สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA

The Security System Using FPGA

โดย

นายภาณุวัฒน์ อุทธวงศ์

นายศุนย์เจน สกุลเจน

วพ.
ว1434ร
2549

เลขหมู่.....
เลขทะเบียน 72004
วัน,เดือน,ปี - 7 มิ.ย. 2550

b 11761702
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The Security System Using FPGA

Mr. Panuwat Uttawong

Mr. Soonjane Sakuljane

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA
	(The Security System Using FPGA)
ชื่อนักศึกษา	นายภาณุวัฒน์ อุทธวงศ์ รหัสนักศึกษา 46012186
	นายศูนย์เจน สกุลเจน รหัสนักศึกษา 46012199
อาจารย์ที่ปรึกษา	อาจารย์สรพงษ์ วชิรรัตนพรกุล
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2549

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

.....

(อาจารย์สรพงษ์ วชิรรัตนพรกุล)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA			
	(The Security System Using FPGA)			
ชื่อนักศึกษา	นายภาณุวัฒน์	อุทธวงศ์	รหัสนักศึกษา	46012186
	นายศูนย์เจน	สกุลเจน	รหัสนักศึกษา	46012199
อาจารย์ที่ปรึกษา	อาจารย์ สรพงษ์ วชิรรัตนพรกุล			
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต			
	สาขาวิศวกรรมสารสนเทศ			
ภาควิชา	วิศวกรรมสารสนเทศ			
ปีการศึกษา	2549			

บทคัดย่อ

โครงงานนี้นำเสนอระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA ซึ่งจะใช้ เซ็นเซอร์ ในการตรวจจับ เมื่อมีสิ่งผิดปกติเกิดขึ้น เช่นเมื่อมีการบุกรุก เข้ามายังเขตต้องห้าม ก็จะทำการส่งสัญญาณเตือน พร้อมกับทำแจ้งเตือนผ่านทางโทรศัพท์ไปยังหมายเลขที่เราได้ทำการตั้งไว้ ทำให้สามารถแจ้งเตือนได้อย่างรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	The Security System Using FPGA
Student	Mr. Panuwat Uttawong ID. 46012186
	Mr. Soonjane Sakuljane ID. 46012199
Advisor	Mr. Sorapong Wachirarattanapornkul
Graduate	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2006

ABSTRACT

In this project present THE SECURITY SYSTEM USING FPGA which use infrared sensor to defend and detect the invader on system's area when the detector detect the intruder, the system will immediately alarm and automatically placing a call via telephone line To ensure that the alarm will immediately alert.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้ สำเร็จเรียบร้อยได้ด้วยดี ก็ด้วยได้รับคำแนะนำ และชี้แนวทางในการศึกษาค้นคว้าข้อมูล รายละเอียด และช่วยแก้ไขในส่วนที่บกพร่อง ต่างๆ จาก อาจารย์สรพงษ์ วชิรรัตนพรกุล ซึ่งปริญญานิพนธ์นี้จะไม่สำเร็จลุล่วงได้เลย ถ้าไม่มีอาจารย์คอยให้คำแนะนำและช่วยเหลือ พระคุณของอาจารย์ คณะผู้จัดทำขอขอบคุณเป็นอย่างสูงมา ณ โอกาสนี้

ขอขอบคุณพี่โจ พี่ตั้ม ที่เสียสละเวลาคอยให้คำปรึกษาเรื่องโปรเจ็ค และช่วยแก้ไขปัญหาต่างๆ เป็นอย่างดีเสมอมา

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่คอยห่วงใย ให้กำลังใจและให้การสนับสนุนในทุกๆเรื่อง รวมทั้งขอขอบคุณ ญาติสนิทและน้องๆ ทุกคนที่เป็นกำลังใจพร้อมทั้งให้ความช่วยเหลือในด้านต่างๆ เสมอมาโดยตลอด

สุดท้ายขอขอบคุณเพื่อนๆ ที่ห้องโปรเจ็ค และเพื่อนๆ ห้อง 4FS ที่คอยช่วยเหลือกันไม่ทิ้งกันเลย แม้ว่าจะเหนื่อยกันซักแค่ไหน ก็ยังคอยมาช่วยอยู่ตลอด ขอบคุณเพื่อนๆ มาก

นาย ภาณุวัฒน์ อุทธวงศ์
นาย ศูนย์เจน สกุลเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
Abstract	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	จ
สารบัญตาราง	ฌ
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 ความมุ่งหมายและจุดประสงค์	1
1.3 ขอบเขตของโครงงาน	1
1.4 ขั้นตอนของการทำโครงงาน	2
1.5 เนื้อหาปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการ	
2.1 FPGA	3
2.2 การออกแบบวงจรดิจิตอลในปัจจุบัน	5
2.3 การออกแบบวงจรดิจิตอลด้วย FPGA	6
2.4 VHDL	7
2.5 ส่วนประกอบต่างๆ ของภาษา VHDL	11
2.6 การเขียนแบบลำดับชั้น (Hierarchical Model)	13
2.7 ความสามารถของภาษา VHDL	14
2.8 ทฤษฎีของโทรศัพท์	15
2.9 ทฤษฎีของไอซีบันทึกเสียงและเล่นกลับ	18
2.10 ทฤษฎีวงจรตรวจสอบสัญญาณโทรศัพท์	22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 3 การออกแบบระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA	
3.1 การออกแบบระบบ	23
3.2 วงจรเซ็นเซอร์ (Sensor)	23
3.3 วงจรไซเรน (Siren)	27
3.4 วงจรแจ้งเตือนทางโทรศัพท์	27
บทที่ 4 ผลการทดลอง	
4.1 ผลการทดลองการใช้งานในส่วนโปรแกรมของ FPGA	33
4.2 ผลการทดลองการใช้งานในส่วนโปรแกรมของ iMPACT	33
4.3 ผลการจำลองการทำงานของโปรแกรม	36
4.4 ผลการทดลองการทำงานของระบบ	44
บทที่ 5 สรุปและแนวทางในการพัฒนาต่อไป	
5.1 สรุปผลโครงงาน	49
5.2 ปัญหาในการทำงานโครงงาน	49
5.3 แนวทางในการพัฒนาต่อไป	49
บรรณานุกรม	50
ภาคผนวก ก วงจรอิเล็กทรอนิกส์	51
ภาคผนวก ข โปรแกรมของโครงงาน	56
ภาคผนวก ค รายละเอียดเฉพาะอุปกรณ์	68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 การแบ่งกลุ่มของวงจรรวม ASIC	3
รูปที่ 2.2 บอร์ด FPGA Discovery-III XC3S200	5
รูปที่ 2.3 การส่งข้อมูลผ่านสายดาวน์โหลดทางพอร์ทของคอมพิวเตอร์	7
รูปที่ 2.4 โครงสร้างของภาษา VHDL	7
รูปที่ 2.5 กระบวนการออกแบบระบบอิเล็กทรอนิกส์	9
รูปที่ 2.6 กระบวนการออกแบบลักษณะ TOP DOWN DESIGN	9
รูปที่ 2.7 บล็อกไดอะแกรมของวงจรโทรศัพท์	15
รูปที่ 2.8 แป้นกดของโทรศัพท์	17
รูปที่ 2.9 วงจรภายในไอซีบันทึกเสียง	19
รูปที่ 2.10 การวางขาของไอซีบันทึกเสียงเบอร์ ISD2560/75/90/120	21
รูปที่ 2.11 วงจรภายในไอซี LM567	22
รูปที่ 3.1 บล็อกไดอะแกรมของระบบ	23
รูปที่ 3.2 แผนผังโปรแกรม	24
รูปที่ 3.3 วงจรตัวส่ง EPS937TX	25
รูปที่ 3.4 วงจรตัวรับสัญญาณ EPS937RX	26
รูปที่ 3.5 แบบการตรวจจับของ เซนเซอร์	26
รูปที่ 3.6 วงจรไซเรน	27
รูปที่ 3.7 บล็อกไดอะแกรมวงจรภายในไอซี	27
รูปที่ 3.8 วงจรตัดต่อคู่สายโทรศัพท์	28
รูปที่ 3.9 วงจรตรวจจับสัญญาณโทรศัพท์	29
รูปที่ 3.10 สัญญาณโทรศัพท์ขณะไม่มีการรับสาย	30
รูปที่ 3.11 สัญญาณโทรศัพท์ขณะมีการรับสาย	30
รูปที่ 3.12 ตำแหน่งการวางขาไอซี	31
รูปที่ 3.13 วงจรของการออกแบบไอซี	32
รูปที่ 4.1 ส่วนโปรแกรมของ ISE WebPack 8.1i	33
รูปที่ 4.2 ส่วนโปรแกรมของ iMPACT	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูปที่ 4.3 ส่วนโปรแกรมของ iMPACT ในขณะเปิดไฟล์	34
รูปที่ 4.4 ส่วนโปรแกรมของ iMPACT ขณะเลือกทำการเบิร์นโปรแกรม	35
รูปที่ 4.5 ส่วนโปรแกรมของ iMPACT ขณะทำการเบิร์นโปรแกรม	35
รูปที่ 4.6 ส่วนโปรแกรมของ iMPACT เมื่อทำการเบิร์นเสร็จสมบูรณ์	35
รูปที่ 4.7 วงจรของโปรแกรมตรวจจับการเคลื่อนไหว	36
รูปที่ 4.8 ผลการจำลองของโปรแกรมตรวจจับการเคลื่อนไหว	36
รูปที่ 4.9 วงจรของโปรแกรม Count5	37
รูปที่ 4.10 ผลการจำลองการทำงานของโปรแกรม Count5	37
รูปที่ 4.11 วงจรของโปรแกรมตัดต่อคู่สายโทรศัพท์	38
รูปที่ 4.12 ผลการจำลองการทำงานของโปรแกรมตัดต่อคู่สายโทรศัพท์	38
รูปที่ 4.13 วงจรของโปรแกรมกดหมายเลขโทรศัพท์	39
รูปที่ 4.14 ผลการจำลองการทำงานของโปรแกรมกดหมายเลขโทรศัพท์	40
รูปที่ 4.15 วงจรของโปรแกรมตรวจจับสัญญาณโทรศัพท์	41
รูปที่ 4.16 ผลการจำลองการทำงานของโปรแกรมตรวจจับสัญญาณโทรศัพท์	41
รูปที่ 4.17 วงจรของโปรแกรมเล่นเสียง	42
รูปที่ 4.18 ผลการจำลองการทำงานของโปรแกรมเล่นเสียง	42
รูปที่ 4.19 ผลการจำลองการทำงานของโปรแกรมทั้งหมด	43
รูปที่ 4.20 ภาพรวมของโครงงาน	44
รูปที่ 4.21 แสดงการติดตั้งเซ็นเซอร์	44
รูปที่ 4.22 แสดงเมื่อมีการบุกรุกเข้า	45
รูปที่ 4.23 LED เมื่ออยู่ในสภาวะปกติ	45
รูปที่ 4.24 เซ็นเซอร์ตรวจจับการบุกรุกได้	46
รูปที่ 4.25 วงจรเซ็นเซอร์	46
รูปที่ 4.26 วงจรไซเรน	47
รูปที่ 4.27 วงจรแจ้งเตือนทางโทรศัพท์	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูปที่ 4.28 เมื่อมีการโทรศัพท์เข้ามาจากวงจรแจ้งเตือนทางโทรศัพท์	48
รูปที่ ก.1 วงจรรวมของโครงงาน	52
รูปที่ ก.2 วงจรเซ็นเซอร์	53
รูปที่ ก.3 วงจรไซเรน	54
รูปที่ ก.4 วงจรตัดต่อคู่สายโทรศัพท์	54
รูปที่ ก.5 วงจรตรวจจับสัญญาณโทรศัพท์	54
รูปที่ ก.6 วงจรบันทึกเสียงและเล่นกลับ	55
รูปที่ ข.1 ภาพรวมของวงจรที่โปรแกรมลง FPGA	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 1.1 ขั้นตอนของการทำโครงงาน	2
ตารางที่ 2.1 หน้าที่สำคัญของอุปกรณ์บันทึกเสียง	22
ตารางที่ 3.1 คุณสมบัติของไอซีบันทึกเสียงตระกูล ISD25xx	31
ตารางที่ 3.2 ข้อความที่บันทึกในตัวไอซี	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันนี้สภาพความเป็นอยู่ของผู้คนโดยเฉพาะผู้ที่อยู่ภายในเมืองใหญ่ๆ เช่น กรุงเทพฯ และเขตปริมณฑลโดยรอบ จะพบว่า คนส่วนมาก นั้นจะต้องออกไปทำงานกัน นอกบ้าน ทำให้ไม่สามารถที่จะดูแลและระวังทรัพย์สินต่างๆ ภายในบ้านได้ตลอดเวลา แต่จะให้หาอุปกรณ์ที่ใช้ในการรักษาความปลอดภัยนั้นก็มีราคาแพงอยู่ โครงงานนี้จึงได้นำเสนอระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA อันเป็นระบบที่มีราคาที่ไม่แพง สามารถใช้งานได้โดยง่าย ประกอบกับในปัจจุบันนี้โทรศัพท์มือถือ ได้กลายเป็นส่วนหนึ่งของชีวิตประจำวันของทุกๆ คนไปแล้ว โครงงานนี้จึงได้เล็งเห็นถึงความสะดวกและรวดเร็วในการที่จะเตือนภัยผ่านทางโทรศัพท์มือถือ โดยเมื่อเกิดการบุกรุก หรือเข้ามาในสถานที่ต้องห้ามก็จะมีการแจ้งเตือนผ่านทางเสียงไซเรน และแจ้งเตือนผ่านทางโทรศัพท์มือถือ ทำให้การป้องกันภัยสามารถทำได้อย่างรวดเร็ว

1.2 ความมุ่งหมายและจุดประสงค์

1. เพื่อทำการศึกษาการเขียนภาษา VHDL ในการออกแบบวงจรทางลอจิก
2. เพื่อศึกษาการใช้งานและการทำงานของอุปกรณ์ FPGA
3. เพื่อการศึกษาและทำการออกแบบวงจร FPGA ร่วมกับวงจรเซ็นเซอร์ และวงจรแจ้งเตือนทางโทรศัพท์เพื่อใช้ในการรักษาความปลอดภัยราคาถูกได้

1.3 ขอบเขตของโครงงาน

โครงงานนี้เป็นการนำเสนอระบบดูแลและรักษาความปลอดภัยโดยใช้บอร์ด FPGA เป็นตัวควบคุมหลักของระบบ ร่วมกับส่วนที่เป็นเซ็นเซอร์ (Sensor) ที่ใช้ในการตรวจจับผู้บุกรุก เพื่อส่งสัญญาณไปแจ้งยัง FPGA เมื่อมีการบุกรุก โดย FPGA จะทำหน้าที่ส่งสัญญาณเตือนภัย และแจ้งเตือนผ่านระบบโทรศัพท์ไปยังเลขหมายที่ได้กำหนดเอาไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนของการทำโครงงาน

ตารางที่ 1.1 ขั้นตอนของการทำโครงงาน

ขั้นตอนของการทำโครงงาน	ช่วงระยะเวลา							
	2549							2550
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.
1. การกำหนดขอบเขตของโครงงาน	←	→						
2. ศึกษาทฤษฎีที่จำเป็นต้องใช้		←	→					
3. ทำการต่อวงจร และเขียนโปรแกรมลง FPGA			←			→		
4. ทำการทดสอบวงจร					←		→	
5. นำข้อมูลที่ได้จากการทดสอบมารวมรวบและสรุปผล					←			→

1.5 เนื้อหาปริญญานิพนธ์

ในบทแรกนี้เป็นการบอกถึงความสำคัญและความเป็นมาของโครงงานและขอบเขตของโครงงาน

ต่อมาบทที่ 2 จะเป็นทฤษฎีที่เกี่ยวข้องในการทำโครงงานนี้ รวมไปถึง ความเป็นมาของ FPGA และ ภาษา VDHL

ในบทที่ 3 จะอธิบายถึงหลักการในการออกแบบ และการทำงานของระบบ

ในบทที่ 4 จะเป็นผลการทดลอง ประกอบไปด้วยการทดลองวงจรเซ็นเซอร์ วงจรแจ้งเตือนทางโทรศัพท์ และโปรแกรมที่ใช้ในการควบคุม

ในบทที่ 5 เป็นการสรุปผลจากการดำเนินงานของโครงงาน

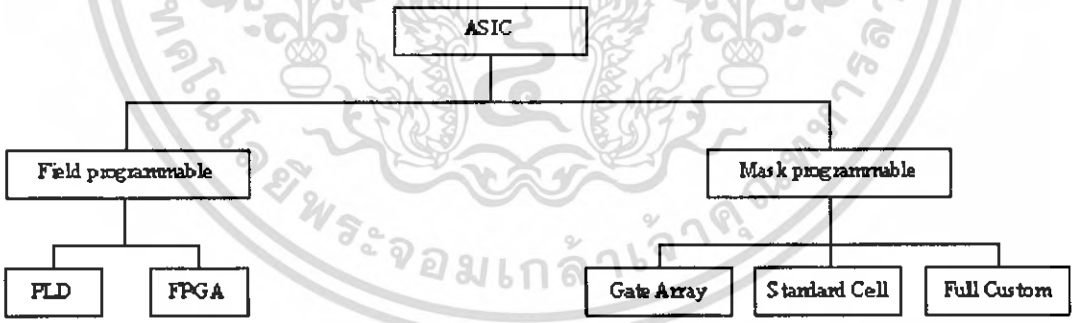
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 FPGA

2.1.1 บทนำ

เนื่องจากในปัจจุบันอุตสาหกรรมทางด้านอิเล็กทรอนิกส์ได้มีความก้าวหน้าไปอย่างรวดเร็วมาก และอีกทั้งประสิทธิภาพที่เพิ่มขึ้นของอุปกรณ์อิเล็กทรอนิกส์ เพื่อให้รองรับกับความต้องการของการนำมาใช้กับงานที่มีความซับซ้อนมากขึ้นเรื่อยๆนั้น สิ่งสำคัญที่สุดย่อมขึ้นอยู่กับประสิทธิภาพของชิ้นส่วนแต่ละชิ้นที่อยู่ภายในอุปกรณ์อิเล็กทรอนิกส์ โดยส่วนประกอบที่สำคัญอย่างมากซึ่งก็คือ วงจรรวม หรือที่เรารู้จักกันดีในนามว่า ไอซี (IC, Integrated Circuit) หรือ ไมโครชิป สำหรับตัวอย่างของการพัฒนาเทคโนโลยีไอซีที่เราสามารถพบเห็นได้อย่างชัดเจน นั้น เช่น เทคโนโลยีไมโครโปรเซสเซอร์ (MPU), ไมโครคอนโทรลเลอร์ (MCU) และหน่วยความจำ (Memory) ในปัจจุบันนี้มีไอซีหลายชนิดเพื่อให้เลือกนำมาใช้งานกัน โดยเฉพาะอย่างยิ่งไอซีที่มีประสิทธิภาพสูงๆ ที่ใช้ในงานเฉพาะด้านหรือที่เรียกว่า ASIC (Application Specific Intergrated Circuit) ซึ่งจะสามารถแบ่งออกเป็นตามโครงสร้างได้เป็น 2 กลุ่มใหญ่ๆ คือ Field Programmable และ Mask Programmable



รูปที่ 2.1 การแบ่งกลุ่มของวงจรรวม ASIC

ถ้าจะพิจารณาถึงข้อดี ในเชิงการออกแบบและการผลิตแล้ว ไอซีตระกูล Mask programmable นั้นอาจจะทำให้ได้ชิป ASIC ที่มีประสิทธิภาพสูงกว่า แต่ต้องมีการส่งวงจรไปทำหน้ากาก (mask set) และทำการเจือสารยังต่างประเทศ จึงทำให้มีค่าใช้จ่ายเริ่มต้นที่สูงมาก เหมาะกับการใช้งานที่ต้องการปริมาณสูง แต่สำหรับในการพัฒนาชิป ASIC ในปริมาณน้อยถึงปานกลางนั้น ควรใช้ชิป ASIC ชนิดโปรแกรมได้ (field programmable) มากกว่า เพราะ เนื่องจากผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถออกแบบและสร้างวงจรที่ต้องการใช้ลงในตัวอุปกรณ์ได้เอง โดยไม่ต้องไปผลิตที่โรงงาน จึงเป็นทางเลือกที่น่าสนใจโดยเฉพาะอย่างยิ่ง เทคโนโลยีที่ก้าวหน้าไปมาก ทำให้ประสิทธิภาพของไอซีแบบโปรแกรมได้สามารถตอบสนอง ต่อความต้องการของการใช้งานที่มีความซับซ้อนสูงได้

2.1.2 FPGA (Field Programmable Gate Array)

เป็นอุปกรณ์ที่ใช้ในการโปรแกรมวงจรที่ได้ออกแบบลงไปเพื่อให้อุปกรณ์ FPGA มีฟังก์ชั่นการทำงานตามที่ผู้ออกแบบต้องการ โดยมีจุดเด่นที่ประสิทธิภาพการทำงานและปริมาณความหนาแน่นของจำนวนเกตสูง การใช้งานที่สะดวก สามารถกำหนดฟังก์ชั่นการทำงานได้ตามความต้องการของผู้ใช้งาน โดยผ่านทางโปรแกรม ดังนั้นการออกแบบวงจรโดยใช้ FPGA จะสามารถทำได้ในเวลาไม่นาน และยังสามารถทำการแก้ไขได้โดยง่าย ทำให้ประหยัดค่าใช้จ่ายในการออกแบบวงจรเป็นอย่างมาก และสามารถตอบสนองต่อการใช้งานที่ได้ซับซ้อนได้เป็นอย่างดี

2.1.3 เปรียบเทียบกับไมโครคอนโทรลเลอร์ (Microcontroller)

ไมโครคอนโทรลเลอร์ เป็นชิปประเภทใช้ประโยชน์ทั่วไป ซึ่งหมายความว่าสถาปัตยกรรม รวมถึงชุดคำสั่งทั้งหมดของชิปนั้นๆ ได้ถูกออกแบบและกำหนดมาเรียบร้อยแล้ว ผู้ใช้มีหน้าที่เพียงเรียบเรียงชุดคำสั่ง เพื่อให้ชิพปฏิบัติหน้าที่ตามที่ต้องการ ซึ่งไมโครคอนโทรลเลอร์จะไม่สามารถตอบสนองต่องานที่มีความซับซ้อนได้ ซึ่งต่างกับ FPGA ที่ผู้ใช้เป็นคนออกแบบสถาปัตยกรรมทั้งหมดเอง นอกจากนั้นผู้ใช้ยังสามารถออกแบบชุดคำสั่งใหม่ๆ ขึ้นมาเองเพื่อให้เหมาะกับการใช้งานนั้นๆ ได้มากกว่า

2.1.4 จุดเด่นของ FPGA

นอกจากระยะเวลาที่ใช้ในการออกแบบและพัฒนาจะเร็วใกล้เคียงกับการใช้ ชิปประเภทใช้ประโยชน์ทั่วไป ขณะที่ประสิทธิภาพใกล้เคียงกับชิปประเภท ASIC แบบ full-custom ซึ่งหมายถึงชิปที่ถูกออกแบบและพัฒนาใหม่ทั้งหมดเฉพาะงานนั้นๆ ก็มีราคาแพงกว่ามากแล้ว ผู้ใช้ชิป FPGA ยังสามารถนำวงจรที่พร้อมใช้งานที่มีผู้ออกแบบไว้แล้ว ในรูปของทรัพย์สินทางปัญญาหรือ มาใช้รวมกับวงจรที่เราออกแบบเองได้ทันที ซึ่งเป็นการเพิ่มประสิทธิภาพ และย่นเวลาการพัฒนาชิป FPGA ของเราได้มาก

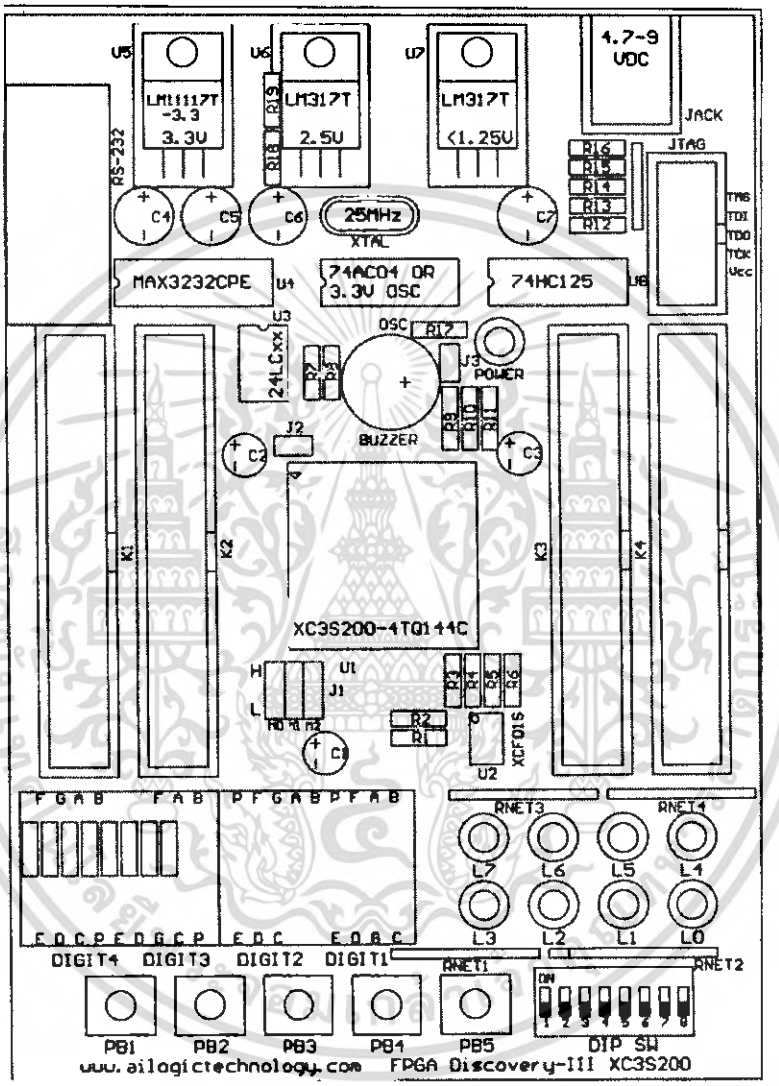
2.1.5 FPGA ที่มีอยู่ในท้องตลาดและการเลือกใช้

ปัจจุบันมีผู้ผลิตชิป FPGA อยู่หลายบริษัท เช่น Xilinx Corp., Altera หรือ Actel โดยบริษัทที่มีขนาดใหญ่ที่สุด และมีส่วนแบ่งทางการตลาดสูงที่สุดคือ Xilinx ซึ่งปกติ FPGA ของแต่ละบริษัทจะมีหลายตระกูลขึ้นอยู่กับราคาและขนาดความจุของชิป FPGA ซึ่งมักถูกวัดด้วยค่าจำนวนเกตเทียบเท่า โดยปัจจุบัน FPGA จะมีขนาดตั้งแต่ น้อยกว่า 10,000 เกตจนถึงมากกว่า 5,000,000 เกต ขึ้นอยู่กับตระกูลของ FPGA โดยความจุของเกทที่มากกว่าก็ หมายถึง ศักยภาพในการใช้งาน FPGA ตัวนั้นสามารถใช้กับงานที่ซับซ้อนได้มากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 บอร์ดที่ใช้ในโครงงาน

จะเป็นบอร์ด FPGA รุ่น Discovery-III XC3S200 ที่ผลิตโดย บริษัท Apex Instrument ซึ่งจะใช้ชิป FPGA ของ Xilinx ตระกูล Spartan 3 โดยมีขนาดความจุ 200,000 เกต เบอร์ XC3S200 ขาแบบ TQ144 Speed Grade-4



รูปที่ 2.2 บอร์ด FPGA Discovery-III XC3S200

2.2 การออกแบบวงจรดิจิตอลในปัจจุบัน

2.2.1 การออกแบบโดยใช้การวาดผังวงจร (Schematic design)

ในอดีตการออกแบบวงจรดิจิตอลที่เรารู้จักกัน จะเป็นการออกแบบวงจรในระดับลอจิกเกต (Logic Level) โดยอาศัยโปรแกรมช่วยในการวาดผังวงจร ซึ่งการออกแบบวงจรลักษณะนี้ ค่อนข้างใช้เวลามากในการออกแบบ เนื่องจากการออกแบบทุกๆ อย่างผู้ออกแบบจะต้องออกแบบเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งหมด ซึ่งถ้าวงจรที่ต้องการออกแบบมีความซับซ้อนสูง ก็จะทำให้การออกแบบทำได้ยาก ซึ่งเป็นข้อจำกัดให้ผู้ออกแบบไม่สามารถออกแบบวงจรดิจิตอลที่มีความซับซ้อนสูงได้ เช่น การออกแบบไมโครโปรเซสเซอร์ และ ไมโครคอนโทรลเลอร์ เป็นต้น

2.2.2 การออกแบบโดยใช้ภาษาระดับสูง (HDL)

ในปัจจุบันเทคโนโลยีการออกแบบวงจรดิจิตอล ได้พัฒนาสูงขึ้นและได้มีการพัฒนาภาษาที่ใช้สำหรับออกแบบวงจรดิจิตอล พร้อมกระบวนการออกแบบแนวใหม่ ที่จะทำให้ผู้ออกแบบไม่ถูกจำกัดในการออกแบบวงจรอีกต่อไป ซึ่งการออกแบบวงจรดิจิตอลแนวใหม่นี้ จะใช้การเขียนภาษาบรรยายพฤติกรรมฮาร์ดแวร์ (HDL : Hardware Description Language) มาใช้ในการออกแบบวงจร โดยภาษาจะบรรยายพฤติกรรมฮาร์ดแวร์ที่เป็นมาตรฐานซึ่งมีอยู่ 2 ภาษาที่นิยมใช้ด้วยกันคือ ภาษา Verilog และ VHDL โดยถ้านำมาเปรียบเทียบกันในโลกของการออกแบบซอฟต์แวร์ (Software) ก็เปรียบได้กับ ภาษา C และภาษา Pascal นั่นเอง โดยที่ภาษา Verilog จะมีโครงสร้างคล้ายกับ ภาษา C ส่วนภาษา VHDL จะมีโครงสร้างคล้ายภาษา Pascal

2.3 การออกแบบวงจรดิจิตอลด้วย FPGA โดยทั่วไปมีองค์ประกอบ 3 ส่วน

2.3.1 ซอฟต์แวร์ Design Entry

- โดยใช้ Schematic Design Entry ใช้ไลบารีของ FPGA

- ใช้ภาษา HDL เช่น VHDL, Verilog, การออกแบบโดยใช้ภาษาชั้นสูงนั้นการทดสอบของวงจรนั้นยังไม่ขึ้นกับเทคโนโลยีเป้าหมาย (Technology independent) ผู้ออกแบบไม่จำเป็นต้องกังวลถึงค่าความหน่วงทางเวลาของอุปกรณ์ที่มากับเทคโนโลยีนั้นการทดสอบความถูกต้องเป็นในลักษณะการตรวจสอบระดับฟังก์ชันการทำงานโดยใช้ซอฟต์แวร์สำหรับจำลองการทำงาน (Simulation)

2.3.2 Design Implementation

ขั้นตอนนี้ ต่อเนื่องจากขั้นตอนที่ 2.3.1 ซึ่งเกี่ยวข้องกับการแปลงแบบที่ได้ออกแบบจาก Schematic หรือ HDL ให้เป็นลอจิก ซึ่งอาจใช้ซอร์ฟแวร์สำหรับสังเคราะห์วงจร (Logic synthesis) แล้วทำการแบ่งลอจิกเป็นส่วนๆ (Partitioning) และวางตำแหน่ง (Placement) ของลอจิกทำการเชื่อมต่อสายสัญญาณ (routing) สุดท้ายเป็นการสร้างไฟล์สำหรับโปรแกรมลงชิป (bit file)

2.3.3 Device Programming

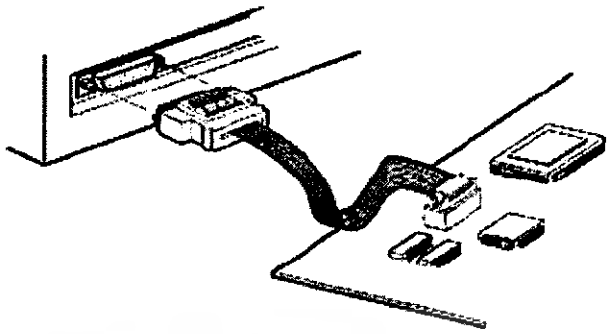
การโปรแกรมอุปกรณ์หรือชิป FPGA นั้น มีเทคนิคหรือวิธีใหญ่ๆ 3 ลักษณะ ทั้งนี้ตัวชิปจะต้องสนับสนุนการทำงานในโหมดของการโปรแกรมเหล่านี้ด้วย

- การโปรแกรมโดยผ่านสายดาวน์โหลด หรือผ่าน JTAG หรือผ่าน ISP

- การโปรแกรมโดยใช้ ตัวเก็บข้อมูลไฟล์บิต

- การโปรแกรมโดยใช้เครื่องโปรแกรมไอซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

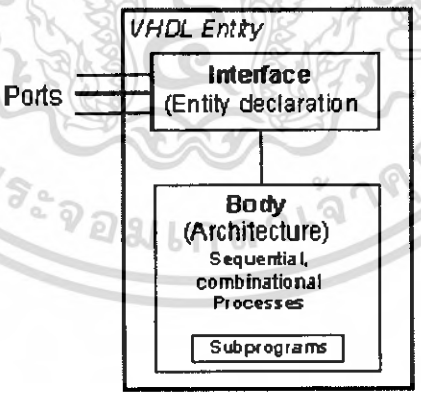


รูปที่ 2.3 การส่งข้อมูลผ่านสายดาวน์โหลดทางพอร์ทของคอมพิวเตอร์

2.4 VHDL

2.4.1 บทนำ

VHDL ย่อมาจากคำว่า VHSIC Hardware Description Language (VSHIC : Very High Speed Integrated Circuit) เป็นภาษาโปรแกรมระดับสูง (High level language) โดย VHDL เป็นภาษาที่ใช้ในการบรรยายหรืออธิบายรูปแบบการทำงานและความสัมพันธ์ของอุปกรณ์ฮาร์ดแวร์ทั้งในระดับเกตจนถึงระดับดิจิตอลที่ซับซ้อน ทั้งนี้เนื่องจาก VHDL เป็นภาษาที่ง่ายต่อการเรียนรู้ อีกทั้งยังมีความยืดหยุ่นและไม่ถูกจำกัดโดย ความสามารถทางเทคโนโลยีใดๆ ทำให้ประหยัดเวลาและค่าใช้จ่ายในการออกแบบ จึงได้มีการนำมาใช้กันอย่างกว้างขวางในวงการอิเล็กทรอนิกส์และอุตสาหกรรม



รูปที่ 2.4 โครงสร้างของภาษา VHDL

2.4.2 ประวัติความเป็นมาของภาษา VHDL

เริ่มต้นตั้งแต่ประมาณ ปี ค.ศ. 1981 โดยที่กระทรวงกลาโหมสหรัฐอมริกา หรือ DoD (Department of Defense) ได้จัดตั้งคณะทำงานขึ้นมาคณะหนึ่ง เพื่อทำการพัฒนาภาษาที่ใช้ในการบรรยายหรืออธิบายรูปแบบการทำงานและความสัมพันธ์ของอุปกรณ์ฮาร์ดแวร์(Hardware) แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใหม่ขึ้น ซึ่งผลจากการทำงานของคณะชุดนี้ได้ก่อให้เกิดภาษาการบรรยายฮาร์ดแวร์ ขึ้นเรียกว่า VHDL (VHSIC Hardware Description Language) โดย VHSIC เป็นชื่อย่อของแผนกหนึ่งของสถาบันที่ทำงานเกี่ยวกับวงจรรวมที่มีความเร็วสูงมาก (Very High Speed Integrated Circuit) ต่อมาในปี 1985 IEEE ได้ทำการผลักดันให้ VHDL กลายเป็นภาษาที่เป็นมาตรฐานและมีการยอมรับกันอย่างกว้างขวางในวงการอุตสาหกรรมคอมพิวเตอร์

2.4.3 ลักษณะการใช้งานภาษา VHDL

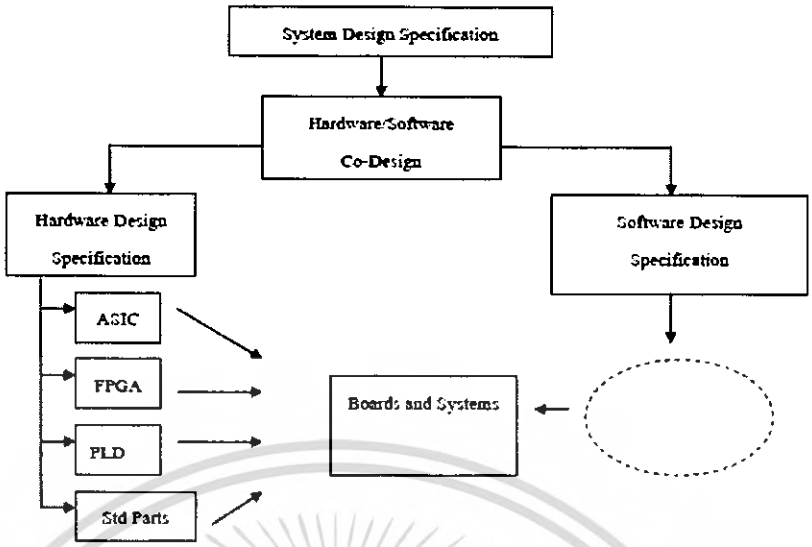
การใช้งานภาษา VHDL อาจจำแนกออกเป็น 5 ประเภทดังนี้คือ

- Document Language : เป็นภาษาที่ใช้อธิบายรายละเอียดการทำงานของวงจรที่ออกแบบ
- Design Language : เป็นภาษาที่ใช้สำหรับออกแบบวงจรที่มีความซับซ้อนสูง
- Verification Language : เป็นภาษาที่ใช้ตรวจสอบความถูกต้องของวงจรที่ออกแบบ
- Test Language : เป็นภาษาที่ใช้สำหรับทดสอบการทำงานของวงจรที่ออกแบบ
- Synthesis Language : เป็นภาษาที่ใช้สำหรับสังเคราะห์วงจร (Synthesis) จริง แต่รูปแบบดังกล่าวนี้ไม่ได้ครอบคลุมทุกรูปแบบทั้งหมดของการเขียนในภาษา VHDL แต่มีเพียงบางรูปแบบเท่านั้นที่สามารถเขียนสามารถนำไปสังเคราะห์เป็นวงจรได้ หรือเรียกว่าเป็น รูปแบบการเขียนสำหรับการสังเคราะห์ (VHDL for Synthesis)

2.4.4 วิธีการออกแบบระบบอิเล็กทรอนิกส์

การออกแบบระบบอิเล็กทรอนิกส์ จะเริ่มต้นจากการกำหนดหน้าที่การทำงานของระบบอิเล็กทรอนิกส์ที่ต้องการ ซึ่งโดยส่วนใหญ่แล้วระบบอิเล็กทรอนิกส์มักจะประกอบด้วย 2 ส่วนคือ ส่วนที่เป็นระบบฮาร์ดแวร์ และส่วนที่เป็นซอฟต์แวร์ โดยทั่วไปแล้วการออกแบบทั้งสองส่วนดังกล่าวนี้จะต้องทำไปพร้อมๆ กัน หลังจากนั้นจะเป็นขั้นตอนในการกำหนดหน้าที่การทำงานของระบบฮาร์ดแวร์และระบบซอฟต์แวร์ โดยสำหรับการออกแบบทางด้าน ฮาร์ดแวร์ สามารถเลือกได้ว่าต้องการออกแบบเป็น ASIC ,FPGA ,PLD หรือจากไอซีมาตรฐานที่มีขายอยู่ทั่วไปตามท้องตลาด ส่วนการออกแบบระบบซอฟต์แวร์ ผู้ออกแบบสามารถเลือกใช้ภาษาโปรแกรมต่างๆ นำไปออกแบบ เช่น ภาษาC/C++ ภาษา Assembly ส่วนสำหรับในการออกแบบ Firmware และ Application โดยในระหว่างการออกแบบนั้น ผู้ออกแบบทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์จะต้องมีการทดสอบการทำทำงานร่วมกันเพื่อทำให้สามารถแก้ไขความผิดพลาดในการออกแบบที่อาจเกิดขึ้นได้

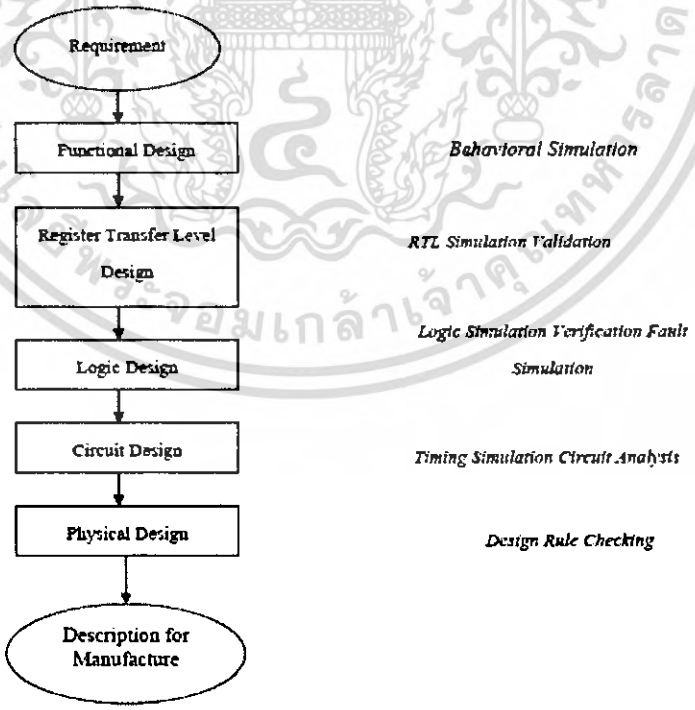
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 กระบวนการออกแบบระบบอิเล็กทรอนิกส์

2.4.5 กระบวนการออกแบบลักษณะ Top-Down Design

กระบวนการออกแบบลักษณะ Top-Down Design เป็นกระบวนการที่ใช้ในการออกแบบระบบดิจิตอลในปัจจุบัน โดยมีขั้นตอนต่างๆ ดังรูปที่ 2.6



รูปที่ 2.6 กระบวนการออกแบบลักษณะ Top-Down Design

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนแรกคือ การวางข้อกำหนดคุณลักษณะของระบบ (Requirements) ทั้งด้านสมรรถนะและด้านกายภาพ และจากข้อกำหนดต่างๆ ในการกำหนดด้านคุณลักษณะของระบบเราก็จะสามารถนำมาออกแบบเชิงหน้าที่ระดับสูง (High-Level function design) ในเบื้องต้นได้ ซึ่งต้องอาศัยการจำลองการทำงาน (Simulation) เพื่อดูว่าระบบที่ออกแบบไว้ทำงานสอดคล้องกับข้อกำหนดต่างๆ หรือไม่ เพื่อแก้ไขให้สอดคล้องในที่สุด

เมื่อเสร็จสิ้นการออกแบบเชิงหน้าที่ (Function design) แล้วขั้นต่อไปคือ การใส่รายละเอียดลงไปในหน่วยความจำ (Memory) หน่วยคำนวณและหน่วยตรรกะ (ALU) และเครื่องสถานะ (State machine) เราเรียก การออกระบบในระดับนี้ว่า Register Transfer Level หรือ RTL หลังจากการออกแบบในระดับ RTL ก็จะเป็นการออกแบบในระดับตรรกะ (Logic design) ซึ่งเป็นการเพิ่มเติมรายละเอียดให้กับหน่วยต่างๆ ที่เป็นส่วนประกอบของ งานออกแบบระดับใน RTL เช่นการกำหนดในรายละเอียดว่า รีจิสเตอร์ตัวหนึ่งประกอบไปด้วยฟลิปฟลอบและ ลอจิกเกตแบบใดจำนวนเท่าใด เป็นต้นการออกแบบในระดับ RTL และระดับลอจิก (Logic) ต้องมีการจำลองการทำงานควบคู่ไปด้วยเช่นกัน เหมือนการออกแบบระดับ Functional ในบางระบบนอกจากการจำลองเชิงตรรกะ (Logic simulation) แล้ว อาจมีการจำลองเพื่อหาข้อบกพร่อง (Fault simulation) ของวงจรหรือระบบด้วย Fault simulation สามารถจำลองผลกระทบที่อาจเกิดจากข้อบกพร่อง ในกระบวนการผลิตรวมถึงข้อบกพร่องที่เกิดจากการเหนี่ยวนำของสิ่งแวดล้อม ตัวอย่างเช่น กรณีที่มีผู้นำชิปประมวลผลภาพไปใช้บนดาวเทียม รังสีต่างๆ ในอวกาศอาจเหนี่ยวนำให้วงจรบนชิปเปลี่ยนสถานะโดยไม่ต้องการทำให้เกิดความผิดพลาด เช่น Signal-bit error ได้ ซึ่งถ้าอัตราความผิดพลาดบิต (bit error rate) สูงเกินไป เราอาจปรับแก้งานออกแบบของเราให้ทนทานต่อข้อบกพร่องด้วยเทคนิคต่างๆ ได้

ขั้นตอนสองขั้นสุดท้าย คือการออกแบบในระดับวงจร (Circuit design) และการออกแบบในระดับกายภาพ (Physical design) การออกแบบในระดับวงจรคือการแปลงลอจิกเกตและฟลิปฟลอปต่างให้กลายเป็นทรานซิสเตอร์ ซึ่งในขั้นตอนนี้จะมีการวิเคราะห์วงจรและจำลองเชิงเวลา (Timing simulation) ควบคู่กันไปด้วยเพื่อ ตรวจสอบว่าวงจรและระบบที่ออกแบบไว้ ยังทำงานสอดคล้องกับข้อกำหนดต่างๆ ที่ตั้งไว้ในตอนแรกหรือไม่

การออกแบบในระดับกายภาพ หรือระดับล่างสุด คือการออกแบบผังภูมิ (Lay out) ของวงจรรวมที่ออกแบบไว้ โดยการแปลงทรานซิสเตอร์ต่างๆ ให้เป็นรูปเหลี่ยมและลายเส้นที่มีสีสันแทนวัตถุชั้นต่างๆ ตามกระบวนการผลิตชิปวงจรรวม ภายหลังการออกแบบจะมีการตรวจสอบความถูกต้องตามกฎการออกแบบ การสกัดพารามิเตอร์ต่างๆ และการจำลองในระดับวงจร เช่น Timing simulation อีกครั้งโดยการนำค่าพารามิเตอร์ที่สกัดได้มาประกอบการคำนวณ ในขั้นนี้เราสามารถประเมินคุณสมบัติทางกายภาพที่ถูกต้องได้ เช่น พื้นที่ชิปและกำลังที่ สูญเสียเป็นความร้อน

การแบ่งระดับขั้นการออกแบบนี้เราเรียกแต่ละระดับว่าระดับของการกำหนดสาระสำคัญ (Level of abstraction) การกำหนดสาระสำคัญระดับสูงจะประกอบไปด้วยส่วนประกอบที่ซับซ้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และทรงพลังเป็นจำนวนน้อยเช่น วงจรบวก (Adder) และหน่วยความจำ (Memory) ตรงข้ามกับการกำหนดสาระสำคัญในระดับล่าง ที่จะประกอบไปด้วยส่วนประกอบจำนวนมากที่ไม่ซับซ้อนและไม่ทรงพลัง เช่น ประตูสัญญาณ (Gate) และทรานซิสเตอร์โดยทั่วไปกระบวนการออกแบบจะเป็นไปลักษณะจากบนลงล่างหากมีการค้นพบความผิดพลาดที่ระดับล่าง การแก้ไขในระดับนี้จะเป็นไปได้ยากและแพงกว่ากรณีที่ตรวจพบข้อผิดพลาดแค่เนิ่นๆ ในระดับสูง เพราะมีรายละเอียดที่ต้องแก้ไขมาก และยังทำให้เสียเวลาอีกด้วย

2.5 ส่วนประกอบต่างๆ ของภาษา VHDL

ในการเขียนรูปแบบบรรยายระบบดิจิตอลในมุมมอง ของการออกแบบลักษณะบนลงล่าง จะต้องทำความเข้าใจในเรื่องของโครงสร้างและส่วนประกอบต่างๆ ของรูปแบบภาษา VHDL เสียก่อน ซึ่งส่วนประกอบที่สำคัญและเป็นพื้นฐานของการเขียนมี 4 หน่วยคือ

2.5.1 หน่วยการออกแบบเอนทิตี (Entity Design Unit)

หน่วยการออกแบบนี้เป็นส่วนที่ใช้สำหรับติดต่อระหว่างโลกภายนอกกับรูปแบบที่เขียนขึ้น ที่เรียกว่าหน่วยการออกแบบเอนทิตี ในส่วนนี้ใช้กำหนดจุดเชื่อมต่อ ของรูปแบบ กำหนดทิศทางการไหลของสัญญาณ และประเภทของค่าที่สามารถกำหนดให้กับสัญญาณตามจุดต่างๆ ของข้อมูลที่ไหลผ่านจุดต่อเหล่านั้น มี 4 ประเภทดังนี้

- In (Input) เป็นพอร์ททิศทางเดียว ที่นำค่าสัญญาณจากอุปกรณ์ภายนอกเข้ามาใช้ภายในวงจร สามารถนำมาป้อนให้กับสัญญาณอื่นหรืออ่านค่าอื่นได้แต่ไม่สามารถถูกเขียนจากภายในวงจรได้

- Out (Output) เป็นพอร์ททิศทางเดียว ที่นำค่าสัญญาณจากวงจร ส่งออกไปยังอุปกรณ์ภายนอกสามารถเขียนจากภายในวงจรได้ แต่ไม่สามารถอ่านจากภายในวงจรได้

- Inout (Bidirectional) เป็นพอร์ท 2 ทิศทาง ที่สามารถส่งถูกเขียนและอ่านได้จากภายในวงจร

- Buffer (Output with internal Feedback) เป็นพอร์ทเอาท์พุตประเภทหนึ่งที่สามารถอ่านค่ากลับ (Feedback) เข้ามาภายในวงจรได้

2.5.2 หน่วยการออกแบบสถาปัตยกรรม (Architecture Design Unit)

เป็นหน่วยการออกแบบส่วนที่ใช้เขียนบรรยายพฤติกรรมการทำงานของวงจรที่ต้องการออกแบบ โดยมีความสัมพันธ์กับสิ่งที่กำหนดใน Entity ถ้าผู้ออกแบบต้องการออกแบบวงจรดิจิตอลใดๆ จะต้องออกแบบหน่วยการออกแบบวงจรพื้นฐานก็คือ ส่วน Entity และส่วนของ Architecture ที่มีความสัมพันธ์กันซึ่งมีด้วยกันหลายรูปแบบสำหรับการออกแบบวงจรเดียวกัน และความเข้าใจของผู้ออกแบบเอง ขึ้นอยู่กับผู้ออกแบบที่จะนำรูปใดมาใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 หน่วยการออกแบบแพ็กเก็ต (Package Design Unit)

Package จะเป็นหน่วยการออกแบบที่ใช้เก็บข้อมูลต่างๆ ตลอดจนโปรแกรมย่อย (Subprogram) ที่เป็นประโยชน์ในการเขียนรูปแบบการบรรยายวงจรดิจิตอล โดยข้อมูลใน Package สามารถถูกเรียกใช้ได้โดย Entity, Architecture หรือด้วย Package อื่นๆ ด้วยคำสั่ง USE statement นอกจากนั้นสิ่งที่นิยมทำกันมากคือรูปแบบมาตรฐานต่างๆ อาทิ เช่น Standard component ต่างๆ จะถูกเก็บไว้ใน Package ที่ทุกคนเรียกใช้งานได้ โดยสิ่งที่สามารถสร้างไว้ใช้ใน Package ได้แก่

- Subprogram
- Type
- Constant
- Signal
- Aliases
- Attributes
- Component
- Disconnection Specification

การเรียกใช้งาน Package จะใช้คำสั่ง Use statement โดยปกติการเขียน Package จะแบ่งออกเป็น 2 ส่วนคือ

- Package declaration
- Package body

Package declaration

เป็นส่วนที่สำคัญที่สุดของ Package เพราะจะเป็นส่วนที่กำหนดชื่อ (Identifier) ของสิ่งที่ประกาศอยู่ภายใน Package สำหรับนำไปใช้ภายนอกตัวของ Package ถ้าสิ่งใดที่ถูกประกาศอยู่ในส่วนของ Package body แต่ไม่ถูกประกาศในส่วนของ Package declaration ก็จะไม่สามารถนำไปใช้งานภายนอกได้ ซึ่งเปรียบเทียบกับการประกาศพอร์ทใน Entity ที่มีหน้าที่ติดต่ออุปกรณ์ภายนอก ดังนั้นโดยทั่วไปแล้ว Package สามารถเขียนได้โดยไม่จำเป็นต้องมีส่วนของ Package body และยังสามารถถูกนำไปใช้ได้จากภายนอก เช่น ใช้สำหรับประกาศชนิดของข้อมูล (Type) หรือสัญญาณ (Signal) ในทางกลับกันกับPackage body ที่ไม่จำเป็นต้องมี Package declaration แต่ Package นั้นจะไม่สามารถถูกนำไปใช้จากภายนอกได้

Package body

โครงสร้างที่ประกอบด้วยด้วยคำสั่งต่างๆ ในรูปแบบของคำสั่งลำดับ (Sequential statement) ที่ใช้บรรยายฟังก์ชันการทำงานของโปรแกรมย่อย (Subprogram) ที่ได้ประกาศไว้ใน Package declaration แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 หน่วยการออกแบบโครงแบบ (Configuration Design Unit)

ดังที่กล่าวมาแล้วว่ารูปแบบการเขียนส่วน Entity สามารถมีหลายๆ Architecture และภายใน Architecture ยังมีอุปกรณ์ที่เรียกว่า Component ที่อ้างถึง Entity อื่นๆ อีก ดังนั้นในการจำลองการทำงาน (Simulate) จึงเกิดปัญหาว่าจะนำเอา Architecture อันไหนไปจำลองการทำงาน ฉะนั้นเราจึงต้องเขียนส่วนของ Configuration ในการกำหนดว่าจะนำ Architecture อันไหนไปใช้งาน Configuration ส่วนใหญ่ไม่สามารถนำไปใช้กับ Synthesis tools ได้จะใช้ได้เฉพาะขั้นตอนการ Simulationเท่านั้น เพราะฉะนั้นในขั้นตอนการ Synthesis Entity หนึ่งๆ ต้องมีเพียง หนึ่ง Architecture เท่านั้น แต่ในบางครั้งถ้าเราไม่ต้องการเขียน Configuration ก็ได้ถ้าเราออกแบบให้หนึ่ง Entity มีเพียงหนึ่ง Architecture เท่านั้น ในกรณีที่ไม่จำเป็นต้องเขียน Configuration บอก Simulator เนื่องจากไม่มีทางเลือกอื่นใด ให้ซอฟต์แวร์ Simulator เลือกวิธีนี้เราเรียกว่า Default configuration หรือถ้าบางครั้งเราออกแบบให้หนึ่ง Entity มีหลาย Architecture แต่เราก็สามารถไม่ต้องเขียน Configuration ได้เช่นกัน แต่ก็จะเกิดคำถามตามมาว่าเมื่อจำลองการทำงาน Simulator จะเลือกเอา Architecture อันไหนมาจำลองการทำงาน คำตอบ ในกรณีไม่เขียน Configuration นั่นคือ Last compile ล่าสุดที่มีการ Compile ใน Architecture อันไหน Simulator ก็จะเอา Architecture อันนั้นไปจำลองการทำงานดังนั้น ควรระวังในการเขียนโค้ดรูปแบบดังกล่าวด้วย

2.6 การเขียนแบบลำดับชั้น (Hierarchical Model)

ในการออกแบบวงจรดิจิตอล บางครั้งต้องมีการแบ่งวงจรออกเป็นบล็อกย่อยๆตามหน้าที่การทำงานเนื่องจากระบบที่ต้องการออกแบบมีความซับซ้อนสูงไม่สามารถที่จะออกแบบได้เพียงบล็อกเดียว ทำให้จำเป็นต้องแบ่งวงจรออกเป็นบล็อกย่อยๆ ซึ่งในภาษา VHDL ก็มีความสามารถในการเขียนลำดับขั้น (Hierarchical) ได้ โดยเขียนบรรยายวงจรตั้งแต่ระดับบนสุด (Top model) ที่มีแต่ความสัมพันธ์ของการเชื่อมต่อในแต่ละบล็อกย่อย โดยยังไม่มีรายละเอียดของวงจรในแต่บล็อก และในระดับล่างลงไปถึงจะมีรายละเอียดของวงจร

2.6.1 การประกาศคอมโปเนนต์ (Component Declaration)

เป็นการประกาศ Component ที่ต้องนำมาเชื่อมต่อกันภายในวงจรส่วนบนสุด (Top model) จะเป็นการเขียนในลักษณะลำดับขั้น (Hierarchical) โดยการประกาศ Component นั้นจะต้องทำการออกแบบ Component ดังกล่าวไว้ก่อนเรียบร้อยแล้ว เพียงแต่เราเรียก Component ดังกล่าวมาใช้งานโดยการออกแบบ Component ก็คล้ายกับการออกแบบวงจรทั่วไป คือสร้างส่วนของ Entity และ Architecture ขึ้นมาก่อน แล้วเก็บไว้ในไลบารีจากนั้นถ้าวงจรที่ออกแบบขึ้นมาใหม่ต้องการเรียกใช้วงจรที่เคยออกแบบไว้แล้ว สามารถทำได้โดยการประกาศส่วน ของโมเดลที่มีอยู่แล้วในรูปของ Component ซึ่งตำแหน่งของการประกาศ Component สามารถประกาศในบริเวณระหว่างคำสั่ง Architecture และ Begin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประกาศส่วนของ Component นั้นถ้านำไปเปรียบเทียบกับการออกแบบวงจรบน PCB (Printed Circuit Board) จะเปรียบเสมือนว่าเป็นซ็อกเก็ตว่างๆไว้สำหรับเสียบไอซีวงไปโดยซ็อกเก็ตมีหน้าที่เพียงเชื่อมต่อสัญญาณเท่านั้นไม่มีหน้าที่สร้างฟังก์ชันการทำงาน การประกาศส่วนของอุปกรณ์ก็เช่นกัน จะไม่มีการแสดงถึงฟังก์ชันการทำงาน แต่มีหน้าที่เพียงเชื่อมต่อสัญญาณต่างๆ เข้ากับพอร์ทของโมดูลย่อยนั้น

2.6.2 การเรียกใช้งาน Component

เป็นการเรียกใช้ Component ที่มีอยู่แล้ว โดยการเชื่อมต่อสัญญาณต่างๆ ของโมดูลย่อยที่ประกาศไว้ในส่วนของ Component declaration เข้ากับพอร์ทหรือสัญญาณภายในของวงจรระดับบนสุด (Top model) ตำแหน่งการเขียน Component instantiation เป็นรูปแบบหนึ่งของ Concurrent statement ดังนั้นสามารถเขียนภายได้ Beginของ Architecture ได้

คำสั่ง Generic Map เป็นคำสั่งเสริม (Optional) ถ้าวงจรมีการใช้คำสั่ง Generic มาก่อนหน้านี้แล้วถ้าเรานำเอาบล็อกของวงจรดังกล่าวมาใช้งาน เวลาเขียน Component instantiation จะต้องมีคำสั่ง Generic Map เสมอ แต่ถ้าบล็อกของวงจรที่เราใช้งานไม่มีการใช้คำสั่ง Generic เวลาเขียน Component instantiation ก็ไม่ต้องมีส่วน Generic Map

ในการเชื่อมต่อสัญญาณต่างๆ เข้ากับพอร์ทของ Component สามารถทำได้ 2 รูปแบบ คือ เชื่อมต่อสัญญาณแบบพอร์ท association list คือ พิจารณาจากตำแหน่งของพอร์ท และตำแหน่งในสัญญาณ ที่ตรงกันจะมีความหมายว่าต่อถึงกันอยู่ ไม่มีการอ้างชื่อพอร์ทเพียงแต่วางตำแหน่งให้ตรงกันก็พอ และอีกรูปแบบหนึ่งการเขียนคำสั่งเชื่อมพอร์ท ในรูปแบบ Name association list จะหมายถึงจะไม่พิจารณาตำแหน่ง แต่ต้องอ้างอิงด้วยชื่อในการเชื่อมต่อ

2.7 ความสามารถของภาษา VHDL

- เป็นภาษามาตรฐานสากลโดยรองรับจากสถาบัน IEEE ทำให้มีโปรแกรมและเครื่องมือต่างๆ และบริษัทที่สนับสนุนการทำงานมากมาย นอกจากนี้วงจรที่ออกแบบโดยภาษา VHDL ก็จะสามารถใช้งานได้ยาวนานเนื่องจากภาษามีความเข้ากันได้กับวงจรที่ได้ออกแบบมาใหม่
- ภาษา VHDL สนับสนุนการออกแบบ Down Design และแบบ Bottom Up Design หรือเป็นแบบผสมทั้งสองแบบ
- เป็นภาษาที่สามารถใช้งานได้หลายระบบ การออกแบบโดยใช้ภาษา VHDL สามารถนำไปจำลองการทำงานหรือสังเคราะห์ด้วยซอฟต์แวร์ ตัวใดก็ได้ที่รองรับ VHDL จึงทำให้การออกแบบด้วยภาษา VHDL เป็นการออกแบบที่ไม่ยึดติดกับซอฟต์แวร์ที่ใช้ในการออกแบบ

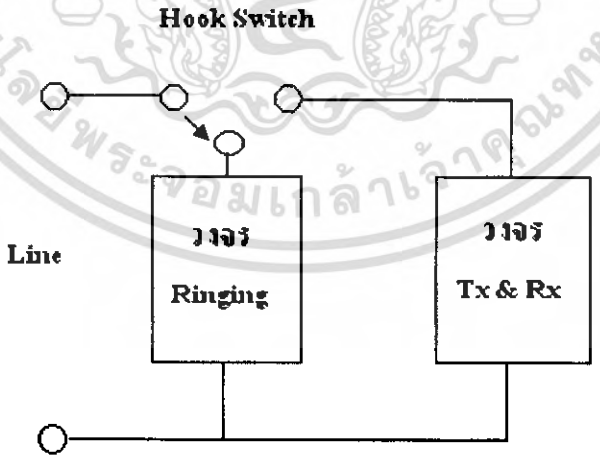
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เป็นภาษาที่สามารถจำลองรูปแบบการทำงานของวงจร ผู้ออกแบบวงจรสามารถออกแบบวงจรโดยใช้ VHDL ได้หลายระดับ ตั้งแต่ ระดับ Macro Block จนถึงระดับ Gate และสามารถออกแบบวงจรที่มีความซับซ้อนสูงและมีขนาดใหญ่มากได้
- เป็นภาษาที่สามารถนำกลับมาใช้ใหม่ได้ เนื่องจากวงจรที่ออกแบบโดยภาษา VHDL สามารถทำการเปลี่ยนแปลงวงจรได้โดยง่าย
- เป็นภาษาที่สามารถนำไปเป็นเอกสารประกอบได้ เพราะเนื่องจากเป็นภาษาในรูปแบบบรรยายพฤติกรรม ทำให้สามารถอธิบายการทำงานของวงจรภายในการออกแบบได้ทันที

2.8 ทฤษฎีของโทรศัพท์

2.8.1 การทำงานของชุมสายโทรศัพท์

การทำงานของเครื่องโทรศัพท์และชุมสายโทรศัพท์โดยมีจุดประสงค์ เพื่อการติดต่อเครื่องโทรศัพท์ที่ยกหูโทรศัพท์ จะเรียกว่า "ผู้เรียกหรือฝ่ายเรียก" หรือ A-SUB ไปยังเครื่องรับอีกเครื่องหนึ่ง เรียกว่า "ผู้ถูกเรียกหรือ ฝ่ายถูกเรียก" หรือ B-SUB เมื่อทั้งสองสามารถสนทนากันได้ เรียกว่า "ต่อครบวงจรสนทนา" ผู้เรียกอาจอยู่ในชุมสายเดียวกันหรือ ต่างชุมสายก็ได้ ซึ่งในกรณีต่างชุมสายจะมีการเรียกไปยังชุมสายนั้น ๆ และส่งข้อมูลตามความต้องการไปให้ โดยชุมสายของผู้ถูกเรียกจะส่งสัญญาณกระดิ่งไปยังผู้ถูกเรียกและส่งสัญญาณเรียกกลับไปยังเครื่อง ผู้เรียกที่ชุมสายที่ถูกผู้เรียกสังกัดอยู่นั้นเราเรียกว่า "ชุมสายปลายทาง" และชุมสายของผู้เรียกคือ "ชุมสายต้นทาง"



รูปที่ 2.7 บล็อกไดอะแกรมของวงจรโทรศัพท์

ระบบโทรศัพท์ที่ใช้กันอยู่ในปัจจุบันจะมีสายสัญญาณ 2 เส้น เป็นตัวกลางในการสื่อสาร โดยจะต่อจากเครื่องโทรศัพท์ไปยังชุมสายปลายทาง และมีค่าอิมพีแดนซ์ของสายประมาณ 500 –

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1000 Ωโดยที่ภายในชุมสายโทรศัพท์จะมีการติดตั้งแหล่งจ่ายไฟตรงรวมขนาด 48 V ในแต่ละลูปของผู้ใช้โทรศัพท์ สายสัญญาณทั้ง 2 เส้น เรียกว่า ทิป (Tip) และ ริง (Ring) โดยสายริงจะต่อกับสัญญาณไฟ 48 V เพื่อกันสนิมส่วนสายทิปต่ออยู่กับกราวด์ที่ชุมสายโทรศัพท์

เมื่อผู้ใช้ยกหูโทรศัพท์ มีผลทำให้ฮุกสวิตซ์ (Hook Switch) ภายในเครื่องโทรศัพท์ปิดลง จากนั้นจะเกิดกระแสไฟตรง 20 mA ไหลวนอยู่ในลูปซึ่งอยู่ในสภาวะยกหู ระดับแรงดันระหว่างสายทิปกับสายริงจะลดลงเหลือประมาณ 8-10 V สัญญาณเสียงพูดจากเครื่องโทรศัพท์จะถูกส่งในทิศทางใดทิศทางหนึ่งของลูป ซึ่งเกิดจากสัญญาณไฟสลับคือสัญญาณเสียงพูดทับซ้อนบนกระแสไฟตรงในลูป สัญญาณเสียงพูดจะถูกคับปลิ้งอยู่ระหว่างวงจรลูปและขดลวดรีเลย์จะคอยรักษาแรงดันกระแสลูป รวมทั้งแจ้งสภาวะของผู้ใช้ว่ายกหูหรือวางหูให้ชุมสายรับทราบด้วย เมื่อชุมสายตรวจพบสภาวะยกหูโทรศัพท์จะทำการส่งสัญญาณให้กดปุ่มหมายเลขโทรศัพท์เข้าไปในลูป

2.8.2 สัญญาณต่างๆ ในระบบชุมสายโทรศัพท์

- สัญญาณการวางหู (On hook) หมายถึง สภาวะที่ผู้เช่าวางหู ลักษณะของวงจรจะเป็นอินพีแดนซ์สูงแบบวงรอบเปิด
- สัญญาณการยกหู (Off hook) หมายถึง สภาวะที่ผู้เช่ายกหู ลักษณะของวงจรจะเป็นอิมพีแดนซ์ต่ำแบบวงรอบปิด
- สัญญาณพร้อมหมุน (Dial Tone) เป็นสัญญาณที่บอกให้ทราบว่าขณะนี้ อุปกรณ์ที่ชุมสายว่าง หรือพร้อมที่จะรอรับรหัสการกดปุ่ม รหัสการกดปุ่มจะมีสัญญาณเสียงสองความถี่ (Dual tone multi frequency : DTMF) ให้ผู้เรียกทำการส่งหมายเลขได้โดย ลักษณะการส่งของสัญญาณจะเป็นรูปไซน์(Sine Wave)ความถี่400 ถึง450 Hz มอดูเลตกับความถี่ 50 Hz โดยผู้ใช้จะได้ยินเสียงเมื่อทำการยกหูโทรศัพท์ เพื่อทำการเรียก
- สัญญาณเรียกกลับ (Ringback Tone) เป็นสัญญาณเพื่อแสดงการติดต่อเมื่อชุมสายสามารถดำเนินการติดต่อได้สำเร็จ และต้องการแจ้งให้ผู้เรียกทราบ ซึ่งเป็นสัญญาณเรียกกลับมีลักษณะเป็นสัญญาณไซน์ ที่มีความถี่ประมาณ 400 ถึง 450 Hz ส่งออกมาเป็นช่วงๆ โดยมีจังหวะ ดัง 4 วินาที เงียบ 1 วินาที สลับกันไป
- สัญญาณสายไม่ว่าง (Busy Tone) เป็นสัญญาณที่ส่งออกมาเพื่อแสดงให้ได้ทราบว่าปลายทาง คือผู้รับสายกำลังใช้โทรศัพท์อยู่ เช่น ถ้าผู้ใช้ยกหูแล้วได้ยินเสียงสัญญาณไม่ว่างแทนที่จะได้ยินเสียงสัญญาณหมุนแสดงว่า อุปกรณ์ในชุมสายไม่ว่าง แต่ถ้าได้ยินเสียงสัญญาณไม่ว่าง หลังจากหมุนหมายเลขไปแล้ว แสดงว่าคู่สายฝ่ายผู้ถูกเรียกไม่ว่าง ในกรณีที่ผู้ถูกเรียกอยู่ต่างชุมสาย ลักษณะของสัญญาณที่ส่งออกมาจะเป็นลักษณะเป็นสัญญาณไซน์ โดยมีจังหวะ ดัง 0.5 วินาที เงียบ 0.5 วินาที สลับกันไปซึ่งมีความถี่ประมาณ 400 ถึง 450 Hz

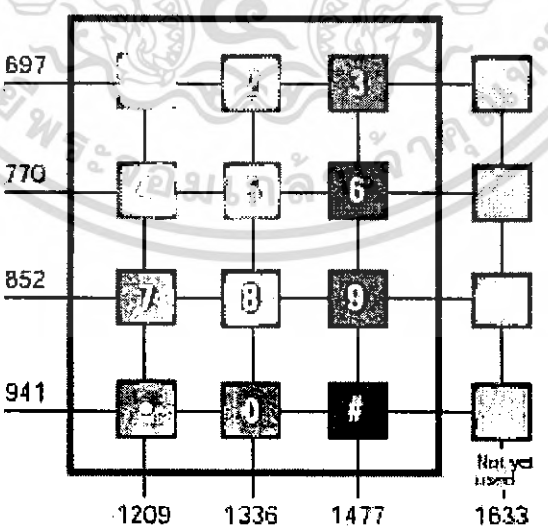
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณกริ่ง (Ringing Tone) หรือสัญญาณกระดิ่งเป็นสัญญาณเรียกเพื่อแสดงว่าการต่อสายของผู้เรียกไปยังผู้รับสาย เครื่องชุมสายสามารถดำเนินการติดต่อได้เสร็จ และส่งสัญญาณกริ่งไปให้ผู้รับมาทำการรับสาย โดยสัญญาณเรียกจะมีลักษณะเป็นสัญญาณรูปไซน์ ที่มีความถี่ประมาณ 400 ถึง 450 Hz ส่งมาเป็นช่วงๆ โดยมีจังหวะ ดัง 4 วินาที เงียบ 1 วินาที มีระดับแรงดันประมาณ 90 V
- สัญญาณนัมเบอร์ อันออปเทนอะเบิล โทน (Number Unobtainable Tone : Nutone) ที่บอกว่าหมายเลขนั้นยังไม่มีการเปิดใช้บริการ สัญญาณนี้มีลักษณะเป็นสัญญาณรูปไซน์ที่มีความถี่ ประมาณ 400 ถึง 450 Hz ดัง 0.1 วินาที เงียบ 0.1 วินาที

สัญญาณที่ถูกส่งไปตามคู่สายโทรศัพท์ เป็นการส่งสัญญาณความถี่ไปพร้อมกับไฟฟ้ากระแสตรง โดยความแตกต่างของสัญญาณที่ใช้นั้นจะแตกต่างกันที่ระดับของแรงดันและความถี่ที่ทางชุมสายส่งไปยังเครื่องโทรศัพท์ เมื่อมีการติดต่อเกิดขึ้น

2.8.3 ระบบดีทีเอ็มเอฟ

ระบบดีทีเอ็มเอฟ สามารถสร้างความถี่มาตราฐานใหม่ในย่านความถี่เสียงแตกต่างกัน 8 ความถี่ โดยแบ่งออกเป็น 2 กลุ่ม คือกลุ่มความถี่ต่ำ 4 ความถี่ ประกอบด้วยความถี่ 697, 770, 852, และ 941 Hz และกลุ่มความถี่สูง 4 ความถี่ ประกอบด้วยความถี่ 1209 , 1336, 1477 และ 1633 Hz สัญญาณดีทีเอ็มเอฟ ได้มาจากการรวมสัญญาณความถี่ต่ำจากกลุ่มความถี่ต่ำ 1 ความถี่ และสัญญาณความถี่สูง จากกลุ่มความถี่สูง 1 ความถี่ และได้ทำการกำหนดปุ่มด้วยตัวเลข 0-9, * (Star), # (Square), A, B, C และ D ซึ่งในการกดปุ่ม ๆ หนึ่งจะให้สัญญาณ ความถี่คู่หนึ่งออกมา



รูปที่ 2.8 แป้นกดของโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
72004

2.8.4 การเชื่อมต่อในระบบโทรศัพท์

เครื่องโทรศัพท์จะติดต่อกับชุมสายโทรศัพท์ด้วยสัญญาณสองเส้น คือ ทิป (Tip) และริง (Ring) ปกติเมื่อไม่มีการใช้โทรศัพท์วงจรของเครื่องโทรศัพท์จะถูกตัดออกจากคู่สายของโทรศัพท์ คงเหลือแต่วงจรกำเนิดเสียงเรียก (Ringing) หรือ วงจรกระดิ่งที่ติดต่ออยู่กับชุมสายโทรศัพท์เท่านั้น เพื่อส่งสัญญาณเรียกเมื่อมีการติดต่อมาจากผู้อื่นเข้ามา ทำให้ในขณะที่โทรศัพท์ไม่ได้ถูกใช้งานจะไม่มีกระแสไหลผ่านเครื่องรับโทรศัพท์แต่เมื่อมีการยกหูโทรศัพท์ก็จะมีการเชื่อมต่อเครื่องโทรศัพท์เข้ากับชุมสายขององค์กรโทรศัพท์ทำให้ระดับแรงดันไฟตรง 48 V จะถูกต่อเข้ากับวงจรโทรศัพท์ โดยใช้ฮุกสวิตซ์ (Hook Switch) เกิดการไหลของกระแสในวงจรโดยกระแสนี้เกิดจากแบตเตอรี่ในชุมสายโทรศัพท์ที่อยู่ใกล้ที่สุด และเมื่อชุมสายโทรศัพท์ได้ทำการเลือกคู่สายที่ต้องการจะติดต่อด้วยแล้วก็จะทำการส่งสัญญาณกระดิ่ง ซึ่งเป็นแรงดันกระแสสลับที่มีระดับแรงดันประมาณ 100 V_{pp} เป็นเวลา 1 วินาที และหยุดเป็นเวลา 4 วินาที ออกไปยังเครื่องรับโทรศัพท์ของผู้ถูกเรียก เพื่อทำการสั่นกระดิ่งให้ดังขึ้นเมื่อมีผู้ถูกเรียกยกหูโทรศัพท์รับสายก็จะเกิดกระแสตรงไหลเมื่อชุมสายโทรศัพท์ตรวจพบก็จะหยุดส่งสัญญาณกระดิ่ง ก็จะทำการสนทนาได้ โดยในส่วนที่เชื่อมต่อระหว่างปากพูดหูฟังกับโทรศัพท์จะต้องมีหม้อแปลงทำหน้าที่ในการปรับอิมพีแดนซ์ของหูฟัง และสายโทรศัพท์ให้สมดุลกันเพื่อให้มีการรับและส่งสัญญาณอย่างมีประสิทธิภาพมากที่สุดรวมทั้งผู้พูดสามารถได้ยินเสียงของตัวเองด้วยเพื่อที่จะได้ปรับระดับการพูดของตนเองไม่ให้ดังเกินไปหรือค่อยจนเกินไป

2.9 ทฤษฎีของไอซีบันทึกเสียงและเล่นกลับ (Single – Chip Voice Record/Playback Device)

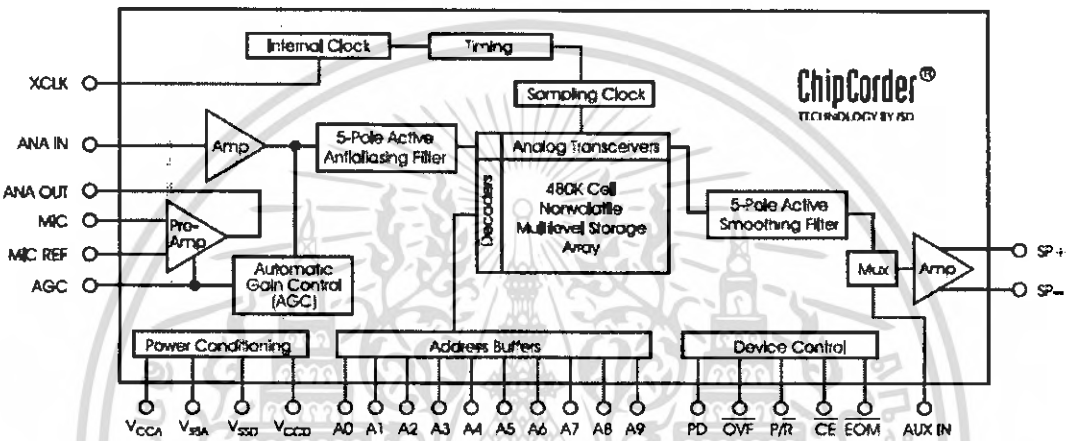
หลักการทำงาน คือ เมื่อมีการบันทึกเสียงเข้ามาจะมีการจัดเก็บข้อมูลต่างๆ ไว้ในหน่วยความจำที่เป็นเซลล์เหล่านี้ ไม่ต้องการแรงดันสำรอง เพื่อเก็บรักษาข้อมูลไม่ให้สูญหาย (Nonvolatile Memory Cell) สัญญาณเสียงที่อยู่ในรูปแบบอนาล็อกจะถูกจัดเก็บในหน่วยความจำโดยตรงโดยอาศัยเทคโนโลยี DAST (Discrete Analog Storage Technology) จะทำการจัดเก็บความจำโดยตรงโดยอาศัยเทคโนโลยี DAST (Discrete Analog Storage Technology) จะทำการจัดเก็บความจำในลักษณะที่ยังคงสัญญาณอนาล็อกไว้เหมือนเดิม จึงทำให้การเปลี่ยนสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล (Analog to Digital) เข้ามาเกี่ยวข้องนอกจากนี้ ไอซีแต่ละตัวที่มีระยะเวลาในการบันทึกเสียงที่แตกต่างกันก็จะมีคุณสมบัติทางไฟฟ้าที่แตกต่างกันด้วย

คุณสมบัติของไอซี

- ไม่มีอุปกรณ์ประเภทไอซีอื่นๆ มาต่อร่วม
- ไม่ต้องพัฒนาระบบอื่นขึ้นมาเสริมเพื่อให้ใช้งานได้
- มีประสิทธิภาพในการบันทึก และเล่นกลับที่ให้เสียงได้เหมือนต้นกำเนิดเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ควบคุมการบันทึก และเล่นกลับด้วยสวิตซ์หรือ ควบคุมด้วยไมโครคอนโทรลเลอร์
- ต่อคาสเคดกันได้โดยตรงเพื่อเพิ่มระยะเวลาให้ยาวนานมากขึ้น
- ปิดการทำงานอัตโนมัติเมื่อไม่มีการบันทึกหรือเล่นกลับนานเกินไป
- สามารถเก็บความจำไว้ได้นาน 100 ปี ไม่ต้องมีแบตเตอรี่สำรอง
- วงรอบการบันทึก 100,000 ครั้ง
- มีวงจรกำเนิดสัญญาณนาฬิกาภายในตัว
- สามารถควบคุมการเล่นกลับเพียงอย่างเดียวเพื่อพัฒนารูปแบบใช้งานได้



รูปที่ 2.9 วงจรภายในไอซีบันทึกเสียง

2.9.1 หลักการทำงานของขาไอซีบันทึกเสียงเบอร์ ISD2560/75/90/120

การทำงานของแต่ละขาของไอซี Address / Mode Input (A0 – A9 / M0 –M6) ขาที่ 1 – 10 คือขาแอดเดรส และโหมดอินพุตจะมีอยู่สองฟังก์ชันที่ขึ้นอยู่กับระดับของสอง MSB ของแอดเดรส ถ้าแอดเดรสใดแอดเดรสหนึ่งเป็น “0” อินพุตก็จะมาปรากฏที่แอดเดรสบิตทั้งหมด และใช้แอดเดรสเริ่มต้นสำหรับวงรอบการบันทึกและเล่นกลับขาแอดเดรสโหมดอินพุตจะมาขึ้นอยู่ที่โหมดบิตทั้งหมด และเกิดการแลตช์เมื่อพัลส์ขอบขาลงปรากฏที่ขา CE

Auxiliary Input (AUX IN) ขา 11 จะเป็นขาที่รับอินพุตจากภายนอกซึ่งเป็นการ มัลติเพลกซ์สัญญาณผ่านออกไปทางเอาต์พุตของวงจรขยายภายใน และขับออกสู่ขาเอาต์พุตของลำโพง โดยขั้นตอนการทำงานนี้จะเกิดขึ้นเมื่อขา CE มีสถานะเป็น “1” วงรอบของการเล่นกลับจะสิ้นสุดลง

Ground Input (V_{SSA} , V_{SSD}) ขา 12 และ 13 โดยคุณสมบัติของไอซีในตระกูล ISD2590 จะแยกกันระหว่างกราวด์ทั้งสองนี่จะถูกต่อ และปิดไว้ภายในตัวถังบรรจุของไอซี การใช้งานขากราวด์ทั้งสองจะเลือกต่อกับกราวด์ของแหล่งจ่ายไฟในส่วนที่มีค่าอิมพีแดนซ์ ต่ำ เพื่อไม่ต้องการให้เกิดค่าแรงดันที่แตกต่างกันระหว่างกราวด์ทั้งสอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Speak Output (SP+ , SP-) ขา 14 และ 15 เป็นขาเอาต์พุตต่อออกลำโพงในตระกูล ISD2590 นี้ จะมีวงจรขับสัญญาณความแตกต่างออกสู่ลำโพงซึ่งประกอบอยู่ภายในตัวไอซี เรียบร้อยแล้ว โดยมีความสามารถในการขับลำโพงเอาต์พุตได้ 50 mW ที่โหลด ลำโพง 16 Ω ขาต่อลำโพงเอาต์พุตทั้งสองขานี้จะไม่ต่อขนานกันโดยตรงเด็ดขาด เมื่อต้องถูกใช้ต่อคาสเคดกันหลายๆ ตัว และไม่เหมาะในการต่อลำโพงขนานกันทางเอาต์พุต หลายตัว โดยเฉพาะในบางครั้งขาเอาต์พุต ลำโพงสามารถต่อคาสเคดกับไอซีอีกตัวได้โดยตรง เพราะตัวเก็บประจุคับปลิ้งอยู่ภายในเรียบร้อยแล้ว

Voltage Input (V_{CC} , V_{DD}) ขา 16 และ 28 เป็นขารับแรงดันที่จะต้องแยกกันต่างหากระหว่างขารับแรงดันของวงจรอนาล็อกและวงจรดิจิตอลที่ประกอบอยู่ภายในตัวไอซีแล้ว แรงดันที่ต้องการคือ แรงดันไฟเลี้ยง +5V และต้องเป็นแรงดันไฟเลี้ยงที่มีสัญญาณรบกวนต่ำมาก

Microphone Input (MIC) ขา 17 จะรับสัญญาณอินพุตที่ผ่านเข้ามายังไมโครโฟนแล้วส่งสัญญาณเข้าสู่วงจรปรีแอมป์ที่ประกอบอยู่ในตัวไอซี ภายในประกอบด้วยวงจรควบคุมอัตราการขยายอัตโนมัติ (AGC) โดยวงจรนี้จะทำหน้าที่ควบคุมอัตราการขยายของวงจรปรีแอมป์ให้อัตราการขยายอยู่ในช่วง -15 ถึง 24 dB ไมโครโฟนภายนอกจะถูกผ่านตัวเก็บประจุภายในลักษณะอนุกรมกับขา 17 นี้ ค่าความจุของตัวเก็บประจุคัปปลิ้งจะกำหนดค่าโดยคำนึงถึงค่าความต้านทาน 10 kΩ ที่ต่ออยู่ภายในกับขา 17 ของไอซีเพื่อทำให้เกิดการคัตออฟที่ความถี่ต่ำ

Microphone Reference (MIC REF) ขา 18 จะกราวด์อนาล็อก (V_{SSA}) โดยมีตัวเก็บประจุต่ออนุกรมอยู่ก่อน เพื่อทำหน้าที่กำจัดสัญญาณรบกวนทางอินพุตที่ขา 17 เพื่อให้การชดเชยทางด้านสัญญาณรบกวนให้ดีกว่า 10 dB

Automatic Gain Control (AGC) ขา 19 เป็นขาอินพุตเพื่อควบคุมการปรับอัตราการขยายของปรีแอมป์ไมโครโฟนทางไดนามิก เพื่อให้เหมาะสมกับระดับสัญญาณที่มีย่านความกว้างมากทางอินพุตจากไมโครโฟน และเพื่อให้ระดับสัญญาณที่ได้ทำการบันทึกมีความผิดเพี้ยนน้อยที่สุด ขา ACG นี้จะต้องต่อร่วมกับอุปกรณ์ RC เพื่อกำหนดค่าเวลาคงที่โดยมีค่าความต้านทานภายใน 5 kΩ และจะต้องต่อกับตัวเก็บประจุข้างนอกอีกหนึ่งตัว ผ่านลงกราวด์อนาล็อกค่าที่เหมาะสมบางครั้งกำหนดไว้ที่ค่าความต้านทาน 470 kΩ และตัวเก็บประจุ 4.7 uF

Analog Input (ANA IN) ขา 20 จะรับสัญญาณที่ผ่านวงจรปรีแอมป์ออกมาทางขา 21 โดยผ่านตัวเก็บประจุคับปลิ้งจกาภายนอก ซึ่งจะทำการคัปปลิ้งสัญญาณเข้าที่ขา 20 เพื่อผ่านสัญญาณที่ได้ทำการบันทึกเข้าไปไว้ในตัวไอซี ตัวเก็บประจุคับปลิ้งภายนอกนี้จะต้องสัมพันธ์กันกับค่าความต้านทานภายใน 3 KΩ ซึ่งเป็นอินพุตอิมพีแดนซ์ เพื่อที่จะทำให้เป็นวงจรกรองความถี่ต่ำ

Analog Output (ANA OUT) ขา 21 เป็นขาเอาต์พุตของวงจรปรีแอมป์ขยายสัญญาณจากไมโครโฟนที่ได้รับจากการควบคุมอัตราขยายจากวงจร AGC ภายในแล้ว

Overflow Output (OVF) ขา 22 สัญญาณพัลส์ "0" จะปรากฏออกมาทางเอาต์พุตนี้เพื่อแสดงว่าสิ้นสุดการเล่นกลับแล้ว หรือเพื่อแสดงว่าความจำภายในตัวไอซีได้ถูกอ่านออกมาหมดแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจะแสดงสภาวะหยุดการเล่นกลับพัลส์เอาต์พุตจากขา OVF นี้จะจ่ายให้กับขา CE อินพุตจนกว่า PD จะได้รับพัลส์เพื่อทำการรีเซ็ต และเริ่มวงรอบการเล่นกลับใหม่อีกครั้ง พัลส์ที่ขา OVF นี้สามารถใช้เริ่มต้นการทำงานของ ISD2590 ในตัวถัดไปได้เมื่อถูกต่อคาสเคดกันอยู่หลายตัว

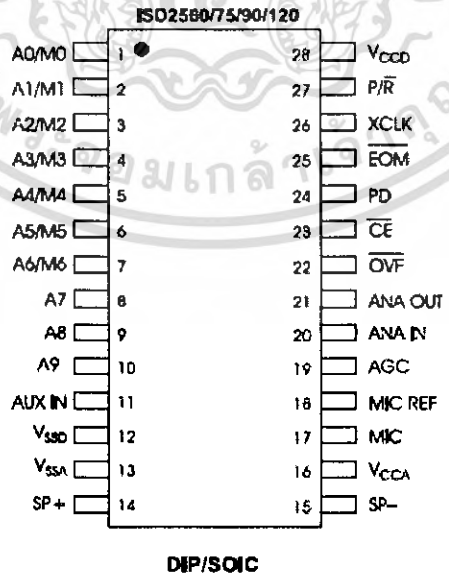
Chip Enable Input (CE) ขา 23 เป็นขา CE จะต้องได้รับพัลส์ "0" เพื่อทำให้เกิดการเปลี่ยนแปลงระหว่างการเล่นกลับและการบันทึกที่ขาแอดเดรสอินพุต และขา P/R อินพุต จะถูกแลตช์จากพัลส์ขอบขาลงของพัลส์ที่ขา CE

Power Down Input (PD) ขา 24 ในขณะที่ไม่มีการบันทึกหรือเล่นกลับ ที่ขา PD จะมีสภาวะเป็น "1" ก็จะเป็นการรักษาระดับการสิ้นเปลืองกำลังงานในระดับต่ำมากๆ แต่เมื่อขา OVF มีสภาวะเป็น "0" ที่แสดงถึงการเล่นกลับสิ้นสุดลงปกติขา PD จะมีสภาวะเป็น "1" อยู่ในขณะนั้นจะถูกรีเซต และจะเริ่มกระบวนการบันทึก หรือเล่นกลับใหม่อีกครั้ง

End of Message / Run Output (EOM) ขา 25 จะทำหน้าที่เป็นส่วนของอุปกรณ์ non-volatile ภายในตัวไอซีที่จะใช้กำหนด หรือระบุการสิ้นสุดของการเก็บข้อมูลที่ทำการบันทึกขา EOM นี้จะให้เอาต์พุตออกมาเป็น "0" เมื่อมีข้อมูลที่ถูกบันทึกอยู่ถูกเล่นกลับออกมาหมดแล้ว

External Clock Input (XCLK) ขา 26 เป็นขารับสัญญาณนาฬิกาการสุ่มสัญญาณถูกกำหนดไว้ภายในแล้วซึ่งจะไม่ขึ้นกับอุณหภูมิภายนอก หรือย่านแรงดันไฟเลี้ยงไม่คงที่ การใช้งานปกติแล้วจะต่อ ขา 25 เข้ากับกราวด์ของไฟเลี้ยง

Playback / Record Input (P/R) ขา 27 เมื่อขาอินพุต ควบคุมการเล่นกลับและการบันทึกเสียงได้รับพัลส์ "1" จะเป็นวงรอบของการเล่นกลับ และถ้าเป็นพัลส์ "0" จะเป็นการเลือกวงรอบการบันทึก หากได้รับพัลส์ที่ขอบขาลงของขา CE จะเป็นการแลตช์อินพุตที่ขา P/R



รูปที่ 2.10 การวางขาของไอซีบันทึกเสียงเบอร์ ISD2560/75/90/120

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

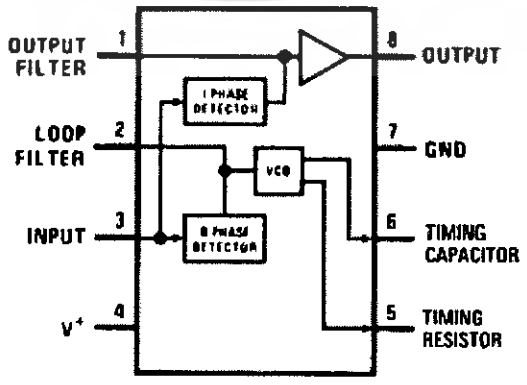
ตารางที่ 2.1 หน้าที่สำคัญของอุปกรณ์บันทึกเสียง

อุปกรณ์	หน้าที่	หลักการทำงาน
R_1 , C_4	เชื่อมต่อไฟเลี้ยงให้กับไมโครโฟน	ลดสัญญาณรบกวนจากแหล่งจ่ายไฟ
R_2 , C_2	กำหนดค่าเวลาคงตัว	ตั้งค่าเวลาคงตัวของ AGC
R_3 , R_5	ตัวต้านทานสำหรับไบแอส	ตั้งการไบแอสในการทำงานของไมโครโฟน
R_4	ตัวต้านทานอนุกรม	ลดระดับความเพี้ยนในกรณีแรงดันไฟเลี้ยงมีค่าสูง
R_6	ตัวต้านทานอนุกรม	ลดระดับแรงดันในกรณีแรงดันไฟเลี้ยงมีค่าสูง
C_1 , C_5	ตัวเก็บประจุกั้นกระแสไฟตรงจากไมโครโฟน	แยกระดับแรงดันไฟตรงของไมโครโฟนออกจากชิป
C_3	ตัวเก็บประจุผ่านความถี่ต่ำ	กำหนดตำแหน่งโพลผ่านความถี่ต่ำ
C_6 , C_7 , C_8	ตัวเก็บประจุสำหรับแหล่งจ่าย	กรองสัญญาณรบกวน

2.10 ทฤษฎีวงจรตรวจจับสัญญาณโทรศัพท์

วงจรตรวจจับสัญญาณโทรศัพท์ เป็นอุปกรณ์ที่ใช้ถอดรหัสสัญญาณความถี่ของอินพุตที่รับเข้ามา ซึ่งวงจรตรวจจับสัญญาณโทรศัพท์เป็นตัวไอซีตัวหนึ่ง ที่ภายในประกอบด้วย วงจรเฟสล็อกลูป และทรานซิสเตอร์ที่ทำหน้าที่เป็นสวิตซ์ สวิตซ์ตัวนี้จะต่อออกไปจากขาคอลเลกเตอร์ให้สัญญาณรูปสี่เหลี่ยมออกมาเมื่อสัญญาณเสียงซึ่งเป็นสัญญาณอินพุต ถูกส่งเข้ามา วงจรตรวจจับสัญญาณโทรศัพท์นี้สามารถถอดรหัสสัญญาณเสียงได้หลายความถี่

LM567 นี้จะทำงานเป็นสวิตซ์ทางเสียงที่จะตรวจจับความถี่กึ่งกลาง ซึ่งเป็นความถี่ที่ผู้ใช้งานกำหนดขึ้นในช่วงความถี่ 0.1 Hz ถึง 500 Hz และแถบกว้างความถี่หรือ แบนด์วิดท์ สามารถกำหนดค่าสูงสุดถึง 14% ของความถี่กึ่งกลาง LM567 จะทำงานในลักษณะเป็น V_{CO} โดยปกติแรงดันที่เอาต์พุตจะเป็น 5 V แต่เมื่อมีความถี่เข้ามาทางอินพุตจะทำให้ แรงดันเอาต์พุตเป็น 0 V



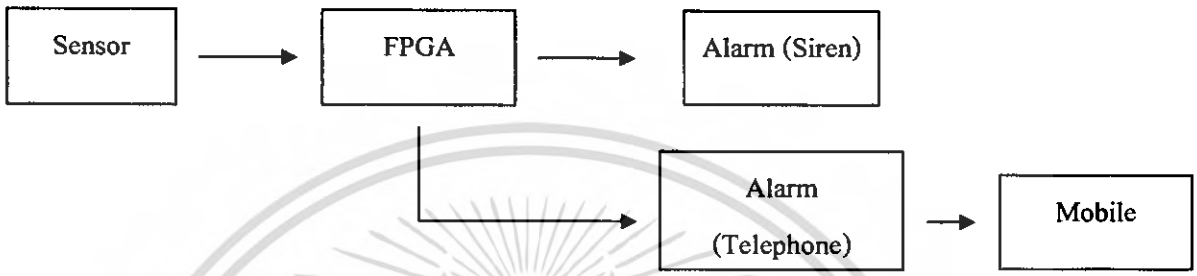
รูปที่ 2.11 วงจรภายในไอซี LM567

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบระบบดูแลและรักษาความปลอดภัยโดยใช้ FPGA

3.1 การออกแบบระบบ



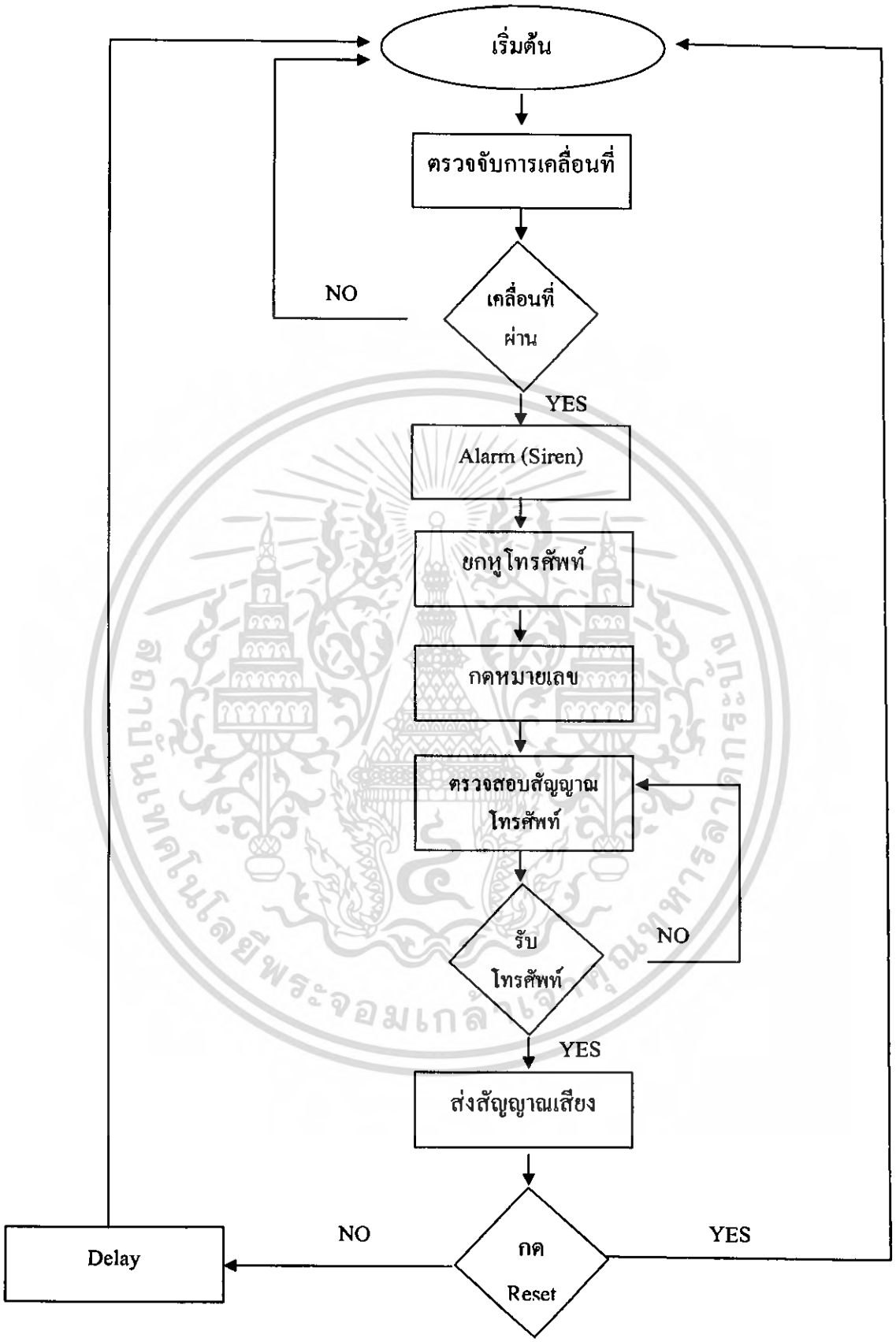
รูปที่ 3.1 บล็อกไดอะแกรมของระบบ

ในการออกแบบของระบบ นั้นจะเริ่มจากการออกแบบวงจรเซ็นเซอร์ โดนเมื่อเซ็นเซอร์ตรวจจับความผิดปกติได้ ก็จะทำการส่งเอาต์พุตไปยัง FPGA และขั้นที่สองจะเป็นการออกแบบวงจรไซเรน โดยจะรับสัญญาณอินพุตเข้ามาจาก FPGA แล้วก็จะทำการร้องเตือนเมื่อมีความผิดปกติเกิดขึ้น ขั้นที่สามจะเป็นการออกแบบวงจรแจ้งเตือนทางโทรศัพท์ โดยจะรับสัญญาณอินพุตเข้ามาจาก FPGA แล้วทำการแจ้งเตือนไปยังเลขหมายที่ได้ตั้งเอาไว้ ดังแผนผังโปรแกรม ในรูปที่ 3.2

3.2 วงจรเซ็นเซอร์ (Sensor Circuit)

ในโครงงานนี้ได้ใช้ วงจรอินฟราเรด มาใช้ในการตรวจจับความเคลื่อนไหว โดยติดตั้งตัวส่งและตัวรับ ไว้ในบริเวณที่เราต้องการ ซึ่งลักษณะการตรวจจับนั้น จะเกิดขึ้นได้เมื่อ ใช้สัญญาณอินฟราเรดจากตัวส่ง ส่งไปยังตัวรับซึ่งในสภาวะปกติที่ไม่มีอะไรมาบังระหว่างตัวส่งกับตัวรับก็จะส่งค่าเป็น Lo แสดงว่าไม่มีความเคลื่อนไหว แต่ถ้าเกิดการบังของตัวส่งและตัวรับเกิดขึ้นก็จะส่งค่าเป็น Hi แสดงว่ามีการเคลื่อนไหวเกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แผนผังโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

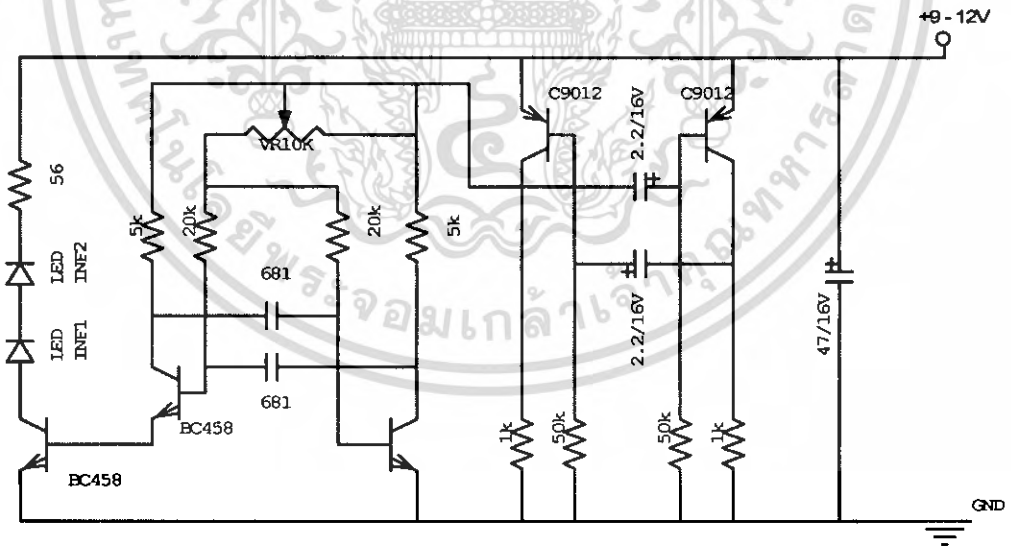
หลักการเลือกเซนเซอร์ให้เหมาะสมกับงาน

เนื่องจากเซนเซอร์ที่มีอยู่ในปัจจุบันนี้มีอยู่หลายชนิดด้วยกัน ในการเลือกใช้จึงจำเป็นต้องเรียนรู้รายละเอียดคุณสมบัติต่างๆ ของเซนเซอร์ ตลอดจนถึงลักษณะการใช้งาน เพื่อที่จะเลือกใช้งานได้อย่างถูกต้องเหมาะสมมากที่สุด ดังนั้นการเลือกใช้จึงคำนึงถึง

- ขนาดของวัตถุที่ตรวจจับ
- ชนิดของวัตถุที่ตรวจจับ
- ความแม่นยำของตำแหน่งที่ตรวจจับ
- ความเร็วในการเคลื่อนที่ของวัตถุที่ตรวจจับ
- สภาพแวดล้อมที่ใช้เซนเซอร์
- ระยะห่างระหว่างเซนเซอร์ที่อยู่ใกล้กัน

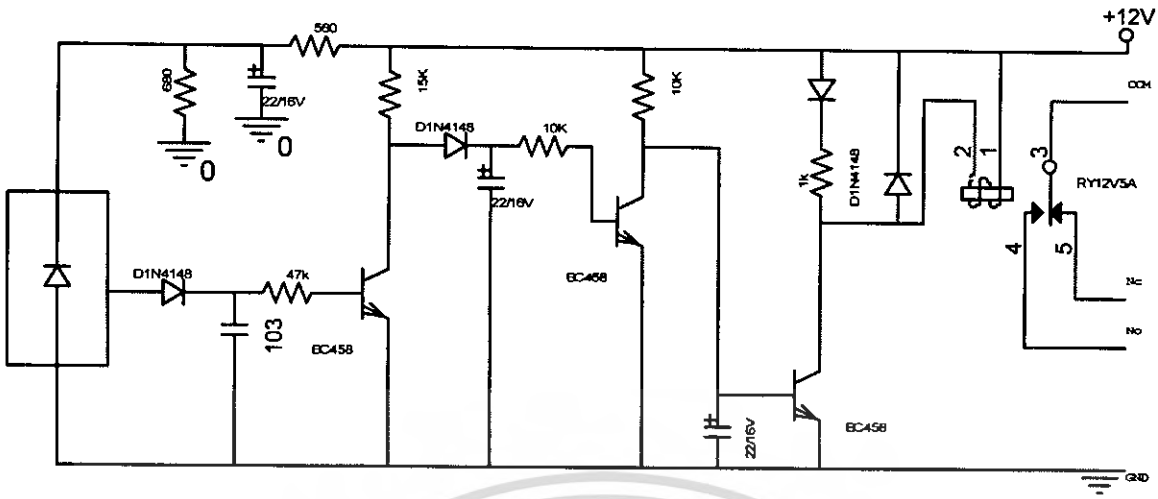
โดยเซนเซอร์ที่ใช้ในโครงงาน คือรุ่น EPS937RX,TX ซึ่งมีคุณสมบัติดังนี้

- วงจรรับใช้ไฟ DC 12V
- วงจรส่งใช้ไฟ DC 9-12V
- หลอดอินฟาเรดความถี่ $0.6*10^{14}$ Hz ความยาวคลื่น 10 μM ให้พลังงาน 2 kcal/mol
- มีระยะทางในการตรวจจับการสะท้อนแบบตรงประมาณ 1-3 เมตร ขึ้นอยู่กับสภาพแวดล้อมที่ติดตั้ง

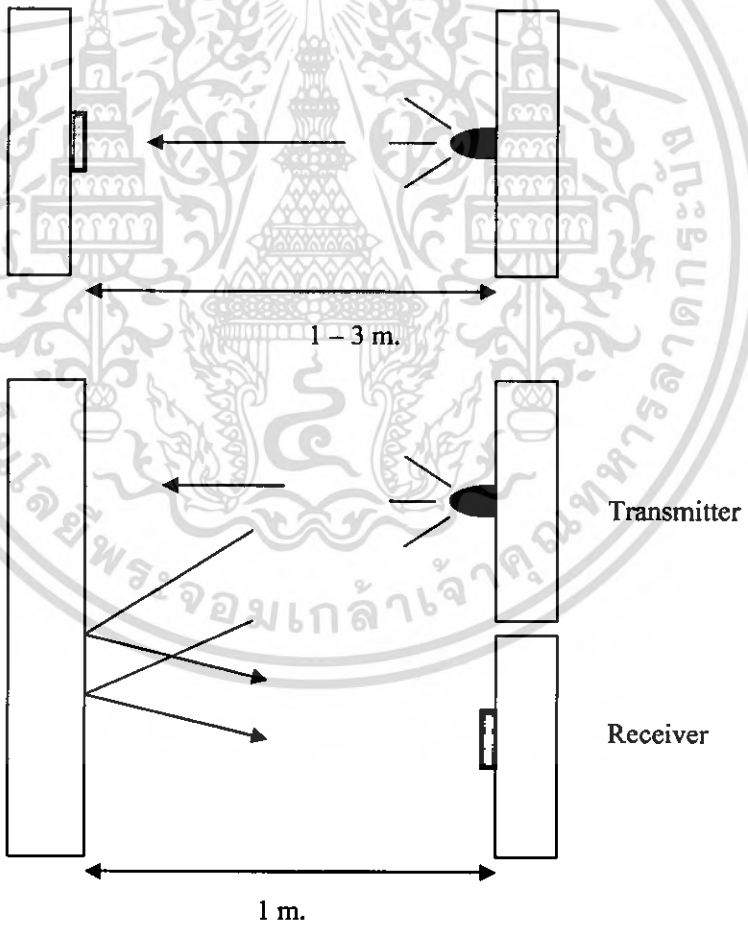


รูปที่ 3.3 รูปวงจรตัวส่ง EPS937TX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 รูปวงจรตัวรับสัญญาณ EPS937RX

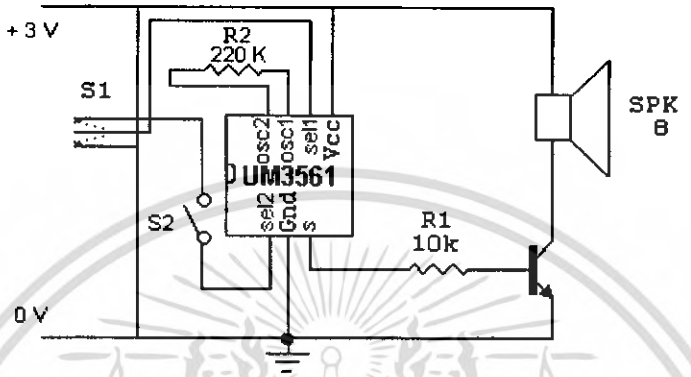


รูปที่ 3.5 รูปแบบการตรวจจับของ เซนเซอร์

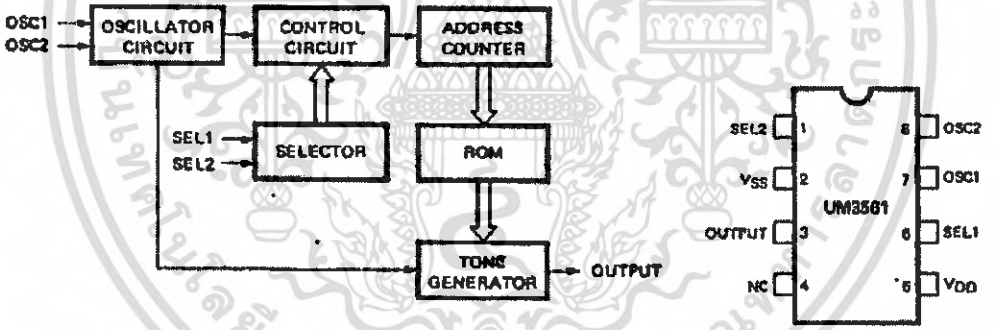
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วงจรไซเรน (Siren Circuit)

ในการออกแบบวงจรไซเรนนี้ได้ใช้ IC UM3561 ในการเป็นตัวกำเนิดเสียงซึ่งสามารถกำเนิดเสียงได้ 3 แบบคือเสียงไซเรนหวอรถพยาบาล หวอรถตำรวจ และ เสียงสัญญาณกันขโมย โดยมีรายละเอียดตามนี้



รูปที่ 3.6 วงจรไซเรน



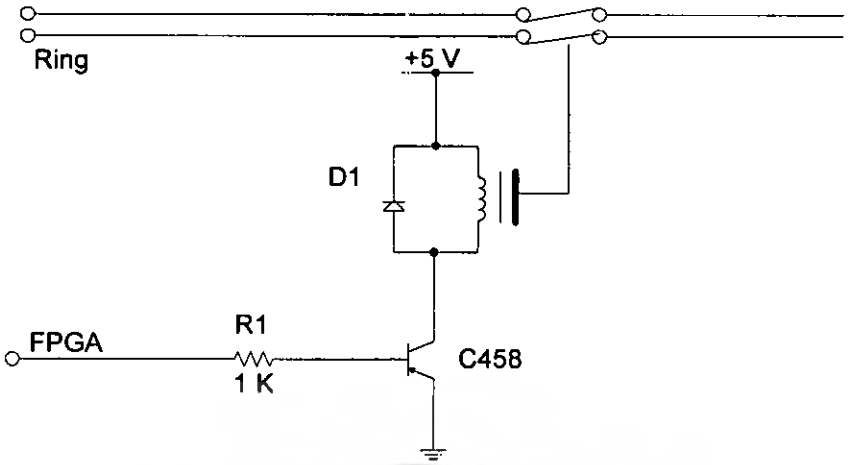
รูปที่ 3.7 บล็อกไดอะแกรมวงจรภายในไอซี

3.4 วงจรแจ้งเตือนทางโทรศัพท์

3.4.1 การออกแบบวงจรตัดต่อคู่สายโทรศัพท์

จากรูปจะเห็นว่าทรานซิสเตอร์ จะทำหน้าที่เป็นสวิตช์ควบคุมการทำงานของรีเลย์โดยทรานซิสเตอร์จะถูกควบคุมโดยชิป FPGA ซึ่งสั่งให้ทรานซิสเตอร์ทำงานได้โดยกำหนดให้มีสภาวะเป็น “1” หรือเมื่อมีแรงดัน 5 V เพื่อไบแอสให้กับตัวทรานซิสเตอร์ เมื่อมีการทำงานเกิดขึ้น รีเลย์จะมีการตัดต่อคู่สายโทรศัพท์ ซึ่งจะทำให้ทางชุมสายโทรศัพท์รับรู้ว่ามีการยกหูโทรศัพท์ และทำการกดหมายเลขต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 วงจรตัดต่อคู่สายโทรศัพท์

3.4.2 การออกแบบวงจรตรวจจับสัญญาณโทรศัพท์

วงจรตรวจจับสัญญาโทรศัพท์ เป็นอุปกรณ์ที่ใช้ถอดรหัสความถี่เสียง ซึ่งในโครงงานนี้ได้ใช้ไอซีเบอร์ LM567 ซึ่งเป็นโทนดีโคดเดอร์มีลักษณะการทำงานเป็นสวิตซ์ โดยจะกินกำลังไฟสูงในช่วงแรกขณะรับอินพุตอย่างต่อเนื่องทางอินพุต ในการทำงานเป็นสวิตซ์นั้น จะทำการตรวจจับความถี่กลางซึ่งเป็นความถี่ที่ถูกกำหนดให้เป็นค่าได้ในช่วง 0.1 ถึง 5 kHz ความกว้างของแถบความถี่หรือแบนด์วิดท์ สามารถกำหนดค่าได้สูงถึง 14% ของความถี่กลาง

วงจรควบคุมแรงดันอินพุต (Voltage Control Oscillator ; V_{CO}) สามารถเพิ่มความกว้างของช่องความถี่ ซึ่งได้จากการกำหนดค่าของ R_1 และ C_1 ความถี่ควบคุมนี้จะอยู่ในช่วงแคบ ๆ เท่านั้น ๆ โดยสัญญาณอินพุตเข้ามาทางขา 3 ผลจากความถี่ในช่วงแคบ ๆ ทำให้วงจรแฟสล็อกลูป ล็อกช่วงความถี่แคบ ๆ ของสัญญาณอินพุต

เฟสดีเทคเตอร์จะเปรียบเทียบความถี่ และเฟสของสัญญาณอินพุตเมื่อมีสัญญาณการเปรียบเทียบทั้งสองสอดคล้องกันจะมีค่าเท่ากับความถี่ ฟรีรันนิ่งของออสซิลเลเตอร์และค่าแบนด์วิดท์จะมีค่าเท่ากับช่วงความถี่ที่ถูกล็อกในวงจรแฟสล็อกลูป

ความถี่กลาง (f_0 : Center Frequency) ของวงจรออสซิลเลเตอร์สามารถคำนวณได้จาก

สมการ $$f_0 = 1/1.1R_C \quad (3.1)$$

โดยเลือกค่าความถี่กลาง f_0 ที่ต้องการคือ 425 Hz

กำหนดให้ $$C_1 = 0.1\ \mu F \quad (3.2)$$

ดังนั้นหาค่า $$R_1 = \frac{1.1}{(0.1 \times 425)} \quad (3.3)$$

$$= 25\ k\Omega$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นเลือกตัวเก็บประจุ C_2 เพื่อเป็นตัวกำหนดค่าความกว้างของแบนด์วิดท์โดยหาได้จากกราฟข้อมูลอุปกรณ์ จากกราฟจะพบว่ายิ่ง C_2 มีค่าเพิ่มขึ้นจะทำให้ค่าเวลาคงตัวมากขึ้น เวลาที่ใช้ในการล็อกสัญญาณก็จะนานขึ้นและทำให้ค่า %BW ลดลงซึ่งคำนวณได้จาก

สมการ $$\%BW = 1070 \times \left(\frac{V_i}{f_0 \times C_2} \right)^{1/2}$$ $V_i < 200\ mV_{rms}$ (3.4)

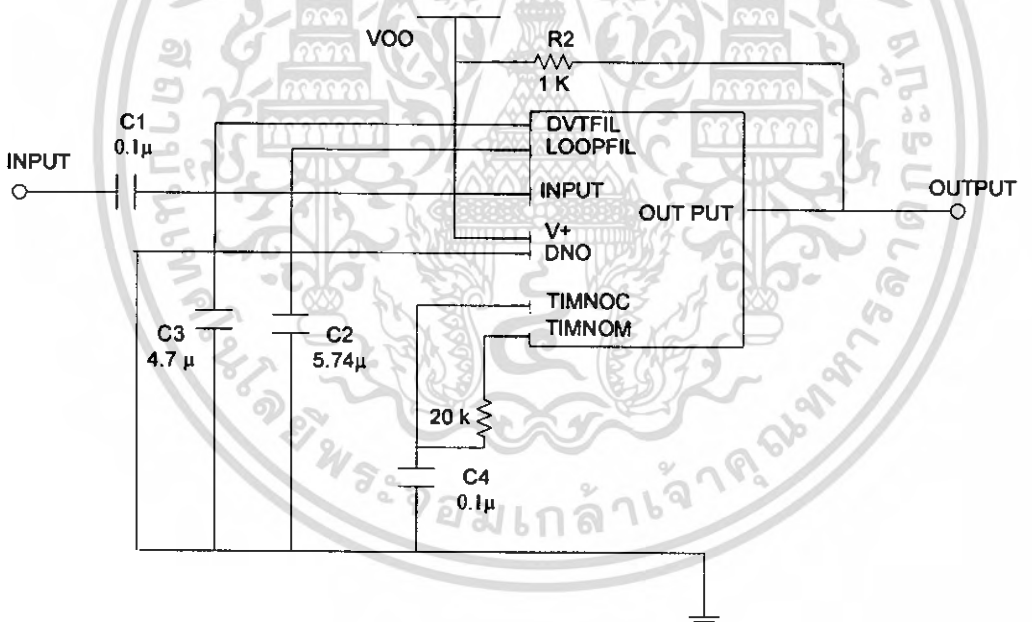
ต้องการ $\%BW = 12\%$ กำหนดให้ $V_i = 200\ mV_{rm}$

ดังนั้น $$C_2 = \frac{V_i}{f_0} \times \left(\frac{1070}{\%BW} \right)^2$$ (3.5)

$$= \frac{200mV}{425Hz} \times \left(\frac{1070}{12} \right)^2$$

$$= 3.74\ \mu F$$

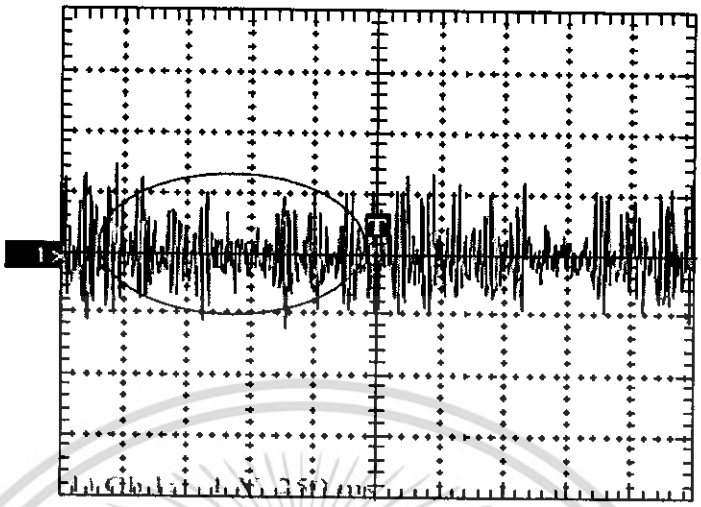
ส่วน C_3 จะเป็นตัวกำหนดความเร็วในการทำงานของส่วนเอาต์พุต คือสามารถตอบสนองได้เร็วเมื่อตัวตรวจจับเฟสสามารถตรวจจับสัญญาณได้ โดยปกติ C_3 จะประมาณเกือบ 2 เท่าของ C_2 ดังนั้นเลือกใช้ค่า $C_3 = 4.7\ \mu F$



รูปที่ 3.9 วงจรตรวจจับสัญญาณโทรศัพท์

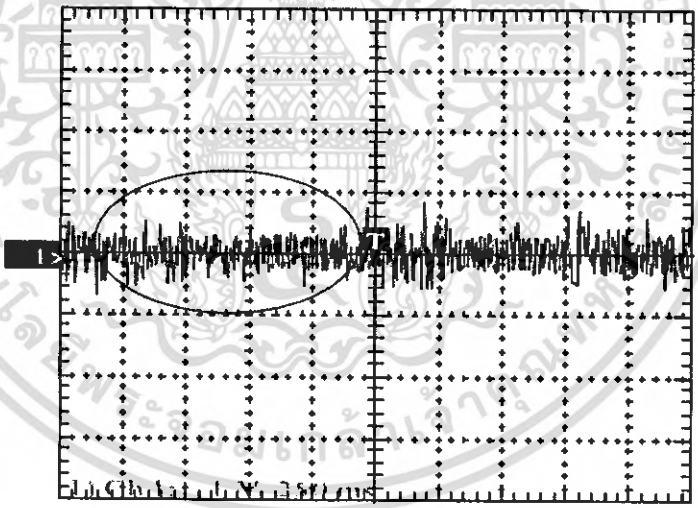
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 สัญญาณที่จับได้จากโทรศัพท์



รูปที่ 3.10 สัญญาณโทรศัพท์ขณะไม่มีการรับสาย

จากรูปที่ 3.9 จะเห็นว่ามีสัญญาณความถี่ประมาณ 400 Hz สลับกับ 50 Hz



รูปที่ 3.11 สัญญาณโทรศัพท์ขณะมีการรับสาย

จากรูปที่ 3.10 จะเห็นว่าไม่มีสัญญาณที่ความถี่ 400 Hz เกิดขึ้น

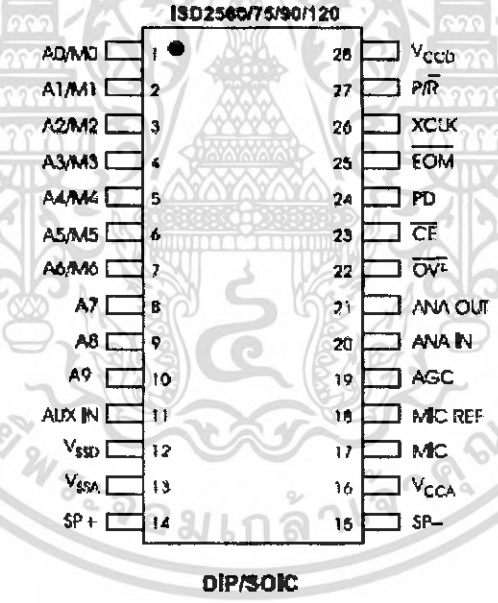
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4 การออกแบบวงจรบันทึกเสียงและเล่นกลับ

ก่อนอื่นต้องทราบหลักการทำงาน และคุณสมบัติทางไฟฟ้าของไอซีบันทึกเสียงก่อนว่าแต่ละเบอร์มีความแตกต่างกันอย่างไร ซึ่งจะทำให้การเลือกใช้งานอุปกรณ์ได้เหมาะสมที่สุด โดยดูจากตารางซึ่งจะยกมาแสดงเฉพาะบางเบอร์ที่ได้รับความนิยมในปัจจุบัน

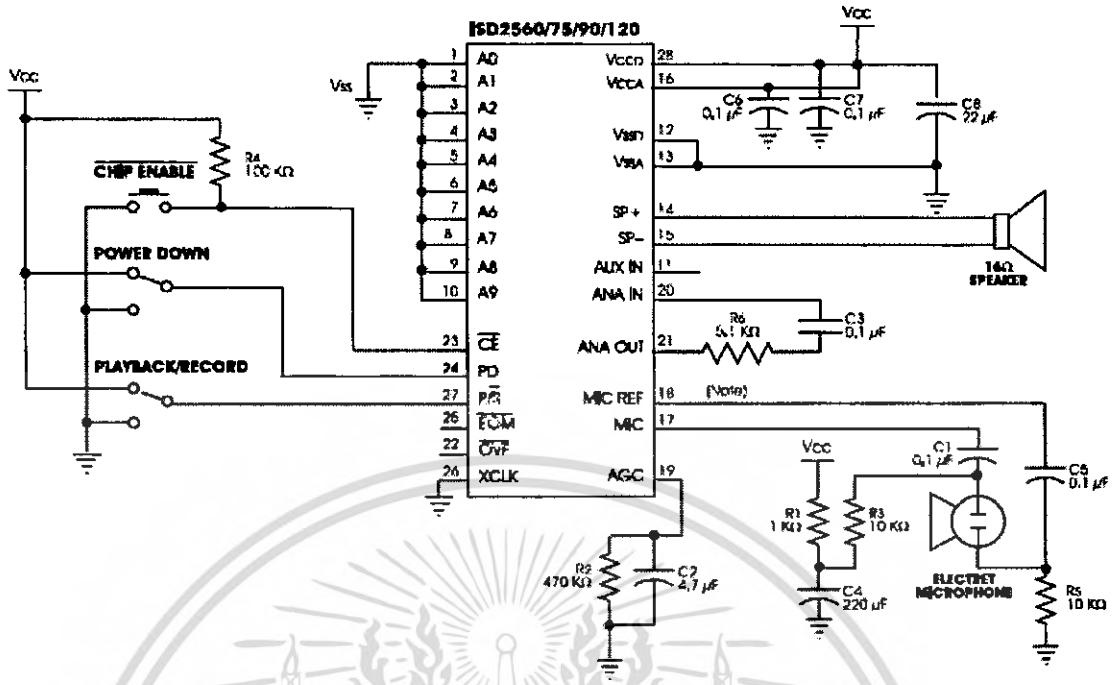
ตารางที่ 3.1 คุณสมบัติของไอซีบันทึกเสียงตระกูล ISD25xx

Part Number	Duration (Seconds)	Input Sample Rate (KHz)	Typical Filter Pass Band (KHz)
ISD2560	60	8.0	3.4
ISD2575	75	6.4	2.7
ISD2590	90	5.3	2.3
ISD25120	120	4.0	1.7



รูปที่ 3.12 ตำแหน่งการวางขาไอซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 วงจรของการออกแบบไอซี

โดยในโครงงานนี้จะเลือกใช้เบอร์ ISD2590 ซึ่งได้พิจารณาแล้วว่าเหมาะสมกับความยาวของข้อความที่ได้ทำการบันทึก โดยไอซีนี้สามารถบันทึกได้นานถึง 90 วินาที ความละเอียดของแต่ละแอดเดรสมีค่าเท่ากับ 150 มิลลิวินาที ซึ่งมีความละเอียดเพียงพอที่จะไม่ทำให้เสียงที่บันทึกจากต้นกำเนิดเสียงเกิดความผิดเพี้ยนเมื่อทำการเล่นกลับ มีแอดเดรสที่สามารถนำไปใช้งานได้ทั้งหมด 600 แอดเดรส ไอซีบันทึกเสียงที่ขาของหน่วยความจำ 10 เส้น (A0 ถึง A9) หรือ 2^{10} เท่ากับ 1024 แอดเดรส แต่มีแอดเดรสสำหรับเก็บข้อมูลเสียง 600 แอดเดรส ซึ่งในที่นี้ได้ใช้ไอซีบันทึกเสียงซึ่งบันทึกได้ 90 วินาที ดังนั้นสามารถคำนวณหาแอดเดรสต่อวินาทีที่ได้ ดังนี้

600 แอดเดรส สามารถบันทึกได้ 90 วินาที

ดังนั้น 1 แอดเดรสสามารถบันทึกได้ 90/600 x 1 = 150 ms

เวลา 150 ms เก็บข้อมูลได้ 1 แอดเดรส

ถ้าต้องการ 15 วินาที ต้องใช้ $1/(150 \times 10^{-3})$ z 15 = 100 แอดเตรส

ตารางที่ 3.2 ข้อความที่บันทึกในตัวไอซี

ข้อความที่	ข้อความที่บันทึก	เวลาที่ใช้ (วินาที)
1	คำเตือน ขณะนี้เกิดการบุกรุก	15

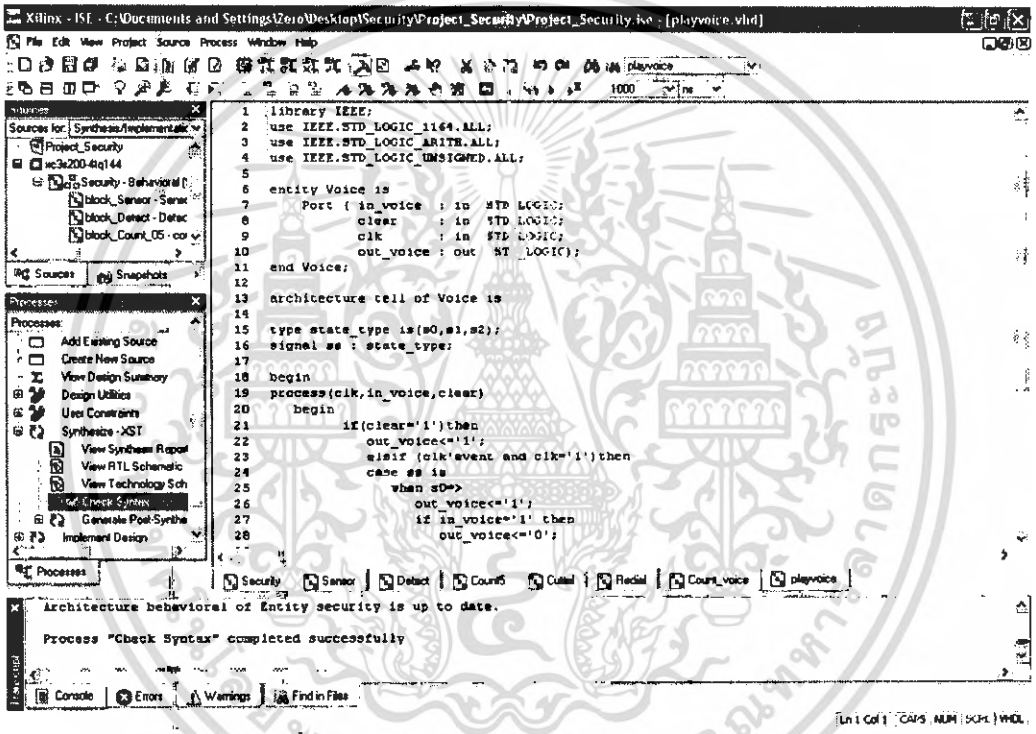
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ผลการทดลองการใช้งานในส่วนโปรแกรมของ FPGA

โปรแกรมของ FPGA นี้เขียนโดยใช้โปรแกรม ISE WebPack 8.1i ซึ่งตัวโปรแกรมสามารถรองรับภาษา VHDL และ Verilog โดยโปรแกรมที่ใช้ในการทำโครงงานนี้ได้เขียนด้วยภาษา VHDL ซึ่งจะมีรูปร่างของโปรแกรมดัง รูปที่ 4.1

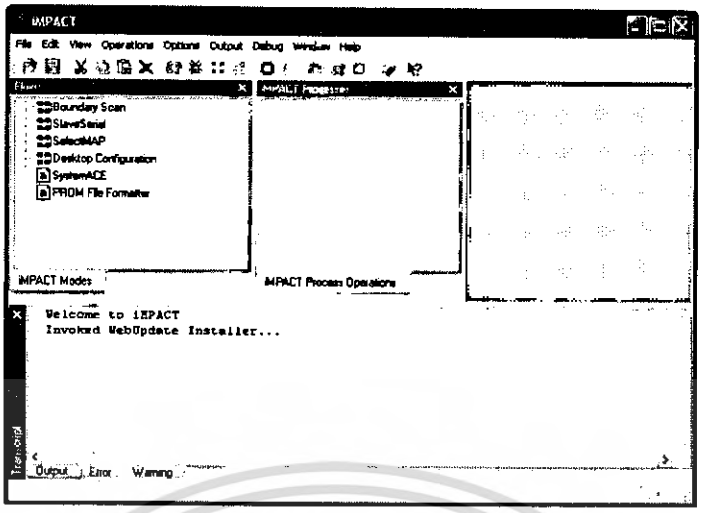


รูปที่ 4.1 ส่วนโปรแกรมของ ISE WebPack 8.1i

4.2 ผลการทดลองการใช้งานในส่วนโปรแกรมของ iMPACT

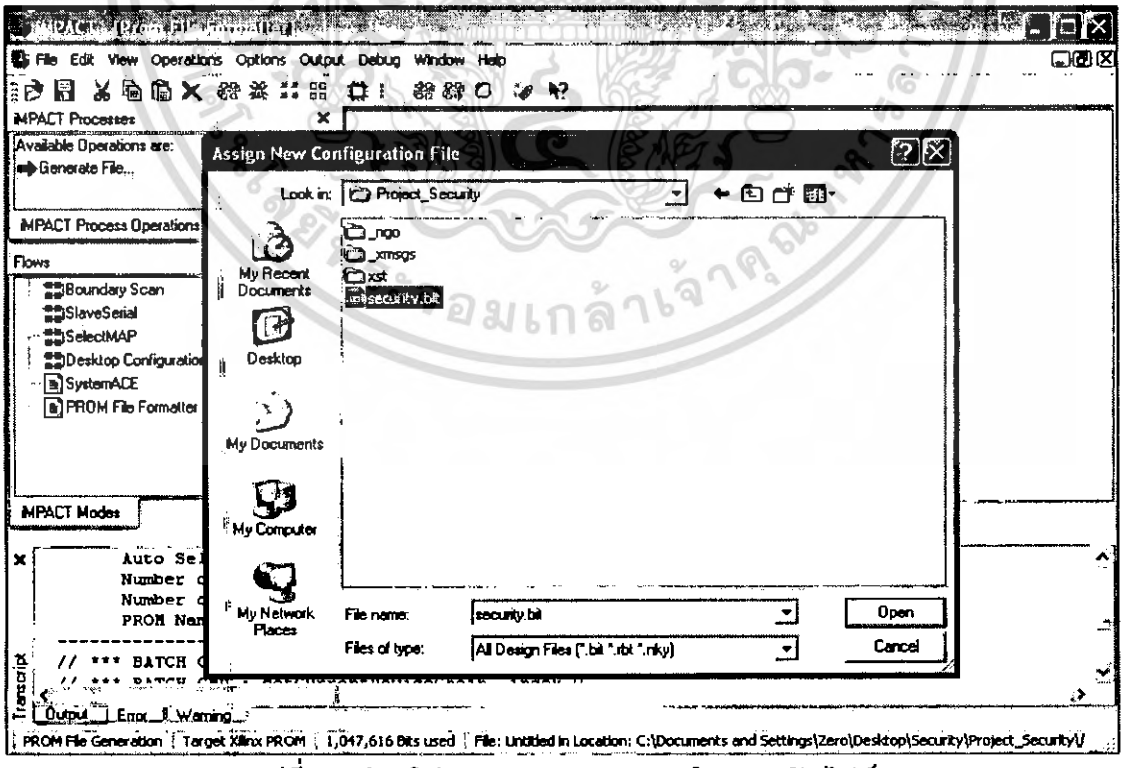
การใช้งานในส่วนโปรแกรมของ iMPACT นี้เป็นโปรแกรมที่ใช้งานในส่วนของการเบิร์นโปรแกรมที่เขียนขึ้นมาลงไปใน FPGA โดยจะมีรูปร่างของโปรแกรมดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



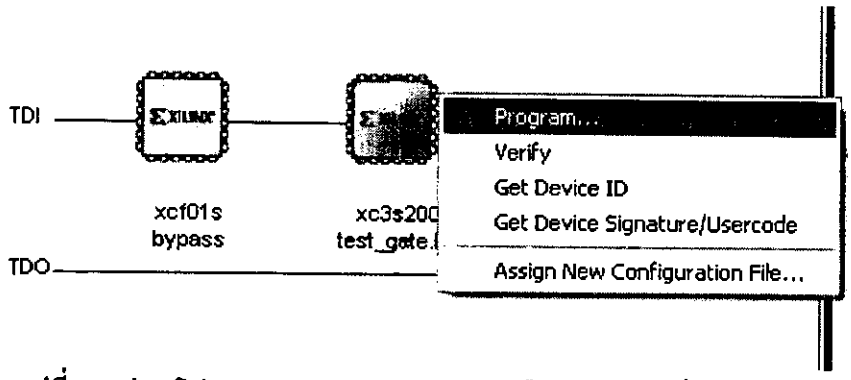
รูปที่ 4.2 ส่วนโปรแกรมของ iMPACT

ในการเปิดไฟล์ที่จะนำมาใช้ในการเบิร์นนั้นจะสามารถใช้ไฟล์ได้ 2 ประเภทคือ .bit และ .mcs โดยไฟล์ .bit นั้นเมื่อเบิร์นลงไปในบอร์ด FPGA แล้วถ้าเราตัดแหล่งจ่ายไฟให้ตัวโปรแกรมที่เบิร์นลงไปก่อนหน้านี้ก็จะหายไป เนื่องจาก FPGA เองไม่สามารถที่จะเก็บโปรแกรมไว้ในตัวเองได้ ดังนั้นจึงต้องใช้ไฟล์ .mcs ซึ่งจะใช้ในการเบิร์นลง Flash prom โดยเมื่อ FPGA ได้เปิดขึ้นมาก็จะทำการดึงโปรแกรมจาก Flash prom ไปใช้โดยอัตโนมัติ



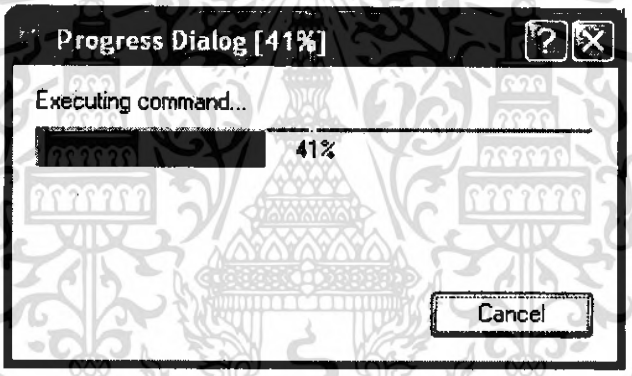
รูปที่ 4.3 ส่วนโปรแกรมของ iMPACT ในขณะเปิดไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

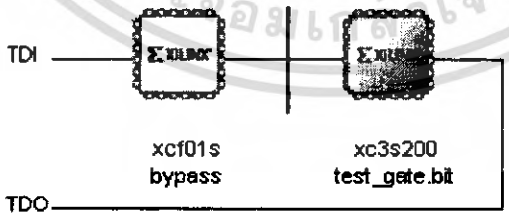


รูปที่ 4.4 ส่วนโปรแกรมของ iMPACT ขณะเลือกทำการเบิร์นโปรแกรม

แสดงสถานการณ์การทำงานของโปรแกรม iMPACT ขณะทำการเบิร์นโปรแกรมลงใน FPGA ดัง รูปที่ 4.5 โดยในการเบิร์นโปรแกรมสามารถที่จะเบิร์นโดยผ่านทางสาย JTAG เท่านั้นซึ่ง เมื่อเบิร์นเสร็จแล้วก็จะมีการแสดงผลว่า Program Succeeded ดัง รูปที่ 4.6



รูปที่ 4.5 ส่วนโปรแกรมของ iMPACT ขณะทำการเบิร์นโปรแกรม

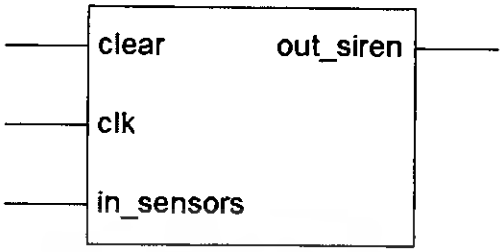


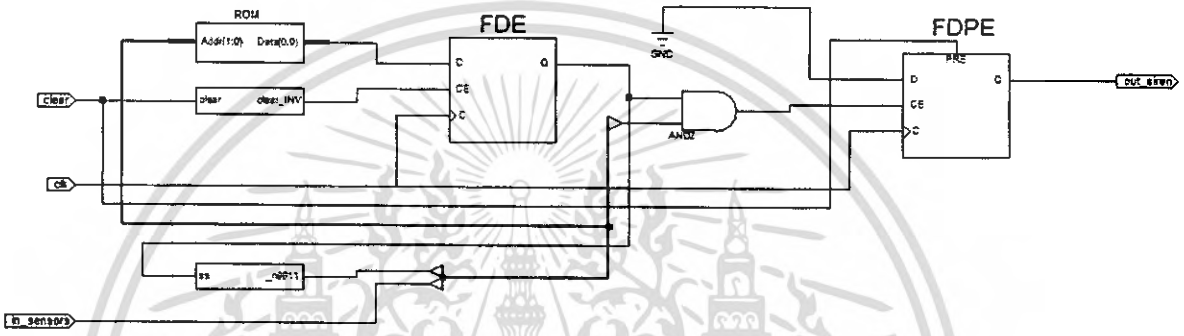
รูปที่ 4.6 ส่วนโปรแกรมของ iMPACT เมื่อทำการเบิร์นเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

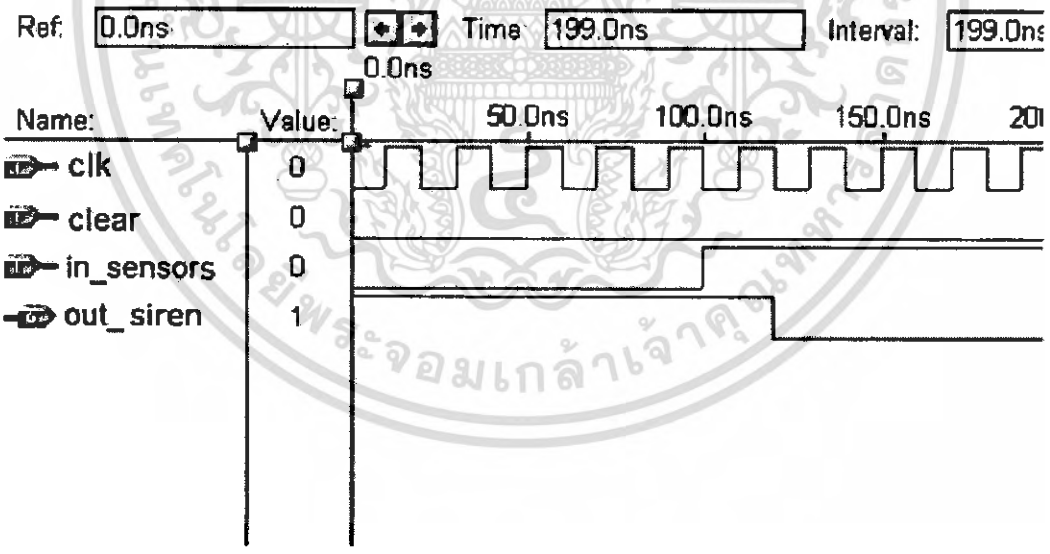
4.3 ผลการจำลองการทำงานของโปรแกรม

4.3.1 โปรแกรมตรวจจับการเคลื่อนไหว





รูปที่ 4.7 วงจรของโปรแกรมตรวจจับการเคลื่อนไหว



รูปที่ 4.8 ผลการจำลองของโปรแกรมตรวจจับการเคลื่อนไหว

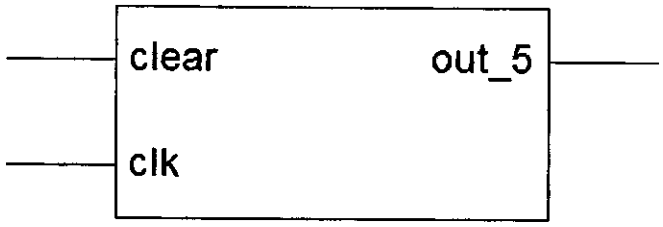
in_sensors คือ สัญญาณอินพุตที่รับจากอินฟราเรดเซ็นเซอร์เข้ามายัง FPGA

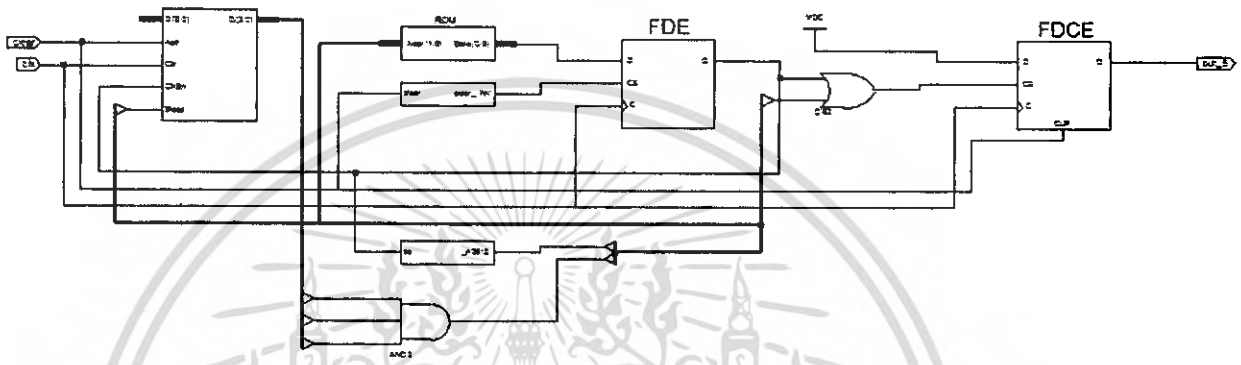
out_siren คือ สัญญาณเอาต์พุตที่ได้จากโปรแกรมตรวจจับการเคลื่อนไหว

เมื่ออยู่ในสภาวะปกติ หรือไม่มีสิ่งเคลื่อนที่ผ่านเซ็นเซอร์ สัญญาณอินพุตที่เข้ามาจะมีสภาวะเป็น "0" จนกระทั่งเมื่อมีการตรวจจับสิ่งเคลื่อนที่ผ่านเซ็นเซอร์ได้ สัญญาณอินพุตที่เข้ามาจะมีสภาวะเป็น "1"

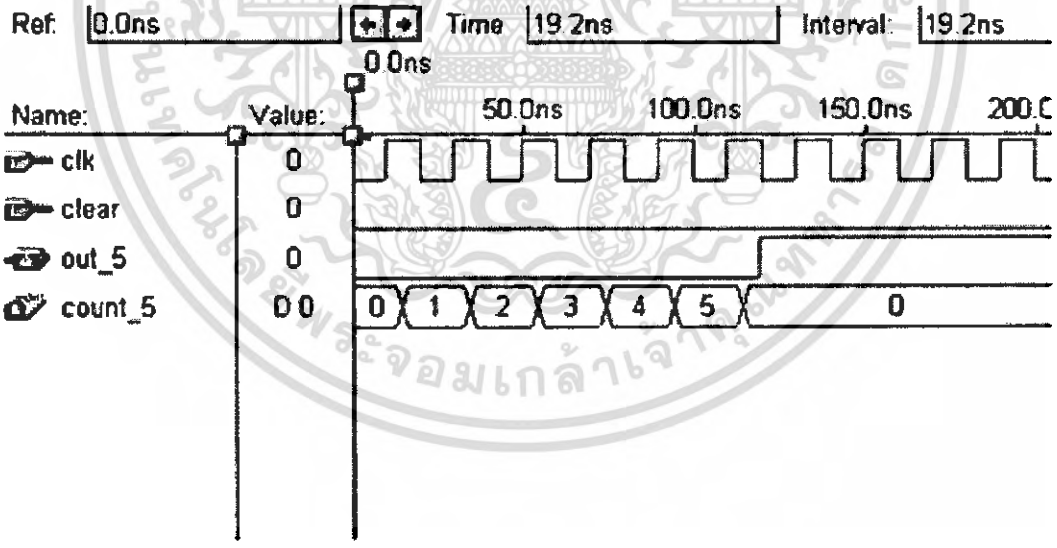
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 โปรแกรม Count5





รูปที่ 4.9 วงจรของโปรแกรม Count5



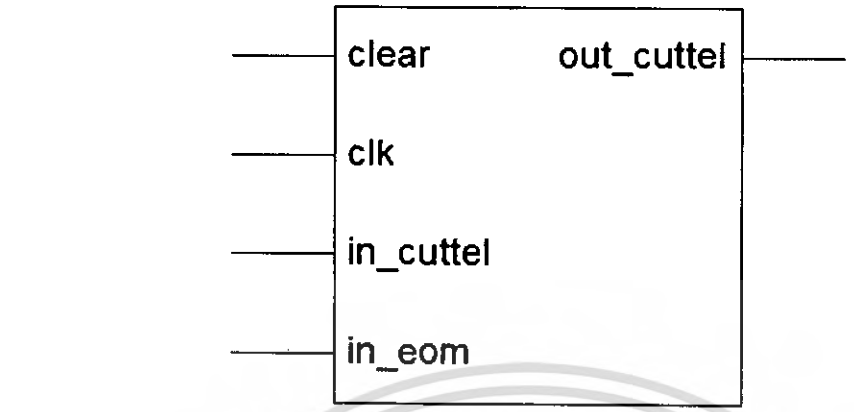
รูปที่ 4.10 ผลการจำลองการทำงานของโปรแกรม Count5

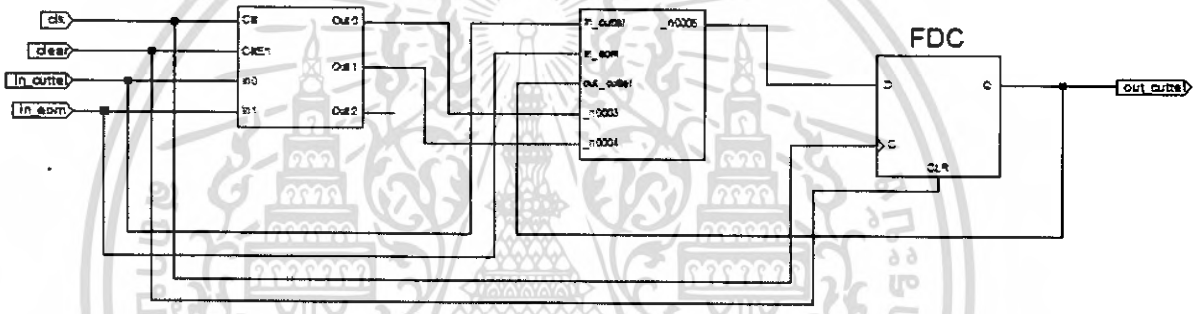
clear คือ สัญญาณอินพุตที่รับจากโปรแกรมตรวจจับการเคลื่อนไหว

out_5 คือ สัญญาณเอาต์พุตที่ได้จากโปรแกรมCount5

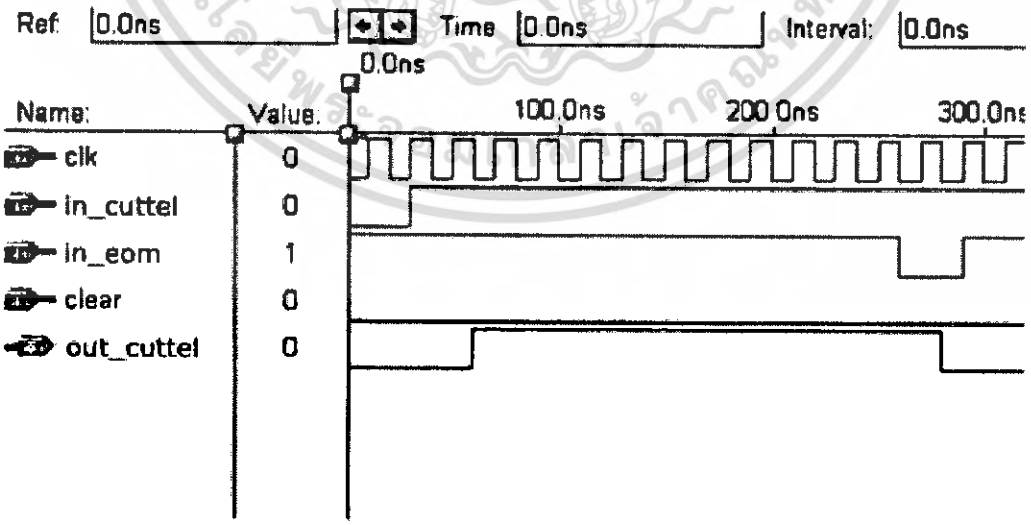
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 โปรแกรมตัดต่อคู่สายโทรศัพท์





รูปที่ 4.11 วงจรของโปรแกรมตัดต่อคู่สายโทรศัพท์



รูปที่ 4.12 ผลการจำลองการทำงานของโปรแกรมตัดต่อคู่สายโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

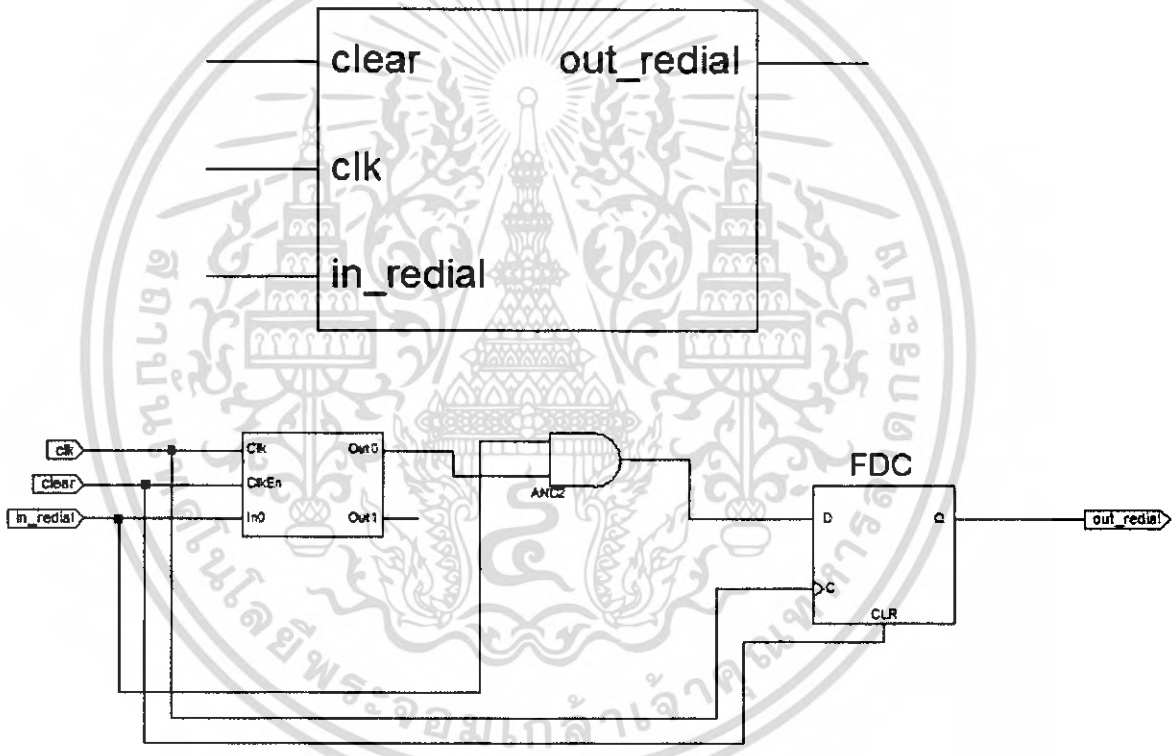
in_eom คือ สัญญาณอินพุตที่รับมาจากวงจรเล่นเสียงโดยจะมีสภาวะเป็น “1” จนกว่าจะมีการเล่นเสียงจบแล้วจะมีสภาวะเป็น “0”

in_cuttel คือ สัญญาณอินพุตของโปรแกรมที่รับมาจากโปรแกรมCount5

out_cuttel คือ สัญญาณเอาต์พุตที่ได้จากโปรแกรมตัดต่อคู่สายโทรศัพท์

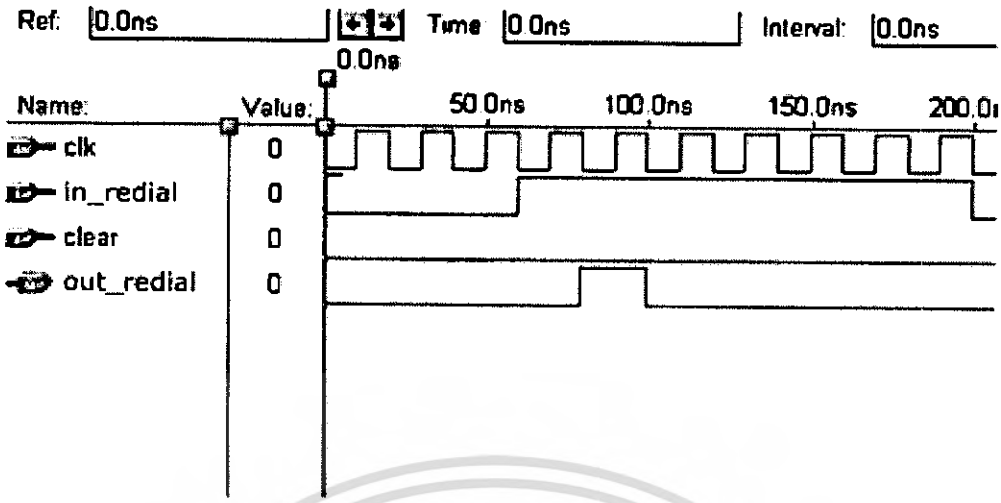
เมื่อมีสัญญาณอินพุตจากโปรแกรมCount5 เข้ามา โปรแกรมจะทำการสั่งให้รีเลย์ทำการสับสวิตซ์เพื่อต่อสายโทรศัพท์ จนกว่าจะมีสัญญาณ eom เป็น “0” เข้ามาแล้วจึงให้ทำการตัดสายโทรศัพท์

4.3.4 โปรแกรมกดหมายเลขโทรศัพท์



รูปที่ 4.13 วงจรของโปรแกรมกดหมายเลขโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



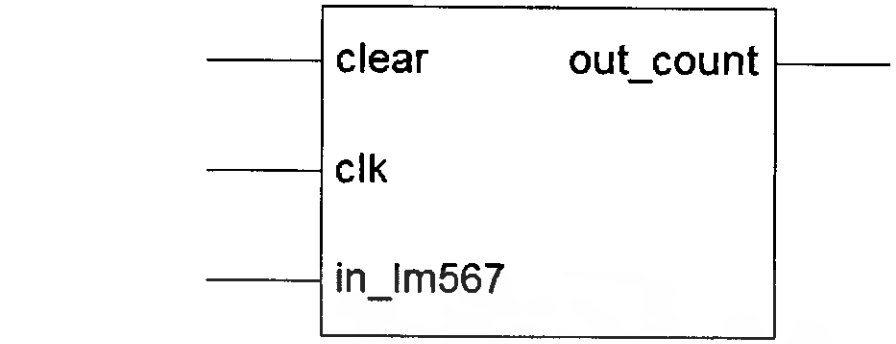
รูปที่ 4.14 ผลการจำลองการทำงานของโปรแกรมกดหมายเลขโทรศัพท์

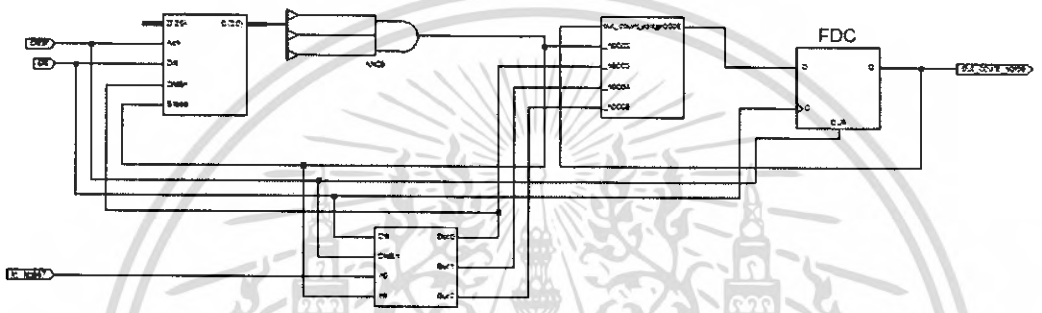
in_redial คือ สัญญาณอินพุตของโปรแกรมที่รับมาจากโปรแกรมตัดต่อคู่สายโทรศัพท์
out_redial คือ สัญญาณเอาต์พุตที่ออกจากโปรแกรมกดหมายเลขโทรศัพท์

เมื่อได้รับสัญญาณอินพุต in_redial มาจากโปรแกรมตัดต่อคู่สายโทรศัพท์แล้วจะทำการกดหมายเลขโทรศัพท์โดยจะมีสัญญาณเอาต์พุตสถานะเท่ากับ "1" ออกมาจากขา out_redial เมื่อทำการกดเสร็จแล้วจะมีเอาต์พุตเท่ากับ "0"

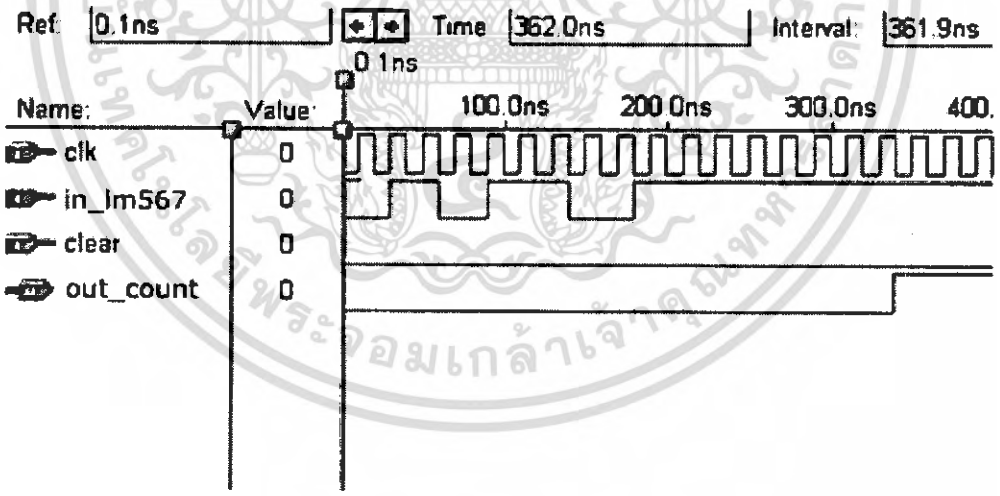
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.5 โปรแกรมตรวจจับสัญญาณโทรศัพท์





รูปที่ 4.15 วงจรของโปรแกรมตรวจจับสัญญาณโทรศัพท์



รูปที่ 4.16 ผลการจำลองการทำงานของโปรแกรมตรวจจับสัญญาณโทรศัพท์

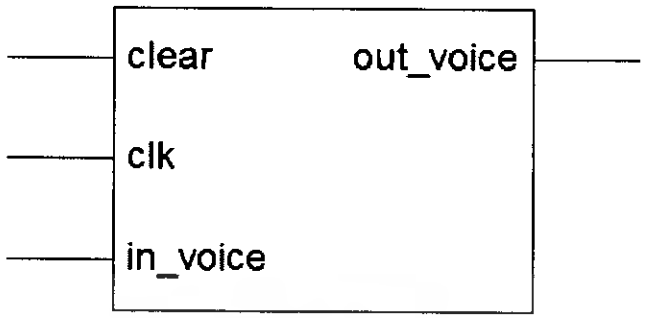
in_lm567 คือ สัญญาณอินพุตของโปรแกรมที่รับมาจากวงจรตรวจจับสัญญาณโทรศัพท์

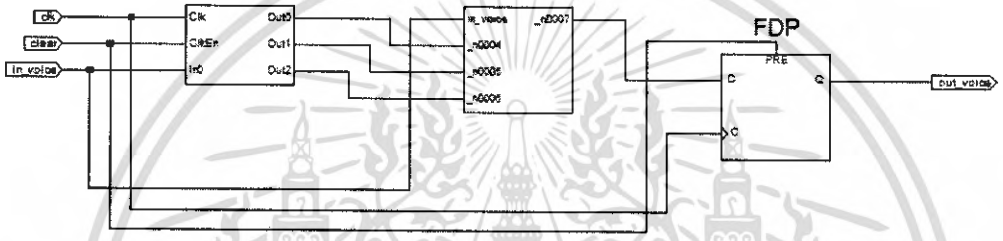
out_count คือ สัญญาณเอาต์พุตที่ได้จากโปรแกรมตรวจจับสัญญาณโทรศัพท์

เมื่อหลังจากโปรแกรมหน่วงเวลาทำงานเรียบร้อยแล้ว และโทรศัพท์ก็ได้ทำโทรออกไปแล้ว โปรแกรมตรวจจับสัญญาณโทรศัพท์จะทำการตรวจสอบสัญญาณเรียกกลับคือสัญญาณที่ได้ยินเมื่อมีการรับสายแล้ว และทำการส่งค่าเอาต์พุตออกไปยังโปรแกรมเล่นเสียงต่อไป

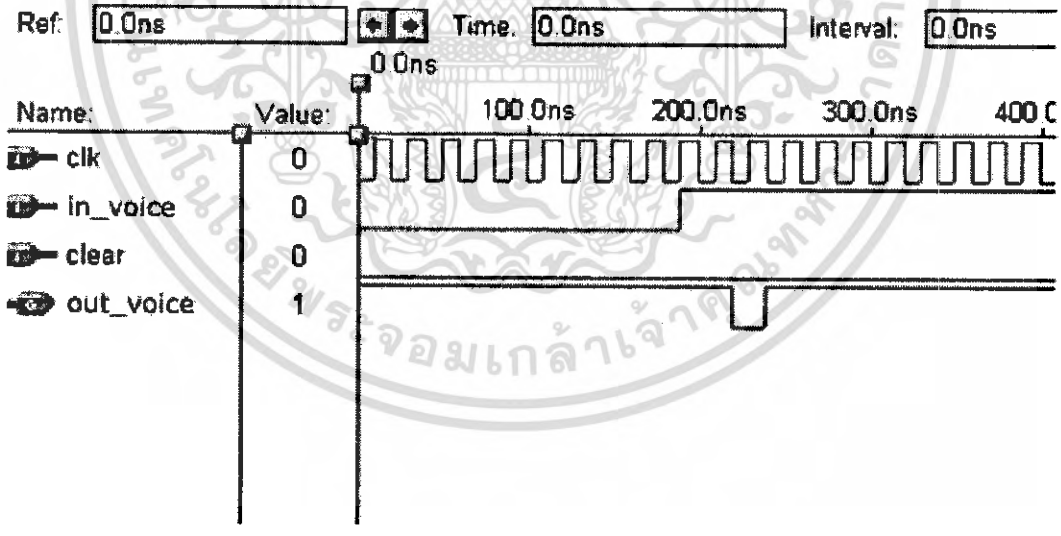
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.6 โปรแกรมเล่นเสียง





รูปที่ 4.17 วงจรของโปรแกรมเล่นเสียง



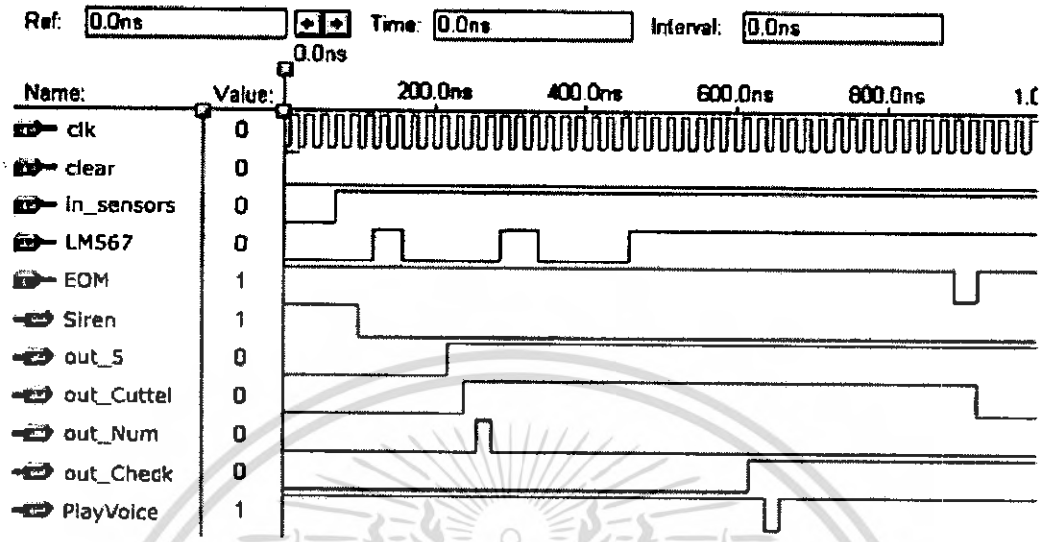
รูปที่ 4.18 ผลการจำลองการทำงานของโปรแกรมเล่นเสียง

in_voice คือ สัญญาณอินพุตที่รับมาจากโปรแกรมตรวจจับสัญญาณโทรศัพท์

out_voice คือ สัญญาณเอาต์พุตที่ได้จากโปรแกรมเล่นเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

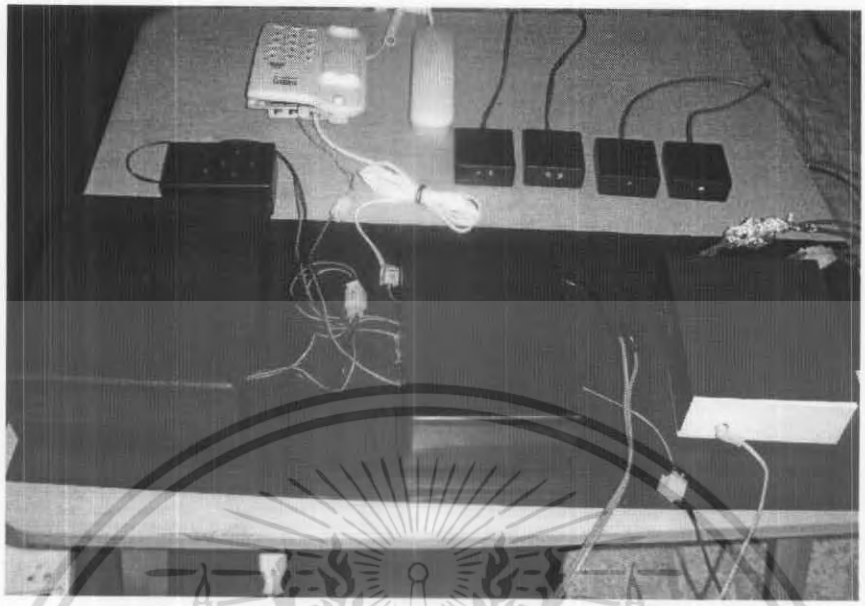
4.3.7 โปรแกรมทั้งหมด



รูปที่ 4.19 ผลการจำลองการทำงานของโปรแกรมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ผลการทดลองการทำงานของระบบ



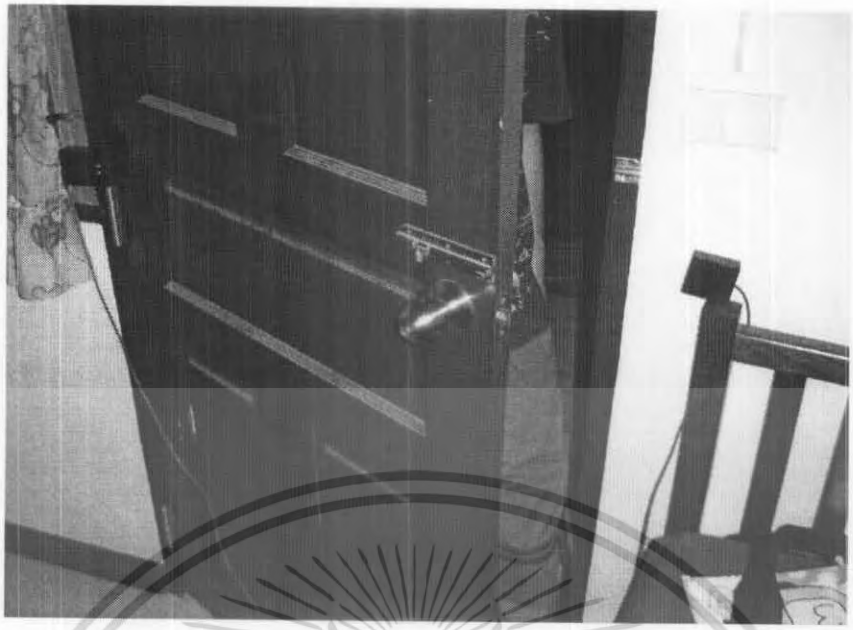
รูปที่ 4.20 ภาพรวมของโครงงาน

4.4.1 ผลการทดสองการทำงานของวงจรเซ็นเซอร์

ในการทดลองนี้จะเป็นการทดลองระบบรักษาความปลอดภัย โดยทำการจำลองว่ามีการบุกรุกเข้ามา โดยทำการติดเซ็นเซอร์ไว้ที่ประตูห้อง

รูปที่ 4.21 แสดงการติดตั้งเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



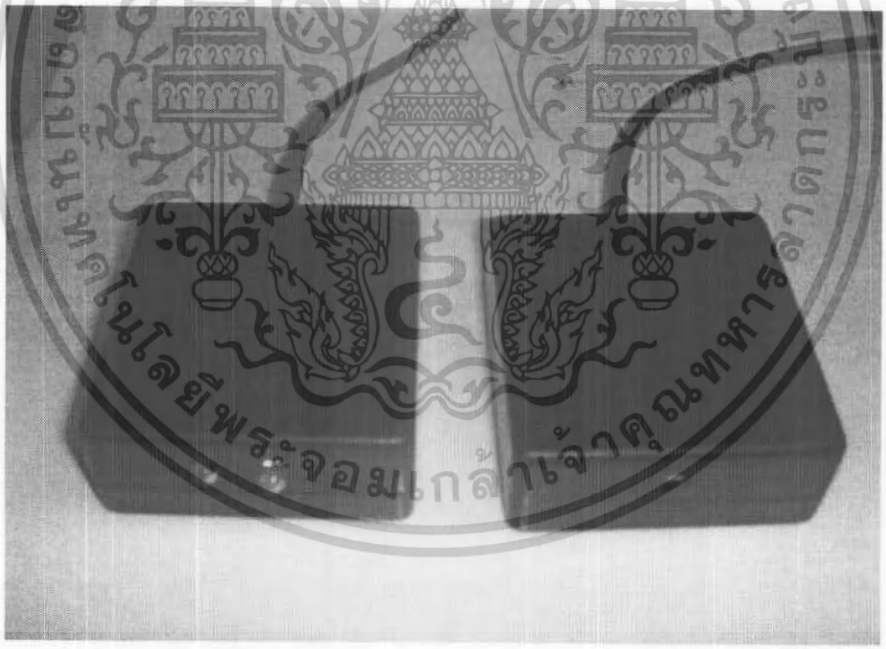
รูปที่ 4.22 แสดงเมื่อมีการบุกรุกเข้า

เมื่อมีการบุกรุกเข้ามาเซ็นเซอร์ก็จะตรวจจับได้โดยจะมี LED แสดงสถานะที่บอร์ด FPGA

รูปที่ 4.23 LED เมื่ออยู่ในสภาวะปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.24 เซ็นเซอร์ตรวจจับการบุกรุกได้



รูปที่ 4.25 วงจรเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2 ผลการทดสอบการทำงานของวงจรไซเรน

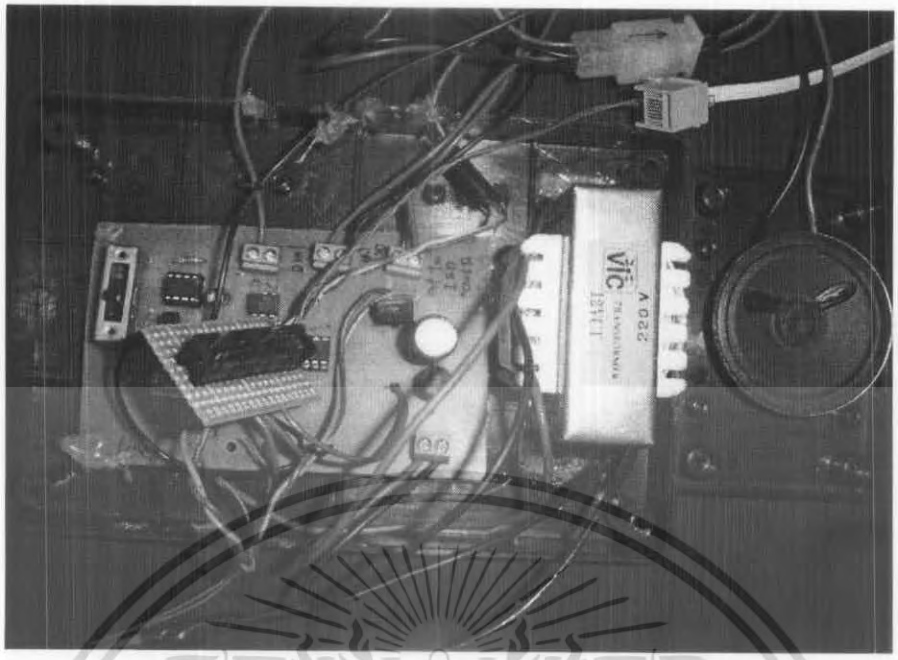
เมื่อเซ็นเซอร์ตรวจจับการบุกรุกได้ก็จะส่งอินพุตไปให้ FPGA และ FPGA ก็จะทำการสั่งงานให้วงจรไซเรนทำการเตือนภัย

รูปที่ 4.26 วงจรไซเรน

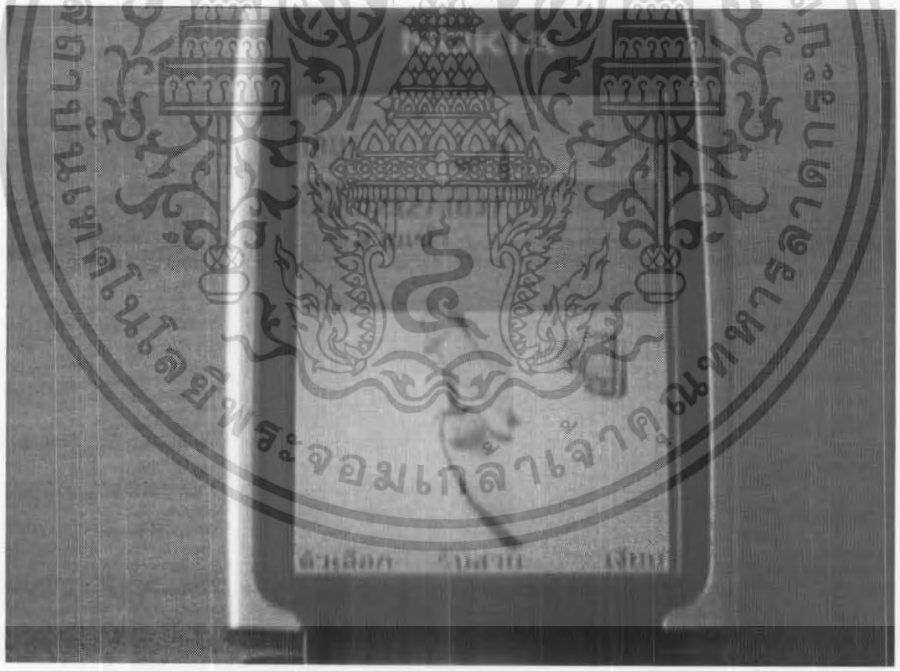
4.4.3 ผลการทดสอบการทำงานของวงจรแจ้งเตือนทางโทรศัพท์

เมื่อเซ็นเซอร์ตรวจจับการบุกรุกได้ FPGA ก็จะทำการสั่งงานให้ต่อสัญญาณโทรศัพท์ แล้วหลังจากนั้น ก็จะทำการกดเบอร์โทรศัพท์ไปยังเบอร์ที่ได้กำหนดไว้ จนเมื่อมีการรับหูโทรศัพท์ ก็จะทำการเล่นเสียงเตือนภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.27 วงจรแจ้งเตือนทางโทรศัพท์



รูปที่ 4.28 เมื่อมีการโทรศัพท์เข้ามาจากวงจรแจ้งเตือนทางโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและแนวทางการพัฒนาต่อไป

5.1 สรุปผลโครงงาน

จากการทดลองวงจรที่ได้ออกแบบไว้นั้น สามารถทำงานได้จริง และสามารถนำไปใช้ในการติดตั้งตามบ้านที่อยู่อาศัย หรือสถานที่ ต่างๆ ที่ต้องการการรักษาความปลอดภัยได้ โดยเมื่อเซ็นเซอร์ตรวจจับการบุกรุกได้ ก็จะสามารถที่จะแจ้งเตือนผ่านสัญญาณเสียงไซเรน และทำการแจ้งเตือนผ่านทางโทรศัพท์ได้ตามเลขหมายที่ได้ทำการตั้งไว้ เช่น แจ้งเตือนเข้าโทรศัพท์มือถือ หรือว่าอาจจะให้แจ้งเตือนไปยังสถานีตำรวจได้ โดยโครงงานนี้ได้เลือกใช้บอร์ด FPGA ซึ่งมีข้อดีคือโปรแกรมที่ใช้ร่วมกับชิปนั้นสามารถเรียนรู้ได้ไม่ยาก หรือ ซับซ้อนจนเกินไป

5.2 ปัญหาในการทำโครงงาน

5.2.1 ปัญหาด้านอุปกรณ์

- บอร์ด FPGA มีปัญหาจากคลื่นรบกวนภายนอก
- วงจรตรวจจับสัญญาณโทรศัพท์ในบางครั้งไม่สามารถใช้งานได้ จึงทำให้ไม่สามารถโทรศัพท์ไปแจ้งเตือนได้ จึงได้ใช้การโปรแกรมใน FPGA มาช่วยในการทำงานแทน
- เซ็นเซอร์ที่ใช้ในโครงงานมีความไวมากเกินไป
- เซ็นเซอร์ที่ใช้ยังมีราคาแพงอยู่ทำให้ในการนำไปใช้งานจริงจะติดตั้งได้เฉพาะส่วน

5.3 แนวทางในการพัฒนาต่อไป

5.3.1 การนำไปประยุกต์เข้ากับกล้องและระบบคอมพิวเตอร์เพื่อการแสดงภาพและการบันทึกเหตุการณ์

5.3.2 ปรับปรุงให้ เซ็นเซอร์มีหลายแบบเพื่อเพิ่มประสิทธิภาพให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. อภิชาต คงสุวรรณ และวัชริน นาชิน. 2542. ระบบรักษาความปลอดภัยในหมู่บ้านโดยผ่านระบบเครือข่ายคอมพิวเตอร์. กรุงเทพฯ : สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
2. อำนาจ บูรณ์เจริญ และนวชาติ จิลลานนท์. 2544. ระบบเตือนภัยครอบคลุมด้วย FPGA. กรุงเทพฯ : มหาวิทยาลัยเทคโนโลยีมหานคร.
3. นันธิดา รัชตเวชกุล และลักขณา ผิวขาว. 2548. การศึกษาการใช้ FPGA เพื่อออกแบบระบบป้องกันขโมย. ขอนแก่น : มหาวิทยาลัยขอนแก่น.
4. ชำนาญ ปัญญาใส และวัชรากร หนูทอง. 2547. ภาษา VHDL สำหรับการออกแบบวงจรดิจิตอล. พิมพ์ครั้งที่ 2. กรุงเทพฯ : บริษัท ซีเอ็ด ยูเคชั่น (มหาชน).

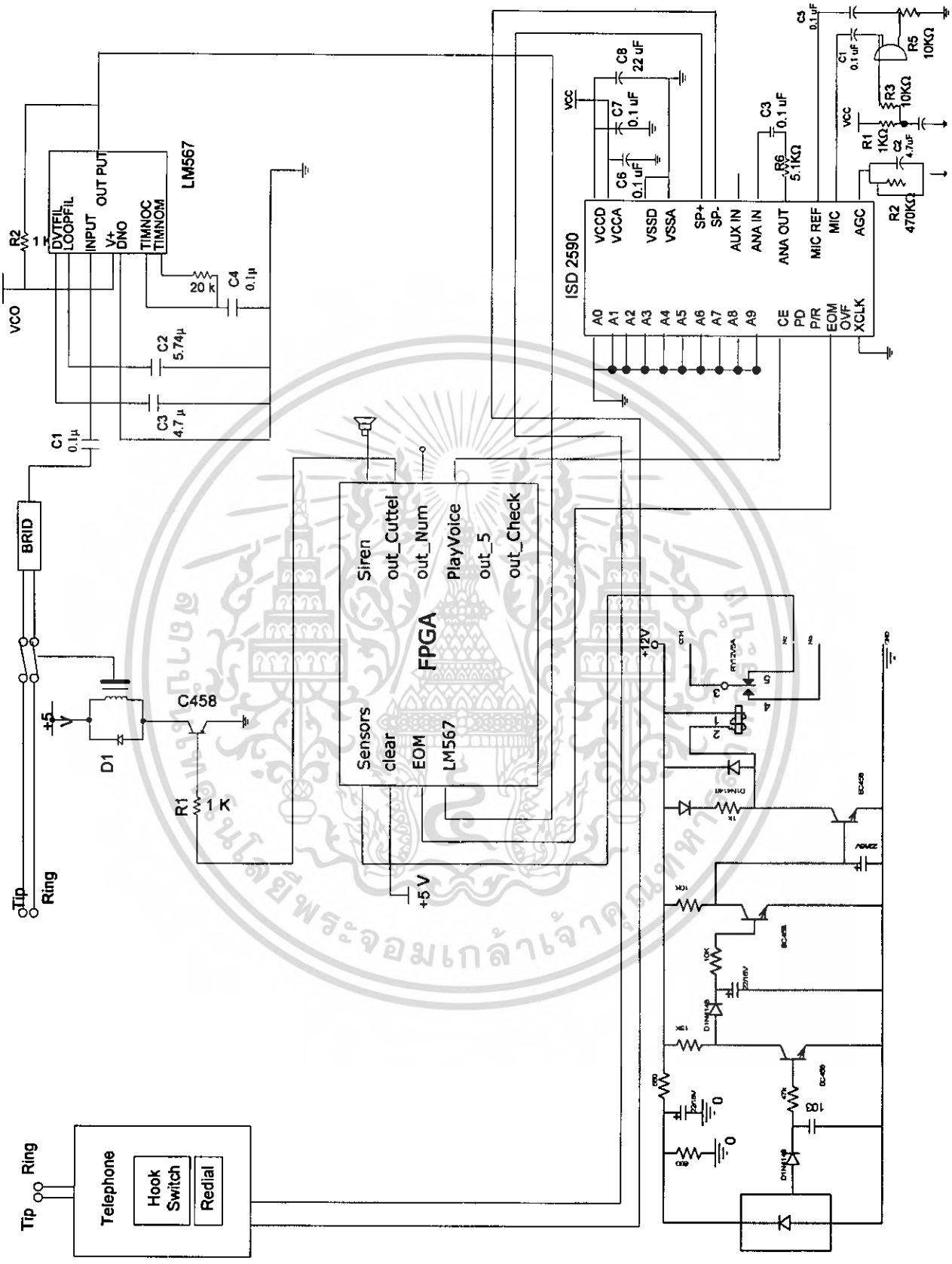
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วงจรอิเล็กทรอนิกส์

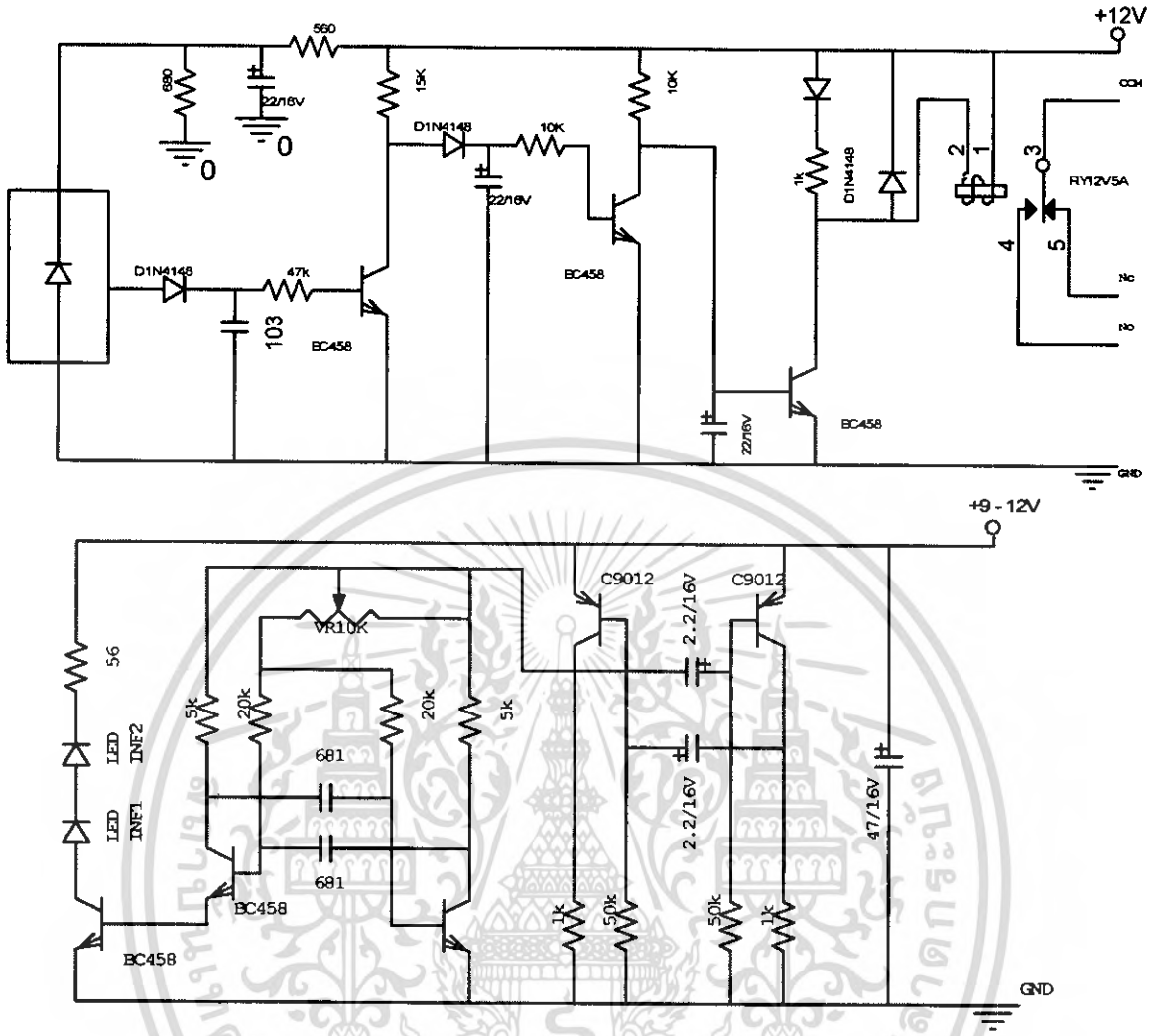
1. วงจรรวมของโครงงาน
2. วงจรเซ็นเซอร์
3. วงจรไซเรน
4. วงจรตัดต่อคู่สายโทรศัพท์
5. วงจรตรวจจับสัญญาณโทรศัพท์
6. วงจรบันทึกเสียงและเล่นกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



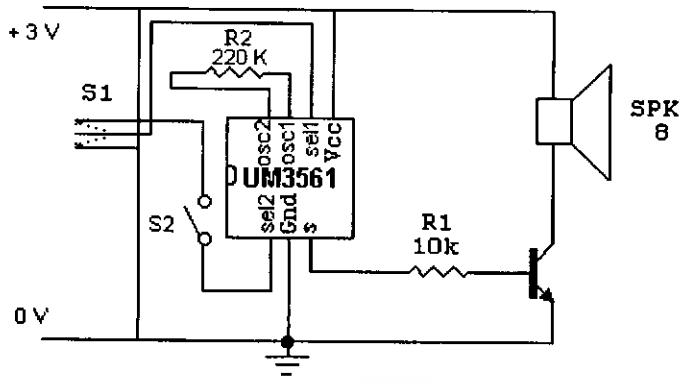
รูปที่ ก.1 วงจรรวมของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

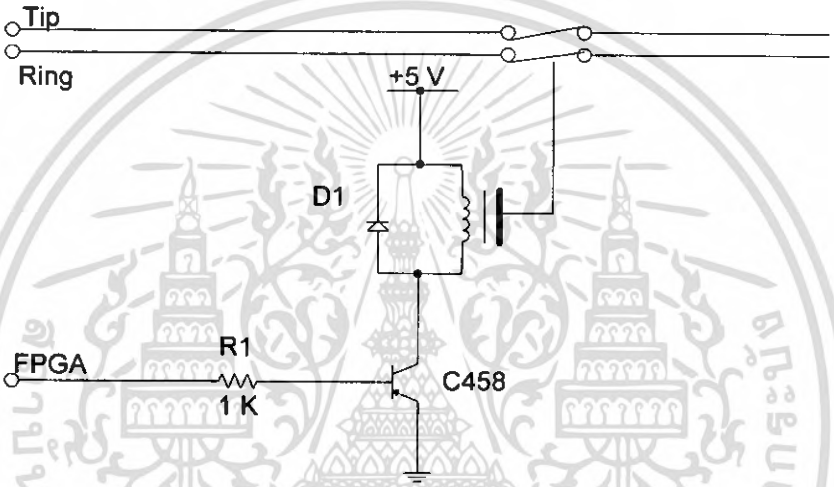


รูปที่ ก.2 วงจรเซ็นเซอร์

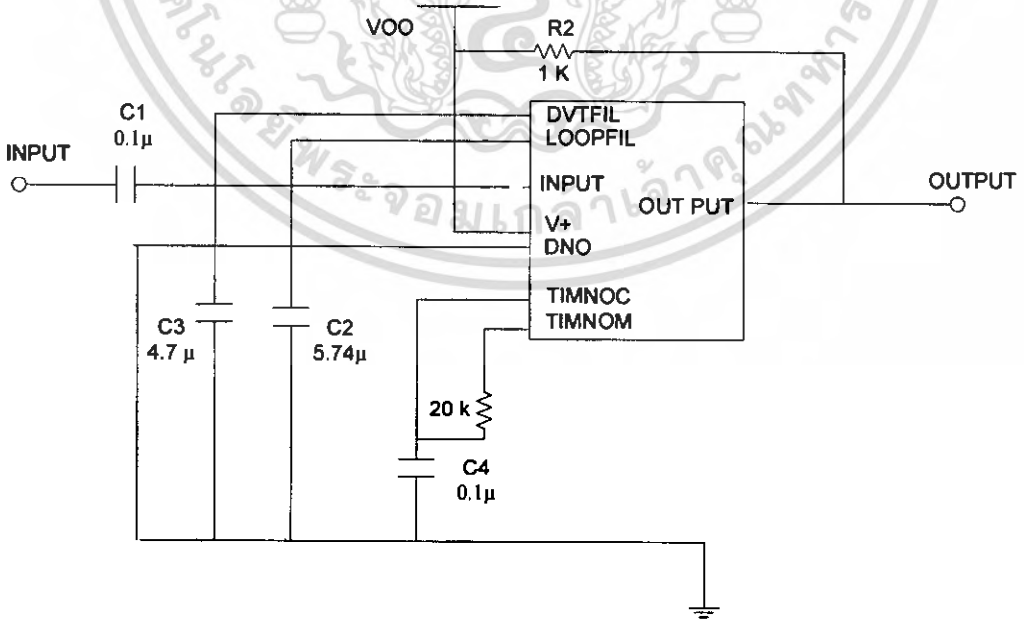
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.3 วงจรไซเรน

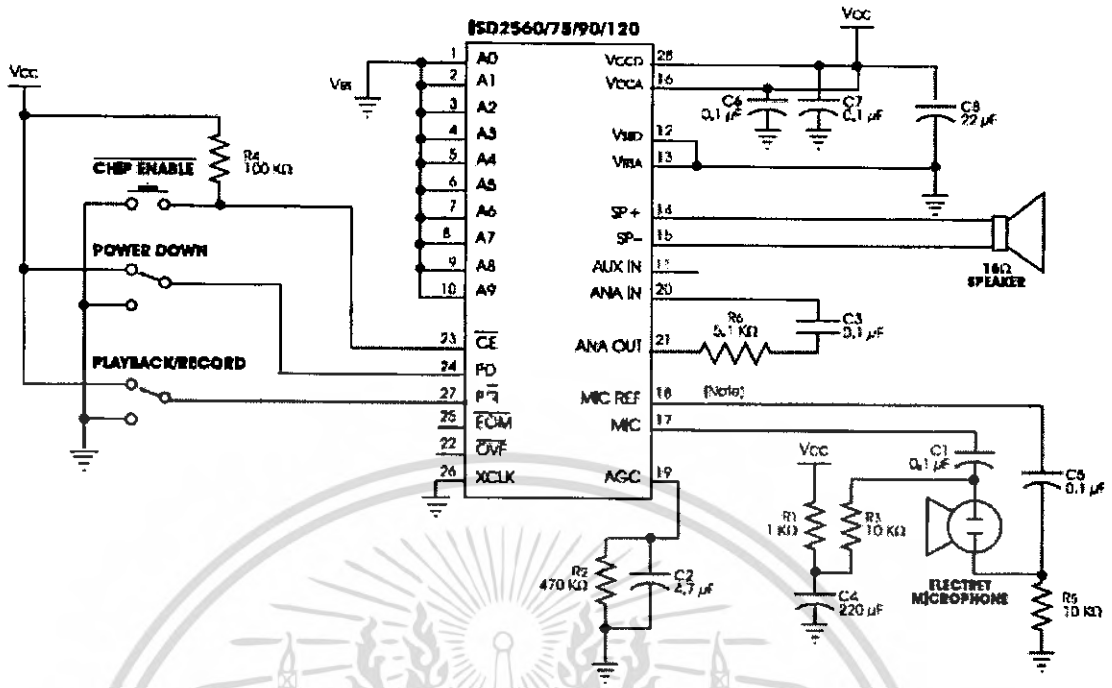


รูปที่ ก.4 วงจรตัดต่อคู่สายโทรศัพท์



รูปที่ ก.5 วงจรตรวจจับสัญญาณโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.6 วงจรบันทึกเสียงและเล่นกลับ

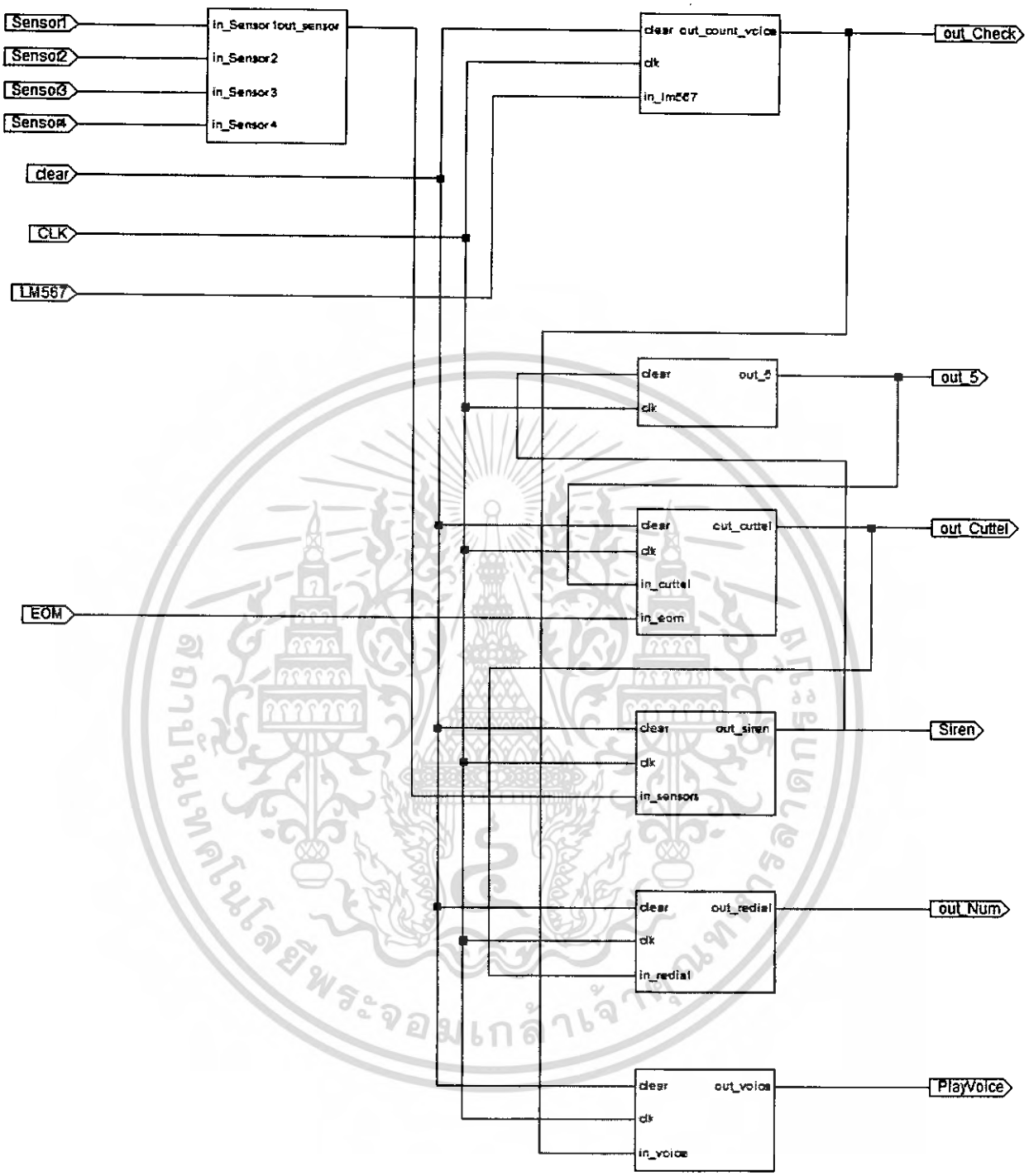
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

โปรแกรมของโครงงาน

1. ภาพรวมของวงจรที่โปรแกรมลง FPGA
2. โปรแกรมตรวจจับการเคลื่อนไหว
3. โปรแกรม Count5
4. โปรแกรมตัดต่อคู่สายโทรศัพท์
5. โปรแกรมกดหมายเลขโทรศัพท์
6. โปรแกรมตรวจจับสัญญาณโทรศัพท์
7. โปรแกรมเล่นเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.1 ภาพรวมของวงจรที่โปรแกรมลง FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมตรวจจับการเคลื่อนไหว

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Detect is
   Port ( clk    : in  STD_LOGIC;
          in_sensors : in  STD_LOGIC;
          clear  : in  STD_LOGIC;
          out_siren : out  STD_LOGIC);
end Detect;

architecture Program of Detect is
type state_type is (s0,s1);
signal ss : state_type;
begin
process(clk,in_sensors,clear)
        begin
                if(clear='1')then
                        out_siren<='1';
                        elsif (clk'event and clk = '1')then
                        case ss is
                                when s0=>
                                        if(in_sensors='1')then
                                                out_siren<='0';
                                                        ss<=s1;
                                                else
                                                        ss<=s0;
                                        end if;
                                when  s1=>
                                                ss<=s0;
                        end case;
                end if;
        end process;
end Program;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Count5

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity count_05a is
   Port ( clk   : in  STD_LOGIC;
          clear  : in  STD_LOGIC;
          out_5 : out  STD_LOGIC);
end count_05a;

architecture Behavioral of count_05a is
type state_type is(s0,s1);
signal ss : state_type;
signal count_5 : std_logic_vector(2 downto 0);
begin
        process(clk,clear)
                begin
                        if(clear='1')then
                                count_5<="000";
                                out_5<='0';
                                elsif(clk'event and clk='1')then
                                case ss is
                                        when s0=>
                                                if(count_5="101")then
                                                        count_5<="000";
                                                        out_5<='1';
                                                        ss<=s1;
                                                        else
                                                                count_5<=count_5+1;
                                                                ss<=s0;
                                                end if;
                                        when s1=>
                                                out_5<='1';
                                                        ss<=s1;
                                end case;
                        end if;
        end process;
end Behavioral;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมตัดต่อคู่สายโทรศัพท์

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity cuttel is
   Port ( in_cuttel : in  STD_LOGIC;
          in_eom    : in  STD_LOGIC;
          clear     : in  STD_LOGIC;
          clk       : in  STD_LOGIC;
          out_cuttel : out  STD_LOGIC);
end cuttel;

architecture relay of cuttel is

type state_type is(s0,s1,s2);
signal ss:state_type;

begin
process(clk,clear,in_cuttel,in_eom)
        begin
                if (clear='1')then
                        out_cuttel <= '0';
                                elsif(clk'event and clk='1')then
                        case ss is
                                when s0=>
                                        out_cuttel<='0';
                                if in_cuttel='1' then
                                        out_cuttel<='1';
                                        ss<=s1;
                                        else
                                                ss<=s0;
                                end if;
                                when s1=>
                                        if in_eom='0' then
                                                out_cuttel<='0';
                                                ss<=s2;
                                                else
                                                        ss<=s1;
                                                end if;
                                                        when s2=>
                                                                out_cuttel<='0';
                                                                ss<=s1;
                                                        end case;
                                        end if;
                                end process;
end relay;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมกดหมายเลขโทรศัพท์

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Redial is
   Port ( in_redial : in  STD_LOGIC;
          clear : in  STD_LOGIC;
          clk : in  STD_LOGIC;
          out_redial : out  STD_LOGIC);
end Redial;

architecture relay of Redial is

type state_type is (s0,s1,s2);
signal ss : state_type;

begin
process(clk,in_redial,clear)
        begin
                if(clear='1')then
                        out_redial<='0';
                        elsif(clk'event and clk='1')then
                        case ss is
                                when s0=>
                                        out_redial<='0';
                                        if in_redial='1' then
                                                out_redial<='1';
                                                ss<=s1;
                                                else
                                                        ss<=s0;
                                        end if;
                                                when s1=>
                                                        out_redial<='0';
                                                        ss<=s1;
                                                when s2=>
                                                        out_redial<='0';
                                                        ss<=s1;
                        end case;
                end if;
end process;
end relay;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมตรวจจับสัญญาณโทรศัพท์

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity count_voice is
   Port ( in_lm567     : in  STD_LOGIC;
          clear        : in  STD_LOGIC;
          clk          : in  STD_LOGIC;
          out_count_voice : out  STD_LOGIC);
end count_voice;

architecture Behavioral of count_voice is

type state_type is(s0,s1,s2);
signal ss : state_type;
signal count_5 : std_logic_vector(2 downto 0);

begin
process(clk,clear,in_lm567)
        begin
                if(clear='1')then
                        count_5<="000";
                        out_count_voice<='0';
                        elsif(clk'event and clk='1')then
                        case ss is
                                when s0=>
                                        if (in_lm567='1')then
                                                ss<=s1;
                                        else
                                                ss<=s0;
                                        end if;
                                when s1=>
                                        if(count_5="101")then
                                                count_5<="000";
                                                out_count_voice<='1';
                                                ss<=s2;
                                        else
                                                count_5<=count_5+1;
                                                ss<=s0;
                                        end if;
                                when s2=>
                                        out_count_voice<='1';
                                        ss <=s0;
                        end case;
                end if;
end process;
end Behavioral;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมเล่นเสียง

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Voice is
   Port ( in_voice          : in  STD_LOGIC;
          clear             : in  STD_LOGIC;
          clk               : in  STD_LOGIC;
          out_voice : out  STD_LOGIC);
end Voice;

architecture tell of Voice is

type state_type is(s0,s1,s2);
signal ss : state_type;

begin
process(clk,in_voice,clear)
          begin
                    if(clear='1')then
                              out_voice<='1';
                              elsif (clk'event and clk='1')then
                              case ss is
                                        when s0=>
                                                  out_voice<='1';
                                                  if in_voice='1' then
                                                            out_voice<='0';
                                                            ss<=s1;
                                                            else
                                                                      ss<=s0;
                                                  end if;
                                        when s1=>
                                                  out_voice<='1';
                                                  ss<=s2;
                                        when s2=>
                                                  out_voice<='1';
                                                  ss<=s1;
                              end case;
                    end if;
end process;
end tell;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Security

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Security is
Port ( Sensor1 : in  STD_LOGIC;
               Sensor2       : in  STD_LOGIC;
               Sensor3       : in  STD_LOGIC;
               Sensor4       : in  STD_LOGIC;
               EOM           : in  STD_LOGIC;
               LM567         : in  STD_LOGIC;
               CLK           : in  STD_LOGIC;
               clear         : in  STD_LOGIC;
               Siren         : out  STD_LOGIC;
               out_Cuttel    : out  STD_LOGIC;
               out_Num       : out  STD_LOGIC;
               PlayVoice     : out  STD_LOGIC;
               out_Check     : out  STD_LOGIC;
               out_5         : out  STD_LOGIC
        );
end Security;

architecture Behavioral of Security is

signal infrared              : STD_LOGIC;
signal sensors_to_count_05 : STD_LOGIC;
signal count_05_to_cuttel    : STD_LOGIC;
signal cuttel_to_redial      : STD_LOGIC;
signal count_voice_to_voice : STD_LOGIC;

component Sensor is
Port    (in_Sensor1 : in  STD_LOGIC;
        in_Sensor2 : in  STD_LOGIC;
        in_Sensor3 : in  STD_LOGIC;
        in_Sensor4 : in  STD_LOGIC;
        out_sensor : out  STD_LOGIC);
end component;

component Detect is
Port ( clk : in  STD_LOGIC;
        in_sensors : in  STD_LOGIC;
        clear : in  STD_LOGIC;
        out_siren : out  STD_LOGIC);
end component;

component Count_05a is
Port    ( clk : in  STD_LOGIC;
        clear : in  STD_LOGIC;
        out_5 : out  STD_LOGIC);
end component;

component cuttel is
Port    ( in_cuttel : in  STD_LOGIC;
        in_eom  : in  STD_LOGIC;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        clear   : in  STD_LOGIC;
        clk     : in  STD_LOGIC;
        out_cuttel : out  STD_LOGIC);
end component;

component Redial is
Port ( in_redial : in  STD_LOGIC;
        clear   : in  STD_LOGIC;
        clk     : in  STD_LOGIC;
        out_redial : out  STD_LOGIC);
end component;

component count_voice is
Port ( in_lm567  : in  STD_LOGIC;
        clear   : in  STD_LOGIC;
        clk     : in  STD_LOGIC;
        out_count_voice : out  STD_LOGIC);
end component;

component Voice is
Port( in_voice : in  STD_LOGIC;
        clear   : in  STD_LOGIC;
        clk     : in  STD_LOGIC;
        out_voice : out  STD_LOGIC);
end component;

begin

Siren       <= sensors_to_count_05;
out_Cuttel  <= cuttel_to_redial;
out_Check   <= count_voice_to_voice;
out_5       <= count_05_to_cuttel;

block_Sensor : Sensor
port map(in_Sensor1 => Sensor1,
        in_Sensor2  => Sensor2,
        in_Sensor3  => Sensor3,
        in_Sensor4  => Sensor4,
        out_sensor => infrared
                    );

block_Detect : Detect
port map(clk  =>    clk,
        in_sensors  =>    infrared,
        clear       =>    clear,
        out_siren   =>    sensors_to_count_05
                    );

block_Count_05 : Count_05a
port map(clk  =>    clk,
            clear   =>    sensors_to_count_05,
            out_5   =>    count_05_to_cuttel
                    );

block_cuttel : cuttel
port map(clk  =>    clk,
            clear   =>    clear,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                in_eom          =>      EOM,
                in_cuttel       =>      count_05_to_cuttel,
                out_cuttel      =>      cuttel_to_redial
                        );

block_Redial : Redial
port map(clk    =>      clk,
                clear           =>      clear,
                in_redial       =>      cuttel_to_redial,
                out_redial      =>      out_Num
                        );

block_count_voice : count_voice
port map(clk    =>      clk,
                clear           =>      clear,
                in_lm567        =>      LM567,
                out_count_voice         => count_voice_to_voice
                        );

block_Voice : Voice
port map(clk    =>      clk,
                clear           =>      clear,
                in_voice        =>      count_voice_to_voice,
                out_voice       =>      PlayVoice
                        );

end Behavioral;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sensors

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Sensor is
   Port ( --clk   : in  STD_LOGIC;
          in_Sensor1 : in  STD_LOGIC;
          in_Sensor2 : in  STD_LOGIC;
          in_Sensor3 : in  STD_LOGIC;
          in_Sensor4 : in  STD_LOGIC;
          out_sensor : out  STD_LOGIC);
end Sensor;

architecture Behavioral of Sensor is
begin
   Process(in_Sensor1,in_Sensor2,in_Sensor3,in_Sensor4)
          begin

                    if   in_Sensor1 = '1' or in_Sensor2 = '1'
                           or in_Sensor3 = '1' or in_Sensor4 = '1'
                         then
                           out_sensor <= '1';
                         Else
                           out_sensor <= '0';
                         end if;
   End Process;
end Behavioral;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

รายละเอียดเฉพาะของอุปกรณ์

1. **LM567 Tone Decoder**
2. **ISD2590 Single-Chip Voice Record/Playback Device**
3. **UM3561**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

May 1999

LM567/LM567C Tone Decoder

General Description

The LM567 and LM567C are general purpose tone decoders designed to provide a saturated transistor switch to ground when an input signal is present within the passband. The circuit consists of an I and Q detector driven by a voltage controlled oscillator which determines the center frequency of the decoder. External components are used to independently set center frequency, bandwidth and output delay.

Features

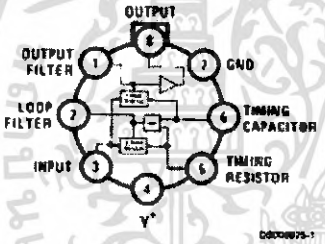
- 20 to 1 frequency range with an external resistor
- Logic compatible output with 100 mA current sinking capability
- Bandwidth adjustable from 0 to 14%
- High rejection of out of band signals and noise
- Immunity to false signals
- Highly stable center frequency
- Center frequency adjustable from 0.01 Hz to 500 kHz

Applications

- Touch tone decoding
- Precision oscillator
- Frequency monitoring and control
- Wide band FSK demodulation
- Ultrasonic controls
- Carrier current remote controls
- Communications paging decoders

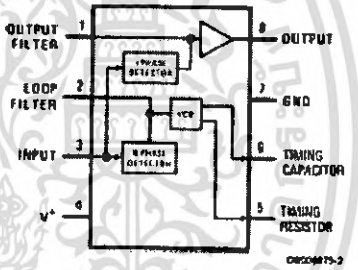
Connection Diagrams

Metal Can Package



Top View
Order Number LM567H or LM567CH
See NS Package Number H08C

Dual-In-Line and Small Outline Packages



Top View
Order Number LM567CM
See NS Package Number M08A
Order Number LM567CN
See NS Package Number N08E

© 1999 National Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Note 1)

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

Supply Voltage Pin	9V
Power Dissipation (Note 2)	1100 mW
V_8	15V
V_3	−10V
V_3	V_8 + 0.5V
Storage Temperature Range	−65°C to +150°C
Operating Temperature Range	
LM567H	−55°C to +125°C
LM567CH, LM567CM, LM567CN	0°C to +70°C
Soldering Information	
Dual-In-Line Package	
Soldering (10 sec.)	260°C
Small Outline Package	
Vapor Phase (60 sec.)	215°C
Infrared (15 sec.)	220°C

See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.

Electrical Characteristics

AC Test Circuit, T_A = 25°C, V⁺ = 5V

Parameters	Conditions	LM567			LM567C/LM567CM			Units
		Min	Typ	Max	Min	Typ	Max	
Power Supply Voltage Range		4.75	5.0	9.0	4.75	5.0	9.0	V
Power Supply Current Quiescent	R_L = 20k		6	8		7	10	mA
Power Supply Current Activated	R_L = 20k		11	13		12	15	mA
Input Resistance		18	20		15	20		kΩ
Smallest Detectable Input Voltage	I_L = 100 mA, $f_i = f_o$		20	25		20	25	mVrms
Largest No Output Input Voltage	I_O = 100 mA, $f_i = f_o$	10	15		10	15		mVrms
Largest Simultaneous Outband Signal to Inband Signal Ratio			6			6		dB
Minimum Input Signal to Wideband Noise Ratio	B_n = 140 kHz		−6			−6		dB
Largest Detection Bandwidth		12	14	16	10	14	18	% of f_o
Largest Detection Bandwidth Skew			1	2		2	3	% of f_o
Largest Detection Bandwidth Variation with Temperature			±0.1			±0.1		%/°C
Largest Detection Bandwidth Variation with Supply Voltage	4.75–6.75V		±1	±2		±1	±5	%V
Highest Center Frequency		100	500		100	500		kHz
Center Frequency Stability (4.75–5.75V)	$0 < T_A < 70$ $-55 < T_A < +125$		35 ± 60 35 ± 140			35 ± 60 35 ± 140		ppm/°C ppm/°C
Center Frequency Shift with Supply Voltage	4.75V–6.75V 4.75V–9V		0.5	1.0 2.0		0.4	2.0 2.0	%/V %/V
Fastest ON-OFF Cycling Rate			f_o/20			f_o/20		
Output Leakage Current	V_8 = 15V		0.01	25		0.01	25	µA
Output Saturation Voltage	e_i = 25 mV, I_8 = 30 mA e_i = 25 mV, I_8 = 100 mA		0.2 0.6	0.4 1.0		0.2 0.6	0.4 1.0	V
Output Fall Time			30			30		ns
Output Rise Time			150			150		ns

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. Operating Ratings indicate conditions for which the device is functional, but do not guarantee specific performance limits. Electrical Characteristics state DC and AC electrical specifications under particular test conditions which guarantee specific performance limits. This assumes that the device is within the Operating Ratings. Specifications are not guaranteed for parameters where no test is given, however, the typical value is a good indication of device performance.

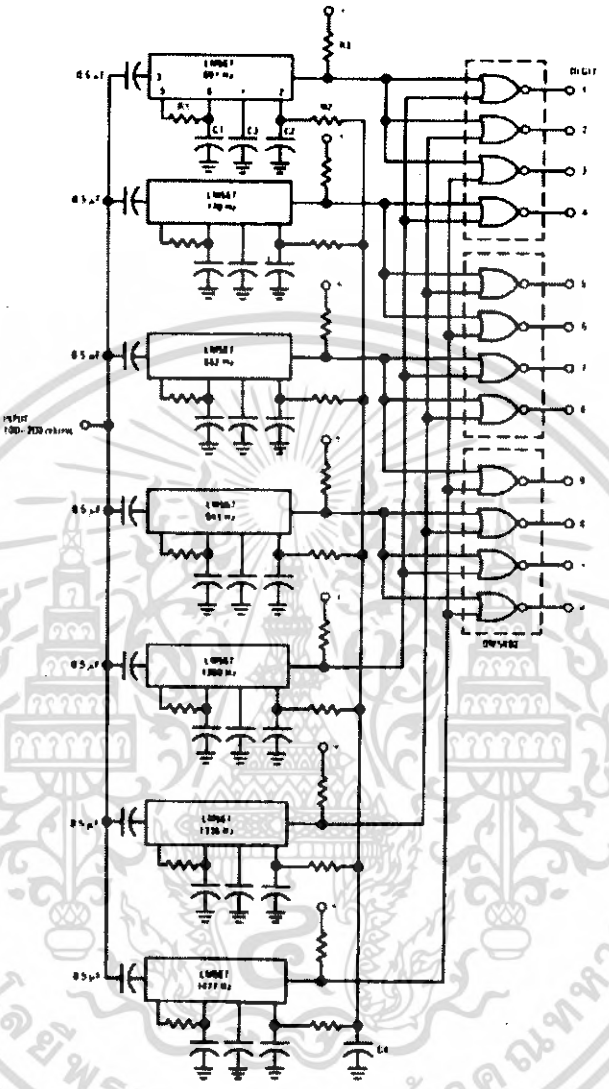
Note 2: The maximum junction temperature of the LM567 and LM567C is 150°C. For operating at elevated temperatures, devices in the TO-5 package must be derated based on a thermal resistance of 150°C/W, junction to ambient, or 45°C/W, junction to case. For the DIP the device must be derated based on a thermal resistance of 110°C/W, junction to ambient. For the Small Outline package, the device must be derated based on a thermal resistance of 160°C/W, junction to ambient.

Note 3: Refer to RETS567X drawing for specifications of military LM567H version

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Applications

Touch-Tone Decoder

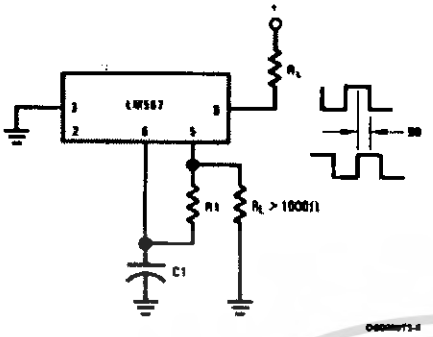


Component values (typ)
R1 6.8 to 15k
R2 4.7k
R3 20k
C1 0.10 mfd
C2 1.0 mfd 6V
C3 2.2 mfd 6V
C4 250 mfd 6V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

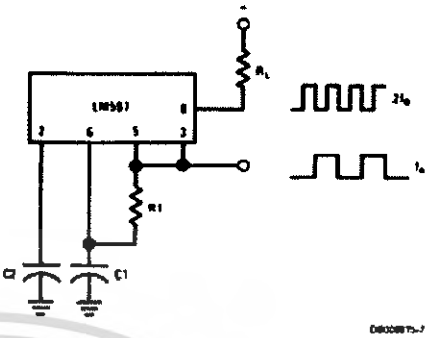
Typical Applications (Continued)

Oscillator with Quadrature Output

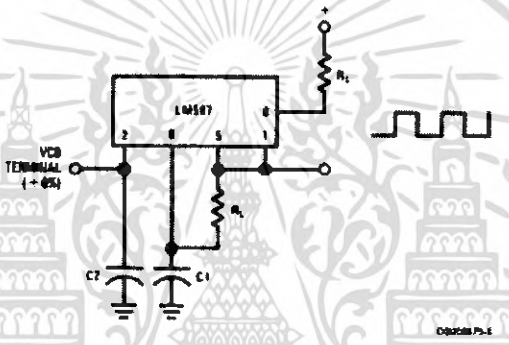


Connect Pin 3 to 2.8V to Invert Output

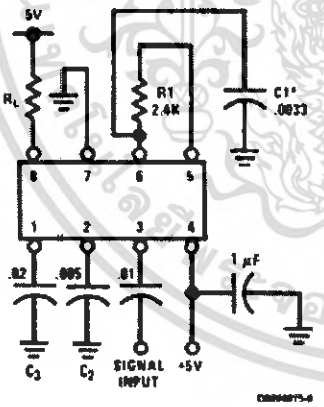
Oscillator with Double Frequency Output



Precision Oscillator Drive 100 mA Loads



AC Test Circuit



f_o = 100 kHz ± 5V

*Note: Adjust for f_o = 100 kHz

Applications Information

The center frequency of the tone decoder is equal to the free running frequency of the VCO. This is given by

$$f_o \cong \frac{1}{1.1\ R_1 C_1}$$

The bandwidth of the filter may be found from the approximation

$$BW = 1070 \sqrt{\frac{V_i}{f_o C_2}} \text{ in \% of } f_o$$

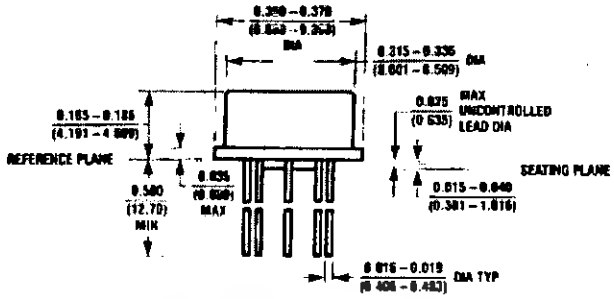
Where:

V_i = Input voltage (volts rms), $V_i \leq 200$ mV

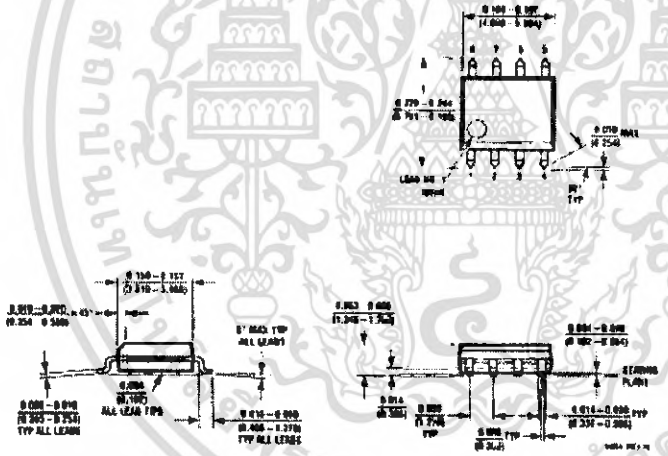
C_2 = Capacitance at Pin 2 (μF)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted



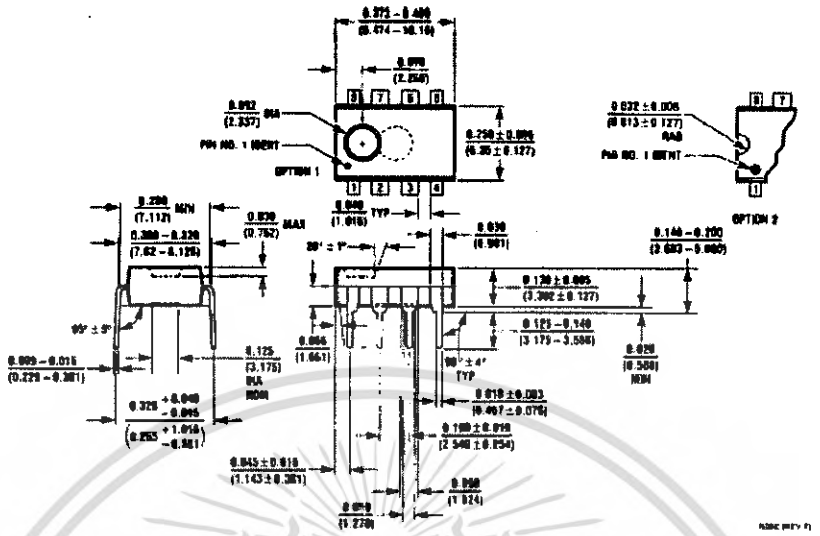
Metal Can Package (H)
Order Number LM567H or LM567CH
NS Package Number H08C



Small Outline Package (M)
Order Number LM567CM
NS Package Number M08A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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



Molded Dual-In-Line Package (N)
Order Number LM567CN
NS Package Number N08E

LIFE SUPPORT POLICY

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

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
Americas
Tel: 1-800-272-9959
Fax: 1-800-737-7018
Email: support@nsc.com

National Semiconductor Europe
Fax: +49 (0) 1 80-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 1 80-530 85 85
English Tel: +49 (0) 1 80-532 78 32
Français Tel: +49 (0) 1 80-532 93 58
Italiano Tel: +49 (0) 1 80-534 16 80

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

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

www.national.com

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ChipCorder®

ISD2560/75/90/120 Products

Single-Chip Voice Record/Playback Devices 60-, 75-, 90-, and 120-Second Durations

GENERAL DESCRIPTION

Information Storage Devices' ISD2500 ChipCorder® Series provides high-quality, single-chip Record/Playback solutions for 60- to 120-second messaging applications. The CMOS devices include an on-chip oscillator, microphone preamplifier, automatic gain control, antialiasing filter, smoothing filter, speaker amplifier, and high density multi-level storage array. In addition, the ISD2500 is microcontroller compatible, allowing complex messaging and addressing to be achieved.

Recordings are stored in on-chip nonvolatile memory cells, providing zero-power message storage. This unique, single-chip solution is made possible through ISD's patented multilevel storage technology. Voice and audio signals are stored directly into memory in their natural form, providing high-quality, solid-state voice reproduction.

FEATURES

- Easy-to-use single-chip voice Record/Playback solution
- High-quality, natural voice/audio reproduction
- Manual switch or microcontroller compatible Playback can be edge- or level-activated
- Single-chip durations of 60, 75, 90, and 120 seconds
- Directly cascadable for longer durations
- Automatic Power-Down (Push-Button Mode)
  - Standby current 1 μA (typical)
- Zero-power message storage
  - Eliminates battery backup circuits
- Fully addressable to handle multiple messages
- 100-year message retention (typical)

Table 2-76: ISD2560/75/90/120 Product Summary

Part Number	Duration (Seconds)	Input Sample Rate (KHz)	Typical Filter Pass Band (KHz)
ISD2560	60	8.0	3.4
ISD2575	75	6.4	2.7
ISD2590	90	5.3	2.3
ISD25120	120	4.0	1.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 100K record cycles (typical)
- On-chip clock source
- Programmer support for play-only applications
- Single +5 volt power supply
- Available in die form, DIP, SOIC, and TSOP packaging
- Industrial temperature (–40°C to +85°C) versions available

DETAILED DESCRIPTION

Speech/Sound Quality

The ISD2500 Series includes devices offered at 4.0, 5.3, 6.4, and 8.0 KHz sampling frequencies, allowing the user a choice of speech quality options. Increasing the duration within a product series decreases the sampling frequency and bandwidth, which affects sound quality. Please refer to the ISD2560/75/90/120 Product Summary table on the previous page to compare filter pass band and product durations.

The speech samples are stored directly into on-chip nonvolatile memory without the digitization and compression associated with other solutions. Direct analog storage provides a very true, natural sounding reproduction of voice, music, tones, and sound effects not available with most solid-state digital solutions.

Duration

To meet end system requirements, the ISD2500 Series offers single-chip solutions at 60, 75, 90, and 120 seconds. Parts may also be cascaded together for longer durations.

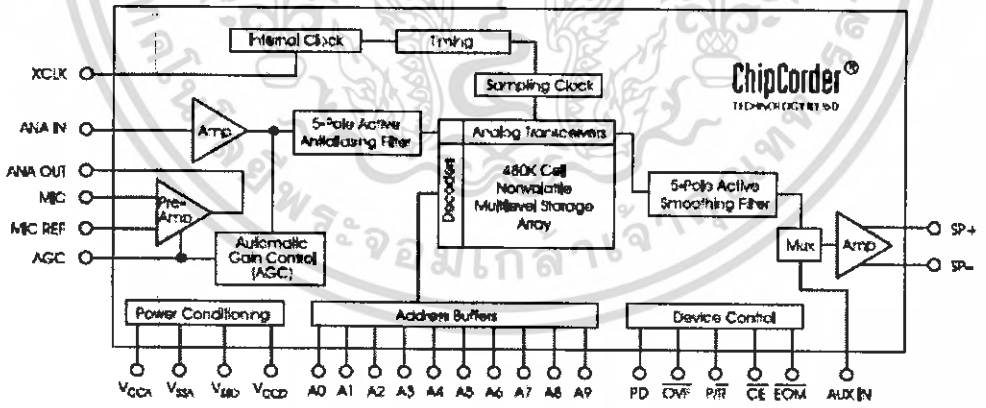
EEPROM Storage

One of the benefits of ISD's ChipCorder technology is the use of on-chip nonvolatile memory, providing zero-power message storage. The message is retained for up to 100 years typically without power. In addition, the device can be re-recorded typically over 100,000 times.

Microcontroller Interface

In addition to its simplicity and ease of use, the ISD2500 Series includes all the interfaces necessary for microcontroller-driven applications. The address and control lines can be interfaced to a microcontroller and manipulated to perform a variety of tasks, including message assembly, message concatenation, predefined fixed message segmentation, and message management.

Figure 2-33: ISD2560/75/90/120 Device Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming

The ISD2500 Series is also ideal for playback-only applications, where single or multiple messages are referenced through buttons, switches, or a microcontroller. Once the desired message configuration is created, duplicates can easily be generated via an ISD programmer.

PIN DESCRIPTIONS

Voltage Inputs (V_{CCA}, V_{CCD})

To minimize noise, the analog and digital circuits in the ISD2500 Series devices use separate power busses. These voltage busses are brought out to separate pins and should be tied together as close to the supply as possible. In addition, these supplies should be decoupled as close to the package as possible.

Ground Inputs (V_{SSA}, V_{SSD})

The ISD2500 Series of devices utilizes separate analog and digital ground busses. These pins should be connected separately through a low-impedance path to power supply ground. The backside of the die is connected to V_{SS} through the substrate resistance. In a chip-on-board design the die attach area must be connected to V_{SS} or left floating.

Power Down Input (PD)

When not recording or playing back, the PD pin should be pulled HIGH to place the part in a very low power mode (see I_{SB} specification). When $\overline{OVF}$ pulses LOW for an overflow condition, PD should be brought HIGH to reset the address pointer back to the beginning of the Record/Playback space. The PD pin has additional functionality in the M6 (Push-Button) Operational Mode described later in the Operational Mode section.

Chip Enable Input ($\overline{CE}$)

The $\overline{CE}$ pin is taken LOW to enable all Playback and Record operations. The address inputs and Playback/Record input (P/$\overline{R}$) are latched by the falling edge of $\overline{CE}$. $\overline{CE}$ has additional functionality in the M6 (Push-Button) Operational Mode described later in the Operational Mode section.

Playback/Record Input (P/$\overline{R}$)

The P/$\overline{R}$ input is latched by the falling edge of the $\overline{CE}$ pin. A HIGH level selects a Playback cycle while a LOW level selects a Record cycle. For a Record cycle, the address inputs provide the starting address and recording continues until PD or $\overline{CE}$ is pulled HIGH or an overflow is detected (i.e. the chip is full). When a Record cycle is terminated by pulling PD or $\overline{CE}$ HIGH, an End-Of-Message (EOM) marker is stored at the current address in memory. For a Playback cycle, the address inputs provide the starting address and the device will play until an EOM marker is encountered. The device can continue past an EOM marker in an operational mode, or if $\overline{CE}$ is held LOW in address mode. (See page 2-122 for more Operational Modes).

End-Of-Message / RUN Output (EOM)

A nonvolatile marker is automatically inserted at the end of each recorded message. It remains there until the message is recorded over. The EOM output pulses LOW for a period of T_{EOM} at the end of each message.

In addition, the ISD2500 Series has an internal V_{CC} detect circuit to maintain message integrity should V_{CC} fall below 3.5V. In this case, $\overline{EOM}$ goes LOW and the device is fixed in Playback-only mode.

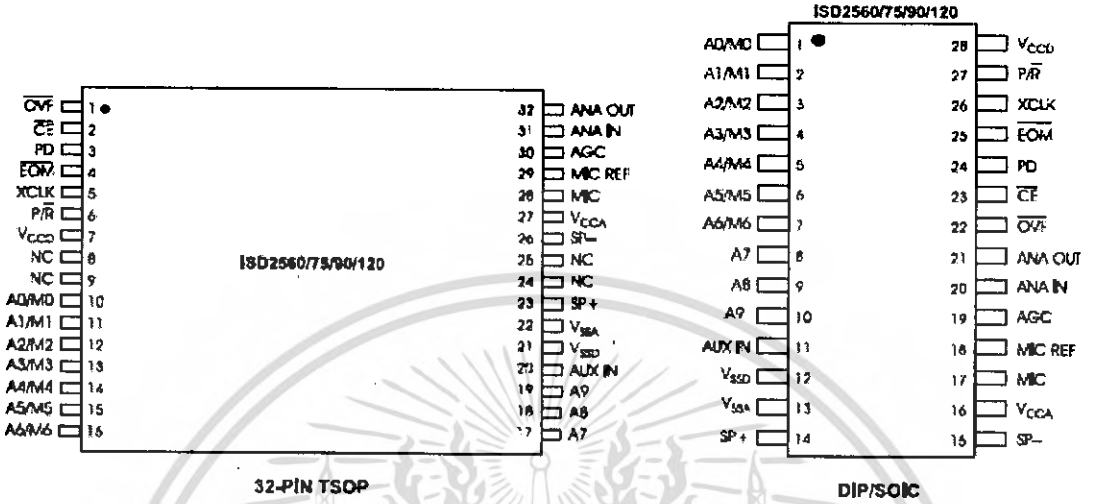
When the device is configured in Operational Mode M6 (Push-Button Mode), this pin provides an active-HIGH RUN signal, indicating the device is currently recording or playing. This signal can conveniently drive an LED for a visual indicator of a Record or Playback operation in process.

Overflow Output ($\overline{OVF}$)

This signal pulses LOW at the end of memory space, indicating the device has been filled and the message has overflowed. The $\overline{OVF}$ output then follows the $\overline{CE}$ input until a PD pulse has reset the device. This pin can be used to cascode several ISD2500 devices together to increase Record/Playback durations.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 2-34: ISD2560/75/90/120 Device Pinouts



Microphone Input (MIC)

The microphone input transfers its signal to the on-chip preamplifier. An on-chip Automatic Gain Control (AGC) circuit controls the gain of this preamplifier from –15 to 24 dB. An external microphone should be AC coupled to this pin via a series capacitor. The capacitor value, together with the internal 10 K ohm resistance on this pin, determines the low-frequency cutoff for the ISD2500 Series passband. See Chapter 5, Application Information for additional information on low-frequency cutoff calculation.

Microphone Reference Input (MIC REF)

The MIC REF input is the inverting input to the microphone preamplifier. This provides a noise-canceling or common-mode rejection input to the device when connected to a differential microphone.

Automatic Gain Control Input (AGC)

The AGC dynamically adjusts the gain of the preamplifier to compensate for the wide range of microphone input levels. The AGC allows the full range of whispers to loud sounds to be recorded with minimal distortion. The "attack" time is determined by the time constant of a 5 KΩ internal resistance and an external capacitor (C2 on the schematic on page 2-135) connected from the AGC pin to V_{SSA} analog ground. The "release" time is determined by the time constant of an external resistor (R2) and an external capacitor (C2) connected in parallel between the AGC Pin and V_{SSA} analog ground. Nominal values of 470 KΩ and 4.7 μF give satisfactory results in most cases.

Analog Output (ANA OUT)

This pin provides the preamplifier output to the user. The voltage gain of the preamplifier is determined by the voltage level at the AGC pin.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analog Input (ANA IN)

The analog input pin transfers its signal to the chip for recording. For microphone inputs, the ANA OUT pin should be connected via an external capacitor to the ANA IN pin. This capacitor value, together with the 3.0 KΩ input impedance of ANA IN, is selected to give additional cutoff at the low-frequency end of the voice passband. If the desired input is derived from a source other than a microphone, the signal can be fed, capacitively coupled, into the ANA IN pin directly.

External Clock Input (XCLK)

The external clock input for the ISD2500 devices has an internal pull-down device. These devices are configured at the factory with an internal sampling clock frequency centered to ±1% of specification. The frequency is then maintained to a variation of ±2.25% over the entire commercial temperature and operating voltage ranges. The internal clock has a ±5% tolerance over the industrial temperature and voltage range. A regulated power supply is recommended for industrial temperature range parts. If greater precision is required, the device can be clocked through the XCLK pin as follows:

Table 2-77: External Clock Sample Rates

Part Number	Sample Rate	Required Clock
ISD2560	8.0 KHz	1024 KHz
ISD2575	6.4 KHz	819.2 KHz
ISD2590	5.3 KHz	682.7 KHz
ISD25120	4.0 KHz	512 KHz

These recommended clock rates should not be varied because the antialiasing and smoothing filters are fixed, and aliasing problems can occur if the sample rate differs from the one recommended. The duty cycle on the input clock is not critical, as the clock is immediately divided by two. ***If the XCLK is not used, this input must be connected to ground.***

Speaker Outputs (SP+/SP–)

All devices in the ISD2500 Series include an on-chip differential speaker driver, capable of driving 50 milliwatts into 16 Ω from AUX IN (12.2 mW from memory).

The speaker outputs are held at V_{SSA} levels during record and power down. It is therefore not possible to parallel speaker outputs of multiple ISD2500 devices or the outputs of other speaker drivers.

NOTE Connection of speaker outputs in parallel may cause damage to the device.

A single output may be used alone (including a coupling capacitor between the SP pin and the speaker). These outputs may be used individually with the output signal taken from either pin. Using the differential outputs results in a 4:1 improvement in output power.

NOTE Never ground or drive an unused speaker output.

Auxiliary Input (AUX IN)

The Auxiliary input is multiplexed through to the output amplifier and speaker output pins when $\overline{CE}$ is HIGH, P/$\overline{R}$ is HIGH, and Playback is currently not active or if the device is in Playback overflow. When cascading multiple ISD2500 devices, the AUX IN pin is used to connect a Playback signal from a following device to the previous output speaker drivers. For noise considerations, it is suggested that the auxiliary input not be driven when the storage array is active.

Address/Mode Inputs (Ax/Mx)

The Address/Mode inputs have two functions depending on the level of the two Most Significant Bits (MSB) of the address (A8 and A9).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 2-78: Operational Modes Table

Mode Control	Function	Typical Use	Jointly Compatible[1]
M0	Message cueing	Fast-forward through messages	M4, M5, M6
M1	Delete EOM markers	Position EOM marker at the end of the last message	M3, M4, M5, M6
M2	Not applicable	Reserved	N/A
M3	Looping	Continuous playback from Address 0	M1, M5, M6
M4	Consecutive addressing	Record/Play multiple consecutive messages	M0, M1, M5
M5	$\overline{CE}$ level-activated	Allows message pausing	M0, M1, M3, M4
M6	Push-button control	Simplified device interface	M0, M1, M3

1. Indicates additional operational modes which can be used simultaneously with the given mode.

If either or both of the two MSBs are LOW, the inputs are all interpreted as address bits and are used as the start address for the current Record or Playback cycle. The address pins are inputs only and do not output internal address information as the operation progresses. Address inputs are latched by the falling edge of $\overline{CE}$.

If both MSBs are HIGH, the Address/Mode inputs are interpreted as Mode bits according to the Operational Mode table. There are six operational modes (M0..M6) available as indicated in the table. It is possible to use multiple operational modes simultaneously. Operational Modes are sampled on each falling edge of $\overline{CE}$, and thus Operational Modes and direct addressing are mutually exclusive.

OPERATIONAL MODES

The ISD2500 Series is designed with several built-in operational modes that provide maximum functionality with minimum additional components. These are described in detail below. The operational modes use the address pins on the ISD2500 devices, but are mapped outside the valid address range. When the two Most Significant Bits (MSBs) are HIGH (A8 and A9), the remaining address signals are interpreted as mode bits and not as address bits. Therefore, operational modes and direct addressing are not compatible and cannot be used simultaneously.

There are two important considerations for using operational modes. First, all operations begin initially at address 0, which is the beginning of the ISD2500 address space. Later operations can begin at other address locations, depending on the operational mode(s) chosen. In addition, the address pointer is reset to 0 when the device is changed from Record to Playback, Playback to Record (except M6 mode), or when a Power-Down cycle is executed.

Second, Operational Modes are executed when $\overline{CE}$ goes LOW and the two MSBs are HIGH. This Operational Mode remains in effect until the next LOW-going $\overline{CE}$ signal, at which point the current address/mode levels are sampled and executed.

OPERATIONAL MODES DESCRIPTION

The Operational Modes can be used in conjunction with a microcontroller, or they can be hard-wired to provide the desired system operation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

M0 — Message Cueing

Message Cueing allows the user to skip through messages, without knowing the actual physical addresses of each message. Each $\overline{CE}$ LOW pulse causes the internal address pointer to skip to the next message. This mode should be used for Playback only, and is typically used with the M4 Operational Mode.

M1 — Delete EOM Markers

The M1 Operational Mode allows sequentially recorded messages to be combined into a single message with only one EOM marker set at the end of the final message. When this operational mode is configured, messages recorded sequentially are played back as one continuous message.

M2 — Unused

When operational modes are selected, the M2 pin should be LOW.

M3 — Message Looping

The M3 Operational Mode allows for the automatic, continuously repeated playback of the message located at the beginning of the address space. A message **can** completely fill the ISD2500 device and will loop from beginning to end without $\overline{OVF}$ going LOW.

M4 — Consecutive Addressing

During normal operations, the address pointer will reset when a message is played through to an EOM marker. The M4 Operational Mode inhibits the address pointer reset on $\overline{EOM}$, allowing messages to be played back consecutively.

M5 — $\overline{CE}$-Level Activated

The default mode for ISD2500 devices is for $\overline{CE}$ to be edge-activated on Playback and level-activated on Record. The M5 Operational Mode causes the $\overline{CE}$ pin to be interpreted as level-activated as opposed to edge-activated during Playback. This is specifically useful for terminating Playback operations using the $\overline{CE}$ signal.

In this mode, $\overline{CE}$ LOW begins a Playback cycle, at the beginning of the device memory. The Playback cycle continues as long as $\overline{CE}$ is held LOW. When $\overline{CE}$ goes HIGH, Playback will immediately end. A new $\overline{CE}$ LOW will restart the message from the beginning unless M4 is also HIGH.

M6 — Push-Button Mode

The ISD2500 Series of devices contain a Push-Button operational mode. The Push-Button mode is used primarily in very low-cost applications and is designed to minimize external circuitry and components, thereby reducing system cost. In order to configure the device in Push-Button operational mode, the two most significant address bits must be HIGH, and the M6 mode pin must also be HIGH. A device in this mode always powers down at the end of each Playback or Record cycle after $\overline{CE}$ goes HIGH.

When this operational mode is implemented, several of the pins on the device have alternate functionality:

Table 2-79: Alternate Functionality in Pins

Pin Name	Alternate Functionality in Push-Button Mode
$\overline{CE}$	Start/Pause Push-Button (LOW pulse-activated)
PD	Stop/Reset Push-Button (HIGH pulse activated)
EOM	Active-HIGH Run Indicator

$\overline{CE}$ Pin (START/PAUSE)

In Push-Button Operational Mode, $\overline{CE}$ acts as a LOW-going pulse-activated START/PAUSE signal. If no operation is currently in progress, a LOW-going pulse on this signal will initiate a Playback or a Record cycle according to the level on the P/$\overline{R}$ pin. A subsequent pulse on the $\overline{CE}$ pin, before an End-Of-Message is reached in Playback or an overflow condition occurs, will cause the device to pause. The address counter is not reset, and another $\overline{CE}$ pulse will cause the device to continue the operation from the place where it was paused.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PD Pin (STOP/RESET)

In push-button Operational Mode, PD acts as a HIGH-going pulse-activated STOP/RESET signal. When a Playback or Record cycle is in progress and a HIGH-going pulse is observed on PD, the current cycle is terminated and the address pointer is reset to address 0, the beginning of the message space.

$\overline{\text{EOM}}$ Pin (RUN)

In Push-Button Operational Mode, $\overline{\text{EOM}}$ becomes an active-HIGH RUN signal which can be used to drive an LED or other external device. It is HIGH whenever a Record or Playback operation is in progress.

Recording in Push-Button Mode

1. The PD pin should be LOW, usually using a pulldown resistor.
2. The P/$\overline{\text{R}}$ pin is taken LOW.
3. The $\overline{\text{CE}}$ pin is pulsed LOW. Recording starts, $\overline{\text{EOM}}$ goes HIGH to indicate an operation in progress.
4. The $\overline{\text{CE}}$ pin is pulsed LOW. Recording pauses, $\overline{\text{EOM}}$ goes back LOW. The internal address pointers are not cleared, but an EOM marker is stored in memory to point to the message end. The P/$\overline{\text{R}}$ pin may be taken HIGH at this time. Any subsequent $\overline{\text{CE}}$ would start a playback at address 0.
5. The $\overline{\text{CE}}$ pin is pulsed LOW. Recording starts at the next address after the previous set EOM marker. $\overline{\text{EOM}}$ goes back HIGH.

NOTE If the M1 operational mode pin is also HIGH, the just previously written EOM bit is erased, and recording starts at that address.)

6. When the recording sequences are finished, the final $\overline{\text{CE}}$ pulse LOW will end the last Record cycle, leaving a set EOM marker at the message end. Recording may also be terminated by a HIGH level on PD, which will leave a set EOM marker.

Playback in Push-Button Mode

1. The PD pin should be LOW.
2. The P/$\overline{\text{R}}$ pin is taken HIGH.
3. The $\overline{\text{CE}}$ pin is pulsed LOW. Playback starts, $\overline{\text{EOM}}$ goes HIGH to indicate an operation in progress.
4. If the $\overline{\text{CE}}$ pin is pulsed LOW or an EOM marker is encountered during an operation, the part will pause. The internal address pointers are not cleared, and $\overline{\text{EOM}}$ goes back LOW. The P/$\overline{\text{R}}$ pin may be changed at this time. A subsequent Record operation would not reset the address pointers and the recording would begin where Playback ended.
5. $\overline{\text{CE}}$ is again pulsed LOW. Playback starts where it left off, with $\overline{\text{EOM}}$ going HIGH to indicate an operation in progress.
6. Playback continues as in steps 4 and 5 until PD is pulsed HIGH or overflow occurs.
7. If in overflow, pulling $\overline{\text{CE}}$ LOW will reset the address pointer and start Playback from the beginning. After a PD pulse, the part is reset to address 0.

NOTE Push-button mode can be used in conjunction with modes M0, M1, and M3.

Good Audio Design Practices

ISD products are very high-quality single-chip voice Recording and Playback systems. To ensure the highest quality voice reproduction, it is important that good audio design practices on layout and power supply decoupling be followed. See the ISD Application Notes in this book for details.

ISD1000A COMPATIBILITY

The ISD2500 Series of devices is designed to provide upward compatibility with the ISD1000A family. When designing with the ISD2500 Series, the following differences should be noted.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Addressing

The ISD2560/75/90/120 devices have 480K storage cells designed to provide 60 seconds of storage at a sampling rate of 8.0 KHz. This is approximately four times the storage of the ISD1000A family. To enable the same addressing resolution, two additional address pins have been added. The address space of each device is divisible into 600 increments with valid addressing from 00 to 257 Hex. Some higher addresses are mapped into the Operational Modes. All other addresses are invalid.

Overflow

The ISD1000A Series combined two functions on the $\overline{EOM}$ pin: end-of-message indication and overflow. The ISD2500 separates these two functions. Pin 25 (PDIP package) remains as $\overline{EOM}$, but outputs only the EOM signal indication. Pin 22 (PDIP package) becomes $\overline{OVF}$ and pulses LOW only when the device reaches its end of memory, or is "full." This change allows easy message cueing and addressability across device boundaries. This also means that the M2 operational mode found in the ISD1000A family is not implemented in the ISD2500 Series.

Push-Button Mode

The ISD2500 Series includes an additional Operational Mode called Push-Button mode. This provides an alternative interface to the Record and Playback functions of the part. The $\overline{CE}$ and PD pins become redefined as edge-activated "push-buttons." A pulse on $\overline{CE}$ initiates a cycle, and if triggered again, pauses the current cycle without resetting the address pointer (i.e., a Start or Pause function). PD stops any current cycle and resets the address pointer to the beginning of the message space (i.e., a Stop and Reset function). Additionally, the $\overline{EOM}$ pin functions as an active-HIGH run indicator, and can be used to drive an LED indicating a Record or Playback operation is in progress. Devices in the Push-Button mode cannot be cascaded.

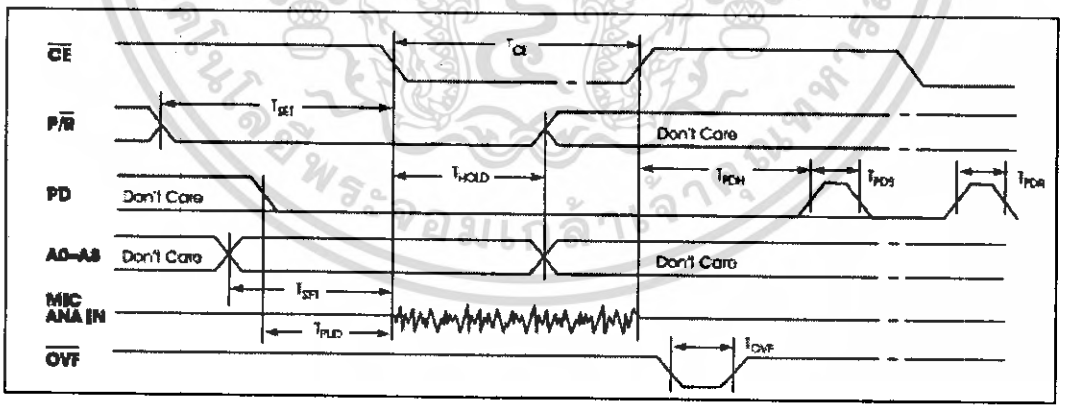
Looping Mode

The ISD2500 Series can loop with a message that completely fills the memory space.

NOTE Additional descriptions of ISD2500 device functionality and application examples are provided in the ISD Application Notes in this book.

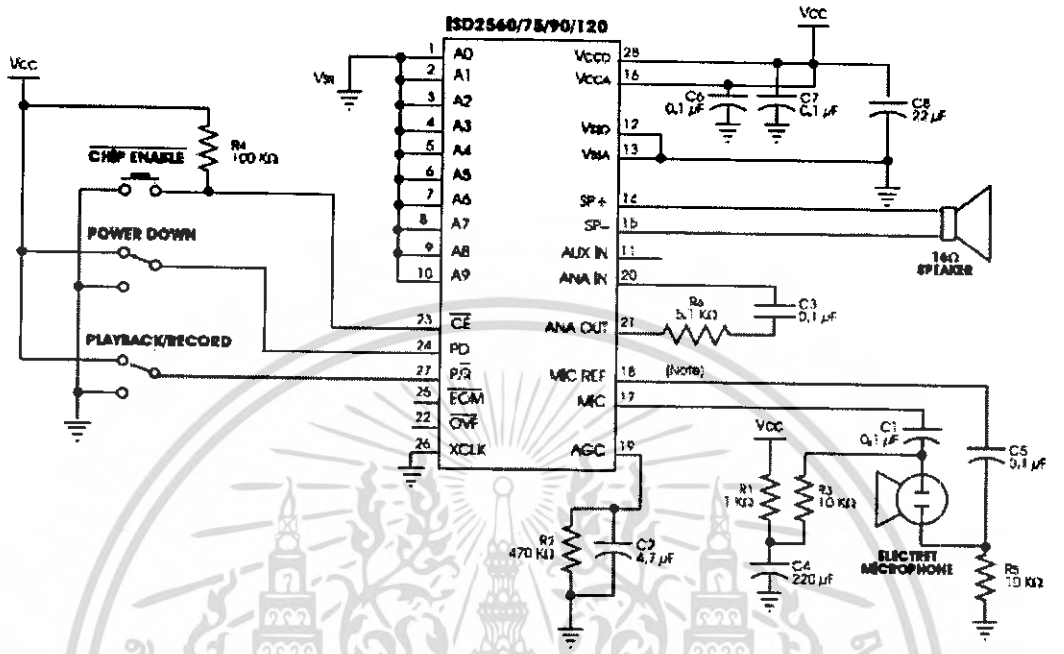
TIMING DIAGRAMS

Figure 2-35: Record



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 2-37: ISD2560/75/90/120 Application Example—Design Schematic



NOTE: *If desired, pin 18 (PDIP package) may be left unconnected (microphone preamplifier noise will be higher). In this case, pin 18 must not be tied to any other signal or voltage. Additional design example schematics are provided in the Application Notes in this book.*

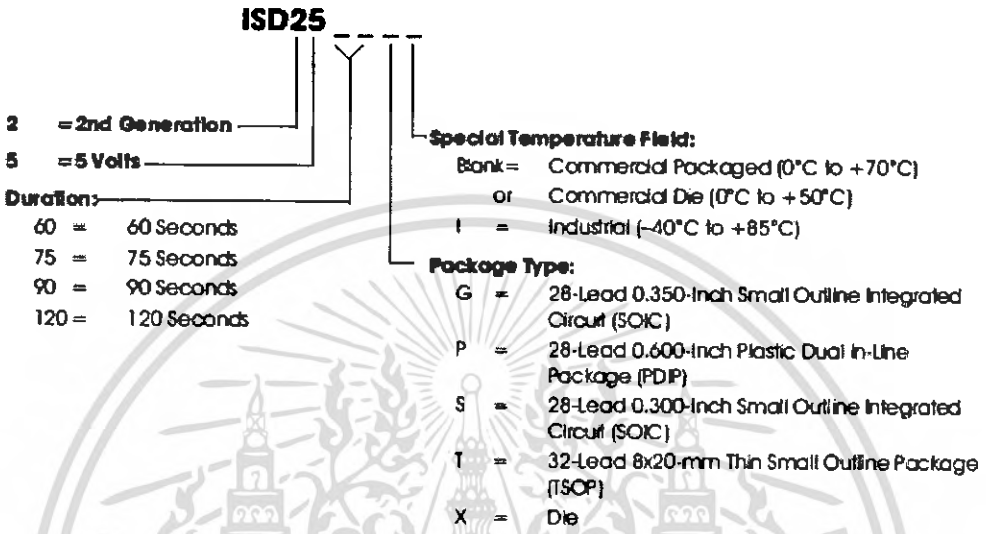
Table 2-88: Application Example—Basic Device Control

Control Step	Function	Action
1	Power up chip and select Record/Playback mode	1. PD = LOW, 2. P/$\overline{R}$ = As desired
2	Set message address for Record/Playback	Set addresses A0–A9
3A	Begin Playback	P/$\overline{R}$ = HIGH, $\overline{CE}$ = Pulsed LOW
3B	Begin Record	P/$\overline{R}$ = LOW, $\overline{CE}$ = LOW
4A	End Playback	Automatic
4B	End Record	PD or $\overline{CE}$ = HIGH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ORDERING INFORMATION

Product Number Descriptor Key



When ordering ISD2560/75/90/120 products, please refer to the following valid part numbers.

Part Number	Part Number	Part Number	Part Number
ISD2560G	ISD2575G	ISD2590G	ISD25120G
ISD2560GI	ISD2575GI	ISD2590P	ISD25120P
ISD2560P	ISD2575P	ISD2590S*	ISD25120X
ISD2560PI	ISD2575PI	ISD2590T	
ISD2560S*	ISD2575S*	ISD2590X	
ISD2560SI*	ISD2575SI*		
ISD2560T	ISD2575T		
ISD2560TI	ISD2575TI		
ISD2560X	ISD2575X		

NOTE: *The asterisk (*) indicates advance information*

For the latest product information, access ISD's worldwide website at http://www.isd.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UMC

T-77-13

UM3561

Three Siren Sound Generator

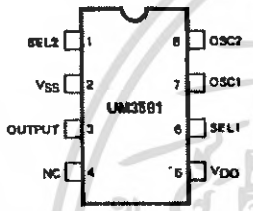
Features

- Four sounds can be selected
- Typical 3V operating voltage
- RC oscillator with an external resistor
- A magnetic speaker can be driven by connecting an NPN transistor
- Power on reset

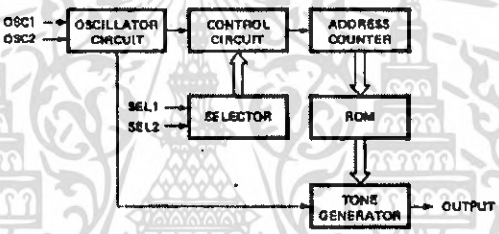
General Description

The UM3561 is a low-cost, low-power CMOS LSI designed for use in toy applications. Since the integrated circuit includes oscillator and selector circuits, a compact sound module can be constructed with only a few additional components. The UM3561 contains a programmed mask ROM to simulate siren sounds.

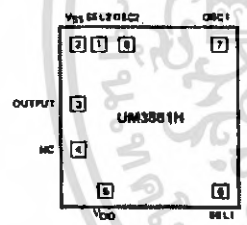
Pin Configuration



Block Diagram



Pad Configuration



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UMC

UM3561

T-77-13

Absolute Maximum Ratings*

DC Supply Voltage . . . . . −0.3V to +5.0V
Input/Output Voltage . . . . . V_{SS} −0.3V to V_{DD} +0.3V
Operating Ambient Temperature . . . . . −10°C to 60°C
Storage Temperature . . . . . −55°C to 125°C

*Comments

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

D.C. Electrical Characteristics (V_{SS} = 0V, T_A = 25°C, F_{OSC} = 106496 Hz, unless otherwise specified.)

Parameter	Symbol	Min.	Typ.	Max.	Conditions
Operating Voltage	V_{DD}	2.4V	3V	3.6V	
Operating Current	I_{DD}	–	–	160μA	V_{DD} = 3V
"H" Input Voltage	V_{IH}	V_{DD} −0.2	–	V_{DD}	
"L" Input Voltage	V_{IL}	V_{SS}	–	V_{SS} +0.2	
Frequency Stability	ΔF/F	–	–	20%	$\frac{F_{OSC}(3.3V) - F_{OSC}(2.7V)}{F_{OSC}(2.7V)}$
Output Current	I_D	3mA	–	–	V_{DD} = 3V
Frequency Deviation Per Lot	ΔF/F	−10%	–	+10%	V_{DD} = 3V

Pin Description

Pin No.	Designation	Description
1	SEL 2	Sound effect selection pin
2	V_{SS}	Negative power supply
3	OUTPUT	Mono-tone output
4	NC	This pad is used for testing; in normal operation, this pad is open.
5	V_{DD}	Positive power supply
6	SEL 1	Sound effect selection pin
7	OSC 1	RC oscillator pin
8	OSC 2	RC oscillator pin or inverted clock output

Functional Description

Oscillating circuit

There are two options for generating oscillator frequency. Either can be selected by changing the mask.
(1) Only one external resistor is required to complete the oscillator circuit.
(2) Oscillator resistor is built-in

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UMC

UM3561

T-97-13

Sound Selection

The SEL2 incorporates a resistor for internal pull low, and SEL1 is a tri-state control pin. 2 pads, SEL1 and SEL2, should be selected for the sound effect mode.

Bonding Pad		Sound Effect
SEL 1	SEL 2	
No Connection	No Connection	Police Siren
V_{DD}	No Connection	Fire Engine Siren
V_{SS}	No Connection	Ambulance Siren
"–" don't care	V_{DD}	Machine Gun

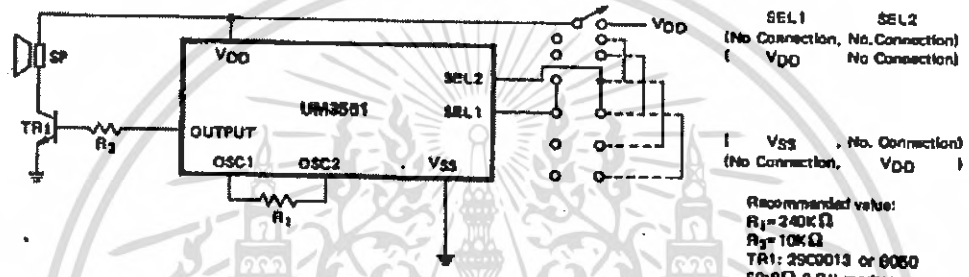
Sound Effect ROM

The sound effect ROM is organized as 256 words by 8 bits. The sound effect program and the option are mask programmable and programmed in the N^+ layer.

Typical Application

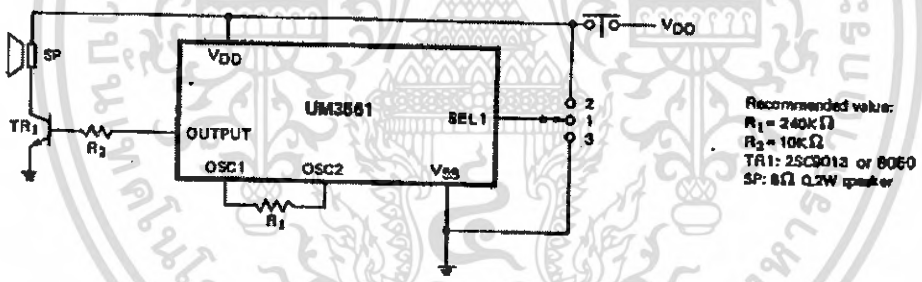
FOUR SOUND APPLICATION

1. Police Siren 2. Fire Engine Siren 3. Ambulance Siren 4. Machine Gun



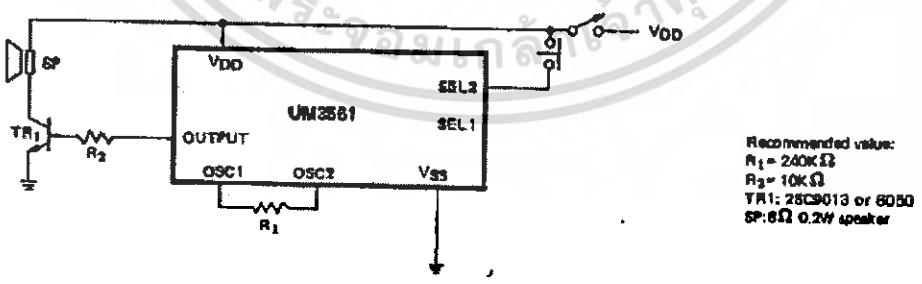
THREE SOUND APPLICATION

1. Police Siren 2. Fire Engine Siren 3. Ambulance Siren



TWO SOUND APPLICATION

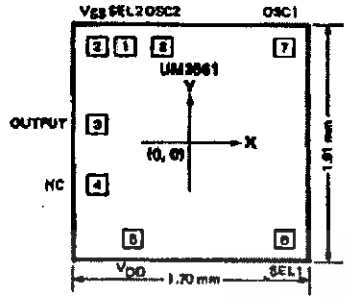
1. Police Siren 2. Machine Gun



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UMC

Bonding Diagram



unit: μm

Pad No.	Designation	X	Y
1	SEL2	−266.70	777.24
2	V_{SS}	−515.62	777.24
3	OUTPUT	−698.50	172.72
4	NC	−698.50	−370.84
5	V_{DD}	−482.60	−767.08
6	SEL1	690.88	−762.00
7	OSC1	698.50	777.24
8	OSC2	− 17.78	777.24

Ordering Information

Part Number	Package
UM3561H	Chip
UM3561	8L DIP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้