการทำงาน	52
4.2.1 ขั้นตอนการทดสอบวงจรเข้ารหัสสัญญาณเสียง และวงจรถอดรหัสสัญญาณเสียง	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
4.2.2 ขั้นตอนในการสร้างไฟล์องค์ประกอบ	53
4.2.3 การดาวน์โหลดโดยใช้โปรแกรม XACT	53
4.3 โปรแกรม Serial PROM	62
4.4 ขั้นตอนการทดลอง	63
4.5 ผลการทดลอง	65
บทที่ 5 บทสรุป ปัญหา แนวทางแก้ไขและพัฒนา	67
5.1 บทสรุป	67
5.2 ปัญหาและแนวทางแก้ไข	67
5.3 แนวทางการพัฒนาโครงงาน	67
ภาคผนวก ก รายละเอียดวงจรและผลการทดสอบวงจร	68
ภาคผนวก ข รายละเอียดของอุปกรณ์	101
บรรณานุกรม	134
ประวัติผู้แต่ง	135

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 การนับเกตของเกตพื้นฐาน	23
ตารางที่ 2.2 คุณสมบัติของ FPGAs ตระกูลต่างๆ	24
ตารางที่ 2.3 รูปแบบต่างๆ ของการกำหนดองค์ประกอบในการทำงาน	26
ตารางที่ 4.1 การตั้งสวิตช์ต่างๆ ในแผงวงจรของวงจร FPGAs	54
ตารางที่ 4.2 การตั้งสวิตช์ต่างๆ ในแผงวงจรเข้ารหัสเสียง	64
ตารางที่ 4.3 การตั้งสวิตช์ต่างๆ ในแผงวงจรถอดรหัสเสียง	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
รูปที่ 2.1 การควอนไตซ์เวกเตอร์แบบมาตรฐาน	6
รูปที่ 2.2 การจัดเก็บโค้ดบุ๊คในหน่วยความจำ	7
รูปที่ 2.3 ระบบที่มีการประมวลผลข้อมูลทางดิจิตอล	11
รูปที่ 2.4 การทดลองวงจร D/A แบบรวมกระแส	13
รูปที่ 2.5 ตัวอย่างวงจร D/A แบบรวมกระแส	15
รูปที่ 2.6 ตัวอย่างวงจร D/A แบบวงจรบันไดของตัวต้านทาน	16
รูปที่ 2.7 วงจร D/A ที่ใช้กับเลขฐานสิบรหัส BCD 2 หลัก	17
รูปที่ 2.8 การทำงานของวงจรอินทิเกรเตอร์	20
รูปที่ 2.9 หลักการทำงานของวงจร A/D แบบสโลปคู่	21
รูปที่ 2.10 สัญญาณเสียงที่ผ่านการสุ่มเทียบกับสัญญาณต้นฉบับ	22
รูปที่ 2.11 แผนผัง CLB ของตระกูล 4000	27
รูปที่ 2.12 แผนผัง IOB ของตระกูล 4000	28
รูปที่ 2.13 ผังลำดับในการคอนฟิกเมื่อป้อนแหล่งจ่ายไฟเข้าในไอซีและการโปรแกรมใหม่	29
รูปที่ 2.14 การต่อใช้งานแบบสเลฟซีเรียล	30
รูปที่ 2.15 แผนภูมิเวลาการป้อนข้อมูลโปรแกรมคอนฟิกในแบบสเลฟซีเรียล	30
รูปที่ 2.16 การต่อใช้งานแบบมาสเตอร์ซีเรียล	31
รูปที่ 2.17 การต่อใช้งานแบบมาสเตอร์พาราแรล	32
รูปที่ 3.1 แผนผังการทำงานวงจรเข้ารหัสเสียง	38
รูปที่ 3.2 แผนผังการทำงานวงจรถอดรหัสเสียง	38
รูปที่ 3.3 วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล	39
รูปที่ 3.4 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก	40
รูปที่ 3.5 วงจรปรีไมโครโฟน	40
รูปที่ 3.6 วงจรขยายสัญญาณเสียง	41
รูปที่ 3.7 ลายวงจรพิมพ์วงจรปรีไมโครโฟนและวงจรขยายสัญญาณเสียง	41
รูปที่ 3.8 การวางอุปกรณ์วงจรปรีไมโครโฟนและวงจรขยายสัญญาณเสียง	41
รูปที่ 3.9 วงจรแหล่งจ่ายไฟ	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 3.10 ลายวงจรพิมพ์วงจรแหล่งจ่ายไฟ	42
รูปที่ 3.11 การวางอุปกรณ์วงจรแหล่งจ่ายไฟ	42
รูปที่ 3.12 วงจรกำเนิดความถี่	43
รูปที่ 3.13 ลายวงจรพิมพ์วงจรกำเนิดความถี่	43
รูปที่ 3.14 การวางอุปกรณ์วงจรกำเนิดความถี่	43
รูปที่ 3.15 วงจรเข้ารหัสเสียง	45
รูปที่ 3.16 ลายวงจรพิมพ์วงจรเข้ารหัสเสียง	46
รูปที่ 3.17 การวางอุปกรณ์วงจรเข้ารหัสเสียง	46
รูปที่ 3.18 วงจรเข้ารหัสเสียงประกอบสำเร็จ	47
รูปที่ 3.19 วงจรถอดรหัสเสียง	48
รูปที่ 3.20 ลายวงจรพิมพ์วงจรถอดรหัสเสียง	49
รูปที่ 3.21 การวางอุปกรณ์วงจรถอดรหัสเสียง	49
รูปที่ 3.22 วงจรถอดรหัสเสียงประกอบสำเร็จ	50
รูปที่ 3.23 การเชื่อมต่อระหว่างวงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียง	51
รูปที่ 3.24 วงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียงเมื่อประกอบสำเร็จ	51
รูปที่ 4.1 การเข้าสู่ XDM	54
รูปที่ 4.2 หน้าต่างของ Design Manager	55
รูปที่ 4.3 Dialog Box ของ Open	55
รูปที่ 4.4 Dialog Box ของ New Project	56
รูปที่ 4.5 Dialog Box ของ Translate Option	56
รูปที่ 4.6 Dialog Box ของการเลือกอุปกรณ์	57
รูปที่ 4.7 Dialog Box ของ Select Part	57
รูปที่ 4.8 Dialog Box ของ Translate Option ที่เลือกอุปกรณ์ที่ถูกต้องแล้ว	57
รูปที่ 4.9 หน้าจอกระบวนการ Translate	58
รูปที่ 4.10 หน้าจอที่แสดงว่ากระบวนการ Translate สมบูรณ์	58
รูปที่ 4.11 Dialog Box ของ XC4000 Design Implement Option	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 4.12 หน้าจอแสดงกระบวนการ Implement	56
รูปที่ 4.13 หน้าจอแสดง Dialog Box ของกระบวนการ Implement ที่เสร็จสมบูรณ์	60
รูปที่ 4.14 Dialog Box ของ Report Browser	60
รูปที่ 4.15 หน้าต่างของ Design Manager ที่พร้อมจะทำการดาวน์โหลด	61
รูปที่ 4.16 Dialog Box ของ Communication Setup	61
รูปที่ 4.17 Dialog Box ของ Hardware Debugger	62
รูปที่ 4.18 สัญญาณเสียงทางด้านอินพุต	65
รูปที่ 4.19 สัญญาณเสียงทางด้านเอาต์พุต	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

เทคโนโลยีในการสื่อสารในปัจจุบันมีการพัฒนาไปมาก การสื่อสารระบบดิจิตอลก็เป็นการสื่อสารระบบหนึ่งที่มีมากในปัจจุบัน ส่วนสำคัญประการหนึ่งที่ต้องคำนึงถึงก็คือ การลดบิตเรต (Bit Rate)ในการส่งข้อมูลให้ต่ำลง ซึ่งจะมีผลทำให้ความจุในการส่งของช่องสัญญาณมากขึ้นแต่การลดบิตเรตในการส่งข้อมูลนั้นอาจจะทำให้คุณภาพของการสื่อสารข้อมูลลดลง แนวความคิดในปริญญานิพนธ์นี้ก็คือการควอนไตซ์เวกเตอร์เป็นวิธีการที่มีประสิทธิภาพวิธีหนึ่งซึ่งนำมาประยุกต์ใช้ในการเข้ารหัสเสียงเพื่อลดบิตเรตในการส่งข้อมูลให้ต่ำลง โดยที่ยังให้คุณภาพของการสื่อสารอยู่ในเกณฑ์ที่ยอมรับได้

เมื่อปี 1948 เคลาด์ แชนนอน (Claude Shannon) ได้พัฒนาทฤษฎีการเข้ารหัสซึ่งเป็นผลงานที่ได้รับความสำเร็จอย่างมาก คือ การแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล (Analog-to-Digital Conversation) และระบบการบีบอัดข้อมูล (Data Compression) ซึ่งทฤษฎีนี้ทำให้เกิดความเที่ยงตรงอย่างมากระหว่างอัตราบิตเรตและคุณภาพของการสร้างข้อมูลกลับคืน เพื่อให้ได้ระบบการสื่อสารที่มีความถูกต้องแน่นอนและทฤษฎีนี้ได้กำเนิดแนวความคิดของโครงสร้างของการเข้ารหัสที่ดีต่อมา

แชนนอนได้นำเสนอแนวความคิด 2 แบบที่นำไปสู่เทคนิคการออกแบบการเข้ารหัสสัญญาณในเวลาต่อมา

แนวความคิดแรกสัญญาณดิจิตอลน่าจะสามารถบีบอัดได้โดยการกำหนดโค้ดเวิร์ด (Codeword) สั้นๆ ซึ่งเป็นสัญญาณที่เชื่อถือได้มากที่สุด และการเข้ารหัสที่ประสบผลสำเร็จที่สุดน่าจะหาได้จากค่าสถิติรูปร่างลักษณะของสัญญาณ ซึ่งนำไปสู่แนวความคิดเกี่ยวกับการเข้ารหัสสัญญาณที่ไม่มีความผิดพลาดหรือไม่เกิดการสูญเสีย

แนวความคิดที่สอง ระบบการเข้ารหัสสัญญาณสามารถปฏิบัติได้ดีถ้าใช้การเข้ารหัสแบบเวกเตอร์มากกว่าที่จะใช้แต่ละแซมเปิ้ล

แนวความคิดแรกของแชนนอน นำมาซึ่งการเปลี่ยนแปลงอย่างรวดเร็วของการเข้ารหัสสัญญาณหลายๆ แบบ แต่ไม่ใช้กับแนวความคิดที่สอง โดยเหตุที่แนวความคิดของการเข้ารหัสแบบเวกเตอร์ได้เกิดขึ้นหลายปีก่อนที่การเข้ารหัสสัญญาณถูกนำมาใช้ให้เกิดผล อย่างไรก็ตามระบบการเข้ารหัสเพื่อแปลงสัญญาณจากแอนะลอกเป็นสัญญาณดิจิตอลและการเข้ารหัสได้ถูกพัฒนาขึ้น ซึ่งมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสำคัญต่อการปฏิบัติงานในแบบสเกลาร์ (Scalar) เป็นอย่างมากในเวลานั้นการเข้ารหัสสัญญาณแบบเวกเตอร์หรือเวกเตอร์ควอนไตซ์ไม่สามารถนำมาใช้ในเวลาจริงได้ โดยเหตุไม่มีสิ่งเอื้ออำนวยเพียงพอ เช่น ไม่มีวงจรประมวลผลสัญญาณที่มีความเร็วเพียงพอและหน่วยความจำไม่มีขนาดใหญ่พอ รวมทั้งขาดการออกแบบอัลกอริธึม (Algorithms) สำหรับการเข้ารหัสอีกด้วยอย่างไรก็ตามในปี 1980 ได้มีการพัฒนาการออกแบบอัลกอริธึมและโครงสร้างของการเข้ารหัสที่นำมาใช้ประโยชน์ได้ การเข้ารหัสแบบเวกเตอร์หรือเวกเตอร์ควอนไตซ์ ได้เริ่มต้นนำมาใช้งานในเทคโนโลยีการเข้ารหัส การพัฒนาอย่างรวดเร็วของตัวประมวลผลสัญญาณดิจิตอลใน 10 ปีที่ผ่านมาและการให้ความสนใจอย่างกว้างขวางในระบบการเข้ารหัสแบบเวกเตอร์นำไปสู่แนวความคิดในการผลิตเป็นสินค้าที่ใช้ในการเข้ารหัสแบบเวกเตอร์เป็นพื้นฐานโดยเฉพาะการเข้ารหัสเสียงพูดและแนวคิดในการเข้ารหัส

เวกเตอร์ควอนไตซ์เป็นโครงร่างของการเข้ารหัสที่พัฒนาขึ้นโดยแชนนอน ในการพัฒนาทฤษฎีของการเข้ารหัสสัญญาณกับบรรทัดฐานที่มีลักษณะตรงกับความเป็นจริงแนวความคิดนี้ ได้ขยายการควอนไตซ์ที่ใช้แบบสเกลาร์ไปเป็นหลายๆ มิติ (Dimension) เวกเตอร์ควอนไตซ์ ทำงานในแบบเวกเตอร์แทนที่จะเป็นสเกลาร์ โครงร่างของเวกเตอร์ควอนไตซ์นี้ง่ายและเหมาะสมที่จะนำมาใช้ รหัส (Code) มักจะถูกออกแบบโดยอัลกอริธึมที่มีกลุ่มข้อมูลที่ซ้ำกัน และการออกแบบนี้สามารถนำมาใช้ในการออกแบบเวกเตอร์ควอนไตซ์ ที่เพิ่มความหนาแน่นของสัญญาณได้อีกด้วย

ในทางทฤษฎีแล้วแนวความคิดนี้น่าจะเป็นไปได้ แต่ในทางปฏิบัติแล้วไม่ง่ายเลยอุปสรรคที่สำคัญได้แก่ การนำเสนอที่เวลาจริง (Real Time) และข้อจำกัดในการนำเวกเตอร์ควอนไตซ์มาใช้งาน สมมุติว่าเราพิจารณาการเข้ารหัสสัญญาณที่มีรูปร่างลำดับการสุ่มคงที่และเราเพ่งเล็งถึงปัญหาที่ทำอย่างไร ที่จะแบ่งลำดับไปเป็นเวกเตอร์ เพื่อใช้ในเวกเตอร์ควอนไตซ์โดยทั่วไป ถ้าเราบังคับให้ค่ารีโซลูชั่น (Resolution) วัดในรูปของบิตเวกเตอร์ เราทราบว่าการทำงานของเวกเตอร์ควอนไตซ์ สามารถเพิ่มไดเมนชั่น (N) ของเวกเตอร์ที่เพิ่มขึ้นได้เท่านั้น การปรับปรุงการทำงานเกี่ยวกับไดเมนชั่นนี้สามารถอธิบายได้โดยอาศัยการจำได้ (Recongnizing) พื้นที่ของโค้ดบุ๊ค (Codebook) ที่ต้องการจัดเก็บในรูปเวิร์ดและการค้นหา (จำนวนของการทำงานต่ออินพุตเวกเตอร์ในการค้นหาอย่างละเอียดในโค้ดบุ๊ค) เป็นปฏิภาคกับ LN (L คือ ขนาดของโค้ดบุ๊ค) ด้วยเหตุนี้ $LN=N2^{bn}$ หนึ่งเวิร์ดที่เก็บไว้สามารถแทนได้ด้วยหนึ่งส่วนประกอบของหนึ่งโค้ดเวกเตอร์ และการทำงานหนึ่งครั้งเทียบได้กับการทำงานแบบสเกลาร์ 2 หรือ 4 ค่า จากที่ทราบว่า สัญญาณพูดจะถูกสุ่มที่ความถี่ Fs = 8KHz สมมุติว่าเราต้องการใช้เวกเตอร์ควอนไตซ์ที่มีขนาด 1 บิตต่อแซมเปิ้ล สำหรับอัตราการส่งขนาด 8 Kbps. ขนาด N แซมเปิ้ล ในหนึ่งเวกเตอร์ต้องการโค้ดบุ๊คขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$L=2^N$ และพื้นที่ของการจัดเก็บ $N2^N$ จำนวนของการทำงานต่อหนึ่งหน่วยเวลา สำหรับการค้นหาในโค้ดบุ๊คกับค่าความผิดพลาดขึ้นอยู่กับความเร็วของตัวประมวลผล โดย $Nfs=2^Nfs$ ตัวอย่างเช่น

ให้ค่า N=16 จะมีขนาดของโค้ดบุ๊ค L=65536 สำหรับตัวประมวลผลสัญญาณดิจิตอลที่ดีที่สุดในปี 1993

การออกแบบดิจิตอลได้มีวิวัฒนาการอย่างรวดเร็ว โดยใช้ช่วงเวลาเพียงเล็กน้อย ซึ่งกระบวนวิธีการออกแบบได้เปลี่ยนแปลงอย่างรวดเร็ว ส่วนประกอบต่างๆ ของวงจรได้พัฒนาจากทรานซิสเตอร์ จนกระทั่งนำไปสู่เทคโนโลยีวีแอลเอสไอ (Very Large Scale Integrated Circuit : VLSI) ทั้งนี้ได้มีการนำระบบคอมพิวเตอร์ช่วยในการออกแบบอุปกรณ์ลอจิกที่สามารถโปรแกรมได้ (Program Logic Device) ซึ่งจะช่วยลดความยุ่งยากในการทำต้นแบบของวงจร โดยทำขั้นตอนการออกแบบทางกายภาพ (Physical) ใช้เวลาน้อยลง อย่างไรก็ตามผู้ที่ทำการออกแบบจะต้องเข้าใจ เทคโนโลยีด้านสถาปัตยกรรมของฮาร์ดแวร์ของอุปกรณ์ชนิดนี้พร้อมกับต้องเข้าใจการใช้ซอฟต์แวร์ที่ช่วยในการออกแบบควบคู่ไปด้วยเช่นกัน

1.2 ขีดความสามารถของโครงงาน

การดำเนินการกับวงจรที่ได้นั้นแบ่งเป็นกลุ่มหลักๆ ได้ 2 กลุ่มคือ แบบเต็มรูปแบบ (Fully Custom) และไม่เต็มรูปแบบ (Semi Custom) โดยในแบบเต็มรูปแบบนั้นเราไม่สามารถดำเนินการออกแบบได้จนสิ้นสุดกระบวนการเพราะในขั้นตอนสุดท้ายเราจะต้องให้โรงงานผลิตเป็นไอซี (IC) ให้แต่ในที่นี้เราจะเลือกใช้แบบไม่เต็มรูปแบบเนื่องจากเราสามารถดำเนินการได้เองจนสิ้นสุดกระบวนการออกแบบโดยวิธีที่เราเลือกใช้คือ การใช้ FPGAs (Field Programable Gate Arrays)

FPGAs จะรวมอะเรย์ (Arrays) ของเกตกับอุปกรณ์โปรแกรมได้ของ PLDs (PLDs จะประกอบด้วยอะเรย์ของเกตที่สามารถโปรแกรมได้) อุปกรณ์ที่ใช้ในการเชื่อมต่ออาจเป็นฟิวส์ (Fuse) หรือทรานซิสเตอร์ (Transistor) โดย PLDs จะเป็นอุปกรณ์ที่มีความหนาแน่นต่ำที่สุดในอุปกรณ์ประเภทที่สามารถโปรแกรมลอจิก การใช้ประโยชน์จะเปลี่ยนแปลงไปตามการประยุกต์ใช้งาน แต่มันสามารถโปรแกรมตามแบบได้ต่ำเพราะสถาปัตยกรรมของเกตมีคงที่ตายตัว โดยภายในตัว FPGAs จะมีส่วนที่สำคัญคือเส้นทางเชื่อมต่อระหว่างบล็อกซึ่งเส้นทางเหล่านี้เราสามารถโปรแกรมได้ ซึ่งจะกล่าวในบทต่อไป จำนวนเกตที่เราสามารถนำมาใช้ได้มีอยู่ระหว่าง 3000 ถึง 10000 เกต ซึ่งถือว่าใช้ประโยชน์ได้กว้างขวางเพราะเราสามารถโปรแกรมให้ทำงานเป็นฟังก์ชั่น (Function) ต่างๆ ได้ และยังมีข้อดีคือ เราสามารถโปรแกรมใหม่ได้อีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 เนื้อหาโดยสังเขป

จะเห็นได้ว่าอุปกรณ์ FPGAs เป็นเทคโนโลยีที่น่าศึกษา จึงได้ทำการออกแบบโครงงานขึ้นมาโดยมีเป้าหมายคือ สร้างอุปกรณ์เพื่อเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์และอุปกรณ์ FPGAs โดยใช้เครื่องมือพัฒนาของบริษัทวิวลอจิก (View Logic) และไซลิงค์ (Xilinx) และทำการดาวน์โหลด (Down Load) ลง FPGAs ซึ่งเนื้อหาในปริญญานิพนธ์นี้ประกอบด้วยรายละเอียด ต่าง ๆ ที่สำคัญดังต่อไปนี้

บทที่ 2 ทฤษฎีและหลักการ กล่าวถึงการควอนไตซ์เวกเตอร์แบบมาตรฐาน หลักการควอนไตซ์เวกเตอร์แบบมาตรฐาน การแปลงสัญญาณแอนะลอกเป็นดิจิตอลและการแปลงสัญญาณดิจิตอลเป็นแอนะลอก ทฤษฎีการสุ่มสัญญาณ ทฤษฎีและหลักการของ FPGAs และการใช้ซอฟต์แวร์ของบริษัทวิวลอจิก และบริษัทไซลิงค์

บทที่ 3 หลักการออกแบบและการสร้าง กล่าวถึงขั้นตอนการออกแบบและการสร้างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียงและการเชื่อมต่อระหว่างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียงเข้าด้วยกัน

บทที่ 4 การทดลองและผลการทดลอง กล่าวถึงขั้นตอนในการดาวน์โหลดโปรแกรม การโปรแกรมพรอม (PROM) ขั้นตอนการออกแบบและผลการทดลอง

บทที่ 5 การสรุปอภิปรายและข้อเสนอแนะ กล่าวถึงปัญหา แนวทางแก้ไขและแนวทางในการพัฒนาเพื่อสามารถนำไปประยุกต์ใช้งานจริง

ภาคผนวก แสดงรายละเอียดของวงจร ผลการทดสอบวงจร และรายละเอียดของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 กล่าวนำ

สำหรับทฤษฎีที่ใช้ในการออกแบบโครงงานนี้ ประกอบด้วย ทฤษฎีของเวกเตอร์ควอนไตซ์ ทฤษฎีการแปลงสัญญาณ ทฤษฎีการสุ่มสัญญาณ ทฤษฎีและหลักการของ FPGAs และการใช้ซอฟต์แวร์วิวลอจิกและไซลิงค์ ซึ่งจะนำเสนอเป็นส่วนๆ ไป

2.2 การควอนไตซ์เวกเตอร์แบบมาตรฐาน

2.2.1 หลักการการควอนไตซ์เวกเตอร์แบบมาตรฐาน

การควอนไซต์เวกเตอร์แบบมาตรฐานเป็นการควอนไตซ์เวกเตอร์แบบแรกสุดที่ได้มีการนำไปประยุกต์ใช้งานในการเข้ารหัสสัญญาณเสียงและเป็นพื้นฐานในการพัฒนาการควอนไตซ์เวกเตอร์ในเวลาต่อมา ซึ่งวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานนี้ได้นำเสนอวิธีการออกแบบโค้ดบุ๊คซึ่งเป็นที่รู้จักกันดีคือวิธีการแบบ LBG (LBG Algorithms) หรือในบางครั้งเราเรียกว่าวิธีการแบบ Spliting Algorithms ซึ่งได้พัฒนามาจากวิธีการของ Lloyd (Lloyd Algorithms) หลักการ ควอนไตซ์เวกเตอร์แบบมาตรฐาน

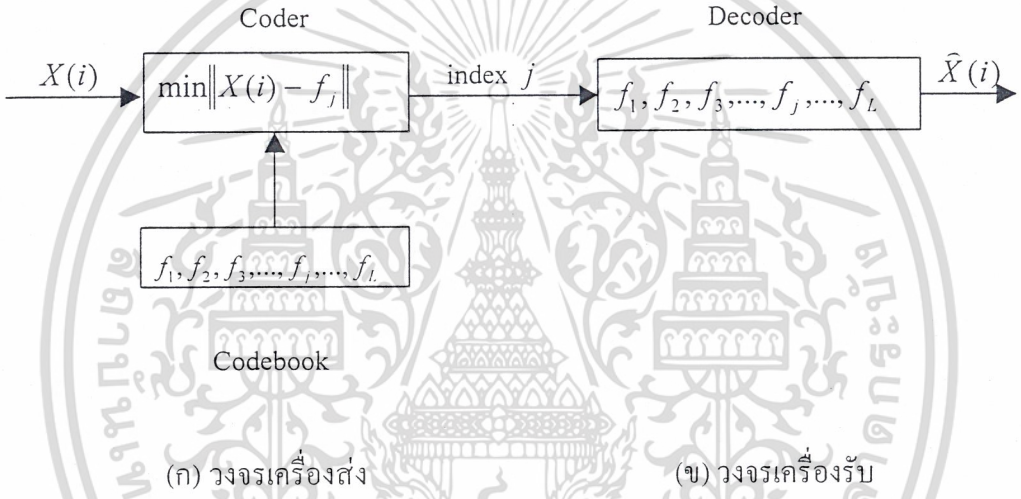
รูปที่ 2.1 เป็นหลักการพื้นฐานของการควอนไตซ์เวกเตอร์แบบมาตรฐาน วงจรเครื่องส่ง (Transmiter) ประกอบไปด้วยส่วนที่เป็นโค้ดบุ๊คเวกเตอร์ที่มี N มิติ ซึ่ง N คือ จำนวนของตัวอย่าง (Samples) หรืออีลีเมนต์ (Element) ในหนึ่งเวกเตอร์ และมีจำนวน L เวกเตอร์ ในหนึ่งโค้ดบุ๊ค คือ $f_j \{ j = 1,2,3,\ldots,L\}$ โดยเป็นส่วนที่มีความสำคัญเนื่องจากโค้ดบุ๊คจะต้องเป็นที่เก็บคุณลักษณะของเสียงไว้ทั้งหมดและการจัดโค้ดบุ๊คเวกเตอร์ในหน่วยความจำได้จัดในลักษณะของอะเรย์ 2 มิติ ซึ่งแสดงได้ดังรูปที่ 2.2 และจากรูปที่ 2.1 ในขบวนการเริ่มต้นสัญญาณอินพุตเวกเตอร์ $X(i)$ จะถูกสุ่มเข้ามา จำนวน N แซมเปิ้ล (เท่ากับมิติของโค้ดบุ๊คหรือหนึ่งเวกเตอร์) โดยวงจรเข้ารหัสจะแทนสัญญาณเวกเตอร์ $X(i)$ ด้วยค่าเวกเตอร์ f_j จากโค้ดบุ๊คซึ่งจะมีเพียงหนึ่งเวกเตอร์ที่มีคุณลักษณะใกล้เคียงกับสัญญาณอินพุตเวกเตอร์ $X(i)$ มากที่สุดโดยการหาค่าต่ำสุด (Minimize the Euclidean Distance) ซึ่งมีเงื่อนไขดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\|X(i) - f_j\| < \|X(i) - f_K\| \tag{2.1}$$

โดยที่ $k = 1,2,3,\ldots,L$; $k \neq j$

และจะได้ค่าดัชนี (Index) ของเวกเตอร์จากโค้ดบุ๊คคือ j มาทำการถอดรหัสจากโค้ดบุ๊คที่เครื่องรับ โดยที่โค้ดบุ๊คที่เครื่องส่งและเครื่องรับจะต้องมีข้อมูลที่เหมือนกันหลังจากนั้นจะได้สัญญาณเอาต์พุต เวกเตอร์ที่ถูกสร้างขึ้นใหม่คือ เวกเตอร์ $\hat{X}$ (i) ซึ่งมีค่าเท่ากับเวกเตอร์ f_j ที่เก็บ อยู่ในโค้ดบุ๊คนั้นเอง โดยที่เวกเตอร์ f_j จะต้องมีค่าใกล้เคียงกับอินพุตเวกเตอร์ $\hat{X}$ (i) มากที่สุด



รูปที่ 2.1 การควอนไตซ์เวกเตอร์แบบมาตรฐาน

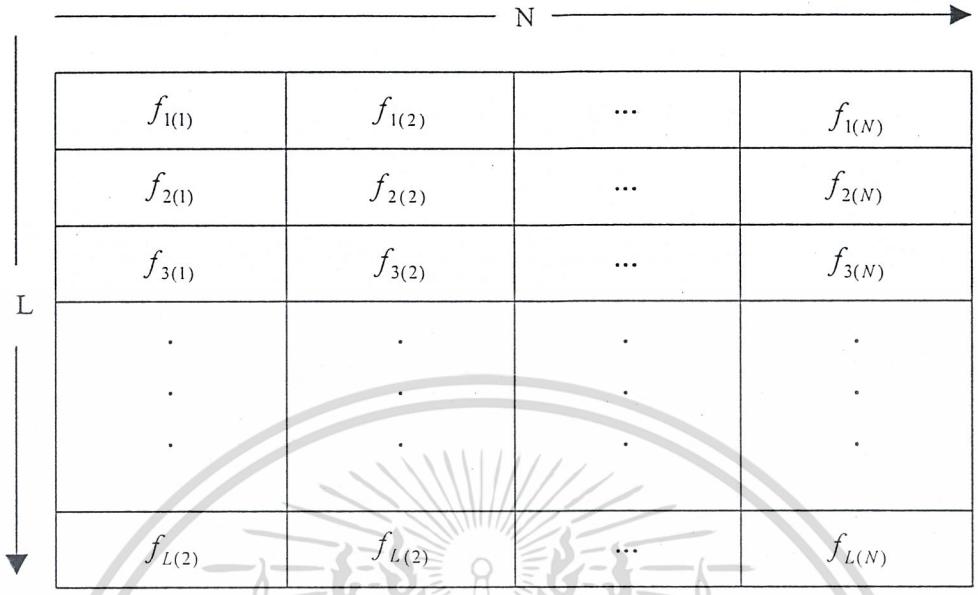
ค่า Parameter ที่สำคัญอีกประการหนึ่งของการควอนไตซ์เวกเตอร์คือ ความละเอียด (Resolution) ของจำนวนบิตที่ใช้ในหนึ่งอีลีเมนต์ของเวกเตอร์สามารถคำนวณได้ดังนี้

$$R = \frac{Log_2 L}{N} (bit / element) \tag{2.2}$$

สมมติถ้ากำหนดให้โค้ดบุ๊คมีมิติ N=5 และจำนวนเวกเตอร์ในโค้ดบุ๊ค L=64 จะได้

$$R = \frac{Log_2 64}{N} = \frac{6}{5} = 1.2 bit / element \tag{2.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การจัดเก็บโค้ดบุ๊คในหน่วยความจำ

นั่นหมายความว่าในหนึ่งแซมเปิ้ลของการสุ่มสัญญาณที่ส่งไปยังเครื่องรับต้องการข้อมูลเพียง1.2 บิต จะสังเกตเห็นว่าความละเอียดของตัวแปลงสัญญาณ A/D จะไม่มีผลต่อบิตเรตในการส่งข้อมูล ถ้าหากเรานำหลักการควอนไตซ์เวกเตอร์ไปประยุกต์ใช้ในการเข้ารหัสสัญญาณเสียงและจากสมการที่ (2.2) สามารถคำนวณหาค่าบิตเรตในการส่งข้อมูลได้ดังนี้

$$Bitrate = R \times Fs(bit/\sec) \tag{2.4}$$

Fs = ความถี่ในการสุ่มสัญญาณ (Hz)

ถ้าอัตราการสุ่มสัญญาณ Fs=8kHz มิติของโค้ดบุ๊ค N= 5 (หมายถึงการเข้ารหัสในแต่ละครั้งใช้สัญญาณจำนวน 5 แซมเปิ้ล) และจำนวนของเวกเตอร์ L=64 นั่นหมายความว่าค่าดัชนี เวกเตอร์ j ที่ส่งไปยังเครื่องรับจำนวน 6 บิต สามารถสร้างสัญญาณเอาต์พุตเวกเตอร์ $\hat{X}(i)$ ได้จำนวน 5 แซมเปิ้ล ดังนั้นจำนวนบิตเรตในการส่งข้อมูลจะได้เป็น

$$Bitrate = 1.2 \times 8000 = 9600(bit/\sec) \tag{2.5}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ตัวอย่างการควอนไตซ์เวกเตอร์แบบมาตรฐาน

เพื่อแสดงให้เห็นถึงหลักการเข้ารหัสและการลดบิตเรตในการส่งข้อมูลให้ชัดเจนยิ่งขึ้น จึงได้ยกตัวอย่าง หลักการควอนไตซ์เวกเตอร์ แบบมาตรฐาน โดยกำหนดให้โค้ดบุ๊คมีมิติ N=2 และจำนวนของเวกเตอร์ในโค้ดบุ๊ค L=4 ดังนี้

$$f_1 = \begin{bmatrix} 3 \\ 5 \end{bmatrix}, f_2 = \begin{bmatrix} 4 \\ 9 \end{bmatrix}, f_3 = \begin{bmatrix} 7 \\ -5 \end{bmatrix}, f_4 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

สัญญาณอินพุตเวกเตอร์ที่ถูกสุ่มเข้ามาจำนวน 2 ตัวอย่าง (เท่ากับมิติของโค้ดบุ๊ค) $\hat{X}(i) = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$ และความถี่ในการสุ่มสัญญาณ F_S= 8000Hz

บิตเรตในการส่งข้อมูลจะได้ดังนี้

$$Bitrate = \frac{Log_2 L}{N} \times F_S = \frac{Log_2 4}{2} \times 8000 = 8000bps \qquad (2.6)$$

คำนวณหาสัญญาณเอาต์พุตเวกเตอร์ $\hat{X}(i)$ ได้จาก $\min\|X(i) - f_j\|$;j=1,2,3,...,L

$$\|X(i) - f_1\| = \left\| \begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 3 \\ 5 \end{bmatrix} \right\| = \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \sqrt{\begin{bmatrix} 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}} = 1 \qquad (2.7)$$

$$\|X(i) - f_2\| = \left\| \begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 4 \\ 9 \end{bmatrix} \right\| = \begin{bmatrix} -1 \\ -5 \end{bmatrix} = \sqrt{\begin{bmatrix} -1 & -5 \end{bmatrix} \begin{bmatrix} -1 \\ -5 \end{bmatrix}} = 5.09 \qquad (2.8)$$

$$\|X(i) - f_3\| = \left\| \begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 7 \\ -5 \end{bmatrix} \right\| = \begin{bmatrix} -4 \\ 9 \end{bmatrix} = \sqrt{\begin{bmatrix} -4 & 9 \end{bmatrix} \begin{bmatrix} -4 \\ 9 \end{bmatrix}} = 9.84 \qquad (2.9)$$

$$\|X(i) - f_4\| = \left\| \begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\| = \begin{bmatrix} 3 \\ 3 \end{bmatrix} = \sqrt{\begin{bmatrix} 3 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 3 \end{bmatrix}} = 4.24 \qquad (2.10)$$

เมื่อพิจารณาผลของการเปรียบเทียบสัญญาณอินพุตเวกเตอร์ $X(i)$ เทียบกับโค้ดบุ๊คทั้ง 4 เวกเตอร์ $\min\|X(i) - f_j\|$ จะเห็นได้ว่าผลของการเปรียบเทียบนั้น สัญญาณอินพุตเวกเตอร์ $X(i)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีค่าใกล้เคียงกับเวกเตอร์ f_1 มากที่สุด (ผลจากการลบระหว่าง $X(i)$ และ f_1 มีค่าน้อยที่สุด) ดังนั้นข้อมูลที่ส่งไปที่เครื่องรับคือ 00 (binary) สัญญาณเอาต์พุตที่เครื่องรับจะมีค่าเป็น $\hat{X}(i)=\begin{bmatrix}3\\5\end{bmatrix}$

2.2.3 การคำนวณค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน

การวัดคุณภาพของเสียงที่ได้จากการควอนไตซ์เวกเตอร์กระทำได้โดยการทดสอบหาค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน (SNRseg หรือ Segment Signal to Noise Ratio) ระหว่างสัญญาณอินพุต $X(i)$ และสัญญาณเอาต์พุต $\hat{X}(n)$ โดยทำการแบ่งสัญญาณ $X(n)$ และ $\hat{X}(n)$ เป็นเฟรม (Frame) พร้อมทั้งทำการคำนวณครั้งละหนึ่งเฟรมซึ่งเขียนแทนด้วย SNRm และในแต่ละเฟรมจะใช้จำนวนตัวอย่าง 128 ดังนี้

$$SNRm = 10Log_{10}\frac{\sum_{n=1}^{128}\{X(n)\}^2}{\sum_{n=1}^{128}\{X(n)-\hat{X}(n)\}^2} \qquad (2.11)$$

$X(n)$ = สัญญาณอินพุต

$\hat{X}(i)$ = สัญญาณที่ผ่านขบวนการควอนไตซ์เวกเตอร์

หลังจากนั้นจะนำเอาค่า SNRm ทั้งหมดที่ได้จากสมการที่ (2.11) มาทำการหาค่าเฉลี่ยซึ่งเขียนแทนด้วย SNRseg ดังนี้

$$SNRseg = \frac{1}{S}\sum_{m=1}^{S} SNRm(dB) \qquad (2.12)$$

S คือ จำนวนของเฟรมทั้งหมด

สำหรับการคำนวณในแต่ละเฟรมถ้าหาก $\sum_{n=1}(X(i))^2 < 32\times10^4$ (หรือต่ำกว่า 43 dB ของสัญญาณ Sinusoidal ที่แปลงสัญญาณ A/D แปลงได้เต็มสเกล) เราจะทำการตัดการคำนวณในเฟรมนี้ออกไปทั้งนี้เนื่องจากสัญญาณที่เกิดขึ้นในระดับนี้ส่วนใหญ่จะเป็นสัญญาณรบกวน (Noise) ที่เกิดจากขบวนการควอนไตซ์ของ A/D เพราะสัญญาณเสียงที่พูดที่ได้จะต้องมีพลังงาน (Energy) ที่มากกว่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

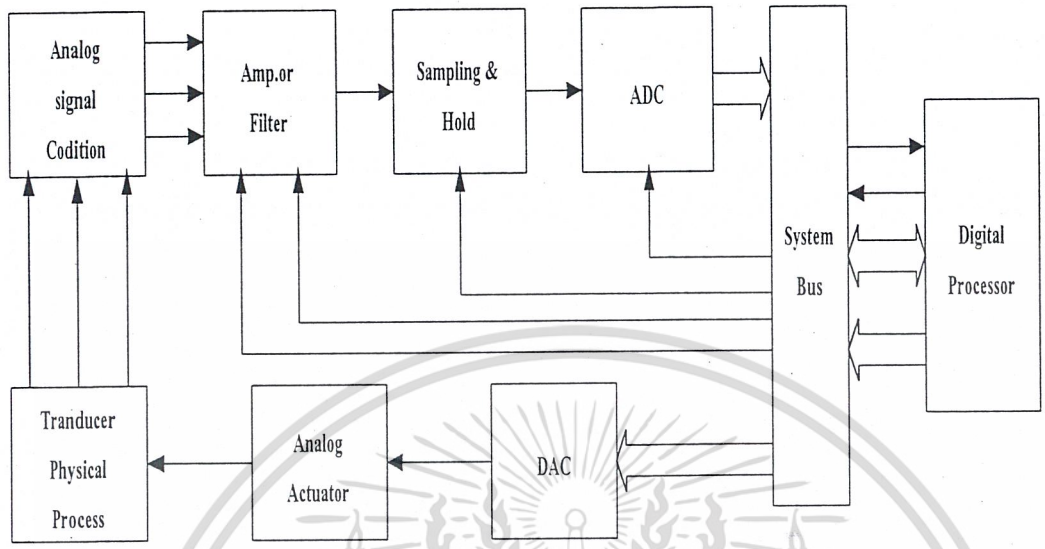
2.3 ทฤษฎีการแปลงสัญญาณ (Data Acquistion Conversion)

รูปแบบของสัญญาณไฟฟ้าโดยมากมักอยู่ในรูปสัญญาณแอนะลอก การนำเอาสัญญาณไฟฟ้ามาประมวลผล เพื่อให้เกิดรูปแบบที่ต้องการนั้น ต้องใช้อุปกรณ์ทางแอนะลอกแต่ปัจจุบันนี้เทคโนโลยีทางด้านดิจิตอลก้าวหน้าไปมาก ทำให้การประมวลผลสัญญาณทางดิจิตอล สามารถทำได้อย่างรวดเร็ว และมีประสิทธิภาพ

ดังนั้นการแปลงรูปแบบสัญญาณ (Conversion) จึงมีความจำเป็นในการแปลงสัญญาณแอนะลอกที่มีอยู่แล้วให้เป็นสัญญาณดิจิตอล โดยอุปกรณ์การแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล และจะถูกประมวลผลโดยตัวประมวลผลสัญญาณดิจิตอล เช่น คอมพิวเตอร์ เป็นต้น จากผลลัพธ์ที่ได้อาจถูกนำมาแสดงผลโดยตรงเลย หรืออาจถูกแปลงให้อยู่ในรูปของสัญญาณแอนะลอกที่ใช้งานได้ การที่จะแปลงสัญญาณแอนะลอกเป็นดิจิตอลได้นั้น สามารถทำได้โดยใช้อุปกรณ์แปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก สำหรับระบบที่มีการประมวลผลข้อมูลทางดิจิตอลแสดงดังรูปที่ 2.3

จากรูปที่ 2.3 การเปลี่ยนแปลงทางกายภาพในลักษณะใดๆ ก็ตาม เช่น อุณหภูมิ ความดัน ความเร็ว จะถูกเปลี่ยนให้มาเป็นสัญญาณไฟฟ้าแบบแอนะลอก โดยทรานสดิวเซอร์ (Transducer) ทฤษฎีการสุ่มที่มีรูปแบบที่เหมาะสมกับลักษณะทางกายภาพนั้นๆ จากสัญญาณทางไฟฟ้าก็จะถูกปรับให้อยู่ในรูปแบบ และขนาดที่เหมาะสมก่อน โดยวงจรต่างๆ เช่น วงจรขยาย หรือวงจรกรองสัญญาณเป็นต้น วงจรแซมปลิ้งแอนด์โฮลด์ (Sampling and Hold) จะสุ่มขนาดของสัญญาณแอนะลอกมาแล้วจะทำการโฮลด์สัญญาณนั้นไว้ชั่วขณะ โดยไม่จำเป็นต้องใช้ วงจร ADC แล้วข้อมูลทางดิจิตอลจะถูกส่งต่อไปยังบัส (Bus) ระบบ จากนั้นตัวโปรเซสเซอร์ จะทำการประมวลผลข้อมูลกลับมาเพื่อควบคุมกิจการทางกายภาพของระบบโดยผ่านตัวกระทำทางกล (Analog Actuator)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ระบบที่มีการประมวลผลข้อมูลทางดิจิตอล

2.4 วงจรแปลงสัญญาณแอนะลอกเป็นดิจิตอล และวงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอก

การติดต่อระหว่างมนุษย์ส่วนมากจะใช้สัญญาณต่อเนื่อง (Analog) เป็นสัญญาณติดต่อกัน แต่ว่าการทำงานของระบบคอมพิวเตอร์ จะใช้สัญญาณเป็นช่วงดิจิตอลเป็นสัญญาณในการทำงาน ดังนั้นถ้าเราต้องการให้คอมพิวเตอร์ช่วยเราทำงานแล้ว เราจะต้องเปลี่ยนสัญญาณแอนะลอกที่เราใช้อยู่ให้เป็นสัญญาณดิจิตอลเพื่อให้คอมพิวเตอร์หรือเครื่องประมวลสัญญาณดิจิตอลรับรู้เมื่อประมวลสัญญาณเสร็จก็จะส่งข้อมูลออกมาเป็นสัญญาณดิจิตอล ซึ่งเป็นเรื่องที่ยุ่งยากที่เราจะเข้าใจข้อมูลนั้น ดังนั้นพอสรุปได้ว่าการเปลี่ยนสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลและการเปลี่ยนแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก ซึ่งเป็นการประสานโลกของคอมพิวเตอร์ให้เข้ากับโลกมนุษย์ได้ เพื่อให้มนุษย์ได้ใช้คอมพิวเตอร์ได้อย่างสมบูรณ์ยิ่งขึ้น การเปลี่ยนแปลงสัญญาณแอนะลอกไปเป็นดิจิตอลเรียกว่า Analog to Digital Converter (ADC) หรือเรียกย่อๆว่า A TO D หรือ A/D ในทำนองเดียวกันการเปลี่ยนแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกเรียกว่า Digital to Analog Converter (DAC หรือ D/A)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลงสัญญาณดิจิตอลกับแอนะลอก

อุปกรณ์ที่ภายในใช้สัญญาณดิจิตอลทั้งหมด เช่น นาฬิกาดิจิตอล เครื่องคิดเลข และคอมพิวเตอร์ มีน้อยมาก อุปกรณ์ส่วนใหญ่ยังต้องรับอินพุต และส่งเอาต์พุตในรูปของสัญญาณแอนะลอกอยู่ ดังนั้นการแปลงสัญญาณแอนะลอกเป็นดิจิตอลซึ่งเรียกว่า A/D Converter และการแปลงสัญญาณดิจิตอลเป็นแอนะลอกหรือที่เรียกว่า D/A Converter จึงเป็นเรื่องสำคัญของวงจรดิจิตอล ถ้าเรามีทั้งวงจร A/D และ D/A ครบแล้ว การสร้างอุปกรณ์ทางอิเล็กทรอนิกส์ไม่ว่าจะซับซ้อนเท่าใดก็สามารถทำได้ง่ายขึ้น อุปกรณ์จะใช้วงจร A/D ในการแปลงสัญญาณอินพุตต่าง ๆ ที่เป็นสัญญาณแอนะลอกให้เป็นดิจิตอล ป้อนสัญญาณดิจิตอลที่ได้เข้าในวงจรดิจิตอลเพื่อประมวลผลจากนั้นแสดงเป็นตัวเลขและใช้วงจร D/A แปลงสัญญาณแอนะลอกใหม่เพื่อเป็นเอาต์พุตนำออกไปใช้ในการควบคุมกระบวนการภายนอกได้

1) วงจร D/A

เริ่มจากวงจร D/A (Digital Analog Converter) ซึ่งเป็นการแปลงสัญญาณดิจิตอลเป็นแอนะลอกก่อน เพราะวงจรทำได้ง่ายกว่าวงจร A/D

วงจร D/A มีด้วยกันหลายแบบด้วยกัน เราคงไม่สามารถพูดอธิบายได้ทุกแบบในที่นี้จะขอกล่าววงจร D/A แบบรวมกระแสกับแบบวงจรบันไดของตัวต้านทาน

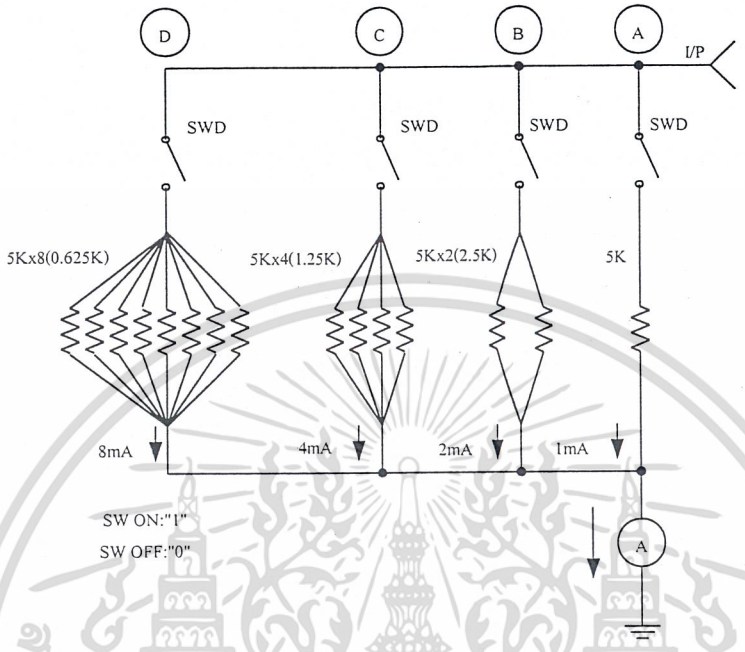
2) วงจร D/A แบบรวมกระแส

ลองต่อวงจรในรูปที่ 2.4 เป็นหลักการของวงจร D/A แบบรวมกระแส (Current Summing Type) นี้เราจะใช้สวิตช์ 4 ตัว ใช้ตัวต้านทาน 5 กิโลโอห์ม จำนวน 15 ตัว และมิเตอร์วัดกระแสพิสัยไม่เกิน 20 มิลลิแอมป์ สวิตช์ทั้ง 4 ตัวนี้คือ SWA, SWB, SWC, และ SWD ซึ่งเปรียบเสมือนปริมาณดิจิตอลแต่ละบิต

เมื่อปิดสวิตช์หมายถึง "1" ถ้าเปิดสวิตช์หมายถึง "0" เมื่อสวิตช์ทั้งสี่แทนเลขฐานสอง SWA จะเป็นบิตนัยสำคัญต่ำสุด (บิตต่ำสุดทางขวามือ) และ SWD จะเป็นบิตสำคัญสูงสุด

เมื่อปิดสวิตช์แต่ละตัวจะมีกระแสขนาดต่างกันไหลมาที่มิเตอร์วัดกระแส เช่น ปิดสวิตช์ SWA จะมีกระแสไหล 1 มิลลิแอมป์ ปิดสวิตช์ SWB จะมีกระแสไหล 2 มิลลิแอมป์ เป็นต้น ถ้าปิดสวิตช์หลายสวิตช์พร้อมกันจะมีกระแสไหลจากแต่ละกิ่งมารวมกันแล้วไหลไปยังมิเตอร์ และกระแสที่ไหลมาที่มิเตอร์นี้คือ ปริมาณแอนะลอกที่แปลงมาได้จากวงจรนี้นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การทดลองวงจร D/A แบบรวมกระแส

สวิตช์ทั้งสี่เรียงกันเป็นเลขฐานสอง 4 บิต ดังนั้นตัวเลขที่สามารถแสดงได้คือ 0-15 เช่นเดียวกัน กระแสที่เป็นปริมาณแอนาลอกก็สามารถมีค่าได้ระหว่าง 0-15 มิลลิแอมป์ เลขฐานสองเพิ่มทีละ 0001 ในขณะที่กระแสเพิ่มทีละ 1 มิลลิแอมป์ สวิตช์แต่ละตัวจะมี "น้ำหนัก" หรือขนาดกระแสที่บิตออกมาได้ต่างกันคือ SWA มีน้ำหนัก 1,SWB มีน้ำหนัก 2,SWC มีน้ำหนัก 4 และ SWD 8 น้ำหนักที่เพิ่มทีละเท่าขึ้นมาเรื่อยๆ ตามหลักการของเลขฐานสองนั่นเอง

เราใช้ตัวต้านทานขนาด 5 กิโลโอห์มเพราะเราใช้แหล่งจ่ายไฟ 5 โวลต์และต้องการให้กระแสไหลผ่านตัวต้านท่านแต่ละตัวเพียง 1 มิลลิแอมป์เท่าๆ กัน เลือกใช้ตัวต้านทานเหมือนกัน 15 ตัวเพื่อให้ทดลองได้ง่ายและเข้าใจง่าย จะใช้ตัวต้านทาน 2.5 กิโลโอห์มเพียงตัวเดียวต่อที่ SWB หรือ 1.25 กิโลโอห์มต่อที่ SWC ก็ได้ แต่ค่าความต้านทานเหล่านี้หาได้ลำบากมาก

ลองเปิดสวิตช์ SWA จนถึง SWD ตามเลขฐานสองจะได้กระแสไหลผ่านมิเตอร์ตามที่แสดงในตารางในรูปที่ 1 เช่น ถ้าปิด SWA จะอ่านมิเตอร์ได้ 1 มิลลิแอมป์ ปิดสวิตช์ SWA, SWB จะได้ 3 มิลลิแอมป์ และถ้า ปิดสวิตช์ทุกตัวจะได้ 15 มิลลิแอมป์ เป็นต้น

การทดลองนี้ใช้สวิตช์เพียง 4 ตัวเท่านั้น ที่จริงเราสามารถเพิ่มสวิตช์ขึ้นอีกไปเรื่อยๆ ค่าความต้านทานของสวิตช์ถัดมาจะน้อยลงไปที่ละเท่า แต่ไม่ควรเพิ่มสวิตช์มากเพราะกระแสจะไหลมากเกินไปจนเกินพิสัยของมิเตอร์ ถ้าต้องการทดลองกับสวิตช์หลายๆ ตัวควรลดขั้น กระแสลงจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

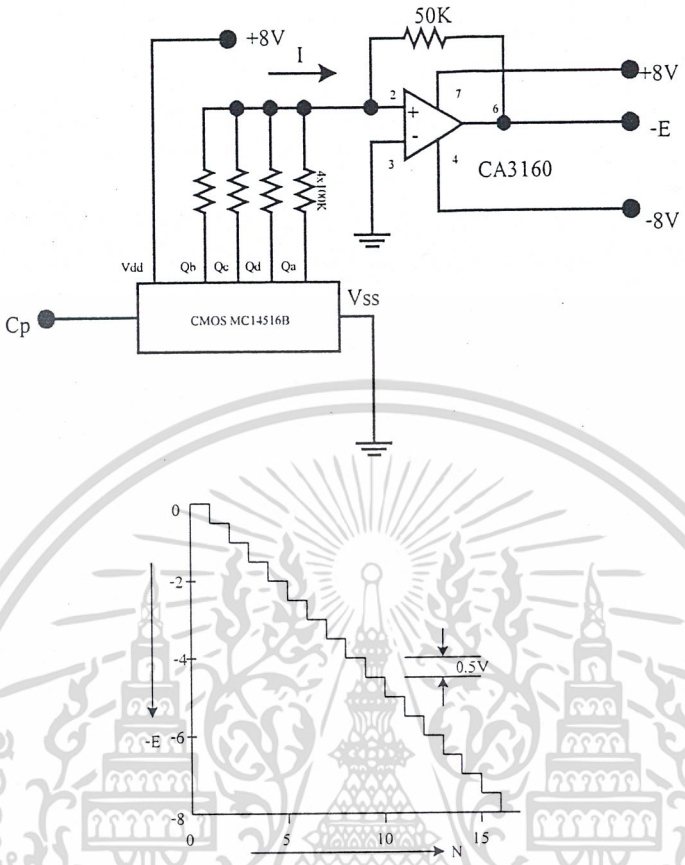
1 มิลลิแอมป์เป็น 0.1 หรือ 0.01 มิลลิแอมป์ก็ได้ โดยการเพิ่ม R จาก 5 กิโลโอห์มเป็น 50 หรือ 500 กิโลโอห์ม

ใช้สวิตช์ดังรูปที่ 2.4 อาจจะยังมองไม่เห็นการใช้งานที่เป็นจริงลองดูวงจรในรูปที่ 2.5 วงจรนี้ใช้หลักการเดียวกันคือการรวมกระแส แต่ใช้เอาต์พุตของวงจรนับชนิด CMOS แทนสวิตช์ทั้งสี่ เอาต์พุต QA ถึง QD ของวงจรนับ เมื่อเป็น "1" จะจ่ายกระแสเข้าไปในวงจรเช่นเดียวกับการปิดสวิตช์ SWA ถึง SED ปกติ CMOS จ่ายกระแสได้ไม่มากนักเพราะไหลน้อย และแทนที่จะใช้มิเตอร์วัดกระแสต่อโดยตรง เราจะใช้ออปแอมป์ (OP-AMP) ต่อรับกระแสแทนและแปลงกระแสเป็นแรงดันออกทางเอาต์พุตของออปแอมป์ด้วย

ขั้วลบ (ขา 2) ของออปแอมป์จะมีศักย์ไฟฟ้า เท่ากับขาบวก (ขา 3) ตามหลักการทำงานของออปแอมป์ ขั้วลบจึงมีศักย์ไฟฟ้าเป็นศูนย์ จึงทำให้กระแส I ที่ไหลจากตัวต้านทานคำนวณให้ได้เหมือนรูปที่ 2.4 นอกจากนั้นกระแส I นี้จะไหลเข้าออปแอมป์เลย เพราะออปแอมป์มีความต้านทานสูงมาก กระแสทั้งหมดจะไหลผ่านตัวต้านทาน R ไปยังเอาต์พุต แรงดันที่ขาออกที่เอาต์พุตจึงเท่ากับแรงดันตกคร่อม R นั่นเอง แรงดัน Eo นี้จะเป็นค่าลบและจะมีขนาด 0.5 โวลต์ ต่อการเปลี่ยนแปลงสัญญาณดิจิตอล 1 ขั้นถ้าเปลี่ยน ค่า R ให้เล็กหรือใหญ่ขึ้นก็สามารถเปลี่ยนขนาดของแรงดันขาออกนี้ได้

เมื่อเราป้อนพัลส์เข้าวงจรนับ เอาต์พุต QA จนถึง QD จะเริ่มเปลี่ยนแปลงและเพิ่มค่าตามหลักการของเลขฐานสอง เอาต์พุตใดเป็น "1" ก็จะจ่ายกระแสออกไป กระแสที่จ่ายออกไปนี้จะมีขนาดตาม "น้ำหนัก" ของบิตนั้น ถ้านำมัลติมิเตอร์มาวัดแรงดันขาออกของออปแอมป์ในขณะที่วงจรนับกำลังพัลส์จะเห็นเข็มมิเตอร์เพิ่มขึ้นเรื่อย ๆ จาก 0 โวลต์เป็น -0.5,-1,-1.5,-2.0 โวลต์โดยเพิ่มทีละ -0.5 โวลต์แบบนี้ไปเรื่อยๆ จนถึง -7.5 โวลต์ ในที่สุด จากนั้นจะตกกลับเป็น 0 โวลต์ใหม่แล้วจึงค่อยๆ เพิ่มแรงดันขึ้นไปอีก ถ้าใช้ออสซิลโลสโคป (Oscilloscope) จับรูปคลื่น Eo จะเห็นรูปคลื่นคล้ายขั้นบันไดตามรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ตัวอย่างวงจร D/A แบบรวมกระแส

3) วงจร D/A แบบวงจรบันไดของตัวต้านทาน

วงจร D/A แบบแรกเป็นแบบรวมกระแส วงจร D/A ที่จะกล่าวต่อไปเป็นแบบรวมแรงดัน ใช้ตัวต้านทานเพียง 2 ค่าคือ R และ 2R (ตัวต้านทานที่มีค่า 2*R โอห์ม) ต่อสลับไปมาเหมือนขั้นบันไดหรือเรียกว่า วงจร D/A แบบวงจรบันไดของตัวต้านทาน ตามที่แสดงในรูปที่ 2.6 ในรูปเป็นวงจร D/A ขนาด 8 บิต ใช้วงจรนับ 4 บิต 2 ตัวต่ออนุกรมกันและใช้เอาต์พุตของวงจรนับทำหน้าที่เป็นสวิตช์ป้อนเลขฐานสองเข้าวงจร D/A

วงจรบันไดของตัวต้านทานนี้ผู้ออกแบบครั้งแรกนับว่ามีความสามารถมาก เพราะไม่ว่าวัดที่จุดไหนของแต่ละขั้นบันไดจะได้ความต้านทานรวมลงมาถึงปลายบันไดเท่ากับ R ดังนั้นถ้าบิตสูงสุดก็คือ QD2 เป็น “1” จะทำให้แรงดันขาออกเป็นครึ่งหนึ่งของแหล่งจ่ายไฟคือ VDD2 ถ้าบิตถัดไปคือ QC2 เป็น “1” ก็จะให้แรงดันขาออกลดลงไปครึ่งหนึ่ง VDD/4 บิตถัดๆ ไปก็จะทำให้แรงดันลดไปทีละครึ่งเรื่อยๆ จนถึงบิตสุดท้ายคือ QA1 จะให้แรงดันที่เอาต์พุตเป็น VDD/256 ถ้านับ แรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

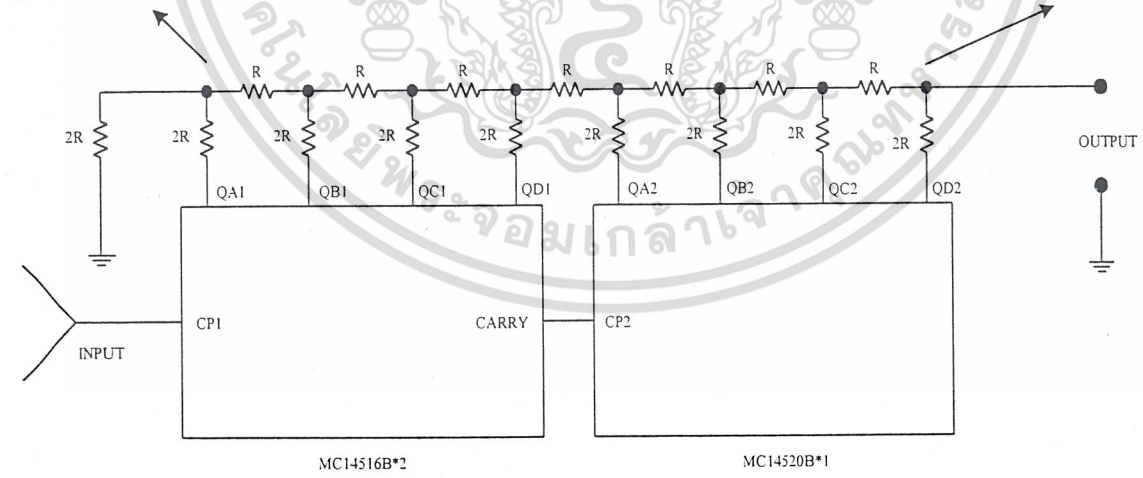
VD/256 เป็นแรงดัน 1 หน่วย แต่ละบิตก็จะให้แรงดันคิดเป็นน้ำหนักเรียงจากบิตต่ำสุดไปสูงสุดคือ 1,2,3,...,128 เหมือนกับแบบรวมกระแสตามที่ได้อธิบายมาแล้ว

ถ้าป้อนพัลส์ให้วงจรนับและวัดแรงดันขาออกด้วยมัลติมิเตอร์ก็จะเห็นเข็มมิเตอร์ค่อยๆ กระดิกขึ้นหรือถ้าใช้ออสซิลโลสโคปจับดูรูปคลื่นก็จะเห็นเป็นรูปคลื่นฟันเลื่อยแบบขั้นบันได เหมือนกัน ตัวต้านทาน R และ 2R ที่ใช้ในวงจรควรเป็นตัวต้านทานที่มีความแม่นยำสูงมาก เช่น ความผิดพลาดไม่เกิน 1 เปอร์เซ็นต์ ถ้าผิดพลาดมากจะทำให้สัญญาณแอนะลอกที่ได้ผิดพลาดได้

วงจรในรูปที่ 2.6 เป็นวงจร D/A ที่ใช้กับเลขฐานสอง ถ้าเป็นเลขฐานสิบรหัส BCD ซึ่งเรานิยมใช้กันมากนั้น ก็จำเป็นต้องดัดแปลงวงจรบันไดของตัวต้านทานใหม่เป็นอย่างในรูปที่ 2.6 วงจร D/A นี้มีวงจรบันไดอยู่ 2 วงจรเป็นของหลักหน่วยวงจรหนึ่งและเป็นของหลักสิบของอีกวงจรหนึ่ง แต่ละวงจรใช้ตัวต้านทาน R และ 2R เหมือนกัน สองวงจรนี้ต่อเชื่อมกันด้วยตัวต้านทานที่มีขนาด 9R เพื่อให้แรงดันขาออกของวงจรบันไดหลักหน่วยมีขนาดเล็กลง 10 เท่า เมื่อมารวมกับแรงดันขาออกของวงจรบันไดหลักสิบ ที่ใช้ค่า 9R เพราะความต้านทานขาออกของวงจรบันไดมีค่า R พอดีตามที่ได้อธิบายไปแล้ว

*แรงดันขาออกเมื่อบิตนั้นเป็น “1”

*	Vdd/256	Vdd/128	Vdd/64	Vdd/32	Vdd/16	Vdd/8	Vdd/4	Vdd/2
น้ำหนัก	1	2	4	8	16	32	64	128



รูปที่ 2.6 ตัวอย่างวงจร D/A แบบวงจรบันไดของตัวต้านทาน

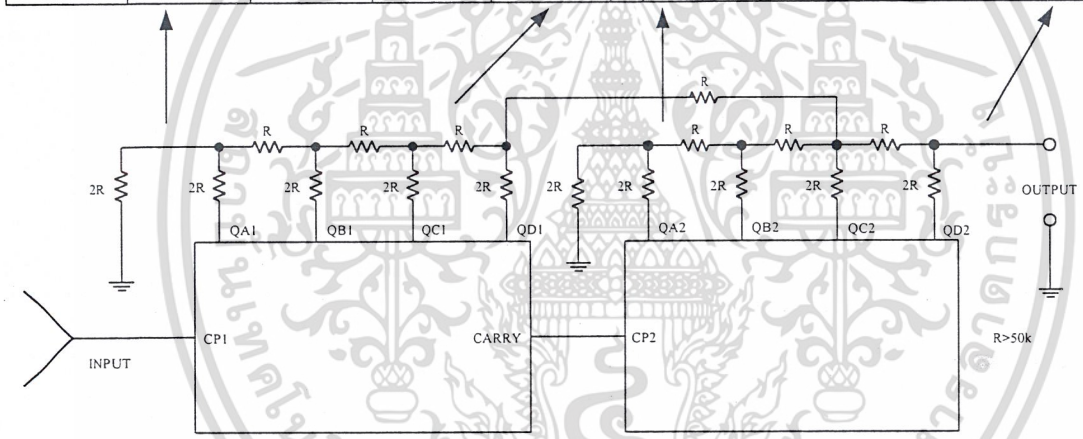
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

วงจรบันไดของตัวต้านทานนี้สามารถใช้งานในทางปฏิบัติได้ถึง 3 หลัก ของเลขฐานสิบ ถ้าเกินกว่า 3 หลักแล้ว จะต้องคัดเลือกค่าความต้านทานก็มีที่ทำเป็นไอซีสำเร็จรูปขาย ไอซีบางตัวที่เป็นวงจร D/A นั้นแท้จริงภายในจะมีวงจรบันไดของตัวต้านทานและแหล่งจ่ายกระแสที่ถูกควบคุมด้วยสัญญาณ “0” และ “1” ของอินพุตเท่านั้น ไอซี D/A ที่นิยมใช้กันมากได้แก่ เบอร์ MC 1408 ของบริษัทโมโตโรล่าเป็นวงจร D/A ขนาด 8 บิต

*แรงดันขาออกเมื่อบิตนั้นเป็น “1”

*	5Vdd / 8*110	5Vdd / 4*110	5Vdd / 2*110	5Vdd / 110	5Vdd / 8*11	5Vdd / 4*11	5Vdd / 2*11	5Vdd / 11
น้ำหนัก	1	2	4	8	10	20	40	80



เช่น MC14510B*2

MC145188*1

รูปที่ 2.7 วงจร D/A ที่ใช้กับเลขฐานสิบรหัส BCD 2 หลัก

4) วงจร A/D

วงจร A/D (Analog to Digital Converter) จะทำหน้าที่แปลงแรงดันหรือกระแสที่เป็นสัญญาณแอนะลอกไปเป็นสัญญาณตัวเลขหรือสัญญาณดิจิตอลวงจร A/D มีด้วยกันหลายแบบแต่ที่นิยมใช้กันแพร่หลายมี 3 แบบคือ แบบสโลปคู่ (Dual Slope) แบบแปลงแรงดันเป็นความถี่ (V to F Converter) และแบบประมาณทีละบิต (Successive Approximation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

32793

วงจร A/D แบบสโลปคู่เป็นแบบที่ง่ายที่สุด ไม่จำเป็นต้องใช้อุปกรณ์ที่มีคุณภาพดีมากนักก็สามารถแปลงสัญญาณได้อย่างเที่ยงตรง แต่มีข้อเสียตรงที่ใช้เวลาการแปลงสัญญาณนานมากไป จึงไม่เหมาะในการใช้วัดแรงดันในช่วงเวลาสั้นๆ เช่น การวัดรูปคลื่น ณ จุดเวลาใดเวลาหนึ่ง แบบสโลปคู่นี้เหมาะสมสำหรับใช้วัดค่าเฉลี่ยของแรงดันและกระแส จึงนิยมใช้กันมากในมัลติมิเตอร์แบบดิจิตอล และเครื่องวัดแสดงผลเป็นตัวเลขทั่วๆ ไป วงจร A/D แบบสโลปคู่ที่เป็นไอซีสำเร็จรูปมีด้วยกันหลายเบอร์ ราคาไม่แพงมากนัก ส่วนมากจะให้ความเที่ยงตรงในการแปลงสัญญาณดีกว่า 0.1 เปอร์เซ็นต์ มีทั้งแบบแปลงเป็นตัวเลขขนาด 3 หลัก และ 4 หลัก หลักสุดท้ายจะแสดงค่าเป็น 0 หรือ 1 เท่านั้น จึงเรียกง่ายๆ ว่า หลัก

วงจร A/D แบบแปลงแรงดันเป็นความถี่และแบบประมาณทีละบิตนั้นมีข้อดีตรงที่สามารถแปลงสัญญาณได้รวดเร็ว มีความเที่ยงตรงดี เพียงแต่วงจรมีความซับซ้อนมากกว่าจึงมีราคาแพง

5) วงจร A/D แบบสโลปคู่

ให้ดูวงจรในรูปที่ 2.8 ซึ่งเป็นวงจรอินทิเกรเตอร์ (Integrator) แบบพื้นฐาน อุปกรณ์ที่มีสัญลักษณ์เป็นสามเหลี่ยมนั้นเป็นออปแอมป์ที่ทำหน้าที่เป็นวงจรขยายสัญญาณแตกต่าง R และ C ในวงจรเป็นอุปกรณ์สำคัญที่ทำให้เกิดการประจุเข้าไปใน C ทำให้ได้รูปคลื่นแรงดันขาออกซึ่งเท่ากับเป็นการอินทิเกรตสัญญาณอินพุต

สมมติว่ามีแรงดัน E_i ป้อนเข้าทางอินพุต จะมีการเกิดกระแส I ซึ่งมีค่า E_i/R ไหลผ่าน R ไหลเข้าไปในวงจร กระแสนี้จะไม่ไหลเข้าออปแอมป์ เพราะออปแอมป์มีความต้านทานขาเข้าสูง แต่จะไหลผ่านไปประจุที่ตัวเก็บประจุ C ทั้งหมดเป็นผลให้แรงดันคร่อม C สูงขึ้นไปเรื่อยๆ ขั้วลบของออปแอมป์จะมีศักย์ไฟฟ้าเป็น 0 โวลต์ เท่ากับขั้วบวกซึ่งต่อลงกราวนด์ (Ground) ดังนั้นแรงดันตกคร่อม C จึงเป็นแรงดันลบเมื่อเทียบกับกราวนด์ แรงดันขาออก E_o ซึ่งเท่ากับแรงดันตกคร่อม C ซึ่งเป็นลบ แรงดันนี้ก็จะค่อยเพิ่มขึ้นเรื่อยๆ เป็นเส้นตรง แรงดันขาออกวงจรจึงเหมือนกับเป็นการอินทิเกรตแรงดันขาเข้า เพราะเมื่อเราอินทิเกรตค่าคงที่ได้เส้นตรงที่เปลี่ยนแปลงตามเวลา

รูปที่ 2.8 แสดงรูปคลื่นขาออกของวงจรอินทิเกรตเมื่อป้อนแรงดันอินพุตต่างๆ เช่น กรณีอินพุตเป็นบวก กรณีอินพุตเป็นลบ และกรณีอินพุตเป็นทั้งบวกและลบโดยเป็นบวกนาน t_1 วินาที และเป็นลบนาน t_2 วินาที จะเห็นว่าถ้าอินพุตเป็นบวกเอาต์พุตจะเป็นเส้นตรงที่มีสโลปเป็นลบ และถ้าอินพุตเป็นลบจะได้สโลปเป็นบวก การให้วงจรอินทิเกรตเตอร์ทำการอินทิเกรตขึ้นและลงแบบนี้จะได้เส้นตรงสโลปคู่

รูปที่ 2.9 เป็นรูปแสดงหลักการทำงานของวงจร A/D แบบสโลปคู่ อินพุตของ วงจรอินทิเกรตเป็นแรงดันไฟที่จะวัด E_i และมีอีกอินพุตหนึ่งเป็นแรงดันไฟมาตราฐาน E_{REF} อินพุตทั้งสองนี้มีขั้วสลับกันเสมอและจะผลัดกันต่อเข้ากับวงจรอินทิเกรเตอร์ ทำให้มีการอินทิเกรตขึ้นและลงเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จังหวะอยู่ตลอดเวลา เอาต์พุต E_O ของวงจรอินทิเกรเตอร์จะต่อเข้ากับวงจรเปรียบเทียบ ซึ่งจะทำหน้าที่ตรวจจับว่ามีแรงดัน E_O เมื่อใดเป็น 0 โวลต์

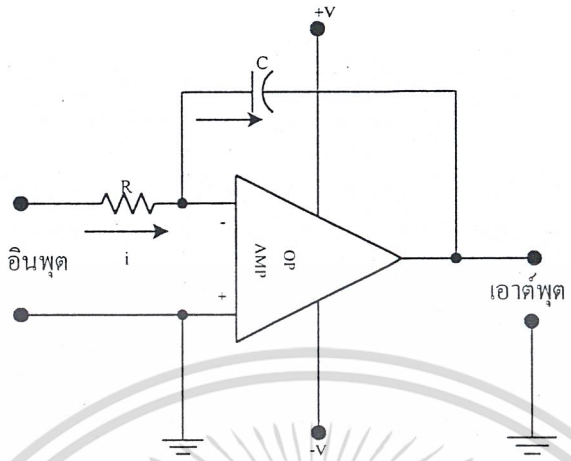
วงจรควบคุมในวงจร A/D นี้เป็นวงจรดิจิตอล ทำหน้าที่ ควบคุมการทำงานของวงจรทั้งหมดที่สำคัญ จะควบคุมสวิตช์ S_1 และ S_2 เพื่อต่ออินพุตให้กับวงจรอินทิเกรเตอร์

การทำงานของวงจรเริ่มจาก เมื่อแรงดัน E_O=0 โวลต์ วงจรควบคุมจะปิดสวิตช์เพื่อให้ แรงดันที่จะวัด E_i ต่อเข้ากับวงจรอินทิเกรเตอร์ วงจรควบคุม จะปล่อยให้วงจรอินทิเกรเตอร์ทำการอินทิเกรตสัญญาณ E_i เป็นระยะเวลา t_1 นี้ทำได้ไม่ยากนัก โดยทั่วไปมักจะใช้วงจรนับ นับพัลส์ที่มีความถี่คงที่จนได้ค่าที่กำหนดไว้

เมื่อครบเวลา t_1 วงจรควบคุมจะเปิดสวิตช์ S_1 ในขณะนั้นแรงดันเอาต์พุตของอินทิเกรเตอร์ E_O จะมีค่าแปรผันกับแรงดัน E_i จากนั้นวงจรควบคุมจะปิดสวิตช์ S_2 เพื่อต่อ แรงดันไฟมาตรฐาน E_{REF} เข้ากับวงจรอินทิเกรเตอร์และรีเซตวงจรนับ แรงดัน E_{REF} จะมีขั้วตรงข้ามกับ E_i สมมติให้ E_{REF} เป็นแรงดันลบ คราวนี้วงจรอินทิเกรเตอร์จะอินทิเกรตสัญญาณทำให้เกิดเส้นตรงที่มีสโลปตรงข้ามแรงดัน E_O จะเพิ่มขึ้นเรื่อยๆ จนในที่สุด E_O= 0 โวลต์ ในช่วงนี้สมมติว่าใช้เวลา t_2 ขณะเดียวกันวงจรนับจะนับพัลส์ไปเรื่อยๆ เมื่อ E_O = 0 โวลต์ วงจรเปรียบเทียบจะตรวจจับเหตุการณ์นี้แล้วส่งสัญญาณไปยังวงจรควบคุม วงจรควบคุมจะเปิดสวิตช์ S_2 และให้วงจรนับหยุดนับพัลส์ ค่าที่นับได้ในวงจรนับขณะนั้นจะเท่ากับค่าแรงดัน ที่ต้องการวัดพอดีที่เป็นเช่นนี้เพราะแรงดัน E_{REF} มีค่าคงที่ช่วงเวลา t_2 ในการอินทิเกรตสัญญาณจนแรงดันเป็นศูนย์ จะแปรผันกับแรงดันขาเข้า E_i ถ้า E_i มีค่ามาก ค่า E_O หลักจากการอินทิเกรตครั้งแรกจะมาก จึงใช้เวลา t_2 ในการอินทิเกรตครั้งที่ 2 มากตามไปด้วย สมมติให้ E_i=1 โวลต์ E_{REF}=-1 โวลต์ ได้ t_1= t_2 เท่ากับการนับพัลส์ 100 ลูก ถ้าให้ E_i=1.5 โวลต์ จะได้ t_2=1.5t_1 หรือเท่ากับการพัลส์ 1500 ลูก จะสังเกตเห็นว่าค่านับพัลส์นี้ตรงกับค่าแรงดัน E_i ซึ่งเป็นแรงดันที่ต้องการวัด

ค่านับของวงจรนับนี้สามารถส่งออกไปที่แลตช์ (Latch) และถอดรหัสเพื่อแสดงผลด้วย LED 7 Segment อีกทีหนึ่ง ทำให้เราสามารถอ่านค่าแรงดันเป็นตัวเลขได้โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

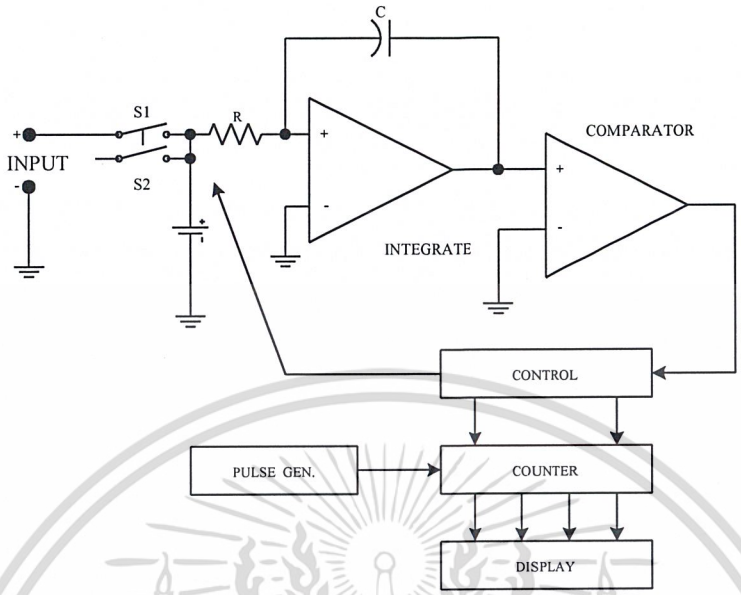


รูปที่ 2.8 การทำงานของวงจรอินทิเกรเตอร์

ดิจิตอลมัลติมิเตอร์ที่มีขายตามท้องตลาดทั่วไป มักแสดงผลสูงสุดเป็น 1999 แสดงว่า t_2 จะเท่ากับการนับพัลส์ไม่เกิน 2000 ลูก ถ้าในวงจรเราใช้พัลส์ความถี่ 10 กิโลเฮิรตซ์ ในการวัดหนึ่งครั้งจะต้องมีการอินทิเกรต 2 ครั้ง กินเวลา $t_1 + t_2$ หรือเท่ากับการนับพัลส์จำนวนสูงสุดไม่เกิน 1000+2000 ลูก ซึ่งจะกินเวลา 0.3 วินาที นับว่าเป็นเวลาที่ยาวนานพอสมควร ในหนึ่งวินาทีจะวัดได้ไม่เกิน 3 ครั้ง เท่านั้น ถ้าเป็นดิจิตอลมัลติมิเตอร์ที่แสดงผลสูงสุดเป็น 19999 ก็ยิ่งมีการนับพัลส์เป็นจำนวนมากขึ้นถึง 10 เท่า ถ้าใช้พัลส์ความถี่ 10 กิโลเฮิรตซ์เท่าเดิม จะต้องใช้เวลาถึง 3 วินาที ในการวัดหนึ่งครั้งซึ่งยาวนานเกินไป ดังนั้นคงต้องเพิ่มความถี่ของพัลส์ให้สูงขึ้น

ไอซีที่เป็นวงจร A/D แบบสโลปคู่นี้มีด้วยกันหลายเบอร์ และผลิตกันหลายบริษัท บางเบอร์ก็ใช้เพียงไอซีตัวเดียว บางเบอร์ก็ใช้ไอซี 2-3 ตัวต่อเป็นชุด ปกติมักใช้อุปกรณ์ภายนอก เช่น RC ต่อเพิ่มอีกเพียงเล็กน้อยก็สามารถใช้งานได้ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

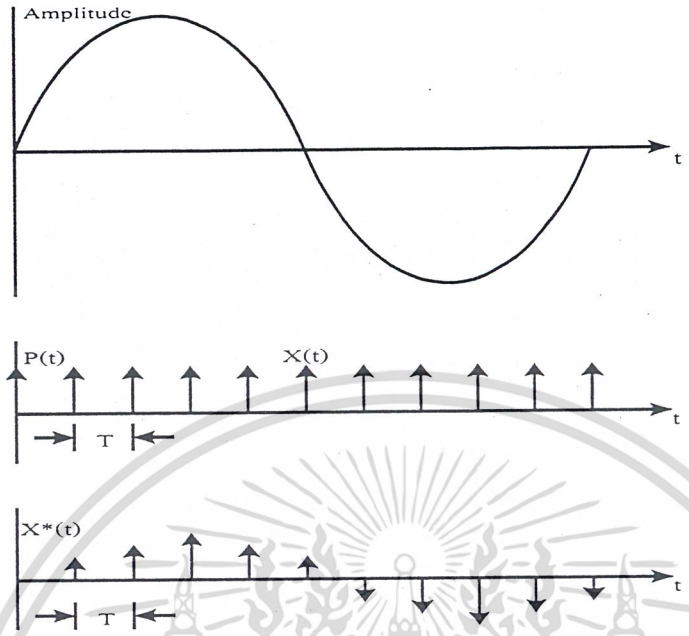


รูปที่ 2.9 หลักการทำงานของวงจร A/D แบบสโลปคู่

2.5 ทฤษฎีการสุ่มสัญญาณ (Sampling)

การสุ่มสัญญาณเป็นขั้นตอนแรกในการแปลงสัญญาณแอนะลอกให้เป็นสัญญาณดิจิตอล โดยวงจรสุ่มสัญญาณจะตรวจจับขนาดของสัญญาณแอนาลอกที่ถูกส่งตามช่วงเวลาที่ถูกกำหนด โดยทั่วไปมักจะวัดขนาดในรูปของแรงดันไฟฟ้า ซึ่งแท้จริงแล้วกระบวนการสุ่มสัญญาณ เป็นกระบวนการตรวจวัดค่าแรงดันของสัญญาณ ที่ช่วงเวลาต่างๆ ซึ่งมีคาบการตรวจจับคงที่ อัตราหรือความถี่ของการสุ่มสัญญาณเป็นคาบนี้จะกำหนดให้อยู่ในหน่วยของจำนวนจุดสุ่มต่อหนึ่งหน่วยเวลา ยกตัวอย่าง เช่น ในระบบโทรศัพท์อัตราการสุ่มดังกล่าวจะมีค่าเป็น 8000 ครั้งต่อวินาที หรือ 8 KHz หรืออาจกล่าวได้ว่าคาบของการสุ่ม (ช่วงเวลาระหว่างการสุ่มแต่ละครั้ง) มีค่าเป็น 1/8000 หรือ 125 ไมโครวินาที สำหรับวิธีในการคำนวณหาค่าอัตราการสุ่มของระบบโทรศัพท์โดยเป็นไปตามข้อกำหนดในทฤษฎีของการสุ่มสัญญาณ ซึ่งถูกกำหนดขึ้นโดยแชนนอน นักคณิตศาสตร์ชาวสหรัฐอเมริกา กล่าวว่าอัตราการสุ่มจะต้องมีความถี่ไม่น้อยไปกว่า 2 เท่า ของความถี่สูงสุดของสัญญาณแอนาลอกที่จะทำการส่งนั้นจึงจะสามารถสร้างสัญญาณต้นฉบับกลับคืนจากสัญญาณสุ่มได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 สัญญาณที่ผ่านการสุ่มเทียบกับสัญญาณต้นฉบับ

2.6 ทฤษฎีและหลักการของ FPGAs

FPGAs จัดเป็นวงจรรวมเฉพาะกิจชนิดหนึ่งที่สามารถโปรแกรมเป็นวงจรดิจิตอลใดๆ ก็ได้ เช่นเดียวกับ EPLD ต่างกันที่ EPLD โปรแกรมลงบนอีพรอม (EPROM) ภายใน และสามารถโปรแกรมใหม่ได้หลังจากนำไปลบด้วยแสงอุลตราไวโอเลต แต่สำหรับ EPGAs จะโปรแกรมลงบนสแตติกแรม (Static RAM) ภายในด้วยข้อมูลที่อยู่ภายนอก และสามารถโปรแกรมใหม่ได้โดยการรีเซตด้วยสัญญาณไฟฟ้านอกจากนั้น FPGAs ยังประหยัดพลังงานและมีความจุวงจรที่มีขนาดใหญ่ (จำนวนเกตมากๆ) ได้อีกด้วย

2.6.1 โครงสร้างภายใน

วงจรรวมชนิดนี้ผลิตโดยบริษัทไซลิงค์ซึ่งเป็นบริษัทที่ร่วมกันทำการค้นคว้ากับ บริษัท เอ็มเอ็มไอ (MMI) สร้างเป็นอะเรย์ที่ประกอบด้วยเกตจำนวน 600-25000 เกต ดังแสดงในตารางที่ 2.2 การที่ต้องบอกขนาดของวงจรรวมเป็นจำนวนเกตเพราะจะได้รู้ว่าขนาดของ วงจรที่ได้ออกแบบไว้สามารถโปรแกรมลงบนวงจรรวม FPGAs ได้หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 การนับเกตของเกตพื้นฐาน

Gate	Equivalent gate count	Gate	Equivalent gate count
INV	1	RS Latch	3
NAIINR2	1	D Latch	4
NAIINR3	2	D Latch with CLR	5
NAIINR4	2	D Lath with PRE	5
NAIINR6	5	D Lath with PRE/CLR	6
NAIINR8	6	DEF	6
NAIINR9	7	DEF with CLR	7
NAIINR12	8	DEF with PRE	7
NAIINR16	11	DEF with PRE/CLR	8
BUFF	2	JKFF	9
ANIIOR2	2	JKFF with CLR	12
ANIIOR2	2	JKFF with CLR/PRE	13
ANIIOR2	3	TFF with CLR	8
XOR2	3	TEF with PRE	8
XNOR2	3	TEF with PRE/CLR	9

FPGAs มีโครงสร้างภายในใกล้เคียงกับสถาปัตยกรรมของเกตอะเรย์มาก สามารถโปรแกรมและลบองค์ประกอบ (Configuration) สแตติกแรม ภายในได้โดยใช้กระแส ไฟฟ้า ซึ่งทำการโปรแกรมได้โดยดึงข้อมูลฐานสิบหกมาจากภายนอก เช่น Parallel EPROM หรือ Serial PROM ต่างกับ EPLD, PAL ที่มีอีพรอมอยู่ในตัว ภายใน FPGAs จะจัดเรียงเป็นลอจิกเซลล้อมรอบภายนอกด้วยอินพุตเอาต์พุตเซล FPGAs ตัวแรกที่ผลิตโดยไซลิงค์คือ XC2064 (2000 Family) ประกอบด้วยเซลเรียงกันเป็นเมตริกซ์ (Matrix) เป็นจำนวน 64 เซล หลังจากนั้นผลิตตระกูล 3000 และ 4000 มีโครงสร้างซับซ้อนขึ้น สามารถบรรจุจำนวนเกตได้สูงและดีขึ้นแต่ละเซลเรียกว่าCLB (Configuration Logic Block)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 คุณสมบัติของ FPGAs ตระกูลต่างๆ

FPGAs	Appr. Gate count	Max I/O	Flip-Flop	Ram bit	Available CLBs
XC2064	1,000	58	122	0	64
XC2018	1,500	74	174	0	100
XC3020/3120	1,800	64	256	0	64
XC3030/3130	2,700	80	360	0	100
XC3042/3142	3,700	96	480	0	144
XC3064/3164	5,500	120	688	0	224
XC3090/3190	7,500	144	928	0	320
XC3195	9,000	176	1,320	0	484
XC4002A	2,000	64	256	2,048	64
XC4003/4003A	3,000	80	360	3,200	100
XC4003H	3,000	160	200	3,200	100
XC4004A	4,000	960	480	4,608	144
XC4005/4005A	5,000	122	616	6,272	196
XC4005H	5,000	192	392	6,272	196
XC4006	6,000	128	768	8,192	256
XC4008	8,000	144	936	103,687	324
XC4010	10,000	160	1,120	12,800	400
XC4010D	10,000	160	1,120	0	400
XC4013	13,000	192	1,536	18,432	576
XC4025	25,000	256	2,560	32,768	1,024

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 ส่วนที่เป็นองค์ประกอบของลอจิก (Configurable Logic Block)

CLB จะจัดเรียงกันเป็นแบบเมตริกซ์แบบอะเรย์ขนาด M*N การออกแบบนั้นสามารถทำได้โดยการจัดวาง CLB ให้ต่อถึงกัน เราสามารถจัด CLB ให้เชื่อมต่อถึงกันได้โดยการทำมือหรือโปรแกรมที่สนับสนุน FPGAs ทำให้โดยอัตโนมัติโดยวิธีของมันเอง สำหรับแฟ้มที่ได้จากโปรแกรมเหล่านี้เราเรียกว่า แฟ้มโครงร่าง (Configuration file) ซึ่งจะบรรจุโครงร่างภายในของ CLB ตามความเหมาะสม ในอีกด้านหนึ่งแฟ้มโครงร่างนั้นจะเป็นแฟ้มกระแสข้อมูล (Bitstrem) ซึ่งสามารถใช้โปรแกรมหน่วยความจำภายในของ FPGAs (Internal FPGAs configuration memory) ได้ สำหรับรูปแสดง CLB ของ FPGAs ตระกูล 4000 แสดงดังรูปที่ 2.11

2.6.3 ส่วนอินพุตและเอาต์พุต (I/O Block)

รอบๆ นอกของ FPGAs จะประกอบด้วย IOBs ประมาณ 64 ถึง 144 ตัวซึ่งขึ้นอยู่กับตระกูลของ FPGAs ซึ่ง IOBs จะเป็นตัวเชื่อมต่อระหว่างภายในกับภายนอกของวงจรลอจิกของ FPGAs ลักษณะของ IOBs จะมีลักษณะ 2 ทิศทาง สามารถโปรแกรมให้เป็นอินพุตหรือเอาต์พุตก็ได้ สำหรับตารางที่ 2.2 แสดง IOBs ของ FPGAs ตระกูล 4000

2.6.4 รายละเอียดการใช้งาน

FPGAs สามารถทำงานได้หลายรูปแบบ (Mode) โดยกำหนดได้ที่ขาสัญญาณ M0 M1 และ M2 ดังแสดงในตารางที่ 2.3 ในแบบมาสเตอร์พาราแรล (Master Parallel) รับโปรแกรมคอนฟิก (Config) ทีละ 1 ไบต์ (Byte) จากหน่วยความจำภายนอกที่เป็นแบบขนานโดยสามารถรับโปรแกรมคอนฟิก จากตำแหน่ง (Address) ต่ำหรือสูงก่อนก็ได้ แบบเพอริเฟอรัล (Peripheral) รับโปรแกรมคอนฟิกทีละ 1 ไบต์จากไมโครโปรเซสเซอร์ โดยสามารถโต้ตอบกันได้ว่าพร้อมหรือไม่พร้อมที่จะรับข้อมูลต่อไป แบบสเลฟ (Slave) รับโปรแกรมคอนฟิก ทีละ1 บิตจากไมโครโปรเซสเซอร์ตามสัญญาณอินพุต CCLK ส่วนในแบบมาสเตอร์ซีเรียล (Master Serial) รับโปรแกรมคอนฟิกทีละ 1 บิต จากหน่วยความจำจากภายนอกที่เป็นแบบอนุกรม

จากความต้องการสร้างให้ใช้กระแสไฟฟ้าต่ำ ในแบบการต่อใช้งานทั้ง 5 แบบจึงมีเพียง 2 แบบเท่านั้นที่เหมาะสม คือ แบบมาสเตอร์ซีเรียล (Master Serial Mode) และแบบสเลฟซีเรียล (Slave Serial Mode) ส่วนในแบบมาสเตอร์พาราแรลต้องใช้อีพรอม 27 Cxxx ซึ่งกินกระแสมากกว่าพรอม XC17xxx เหมาะในการทดสอบต้นแบบก่อนเมื่อวงจรต้นแบบทำงานได้ถูกต้องแล้วจึงทำการอัดโปรแกรมลงพรอมอีกทีหนึ่งเพราะว่าในแบบพาราแรลอีพรอมสามารถโปรแกรมได้ใหม่ต่างกับอีพรอมที่โปรแกรมได้ครั้งเดียว

การใช้งาน FPGAs ในแบบสเลฟซีเรียล เมื่อเริ่มจ่ายไฟเข้าตัว FPGAs จะทำการเคลียร์ (Clear) หน่วยความจำที่ใช้ในคอนฟิก (Configuration Memory) ตรวจสอบแบบกระทำการคอนฟิก

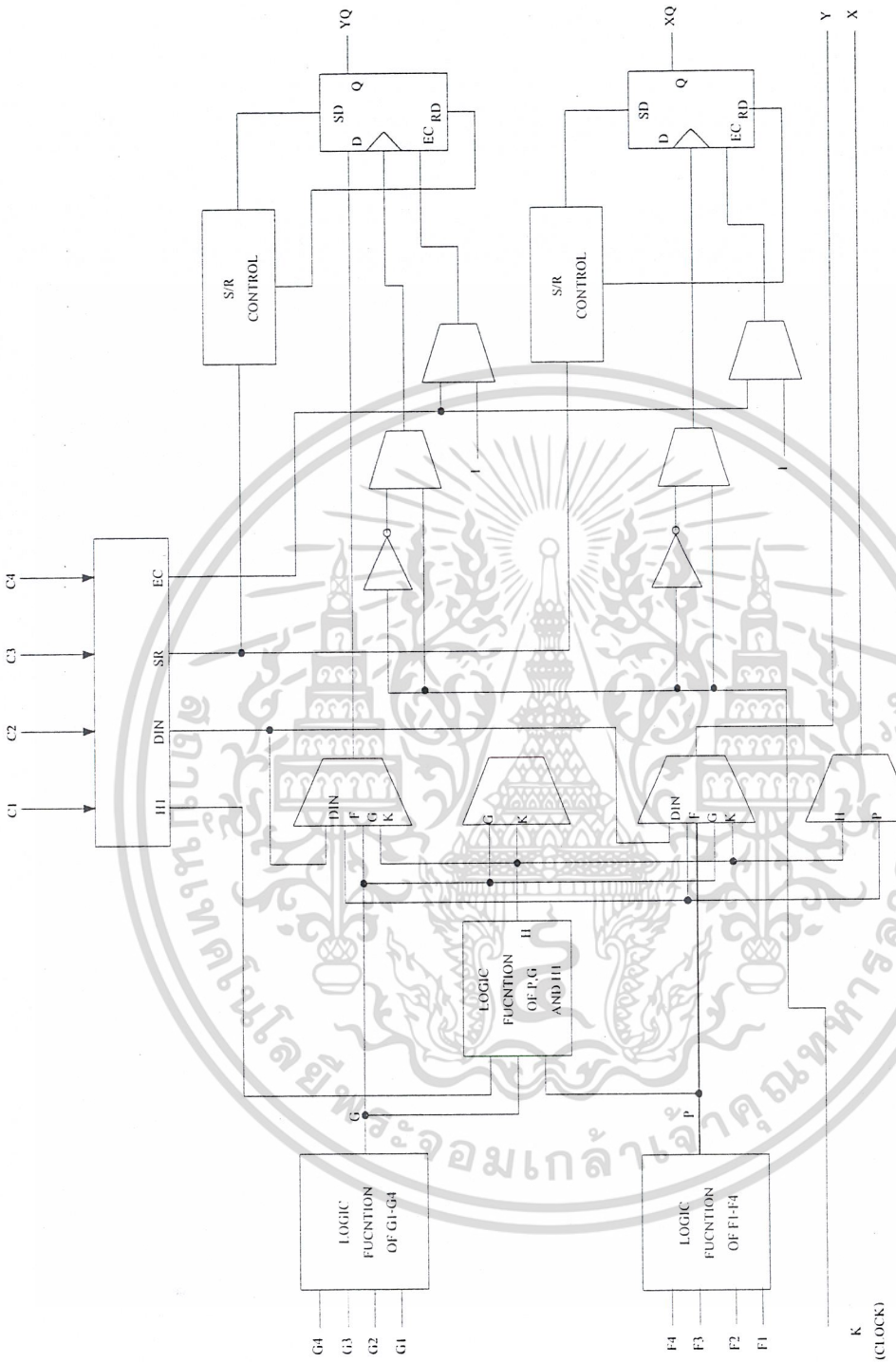
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าแบบใดในตารางที่ 2.3 (เป็นแบบอนุกรมหรือขนาน) หลังจากนั้นจะเริ่มทำการโปรแกรมคอนฟิก สัญญาณ Done/Program เป็น “0” (อยู่ในระหว่างโปรแกรม) และเมื่อข้อมูลในการคอนฟิกที่รับมาจากภายนอกเต็มหน่วยความจำที่ใช้ในการคอนฟิกและความยาวของข้อมูลตรงกับที่ส่วนหัวคอนฟิก สัญญาณ Done/Program เป็น “1” (โปรแกรมคอนฟิกเสร็จสิ้น) ดูรูป 2.13 ประกอบ

ตารางที่ 2.3 รูปแบบต่างๆ ของการกำหนดองค์ประกอบในการทำงาน

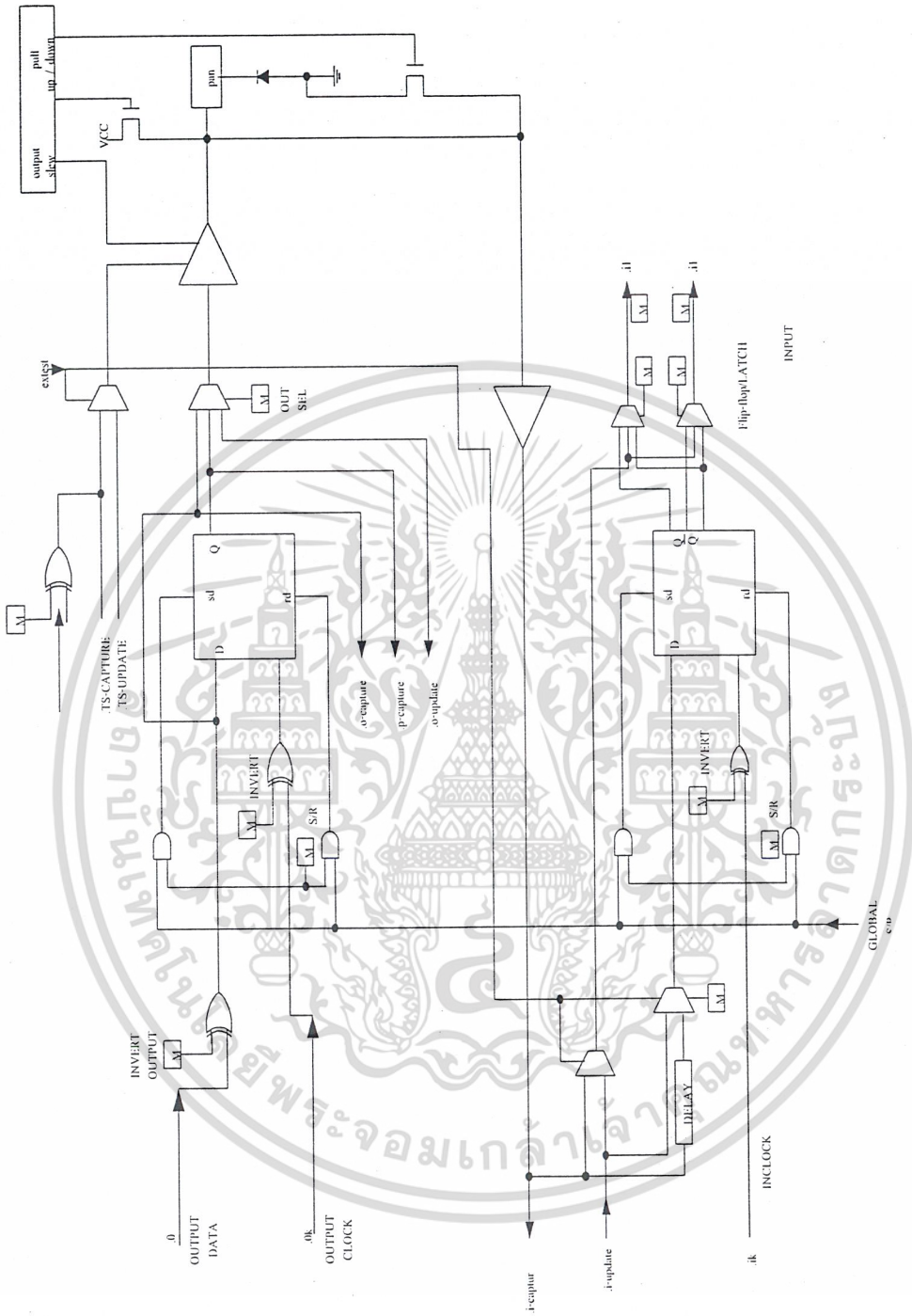
M0	M1	M2	CLK	MODE	DATA
0	0	0	Output	Master Serail	Bit Serial
0	0	1	Output	Master Parallel	Bit Wide Add.=0000 up
0	1	0	--------	Reseved	--------
0	1	1	Output	Master Parallel	Bit Wide Add.=FFFF down
1	0	0	--------	Reseved	--------
1	0	1	Output	Peripheral	Byte Wide
1	1	0	--------	Reseved	--------
1	1	1	Input	Slave Serail	Bit Serial

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



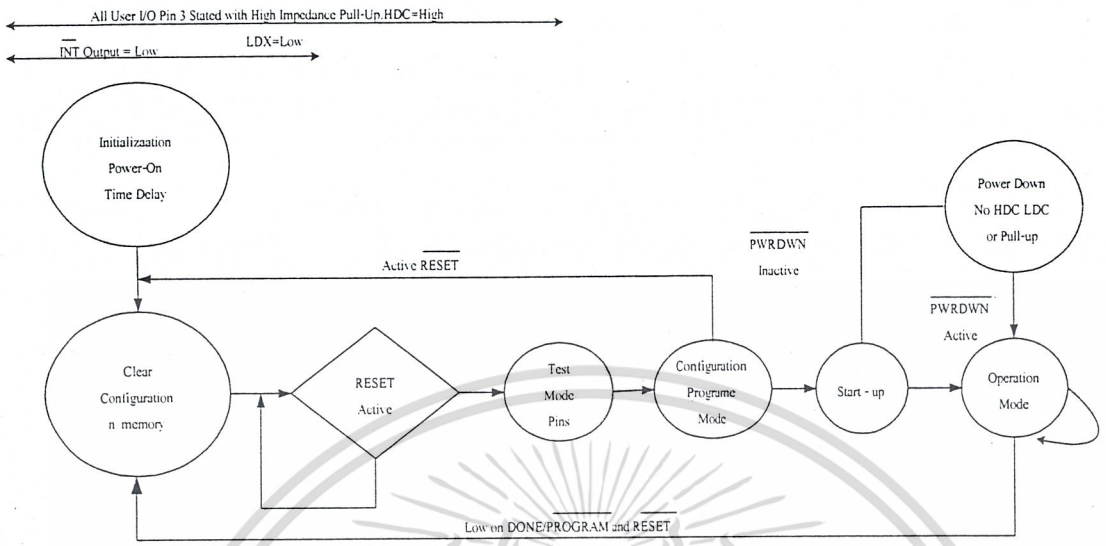
รูปที่ 2.11 แผนผัง CLB ของตระกูล 4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แผนผัง IOB ของตระกูล 4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

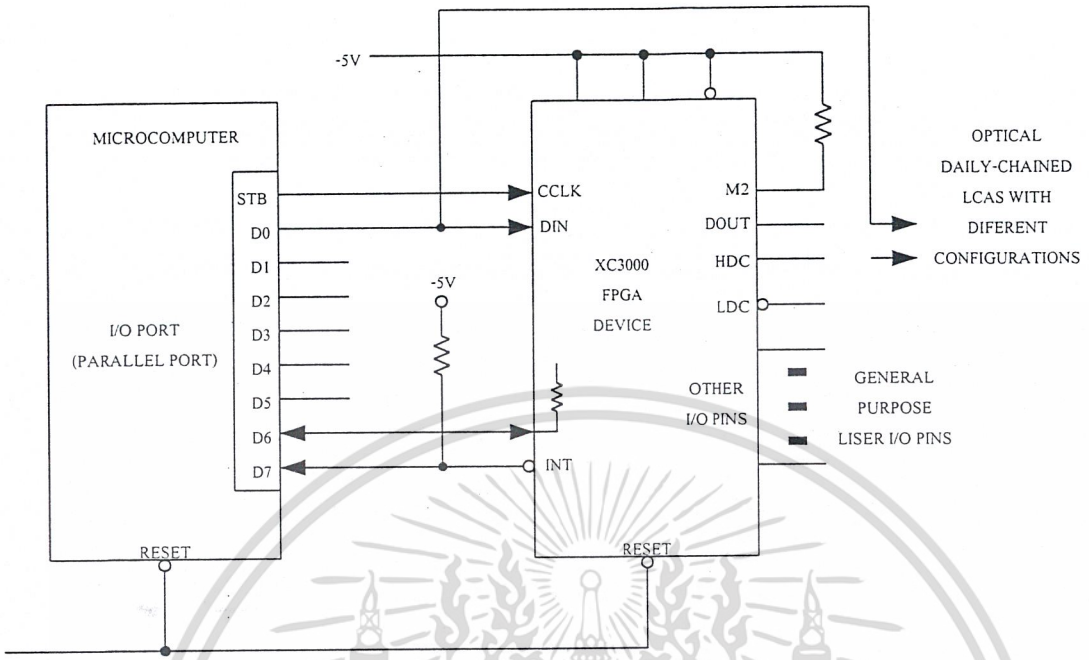


รูปที่ 2.13 แผนผังลำดับในการคอนฟิกเมื่อป้อนแหล่งจ่ายไฟเข้าในไอซีและการโปรแกรมใหม่

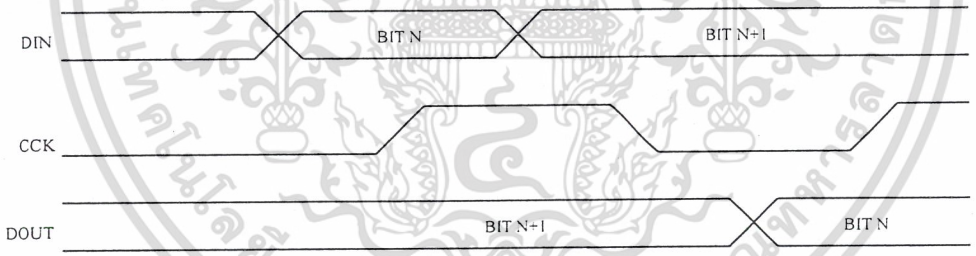
1) การใช้งานแบบสเลฟซีเรียล (Slave Serial)

การต่อใช้งานแบบนี้เหมาะสมกับวงจรที่ออกแบบมาเพื่อทำงานร่วมกับไมโครคอมพิวเตอร์อยู่แล้วทั้งนี้เพราะ FPGAs ได้ใช้ความสามารถของไมโครคอมพิวเตอร์ในการเก็บและส่งข้อมูลคอนฟิกให้ เพียงแต่ต้องเขียนโปรแกรมเพื่อส่งโปรแกรมคอนฟิกให้เพิ่ม ลักษณะการต่อในแบบนี้เป็นดังรูปที่ 2.14 ซึ่งในแบบนี้ไมโครคอมพิวเตอร์จะสร้างสัญญาณเพื่อการทำคอนฟิกให้กับอุปกรณ์ FPGAs ทำได้โดยต่อสัญญาณ Strobe เข้ากับขา CCLK และพอร์ต DO เข้ากับขา DIN สร้างสัญญาณนาฬิกาป้อนเข้าที่ขา CCLK และป้อนโปรแกรมคอนฟิกแบบอนุกรมเข้าที่ขา DIN ดังแผนภูมิใน รูปที่ 2.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 การต่อใช้งานในแบบสเลฟซีเรียล



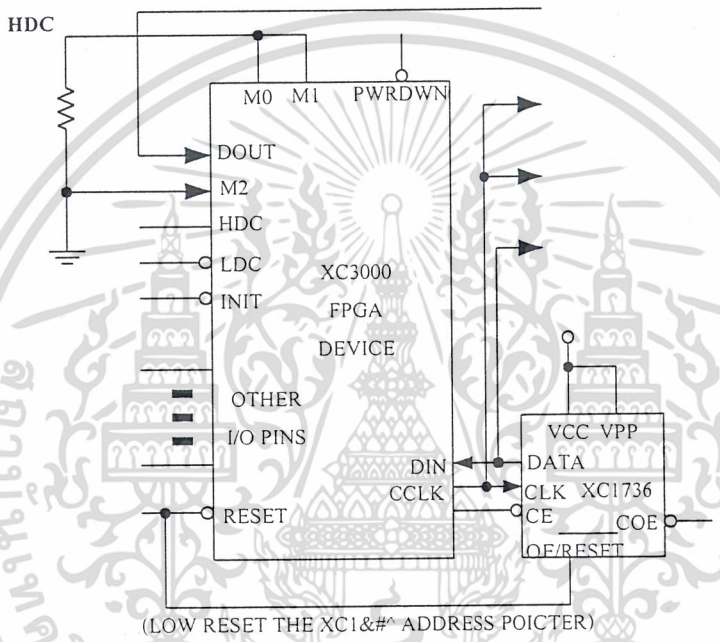
รูปที่ 2.15 แผนภูมิเวลาการป้อนข้อมูลโปรแกรมคอนฟิกในแบบสเลฟซีเรียล

2) การต่อใช้งานแบบมาสเตอร์ซีเรียล (Master Serial)

การต่อใช้งานในแบบนี้ ส่วนที่เก็บโปรแกรมคอนฟิกจะต่างจากแบบแรกคือใช้พรอม XC17xxx เป็นตัวเก็บโปรแกรม ทำให้ไม่ต้องเสียเวลาเขียนโปรแกรมเพื่อทำการคอนฟิก ซึ่งวิธีการอัดโปรแกรมคอนฟิกลงพรอมทำตามขั้นตอนดังนี้คือ เมกบิต (Make Bits) สร้างแฟ้ม BIT จากวงจรที่ออกแบบและใช้โปรแกรม Make PROM สร้างแฟ้ม Hex แล้วทำการอัดโปรแกรมลงพรอม ด้วยอุปกรณ์อัดพรอมที่มาพร้อมกับตัวโปรแกรมของไซลิงค์ (Xilinx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบนี้พรอม XC17xxx จะส่งสัญญาณเพื่อทำการคอนฟิกให้กับอุปกรณ์ FPGAs ดังแสดงในรูปที่ 2.16 DIN เป็นขารับข้อมูลที่ใช้ในการคอนฟิกแบบอนุกรม ส่วนอุปกรณ์ FPGAs จะกำเนิดสัญญาณ CCLK ให้กับพรอมเพื่อเป็นจังหวะในการอ่านข้อมูลโปรแกรมคอนฟิกมาไว้ในสแตติกแรม

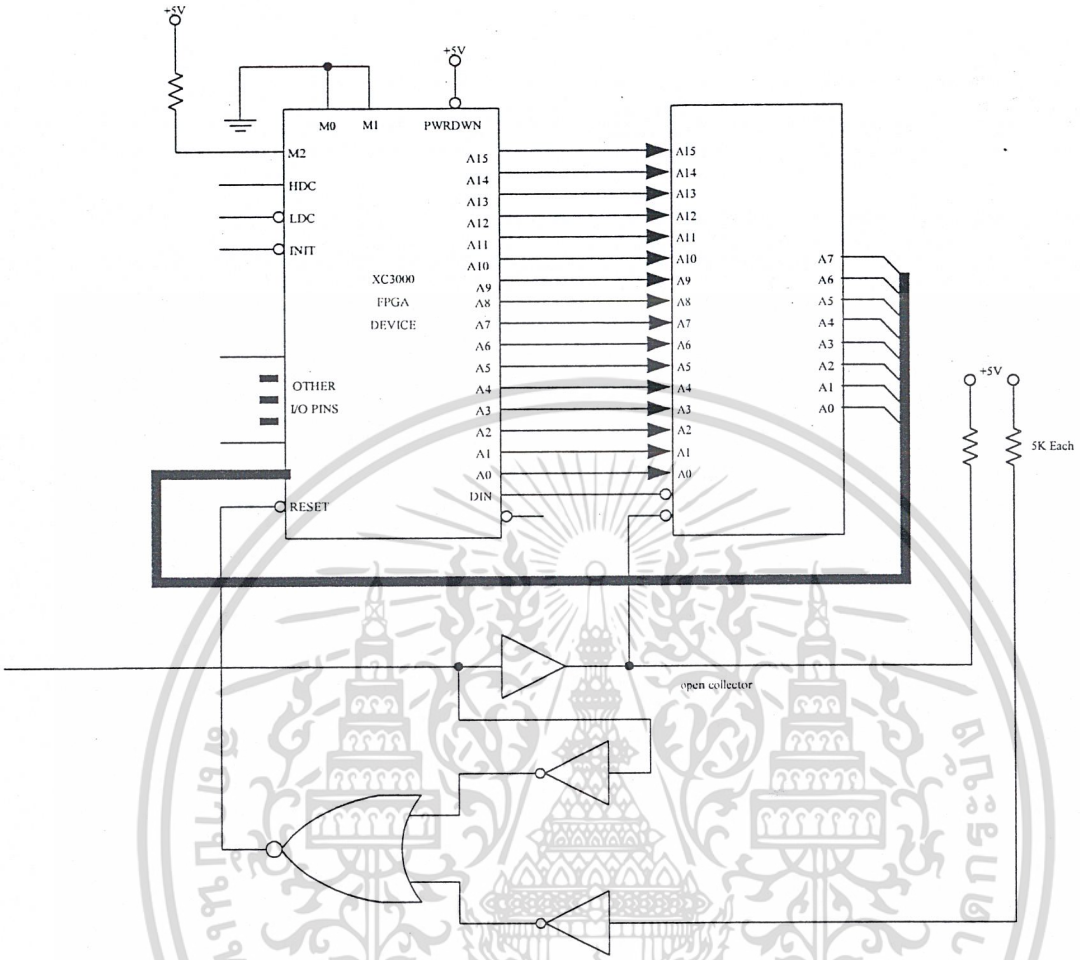


รูปที่ 2.16 การต่อใช้งานในแบบมาสเตอร์ซีเรียล

3) การใช้งานในแบบมาสเตอร์พาราแรล (Master Parallel)

การต่อใช้งานในแบบนี้ ส่วนที่เก็บโปรแกรมคอนฟิกจะต่างจากแบบมาสเตอร์ซีเรียลคือใช้พรอม 27 xzx เป็นตัวโปรแกรม ทำให้ไม่ต้องเสียเวลาเขียนโปรแกรมเพื่อทำการคอนฟิก ซึ่งวิธีการอัดโปรแกรมคอนฟิกลงพรอม ทำได้โดยใช้โปรแกรม MakePROM ใน XACT สร้างแฟ้ม Hex MCS แล้วทำการอัดโปรแกรมลงอีพรอมด้วยอุปกรณ์อัดอีพรอมที่มีใช้ทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 การต่อใช้งานในแบบมาสเตอร์พาราแรล

แบบนี้อีพรอม 27 xxx จะส่งสัญญาณทีละไบต์เพื่อทำการคอนฟิกให้กับอุปกรณ์ FPGAs ดังแสดงในรูปที่ 2.17 D0-D7 เป็นขารับข้อมูลที่ใช้ในการคอนฟิกแบบขนานนี้ A0-A15 เป็นตำแหน่งที่ FPGAs สร้างให้กับอีพรอมเพื่ออ่านข้อมูลจากหน่วยความจำ (โปรแกรมคอนฟิก) มาเก็บไว้ในสแตติกแรม ตำแหน่งทั้ง 16 เส้นไม่จำเป็นต้องต่อครบก็ได้ขึ้นอยู่กับขนาดหน่วยความจำอีพรอมที่ใช้ และสามารถกำหนดให้นับขึ้นหรือลงก็ได้

4) ข้อควรระวังในการใช้ FPGAs

สิ่งแรกที่สำคัญคือ ไอซีกลัวความร้อนเป็นที่สุด การบัดกรีโดยหัวแร้งกำลังสูง หรือ บัดกรีโดยจี้หัวแร้งที่ขาไอซีนานๆ จะทำให้ไอซีเสียหายได้ง่าย ระยะเวลาในการบัดกรีหนึ่งจุดไม่ควรเกิน 5-10 วินาที ควรใช้ซ็อกเกต (Socket) ไอซีในการประกอบวงจรลงแผ่นลายวงจรพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การป้องกันแหล่งจ่ายไฟให้ไอซีไม่ควรผิดขั้ว ถ้าสลับขั้วบวกหรือลบกันจะทำให้ไอซีเสียได้ นอกจากนั้นแรงดันของแหล่งจ่ายไฟต้องอยู่ในช่วงที่โรงงานกำหนดมา สำหรับ FPGAs แรงดันที่ไอซีทำงานอยู่ในช่วงการทำงานได้ Vcc = 4.75-5.25 โวลต์ และแรงดันที่ทนได้อยู่ในช่วง -0.5-7 โวลต์ ดังนั้นก่อนป้อนแรงดันควรตรวจสอบให้แน่ใจก่อน

การต่อวงจรผิดที่มีผลให้ไอซีเสียหายทันทีได้แก่ การต่อเอาต์พุตของไอซีไปเข้า ขั้วบวกหรือลบของแหล่งจ่ายไฟโดยตรง ไอซีทีทีแอล (TTL) ในขณะที่ CMOS ยังพอทนได้บ้างเพราะมีการจำกัดกระแสขาออกอยู่ภายในตัว ดังนั้นก่อนป้อนแรงดันเข้าไอซีควรตรวจ ดูบริเวณเอาต์พุตของไอซีด้วย

อีกประการหนึ่งที่ไอซีประเภท CMOS กลัวมากที่สุด คือ ไฟฟ้าสถิตย์ ขาที่เป็นอัตราได้ง่ายที่สุดคือขาอินพุต ความต้านทานขาเข้าของขาอินพุตของไอซีประเภทนี้มักจะสูงมากจนมีคุณสมบัติเหมือนกันกับเป็นฉนวนที่ขาอินพุตนี้อาจรับไฟฟ้าสถิตย์จากที่ใดมาเก็บไว้ เมื่อสะสมไว้มากๆ ไม่มีทางระบายออกก็จะทำให้วงจรบริเวณอินพุตเสียหายได้ ในปัจจุบันเทคโนโลยีการผลิตไอซีเจริญรุดหน้าไปมาก ทางด้านอินพุตของไอซี CMOS จะมีวงจรป้องกันไว้ภายในอย่างไรก็ตามเพื่อป้องกันเหตุการณ์ดังกล่าวควรเก็บรักษาไอซีที่ถูกต้องและควรเก็บไว้ในที่ปลอดภัยจากไฟฟ้าสถิตย์โดยเก็บไว้บนแผ่นเยื่อสังเคราะห์แบบนำกระแสได้หรือห่อหุ้มด้วยกระดาษอะลูมิเนียม

ไอซีประเภท CMOS มีความต้านทานขาเข้าสูงมาก กระแสจึงไม่ไหลจากแหล่งจ่ายไฟเข้าไปที่ขาอินพุตของ CMOS นั้นถึงแม้จะต่อความต้านทานคั่นกลางอยู่ แรงดัน 5 โวลต์จากแหล่งจ่ายไฟก็ยังป้อนเข้าขาอินพุต 5 โวลต์ โดยที่ไม่มีแรงดันตกคร่อมบนตัวต้านทานเลย ถ้ามีกระแสไหลแสดงว่าไอซีตัวนั้นเสีย เอาต์พุตของไอซี CMOS อยู่ในสภาวะ 1 จะสามารถขับโหลดได้หลายมิลลิแอมป์ และถ้าเอาต์พุตอยู่ในสภาวะ 0 ก็สามารถจะรับกระแสไหลเข้าได้หลายมิลลิแอมป์

2.7 การใช้ซอฟต์แวร์ของบริษัทวิวลอจิกและของบริษัทไซลิงค์

2.7.1 ขั้นตอนในการออกแบบและจำลองการทำงานโดยใช้ซอฟต์แวร์ของบริษัทวิวลอจิก

เริ่มต้นกำหนดไลบรารี่ที่ต้องการจะใช้ให้เป็น FPGAs ตระกูล XC4000 และนำโปรแกรม View Draw เขียนวงจร จากนั้นแปลงแฟ้มที่เป็นผังวงจรเป็น Wirlist เพื่อนำไปจำลองการทำงานโดยโปรแกรม View Sim และใช้ View Trace เพื่อดูรูปร่างของสัญญาณที่ขาอินพุตและเอาต์พุตโดยรับสัญญาณอินพุตที่ผู้ออกแบบวงจรได้สร้างไว้ เพื่อทดสอบวงจรให้ได้ตรงกับความต้องการ ซึ่งมีหลายวิธีโดยอาจจะป้อนทีละคำสั่งหรือเขียนเป็นคำสั่งต่อๆ กันเก็บไว้ในรูปของแฟ้มข้อความนามสกุล .CMD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 ขั้นตอนในการทดสอบวงจรโดยใช้ซอฟต์แวร์ของบริษัท ไซลิงค์

สร้างแม็ปไฟล์ (Map File) จากเน็ทลิส (Netlist) โดยใช้โปรแกรม MAP2LCA สร้างแฟ้ม .LCA ซึ่งสามารถแก้ไขได้โดยผู้ออกแบบหรือใช้โปรแกรม APR (Automatic Place and Poute) จัดการให้อัตโนมัติ ในขั้นตอนทั้งหมดที่กล่าวมาข้างต้นสามารถใช้โปรแกรม XMAKE จากนั้นแปลงแฟ้มจาก .LCA เป็น .BIT โดยโปรแกรม Make Bit จากนั้นแปลงเป็น Intel Hex File เพื่อใช้เป็นคอนฟิกให้กับ FPGAs ในกรณีที่ทดสอบวงจรในแบบมาสเตอร์พาราแรลหรือแปลงเป็นบิตสตรีม เพื่อใช้โปรแกรมลงบนพรอมตระกูล XC17xxx ซึ่งเป็นตัวโหลดโปรแกรมคอนฟิกให้กับอุปกรณ์ FPGAs ในกรณีที่ทดสอบวงจรในแบบมาสเตอร์ซีเรียล เพื่อจำลองการทำงานของวงจรสมบูรณ์มากยิ่งขึ้น หลังจากที่ได้แฟ้มที่มีโครงสร้างเป็น .LCA แล้วโดยใช้ซอฟต์แวร์แปลงกลับไปเป็นผัง วงจรเพื่อจำลองการทำงานใหม่อีกครั้ง ให้ผลการจำลองตรงกันกับครั้งแรกที่จำลองการทำงาน ซึ่งข้อดีของผังวงจรใหม่ที่สร้างขึ้นนี้เปรียบเสมือนการยุบวงจรให้นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

3.1 กล่าวนำ

ในการออกแบบและการสร้างวงจรการเข้ารหัสเสียงด้วยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs ได้แบ่งการออกแบบเป็นสองส่วน คือส่วนที่ 1 ฮาร์ดแวร์ (Hardware) ภายใน และ ส่วนที่ 2 ฮาร์ดแวร์ภายนอก

ส่วนที่ 1 ออกแบบระบบและวงจรการเข้ารหัสเสียงด้วยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานในการออกแบบส่วนนี้จะเป็นการสร้างวงจรลอจิกเกตต่างๆ ขึ้นภายในอุปกรณ์ FPGAs ซึ่งจำเป็นจะต้องออกแบบให้สอดคล้องกับอุปกรณ์ภายในของ FPGAs ที่มีอยู่แล้ว ดังนั้นส่วนประกอบของวงจรจึงใช้ไลบรารี่ (Library) FPGAs ของบริษัทไซลิงค์ในการออกแบบ

ส่วนที่ 2 ออกแบบระบบเพื่อให้วงจรเข้ารหัสเสียงภายในอุปกรณ์ FPGAs นำมาใช้งานได้จริงตามที่ออกแบบไว้ในการออกแบบส่วนนี้จะนำ FPGAs ที่ได้ออกแบบไว้แล้วนำมาทำการเชื่อมต่อภาครับเสียงและภาคขยายเสียงรวมทั้งตัวโค้ดบุ๊ค (Code book) ด้วย

3.2 หลักการออกแบบวงจร

ในการออกแบบฮาร์ดแวร์ภายใน FPGAs นี้ จะต้องทำการออกแบบและการสร้างโดยใช้ซอฟต์แวร์ของวิวลอจิก (Viewlogic) เขียนวงจรการเข้ารหัสเสียงด้วยโปรแกรม Viewdraw โดยการออกแบบเป็นภาคเข้ารหัสเสียงและถอดรหัสเสียง โดยภาคเข้ารหัสเสียงใช้ไลบรารี่ FPGAs ตระกูล XC 4000 ส่วนภาคถอดรหัสเสียงใช้ไลบรารี่ FPGAs ตระกูล XC 3000 และทำการจำลองการทำงานโดยใช้โปรแกรม Viewsim จนได้ผลการทำงานตามที่ต้องการ จากนั้นใช้ซอฟต์แวร์ของไซลิงค์ เพื่อโปรแกรมวงจรต่าง ๆ ที่ออกไว้ในวิวลอจิกลงในอุปกรณ์ FPGAs

การออกแบบและการจำลองการทำงานของวงจรต่างๆ โดยใช้โปรแกรมวิวลอจิกนั้นอันดับแรกจะใช้โปรแกรม Viewdraw เขียนวงจรภาคเข้ารหัสเสียงและภาคถอดรหัสเสียง จากนั้นทำการแปลงแฟ้มที่เป็นผังวงจรให้เป็น Wirelist โดยใช้โปรแกรม VSM เพื่อใช้สำหรับจำลองการทำงาน

3.3 ขั้นตอนการออกแบบวงจร

ในการออกแบบวงจรแบ่งเป็น 2 ส่วนคือ ส่วนของวงจรเข้ารหัสเสียง และส่วนของวงจรถอดรหัสเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1การออกแบบวงจรเข้ารหัสเสียง

การออกแบบวงจรเข้ารหัสเสียงสามารถทำได้ดังนี้ อันดับแรกนั้นคือทำการกำหนดแบบ วงจรเข้ารหัสเป็นผังการทำงานดังรูปที่ 3.1 หลังจากนั้นทำการออกแบบวงจรตามผังการทำงาน จากนั้นจึงสร้างวงจรตามที่ได้ออกแบบไว้ เมื่อออกแบบและสร้างเรียบร้อยแล้วจึงทำการตรวจสอบ วงจรที่ได้สร้างไว้

ลักษณะการทำงานโดยรวมของวงจรเข้ารหัสเสียงคือ จะนำสัญญาณเสียงแปลงให้เป็นสัญญาณดิจิตอลแล้วนำมาเปรียบเทียบกับข้อมูลในโค้ดบุ๊คซึ่งโค้ดบุ๊คจะเก็บสัญญาณเสียงระดับต่างๆ ไว้ในรูปแบบของสัญญาณดิจิตอล โดยในขบวนการเปรียบเทียบนี้จะใช้ค่าที่เปรียบเทียบที่มีค่าน้อยที่สุดแล้วจึงนำเฉพาะดัชนี (Index) ของโค้ดบุ๊คที่มีค่าใกล้เคียงที่สุดนั้นส่งออกไปเป็นเอาต์พุตซึ่งจะอธิบายลักษณะการทำงานดังนี้

สัญญาณเสียงซึ่งเป็นสัญญาณแอนะลอกจะถูกแปลงให้เป็นสัญญาณดิจิตอลโดยวงจร Analog to Digital Converter ซึ่งใช้ความถี่ในการสุ่มสัญญาณเท่ากับ 8 kHz และผ่านขบวนการเข้ารหัสจะได้เป็นสัญญาณดิจิตอลขนาด 8 บิต โดยในวงจรเข้ารหัสนี้จะทำการเก็บสัญญาณดิจิตอลขนาด 8 บิต ไว้จำนวน 5 ชุด หรือจำนวน 5 แซมเปิ้ล เพื่อใช้ในการนำมาเปรียบเทียบกับโค้ดบุ๊ค ข้อมูลที่อยู่ในโค้ดบุ๊คจะมีลักษณะเป็นสัญญาณดิจิตอลขนาด 8 บิต จำนวน 5 ชุดเช่นกัน แต่จะมีข้อมูลในลักษณะเช่นนี้อยู่จำนวน 64 เวกเตอร์

สัญญาณเสียงที่ได้จากการแปลงให้เป็นสัญญาณดิจิตอลจะถูกส่งเข้าไปยังวงจร Subtractor (SUB) ซึ่งเป็นวงจรลบข้อมูลขนาด 8 บิต โดยจะส่งเข้าไปลบครั้งละ 8 บิตเพื่อลบกับข้อมูลในโค้ดบุ๊คที่ส่งออกมาครั้งละ 8 บิตเช่นเดียวกัน ซึ่งในการนำเอาข้อมูลในโค้ดบุ๊คออกมานั้นจะใช้วงจร Modulus 5 (MOD5) เป็นวงจรควบคุม เนื่องจากข้อมูลในโค้ดบุ๊คมีลักษณะของสัญญาณดิจิตอล 8 บิต จำนวน 5 ชุด อยู่ทั้งหมด 64 เวกเตอร์ ดังนั้นสัญญาณเสียงที่นำมาเปรียบเทียบจึงต้องเปรียบเทียบให้ครบทั้ง 64 เวกเตอร์ โดยจะใช้วงจร Counter 64 (COUNT 64) เป็นวงจรควบคุมในการนับจำนวนเวกเตอร์ของโค้ดบุ๊ค

เอาต์พุตที่ได้จากวงจร Subtractor จะถูกส่งไปยังวงจร Absolute (ABS) เพื่อป้องกันการเกิดการผิดพลาดของผลลัพธ์ที่ได้จากการลบในกรณีที่เกิดการยืมขึ้น โดยจะทำให้ผลลัพธ์นั้นมีค่าของขนาดที่ถูกต้อง จากนั้นจะส่งเอาต์พุตที่ได้จากวงจร Absolute เข้าไปวงจร Adder (ADD) เพื่อทำการบวก ในครั้งเริ่มต้นการทำงานนี้ผลลัพธ์ที่ได้จากการลบครั้งแรกจะถูกนำมาบวกกับค่าที่เก็บไว้ในรีจิสเตอร์ 1 (REG 1) ซึ่งในรีจิสเตอร์ 1 จะกำหนดค่าเริ่มต้นให้น้อยที่สุดจากนั้นที่รีจิสเตอร์ 1 จะนำผลลัพธ์ที่ได้กลับมารอการบวกที่วงจร Adder โดยผ่านวงจร Protector พร้อมทั้งนำผลลัพธ์ไปเก็บไว้ที่รีจิสเตอร์ 2 เนื่องจากในแต่ละเวกเตอร์ของโค้ดบุ๊คจะมีข้อมูลขนาด 8 บิต จำนวน 5 ชุด ดังนั้นจึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องทำการบวกค่าให้ครบ 5 ครั้งซึ่งวงจร Protector จะทำหน้าที่ตรวจสอบจำนวนครั้งของการป้อนกลับมาบวกที่วงจร Adder เมื่อครบ 5 ครั้งแล้วก็จะส่งสัญญาณไปให้รีจิสเตอร์ 2 เพื่อส่งข้อมูลเข้าไปเปรียบเทียบที่วงจร Comparator

ในการเปรียบเทียบครั้งแรกจะต้องกำหนดค่าเริ่มต้นในรีจิสเตอร์ 3 ให้มีค่ามากที่สุดเพื่อใช้เปรียบเทียบกับผลลัพธ์ครั้งแรกที่ได้จากรีจิสเตอร์ 2 ในขณะเดียวกันเมื่อผลการเปรียบเทียบเสร็จสิ้นเอาต์พุตของรีจิสเตอร์ 2 จะส่งมาเก็บไว้ที่รีจิสเตอร์ 3 แทนเพื่อใช้ในการเปรียบเทียบในครั้งต่อไป ซึ่งในขบวนการเปรียบเทียบ นี้จะทำเป็นขั้นตอนเหมือนกันทั้ง 64 เวกเตอร์ของโค้ดบุ๊ค

ถ้าผลของการเปรียบเทียบที่ได้มีค่าน้อยกว่าวงจร Comparator จะส่งสัญญาณมาให้รีจิสเตอร์ 4 เพื่อนำข้อมูลไปเก็บไว้ที่วงจร Transfer ข้อมูลที่อยู่ในรีจิสเตอร์ 4 นี้ จะได้มาจากวงจร Counter 64 ซึ่งเป็นวงจรที่ใช้นับเวกเตอร์ของโค้ดบุ๊คเช่นเดียวกัน ดังนั้นการทำงานทั้งหมดของวงจรซิงโครนัส (Synchronous) ต่อเนื่องกัน

ในทุกๆ การเปรียบเทียบแต่ละครั้งที่มีผลลัพธ์น้อยกว่า ข้อมูลที่อยู่ในรีจิสเตอร์ 4 จะถูกโหลดมาเก็บไว้ที่วงจร Transfer ตลอด เมื่อทำการเปรียบเทียบค่าครบทั้ง 64 เวกเตอร์ของโค้ดบุ๊คแล้วข้อมูลสุดท้ายที่อยู่ในวงจร Transfer จะเป็นดัชนีของโค้ดบุ๊คที่มีข้อมูลเสียงใกล้เคียงกับสัญญาณเสียงทางอินพุตมากที่สุดจึงจะถูกส่งออกไปเป็นเอาต์พุตของวงจรเข้ารหัส

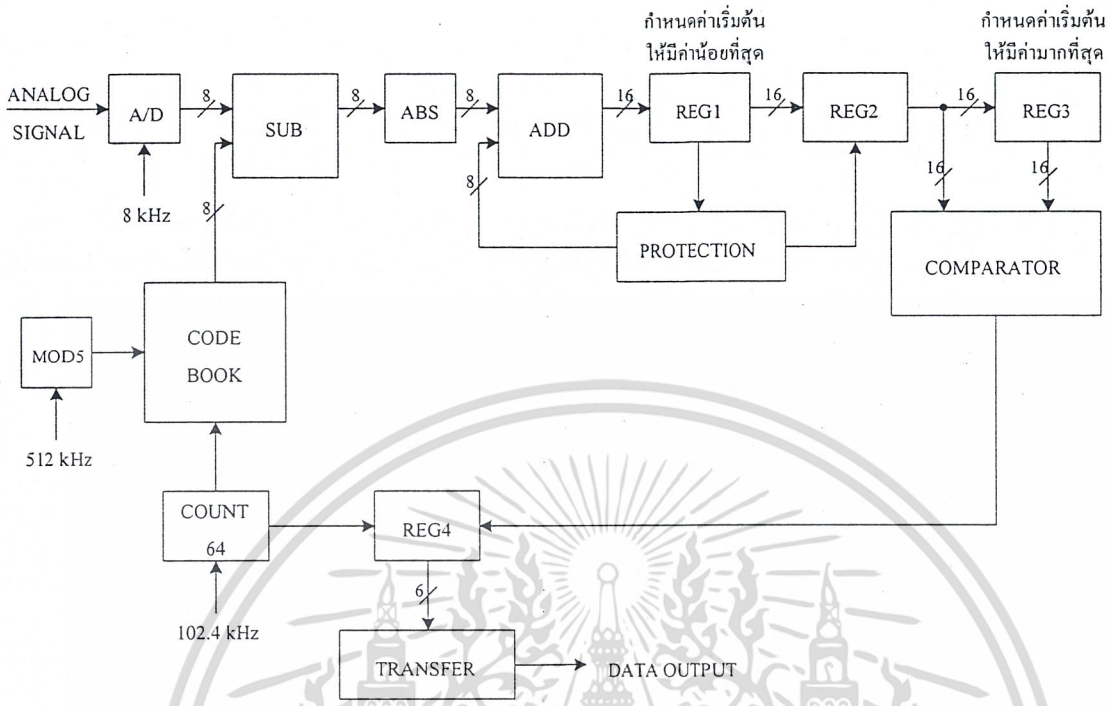
3.3.2 การออกแบบวงจรถอดรหัสเสียง

ลำดับขั้นตอนการออกแบบวงจรถอดรหัสเสียง คือการออกแบบแผนผังการทำงานตามรูปที่ 3.2 จากนั้นจึงสร้างวงจรตามที่ออกแบบไว้ แล้วทำการทดสอบ

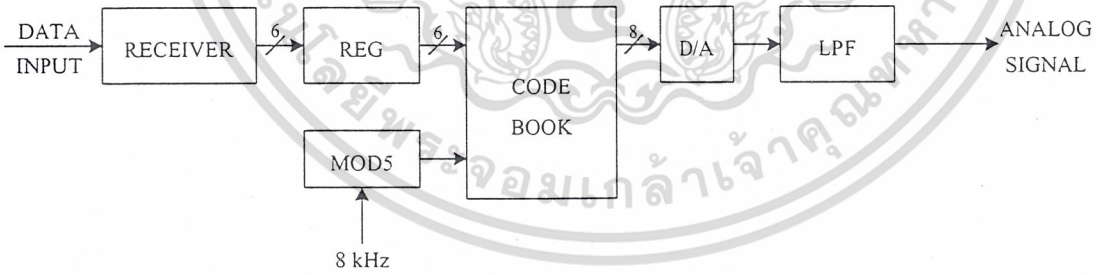
ในการทำงานของวงจรถอดรหัสจะใช้วงจร Receiver ในการรับข้อมูลที่ส่งมาจากวงจรเข้ารหัสเสียงซึ่งเป็นดัชนีของโค้ดบุ๊คนั่นเอง เมื่อรับข้อมูลครบแล้วจึงจะโหลดข้อมูลไปเก็บไว้ที่รีจิสเตอร์จากนั้นจึงนำข้มูลในรีจิสเตอร์นี้ไปเปิดโค้ดบุ๊คตามดัชนีที่รับได้ซึ่งโค้ดบุ๊คที่อยู่ในวงจรถอดรหัสนี้จะต้องเป็นโค้ดบุ๊คที่มีข้อมูลเหมือนกันกับโค้ดบุ๊คที่อยู่ในวงจรเข้ารหัสเสียง

จากนั้นจะใช้วงจร Modulus 5 ทำการดึงข้อมูลออกมาจากโค้ดบุ๊คครั้งละ 8 บิต เพื่อไปทำการแปลงสัญญาณให้อยู่ในรูปของสัญญาณแอนะลอก เมื่อผ่านวงจรกองความถี่ต่ำ (LPF) ก็จะทำให้ได้สัญญาณเสียงกลับคืนมาดังเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แผนผังการทำงานของวงจรเข้ารหัสเสียง

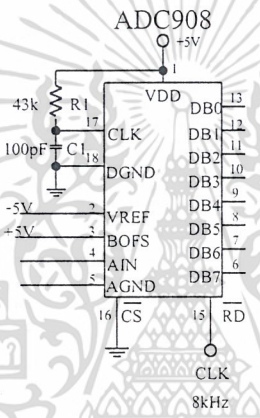


รูปที่ 3.2 แผนผังการทำงานของวงจรถอดรหัสเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล ในโครงงานนี้ใช้ไอซีเบอร์ ADC908 ซึ่งเป็นไอซีสำหรับแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลแบบ Successive Approximation และให้ผลการแปลงเป็นสัญญาณดิจิตอล 8 บิต โดยการต่อวงจรใช้งานแสดงดังรูปที่ 3.3 จะเห็นได้ว่า วงจรนั้นต่อ R1 และ C1 ไว้สำหรับสร้างสัญญาณนาฬิกาภายในตัวไอซี ซึ่งทำให้มีช่วงเวลาในการแปลงสัญญาณ (Conversion times)ได้ต่ำสุด 6 ไมโครวินาที ส่วนขา $\overline{RD}$ ใช้สำหรับการต่อสัญญาณนาฬิกาจากภายนอกซึ่งสามารถใช้ความถี่ได้สูงถึง 1.35 MHz



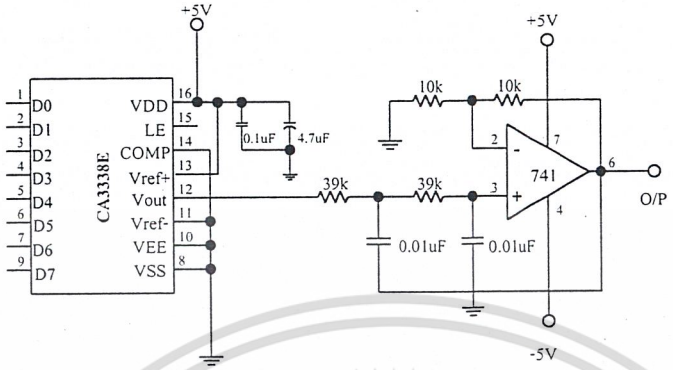
รูปที่ 3.3 วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

3.3.4 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกและวงจรกรองความถี่ต่ำ

วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกในโครงงานนี้ใช้ไอซีเบอร์ CA3338 ซึ่งอาศัยหลักการของ R-2R ในการแปลงสัญญาณสามารถใช้งานได้กับความถี่สูงถึง 50 MHz สำหรับการต่อวงจรใช้งานแสดงไว้ดังรูปที่ 3.4

สำหรับวงจรกรองความถี่ต่ำ (LPF) ใช้ไอซีออปแอมป์เบอร์ 741 เป็นวงจร Active Filter ซึ่งวงจรที่ออกแบบมาเป็นวงจรกรองความถี่ต่ำอันดับที่ 2 ที่มีความถี่คัตออฟที่ 4.08 kHz สำหรับใช้งานกับสัญญาณเสียงพูด

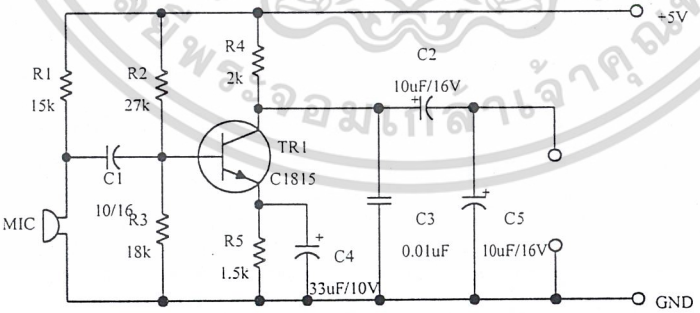
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก และวงจรกรองความถี่ต่ำ

3.3.5 วงจรปรีไมโครโฟน

วงจรปรีไมโครโฟนซึ่งทำหน้าที่ขยายสัญญาณเสียงที่ได้จากไมโครโฟนชนิดคอนเดนเซอร์ ให้มีความแรงของสัญญาณสูงขึ้นเพื่อส่งสัญญาณเสียงนั้นต่อไปที่วงจรภาคเข้ารหัสสัญญาณเสียง ซึ่งวงจรปรีไมโครโฟนแสดงดังรูปที่ 3.5 โดยวงจรใช้ทรานซิสเตอร์ชนิด NPN (ซิลิกอน) เบอร์ C1815 ซึ่งเป็นทรานซิสเตอร์ขยายสัญญาณเสียงทำหน้าที่เป็นตัวขยายสัญญาณ วงจรมีอินพุตอิมพีแดนซ์เท่ากับ 3.587 kΩ, เอาต์พุตอิมพีแดนซ์เท่ากับ 2kΩ, อัตราขยายแรงดันประมาณ 66 เท่า, และอัตราขยายกระแสประมาณ 127 เท่า

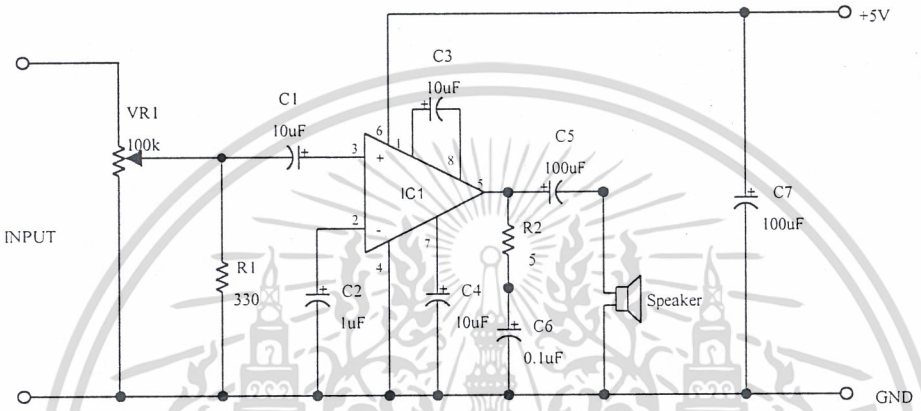


รูปที่ 3.5 วงจรปรีไมโครโฟน

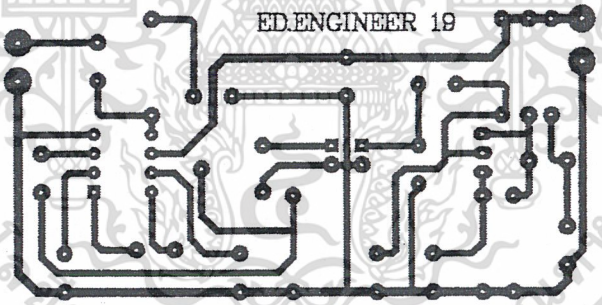
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.6 วงจรขยายสัญญาณเสียง

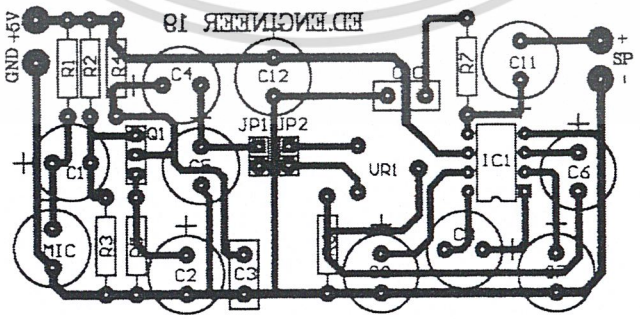
วงจรขยายสัญญาณเสียง (Power Amplifier) ใช้เพื่อขยายสัญญาณเสียงที่ได้จากวงจรภาคถอดรหัสสัญญาณเสียงให้มีความแรงพอที่จะขับลำโพงให้ดังได้ โดยวงจรแสดงดังรูปที่ 3.6 วงจรใช้ไอซีเบอร์ LM386 เพื่อความสะดวกในการสร้าง ไอซีเบอร์ LM336 นี้เป็นไอซีขยายสัญญาณ มีกำลังขยายสูงสุด 1 วัตต์ ส่วนลายวงจรพิมพ์แสดงดังรูปที่ 3.7



รูปที่ 3.6 วงจรขยายสัญญาณเสียง



รูปที่ 3.7 ลายวงจรพิมพ์วงจรปรีไมโครโฟนและวงจรขยายสัญญาณเสียง

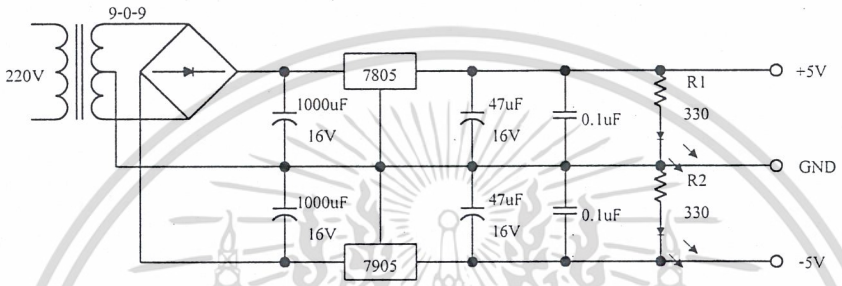


รูปที่ 3.8 การวางอุปกรณ์วงจรปรีไมโครโฟนและวงจรขยายสัญญาณเสียง

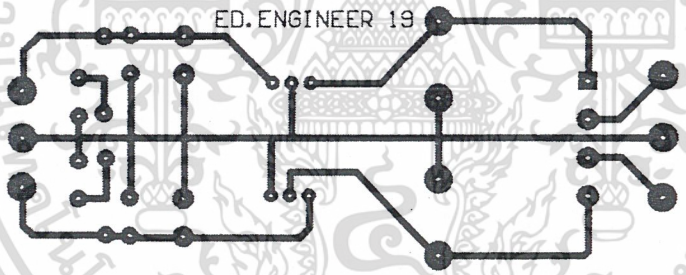
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.7 วงจรแหล่งจ่ายไฟ

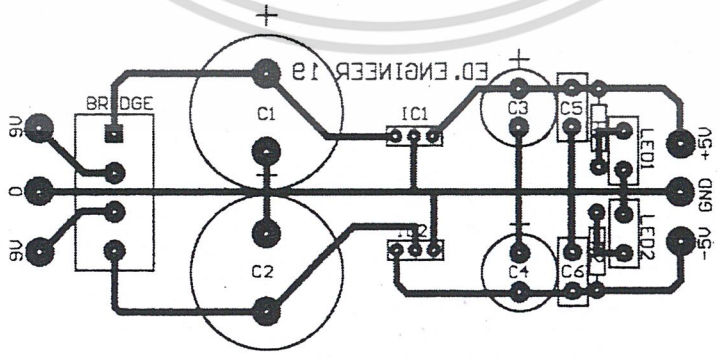
เนื่องจากวงจรทั้งหมดที่ออกแบบมาต้องการใช้แรงดันเลี้ยงวงจร ±5 โวลต์ ดังนั้นเราจึงต้องใช้วงจรแหล่งจ่ายไฟที่ให้แรงดันเป็น ±5 โวลต์ ซึ่งวงจรแหล่งจ่ายไฟแสดงดังรูปที่ 3.9 ทำงานโดยวงจรใช้ไอซีเรกกูเลเตอร์ เบอร์ 7805 และ 7905 เป็นตัวให้แรงดันที่เอาต์พุตของวงจรเป็น ±5 โวลต์ ส่วนลายวงจรพิมพ์แสดงดังรูปที่ 3.10



รูปที่ 3.9 วงจรแหล่งจ่ายไฟ



รูปที่ 3.10 ลายวงจรพิมพ์วงจรแหล่งจ่ายไฟ

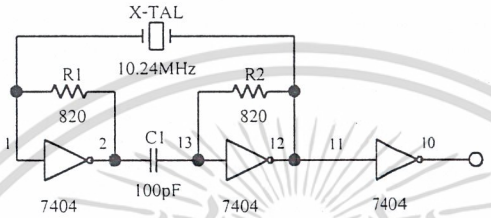


รูปที่ 3.11 การวางอุปกรณ์วงจรแหล่งจ่ายไฟ

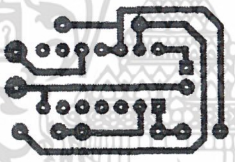
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.8 วงจรกำเนิดความถี่ (Oscillator)

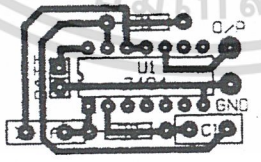
วงจรกำเนิดความถี่จะใช้สำหรับผลิตสัญญาณความถี่ 10.24 MHz เพื่อนำสัญญาณที่ได้ไปสร้างเป็นความถี่อื่นๆ ใน FPGAs เพื่อนำสัญญาณความถี่เหล่านั้นไปควบคุมจังหวะการทำงานของวงจรต่างที่ออกแบบไว้ใน FPGAs สำหรับวงจรกำเนิดความถี่แสดงไว้ดังรูปที่ 3.12



รูปที่ 3.12 วงจรกำเนิดความถี่



รูปที่ 3.13 ลายวงจรพิมพ์วงจรกำเนิดความถี่



รูปที่ 3.14 การวางอุปกรณ์วงจรกำเนิดความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

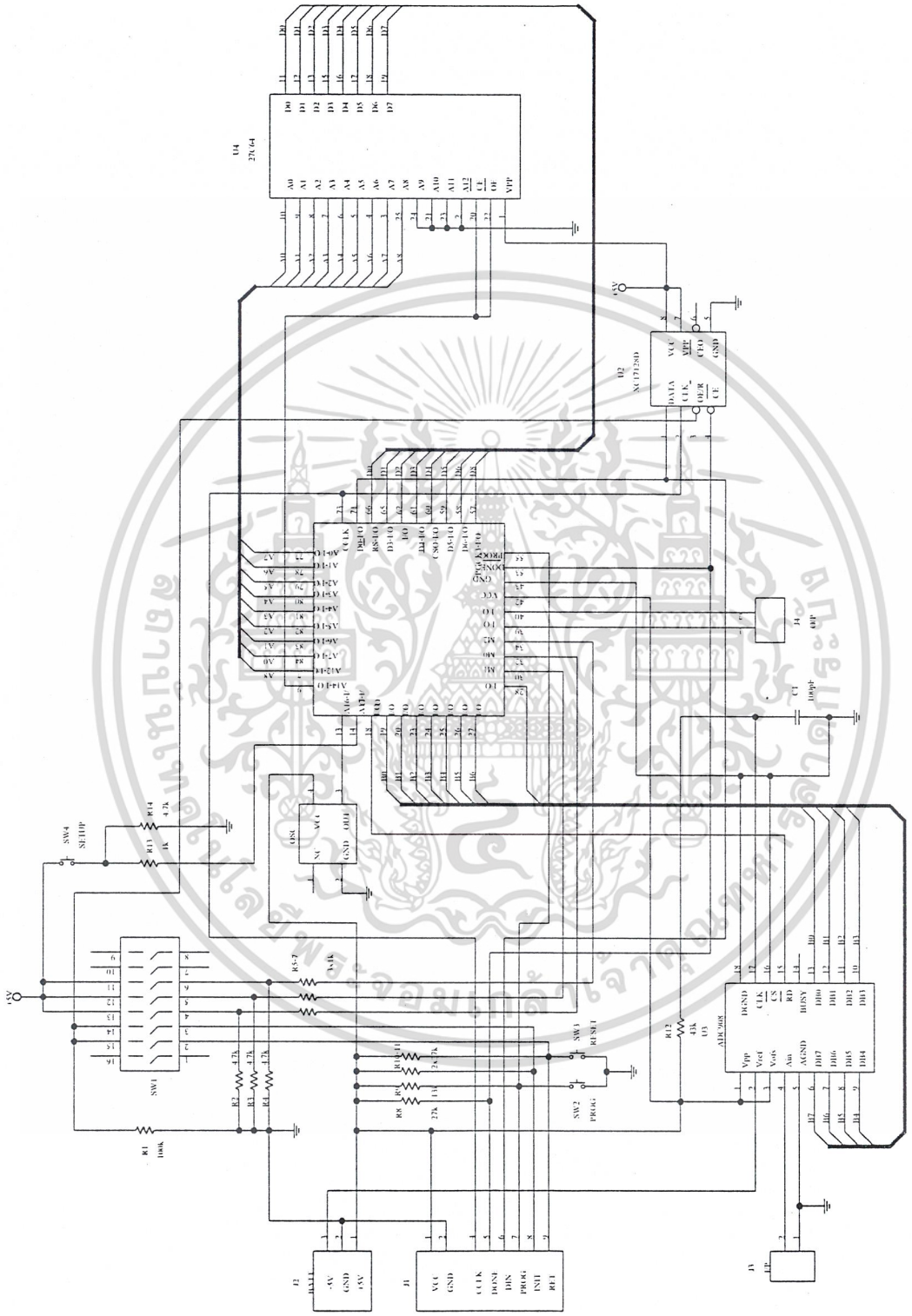
3.4 การสร้างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียง

การสร้างวงจรเข้ารหัสเสียงและถอดรหัสเสียงนั้นจะมีวิธีการสร้างเป็นวงจรทดสอบโดยใช้การดัดแปลงและปรับปรุงมาจากแผงวงจรตัวอย่างของ FPGAs และทำการสร้างวงจรเข้ารหัสเสียงด้วยโปรแกรมวิวลอจิก (Viewlogic) และต้องทำการโปรแกรมลงในอุปกรณ์ FPGAs ซึ่งจะทำการโปรแกรมโดยใช้ซอฟต์แวร์ของไซลิงค์ (Xilinx) ซึ่งจะโปรแกรมผ่านทางดาวน์โหลดเคเบิล (Down load cable) โดยวงจรที่ใช้ทดสอบแบ่งเป็นส่วนต่างๆ ดังนี้

3.4.1 การสร้างวงจรเข้ารหัสเสียง

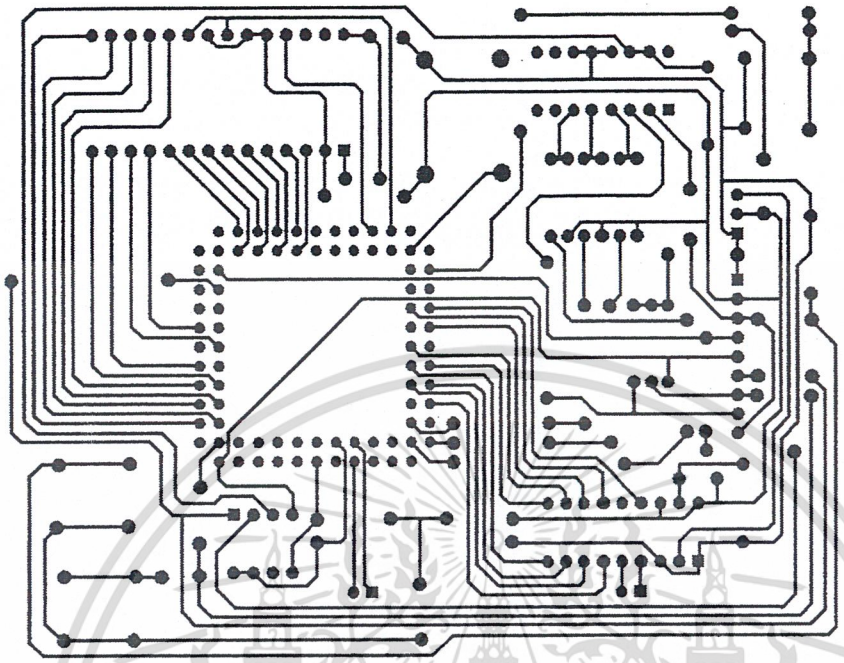
หลังจากที่ได้ทำการออกแบบวงจรเข้ารหัสเสียงตามแผนผังการทำงานที่ได้ออกแบบไว้แล้วตามรูปที่ 3.1 จะนำวงจรนี้มาทำการโปรแกรมลงในอุปกรณ์ FPGAs เบอร์ XC 4005 ซึ่งเป็นส่วนสำคัญของวงจรทดสอบนี้ วงจรเข้ารหัสเสียงจะประกอบไปด้วย วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลซึ่งใช้ไอซีเบอร์ ADC908 ข้อมูลเสียงที่ถูกบรรจุอยู่ในโค้ดบุ๊ค (Code Book) เป็นอีพรอมเบอร์ 27C64 โปรแกรมที่ทำการกำหนดโครงสร้างให้กับอุปกรณ์ FPGAs จะถูกโปรแกรมลงในพรอมเบอร์ 17C128 คริสตอล (Crystal) สำหรับกำเนิดความถี่เพื่อใช้ในการกำหนดเวลาทำงานให้กับวงจร และสวิตช์สำหรับเลือกโหมดในการทำงานของอุปกรณ์ FPGAs โดยที่การต่อร่วมของวงจรต่างๆ กับอุปกรณ์ FPGAs นั้นจะทำการต่อร่วมโดยคำนึงถึงวงจรภายใน FPGAs ที่ได้ออกแบบไว้ว่าหลังจากที่ได้ออกแบบวงจรภายในแล้วส่วนของอินพุตและส่วนของเอาต์พุตจะต้องตรงกับวงจรที่ต่อร่วมด้วย ซึ่งวงจรทดสอบของวงจรเข้ารหัสเสียงนั้นแสดงดังรูปที่ 3.15 รูปลายวงจรพิมพ์ของวงจรเข้ารหัสเสียงแสดงดังรูปที่ 3.16 รูปการวางอุปกรณ์แสดงดังรูปที่ 3.17 และ รูปวงจรเข้ารหัสเสียงเมื่อประกอบสำเร็จแสดงดังรูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

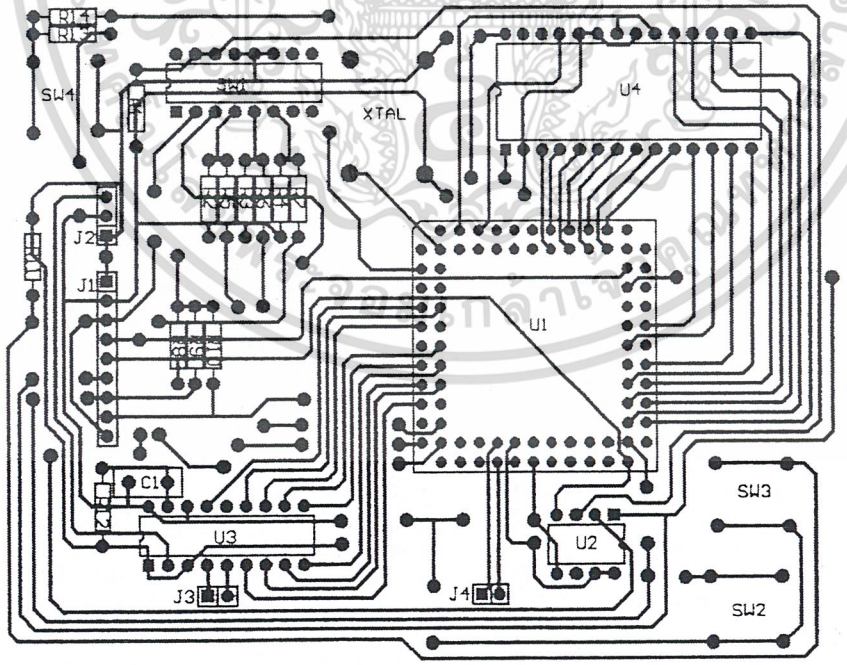


รูปที่ 3.15 วงจรเข้ารหัสเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

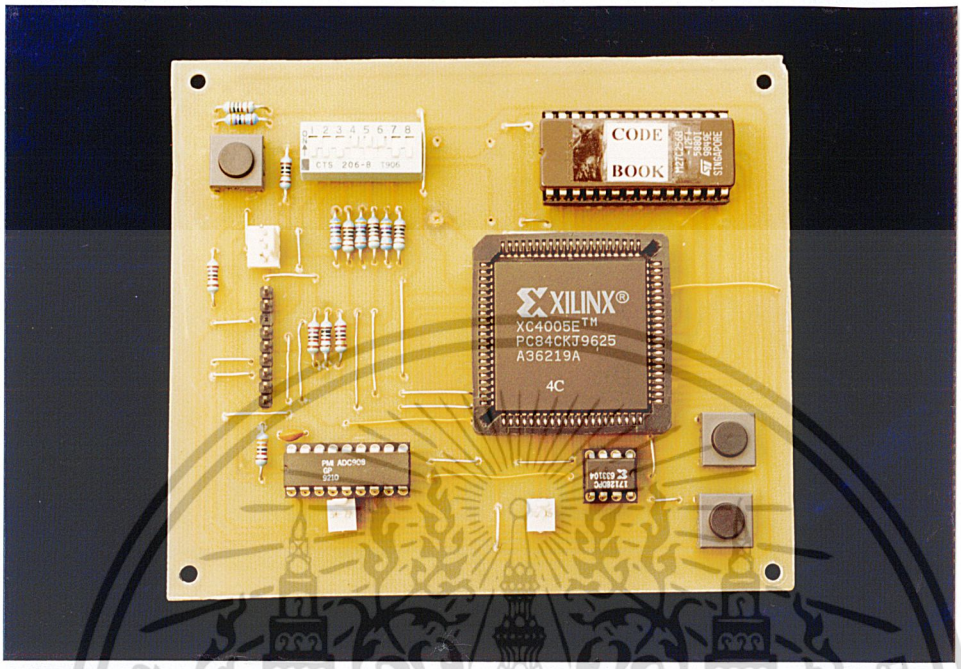


รูปที่ 3.16 ลายวงจรพิมพ์วงจรเข้ารหัสเสียง



รูปที่ 3.17 การวางอุปกรณ์วงจรเข้ารหัสเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

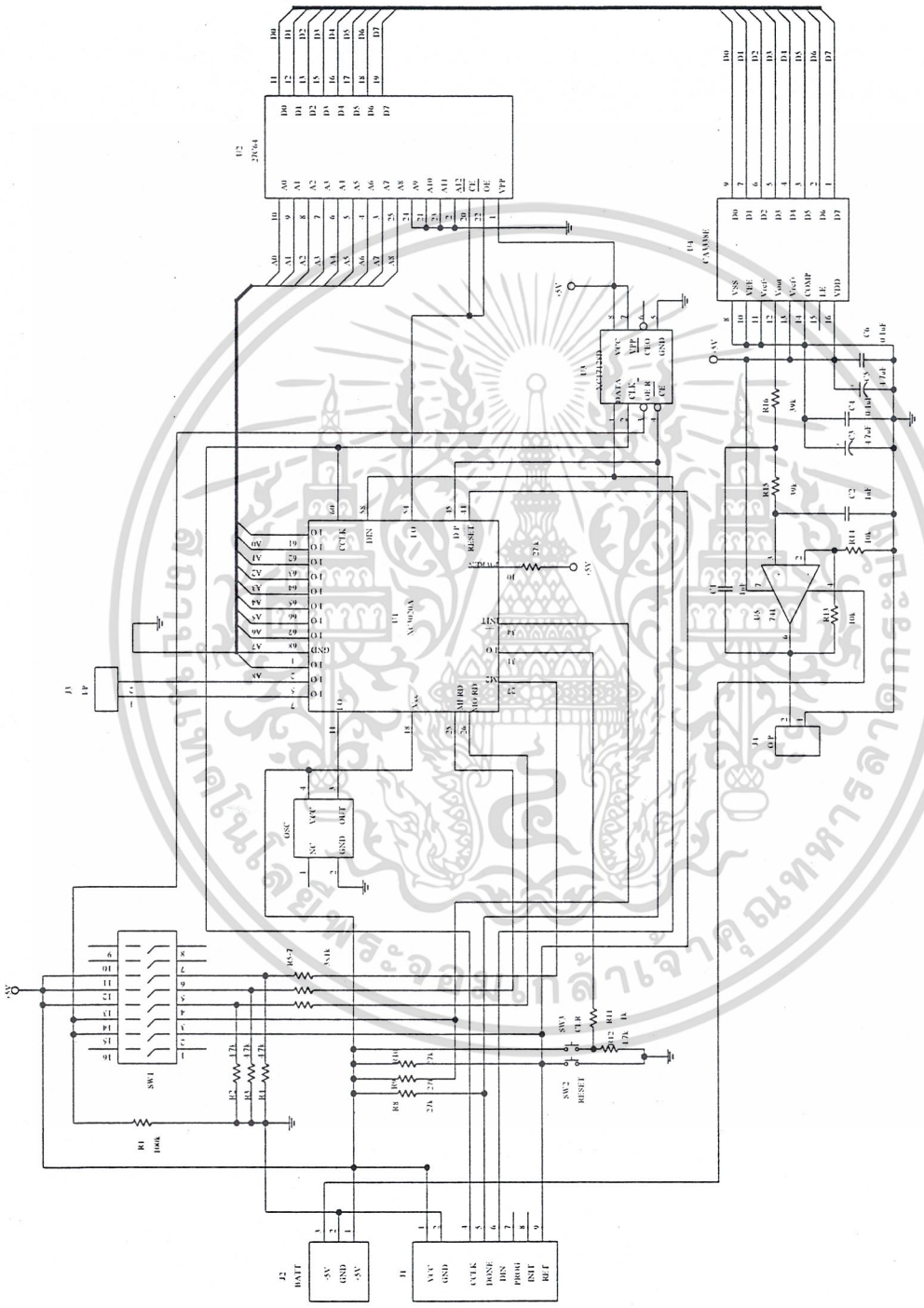


รูปที่ 3.18 วงจรเข้ารหัสเสียงประกอบสำเร็จ

3.4.2 การสร้างวงจรถอดรหัสเสียง

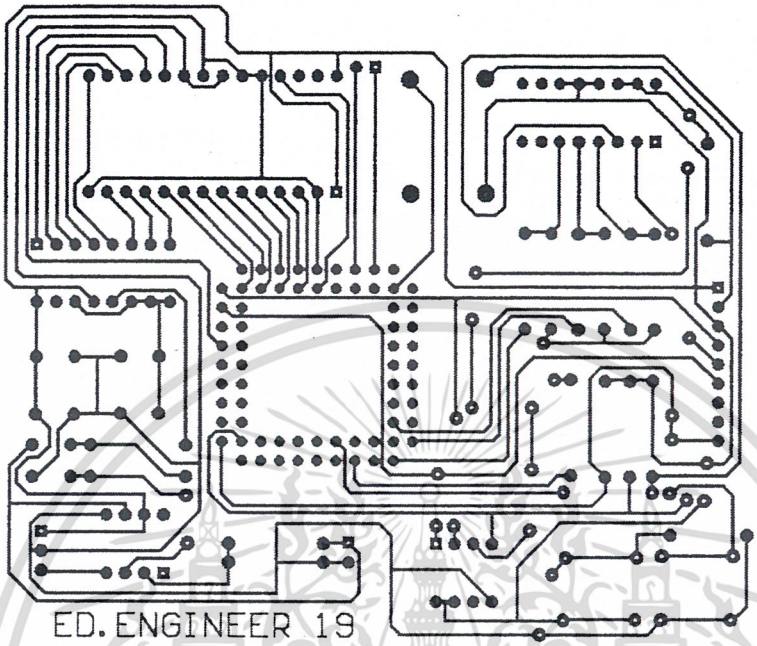
หลังจากที่ได้ทำการออกแบบวงจรถอดรหัสเสียงตามแผนผังการทำงานที่ได้ออกแบบไว้ตามรูปที่ 3.2 จะนำวงจรนี้มาทำการโปรแกรมลงในอุปกรณ์ FPGAs เบอร์ XC 3020 ซึ่งเป็นส่วนที่สำคัญของวงจรทดสอบนี้ วงจรถอดรหัสเสียงจะประกอบด้วย วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกซึ่งใช้ไอซีเบอร์ CA3338 และใช้ไอซีเบอร์ 741 สำหรับการกรองสัญญาณความถี่ต่ำเพื่อทำให้สัญญาณเอาต์พุตอยู่ในช่วงความถี่เสียง ข้อมูลเสียงที่ถูกบรรจุอยู่ในโค้ดบุ๊คเป็นอีพรอมเบอร์ 27C64 ซึ่งเป็นข้อมูลเดียวกับทางด้านวงจรเข้ารหัสเสียง โครงสร้างของวงจรภายใน FPGAs ทางด้านถอดรหัสเสียงจะถูกโปรแกรมลงในพรอมเบอร์ 17C128 คริสตอลสำหรับกำเนิดความถี่เพื่อกำหนดเวลาในการทำงานของวงจรและสวิตช์สำหรับเลือกโหมดการทำงานของ FPGAs วงจรทดสอบของวงจรถอดรหัสเสียงแสดงดังรูปที่ 3.19 รูปลายวงจรพิมพ์ของวงจรถอดรหัสเสียงแสดงดังรูปที่ 3.20 รูปการวางอุปกรณ์แสดงดังรูปที่ 3.21 และรูปวงจรถอดรหัสเสียงเมื่อประกอบสำเร็จแสดงดังรูปที่ 3.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

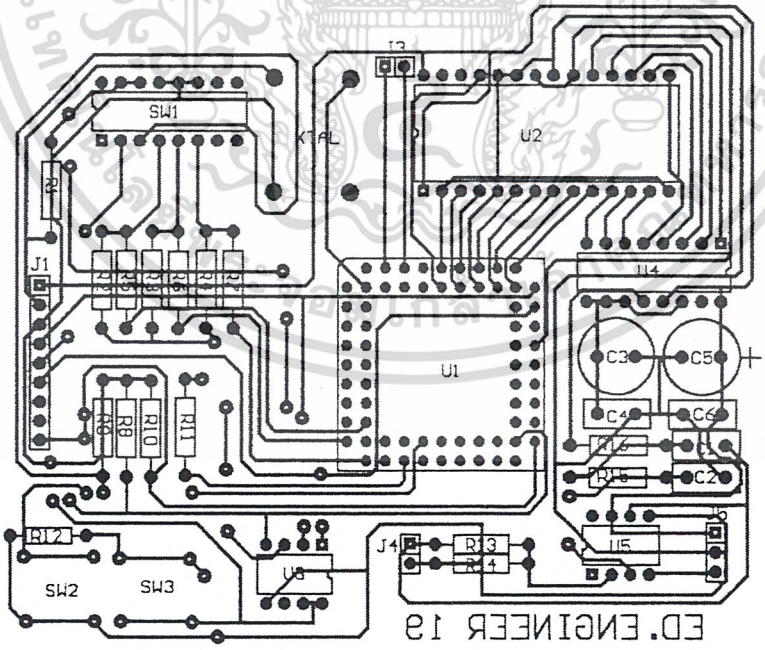


รูปที่ 3.19 วงจรถอดรหัสเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

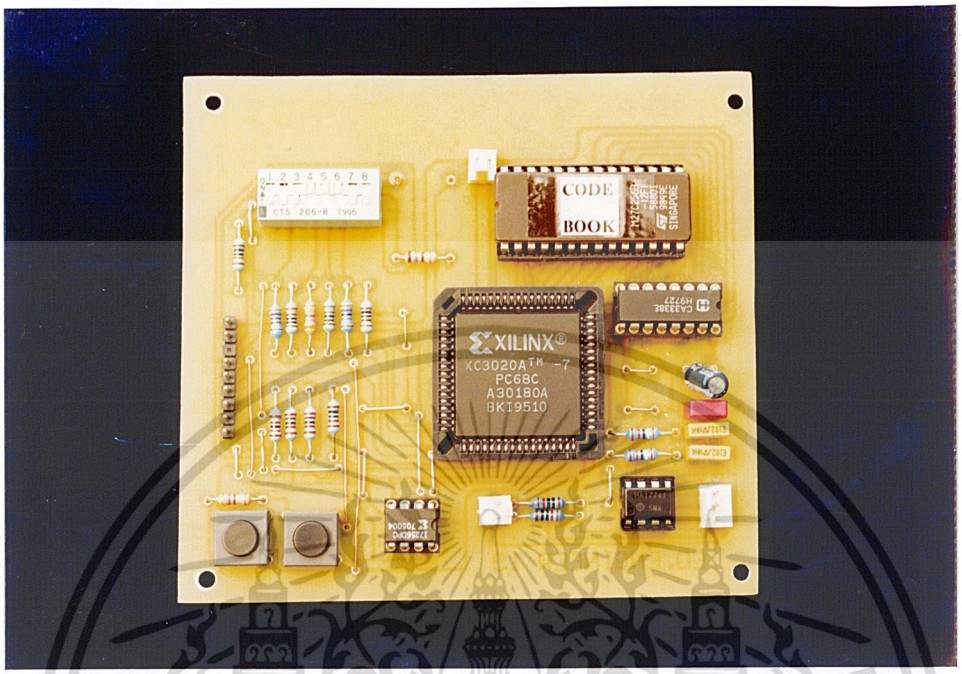


รูปที่ 3.20 ลายวงจรพิมพ์วงจรถอดรหัสเสียง



รูปที่ 3.21 การวางอุปกรณ์วงจรถอดรหัสเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



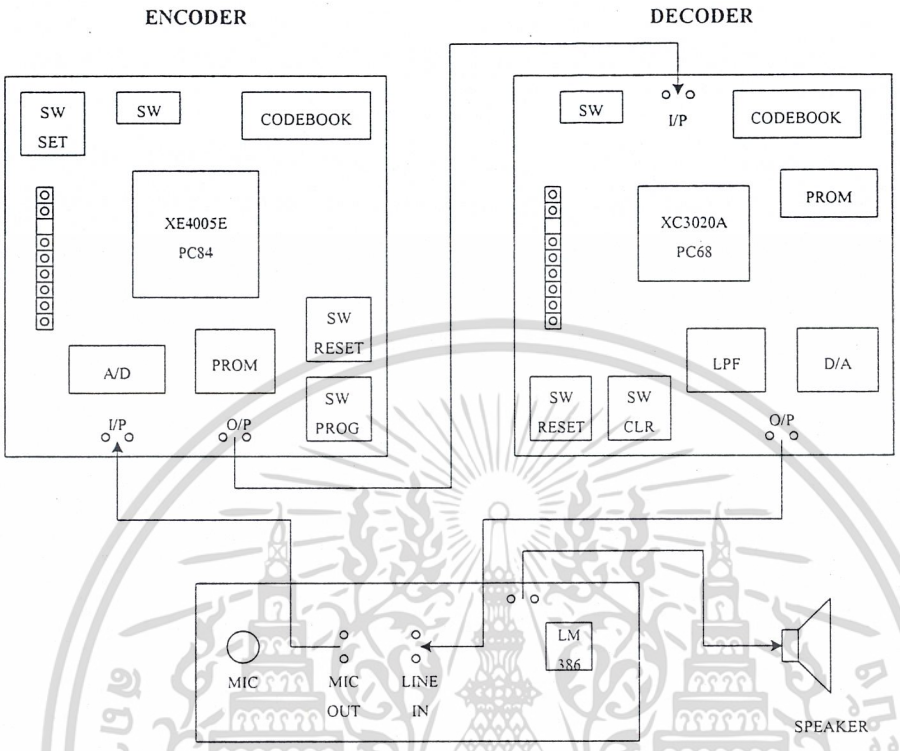
รูปที่ 3.22 วงจรถอดรหัสเสียงประกอบสำเร็จ

3.5 การเชื่อมต่อระหว่างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียง

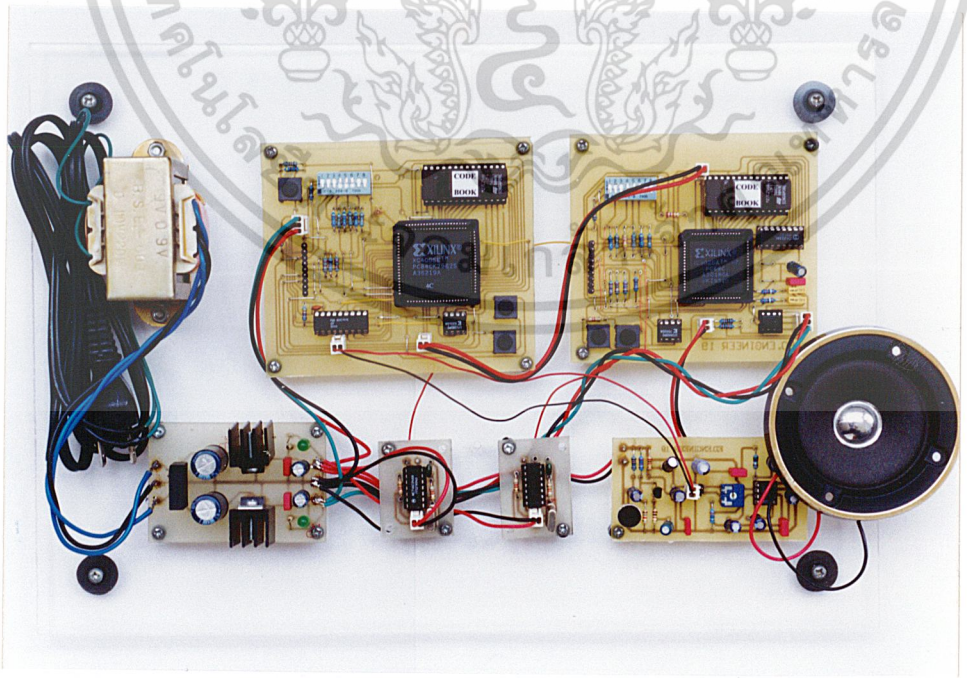
การทดสอบวงจรควอนไตซ์เวกเตอร์นั้นจะต้องทำการเชื่อมต่อวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียงเข้าด้วยกัน โดยที่ส่วนอินพุตของวงจรเข้ารหัสเสียงคือวงจรปรีไมโครโฟน และส่วนเอาต์พุตของวงจรถอดรหัสเสียงคือวงจรขยายสัญญาณเสียงจะต้องทำการเชื่อมต่อเข้าไป รวมทั้งแหล่งจ่ายไฟขนาด 5 โวลต์ด้วยเพื่อให้ได้วงจรที่สมบูรณ์แบบ

สำหรับวิธีการเชื่อมต่อวงจรทั้งหมดนั้นได้แสดงรูปแบบการต่อใช้งานไว้ดังรูปที่ 3.23 และรูปการเชื่อมต่อที่สมบูรณ์แบบเมื่อทำการเชื่อมต่อสำเร็จเรียบร้อยแล้วได้แสดงดังรูปที่ 3.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 การเชื่อมต่อระหว่างวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียง



รูปที่ 3.24 รูปวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียงเมื่อประกอบสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 กล่าวนำ

เครื่องควอนไตซ์เวกเตอร์ที่เวลาจริง โดยใช้อุปกรณ์ FPGAs เป็นเครื่องที่สร้างขึ้นเพื่อควอนไตซ์สัญญาณเสียง โดยวิธีการเข้ารหัส ซึ่งในบทที่ 1-3 ได้กล่าวถึงแนวความคิด ทฤษฎีการออกแบบและการสร้างมาแล้ว ในบทนี้จะกล่าวถึงการทดลองการทำงานของเครื่องควอนไตซ์เวกเตอร์

การทดลองการทำงานของเครื่องควอนไตซ์เวกเตอร์จะเริ่มกระทำตั้งแต่การทดสอบการทำงานวงจรส่วนประกอบวงจรเข้ารหัสสัญญาณเสียง การต่อเครื่องควอนไตซ์เวกเตอร์ทั้งชุดเข้าด้วยกัน จากนั้นทำการดาวน์โหลด (Down Load) วงจรที่ได้ออกแบบไว้แล้วลงในอุปกรณ์ FPGAs จากนั้นทำการทดลองป้อนสัญญาณเสียงที่ต้องการส่งโดยผ่านการควอนไตซ์เวกเตอร์เข้าที่เครื่องส่งโดยให้เก็บข้อมูลและวัดความถี่ก่อนส่ง จากนั้นทดลองฟังเสียงและทำการวัดสัญญาณที่เครื่องรับ

4.2 ขั้นตอนการทดสอบการทำงาน

4.2.1 ขั้นตอนการทดสอบวงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียง

จากที่ได้ออกแบบวงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียงซึ่งประกอบด้วยวงจรต่างๆ ตามแผนผังการทำงานแล้วนั้น ขั้นต่อไปเป็นการทดสอบการทำงานของแต่ละวงจรที่ได้ออกแบบไว้ เมื่อทำการทดสอบการทำงานแต่ละวงจรเรียบร้อยแล้ว ขั้นต่อไปจะนำวงจรส่วนต่างๆ มาประกอบร่วมกันเพื่อให้ได้เป็นวงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียงที่สมบูรณ์แบบ ซึ่งขั้นตอนการทดสอบการทำงานของวงจรที่ได้ออกแบบไว้มีดังนี้

1) เมื่อทำการสร้างวงจรตามที่ได้ออกแบบไว้โดยใช้โปรแกรม View Draw และทำการบันทึก ส่วนของโปรแกรมจะกำหนดชื่อแฟ้มให้เป็นนามสกุล .1
2) เลือกรายการ Tool Create Digital ในโปรแกรม View Draw เพื่อทำการตรวจสอบข้อผิดพลาดของวงจรและโปรแกรมจะเปลี่ยนชื่อแฟ้มให้เป็นนามสกุล .VSM
3) เลือกโปรแกรม Speed Wave เพื่อเข้าสู่ขั้นตอนการกำหนดสัญญาณอินพุต โดยทำการเปิดชื่อแฟ้มนามสกุล .VSM ที่ได้บันทึกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) เลือกรายการ Signal Stimulus ในโปรแกรม Speed Wave เพื่อทำการกำหนดสัญญาณอินพุตแต่ละประเภทการใช้งานตามวงจรที่ได้ออกแบบไว้
5) เลือกรายการ Wave Form Stream ในโปรแกรม Speed Wave เพื่อกำหนดรายการข้อมูลที่ต้องการให้โปรแกรมแสดงผล
6) เลือกโปรแกรม View Trace เพื่อทดสอบการทำงานของวงจร โดยโปรแกรมจะทำการทดสอบการทำงานของวงจรจากรายการ Simulation
7) เลือกรายการ Run ในโปรแกรม View Trace เพื่อทำการทดสอบการทำงานของวงจรและผลของการทดสอบวงจรจะปรากฏทางจอแสดงผล

ในการทดสอบการทำงานของแต่ละวงจรส่วนประกอบวงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียงนั้นได้ทำการทดสอบการทำงานและได้ผลการทำงานถูกต้องตามความเป็นจริง ซึ่งรายละเอียดวงจรและผลของการทดสอบการทำงานของแต่ละวงจรได้แสดงในภาคผนวก ก และผลการทำงานของวงจรเข้ารหัสสัญญาณเสียงและวงจรถอดรหัสสัญญาณเสียงที่นำวงจรแต่ละวงจรมาประกอบร่วมกันแล้วทำการทดสอบปรากฏว่าได้ผลการทำงานที่ถูกต้องและได้แสดงผลการทำงานในภาคผนวก ก

4.2.2 ขั้นตอนในการสร้างไฟล์องค์ประกอบ

หลังจากทำขั้นตอนการทดสอบโดยโปรแกรมเสร็จเรียบร้อย ต่อไปเป็นการตรวจสอบวงจรโดยการดาวน์โหลดลงบนอุปกรณ์ FPGAs และขั้นตอนการทดสอบนี้จะต้องใช้ดาวน์โหลดเคเบิ้ล (Download cable) เป็นตัวส่งองค์ประกอบสำหรับควบคุมให้ FPGAs เบอร์ XC4005E ทำงานโดยองค์ประกอบที่ใช้ควบคุม FPGAs จะถูกเก็บเป็นแฟ้ม .BIT ซึ่งขั้นตอนการสร้างไฟล์สามารถทำได้ตามขั้นตอนต่าง ๆ ดังต่อไปนี้ การกำหนดองค์ประกอบให้กับ XDM และ XMAKE ใน XDM (XACT Design Manager) นั้น XMAKE จะทำการแปลงโปรแกรมส่วนต่าง ๆ โดยอัตโนมัติโดยใช้กระบวนการของ Xilink (Xilink design flow) ในการทำกระบวนการออกแบบที่ซับซ้อน เช่น ในการทำโปรแกรมหนึ่งต้องป้อนชื่อในชั้นสูงสุดของ Schemetics ที่สร้างไว้แล้ว XMAKE จะทำการค้นและกระทำทุก ๆ ชั้นที่ต่ำลงมาของการออกแบบซึ่งจะได้แฟ้ม LCA ที่ทำการจัดวางและหาเส้นทางแล้วและให้แฟ้ม .BIT ออกมาซึ่งพร้อมที่จะดาวน์โหลดให้ FPGAs ทำงานได้

4.2.3 การดาวน์โหลดโดยใช้โปรแกรม XACT

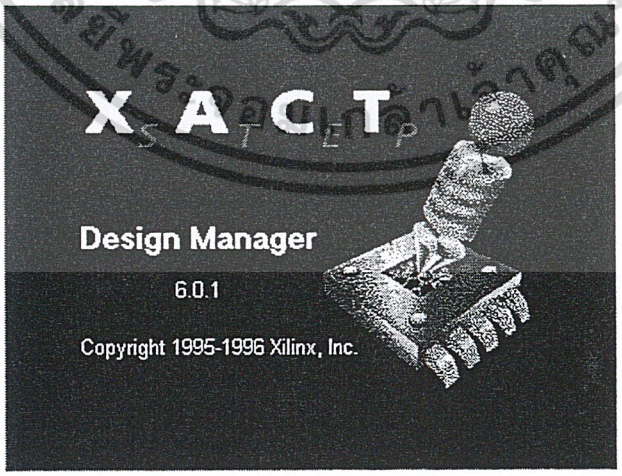
เมื่อเราได้ไฟล์องค์ประกอบออกมาแล้ว ขั้นต่อไปจะทำการทดลองโดยการดาวน์โหลดบิตสตรีม (Bitstream) นั้นลงไป เพื่อทำการทดสอบเมื่อต่อสายดาวน์โหลดเคเบิลเข้ากับเครื่องคอมพิวเตอร์เรียบร้อยแล้ว สามารถทำการดาวน์โหลดได้ด้วยขั้นตอนต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ทำการตั้งสวิตช์ต่าง ๆ ในแผงวงจรของ FPGAs โดยให้อยู่ในแบบสเลฟ ซึ่งสามารถควบคุมได้โดยการตั้งสวิตช์ ๆ ดังตารางที่ 4.1
2) เลือกรายการ Shortcut to Sgnmgr.exe เพื่อเข้าสู่โปรแกรม XACT จะปรากฏหน้าต่างของ Design Manager ดังรูปที่ 4.2

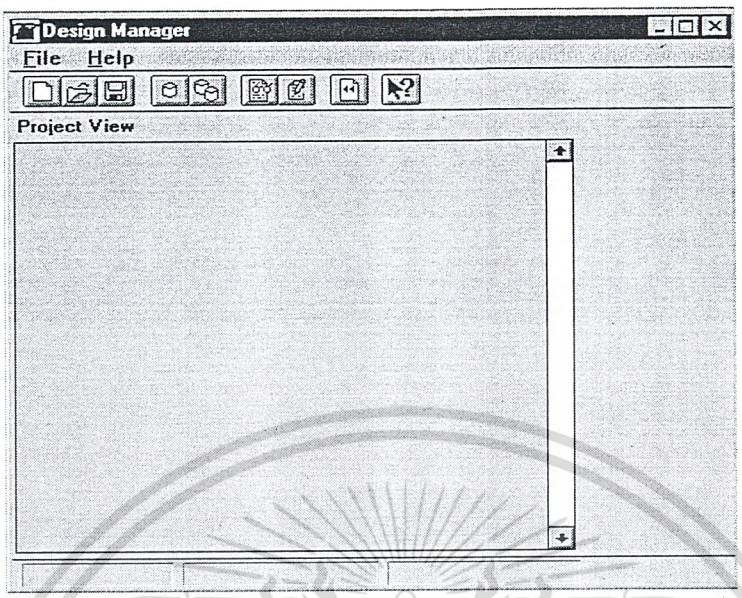
ตารางที่ 4.1 การตั้งสวิตช์ต่าง ๆ ในแผงวงจรของ FPGAs

SWITCH (DIP.SW)	LABEL	SETTING
1	PWR	X
2	MPE	OFF
3	SPE	OFF
4	M0	ON
5	M1	ON
6	M2	ON
7	RST	X
8	INIT	OFF



รูปที่ 4.1 การเข้าสู่ XDM...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

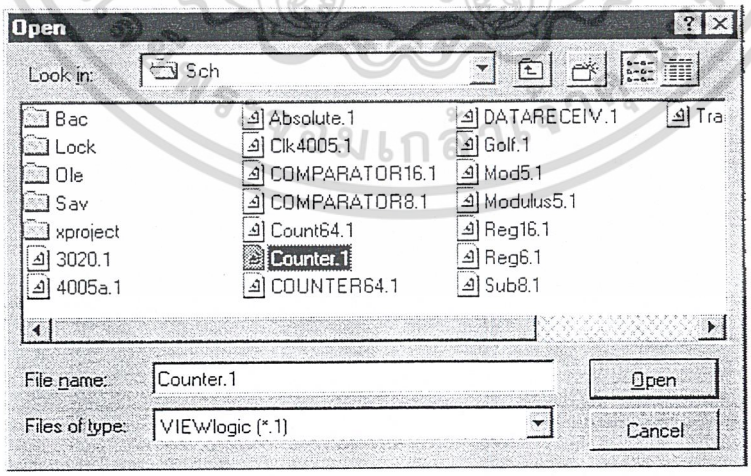


รูปที่ 4.2 หน้าต่างของ Design Manager

3) เมื่อเข้าสู่โปรแกรม XDM แล้ว ทำการเลือก File และ New Project จะปรากฏ Dialog Box ดังรูปที่ 4.3

-ใส่ชื่อแฟ้มใน Project Name

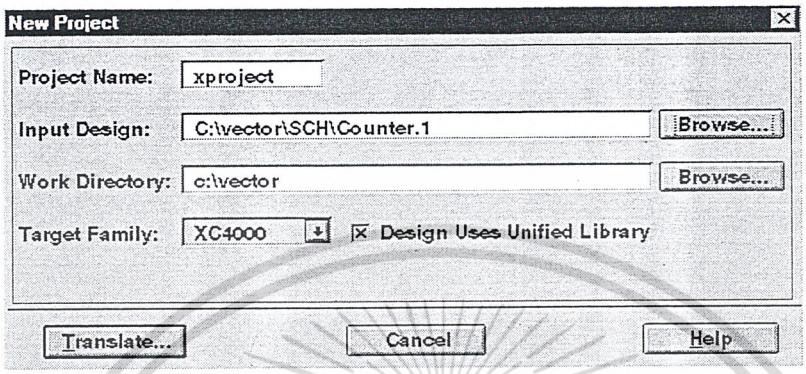
-เปิดวงจรที่ต้องการดาวน์โหลดที่ Input Design โดยเลือกที่ Browse จะปรากฏ Dialog Box ของ Open ดังรูปที่ 4.3



รูปที่ 4.3 Dialog Box ของ Open

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

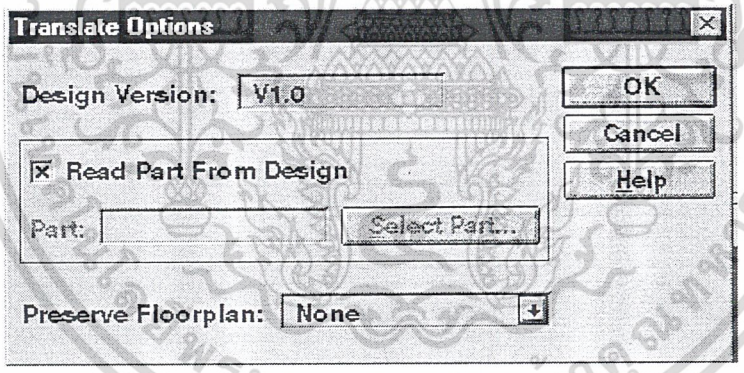
-เลือกเปิดแฟ้มชื่อ Counter จะปรากฏที่ Dialog Box ของ New Project ดังแสดงในรูปที่ 4.4



รูปที่ 4.4 Dialog Box ของ New Project

-เลือก Target Family เป็น XC 4000

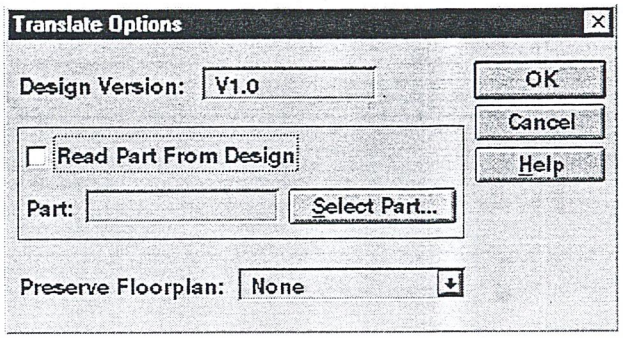
4) เลือกที่ Translate จะปรากฏ Dialog Box ของ Translate Option ดังรูปที่ 4.5



รูปที่ 4.5 Dialog Box ของ Translate Option

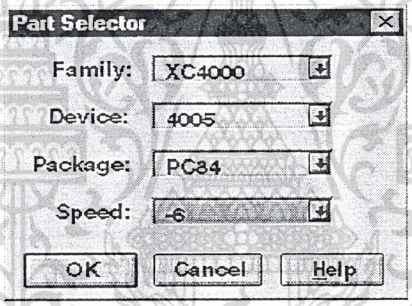
5) ยกเลิกการกำหนดอุปกรณ์ตามที่ซอฟต์แวร์กำหนดมาให้ โดยยกเลิกกากบาทที่ Read Part From Design จะปรากฏ Dialog Box การเลือกอุปกรณ์ดังรูปที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



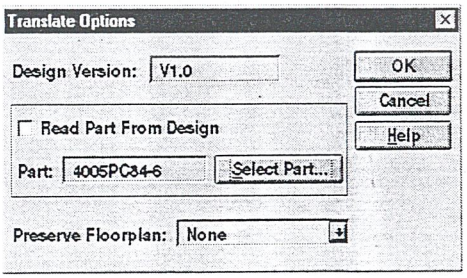
รูปที่ 4.6 Dialog Box การเลือกอุปกรณ์

6) เลือกอุปกรณ์ให้ถูกต้องตามที่ออกแบบโดยกดที่ Select Part จะปรากฏ Dialog Box ของ Select Part ดังรูปที่ 4.7



รูปที่ 4.7 Dialog Box ของ Select Part

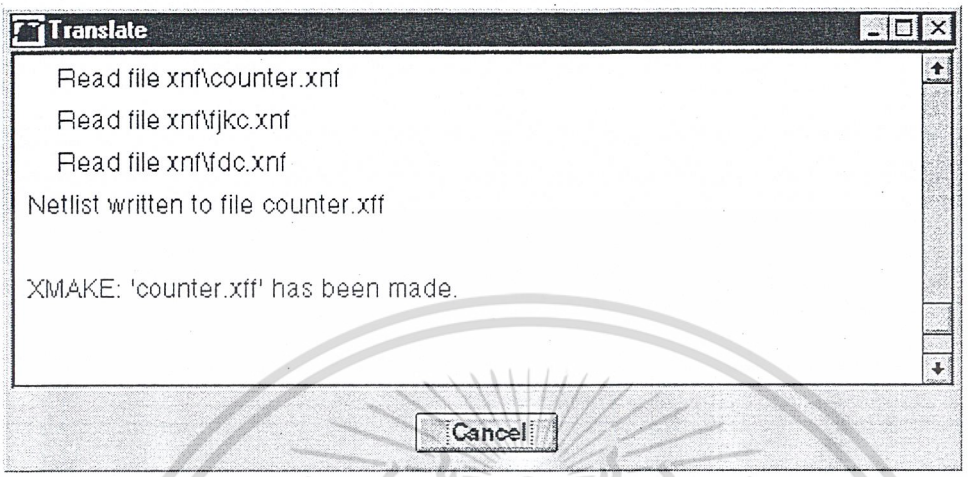
7) เปลี่ยน Device เป็น 4005 และเปลี่ยน Speed เป็น -6 จากนั้นกด OK จะปรากฏข้อความที่ช่อง Part เป็น 4005PC84-6 ดังรูปที่ 4.8



รูปที่ 4.8 Dialog Box ของ Translate Option ที่เลือกอุปกรณ์ที่ถูกต้องแล้ว

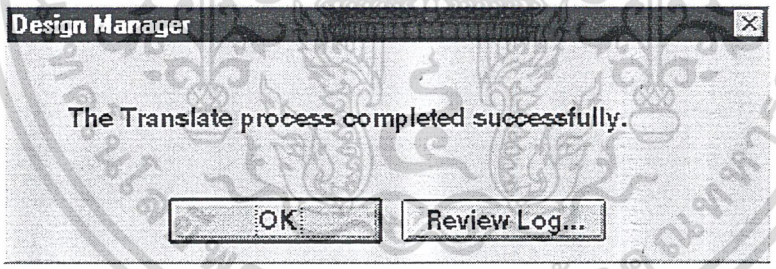
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) กด OK ที่หน้าจอจะแสดงกระบวนการ Translate ดังรูปที่ 4.9



รูปที่ 4.9 หน้าจอกระบวนการ Translate

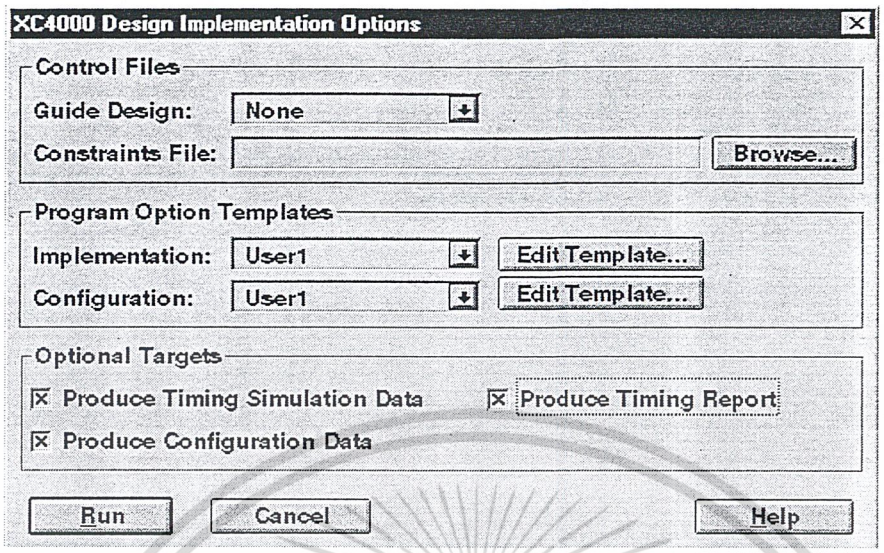
หลังจากนั้นที่หน้าจอจะปรากฏ Box ที่แสดงว่ากระบวนการ Translate สมบูรณ์ดังแสดงในรูปที่ 4.10



รูปที่ 4.10 หน้าจอที่แสดงว่ากระบวนการ Translate สมบูรณ์

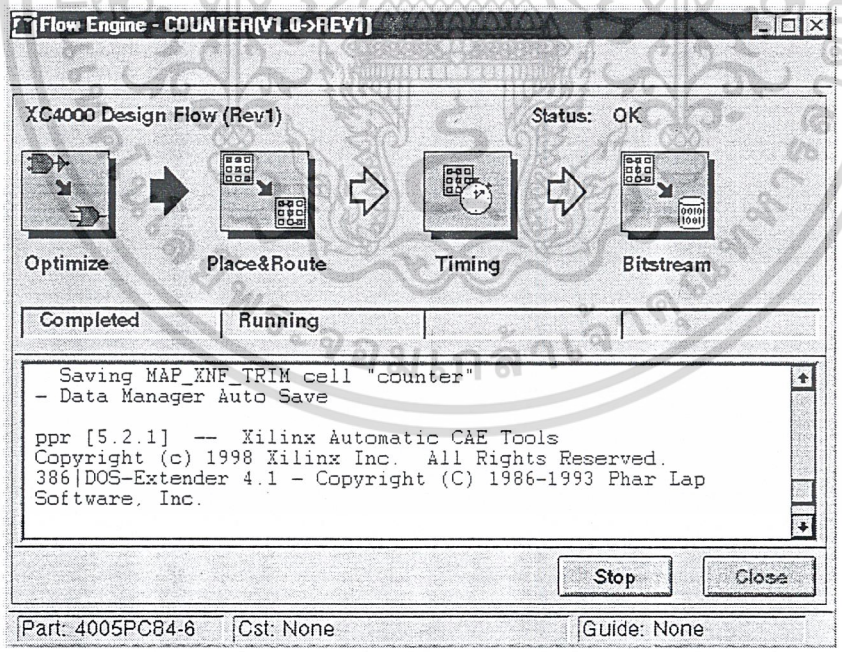
9) กด OK ที่หน้าจอก็จะกลับเข้าสู่หน้าต่างของ Design Manager จากนั้นเลือกที่ Menu Design จากนั้นเลื่อน Tab เลือกคำสั่ง Implement จะปรากฏ Dialog Box ของ XC400 Design Implement Options จากนั้นเลือก Option Targets ทุกช่อง ดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 Dialog box ของ XC400 Design Implement Options

10) กด Run เพื่อทำการ Implement จากนั้นที่หน้าจอจะปรากฏกระบวนการ Implement ดังรูปที่ 4.12



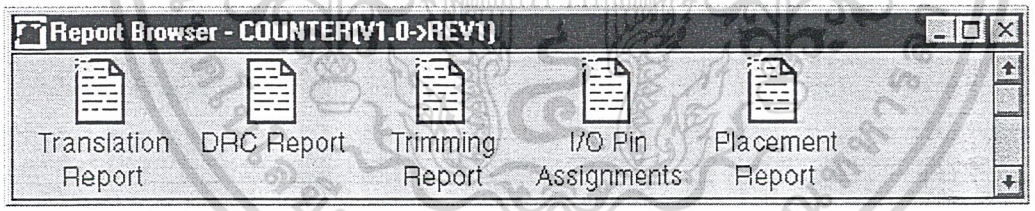
รูปที่ 4.12 หน้าจอแสดงกระบวนการ Implement

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกระบวนการ Implement เสร็จสมบูรณ์แล้วที่หน้าจอจะแสดง Dialog Box ของ Design Manager ดังรูปที่ 4.13

รูปที่ 4.13 หน้าจอแสดง Dialog Box ของกระบวนการ Implement ที่เสร็จสมบูรณ์

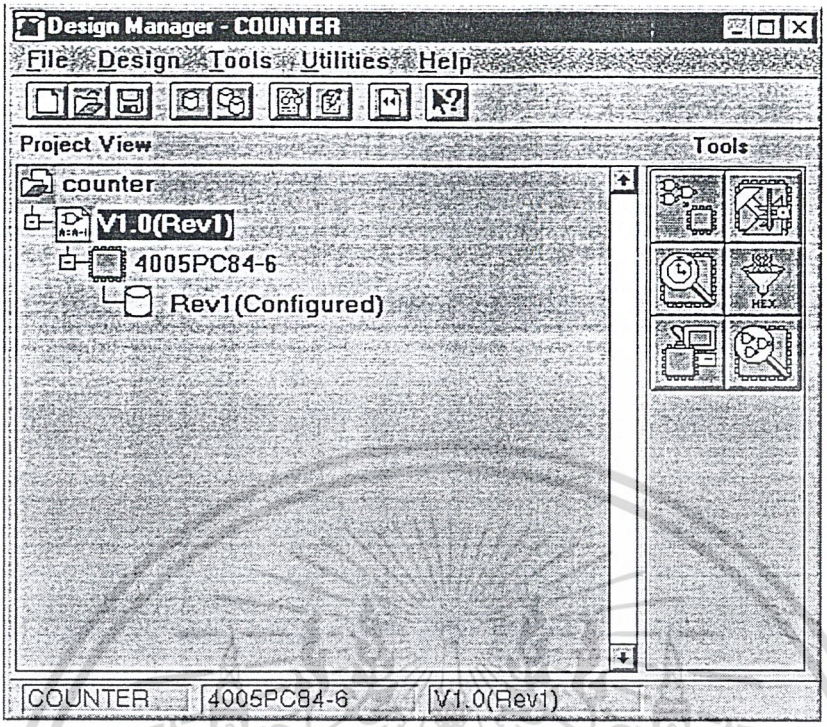
นอกจากนั้นที่หน้าจอยังจะแสดง Dialog Box แสดง Report Browser ที่สร้างขึ้นระหว่างกระบวนการ Implement เพื่อเป็นการรายงานกระบวนการต่าง ๆ ระหว่างการ Implement ผู้ใช้สามารถเข้าไปอ่าน Report เหล่านั้นได้โดย Double Click ที่ Report นั้น ๆ ถ้าหาก Report ที่ถูกสร้างขึ้นยังไม่ได้เปิดอ่านจะปรากฏรูปดาวที่มุมบนด้านขวา และถ้าหากเปิดอ่านแล้วดาวก็จะหายไปดังรูปที่ 4.14



รูปที่ 4.14 Dialog Box ของ Report Browser

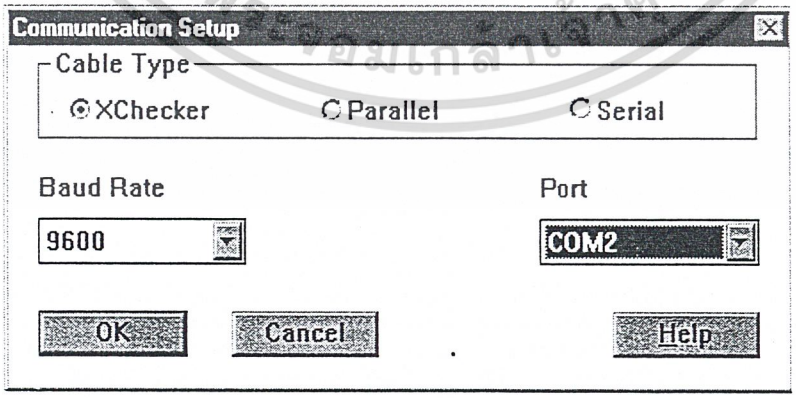
11) เมื่อถึงขั้นตอนนี้ก็แสดงว่าแฟ้มนั้นๆ พร้อมที่จะดาวน์โหลดแล้วโดยที่หน้าต่างของ Design Manager จะแสดงดังรูปที่ 4.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



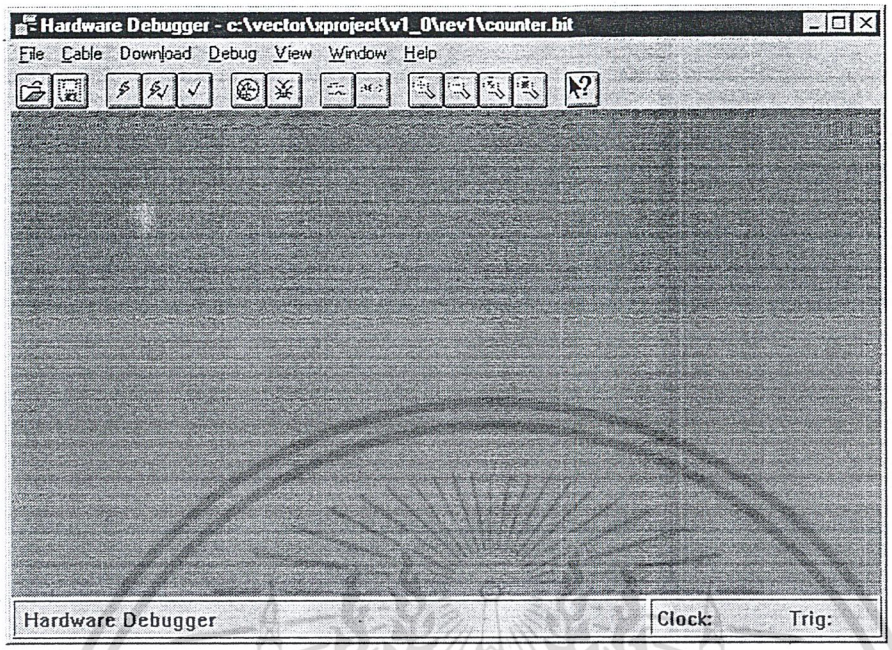
รูปที่ 4.15 หน้าต่างของ Design Manager ที่พร้อมที่จะทำการดาวน์โหลด

12) ที่หน้าต่างของ Design Manager เลือกรายการ Tool แล้วเลื่อน Tab มาที่คำสั่ง Hardware Debugger จะปรากฏ Dialog Box ของ Hardware Debugger และจะปรากฏ Dialog Box ของ Communication Setup ดังรูปที่ 4.16 เลือก Port ไปที่ Com2 จากนั้นกด OK จะปรากฏ Dialog Box ของ Hardware Debugger ดังแสดงในรูปที่4.17



รูปที่ 4.16 Dialog Box ของ Communication Setup

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 Dialog Box ของ Hardware Debugger

12) ที่ Dialog Box ของ Hardware Debugger เลือกรายการไปที่ Download แล้วเลื่อน Tab มาที่คำสั่ง Download Design เครื่องก็จะทำการ Download ลงไปยังแผงวงจร FPGAs ให้ในกระบวนการดาวน์โหลดที่ยกตัวอย่างมานี้เป็นการดาวน์โหลดสำหรับ FPGAs ตระกูล XC4000 ส่วนการดาวน์โหลด FPGAs ตระกูล XC3000 นั้นจะมีขั้นตอนเดียวกันเพียงแต่ในขั้นตอน Port Selector ที่โปรแกรมต้องการรับทราบตระกูลของFPGAs นั้นให้ใส่เป็น XC3000 และจำนวนขาของ FPGAs ให้กำหนดตามจำนวนขาของ FPGAs นั้น

4.3 การโปรแกรม Serial PROM

การโปรแกรม Serial PROM จะเป็นการกำหนดโครงสร้างภายในให้กับอุปกรณ์ FPGAs โดยไม่ต้องใช้วิธีการดาวน์โหลดจากโปรแกรมทุกครั้งที่ใช้งาน ซึ่งการดาวน์โหลดจากโปรแกรมนั้นจะมีข้อเสียคือ หลังจากที่ทำการดาวน์โหลดแล้ววงจรไม่สามารถที่จะขาดไฟเลี้ยงวงจรได้เพราะโครงสร้างภายใน FPGAs นั้นจะหายไป ซึ่งจำเป็นที่จะต้องทำการดาวน์โหลดใหม่ ดังนั้นการใช้อุปกรณ์ Serial PROM มาเก็บโครงสร้างให้กับอุปกรณ์ FPGAs นั้นเป็นวิธีการที่ดีสุด ซึ่งมีวิธีการดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) เข้าสู่โปรแกรม XDM (XACT Design Manager)
2) ทดสอบไดเร็กทอรี (Directory) ที่ตั้งไว้ ซึ่งในที่นี้คือ ENCODE (วงจรเข้ารหัสเสียง) และ DECODE (วงจรถอดรหัสเสียง) ซึ่งจะแสดงที่มุมล่างด้านขวาของจอภาพ ถ้ายังไม่ได้กำหนดหรือตั้งไว้ไม่ถูกต้องให้เลือกที่ Directory: และใช้เมาส์ (Mouse) เลือกไดเร็กทอรีที่ต้องการ
3) ทำการเลือกที่ Family: ที่มุมล่างด้านซ้ายของจอภาพก็จะปรากฏซึ่งเราสามารถเลือกได้ ในที่นี้เลือก 4005APC84
4) จากเมนูหลักของ XDM เลือก Venify เลือก MAKEPROM จากนั้นจะเข้าสู่หน้าต่างของ MAKEPROM กดคีย์ (Key) ใดๆ เพื่อให้โปรแกรมทำงานต่อไป
5) เลือกเมนู Mics เลือก Set เลือก Promsize เลือก XC17128D (สำหรับวงเข้ารหัสเสียง) และ XC17136D (สำหรับวงจรถอดรหัสเสียง)
6) เลือก PROM เลือก Format เลือก Mcs86
7) เลือก PROM เลือก Load จากนั้นให้พิมพ์ 00000000 ที่ PROM address: ที่อยู่มุมด้านล่างซ้าย เลือก UP เลือกแฟ้มที่ต้องการ เช่น COMPLETE. BIT หลังจากนั้นจะมีตัวเลขในตาราง
8) ทำการเก็บค่าโดยเลือก PROM เลือก Save แล้วป้อนชื่อแฟ้มที่ต้องการเก็บและจะได้แฟ้มที่มีนามสกุล .MCS
9) ออกจากโปรแกรม MAKEPROM และ XDM
10) นำแฟ้ม COMPLETE.MCS มาทำการโปรแกรมลงบนพรอม โดยใช้โปรแกรม HW-130 Programmer ซึ่งเป็นเครื่องมือการพัฒนาของบริษัทไซลิงค์ (Xilinx) ก็จะได้ Serial ที่มีการกำหนดขนาดโครงสร้างให้กับอุปกรณ์ FPGAs ไว้แล้ว

4.4 ขั้นตอนการทดลอง

เมื่อได้กำหนดโครงสร้างของวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียงลงในพรอมเรียบร้อยและทำการเชื่อมต่อวงจรทดสอบวงจรเข้ารหัสเสียงและถอดรหัสเสียงเข้าด้วยกันดังรูปที่ 3.23 หลังจากนั้นทำการทดลองตามขั้นตอนต่อไปนี้

1) นำพรอมที่กำหนดโครงสร้างเรียบร้อยแล้วประกอบเข้าไปในวงจรทดสอบวงจรเข้ารหัสเสียงและวงจรถอดรหัสเสียง
2) ตั้งสวิตช์ที่แผงวงจรเข้ารหัสเสียงตามตารางที่ 4.2 และแผงวงจรเข้ารหัสเสียงตามตารางที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) จ่ายไฟเลี้ยงให้กับวงจรเข้ารหัสเสียง วงจรถอดรหัสเสียง วงจรปรีไมโครโฟนและวงจรขยายสัญญาณเสียง
4) นำออสซิลโลสโคป (Oscilloscope) วัดสัญญาณเสียงที่ทำการทดสอบ โดยทำการวัดที่ส่วนอินพุทของวงจรแปลงสัญญาณแอนะลอกเป็นดิจิตอลในแผงวงจรเข้ารหัสเสียง
5) ทดสอบสัญญาณเสียงโดยทำการส่งสัญญาณเสียงผ่านทางไมโครโฟนแล้วทำการวัดรูปสัญญาณ
6) นำออสซิลโลสโคป (Oscilloscope) วัดสัญญาณเสียงที่ทำการทดสอบ โดยทำการวัดที่ส่วนเอาต์พุตของวงจรกรองความถี่ต่ำในแผงวงจรถอดรหัสเสียง
7) ทดสอบสัญญาณเสียงโดยทำการส่งสัญญาณเสียงเดียวกันกับขั้นตอนที่ 5 ผ่านทางไมโครโฟนแล้วทำการวัดรูปสัญญาณ
8) ทำการทดลองซ้ำตั้งแต่ขั้นตอนที่ 4 โดยทำการเปลี่ยนสัญญาณเสียงทดสอบให้แตกต่างกันในการทดลองแต่ละครั้ง

ตารางที่ 4.2 การตั้งสวิตช์ต่าง ๆ ในแผงวงจรเข้ารหัสเสียง

SWITCH (DIP.SW)	LABEL	SETTING
1	PWR	X
2	MPE	ON
3	SPE	ON
4	M0	OFF
5	M1	OFF
6	M2	OFF
7	RST	X
8	INIT	OFF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

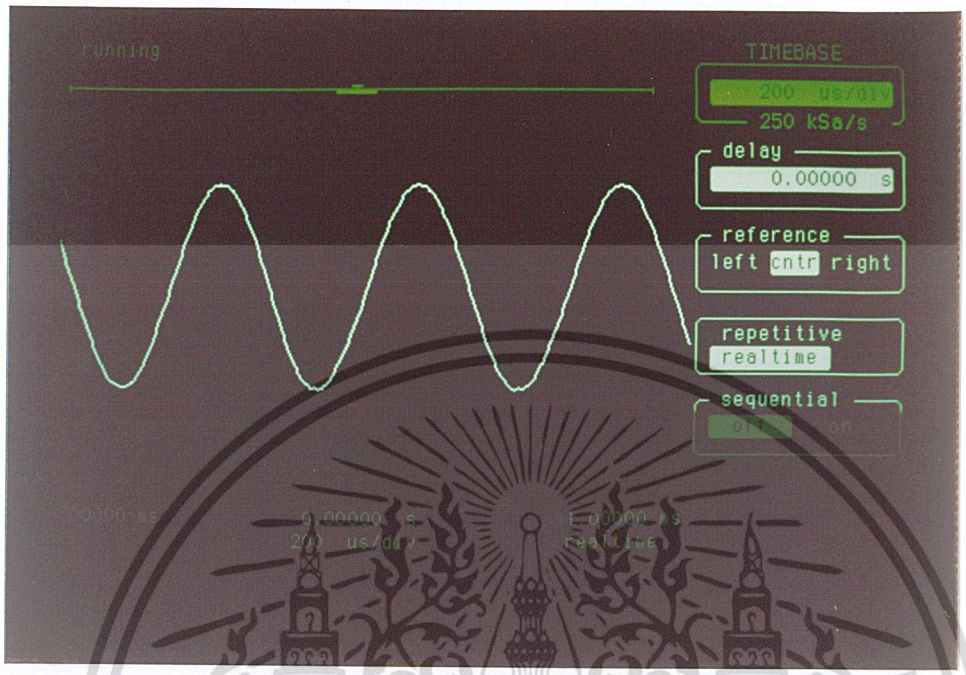
ตารางที่ 4.3 การตั้งสวิตช์ต่าง ๆ ในแผงวงจรถอดรหัสเสียง

SWITCH (DIP.SW)	LABEL	SETTING
1	INP	X
2	MPE	ON
3	SPE	ON
4	M0	OFF
5	M1	OFF
6	M2	OFF
7	MCLK	X
8	DOUT	OFF

4.5 ผลการทดลอง

จากที่ได้ทำการทดลองวงจรเข้ารหัสสัญญาณเสียงนั้น โดยทำการป้อนสัญญาณเสียงที่วงจรแปลงสัญญาณแอนะลอกเป็นดิจิตอล ผลการทดลองปรากฏว่าสัญญาณเสียงที่ได้ทางด้านเอาต์พุตกับสัญญาณเสียงทางด้านอินพุตเป็นสัญญาณที่มีลักษณะแตกต่างกันมาก เนื่องจากความผิดพลาดในการประมวลผลสัญญาณซึ่งขึ้นอยู่กับข้อจำกัดทางด้านเวลาในการประมวลผลและสัญญาณเกิดการลดทอนและผิดเพี้ยนเนื่องจากคุณภาพของอุปกรณ์ที่นำมาใช้งานซึ่งสามารถเปรียบเทียบสัญญาณทางด้านอินพุตและทางด้านเอาต์พุตได้จากรูปที่ 4.18 และรูปที่ 4.19 ซึ่งสัญญาณเสียงที่ได้ทางเอาต์พุตนั้นเป็นเสียงที่เกิดจากการทำงานของวงจรที่เวลาจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 สัญญาณเสียงทางด้านอินพุต

รูปที่ 4.19 สัญญาณเสียงทางด้านเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป ปัญหา แนวทางแก้ไขและ พัฒนา

5.1 บทสรุป

การพัฒนาตัวเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานและอุปกรณ์ FPGAs สร้างขึ้นเพื่อศึกษา พัฒนาและประยุกต์ใช้งานระบบประมวลผลสัญญาณดิจิตอลในการเข้ารหัสสัญญาณเสียงด้วยวิธีการควอนไตซ์เวกเตอร์แบบมาตรฐานซึ่งขอบเขตในการพัฒนาโครงงานชิ้นนี้คือทำให้เครื่องเข้ารหัสเสียงสามารถทำงานที่เวลาจริงและมีบิตเรตในการส่งข้อมูลที่ 9600 บิตต่อวินาที จากที่ได้ทำการศึกษาและทดลองสร้างปรากฏว่าเครื่องเข้ารหัสเสียงสามารถทำงานที่เวลาจริงและมีบิตเรตลดลง ซึ่งเป็นไปตามทฤษฎีและจุดมุ่งหมายที่วางไว้ แต่เกิดข้อจำกัดในด้าน Hardware ที่ไม่สามารถทำงานได้ทันเวลาทำให้สัญญาณเสียงเอาต์พุตผิดพลาด

5.2 ปัญหาและแนวทางแก้ไข

1) เวลาที่ใช้ในการประมวลผลสัญญาณของวงจรทั้งระบบ มากกว่าช่วงเวลาในการสุ่มสัญญาณเสียงทำให้การประมวลผลผิดพลาด
   - วิธีแก้ปัญหา ลดความถี่ของอัตราการสุ่มสัญญาณลงแต่จะต้องอยู่ในเงื่อนไขของทฤษฎีการสุ่มสัญญาณของไนต์ควิสต์ (Nytquist)
2) สัญญาณเสียงเอาต์พุตเกิดการลดทอนและผิดเพี้ยนสูง
   - วิธีแก้ปัญหา ปรับปรุงส่วนวงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอกให้ดีขึ้น

5.3 แนวทางในการพัฒนา

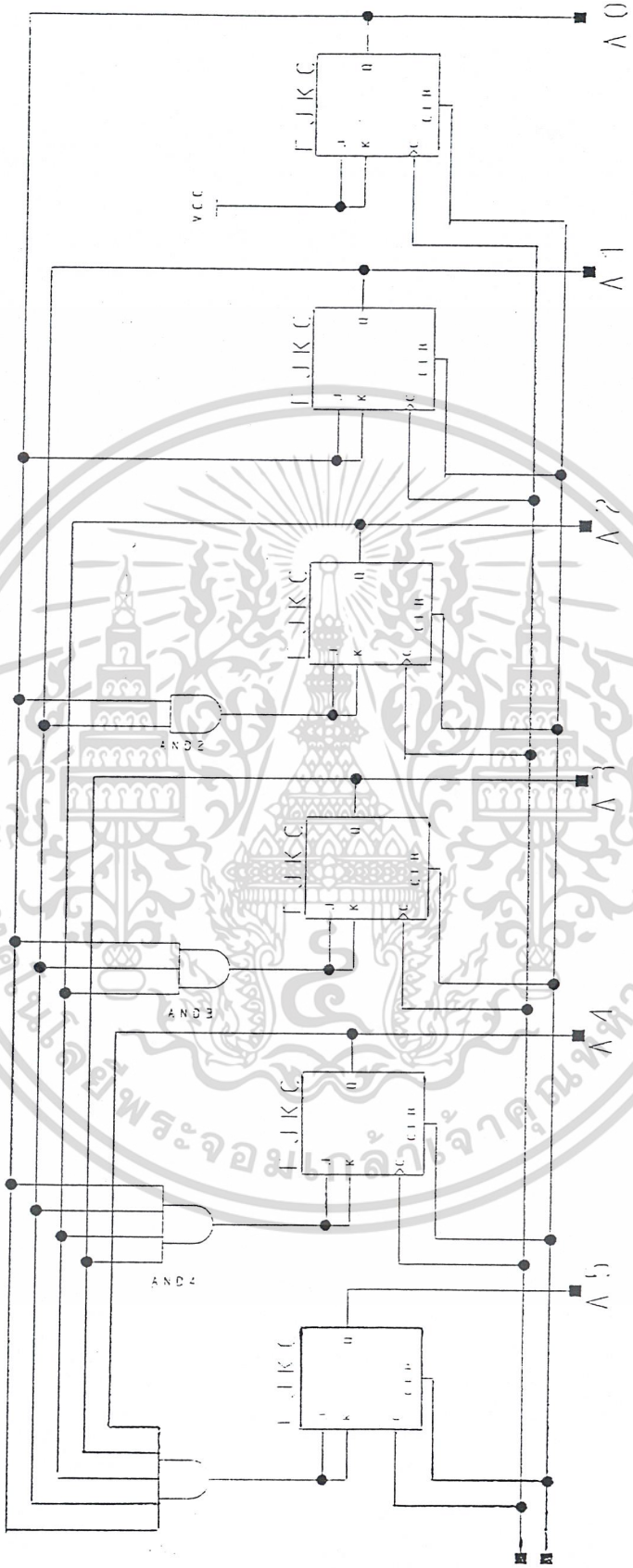
1) ลดขนาดฮาร์ดแวร์ (Hard ware) ให้เล็กลง เพื่อสามารถนำไปใช้งานเป็นอุปกรณ์เข้ารหัสสัญญาณเสียงได้จริง
2) ออกแบบวงจรภายใน FPGAs ให้เล็กลง เพื่อให้สามารถใช้งานได้ที่ความถี่สุ่ม 8 kHz (หรือ 125 ไมโครวินาที ต่อ 1 แซมเปิ้ล) ตามมาตรฐานของ CCITT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

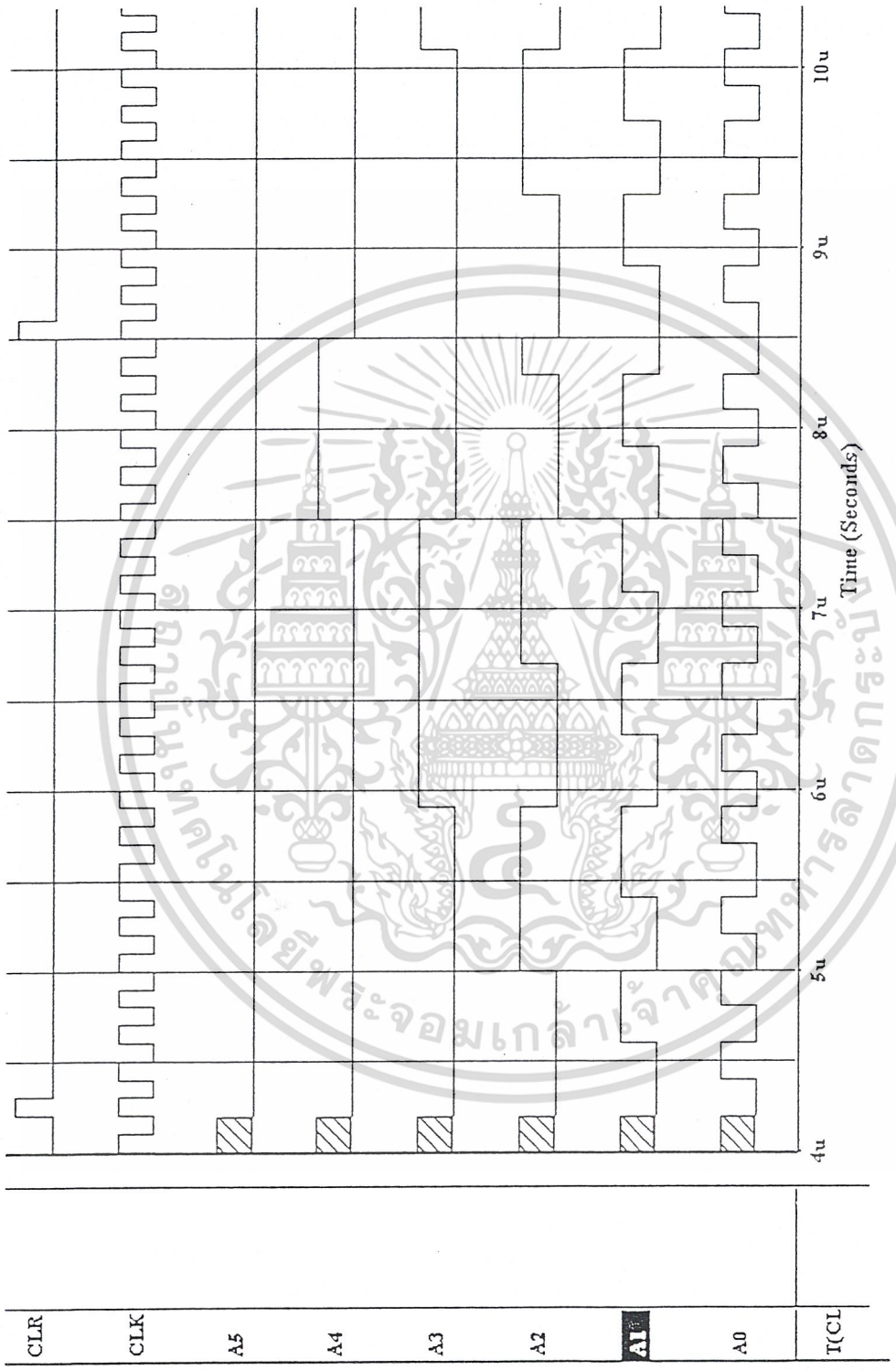
รายละเอียดวงจรและผลการทดสอบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



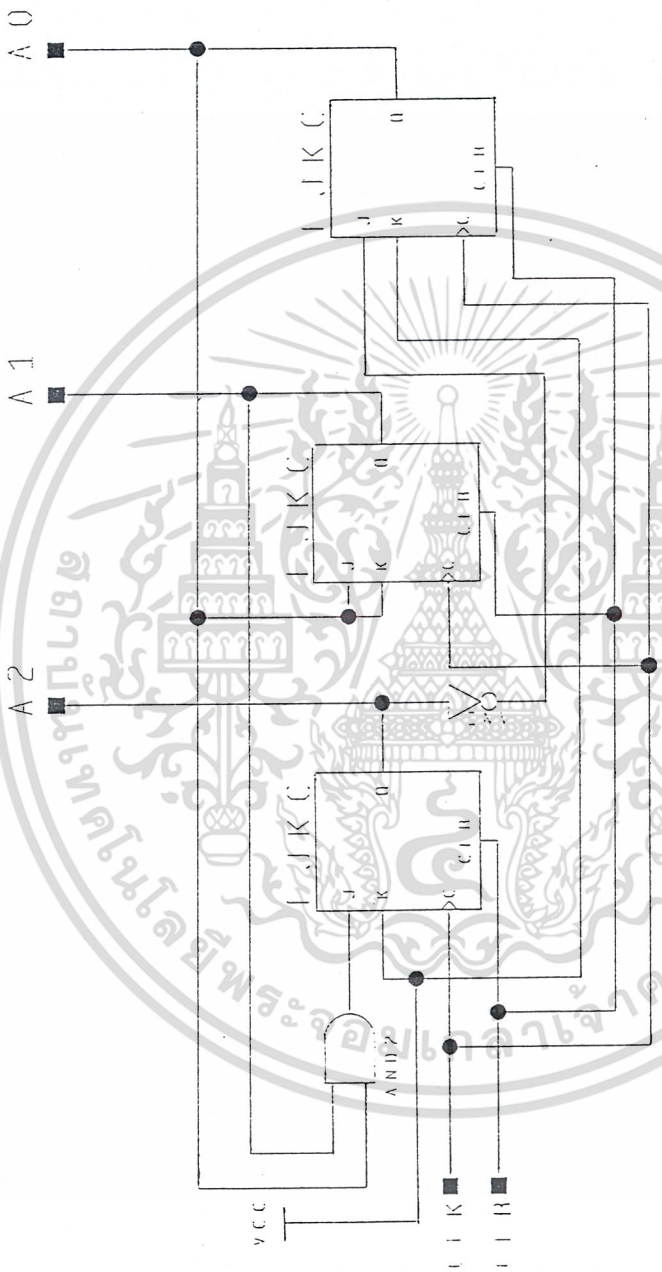
รูปที่ ก.1 วงจรนับ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



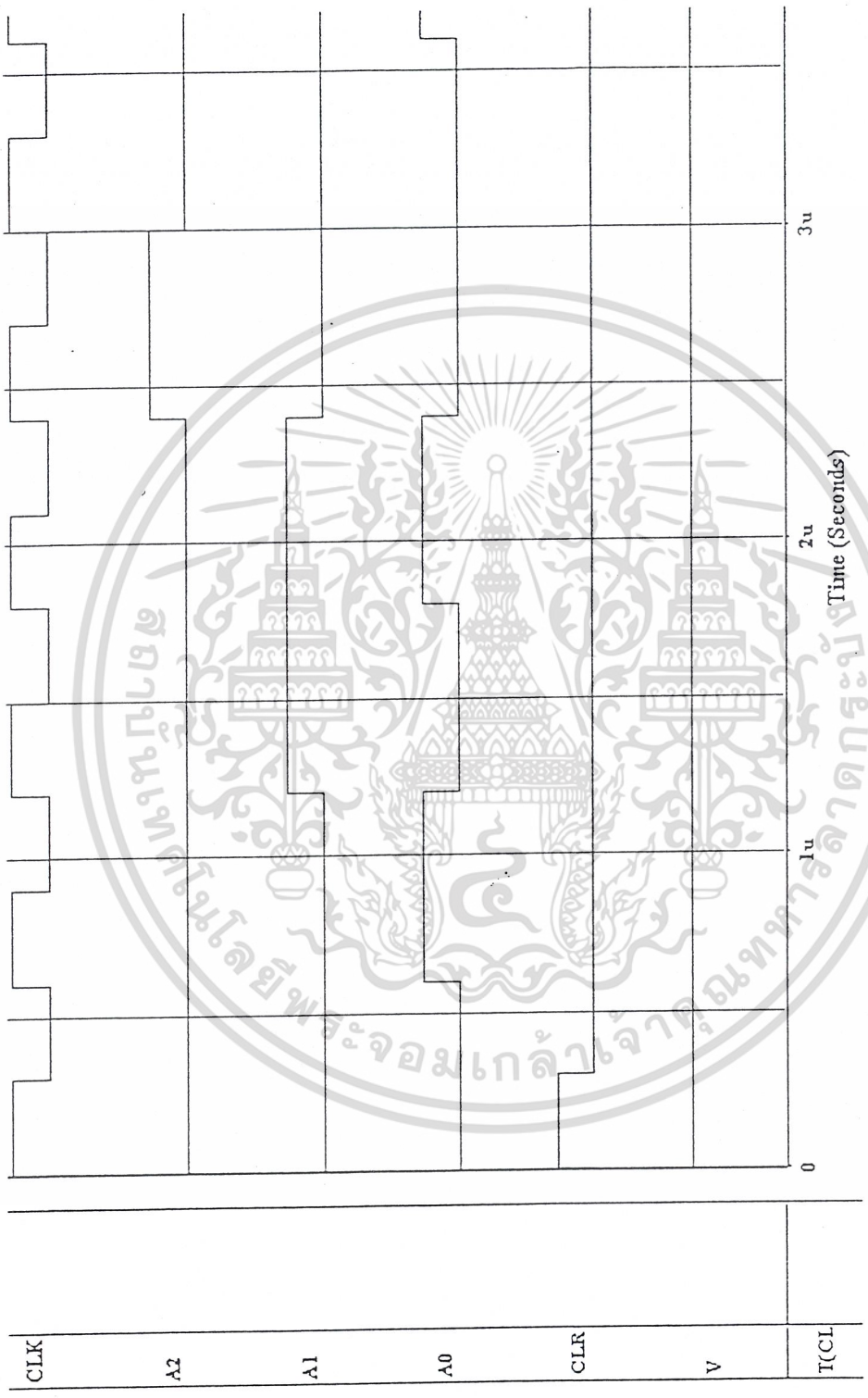
รูปที่ ก.2 ผลการทดสอบวงจรนับ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



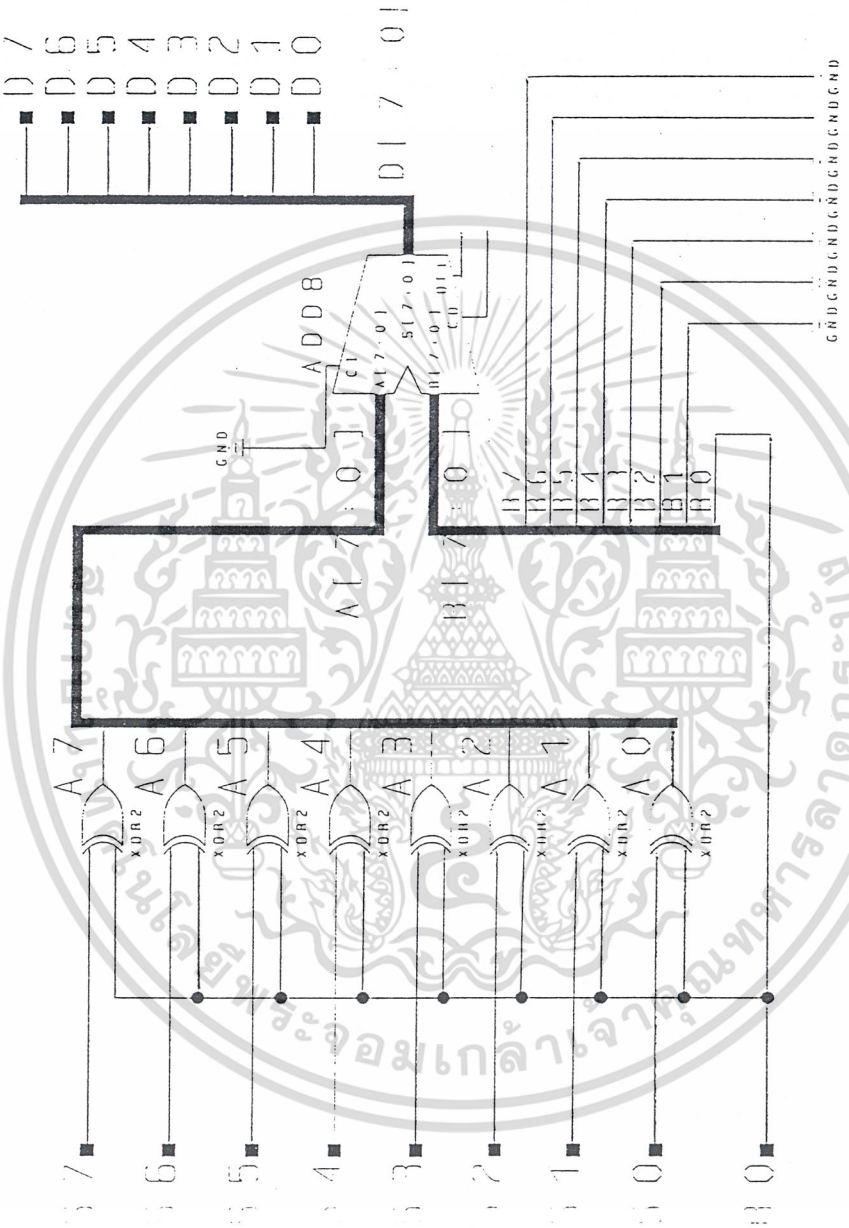
รูปที่ ก.3 วงจร MOD 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



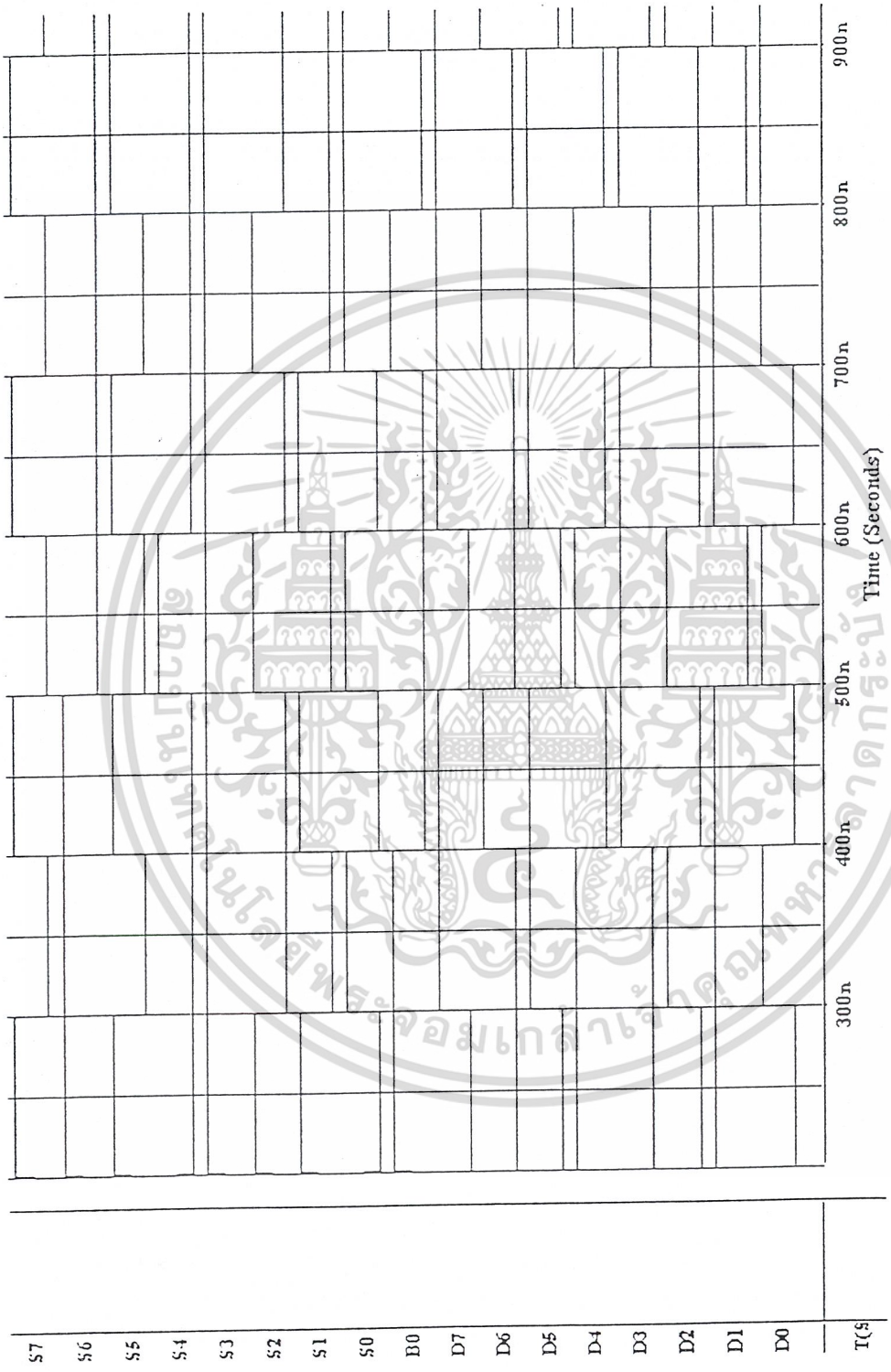
รูปที่ ก.4 ผลการทดสอบวงจร MOD 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



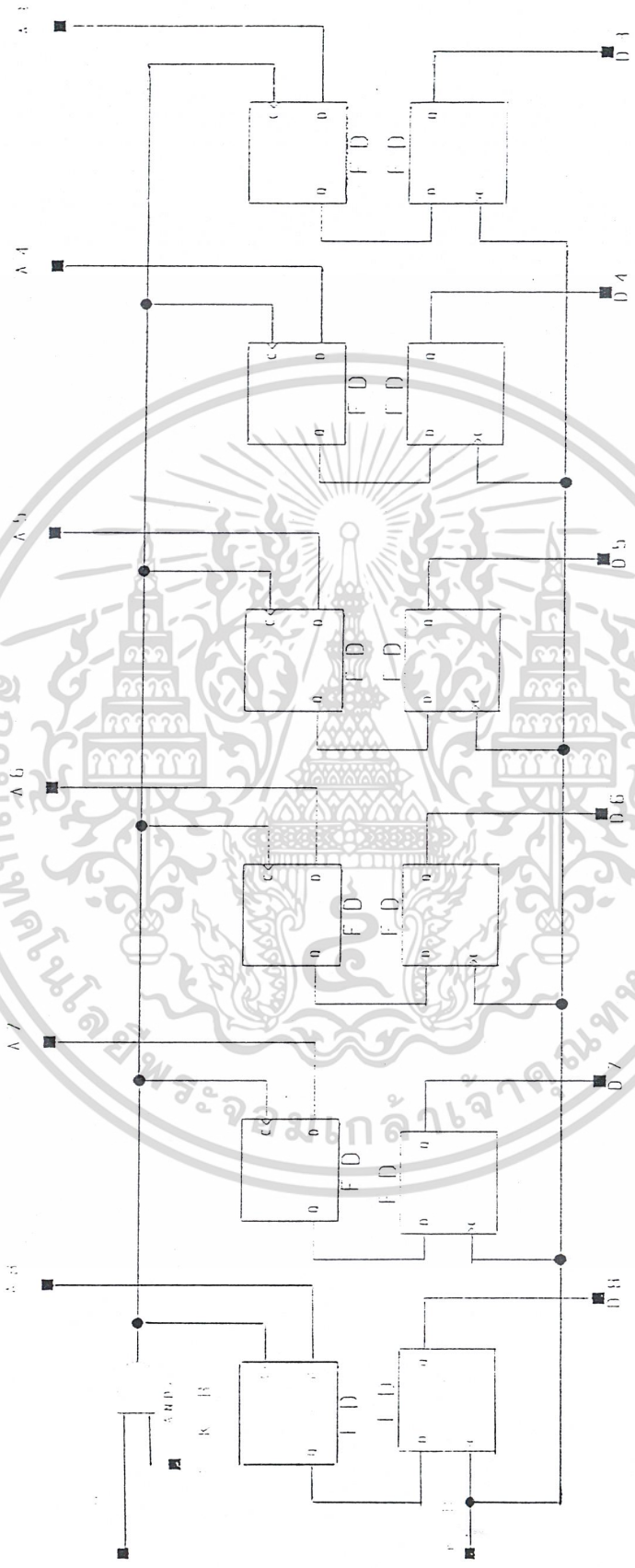
รูปที่ ก.5 วงจร Absolute

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



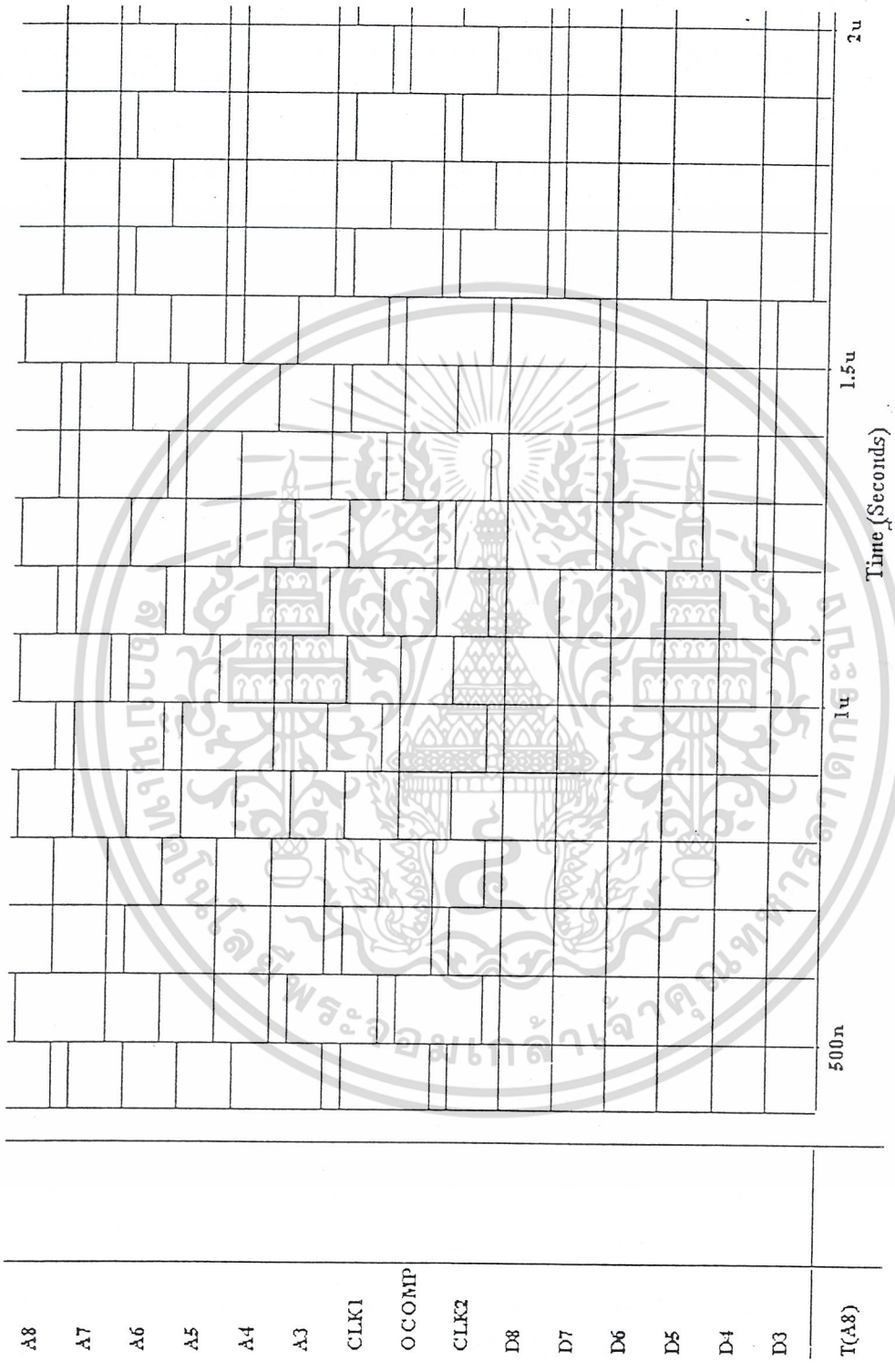
รูปที่ ก.6 ผลการทดสอบวงจร Absolute

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



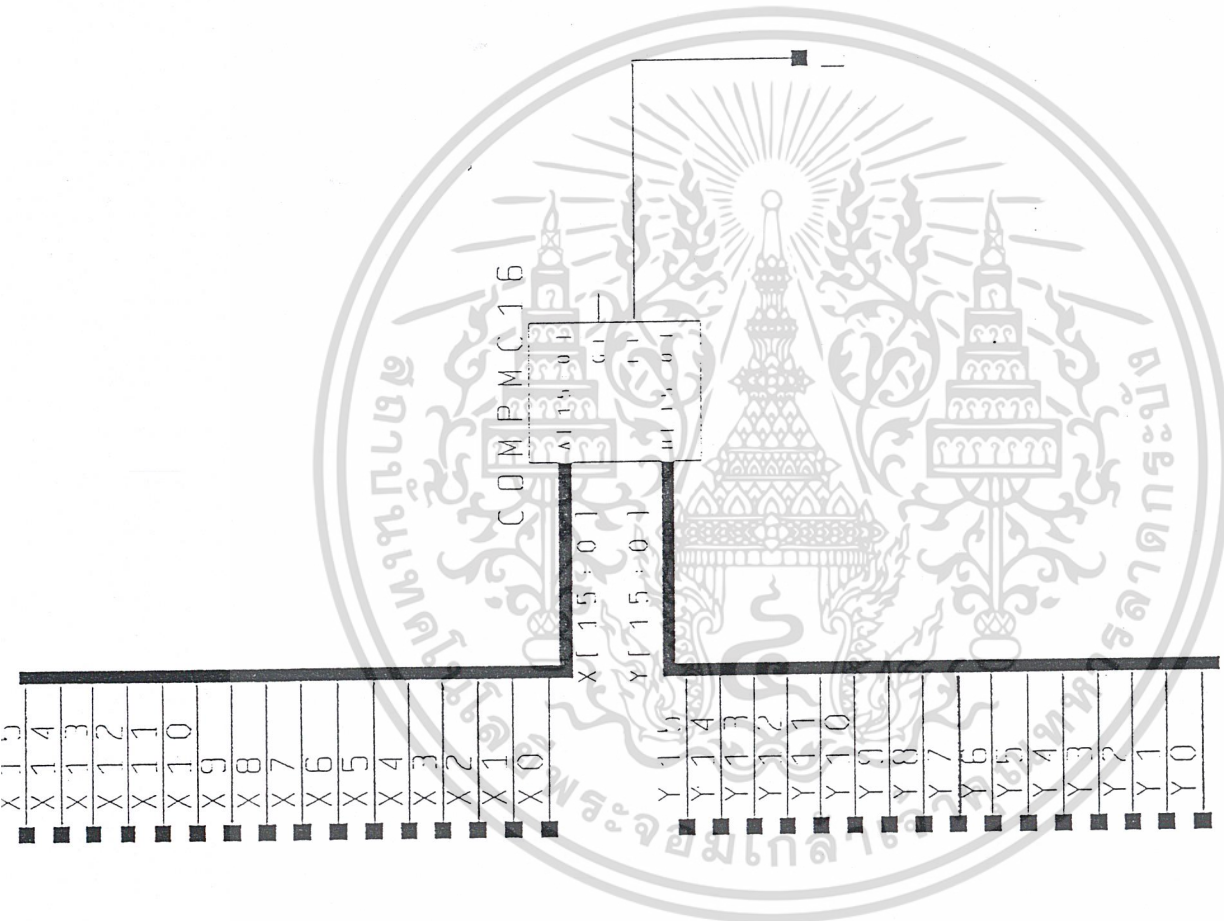
รูปที่ ก.7 วงจรเก็บรีจิสเตอร์ 6 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



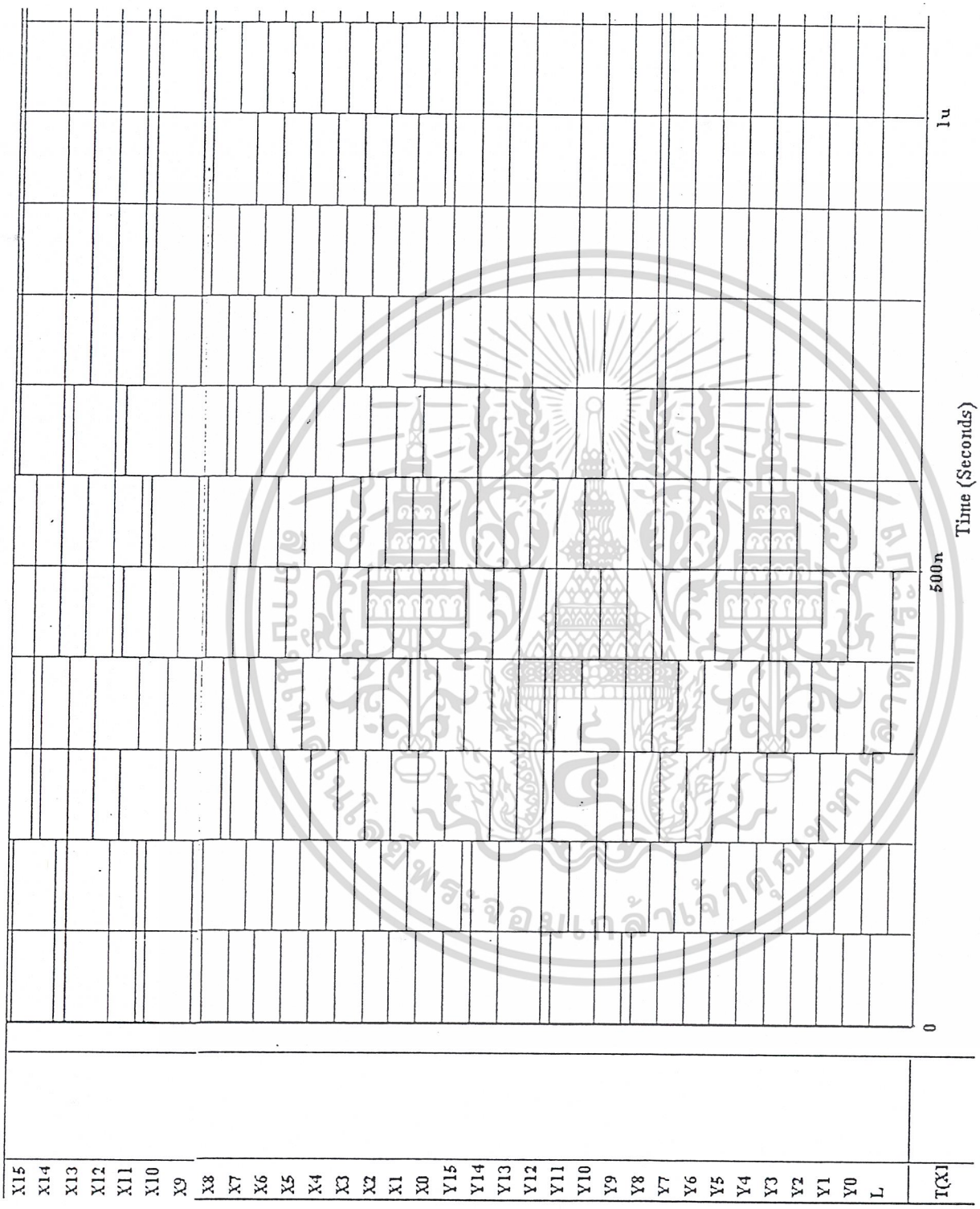
รูปที่ ก.8 ผลการทดสอบวงจรเก็บรีจิสเตอร์ 6 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



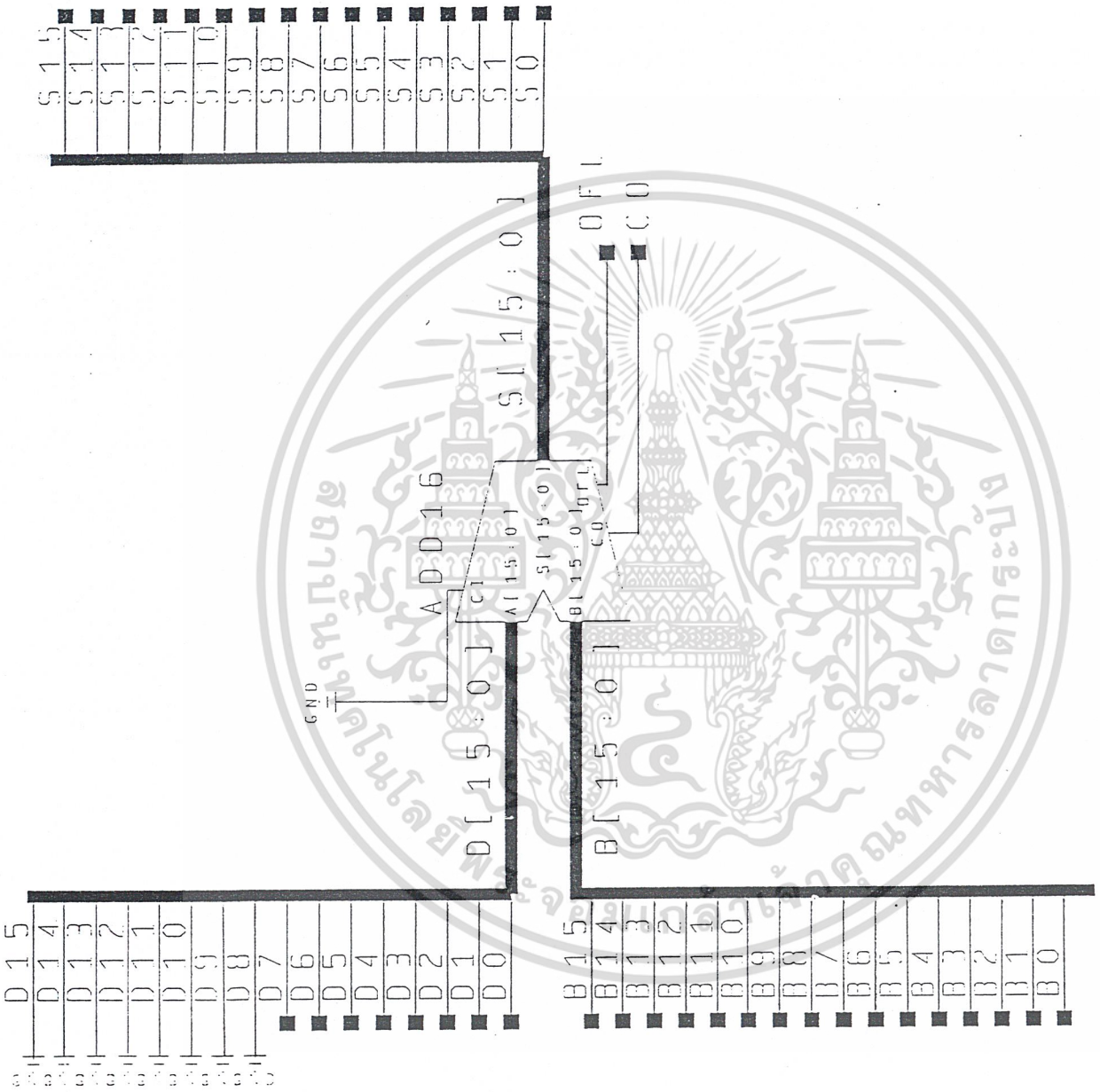
รูปที่ ก.9 วงจร Comparator 16 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



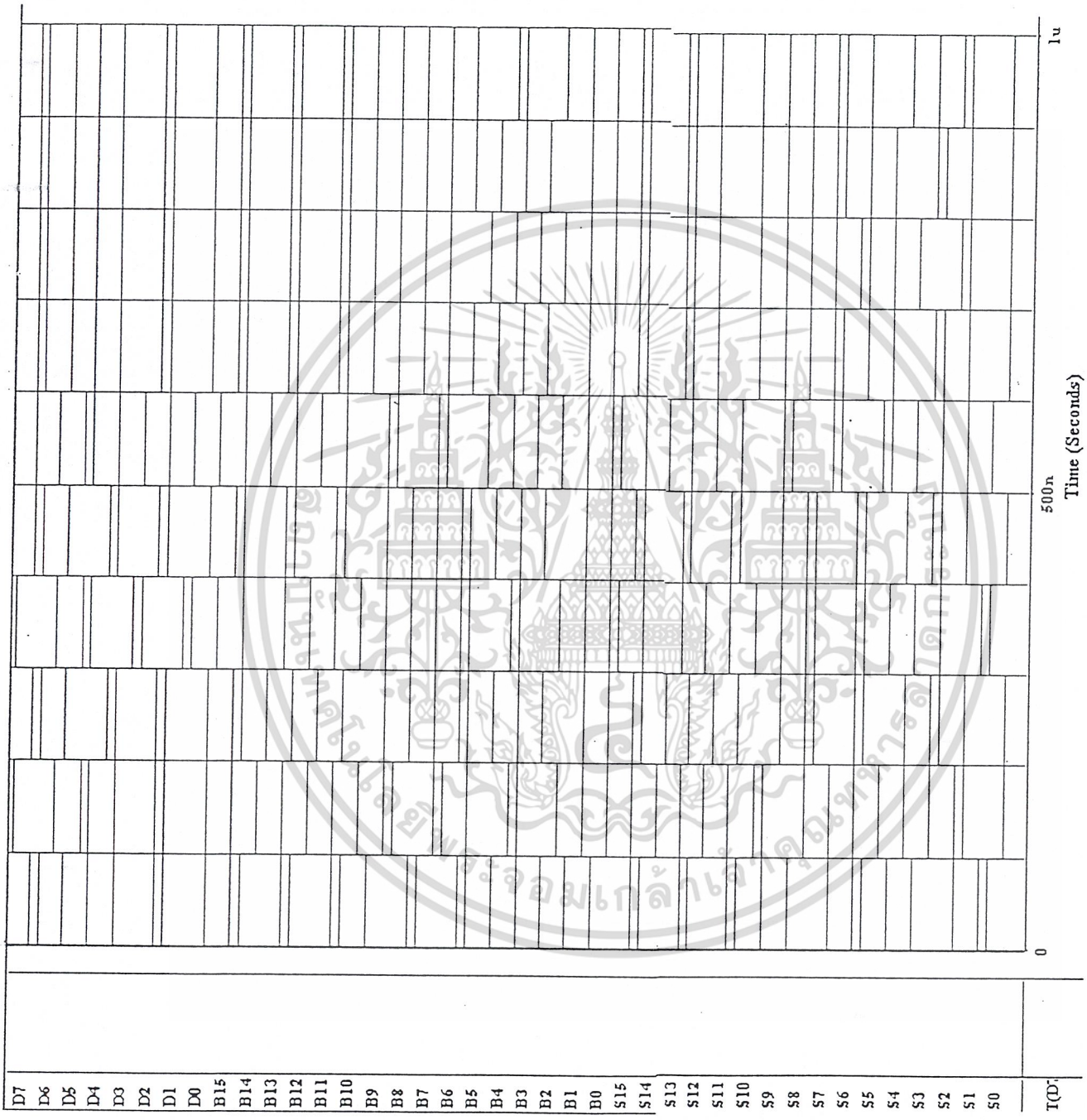
รูปที่ ก.10 ผลการทดสอบวงจร Comparator 16 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.11 วงจร Adder

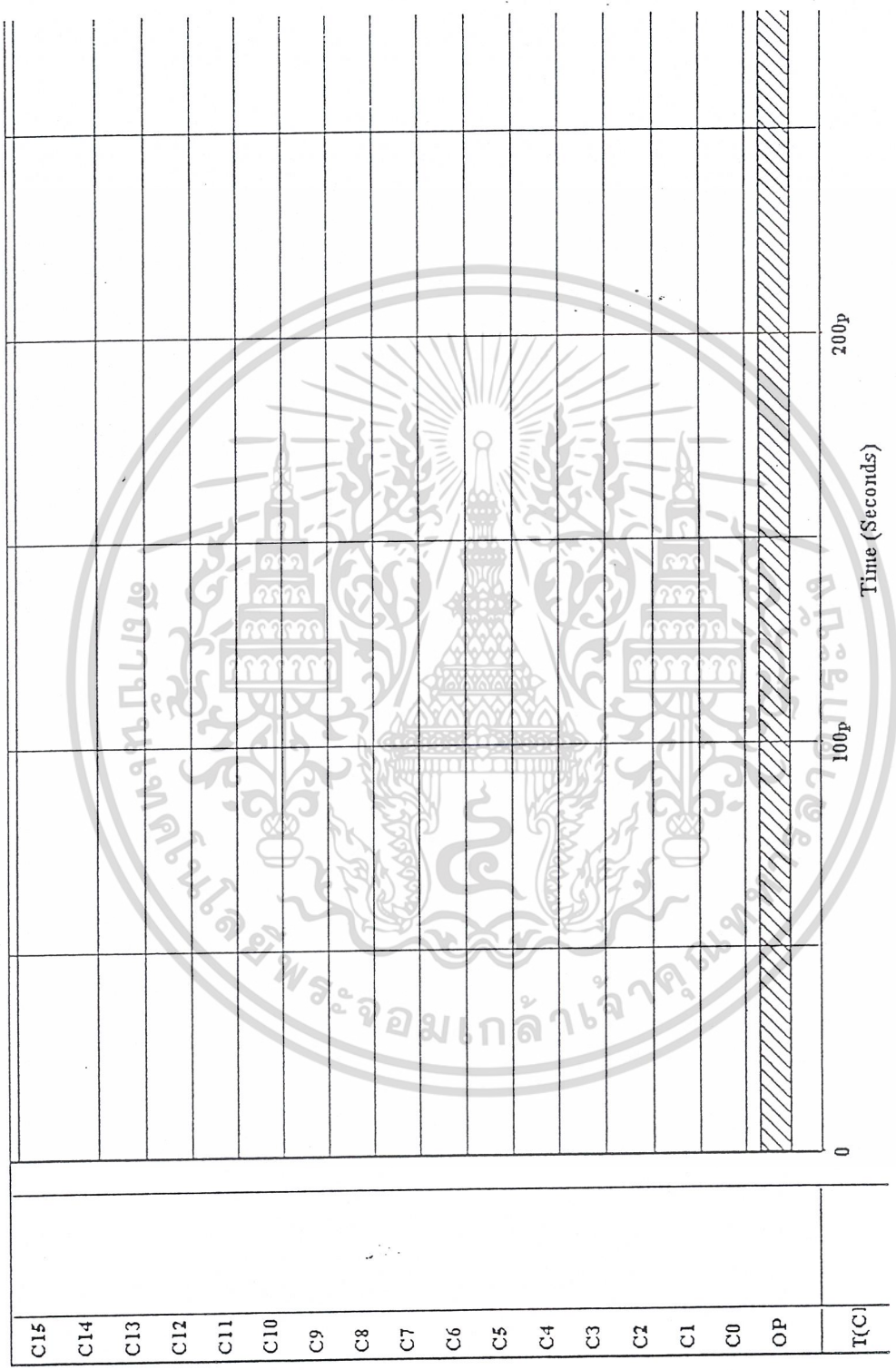
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

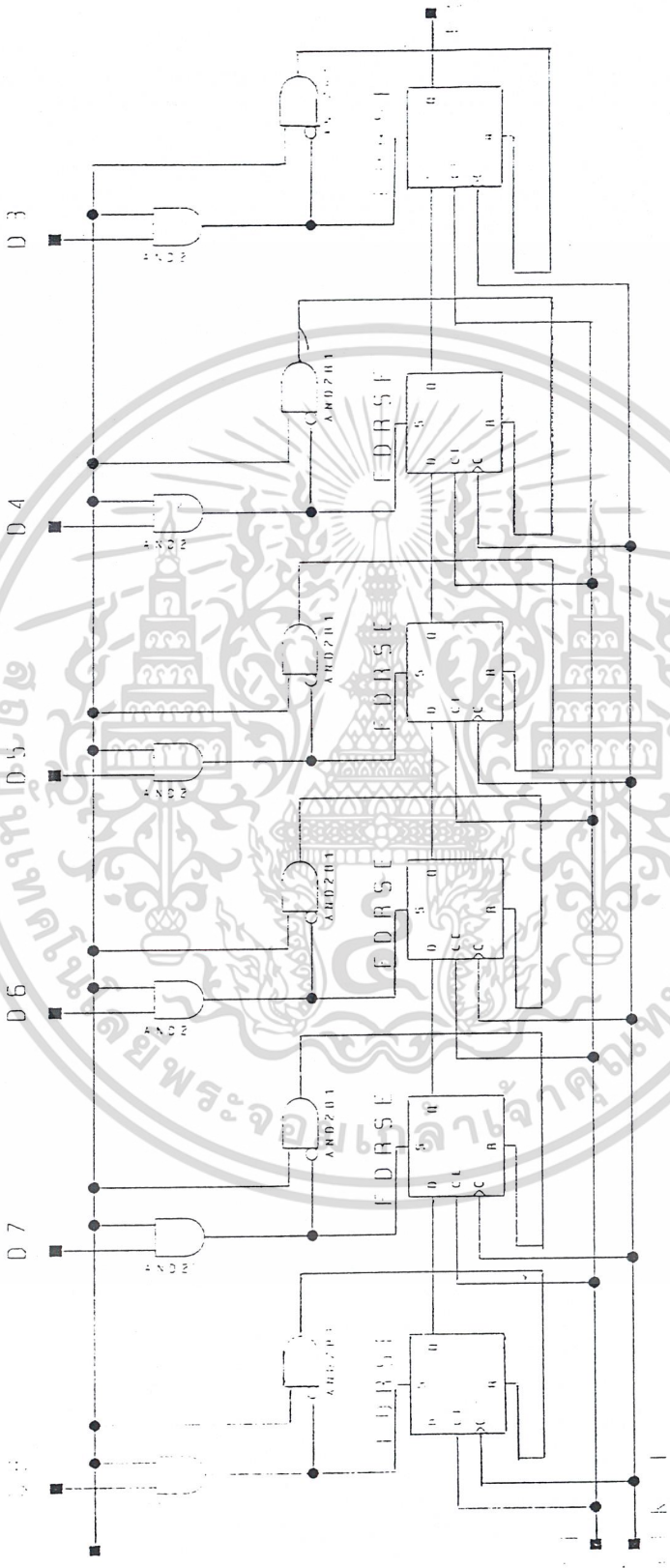
รูปที่ ก.13 วงจร OR 16 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.14 ผลการทดสอบวงจร OR 16 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

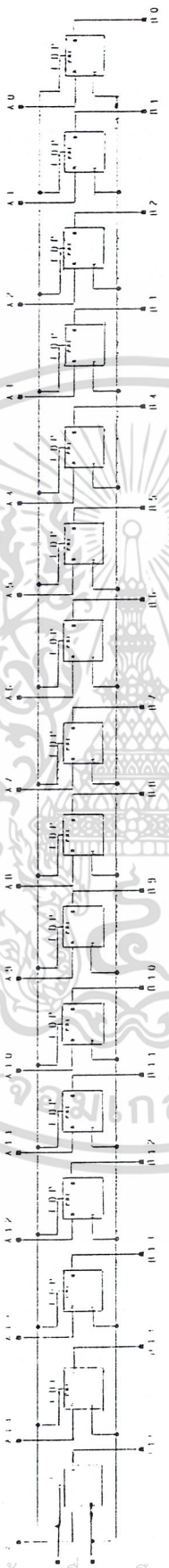


รูปที่ ก.15 วงจรส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

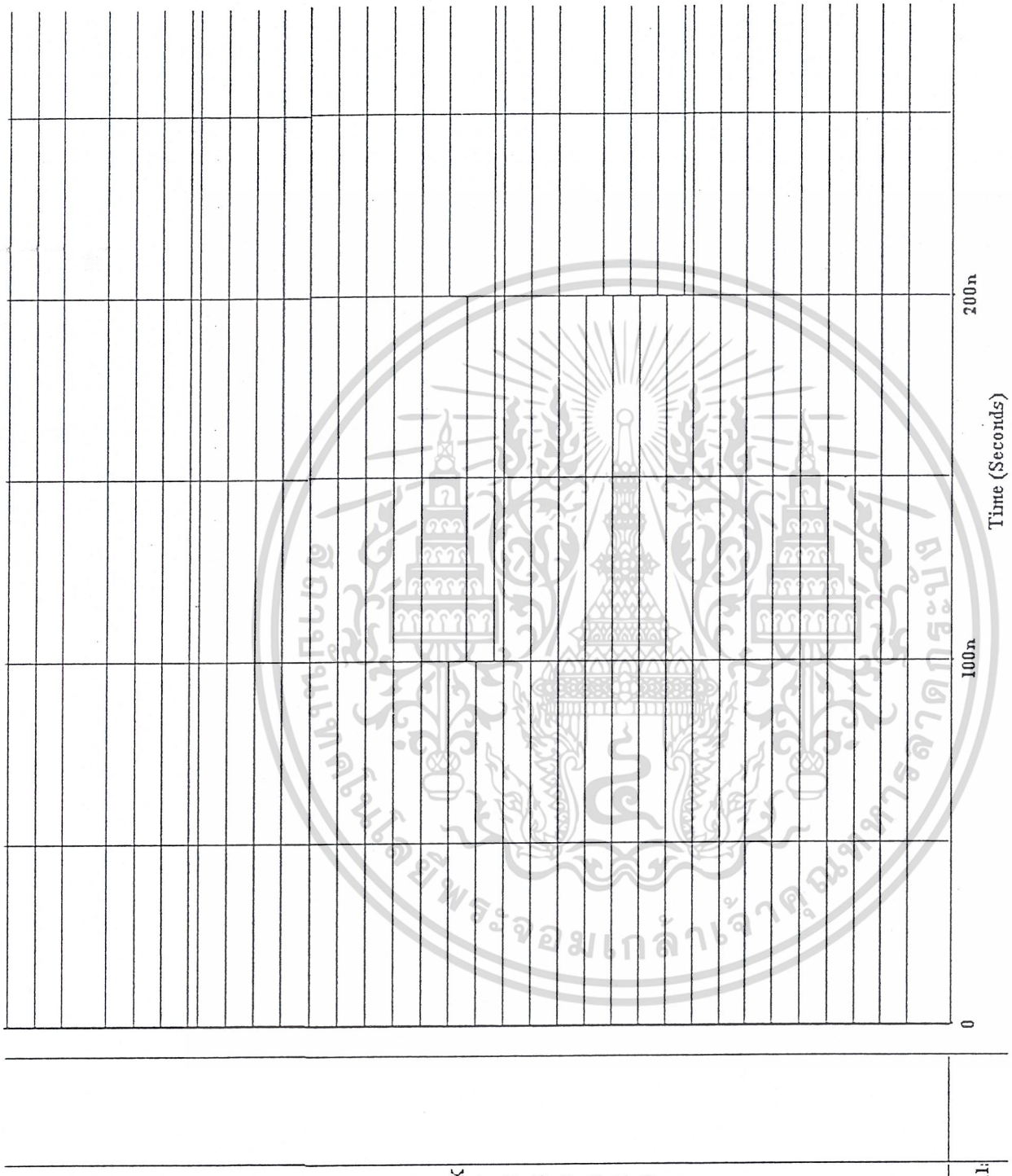
รูปที่ ก.16 ผลการทดสอบวงจรส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



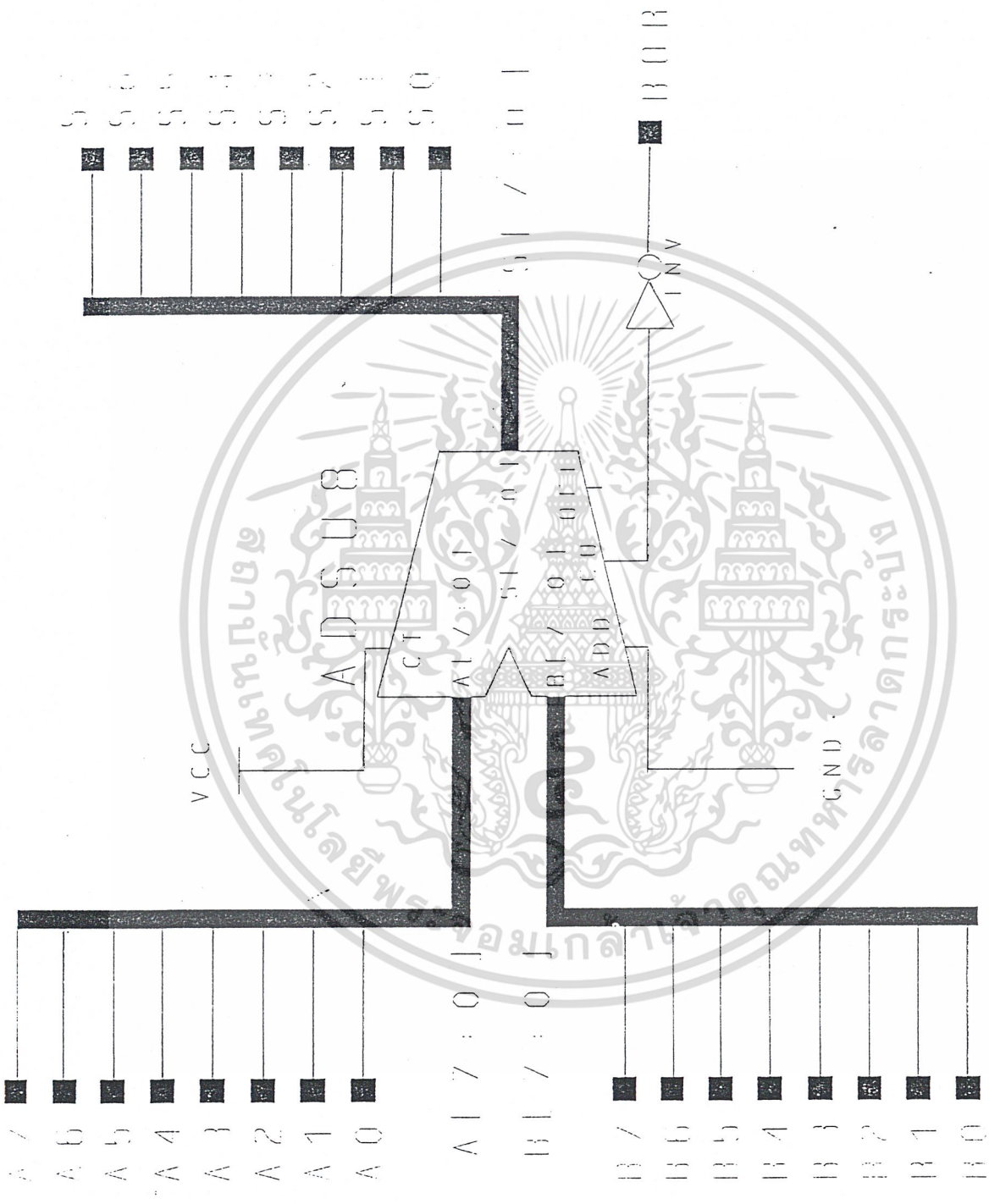
รูปที่ ก.17 วงจร Spec Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.18 ผลการทดสอบวงจร Spec Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

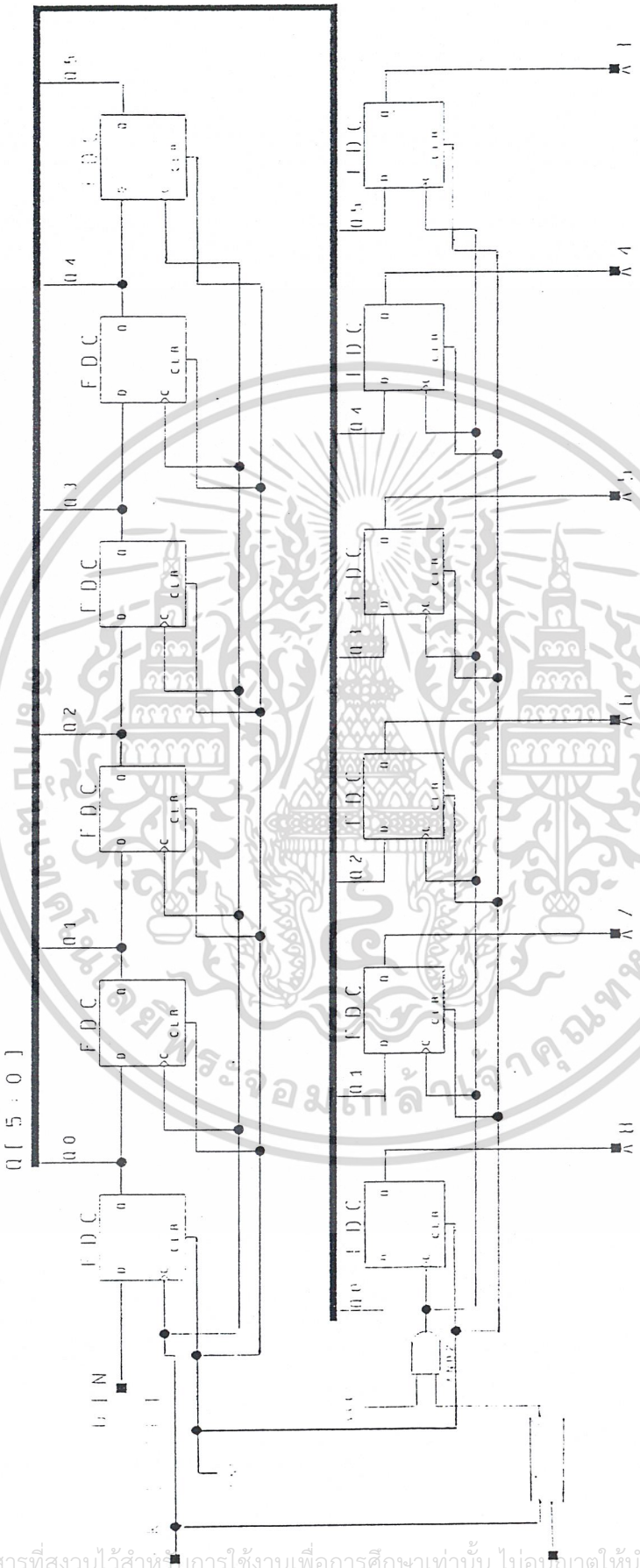


รูปที่ ก.19 วงจรบวก 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

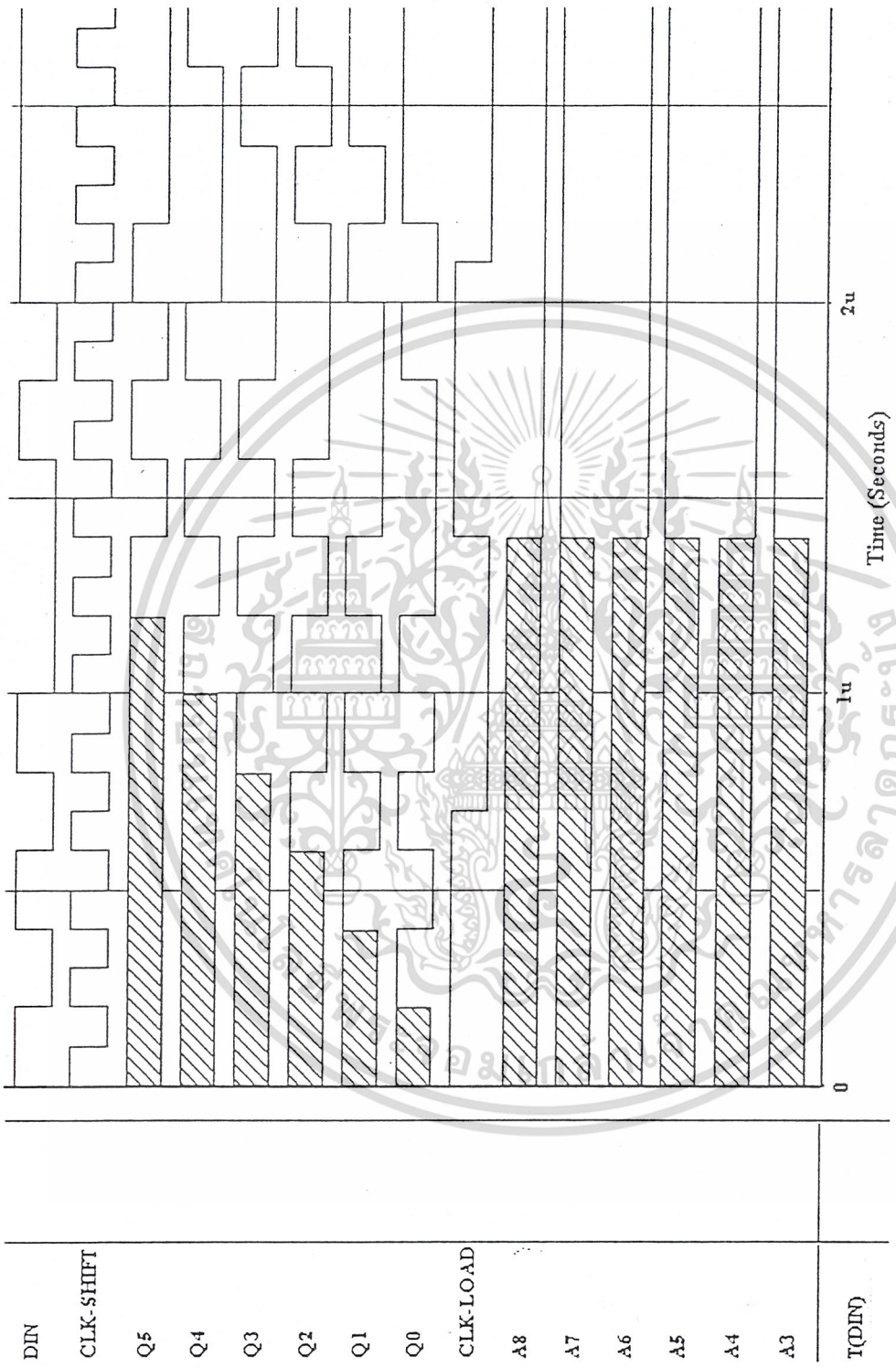
รูปที่ ก.20 ผลการทดสอบวงจรลบ 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.21 วงจรบัฟเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

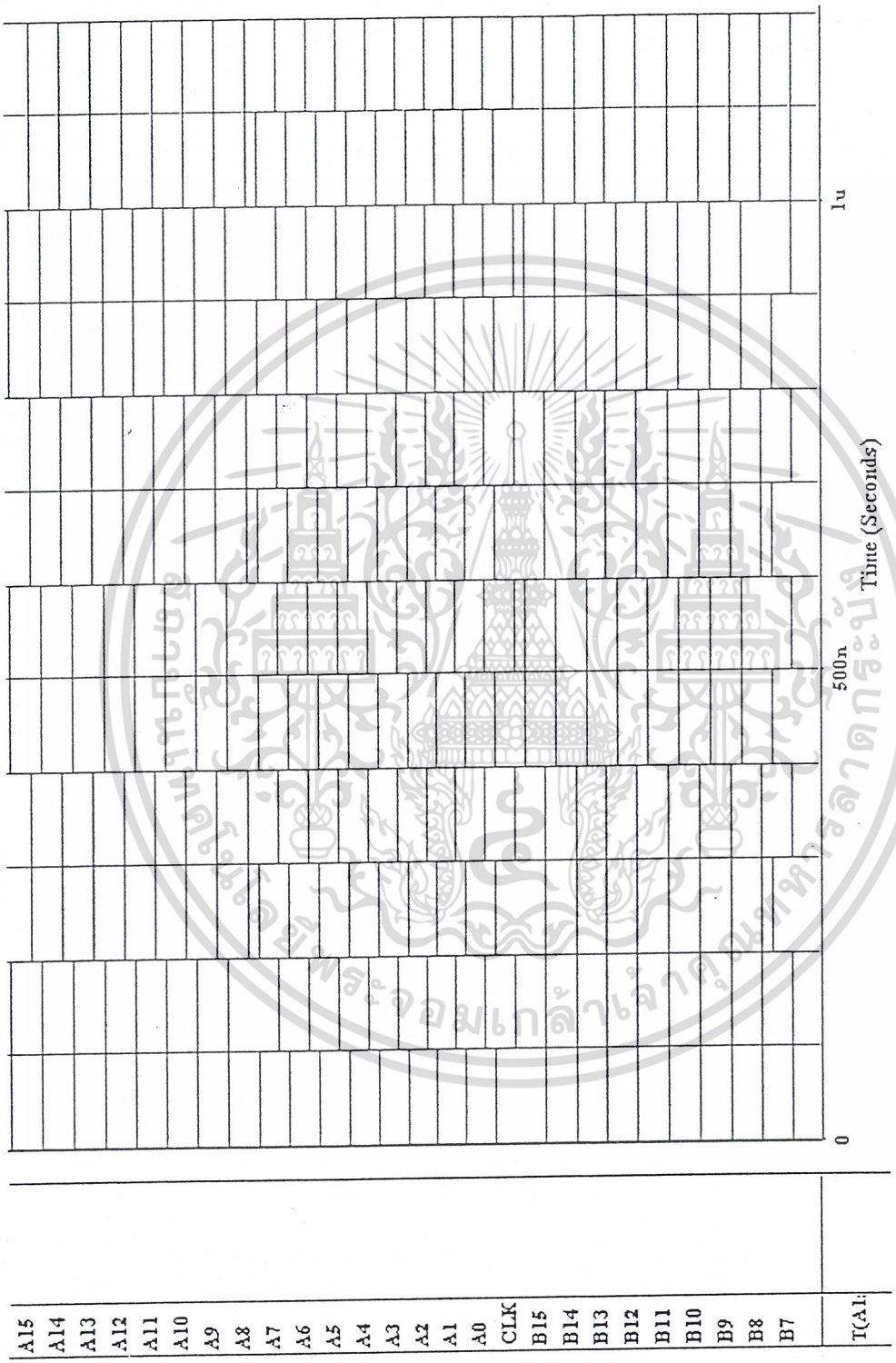


รูปที่ ก.22 ผลการทดสอบวงจรรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

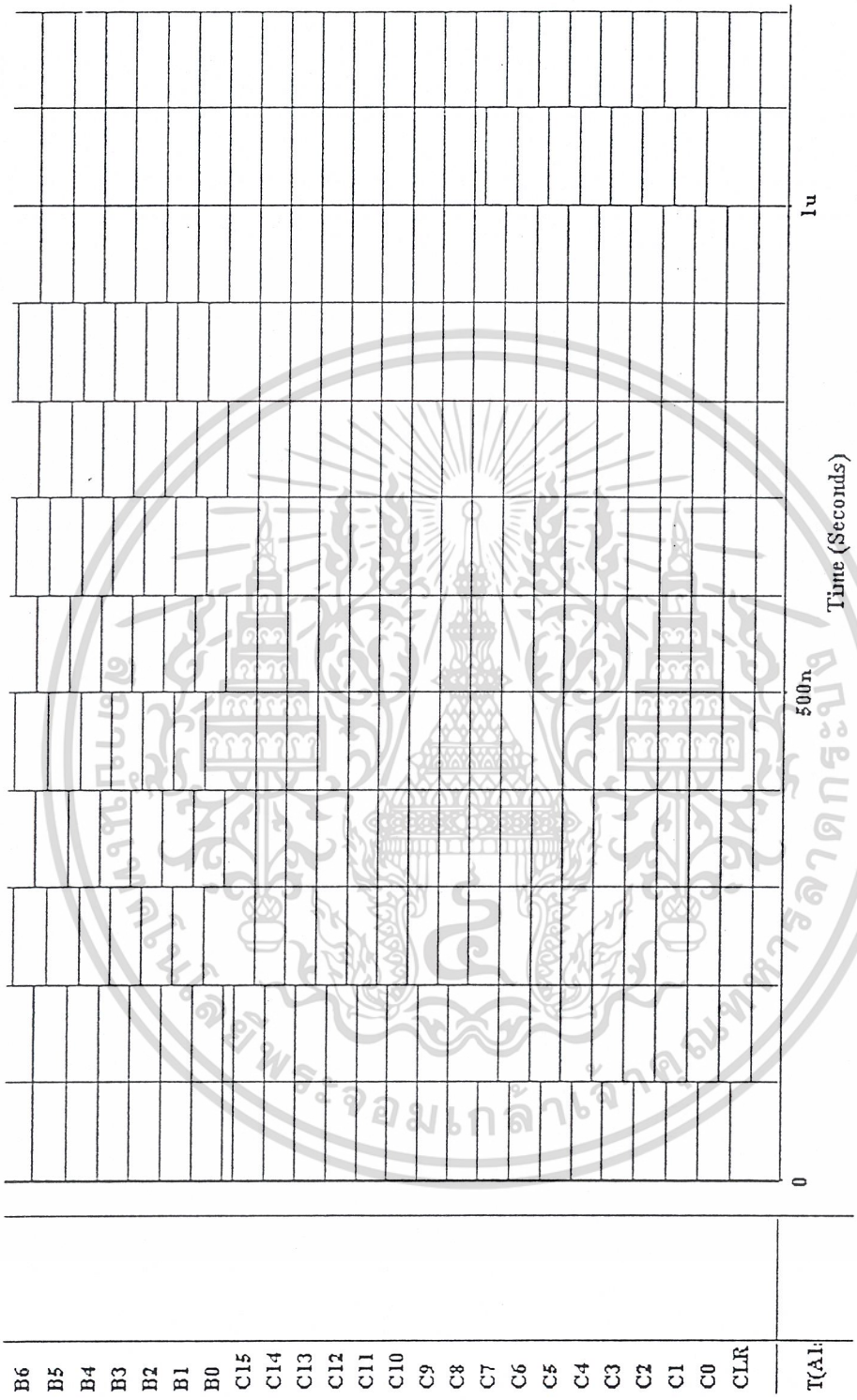
รูปที่ ก.23 วงจร Protector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



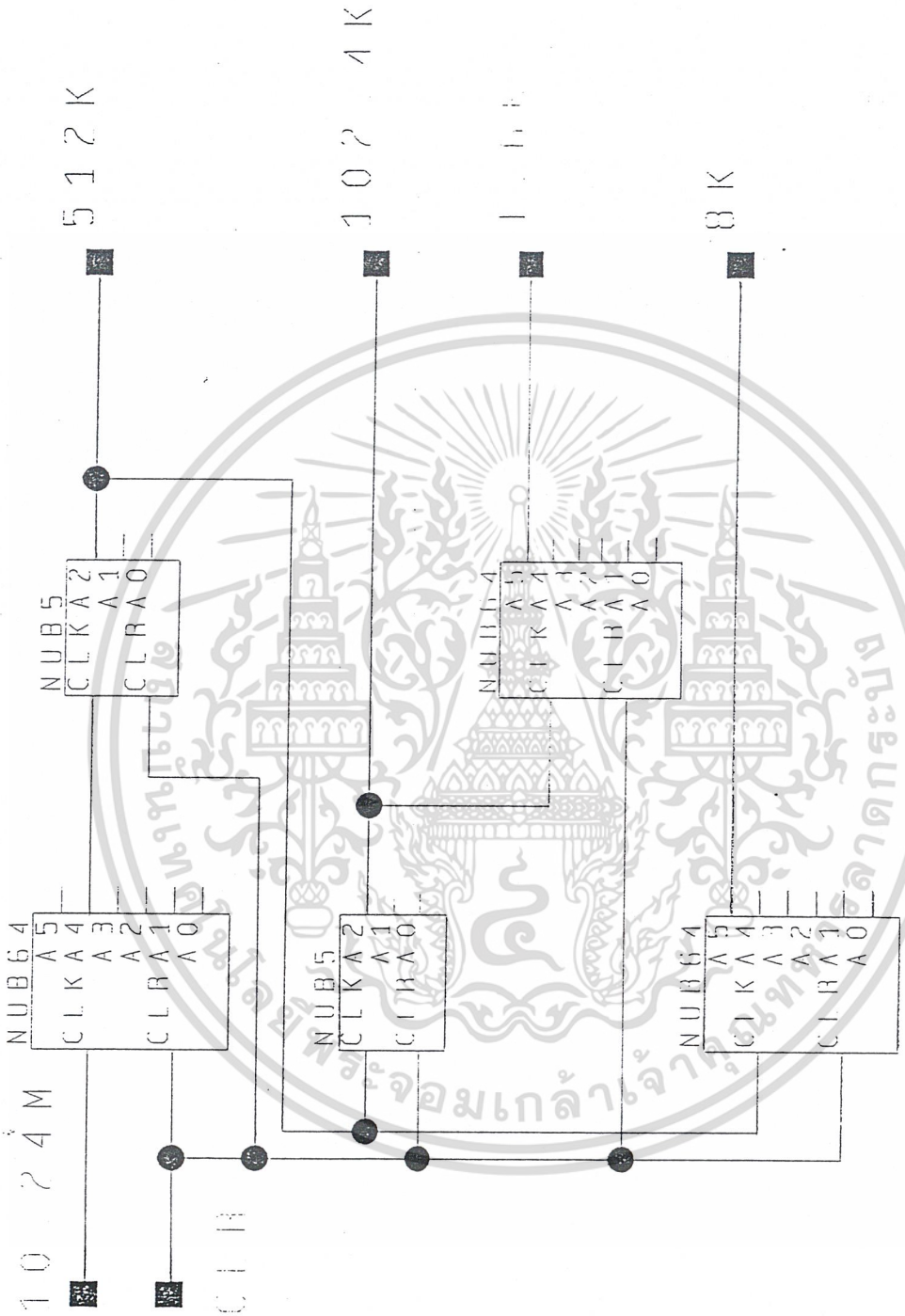
รูปที่ ก.24 ผลการทดสอบวงจร Protector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



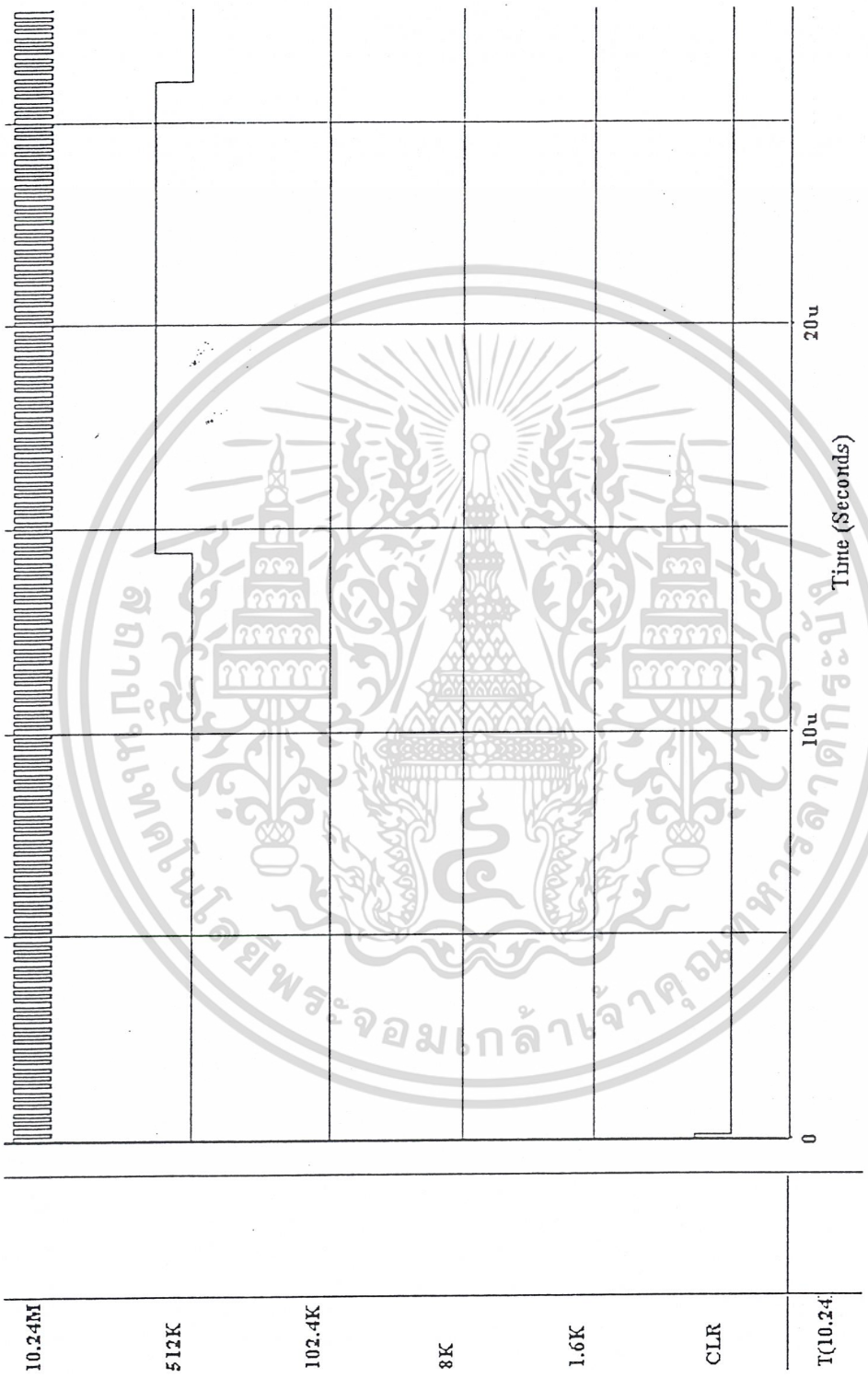
รูปที่ ก.24 (ต่อ) ผลการทดสอบวงจร Protector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



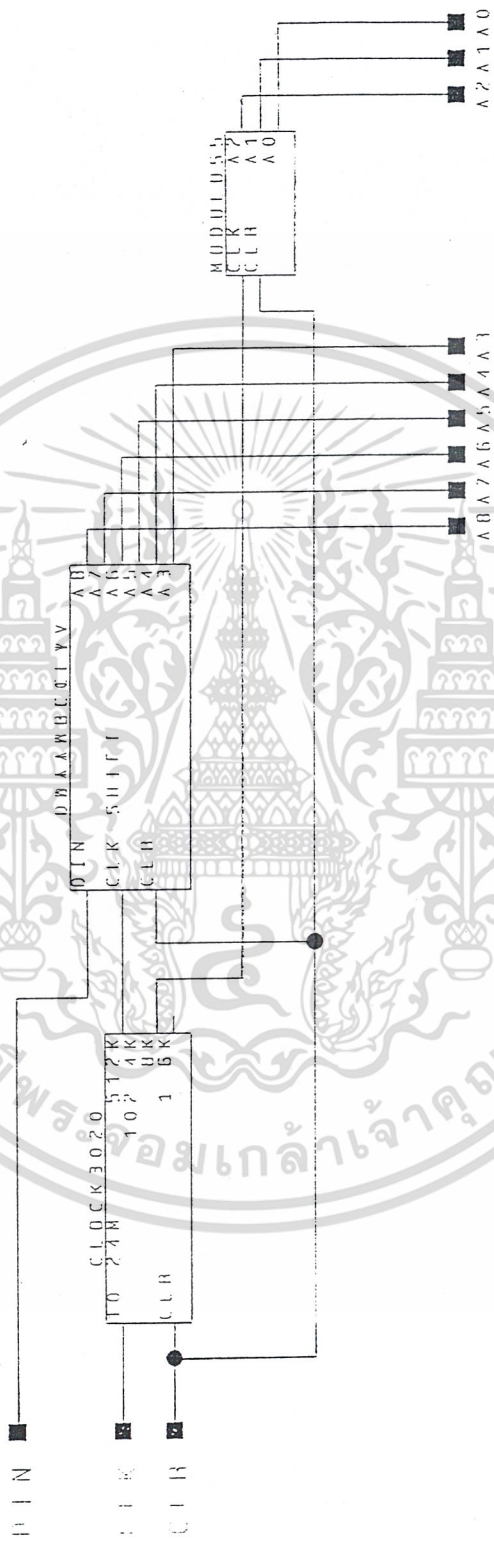
รูปที่ ก.25 วงจรสร้างสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



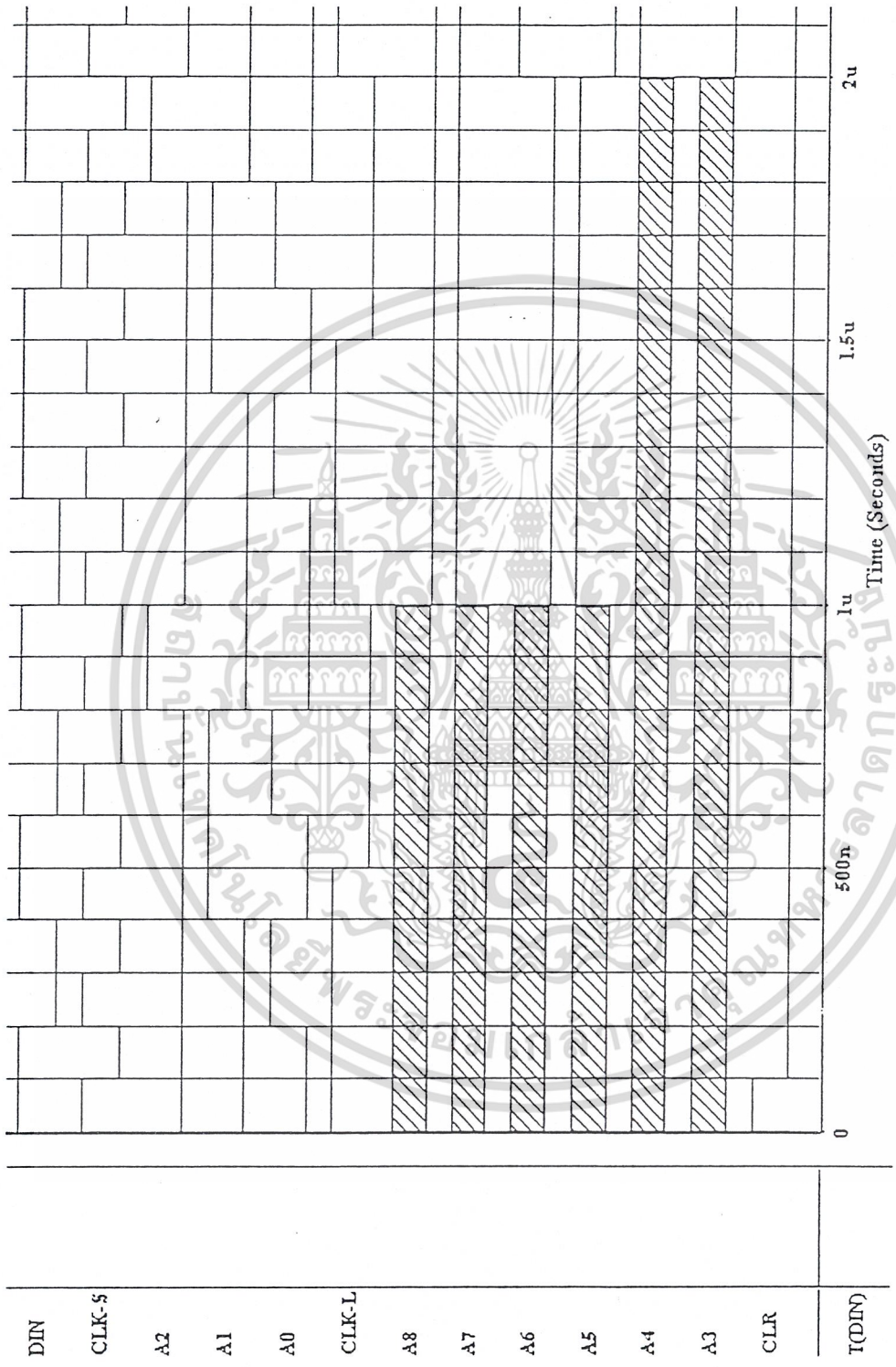
รูปที่ ก.26 ผลการทดสอบวงจรสร้างสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



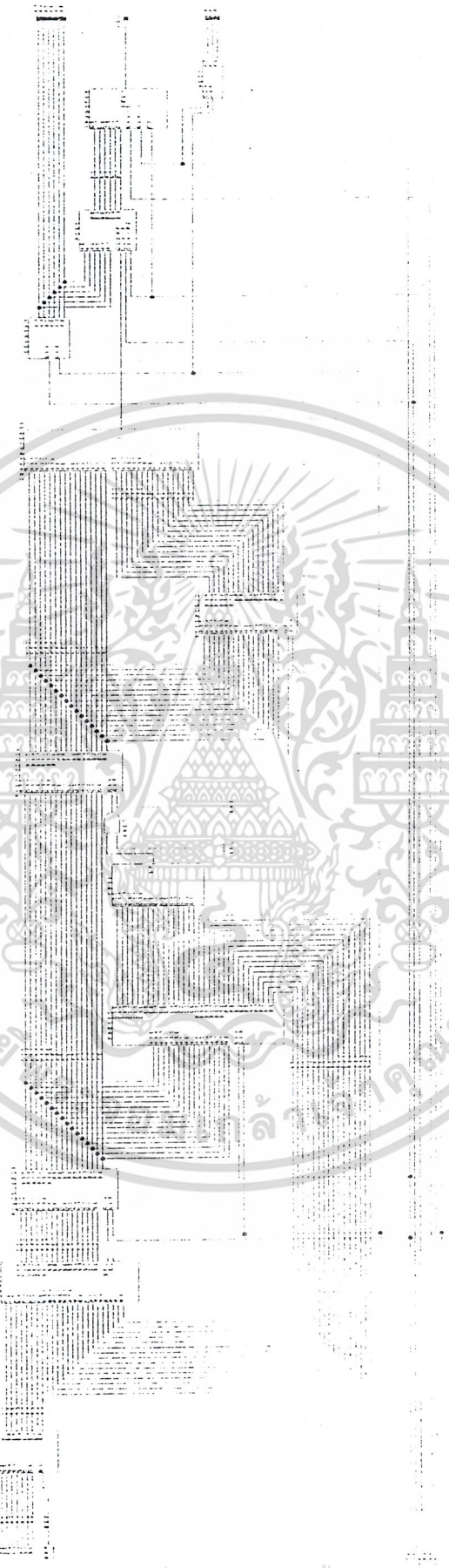
รูปที่ ก.27 วงจรถอดรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



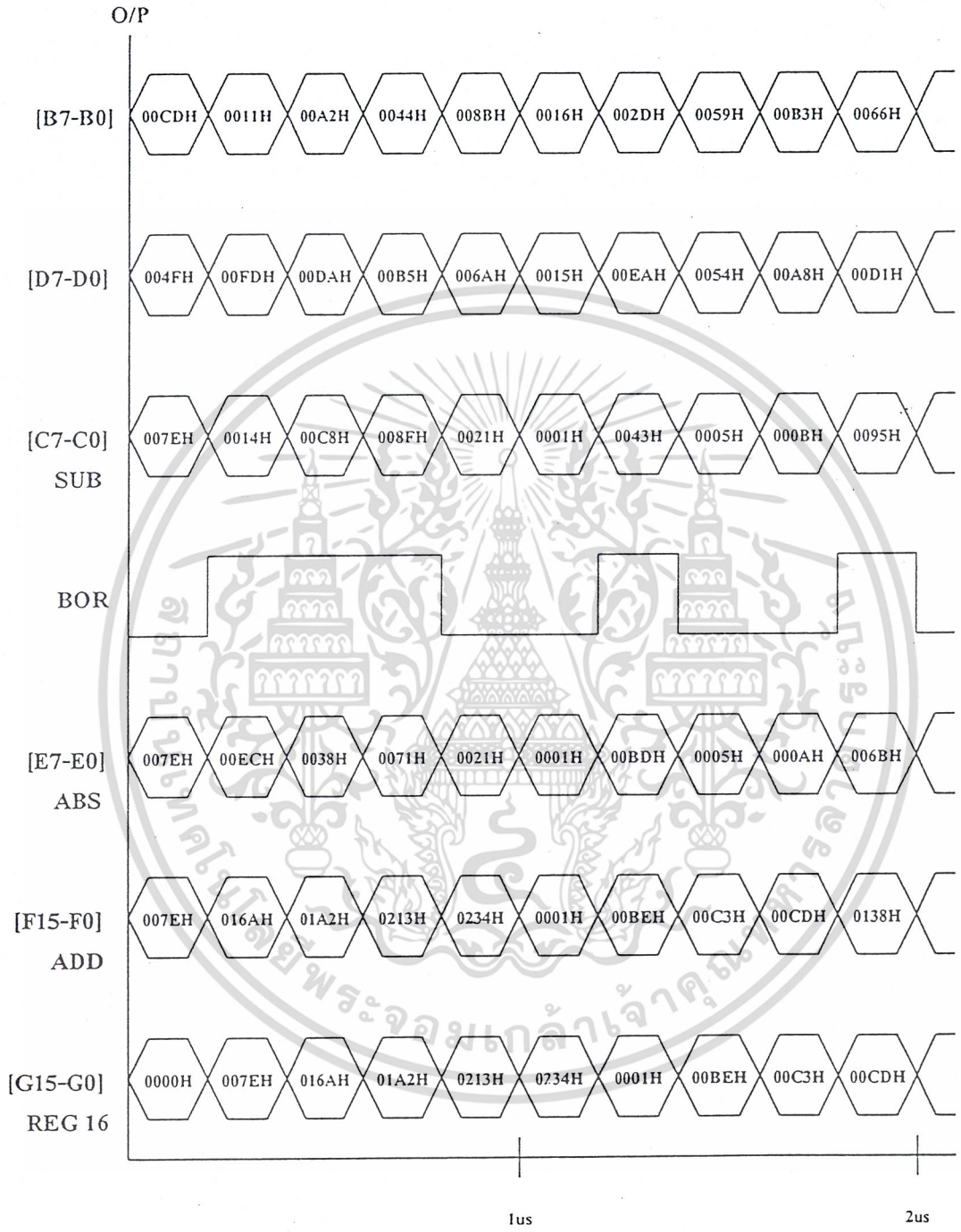
รูปที่ ก.28 ผลการทดสอบวงจรถอดรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



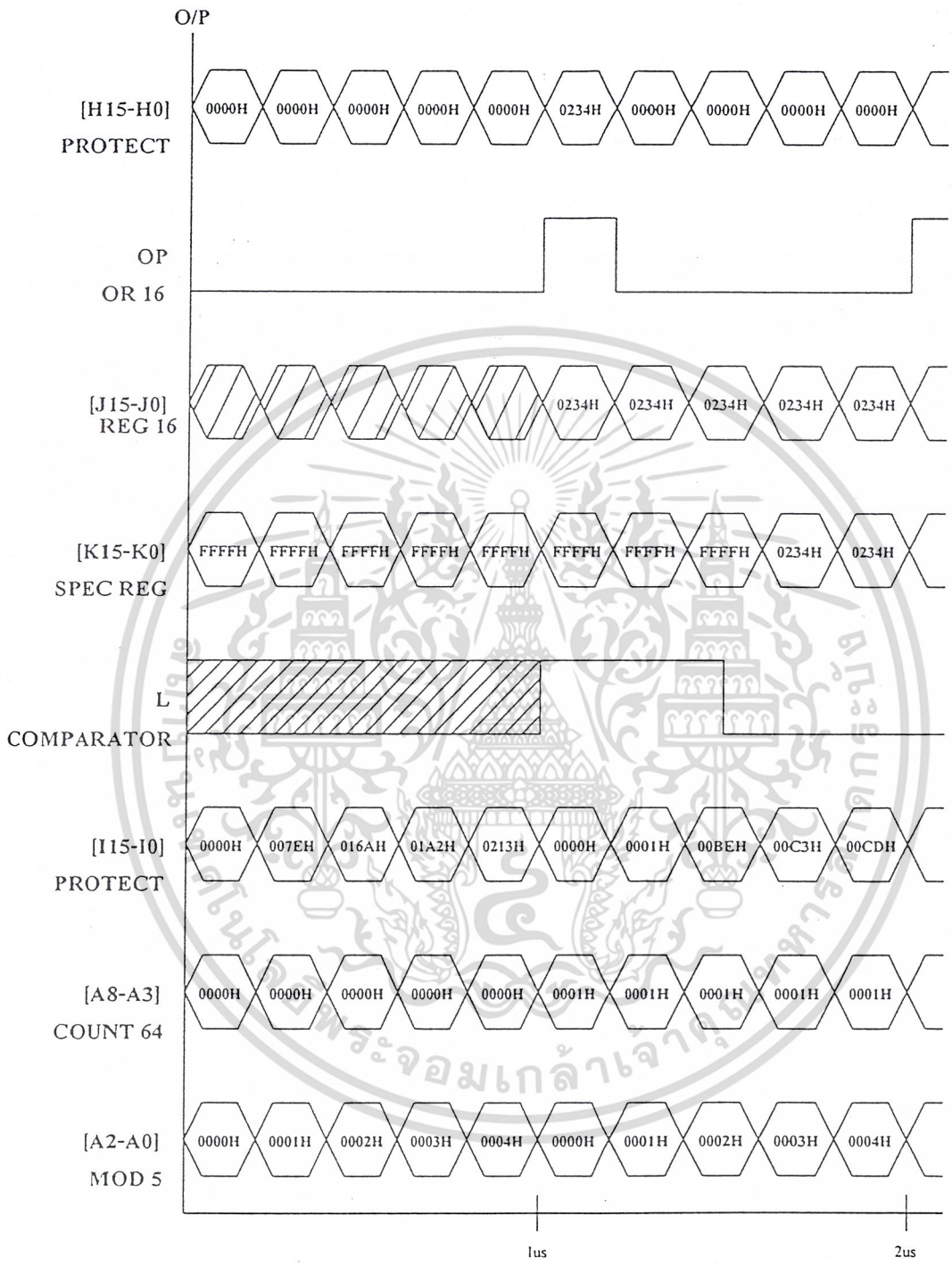
รูปที่ ก.29 วงจรเข้ารหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.30 ผลการทดสอบวงจรเข้ารหัสสัญญาณเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.30 (ต่อ) ผลการทดสอบวงจรเข้ารหัสสัญญาณเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
รายละเอียดของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XILINX

XC3000A
Logic Cell Array Family

Product Specifications

Features

- Enhanced, high performance FPGA family with five device types
  - Improved redesign of the basic XC3000 LCA Family
  - Logic densities from 1,000 to 6,000 gates
  - Up to 144 user-definable I/Os
- Superset of the industry-leading XC3000 family
  - Identical to the basic XC3000 in structure, pin out, design methodology, and software tools
  - 100% compatible with all XC3000, XC3000L, and XC3100 bitstreams
  - Improved routing and additional features
- Additional programmable interconnection points (PIPs)
  - Improved access to longlines and CLB clock enable inputs
  - Most efficient XC3000-class solution to bus-oriented designs
- Advanced 0.8 µ CMOS static memory technology
  - Low quiescent and active power consumption
- Performance specified by logic delays, faster than corresponding XC3000 versions
- XC3000A-specific features
  - 4 mA output sink and source current
  - Error checking of the configuration bitstream
  - Soft startup starts all outputs in slew-limited mode upon power-up
  - Easy migration to the XC3400 series of HardWire mask programmed devices for high-volume production.

Description

The XC3000A family offers the following enhancements over the popular XC3000 family:

The XC3000A family has additional interconnect resources to drive the I-inputs of TBUFs driving horizontal Longlines. The CLB Clock Enable input can be driven from a second vertical Longline. These two additions result in more efficient and faster designs when horizontal Longlines are used for data bussing.

During configuration, the XC3000A devices check the bitstream format for stop bits in the appropriate positions. Any error terminates the configuration and pulls INIT Low.

When the configuration process is finished and the device starts up in user mode, the first activation of the outputs is automatically slew-rate limited . This feature, called Soft Startup, avoids the potential ground bounce when all outputs are turned on simultaneously. After start-up, the slew rate of the individual outputs is, as in the XC3000 family, determined by the individual configuration option.

The XC3000A family is a superset of the XC3000 family. Any bitstream used to configure an XC3000 or XC3100 device configures an XC3000A device exactly the same way.

Device	CLBs	Array	User I/Os Max	Flip-Flops	Horizontal Longlines	Configurable Data Bits
XC3020A	64	8 x 8	64	256	16	14,779
XC3030A	100	10 x 10	80	360	20	22,176
XC3042A	144	12 x 12	96	480	24	30,784
XC3064A	224	16 x 14	120	688	32	46,064
XC3090A	320	16 x 20	144	928	40	64,160

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Functions During Configuration

Configuration Mode <M2:M1:M0>					...	68	84	84	100	100	132	160	175	....	User
SLAVE <1:1:1>	MASTER-SER <0:0:0>	PERIPHERAL <1:0:1>	MASTER-HIGH <1:1:0>	MASTER-LOW <1:0:0>	44 PLCC	PLCC	PLCC	PGA	PQFP	TQFP	PGA	PQFP	PGA	208 PQFP	Operation
PWRDWN (I)	PWRDWN (I)	PWRDWN (I)	PWRDWN (I)	PWRDWN (I)	7	10	12	B2	29	26	A1	159	B2	3	PWRDWN I
VCC	VCC	VCC	VCC	VCC	12	18	22	F3	41	38	C8	20	D9	26	VCC
M1 (HIGH) (I)	M1 (LOW) (I)	M1 (LOW) (I)	M1 (HIGH) (I)	M1 (LOW) (I)	16	25	31	J2	52	49	B13	40	B14	48	RCLK
M0 (HIGH) (I)	M0 (LOW) (I)	M0 (HIGH) (I)	M0 (LOW) (I)	M0 (LOW) (I)	17	26	32	L1	54	51	A14	42	B15	50	RTRIG
M2 (HIGH) (I)	M2 (LOW) (I)	M2 (HIGH) (I)	M2 (HIGH) (I)	M2 (HIGH) (I)	18	27	33	K2	56	53	C13	44	C15	56	I/O
HDC (HIGH)	HDC (HIGH)	HDC (HIGH)	HDC (HIGH)	HDC (HIGH)	19	28	34	K3	57	54	B14	45	E14	57	I/O
LDC (LOW)	LDC (LOW)	LDC (LOW)	LDC (LOW)	LDC (LOW)	20	30	36	L3	59	56	D14	49	D16	61	I/O
INIT*	INIT*	INIT*	INIT*	INIT*	22	34	42	K6	65	62	G14	59	H15	77	I/O
GND	GND	GND	GND	GND	23	35	43	J6	66	63	H12	19	J14	79	GND
					26	43	53	L11	76	73	M13	76	P15	100	GND
RESET (I)	RESET (I)	RESET (I)	RESET (I)	RESET (I)	27	44	54	K10	78	75	P14	78	R15	102	XTL2 OR I
DONE	DONE	DONE	DONE	DONE	28	45	55	J10	80	77	N13	80	R14	107	RESET I
		DATA 7 (I)	DATA 7 (I)	DATA 7 (I)		46	56	K11	81	78	M12	81	N13	109	PROGRAM I
					30	47	57	J11	82	79	P13	82	T14	110	I/O
		DATA 6 (I)	DATA 6 (I)	DATA 6 (I)		48	58	H10	83	80	N11	86	P12	115	XTL1 OR I
		DATA 5 (I)	DATA 5 (I)	DATA 5 (I)		49	60	F10	87	84	M9	92	T11	122	I/O
		CS0 (I)				50	61	G10	88	85	N9	93	R10	123	I/O
		DATA 4 (I)	DATA 4 (I)	DATA 4 (I)		51	62	G11	89	86	N8	98	R9	128	I/O
VCC	VCC	VCC	VCC	VCC	34	52	64	F9	91	88	M8	100	N9	130	I/O
		DATA 3 (I)	DATA 3 (I)	DATA 3 (I)		53	65	F11	92	89	N7	102	P8	132	Vcc
		CS1 (I)				54	66	E11	93	90	P6	103	R8	133	I/O
		DATA 2 (I)	DATA 2 (I)	DATA 2 (I)		55	67	E10	94	91	M6	108	R7	138	I/O
		DATA 1 (I)	DATA 1 (I)	DATA 1 (I)		56	70	D10	98	95	M5	114	R5	145	I/O
		RDY/BUSY	RCLK	RCLK		57	71	C11	99	96	N4	115	P5	146	I/O
DIN (I)	DIN (I)	DATA 0 (I)	DATA 0 (I)	DATA 0 (I)	38	58	72	B11	100	97	N2	119	R3	151	I/O
DOUT	DOUT	DOUT	DOUT	DOUT	39	59	73	C10	1	98	M3	120	N4	152	I/O
CCLK (I)	CCLK(O)	CCLK(O)	CCLK(O)	CCLK(O)	40	60	74	A11	2	99	P1	121	R2	153	CCLK I
		WS (I)	A0	A0		61	75	B10	5	2	M2	124	P2	161	I/O
		CS2 (I)	A1	A1		62	76	B9	6	3	N1	125	M3	162	I/O
			A2	A2		63	77	A10	8	5	L2	128	P1	165	I/O
			A3	A3		64	78	A9	9	6	L1	129	N1	166	I/O
			A15	A15		65	81	B6	12	9	K1	132	M1	172	I/O
			A4	A4		66	82	B7	13	10	J2	133	L2	173	I/O
			A14	A14		67	83	A7	14	11	H1	136	K2	178	I/O
			A5	A5		68	84	C7	15	12	H2	137	K1	179	I/O
GND	GND	GND	GND	GND	1	1	1	C6	16	13	H3	139	J3	182	GND
			A13	A13		2	2	A6	17	14	G2	141	H2	184	I/O
			A6	A6		3	3	A5	18	15	G1	142	H1	185	I/O
			A12	A12		4	4	B5	19	16	F2	147	F2	192	I/O
			A7	A7		5	5	C5	20	17	E1	148	E1	193	I/O
			A11	A11		6	8	A3	23	20	D1	151	D1	199	I/O
			A8	A8		7	9	A2	24	21	D2	152	C1	200	I/O
			A10	A10		8	10	B3	25	22	B1	155	E3	203	I/O
			A9	A9		9	11	A1	25	26	C2	156	C2	204	I/O
															All Other
						X	X	X	X	X					XC3020 PC
					X	X	X	X	X	X					XC3030 PC
							X	X	X	X	X				XC3042 PC
							X**				X				XC3064 PC
							X**					X	X	X	XC3090 PC
							X**					X	X	X	XC3195

Represents a 50-kΩ to 100-kΩ pull-up before and during configuration

* INIT is an open drain output during configuration

(I) Represents an input

** Pin assignmnent for the XC3064/XC3090 and XC3195 differ from those shown. See page 2-138.

*** Peripheral mode and master parallel mode are not supported in the PC44 package. See page 2-135.

**** Pin assignments for the XC3195 PQ208 differ from those shown. See page 2-146.

Pin assignments of PGA Footprint PLCC sockets and PGA packages are not electrically identical.

Generic I/O pins are not shown.

The information on this page is provided as a convenient summary. For detailed pin descriptions, see the preceding two pages

For a detailed description of the configuration modes, see pages 2-190 through 2-200.

For pinout details, see pages 2-136 through 2-146.

Before and during configuration, all outputs that are not used for the configuration process are 3-stated with a 50 kΩ to 100 kΩ pull-up resistor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Descriptions

Permanently Dedicated Pins.

V_{CC}
Two to eight (depending on package type) connections to the positive V supply voltage. All must be connected.

GND
Two to eight (depending on package type) connections to ground. All must be connected.

PWRDWN
A Low on this CMOS-compatible input stops all internal activity, but retains configuration. All flip-flops and latches are reset, all outputs are 3-stated, and all inputs are interpreted as High, independent of their actual level. When PWDWN returns High, the LCA device becomes operational with DONE Low for two cycles of the internal 1-MHz clock.Before and during configuration, PWRDWN must be High. If not used, PWRDWN must be tied to V_{CC}.

RESET
This is an active Low input which has three functions.

Prior to the start of configuration, a Low input will delay the start of the configuration process. An internal circuit senses the application of power and begins a minimal time-out cycle. When the time-out and RESET are complete, the levels of the M lines are sampled and configuration begins.

If RESET is asserted during a configuration, the LCA device is re-initialized and restarts the configuration at the termination of RESET.

If RESET is asserted after configuration is complete, it provides a global asynchronous RESET of all IOB and CLB storage elements of the LCA device.

CCLK
During configuration, Configuration Clock is an output of an LCA device in Master mode or Peripheral mode, but an input in Slave mode. During Readback, CCLK is a clock input for shifting configuration data out of the LCA device

CCLK drives dynamic circuitry inside the LCA device. The Low time may, therefore, not exceed a few microseconds. When used as an input, CCLK must be "parked High". An internal pull-up resistor maintains High when the pin is not being driven.

DONE/PROG (D/P)
DONE is an open-drain output, configurable with or an internal pull-up resistor of 2 to 8 k Ω. At the comple configuration, the LCA device circuitry becomes acti synchronous order; DONE is programmed to go activ one cycle either before or after the outputs go active.

Once configuration is done, a High-to-Low transition pin will cause an initialization of the LCA device and a reconfiguration.

M0/RTRIG
As Mode 0, this input is sampled on power-on to dete the power-on delay (2^{14} cycles if M0 is High, 2^{16} cy M0 is Low). Before the start of configuration, this in again sampled together with M1, M2 to determi configuration mode to be used .

A Low-to-High input transition, after configuration is plete, acts as a Read Trigger and initiates a Readba configuration and storage-element data clocked by C By selecting the appropriate Readback option when erating the bitstream, this operation may be limited single Readback, or be inhibited altogether.

M1/RDATA
As Mode 1, this input and M0, M2 are sampled befor start of configuration to establish the configuration mo be used. If Readback is never used, M1 can be tied d to ground or V_{CC}. If Readback is ever used, M1 mus a 5-kΩ resistor to ground or V_{CC}, to accommodate RDATA output.

As an active-Low Read Data, after configuration is plete, this pin is the output of the Readback data.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

User I/O Pins that can have special functions.

M2
During configuration, this input has a weak pull-up resistor. Together with M0 and M1, it is sampled before the start of configuration to establish the configuration mode to be used. After configuration, this pin is a user-programmable I/O pin.

MOC
During configuration, this output is held at a High level to indicate that configuration is not yet complete. After configuration, this pin is a user-programmable I/O pin.

LDC
During Configuration, this output is held at a Low level to indicate that the configuration is not yet complete. After configuration, this pin is a user-programmable I/O pin. LDC is particularly useful in Master mode as a Low enable for an EPROM, but it must then be programmed as a High after configuration.

INIT
This is an active Low open-drain output with a weak pull-up and is held Low during the power stabilization and internal clearing of the configuration memory. It can be used to indicate status to a configuring microprocessor or, as a wired AND of several slave mode devices, a hold-off signal for a master mode device. After configuration this pin becomes a user-programmable I/O pin.

BCLKIN
This is a direct CMOS level input to the alternate clock buffer (Auxiliary Buffer) in the lower right corner.

XTL1
This user I/O pin can be used to operate as the output of an amplifier driving an external crystal and bias circuitry.

XTL2
This user I/O pin can be used as the input of an amplifier connected to an external crystal and bias circuitry. The I/O block is left unconfigured. The oscillator configuration is activated by routing a net from the oscillator buffer symbol output and by the MakeBits program.

CS0, CS1, CS2, $\overline{WS}$
These four inputs represent a set of signals, three active Low and one active High, that are used to control configuration-data entry in the Peripheral mode. Simultaneous assertion of all four inputs generates a Write to the internal data buffer. The removal of any assertion clocks in the D0-D7 data. In Master-Parallel mode, WS and CS2 are the A0 and A1 outputs. After configuration, these pins are user-programmable I/O pins.

RDY/$\overline{BUSY}$
During Peripheral Parallel mode configuration this pin indicates when the chip is ready for another byte of data to be written to it. After configuration is complete, this pin becomes a user-programmed I/O pin.

$\overline{RCLK}$
During Master Parallel mode configuration, each change on the A0-15 outputs is preceded by a rising edge on $\overline{RCLK}$, a redundant output signal. After configuration is complete, this pin becomes a user-programmed I/O pin.

D0-D7
This set of eight pins represents the parallel configuration byte for the parallel Master and Peripheral modes. After configuration is complete, they are user-programmed I/O pins.

A0-A15
During Master Parallel mode, these 16 pins present an address output for a configuration EPROM. After configuration, they are user-programmable I/O pins.

DIN
During Slave or Master Serial configuration, this pin is used as a serial-data input. In the Master or Peripheral configuration, this is the Data 0 input. After configuration is complete, this pin becomes a user-programmed I/O pin.

DOUT
During configuration this pin is used to output serial-configuration data to the DIN pin of a daisy-chained slave. After configuration is complete, this pin becomes a user-programmed I/O pin.

TCLKIN
This is a direct CMOS-level input to the global clock buffer. This pin can also be configured as a user programmable I/O pin. However, since TCLKIN is the preferred input to the global clock net, and the global clock net should be used as the primary clock source, this pin is usually the clock input to the chip.

Unrestricted User I/O Pins.

I/O
An I/O pin may be programmed by the user to be an Input or an Output pin following configuration. All unrestricted I/O pins, plus the special pins mentioned on the following page, have a weak pull-up resistor of 50 kΩ to 100 kΩ that becomes active as soon as the device powers up, and stays active until the end of configuration.

Before and during configuration, all outputs that are not used for the configuration process are 3-stated with a 50 kΩ to 100 kΩ pull-up resistor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XC3000 Families 68-Pin PLCC, 84-Pin PLCC and PGA Pinouts

XC3000, XC3000A, XC3000L, XC3100 and XC3100A families have identical pinouts

68 PLCC XC3030	68 PLCC XC3020	XC3020 XC3030, XC3042	84 PLCC	84 PGA
10	10	PWRDN	12	B2
11	11	TCLKIN-I/O	13	C2
12	—	I/O*	14	B1
13	12	I/O	15	C1
14	13	I/O	16	D2
—	—	I/O	17	D1
15	14	I/O	18	E3
16	15	I/O	19	E2
—	16	I/O	20	E1
17	17	I/O	21	F2
18	18	VCC	22	F3
19	19	I/O	23	G3
—	—	I/O	24	G1
20	20	I/O	25	G2
—	21	I/O	26	F1
21	22	I/O	27	H1
22	—	I/O	28	H2
23	23	I/O	29	J1
24	24	I/O	30	K1
25	25	M1-RDATA	31	J2
26	26	M0-RTRIG	32	L1
27	27	M2-I/O	33	K2
28	28	HDC-I/O	34	K3
29	29	I/O	35	L2
30	30	LDC-I/O	36	L3
—	31	I/O	37	K4
—		I/O*	38	L4
31	32	I/O	39	J5
32	33	I/O	40	K5
33	—	I/O*	41	L5
34	34	INIT-I/O	42	K6
35	35	GND	43	J6
36	36	I/O	44	J7
37	37	I/O	45	L7
38	38	I/O	46	K7
39	39	I/O	47	L6
—	40	I/O	48	L8
—	41	I/O	49	K8
40		I/O*	50	L9
41		I/O*	51	L10
42	42	I/O	52	K9
43	43	XTL2(IN)-I/O	53	L11

68 PLCC XC3030 XC3020	XC3020 XC3030, XC3042	84 PLCC	84 PGA
44	RESET	54	K10
45	DONE-PG	55	J10
46	D7-I/O	56	K11
47	XTL1(OUT)-BCLKIN-I/O	57	J11
48	D6-I/O	58	H10
—	I/O	59	H11
49	D5-I/O	60	F10
50	CS0-I/O	61	G10
51	D4-I/O	62	G11
—	I/O	63	G9
52	VCC	64	F9
53	D3-I/O	65	F11
54	CS1-I/O	66	E11
55	D2-I/O	67	E10
—	I/O	68	E9
—	I/O*	69	D11
56	D1-I/O	70	D10
57	RDY/BUSY-RCLK-I/O	71	C11
58	D0-DIN-I/O	72	B11
59	DOUT-I/O	73	C10
60	CCLK	74	A11
61	A0-WS-I/O	75	B10
62	A1-CS2-I/O	76	B9
63	A2-I/O	77	A10
64	A3-I/O	78	A9
—	I/O*	79	B8
—	I/O*	80	A8
65	A15-I/O	81	B6
66	A4-I/O	82	B7
67	A14-I/O	83	A7
68	A5-I/O	84	C7
1	GND	1	C6
2	A13-I/O	2	A6
3	A6-I/O	3	A5
4	A12-I/O	4	B5
5	A7-I/O	5	C5
—	I/O*	6	A4
—	I/O*	7	B4
6	A11-I/O	8	A3
7	A8-I/O	9	A2
8	A10-I/O	10	B3
9	A9-I/O	11	A1

Unprogrammed IOBs have a default pull-up. This prevents an undefined pad level for unbonded or unused IOBs. Programmed outputs are default slew-rate limited.

This table describes the pinouts of three different chips in three different packages. The pin-description column lists 84 of the 118 pads on the XC3042 (and 84 of the 98 pads on the XC3030) that are connected to the 84 package pins. Ten pads, indicated by an asterisk, do not exist on the XC3020, which has 74 pads; therefore the corresponding pins on the 84-pin packages have no connections to an XC3020. Six pads on the XC3020 and 16 pads on the XC3030, indicated by a dash (—) in the 68 PLCC column, have no connection to the 68 PLCC, but are connected to the 84-pin packages.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XC3000 Families 68-Pin PLCC, 84-Pin PLCC and PGA Pinouts

XC3000, XC3000A, XC3000L, XC3100 and XC3100A families have identical pinouts

68 PLCC XC3030	68 PLCC XC3020	XC3020 XC3030, XC3042	84 PLCC	84 PGA
10	10	PWRDN	12	B2
11	11	TCLKIN-I/O	13	C2
12	—	I/O*	14	B1
13	12	I/O	15	C1
14	13	I/O	16	D2
—	—	I/O	17	D1
15	14	I/O	18	E3
16	15	I/O	19	E2
—	16	I/O	20	E1
17	17	I/O	21	F2
18	18	VCC	22	F3
19	19	I/O	23	G3
—	—	I/O	24	G1
20	20	I/O	25	G2
—	21	I/O	26	F1
21	22	I/O	27	H1
22	—	I/O	28	H2
23	23	I/O	29	J1
24	24	I/O	30	K1
25	25	M1-RDATA	31	J2
26	26	M0-RTRIG	32	L1
27	27	M2-I/O	33	K2
28	28	HDC-I/O	34	K3
29	29	I/O	35	L2
30	30	LDC-I/O	36	L3
—	31	I/O	37	K4
—		I/O*	38	L4
31	32	I/O	39	J5
32	33	I/O	40	K5
33	—	I/O*	41	L5
34	34	INIT-I/O	42	K6
35	35	GND	43	J6
36	36	I/O	44	J7
37	37	I/O	45	L7
38	38	I/O	46	K7
39	39	I/O	47	L6
—	40	I/O	48	L8
—	41	I/O	49	K8
40		I/O*	50	L9
41		I/O*	51	L10
42	42	I/O	52	K9
43	43	XTL2(IN)-I/O	53	L11

68 PLCC XC3030 XC3020	XC3020 XC3030, XC3042	84 PLCC	84 PGA
44	RESET	54	K10
45	DONE-PG	55	J10
46	D7-I/O	56	K11
47	XTL1(OUT)-BCLKIN-I/O	57	J11
48	D6-I/O	58	H10
—	I/O	59	H11
49	D5-I/O	60	F10
50	CS0-I/O	61	G10
51	D4-I/O	62	G11
—	I/O	63	G9
52	VCC	64	F9
53	D3-I/O	65	F11
54	CS1-I/O	66	E11
55	D2-I/O	67	E10
—	I/O	68	E9
—	I/O*	69	D11
56	D1-I/O	70	D10
57	RDY/BUSY-RCLK-I/O	71	C11
58	D0-DIN-I/O	72	B11
59	DOUT-I/O	73	C10
60	CCLK	74	A11
61	A0-WS-I/O	75	B10
62	A1-CS2-I/O	76	B9
63	A2-I/O	77	A10
64	A3-I/O	78	A9
—	I/O*	79	B8
—	I/O*	80	A8
65	A15-I/O	81	B6
66	A4-I/O	82	B7
67	A14-I/O	83	A7
68	A5-I/O	84	C7
1	GND	1	C6
2	A13-I/O	2	A6
3	A6-I/O	3	A5
4	A12-I/O	4	B5
5	A7-I/O	5	C5
—	I/O*	6	A4
—	I/O*	7	B4
6	A11-I/O	8	A3
7	A8-I/O	9	A2
8	A10-I/O	10	B3
9	A9-I/O	11	A1

Unprogrammed IOBs have a default pull-up. This prevents an undefined pad level for unbonded or unused IOBs. Programmed outputs are default slew-rate limited.

This table describes the pinouts of three different chips in three different packages. The pin-description column lists 84 of the 118 pads on the XC3042 (and 84 of the 98 pads on the XC3030) that are connected to the 84 package pins. Ten pads, indicated by an asterisk, do not exist on the XC3020, which has 74 pads; therefore the corresponding pins on the 84-pin packages have no connections to an XC3020. Six pads on the XC3020 and 16 pads on the XC3030, indicated by a dash (—) in the 68 PLCC column, have no connection to the 68 PLCC, but are connected to the 84-pin packages.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XC4000A
Logic Cell Array Family

Product Specifications

Features

- Third Generation Field-Programmable Gate Arrays
  - Abundant flip-flops
  - Flexible function generators
  - On-chip ultra-fast RAM
  - Dedicated high-speed carry-propagation circuit
  - Wide edge decoders (two per edge)
  - Hierarchy of interconnect lines
  - Internal 3-state bus capability
  - Eight global low-skew clock or signal distribution network
- Flexible Array Architecture
  - Programmable logic blocks and I/O blocks
  - Programmable interconnects and wide decoders
- Sub-micron CMOS Process
  - High-speed logic and Interconnect
  - Low power consumption
- Systems-Oriented Features
  - IEEE 1149.1-compatible boundary-scan logic support
  - Programmable output slew rate (4 modes)
  - Programmable input pull-up or pull-down resistors
  - 24-mA sink current per output (48 per pair)
- Configured by Loading Binary File
  - Unlimited reprogrammability
  - Six programming modes
- XACT Development System runs on '386/'486-type PC, NEC PC, Apollo, Sun-4, and Hewlett-Packard 700 Series
  - Interfaces to popular design environments like Viewlogic, Mentor Graphics and OrCAD
  - Fully automatic partitioning, placement and routing
  - Interactive design editor for design optimization
  - 288 macros, 34 hard macros, RAM/ROM compiler

Description

The XC4000A family of FPGAs offers four devices at the low end of the XC4000 family complexity range. XC4000A differs from XC4000 in four areas: fewer routing resources, fewer wide-edge decoders, higher output sink current, and improved output slew-rate control.

- The XC4000 routing structure is optimized for smaller designs, naturally requiring fewer routing resources. The XC4000A devices have four Longlines and four single-length lines per row and column, while the XC4000 devices have six Longlines and eight single-length lines per row and column. This results in a smaller chip area and lower cost per device.
- XC4000A has two wide-edge decoders on every device edge, while the XC4000 has four. All other wide-decoder features are identical in XC4000 and XC4000A.
- XC4000A outputs are specified at 24 mA, sink current, while XC4000 outputs are specified at 12 mA. The source current is the same 4 mA for both families.
- The XC4000A family offers a more sophisticated output slew-rate control structure with four configurable options for each individual output driver: fast, medium fast, medium slow, and slow. Slew-rate control can alleviate ground-bounce problems when multiple outputs switch simultaneously, and it can reduce or eliminate crosstalk and transmission-line effects on printed circuit boards.

Note that the XC4003 and XC4005 devices are available in both flavors, the lower-priced XC4003A/XC4005A with reduced routing, and the higher-priced XC4003/XC4005 with more abundant routing resources. The XC4000A devices are intended for less demanding and more structured designs, and the XC4000 devices for more random designs requiring additional routing resources.

The equivalent devices are pin-compatible and are available in identical packages, but they are not bitstream compatible. In order to move from a XC4000A to a XC4000, or vice versa, the design must be recompiled.

Table 1. The XC4000A Family of Field-Programmable Gate Arrays

Device	XC4002A	XC4003A	XC4004A	XC4005A
Appr. Gate Count	2,000	3,000	4,000	5,000
CLB Matrix	8 x 8	10 x 10	12 x 12	14 x 14
Number of CLBs	64	100	144	196
Number of Flip-Flops	256	360	480	616
Max Decode Inputs (per side)	24	30	36	42
Max RAM Bits	2,048	3,200	4,608	6,272
Number of IOBs	64	80	96	112

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XC4005 Pinouts

Pin Description	PC 84	PQ 100	PQ 160	PQ 208	PG 156	Bound Scan
VCC	2	92	142	183	H3	-
I/O (A8)	3	93	143	184	H1	44
I/O (A9)	4	94	144	185	G1	47
I/O	-	95	145	186	G2	50
I/O	-	96	146	187	G3	53
-	-	-	-	188*	-	-
-	-	-	-	189*	-	-
I/O (A10)	5	97	147	190	F1	56
I/O (A11)	6	98	148	191	F2	59
I/O	-	-	149	192	E1	62
I/O	-	-	150	193	E2	65
GND	-	-	151	194	F3	-
-	-	-	-	195*	-	-
-	-	-	-	196*	-	-
-	-	-	152*	197*	D1*	-
-	-	-	153*	198*	D2*	-
I/O (A12)	7	99	154	199	E3	68
I/O (A13)	8	100	155	200	C1	71
-	-	-	-	-	-	-
I/O	-	-	156	201	C2	74
I/O	-	-	157	202	D3	77
I/O (A14)	9	1	158	203	B1	80
SGCK1 (A15, I/O)	10	2	159	204	B2	83
VCC	11	3	160	205	C3	-
-	-	-	-	206*	-	-
-	-	-	-	207*	-	-
-	-	-	-	208*	-	-
-	-	-	-	1*	-	-
GND	12	4	1	2	C4	-
-	-	-	-	3*	-	-
PGCK1 (A16, I/O)	13	5	2	4	B3	86
I/O (A17)	14	6	3	5	A1	89
I/O	-	-	4	6	A2	92
I/O	-	-	5	7	C5	95
-	-	-	-	-	-	-
I/O (TDI)	15	7	6	8	B4	98
I/O (TCK)	16	8	7	9	A3	101
-	-	-	8*	10*	A4*	-
-	-	-	9*	11*	-	-
-	-	-	-	12*	-	-
-	-	-	-	13*	-	-
GND	-	-	10	14	C6	-
I/O	-	-	11	15	B5	104
I/O	-	-	12	16	B6	107
I/O (TMS)	17	9	13	17	A5	110
I/O	18	10	14	18	C7	113
-	-	-	-	19*	-	-
-	-	-	-	20*	-	-
I/O	-	-	15	21	B7	116
I/O	-	11	16	22	A6	119
I/O	19	12	17	23	A7	122
I/O	20	13	18	24	A8	125
GND	21	14	19	25	C8	-
VCC	22	15	20	26	B8	-
I/O	23	16	21	27	C9	128
I/O	24	17	22	28	B9	131
I/O	-	18	23	29	A9	134
I/O	-	-	24	30	B10	137
-	-	-	-	31*	-	-
-	-	-	-	32*	-	-
I/O	25	19	25	33	C10	140
I/O	26	20	26	34	A10	143
I/O	-	-	27	35	A11	146
I/O	-	-	28	36	B11	149
GND	-	-	29	37	C11	-
-	-	-	-	38*	-	-
-	-	-	-	39*	-	-
-	-	-	30*	40*	A12*	-
-	-	-	31*	41*	-	-
I/O	27	21	32	42	B12	152
I/O	-	22	33	43	A13	155
I/O	-	-	34	44	A14	158

Pin Description	PC 84	PQ 100	PQ 160	PQ 208	PG 156	Bound Scan
I/O	-	-	35	45	C12	161
-	-	-	-	-	-	-
I/O	28	23	36	46	B13	164
SGCK2 (I/O)	29	24	37	47	B14	167
O (M1)	30	25	38	48	A15	170
GND	31	26	39	49	C13	-
I (M0)	32	27	40	50	A16	173†
-	-	-	-	51*	-	-
-	-	-	-	52*	-	-
-	-	-	-	53*	-	-
-	-	-	-	54*	-	-
VCC	33	28	41	55	C14	-
I (M2)	34	29	42	56	B15	174†
PGCK2 (I/O)	35	30	43	57	B16	175
I/O (HDC)	36	31	44	58	D14	178
I/O	-	-	45	59	C15	181
-	-	-	-	-	-	-
I/O	-	-	46	60	D15	184
I/O	-	32	47	61	E14	187
I/O (LDC)	37	33	48	62	C16	190
-	-	-	49*	63*	E15*	-
-	-	-	50*	64*	D16*	-
-	-	-	-	65*	-	-
-	-	-	-	66*	-	-
GND	-	-	51	67	F14	-
I/O	-	-	52	68	F15	193
I/O	-	-	53	69	E16	196
I/O	38	34	54	70	F16	199
I/O	39	35	55	71	G14	202
-	-	-	-	72*	-	-
-	-	-	-	73*	-	-
I/O	-	36	56	74	G15	205
I/O	-	37	57	75	G16	208
I/O	40	38	58	76	H16	211
I/O (ERR, INIT)	41	39	59	77	H15	214
VCC	42	40	60	78	H14	-
GND	43	41	61	79	J14	-
I/O	44	42	62	80	J15	217
I/O	45	43	63	81	J16	220
I/O	-	44	64	82	K16	223
I/O	-	45	65	83	K15	226
-	-	-	-	84*	-	-
-	-	-	-	85*	-	-
I/O	46	46	66	86	K14	229
I/O	47	47	67	87	L16	232
I/O	-	-	68	88	M16	235
I/O	-	-	69	89	L15	238
GND	-	-	70	90	L14	-
-	-	-	-	91*	-	-
-	-	-	-	92*	-	-
-	-	-	71*	93*	N16*	-
-	-	-	72*	94*	M15*	-
I/O	48	48	73	95	P16	241
I/O	49	49	74	96	M14	244
I/O	-	-	75	97	N15	247
I/O	-	-	76	98	P15	250
I/O	50	50	77	99	N14	253
SGCK3 (I/O)	51	51	78	100	R16	256
GND	52	52	79	101	P14	-
-	-	-	-	102*	-	-
DONE	53	53	80	103	R15	-
-	-	-	-	104*	-	-
-	-	-	-	105*	-	-
VCC	54	54	81	106	P13	-
-	-	-	-	107*	-	-
PROG	55	55	82	108	R14	-
I/O (D7)	56	56	83	109	T16	259
PGCK3 (I/O)	57	57	84	110	T15	262
I/O	-	-	85	111	R13	265
-	-	-	-	-	-	-
I/O	-	-	86	112	P12	268
I/O (D6)	58	58	87	113	T14	271

Pin Description	PC 84	PQ 100	PQ 160	PQ 208	PG 156	Bound Scan
I/O	-	59	88	114	T13	274
-	-	-	89*	115*	R12*	-
-	-	-	89*	115*	R12*	-
-	-	-	90†	116*	T12*	-
-	-	-	-	117*	-	-
-	-	-	-	118*	-	-
GND	-	-	91	119	P11	-
I/O	-	-	92	120	R11	277
I/O	-	-	93	121	T11	280
I/O (D5)	59	60	94	122	T10	283
I/O (CS0)	60	61	95	123	P10	286
-	-	-	-	124*	-	-
-	-	-	-	125*	-	-
I/O	-	62	96	126	R10	289
I/O	-	63	97	127	T9	292
I/O (D4)	61	64	98	128	R9	295
I/O	62	65	99	129	P9	298
VCC	63	66	100	130	R8	-
GND	64	67	101	131	P8	-
I/O (D3)	65	68	102	132	T8	301
I/O (RS)	66	69	103	133	T7	304
I/O	-	70	104	134	T6	307
I/O	-	-	105	135	R7	310
-	-	-	-	136*	-	-
-	-	-	-	137*	-	-
I/O (D2)	67	71	106	138	P7	313
I/O	68	72	107	139	T5	316
I/O	-	-	108	140	R6	319
I/O	-	-	109	141	T4	322
GND	-	-	110	142	P6	-
-	-	-	-	143*	-	-
-	-	-	-	144*	-	-
-	-	-	111*	145*	R5*	-
-	-	-	112*	146*	-	-
I/O (D1)	69	73	113	147	T3	325
I/O (RCLK-BUSY/RDY)	70	74	114	148	P5	328
I/O	-	-	115	149	R4	331
-	-	-	-	-	-	-
I/O	-	-	116	150	R3	334
I/O (D0, DIN)	71	75	117	151	P4	337
SGCK4 (DOUT, I/O)	72	76	118	152	T2	340
CCLK	73	77	119	153	R2	-
VCC	74	78	120	154	P3	-
-	-	-	-	155*	-	-
-	-	-	-	156*	-	-
-	-	-	-	157*	-	-
-	-	-	-	158*	-	-
O (TDO)	75	79	121	159	T1	-
GND	76	80	122	160	N3	-
I/O (A0, WS)	77	81	123	161	R1	2
PGCK4 (A1, I/O)	78	82	124	162	P2	5
I/O	-	-	125	163	N2	8
-	-	-	-	-	-	-
I/O	-	-	126	164	M3	11
I/O (CS1, A2)	79	83	127	165	P1	14
I/O (A3)	80	84	128	166	N1	17
-	-	-	129*	167*	M2*	-
-	-	-	130*	168*	M1*	-
-	-	-	-	169*	-	-
-	-	-	-	170*	-	-
GND	-	-	131	171	L3	-
I/O	-	-	132	172	L2	20
I/O	-	-	133	173	L1	23
I/O (A4)	81	85	134	174	K3	26
I/O (A5)	82	86	135	175	K2	29
-	-	-	-	176*	-	-
-	-	-	136*	177*	-	-
I/O	-	87	137	178	K1	32
I/O	-	88	138	179	J1	35
I/O (A6)	83	89	139	180	J2	38
I/O (A7)	84	90	140	181	J3	41
GND	1	91	141	182	H2	-

* Indicates unconnected package pins.

† Contributes only one bit (.I) to the boundary scan register.

Boundary Scan Bit 0 = TDO.T
Boundary Scan Bit 1 = TDO.O
Boundary Scan Bit 343 = BSCANT.UPD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Descriptions

Permanently Dedicated Pins

V_{CC}

Eight or more (depending on package type) connections to the nominal +5 V supply voltage. All must be connected.

GND

Eight or more (depending on package type) connections to ground. All must be connected.

CCLK

During configuration, Configuration Clock is an output of the LCA in Master modes or asynchronous Peripheral mode, but is an input to the LCA in Slave mode and Synchronous Peripheral mode.

After configuration, CCLK has a weak pull-up resistor and can be selected as Readback Clock.

DONE

This is a bidirectional signal, configurable with or without a pull-up resistor of 2 to 8 kΩ.

As an output, it indicates the completion of the configuration process. The configuration program determines the exact timing, the clock source for the Low-to-High transition, and enable of the pull-up resistor.

As an input, a Low level on DONE can be configured to delay the global logic initialization or the enabling of outputs

$\overline{\text{PROGRAM}}$

This is an active Low input that forces the LCA to clear its configuration memory.

When $\overline{\text{PROGRAM}}$ goes High, the LCA finishes the current clear cycle and executes another complete clear cycle, before it goes into a WAIT state and releases $\overline{\text{INIT}}$.

User I/O Pins that can have Special Functi

RDY/$\overline{\text{BUSY}}$

During peripheral modes, this pin indicates appropriate to write another byte of data int device. The same status is also available on t chronous peripheral mode, if a read operat formed when the device is selected. After cor this is a user-programmable I/O pin.

$\overline{\text{RCLK}}$

During Master Parallel configuration, each cha A0-15 outputs is preceded by a rising edge o redundant output signal. After configuration, thi programmable I/O pin.

M0, M1, M2

As Mode inputs, these pins are sampled before configuration to determine the configuration m used.

After configuration, M0 and M2 can be used as ir M1 can be used as a 3-state output. These three no associated input or output registers.

These pins can be user inputs or outputs only wi out by special schematic definitions.

TDO

If boundary scan is used, this is the Test Data (

If boundary scan is not used, this pin is a 3-sta without a register, after configuration is complet

This pin can be user output only when called out b schematic definitions.

TDI, TCK, TMS

If boundary scan is used, these pins are Test Dat Clock, and Test Mode Select inputs respectivel directly from the pads, bypassing the IOBs. These also be used as inputs to the CLB logic after conf is completed.

If the boundary scan option is not selected, all b scan functions are inhibited once configuration pleted, and these pins become user-programma

Note:

The XC4000 families have no Powerdown control input; use the global 3-state net instead.
The XC4000 families have no dedicated Reset input. Any user I/O can be configured to drive the global Set/Reset net.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HDC

High During Configuration is driven High until configuration is completed. It is available as a control output indicating that configuration is not yet completed. After configuration, this is a user-programmable I/O pin.

LDC

Low During Configuration is driven Low until configuration. It is available as a control output indicating that configuration is not yet completed. After configuration, this is a user-programmable I/O pin.

INIT

Before and during configuration, this is a bidirectional signal. An external pull-up resistor is recommended.

As an active-Low open-drain output, INIT is held Low during the power stabilization and internal clearing of the configuration memory. As an active-Low input, it can be used to hold the LCA device in the internal WAIT state before the start of configuration. Master mode devices stay in a WAIT state an additional 30 to 300 µs after INIT has gone High.

During configuration, a Low on this output indicates that a configuration data error has occurred. After configuration, this is a user-programmable I/O pin.

PGCK1 - PGCK4

Four Primary Global Inputs each drive a dedicated internal global net with short delay and minimal skew. If not used for this purpose, any of these pins is a user-programmable I/O.

SGCK1 - SGCK4

Four Secondary Global Inputs can each drive a dedicated internal global net, that alternatively can also be driven from internal logic. If not used for this purpose, any of these pins is a user-programmable I/O pin.

CS0, CS1, WS, RS

These four inputs are used in Peripheral mode. The chip is selected when CS0 is Low and CS1 is High. While the chip is selected, a Low on Write Strobe (WS) loads the data present on the D0 - D7 inputs into the internal data buffer; a Low on Read Strobe (RS) changes D7 into a status output: High if Ready, Low if Busy, and D0...D6 are active Low. WS and RS should be mutually exclusive, but if both are Low simultaneously, the Write Strobe overrides. After configuration, these are user-programmable I/O pins.

A0 - A17

During Master Parallel mode, these 18 output pins address the configuration EPROM. After configuration, these are user-programmable I/O pins.

D0 - D7

During Master Parallel and Peripheral configuration modes, these eight input pins receive configuration data. After configuration, they are user-programmable I/O pins.

DIN

During Slave Serial or Master Serial configuration modes, this is the serial configuration data input receiving data on the rising edge of CCLK.

During parallel configuration modes, this is the D0 input. After configuration, DIN is a user-programmable I/O pin.

DOUT

During configuration in any mode, this is the serial configuration data output that can drive the DIN of daisy-chained slave LCA devices. DOUT data changes on the falling edge of CCLK, one-and-a-half CCLK periods after it was received at the DIN input. After configuration, DOUT is a user-programmable I/O pin.

Unrestricted User-Programmable I/O Pins

I/O

A pin that can be configured to be input and/or output after configuration is completed. Before configuration is completed, these pins have an internal high-value pull-up resistor that defines the logic level as High.

Before and during configuration, all outputs that are not used for the configuration process are 3-stated with a 50 kΩ to 100 kΩ pull-up resistor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XC1700 Family of Serial Configuration PROMs

Preliminary Product Specifications

Features

- Family of one-time programmable (OTP) bit-stream read-only memories used for storing the configuration bitstreams of Xilinx FPGAs
- Simple address counter, incremented by each rising edge on the clock input
- Simple interface to the FPGA requires only one user I/O pin
- Cascadable for storing longer or multiple bitstreams
- Programmable reset polarity (active High or active Low) for compatibility with different FPGA solutions
- XC17128D or XC17256D supports XC4000 fast configuration mode (12.5 MHz)
- Low power CMOS EPROM process
- Available in 5 V and 3.3 V versions
- Available in plastic and ceramic packages, and commercial, industrial and military temperature ranges
- Space efficient 8-pin DIP, 8-pin SOIC or 20-pin surface-mount packages.
- Programming support by leading programmer manufacturers.

Description

The XC1700 family of serial configuration PROMs (SCPs) provides an easy-to-use, cost-effective method for storing Xilinx FPGA configuration bitstreams.

When the FPGA is in master serial mode, it generates a configuration clock that drives the SCP. A short access time after the rising clock edge, data appears on the SCP DATA output pin that is connected to the FPGA DIN pin. The FPGA generates the appropriate number of clock pulses to complete the configuration. Once configured, it disables the SCP. When the FPGA is in slave mode, the SCP and the FPGA must both be clocked by an incoming signal.

Multiple devices can be concatenated by using the $\overline{CEO}$ output to drive the $\overline{CE}$ input of the following device. The clock inputs and the DATA outputs of all SCPs in this chain are interconnected. All devices are compatible and can be cascaded with other members of the family.

For device programming, the XACT development system compiles the LCA design file into a standard Hex format, which is then transferred to the programmer.

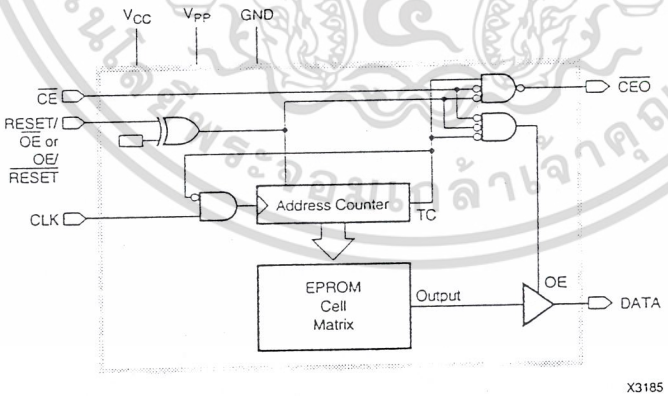


Figure 1: Simplified Block Diagram (does not show programming circuit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

DATA
Data output, 3-stated when either $\overline{CE}$ or $\overline{OE}$ are inactive. During programming, the DATA pin is I/O. Note that $\overline{OE}$ can be programmed to be either active High or active Low.

CLK
Each rising edge on the CLK input increments the internal address counter, if both $\overline{CE}$ and $\overline{OE}$ are active.

RESET/$\overline{OE}$
When High, this input holds the address counter reset and 3-states the DATA output. The polarity of this input pin is programmable as either RESET/$\overline{OE}$ or OE/$\overline{RESET}$. To avoid confusion, this document describes the pin as RESET/$\overline{OE}$, although the opposite polarity is possible on all devices. When RESET is active, the address counter is held at zero, and the DATA output is 3-stated. The polarity of ths input is programmable. The default is active High RESET, but the preferred option is active Low $\overline{RESET}$, because it can be driven by the FPGA's $\overline{INIT}$ pin.

The polarity of this pin is controlled in the programmer interface by writing data into four high-end byte locations. This input pin is easily inverted using the Xilinx PROM programmer software (XPP). Third-party programmers have differant methods to invert this pin.

For RESET/$\overline{OE}$, fill the four polarity bytes with Ones or do nothing.

For $\overline{RESET}$/OE, fill these four bytes with Zeros.

$\overline{CE}$
When High, this pin disables the internal address counter, 3-states the DATA output, and forces the device into low-I_{CC} standby mode.

$\overline{CEO}$
Chip Enable output, to be connected to the CE input of the next SCP in the daisy chain. This output is Low when the $\overline{CE}$ and $\overline{OE}$ inputs are both active AND the internal address counter has been incremented beyond its Terminal Count (TC) value. In other words: when the PROM has been read, $\overline{CEO}$ will follow $\overline{CE}$ as long as $\overline{OE}$ is active. When $\overline{OE}$ goes inactive, $\overline{CEO}$ stays High until the PROM is reset. Note that $\overline{OE}$ can be programmed to be either active High or active Low.

V_{PP}
Programming voltage. No overshoot above the specified max voltage is permitted on this pin. For normal read operation, this pin *must* be connected to V_{CC}. Failure to do so may lead to unpredictable, temperature-dependent operation and severe problems in circuit debugging. *Do not leave V_{PP} floating!*

V_{CC}
Positive supply pin.

GND
Ground pin

Serial PROM Pinouts

Pin Name	8-Pin
DATA	1
CLK	2
RESET/$\overline{OE}$ (OE/$\overline{RESET}$)	3
$\overline{CE}$	4
GND	5
$\overline{CEO}$	6
V_{PP}	7
V_{CC}	8

Capacity

Device	Configuration Bits
XC1718D or L	18,144
XC1736D	36,288
XC1765D or L	65,536
XC17128D	131,072
XC17256D or L	262,144

plus 32 bits for reset polarity control

Number of Configuration Bits, Including Header for all Xilinx FPGAs and Compatible SCP Types

Device	Configuration Bits	SCP
XC2064	12,038	XC1718D
XC2018	17,878	XC1718D
XC3020/3120/A	14,819	XC1718D
XC3030/3130/A	22,216	XC1736D
XC3042/3142/A	30,824	XC1736D
XC3064/3164/A	46,104	XC1765D
XC3090/3190/A	64,200	XC1765D
XC3195/A	94,984	XC17128D
XC4002A	31,668	XC1736D
XC4003A	45,676	XC1765D
XC4003H	53,967	XC1765D
XC4004A	62,244	XC1765D
XC4005A	81,372	XC17128D
XC4005/4005H	95,000	XC17128D
XC4006	119,832	XC17128D
XC4008	147,544	XC17256D
XC4010/D	178,136	XC17256D
XC4013/D	247,960	XC17256D
XC4025	422,168	XC17256 + XC1725

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PMI

ADC-908

CMOS MICROPROCESSOR-COMPATIBLE FAST 8-BIT A/D CONVERTER

Precision Monolithics Inc.

FEATURES

- 8-Bit Resolution and Accuracy
- No Missing Codes over Full Temperature Range
- 6μs Conversion Time
- Flexible μP Interface
- 2.5mA Maximum Standby Current
- Replaces AD7574 with Improved Speed
- Available in Die Form

ORDERING INFORMATION †

		PACKAGE: 18-PIN DIP AND SO		
INL (LSB)	DNL (LSB)	MILITARY* TEMPERATURE −55°C TO +125°C	EXTENDED INDUSTRIAL TEMPERATURE −40°C TO +85°C	COMMERCIAL TEMPERATURE 0°C TO +70°C
±1/2	±3/4	ADC908AX	ADC908EX	ADC908GP
±3/4	±7/8	ADC908BX	ADC908FX	—
±3/4	±7/8	—	ADC908FP	—
±3/4	±7/8	—	ADC908FS	—

* For devices processed in total compliance to MIL-STD-883, add /883 after part number. Consult factory for 883 data sheet.

† Burn-in is available on commercial and industrial temperature range parts in CerDIP, plastic DIP, and TO-can packages. For ordering information, see PMI's Data Book, Section 2.

GENERAL DESCRIPTION

The ADC-908 is a monolithic CMOS successive-approximation analog-to-digital converter. When used with a 1.35MHz clock, a conversion time of 6μs is achieved, with full accuracy over the operating temperature range.

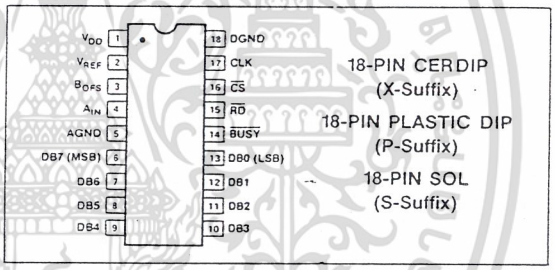
The ADC-908 outputs use 3-state logic, allowing direct connection to the data bus or system input port. Active-LOW chip select ($\overline{CS}$) and read/write ($\overline{RD}$) inputs are used to control all operations. This input structure permits the ADC-908 to be used as a memory-mapped input device. Depending on the control timing waveforms, the ADC-908 is interfaced like static RAM, ROM, or slow memory.

The low power consumption of the ADC-908 is derived from a single +5V supply. A negative reference voltage must also be supplied. Optimum accuracy is achieved when the reference is at −10.00V with a low output resistance. For a low-cost precision −10V/−10.24V reference, ask your PMI sales representative about the REF-08.

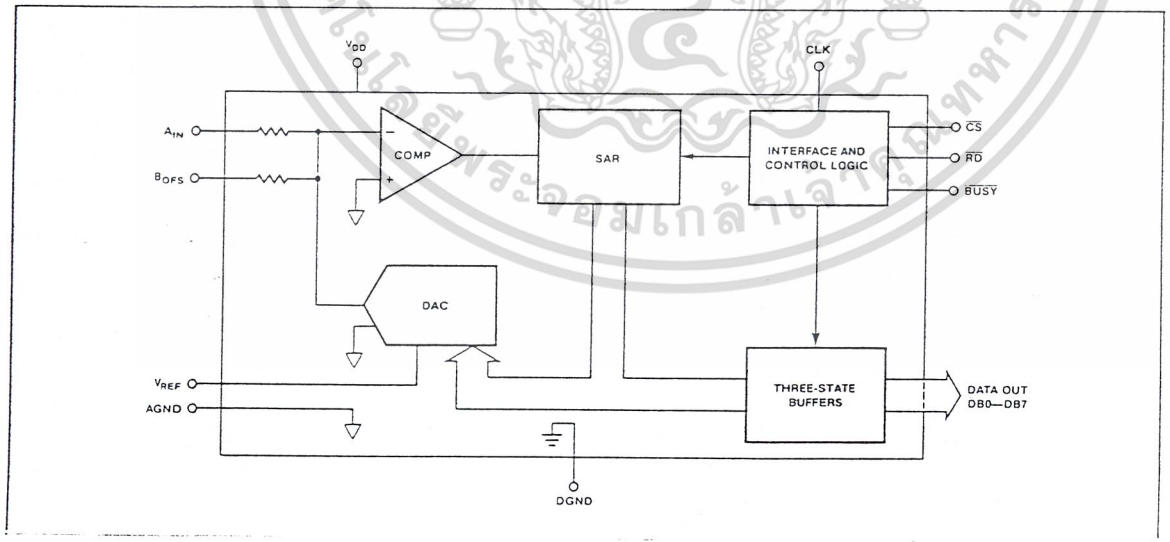
With its on-board comparator, interface logic, optional internal clock, and +5V operation, the ADC-908 is the ideal low-cost solution for microprocessor-based 8-bit A/D systems.

PMI's ADC-908 is pin-and-function compatible with the PM-7574, but offers faster conversion time and faster microprocessor bus interface timing. Conversion time has been reduced by 60% and most key timing specifications, including data access time, START command propagation delay (t_{WBPD}), and reset time, have been improved.

PIN CONNECTIONS



FUNCTIONAL DIAGRAM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS (T_A = +25°C, unless otherwise noted)

V_{DD} to AGND ..... 0V, +7.0V
V_{DD} to DGND ..... 0V, +7.0V
AGND to DGND ..... –0.3V, V_{DD}
$\overline{CS}$, $\overline{RD}$ to DGND ..... –0.3V, V_{DD} + –.3V
DB_0-DB_7 to DGND ..... –0.3V, V_{DD}
CLK, $\overline{BUSY}$ to DGND ..... –0.3, V_{DD}
B_{OFS}, A_{IN} ..... ±20V
V_{REF} ..... 0V, –20V
Operating Temperature Range
ADC-908AX, BX ..... –55°C to +125°C
ADC-908EX, FX, FP, FS ..... –40°C to +85°C
ADC-908GP ..... 0°C to +70°C
Storage Temperature ..... –65°C to +150°C
Lead Temperature (Soldering, 10 sec) ..... +300°C

PACKAGE TYPE	Θ_{JA} (Note 2)	Θ_{JC}	UNITS
18-Pin Hermetic DIP (X)	79	11	°C/W
18-Pin Plastic DIP (P)	70	30	°C/W
18-Pin SOL (S)	88	25	°C/W

NOTES:
1. Digital pins are zener protected. However, proper ESD handling precautions are recommended.
2. Θ_{JA} is specified for worst case mounting conditions, i.e., Θ_{JA} is specified for device in socket for TO, CerDIP, P-DIP, and LCC packages; Θ_{JA} is specified for device soldered to printed circuit board for SO and PLCC packages.

ELECTRICAL CHARACTERISTICS at V_{DD} = +5V, V_{REF} = –10V, Unipolar Configuration, R_{CLK} = 43kΩ, C_{CLK} = 100pF; –40°C ≤ T_A ≤ +85°C for ADC-908E/F, 0°C ≤ T_A ≤ +70°C for ADC-908G, –55°C ≤ T_A ≤ +125°C for ADC-908A/B, unless otherwise noted.

PARAMETER	SYMBOL	CONDITIONS		ADC-908 MIN	ADC-908 TYP	ADC-908 MAX	UNITS
ACCURACY							
Resolution	N			8	–	–	Bits
Integral Nonlinearity	INL	A/E/G Grades		–1/2	–	+1/2	LSB
		B/F Grades		–3/4	–	+3/4	
Differential Nonlinearity	DNL	A/E/G Grades		–3/4	–	+3/4	LSB
		B/F Grades		–7/8	–	+7/8	
Gain Error	G_{FSE}	A/E/G Grades	T_A = +25°C	–3	–	+3	LSB
			T_A = Full Temp Range	–4.5	–	+4.5	
		B/F Grades	T_A = +25°C	–5	–	+5	
			T_A = Full Temp Range	–6.5	–	+6.5	
Offset Error	V_{ZSE}	A/E/G Grades	T_A = +25°C	–30	–	+30	mV
			T_A = Full Temp Range	–50	–	+50	
		B/F Grades	T_A = +25°C	–60	–	+60	
			T_A = Full Temp Range	–80	–	+80	
ANALOG INPUTS							
Resistance Mismatch B_{OFS} to A_{IN}	ΔR_{AB}			–1	–	+1	%
Input Resistance at V_{REF} (Note 1)	R_{REF}			5	–	15	kΩ
Input Resistance at B_{OFS}, A_{IN}	R_{BOFS} R_{AIN}			10	–	30	kΩ
Reference Voltage Range	V_{REF}	Specified Conversion Accuracy		–	–10	–	V
Reference Voltage Range	V_{REF}	Degraded Conversion Accuracy		–5	–	–15	V
Reference Current (Note 6)	I_{REF}	Conversion Complete Prior to Reset		–	–	2.4	mA
Nominal Analog Input Range							
Unipolar Mode	V_{INU}			–	0 to +$\lvert V_{REF}\rvert$	–	V
Bipolar Mode	V_{INB}			–	–$\lvert V_{REF}\rvert$ to +$\lvert V_{REF}\rvert$	–	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS at V_{DD} = +5V, V_{REF} = −10V, Unipolar Configuration, R_{CLK} = 43kΩ, C_{CLK} = 100pF; −40°C ≤ T_A ≤ +85°C for ADC-908E/F, 0°C ≤ T_A ≤ +70°C for ADC-908G, −55°C ≤ T_A ≤ +125°C for ADC-908A/B, unless otherwise noted. *Continued*

PARAMETER	SYMBOL	CONDITIONS	ADC-908 MIN	TYP	MAX	UNITS
LOGIC INPUTS						
Input HIGH Voltage $\overline{RD}$, $\overline{CS}$ Inputs	V_{IH}		2.4	—	—	V
Input LOW Voltage $\overline{RD}$, $\overline{CS}$ Inputs	V_{IL}		—	—	0.8	V
Input Current $\overline{RD}$, $\overline{CS}$ Inputs	I_{IN}	T_A = +25°C T_A = Full Temp Range	— —	— —	1 10	μA
Input Capacitance $\overline{RD}$, $\overline{CS}$ Inputs (Note 6)	C_{IN}		—	—	5	pF
Input HIGH Voltage, Clock Input	V_{IH}		2.4	—	—	V
Input LOW Voltage, Clock Input	V_{IL}		—	—	0.8	V
Input HIGH Current, Clock Input	I_{IH}		—	—	2	mA
Input LOW Current, Clock Input	I_{IL}	T_A = +25°C T_A = Full Temp Range	— —	— —	1 10	μA
LOGIC OUTPUTS						
Output HIGH Voltage $\overline{BUSY}$, DB0-7	V_{OH}	I_{SOURCE} = 40μA	4.0	—	—	V
Output LOW Voltage $\overline{BUSY}$, DB0-7	V_{OL}	I_{SINK} = 1.6mA	—	—	0.4	V
Floating Leakage Current, DB0-7	I_{LKG}	T_A = +25°C T_A = Full Temp Range	— —	— —	1 10	μA
Floating State Output Capacitance	C_{OZ}	(Note 6)	—	—	7	pF
POWER REQUIREMENTS						
Standby Current	I_{DD}	V_{DD} = +4.75V to +5.25V	—	—	2.5	mA
DIGITAL INTERFACE TIMING						
$\overline{CS}$ Minimum Pulse Width (Note 6)	t_{CS}	T_A = +25°C T_A = T_{MIN} T_A = T_{MAX}	60 50 90	— — —	— — —	ns
$\overline{RD}$ to $\overline{CS}$ Setup Time (Note 6)	t_{WSCS}		0	—	—	ns
$\overline{CS}$ to $\overline{BUSY}$ Propagation Delay (Note 6)	t_{CBPD}	$\overline{BUSY}$ Load = 20pF T_A = +25°C T_A = T_{MIN} T_A = T_{MAX} $\overline{BUSY}$ Load = 100pF T_A = +25°C T_A = T_{MIN} T_A = T_{MAX}	— — — — — —	— — — — — —	120 100 150 150 120 200	ns
$\overline{BUSY}$ to $\overline{RD}$ Setup Time (Notes 2, 6)	t_{BSR}		0	—	—	ns
$\overline{BUSY}$ to $\overline{CS}$ Setup Time (Note 6)	t_{BSCS}		0	—	—	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

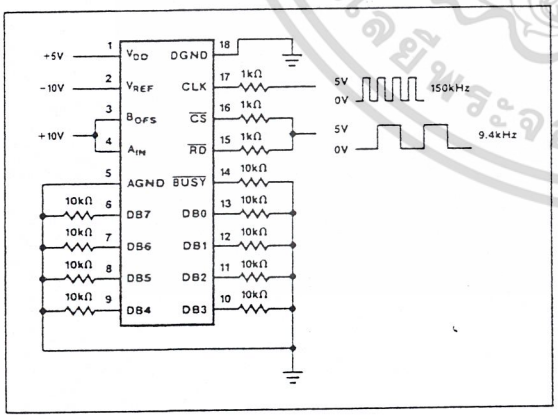
ELECTRICAL CHARACTERISTICS at V_{DD} = +5V, V_{REF} = –10V, Unipolar Configuration, R_{CLK} = 43kΩ, C_{CLK} = 100pF; –40°C ≤ T_A ≤ +85°C for ADC-908E/F, 0°C ≤ T_A ≤ +70°C for ADC-908G, –55°C ≤ T_A ≤ +125°C for ADC-908A/B, unless otherwise noted. *Continued*

PARAMETER	SYMBOL	CONDITIONS	ADC-908 MIN	TYP	MAX	UNITS
Data Access Time (Note 6)	t_{RAD}	C_L = 20pF				ns
		T_A = +25°C	—	—	140	
		T_A = T_{MIN}	—	—	100	
		T_A = T_{MAX}	—	—	200	
		C_L = 100pF				
		T_A = +25°C	—	—	170	
		T_A = T_{MIN}	—	—	150	
		T_A = T_{MAX}	—	—	230	
Data Hold Time (Notes 3, 6)	t_{RHD}	T_A = +25°C (Note 3)	30	—	100	ns
		T_A = T_{MIN}	20	—	70	
		T_A = T_{MAX}	40	—	140	
$\overline{CS}$ to $\overline{RD}$ Hold Time (Note 6)	t_{RHCS}	T_A = +25°C	—	—	200	ns
		T_A = T_{MIN}	—	—	120	
		T_A = T_{MAX}	—	—	250	
Reset Time Requirement (Note 6)	t_{RESET}	T_A = +25°C	450	—	—	ns
		T_A = Full Temp. Range	500	—	—	
Conversion Time (Note 4)	$t_{CONVERT}$	Static RAM Mode External Clock f = 1.35MHz	—	—	6	µs
(Notes 4, 5, 6)		ROM Mode Internal Clock	—	—	7	
$\overline{RD}$ HIGH to $\overline{BUSY}$ Propagation Delay, ROM Mode (Notes 4, 5, 6)	t_{WBPD}	C_L = 20pF				ns
		T_A = +25°C	—	—	600	
		T_A = T_{MIN}	—	—	400	
		T_A = T_{MAX}	—	—	800	

NOTES:

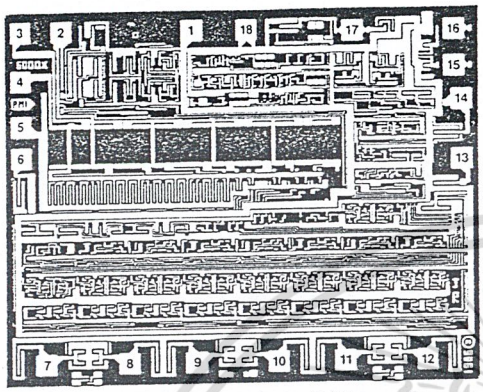
1. For optimum gain accuracy over the full temperature range, the source resistance at pin 2 should be kept low.
2. In ROM mode, $\overline{RD}$ can go LOW prior to $\overline{BUSY}$ = HIGH, but must not return HIGH until $\overline{BUSY}$ = HIGH.
3. Output loading 10pF. A 3kΩ pullup resistor to +5V is used for V_{OL} to High-Z, for V_{OH} to High-Z, a 3kΩ pulldown to GND is used. Measured to 0.5V output change.
4. When using the ADC-908 internal oscillator, actual conversion time depends on clock resistor and capacitor as well as temperature.
5. ROM interface mode conversion times are typically 1µs longer than conversion times for other modes, but the ROM interface mode includes an automatic reset in the conversion time.
6. Guaranteed but not tested.

BURN-IN CIRCUIT



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DICE CHARACTERISTICS



DIE SIZE 0.129 × 0.103 inch, 13,287 sq. mils
(3.28 × 2.62 mm, 8.58 sq. mm)

1. V_{DD}
2. V_{REF}
3. B_{OFS}
4. A_{IN}
5. AGND
6. DB7(MSB)
7. DB6
8. DB5
9. DB4
10. DB3
11. DB2
12. DB1
13. DB0(LSB)
14. $\overline{BUSY}$
15. $\overline{RD}$
16. $\overline{CS}$
17. CLK
18. DGND

For additional DICE ordering infromation, refer to PMI's Data Book, Secion 2.

WAFER TEST LIMITS at V_{DD} = +5V, V_{REF} = −10.000V, AGND = DGND = 0V, T_A = +25°C, unless otherwise noted.

PARAMETER	SYMBOL	CONDITIONS	ADC-908 LIMIT	UNITS
STATIC ACCURACY				
Resolution	N		8	Bits MIN
Integral Nonlinearity	INL		±3/4	LSB MAX
Differential Nonlinearity	DNL		±7/8	LSB MAX
Gain Error	G_{FSE}		±5	LSB MAX
Offset Error	V_{ZSE}		±60	mV MAX
ANALOG INPUTS				
Resistance Mismatch B_{OFS} to A_{IN}	ΔR_{AB}		±1	% MAX
Input Resistance at V_{REF}	R_{REF}		5/15	kΩ MIN/MAX
Input Resistance at B_{OFS}, A_{IN}	R_{BOFS}, R_{IN}		10/30	kΩ MIN/MAX
DIGITAL INPUTS				
Input HIGH Voltage at $\overline{RD}$, $\overline{CS}$ Inputs	V_{IH}		2.4	V MIN
Input LOW Voltage at $\overline{RD}$, $\overline{CS}$ Inputs	V_{IL}		0.8	V MAX
Input Current $\overline{RD}$, $\overline{CS}$ Inputs	I_{IN}		±1	μA MAX
Input HIGH Voltage Clock Input	V_{IH}		2.4	V MIN
Input LOW Voltage Clock Input	V_{IL}		0.8	V MAX
Input HIGH Current Clock Input	I_{IH}		2	mA MAX
Input LOW Current Clock Input	I_{IL}		1	μA MAX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

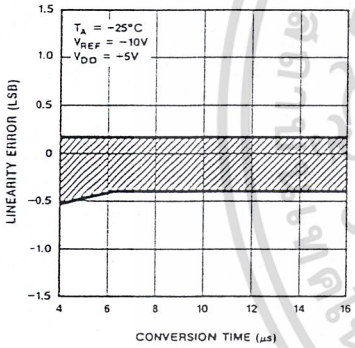
WAFER TEST LIMITS at V_{DD} = +5V, V_{REF} = −10.000V, AGND = DGND = 0V, T_A = +25°C, unless otherwise noted. (Continued)

PARAMETER	SYMBOL	CONDITIONS	ADC-908 LIMIT	UNITS
DIGITAL OUTPUTS				
Output HIGH Voltage $\overline{BUSY}$, DB0-7	V_{OH}	I_{SOURCE} = 40μA	4	V MIN
Output LOW Voltage $\overline{BUSY}$, DB0-7	V_{OL}	I_{SINK} = 1.6mA	0.4	V MAX
Floating Leakage Current	I_{LKG}		1	μA
POWER REQUIREMENTS				
Standby Current	I_{DD}	V_{DD} = +4.75V to 5.25V	2.5	mA MAX
TIMING				
Conversion Time	$t_{CONVERT}$	Static RAM Mode, External Clock, f = 1.35MHz	6	μs MAX

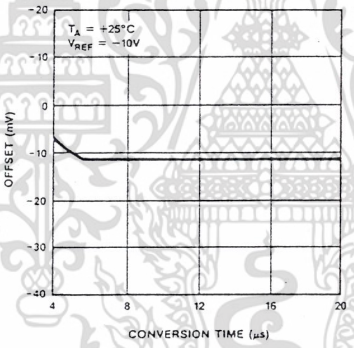
NOTE:
Electrical tests are performed at wafer probe to the limits shown. Due to variations in assembly methods and normal yield loss, yield after packaging is not guaranteed for standard product dice. Consult factory to negotiate specifications based on dice lot qualification through sample lot assembly and testing.

TYPICAL PERFORMANCE CHARACTERISTICS

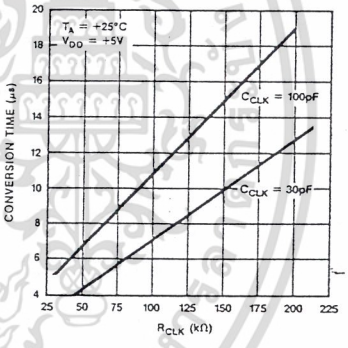
LINEARITY ERROR vs CONVERSION TIME



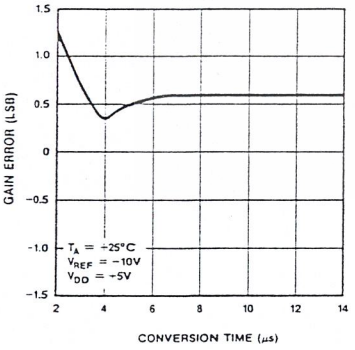
OFFSET ERROR vs CONVERSION TIME RAM MODE



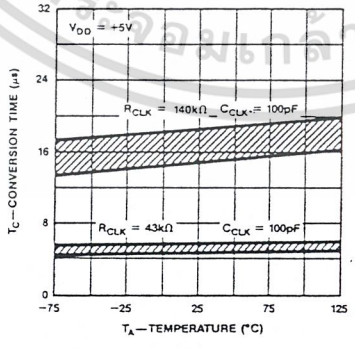
CONVERSION TIME vs R_{CLK} AND C_{CLK}



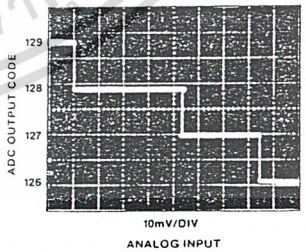
GAIN ERROR vs CONVERSION TIME



CONVERSION TIME vs TEMPERATURE WITH INTERNAL CLOCK



CONVERSION UNCERTAINTY AROUND MAJOR CARRY



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GENERAL CIRCUIT INFORMATION

The ADC-908 is an 8-bit analog-to-digital converter which uses a successive approximation technique to convert an unknown analog input into a digital code output. The control logic inputs allow easy interface to most microprocessors while three-state outputs allow direct connection to the data bus. Most applications require only passive RC clock components, a −10V reference, and a +5V power supply. The RC-timed internal clock may be used, or an external clock may be applied to the ADC to maximize performance.

When a Start Conversion command is applied to the $\overline{CS}$ or $\overline{RD}$ inputs (see Operating Descriptions for details), $\overline{BUSY}$ goes LOW indicating a conversion in progress. $\overline{BUSY}$ may be used as an interrupt to halt the controlling microprocessor during conversion or may be polled to prevent premature data reads.

Starting with the most significant bit (MSB), each successive bit in the DAC is turned on (see Figure 1). The comparator then decides if the DAC output is less than or greater than the signal being converted, and that bit is latched on or off, respectively, before proceeding to the next lower bit and repeating the cycle. When all eight bits have been tested, $\overline{BUSY}$ goes HIGH, signaling a completed conversion.

Under control of the $\overline{RD}$ input, the three-state data outputs (D0-D7) change from high-impedance to presenting the new conversion results to the data bus. Following the data read, $\overline{RD}$ returns HIGH resetting the SAR to 1000 0000 and preparing the ADC for its next conversion.

PIN FUNCTIONS

NOTE: For greater detail on digital input functions, consult Truth Tables and Timing Diagrams.

Pin 1. V_{DD} Power Supply input, +5V.

Pin 2. V_{REF} Voltage Reference input, nominal −10V.

Pin 3. B_{OFS} Bipolar Offset input. +10V input for bipolar mode operation, tie to V_{IN} for unipolar mode operation.

Pin 4. A_{IN} Analog Input. 0V to +10V in unipolar mode, −10V to +10V in bipolar mode.

Pin 14. $\overline{BUSY}$ Conversion status output. BUSY indicates conversion in progress by going LOW at start of conversion and returning HIGH at end of conversion. May be used to interrupt controlling microprocessor or to gate control inputs.

Pin 15. $\overline{RD}$ READ input. Used to read data (on falling edge) and to reset converter (on rising edge).

Pin 16. $\overline{CS}$ Chip Select input. Asserted to allow ADC operation. Starts conversion when converter is in reset condition. Note: Holding $\overline{CS}$ HIGH will not prevent a rising edge on $\overline{RD}$ from resetting the converter.

Pin 17. CLK External clock input/internal clock RC timing input.

APPLICATIONS INFORMATION

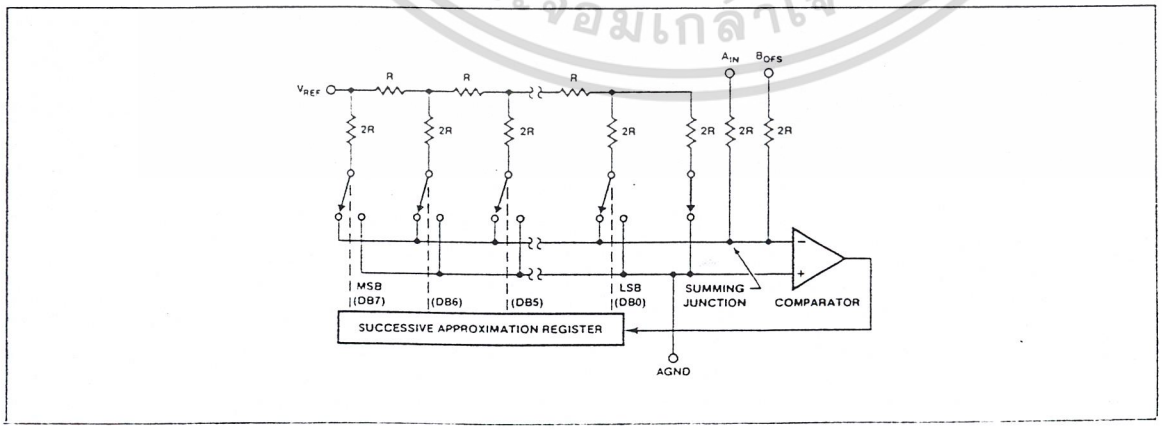
The ADC-908 may be interfaced as if it were a static RAM, a ROM, or a slow-memory device. Each of these interface modes has its own timing and software requirements as described below. These requirements must be rigidly met, as improper timing may cause the ADC-908 to change modes.

HOW TO CHOOSE AN OPERATING MODE

The static-RAM interface mode offers advantages in a tightly controlled hardware and software environment, where the relationship between WRITE and READ instruction pairs is certain. As long as minimum timing is satisfied, converted data may be read at any convenient time after conversion. The use of separate commands to start a conversion, and then read the results, is conceptually easy. However, if the software is subject to uncontrolled modifications, then the paired relationship between WRITE and READ instructions may be lost. Resulting software bugs may result in converted data of unknown age, or altogether invalid data being read.

By contrast, the ROM mode may be more resistant to software bugs. As long as minimum timing is satisfied, each READ instruction obtains new, valid data. However, since the data

FIGURE 1: D/A Converter Used in ADC-908



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

output at any previous READ instruction is obtained from a conversion performed just after the previous READ instruction, data may be out-of-date. To be sure of obtaining up-to-date data, READ instructions may be coded in pairs (with some NOPs between them); use only the data from the second READ in each pair. The first READ starts the conversion, acting as a substitute for the static-RAM mode WRITE command; the second READ gets the results. The advantage of the ROM mode is the use of a single command, rather than the alternating READ-WRITE required by static-RAM mode.

The slow-memory mode is the simplest mode of all. It is the method of choice where compact coding is essential, or where software bugs are a hazard. In this mode, a single READ instruction will initiate a data conversion, interrupt the microprocessor until completion (WAIT states are introduced), then read the results. If the system throughput tolerates WAIT states, and the hardware is correct, then the slow-memory mode is virtually immune to subsequent software modifications.

OPERATING DESCRIPTION: STATIC-RAM MODE

In this mode, input $\overline{CS}$ is derived from the ADC-908 address decoder, and input $\overline{RD}$ is derived from an active-LOW memory READ signal. (See Figure 2.)

To start a conversion, execute a memory WRITE to the ADC-908. The completed conversion data is obtained by executing a memory READ to the ADC-908. During conversion, output $\overline{BUSY}$ will be LOW. Do not attempt to read data until $\overline{BUSY}$ returns HIGH. The required minimum time between WRITE and READ is usually obtained by including one or more NOP or other program instructions. The use of branch or conditional commands between the WRITE and READ instructions is not recommended due to the possibility of software bugs.

It is important that the WRITE and READ commands be alternately executed. A WRITE instruction has no effect unless the results of the previous WRITE have already been read. Once data has been read, the ADC-908 is internally reset. In other words, two or more READ operations cannot be used in succession, since only the first READ will produce valid data. A new conversion must be started using WRITE, and the conversion must be completed, before a new READ will produce valid data.

TABLE 1: Truth Table, Static RAM Mode

INPUTS $\overline{CS}$	INPUTS $\overline{RD}$	OUTPUTS $\overline{BUSY}$	OUTPUTS DB7-DB0	ADC-908 OPERATION
L	H	H	HIGH-Z	Start Convert (Write Cycle)
L	↓	H	HIGH-Z to DATA	Read Data (Read Cycle)
L	↑	H	DATA to HIGH-Z	Reset Converter
H	X (Note 1)	X	HIGH-Z	No Effect (Not Selected)
L	H	L	HIGH-Z	No Effect (Converter Busy)
L	↓	L	HIGH-Z	No Effect (Converter Busy)
L	↑ (Note 1)	L	HIGH-Z	Conversion Error Not Allowed

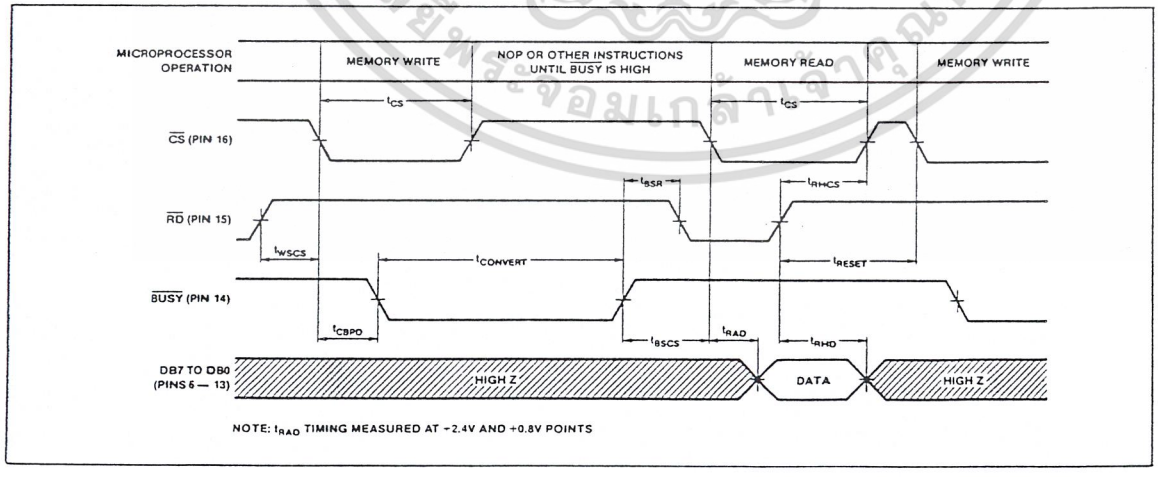
NOTE 1: If $\overline{RD}$ goes LOW to HIGH, the ADC is internally reset, regardless of the states of $\overline{CS}$ or $\overline{BUSY}$.

OPERATING DESCRIPTION: ROM MODE

In ROM mode, input $\overline{CS}$ is tied LOW, and input $\overline{RD}$ is derived from the ADC-908 address decoder. To satisfy timing, it is recommended that the decoder be enabled by a system MEMRD (8080), VMA (6800), or similar strobe. (See Figure 3.)

In ROM mode, data is read by executing a READ instruction to the ADC-908 address. At the conclusion of the READ instruction, the ADC-908 automatically resets itself and then proceeds to perform a new data conversion. Output $\overline{BUSY}$ is LOW during conversion. A new READ instruction to the ADC-908 must not be executed until $\overline{BUSY}$ returns HIGH.

FIGURE 2: Static RAM Mode Timing Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This requirement may be met by inserting NOP or other program instructions between consecutive READ operations. Conditional or branch instructions may be used, but keep in mind that data may become out-of-date if excessive time elapses between consecutive READ instructions.

TABLE 2: Truth Table, ROM Mode

INPUTS $\overline{CS}$	INPUTS $\overline{RD}$	OUTPUTS $\overline{BUSY}$	OUTPUTS DB7-DB0	ADC-908 OPERATION
L	↓	H	HIGH-Z to DATA	Read Data
L	↑	↓	DATA to HIGH-Z	Reset and Start New Conversion
L	↓	L	HIGH-Z	No Effect (Converter Busy)
L	↑ (Note 1)	L	HIGH-Z	Conversion Error Not Allowed

NOTE 1: If $\overline{RD}$ goes LOW to HIGH, the ADC is internally reset, regardless of the states of $\overline{CS}$ or $\overline{BUSY}$.

OPERATING DESCRIPTION: SLOW-MEMORY MODE

The slow-memory mode is intended for systems in which the ADC-908 $\overline{BUSY}$ output is used as an interrupt to force the microprocessor into WAIT states during data conversion.

In slow-memory mode, inputs $\overline{CS}$ and $\overline{RD}$ are tied together. The common $\overline{RD}$ and $\overline{CS}$ signal is derived from the ADC-908 address decoder. To satisfy the timing requirements, it is advisable to latch the address using ALE (8085) or SYNC (8080). For 8080 or 8085-based systems, connect the microprocessor READY input to the ADC-908 $\overline{BUSY}$ output. (See Figure 4.)

TABLE 3: Truth Table, Slow-Memory Mode

INPUTS $\overline{CS}$ & $\overline{RD}$	OUTPUTS $\overline{BUSY}$	OUTPUTS DB7-DB0	ADC-908 OPERATION
H	H	HIGH-Z	No Effect (Not Selected)
↓	↓	HIGH-Z	Start Conversion
L	L	HIGH-Z	Conversion in Progress. µP in WAIT State
L	↑	HIGH-Z to DATA	Conversion Complete. Read Data
↑	H	DATA to HIGH-Z	Reset and Deselect Converter

FIGURE 3: ROM Mode Timing Diagram ($\overline{CS}$ Held LOW)

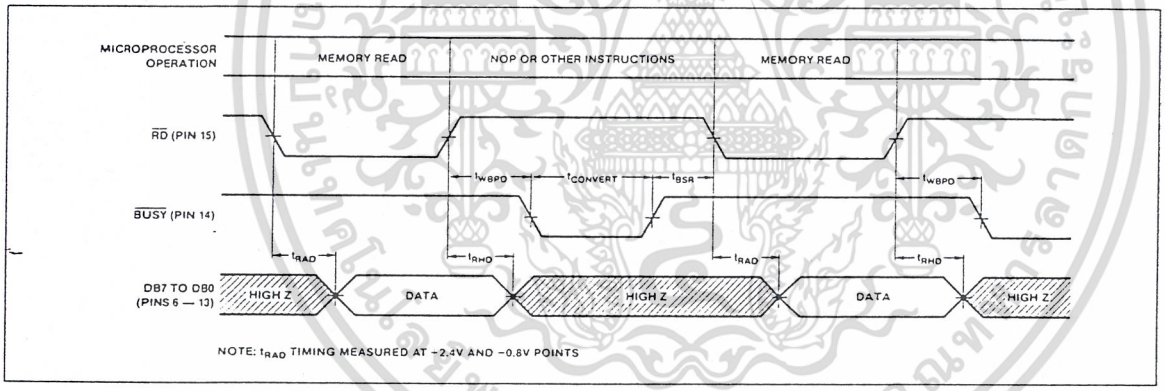
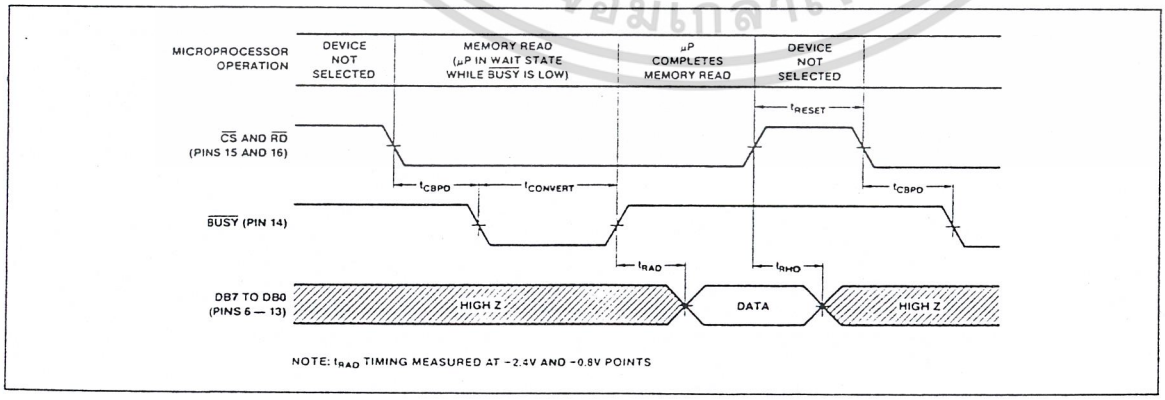


FIGURE 4: Slow-Memory Mode Timing Diagram ($\overline{CS}$ and $\overline{RD}$ Tied Together)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Do not execute a WRITE instruction at the ADC-908 address when in slow-memory mode, since bus conflicts will arise. In some architectures, an accidental WRITE instruction may be locked out in hardware, by proper strobing of the ADC-908 address decoder.

INITIALIZATION

In all operating modes, the ADC-908 is initialized by executing a READ instruction to the ADC-908 address. The data obtained should be ignored.

CLOCK OSCILLATOR

The ADC-908 may be used with its internal asynchronous clock oscillator. An external resistor and capacitor are required. Typical values are R = 43kΩ and C = 100pF, for conversion times in the 6μs range. For applications in which the fastest conversion times are required, an external clock is recommended. The external clock must be gated by the use of a 74125-type three-state buffer, with an output pullup resistor. Optimum conversion accuracy is obtained when $\overline{CS}$ goes LOW on a positive clock edge. The maximum external clock frequency is 1.35MHz (See Figure 5 and 6.)

REFERENCE VOLTAGE

A negative reference voltage must be applied to the ADC-908 V_{REF} input. Optimum full-scale accuracy is obtained using −10.00V, although V_{REF} may be −5.00V, −10.24V, or other voltages within its specified range.

Over the full temperature range, optimum gain accuracy is obtained when the input to the V_{REF} pin is from a low-impedance source. A resistor or trimmer may be used in series with the V_{REF} pin, but this trim technique is not as accurate as a low-impedance source. (See Figure 7.)

For a cost-effective −10.00V or −10.24V reference with excellent accuracy and low temperature coefficient, ask for PMI's REF-08. Consult your sales representative for availability.

FIGURE 5: Using the Internal Clock Oscillator

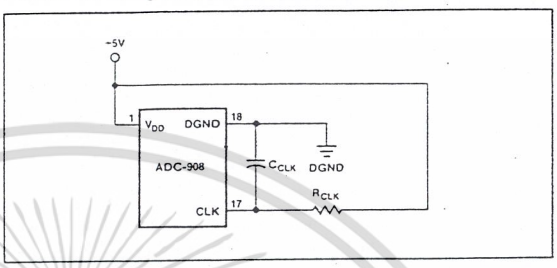


FIGURE 6: Using an External Clock

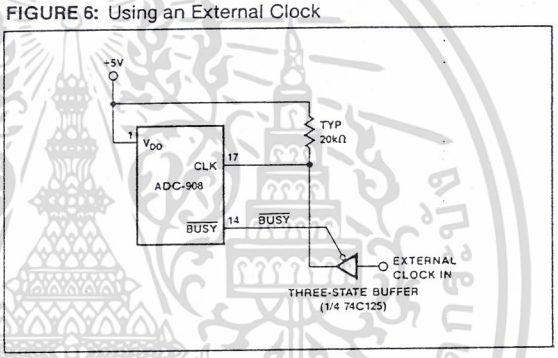
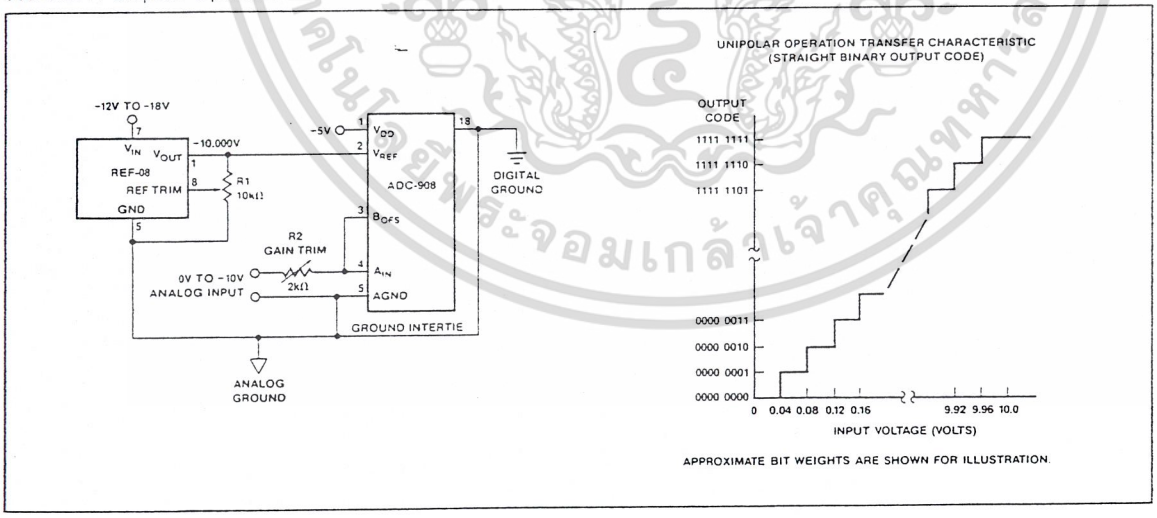


FIGURE 7: Unipolar Operation



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ANALOG INPUT VOLTAGE

The ADC-908 unipolar operation is obtained when the analog input voltage is between 0V and $|V_{REF}|$. With the A_{IN} and B_{OFS} pins tied together, input 0V will correspond to code 0000 0000, and input full-scale will correspond to code 1111 1111.

Bipolar operation is obtained by using the B_{OFS} input to offset the A_{IN} input voltage. For example, with $V_{REF} = -10V$, an offset voltage of +10V may be applied to B_{OFS}. The analog signal range will then be −10V to +10V at A_{IN}. Code 0000 0000 will correspond to −10V, and positive full scale will be code 1111 1111. Calibration may be performed using trimmers in series with A_{IN} and B_{OFS}. (See Figure 8).

Another method of obtaining bipolar operation is to use an op-amp with gain = −1/2, to sum the analog signal with the reference voltage. With a −10V reference and −10V to +10V analog signal, the op amp output will then be 0V to +10V. This signal is then treated as an ordinary unipolar input to the ADC-908. With this arrangement, input +10V corresponds to code 0000 0000, and negative full-scale corresponds to code 1111 1111.

UNIPOLAR BINARY OPERATION

Figure 7 shows the analog circuit connections for unipolar operation. The REF-08 supplies the necessary −10V reference input.

Calibration for offset should be made before gain calibration is attempted.

Offset calibration must be performed in the signal conditioning circuitry which drives the A_{IN} input.

To adjust offset:
1) Apply −39.1mV (1 LSB) to the input of the buffer amplifier driving A_{IN}.
2) While performing continuous conversions, adjust the buffer amplifier's offset adjustment potentiometer until DB7 to DB1 are LOW and DB0 (LSB) flickers.

Following offset calibration, full scale gain can be calibrated:
1) Apply −9.961V to the input of the buffer amplifier.
2) While performing continuous conversions, adjust the reference trim pot until DB7 to DB1 are HIGH, and DB0 (LSB) flickers.

BIPOLAR OPERATION

Offset Binary—Figure 8 shows a circuit for offset binary bipolar operation. Offset correction should be made at the buffer amplifier driving A_{IN}. Gain error correction should be accomplished by adjusting V_{REF}.

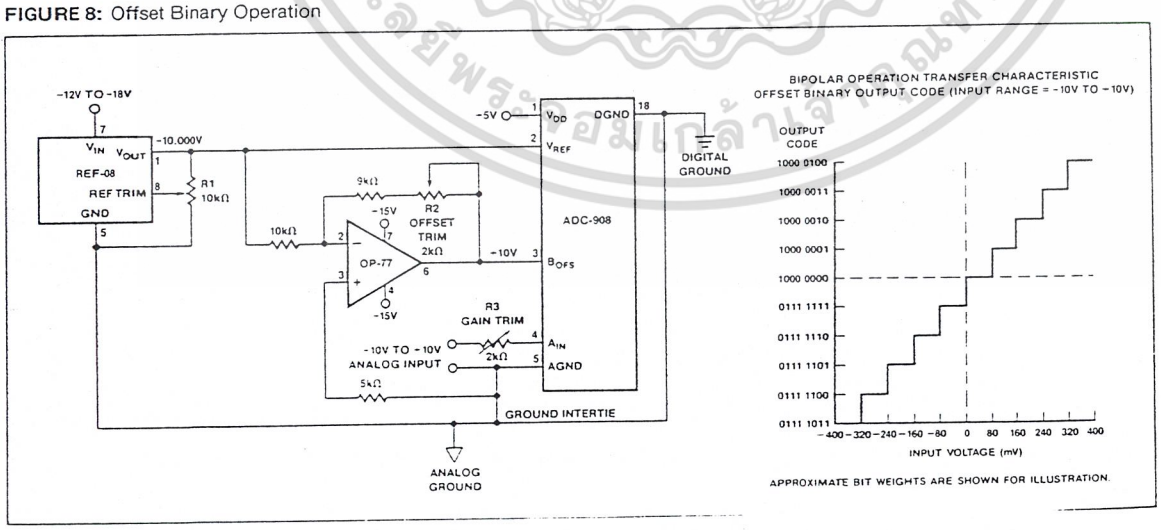
To calibrate this circuit:
1) Adjust R1 until $V_{REF} = -10.00V$.
2) Adjust R2 and R3 to their mid-points.
3) Apply +10.000V to the input buffer amplifier.
4) While performing continuous conversions, adjust R2 until DB7 to DB1 are LOW and DB0 (LSB) flickers.
5) Ground the input of the input buffer circuit.
6) While performing continuous conversions, adjust R3 until the ADC's output code flickers between 0111 1111 and 1000 0000.
7) Apply −10.000V to the signal input.
8) While performing continuous conversions, adjust R1 until DB7 to DB1 are LOW and the DB0 (LSB) flickers.
9) Apply +9.922V to the signal input.
10) If the ADC output code is not 1111 1110 ±1 bit, repeat the calibration procedure, omitting step 1.

Complementary Offset Binary—Figure 9 shows a complementary offset binary circuit. In this bipolar mode, the +10V to −10V analog input is conditioned to a 0 to +10V signal range for normal unipolar conversion.

In calibrating this circuit, adjust offset before gain.

Offset Adjustment:
1) Adjust R1 until $V_{REF} = -10.000V$.
2) Adjust R3 to its mid-point.
3) Adjust R2 until its tap is at 0V.
4) Ground the analog input.
5) While performing continuous conversions, adjust R2 until the ADC output flickers between 0111 1111 and 1000 0000.

FIGURE 8: Offset Binary Operation



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Gain Adjustment:
1) Apply +9.922V across the analog input.
2) While performing continuous conversions, adjust R3 until DB7 to DB1 are HIGH and DB0 (LSB) flickers.

DIGITAL CONSIDERATIONS

Control Timing—Fresh data from a recent conversion must be read before beginning a new conversion. Following the data READ, as $\overline{RD}$ goes HIGH, it resets the SAR and clears the data from the previous conversion.

The timing restrictions detailed in the interface timing diagrams must be observed to prevent the ADC-908 from changing interface modes. For example, if $\overline{CS}$ is held LOW too long while in RAM mode, the converter will change to ROM mode and initiate a new conversion.

Logic Deglitching—Unrelated activity on the address bus may cause unexpected glitch inputs to the ADC. The glitches may cause unwanted READs, resets, or conversions. In ROM or RAM modes, these may be avoided by gating the address decode logic with $\overline{RD}$ or $\overline{WR}$ (8080) or VMA (6800). In slow-memory mode, ALE (8085) or SYNC (8080) may be used to latch the address.

Initialization—Following power-up, the SAR is in an unknown state. Executing a memory READ (disregard the data) will reset the ADC.

ANALOG CONSIDERATIONS

Analog Input Impedances—Low impedance sources must be used to drive the V_{REF}, A_{IN}, and B_{OFS} inputs. Excessive source impedances may cause errors due to the loading effects of the inputs' finite impedances.

Ground Management—AGND and DGND pins should be connected at or near the ADC to minimize noise effects. If the two grounds cannot be connected near the ADC, the grounds should be clamped with back-to-back Schottky diodes between the AGND and DGND pins.

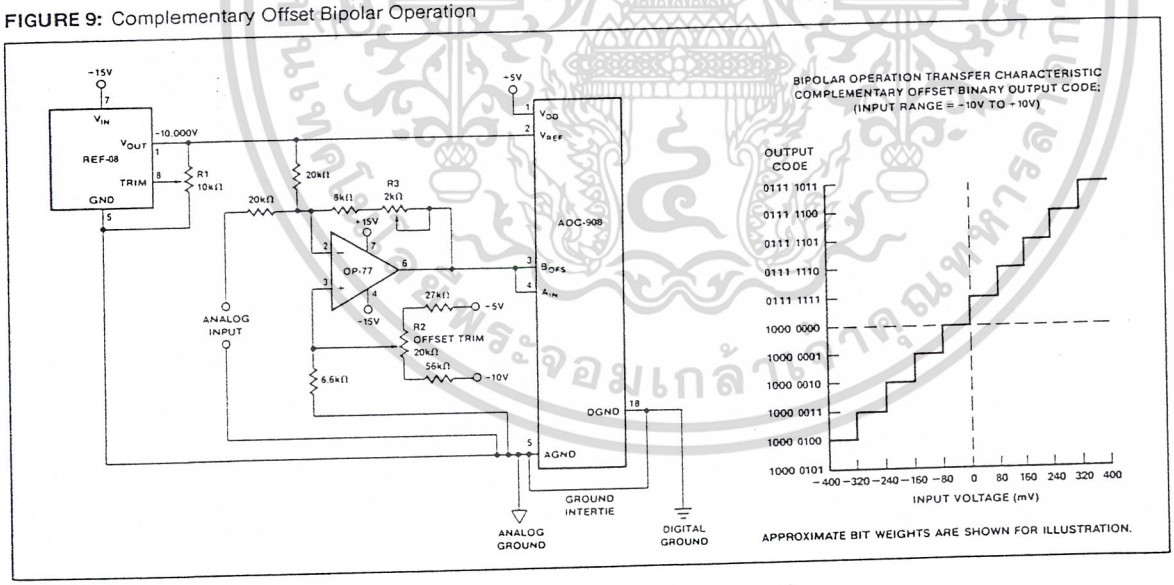
Offset Correction—Conversion offset errors may be corrected by counter-offsetting the buffer amplifier driving A_{IN}. This offset correction may be accomplished by applying a correction current to the buffer's summing junction or by tapping a voltage divider sitting between V_{DD} and V_{REF}, and applying this tap voltage to the noninverting input of the buffer.

Ratiometric Operation—The R-2R type DAC in the ADC-908 permits ratiometric operation of the ADC. Performance degradation may, however, occur as V_{REF} varies from −10.000V. This decrease in performance is due to comparator limitations including offset-voltage, gain, and input noise.

The ADC-908 uses the reference as a power supply for the comparator to increase speed and accuracy. Reference voltages of a magnitude less than −9V must be avoided for accurate comparator operation. For best accuracy, the use of a 0.1μF bypass capacitor from V_{REF}(Pin 2 to AGND) is recommended.

Power Supply Bypassing—For best accuracy, V_{DD} (Pin 1) should be bypassed to AGND with a 0.1μF capacitor.

FIGURE 9: Complementary Offset Bipolar Operation



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HARRIS

CA3338,CA3338A

CMOS Video-Speed 8-Bit R-2R Digital-to-Analog Converter

GENERAL DESCRIPTION

The Harris CA3338 family are CMOS/SOS high-speed R-2R voltage output digital-to analog converters. They operate from a single 5V supply at video speeds, and can produce "rail-to-rail" output swings. Internal level shifters and a pin for an optional second supply provide for an output range below the digital ground. The data complement control allows the inversion of input data while the latch enable control provides either feed through or latched operation. Both ends of the R-2R ladder network are available externally and may be modulated for gain or offset adjustments. In addition, "glitch" energy has been kept very low by segmenting and "bar graph" decoding of the upper 3 bits.

The CA3338 is manufactured on a sapphire substrate to give low dynamic power dissipation, low output capacitance, and inherent latch up resistance.

FEATURES

- CMOS/SOS Low Power
- R-2R Output, Segmented for Low "Glitch"
- CMOS or TTL Compatible Inputs
- Fast Settling: 20 ns (typ.) to ½ LSB
- Feedthrough Latch for Clocked or Unclocked Use
- Single or Dual Supplies, 4.5V to 7.5V Total
- ½ LSB Accuracy (Typ.)
- Data Complement Control
- High Update Rate: 50 MHz (Typ.)
- Unipolar or Bipolar Operation

APPLICATIONS

- TV/Video Display
- High-Speed Oscilloscope Display
- Digital Waveform Generator
- Feed-Forward A/D Systems

ORDERING INFORMATION

Part Number	Linearity (INL)	Temperature Range	Package
CA3338E CA3338AE	±1 LSB ±0.75 LSB	-40°C to +85°C -40°C to +85°C	16 Pin Plastic DIP 16 Pin Plastic DIP
CA3338D CA3338AD	±1 LSB ±0.75 LSB	-55°C to +125°C -55°C to +125°C	16 Pin Ceramic DIP 16 Pin Ceramic DIP

NOTE: Consult sales office for availability of SOIC package.

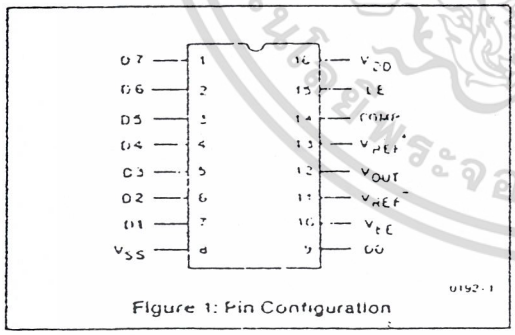


Figure 1: Pin Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS

DC Supply-Voltage Range
(V_{DD} - V_{SS} or V_{DD} - V_{EE}, whichever is greater) ..... 0.5V to + 8V

Input Voltage Range
Digital Inputs (LE, COMP, D0–D7) ..... V_{SS} - 0.5V to V_{DD} + 0.5V
Analog Pins
(V_{REF}^+, V_{REF}^-, V_{OUT}) ..... V_{DD} - 8V to V_{DD} + 0.5V

DC Input Current
Digital Inputs (LE, COMP, D0–D7) ..... ± 20 mA

Power Dissipation per Package (P_D):
For T_A = - 55°C to + 55°C ..... 315 mW
For T_A = + 55°C to + 125°C
..... Derate Linearly at 3.3 mW/°C

Operating-Temperature Range (T_A):
Ceramic Package—D Suffix ..... - 55°C to + 125°C
Plastic Package—E Suffix ..... - 40°C to + 85°C

Storage-Temperature Range (T_{STG}) ..... - 65°C to + 150°C

Lead Temperature (During Soldering):
At Distance 1/16 ± 1/32 in. (1.59 ± 0.79 mm) from case for 10s max. ..... + 265°C

NOTE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Supply Voltage Range ..... 4.5V to 7.5V

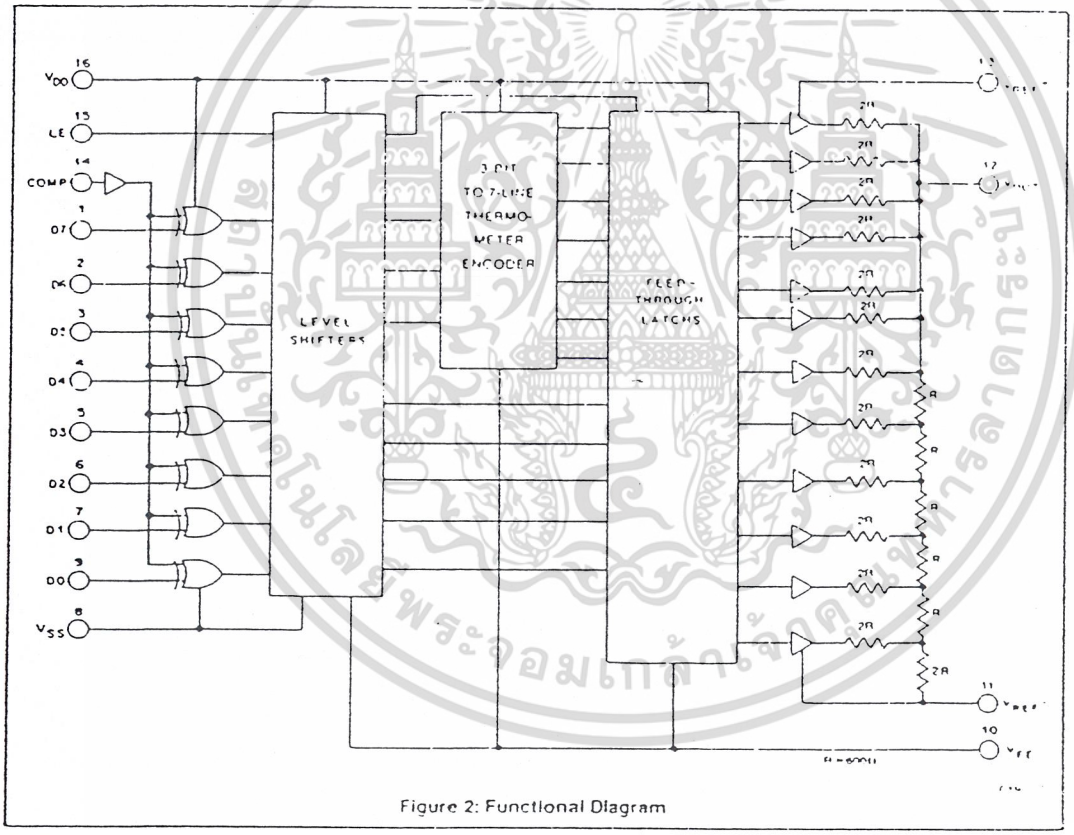


Figure 2: Functional Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS $T_A = 25°C$, $V_{DD} = 5V$, $V_{REF}^+ = 4.608V$, $V_{SS} = V_{EE} = V_{REF}^- = GND$, LE clocked at 20 MHz, $R_L \geq 1 M\Omega$ (unless otherwise specified)

Parameter	Test Conditions	Limits			Units
		Min	Typ	Max	
Accuracy					
Resolution		8			Bits
Integral Linearity Error	See Figure 6				
CA3338				±1	LSB
CA3338A				±0.75	LSB
Differential Linearity Error	See Figure 6				
CA3338				±0.75	LSB
CA3338A				±0.5	LSB
Gain Error	Input Code = FF_{16}; See Fig. 5				
CA3338				±0.75	LSB
CA3338A				±0.5	LSB
Offset Error	Input Code = 00_{16}; See Fig. 5			±0.25	LSB
Digital Input Timing					
Update Rate	To Maintain ½ LSB Settling	DC	50		MHz
Update Rate	$V_{REF}^- = V_{EE} = -2.5V$, $V_{REF}^+ = +2.5V$	DC	20		MHz
Set Up Time T_{SU1}	For Low Glitch		−2		ns
Set Up Time T_{SU2}	For Data Store		8		ns
Hold Time T_H	For Data Store		5		ns
Latch Pulse Width T_W	For Data Store		5		ns
Latch Pulse Width T_W	$V_{REF}^- = V_{EE} = -2.5V$, $V_{REF}^+ = +2.5V$		25		ns
Output Parameters	R_L Adjusted for 1 V_{p-p} Output				
Output Delay T_{D1}	From LE Edge		25		ns
Output Delay T_{D2}	From Data Changing		22		ns
Rise Time T_R	10 to 90% of Output		4		ns
Settling Time T_S	10% to Settling to ½ LSB		20		ns
Output Impedance	$V_{REF}^+ = 6V$, $V_{DD} = 6V$	120	160	200	Ω
Glitch Area			150		pV-s
Glitch Area	$V_{REF}^- = V_{EE} = -2.5V$, $V_{REF}^+ = +2.5V$		250		pV-s
Reference Voltage					
V_{REF}^+ Range	(+) Full Scale (Note 1)	$V_{REF}^- + 3$		V_{DD}	V
V_{REF}^- Range	(−) Full Scale (Note 1)	V_{EE}		$V_{REF}^+ - 3$	V
V_{REF}^+ Input Current	$V_{REF}^+ = 6V$, $V_{DD} = 6V$		40	50	mA
Supply Voltage					
Static I_{DD} or I_{EE}	LE = Low, D_0–D_7 = High		100	220	µA
	LE = Low, D_0–D_7 = Low			100	µA
Dynamic I_{DD} or I_{EE}	V_{OUT} = 10 MHz, 0V to 5V Sq. Wave		20		mA
Dynamic I_{DD} or I_{EE}	V_{OUT} = 10 MHz, ±2.5V Sq. Wave		25		mA
V_{DD} Rejection	50 kHz Sine Wave Applied		3		mV/V
V_{EE} Rejection	50 kHz Sine Wave Applied		1		mV/V
Digital Inputs	D0–D7, LE, COMP				
High Level Input Voltage	(Note 1)	2			V
Low Level Input Voltage	(Note 1)			0.8	V
Leakage Current			±1	±5	µA
Capacitance			5		pF
Temperature Coefficients					
Output Impedance			200		PPM/°C

NOTE 1: Parameter not tested, but guaranteed by design or characterization.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1: Pin Descriptions

Pin	Name	Description
1	D7	Most Significant Bit
2	D6	Input
3	D5	Data
4	D4	Bits
5	D3	(High = True)
6	D2	
7	D1	
8	V_{SS}	Digital ground
9	D_0	Least Significant Bit. Input Data Bit.
10	V_{EE}	Analog ground
11	V_{REF-}	Reference voltage negative input
12	V_{OUT}	Analog output
13	V_{REF+}	Reference voltage positive input
14	COMP	Data complement control input. Active high.
15	LE	Latch enable input. Active low.
16	V_{DD}	Digital power supply, + 5V

DIGITAL SIGNAL PATH

The digital inputs (LE, COMP, and D0-D7) are of TTL compatible HCT High Speed CMOS design the loading is essentially capacitive and the logic threshold is typically 1.5V.

The 8 data bits, D0 (weighted 2^0) through D7 (weighted 2^7), are applied to Exclusive OR gates (see Figure 2). The COMP (data complement) control provides the second input to the gates; if COMP is high, the data bits will be inverted as they pass through.

The input data and the LE (latch enable) signals are next applied to a level shifter. The inputs, operating between the levels of V_{DD} and V_{SS}, are shifted to operate between V_{DD} and V_{EE}. V_{EE} optionally at ground or at a negative voltage, will be discussed under bipolar operation. All further logic elements except the output drivers operate from the V_{DD} and V_{EE} supplies.

The upper 3 bits of data, D5 through D7, are input to a 3-to-7 line bar graph encoder. The encoder outputs and D0 through D4 are applied to a feedthrough latch, which is controlled by LE (latch enable).

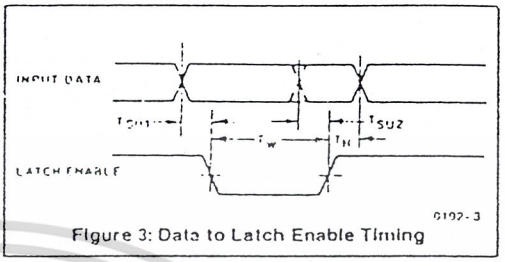


Figure 3: Data to Latch Enable Timing

LATCH OPERATION

Data is fed from input to output while LE is low; LE should be tied low for non-clocked operation.

Non-clocked operation or changing data while LE is low is not recommended for applications requiring low output "glitch" energy, there is no guarantee of the simultaneous changing of input data or the equal propagation delay of all bits through the converter. Several parameters are given if the converter is to be used in either of these modes. T_{D2} gives the delay from the input changing to the output changing (10%), while t_{SU2} and T_H give the set up and hold times (referred to LE rising edge) needed to latch data. See Figures 3 and 4.

Clocked operation is needed for low "glitch" energy use. Data must meet the given T_{SU1} set up time to the LE falling edge, and the T_H hold time from the LE rising edge. The delay to the output changing, T_{D1}, is now referred to the LE falling edge.

There is no need for a square wave LE clock, LE must only meet the minimum TW pulse width for successful latch operation. Generally, output timing (desired accuracy of setting) sets the upper limit of usable clock frequency.

OUTPUT STRUCTURE

The latches feed data to a row of high current CMOS drivers, which in turn feed a modified R-2R ladder network.

The "N" channel (pull down) transistor of each driver plus the bottom "2R" resistor are returned to V_{REF-} ; this is the ( ) full-scale reference. The "P" channel (pull up) transistor of each driver is returned to V_{REF+}, the ( ) full-scale reference.

In unipolar operation, V_{REF-} would typically be returned to analog ground, but may be raised above ground (see specifications). There is substantial code dependent current that flows from V_{REF+} to V_{REF-} (see V_{REF+} input current in specifications), so V_{REF-} should have a low impedance path to ground.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT STRUCTURE (Continued)

In bipolar operation, V_{REF}^- would be returned to a negative voltage (the maximum voltage rating to V_{DD} must be observed). V_{EE}, which supplies the gate potential for the output drivers, must be returned to a point at least as negative as V_{REF}^-. Note that the maximum clocking speed decreases when the bipolar mode is used.

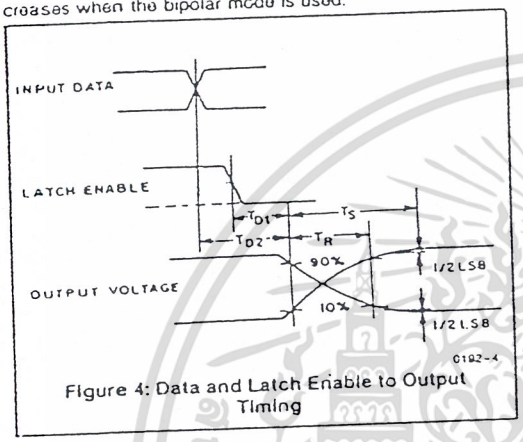


Figure 4: Data and Latch Enable to Output Timing

STATIC CHARACTERISTICS

The ideal 8-bit D/A would have an output equal to V_{REF}^- with an input code of 00_{16} (zero scale output), and an output equal to 255/256 of V_{REF}^+ (referred to V_{REF}^-) with an input code of FF_{16} (full-scale output). The difference between the ideal and actual values of these two parameters are the OFFSET and GAIN errors respectively; see Figure 5.

If the code into an 8-bit D/A is changed by 1 count, the output should change by 1/255 (full-scale output—zero-scale output). A deviation from this step-size is a differential linearity error; see Figure 6. Note that the error is expressed in fractions of the ideal step size (usually called an LSB). Also note that if the (−) differential linearity error is less (in absolute numbers) than 1 LSB, the device is monotonic. (The output will always increase for increasing code or decrease for decreasing code).

If the code into an 8-bit D/A is at any value, say "N", the output voltage should be N/255 of the full-scale output (referred to the zero-scale output). Any deviation from that output is an integral linearity error, usually expressed in LSB's. See Figure 6.

Note that OFFSET and GAIN errors do not affect integral linearity, as the linearity is referenced to actual zero and full-scale outputs, not ideal. Absolute accuracy would have to also take these errors into account.

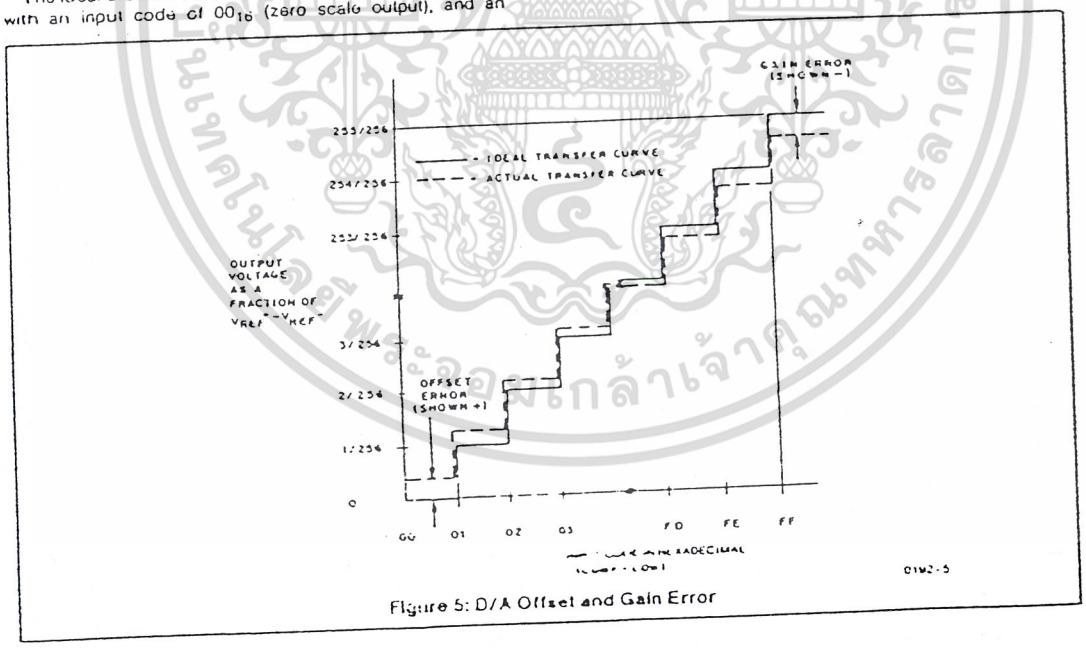


Figure 5: D/A Offset and Gain Error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

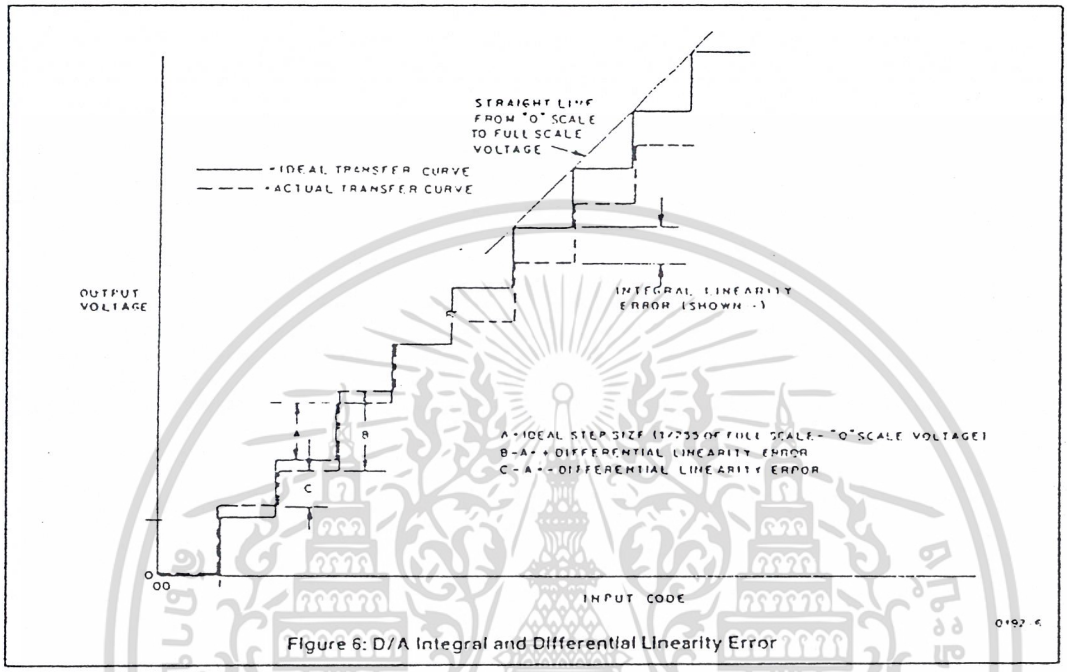


Figure 6: D/A Integral and Differential Linearity Error

DYNAMIC CHARACTERISTICS

Keeping the full-scale range ($V_{REF}^+ - V_{REF}^-$) as high as possible gives the best linearity and lowest "glitch" energy (referred to 1V). This provides the best "P" and "N" channel gate drives (hence saturation resistance) and propagation delays. The V_{REF}^+ (and V_{REF}^- if bipolar) terminal should be well bypassed as near the chip as possible.

"Glitch" energy is defined as a spurious voltage that occurs as the output is changed from one voltage to another. In a binary input converter, it is usually highest at the most significant bit transition ($7F_{16}$ to 80_{16} for an 8 bit device), and can be measured by displaying the output as the input code alternates around that point. The "glitch" energy is the area between the actual output display and an ideal one LSB step voltage (subtracting negative area from positive), at either the positive or negative-going step. It is usually expressed in pV-s.

The CA3338 uses a modified R-2R ladder, where the 3 most significant bits drive a bar graph decoder and 7 equally weighted resistors. This makes the "glitch" energy at each ⅛ scale transition ($1F_{16}$ to 20_{16}, $3F_{16}$ to 40_{16}, etc.) essentially equal, and far less than the MSB transition would otherwise display.

For the purpose of comparison to other converters, the output should be resistively divided to 1V full-scale. Figure 7 shows a typical hook-up for checking "glitch" energy or settling time.

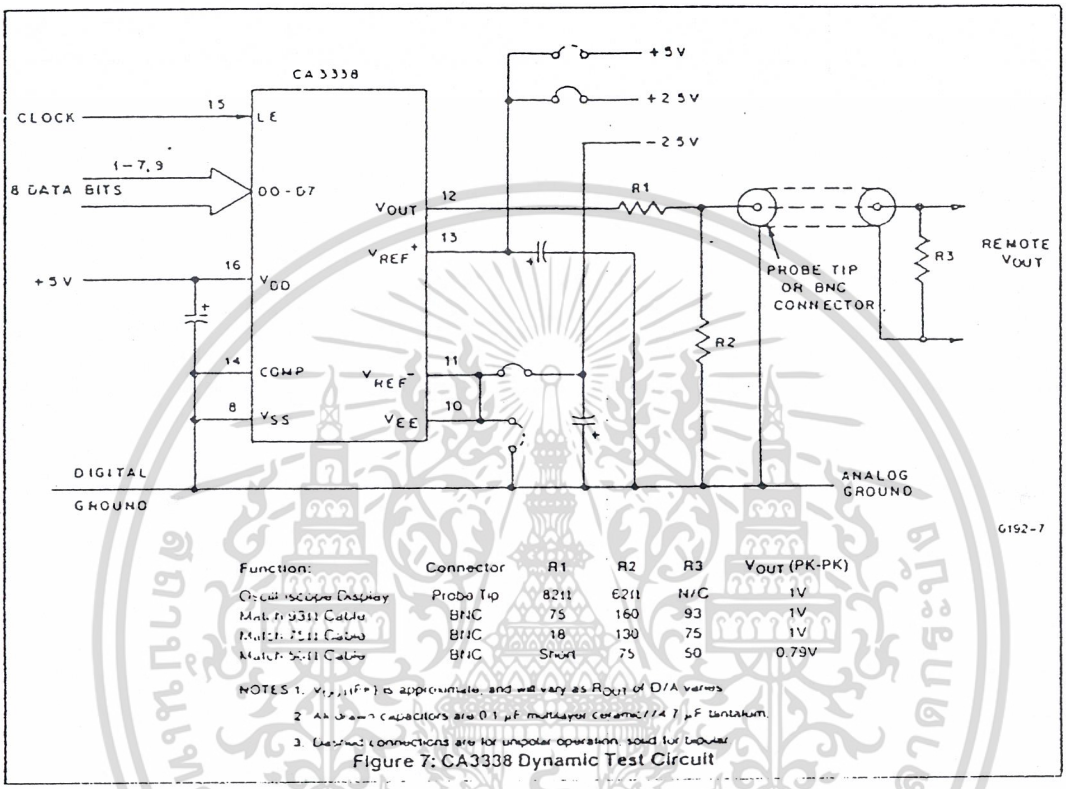
The settling time of the A/D is mainly a function of the output resistance (approximately 160Ω in parallel with the load resistance) and the load plus internal chip capacitance. Both "glitch" energy and settling time measurements require very good circuit and probe grounding; a probe tip connector such as Tektronix part number 131-0258-00 is recommended.

APPLICATIONS

The output of the CA3338 can be resistively divided to match a doubly terminated 50Ω or 75Ω line, although peak-to-peak swings of less than 1V may result. The output magnitude will also vary with the converter's output impedance. Figure 7 shows such an application. Note that because of the HCT input structure, the CA3338 could be operated up to +7.5V V_{DD} and V_{REF}^+ supplies and still accept 0V to 5V CMOS input voltages.

If larger voltage swings or better accuracy is desired, a high speed output buffer, such as the HA-5033, HA-2542, or CA3450, can be employed. Figure 8 shows a typical application, with the output capable of driving ±2V into multiple 50Ω terminated lines.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



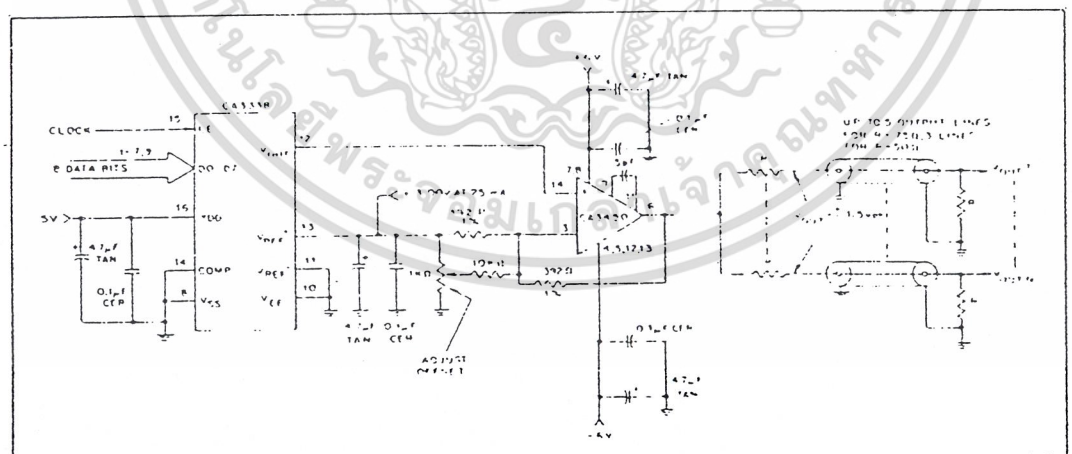
Function:	Connector	R1	R2	R3	VOUT (PK-PK)
Oscilloscope Display	Probe Tip	821Ω	621Ω	N/C	1V
Multiply 93Ω Cable	BNC	75	160	93	1V
Multiply 75Ω Cable	BNC	18	130	75	1V
Multiply 50Ω Cable	BNC	Short	75	50	0.79V

NOTES 1. V_{OUT} (FS) is approximate, and will vary as R_{OUT} of D/A varies.

2. All bypass capacitors are 0.1 µF multilayer ceramic / 4.7 µF tantalum.

3. Dashed connections are for unipolar operation; solid for bipolar.

Figure 7: CA3338 Dynamic Test Circuit



NOTES 1: Both V_{REF} + pin and 392Ω resistor should be bypassed within ¼ in.

2: Keep output capacitance at CA3450 pin 3 as low as possible.

3: V_{OUT} Range = ±2V at CA3450.

Figure 8: CA3338 and CA3450 for Driving Multiple Coaxial Lines

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 2: Output Voltage vs Input Code and V_{REF}

V_{REF}^+ V_{REF}^- Step Size	5.12V 0 0.0200V	5.00V 0 0.0195V	4.608V 0 0.0180V	2.56V - 2.56V 0.0200V	2.50V - 2.50V 0.0195V
Input Code					
$1111\ 1111_2 = FF_{16}$	5.1000V	4.9805V	4.5900V	2.5400V	2.4805V
$1111\ 1110_2 = FE_{16}$	5.0800	4.9610	4.5720	2.5200	2.4610
•					
•					
•					
$1000\ 0001_2 = 81_{16}$	2.5800	2.5195	2.3220	0.0200	0.0195
$1000\ 0000_2 = 80_{16}$	2.5600	2.5000	2.3040	0.0000	0.0000
$0111\ 1111_2 = 7F_{16}$	2.5400	2.4805	2.2860	- 0.0200	- 0.0195
•					
•					
•					
$0000\ 0001_2 = 01_{16}$	0.0200	0.0195	0.0180	- 2.5400	- 2.4805
$0000\ 0000_2 = 00_{16}$	0.0000	0.0000	0.0000	- 2.5600	- 2.5000

OPERATING AND HANDLING OPERATIONS

1. Handling

   All inputs and outputs of CMOS devices have a network for electrostatic protection during handling. Recommended handling practices for CMOS devices are described in ICAN-6525, "Guide to Better Handling and Operation of CMOS Integrated Circuits."

2. Operating

   Operating Voltage

   During operation near the maximum supply voltage limit, care should be taken to avoid or suppress power supply turn-on and turn-off transients, power supply ripple, or ground noise; any of these conditions must not cause the absolute maximum ratings to be exceeded.

   Input Signals

   To prevent damage to the input protection circuit, input signals should never be greater than V_{DD} nor less than V_{SS}. Input currents must not exceed 20 mA even when the power supply is off.

   Unused Inputs

   A connection must be provided at every input terminal. All unused input terminals must be connected to either V_{CC} or GND, whichever is appropriate.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

บุญยอด แสงคุณธรรม และคณะ. “การเข้ารหัสเสียงโดยวิธีการควอนไตซ์เวกเตอร์ด้วยอุปกรณ์ FPGAs.” ปริญญานิพนธ์ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ คณะครุศาสตร์อุตสาหกรรม, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2540

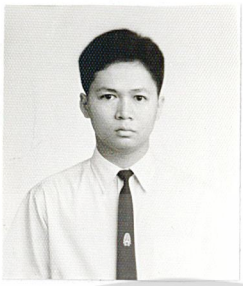
Xilinx. **The Progamable logic data Book** : 3 rd ed. Xilinx Inc. 1994

Viewlogic System. **Workview PLUS on Windows** : Viewlogic System Inc. 1993

Xilinx. **XACT Hardware & Peripherals Guide** : Xilinx Inc. 1994

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปริญญานิพนธ์	นายดาบชัย พัวพานิช
วันเดือนปีเกิด	19 เมษายน 2520
สถานที่เกิด	จังหวัดลพบุรี
ภูมิลำเนาเดิม	130/538 หมู่ที่ 3 ถ.นารายณ์มหาราช ต.ทะเลชุบศร อ.เมือง จ.ลพบุรี
ที่อยู่ปัจจุบัน	257 หมู่ที่ 1 ถ.อ่อนนุช แขวงลาดกระบัง เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(036) 610796
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนบรรจงรัตน์
มัธยมศึกษาตอนต้น	โรงเรียนวินิตศึกษาในพระราชูปถัมภ์สมเด็จพระเทพ รัตนราชสุดา สยามบรมราชกุมารี
ประกาศนียบัตรวิชาชีพ	วิทยาลัยเทคนิคลพบุรี
ประกาศนียบัตรวิชาชีพชั้นสูง	สถาบันเทคโนโลยีราชมงคล วิทยาเขตภาคตะวันออกเฉียงเหนือ นครราชสีมา
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	ทำหน้าที่ให้ดีที่สุด

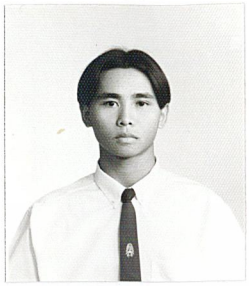
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อผู้ทำปริญญานิพนธ์	นายวรพงศ์ เลปนานนท์
วันเดือนปีเกิด	16 สิงหาคม 2519
สถานที่เกิด	จังหวัดสมุทรสงคราม
ภูมิลำเนาเดิม	66/2 ต.แม่กลอง อ.เมือง จ.สมุทรสงคราม
ที่อยู่ปัจจุบัน	777 หมู่ที่ 1 ถ.อ่อนนุช แขวงลาดกระบัง
	เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(034) 713038
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนเมืองสมุทรสงคราม
มัธยมศึกษาตอนต้น	โรงเรียนถาวรานุกูล
ประกาศนียบัตรวิชาชีพ	วิทยาลัยเทคนิคสมุทรสงคราม
ประกาศนียบัตรวิชาชีพชั้นสูง	วิทยาลัยเทคนิคสมุทรสงคราม
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม
	ภาควิชาครุศาสตร์วิศวกรรม
	คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	ความกตัญญูเป็นเครื่องหมายของคนดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปริญญานิพนธ์	นายวิระศักดิ์ ตรีโอษฐ์
วันเดือนปีเกิด	9 มีนาคม 2518
สถานที่เกิด	จังหวัดร้อยเอ็ด
ภูมิลำเนาเดิม	121 หมู่ที่ 5 ต.ชุมพร อ.เมยวดี จ.ร้อยเอ็ด
ที่อยู่ปัจจุบัน	333 หมู่ที่ 1 ถ.อ่อนนุช แขวงลาดกระบัง
	เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	-
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนบ้านห้วยทราย
มัธยมศึกษาตอนต้น	โรงเรียนโพนทองวิทยายน
ประกาศนียบัตรวิชาชีพ	วิทยาลัยเทคนิคร้อยเอ็ด
ประกาศนียบัตรวิชาชีพชั้นสูง	สถาบันเทคโนโลยีราชมงคล
	วิทยาเขตภาคตะวันออกเฉียงเหนือ นครราชสีมา
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม
	ภาควิชาครุศาสตร์วิศวกรรม
	คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	ความพยายามอยู่ที่ไหน ความสำเร็จอยู่ที่นั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อผู้ทำปริญญานิพนธ์	นายศิริชัย จำรัสเลิศลักษณ์
วันเดือนปีเกิด	19 กรกฎาคม 2519
สถานที่เกิด	จังหวัดราชบุรี
ภูมิลำเนาเดิม	30 หมู่ที่ 5 ต.คอนไผ่ อ.ดำเนินสะดวก จ.ราชบุรี
ที่อยู่ปัจจุบัน	333 หมู่ที่ 1 ถ.อ่อนนุช แขวงลาดกระบัง เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	(032) 254430
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนวันทามารีอา ราชบุรี
มัธยมศึกษาตอนต้น	โรงเรียนสายธรรมจันทร์
ประกาศนียบัตรวิชาชีพ	โรงเรียนดรุณาราชบุรีโปลีเทคนิค
ประกาศนียบัตรวิชาชีพชั้นสูง	วิทยาลัยเทคนิคสมุทรสงคราม
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	ฝันให้ไกล พยายามไปให้ถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้