บทที่ 3

การเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอก

3.1 หลักการการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอก

ไมโครคอมพิวเตอร์ PC ที่ใช้กันทั่วไปในปัจจุบัน ไม่ว่าสำนักงานหรือที่บ้าน นอกเหนือจากการใช้งานด้านซอฟต์แวร์แล้ว ยังสามารถเชื่อมต่อกับอุปกรณ์ภายนอก (External instrument) ต่าง ๆ เพื่อวัดปริมาณทางกายภาพ และส่งผลกลับในการควบคุมปริมาณทางกายภาพที่แวดล้อมตัวเรา อาจจะเป็นภายในบ้าน ที่ทำงาน หรือห้องทดลองก็ได้ ปริมาณทางกายภาพเหล่านี้ ได้แก่ อุณหภูมิ ความชื้น ความเข้มแสง โมเมนต์การเปิด – ปิดประตู เป็นต้น

คอมพิวเตอร์ มีเส้นทางที่ใช้ในการเชื่อมต่อกับเครื่องมือภายนอก โดยมีการเรียกข้อมูลเข้ามา และต้องทำการควบคุมก็ทำการส่งข้อมูลออกไป โดยสั่งผ่านอุปกรณ์ที่ใช้ในการเชื่อมต่อ คือ Interface Card ET – PCI8255

3.2 อุปกรณ์ที่ใช้ในการเชื่อมต่อ

ET – PCI8255 คือการ์ดต่อขยายระบบคอมพิวเตอร์ PC เพื่อทำหน้าที่เป็นอุปกรณ์รับส่งสัญญาณดิจิตอลกับอุปกรณ์ภายนอก เนื่องจากปัจจุบัน ผู้ผลิตคอมพิวเตอร์ส่วนมากได้ยกเลิกระบบบัสแบบ ISA กับ Mainboard รุ่นใหม่ บริษัท ETT ได้มองเห็นความต้องการในการนำคอมพิวเตอร์ PC ไปใช้ในงานควบคุมยังมีอยู่ จึงได้พัฒนา ET – PCI8255 ซึ่งเป็น Card I / O แบบ PCI Bus สามารถนำมาทดแทน Card I / O แบบ ISA Bus ได้

รูปที่ 3.1 Interface Card ET – PCI8255

3.2.1 การติดตั้ง ET – PCI8255 กับเครื่องคอมพิวเตอร์

การติดตั้ง ET – PCI8255 กับเครื่องคอมพิวเตอร์ มีขั้นตอนดังนี้

1.) รันโปรแกรม PCI32.EXE จาก แผ่น CDROM ที่ให้มากับ ET – PCI8255 CARD เพื่อหา ADDRESS ที่ว่างสำหรับ ET – PCI8255 CARD ผู้ที่ต้องทำการเลือกตำแหน่ง ADDRESS ของ ET – PCI8255 ไม่ให้ตรงกับอุปกรณ์อื่น ๆ ที่อยู่ในระบบของ PC แอดเดรสที่แนะนำให้ใช้งานคือ F300H ถ้าโปรแกรมสำหรับตรวจสอบ PCI BASE Address นี้มีปัญหาไม่สามารถรันโปรแกรมนี้กับเครื่องของท่านได้ ให้ข้ามขั้นตอนที่ 1 นี้ไปยังหัวข้อ การใช้งานโปรแกรม PCITREE หน้า 5 (เสร็จแล้วกลับมาต่อขั้นตอนที่ 2) โปรมแกรมจะแสดงข้อมูลของ PCI ทั้งหมด ดังรูป

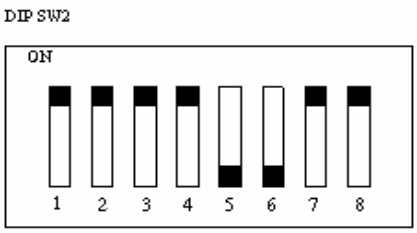
ตรวจสอบส่วนของ I / O ว่าไม่ซ้ำกับ Address ของ ET – PCI8255 ที่จะเลือก โดยทั่วไปแอดเดรสตั้งแต่ 000 H จนถึง 3FFH จะสงวนไว้กับระบบไม่ควรจะใช้งาน และตั้งแต่ 3FFH จนถึง FFFFH สามารถใช้เป็นแอดเดรสของ PCI ได้ แต่ทั้งนี้ต้องดู IO BAR ของ PCI CARD อื่น ๆ ด้วยว่าไม่ซ้ำตำแหน่งกันและ BASE ADDRESS ที่เลือก ควรจะห่างจาก BASE ADDRESS ของ CARD อย่างน้อย 04H เพื่อป้องกันข้อผิดพลาด

2.)หลังจากตรวจสอบ BASE ADDRESS ของระบบแล้ว ในกรณีที่ F300H ไม่ว่าง หรือต้องการเปลี่ยน ADDRESS ของ CARD ผู้ใช้งานต้องทำการเลือกดิพสวิตซ์ใหม่

แอดเดรสที่แนะนำ

DIP SW ON = 1 OFF = 0

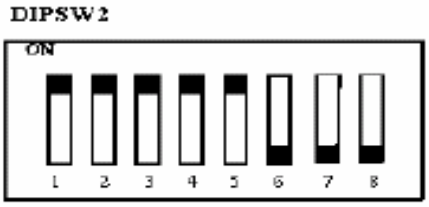
2.1 F300 HEX



1 1 1 1 0 0 1 1

F | 3

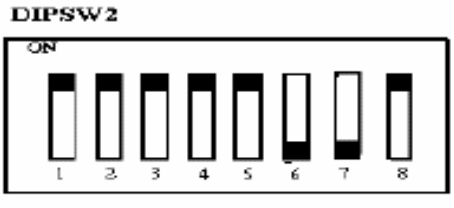
2.2 F400 HEX



1 1 1 1 1 0 0 0

F | 8

2.3 F500 HEX



1 1 1 1 1 0 0 1

F | 9

3.) สำหรับเครื่องคอมพิวเตอร์รุ่นใหม่ ๆ บางรุ่น จะมีฟังชั่นตรวจสอบ SLOT PCI ที่ว่างอยู่ และจะไม่ส่งสัญญาณ CLOCK ไปยัง SLOT ที่ว่างอยู่นั้น ผู้ใช้ต้อง Disable ฟังก์ชั่นนี้ มิฉะนั้นจะไม่สามารถใช้งาน ET – PCI16IO ได้

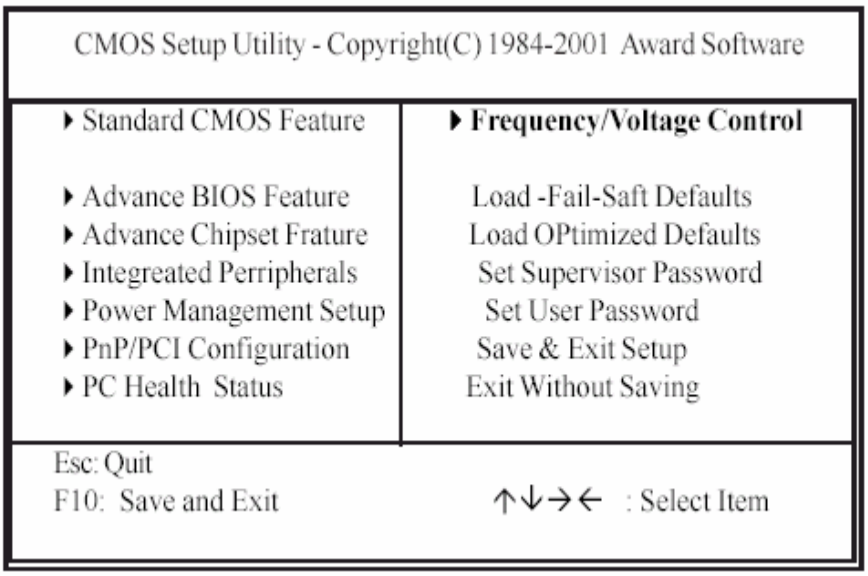
3.1 เข้าโหมด SETUP BIOS กดคีย์ DEL ขณะ BOOT เครื่อง (และอาจใช้คีย์อื่น ๆ ก็ได้ในการเข้า SETUP BIOS ในเครื่องบางรุ่น)

3.2 เข้า MENU CLOCK / FREQUEMCY CONTROL หรือ Auto Detect DIMM / PCI clock

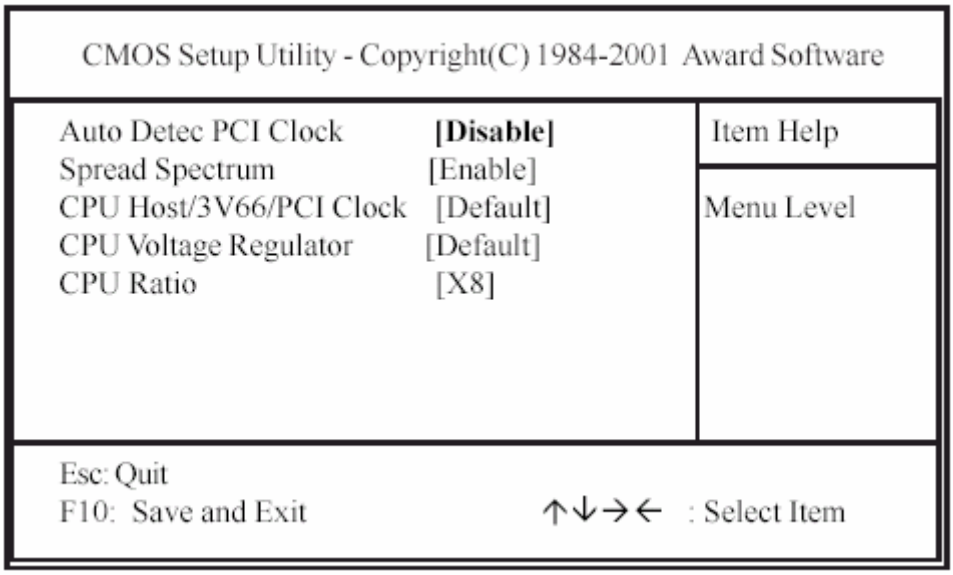
3.3 ทำการ Disable ฟังก์ชั่น Empty slot detection หรือ Auto Detect DIMM / PCI clock

3.4 SAVE ค่า BIOS ที่แก้ไขแล้ว และ ออกจาก BIOS SETUP (Save CMOS exit) สำหรับการ SETUP BIOS จะแตกต่างกันขึ้นอยู่กับ บริษัทผู้ผลิต และรุ่นของ BIOS

สำหรับคอมพิวเตอร์รุ่นใดที่ไม่มีฟังก์ชั่นนี้ให้ข้ามขั้นตอนในข้อนี้ไปได้



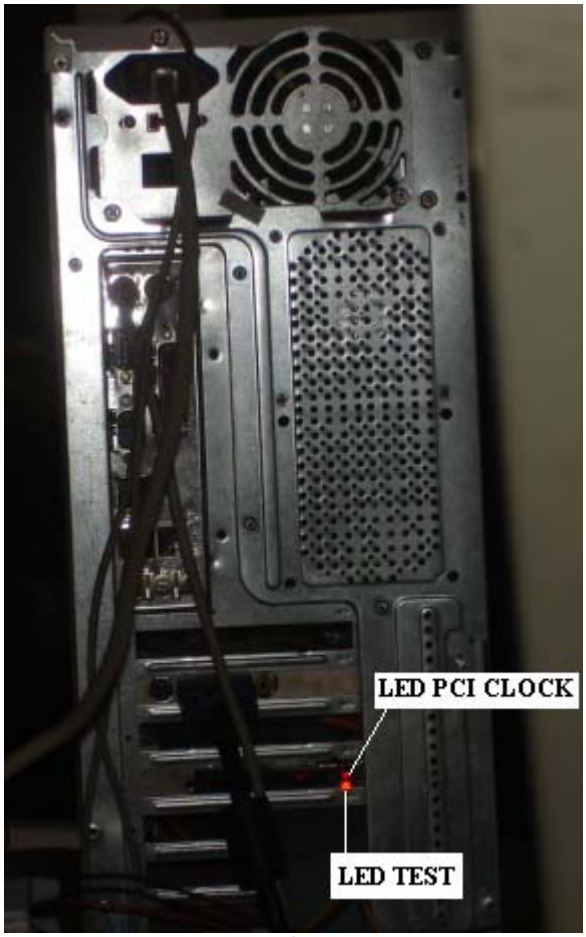
ตัวอย่าง Bios ของ บริษัท Award



เลือกให้ฟังก์ชั่น Auto Detect PCI Clock เป็น Disable

4.) Shut Down Windows และ ปิดสวิตซ์เครื่องคอมพิวเตอร์ ถอดสายไฟออกให้เรียบร้อย เปิดฝาเครื่องคอมพิวเตอร์ หากมีการใช้สายแพร์ให้ทำการต่อสายให้เรียบร้อยก่อนเสียบการ์ดลงสล็อต ตรวจดูความเรียบร้อยว่าการ์ดเข้าได้สนิทดีกับสล็อต

5.) เปิดสวิตซ์เครื่องคอมพิวเตอร์และตรวจสอบ LED ทั้งสองดวงบนการ์ดควรจะติด



รูปที่ 3.2 แสดงตำแหน่ง LED เมื่อมองจากด้านหลังคอมพิวเตอร์

ถ้า LED PCI CLOCK ไม่ติดให้ทำการตรวจสอบการติดตั้ง ข้อที่ 3 อีกครั้ง

6) ยึดการ์ดเข้ากับเครื่องคอมพิวเตอร์ด้วยน๊อต เสร็จสิ้นการติดตั้ง

7) LED TEST ต่อเข้ากับ PORT A0 ของ 8255#1 หลังจากติดตั้งโปรแกรมแล้ว สามารถใช้ LED TEST ทดสอบการทำงานของการ์ดได้

3.2.2 การติดตั้งและใช้งานโปรแกรม PCI TREE (สำหรับผู้ใช้โปรแกรม PCI32 ไม่ได้)

1. ตรวจสอบไฟล์ด้านล่างว่าครบถ้วนและอยู่ที่ Folder เดียวกันกับโปรแกรม PCITREE.EXE

(ต้องทำการตั้งค่าใน Option ของ Windows Explorer ให้เป็น View All File หน้า 20)

1.1 gbhlp.dll

1.2 mfc42d.dll

1.3 msvcrt.dll

1.4 msvcrtd.dll

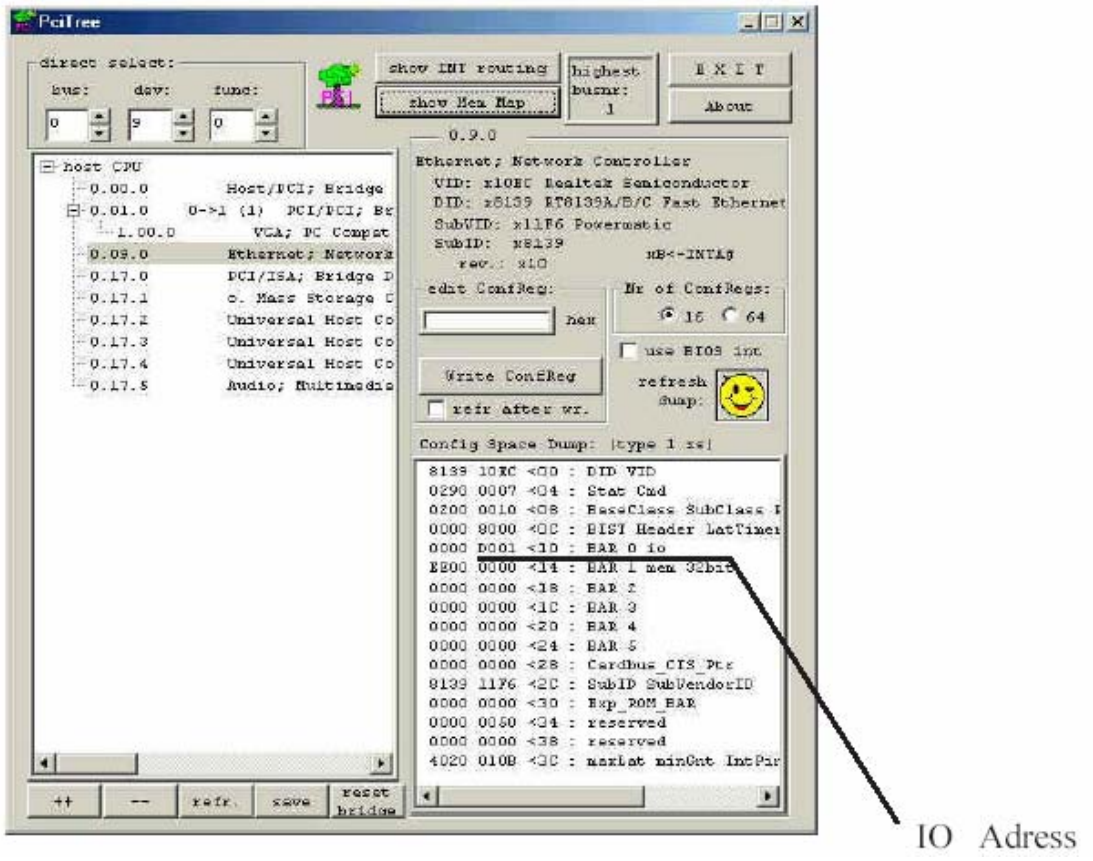
1.5 hlp.vxd

1.6 (pcidevs.txt) if present VIDs and DIDs are taken from this list

2. ทำการก๊อปปี้ไฟล์ hlp.sys ดังนี้

Win NT / 2000 : hlp.sys must be copied to \ WINNT \ SYSTEM32 \ DRIVERS directory

3. หลังจากการนั้น รันโปรแกรม PCITREE



รูปที่ 3.3 การทำงานของโปรแกรม

4. ทำการจดบันทึก ค่า PCI I / O Address ทั้งหมด โดยคลิกเม้าส์ไปที่แต่ละบรรทัดของ Dialog ด้านขวามือ หากบรรทัดใดมีเครื่องหมายบวกให้คลิกที่เครื่องหมายบวกด้วย จากนั้นสังเกตข้อมูลของ Dialog ด้านขวามือจะเปลี่ยนไปตามข้อมูลของแต่ละ PCI Device หมายเหตุ สำหรับแต่

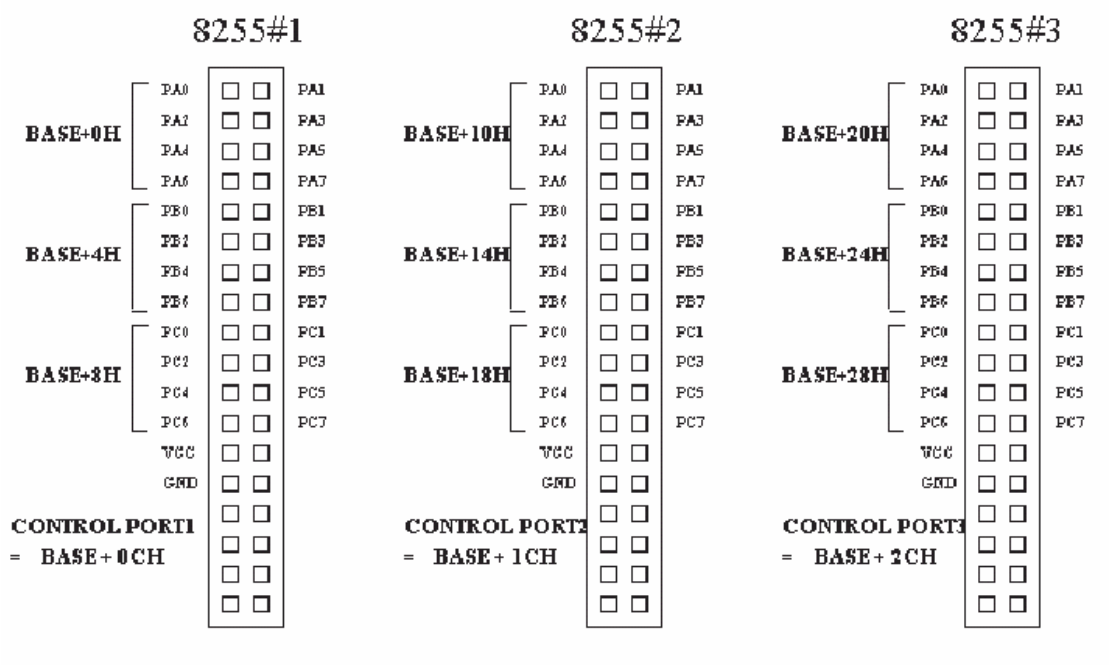
ละ PCI Device ไม่จำเป็นต้องมี IO Address ทุกตัวเสมอไป และ PCI TREE ไม่สามารถมองเห็นการ์ด ET – PCI8255 ได้

5. เลือกแอดเดรสที่เหมาะสม โดยทั่วไป แอดเดรสตั้งแต่ 00 00 H จนถึง 03 FFH จะสงวนไว้สำหรับระบบไม่ควรจะใช้งานและ ตั้งแต่ 03 FF H ขึ้นไป สามารถใช้เป็นแอดเดรสของ PCI ได้ แต่ทั้งนี้ต้องดู IO BAR ของ PCI CARD อื่น ๆ ด้วย ว่าไม่ซ้ำตำแหน่งกันและ BASE ADDRESS ที่เลือก ควรจะห่างจาก BASE ADDRESS ของ CARD อย่างน้อย 04H เพื่อป้องกันข้อผิดพลาด

6. ไม่แนะนำให้ใช้งานฟังก์ชันอื่น ๆ ของโปรมแกรมนี้ เพราะอาจจะทำความเสียหายให้กับระบบคอมพิวเตอร์

3.2.3 การใช้งาน 8255

8255 เป็น IC ประเภท CMOS Programmable Peripheral Interface ก่อนใช้งานจะต้องเลือกโปรแกรมเลือกให้ 8255 เป็นอินพุทหรือเอ้าพุทก่อนทุกครั้งหลังจาก เปิด switch POWER การเลือกค่าคือการส่ง Control Word ไปยัง Control Port หมายเลข พอร์ตของ ET – PCI8255 มีทั้งหมด 9 พอร์ต หลังจากดิพสวิตซ์เลือก Base Address แล้ว นำ Base Address บวกกับ ค่าคงที่ตามภาพด้านล่างพอร์ต ก็จะได้หมายเลขพอร์ต



ตัวอย่าง เมื่อเลือก Base Address เป็น F300 หมายเลขพอร์ตของ 8255#1 พอร์ต C คือ F308H

Control Port ของ 8255#1 คือ F30CH (F300H + 0CH = F30CH)

Control Port ของ 8255#2 คือ F31CH (F300H + 1CH = F31CH)

Control Port ของ 8255#3 คือ F32CH (F300H + 2CH = F32CH)

BASE ADRESS=F300H					
8255#1		8255#2		8255#3	
PORT A	F300h	PORT A	F310h	PORT A	F320h
PORT B	F304h	PORT B	F314h	PORT B	F324h
PORT C	F308h	PORT C	F318h	PORT C	F328h
CONTROL	F30Ch	CONTROL	F31Ch	CONTROL	F32Ch

BASE ADRESS=F200H					
8255#1		8255#2		8255#3	
PORT A	F200h	PORT A	F210h	PORT A	F220h
PORT B	F204h	PORT B	F214h	PORT B	F224h
PORT C	F208h	PORT C	F218h	PORT C	F228h
CONTROL	F20Ch	CONTROL	F21Ch	CONTROL	F22Ch

BASE ADRESS=F100H					
8255#1		8255#2		8255#3	
PORT A	F100h	PORT A	F110h	PORT A	F120h
PORT B	F104h	PORT B	F114h	PORT B	F124h
PORT C	F108h	PORT C	F118h	PORT C	F128h
CONTROL	F10Ch	CONTROL	F11Ch	CONTROL	F12Ch

BASE ADRESS user					
8255#1		8255#2		8255#3	
PORT A		PORT A		PORT A	
PORT B		PORT B		PORT B	
PORT C		PORT C		PORT C	
CONTROL		CONTROL		CONTROL	

ตารางที่ 3.1 แสดงหมายเลขพอร์ตเมื่อเลือก Base Address ค่าต่าง ๆ

เมื่อได้กำหนด Address ของแต่ละพอร์ตแล้วจะต้องส่งค่า Control Word ตามตารางหน้า 9 -10 เพื่อกำหนดแต่ละพอร์ตให้เป็นอินพุตหรือเอ้าพุท

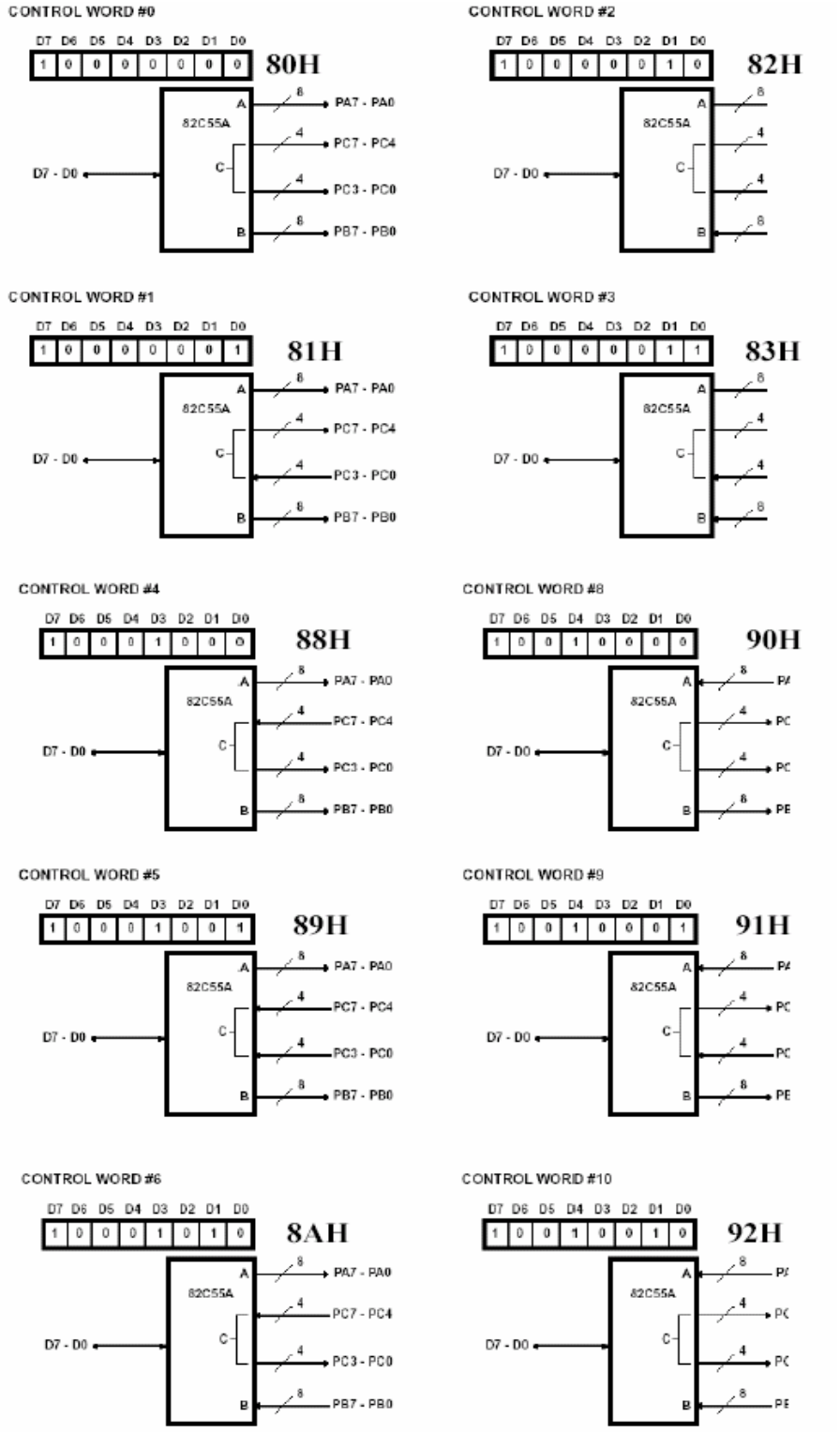
ตัวอย่างโปรแกรม

SetPortByte (HwCtrl, 0xF30C , 0x80) ; เป็นการกำหนดให้ Port A ,Port B , Port C ของ 8255#1 เป็นเอ้าพุท

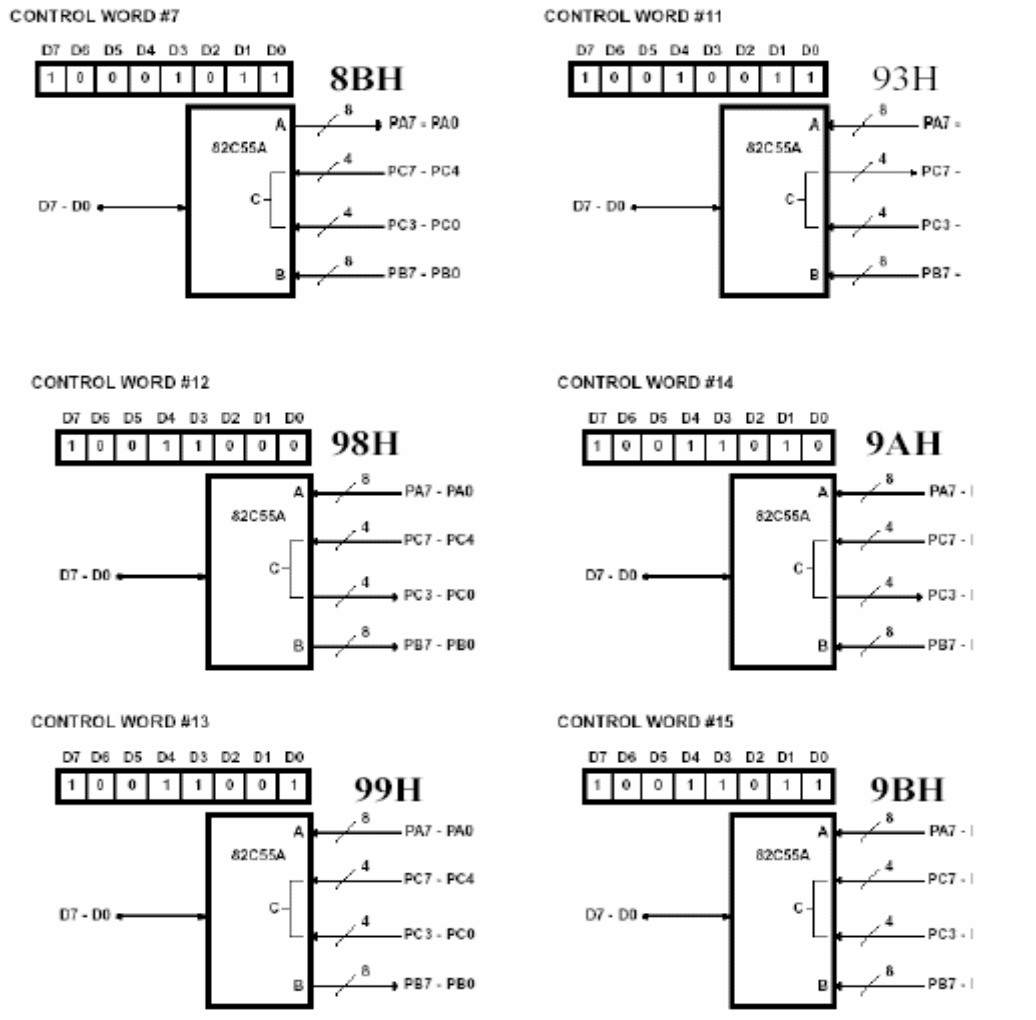
SetPortByte (HwCtrl, 0xF30C , 0x9B) ; เป็นการกำหนดให้ Port A ,Port B , Port C ของ 8255#1 เป็นอินพุท

จากตัวอย่าง 80H เป็น Control Word ที่กำหนดให้ทุกพอร์ตเป็นเอ้าพุต และ 9BH กำหนดให้ทุกพอร์ตเป็นอินพุท ส่วน F30C เป็น Control Port ของ 8255#1

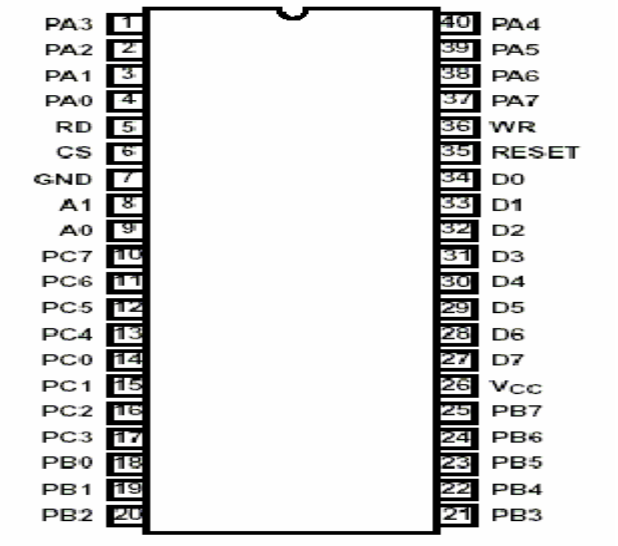
Mode 0 Configurations



ตารางที่ 3.2 แสดง Control Word ของ 8255



ตารางที่ 3.2 แสดง Control Word ของ 8255 (ต่อ)



ตำแหน่งขาของ 8255

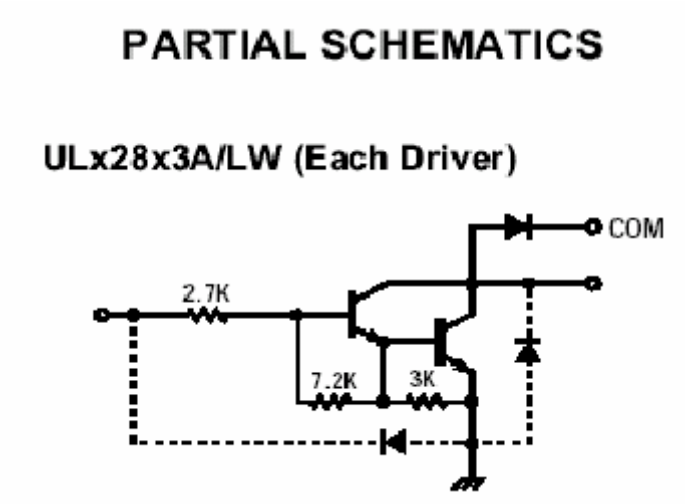
Pinouts

82C55A (DIP)

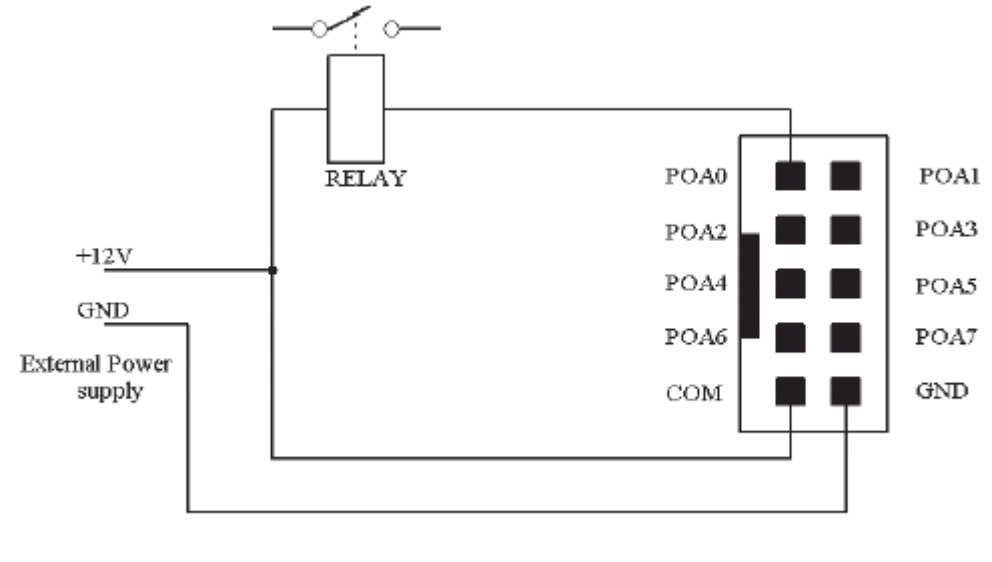
TOP VIEW

3.2.4 การใช้งาน OPEN COLLECTOR OUTPUT

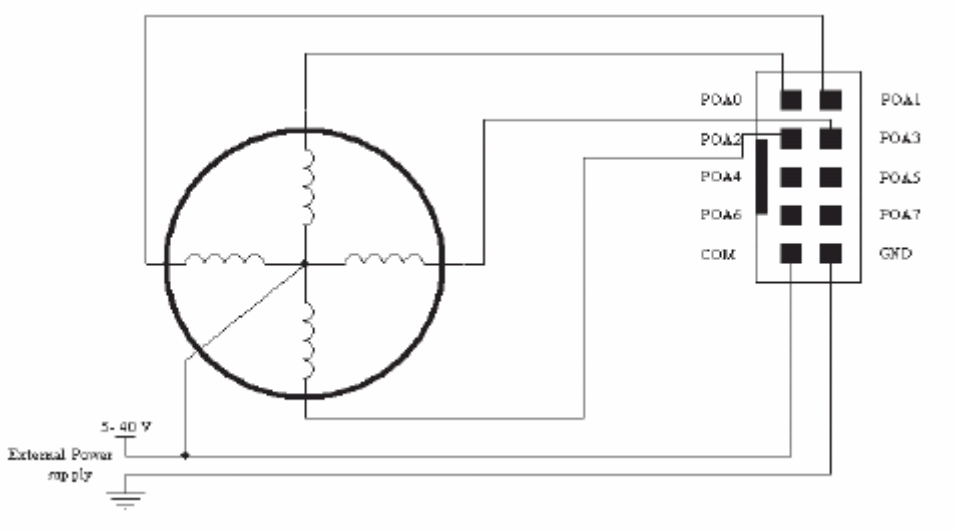
ET – PCI8255 ได้เพิ่มเติมวงจร OPEN COLLECTOR OUTPUT ไว้ในพอร์ตของ 8255#1 PORT A โดยใช้ IC ULN 2803 เป็นทางเลือกทางหนึ่งสำหรับผู้ที่ใช้งาน I / O ไม่เกิน 8 บิต สามารถต่อกับรีเลย์ได้ และโหลดอื่น ๆ ที่ใช้กระแสไม่เกิน 500 mA / CH โดยต่อทางขั้ว 10 Pin



รูปที่ 3.4 โครงสร้างภายในของ ULN 2803



รูปที่ 3.5 การต่อกับรีเลย์กับขั้ว 10 PIN



รูปที่ 3.6 การต่อกับ Stepping Motor กับขั้ว 10 PIN

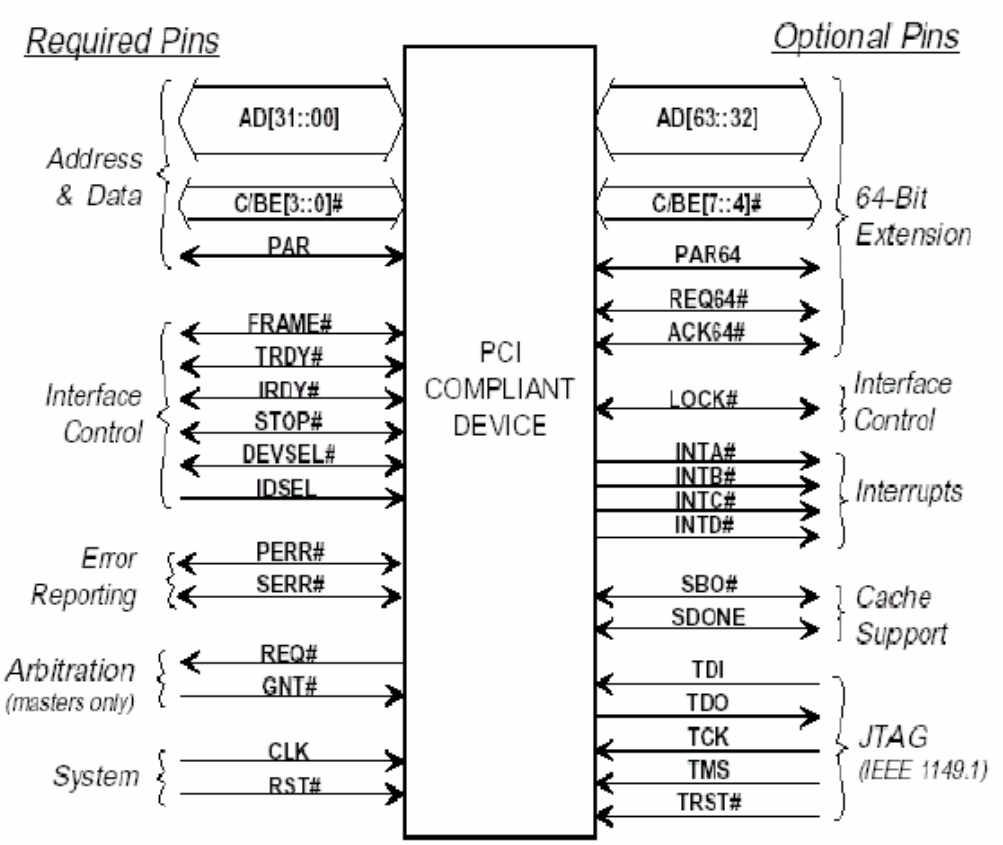
ULN 2803 สามารถรับกระแสสูงสุดได้ประมาณ 500 mA / Pin ดังนั้น Stepping Motor มีขนาดไม่เกิน 2 A โดยประมาณ

3.2.5 การทำงานของ ET – PCI8255

ET – PCI8255 การ์ด ประกอบด้วยสองส่วนคือ ส่วนของ IC8255 ซึ่งทำหน้าที่เป็นอินพุท เอ้าพุท ส่วนของวงจร ดีโค๊ดเดอร์ทำหน้าที่ถอดรหัสโปรโตคอลของ PCI BUS ได้แก่ ไอซี PLD XC95108 เมื่อเกิดขบวนการอ่านเขียนข้อมูล ไอซี PLD จะทำการตรวจลำดับและรูปแบบของสัญญาณ จากนั้นสร้าง Cycle และส่งสัญญาณ ที่จำเป็นในการอ่านเขียน 8255 ได้แก่ CS , A0 – A1 , RD WR ให้กับไอซี 8255 เป็นการถ่ายโอนข้อมูลระหว่าง PCI data bus กับ IO ภายนอก

3.2.6 ลักษณะโดยทั่วไปของ PCI BUS

PCI BUS เป็นสถาปัตยกรรมยุคใหม่ที่มีความสามารถถ่ายโอนข้อมูลได้มากมีความซับซ้อนมากกว่า ISA BUS แบบเก่า อุปกรณ์ที่จะนำมาเชื่อมต่อกับ PCI จะต้องได้มาตรฐานตามคุณสมบัติที่ได้กำหนดไว้แล้ว แต่การออกแบบอุปกรณ์ PCI ชนิด Target only สำหรับนำมาใช้ในงานควบคุมที่ไม่ซับซ้อน นับเป็นทางเลือกอีกอย่างหนึ่ง



รูปที่ 3.7 สัญญาณอินเตอร์เฟสของ PCI BUS

สัญญาณอินเตอร์เฟสของ PCI BUS สามารถแยกเป็นสองส่วนได้แก่ Option PIN และ Request PIN

Request PIN แบ่งตามหน้าที่ได้ดังนี้

1. SYSTEM PIN

1.1 CLOCK Clock ของระบบ PCI BUS การเปลี่ยนแปลงของสัญญาณต่าง ๆ จะมีผลที่ขอบขาขึ้นของ Clock

1.2 RESET สัญญาณ RESET

2. ADRESS and DATA PIN

2.1 AD0 – AD31 สัญญาณ Address และ Data ใช้ร่วมกันมีทั้งหมด 32 บิต

2.2 C / BE[0::3] สัญญาณ ชุดคำสั่ง ของบัส เช่น อ่าน เขียน

2.3 PAR สัญญาณ ตรวจสอบ Parity

3. Interface Control Pin

3.1 FRAME สัญญาณ สำหรับบ่งชี้ว่าเริ่มต้น และ สิ้นสุด ขบวนการถ่ายโอนข้อมูล

3.2 IRDY สัญญาณแสดงความพร้อมของ Master

3.3 TRDY สัญญาณแสดงความพร้อมของ Target

3.4 DEVSEL สัญญาณเลือกอุปกรณ์

Option PIN เป็นสัญญาณสำหรับอุปกรณ์ PCI ความสามารถสูง

3.2.7 หลักการถ่ายโอนข้อมูลของ PCI

หลักการถ่ายโอนข้อมูลของ PCI ประกอบด้วย 3 สัญญาณ คือ

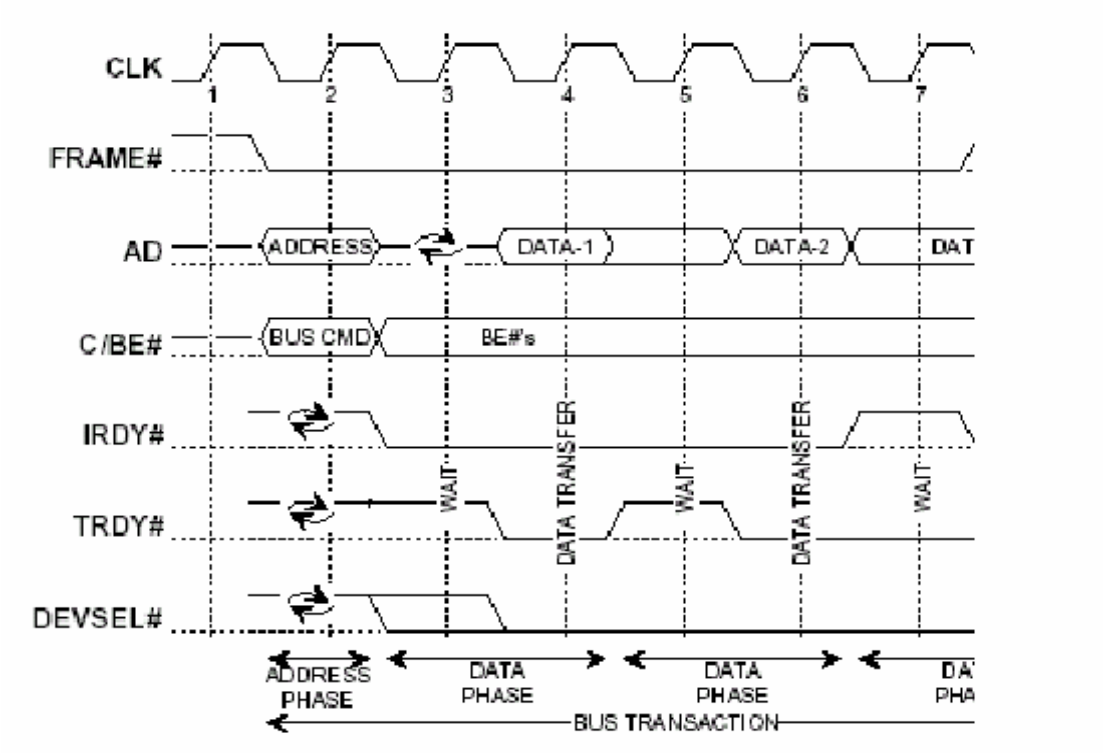
1. FRAME เป็นสัญญาณจาก Master สำหรับบ่งชี้ว่าเริ่มต้นแล สิ้นสุด ขบวนการถ่ายโอนข้อมูล

2. IRDY Innitiator Ready เป็นสัญญาณจาก Master สำหรับบ่งชี้ว่า พร้อมที่จะรับส่งข้อมูล

3. TRDY Target Ready เป็นสัญญาณจาก Target สำหรับบ่งชี้ว่า พร้อมที่จะรับส่งข้อมูล

เมื่อ Bus อยู่ในสภาวะ Idle สัญญาณ FRAME IRDY จะไม่ ACTIVE

สัญญาณ CLOCK ที่ขอบขาขึ้นลูกแรกเมื่อ สัญญาณ FRAME เริ่ม Active เรียกสภาวะนี้ว่า Address Phase และ สัญญาณ Address, Command จะถูกถ่ายโอนในสัญญาณ Clock ที่ขอบขาขึ้นนี้ สัญญาณที่ขอบขาขึ้น Clock เรียกสภาวะนี้ว่า Data Phase ในสภาวะนี้ สัญญาณ IRDY TRDY จะ Active สภาวะ Wait Cycle อาจเกิดขึ้นได้ เมื่อสัญญาณ IRDY หรือ TRDY ไม่ Active ขบวนการถ่ายโอนข้อมูลจะสิ้นสุดเมื่อ สัญญาณ FRAME ไม่ Active



รูปที่ 3.8 Basic Read & Write Operation

3.3 ภาษาที่ใช้ในการเขียนโปรแกรม

Visual Basic เป็นภาษาคอมพิวเตอร์ (Programming Language) โดยตัวภาษามีพื้นฐานมาจากภาษา Basic ซึ่งย่อมาจาก Beginner's All Purpose Symbolic Instruction ซึ่งหมายถึง "ชุดคำสั่งหรือภาษาคอมพิวเตอร์สำหรับผู้เริ่มต้น" ภาษา Basic มีจุดเด่นคือผู้ใช้ที่ไม่มีพื้นฐานเรื่องการเขียนโปรแกรมเลยก็สามารถเรียนรู้และนำไปใช้งานได้อย่างง่ายดายและรวดเร็ว เมื่อเทียบกับการเรียนภาษาคอมพิวเตอร์อื่น ๆ เช่น ภาษาซี, ปาสคาล เป็นต้น

3.4 การโปรแกรมเพื่อติดต่อกับ ET – PCI8255

ผู้ใช้งานสามารถสร้าง Application หรือโปรแกรมใด ๆ ที่มีฟังก์ชันสำหรับอินเอ้าพอร์ท ติดต่อกับ ET – PCI8255 ได้โดยตรง แต่ปัจจุบันโปรแกรมที่สร้าง Application สำหรับรันบน Windows ส่วนหนึ่งไม่มีมีฟังก์ชันสำหรับอินเอ้าพอร์ท และถึงแม้ว่าบางโปรแกรมจะมีฟังก์ชันสำหรับอินเอ้าพอร์ท ก็ไม่สามารถรัน Application ที่สร้างขึ้นนั้นกับระบบปฏิบัติการรุ่นใหม่ เช่น Windows NT, Windows 2000, Windows XP เนื่องจากข้อจำกัดที่ว่า ระบบปฏิบัติการรุ่นใหม่ ไม่อนุญาตให้ใช้

คำสั่งอินเอ้าพอร์ทได้โดยตรงจาก User Mode แต่สามารถใช้คำสั่งอินเอ้าพอร์ทได้โดยผ่าน Driver Mode เท่านั้น ทั้งนี้ การพัฒนาโปรแกรมในส่วนของ Driver Mode มีความยุงยากซับซ้อน

ดังนั้นทางเลือกอย่างอื่นที่ง่ายกว่าสำหรับผู้ใช้คือ การเรียกใช้ Library เหล่านั้นจะถูกพัฒนาจากภาษา C และชุด DDK (Driver Development KIT) จากโปรแกรมเมอร์ที่มีความชำนาญและข้อดีอีกอย่างหนึ่งของการเรียกใช้ Library (DLL) คือผู้ใช้สามารถใช้โปรแกรมภาษาที่ตัวเองถนัดเรียกใช้ DLL Library เหล่านั้นได้ สำหรับ Library ที่ให้มาพร้อมกับ ET – PCI8255 นี้เป็น Library ตัวหนึ่งที่ใช้งานได้ดีพอสมควร หรือผู้ใช้จะจัดหา Library ตัวอื่น ๆ มาทดลองใช้งานกับ ET – PCI8255 ก็ไม่เป็นปัญหาแต่ประการใด

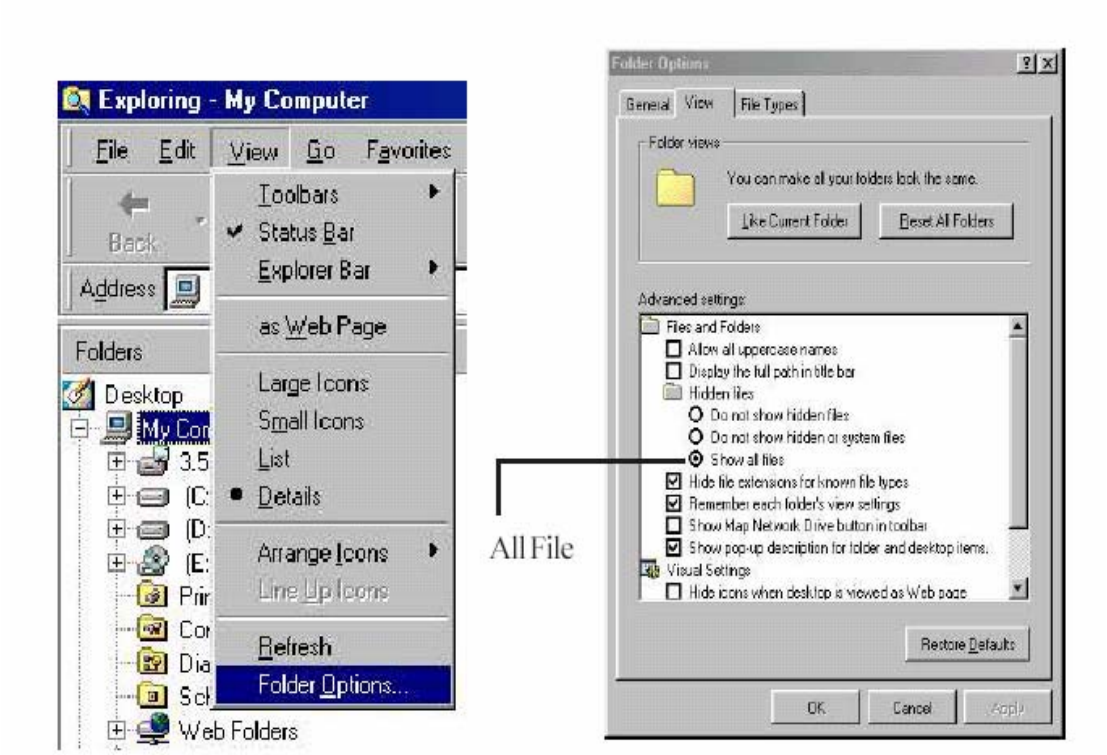
3.4.1 การติดตั้งไดรเวอร์และการใช้งาน Library

1. Library ที่มีอยู่บน CDROM มีทั้งหมด 3 ไฟล์ เนื่องจากทั้ง 3 ไฟล์เป็น System File ผู้ใช้ต้องทำการตั้งค่าให้ Windows Exploror มองเห็นไฟล์ System จึงจะสามารถ Copy ไฟล์จากแผ่น CDROM ได้

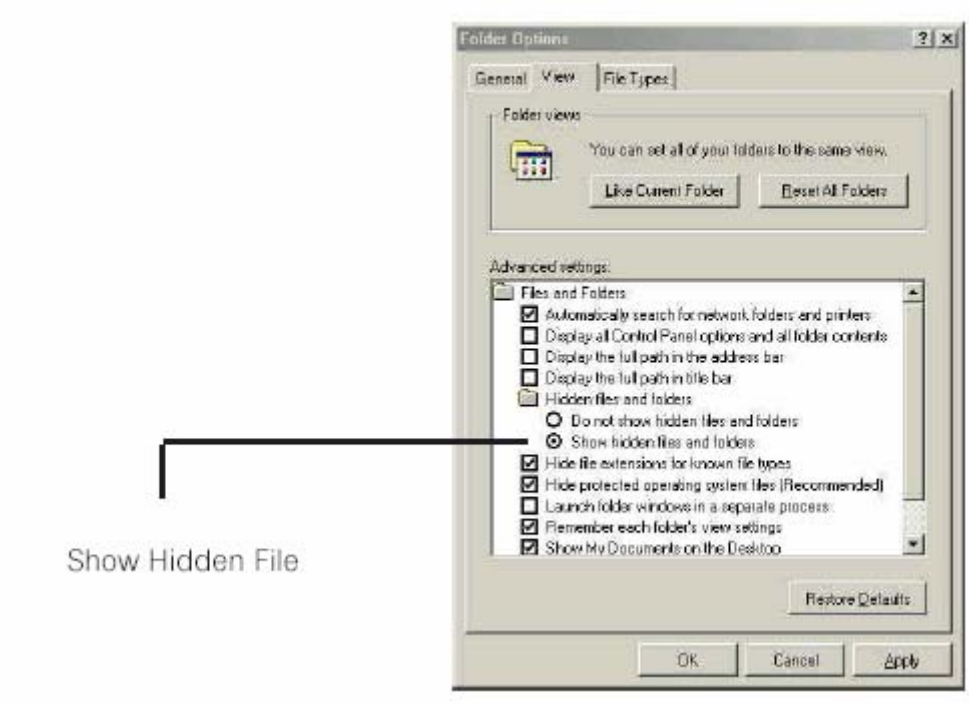
Klibdrv.SYS ไฟล์สำหรับระบบปฏิบัติการ

Klibdrv.VXD ไฟล์สำหรับระบบปฏิบัติการ

Klibdrv.DLL ไฟล์สำหรับ Compiler ของโปรแกรมตัวอย่าง



รูปที่ 3.9 การ SET ค่า ให้มองเห็น System File สำหรับ windows 95, 98



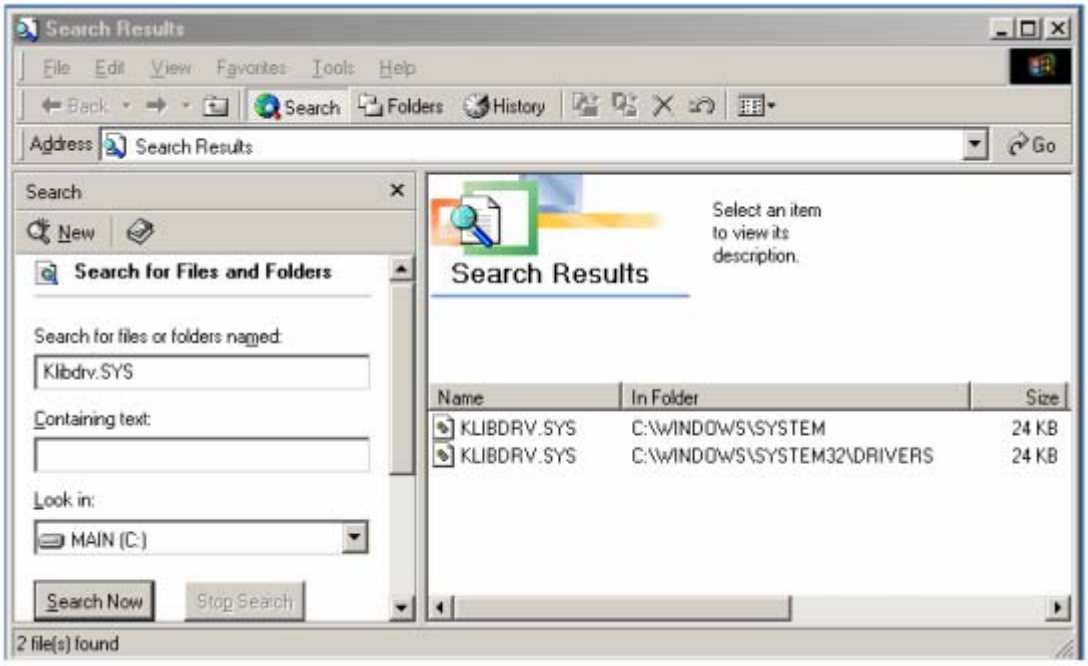
รูปที่ 3.10 การ SET ค่า ให้มองเห็น System File สำหรับ windows me

2. ทำการ Copy ไฟล์ลง System Directory ซึ่งกำหนดไว้ตอนติดตั้ง Windows

สำหรับ Windows 95, Windows 98, Windows Me

ทำการ Copy ไฟล์ Klibdrv.VXD, Klibdrv.SYS, Klibdrv.DLL ไปที่ System Directory ในกรณีติดตั้งไว้ที่ Driver C C:\Windows \System\ และ C:\Windows\System32\Driver\

3. ทดสอบว่าการ Copy File ได้ถูกต้อง โดยการ Search File Name Klibdrv.SYS หรือ Klibdrv.VXD

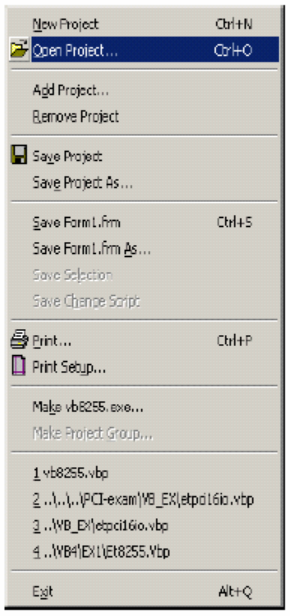


การใช้งานตัวอย่างการโปรแกรมของ Visual Basic Version 6.0

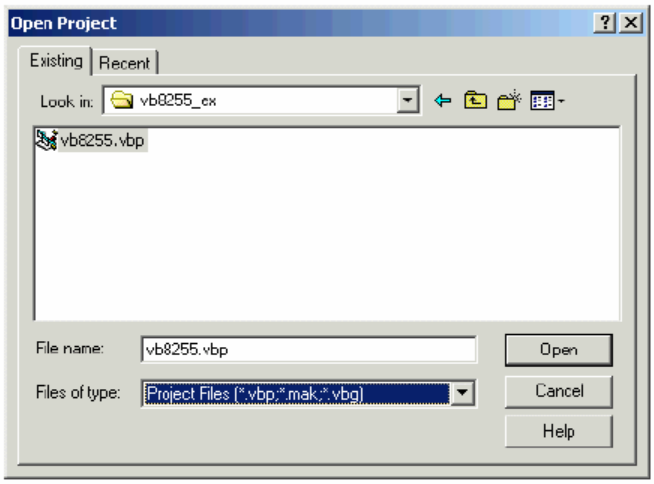
ก่อน copy ตัวอย่างการโปรแกรมจากแผ่น CDROM จะต้องตั้งค่าใน Windows Explorer ให้เป็น Show All File

Name	Size	Type	Modified
Form1.frm	16 KB	Visual Basic Form File	2/5/2546 11:25
KlibDrv.dll	60 KB	Application Extension	3/11/2543 4:13
KLIBDRV.sys	24 KB	System file	3/11/2543 4:13
KLIBDRV.VXD	18 KB	Virtual device driver	5/11/2543 7:23
KlibHWT.bas	10 KB	Visual Basic Module	24/10/2543 6:10
vb8255.exe	40 KB	Application	8/5/2546 14:26
vb8255.vbp	1 KB	Visual Basic Project	2/5/2546 11:25
vb8255.vbw	1 KB	Visual Basic Project Workspace	2/5/2546 11:25

1. open project



2. OPEN vb8255.vbp



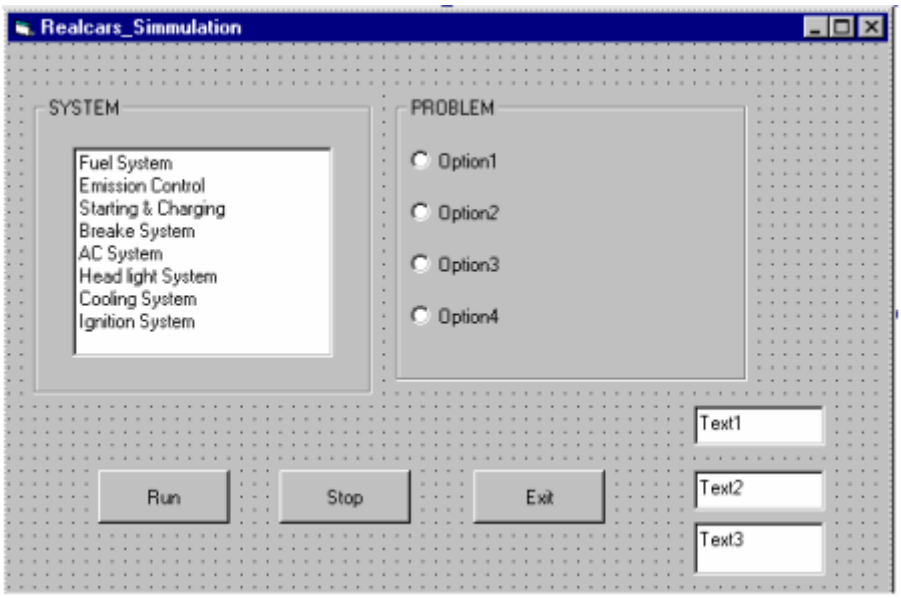
3.4 การออกแบบโปรแกรม

ขั้นตอนที่ 1 ศึกษาความต้องการของผู้ใช้ (user requirement) และกำหนด วัตถุประสงค์ของโปรแกรม (Objective) ในที่นี้เราต้องการที่จะออกแบบโปรแกรมให้สามารถควบคุมการทำงานของรถยนต์ ซึ่งเราต้องทราบรายละเอียดของรถยนต์ ระบบต่าง ๆ รวมทั้งวงจรไฟฟ้าที่ควบ จากรูปข้างบน ทำให้เราทราบว่าเรามีรีเลย์ 40 ตัว ที่ทำหน้าที่ตัดต่อระบบการทำงานต่าง ๆ ของเครื่องยนต์ โดยที่รีเลย์แต่ละตัวมีการต่ออนุกรมและขนานกับรีเลย์ตัวอื่น แล้วจึงต่อเข้ากับอุปกรณ์ต่าง ๆ

ขั้นตอนที่ 2 ออกแบบหน้าจอของโปรแกรมที่เราต้องการ (prototype) พร้อมกับกำหนดคุณสมบัติและส่วนประกอบต่าง ๆ ของโปรแกรมโดยละเอียด (program specification) การออกแบบหน้าจอเราจะเลือกคอนโทรลต่าง ๆ ตามลักษณะการใช้งาน ตัวอย่าง เช่น หน้าจอโปรแกรมที่เราออกแบบ มีการเลือกใช้ Combo Box เนื่องจากเราต้องการทดสอบว่าเมื่อมีปัญหาแบบต่าง ๆเกิดขึ้นกับรถยนต์แล้วจะส่งผลอะไรบ้างต่อรถยนต์

ขั้นตอนที่ 3 เริ่มเขียนโปรแกรม (coding)

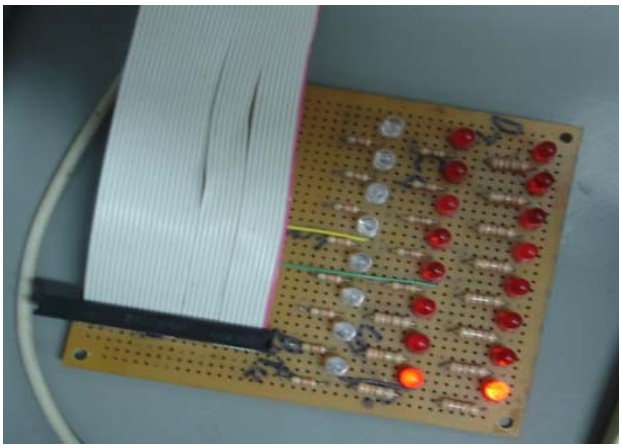
ขั้นตอนที่ 4 รวบรวมโปรแกรม แปลงให้อยู่ในรูปที่พร้อมใช้งาน (compile) และ ทดสอบการทำงานของโปรแกรม (testing) ซึ่งเราจะได้หน้าจอของโปรแกรมและ Coding ดังนี้



รูปที่ 3.11 หน้าจอโปรแกรม (ที่ออกแบบ)

จากรูปที่ จะเห็นว่าเรามีการแบ่ง หน้าจอเป็น 2 กลุ่ม คือ ใน Combo Box เป็นกลุ่มของระบบต่าง ๆ ในรถยนต์ ส่วนใน Frame เป็นการจัดกลุ่มปัญหาต่าง ๆ ไว้ด้วยกัน ซึ่งปัญหาเหล่านี้จะเป็นไปตามระบบต่าง ๆ ที่เราเลือก ส่วน Command Button ทั้ง 3 อันมีหน้าที่ดังนี้ Run เมื่อเรากด จะทำหน้าที่ประมวลผล Stop เมื่อเรากด จะทำหน้าที่หยุดการประมวลผล Exit เมื่อเรากด จะเป็นการออกจากโปรแกรม ส่วน Text Box 2 อัน จะแสดงตัวเลขกลุ่มของระบบและลำดับของปัญหา โดยปุ่มแต่ละปุ่มต้องเขียนโปรแกรมควบคุม ซึ่งเป็นโปรแกรมย่อย (Sub program) ของโปรแกรมหลัก (Main program)

ขั้นตอนที่ 6 ทดสอบโปรแกรมโดยใช้แผงวงจร LED TEST



รูปที่ 3.12 การดสอบโดยใช้แผง LED TEST