

บทที่ 7

โปรแกรมถอดรหัส

7.1 ข้อกำหนดการพัฒนาโปรแกรม

โปรแกรมถอดรหัส เป็นโปรแกรมที่พัฒนาด้วยภาษาแอสเซมบลี (Assembly Language) ของหน่วยประมวลผลกลางที่ออกแบบขึ้นมา แล้วทำการแปลเป็นรหัสคำสั่ง (Instruction Code) ด้วยตัวแปลภาษาแอสเซมบลี (Assembler) นำไปบันทึกลงในหน่วยความจำอ่านอย่างเดียว (Read Only Memory) โปรแกรมที่พัฒนาขึ้นมาี้ความสามารถดังนี้

1. โปรแกรมสามารถถอดรหัสคำบรรยายภาพได้จากทั้ง 4 ช่องสัญญาณข้อมูล ดังนี้
 - ช่องสัญญาณข้อมูล CC1 ข้อมูลจากช่องสัญญาณนี้ จะถูกถอดรหัส และนำมาแสดงในโมดคำบรรยายภาพ เป็นภาษาอังกฤษ
 - ช่องสัญญาณข้อมูล CC2 ข้อมูลจากช่องสัญญาณนี้ จะถูกถอดรหัส และนำมาแสดงในโมดคำบรรยายภาพ เป็นภาษาไทย
 - ช่องสัญญาณข้อมูล T1 ข้อมูลจากช่องสัญญาณนี้ จะถูกถอดรหัส และนำมาแสดงในโมดข้อความ เป็นภาษาอังกฤษ
 - ช่องสัญญาณข้อมูล T2 ข้อมูลจากช่องสัญญาณนี้ จะถูกถอดรหัส และนำมาแสดงในโมดข้อความ เป็นภาษาไทย
2. โปรแกรมจะเปลี่ยนการแสดงผลคำบรรยายภาพในทันทีที่ ผู้ใช้กดปุ่มบนแผงควบคุมของเครื่องถอดรหัส

7.2 การทำงานของโปรแกรม

เนื่องจากว่าโปรแกรมที่เขียนขึ้นมาเป็นภาษาแอสเซมบลี ซึ่งทำความเข้าใจได้ยาก จึงขอบรรยายแนวความคิดในการออกแบบโปรแกรมด้วยภาษาบรรยายโปรแกรม (Program Description

Language:PDL) โดยจะเริ่มบรรยายจากแนวความคิดกว้าง ๆ และค่อย ๆ เพิ่มรายละเอียดลงไปทีละระดับจนถึงระดับที่มีรายละเอียดพอที่จะนำไปพัฒนาเป็นโปรแกรมได้ ดังนี้

7.2.1 ภาษาบรรยายโปรแกรมระดับ 1 (PDL Level 1)

MODULE:Main

System Initialization

REPEAT

IF (caption data come in) THEN Decode Data

IF (control panel switch is pressed) THEN Change Caption Display

UNTIL Forever

END Main

แนวความคิดหลักของโปรแกรมก็คือ โปรแกรมจะทำงานเป็น 2 ช่วง ในช่วงแรกจะทำการกำหนดเริ่มต้นระบบ (System Initialization) หลังจากนั้นจะเป็นช่วงการทำงานที่จะวนรอบคอยตรวจว่ามีข้อมูลคำบรรยายภาพเข้ามาให้ถอดรหัสหรือไม่ และคอยตรวจการกดปุ่มบนแผงควบคุมของผู้ใช้เพื่อที่จะได้ทำการตอบสนองได้ถูกต้อง

7.2.2 ภาษาบรรยายโปรแกรมระดับ 2 (PDL Level 2)

MODULE:Main

System Initialization:

Clear Display Memory

Set Up OSD Control Register

Set Up Status of Each Data Channel

REPEAT

IF (caption data come in) THEN Decode Data:

IF (check parity pass) THEN

IF (control code) THEN Execute Command

ELSE Print Character

```

IF (control panel switch is pressed) THEN Change Caption Display:
    Write OSD Control Register
UNTIL Forever
END Main

```

ภาษาบรรยายโปรแกรมระดับ 2 (PDL Level 2) นี้ กล่าวถึงรายละเอียดในการทำงานแต่ละเอียดอย่างมากขึ้น แต่ยังไม่เพียงพอที่จะเขียนโปรแกรมได้ เนื่องจากยังขาดรายละเอียดที่มีความผูกพันกับฮาร์ดแวร์อยู่ เช่น การพิมพ์ตัวอักษร (Print Character) เพื่อให้ปรากฏบนหน้าจอ ซึ่งงานส่วนนี้จะทำโดยใช้ฮาร์ดแวร์ช่วย ดังนั้นสิ่งที่โปรแกรมจะต้องทำจริง ๆ จึงมีเพียงการเขียนข้อมูลลงในหน่วยความจำเท่านั้น เป็นต้น ภาษาบรรยายโปรแกรมระดับ 3 จะมีรายละเอียดส่วนนี้เพิ่มเข้ามา

7.2.3 ภาษาบรรยายโปรแกรมระดับ 3 (PDL Level 3)

MODULE:Main

System Initialization:

Clear Display Memory:

write 00H to address 8000H-9FFFH (Caption page)

write SPACE to address A000H-BFFFH (Text page)

Set Up OSD Control Register:

write 00H to PAGE register at 3FF4H (show page 0)

Set Up Status of Each Data Channel:

set data structure for each data channel like this

style = pop on (Caption Mode), roll up (Text Mode)

language = English (CC1, T1), Thai (CC2, T2)

cursor row = 1

cursor column = 1

current attribute = white normal

caption height = 4 (Caption Mode), 15 (Text Mode)

disp. mem. = address of each channel's disp. mem.

non disp. mem. = address of each channel's non disp. mem.

REPEAT

IF (caption data come in) THEN Decode Data:

IF (check parity pass) THEN

IF (control code) THEN Execute Command:

IF (new code != old code) THEN

CASE (control code)

PAC: cursor row = row defined in PAC
 cursor column = column defined in PAC
 attribute = attribute defined in PAC
 write PAC code to memory

MID: attribute = attribute defined in MID
 write MID code to memory

RCL: style = pop on

BS: cursor column = cursor column - 1
 write 00H to memory

DER: write 00H until end of row

RU2: style = roll up
 caption height = 2

RU3: style = roll up
 caption height = 3

RU4: style = roll up
 caption height = 4

FON: attribute = attribute + flash
 write FON code to memory

RDC: style = paint on

TR: change to Text Mode
 clear memory at disp. mem.

RTD: change to Text Mode

EDM: clear memory at disp. mem. of Caption Mode

```

CR:   IF (style = roll up) THEN
        move data up 1 row
        write to SCROLL register at 3FF5H (scroll up)
        cursor column = 1

ENM:  clear memory at non disp. mem. of Caption Mode

EOC:  swap disp. mem. and non disp. mem
        write PAGE register at 3FF4H (change page)

TO1:  cursor column = cursor column + 1
TO2:  cursor column = cursor column + 2
TO3:  cursor column = cursor column + 3

ELSE Print Character:
    IF (character = Thai vowel) THEN
        write Thai Vowel code at address = row * 128 + column * 2
    ELSE
        write character at address = row * 128 + column * 2 + 2

IF (control panel switch is pressed) THEN Change Caption Display:
    Write OSD Control Register:
        write PAGE register at 3FF4H (change page)

UNTIL Forever
END Main

```

เมื่อได้ออกแบบโครงร่างของโปรแกรมเป็นภาษาบรรยายโปรแกรม (Program Description Language) ถึงระดับที่ 3 แล้ว ก็สามารถสร้างโปรแกรมถอดรหัสได้ดังภาคผนวก ข