

การพัฒนาเครื่องวิเคราะห์แบบหลายช่องขนาดกะทัดรัดที่ใช้เทคนิควิลกินสันเอ็ดจี



นายสุภฤกษ์ ลีตระกูลพิชิตชัย

สถาบันวิทยบริการ

จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

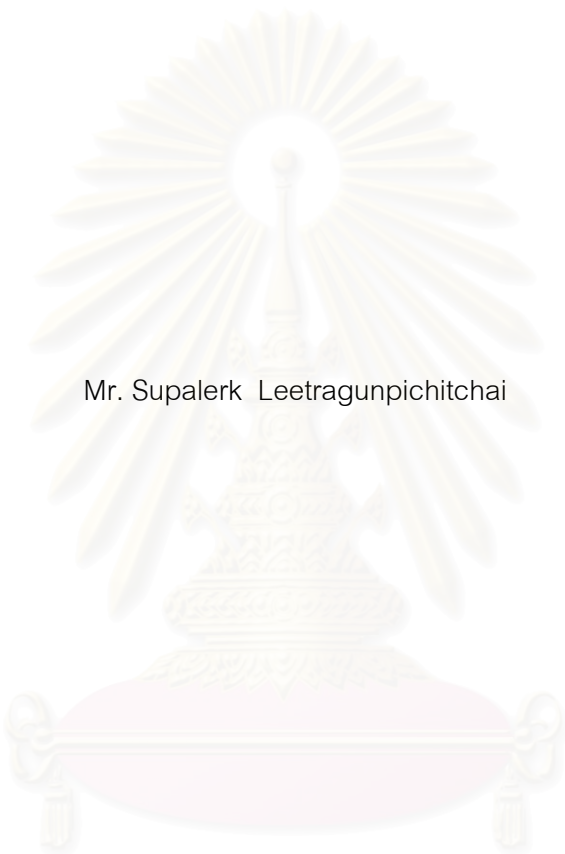
สาขาวิชานิวเคลียร์เทคโนโลยี ภาควิชานิวเคลียร์เทคโนโลยี

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2549

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

DEVELOPMENT OF A COMPACT MULTICHANNEL ANALYZER USING WILKINSON ADC
TECHNIQUE



Mr. Supalerk Leetragunpichitchai

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Nuclear Technology

Department of Nuclear Technology

Faculty of Engineering

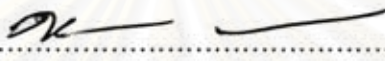
Chulalongkorn University

Academic Year 2006

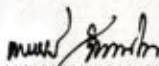
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การพัฒนาเครื่องวิเคราะห์แบบหลายช่องขนาดกะทัดรัดที่ใช้เทคนิควัด คินตันเอ็ดซี
โดย	นายศุภฤกษ์ ลีตระกูลพิชิตชัย
สาขาวิชา	นิวเคลียร์เทคโนโลยี
อาจารย์ที่ปรึกษา	อาจารย์เดโช ทองอร่าม
อาจารย์ที่ปรึกษาร่วม	ผู้ช่วยศาสตราจารย์สุวิทย์ ปุณณชัยยะ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยาลัยรับเป็น
ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

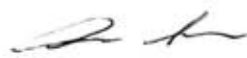

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. ดิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์


..... ประธานกรรมการ
(รองศาสตราจารย์นเรศร์ จันทน์ขาว)


..... อาจารย์ที่ปรึกษา
(อาจารย์เดโช ทองอร่าม)


..... อาจารย์ที่ปรึกษาร่วม
(ผู้ช่วยศาสตราจารย์สุวิทย์ ปุณณชัยยะ)


..... กรรมการ
(ผู้ช่วยศาสตราจารย์อรรถพร ภัทรสุมันต์)

ศุภฤกษ์ ถีระกุลพิชิตชัย : การพัฒนาเครื่องวิเคราะห์แบบหลายช่องขนาดกะทัดรัดที่ใช้เทคนิควิลกินสันเอดีซี (DEVELOPMENT OF A COMPACT MULTICHANNEL ANALYZER USING WILKINSON ADC TECHNIQUE) อ.ที่ปรึกษา: อ.เค โข ทองอร่าม, อ.ที่ปรึกษาร่วม: ผศ. สุวิทย์ ปุณณชัยยะ, 117 หน้า.

งานวิจัยนี้เป็นการพัฒนาเครื่องวิเคราะห์ความสูงพัลส์แบบหลายช่องขนาดกะทัดรัดด้วยเทคนิควิลกินสันเอดีซีที่ฐานความถี่ 100 MHz ชนิดทำงานแบบเบ็ดเสร็จในตัว (Stand- Alone Type) ขนาด 2048 ช่องวิเคราะห์ โดยการพัฒนาได้นำเทคโนโลยี CPLD และไมโครคอนโทรลเลอร์ (LPC2148) ขนาด 16/32 บิต เข้ามาประยุกต์ใช้ในส่วนของวงจรแปลงผันสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล และวงจรควบคุมการทำงานของระบบและการแสดงผลของจอภาพชนิดแอลซีดีสี (Color LCD) พร้อมโปรแกรมควบคุมการจัดการข้อมูลของระบบวิเคราะห์ จึงทำให้สามารถลดได้ทั้งขนาด ความซับซ้อนของวงจร และอัตราการสูญเสียกำลังไฟฟ้า ขณะเดียวกันการทำงานที่ความถี่สูงก็ดีขึ้น ซึ่งระบบที่ทำการพัฒนาประกอบด้วย แหล่งจ่ายไฟฟ้าศักดาต่ำ แหล่งจ่ายไฟฟ้าศักดาสูง อุปกรณ์ขยายสัญญาณพัลส์ และวงจรแปลงผันสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล

จากผลการวิจัยพบว่า แหล่งจ่ายไฟฟ้าศักดาสูงสามารถ ปรับค่าได้จาก 0 ถึง 2000 โวลต์ จ่ายกระแสสูงสุด 1 mA มีแรงดันริปเปิ้ลน้อยกว่า 50 mV วงจรขยายสัญญาณพัลส์ให้สัญญาณทางออกเป็นยูนิโพลลาขนาดความกว้าง 2.5 μ s สามารถปรับอัตราขยายได้ต่อเนื่อง 100, 500 และ 1000 เท่า เป็นสัดส่วนกับระดับพลังงานที่ได้จากหัววัดรังสี การทำงานของเอดีซีพบว่ามีความเป็นเชิงเส้นที่ $R^2 = 0.9997$ และแปลงอัตราสูงสุดได้ 48.828 kcps และเมื่อทดสอบการวิเคราะห์สเปกตรัมพลังงานของรังสีแกมมาโดยใช้หัววัดรังสี NaI (TI) ขนาด 2" \times 2" พบว่าสามารถวิเคราะห์สเปกตรัมได้ถูกต้องรองรับอัตรานับรังสีได้ 100 cps อย่างไรก็ตามการปรับระบบให้สามารถรับอัตรานับรังสีได้สูงกว่านี้จำเป็นต้องแก้ระบบวิธีทางโปรแกรมในการบันทึกข้อมูลและแสดงผล

ภาควิชา...นิวเคลียร์เทคโนโลยี.....ลายมือชื่อนิสิต.....
 สาขาวิชา...นิวเคลียร์เทคโนโลยี.....ลายมือชื่ออาจารย์ที่ปรึกษา.....
 ปีการศึกษา 2549.....ลายมือชื่ออาจารย์ที่ปรึกษาร่วม.....

4670529921: MAJOR NUCLEAR TECHNOLOGY

KEY WORD: MULTICHANNEL ANALYZER / WILKINSON / MCA/ ADC

SUPALERK LEETRAGUNPICHITTHAI: DEVELOPMENT OF A COMPACT MULTICHANNEL ANALYZER USING WILKINSON ADC TECHNIQUE. THESIS ADVISOR: DECHO THONG-ARAM, THESIS CO – ADVISOR: ASST. PROF. SUVIT PUNNACHAIYA, 117 pp.

This thesis a compact size stand-alone multichannel analyzer (MCA) used Wilkinson ADC technique at 100 MHz clock frequency with 2048 channel was developed. This development applied CPLD technology and 16/32 bit microcontroller (LPC2148) for analog to digital converter circuit, control circuit and the color LCD display of the system, including the data manipulation and control programs of the analysis system. Complexity of the circuits, equipment size and also the power consumption were therefore reduced while the operation speed was increased. The developed system consisted of low voltage power supply, High voltage power supply, pulse amplifier and analog to digital converter circuit.

The results of this research were found that the high voltage power supply could be adjusted from 0 to 2000 V at a maximum current of 1 mA with a ripple voltage of less than 50 mV. The pulse amplification could produce a unipolar pulse output with 2.5 μ s shaping time which continuously gain adjustment of 100, 500 and 1000 times corresponding to energy level from the detector. The linearity test of the ADC was found to be excellent with $R^2 = 0.9997$ and a maximum pulse rate of 48.828 kcps could be converted. The gamma energy spectrum analysis using 2" x 2" NaI (TI) scintillation detector showed that a smooth spectrum at counting rate less than 100 cps could be obtained. However, the count rate could be increased by improving the software algorithm of data recording and displaying.

Department.....Nuclear TechnologyStudent's signature.....
 Field of study...Nuclear Technology..... Advisor's signature.....
 Academic year 2006.....Co-Advisor's signature.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี ผู้เขียนขอกราบขอบพระคุณ อาจารย์เดโช ทองอร่าม ผู้ช่วยศาสตราจารย์สุวิทย์ ปุณณชัยยะ ที่ได้ประสิทธิ์ประสาทวิชาและองค์ความรู้ต่างๆ ให้คำปรึกษาชี้แนะ อีกทั้งให้โอกาสในการศึกษาและทำวิทยานิพนธ์นี้ให้สำเร็จลุล่วงไปด้วยดี

ขอขอบคุณคุณกมลทิพย์ พลอยกระจ่าง และ ศูนย์เชี่ยวชาญนวัตกรรมเทคโนโลยีสำหรับวิเคราะห์และทดสอบวัสดุที่ได้อำนวยความสะดวกด้านเครื่องมือและอุปกรณ์ต่างๆ สำหรับงานวิจัยนี้ และขอขอบคุณเพื่อนๆ พี่ๆ และน้องๆ ที่ให้ความช่วยเหลือสำหรับงานวิจัยนี้

ขอขอบคุณบัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัยที่สนับสนุนทุนวิจัยวิทยานิพนธ์

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และพี่ๆ ทุกคนที่บ้านที่คอยเป็นกำลังใจ ให้คำปรึกษา และให้โอกาสในการศึกษา



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ญ
สารบัญภาพ.....	ฎ
บทที่	
1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	2
1.3 ขอบเขตของการวิจัย.....	2
1.4 ขั้นตอนและวิธีดำเนินการวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย.....	2
1.6 เอกสารและงานวิจัยที่เกี่ยวข้อง.....	3
2 อุปกรณ์วิเคราะห์แบบหลายช่อง.....	4
2.1 การวิเคราะห์ความสูงของพัลส์.....	4
2.1.1 สัญญาณพัลส์นิวเคลียร์.....	4
2.1.2 การวิเคราะห์ความสูงพัลส์แบบช่องเดียว.....	5
2.2 อุปกรณ์วิเคราะห์แบบหลายช่อง.....	6
2.2.1 โครงสร้างพื้นฐานของอุปกรณ์วิเคราะห์แบบหลายช่อง.....	6
2.2.2 การแปลงความสูงของพัลส์เป็นสัญญาณดิจิทัล.....	8
2.3 รูปแบบของอุปกรณ์วิเคราะห์แบบหลายช่อง.....	10
2.4 การออกแบบวงจรดิจิทัลด้วย FPGA หรือ CPLD.....	11
2.4.1 คุณสมบัติพื้นฐานของ CPLD.....	11
2.4.2 คุณสมบัติพื้นฐานของ FPGA.....	12
2.4.3 การออกแบบวงจรในไอซี CPLD หรือ FPGA.....	12
2.4.4 ขั้นตอนการออกแบบในไอซี CPLD หรือ FPGA.....	13
2.5 ไมโครคอนโทรลเลอร์ ARM7.....	14
3 การดำเนินการวิจัย.....	16

3.1	ข้อมูลพื้นฐานของระบบ.....	16
3.1.1	ข้อมูลพื้นฐานสำหรับหัววัดรังสี.....	16
3.1.2	ข้อมูลพื้นฐานสำหรับระบบรังสีแกมมา.....	18
3.1.3	โครงสร้างระบบควบคุมและการแสดงผลที่พัฒนาขึ้น.....	19
3.2	การพัฒนาระบบวัดรังสี.....	20
3.2.1	การพัฒนางจรแหล่งจ่ายไฟฟ้าศักดาสูง.....	20
3.2.2	การพัฒนางจรขยายสัญญาณพัลส์.....	23
3.2.3	การพัฒนางจรแปลงสัญญาณอนาลอกเป็นดิจิทัล.....	25
3.2.3.1	ส่วนของการทำงานของสัญญาณอนาลอก.....	26
3.2.3.2	ส่วนของการทำงานของสัญญาณดิจิทัล.....	27
3.3	การพัฒนาระบบควบคุมการทำงานและการแสดงผล.....	33
3.3.1	ระบบควบคุมการทำงานโดยใช้ไมโครคอนโทรลเลอร์ ARM7.....	33
3.3.2	สถาปัตยกรรมซีพียู ARM7.....	34
3.3.2.1	การจัดวางขาของ LPC2138.....	35
3.3.2.2	บล็อกไดอะแกรมของ LPC 2138.....	36
3.3.2.3	การจัดหน่วยความจำของ LPC2138.....	37
3.3.3	ระบบการแสดงผล.....	41
3.3.3.1	คุณสมบัติเฉพาะของจอแสดงผล LM32C041 ยี่ห้อ SHARP.....	41
3.3.3.2	การออกแบบแหล่งจ่ายไฟฟ้าให้กับจอแสดงผล.....	42
3.3.3.3	การเขียนโปรแกรมควบคุมการแสดงผล.....	44
4	การทดสอบสมรรถนะของระบบ.....	47
4.1	การทดสอบระบบในแต่ละส่วน.....	47
4.1.1	การทดสอบแหล่งจ่ายไฟฟ้าศักดาสูง.....	47
4.1.2	การทดสอบวงจรขยายสัญญาณพัลส์.....	51
4.1.3	การทดสอบการทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล.....	57
4.1.4	การทดสอบระบบควบคุมการทำงานและการแสดงผล.....	67
4.2	การทดสอบทั้งระบบ.....	68
5	สรุปผลการวิจัยและข้อเสนอแนะ.....	72
5.1	สรุปผลการวิจัย.....	72
5.1.1	ผลทดสอบการทำงานของเครื่อง.....	72

บทที่	ณ หน้า
5.1.2 สมรรถนะและประโยชน์การใช้งานของเครื่อง.....	72
5.2 คุณสมบัติของเครื่อง.....	73
5.2.1 คุณสมบัติของระบบวิเคราะห์ทางด้านฮาร์ดแวร์.....	73
5.2.2 คุณสมบัติของระบบวิเคราะห์ทางด้านซอฟต์แวร์.....	73
5.3 ปัญหาและอุปสรรค.....	74
5.4 ข้อเสนอแนะ.....	74
รายการอ้างอิง.....	75
ภาคผนวก ก.....	77
ภาคผนวก ข.....	86
ประวัติผู้เขียนวิทยานิพนธ์.....	117

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ณ

ตาราง	หน้า
ตารางที่ 3.1 คุณสมบัติทางไฟฟ้าของจอ LCD32C041.....	41
ตารางที่ 3.2 คุณสมบัติเวลาในการอินเทอร์เฟสของจอ LCD32C041.....	41
ตารางที่ 4.1 การทดสอบความเป็นเชิงเส้นของสัญญาณที่ได้จากแหล่งจ่ายไฟฟ้าคิกดาสูง.....	49
ตารางที่ 4.2 แสดงค่า Load Regulation ของแหล่งจ่ายไฟฟ้าคิกดาสูง.....	51
ตารางที่ 4.3 การทดสอบความเป็นเชิงเส้นของวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล.....	66



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญญภาพ

	ฉ
ภาพประกอบ	หน้า
รูปที่ 2.1 แผนภาพของระบบวิเคราะห์พลังงานของอนุภาคนิวเคลียร์.....	4
รูปที่ 2.2 การวิเคราะห์ความสูงของพัลส์ด้วยอุปกรณ์วิเคราะห์ช่องเดียว.....	5
รูปที่ 2.3 แผนภาพการทำงานเบื้องต้นของ MCA	7
รูปที่ 2.4 การแสดงสเปกตรัมนิวเคลียร์ของ MCA.....	7
รูปที่ 2.5 แผนภาพหลักการทำงานของวงจรแปลงสัญญาณแบบวิลคินสัน	8
รูปที่ 2.6 เปรียบเทียบการแปลงความสูงของพัลส์เป็นรหัสไบนารี.....	10
รูปที่ 2.7 ขั้นตอนการออกแบบวงจรดิจิทัลด้วย FPGA หรือ CPLD	13
รูปที่ 3.1 โครงสร้างของหัววัดรังสีชนิดซิลิคอน.....	17
รูปที่ 3.2 โครงสร้างภายในหลอดทวิคูณอิเล็กทรอนิกส์.....	18
รูปที่ 3.3 แผนภาพของระบบวิเคราะห์แบบหลายช่อง.....	19
รูปที่ 3.4 แผนภาพของวงจรแหล่งจ่ายไฟฟ้าศักดาสูง.....	20
รูปที่ 3.5 แผ่นวงจรแหล่งจ่ายไฟฟ้าศักดาสูง.....	21
รูปที่ 3.6 วงจรแหล่งจ่ายไฟฟ้าศักดาสูง.....	22
รูปที่ 3.7 แผนภาพแสดงการทำงานของวงจรมายสัญญาณ.....	23
รูปที่ 3.8 แผ่นวงจรมายสัญญาณพัลส์.....	23
รูปที่ 3.9 วงจรมายสัญญาณพัลส์.....	24
รูปที่ 3.10 แผนภาพวงจรแปลงสัญญาณพัลส์ในส่วนของสัญญาณอนาล็อก.....	25
รูปที่ 3.11 แผนภาพวงจรแปลงสัญญาณพัลส์ในส่วนของสัญญาณดิจิทัล.....	25
รูปที่ 3.12 แผ่นวงจรแปลงสัญญาณพัลส์ในส่วนของแปลงผันสัญญาณอนาล็อก.....	26
รูปที่ 3.13 แผ่นวงจรแปลงสัญญาณพัลส์ในส่วนของแปลงผันสัญญาณดิจิทัล.....	28
รูปที่ 3.14 วงจรแปลงสัญญาณพัลส์ในส่วนของแปลงผันสัญญาณอนาล็อก.....	29
รูปที่ 3.15 วงจรแปลงสัญญาณพัลส์ในส่วนของแปลงผันสัญญาณดิจิทัล.....	30
รูปที่ 3.16 แผนภาพเวลาของการแปลงผันสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล.....	31
รูปที่ 3.17 แสดงการจัดขาของ LPC2148.....	35
รูปที่ 3.18 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ LPC2138.....	36
รูปที่ 3.19 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ LPC2138.....	37

ภาพประกอบ	หน้า
รูปที่ 3.20 แหล่งจ่ายไฟฟ้า +3.3 โวลต์ของไมโครคอนโทรลเลอร์ LPC2138.....	38
รูปที่ 3.21 แผ่นวงจรแหล่งจ่ายไฟฟ้า +3.3 โวลต์และพอร์ตต่อใช้งาน.....	39
รูปที่ 3.22 แผ่นวงจรของไมโครคอนโทรลเลอร์ LPC2138.....	39
รูปที่ 3.23 วงจรของไมโครคอนโทรลเลอร์ LPC2138.....	40
รูปที่ 3.24 แสดงวงจรการจ่ายไฟฟ้าศักดาสูง +1000 โวลต์ ให้กับจอแสดงผล.....	42
รูปที่ 3.25 แสดงวงจรการจ่ายไฟฟ้าศักดาต่ำ +5 โวลต์.....	42
รูปที่ 3.26 แสดงวงจรการจ่ายไฟฟ้าศักดาต่ำ +30 โวลต์.....	43
รูปที่ 3.27 แผ่นวงจรแหล่งจ่ายไฟฟ้า +5, +30 และ +1200 โวลต์.....	43
รูปที่ 3.28 แสดงการส่งสัญญาณ VSYNC, HSYNC และ DCK.....	44
รูปที่ 3.29 ผังการทำงานของโปรแกรมสำหรับในการควบคุมจอแสดงผล.....	45
รูปที่ 3.30 แสดงภาพจอแสดงผล LCD.....	46
รูปที่ 3.31 จอภาพ LCD ที่แสดงผลสเปกตรัมรังสี.....	46
รูปที่ 4.1 แผนภาพการจับอุปกรณ์ทดสอบแหล่งจ่ายไฟฟ้าศักดาสูง.....	48
รูปที่ 4.2 ภาพการจับอุปกรณ์ทดสอบแหล่งจ่ายไฟฟ้าศักดาสูง โดยใช้ Electrostatic.....	48
รูปที่ 4.3 แสดงค่าความเป็นเชิงเส้นของแหล่งจ่ายไฟฟ้าศักดาสูง.....	49
รูปที่ 4.4 แสดงค่าความเรียบ (Ripple) ของแหล่งจ่ายไฟฟ้าศักดาสูง.....	49
รูปที่ 4.5 ภาพการทดสอบแหล่งจ่ายไฟฟ้าศักดาสูง โดยใช้ Chart Recorder.....	50
รูปที่ 4.6 กราฟแสดงค่าความ Stability ของแหล่งจ่ายไฟฟ้าศักดาสูง.....	51
รูปที่ 4.7 แผนภาพการจับอุปกรณ์ทดสอบวงจรขยายสัญญาณพัลส์.....	52
รูปที่ 4.8 ภาพการจับอุปกรณ์ทดสอบวงจรขยายสัญญาณพัลส์.....	53
รูปที่ 4.9 สัญญาณพัลส์ที่ได้จาก โมดูลแหล่งกำเนิดสัญญาณพัลส์.....	53
รูปที่ 4.10 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier.....	54
รูปที่ 4.11 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ.....	54
รูปที่ 4.12 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier.....	55
รูปที่ 4.13 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ.....	55
รูปที่ 4.14 สัญญาณพัลส์ที่ได้จาก โมดูลแหล่งกำเนิดสัญญาณพัลส์.....	55
รูปที่ 4.15 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier.....	56
รูปที่ 4.16 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ.....	56
รูปที่ 4.17 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier.....	57

ภาพประกอบ	หน้า
รูปที่ 4.18 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ.....	57
รูปที่ 4.19 แผนภาพการจับอุปกรณ์ทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล.....	58
รูปที่ 4.20 ภาพการทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลส่วนของอนาลอก.....	58
รูปที่ 4.21 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier.....	59
รูปที่ 4.22 สัญญาณพัลส์จากโมดูลขยายสัญญาณพัลส์ที่ถูกยึดออก.....	59
รูปที่ 4.23 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ.....	60
รูปที่ 4.24 สัญญาณพัลส์จากวงจรขยายสัญญาณพัลส์ที่ออกแบบถูกยึดออก.....	60
รูปที่ 4.25 สัญญาณ ULD.....	60
รูปที่ 4.26 สัญญาณ LLD.....	61
รูปที่ 4.27 สัญญาณ THR.....	61
รูปที่ 4.28 แสดงความสูงสัญญาณพัลส์ต่ำสุดที่ THR และ LLD จะสามารถตรวจจับได้.....	61
รูปที่ 4.29 สัญญาณ THR และ LLD ที่ความสูงสัญญาณพัลส์ประมาณ 250 mV.....	62
รูปที่ 4.30 ทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลส่วนของอนาลอกและดิจิทัล.....	62
รูปที่ 4.31 ภาพการต่อวงจรความถี่ป้อนในหัวจรแปลงสัญญาณอนาลอกเป็นดิจิทัล.....	63
รูปที่ 4.32 สัญญาณความถี่ที่ป้อนในหัวจรแปลงสัญญาณอนาลอกเป็นดิจิทัล.....	63
รูปที่ 4.33 สัญญาณพัลส์ที่ได้จากโมดูลขยายสัญญาณพัลส์.....	64
รูปที่ 4.34 การยึดความสูงพัลส์เป็นค่าเวลาที่เป็นเชิงเส้นตามความสูงของพัลส์.....	64
รูปที่ 4.35 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ.....	65
รูปที่ 4.36 การยึดความสูงพัลส์เป็นค่าเวลาที่เป็นเชิงเส้นตามความสูงของพัลส์.....	65
รูปที่ 4.37 ภาพการแปลงสัญญาณอนาลอกเป็นดิจิทัลที่มีความกว้างขนาด 0.9 μ Sec.....	66
รูปที่ 4.38 แผนภาพการจับอุปกรณ์ทดสอบระบบควบคุมการทำงานและการแสดงผล.....	67
รูปที่ 4.39 แสดงการทดสอบระบบควบคุมการทำงานและการแสดงผล.....	68
รูปที่ 4.40 แผนภาพการจับอุปกรณ์ทดสอบระบบวิเคราะห์สเปกตรัมพลังงานรังสีแกมมา.....	69
รูปที่ 4.41 อุปกรณ์วิเคราะห์แบบหลายช่องและหัววัดรังสีแกมมาจาก Cs-137.....	69
รูปที่ 4.42 สเปกตรัมพลังงานรังสีแกมมาของ Cs-137.....	70
รูปที่ 4.43 อุปกรณ์วิเคราะห์แบบหลายช่องและหัววัดรังสีแกมมาจาก Co-60.....	70
รูปที่ 4.44 สเปกตรัมพลังงานรังสีแกมมาของ Co-60.....	71

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

อุปกรณ์วิเคราะห์แบบหลายช่องสัญญาณ (Multichannel Analyzer) เป็นอุปกรณ์ที่ใช้ในการวิเคราะห์พลังงานของอนุภาคนิวเคลียร์ โดยใช้วิธีการจำแนกความสูงของสัญญาณพัลส์ (Pulse Height Analysis) จากระบบวัดนิวเคลียร์ด้วยการทำงานในระบบเชิงตัวเลขมาทำการแสดงผลของสเปกตรัมทางจอภาพพร้อมรายละเอียดของตำแหน่งพลังงาน และปริมาณของอนุภาคนิวเคลียร์ที่วิเคราะห์ได้ ทำให้การแปลผลวิเคราะห์ทำได้สะดวก เนื่องจากในตัวอุปกรณ์วิเคราะห์แบบหลายช่องมีสมรรถนะการทำงานสูง จึงมีความสำคัญต่องานวิจัยด้านการวิเคราะห์ธาตุ ทั้งเชิงคุณภาพและปริมาณ โดยเฉพาะการตรวจหาองค์ประกอบของสารในระดับความเข้มข้นต่ำมากด้วยเทคนิคนิวเคลียร์ ได้แก่ เทคนิคการเรืองรังสีเอกซ์ (x-ray Fluorescence) เทคนิคการวัดพรอมต์แกมมา (Prompt Gamma-ray Analysis) เทคนิคการเหนี่ยวนำให้เกิดรังสีเรืองด้วยอนุภาค (Particle-Induced x-ray Emission) และเทคนิคการวิเคราะห์ด้วยลำอิเล็กตรอน (Electron Probe Micro Analysis) เป็นต้น ที่ผ่านมามีประเทศไทยยังไม่มีการพัฒนาอย่างจริงจังและต่อเนื่องเกี่ยวกับอุปกรณ์วิเคราะห์แบบหลายช่องสัญญาณนี้ ทำให้เครื่องมือที่อยู่ในปัจจุบันหรือที่ได้ทำการพัฒนาไปแล้วมีขนาดใหญ่ไม่สะดวกในการเคลื่อนย้ายและเทคโนโลยียังล้าหลังอยู่เมื่อเทียบกับเครื่องจากต่างประเทศ ในปัจจุบันจะพบว่าเทคโนโลยีของจีนส่วนและอุปกรณ์อิเล็กทรอนิกส์มีความทันสมัย โดยมีสมรรถนะการทำงานที่สูงขึ้น ขณะเดียวกันก็มีขนาดเล็กและสะดวกต่อการใช้งานมากขึ้น

จึงทำให้เกิดแนวคิดที่จะทำการพัฒนาเครื่องวิเคราะห์ความสูงของสัญญาณพัลส์นิวเคลียร์แบบหลายช่องสัญญาณชนิดพกพาให้มีประสิทธิภาพที่ทัดเทียมเครื่องจากต่างประเทศ โดยให้มีขนาดเล็กสามารถนำติดตัวไปใช้งานได้อย่างสะดวก

จุฬาลงกรณ์มหาวิทยาลัย

1.2 วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาเครื่องวิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิคซิลิคอนสันเอ็ดจีทำงานร่วมกับ CPLD (Complex Programmable Logic Device) หรือ FPGA (Field Programmable Gate Array)

1.3 ขอบเขตของการวิจัย

1. ออกแบบและสร้างเครื่องมือวิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิคซิลิคอนสันเอ็ดจีทำงานร่วมกับ CPLD หรือ FPGA โดยมีขนาดช่องวิเคราะห์ไม่ต่ำกว่า 2048 ช่อง ที่ความถี่ 100 MHz และแสดงผลทางจอ LCD
2. สร้างระบบควบคุมการทำงานและระบบเชื่อมโยงสัญญาณ ด้วยไมโครคอนโทรลเลอร์ พร้อมโปรแกรมควบคุมและประมวลผล
3. ทดสอบสมรรถนะการทำงานของเครื่องที่พัฒนาขึ้นเปรียบเทียบกับเครื่องมือที่ผลิตมาจากต่างประเทศอันได้แก่ ความเป็นเชิงเส้น ความเร็วในการแปลงผันสัญญาณและเดดไทม์ เป็นต้น

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย

1. ศึกษาค้นคว้าเอกสารและงานวิจัยที่เกี่ยวข้อง
2. ศึกษาการประยุกต์ใช้ไมโครคอนโทรลเลอร์ และ CPLD
3. ออกแบบและสร้างวงจรจ่ายศักดาไฟฟ้าสูง วงจรขยายสัญญาณพัลส์ วงจรแปลงสัญญาณพัลส์พร้อมระบบควบคุมการทำงานและระบบแสดงผล
4. ทดสอบและเปรียบเทียบสมรรถนะการทำงานของเครื่องวิเคราะห์แบบหลายช่องที่พัฒนาขึ้นกับเครื่องมือที่ผลิตจากต่างประเทศ
5. สรุปผลการทดลองและเขียนวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

ได้ต้นแบบเครื่องวิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิคซิลิคอนสันเอ็ดจีเป็นแนวทางในการพัฒนาเครื่องวิเคราะห์แบบหลายช่องขึ้นใช้เองภายในประเทศ

1.6 เอกสารและงานวิจัยที่เกี่ยวข้อง

1. ปี 2536 วิรุฬห์ มังคละวิรัช และ สุวิทย์ ปุณณชัยยะ ภาควิชาวิศวกรรมเทคโนโลยี คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ทำวิจัยเรื่องอุปกรณ์วิเคราะห์แบบหลายช่อง โดยออกแบบระบบแปลงสัญญาณพัลส์อนาล็อกเป็นสัญญาณเชิงตัวเลขแบบวิลคินสัน (Wilkinson) ทำงานด้วยความถี่ฐานเวลา 50 MHz มีความละเอียดในการวิเคราะห์ 1024 ช่องวัดและมีความจุแต่ละช่องวัดเท่ากับ $10^{16}-1$ จำนวนนับ และพัฒนาโปรแกรมควบคุมการส่งข้อมูลดิจิทัลจากระบบแปลงสัญญาณผ่านช่องขยายการทำงาน (Slot) ขนาด 8 บิต บนไมโครคอมพิวเตอร์
2. ปี 2543 Cardoso, J.M.; Amorim, V. ; Bastos, R. ; Madeira, R. ; Basilio Simoes J.; Correia, C.M.B.A ; ได้เสนองานวิจัยใน Nuclear Science Symposium Conference Record โดยวิจัยเกี่ยวกับการลดต้นทุนราคาของอุปกรณ์วิเคราะห์แบบหลายช่อง โดยพัฒนาส่วนประกอบของเครื่องให้ทำงานโดยใช้ไมโครคอนโทรลเลอร์เป็นพื้นฐาน และแหล่งจ่ายไฟฟ้าแบบ Li-ion มีหน่วยความจำที่แยกต่างหากและเก็บสัญญาณพัลส์ได้นานหลายชั่วโมง และยังได้รวม Pulse shape amplifier, Base line restorer, Peak detector และ Logic control circuitry และสัญญาณพัลส์ที่วิเคราะห์ได้ก็จะแสดงออกทางจอแสดงผลชนิด คอท-เมตริก สามารถทำการเชื่อมต่อกับคอมพิวเตอร์ได้ทางพอร์ต USB หรือพอร์ตอนุกรม RS-232
3. ปี 2547 Lima, H.P., Jr. ; Guedes, G.P. ; Barbosa, A.F. และ Seixas, J.M. ได้เสนองานวิจัยใน Instrumentation and Measurement, IEEE โดยพัฒนาอุปกรณ์วิเคราะห์แบบหลายช่องด้วยหลักการแปลงสัญญาณพัลส์อนาล็อกเป็นสัญญาณดิจิทัลแบบรวดเร็ว (fast analog-to-digital) ทำงานร่วมกับ CPLD (Complex Programmable Logic Device) ทำงานด้วยความถี่ฐานเวลา 20 MHz โดยลดเวลา Dead Time ลงเหลือเท่ากับ 200 nsec

บทที่ 2

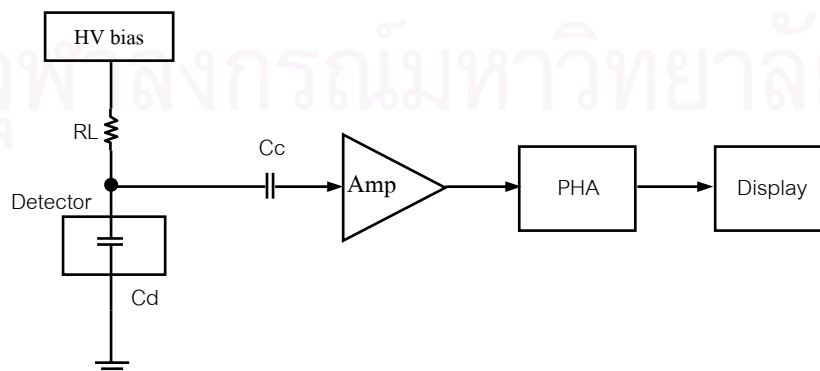
อุปกรณ์วิเคราะห์แบบหลายช่อง

2.1 การวิเคราะห์ความสูงของพัลส์

การวิเคราะห์พลังงานของอนุภาคนิวเคลียร์ที่เกิดจากกระบวนการทางนิวเคลียร์ต่างๆ อันได้แก่ การสลายตัวของสารกัมมันตรังสี (Radioactive Decay) ปฏิกิริยานิวเคลียร์ (Nuclear Reaction) และอันตรกิริยานิวเคลียร์ (Nuclear Interaction) ฯลฯ สามารถทำการวิเคราะห์ได้โดยใช้ อุปกรณ์วิเคราะห์ความสูงของพัลส์ (Pulse Height Analyzer : PHA) ซึ่งจะทำได้ข้อมูลอันนำไปสู่การวิเคราะห์ธาตุในเชิงคุณภาพและปริมาณด้วยเทคนิคทางนิวเคลียร์ โดยค่าพลังงานที่ผ่านเข้ามาในระบบวัดทั้งหมดจะถูกแจกแจงเป็นความถี่ของจำนวนพัลส์บนระนาบของแกนแนวดิ่ง และการแสดงตำแหน่งพลังงานที่สอดคล้องพลังงานในแกนแนวนอนของเส้นกราฟที่สัมพันธ์กับระดับพลังงานที่ปลดปล่อยจากต้นกำเนิดรังสี (dN/dE) หรือ (dN/dH) ที่เรียกว่า “สเปกตรัมนิวเคลียร์” และผลรวมของจำนวนนับอนุภาคนิวเคลียร์บนพื้นที่ใต้พีคพลังงาน (Integral Area) จะบอกสัดส่วนปริมาณของธาตุนั้น ๆ

2.1.1 สัญญาณพัลส์นิวเคลียร์ [2]

กระบวนการตรวจวัดนิวเคลียร์อาศัยหลักการเปลี่ยนพลังงานจลน์ของอนุภาคนิวเคลียร์จากการถ่ายโอนพลังงานแก่ตัวกลางที่เหมาะสมภายในหัววัดรังสีให้กลายเป็นปริมาณประจุไฟฟ้า โดยปริมาณประจุไฟฟ้าจะมากจะน้อยก็จะขึ้นกับระดับพลังงานที่ถ่ายโอนในหัววัดรังสี หลังจากนั้นปริมาณประจุไฟฟ้าจะถูกเปลี่ยนเป็นสัญญาณไฟฟ้าในรูปพัลส์ที่มีความสูงสัมพันธ์กับสัดส่วนของพลังงานอนุภาคนิวเคลียร์นั้น ๆ ดังแสดงในรูปที่ 2.1

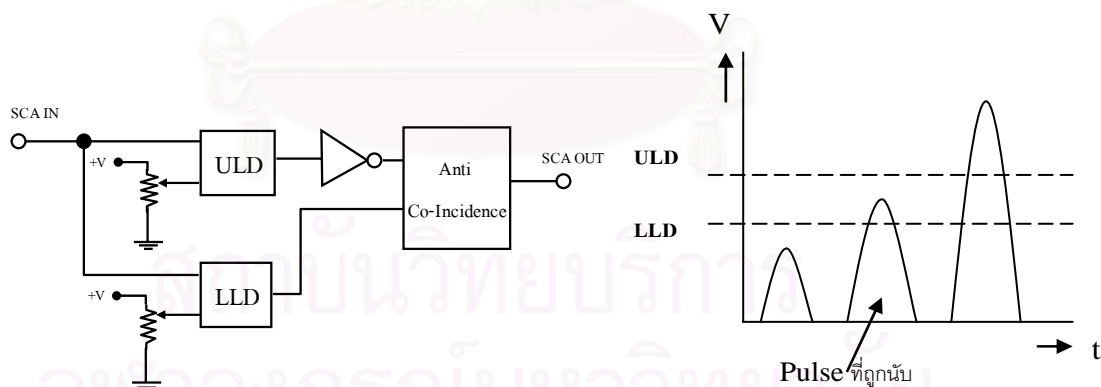


รูปที่ 2.1 แผนภาพของระบบวิเคราะห์พลังงานของอนุภาคนิวเคลียร์

จากระบบวัดนิวเคลียร์ ตามรูปที่ 2.1 เนื่องจากสัญญาณพัลส์จากหัววัดรังสีมีขนาดเล็กมาก จึงต้องส่งต่อไปยังภาคขยายสัญญาณ เพื่อทำการขยายสัญญาณและแต่งรูปสัญญาณด้วยกระบวนการกรองความถี่ที่มีย่านการตอบสนองความถี่แคบ (Narrow Band Amplifier) ตามเทคนิคการวิเคราะห์สเปกตรัม (Spectroscopy) ทำให้ได้อัตราส่วนสัญญาณพัลส์ต่อสิ่งรบกวน Signal to Noise Ratio: S/N สูง สัญญาณพัลส์ที่รับการขยายแล้วจึงมีขนาดความสูงของพัลส์แปรเปลี่ยนตามสัดส่วนพลังงาน (dH/dE) ถูกส่งให้กับระบบวิเคราะห์พลังงานเพื่อจำแนกความสูงของพัลส์ เราเรียกกระบวนการวิเคราะห์พลังงานของอนุภาคนิวเคลียร์นี้ว่า “การวิเคราะห์ความสูงของพัลส์ (pulse height analysis)”

2.1.2 การวิเคราะห์ความสูงพัลส์แบบช่องเดียว (Single Channel Analyzer: SCA)

เมื่อสัญญาณพัลส์จากภาคขยายของระบบวัดนิวเคลียร์ที่มีลักษณะเป็นสัญญาณอนาล็อก ถูกป้อนให้กับอุปกรณ์วิเคราะห์ความสูงของพัลส์แบบช่องวิเคราะห์สัญญาณเดี่ยว ดังแสดงในรูปที่ 2.2 ก. การจำแนกความสูงพัลส์ทำได้โดยการเปรียบเทียบความสูงของพัลส์กับระดับอ้างอิง 2 ระดับ ด้วยวงจรเปรียบเทียบศักดาไฟฟ้า วงจรเปรียบเทียบศักดาไฟฟ้าที่มีศักดาอ้างอิงระดับล่างเรียกว่า “Lower Level Discriminator: LLD” และวงจรเปรียบเทียบศักดาไฟฟ้าที่มีศักดาอ้างอิงระดับบนเรียกว่า “Upper Level Discriminator: ULD” ผลต่างของศักดาอ้างอิงไฟฟ้าทั้งสอง เรียกว่า “หน้าต่างวิเคราะห์พลังงาน (Energy Window)”



ก. ส่วนประกอบหลักของ SCA

ข. การวิเคราะห์พัลส์ด้วย SCA

รูปที่ 2.2 การวิเคราะห์ความสูงของพัลส์ด้วยอุปกรณ์วิเคราะห์ช่องเดียว

การวิเคราะห์ความสูงของพัลส์ตลอดย่านวิเคราะห์ด้วยอุปกรณ์วิเคราะห์ช่องเดียวทำได้โดยการเลื่อนช่องวิเคราะห์ (ΔE) ครั้งละ 1 ช่องต่อเนื่องกัน โดยตั้งเวลานับเท่าๆ กัน ข้อมูลจากการ

บันทึกผลการวัดแบบแยกนับเฉพาะความสูงของพัลส์ เมื่อสร้างเป็นเส้นกราฟความสัมพันธ์ระหว่าง dN/dE จะได้สเปกตรัมจากรูปที่ 2.2 ข. อย่างไรก็ตามการวิเคราะห์ความสูงของพัลส์แบบช่องเดี่ยวนี้มีความไม่สะดวกทั้งด้านวิธีการวิเคราะห์และการแสดงผลเพราะใช้เวลาในการวิเคราะห์มาก ทำให้มีข้อจำกัดในการวิเคราะห์ความสูงของพัลส์แบบต่อเนื่อง ซึ่งต่อมาได้มีการพัฒนาวิธีวิเคราะห์ความสูงของพัลส์แบบต่อเนื่องโดยอาศัยเทคนิคทางดิจิทัล (Digital Technique) เข้ามาช่วยทำให้สามารถวิเคราะห์ความสูงของพัลส์ได้หลายช่องวิเคราะห์ในเวลาเดียวกัน เสมือนการตั้งช่องการวิเคราะห์พลังงานเดี่ยวต่อเนื่องกันและมีอุปกรณ์บันทึกข้อมูลประจำช่องวิเคราะห์ความสูงของพัลส์ทำงานพร้อมกันแสดงผลได้ทันทีหลังสิ้นสุดเวลานับ ซึ่งก็คืออุปกรณ์วิเคราะห์แบบหลายช่อง

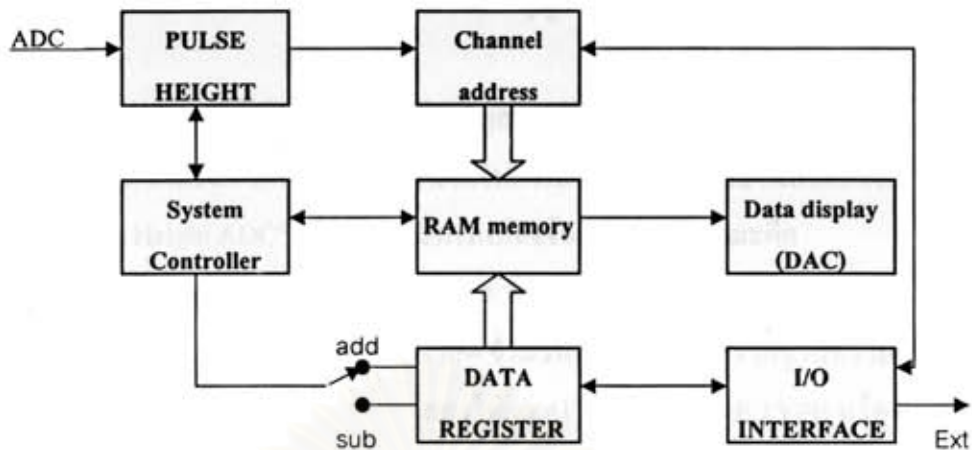
2.2 อุปกรณ์วิเคราะห์แบบหลายช่อง (Multichannel Analyzer: MCA)

อุปกรณ์วิเคราะห์แบบหลายช่องเป็นอุปกรณ์ในการวิเคราะห์ความสูงของพัลส์ที่มีขีดความสามารถในการวิเคราะห์สูง โดยการรับข้อมูลความสูงของพัลส์มาวิเคราะห์ในระบบดิจิทัลอย่างต่อเนื่องและถูกเก็บไว้ในหน่วยความจำ ข้อมูลที่เก็บอยู่ในหน่วยความจำนี้จะถูกสะสมในรูปแบบของการกระจายข้อมูลของสัญญาณพัลส์ (Pulse Height Distribution) และแสดงผลสเปกตรัมทางจอภาพได้ทันทีเมื่อต้องการ

2.2.1 โครงสร้างพื้นฐานของอุปกรณ์วิเคราะห์แบบหลายช่อง

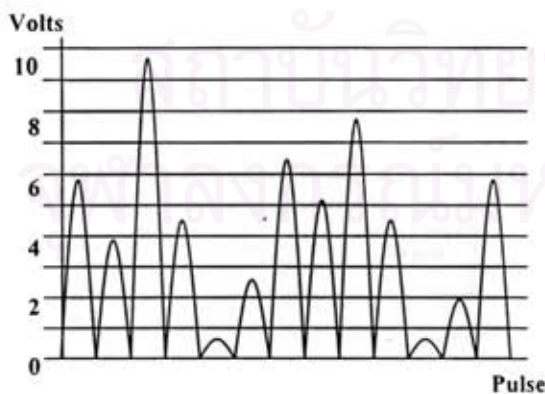
อุปกรณ์วิเคราะห์แบบหลายช่องมีส่วนประกอบที่สำคัญในการสร้างกระบวนการวิเคราะห์ทั้งหมด 7 ส่วนดังแสดงในรูปที่ 2.3 ซึ่งได้แก่

- วงจรแปลงความสูงของสัญญาณพัลส์เป็นสัญญาณดิจิทัล (Pulse Height To Digital Converter)
- หน่วยความจำ (Memory : RAM, ROM)
- วงจรกำหนดแอดเดรสของหน่วยความจำ (Channel Address Register)
- วงจรบันทึกข้อมูล (Data Register)
- วงจรแสดงผล (Data Display)
- วงจรควบคุมการทำงานของระบบ (System Controller)
- วงจรเชื่อมโยงกับอุปกรณ์ภายนอก (I/O Interface)

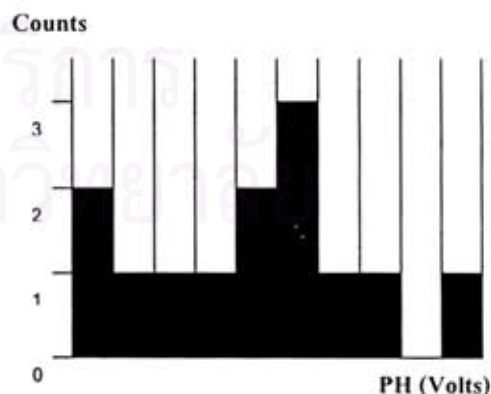


รูปที่ 2.3 แผนภาพการทำงานเบื้องต้นของ MCA

อุปกรณ์วิเคราะห์แบบหลายช่องมีการทำงานด้วยระบบดิจิทัลที่ค่อนข้างซับซ้อน เริ่มจากวงจรแปลงความสูงของพัลส์จะรับพัลส์ศักดาไฟฟ้าดังแสดงในรูปที่ 2.4 ก. จากระบบวัดรังสีมาทำการแปลงเป็นข้อมูลดิจิทัลที่สอดคล้องกับความสูงของพัลส์ ข้อมูลนี้จะทำหน้าที่กำหนดแอดเดรสของหน่วยความจำ และทุกครั้งที่การแปลงข้อมูลความสูงของพัลส์ถูกโคตรงกับตำแหน่งของแอดเดรสในหน่วยความจำนั้น จะมีการบันทึกข้อมูลด้วยการบวกเพิ่มข้อมูลสะสม ณ ตำแหน่งนั้น และข้อมูลจำแนกการกระจายความสูงของพัลส์ที่ถูกเก็บในหน่วยความจำจะมีลักษณะเป็นแบบฮิสโตแกรม (Histogram) และถูกนำไปแสดงผลทางจอภาพด้วยการแปลงข้อมูลเชิงตัวเลข เป็นสัญญาณควบคุมระนาบแกนแนวนอนและแกนแนวตั้งของระบบจอภาพ ดังแสดงในรูปที่ 2.4 ข. ขั้นตอนการทำงานของระบบจะถูกควบคุมโดยโปรแกรมในหน่วยความจำดาวรผ่านวงจรควบคุมการทำงานของระบบ



Input voltage pulse



PHA Histogram

ก. สัญญาณพัลส์จากระบบวัดรังสี

ข. ฮิสโตแกรมความสูงของพัลส์

รูปที่ 2.4 การแสดงสเปกตรัมนิวเคลียร์ของ MCA

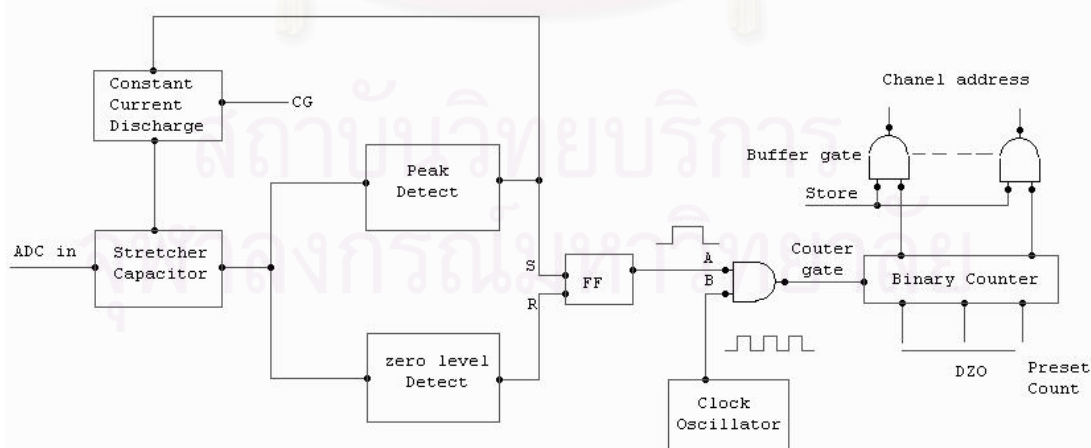
รูปที่ 2.4 การแสดงสเปกตรัมนิวเคลียร์ของ MCA

2.2.2 การแปลงความสูงของพัลส์เป็นสัญญาณดิจิทัล

การแปลงความสูงของพัลส์เป็นสัญญาณดิจิทัลเป็นส่วนสำคัญส่วนหน้าของอุปกรณ์วิเคราะห์แบบหลายช่อง โดยวงจรที่ทำหน้าที่ในการแปลงความสูงของพัลส์เป็นสัญญาณดิจิทัลที่เรียกว่า “Pulse Height ADC” ปัจจุบันมีวิธีการแปลงข้อมูลใน 2 ลักษณะคือ

- CONVENTIONAL ADC คือการแปลงความสูงของพัลส์ให้เป็นเวลาสำหรับใช้ในการเปิดและปิดเกต จำนวนสัญญาณพัลส์จากฐานความถี่ซึ่งจะให้ความเป็นเชิงเส้นของข้อมูลสูงได้แก่ ADC แบบ RAMP TYPE หรือ WILLKINSON
- CONSTANT FRACTION ADC คือการแปลงความสูงของพัลส์เป็นรหัสดิจิทัลที่มีเวลาในการแปลงคงที่ และมีความไวในการทำงานสูงแต่มีความเป็นเชิงเส้นที่ต้องอาศัยการปรับแก้ ได้แก่ ADC แบบ SUCCESSIVE APPROXIMATION และ FLASH TYPE เป็นต้น

ในการออกแบบวงจรแปลงความสูงของพัลส์ที่ต้องการความเป็นเชิงเส้นในการแปลงข้อมูลดิจิทัลโดยไม่ต้องปรับแก้ความคลาดเคลื่อนจะเลือกการทำงานแบบวิลคินสัน ซึ่งในระยะแรกทำงานที่ฐานความถี่ 10 MHz ปัจจุบันสามารถทำงานด้วยความถี่ฐานเวลาสูงถึง 400 MHz ระบบการแปลงสัญญาณพัลส์มีหลักการการทำงานดังแสดงในรูปที่ 2.5 ส่วนรูปที่ 2.6 แสดงให้เห็นถึงวิธีการแปลงความสูงของพัลส์ที่มีขนาดต่างกันให้เป็นข้อมูลรหัสไบนารี

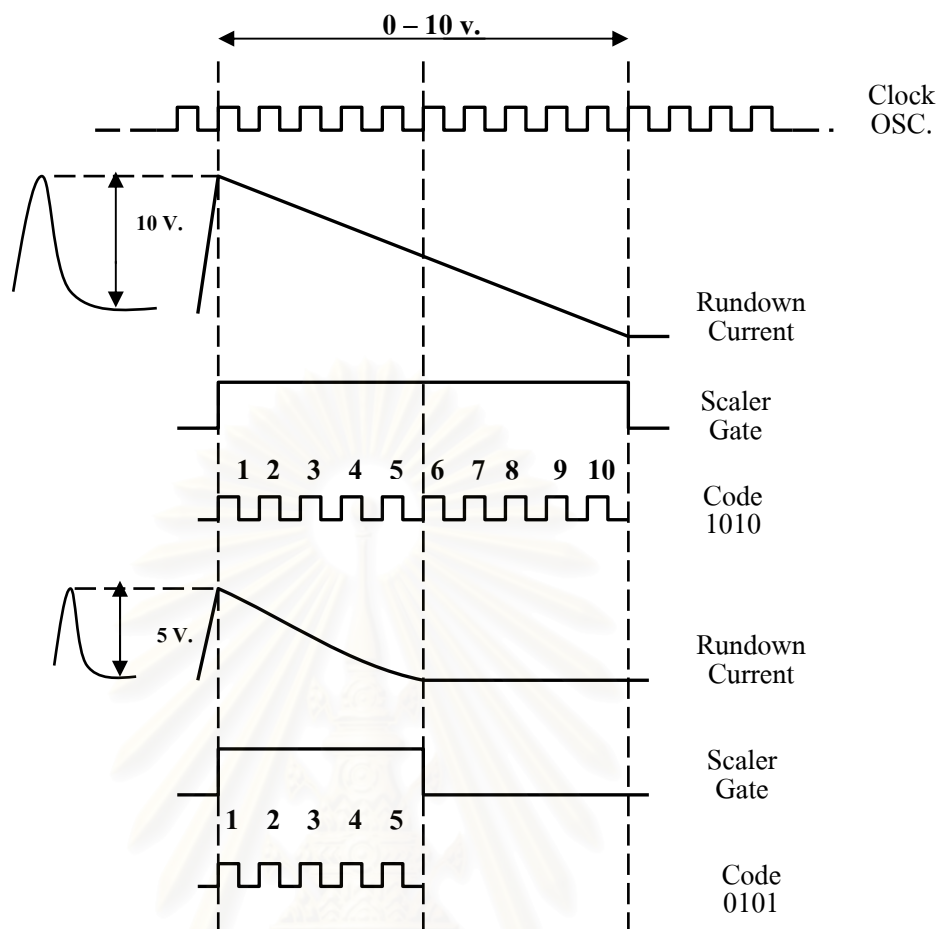


รูปที่ 2.5 แผนภาพหลักการทำงานของวงจรแปลงสัญญาณแบบวิลคินสัน

เมื่อคาปาซิเตอร์ในแผนภาพถูกควบคุมให้รับความสูงของพัลส์ที่ตำแหน่งสูงสุด (Peak) เท่ากับ V_c และเริ่มคายประจุด้วยค่าเวลาคงที่ RC สมมูล ผ่านตัวต้านทานตำแหน่งของสัญญาณ V_0 ที่ตัดเส้นศูนย์จะเป็นสัดส่วนระหว่างความสูงของพัลส์ ช่วงเวลาที่คายประจุนี้เรียกว่า “Run Down Time” หรือเวลาการแปลงผัน (Conversion Time)” พิจารณาการทำงานของวงจรรูปที่ 2.5 มีขั้นตอนการทำงานดังนี้

1. Clock Oscillator เป็นต้นกำเนิดความถี่ฐานเวลาคงที่รออยู่ที่ Counter Gate ขา B ตลอดเวลา
2. Stretcher Capacitor คาปาซิเตอร์จะทำหน้าที่เก็บประจุจนได้ระดับสูงสุดของพัลส์และจะคายประจุด้วยแหล่งจ่ายกระแสคงที่ โดยระดับของพัลส์ที่เข้ามาจะถูกตรวจสอบด้วยวงจร Peak Detect ว่ามีระดับสูงสุด
3. Peak Detect วงจรนี้จะทำหน้าที่ตรวจสอบระดับสูงสุดของการประจุสัญญาณพัลส์ที่คาปาซิเตอร์และควบคุมการทำงานใน 2 ส่วน คือสั่งให้สวิทช์อิเล็กทรอนิกส์คายประจุผ่านวงจรแหล่งจ่ายกระแสคงที่ทำงานและพร้อมกันสั่งให้ฟลิป-ฟลอป (Flip-Flop) เปิดเกตรระหว่างช่วงเวลานี้ Counter Gate จะถูกเปิดและวงจรมอนิเตอร์จะเริ่มนับความถี่ฐานเวลา
4. Zero Level Detect วงจรนี้จะทำหน้าที่ตรวจสอบระดับการคายประจุของคาปาซิเตอร์เมื่อถึงเส้นศูนย์ของสัญญาณจะกระตุ้นฟลิปฟลอปให้เปิดเกตรวงจรมอนิเตอร์จะหยุดนับและค้างข้อมูลสูงสุดที่นับได้ไว้เป็นตำแหน่งแอดเดรสของหน่วยความจำ

หลังจากขั้นตอนการทำงานของวงจร Zero Level Detect วงจรควบคุมการทำงานของอุปกรณ์วิเคราะห์แบบหลายช่องจะให้กำเนิดสัญญาณ Store เพื่อเปิด Buffer Gate และบอกหน่วยความจำให้ทำการบันทึกข้อมูล ณ ตำแหน่งความสูงของพัลส์ที่ประมวลได้ เมื่อสิ้นสุดการบันทึกข้อมูลจะส่งสัญญาณ Clear มาให้วงจรแปลงความสูงพัลส์เริ่มต้นการทำงานใหม่ ถ้าขนาดความสูงของพัลส์ต่างกัน Conversion Time ก็จะต่างกัน ดังแผนภาพเวลาการแปลงสัญญาณขนาดความสูง 5 โวลต์ และ 10 โวลต์ที่ได้เปรียบเทียบไว้ดังในรูปที่ 2.6 โดยความสูงที่แตกต่างกันของพัลส์แต่ละลูกก็จะได้ขนาดของความกว้างในการ Stretcher ต่างกัน ความกว้างที่ได้จะเทียบเป็นค่าของเวลา จากนั้นก็จะนำไปควบคุมการเปิดเกตรความถี่ที่มาจากตัวกำเนิดความถี่ แล้วนำความถี่ที่ได้ไปเข้าในวงจรแปลงรหัสเป็นเลขฐานกำหนดเป็นแอดเดรสของแต่ละช่องสัญญาณตามความสูงพัลส์



รูปที่ 2.6 เปรียบเทียบการแปลงความสูงของพัลส์เป็นรหัสไบนารี

2.3 รูปแบบของอุปกรณ์วิเคราะห์แบบหลายช่อง

วิวัฒนาการของอุปกรณ์วิเคราะห์แบบหลายช่องในปัจจุบันได้ถูกปรับเปลี่ยนตามความก้าวหน้าของเทคโนโลยีคอมพิวเตอร์ซึ่งเดี่ยวกั้นกระบวนการผลิตชิ้นส่วนอิเล็กทรอนิกส์ไมโครได้พัฒนาการผลิตวงจรรวม (Integrated Circuit) ทำให้อุปกรณ์อิเล็กทรอนิกส์ในปัจจุบันมีขนาดเล็กและมีประสิทธิภาพสูงมากต่างจากในอดีต ทำให้รูปแบบเครื่องวิเคราะห์แบบหลายช่องมีรูปแบบที่ทันสมัย ประสิทธิภาพในการทำงานสูง และสะดวกต่อในการใช้งาน โดยสามารถแบ่งรูปแบบและโครงสร้างของระบบได้เป็น 2 ประเภทคือ

1. Stand Alone MCA เป็นอุปกรณ์วิเคราะห์ที่ออกแบบให้มีระบบการทำงานอิสระด้วยตัวเองโดด ๆ เฉพาะหน้าที่ในการวิเคราะห์ความสูงของพัลส์ แบ่งได้เป็น
 - แบบวงจรอิเล็กทรอนิกส์ล้วน
 - แบบควบคุมการทำงานด้วยไมโครโปรเซสเซอร์

2. MCA Card มีลักษณะเป็นแผ่นวงจรที่ต้องพึ่งช่องขยายการทำงานบนไมโครคอมพิวเตอร์ และมีโปรแกรมสำเร็จรูปในการทำงานเป็นอุปกรณ์วิเคราะห์แบบหลายช่อง
3. Pocket MCA เป็นอุปกรณ์ที่ออกแบบให้มีขนาดเล็ก การแสดงผลจะต้องแสดงที่ไมโครคอมพิวเตอร์และใช้โปรแกรมสำเร็จรูปในการควบคุมการทำงาน
4. MCA Tube Base เป็นอุปกรณ์วิเคราะห์แบบหลายช่องที่ออกแบบให้ทำหน้าที่เป็นทั้ง Tube Base และ MCA โดยต่อเข้ากับหัววัดรังสีแทน Tube Base และมีโปรแกรมสำเร็จรูปในการทำงานร่วมกับไมโครคอมพิวเตอร์

2.4 การออกแบบวงจรดิจิทัลด้วย FPGA หรือ CPLD [5]

การออกแบบวงจรดิจิทัลที่มีขนาดเล็กๆ โดยปกติจะนิยมใช้ไอซีหรือชิพมาตรฐานตระกูล CMOS หรือ TTL โดยจะเห็นได้ว่าไอซีสำเร็จรูปเหล่านี้จะมีฟังก์ชันทำงานทางลอจิกแบบตายตัว และเป็นวงจรมีขนาดเล็กๆ อยู่ภายในเพียงไม่กี่ตัว จึงไม่เหมาะกับงานออกแบบวงจรมีขนาดใหญ่หรือวงจรความถี่สูง เนื่องจากเกิดเวลาหน่วงหรือล่าช้าในตัวไอซี และสายสัญญาณที่ต่อระหว่างตัวไอซี ทำให้วงจรที่ต้องใช้ไอซีดังกล่าวหลายๆ ตัวทำงานได้ช้าและแผงวงจรมีขนาดใหญ่

ปัจจุบันมีการออกแบบวงจรมีขนาดใหญ่โดยชิพดิจิทัลเอนกประสงค์แทนการออกแบบด้วยไอซีมาตรฐาน CMOS และ TTL นักออกแบบดิจิทัลที่ประสบปัญหาในอดีตเรื่องการซื้อไอซีไม่ได้ตรงตามต้องการหรือคิดจะสร้างไอซีขึ้นเองเพื่อใช้เฉพาะงานซึ่งเป็นไอซีที่หาซื้อตามท้องตลาดไม่ได้ จึงหันมานิยมใช้ชิพดิจิทัลเอนกประสงค์ ซึ่งนับว่าเป็นทางเลือกใหม่ อุปกรณ์หรือเครื่องมือที่ทันสมัยทางด้านการศึกษา การแพทย์ การทหาร ระบบเครือข่ายหรือเครื่องมือวัดต่างๆ จากต่างประเทศนั้น บ่อยครั้งเราจะพบว่า มีชิพ FPGA (Field Programmable Gate Array) หรือ CPLD (Complex Programmable Logic Device) เป็นส่วนประกอบที่อยู่บนแผงวงจรด้วยเสมอ

การที่เราออกแบบวงจรร้อยส่วนต่างๆ รวมไว้ใน FPGA หรือ CPLD เพียงตัวเดียวหรือเพียงไม่กี่ตัวนั้นจะทำให้แผงวงจรมีขนาดเล็กลงอย่างมาก ทำให้ได้วงจรทำงานได้เร็วขึ้นเพราะสายสัญญาณต่างๆ สั้นลงและสัญญาณส่วนใหญ่จะอยู่ภายในชิพทำให้ได้ผลิตภัณฑ์ที่มีขนาดเล็กกระทัดรัด

2.4.1 คุณสมบัติพื้นฐานของ CPLD

CPLD เป็นไอซีประเภท PLD (Programmable Logic Device) ซึ่งเป็นชิพดิจิทัลเอนกประสงค์ชนิดหนึ่งที่สามารถโปรแกรมให้มีฟังก์ชันการทำงานตามที่ต้องการได้ ตัวอย่าง CPLD เบอร์ XC9536XL ที่มีความจุวงจร 800 เกตและมี 34 อินพุตเอาต์พุต (I/O)

จะเห็นได้ว่าโครงสร้าง CPLD ประกอบด้วยส่วนหลักๆ คือ Function Block (FB) และ I/O Block (IOB) ที่สามารถเชื่อมต่อถึงกันด้วย Matrix Switch โดยที่ภายใน Function Block แต่ละชุดจะประกอบด้วยวงจรลอจิกพื้นฐานต่างๆ ที่สามารถโปรแกรมเป็นวงจรดิจิทัลได้ตามต้องการ ส่วน I/O Block จะทำหน้าที่เป็นบัฟเฟอร์ที่อินพุตหรือเอาต์พุตของตัวชิพ วงจร In-System Programming ใช้สำหรับโปรแกรมชิพผ่านทางพอร์ต JTAG โดยมี JTAG Controller เป็นตัวควบคุมชิพ CPLD เพียงตัวเดียวอาจสร้างวงจรดิจิทัลแทนที่วงจรเดิมที่เคยใช้ไอซีมาตรฐานตระกูล 7400 นับสิบๆ ตัวก็ได้ แม้ว่าจะสามารถสร้างวงจรดิจิทัลต่างๆ ไปได้มากมาย แต่ต้องถือว่า CPLD นั้นมีความจุต่ำมากเมื่อเทียบกับชิพ FPGA โดยทั่วไป CPLD จะมีความจุวงจรไม่เกิน 10,000 เกต และเนื่องจากข้อจำกัดของ CPLD ในการโปรแกรมจะใช้เวลาก่อนข้างนานเนื่องจากมีโครงสร้างตัวเก็บข้อมูลภายในเป็นพวก PROM ซึ่งเขียนข้อมูลลงไปได้ช้า แต่เมื่อโปรแกรมวงจรไว้ใน CPLD แล้วข้อมูลวงจรมันจะคงอยู่แม้ว่าจะไม่มีไฟเลี้ยงตัวชิพแล้วก็ตาม การโปรแกรมสามารถทำซ้ำได้หลายๆ ครั้ง

2.4.2 คุณสมบัติพื้นฐานของ FPGA

FPGA เป็นชิพประเภท PLD เช่นเดียวกัน จึงเป็นชิพอเนกประสงค์ที่สามารถโปรแกรมให้เป็นวงจรดิจิทัลได้ตามต้องการ แต่จะมีโครงสร้างภายในแตกต่างจาก CPLD อย่างสิ้นเชิงและซับซ้อนกว่ามาก

โครงสร้างภายในของ FPGA ประกอบด้วย 2 ส่วนหลักๆ คือ Configurable Logic Block (CLB) และ I/O Logic เพื่อใช้โปรแกรมให้เป็นวงจรดิจิทัลตามที่ต้องการและมีการเชื่อมต่อถึงกันภายในชิพมีหลากหลายลำดับชั้น นอกจากนี้ยังมีอุปกรณ์ภายในที่ทำหน้าที่เฉพาะงาน เช่น Delay-Locked Loop (DLL) และหน่วยความจำ (BLOCK RAM) เพื่ออำนวยความสะดวกในการออกแบบและเพิ่มสมรรถนะให้วงจร

FPGA มีความจุของเกตค่อนข้างสูงตั้งแต่ระดับประมาณ 10,000 ถึงประมาณ 10,000,000 เกต ขึ้นอยู่กับเทคโนโลยีที่ใช้ในการผลิต การโปรแกรม FPGA สามารถทำได้โดยโหลดข้อมูลวงจร (Configuration Data) ลงไปที่เซลล์หน่วยความจำแบบแรม (RAM) เป็นคนละส่วนกับ BLOCK RAM ที่อยู่ภายใน FPGA ดังนั้น FPGA จึงไม่มีปัญหาเรื่องการโปรแกรมซ้ำและโปรแกรมข้อมูลวงจรลงไปได้เร็ว แต่ RAM มีข้อเสียที่สำคัญคือข้อมูลวงจรจะสูญหายหากไม่มีไฟเลี้ยง จึงต้องใช้หน่วยความจำภายนอกชิพที่สามารถเก็บข้อมูลวงจรไว้ได้แม้ว่าไม่มีไฟเลี้ยงตัวชิพ

2.4.3 การออกแบบวงจรในไอซี CPLD หรือ FPGA

ในกระบวนการออกแบบวงจร โดยทั่วไปมักจะมียุ่ขั้นตอนการทำงานดังนี้

- System Requirement
- System Design

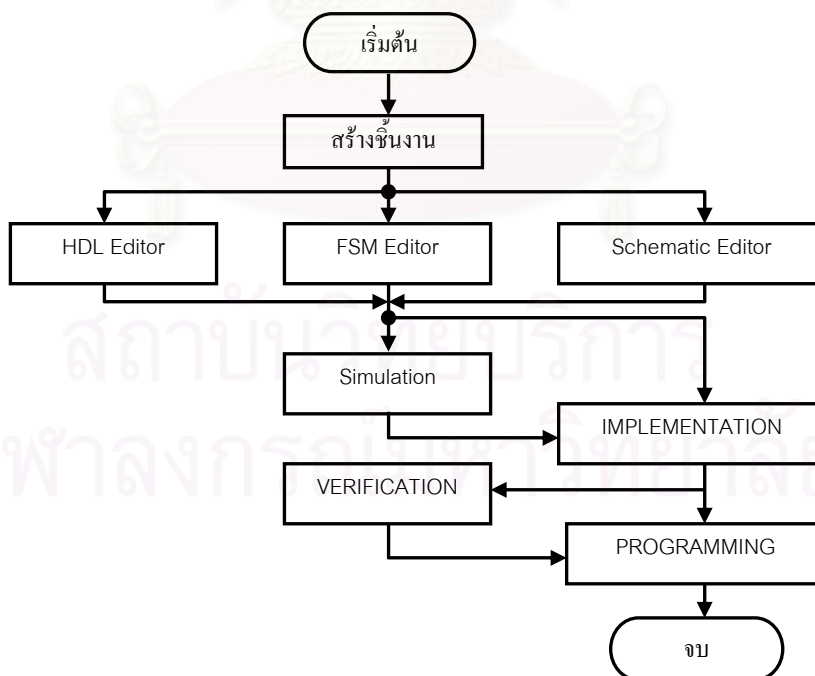
- System Implementation
- Testing And Debugging
- Documentation

กระบวนการทำ System Requirement คือ การหาความต้องการว่าเราต้องการสร้างวงจรอะไร วงจรทำงานอย่างไรและมีอินพุตเอาต์พุตอะไรบ้าง กระบวนการทำ System Design คือการออกแบบและหาวิธีแก้ไขปัญหามาจากโจทย์ที่ได้มาในขั้นตอน System Requirement จากนั้นจึงเป็นการทำ System Implementation เพื่อสร้างวงจรตามที่ได้ออกแบบไว้ เมื่อได้วงจรที่ต้องการแล้วจึงทำ Testing and Debugging เพื่อทำการทดสอบและแก้ไขวงจรในกรณีที่มีปัญหา จากนั้นจึงทำ Documentation เพื่อสร้างเอกสารอธิบายการทำงานของวงจรที่ออกแบบทั้งหมด

ในการออกแบบวงจรดิจิทัลด้วย FPGA หรือ CPLD เริ่มจากขั้นตอนการทำ System Implementation จนถึง Testing and Debugging นั้นจะต้องใช้ซอฟต์แวร์ (Software Tool) ช่วยในการออกแบบ

2.4.4 ขั้นตอนการออกแบบในไอซี CPLD หรือ FPGA

ขั้นตอนการออกแบบวงจรดิจิทัลด้วย FPGA หรือ CPLD มีการทำงานหลายขั้นตอน แสดงรายละเอียดดังรูป 2.7



รูปที่ 2.7 ขั้นตอนการออกแบบวงจรดิจิทัลด้วย FPGA หรือ CPLD

จากรูปที่ 2.7 จะเห็นว่าขั้นตอนการออกแบบวงจรดิจิทัลประกอบด้วย

- Design Entry
- Design Synthesis
- Design Verification
- Design Implementation
- Device Programming

1. Design Entry

คือการนำวงจรที่ออกแบบวงจรมาเขียนด้วยวิธีการต่างๆ เช่น ออกแบบด้วยวิธีวาดผังวงจร (Schematic) หรือออกแบบด้วยภาษาระดับสูงที่เรียกว่า HDL (Hardware Description Language) เช่น ภาษา VHDL หรือ Verilog หรืออาจออกแบบวงจรด้วยวิธีการเขียนแผนภูมิสถานะ (State diagram) เพื่อให้ได้วงจรตามต้องการ

2. Design Synthesis

คือการสังเคราะห์วงจรโดยการแปลงโค้ดที่ออกแบบวงจรด้วยภาษาระดับสูงที่เรียกว่า ภาษา HDL (Hardware Description Language) ให้เป็นวงจรในระดับเกต (Gate-level)

3. Design Verification

คือการตรวจสอบความถูกต้องของวงจรที่ออกแบบ เป็นการนำโค้ด HDL ของวงจรที่ออกแบบไปทำการตรวจสอบความถูกต้องด้วยโปรแกรมจำลองการทำงาน (Simulation)

4. Design Implementation

คือการเริ่มจากขั้นตอนการแปล (Translate) โดยนำไฟล์ Design Netlist มาทำการถอดไม่ซ่วงจรก่อน จากนั้นจะนำไปทำการตรวจสอบว่าสามารถวางหรือบรรจุในชิพเบอร์ที่เรา กำหนดได้หรือไม่ ขั้นตอนต่อไปเป็นการแมพ (Map) โดยเลือกอุปกรณ์จากไฟล์ของวงจรเข้าไปวางในอุปกรณ์พื้นฐานต่างๆ ที่อยู่ภายใน FPGA หรือ CPLD เบอร์ที่เรากำหนด

5. Device Programming

คือการโปรแกรมข้อมูลวงจรลงบนชิพ ทำได้โดยการนำไฟล์นามสกุล .bit หรือไฟล์นามสกุล .jed ที่ได้ในขั้นตอนการ Implementation มาดาวน์โหลดลงชิพโดยใช้เครื่องโปรแกรม หรือใช้ดาวน์โหลดเคเบิล

2.5 ไมโครคอนโทรลเลอร์ ARM7 [4]

ในยุคแรกชิพ ARM เป็นชิพ RISC (Reduced Instruction Set Computer) ขนาด 32 บิต การทำงานจำเป็นจะต้องต่อกับหน่วยความจำและอุปกรณ์ภายนอก เมื่อมีบริษัทผู้ผลิต

ไมโครคอนโทรลเลอร์จำนวนมากได้นำลักษณะของซีพียู ARM ไปพัฒนาต่อได้มีการเพิ่มหน่วยความจำภายในทั้ง ROM และ RAM และเพิ่มโมดูลอุปกรณ์เสริมต่างๆ เช่น วงจรสื่อสารแบบอนุกรมวงจรแปลงดิจิทัลเป็นอนาล็อก เป็นต้น ทำให้กลายเป็นไมโครคอนโทรลเลอร์แบบ 32 บิตที่กินพลังงานต่ำ สามารถทำงานได้โดยใช้ชิพไอซีเพียงตัวเดียวโดยไม่ต้องต่ออุปกรณ์เพิ่มเติมภายนอก

ไมโครคอนโทรลเลอร์ ARM7 จะมีหน่วยความจำภายในขนาด 8 kB ถึง 40 kB ส่วนหน่วยความจำโปรแกรมจะมีขนาด 32 kB ถึง 512 kB อยู่ภายในชิพส่วน การประมวลผลเป็นแบบ 16/32 บิต แล้วยังมีข้อดีในเรื่องสถาปัตยกรรมที่ไม่ซับซ้อน ทำให้ประหยัดพื้นที่ในการผลิตชิพและชิพมีขนาดเล็กกินพลังงานน้อย โดยที่ยังคงมี สมรรถนะที่สูง

สถาปัตยกรรมของ ARM7 จะเป็นแบบ Load-And-Store ภายในมีบัสขนาด 32 บิตตัวเดียวที่ใช้สำหรับรับส่งข้อมูล และคำสั่ง ชุดคำสั่งจะมีขนาด 32 บิตคงที่ในขณะที่ข้อมูลสามารถเลือกได้ว่าจะมีขนาด 8, 16 หรือ 32 บิต โครงสร้างของ ARM7 จะเป็นแบบที่เรียบง่าย มีชุดคำสั่งไม่มากนัก ประหยัดพื้นที่ สารกึ่งตัวนำที่ใช้สร้างประหยัดพลังงาน ในการประมวลผลข้อมูลใดๆต้องกระทำผ่านทางรีจิสเตอร์เริ่มต้นด้วยการโหลดค่าจากหน่วยความจำเก็บในรีจิสเตอร์นำค่ามาประมวลผลเสร็จแล้วจะเขียนค่าเก็บในหน่วยความจำดั้งเดิม

รีจิสเตอร์ของ ARM7 ที่ใช้งานได้สำหรับผู้ใช้งานทั้งหมด 16 ตัวคือ R0-R15 โดยทุกตัวมีขนาด 32 บิต โดย R0-R12 เป็นรีจิสเตอร์ทั่วไปที่ไม่ได้กำหนดหน้าที่การทำงานพิเศษ ส่วน R13 ทำหน้าที่เป็น Stack Pointer (SP) R14 ทำหน้าที่เป็น Link Register (LR) และ R15 ทำหน้าที่เป็น Program Counter (PC)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3

การดำเนินการวิจัย

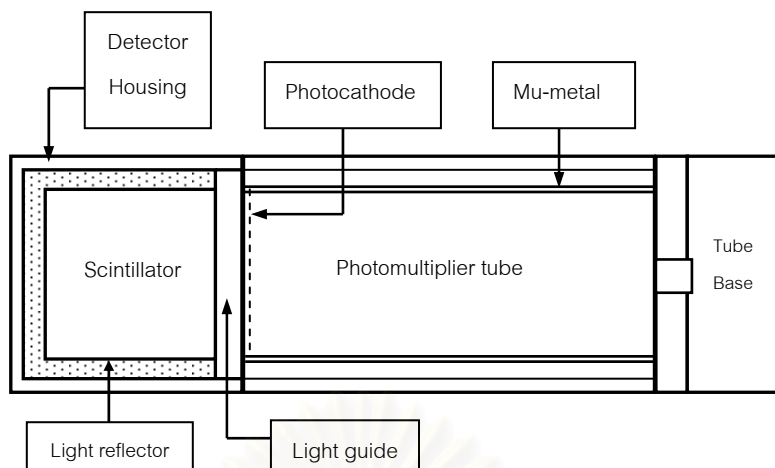
3.1 ข้อมูลพื้นฐานของระบบ

อุปกรณ์วิเคราะห์แบบหลายช่องที่ทำการพัฒนาขึ้นนี้เป็นแบบ Stand Alone ที่ใช้เทคนิคของวิลคินสัน การพัฒนาจะออกแบบให้อุปกรณ์วิเคราะห์แบบหลายช่องมีขนาดเล็กกะทัดรัดและมีขนาดช่องวิเคราะห์ไม่ต่ำกว่า 2048 ช่อง ที่ความถี่ 100 MHz การทำงานภายในตัวของอุปกรณ์วิเคราะห์แบบหลายช่องออกแบบให้ CPLD ทำงานร่วมกับไมโครคอนโทรลเลอร์ โดยในส่วนของ CPLD จะทำหน้าที่แปลงความสูงของพัลส์ให้เป็นสัญญาณดิจิทัลโดยมีตำแหน่งจากช่อง 0 ถึงช่องที่ 2048 เพื่อเตรียมที่จะเก็บลงในหน่วยความจำ ในส่วนของไมโครคอนโทรลเลอร์จะทำหน้าที่กำหนดช่วงเวลาที่จะเก็บค่าตำแหน่งที่แปลงมาจาก CPLD มาเก็บในหน่วยความจำหลังจากนั้นก็ทำการประมวลผลและแสดงผลป้อนออกทางจอแสดงผล

3.1.1 ข้อมูลพื้นฐานสำหรับหัววัดรังสี [6]

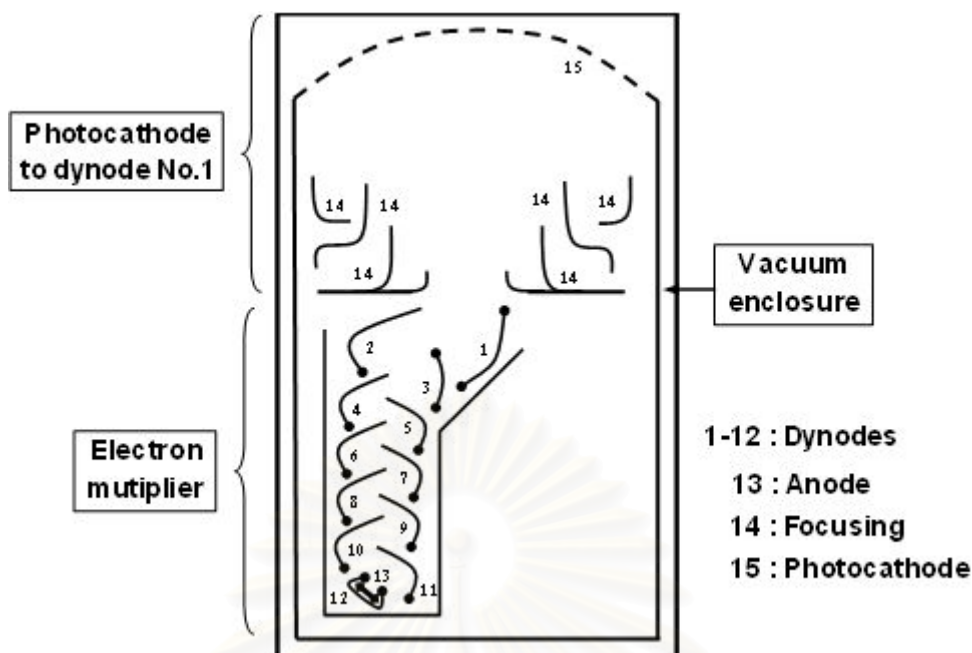
หัววัดรังสีสำหรับรังสีแกมมาที่ใช้ทดลองเป็นชนิดซินทิลเลชัน รุ่น 2M2/2PR ยี่ห้อ BICRON เป็นหัววัดที่มีกระบวนการวัดรังสีทางอ้อมมีโครงสร้างที่ประกอบด้วยส่วนสำคัญ 2 ส่วน คือ ส่วนที่รับอันตรกิริยาและเกิดประกายแสง (Scintillator) ต่อเชื่อมกับส่วนที่ทำหน้าที่เปลี่ยนประกายแสงเป็นอิเล็กตรอนและทวีปริมาณอิเล็กตรอน (Photomultiplier tube, PMT) ซึ่งถูกบรรจุไว้ด้วยภาชนะปิดป้องกันแสงจากภายนอกและบริเวณหลอดทวีคูณอิเล็กตรอน (PMT) จะห่อหุ้มด้วยแผ่นกันสนามแม่เหล็ก (Mu-Metal) เพื่อป้องกันการรบกวนสนามแม่เหล็กซึ่งจะทำให้กลุ่มอิเล็กตรอนเบี่ยงเบนจากบริเวณไดโนด (Dynode) ดังรูปที่ 3.1

ขั้นตอนการทำงานของหัววัดรังสีซินทิลเลชัน เมื่ออนุภาคนิวเคลียร์ตกกระทบซินทิลเลเตอร์จะเกิดประกายแสงขึ้นในบริเวณที่มีการถ่ายโอนพลังงานแสงที่เกิดขึ้นส่วนหนึ่งจะสะท้อนกลับโดยตัวสะท้อน (Reflector) และส่วนหนึ่งจะส่งตรงไปยังตัวนำแสง (Light Guide) ผ่านออกไปยังโฟโตแคโทด (Photocathode) ภายในหลอดทวีคูณอิเล็กตรอน ต่อมาโฟโตอิเล็กตรอน (Photoelectrons) จะถูกปล่อยจากโฟโตแคโทดตรงไปยังไดโนดและถูกทวีคูณด้วยส่วนทวีคูณอิเล็กตรอน (Electron Multiplier) ปริมาณอิเล็กตรอนที่ถูกรวบรวมบนแอโนดจะสร้างพัลส์ของกระแสขึ้นที่อินพุทของภาคขยายสัญญาณส่วนหน้า (Preamplifier) ปริมาณประจุที่ปรากฏบนแอโนดหรือไดโนดตัวสุดท้ายของหลอดทวีคูณจะเป็นสัดส่วนกับพลังงานของอนุภาคนิวเคลียร์



รูปที่ 3.1 โครงสร้างของหัววัดรังสีชนิดซินทิลเลชัน

หลอดทวิคูณอิเล็กตรอนเป็นส่วนประกอบสำคัญในหัววัดซินทิลเลชันทำหน้าที่เปลี่ยนประกายแสงให้เป็นอิเล็กตรอนและขยายปริมาณอิเล็กตรอนให้สูงพอที่จะสร้างสัญญาณพัลส์ หลอดทวิคูณอิเล็กตรอนที่ใช้กับหัววัดรังสีซินทิลเลชันจะเป็นแบบแสงเข้าทางด้านปลายของหลอด (End-on Type) มีโครงสร้างดังรูปที่ 3.2 ซึ่งประกอบด้วยส่วนต่าง ๆ เช่น โฟโตแคโทดแบบกึ่งแสงผ่าน (Semitransparent Photocathode) เป็นสารไวแสงประเภทไบอัลคาไล เมื่อแสงตกกระทบสารไวแสงจะปลดปล่อยอิเล็กตรอน (Photoelectron), โฟกัส (Focusing Electrode) เป็นแผ่นขั้วไฟฟ้าทำหน้าที่สร้างเลนส์ไฟฟ้าสถิตย์ควบคุมลำอิเล็กตรอนให้รวมกันสู่ไดโนดตัวแรก, ไดโนด (Dynode) เป็นแผ่นเคลือบสารที่ไวต่อการเกิด Secondary Emission จัดเรียงทำมุมเพื่อให้เกิดการทวีปริมาณอิเล็กตรอนซึ่งอาจจะมีแผ่นขั้วถึง 15 แผ่น, แอนโนด (Anode) เป็นแผ่นรับประจุอิเล็กตรอนจากการทวีปริมาณของชุดไดโนด (Dynode Chain) เพื่อสร้างสัญญาณพัลส์กระแสหรือพัลส์ศักดาไฟฟ้าทางออกโดยชุดแผ่นขั้วไฟฟ้าไดโนดและแอนโนดนี้รวมเรียกว่า “ส่วนทวีปริมาณอิเล็กตรอน (Electron Multiplier)”



รูปที่ 3.2 โครงสร้างภายในหลอดทวีคูณอิเล็กตรอน

3.1.2 ข้อมูลพื้นฐานสำหรับระบบรังสีแกมมา [7]

ระบบวัดปริมาณรังสีประกอบด้วย วงจรแหล่งจ่ายไฟฟ้าศักดาสูง วงจรแหล่งจ่ายไฟฟ้าศักดาต่ำ วงจรขยายสัญญาณพัลส์ และวงจรวิเคราะห์ความสูงสัญญาณพัลส์ โดยวงจรแหล่งจ่ายไฟฟ้าศักดาสูงสามารถจ่ายแรงดันได้จาก 0 ถึง 2000 โวลต์ ให้กับหัววัดรังสีเพื่อทำการไบอัส ให้กับหัววัดรังสีโดยมีปริมาณการจ่ายกระแสได้ 1 มิลลิแอมป์ วงจรแหล่งจ่ายไฟฟ้าศักดาต่ำจะสามารถจ่ายแรงดันไฟฟ้าที่ระดับ 5, ± 12 และ ± 24 โวลต์ โดยแรงดัน 5 โวลต์จ่ายกระแสได้สูงสุดที่ 500 มิลลิแอมป์ และที่ ± 12 โวลต์และ ± 24 โวลต์จ่ายกระแสได้สูงสุด 1000 มิลลิแอมป์ วงจรขยายสัญญาณพัลส์จะทำหน้าที่รับสัญญาณพัลส์จากอุปกรณ์วัดรังสีมาขยายให้ได้สัญญาณที่มีอัตราส่วนของสัญญาณพัลส์สูงกว่าสัญญาณรบกวน (S/N) พร้อมกับแต่งรูปสัญญาณในรูปแบบยูนิโพลาร์ (Unipolar) สัญญาณพัลส์ ที่ออกจากวงจรขยายจะมีขนาดความสูงเป็นสัดส่วนกับระดับพลังงาน

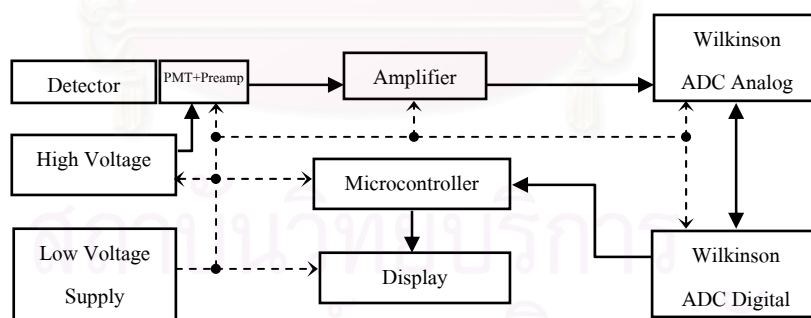
วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลจะทำหน้าที่แปลงความสูงของพัลส์ที่เป็นสัญญาณอนาลอกไปเป็นสัญญาณดิจิทัลขนาด 11 บิต หรือเทียบเท่าตำแหน่งของช่องสัญญาณที่จะจัดเก็บได้ตั้งแต่ 0 ถึง 2048 ช่องโดยสัดส่วนการแปลงจะเป็นสัดส่วนโดยตรง

3.1.3 โครงสร้างระบบควบคุมและการแสดงผลที่พัฒนาขึ้น

การควบคุมระบบทั้งหมดจะใช้ไมโครคอนโทรลเลอร์ตระกูล ARM7TDMI-S แบบ 16/32 บิต เบอร์ LPC2138 ของบริษัท Philips เป็นตัวควบคุมโดยใช้คริสตัลความถี่ 19.6608 MHz สามารถประมวลผลได้ความเร็วสูงสุดที่ความถี่ 58.9824 MHz เมื่อทำงานร่วมกับ PLL (Phase-Locked Loop) ที่อยู่ภายในตัวไมโครคอนโทรลเลอร์ ในส่วนของการเขียนโปรแกรมจะเริ่มต้นการเขียนควบคุมในส่วนของการเปิดและปิดขาสัญญาณที่เชื่อมต่อกับส่วนของการแปลงสัญญาณอนาล็อกเป็นดิจิทัล โดยควบคุมการจัดเก็บตำแหน่งที่ได้ลงในหน่วยความจำและควบคุมการแสดงผลด้วยการส่งสัญญาณควบคุมรวมทั้งข้อมูลส่งให้กับจอแสดงผลเป็นค่าตัวเลขดิจิทัลของช่วงเวลาที่ทำการแปลงสัญญาณจากอนาล็อกเป็นดิจิทัล ค่าจำนวนพัลส์ทั้งหมดที่ถูกนับได้ในแต่ละช่องและแสดงตำแหน่งช่องที่จัดเก็บจากช่องที่ 0 ถึงช่องที่ 2048

ส่วนของการแสดงผลจะใช้จอ Monitor LCD รุ่น LM32C041 ยี่ห้อ SHARP ที่มีความละเอียด 320x240 จุด แต่ละจุดจะมี 3 สีหลักอยู่ในคือ แดง เขียวและน้ำเงิน ในแต่ละสีก็จะสามารถเลือกความสว่างได้โดยมีสีละ 3 บิต นั่นก็หมายถึงเราสามารถเลือกความสว่างได้ 8 ระดับ ทำให้เมื่อนำทั้ง 3 สีมาผสมกันก็จะสามารถเกิดสีต่างๆ ได้ถึง 512 สี

จากแนวความคิดในการจัดอุปกรณ์วัดรังสีและควบคุมการทำงานจึงสามารถออกแบบอุปกรณ์วิเคราะห์แบบหลายช่องขนาดกะทัดรัดได้ดังรูปที่ 3.3

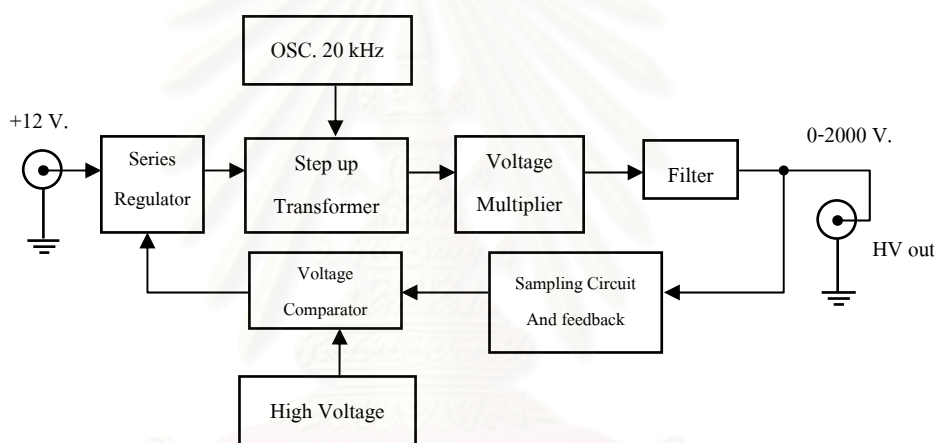


รูปที่ 3.3 แผนภาพของระบบวิเคราะห์แบบหลายช่อง

3.2 การพัฒนาระบบวัดรังสี

3.2.1 การพัฒนาวงจรแหล่งจ่ายไฟฟ้าศักดาสูง [8]

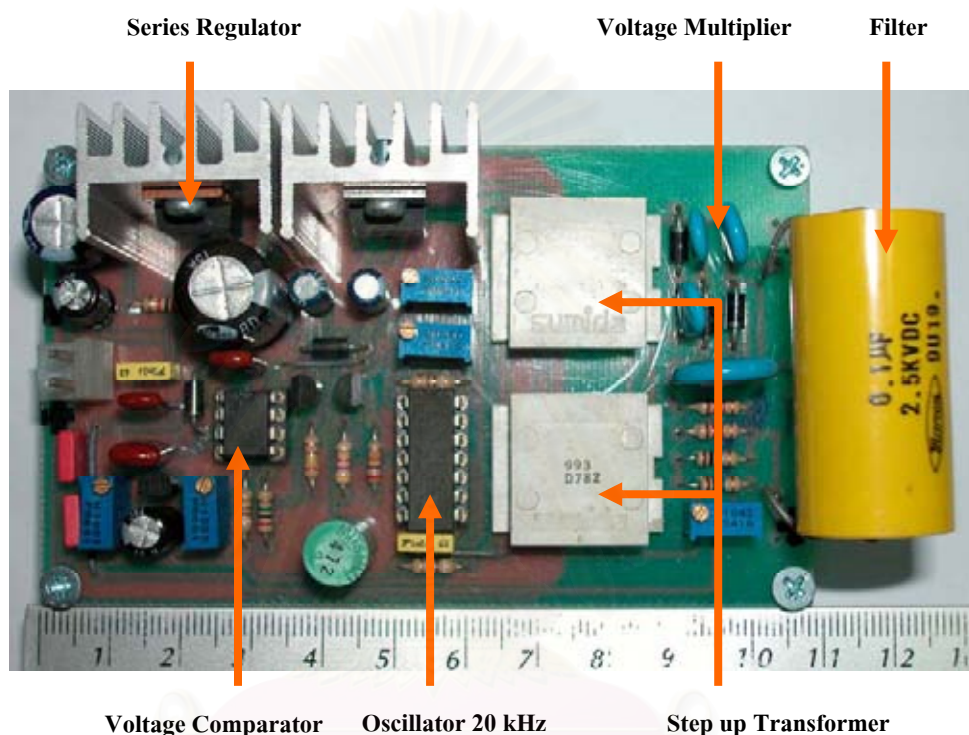
วงจรแหล่งจ่ายไฟฟ้าศักดาสูงที่พัฒนาขึ้นเป็นชนิดฟลายแบค (FLY BACK) โดยใช้หม้อแปลงสวิตชิงความถี่สูง ทำการรับแรงดันไฟฟ้าขาเข้าที่ +12 โวลต์ แล้วแปลงเป็นไฟฟ้าศักดาสูงปรับค่าได้จาก 0 ถึง +2000 โวลต์ สามารถจ่ายกระแสสูงสุดได้ 1000 ไมโครแอมป์ ในวงจรแหล่งจ่ายไฟฟ้าศักดาสูงจะประกอบด้วย วงจรกำเนิดความถี่ ซีรี่เร็กกูเลเตอร์ (Series Regulator) หม้อแปลงแบบเพิ่มศักดาไฟฟ้า วงจรทวีศักดาไฟฟ้า วงจรกรองกระแส วงจรเปรียบเทียบศักดาไฟฟ้า และวงจรควบคุมศักดาไฟฟ้าคงที่ ดังแสดงในรูปที่ 3.4



รูปที่ 3.4 แผนภาพของวงจรแหล่งจ่ายไฟฟ้าศักดาสูง

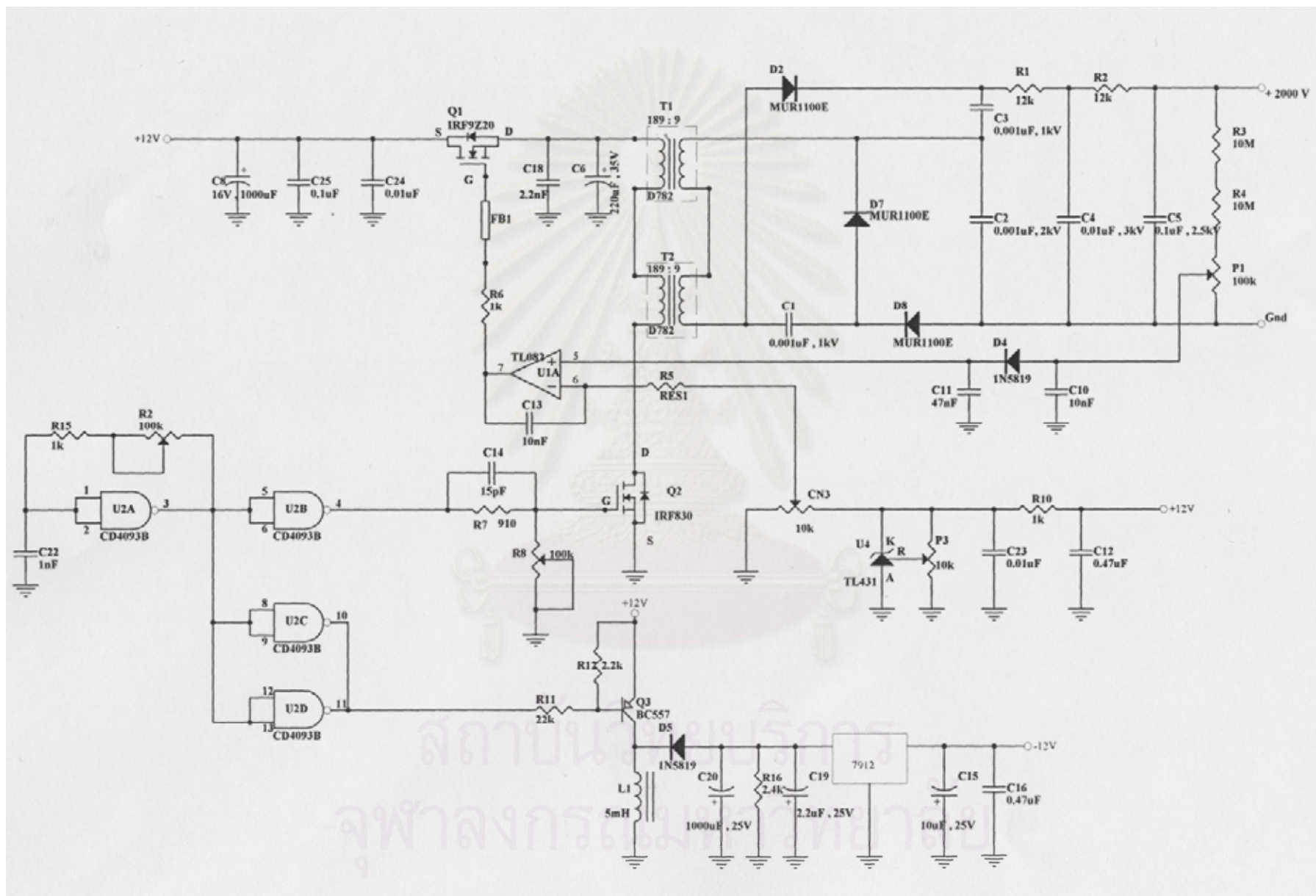
การทำงานในวงจรแหล่งจ่ายไฟฟ้าศักดาสูงดังแสดงในรูปที่ 3.6 ไอซี U2A และ U2B ทำหน้าที่เป็นออสซิลเลเตอร์ (Oscillator) กำเนิดความถี่ 20 kHz (duty cycle 50%) เพื่อควบคุมให้มอเตอร์ Q2 ที่ต่ออนุกรมกับหม้อแปลงเปิดและปิดวงจรตามจังหวะความถี่ ทำให้เกิดการเหนี่ยวนำในขดลวดปฐมภูมิ และทุติยภูมิให้มีศักดาไฟฟ้าเพิ่มขึ้น ที่อัตรารอบ 1:21 ศักดาไฟฟ้ากระแสสลับจะได้รับการทวีศักดาเป็น 3 เท่า ที่วงจรทวีแรงดันพร้อมทั้งเปลี่ยนเป็นไฟฟ้ากระแสตรง จากนั้นก็จะถูกกรองกระแสให้เรียบด้วยตัวต้านทานและตัวเก็บประจุ R1, C4 และ R2, C5 ไฟฟ้าศักดาสูงจะถูกแบ่งด้วยอัตราลดทอนของความต้านทาน 200:1 ด้วย R3, R4 และ P1 ส่งเข้าวงจรเปรียบเทียบศักดาไฟฟ้าที่ IC U1A เปรียบเทียบกับค่าศักดาไฟฟ้าอ้างอิงที่สามารถปรับได้ด้วย P2 เพื่อทำการปรับค่าไฟฟ้าศักดาสูงและควบคุมศักดาไฟฟ้าให้คงที่ ส่วนค่าเปรียบเทียบที่ได้จาก IC U1A จะ

นำไปควบคุมการทำงานของวงจรควบคุมไฟฟ้าสัปดาห์ที่มอสเฟส Q1 โดยค่าเปรียบเทียบที่ได้เป็นแรงดันไฟฟ้าที่จ่ายให้กับขาเกต (Gate) ของ Q1 ซึ่งจะเป็นตัวควบคุมกระแสที่ไหลผ่าน Q1 สำหรับ U2C และ U2D จะทำหน้าที่เป็น Buffer ระหว่าง U2B และ Q3 ซึ่ง Q3 ถูกออกแบบให้เป็นสวิตช์เพื่อใช้ควบคุมการทำงานของวงจรสร้างไฟฟ้ากระแสตรงขนาด 12 โวลต์ เพื่อจ่ายให้กับวงจรเปรียบเทียบศักดาไฟฟ้าจากวงจรทั้งหมดที่กล่าวมา สามารถสร้างแผ่นวงจรแหล่งจ่ายไฟฟ้าสัปดาห์สูงได้ดังรูปที่ 3.5



รูปที่ 3.5 แผ่นวงจรแหล่งจ่ายไฟฟ้าสัปดาห์สูง

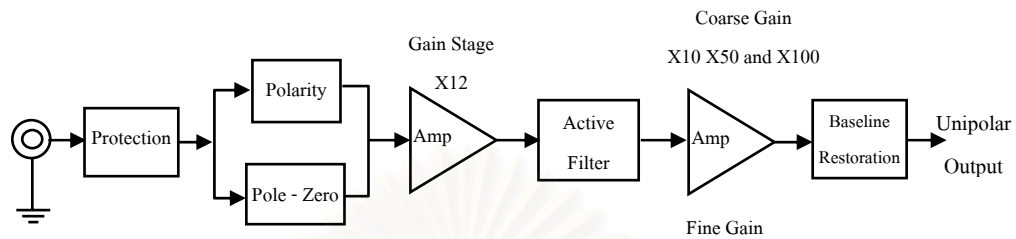
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.6 วงจรแหล่งจ่ายไฟฟ้าศักดาสูง

3.2.2 การพัฒนาวงจรขยายสัญญาณพัลส์

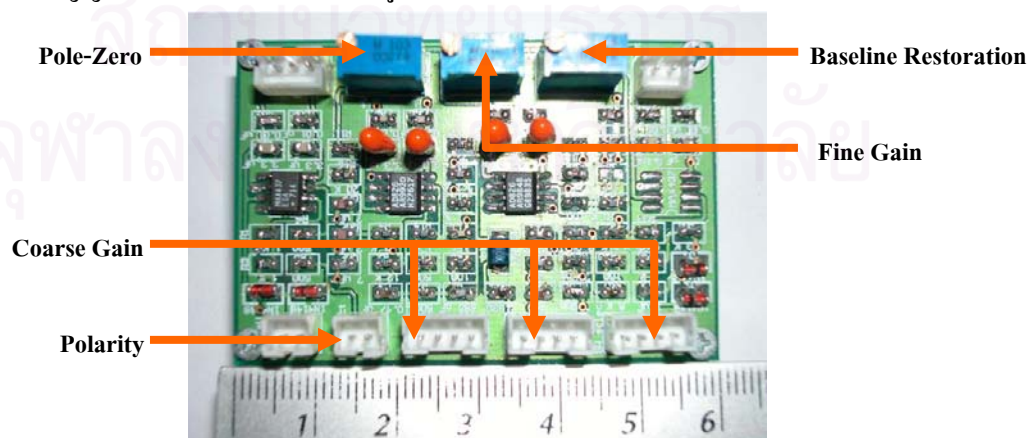
วงจรขยายสัญญาณพัลส์ทำหน้าที่ขยายและปรับแต่งรูปสัญญาณพัลส์ที่มาจากหัววัดรังสีให้เหมาะสมกับวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล มีแผนภาพการทำงานและออกแบบวงจรได้ ดังรูปที่ 3.7



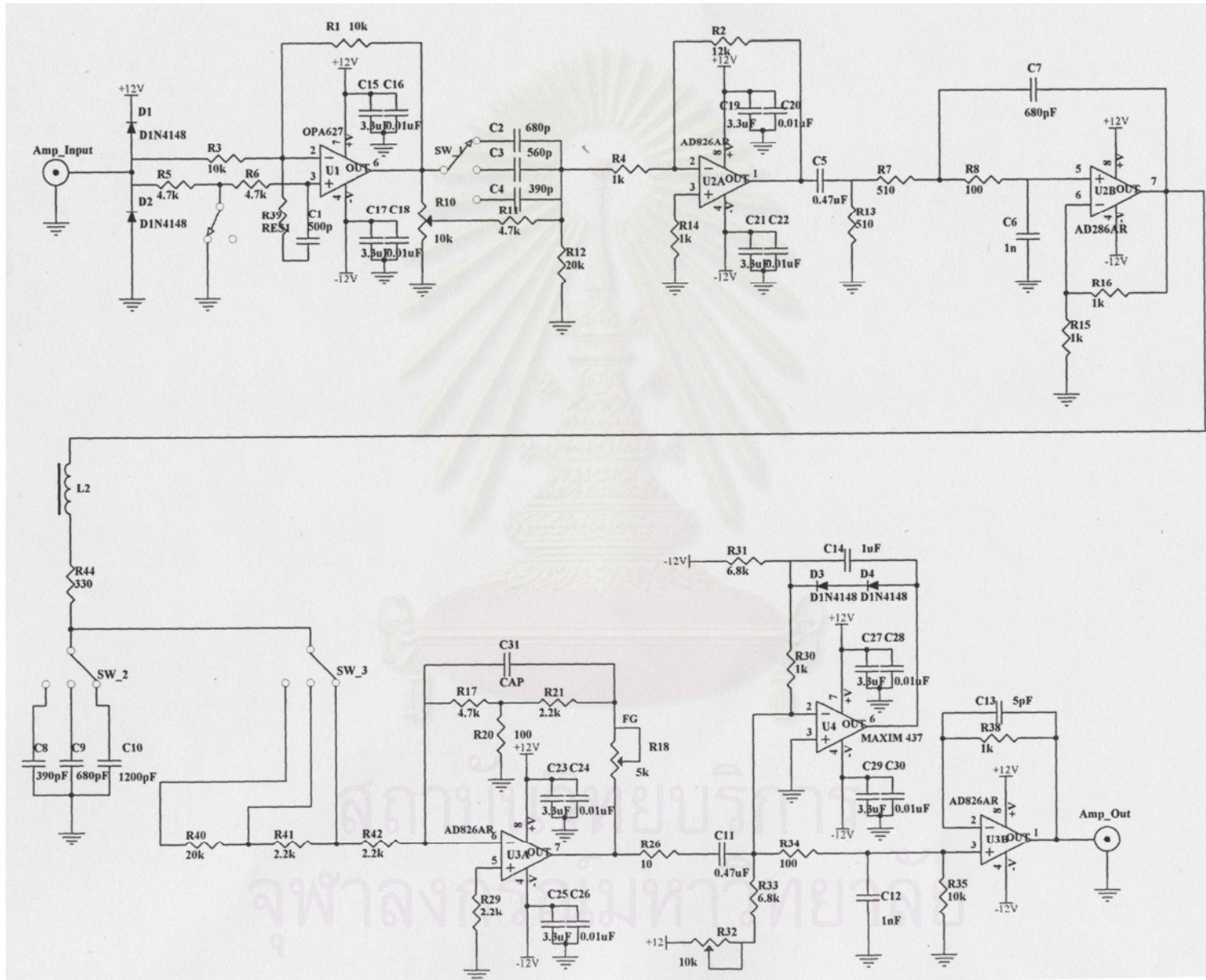
รูปที่ 3.7 แผนภาพแสดงการทำงานของวงจรขยายสัญญาณ

จากรูปที่ 3.9 วงจรขยายสัญญาณพัลส์ถูกออกแบบให้มีสวิตช์เลือกโพลาริตี (Polarity) ได้ และมีวงจรลบคั้งโพลซีโร (Pole/Zero Cancellation) เพื่อใช้ปรับแก้แอนเดอร์ชูท (Undershoot) และโอเวอร์ชูท (Overshoot) ของสัญญาณด้านขาเข้าก่อนถูกส่งไปขยายต่อ 12 เท่าด้วยวงจรขยายแบบกลับเฟส (Inverting Amplifier) จากนั้นสัญญาณพัลส์จะถูกปรับแต่งรูปร่างสัญญาณด้วยวงจรกรองความถี่สูงผ่านแบบพาสซีฟ (Passive high pass filter) และวงจรกรองความถี่ต่ำผ่านแบบแอคทีฟ (Active low pass filter) ที่มีการจัดรูปแบบการทำงานเป็น $CR-(RC)^n$ โดย $n = 2$ เพื่อแต่งรูปสัญญาณให้เป็นพัลส์แบบ Gaussian approximation

วงจรขยายสัญญาณหลัก U3A (ไอซีเบอร์ AD826/AD) ต่อแบบอินเวอร์ตติงที่สามารถเลือกอัตราขยายแบบหยาบ (Coarse Gain) เท่ากับ 100, 500 และ 1000 เท่า ด้วยสวิตช์และปรับอัตราขยายแบบละเอียด (Fine Gain) ด้วย R18 หลังจากนั้นสัญญาณพัลส์จะถูกปรับแต่งค่าเบสไลน์กลับสู่ศูนย์ (Baseline Restorer) ที่ U4 ก่อนส่งสัญญาณพัลส์ไปยังวงจรบัฟเฟอร์ (Buffer Amplifier: USB) ซึ่งวงจรขยายสัญญาณได้ทำการออกแบบดังรูปที่ 3.8



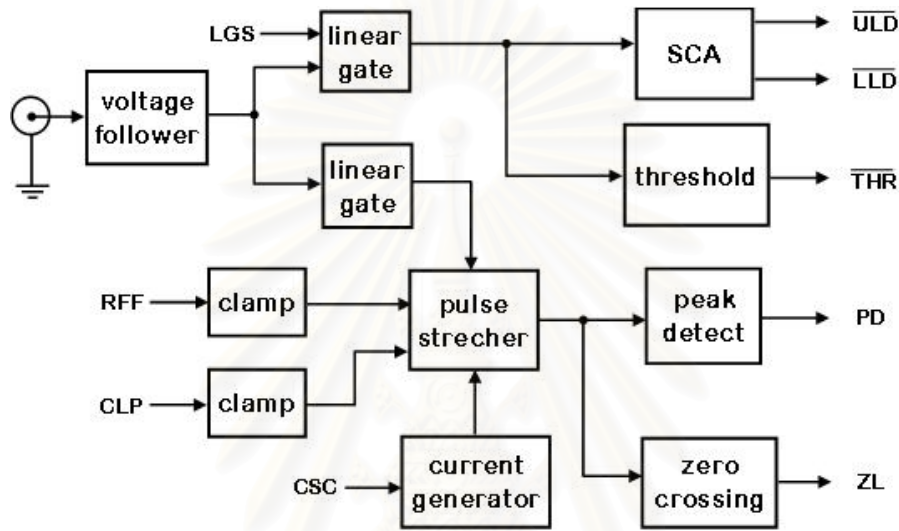
รูปที่ 3.8 แผ่นวงจรขยายสัญญาณพัลส์



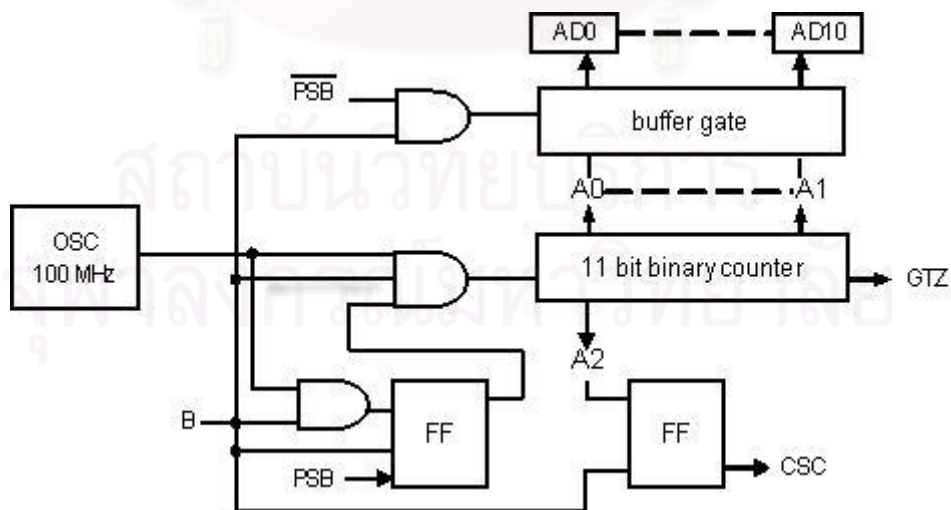
รูปที่ 3.9 วงจรขยายสัญญาณพัลส์

3.2.3 การพัฒนาวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล [9]

วงจรแปลงสัญญาณพัลส์อนาลอกเป็นสัญญาณดิจิทัลที่พัฒนาขึ้นเป็นแบบวิลกินสันซึ่งใช้ความถี่ฐานเวลา 100 MHz มีความละเอียดในการแปลงความสูงพัลส์ไม่ต่ำกว่า 11 บิต ระบบแปลงสัญญาณนี้แบ่งการทำงานออกเป็น 2 ส่วน คือ ส่วนของการทำงานของสัญญาณอนาลอกและส่วนการทำงานของสัญญาณดิจิทัล ดังแสดงในแผนภาพที่ 3.10 และ 3.11



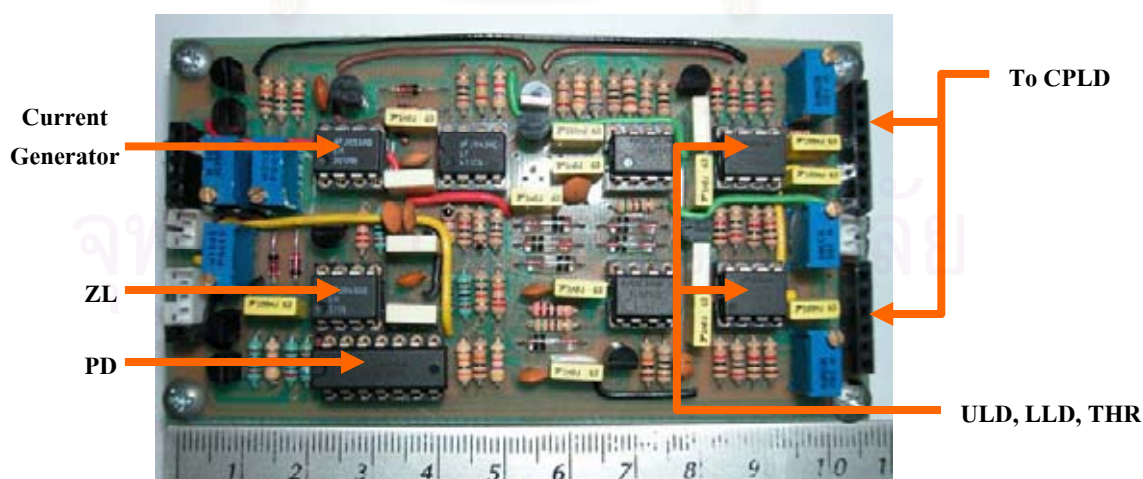
รูปที่ 3.10 แผนภาพวงจรแปลงสัญญาณพัลส์ในส่วนของสัญญาณอนาลอก



รูปที่ 3.11 แผนภาพวงจรแปลงสัญญาณพัลส์ในส่วนของสัญญาณดิจิทัล

3.2.3.1 ส่วนของการทำงานของสัญญาณนาฬิกา

ในการพัฒนาวงจรได้อาศัยแนวทางจากเครื่องวิเคราะห์ความสูงของพัลส์แบบหลายช่องของ Tracor Northern รุ่น TN 1705/1706 โดยออกแบบวงจรให้ใช้อุปกรณ์อิเล็กทรอนิกส์ที่มีขายในปัจจุบัน ที่มีขนาดเล็กสิ้นเปลืองพลังงานน้อย ใช้กับความถี่สูงได้ดีสัญญาณรบกวนต่ำและมีความเสถียรภาพในการทำงานสูง โดยอุปกรณ์ทั้งหมดจะใช้กำลังไฟฟ้าจากแหล่งจ่ายไฟฟ้าศักดาต่ำขนาด +5 และ ± 12 โวลต์ จากวงจรที่เป็นส่วนของการแปลงผันสัญญาณนาฬิกาแสดงในรูปที่ 3.14 ประกอบด้วย U2 และ U3 ซึ่งทำหน้าที่เป็นบัฟเฟอร์ และมีวงจรเปรียบเทียบศักดาไฟฟ้า (Voltage comparator) สำหรับสัญญาณเทรฮิส (Threshold: THR), LLD และ ULD โดยอาศัยการทำงานของไอซีเปรียบเทียบศักดาไฟฟ้า (Comparator) U7A, U8A และ U9A ตามลำดับ ซึ่งสัญญาณที่ด้านทางออกของไอซีเปรียบเทียบศักดาไฟฟ้าแต่ละตัวจะถูกนำไปใช้เป็นสัญญาณควบคุมการทำงานของส่วนเอ็ดจีซีชนิดวิลคินสันที่เป็นดิจิทัล สำหรับโครงสร้างวงจรของไอซี U1 ถูกออกแบบให้ทำหน้าที่เป็นวงจรพีคดีเทคเตอร์ซึ่งค่าศักดาไฟฟ้าที่ยอดของสัญญาณพัลส์แต่ละลูกจะถูกซาร์จประจุไว้ที่ C8 โดยมี J1, J2, Q17 และ U12 ทำหน้าที่เป็นแหล่งจ่ายกระแสคงที่ (Constant current source) และสวิตช์เพื่อใช้ดีสซาร์จประจุที่เกิดขึ้นบน C8 สำหรับสร้างเป็นสัญญาณ Run-down time ในส่วนของ MOS-FET M2, M6, M7 และ M13 ได้ถูกออกแบบให้ทำงานเป็นสวิตช์ โดย M2 และ M13 จะทำหน้าที่เป็นลิเนียร์เกต (Linear gate) สำหรับโครงสร้างวงจรที่ประกอบด้วยไอซี U25, Q2, Q5 และ Q6 เป็นวงจรที่ใช้ในการสร้างสัญญาณควบคุมที่เรียกว่าซีโรครอสซิง (Zero crossing : ZL) ของ MCA และสัญญาณพัลส์ของพีคดีเทคเตอร์ (Peak detector : PD) ถูกสร้างจากวงจรสวิตช์ Q1 ซึ่งแผ่นวงจรแปลงสัญญาณพัลส์ในส่วนของการแปลงผันสัญญาณนาฬิกาแสดงไว้ในรูปที่ 3.12



รูปที่ 3.12 แผ่นวงจรแปลงสัญญาณพัลส์ในส่วนของการแปลงผันสัญญาณนาฬิกา

3.2.3.2 ส่วนของการทำงานของสัญญาณดิจิทัล

การพัฒนาในส่วนนี้จะเปลี่ยนจากไอซีลอจิกจากตระกูล TTL และ CMOS ที่ใช้ในวงจรส่วนใหญ่มาเป็นชิพ CPLD (Complex Programmable Logic Device) เพียงตัวเดียวโดยชิพนี้เป็นอุปกรณ์ที่ได้มีการออกแบบวงจรโครงสร้างภายในเป็นวงจรลอจิกพื้นฐานต่างๆ AND array, OR array และ Macrocells ต่อกันอยู่เป็นกลุ่มมีทั้งวงจรคอมบิเนชัน (Combination) และ (Sequential) อีกทั้งยังมีความยืดหยุ่นในการออกแบบสูง คือ สามารถกำหนดโครงสร้างการทำงานภายในไอซี CPLD ได้อย่างอิสระโดยไอซี CPLD นี้จะเป็นไอซีประเภทเดียวกันกับไอซี FPGA (Field Programmable Gate Array) จะต่างกันตรงที่ไอซี CPLD เมื่อทำการโปรแกรมแล้วข้อมูลไม่สูญหาย แม้จะไม่มีไฟเลี้ยงจ่ายให้ก็ตาม ส่วน FPGA ข้อมูลที่โปรแกรมจะเกิดการสูญหายเมื่อเราหยุดจ่ายไฟให้กับตัวไอซีและมีวงจรโครงสร้างตลอดจนฟังก์ชันการทำงานที่ซับซ้อนกว่า CPLD ซึ่งโดยทั่วไปไอซีประเภท FPGA จะถูกใช้เป็นตัวแบบการสร้างหรือทดลองออกแบบไอซีต่างๆ ซึ่งผู้ออกแบบทำได้เองโดยไม่ต้องไปพึ่งโรงงาน อีกทั้งการตรวจสอบหรือการจำลองการทำงานยังทำได้ง่าย และยังมีเครื่องมือที่ช่วยในการพัฒนางานทางด้านนี้อีกมากมาย

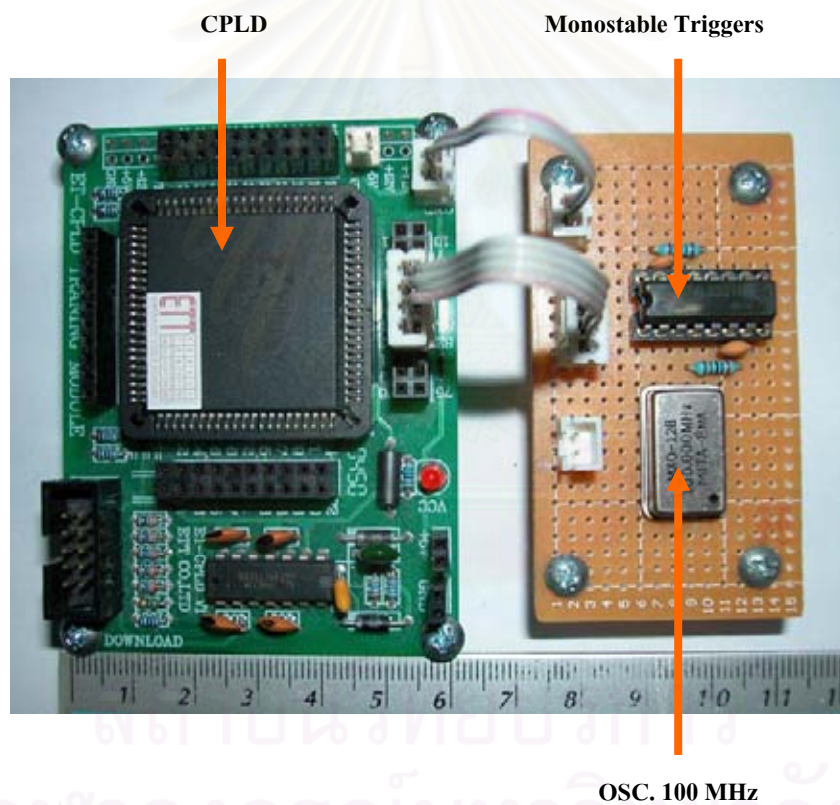
เนื่องจากไอซีประเภท CPLD สามารถเก็บข้อมูลไว้ได้ตลอดแม้ไม่มีไฟเลี้ยงจ่ายให้อีกทั้งราคายังถูกกว่าอุปกรณ์จำพวก FPGA ดังนั้นจึงเหมาะสำหรับการนำไปใช้หรือการผลิตเป็น Product หรือชิ้นงานที่ไม่มีความซับซ้อนมากนัก เนื่องจากมีข้อจำกัดในเรื่องจำนวนเกตและโครงสร้างภายในโดยจะมีจำนวนเกตและความซับซ้อนของโครงสร้างน้อยกว่าไอซีประเภท FPGA ทำให้การออกแบบวงจรที่มีความซับซ้อนมากๆ ซึ่งต้องใช้เกตภายในจำนวนมากนั้น ไอซี CPLD ก็อาจจะไม่สามารถทำได้ตามที่ FPGA และ CPLD ต่างก็มีวิธีการและเครื่องมือในการพัฒนาที่เหมือนกันดังนั้นเมื่อเราศึกษาการใช้งานไอซี CPLD จนเข้าใจแล้วเราก็สามารถที่จะใช้งาน FPGA ได้เช่นเดียวกัน

ในส่วนของการพัฒนาโปรแกรมนั้น ชุด ET-CPLD TRAINING MODULE สามารถใช้การเขียนด้วยรูปแบบ SCHEMATIC หรือในรูปแบบภาษา HDL ก็ได้ด้วยโปรแกรม XILINX FOUNDATION SERIES และ XILINX WEBPACK โดยเราเขียนออกแบบโปรแกรมบนเครื่องคอมพิวเตอร์พีซี จากนั้นการ DOWNLOAD ข้อมูลผ่านทาง PRINTER PORT ด้วยสายต่อ JTAG เข้าไปในชิพ CPLD XC95108 ได้โดยตรง

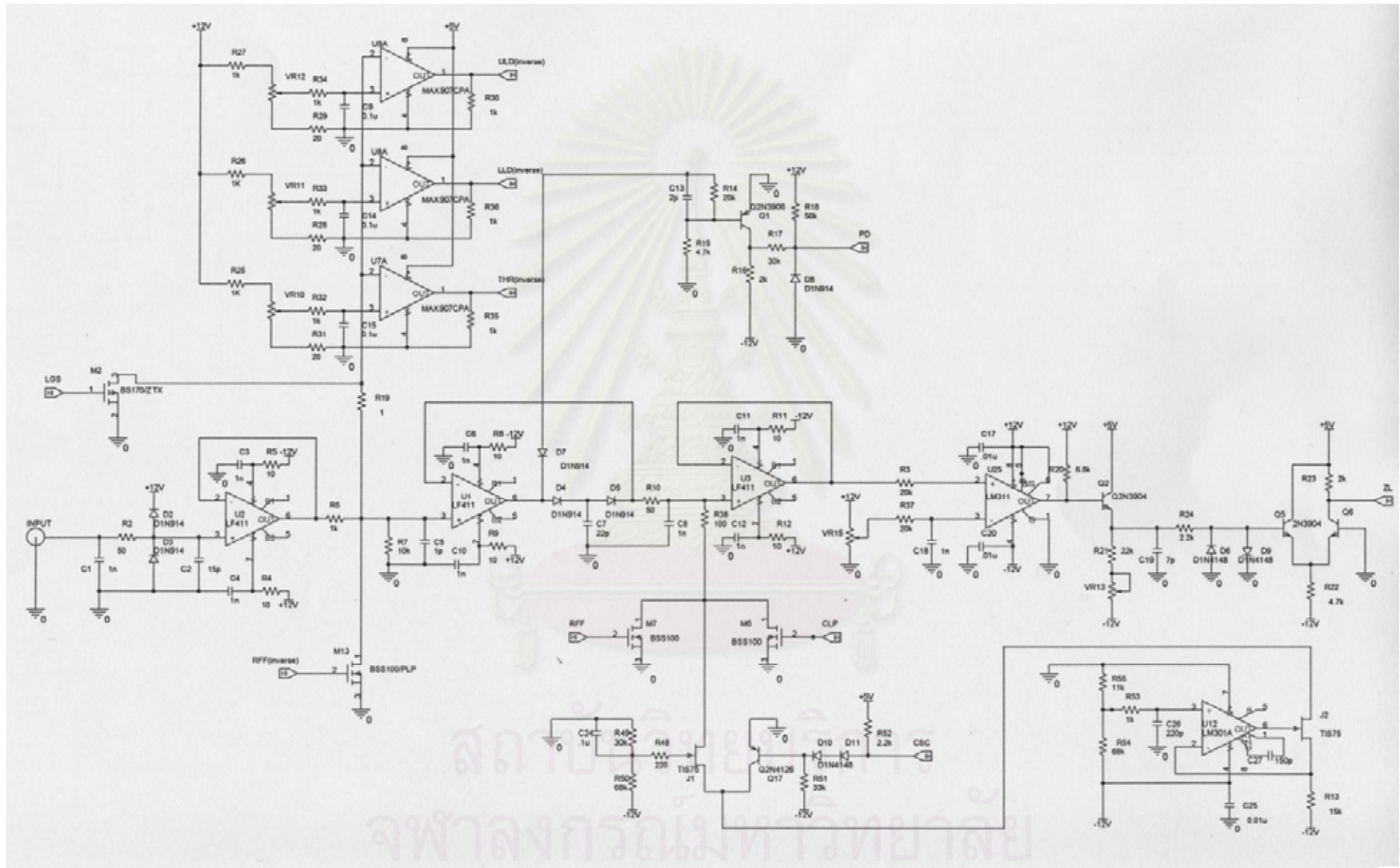
- คุณสมบัติทางเทคนิคของ CPLD XC95108

CPLD เบอร์ XC95108 ถือว่าเป็นไอซีที่มีจำนวนเกตภายในมากพอสมควร สามารถนำไปออกแบบวงจรรวมที่ซับซ้อนได้ในระดับหนึ่งโดยจะมีคุณสมบัติต่างๆ ดังนี้

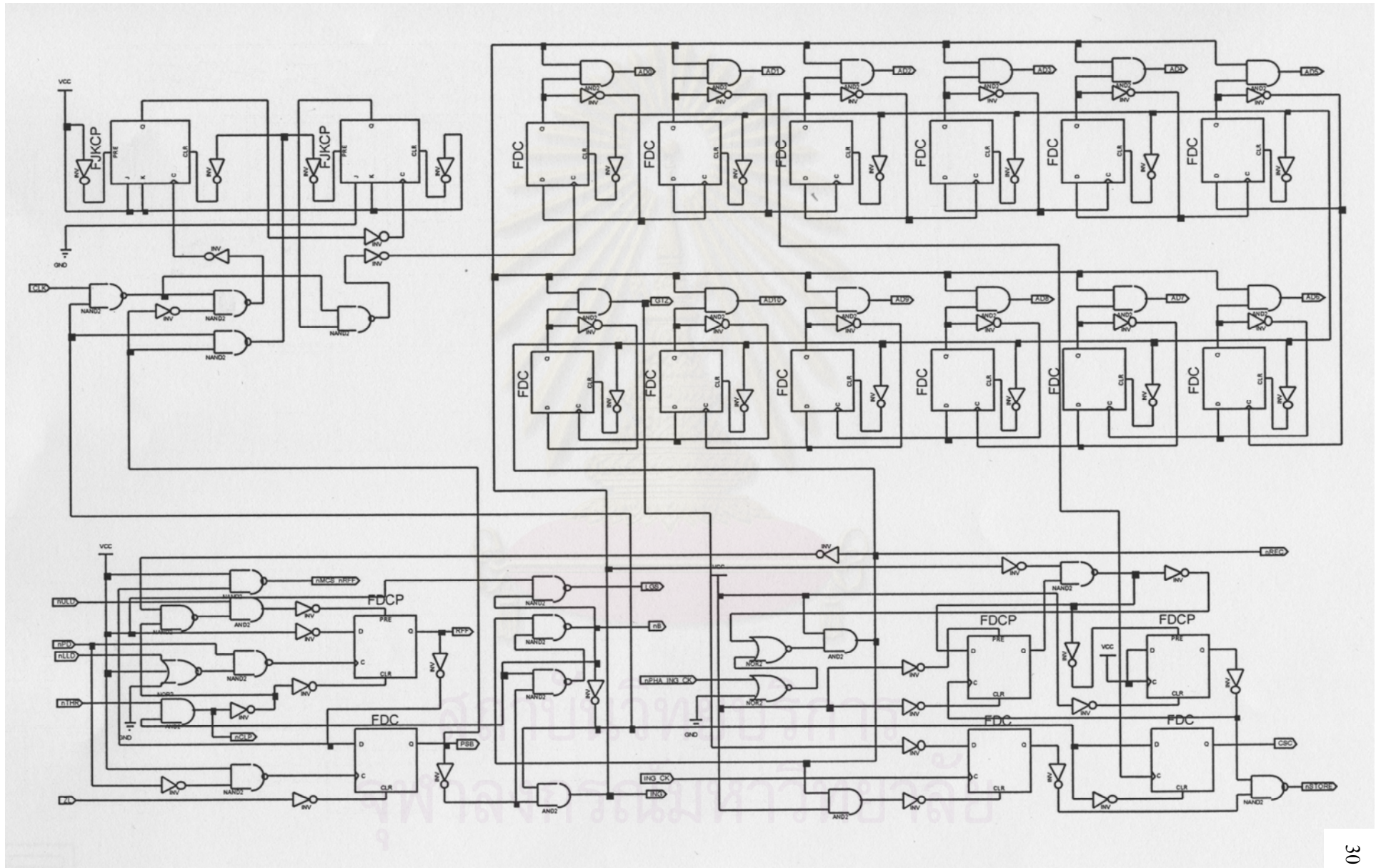
- 7.5 ns PIN TO PIN LOGIC DELAYS ON ALL PINS
- 108 MACROCELLS WITH 2400 USABLE GATES
- 5V IN-SYSTEM PROGRAMMABLE (ISP)
- ENDURANCE OF 10000 PROGRAM / ERASE CYCLES
- HIGH-DRIVE 24 mA OUTPUTS
- 3.3 V OR 5 V I/O CAPABILITY
- 84 PIN PLCC TYPE



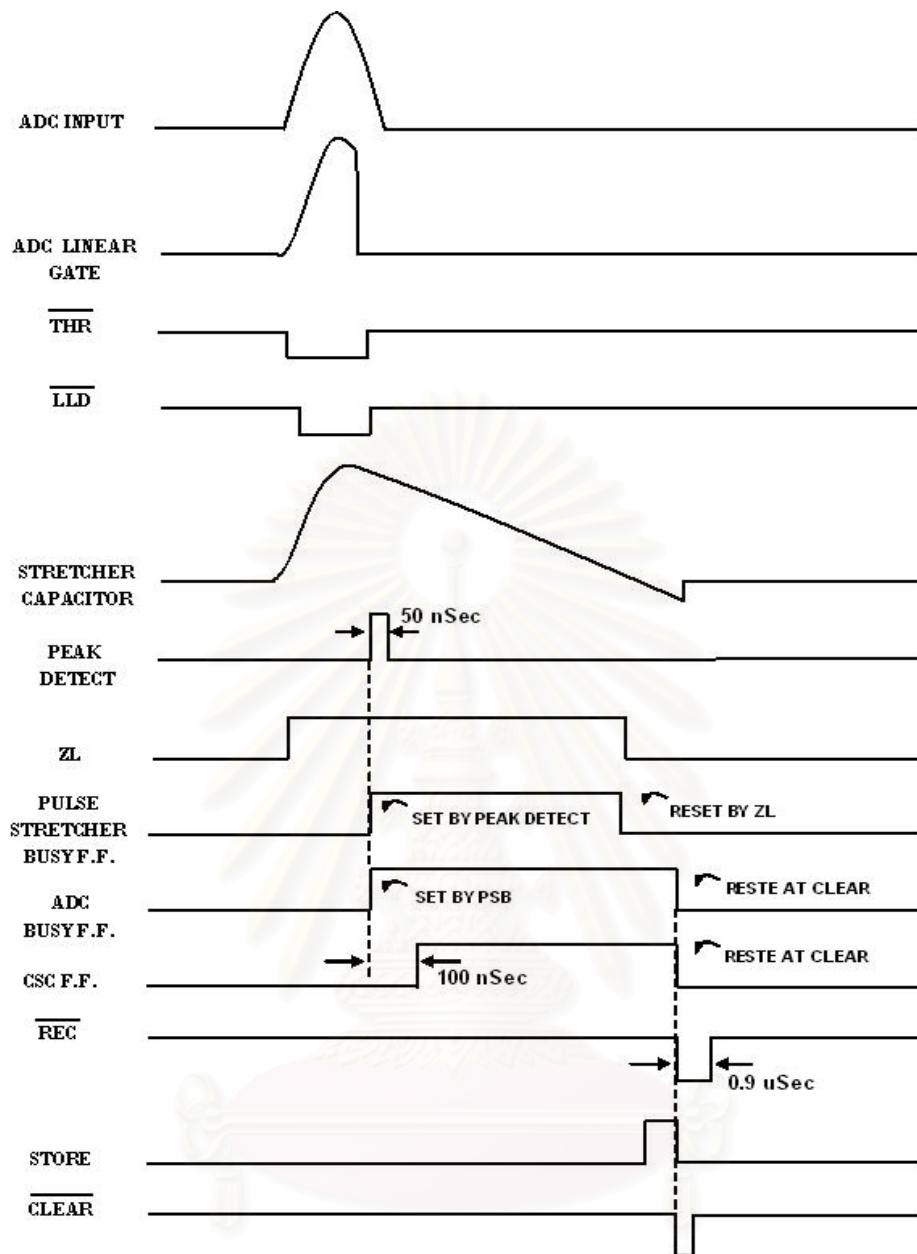
รูปที่ 3.13 แผ่นวงจรแปลงสัญญาณพัลส์ในส่วนของการแปลงผันสัญญาณดิจิทัล



รูปที่ 3.14 วงจรแปลงสัญญาณพัลส์ในส่วนของการแปลงผันสัญญาณ



รูปที่ 3.15 วงจรแปลงสัญญาณพัลส์ในส่วนของการแปลงผันสัญญาณดิจิทัล



รูปที่ 3.16 แผนภาพเวลาของการแปลงผันสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

วงจรทั้งหมดในระบบแปลงสัญญาณพัลส์นี้ถูกพัฒนาขึ้น โดยอาศัยแนวทางจากเครื่องวิเคราะห์ความสูงของพัลส์แบบหลายช่องของ Tracor NorTerm รุ่น TN 1706 การทำงานของระบบเริ่มจากสัญญาณพัลส์จากอุปกรณ์ขยายสัญญาณรูปยูนิโพลาร์ ขนาด 0-10 โวลต์ ส่งผ่านมายัง U2 ซึ่งเป็นวงจรขยายที่มีอัตราขยาย 1 สัญญาณทางออกของ U2 จะนำไปผ่านวงจรเปรียบเทียบศักดาไฟฟ้า เพื่อคัดเลือกช่วงวิเคราะห์ความสูงของพัลส์ตามย่านวิเคราะห์ที่ผู้ใช้ต้องการได้แก่ วงจรเปรียบเทียบระดับต่ำสุดและระดับบนสุด ซึ่งทำงานแบบ SCA เพื่อสร้างสัญญาณ ULD และ LLD ที่ไอซี U9A และ U8A และอีกวงจรหนึ่งเป็นวงจรเปรียบเทียบระดับความสูงต่ำสุด ที่จะยอมให้

ระบบทำการวิเคราะห์ (Threshold) เพื่อสร้างสัญญาณ THR ที่ U7A ในกรณีที่สัญญาณอยู่นอกสถานะที่จะวิเคราะห์ วงจรฟลิป-ฟลอปจะสร้างสัญญาณ RFF และ CLP ยกเลิกสัญญาณด้วยการปิดลิเนียร์เกต M13 และคายประจุสัญญาณจากคาปาซิเตอร์ด้วยวงจรแคลมป์เมื่อเกินสถานะของ SCA และแคลมป์เมื่อสัญญาณต่ำกว่าระดับความสูงที่ยอมรับ นอกจากนี้วงจรแปลงสัญญาณส่วนหน้ายังมีทางเข้าของสัญญาณเกตซึ่งสามารถควบคุมจากระบบวัดภายนอกได้ โดยควบคุมที่สัญญาณ LGS ให้ปิดลิเนียร์เกต M2 ด้วย

ในกรณีที่สัญญาณพัลส์จากอุปกรณ์ขยายสัญญาณมีความสูงอยู่ในเงื่อนไขที่ระบบพร้อมวิเคราะห์ คาปาซิเตอร์ C8 จะได้รับการประจุตามขนาดความสูงของพัลส์ด้วยวงจร Pulse Stretcher ที่ U1 ขนาดของพัลส์บนคาปาซิเตอร์ดังกล่าวจะถูกตรวจสอบด้วยวงจร 2 วงจร คือ พีคดีเทค (Peak detect) ที่ Q1 และซีโรครอสซิง (Zero crossing) ที่ U25 กล่าวคือเมื่อสัญญาณพัลส์ถึงค่าสูงสุด สัญญาณทางเข้าของ U1 ซึ่งเปรียบเทียบกับระหว่างขนาดความสูงของพัลส์ที่ประจุบนคาปาซิเตอร์ และขนาดของสัญญาณพัลส์ที่อุปกรณ์ขยายจะทำให้สัญญาณทางออกของ U1 เปลี่ยนระดับอย่างรวดเร็ว กระตุ้นวงจรพีคดีเทคให้สร้างสัญญาณ PD ดังในแผนภาพเวลารูปที่ 3.16 สัญญาณนี้จะกระตุ้นฟลิป-ฟลอปให้สร้างสัญญาณ PSB และสัญญาณ B เพื่อเปิดเกตของวงจรมับไบนารี ดังแสดงในแผนภาพรูปที่ 3.11 และสร้างสัญญาณ CSC สัญญาณนี้จะควบคุมการคายประจุของ C8 ด้วยวงจร Current Generator อันประกอบด้วย J1, J2, Q17 และ U12 ซึ่งช่วงเวลาการคายประจุนี้เรียกว่า “เวลาการแปลงผันสัญญาณ” เวลาจะมีความยาวเท่าใดขึ้นอยู่กับอัตราการคายประจุตามขนาดความต้านทานที่จัดไว้เป็นลำดับ อันเป็นการควบคุมระหว่างเวลาการแปลงผันนี้วงจรมับไบนารีจะนับความถี่ฐานเวลา จนกระทั่งวงจรตรวจสอบระดับสัญญาณจากการคายประจุตัดเส้นศูนย์ (U25) สร้างสัญญาณ ZL กระตุ้นฟลิปฟลอปให้เปลี่ยนสถานะปิดเกต ข้อมูลของขนาดความสูงของพัลส์อนาล็อกที่ถูกเปลี่ยนเป็นรหัสไบนารีจะปรากฏอยู่บนทางออกของวงจรมับไบนารี (A0-A10) และสัดส่วนของข้อมูลนี้อาจจะเริ่มนับจากฐาน 0 หรือต่ำกว่านั้นขึ้นกับรหัสที่เริ่มต้น (preset) ไว้ก่อนนับ เรียกวิธีการนี้ว่า “ADC offset” เมื่อสิ้นสุดการนับของวงจรมับไบนารีสัญญาณ B และ PSB จะสร้างสัญญาณ ING เพื่อเปิดบัฟเฟอร์เกต สำหรับส่งข้อมูลไบนารีซึ่งจะใช้เป็นข้อมูลกำหนดช่องวิเคราะห์ ไมโครคอนโทรลเลอร์จะรับข้อมูลดังกล่าวเมื่อชุดเกตในแผนภาพรูปที่ 3.12 สร้างสัญญาณส่งผ่านวงจรเชื่อมโยงสัญญาณเพื่อขอขัดจังหวะการทำงานของ CPU ทำการอ่านข้อมูลจากบัฟเฟอร์เกตเมื่อสิ้นสุดการอ่านข้อมูลจะส่งสัญญาณ Clear สำหรับเริ่มต้นการแปลงสัญญาณต่อไป สัดส่วนของข้อมูลไบนารีจะขึ้นกับช่วงเวลาที่เกิดจากการแปลงผันความสูงของพัลส์เป็นคาบเวลา นอกจากนี้ช่วงเวลาของการแปลงสัญญาณ (ADC busy) ยังถูกนำไปเฉลี่ยเป็นค่าเปอร์เซ็นต์เดดไทม์ (% dead time) สัญญาณ DTM จะถูกเปลี่ยนเป็นสัญญาณเชิงตัวเลขโดยทำการจัดเก็บค่าลงในหน่วยความจำที่อยู่ภายในไมโครคอนโทรลเลอร์

3.3 การพัฒนาระบบควบคุมการทำงานและการแสดงผล

3.3.1 ระบบควบคุมการทำงานโดยใช้ไมโครคอนโทรลเลอร์ ARM7 [1]

ไมโครคอนโทรลเลอร์ ARM7 ข้อดีคือจะมีหน่วยความจำภายในขนาด 8 kB ถึง 40 kB ส่วนหน่วยความจำโปรแกรมจะมีขนาด 32 kB ถึง 512 kB อยู่ภายในชิพส่วนการประมวลผลเป็นแบบ 16/32 บิตแล้วยังมีข้อดีในเรื่องสถาปัตยกรรมที่ไม่ซับซ้อน ทำให้ประหยัดพื้นที่ในการผลิตชิพ และมีขนาดเล็กกินพลังงานน้อย โดยที่ยังคงมีสมรรถนะที่สูง

ในยุคแรกชิพ ARM เป็นชิพ RISC (Reduced Instruction Set Computer) ขนาด 32 บิต การทำงานจำเป็นจะต้องต่อกับหน่วยความจำและอุปกรณ์ภายนอก เมื่อมีบริษัทผู้ผลิตไมโครคอนโทรลเลอร์จำนวนมากได้นำลิขสิทธิ์ของชิพ ARM ไปพัฒนาต่อได้มีการเพิ่มหน่วยความจำภายในทั้ง ROM และ RAM และเพิ่มโมดูลอุปกรณ์เสริมต่างๆ เช่น วงจรสื่อสารแบบอนุกรมวงจรแปลงดิจิทัลเป็นอนาลอก เป็นต้น ทำให้กลายเป็นไมโครคอนโทรลเลอร์แบบ 32 บิตที่กินพลังงานต่ำ สามารถทำงานได้โดยใช้ชิพไอซีเพียงตัวเดียวโดยไม่ต้องต่ออุปกรณ์เพิ่มเติมภายนอก

คุณสมบัติต่างๆ ในไมโครคอนโทรลเลอร์ ARM7 LPC2138

- ไมโครคอนโทรลเลอร์ขนาด 16/32 บิต ในตัวถัง LQFP 64 ขา
- หน่วยความจำ Static RAM LPC2138 มีขนาด 32 kB
- หน่วยความจำ Flash Program Memory ขนาด 512 kB อยู่ภายในชิพที่สามารถลบและเขียนซ้ำได้ถึง 10,000 ครั้ง
- โปรแกรมชิพได้ทันทีผ่าน In-System Programming (ISP) และ In-Application Programming (IAP) โดยใช้ซอฟต์แวร์ boot-loader ที่อยู่ภายในชิพ
- ตัวควบคุมอุปกรณ์ USB 2.0 Full-speed โดยมี RAM สำหรับ Endpoint ขนาด 2 kB ถ้าเป็น
- วงจรแปลงอนาลอกเป็นดิจิทัลความละเอียด 10 บิตจำนวน 2 ชุด ที่รับอินพุตได้ถึง 14 อินพุต โดยมีเวลาในการแปลงค่าต่ำถึง 2.44 μ s
- วงจรแปลงดิจิทัลเป็นอนาลอกความละเอียด 10 บิต 1 ตัว
- วงจรไทมเมอร์ขนาด 32 บิต 2 ชุด

- PWM (Pulse width modulation) 6 เอาท์พุท
- โมดูลนาฬิกาเวลาจริง (Real Time Clock) ที่สามารถต่อกับคริสตอลความถี่ 32 kHz และแบตเตอรี่ภายนอกได้
- มีวงจรถ่าย Phase Lock Loop ภายในเพื่อคุณค่าให้สัญญาณนาฬิกาภายในทำงานที่ความถี่สูงสุดถึง 60 MHz
- ใช้กับแหล่งจ่ายไฟชุดเดียวขนาด 3.0 โวลต์ ถึง 3.6 โวลต์
- มี I/O pin อเนกประสงค์ที่สามารถใช้กับระดับแรงดัน 5 โวลต์ ได้สูงสุด 45 ขา โดยสามารถจัดเป็นขาอินเทอร์รัปต์จากภายนอกได้สูงสุด 21 ขา
- มีโหมดประหยัดพลังงาน 2 โหมด ได้แก่ Idle และ Power-down

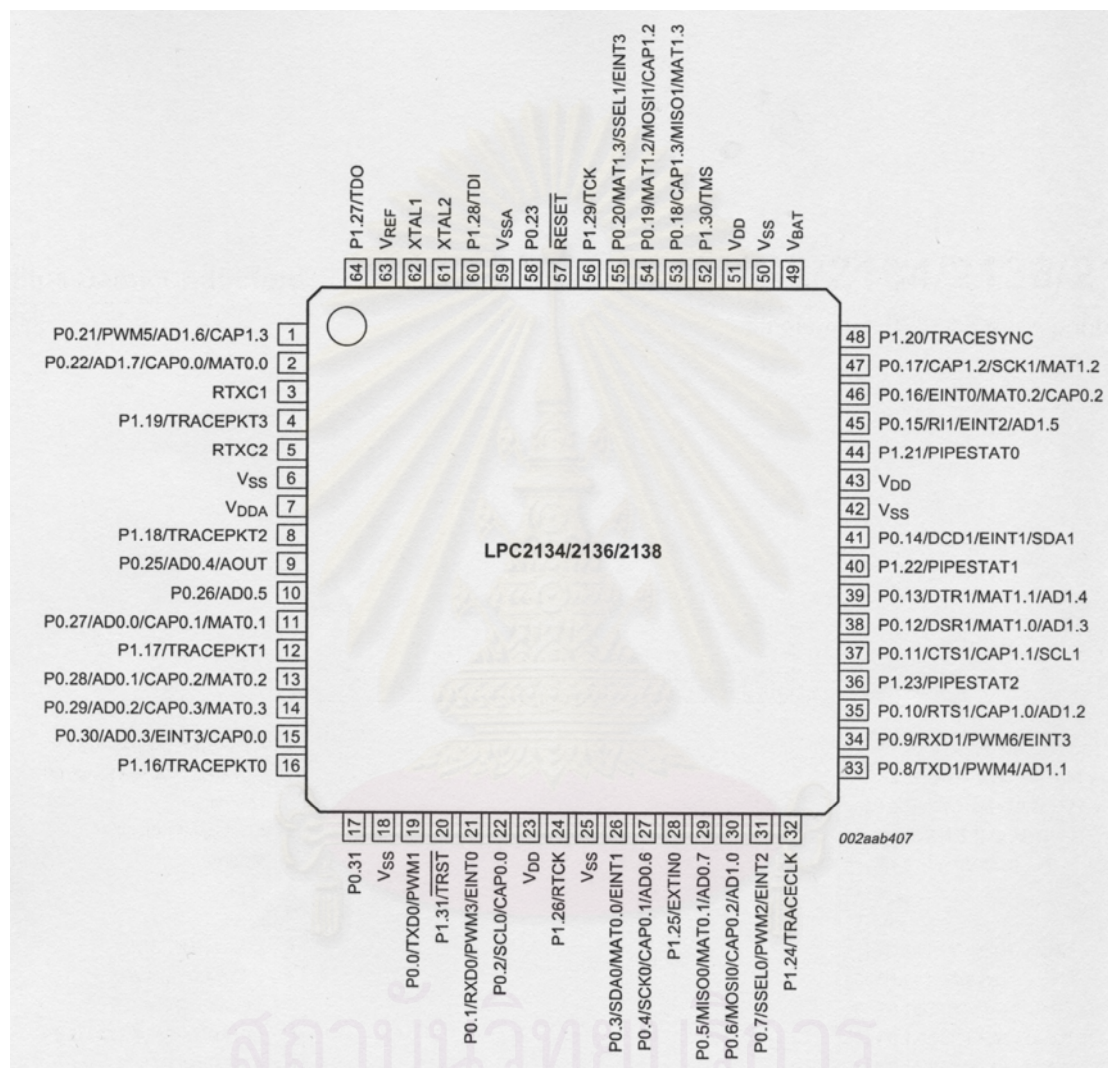
3.3.2 สถาปัตยกรรมซีพียู ARM7

สถาปัตยกรรมของ ARM7 เป็นซีพียูแบบ RISC ขนาด 32 บิต ภายในมีบัสขนาด 32 บิตตัวเดียวที่ใช้สำหรับส่งข้อมูลและคำสั่ง ชุดคำสั่งจะมีขนาด 32 บิตคงที่ ในขณะที่ข้อมูลสามารถเลือกได้ว่าจะมีขนาด 8, 16, หรือ 32 บิต โครงสร้างของ ARM7 จะเป็นแบบที่เรียบง่ายมีชุดคำสั่งไม่มากนัก ประหยัดพื้นที่สารกึ่งตัวนำที่ใช้สร้างและประหยัดพลังงาน

สถาปัตยกรรมของ ARM7 จะเป็นแบบ load-and-store ในการประมวลผลข้อมูลใดๆ ต้องกระทำผ่านทางรีจิสเตอร์เริ่มต้นด้วยการโหลดค่าจากหน่วยความจำเก็บในรีจิสเตอร์นำค่ามาประมวลผลเสร็จแล้วจะเขียนค่าเก็บในหน่วยความจำดังเดิม

3.3.2.1 การจัดวางขาของ LPC2138

ไมโครคอนโทรลเลอร์ LPC2138 อยู่ในตัวถังแบบ plastic low profile quad flat package (LQFP64) ที่มีขาต่อจำนวน 64 ขาโดยมีการจัดขาแสดงได้ในรูปที่ 3.17

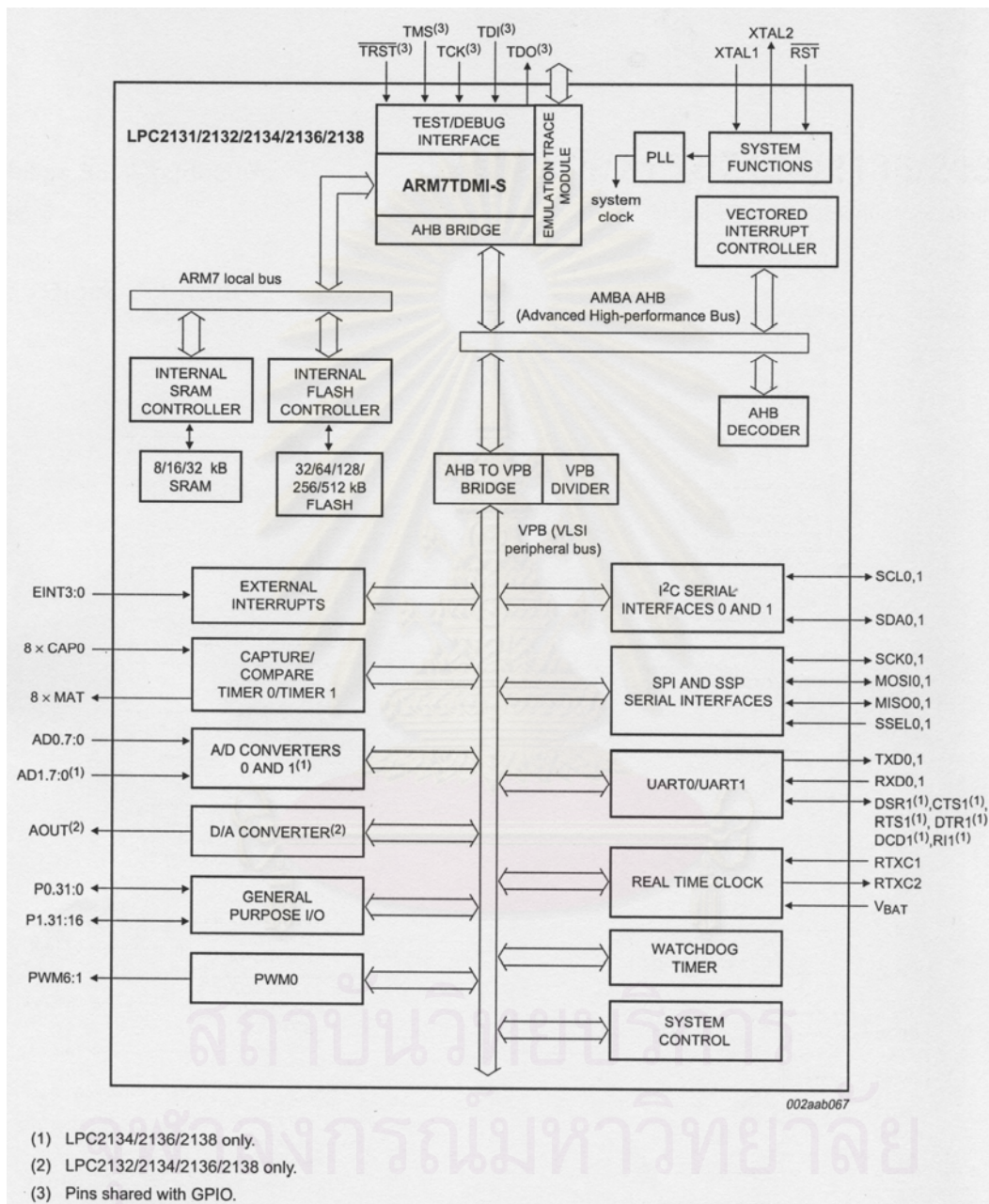


รูปที่ 3.17 แสดงการจัดขาของ LPC2148

หลังจากเกิดการรีเซ็ตขาพอร์ตทั้งหมดจะถูกกำหนดให้ทำหน้าที่เป็นอินพุต ขาแต่ละขาจะมีหน้าที่การทำงานหลายหน้าที่ ถ้าเป็นพอร์ตอินพุตจะเรียกว่าเป็น อินพุตเอาต์พุตอเนกประสงค์ (General Purpose Input Output: GPIO)

3.3.2 บล็อกไดอะแกรมของ LPC 2138

บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ LPC2138 แสดงได้ดังรูป 3.18



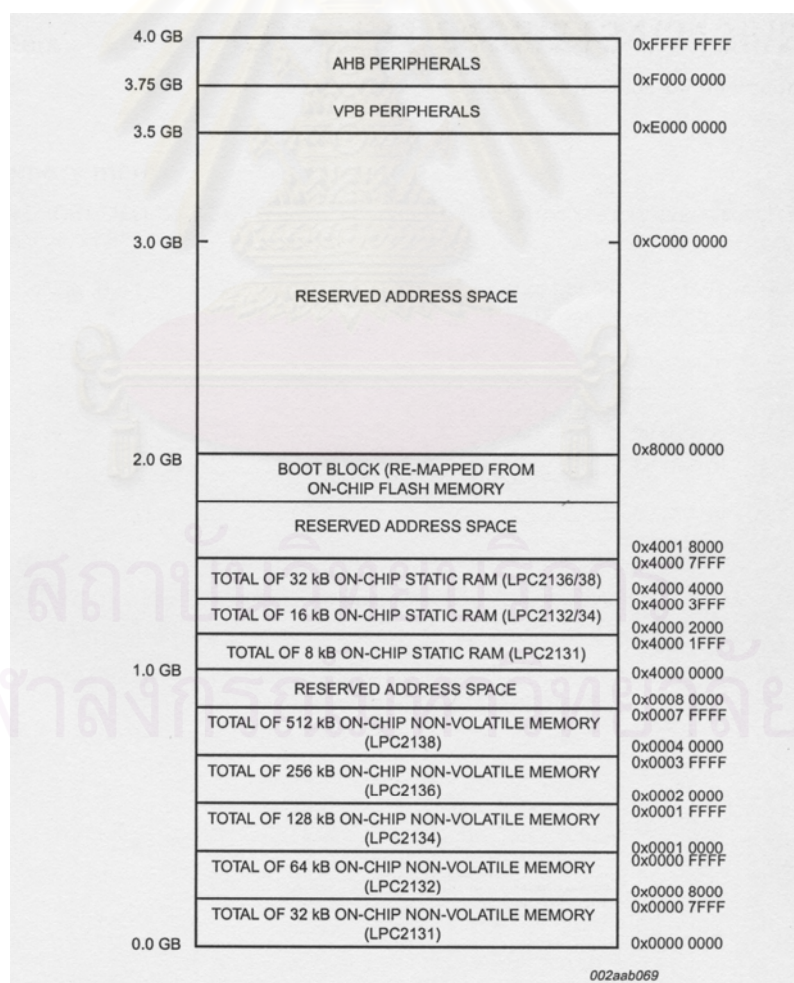
รูปที่ 3.18 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ LPC2138

จากรูปส่วนบนเป็นไมโครคอนโทรลเลอร์ ARM7TMI-S ซึ่งเป็นหัวใจหลักด้านซ้ายมือ ที่เป็นส่วนของ ARM7 Local Bus ที่ใช้ในการติดต่อกับหน่วยความจำแบบ Flash ที่ใช้เก็บโปรแกรม

และหน่วยความจำ SRAM ที่ใช้เก็บข้อมูลส่วนที่ใช้ในการติดต่อกับหน่วยความจำภายนอก มีการติดต่อผ่านบัส AMBA AHB (Advanced High-performance Bus)

3.3.2.3 การจัดหน่วยความจำของ LPC2138

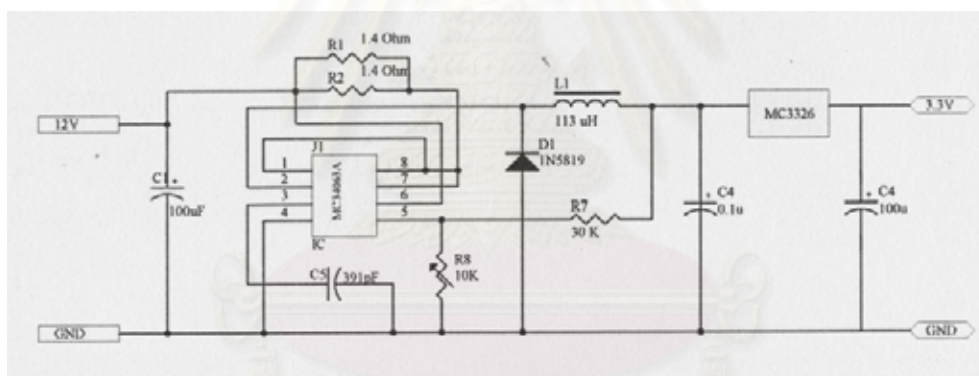
เนื่องจาก ARM7 เป็นซีพียูขนาด 32 บิต ที่มีขนาดแอดเดรสต่อกับหน่วยความจำ จำนวน 32 เส้น ทำให้สามารถอ้างหน่วยความจำได้ถึง 4 Gigabit ($2^{32} = 4 \text{ GB}$) ตัวแกนหลักของ ARM7TDMI จะมีสถาปัตยกรรมแบบ Von Neumann ที่ใช้บัสขนาด 32 บิต ชุดเดียวกันสำหรับตัวคำสั่งของโปรแกรมและข้อมูล โดยมีแค่คำสั่ง Load, Store และ Swap เท่านั้นที่ใช้คำสั่งเดียวกันกับการใช้คำสั่งจัดการเกี่ยวกับหน่วยความจำ การติดต่อกับพอร์ตอินพุตหรือเอาต์พุตก็จะใช้คำสั่งเดียวกันกับการใช้คำสั่งจัดการเกี่ยวกับหน่วยความจำในไมโครคอนโทรลเลอร์ LPC2148 ได้จัดสรรหน่วยความจำโดยรวม ดังรูปที่ 3.19



รูปที่ 3.19 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ LPC2138

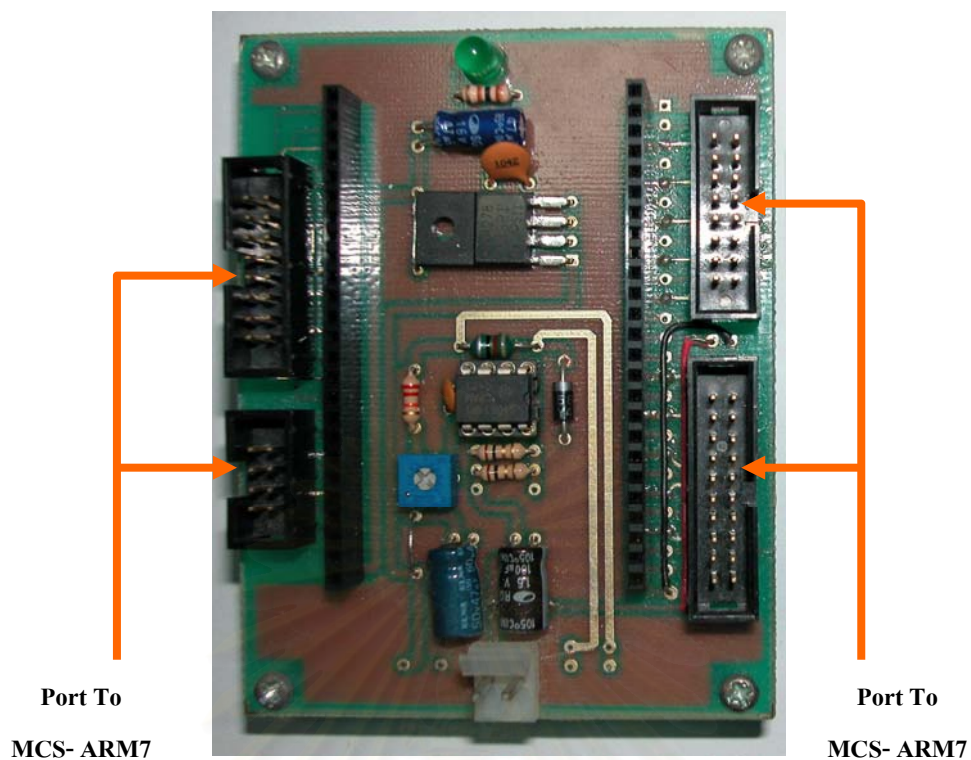
ET-ARM7 STAMP LPC2138 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล ARM7TDMI-S Core ซึ่งเลือกใช้ไมโครคอนโทรลเลอร์ 16/32-Bit ขนาด 64 Pin แบบใช้พลังงานต่ำเป็น MCU ประจำบอร์ด ซึ่งบอร์ดนี้เลือกใช้ MCU เบอร์ LPC2138 ของ Philips โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องการจัดวางบอร์ดให้มีขนาดเล็กเพื่อให้ง่ายต่อการนำไปประยุกต์ใช้งาน โดยได้นำ MCU มาจัดวางร่วมกับอุปกรณ์พื้นฐานที่จำเป็น และจัดขาออกมาให้ใช้งานภายนอกซึ่งการจัดเรียงขาสัญญาณจะทำการจัดเรียงอย่างเป็นระเบียบ เพื่อให้สามารถต่อใช้งานได้โดยสะดวก ตัวบอร์ดใช้ไฟ +3.3 โวลต์ สามารถรองรับ I/O ที่เป็นสัญญาณ 5 โวลต์ ได้ตัวบอร์ดมี Connector UART0 (RS-232) จำนวน 1 Port สำหรับทำการ Download Hex File หรือ ใช้งานในการสื่อสาร RS232 ในโปรแกรม Application ที่เขียนขึ้นเอง

โดยการจัดวางขาและแหล่งจ่าย ไฟฟ้าจะถูกออกแบบมาให้ด้วย แต่เนื่องจากแผ่นวงจรแหล่งจ่ายไฟฟ้าขนาด +3.3 โวลต์ ที่ออกแบบมามีขนาดที่ใหญ่เกินไปไม่เหมาะสม จึงได้ทำการออกแบบใหม่ในส่วนของแหล่งจ่ายไฟฟ้า +3.3 โวลต์ และออกแบบในส่วนของการต่อใช้งานของพอร์ต ARM7 กับ จอ LCD และ CPLD การออกแบบแสดงได้ดังรูปที่ 3.20

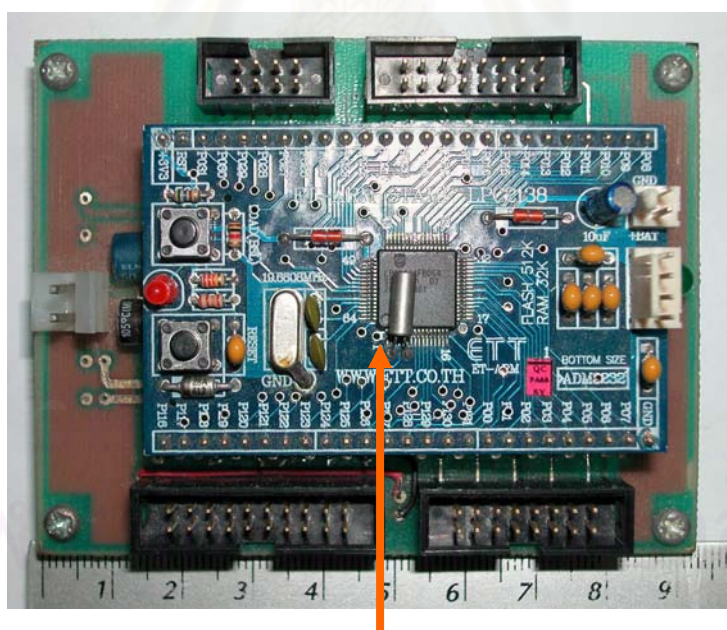


รูปที่ 3.20 แหล่งจ่ายไฟฟ้า +3.3 โวลต์ของไมโครคอนโทรลเลอร์ LPC2138

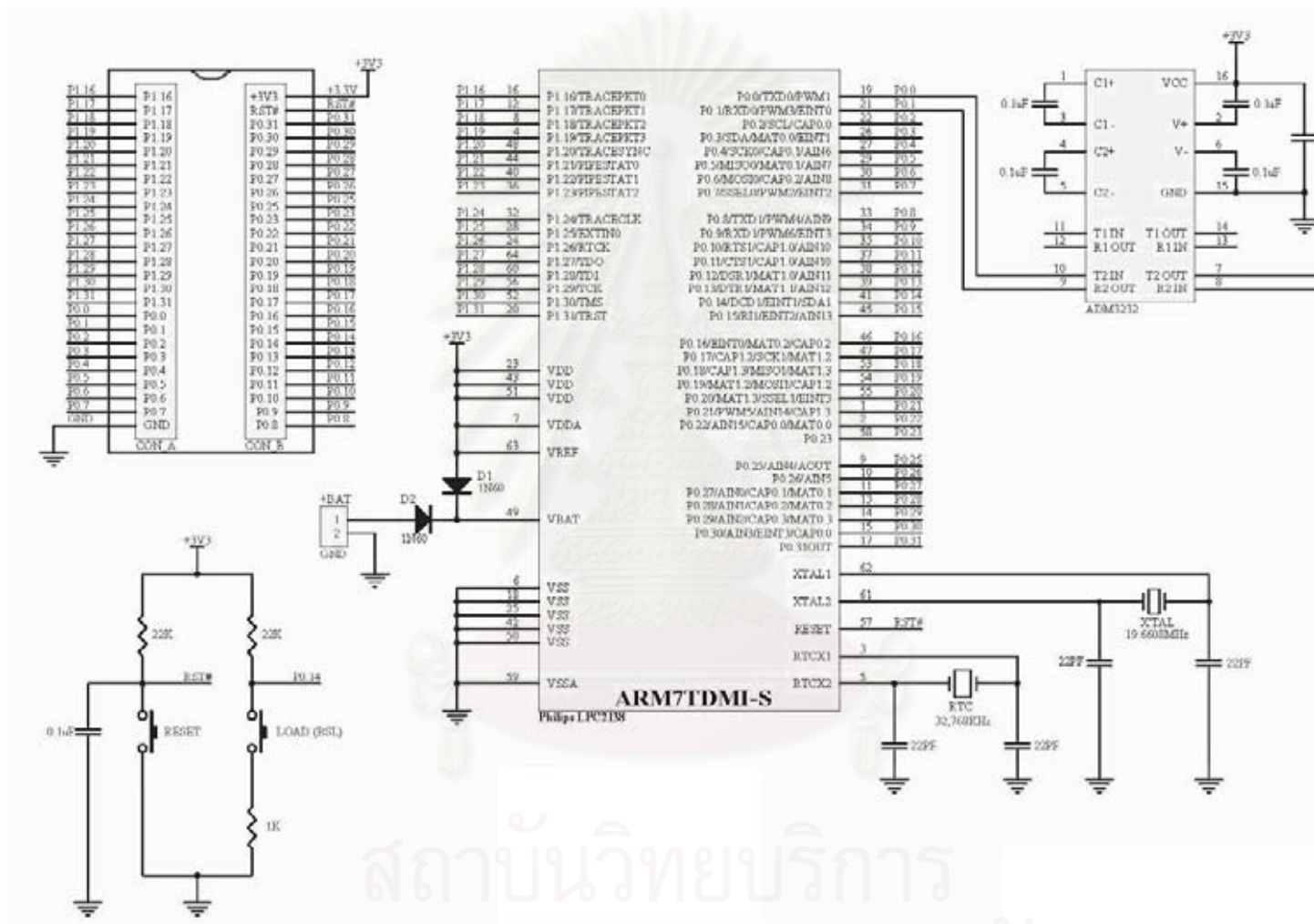
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.21 แผ่นวงจรแหล่งจ่ายไฟฟ้า +3.3 โวลต์และพอร์ตต่อใช้งาน



รูปที่ 3.22 แผ่นวงจรของไมโครคอนโทรลเลอร์ LPC2138



รูปที่ 3.23 วงจรของไมโครคอนโทรลเลอร์ LPC2138

3.3.3 ระบบการแสดงผล

3.3.3.1 คุณสมบัติเฉพาะของจอแสดงผล LM32C041 ยี่ห้อ SHARP

เป็นจอขนาด 5.5 นิ้ว แสดงสีด้วย Color Liquid Indicator ความละเอียดจำนวนสีมีขนาด 320x240xRGB (Red, Green, Blue) ในแต่ละสีแสดงความสว่างได้ 3 บิต ดังนั้นจึงสามารถแสดงสีได้ทั้งหมด 512 สี จอแสดงผลนี้ต้องการแหล่งจ่ายไฟฟ้า 3 ระดับคือ +5 โวลต์ สำหรับจ่ายให้กับส่วนที่เป็นลอจิก +30 โวลต์ จ่ายให้กับส่วนของการแสดงสีในจอ LCD และ +1000 โวลต์ จ่ายให้กับหลอด BACK LIGHT ที่อยู่ด้านหลังของจอ LCD

ตารางที่ 3.1 คุณสมบัติทางไฟฟ้าของจอ LCD32C041

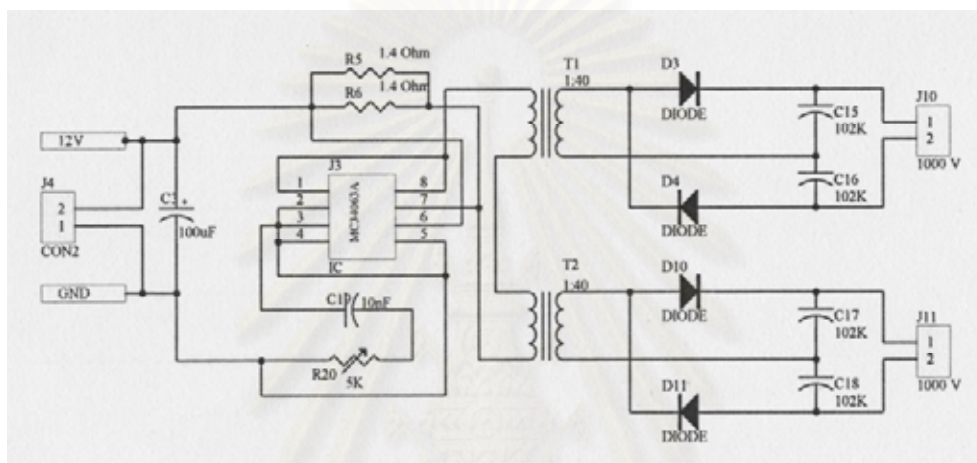
Item	sign	Min	Type	Max	Unit
Logic voltage	VDD	4.25	5	5.25	V
LCD drive voltage	VEE	28.5	30	31.54	V
"H" level input volt.	VIH	3.5		VDD	V
"L" level input volt.	VIL	0		0.8	V
Logic consumption current	IDD	-	82	164	mA
LCD consumption current	IEE	-	11	20	mA
Consumption power	Pd	-	740	1420	mW

ตารางที่ 3.2 คุณสมบัติเวลาในการอินเทอร์เฟสของจอ LCD32C041

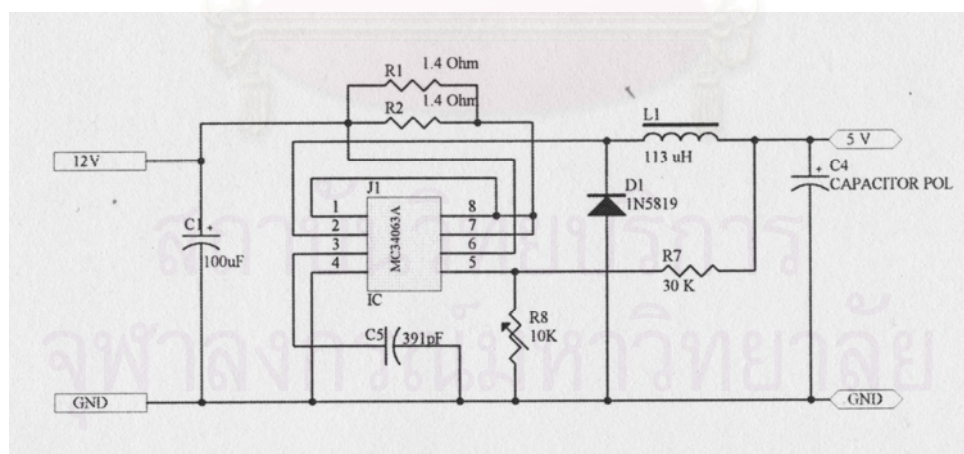
Item	sign	Rating value			Unit
		MIN	TYP	MAX	
Frame cycle	tFRM	12.5	-	20	ms
Clock cycle	Tck	100	-	-	ns
HSYNC "H" level width	th	376 DCK	-	-	-
VSYNC "H" level width	tv	248 HSYNC	-	-	-
"H" Level clock width	tWCKH	40	-	-	ns
"L" Level clock width	tWCKL	40	-	-	ns
"L" Level HSYNC width	tWLh	3	-	-	DCK
"L" Level VSYNC width	tWLv	3	-	-	HSYNC
HSYNC to DCK margin	thd	20	-	-	ns
VSYNC to HSYNC margin	tvh	20	-	-	ns
Data setup time	tDS	30	-	-	ns
Data hold time	tDH	20	-	-	ns
Clock rise & fall time	tr, tf	-	-	30	ns

3.3.3.2 การออกแบบแหล่งจ่ายไฟฟ้าให้กับจอแสดงผล

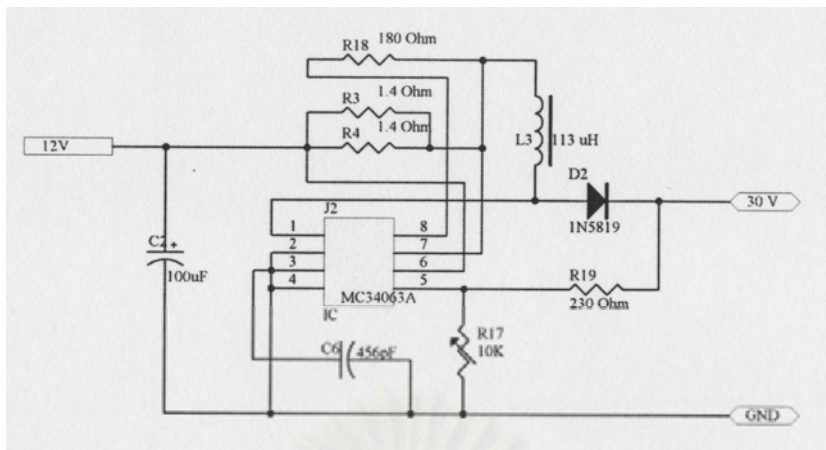
การใช้งานของจอ LCD รุ่น LM32C041 จะต้องมีแหล่งจ่ายไฟฟ้าศักดาสูงขนาด +1000 โวลต์และแหล่งจ่ายไฟฟ้าศักดาต่ำขนาด + 5 โวลต์และ+30 โวลต์ จึงได้ทำการออกแบบระบบจ่ายไฟฟ้าศักดาสูงและระบบจ่ายไฟฟ้าศักดาต่ำ ดังแสดงในรูปที่ 3.24 , 3.25 และรูปที่ 3.26 โดยไฟฟ้าศักดาสูง จะถูกจ่ายให้กับหลอด BACK LIGHT ส่วนไฟฟ้าศักดาต่ำ +5 โวลต์ จ่ายแรงดันให้กับลอจิก (Logic voltage) และไฟฟ้าศักดาต่ำ +30 โวลต์ จ่ายให้กับ LCD Drive Voltage



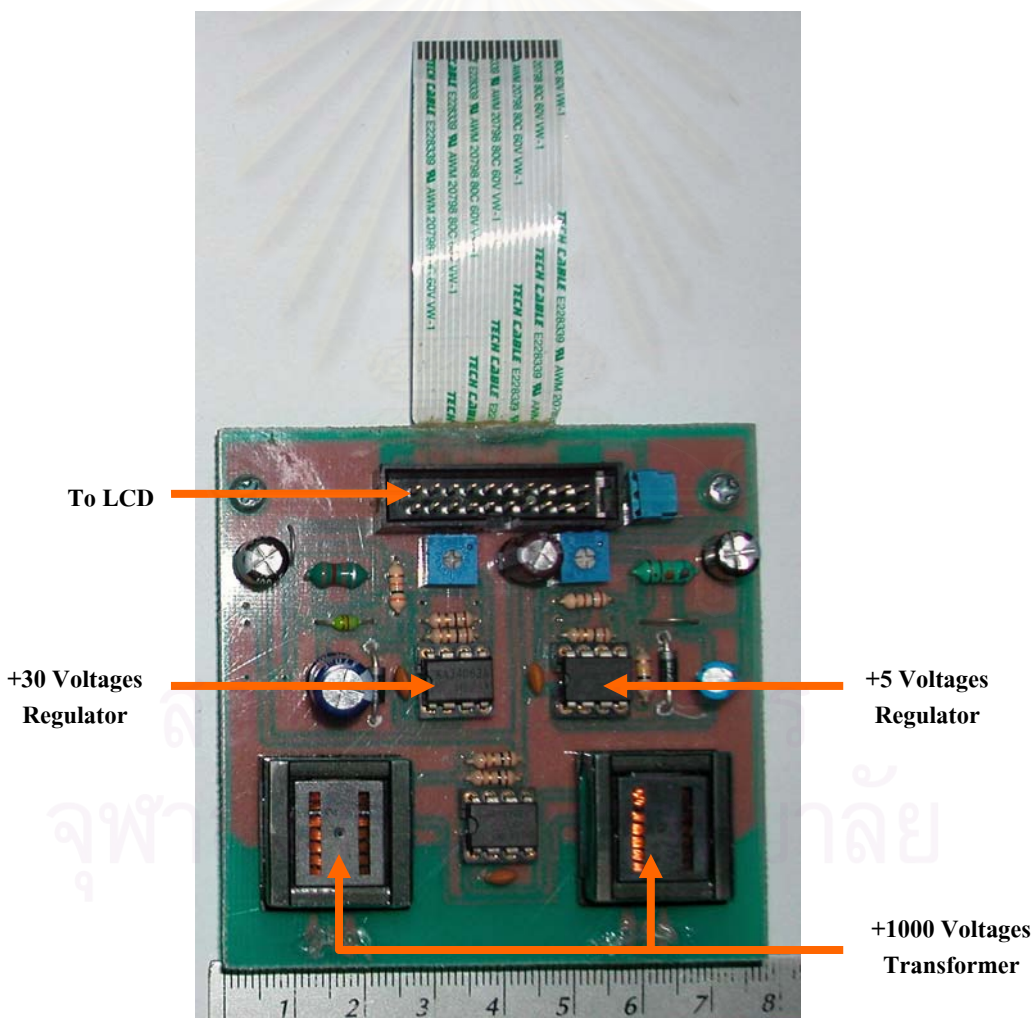
รูปที่ 3.24 วงจรการจ่ายไฟฟ้าศักดาสูง +1000 โวลต์ ให้กับจอแสดงผล



รูปที่ 3.25 วงจรการจ่ายไฟฟ้าศักดาต่ำ +5 โวลต์



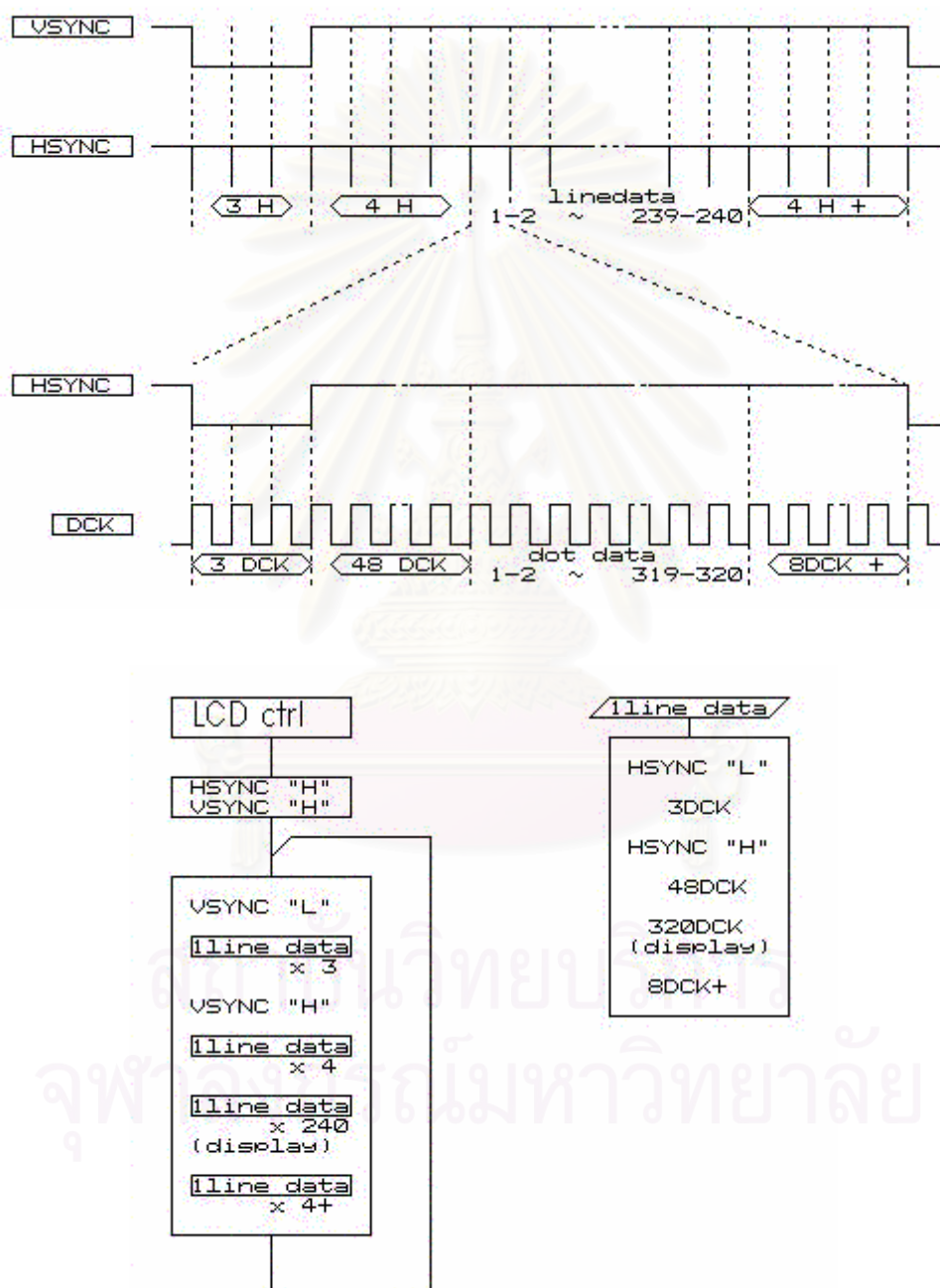
รูปที่ 3.26 วงจรการจ่ายไฟฟ้าสัปดาห์ต่ำ +30 โวลต์



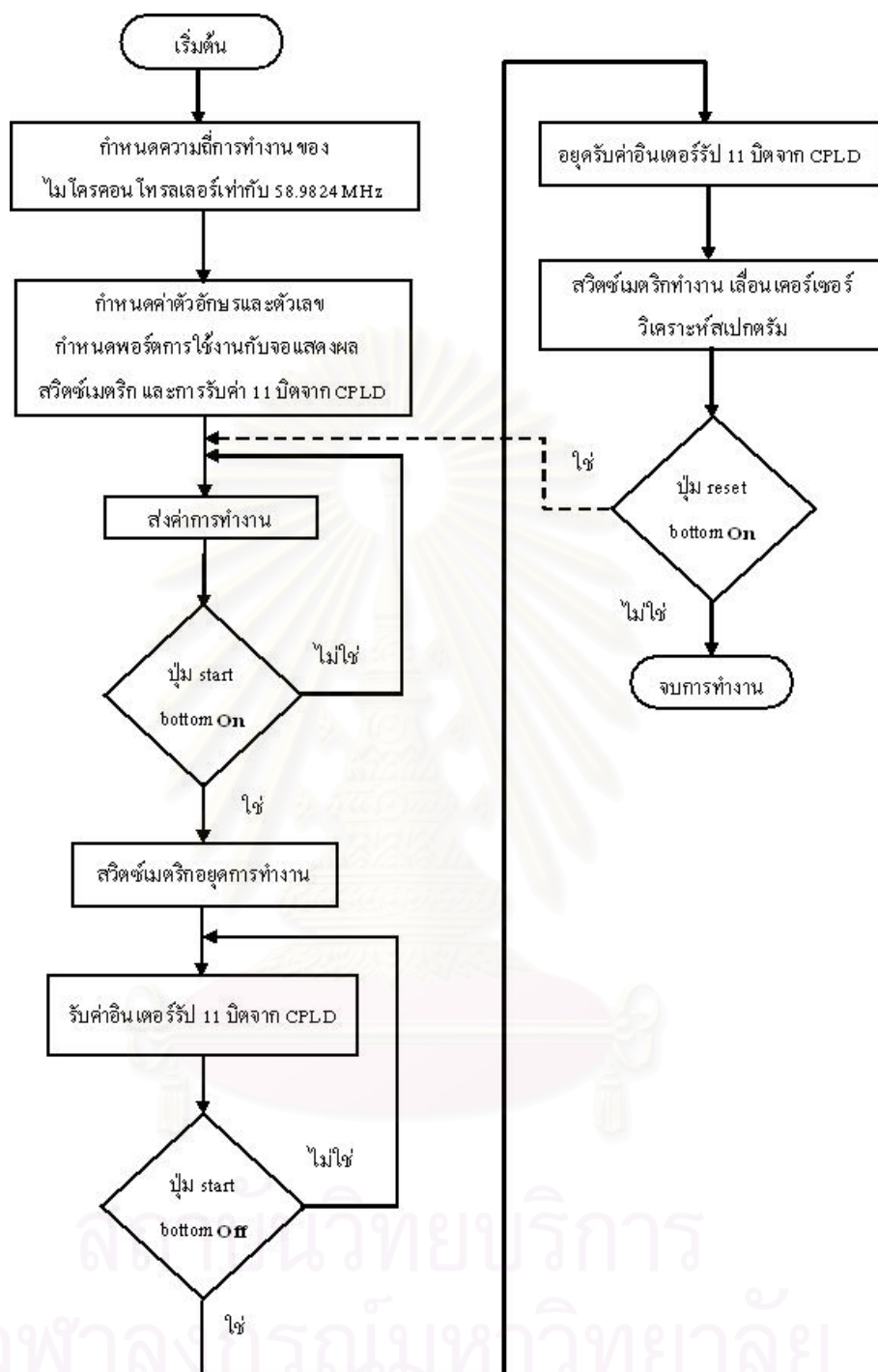
รูปที่ 3.27 แผ่นวงจรแหล่งจ่ายไฟฟ้า +5 โวลต์ +30 โวลต์ และ +1000 โวลต์

3.3.3.3 การเขียนโปรแกรมควบคุมการแสดงผล [4]

การควบคุมให้จุดสีแต่ละจุดที่อยู่บนหน้าจอ LCD ติดหรือดับนั้นจำเป็นต้องส่งสัญญาณข้อมูลให้ถูกต้องตามจังหวะของสัญญาณ VSYNC (Vertical Sync.) HSYNC (Horizontal Sync.) และ DCK (Dot Clock) ดังแสดงในรูปที่ 3.28



รูปที่ 3.28 แสดงการส่งสัญญาณ VSYNC, HSYNC และ DCK

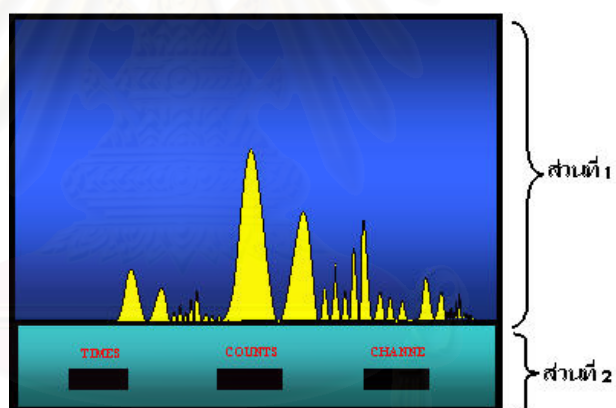


รูปที่ 3.29 ฟังก์ชันการทำงานของโปรแกรมสำหรับในการควบคุมจอแสดงผล

หน้าจอแสดงผลของจอ LCD จะมีขนาดกว้าง 320 จุด ยาว 240 จุด แบ่งเป็น 2 ส่วนโดยส่วนแรกให้ทำการแสดงภาพสเปกตรัมในส่วนที่สองจะทำการแสดงค่าการนับสเปกตรัมรวมทั้งค่าเวลาและตำแหน่งช่องที่ทำการวิเคราะห์ ดังแสดงในรูปที่ 3.30



รูปที่ 3.30 ภาพของจอแสดงผล LCD



รูปที่ 3.30 จอภาพ LCD ที่แสดงผลสเปกตรัมรังสี

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

การทดสอบสมรรถนะของระบบ

เครื่องวิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิควิดิคินสันเอ็ดจีซีที่พัฒนาได้ จะใช้ CPLD ในการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลและใช้ไมโครคอนโทรลเลอร์ในการควบคุมการจัดเก็บข้อมูลจากการแปลงเป็นสัญญาณดิจิทัลลงในหน่วยความจำภายในไมโครคอนโทรลเลอร์ พร้อมทั้งควบคุมการแสดงผลออกทางจอ LCD ด้วยการพัฒนาโปรแกรมควบคุมลำดับขั้นการใช้งานที่ง่าย หลังจากประกอบเครื่องสมบูรณ์ได้ทำการทดสอบการทำงานของระบบต่าง ๆ ดังนี้

1. ทดสอบระบบในแต่ละส่วน ได้แก่ การทดสอบสมรรถนะของแหล่งจ่ายไฟฟ้าศักดาสูง การทดสอบสมรรถนะของวงจรขยายสัญญาณพัลส์ การทดสอบสมรรถนะของวงจรแปลงสัญญาณอนาล็อกเป็นดิจิทัลที่ใช้เทคนิควิดิคินสันเอ็ดจีซี การทดสอบสมรรถนะของวงจรควบคุมระบบการทำงานและการแสดงผล
2. การทดสอบอุปกรณ์วิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิควิดิคินสันเอ็ดจีซี

4.1 การทดสอบระบบในแต่ละส่วน

หลังจากออกแบบระบบวิเคราะห์รังสีแบบหลายช่องแล้วจึงได้มีการทดสอบสมรรถนะการทำงานของระบบ เพื่อประเมินประสิทธิภาพในการทำงานและหาข้อบกพร่องของระบบวิเคราะห์รังสีแบบหลายช่องในส่วนต่างๆ ซึ่งมีผลการทดสอบ ดังนี้

4.1.1 การทดสอบแหล่งจ่ายไฟฟ้าศักดาสูง

หลังจากออกแบบแหล่งจ่ายไฟฟ้าศักดาสูงขนาด 0–2000 V. เพื่อใช้งานสำหรับจ่ายแรงดันไฟฟ้าศักดาสูงให้กับหัววัดรังสีซินทิลเลชัน โดยสามารถจ่ายกระแสได้สูงสุดที่ 1 mA. ที่แรงดัน 2000V. ในการทดสอบจะทดสอบความเป็นเชิงเส้น ทดสอบความเรียบของสัญญาณ (Ripple) ทดสอบ Stability และทดสอบ Load Regulation

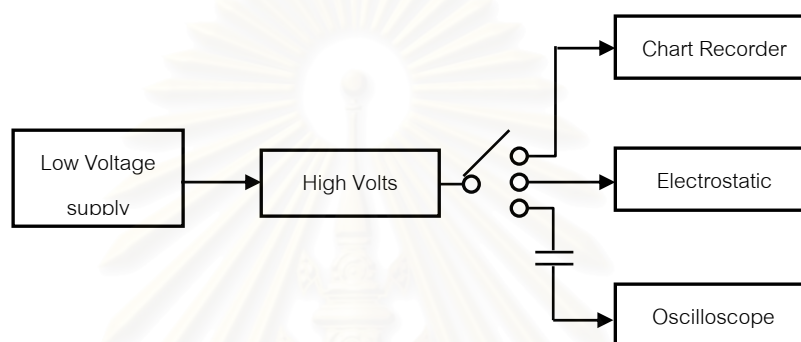
4.1.1.1 เครื่องมือและอุปกรณ์

1. แหล่งจ่ายแรงดันไฟฟ้าศักดาต่ำขนาด +5, ±12 และ ±24 โวลต์ ที่พัฒนาขึ้น
2. อิเล็กโตรสแตติก โวลต์มิเตอร์ของ SINGER COMPANY รุ่น ESD.7

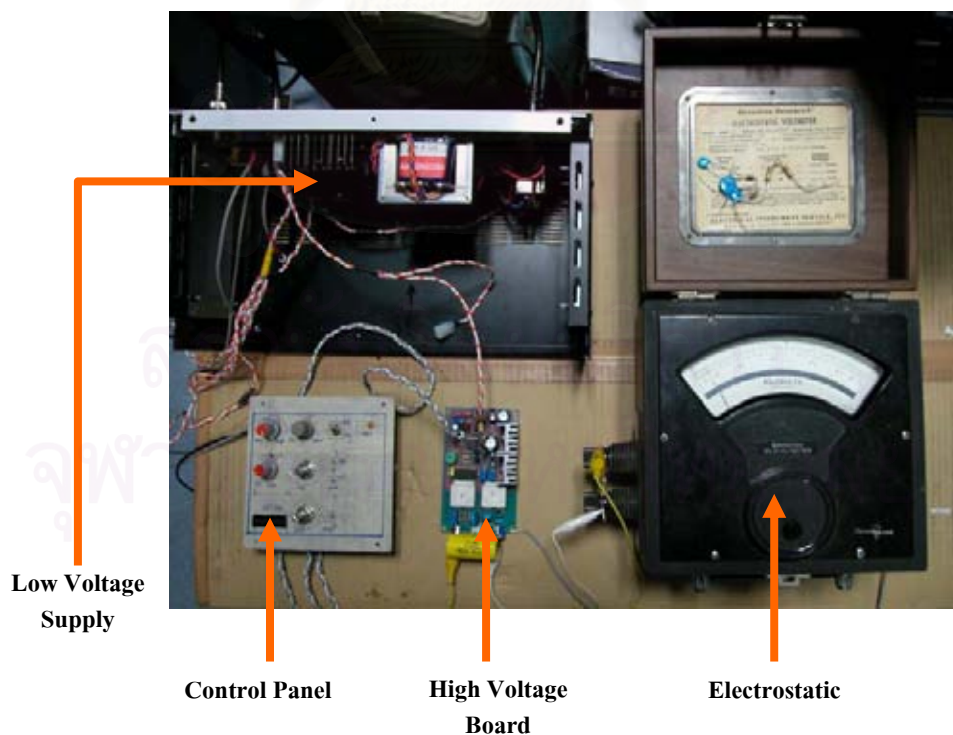
3. ออสซิลโลสโคปของ TEKTRONIX รุ่น TDS 360
4. Chart Recorder ยี่ห้อ FUJITSU รุ่น M3349B
5. ตัวต้านทานชนิดธรรมดาขนาด 0.5 วัตต์ค่าต่าง ๆ
6. แผงวงจรแรงดันไฟฟ้าสัปดาห์สูง 0–2000 โวลต์ ที่พัฒนาขึ้น

4.1.1.2 ขั้นตอนการทดสอบ

สามารถจัดอุปกรณ์สำหรับทดสอบสมรรถนะและการทำงานของแหล่งจ่ายไฟฟ้าสัปดาห์สูงได้ดังรูปที่ 4.1



รูปที่ 4.1 แผนภาพการจัดอุปกรณ์ทดสอบแหล่งจ่ายไฟฟ้าสัปดาห์สูง



รูปที่ 4.2 ภาพการจัดอุปกรณ์ทดสอบแหล่งจ่ายไฟฟ้าสัปดาห์สูงโดยใช้อิเล็กทรอนิกส์

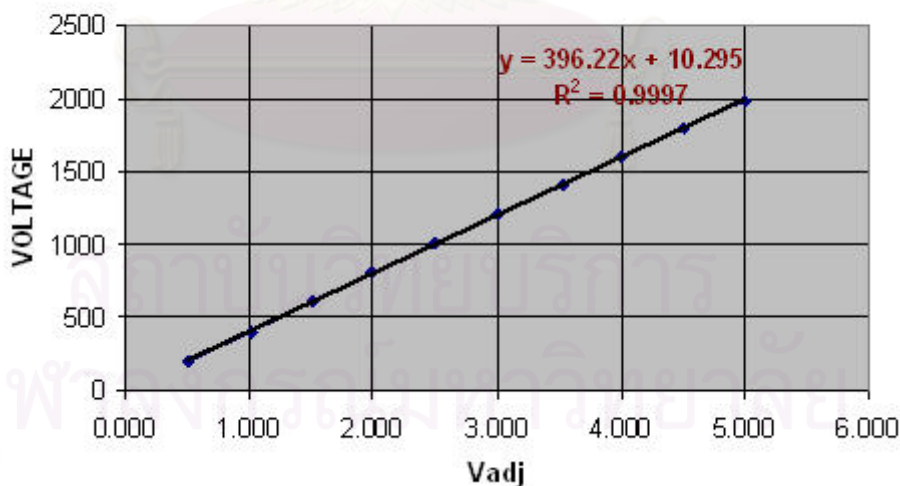
4.1.1.3 ผลการทดสอบ

การทดสอบความเป็นเชิงเส้นของสัญญาณที่ได้จากแหล่งจ่ายไฟฟ้าศักดาสูงจะมีค่าที่ได้ในตารางดังรูปที่ 4.1 ค่าความต้านทานที่ใช้ในการทดสอบมีค่าเท่ากับ 2 M Ω ประสิทธิภาพสูงสุดเท่ากับ 58.8 เปอร์เซ็นต์ ที่แรงดันเท่ากับ 1,980 โวลต์ และกระแสเท่ากับ 1 มิลลิแอมป์

ตารางที่ 4.1 การทดสอบความเป็นเชิงเส้นของสัญญาณที่ได้จากแหล่งจ่ายไฟฟ้าศักดาสูง

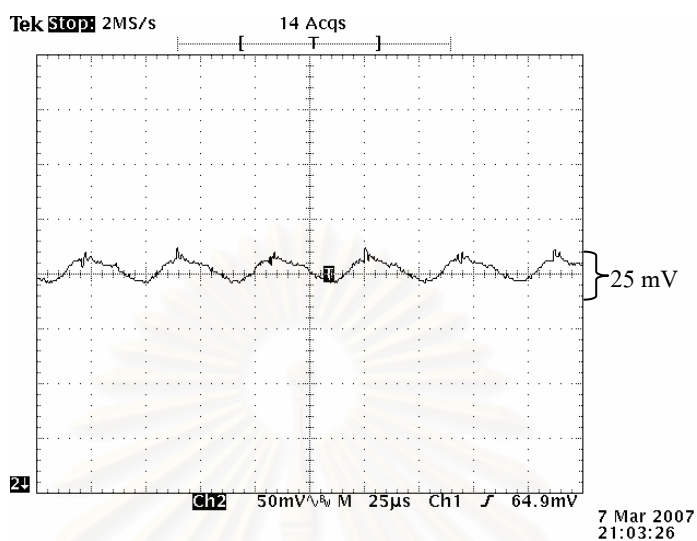
V _{adj}	V _{IN}	I _{IN}	P _{IN}	V _{HV}
0.510	12	0.060	0.72	200
1.009	12	0.090	1.08	400
1.509	12	0.120	1.44	610
2.000	12	0.145	1.74	810
2.500	12	0.180	2.16	1015
3.000	12	0.200	2.40	1210
3.530	12	0.225	2.70	1415
4.000	12	0.245	2.94	1600
4.510	12	0.265	3.18	1790
5.010	12	0.280	3.36	1980

HIGH VOLTS LINEARITY



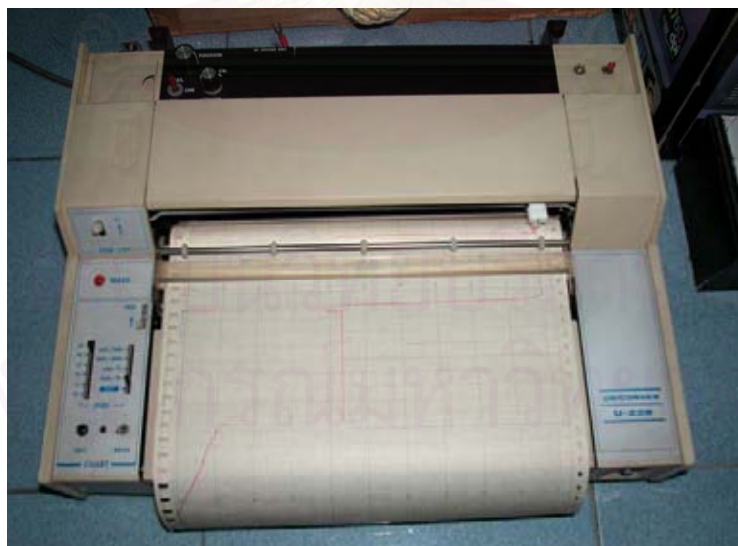
รูปที่ 4.3 แสดงค่าความเป็นเชิงเส้นของแหล่งจ่ายไฟฟ้าศักดาสูง

การทดสอบความเรียบ (Ripple) ของสัญญาณที่ได้จากแหล่งจ่ายไฟฟ้าสัปดาห์สูงโดยใช้
ออสซิลโลสโคปวัดจะมีค่าที่ได้ดังรูปที่ 4.4

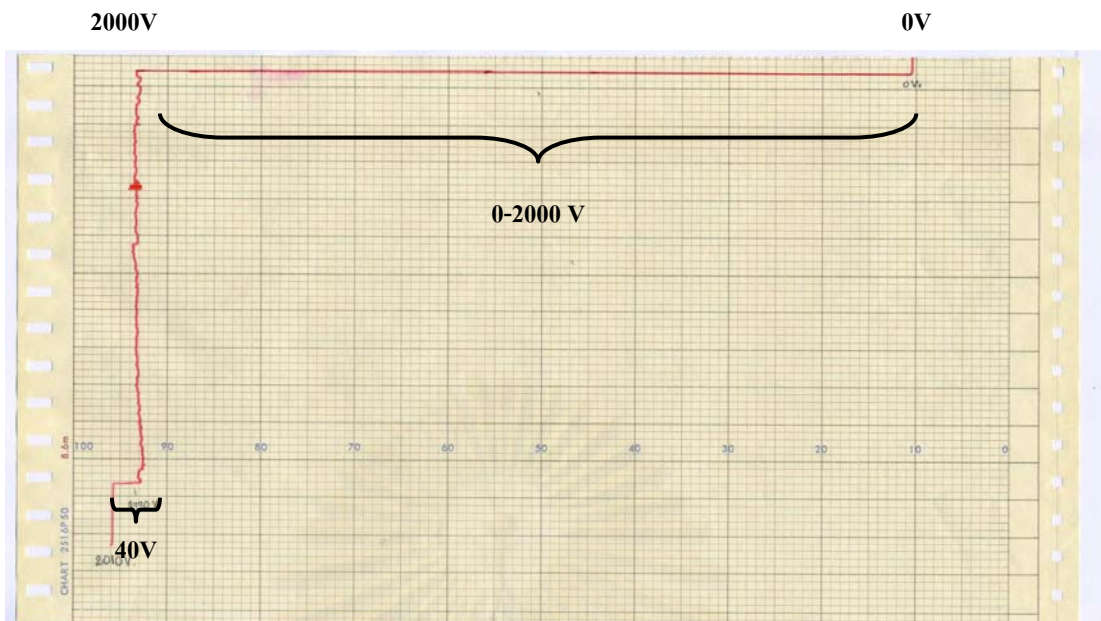


รูปที่ 4.4 แสดงค่าความเรียบ (Ripple) ของแหล่งจ่ายไฟฟ้าสัปดาห์สูง

การทดสอบความ Stability ของสัญญาณที่ได้จากแหล่งจ่ายไฟฟ้าสัปดาห์สูงโดยใช้ Chart Recorder วัดดังรูปที่ 4.5 และค่าที่วัดได้อยู่ในกราฟรูปที่ 4.6



รูปที่ 4.5 ภาพการทดสอบแหล่งจ่ายไฟฟ้าสัปดาห์สูงโดยใช้ Chart Recorder



รูปที่ 4.6 กราฟแสดงค่าความ Stability ของแหล่งจ่ายไฟฟ้าศักดาสูง

การทดสอบ Load regulation ของสัญญาณที่ได้จากแหล่งจ่ายไฟฟ้าศักดาสูงโดยจะมีค่าที่ได้ดังตารางในรูปที่ 4.2

ตารางที่ 4.2 แสดงค่า Load regulation ของแหล่งจ่ายไฟฟ้าศักดาสูง

VOLTS	LOAD		0.5 MΩ		1 MΩ		2 MΩ		4 MΩ		6 MΩ	
	V	%ΔV	V	%ΔV	V	%ΔV	V	%ΔV	V	%ΔV	V	%ΔV
0	0	0	0	0	0	0	0	0	0	0	0	0
250	250	0	250	0	250	0	250	0	250	0	250	0
500	500	0	500	0	500	0	500	0	500	0	500	0
750	750	0	750	0	750	0	750	0	750	0	750	0
1000	1000	0	1000	0	1000	0	1000	0	1000	0	1000	0
1250	1250	0	1250	0	1250	0	1250	0	1250	0	1250	0
1500	1380	8%	1480	1.3%	1500	0	1500	0	1500	0	1500	0
1750	1330	2.4%	1480	1.5%	1750	0	1750	0	1750	0	1750	0
2000	1330	3.3%	1560	2.2%	2000	0	2000	0	2000	0	2000	0

4.1.2 การทดสอบวงจรมายสัญญาณพัลส์

จากวงจรมายสัญญาณพัลส์ที่ออกแบบมาเพื่อแต่งรูปสัญญาณพัลส์นิวเคลียร์ที่ได้จากวงจรมายส่วนหน้าซึ่งมีค่าไรส์ไทม์ (Rise time) ที่สั้นประมาณ 5-100 ns และมีค่าเวลาการสลายตัว (Decay time) ยาว 40-400 μs โดยที่ความสูงพัลส์เกิดตามพลังงานของรังสีที่ตกกระทบกับหัววัด

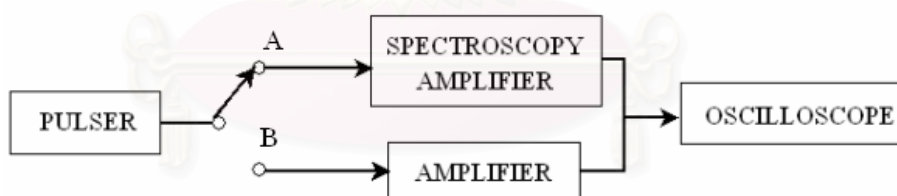
วงจรรขยายสัญญาณพัลส์ที่ออกแบบจึงออกแบบให้มีค่า Shaping time constant (τ) เท่ากับ 1 สัญญาณที่ได้จากวงจรรขยายสัญญาณจึงเป็นแบบ Semi gaussian ซึ่งมีค่าไรส์ไทม์ 2τ และความกว้างพัลส์ขนาด 5τ

4.1.2.1 เครื่องมือและอุปกรณ์

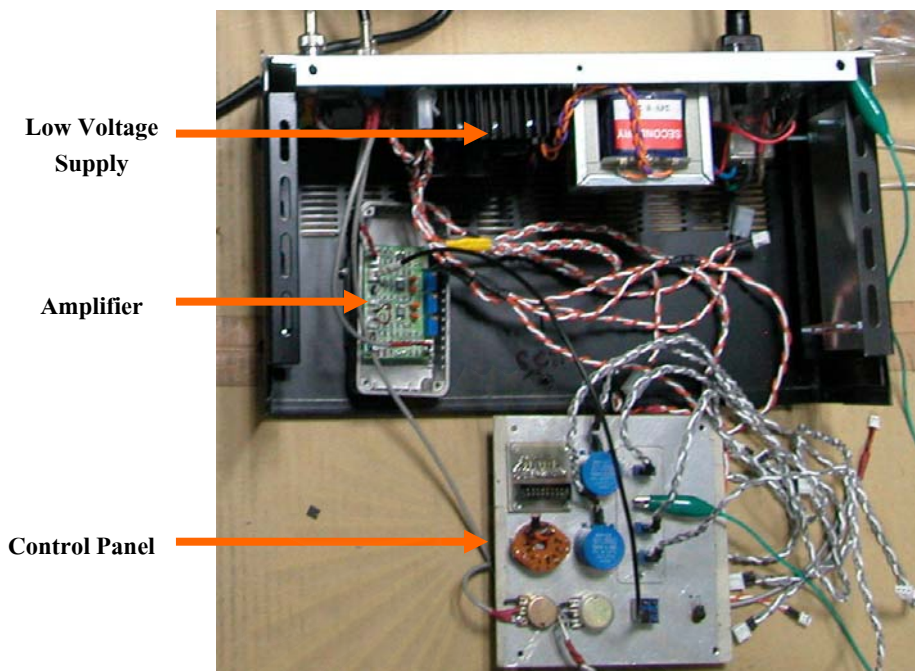
1. แหล่งกำเนิดสัญญาณพัลส์ (Pulser) ของ CANBERRA รุ่น 807
2. แหล่งจ่ายแรงดันไฟฟ้าสัปดาห์ขนาด +5, ± 12 และ ± 24 โวลต์ ที่พัฒนาขึ้น
3. โมดูลขยายสัญญาณพัลส์ (Spectroscopy Amplifier) CANBERRA รุ่น 807
4. ออสซิลโลสโคปของ TEKTRONIX รุ่น TDS 360
5. วงจรรขยายสัญญาณพัลส์ที่พัฒนาได้

4.1.2.2 ขั้นตอนการทดสอบ

จากรูปที่ 4.7 เป็นการจัดอุปกรณ์เพื่อทดสอบวงจรรขยายสัญญาณพัลส์โดยเปรียบเทียบการปรับแต่งรูปสัญญาณเปรียบเทียบกับ โมดูลขยายสัญญาณพัลส์ (Spectroscopy Amplifier) ของ CANBERRA รุ่น 807 และการวัดสัญญาณที่เป็นบวกและสัญญาณลบ



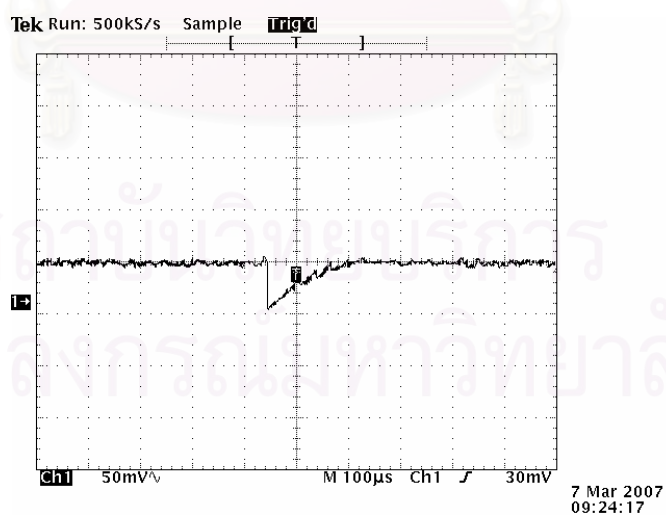
รูปที่ 4.7 แผนภาพการจัดอุปกรณ์ทดสอบวงจรรขยายสัญญาณพัลส์



รูปที่ 4.8 ภาพการจัดอุปกรณ์ทดสอบวงจรขยายสัญญาณพัลส์

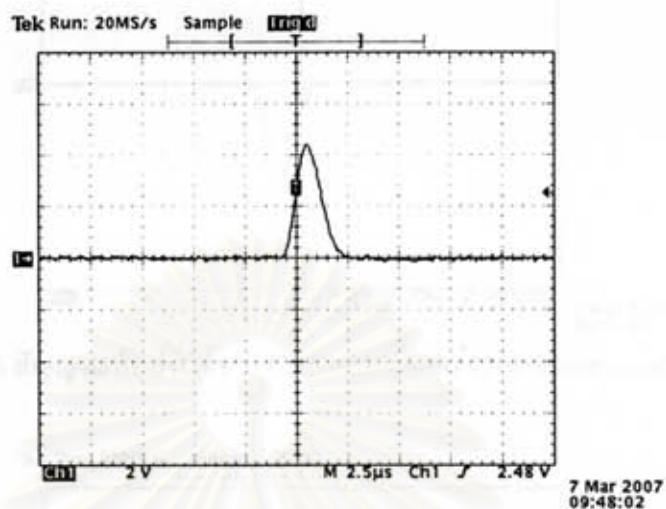
4.1.2.3 ผลการทดสอบ

ซึ่งสัญญาณที่วัดจากโมดูลแหล่งกำเนิดสัญญาณพัลส์ (Pulser) ของ CANBERRA รุ่น 807 วัดได้ดังรูปที่ 4.9 โดยปรับค่าสัญญาณพัลส์ให้มีขนาด 50 mV และปรับอัตราขยายเท่ากับ 100 เท่า

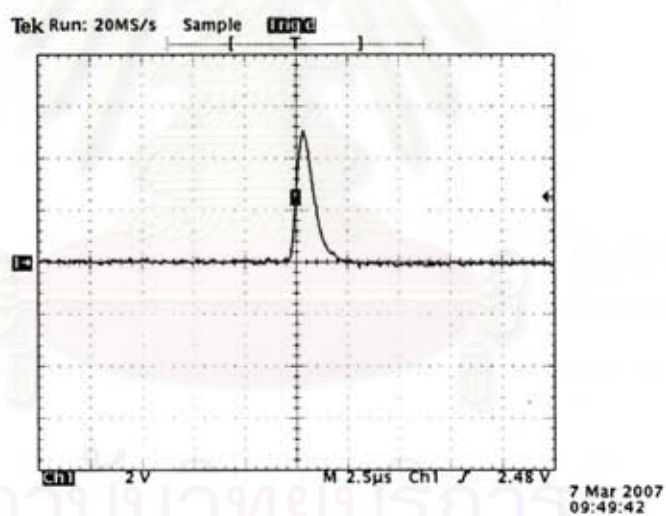


รูปที่ 4.9 สัญญาณพัลส์ที่ได้จากโมดูลแหล่งกำเนิดสัญญาณพัลส์

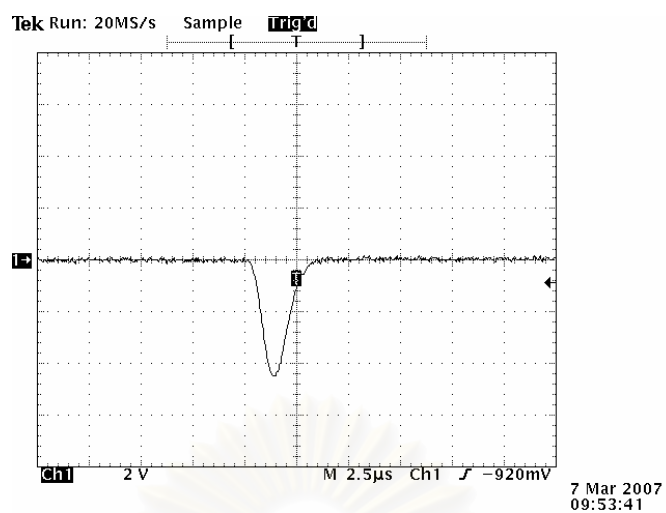
ซึ่งสัญญาณที่วัดจากโมดูลขยายสัญญาณพัลส์ (Spectroscopy Amplifier) CANBERRA รุ่น 2202 วัดได้ดังรูปที่ 4.10



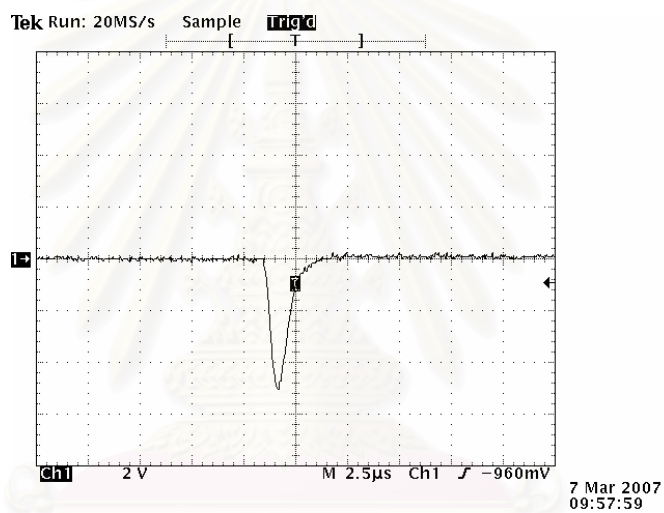
รูปที่ 4.10 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier



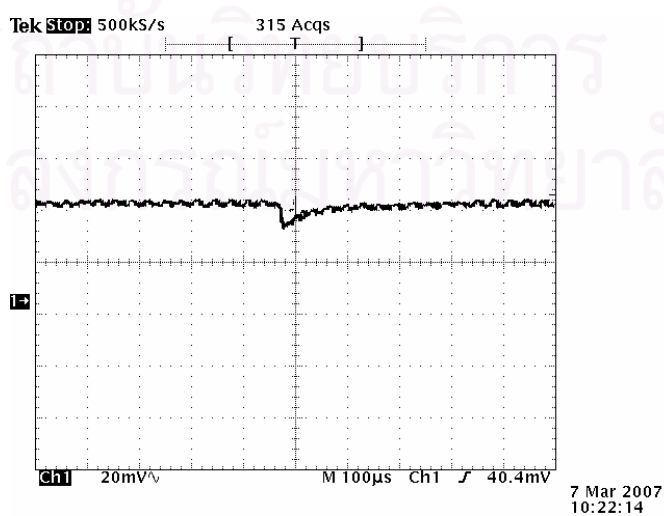
รูปที่ 4.11 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.12 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier

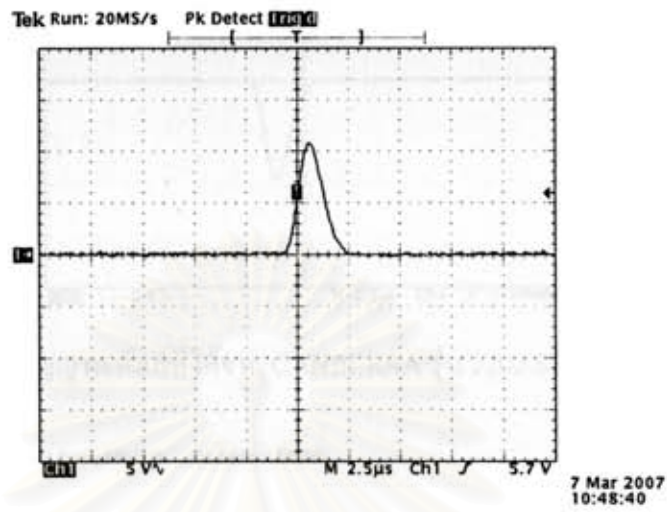


รูปที่ 4.13 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ
จากนั้นปรับค่าสัญญาณพัลส์ให้มีขนาด 10 mV และปรับอัตราขยายเท่ากับ 1000 เท่า

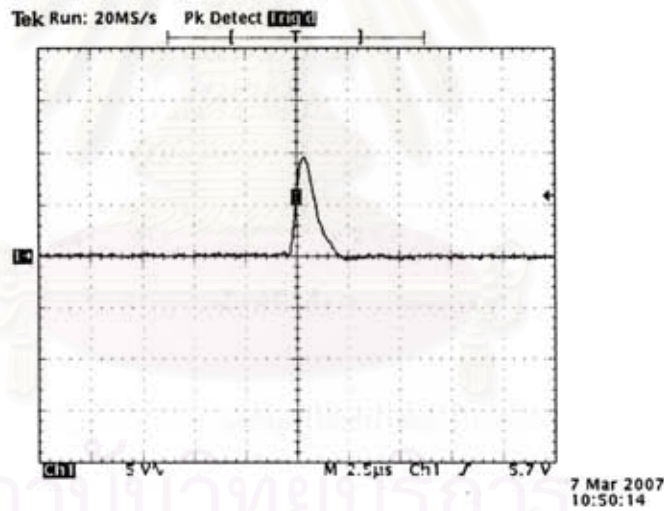


รูปที่ 4.14 สัญญาณพัลส์ที่ได้จากโมดูลแหล่งกำเนิดสัญญาณพัลส์

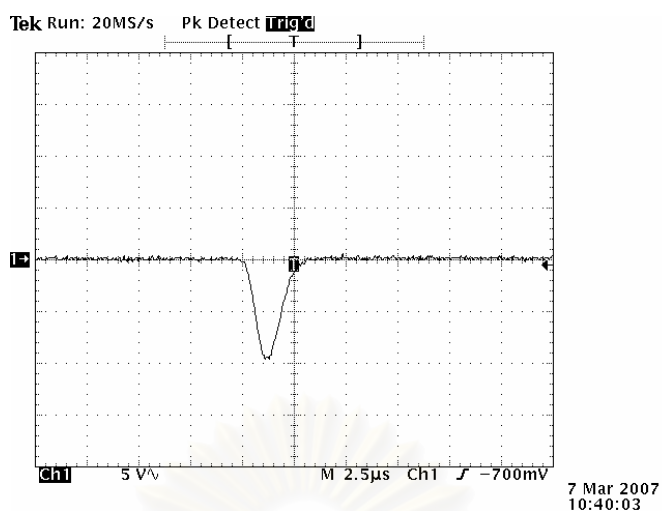
ซึ่งสัญญาณที่วัดจาก โมดูลขยายสัญญาณพัลส์ (Spectroscopy Amplifier) ของ CANBERRA รุ่น 2202 วัดได้ดังรูปที่ 4.15



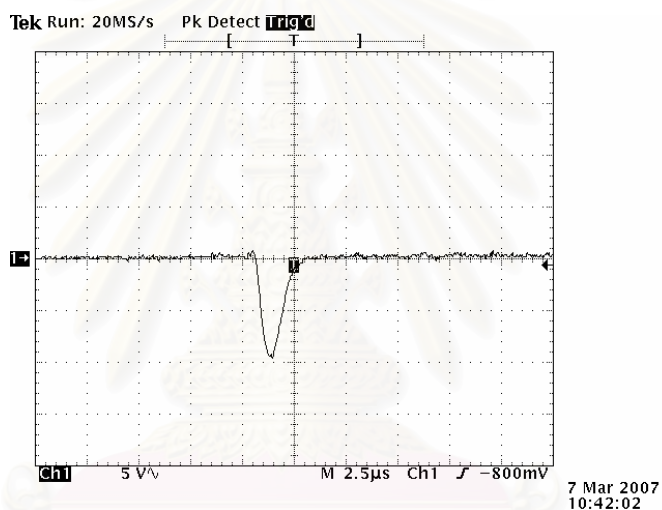
รูปที่ 4.15 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier



รูปที่ 4.16 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.17 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier



รูปที่ 4.18 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ

4.1.3 การทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล

จากการออกแบบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลสามารถทดสอบสมรรถนะของระบบดังนี้

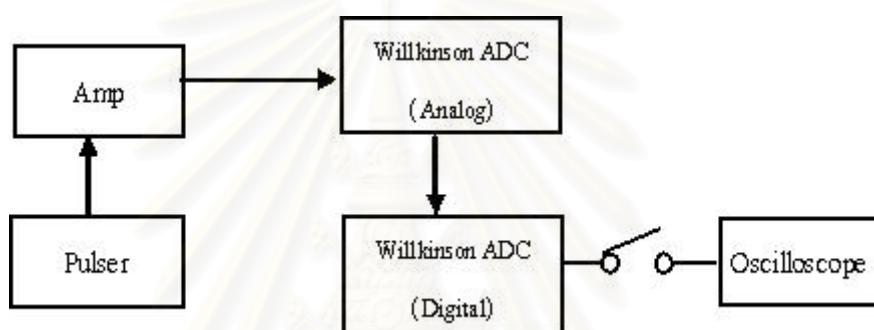
4.1.3.1 เครื่องมือและอุปกรณ์

1. แหล่งจ่ายแรงดันไฟฟ้าสัปดาห์ขนาด +5V, ±12 และ ±24 โวลต์ ที่พัฒนาขึ้น
2. แหล่งกำเนิดสัญญาณพัลส์ (Pulser) ของ CANBERRA รุ่น 807
3. โมดูลขยายสัญญาณพัลส์ (Amplifier) ของ CANBERRA รุ่น 807

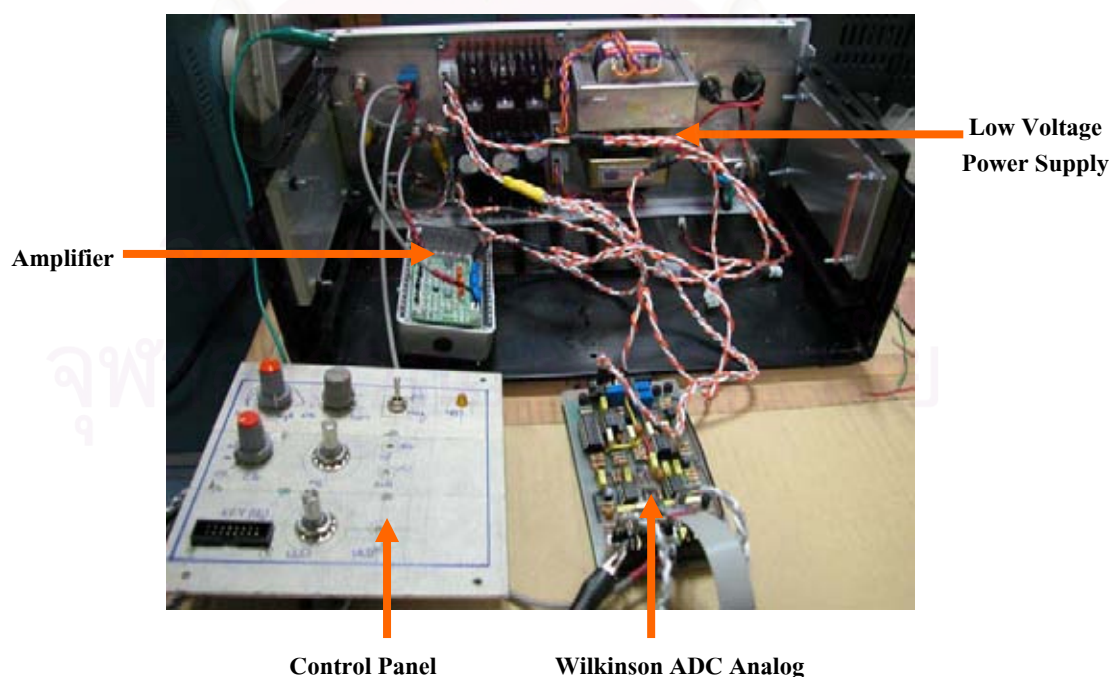
4. วงจรไมโครคอนโทรลเลอร์ที่พัฒนาให้ทำงานเป็นวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลที่ใช้เทคนิควิลคินสันเอดีซีในส่วนของอนาลอก
5. วงจร CPLD ที่พัฒนาให้ทำงานเป็นวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลที่ใช้เทคนิควิลคินสันเอดีซี ในส่วนของดิจิทัล
6. ออสซิลอ스코ปของ TEKTRONIX รุ่น TDS 360

4.1.3.2 ขั้นตอนการทดสอบ

สามารถจัดอุปกรณ์สำหรับการทดสอบได้ดังรูปที่ 4.19



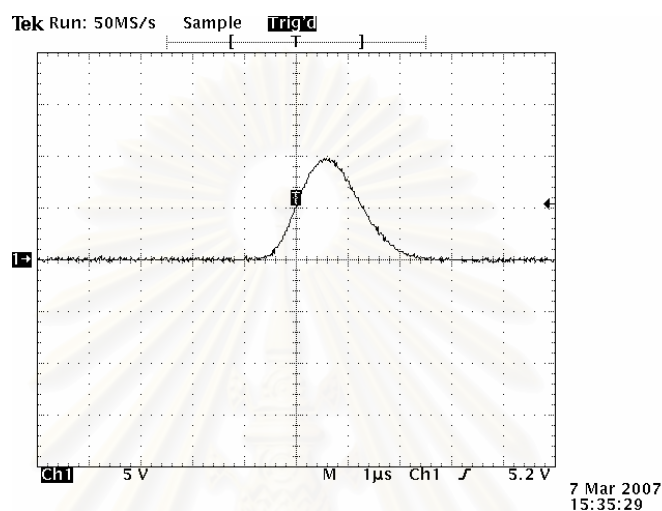
รูปที่ 4.19 แผนภาพการจัดอุปกรณ์ทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล



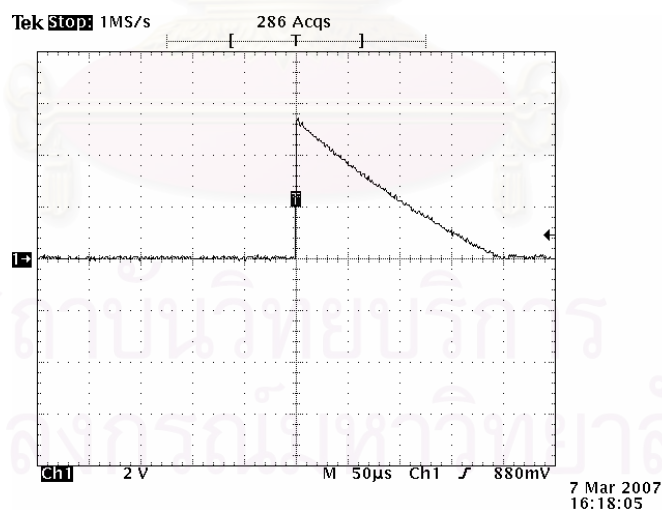
รูปที่ 4.20 ภาพการทดสอบวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลส่วนของอนาลอก

4.1.3.3 การทดสอบในส่วนของอนาล็อก

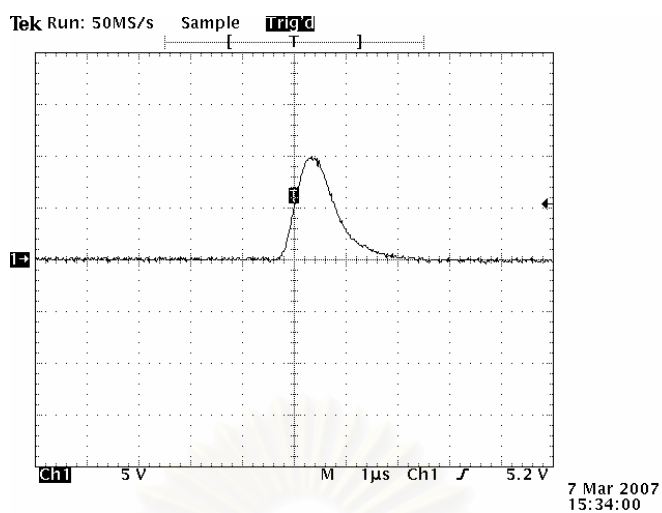
การทดสอบในส่วนของอนาล็อกจะทดสอบการยืดความสูงของพัลส์ให้เป็นค่าของเวลา โดยที่ค่าความสูงต่างกันก็จะมีค่าเวลาที่ยืดได้แตกต่างกันด้วย ทดสอบได้โดยการปรับสัญญาณที่มา จากส่วนการขยายสัญญาณพัลส์ให้ได้ค่าความสูงพัลส์ที่ 10 โวลต์ ดังแสดงในรูปที่ 4.21 และ 4.22 ส่วนรูปที่ 4.23 และ 4.24 แสดงการยืดความสูงของพัลส์ให้เป็นคาบเวลา



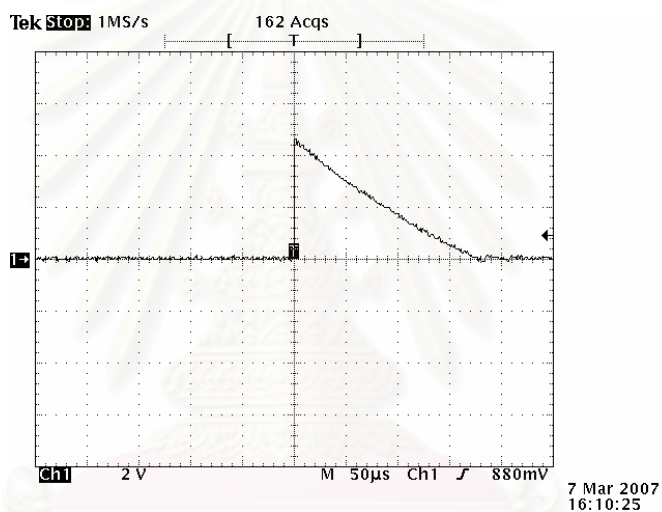
รูปที่ 4.21 สัญญาณพัลส์ที่ได้จาก CANBERRA รุ่น 2202 Spectroscopy Amplifier



รูปที่ 4.22 สัญญาณพัลส์จากโมดูลขยายสัญญาณพัลส์ที่ถูกยืดออก

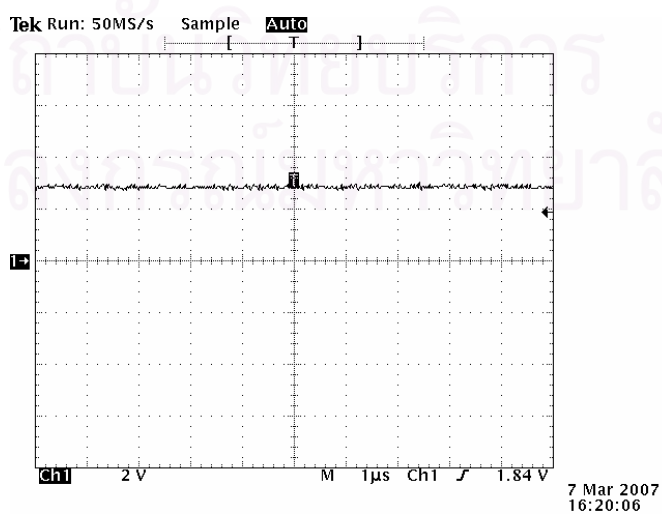


รูปที่ 4.23 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ

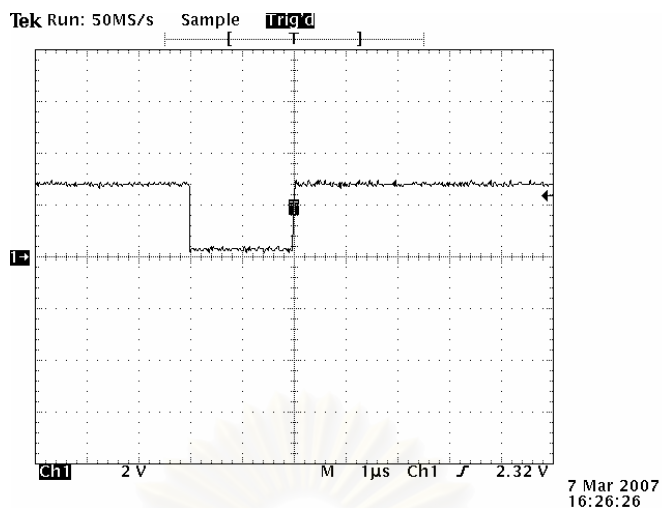


รูปที่ 4.24 สัญญาณพัลส์จากวงจรขยายสัญญาณพัลส์ที่ออกแบบถูกขี้ออก

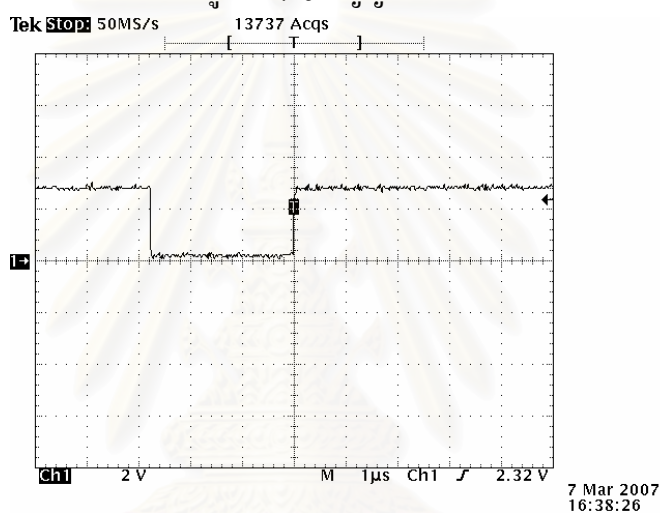
ทำการวัดสัญญาณ ULD, LLD และTHR ได้สัญญาณดังในรูปที่ 4.25, 4.26 และ 4.27 ตามลำดับ



รูปที่ 4.25 สัญญาณULD

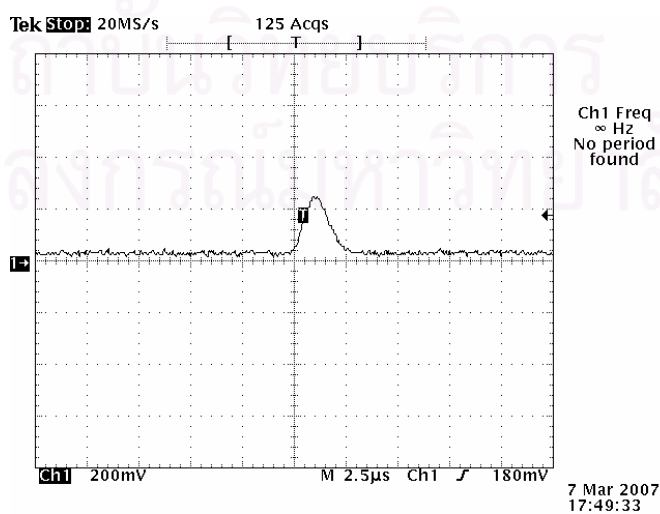


รูปที่ 4.26 สัญญาณLLD

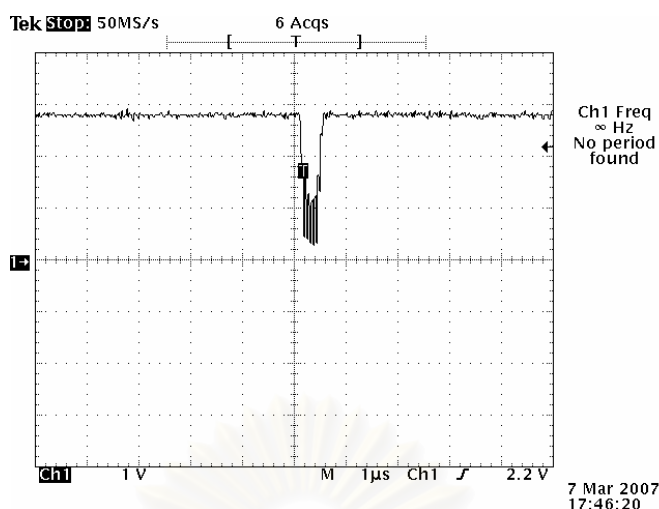


รูปที่ 4.27 สัญญาณTHR

ในการตรวจจับความสูงของพัลส์ที่สามารถปรับที่ THR และ LLD ที่มีค่าความสูงต่ำที่สุดคือค่าความสูงประมาณ 250 mV ดังแสดงในรูปที่ 4.28 และ 4.29



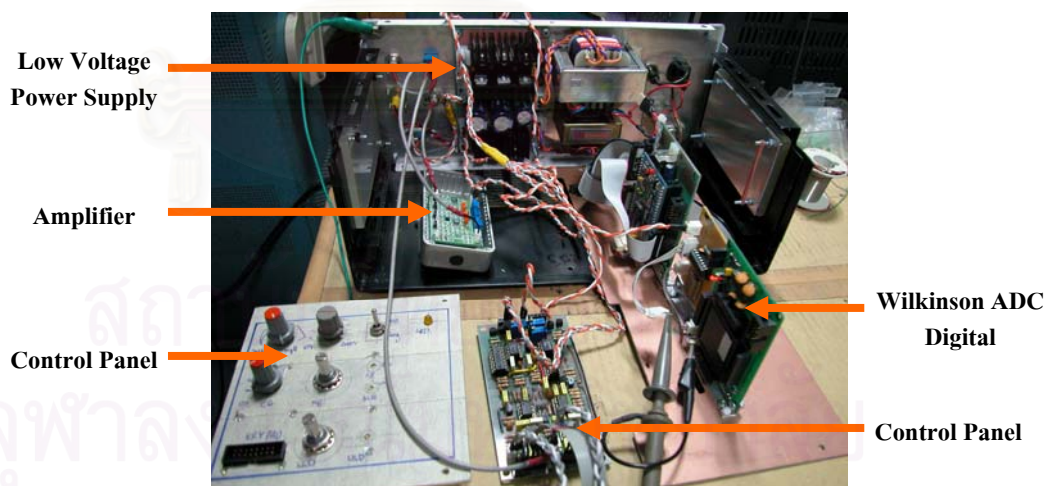
รูปที่ 4.28 แสดงความสูงสัญญาณพัลส์ต่ำสุดที่ THR และ LLD จะสามารถตรวจจับได้



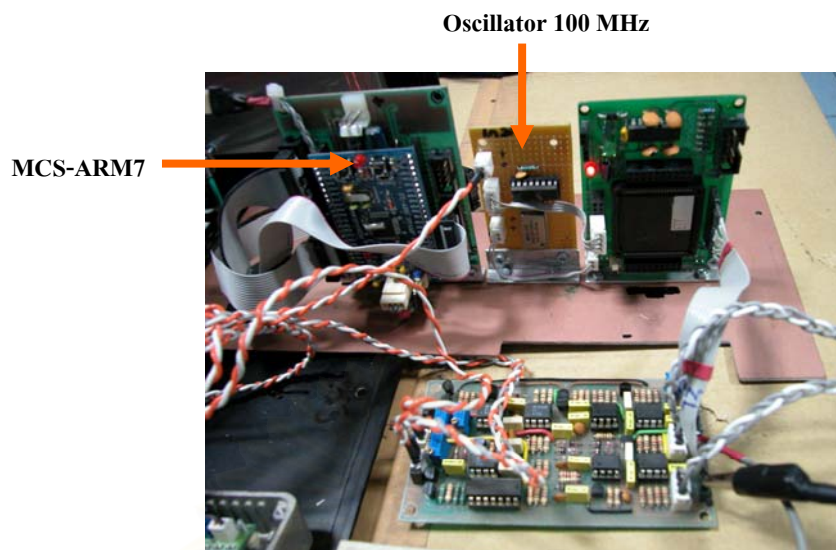
รูปที่ 4.29 สัญญาณTHR และLLD ที่ความสูงสัญญาณพัลส์ประมาณ 250 mV

4.1.3.4 การทดสอบในส่วนของอนาล็อกและดิจิตอล

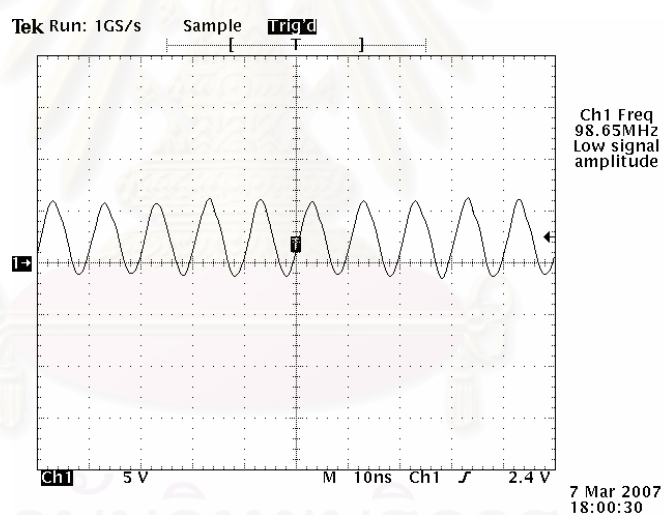
การทดสอบทำได้โดยการสร้างสัญญาณพัลส์จำลองจากเครื่องกำเนิดสัญญาณพัลส์ผ่านไปยังส่วนของกรขยายสัญญาณก็จะได้สัญญาณพัลส์ที่มีความสูงจาก 0 – 10 โวลต์ ให้กับวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลโดยประกอบไปด้วย 2 ส่วนหลักๆ คือ ส่วนอนาล็อกและส่วนดิจิตอลเพื่อแปลงเป็นสัญญาณอนาล็อกไปเป็นสัญญาณดิจิตอลขนาด 11 บิต ในรูปที่ 4.30 แสดงภาพการ ต่อวงจรทดสอบวงจรการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล



รูปที่ 4.30 ทดสอบวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลส่วนของอนาล็อกและดิจิตอล



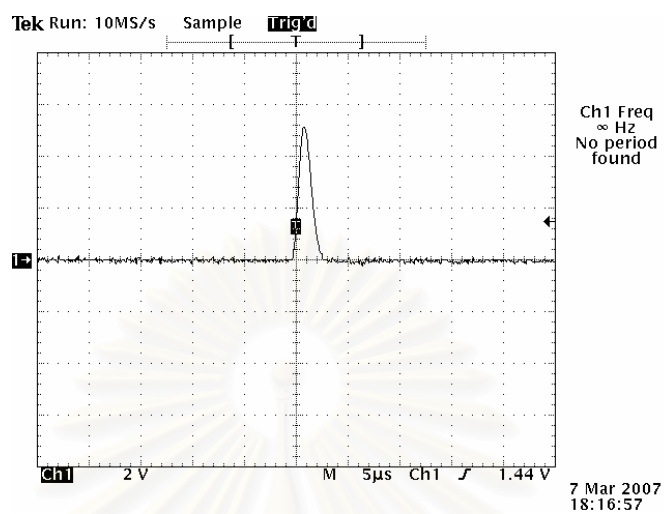
รูปที่ 4.31 ภาพการต่อวงจรความถี่ป้อนให้วงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล



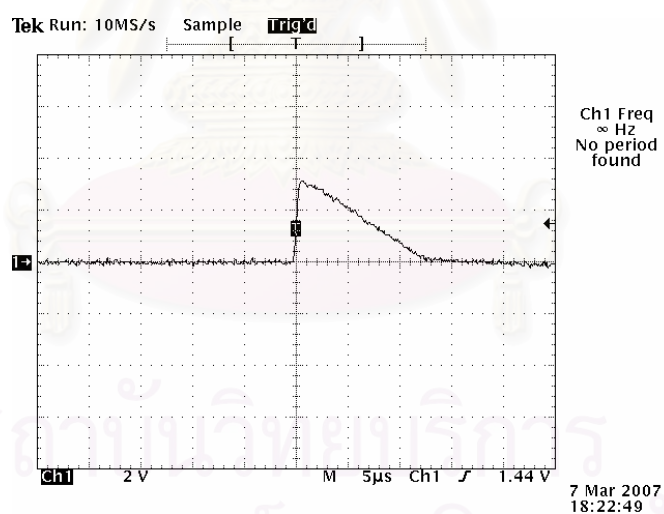
รูปที่ 4.32 สัญญาณความถี่ที่ป้อนให้วงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สัญญาณพัลส์มีความสูงเท่ากับ 5 โวลต์ จากโมดูลขยายสัญญาณพัลส์จะถูกยืดความสูงพัลส์ให้ได้เป็นค่าเวลาที่เป็นเชิงเส้นตามความสูงของพัลส์ ดังแสดงในรูปที่ 4.33 และ 4.34

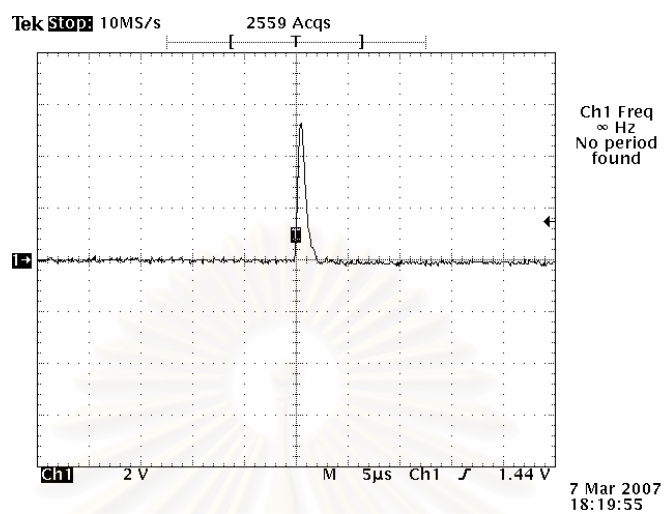


รูปที่ 4.33 สัญญาณพัลส์ที่ได้จากโมดูลขยายสัญญาณพัลส์

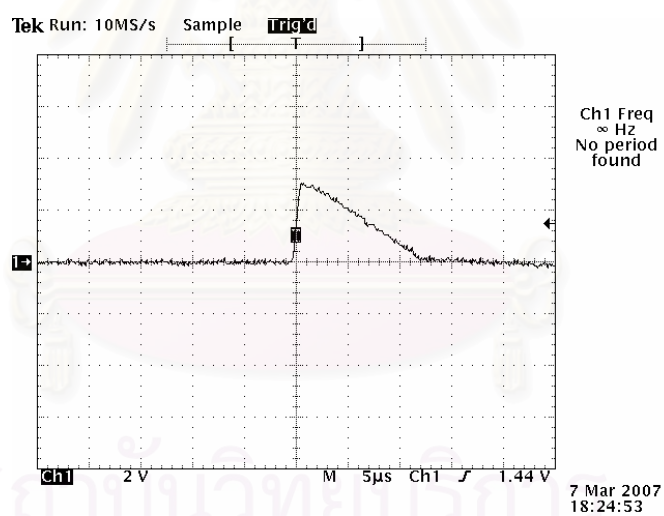


รูปที่ 4.34 การยืดความสูงพัลส์เป็นค่าเวลาที่เป็นเชิงเส้นตามความสูงของพัลส์

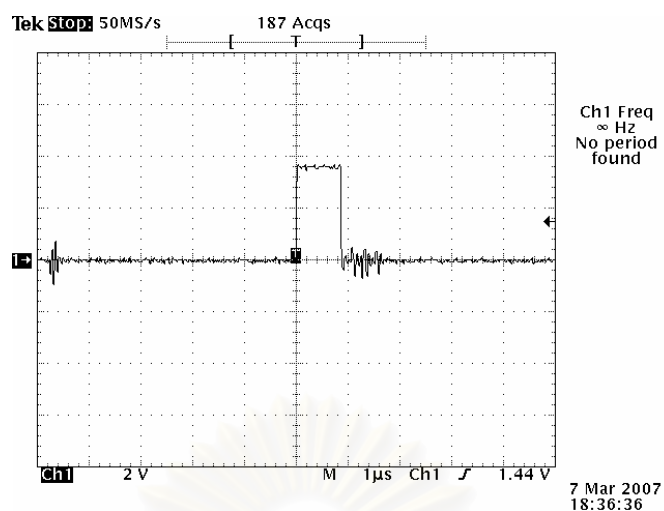
สัญญาณพัลส์มีความสูงเท่ากับ 5 โวลต์ จากวงจรขยายสัญญาณที่ออกแบบจะถูกยืด
ความสูง พัลส์ให้ได้เป็นค่าเวลาที่เป็นเชิงเส้นตามความสูงของพัลส์ ดังแสดงในรูปที่ 4.35 และ 4.36



รูปที่ 4.35 สัญญาณพัลส์ที่ได้จากวงจรขยายสัญญาณที่ออกแบบ



รูปที่ 4.36 การยืดความสูงพัลส์เป็นค่าเวลาที่เป็นเชิงเส้นตามความสูงของพัลส์



รูปที่ 4.37 ภาพการแปลงสัญญาณอนาลอกเป็นดิจิทัลที่มีความกว้างขนาด 0.9 μsec

ตารางที่ 4.3 การทดสอบความเป็นเชิงเส้นของวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล

Volts	Clocks	Times (μs)	Measure (μs)
0.5	102.4 CLK	1.024	1.3
1	204.8 CLK	2.048	2.5
2	409.6 CLK	4.096	4.5
3	614.4 CLK	6.144	6.5
4	819.2 CLK	8.192	8.4
5	1024 CLK	10.240	10.4
6	1228.8 CLK	12.288	12.2
7	1433.6 CLK	14.336	14.2
8	1638.4 CLK	16.384	16.5
9	1843.2 CLK	18.432	18.6
10	2048 CLK	20.480	20.6

4.1.4 การทดสอบระบบควบคุมการทำงานและการแสดงผล

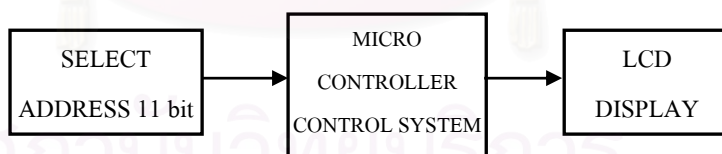
จากการออกแบบในส่วนของระบบควบคุมการทำงานและการแสดงผลเพื่อที่จะให้มีการจัดการแปลงค่าแอดเดรสจากสัญญาณอนาล็อกไปเป็นดิจิตอลลงในหน่วยความจำได้เร็วและนำค่าที่เก็บในหน่วยความจำมาจำแนกเป็นสเปกตรัมแสดงที่หน้าจอแสดงผลได้อย่างถูกต้องซึ่งสามารถทดสอบสมรรถนะในการทำงานของระบบควบคุมการทำงานและการแสดงผลได้ดังนี้

4.1.4.1 เครื่องมือและอุปกรณ์

1. แหล่งจ่ายแรงดันไฟฟ้าศักย์ค่าขนาด +5, ± 12 และ ± 24 โวลต์ ที่พัฒนาขึ้น
2. แผงวงจรไมโครคอนโทรลเลอร์ที่ออกแบบเป็นระบบควบคุมการทำงานและการแสดงผลที่ได้พัฒนาขึ้น
3. แผงวงจรการปรับเปลี่ยนค่าแอดเดรสขนาด 11 บิต
4. จอแสดงผลพร้อมแผงวงจรแหล่งจ่ายไฟฟ้าขนาด +5, +30 และ +1200 โวลต์

4.1.4.2 ขั้นตอนการทดสอบ

จัดอุปกรณ์สำหรับทดสอบสมรรถนะของระบบควบคุมการทำงานและการแสดงผลซึ่งทดสอบด้วยตัวกำหนดตำแหน่งขนาด 11 บิตที่สามารถเลือกได้ ทำการกำหนดค่าแอดเดรสแล้วนำไปเก็บยังหน่วยความจำภายในไมโครคอนโทรลเลอร์จากนั้นก็แสดงผลออกมาเป็นสเปกตรัมแสดงได้ดังรูปที่ 4.38



รูปที่ 4.38 แผนภาพการจัดอุปกรณ์ทดสอบระบบควบคุมการทำงานและการแสดงผล

4.1.4.3 ผลการทดสอบ

ทำการทดสอบเพื่อหาค่า Resolution แสดงออกมาที่จอภาพโดยผลการทดสอบความถูกต้องในการแสดงค่าในแต่ละช่องสัญญาณของระบบโดยป้อนสัญญาณพัลส์พลังงานเดียวให้กับระบบสามารถสร้างสเปกตรัมโดยความเที่ยงตรงในการแสดงผลออกมาที่หน้าจอจะขึ้นอยู่กับการเขียนโปรแกรมค่าที่ได้จากการทดสอบมีทั้งแบบข้อมูลและรูปภาพดังแสดงในรูปที่ 4.39



รูปที่ 4.39 แสดงการทดสอบระบบควบคุมการทำงานและการแสดงผล

4.2 การทดสอบทั้งระบบ

4.2.1 การทดสอบอุปกรณ์วิเคราะห์แบบหลายช่อง

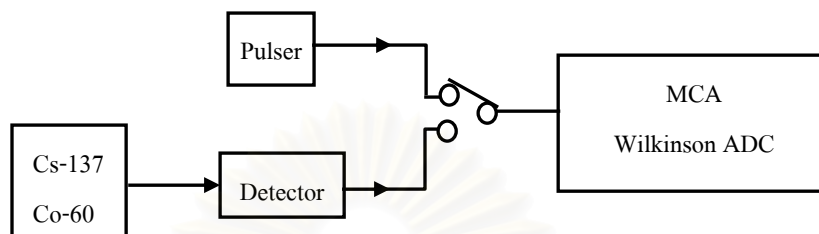
หลังจากออกแบบอุปกรณ์ส่วนย่อยสำหรับเครื่องวิเคราะห์ความสูงแบบหลายช่องขนาดเล็กตามข้อที่ 4.1.1 ถึงข้อที่ 4.1.4 แล้วสามารถสร้างเครื่องมือวิเคราะห์ความสูงแบบหลายช่องขนาดเล็กซึ่งประกอบไปด้วย วงจรแหล่งจ่ายไฟฟ้าสีกาสูง วงจรขยายสัญญาณพัลส์ วงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล และวงจรควบคุมระบบการทำงานและการแสดงผล ซึ่งสามารถทดสอบสมรรถนะในการทำงานของเครื่องวิเคราะห์ความสูงแบบหลายช่องขนาดเล็กโดยรวมได้ดังนี้

4.2.1.1 เครื่องมือและอุปกรณ์

1. เครื่องวิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิควัดคินสันเอดิซีที่พัฒนาขึ้น
2. แหล่งกำเนิดสัญญาณพัลส์ (Pulser) ของ CANBERRA รุ่น 807
3. หัววัดรังสีชนิดซิลิคิลเลชันรุ่น 2M2/2PR ยี่ห้อ BICRON ซึ่งรวมหลอดทิวคณิเล็กตรอน (PMT) ไว้ด้วย
4. สารกัมมันตรังสีโคบอลต์ 60 และ ซีเซียม 137

4.2.1.2 ขั้นตอนการทดสอบ

จัดอุปกรณ์สำหรับทดสอบสมรรถนะของเครื่องวิเคราะห์แบบหลายช่องขนาดเล็กที่ใช้เทคนิควิลคินสันเอ็นเอคี่ซี ซึ่งทดสอบด้วยแหล่งกำเนิดสัญญาณพัลส์ และการวัดรังสี Cs-137 และ Co-60 ดังรูปที่ 4.40



รูปที่ 4.40 แผนภาพการจัดอุปกรณ์ทดสอบระบบวิเคราะห์สเปกตรัมพลังงานรังสีแกมมา

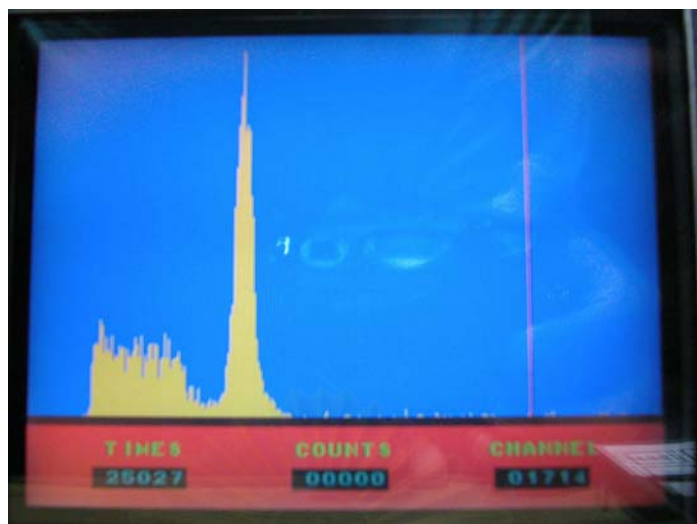
4.2.1.3 ผลการทดสอบ



Key Pad

Cs-137

รูปที่ 4.41 อุปกรณ์วิเคราะห์แบบหลายช่องและหัววัดรังสีแกมมาจาก Cs-137



รูปที่ 4.42 สเปกตรัมพลังงานรังสีแกมมาของ Cs-137

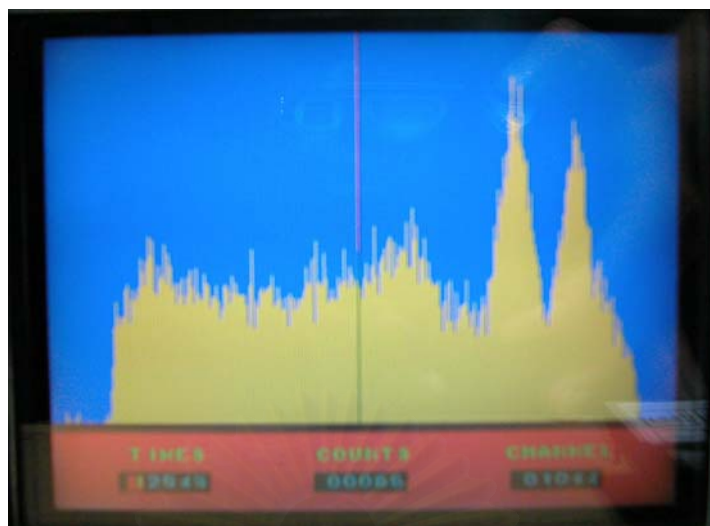


Key Pad

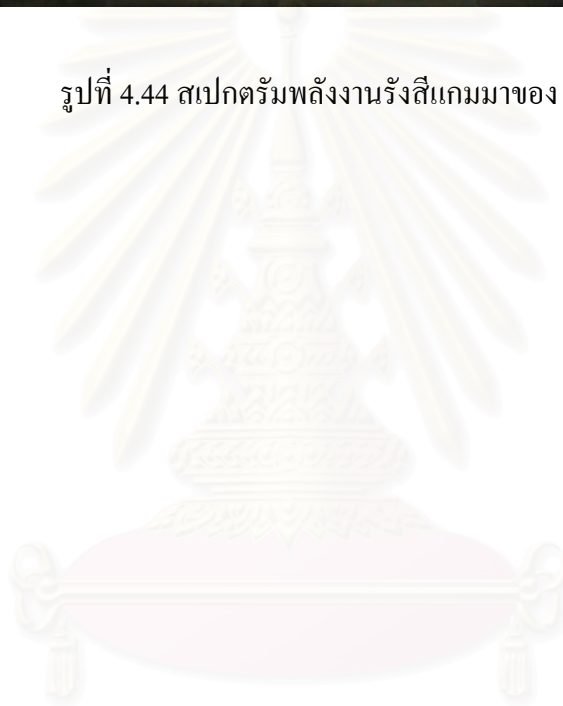


Co-60

รูปที่ 4.43 อุปกรณ์วิเคราะห์แบบหลายช่องและหัววัดรังสีแกมมาจาก Co-60



รูปที่ 4.44 สเปกตรัมพลังงานรังสีแกมมาของ Co-60



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากการพัฒนาอุปกรณ์วิเคราะห์แบบหลายช่องประเภท Stand Alone CPLD จะใช้คุณสมบัติร่วมกันระหว่างการใช้วงจรรีเลย์ทรอนิกส์สร้างส่วนของการแปลงสัญญาณอนาลอกเป็นดิจิทัลชนิดบิตซินต์ ส่วนในระบบควบคุมการทำงานจะใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงานและการแสดงผล

5.1.1 ผลทดสอบการทำงานของเครื่อง

ก. แหล่งจ่ายไฟฟ้าศักดาสูงปรับค่าได้ 0-2000 โวลต์ สามารถจ่ายกระแสได้สูงสุดถึง 1 mA ความเป็นเชิงเส้นของค่าที่อ่านได้บนสเกลมีค่าเท่า 0.99 มีขนาดของศักดาเรลอกคลื่นขณะจ่ายกระแสเต็มที่น้อยกว่า 30 mV_{pp} ซึ่งมีคุณภาพเพียงพอที่จะใช้กับหัววัดรังสีชนิดซินทิลเลชัน

ข. วงจรขยายสัญญาณพัลส์ที่พัฒนาขึ้นให้เหมาะสมหัววัดเรืองรังสีชนิดซินทิลเลชันที่มีความสูงพัลส์ 10-20 mV ค่าเวลาการสลายตัว (Decay time) ยาว 40-400 μ s จึงสร้างวงจรขยายสัญญาณพัลส์ที่มีอัตราขยาย 100 , 500 เท่าและ 1000 เท่า และความกว้างพัลส์ขนาด 2.5 μ s ขยายสัญญาณให้มีความสูงพัลส์ 0-10 V

ค. วงจรจ่ายไฟฟ้าศักดาต่ำวงจรแหล่งจ่ายไฟฟ้าศักดาต่ำที่พัฒนาขึ้นสามารถจ่ายแรงดันไฟฟ้าขนาด +5V /-5 V, +12V/-12V, +24V/-24V มีขนาดศักดาเรลอกคลื่นไม่เกิน 20 mV

ง. ระบบแปลงสัญญาณจากอนาลอกเป็นดิจิทัลที่พัฒนาขึ้นสามารถแปลงศักดาไฟฟ้าจาก 0-10 V เป็นสัญญาณดิจิทัลขนาด 11 บิตจากการทดสอบสมรรถนะความสามารถในการแปลงสัญญาณความเป็นเชิงเส้นทำได้เกือบ 100% ใช้งานที่ความถี่ 100 MHz

5.1.2 สมรรถนะและประโยชน์การใช้งานของเครื่อง

ก. สามารถใช้งานในด้านการเรียนการสอนและในการวิจัยขั้นพื้นฐานด้านเทคโนโลยีนิวเคลียร์และฟิสิกส์นิวเคลียร์ได้

ข. เครื่องที่พัฒนานี้สามารถพัฒนาขึ้นได้ด้วยตนเองโดยชิ้นส่วนอุปกรณ์สามารถหาได้จากภายในประเทศเกือบทั้งหมดทำให้ต้นทุนการผลิตมีราคาไม่เกิน 25,000 บาท เมื่อเปรียบเทียบกับเครื่องที่นำมาจากต่างประเทศจะมีราคาถูกกว่าหลายเท่า

5.2 คุณสมบัติของเครื่องมือ

5.2.1 คุณสมบัติของระบบวิเคราะห์ทางด้านฮาร์ดแวร์

1. แหล่งจ่ายไฟฟ้าสัปดาห์สูง
 - สามารถปรับค่าได้จาก 0-2000 โวลต์จ่ายกระแสได้สูงสุด 1 mA อ่านค่าศักดาไฟฟ้าจากสเกลปรับเทียบ
2. อุปกรณ์ขยายสัญญาณพัลส์
 - เป็นภาคขยายหลักปรับอัตราขยายได้ 100, 500 และ 1000 เท่า สัญญาณทางออกเป็นแบบยูนิโพลาร์มีขนาดความกว้างพัลส์ 2.5 μ s สามารถปรับความคลาดเคลื่อนของเส้นฐานสัญญาณด้วย P/Z และ BLR
3. ระบบแปลงสัญญาณพัลส์อนาล็อกเป็นสัญญาณดิจิทัล
 - ชนิดวิลคินสันขนาด 11 บิตใช้ความถี่ฐานเวลา 100 MHz และมีช่องของการวิเคราะห์ 2048 ช่อง
4. ระบบควบคุมการทำงาน
 - เป็นแบบ Interrupt Service Routine Controlled I/O รับข้อมูล 11 บิต จาก ADC อ่านข้อมูลเวลาและแสดงสเปกตรัมที่จอ LCD
5. ระบบแสดงผลสเปกตรัม
 - เป็นจอแสดงผล LCD ความละเอียดที่ 512 สีควบคุมด้วยไมโครคอนโทรลเลอร์ขนาด 16 บิต

5.2.2 คุณสมบัติของระบบวิเคราะห์ทางด้านซอฟต์แวร์

โปรแกรมควบคุม

สำหรับควบคุมการทำงานของฮาร์ดแวร์อ่านข้อมูลได้จากระบบแปลงสัญญาณการแสดงผลของสเปกตรัมและนับค่าปริมาณ ตลอดจนการจัดการเก็บข้อมูลวิเคราะห์ ได้แก่

- การอ่านปริมาณนับรังสี
- การอ่านผลรวมปริมาณรังสี

5.3 ปัญหาและอุปสรรค

5.3.1 การออกแบบที่ต้องใช้อุปกรณ์อิเล็กทรอนิกส์หลายชนิดจะมีปัญหาที่การจัดการอุปกรณ์ภายในประเทศไม่สามารถหาได้หรืออยู่ในภาวะของขาดตลาด หรือบางอุปกรณ์จำเป็นต้องสั่งซื้อจากต่างประเทศเท่านั้น เช่น สวิตช์เลือกแบบหลายชั้น (Selector) ออปแอมป์บางเบอร์ (Opamp) ซึ่งการสั่งซื้อจากต่างประเทศจะต้องใช้ระยะเวลาไม่ต่ำกว่า 1 เดือนหรือมากกว่านั้น และราคาค่อนข้างสูง

5.3.2 การสั่งทำแผ่นพิมพ์วงจร (Printed Circuit Board) คุณภาพสูงแบบเพลตทรูโฮล (Plate Through Hole) แม้แผ่นจะมีขนาดเล็กก็ยังมีราคาค่อนข้างสูง ในกรณีที่ต้องการทำโดยไม่มีเพลตทรูโฮล ก็สามารถเลือกทำได้ในราคาที่ค่อนข้างถูกแต่จะมีปัญหาเรื่องคุณภาพเพราะเส้นแตรีกทองแดงจะหลุดลอกได้ง่าย

5.3.3 อุปกรณ์ที่ออกแบบเป็น Surface Mount Size 0603 ค่าความต้านทานหรือค่าคาปาซิเตอร์บางค่าไม่สามารถหาได้ต้องใช้ที่ใกล้เคียงที่สุด และเนื่องจากขนาดอุปกรณ์ค่อนข้างเล็กมากจึงค่อนข้างลำบากเวลาลงอุปกรณ์บนแผ่นพิมพ์วงจร

5.3.4 การออกแบบด้านการควบคุมระบบและการแสดงผลในช่วงเริ่มต้นของการออกแบบจะต้องทดลองหาไมโครคอนโทรลเลอร์หลายๆ แบบเพื่อให้ได้ค่าความถี่ฐานเวลาในการทำงานสูงสุด แต่ที่ใช้งานออกแบบไปแล้วความถี่ฐานเวลาก็ยังไม่สูงตามความต้องการ

5.3.5 การพัฒนาด้านการเขียนโปรแกรมของอุปกรณ์วิเคราะห์แบบหลายช่อง ติดปัญหาการเขียนโปรแกรมด้วยความจำกัดเนื่องจากโปรแกรมที่ใช้จะเป็นโปรแกรมทดลองทั้งหมด ทำให้เขียนโปรแกรมไม่ได้ตามต้องการที่ออกแบบ

5.4 ข้อเสนอแนะ

ระบบควบคุมการเลือกฟังก์ชันหรือย่านการวิเคราะห์ความสูงของพัลส์ ซึ่งปรับด้วยสวิตช์ทางกลจะมีปัญหาเรื่องของหน้าสัมผัส ถ้าใช้เทคนิคการปรับแบบเขียนโปรแกรมควบคุมสวิตช์ทางอิเล็กทรอนิกส์จะช่วยแก้ปัญหานี้ได้เป็นอย่างดี

ในการพัฒนาโปรแกรมควรจะใช้โปรแกรมที่ใช้งานจริงถูกต้องตามลิขสิทธิ์เพื่อการพัฒนาโปรแกรมสามารถทำได้เต็มที่

รายการอ้างอิง

- [1] ชีรวัดน์ ประกอบผล. การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี. บริษัท ดวงกลมสมัย จำกัด ถนนรัชดาภิเษก แขวงจันทระเกษม เขตจตุจักร กรุงเทพฯ, 2545
- [2] สุวิทย์ ปุณณชัยยะ. Nuclear Radiation Detection and Instrumentation. เอกสารการสอนวิชา 2111606 นิเวศวิทยเทคโนโลยี คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2538
- [3] เอกชัย มะการ. เรียนรู้และเข้าใจไมโครคอนโทรลเลอร์ MCS-51 ด้วยภาษาเบสิก BASCOM-8051. บริษัท อีทีที จำกัด ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ, 2547.
- [4] โอภาส ศิริครรชิตถาวร. เรียนรู้และพัฒนาไมโครคอนโทรลเลอร์ ARM7 LPC2148 ด้วยภาษาซี. โรงพิมพ์วัชรินสาสน์ รัชดา รัชดาภิเษก ลาดยาว จตุจักร กรุงเทพฯ, 2537
- [5] APEX Team. สร้างไอซีดิจิทัลสมัยใหม่ด้วย FPGA และ CPLD โดยใช้ ISE WebPACK 8.1i. Ailogictechnology Division, Apex Instrument Co.,Ltd, 2004
- [6] Knoll, Glenn F. Radiation Detection and Measurement. New York: McGraw-Hill Book, 1983
- [7] Gillmore G. And Hemingway J. Practical Gamma-Ray Spectrometry. New York: John Wiley & Son, 1995
- [8] Mary Brown. Power Supply Cookbook. Elsevier Wheeler Road Burlington, MA, 2001
- [9] Tracor Team. TN-1705 and TN-1706 Pulse Height Analyzers. Tracor Northern, INC., 1987



ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



XC95108 In-System Programmable CPLD

DS066 (v4.1) August 21, 2003

Product Specification

Features

- 7.5 ns pin-to-pin logic delays on all pins
- f_{CNT} to 125 MHz
- 108 macrocells with 2,400 usable gates
- Up to 108 user I/O pins
- 5V in-system programmable
 - Endurance of 10,000 program/erase cycles
 - Program/erase over full commercial voltage and temperature range
- Enhanced pin-locking architecture
- Flexible 36V18 Function Block
 - 90 product terms drive any or all of 18 macrocells within Function Block
 - Global and product term clocks, output enables, set and reset signals
- Extensive IEEE Std 1149.1 boundary-scan (JTAG) support
- Programmable power reduction mode in each macrocell
- Slew rate control on individual outputs
- User programmable ground pin capability
- Extended pattern security features for design protection
- High-drive 24 mA outputs
- 3.3V or 5V I/O capability
- Advanced CMOS 5V Fast FLASH™ technology
- Supports parallel programming of more than one XC9500 concurrently
- Available in 84-pin PLCC, 100-pin PQFP, 100-pin TQFP, and 160-pin PQFP packages

Description

The XC95108 is a high-performance CPLD providing advanced in-system programming and test capabilities for general purpose logic integration. It is comprised of eight 36V18 Function Blocks, providing 2,400 usable gates with propagation delays of 7.5 ns. See Figure 2 for the architecture overview.

Power Management

Power dissipation can be reduced in the XC95108 by configuring macrocells to standard or low-power modes of operation. Unused macrocells are turned off to minimize power dissipation.

Operating current for each design can be approximated for specific operating conditions using the following equation:

$$I_{CC} \text{ (mA)} = MC_{HP} (1.7) + MC_{LP} (0.9) + MC (0.006 \text{ mA/MHz}) f$$

Where:

MC_{HP} = Macrocells in high-performance mode

MC_{LP} = Macrocells in low-power mode

MC = Total number of macrocells used

f = Clock frequency (MHz)

Figure 1 shows a typical calculation for the XC95108 device.

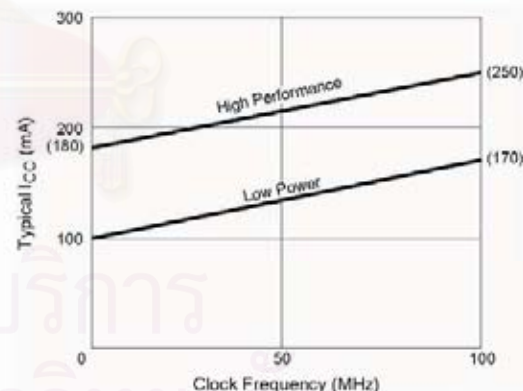
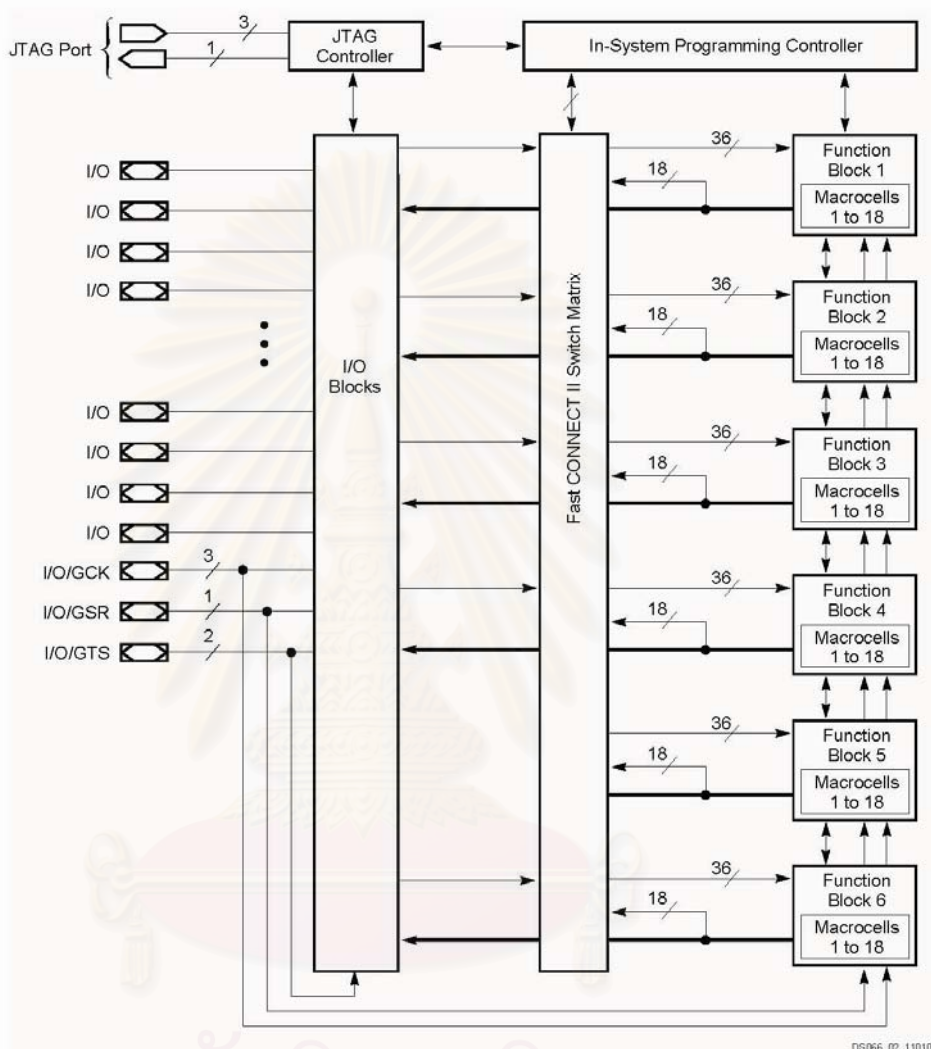


Figure 1: Typical I_{CC} vs. Frequency for XC95108

© 2003 Xilinx, Inc. All rights reserved. All Xilinx trademarks, registered trademarks, patents, and disclaimers are as listed at <http://www.xilinx.com/legal.htm>. All other trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.



DS066_02_110101

Figure 2: XC95108 Architecture
Function block outputs (indicated by the bold line) drive the I/O blocks directly.

สถาบันวิจัยและพัฒนา
จุฬาลงกรณ์มหาวิทยาลัย

Absolute Maximum Ratings

Symbol	Description	Value	Units
V_{CC}	Supply voltage relative to GND	-0.5 to 7.0	V
V_{IN}	Input voltage relative to GND	-0.5 to $V_{CC} + 0.5$	V
V_{TS}	Voltage applied to 3-state output	-0.5 to $V_{CC} + 0.5$	V
T_{STG}	Storage temperature (ambient)	-65 to +150	°C
T_J	Junction temperature	+150	°C

Notes:

- Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those listed under Operating Conditions is not implied. Exposure to Absolute Maximum Ratings conditions for extended periods of time may affect device reliability.

Recommended Operation Conditions

Symbol	Parameter		Min	Max	Units
V_{CCINT}	Supply voltage for internal logic and input buffers	Commercial $T_A = 0^\circ\text{C}$ to 70°C	4.75	5.25	V
		Industrial $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$	4.5	5.5	
V_{CCIO}	Supply voltage for output drivers for 5V operation	Commercial $T_A = 0^\circ\text{C}$ to 70°C	4.75	5.25	V
		Industrial $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$	4.5	5.5	
	Supply voltage for output drivers for 3.3V operation	3.0	3.6		
V_{IL}	Low-level input voltage		0	0.80	V
V_{IH}	High-level input voltage		2.0	$V_{CCINT} + 0.5$	V
V_O	Output voltage		0	V_{CCIO}	V

Quality and Reliability Characteristics

Symbol	Parameter	Min	Max	Units
T_{DR}	Data Retention	20	-	Years
N_{PE}	Program/Erase Cycles (Endurance)	10,000	-	Cycles

DC Characteristic Over Recommended Operating Conditions

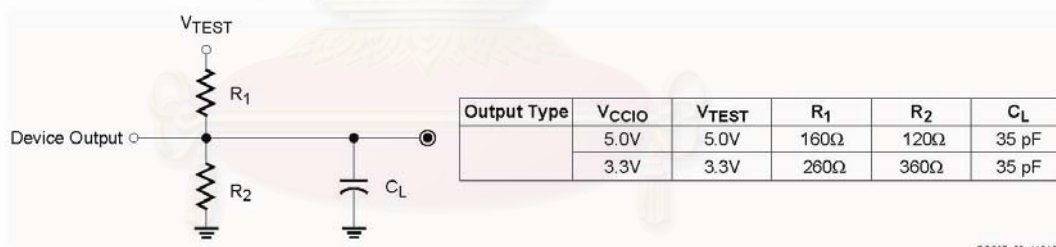
Symbol	Parameter	Test Conditions	Min	Max	Units
V_{OH}	Output high voltage for 5V outputs	$I_{OH} = -4.0\text{ mA}$, $V_{CC} = \text{Min}$	2.4	-	V
	Output high voltage for 3.3V outputs	$I_{OH} = -3.2\text{ mA}$, $V_{CC} = \text{Min}$	2.4	-	V
V_{OL}	Output low voltage for 5V outputs	$I_{OL} = 24\text{ mA}$, $V_{CC} = \text{Min}$	-	0.5	V
	Output low voltage for 3.3V outputs	$I_{OL} = 10\text{ mA}$, $V_{CC} = \text{Min}$	-	0.4	V
I_{IL}	Input leakage current	$V_{CC} = \text{Max}$ $V_{IN} = \text{GND or } V_{CC}$	-	± 10	μA
I_{IH}	I/O high-Z leakage current	$V_{CC} = \text{Max}$ $V_{IN} = \text{GND or } V_{CC}$	-	± 10	μA
C_{IN}	I/O capacitance	$V_{IN} = \text{GND}$ $f = 1.0\text{ MHz}$	-	10	pF
I_{CC}	Operating supply current (low power mode, active)	$V_I = \text{GND}$, No load $f = 1.0\text{ MHz}$	100 (Typical)		mA

AC Characteristics

Symbol	Parameter	XC95108-7		XC95108-10		XC95108-15		XC95108-20		Units
		Min	Max	Min	Max	Min	Max	Min	Max	
T_{PD}	I/O to output valid	-	7.5	-	10.0	-	15.0	-	20.0	ns
T_{SU}	I/O setup time before GCK	4.5	-	6.0	-	8.0	-	10.0	-	ns
T_H	I/O hold time after GCK	0	-	0	-	0	-	0	-	ns
T_{CO}	GCK to output valid	-	4.5	-	6.0	-	8.0	-	10.0	ns
$f_{CNT}^{(1)}$	16-bit counter frequency	125.0	-	111.1	-	95.2	-	83.3	-	MHz
$f_{SYSTEM}^{(2)}$	Multiple FB internal operating frequency	83.3	-	66.7	-	55.6	-	50.0	-	MHz
T_{PSU}	I/O setup time before p-term clock input	0.5	-	2.0	-	4.0	-	4.0	-	ns
T_{PH}	I/O hold time after p-term clock input	4.0	-	4.0	-	4.0	-	6.0	-	ns
T_{PCO}	P-term clock output valid	-	8.5	-	10.0	-	12.0	-	16.0	ns
T_{OE}	GTS to output valid	-	5.5	-	6.0	-	11.0	-	16.0	ns
T_{OD}	GTS to output disable	-	5.5	-	6.0	-	11.0	-	16.0	ns
T_{POE}	Product term OE to output enabled	-	9.5	-	10.0	-	14.0	-	18.0	ns
T_{POD}	Product term OE to output disabled	-	9.5	-	10.0	-	14.0	-	18.0	ns
T_{WLH}	GCK pulse width (High or Low)	4.0	-	4.5	-	5.5	-	5.5	-	ns

Notes:

- f_{CNT} is the fastest 16-bit counter frequency available, using the local feedback when applicable.
 f_{CNT} is also the Export Control Maximum flip-flop toggle rate, f_{TOG} .
- f_{SYSTEM} is the internal operating frequency for general purpose system designs spanning multiple FBs.



DS067_03_110101

Figure 3: AC Load Circuit



Internal Timing Parameters

Symbol	Parameter	XC95108-7		XC95108-10		XC95108-15		XC95108-20		Units
		Min	Max	Min	Max	Min	Max	Min	Max	
Buffer Delays										
T _{IN}	Input buffer delay	-	2.5	-	3.5	-	4.5	-	6.5	ns
T _{GCK}	GCK buffer delay	-	1.5	-	2.5	-	3.0	-	3.0	ns
T _{GSR}	GSR buffer delay	-	4.5	-	6.0	-	7.5	-	9.5	ns
T _{GTS}	GTS buffer delay	-	5.5	-	6.0	-	11.0	-	16.0	ns
T _{OUT}	Output buffer delay	-	2.5	-	3.0	-	4.5	-	6.5	ns
T _{EN}	Output buffer enable/disable delay	-	0	-	0	-	0	-	0	ns
Product Term Control Delays										
T _{PTCK}	Product term clock delay	-	3.0	-	3.0	-	2.5	-	2.5	ns
T _{PTSR}	Product term set/reset delay	-	2.0	-	2.5	-	3.0	-	3.0	ns
T _{PTTS}	Product term 3-state delay	-	4.5	-	3.5	-	5.0	-	5.0	ns
Internal Register and Combinatorial Delays										
T _{PDI}	Combinatorial logic propagation delay	-	0.5	-	1.0	-	3.0	-	4.0	ns
T _{SUI}	Register setup time	1.5	-	2.5	-	3.5	-	3.5	-	ns
T _{HI}	Register hold time	3.0	-	3.5	-	4.5	-	6.5	-	ns
T _{COI}	Register clock to output valid time	-	0.5	-	0.5	-	0.5	-	0.5	ns
T _{AOI}	Register async. S/R to output delay	-	6.5	-	7.0	-	8.0	-	8.0	ns
T _{RAI}	Register async. S/R recover before clock	7.5	-	10.0	-	10.0	-	10.0	-	ns
T _{LOGI}	Internal logic delay	-	2.0	-	2.5	-	3.0	-	3.0	ns
T _{LOGILP}	Internal low power logic delay	-	10.0	-	11.0	-	11.5	-	11.5	ns
Feedback Delays										
T _F	Fast CONNECT II feedback delay	-	8.0	-	9.5	-	11.0	-	13.0	ns
T _{LF}	Function block local feedback delay	-	4.0	-	3.5	-	3.5	-	5.0	ns
Time Adders										
T _{PTA} ⁽¹⁾	Incremental product term allocator delay	-	1.0	-	1.0	-	1.0	-	1.5	ns
T _{SLEW}	Slew-rate limited delay	-	4.0	-	4.5	-	5.0	-	5.5	ns

Notes:

1. T_{PTA} is multiplied by the span of the function as defined in the XC9500 family data sheet.

XC95108 I/O Pins

Function Block	Macro-cell	PC84	PQ100	TQ100	PQ160	BScan Order	Function Block	Macro-cell	PC84	PQ100	TQ100	PQ160	BScan Order
1	1	–	–	–	25	321	3	1	–	–	–	45	213
1	2	1	15	13	21	318	3	2	14	31	29	47	210
1	3	2	16	14	22	315	3	3	15	32	30	49	207
1	4	–	21	19	29	312	3	4	–	36	34	57	204
1	5	3	17	15	23	309	3	5	17	34	32	54	201
1	6	4	18	16	24	306	3	6	18	35	33	56	198
1	7	–	–	–	27	303	3	7	–	–	–	50	195
1	8	5	19	17	26	300	3	8	19	37	35	58	192
1	9	6	20	18	28	297	3	9	20	38	36	59	189
1	10	–	26	24	36	294	3	10	–	45	43	69	186
1	11	7	22	20	30	291	3	11	21	39	37	60	183
1	12	9 ^[1]	24 ^[1]	22 ^[1]	33 ^[1]	288 ^[1]	3	12	23	41	39	62	180
1	13	–	–	–	34	285	3	13	–	–	–	52	177
1	14	10 ^[1]	25 ^[1]	23 ^[1]	35 ^[1]	282 ^[1]	3	14	24	42	40	63	174
1	15	11	27	25	37	279	3	15	25	43	41	64	171
1	16	12 ^[1]	29 ^[1]	27 ^[1]	42 ^[1]	276 ^[1]	3	16	26	44	42	68	168
1	17	13	30	28	44	273	3	17	31	51	49	77	165
1	18	–	–	–	43	270	3	18	–	–	–	74	162
2	1	–	–	–	158	267	4	1	–	–	–	123	159
2	2	71	98	96	154	264	4	2	57	83	81	134	156
2	3	72	99	97	156	261	4	3	58	84	82	135	153
2	4	–	4	2	4	258	4	4	–	82	80	133	150
2	5	74 ^[1]	1 ^[1]	99 ^[1]	159 ^[1]	255 ^[1]	4	5	61	87	85	138	147
2	6	75	3	1	2	252	4	6	62	88	86	139	144
2	7	–	–	–	9	249	4	7	–	–	–	128	141
2	8	76 ^[1]	5 ^[1]	3 ^[1]	6 ^[1]	246 ^[1]	4	8	63	89	87	140	138
2	9	77 ^[1]	6 ^[1]	4 ^[1]	8 ^[1]	243 ^[1]	4	9	65	91	89	142	135
2	10	–	9	7	12	240	4	10	–	–	–	147	132
2	11	79	8	6	11	237	4	11	66	92	90	143	129
2	12	80	10	8	13	234	4	12	67	93	91	144	126
2	13	–	–	–	14	231	4	13	–	–	–	153	123
2	14	81	11	9	15	228	4	14	68	95	93	146	120
2	15	82	12	10	17	225	4	15	69	96	94	148	117
2	16	83	13	11	18	222	4	16	–	94	92	145	114
2	17	84	14	12	19	219	4	17	70	97	95	152	111
2	18	–	–	–	16	216	4	18	–	–	–	155	108

Notes:

1. Global control pin.



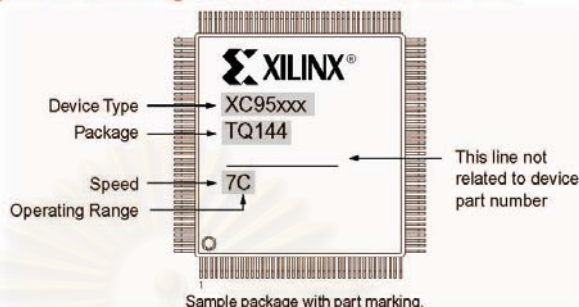
XC95108 I/O Pins (Continued)

Function Block	Macro-cell	PC84	PQ100	TQ100	PQ160	BScan Order	Function Block	Macro-cell	PC84	PQ100	TQ100	PQ160	BScan Order
5	1	–	–	–	76	105	6	1	–	–	–	91	51
5	2	32	52	50	79	102	6	2	45	67	65	103	48
5	3	33	54	52	82	99	6	3	46	68	66	104	45
5	4	–	48	46	72	96	6	4	–	75	73	116	42
5	5	34	55	53	86	93	6	5	47	69	67	106	39
5	6	35	56	54	88	90	6	6	48	70	68	108	36
5	7	–	–	–	78	87	6	7	–	–	–	105	33
5	8	36	57	55	90	84	6	8	50	72	70	111	30
5	9	37	58	56	92	81	6	9	51	73	71	113	27
5	10	–	–	–	84	78	6	10	–	–	–	107	24
5	11	39	60	58	95	75	6	11	52	74	72	115	21
5	12	40	62	60	97	72	6	12	53	76	74	117	18
5	13	–	–	–	87	69	6	13	–	–	–	112	15
5	14	41	63	61	98	66	6	14	54	78	76	122	12
5	15	43	65	63	101	63	6	15	55	79	77	124	9
5	16	–	61	59	96	60	6	16	–	81	79	129	6
5	17	44	66	64	102	57	6	17	56	80	78	126	3
5	18	–	–	–	89	54	6	18	–	–	–	114	0

XC95108 Global, JTAG, and Power Pins

Pin Type	PC84	PQ100	TQ100	PQ160
I/O/GCK1	9	24	22	33
I/O/GCK2	10	25	23	35
I/O/GCK3	12	29	27	42
I/O/GTS1	76	5	3	6
I/O/GTS2	77	6	4	8
I/O/GSR	74	1	99	159
TCK	30	50	48	75
TDI	28	47	45	71
TDO	59	85	83	136
TMS	29	49	47	73
V _{CCINT} 5V	38,73,78	7,59,100	5,57,98	10,46,94,157
V _{CCIO} 3.3V/5V	22,64	28,40,53,90	26,38,51,88	1,41,61,81,121,141
GND	8,16,27,42,49,60	2,23,33,46,64,71,77,86	100,21,31,44,62,69,75,84	20,31,40,51,70,80,99
GND	–	–	–	100,110,120,127,137
GND	–	–	–	160
No connects	–	–	–	3,5,7,32,38,39,48,53,55,6 5,66,67,83,85,93,109, 118,119,125,130,131, 132,149,150,151

Device Part Marking and Ordering Combination Information



Device Ordering and Part Marking Number	Speed (pin-to-pin delay)	Pkg. Symbol	No. of Pins	Package Type	Operating Range ⁽¹⁾
XC95108-7PC84C	7.5 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	C
XC95108-7PQ100C	7.5 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-7TQ100C	7.5 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	C
XC95108-7PQ160C	7.5 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-7PC84I	7.5 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	I
XC95108-7PQ100I	7.5 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-7TQ100I	7.5 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	I
XC95108-7PQ160I	7.5 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-10PC84C	10 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	C
XC95108-10PQ100C	10 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-10TQ100C	10 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	C
XC95108-10PQ160C	10 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-10PC84I	10 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	I
XC95108-10PQ100I	10 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-10TQ100I	10 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	I
XC95108-10PQ160I	10 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-15PC84C	15 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	C
XC95108-15PQ100C	15 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-15TQ100C	15 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	C
XC95108-15PQ160C	15 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-15PC84I	15 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	I
XC95108-15PQ100I	15 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-15TQ100I	15 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	I
XC95108-15PQ160I	15 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-20PC84C	20 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	C
XC95108-20PQ100C	20 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-20TQ100C	20 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	C
XC95108-20PQ160C	20 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	C
XC95108-20PC84I	20 ns	PC84	84-pin	Plastic Lead Chip Carrier (PLCC)	I
XC95108-20PQ100I	20 ns	PQ100	100-pin	Plastic Quad Flat Pack (PQFP)	I
XC95108-20TQ100I	20 ns	TQ100	100-pin	Thin Quad Flat Pack (TQFP)	I
XC95108-20PQ160I	20 ns	PQ160	100-pin	Plastic Quad Flat Pack (PQFP)	I

Notes:

1. C = Commercial: $T_A = 0^\circ$ to $+70^\circ\text{C}$; I = Industrial: $T_A = -40^\circ$ to $+85^\circ\text{C}$

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
12/04/98	3.0	Update AC characteristics and internal parameters.
06/18/03	4.0	Updated format.
08/21/03	4.1	Updated Package Device Marking Pin 1 orientation.



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข.

โปรแกรมการทำงานของไมโครคอนโทรลเลอร์ ARM7

```

/*****/
#include <LPC213x.H> // LPC2138 MPU Register
#define LCD_VS 0x8000000 // P1.31
#define LCD_HS 0x0002000 // P1.17
#define LCD_DK 0x0004000 // P1.18
#define LCD_RED 0X0038000 // P1.19 - P1.21
#define LCD_GREEN 0X01C0000 // P1.22 - P1.24
#define LCD_BLUE 0X0E00000 // P1.25 - P1.27
#define LCD_YELLOW 0X00E00000
#define LCD_BLACK 0X00000000
#define LCD_ALL 0X0FF80000
void VSYNC();
void HSYNC();
void HSYNC1(); // Delay Time Function
void HSYNC2();
void HSYNC3(); // Delay Time Function
void HSYNC4();
void DCK();
void Key_switch();
void delay_ms(long int ms);
int cu,CURSOR;
unsigned int LOOPVS,LOOPHS;
unsigned int KEY_SCAN,KEY,LOOP,n,i,SECOND;
unsigned int TIME1,TIME2,TIME3,TIME4,TIME5;
unsigned int NUM1,NUM2,NUM3,NUM4,NUM5;
unsigned int CHAN1,CHAN2,CHAN3,CHAN4,CHAN5;
unsigned long int A,B,SPECTRUM,CHANNEL,ADDRESS;
/*****/
/* Array TYPE */
/*****/
unsigned int CON[16]; //CON[0]-[15]
unsigned int CH[2100]; //CH[0]-CH[2099]
/*****/
/* Array TYPE NUM */
/*****/
unsigned int Table_NUMBER[80][8]= //0 TO 9//[80][8]= [row][column]
{

```



```

0x00380000, 0x00380000, 0x01C00000, 0x01C00000, 0x00380000, 0x00380000, 0x01C00000, 0x01C00000,
0x00380000, 0x00380000, 0x01C00000, 0x01C00000, 0x00380000, 0x00380000, 0x01C00000, 0x01C00000,
0x00380000, 0x00380000, 0x01C00000, 0x01C00000, 0x00380000, 0x00380000, 0x01C00000, 0x01C00000,
0x00380000, 0x00380000, 0x01C00000, 0x01C00000, 0x00380000, 0x00380000, 0x01C00000, 0x01C00000,
0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000,
0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000, 0x00380000,
};
/*****
/* Array Function */
/*****
unsigned int Table[4]=                               //[4 row]
{0x00000004, 0x00000001, 0x00000040, 0x00000010 };
/*****
/* INTERUP RCT */
/*****
void RTC(void) __irq
{
    if(ILR&0X01)
    {
        VSYNC0;
        //SECOND++;
        ILR = 0X01;
    }
    VICVectAddr = 0x00000000;
}
/*****
/* INTERUP 3 */
/*****
void EINT3(void) __irq
{
    CHANNEL = IOPIN0 & 0X0007FF00; //READ FROM P0.8 - P0.18
    if((IOPIN0 & 0X00080000)== 0) //TEST GTZ IS BIT 1 OR 0
    {
        ADDRESS = 0X00000000; //ADDRESS AT 0 START COMPARE

        for(LOOP=1;LOOP<=2047;LOOP++)
        {
            if(CHANNEL == ADDRESS)
            {
                CH[LOOP]++;
            }
        }
    }
    else

```

```

        {

        }

        ADDRESS += 0X00000100;
    }

    IOCLR1 = LCD_ALL;
    VSYNC0;
}
else
{
    VSYNC0;
}
SECOND++;
EXTINT |= 0x08; //CLEAR BIT INTERUP EINT3//
VICVectAddr = 0; //END OF INTERUP
}
/*****
/* MAIN */
*****/
void main(void)
{
    PINSEL2 = 0X00000000; // SET P1.16 - P1.31 = GPIO
    VSYNC,HSYNC,DCK,LCD_RED,GREEN,BLUE;
    PINSEL1 &= 0XFFFFFF00; // SET P0.16 - P0.21 = GPIO ADDRESS_MCA(BIT
9,10,11,GTZ,EINT3,REC);
    PINSEL0 = 0X00000000; // SET P0.0 - P0.15 = GPIO KEY SWITCH,ADDRESS_MCA(BIT
1,2,3,4,5,6,7,8);

    IODIR1 = 0xFFFF0000; // SET P1.16 - P1.31 = Output
    VSYNC,HSYNC,DCK,LCD_RED,GREEN,BLUE;
    IODIR0 = 0X00000055; // SET P0.0,P0.2,P0.4,P0.6 = Output (P0.1,P0.3,P0.5,P0.7 = input =
P0.8 -P0.31)

    IOSET1 = 0X80060000; // SET P1.17,P1.18,P1.31 = 1
    IOCLR0 = 0X00000055; // SET P0.0,P0.2,P0.4,P0.6 = 0
    for(n=0;n<=2100;n++)
    {
        CH[n] = 0;
    }

    NUM1 =0,NUM1 =0,NUM3 =0,NUM4 =0,NUM5 =0;
    CHAN1=0,CHAN2=0,CHAN3=0,CHAN4=0,CHAN5=0;
    CURSOR=0;
    SECOND=0;

```

```

cu = 1050;
////////////////////////////////////////////////////////////////
CIIR = 0X0000001;           // ENABLE SECOND COUNTER INTERUPT
AMR = 0X000000FE;          // ALARM VALUE FOR SECONDS
CCR = 0X00000011;          // CLOCK CONTROL REGISTER(BIT0=1,BIT1=1)
VICVectAddr2 = (unsigned)RTC; //CALL ADDRESS SUB FUNCTION   RTC
VICVectCntl2 = 0x20 | 13;   //CONTROL SLOT 0 RECRIVE RTC
// VICIntEnable |= 1<<13;   //SET BIT 13 AS 1 OPEN WORK RTC
////////////////////////////////////////////////////////////////
EXTMODE |= 0X08;           //SET EINT3: 0 WORK AT LEVEL : SET 1 WORK AT TRIG ( &= AND, |= OR )
EXTPOLAR |= 0X08;         //SET EINT3 WORK AT POSITIVE EAGE
PINSEL1 |= 0X00000300;     //SET P0.20 = EINT3
VICVectAddr0 = (unsigned)EINT3; //CALL ADDRESS SUB FUNCTION   EINT3
VICVectCntl0 = 0x20 | 13;   //CONTROL SLOT 1 RECRIVE EINT1 P0.3
VICIntEnable |= 1<<17;     //SET BIT 17 AS 1 OPEN WORK EINT1
////////////////////////////////////////////////////////////////
// VICIntEnable = 0X00022000;
////////////////////////////////////////////////////////////////
VSYNCO);
while(1)
{
for(n=0;n<4;n++)
{
IOSET0 = 0X000000FF;
IOCLR0 |= Table[n];

A = ~Table[n]& 0X00000055;
B = IOPIN0 & 0X000000AA;
if(B != 0x000000AA)
{
delay_ms(1);
B = IOPIN0 & 0X000000AA;
if(B != 0x000000AA)
{
KEY_SCAN = A | B;
KEY = KEY_SCAN;
Key_switch();
}
}
else

Key_switch();
KEY = 0;
}
}

```

```

else
    {
        Key_switch();
    }
}
}
}
/*****
/*  switch Function */
*****/
void Key_switch()
{
    switch(KEY)
    {
        case 0x000000DB:
            cu++; // key 1
            VSYNC();
            break;

        case 0x000000DE:
            cu--; // key 2
            VSYNC();
            break;

        case 0x0000009F:
            CON[2]++; // key 3
            VSYNC();
            break;

        case 0x000000CF:
            CON[3]++; // key 4
            VSYNC();
            break;

        case 0x0000007B:
            CON[4]++; // key 5
            VSYNC();
            break;

        case 0x0000007E:

```

```
CON[5]++; // key 6
VSYNC();
break;

case 0x0000003F:

CON[6]++; // key 7
VSYNC();
break;

case 0x0000006F:

CON[7]++; // key 8
VSYNC();
break;

case 0x000000F9:

CON[8]++; // key 9
VSYNC();
break;

case 0x000000FC:

CON[9]++; // key 10
VSYNC();
break;

case 0x000000BD:

CON[10]++; // key 11
VSYNC();
break;

case 0x000000ED:

CON[11]++; // key 12
VSYNC();
break;

case 0x000000F3:

CON[12]++; // key 13
VSYNC();
break;

case 0x000000F6:

CON[13]++; // key 14
VSYNC();
break;
```

```

case 0x000000B7:
    CON[14]++;
    VSYNC0;
    break;

case 0x000000E7:
    for(n=0;n<2101;n++)
    {
        CH[n] = 0;
    }
    SECOND= 0;
    VSYNC0;
//default: VSYNC0;
}
}

//.....
/*****
/* VSYNC Function */
/*****

void VSYNC()
{
//.....

    IOCLR1 = LCD_VS;
//.....

    for(LOOPVS=0;LOOPVS<3;LOOPVS++)
    {
        HSYNC10;
    }
//.....

    IOSET1 = LCD_VS;
//.....

    for(LOOPVS=0;LOOPVS<4;LOOPVS++)
    {
        HSYNC10;
    }
//.....
    i = 0;
}

```

```

for(LOOPVS=0;LOOPVS<195;LOOPVS++)
{
    /* insert data line */ // LOOP 195 ROUND
    i = (195 - LOOPVS);
    /* insert data line */
    HSYNC0;
}
//.....
for(LOOPVS=0;LOOPVS<5;LOOPVS++)
{
    /* insert data line */ // LOOP 5 ROUND
    /* insert data line */
    HSYNC10;
}
//.....
//.....
for(LOOPVS=0;LOOPVS<8;LOOPVS++)
{
    /* insert data line */ // LOOP 8 ROUND
    IOSET1 = LCD_RED;
    /* insert data line */
    HSYNC10;
}
IOCLR1 = LCD_RED;
//.....
for(LOOPVS=0;LOOPVS<8;LOOPVS++)
{
    /* insert data line */ // LOOP 8 ROUND ***HSYNC2***
    /* insert data line */
    HSYNC20;
}
//.....
for(LOOPVS=0;LOOPVS<4;LOOPVS++)
{
    /* insert data line */ // LOOP 4 ROUND
    IOSET1 = LCD_RED;
    /* insert data line */
    HSYNC10;
}
IOCLR1 = LCD_RED;
//.....
for(LOOPVS=0;LOOPVS<2;LOOPVS++)
{

```

```

    /* insert data line */
    /* insert data line */
    HSYNC30);
}

//=====
for(LOOPVS=0;LOOPVS<8;LOOPVS++)
{
    /* insert data line */
    /* insert data line */
    HSYNC40);
}

//=====
for(LOOPVS=0;LOOPVS<1;LOOPVS++)
{
    /* insert data line */
    /* insert data line */
    HSYNC30);
}

//=====
for(LOOPVS=0;LOOPVS<9;LOOPVS++)
{
    /* insert data line */
    IOSET1 = LCD_RED;
    /* insert data line */
    HSYNC10);
}
IOCLR1 = LCD_RED;
//=====
for(LOOPVS=0;LOOPVS<4;LOOPVS++)
{
    HSYNC10);
}
}

/*****
/* HSYNC CURSOR + SPECTRUM */
*****/

void HSYNC()
{
    //=====
    IOCLR1 = LCD_HS;
    //=====
    for(LOOPHS=0;LOOPHS<3;LOOPHS++)

```



```

{
DCK();
}
//*****
IOSET1 = LCD_HS; //HSYNC HIGH//
//*****
for(LOOPHS=0;LOOPHS<48;LOOPHS++) // LOOP 48 ROUND
{
DCK();
}
//*****
for(LOOPHS=0;LOOPHS<10;LOOPHS++) // LOOP 10 ROUND
{
/* insert data dot */
IOSET1 = LCD_BLUE;
/* insert data dot */
DCK();
IOCLR1 = LCD_BLUE;
}
//*****
SPECTRUM = 1;
CURSOR = ((cu-1)/7); //|1-7:CH1|,|8-14:CH2|,.....//
//*****
for(LOOPHS=0;LOOPHS<(CURSOR);LOOPHS++) // LOOP 300 times (0 - 299)
{
    if(i <= CH[SPECTRUM*7])
    {
        IOSET1 = LCD_RED;
        IOSET1 = LCD_GREEN;
        DCK();
        IOCLR1 = LCD_RED;
        IOCLR1 = LCD_GREEN;
    }
    else
    {
        IOSET1 = LCD_BLUE;
        DCK();
        IOCLR1 = LCD_BLUE;
    }
}
SPECTRUM++;
}
//*****
for(LOOPHS=0;LOOPHS<1;LOOPHS++) // LOOP 1 CURSOR BLINK FOLLOW cu

```

```

{
    if(i <= CH[SPECTRUM*7])
    {
        IOSET1 = 0X00080000;
        IOSET1 = 0X01000000;
        DCK();
        IOCLR1 = 0X00080000;
        IOCLR1 = 0X01000000;
    }
    else
    {
        IOSET1 = LCD_RED;
        DCK();
        IOCLR1 = LCD_RED;
    }
    SPECTRUM++;
}
/////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<(299-CURSOR);LOOPHS++) // LOOP 300 times (0 - 299)
{
    if(i <= CH[SPECTRUM*7])
    {
        IOSET1 = LCD_RED;
        IOSET1 = LCD_GREEN;
        DCK();
        IOCLR1 = LCD_RED;
        IOCLR1 = LCD_GREEN;
    }
    else
    {
        IOSET1 = LCD_BLUE;
        DCK();
        IOCLR1 = LCD_BLUE;
    }
    SPECTRUM++;
}
/////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<10;LOOPHS++) // LOOP 10 ROUND
{
    /* insert data dot */
    IOSET1 = LCD_BLUE;
    /* insert data dot */
    DCK();
}

```

```

IOCLR1 = LCD_BLUE;
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP 8 ROUND
{
DCK();
}
//*****
}
/*****/
/* HSYNC1 NONE Function */
/*****/
void HSYNC1()
{
//*****
IOCLR1 = LCD_HS;                          ///HSYNC LOW///
//*****
for(LOOPHS=0;LOOPHS<3;LOOPHS++)          // LOOP 3 ROUND
{
DCK();
}
//*****
IOSET1 = LCD_HS;                          ///HSYNC HIGH///
//*****
for(LOOPHS=0;LOOPHS<48;LOOPHS++)         // LOOP 48 ROUND
{
DCK();
}
//*****
for(LOOPHS=0;LOOPHS<320;LOOPHS++)        // LOOP 320 ROUND
{
DCK();
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP 8 ROUND
{
DCK();
}
//*****
}
/*****/
/* HSYNC2 TYPING Function */
/*****/

```

```

void HSYNC20
{
//*****
IOCLR1 = LCD_HS;                               ///HSYNC LOW///
//*****
for(LOOPHS=0;LOOPHS<3;LOOPHS++)                // LOOP 3 ROUND
{
DCK();
}
//*****
IOSET1 = LCD_HS;                               ///HSYNC HIGH///
//*****
for(LOOPHS=0;LOOPHS<48;LOOPHS++)              // LOOP 48 ROUND
{
DCK();
}
//*****
for(LOOPHS=0;LOOPHS<40;LOOPHS++)              // LOOP 40 ROUND
{
/* insert data dot */
IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++)               // LOOP T ROUND *****
{
/* insert data dot */
IOSET1 = Table_T[LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_T[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)              // LOOP I ROUND *****
{
/* insert data dot */
IOSET1 = Table_I[LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_I[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)              // LOOP M ROUND*****

```

```

{
/* insert data dot */
IOSET1 = Table_M[LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_M[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP E ROUND *****
{
/* insert data dot */
IOSET1 = Table_E[LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_E[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP S ROUND *****
{
/* insert data dot */
IOSET1 = Table_S[LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_S[LOOPVS][LOOPHS];
}
//////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<56;LOOPHS++)          // LOOP 56 ROUND
{
/* insert data dot */
IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
//////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP C ROUND *****
{
/* insert data dot */
IOSET1 = Table_C[LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_C[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP O ROUND *****
{

```

```

    /* insert data dot */
    IOSET1 = Table_O[LOOPVVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_O[LOOPVVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP U ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_U[LOOPVVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_U[LOOPVVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP N ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_N[LOOPVVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_N[LOOPVVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP T ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_T[LOOPVVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_T[LOOPVVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP S ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_S[LOOPVVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_S[LOOPVVS][LOOPHS];
}
////////////////////////////////////
for(LOOPHS=0;LOOPHS<48;LOOPHS++)          // LOOP 47 ROUND
{
    /* insert data dot */
    IOSET1 = LCD_RED;

```

```

    /* insert data dot */
    DCK();
    IOCLR1 = LCD_RED;
}

/////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP C ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_C[LOOPVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_C[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP H ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_H[LOOPVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_H[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP A ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_A[LOOPVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_A[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP N ROUND *****
{
    /* insert data dot */
    IOSET1 = Table_N[LOOPVS][LOOPHS];
    /* insert data dot */
    DCK();
    IOCLR1 = Table_N[LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP N ROUND
{
    /* insert data dot */
    IOSET1 = Table_N[LOOPVS][LOOPHS];
    /* insert data dot */
    DCK();
}

```

```

IOCLR1 = Table_N[LOOPVVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP E ROUND
{
  /* insert data dot */
  IOSET1 = Table_E[LOOPVVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_E[LOOPVVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP L ROUND
{
  /* insert data dot */
  IOSET1 = Table_L[LOOPVVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_L[LOOPVVS][LOOPHS];
}
/////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<34;LOOPHS++) // LOOP 34 ROUND
{
  /* insert data dot */
  IOSET1 = LCD_RED;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_RED;
}
/////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP 8 ROUND
{
  DCK();
}
}
/*****
/* HSYNC3 BLACK BOX */
*****/

void HSYNC3()
{
  ///////////////////////////////////////////////////////////////////
  IOCLR1 = LCD_HS; //HSYNC LOW//
  ///////////////////////////////////////////////////////////////////
  for(LOOPHS=0;LOOPHS<3;LOOPHS++) // LOOP 3 ROUND

```



```

{
DCK();
}
//*****
IOSET1 = LCD_HS; //HSYNC HIGH//
//*****
for(LOOPHS=0;LOOPHS<48;LOOPHS++) // LOOP 48 ROUND
{
DCK();
}
//*****
////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<36;LOOPHS++) // LOOP 36 ROUND
{
/* insert data dot */
IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
for(LOOPHS=0;LOOPHS<48;LOOPHS++) // LOOP 48 ROUND
{
/* insert data dot */
IOSET1 = LCD_BLACK;
/* insert data dot */
DCK();
IOCLR1 = LCD_BLACK;
}
for(LOOPHS=0;LOOPHS<52;LOOPHS++) // LOOP 52 ROUND
{
/* insert data dot */
IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
for(LOOPHS=0;LOOPHS<48;LOOPHS++) // LOOP 48 ROUND
{
/* insert data dot */
IOSET1 = LCD_BLACK;
/* insert data dot */
DCK();
}

```

```

IOCLR1 = LCD_BLACK;
}
for(LOOPHS=0;LOOPHS<52;LOOPHS++)           // LOOP 52 ROUND
{
  /* insert data dot */
  IOSET1 = LCD_RED;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_RED;
}
for(LOOPHS=0;LOOPHS<48;LOOPHS++)           // LOOP 48 ROUND
{
  /* insert data dot */
  IOSET1 = LCD_BLACK;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_BLACK;
}
for(LOOPHS=0;LOOPHS<38;LOOPHS++)           // LOOP 38 ROUND
{
  /* insert data dot */
  IOSET1 = LCD_RED;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_RED;
}
//////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND
{
  DCK();
}
}
/*****
/* HSYNC4 NUMBER Function */
*****/
void HSYNC4()
{
  ////////////////////////////////////////////////////////////////////
  IOCLR1 = LCD_HS;                               ///HSYNC LOW///
  ////////////////////////////////////////////////////////////////////
  for(LOOPHS=0;LOOPHS<3;LOOPHS++)           // LOOP 3 ROUND
  {
    DCK();
  }
}

```

```

}
//*****
IOSET1 = LCD_HS; //HSYNC HIGH//
//*****
for(LOOPHS=0;LOOPHS<48;LOOPHS++) // LOOP 48 ROUND
{
DCK();
}
//*****
for(LOOPHS=0;LOOPHS<36;LOOPHS++) // LOOP 36 ROUND
{
/* insert data dot */
IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
//*****
TIME1 = SECOND/10000;
TIME2 = (SECOND-(10000*TIME1))/1000;
TIME3 = ((SECOND-(10000*TIME1))-(1000*TIME2))/100;
TIME4 = (((SECOND-(10000*TIME1))-(1000*TIME2))-(100*TIME3))/10;
TIME5 = ((((SECOND-(10000*TIME1))-(1000*TIME2))-(100*TIME3))-(10*TIME4))/1;
//*****
for(LOOPHS=0;LOOPHS<5;LOOPHS++) // LOOP 8 ROUND BLACK TIMES
{
/* insert data dot */
IOSET1 = LCD_BLACK;
/* insert data dot */
DCK();
IOCLR1 = LCD_BLACK;
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP 8 ROUND NUMBER 1
{
/* insert data dot */
IOSET1 = Table_NUMBER[(TIME1*8)+LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(TIME1*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP 8 ROUND NUMBER 2
{

```

```

/** insert data dot */
IOSET1 = Table_NUMBER[(TIME2*8)+LOOPVS][LOOPHS];
/** insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(TIME2*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 3
{
/** insert data dot */
IOSET1 = Table_NUMBER[(TIME3*8)+LOOPVS][LOOPHS];
/** insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(TIME3*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 4
{
/** insert data dot */
IOSET1 = Table_NUMBER[(TIME4*8)+LOOPVS][LOOPHS];
/** insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(TIME4*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 4
{
/** insert data dot */
IOSET1 = Table_NUMBER[(TIME5*8)+LOOPVS][LOOPHS];
/** insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(TIME5*8)+LOOPVS][LOOPHS];
}
//*****
for(LOOPHS=0;LOOPHS<3;LOOPHS++)           // LOOP 3 ROUND BLACK
{
/** insert data dot */
IOSET1 = LCD_BLACK;
/** insert data dot */
DCK();
IOCLR1 = LCD_BLACK;
}
//*****
for(LOOPHS=0;LOOPHS<52;LOOPHS++)           // LOOP 52 ROUND
{
/** insert data dot */

```

```

IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
/////////////////////////////////////////////////////////////////
NUM1 = CH[cu]/10000;
NUM2 = (CH[cu]-(10000*NUM1))/1000;
NUM3 = ((CH[cu]-(10000*NUM1))-(1000*NUM2))/100;
NUM4 = (((CH[cu]-(10000*NUM1))-(1000*NUM2))-(100*NUM3))/10;
NUM5 = (((((CH[cu]-(10000*NUM1))-(1000*NUM2))-(100*NUM3))-(10*NUM4))/1;
/////////////////////////////////////////////////////////////////
for(LOOPHS=0;LOOPHS<5;LOOPHS++) // LOOP 5 ROUND BLACK COUNTS
{
/* insert data dot */
IOSET1 = LCD_BLACK;
/* insert data dot */
DCK();
IOCLR1 = LCD_BLACK;
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP 8 ROUND NUMBER 1
{
/* insert data dot */
IOSET1 = Table_NUMBER[(NUM1*8)+LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(NUM1*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP 8 ROUND NUMBER 2
{
/* insert data dot */
IOSET1 = Table_NUMBER[(NUM2*8)+LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(NUM2*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++) // LOOP 8 ROUND NUMBER 3
{
/* insert data dot */
IOSET1 = Table_NUMBER[(NUM3*8)+LOOPVS][LOOPHS];
/* insert data dot */
DCK();
}

```

```

IOCLR1 = Table_NUMBER[(NUM3*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 4
{
  /* insert data dot */
  IOSET1 = Table_NUMBER[(NUM4*8)+LOOPVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_NUMBER[(NUM4*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 5
{
  /* insert data dot */
  IOSET1 = Table_NUMBER[(NUM5*8)+LOOPVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_NUMBER[(NUM5*8)+LOOPVS][LOOPHS];
}
//=====
for(LOOPHS=0;LOOPHS<3;LOOPHS++)           // LOOP 3 ROUND BLACK
{
  /* insert data dot */
  IOSET1 = LCD_BLACK;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_BLACK;
}
//=====
for(LOOPHS=0;LOOPHS<52;LOOPHS++)           // LOOP 52 ROUND
{
  /* insert data dot */
  IOSET1 = LCD_RED;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_RED;
}
//=====
CHAN1 = cu/10000;
CHAN2 = (cu-(10000*CHAN1))/1000;
CHAN3 = ((cu-(10000*CHAN1))-(1000*CHAN2))/100;
CHAN4 = (((cu-(10000*CHAN1))-(1000*CHAN2))-(100*CHAN3))/10;
CHAN5 = (((cu-(10000*CHAN1))-(1000*CHAN2))-(100*CHAN3))-(10*CHAN4))/1;
//=====

```

```

for(LOOPHS=0;LOOPHS<5;LOOPHS++)           // LOOP 5 ROUND BLACK CHANNEL
{
  /* insert data dot */
  IOSET1 = LCD_BLACK;
  /* insert data dot */
  DCK();
  IOCLR1 = LCD_BLACK;
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 1
{
  /* insert data dot */
  IOSET1 = Table_NUMBER[(CHAN1*8)+LOOPVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_NUMBER[(CHAN1*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 2
{
  /* insert data dot */
  IOSET1 = Table_NUMBER[(CHAN2*8)+LOOPVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_NUMBER[(CHAN2*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 3
{
  /* insert data dot */
  IOSET1 = Table_NUMBER[(CHAN3*8)+LOOPVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_NUMBER[(CHAN3*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 4
{
  /* insert data dot */
  IOSET1 = Table_NUMBER[(CHAN4*8)+LOOPVS][LOOPHS];
  /* insert data dot */
  DCK();
  IOCLR1 = Table_NUMBER[(CHAN4*8)+LOOPVS][LOOPHS];
}
for(LOOPHS=0;LOOPHS<8;LOOPHS++)           // LOOP 8 ROUND NUMBER 5
{

```

```

/* insert data dot */
IOSET1 = Table_NUMBER[(CHAN5*8)+LOOPVS][LOOPHS];
/* insert data dot */
DCK();
IOCLR1 = Table_NUMBER[(CHAN5*8)+LOOPVS][LOOPHS];
}
//*****
for(LOOPHS=0;LOOPHS<3;LOOPHS++)          // LOOP 3 ROUND BLACK
{
/* insert data dot */
IOSET1 = LCD_BLACK;
/* insert data dot */
DCK();
IOCLR1 = LCD_BLACK;
}
//*****
for(LOOPHS=0;LOOPHS<38;LOOPHS++)          // LOOP 38 ROUND
{
/* insert data dot */
IOSET1 = LCD_RED;
/* insert data dot */
DCK();
IOCLR1 = LCD_RED;
}
//*****
for(LOOPHS=0;LOOPHS<8;LOOPHS++)          // LOOP 8 ROUND
{
DCK();
}
//*****
}
/*****
/* DCK Function */
/*****
void DCK()
{
IOCLR1 = LCD_DK;
IOSET1 = LCD_DK;
}
/*****
/* DELAY */
/*****
void delay_ms(long int ms)

```



```
{  
  long int x,y;  
  for(x=0;x<ms;x++)  
  for(y=0;y<1000;y++);  
}  
////////////////////////////////////  
//////      END      //////////  
////////////////////////////////////
```



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายสุภฤกษ์ ลีตระกูลพิชิตชัย เกิดวันที่ 15 เมษายน พ.ศ. 2520 ที่จังหวัดขอนแก่น สำเร็จ การศึกษาระดับปริญญาบัณฑิตจากภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เมื่อปีการศึกษา 2544 และในปีการศึกษา 2546 ได้เข้าศึกษาระดับปริญญาโทที่ภาควิชาวิศวกรรมเทคโนโลยี คณะ วิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย