

บทที่ 4

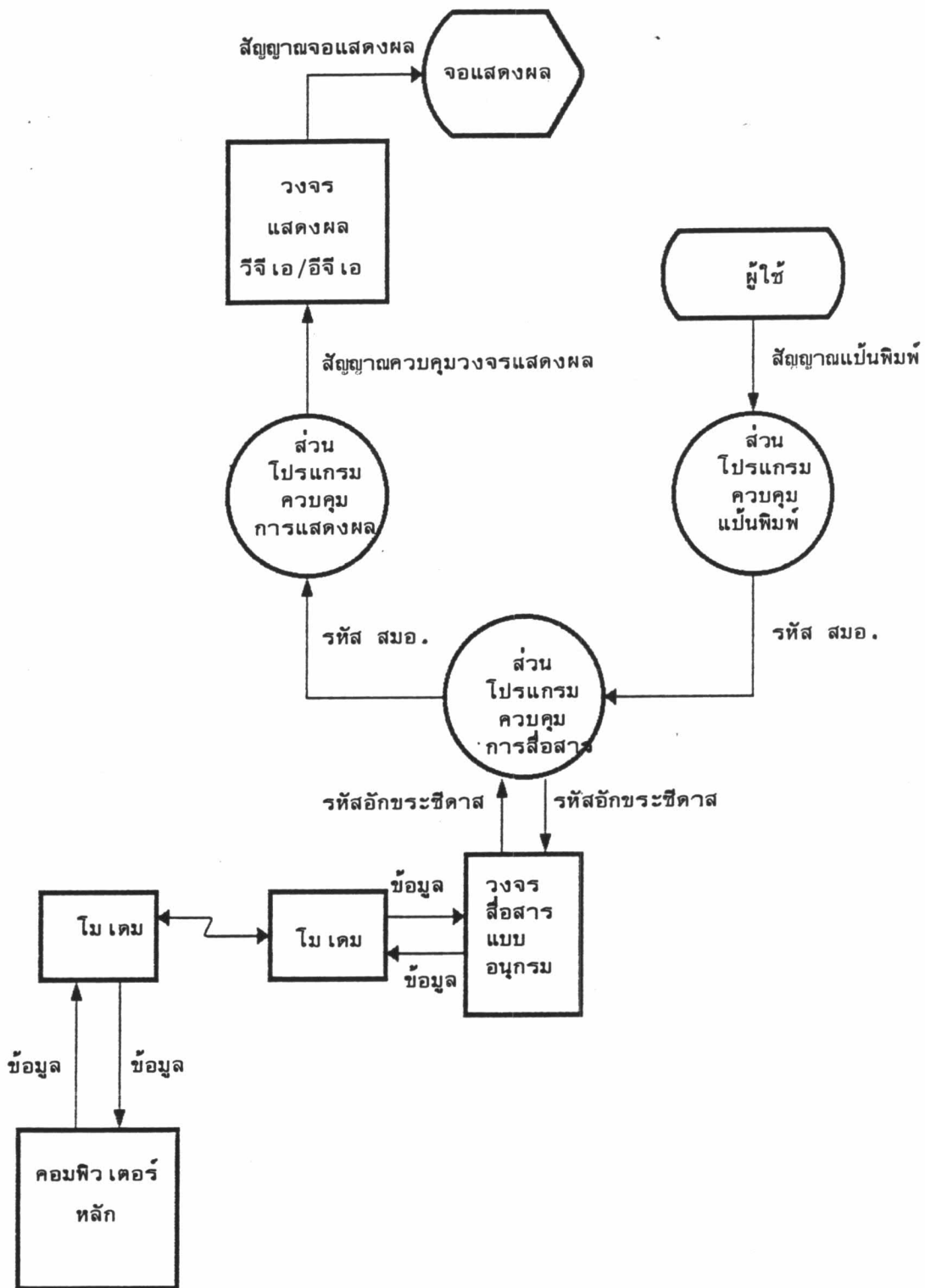
การออกแบบและพัฒนาโปรแกรมเลียนแบบเทอร์มินัลที่อีที

โปรแกรมเลียนแบบเทอร์มินัลที่อีทีทำงานโดยรับข้อมูลจากอุปกรณ์การสื่อสารและนำข้อมูลที่ได้รับมาแสดงบนจอแสดงผล ในขณะที่เดียวกันโปรแกรมสามารถรับข้อมูลจากแป้นพิมพ์และส่งให้อุปกรณ์การสื่อสารเพื่อส่งไปยังคอมพิวเตอร์หลักต่อไป สำหรับความเร็วในการรับส่งข้อมูลคือ 9600 บิตต่อวินาทีและโปรแกรมเลียนแบบเทอร์มินัลที่อีทีสามารถแสดงผลภาษาไทยได้โดยไม่ต้องตัดแปลงฮาร์ดแวร์ นอกจากนี้ผู้ใช้สามารถพิมพ์ข้อมูลที่ปรากฏบนจอแสดงผลทางเครื่องพิมพ์ได้อีกด้วย

การพัฒนาโปรแกรมเลียนแบบเทอร์มินัลที่อีทีได้ออกแบบและพัฒนาสำหรับเครื่องไมโครคอมพิวเตอร์ ซึ่งสามารถแยกการพัฒนาตามหน้าที่การทำงานออกเป็นส่วนต่างๆได้ 3 ส่วนคือ

- 4.1 ส่วนโปรแกรมควบคุมแป้นพิมพ์
- 4.2 ส่วนโปรแกรมควบคุมการแสดงผล
- 4.3 ส่วนโปรแกรมควบคุมการสื่อสาร

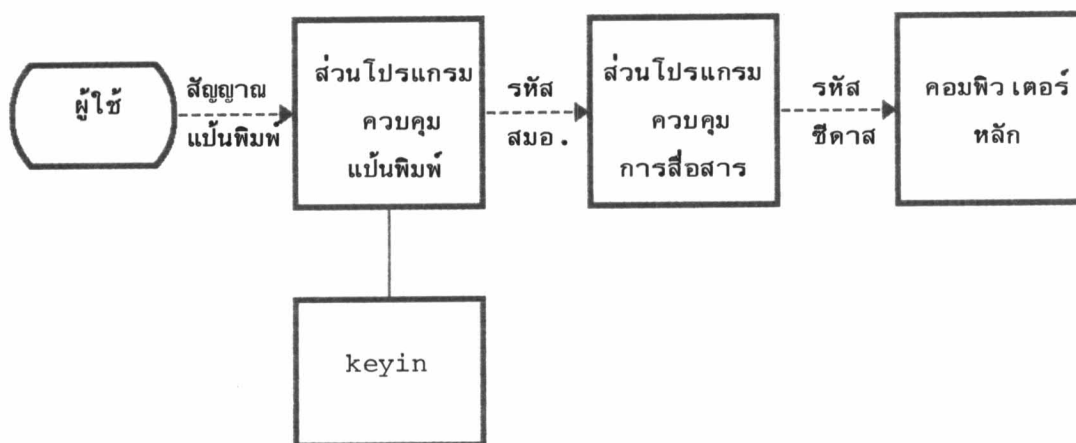
โดยมีผังแสดงการทำงานของโปรแกรมควบคุมแต่ละส่วนดังแสดงในรูปที่ 4.1



รูปที่ 4.1 แสดงผังการทำงานของส่วนโปรแกรมควบคุมแต่ละส่วน

4.1 ส่วนโปรแกรมควบคุมแบ็นพิมพ์

ส่วนโปรแกรมควบคุมแบ็นพิมพ์จะทำหน้าที่รับสัญญาณแบ็นพิมพ์จากผู้ใช้แล้วส่ง เป็นรหัส สมอ. ไปยังส่วนโปรแกรมควบคุมการสื่อสารดังแสดงในรูปที่ 4.2 ซึ่งในส่วนนี้ประกอบด้วยโปรแกรมย่อย "keyin" โดยโปรแกรมย่อยนี้จะมีการตรวจสอบลำดับการพิมพ์สระ วรรณยุกต์และ ตรวจสอบการพิมพ์สระหรือวรรณยุกต์ซ้ำด้วย



รูปที่ 4.2 แสดงส่วนโปรแกรมควบคุมแบ็นพิมพ์

สำหรับโปรแกรมย่อย "keyin" สามารถแสดงในส่วนต่อไป

```

int keyin(int ch,char Tmode)
{
    /*...โปรแกรมย่อยควบคุมการพิมพ์ */
    int ThaiAscii[] = { 32,218,218,50,51,52,219,167,54,55,53,
                        57,193,162,227,189,168,197,45,47,192,
                        182,216,214,164,181,171,199,178,170,204,
                        198,49,196,218,169,175,174,226,172,231,
                        179,235,201,200,63,236,207,173,48,177,
                        166,184,234,206,34,41,237,40,186,92,
                        197,217,56,223,191,212,225,161,211,180,
                        224,233,195,232,210,202,183,215,185,194,
                        230,190,203,208,213,205,228,187,209,188,
                        176,124,44,42 };

    if((Tmode == 0) && (ch > 31)) /*...Typing Thai mode */
    {
        :
        :
        :
        return(f);
    }
    else if((Tmode == 0) && (ch == 8)) /*...Backspace */
    {
        :
        :
        :
        return f;
    }
}

```

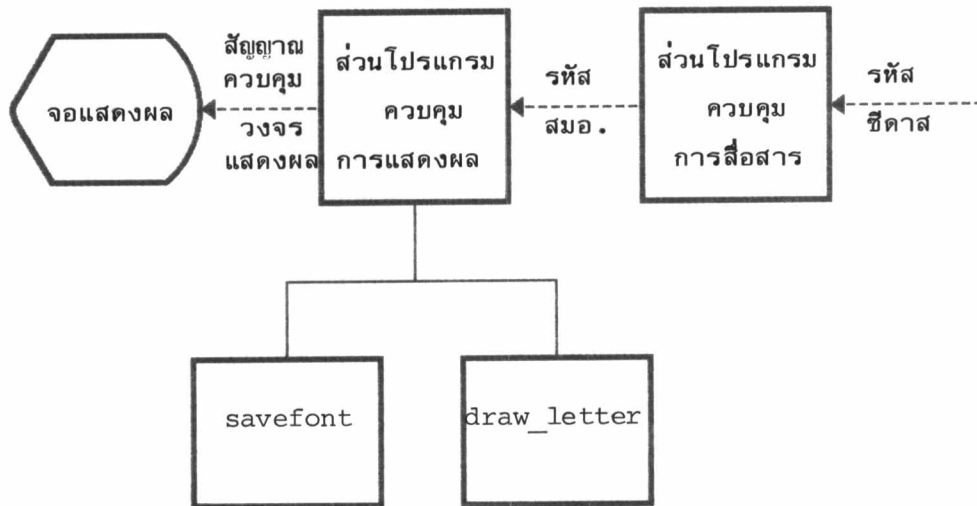
```

else if((Tmode == 1) && (ch > 31))
{
    /*...Typing English mode */
    :
    :
    :
    return f;
}
else if((Tmode == 1) && (ch == 8)) /*...Backspace */
{
    :
    :
    :
    return f;
}
}

```

4.2 ส่วนโปรแกรมควบคุมการแสดงผล

ส่วนโปรแกรมควบคุมการแสดงผล ทำหน้าที่นำรหัส สมอ. จากส่วนโปรแกรมควบคุมการสื่อสารไปแสดงผลบนจอแสดงผลผ่านวงจรแสดงผลอีซีเอหรือวีจีเอดังแสดงในรูปที่ 4.3 ซึ่งในส่วนนี้ประกอบด้วยโปรแกรมย่อย "savefont" และ "draw_letter" โดยในส่วนของโปรแกรมย่อย "savefont" จะทำหน้าที่อ่านค่าจากแฟ้มข้อมูล NORMAL.FON ซึ่งเป็นแฟ้มข้อมูลที่จัดเก็บรูปแบบอักขระที่ใช้ในการแสดงผลของซียูไรท์เตอร์แล้วนำมาจัดเก็บในแถวลำดับ (Array) ที่กำหนดไว้และโปรแกรมย่อย "draw_letter" ทำหน้าที่แสดงผลอักขระภาษาไทยและภาษาอังกฤษบนจอแสดงผลซึ่งอักขระจะมีขนาด 8x20 จุดภาพ



รูปที่ 4.3 แสดงส่วนโปรแกรมควบคุมการแสดงผล

สำหรับโปรแกรมย่อย "savefont" และ "draw_letter" สามารถแสดงในส่วนต่อไป

```

void savefont(void)
{
    /*...บรรจุรูปแบบอักขระจาก normal.fon ในแถวลำดับ */
    int i,j;
    fptr = fopen("normal.fon","rb");
    for(i=0;i<ROWS;i++)
        for(j=0;j<COLS;j++)
        {
            fread(&font[i][j],1,1,fptr);
        }
    fclose(fptr);
}
  
```

```

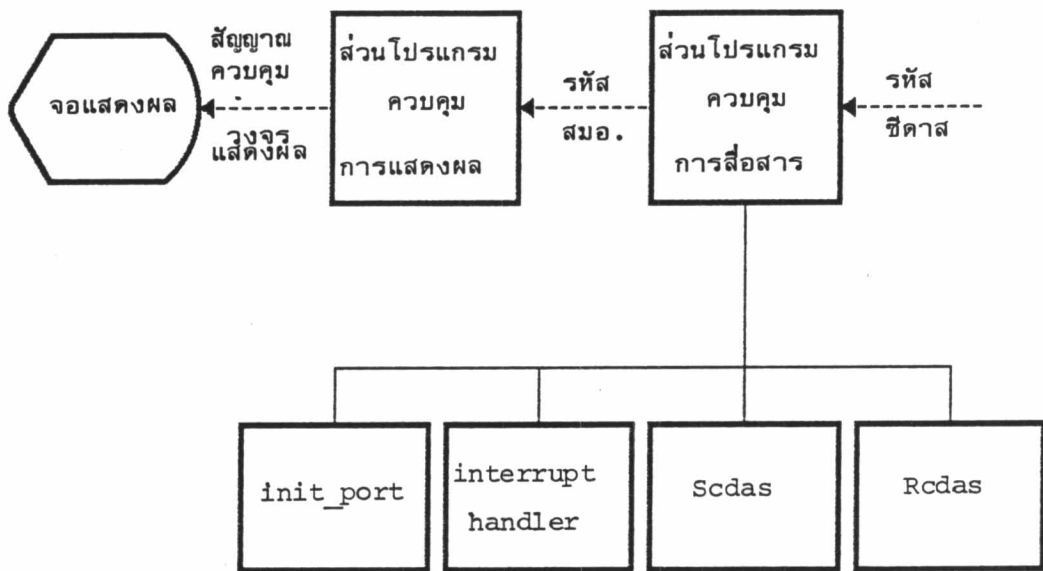
void draw_letter(unsigned char f,unsigned x,unsigned y)
{
    /*...แสดงอักขระภาษาไทยและภาษาอังกฤษบนจอแสดงผล */
    int c,j,bit;
    unsigned int mask;
    for(c=0;c<20;c++)
    {
        mask = 0x80;
        for(j=0;j<8;j++)
        {
            bit = (*(font+f)+c) & mask) ? 1 : 0 ;
            :
        }
    }
}

```

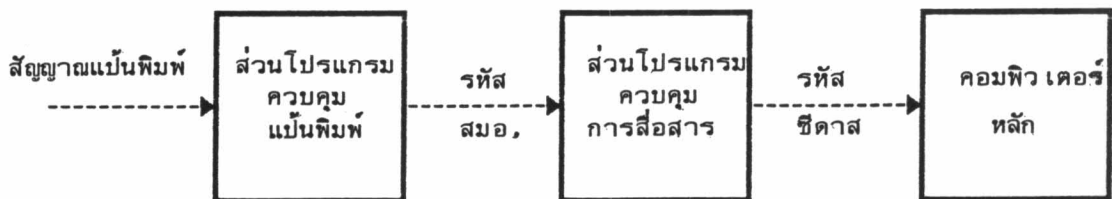
4.3 ส่วนโปรแกรมควบคุมการสื่อสาร

ส่วนโปรแกรมควบคุมการสื่อสารจะทำหน้าที่รับรหัส สมอ. จากส่วนโปรแกรมควบคุมแป้นพิมพ์และแปลงเป็นรหัสอักขระของระบบซีดาสเพื่อส่งต่อไปยังวงจรรีจิสเตอร์แบบอนุกรม และในขณะเดียวกันส่วนโปรแกรมควบคุมการสื่อสารจะรับรหัสอักขระของระบบซีดาสจากวงจรรีจิสเตอร์แบบอนุกรมและแปลงเป็นรหัส สมอ. เพื่อส่งไปยังส่วนโปรแกรมควบคุมการแสดงผล โดยส่วนโปรแกรมควบคุมการสื่อสารประกอบด้วยโปรแกรมย่อย "init_port", "interrupt handler" "Rcdas" และ "Scdas" ซึ่งโปรแกรมย่อย "init_port" จะทำหน้าที่กำหนดค่าเริ่มต้นให้รีจิสเตอร์ภายในของวงจรรวม 8250 ได้แก่ตัวหารแลตซ์นัยสำคัญน้อยที่สุด ตัวหารแลตซ์นัยสำคัญมากที่สุด รีจิสเตอร์ควบคุมสายสื่อสาร รีจิสเตอร์ควบคุมโมเด็มและรีจิสเตอร์อื่นา เบลสัญญาณขัดจังหวะ โปรแกรมย่อย "interrupt handler" เป็นโปรแกรมย่อยบริการสัญญาณขัดจังหวะ โดยจะนำข้อมูลที่ได้รับจากสายสื่อสารมาจัดเก็บในบัฟเฟอร์ที่จัดเตรียมไว้

แต่เนื่องจากรหัสอักขระของระบบซีดาสใช้รหัสของบริษัท ซีสไคแมท จำกัดซึ่งมีการจัดวางตำแหน่งของรหัสอักขระภาษาไทยแตกต่างจากรหัสอักขระภาษาไทยอื่น ๆ ดังนั้นจึงต้องมีโปรแกรมย่อย "Rcdas" ทำหน้าที่แปลงรหัสอักขระจากคอมพิวเตอร์หลักให้เป็นรหัส สมอ. ก่อนส่งให้ส่วนโปรแกรมควบคุมการแสดงผลดังแสดงในรูปที่ 4.4 และโปรแกรมย่อย "Scdas" ทำหน้าที่แปลงรหัสอักขระจากส่วนโปรแกรมควบคุมแบ็นพิมพ์ก่อนส่งต่อไปยังคอมพิวเตอร์หลักดังแสดงในรูปที่ 4.5



รูปที่ 4.4 แสดงการแปลงรหัสอักขระก่อนส่งให้ส่วนโปรแกรมควบคุมการแสดงผล



รูปที่ 4.5 แสดงการแปลงรหัสอักขระก่อนส่งให้คอมพิวเตอร์หลัก

โปรแกรมย่อย "init_port", "interrupt handler", "Rcdas" และ "Scdas" สามารถแสดงในส่วนต่อไป

```

void init_port(void)
    /*...การกำหนดค่าเริ่มต้นให้รีจิสเตอร์ภายในของวงจรรวม 8250 */
{
    int in;

    in = inportb(0x21);
    in = in & 0xE7;
    outportb(0x21, in);
    outportb(LCR, 0x80);
    outportb(LSB, 0x0C);
    outportb(MSB, 0x00);
    outportb(LCR, 0x03);
    outportb(MCR, 0x0B);
    outportb(IER, 0x01);
}

void interrupt_handler(void)
    /*...โปรแกรมย่อยบริการสัญญาณขัดจังหวะ */
{
    unsigned char in_ch;

    in_ch = inportb(RBR);
    putchar(in_ch, inbuf);
    outportb(0x20, 0x20);
}

```

```
unsigned char Rcdas(unsigned char code)
    /*...แปลงรหัสอักขระของระบบบีตาเป็นรหัส สมอ. */
{
    unsigned char Rcdas[] = { 232,34,233,234,37,235,231,40,
                              45,46,47,48,49,50,51,52,
                              53,54,55,56,57,58,59,236,
                              61,198,63,64,161,162,164,166,
                              167,168,169,170,171,172,173,174,
                              175,176,177,178,179,180,181,182,
                              183,184,185,186,187,188,189,190,
                              191,192,193,194,195,196,197,199,
                              200,201,202,203,204,205,206,208,
                              209,210,211,212,213,214,215,216,
                              217,224,225,226,227,228,123,230,
                              125,207,255 };

    if((code > 160) && (code < 169))
    {
        f = Rcdas[code-161];
    }
    /*...Check for the other cases */
    :
    :
    return f;
}
```

```

int Scdas(int f,int Tmode)
    /*...แปลงรหัส สมอ.เป็นรหัสอักษรของระบบขีดาศ */
{
    char Scdas[] = { 0xA1,0x22,0xA3,0xA4,0x25,0xA6,0xA7,0x28,
                    0x2D,0x2E,0x2F,0x30,0x31,0x32,0x33,0x34,
                    0x35,0x36,0x37,0x38,0x39,0x3A,0x3B,0xBC,
                    0x3D,0xBE,0x3F,0x40,0xC1,0xC2,0xC3,0xC4,
                    0xC5,0xC6,0xC7,0xC8,0xC9,0xCA,0xCB,0xCC,
                    0xCD,0xCE,0xCF,0xD0,0xD1,0xD2,0xD3,0xD4,
                    0xD5,0xD6,0xD7,0xD8,0xD9,0xDA,0xDB,0xDC,
                    0xDD,0xDE,0xDF,0xE0,0xE1,0xE2,0xE3,0xE4,
                    0xE5,0xE6,0xE7,0xE8,0xE9,0xEA,0xEB,0xEC,
                    0xED,0xEE,0xEF,0xF0,0xF1,0xF2,0xF3,0xF4,
                    0xF5,0xF6,0xF7,0xF8,0xF9,0xFA,0xFB,0xFC,
                    0xFD,0xFE };

    if(Tmode == 0) /*...Thai code */
    {
        :
        :
    }
    else if(Tmode == 1) /*...English code */
    {
        :
    }
    return code;
}

```

สำหรับปัญหาสำคัญที่มักเกิดขึ้นในการสื่อสารข้อมูลคือการที่ไม่สามารถรับข้อมูลจากวงจรสื่อสารแบบอนุกรมได้ทัน (Overrun) ซึ่งวิธีการแก้ไขปัญหาคือจะต้องใช้ระยะเวลาในการบริการสัญญาณขัดจังหวะให้น้อยที่สุดและหลีกเลี่ยงการใช้สัญญาณขัดจังหวะที่มีลำดับความสำคัญสูงกว่าสัญญาณขัดจังหวะของอุปกรณ์การสื่อสาร