

## บทที่ 4

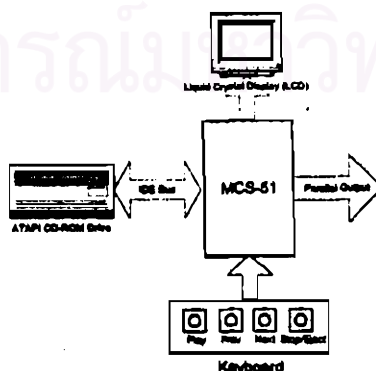
### ส่วนควบคุมเครื่องอ่านแผ่นซีดีรอม

ส่วนควบคุมเครื่องอ่านแผ่นซีดีรอมมีหน้าที่ที่สำคัญสองอย่างคือ ทำการติดต่อกับผู้ใช้โดยรับการควบคุมจากผู้ใช้ผ่านทางคีย์บอร์ดขนาด 4 คีย์ พร้อมกับแสดงผลที่ได้จากการรับคำสั่งออกทางจอแสดงผลผลึกเหลว และทำการควบคุมการเล่น, หยุด และเปิด/ปิดถาดของเครื่องเล่นซีดีรอม

#### 4.1 โครงสร้างของฮาร์ดแวร์

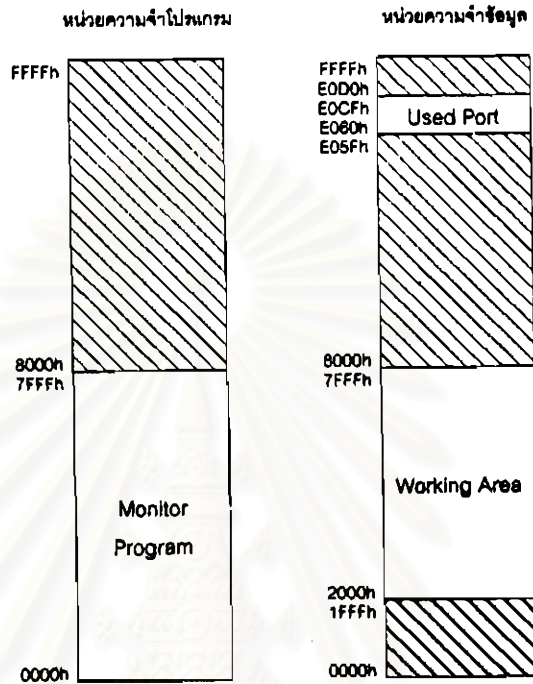
โครงสร้างทางฮาร์ดแวร์ของส่วนควบคุมเครื่องอ่านแผ่นซีดีรอมตามรูปที่ 4.1 ประกอบด้วยส่วนต่างๆ ที่สำคัญดังนี้

1. ตัวประมวลผลกลาง (CPU) เบอร์ 80C320 ของบริษัท Dallas ซึ่งเป็นตัวประมวลผลกลางตระกูล MCS-51 ทำงานที่ความถี่สัญญาณนาฬิกา 11.059 MHz
2. หน่วยความจำอ่านอย่างเดียว (ROM) ขนาด 32 กิโลไบต์ สำหรับเก็บโปรแกรมหลัก (Monitor Program) โดยจะถูกจัดอยู่ในส่วนของหน่วยความจำโปรแกรม
3. หน่วยความจำเข้าถึงแบบสุ่ม (RAM) ขนาด 32 กิโลไบต์ สำหรับเป็นที่เก็บข้อมูลชั่วคราวของตัวแปรต่างๆ ขณะทำงาน โดยจะถูกจัดอยู่ในส่วนของหน่วยความจำข้อมูล
4. พอร์ตขนานขนาด 24 บิต เบอร์ 8255 จำนวน 3 ตัว ใช้สำหรับต่อกับเครื่องเล่นซีดีรอม, คีย์บอร์ด และพอร์ตส่งข้อมูลออกไปให้ส่วนควบคุมการไหลของข้อมูล
4. คีย์บอร์ดขนาด 4 คีย์ทำหน้าที่รับคำสั่งการควบคุมจากผู้ใช้
5. จอแสดงผลผลึกเหลวทำหน้าที่แสดงผลที่ได้จากการรับคำสั่งจากผู้ใช้

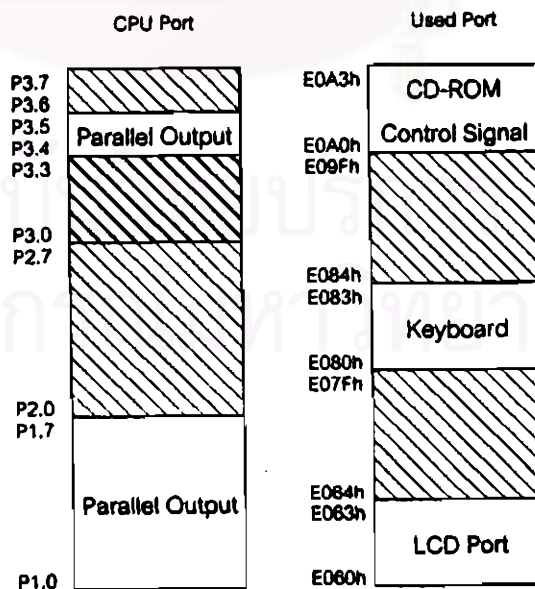


รูปที่ 4.1 โครงสร้างของส่วนควบคุมเครื่องเล่นซีดีรอม

สำหรับการจัดแอดเดรสของหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลแสดงในรูปที่ 4.2 ส่วนการจัดหมายเลขพอร์ตเพื่อติดต่อกับอุปกรณ์ภายนอกแสดงดังรูปที่ 4.3



รูปที่ 4.2 แผนภาพการจัดตำแหน่งของหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล



รูปที่ 4.3 แผนภาพการจัดตำแหน่งของพอร์ตในการควบคุมอุปกรณ์ภายนอกต่างๆ

#### 4.2 ขอบเขตการทำงานของโปรแกรม

ส่วนควบคุมการทำงานของเครื่องเล่นซีดีรอมนี้จะทำงานตามโปรแกรมที่เก็บไว้ในหน่วยความจำอ่านอย่างเดียว โดยที่หน่วยประมวลผลกลางเป็นตัวทำงานตามโปรแกรมคำสั่งที่ได้รับ โปรแกรมการทำงานนี้พัฒนาขึ้นมาจากภาษาแอสเซมบลี (Assembly Language) แล้วแปลเป็นรหัสคำสั่ง (Instruction Code) ด้วยตัวแปลภาษาแอสเซมบลี (Assembler) โดยมีขอบเขตการทำงานของโปรแกรม คือ

1. รับคำสั่งจากคีย์บอร์ด
2. แสดงผลออกทางจอแสดงผลเล็กเหลว
3. สั่งให้เครื่องเล่นซีดีรอมอ่านข้อมูล, หยุดการอ่าน, เปิด/ปิดถาดใส่แผ่นซีดีรอม
4. ตีความข้อมูลที่อ่านออกมาในกรณีที่เป็นไดเรกทอรี
5. ส่งข้อมูลออกทางพอร์ตขนานไปให้ยังส่วนควบคุมการไหลของข้อมูลในกรณีที่มีข้อมูลนั้นเป็นแฟ้มข้อมูล (file)

#### 4.3 การพัฒนาโปรแกรม

เมื่อกำหนดขอบเขตการทำงานของโปรแกรมแล้ว ผู้วิจัยจึงเขียนโปรแกรมโดยเริ่มจากการเขียนเป็นภาษาระบายโปรแกรม (Program Description Language: PDL) ซึ่งเป็นการอธิบายด้วยภาษาที่เข้าใจได้ง่ายกว่าภาษาแอสเซมบลี และไม่ขึ้นกับหน่วยประมวลผลที่ใช้ ภาษาระบายโปรแกรมจะแบ่งออกเป็น 3 ระดับ ในแต่ละระดับจะเพิ่มรายละเอียดสำหรับเขียนโปรแกรมมากขึ้นตามลำดับ โปรแกรมจะแบ่งเป็นสองส่วนคือ โปรแกรมหลัก และโปรแกรมบริการอินเตอร์พรีต ซึ่งจะรองรับคำสั่งจากคีย์บอร์ดก่อนมาแปลเป็นรหัสเพื่อส่งให้โปรแกรมหลักทำงานตามที่ผู้ใช้สั่ง แล้วจึงเขียนเป็นภาษาแอสเซมบลีต่อไป

#### 4.4 โปรแกรมหลัก

1. ภาษาระบายโปรแกรมระดับที่ 1 (Program Description Language Level 1: PDL1)

MODULE: MAIN

Initialize

```

REPEAT
  IF (KeyPlayPress) THEN KeyPlay_routine
    ELSE IF (KeyPrevPress) THEN KeyPrev_routine
    ELSE IF (KeyNextPress) THEN KeyNext_routine
    ELSE IF (KeyStopPress) THEN KeyStop_routine
    ELSE IF (Play_Mode) THEN PlayFile_routine
UNTIL forever
END:

```

โปรแกรมเริ่มต้นทำงานจากการกำหนดเริ่มต้น (initialize) จากนั้นจะเป็นการวนรอรับค่าจากคีย์บอร์ดที่กด ค่านี้จะเกิดขึ้นจากโปรแกรมบริการอินเตอร์รัปต์เมื่อผู้ใช้มีการกดคีย์บอร์ด หากไม่มีการกดคีย์ใดๆ โปรแกรมก็จะทำการตรวจสอบว่าขณะนี้กำลังอ่านข้อมูลจากไฟล์อยู่หรือไม่ (Play\_Mode) หากกำลังอ่านอยู่ก็จะอ่านไฟล์อีกหนึ่งบล็อก ก่อนกลับไปวนตรวจสอบการกดคีย์ต่อไป

## 2. ภาษาบรรยายโปรแกรมระดับที่ 2 (Program Description Language Level 2: PDL2)

MODULE: MAIN

Initialize:

Init LCD

Init CD-ROM

Init ParalelOutput

Reset All Variables (in RAM)

REPEAT

```

IF (KeyPlayPress) THEN KeyPlay_routine
  ELSE IF (KeyPrevPress) THEN KeyPrev_routine
  ELSE IF (KeyNextPress) THEN KeyNext_routine
  ELSE IF (KeyStopPress) THEN KeyStop_routine
  ELSE IF (Play_Mode) THEN PlayFile_routine

```

UNTIL forever

END:

### 3. ภาษาบรรยายโปรแกรมระดับที่ 3 (Program Description Language Level 3: PDL3)

MODULE: *MAIN*

Initialize:

Init LCD

Clear VRAM in LCD

Clear LCD

Init CD-ROM

Reset CD-ROM

Init ParalelOutput

Write 00 to P1

Write 1 to P3.4 (HostClk)

Reset All Variables (in RAM)

Set all variables to initialize data

REPEAT

IF (KeyPlayPress) THEN

KeyPlay\_routine:

CASE ( Mode) IS

Mode=Idle\_Mode:

ScanDisc;

Mode=Display\_Mode;

Display\_Mode:

IF (Directory) THEN ExploreDirectory

ELSE Mode=Play\_Mode;

ELSE IF (KeyPrevPress) THEN

KeyPrev\_routine:

ScrollDisplayUp;

ELSE IF (KeyNextPress) THEN

```

KeyNext_routine:
ScrollDisplayDown;
ELSE IF (KeyStopPress) THEN
KeyStop_routine:
CASE (Mode) IS
Idle_Mode or Display_Mode:
EjectTray;
Play_Mode
StopFile;
ELSE IF (Play_Mode) THEN
PlayFile_routine:
ReadNewData1Block;
SendDat;
UNTIL forever
END:

```

#### 4.5 โปรแกรมบริการอินเตอร์รัปต์

1. ภาษาบรรยายโปรแกรมระดับที่ 1 (Program Description Language Level 1: PDL1)

MODULE: *INTERRUPT*

CheckKey;

Return;

END:

โปรแกรมนี้จะทำงานเมื่อมีการกดคีย์บอร์ดเท่านั้น เมื่อมีการอินเตอร์รัปต์เกิดขึ้นก็จะตรวจสอบว่าคีย์ตัวใดที่ถูกกด เริ่มต้นทำงานจากการกำหนดเริ่มต้น (initialize) จากนั้นจะเป็นการวนรอรับค่าจากคีย์ที่กด ค่านี้จะเกิดขึ้นจากโปรแกรมบริการอินเตอร์รัปต์เมื่อผู้ใช้นำคีย์บอร์ด หากไม่มีการกดคีย์ใดๆ โปรแกรมก็จะตรวจสอบว่าขณะนี้กำลังอ่านข้อมูลจากไฟล์อยู่หรือไม่

(Play\_Mode) หากกำลังอ่านอยู่ก็จะอ่านไฟล์อีกหนึ่งบล็อก ก่อนกลับไปวนตรวจสอบการกดคีย์ต่อไป

## 2. ภาษาบรรยายโปรแกรมระดับที่ 2 (Program Description Language Level 2: PDL2)

MODULE: *INTERRUPT*

CheckKey:

```
IF (Key Play is Pressed) THEN KeyPlayPress
ELSE IF (Key Prev is Pressed) THEN KeyPrevPress
ELSE IF (Key Next is Pressed) THEN KeyNextPress
ELSE IF (Key Stop is Pressed) THEN KeyStopPress
```

Return;

END:

เนื่องจากโปรแกรมในส่วนนี้มีความซับซ้อนไม่มากนัก การเขียน PDL เพียงระดับที่สองก็สามารถเข้าถึงรายละเอียดของโปรแกรมได้เพียงพอแล้ว

เมื่อพัฒนาภาษาบรรยายโปรแกรมทั้งสองส่วนแล้ว จึงนำมาเขียนเป็นภาษาแอสเซมบลี ความยาวประมาณ 2,500 บรรทัด แปลเป็นรหัสคำสั่งขนาดประมาณ 4,100 ไบต์และโปรแกรมลงบนหน่วยความจำอ่านอย่างเดียว เพื่อใช้งานต่อไป ผลการทดสอบการทำงาน of โปรแกรมจะแสดงไว้ในบทที่ 7

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย