สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การควบคุมกล้องระยะไกลผ่านเครือข่ายอินเตอร์เน็ต

Remote control via Internet for room monitoring camera

T 050324

โดย

นาย พิชิต ไวทย์เลิศสกุล

นาย สารัติ วัฒนปกรณ์

นาย สุทธิรักษ์ ชุ่มจิต

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....
เลขทะเบียน....50324....
วัน,เดือน,ปี 29 เม.ย. 2547

.b.....
.i.....

สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมกล้องระยะไกลผ่านเครือข่ายอินเตอร์เน็ต

Remote control via Internet for room monitoring camera

โดย

นาย พิชิต	ไวทย์เลิศสกุล	รหัสประจำตัว	42010230
นาย สารัติ	วัฒนปกรณ์	รหัสประจำตัว	42010377
นาย สุทธิรักษ์	ชุ่มจิต	รหัสประจำตัว	42010389

อาจารย์ที่ปรึกษา

รศ.ดร.มนัส สังวรศิลป์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงาน ปีการศึกษา 2545

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมกล้องระยะไกลผ่านเครือข่ายอินเตอร์เน็ต

ผู้จัดทำ

1. นาย พิชิต	ไวทย์เลิศสกุล	รหัสประจำตัว	42010230
2. นาย สารัติ	วัฒนปกรณ์	รหัสประจำตัว	42010377
3. นาย สุทธิรักษ์	ชุ่มจิต	รหัสประจำตัว	42010389

.....อาจารย์ที่ปรึกษา

(รศ. ดร. มนัส สังวรศิลป์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงงานเรื่อง (ภาษาไทย) การควบคุมกล้องระยะไกลผ่านเครือข่ายอินเตอร์เน็ต

(ภาษาอังกฤษ) Remote control via Internet for room monitoring camera

จัดทำโดย

1. นาย พิชิต	ไวทย์เลิศสกุล	รหัสประจำตัว	42010230
2. นาย สารัติ	วัฒนปกรณ์	รหัสประจำตัว	42010377
3. นาย สุทธิรักษ์	ชุ่มจิต	รหัสประจำตัว	42010389

โครงงานได้รับการตรวจสอบแล้ว พร้อมจะทำการสอบได้

.....

(รศ.ดร. มนัส สังวรศิลป์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมกล้องระยะไกลผ่านเครือข่ายอินเตอร์เน็ต

นาย พิชิต ไวทย์เลิศสกุล
นาย สารัติ วัฒนปกรณ์
นาย สุทธิรักษ์ ชุ่มจิต
รศ.ดร.มนัส สังวรศิลป์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2545

บทคัดย่อ

ปัจจุบันระบบอินเตอร์เน็ตได้ถูกใช้อย่างแพร่หลายทั่วโลก จึงทำให้สามารถเข้าถึงข้อมูลต่างๆ ที่อยู่ในเครื่องคอมพิวเตอร์ที่อยู่ห่างไกลได้ ดังนั้นจึงมีแนวคิดในการส่งข้อมูลภาพจากบริเวณที่ต้องการสนใจ ซึ่งอาจอยู่ในสภาพที่มนุษย์ไม่สามารถทำงานตลอดเวลาได้ เช่น ห้องสูญญากาศ หรือบริเวณที่มีสารพิษ ไปยังเครื่องคอมพิวเตอร์ที่อยู่ในสถานที่ที่เหมาะสมกับการทำงาน เพื่อสังเกตการณ์ ในบริเวณนั้นๆ ได้ตลอดเวลา โดยไม่เกิดอันตรายแก่ผู้สังเกตการณ์ และยังสามารถสังเกตการณ์ได้จากทุกที่ที่เครือข่ายอินเตอร์เน็ตเข้าถึง ทำให้ไม่ต้องเสียเวลาในการเดินทาง และประหยัดค่าใช้จ่าย

ในวิทยานิพนธ์ฉบับนี้จะได้นำเสนอถึงแนวทางการออกแบบอุปกรณ์ ซึ่งทำหน้าที่บันทึกภาพ และส่งข้อมูลไปยังเครื่องคอมพิวเตอร์ ประกอบไปด้วย CCD-Sensor ซึ่งเป็นอุปกรณ์สำหรับรับภาพ สเตปเปอร์มอเตอร์ ควบคุมทิศทางของอุปกรณ์รับภาพ การเขียนโปรแกรมสำหรับไมโครคอนโทรลเลอร์เพื่อประมวลผลคำสั่ง และควบคุมอุปกรณ์ต่าง ๆ และการเชื่อมต่อระหว่างอุปกรณ์กับเครื่องคอมพิวเตอร์ และการเขียนโปรแกรม Visual Basic สำหรับติดต่อกับอุปกรณ์ และระบบอินเตอร์เน็ตเพื่อรอรับการควบคุมจากระยะไกล ซึ่งทำงานเป็นโฮสต์ และโปรแกรมปลายทางสำหรับติดต่อเพื่อควบคุมจากระยะไกล

สำหรับวิทยานิพนธ์ฉบับนี้จะทำการศึกษา ออกแบบ และสร้างอุปกรณ์ ดังกล่าว โดยเริ่มจากการศึกษาการใช้งานไมโครคอนโทรลเลอร์ เพื่อติดต่ออุปกรณ์ต่าง ๆ และการศึกษาการประยุกต์ใช้งานเครือข่ายอินเตอร์เน็ต ในการควบคุมจากระยะไกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Remote control via Internet for room monitoring camera

Mr. Pichit Wailertsakul
Mr. Sarat Wattanapakorn
Mr. Suthiruk Chumjit
Advisor Mr. Manus Sunkvornsin
year 2002

Abstract

Nowadays, internet system becomes a part of the world. It makes people to get any data from other computers which are distant from them. Consequently, there are many applications that use this system to make living better. One of these applications is sending any desired pictures from a particular place that people cannot live in or work all the time, for example, the vacuum space or any dangerous places, to a computer where it can be operated by human. People can see what happens to that place and it has no danger for working. Moreover, not only can this computer be shown those pictures, but other computers which are connected to an internet system can do it as well some expenses for this matter will also be reduced.

This thesis shows the method how to design all devices which are about recording and transfering data to a computer. They consist of CCD-Sensor -- a device for receive a picture, stepping motor -- a device for controlling a picture-receiver, microcontroller program for compiler, the way to connect this device with a computer and other programs which are related to this thesis.

In this thesis which is separated in three main parts, studying, designing and making this device, initially has a study of using microcontroller for connecting other devices and a study of the application of using internet system for controlling any distant areas.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
สารบัญ	II
สารบัญรูป	III
สารบัญตาราง	IV
บทที่ 1 บทนำ	1
บทที่ 2 พื้นฐานไมโครคอนโทรลเลอร์	2
2.1 สถาปัตยกรรม RISC	2
2.3 ทำไม RISC จึงทำงานเร็วขึ้น	3
2.4 คุณสมบัติเทคนิคของ PIC16F877 ที่สำคัญ	4
2.5 การใช้ภาษาเบสิคใน PIC16F877	5
บทที่ 3 กล้องเกมบอย (Game Boy Camera) และ การติดต่อกับไมโครคอนโทรลเลอร์ PIC16F877	7
3.1 กล้องเกมบอย(Game Boy Camera)	7
3.2 M64282FP CMOS Image Sensor	8
3.3 การควบคุมการทำงานของ M64282FP CMOS Image Sensor	10
3.4 หน้าที่ของรีจิสเตอร์ต่างภายใน M64282FP	10
3.5 การควบคุมการทำงานของกล้องเกมบอยโดยไมโครคอนโทรลเลอร์ PIC16F877	14
บทที่ 4 การออกแบบส่วนควบคุมกล้องโดยสเต็ปเปอร์มอเตอร์(Stepper Motor) และดีซีมอเตอร์(DC Motor)	19
4.1 การกระตุ้นและควบคุมการหมุนของสเต็ปเปอร์มอเตอร์	19
4.2 การขับสเต็ปเปอร์มอเตอร์ของ PIC16F877	21
4.3 การควบคุมการทำงานสเต็ปเปอร์มอเตอร์โดยใช้ไอซี	22
4.4 การควบคุมทิศทางการหมุนของดีซีมอเตอร์	25
4.5 การควบคุมทิศทางของดีซีมอเตอร์โดยใช้ไอซี H-Bridge Switching	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

4.6 การควบคุมการทำงาน 26

บทที่ 5 พื้นฐานการส่งข้อมูลผ่านพอร์ตยูเอสบี (USB) 28

5.1 หลักการพื้นฐานของยูเอสบี 28

5.2 ยูเอสบีคอนเนคเตอร์ 28

5.3 การเลือกใช้ความเร็ว 29

5.4 การรับส่งข้อมูลของอุปกรณ์ยูเอสบี 30

5.5 เอนพอยท์ (Endpoint) 31

5.6 ไปป์ (Pipe) 31

5.7 ยูเอสบี เดสคริปเตอร์ (USB Descriptors) 31

5.8 Setup Packet 32

บทที่ 6 การสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์ผ่านพอร์ตยูเอสบี 37

6.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C 37

6.2 คุณสมบัติโดยทั่วไปของบัส I^2C 38

6.3 หลักการของบัส I^2C 38

6.4 การเขียนโปรแกรมติดต่อบัส I^2C 40

6.5 การสื่อสารข้อมูลกับไอซี PDIDSBD11 บนบัส I^2C 40

6.6 การเขียนโปรแกรมติดต่ออุปกรณ์ยูเอสบี 42

6.7 การเรียกใช้ยูเอสบีไอโอคอมด้วยวิชวลเบสิค 42

6.8 ฟังก์ชั่นของยูเอสบีไอโอ 43

บทที่ 7 เครือข่ายอินเตอร์เน็ต 50

7.1 สถาปัตยกรรม อินเตอร์เน็ต (Internet Architectures) 50

7.2 OSI โมเดล 51

7.3 รูปแบบมาตรฐานโพรโตคอลของอินเตอร์เน็ต (Internet Protocol Standards) 53

7.4 Internet IP 53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

7.4.1 Address Structure 53

7.4.2 รูปแบบของข้อมูล (Datagram) 54

7.5 การแบ่งส่วนของข้อมูลและการประกอบขึ้นใหม่ 56
(Fragmentation and Reassemble)

7.6 การเลือกเส้นทาง (Routing) 56

บทที่ 8 หลักการเขียนโปรแกรม Winsock ที่ใช้กับ TCP/IP

8.1 Process Layer 58

8.2 Host-to-Host Layer 60

8.3 โพรโตคอล TCP 61

8.4 โพรโตคอล UDP 62

8.5 Internetwork Layer 63

8.6 โพรโตคอล IP 63

8.7 โพรโตคอล ICMP 63

8.8 โพรโตคอล ARP 64

8.9 Network Interface Layer 64

8.10 กลไกของโพรโตคอล IP 66

8.11 การเขียนโปรแกรมส่งข้อมูลผ่าน TCP/IP ด้วย Visual Basic 68

บทที่ 9 ผลการทดลอง 71

บทที่ 10 บทสรุป 87

หนังสืออ้างอิง

กิตติกรรมประกาศ

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

		หน้า
รูปที่ 2.1	แสดงสถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC16F877	5
รูปที่ 3.1	แสดงกล้องเกมบอย (Game Boy Camera)	7
รูปที่ 3.2	แสดงไอซี M64282FP CMOS Image Sensor ภายในกล้องเกมบอย	7
รูปที่ 3.3	แสดงบล็อคไดอะแกรมของ M64282FP CMOS Image Sensor	8
รูปที่ 3.4	แสดงขาต่างๆของกล้องเกมบอย	9
รูปที่ 3.5	แสดงโฟล์วชาร์ตการทำงานของ M64282FP	10
รูปที่ 3.6	แสดงบล็อกไดอะแกรมการใช้ PIC16F877 ควบคุมการทำงานกล้องเกมบอย	14
รูปที่ 3.7	แสดงไทม์มิ่งไดอะแกรมการส่งสัญญาณรีเซต	15
รูปที่ 3.8	แสดงไทม์มิ่งไดอะแกรมส่งสัญญาณข้อมูลอินพุทไปยังรีจิสเตอร์แอดเดรสต่างๆ	15
รูปที่ 3.9	แสดงแสดงไทม์มิ่งไดอะแกรมแสดงการส่งสัญญาณภาพเอาท์พุท ขณะเริ่มต้นส่งสัญญาณภาพ	15
รูปที่ 3.10	แสดงแสดงไทม์มิ่งไดอะแกรมแสดงการส่งสัญญาณภาพเอาท์พุท ขณะสิ้นสุดการส่งสัญญาณภาพ	16
รูปที่ 3.11	แสดงสัญญาณการเขียนข้อมูลลงในหน่วยความจำ	16
รูปที่ 3.12	แสดงสัญญาณการอ่านข้อมูลจากหน่วยความจำ	16
รูปที่ 3.13	แสดงการอ่านข้อมูลจากไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอล	17
รูปที่ 3.14	แสดงการต่อวงจรเพื่อรับภาพจากกล้องเกมบอย	18
รูปที่ 4.1	แสดงไอซี UCN5804	22
รูปที่ 4.2	แสดงไทม์มิ่งไดอะแกรมการทำงานของ UCN 5804	22
รูปที่ 4.3	แสดงการต่อ UCN5804 กับ PIC16F877	23
รูปที่ 4.4	แสดงโฟลว์ชาร์ตการเขียนโปรแกรมสำหรับ PIC16F877 ติดต่อกับ ucn5804	24
รูปที่ 4.5	แสดงหลักการทำงานของ H-Bridge Switching	25
รูปที่ 4.6	แสดงการต่อวงจร TP7279P	26
รูปที่ 4.7	แสดงโฟลว์ชาร์ตการเขียนโปรแกรมสำหรับ PIC16F877 ติดต่อกับ TA7279P	27
รูปที่ 5.1	Type A USB Connector	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 5.2 Type B USB Connector 28

รูปที่ 5.3 Full Speed Device 29

รูปที่ 5.4 Low Speed Device 29

รูปที่ 5.5 ไดอะแกรมการรับส่งข้อมูลของอุปกรณ์ยูเอสบี 30

รูปที่ 5.6 Device Descriptor 32

รูปที่ 5.7 Configure Descriptor 32

รูปที่ 6.1 แสดงบล็อกไดอะแกรมการต่อ PDIUSB11 กับไมโครคอนโทรลเลอร์ 37

รูปที่ 6.2 ไดอะแกรมเวลาแสดงสถานะต่างๆในบัส I^2C 39

รูปที่ 6.3 ตัวอย่างข้อมูลที่สื่อสารบนบัส I^2C 41

รูปที่ 6.4 แสดงการนำไอซี PDIDSBD11 ไปใช้ในวงจร 41

รูปที่ 6.5 การเรียกใช้ ยูเอสบีไอโอ 42

รูปที่ 7.1 แสดงสถาปัตยกรรมของ Internet 50

a) Single LAN and WAN

b) Interconnected LAN/WAN

รูปที่ 7.2 แสดงการแบ่งเครือข่ายออกเป็น OSI Model 51

รูปที่ 7.3 โครงสร้าง Address ที่ใช้ใน Class ต่างๆ ของเครือข่าย โดยมีทั้งหมด 32 bit 54

รูปที่ 7.4 Internet Datagram Format and Contents 55

รูปที่ 8.1 แสดง TCP/IP Stack เปรียบเทียบกับมาตรฐาน OSI 59

รูปที่ 8.2 เมื่อแอปพลิเคชั่น หรือ โพรเซส สื่อสารกัน ผ่านจุดเชื่อมต่อ หรือ Port 60

รูปที่ 8.3 การใช้งาน โพรโตคอล TCP และ UDP ของ Process หลัก ๆ 61

รูปที่ 8.4 โพรโตคอล TCP และ UDP 64

รูปที่ 8.5 โครงสร้างของโพรโตคอล TCP/IP ในแต่ละชั้นหรือ Layer 65

รูปที่ 8.6 แสดง IP Header 67

รูปที่ 8.7 การเรียกใช้งาน Winsock Control 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 3.1 แสดงรายละเอียดการทำงานของแต่ละขา	9
ตารางที่ 3.2 แสดงแอดเดรสต่างๆใน M64282FP	10
ตารางที่ 4.1 แสดงลำดับการทำงานของขดลวดในแต่ละเฟสของมอเตอร์ เมื่อได้รับการกระตุ้นแบบฟูลสเต็ปหนึ่งเฟส	19
ตารางที่ 4.2 แสดงลำดับการทำงานของขดลวดในแต่ละเฟสของมอเตอร์ เมื่อได้รับการกระตุ้นแบบฟูลสเต็ปสองเฟส	20
ตารางที่ 4.3 แสดงลำดับการทำงานของขดลวดในแต่ละเฟสของมอเตอร์ เมื่อได้รับการกระตุ้นแบบฟูลสเต็ปสองเฟส	21
ตารางที่ 4.4 แสดงการกำหนดรูปแบบการกระตุ้นสเต็ปเปอร์มอเตอร์	22
ตารางที่ 4.5 แสดงฟังก์ชันการทำงานของ TA7279P	26
ตารางที่ 5.1 Setup Packet	33
ตารางที่ 5.2 Standard Device Request	34
ตารางที่ 5.3 Standard Endpoint Request	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในทุกวันนี้ระบบอินเตอร์เน็ต ได้เข้ามามีบทบาทในชีวิตประจำวันเป็นอย่างมาก โดยเฉพาะอย่างยิ่งการติดต่อสื่อสาร การแลกเปลี่ยนข้อมูล การค้นหาข้อมูลต่าง ๆ ซึ่งระบบอินเตอร์เน็ตทำให้ระยะทางไม่เป็นข้อจำกัดในการสื่อสาร แลกเปลี่ยนข้อมูล

จากข้อดีดังกล่าวจึงมีแนวคิดในการส่งข้อมูลภาพจากบริเวณที่ต้องการสนใจ ซึ่งอาจอยู่ในสภาพที่มนุษย์ไม่สามารถทำงานตลอดเวลาได้ เช่น ห้องสุญญากาศ หรือบริเวณที่มีสารพิษ ไปยังเครื่องคอมพิวเตอร์ที่อยู่ในสถานที่ที่เหมาะสมกับการทำงาน เพื่อสังเกตการณ์ ในบริเวณนั้นๆ ได้ตลอดเวลา โดยไม่เกิดอันตรายแก่ผู้สังเกตการณ์ และยังสามารถสังเกตการณ์ได้จากทุกที่ที่เครือข่ายอินเตอร์เน็ตเข้าถึง ทำให้ไม่ต้องเสียเวลาในการเดินทาง และประหยัดค่าใช้จ่าย

โครงงานนี้ได้นำแนวคิดจากแนวคิดข้างต้นมาศึกษา ออกแบบ และทดลองใช้งาน เพื่อเป็นต้นแบบในการพัฒนาต่อไป

วิทยานิพนธ์ฉบับนี้จะกล่าวถึง

- คุณสมบัติที่สำคัญของไมโครคอนโทรลเลอร์ PIC 16F877
- การควบคุมและรับข้อมูลภาพจาก CCD
- การทำงานของสเต็ปเปอร์มอเตอร์ และการใช้งาน
- ทฤษฎีการเชื่อมต่อผ่านพอร์ต USB และการเขียนโปรแกรมเพื่อรับส่งข้อมูล
- ทฤษฎีพื้นฐานของระบบอินเตอร์เน็ต และ การเขียนโปรแกรมด้วย Visual Basic เพื่อรับส่งข้อมูลผ่านระบบอินเตอร์เน็ต
- การใช้งานโปรแกรมควบโดยตรง และโปรแกรมควบคุมผ่านระบบอินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
พื้นฐานไมโครคอนโทรลเลอร์

สำหรับไมโครคอนโทรลเลอร์ที่ใช้จะเป็น ไมโครคอนโทรลเลอร์ในตระกูล PIC (Peripheral Interface Controller) ของบริษัท ไมโครชิป เทคโนโลยี (MicroChip Technology) ไมโครคอนโทรลเลอร์ในตระกูล PIC มีด้วยกันหลายเบอร์แต่ละเบอร์มีขีดความสามารถแตกต่างกันไป สำหรับโครงงานนี้เบอร์ที่มีความเหมาะสมที่สุดก็คือ PIC16F877 ที่ความถี่ 20 MHz เนื่องจากในวงจรดิจิไทซ์สัญญาณภาพ ต้องใช้อุปกรณ์ที่มีความเร็วสูง

ไมโครคอนโทรลเลอร์ PIC16F877 เป็นไมโครคอนโทรลเลอร์ที่จัดอยู่ในไมโครโพรเซสเซอร์ประเภท RISC หรือ Reduced Instruction Set Computer กล่าวคือเป็นไมโครคอนโทรลเลอร์ที่มีคำสั่งน้อยมากคือ 35 คำสั่งพื้นฐานเท่านั้นและทุกคำสั่งจะทำงานเสร็จใน 1 ไซเกิลทั้งยังทำงานในลักษณะไปป์ไลน์ (Pipe line) เหมือนกับไมโครโพรเซสเซอร์สมัยใหม่ ความเร็วในการทำงานจึงสูงมากเมื่อเทียบกับไมโครคอนโทรเลอร์อื่นที่มีสัญญาณนาฬิกาเท่านั้น

2.1 สถาปัตยกรรม RISC

แต่เดิมแนวความคิดที่จะทำให้คอมพิวเตอร์ทำงานได้เร็วขึ้น จะใช้วิธีการเพิ่มขีดความสามารถของคำสั่ง คำสั่งหนึ่งคำสั่งจึงทำงานเพิ่มขึ้นและซับซ้อนขึ้น เช่นคำสั่งค้นหาข้อมูลในสตริงเป็นต้น ด้วยวิธีนี้ทำให้สถาปัตยกรรมเฉพาะของซีพียูต้องสนับสนุนคำสั่งใหม่ๆเพิ่มขึ้น ประจวบกับการทำงานของแต่ละคำสั่งใช้จำนวนไซเกิลไม่เท่ากันบางคำสั่งทำงานเสร็จภายในไซเกิลเดียวบางคำสั่งใช้หลายไซเกิลความคิดนี้จึงกลายมาเป็นคอมพิวเตอร์ในกลุ่ม CISC หรือ Complex Instruction Set Computer และความคิดนี้ได้พัฒนาต่อเนื่องมาเป็นลำดับจนถึงปัจจุบัน อีกแนวความคิดหนึ่งได้เริ่มมาตั้งแต่ปี พ.ศ.2518 โดยกลุ่มนักวิจัยแห่งมหาวิทยาลัยแคลิฟอร์เนีย ได้พัฒนาซีพียูที่ชื่อว่า RISC แนวความคิดคือ ให้ซีพียูทำงานที่ไซเกิลแน่นอน โดยลดจำนวนคำสั่งลงให้เหลือคำสั่งพื้นฐานมากที่สุด แล้วใช้หลักการทำงานแบบไปป์ไลน์ (pipeline) คือนำคำสั่งมาเรียงกันแบบขนานเหลื่อมกัน หรือเข้าทำงานแต่ละสถานีตามลำดับเรียงกันไป ทุกสถานีงานจะมีงานทำตลอดเวลาการลดจำนวนคำสั่งลงนี้เอง จึงกลายเป็นซีพียูประเภท RISC หรือ Reduce Instruction Set คอมพิวเตอร์แบบ RISC จึงทำงานได้เร็วและเป็นกลไกที่สามารถเพิ่มขีดความสามารถโดยรวมได้ หลายบริษัทจึงหันมาพัฒนาคอมพิวเตอร์แบบ RISC จนในที่สุดมีซับซีพียูที่กำหนดขึ้นมาและบริษัทซันไมโครซิสเต็ม นำมาใช้เป็นซีพียูหลักของตน คือ SPARC โดย SPARC เป็นซีพียูที่มีการกำหนดโครงสร้างสถาปัตยกรรมและคำสั่งไว้ชัดเจน จึงมีบริษัทนำไปพัฒนาโครงสร้างชิป จนชิป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้นำไปดัดแปลง เผยแพร่ หรือนำไปฝากขายต่อ

ไมโครโปรเซสเซอร์ SPARC ได้รับการพัฒนาขีดความสามารถ และการประยุกต์เข้ากับเครื่องเวิร์คสเตชันและเซริฟเวอร์อย่างกว้างขวาง

2.2 ทำไม RISC จึงทำงานเร็วขึ้น

การทำงานมีลักษณะตามปรัชญาของ RISC ดังนี้ ทำงานไซเกิลเดียวต่อ 1 คำสั่ง ความเป็นจริงแล้วการทำงานหนึ่งคำสั่งต้องใช้เวลามากกว่า 1 ไซเกิล แต่ที่ทำได้เพราะคำสั่งเหล่านี้มีลักษณะเป็นไปป์ไลน์ และขนานกันจึงทำให้เวลาโดยรวมเฉลี่ยออกมาแล้วได้คำสั่งละไซเกิลเดียว ออกแบบโหลดและสตอร์ เพื่อลดการทำงานที่ล่าช้าคำสั่งเหล่านี้จะต้องแน่นอนและทำงานได้ในลักษณะขนานตามขบวนการที่กำหนด ใช้การควบคุมด้วยฮาร์ดแวร์ โดยอาศัยวงจรอิเล็กทรอนิกส์ที่ทำให้การทำงานเป็นไปในลักษณะที่ทำคำสั่งได้ในไซเกิลเดียว โดยไม่ต้องใช้ไมโครโค้ด มีจำนวนคำสั่งน้อย และอ้างแอดเดรสจำกัด เพื่อให้โครงสร้างตายตัวในเรื่องไซเกิล การทำงานจำต้องลดคำสั่งลงให้เหลือเท่าที่จำเป็นเพื่อให้ฮาร์ดแวร์ควบคุมฟอร์แมตคำสั่งได้ง่าย เพื่อให้การแปลความหมายสามารถทำโดยฮาร์ดแวร์ได้ จึงต้องใช้เวลาคอมไฟล์ภาษาระดับสูงมาก เพื่อเปลี่ยนแปลงภาษาระดับสูงมาเป็นระดับต่ำ ต้องใช้คอมไฟล์เลอร์ที่แปลงมาเป็นรหัสภาษาเครื่องได้โดยคำสั่งห่างไกลจากความหมายของภาษาระดับสูงมาก จึงต้องเสียเวลาในการคอมไฟล์นานกว่า

การที่ RISC ทำงานได้เร็วกว่าเพราะการวางโครงสร้างการทำงานได้โดยกำหนดส่วนของการทำงานไว้ชัดเจนและเป็นแบบขนานเช่น MIPS โมเดล R2000 ได้วางคำสั่งที่มีการทำงาน 5 ขั้นตอน ดังนั้นจึงวางโครงสร้างการทำงานแบบขนานไว้ 5 ระดับ โดยแบ่งแยกส่วนของการทำงานแต่ละคำสั่งเป็น

IF การแฟตช์คำสั่ง (Instruction Fetch) ค่าจังหวะที่แฟตช์คำสั่งจากหน่วยความจำมาตีความหมาย

RD อ่านโอเปอแรนด์จากรีจิสเตอร์ในซีพียู

ALU การคำนวณในหน่วย ALU ตามรหัสของคำสั่งนั้น ๆ

MEM การติดต่อกับหน่วยความจำ

WB การเขียนข้อมูลผลลัพธ์ลงในรีจิสเตอร์

การใช้วิธีการของคำสั่งในการทำงานแบบขนานนี้ เป็นผลทำให้สามารถทำงานได้เสร็จใน 1 ไซเกิล ทั้งๆที่แท้จริงแล้วต้องใช้ถึง 5 ไซเกิล แต่ทุกขณะจะมีส่วนของซีพียูทำงานขนานกันทั้ง 5 ส่วนนี้อยู่ ขณะที่แฟตช์คำสั่งหนึ่งอยู่ภายในก็มีการกระทำ RD, ALU, MEM, WB ของแต่ละคำสั่งที่ผ่านมา ดังนั้นการแฟตช์จึงเกิดได้ทุกไซเกิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

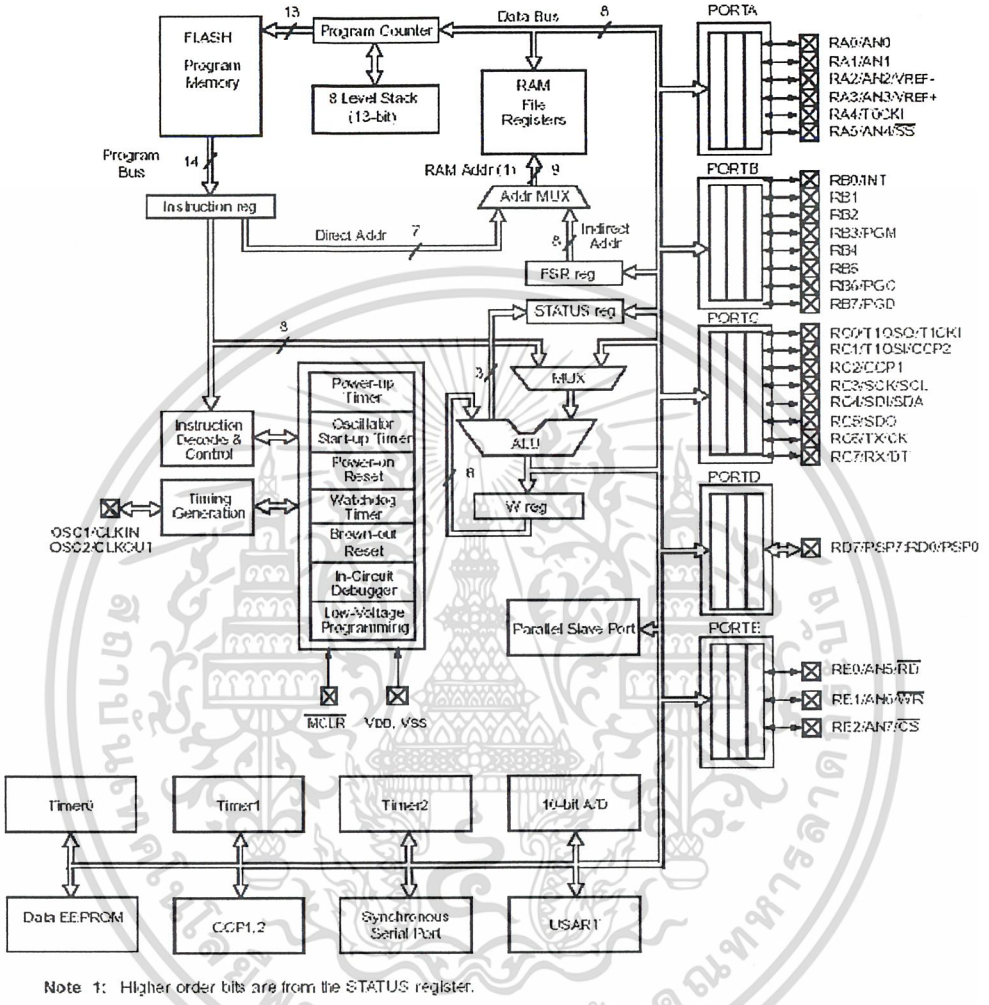
กลไกการทำงานในลักษณะนี้ใช้กับซีพียู RISC ทุกตัว ตั้งแต่จุดเริ่มต้นของการพัฒนาจนถึงปัจจุบัน โครงสร้างของคำสั่งจึงอยู่ที่ ALU ว่าจะรับข้อมูลได้กว้างเพียงไร เช่น ถ้ารับข้อมูลได้ 64 บิต และให้ ALU เป็นหน่วยคำนวณทางคณิตศาสตร์แบบตัวเลขโฟลตดิ้งพอยต์ ก็ยังทำงานได้เร็วยิ่งขึ้น เช่น ซีพียู 1860 ของอินเทลก็ใช้หลักการของ RISC เช่นกัน

2.3 คุณสมบัติเทคนิคของ PIC16F877 ที่สำคัญ

- หน่วยประมวลผลกลางเป็นแบบ RISC
- คำสั่งเพียง 35 คำสั่ง
- ทุกคำสั่งใช้เวลาประมวลผลเพียง 1 ไซเกิลของสัญญาณนาฬิกา ยกเว้นคำสั่งการกระโดดจะใช้เวลา 2 ไซเกิลของสัญญาณนาฬิกา
- ความเร็วในการประมวลผล : DC – 20MHz Clock input
  DC – 200ns instruction cycle
- แหล่งกำเนิดอินเตอร์รัปต์ (มากกว่า 14 แหล่ง)
- โหมดการอ้างอิงแอดเดรส 3 โหมดคือ แบบโดยตรง (Direct) ทางอ้อม (Indirect) แบบสัมพัทธ์ (Relation)
- เพาเวอร์ออนรีเซตในตัว (POR)
- เพาเวอร์อัพไทม์เมอร์ (PWRT) และ ออสซิลเลอร์เตอร์สตาร์ตอัพไทม์เมอร์ (OST)
- วอตช์ดอกไทม์เมอร์
- โหมดประหยัดพลังงาน (Sleep mode)
- สามารถเลือกใช้ออสซิลเลเตอร์ทั้งภายในและภายนอก
- เทคโนโลยี CMOS FLASH/EEPROM ใช้พลังงานต่ำ ความเร็วสูง
- สามารถโปรแกรมแบบอนุกรมโดยใช้งานเพียง 2 ขา
- ย่านไฟเลี้ยง 2.0 - 5.5 โวลท์
- กระแสซิงค์และซอร์สสูงประมาณ 25mA
- มีขนาดหน่วยความจำโปรแกรมซึ่งเป็นแบบแฟลช มีขนาด 8K × 14 เวิร์ด หน่วยความจำอีอีพรอมภายในขนาด 256 × 8 ไบต์ และหน่วยความจำแรม 368 × 8 ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874	4K	192 Bytes	128 Bytes
PIC16F877	8K	368 Bytes	256 Bytes



Note 1: Higher order bits are from the STATUS register.

รูปที่ 2.1 แสดงสถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC16F877

2.4 การใช้ภาษาเบสิคใน PIC16F877

โดยปกติในการใช้ไมโครคอนโทรลเลอร์ทั่วๆไป จะใช้ภาษาแอสเซมบลี้ในการเขียนโปรแกรม แต่สำหรับในโครงงานนี้จะใช้ภาษาเบสิคในการเขียนโปรแกรม

ข้อดีของภาษาเบสิค

1. ใช้งานง่าย เนื่องจากมีคำสั่งสำเร็จรูปอยู่แล้ว เช่น คำสั่งติดต่อกับพอร์ตอนุกรมจะเป็น SERIN,SEROUT เป็นต้น
2. โค้ดสั้น เข้าใจง่าย สะดวกในการตรวจสอบหาจุดผิด
3. สามารถแทรกคำสั่งที่เป็นภาษาแอสเซมบลี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสียของภาษาเบสิค

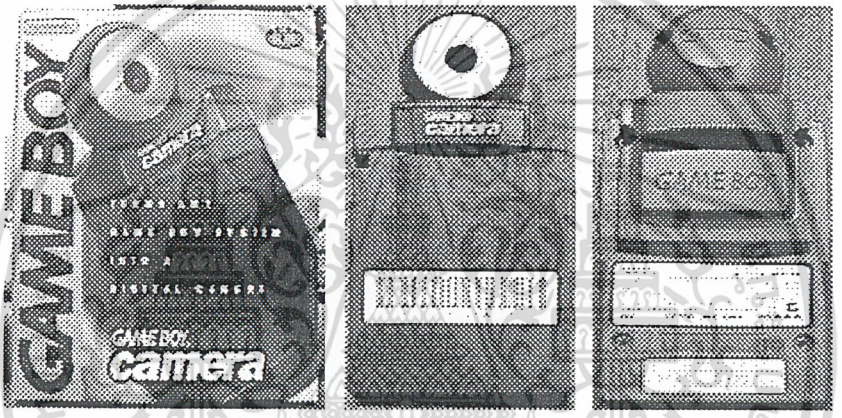
1. ไม่สามารถคำนวณเวลาที่ใช้ในการทำงานของแต่ละคำสั่งในภาษาเบสิคได้
2. ใช้หน่วยความจำโปรแกรมมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

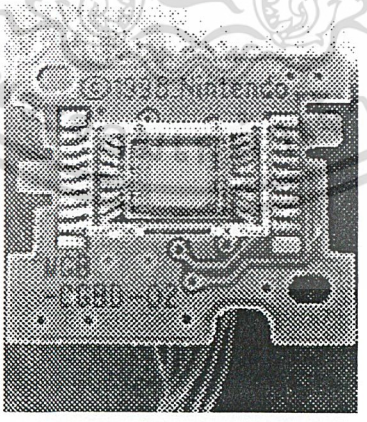
บทที่ 3
กล้องเกมบอย(Game Boy Camera)และการติดต่อกับไมโครคอนโทรลเลอร์ PIC16F877

3.1 กล้องเกมบอย(Game Boy Camera)

กล้องเกมบอยผลิตโดยบริษัทนินเทนโด(Nintendo) ซึ่งทำขึ้นเพื่อนำไปใช้งานร่วมกับเกมบอย(Game Boy) ทำให้เกมบอยนั้นสามารถถ่ายภาพได้ ภายในกล้องเกมบอยจะประกอบด้วยไอซี M64282FP CMOS Image Sensor ของบริษัทมิตซูบิชิ เป็นหัวใจสำคัญในการเก็บภาพ



รูปที่ 3.1 แสดงกล้องเกมบอย (Game Boy Camera)



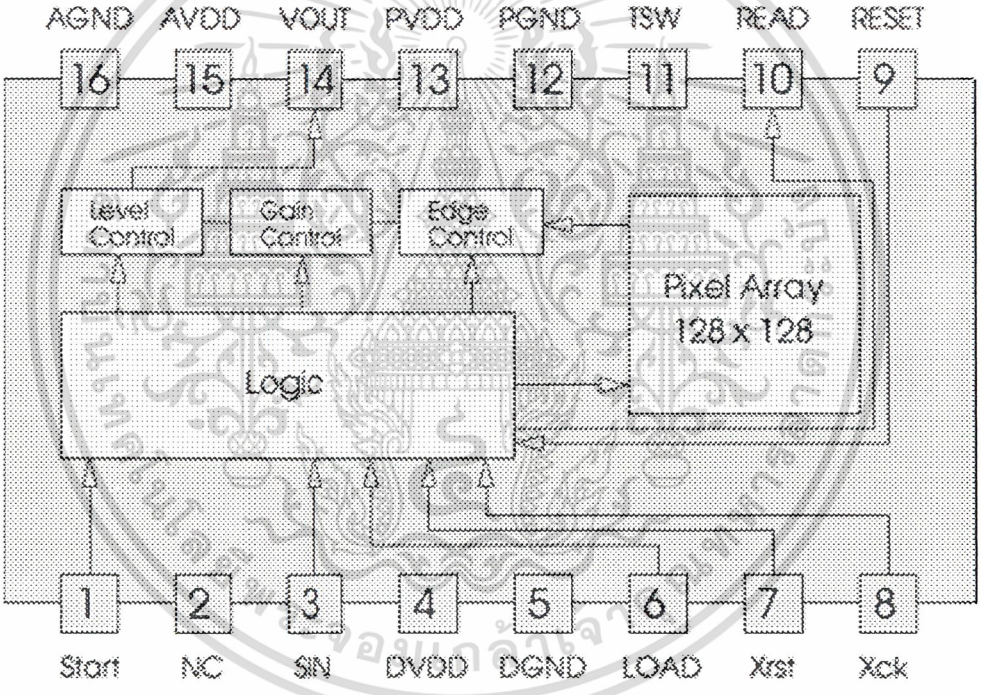
รูปที่ 3.2 แสดงไอซี M64282FP CMOS Image Sensor ภายในกล้องเกมบอย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 M64282FP CMOS Image Sensor

คุณสมบัติที่สำคัญ

- ภาพขนาด 128 X 128 พิกเซล
- ไฟเลี้ยง 5 โวลต์ กินกำลังงานต่ำ
- ประกอบด้วยฟังก์ชันการทำงาน
  - Positive/negative output image
  - Edge enhancement / extraction
  - Output level & gain tuning



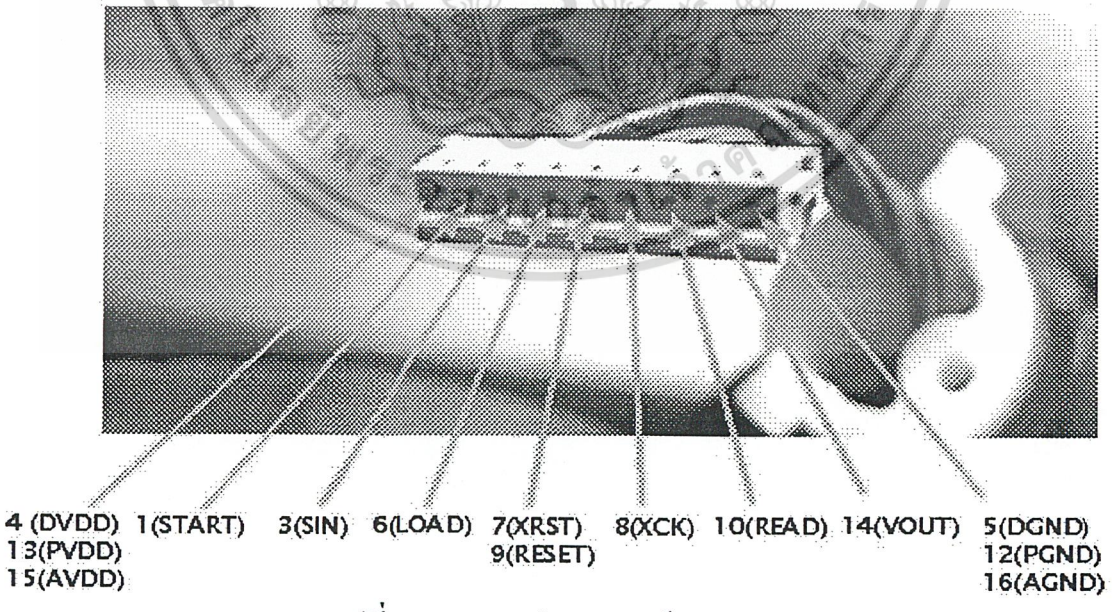
รูปที่ 3.3 แสดงบล็อคไดอะแกรมของ M64282FP CMOS Image Sensor

ขา	สัญลักษณ์	ฟังก์ชัน	รายละเอียด
1	START	Start Input	รับสัญญาณเริ่มต้นการส่งสัญญาณภาพ
2	NC		ไม่มีการเชื่อมต่อ
3	SIN	Data Input	รับสัญญาณข้อมูลอินพุท
4	DVDD	Digital Power Supply	แหล่งจ่ายไฟสำหรับสัญญาณดิจิตอล(5 V)
5	DGND	Digital Ground	กราวด์สำหรับสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6	LOAD	Data Set Input	อีนาเบิลสัญญาณข้อมูลอินพุท
7	Xrst	Data Set Input	รับรีเซตสัญญาณทั้งหมด
8	Xck	System Clock Input	รับสัญญาณ clock ที่ใช้ในวงจร
9	RESET	Memory Reset Input	รับสัญญาณรีเซตค่าในรีจิสเตอร์
10	READ	Read Image	ส่งสัญญาณแสดงการอ่านข้อมูลภาพได้
11	TSW	Reserved	ไม่มีการเชื่อมต่อ
12	AGND1	Analog GND	กราวด์สำหรับสัญญาณอนาลอก
13	AVDD1	Analog Power Supply	แหล่งจ่ายสำหรับสัญญาณอนาลอก(5 V)
14	Vout	Signal Output	ส่งสัญญาณภาพเอาท์พุท
15	AVDD2	Analog Power Supply	แหล่งจ่ายสำหรับสัญญาณอนาลอก(5 V)
16	AGND1	Analog GND	กราวด์สำหรับสัญญาณอนาลอก

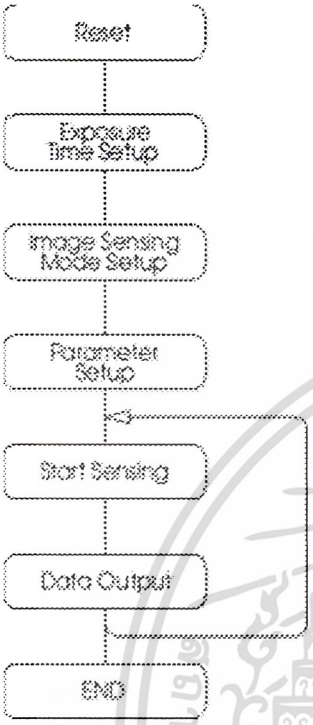
ตารางที่ 3.1 แสดงรายละเอียดการทำงานของแต่ละขา



รูปที่ 3.4 แสดงขาต่างๆของกล้องเกมบอย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การควบคุมการทำงานของ M64282FP CMOS Image Sensor



รูปที่ 3.5 แสดงโฟล์วชาร์ตการทำงาน ของ M64282FP

เริ่มจากทำการรีเซตวงจรและค่าต่างในรีจิสเตอร์โดยส่งสัญญาณ RESET และ Xrst ตามไทม์มิ่งไดอะแกรม

เซตค่า Exposure Time, โหมดการจับภาพ ทำได้ส่งข้อมูลอินพุท(SIN) รีจิสเตอร์ต่างๆ ข้อมูล SIN จะประกอบด้วยข้อมูลแอดเดรสรีจิสเตอร์ 3 บิต และข้อมูล 8 บิต ข้อมูลแต่ละชุดเมื่อส่งครบให้ส่งสัญญาณ LOAD ตามไทม์มิ่งไดอะแกรมด้านบน

ส่งสัญญาณ START เป็นการเริ่มต้นจับภาพ เมื่อสิ้นสุด Exposure Time แล้ว สัญญาณภาพส่งออกมาพร้อมสัญญาณ READ

ส่งสัญญาณ START หากต้องการเก็บภาพต่อไป เมื่อสิ้นสุดภาพแรกแล้ว

Reg. No.	Address	7	6	5	4	3	2	1	0
1	001	N	VH1	VH0	G4	G3	G2	G1	G0
2	010	C17	C16	C15	C14	C13	C12	C11	C10
3	011	C07	C06	C05	C04	C03	C02	C01	C00
4	100	P7	P6	P5	P4	P3	P2	P1	P0
5	101	M7	M6	M5	M4	M3	M2	M1	M0
6	110	X7	X6	X5	X4	X3	X2	X1	X0
7	111	E3	E2	E1	E0	I	V2	V1	V0
0	000	Z1	Z0	O5	O4	O3	O2	O1	O0

ตารางที่ 3.2 แสดงแอดเดรสต่างๆใน M64282FP

3.4 หน้าที่ของรีจิสเตอร์ต่างภายใน M64282FP

1) รีจิสเตอร์ “N” (1 บิต)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าทำการเซตรีจิสเตอร์ “N” จะทำให้รีจิสเตอร์ “P” และ “M” ทำหน้าที่ในโหมด vertical edge extraction/enhancement

2) รีจิสเตอร์ “VH” (2 บิต)

ทำหน้าที่ในโหมด Edge Extraction (V,H,2-D) โดย

VH1	VH0	การทำงาน
0	0	No edge operation
0	1	Horizontal edge mode
1	0	Vertical edge mode
1	1	2-D edge mode

3) รีจิสเตอร์ “E” (4 บิต)

สำหรับเซตค่า Edge Enhancement Ratio

E2	E1	E0	Edge Enhancement Ratio
0	0	0	50%
0	0	1	75%
0	1	0	100%
0	1	1	125%
1	0	0	200%
1	0	1	300%
1	1	0	400%
1	1	1	500%

4) รีจิสเตอร์ “Z” (2 บิต)

เป็นการปรับแต่งค่าศูนย์ของสัญญาณภาพเอาท์พุท

Z1	Z0	Calibration
0	0	No calibration
1	0	Calibration for positive signal
0	1	Calibration for negative signal

5) รีจิสเตอร์ “I” (1 บิต)

ถ้ารีจิสเตอร์ “I” เป็น “1” สัญญาณภาพเอาท์พุทจะจะอยู่ในลักษณะของสัญญาณอินเวิร์ท

6) รีจิสเตอร์ “C0 & C1” (8 บิต x 2)

สำหรับเซตค่า Exposure Time โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ “C0” (8 บิต)

ค่าภายในรีจิสเตอร์	Exposure Time(msec)
00H	0
FFH	4.080

ช่วงละ	16 µsec
จำนวนช่วง	256

รีจิสเตอร์ “C1” (8 บิต)

ค่าภายในรีจิสเตอร์	Exposure Time(msec)
00H	0
FFH	1044.5

ช่วงละ	4.096 msec
จำนวนช่วง	256

7) รีจิสเตอร์ “O” (6 บิต)

สำหรับปรับแต่งค่า ออฟเซต (offset) ของสัญญาณภาพเอาท์พุท

ถ้า O5 = “1”

ค่าภายในรีจิสเตอร์	แรงดันออฟเซต(โวลท์)
20H	0
2FH	1

ช่วงละ	32 mV
จำนวนช่วง	32

ถ้า O5 = “0”

ค่าภายในรีจิสเตอร์	แรงดันออฟเซต(โวลท์)
00H	0
1FH	-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วงละ	-32 mV
จำนวนช่วง	32

8) รีจิสเตอร์ “V” (3 บิต)

สำหรับเซตค่าแรงอ้างอิง(Vref)ของสัญญาณภาพเอาท์พุท

V2	V1	V0	แรงดันอ้างอิง
0	0	0	0.0
0	0	1	0.5
0	1	0	1.0
0	1	1	1.5
1	0	0	2.0
1	0	1	2.5
1	1	0	3.0
1	1	1	3.5

9) รีจิสเตอร์ “G” (3 บิต)

เซตค่าอัตราขยาย(gain)ของสัญญาณเอาท์พุท

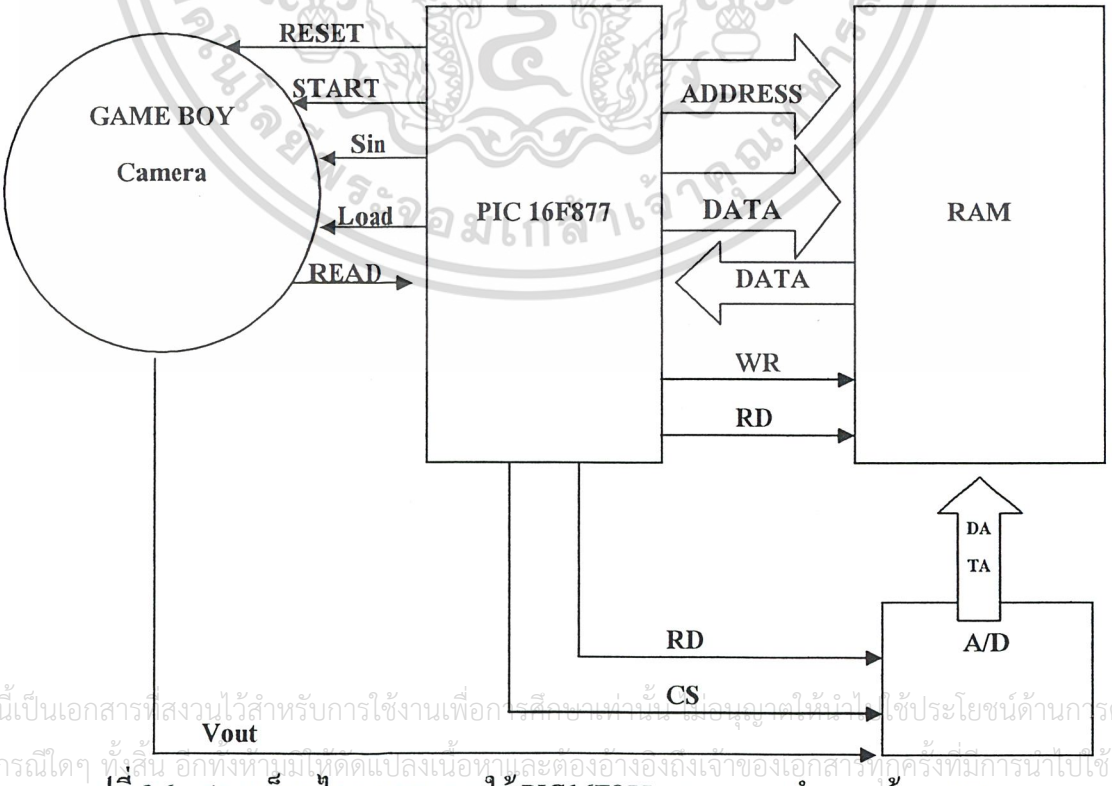
G3	G2	G1	G0	อัตราขยาย(dB)	
				G4	
				0	1
0	0	0	0	14.0	20.0
0	0	0	1	15.5	21.5
0	0	1	0	17.0	23.0
0	0	1	1	18.5	24.5
0	1	0	0	20.0	26.0
0	1	0	1	21.5	27.5
0	1	1	0	23.0	29.0
0	1	1	1	24.5	30.5
1	0	0	0	26.0	32.0
1	0	0	1	29.0	35.0
1	0	1	0	32.0	38.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1	0	1	1	35.0	41.0
1	1	0	0	38.0	44.0
1	1	0	1	41.0	47.0
1	1	1	0	45.5	51.5
1	1	1	1	51.5	57.5

3.5 การควบคุมการทำงานของกล้องเกมบอยโดยไมโครคอนโทรลเลอร์ PIC16F877

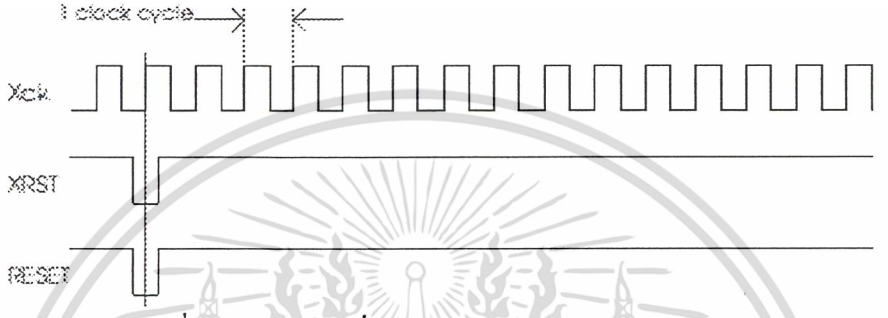
เนื่องจากสัญญาณเอาท์พุทของกล้องเกมบอยมีลักษณะเป็นสัญญาณอนาลอก โดยสัญญาณภาพแต่ละจุดขึ้นอยู่กับสัญญาณ XCK ดังนั้น สัญญาณเอาท์พุทจะช้าหรือเร็วนั้นขึ้นอยู่กับสัญญาณ XCK โดยจะเรียงลำดับออกมาตั้งแต่จุดแรกถึงจุดสุดท้าย ทั้งหมด 128X128 จุด ต้องทำการแปลงสัญญาณภาพแต่ละจุดเป็นสัญญาณภาพที่เป็นสัญญาณดิจิตอลโดยไอซีแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลขนาดจุดละ 8 บิต ข้อมูลภาพดิจิตอลจะถูกเก็บในหน่วยความจำ ซึ่งเราเลือกใช้ RAM ขนาด 128 X 128 = 16384 ไบต์ สัญญาณควบคุมกล้องเกมบอย,สัญญาณควบคุมหน่วยความจำและไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอลสามารถสร้างจากไมโครคอนโทรลเลอร์ได้ทั้งหมด จะเห็นได้จากบล็อกไดอะแกรม



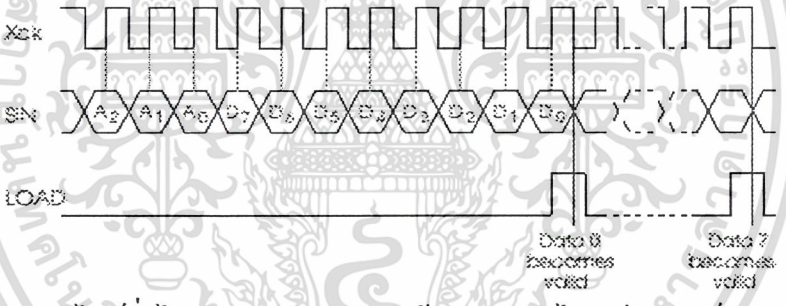
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6 แสดงบล็อกไดอะแกรมการใช้ PIC16F877 ควบคุมการทำงานกล้องเกมบอย

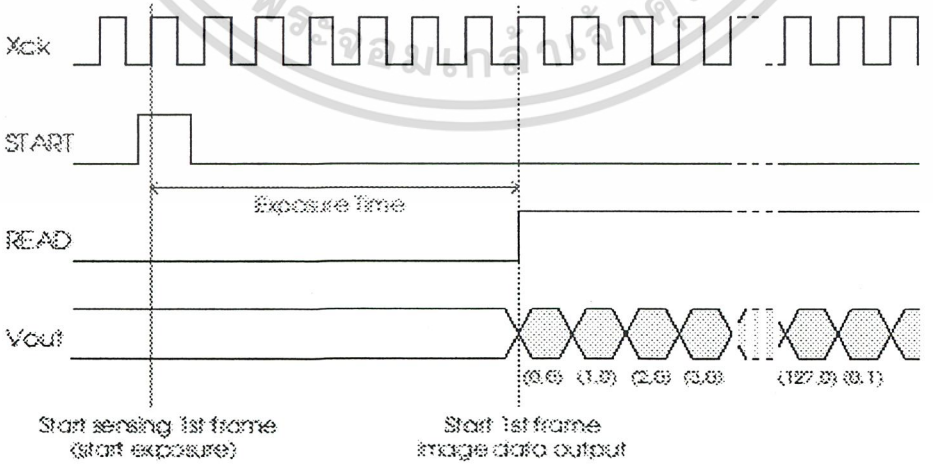
ขั้นตอนการทำงานเริ่มจากไมโครคอนโทรลเลอร์สร้างสัญญาณ RESET ไปยังกล้องเกมบอย จากนั้นทำการส่งข้อมูลโหมดการเก็บภาพไปยังรีจิสเตอร์ภายในกล้องเกมบอยโดยส่งสัญญาณ SIN และ LOAD ส่งสัญญาณ START แล้วรอจนกว่าจะสิ้นสุดค่า Exposure time โดยเช็คจากสัญญาณ READ ทำการแปลงข้อมูลเป็นดิจิตอลโดยการส่งสัญญาณ RD และ CS ไปยัง A/D และส่งค่าแอดเดรส, สัญญาณ WR ไปยังหน่วยความจำ ทำเช่นเดิมจนครบข้อมูลภาพทุกจุด



รูปที่ 3.7 แสดงไทม์มิ่งไดอะแกรมการส่งสัญญาณรีเซต

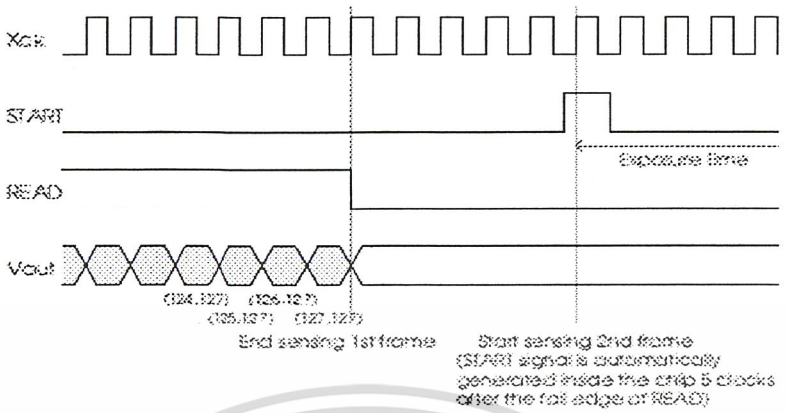


รูปที่ 3.8 แสดงไทม์มิ่งไดอะแกรมส่งสัญญาณข้อมูลอินพุทไปยังรีจิสเตอร์แอดเดรสต่างๆ

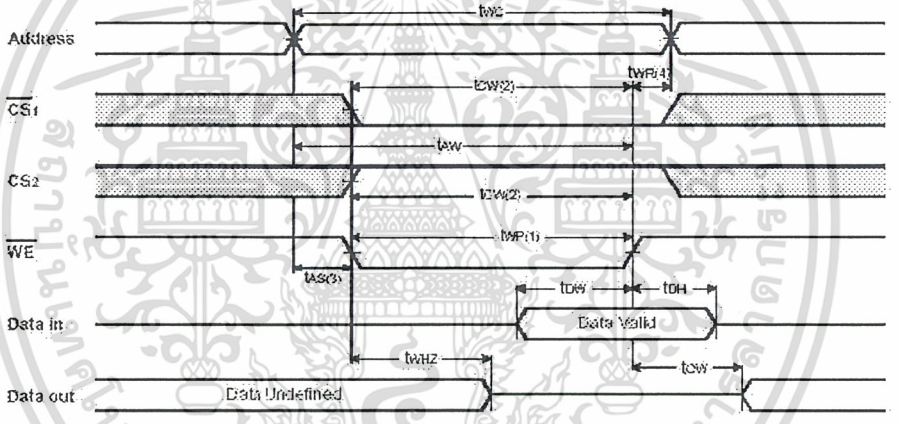


รูปที่ 3.9 แสดงแสดงไทม์มิ่งไดอะแกรมแสดงการส่งสัญญาณภาพเอาท์พุท ขณะเริ่มต้นส่งสัญญาณภาพ

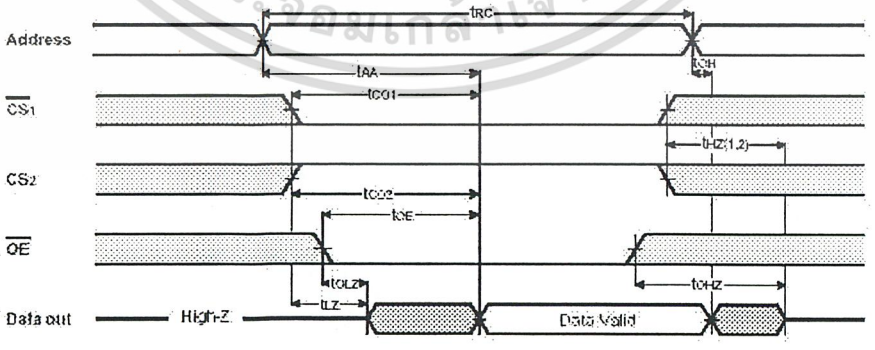
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 แสดงแสดงไทม์มิ่งไดอะแกรมแสดงการส่งสัญญาณภาพเอาท์พุท ขณะสิ้นสุดการส่งสัญญาณภาพ



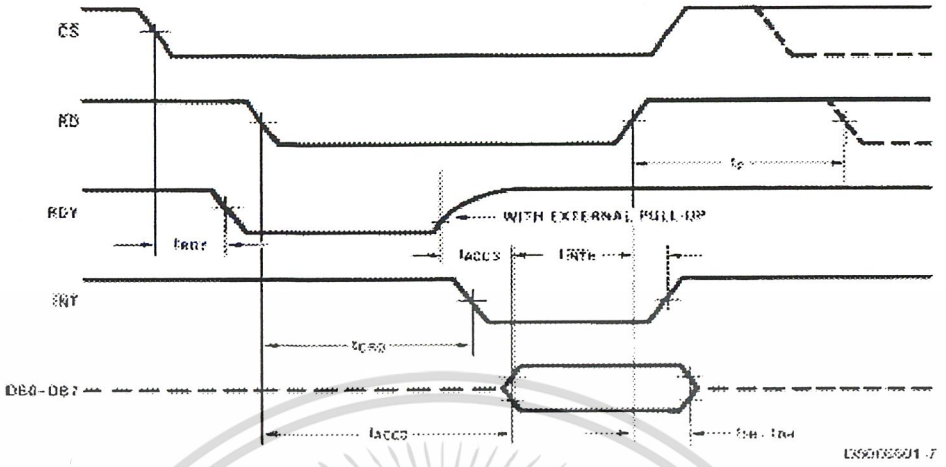
รูปที่ 3.11 แสดงสัญญาณการเขียนข้อมูลลงในหน่วยความจำ



รูปที่ 3.12 แสดงสัญญาณการอ่านข้อมูลจากหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

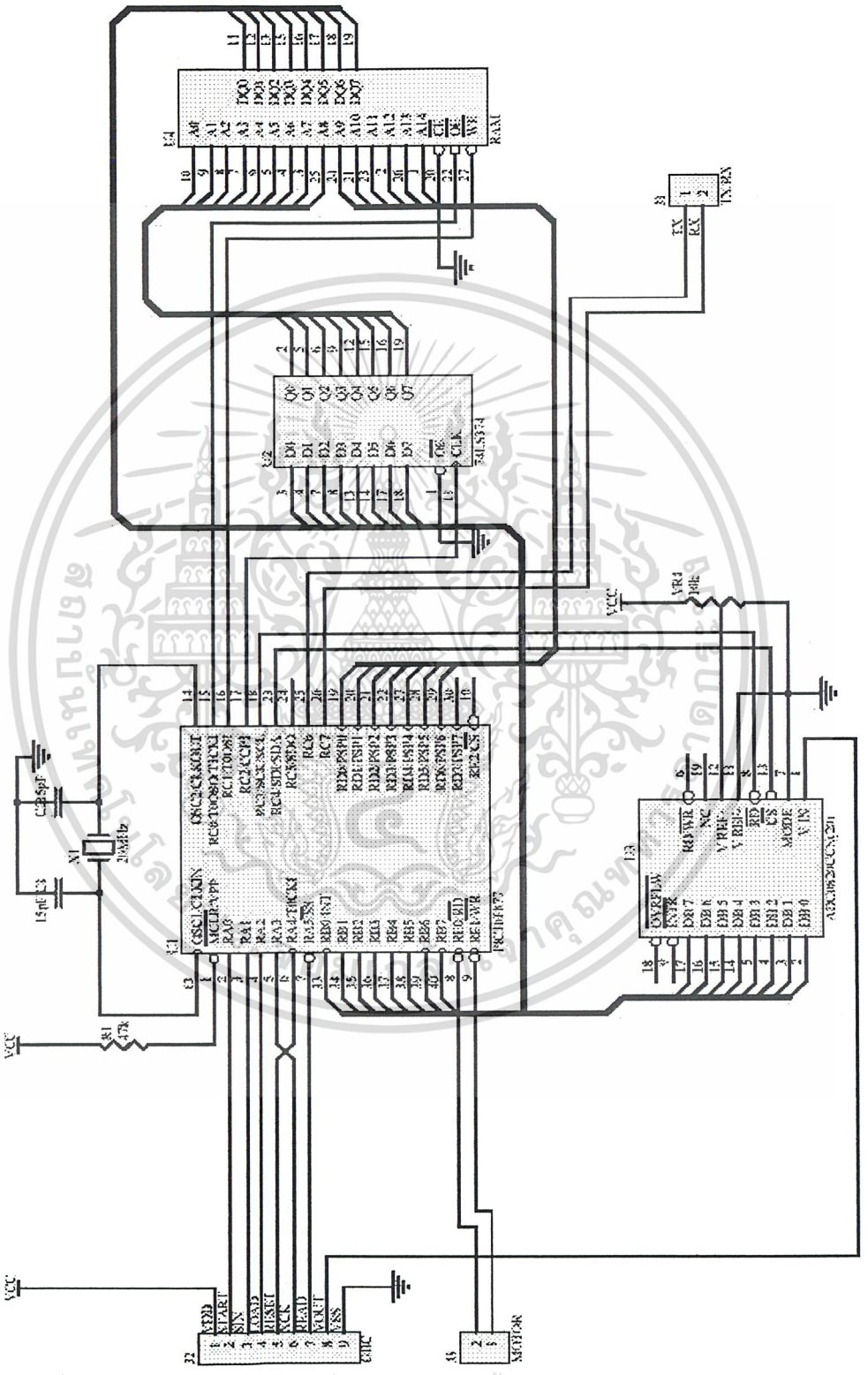
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 3.13 แสดงการอ่านข้อมูลจากไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

50324



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.14 แสดงการต่อวงจรเพื่อรับภาพจากกล้องเกมบอย

บทที่ 4
การออกแบบส่วนควบคุมกล้องโดยสเต็ปเปอร์มอเตอร์(Stepper Motor) และดีซีมอเตอร์(DC Motor)

สำหรับส่วนควบคุมการหมุนของกล้องในโครงงานนี้จะใช้ไมโครคอนโทรลเลอร์ควบคุมการหมุนของมอเตอร์ 2 ตัว ในแนวแกน x-y โดยจะใช้สเต็ปเปอร์มอเตอร์หมุนกล้องในแนวแกน x หมุนได้ 360 องศา เนื่องจากเป็นมอเตอร์ที่อยู่ด้านล่างจะต้องใช้แรงบิดมาก ส่วนในแกน y จะใช้ดีซีมอเตอร์เนื่องจากมีน้ำหนักเบาและไม่ต้องใช้แรงบิดมาก

4.1 การกระตุ้นและควบคุมการหมุนของสเต็ปเปอร์มอเตอร์

การกระตุ้นและควบคุมการหมุนของมอเตอร์ให้เคลื่อนที่ไปแต่ละสเต็ปทำได้โดยจ่ายกำลังไฟฟ้าไปยังขดลวดแต่ละขดบนสเตเตอร์ ซึ่งต้องป้อนเป็นแบบซีเควนเชียลในรูปแบบที่ถูกต้องด้วย สามารถแบ่งได้ 3 รูปแบบคือ แบบฟูลสเต็ป 1 เฟส (full step 1 phase), แบบฟูลสเต็ป 2 เฟส และแบบฮาล์ฟสเต็ป (half step)

แบบฟูลสเต็ป 1 เฟส เป็นการกระตุ้นที่มีรูปแบบง่ายที่สุด โดยการทำการกระตุ้นขดลวดทีละขดในเวลาหนึ่งไล่เรียงถัดกันไป เช่น เริ่มต้นที่ขดที่ 1, 2, 3, 4 แล้ววนกลับมาที่ขดที่ 1 วนไปเรื่อยๆ หรือเริ่มที่ขดที่ 1 แล้วย้อนไปยังขดที่ 4, 3, 2 แล้วกลับมายังขดที่ 1 อีกครั้ง ซึ่งทำให้ทิศทางของการหมุนสวนกัน ในการกระตุ้นรูปแบบนี้จึงมีขดลวดเพียงขดเดียวในเวลาหนึ่งที่ถูกกระตุ้นเท่านั้น วงจรกระตุ้นแบบฟูลสเต็ป 1 เฟส จึงมีราคาถูกและง่าย ขั้นตอนการทำงานต่างๆแสดงดังตารางที่ 4.1

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	-	ทำงาน	-	-
3	-	-	ทำงาน	-
4	-	-	-	ทำงาน

ตารางที่ 4.1 แสดงลำดับการทำงานของขดลวดในแต่ละเฟสของมอเตอร์เมื่อได้รับการกระตุ้นแบบฟูลสเต็ปหนึ่งเฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบฟูลสเต็ป 2 เฟสเป็นการกระตุ้นซึ่งคล้ายกับแบบฟูลสเต็ป 1 เฟส แต่การกระตุ้นแบบนี้จะทำการกระตุ้นโดยจ่ายกำลังไฟฟ้าไปที่ขดลวด 2 ขดที่อยู่ใกล้กันในเวลาเดียวกันและเรียงถัดกันไปเช่นเดียวกับแบบฟลูสเต็ป 1 เฟส ดังตัวอย่าง ขดลวดชุดแรกที่ถูกกระตุ้นจะเป็นขดที่ 1 และ 2ตามด้วยการกระตุ้นขดที่ 2 และ 3 ต่อไปที่ขดที่ 3 และ 4 ถัดไปเป็นขดที่ 4 และ 1 แล้วกลับมาที่ขดที่1 และ 2 วนไปตามลำดับเช่นนี้ หรือเริ่มที่ขด 1และ4 ตามด้วยขดที่4 และ 3 ถัดไปเป็นขดที่ 3 และ 2 ต่อไปเป็นขดที่2 และ 1แล้ววนกลับมาที่ขดที่1 และ 4ทิศทางการหมุนจะสวนทางกัน การกระตุ้นสเต็ปเปอร์มอเตอร์แบบนี้สามารถเพิ่มแรงบิดได้มากกว่าแบบเวฟ โรเตอร์จะเคลื่อนที่ด้วยแรงดึงอย่างเต็มแรงจาก 2 ขดลวดที่ถูกกระตุ้นพร้อมกัน และต่อไปด้วยแรงดึงจากอีก 2 ขดลวดถัดไป สำหรับข้อเสียคือการกระตุ้นแบบนี้ต้องใช้กำลังไฟฟ้ามากขึ้น ขั้นตอนการทำงานต่างๆ แสดงดังตารางที่ 4.2

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	ทำงาน	-	-
2	-	ทำงาน	ทำงาน	-
3	-	-	ทำงาน	ทำงาน
4	ทำงาน	-	-	ทำงาน

ตารางที่ 4.2 แสดงลำดับการทำงานของขดลวดในแต่ละเฟสของมอเตอร์
เมื่อได้รับการกระตุ้นแบบฟูลสเต็ปสองเฟส

แบบฮาล์ฟสเต็ปเป็นรูปแบบที่ผสมผสานระหว่างการกระตุ้นแบบฟูลสเต็ป 1 และ 2 เฟส เพื่อเพิ่มจำนวนของสเต็ปต่อรอบอีกเท่าตัวหนึ่ง ในระบบนี้จะทำการกระตุ้นขดลวดกันไปเป็นลำดับดังนี้ เริ่มจากขดลวดที่ 1, 1 และ 2, 2, 2 และ 3, 3, 3 และ 4, 4, 4 และ 1 แล้ววนกลับมายังขดลวดที่ 1 แรงบิดที่ได้จากการกระตุ้นแบบนี้จะเพิ่มมากขึ้นอีก เพราะช่วงสเต็ปมีระยะสั้นลง แต่ละสเต็ปเกิดแรงดึงจากขดลวด 2 ขดที่ถูกกระตุ้นพร้อมกัน ความถูกต้องของตำแหน่งมีเพิ่มมากขึ้น แต่ต้องพึงระวังไว้อีกประการหนึ่งว่า เมื่อกระตุ้นให้ทำงานในรูปแบบนี้จะต้องทำการหมุนถึง 2 สเต็ป จึงจะได้เท่ากับระยะเท่ากับ 1 สเต็ปเต็มของการควบคุมใน 2 แบบแรก สำหรับแหล่งจ่ายกำลังไฟฟ้าต้องใช้ขนาดท่ากับแบบ 2 เฟสเป็นอย่างน้อย จึงจะเพียงพอ ขั้นตอนการทำงานต่างๆ แสดงดังตารางที่ 4.3

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	ทำงาน	ทำงาน	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3	-	ทำงาน	-	-
4	-	ทำงาน	ทำงาน	-
5	-	-	ทำงาน	-
6	-	-	ทำงาน	ทำงาน
7	-	-	-	ทำงาน
8	ทำงาน	-	-	ทำงาน

ตารางที่ 4.3 แสดงลำดับการทำงานของขดลวดในแต่ละเฟสของมอเตอร์

เมื่อได้รับการกระตุ้นแบบฟูลสเต็ปสองเฟส

4.2 การขับสเต็ปเปอร์มอเตอร์ของ PIC16F877

การใช้ PIC16F877 ต่ออุปกรณ์แม่เหล็กไฟฟ้าอย่างมอเตอร์โดยตรงไม่สามารถทำได้ เนื่องจากกระแสทางเอาต์พุตของ PIC16F877 จำกัดไว้ที่ 20mA เท่านั้น จึงต้องอาศัยอุปกรณ์ขับช่วย ได้แก่ ทรานซิสเตอร์กำลังปานกลางหรือกำลังสูง, มอสเฟตกำลังปานกลางหรือกำลังสูง, ไอซีเบอร์ ULN2003 หรือ ULN2803 หรือ ULN2804 เป็นต้น

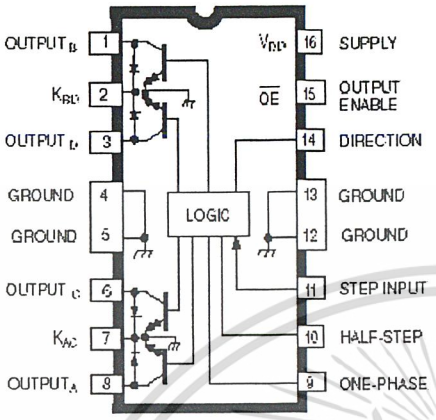
สำหรับสเต็ปเปอร์มอเตอร์เป็นมอเตอร์ที่มีลักษณะการทำงานแตกต่างจากมอเตอร์ทั่วไป เพราะจะต้องป้อนสัญญาณเป็นพัลส์ให้แก่ขดลวดของมอเตอร์เป็นจังหวะอย่างเหมาะสม และการหมุนของมอเตอร์ชนิดนี้จะหมุนเป็นจังหวะตามพัลส์ที่ป้อนเข้ามา ไม่หมุนต่อเนื่องเหมือนกับมอเตอร์ธรรมดา ทำให้ผู้ควบคุมสามารถเลือกตำแหน่งที่ต้องการให้มอเตอร์หยุดหมุนได้ จังหวะการหมุนของสเต็ปเปอร์มอเตอร์เรียกว่า สเต็ป(step) ความละเอียดของมอเตอร์กำหนดเป็นองศาที่หมุนไปในหนึ่งสเต็ป หากมอเตอร์มีจำนวนองศาต่อสเต็ปมาก หมายความว่ามอเตอร์ตัวนี้มีความละเอียดของการหมุนต่ำ ยกตัวอย่าง การหมุนครบ 1 รอบเท่ากับ 360 องศา หากมอเตอร์มี สเต็ปการหมุนเท่ากับ 7.5 องศาต่อสเต็ป มอเตอร์ตัวนี้มีความละเอียดของการหมุนเท่ากับ 48 ตำแหน่ง แต่ถ้าหากมีสเต็ปการหมุนเท่ากับ 1.8 องศาต่อสเต็ป ความละเอียดของการหมุนเท่ากับ 200 จะเห็นได้ว่ามอเตอร์ตัวหลังมีความละเอียดสูงกว่าตัวแรกมาก ทำให้นำมาใช้ในงานที่ต้องการกำหนดตำแหน่งได้ดีกว่า แม่นยำกว่า ผนวกเข้ากับวงจรขับแบบครึ่งสเต็ป ความละเอียดของการหมุนจะเพิ่มขึ้นอีก 2 เท่า ทำให้มีความละเอียดของการหมุนกลายเป็น 400 ตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

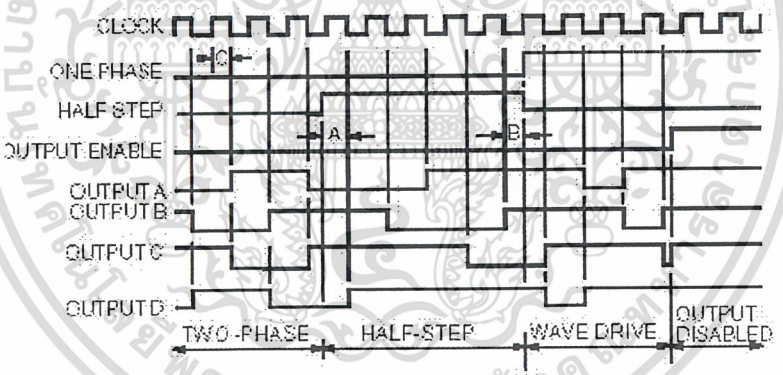
4.3 การควบคุมการทำงานสเต็ปเปอร์มอเตอร์โดยใช้ไอซี

ในโครงงานนี้เราเลือกใช้ไอซี ucn5804 BiMOS II Unipolar Stepper Motor translator/driver การควบใช้งานสามารถทำได้โดยจ่ายสัญญาณ clock เข้าไป

สามารถเลือกรูปแบบการกระตุ้นเฟสของมอเตอร์ และทิศทางได้ตามไดอะแกรมดังรูปที่ 4.1



รูปที่ 4.1 แสดงไอซี UCN5804

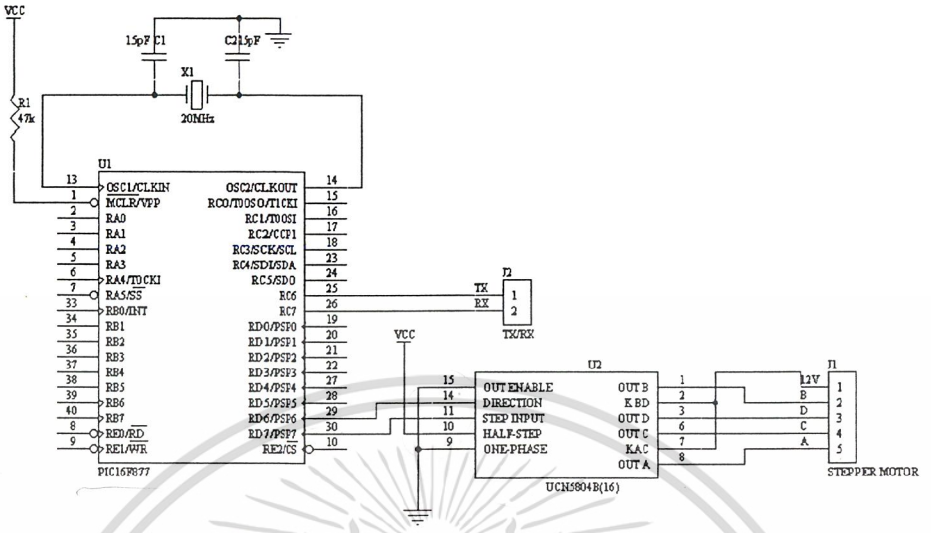


รูปที่ 4.2 แสดงไทม์มิ่งไดอะแกรมการทำงานของ UCN 5804

Drive Format	Pin 9	Pin 10
Two-Phase	L	L
One-Phase	H	L
Half-Step	L	H
Step-Inhibit	H	H

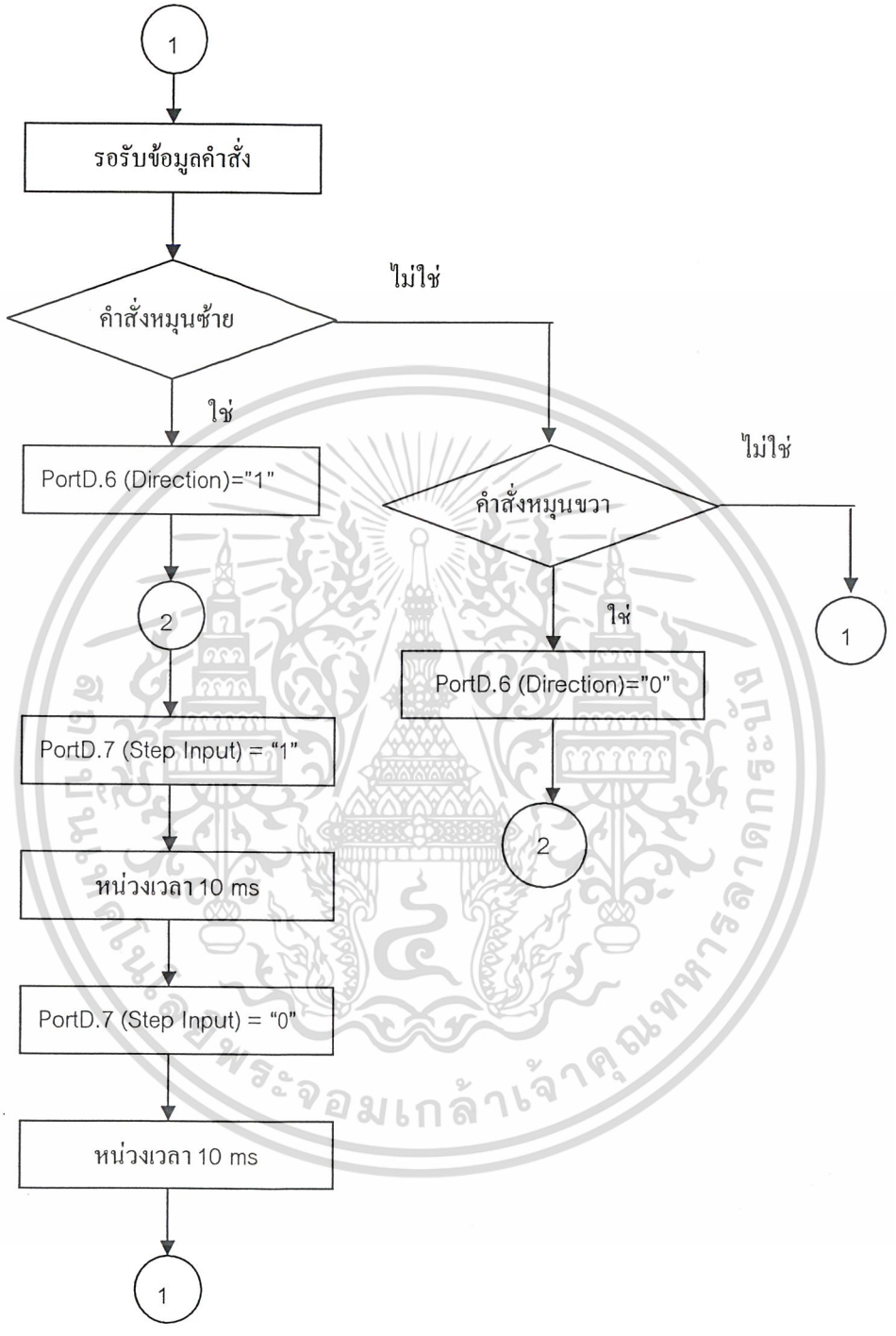
ตารางที่ 4.4 แสดงการกำหนดรูปแบบการกระตุ้นสเต็ปเปอร์มอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงการต่อ UCN5804 กับ PIC16F877

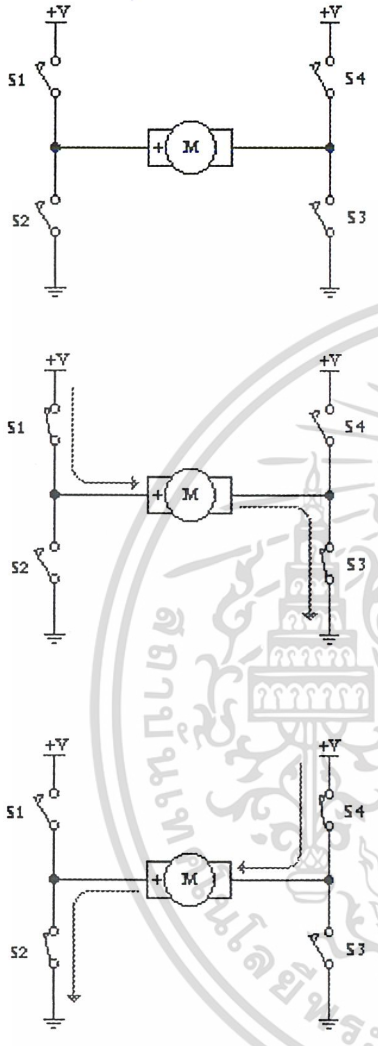
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงโฟลว์ชาร์ตการเขียนโปรแกรมสำหรับ PIC16F877 ติดต่อกับ ucn5804

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การควบคุมทิศทางการหมุนของดีซีมอเตอร์



ในการควบคุมการหมุนของดีซีมอเตอร์นั้น โดยปกติแล้วจะใช้หลักการของวงจร H-Bridge Switching หลักการของวงจรนั้น จะประกอบไปด้วย สวิตช์ 4 ตัว นั่นก็คือ S1, S2, S3 และ S4 นั่นเอง ซึ่งในรูปตัวอย่าง จะใช้ DC-Motor เป็น Load ของวงจรนั่นเอง

ในสภาวะเริ่มต้น สวิตช์ ทุกตัว Off อยู่ ก็จะไม่มีอะไรเกิดขึ้นทั้งสิ้น เพราะไม่มีกระแสไฟฟ้าไหลเข้าสู่มอเตอร์ (รูปบน)

และเมื่อเราทำการ On สวิตช์ S1 และ S3 พร้อมกัน (รูปกลาง) จะเป็นการเชื่อมวงจร ทำให้มีกระแสไฟฟ้า ไหลผ่านมอเตอร์ จากขั้วบวกของมอเตอร์ ไปยังขั้วลบของมอเตอร์ จึงทำให้มอเตอร์สามารหมุนได้ ในทิศทาง Forward (จะหมุนแบบตามเข็มนาฬิกา หรือทวนเข็มนาฬิกากานั้น ขึ้นอยู่กับลักษณะของการพันขดลวดภายในมอเตอร์)

และในทางกลับกัน ถ้าหากเราทำการ On สวิตช์ S2 และ S4 พร้อมกัน (รูปล่าง) ก็จะเป็นการเชื่อมวงจร และทำให้เกิดกระแสไฟฟ้า ไหลผ่านมอเตอร์ จากขั้วลบของมอเตอร์ ไปยังขั้วบวกของมอเตอร์ จึงทำให้มอเตอร์สามารถหมุนได้ และเป็นการหมุนในทิศทาง Reverse (กลับทิศทางกับกรณีแรก)

รูปที่ 4.5 แสดงหลักการทำงานของ H-Bridge Switching

4.5 การควบคุมทิศทางของดีซีมอเตอร์โดยใช้ไอซี H-Bridge Switching

ในโครงงานนี้เราเลือกใช้ไอซี Dual H-Bridge Switching เบอร์ TA7279P เป็น IC ที่ผลิตขึ้นโดยบริษัทโตชิบา ภายใน IC จะประกอบด้วยวงจร H-Bridge Switching จำนวน 2 ชุด(สามารถควบคุมมอเตอร์ได้ 2 ตัว) แต่ละชุดมีคุณสมบัติดังนี้

- รับแรงดันได้ตั้งแต่ 0 ~ 18 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เอาท์พุทจ่ายกระแสได้ถึง 1.0A (AVE) ,3.0A (Peak)
  นอกจากนั้นภายใน ยังมีฟังก์ชั่นหยุดการทำงานเมื่ออุณหภูมิสูงเกินไป และยังมีวงจรจำกัดกระแสอีกด้วย

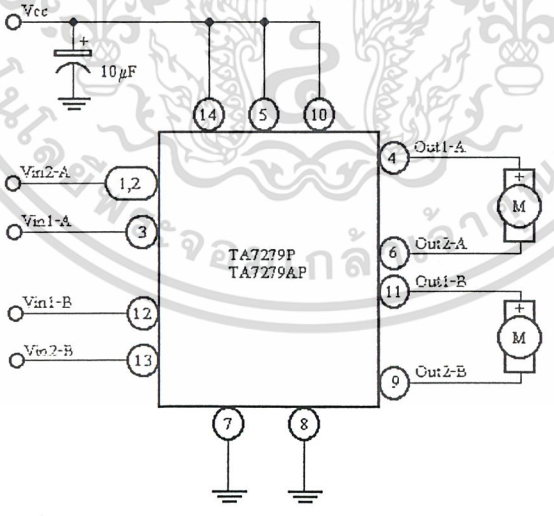
IN1	IN2	OUT1	OUT2	MODE
H	H	L	L	Brake
H	L	L	H	CW / CCW
L	H	H	L	CCW / CW
L	L	High impedance		STOP

ตารางที่ 4.5 แสดงฟังก์ชันการทำงานของ TA7279P

4.6 การควบคุมการทำงาน

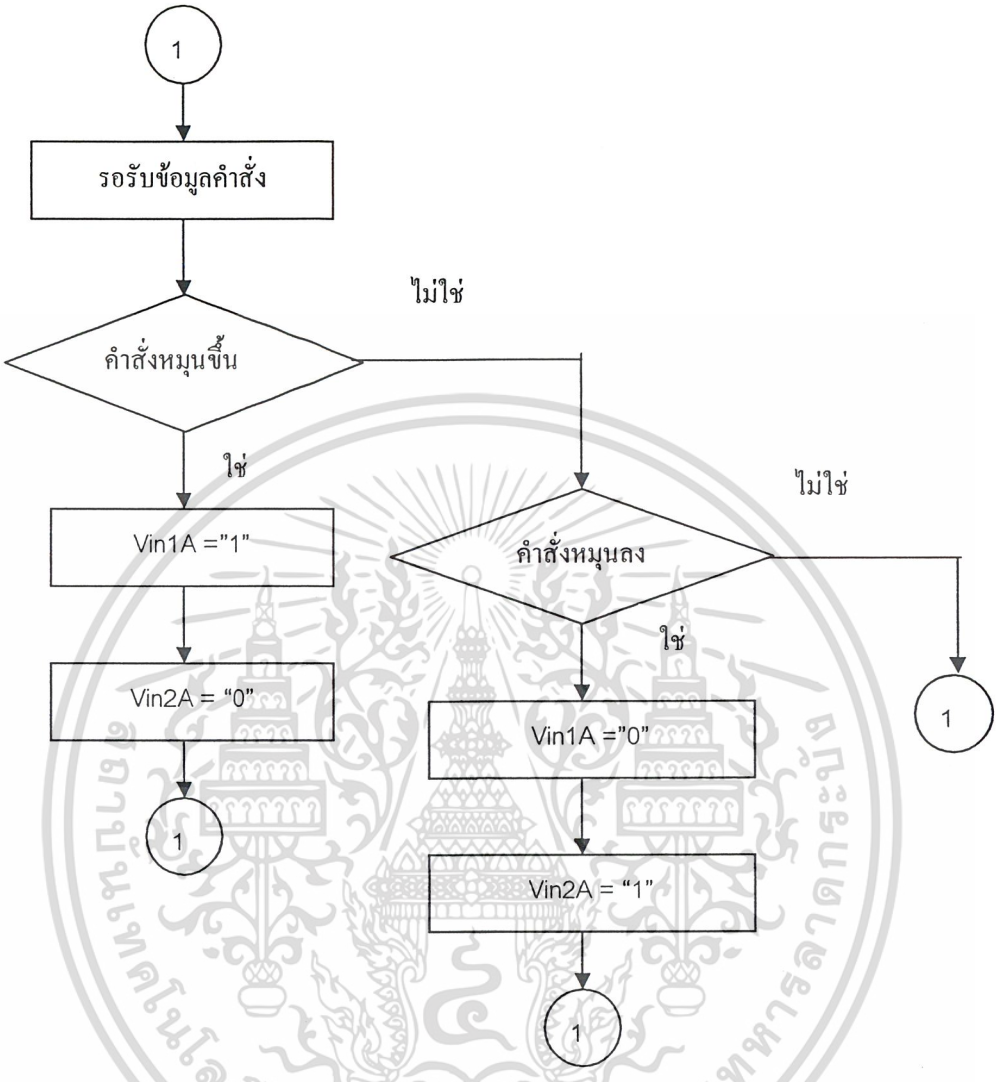
การควบคุมนั้นสามารถทำได้โดย การป้อนสัญญาณลอจิก "1" หรือสัญญาณลอจิก "0" เข้าที่ด้านอินพุต 1, 2 เท่านั้น วงจรก็สามารถทำงานได้ ส่วนมอเตอร์จะหมุน Forward หรือ Reward หรือหยุดหมุนขึ้นอยู่กับสัญญาณควบคุมที่ป้อนให้ ดังแสดงรายละเอียดดังตารางด้านบน

- Stop : จะทำงานเปรียบเสมือน การปลดมอเตอร์ออกจากแหล่งจ่าย
- Break : จะทำงานเปรียบเสมือน การล็อคมอเตอร์เอาไว้



รูปที่ 4.6 แสดงการต่อวงจร TP7279P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงโฟลว์ชาร์ตการเขียนโปรแกรมสำหรับ PIC16F877 ติดต่อกับ TA7279P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
พื้นฐานการส่งข้อมูลผ่านพอร์ตยูเอสบี (USB)

การสื่อสารข้อมูลผ่านพอร์ตยูเอสบี

พอร์ตยูเอสบี เป็นพอร์ตแบบใหม่ล่าสุด ที่ถูกออกแบบมาให้ใช้กับพีซีคอมพิวเตอร์ ให้สามารถรับส่งข้อมูลให้รวดเร็วยิ่งขึ้น สามารถต่ออุปกรณ์ได้มากถึง 127 ชิ้น เพราะมีแบนด์วิดธ์ในการรับส่งข้อมูลสูงกว่าพอร์ตแบบอื่นๆ พอร์ตแบบนี้ถูกออกแบบมาให้ใช้กับระบบ Plug and Play บนวินโดวส์ 98 ขึ้นไป

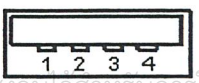
5.1 หลักการพื้นฐานของยูเอสบี

ในมาตรฐาน ยูเอสบี 1.1 นั้นจะรองรับความเร็วในการทำงานได้ 2 แบบ คือ ฟูลสปีด(Full speed) ซึ่งรับส่งข้อมูลที่ความเร็ว 12Mbits/s และ โลว์สปีด (Low speed) ซึ่งทำงานที่ความเร็ว 1.5Mbits/s โดยที่ความเร็ว 1.5Mbits/s จะมีการรบกวนทางไฟฟ้าน้อยที่สุด แต่ใจปัจจุบันนี้ได้มีมาตรฐาน ยูเอสบี 2.0 ซึ่งสามารถรับส่งข้อมูลได้ที่ความเร็วสูงถึง 480Mbits/s ซึ่งเร็วกว่ามาตรฐานการเชื่อมต่อแบบ ไฟร์ไวร์ ซีเรียล บัส ซึ่ง การทำงานของยูเอสบีจะถูกควบคุมด้วยโฮสต์ โดยสามารถมีได้ 1 โอสต์ ต่อ 1 บัส

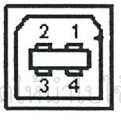
การทำงานแบบ Plug and Play ซึ่งสามารถเรียกใช้ Driver และยกเลิก Driver ได้ เมื่อผู้ใช้เชื่อมต่ออุปกรณ์ไปยังบัส โฮสจะตรวจพบการเชื่อมต่อและเรียกข้อมูลจากภายในอุปกรณ์เช่นรหัสผู้ผลิต และรหัสสินค้าเพื่อนำไปเรียก Driver ขึ้นมาใช้ โดยที่ผู้ใช้ไม่จำเป็นต้องจัดการกับ IRQ หรือ Port Address หรือเริ่มต้นคอมพิวเตอร์ใหม่อีกครั้ง และเมื่อใช้งานเสร็จก็สามารถถอดอุปกรณ์ออกได้ทันที เมื่อโฮสไม่เจออุปกรณ์นั้น โฮสจะยกเลิก Driver เอง

5.2 ยูเอสบีคอนเนคเตอร์

อุปกรณ์ทุกชนิดจะมีพอร์ตที่ใช้เชื่อมต่อกับโฮสต์ และโฮสต์ก็จะมีพอร์ตสำหรับเชื่อมต่อไปยังอุปกรณ์ โดยพอร์ตทั้งสองชนิดนี้ไม่สามารถใช้แทนกันได้ เพื่อป้องกันความผิดพลาดในการเชื่อมต่อ จึงมีคอนเนคเตอร์สองชนิดดังรูป



รูปที่ 5.1 Type A USB Connector



รูปที่ 5.2 Type B USB Connector

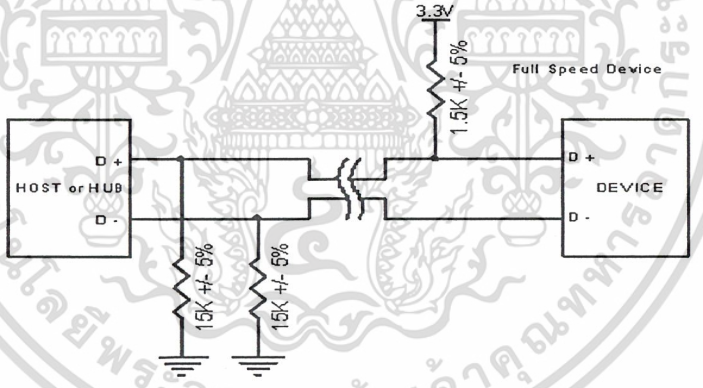
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยแบบ A จะถูกใช้เพื่อต่อกับโฮสหรือฮับ และแบบ B จะถูกต้อเข้ากับอุปกรณ์ โดยสีมาตรฐานของสายภายในสายยูเอสบีคือ

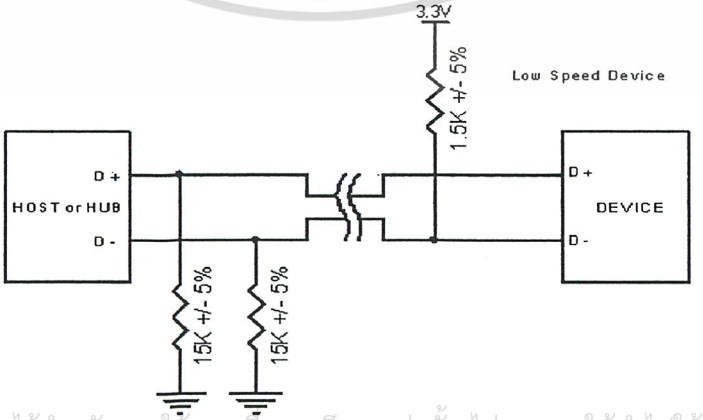
Pin Number	Cable Colour	Function
1	Red	Vbus (% Volts)
2	White	D-
3	Green	D+
4	Black	Ground

5.3 การเลือกใช้ความเร็ว

อุปกรณ์ยูเอสบีจะต้องกำหนดความเร็วโดยการพูลอัพ D+ หรือ D- ไปยัง 3.3 โวล์ท ในอุปกรณ์แบบฟูลสปีด จะต้อง พูลอัพ D+ เพื่อแสดงว่าเป็นอุปกรณ์ชนิดฟูลสปีด ซึ่งจะถูกพบโดยโฮสหรือฮับ โดยถ้าไม่มี พูลอัพ จะถือว่าไม่มีอุปกรณ์ใดๆ ต่อมายังบัส โดยในอุปกรณ์บางตัวที่สามารถกำหนดความเร็วได้ จะใช้อุปกรณ์สารกึ่งตัวนำแทน ตัวต้านทาน ในการพูดอัพ



รูปที่ 5.3 Full Speed Device



รูปที่ 5.4 Low Speed Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

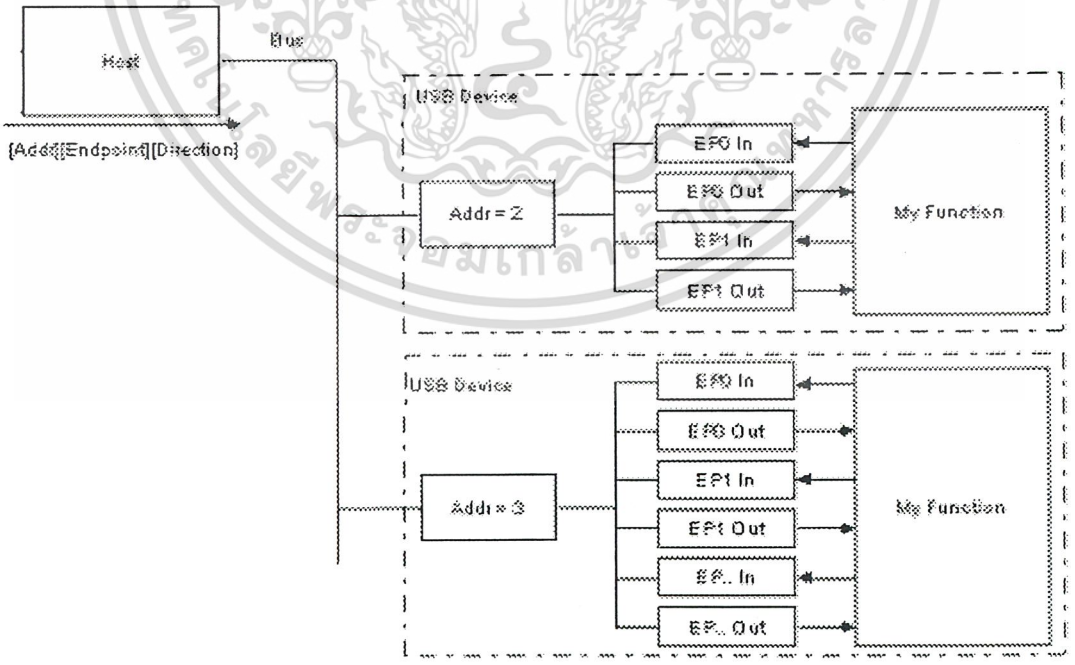
สายส่งข้อมูลของระบบ USB มี 2 สาย สำหรับ สัญญาณ D+ และ D- ในการส่งสัญญาณ สัญญาณจะถูกส่งในลักษณะส่งสัญญาณความต่าง ซึ่งก็ คือกรณีในการส่งสัญญาณ “0” สัญญาณ D- จะมีระดับแรงดันที่สูงกว่า D+ กรณีในการส่งสัญญาณ “1” สัญญาณD- จะมีระดับแรงดันที่ต่ำกว่า D+

การจ่ายกำลังไฟฟ้าสามารถจ่ายให้ได้สูงสุด 5 โวลต์ และ ต่ำสุด 4.75 โวลต์ จ่ายกระแสได้สูงสุด 5 แอมป์ สำหรับอุปกรณ์ต่างๆเฉพาะที่มีการใช้พลังงานไฟฟ้าต่ำ

5.4 การรับส่งข้อมูลของอุปกรณ์ยูเอสบี

เมื่อพูดถึงอุปกรณ์ยูเอสบี เราอาจนึกถึงอุปกรณ์ที่จะใช้งาน เช่น เม้าส์ คีย์บอร์ด ฯลฯ แต่อุปกรณ์ยูเอสบีจะหมายถึง การรับส่งข้อมูลยูเอสบีด้วย ดังนั้นอุปกรณ์ยูเอสบีจึงหมายถึงอุปกรณ์ที่มีการเชื่อมต่อแบบยูเอสบีที่สามารถทำหน้าที่เป็น เม้าส์ คีย์บอร์ด ฯลฯ

ฟังก์ชั่นการทำงานจะประกอบด้วย ลำดับของบัฟเฟอร์ ปรกติจะมีขนาด 8 ไบต์ โดยแต่ละบัฟเฟอร์จะเป็นของ Endpoint แต่ละอัน เช่น EP0 in EP0 out ยกตัวอย่างเช่นโฮสจะส่ง descriptor request อุปกรณ์ก็จะอ่านค่า SetupPacket แล้วตรวจสอบ Address Field ว่าเป็นข้อความของตัวมันเองหรือไม่ ถ้าใช่ก็จะรับและทำตามข้อมูลที่ได้รับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.5 ไดอะแกรมการรับส่งข้อมูลของอุปกรณ์ยูเอสบี

5.5 เอนพอยท์ (Endpoint)

เอนพอยท์จะเป็นได้ทั้งที่มาของข้อมูลหรือปลายทางของข้อมูล ในระบบที่มีโฮสต์เป็นศูนย์กลาง เอนพอยท์จะเป็นปลายทางของช่องการสื่อสารข้อมูลของยูเอสบี เช่นการทำงานในระดับซอฟต์แวร์ ไดรฟ์เวอร์จะส่งส่วนของข้อมูลไปยัง EP1 ของอุปกรณ์ จากการที่ข้อมูลมาจากโฮส ข้อมูลจะไปสิ้นสุดที่บัฟเฟอร์ EP1 out อุปกรณ์ปลายทางจะอ่านข้อมูล และถ้าอุปกรณ์ต้องการจะส่งข้อมูล ก็จะนำข้อมูลไปไว้ยัง EP1 in ซึ่งจะอยู่ในบัฟเฟอร์จนกระทั่งโฮสต์เรียกข้อมูล

5.6 ไปป์ (Pipe)

ในขณะที่อุปกรณ์รับหรือส่งข้อมูลบนเอนพอยต์ต่าง ๆ ซอฟท์แวร์จะส่งข้อมูลผ่านไปป์ โดยไปป์คือการเชื่อมต่อทางลอจิคระหว่างโฮสต์และเอนพอยต์ ไปป์จะมีชุดของพารามิเตอร์ เช่น จำนวนแบนด์วิธที่ต้องการ ลักษณ์ของการส่งข้อมูล เส้นทางของข้อมูล จำนวนชุดข้อมูลมากที่สุดที่ส่งได้ ขนาดของบัฟเฟอร์ โดนไปป์จะมี 2 แบบคือ

สตรีมไปป์ จะไม่มีรูปแบบของข้อมูล สามารถส่งข้อมูลได้หลายแบบ

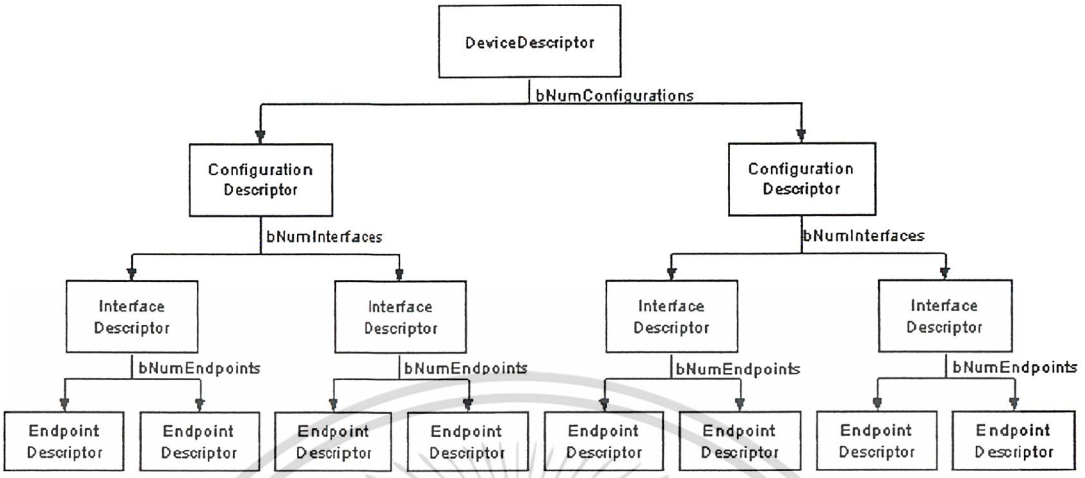
แมสเสจไปป์ มีรูปแบบที่แน่นอน โดยควบคุมโดยโฮสต์ เริ่มต้นข้อมูลด้วยการส่ง รีเควสต์จากโฮสต์

5.7 ยูเอสบี เดสคริปเตอร์ (USB Descriptors)

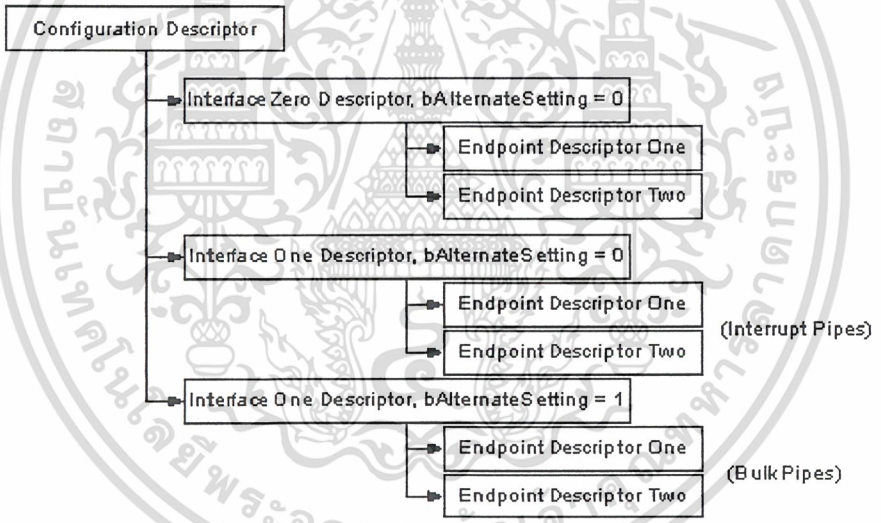
อุปกรณ์ยูเอสบีทุกชนิดจะมีเดสคริปเตอร์ ซึ่งจะให้ข้อมูลกับโฮสต์ เช่นเป็นอุปกรณ์ชนิดไหน สร้างโดยใคร เวอร์ชั่นที่เท่าไหร่ และรองรับการทำงานยูเอสบีเวอร์ชั้นไหน จะกำหนดค่าได้ที่วิธี มีเอนพอยต์เท่าไหร่ ดังนี้

- Device Descriptor ประกอบไปด้วยข้อมูลต่าง ๆ เช่น หมายเลขผู้ผลิต หมายเลขผลิตภัณฑ์
- Configuration Descriptors กำหนดค่าเฉพาะ เช่น กำลังทางไฟฟ้าสูงสุด ในพลังงานจากไหน
- Endpoint Descriptors ใช้บอกค่าลักษณ์การรับส่งข้อมูล ทิศทาง จำนวนข้อมูลสูงสุด
- Interface Descriptors กำหนด หรือ จัดกลุ่ม Endpoint ตามฟังก์ชั่นการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 Device Descriptor



รูปที่ 5.7 Configure Descriptor

5.8 Setup Packet

อุปกรณ์ยูเอสบีทุกอุปกรณ์ต้องตอบสนอง SetupPacket ในไปป์ปรกติ โดน Setup Packet ใช้สำหรับค้นหาและปรับค่าของอุปกรณ์ และนำค่าต่าง ๆ ของอุปกรณ์มาใช้ โดยมีช่วงเวลาในการทำงานดังนี้

- Standard Device Request โดยไม่มีส่วนของข้อมูล ต้องทำงานเสร็จใน 50 ms
- Standard Device Requestมีส่วนของข้อมูล ต้องเริ่มส่งค่ากลับภายใน 500ms หลังจากเรียกข้อมูล

1. แต่ละส่วนข้อมูลต้องส่งภายใน 500ms หลังจากส่วนก่อน สำเร็จ
2. สถานะต้องส่งเสร็จภายใน 50ms หลังจากส่งข้อมูลสุดท้ายเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คำสั่ง SetAddress (มีส่วนของข้อมูล) ต้องถูกประมวลผลและส่งค่ากลับภายใน 50ms และต้องเปลี่ยนค่าภายใน 2ms ก่อน Request ครั้งต่อไป

Offset	BmRequestType	Size	Value	Description
0	BmRequestType	1	Bit-Map	D7 Data Phase Transfer Direction 0 = Host to Device 1 = Device to Host D6..5 Type 0 = Standard 1 = Class 2 = Vendor 3 = Reserved D4..0 Recipient 0 = Device 1 = Interface 2 = Endpoint 3 = Other 4..31 = Reserved
1	BRequest	1	Value	Request
2	WValue	2	Value	Value
4	WIndex	2	Index or Offset	Index
6	WLenhtg	2	Count	Number of bytes to transfer if there is a data phase

ตารางที่ 5.1 Setup Packet

bmRequest : กำหนดทิศทางของข้อมูล

bRequest : กำหนดว่าเป็นการร้องขอข้อมูล

wValue : พารามิเตอร์ที่กำหนดไปพร้อมกับ ชุดข้อมูล

wIndex : พารามิเตอร์ที่กำหนดไปพร้อมกับ ชุดข้อมูล

wLength : จำนวนข้อมูลที่อยู่ในส่วนของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Standard Device Request

bmRequestType	bRequest	wValue	wIndex	Wlength	Data
1000 0000	GET_STATUS (0x00)	Zero	Zero	Two	Device Status
0000 0000	CLEAR_FEATURE (0x01)	Feature Selector	Zero	Zero	None
0000 0000	SET_FEATURE (0x03)	Feature Selector	Zero	Zero	None
0000 0000	SET_ADDRESS (0x05)	Device Address	Zero	Zero	None
1000 0000	GET_DESCRIPTOR (0x06)	Descriptor Type & Index	Zero Lenguage ID	Descriptor Length	Descriptor
0000 0000	SET_DESCRIPTOR (0x07)	Descriptor Type & Index	Zero Lenguage ID	Descriptor Length	Descriptor
1000 0000	GET_CONFIGURATION (0x08)	Zero	Zero	1	Configuration Value
0000 0000	SET_CONFIGURATION (0x09)	Configuration Value	Zero	Zero	None

ตารางที่ 5.2 Standard Device Request

- Clear Feature / Set Feature กำหนดค่า Boolean Feature
- Set Address กำหนดค่า unique address สำหรับ อุปกรณ์ยูเอสบี โดยกำหนดค่าใน wValue
- Set Descriptor/Get Descriptor ส่งค่าที่กำหนดใน wValue กลับมายังโฮสต์
- Get Configutation/Set Configuration ใช้เรียกหรือกำหนดค่า Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Standard Interface Request

BmRequestType	BRequest	WValue	WIndex	WLength	Data
1000 0000	GET_STATUS (0x00)	Zero	Interface	Two	Interface Status
0000 0000	CLEAR_FEATURE(0x01)	Feature Selector	Interface	Zero	None
0000 0000	SET_FEATURE (0x03)	Feature Selector	Interface	Zero	None
1000 0000	GET_INTERFACE (0x11)	Zero	Interface	One	Alternative Setting
0000 0000	SET_INTERFACE (0x11)	Alternative Setting	Interface	Zero	None

- wIndex ใช้กำหนด Interface ที่ต้องการติดต่อ
- Get Status ส่งค่าสถานะของอินเตอร์เฟสกลับ
- Get Interface / Set Interface กำหนดค่า Alternative Interface

Standard Endpoint Request

BmRequestType	Brequest	WValue	WIndex	WLength	Data
1000 0010	GET_STATUS (0x00)	Zero	Endpoint	Two	Endpoint Status
0000 0010	CLEAR_FEATURE	Feature Selector	Endpoint	Zero	None
0000 0010	SET_FEATURE (0x03)	Feature Selector	Endpoint	Zero	None
1000 0010	SYNCH_FRAME (0x12)	Zero	Endpoint	Two	FrameNumber

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

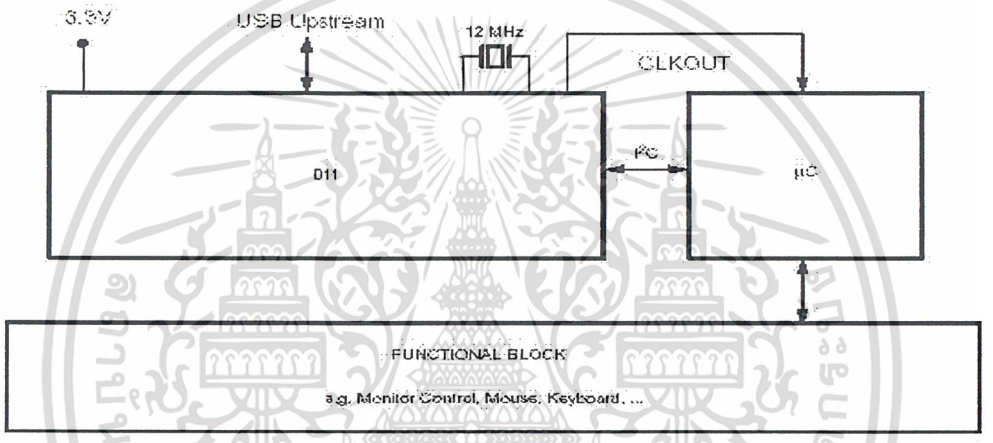
ตารางที่ 5.3 Standard Endpoint Request

- wIndex ใช้สำหรับอ้างถึงเอนพอนต์ และทิศทางของการร้องขอ
- Get Status ส่งค่าสถานะของเอนพอนต์กลับ 2 ไบต์
- Clear Feature / Set Feature กำหนด Endpoint Feature
- Synch Frame แสดงผลของ Endpoint Synchronization Frame

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์ผ่านพอร์ตยูเอสบี

สำหรับในโครงงานนี้เราเลือกไอซี PDIUSBD11 USB Device with Serial Interface โดยบริษัทฟิลิปส์ เซมิคอนดักเตอร์ โดยจะติดต่อกับไมโครคอนโทรลเลอร์ผ่านบัสแบบ I^2C ด้วยความเร็วสูงสุด 568 กิโลไบต์/วินาทีสัญญาณนาฬิกา(CLOCK)สูงสุด 1 MHz และติดต่อกับพอร์ตยูเอสบีโดยมาตรฐานยูเอสบี 1.1



รูปที่ 6.1 แสดงบล็อกไดอะแกรมการต่อ PDIUSB11 กับไมโครคอนโทรลเลอร์

6.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C

I^2C ย่อมาจาก Inter – IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี โดยบัส I^2C ได้รับการพัฒนาขึ้นโดยฟิลลิปส์(Philips) ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่งคือ สายข้อมูล อีกเส้นหนึ่งคือ สายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัสทำได้ง่ายมาก เพียงต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัสมีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรมหรือ SDA (Serial Data Line)ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรมหรือ SCL (Serial Clock Line)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 คุณสมบัติโดยทั่วไปของบัส I^2C

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (bi directional line) ต้องมีการต่อตัวต้านทานพูลอัปกับแรงดัน 5 โวลต์ตลอดเวลา เพื่อให้สายสัญญาณ มีสถานะลอจิกสูงในขณะไม่มีการติดต่อใช้งาน ทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสอง

อัตราการถ่ายทอดข้อมูลบนบัสสูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ และสูงถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง อุปกรณ์ที่ต่อร่วมอยู่บนบัส จะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF

ข้อเด่นอีกประการหนึ่ง คือ สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้

6.3 หลักการของบัส I^2C

อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูแบบของติดต่อบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบการติดต่อบนบัส หรือเรียกว่าโปรโตคอล(Protocol) เพื่อให้ผู้ใช้งานได้ทราบว่า ขณะนี้กำลังติดต่อกับอุปกรณ์ตัวใดอยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง โดยมีข้อตกลงพื้นฐานดังนี้

อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง(Transmitter)

อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ(Reciever) อุปกรณ์บนบัสสามารถเป็นได้ทั้งตัวรับตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัสที่ทำหน้าที่เป็นตัวส่งอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส เรียกว่า มาสเตอร์ (Master)

อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส เรียกว่า สเลฟ(Slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อ คือ

1. การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น
2. ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้นสัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุม

สภาวะที่เกิดขึ้นบนบัส I^2C

มีด้วยกัน 5 สภาวะ ดังนี้

1.บัสว่าง สภาวะนี้เกิดขึ้นเมื่อสถานะของลอจิกบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

เอกสารนี้มีลิขสิทธิ์ ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

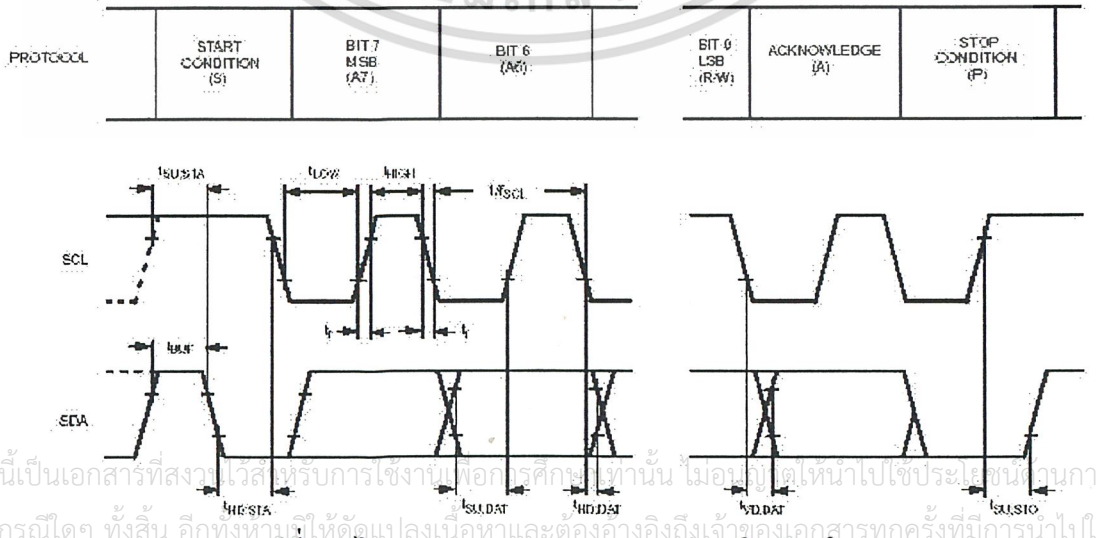
2.เริ่มต้นการถ่ายทอดข้อมูล เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะนี้ว่า สภาวะเริ่มต้น(START)

3.หยุดการถ่ายทอดข้อมูล เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกว่า สภาวะหยุด(Stop)

4.ข้อมูลดำรงอยู่บนบัส สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่กำลังถ่ายทอด เมื่อสาย SCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่า เป็น “0” หรือ “1” ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่ SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ SDA ต้องคงที่ตลอดเวลา ช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะในขณะที่สาย SCL มีสถานะลอจิกสูงนั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นได้

5.รับรู้ข้อมูล เกิดขึ้นหลังจากการถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิต เรียกว่า บิตรับรู้ มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา เพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่ง ทางด้านตัวรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังจะติดต่อขณะนั้นก็จะกำเนิดบิตรับรู้เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลแต่ละไบต์เรียบร้อยแล้ว

ในรูปที่ 6.2 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสภาวะต่างๆ บนบัส ไม่ว่าจะเป็นสภาวะบัสว่าง, เริ่มต้น, ถ่ายทอดข้อมูล, รับรู้, และหยุดการถ่ายทอดข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.2 ไดอะแกรมเวลาแสดงสถานะต่างๆในบัส I^2C

6.4 การเขียนโปรแกรมติดต่อบัส I^2C

<u>การสร้างสภาวะเริ่มต้น</u>

1. เมื่อต้องการติดต่อกับบัส I2C สิ่งแรกที่ต้องทำสำหรับไมโครคอนโทรลเลอร์ ซึ่งเป็นอุปกรณ์มาสเตอร์ คือ การทำให้บัสว่างด้วยการกำหนดขา SCL และ SDA มีลอจิกเป็น "1" ทั้งคู่

2. จากนั้นทำให้ขา SDA มีลอจิก "0" โดย SCL เป็น "1" อยู่

3.กำหนดให้ขา SCL มีลอจิก เป็น "0" ถึงตอนนี้ SDA และ SCL มีสถานะเป็น "0" ทั้งคู่ พร้อมที่จะติดต่อแล้ว

<u>การสร้างสภาวะหยุด</u>

1.เมื่อต้องการหยุดส่งข้อมูลจะต้องส่งสภาวะหยุดออกไป โดยตอนแรกต้องกำหนดให้ขา SCL และ SDA เป็นลอจิก " 0 " ทั้งคู่

2.กำหนดให้ขา SCL มีลอจิกเป็น "1" โดย SDA ยังคงมีลอจิกเป็น "0" อยู่

3.จากนั้นทำให้ขา SDA มีลอจิก เป็น "1" ซึ่งจะทำให้ระบบบัสเข้าสู่สภาวะบัสว่าง พร้อมที่จะรับส่งข้อมูลต่อไป

<u>การส่งข้อมูลลอจิก "0" และลอจิก "1"</u>

การส่งข้อมูลลอจิก "0"

1. ทำให้ขา SDA เป็น "0" สำหรับการส่งข้อมูลลอจิก "0"
2. ทำให้ขา SCL เป็น "1" สำหรับการป้อนสัญญาณนาฬิกา ในขณะที่SDA ยังคงเป็น "0"
3. จากนั้นทำให้ขา SCL กลับมาเป็นลอจิก "0" เหมือนเดิม

<u>การส่งข้อมูลลอจิก "1"</u>

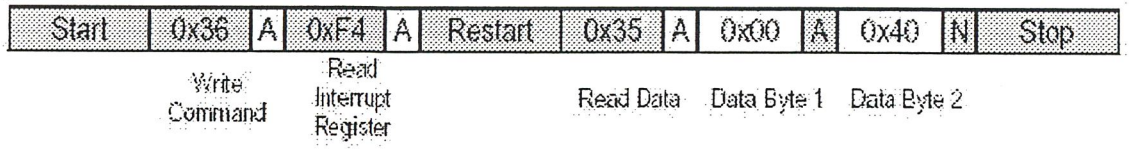
1. ทำให้ขา SDA เป็น "1" สำหรับการส่งข้อมูลลอจิก "1"
2. ทำให้ขา SCL เป็น "1" สำหรับการป้อนสัญญาณนาฬิกา ในขณะที่SDA ยังคงเป็น "0"
3. จากนั้นทำให้ขา SCL กลับมาเป็นลอจิก "0" เหมือนเดิม

6.5 การสื่อสารข้อมูลกับไอซี PDIDSBD11 บนบัส I^2C

การสื่อสารจะอยู่ใน 2 ลักษณะ คือ การส่งคำสั่ง (Command) และการรับ/ส่งข้อมูล (Data) ซึ่งจะใช้ข้อมูลแอดเดรสต่างกันดังนี้

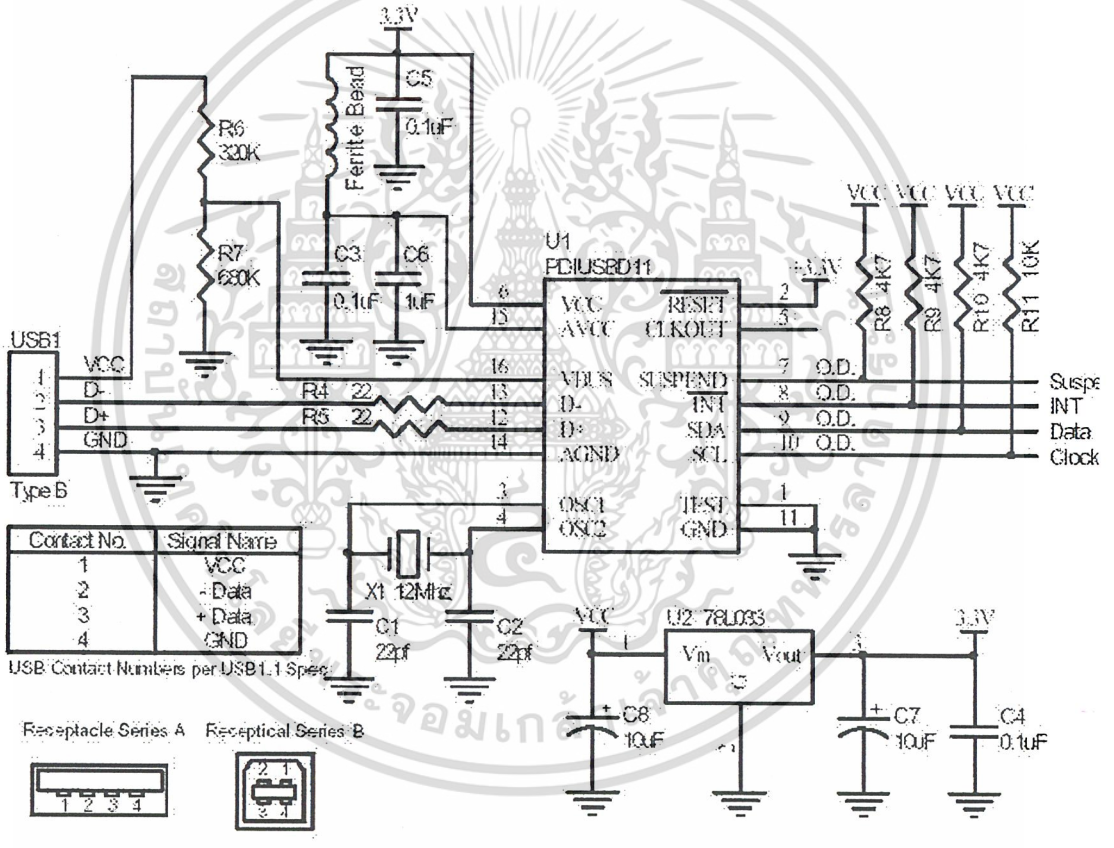
ฟังก์ชันการทำงาน	แอดเดรส
ส่งคำสั่ง	36H
ส่งข้อมูล	34H
รับข้อมูล	33H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.3 ตัวอย่างข้อมูลที่สื่อสารบนบัส I^2C

จากตัวอย่างเริ่มจากส่งข้อมูลคำสั่ง (0X36) คำสั่งอ่านค่าจากอินเทอร์รัปท์รีจิสเตอร์ จากนั้นทำการอ่านข้อมูลที่ได้โดยส่ง 0x35



รูปที่ 6.4 แสดงการนำไอซี PDIDSBD11 ไปใช้ในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

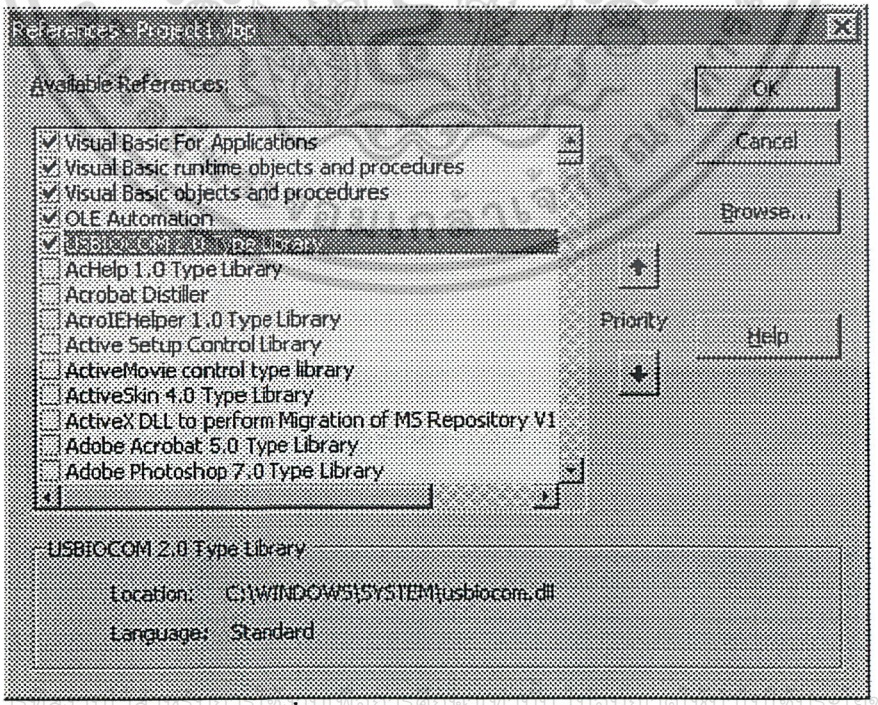
6.6 การเขียนโปรแกรมติดต่ออุปกรณ์ยูเอสบี

การเขียนโปรแกรมเพื่อติดต่ออุปกรณ์ที่เชื่อมต่อผ่านพอร์ตอนุกรม จะสามารถเรียกใช้ไดรว์เวอร์ของระบบปฏิบัติการในเครื่องคอมพิวเตอร์นั้นๆ ได้ทันที เนื่องจากพอร์ตอนุกรมเป็นพอร์ตมาตรฐานในการสื่อสานข้อมูลแบบอะซิงโครนัส แต่ในการติดต่ออุปกรณ์ยูเอสบีซึ่งไม่ได้เป็นอุปกรณ์มาตรฐาน เนื่องจากเป็นอุปกรณ์เฉพาะทาง จึงไม่มีไดรว์เวอร์สำหรับอุปกรณ์นั้น ๆ จึงจำเป็นต้องสร้างไดรว์เวอร์ขึ้นมาสำหรับอุปกรณ์ยูเอสบีแต่ละชนิด เพื่อติดต่อกับอุปกรณ์ยูเอสบี

ในโครงงานนี้เลือกใช้ชุดพัฒนาไดรว์เวอร์ของบริษัท Thesycon Systemsoftware & Consulting จากประเทศเยอรมัน ที่มีชื่อว่า USBID Development Kit ซึ่งสามารถ สร้างไดร์เวอร์สำหรับอุปกรณ์ยูเอสบีต่าง ๆ ได้ทุกชนิด จึงสามารถเขียนโปรแกรมภาษาต่าง ๆ ที่สามารถเรียกใช้ฟังก์ชันตามมาตรฐาน Microsoft's COM Technology เช่น วิชวลเบสิค และเดลไฟ

6.7 การเรียกใช้ยูเอสบีไอโอคอมด้วยวิชวลเบสิค

เมื่อทำการติดตั้งชุดพัฒนา ไดรว์เวอร์ยูเอสบีไอโอจะถูกรีจิสเตอร์เป็นออปเจคสำหรับการพัฒนาซอฟต์แวร์ทันที สามารถเรียกใช้ออปเจคได้โดยเลือก USBIOCOM 2.0 Type Library เพื่อให้สามารถเรียกใช้ ฟังก์ชันของยูเอสบีไอโอ



รูปที่ 6.5 การเรียกใช้ ยูเอสบีไอโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นกำหนดตัวแปลของรูปแบบ USBIOCOMLib.USBIOInterface ชนิด Global เพื่อเรียกใช้ไดร์เวอร์ในการติดต่อกับอุปกรณ์ยูเอสบี

```
Dim WithEvents Device1 As USBIOCOMLib.USBIOInterface2
```

สร้างตัวแปลสำหรับติดต่อกับไดรว์เวอร์ด้วยคำสั่ง

```
Device1 = New USBIOCOMLib.USBIOInterface2
```

ค้นหาอุปกรณ์ยูเอสบีที่อยู่ในกลุ่มของ ยูเอสบีไอโอทั้งหมด เพื่อเรียกใช้ภายหลัง

```
Device1.EnumerateDevices "{325ddf96-938c-11d3-9e34-0080c82727f4}", Devices
```

6.8 ฟังก์ชั่นของยูเอสบีไอโอ

เมื่อกำหนดค่าเริ่มต้นดังกล่าว จะสามารถเรียกใช้ฟังก์ชั่นต่าง ๆ ในออปเจคของยูเอสบีไอโอได้ดังนี้

- **EnumerateDevices** Enumerate USB devices
- **OpenDevice** Open a USB device
- **CloseDevice** Close a USB device
- **DevicePathName** The path name of the driver interface
- **GetDriverInfo** Get version information about the driver
- **IsCheckedBuild** TRUE if a debug build of the driver is running
- **IsDemoVersion** TRUE if a demo version of the driver is running
- **IsLightVersion** TRUE if a light version of the driver is running
- **DeviceOptions** Set or retrieve device options
- **DeviceRequestTimeout** Set or retrieve the request timeout parameter
- **GetDescriptor** Get a USB descriptor
- **GetDeviceDescriptor** Get the USB device descriptor
- **GetConfigurationDescriptor** Get a USB configuration descriptor
- **GetStringDescriptor** Get a USB string descriptor
- **SetDescriptor** Set a descriptor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **AddInterface** Add a USB interface to the configuration
- **DeleteInterfaces** Delete all interfaces

- **SetConfiguration** Configure the USB device
- **GetConfiguration** Get the current configuration value
- **UnconfigureDevice** Set the USB device to unconfigured state
- **SetInterface** Change the alternate setting
- **GetInterface** Get the current alternate setting
- **ClassOrVendorInRequest** Issue a class or vendor IN request
- **ClassOrVendorOutRequest** Issue a class or vendor OUT request
- **SetFeature** Issue a set feature request
- **ClearFeature** Issue a clear feature request
- **GetDevicePowerState** Get current device power state
- **SetDevicePowerState** Set the device power state
- **ResetDevice** Force a USB reset
- **CyclePort** Simulate a disconnect/connect cycle
- **GetStatus** Issue a get status request
- **GetCurrentFrameNumber** Get current USB frame number
- **ErrorText** Translate an error code to a description text
- **Bind** Bind a USBIO COM instance to an endpoint
- **Unbind** Delete a binding
- **StartReading** Start data transfer from an endpoint
- **ReadData** Read data from an endpoint
- **ReadIsoData** Read data from an isochronous endpoint
- **StopReading** Stop data transfer from an endpoint
- **StartWriting** Start data transfer to an endpoint
- **WriteData** Write data to an endpoint
- **GetWriteStatus** Get completion status of a write operation
- **WriteIsoData** Write data to an isochronous endpoint
- **GetIsoWriteStatus** Get completion status of a write operation
- **StopWriting** Stop data transfer to an endpoint
- **ResetPipe** Reset pipe, clear an error condition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **AbortPipe** Abort all pending I/O requests
- **ShortTransferOK** Allow or disallow short transfers
- **EndpointFifoSize** Query the size of the endpoint's FIFO
- **EnablePnPNotification** Enable Plug and Play notification events
- **DisablePnPNotification** Disable Plug and Play notification events
- **GetBandwidthInfo** Get information on the USB bandwidth consumption
- **IsOperatingAtHighSpeed** TRUE if a USB 2.0 device is operating at high-speed (480Mbit/s)
- **SetupPipeStatistics** Enable and configure statistical analysis for a pipe
- **QueryPipeStatistics** Query statistical data related to the pipe

โดยสามารถดูรูปแบบการใช้งานได้ที่ภาคผนวก

ตัวอย่างซอสต์โค้ดสำหรับติดต่ออุปกรณ์ยูเอสบี

```
Dim WithEvents Device1 As USBIOCOMLib.USBIOInterface2
Dim ReadEventCount1 As Long

Private Sub Command1_Click()
   Dim Buffer() As Byte
   ReDim Buffer(0 To 7) As Byte
   Dim writeStatus As Long
   Dim writeStatus2 As Long

   Dim i As Integer
   For i = 0 To 7
      Buffer(i) = &HAB
   Next i

   Device1.Bind 1, writeStatus

   fifoSize = Device1.EndpointFifoSize

   Device1.StartWriting fifoSize, 5, 5, writeStatus, writeStatus2
   If writeStatus = USBIO_ERR_SUCCESS Then
      Debug.Print "Start Writing!"
```

เอกสารนี้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    End If
    ShowError writeStatus
    ShowError writeStatus2

    Device1.WriteData Buffer, 1, writeStatus
    ShowError writeStatus



End Sub

Private Sub Device1_WriteComplete(ByVal Obj As Object)
    Dim writeStatus As Long

    Device1.StopWriting
    Device1.Unbind writeStatus

End Sub

Private Sub Form_Load()
' some local variables
Dim Devices As Long
Dim Status As Long
' Create the USBIO COM object instance.
Set Device1 = New USBIOCOMLib.USBIOInterface2
' Enumerate the available devices.
' We use the USBIO default GUID here.
' In production-level code a custom GUID should be used which is defined in usbio.inf.
Device1.EnumerateDevices "{325ddf96-938c-11d3-9e34-0080c82727f4}", Devices
' check the number of available devices
If Devices >= 1 Then
    ' open the first device (index 0)
    Device1.OpenDevice 0, Status
    If Status = USBIO_ERR_SUCCESS Then
        Form1.State1 = "USB device opened"

    Else
        MsgBox "Cannot open the USB device"
    End If
Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  ' show a message if no device is found
  MsgBox "No USB device connected or USBIO device driver not installed"
End If

  ' add an interface, we assume 0 is a valid interface
  Device1.AddInterface 0, 0, 4096, Status
  If Status = USBIO_ERR_SUCCESS Then
    Form1.State1 = "Interface added"
  End If
  ShowError Status

  ' set the configuration, 0 is ok for single configuration devices
  Device1.SetConfiguration 0, Status
  If Status = USBIO_ERR_SUCCESS Then
    Form1.State1 = "Device configured"
  End If
  ShowError Status

  ' Bind to an IN endpoint provided by a mouse or a keyboard for example.
  ' Typically, the endpoint has the address 0x81.

End Sub

Private Sub StartReading_Click()
  Dim Status As Long

  Device1.Bind Val(EPAddr), Status
  If Status = USBIO_ERR_SUCCESS Then
    Form1.State1 = "EP bound"
  End If
  ShowError Status

  ' get the FIFO size of the endpoint
  fifoSize = Device1.EndpointFifoSize
  ' Start reading from the endpoint
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Device1.StartReading fifoSize, 5, 5, Status
    If Status = USBIO_ERR_SUCCESS Then
        Form1.State1 = "Reading started"
    End If
    ShowError Status

    ' disable the start button
    StartReading.Enabled = False
End Sub

Sub ShowError(ErrorCode As Long)
    If ErrorCode <> USBIO_ERR_SUCCESS Then
        MsgBox Device1.ErrorText(ErrorCode)
        ' End the program in case of an error.
        End
    End If
End Sub

' This function is called if data was received from the device.
Sub Device1_ReadComplete(ByVal Obj As Object)
Dim st As Long
Dim count As Long
' We set the buffer size to 64. This is the maximum FIFO size for bulk.
Dim Buffer(64) As Byte
    ' count the received read events
    ReadEventCount1 = ReadEventCount1 + 1
    Form1.EventCount1 = ReadEventCount1
    ' read data from the COM object (FifoSize bytes)
    'count = fifoSize
    count = 1
    Device1.ReadData Buffer, count, st
    ' check status
    If st = USBIO_ERR_SUCCESS Then
        ' display the data bytes in the edit field
        Data1.Text = ""
        For n = 0 To count - 1
            If (Buffer(n) < 16) Then
                Data1.Text = Data1.Text + " 0" + Hex(Buffer(n))
            Else
                Data1.Text = Data1.Text + " " + Hex(Buffer(n))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

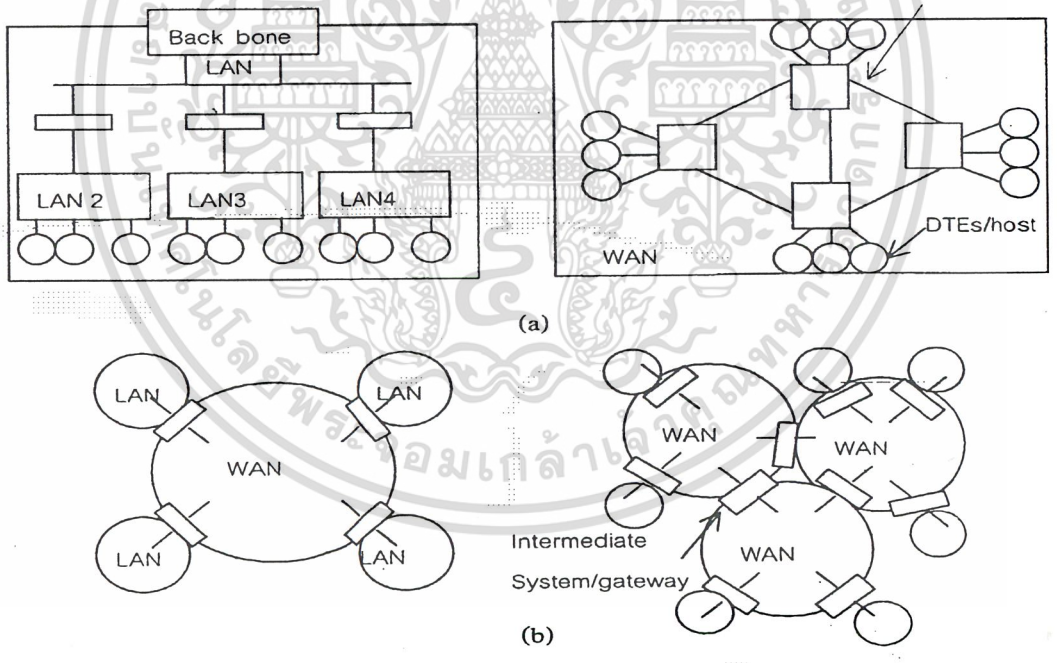
```
        End If
    Next n
    Dim readStatus As Long

    Device1.StopReading
    Device1.Unbind readStatus
    StartReading.Enabled = True

  Else
    MsgBox Device1.ErrorText(st)
  End If
End Sub

Private Sub Form_Terminate()
  ' stop reading
  ' close the device
  Device1.CloseDevice
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
เครือข่ายอินเตอร์เน็ต

อินเตอร์เน็ต (Internet) คือ การที่เครือข่าย 2 เครือข่าย หรือมากกว่า เชื่อมต่อเข้าด้วยกัน และการทำงานเสมือนเป็นเครือข่ายเดียวกัน โยง Network ที่เป็นส่วนประกอบของ Internet คือ Subnetwork (Subnet) ซึ่งอาจเป็นเครือข่าย Local Area Network (LAN) หรือ Wide Area Network (WAN) อุปกรณ์ที่ใช้ในการเชื่อมต่อ 2 เครือข่ายเข้าด้วยกันก็คือ Intermediate System (IS) หรือ Internetworking (IWU) การเชื่อมโยงระหว่างระบบที่แตกต่างกัน จำเป็นต้องมีมาตรฐานการติดต่อกัน ซึ่งเรียกเป็นศัพท์เฉพาะว่าโพรโตคอล (Protocol)

7.1 สถาปัตยกรรม อินเตอร์เน็ต (Internet Architectures)



รูปที่ 7.1 แสดงสถาปัตยกรรมของ Internet

a)Single LAN and WAN

b)Interconnected LAN/WAN

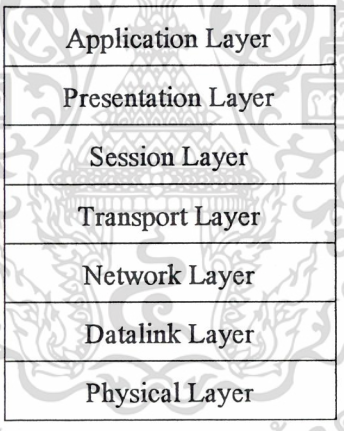
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

ในรูป a) แสดงตัวอย่าง 2 ตัวอย่างของเครือข่ายเดี่ยว (Single Network) ซึ่ง อย่างแรกเป็น Site-wide LAN ซึ่งประกอบขึ้นจากชุดของ LANs ซึ่งถูกต่อเข้ากับเครือข่ายหลัก (Backbone)

อุปกรณ์ที่ใช้ต่อ LAN เข้ากับเครือข่ายหลัก ถ้า LAN ทุกเครือข่ายใช้ระบบเดียวกันก็จะใช้ Bridge ถ้าต่างชนิดกันก็จะใช้ Router ตัวอย่างที่ 2 เป็นตัวอย่างของ WAN เดี่ยว ๆ ในรู) b) แสดงถึงเครือข่ายอินเตอร์เน็ตซึ่งประกอบด้วย Network ทั้ง 2 ชนิดข้างต้น

7.2 OSI โมเดล

องค์ประกอบมาตรฐานสากล ISO (International Organization for Standardization) ได้กำหนดมาตรฐานของเครื่องข่าย โดยจัดแบ่งตามกิจกรรมของเครือข่าย ออกเป็นงานย่อย ๆ และกำหนดโมเดลแบ่งเป็นชั้น ๆ ตามลำดับ เรียกว่ามาตรฐาน OSI (Open System Interconnection) โดยที่จะแบ่งกิจกรรมที่ซับซ้อนในเครือข่ายออกเป็นงานย่อย ๆ ซึ่งช่วยให้การออกแบบ ในการใช้งานเครือข่ายรวมถึงการเชื่อมโยงกัน เป็นไปได้ด้วยความสะดวก และมีวิธีการทำงานอยู่ในกรอบเดียวกัน



รูปที่ 7.2 แสดงการแบ่งเครือข่ายออกเป็น OSI Model

ในแต่ละชั้นของ OSI Model จะมีการติดต่อสื่อสารเป็นชั้น ๆ ตามลำดับลงมาเช่น Application Layer ก็จะสื่อสารกัน Presentation Layer ตามลำดับจนไปถึงขั้นล่างสุดคือ Physical Layer

Application Layer เป็นชั้นบนสุดของ โมเดล เป็นส่วนที่ทำให้การติดต่อสื่อสารระหว่างเครือข่ายกับผู้ใช้เป็นไปตามต้องการ ตัวอย่างแอปพลิเคชั่นของเครือข่ายเช่น ระบบ e-mail, File Transfer, การขอเข้าใช้ระบบคอมพิวเตอร์ในเครือข่าย เป็นต้น Presentation Layer มีการกำหนดหน้าที่ไม่ชัดเจน และมีการนำไปใช้ไม่มาก ซึ่งหน้าที่หลักก็คือ เป็นส่วนที่จัดรูปและนำเสนอข้อมูลให้เป็นไปตามต้องการ รวมถึงการแปลงข้อมูล ในรูปมาตรฐาน ASCII หรือ EBCDIC การลดขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูล (Data Compression) การเข้ารหัส หรือ Session ให้ระบบ คอมพิวเตอร์ทั้ง 2 ฝั่งโดยทำหน้าที่ตั้งแต่เริ่มการติดต่อ ดูแลกานส่งข้อมูลในการติดต่อครั้งนั้น ๆเป็นไปโดยไม่มีปัญหา จนถึงเลิกการติดต่อเมื่อเสร็จงาน

Transport Layer ทำหน้าที่ควบคุมปริมาณ และ รายละเอียดการรับส่งข้อมูล ให้เป็นตามกำหนดที่ตั้งไว้ และจัดการให้การเชื่อมโยงเครือข่ายเป็นไปด้วยความราบรื่น Transport Layer จะเป็นชั้นสุดท้ายที่จัดการเรื่องเส้นทางในหารส่งข้อมูล และจัดการตรวจสอบความผิดพลาดของข้อมูล ซึ่งส่วนของ TCP (Transmission Control Protocol) ในโพรโตคอล TCP/IP ทำงานในระดับนี้

Network Layer ทำหน้าที่ควบคุมวิธีการส่งผ่านข้อมูลระหว่างเครือข่ายให้ถูกต้อง และเป็นไปตามเส้นทางที่กำหนดโดยจะจัดการส่งผ่าน Package ข้อมูลผ่านอุปกรณ์ต่าง ๆไปยังเครือข่ายย่อยได้อย่างถูกต้องตามที่ต้องการ นอกจากนี้ยังจัดการดูแลเส้นทางในการส่งข้อมูล (Routing Table) และกลั่นกรอง Package ข้อมูลที่ส่งไปยังเครือข่ายเดียวกันไม่ให้ข้ามไปยังเครือข่ายอื่น ซึ่งจะช่วยลดปริมาณข้อมูลที่จะวิ่งบนเครือข่าย ได้ส่วนหนึ่ง โพรโตคอล IP, TCP/IP และ IPX เป็นโพรโตคอลที่ทำงานอยู่ใน Layer นี้

Data Link Layer ทำหน้าที่เรียกใช้หรือ กำหนดช่องทาง ในการรับส่งข้อมูลที่ถูกต้อง เช่น Ethernet, Tokenring หรือ FDDI เป็นต้น รวมถึงการลำดับและอัตราการรับส่งข้อมูลหรือ Flow Control และสถานที่ที่จะส่งข้อมูลไป (Address) ทั้งนี้ Data Link Layer จะเป็นชั้นแรกที่จัดการแปลงข้อมูลจาก bit ให้เป็น Package โดยจะมีการเพิ่มข้อมูลเพื่อตรวจสอบผ่าน Checksum เพื่อดูว่าข้อมูลที่ได้รับมาถูกต้องครบถ้วน และถ้าได้รับ Package ข้อมูลที่ไม่ถูกต้องก็จะไม่เอาข้อมูลนั้นมาใช้งาน และบอกให้ต้นทางส่งข้อมูลเดิมมาใหม่

Physical Layer รับผิดชอบดูแลในรายละเอียดในการส่งข้อมูลในด้าน Hardware เช่นการควบคุม Network Interface Card การส่งสัญญาณแบบต่าง ๆการเชื่อมต่อเข้ากับเครือข่ายต่าง ๆ โดยใช้ Physical Layer จะจัดสร้างสัญญาณทางไฟฟ้า หรือสัญญาณเสียง หรือสัญญาณที่จำเป็นในการสื่อสารโดยตรง

เนื่องจาก Network Layer ในแต่ละ End System (ES) จะเป็นตัวจัดการการติดต่อ แบบ end-to-end ของการบริการ Internet wide ไปยังผู้ใช้บริการ (NS-User)

โดย ISO ได้จัด Network Layer เป็น 3 โพรโตคอล (Sub layer) ซึ่งทำงานร่วมกัน เพื่อให้บริการใน Network Layer ได้แก่

- Sub network Independent Convergence Protocol (SNICP)
- Sub network dependent Convergence Protocol (SNDCP)
- Sub network Dependent Access Protocol (SNDAP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ SNICP จะเป็นตัวสนับสนุนการจัดการให้ผู้ใช้บริการ (NS-User) สามารถ Interface กับ Internet ซึ่งมันจะมีหน้าที่เป็นตัวประสานฟังก์ชันต่าง ๆ ที่จำเป็นในการเลือกเส้นทางและการถ่ายทอดข้อมูลของผู้ใช้ข้าม Internet ซึ่งการทำงานของมันไม่ขึ้นอยู่กับคุณสมบัติเฉพาะของเครือข่ายย่อย (Subnet)

SNDAP จะเป็นตัวโพรโตคอลที่ติดต่อกับเครือข่าย (Subnet) ที่มีลักษณะเฉพาะในอินเตอร์เน็ต เช่น X.25 Packet Layer Protocol สำหรับเครือข่าย X.25 ซึ่งใช้บ่อยใน LAN เพราะว่าการบริการและการทำงาน SNDAP แตกต่างจาก Network แบบอื่น ๆ Sublayer ที่อยู่ตรงกลางคือ SNDCP จะเป็นตัวจัดการระหว่าง SNICP และ SNDAP

7.3 รูปแบบมาตรฐานโพรโตคอลของอินเตอร์เน็ต (Internet Protocol Standards)

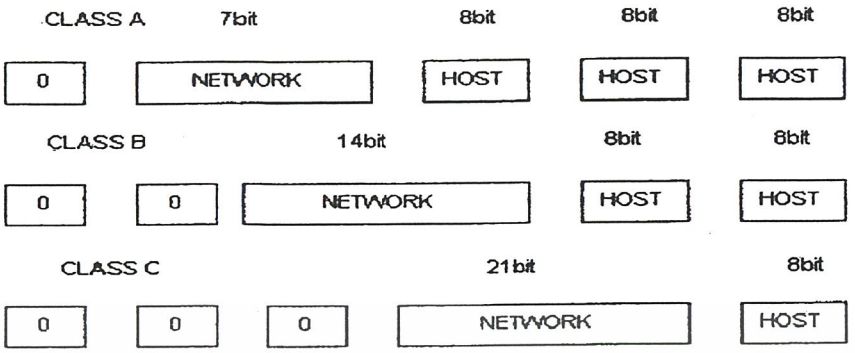
อินเตอร์เน็ตโพรโตคอลที่ถูกใช้ในอินเตอร์เน็ต คือ TCP/IP (Transfer Control Protocol / Internet Protocol) ซึ่งรวมถึง Transport และ Application โพรโตคอล ซึ่งทั้งหมดของ TCP/IP จะกำหนด ให้เหมาะสมกับการใช้เชิงสาธารณะ ซึ่งรูปแบบโดยทั่วไปแสดงดังรูปที่ 2.4 โดย IP เป็น Internetwide Protocol ซึ่งทำให้สอง Transport Protocol ที่ต่างสถานที่กันและต่าง Ess/Hosts กันสามารถแลกเปลี่ยนหน่อยข้อมูล (NSDUS) กันได้ ซึ่งหมายถึงว่า หลาย ๆ Network / Subnet และ ISO/Gateway ที่แตกต่างกันสามารถติดต่อสื่อสารกันได้อย่างสมบูรณ์

7.4 Internet IP

7.4.1 Address Structure

ในศัพท์ของ ISO เมื่อ 2 Network ติดต่อกันด้วย host / ES ที่ต่อกับอินเตอร์เน็ต Network เหล่านี้ ติดต่อกันโดยใช้ Network Service Access Point (NSAP) Address และ Subnet Point of Attachment (SNPA) สำหรับใน TCP/IP ก็จะมี IP Address และ NPA Address ตามลำดับ โดย NPA Address จะแตกต่างกันในแต่ละชนิดของ Network / Subnet และ IP Address ตามลำดับ โดย NPA Address จะเป็นรูปแบบเดียวกัน โครงสร้างของ IP Address ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.3 โครงสร้าง Address ที่ใช้ใน Class ต่างๆ ของเครือข่าย โดยมีทั้งหมด 32 bit

IP Address นี้มีการจัดแบ่งออกเป็นทั้งหมด 5 ระดับ (Class) แต่ที่ใช้งานทั่ว ๆ ไปจะมีเพียง 3 ระดับ โดยคอมพิวเตอร์ที่เชื่อมต่ออยู่มาก จะมีหมายเลขอยู่ใน Class A และลดหลั่นกันมาใน Class B และ Class C ตามลำดับ

จากรูปจะเห็นว่าหมายเลข IP ของ Class A มีตัวแรกเป็น 0 และหมายเลขเครือข่าย (Network Number) ขนาด 7 bit และมีหมายเลขเครื่องคอมพิวเตอร์ (Host Number) ขนาด 24 bit ทำให้ในหนึ่งเครือข่ายของ Class A สามารถ มีคอมพิวเตอร์เชื่อมต่ออยู่ในเครือข่ายได้ถึง $2^{24} = 16$ ล้านเครื่อง แต่ใจ Class A จะมีหมายเลขเครือข่ายได้ 128 เครือข่ายแบบนี้เพียง 128 เครือข่ายเท่านั้น

สำหรับ Class B จะมีหมายเลขเครือข่ายแบบเครื่องคอมพิวเตอร์แบบ 16 bit โดย 2 bit แรกบังคับว่าต้องเป็น 10 (ในเลขฐาน 10) จึงทำให้มีเครื่องคอมพิวเตอร์ในเครือข่ายแบบนี้ได้ 64K เครื่อง

จะเห็นได้ว่า เมื่อเครือข่าย และเครื่องคอมพิวเตอร์ที่ต่ออยู่ใน อินเตอรเนต มีหมายเลข IP Address ให้ใช้อ้างอิงได้ไม่ซ้ำกัน และมีความหมายให้ทราบถึงขนาดเครือข่าย และการติดต่อส่งผ่านข้อมูลจึงกระทำได้โดยไม่สับสน

7.4.2 รูปแบบของข้อมูล (Datagram)

รูปแบบของ IP data unit ก็คือ Datagram ซึ่งโครงสร้างของ Datagram เป็นดังรูป

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Version			Header Length									Type of service			
Total length															
Identification															
D		M	Fragment offset												
Time to live									Protocol						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Header checksum
Source IP Address
Destination IP Address
Option
Data (65536 byte)

รูปที่ 7.4 Internet Datagram Format and Contents

หน่วยข้อมูล IP (IP Datagram) แต่ละหน่วยจะประกอบด้วย ส่วนของข้อมูลที่รับมาจากส่วนของงาน TCP/IP หรือ UDP และส่วนของข้อมูลนำทาง (Header) ซึ่งมีรายละเอียดดังนี้

Version	รุ่นของข้อกำหนด IP
Header Length	ความยาวของข้อมูลนำทาง
Type of Service	วิธีการจัดเก็บข้อมูล
Total Length	ความยาวของหน่วยข้อมูล
Identification, Flags และ Fragment Offset	รายละเอียดที่เกี่ยวข้องกับการแบ่งย่อยข้อมูล ซึ่งจะถูกใช้ในการรวบรวมข้อมูล
Time to live	เวลาสูงสุดที่ใช้ในการเดินทาง ซึ่งกำหนดมาจากต้นทาง โดยเวลาจะลดลงเรื่อย ๆระหว่างทาง ถ้าลดลงถึง 0 หน่วยข้อมูลนั้นจะถูกกำจัดไป
Protocol	ชนิดของข้อมูล UDP หรือ TCP
Header Checksum	ค่าตรวจสอบข้อมูลนำทาง
IP Address	หมายเลข Internetwide IP (NSP) ของเครื่องต้นทางและปลายทาง
Option	ข้อมูลอื่น ๆ เช่น ข้อมูลเกี่ยวกับการรักษาความปลอดภัย บันทึกเส้นทางเดินของข้อมูล เวลาที่ข้อมูลนั้นมาถึง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.5 การแบ่งส่วนของข้อมูลและการประกอบขึ้นใหม่ (Fragmentation and Reassemble)

ขนาดของข้อมูลของผู้ใช้ซึ่งอ้างอิง NSDU มีความจุได้ถึง 64K หรือ 65536 byte ต่อขนาดของหน่วย ข้อมูล (Packet Size) ที่สามารถติดต่อกันในระบบที่ต่างกันสามารถที่ได้ตั้งแต่ 128 byte สำหรับระบบ X.25 Packet Switching จนถึง 8000 byte สำหรับบาง LAN ดังนั้น กระบวรการ Fragmentation และ Reassemble จึงถูกนำมาใช้เพื่อ ทำให้ขนาดของข้อมูลเล็กลง และสามารถส่งไปในระบบได้ และเมื่อถึงปลายทาง IP ก็จะทำการประกอบข้อมูล (Reassemble) ขึ้นมาใหม่ก่อนที่จะส่งผ่านไปยังผู้ใช้

อันดับแรก IP ใน Host ต้นทางจะแยกข้อมูลของผู้ใช้ (NS-User), NSDU เป็น Datagram ซึ่งมี Address กำกับเป็นเฉพาะส่วน ๆ ไปซึ่งจะถูกออกคำสั่งโดย Network ที่ติดต่ออยู่ด้วย และการส่ง Datagram ไปยัง IP ใน Gateway ตัวแรกโดย IP ใน Gateway จะไม่ Reassemble NSDU แต่จะปรับปรุงในขอบเขตที่เหมาะสม และส่ง Datagram ที่ได้รับไปยัง Network ที่สอง (ถ้า Network ที่สองสามารถรองรับขนาดของ Datagram นี้) หรือทำการ Fragment Datagram ให้มีขนาดเล็กลง ซึ่งขั้นตอนนี้จะถูกทำซ้ำที่ Gateway ตัวต่อไป Network ตัวสุดท้ายสามารถรองรับขนาดของ Packet ได้มากกว่า Packet ที่มันได้รับข้อมูล จึงถูกส่งได้โดยตรง โดยที่จะมีการปรับปรุงในบางส่วน Header ของ Datagram เท่านั้น จากนั้น IP ใน Host ปลายทางจะทำการประกอบข้อมูล (Reassemble) ที่ได้รับขึ้นมาใหม่ และส่งผลที่ได้ (NSDU) ไปยังผู้ใช้ (NS-User)

ในการคิดค่าเวลาสูงสุดที่ Host ต้นทางกำหนดให้ Gateway รอ Datagram (NSDU) ระหว่างการ Reassemble ซึ่งก็คือ Time to live ซึ่งจะถูกต้องค่าโดย IP ใน Host ต้นทาง ซึ่งจะมีค่าลดลงเรื่อย ๆ ในแต่ละขั้นตอนของการ Process Datagram ตัวใหม่ ถ้ามันมีค่าถึง 0 ที่จุดใด ๆ ระหว่างการ Process ใน Gateway หรือ Host การ Reassemble ก็จะล้มเหลว และทุก ๆ การ Fragment ที่เกี่ยวกับ NSDU ก็จะถูกตัดทิ้ง

ค่า Time to live ในแต่ละ Datagram จะเป็นจำนวนเท่าของวินาที โดยที่จำนวนของมันจะถูกลดลงโดยแต่ละ IP ซึ่งจะเปลี่ยนแปลงไปตามค่าเวลาจริงในการส่งถ่ายข้อมูลของ Network ที่ติดต่อด้วย

7.6 การเลือกเส้นทาง (Routing)

ในแต่ละ Network (หรือ Subnet) ใน Internet จะมีชนิดของ PA Address ที่แตกต่างกัน ซึ่งระบบ (System Host) หรือ Gateway ที่ถูกต่อเข้ากับ Network จะสามารถส่ง Datagram ไปยังระบบอื่นได้โดยตรงเฉพาะ Network ที่เหมือนกันเท่านั้น ในการเลือกเส้นทาง (Routing) ให้ Datagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้ามไปยังหลาย ๆ Network IP ในแต่ละ Internetwide Gateway ต้องรู้ PA Address ของ Host ปลายทาง

ซึ่งมี 2 วิธีการพื้นฐานที่ถูกใช้ในการหาเส้นทางภายใน Internet คือ Centralized และ Distributed ด้วยวิธีการ Centralized Routing ข้อมูลเกี่ยวกับการเลือกเส้นทางที่เกี่ยวข้องกับแต่ละ Gateway จะถูก Download จาก Site ส่วนกลางโดยใช้ข้อมูล Network และ Special Network Management จะพยายามตรวจสอบ Network และ Host ที่ถูกเพิ่มเข้า และ ถอดออก และข้อบกพร่องที่ถูกวินิจฉัยจะถูกถอดออก ในขณะที่ถูกตรวจสอบ

ด้วยวิธีการ Distributes Routing ทุก ๆ Host และ Gateway จะร่วมกันในการแบ่งปัน ข้อมูลเกี่ยวกับการเลือกเส้นทางจะถูกจดจำไว้โดยแต่ละระบบ ในรูปของ Routing Table ซึ่งจะมี NPA Address ไว้ในการส่งแต่ละ Datagram ซึ่ง Internet จะใช้วิธีการแบบนี้

ขั้นตอนการ Routing ที่เกี่ยวกับ IP ขั้นตอนแรกจะอ่าน IP Address (NSAP) ปลายทางจากภายใน Datagram และใช้มันในการหาการตอบสนอง PA Address ของ Host หรือ Gateway จาก Routing Table ในส่วนที่เพิ่มเติมชุดของ Routing Protocol จะถูกใช้เพิ่มและรักษาส่วนที่อยู่ในแต่ละ Routing Table ในแบบของ Distributed ซึ่งรูปแบบทั่วไปถูกใช้ภายใน Host IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
หลักการเขียนโปรแกรม Winsock ที่ใช้กับ TCP/IP

Application Programming ที่ส่วนมากใช้กันอย่างกว้างขวางสำหรับ TCP/IP Programming คือ Windows Socket Library ซึ่งได้ถูกใช้ในการส่งข้อมูลออกไปยัง Internet สำหรับการใช้สิ่งแวดล้อม TCP/IP Applications และภายหลัง Microsoft ได้คิดค้นเป็น API สำหรับ TCP/IP ภายใต้ระบบปฏิบัติการของ Windows

ในการใช้งานของ Windows Socket จะไม่ถูกจำกัดเฉพาะในเรื่องของ TCP/IP Programming โดย Windows Socket จะแสดงให้เห็นว่าแม้การติดต่อที่เป็นงานหนัก เมื่อใช้ TCP/IP หรือ Protocol อื่น ๆ สำหรับส่งค่าจริง ๆ ของข้อมูล อย่างไรก็ตามในปัจจุบัน Winsock Implementation จะจำกัดการใช้งานของ Windows Socket ในหารติดต่อกับ TCP/IP

โพรโตคอล TCP/IP มีกลไกการทำงานเป็นชั้น หรือ Layer เรียงต่อกัน โดยในแต่ละ Layer จะมีการทำงานเทียบได้กับ OSI Model มาตรฐาน แต่บาง Layer ของ Protocol TCP/IP จะทำงานเทียบกับ OSI หลาย Layer ปนกัน ซึ่งในแต่ละ Layer ของ Protocol TCP/IP จะประกอบได้

Process Layer

Host-to-Host Layer

Internetwork Layer

Network Interface Layer

โดยเมื่อเทียบกับมาตรฐาน OSI model แล้วจะเป็นดังรูป ซึ่งเราเห็นว่าบางกลไกของโพรโตคอล TCP/IP เทียบได้กับมาตรฐาน OSI Model สองชั้น หรือบางกลไกก็จะทำงานคาบเกี่ยวกันระหว่างบางชั้นของ OSI Model ตัวอย่างเช่น กลไกการทำงานของโพรโตคอล TCP/IP ในส่วน Network Interface Layer เมื่อเทียบกับมาตรฐาน OSI Model จะเทียบได้กับ Data Link Layer และ Physical Layer 2 ชั้นรวมกันเป็นต้น ในแต่ละกลไกของโพรโตคอล TCP/IP จะมีโพรโตคอลอื่น ๆ ในชุดของ TCP/IP ร่วมทำงานอยู่ด้วย ซึ่งจะกล่าวโดยละเอียดต่อไป

8.1 Process Layer

จากรูปแสดงลำดับชั้นการทำงานของโพรโตคอล TCP/IP เทียบกับมาตรฐาน OSI Model นั้น ในชั้นบนสุดเรียกว่า Process Layer ทำงาน 2 หน้าที่เทียบได้กับ Application Layer และ Presentation Layer ในชั้นนี้จะรองรับการทำงานของแอปพลิเคชั่นต่าง ๆ ที่ทำงานเป็น Process อยู่ในเครื่องเซิร์ฟเวอร์ให้บริการและเครื่องที่ขอใช้บริการ หรือ ไคลเอนต์ (Client) ซึ่งจะติดต่อกันผ่านโพรโตคอล เฉพาะแอปพลิเคชั่นอีกทีหนึ่ง ตัวอย่างเช่น เมื่อผู้ใช้งานอินเตอร์เน็ต ต้องการโอนถ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

ไฟล์ หรือ Download ข้อมูลจากเครื่องเซิร์ฟเวอร์ที่ให้บริการ เช่น ถ้าผู้ใช้ต้องการเรียกใช้งานคอมพิวเตอร์ที่อยู่ห่างไกลด้วยการใช้โปรแกรม Telnet เครื่องเซิร์ฟเวอร์ที่ให้บริการ Process Telnet ก็จะเรียกใช้โพรโตคอล Telnet เพื่อติดต่อสื่อสารกัน หรือในกรณีที่มีการเรียกใช้โปรแกรม Web Browser เช่น Netscape Navigator เพื่อเรียกดู WebPages ใน Site CNN เครื่องซึ่งในบริการ Web ของ CNN ก็จะมี Process HTTP (Hypertext Transfer Protocol) ทำงานอยู่ และติดต่อกับผู้ใช่ผ่านโพรโตคอล HTTP เป็นต้น

ftp, telnet, mail application	Process Layer
โพรโตคอล TCP UDP	Host-to-Host Layer
โพรโตคอล IP	Internetwork Layer
ไดรเวอร์ Ethernet Token-Ring และ อื่นๆ	Networkinterface Layer

TCP/IP Stack

Application	7
Presentation	6
Session	5
Transport	4
Network	3
Data Link	2
Physical	1

มาตรฐาน OSI

รูปที่ 8.1 แสดง TCP/IP Stack เปรียบเทียบกับมาตรฐาน OSI

การทำงานของแอปพลิเคชั่นต่าง ๆ จะอยู่ที่ชั้น Process Layer นี้ และมีการติดต่อตามแต่โพรโตคอลเฉพาะแล้วแต่ แอปพลิเคชั่นที่ใช้งาน จากการที่ Process Layer ของ TCP/IP รองรับให้โปรโตคอลอื่นทำงานได้หลายโปรเซสและหลายโปรโตคอลได้พร้อมกันนั้น ทำให้ผู้ใช้สามารถเปิดโปรแกรมใช้งานได้หลาย ๆ อย่างพร้อม ๆ กัน เช่น เปิดโปรแกรม Internet Explorer เพื่อเรียกดู WebPages พร้อมเรียกใช้โปรแกรม Outlook เพื่อรับส่ง E-mail ไปพร้อม ๆ กันโดยไม่ต้องรอให้การทำงานอย่างใดอย่างหนึ่งเสร็จสิ้นก่อน

โพรโตคอลหลัก ๆ ที่ทำงานใน Process Layer ซึ่งผู้ใช้มักคุ้นเคยได้แก่ FTP (Files Transfer Protocol), Telnet, HTTP (Hypertext Transfer Protocol), SMTP (Simple Mail Transfer Protocol) นอกจากนี้ยังมี Protocol ที่ทำงานอยู่เบื้องหลัง ซึ่งผู้ใช้ไม่ได้ใช้งานโดยตรง เช่น

โพรโตคอล DNS (Domain Name System) ทำหน้าที่แปลงข้อมูลชื่อ Domain Name หรือชื่อ Web Site ทั้งหลายให้กลายเป็น IP Address

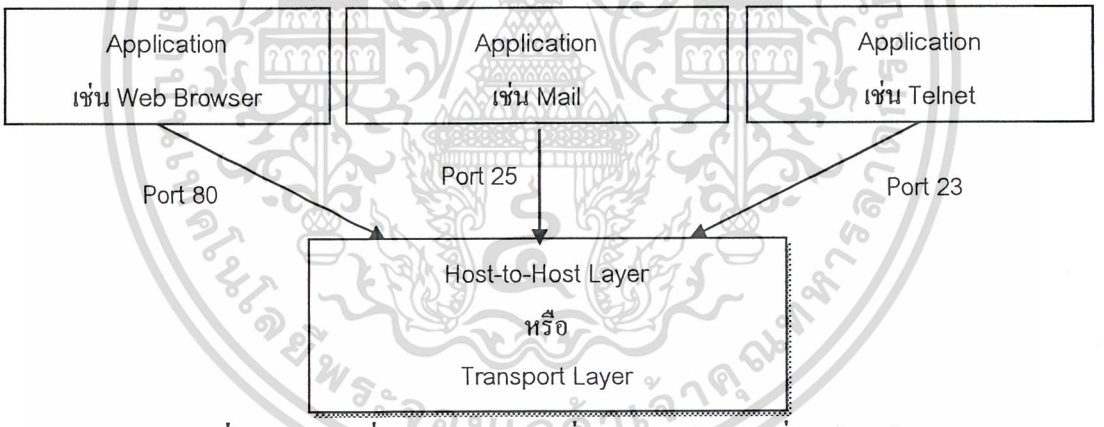
โพรโตคอล SNMP (Simple Network Management Protocol) ใช้ในการควบคุมและตรวจสอบอุปกรณ์ที่อยู่ในเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โพรโตคอล DHCP (Dynamic Host Configuration Protocol) ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายที่เชื่อมอยู่

8.2 Host-to-Host Layer

การทำงานที่ชั้นของ Host-to-Host Layer นี้จะมีบทบาทในการจัดการต่อจาก Process Layer บางครั้งเรามักเรียกชั้น Host-to-Host ว่าเป็น Transport Layer ซึ่งไม่ใช่ชั้น Transport Layer ในมาตรฐาน OSI Model การทำงานของ Host-to-Host Layer นี้จะมีการสร้าง Connection หรือ การเชื่อมต่อกันระหว่าง แอปพลิเคชั่น กับ Host-to-Host Layer โดยจุดที่เชื่อมกันเพื่อรับส่งข้อมูลนี้เรียกว่า Port หรือ Socket (Port ในที่นี้ไม่ได้หมายถึง Port ทาง Hardware) และในแต่ละแอปพลิเคชั่นก็จะสร้างการเชื่อมต่อผ่าน Port ได้พร้อม ๆ กันหลาย ๆ แอปพลิเคชั่นซึ่งการใช้งาน Port ของแต่ละแอปพลิเคชั่นที่อยู่ในชั้น Process Layer จะแตกต่างกันตามหมายเลขที่กำหนดไว้ และแต่ละโพรโตคอลจะมีการใช้งาน Port หมายเลขต่าง ๆ ไม่ซ้ำกัน



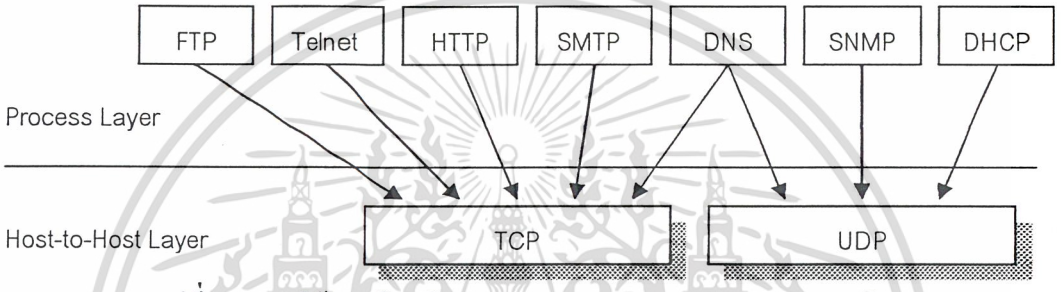
รูป 8.2 เมื่อแอปพลิเคชั่น หรือ โพรเซส สื่อสารกัน ผ่านจุดเชื่อมต่อ หรือ Port

เมื่อแอปพลิเคชั่นทำงานผ่านโพรโตคอลในชั้น Process Layer จะมีการส่งผ่านข้อมูลไปยัง Host-to-Host Layer ที่ชั้นนี้จะมีการเชื่อมต่อผ่าน Port ที่กำหนด ทำให้การรับส่งข้อมูลในแต่ละโพรโตคอล ทำได้ถูกต้อง ถึงแม้ว่า เซิร์ฟเวอร์จะทำงานหลายแอปพลิเคชั่นในเวลาเดียวกันก็ตาม ในชั้น Host-to-Host Layer หรือ Transport Layer ของ TCP/IP นี้จะมีโพรโตคอลทำงานอยู่ 2 โพรโตคอลที่แตกต่างกันคือ โพรโตคอล TCP และ โพรโตคอล UDP (User Datagram Protocol) ในการส่งข้อมูลลงไปที่ชั้นถัด ๆ ไปเราจะเห็นว่าโพรโตคอล TCP และ UDP จะถูกผนึกเข้าไปในโพรโตคอล IP อีกทีหนึ่งและส่งต่อไปยังเครือข่ายอินเตอร์เน็ต ต่อไป

เอกสารนี้มีลิขสิทธิ์ ห้ามมิให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวโพรโตคอล TCP และโพรโตคอล UDP จะมีแอปพลิเคชั่น เฉพาะเผื่อเรียกใช้งานแยกกัน คือ แอปพลิเคชั่นที่ใช้โพรโตคอล FTP, Telnet, HTTP และ SMTP จะมีการส่งข้อมูลโดยเรียกใช้โพรโตคอล TCP ส่วนแอปพลิเคชั่นที่ใช้ในโพรโตคอล SNMP และ DHCP จะส่งผ่านข้อมูลโดยเรียกใช้โพรโตคอล UDP และสำหรับโพรโตคอล DNS นั้นสามารถเรียกใช้งานได้ทั้ง TCP และ UDP สาเหตุที่มีการเรียกใช้โพรโตคอลที่แตกต่างกัน ก็เพราะวิธีการทำงานของโพรโตคอลทั้ง 2 ต่างกันนั่นเอง

โพรโตคอล TCP



รูปที่ 8.3 การใช้งาน โพรโตคอล TCP และ UDP ของ Process หลัก ๆ

8.3 โพรโตคอล TCP

โพรโตคอล TCP (Transmission Control Protocol) เป็นโพรโตคอลที่มีการรับส่งข้อมูลแบบ Stream Oriented Protocol หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณของข้อมูลที่จะส่งไป แต่จะแบ่งข้อมูลเป็นส่วนย่อย ๆ ก่อนแล้วจึงส่งไปปลายทางตามลำดับข้อมูล ในกรณีที่ข้อมูลบางส่วนหายไป ก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทางก็จะจัดเรียงข้อมูลส่วน Datagram ใหม่ให้ต่อเนื่อง และประกอบกลับเป็นข้อมูล ซึ่งจะแยกข้อมูลที่ไม่ถูกต้องออกดังนั้น แอปพลิเคชั่นหรือ Process ใดที่อาศัยการส่งผ่านข้อมูลด้วยโพรโตคอล TCP จะต้องใช้หน่อยความจำและขนาดช่องสัญญาณมากกว่า UDP

การติดต่อระหว่างกันจะต้องเป็นแบบ Connection Oriented คือต้องมีการสร้างการติดต่อกันเป็น Session ทั้ง 2 ด้านก่อน แล้วจึงจะรับส่งข้อมูลได้พร้อม ๆ กัน (Full Duplex) เหมือนการใช้โทรศัพท์ติดต่อกัน เมื่อแอปพลิเคชั่นต้องการส่งข้อมูลจะใช้โพรโตคอลที่เหมาะสมในชั้น Process Layer ติดต่อไปและมีการสร้างข้อมูลผ่าน Port ที่กำหนดเพื่อส่งข้อมูลผ่าน โพรโตคอล TCP

ในระหว่างการรับส่งข้อมูลนี้ โพรโตคอล TCP จะเพิ่มขบวนการตรวจทานข้อมูลขึ้นเพื่อให้ข้อมูลมีความถูกต้องไม่ผิดพลาดไปจากเดิม โดยการส่งสัญญาณตรวจทานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Acknowledgement) และส่งข้อมูลใหม่อีกครั้ง ถ้าปลายทางไม่ได้รับ หรือเกิดความผิดพลาดขึ้นก็จะส่งใหม่

ความน่าเชื่อถือของการส่งข้อมูลโดยใช้ โพรโตคอล TCP จะมีมากกว่าแต่ก็ต้องอาศัยทรัพยากรของระบบมากกว่าเช่นกัน

8.4 โพรโตคอล UDP

ใน Host-to-Host Layer นอกจากจะมีโพรโตคอล TCP ทำงานแล้วยังมี โพรโตคอล UDP (User Datagram Protocol) ที่มีคุณสมบัติแตกต่างกันอยู่ด้วยในการรับส่งข้อมูลผ่านโพรโตคอล UDP จะเป็นแบบที่ทั้ง 2 ด้านไม่จำเป็นต้องอาศัยการสร้างช่องทางเชื่อมต่อกัน (Connection Less) และไม่มีการตรวจสอบความถูกต้องของข้อมูล ในการส่งข้อมูลแต่ละครั้ง แล้ไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิดพลาดในการรับส่งข้อมูล เมื่อเป็นเช่นนี้ แอปพลิเคชั่น หรือ Process ใดที่ต้องอาศัย UDP อาจจะต้องสร้างขบวนการตรวจสอบข้อมูลขึ้นมาเอง

จะเห็นว่าโพรโตคอลชั้นบนขั้นไป ที่ใช้การส่งผ่านข้อมูลโดยโพรโตคอล UDP เช่น โพรโตคอล SNMP หรือ DHCP การส่งข้อมูลเหล่านั้นไม่ต้องรับทราบหรือตรวจสอบข้อมูล ว่าไปถึงถูกต้องหรือไม่ แต่กลไกตรวจสอบจะอยู่ในชั้นโพรโตคอลที่สูงกว่านั่นเอง

ตัวอย่างขั้นตอนกลไกการทำงานโดยใช้โพรโตคอล UDP มีดังต่อไปนี้

1. ในขั้นของ Process Layer เมื่อโปรแกรมควบคุมอุปกรณ์เครื่องข่ายเช่น โปรแกรม Network Management ต่อการส่งข้อมูลไปยังอุปกรณ์ที่ต้องการ แอปพลิเคชั่นนั้นจะติดต่อผ่านโพรโตคอล SNMP ในชั้น Process Layer
2. โพรโตคอล SNMP จะติดต่อกับโพรโตคอล UDP ในชั้นถัดไป เพื่อติดต่อผ่าน Port ที่กำหนด
3. โพรโตคอล SNMP เตรียมข้อมูลที่จะส่ง รวมทั้งที่อยู่ปลายทาง
4. โพรโตคอล SNMP จะส่งผ่านข้อมูลให้โพรโตคอล UDP ที่อยู่ในชั้น Host-to-Host Layer
5. โพรโตคอล UDP ทำหน้าที่ผนึกข้อมูลหรือ Datagram นั้น ไปกับโพรโตคอล IP ในชั้นถัดลงไป เพื่อส่งข้อมูลออกจากเครื่อง

ซึ่งจะเห็นว่ามีกลไกที่ต่างจากการส่งข้อมูลด้วยโพรโตคอล TCP ซึ่งต้องมีการติดต่อกันก่อน แล้วจึงส่งข้อมูลนั้น ๆ

เอกสารนี้มีไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.5 Internetwork Layer

ในชั้นนี้มีหน้าที่ส่งผ่านข้อมูลระหว่างเครือข่ายโดยมี โพรโตคอล ที่ทำงานเป็นกลไกสำคัญในการส่งผ่านข้อมูลไปยังเครือข่ายใด ๆ บน Internet คือ โพรโตคอล IP (Internet Protocol) นอกจากนี้ในชั้น Internetwork Layer ยังมีโพรโตคอลทำงานอยู่ด้วยกันอีก 2 ชนิดคือ โพรโตคอล Internet Control Protocol (ICMP) และ โพรโตคอล Address Resolution Protocol (ARP)

8.6 โพรโตคอล IP

โพรโตคอล IP ทำหน้าที่ให้บริการส่งผ่านข้อมูลที่มาจาก Host-to-Host Layer เพื่อส่งข้ามไปยังเครือข่ายใด ๆ ได้อย่างถูกต้อง เนื่องจากโพรโตคอล IP จะมีข้อมูลตำแหน่ง IP ปลายทางที่จะส่งข้อมูลไปให้ โดยทำงานร่วมกับอุปกรณ์ Router เพื่อส่งข้อมูลข้ามเครือข่ายออกไปได้ ตัวโพรโตคอล IP จะทำงานแบบ Packet Switching คือมีการส่งข้อมูลผ่าน Switch ไปยังปลายทางวงจร Switch นี้อาจเป็น Gateway หรือ Router ในระบบเครือข่ายก็ได้ ซึ่งในข้อมูลของโพรโตคอล IP จะมีข้อมูลหมายเลข IP ปลายทางที่จะส่งข้อมูลไป และเมื่อไปถึงเครือข่ายปลายทางแล้ว จะมีกลไกแปลงหมายเลข IP ให้เป็นหมายเลข Hardware ประจำเครื่องให้ถูกต้องอีกทีหนึ่งด้วยโพรโตคอล ARP

8.7 โพรโตคอล ICMP

หน้าที่หลักของโพรโตคอล ICMP (Internet Control Message Protocol) คือการแจ้งปัญหาหรือแสดงข้อความจากระบบ เพื่อบอกให้ผู้ใช้ทราบว่าเกิดอะไรขึ้น ในการส่งผ่านข้อมูลนั้น ซึ่งปัญหาส่วนมากที่พบคือส่งไปไม่ได้ หรือปลายทางรับข้อมูลไม่ได้ เป็นต้น นอกจากนี้โพรโตคอล ICMP ยังถูกเรียกใช้งานจากเครื่องเซิร์ฟเวอร์ และ Router อีกด้วย เพื่อแลกเปลี่ยนข้อมูลที่ใช้ควบคุม ส่วนรูปแบบการทำงานของโพรโตคอล ICMP นั้นจะทำควบคู่กับโพรโตคอล IP ในระดับเดียวกัน และข้อความต่าง ๆ ที่จะแจ้งให้ทราบจะถูกผนึกอยู่ในข้อมูล IP อีกทีหนึ่ง

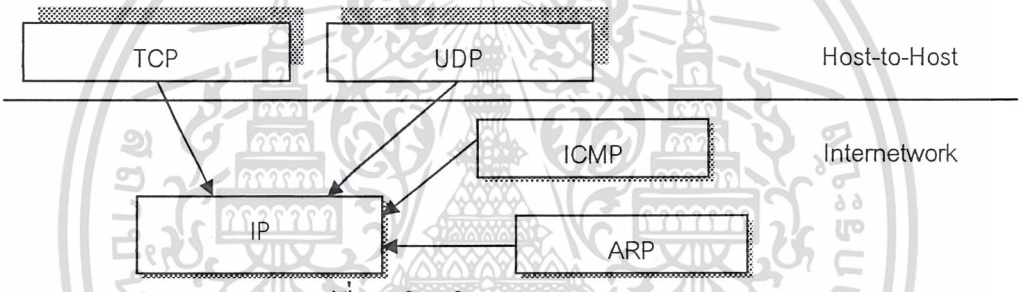
ข้อความที่โพรโตคอล ICMP นั้นแบ่งออกได้เป็น 2 แบบ คือ ICMP error message และ ICMP query อย่างเช่น เมื่อมีการส่งข้อมูลไปยังหลายทางไม่ถูกต้อง หรือปลายทางมีปัญหาไม่สามารถรับข้อมูลได้ ที่ Router จะส่งข้อความแจ้งเป็น ICMP Message ที่ชื่อ Destination Unreachable ให้กับผู้ส่งข้อมูล นอกจากนี้ตัวข้อมูลที่แจ้งข้อความก็จะมีส่วนข้อมูล IP Datagram ที่เกิดปัญหาด้วย ดังนั้นเมื่อผู้ส่งข้อมูลได้รับข้อความแล้ว ก็จะทราบว่าจุดที่เกิดปัญหานั้นอยู่จุดใด

ดังนั้นโพรโตคอล ICMP จึงกลายเป็นเครื่องมือในการช่วยทดสอบเครือข่าย เช่น คำสั่ง ping โดยคำสั่ง ping จะมีการเรียกใช้งานโพรโตคอล ICMP แจ้งเป็นข้อความให้ทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.8 โพรโตคอล ARP

โพรโตคอล ARP (Address Resolution Protocol) ถูกเรียกใช้งานโดย โพรโตคอล IP เพื่อช่วยแปลงหมายเลข IP ไปเป็นหมายเลขฮาร์ดแวร์ปลายทาง ตัวอย่างเช่น Web Server เครื่องหนึ่งต่อเชื่อมอยู่ในเครือข่ายอินเตอร์เน็ต และในการเชื่อมต่อนี้ต้องอาศัย Network Interface Card (NIC) หรือ LAN Card นี้เองจะมีหมายเลขเฉพาะประจำฮาร์ดแวร์ที่ไม่ซ้ำกับใคร เพื่อใช้อ้างอิงในการส่งข้อมูลในเครือข่าย แต่เมื่อมาใช้งานในโพรโตคอล TCP/IP ก็จะต้องมีการกำหนดหมายเลข IP Address ประจำตัวเพื่อใช้อ้างอิงกัน และโพรโตคอล ARP จะทำหน้าที่ แปลงค่าหมายเลขฮาร์ดแวร์จริง ให้ในระดับการทำงานที่ Internetwork Layer ซึ่งในกลไกการแปลงนี้เรียกว่า Address Resolution



รูปที่ 8.4 โพรโตคอล TCP และ UDP

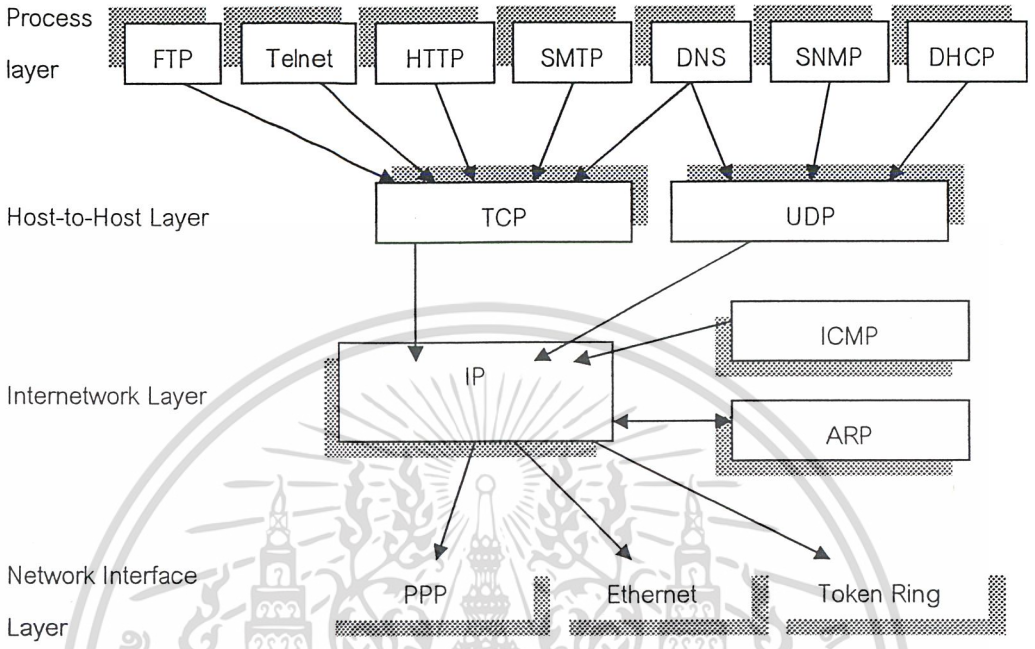
โพรโตคอล TCP และ UDP อาศัยโพรโตคอล IP ที่อยู่ชั้นล่างเพื่อผ่านข้อมูลระหว่างเครือข่าย และในชั้น Internetwork Protocol ยังมีโพรโตคอล ICMP ส่งข้อความแจ้งเตือน และโพรโตคอล ARP ทำหน้าที่แปลงหมายเลข IP ไปเป็นหมายเลข Hardware

8.9 Network Interface Layer

เนื่องจากในด้านกายภาพของเครือข่ายนั้น มีหลายวิธีการและหลายรูปแบบในการเชื่อมต่อระบบให้เป็นเครือข่าย แต่อย่างไรก็ตาม ในเครือข่ายอินเตอร์เน็ตนี้ ข้อมูล หรือ IP Datagram จะถูกถ่ายทอดและส่งผ่านไปยังปลายทางโดยไม่คำนึงถึงรูปแบบการเชื่อมต่อทางกายภาพ ไม่ว่าจะเป็นการใช้เครือข่ายใยแก้วนำแสง หรือเครือข่ายสาย (Unshielded Twist Pair, UTP) การเชื่อมต่อเป็นแบบเครือข่าย Ethernet ธรรมดา หรือเครือข่าย Token Ring, ATM, ISDN ฯลฯ

การทำงานระดับล่างสุดต่อจาก Internetwork Layer จะเป็นการแปลงข้อมูล IP Datagram ให้อยู่ในรูปที่เหมาะสม และแปลงเป็นสัญญาณไฟฟ้าส่งไปยังเครือข่ายต่อไป ซึ่งในชั้น Network Interface Layer นี้ เมื่อเทียบกับมาตรฐาน OSI Model แล้ว จะเป็นการรวม 2 Layer เข้าด้วยกันคือ Data Link Layer และ Physics Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 8.5 โครงสร้างของโพรโตคอล TCP/IP ในแต่ละชั้นหรือ Layer

โครงสร้างของโพรโตคอล TCP/IP ในแต่ละชั้นหรือ Layer จะมีโพรโตคอลหลักทำหน้าที่ต่าง ๆและส่งผ่านข้อมูลไปยังเครือข่ายและออกสู่อินเตอร์เน็ต

โดยสรุปก็คือ โพรโตคอล TCP/IP ทำงานโดยแบ่งเป็นชั้นเทียบกับ OSI Model ได้ กลไกในการทำงานของโพรโตคอล TCP/IP มี 4 ชั้น ซึ่งในชั้นแรกคือ Process Layer ทำหน้าที่ติดต่อกับ Application และ โพรโตคอลที่ Application ต่าง ๆ นั้นเรียกใช้งาน และส่งต่อมาให้ชั้น Host-to-Host Layer เพื่อต่อก่อกันระหว่างเครื่องเซิร์ฟเวอร์ที่ให้บริการ กับเครื่องลูกข่ายซึงมาขอใช้บริการ ในชั้นนี้จะมีการสร้าง Session หรือ การเชื่อต่อระหว่างระบบขึ้น ตามที่แต่ละโพรโตคอลต้องการต่อมา เป็นหารผนึกข้อมูลไปเป็น IP Datagram ที่ชั้น Internetwork Layer โดยอาศัยโพรโตคอล IP เพื่อให้สามารถติดต่อส่งข้อมูลข้ามเครือข่ายไปยังเครือข่ายและเครื่องที่ถูกต้องได้ และสุดท้ายการส่งข้อมูลออกสู่โลกภายนอกต้องอาศัยกลไกในชั้น Network Interface Layer เพื่อแปลงข้อมูลใหม่ เพิ่มข้อมูลที่จำเป็นในการอ้างอิงตำแหน่ง และแปลงข้อมูลเป็นสัญญาณไฟฟ้าส่งออกไปเครือข่าย และอาจจะออกไปยัง Gateway หรือ Router เพื่อข้ามเครือข่ายออกไปยังเส้นทางที่กำหนดไว้ในอินเตอร์เน็ตต่อไป ในแต่ละโพรโตคอลเหล่านี้ก็ะรับผิดชอบหน้าที่ของตนเพื่อผ่านข้อมูลไปยังระดับล่าง และออกสู่เครือข่ายอินเตอร์เน็ตในที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.10 กลไกของโพรโตคอล IP

ในการส่งผ่านข้อมูล หรือ IP Datagram ไปยังเครือข่ายอินเตอร์เน็ตนั้น โพรโตคอล IP จะทำหน้าที่พิจารณาว่าปลายทางในการส่ง IP Datagram นั้นจะไปยังเครื่องข่ายของตนเอง หรือต้องส่งข้อมูลข้ามเครือข่ายไปอีก โดยการพิจารณานี้ โพรโตคอล IP Address จะดูว่า ส่วนที่เป็นหมายเลขเครือข่าย (Network Address) จะเหมือนกับค่าหมายเลขเครือข่ายของ IP Address ต้นทางหรือไม่ถ้าตรงกันแสดงว่าการส่งข้อมูลอยู่ในเครือข่ายเดียวกัน แต่ถ้าค่าต่างกันแสดงว่าต้องส่งข้อมูลไปยังปลายทางที่อยู่คนละเครือข่ายกัน

การส่งข้อมูลภายในเครือข่ายเดียวกันมีกลไกดังนี้

1. โพรโตคอล IP จะเรียกใช้บริการโพรโตคอล ARP (Address Resolution Protocol) เพื่อแปลงหมายเลข IP ปลายทางให้เป็นหมายเลข ฮาร์ดแวร์ เช่น MAC Address
2. เมื่อโพรโตคอล IP ได้รับค่าหมายเลขฮาร์ดแวร์แล้ว ก็จะส่งข้อมูลนั้นไปยังฮาร์ดแวร์ที่ระบุไว้

การส่งข้อมูลข้ามเครือข่ายมีกลไกดังนี้

1. โพรโตคอล IP จะตรวจสอบพบว่าหมายเลย IP Address ปลายทางอยู่คนละเครือข่ายกัน แล้วโพรโตคอล IP จะอ่านค่า IP Address ของ Router เพื่อเตรียมส่งข้อมูลไปที่ Router แทน
2. โพรโตคอล IP จะเรียกใช้ โพรโตคอล ARP เพื่อเปลี่ยนค่า IP Address ของ Router ให้เป็นหมายเลขฮาร์ดแวร์
3. โพรโตคอล IP จะส่งข้อมูล IP Datagram ไปยัง Router ที่กำหนดไว้จากนั้น Router จะส่งข้อมูลข้ามไปยังเครือข่ายปลายทาง

IP Datagram

หัวใจหลักของการส่งข้อมูล TCP/IP คือ IP Datagram ซึ่งจะเป็นชุดของข้อมูล ซึ่งประกอบด้วย Source, Address ปลายทาง, Type of Service Information, User Data และ Error Correction Information

IP Datagram จะประกอบไปด้วยส่วนของ Header Block ของข้อมูล ซึ่งขึ้นอยู่กับชนิดของ Service และความต้องการของผู้ใช้ ในส่วน Header จะประกอบด้วยชุดของ Well-Defined Field

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IP Header

ในส่วนของ IP Datagram หรือ IP Header ส่วนมากจะประกอบไปด้วย 20 bytes ตามรูป

Version (4bit)	Header Length (4 bit)	Type of Service (8 bit)
Packet Length (16 bit)		
Packet Identifier (16 bit)		
Fragmentation Data (16 bit)		
Time to live (8 bit)	Protocol (8bit)	
Header Checksum (16 bit)		
Source Address (32 bit)		
Destination Address (32 bit)		

รูปที่ 8.6 แสดง IP Header

IP Host Address and Routing

IP Host Address จะเป็นส่วนข้อข้อมูล 32 bits ซึ่งมีเพียงหมายเลขเดียวเท่านั้นที่เหมือนกัน ซึ่งใช้ในการติดต่อกับ Internet Host ส่วนของ Gateway จะมี Host อยู่มากมาย

ตามหลักที่ถูกต้องคือ 4 bytes ของ Internet Host Address มักเขียนให้อยู่ในรูปของเลขฐาน 10 ตัวอย่างเช่น 127.0.0.1

Internet Host Address ส่วนมากจะถูกแบ่งออกเป็น 2 ส่วนคือ Network Address และ Actual Host Address โดยเรียงตามความยาวทั้ง 2 ตาม Address โดยมากจะขึ้นอยู่กับการกำหนด bytes ใน Address เป็นส่วนใหญ่

Class A Network Address จะสังเกตได้ว่า Byte แรกจะอยู่ระหว่าง 0 – 127 และจะประกอบไปด้วย 8 bit Network จะประกอบด้วย 24 bit Host Address เนื่องจาก Address เริ่มต้นที่ 0 และ 127 จะถูกสงวนไว้ ซึ่งสามารถมีค่าสูงสุดคือ 126 ในClass A Subnet ใน Class A นี้สามารถมีได้ทั้งหมด 16,777,214 Hosts (Address จะอยู่ในรูป nnn.0.0.0 และ mmm.0.0.0 จะถูกสงวนเอาไว้)

Class B Network Address สังเกตได้จากตัวเลขหลักแรกสุดอยู่ระหว่าง 128 – 191 โดยที่ Class B Network จะประกอบไปด้วย 16 bit Network Address และ 16 bit Host Address และสามารถมีได้ 16,383 Subnet และ Class B Subnet สามารถที่จะบรรจุได้ถึง 65,534 Host Address

Class C Network Address จะสังเกตได้จากตัวเลขหน้าสุดอยู่ระหว่าง 192 – 223 ซึ่งจะมีได้ 2,097,152 Subnet โดย Class C Subnet จะสามารถรองรับได้ 254 Host

Class D Network Address โดยทั่วไป Byte แรกที่อยู่ระหว่าง 224-255 จะถูกสงวนเอาไว้เพื่อ IP Multitasking ซึ่ง Winsock จะไม่สามารถเข้าไปใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host Names

ในระหว่างการประดิษฐ์คิดค้น Internet พบว่าตัวเลขที่จะนำมาแทน Host Address จะมีไม่เพียงพอและยากต่อการจดจำ และถ้ามีการเปลี่ยนแปลงอันเนื่องจากหลาย ๆ สาเหตุ อาจทำให้เกิดความยุ่งยาก ดังนั้นระบบ Naming System จึงถูกสร้างขึ้นมาเพื่อใช้ Map ตัวเลขของ IP Address ให้เป็น Memories Host Name

8.11 การเขียนโปรแกรมส่งข้อมูลผ่าน TCP/IP ด้วย Visual Basic

เนื่องจากในโครงงานนี้ ได้ใช้ภาษา Visual Basic ในการเขียนโปรแกรม เพื่อควบคุมใช้งานอุปกรณ์ และต้องการที่จะควบคุมผ่านระบบอินเตอร์เน็ต ดังนั้นจึงต้องส่งข้อมูลผ่านเน็ตเวิร์ค ซึ่งในการติดต่อผ่านระบบเน็ตเวิร์คนั้น มีหลายมาตรฐานในการเชื่อมต่อ ในดังนั้นในโครงงานนี้จึงเลือกโพรโตคอล TCP/IP เนื่องจากเป็นโพรโตคอลที่ใช้กันอย่างแพร่หลาย และเป็นมาตรฐานเดียวกับอินเตอร์เน็ตในปัจจุบัน

Winsock

ระบบปฏิบัติการยูนิกส์จะมีความสามารถอย่างหนึ่งที่เรียกว่า Socket ซึ่งก็คือ การที่โปรแกรมต่าง ๆ จะสามารถสื่อสารข้อมูลระหว่างกันได้โดยไม่จำเป็นว่าโปรแกรมนั้นจะต้องรันอยู่บนคอมพิวเตอร์เครื่องเดียวกัน

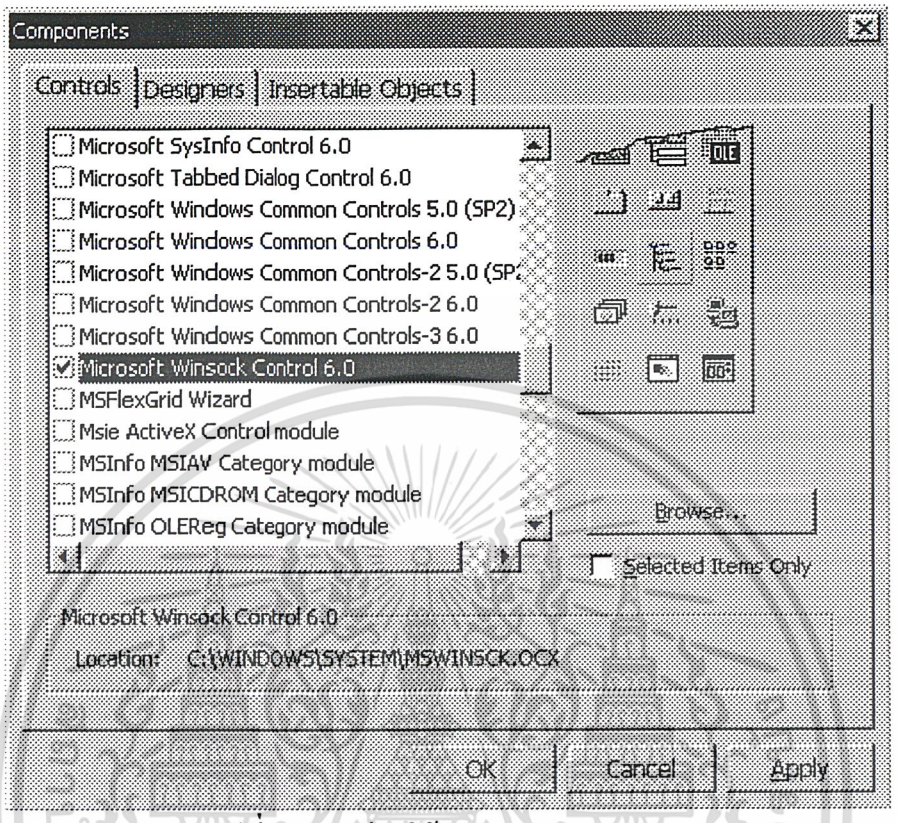
การติดต่อผ่าน Socket ก็คือ โปรแกรมส่วนที่ทำหน้าที่เป็น Server จะสร้าง Socket (ซึ่งถ้าแปลตรงตามตัวก็คือ ช่องเสียบ) โดยสามารถมีได้หลายช่อง จากนั้นเมื่อ Client ต้องการติดต่อสื่อสาร ก็จะติดต่อมาที่ Socket ที่ฝั่ง Server เตรียมไว้ให้ เมื่อเชื่อมต่อกันเสร็จก็พร้อมที่จะแลกเปลี่ยนข้อมูลระหว่างกันได้ โดยเป็นการสื่อสารแบบ 2 ทางชิดเต็มรูปแบบ (2 Way Full-Duplex)

การใช้งาน Winsock ในระบบปฏิบัติการของ Microsoft

ในระบบปฏิบัติการของ Microsoft เช่น Windows95/98/ME/NT/2000/XP ได้เตรียมการทำงานในลักษณะ Socket เอาไว้ให้ นักพัฒนาแอปพลิเคชั่นได้ใช้งาน โดยตั้งชื่อว่า Winsock โดยมีหลักการคล้ายคลึงกับ Socket ของ ระบบปฏิบัติการยูนิกส์

ในมุมมองของการพัฒนาแอปพลิเคชั่นแล้ว Winsock ถือเป็น API (Application Program Interface) หรือ ชุดฟังก์ชันมาตรฐาน ซึ่งออกแบบมาเพื่อให้การเขียนโปรแกรม ใช้งานอินเตอร์เน็ตเป็นเรื่องง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.7 การเรียกใช้งาน Winsock Control

การใช้งาน Winsock Control

ใน Visual Basic ได้เตรียม ActiveX Control ที่มีชื่อว่า Winsock Control ไว้ให้เราใช้งานซึ่งพร้อมให้เราใช้งานเพื่อสร้างการเชื่อต่อระหว่างเครื่องคอมพิวเตอร์ 2 เครื่องในเครือข่าย ทำให้สามารถแลกเลี่ยนข้อมูลระหว่างกันได้ สำหรับ Winsock Control จะถูกเก็บไว้ในไฟล์ MSWINSCK.OCX

โหมดการทำงานของ Winsock Control

ใน Winsock Control สามารถรองรับการทำงานได้ 2 โหมด คือ TCP และ UDP ซึ่งสามารถเลือกลักษณะการทำงานได้ โดย

TCP (Transmission Control Protocol) โดยเป็นการทำงานในลักษณะ Connection-Base เปรียบเสมือนการทำงานของโทรศัพท์ คือจะต้องมีการเชื่อมต่อการสื่อสารก่อน ถึงจะสามารถแลกเปลี่ยนข้อมูลได้ จนกระทั่งยุติการสื่อสาร ซึ่งในระว่างที่เชื่อมต่อกันอยู่ จะมีการตรวจสอบการทำงาน ทำให้การสื่อสารแบบนี้มีประสิทธิภาพมากกว่า โดย TCP จะเหมาะกับการส่งข้อมูลที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการตรวจสอบความถูกต้อง ของข้อมูล เช่นการส่งภาพ เสียง ในระบบเครือข่าย แต่เนื่องจากมีการตรวจสอบความถูกต้องของข้อมูล จึงทำให้ใช้ทรัพยากรของระบบค่อนข้างสูง

UDP (Datagram Protocol) เป็นการทำงานแบบ Connection Less คือไม่มีการสร้างการเชื่อมต่อ จึงทำให้ไม่เปลืองช่องสัญญาณในการติดต่อ เนื่องจากไม่มีการสร้างการเชื่อมต่อไว้ แต่ไม่สามารถตรวจสอบได้ ว่าปลายทางได้รับข้อมูลหรือยัง โดย UDP เหมาะสำหรับส่งข้อมูลที่ไม่สำคัญนัก เช่น สถานะการทำงาน โดยเป็นข้อมูลขนาดเล็ก จึงทำให้ UDP ใช้ทรัพยากรของระบบ น้อยกว่า TCP

ในโครงงานนี้เลือกที่จะใช้โหมดการทำงานแบบ TCP เนื่องจากต้องการความถูกต้องแม่นยำของข้อมูลที่จะแลกเปลี่ยน

Property ที่สำคัญของ Winsock Control

Protocol	เป็นการเลือกโพรโตคอลที่จะใช้งาน TCP / UDP
Local IP	หมายเลข IP Address ของเครื่องที่รันโปรแกรม
Local Port	กำหนด Port ที่จะรอรับการเชื่อมต่อ
Remote HostIP	หมายเลข IP Address ของเครื่องปลายทาง
Remote Port	หมายเลข Port ของเครื่องปลายทาง

Method ที่สำคัญของ Winsock

Listen	เป็นการสร้าง Socket เพื่อให้เครื่องคอมพิวเตอร์เครื่องอื่น ๆ ติดต่อเข้ามาได้
Connect	เป็นการสร้างการติดต่อสื่อสารไปยังเครื่องที่เปิด Socket รอรับการติดต่อ โดยจะต้องระบุหมายเลข IP Address และ Port ของเครื่องที่รอรับการติดต่อ
Accept	เมื่อมีการติดต่อเพื่อต้องการ Connect จากเครื่อง Client ที่เครื่อง Server จะได้รับการ Request Connect ดังนั้น ถ้าอนุญาตให้เครื่อง Client เข้ามาเชื่อมต่อได้ ต้อง Accept เพื่อสร้างการเชื่อมต่อ
Close	เป็นการยุติการเชื่อมต่อ
SendData	เป็นการส่งข้อมูลในตัวแปรที่กำหนดไปยัง Buffer ของการส่งข้อมูล
GetData	เป็นการรับข้อมูลที่อยู่ใน Buffer เข้ามาเก็บในตัวแปรที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9
การทดลอง

การทดลองที่ 1 การใช้งาน Server Application เพื่อควบคุมการทำงานของมอเตอร์ ผ่านพอร์ตอนุกรม

วัตถุประสงค์

1. ตรวจสอบการรับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ผ่านพอร์ตอนุกรม
2. ศึกษาการทำงานของ Server Application

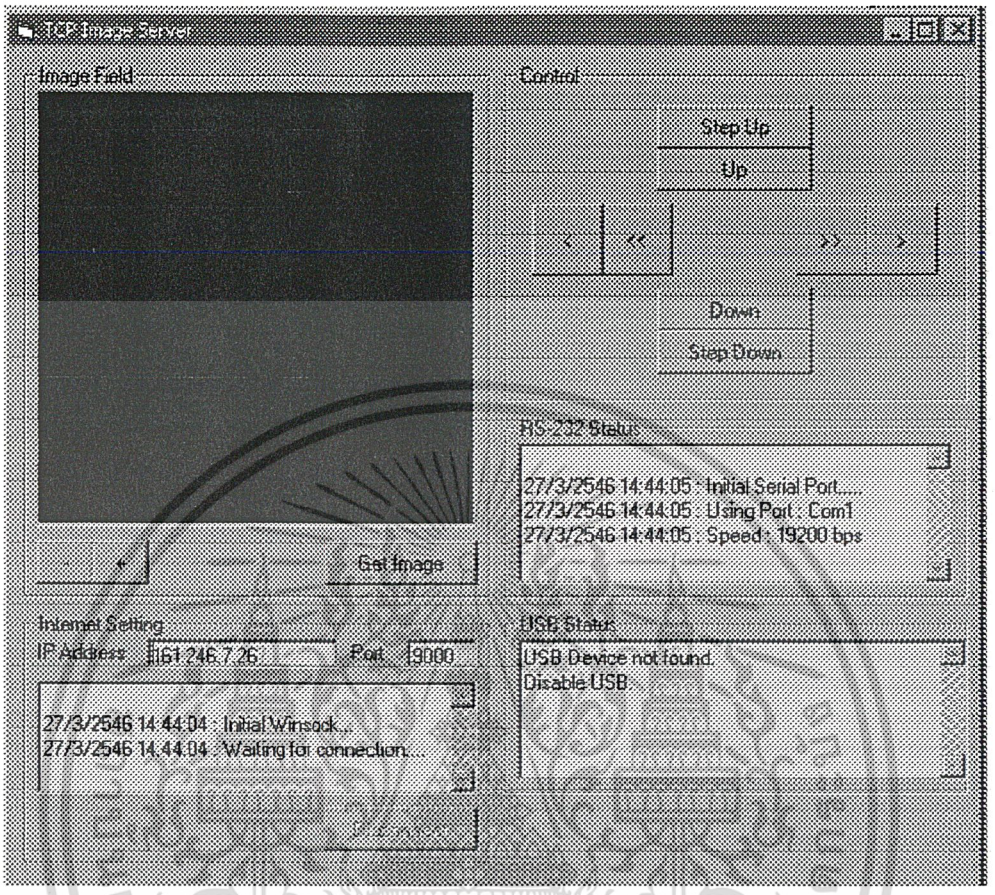
ขั้นตอนการทดลอง

1. เชื่อมต่ออุปกรณ์เข้ากับเครื่องคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม
2. เรียกใช้โปรแกรม Server Application
3. ทดลองส่งคำสั่งควบคุมมอเตอร์
4. สังเกตและบันทึกผล
5. วิเคราะห์ผลการทดลอง

ผลการทดลอง

เมื่อเรียกใช้โปรแกรม Server Application โปรแกรมจะไม่สามารถติดต่อเข้ากับอุปกรณ์ USB ได้ โปรแกรมจะเลือกใช้การติดต่อผ่านพอร์ตอนุกรมแทน โดยเลือกใช้พอร์ต Com1 ที่ความเร็ว 19200bps ในการเชื่อมต่อ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปผลการทดลอง 1.1

ทดลองควบคุมการทำงานของมอเตอร์โดยการ คลิกเม้าบนคำสั่งที่ต้องการ และสังเกตผลที่เกิดขึ้นกับ มอเตอร์แต่ละตัว

ตารางผลการทดลองที่ 1 ผลการทดลองการควบคุมการทำงานของ Stepper Motor

คำสั่ง	ผลที่เกิดขึ้น	หมายเหตุ
Up	มอเตอร์หมุนขึ้นจนกว่าจะปล่อยเม้า	มอเตอร์ไม่ทำงานเป็นบางครั้ง
Down	มอเตอร์หมุนลงจนกว่าจะปล่อยเม้า	มอเตอร์ไม่ทำงานเป็นบางครั้ง
<<	มอเตอร์หมุนซ้ายจนกว่าจะปล่อยเม้า	-
>>	มอเตอร์หมุนขวาจนกว่าจะปล่อยเม้า	-
Step Up	มอเตอร์หมุนขึ้นเป็นช่วงสั้น ๆ	มอเตอร์ไม่ทำงานเป็นบางครั้ง
Step Down	มอเตอร์หมุนลงเป็นช่วงสั้น ๆ	มอเตอร์ไม่ทำงานเป็นบางครั้ง
<	มอเตอร์หมุนซ้ายเป็นช่วงสั้น ๆ	-
>	มอเตอร์หมุนขวาเป็นช่วงสั้น ๆ	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์ผลการทดลอง

จากตารางผลการทดลองพบว่า ไม่สามารถสั่งการมอเตอร์ชุดควบคุมการหมุนขึ้น/หมุนลงได้ในบางครั้ง เนื่องจากไมโครคอนโทรลเลอร์ไม่ได้รับคำสั่งในการควบคุมการทำงาน เพราะจังหวะการทำงานระหว่าง ไมโครคอนโทรลเลอร์ไม่เท่ากัน เนื่องจากข้อจำกัดด้ายสายข้อมูล ซึ่งถูกทำให้มีจำนวนน้อยที่สุดเพราะต้องให้หมุนได้รอบ (360 องศา) จึงทำให้ไม่สามารถใช้สัญญาณนาฬิกาเดียวกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 2 การใช้งาน Server Application เพื่อรับภาพจากไมโครคอนโทรลเลอร์ผ่านพอร์ตอนุกรม

วัตถุประสงค์

1. ตรวจสอบการทำงานในส่วนของการรับส่งข้อมูลภาพของไมโครคอนโทรลเลอร์และโปรแกรม Server Application
2. ศึกษาการทำงานในส่วนของการรับข้อมูลภาพ

ขั้นตอนการทดลอง

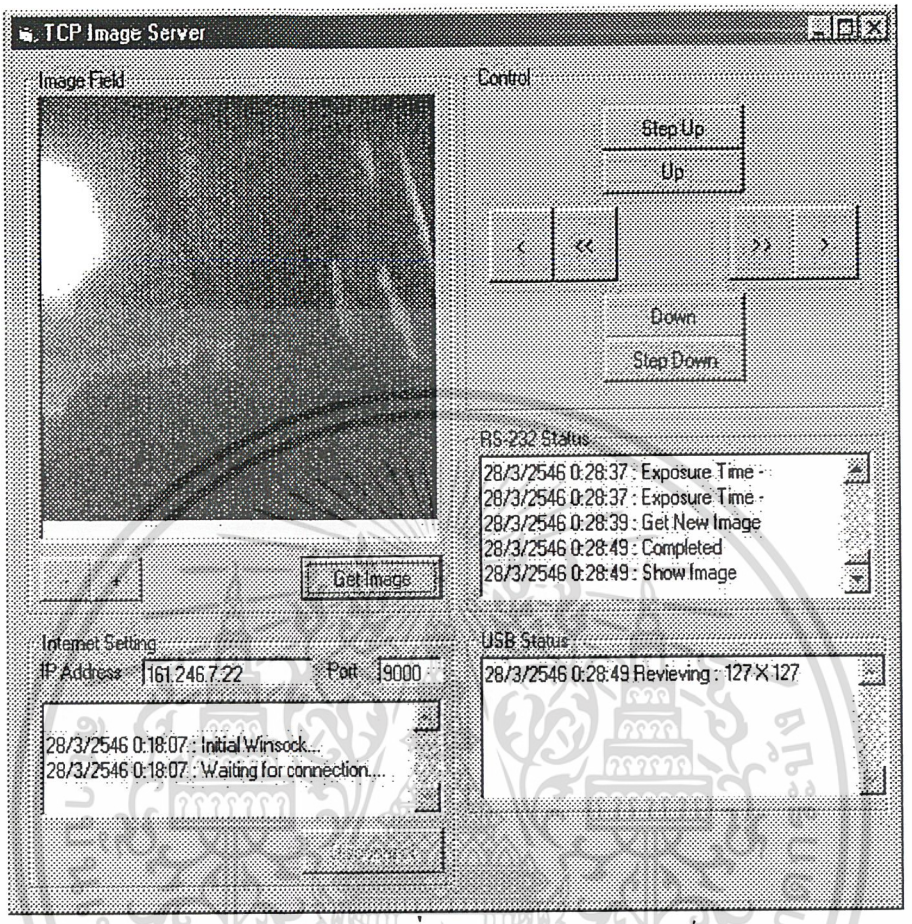
1. เชื่อมต่ออุปกรณ์เข้ากับเครื่องคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม
2. เรียกใช้โปรแกรม Server Application
3. ทดลองส่งคำสั่งรับข้อมูลภาพ
4. สังเกตและบันทึกผล
5. วิเคราะห์ผลการทดลอง

ผลการทดลอง

ตารางผลการทดลองที่ 2

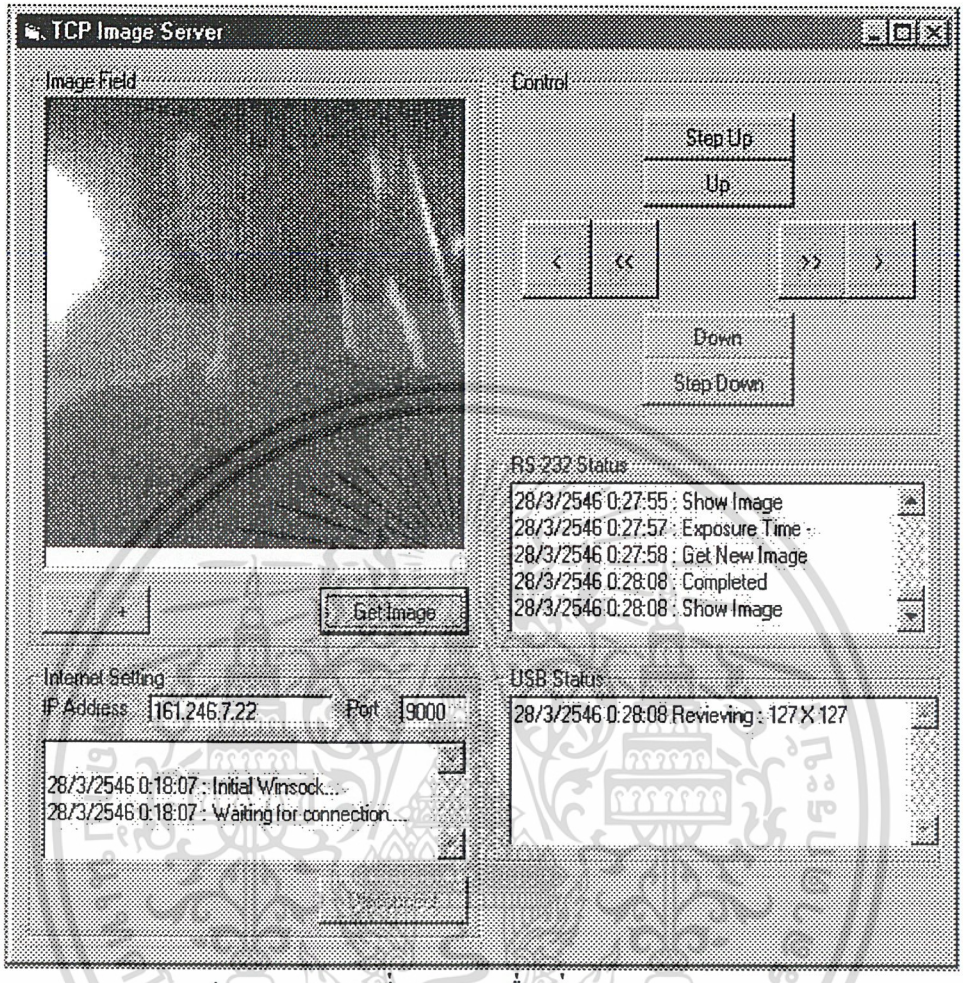
คำสั่ง	ผลที่เกิดขึ้น	หมายเหตุ
Get Image	ได้รับข้อมูลภาพจากไมโครคอลโทรลเลอร์	ไม่ได้รับข้อมูลภาพในบางครั้ง
+	ข้อมูลภาพครั้งต่อไปมีความสว่างมากขึ้น	ความสว่างไม่เปลี่ยนแปลงในบางครั้ง
-	ข้อมูลภาพครั้งต่อไปมีความสว่างลดลง	ความสว่างไม่เปลี่ยนแปลงในบางครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



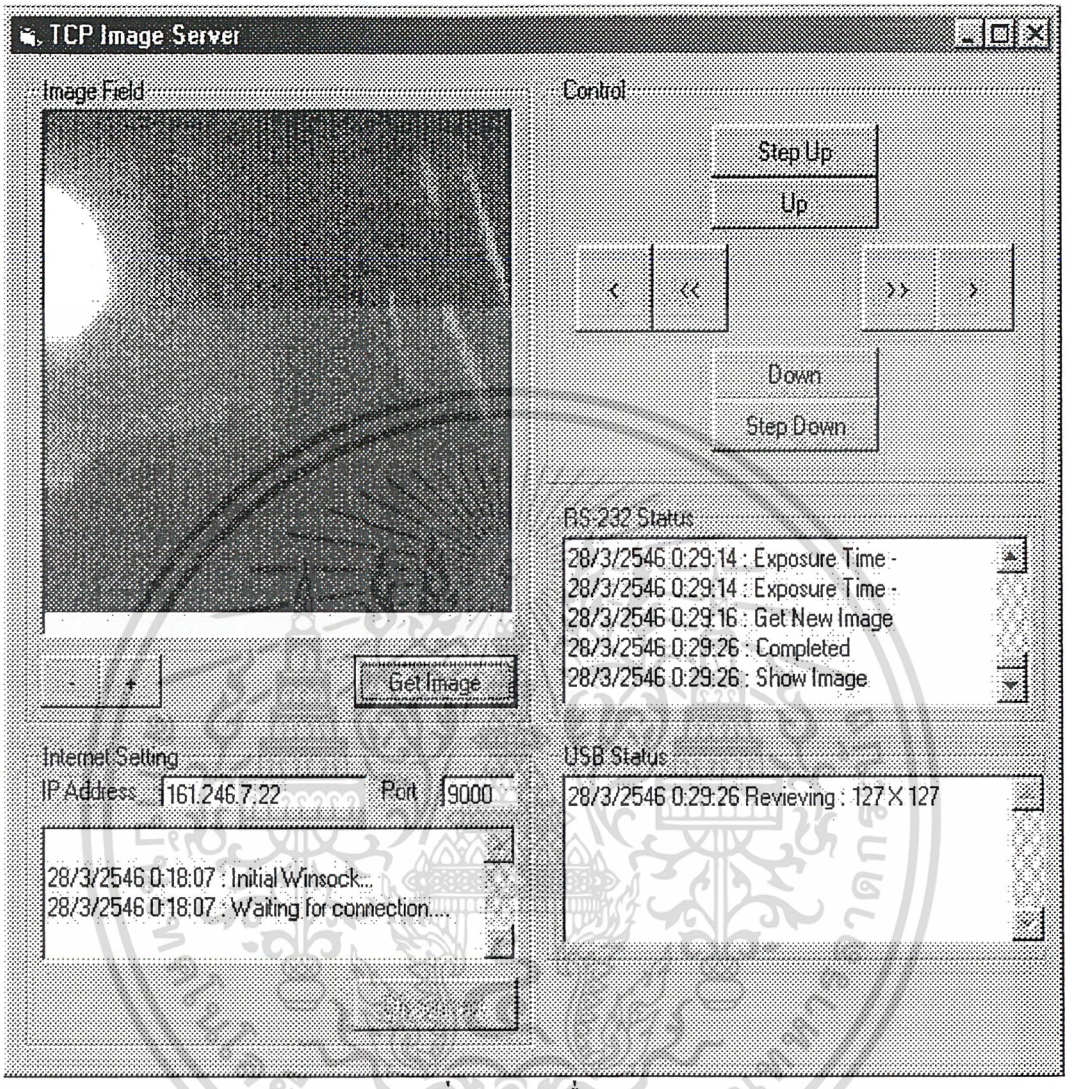
รูปผลการทดลองที่ 2.1 การรับภาพสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปผลการทดลองที่ 2.2 ภาพเมื่อเพิ่ม Exposure Time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปผลการทดลองที่ 2.3 ภาพเมื่อลด Exposure Time

วิเคราะห์ผลการทดลอง

การทดลองนี้เป็นการส่งคำสั่งไปยังไมโครคอนโทรลเลอร์ชุดบน ซึ่งมีปัญหาในการรับคำสั่งจากไมโครคอนโทรลเลอร์ชุดล่าง จึงเกิดปัญหาเช่นเดียวกับการทดลองที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 3 การใช้งาน Server Application เพื่อควบคุมการทำงานของมอเตอร์ ผ่านพอร์ตยูเอสบี

วัตถุประสงค์

1. ตรวจสอบการรับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ผ่านพอร์ตอนุกรม
2. ศึกษาการทำงานของ Server Application

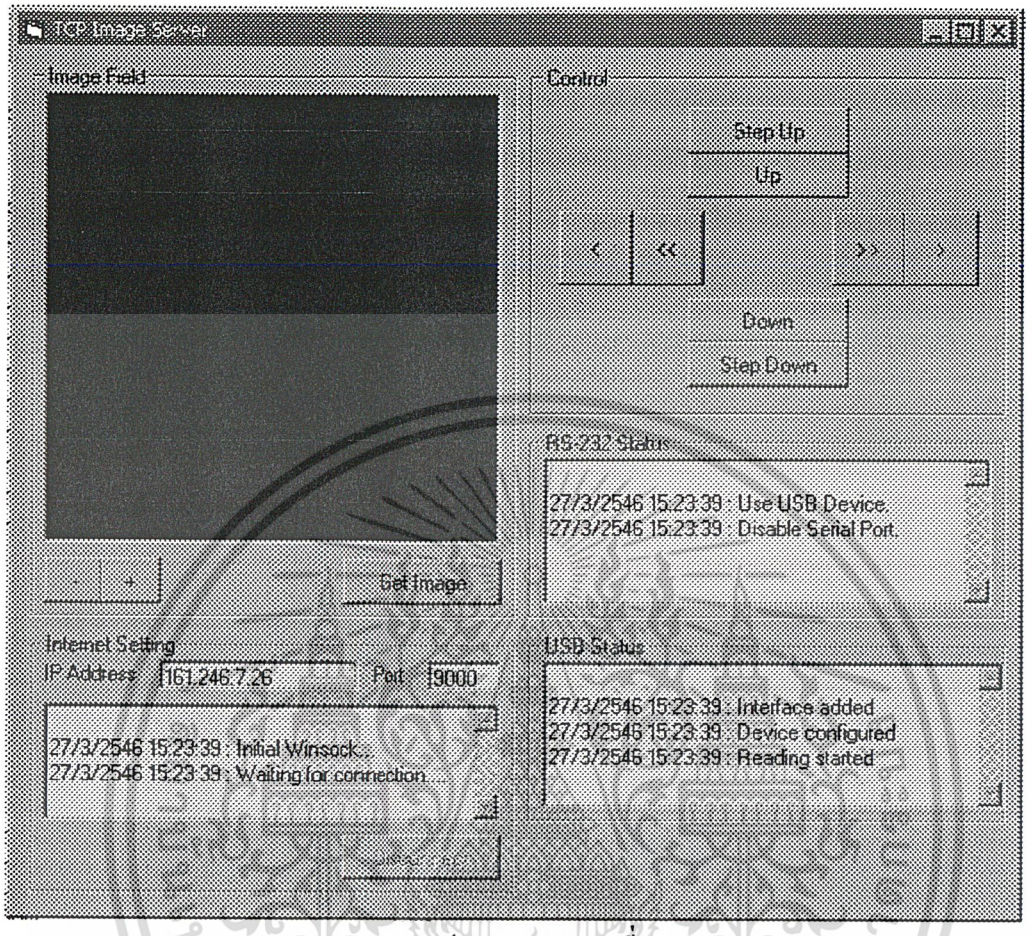
ขั้นตอนการทดลอง

1. เชื่อมต่ออุปกรณ์เข้ากับเครื่องคอมพิวเตอร์ผ่านทางพอร์ตยูเอสบี
2. เรียกใช้โปรแกรม Server Application
3. ทดลองส่งคำสั่งควบคุมมอเตอร์
4. สังเกตและบันทึกผล
5. วิเคราะห์ผลการทดลอง

ผลการทดลอง

เมื่อเรียกใช้โปรแกรม Server Application โปรแกรมจะตรวจพบอุปกรณ์ยูเอสบี และเรียกใช้ Driver ในการควบคุมอุปกรณ์ยูเอสบี และกำหนดค่าต่าง ๆ ให้กับอุปกรณ์เพื่อให้สามารถติดต่อกับอุปกรณ์โดยผ่านทางยูอสบีได้ โดยจะแสดงผลดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปผลการทดลองที่ 3.1

ทดลองควบคุมการทำงานของมอเตอร์โดยการ คลิกเม้าบนคำสั่งที่ต้องการ และสังเกตผลที่เกิดขึ้นกับ มอเตอร์แต่ละตัว

ตารางผลการทดลองที่ 3 ผลการทดลองการควบคุมการทำงานของ Stepper Motor

คำสั่ง	ผลที่เกิดขึ้น	หมายเหตุ
Up	มอเตอร์หมุนขึ้นจนกว่าจะปล่อยเม้า	มอเตอร์ไม่ทำงานเป็นบางครั้ง
Down	มอเตอร์หมุนลงจนกว่าจะปล่อยเม้า	มอเตอร์ไม่ทำงานเป็นบางครั้ง
<<	มอเตอร์หมุนซ้ายจนกว่าจะปล่อยเม้า	-
>>	มอเตอร์หมุนขวาจนกว่าจะปล่อยเม้า	-
Step Up	มอเตอร์หมุนขึ้นเป็นช่วงสั้น ๆ	มอเตอร์ไม่ทำงานเป็นบางครั้ง
Step Down	มอเตอร์หมุนลงเป็นช่วงสั้น ๆ	มอเตอร์ไม่ทำงานเป็นบางครั้ง
<	มอเตอร์หมุนซ้ายเป็นช่วงสั้น ๆ	-
>	มอเตอร์หมุนขวาเป็นช่วงสั้น ๆ	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์ผลการทดลอง

จากตารางผลการทดลองพบว่า ไม่สามารถสั่งการมอเตอร์ชุดควบคุมการหมุนขึ้น/หมุนลงได้ในบางครั้ง เนื่องจากไมโครคอนโทรลเลอร์ไม่ได้รับคำสั่งในการควบคุมการทำงาน เพราะจังหวะการทำงานระหว่าง ไมโครคอนโทรลเลอร์ไม่เท่ากัน เนื่องจากข้อจำกัดด้ายสายข้อมูล ซึ่งถูกทำให้มีจำนวนน้อยที่สุดเพราะต้องให้หมุนได้รอบ (360 องศา) จึงทำให้ไม่สามารถใช้สัญญาณนาฬิกาเดียวกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านกฎรค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 4 การใช้งาน Server Application เพื่อรับภาพจากไมโครคอนโทรลเลอร์ผ่านพอร์ตยูเอสบี

วัตถุประสงค์

1. ตรวจสอบการทำงานในส่วนของการรับส่งข้อมูลภาพของไมโครคอนโทรลเลอร์และโปรแกรม Server Application
2. ศึกษาการทำงานในส่วนของการรับข้อมูลภาพ

ขั้นตอนการทดลอง

1. เชื่อมต่ออุปกรณ์เข้ากับเครื่องคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม
2. เรียกใช้โปรแกรม Server Application
3. ทดลองส่งคำสั่งรับข้อมูลภาพ
4. สังเกตและบันทึกผล
5. วิเคราะห์ผลการทดลอง

ผลการทดลอง

ตารางผลการทดลองที่ 2

คำสั่ง	ผลที่เกิดขึ้น	หมายเหตุ
Get Image	ไม่ได้รับข้อมูลภาพ	-
+	ไม่สามารถตรวจสอบได้	-
-	ไม่สามารถตรวจสอบได้	-

วิเคราะห์ผลการทดลอง

เนื่องจากเกิดความผิดพลาดในการส่งข้อมูลภาพจากไมโครคอนโทรลเลอร์ไปยังคอมพิวเตอร์โดยผ่านพอร์ทยูเอสบี ซึ่งเกิดจากการไมโครคอนโทรลเลอร์ส่วนติดต่อกับอุปกรณ์ยูเอสบี ไม่สามารถรับข้อมูลรูปภาพได้เนื่องจากปัญหาของจังหวะการส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์ที่ใช้รับส่งข้อมูลด้วยพอร์ตยูเอสบีกับไมโครคอนโทรลเลอร์ที่บันทึกข้อมูลภาพ

เนื่องจากไม่สามารถรับข้อมูลภาพได้ จึงไม่สามารถตรวจสอบผลของภาพเมื่อปรับค่า Exposure Timer โดยการส่งคำสั่ง + / - ไปยังอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 5 การใช้งานโปรแกรม Remote Client ในการเชื่อมต่อเข้ากับโปรแกรม Server Application เพื่อควบคุมการทำงานของอุปกรณ์ผ่านเครือข่ายอินเตอร์เน็ต

วัตถุประสงค์

1. ตรวจสอบการทำงานของ Remote Client ในการเชื่อมต่อเข้ากับ Server Application ผ่านเครือข่ายอินเตอร์เน็ต
2. ตรวจสอบการทำงานของ Server Application เมื่อมีการเชื่อมต่อจากเครือข่ายอินเตอร์เน็ต
3. ตรวจสอบการทำงานของ Remote Client ในการควบคุมมอเตอร์
4. ตรวจสอบการทำงานของ Remote Client เมื่อสั่งให้เก็บข้อมูลภาพใหม่
5. ตรวจสอบการทำงานของ Server Application ในการส่งข้อมูลภาพผ่านเครือข่ายอินเตอร์เน็ต
6. ตรวจสอบการทำงานของ Remote Client ในการรับข้อมูลภาพผ่านเครือข่ายอินเตอร์เน็ต
7. ศึกษาการทำงานของโปรแกรม Remote Client

ขั้นตอนการทดลอง

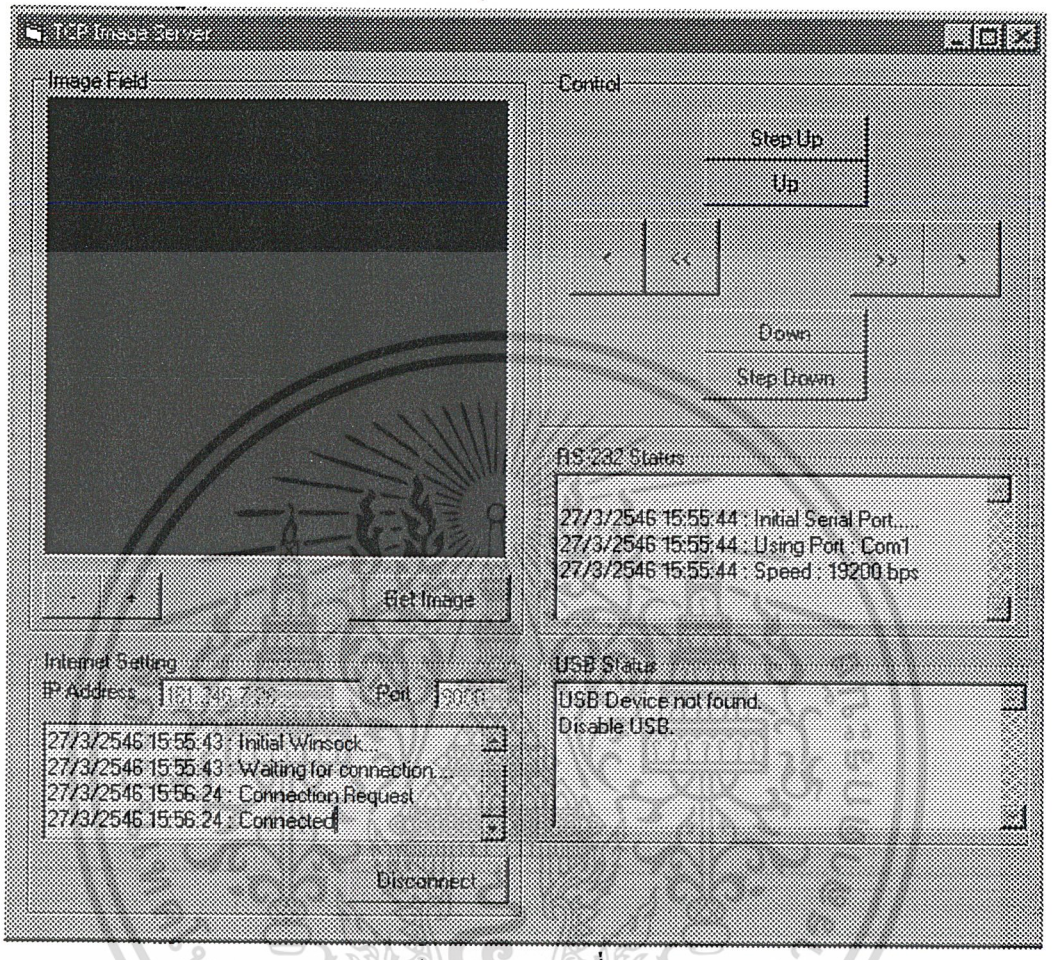
1. เชื่อมต่ออุปกรณ์เข้ากับเครื่องคอมพิวเตอร์ Server
2. เชื่อมต่อคอมพิวเตอร์ Server เข้ากับเครือข่ายอินเตอร์เน็ต
3. เรียกใช้โปรแกรม Server Application ในคอมพิวเตอร์ Server
4. เชื่อมต่อคอมพิวเตอร์ Client เข้ากับเครือข่ายอินเตอร์เน็ต
5. เรียกใช้โปรแกรม Remote Client ในคอมพิวเตอร์ Client
6. ทดลอง สร้าง/ยุติ การเชื่อมต่อระหว่าง Server กับ Client ผ่านโปรแกรม
7. ทดลองสั่งการมอเตอร์ผ่านเครือข่ายอินเตอร์เน็ต
8. ทดลองสั่งอุปกรณ์เพื่อเก็บภาพปัจจุบันผ่านเครือข่ายอินเตอร์เน็ต
9. ทดลองรับ/ส่งไฟล์ข้อมูลภาพผ่านเครือข่ายอินเตอร์เน็ต

ผลการทดลอง

เมื่อเรียกใช้โปรแกรม Server Application ในเครื่อง Server และเรียกใช้ Remote Client ในเครื่อง Client จะสามารถสร้างการเชื่อมต่อได้โดยการระบุหมายเลข IP Address ของเครื่อง Server และเลือกคำสั่ง Connect เพื่อเริ่มการเชื่อมต่อผ่านเครือข่ายอินเตอร์เน็ต

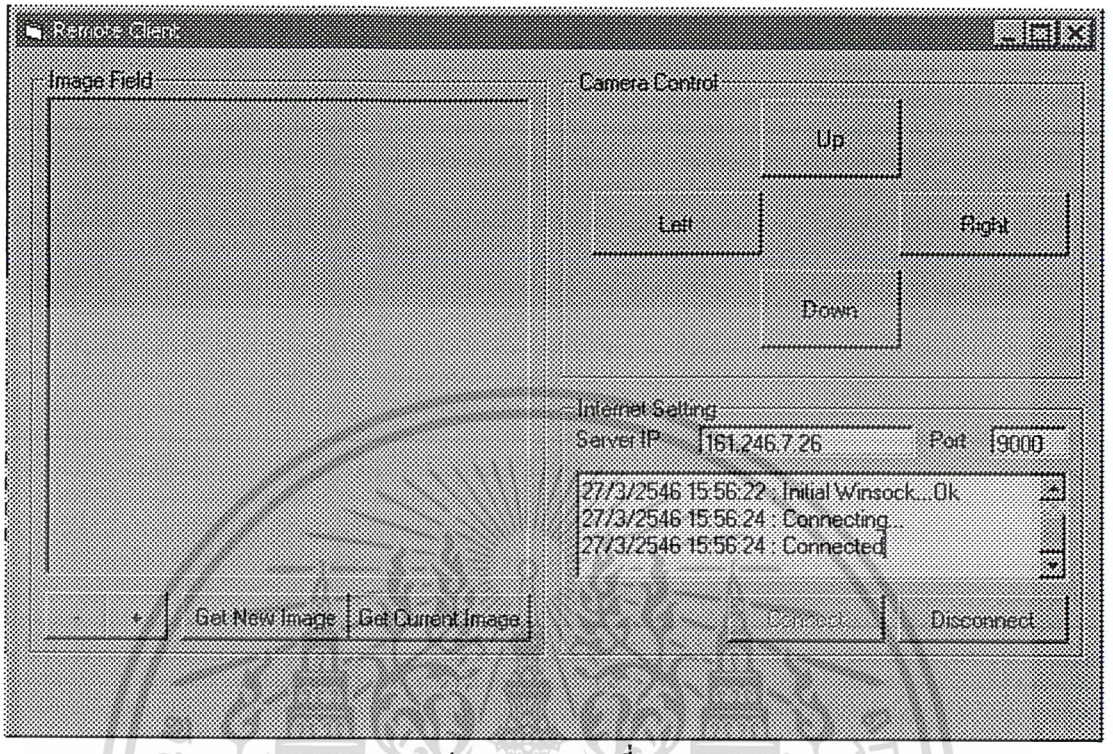
เอกสารนี้เป็นการเผยแพร่ผลงานวิจัยเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งนี้ห้ามมิให้ลบเครื่องหมายลิขสิทธิ์ของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อการเชื่อมต่อสำเร็จจะแสดงผลดังรูป



รูปผลการทดลองที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปผลการทดลองที่ 5.2

จากรูปผลการทดลองที่ 5.1 และรูปผลการทดลองที่ 5.2 เมื่อการเชื่อมต่อสมบูรณ์โปรแกรมทั้งสองจะระบุสถานะ Connected เพื่อแสดงว่าการเชื่อมต่อสมบูรณ์ โดยสามารถยุติการเชื่อต่อได้โดยการเลือกคำสั่ง Disconnect

ตารางผลการทดลองที่ 5.1 ทดลองสั่งการมอเตอร์ผ่านเครือข่ายอินเตอร์เน็ต

คำสั่ง	ผลที่เกิดขึ้นกับ Server Application	หมายเหตุ
Up	มีการส่งคำสั่งเพื่อหมุนมอเตอร์ขึ้นไปยังอุปกรณ์	-
Down	มีการส่งคำสั่งเพื่อหมุนมอเตอร์ลงไปยังอุปกรณ์	-
Left	มีการส่งคำสั่งเพื่อหมุนมอเตอร์ซ้ายไปยังอุปกรณ์	-
Right	มีการส่งคำสั่งเพื่อหมุนมอเตอร์ขวาไปยังอุปกรณ์	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางผลการทดลองที่ 5.2 ทดลองสั่งการมอเตอร์ผ่านเครือข่ายอินเตอร์เน็ต 2

คำสั่ง	ผลที่เกิดขึ้นกับอุปกรณ์	หมายเหตุ
Up	มอเตอร์หมุนขึ้นเป็นช่วงสั้น ๆ	มอเตอร์ไม่ทำงานเป็นบางครั้ง
Down	มอเตอร์หมุนลงเป็นช่วงสั้น ๆ	มอเตอร์ไม่ทำงานเป็นบางครั้ง
Left	มอเตอร์หมุนซ้ายเป็นช่วงสั้น ๆ	-
Right	มอเตอร์หมุนขวาเป็นช่วงสั้น ๆ	-

ตารางผลการทดลองที่ 5.3 ทดลองสั่งไมโครคอนโทรลเลอร์ในการรับส่งข้อมูลภาพ

คำสั่ง	ผลที่เกิดขึ้นกับ Server Application	หมายเหตุ
Get Image	มีการส่งคำสั่งให้ส่งไฟล์ภาพปัจจุบันไปยัง Remote Client	ความเร็วในการส่งขึ้นอยู่กับความเร็วในการเชื่อมต่อเข้ากับเครือขายอินเตอร์เน็ต
Get New Image	มีการส่งคำสั่งเพื่อเก็บภาพใหม่ไปยังอุปกรณ์	
+	มีการส่งคำสั่งเพื่อเพิ่ม Exposure Time ไปยังอุปกรณ์	-
-	มีการส่งคำสั่งเพื่อลด Exposure Time ไปยังอุปกรณ์	-

ตารางผลการทดลองที่ 5.4 ทดลองสั่งไมโครคอนโทรลเลอร์ในการรับส่งข้อมูลภาพ 2

คำสั่ง	ผลที่เกิดขึ้นกับอุปกรณ์	หมายเหตุ
Get Image	-	-
Get New Image	อุปกรณ์ส่งข้อมูลภาพไปยัง Server Application	ไม่มีการส่งข้อมูลภาพในบางครั้ง
	ในกรณีใช้การเชื่อมต่อผ่านพอร์ตยูเอสบี ไม่มีการส่งข้อมูลภาพ	-
+	ข้อมูลภาพครั้งต่อไปมีความสว่างมากขึ้น	ข้อมูลภาพไม่เปลี่ยนแปลงในบางครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	ในกรณีใช้การเชื่อมต่อผ่านพอร์ตยู เอสบี ไม่สามารถตรวจสอบได้	-
-	ข้อมูลภาพครั้งต่อไป มีความสว่างลดลง	ข้อมูลภาพไม่เปลี่ยนแปลง ในบางครั้ง
	ในกรณีใช้การเชื่อมต่อผ่านพอร์ตยู เอสบี ไม่สามารถตรวจสอบได้	-

วิเคราะห์ผลการทดลอง

จากตารางผลการทดลองที่ 5.1 และ 5.2 ซึ่งเป็นการทดลองการควบคุมการทำงานของมอเตอร์โดยทำงานผ่านโปรแกรม Remote Client พบว่าสามารถควบคุมผ่านเครือข่ายอินเตอร์เน็ตได้อย่างสมบูรณ์เช่นเดียวกับการสั่งงานที่ Server Application แต่ยังเกิดปัญหาเช่นเดียวกับการทดลองที่ 1

จากตารางผลการทดลองที่ 5.3 และ 5.4 ซึ่งเป็นการทดลองการควบคุมการทำงานในส่วนจัดการภาพโดยทำงานผ่านโปรแกรม Remote Client พบว่าสามารถทำงานได้อย่างสมบูรณ์เช่นเดียวกับการสั่งงานที่ Server Application โดยความเร็วในการตอบสนองของอุปกรณ์จะแปลผันกับความเร็วในการเชื่อมต่อเข้ากับเครือข่ายอินเตอร์เน็ตของทั้ง Server Application และ Remote Client

เมื่อใช้การเชื่อมต่อผ่านพอร์ตยูเอสบี จะไม่สามารถเรียกข้อมูลภาพจากไมโครคอนโทรลเลอร์ได้ซึ่งเกิดจากปัญหาในการทดลองที่ 4 จึงทำให้ไม่สามารถตรวจสอบการทำงานของคำสั่ง +/- ได้ เช่นเดียวกับการทดลองที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 10
สรุป

โครงงานนี้มีจุดประสงค์เพื่อศึกษาและออกแบบอุปกรณ์ต้นแบบสำหรับสังเกตการณ์จากระยะไกล ประกอบด้วยอุปกรณ์รับภาพชนิด CCD ซึ่งสามารถควบคุมทิศทางของอุปกรณ์รับภาพได้โดยมอเตอร์ 2 ตัวคือ DC Motor และ Stepper Motor โดยสามารถหมุนได้รอบ (360 องศา) ควบคุมการทำงานด้วยไมโครคอนโทรลเลอร์ จำนวน 3 ตัว โดย

1. ควบคุมการทำงานของ CCD และ DC Motor เพื่อควบคุมการก้ม/เงยของกล้อง
2. ควบคุมการหมุนซ้าย/ขวา และเชื่อมต่อกับพอร์ตอนุกรม
3. ควบคุมการเชื่อมต่อแบบ USB

เหตุที่เพิ่มการรับส่งข้อมูลผ่านยูเอสบี ต้องการเพิ่มความเร็วในการรับส่งข้อมูลต่าง ๆ ให้เร็วขึ้น จึงมีความคิดที่จะเปลี่ยนจากการเชื่อมต่อแบบอนุกรมเป็นการเชื่อต่อผ่านยูเอสบีแทน โดยสร้างโปรแกรมสำหรับใช้งานอุปกรณ์นี้ด้วย Visual Basic และใช้ Driver ของชุดพัฒนาไดรว์เวอร์ของบริษัท Thesycon Systemsoftware & Consulting จากประเทศเยอรมัน ที่มีชื่อว่า USBID Development Kit ในการติดต่อกับอุปกรณ์ยูเอสบี แต่เนื่องจากเป็นรุ่นทดลอง จึงสามารถติดต่อกับอุปกรณ์ได้เพียง 20 นาทีเท่านั้น จึ่งต้องเริ่มต้นเครื่องคอมพิวเตอร์ใหม่ทุก ๆ 20 นาที

จากผลการทดลองพบว่า เมื่อใช้การเชื่อมต่อระหว่างอุปกรณ์ กับเครื่องคอมพิวเตอร์เป็นแบบอนุกรม จะสามารถทำงานได้ครบทุกฟังก์ชั่นการทำงานคือ ควบคุมการทำงานของมอเตอร์ และ ควบคุมการทำงานของ CCD แต่จะมีปัญหาเมื่อมีการส่งคำสั่งไปยังไมโครคอนโทรลเลอร์ตัวบน เนื่องจากจังหวะการทำงานของไมโครคอนโทรลเลอร์ไม่เท่ากัน จึงไม่สามารถสั่งงานได้เป็นบางครั้ง

เมื่อเชื่อมต่อระหว่างอุปกรณ์ กับเครื่องคอมพิวเตอร์ด้วยพอร์ตยูเอสบี จะไม่สามารถรับข้อมูลภาพจากไมโครคอนโทรลเลอร์ได้ เนื่องจากจังหวะการทำงานระหว่างไมโครคอนโทรลเลอร์ที่ใช้ติดต่อยูเอสบี กับไมโครคอนโทรลเลอร์ที่เก็บข้อมูลภาพ ไม่ตรงกัน จึงไม่สามารถส่งข้อมูลภาพซึ่งมีจำนวนมากได้

เนื่องจากข้อจำกัดของระยะทางในการรับส่งข้อมูลของการสื่อสารแบบอนุกรมและการสื่อสารแบบยูเอสบีมีความจำกัดมาก จึงพัฒนาให้สามารถใช้งานอุปกรณ์ได้จากระยะไกลได้ โดยรวมความสามารถในการเชื่อมต่อเข้ากับเครือข่ายอินเตอร์เน็ต ซึ่งสามารถเชื่อมโยงเครื่องคอมพิวเตอร์จากทั่วโลกเข้าเป็นเครือข่ายเดียวกัน ซึ่งหมายความว่า อุปกรณ์นี้สามารถใช้งานได้จากทุกที่ที่มีการเชื่อมต่อเข้ากับเครือข่ายอินเตอร์เน็ตได้

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถควบคุมการทำงานของอุปกรณ์ดังกล่าวได้โดยโปรแกรม Server Application และ Remote Client ผ่านพอร์ตอนุกรม หรือ ยูเอสบี โดย

1. Server Application เป็นโปรแกรมสำหรับควบคุมอุปกรณ์ ที่เครื่องคอมพิวเตอร์ที่ต่อเข้ากับอุปกรณ์โดยตรง ซึ่งรองรับการเชื่อมต่อผ่านเครือข่ายอินเตอร์เน็ต
2. Remote Client เป็นโปรแกรมสำหรับควบคุมอุปกรณ์ จากคอมพิวเตอร์จากระยะไกล โดยจะเชื่อมต่อไปยัง Server Application ผ่านเครือข่ายอินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. กฤษดา ใจเย็น, ชัยวัฒน์ ลิ้มพรจิตรวิไล, "เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ PIC16F84, บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด, ฉบับปรับปรุงครั้งที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการทำโครงงานชิ้นนี้สำเร็จลุล่วงไปด้วยดีต้องขอขอบพระคุณ รศ.ดร. มนัส สังวรศิลป์ อาจารย์ที่ปรึกษา ที่ให้คำแนะนำในการทำโครงงาน ตลอดจนการแก้ปัญหาต่างๆที่เกิดขึ้น และขอขอบคุณเพื่อนๆที่เป็นกำลังใจและให้คำแนะนำต่างๆในโครงงานนี้

สุดท้ายนี้ขอขอบคุณผู้ที่ให้ความสนใจในโครงงานนี้ทุกท่าน

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIC16F87X

28/40-Pin 8-Bit CMOS FLASH Microcontrollers

Devices Included in this Data Sheet:

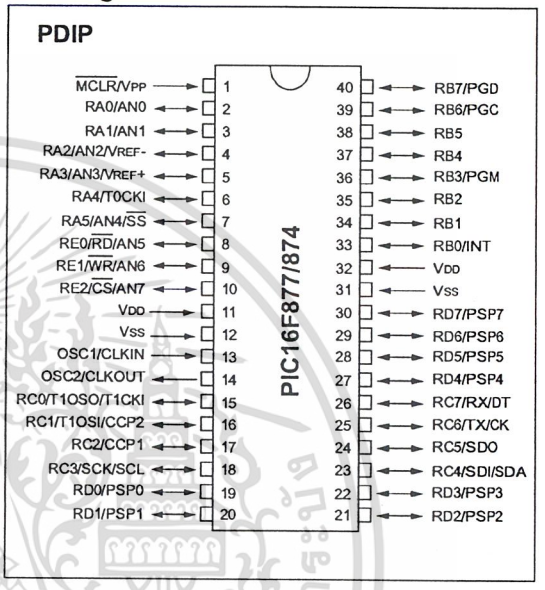
- PIC16F873
- PIC16F874
- PIC16F876
- PIC16F877

Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
  Up to 368 x 8 bytes of Data Memory (RAM)
  Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
  - < 0.6 mA typical @ 3V, 4 MHz
  - 20 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

Pin Diagram

PDIP



Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

© 2001 Microchip Technology Inc.

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

June 1999

National Semiconductor

ADC0820
8-Bit High Speed µP Compatible A/D Converter with Track/Hold Function

General Description

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

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

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

Key Specifications

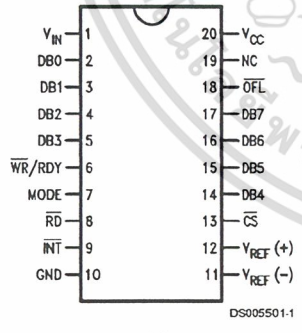
- Resolution — 8 Bits
- Conversion Time — 2.5 µs Max (RD Mode); 1.5 µs Max (WR-RD Mode)
- Low Power — 75 mW Max
- Total Unadjusted Error — ±½ LSB and ± 1 LSB

Features

- Built-in track-and-hold function
- No missing codes
- No external clocking
- Single supply—5 V_{DC}
- Easy interface to all microprocessors, or operates stand-alone
- Latched STRI-STATE output
- Logic inputs and outputs meet both MOS and T^2L voltage level specifications
- Operates ratiometrically or with any reference value equal to or less than V_{CC}
- 0V to 5V analog input voltage range with single 5V supply
- No zero or full-scale adjust required
- Overflow output available for cascading
- 0.3" standard width 20-pin DIP
- 20-pin molded chip carrier package
- 20-pin small outline package
- 20-pin shrink small outline package (SSOP)

Connection and Functional Diagrams

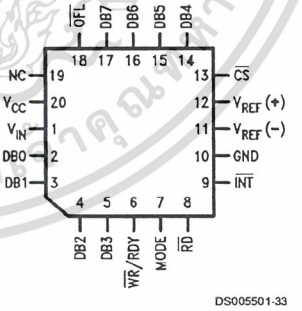
Dual-In-Line, Small Outline and SSOP Packages



DS005501-1

Top View

Molded Chip Carrier Package



DS005501-33

© 2001 National Semiconductor Corporation

เอกสารนี้จัดทำขึ้นสำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connection and Functional Diagrams (Continued)

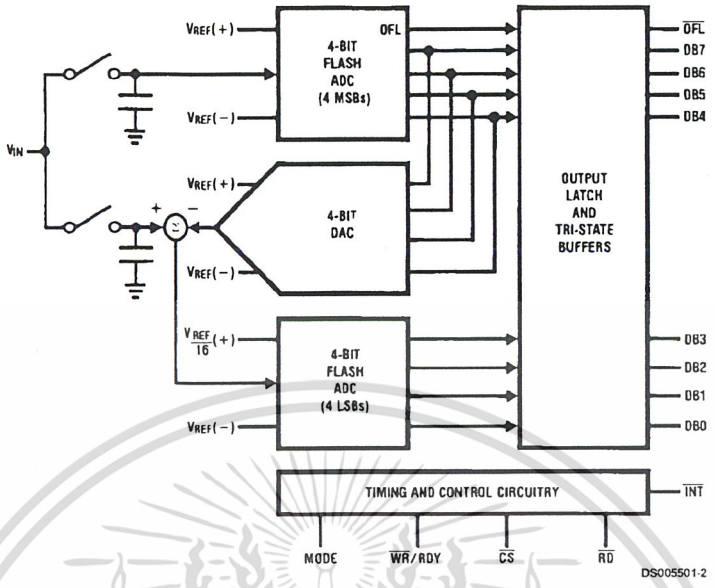


FIGURE 1.

Ordering Information

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Notes 1, 2)

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

Supply Voltage (V_{CC})	10V
Logic Control Inputs	−0.2V to V_{CC} +0.2V
Voltage at Other Inputs and Output	−0.2V to V_{CC} +0.2V
Storage Temperature Range	−65°C to +150°C
Package Dissipation at T_A = 25°C	875 mW
Input Current at Any Pin (Note 5)	1 mA
Package Input Current (Note 5)	4 mA
ESD Susceptability (Note 9)	1200V
Lead Temp. (Soldering, 10 sec.)	
Dual-In-Line Package (plastic)	260°C
Dual-In-Line Package (ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 sec.)	215°C
Infrared (15 sec.)	220°C

Operating Ratings (Notes 1, 2)

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

Converter Characteristics

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

Parameter	Conditions	ADC0820CCJ			ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM			Limit Units
		Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	
Resolution			**8**			8	**8**	Bits
Total Unadjusted Error (Note 3)	ADC0820BCN, BCWM					±½	**±½**	LSB
	ADC0820CCJ		**±1**					LSB
	ADC0820CCN, CCWM, CIWM,					±1	**±1**	LSB
	ADC0820CCMSA					±1	**±1**	LSB
Minimum Reference Resistance		2.3	**1.00**		2.3	1.2		kΩ
Maximum Reference Resistance		2.3	**6**		2.3	5.3	**6**	kΩ
Maximum $V_{REF}(+)$ Input Voltage			**V_{CC}**			V_{CC}	**V_{CC}**	V
Minimum $V_{REF}(-)$ Input Voltage			**GND**			GND	**GND**	V
Minimum $V_{REF}(+)$ Input Voltage			**$V_{REF}(-)$**			$V_{REF}(-)$	**$V_{REF}(-)$**	V
Maximum $V_{REF}(-)$ Input Voltage			**$V_{REF}(+)$**			$V_{REF}(+)$	**$V_{REF}(+)$**	V
Maximum V_{IN} Input Voltage			**V_{CC}+0.1**			V_{CC}+0.1	**V_{CC}+0.1**	V
Minimum V_{IN} Input Voltage			**GND−0.1**			GND−0.1	**GND−0.1**	V
Maximum Analog Input Leakage Current	$\overline{CS}$ =V_{CC}							
	V_{IN}=V_{CC}		**3**			0.3	**3**	µA
	V_{IN}=GND		**−3**			−0.3	**−3**	µA
Power Supply Sensitivity	V_{CC}=5V±5%	±1/16	**±¼**		±1/16	±¼	**±¼**	LSB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC Electrical Characteristics

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

Parameter	Conditions		ADC0820CCJ			ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM			Limit Units
			Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	
$V_{IN(1)}$, Logical "1" Input Voltage	V_{CC}=5.25V	$\overline{CS}$, $\overline{WR}$, $\overline{RD}$		**2.0**			2.0	**2.0**	V
		Mode		**3.5**			3.5	**3.5**	V
$V_{IN(0)}$, Logical "0" Input Voltage	V_{CC}=4.75V	$\overline{CS}$, $\overline{WR}$, $\overline{RD}$		**0.8**			0.8	**0.8**	V
		Mode		**1.5**			1.5	**1.5**	V
$I_{IN(1)}$, Logical "1" Input Current	$V_{IN(1)}$=5V; $\overline{CS}$, $\overline{RD}$		0.005	1		0.005		**1**	µA
	$V_{IN(1)}$=5V; $\overline{WR}$		0.1	3		0.1	0.3	**3**	µA
	$V_{IN(1)}$=5V; Mode		50	**200**		50	170	**200**	µA
$I_{IN(0)}$, Logical "0" Input Current	$V_{IN(0)}$=0V; $\overline{CS}$, $\overline{RD}$, $\overline{WR}$, Mode		−0.005	**−1**		−0.005		**−1**	µA
$V_{OUT(1)}$, Logical "1" Output Voltage	V_{CC}=4.75V, I_{OUT}=−360 µA; DB0–DB7, $\overline{OFL}$, $\overline{INT}$			**2.4**			2.8	**2.4**	V
	V_{CC}=4.75V, I_{OUT}=−10 µA; DB0–DB7, $\overline{OFL}$, $\overline{INT}$			**4.5**			4.6	**4.5**	V
$V_{OUT(0)}$, Logical "0" Output Voltage	V_{CC}=4.75V, I_{OUT}=1.6 mA; DB0–DB7, $\overline{OFL}$, $\overline{INT}$, RDY			**0.4**			0.34	**0.4**	V
I_{OUT}, TRI-STATE Output Current	V_{OUT}=5V; DB0–DB7, RDY		0.1	3		0.1	0.3	**3**	µA
	V_{OUT}=0V; DB0–DB7, RDY		−0.1	**−3**		−0.1	−0.3	**−3**	µA
I_{SOURCE}, Output Source Current	V_{OUT}=0V; DB0–DB7, $\overline{OFL}$		−12	**−6**		−12	−7.2	**−6**	mA
	$\overline{INT}$		−9	**−4.0**		−9	−5.3	**−4.0**	mA
I_{SINK}, Output Sink Current	V_{OUT}=5V; DB0–DB7, $\overline{OFL}$, $\overline{INT}$, RDY		14	7		14	8.4	**7**	mA
I_{CC}, Supply Current	$\overline{CS}$ =$\overline{WR}$ =$\overline{RD}$ =0		7.5	**15**		7.5	13	**15**	mA

AC Electrical Characteristics

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Electrical Characteristics (Continued)

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

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

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

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

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

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

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

Note 6: Typicals are at 25°C and represent most likely parametric norm.

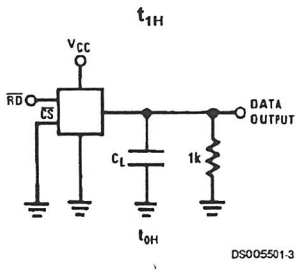
Note 7: Tested limits are guaranteed to National's AOQL (Average Outgoing Quality Level).

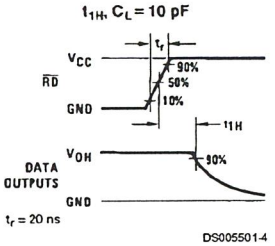
Note 8: Design limits are guaranteed but not 100% tested. These limits are not used to calculate outgoing quality levels.

Note 9: Human body model, 100 pF discharaged through a 1.5 kΩ resistor.

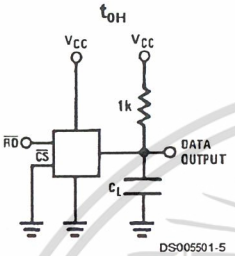
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

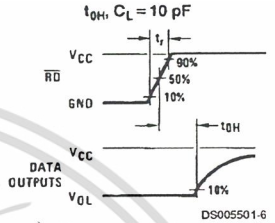
TRI-STATE Test Circuits and Waveforms





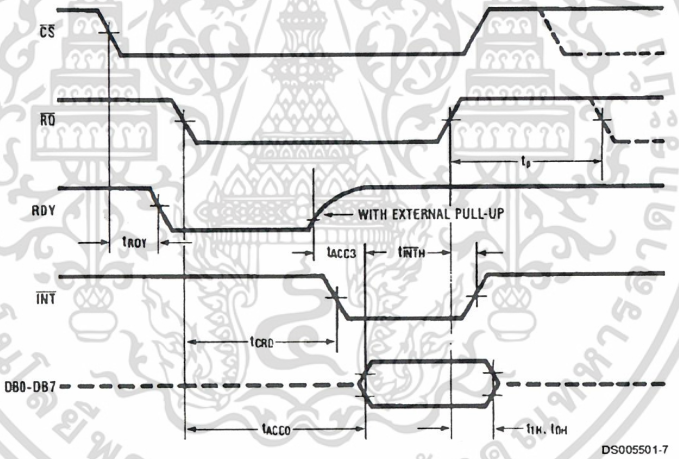
t_r=20 ns





t_r=20 ns

Timing Diagrams



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

FIGURE 2. RD Mode (Pin 7 is Low)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timing Diagrams (Continued)

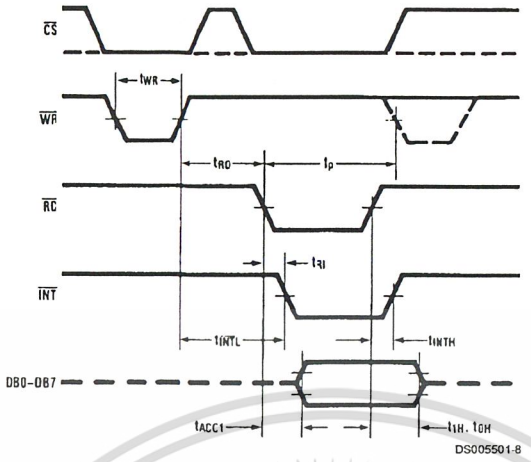


FIGURE 3. WR-RD Mode (Pin 7 is High and $t_{RD} < t_I$)

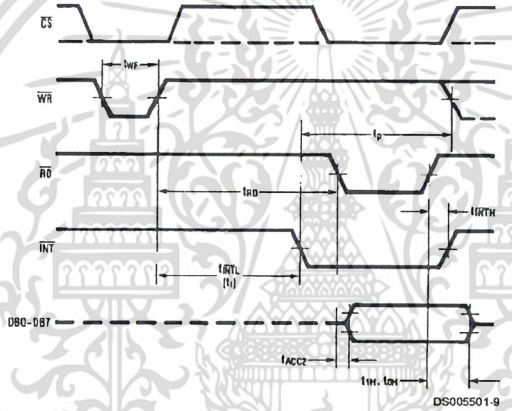


FIGURE 4. WR-RD Mode (Pin 7 is High and $t_{RD} > t_I$)

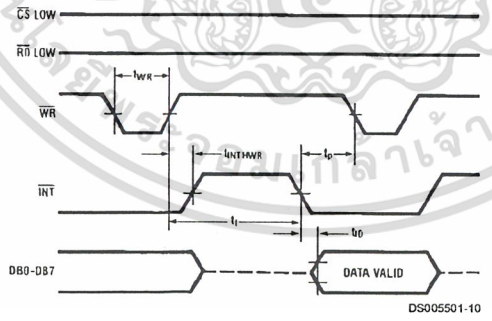
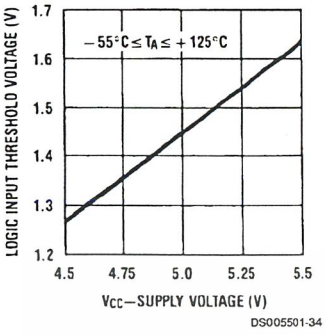


FIGURE 5. WR-RD Mode (Pin 7 is High)
Stand-Alone Operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

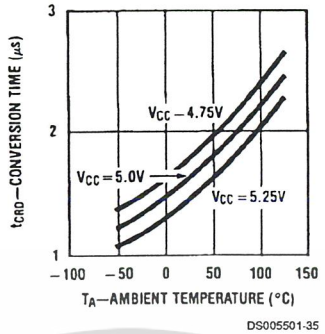
Typical Performance Characteristics

Logic Input Threshold Voltage vs Supply Voltage



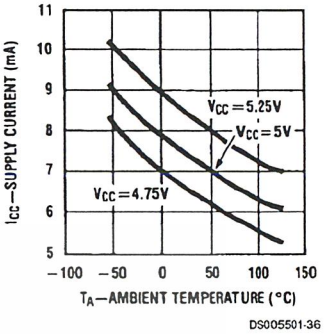
DS005501-34

Conversion Time (RD Mode) vs Temperature



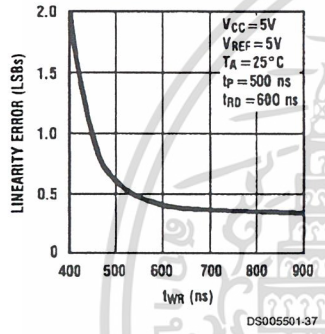
DS005501-35

Power Supply Current vs Temperature (not including reference ladder)



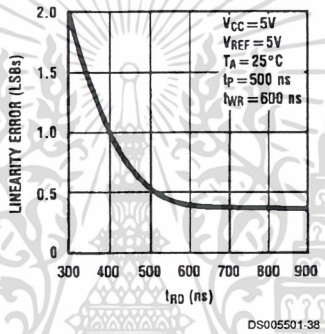
DS005501-36

Accuracy vs t_{WR}



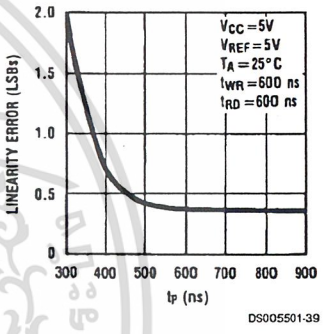
DS005501-37

Accuracy vs t_{RD}



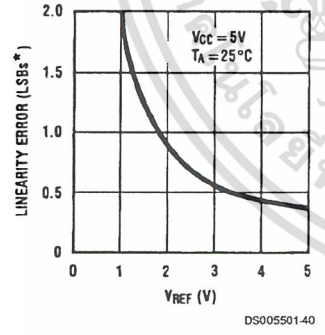
DS005501-38

Accuracy vs t_p



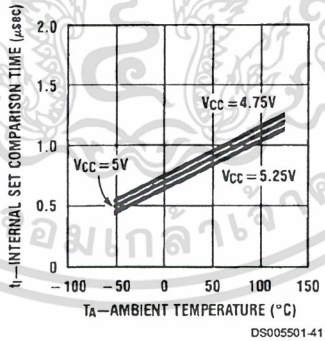
DS005501-39

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



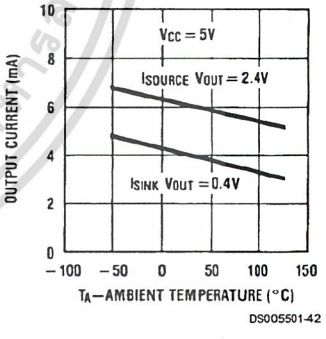
DS005501-40

t_I, Internal Time Delay vs Temperature



DS005501-41

Output Current vs Temperature



DS005501-42

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Description of Pin Functions

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

1.0 Functional Description

1.1 GENERAL OPERATION

The ADC0820 uses two 4-bit flash A/D converters to make an 8-bit measurement (*Figure 1*). Each flash ADC is made up of 15 comparators which compare the unknown input to a reference ladder to get a 4-bit result. To take a full 8-bit reading, one flash conversion is done to provide the 4 most significant data bits (via the MS flash ADC). Driven by the 4 MSBs, an internal DAC recreates an analog approximation of the input voltage. This analog signal is then subtracted from the input, and the difference voltage is converted by a second 4-bit flash ADC (the LS ADC), providing the 4 least significant bits of the output data word.

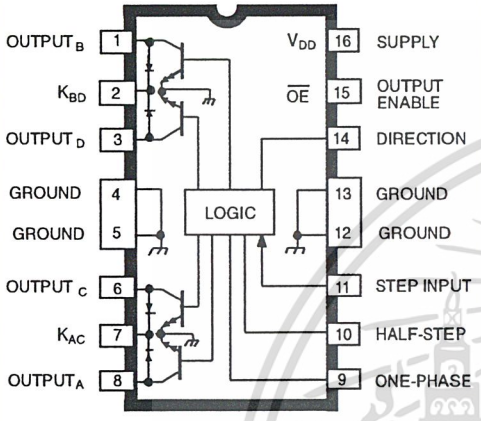
The internal DAC is actually a subsection of the MS flash converter. This is accomplished by using the same resistor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5804

Data Sheet
26184.12C*

BiMOS II UNIPOLAR STEPPER-MOTOR TRANSLATOR/DRIVER



Dwg. W-194

Note that the UCN5804B (dual in-line package) and UCN5804LB (small outline IC package) are electrically identical and share a common terminal number assignment.

ABSOLUTE MAXIMUM RATINGS

Output Voltage, V_{CE} ..... **50 V**
Output Sustaining Voltage,
$V_{CE\ (sus)}$ ..... **35 V**
Output Sink Current, I_{OUT} ..... **1.5 A**
Logic Supply Voltage, V_{DD} ..... **7.0 V**
Input Voltage, V_{IN} ..... **7.0 V**
Package Power Dissipation,
P_D ..... **See Graph**
Operating Temperature Range,
T_A ..... **-20°C to +85°C**
Storage Temperature Range,
T_S ..... **-55°C to +150°C**

Combining low-power CMOS logic with high-current and high-voltage bipolar outputs, the UCN5804B and UCN5804LB BiMOS II translator/drivers provide complete control and drive for a four-phase unipolar stepper-motor with continuous output current ratings to 1.25 A per phase (1.5 A startup) and 35 V.

The CMOS logic section provides the sequencing logic, DIRECTION and OUTPUT ENABLE control, and a power-on reset function. Three stepper-motor drive formats, wave-drive (one-phase), two-phase, and half-step are externally selectable. The inputs are compatible with standard CMOS, PMOS, and NMOS circuits. TTL or LSTTL may require the use of appropriate pull-up resistors to ensure a proper input-logic high.

The wave-drive format consists of energizing one motor phase at a time in an A-B-C-D (or D-C-B-A) sequence. This excitation mode consumes the least power and assures positional accuracy regardless of any winding inbalance in the motor. Two-phase drive energizes two adjacent phases in each detent position (AB-BC-CD-DA). This sequence mode offers an improved torque-speed product, greater detent torque, and is less susceptible to motor resonance. Half-step excitation alternates between the one-phase and two-phase modes (A-AB-B-BC-C-CD-D-DA), providing an eight-step sequence.

The bipolar outputs are capable of sinking up to 1.5 A and withstanding 50 V in the off state (sustaining voltages up to 35 V). Ground-clamp and flyback diodes provide protection against inductive transients. Thermal protection circuitry disables the outputs when the chip temperature is excessive.

Both devices are rated for operation over the temperature range of -20°C to +85°C. The UCN5804B is supplied in a 16-pin dual in-line plastic batwing package with a copper lead frame and heat-sinkable tabs for improved power dissipation capabilities; the UCN5804LB is supplied in a 16-lead plastic SOIC batwing package with a copper lead frame and heat-sinkable tabs.

FEATURES

- 1.5 A Maximum Output Current
- 35 V Output Sustaining Voltage
- Wave-Drive, Two-Phase, and Half-Step Drive Formats
- Internal Clamp Diodes
- Output Enable and Direction Control
- Power-On Reset
- Internal Thermal Shutdown Circuitry

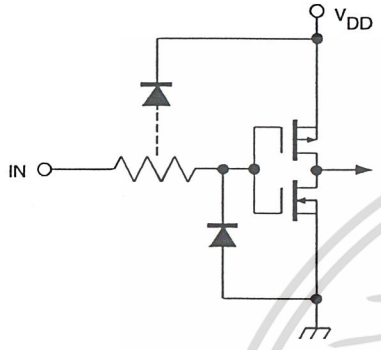
Always order by complete part number, e.g., **UCN5804B**.

Allegro MicroSystems, Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

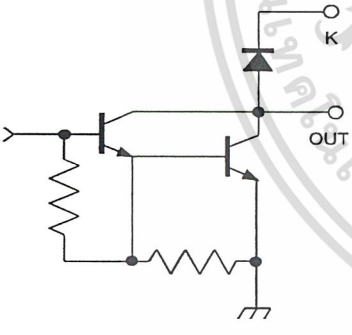
5804
BiMOS II UNIPOLAR STEPPER-MOTOR TRANSLATOR/DRIVER

TYPICAL INPUT CIRCUIT

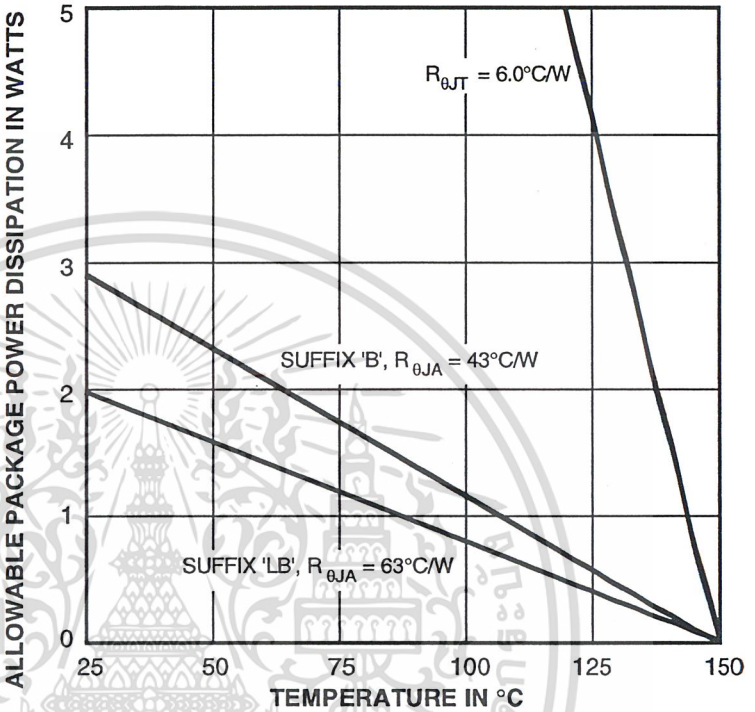


Dwg. EP-010-5

TYPICAL OUTPUT DRIVER



Dwg. EP-021-4



Dwg. GP-049-2A

TRUTH TABLE

Drive Format	Pin 9	Pin 10
Two-Phase	L	L
One-Phase	H	L
Half-Step	L	H
Step-Inhibit	H	H

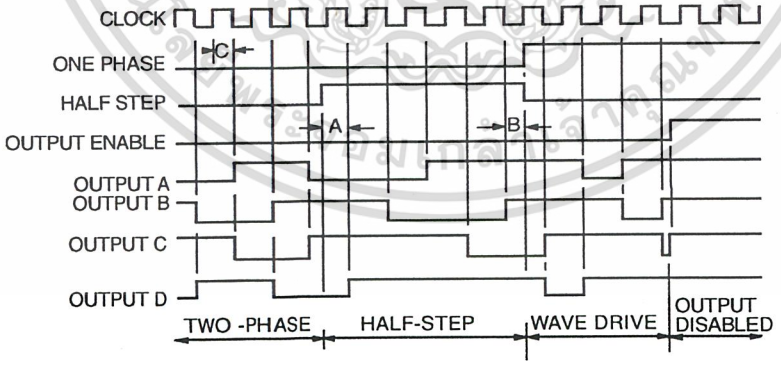
Allegro MicroSystems, Inc.
115 Northeast Cutoff, Box 15036
Worcester, Massachusetts 01615-0036 (508) 853-5000

Copyright © 1987, 2000 Allegro MicroSystems, Inc.

ELECTRICAL CHARACTERISTICS at $T_A = 25°C$, $T_J \leq 150°C$, $V_{DD} = 4.5$ V to 5.5 V (unless otherwise noted).

Characteristic	Symbol	Test Conditions	Limits			Units
			Min.	Typ.	Max.	
Output Leakage Current	I_{CEX}	$V_{OUT} = 50$ V	—	10	50	µA
Output Sustaining Voltage	$V_{CE(sus)}$	$I_{OUT} = 1.25$ A, L = 3 mH	35	—	—	V
Output Saturation Voltage	$V_{CE(SAT)}$	$I_{OUT} = 700$ mA	—	1.0	1.2	V
		$I_{OUT} = 1$ A	—	1.1	1.4	V
		$I_{OUT} = 1.25$ A	—	1.2	1.5	V
Clamp Diode Leakage Current	I_R	$V_R = 50$ V	—	10	50	µA
Clamp Diode Forward Voltage	V_F	$I_F = 1.25$ A	—	1.5	3.0	V
Input Current	$I_{IN(1)}$	$V_{IN} = V_{DD}$	—	0.5	5.0	µA
	$I_{IN(0)}$	$V_{IN} = 0.8$ V	—	-0.5	-5.0	µA
Input Voltage	$V_{IN(1)}$	$V_{DD} = 5$ V	3.5	—	5.3	V
	$V_{IN(0)}$		-0.3	—	0.8	V
Supply Current	I_{DD}	2 Outputs ON	—	20	30	mA
Turn-Off Delay	t_{ON}	50% Step Inputs to 50% Output	—	—	10	µs
Turn-On Delay	t_{OFF}	50% Step Inputs to 50% Output	—	—	10	µs
Thermal Shutdown Temperature	T_J		—	165	—	°C

TIMING CONDITIONS



Dwg. W-110A

A. Minimum Data Set Up Time ..... **100 ns**
B. Minimum Data Hold Time ..... **100 ns**
C. Minimum Step Input Pulse Width ..... **3.0 µs**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

128Kx8 bit Low Power CMOS Static RAM

FEATURES

- Process Technology: TFT
- Organization: 128Kx8
- Power Supply Voltage: 4.5~5.5V
- Low Data Retention Voltage: 2V(Min)
- Three state output and TTL Compatible
- Package Type: 32-DIP-600, 32-SOP-525, 32-TSOP1-0820F/R

GENERAL DESCRIPTION

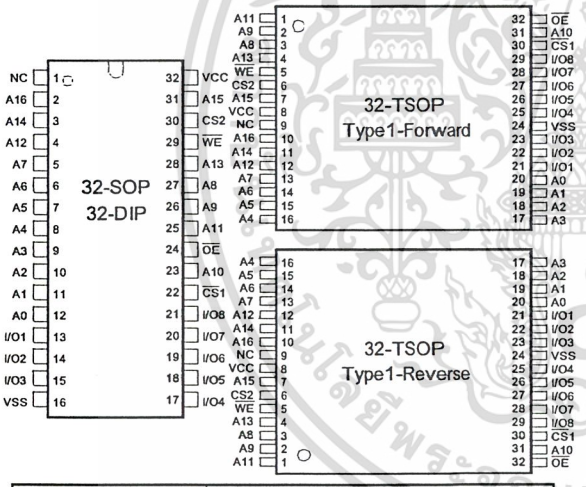
The K6T1008C2E families are fabricated by SAMSUNG's advanced CMOS process technology. The families support various operating temperature ranges and have various package types for user flexibility of system design. The families also support low data retention voltage for battery back-up operation with low data retention current.

PRODUCT FAMILY

Product Family	Operating Temperature	Vcc Range	Speed	Power Dissipation		PKG Type
				Standby (I_{SB1}, Max)	Operating (I_{CC2}, Max)	
K6T1008C2E-L	Commercial(0~70°C)	4.5~5.5V	55[1]/70ns	50μA	50mA	32-DIP-600, 32-SOP-525 32-TSOP1-0820F/R
K6T1008C2E-B				10μA		
K6T1008C2E-P	Industrial(-40~85°C)			50μA		32-SOP -525 32-TSOP1-0820F/R
K6T1008C2E-F				15μA		

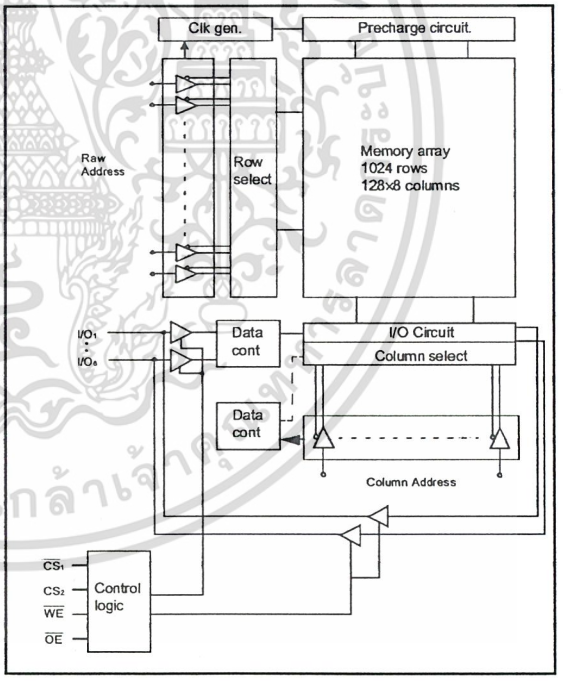
1. The parameters are tested with 50pF test load

PIN DESCRIPTION



Name	Function
$\overline{CS_1}$, CS_2	Chip Select Input
$\overline{OE}$	Output Enable Input
$\overline{WE}$	Write Enable Input
I/O_1~I/O_8	Data Inputs/Outputs
A_0~A_{16}	Address Inputs
Vcc	Power
Vss	Ground
NC	No Connection

FUNCTIONAL BLOCK DIAGRAM



SAMSUNG ELECTRONICS CO., LTD. reserves the right to change products and specifications without notice.

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRODUCT LIST

Commercial Temperature Products(0~70°C)		Industrial Temperature Products(-40~85°C)	
Part Name	Function	Part Name	Function
K6T1008C2E-DL55	32-DIP, 55ns, Low Power	K6T1008C2E-GP55	32-SOP, 55ns, Low Power
K6T1008C2E-DL70	32-DIP, 70ns, Low Power	K6T1008C2E-GP70	32-SOP, 70ns, Low Power
K6T1008C2E-DB55	32-DIP, 55ns, Low Low Power	K6T1008C2E-GF55	32-SOP, 55ns, Low Low Power
K6T1008C2E-DB70	32-DIP, 70ns, Low Low Power	K6T1008C2E-GF70	32-SOP, 70ns, Low Low Power
K6T1008C2E-GL55	32-SOP, 55ns, Low Power	K6T1008C2E-TF55	32-TSOP-F, 55ns, Low Low Power
K6T1008C2E-GL70	32-SOP, 70ns, Low Power	K6T1008C2E-TF70	32-TSOP-F, 70ns, Low Low Power
K6T1008C2E-GB55	32-SOP, 55ns, Low Low Power	K6T1008C2E-RF55	32-TSOP-R, 55ns, Low Low Power
K6T1008C2E-GB70	32-SOP, 70ns, Low Low Power	K6T1008C2E-RF70	32-TSOP-R, 70ns, Low Low Power
K6T1008C2E-TB55	32-TSOP-F, 55ns, Low Low Power		
K6T1008C2E-TB70	32-TSOP-F, 70ns, Low Low Power		
K6T1008C2E-RB55	32-TSOP-R, 55ns, Low Low Power		
K6T1008C2E-RB70	32-TSOP-R, 70ns, Low Low Power		

FUNCTIONAL DESCRIPTION

$\overline{CS_1}$	CS_2	$\overline{OE}$	$\overline{WE}$	I/O	Mode	Power
H	X[1]	X[1]	X[1]	High-Z	Deselected	Standby
X[1]	L	X[1]	X[1]	High-Z	Deselected	Standby
L	H	H	H	High-Z	Output Disabled	Active
L	H	L	H	Dout	Read	Active
L	H	X[1]	L	Din	Write	Active

1. X means don't care (Must be in high or low states)

ABSOLUTE MAXIMUM RATINGS[1]

Item	Symbol	Ratings	Unit	Remark
Voltage on any pin relative to Vss	V_{IN},V_{OUT}	-0.5 to 7.0	V	-
Voltage on Vcc supply relative to Vss	V_{CC}	-0.5 to 7.0	V	-
Power Dissipation	P_D	1.0	W	-
Storage temperature	T_{STG}	-65 to 150	°C	-
Operating Temperature	T_A	0 to 70	°C	K6T1008C2E-L/-B
		-40 to 85	°C	K6T1008C2E-P/-F

1. Stresses greater than those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. Functional operation should be restricted to recommended operating condition. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RECOMMENDED DC OPERATING CONDITIONS[1)]

Item	Symbol	Product	Min	Typ	Max	Unit
Supply voltage	Vcc	K6T1008C2E Family	4.5	5.0	5.5	V
Ground	Vss	All Family	0	0	0	V
Input high voltage	V_{IH}	K6T1008C2E Family	2.2	-	Vcc+0.5[2)]	V
Input low voltage	V_{IL}	K6T1008C2E Family	-0.5[3)]	-	0.8	V

Note:
1. Commercial Product: T_A=0 to 70°C
   Industrial Product: T_A=-40 to 85°C, otherwise specified.
2. Overshoot: Vcc+3.0V in case of pulse width≤30ns.
3. Undershoot: -3.0V in case of pulse width≤30ns.
4. Overshoot and undershoot are sampled, not 100% tested.

CAPACITANCE[1)] (f=1MHz, TA=25°C)

Item	Symbol	Test Condition	Min	Max	Unit
Input capacitance	C_{IN}	V_{IN}=0V	-	6	pF
Input/Output capacitance	C_{IO}	V_{IO}=0V	-	8	pF

1. Capacitance is sampled, not 100% tested

DC AND OPERATING CHARACTERISTICS

Item	Symbol	Test Conditions	Min	Typ	Max	Unit
Input leakage current	I_{LI}	V_{IN}=Vss to Vcc	-1	-	1	μA
Output leakage current	I_{LO}	$\overline{CS_1}$=V_{IH} or CS_2=V_{IL} or $\overline{OE}$=V_{IH} or $\overline{WE}$=V_{IL}, V_{IO}=Vss to Vcc	-1	-	1	μA
Operating power supply current	I_{CC}	I_{IO}=0mA, $\overline{CS_1}$=V_{IL}, CS_2=V_{IH}, V_{IN}=V_{IH} or V_{IL}, Read	-	-	10	mA
Average operating current	I_{CC1}	Cycle time=1μs, 100%duty, I_{IO}=0mA, $\overline{CS_1}$≤0.2V, CS_2≥Vcc-0.2V, V_{IN}≤0.2V or V_{IN}≥Vcc-0.2V	-	-	7	mA
	I_{CC2}	Cycle time=Min, 100% duty, I_{IO}=0mA, $\overline{CS_1}$=V_{IL}, CS_2=V_{IH}, V_{IN}=V_{IH} or V_{IL}	-	-	50	mA
Output low voltage	V_{OL}	I_{OL}=2.1mA	-	-	0.4	V
Output high voltage	V_{OH}	I_{OH}=-1.0mA	2.4	-	-	V
Standby Current(TTL)	I_{SB}	$\overline{CS_1}$=V_{IH}, CS2=V_{IL}, Other inputs=V_{IH} or V_{IL}	-	-	3	mA
Standby Current(CMOS)	I_{SB1}	$\overline{CS_1}$≥Vcc-0.2V, CS_2≥Vcc-0.2V or CS_2≤0.2V, Other inputs=0~Vcc	-	-	50[1)]	μA

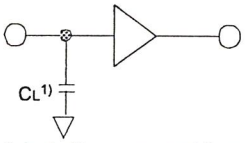
1. 50μA for Low power product, in case of Low Low power products are comercial=10μA, industrial=15μA.

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC OPERATING CONDITIONS

TEST CONDITIONS(Test Load and Input/Output Reference)

Input pulse level: 0.8 to 2.4V
Input rising and falling time: 5ns
Input and output reference voltage:1.5V
Output load(see right): C_L=100pF+1TTL
C_L=50pF+1TTL



1. Including scope and jig capacitance

AC CHARACTERISTICS (V_{CC}=4.5~5.5V, Commercial Product: T_A=0 to 70°C, Industrial Product: T_A=-40 to 85°C)

Parameter List		Symbol	Speed Bins				Units
			55ns		70ns		
			Min	Max	Min	Max	
Read	Read Cycle Time	t_{RC}	55	-	70	-	ns
	Address Access Time	t_{AA}	-	55	-	70	ns
	Chip Select to Output	t_{CO}	-	55	-	70	ns
	Output Enable to Valid Output	t_{OE}	-	25	-	35	ns
	Chip Select to Low-Z Output	t_{LZ}	10	-	10	-	ns
	Output Enable to Low-Z Output	t_{OLZ}	5	-	5	-	ns
	Chip Disable to High-Z Output	t_{HZ}	0	20	0	25	ns
	Output Disable to High-Z Output	t_{OHZ}	0	20	0	25	ns
	Output Hold from Address Change	t_{OH}	10	-	10	-	ns
Write	Write Cycle Time	t_{WC}	55	-	70	-	ns
	Chip Select to End of Write	t_{CW}	45	-	60	-	ns
	Address Set-up Time	t_{AS}	0	-	0	-	ns
	Address Valid to End of Write	t_{AW}	45	-	60	-	ns
	Write Pulse Width	t_{WP}	40	-	50	-	ns
	Write Recovery Time	t_{WR}	0	-	0	-	ns
	Write to Output High-Z	t_{WHZ}	0	20	0	25	ns
	Data to Write Time Overlap	t_{DW}	20	-	25	-	ns
	Data Hold from Write Time	t_{DH}	0	-	0	-	ns
	End Write to Output Low-Z	t_{OW}	5	-	5	-	ns

DATA RETENTION CHARACTERISTICS

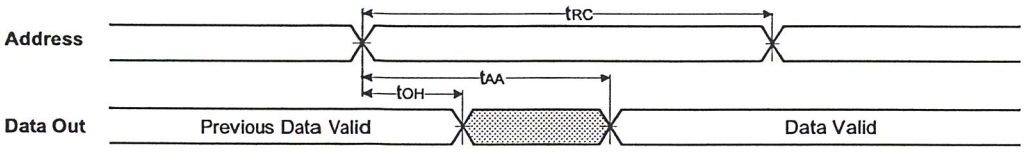
Item	Symbol	Test Condition		Min	Typ	Max	Unit
Vcc for data retention	V_{DR}	$\overline{CS_1} \geq V_{CC}-0.2V$ 1)		2.0	-	5.5	V
Data retention current	I_{DR}	V_{CC}=3.0V, $\overline{CS_1} \geq V_{CC}-0.2V$ 1)	K6T1008C2E-L	-	-	20	µA
			K6T1008C2E-B	-	-	10	
			K6T1008C2E-P	-	-	25	
			K6T1008C2F-F	-	-	10	
Data retention set-up time	t_{SDR}	See data retention waveform		0	-	-	ms
Recovery time	t_{RDR}			5	-	-	

1. $\overline{CS_1} \geq V_{CC}-0.2V$, $CS_2 \geq V_{CC}-0.2V$($\overline{CS_1}$ controlled) or $CS_2 \leq 0.2V$(CS_2 controlled)

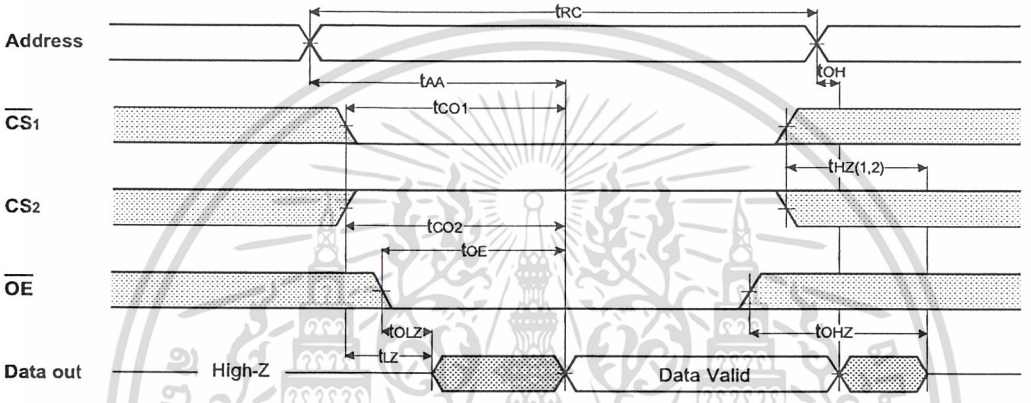
เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING DIAGRAMS

TIMING WAVEFORM OF READ CYCLE(1) (Address Controlled, $\overline{CS1}$=$\overline{OE}$=V_{IL}, CS2=$\overline{WE}$=V_{IH})



TIMING WAVEFORM OF READ CYCLE(2) ($\overline{WE}$=V_{IH})

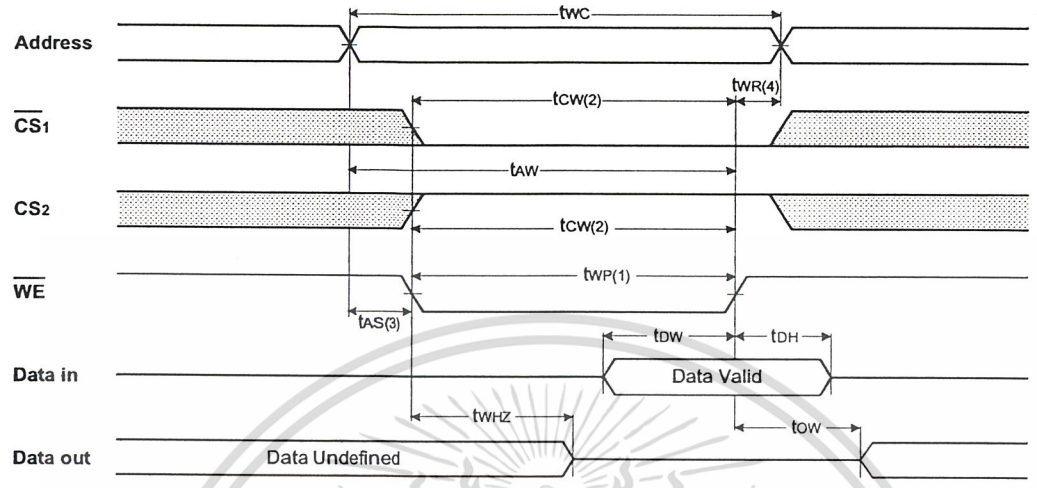


NOTES (READ CYCLE)

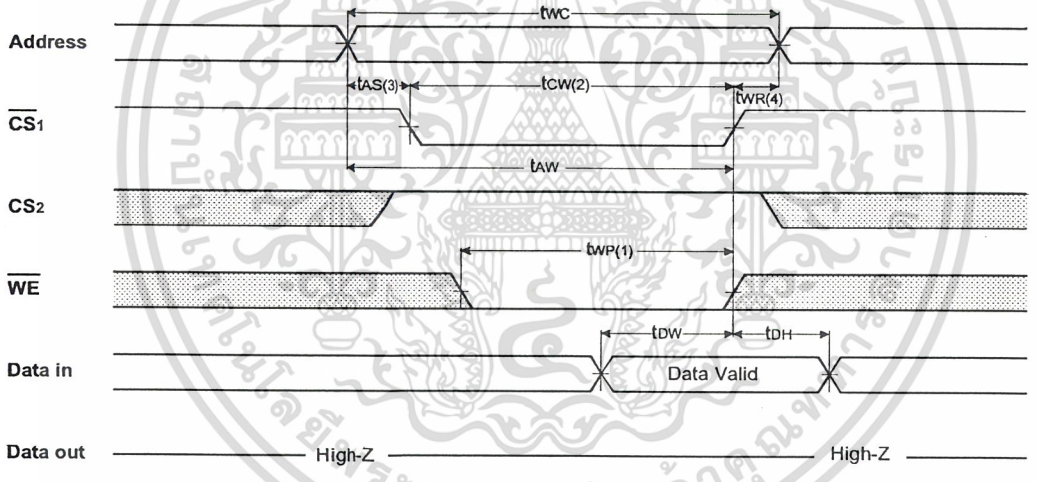
1. t_{HZ} and t_{OHZ} are defined as the time at which the outputs achieve the open circuit conditions and are not referenced to output voltage levels.
2. At any given temperature and voltage condition, t_{HZ}(Max.) is less than t_{LZ}(Min.) both for a given device and from device to device interconnection.

เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING WAVEFORM OF WRITE CYCLE(1) ($\overline{WE}$ Controlled)

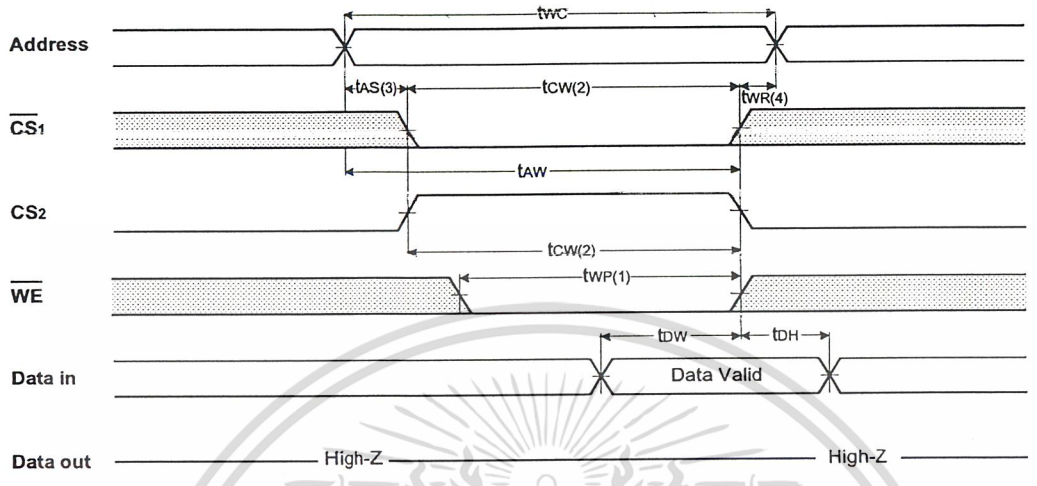


TIMING WAVEFORM OF WRITE CYCLE(2) ($\overline{CS_1}$ Controlled)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING WAVEFORM OF WRITE CYCLE(3) (CS_2 Controlled)

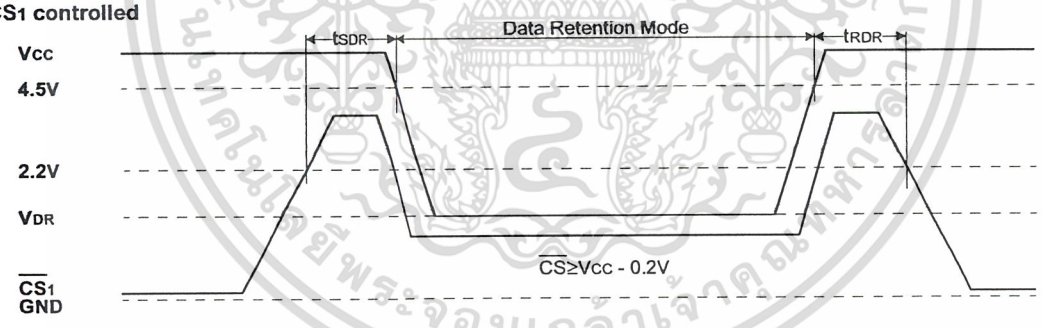


NOTES (WRITE CYCLE)

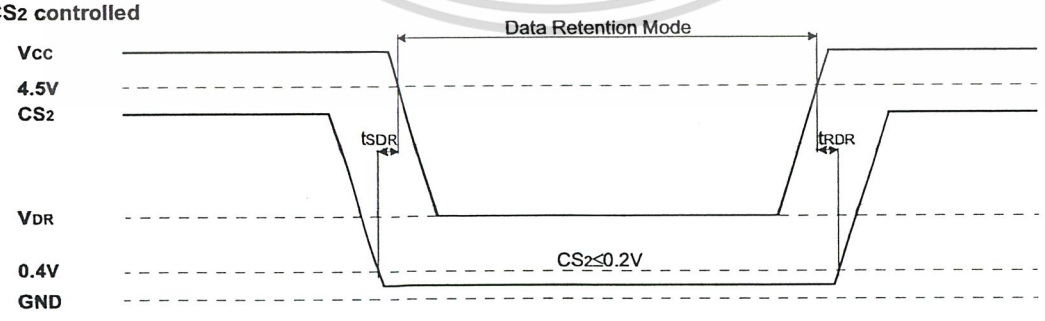
1. A write occurs during the overlap of a low $\overline{CS_1}$, a high CS_2 and a low $\overline{WE}$. A write begins at the latest transition among $\overline{CS_1}$ goes low, CS_2 going high and $\overline{WE}$ going low: A write end at the earliest transition among $\overline{CS_1}$ going high, CS_2 going low and $\overline{WE}$ going high, t_{WP} is measured from the begining of write to the end of write.
2. t_{CW} is measured from the $\overline{CS_1}$ going low or CS_2 going high to the end of write.
3. t_{AS} is measured from the address valid to the beginning of write.
4. t_{WR} is measured from the end of write to the address change. t_{WR} applied in case a write ends as $\overline{CS_1}$ or $\overline{WE}$ going high t_{WR2} applied in case a write ends as CS_2 going to low.

DATA RETENTION WAVE FORM

$\overline{CS_1}$ controlled



CS_2 controlled

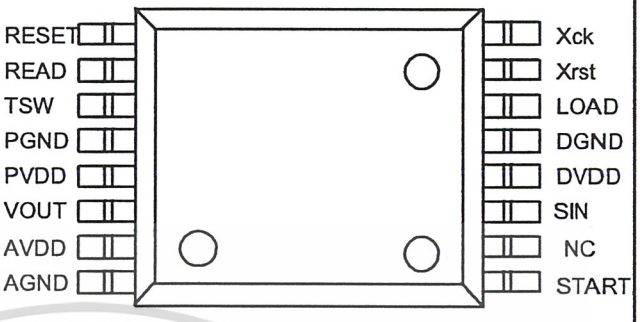


เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Description

M64282FP is a 128 x 128 pixel CMOS image sensor with built-in image processing and analog image output tuning functions. This device can detect an image and process the image simultaneously as human retinas can. M64282FP can achieve smaller system size, lower power consumption, and more intelligent image processing functions.

PIN CONFIGURATION (TOP VIEW)



Outline : 16C 9-B

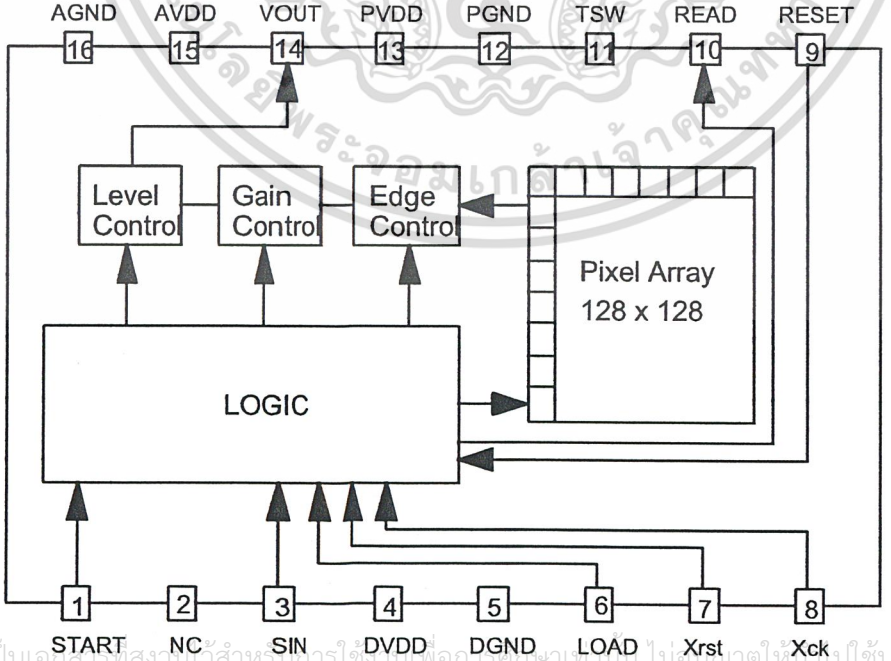
2. Features

- Single 5.0V supply
- Low power dissipation (Typ. 15 mW)
- Positive and negative image output
- Edge enhancement / extraction
- Output level & gain tuning

3. Application

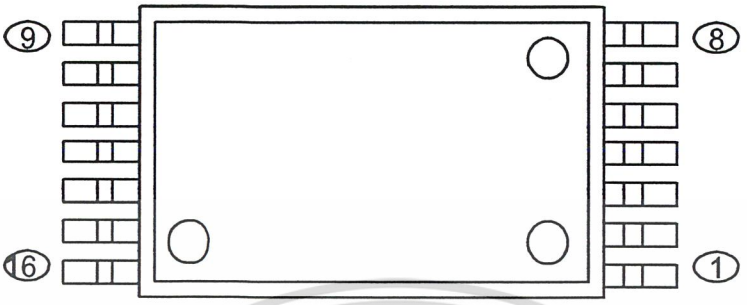
Image input device, Gaming, Human interface for PC, etc.

4. Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Pin Configuration



Pin No.	Symbol	Function	
1	START	Start Input	Image sensing start. Pulled down internally by 10k ohm.
2	NC1		Non Connect
3	SIN	Data Input	Parameter input. Pulled down internally by 10k ohm.
4	DVDD	Digital Power Supply	Power for logic circuits. Must be connected to 5.0 V digital supply.
5	DGND	Digital GND	Ground for logic parts.
6	LOAD	Data Set Input	Parameter set enable. Pulled down internally by 10k ohm.
7	Xrst	System Reset	System reset terminal. Pulled up internally by 10k ohm. Low active.
8	Xck	System Clock Input	Clock input for MUX. Pulled down internally by 10k ohm.
9	RESET	Memory Reset Input	Parameter register reset. Pulled up internally by 10k ohm. Low active
10	READ	Read Image	Read image signal.
11	TSW	Reserved	NOTE: Don't connect to this pin.
12	AGND1	Analog GND	Ground for analog circuits.
13	AVDD1	Analog Power Supply	Power for analog circuits. Must be connected to 5.0 V analog supply.
14	Vout	Signal Output	Analog image signal output in voltage.
15	AVDD2	Analog Power Supply	Power for analog circuits. Must be connected to 5.0 V analog supply.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16	AGND2	Analog GND	Ground for analog circuits.

6. Image Sensing Specifications 1

	Item	Specification
1	Resolution	128 x 123
2	Optical System	1/4 inch

7. Image Sensing Specifications 2

	Item	Specification
1	Detectable Illumination Range (Faceplate)	1lx ~ 10000lx*
2	Exposure Time Range	16 µsec ~ 1sec
3	System Clock (Xck)	500KHz
4	Frame Rate	10 fps ~ 30 fps
5	Output Voltage Range (Vout)	2.0Vp-p

* Under Halogen Light Valve Illumination

8. Electrical Specifications - Absolute Maximum Ratings

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
DVDD	Digital Power Supply Voltage	4.5	5.0	5.5	V
AVDD	Analog Power Supply Voltage	4.5	5.0	5.5	V

9. Electrical specifications - DC Specifications

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
VOH	"H" Output Voltage (READ)	4.5		DVDD	V
VOL	"L" Output Voltage (READ)	0.0		0.5	V
VIH	"H" Input Voltage	2.2		DVDD	V
VIL	"L" Input Voltage	0.0		0.8	V

10. AC Timing Requirements

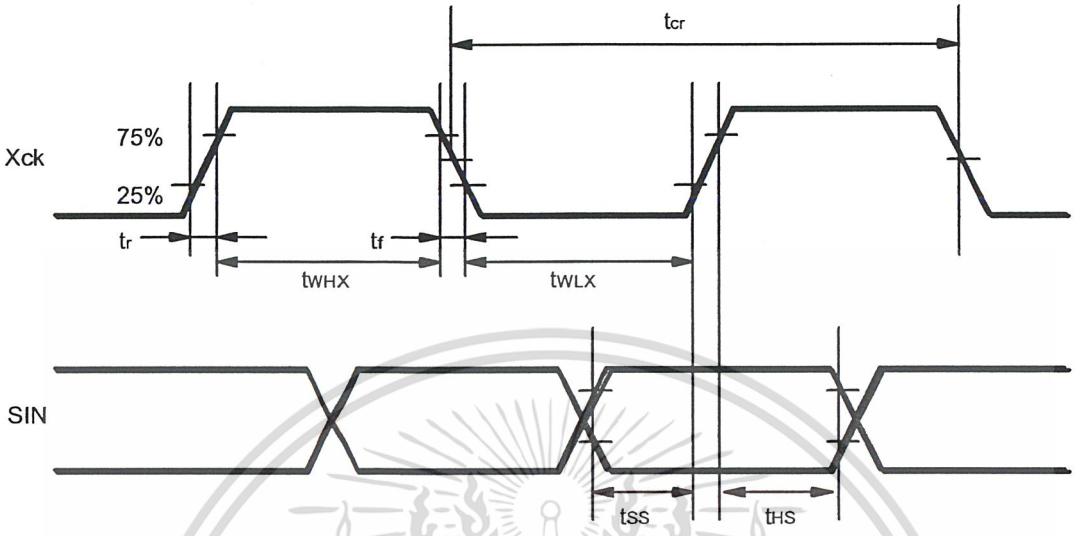
See the waveforms on page 5 and 6.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

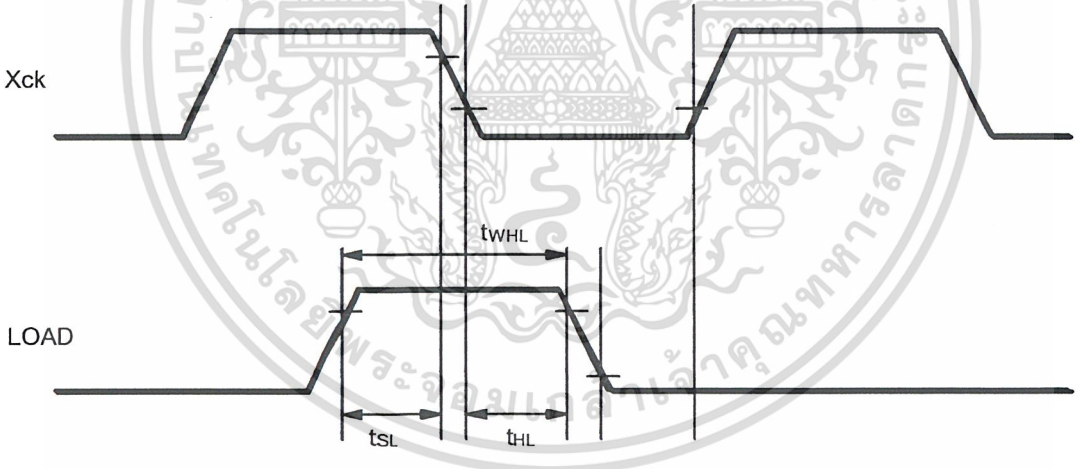
Symbol	Parameter	Limits			Units
		Min	Typ.	Max	
tcr	Xck cycle time	2	-	-	µs
tWHX	Xck high pulse width	0.8	-	-	µs
tWLX	Xck low pulse width	0.8	-	-	µs
tr	Xck rise time	-	-	0.2	µs
tf	Xck fall time	-	-	0.2	µs
tSS	SIN setup time	0.4	-	-	µs
tHS	SIN hold time	0.4	-	-	µs
tSL	LOAD setup time	0.4	-	-	µs
tHL	LOAD hold time	0.4	-	tWLX-0.4	µs
tWHL	LOAD high pulse width	0.8	-	-	µs
tSXR	Xrst setup time	0.4	-	-	µs
tHXR	Xrst hold time	0.4	-	-	µs
tSR	RESET setup time	0.4	-	-	µs
tHR	RESET hold time	0.4	-	-	µs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(A) Xck, SIN Timing

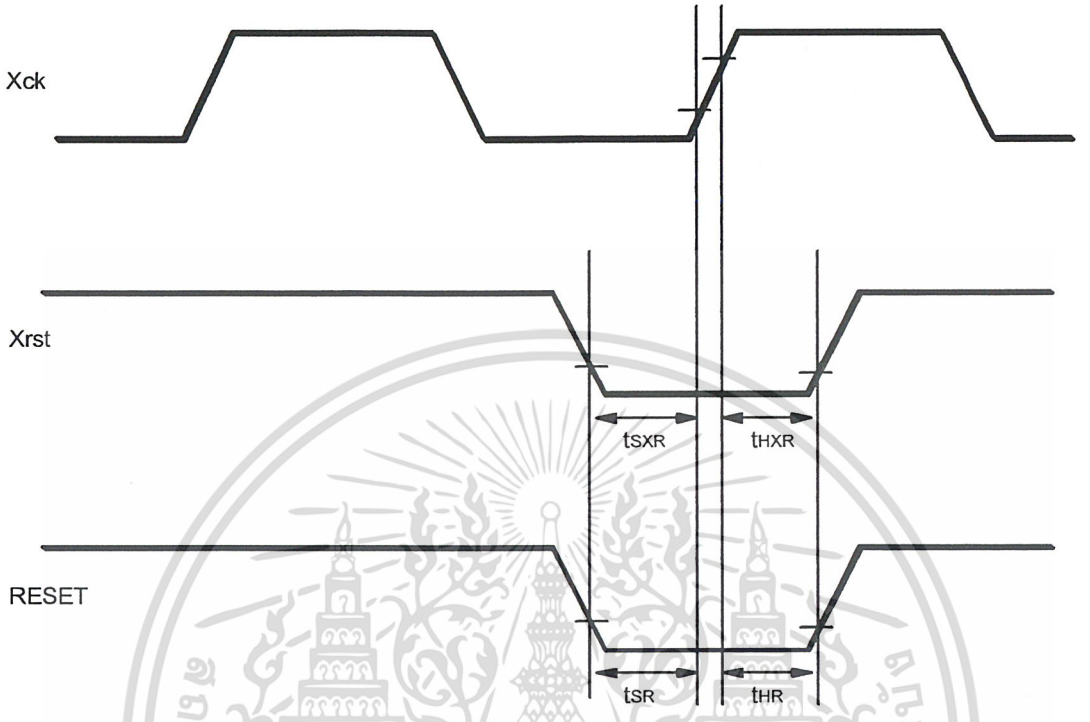


(B) Xck, LOAD Timing

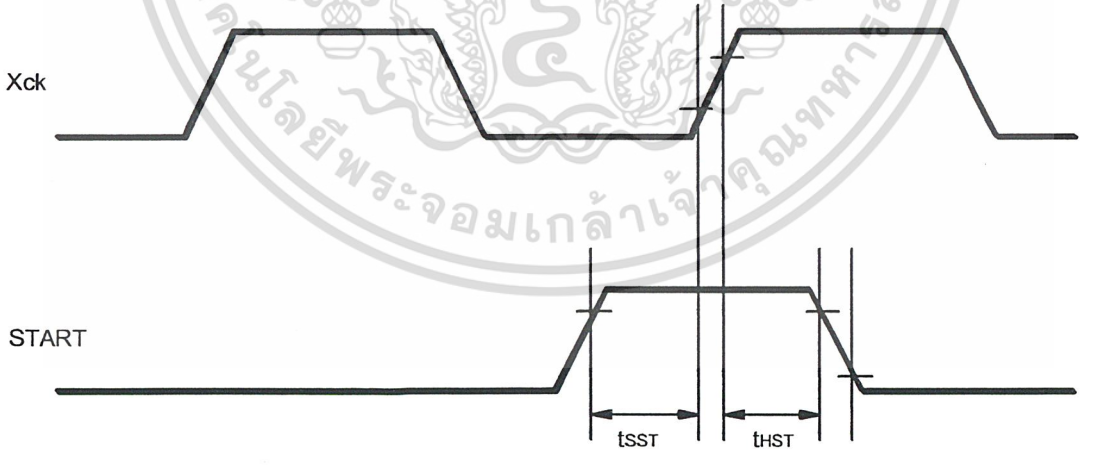


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(C) Xck, Xrst, RESET Timing



(D) Xck, START Timing



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. Operation

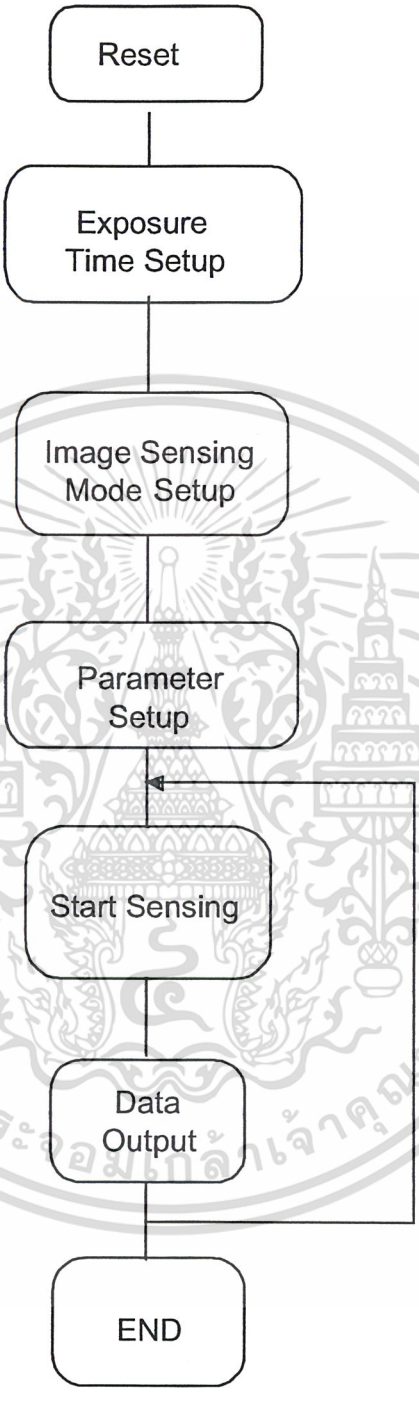


Fig. 11-1 Operation Flow Chart

Figure 11-1 shows the image sensing sequence. First of all, all the registers must be reset and must be initialized to the appropriate values. The reset sequence completes when both Xrst and RESET signals are set low. There are 8 sets of registers, each of

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

which consists of 8 bits of data. Each input data consists of 11 bits; of these 11 bits, the first 3 bits are the address and the last 8 bits are the data. The input data is latched at the rising edge of Xck when LOAD signal is high, and the data of a register become valid at the falling edge of Xck.

After all register are set, START signal must be asserted. Then, image sensing sequence starts at the rising edge of Xck. Image sensing sequence consists of two different processes: the exposing process to adjust the light intensity and the image read process to put out the image data after converting optical signal into electrical signal. After the exposure time defined by the registers 2 and 3 has passed, analog image data (total 16k pixels) is read out. To read image signal, READ signal must be asserted. At this moment, it becomes possible to change the registers, because the registers are irrelevant to the image read sequence.

Once image sensing sequence starts, the chip will continue to put out image data until it is reset.

11.1. Parameter Register Assignments

Symbol	Bit Assignment	Operation
N	1 bit	Exclusively set edge enhancement mode
VH	2 bits	Select vertical - horizontal edge operation mode
E	4 bits	Edge enhancement ratio
Z	2 bits	Zero point calibration (Set the dark level output signal to Vref)
I	1 bit	Select inverted/non-inverted output
C0, C1	8 bits x 2	Exposure time
O	6 bits	Output reference voltage (In both plus and minus direction)
V	3 bits	Output node bias voltage (Vref)
G	5 bits	Analog output gain
P, M, X	8 bits x 3	1-D filtering kernel.

11.2. Image Acquisition Modes

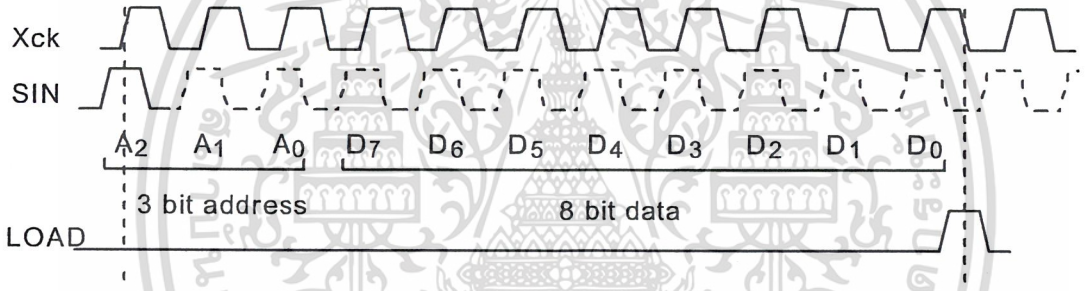
(a)	Positive Image	Set “P” register
(b)	Negative Image 1	Set “I” register
(c)	Negative Image 2	Set “M” register (optional)
(d)	Edge Extraction (V, H, 2-D)	Set “N” and “VH” register
(e)	Edge Extraction (1-D)	Set “P” and “M” register (optional)
(f)	Edge Enhancement	Set “N”, “VH” and “E” register
(g)	Offset Level Output	Set “0” to both “C0” and “C1”

11.3 Register Assignment

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reg. No.	Address	7	6	5	4	3	2	1	0
1	001	N	VH1	VH0	G4	G3	G2	G1	G0
2	010	C17	C16	C15	C14	C13	C12	C11	C10
3	011	C07	C06	C05	C04	C03	C02	C01	C00
4	100	P7	P6	P5	P4	P3	P2	P1	P0
5	101	M7	M6	M5	M4	M3	M2	M1	M0
6	110	X7	X6	X5	X4	X3	X2	X1	X0
7	111	E3	E2	E1	E0	I	V2	V1	V0
0	000	Z1	Z0	O5	O4	O3	O2	O1	O0

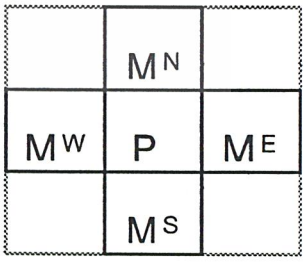
11.4. Data LOAD sequence



11.5. About the image processing functions

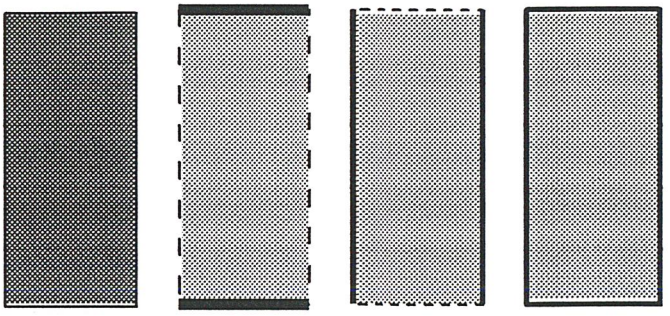
Artificial retina chip can put out positive, negative, edge extracted, and edge enhanced image in accordance with the parameter register settings.

On-chip image processing is done with the 3 x 3 neighboring pixels. This chip executes subtraction between the central pixel P and the four neighboring pixels MN, MS, MW, and ME(see the right figure) to realize edge extraction as shown below: vertical edge (V edge), horizontal edge (H edge), and 2 dimensional edge (2-D edge). Moreover, this chip can program the weight value of the central pixel P and the other four pixels to produce edge enhanced images.



Four connected neighboring pixels

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Positive V edge H edge 2-D edge

Edge Modes	Output Signal	Effective Pixels
V-Edge Extraction	{2P - (MN+MS)} x α P-MS	128(H) x 121(V) 128(H) x 123(V)
H-Edge Extraction	{2P - (MW+ME)} x α	128(H) x 123(V)
2D-Edge Extraction	{4P - (MN+MS+ME+MW)} x α	128(H) x 121(V)
V-Edge Enhancement	P + {2P - (MN+MS)} x α	128(H) x 121(V)
H-Edge Enhancement	P + {2P - (MW+ME)} x α	128(H) x 123(V)
2D-Edge Enhancement	P+{4P - (MN+MS+ ME+ MW)} x α	128(H) x 121(V)

α is the edge enhancement ratio set by "E" register.

P and M indicate the signal value from each pixel.

11.6. Register Descriptions

11.6.1. "N" register (1 bit)

If the "N" register is set, P and M registers (see 11.6.11.) are exclusively set to a specific vertical edge extraction / enhancement mode. In the case of "H", for example, P register is set to 02 (HEX), and M register is set to 05 (HEX). If "N" register is set, access to P and M registers is always ignored.

11.6.2. "VH" register (2 bits)

The "VH" register selects vertical, horizontal, and 2-dimensional edge extraction/enhancement operation.

Register Setting		Edge Mode
VH1	VH0	
0	0	No edge operation
0	1	Horizontal edge mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1	0	Vertical edge mode
1	1	2-D edge mode

11.6.3. "E" register (4 bits)

The "E" register sets the edge enhancement ratio α. The most significant bit E3 specifies edge enhancement mode or edge extraction mode: "H" for edge extraction mode and "L" for edge enhancement mode. (In the case of normal image sensing operation, E3 should be set low) The ratio α is set as follows. 100 % means the same level as the P signal, which is the signal of the central pixel in the 3x3 processing kernel.

Register Setting			Edge Enhancement Ratio
E2	E1	E0	
0	0	0	50 %
0	0	1	75 %
0	1	0	100 %
0	1	1	125 %
1	0	0	200 %
1	0	1	300 %
1	1	0	400 %
1	1	1	500 %

11.6.4. "Z" Register (2 bits)

It calibrates the zero value by setting the dark level output signal to Vref.

Register Setting		Calibration
Z1	Z0	
0	0	No calibration
1	0	Calibration for positive signal
0	1	Calibration for negative signal

11.6.5. "I" register (1 bit)

If the "I" register is set to "H", the output signal is inverted. if it is set to "L", the signal is not inverted.

11.6.6. "C0 & C1" register (8 bits x 2)

Both C0 and C1 registers determine Exposure time; the sum of the value of C0 register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

and that of C1 register determines the actual exposure time.

The offset level of image output can be obtained by setting both of C0 and C1 registers to 00 (the minimum exposure time). In this case, all pixels are read out as black level (optical black). The signal output format is the same as that of the normal output image. (Synchronized with the READ signal.)

"C0" register (8 bits)

Register Setting	Exposure time (msec)
00 (HEX)	0
FF (HEX)	4.080

Step width	16 μsec
Step number	256

"C1" register (8 bits)

Register Setting	Exposure time (msec)
00 (HEX)	0
FF (HEX)	1044.5

Step width	4.096 msec
Step number	256

Notice: In the case of vertical edge extraction / enhancement mode, the exposure time should be greater than 0.768 msec.

11.6.7. "O" register (6 bits)

The "O" register adjusts the offset level of the signal voltage. The most significant bit O5 is the sign bit: "H" for plus direction, "L" for minus direction modulation. The offset is adjusted by 5 bit accuracy. The maximum absolute value of the offset level is 1V.

(In the case O5 is "H".)

Register Setting	Offset voltage (V)
20 (HEX)	0
3F (HEX)	1

Step width	32mV
Step number	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(In the case O5 is "L")

Register Setting	Offset voltage (V)
00 (HEX)	0
1F (HEX)	-1

Step width	-32mV
Step number	32

11.6.8. "V" register (3 bits)

It sets the output node voltage Vref.

Register Setting			Vref (V)
V2	V1	V0	
0	0	0	0.0
0	0	1	0.5
0	1	0	1.0
0	1	1	1.5
1	0	0	2.0
1	0	1	2.5
1	1	0	3.0
1	1	1	3.5

11.6.9. "G" register (5 bits)

The "G" register sets the output gain of the image output signal. If the most significant bit G4 is "H", The total gain increases by 6dB.

Register Setting				Total Gain (dB)	
G3	G2	G1	G0	G4	
				0	1
0	0	0	0	14.0	20.0
0	0	0	1	15.5	21.5
0	0	1	0	17.0	23.0
0	0	1	1	18.5	24.5
0	1	0	0	20.0	26.0
0	1	0	1	21.5	27.5
0	1	1	0	23.0	29.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0	1	1	1	24.5	30.5
1	0	0	0	26.0	32.0
1	0	0	1	29.0	35.0
1	0	1	0	32.0	38.0
1	0	1	1	35.0	41.0
1	1	0	0	38.0	44.0
1	1	0	1	41.0	47.0
1	1	1	0	45.5	51.5
1	1	1	1	51.5	57.5

11.6.10. Typical register settings

The following table shows the typical image acquisition modes and the register settings to each mode.

Mode	Z1	Z0	N	VH1	VH0	P	M	X	E3	I
Posi.	1	0	0	0	0	01	00	01	0	0
Inv.	1	0	0	0	0	01	00	01	0	1
H Enh.	1	0	0	0	1	01	00	01	0	0
H Enh. Inv.	1	0	0	0	1	01	00	01	0	1
H Ext.	0	0	0	0	1	01	00	01	1	0
H Ext. Inv.	0	0	0	0	1	01	00	01	1	1
V Enh.	1	0	1	1	0	01	00	01	0	0
V Enh. Inv.	1	0	1	1	0	01	00	01	0	1
V Ext.	0	0	1	1	0	01	00	01	1	0
V Ext. Inv.	0	0	1	1	0	01	00	01	1	1
2D Enh.	1	0	1	1	1	01	00	01	0	0
2D Enh. Inv.	1	0	1	1	1	01	00	01	0	1
2D Ext.	0	0	1	1	1	01	00	01	1	0
2D Ext. Inv.	0	0	1	1	1	01	00	01	1	1

11.6.11. P and M register settings for programmable 1-D filtering

Both P and M registers specify 1-D filtering kernel consisting of +1, 0 and -1 weight values. The kernel is processed with the image pixels along vertical direction.

The following table shows the typical image acquisition modes and the register settings. Some modes are the same as those of the table in 11.6.10. Moreover, "Negative" output in the table is the same as the "Inverted" output in the table in 11.6.10. V(PM) mode works similar to the filtering pattern of (+1, -1).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mode	Z1	Z0	N	VH1	VH0	P	M	X	E3	I
Posi.	1	0	0	0	0	01	00	01	0	0
Neg.	0	1	0	0	0	00	01	01	0	1
V (PM)	0	0	0	0	0	01	02	01	0	0
V Enh.	1	0	0	1	0	02	05	01	0	0
V Ext.	0	0	0	1	0	02	05	01	1	0
2D Enh.	1	0	0	1	1	02	05	01	0	0
2D Ext.	0	0	0	1	1	02	05	01	1	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

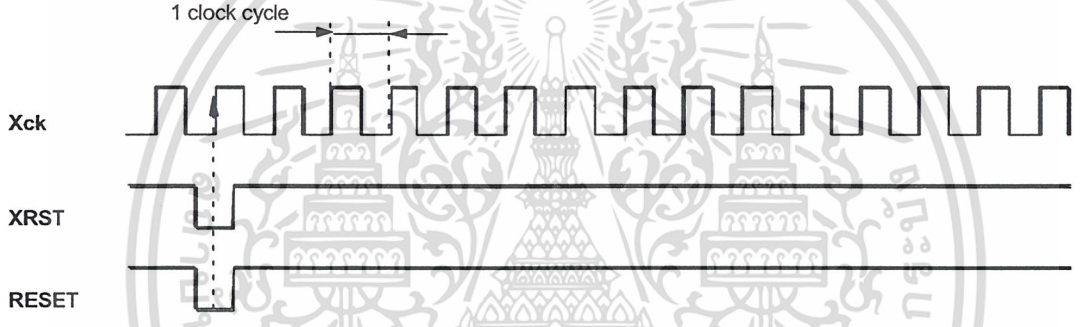
12. Operation Timing

(1) Chip Reset
Reset timing of the chip. Reset timing is synchronized with the rising edge of Xck clock.

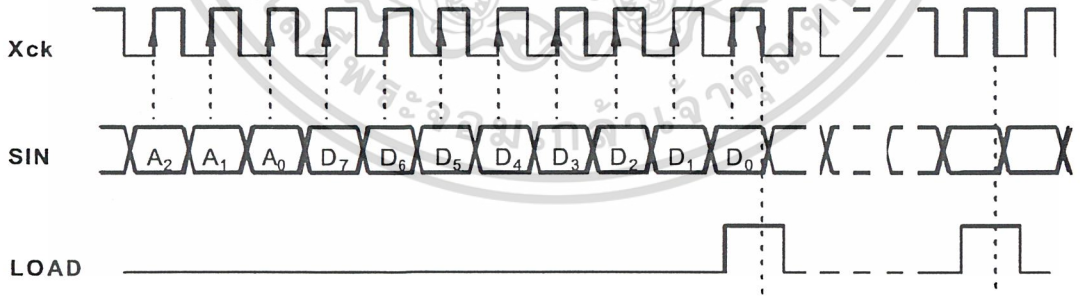
(2) Data Input
Settings of the exposure time, initial value of each scanner, Vref value, and gain value. Data (8 bits x 8) input timing is synchronized with the rising edge of Xck clock.

(3) Image Data Output
Image pixel data is serially read out synchronized with Xck clock.

(1) Chip Reset



(2) Data Input

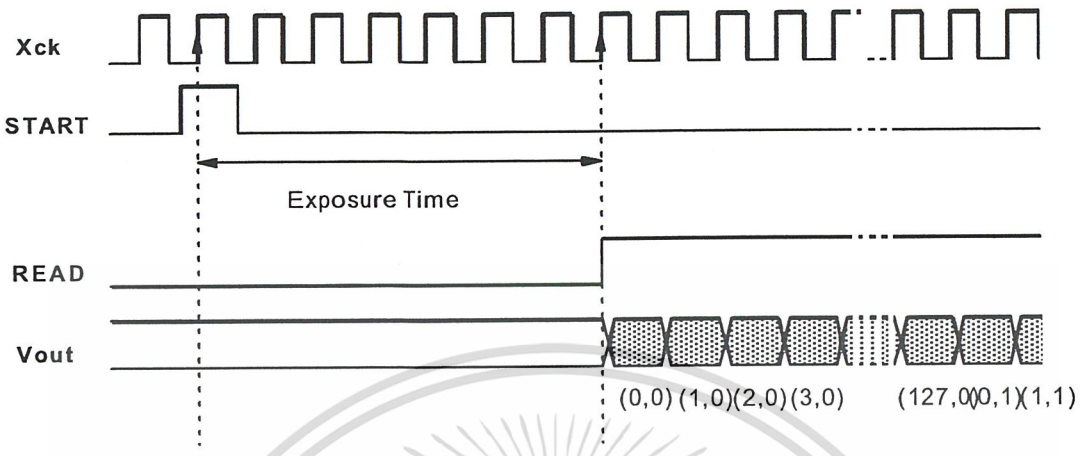


Data 0 becomes valid

Data 7 becomes valid

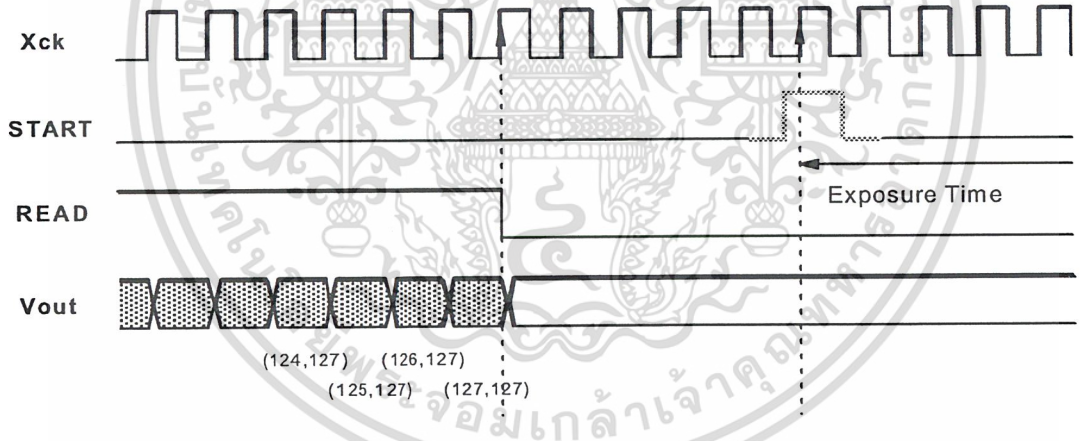
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) Image Data Output



Start sensing 1st frame.
(start exposre)

Start 1st frame
image data output.



End sensing 1st frame.

Start sensing 2nd frame.
(START signal is automatically generated inside the chip 5 clocks after the fall edge of READ.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA SHEET

PDIUSBD11
USB device with serial interface

Product specification
Supersedes data of 1999 Nov 19

1999 Jul 22

Philips Semiconductors

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ทำซ้ำ ดัดแปลง เผยแพร่ หรือกระทำการใดๆ ในลักษณะอื่นใดที่ก่อให้เกิดความเสียหายแก่เจ้าของลิขสิทธิ์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงความเป็นเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

USB device with serial interface — PDIUSBD11

FEATURES

- Complies with the Universal Serial Bus specification Rev. 1.1
- Complies with the ACPI, OnNOW, and USB power management requirements
- Compliant with USB Human Interface Devices and Monitor Control Class
- Compliant with System Management Bus Specification Rev. 1.0
- Integrated SIE (Serial Interface Engine), FIFO memory and transceivers
- Automatic USB protocol handling
- High speed I^2C Interface (up to 1 Mbit/s)
- Compatible with the PDIUSBH11 software
- Software controllable connection to USB bus (SoftConnect™)
- Low frequency 12 MHz crystal oscillator eases EMI design issues
- Programmable output clock frequency
- Bus powered capability with very low suspend current
- Controllable LazyClock output during suspend
- Single 3.3 V supply with 5 V tolerant I/O
- Available in 16-pin DIP and SO packages
- Full-scan design with high fault coverage (>99%) insures high quality
- Higher than 8 kV in-circuit ESD protection lowers cost of extra components

DESCRIPTION

The Universal Serial Bus hub PDIUSBD11 is a cost and feature-optimized USB interface device. It is used in microcontroller-based systems and communicates with the system microcontroller over the high speed I^2C serial bus. This modular approach to implementing USB functions allows the designer to choose the optimum system microcontroller from the available wide variety. This flexibility cuts down the development time, risks, and costs by allowing the use of the existing architecture and the firmware investments. This results in the fastest way to develop the most cost-effective USB peripheral solutions. The PDIUSBD11 is ideally suited for computer monitors, docking stations, keyboards, and many other applications that use the I^2C or the SMBus-based architecture.

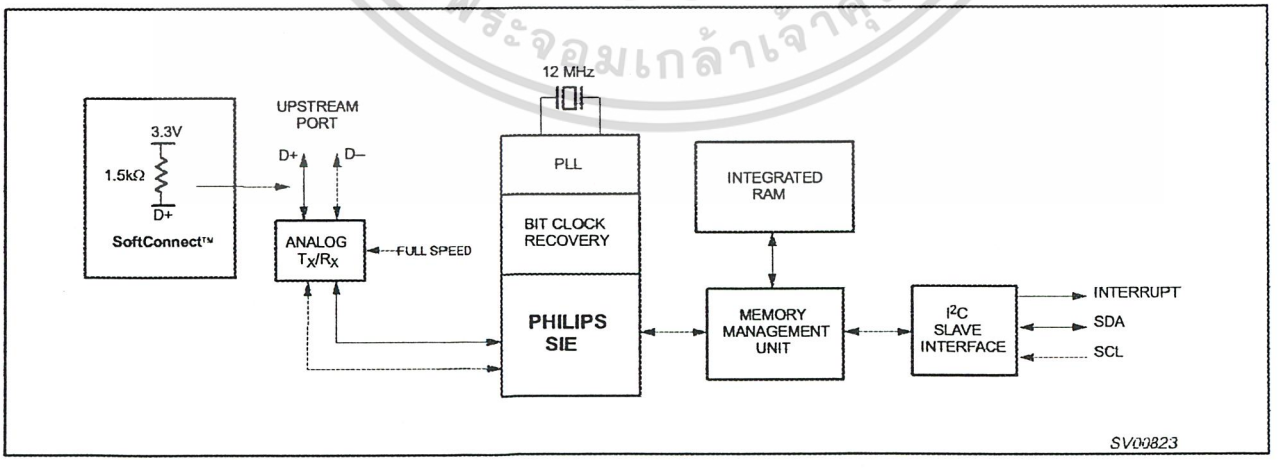
The PDIUSBD11 conforms to the USB specification Rev. 1.1, I^2C serial interface and the SMBus specifications. It is fully compliant with the Human Interface Device Class and Monitor Control Class specifications. Its low suspend power consumption along with the programmable LazyClock output allows for easy implementation of equipment that is compliant to the ACPI, OnNOW, and USB power management requirements. The low operating power allows the implementation of bus-powered function.

The PDIUSBD11 is fully backward compatible to the PDIUSBH11/PDIUSBH11A software. In addition, it also incorporates the feature enhancements like SoftConnect™, LazyClock, programmable clock output, lower frequency crystal oscillator, multiple function endpoints and integration of termination resistors. All of these feature enhancements contribute to significant cost savings in the system implementation and at the same time ease the implementation of advanced USB functionality into the peripherals.

ORDERING INFORMATION

PACKAGES	TEMPERATURE RANGE	OUTSIDE NORTH AMERICA	NORTH AMERICA	PKG. DWG. #
16-pin plastic SO	–40°C to +85°C	PDIUSBD11 D	PDIUSBD11 D	SOT162-1
16-pin plastic DIP	–40°C to +85°C	PDIUSBD11 N	PDIUSBD11 N	SOT38-4

BLOCK DIAGRAM



NOTE:

1. This is a conceptual block diagram and does not include each individual signal.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analog Transceiver

The transceiver interfaces directly to the USB cables through some termination resistors. They are capable of transmitting and receiving serial data at "full speed" (12 Mbit/s) only.

PLL

A 12 MHz to 48 MHz clock multiplier PLL (Phase-Locked Loop) is integrated on-chip. This allows for the use of low-cost 12 MHz crystal. EMI is also minimized due to lower frequency crystal. No external components are needed for the operation of the PLL.

Bit Clock Recovery

The bit clock recovery circuit recovers the clock from the incoming USB data stream using 4X over-sampling principle. It is able to track jitter and frequency drift specified by the USB specification.

Philips Serial Interface Engine (PSIE)

The Philips SIE implements the full USB protocol layer. It is completely hardwired for speed and needs no firmware intervention. The functions of this block include: synchronization pattern recognition, parallel/serial conversion, bit stuffing/de-stuffing, CRC checking/generation, PID verification/generation, address recognition, handshake evaluation/generation.

Memory Management Unit (MMU) and Integrated RAM

The MMU and the integrated RAM is used to handle the large difference in data rate between USB, running in bursts of 12 Mbit/s and the I^2C interface to the microcontroller, running at up to 1 Mbit/s. This allows the microcontroller to read and write USB packets at its own speed through I^2C.

I^2C Slave Interface

This block implements the necessary I^2C interface protocol. A slave I^2C allows for simple micro-coding. An interrupt is used to alert the microcontroller whenever the PDIUSBD11 needs attention. As a slave I^2C device, the PDIUSBD11 I^2C clock: SCL is an input and is controlled by the microcontroller. The I^2C interface can run up to 1 Mbit/s.

SoftConnect™

The connection to the USB is accomplished by bringing D+ (for high-speed USB device) high through a 1.5 kΩ pull-up resistor. In the PDIUSBD11, the 1.5 kΩ pull-up resistor is integrated on-chip and is not connected to V_{CC} by default. The connection of the internal resistor to Vcc is established through a command sent by the external/system microcontroller. This allows the system microcontroller to complete its initialization sequence before deciding to establish connection to the USB. Re-initialization of the USB bus connection can also be affected without requiring the pull out of the cable.

The PDIUSBD11 will check for USB VBUS availability before the connection can be established. VBUS sensing is provided through VBUS pin.

It should be noted that the tolerance of the internal resistors is higher (30%) than that specified by the USB specification (5%). However, the overall V_{SE} voltage specification for the connection can still be met with good margin. The decision to make sure of this feature lies with the users.

SoftConnect™ is a patent pending technology from Philips Semiconductors.

ENDPOINT DESCRIPTIONS

ENDPOINT#	ENDPOINT INDEX	TRANSFER TYPE	DIRECTION	MAX PACKET SIZE (BYTES)
0	2	Control	OUT	8
	3		IN	8
1	5	Generic	OUT	8
	4		IN	8
2	6	Generic	OUT	8
	7		IN	8
3	8	Generic	OUT	8
	9		IN	8

NOTE:
1. Generic endpoint can be used for Interrupt or Bulk endpoint.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

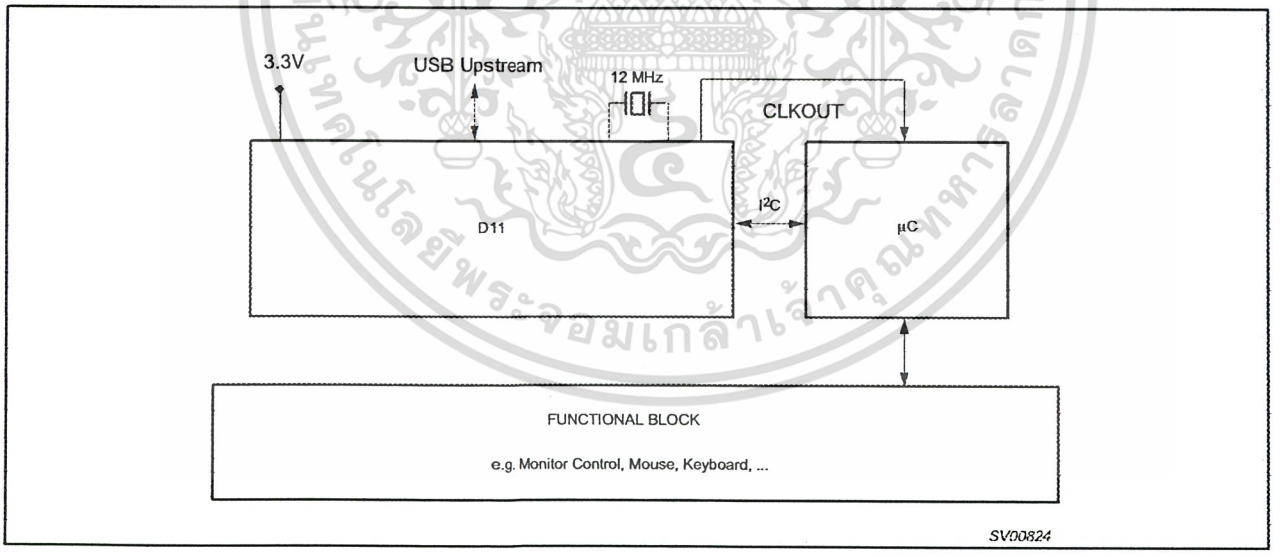
PIN DESCRIPTION

PIN NO	PIN SYMBOL	TYPE	DRIVE	DESCRIPTION
1	TEST	Input		Connect to GND for normal operation
2	RESET_N	Input	ST	Power-on reset
3	XTAL1	Input		Crystal connection 1 (12MHz)
4	XTAL2	Output		Crystal connection 2 (12MHz)
5	CLKOUT	Output	3 mA	Programmable output clock for external devices
6	V_{CC}	Power		Voltage supply 3.3V±0.3V
7	SUSPEND	Output	OD6	Device is in suspended state
8	INT_N	Output	OD6	Connect to microcontroller interrupt
9	SDA	I/O	OD6	I^2C bi-directional data
10	SCL	I/O	OD6	I^2C bit-clock
11	GND	Power		Ground reference
12	DP	AI/O		USB D+ connection
13	DM	AI/O		USB D– connection
14	AGND	Power		Analog ground reference
15	AV_{CC}	Power		Analog voltage supply 3.3V±0.3V
16	VBUS	Input		USB VBUS sensing pin

NOTES:

1. Signals ending in _N indicate active LOW signals.
   ST: Schmitt Trigger
   OD6: Open Drain with 6 mA drive
   AI/O: Analog I/O

APPLICATION DIAGRAM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I²C Interface

The I²C bus is used to interface to an external microcontroller needed to control the operation of the USB device. For cost consideration, the target system microcontroller can be shared and utilized for both the functional part as well as the USB protocol interfacing. The PDIUSBD11 implements a slave I²C interface. When the PDIUSBD11 needs to communicate with the microcontroller it asserts an interrupt signal. The microcontroller services this interrupt by reading the appropriate status register on the PDIUSBD11 through the I²C bus. (For more information about the I²C serial bus, refer to the *I²C Handbook*, Philips order number 9397 750 00013).

The I²C interface on the PDIUSBD11 defines two types of transactions:

- **command transaction** – A command transaction is used to define which data (e.g., status byte, buffer data, ...) will be read from/written to the USB interface in the next data transaction. A data transaction usually follows a command transaction.
- **data transaction** – A data transaction reads data from/writes data to the USB interface. The meaning of the data is dependent on the command transaction which was sent before the data transaction.

Two addresses are used to differentiate between command and data transactions. Writing to the command address is interpreted as a command, while reading from/writing to the data address is used to transfer data between the PDIUSBH11A and the controller.

ADDRESS TABLE

Type of Address	Physical Address MSB to LSB (Binary)
Command	0011 011
Data	0011 010

Protocol

An I²C transaction starts with a Start Condition, followed by an address. When the address matches either the command or data address the transaction starts and runs until a Stop Condition or another Start Condition (repeated start) occurs.

The command address is write-only and is unable to do a read. The next bytes in the message are interpreted as commands. Several command bytes can be sent after one command address. Each of the command bytes is acknowledged and passed on to the Memory Management Unit inside the PDIUSBD11.

When the Start Condition address matches the data address, the next bytes are interpreted as data. When the RW bit in the address indicates a *master writes data to slave* (='0') the bytes are received, acknowledged and passed on to the Memory Management Unit. If the RW bit in the address indicates a *master reads data from slave* (='1') the PDIUSBD11 will send data to the master. The I²C-master must acknowledge all data bytes except the last one. In this way the I²C interface knows when the last byte has been transmitted and it then releases the SDA line so that the master controller can generate the Stop Condition.

Repeated start support allows another packet to be sent without generating a Stop Condition.

Timing

The I²C interface in the PDIUSBD11 can support clock speeds up to 1 MHz.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMAND SUMMARY

Some commands have the same command code (e.g., Read Buffer and Write Buffer). In these cases, the direction of the Data Phase (read or write) indicates which command is executed.

COMMAND NAME	RECIPIENT	CODING	DATA PHASE
Initialization Commands			
Set Address/Enable	Function	D1h	Write 1 byte
Set Endpoint Enable	Function	D8h	Write 1 byte
Set Mode	Function	F3h	Write 2 byte
Data Flow Commands			
Read Interrupt Register		F4h	Read 2 bytes
Select Endpoint	Control OUT Endpoint	00h	Read 1 byte (optional)
	Control IN Endpoint	01h	Read 1 byte (optional)
	Other Endpoints	00h+Endpoint Index	Read 1 byte (optional)
Read Last Transaction Status	Control OUT Endpoint	40h	Read 1 byte
	Control IN Endpoint	41h	Read 1 byte
	Other Endpoints	40h+Endpoint Index	Read 1 byte
Read Endpoint Status	Control OUT Endpoint	80h	Read 1 byte
	Control IN Endpoint	81h	Read 1 byte
	Other Endpoints	80h+Endpoint Index	Read 1 byte
Read Buffer	Selected Endpoint	F0h	Read n bytes
Write Buffer	Selected Endpoint	F0h	Write n bytes
Set Endpoint Status	Control OUT Endpoint	40h	Write 1 byte
Set Endpoint Status	Control IN Endpoint	41h	Write 1 byte
	Other Endpoints	40h+Endpoint Index	Write 1 byte
Acknowledge Setup	Selected Endpoint	F1h	None
Clear Buffer	Selected Endpoint	F2h	None
Validate Buffer	Selected Endpoint	FAh	None
General Commands			
Send Resume		F6h	None
Read Current Frame Number		F5h	Read 1 or 2 bytes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMAND DESCRIPTIONS

Command Procedure

There are three basic types of commands: Initialization, Data Flow, and General commands. Respectively, these are used to initialize the function; for data flow between the function and the host; and some general commands.

Initialization Commands

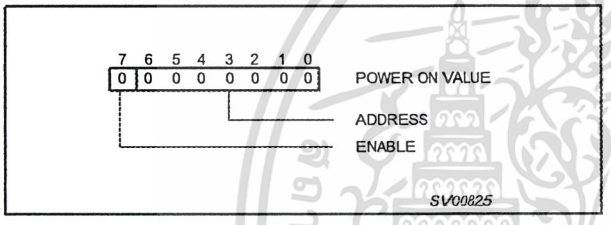
Initialization commands are used during the enumeration process of the USB network. These commands are used to enable the function endpoints. They are also used to set the USB assigned address.

Set Address / Enable

Command : D1h, (Function)

Data : Write 1 byte

This command is used to set the USB assigned address and enable the function.



Address — The value written becomes the address.

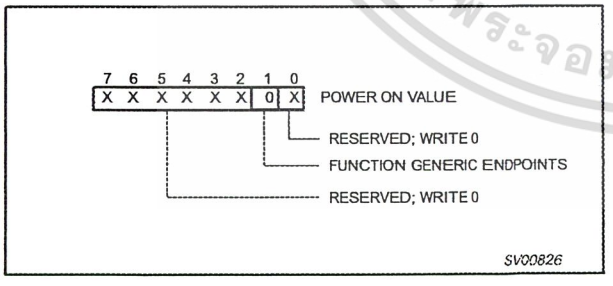
Enable — A '1' enables this function.

Set Endpoint Enable

Command : D8h

Data : Write 1 byte

The generic endpoints can only be enabled when the function is enabled via the Set Address/Enable command.



Function Generic Endpoint — A value of '1' indicates the function generic endpoints are enabled.

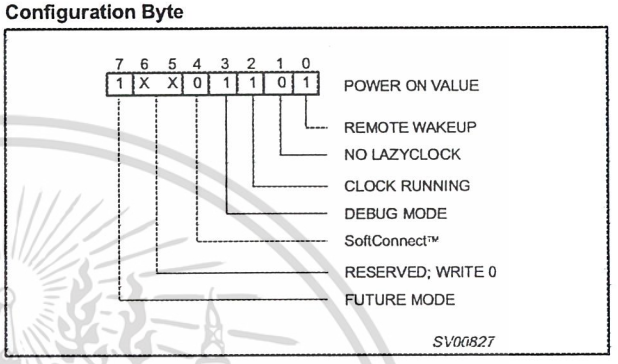
Set Mode

Command : F3h

Data : Write 2 bytes

The Set Mode command is followed by two data writes. The first byte contains the configuration byte values. The second byte is the clock division factor byte.

Configuration Byte



Remote Wakeup — A '1' indicates that a remote wakeup feature is ON. Bus reset will set this bit to '1'.

No LazyClock — A '1' indicates that CLKOUT will not switch to LazyClock. A '0' indicates that the CLKOUT switches to LazyClock 1ms after the Suspend pin goes high. LazyClock frequency is 30KHz ±40%. The programmed value will not be changed by a bus reset.

Clock Running — A '1' indicates that the internal clocks and PLL are always running even during Suspend state. A '0' indicates that the internal clock, crystal oscillator and PLL are stopped whenever not needed. To meet the strict Suspend current requirement, this bit needs to be set to '0'. The programmed value will not be changed by a bus reset.

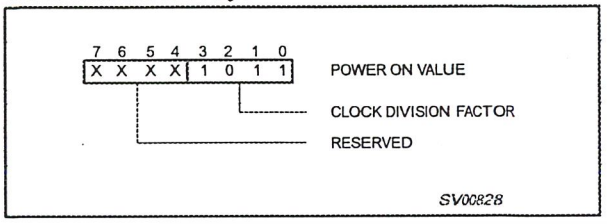
Debug Mode — A '1' indicates that all errors and "NAKing" are reported and a '0' indicates that only OK and babbling are reported. The programmed value will not be changed by a bus reset.

SoftConnect™ — A '1' indicates that the upstream pull-up resistor will be connected if VBUS is available. A '0' means that the upstream resistor will not be connected. The programmed value will not be changed by a bus reset.

FutureMode — Write a '1'.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Clock Division Factor Byte



Clock Division Factor — The value indicates clock division factor for CLKOUT. The output frequency is 48 MHz/(N+1) where N is the Clock Division Factor. The reset value is 11. This will produce the output frequency of 4 MHz which can then be programmed up (or down) by the user. The minimum value is one giving the range of frequency from 4 to 24 MHz. The PDIUSBD11 design ensures no glitching during frequency change. The programmed value will not be changed by a bus reset.

Data Flow Commands

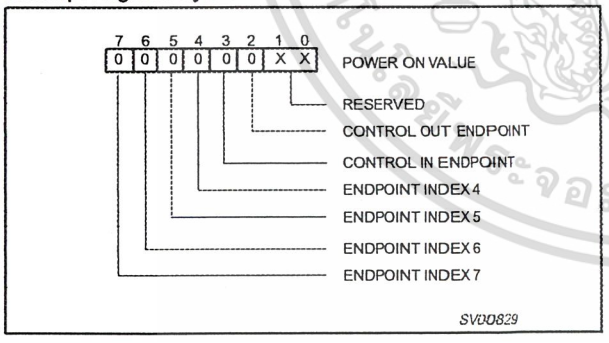
Data flow commands are used to manage the data transmission between the USB endpoints and the monitor. Much of the data flow is initiated via an interrupt to the microcontroller. The microcontroller utilizes these commands to access and determine whether the endpoint FIFOs have valid data.

Read Interrupt Register

Command : **F4h**

Data : **Read 2 bytes**

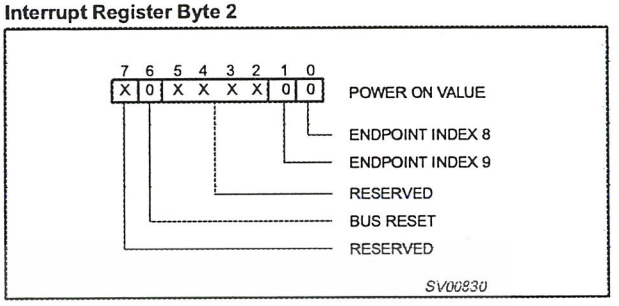
Interrupt Register Byte 1



This command indicates the origin of an interrupt. A '1' indicates an interrupt occurred at this endpoint. The bits are cleared by reading the endpoint status register through the Read Endpoint Status command.

After a bus reset, an interrupt will be generated and bit 6 of the Interrupt Register Byte 2 will be '1'. The interrupt is internally cleared by reading the interrupt register. A bus reset is completely identical to the hardware reset through the RESET_N pin with the sole difference of interrupt notification.

Interrupt Register Byte 2

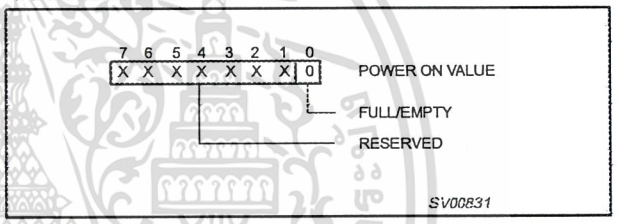


Select Endpoint

Command : **00-0Dh**

Data : **Optional Read 1 byte**

The Select Endpoint command initializes an internal pointer to the start of the Selected buffer. Optionally, this command can be followed by a data read, which returns '0' if the buffer is empty and '1' if the buffer is full.



Full/Empty — A '1' indicates the buffer is full, '0' indicates an empty buffer.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

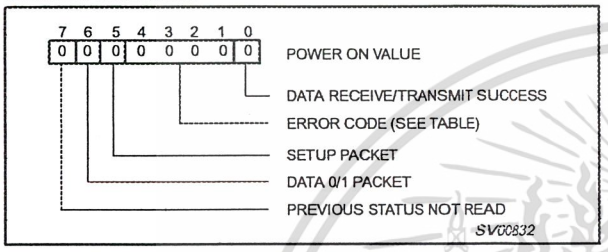
Read Last Transaction Status

Command : 40–4Dh

Data : Read 1 byte

The Read Last Transaction Status command is followed by one data read that returns the status of the last transaction of the endpoint. This command also resets the corresponding interrupt flag in the interrupt register, and clears the status, indicating that it was read.

This command is useful for debugging purposes. Since it keeps track of every transaction, the status information is overwritten for each new transaction.



Data Receive/ Transmit Success — A '1' indicates data has been received or transmitted successfully.

Error Code — See Table 1, Error Codes.

Setup Packet — A '1' indicates the last successful received packet had a SETUP token (this will always read '0' for IN buffers).

Data 0/1 Packet — A '1' indicates the last successful received or sent packet had a DATA1 PID.

Previous Status not Read — A '1' indicates a second event occurred before the previous status was read.

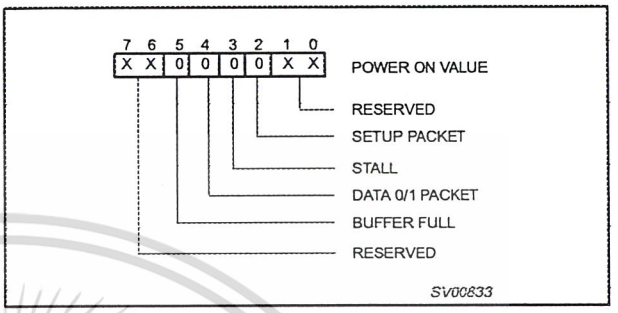
Table 1. ERROR CODES

ERROR CODE	RESULT
0000	No Error
0001	PID encoding Error; bits 7–4 are not the inversion of bits 3–0
0010	PID unknown; encoding is valid, but PID does not exist
0011	Unexpected packet; packet is not of the type expected (= token, data or acknowledge), or SETUP token to a non-control endpoint
0100	Token CRC Error
0101	Data CRC Error
0110	Time Out Error
0111	Babble Error
1000	Unexpected End-of-packet
1001	Sent or received NAK
1010	Sent Stall, a token was received, but the endpoint was stalled
1011	Overflow Error, the received packet was longer than the available buffer space
1101	Bitstuff Error
1111	Wrong DATA PID; the received DATA PID was not the expected one

Read Endpoint Status

Command : 80–8Dh

Data : Read 1 byte



Setup Packet — A '1' indicates the last received packet had a SETUP token.

STALL — A '1' indicates the endpoint is stalled.

Data 0/1 Packet — A '1' indicates if the last received or sent packet had a DATA1 PID.

Buffer Full — A '1' indicates that the buffer is full.

Read Buffer

Command : F0h

Data : Read multiple bytes (max 10)

The Read Buffer command is followed by a number of data reads, which return the contents of the selected endpoint data buffer. After each read, the internal buffer pointer is incremented by 1.

The buffer pointer is not reset to the buffer start by the Read Buffer command. This means that reading or writing a buffer can be interrupted by any other command (except for Select Endpoint), or can be done by more than one I^2C transaction (read the first 2 bytes to get the number of data bytes, then read the rest in other transactions).

The data in the buffer are organized as follows:

- byte 0: Reserved: can have any value
- byte 1: Number/length of data bytes
- byte 2: Data byte 1
- byte 3: Data byte 2

. . . . . .

Write Buffer

Command : F0h

Data : Write multiple bytes (max 10)

The Write Buffer command is followed by a number of data writes, which load the endpoints buffer. The data must be organized in the same way as described in the Read Buffer command. The first byte (reserved) should always be '0'. As in the Read Buffer command, the data can be split up into different I^2C data transactions.

WARNING:

There is no protection against writing or reading over a buffer's boundary or against writing into an OUT buffer or reading from an IN buffer. Any of these actions could cause an incorrect operation. Data in an OUT buffer are only meaningful after a successful transaction.

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Clear Buffer

Command : **F2h**

Data : **None**

When a packet is received completely, an internal endpoint buffer full flag is set. All subsequent packets will be refused by returning a NAK. When the microcontroller has read the data, it should free the buffer by the Clear Buffer command. When the buffer is cleared, new packets will be accepted.

Validate Buffer

Command : **FAh**

Data : **None**

When the microprocessor has written data into an IN buffer, it should set the buffer full flag by the Validate Buffer command. This indicates that the data in the buffer are valid and can be sent to the host when the next IN token is received.

Set Endpoint Status

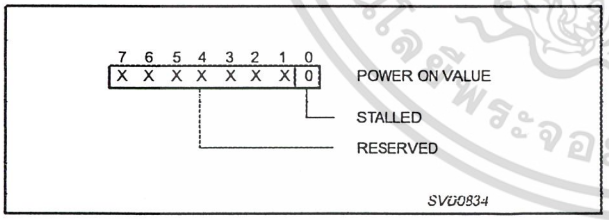
Command : **40–4Dh**

Data : **Write 1 byte**

A stalled control endpoint is automatically unstalled when it receives a SETUP token, regardless of the content of the packet. If the endpoint should stay in its stalled state, the microcontroller can re-stall it.

When a stalled endpoint is unstalled (either by the Set Endpoint Status command or by receiving a SETUP token), it is also re-initialized. This flushes the buffer and if it is an OUT buffer it waits for a DATA 0 PID, if it is an IN buffer it writes a DATA 0 PID.

Even when unstalled, writing Set Endpoint Status to '0' initializes the endpoint.



Stalled A '1' indicates the endpoint is stalled.

Acknowledge Setup

Command : **F1h**

Data : **None**

The arrival of a SETUP packet flushes the IN buffer and disables the Validate Buffer and Clear Buffer commands for both IN and OUT endpoints.

The microcontroller needs to re-enable these commands by the Acknowledge Setup command. This ensures that the last SETUP packet stays in the buffer and no packet can be sent back to the host until the microcontroller has acknowledged explicitly that it has seen the SETUP packet.

The microcontroller must send the Acknowledge Setup command to both the IN and OUT endpoints.

GENERAL COMMANDS

Send Resume

Command : **F6h**

Data : **None**

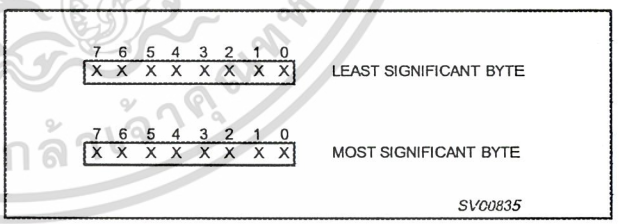
Sends an upstream resume signal for 10 ms. This command is normally issued when the device is in suspend. The RESUME command is not followed by a data read or write.

Read Current Frame Number

Command : **F5h**

Data : **Read One or Two Bytes**

This command is followed by one or two data reads and returns the frame number of the last successfully received SOF. The frame number is returned Least Significant Byte first.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RECOMMENDED OPERATING CONDITIONS

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
V_{CC}	DC supply voltage		3.0	3.6	V
V_I	DC input voltage range		0	5.5	V
$V_{I/O}$	DC input voltage range for I/O		0	5.5	V
$V_{AI/O}$	DC input voltage range for analog I/O		0	V_{CC}	V
V_O	DC output voltage range		0	V_{CC}	V
T_{amb}	Operating ambient temperature range in free air	See DC and AC characteristics per device	–40	85	°C

ABSOLUTE MAXIMUM RATINGS[1]

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
V_{CC}	DC supply voltage		–0.5	+4.6	V
I_{IK}	DC input diode current	$V_I < 0$		–50	mA
V_I	DC input voltage	Note 2	–0.5	+5.5	V
$V_{I/O}$	DC input voltage range for I/O		–0.5	V_{CC} + 0.5	V
I_{OK}	DC output diode current	$V_O > V_{CC}$ or $V_O < 0$		±50	mA
V_O	DC output voltage	Note 2	–0.5	V_{CC} + 0.5	V
I_O	DC output sink or source current for other pins	V_O = 0 to V_{CC}		±15	mA
I_O	DC output sink or source current for D+/D– pins	V_O = 0 to V_{CC}		±50	mA
I_{GND}, I_{CC}	DC V_{CC} or GND current			±100	mA
T_{STG}	Storage temperature range		–60	+150	°C
P_{TOT}	Power dissipation per package				

NOTES:
1. Stresses beyond those listed may cause damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those listed in the RECOMMENDED OPERATING CONDITIONS table is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.
2. The input and output voltage ratings may be exceeded if the input and output current ratings are observed.

DC CHARACTERISTICS (Digital pins)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
Input Levels						
V_{IL}	LOW level input voltage				0.6	V
VIH	HIGH level input voltage		2.7			V
V_{TLH}	LOW to HIGH threshold voltage	ST (Schmitt Trigger) pins	1.4		1.9	V
V_{THL}	HIGH to LOW threshold voltage	ST pins	0.9		1.5	V
V_{HYS}	Hysteresis voltage	ST pins	0.4		0.7	V
Output Levels						
V_{OL}	LOW level output voltage	I_{OL} = rated drive			0.4	V
		I_{OL} = 20 µA			0.1	V
V_{OH}	HIGH level output voltage	I_{OH} = rated drive	2.4			V
		I_{OH} = 20 µA	V_{CC} – 0.1			V
Leakage Current						
I_{OZ}	OFF state current	OD (Open Drain) pins			±5	µA
I_L	Input leakage current				±5	µA
I_S	Suspend current	Oscillator stopped & inputs to GND/V_{CC}			15	µA
I_O	Operating current	I^2C operating		10		mA

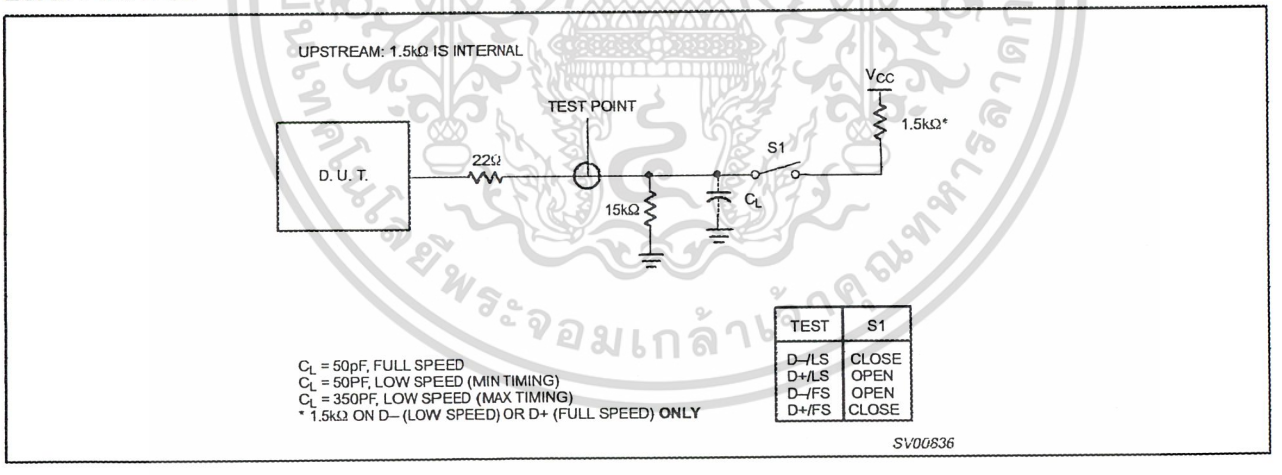
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC CHARACTERISTICS (AI/O pins)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Leakage Current					
I_{LO}	Hi-Z state data line leakage	$0V < V_{IN} < 3.3V$		±10	µA
Input Levels					
V_{DI}	Differential input sensitivity	$\|(D+) - (D-)\|$[1]	0.2		V
V_{CM}	Differential common mode range	Includes V_{DI} range	0.8	2.5	V
V_{SE}	Single-ended receiver threshold		0.8	2.0	V
Output Levels					
V_{OL}	Static output LOW	R_L of 1.5kΩ to 3.6V		0.3	V
V_{OH}	Static output HIGH	R_L of 15kΩ to GND	2.8	3.6	V
Capacitance					
C_{IN}	Transceiver capacitance	Pin to GND		20	pF
Output Resistance					
Z_{DRV}[2]	Driver output resistance	Steady state drive	29	44	Ω
Integrated Resistance					
Z_{PU}	Pull-up resistance	SoftConnect™ = ON	1.1	1.9	kΩ
Z_{PD}	Pull-down resistance	Pull-down = ON	11	19	kΩ

NOTES:
1. D+ is the symbol for the USB positive data pin: DP.
   D– is the symbol for the USB negative data pin: DM.
2. Includes external resistors of 22 Ω ± 1% each on D+ and D–.

LOAD FOR D+/D–



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC CHARACTERISTICS (AI/O pins, FULL speed)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Driver characteristics		C_L = 50 pF; R_{pu} = 1.5 kΩ on D+ to V_{CC}			
	Transition Time:				
t_r	Rise time	Between 10% and 90%	4	20	ns
t_f	Fall time		4	20	ns
t_{RFM}	Rise/fall time matching	(t_r/t_f)	90	110	%
V_{CRS}	Output signal crossover voltage		1.3	2.0	V
Driver Timings					
t_{EOPT}	Source EOP width	Figure 1	160	175	ns
t_{DEOP}	Differential data to EOP transition skew	Figure 1	–2	5	ns
Receiver Timings					
	Receiver Data Jitter Tolerance				
t_{JR1}	To next transition	Characterized and not tested.	–18.5	18.5	ns
t_{JR2}	For paired transitions	Guaranteed by design.	–9	9	ns
	EOP Width at Receiver				
t_{EOPR1}	Must reject as EOP	Figure 1	40		ns
t_{EOPR2}	Must accept		82		ns

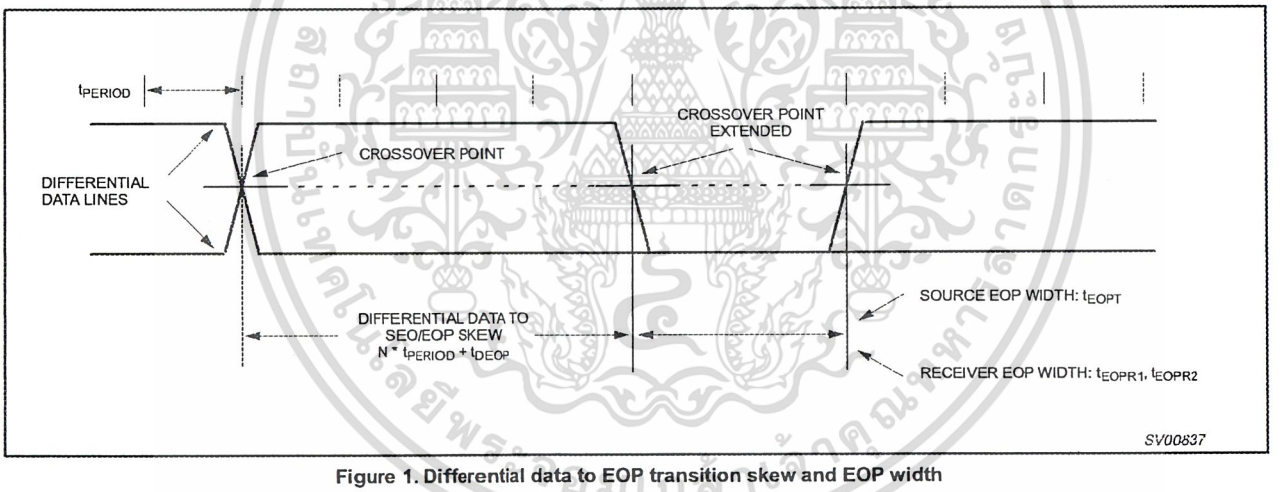


Figure 1. Differential data to EOP transition skew and EOP width

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC CHARACTERISTICS (I^2C pins)

All timing values are valid within the operating supply voltage and ambient temperature range and reference to V_{IL} and V_{IH} with an input voltage swing of V_{SS} and V_{DD}.

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
F_{SCL}	S_{CL} clock frequency			1000	kHz
t_{BUF}	Bus free time		0.5		µs
$t_{SU;STA}$	Start condition set-up time		0.25		µs
$t_{HD;STA}$	Start condition hold time		0.25		µs
t_{LOW}	S_{CL} LOW time		0.45		µs
t_{HIGH}	S_{CL} HIGH time		0.45		µs
t_r	S_{CL} and S_{DA} rise time			0.3	µs
t_f	S_{CL} and S_{DA} fall time			0.1	µs
$t_{SU;DAT}$	Data set-up time		100		ns
$t_{HD;DAT}$	Data hold time		0		ns
$t_{VD;DAT}$	S_{CL} LOW to data out valid			0.4	µs
$t_{SU;STO}$	Stop condition set-up time		0.25		µs

A detailed description of the I^2C-bus specification, with applications, is given in the brochure "*The I^2C-bus and how to use it*". This brochure may be ordered using the Philips order number 9398 393 40011.

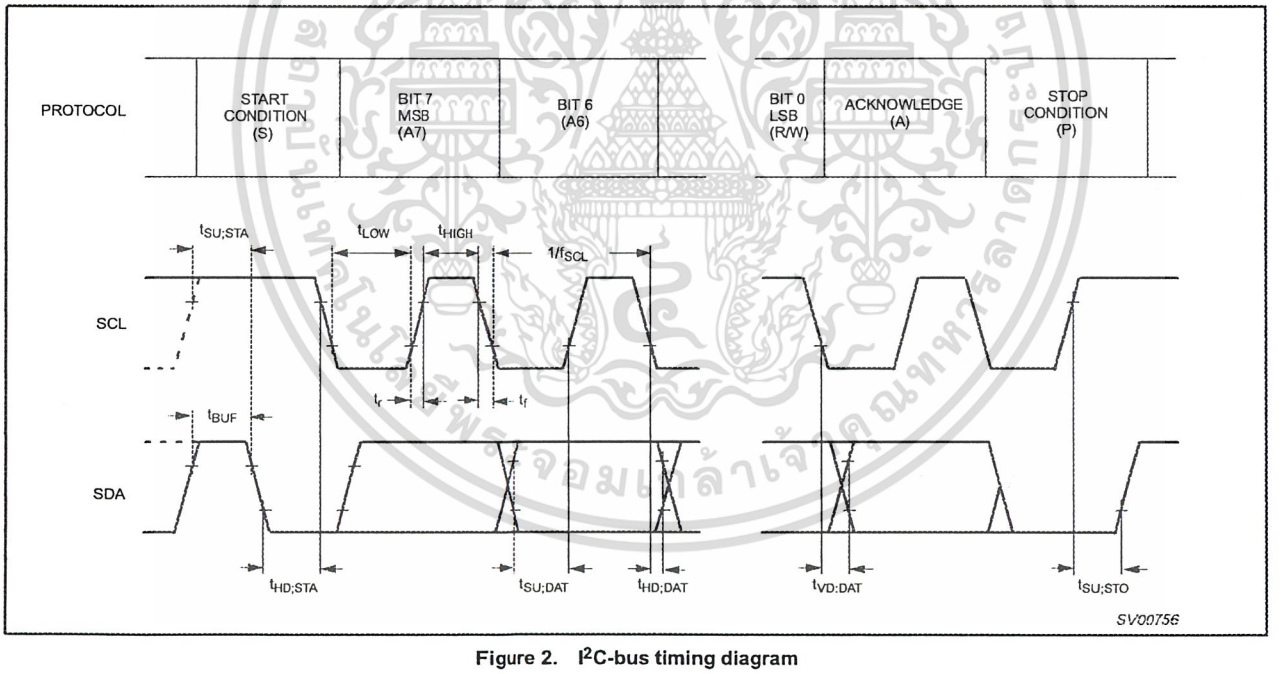
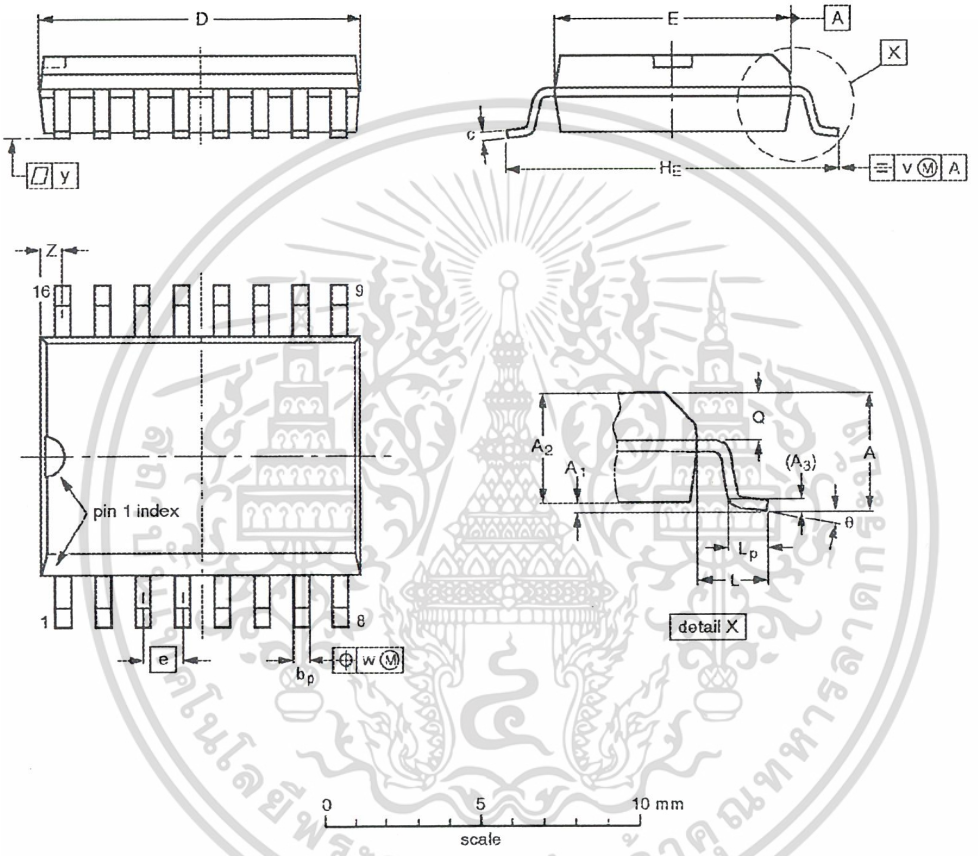


Figure 2. I^2C-bus timing diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SO16: plastic small outline package; 16 leads; body width 7.5 mm — SOT162-1



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_E	L	L_p	Q	v	w	y	$Z^{(1)}$	θ
mm	2.65	0.30 0.10	2.45 2.25	0.25	0.49 0.36	0.32 0.23	10.5 10.1	7.6 7.4	1.27	10.65 10.00	1.4	1.1 0.4	1.1 1.0	0.25	0.25	0.1	0.9 0.4	8° 0°
inches	0.10	0.012 0.004	0.096 0.089	0.01	0.019 0.014	0.013 0.009	0.41 0.40	0.30 0.29	0.050	0.42 0.39	0.055	0.043 0.016	0.043 0.039	0.01	0.01	0.004	0.035 0.016	

Note

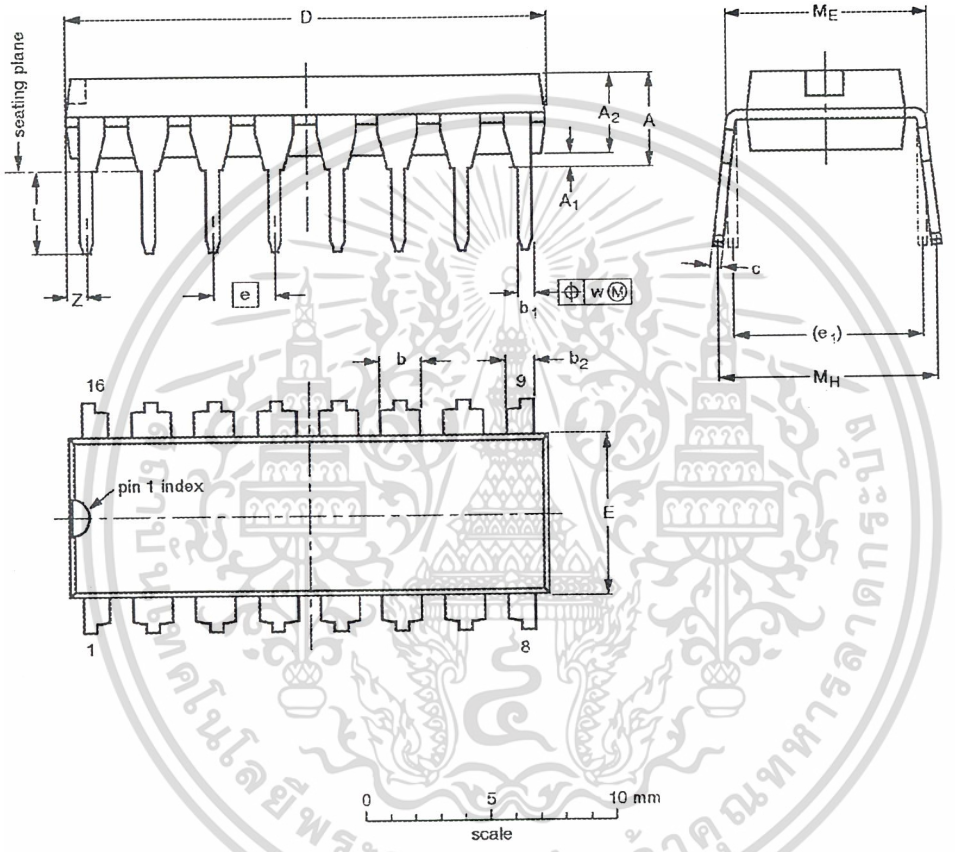
1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT162-1	075E03	MS-013AA				~~92-11-17~~ 95-01-24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DIP16: plastic dual in-line package; 16 leads (300 mil)

SOT38-4



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A_1 min.	A_2 max.	b	b_1	b_2	c	$D^{(1)}$	$E^{(1)}$	e	e_1	L	M_E	M_H	w	$Z^{(1)}$ max.
mm	4.2	0.51	3.2	1.73 1.30	0.53 0.38	1.25 0.85	0.36 0.23	19.50 18.55	6.48 6.20	2.54	7.62	3.60 3.05	8.25 7.80	10.0 8.3	0.254	0.76
inches	0.17	0.020	0.13	0.068 0.051	0.021 0.015	0.049 0.033	0.014 0.009	0.77 0.73	0.26 0.24	0.10	0.30	0.14 0.12	0.32 0.31	0.39 0.33	0.01	0.030

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT38-4						~~92-11-17~~ 95-01-14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOLDERING

Introduction

This text gives a very brief insight to a complex technology. A more in-depth account of soldering ICs can be found in our *"Data Handbook IC26; Integrated Circuit Packages"* (document order number 9398 652 90011).

There is no soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and surface mount components are mixed on one printed circuit board. However, wave soldering is not always suitable for surface mount ICs, or for printed-circuit boards with high population densities. In these situations, reflow soldering is often used.

Through-hole mount packages

SOLDERING BY DIPPING OR BY SOLDER WAVE

The maximum permissible temperature of the solder is 260°C; solder at this temperature must not be in contact with the joints for more than 5 seconds. The total contact time of successive solder waves must not exceed 5 seconds.

The device may be mounted up to the seating plane, but the temperature of the plastic body must not exceed the specified maximum storage temperature ($T_{stg(max)}$). If the printed-circuit board has been pre-heated, forced cooling may be necessary immediately after soldering to keep the temperature within the permissible limit.

MANUAL SOLDERING

Apply the soldering iron (24 V or less) to the lead(s) of the package, either below the seating plane or not more than 2 mm above it. If the temperature of the soldering iron bit is less than 300°C, it may remain in contact for up to 10 seconds. If the bit temperature is between 300 and 400°C, contact may be made for up to 5 seconds.

Surface mount packages

REFLOW SOLDERING

Reflow soldering requires solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the printed-circuit board by screen printing, stencilling or pressure-syringe dispensing before package placement.

Several methods exist for reflowing; for example, infrared/convection heating in a conveyor-type oven. Throughput times (preheating, soldering and cooling) vary between 100 and 200 seconds, depending on heating method.

Typical reflow peak temperatures range from 215 250°C. The top-surface temperature of the packages should preferably be kept below 230°C.

WAVE SOLDERING

Conventional single-wave soldering is not recommended for surface mount devices (SMDs) or printed-circuit boards with a high component density, as solder bridging and non-wetting can present major problems.

To overcome these problems, the double-wave soldering method was specifically developed.

If wave soldering is used, the following conditions must be observed for optimal results:

- Use a double-wave soldering method comprising a turbulent wave with high upward pressure followed by a smooth laminar wave.
- For packages with leads on two sides and a pitch (e):
  - larger than or equal to 1.27 mm, the footprint longitudinal axis is preferred to be parallel to the transport direction of the printed-circuit board;
  - smaller than 1.27 mm, the footprint longitudinal axis must be parallel to the transport direction of the printed-circuit board.

  The footprint must incorporate solder thieves at the downstream end.
- For packages with leads on four sides, the footprint must be placed at a 45° angle to the transport direction of the printed-circuit board. The footprint must incorporate solder thieves downstream and at the side corners.

During placement, and before soldering, the package must be fixed with a droplet of adhesive. The adhesive can be applied by screen printing, pin transfer or syringe dispensing. The package can be soldered after the adhesive has cured.

Typical dwell time is 4 seconds at 250°C. A mildly-activated flux will eliminate the need for removal of corrosive residues in most applications.

MANUAL SOLDERING

Fix the component by first soldering two diagonally-opposite end leads. Use a low-voltage (24 V or less) soldering iron applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300°C.

When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds between 270 and 320°C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SUITABILITY OF IC PACKAGES FOR WAVE, REFLOW AND DIPPING SOLDERING METHODS

Mounting	Package	Soldering Method		
		Wave	Reflow [1]	Dipping
Through-hole mount	DBS, DIP, HDIP, SDIP, SIL	suitable [2]	–	suitable
Surface mount	BGA, SQFP,	not suitable	suitable	–
	HLQFP, HSQFP, HSOP, SMS	not suitable [3]	suitable	–
	PLCC, SO, SOJ	suitable	suitable	–
	LQFP, QFP, TQFP	not recommended [4, 5]	suitable	–
	SSOP, TSSOP, VSO	not recommended [6]	suitable	–

NOTES:

1. All surface mount (SMD) packages are moisture sensitive. Depending upon the moisture content, the maximum temperature (with respect to time) and body size of the package, there is a risk that internal or external package cracks may occur due to vaporization of the moisture in them (the so-called "popcorn" effect). For details, refer to the Drypack information in the *"Data Handbook IC26; Integrated Circuit Packages; Section: Packing Methods"*.
2. For SDIP packages, the longitudinal axis must be parallel to the transport direction of the printed-circuit board.
3. These packages are not suitable for wave soldering as a solder joint between the printed-circuit board and heatsink (at bottom version) cannot be achieved, and as solder may stick to the heatsink (on top version).
4. If wave soldering is considered, then the package must be placed at a 45° angle to the solder wave direction. The package footprint must incorporate solder thieves downstream and at the side corners.
5. Wave soldering is only suitable for LQFP, QFP, and TQFP packages with a pitch (e) equal to or larger than 0.8 mm; it is definitely not suitable for packages with a pitch (e) equal to or smaller than 0.65 mm.
6. Wave soldering is only suitable for SSOP and TSSOP packages with a pith (e) equal to or larger than 0.65 mm; it is definitely not suitable for packages with a pitch (e) equal to or smaller than 0.5 mm.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data sheet status

Data sheet status	Product status	Definition [1]
Objective specification	Development	This data sheet contains the design target or goal specifications for product development. Specification may change in any manner without notice.
Preliminary specification	Qualification	This data sheet contains preliminary data, and supplementary data will be published at a later date. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.
Product specification	Production	This data sheet contains final specifications. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.

[1] Please consult the most recently issued datasheet before initiating or completing a design.

Definitions

Short-form specification — The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

Limiting values definition — Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

Application information — Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Disclaimers

Life support — These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

Right to make changes — Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

Philips Semiconductors
811 East Arques Avenue
P.O. Box 3409
Sunnyvale, California 94088–3409
Telephone 800-234-7381

© Copyright Philips Electronics North America Corporation 1998
All rights reserved. Printed in U.S.A.

Date of release: 07-99

Document order number: 9397–750–06219

Let's make things better.

Philips Semiconductors

PHILIPS

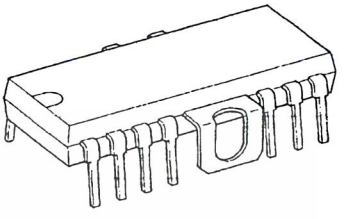
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TOSHIBA BIPOLAR LINEAR INTEGRATED CIRCUIT SILICON MONOLITHIC

TA7279P,TA7279AP

DUAL BRIDGE DRIVER

The TA7279P, TA7279AP are dual bridge driver designed for DC motor rotation control.



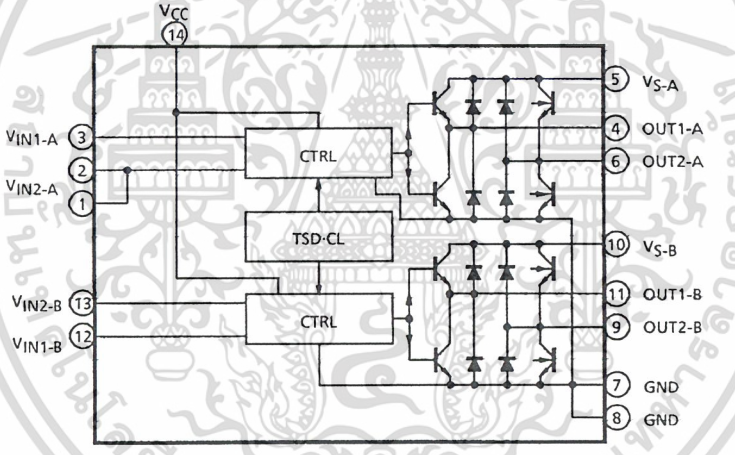
HDIP14-P-500-2.54A

Weight: 3.00 g (Typ.)

FEATURES

- Wide Range of Operating Voltage
  : V_{CC} (opr.) = 6~18 V (P, AP),
  V_S (opr.) = 0~16 V (P) / = 0~18 V (AP)
- Output Current Up to 1.0 A (AVE.), 3.0 A (PEAK)
- Built-in Thermal Shut Down and Current Limiter
- Input Hysteresis for Stable Operation

BLOCK DIAGRAM



961001EBA1

- TOSHIBA is continually working to improve the quality and the reliability of its products. Nevertheless, semiconductor devices in general can malfunction or fail due to their inherent electrical sensitivity and vulnerability to physical stress. It is the responsibility of the buyer, when utilizing TOSHIBA products, to observe standards of safety, and to avoid situations in which a malfunction or failure of a TOSHIBA product could cause loss of human life, bodily injury or damage to property. In developing your designs, please ensure that TOSHIBA products are used within specified operating ranges as set forth in the most recent products specifications. Also, please keep in mind the precautions and conditions set forth in the TOSHIBA Semiconductor Reliability Handbook.
- The products described in this document are subject to foreign exchange and foreign trade control laws.
- The information contained herein is presented only as a guide for the applications of our products. No responsibility is assumed by TOSHIBA CORPORATION for any infringements of intellectual property or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any intellectual property or other rights of TOSHIBA CORPORATION or others.
- The information contained herein is subject to change without notice.

เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

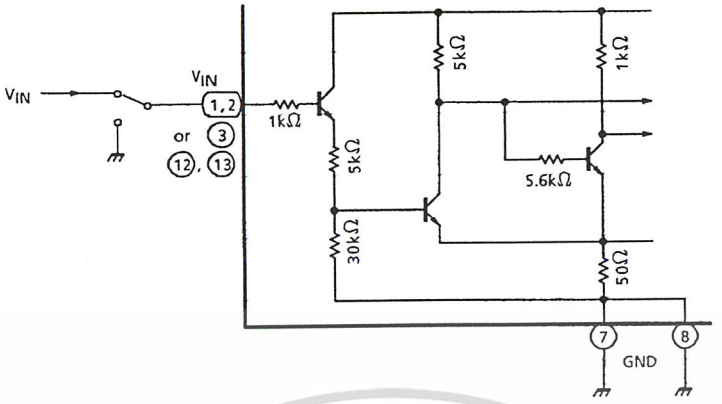
PIN FUNCTION

PIN No.	SYMBOL	FUNCTIONAL DESCRIPTION
1	V_{IN2-A}	A−ch input terminal
2	V_{IN2-A}	
3	V_{IN1-A}	A−ch input terminal
4	OUT1−A	A−ch output terminal
5	V_{S-A}	A−ch Motor drive power supply
6	OUT2−A	A−ch output terminal
7	GND	GND terminal
8	GND	
9	OUT2−B	B−ch output terminal
10	V_{S-B}	B−ch Moter drive power supply
11	OUT1−B	B−ch output terminal
12	V_{IN1-B}	B−ch input terminal
13	V_{IN2-B}	B−ch input terminal
14	V_{CC}	Logic power supply

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

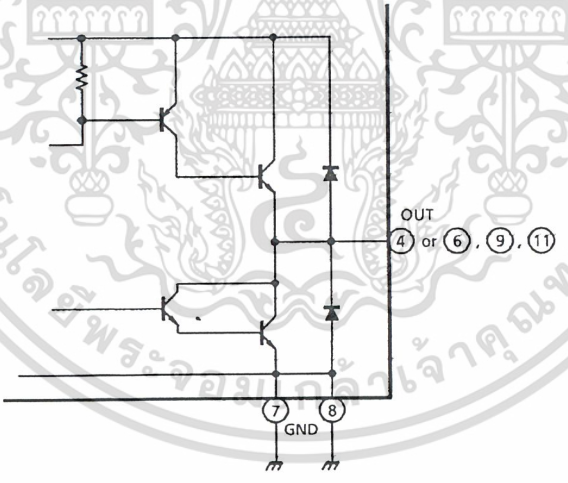
APPLICATION NOTE

(1) Input circuit



Input terminals of (2), (3), (12) and (13) Pin are all high active type and have a hysteresis.
3 μA Typ. of input current is required.
The input circuit is an active high type, as shown in the diagram. When voltage higher than the specified V_{IN} (H) is applied, the output is logic "H". When voltage lower than the specified V_{IN} (L) is applied or if the input is grounded, the output is logic "L". Since the input current IN flows to the input when logic "H", be careful with the output impedance at the previous step.

(2) Output circuit



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FUNCTION

IN1	IN2	OUT1	OUT2	MODE
1	1	L	L	Brake
0	1	L	H	CW / CCW
1	0	H	L	CCW / CW
0	0	High Impedance		STOP

MAXIMUM RATINGS (Ta = 25°C)

CHARACTERISTIC		SYMBOL	RATING	UNIT
Supply Voltage	AP	V_{CC} (MAX.)	25	V
	P		20	
Motor Drive Voltage	AP	V_S (MAX.)	25	V
	P		18	
Output Current	PEAK	I_O (PEAK)	3.0	A
	AVE.	I_O (AVE.)	1.0	
Power Dissipation		P_D (Note)	2.3	W
Operating Temperature		T_{opr}	−30~75	°C
Storage Temperature		T_{stg}	−55~150	°C

Note: No heat sink.

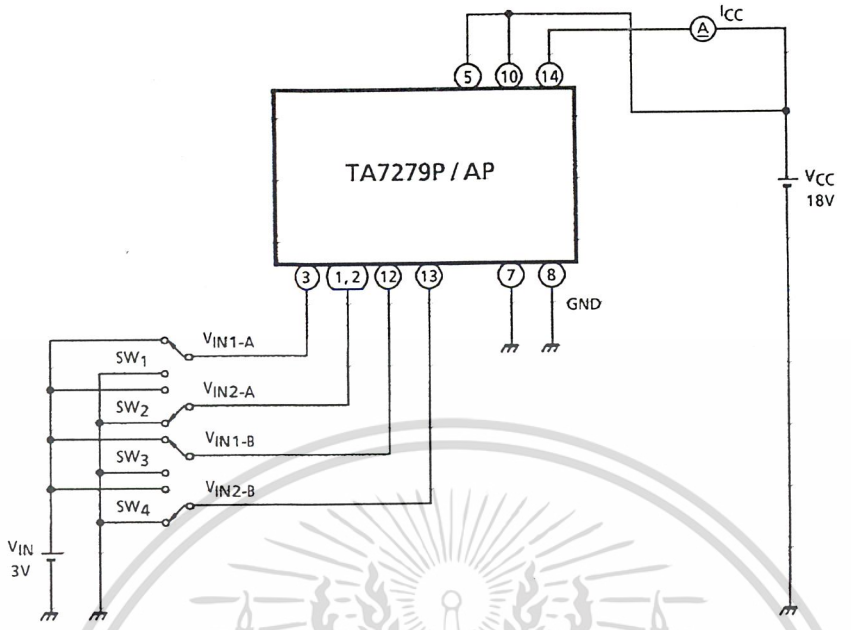
ELECTRICAL CHARACTERISTICS (Ta = 25°C)

CHARACTERISTIC		SYMBOL	TEST CIR-CUIT	TEST CONDITION	MIN	TYP.	MAX	UNIT
Supply Current		I_{CC1}	1	V_{CC} = 18 V, Output Off, Stop mode	14	28	41	mA
		I_{CC2}	1	V_{CC} = 18 V, Output Off, CW / CCW mode	10	29	38	
		I_{CC3}	1	V_{CC} = 18 V, Output Off, Brake mode	8	20	35	
Input Operating Voltage	1 (High)	$V_{IN\ (H)}$	—	T_j = 25°C	3.0	—	V_{CC}	V
	2 (Low)	$V_{IN\ (L)}$	—	T_j = 25°C	—	—	0.8	
Input Current		I_{IN}	2	Sink, V_{IN} = 3 V	—	3	10	μA
Output Saturation Voltage	Upper	V_{SATU-1}	3	I_O = 0.1 A, V_{CC} = V_S = 18 V	—	—	1.1	V
	Lower	V_{SATL-1}	3	I_O = 0.1 A, V_{CC} = V_S = 18 V	—	—	1.0	
	Upper	V_{SATU-2}	3	I_O = 1.0 A, V_{CC} = V_S = 18 V	—	1.2	1.5	
	Lower	V_{SATL-2}	3	I_O = 1.0 A, V_{CC} = V_S = 18 V	—	1.05	1.4	
Leakage Current	Upper	I_{LU}	—	V_S = 25 V	—	—	50	μA
	Lower	I_{LL}	—	V_S = 25 V	—	—	50	
Diode Forward Drop	Upper	V_{FU}	4	I_F = 1 A	—	2.0	—	V
	Lower	V_{FL}	4	I_F = 1 A	—	1.3	—	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

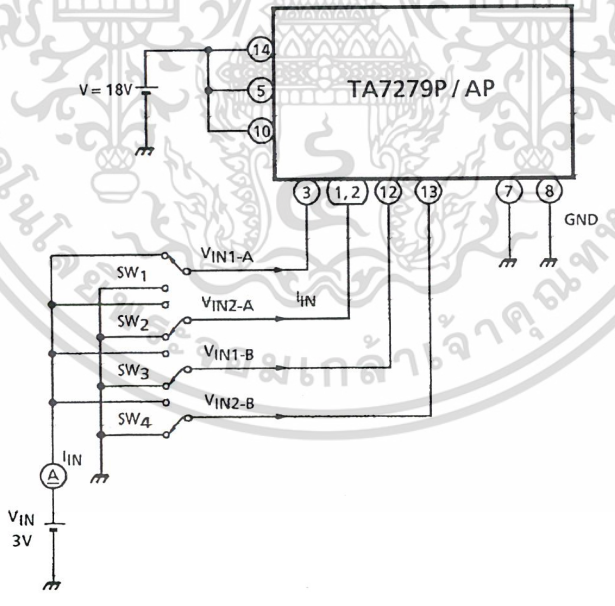
TEST CIRCUIT 1.

$I_{CC1, 2, 3}$



TEST CIRCUIT 2.

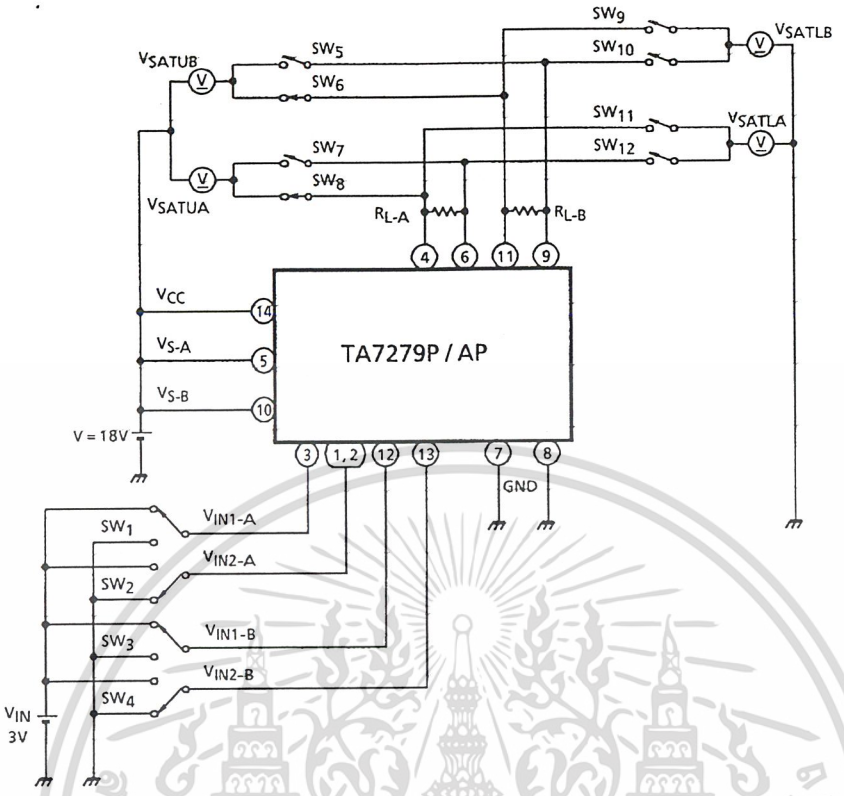
$I_{IN (H), (L)}$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

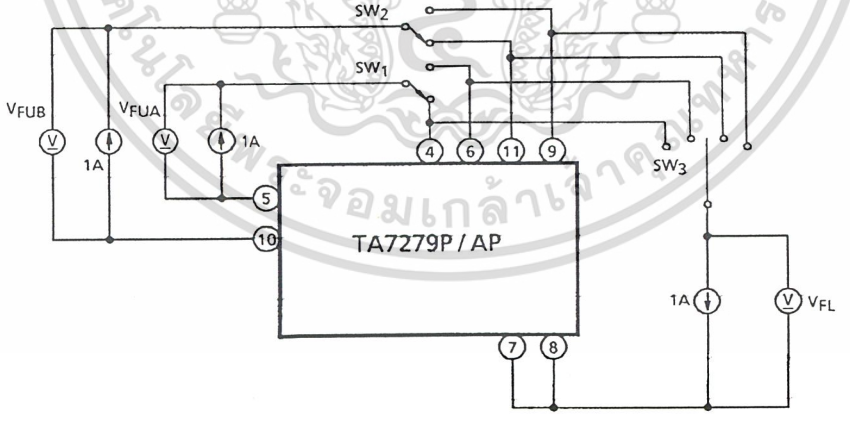
TEST CIRCUIT 3.

$V_{SATU-1, 2}$ / $V_{SATL-1, 2}$

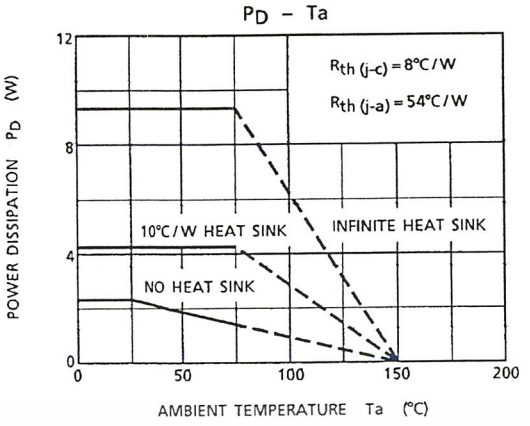


TEST CIRCUIT 4.

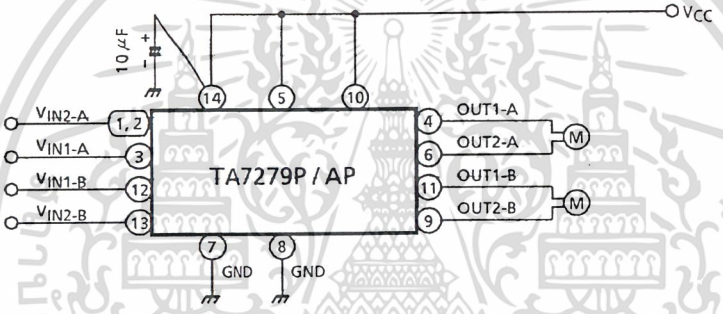
$V_{FU, L}$



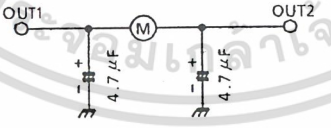
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



APPLICATION CIRCUIT



Problems may result if a capacitor is inserted in parallel to the motor as a measure against noise.
If measures against noise are necessary, connect capacitors as shown in the diagram below.
A larger bypass capacitor between V_{CC} and GND is effective against noise and other problems.
(A capacitance higher than 100 μF is recommended.)



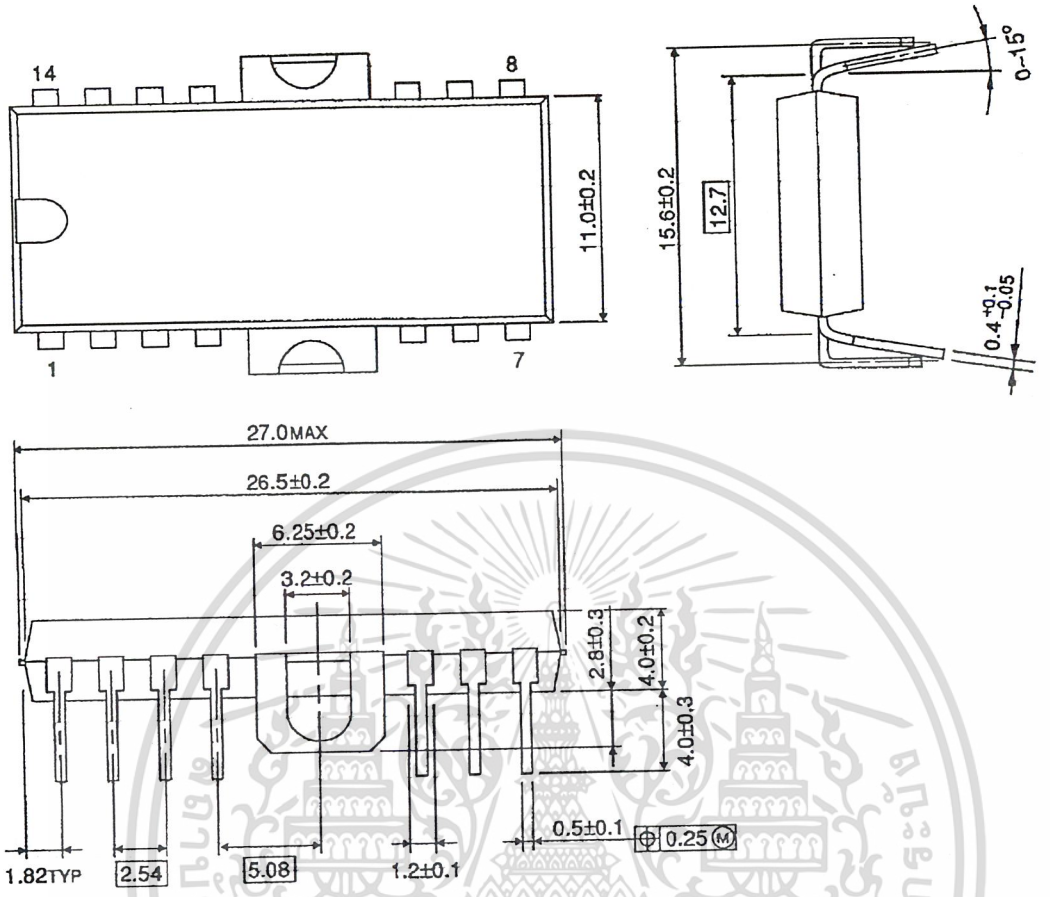
Note: Utmost care is necessary in the design of the output line, V_S and GND line since IC may be destroyed due to short-circuit between outputs, air contamination fault, or fault by improper grounding.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสาร

OUTLINE DRAWING

HDIP14-P-500-2.54A

Unit: mm



Weight: 3.00 g (Typ.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้