บทที่ 4

ไบออส

ในบทที่ 2 ได้กล่าวถึงการพัฒนาโปรแกรมแล้วว่า โปรแกรมจะถูกแบ่งเป็นชั้น ๆ เพื่อความสะดวกในการพัฒนาโปรแกรม และจะทำให้โปรแกรมมีคุณสมบัติที่สามารถเพิ่มเติม เปลี่ยนแปลงได้ง่าย โดยไม่ต้องแก้ไขเปลี่ยนแปลงใหม่ทั้งหมด

โปรแกรมในชั้นแรกสุดที่ทำหน้าที่เป็นส่วนที่ติดต่อกับฮาร์ดแวร์ของระบบโดยตรง คือ ไบออส (BIOS-Basic Input Output System) โปรแกรมชั้นแรกนี้จะเป็นโปรแกรมย่อย ๆ หลาย ๆ โปรแกรม ที่มีหน้าที่สั่งงานของอินพุต (Input) และเอาท์พุต (Output) ของฮาร์ดแวร์โดยตรง เพื่อให้เกิดการสั่งงานต่าง ๆ ขึ้น

ไบออสจะถูกแบ่งเป็นส่วน ๆ อยู่บนการ์ดหน่วยเชื่อมโยงอุปกรณ์สื่อสารต่าง ๆ ทั้ง 6 ชนิด และอยู่บนหน่วยควบคุมหลักอีกส่วนหนึ่งด้วย ไบออสบนหน่วยเชื่อมโยงอุปกรณ์สื่อสารเหล่านี้ จะเป็นโปรแกรมที่ทำตามคำสั่งของหน่วยควบคุมย่อยบนการ์ดเหล่านั้น ไบออสบนหน่วยควบคุมย่อยเหล่านี้จะมีบางส่วนที่เหมือนกัน เช่น ไบออสในส่วนที่ทำหน้าที่ติดต่อกับฮาร์ดแวร์ส่วนวงจรเชื่อมโยงหน่วยควบคุม และจะมีส่วนที่ต่างกัน คือ ส่วนที่ทำหน้าที่ติดต่อกับฮาร์ดแวร์ส่วนอื่น ๆ ภายในหน่วยเชื่อมโยงอุปกรณ์สื่อสารชนิดนั้น ๆ

บนการ์ดหน่วยเชื่อมโยงอุปกรณ์แต่ละแผ่นจะมีหน่วยควบคุมย่อยอยู่ โปรแกรมบนหน่วยควบคุมย่อยจะมีโครงสร้างโปรแกรมแบ่งเป็นชั้น ๆ ตามที่กล่าวมาแล้ว โดยโปรแกรมที่อยู่ชั้นบนจะสั่งการทำงานผ่านโปรแกรมชั้นล่างที่อยู่ติดต่อกัน โปรแกรมในส่วนควบคุมย่อยจะมีส่วนไบออส, ฟังก์ชันบริการ และโปรแกรมหลัก ครบทั้งสามส่วน โปรแกรมไบออสจะทำหน้าที่ติดต่อกับฮาร์ดแวร์ โดยได้รับการสั่งงาน หรือถูกเรียกใช้จากโปรแกรมในส่วนฟังก์ชันบริการ ในขณะที่โปรแกรมในส่วนฟังก์ชันเหล่านี้ก็จะถูกเรียกใช้โดยโปรแกรมในระดับเดียวกัน หรืออาจจะถูกเรียกใช้จากโปรแกรมหลักอีกทีหนึ่ง

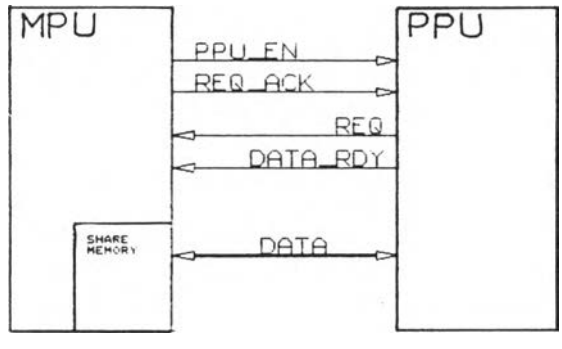
หน่วยควบคุมย่อยจะทำหน้าที่ตามโปรแกรมหลักที่มีอยู่บนหน่วยควบคุมย่อย โดยจะทำงานตามคำสั่งที่ส่งมาจากหน่วยควบคุมหลักอีกทีหนึ่ง ดังนั้นสามารถจะมองได้ว่าโปรแกรมทั้งหมดในหน่วยควบคุมย่อยเป็นไบออสส่วนหนึ่งของหน่วยควบคุมหลัก ซึ่งจะทำหน้าที่ตามคำสั่งที่ส่งมาจากโปรแกรมส่วนฟังก์ชันของหน่วยควบคุมหลัก ซึ่งผ่านมาทางหน่วยเชื่อมโยงหน่วยควบคุมอีกที่หนึ่ง

4.1 ไบออสส่วนเชื่อมโยงหน่วยควบคุม

เนื่องจากการ์ดหน่วยเชื่อมโยงอุปกรณ์สื่อสารแบบต่าง ๆ จะมีฮาร์ดแวร์ของหน่วยควบคุมย่อยเหมือนกันคือ ใช้ไมโครโปรเซสเซอร์ Z - 80 เป็นหน่วยประมวลผล มีหน่วยความจำหลัก คือ หน่วยความจำแรม 8 กิโลไบต์ และหน่วยความจำรอม 8 กิโลไบต์ นอกจากนี้ยังมีหน่วยวงจรเชื่อมโยงหน่วยควบคุมที่เหมือนกัน ดังนั้นจึงมีไบออสบางส่วนของโปรแกรมควบคุมที่เหมือนกัน ไบออสส่วนนี้จะเรียกว่า ไบออสร่วม

ไบออสร่วมนี้จะมีอยู่ในหน่วยเชื่อมโยงอุปกรณ์สื่อสารทุก ๆ หน่วย เพราะเป็นส่วนที่ทำหน้าที่ติดต่อกับหน่วยควบคุมหลัก และจะมีไบออสอีกส่วนที่ทำหน้าที่ติดต่อกับหน่วยควบคุมหลัก ซึ่งเรียกว่า ไบออสส่วนเชื่อมโยงหน่วยควบคุม

ก่อนที่จะกล่าวถึงรายละเอียดของไบออสส่วนนี้ จะขอกล่าวถึงฮาร์ดแวร์และซอฟต์แวร์ของหน่วยเชื่อมโยงหน่วยควบคุม เพื่อจะได้ทำความเข้าใจได้มากขึ้น



รูปที่ 4.1 แผนภาพบล็อกของหน่วยเชื่อมโยงหน่วยควบคุม

ตามที่กล่าวมาแล้วในบทที่ 3 ว่า วงจรเชื่อมโยงหน่วยควบคุมซึ่งแสดงเป็นแผนภาพบล็อกในรูปที่ 4.1 นี้ เป็นวงจรที่ทำหน้าที่เชื่อมโยงระหว่างหน่วยควบคุมหลัก (MPU) และหน่วยควบคุมย่อย (PPU) จะมีหน่วยความจำร่วม (Share Memory) อยู่บนวงจรส่วนที่อยู่ในหน่วยควบคุมหลัก และจะมีสายสัญญาณแฮนด์เชคจำนวน 4 เส้น คือ

1. PPU_EN เป็นสายสัญญาณที่ใช้เริ่มกระบวนการติดต่อระหว่างหน่วยควบคุม โดยสายสัญญาณนี้ส่งมาจากหน่วยควบคุมหลัก สายสัญญาณนี้จะทำให้เกิดการอินเตอร์รัปต์ขึ้น ในหน่วยควบคุมย่อยเพื่อทำให้เกิดกระบวนการติดต่อสื่อสาร

2. REQ_ACK เป็นสายสัญญาณที่กำหนดทิศทางการไหลของข้อมูล ระหว่างหน่วยควบคุมหลักและหน่วยควบคุมย่อย

3. REQ เป็นสายสัญญาณที่บอกความต้องการในการส่งข้อมูลของหน่วยควบคุมย่อย

4. DATA_RDY เป็นสายสัญญาณที่บอกว่า หน่วยควบคุมย่อยทำกระบวนการติดต่อสื่อสารเสร็จสิ้นแล้ว

ในวงจรเชื่อมโยงหน่วยควบคุมนี้ จะมีวงจรนับแอดเดรส (Address Counter) ที่ทำหน้าที่นับค่าแอดเดรสขึ้นทุก ๆ ครั้งที่มีการอ่านหรือเขียนหน่วยความจำร่วมนี้ ระหว่างการอ่านเขียนหน่วยความจำร่วมนี้ หน่วยควบคุมหลักหรือหน่วยควบคุมย่อยจะทำเพียงเขียนหรืออ่านพอร์ต อินพุตหรือเอาท์พุตเท่านั้น ก็จะมีการรับส่งข้อมูลเข้าออกจากหน่วยความจำร่วมนี้และจะมีการเพิ่มค่าแอดเดรสขึ้น 1 ไบต์ จากวงจรนับแอดเดรสโดยอัตโนมัติ หากจะต้องการอ่าน/เขียนข้อมูลไบต์แรกของหน่วยความจำร่วม จะต้องทำการรีเซ็ตวงจรนับแอดเดรสก่อนทุกครั้ง การรับส่งข้อมูลนี้จะทำผ่านบัสข้อมูลร่วมแบบขนานที่ต่อไปจากหน่วยความจำร่วม โดยจะมีการควบคุมการทำงานทางสายสัญญาณแฮนด์เชค

การติดต่อระหว่างหน่วยควบคุมหลักและหน่วยควบคุมย่อย จะมีสายสัญญาณแฮนด์เชค REQ ทำหน้าที่บอกหน่วยควบคุมหลักว่าหน่วยควบคุมย่อยต้องการติดต่อ (Request) สื่อสารหรือไม่ และจะมีสายสัญญาณแฮนด์เชคต่อกลับมายังหน่วยควบคุมย่อย เพื่อบอกว่าหน่วยควบคุมหลักให้รับหรือส่งข้อมูลกับหน่วยควบคุมหลัก สายสัญญาณแฮนด์เชคนี้เรียกว่า REQ_ACK หาก REQ_ACK แอกทีฟ หมายความว่าหน่วยควบคุมหลักรับรู้ความต้องการส่งข้อมูลแล้ว ให้หน่วยควบคุมย่อยนั้นส่งข้อมูลไปได้ แต่หากไม่มีสัญญาณ REQ_ACK ก็หมายความว่า ให้หน่วยควบคุมย่อยรับข้อมูลจากหน่วยควบคุมหลักมาประมวลผลแทน

ทิศทางการส่งข้อมูลระหว่างหน่วยควบคุมหลักและหน่วยควบคุมย่อย จะถูกกำหนดโดยสัญญาณแฮนด์เชค REQ_ACK ที่ส่งมาจากหน่วยควบคุมย่อย โดยเมื่อหน่วยควบคุมย่อยต้องการส่งข้อมูลไปยังหน่วยควบคุมหลัก หน่วยควบคุมย่อยจะส่งสัญญาณ REQ ออกไป เมื่อหน่วยควบคุมหลักพบสัญญาณ REQ ก็จะบอกกลับมายังหน่วยควบคุมย่อยให้เริ่มส่งข้อมูลมาได้โดยการแอกทีฟสัญญาณ REQ_ACK พร้อมทั้งส่งสัญญาณ PPU_EN มายังหน่วยควบคุมย่อย สัญญาณ PPU_EN จะทำการอินเตอร์รัปต์หน่วยควบคุมย่อยให้เริ่มทำการส่งข้อมูลได้ แต่หากไม่มีการส่งสัญญาณ REQ_ACK (ถึงแม้ว่าจะมีสัญญาณ REQ อยู่ก็ตาม) หน่วยควบคุมย่อยก็จะต้องมารับข้อมูลจากหน่วยควบคุมหลักไปแทน

ไบออสส่วนเชื่อมโยงหน่วยควบคุม แบ่งออกเป็น 2 ส่วนคือ ไบออสส่วนเชื่อมโยงหน่วยควบคุมในหน่วยควบคุมย่อย และไบออสส่วนเชื่อมโยงหน่วยควบคุมในหน่วยควบคุมหลัก

4.2 ไบออสส่วนเชื่อมโยงหน่วยควบคุมในหน่วยควบคุมย่อย

ไบออสส่วนนี้จะอยู่ในโปรแกรมควบคุมของหน่วยควบคุมย่อย ซึ่งถูกเขียนด้วยภาษาแอสเซมบลี้ Z-80 โดยใช้คอมไพเลอร์ PARAGON ASMZ80 เป็นตัวคอมไพล์โปรแกรมที่เขียนขึ้นมา โปรแกรมส่วนใหญ่ในส่วนนี้จะเขียนเป็นรูปของมาโคร (Macro) ซึ่งจะถูกเรียกใช้ในโปรแกรมส่วนฟังก์ชันบริการอีกทีหนึ่ง โปรแกรมย่อยในส่วน ไบออสส่วนเชื่อมโยงหน่วยควบคุมในหน่วยควบคุมย่อย จะมีดังต่อไปนี้

1. INT_SERVICE (Interrupt Service Macro) โปรแกรมนี้จะทำหน้าที่กำหนดตำแหน่งแอดเดรส ของโปรแกรมตอบสนองการอินเตอร์รัปต์ (Interrupt Service Routine) ต่าง ๆ ในหน่วยควบคุมย่อย อันได้แก่ อินเตอร์รัปต์ที่แอดเดรส 028H, 030H, 038H และ Non-maskable Interrupt (066H)

2. RESET_CNT (Reset Address-counter Macro) มาโครสำหรับทำการรีเซ็ตวงจรนับแอดเดรสให้เป็นศูนย์ เพื่อจะทำการอ่านเขียนหน่วยความจำร่วมตั้งแต่แอดแดรสแรกได้

3. DIS_H_OUT (Disable handshake Signal Output) มาโครสำหรับยกเลิก (Disable) สัญญาณแฮนด์เชคของวงจรเชื่อมโยงหน่วยควบคุมที่ส่งมาจากหน่วยควบคุมย่อยไปยังหน่วยควบคุมหลัก โดยจะมีค่าที่ส่งไประหว่างเรียกใช้มาโครคือ ค่าหมายเลขสัญญาณแฮนด์เชค

4. EN_H_OUT (Enable Handshake Signal Output) เป็นมาโครสำหรับเปิดทาง (Enable) สัญญาณแฮนด์เชคที่ส่งมาจากหน่วยควบคุมย่อยไปหน่วยควบคุมหลัก

5. WR_INF (Write Interface) มาโครนี้จะทำหน้าที่คัดลอก (Copy) ข้อมูลในบัฟเฟอร์ที่เก็บข้อมูลที่ต้องการส่งจากหน่วยควบคุมย่อยไปหน่วยควบคุมหลัก โดยบัฟเฟอร์นี้จะเป็นส่วนของหน่วยความจำของหน่วยควบคุมย่อย จะทำการคัดลอกข้อมูลมีจำนวนเท่ากับค่าที่กำหนดไว้ในไบต์แรกของบัฟเฟอร์

6. ADD_WR_CBUFF (Add Write Command Buffer) เป็นมาโครที่จะเขียนข้อมูลลงในบัฟเฟอร์ พร้อมทั้งจะเพิ่มตัวชี้ (Intex) ของข้อมูลไปยังไบต์ต่อไปด้วย

7. CLR_CBUFF (Clear Command Buffer) มาโครนี้ใช้สำหรับลบค่าข้อมูลเก่าใน Command Buffer ให้มีค่าเป็นศูนย์ โดยจะมีการกำหนดค่าว่าบัฟเฟอร์ชุดนี้เป็นบัฟเฟอร์ชุดรับข้อมูลเข้า (Read Command Buffer) หรือเป็นบัฟเฟอร์ชุดส่งข้อมูลออก (Write Command Buffer)

8. SET_REQ (Set Request Signal) เป็นมาโครที่ทำหน้าที่แอกทีฟสัญญาณแฮนด์เชค REQ ของหน่วยควบคุมย่อย เพื่อบอกให้หน่วยควบคุมหลักทราบว่าขณะนี้หน่วยควบคุมย่อยนี้มีข้อมูลที่ต้องการส่งให้กับหน่วยควบคุมหลัก

9. INC_INDEX (Increment Index) ในหน่วยควบคุมย่อยจะมีส่วนของหน่วยความจำที่ทำหน้าที่เป็นบัฟเฟอร์ สำหรับรับ/ส่งข้อมูลจำนวน 2 ชุด โดยแยกเป็นบัฟเฟอร์ส่วนรับข้อมูล (Read Command Buffer) และบัฟเฟอร์ส่วนส่งข้อมูล (Write Command Buffer) บัฟเฟอร์ส่วนรับ/ส่งข้อมูล 2 ชุดนี้จะมีตัวชี้ตำแหน่ง (Index) เพื่อบอกตำแหน่งแอดเดรสต่อไปของการรับ/ส่งข้อมูล มาโคร INC_INDEX จะทำหน้าที่เพิ่มค่าตัวชี้ตำแหน่งนี้

มาโครทั้ง 9 มาโครนี้จะเป็นตัวสำคัญ ในการทำการติดต่อกับฮาร์ดแวร์ส่วนเชื่อมโยงหน่วยควบคุม ซึ่งมีส่วนทำให้โปรแกรมควบคุมในหน่วยควบคุมย่อยมีลักษณะทีสามารถเปลี่ยนแปลงฮาร์ดแวร์ได้ โดยเมื่อทำการเปลี่ยนแปลงฮาร์ดแวร์แล้ว จะมีการเปลี่ยนแปลงซอฟท์แวร์ส่วน ไบออสเหล่านี้เท่านั้น ก็สามารถทำให้โปรแกรมในส่วนที่อยู่เหนือขึ้นไปทำงานได้เหมือนเดิม

4.3 ไบออสส่วนเชื่อมโยงหน่วยควบคุมในหน่วยควบคุมหลัก

หน่วยควบคุมหลักตามที่กล่าวมาแล้ว ได้ใช้ไมโครคอมพิวเตอร์ IBM PC ทำหน้าที่เป็นหน่วยควบคุมหลัก โดยวงจรเชื่อมโยงหน่วยควบคุมจะเป็นแผ่นวงจรพิมพ์ ที่มีลักษณะเป็นอินเตอร์เฟสการ์ดเสียบอยู่ในสล็อตของเครื่องไมโครคอมพิวเตอร์ และจะมีสายเชื่อมโยงแบบขนาน เชื่อมโยงมายังการ์ดวงจรเชื่อมโยงหน่วยควบคุมอีกส่วน ที่อยู่บนโครงเสียบการ์ด ซึ่งในแผ่นวงจรจะมีหน่วยความจำร่วมและวงจรนับแอดเดรสอยู่ด้วย

โปรแกรมควบคุมของหน่วยควบคุมหลักเขียนด้วยภาษา C ทั้งหมดตามที่กล่าวมาแล้ว ไบออสส่วนเชื่อมโยงหน่วยควบคุมในหน่วยควบคุมหลัก จะเขียนด้วยภาษา C เช่นเดียวกัน โดยจะประกอบด้วยโปรแกรมย่อยดังต่อไปนี้

1. call_slot โปรแกรมย่อย call_slot จะทำหน้าที่ส่งหมายเลขสล็อตของอุปกรณ์ (Ø-63) ออกไป หน่วยควบคุมย่อยที่มีหมายเลขสล็อตตรงกับหมายเลขที่ส่งไปก็จะอยู่ในสภาวะพร้อมที่จะทำการรับ/ส่งข้อมูลกับหน่วยควบคุมหลัก

2. reset_cnt โปรแกรมย่อย reset_cnt ทำหน้าที่รีเซ็ตวงจรนับแอดเดรสให้มีค่าเป็นศูนย์ ให้อยู่ในสภาวะพร้อมที่จะทำการอ่านหรือเขียนข้อมูลลงในหน่วยความจำร่วม

3. H_out (signal,on_off) H_out (Handshake Output) เป็นโปรแกรมย่อยที่ทำหน้าที่เปิด-ปิด (enable-disable) สัญญาณแฮนด์เชคที่ส่งไปควบคุมการติดต่อของหน่วยเชื่อมโยงหน่วยควบคุม โดยค่าที่ส่งไปคือสัญญาณแฮนด์เชคและสถานะ on/off ของสัญญาณ

4. data_rdy ฟังก์ชันนี้จะใช้ตรวจสัญญาณแฮนด์เชค data_rdy ที่ส่งมาจากหน่วยควบคุมย่อย ซึ่งจะส่งมาเมื่อหน่วยควบคุมย่อยทำการรับ/ส่งข้อมูลเรียบร้อยแล้ว

5. active (signal) ฟังก์ชัน active จะทำการตรวจสอบสัญญาณแฮนด์เชคที่กำหนดว่าแอกทีฟหรือไม่ หากมีการแอกทีฟจะส่งค่า 1 กลับมา ถ้าไม่จะส่งค่า 0 กลับมา

6. PPU_EN (on_off) PPU_EN เป็นโปรแกรมย่อยที่จะทำหน้าที่ส่งสัญญาณแฮนด์เชค PPU_EN จากหน่วยควบคุมหลักไปทำการอินเตอร์รัปต์หน่วยควบคุมย่อยให้ทำการรับ/ส่งข้อมูล

7. Commread (C2) โปรแกรมย่อย commread นี้จะอ่านข้อมูลจากหน่วยความจำร่วมที่อยู่ในหน่วยเชื่อมโยงหน่วยควบคุม และนำมาเก็บไว้ในบัฟเฟอร์ที่รับข้อมูลที่อยู่บนหน่วยควบคุมหลัก

8. wait_data_rdy เป็นฟังก์ชันที่ทำหน้าที่เป็นวงรอบรอการแอกทีฟของสัญญาณแฮนด์เชค data_rdy (Data ready) ที่ส่งมาจากหน่วยควบคุมย่อยหลังขั้นตอนการรับส่งข้อมูล โดยจะทำการตรวจสอบสัญญาณทั้งหมด 20 รอบ หากมีสัญญาณ data_rdy เข้ามาก็จะ return ออกจากโปรแกรมย่อยไปทำงานต่อทันที

4.4 ไบออสหน่วยเชื่อมโยงโทรศัพท์สายภายใน

ตามที่กล่าวมาแล้วในหัวข้อ 3.4 หน่วยเชื่อมโยงโทรศัพท์สายภายในเป็นหน่วยที่ทำหน้าที่เชื่อมโยงกับโทรศัพท์สายภายในจำนวน 8 คู่สายบบการ์ดเดียวกัน หน้าที่หลัก คือ ตรวจสอบสภาวะการ ยกหู, วางหู, การหมุน แล้วรายงานการเปลี่ยนแปลงไปยังหน่วยควบคุมหลัก ดังนั้นจะต้องมี ไบออสที่จะทำหน้าที่ติอต่อกับฮาร์ดแวร์ที่ควบคุม การยกหู,วางหู,การหมุน และอื่น ๆ ด้วย ไบออสของหน่วยเชื่อมโยงโทรศัพท์สายภายในจะมีรายละเอียดดังต่อไปนี้

1. EN_RING (Enable Ring) โปรแกรมย่อย EN_RING จะทำหน้าที่เปิดสัญญาณกระดิ่ง (Ringing) ของเครื่องโทรศัพท์ ที่อยู่ในความควบคุมของหน่วยเชื่อมโยงโทรศัพท์สายภายใน

2. DIS_RING (Disable Ring) โปรแกรมย่อย DIS_RING ทำหน้าที่ตรงข้ามกับ EN_RING คือทำหน้าที่ปิดเสียงกระดิ่งของเครื่องโทรศัพท์ โดยจะส่งพารามิเตอร์ทางรีจิสเตอร์ B เป็นหมายเลขของเครื่องโทรศัพท์ที่จะปิดเสียงกระดิ่ง

3. PRG_8253 (Program 8253) หน่วยเชื่อมโยงโทรศัพท์สายภายใน จะต้องทำหน้าที่สำคัญหน้าที่หนึ่ง คือ ตรวจสอบการหมุนหมายเลขแบบพัลส์ พัลส์ที่เกิดจากจากหมุนหมายเลขของโทรศัพท์ (Dial Pulse) ทั่ว ๆ ไป จะมีค่าเป็น 10 พัลส์ต่อนาที (ppm-Pulse per minute) หรือ 20 พัลส์ต่อนาที ดังนั้นอัตราการหมุนจะมีค่าเป็นวงจรเปิด 66 2/3 มิลลิวินาที (ms) ต่อวงจรปิด 33 1/3 มิลลิวินาที [3] และมีช่วงการปิดวงจรระหว่างพัลส์ของหมายเลขสองหมายเลข มีค่ามากกว่า 200 มิลลิวินาที การตรวจสอบพัลส์การหมุนหมายเลขโทรศัพท์นี้ อาศัยการอินเตอร์รัปต์การทำงานของไมโครโปรเซสเซอร์ทุก 11 มิลลิวินาที เป็นตัวตรวจสอบพัลส์การหมุน โปรแกรมย่อย PRG_8253 จะทำหน้าที่โปรแกรมวงจรนับ 8253 (8253 Timer Counter Circuit) ให้ทำการนับสัญญาณนาฬิกาที่ส่งเข้ามา และวงจรนับ 8253 จะทำการอินเตอร์รัปต์การทำงานของไมโครโปรเซสเซอร์ทุก ๆ 11 มิลลิวินาที

4. SCAN_LC (Scan Line Circuit) ในหน่วยเชื่อมโยงโทรศัพท์สายภายใน ทุก 11 มิลลิวินาทีจะมีการอินเตอร์รัปต์จากวงจรนับ 8253 หน่วยควบคุมย่อยที่อยู่บนหน่วยเชื่อมโยงโทรศัพท์สายภายใน จะเรียกใช้โปรแกรมย่อย SCAN_LC โปรแกรมย่อย SCAN_LC นี้จะรับค่าของสถานะการยกหูวางหูจากอินพุตพอร์ต เข้าไปเก็บไว้ในหน่วยความจำ และจะจัดเก็บสถานะนี้ไว้ในตารางเก็บข้อมูล lc_table ซึ่งจะเป็นตารางเก็บข้อมูลของโทรศัพท์แต่ละเครื่องเอาไว้ ข้อมูลที่ได้จะนำไปประมวลผลการทำงานต่อไปภายหลัง

5. INIT_8SSI (Initialize 8-SSI) โปรแกรมย่อย INIT_8SSI เป็นส่วนที่ทำการตั้งค่าเริ่มต้น (Initialize) ของสัญญาณแฮนด์เชคที่ส่งออกไป, ทำการโปรแกรมไอซี 8255 ให้เป็นเอาท์พุตพอร์ตทั้ง 3 พอร์ต และตั้งค่าในหน่วยความจำที่เก็บค่าสถานะบางสถานะด้วย

4.5 ไบออสหน่วยเชื่อมโยงโทรศัพท์สายภายนอก

หน่วยเชื่อมโยงโทรศัพท์สายภายนอก เป็นส่วนที่เชื่อมโยงกับคู่สายโทรศัพท์ภายนอก (Central Office Line) ที่ต่อมาจากองค์การโทรศัพท์ฯ หน่วยควบคุมย่อยควบคุมหน่วยเชื่อมโยงโทรศัพท์สายภายนอกนี้ จะทำหน้าที่ตรวจสอบสัญญาณกระดิ่งที่ส่งมาจากภายนอกและ

จะทำหน้าที่ส่งพัลส์การหมุนออกไป ตามคำสั่งของหน่วยควบคุมหลัก ไบออสของหน่วยเชื่อมสายภายนอกมีดังต่อไปนี้

1. SCAN_RING (Scan Ringing Signal) ไบออสนี้จะทำหน้าที่ตรวจสอบสัญญาณกระดิ่งที่เกิดขึ้น โดยจะตรวจสอบโดยการอินพุตค่าจากพอร์ต เข้ามาเก็บไว้ในหน่วยความจำ และจะส่งให้โปรแกรมฟังก์ชันชื่อ ANL_RING ทำการประมวลผลตีความสัญญาณต่อไป

2. INIT_8253 (Initialize 8253) โปรแกรมย่อย INIT_8255 จะโปรแกรมวงจรนับ 8253 ให้ทำหน้าที่นับพัลส์ของสัญญาณนาฬิกา และทำการอินเตอร์รัปต์การทำงานทุก ๆ ช่วงเวลา เพื่อใช้เป็นฐานเวลาในการตรวจสอบสัญญาณกระดิ่งที่ส่งเข้ามาจากสายภายนอก

3. INIT_8255 โปรแกรมนี้จะทำหน้าที่โปรแกรมไอซีชิพชัพพอร์ต 8255 ให้เป็นเอาท์พุตพอร์ต 3 พอร์ต เอาท์พุตพอร์ต 8255 นี้จะเป็นเอาท์พุตพอร์ตสำหรับใช้ในการโปรแกรม MUX/DEMUX ที่ใช้เลือกสายสัญญาณ PCM และจะเป็นพอร์ตสำหรับโปรแกรมไอซี Time Slot Assigner Circuit (TSAC) เบอร์ MC 14416 ซึ่งจะเป็นตัวเลือกช่องสัญญาณ (Time Slot) ที่จะรับส่งสัญญาณ PCM

4. INIT_HS (Initialize Handshake) โปรแกรม INIT_HS เป็นโปรแกรมย่อยที่ทำการ Initialize ค่าสัญญาณแฮนด์เชคที่ส่งออกไประหว่างการติดต่อกับหน่วยควบคุมหลัก และจะตั้งค่าหน่วยความจำที่เก็บสถานะบางสถานะของวงจรสายภายนอก

4.6 ไบออสหน่วยแลกเปลี่ยนช่องเวลา

หน่วยแลกเปลี่ยนช่องเวลา นอกจากมีหน้าที่จัดการตัดต่อช่องสัญญาณทำให้เกิดการติดต่อสื่อสารขึ้นแล้ว ยังมีหน้าที่ตัดต่อสัญญาณเสียงโทนต่าง ๆ ไปยังโทรศัพท์สายภายใน, สายภายนอก และจัดส่งสัญญาณเสียง DTMF ส่งออกไปสายภายนอก เพื่อเป็นการหมุนหมายเลขของโทรศัพท์สายภายนอก ดังนั้นไบออสของหน่วยแลกเปลี่ยนช่องเวลาจะมีโปรแกรมย่อย ๆ ดังต่อไปนี้

1. EN_TONE (Macro Enable Tone) มาโคร EN_TONE จะทำหน้าที่เปิดเสียงโทนให้สามารถส่งออกไปในช่องสัญญาณที่กำหนดไว้ โดยชนิดของสัญญาณเสียงโทนจะกำหนดไว้เป็นพารามิเตอร์อีกตัวที่ตามหลังมาโครอีกครั้งหนึ่ง

2. DIS_TONE (Macro Disable Tone) มาโคร DIS_TONE ทำหน้าที่

๑๗๖๖๕๒๖

ตรงข้ามกับ มาโคร EN_TONE คือทำการปิด (disable) สัญญาณเสียงโทนไม่ให้ออกไปในช่องสัญญาณที่กำหนด โดยจะทำการส่งสัญญาณเสียงเงียบ (Null tone) ออกไปแทน

3. GEN_TONE (Macro Generate Tone) มาโคร GEN_TONE จะเรียกใช้โดยโปรแกรมตองสนองอินเตอร์รัปต์อีกทีหนึ่ง GEN_TONE จะทำการเปิดปิดโทนที่กำหนดในช่องเวลาที่กำหนด โดยตรวจสอบสภาวะการเปิดปิดตามค่าในหน่วยความจำที่เก็บสถานะของสัญญาณเสียงโทน และหน่วยความจำที่เป็นตัวนับช่วงเวลา

4. EN_NMI (Enable Non-maskable Interrupt) EN_NMIเป็นโปรแกรมย่อยทีทำหน้าที่เปิดพอร์ตที่ควบคุม สัญญาณ Non-maskble Interrupt (NMI) เนื่องจากเริ่มทำงานโปรแกรมควบคุมหน่วยควบคุมย่อยจะทำการปิดพอร์ตที่ควบคุมสัญญาณ Non-maskable Interrupt เอาไว้ เพื่อจะทำการตั้งค่าเริ่มต้นต่าง ๆ ของโปรแกรมให้เรียบร้อยก่อน เมื่อทำการตั้งค่าต่าง ๆ เสร็จแล้ว จึงเริ่ม Enable สัญญาณอินเตอร์รัปต์ให้เริ่มการทำงานได้

5. INIT_8253 (Initialize 8253) เป็นโปรแกรมย่อยที่ทำหน้าที่โปรแกรมวงจรนับ 8253 ให้นับสัญญาณนาฬิกาที่กำหนด แล้วทำการอินเตอร์รัปต์การทำงานของไมโครโปรเซสเซอร์ เพื่อกำหนดช่วงเวลาการกำเนิดโทนต่าง ๆ

6. INIT_8255 (Initialize 8255) เป็นโปรแกรมย่อยที่ทำหน้าที่โปรแกรมไอซี 8255 ให้เป็นเอาท์พุตพอร์ตทั้ง 3 พอร์ต พอร์ต 8255 นี้จะเป็นพอร์ตที่ใช้โปรแกรมไอซี TSAC และ MUX_DE MUX

7. INIT_HS (Initialize Handshake) โปรแกรม INIT_HS จะทำหน้าที่ตั้งค่าเริ่มต้นของสัญญาณแฮนด์เชคของวงจรเชื่อมโยงหน่วยควบคุม

4.7 ไบออส หน่วยเชื่อมโยงโอเปอเรเตอร์

โอเปอเรเตอร์คอนโซล เป็นส่วนที่จะทำหน้าที่บริการรับสายภายนอกที่เรียกสายเข้ามายังตู้ชุมสายโทรศัพท์อัตโนมัติ โอเปอเรเตอร์คอนโซลถูกสร้างขึ้น ในลักษณะที่มีไมโครโปรเซสเซอร์ Z-80 ควบคุมการทำงาน ในวงจรมีภาคคีย์บอร์ดและภาคแสดงผลเป็น LED 7 Segment จำนวนหนึ่ง ที่สามารถแสดงผลตามคำสั่งของหน่วยควบคุมหลัก โอเปอเรเตอร์คอนโซลจะต่อเข้ากับตู้ชุมสายโทรศัพท์อัตโนมัติ ผ่านหน่วยเชื่อมโยงโอเปอเรเตอร์ โดยการติดต่อสื่อสารระหว่างหน่วยเชื่อมโยงโอเปอเรเตอร์ และโอเปอเรเตอร์คอนโซลเป็นแบบการติดต่อสื่อสารผ่านทางพอร์ตอนุกรม ที่มีการส่งข้อมูลแบบอะซิงโครนัส อัตราเร็วเป็น 1200

บิตต่อวินาที

นอกจากไบออสร่วมที่ทำหน้าที่ติดต่อสื่อสารกับหน่วยควบคุมหลักแล้ว หน่วยเชื่อมโยงโอเปอเรเตอร์จะมี ไบออสอื่น ๆ ดังนี้

1. INIT_8251 โปรแกรมย่อยนี้จะเป็นโปรแกรมย่อยที่จะทำการโปรแกรมไอซีชิพชัพพอร์ต 8251 Programmable Communication Interface ให้ทำงานแบบอะซิงโครนัส ที่ความเร็วข้อมูล 2400 บิตต่อวินาที ข้อมูล บิต ไม่มี stop บิต การติดต่อนี้จะใช้ติดต่อโดยตรงกับโอเปอเรเตอร์คอนโซล

2. INIT_8253 โปรแกรมย่อย INIT_8253 เป็น ไบออสให้ทำหน้าที่ตั้งค่าเริ่มต้น (Initialize) ของวงจรนับ 8253 ให้ทำงานในโหมด 3 ซึ่งจะทำงานเป็น Bit Rate Generator ซึ่งจะทำหน้าที่ป้อนสัญญาณนาฬิกาให้ 8251 ใช้ในการส่งข้อมูลแบบอะซิงโครนัส

3. INIT_8255 โปรแกรมย่อยที่ทำหน้าที่โปรแกรมไอซี 8255 ให้เป็นเอาท์พุตพอร์ต ทั้ง 3 พอร์ต

4. INIT_HS โปรแกรมนี้จะทำหน้าที่ตั้งค่าเริ่มต้นของเอาท์พุตพอร์ตที่จะเป็นทางออกของสัญญาณแฮนด์เชคที่ใช้งานในระหว่างการติดต่อของหน่วยควบคุม

5. SCAN_OPC2 โปรแกรมย่อยส่วนนี้จะทำหน้าที่รับสัญญาณอินพุต ที่เป็นสัญญาณอะซิงโครนัสที่ส่งมาจากโอเปอเรเตอร์คอนโซล ซึ่งจะอ่านค่าจากทางพอร์ตอนุกรม 8251

4.8 ไบออสหน่วยเชื่อมโยงอุปกรณ์สื่อสารแบบอะซิงโครนัส

หน่วยเชื่อมโยงอุปกรณ์สื่อสารแบบอะซิงโครนัส เป็นหน่วยเชื่อมโยงที่ทำหน้าที่เชื่อมโยงอุปกรณ์สื่อสารที่มีการรับส่งข้อมูลแบบอะซิงโครนัส ในการออกแบบตู้ชุมสายโทรศัพท์อัตโนมัติ มีจุดมุ่งหมายให้ใช้งานกับเครื่องไมโครคอมพิวเตอร์ IBM PC การติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์แบบอะซิงโครนัสนี้ โปรแกรมออกแบบให้ทำการเชื่อมโยงกับอุปกรณ์สื่อสารที่ความเร็วข้อมูล 4800 บิตต่อวินาที (bps)

ไบออสของหน่วยเชื่อมโยงอุปกรณ์สื่อสารแบบอะซิงโครนัส จะมีส่วนสำหรับตั้งค่าเริ่มต้น , โปรแกรมการทำงานของไอซีชิพชัพพอร์ต เป็นต้น โดยจะมีรายละเอียดดังต่อไปนี้

1. INIT_8251 เป็นโปรแกรมย่อยที่จะทำหน้าที่โปรแกรมการทำงานของ ไอซีชิพชัพพอร์ต 8251 จำนวน 2 ชุด ที่อยู่บนการ์ดหน่วยเชื่อมโยงนี้ ให้ทำหน้าที่เป็นพอร์ตสื่อสารแบบอนุกรม ที่รับส่งข้อมูลแบบอะซิงโครนัส มีอัตราเร็วข้อมูลเป็น 4800 บิตต่อวินาที

2. INIT_8253 ไอซี 8253 จะถูกโปรแกรมให้ทำหน้าที่เป็นวงจรหาร ซึ่งจะ

ทำหน้าที่หารสัญญาณนาฬิกาความถี่ 1.8432 MHz เพื่อจะป้อนให้กับ 8251 ใช้เป็นฐานเวลาที่ใช้ในการส่งข้อมูล

3. INIT_HS โปรแกรมย่อยส่วนนี้จะทำหน้าที่ตั้งค่าเริ่มต้นให้กับ สัญญาณแฮนด์เชคต่าง ๆ ที่ใช้ในการติดต่อสื่อสารกับหน่วยควบคุมหลัก

สำหรับ ไบออสในหน่วยถอดรหัสสัญญาณเสียง DTMF นั้น นอกจากจะมี ไบออสร่วมที่ใช้ติดต่อสื่อสารกับหน่วยควบคุมหลักแล้ว จะมี ไบออสอีกเพียงโปรแกรมเดียวเท่านั้น คือ โปรแกรมย่อยชื่อว่า INIT ที่ทำหน้าที่โปรแกรมการทำงานของไอซี 8255 ให้ทำงานเป็นเอาท์พุตพอร์ตทั้ง 3 พอร์ต เพื่อใช้ในการโปรแกรมการทำงานของ MUX/DEMUX และ TSAC และจะทำหน้าที่ตั้งค่าเริ่มต้นของสัญญาณแฮนด์เชค

จากที่กล่าวมาแล้ว จะเห็นได้ว่าไบออสสามารถแบ่งออกตามลักษณะการทำงานเป็น 2 กลุ่ม คือ กลุ่มแรก เป็น กลุ่มที่ทำหน้าที่ติดต่อกับวงจรเชื่อมโยงหน่วยควบคุม ซึ่งจะทำหน้าที่รับส่งข้อมูลกับหน่วยควบคุมหลัก กลุ่มที่สอง คือกลุ่มที่ทำหน้าที่เฉพาะงาน ซึ่งจะทำการติดต่อกับฮาร์ดแวร์อื่น ๆ ได้แก่ ฮาร์ดแวร์ของหน่วยเชื่อมโยงอุปกรณ์สื่อสารแบบต่าง ๆ นั่นเอง ไบออสทั้งสองกลุ่มนี้จะทำงาน ตามการเรียกใช้งานจากฟังก์ชันบริการหรือโปรแกรมหลัก ซึ่งจะกล่าวถึงในบทต่อไป