บทที่ 2
แนวความคิดของชุดฝึกทดลอง

2.1 โครงสร้างบทเรียน

ชุดฝึกทดลองในงานวิทยานิพนธ์นี้ จะประกอบด้วยการทดลองรวมทั้งสิ้น 12 การทดลองโดยมีรายละเอียดและวัตถุประสงค์ของแต่ละการทดลอง ดังตารางข้างล่างนี้

ตารางที่ 2.1 โครงสร้างการทดลอง

ลำดับ	เรื่อง	วัตถุประสงค์	รายละเอียด
1	การทำความคุ้นเคยกับชุดฝึกทดลอง	1) เพื่อเรียนรู้ส่วนประกอบต่างๆ ของชุดฝึกทดลองอย่างกว้าง ๆ 2) เพื่อเรียนรู้การใช้งานชุดฝึกทดลอง	- ทดลองรันโปรแกรมตัวอย่าง - การคอมไพล์และรันโปรแกรมง่าย ๆ
2	การเขียนโปรแกรมด้วยภาษา C	1) เพื่อเรียนรู้โครงสร้างและคำสั่งที่สำคัญ ๆ ของภาษา C 2) เพื่อเรียนรู้การเขียนโปรแกรมภาษา C และการทดสอบการแก้ไขโปรแกรม	- เขียนโปรแกรมตัวอย่าง - การรับ/ส่งข้อมูลกับผู้ใช้ทางคีย์แผงแป้นกดและจอภาพ - การเขียนฟังก์ชัน - การใช้งานตัวแปรพอยน์เตอร์ - การทดสอบ แก้ไข โปรแกรม
3	การรับ/ส่งข้อมูลดิจิทัล	1) เพื่อเรียนรู้การรับ/ส่งสัญญาณดิจิทัลผ่านทางพอร์ตของคอมพิวเตอร์ 2) เพื่อเรียนรู้การเขียนโปรแกรมตัวควบคุมเชิงผสม (combinational) 3) เพื่อเรียนรู้การเขียนโปรแกรมตัวควบคุมเชิงลำดับ (sequential)	- โปรแกรมตัวอย่าง แสดงให้เห็นถึงการใช้ฟังก์ชัน IN , OUT - การเขียนโปรแกรมตัวควบคุมเชิงผสม และเชิงลำดับ - การแก้การกระเด้งของแป้นกด - การทดสอบ IC

ตารางที่ 2.1 โครงสร้างการทดลอง(ต่อ)

ลำดับ	เรื่อง	วัตถุประสงค์	รายละเอียด
4	ตัวแปลงผัน D/A	1) เพื่อทดลองสร้างรูปคลื่นสัญญาณโดยใช้ตัวแปลงผัน D/A 2) เพื่อทดลองใช้คอมพิวเตอร์ในการควบคุมระดับแรงดัน	- สร้างรูปสัญญาณไซน์ - หาลักษณะการทำงานของตัวแปลงผัน D/A
5	ตัวแปลงผัน A/D	1) เพื่อพิจารณาคุณสมบัติต่างๆ ของตัวแปลงผัน A/D 2) เพื่อทดลองนำตัวแปลงผัน A/D ไปใช้ในการบันทึกรูปคลื่นสัญญาณ 3) เพื่อทดลองนำตัวแปลงผัน A/D ไปใช้ในงานสัญญาณอื่น ๆ	- การวัดอุณหภูมิ วัดแสง วัดเสียง - การบันทึกสัญญาณเสียง และการเล่นกลับ
6	ตัวจับเวลา/ตัวนับ (Timer/Counter)	1) เพื่อเรียนรู้วิธีการใช้งาน ตัวจับเวลา/ตัวนับ 2) เพื่อเรียนรู้การประยุกต์ใช้ตัวจับเวลา/ตัวนับ ในการวัดคาบสัญญาณ การกำเนิดสัญญาณที่มีคาบเวลาที่ต้องการ การวัดความถี่ของสัญญาณ	- การสร้างพัลส์ - การวัดคาบสัญญาณ - การนับพัลส์ / วัดความถี่ของสัญญาณ - เครื่องวัดค่าความจุไฟฟ้าของตัวเก็บประจุ
7	การอินเตอร์รัปต์	1) เพื่อเรียนรู้การใช้งานการอินเตอร์รัปต์ บน IBM PC 2) เพื่อเรียนรู้การประยุกต์ใช้งานการอินเตอร์รัปต์ในการควบคุม	- การตั้งค่าและการคืนค่าในตารางอินเตอร์รัปต์เวกเตอร์ - การเขียนรูทีนใช้งานอินเตอร์รัปต์ - การสุ่มสัญญาณแอนะลอกด้วยคาบเวลาคงที่
8	การสื่อสารข้อมูล (data communication)	1) เพื่อเรียนรู้การสื่อสารข้อมูลแบบอนุกรม และแบบขนาน 2) เพื่อทดลองเขียนโปรแกรมเพื่อสื่อสารข้อมูลระหว่างเครื่องคอมพิวเตอร์ 2 เครื่อง ด้วยวิธีอนุกรมและขนาน 3) เพื่อทดลองการควบคุมผ่านทางพอร์ตอนุกรมและพอร์ต	- การรับ/ส่งข้อมูลแบบอนุกรม - การควบคุมสถานะของสัญญาณต่าง ๆ - การรับ/ส่งข้อมูลแบบขนาน

ตารางที่ 2.1 โครงสร้างการทดลอง(ต่อ)

ลำดับ	เรื่อง	วัตถุประสงค์	รายละเอียด
		ขนานของเครื่องคอมพิวเตอร์	
9	การประยุกต์ใช้งาน 1 (Applications1)	1) เพื่อทดลองประยุกต์ใช้งานการควบคุม โดยใช้ไมโครคอมพิวเตอร์ในงานต่าง ๆ 2) เพื่อทดลองเขียนโปรแกรมควบคุมสเตปปิงมอเตอร์	- การควบคุมสเตปปิงมอเตอร์
10	การประยุกต์ใช้งาน 2 (Applications2)	- เพื่อทดลองเขียนโปรแกรมควบคุมตัวแสดงผลแบบเมทริกซ์ของ LED	- การควบคุมการแสดงผลแบบเมทริกซ์ของ LED
11	ระบบควบคุมแบบดิจิทัล	1) เพื่อทดลองการควบคุมระบบ ด้วยตัวควบคุมแบบดิจิทัล 2) เพื่อทดลองการควบคุมระบบแบบเปิดและแบบปิด โดยใช้ไมโครคอมพิวเตอร์	- การวัดความเร็วมอเตอร์ - การควบคุมความเร็วมอเตอร์
12	โครงงาน	- เพื่อทดลองนำความรู้ที่ได้เรียนมา มาประยุกต์ใช้งานในการควบคุม โดยใช้ไมโครคอมพิวเตอร์	- มีการกำหนดตัวอย่างหัวข้อ เช่น > การวัดระยะทางโดยใช้คลื่นอัลตราโซนิก > ออสซิลโลสโคปแบบดิจิทัล > ตัวอ่านรหัสบาร์โค้ด

2.2 วัตถุประสงค์ของการเรียนรู้

1) เพื่อเรียนรู้การใช้ไมโครคอมพิวเตอร์ในการควบคุมอินเตอร์เฟซ
2) เพื่อเรียนรู้การเขียนซอฟต์แวร์ในการควบคุมอินเตอร์เฟซ
3) เพื่อเรียนรู้การเชื่อมต่อฮาร์ดแวร์ในการควบคุมอินเตอร์เฟซ
4) เพื่อเรียนรู้หลักการทำงานพื้นฐานของอุปกรณ์อินเตอร์เฟซ

2.3 แผนการสอน

ชุดการทดลองประกอบด้วยการทดลอง 12 ครั้ง ใช้ระยะเวลาเรียนรู้ 1 ภาคการศึกษา คิดเป็น 12-15 ครั้ง แต่ละครั้งใช้เวลา 3-4 ชั่วโมง โดย

- การทดลองครั้งแรก มีวัตถุประสงค์ คือ ให้เกิดความคุ้นเคยกับชุดทดลอง เพื่อเตรียมสู่การใช้งานในชุดทดลองอื่น ๆ ต่อไป

- การทดลองครั้งที่ 2 จะฝึกทักษะ การเรียนรู้การเขียนโปรแกรมด้วยภาษา C ซึ่งเป็นภาษาที่ใช้ในการทดลอง ขอบข่ายของการเรียนรู้ เกี่ยวกับโครงสร้างของภาษา และขั้นตอนการเขียนโปรแกรม การใช้โปรแกรมบรรณาธิการ(editor) การคอมไพล์(compile)โปรแกรม การดำเนิน(execute)โปรแกรม การทำ single step และ break point รวมถึง ส่วนประกอบต่าง ๆ ของคอมไพเลอร์ Turbo C

- การทดลองครั้งที่ 3 เริ่มต้นด้วยการให้ผู้เรียนทดลองเขียนโปรแกรมตัวอย่าง เพื่อให้เห็นโครงสร้างของโปรแกรมควบคุมการรับ/ส่งข้อมูลดิจิทัล หลังจากนั้นจะเป็นการเรียนรู้วิธีการจำลองวงจรเชิงผสมโดยใช้คอมพิวเตอร์ ซึ่งสามารถทำได้หลายวิธี และเรียนรู้วิธีการจำลองวงจรเชิงลำดับ โดยทดลองเขียนโปรแกรมถอดรหัสการกดปุ่มเพื่อควบคุมการเปิด-ปิดเครื่องใช้ไฟฟ้า ผู้เรียนจะได้เรียนรู้วิธีกแก้การกระเด้งของแป้นกด และสุดท้ายจะเป็นการทดสอบ IC แบบดิจิทัล

- การทดลองครั้งที่ 4 จะเป็นการเรียนรู้การใช้งานตัวแปลงผัน D/A โดยจะพิจารณาคุณสมบัติบางประการของอุปกรณ์นี้ และทดลองปรับแต่งวงจรปรับภาวะสัญญาณ หลังจากนั้นจะทดลองนำตัวแปลงผัน D/A มาประยุกต์ใช้ในการสร้างรูปคลื่นสัญญาณ ทำให้คอมพิวเตอร์เป็นเสมือนฟังก์ชันเจนเนอเรเตอร์จำลอง

- การทดลองครั้งที่ 5 จะเป็นการเรียนรู้การใช้งานตัวแปลงผันA/D โดยจะพิจารณาคุณสมบัติบางประการของอุปกรณ์นี้ และทดลองใช้มันในการวัดแสง เสียง อุณหภูมิ ผู้เรียนจะได้เรียนรู้การแสดงผลในโมดกราฟฟิกของ Turbo C ด้วย ซึ่งเป็นการพัฒนาวิธีการแสดงผลในงานควบคุมให้มีความสวยงาม และน่าสนใจยิ่งขึ้น หลังจากนั้นจะทดลองประยุกต์ใช้ตัวแปลงผัน A/D ในการบันทึกเสียงและเล่นกลับ

- การทดลองครั้งที่ 6 เป็นการเรียนรู้เรื่อง ตัวจับเวลา/ตัวนับ (Timer/Counter) ซึ่งมีความสำคัญอย่างมากในงานควบคุม ผู้เรียนจะได้เรียนรู้วิธีการโปรแกรมชิป เบอร์ 8253 ซึ่งนิยมใช้งานกันมาก และทดลองประยุกต์ใช้ตัวจับเวลา/ตัวนับ ในการสร้างพัลส์ การวัดคาบสัญญาณ การวัดความถี่สัญญาณ และสุดท้ายจะทดลองนำตัวจับเวลา/ตัวนับ มาใช้ในการวัดค่าความจุไฟฟ้าของตัวเก็บประจุ

- การทดลองครั้งที่ 7 เป็นการเรียนรู้เรื่องการอินเตอร์รัปต์ การใช้งานอินเตอร์รัปต์บนเครื่องไมโครคอมพิวเตอร์จะมีรายละเอียดเพิ่มเติมค่อนข้างมากเมื่อเทียบกับไมโครโปรเซสเซอร์บอร์ดเดี่ยว จึงได้ให้โปรแกรมตัวอย่างไว้เพื่อให้ผู้เรียนศึกษาด้วย ในการทดลองนี้จะเป็นการทดลองนำมอดูลต่างๆ คือมอดูลตัวแปลงผัน A/D และมอดูลตัวจับเวลา/ตัวนับ (T/C) มาใช้ในงานร่วมกันด้วย

- การทดลองครั้งที่ 8 เป็นการเรียนรู้เรื่องการสื่อสารข้อมูล ซึ่งแบ่งออกเป็น 2 ส่วน คือ การสื่อสารข้อมูลแบบขนาน และการสื่อสารข้อมูลแบบอนุกรม ผู้เรียนจะได้เรียนรู้การใช้งานระบบ RS-232 ซึ่งเป็นที่นิยมใช้ในระบบไมโครโปรเซสเซอร์ทั่วไป โดยเฉพาะในการติดต่อกับเทอร์มินอล และจะได้ทดลองประยุกต์ใช้ในการส่งจดหมายโต้ตอบทางไปรษณีย์อิเล็กทรอนิกส์ (E-mail)

- การทดลองครั้งที่ 9 จะเป็นการทดลองควบคุมสเตปปิงมอเตอร์ ซึ่งมีวิธีการควบคุมการกระตุ้นหลายแบบและมีคุณสมบัติที่แตกต่างกัน

- การทดลองครั้งที่ 10 เป็นการทดลองเขียนโปรแกรมเพื่อควบคุมการแสดงผลเป็นตัวอักษรบนเมทริกซ์ LED และทดลองทำตัวอักษรวิ่ง ผู้เรียนจะได้เรียนรู้เทคนิคในการเขียนโปรแกรมบางประการ

- การทดลองครั้งที่ 11 เป็นการนำไมโครคอมพิวเตอร์มาใช้ในการควบคุมระบบควบคุมแบบปิด โดยจะมีการควบคุมความเร็วของมอเตอร์กระแสตรง ผู้เรียนยังจะได้เรียนรู้เทคนิคการวัดความเร็วมอเตอร์ การประยุกต์ใช้งาน ตัวจับเวลา/ตัวนับ และการควบคุมแบบ PMW (pulse-width modulation)

- การทดลองครั้งที่ 12 จะให้ผู้เรียนได้ทดลองออกแบบระบบควบคุมขึ้นมาเอง โดยนำความรู้ที่เรียนแล้ว มาประยุกต์ใช้

การทดลองแต่ละการทดลองจะใช้เวลา 1 ครั้ง ยกเว้นการทดลองที่ 12 ซึ่งจะใช้เวลา 2 - 3 ครั้ง ทั้งนี้ขึ้นอยู่กับความยากง่ายของโครงงานที่ทำ โดยในครั้งแรกอาจใช้เวลาในการออกแบบและลงมือในการทำครั้งที่ 2 - 3

การทดลองเหล่านี้สามารถนำไปใช้ประกอบการสอนวิชาไมโครคอมพิวเตอร์ โดยให้เป็นวิชาปฏิบัติการของวิชาบรรยายเรื่องการออกแบบระบบที่ใช้ไมโครโปรเซสเซอร์ (microprocessor-based system design) ผู้สอนสามารถประยุกต์และปรับปรุงเปลี่ยนแปลงการทดลองเพื่อให้เข้ากับเนื้อหาวิชาบรรยายได้ตามความเหมาะสม และเนื่องจากชุดฝึกทดลองนี้ได้ออกแบบมาเพื่อให้สามารถขยายระบบได้ง่ายโดยผ่านระบบบัส ฉะนั้นผู้สอนอาจจะออกแบบบอร์ดเพิ่มเติมเพื่อใช้ในการทดลองที่ออกแบบขึ้นได้

2.4 แนวความคิดของระบบ

ชุดทดลองทั้งหมดจะแบ่งเป็น 2 ส่วน คือ ส่วนของแผงวงจรหลัก ซึ่งในที่นี้ขอเรียกว่า เมนบอร์ด และส่วนของมอดูล ส่วนของเมนบอร์ดเป็นส่วนที่ต่อเข้ากับเครื่อง IBM PC ทางพอร์ตขนานเพื่อทำหน้าที่ในการสร้างสัญญาณการเขียน/อ่านข้อมูล เพื่อใช้ในการติดต่อกับอุปกรณ์ภายนอกหรือมอดูล มีลักษณะของสัญญาณเช่นเดียวกับระบบไมโครโปรเซสเซอร์ทั่วไป คือ ประกอบด้วย บัสข้อมูล (data bus) แอดเดรสบัส (address bus) และ สัญญาณ READ, WRITE

ส่วนของมอดูลเป็นส่วนที่จะต่อเข้ากับเมนบอร์ด (บางมอดูลอาจต่อโดยตรงกับเครื่อง IBM PC หรือ ต่อเข้ากับมอดูลอื่นอีกต่อหนึ่ง) เพื่อใช้ทำการทดลอง โดยแต่ละการทดลองอาจจะใช้มอดูลเพียงหนึ่งมอดูล หรือใช้ 2, 3, 4 มอดูลเพื่อทำงานร่วมกัน ภายในมอดูลแต่ละชุดจะประกอบด้วยวงจร address decoder ซึ่งจะถอดรหัสแอดเดรสของตัวเองโดยไม่ซ้ำกับแอดเดรสของมอดูลอื่น ๆ และวงจรที่ทำหน้าที่เฉพาะของแต่ละมอดูล เช่น ตัวจับเวลา/ตัวนับ, ตัวแปลงผัน A/D, ตัวเฝ้าตรวจแบบที่ใช้แอลอีดี (LED monitor) เป็นต้น

2.4.1 เมนบอร์ด

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

เราจะใช้พอร์ตขนานของเครื่อง IBM PC เป็นช่องทางในการติดต่อ โดยนำสัญญาณของพอร์ตขนานมาสร้างเป็นสัญญาณการเขียน/อ่านข้อมูล ซึ่งประกอบด้วย

1) บัสข้อมูล (data bus)

บัสนี้เป็นแบบที่รับ/ส่งข้อมูลได้สองทิศทาง (bi-directional bus) ในที่นี้ ขนาดที่ใช้คือ 8 บิต บัสนี้ใช้ส่งผ่านข้อมูลจากอุปกรณ์ไปยังเครื่องคอมพิวเตอร์หรือจากเครื่องคอมพิวเตอร์มายังอุปกรณ์ หรือแม้แต่จากอุปกรณ์ไปยังอุปกรณ์ด้วยกัน ในขณะที่ไม่มีการเขียน/อ่านข้อมูล บัสนี้จะมีสถานะอิมพีแดนซ์สูง(high impedance หรือ Hi-Z)

2) บัสแอดเดรส (address bus)

บัสนี้เป็นทางผ่านของสัญญาณออกจากเครื่องคอมพิวเตอร์ที่ใช้ในการเลือกแอดเดรสของอุปกรณ์ที่เครื่องคอมพิวเตอร์ต้องการจะติดต่อด้วย โดยจะนำสัญญาณแอดเดรสนี้ไปถอดรหัส เป็นสัญญาณเลือกชิป อุปกรณ์แต่ละตัวจะมีแอดเดรสต่าง ๆ กัน บัสนี้มีขนาด 8 บิต จึงทำให้สามารถเลือกพอร์ตได้จำนวน $2^8 = 256$ พอร์ต

3) สัญญาณการอ่านข้อมูล [READ($\overline{RD}$)]

สัญญาณนี้เป็นสัญญาณการอ่านข้อมูลจากอุปกรณ์เข้าสู่เครื่องคอมพิวเตอร์ เป็นสัญญาณที่ส่งจากเครื่องคอมพิวเตอร์ไปยังอุปกรณ์ เพื่อบอกให้อุปกรณ์เตรียมข้อมูลที่จะส่งให้แก่เครื่องคอมพิวเตอร์มายังบัสข้อมูล และหลังจากนั้นเครื่องคอมพิวเตอร์จะมาอ่านข้อมูลไปจากบัสข้อมูล สัญญาณนี้จะมีสถานะแบบ active low คือเมื่อสัญญาณนี้มีสถานะเป็น Low จะเป็นการบอกให้อุปกรณ์ส่งข้อมูลมายังบัสข้อมูล

4) สัญญาณการเขียนข้อมูล [WRITE ($\overline{WR}$)]

สัญญาณนี้คือสัญญาณการเขียนข้อมูลจากเครื่องคอมพิวเตอร์ไปยังอุปกรณ์ เป็นสัญญาณที่ส่งจากเครื่องคอมพิวเตอร์ไปยังอุปกรณ์เพื่อบอกให้อุปกรณ์มาอ่านข้อมูลไปจากบัสข้อมูล ซึ่งคอมพิวเตอร์ได้เตรียมไว้ให้ เป็นสัญญาณ active

low คือ เมื่อสัญญาณนี้มีสถานะเป็น Low เครื่องคอมพิวเตอร์จะส่งข้อมูลมายังบัสข้อมูล หลังจากนั้นอุปกรณ์จะมาอ่านข้อมูลไปจากบัสข้อมูล

5) สัญญาณการร้องขออินเตอร์รัปต์ (interrupt request)

สัญญาณนี้คือสัญญาณที่อุปกรณ์ใช้ร้องขอการอินเตอร์รัปต์ไปยังเครื่องคอมพิวเตอร์ โดยมีลำดับความสำคัญเป็น หมายเลข 7 (Interrupt No.7 ($\overline{IRQ7}$)) สัญญาณนี้สามารถถูก disable ได้ โดยการสั่งงานผ่านทาง พอร์ตขนาน

6) สัญญาณกราวด์ (GND)

สัญาณนี้เป็นสัญญาณที่แสดงถึงระดับแรงดันกราวด์ของระบบ

สัญญาณเหล่านี้ จะมีลักษณะเหมือนกับสัญญาณบนระบบไมโครโปรเซสเซอร์ทั่ว ๆ ไป จึงสามารถนำไปต่อใช้งานกับ อุปกรณ์อื่น ๆ ได้ หรือผู้ใช้สามารถออกแบบมอดูลใช้งานเพิ่มเติมขึ้นเอง โดยใช้สัญญาณเหล่านี้ในการอินเตอร์เฟซ สัญญาณเหล่านี้ถูกสร้างขึ้นโดยการสั่งงานผ่านทางพอร์ตขนาน และใช้โปรแกรมสั่งงานร่วมกับอุปกรณ์ฮาร์ดแวร์บนเมนบอร์ด โปรแกรมสั่งงานจะถูกเขียนในลักษณะของฟังก์ชัน โดยอยู่ใน Library เพื่อให้ผู้ใช้สามารถเรียกใช้งานได้อย่างสะดวก สัญญาณเหล่านี้เป็นสัญญาณแบบอะซิงโครนัส(asynchronous) คือไม่มีสัญญาณนาฬิกาของระบบที่ให้จังหวะการทำงาน

2.4.1.1 ไดอะแกรมการจัดจังหวะเวลาของระบบบัส

จังหวะเวลาของระบบบัส ประกอบด้วยบัสไซเคิล(bus cycle) 2 บัสไซเคิล คือบัสไซเคิลการอ่านข้อมูล (read cycle) และ บัสไซเคิลการเขียนข้อมูล (write cycle) ซึ่งมีรายละเอียดดังต่อไปนี้

1) บัสไซเคิลการอ่านข้อมูล

บัสไซเคิลนี้เป็นขั้นตอนการอ่านข้อมูลจากอุปกรณ์ ซึ่งจะมีการทำงานดังนี้ คือ

1.1) เครื่อง IBM PC ส่งสัญญาณมายัง บัสแอดเดรส เพื่อเลือกอุปกรณ์ที่จะทำการติดต่อด้วย

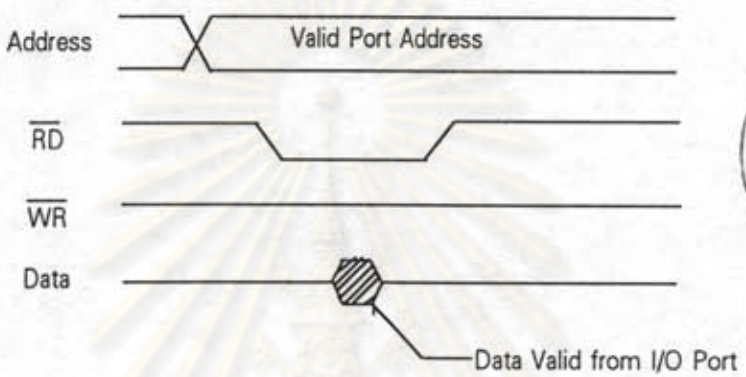
1.2) สัญญาณการอ่านข้อมูล ($\overline{RD}$) เปลี่ยนสถานะ เป็น low

1.3) อุปกรณ์ ส่งข้อมูลมายังบัสข้อมูล

1.4) เครื่อง IBM PC อ่านข้อมูลจากบัสข้อมูล

1.5) สัญญาณ $\overline{RD}$ เปลี่ยนสถานะ เป็น High

1.6) อุปกรณ์ปล่อยการยึดเกาะบัสข้อมูล บัสข้อมูลเปลี่ยนสถานะกลับเป็นอิมพีแดนซ์สูง (Hi-Z)



รูปที่ 2.1 แผนภาพการจัดจังหวะเวลาของบัสไซเคิลการอ่านข้อมูล

2) บัสไซเคิลการเขียนข้อมูล (WRITE Cycle)

บัสไซเคิลนี้เป็นขั้นตอนการเขียนข้อมูลไปยังอุปกรณ์ โดยมีการทำงานดังนี้ คือ

2.1) เครื่อง IBM PC ส่งสัญญาณมายังบัสแอดเดรส

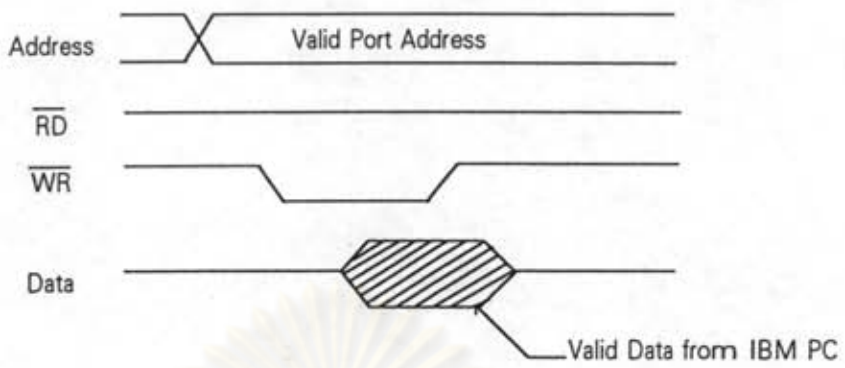
2.2) สัญญาณ $\overline{WR}$ เปลี่ยนสถานะเป็น Low

2.3) เครื่อง IBM PC ส่งข้อมูลมายังบัสข้อมูล

2.4) สัญญาณ $\overline{WR}$ เปลี่ยนสถานะ เป็น High

2.5) อุปกรณ์อ่านข้อมูลจากบัสข้อมูลในช่วงขอบขาขึ้นของสัญญาณ $\overline{WR}$

2.6) เครื่อง IBM PC ปล่อยการยึดเกาะบัสข้อมูล เปลี่ยนสถานะกลับเป็น Hi-Z



รูปที่ 2.2 การจัดจังหวะเวลาสำหรับบัสไซเคิลการเขียนข้อมูล

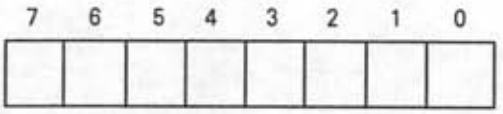
2.4.1.2 ทฤษฎีพอร์ตขนาน (Parallel Port Theory) [8,9,10]

พอร์ตขนานของเครื่อง IBM PC เป็นส่วนที่จะใช้ในการอินเตอร์เฟซกับชุดทดลอง ประกอบด้วยพอร์ตย่อย 3 พอร์ต ซึ่งอยู่ที่ตำแหน่งแอดเดรส 378h - 37Ah [สำหรับพอร์ตขนานหมายเลข 1 (LPT1)] หรือ 278h - 27Ah [สำหรับพอร์ตขนานหมายเลข 2 (LPT2)] หรือ 3BCh - 3BEh (สำหรับพอร์ตขนานบนบอร์ดโมโนโครม) พอร์ตขนานเป็นพอร์ตที่ต่อสัญญาณออกโดยใช้คอนเนกเตอร์แบบ DB 25 พอร์ตนี้บางทีเรียกว่า "พอร์ตเครื่องพิมพ์" เนื่องจากว่าส่วนใหญ่ใช้ในการติดต่อกับเครื่องพิมพ์ รายละเอียดของพอร์ตย่อยทั้งสามมีดังนี้

1) พอร์ต A (รูปที่ 2.3)

พอร์ตนี้เป็นพอร์ตสัญญาณขาออกขนาด 8 บิต ทุกบิตต่อเข้ากับขาของคอนเนกเตอร์ จึงทำให้สามารถส่งข้อมูลมาที่ขาของคอนเนกเตอร์ได้โดยตรง โดยใช้คำสั่งเขียนข้อมูลมาที่พอร์ตสัญญาณเข้า/ออก (พอร์ต I/O) พอร์ตนี้มีสถานะทางตรรกเป็นบวกทุกบิต คือเมื่อเขียนข้อมูล 1 มาที่บิตใด ขาของคอนเนกเตอร์ที่ตรงกับบิตนั้นจะมีระดับสัญญาณเป็น High และเมื่อเขียนข้อมูล 0 มาที่บิตใด ขาของคอนเนกเตอร์ที่สอดคล้องกันจะมีระดับสัญญาณเป็น Low (สัญญาณไม่ถูกคอมพลีเมนต์) ต่อไปในวิทยานิพนธ์เล่มนี้ เมื่อใช้คำว่า มีสถานะทางตรรกเป็นบวก ก็จะแสดงถึงความหมายดังกล่าว และถ้าใช้คำว่า มีสถานะทางตรรกเป็นลบ ก็จะหมายถึงว่าเมื่อเขียนข้อมูล 1 มาที่บิต สัญญาณที่ได้จะมีระดับสัญญาณเป็น Low และถ้าเขียนข้อมูล 0 มาที่บิต สัญญาณที่ได้มีระดับสัญญาณเป็น High เราสามารถอ่านสถานะของพอร์ตนี้เข้ามาได้โดยใช้คำสั่งการอ่านข้อมูลจากพอร์ต I/O ณ ตำแหน่งแอดเดรสเดียวกัน และเราสามารถใช้พอร์ตนี้ใน

การรับสัญญาณขาเข้าได้ โดยข้อมูลที่อ่านเข้ามาจะเป็นข้อมูลที่เกิดจากการ OR กันระหว่างข้อมูลที่เอาต์ออกไป กับสัญญาณขาเข้า แต่โดยทั่วไปไม่นิยมใช้พอร์ตนี้ในการรับสัญญาณขาเข้า

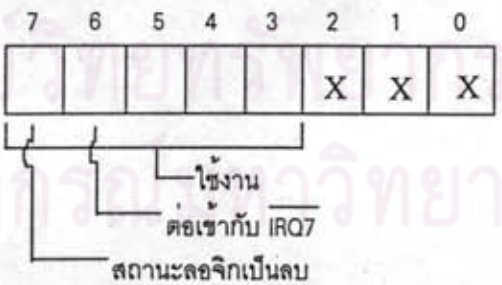


พอร์ตแอดเดรส: บนบอร์ดโมโนโครม = 3BCh
พอร์ตขนานหมายเลข 1 = 378h
พอร์ตขนานหมายเลข 2 = 278h

รูปที่ 2.3 พอร์ต A ของชุดพอร์ตขนานของ IBM PC

2) พอร์ต B (รูปที่ 2.4)

พอร์ตนี้เป็นพอร์ตสัญญาณขาเข้า ขนาด 5 บิต อยู่ที่ ตำแหน่งบิต 3 ถึงบิต 6 ของพอร์ต I/O แอดเดรส 379h มีสถานะทางตรรกเป็นบวกทุกบิต ยกเว้นบิต 7 ที่มีสถานะทางตรรกเป็นลบ บิต 6 ต่อเข้ากับสัญญาณอินเตอร์รัปต์หมายเลข 7 ของเครื่อง IBM PC แบบกลับสถานะ กล่าวคือ เมื่อต้องการให้เกิดการอินเตอร์รัปต์ จะต้องให้มีสถานะเป็น Low บิต 0 - 2 มีสถานะทางตรรกไม่แน่นอน

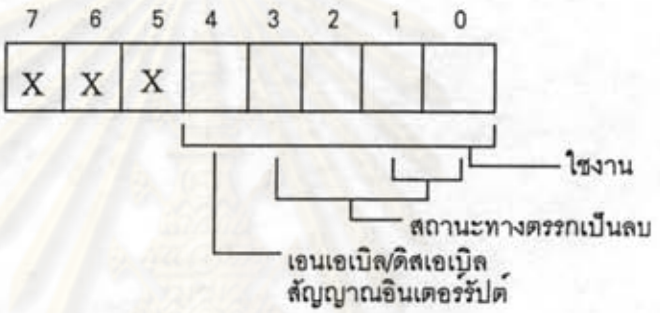


พอร์ตแอดเดรส: บนบอร์ดโมโนโครม = 3BDh
พอร์ตขนานหมายเลข 1 = 379h
พอร์ตขนานหมายเลข 2 = 279h

รูปที่ 2.4 พอร์ต B ของชุดพอร์ตขนานของ IBM PC

3) พอร์ต C

พอร์ตนี้เป็นพอร์ตสัญญาณเข้า/ออกขนาด 4 บิต อยู่ที่ตำแหน่งบิตที่ 0 - 4 ของพอร์ต I/O แอดเดรส 37Ah ตำแหน่งบิตที่ 0, 1, 3 มีสถานะทางตรรกเป็นลบ ส่วนบิตที่ 2 มีสถานะ ทางตรรกเป็นบวก วงจรสัญญาณขาออกเป็นแบบคอลเลกเตอร์เปิด ฉะนั้น เมื่อมีการป้อนสัญญาณขาเข้า ข้อมูลที่อ่านได้จากพอร์ตจะเป็นข้อมูลที่เกิดจากการ AND กัน ระหว่างสัญญาณขาเข้าและสัญญาณขาออกในขณะนั้น โดยสัญญาณขาออกจะไม่มีการเปลี่ยนแปลงข้อมูลใด ๆ



ตำแหน่งแอดเดรส: บนบอร์ดโมโนโครม = 3BEh

พอร์ตขนานหมายเลข 1 = 37Ah

พอร์ตขนานหมายเลข 2 = 27Ah

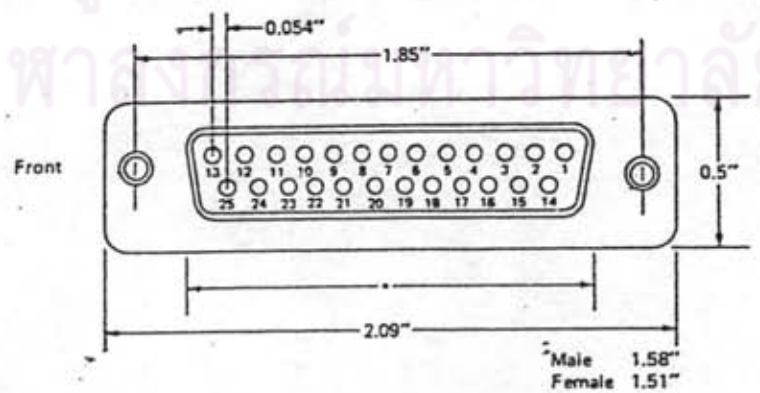
รูปที่ 2.5 พอร์ต C ของชุดพอร์ตขนานของ IBM PC

สำหรับบิตที่ 4 จะเป็นบิตที่ไม่ได้ต่อขาสัญญาณออกมาที่ขั้วของพอร์ตขนาน เป็นบิตที่ใช้ในการเอนเอเบิลและดิสเอเบิลสัญญาณอินเตอร์รัปต์ที่ผ่านมาทางบิตที่ 6 (-ACK) ของพอร์ต B บิตที่ 5 - 7 มีสถานะทางตรรกไม่แน่นอน

สัญญาณของพอร์ตต่าง ๆ จะต่อเข้ากับคอนเนกเตอร์แบบ DB 25 ตัวเมีย และมีชื่อสัญญาณที่เรียกตามการใช้งานในการติดต่อกับเครื่องพิมพ์ (ตามมาตรฐาน Centronix) นอกจากนี้ จาก DB 25 มักจะต่อกับสายเครื่องพิมพ์ ซึ่งมีปลายอีกด้านหนึ่งเป็นคอนเนกเตอร์แบบ Centronix 36 ตัวผู้ ที่ใช้ต่อเข้ากับเครื่องพิมพ์

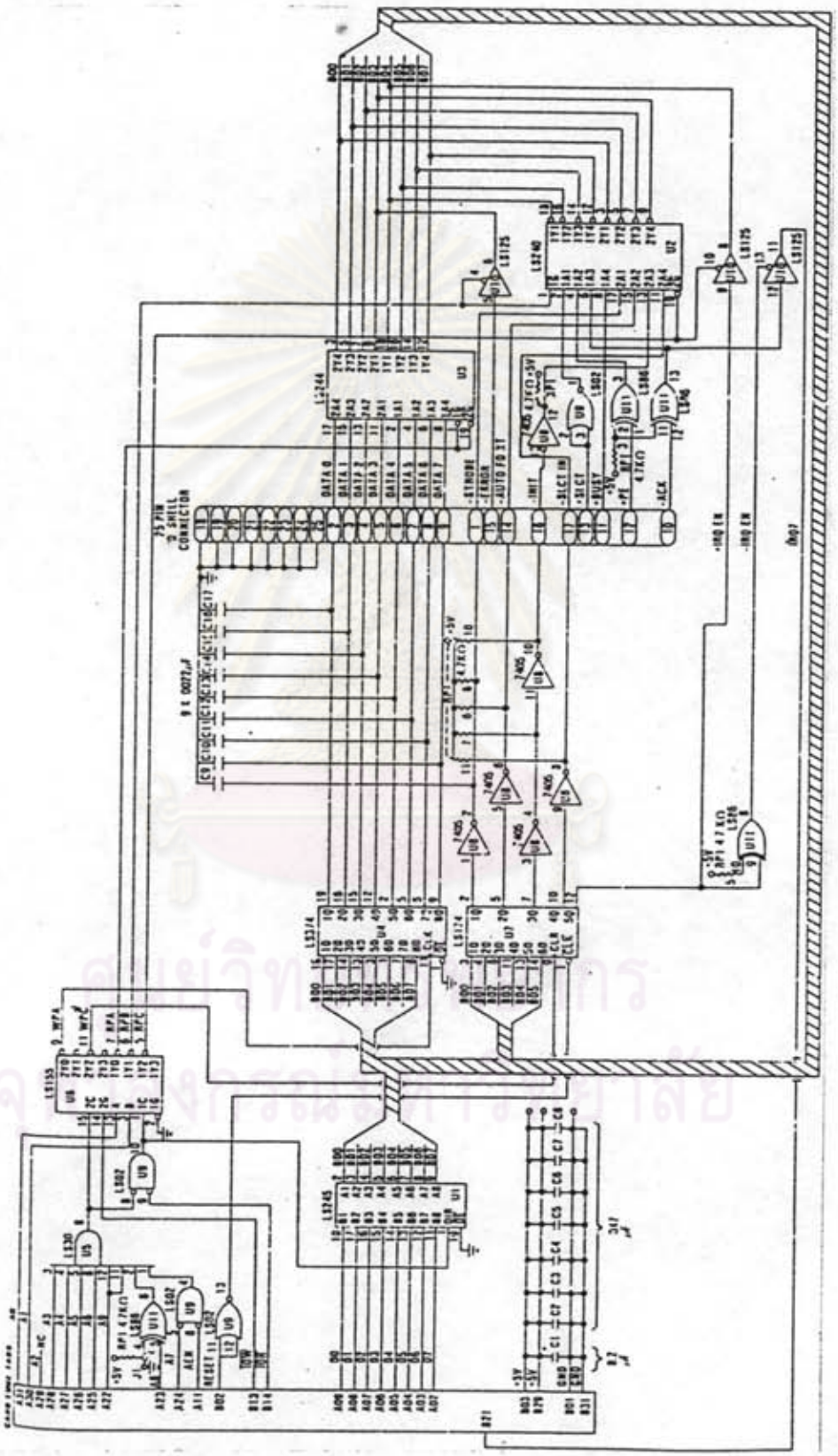
ตารางที่ 2.2 ชื่อสัญญาณของพอร์ตขนานของ IBM PC
และการต่อเข้ากับคอนเนกเตอร์แบบ DB25 และ Centronix36

พอร์ต	บิตที่	ชื่อสัญญาณ	ตำแหน่งขา บน DB 25	ตำแหน่งขา บน centronix 36	หมายเหตุ
A	0	Data 0	2	2	
	1	Data 1	3	3	
	2	Data 2	4	4	
	3	Data 3	5	5	
	4	Data 4	6	6	
	5	Data 5	7	7	
	6	Data 6	8	8	
	7	Data 7	9	9	
B	3	-ERROR	15	32	
	4	+SLCT	13	13	
	5	+PE	12	12	
	6	-ACK	10	10	ต่อเข้ากับ IRQ7 แบบกลับสถานะ
	7	+BUSY	11	11	สถานะทางตรรกเป็นลบ
C	0	-STROBE	1	1	สถานะทางตรรกเป็นลบ
	1	+AUTO FD	14	14	สถานะทางตรรกเป็นลบ
	2	-INIT	16	31	
	3	+SLCT IN	17	36	สถานะทางตรรกเป็นลบ
	4	IRQ EN	-	-	แอนเอเบิล/ดิสเอเบิล สัญญาณอินเตอร์รัปต์ จากบิตที่ 6 ของพอร์ต B



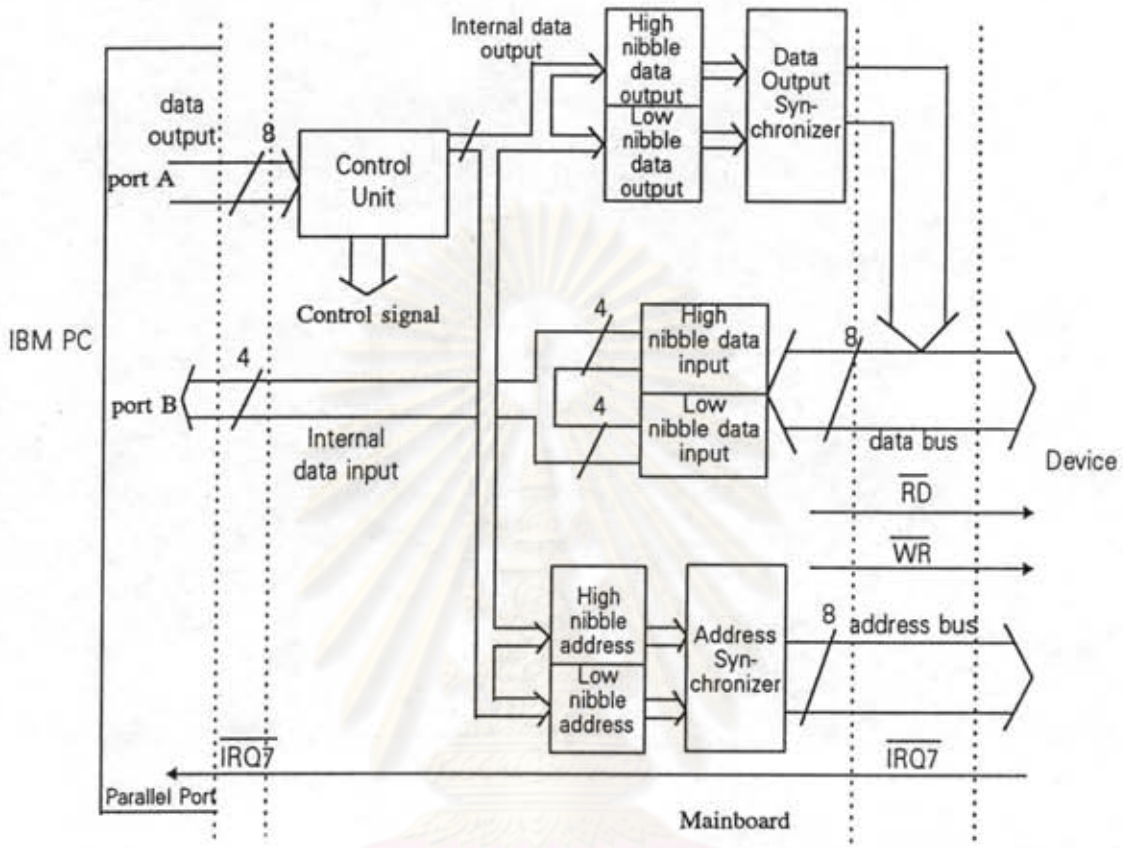
รูปที่ 2.6 แสดงคอนเนกเตอร์แบบ DB 25 ตัวเมีย
(มองจากด้านหน้า)

วงจรของพอร์ตขนาน แสดงดังรูปที่ 2.7



รูปที่ 2.7 วงจรพอร์ตขนาน

2.4.1.3 บล็อกไดอะแกรมของเมนบอร์ด



รูปที่ 2.8 บล็อกไดอะแกรมของเมนบอร์ด

รูปที่ 2.8 แสดงบล็อกไดอะแกรมของเมนบอร์ด ซึ่งประกอบด้วย

1) หน่วยควบคุม (Control Unit)

หน่วยนี้ทำหน้าที่รับสัญญาณขาออกขนาด 8 บิต จาก IBM PC แล้วนำมาสร้างเป็นสัญญาณขาออกของตัวมันเอง ขนาด 4 บิต และสร้างสัญญาณควบคุมต่าง ๆ ข้อมูลออกขนาด 4 บิต สร้างจากพอร์ต A (แอดเดรส 378h สำหรับ LPT1 หรือแอดเดรส 278h สำหรับ LPT2) ที่บิต D0-D3 ของพอร์ตขนานบนเครื่อง IBM PC โดยข้อมูลออกขนาด 4 บิตภายในของ IBM PC จะถูกนำไปสร้างเป็นข้อมูลออกขนาด 8 บิต เพื่อส่งให้บัสข้อมูลของระบบต่อไป สัญญาณควบคุม

ต่าง ๆ จะถูกสร้างจากพอร์ต A ที่บิต D4 - D6 ของพอร์ตขนาน โดยผ่านวงจรตรรกเชิงผสม และนำไปควบคุมบล็อกต่าง ๆ

2) ส่วนข้อมูลออกสี่บิตบน (High Nibble Data Output)

ส่วนนี้เป็นส่วนที่รับข้อมูลออกภายใน ขนาด 4 บิต จากหน่วยควบคุม เพื่อนำไปสร้างเป็น ข้อมูลออกสี่บิตบน D4 - D7 ของบัสข้อมูล ในช่วงของการส่งข้อมูลจาก IBM PC ไปยังอุปกรณ์

3) ส่วนข้อมูลออกสี่บิตล่าง (Low Nibble Data Output)

ส่วนนี้เป็นส่วนที่รับข้อมูลออกภายในขนาด 4 บิต จากหน่วยควบคุม เพื่อนำไปสร้างเป็นข้อมูลออกสี่บิตล่าง D0 - D3 ของบัสข้อมูล ในช่วงของการส่งข้อมูลจาก IBM PC ไปยังอุปกรณ์

4) ส่วนประสานจังหวะข้อมูลออก (Data Output Synchronizer)

ส่วนนี้เป็นส่วนที่ทำหน้าที่รับข้อมูลจากส่วนข้อมูลออกสี่บิตบนและส่วนข้อมูลออกสี่บิตล่าง แล้วส่งข้อมูลไปยังบัสข้อมูลพร้อม ๆ กัน เพื่อทำให้บัสข้อมูลมีการเปลี่ยนแปลงข้อมูลในส่วนของ D0 - D3 และ D4 - D7 พร้อม ๆ กัน นอกจากนี้ ในส่วนประสานจังหวะข้อมูลออกยังทำหน้าที่ในการยึดเกาะ หรือปล่อยบัสข้อมูลตามช่วงเวลาบัสไซเคิลที่มีการเขียนและอ่านข้อมูล

5) ส่วนแอดเดรสสี่บิตบน (High Nibble Address)

ส่วนนี้เป็นส่วนที่รับข้อมูลออกภายในขนาด 4 บิต จากหน่วยควบคุม เพื่อนำไปสร้างเป็นข้อมูลสี่บิตบน D4 - D7 ของแอดเดรสบัส

6) ส่วนแอดเดรสสี่บิตล่าง (Low Nibble Address)

ส่วนนี้เป็นส่วนที่รับข้อมูลออกภายในขนาด 4 บิต จากหน่วยควบคุม เพื่อนำไปสร้างเป็นข้อมูลสี่บิตล่าง D0 - D3 ของแอดเดรสบัส

7) ส่วนประสานจังหวะแอดเดรส (Address Synchronizer)

ส่วนนี้เป็นส่วนที่ทำหน้าที่รับข้อมูล จากแอดเดรสสี่บิตบนและแอดเดรสสี่บิตล่าง แล้วส่งข้อมูลไปยังแอดเดรสบัสพร้อม ๆ กัน เพื่อทำให้แอดเดรสบัสมีการเปลี่ยนแปลงข้อมูลในส่วนของ D0 - D3 และ D4 - D7 พร้อม ๆ กัน

8) ส่วนข้อมูลเข้าสี่บิตบน (High Nibble Data Input)

ส่วนนี้เป็นส่วนที่รับข้อมูลจากบิต D4 - D7 ของบัสข้อมูล เพื่อส่งต่อให้กับ IBM PC

9) ส่วนข้อมูลเข้าสี่บิตล่าง (Low Nibble Data Input)

ส่วนนี้เป็นส่วนที่รับข้อมูลจากบิต D0 - D3 ของบัสข้อมูลเพื่อส่งต่อให้กับ IBM PC ส่วนข้อมูลเข้าสี่บิตบน และส่วนข้อมูลเข้าสี่บิตล่าง จะมีสัญญาณ strobe เพื่อถ่ายข้อมูลเข้าเป็นสัญญาณเส้นเดียวกัน ทั้งนี้ เพื่อให้ข้อมูลที่อ่านได้ในส่วนของสี่บิตบน และสี่บิตล่าง เป็นข้อมูล ณ จุดเวลาเดียวกัน

สัญญาณควบคุมต่าง ๆ จำนวนทั้งหมด 12 เส้น ที่ไม่ได้แสดงไว้ในบล็อกไดอะแกรม ประกอบด้วย

1) LCLK0 : สัญญาณ strobe ข้อมูลออกภายใน ขนาด 4 บิต ไปยังบัสข้อมูลในส่วนข้อมูลออกสี่บิตล่าง ส่วนของข้อมูลสี่บิตล่าง (D0 - D3) เป็นสัญญาณควบคุมบล็อกส่วนข้อมูลออกสี่บิตล่าง

2) HCLK0 : สัญญาณ strobe ข้อมูลออกภายใน ขนาด 4 บิต ไปยังบัสข้อมูลในส่วนข้อมูลออกสี่บิตบน ส่วนของข้อมูลสี่บิตบน (D4 - D7) เป็นสัญญาณควบคุมบล็อกส่วนข้อมูลออกสี่บิตบน

3) OSTB0 : สัญญาณ strobe ข้อมูลสี่บิตบน และข้อมูลสี่บิตล่าง ไปยังบัสข้อมูล สัญญาณนี้จะทำงานภายหลังจากที่ได้เตรียมข้อมูลสี่บิตบน และ ข้อมูลสี่บิตล่างเรียบร้อยแล้ว อีกทั้งเป็นสัญญาณควบคุมบล็อก ส่วนประสานจังหวะข้อมูลออกอีกด้วย

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4) $\overline{OE0}$: สัญญาณเอนเอเบิลข้อมูลออก เพื่อให้ข้อมูลถูกส่งจาก ตัวประสานจังหวะข้อมูลออกมายังบัสข้อมูล อีกทั้งเป็นสัญญาณควบคุมการยึดเกาะบัสของ IBM PC ถ้าสัญญาณนี้มีสถานะเป็น High IBM PC จะปล่อยบัสข้อมูลให้อุปกรณ์อื่น ๆ สามารถใช้งานต่อไป

5) $\overline{WR}$: สัญญาณที่ใช้ในการเขียนข้อมูลจากเครื่อง IBM PC ไปยังอุปกรณ์ (ดังรายละเอียดที่ได้กล่าวแล้วข้างต้น)

6) LCLK1 : สัญญาณ strobe ข้อมูลออกภายในขนาด 4 บิต ไปยังแอดเดรสบัสในส่วนของสี่บิตล่าง(D0 - D3) และเป็นสัญญาณควบคุมบล็อกแอดเดรสสี่บิตล่าง

7) HCLK1 : สัญญาณ strobe ข้อมูลออกภายในขนาด 4 บิตไปยังแอดเดรสบัสในส่วนของสี่บิตบน (D4 - D7) และเป็นสัญญาณควบคุมบล็อกแอดเดรสสี่บิตล่าง

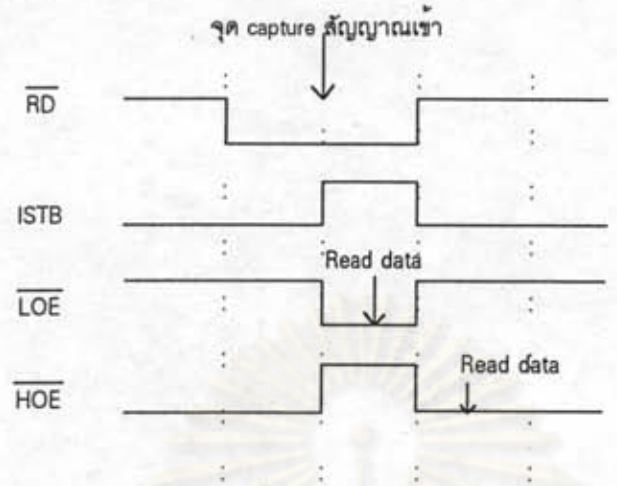
8) OSTB1 : สัญญาณ strobe ข้อมูลแอดเดรสสี่บิตบน และข้อมูลแอดเดรสสี่บิตล่าง ไปยังแอดเดรสบัส สัญญาณนี้จะทำงานภายหลังจากที่ได้เตรียมข้อมูลจากแอดเดรสสี่บิตบน และแอดเดรสสี่บิตล่างเรียบร้อยแล้ว

9) $\overline{RD}$: สัญญาณที่ใช้ในการอ่านข้อมูลจากอุปกรณ์ไปยังเครื่อง IBM PC (ดังรายละเอียดที่ได้กล่าวแล้วข้างต้น)

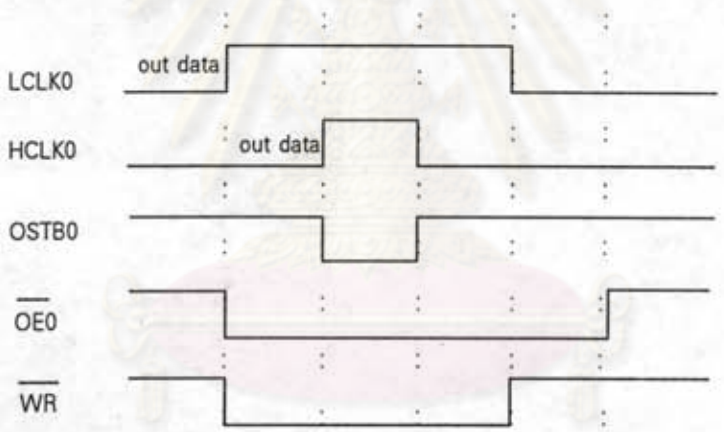
10) ISTB : สัญญาณที่ใช้ strobe ข้อมูลจากบัสข้อมูล ไปยัง IBM PC ในการ strobe นั้นจะทำทั้ง 8 บิต พร้อม ๆ กัน เพื่อให้ข้อมูลเข้าที่ได้ เป็นข้อมูล ณ จุดเวลาหนึ่ง ๆ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

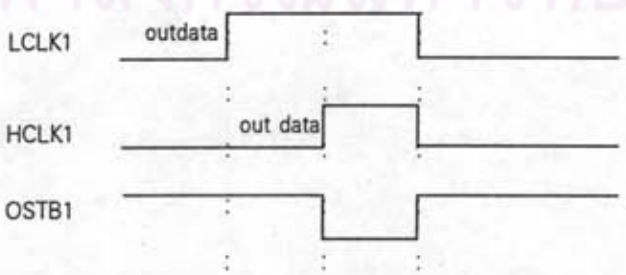
ขณะรับข้อมูลเข้า



ขณะส่งข้อมูลออกทางบัสข้อมูล



ขณะส่งข้อมูลแอดเดรสออก



รูปที่ 2.9 ไดอะแกรมการจัดจังหวะเวลาของสัญญาณควบคุม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

11)$\overline{LOE}$: สัญญาณการอ่านข้อมูลเข้าในส่วนของสี่บิตล่าง สัญญาณนี้เป็นสัญญาณควบคุมบล็อกส่วนข้อมูลเข้าสี่บิตล่าง ซึ่งจะทำให้บล็อกดังกล่าวส่งข้อมูลมายังส่วนข้อมูลเข้าภายในที่มีขนาด 4 บิต

12)$\overline{HOE}$: สัญญาณการอ่านข้อมูลเข้าในส่วนของสี่บิตบน สัญญาณนี้เป็นสัญญาณควบคุมบล็อกส่วนข้อมูลเข้าสี่บิตบน ซึ่งจะทำให้บล็อกดังกล่าวส่งข้อมูลมายังส่วนข้อมูลเข้าภายในที่มีขนาด 4 บิต

2.4.1.4 แนวทางการออกแบบเมนบอร์ด

ในการออกแบบจะต้องพยายามทำให้ระบบทำงานได้เร็วที่สุด โดยได้กำหนดไว้ว่า ระบบจะต้องมีความเร็วในการเขียน-อ่านข้อมูลมากกว่า 10 kHz

จากบล็อกไดอะแกรมของเมนบอร์ด เราจะนำสัญญาณจากพอร์ต A ของพอร์ตขนานมาทำหน้าที่เป็นข้อมูลออกขนาด 8 บิต เนื่องจากเราพบว่า พอร์ต A มีลักษณะเป็นบวกทุกบิต เราจึงไม่ต้องกลับสถานะของสัญญาณแต่อย่างใด บิต D0 - D3 ของพอร์ต A จะนำมาสร้างเป็นข้อมูลออกภายในขนาด 4 บิต และบิต D4 - D6 จะนำมาสร้างเป็นสัญญาณควบคุม โดยนำไปผ่านวงจร เชิงผสม ได้เป็นสัญญาณควบคุมจำนวน 12 เส้นดังที่ได้กล่าวไปแล้ว สำหรับบิต D7 จะใช้ในการแยกสัญญาณข้อมูลออกภายใน(D0 - D3) ออกจากสัญญาณควบคุม(D4 - D6) กล่าวคือ ถ้าข้อมูลที่ส่งมาจากพอร์ต A มีสถานะของ D7 เป็น 1 ข้อมูลดังกล่าวเป็นสัญญาณควบคุม แต่ถ้าข้อมูลที่ส่งมาจากพอร์ต A มีสถานะของ D7 เป็น 0 ข้อมูลดังกล่าวจะเป็นข้อมูลที่ส่งให้แก่ส่วนข้อมูลออกภายใน ซึ่งจะมีวงจรที่ทำหน้าแยกข้อมูลดังกล่าว

ในที่นี้จะเห็นว่า พอร์ต A ถูกนำมาใช้เป็นทั้งสัญญาณข้อมูลและสัญญาณควบคุม สาเหตุที่ไม่ใช้พอร์ต C มาสร้างเป็นพอร์ตที่สร้างสัญญาณควบคุมหรือสัญญาณข้อมูล ก็เนื่องจากพอร์ต C มีบิตที่ทำหน้าที่เอนเอเบิล/ดิสเอเบิลสัญญาณ $\overline{IRQ7}$ อยู่ ซึ่งถ้านำมาใช้งานจะทำให้ไม่สามารถควบคุมการเอนเอเบิล/ดิสเอเบิลสัญญาณอินเตอร์รัปต์ในขณะที่มีการรับ/ส่งข้อมูลได้ อย่างไรก็ตามพอร์ต C ได้ถูกนำไปใช้ในการจ่ายไฟเลี้ยงให้แก่เมนบอร์ดซึ่งจะกล่าวถึงในบทที่ 3

2.4.2 มอดูล

การแบ่งเป็นมอดูล

ภายหลังจากที่ได้สัญญาณบัสข้อมูล, แอดเดรสบัส, $\overline{RD}$, $\overline{WR}$, $\overline{IRQ7}$ แล้วก็จะนำไปเชื่อมโยงเข้ากับอุปกรณ์ต่าง ๆ โดยอุปกรณ์ต่าง ๆ มีลักษณะเป็นมอดูล ในแต่ละมอดูลจะประกอบด้วยส่วนของวงจรถอดรหัสแอดเดรส และส่วนของวงจรที่ทำหน้าที่ตามชนิดของมอดูล มอดูลต่างๆ จะถูกสร้างขึ้นเพื่อครอบคลุมเนื้อหาของการทดลองทั้งหมด ประกอบด้วย

1) มอดูลข้อมูลดิจิทัลเข้า/ออก

ใช้ในการทดลองเรื่องข้อมูลดิจิทัลเข้า/ออก และ การควบคุมสเตปปิงมอเตอร์ มอดูลนี้ประกอบด้วยสวิตช์ ที่ทำหน้าที่ป้อนสัญญาณดิจิทัล ให้แก่เครื่องคอมพิวเตอร์ มี LED ใช้แสดงสถานะทางตรรกของสัญญาณดิจิทัล วงจรขับทรานซิสเตอร์เพื่อช่วยขับอุปกรณ์ที่ต้องการกระแสสูงๆ เช่น สเตปปิงมอเตอร์ และรีเลย์ที่สามารถนำไปใช้ในการควบคุมแบบลำดับ(sequential control)ได้

2) มอดูลพอร์ตสัญญาณขาเข้า/สัญญาณขาออก

ใช้ในการทดลองเรื่องตัวจับเวลา/ตัวนับ การควบคุมการแสดงผลแบบเมทริกซ์ LED และระบบควบคุมแบบดิจิทัล มอดูลพอร์ตสัญญาณขาเข้า/สัญญาณขาออก ใช้ในการรับ/ส่งสัญญาณดิจิทัลกับอุปกรณ์ภายนอก มอดูลนี้ประกอบด้วยพอร์ตสัญญาณขาเข้าจำนวน 2 พอร์ต และ พอร์ตสัญญาณขาออกจำนวน 2 พอร์ตซึ่งมีแลตช์ฟังก์ชัน

3) มอดูลตัวจับเวลา/ตัวนับ (มอดูล T/C)

ใช้ในการทดลองเรื่องตัวจับเวลา/ตัวนับ การอินเตอร์รัปต์ และระบบควบคุมแบบดิจิทัล มอดูลนี้ประกอบด้วยรีจิสเตอร์เคาน์เตอร์ขนาด 16 บิต 3 แชนแนล แยกจากกันโดยอิสระ และวงจรสร้างสัญญาณนาฬิกา ป้อนสัญญาณให้แก่ ตัวจับเวลา/ตัวนับเพื่อใช้เป็นฐานเวลาในการนับ การควบคุมการทำงานของมอดูลจะควบคุมด้วยซอฟต์แวร์ โดยการโปรแกรมรีจิสเตอร์ควบคุม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

I 17465333

4) มอดูล ADAC

ใช้ในการทดลองเรื่องตัวแปลงผัน D/A ตัวแปลงผัน A/D และการอินเตอร์รัปต์ มอดูลประกอบด้วยวงจรแปลงสัญญาณจากแอนะลอกเป็นดิจิทัล และแปลงสัญญาณจากดิจิทัลไปเป็นแอนะลอก และวงจรปรับภาวะสัญญาณเพื่อปรับขนาดและออฟเซ็ตของสัญญาณแอนะลอกขาเข้า/ขาออกให้มีขนาดตามต้องการ อีกทั้งมีวงจรรับสัญญาณจากเซนเซอร์ต่างๆ เช่น ไมโครโฟนตัววัดอุณหภูมิ โฟโตทรานซิสเตอร์ เป็นต้น การควบคุมการทำงานของมอดูลจะควบคุมด้วยซอฟต์แวร์เช่นเดียวกับมอดูล T/C

นอกจากมอดูลที่กล่าวาแล้วนี้ซึ่งใช้งานโดยต่อเข้ากับระบบบัส ยังมีมอดูลอื่น ๆ ที่ไม่ได้ต่อผ่านระบบบัส คือ

1) มอดูลการสื่อสารแบบอนุกรม (serial communication)

ใช้ในการทดลองเรื่องการสื่อสารข้อมูลแบบอนุกรม โดยจะต่อผ่านทางพอร์ตอนุกรมของเครื่อง IBM PC

2) มอดูลการสื่อสารแบบขนาน (parallel communication)

ใช้ในการทดลองเรื่องการสื่อสารข้อมูลแบบขนาน โดยจะต่อผ่านทางพอร์ตขนานของเครื่อง IBM PC โดยตรง

3) มอดูลแสดงผล (display)

ใช้ในการทดลองเรื่องการควบคุมการแสดงผลแบบเมทริกซ์LED ลักษณะการแสดงผลเป็นแบบเมทริกซ์ของ LED ขนาด 8x12 จุด การใช้งานทำโดยต่อมอดูลเข้ากับพอร์ตสัญญาณขาออกของมอดูลพอร์ต I/O

4) มอดูลสเตปปิงมอเตอร์

ใช้ในการทดลองเรื่อง การควบคุมสเตปปิงมอเตอร์ โดยการใช้งานจะต่อต่อเข้ากับมอดูลรับ/ส่งข้อมูลดิจิทัล ในส่วนของวงจรขับทรานซิสเตอร์

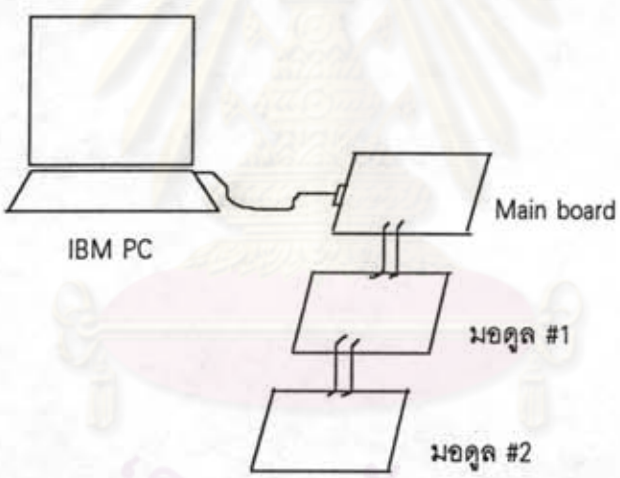
5) มอดูลควบคุมมอเตอร์

ใช้ในการทดลองเรื่องระบบควบคุมแบบดิจิทัล เพื่อควบคุมความเร็วของมอเตอร์กระแสตรง โดยจะใช้งานร่วมกับมอดูล T/C และมอดูลพอร์ต I/O

6) มอดูลเครื่องขยายสัญญาณเสียง (audio amplifier)

ใช้ทดสอบสัญญาณที่อยู่ในช่วงความถี่เสียง โดยใช้ลำโพงเปล่งเสียง

ในการทำการทดลองแต่ละครั้ง เราจะต่อมอดูลที่เกี่ยวข้องเข้ากับเมนบอร์ด และถ้าใช้มอดูลจำนวนหลายตัว เราจะต่อแบบขนานเข้ากับบัสของระบบ โดยรูปแบบที่แสดงภายนอกจะเป็นแบบการต่อแบบคาสเคด (cascade) ระหว่างมอดูลต่าง ๆ (ดังแสดงในรูปที่ 2.10)



รูปที่ 2.10 การต่อมอดูลต่าง ๆ เข้าด้วยกัน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.4.3 โปรแกรมควบคุมการทำงาน

ภาษา C ภาษาที่เลือกใช้ในการควบคุมการทำงาน และใช้ในการทำการทดลอง คือภาษา C ซึ่งเป็นภาษาที่ได้รับความนิยมเป็นอย่างสูงในปัจจุบัน เนื่องจากมีจุดเด่น คือ ได้รวบรวมจุดเด่นของภาษาระดับสูง และภาษาระดับต่ำเข้าด้วยกัน และ เป็นภาษาที่มีลักษณะเป็นโครงสร้างเหมือนอย่างภาษาปาสคาล ซึ่งทำให้แนวการศึกษาการเขียนโปรแกรมด้วยภาษา C จึงเป็นการเริ่มต้นที่ถูกต้อง และมีหลักการ ทำให้ผู้ใช้พร้อมที่จะก้าวขึ้นไปพัฒนาโปรแกรมในระดับ

สูงขึ้นไป ภาษา C สามารถใช้ได้ตั้งแต่ไมโครคอมพิวเตอร์ 8 , 16 และ 32 บิต มินิคอมพิวเตอร์ หรือเมนเฟรม และในปัจจุบันได้มีการพัฒนาให้สามารถใช้กับไมโครคอนโทรลเลอร์เบอร์ต่าง ๆ ได้ โปรแกรมภาษา C มีหลายรุ่น หลายผู้ผลิต แต่ทุก ๆ รุ่นมีโครงสร้างคล้ายกันจนเกือบจะมีลักษณะเป็นมาตรฐาน

จุดเด่นของภาษาระดับสูง

1. สามารถโปรแกรมงานที่มีความซับซ้อนได้ดี
2. ไม่ขึ้นกับเครื่องคอมพิวเตอร์ที่ใช้
3. มีลักษณะเป็นบล็อกโครงสร้าง
4. มีฟังก์ชันให้เลือกใช้งานมาก

จุดเด่นของภาษาระดับต่ำ

1. สามารถเจาะลึกถึงระดับฮาร์ดแวร์
2. ทำงานได้เร็ว
3. ใช้เนื้อที่ในหน่วยความจำน้อย

โปรแกรมภาษา C ที่จะเลือกใช้นั้น ก็คือโปรแกรม Turbo C ของบริษัท Borland เนื่องจากจัดหาได้ง่าย มีราคาไม่แพง และทำงานได้เร็ว รวมถึงมีขนาดที่เล็ก โปรแกรมควบคุมการทำงานจะเขียนในลักษณะของฟังก์ชัน ซึ่งมีอยู่ด้วยกัน 2 ฟังก์ชัน คือ ฟังก์ชัน IN ทำหน้าที่อ่านข้อมูลจากชุดทดลองเข้าเครื่อง IBM PC โดยจะมีลักษณะเหมือนคำสั่ง IN ในภาษาแอสเซมบลี หรือคำสั่ง Inportb( ) ในภาษา C แต่ในการอ่านข้อมูล จะอ่านจากพอร์ตขนาน ไม่ใช่อ่านจากระบบบัสของเครื่องคอมพิวเตอร์ อีกฟังก์ชันหนึ่งคือ ฟังก์ชัน OUT ทำหน้าที่เขียนข้อมูลจากเครื่อง IBM PC ไปยังชุดทดลอง เปรียบเสมือน คำสั่ง OUT ในภาษาแอสเซมบลี หรือ คำสั่ง Outportb( ) ในภาษา C ฟังก์ชันทั้งสองจะถูกบรรจุเข้าไว้ในไลบรารีของ Turbo C ในขณะทำการ install ชุดฝึกทดลอง ซึ่งทำให้ผู้ใช้สามารถใช้งานได้สะดวก โดยในการใช้งาน ผู้ใช้เพียงเพิ่มส่วนของ file Inclusion เข้าไปเพียงบรรทัดเดียวในโปรแกรมที่เขียนขึ้น ก็สามารถเรียกใช้ฟังก์ชันทั้งสองได้

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย