

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การสุ่มสัญญาณแบบสม่ำเสมอเพื่อประยุกต์ใช้กับระบบเครื่องมือวัดและควบคุม
Uniformly Sampled Signal with Applications to Instruments and Controls



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2538

ISBN 974-621-300-8

ลิขสิทธิ์ของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่จำกัดทุกสิ่งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าพนักงานหอสมุด
เลขหมู่.....
เลขทะเบียน..... 23562
วัน, เดือน, ปี 6 ก.ค. 2538

Uniformly Sampled Signal with Applications to Instruments and Controls



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING
GRADUATE SCHOOL
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

1995

ISBN 974-621-300-8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title Uniformly Sampled Signal with Applications to Instruments and Controls
Student Mr. Jirasak Chanwutitum
Thesis Advisor Assoc.Prof.Dr. Kobchai Dejhan
Level of Study Master of Engineering
Department Telecommunication Engineering King Mongkut's Institute of
Technology Ladkrabang
Year 1995

Abstract

The desired frequency of the digital signal processing system depends upon the sampling rate, in normal situation of double of signal frequency has to be considered. The needs is the sampling rate speed of A/D is high or its conversion time. This thesis concerns with the technical concept of the low speed A/D by increasing rate based on the principle of time-interleaving. A/D have to be in parallel connection and must be controlled by the timing control circuit. Then, it will result the constant and equal sampling rate to be uniformly sampled signal. In addition, the data acquisition unit of the instrumentation and control system will be improved in the case of speed by using this technique, many applications of the high speed and high frequency of signal processing unit can be implemented.

กิตติกรรมประกาศ

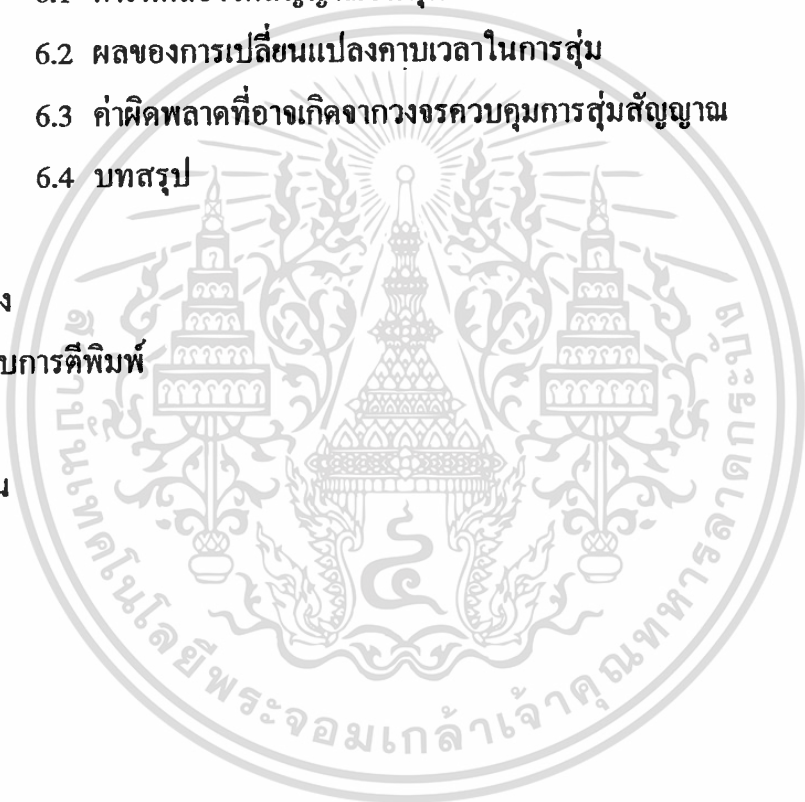
ผู้เขียนขอขอบคุณ รศ.ดร.กอบชัย เดชหาญ ซึ่งเป็นผู้ควบคุมวิทยานิพนธ์ ในการให้คำปรึกษาและคำแนะนำเกี่ยวกับการทำวิจัยด้วยดีตลอดมา ทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงตามวัตถุประสงค์ที่ได้วางไว้แต่ต้น และขอขอบคุณบัณฑิตวิทยาลัยที่ได้กรุณาอบทุนอุดหนุนการทำวิทยานิพนธ์ ทำให้การจัดหาอุปกรณ์ต่างๆ มีความสะดวกและคล่องตัวมากยิ่งขึ้น ขอขอบคุณ รศ.ดร.มนัส สังวรศิลป์ คณบดีบัณฑิตวิทยาลัย ที่กรุณาให้คำแนะนำเกี่ยวกับการจัดทำข้อสรุปของหนังสือวิทยานิพนธ์ฉบับนี้ ซึ่งทำให้มีความสมบูรณ์มากยิ่งขึ้น ทั้งนี้ผู้เขียนขอขอบคุณผู้ที่มีส่วนเกี่ยวข้องที่ช่วยอำนวยความสะดวกต่างๆ ในการดำเนินงานจัดทำหนังสือวิทยานิพนธ์ฉบับนี้จนสามารถสำเร็จลุล่วงได้อย่างสมบูรณ์

นายจิระศักดิ์ ชาญวุฒิชัยธรรม

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VI
บทที่ 1 บทนำ	1
1.1 แนวความคิดของวิทยานิพนธ์	1
1.2 วัตถุประสงค์ของวิทยานิพนธ์	2
บทที่ 2 ทฤษฎีการสุ่มสัญญาณ	3
2.1 อัตราการสุ่มสัญญาณของไนควิสต์ (Nyquist)	3
2.2 การสุ่มสัญญาณแบบสม่ำเสมอ	9
บทที่ 3 คุณสมบัติของวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	10
3.1 ลักษณะต่างๆ ไปของวงจรเอทวดี	10
3.2 ความเร็วในการแปลงสัญญาณ	20
3.3 ค่าผิดพลาดที่เกิดขึ้นจากการสุ่มสัญญาณ	22
บทที่ 4 หลักการ Time-Interleaving กับการสุ่มสัญญาณแบบสม่ำเสมอ	27
4.1 การใช้เอทวดีในการแปลงสัญญาณจำนวน 1 ตัว	27
4.2 การใช้เอทวดีในการแปลงสัญญาณจำนวนมากกว่า 1 ตัว	29
4.3 การควบคุมคาบเวลาในการสุ่มสัญญาณ	31

สารบัญ	หน้า
บทที่ 5 การประยุกต์ใช้งาน	35
5.1 การเชื่อมโยงวงจรเข้ากับระบบคอมพิวเตอร์	35
5.2 โปรแกรมที่ใช้ในการควบคุมการทำงานของวงจร	40
บทที่ 6 ผลการทดลองและบทสรุป	47
6.1 การทดลองวัดสัญญาณอินพุตด้วยไมโครคอนโทรลเลอร์	47
6.2 ผลของการเปลี่ยนแปลงคาบเวลาในการสุ่ม	54
6.3 ค่าผิดพลาดที่อาจเกิดจากวงจรควบคุมการสุ่มสัญญาณ	54
6.4 บทสรุป	58
เอกสารอ้างอิง	59
ผลงานที่ได้รับการตีพิมพ์	60
ภาคผนวก	61
ประวัติผู้เขียน	75



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ

	หน้า
1 แสดงช่วงเวลาเหลื่อมล้ำของเอทูดี้	2
2.1 เปรียบเทียบสัญญาณอนาลอกกับสัญญาณดิจิตอล	3
2.2 บล็อกไดอะแกรมของการแปลงสัญญาณ Continuous-Time เป็น Discrete Time	4
2.3 แสดงการสุ่มสัญญาณ $x(t)$	5
2.4 สัญญาณมอดคูลเลทระหว่าง $x(t)$ กับ $p(t)$	5
2.5 Fourier Transform ของสัญญาณในรูปที่ 2.2	6
2.6 สเปกตรัมของสัญญาณที่มีค่า $f_s - f_M < f_M$	7
2.7 แสดงการสุ่มสัญญาณซายน์ด้วยความถี่ค่าต่างๆกัน	7
3.1 ไดอะแกรมวงจร single-slope A/D converter	11
3.2 ไดอะแกรมวงจร dual-slope A/D converter	12
3.3 ไดอะแกรมวงจร successive approximation A/D converter	14
3.4 หลักการทำงานของ successive approximation A/D ขนาด 4 บิต	15
3.5 ค่าดิจิตอลเอาท์พุทกับค่าแรงดันอนาลอกอินพุท	16
3.6 ตัวอย่างการแปลงสัญญาณของ 4-bit successive approximation A/D	17
3.7 เอทูดี้ขนาด 3 บิต แบบ flash converter	18
3.9 เอทูดี้ 8 บิต แบบ flash converter 4 บิต 2 ตัว	19
3.10 บล็อกไดอะแกรมการแปลงสัญญาณอนาลอกเป็นดิจิตอล	20
3.11 แรงดันอนาลอกอินพุทกับดิจิตอลเอาท์พุทขนาด 3 บิต	23
3.12 ค่าผิดพลาดที่เกิดกับเอทูดี้	23
3.13 ความสัมพันธ์ระหว่าง T_C กับระดับแรงดันอินพุท	25
4.1 การสุ่มสัญญาณแบบสม่ำเสมอ มีคาบเวลา = T	27
4.2 ช่วงเวลาในการสุ่มสัญญาณของเอทูดี้ตัวเดียว	28
4.3 การแปลงสัญญาณแบบ Time-Interleaving	29
4.4 การใช้เอทูดี้ 2 ตัว ทำงานแบบ Time-Interleaving	30

4.5	ช่วงเวลาในการสุ่มสัญญาณแบบสม่ำเสมอของเอทูดี้ 2 ตัว	31
4.6	บล็อกไคอะแกรมของวงจรควบคุมเอทูดี้ตัวที่ 2	32
4.7	วงจรควบคุมการทำงานของเอทูดี้ตัวที่ 2	33
5.1	บล็อกไคอะแกรมของ Data Acquisition Unit	35
5.2	การต่อ Time-Interleaving เอทูดี้เข้ากับคอมพิวเตอร์	36
5.3	Timing ไคอะแกรมการทำงานของเอทูดี้ตัวที่ 1 และ 2	36
5.4	ซีพียูบอร์ด	37
5.5	Data Acquisition Unit แบบใช้เอทูดี้ 2 ตัว	38
5.6	วงจรพอร์ทอนุกรมสำหรับ 8031	39
5.8	โพล์ซาร์ทสำหรับ Data Acquisition แบบใช้เอทูดี้ตัวเดียว	40
5.9	โพล์ซาร์ทสำหรับ Data Acquisition แบบใช้เอทูดี้ 2 ตัว	43
6.1	สัญญาณ chip select ของ AD1	48
6.2	สัญญาณชาน์ความถี่ 1kHz ที่ได้จากการสุ่มโดยใช้เอทูดี้ตัวเดียว แสดงเป็นแบบ Discrete	48
6.3	สัญญาณชาน์ความถี่ 1kHz ที่ได้จากการสุ่มโดยใช้เอทูดี้ตัวเดียว แสดงเป็นแบบ Continuous	49
6.4	สัญญาณ chip select ของ AD1 และ AD2	50
6.5	สัญญาณชาน์ความถี่ 1kHz ที่ได้จากการสุ่มโดยใช้เอทูดี้สองตัว แสดงเป็นแบบ Discrete	50
6.6	สัญญาณชาน์ความถี่ 1kHz ที่ได้จากการสุ่มโดยใช้เอทูดี้สองตัว แสดงเป็นแบบ Continuous	51
6.7	สัญญาณชาน์ 1kHz ใช้เอทูดี้ตัวเดียวที่มีอัตราสุ่ม 40000 Samples/sec	52
6.8	ผลการทดลองสุ่มสัญญาณ โดยใช้เอทูดี้ตัวเดียว	54
6.9	ผลการทดลองสุ่มสัญญาณ โดยใช้เอทูดี้สองตัว	56

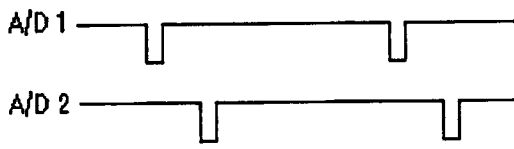
บทที่ 1

บทนำ

งานวิจัยที่ได้นำเสนอในวิทยานิพนธ์ฉบับนี้เป็นเทคนิคการเพิ่มอัตราการสุ่มสัญญาณให้กับไอซีเอทีประเภท Successive Approximation โดยที่คาบเวลาในการสุ่มมีค่าเท่ากันตลอด แนวความคิดและวัตถุประสงค์ในการดำเนินงานเพื่อจัดทำวิทยานิพนธ์ดังกล่าวมีดังต่อไปนี้

1.1 แนวความคิดของวิทยานิพนธ์

ในการประมวลสัญญาณดิจิทัล โครงสร้างทางฮาร์ดแวร์ที่ทำหน้าที่รับสัญญาณอนาล็อกอินพุตนั้น เรียกว่า Data Acquisition Unit มีหน้าที่หลักคือแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลส่งให้กับระบบคอมพิวเตอร์ ความถี่ของสัญญาณที่ต้องการประมวลผลจะขึ้นอยู่กับอัตราการสุ่มของสัญญาณ โดยปกติจะต้องมีค่าไม่น้อยกว่า 2 เท่าของความถี่ของสัญญาณที่ต้องการวัด องค์ประกอบหลักของอัตราการสุ่มคือความเร็วในการแปลงสัญญาณของเอที เอทีประเภท Successive Approximation จะมีค่า Conversion Time เป็น μs โดยมีค่าตั้งแต่ $6 \mu\text{s}$ ถึง $125 \mu\text{s}$ ขึ้นอยู่กับไอซีแต่ละเบอร์ เมื่อเทียบกับไอซีเอทีประเภท Flash Converter ที่มีความเร็วเป็น nS ในการสร้าง Data Acquisition Unit เพื่อใช้ในระบบเครื่องมือวัดและควบคุม^[3] ที่สัญญาณอินพุตมีการเปลี่ยนแปลงสูงจำเป็นต้องใช้เอทีที่มีความเร็วสูงด้วยเช่นกัน เนื่องจากความเร็วในการแปลงสัญญาณของไอซีเอทีแต่ละเบอร์จะมีค่าคงที่ เพื่อเป็นการเพิ่มความเร็วในการแปลงสัญญาณให้กับเอทีประเภท Successive Approximation จึงได้มีการใช้ไอซีเอที จำนวน 2 ตัวต่อขนานกันเพื่อวัดสัญญาณที่จุดๆเดียวกัน และให้การทำงานของเอทีทั้ง 2 ตัวมีช่วงเวลาที่เหลื่อมล้ำกัน ดังรูปที่ 1 ก็จะทำให้ได้อัตราการสุ่มสัญญาณเพิ่มขึ้น



รูปที่ 1 แสดงช่วงเวลาเหลื่อมล้ำของเอชดูดี

1.2 วัตถุประสงค์ของวิทยานิพนธ์

จากแนวความคิดดังกล่าวที่ต้องการเพิ่มอัตราการสุ่มสัญญาณให้กับไอซีเอชดูดีจึงได้ทำการทดลองและสร้างวงจรควบคุมการสุ่มสัญญาณแบบสุ่มสม่ำเสมอ โดยมีวัตถุประสงค์เพื่อทดลองแนวความคิดดังกล่าวว่าสามารถสร้างขึ้นได้ เพื่อเป็นแนวทางในการศึกษาและค้นคว้าเกี่ยวกับทฤษฎีการสุ่มสัญญาณสำหรับการแปลงค่าสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัล ตลอดจนจนการออกแบบวงจรเพิ่มอัตราการสุ่มสัญญาณเพื่อให้คอมพิวเตอร์สามารถประมวลผลสัญญาณไฟฟ้าที่มีความถี่สูงขึ้นกว่าเดิมได้ และเพื่อเป็นการประยุกต์ใช้กับระบบเครื่องมือวัดและความคุมด้วยคอมพิวเตอร์ต่างๆไป

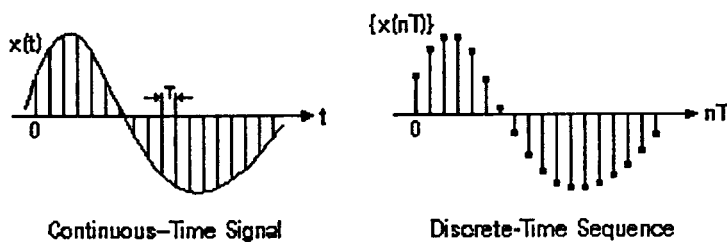
บทที่ 2

ทฤษฎีการสุ่มสัญญาณ

ในการประมวลผลหรือวิเคราะห์สัญญาณไฟฟ้าแบบดิจิทัล หัวใจสำคัญชุดแรกก็คือวงจรการแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัล ค่าของสัญญาณดิจิทัลแต่ละค่าจะได้มาจากการแปลงสัญญาณของวงจรเอทูดิ ซึ่งเปรียบเสมือนเป็นการสุ่มสัญญาณอนาลอกที่เวลาใดเวลาหนึ่ง ที่วงจรเอทูดิเริ่มต้นทำการแปลงสัญญาณ อัตราการสุ่มสัญญาณนี้จะมีผลต่อความถูกต้องระหว่างสัญญาณอนาลอกกับสัญญาณดิจิทัล ถ้าอัตราการสุ่มสัญญาณมีค่าสูง จะทำให้สัญญาณดิจิทัลที่ได้มีค่าผิดเพี้ยนจากค่าจริงน้อย แต่ถ้าอัตราการสุ่มของสัญญาณมีค่าต่ำ ก็จะทำให้สัญญาณดิจิทัลที่ได้มีความผิดเพี้ยนมาก ทำให้การประมวลผลหรือวิเคราะห์สัญญาณไฟฟ้าได้ไม่ถูกต้องตามความเป็นจริง

2.1 อัตราการสุ่มสัญญาณของไนควิสต์

โดยปกติสัญญาณไฟฟ้าจะมีค่าเปลี่ยนไปตามเวลา ในเชิงคณิตศาสตร์สามารถกล่าวได้ว่าสัญญาณไฟฟ้าเป็นฟังก์ชันของเวลา สัญญาณไฟฟ้าสามารถแบ่งได้เป็น 2 ประเภทคือ Continuous-time Signal และ Discrete-time Signal สัญญาณ Continuous คือสัญญาณที่มีค่าต่อเนื่องตลอดเวลา เมื่อนำเอาขนาดของสัญญาณมาเขียนเทียบกับเวลา จะได้เป็นเส้นที่ต่อเนื่องกัน สัญญาณนั้นจะถูกเรียกว่าสัญญาณอนาลอก สัญญาณ Discrete คือสัญญาณที่มีค่าเป็นช่วงๆ สัญญาณจะมีค่าเฉพาะบางช่วงเวลาเท่านั้น โดยเวลาจะมีค่าไม่ต่อเนื่องมีค่าเฉพาะบางจุดเช่นเดียวกัน ปกติแล้วสัญญาณ Discrete จะอยู่ในรูปของลำดับของตัวเลข

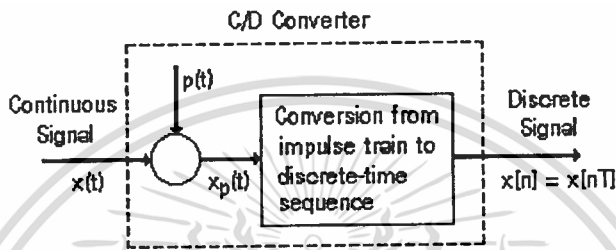


รูปที่ 2.1 เปรียบเทียบสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ Discrete นี้ เรียกว่าสัญญาณดิจิทัล ลักษณะของสัญญาณทั้ง 2 แสดงดังรูปที่ 2.1
 การแทนค่าสัญญาณ Continuous-Time ด้วย Discrete-Time ที่ได้จากการสุ่ม
 สัญญาณแบบสม่ำเสมอ แสดงดังรูปที่ 2.2 โดยมีสมการดังนี้

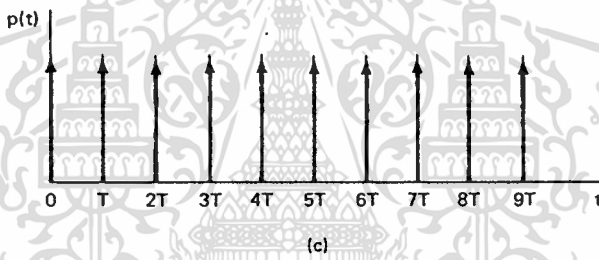
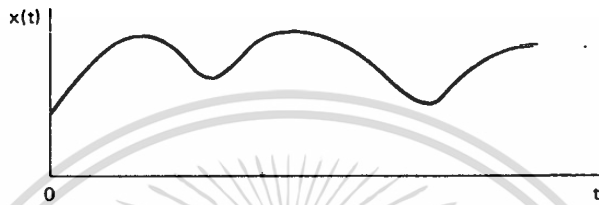
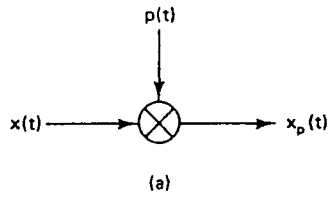
$$x[n] = x(nT), \quad -\infty < n < +\infty$$



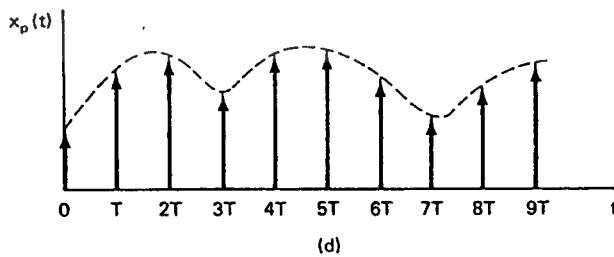
รูปที่ 2.2 บล็อกโคแอดเวอร์ชันของการแปลงสัญญาณ
 Continuous-Time เป็น Discrete-Time

- ค่า T คือ ช่วงเวลาในการสุ่ม
- f_s คือ ความถี่ในการสุ่มมีค่าเท่ากับ $1/T$
- C/D คือ Continuous-to-Discrete-Time Converter
- $p(t)$ คือ Periodic Impulse Train
- $x_p(t)$ คือ สัญญาณมอดคูลระหว่าง $p(t)$ กับ $x(t)$

ลักษณะของการสุ่มสัญญาณแสดงดังรูปที่ 2.3 สัญญาณ $x(t)$ จะเป็นสัญญาณอนาลอกทั่วไป สัญญาณ $p(t)$ เป็นสัญญาณ Impulse เป็นช่วงๆ เปรียบเสมือนสัญญาณ Carrier มีช่วงเวลาระหว่าง Impulse เท่ากับ T ถ้านำสัญญาณทั้งสองมอดคูลกัน โดยให้ขนาดของสัญญาณถูกจำกัดด้วยขนาดของสัญญาณ $x(t)$ จะได้สัญญาณ $x_p(t)$ ตามรูปที่ 2.4 สัญญาณ Impulse ที่ได้ใหม่นี้จะมีขนาดตามสัญญาณอนาลอก $x(t)$ เป็นการพิจารณาในโดเมนของเวลา สัญญาณที่กล่าวมานี้ ถ้าพิจารณาในโดเมนของความถี่โดยวิธี Fourier Transform จะได้สัญญาณตาม รูปที่ 2.5

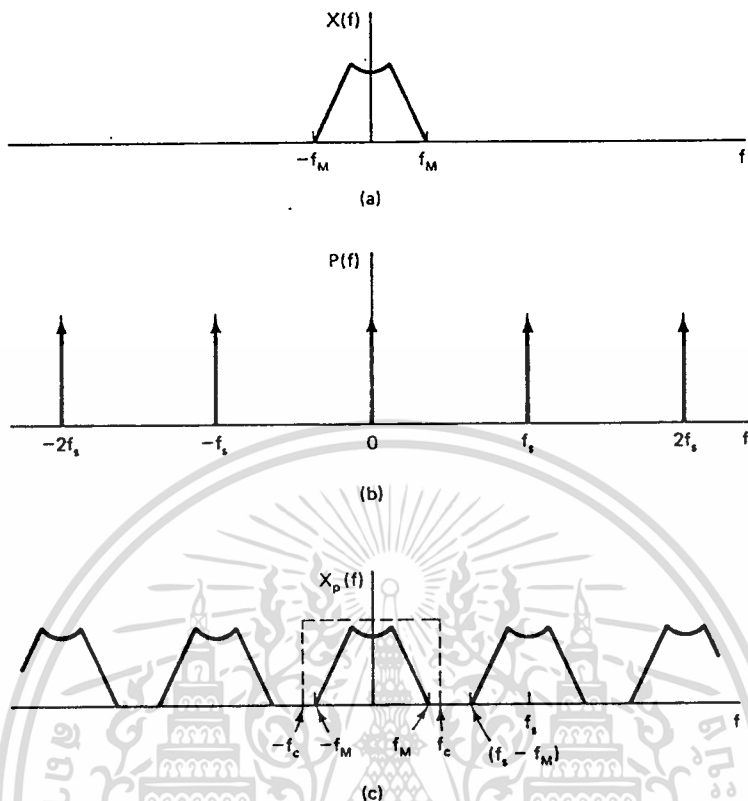


รูปที่ 2.3 แสดงการสุ่มสัญญาณ $x(t)$



รูปที่ 2.4 สัญญาณมอดูเลทระหว่าง $x(t)$ กับ $p(t)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



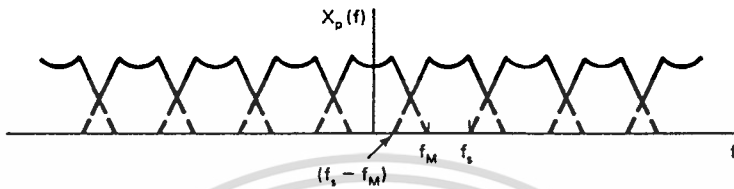
รูปที่ 2.5 Fourier Transform ของสัญญาณในรูปที่ 2.2

สัญญาณ $X(f)$ จะแทนสเปกตรัมความถี่ของสัญญาณอนาล็อก $x(t)$ จะเห็นได้ว่า $X(f) = 0$ ที่ความถี่สูงกว่าความถี่ f_M นั่นคือสัญญาณ $x(t)$ จะไม่มีส่วนประกอบของสัญญาณที่มีความถี่สูงกว่าความถี่ f_M สัญญาณ $p(f)$ จะแทนสเปกตรัมของสัญญาณ Impulse $p(t)$ โดยมีระยะห่างเท่ากับความถี่ของสัญญาณ Impulse $p(t)$ ซึ่งความถี่ของสัญญาณ Impulse ก็คือความถี่ในการสุ่มสัญญาณนั่นเอง ใช้สัญลักษณ์ f_s (Sampling frequency) สัญญาณ $X_p(f)$ แทนสเปกตรัมของสัญญาณที่ได้มอดคูลเลทแล้ว $x_p(t)$ โดยจะมีรูปร่างเหมือนกับสัญญาณ $X(f)$ แต่จะมีจำนวนหลายๆ รูปตามจำนวนของสัญญาณ Impulse โดยมีระยะห่างเท่ากับความถี่ในการสุ่มสัญญาณ (f_s)

การรวบรวมค่าสเปกตรัมทั้งหมดใน โดเมนความถี่คือการคืนสภาพของสัญญาณอนาล็อก $x(t)$ จากสัญญาณที่ได้สุ่มมา เพื่อที่จะให้ได้สัญญาณคืนสภาพกลับมาได้อย่างถูกต้อง ความถี่คัทออฟของวงจรรองความถี่ต่ำแสดงเป็นเส้นประในรูปที่ 2.5 จะต้องมีค่ามากกว่า

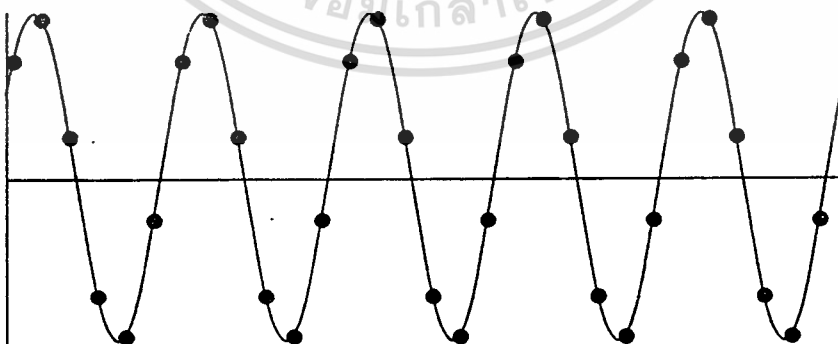
ความถี่ f_M เพื่อที่จะไม่ให้สเปกตรัมของสัญญาณเหลื่อมล้ำกัน ความถี่ในการสุ่มสัญญาณ f_s จะต้องมามีค่า

$$f_s - f_M \geq f_M \text{ หรือ } f_s \geq 2f_M$$

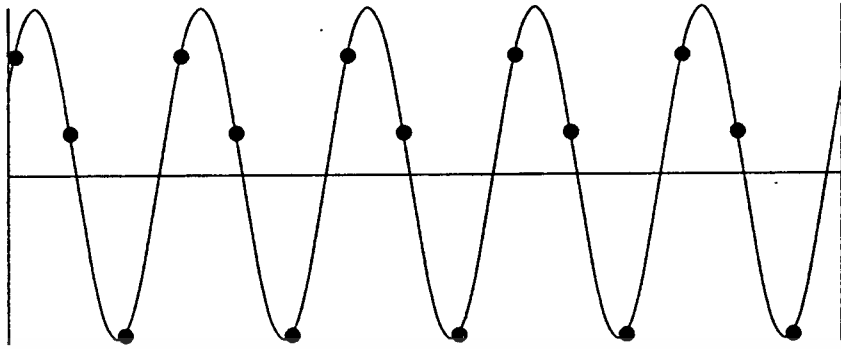


รูปที่ 2.6 สเปกตรัมของสัญญาณที่มีค่า $f_s - f_M < f_M$

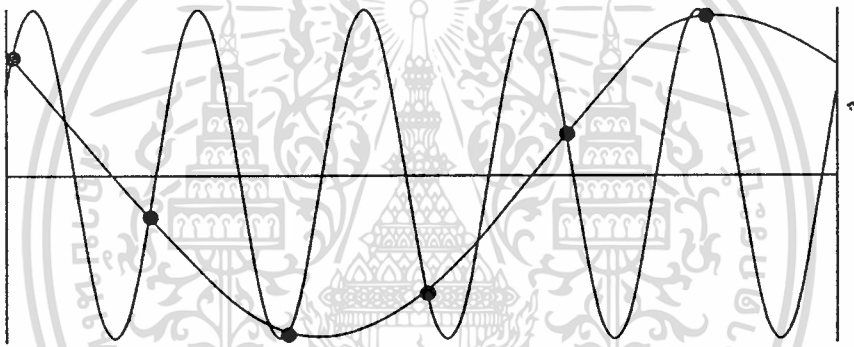
เมื่ออัตราการสุ่มสัญญาณมีความถี่น้อยกว่า 2 เท่าของสัญญาณที่จะวัด ($f_s < 2f_M$) แสดงว่า $f_s < f_M$ มีค่าน้อยกว่า f_M สเปกตรัมของสัญญาณจะเกิดการเหลื่อมล้ำกัน แสดงดังรูปที่ 2.6 ทำให้การคืนสภาพของสัญญาณอนาล็อก $x(t)$ จากสัญญาณที่ได้สุ่มมานั้นมีค่าผิดพลาดขึ้น ค่าผิดพลาดที่เกิดขึ้นเนื่องจากอัตราการสุ่มสัญญาณมีความถี่น้อยกว่า 2 เท่าของสัญญาณที่ต้องการวัดนี้ เรียกว่า Aliasing หรือ Frequency Aliasing ลักษณะของการเกิด Aliasing สามารถแสดงให้เห็นได้ ดังรูปที่ 2.7



รูปที่ 2.7 แสดงการสุ่มสัญญาณชายน้ด้วยความถี่ค่าต่างๆ กัน
ก) ความถี่การสุ่มสัญญาณเท่ากับ 6 เท่าของสัญญาณชายน้



ข) ความถี่การสุ่มสัญญาณเท่ากับ 3 เท่าของสัญญาณชาชนัน



ค) ความถี่การสุ่มสัญญาณเท่ากับ 1.2 เท่าของสัญญาณชาชนัน
รูปที่ 2.7 แสดงการสุ่มสัญญาณชาชนันด้วยความถี่ค่าต่างๆ กัน

จากรูป ความถี่ในการสุ่มสัญญาณที่มีค่ามากกว่า 2 เท่า ของความถี่สัญญาณชาชนัน จะทำให้สัญญาณที่ได้จากการสุ่มนี้มีความถี่เท่าเดิม และถ้าความถี่ในการสุ่มสัญญาณยิ่งมีค่ามาก ความถี่ของสัญญาณยิ่งมีค่าน้อยลง จากรูป 2.7 ค. ความถี่ในการสุ่มสัญญาณมีค่า 1.2 เท่าของความถี่สัญญาณชาชนัน (น้อยกว่า 2 เท่า) มีผลทำให้เกิด Aliasing คือสัญญาณที่ได้จากการสุ่มจะมีความถี่น้อยกว่าสัญญาณจริง ทำให้การประมวลผลสัญญาณเกิดการผิดพลาดได้ ดังนั้นค่าความถี่ต่ำสุดในการสุ่มสัญญาณคือ $f_s = 2f_M$ เรียกความถี่นี้ว่าความถี่ไนควิสต์ (Nyquist frequency) หรืออัตราการสุ่มของไนควิสต์ (Nyquist Sampling Rate) ในการป้องกันไม่ให้คอมพิวเตอร์ประมวลผลสัญญาณผิดพลาดอันเนื่องมาจาก Aliasing จึงมี

การต่อวงจรกรองความถี่ต่ำให้กับสัญญาณที่ต้องการจะวัด เรียกว่าวงจรนี้ว่า Anti-Aliasing Filter โดยให้มีจุดคัทออฟต่ำกว่าครึ่งหนึ่งของความถี่ที่ใช้ในการสุ่มสัญญาณ โดยสรุปแล้ว ทฤษฎีในการสุ่มสัญญาณจะต้องมีการจำกัดแถบความถี่ของสัญญาณ และความถี่ในการสุ่มสัญญาณจะต้องมีค่าไม่น้อยกว่า 2 เท่าของความถี่ของสัญญาณที่ต้องการวัด

2.2 การสุ่มสัญญาณแบบสม่ำเสมอ

หลักการส่วนใหญ่ทางคณิตศาสตร์ของทฤษฎีการประมวลผลสัญญาณดิจิทัล จะสมมุติให้อัตราการสุ่มสัญญาณมีค่าสม่ำเสมอ (Uniformly Sampling or Periodic Sampling) โดยกำหนดให้มีค่าเท่ากับ T ในทางปฏิบัติการแปลงสัญญาณ Continuous-Time ให้เป็น Discrete-Time จะใช้ไอซีเอทูดิ (Analog to Digital Converter) ทำหน้าที่แปลงสัญญาณ ซึ่งจะได้กล่าวต่อไปในบทที่ 3 จากรูป 2.3 สัญญาณ $p(t)$ คือสัญญาณที่ใช้ในการควบคุมให้เอทูดิเริ่มต้นแปลงสัญญาณ หรือสัญญาณ Start of Conversion ของเอทูดินั้นเอง สัญญาณ $p(t)$ นี้ได้มาจากระบบคอมพิวเตอร์เป็นตัวสั่ง เนื่องจากคอมพิวเตอร์มีการทำงานเป็นแบบวนรูป ดังนั้นช่วงเวลาในการสุ่ม T จึงมีค่าเท่ากันตลอด ทำให้สามารถประมวลผลสัญญาณได้ตามหลักการประมวลผลสัญญาณดิจิทัลทั่วไป แต่ในกรณีที่ค่า T มีค่าไม่เท่ากันตลอด เนื่องมาจากการเขียน โปรแกรมควบคุมไม่ดี หรือมีการเพิ่มเติมวงจรให้ทำงานเป็นแบบ Time-Interleaving ซึ่งจะกล่าวในบทที่ 4 จะทำให้การสุ่มสัญญาณเป็นแบบไม่สม่ำเสมอ (Nonuniformly Sampled Signal) ลำดับข้อมูลดิจิทัลที่ได้จึงไม่สามารถนำมาคำนวณตามหลักการประมวลผลสัญญาณดิจิทัลทั่วไป หรือ Fourier Transform ได้

บทที่ 3

คุณสมบัติของวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

เป็นที่ทราบกันดีแล้วว่า ระบบไมโครโปรเซสเซอร์หรือคอมพิวเตอร์มีการทำงานในรูปแบบของสัญญาณดิจิทัล ซึ่งสัญญาณไฟฟ้าโดยทั่วๆ ไปจะเป็นสัญญาณอนาลอก เช่น แรงดัน กระแส อุณหภูมิ ความดันและอื่นๆ อีกมากมาย ดังนั้นการนำระบบคอมพิวเตอร์มาวิเคราะห์สัญญาณอนาลอกจึงต้องมีการแปลงรูปแบบสัญญาณให้อยู่ในรูปสัญญาณดิจิทัลเสียก่อน วงจรที่ทำหน้าที่ในการแปลงสัญญาณนี้เรียกว่า วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (Analog to Digital Converter) หรือวงจรเอทีซี วงจรดังกล่าวถูกผลิตขึ้นมาเป็นชิปไอซีเรียกว่า ไอซีเอทีซี (A/D Integrated Circuit)

3.1 ลักษณะต่างๆ ไปของวงจรเอทีซี

หลักการของวงจรเอทีซี แบ่งเป็นประเภทใหญ่ ๆ ได้ ดังนี้

- Integrating A/D Converter
- Successive Approximation A/D Converter
- Flash A/D Converter

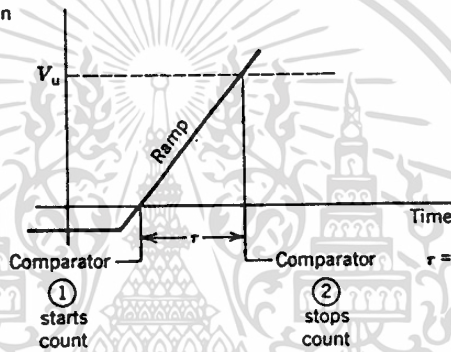
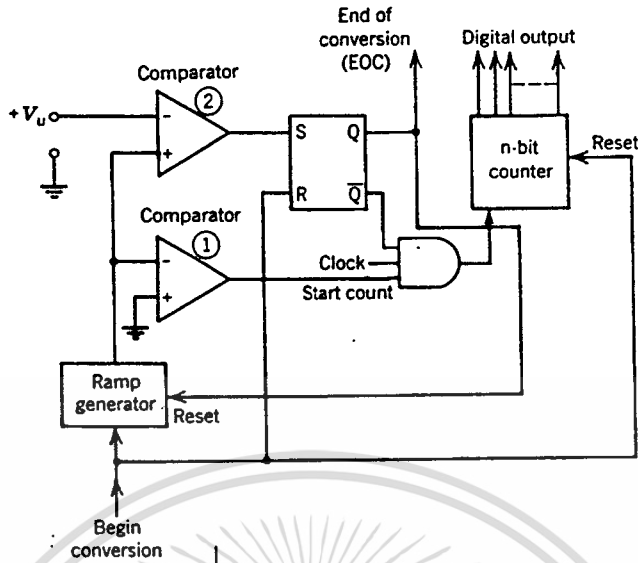
3.1.1 Integrating A/D converter

การแปลงสัญญาณโดยวิธีอินทิเกรตแรงดันอินพุต เพื่อเปลี่ยนค่าให้อยู่ในรูปของเวลา ซึ่งมีความสัมพันธ์กับ Counter หรือดิจิทัลเอาต์พุต มีวิธีการด้วยกัน 2 แบบคือ Single-slope และ Dual-slope หรือ Single-ramp และ Dual-ramp ตามลำดับ

Single-slope (Single-ramp) A/D converter

ประกอบด้วยวงจรกำเนิดสัญญาณ Ramp โดยมีระดับอ้างอิงที่ค่าลบ และจะเพิ่มค่าตัดที่จุดศูนย์ จนกระทั่งเท่ากับแรงดัน Input (V_U) ช่วงระยะเวลาตั้งแต่สัญญาณ Ramp ตัดกับจุดศูนย์จนถึงแรงดันอินพุต ให้มีค่าเท่ากับ τ ซึ่งเป็นสัดส่วนกับแรงดันอินพุต ดังรูปที่ 3.1 เนื่องจาก n-bit counter ทำงานด้วยสัญญาณนาฬิกาซึ่งมีค่าเวลาคงที่ (เมื่อ

สัญญาณ Ramp มีค่าเท่ากับศูนย์ n-bit counter จึงจะเริ่มทำงานนับ โดยเริ่มต้นจากศูนย์เช่นกัน และจะหยุดนับเมื่อสัญญาณ Ramp มีค่าเท่ากับแรงดันอินพุต)

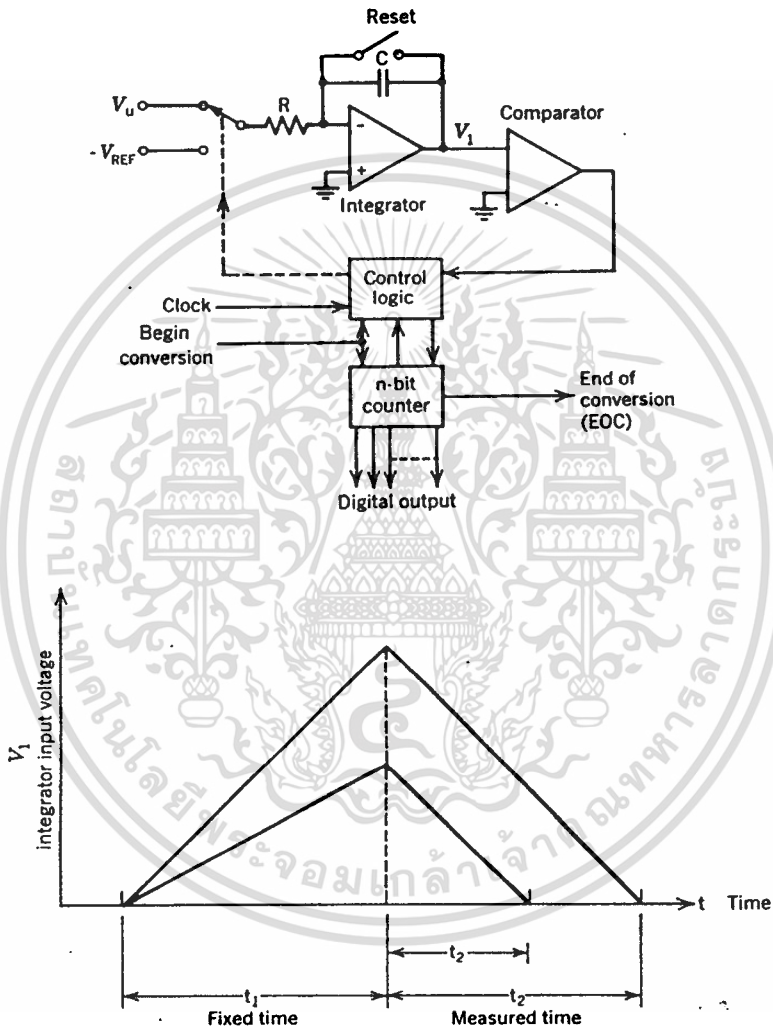


รูปที่ 3.1 โค้ดแอมพลิจูด single-slope A/D converter

รูปที่ 3.1 แสดงโค้ดแอมพลิจูดของ single-slope A/D การทำงาน เมื่อสัญญาณ Ramp มีค่าเท่ากับศูนย์ n-bit counter จะเริ่มนับจากศูนย์โดย Comparator ตัวที่ 1 และจะหยุดนับเมื่อสัญญาณ Ramp มีค่าเท่ากับแรงดันอินพุตโดย Comparator ตัวที่ 2 ช่วงที่ n-bit counter นับนี้จะเป็นสัดส่วนกับแรงดันอนาล็อกอินพุต ทำให้ค่าดิจิตอลที่เอาต์พุตของ n-bit counter เปลี่ยนแปลงตามแรงดันอินพุต ช่วงเวลาในการแปลงสัญญาณจะขึ้นอยู่กับขนาดของแรงดันอินพุต โดยปกติแล้วสัญญาณ Ramp จะได้จากวงจร RC network การเปลี่ยนแปลงของอุณหภูมิโดยรอบอาจส่งผลให้ค่าของเวลาเปลี่ยนไปทำให้เกิด Error ดังนั้นวิธีนี้จึงไม่เป็นที่นิยมนัก แต่ผลของอุณหภูมิที่วุ่นสามารถแก้ไขได้โดยใช้วิธี Dual-slope

Dual-slope (Dual-ramp) A/D converter

วิธีนี้จะช่วยแก้ปัญหาเรื่องความไวต่ออุณหภูมิของ Single-slope ได้ โดยทำให้อยู่ในรูปสัดส่วนของเวลา ดังนั้นแรงดันอินพุตก็จะไม่ขึ้นกับ n-bit counter โดยตรง โคะแกรมของวงจร Dual-Slope A/D converter แสดงดังรูปที่ 3.2



รูปที่ 3.2 โคะแกรมวงจร dual-slope A/D converter

ในช่วงแรกแรงดันอินพุตจะถูกซาร์ทเข้าไปในคาปาซิเตอร์ หรือวงจร Integrator นั้นเอง ใช้เวลาเท่ากับ t_1 ช่วงเวลา t_1 นี้ n-bit counter จะนับจากศูนย์จนถึงค่าสูงสุด ต่อจากนั้น Control logic จะสวิตช์อินพุตของ Integrator จากแรงดันอินพุตมายังแรงดันอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$(-V_{REF})$ ซึ่งมีค่าเป็นลบ เพื่อให้คาปาซิเตอร์คายประจุได้ ในขณะที่คาปาซิเตอร์เริ่มคายประจุนี้ n-bit counter ก็จะเริ่มทำการนับใหม่ โดยเริ่มจากศูนย์อีก จนกระทั่งเอาท์พุทของ Integrator มีค่าเป็นศูนย์ Comparator ก็จะสั่งให้ n-bit counter หยุดนับ ช่วงเวลาคายประจุกำหนดให้เป็น t_2 ดังนั้นอินทิกรัลของแรงดันทั้ง 2 ช่วงเวลาจึงมีค่าเท่ากัน

$$\frac{1}{RC} \int_0^{t_1} V_U dt = \frac{1}{RC} \int_{t_1}^{t_1+t_2} V_{REF} dt$$

ค่าเฉลี่ยของ V_U จะเป็นสัดส่วนกับ t_2 และ t_1

$$V_U = \left(\frac{t_2}{t_1} \right) V_{REF}$$

ถ้า n-bit counter ใช้ความถี่ f_{CLK}

$$t_1 = \frac{2^n}{f_{CLK}}$$

เช่นเดียวกันถ้าให้ดิจิตอลเอาท์พุทที่นับได้ในช่วงเวลา t_2 มีค่าเท่ากับ x

$$t_2 = \frac{x}{f_{CLK}}$$

แทนค่า t_1 และ t_2

$$V_U = \left(\frac{x}{f_{CLK}} \right) / \left(\frac{2^n}{f_{CLK}} \right) V_{REF}$$

$$V_U = \frac{x}{2^n} V_{REF}$$

จะพบว่า ค่า R, C และ f_{CLK} ไม่เกี่ยวข้องกับสมการแต่อย่างใด นั่นแสดงว่าผลของอุณหภูมิที่มีต่อ R และ C นั้น จะไม่กระทบกระเทือนกับวงจร Dual-slope นี้ ระยะเวลาที่ใช้ในการแปลงสัญญาณจะมีค่าเท่ากับ $t_1 + t_2$ หรือ $(2^n + x)/f_{CLK}$ เนื่องจาก x_{max} มีค่าเท่ากับ 2^n เพราะฉะนั้นระยะเวลาในการแปลงสัญญาณสูงสุดเท่ากับ

$$t_{max} = \frac{(2^n + 2^n)}{f_{CLK}} = \frac{2^{n+1}}{f_{CLK}}$$

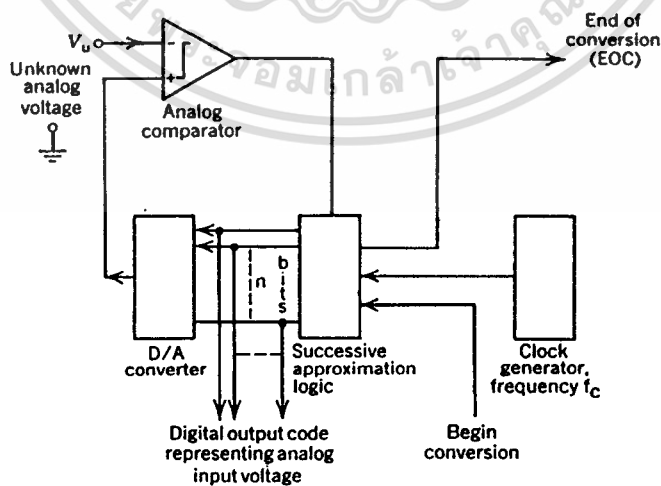
ตัวอย่างเช่น 10 บิต dual-slope A/D converter ใช้ความถี่ 2 MHz จะใช้เวลาในการแปลงสัญญาณสูงสุดเท่ากับ

$$\begin{aligned}
 t_{\max} &= \frac{2^{10+1}}{2 \times 10^6} \\
 &= 1.024 \text{ ms} \\
 &= 976.6 \text{ ครั้ง/วินาที}
 \end{aligned}$$

Dual-slope integrating A/D converter จะให้ค่าของความแม่นยำในการแปลงสัญญาณสูง ทั้งยังมีราคาถูกด้วย ประกอบกับการแปลงสัญญาณไม่มีผลกระทบจากอุณหภูมิแวดล้อม จึงได้นำไปใช้เป็นหัวใจหลักของดิจิตอลโวลท์มิเตอร์เป็นส่วนใหญ่ ข้อเสียของเอชทูตีประเภทนี้คือระยะเวลาที่ใช้ในการแปลงสัญญาณมีค่าสูงทำให้มีอัตราการสุ่มสัญญาณต่ำ ไม่เหมาะกับการนำไปใช้ในวงจร Data Acquisition Unit

3.1.2 Successive approximation A/D convertets

เป็นวิธีการซึ่งใช้เวลาในการแปลงสัญญาณคงที่ เวลาในการแปลงสัญญาณไม่ขึ้นอยู่กับแรงดันอินพุต หัวใจสำคัญของการทำงานอยู่ที่ชุด Successive approximation register (SAR) หรือ Successive approximation logic (SAL) ชุด SAR นี้ จะทำหน้าที่กำหนดค่าแรงดันอินพุตเป็นดิจิตอล แล้วส่งค่าให้ D/A converter เพื่อแปลงเป็นสัญญาณกลับเป็นอนาลอกนำไปเปรียบเทียบกับแรงดันอินพุต แล้วนำผลการเปรียบเทียบที่ได้ส่งกลับมาให้ SAR เพื่อกำหนดค่าดิจิตอลใหม่ต่อไป ดังรูปที่ 3.3



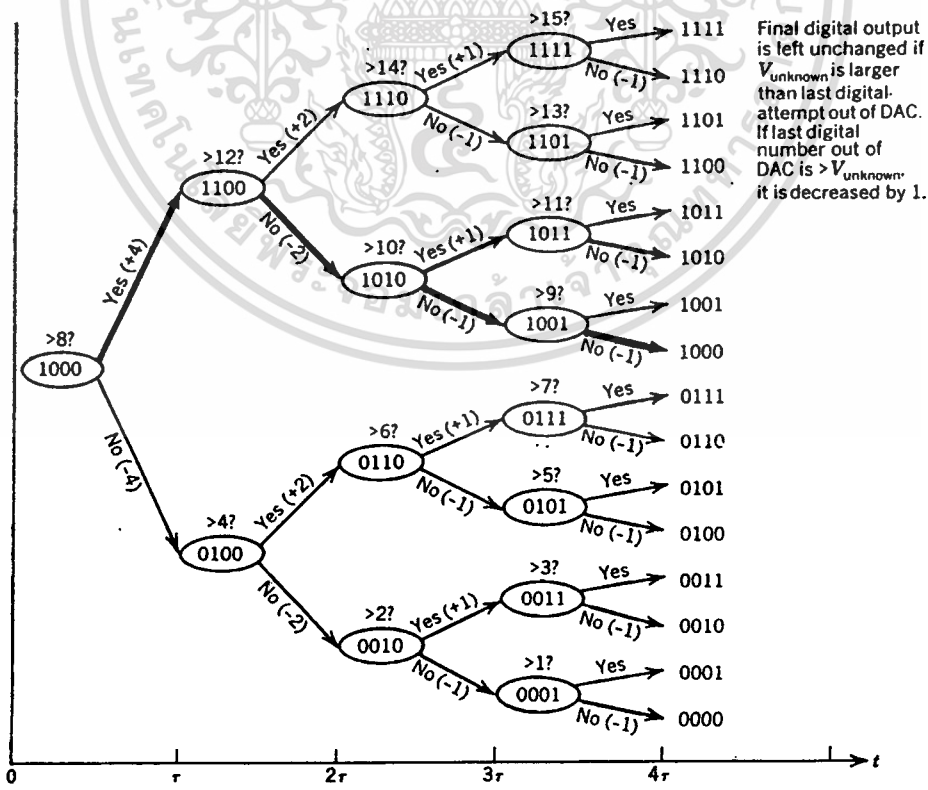
รูปที่ 3.3 โค้ดแอมวจร successive approximation A/D converter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดค่าแรงดันอินพุตนี้ จะเป็นการกำหนดค่าที่มีหลักการแน่นอน มีจำนวนครั้งของการกำหนดค่าต่อการแปลงสัญญาณ 1 ครั้ง เท่านั้น หลักในการกำหนดค่าของ SAR พอจะสรุปได้ดังรูปที่ 3.4 กล่าวคือ SAR จะทำการกำหนดค่าดิจิตอลเอาต์พุตให้มีลอจิกเป็น "1" ทีละบิต จนครบ n บิต โดยเริ่มจากบิต MSB(Most Significant Bit) ก่อนเพื่อนำไปเปรียบเทียบกับแรงดันอินพุตโดยผ่าน D/A Converter ถ้าแรงดันอินพุตมีค่ามากกว่า SAR จะเซ็ทบิตนั้นเป็น "1" เหมือนเดิม แต่ถ้ามีค่าน้อยกว่า SAR จะรีเซ็ทบิตนั้นเป็น "0" จากนั้นจะเซ็ทบิตถัดมา (บิตที่ n-1) ให้เป็น "1" แล้วแปลงเป็นค่าอนาลอกโดยผ่าน D/A converter เพื่อนำไปเปรียบเทียบกับค่าที่ได้อีกครั้งหนึ่งของแรงดันสูงสุด ($V_{FS}/2$) กรณีที่เอาต์พุตมีขนาด 4 บิต ค่าที่ได้ก็จะเท่ากับ "1000" ดังนั้น D/A converter ก็จะแปลงค่าแรงดันได้ $V_{FS}/2$ จากรูปที่ 3.5 ค่าดิจิตอล "1000" คือค่าอนาลอกตั้งแต่ $\frac{V_{FS}}{2} - \frac{V_{FS}}{2^{n+1}}$ ถึง $\frac{V_{FS}}{2} + \frac{V_{FS}}{2^{n+1}}$

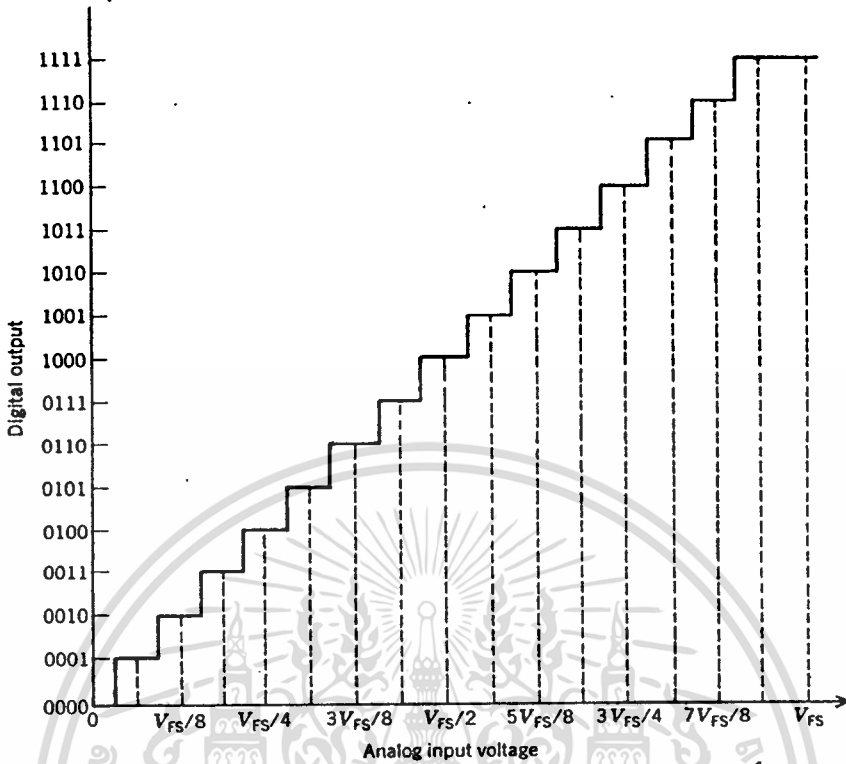
$$\frac{V_{FS}}{2^{n+1}} = \frac{1}{2} \times \frac{V_{FS}}{2^n} = \frac{1}{2} \text{ LSB}$$

n คือ จำนวนบิตของ A/D Converter
 V_{FS} คือ แรงดันอินพุตสูงสุดของ A/D Converter



รูปที่ 3.4 หลักการทำงานของ successive approximation A/D ขนาด 4 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ค่าดิจิตอลเอาต์พุตกับค่าแรงดันอนาล็อกอินพุต

ดังนั้นจึงมีการปรับค่า Offset ของ D/A Converter ให้มีค่าเท่ากับ $-\frac{V_{FS}}{2^{n+1}}$ หรือ $-\frac{1}{2}LSB$ เพื่อให้การเปรียบเทียบกับค่าอนาล็อกอินพุตอยู่ที่ระดับต่ำสุดของค่าดิจิตอล เพราะฉะนั้นที่ค่า

"1000" D/A Converter จะให้ Output เท่ากับ $\frac{V_{FS}}{2} - \frac{V_{FS}}{2^{n+1}}$ เป็นแรงดันเปรียบเทียบ นำไปเปรียบเทียบกับแรงดันอินพุต โดยมี Analog comparator ทำหน้าที่เปรียบเทียบแรงดันทั้ง 2 นี้ การเปรียบเทียบจะทำต่อเนื่องกันทุกๆ สัญญาณนาฬิกา เพราะฉะนั้นในการแปลงสัญญาณแต่ละครั้งของ Successive approximation A/D จะต้องใช้สัญญาณนาฬิกา n ลูก

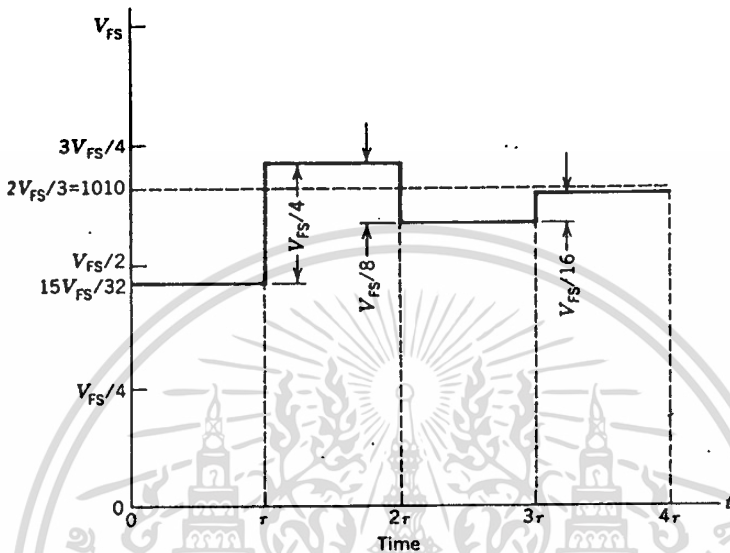
หรือใช้เวลาเท่ากับ $\frac{n}{f_{CLK}}$ วินาที เมื่อ f_{CLK} คือความถี่ของสัญญาณนาฬิกาที่ใช้กับเอชทุติเช่นเอชทุติขนาด 12 บิต มีหลักการทำงานเป็นแบบ Successive approximation ใช้สัญญาณนาฬิกาความถี่ 500 kHz เวลาที่ใช้ในการแปลงสัญญาณแต่ละครั้งมีค่าเท่ากับ $\frac{12}{500kHz}$ วินาที หรือ

24 μs Successive approximation A/D นี้ มีความเร็วในการแปลงสัญญาณสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กว่า Dual-slope A/D เป็นจำนวนมาก จึงเหมาะที่จะนำไปใช้กับวงจร Data Acquisition ให้กับระบบคอมพิวเตอร์ รูปที่ 3.6 เป็นไดอะแกรมตัวอย่างในการแปลงสัญญาณอนาลอกซึ่งมีแรงดันอินพุตเท่ากับ $\frac{2}{3}V_{FS}$ ให้เป็นสัญญาณดิจิทัลขนาด 4 บิต

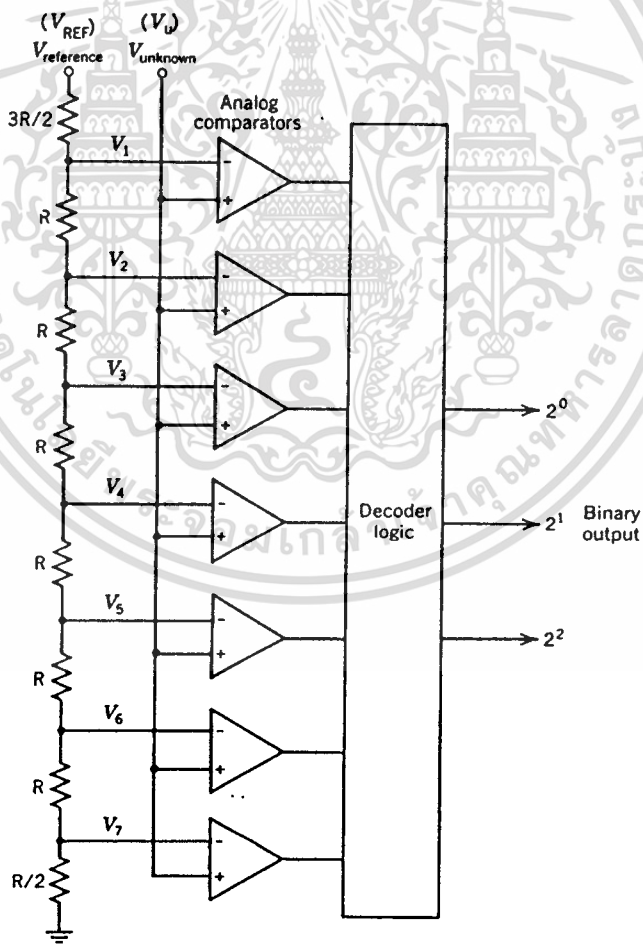


รูปที่ 3.6 ตัวอย่างการแปลงสัญญาณของ 4-bit successive approximation A/D

- ช่วงแรก SAR จะกำหนดค่าดิจิทัล "1000" เป็นแรงดันเปรียบเทียบ มีค่าอนาลอกเท่ากับ $15V_{FS}/32$ แรงดันอินพุตมีค่ามากกว่าแรงดันเปรียบเทียบ
- ช่วงสอง SAR กำหนดค่าดิจิทัล "1100" (เพราะแรงดันอินพุตมีค่ามากกว่าแรงดันเปรียบเทียบ) มีค่าอนาลอกเท่ากับ $23V_{FS}/32$ แรงดันอินพุตมีค่าน้อยกว่าแรงดันเปรียบเทียบ
- ช่วงสาม SAR กำหนดค่าดิจิทัล "1010" (เพราะแรงดันอินพุตมีค่าน้อยกว่าแรงดันเปรียบเทียบ) มีค่าอนาลอกเท่ากับ $19V_{FS}/32$ แรงดันอินพุตมีค่ามากกว่าแรงดันเปรียบเทียบ
- ช่วงสุดท้าย SAR กำหนดค่าดิจิทัล "1011" (เพราะแรงดันอินพุตมีค่ามากกว่าแรงดันเปรียบเทียบ) มีค่าอนาลอกเท่ากับ $21V_{FS}/32$ แรงดันอินพุตมีค่ามากกว่าแรงดันเปรียบเทียบ ดังนั้นค่าดิจิทัลที่ได้ จึงเท่ากับ "1011"

3.1.3 Flash converter

ถึงแม้ว่าวิธีการแบบ Successive approximation จะทำงานได้รวดเร็วกว่า Dual-Slope A/D แต่ก็ยังมีวิธีอีกวิธีหนึ่งที่สามารถทำงานได้รวดเร็วกว่านี้อีก นั่นคือวิธีการของ Flash converter หรือ Parallel converter ความเร็วในการแปลงสัญญาณแต่ละครั้งขึ้นอยู่กับเวลาการทำงานของวงจร Decoder logic หลักการของวงจรเป็นลักษณะของการเปรียบเทียบระดับแรงดันอนาล็อกอินพุตตามขนาดของ Binary weights ตามรูปที่ 3.7 และตารางที่ 1 แรงดันอ้างอิง $V_7 = (1/16)V_{REF}$, $V_6 = (3/16)V_{REF}$,..... $V_1 = (13/16)V_{REF}$ แรงดันอนาล็อกอินพุตจะถูกเปรียบเทียบกับแรงดันอ้างอิงทั้ง 7 ระดับพร้อมๆ กัน โดยชุดเปรียบเทียบทั้ง 7 ชุด ถ้าแรงดันอินพุตมีค่ามากกว่าแรงดันอ้างอิงเอาต์พุตของชุดเปรียบเทียบจะมีค่าเป็น "1"



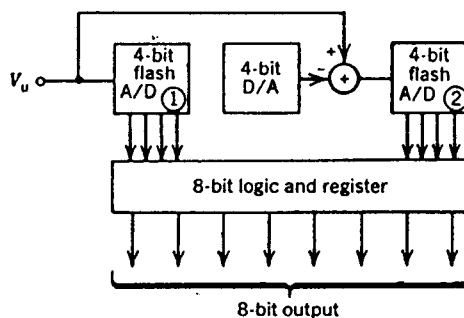
รูปที่ 3.7 เทอติขขนาด 3 บิต แบบ flash converter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V_U	Decoder Logic	
	Input	Output
$< V_{REF}/16$	0000000	000
$< 3V_{REF}/16$	0000001	001
$< 5V_{REF}/16$	0000011	010
$< 7V_{REF}/16$	0000111	011
$< 9V_{REF}/16$	0001111	100
$< 11V_{REF}/16$	0011111	101
$< 13V_{REF}/16$	0111111	110
$\geq 13V_{REF}/16$	1111111	111

ตารางที่ 3.1 เปรียบเทียบแรงดันอินพุตกับกับเอาต์พุตของ Decoder

เอาต์พุตของชุดเปรียบเทียบทั้งหมดจะต้องเข้ากับวงจร Decoder เพื่อแปลงรหัสเป็นค่าดิจิตอลดังตารางที่ 3.1 ชุดเปรียบเทียบจะมีจำนวน $2^n - 1$ ชุด สำหรับเอาต์พุตขนาด n บิต ตามตัวอย่างเป็นเอาต์พุตขนาด 3 บิต มีชุดเปรียบเทียบเท่ากับ $2^3 - 1 = 7$ ชุด ถ้าขนาด 8 บิต จะต้องมีชุดเปรียบเทียบจำนวน 255 ชุด ข้อเสียของ Flash A/D converter คือ จำนวนบิตสูงๆ จะต้องใช้ชุดเปรียบเทียบจำนวนมาก แต่สำหรับข้อดีคือ สามารถแปลงสัญญาณได้ด้วยความเร็วสูงถึง 108 ครั้งต่อวินาที วิธีการแก้ปัญหาจำนวนชุดเปรียบเทียบมากๆ วิธีหนึ่งก็คือ ใช้ Flash A/D converter ขนาด 4 บิต จำนวน 2 ตัว ต่อร่วมกันเป็น Flash A/D converter ขนาด 8 บิต



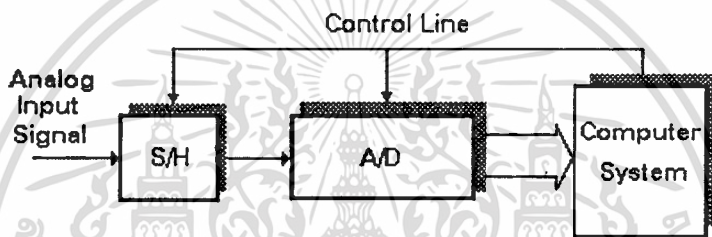
รูปที่ 3.9 เอาต์พุต 8 บิต แบบ flash converter 4 บิต 2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของวิทยานิพนธ์ฉบับนี้จะเป็นการนำ Successive Approximation A/D Converter มาใช้ในการออกแบบวงจรวัดสัญญาณไฟฟ้า โดยใช้เทคนิคลดเวลาการแปลงสัญญาณให้เร็วขึ้นกว่าเดิมประมาณ 2 เท่า

3.2 ความเร็วในการแปลงสัญญาณ

อัตราการสุ่มสัญญาณแต่ละครั้งสามารถคำนวณได้จากส่วนกลับของเวลาทั้งหมดที่ใช้ในการแปลงสัญญาณต่อหนึ่งครั้ง องค์ประกอบหลักของค่านฮาร์ดแวร์ที่ใช้ในการแปลงสัญญาณอนาลอกเป็นดิจิทัลให้กับคอมพิวเตอร์แสดงดังรูปที่ 3.10



รูปที่ 3.10 บล็อกโคะแกรมการแปลงสัญญาณอนาลอกเป็นดิจิทัล

ขั้นตอนในการสุ่มสัญญาณแต่ละครั้งจะประกอบด้วย

1. คอมพิวเตอร์จะส่งคำสั่งให้กับวงจร Sample & Hold เพื่อทำหน้าที่คงค่าสัญญาณให้กับเอทูดิและส่งคำสั่งเริ่มต้นแปลงสัญญาณให้กับเอทูดิ
2. เมื่อเอทูดิแปลงสัญญาณเรียบร้อยแล้ว จะส่งสัญญาณไปขัดจังหวะคอมพิวเตอร์เพื่อแจ้งให้รู้ว่ามิข้อมูลใหม่มาแล้ว
3. คอมพิวเตอร์ทำการอ่านและเก็บข้อมูลที่ได้จากเอทูดิไว้ในหน่วยความจำ

ในขั้นตอนที่ 2 ถ้าใช้วิธีอินเทอร์รัพท์ ผู้เขียนโปรแกรมต้องเขียนโปรแกรมใน Interrupt Service Routine ให้คอมพิวเตอร์อ่านข้อมูลจากเอทูดิ แต่ถ้าใช้แบบหน่วงเวลาที่ไม่ต้องใช้สัญญาณอินเทอร์รัพท์ ผู้เขียนโปรแกรมต้องเขียนโปรแกรมหน่วงเวลาเพื่อรอให้เอทูดิแปลงสัญญาณเสร็จเรียบร้อยแล้วเสียก่อน จึงจะไปอ่านข้อมูลจากเอทูดิได้ มิฉะนั้นแล้วข้อมูลที่อ่านได้จะผิดพลาดจากความเป็นจริง จากขั้นตอนในการสุ่มสัญญาณ จะเห็นว่าเวลาทั้งหมดที่ใช้ในการแปลงสัญญาณหนึ่งครั้งนั้นจะประกอบด้วย เวลาที่วงจร Sample & Hold

คงค่าสัญญาณอินพุต แบ่งเป็นช่วง Acquisition Time กับช่วง Settling Time เวลาที่ใช้ในการแปลงสัญญาณของวงจรเอชดี และเวลาที่คอมพิวเตอร์ทำคำสั่ง OUT เพื่อควบคุมวงจร Sample & Hold และเอชดีให้ทำคำสั่ง IN เพื่ออ่านข้อมูลแล้วเก็บค่าในหน่วยความจำ ตลอดจนคำสั่งอื่นๆที่เกี่ยวข้องด้วย

สมมุติให้ค่าเวลาต่างๆ ที่ใช้ในการแปลงสัญญาณมีค่าดังต่อไปนี้

S/H Acquisition Time	6 μ s
S/H Output Settling Time	1 μ s
A/D Converter Time	110 μ s
คำสั่ง IN,OUT และคำสั่งอื่นๆที่เกี่ยวข้อง	40 μ s

เวลาทั้งหมดที่ใช้ในการแปลงสัญญาณต่อหนึ่งครั้งคือ $6+1+110+40 \mu\text{s} = 157 \mu\text{s}$ เพราะฉะนั้นอัตราการสุ่มสัญญาณสูงที่สุดที่สามารถทำได้คือ

$$\frac{1}{157} \mu\text{s} = \frac{6369 \text{ samples}}{\text{sec}}$$

จะเห็นว่าอัตราการสุ่มสัญญาณนั้นขึ้นอยู่กับเวลาในการแปลงสัญญาณของเอชดี และความเร็วในการทำงานของคอมพิวเตอรืเป็นส่วนใหญ่ ดังนั้นการที่จะเพิ่มอัตราการสุ่มสัญญาณ จึงสามารถกระทำได้ 2 วิธีด้วยกันคือ ลดค่าเวลาในการแปลงสัญญาณของเอชดี ซึ่งเป็นการเพิ่มความเร็วในการแปลงสัญญาณนั่นเอง วิธีที่สองคือ เลือกคอมพิวเตอรืที่มีความเร็วในการทำงานสูง วิทยานิพนธ์ฉบับนี้ได้กล่าวถึงวิธีแรกเท่านั้น โดยเลือกใช้ไอซีเอชดีแบบ Successive Approximation Converter ขนาด 8 บิต เบอร์ AD7574 มีค่า Conversion Time ต่ำสุด 15 μ s. จากหัวข้อ 3.1 ค่า Conversion Time ของเอชดีประเภทนี้ขึ้นอยู่กับคาบเวลาของสัญญาณนาฬิกาที่ใช้ และจำนวนบิตของเอชดี

$$T_c = n \times \frac{1}{f_{\text{CLK}}}$$

ดังนั้นความถี่ของสัญญาณนาฬิกาสูงที่สุดที่ใช้กับเอชดีขนาด 8 บิต ความเร็ว 15 μ s จะมีค่าสูงสุดไม่เกิน 533.33 kHz

3.3 ค่าผิดพลาดที่เกิดขึ้นจากการแปลงสัญญาณ

เอาต์พุตขนาด n บิต สามารถแทนค่าแรงดันอนาล็อกอินพุตได้ตั้งแต่ 0 ถึงค่าสูงสุด $2^n - 1$ ระดับ ในหนึ่งระดับจะมีค่าเท่ากับแรงดันสูงสุดหารด้วยจำนวนระดับทั้งหมด คือ 2^n ค่า 1 ระดับนี้ถือว่าเป็นค่าความละเอียดของเอาต์พุต เรียกว่า Resolution หรือ ค่า 1 LSB (Least Significant Bit)

$$\text{ค่า 1 LSB} = \frac{V_{FS}}{2^n}$$

V_{FS} = ค่าแรงดันอนาล็อกอินพุตสูงสุด

n = จำนวนบิตของเอาต์พุต

ตัวอย่างเช่นเอาต์พุตขนาด 4 บิต สามารถแทนค่าแรงดันอนาล็อกอินพุตที่มีค่าตั้งแต่ 0 V ถึง 1 V ได้ทั้งสิ้น 16 ระดับ โดยเริ่มตั้งแต่ 0000 ถึง 1111

ค่า 1 LSB มีค่าเท่ากับ $\frac{1}{16}$ V ความสัมพันธ์ระหว่างค่าแรงดันอนาล็อกอินพุตกับค่าดิจิทัลเอาต์พุตเป็นไปตามสมการ

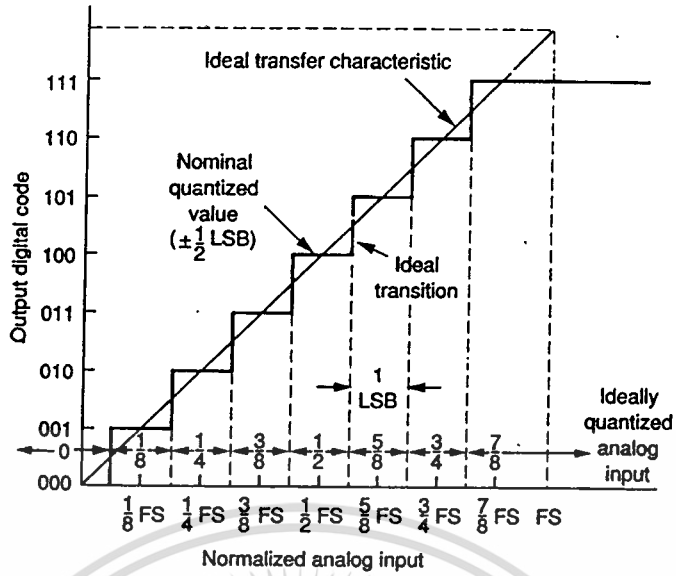
$$V_A = V_D \times 1 \text{ LSB}$$

V_A = แรงดันอนาล็อกอินพุต

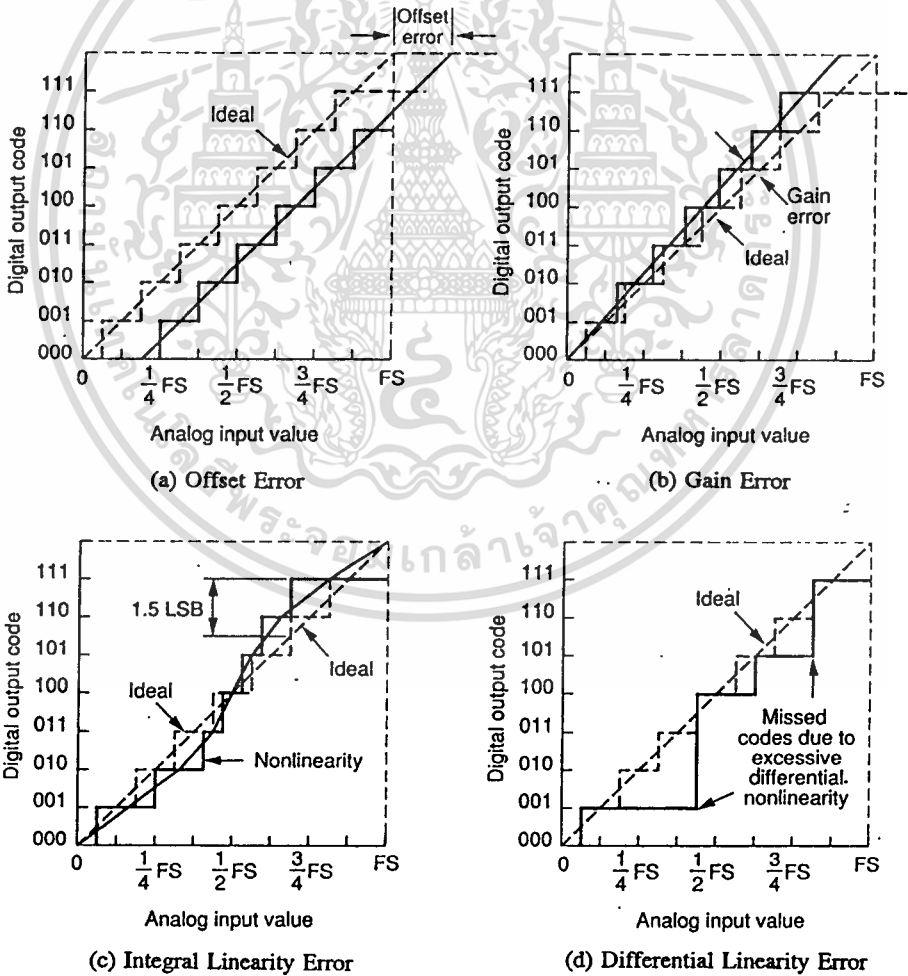
V_D = ค่าดิจิทัลเอาต์พุต (คิดเป็นเลขฐานสิบ)

จากตัวอย่างเอาต์พุตขนาด 3 บิต สามารถแสดงความสัมพันธ์ระหว่างค่าแรงดันอนาล็อกอินพุตกับค่าดิจิทัลเอาต์พุตได้ดังรูปที่ 3.11

ค่าความแตกต่างระหว่างระดับต่างๆ ซึ่งเมื่อแรงดันอนาล็อกอินพุตมีการเปลี่ยนแปลงในขณะที่ค่าดิจิทัลเอาต์พุตมีค่าคงที่ เรียกว่าช่วง quantization หรือ Q แสดงดังรูปที่ 3.11 ช่วง quantization นี้มีค่าเท่ากับ 1 LSB ดังนั้นเอาต์พุตในทางอุดมคติจะมีค่าผิดพลาดที่เกิดขึ้นจาก quantization เท่ากับ $\pm \frac{1}{2}$ LSB เท่านั้น ในทางปฏิบัติค่าผิดพลาดที่เกิดขึ้นกับไอซีเอาต์พุตประกอบด้วย Offset Error, Gain Error และ Linearity Error แสดงดังรูปที่ 3.12



รูปที่ 3.11 แรงคั่นอนาลอกอินพุตกับดิจิตอลเอาต์พุตขนาด 3 บิต



รูปที่ 3.12 ค่าผิดพลาดที่เกิดขึ้นเหตุใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.12 (a) และ (b) คือ Offset Error และ Gain Error ตามลำดับ ซึ่งยังคงให้ค่า quantization เท่ากับ $\pm \frac{1}{2}$ LSB เท่าเดิม สำหรับ Linearity Error นั้นจะมีผลให้ค่า quantization

มีค่ามากกว่า $\pm \frac{1}{2}$ LBS แบ่งเป็น 2 ประเภทด้วยกันคือ Integral Linearity Error เป็นค่าผิดพลาดที่เกิดจากการแปลงสัญญาณกำหนดจากค่าเบี่ยงเบนสูงสุดจากค่าจริงหรือจากเส้นตรงที่ลากตั้งแต่จุด 0 ถึง FS ในรูปที่ 3.12 (C) ค่า Integral Linearity Error มีค่าสูงสุดในช่วง 110 ถึง 111 วัดได้ 1.5 LSB Linearity Error ประเภทที่ 2 คือ Differential Linearity Error เป็นค่าผิดพลาดที่เกิดจากการแปลงสัญญาณ โดยกำหนดจากค่าเบี่ยงเบนสูงสุดของอนาล็อก อินพุท ในช่วงที่ดิจิตอลเอาต์พุทมีค่าคงที่ ในรูปที่ 3.12 (d) ค่า Differential Linearity Error มีค่าสูงสุดที่ 001 มีค่า 2 LSB ในกรณีที่ค่า Differential Linearity Error มีค่าเท่ากับหรือมากกว่า 1 LSB จะทำให้เกิด Missing Code คือค่าดิจิตอลเอาต์พุทบางค่าจะหายไป ซึ่งในรูปจะเห็นว่าดิจิตอลเอาต์พุทค่า 010 และ 011 จะหายไปเนื่องจาก Differential Linearity Error มีค่า 2 LSB ส่วนที่ตำแหน่ง 100 มีค่า Differential Linearity Error 0.5 LSB ซึ่งไม่ทำให้เกิด Missing Code.

นอกจากค่าผิดพลาดที่กล่าวมาแล้วนั้น ยังมีค่าผิดพลาดที่เกิดจากอัตราการเปลี่ยนแปลงของแรงดันอินพุทในขณะที่ทำการแปลงสัญญาณหรือที่เรียกว่า Slew Rate ในการแปลงสัญญาณของเอชดีนั้น สัญญาณอินพุทจะมีค่าเปลี่ยนแปลงได้ไม่เกินค่า quantization หรือ $\pm \frac{1}{2}$ LSB จึงจะไม่ทำให้เกิดค่าผิดพลาด ค่าผิดพลาดที่เกิดขึ้นนี้เรียกว่า Aperture Error ถ้าสมมุติให้สัญญาณอินพุทของเอชดีเป็นสัญญาณไซน์มีค่า

$$v_i = V_p \sin \omega t$$

ความชันของสัญญาณจะมีค่า

$$\frac{dv_i}{dt} = V_p \omega \cos \omega t$$

เนื่องจากสัญญาณไซน์จะมีความชันสูงสุดที่มุมศูนย์องศา ดังนั้น

$$\left(\frac{dv_i}{dt}\right)_{\max} = V_p \omega = V_p 2\pi f$$

เหตุนี้ Successive Approximation ใช้เวลาในการแปลงสัญญาณ

$$T_c = \frac{n}{f_{\text{CLK}}}$$

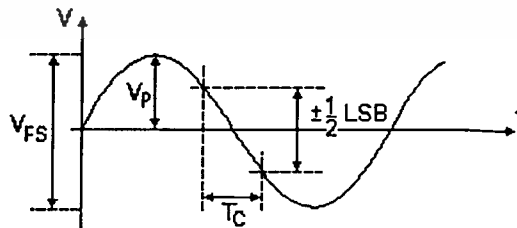
และ $1 \text{ LSB} = \frac{V_{\text{FS}}}{2^n}$

$$\left(\frac{dv_i}{dt}\right)_{\max} \geq \frac{V_{\text{FS}}}{2^n T_c}$$

ถ้าให้แรงดันสูงสุด (V_{FS}) มีค่า $2V_p$ ก็คือค่าแรงดัน peak-to-peak ของสัญญาณซายด์

$$V_p 2\pi f \leq \frac{2V_p}{2^n T_c}$$

$$f_{\max} = \frac{1}{2^n T_c \pi}$$



รูปที่ 3.13 ความสัมพันธ์ระหว่าง T_c กับระดับแรงดันอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าผิดพลาดที่เกิดจาก Aperture Error นี้สามารถแก้ไขได้โดยการต่อวงจร Sample & Hold ไว้ระหว่างสัญญาณอินพุตกับเอาต์พุต เพื่อที่จะทำให้ระดับแรงดันอินพุตที่เอาต์พุตมีค่าคงที่ ในขณะที่เอาต์พุตทำการแปลงสัญญาณ จะเห็นว่าผลของ Aperture Error นั้นเกี่ยวข้องกับความถี่ของสัญญาณอินพุต ถ้าความถี่สูงกว่า f_{max} จะทำให้เอาต์พุตแปลงค่าผิดพลาดไปจากค่าจริง ในเรื่องของความถี่กับค่าผิดพลาดยังมีอีกประการหนึ่งคือ ถ้าความถี่ของสัญญาณอินพุตมีค่ามากเกินไปกว่าครึ่งหนึ่งของความถี่ที่ใช้ในการสุ่มสัญญาณ จะทำให้เกิด Aliasing ขึ้น ผลที่ได้ก็คือเอาต์พุตแปลงสัญญาณได้ค่าระดับแรงดันถูกต้อง แต่ความถี่ที่ได้จะมีค่าต่ำกว่าความถี่ของสัญญาณอินพุต กรณีนี้แก้ไขได้ด้วยการเพิ่มอัตราการสุ่มสัญญาณให้มากขึ้น ตามที่ได้กล่าว มาแล้วในบทที่ 2.



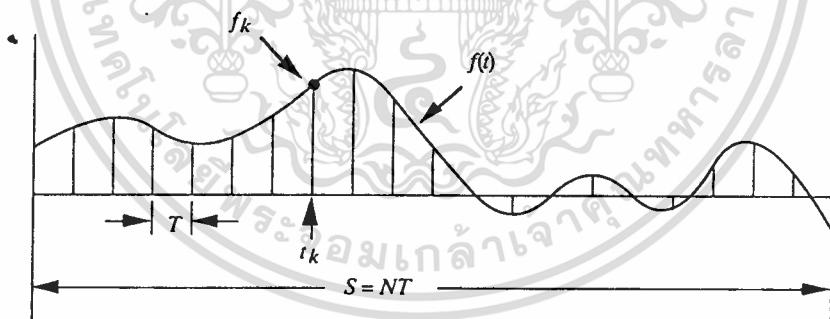
บทที่ 4

หลักการ Time-Interleaving กับการสุ่มสัญญาณแบบสุ่มสม่ำเสมอ

โดยทั่วไป ในการสุ่มสัญญาณไฟฟ้าให้กับคอมพิวเตอร์นั้น จะต้องมีคาบเวลาที่เท่ากัน เพื่อการประมวลผลด้วย Fourier Transform ปกติถ้าใช้ไอซีเอทวูตีเพียงตัวเดียว อัตราการสุ่มสัญญาณก็จะมีคาบเวลาที่เท่ากัน โดยอัตโนมัติอยู่แล้ว แต่สำหรับหลักการของ Time-Interleaving นั้นเป็นการนำไอซีเอทวูตีหลายๆตัวมาต่อขนานกันและวัดสัญญาณไฟฟ้าที่จุดเดียวกัน เป็นเทคนิคในการลดค่า Conversion time ของเอทวูตีนั่นเอง

4.1 การสุ่มสัญญาณในการแปลงสัญญาณจำนวน 1 ตัว

การทรานฟอร์มสัญญาณแบบ Discrete Fourier Transform. หรือ DFT จะมียอดประกอบ Sine และ Cosine เป็นส่วนของรูปคลื่นสัญญาณ ซึ่งรูปคลื่น $f(t)$ จะถูกสุ่มด้วยจำนวน N ครั้ง ในช่วง $t_0 = 0, t_1 = T, t_k = kT, \dots, t_{N-1} = (N-1)T$ การสุ่มสัญญาณในช่วง $S = NT$ แสดงดังรูปที่ 4.1



รูปที่ 4.1 การสุ่มสัญญาณแบบสุ่มสม่ำเสมอ มีคาบเวลา = T

กำหนดให้

$$f_k = f(t_k)$$

ดังนั้น DFT ของ f_k จะถูกกำหนดโดย

$$F_n = \sum_{k=0}^{N-1} f_k e^{-i2\pi nk/N}$$

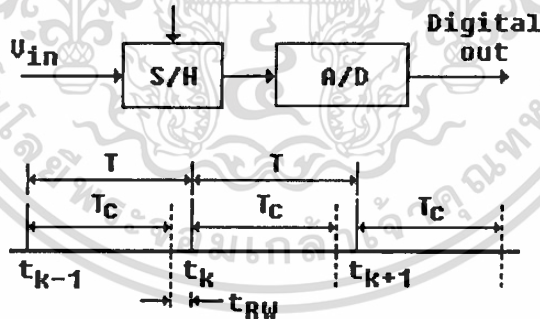
F_n = สัมประสิทธิ์ Fourier ที่ความถี่ n

$n = 0, 1, 2, \dots, N-1$

N = จำนวนการสุ่มสัญญาณทั้งหมด

k = ค่าคงที่ ใด ๆ

จากการคำนวณของ DFT จะเห็นว่าช่วงระยะเวลาในการสุ่มสัญญาณแต่ละครั้งจะต้องมีคาบเวลาที่เท่ากัน (เท่ากับ T) คือการสุ่มสัญญาณแบบสม่ำเสมอ (Uniformly Sampled Signal) ปกติในการวัดสัญญาณอนาลอกให้กับคอมพิวเตอร์ที่ใช้ไอซีเอทีในการแปลงสัญญาณเพียงตัวเดียว จะมีช่วงระยะเวลาในการสุ่มสัญญาณ (T) แต่ละช่วงเท่ากันตลอด เพราะคอมพิวเตอร์จะมีการทำคำสั่งในการรับข้อมูลเป็นแบบวนลูป ช่วงระยะเวลาในการสุ่มสัญญาณจะขึ้นอยู่กับค่า Conversion Time (T_C) รวมกับระยะเวลาที่คอมพิวเตอร์ทำคำสั่งรับข้อมูลและคำสั่งที่สั่งให้เอทีซีเริ่มต้นแปลงสัญญาณใหม่อีกครั้งหนึ่ง ช่วงเวลาดังกล่าวแสดงดังรูปที่ 4.2



รูปที่ 4.2 ช่วงเวลาในการสุ่มสัญญาณของเอทีซีตัวเดียว

กำหนดให้

t_k = ช่วงเวลาในการเริ่มต้นแปลงสัญญาณของเอทีซี

$t_k + T_C$ = ช่วงเวลาที่เอทีซีทำการแปลงสัญญาณเสร็จเรียบร้อยแล้ว

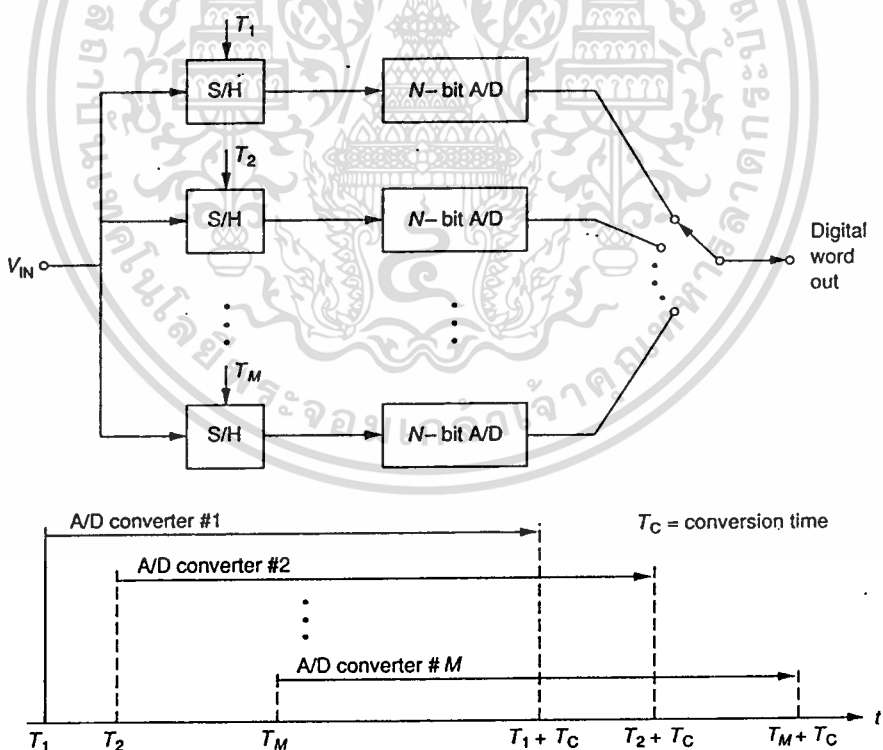
k = ลำดับของข้อมูลที่ได้จากเอทีซี มีค่า $0, 1, 2, 3, \dots, N-1$

T = ช่วงเวลาระหว่าง $t_{k+1} - t_k$

ช่วงเวลาระหว่าง T_{RW} คือช่วงเวลาที่คอมพิวเตอร์ทำคำสั่งรับข้อมูลและคำสั่งให้เอาต์พุตเริ่มต้นแปลงสัญญาณ จากรูปที่ 4.2 จะเห็นว่าช่วงเริ่มต้นในการสุ่มสัญญาณแต่ละครั้งจะมีคาบเวลา T เท่ากันตลอด ($t_k - t_{k-1} = t_{k+1} - t_k$) ดังนั้นข้อมูลที่รับจากเอาต์พุตก็คือข้อมูล \mathcal{M} ที่เวลาเริ่มต้นสุ่มสัญญาณ ซึ่งทำให้ข้อมูลที่รับแต่ละค่ามีคาบเวลาเท่ากันด้วย แสดงว่าในการใช้เอาต์พุตจำนวน 1 ตัว วัดสัญญาณให้กับคอมพิวเตอร์ จะเป็นการสุ่มสัญญาณแบบสม่ำเสมอ นั่นเอง ข้อมูลที่ได้รับสามารถนำไปประมวลผลสัญญาณตามหลักการวิเคราะห์แบบ Fourier Transform ทั่วๆ ไปได้เลย

4.2 การใช้เอาต์พุต ในการแปลงสัญญาณจำนวนมากกว่า 1 ตัว

จากการใช้เอาต์พุตเพียงตัวเดียวในการแปลงสัญญาณในหัวข้อ 4.1 นั้น อัตราการสุ่มสัญญาณขึ้นอยู่กับค่า T_C บวกกับ T_{RW} ซึ่ง T_{RW} ขึ้นอยู่กับความเร็วของคอมพิวเตอร์ ส่วน



รูปที่ 4.3 การแปลงสัญญาณแบบ Time Interleaving

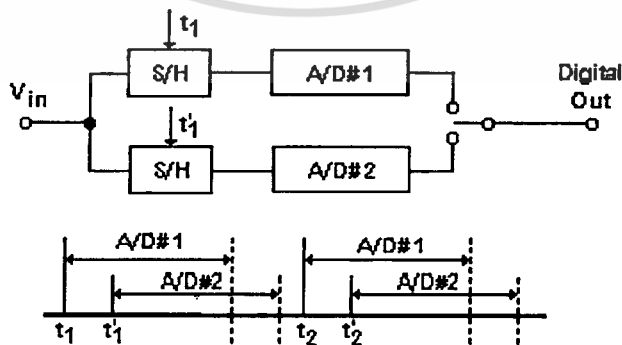
T_C ขึ้นอยู่กับความเร็วในการแปลงสัญญาณของไอซีเอพูดี ในช่วงเวลา T_C ถ้ามีการนำไอซีเอพูดีอีกตัวหนึ่งมาทำการแปลงสัญญาณ ก็จะทำให้ได้ข้อมูลจากการสุ่มสัญญาณเพิ่มขึ้นโดยไม่มีผลกระทบใดๆ กับข้อมูลเดิมทั้งสิ้น ในการใช้ไอซีเอพูดีจำนวนมากกว่า 1 ตัว มาต่อขนานกันเพื่อวัดสัญญาณที่จุดอินพุตเดียวกัน แต่ให้ทำการเริ่มต้นแปลงสัญญาณไม่พร้อมกันหรือเหลื่อมล้ำกัน เรียกวิธีการนี้ว่า " Time Interleaving "[1] ลักษณะการต่อวงจรแสดงดังรูปที่ 4.3 จากรูปเป็นการใช้ไอซีเอพูดี Successive Approximation ขนาด N bit จำนวน M ตัว ต่อขนานกันเพื่อแปลงสัญญาณอนาลอก (V_{in}) ที่จุดเดียวกัน เนื่องจากการแปลงสัญญาณของเอพูดีแบบ Successive Approximation นี้ จะใช้เวลาในการแปลงสัญญาณเท่ากับจำนวนบิต คูณด้วย 1 คาบเวลาของสัญญาณนาฬิกาที่ใช้กับเอพูดี

$$T_C = N \times \frac{1}{f_{CLK}}$$

N คือ จำนวนบิตของเอพูดี

f_{CLK} คือ ความถี่ของสัญญาณนาฬิกาที่ใช้กับเอพูดี

ถ้าให้เอพูดีแต่ละตัวเริ่มทำงานที่ช่วงเวลาต่างกัน 1 คาบสัญญาณนาฬิกา ดังนั้นจำนวนเอพูดีสูงสุดที่สามารถนำมาต่อขนานกัน จะมีค่าเท่ากับ N ตัว ($M=N$) ทำให้ได้ข้อมูลเพิ่มขึ้น M-1 ค่า นั่นก็คืออัตราการสุ่มสัญญาณเพิ่มขึ้น M เท่า วิทยานิพนธ์ฉบับนี้ได้ทำการทดลองใช้เอพูดีจำนวน 2 ตัว ต่อขนานกัน ซึ่งจะทำได้อัตราการสุ่มสัญญาณเพิ่มขึ้นเป็น 2 เท่า ลักษณะการต่อวงจรแสดงดังรูปที่ 4.4



รูปที่ 4.4 การใช้เอพูดี 2 ตัว ทำงานแบบ Time-Interleaving

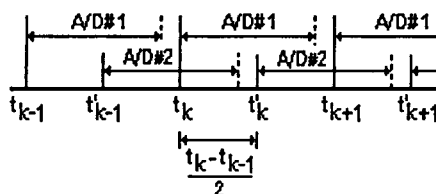
จากรูปจะพบว่าการทำงานแบบ Time-Interleaving นี้ ช่วงเวลาในการสุ่มสัญญาณแต่ละครั้งจะมีค่าไม่เท่ากัน ซึ่งถือว่าเป็นการสุ่มสัญญาณแบบไม่สม่ำเสมอ (Nonuniformly Sampled Signal) กล่าวคือ ช่วงเวลา t_1 ถึง t'_1 และ t'_1 ถึง t_2 มีค่าไม่เท่ากัน ทำให้การประมวลผลสัญญาณแบบ Fourier Transform มีค่าผิดพลาดได้ แต่อาจจะใช้การประมวลผลตามเอกสารอ้างอิง [2] แทนได้ ในกรณีที่ต้องการให้ช่วงเวลาในการสุ่มสัญญาณแบบ Time-Interleaving นี้มีค่าเท่ากันตลอดสามารถทำได้ 2 วิธีด้วยกันคือ

1. ควบคุมที่ซอร์ฟแวร์ โดยควบคุมจากคำสั่งที่ส่งให้เอทูดิแต่ละตัวทำงาน วิธีนี้จะต้องใช้ออสซิลโลสโคปหรือลอจิกอนาไลเซอร์วัดสัญญาณและปรับแต่งที่ซอร์ฟแวร์โดยการเขียนโปรแกรมหน่วงเวลา ซึ่งเป็นวิธีที่ค่อนข้างจะยุ่งยากในทางปฏิบัติ เพราะค่าของเวลาขึ้นอยู่กับการทำงานของไมโครโปรเซสเซอร์หรือตัวโปรแกรมที่ใช้เขียนซอร์ฟแวร์นั้นๆ

2. ควบคุมที่ฮาร์ดแวร์ โดยการออกแบบวงจรควบคุมการทำงานของเอทูดิตัวที่สองให้ได้ช่วงเวลาที่สอดคล้องกับเอทูดิตัวที่หนึ่ง วิธีนี้จะเป็นวิธีการที่นำเสนอในวิทยานิพนธ์ฉบับนี้ ถึงแม้ว่าจะต้องเสียเวลาในการออกแบบวงจรควบคุมก็ตาม แต่ก็เป็นที่สะดวกในทางปฏิบัติ เพราะจะทำให้ผู้เขียนโปรแกรมไม่ต้องคำนึงถึงช่วงเวลาในการสุ่มสัญญาณ โดยวงจรควบคุมจะทำหน้าที่คอยตรวจสอบให้มีช่วงเวลาในการสุ่มเท่ากันอยู่ตลอดเวลา

4.3 การควบคุมเวลาในการสุ่มสัญญาณ

เพื่อที่จะให้การทำงานแบบ Time-Interleaving ที่ใช้เอทูดิจำนวน 2 ตัว ทำการสุ่มสัญญาณแบบสม่ำเสมอ (Uniformly Time-Interleaving) จากวงจรในรูปที่ 4.4 ถ้ากำหนดให้เอทูดิตัวที่ 2 เริ่มทำการแปลงสัญญาณในช่วงกึ่งกลางระหว่างเวลา t_1 กับ t_2 และ t_2 กับ t_3 ตามลำดับ ก็จะทำให้ช่วงระยะเวลาในการสุ่มสัญญาณของเอทูดิตั้งสองตัวมีค่าเท่ากันตลอด เป็นการสุ่มสัญญาณแบบสม่ำเสมอ แสดงผังรูปที่ 4.5 สามารถประมวลผลได้ถูกต้องตามหลักการของ Fourier Transform



รูปที่ 4.5 ช่วงเวลาในการสุ่มสัญญาณแบบสม่ำเสมอของเอทูดิ 2 ตัว

การกำหนดช่วงเวลาเพื่อให้เอาต์พุตตัวที่ 2 เริ่มทำการแปลงสัญญาณตามที่ได้กล่าวมาแล้วนั้น สามารถกำหนดได้ ตามสมการ

$$t'_k = t_k + \left(\frac{t_k - t_{k-1}}{2} \right)$$

t_k คือ เวลาเริ่มต้นแปลงสัญญาณของเอาต์พุตตัวที่ 1

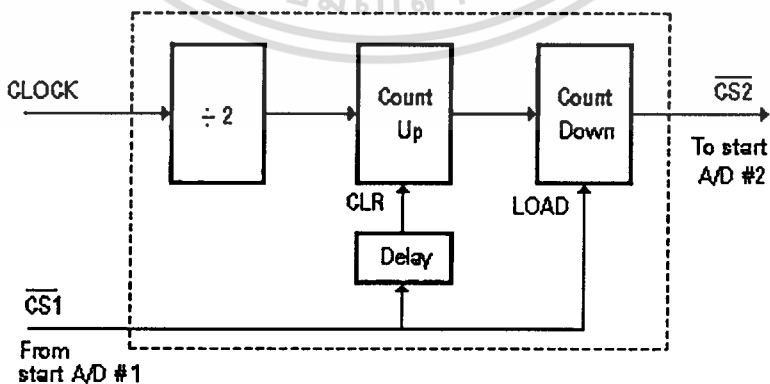
t'_k คือ เวลาเริ่มต้นแปลงสัญญาณของเอาต์พุตตัวที่ 2

k คือ ค่าคงที่มีค่า 0, 1, 2,, N-1

N คือ จำนวนที่ต้องการสุ่มสัญญาณทั้งหมด

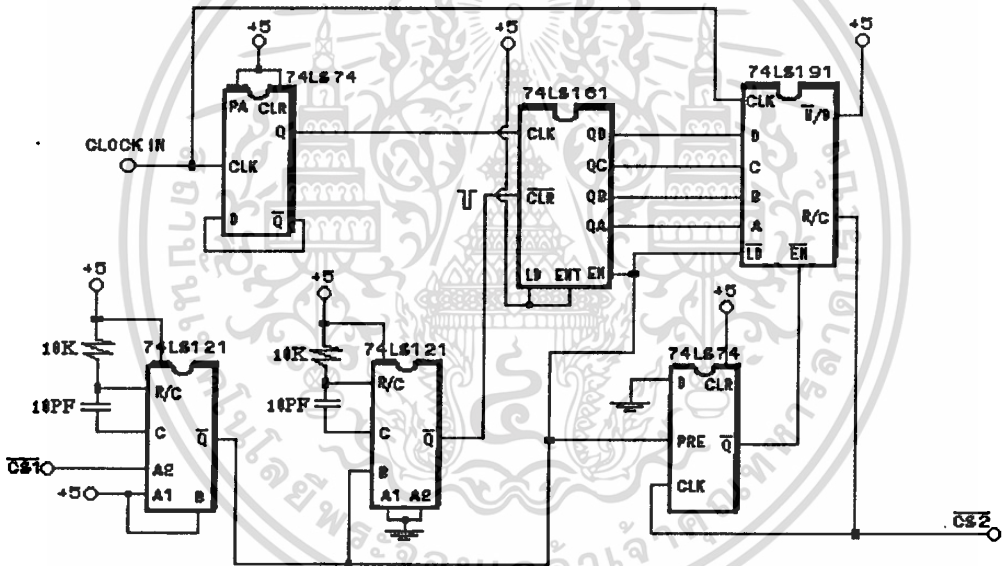
เนื่องจากเอาต์พุตทำงานตามจังหวะของสัญญาณนาฬิกา และสมการที่ใช้กำหนดการเริ่มต้นทำการแปลงสัญญาณของเอาต์พุตตัวที่ 2 เป็นสมการที่เกี่ยวข้องกับเวลา ดังนั้นในการออกแบบวงจรควบคุมจึงใช้สัญญาณนาฬิกาที่ใช้กับเอาต์พุตเป็นตัวนับเวลา เมื่อนับเวลาครบตามสมการที่กำหนดไว้ วงจรควบคุมก็จะส่งพัลส์ไปให้กับเอาต์พุตตัวที่ 2 เป็นการเริ่มแปลงสัญญาณ บล็อกไดอะแกรมแสดงการทำงานของวงจรควบคุม แสดงดังรูปที่ 4.6

จากบล็อกไดอะแกรมรูปที่ 4.6 ขา Clock จะเป็นขาอินพุตมาจกสัญญาณนาฬิกาของเอาต์พุต ขา \overline{CS} $\overline{CS1}$ เป็นขาอินพุตมาจกขาเริ่มต้นแปลงสัญญาณของเอาต์พุตตัวที่ 1 และขา $\overline{CS2}$ เป็นขาเอาต์พุตต่อ ไปยังขาเริ่มต้นแปลงสัญญาณของเอาต์พุตตัวที่ 2 เมื่อมีสัญญาณสั่งให้เอาต์พุตตัวที่ 1 ทำงาน (t_1)



รูปที่ 4.6 บล็อกไดอะแกรมและวงจรควบคุมเอาต์พุตตัวที่ 2

วงจรควบคุมจะโหลดค่าจากวงจรนับเวลา Count up มาให้กับวงจรนับเวลา Count down พร้อมทั้งเคลียร์ค่าในวงจรนับเวลา Count up ให้มีค่าเป็นศูนย์ เป็นการเริ่มต้นนับเวลาจากสัญญาณนาฬิกา โดยเวลาที่นับนี้เป็นเวลาที่ได้ทำการหาร 2 จากวงจรหาร 2 เรียบร้อยแล้ว ดังนั้นค่าที่นับได้คือค่า $\frac{t_k - t_{k-1}}{2}$ เมื่อมีสัญญาณสั่งให้เอทูดิวที่ 1 เริ่มทำการแปลงสัญญาณอีกเป็นครั้งที่ 2 (t_2) วงจรนับเวลา Count down ก็จะโหลดค่าจากวงจรนับเวลา Count up และเคลียร์ค่าของวงจรนับเวลา Count up ให้เป็นศูนย์เพื่อทำการเริ่มนับเวลาใหม่ เมื่อวงจรนับเวลา Count down นับถอยหลังถึงศูนย์เมื่อไหร่ วงจรจะส่งพัลส์เล็กๆ ไปทำการเริ่มต้นแปลงสัญญาณให้กับเอทูดิวที่ 2



รูปที่ 4.7 วงจรควบคุมการทำงานของเอทูดิวที่ 2

จะเห็นว่าเมื่อไรก็ตามที่เอทูดิวที่ 1 เริ่มทำการแปลงสัญญาณ วงจรนับเวลา Count down ก็จะได้รับค่า $\frac{t_k - t_{k-1}}{2}$ มาทำการนับถอยหลังเพื่อรอส่งสัญญาณให้กับเอทูดิวที่ 2 ต่อไป ดังนั้นช่วงเวลา t_k ถึง t'_k ก็คือค่า $\frac{t_k - t_{k-1}}{2}$ ตามรูปที่ 4.6 วงจรควบคุมที่ได้ออกแบบใช้งานจริงแสดงดังรูปที่ 4.7

ไอซี 74LS74 ทำหน้าที่เป็นวงจรถ่ายกลับ ไอซี 74LS161 ทำหน้าที่เป็นวงจรถ่ายกลับ
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Count up และไอซี 74LS191 เป็นวงจรมับเวลา Count down ส่วนไอซี 74LS121 ทำหน้าที่เป็นวงจรมับเวลา (Delay time) เพื่อไม่ให้การไหลคข้อมูลจากวงจรมับ Count up (74LS161) ไปยังวงจรมับ Count down (74LS191) และการเคลียร์ค่าในวงจรมับเวลา Count up ทำงานพร้อมกัน โดยการไหลคข้อมูลจะกระทำก่อนการเคลียร์ค่าของวงจรมับเวลา Count up



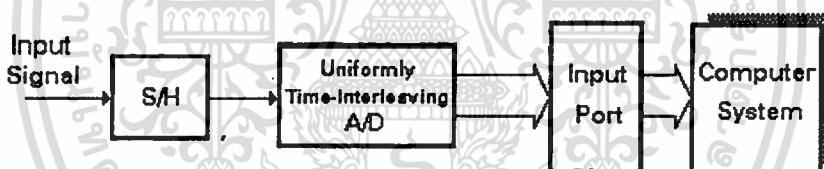
บทที่ 5

การประยุกต์ใช้งาน

จากวงจรควบคุมเอทูดิ จำนวน 2 ตัว ที่ทำงานแบบ Time-Interleaving ในบทที่ 4 สามารถนำไปประยุกต์ใช้งานกับคอมพิวเตอร์ต่างๆ ได้ โดยทำหน้าที่เป็นส่วนของ Data Acquisition Unit. ทำการวัดสัญญาณอนาลอกอินพุตให้กับคอมพิวเตอร์ เพื่อนำค่าไปประมวลผลต่อไป

5.1 การเชื่อมโยงวงจรเข้ากับระบบคอมพิวเตอร์

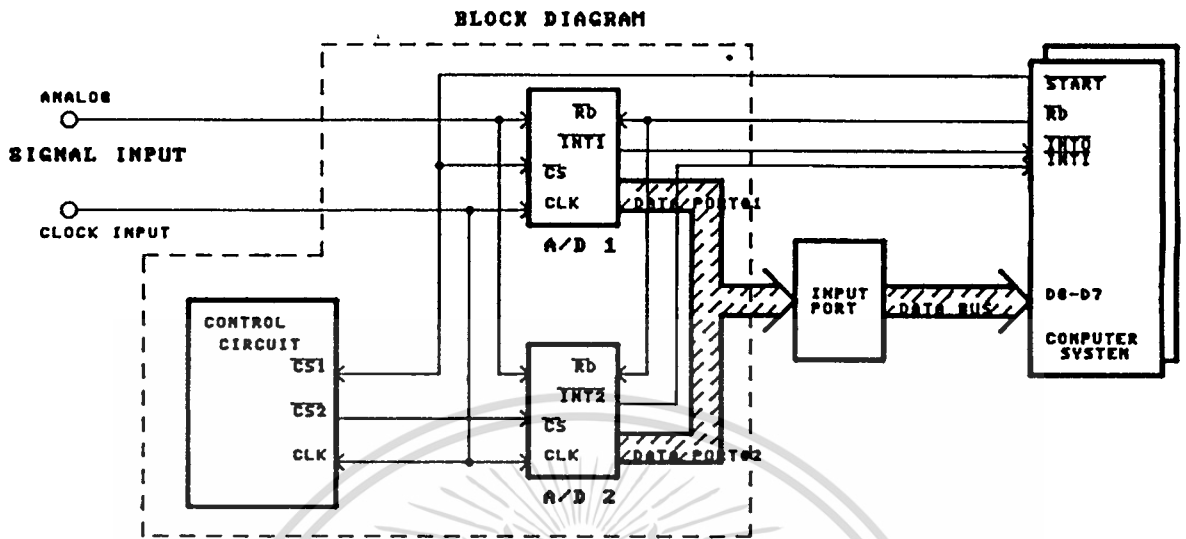
Uniformly Time-Interleaving เอทูดิ ที่ออกแบบนี้สามารถต่อใช้งานเข้ากับระบบคอมพิวเตอร์ต่างๆ ไปเหมือนกับเอทูดิธรรมดา บล็อกไดอะแกรมของ Data Acquisition Unit ที่ใช้ Uniformly Time-Interleaving เอทูดิ เป็นตัวแปลงสัญญาณ แสดงดังรูปที่ 5.1



รูปที่ 5.1 บล็อกไดอะแกรมของ Data Acquisition Unit

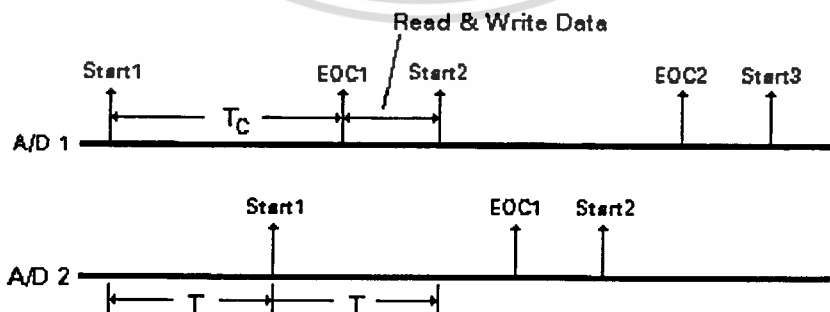
เนื่องจากใช้เอทูดิจำนวน 2 ตัว ดังนั้นคอมพิวเตอร์จึงมองเห็นเป็นพอร์ตจำนวน 2 พอร์ต คือ พอร์ตข้อมูลของเอทูดิตัวที่ 1 และพอร์ตข้อมูลของเอทูดิตัวที่ 2 ซึ่งพอร์ตข้อมูลทั้งสองนี้จะ ต่อขนานกัน แต่คอมพิวเตอร์จะใช้วงจร Address Decoder ทำหน้าที่ในการเลือกอ่านข้อมูล จากพอร์ตทั้งสอง ลักษณะการต่อวงจรแสดงดังรูปที่ 5.2 โดยมีลักษณะการทำงานดังนี้

คอมพิวเตอร์จะส่งคำสั่ง Start ผ่านไปยังขา $\overline{CS1}$ เพื่อให้เอทูดิตัวที่ 1 เริ่มทำการแปลง สัญญาณ หลังจากนั้นวงจรควบคุมภายใน Uniformly Time-Interleaving เอทูดิ จะสั่งให้ เอทูดิตัวที่ 2 เริ่มแปลงสัญญาณโดยมีช่วงเวลาเหลื่อมล้ำกันตามที่ได้ออกแบบไว้ในบทที่ 4 เมื่อเอทูดิตัวที่ 1 แปลงสัญญาณเรียบร้อยแล้ว ก็จะส่งสัญญาณ $\overline{INT1}$ มาบอกกับคอมพิวเตอร์



รูปที่ 5.2 การต่อ Time-Interleaving เหนือคิ เข้ากับคอมพิวเตอร์

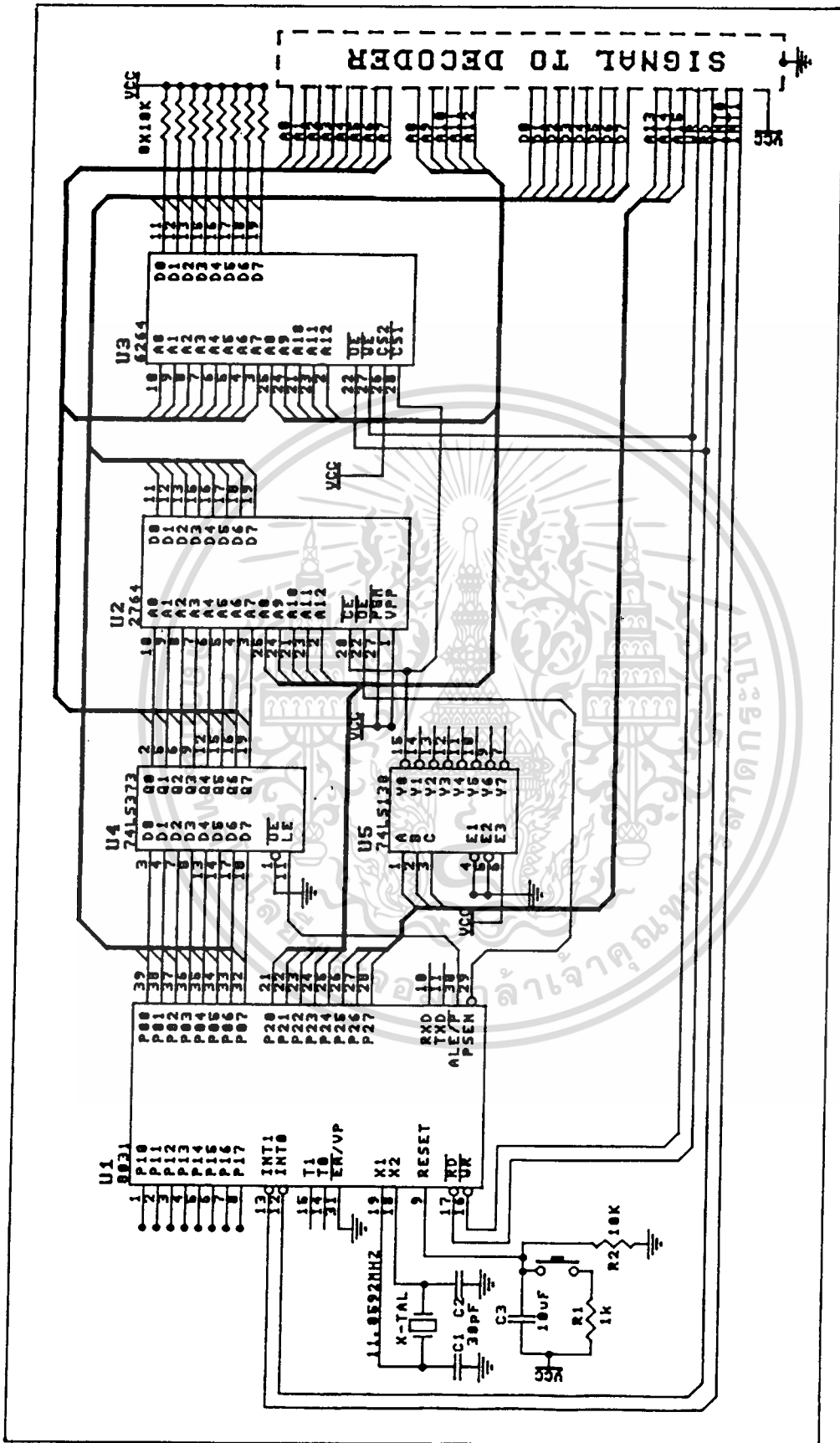
เพื่อให้มารับข้อมูลดิจิทัลไป คอมพิวเตอร์ก็จะอ่านข้อมูลจากพอร์ทข้อมูลที่ 1 นำค่าไปเก็บไว้ในหน่วยความจำ แล้วสั่งให้เหนือคิตัวที่ 1 เริ่มแปลงสัญญาณต่อไป หลังจากนั้นเมื่อเหนือคิตัวที่ 2 แปลงสัญญาณเรียบร้อยแล้ว ก็จะส่งสัญญาณ $\overline{INT2}$ ให้กับคอมพิวเตอร์ และคอมพิวเตอร์ก็จะอ่านข้อมูลดิจิทัลจากพอร์ทข้อมูลที่ 2 นำไปเก็บไว้ในหน่วยความจำต่อไป ต่อมาเมื่อเหนือคิตัวที่ 1 แปลงสัญญาณเสร็จเรียบร้อยแล้วก็จะส่งสัญญาณ $\overline{INT1}$ มายังคอมพิวเตอร์อีก และคอมพิวเตอร์ก็อ่านข้อมูลจากพอร์ทที่ 1 ไปเก็บไว้ในหน่วยความจำ การทำงานก็จะเป็นอย่างนี้เรื่อยๆ ไป จนกว่าคอมพิวเตอร์จะได้ข้อมูลครบตามจำนวนการสุ่มที่ต้องการ



รูปที่ 5.3 Timing ไคอะแกรมการทำงานของเหนือคิตัวที่ 1 และ 2

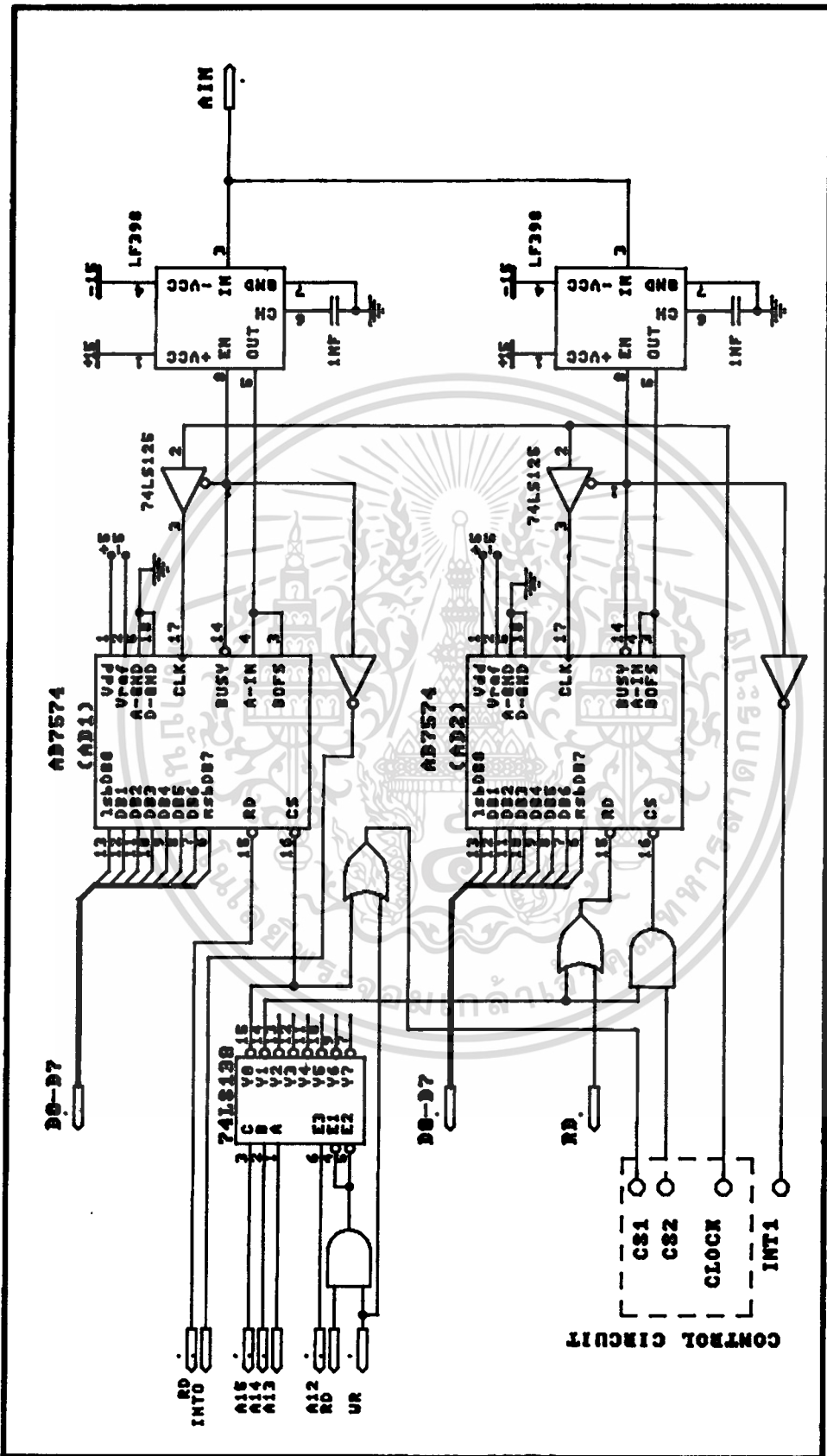
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 ซีพียูบอร์ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

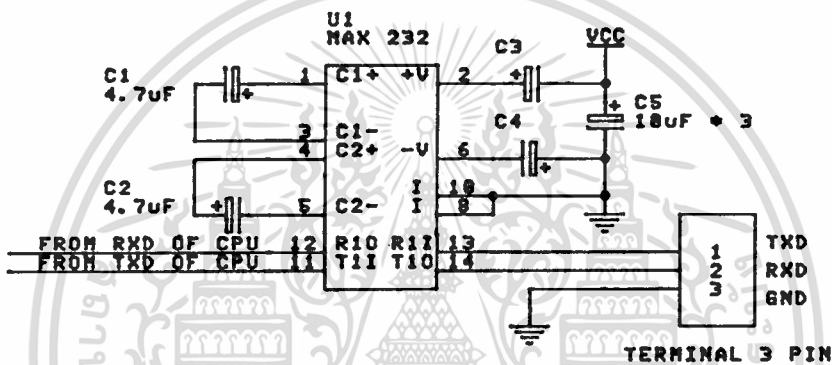


รูปที่ 5.5 Data Acquisition Unit แบบใช้ไอพูตี สองตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 38 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timing ไดอะแกรมในการสั่งให้เอชทูดีตัวที่ 1 ทำงานและคอมพิวเตอร์รับข้อมูลจากพอร์ตข้อมูล 1 ของเอชทูดีตัวที่ 1 และพอร์ตข้อมูล 2 ของเอชทูดีตัวที่ 2 แสดงดังรูปที่ 5.3

วิทยานิพนธ์ฉบับนี้ได้ทำการทดลองต่อวงจร Uniformly Time-Interleaving เอชทูดีเข้ากับไมโครคอนโทรลเลอร์เบอร์ 8031 เพื่อทำการอ่านข้อมูลจาก Data Acquisition Unit เก็บค่าไว้ในหน่วยความจำ แล้วส่งค่าที่อ่านได้ทั้งหมดให้กับไมโครคอมพิวเตอร์เพื่อเก็บลงบนแผ่นดิสเก็ตหรือนำไปประมวลผลบนไมโครคอมพิวเตอร์ต่อไป วงจรทางด้านฮาร์ดแวร์แสดงให้เห็นดังรูปที่ 5.4, 5.5 และ 5.6



รูปที่ 5.6 วงจรพอร์ตอนุกรมสำหรับ 8031

ชุดทดลองที่ใช้จะประกอบด้วยส่วนใหญ่ว่า 3 ส่วนคือ ส่วนของซีพียู ส่วนของ Data Acquisition Unit และส่วนที่ใช้ติดต่อกับไมโครคอมพิวเตอร์ ในส่วนของซีพียูแสดงดังรูปที่ 5.4 ประกอบด้วย 8031 ไมโครคอนโทรลเลอร์ทำหน้าที่เป็นหน่วยประมวลผลกลาง และมีไอซี 2764 เป็น Program Memory ขนาด 8k ไอซี 6264 เป็น Data Memory ขนาด 8k ส่วนที่ใช้ติดต่อกับไมโครคอมพิวเตอร์จะใช้พอร์ตอนุกรมของ 8031 ต่อผ่าน MAX232 เพื่อติดต่อกับพอร์ตอนุกรม RS-232C ของไมโครคอมพิวเตอร์ IBM PC วงจรของพอร์ตอนุกรมแสดงดังรูปที่ 5.6 สำหรับ Data Acquisition Unit แสดงดังรูปที่ 5.5 ประกอบด้วย Time-Interleaving เอชทูดีแบบ 2 ตัว โดยใช้ไอซีเอชทูดีเบอร์ AD7574 ซึ่งมีค่า Conversion Time เท่ากับ 16us ที่ความถี่ 500 kHz และมีไอซี LF398 ทำหน้าที่เป็นวงจร Sample & Hold ให้กับเอชทูดีแต่ละตัว การเชื่อมโยงเข้ากับ 8031 ไมโครคอนโทรลเลอร์จะใช้ไอซี 74LS138 ทำหน้าที่เป็น Address Decoder โดยมีพอร์ตหมายเลข "1FFFH" เป็นพอร์ต

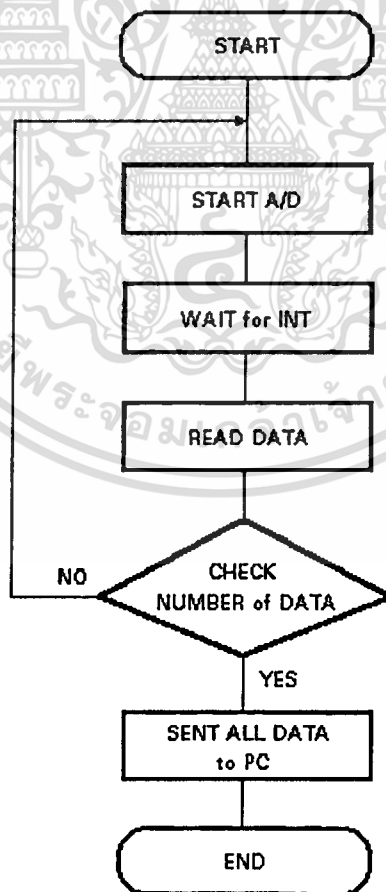
ที่ใช้กับเหตุเดี่ยวที่ 1 เพื่อสั่งให้เริ่มแปลงสัญญาณหรือรับข้อมูลดิจิทัลที่แปลงเรียบร้อยแล้ว พอร์ทหมายเลข "3FFFH" เป็นพอร์ทที่ใช้ในการรับข้อมูลดิจิทัลที่แปลงเรียบร้อยแล้ว จากเหตุเดี่ยวที่ 2 ขา $\overline{\text{BUSY}}$ ของเหตุเดี่ยวที่ 1 และเดี่ยวที่ 2 จะนำมาใช้เป็นขาอินเทอร์รัพท์ $\overline{\text{INT0}}$ และ $\overline{\text{INT1}}$ ตามลำดับ ให้กับ 8031 ไมโครคอนโทรลเลอร์

5.2 โปรแกรมที่ใช้ในการควบคุมการทำงานของวงจร

จากวงจร Data Acquisition ในรูปที่ 5.6 สามารถเขียนโปรแกรมควบคุมการทำงานได้ 2 แบบด้วยกันคือ

1. เลือกใช้เหตุเดี่ยวเพียงตัวเดียว
2. เลือกใช้เหตุเดี่ยวสองตัว

กรณีที่ใช้เหตุเดี่ยวเพียงตัวเดียว สามารถเขียนโปรแกรมได้ตามโฟลว์ชาร์ทรูปที่ 5.8



รูปที่ 5.8 โฟลว์ชาร์ทสำหรับ Data Acquisition แบบใช้เหตุเดี่ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอทิวดีที่ใ้ใช้คือเอทิวดีตัวที่ 1 เท่านั้น ส่วนเอทิวดีตัวที่ 2 จะไม่มีผลใดๆ กับการทำงาน เริ่มต้นด้วยการสั่งใ้เอทิวดีเริ่มทำการแปลงสัญญาณ จากนั้นก็จะรอนจนกว่าเอทิวดีจะแปลงสัญญาณเสร็จ โดยเอทิวดีจะส่งสัญญาณอินเตอร์รัพท์มาใ้ซีพียูที่ขา INTO เมื่อซีพียูได้รับสัญญาณอินเตอร์รัพท์จะกระโดดไปทำคำสั่งที่ตำแหน่งแอดเดรส 0003H ณ ที่ตำแหน่งนี้ ก็จะมีการเขียนโปรแกรมใ้ใ้ซีพียูอ่านค่าจากเอทิวดีนำไปเก็บไว้ยังหน่วยความจำ หลังจากนั้นซีพียูจะตรวจสอบว่าข้อมูลที่ได้จากเอทิวดีแปลงสัญญาณนั้น ครอบคลุมจำนวนที่กำหนดหรือไม่ ถ้ายังไม่ครบก็จะใ้เริ่มต้นใหม่ ถ้าครอบคลุมจำนวนที่ต้องการแล้วก็ส่งข้อมูลที่ใ้ทั้งหมดผ่านพอร์ทอนุกรมไปยังเครื่องพีซีใ้เก็บเป็นแฟ้มข้อมูลลงบนแผ่นดิสเก็ตต่อไป โปรแกรมภาษาแอสเซมบลีของโพลีซาร์ทรูปที่ 5.8 มีดังนี้

```

CS1 EQU 1F00H
ST_RAM EQU 80H
ORG 0000H
LJMP START

ORG 0003H
LJMP 0300H

ORG 0100H
START: MOV TMOD,#20H
MOV SCON,#50H
MOV TH1,#0FDH
MOV R6,#00H
SETB TR1
SETB IE.0
SETB IE.7
MOV DPTR,#CS1
MOVX @DPTR,A
CJNE R6,#20H,$

CLR IE.7
MOV R6,#00H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานใ้การศึกษาเท่านั้น ไม่อนุญาตใ้ให้นำไปใ้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิใ้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรใ้ นำไปใ้

```

MOV   DPH,#ST_RAM
LOOP: MOV   DPL,R6
      MOVX  A,@DPTR
      MOV   SBUF,A
      JNB   TI,$
      CLR   TI
      INC   R6
      CJNE  R6,#20H,LOOP
      AJMP  $

```

```

ORG   0300H      ;INT0
CLR   IE.7
MOV   DPTR,#CS1
MOVX  A,@DPTR
MOV   DPH,#ST_RAM
MOV   DPL,R6
MOVX  @DPTR,A
INC   R6
MOV   DPTR,#CS1
MOVX  @DPTR,A
LCALL DELAY
SETB  IE.7
RETI

```

```

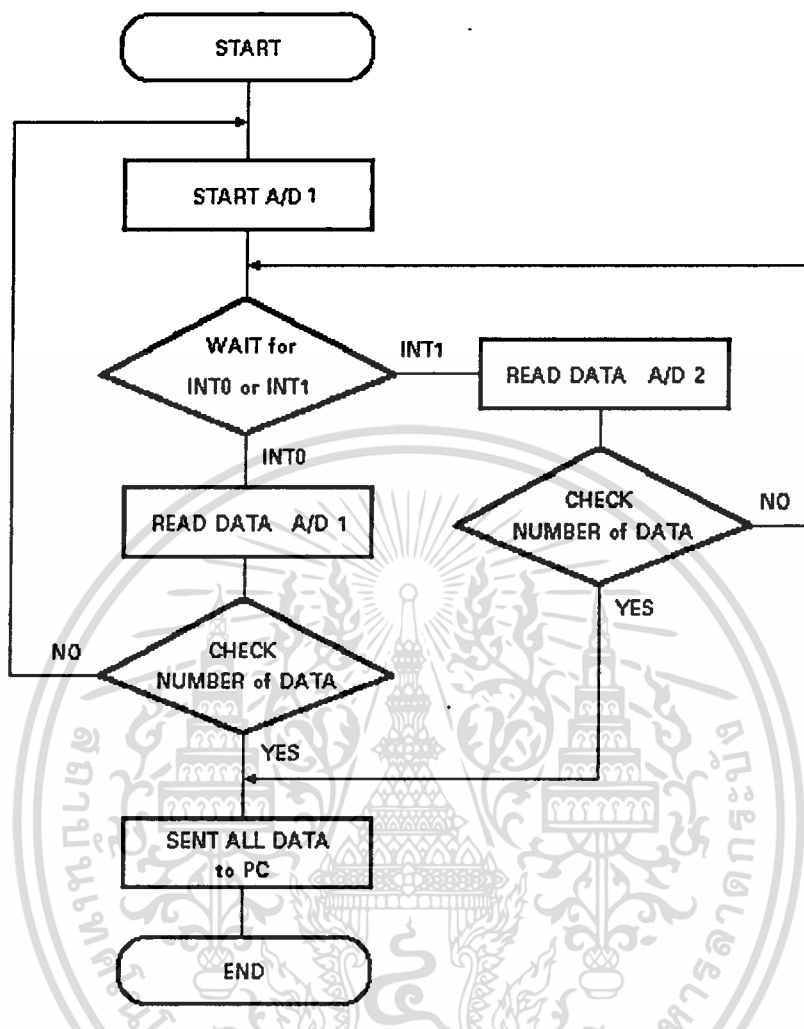
DELAY: MOV   R1,#08h      ;DELAY TIME
      DJNZ  R1,$
      RET
      END

```

กรณีที่ใช้เอ็ดิตสองตัว เพื่อให้ทำงานเป็นแบบ Uniformly Time - Interleaving สามารถเขียนโปรแกรมได้ตามโพล์ซาร์ทรูปที่ 5.9 เริ่มต้นด้วยการสั่งให้เอ็ดิตตัวที่ 1 (AD1) เริ่มทำการแปลงสัญญาณ จากนั้นก็จะรอสัญญาณอินเตอร์รัพท์จากเอ็ดิตตัวที่ 1 หรือเอ็ดิตตัวที่ 2 (AD2) โดยที่เอ็ดิตตัวที่ 1 จะส่งสัญญาณอินเตอร์รัพท์มาให้ซีพียูที่ขา $\overline{INT0}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.9 โฟลว์ชาร์ตสำหรับ Data Acquisition แบบใช้เอชดี 2 ตัว

และเอชดีตัวที่ 2 จะส่งสัญญาณอินเทอร์รัพท์มาที่ขา $\overline{INT1}$ (ถ้าซีพียูได้รับอินเทอร์รัพท์ $\overline{INT1}$ ซีพียูจะกระโดดไปทำคำสั่งที่ตำแหน่งแอดเดรส 0013H) ในกรณีที่เอชดีตัวที่ 2 ส่งสัญญาณอินเทอร์รัพท์มา ซีพียูก็จะอ่านข้อมูลที่ได้จากเอชดีตัวที่ 2 นำไปเก็บไว้ยังหน่วยความจำ หลังจากนั้นซีพียูจะตรวจสอบว่าข้อมูลที่ได้จากเอชดีแปลงสัญญาณนั้น ครบตามจำนวนที่กำหนดหรือไม่ ถ้าครบก็จะส่งข้อมูลทั้งหมดผ่านพอร์ตอนุกรมไปยังเครื่องพีซี ถ้ายังไม่ครบก็จะกลับไปรอสัญญาณอินเทอร์รัพท์ต่อ แต่ถ้าเอชดีตัวที่ 1 ส่งสัญญาณอินเทอร์รัพท์มา ซีพียูจะอ่านข้อมูลที่ได้จากเอชดีตัวที่ 1 นำไปเก็บไว้ยังหน่วยความจำ หลังจากนั้นซีพียูจะตรวจสอบว่าข้อมูลที่ได้จากเอชดีแปลงสัญญาณนั้นครบตามจำนวนที่กำหนดหรือไม่ ถ้าครบ

ก็จะส่งข้อมูลทั้งหมดผ่านพอร์ทอนุกรมไปยังเครื่องพีซี ถ้ายังไม่ครบก็จะกลับไปเริ่มต้นส่งให้เอ็ดดีตัวที่ 1 เริ่มต้นแปลงสัญญาณใหม่อีกครั้ง จนกว่าจะได้ข้อมูลครบตามต้องการ

จะเห็นว่าซีพียูไม่ต้องส่งคำสั่งให้เอ็ดดีตัวที่สองเริ่มต้นแปลงสัญญาณเลย เนื่องจากว่าวงจรควบคุมคาบเวลาจะทำหน้าที่ส่งสัญญาณ CS2 ให้กับเอ็ดดีตัวที่ 2 เริ่มต้นแปลงสัญญาณเองโดยอัตโนมัติ ดังนั้นซีพียูจึงมีหน้าที่รับข้อมูลจากเอ็ดดีตัวที่ 2 เพียงอย่างเดียวเท่านั้น โปรแกรมภาษาแอสเซมบลีของโพลีซาร์รูปที่ 5.9 มีดังนี้

```
CS1 EQU 1F00H
CS2 EQU 3F00H
ST_RAM EQU 80H

ORG 0000H
LJMP START

ORG 0003H
LJMP 0300H

ORG 0013H
LJMP 0400H

ORG 0100H
START: MOV TMOD,#20H
MOV SCON,#50H
MOV TH1,#0FDH
MOV R6,#00H
SETB TR1
SETB IE.0
SETB IE.2
SETB IE.7
LCALL FIRST
CJNE R6,#40H,$

CLR IE.7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา บุคคลต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R6,#00H
MOV DPH,#ST_RAM
LOOP: MOV DPL,R6

```

```

MOVX A,@DPTR
MOV SBUF,A
JNB TI,$
CLR TI
INC R6
CJNE R6,#40H,LOOP
AJMP $

```

```

ORG 0300h ;INT0
CLR IE.7
MOV DPTR,#CS1
MOVX A,@DPTR
MOV DPH,#ST_RAM
MOV DPL,R6
MOVX @DPTR,A
INC R6
FIRST: MOV DPTR,#CS1
MOVX @DPTR,A
SETB IE.7
RETI

```

```

ORG 0400H ;INT1
CLR IE.7
MOV DPTR,#CS2
MOVX A,@DPTR

MOV DPH,#ST_RAM
MOV DPL,R6
MOVX @DPTR,A

```

```

INC R6

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SETB IE.7

RETI

END

ที่เครื่องพีซีจะมีการเขียนโปรแกรมเพื่อรับข้อมูลจากไมโครคอนโทรลเลอร์ที่ส่งผ่าน มาทางพอร์ทอนุกรม RS-232C โดยเลือกใช้โปรแกรม Microsoft QuickBASIC ทำการเก็บ ข้อมูลที่ได้ทั้งหมดไว้ในแฟ้มข้อมูลชื่อ "data.dat" ข้อมูลที่เก็บนั้นจะเป็นลักษณะของค่าค่า ไฟล์ หรือ ASCII ไฟล์ โดยมีค่าเป็นระดับแรงดันของสัญญาณอินพุต ตัวโปรแกรมที่ใช้ใน การเก็บข้อมูลมีดังนี้

```
OPEN "data.dat" FOR OUTPUT AS #1
```

```
OPEN "com1:9600,n,8,1" FOR INPUT AS #2
```

```
FOR i = 1 to 256
```

```
in$ = INPUT$(1, #2)
```

```
volt = ASC(in$)
```

```
volt = (5/256)*volt
```

```
PRINT #1, volt
```

```
NEXT i
```

```
CLOSE #1
```

```
CLOSE #2
```

```
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

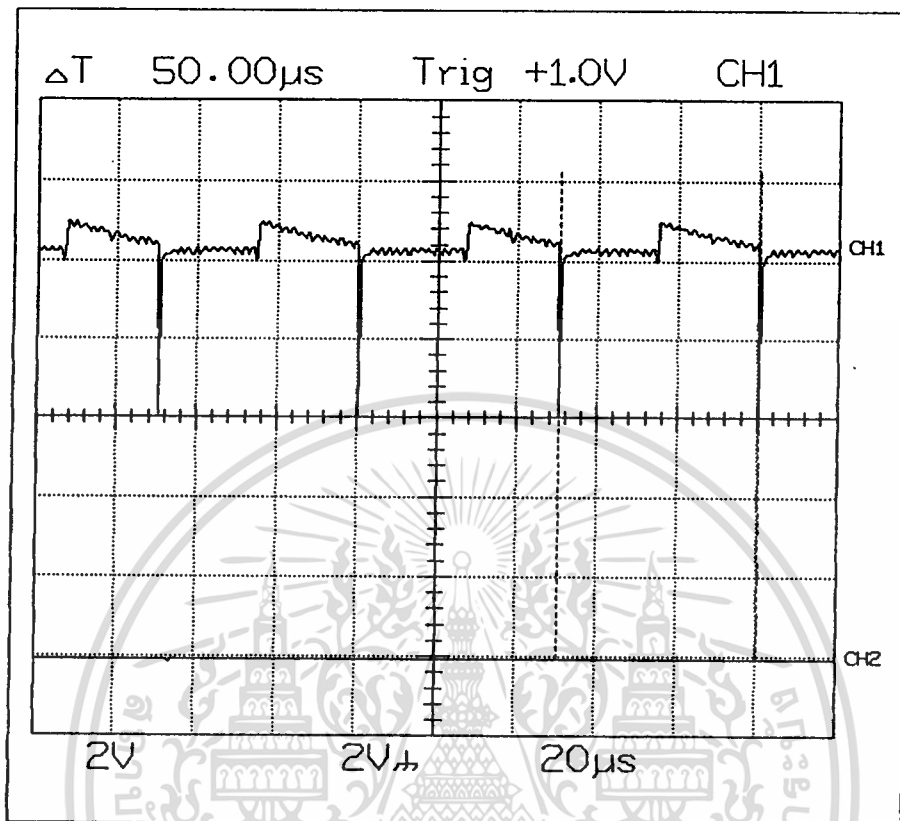
ผลการทดลองและบทสรุป

ลักษณะของการทดลองวงจรจะเป็นการทดลองวัดคาบเวลาในการสุ่มสัญญาณระหว่างเอชดีตัวที่ 1 กับเอชดีตัวที่ 2 ว่ามีคาบเวลาเท่ากันตรงตามทฤษฎีหรือไม่ โดยทดลองสุ่มสัญญาณที่อัตราการสุ่มค่าต่างกัน เปรียบเทียบการสุ่มสัญญาณแบบใช้เอชดีเพียงตัวเดียว และแบบ Time-Interleaving ใช้เอชดีสองตัว ที่มีวงจรควบคุมการสุ่มสัญญาณเป็นแบบ Uniformly Sampled Signal

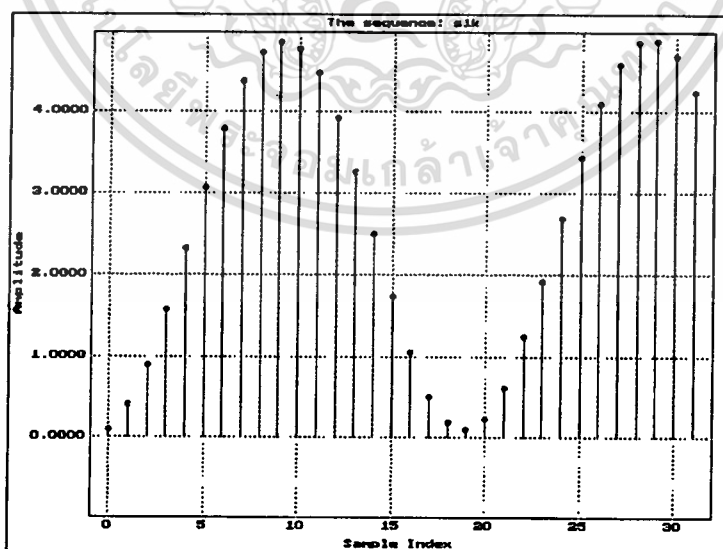
6.1 การทดลองวัดสัญญาณอินพุตด้วยไมโครคอนโทรลเลอร์

การทดลองในวิทยานิพนธ์ฉบับนี้ ได้ทำการทดลองวัดสัญญาณอินพุตที่เป็นสัญญาณไซน์ ค่าความถี่ต่างๆ กัน โดยทำการวัดสัญญาณทั้ง 2 แบบคือ แบบใช้เอชดีเพียงตัวเดียว และแบบใช้เอชดีสองตัว ที่มีวงจรควบคุมการสุ่มสัญญาณ ในแบบที่ใช้เอชดีเพียงตัวเดียว ได้ทำการตรวจสอบการทำงานของเอชดี โดยใช้โปรแกรมตามรูปที่ 5.8 และวัดสัญญาณเริ่มต้นการแปลงสัญญาณของเอชดีตัวที่ 1 (AD1) คือขา Chip Select (\overline{CS}) ได้รูปสัญญาณตามรูปที่ 6.1 สัญญาณอินพุตที่วัดเป็นสัญญาณไซน์ความถี่ 1kHz ($2.5+2.5 \sin 2\pi 1000t$) ชุดทดลองสามารถสุ่มสัญญาณได้ประมาณ 20000 Samples/sec หรือสามารถวัดสัญญาณที่มีความถี่สูงสุดไม่เกิน 10kHz ตามหลักการของ Nyquist จากโปรแกรมที่ใช้ทดลองทำการสุ่มสัญญาณจำนวน 32 ค่า ได้สัญญาณตามรูปที่ 6.2 และ 6.3 ซึ่งแสดงเป็นแบบ Discrete และ Continuous ตามลำดับ สำหรับค่า ณ.จุดต่างๆ ที่ได้จากการอ่านค่าของเอชดี แสดงในตารางที่ 6.1 มีหน่วยเป็น โวลท์

จากวงจรในรูปที่ 5.6 การทดลองแบบใช้เอชดีจำนวนสองตัวทำงานแบบ Uniformly Time-Interleaving ได้ทำการตรวจสอบการทำงานของเอชดีทั้งสองตัว โดยใช้โปรแกรมตามรูปที่ 5.9 และวัดสัญญาณเริ่มต้นการแปลงสัญญาณของเอชดีแต่ละตัว โดยเอชดีตัวที่หนึ่ง (AD1) ใช้สไลป์ CH1 จับสัญญาณที่ขา $\overline{CS1}$ ของ Control Circuit ส่วนเอชดีตัวที่สอง (AD2) ใช้สไลป์ CH2 จับสัญญาณที่ขา $\overline{CS2}$ ของ Control Circuit ได้รูปสัญญาณตามรูปที่ 6.4 สัญญาณอินพุตที่วัดเป็นสัญญาณไซน์ความถี่ 1kHz ($2.5+2.5 \sin 2\pi 1000t$)

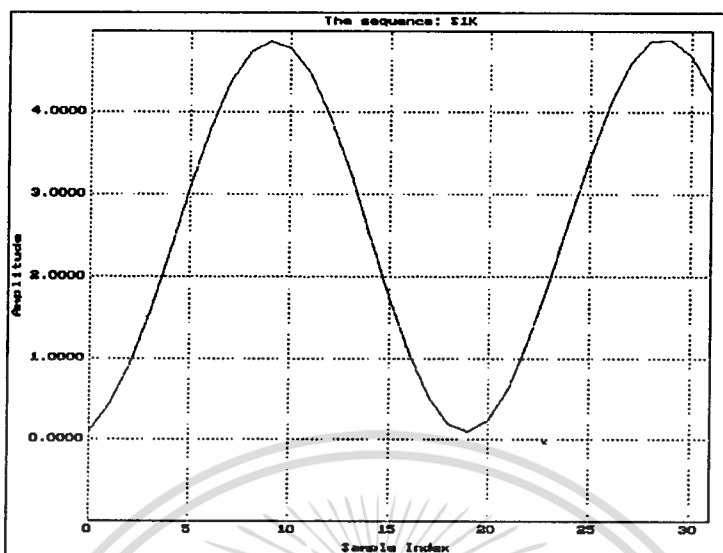


รูปที่ 6.1 สัญญาณ Chip select ของ AD1



รูปที่ 6.2 สัญญาณ ไซน์ความถี่ 1kHz ที่ได้จากการสุ่มโดยใช้เอทูดิตัวเดียว
แสดงเป็นแบบ Discrete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



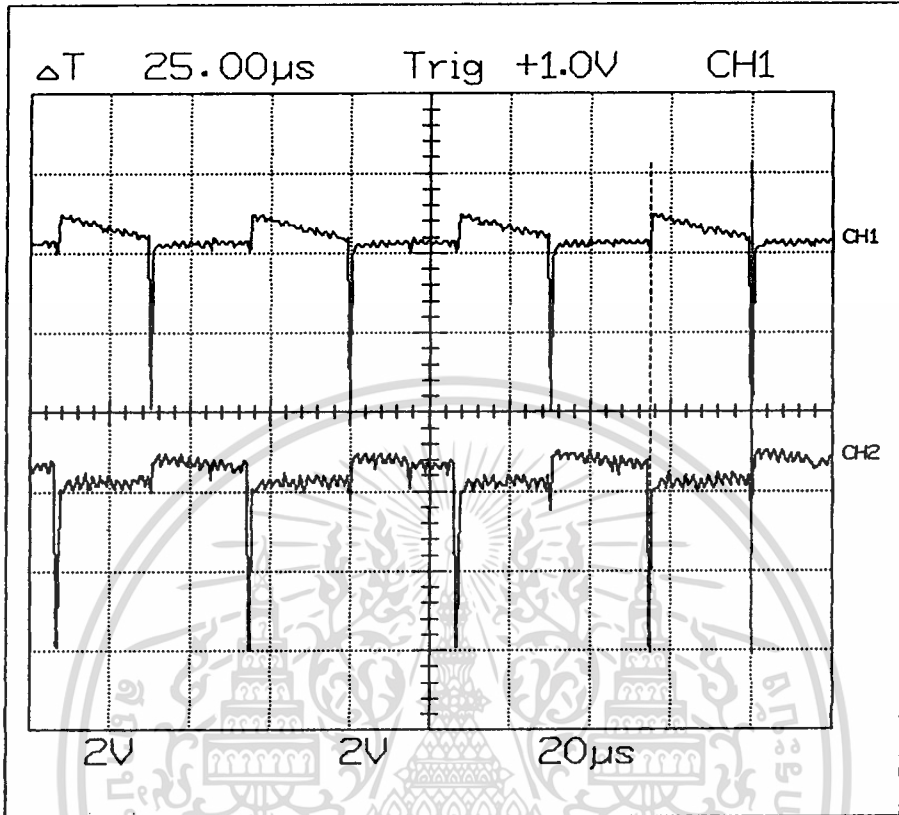
รูปที่ 6.3 สัญญาณไซน์ความถี่ 1kHz ที่ได้จากการสุ่มโดยใช้เอทาคิตัวเดียว แสดงเป็นแบบ Continuous

Index	Amplitude	Index	Amplitude	Index	Amplitude	Index	Amplitude
0	0.097656	8	4.746094	16	1.054688	24	2.675781
1	0.410156	9	4.863281	17	0.507813	25	3.437500
2	0.898438	10	4.785156	18	0.195313	26	4.101563
3	1.562500	11	4.472656	19	0.097656	27	4.589844
4	2.324219	12	3.925781	20	0.234375	28	4.863281
5	3.066406	13	3.261719	21	0.625000	29	4.882813
6	3.789063	14	2.500000	22	1.250000	30	4.687500
7	4.375000	15	1.738281	23	1.914063	31	4.238281

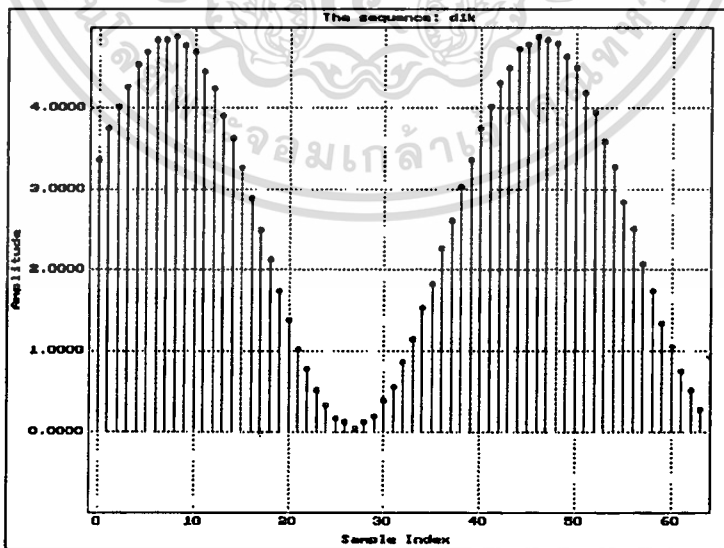
ตารางที่ 6.1 ค่าสัญญาณที่วัดได้จากรูปที่ 6.2

จะสังเกตเห็นว่าเมื่อใช้เอทาคิตัวจำนวน 2 ตัว ช่วงห่างระหว่างการสุ่มสัญญาณแต่ละครั้งจะมีค่าใกล้เคียงกันคือ $25 \mu\text{s}$ ทำให้ได้อัตราการสุ่มสัญญาณมีค่าเพิ่มขึ้นเป็น 40000 Samples/sec หรือสามารถวัดสัญญาณที่มีความถี่สูงสุดไม่เกิน 20kHz ซึ่งแสดงให้เห็นว่าวงจรควบคุมคาบเวลาสำหรับการเริ่มต้นแปลงสัญญาณของเอทาคิตัวที่สองทำงานได้ถูกต้องตามหลักการที่ได้กล่าวไว้ในบทที่ 4 จากโปรแกรมรูปที่ 5.9 ซึ่งเป็นโปรแกรมที่ใช้ทดลองทำการสุ่มสัญญาณจำนวน 64 ค่า ได้สัญญาณตามรูปที่ 6.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

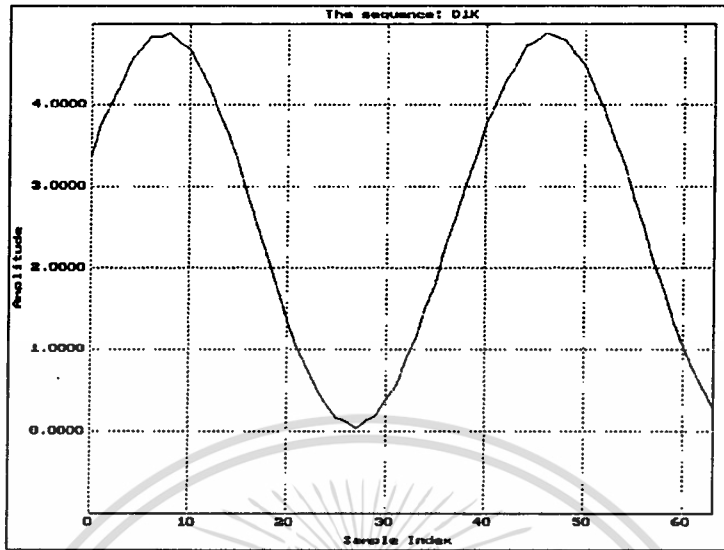


รูปที่ 6.4 สัญญาณ Chip select ของ AD1 และ AD2



รูปที่ 6.5 สัญญาณไซน์ความถี่ 1 kHz ที่ได้จากการสุ่มโดยใช้เอทิตสองตัว
แสดงเป็นแบบ Discrete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



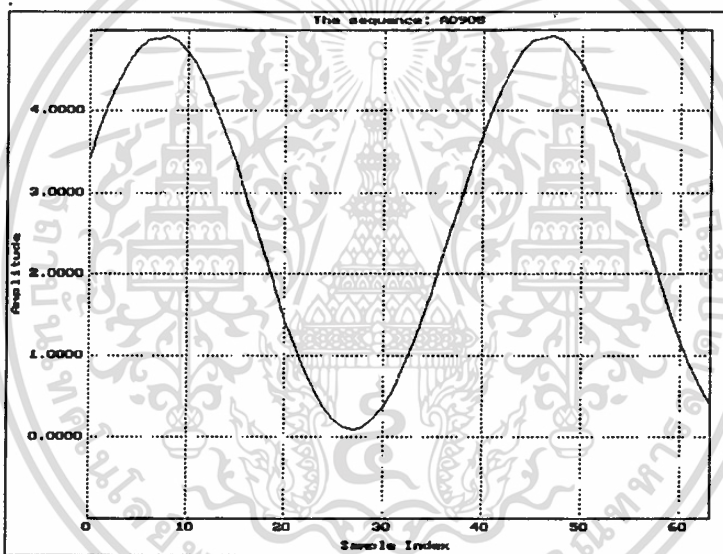
รูปที่ 6.6 สัญญาณไซน์ความถี่ 1 kHz ที่ได้จากการสุ่มโดยใช้เอทาคิสองตัว แสดงเป็นแบบ Continuous

Index	Amplitude	Index	Amplitude	Index	Amplitude	Index	Amplitude
0	3.359375	16	2.890625	32	0.859375	48	4.804688
1	3.750000	17	2.500000	33	1.132813	49	4.628906
2	4.023438	18	2.128906	34	1.523438	50	4.492188
3	4.257813	19	1.738281	35	1.816406	51	4.179688
4	4.531250	20	1.367188	36	2.265625	52	3.945313
5	4.687500	21	1.015625	37	2.617188	53	3.593750
6	4.843750	22	0.781250	38	3.027344	54	3.281250
7	4.843750	23	0.507813	39	3.359375	55	2.851563
8	4.882813	24	0.332031	40	3.750000	56	2.519531
9	4.765625	25	0.156250	41	4.023438	57	2.070313
10	4.687500	26	0.117188	42	4.316406	58	1.738281
11	4.453125	27	0.039063	43	4.492188	59	1.328125
12	4.238281	28	0.117188	44	4.726563	60	1.035156
13	3.906250	29	0.195313	45	4.785156	61	0.742188
14	3.632813	30	0.390625	46	4.882813	62	0.507813
15	3.261719	31	0.546875	47	4.843750	63	0.273438

ตารางที่ 6.2 ค่าสัญญาณที่วัดได้จากรูปที่ 6.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปรียบเทียบสัญญาณที่ได้จากอัตราการสุ่ม 20000 Samples/sec และ 40000 Samples/sec ตามรูปที่ 6.2 และ 6.5 ตามลำดับ จะพบว่าสัญญาณที่ได้เหมือนกัน แต่ที่อัตราการสุ่มค่าสูงๆ รูปสัญญาณที่ได้จะมีความละเอียดมากกว่าอัตราการสุ่มค่าต่ำๆ เพื่อเป็นการตรวจสอบการทำงานของการใช้เทอติกแบบ Time-Interleaving ว่าทำงานถูกต้อง ได้ทำการใช้ไอซีเทอติก AD908 ที่มีความเร็วในการแปลงสัญญาณ 6 μ S ต่อกับไมโครคอมพิวเตอร์ให้มีอัตราการสุ่มเท่ากับ 40000 Samples/sec ซึ่งมีค่าเท่ากับชุดทดลองที่ใช้ในวิทยานิพนธ์ฉบับนี้ วัดสัญญาณไซน์ความถี่ 1kHz จำนวน 64 ค่า ได้ดังรูปที่ 6.7 ซึ่งค่าที่ได้จากการสุ่มสัญญาณทั้ง 64 ค่า แสดงในตารางที่ 6.3



รูปที่ 6.7 สัญญาณไซน์ 1kHz ใช้เทอติกตัวเดียวที่มีอัตราการสุ่ม 40000 Samples/sec

Index	Amplitude	Index	Amplitude	Index	Amplitude	Index	Amplitude
0	3.417969	16	2.968750	32	0.839844	48	4.882813
1	3.769531	17	2.597656	33	1.132813	49	4.746094
2	4.101563	18	2.226563	34	1.464844	50	4.570313
3	4.375000	19	1.816406	35	1.835938	51	4.335938
4	4.589844	20	1.445313	36	2.226563	52	4.082031
5	4.765625	21	1.132813	37	2.617188	53	3.750000
6	4.882813	22	0.820313	38	2.988281	54	3.398438
7	4.902344	23	0.566406	39	3.398438	55	3.027344
8	4.921875	24	0.371094	40	3.691406	56	2.656250

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9	4.863281	25	0.214844	41	4.023438	57	2.265625
10	4.726563	26	0.117188	42	4.316406	58	1.894531
11	4.550781	27	0.097656	43	4.550781	59	1.503906
12	4.316406	28	0.117188	44	4.746094	60	1.171875
13	4.003906	29	0.234375	45	4.863281	61	0.859375
14	3.691406	30	0.371094	46	4.902344	62	0.605469
15	3.359375	31	0.585938	47	4.921875	63	0.390625

ตารางที่ 6.3 ค่าสัญญาณที่วัดได้จากรูปที่ 6.7

เมื่อเปรียบเทียบรูปที่ 6.6 กับรูปที่ 6.7 และตารางที่ 6.2 กับตารางที่ 6.3 จะเห็นว่า มีค่าเหมือนกัน แสดงว่าการใช้ไอซีเอชดีจำนวน 2 ตัวแบบ Time-Interleaving นี้สามารถวัดสัญญาณได้ถูกต้องเหมือนกับการใช้ไอซีเอชดีตัวเดียวที่มีอัตราการสุ่มสัญญาณเท่ากัน

	เอชดีตัวเดียว (AD7574)	เอชดี 2 ตัว (AD7574x2)
Resolution	8 บิต	8 บิต
conversion time	15 μ S	15 μ S
Sampling rate (ต่อกับ CPU 8031)	20000 Samples/sec	40000 Samples/sec
Cost	ปกติ	เพิ่มขึ้นประมาณ 2 เท่า
Application	Data Acquisition	Data Acquisition

ตารางที่ 6.4 เปรียบเทียบการใช้งานเอชดีตัวเดียวกับเอชดี 2 ตัว

ค่า Quantization ของทั้ง 2 วิธี มีค่าเท่ากับ $1/256$ เพราะต่างก็มีค่า Resolution ขนาด 8 บิตเท่ากัน สำหรับค่า gain error, offset error, integral nonlinearity และ Differential nonlinearity ในกรณีที่ใช้การสุ่มสัญญาณแบบเอชดี 2 ตัวต่อแบบ Time-Interleaving นั้น จะมีค่าเท่ากับการใช้เอชดีเพียงตัวเดียว เนื่องจากเอชดีที่นำมาใช้ในการสุ่มสัญญาณเป็นไอซีเบอร์เดียวกัน โดยเราจะพิจารณาว่าเอชดีทั้งสองตัวมีพารามิเตอร์ต่างๆ เหมือนกันทุกประการ จากตารางที่ 6.4 การใช้ไอซีเอชดีจำนวนสองตัวสลับกันทำงานนั้น สามารถให้ค่า Sampling เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

rate เพิ่มขึ้นเป็นสองเท่า โดยที่ค่า Conversion time ของไอซีเอทวูดีแต่ละตัวยังคงมีค่าเท่าเดิม จะเห็นได้ว่าในกรณีที่ต้องใช้ไอซีเอทวูดีตัวเดียวที่มีความเร็วในการแปลงสัญญาณหรืออัตราการสุ่มที่สูงขึ้นนั้น จำเป็นจะต้องใช้ชิพหน่วยความจำที่มีความเร็วในการเข้าถึงข้อมูลสูงตามไปด้วย ซึ่งอาจจะทำให้เกิดการรบกวนจากสัญญาณความถี่สูงได้ง่าย หลักการใช้ไอซีเอทวูดีสองตัวสลับกันทำงานนี้สามารถช่วยลดปัญหาดังกล่าวได้วิธีหนึ่ง เนื่องจากว่าไอซีเอทวูดีทั้งสองตัวที่นำมาใช้งานยังคงมีค่าความเร็วในการแปลงสัญญาณเท่าเดิม

6.2 ผลของการเปลี่ยนแปลงคาบเวลาในการสุ่ม

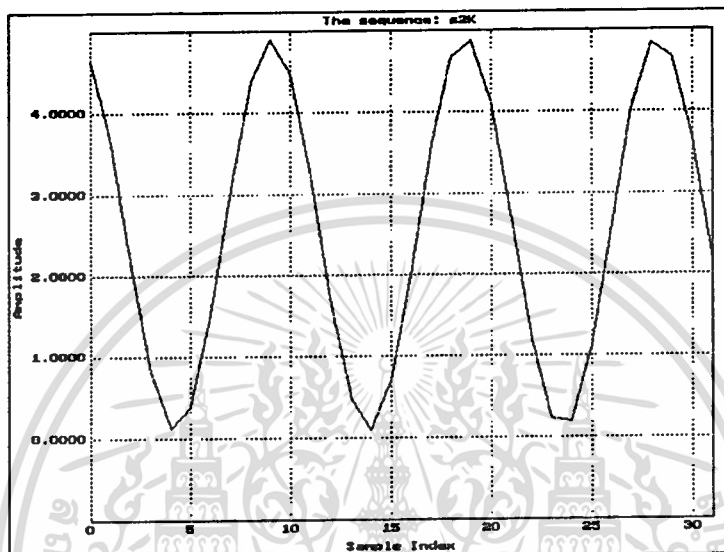
จากการทดลองในหัวข้อ 6.1 เมื่อทำการเปลี่ยนความถี่ของสัญญาณอินพุตเป็น 2kHz, 10kHz, และ 20kHz โดยทำการทดลองทั้งแบบที่ใช้เอทวูดีตัวเดียว และแบบเอทวูดีสองตัวที่มีวงจรควบคุมการสุ่มสัญญาณ ได้สัญญาณตามรูปที่ 6.8 และ 6.9

จากผลการทดลองในรูปที่ 6.8 กับ 6.9 จะเห็นว่าที่สัญญาณอินพุตความถี่ 2kHz วงจรทั้ง 2 แบบสามารถวัดสัญญาณได้เหมือนกัน แต่ที่ความถี่ 10kHz วงจรที่ใช้เอทวูดีเพียงตัวเดียวเริ่มวัดค่าได้ผิดพลาด ส่วนวงจรที่ใช้เอทวูดี 2 ตัว ยังคงสามารถวัดค่าได้ เนื่องจากว่ามีอัตราสุ่มสัญญาณสูงกว่า แต่ที่ความถี่ 20kHz ขึ้นไป วงจรทั้งสองไม่สามารถวัดค่าได้ เพราะจะเกิด Aliasing กับสัญญาณที่วัด เนื่องจากชุดทดลองที่ใช้เอทวูดี 2 ตัว แบบมีวงจรควบคุมการสุ่มสัญญาณ มีอัตราการสุ่มสัญญาณเพียง 40000 Samples/sec เท่านั้น

6.3 ค่าผิดพลาดที่อาจเกิดจากวงจรควบคุมการสุ่มสัญญาณ

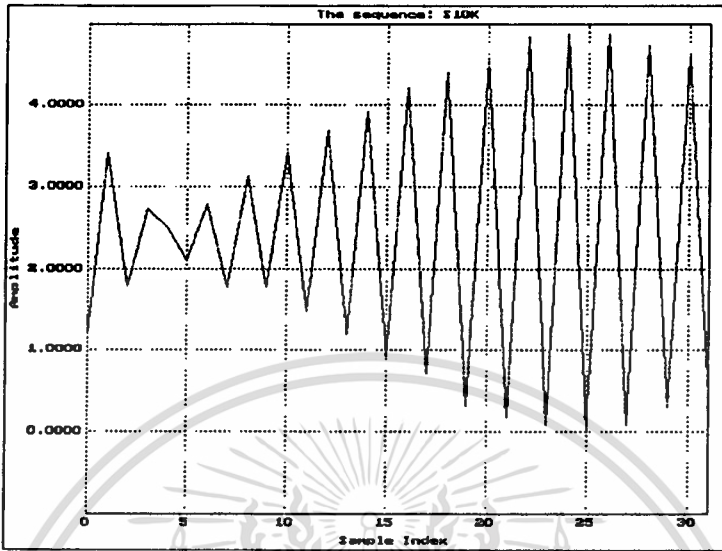
จากวงจรควบคุมคาบเวลาในการสุ่มสัญญาณในรูปที่ 4.7 ถ้าช่วงระยะเวลาในการเริ่มต้นแปลงสัญญาณของเอทวูดีตัวที่ 1 แต่ละช่วง (ช่วงเวลา 2T) มีค่ามากกว่า 32 ลูกสัญญาณนาฬิกา วงจรควบคุมคาบเวลาจะทำงานผิดพลาด สาเหตุเนื่องมาจากการออกแบบวงจรนับมีขนาดเพียง 4 บิตเท่านั้น ดังนั้นวงจรนับ (74LS161) จึงสามารถนับสัญญาณนาฬิกาได้สูงสุดเท่ากับ 2×2^4 เท่ากับ 32 ลูกสัญญาณนาฬิกา ฉะนั้นวงจรที่ได้ทำการทดลองจึงมีขีดจำกัดดังกล่าว ในกรณีที่ต้องการได้ช่วงเวลา 2T มีค่ามากกว่านี้ สามารถ

กระทำได้โดยเพิ่มจำนวนบิตของชุดวงจรมอดูเลชัน เช่นถ้าวงจรมอดูเลชันมีขนาด 8 บิต ช่วงเวลา $2T$ จะมีค่าสูงสุดเท่ากับ 2×2^8 เท่ากับ 512 ลูกสัญญาณนาฬิกา เป็นต้น

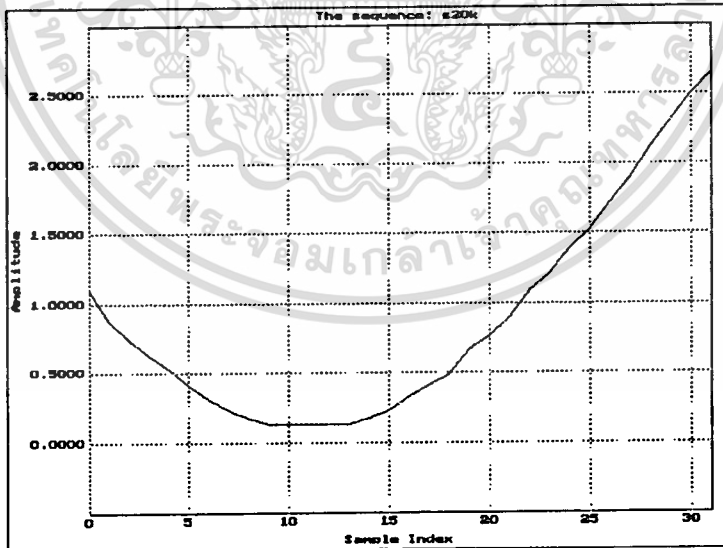


ความถี่ 2 kHz

รูปที่ 6.8 ผลการทดลองสุ่มสัญญาณ โดยใช้เฮอทุติตัวเดียว



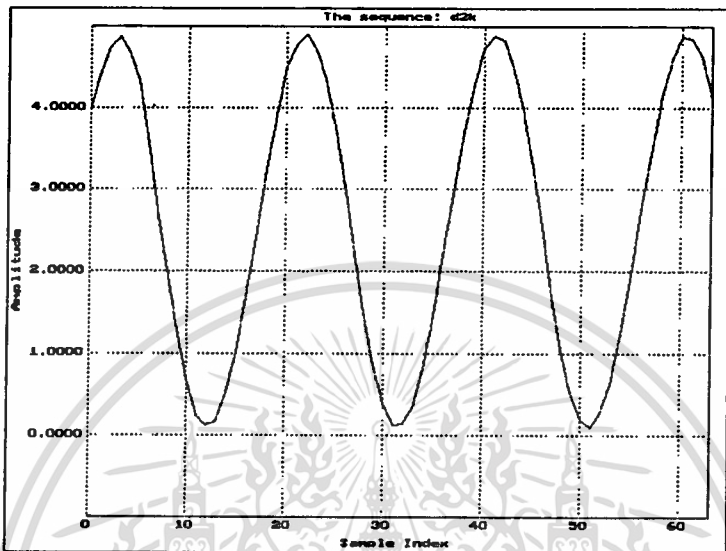
ความถี่ 10 kHz



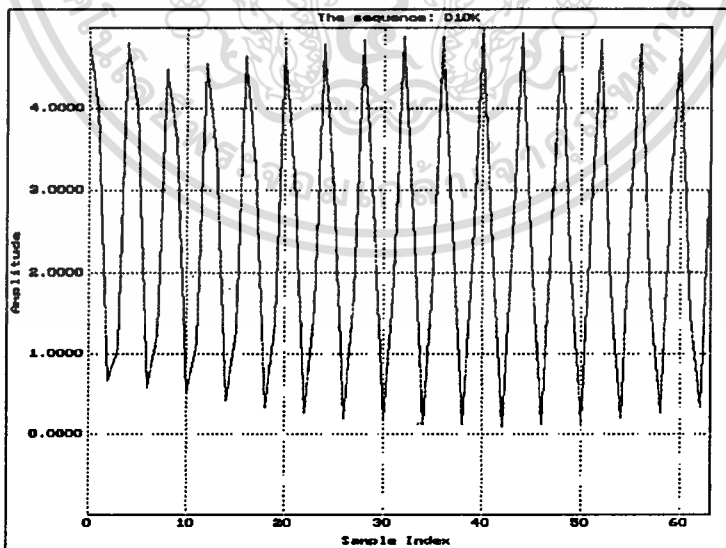
ความถี่ 20 kHz

รูปที่ 6.8 (ต่อ) ผลการทดลองสุ่มสัญญาณ โดยใช้เอทูดิวเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ความถี่ 2 kHz



ความถี่ 10 kHz

รูปที่ 6.9 ผลการทดลองสุ่มสัญญาณโดยใช้เหตุตีสองตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 บทสรุป

จากผลการทดลองข้างต้น พบสรุปได้ว่าถ้าอัตราการสุ่มสัญญาณมีค่าสูงขึ้นก็จะสามารถวัดสัญญาณอินพุตที่มีความถี่สูงขึ้นได้ด้วย วิธีการใช้ไอซีเอทวูตีสองตัว ต่อขนานกัน และให้ทำงานในลักษณะที่เหลื่อมล้ำกัน (Time-Interleaving) โดยมีวงจรควบคุมคาบเวลาในการสุ่มสัญญาณของเอทวูตีสองตัวที่ 2 เพื่อให้ได้ช่วงระยะเวลาสำหรับการสุ่มสัญญาณแต่ละครั้งมีค่าเท่า ๆ กัน (เท่ากับ T) นั้น สามารถทำให้ได้อัตราการสุ่มสัญญาณเพิ่มขึ้นจริง และสามารถนำค่าสัญญาณที่ได้ไปประมวลผลตามหลักคณิตศาสตร์ของ Discrete Fourier Transform ได้ เนื่องจากช่วงระยะเวลาในการสุ่มสัญญาณแต่ละครั้งมีค่าเท่ากัน สัญญาณที่วัดได้จึงมีค่าเหมือนกับการใช้ไอซีเอทวูตีสองตัวเดี่ยวที่อัตราการสุ่มค่าเดียวกัน และการที่ได้อัตราการสุ่มสัญญาณเพิ่มขึ้นนี้จะไม่มีความกระทบในเรื่องความเร็วในการเข้าถึงข้อมูลของชิพหน่วยความจำที่ใช้อยู่เลย เนื่องจากว่าไอซีเอทวูตีสองตัวที่นำมาใช้งานยังคงมีค่าความเร็วในการแปลงสัญญาณเท่าเดิม จากหลักการดังกล่าวสามารถนำไปพัฒนาเพื่อเพิ่มขีดความสามารถในการแปลงสัญญาณให้กับไอซีเอทวูตีสองตัวประเภท Successive Approximation ได้เป็นอย่างดี ทั้งยังสามารถนำไปใช้กับไอซีเอทวูตีสองตัวประเภท Flash Converter ได้อีกด้วย

เอกสารอ้างอิง:

- [1] R. L. Geiger, P. E. Allen, Noel R. Strader, "VLSI design techniques for analog and digital circuits," McGRAW-HILL, 1990
- [2] Y.C.Jenq, "Digital spectra of nonuniformly sampled signals: Fundamentals and high speed waveform digitizers," IEEE Trans. Instrum. Meas., vol.37, pp.245-251, June 1988
- [3] R. L. Krutz, "Interfacing Techniques In Digital Design," John Wiley & Sons, 1988
- [4] W.J.Tompkins, J.G.Webster, "Interfacing sensors to the IBM PC," Prentice-Hall, 1988
- [5] S. E. Derenzo, "Interfacing," Prentice-Hall, 1988
- [6] Oktay Alkin, "Digital Signal Processing A Laboratory Approach Using PC-DSP", Prentice-Hall, 1994

ผลงานที่ได้รับการตีพิมพ์

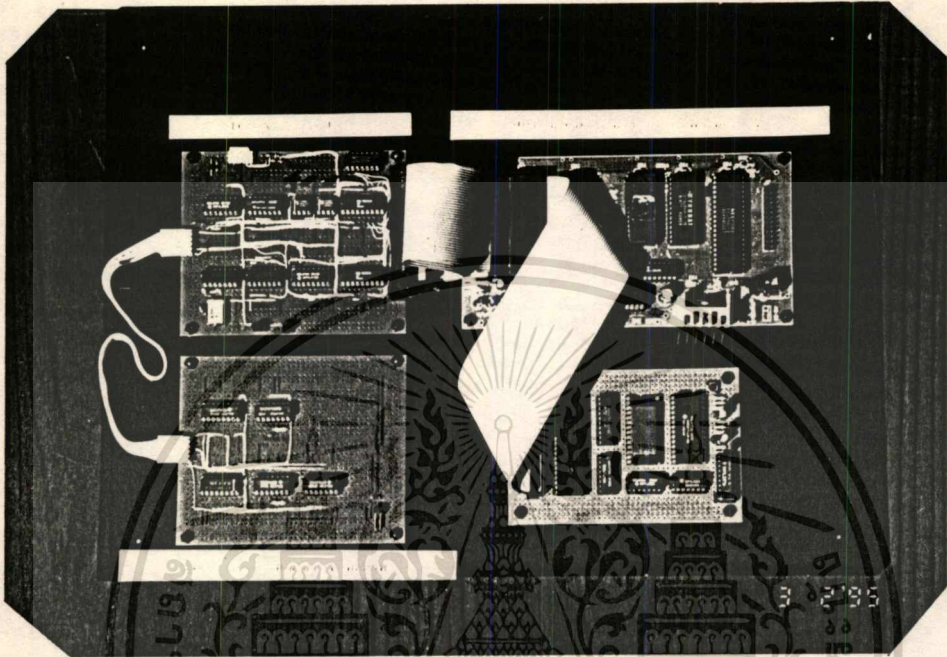
- [1] กอบชัย เศษหาญ, จิระศักดิ์ ชาญวุฒิชธรรม, อิศราพงศ์ สิ้นันตา "ไมโครคอนโทรลเลอร์ในระบบควบคุมแบบแยกส่วนผ่านพอร์ทอนุกรมเพื่องานควบคุมในโรงงานขนาดเล็ก" การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 15, สถาบันเทคโนโลยีพระจอมเกล้า ธนบุรี, 2535
- [2] กอบชัย เศษหาญ, จิระศักดิ์ ชาญวุฒิชธรรม, ศวุฒ บุษยอังกูร "เทคนิคการประมวลผลใช้กับเครื่องบันทึกข้อมูลระบบดิจิทัล โดยใช้ไลดัส 1-2-3" จะตีพิมพ์ในวารสารลาดกระบัง





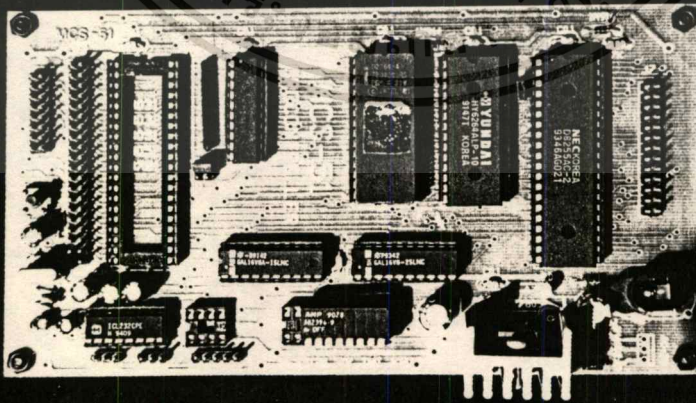
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
ฮาร์ดแวร์ที่ใช้ในการทดลอง



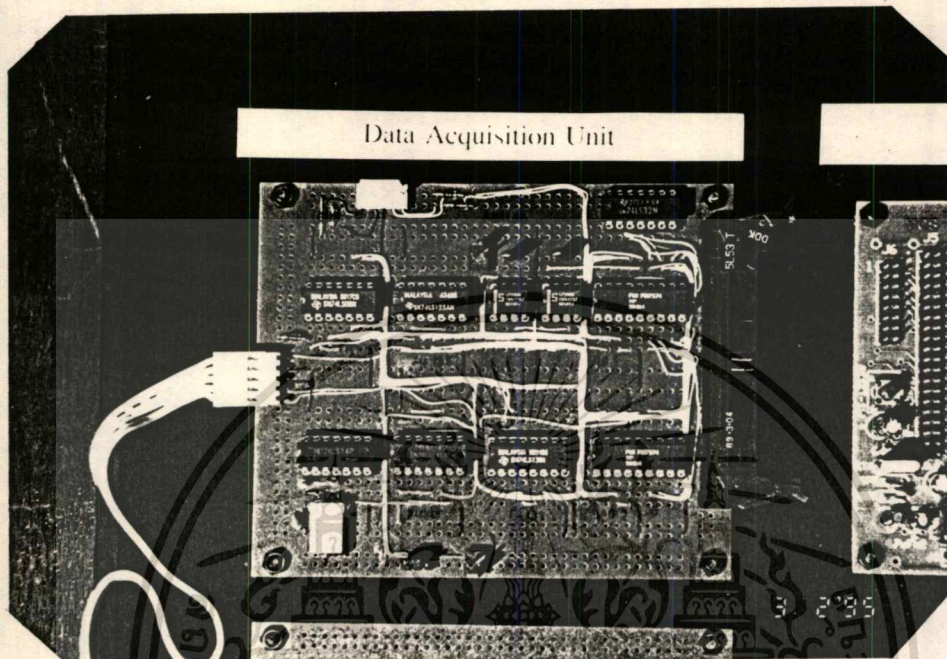
แผ่นวงจรที่ใช้ในการทดลอง

8031 ไมโครคอนโทรลเลอร์ และพอร์ตอนุกรม

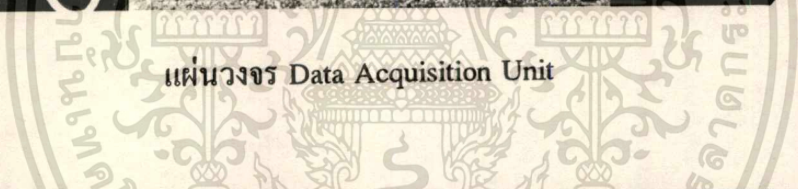


แผ่นวงจร 8031 ไมโครคอนโทรลเลอร์และพอร์ตอนุกรม

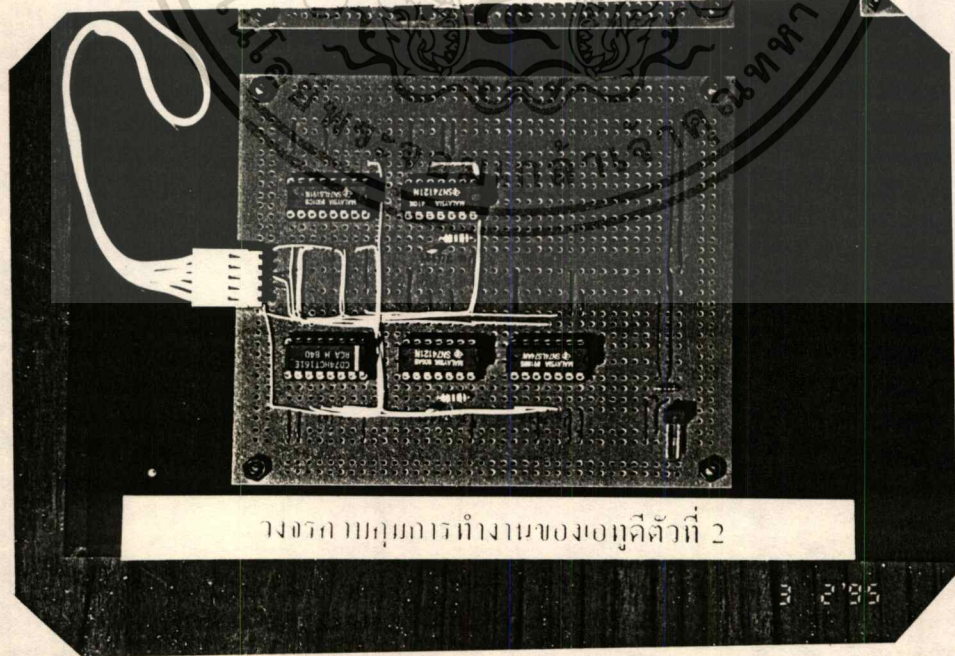
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Data Acquisition Unit



แผ่นวงจร Data Acquisition Unit



วงจรถามุมการทำงานของเอทิวิตีตัวที่ 2

แผ่นวงจรควบคุมการทำงานของเอทิวิตีตัวที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลไอซีเอพดี AD7574



PM-7574

CMOS MICROPROCESSOR-COMPATIBLE
8-BIT A/D CONVERTER

Precision Monolithics Inc.

FEATURES

- 8-Bit Resolution and Accuracy
- No Missing Codes Over Full Temperature Range
- 15µs Conversion Time
- Flexible µP Interface
- 5mA Maximum Standby Current
- Low Cost
- Pin and Function Compatible With AD7574
- Available In Die Form

ORDERING INFORMATION †

PACKAGE: 18-PIN DIP AND SO				
INL (LSB)	DNL (LSB)	MILITARY* TEMPERATURE -55°C to +125°C	EXTENDED INDUSTRIAL TEMPERATURE -40°C to +85°C	COMMERCIAL TEMPERATURE 0°C to +70°C
±1/2	±3/4	PM7574AX	PM7574EX	PM7574GP
±3/4	±7/8	PM7574BX	PM7574FX	-
±3/4	±7/8	-	PM7574FP	-
±3/4	±7/8	-	PM7574FS	-

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

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

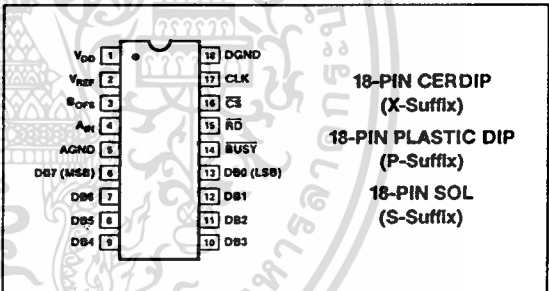
select (\overline{CS}) and read/write (\overline{RD}) inputs are used to control all operations. This simplicity permits the PM-7574 to be used as a memory-mapped input device. Depending on the control timing waveforms, the PM-7574 is interfaced like static RAM, ROM, or slow memory.

The low power consumption of the PM-7574 is derived from a single +5V supply. A negative reference voltage must also be supplied. Optimum accuracy is achieved when the reference is at -10.00V with low output resistance. For a low-cost precision -10V/-10.24V reference, ask your PMI sales representative about the REF-08.

With its on-board comparator, interface logic, optional internal clock, and +5V operation, the PM-7574 is the ideal low-cost solution for microprocessor-based 8-bit A/D systems.

For new designs, PMI's ADC-908 is pin-and-function compatible with the PM-7574, but offers faster conversion time and faster microprocessor bus interface timing.

PIN CONNECTIONS

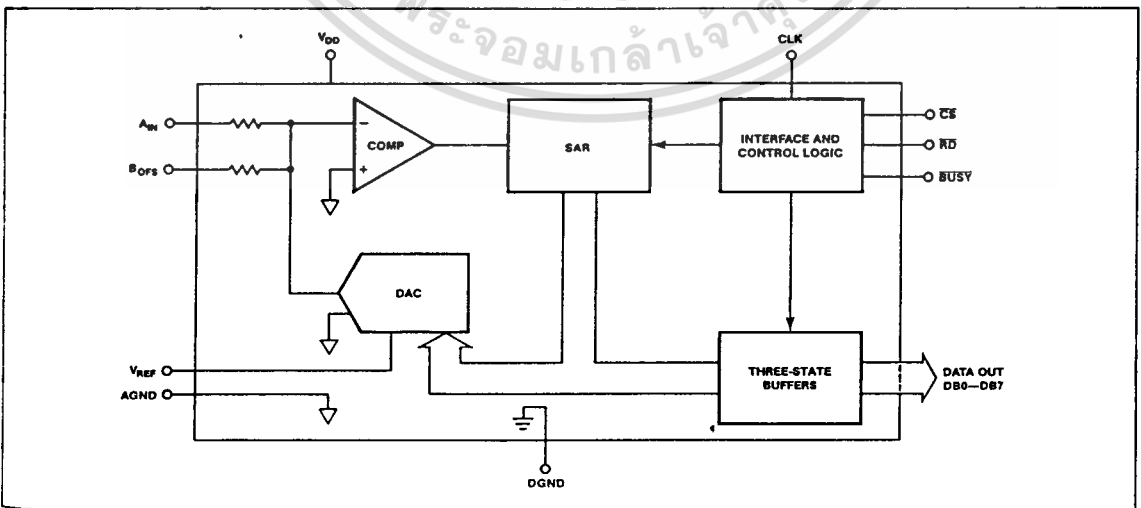


GENERAL DESCRIPTION

The PM-7574 is a monolithic CMOS successive-approximation analog-to-digital converter. When used with a 550kHz clock, a conversion time of 15µs is achieved, with full accuracy over the operating temperature range.

The PM-7574 outputs use 3-state logic, allowing direct connection to the data bus or system input port. Active-LOW chip

FUNCTIONAL DIAGRAM



**ABSOLUTE MAXIMUM RATINGS** ($T_A = +25^\circ\text{C}$, unless otherwise noted.)

V_{DD} to AGND	0V, +7.0V
V_{DD} to DGND	0V, +7.0V
AGND to DGND	-0.3V, V_{DD}
CS, RD to DGND	-0.3V, $V_{DD} + 0.3V$
DB ₀ - DB ₇ to DGND	-0.3V, V_{DD}
CLK, BUSY to DGND	-0.3V, V_{DD}
B _{OFS} , A _{IN}	$\pm 20V$
V_{REF}	0V, -20V
Operating Temperature Range	
PM-7574AX, BX	-55°C to +125°C
PM-7574EX, FX, FP, FS	-40°C to +85°C
PM-7574GP	0°C to +70°C

Storage Temperature -65°C to +150°C
 Lead Temperature (Soldering, 10 sec) +300°C

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

NOTES:

- Digital pins are zener protected. However, proper ESD handling precautions are recommended.
- θ_{JA} is specified for worst case mounting conditions, i.e., θ_{JA} is specified for device in socket for CerDIP and P-DIP packages; θ_{JA} is specified for device soldered to printed circuit board for SOL package.

ELECTRICAL CHARACTERISTICS at $V_{DD} = +5V$, $V_{REF} = -10V$, Unipolar Configuration, $R_{CLK} = 150k\Omega$, $C_{CLK} = 100pF$; $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$ for PM-7574E/F, $0^\circ\text{C} < T_A \leq +70^\circ\text{C}$ for PM-7574G, $-55^\circ\text{C} \leq T_A \leq +125^\circ\text{C}$ for PM-7574A/B, unless otherwise noted.

PARAMETER	SYMBOL	CONDITIONS	PM-7574			UNITS
			MIN	TYP	MAX	
ACCURACY						
Resolution	N		8	—	—	Bits
Integral Nonlinearity	INL	A/E/G Grades	-1/2	—	+1/2	LSB
		B/F Grades	-3/4	—	+3/4	
Differential Nonlinearity	DNL	A/E/G Grades	-3/4	—	+3/4	LSB
		B/F Grades	-7/8	—	+7/8	
Gain Error	G _{FSE}	A/E/G Grades $T_A = +25^\circ\text{C}$	-3	—	+3	LSB
		$T_A = \text{Full Temp Range}$	-4.5	—	+4.5	
		B/F Grades $T_A = +25^\circ\text{C}$	-5	—	+5	
		$T_A = \text{Full Temp Range}$	-6.5	—	+6.5	
Offset Error	V _{ZSE}	A/E/G Grades $T_A = +25^\circ\text{C}$	-30	—	+30	mV
		$T_A = \text{Full Temp Range}$	-50	—	+50	
		B/F Grades $T_A = +25^\circ\text{C}$	-60	—	+60	
		$T_A = \text{Full Temp Range}$	-80	—	+80	
ANALOG INPUTS						
Resistance Mismatch B _{OFS} to A _{IN}	ΔR_{AB}		-1.5	—	+1.5	%
Input Resistance at V_{REF} (Note 1)	R_{REF}		5	—	15	k Ω
Input Resistance at B _{OFS} , A _{IN}	R_{BOFS}		10	—	30	k Ω
	R_{IN}					
Reference Voltage	V_{REF}	Specified Conversion Accuracy	—	-10	—	V
Reference Voltage Range	V_{REF}	Degraded Conversion Accuracy	-5	—	-15	V
Reference Current (Note 6)	I_{REF}	Conversion Complete Prior to Reset.	—	—	2.4	mA
Nominal Analog Input Range	V_{INU}		—	0 to + V_{REF1}	—	V
	V_{INB}		—	- V_{REF1} to + V_{REF1}	—	
LOGIC INPUTS						
Input HIGH Voltage RD, CS Inputs	V_{IH}		2.4	—	—	V
Input LOW Voltage RD, CS Inputs	V_{IL}		—	—	0.8	V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



PM-7574 CMOS MICROPROCESSOR-COMPATIBLE 8-BIT A/D CONVERTER

ELECTRICAL CHARACTERISTICS at $V_{DD} = +5V$, $V_{REF} = -10V$, Unipolar Configuration, $R_{CLK} = 150k\Omega$, $C_{CLK} = 100pF$; $-40^\circ C \leq T_A \leq +85^\circ C$ for PM-7574E/F, $0^\circ C < T_A \leq +70^\circ C$ for PM-7574G, $-55^\circ C \leq T_A \leq +125^\circ C$ for PM-7574A/B, unless otherwise noted.

Continued

PARAMETER	SYMBOL	CONDITIONS	PM-7574			UNITS
			MIN	TYP	MAX	
Input Current RD, CS Inputs	I_{IN}	$T_A = +25^\circ C$ $T_A = \text{Full Temp Range}$	—	—	1 10	μA
Input Capacitance RD, CS Inputs (Note 6)	C_{IN}		—	—	5	pF
Input HIGH Voltage, Clock Input	V_{IH}		3	—	—	V
Input LOW Voltage, Clock Input	V_{IL}		—	—	0.4	V
Input HIGH Current, Clock Input	I_{IH}		—	—	2	mA
Input LOW Current, Clock Input	I_{IL}	$T_A = +25^\circ C$ $T_A = \text{Full Temp Range}$	—	—	1 10	μA
LOGIC OUTPUTS						
Output HIGH Voltage BUSY, DB0-7	V_{OH}	$I_{SOURCE} = 40\mu A$	4.0	—	—	V
Output LOW Voltage BUSY, DB0-7	V_{OL}	$I_{SINK} = 1.6mA$	—	—	0.4	V
Floating Leakage Current, DB0-7	I_{LKG}	$T_A = +25^\circ C$ $T_A = \text{Full Temp Range}$	—	—	1 10	μA
Floating State Output Capacitance		(Note 6)	—	—	7	pF
POWER REQUIREMENTS						
Standby Current	I_{DD}	$V_{DD} = +4.75V \text{ to } +5.25V$	—	—	5	mA
DIGITAL INTERFACE TIMING						
CS Minimum Pulse Width (Note 6)	t_{CS}	$T_A = +25^\circ C$ $T_A = \text{Full Temp Range}$	100 150	—	—	ns
RD to CS Setup Time (Note 6)	t_{WCS}		0	—	—	ns
CS to BUSY Propagation Delay (Note 6)	t_{CBPD}	BUSY Load = 20pF $T_A = +25^\circ C$ $T_A = T_{MIN}$ $T_A = T_{MAX}$ BUSY Load = 100pF $T_A = +25^\circ C$ $T_A = T_{MIN}$ $T_A = T_{MAX}$	— — — — — — —	— — — — — — —	120 120 180 150 150 200	ns
BUSY to RD Setup Time (Notes 2, 6)	t_{BSR}		0	—	—	ns
BUSY to CS Setup Time (Note 6)	t_{BSCS}		0	—	—	ns
Data Access Time (Note 6)	t_{RAD}	$C_L = 20pF$ $T_A = +25^\circ C$ $T_A = T_{MIN}$ $T_A = T_{MAX}$ $C_L = 100pF$ $T_A = +25^\circ C$ $T_A = T_{MIN}$ $T_A = T_{MAX}$	— — — — — — —	— — — — — — —	150 150 220 300 300 400	ns
Data Hold Time (Notes 3, 6)	t_{RHD}	$T_A = +25^\circ C$ (Note 3) $T_A = T_{MIN}$ $T_A = T_{MAX}$	50 30 80	— — —	120 80 140	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



PM-7574 CMOS MICROPROCESSOR-COMPATIBLE 8-BIT A/D CONVERTER

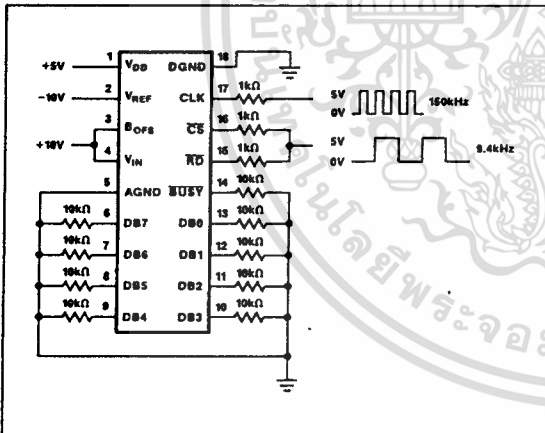
ELECTRICAL CHARACTERISTICS at $V_{DD} = +5V$, $V_{REF} = -10V$, Unipolar Configuration, $R_{CLK} = 150k\Omega$, $C_{CLK} = 100pF$; $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ for PM-7574E/F, $0^{\circ}C < T_A \leq +70^{\circ}C$ for PM-7574G, $-55^{\circ}C \leq T_A \leq +125^{\circ}C$ for PM-7574A/B, unless otherwise noted.
Continued

PARAMETER	SYMBOL	CONDITIONS	PM-7574			UNITS
			MIN	TYP	MAX	
\overline{CS} to \overline{RD} Hold Time (Note 6)	t_{RHCS}	$T_A = +25^{\circ}C$	—	—	250	ns
		$T_A = T_{MIN}$	—	—	200	
		$T_A = T_{MAX}$	—	—	500	
Reset Time Requirement (Note 6)	t_{RESET}		3	—	—	μs
Conversion Time (Note 4) (Notes 4, 5, 6)	$t_{CONVERT}$	Static RAM Mode External Clock $f = 550kHz$	—	—	15	μs
		ROM Mode Internal Clock	—	—	17	
			—	—	—	
\overline{RD} HIGH to \overline{BUSY} Propagation Delay, ROM Mode (Notes 4, 5, 6)	t_{WBPD}	$C_L = 20pF$ $T_A = +25^{\circ}C$ $T_A = T_{MIN}$ $T_A = T_{MAX}$	—	—	1.5 1.0 2.0	μs

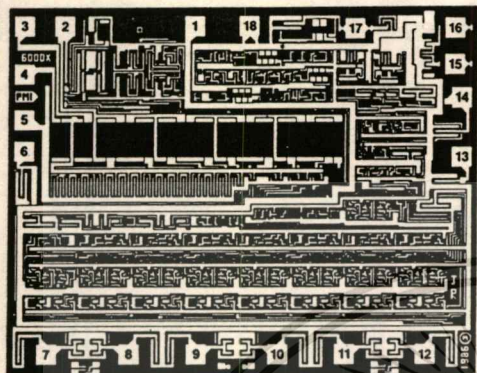
NOTES:

- For optimum gain accuracy over the full temperature range, the source resistance at pin 2 should be kept low.
- In ROM mode, \overline{RD} can go LOW prior to $\overline{BUSY} = HIGH$, but must not return HIGH until $\overline{BUSY} = HIGH$.
- Output loading 10pF. A 3k Ω pullup resistor to +5V is used for V_{OL} to High-Z; for V_{OH} to High-Z, a 3k Ω pulldown to GND is used. Measured to 0.5V output change.
- When using the PM-7574 internal oscillator, actual conversion time depends on clock resistor and capacitor as well as temperature.
- ROM interface mode conversion times are typically 2 μs longer than conversion times for other modes, but the ROM interface mode includes an automatic reset in the conversion time.
- Guaranteed but not tested.

BURN-IN CIRCUIT



DICE CHARACTERISTICS



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

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

For additional DICE ordering information, refer to PMI's Data Book, Section 2.

WAFER TEST LIMITS at $V_{DD} = +5V$, $V_{REF} = -10.000V$, AGND = DGND = 0V, $T_A = +25^\circ C$, unless otherwise noted.

PARAMETER	SYMBOL	CONDITIONS	PM-7574G LIMIT	UNITS
STATIC ACCURACY				
Resolution	N		8	Bits MIN
Integral Nonlinearity	INL		$\pm 3/4$	LSB MAX
Differential Nonlinearity	DNL		$\pm 7/8$	LSB MAX
Gain Error	G_{FSE}		± 5	LSB MAX
Offset Error	V_{ZSE}		± 60	mV MAX
ANALOG INPUTS				
Resistance Mismatch B_{OFS} to A_{IN}	ΔR_{AB}		± 1.5	% MAX
Input Resistance at V_{REF}	R_{REF}		5/15	k Ω MIN/MAX
Input Resistance at B_{OFS} , A_{IN}	$R_{B_{OFS}}$, R_{IN}		10/30	k Ω MIN/MAX
DIGITAL INPUTS				
Input HIGH Voltage at RD, CS Inputs	V_{IH}		3	V MIN
Input LOW Voltage at RD, CS Inputs	V_{IL}		0.8	V MAX
Input Current RD, CS Inputs	I_{IN}		± 1	μA MAX
Input HIGH Voltage Clock Input	V_{IH}		3	V MIN
Input LOW Voltage Clock Input	V_{IL}		0.4	V MAX
Input HIGH Current Clock Input	I_{IH}		2	mA MAX
Input LOW Current Clock Input	I_{IL}		1	μA MAX

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

PARAMETER	SYMBOL	CONDITIONS	PM-7574G	
			LIMIT	UNITS
DIGITAL OUTPUTS				
Output HIGH Voltage BUSY, DB0-7	V_{OH}	$I_{SOURCE} = 40\mu A$	4	V MIN
Output LOW Voltage BUSY, DB0-7	V_{OL}	$I_{SINK} = 1.6mA$	0.4	V MAX
Floating Leakage Current	I_{LKG}		1	μA
POWER REQUIREMENTS				
Standby Current	I_{DD}	$V_{DD} = +4.75V$ to $5.25V$	5	mA MAX
TIMING				
Conversion Time	$t_{CONVERT}$	Static RAM Mode, External Clock, $f = 550kHz$	15	μs MAX

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

GENERAL CIRCUIT INFORMATION

The PM-7574 is an 8-bit analog-to-digital converter which uses a successive approximation technique to convert an unknown analog input into a digital code output. The control logic inputs allow easy interface to most microprocessors while three-state outputs allow direct connection to the data bus. Most applications require only passive RC clock components, a -10V reference, and a +5V power supply. The RC-timed internal clock may be used, or an external clock may be applied to the ADC to maximize performance.

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

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

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

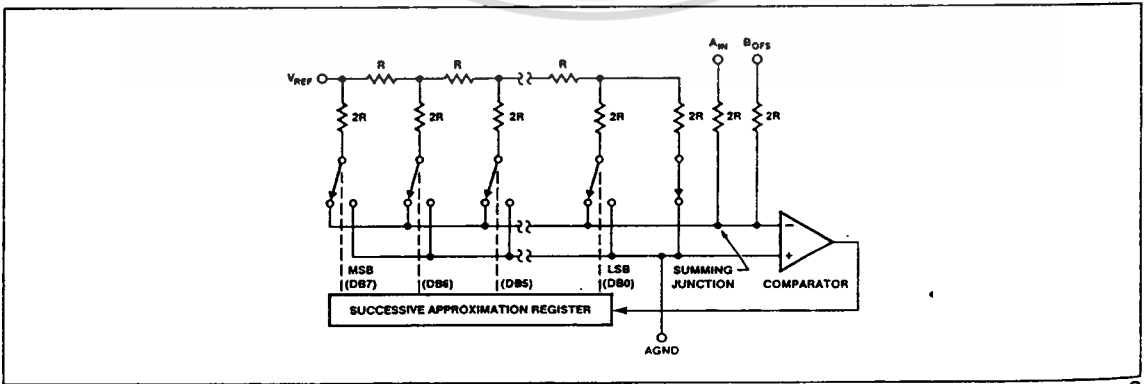
APPLICATIONS INFORMATION

The PM-7574 may be interfaced as if it were a static RAM, a ROM, or a slow-memory device. Each of these interface modes has its own timing and software requirements as described below. These requirements must be rigidly met, as improper timing may cause the PM-7574 to change modes.

HOW TO CHOOSE AN OPERATING MODE

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

FIGURE 1: D/A Converter Used in PM-7574



resulting software bugs may result in converted data of unknown age, or altogether invalid data being read.

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

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

OPERATING DESCRIPTION: STATIC-RAM MODE

In this mode, input CS is derived from the PM-7574 address decoder, and input RD is derived from an active-LOW memory READ signal. (See Figure 2.)

To start a conversion, execute a memory WRITE to the PM-7574. The completed conversion data is obtained by executing a memory READ to the PM-7574. During conversion, output BUSY will be LOW. Do not attempt to read data until BUSY returns HIGH. The required minimum time between WRITE and READ is usually obtained by including one or more NOP or other program instructions. The use of branch or conditional

commands between the WRITE and READ instructions is not recommended due to the possibility of software bugs.

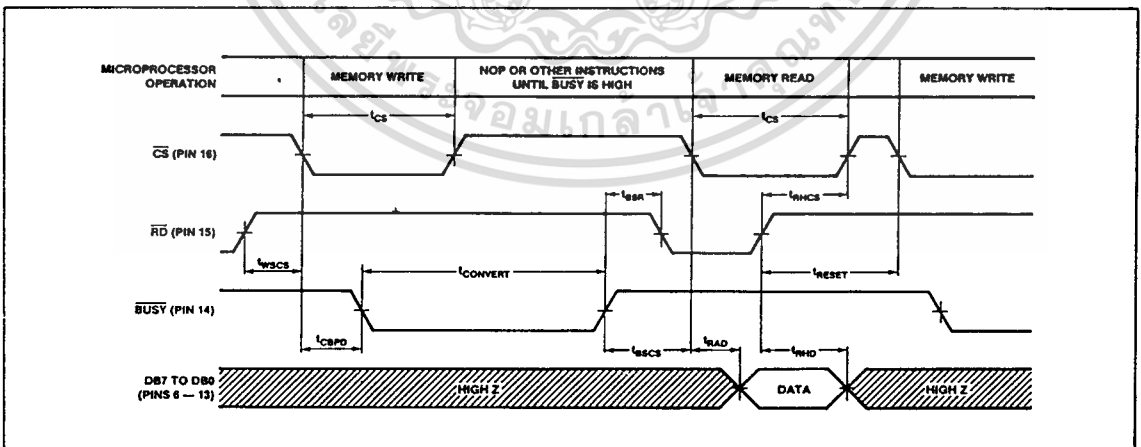
It is important that the WRITE and READ commands be alternately executed. A WRITE instruction has no effect unless the results of the previous WRITE have already been read. Once data has been read, the PM-7574 is internally reset. In other words, two or more READ operations cannot be used in succession, since only the first READ will produce valid data. A new conversion must be started using WRITE, and the conversion must be completed, before a new READ will produce valid data.

TABLE 1: Truth Table, Static RAM Mode

INPUTS		OUTPUTS		PM-7574 OPERATION
CS	RD	BUSY	DB7-DB0	
\bar{L}	H	\bar{L}	HIGH-Z	Start Convert (Write Cycle)
L	\bar{L}	H	HIGH-Z to DATA	Read Data (Read Cycle)
L	\bar{H}	H	DATA to HIGH-Z	Reset Converter
H	X (Note 1)	X	HIGH-Z	No Effect (Not Selected)
L	H	L	HIGH-Z	No Effect (Converter Busy)
L	\bar{L}	L	HIGH-Z	No Effect (Converter Busy)
L	\bar{H} (Note 1)	L	HIGH-Z	Conversion Error Not Allowed

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

FIGURE 2: Static RAM Mode Timing Diagram



OPERATING DESCRIPTION: ROM MODE

In ROM mode, input \overline{CS} is tied LOW, and input \overline{RD} is derived from the PM-7574 address decoder. To satisfy timing, it is recommended that the decoder be enabled by a system MEMRD (8080), VMA (6800), or similar strobe. (See Figure 3.)

TABLE 2: Truth Table, ROM Mode

INPUTS		OUTPUTS		PM-7574 OPERATION
\overline{CS}	\overline{RD}	BUSY	DB7-DB0	
L	\downarrow	H	HIGH-Z to DATA	Read Data
L	\uparrow	\downarrow	DATA to HIGH-Z	Reset and Start New Conversion
L	\downarrow	L	HIGH-Z	No Effect (Converter Busy)
L	\uparrow (Note 1)	L	HIGH-Z	Conversion Error Not Allowed

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

In ROM mode, data is read by executing a READ instruction to the PM-7574 address. At the conclusion of the READ instruction, the PM-7574 automatically resets itself and then proceeds to perform a new data conversion. Output BUSY is LOW during conversion. A new READ instruction to the PM-7574 must not be executed until BUSY returns HIGH. This requirement may be met by inserting NOP or other program instructions between consecutive READ operations. Conditional or branch instructions may be used, but keep in mind that data may become out-of-date if excessive time elapses between consecutive READ instructions.

OPERATING DESCRIPTION: SLOW-MEMORY MODE

The slow-memory mode is intended for systems in which the PM-7574 BUSY output is used as an interrupt to force the microprocessor into WAIT states during data conversion.

In slow-memory mode, inputs \overline{CS} and \overline{RD} are tied together. The common \overline{RD} and \overline{CS} signal is derived from the PM-7574 address decoder. To satisfy the timing requirements, it is advisable to latch the address using ALE (8085) or SYNC (8080). For 8080 or 8085-based systems, connect the microprocessor READY input to the PM-7574 BUSY output. (See Figure 4.)

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

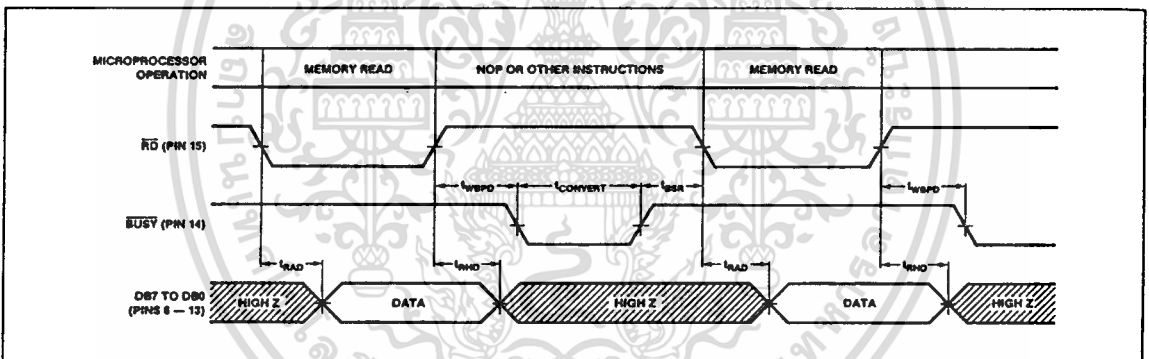


FIGURE 4: Slow-Memory Mode Timing Diagram (\overline{CS} and \overline{RD} Tied Together)

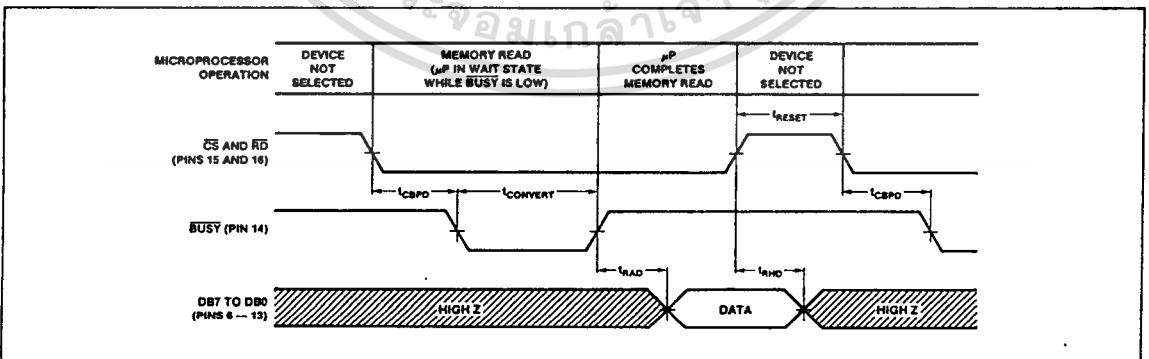


TABLE 3: Truth Table, Slow-Memory Mode

INPUTS		OUTPUTS		PM-7574 OPERATION
\overline{CS} & \overline{RD}	BUSY	DB7-DB0		
H	H	HIGH-Z		No Effect (Not Selected)
		HIGH-Z		Start Conversion
L	L	HIGH-Z		Conversion in Progress. μ P in WAIT State
L		HIGH-Z to DATA		Conversion Complete. Read Data
	H	DATA to HIGH-Z		Reset and Deselect Converter

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

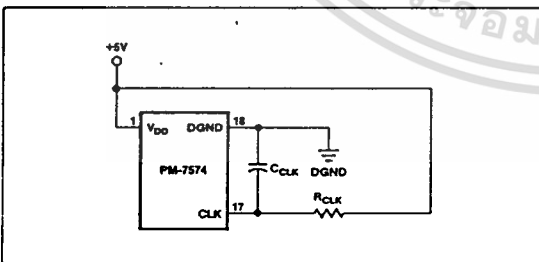
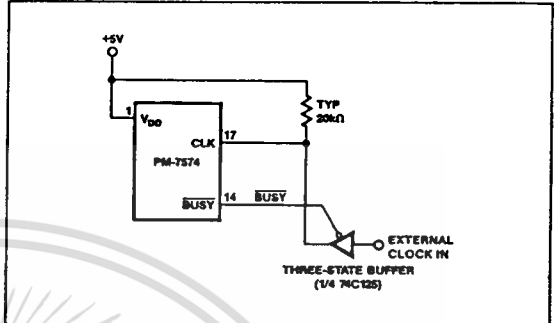
Do not execute a WRITE instruction at the PM-7574 address when in slow-memory mode, since bus conflicts will arise. In some architectures, an accidental WRITE instruction may be locked out in hardware, by proper strobing of the PM-7574 address decoder.

INITIALIZATION

In all operating modes, the PM-7574 is initialized by executing a READ instruction to the PM-7574 address. The data obtained should be ignored.

CLOCK OSCILLATOR

The PM-7574 may be used with its internal asynchronous clock oscillator. An external resistor and capacitor are required. Typical values are $R = 150k\Omega$ and $C = 100pF$, for conversion times in the $15\mu s$ range. For applications in which the fastest conversion times are required, an external clock is recommended. The external clock must be gated by the use of a 74C125-type three-state buffer, with an output pullup resistor. Optimum conversion accuracy is obtained when \overline{CS} goes LOW on a positive clock edge. The maximum external clock frequency is 550kHz. (See Figure 5 and 6.)

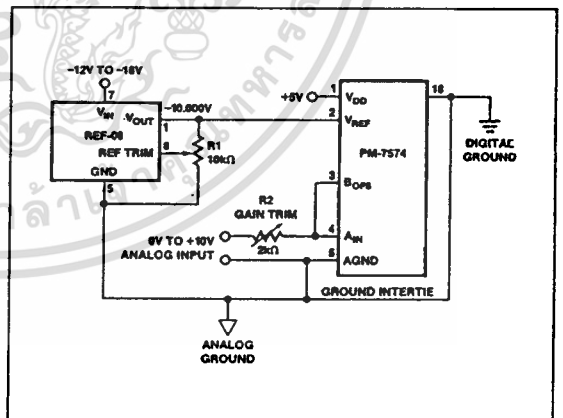
FIGURE 5: Using the Internal Clock Oscillator

FIGURE 6: Using an External Clock


REFERENCE VOLTAGE

A negative reference voltage must be applied to the PM-7574 V_{REF} input. Optimum full-scale accuracy is obtained using $-10.00V$, although V_{REF} may be $-5.00V$, $-10.24V$, or other voltages within its specified range.

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

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

FIGURE 7: Unipolar Operation


ANALOG INPUT VOLTAGE

The PM-7574 unipolar operation is obtained when the analog input voltage is between 0V and $|V_{REF}|$. With the A_{IN} and B_{OFS} pins tied together, input 0V will correspond to code 0000 0000, and input full-scale will correspond to code 1111 1111.

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

Another method of obtaining bipolar operation is to use an op amp with gain = -1/2, to sum the analog signal with the reference voltage. With a -10V reference and -10V to +10V analog signal, the op amp output will then be 0V to +10V. This signal is then treated as an ordinary unipolar input to the PM-7574. With this arrangement, input +10V corresponds to code 0000 0000, and negative full-scale corresponds to code 1111 1111.

UNIPOLAR BINARY OPERATION

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

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

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

To adjust offset:

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

Following offset calibration, full scale gain can be calibrated:

- 1) Apply -9.961V to the input of the buffer amplifier.
- 2) While performing continuous conversions, adjust the reference trim pot until DB7 to DB1 are HIGH, and DB0 (LSB) flickers.

BIPOLAR OPERATION

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

To calibrate this circuit:

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

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

In calibrating this circuit, adjust offset before gain.

Offset Adjustment:

- 1) Adjust R1 until $V_{REF} = -10.000V$.

FIGURE 8: Offset Binary Operation

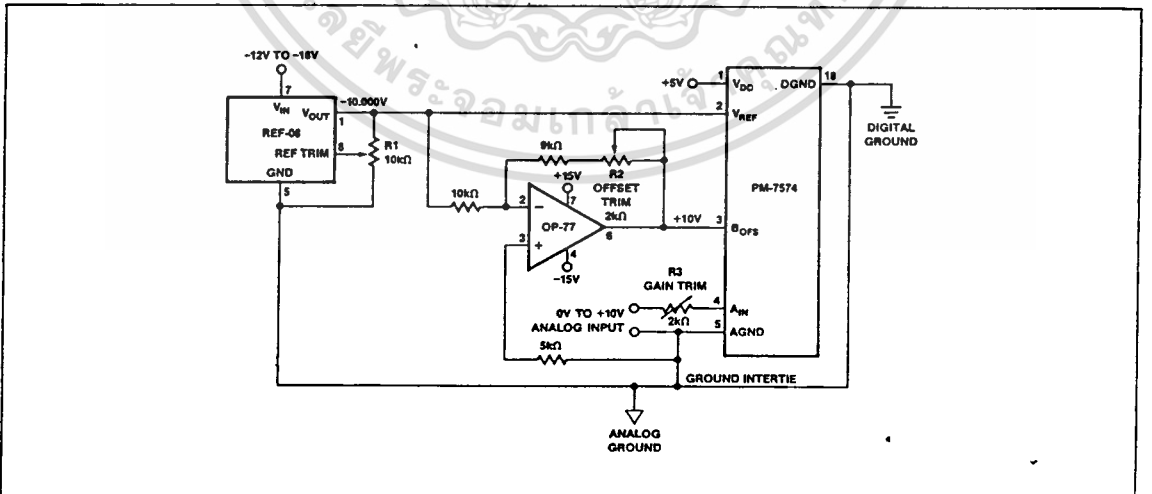
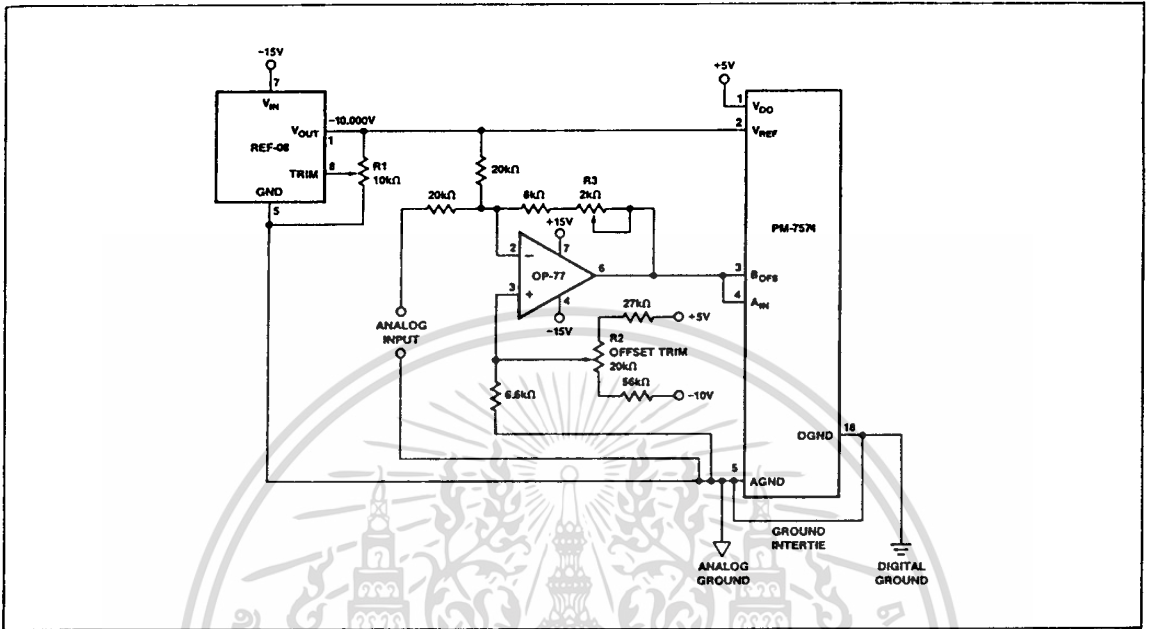


FIGURE 9: Complementary Offset Bipolar Operation



- 2) Adjust R3 to its mid-point.
- 3) Adjust R2 until its tap is at 0V.
- 4) Ground the analog input.
- 5) While performing continuous conversions, adjust R2 until the ADC output flickers between 0111 1111 and 1000 0000.

Gain Adjustment:

- 1) Apply +9.922V across the analog input.
- 2) While performing continuous conversions, adjust R3 until DB7 to DB0 (LSB) flickers.

DIGITAL CONSIDERATIONS

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

The timing restrictions detailed in the interface timing diagrams must be observed to prevent the PM-7574 from changing interface modes. For example, if \overline{CS} is held LOW too long while in RAM mode, the converter will change to ROM mode and initiate a new conversion.

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

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

ANALOG CONSIDERATIONS

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

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

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

Ratiometric Operation—The R-2R type DAC in the PM-7574 permits ratiometric operation of the ADC. Performance degradation may, however, occur as V_{REF} varies from $-10.000V$. This decrease in performance is due to comparator limitations including offset voltage, gain, and input noise.

The PM-7574 uses the reference as a power supply for the comparator to increase speed and accuracy. Reference voltages of a magnitude less than $-9V$ must be avoided for accurate comparator operation. For best accuracy, the use of a $0.1\mu F$ bypass capacitor (Pin 2 to AGND) is recommended.

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

ประวัติผู้เขียน

ชื่อผู้เขียน	นายจิระศักดิ์ ชาญวุฒิธรรม
วันเดือนปีเกิด	วันที่ 28 พฤศจิกายน พ.ศ.2504
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษาระดับปริญญาตรี	อุตสาหกรรมศาสตรบัณฑิต สาขาเทคโนโลยีไฟฟ้าอุตสาหกรรม
สถานที่สำเร็จการศึกษา	สถาบันเทคโนโลยีพระจอมเกล้า พระนครเหนือ
ปีที่สำเร็จการศึกษา	ปีการศึกษา 2529
ผลงานทางวิชาการที่ได้รับการตีพิมพ์	เรื่องไมโครคอนโทรลเลอร์ในระบบควบคุมแบบ แยกส่วนผ่านพอร์ตอนุกรมเพื่องานควบคุมใน โรงงานขนาดเล็ก
รางวัลหรือทุนที่เคยได้รับ	ทุนอุดหนุนการทำวิทยานิพนธ์ บัณฑิตวิทยาลัย
ประสบการณ์ทำงาน	วิศวกรฝ่ายบำรุงรักษาระบบไฟฟ้า การไฟฟ้าฝ่ายผลิตแห่งประเทศไทย
อาชีพปัจจุบัน	อาจารย์ระดับ 4 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า พระนครเหนือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้