

## บทที่ 4

### โครงสร้างข้อมูล

#### 4.1 โครงสร้างเพิ่มคอนฟิกูเรชันของระบบ

เพิ่มข้อมูลคอนฟิกูเรชันของระบบ คือ เพิ่มที่เก็บข้อมูลที่ระบบจะใช้ในการเริ่มต้นการทำงาน (Initialize) ใช้ชื่อเพิ่มว่า DTS.CFG ข้อมูลในเพิ่มมีอยู่ 4 บรรทัด คือ

- 1.) ชื่อเพิ่มแม่แบบตัวอักษรเพื่อใช้ในการรู้จำตัวอักษร รูปแบบคือ  
FONT=ชื่อเพิ่ม
- 2.) ชื่อเพิ่มคำศัพท์ รูปแบบคือ  
DICTIONARY=ชื่อเพิ่ม
- 3.) ชื่อเพิ่มข้อมูลเสียง รูปแบบคือ  
VOICE=ชื่อเพิ่ม
- 4.) ชื่อเพิ่มคำย่อ รูปแบบคือ  
ABBREVIATION=ชื่อเพิ่ม

#### 4.2 โครงสร้างข้อมูลที่เกี่ยวข้องกับการรู้จำตัวอักษร

##### 4.2.1 โครงสร้างเพิ่มรูปแบบตัวอักษร

เพิ่มรูปแบบตัวอักษร คือ เพิ่มที่เก็บข้อมูลลักษณะเฉพาะของตัวอักษรแต่ละตัว เพื่อใช้ในการรู้จำตัวอักษร เพิ่มชนิดนี้มีนามสกุลเป็น .FON ข้อมูลในเพิ่มแบ่งเป็น 6 section คือ

## 1.) ขนาดตัวอักษร รูปแบบคือ

[Font Size]

Character Width=ความกว้างของตัวอักษร

Short High=ความสูงของตัวอักษรเตี้ย

Normal High=ความสูงของตัวอักษรปกติ

Short Slim Width=ความกว้างของตัวอักษรเตี้ยผอม

Tall Slim Width=ความกว้างของตัวอักษรสูงผอม

## 2.) ข้อมูลตัวอักษรปกติ รูปแบบคือ

[Font Normal Section]

Font No.=จำนวนตัวอักษรปกติ

H1=เส้นในแนวนอนที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

H2=เส้นในแนวนอนที่ 2 ที่ใช้ในการวิเคราะห์ตัวอักษร

H3=เส้นในแนวนอนที่ 3 ที่ใช้ในการวิเคราะห์ตัวอักษร

H4=เส้นในแนวนอนที่ 4 ที่ใช้ในการวิเคราะห์ตัวอักษร

V1=เส้นในแนวตั้งที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

V2=เส้นในแนวตั้งที่ 2 ที่ใช้ในการวิเคราะห์ตัวอักษร

V3=เส้นในแนวตั้งที่ 3 ที่ใช้ในการวิเคราะห์ตัวอักษร

ตัวอักษร จำนวนจุดตัดที่ขอบบน จำนวนจุดตัดที่ขอบล่าง จำนวนจุดตัดที่ขอบซ้าย จำนวนจุดตัดที่ขอบขวา จำนวนจุดตัดในแนวH1 จำนวนจุดตัดในแนวH2 จำนวนจุดตัดในแนวH3 จำนวนจุดตัดในแนวH4 จำนวนจุดตัดในแนวV1 จำนวนจุดตัดในแนวV2 จำนวนจุดตัดในแนวV3

## 3.) ข้อมูลตัวอักษรเตี้ยอ้วน รูปแบบคือ

[Font Short Fat Section]

Font No.=จำนวนตัวอักษรเตี้ยอ้วน

H1=เส้นในแนวนอนที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

V1=เส้นในแนวตั้งที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

V2=เส้นในแนวตั้งที่ 2 ที่ใช้ในการวิเคราะห์ตัวอักษร

ตัวอักษร จำนวนจุดตัดที่ขอบบน จำนวนจุดตัดที่ขอบซ้าย จำนวนจุดตัดที่ขอบขวา จำนวนจุดตัดในแนวH1 จำนวนจุดตัดในแนวV1 จำนวนจุดตัดในแนวV2

4.) ข้อมูลตัวอักษรเตี้ยผอม รูปแบบคือ

[Font Short Slim Section]

Font No.=จำนวนตัวอักษรเตี้ยผอม

H1=เส้นในแนวนอนที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

V1=เส้นในแนวตั้งที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

ตัวอักษร จำนวนจุดตัดในแนวH1 จำนวนจุดตัดในแนวV1

5.) ข้อมูลตัวอักษรสูงอ้วน รูปแบบคือ

[Font Tall Fat Section]

Font No.=จำนวนตัวอักษรสูงอ้วน

H1=เส้นในแนวนอนที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

V1=เส้นในแนวตั้งที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

ตัวอักษร จำนวนจุดตัดที่ขอบบน จำนวนจุดตัดที่ขอบล่าง จำนวนจุดตัดที่ขอบซ้าย จำนวนจุดตัดที่ขอบขวา จำนวนจุดตัดในแนวH1 จำนวนจุดตัดในแนวV1

6.) ข้อมูลตัวอักษรสูงผอม รูปแบบคือ

[Font Tall Slim Section]

Font No.=จำนวนตัวอักษรสูงผอม

V1=เส้นในแนวตั้งที่ 1 ที่ใช้ในการวิเคราะห์ตัวอักษร

V2=เส้นในแนวตั้งที่ 2 ที่ใช้ในการวิเคราะห์ตัวอักษร

ตัวอักษร จำนวนจุดตัดในแนวV1 จำนวนจุดตัดในแนวV2

#### 4.2.2 โครงสร้างข้อมูลรูปแบบตัวอักษร

คือ โครงสร้างข้อมูลในหน่วยความจำหลักของคอมพิวเตอร์ที่เก็บข้อมูลรูปแบบตัวอักษรแต่ละตัวไว้ ซึ่งข้อมูลเหล่านี้ก็คือ ข้อมูลที่อ่านขึ้นมาจากแฟ้มรูปแบบตัวอักษรนั่นเอง ข้อมูลรูปแบบตัวอักษรแบ่งเป็น array 5 ชุด ตามชนิดตัวอักษร คือ

1.) FNormal เป็น array ที่เก็บข้อมูลรูปแบบตัวอักษรปกติ มีจำนวน element เท่ากับที่กำหนดไว้โดย Font No. ใน [Font Normal Section] ในแฟ้มรูปแบบตัวอักษร โดยมีโครงสร้าง

```
typedef struct {
    char ch;
    unsigned int top,bottom,left,right,H1,H2,H3,H4,V1,V2,V3;
} FONTNORMAL;
```

2.) FShortSlim เป็น array ที่เก็บข้อมูลรูปแบบตัวอักษรเตี้ยผอม มีจำนวน element เท่ากับที่กำหนดไว้โดย Font No. ใน [Font Short Slim Section] ในแฟ้มรูปแบบตัวอักษร โดยมีโครงสร้าง

```
typedef struct {
    char ch;
    unsigned int H1,V1;
} FONTSHORTSLIM;
```

3.) FShortFat เป็น array ที่เก็บข้อมูลรูปแบบตัวอักษรเตี้ยอ้วน มีจำนวน element เท่ากับที่กำหนดไว้โดย Font No. ใน [Font Short Fat Section] ในแฟ้มรูปแบบตัวอักษร โดยมีโครงสร้าง

```
typedef struct {
    char ch;
    unsigned int top,left,right,H1,V1,V2;
} FONTSHORTFAT;
```

4.) FTallSlim เป็น array ที่เก็บข้อมูลรูปแบบตัวอักษรสูงผอม มีจำนวน element เท่ากับที่กำหนดไว้โดย Font No. ใน [Font Tall Slim Section] ในแฟ้มรูปแบบตัวอักษร โดยมีโครงสร้าง

```
typedef struct {
    char ch;
    unsigned int V1,V2;
} FONTTALLSLIM;
```

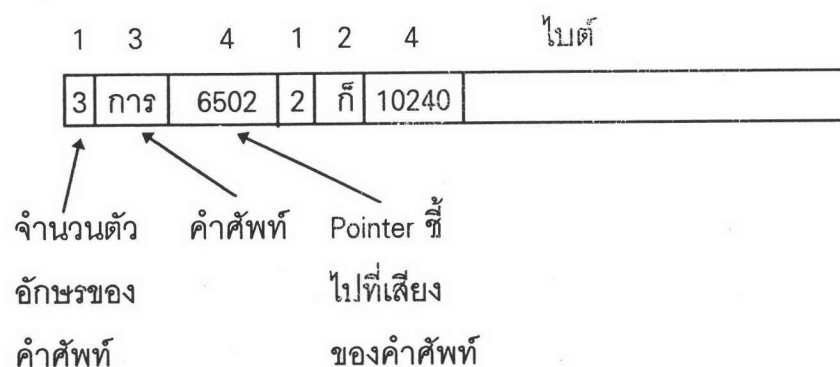
5.) FTallFat เป็น array ที่เก็บข้อมูลรูปแบบตัวอักษรสูงอ้วน มีจำนวน element เท่ากับที่กำหนดไว้โดย Font No. ใน [Font Tall Fat Section] ในแฟ้มรูปแบบตัวอักษร โดยมีโครงสร้าง

```
typedef struct {
    char ch;
    unsigned int top,bottom,left,right,H1,V1;
} FONTTALLFAT;
```

## 4.3 โครงสร้างข้อมูลที่เกี่ยวข้องกับการอ่านออกเสียง

### 4.3.1 โครงสร้างแฟ้มคำศัพท์

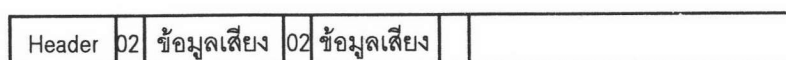
คือ แฟ้มที่เก็บข้อมูลคำศัพท์และ Pointer ที่ชี้ไปยังเสียงของคำศัพท์นั้นในแฟ้มข้อมูลเสียงมีนามสกุลจริงเป็น .DAT แฟ้มคำศัพท์นี้จะใช้ในตอนเริ่มต้นระบบ ( Initialize ) โดยใช้เป็นข้อมูลสร้างต้นไม้คำศัพท์ขึ้นในหน่วยความจำหลักของคอมพิวเตอร์ เมื่อใช้ในการค้นหาคำศัพท์



รูปที่ 4.1 โครงสร้างแฟ้มคำศัพท์

### 4.3.2 โครงสร้างแฟ้มข้อมูลเสียง

คือ แฟ้มที่ใช้เก็บข้อมูลเสียงคำศัพท์แต่ละคำที่บันทึกไว้ มีนามสกุลเป็น .VOC



รูปที่ 4.2 โครงสร้างแฟ้มข้อมูลเสียง

#### 4.3.3 โครงสร้างเพิ่มคำย่อ

คือ แฟ้มที่ใช้เก็บคำย่อและคำเต็มของคำย่อนั้น เช่น คำย่อ ด.ช. จะมีคำเต็มว่าเด็กชาย ในแฟ้มก็จะมีข้อมูลว่า

ด.ช.=เด็กชาย

แฟ้มจะมีนามสกุลเป็น .DAT โดยมีรูปแบบ

คำย่อ=คำเต็ม

#### 4.3.4 โครงสร้างเพิ่มคำศัพท์ใหม่

คือ แฟ้มที่เก็บข้อมูลคำศัพท์ และชื่อแฟ้มข้อมูลเสียงของคำศัพท์นั้น หรือคำศัพท์อื่นที่อ่านออกเสียงเหมือนกัน ( คำพ้องเสียง ) ใช้เป็นข้อมูลในการเพิ่มคำศัพท์ใหม่เข้าไปในพจนานุกรมของระบบ

ข้อมูลในแฟ้มนี้สามารถแบ่งออกได้เป็น 2 รูปแบบ คือ

1.) คำศัพท์ใหม่ และชื่อแฟ้มข้อมูลเสียง เช่น

การ , T10 .VOC

หมายถึง เป็นการบอกระบบให้เพิ่มคำว่า “การ” เข้าไปในพจนานุกรม โดยมีข้อมูลเสียงอยู่ที่แฟ้ม T10 .VOC

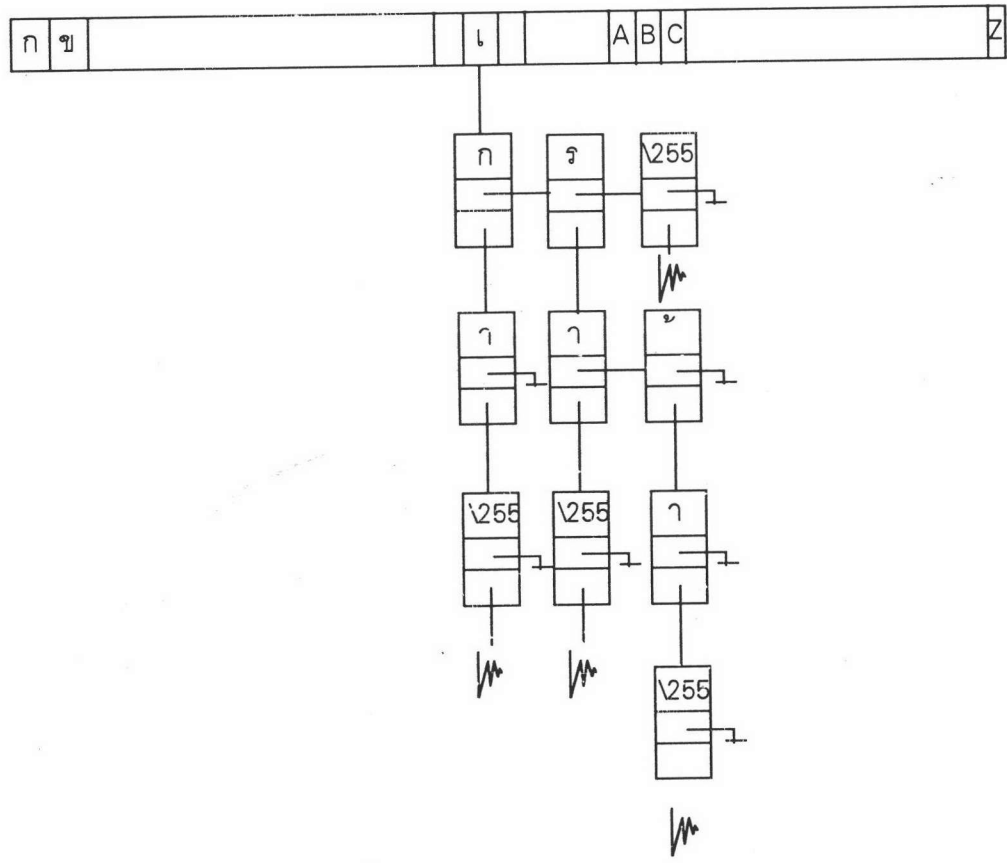
2.) คำศัพท์ใหม่ และคำศัพท์ที่มีอยู่แล้วในพจนานุกรมของระบบ และมีเสียงพ้องกัน เช่น

กาล , “การ”

หมายถึง เป็นการบอกระบบให้เพิ่มคำว่า “กาล” เข้าไปในพจนานุกรม โดยให้ Pointer ชี้ไปยังข้อมูลเสียงคำเดียวกับคำว่า “การ”

4.3.5 โครงสร้างข้อมูลต้นไม้คำศัพท์

คือ ข้อมูลแบบต้นไม้ (Tree) ที่เก็บข้อมูลคำศัพท์และ Pointer ที่ชี้ไปยังข้อมูลเสียงของคำศัพท์นั้น ดังรูปที่ 4.3 ข้อมูลต้นไม้คำศัพท์ทั้งหมดจะอยู่ในหน่วยความจำหลักทั้งหมดตลอดเวลา ตั้งแต่เริ่ม RUN ระบบ โดยสร้างจากเพิ่มคำศัพท์ การที่ต้องสร้างข้อมูลต้นไม้คำศัพท์ในหน่วยความจำหลักก็เพื่อให้การค้นหาคำศัพท์ทำได้รวดเร็วขึ้น



รูปที่ 4.3 โครงสร้างข้อมูลต้นไม้คำศัพท์



#### 4.3.6 โครงสร้างข้อมูลค้าย่อ

คือ ข้อมูลแบบ Link-List โดยที่แต่ละโหนดเก็บข้อมูลค้าย่อและคำเต็มไว้ ข้อมูลค้าย่อนี้สร้างขึ้นจากแฟ้มค้าย่อ ใช้เป็นข้อมูลในการขยายค้าย่อในข้อความให้เป็นคำเต็ม