

## รายการอ้างอิง

1. ดร.ธวัช เมฆสุวรรณค์ และนายฟูมิโอะ มิกุมะ. เทคนิคการซ่อมเครื่องรับโทรทัศน์.
2. สมศักดิ์ เตชะเศรษฐ์ธนะ. ทฤษฎีและปฏิบัติ โทรทัศน์สีระบบ PAL. บริษัท ซีเอ็ดยูเคชั่น จำกัด
3. Bernard Grob. Basic Television and Video Systems. McGraw-Hill Book Co.
4. Dewan N.K. and Grad I.E.T.E. Practical T.V. Servicing. New Heights
5. บริษัท SMART ADSAT CO.,LTD. Multi-Screen Video Projection System. (แผ่นโฆษณา)
6. บริษัท 3 MEDIA CO.,LTD. TV WALL SYSTEM. (แผ่นโฆษณา)
7. Richard S. Sandige. Modern Digital Design. McGraw-Hill Book Co. 1990
8. Peter D. Lawrence and Konrad Mauch. Real-Time Microcomputer system desing: an introduction. McGraw-Hill Book Co. 1988
9. High Speed CMOS-logic Databook 1984. Texas Instruments
10. Video Data Handbook 1991. Philips
11. Memory Data Book 1989/1990. Sharp
12. Walter Sikonowiz. Designing and Creating Printed Circuits. Hayden Book Company, inc, New Jersey. 1981
13. Henry W. Ott. Noise Reduction Techniques in Electronic System. A Wiley-Interscience Publication. 1989
14. ชานินทร์ ถาวรศาสนวงศ์ และ ทินกร ดู่ก. การอินเทอร์เฟส IBM PC. ฟิสิกส์เซ็นเตอร์ การพิมพ์



จากนั้นใช้ Karnaugh map เพื่อลดรูปเทอมต่างๆดังนี้  
สำหรับอินพุทของ  $Q_D$

	BA			
DC	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	1	1	0	1
10	1	1	1	1

สำหรับอินพุทของ  $Q_C$

	BA			
DC	00	01	11	10
00	0	0	1	0
01	1	1	0	1
11	1	1	0	1
10	0	0	1	0

สำหรับอินพุทของ  $Q_B$

	BA			
DC	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

สำหรับอินพุตของ  $Q_A$

	BA			
DC	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	0	0	1

จาก Karnaugh map ทั้ง 4 ตารางสามารถสรุปสมการของฟลิปฟล็อปทั้ง 4 ตัว  
ได้ดังนี้

$$Q_D = DC' + DB' + DA' + D'CB A$$

$$Q_C = CB' + CA' + C'BA$$

$$Q_B = BA' + B'A$$

$$Q_A = A'$$

เราสามารถประยุกต์วงจรนับให้เป็นวงจรรหัสได้โดยการทำให้วงจรรับนี้เป็น  
วงจรรับที่กำหนดค่าเริ่มต้นที่จะนับได้และสร้างเอาต์พุตมาอีกค่า ซึ่งเอาต์พุตค่าใหม่ที่สร้างมา  
นี้จะมีค่าเป็นลอจิกสูงก็ต่อเมื่อ เอาต์พุตของวงจรรับทั้งหมดมีค่าลอจิกสูงทั้งหมด เอาต์พุต  
ใหม่นั้นเขียนเป็นสมการได้ดังนี้

$$OUT = Q_D Q_C Q_B Q_A$$

สัญญาณ OUT ที่สร้างขึ้นใหม่นั้นก็คือผลลัพธ์ของวงจรรับนั่นเอง

เพื่อให้วงจรรับที่สร้างขึ้นสามารถกำหนดค่าที่เริ่มนับได้ , สามารถโหลดค่าที่  
กำหนดไว้ทุกครั้งที่นับครบรอบแล้ว( สัญญาณ  $OUT = 1$ ) และให้เอาต์พุตมีค่าเป็นลอจิกสูงทุก  
ครั้งเมื่อเขา clear (CLR) ทำงาน สมการอินพุตของฟลิปฟล็อปทั้ง 4 ตัวจึงเปลี่ยนไปดังนี้

$$Q_D = (OUT)' Q_D Q_C' (CLR) \quad /* \text{ equation for counter } */$$

$$+ (OUT)' Q_D Q_B' (CLR) \quad /* \text{ equation for counter } */$$

$$+ (OUT)' Q_D Q_A' (CLR) \quad /* \text{ equation for counter } */$$

$$+ (OUT)' Q_D' Q_C Q_B Q_A (CLR) \quad /* \text{ equation for counter } */$$

```

+ (CLR)'                /* clear */
+ A3(OUT)(CLR)         /* load preset value when OUT is high */
QC = (OUT)' QC QB' (CLR)    /* equation for counter */
+ (OUT)' QC QA' (CLR)    /* equation for counter */
+ (OUT)' QC' QB QA (CLR)  /* equation for counter */
+ (CLR)'                /* clear */
+ A2(OUT) (CLR)         /* load preset value when OUT is high */
QB = (OUT)' QB QA' (CLR)    /* equation for counter */
+ (OUT)' QB' QA (CLR)    /* equation for counter */
+ (CLR)'                /* clear */
+ A1 (OUT)(CLR)         /* load preset value when OUT is high */
QA = (OUT)' QA'(CLR)    /* equation for counter */
+ (CLR)'                /* clear */
+ (OUT)' A0 (CLR)       /* load preset value when OUT is high */

```

โดย  $A_3, A_2, A_1, A_0$  คือ ขาอินพุตที่ใช้กำหนดค่าที่จะเริ่มต้นนับ (CLR) คือ สัญญาณ clear และ (OUT) คือเอาต์พุตของวงจรหาร

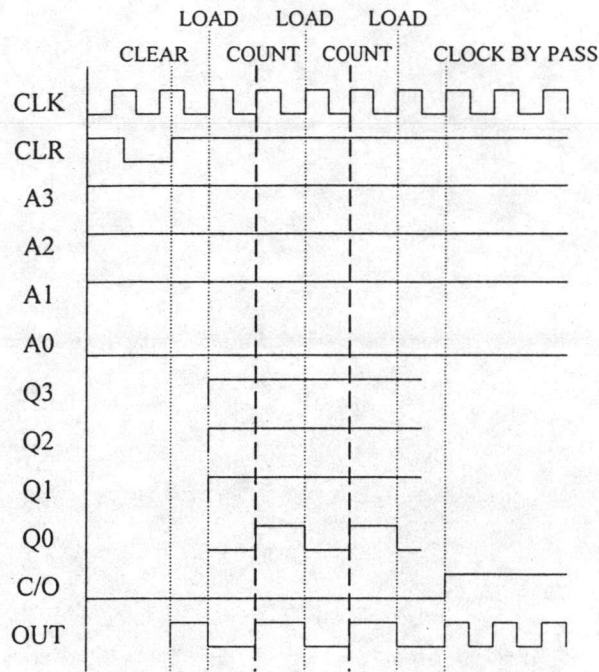
และเนื่องจากวงจรหารที่ต้องการจะต้องสามารถหารด้วย 1 ได้ แต่วงจรนับไม่สามารถทำได้ แต่ความจริงนั้นวงจรหารด้วย 1 คือการผ่านสัญญาณนาฬิกาโดยตรงนั่นเอง ผู้วิจัยจึงออกแบบ สมการของสัญญาณ (OUT) ใหม่ โดยเพิ่มสัญญาณ C/O เพื่อเป็นตัวเลือกว่า สัญญาณ OUT จะเป็น สัญญาณนาฬิกาโดยตรง หรือสัญญาณนาฬิกาที่ผ่านการหารด้วยค่าตั้ง แต่ 2 ขึ้นไป ดังนั้นสมการของสัญญาณ OUT เขียนใหม่ได้ดังนี้

```

OUT = (C/O)' QD QC QB QA (CLR)
+ (CLK)(C/O)(CLR)
+ (CLR)'

```

การทำงานทั้งหมดของวงจรที่ออกแบบสามารถเขียนเป็นผังเวลาได้ดังนี้



รูปแสดงผังเวลาของวงจรหารที่ออกแบบขึ้น

จากสมการทั้งหมดสามารถนำไปเขียนผังพีลส์เพื่อโปรแกรม PEEL18CV8 ได้

ดังนี้

```

L00000 000000000000000000000000000000000000*
L00036 111011111101110111011101110111111111*
L00072 010111111111111111111111111101111111*
L00108 111111111111111111111111111101111111*
L00144 000000000000000000000000000000000000*
L00180 000000000000000000000000000000000000*
L00216 000000000000000000000000000000000000*
L00252 000000000000000000000000000000000000*
L00288 111111101110110111011101110111111111*
L00324 1111010111111111111111111111011111101*
L00360 1111110111111111111111011110111111110*
L00396 111111111111111111111111111101111111*
    
```

L00432 1111110110111101111111110111111111\*  
L00468 11111101101111111101111110111111111\*  
L00504 11111101101111111111110110111111111\*  
L00540 0000000000000000000000000000000000\*  
L00576 11111101111111011011101110111111111\*  
L00612 11111101011111111111111111011111101\*  
L00648 11111101111111111111111101011111110\*  
L00684 11111111111111111111111110111111111\*  
L00720 11111101111110111101111110111111111\*  
L00756 11111101111110111111110110111111111\*  
L00792 0000000000000000000000000000000000\*  
L00828 0000000000000000000000000000000000\*  
L00864 11111101111111111101101110111111111\*  
L00900 111111011111011111111111111011111101\*  
L00936 111111011111111111111111110101111110\*  
L00972 11111111111111111111111111011111111\*  
L01008 11111101111111111011110110111111111\*  
L01044 0000000000000000000000000000000000\*  
L01080 0000000000000000000000000000000000\*  
L01116 0000000000000000000000000000000000\*  
L01152 11111101111111110111111110111111101\*  
L01188 11111101111111111111111110111110110\*  
L01224 11111101111111111111110110111111111\*  
L01260 11111111111111111111111111011111111\*  
L01296 0000000000000000000000000000000000\*  
L01332 0000000000000000000000000000000000\*  
L01368 0000000000000000000000000000000000\*  
L01404 0000000000000000000000000000000000\*  
L01440 0000000000000000000000000000000000\*

L01476 00\*  
L01512 00\*  
L01548 00\*  
L01584 00\*  
L01620 00\*  
L01656 00\*  
L01692 00\*  
L01728 00\*  
L01764 00\*  
L01800 00\*  
L01836 00\*  
L01872 00\*  
L01908 00\*  
L01944 00\*  
L01980 00\*  
L02016 00\*  
L02052 00\*  
L02088 00\*  
L02124 00\*  
L02160 00\*  
L02196 00\*  
L02232 00\*  
L02268 00\*  
L02304 11\*  
L02340 11\*  
L02376 11\*  
L02412 11\*  
L02448 11\*  
L02484 00\*



L02520 000000000000000000000000000000000000\*  
 L02556 000000000000000000000000000000000000\*  
 L02592 000000000000000000000000000000000000\*  
 L02628 000000000000000000000000000000000000\*  
 L02664 00110111011101110111001100110011\*

การออกแบบวงจร line selector

จากรูปที่ 3.7 สามารถนำความสัมพันธ์ของการเกิดภาพมาแสดงดังตารางดังต่อไปนี้

ขนาดที่ต้องการขยาย	ตำแหน่งของเฟรมที่กำลังแสดง	ลำดับข้อมูลภาพที่ปรากฏ 0=เฟรมที่, 1=เฟรมถัด
2	ค	0,1,0,1,0,1,.....
2	จ	0,1,0,1,0,1,.....
3	ค	0,0,1,0,0,1,0,0,1,....
3	จ	0,1,1,0,1,1,0,1,1,....
4	ค	0,0,1,1,0,0,1,1,.....
4	จ	0,0,1,1,0,0,1,1,.....
5	ค	0,0,0,1,1,0,0,0,1,1,...
5	จ	0,0,1,1,1,0,0,1,1,1,...
6	ค	0,0,0,1,1,1,0,0,0,1,1,1,...
6	จ	0,0,0,1,1,1,0,0,0,1,1,1,...
7	ค	0,0,0,0,1,1,1,0,0,0,0,1,1,1,...
7	จ	0,0,0,1,1,1,1,0,0,0,1,1,1,1,...
8	ค	0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,...
8	จ	0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,...
9	ค	0,0,0,0,0,1,1,1,1,...
9	จ	0,0,0,0,1,1,1,1,1,...

10	พ1.	0,0,0,0,0,1,1,1,1,1,...
10	พ2.	0,0,0,0,0,1,1,1,1,1,...
11	พ1.	0,0,0,0,0,0,1,1,1,1,1,...
11	พ2.	0,0,0,0,0,1,1,1,1,1,1,...
12	พ1.	0,0,0,0,0,0,1,1,1,1,1,1,...
12	พ2.	0,0,0,0,0,0,1,1,1,1,1,1,...
13	พ1.	0,0,0,0,0,0,0,1,1,1,1,1,1,...
13	พ2.	0,0,0,0,0,0,1,1,1,1,1,1,1,...
14	พ1.	0,0,0,0,0,0,0,1,1,1,1,1,1,1,...
14	พ2.	0,0,0,0,0,0,0,1,1,1,1,1,1,1,...
15	พ1.	0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,...
15	พ2.	0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,...
16	พ1.	0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,...
16	พ2.	0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,...

เนื่องจากข้อมูลที่ใส่จะต้องป้อนลงไอซีหน่วยความจำหมายเลข 27128 ซึ่งเป็นหน่วยความจำแบบ EPROM ซึ่งมีขนาดใหญ่ จึงใช้โปรแกรม OrCAD/PLD ช่วยสร้างข้อมูลที่ต้องใช้ขึ้นมาดังนี้

```
|PROM16KB8 in:A[13~0], out:LE
```

```
|
```

```
| Stream: STEP[13~0] -> LE
```

```
| { 32(8(0),8(1)), 34(8(0),7(1)),2(0), 36(7(0),7(1)),8(0), 39(7(0),6(1)),5(0),
```

```
| 42(6(0),6(1)),5(8), 46(6(0),5(1)),6(0), 51(5(0),5(1)),2(0),
```

```
| 56(5(0),4(1)),8(0), 64(4(0),4(1)), 73(4(0),3(1)),0, 85(3(0),3(1)),2(0),
```

```
| 102(3(0),2(1)),2(0), 128(4(0),4(1)), 170(2(0),1),2(0), 256(0,1), 512(0),
```

```
|
```

```
| 32(8(0),8(1)), 34(7(0),8(1)),2(0), 36(7(0),7(1)),8(0), 39(6(0),7(1)),5(0),
```

```
| 42(6(0),6(1)),5(8), 46(5(0),6(1)),6(0), 51(5(0),5(1)),2(0),
```

```

| 56(4(0),5(1)),8(0), 64(4(0),4(1)), 73(3(0),4(1)),0, 85(3(0),3(1)),2(0),
| 102(2(0),3(1)),2(0), 128(4(0),4(1)), 170(0,2(1)),2(0), 256(0,1), 512(1)
| }

```

### ซอฟต์แวร์ที่ทดสอบระบบ

```

#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <string.h>

#define data_port 0x378
#define ctrl_port 0x37a

/* Created by Chaya Limchitti on 03 March 1996 */
/* 199603031500 */
/* 199603201300 */
/* 199603201800 */
/* 199603211730 */
/* 199603271500 */
/* Port 378 is for sending data to 74HC374 */
/* Port 37A is a control signal (see next line) */
/* Bit3=write signal(0->1), Bit2 0=Select board 1=select 74HC374 */
/* Bit1 and Bit 0= select each board or 74HC374 depends on Bit2 */
/* Sequence of Bit1 and Bit 0 must be 00,10,11 and 01. */
/* This program is for testing 2x2 TV sets.

/* Prototype */
void board1();

```

```
void board2();
void board3();
void board4();

void main()
{
    char input;
    char stop=1;
    while (stop==1)
    {
        clrscr();
        printf("\n\n\t\t\t Please Enter your choice \n\n");
        printf("\t\t\t 1. Normal size          \n");
        printf("\t\t\t 2. Video wall 2x2   \n");
        printf("\t\t\t 3. Freeeze 2x2     \n");
        printf("\t\t\t 4. Horizontal zoom \n");
        printf("\t\t\t 5. Vertical zoom   \n");
        printf("\t\t\t 6. Exit              \n\n");
        printf("\t\t\t Enter your choice --> ");
        input=getch();
        printf("\a");
        switch(input)
        {
            case '1':
            {
                board1(0x00,0x3E,0x00,0x1F);
                board2(0x00,0x3E,0x00,0x1F);
                board4(0x00,0x3E,0x00,0x1F);
                board3(0x00,0x3E,0x00,0x1F);
            }
        }
    }
}
```

```
};break;
```

```
case '2':
```

```
{
```

```
board1(0x00,0x1C,0x00,0x0E);
```

```
board2(0x00,0x1D,0x00,0x0E);
```

```
board4(0x00,0x5D,0x24,0x0E);
```

```
board3(0x00,0x5C,0x24,0x0E);
```

```
};break;
```

```
case '3':
```

```
{
```

```
board1(0x00,0x1C,0x00,0x2E);
```

```
board2(0x00,0x1D,0x00,0x2E);
```

```
board4(0x00,0x5D,0x24,0x2E);
```

```
board3(0x00,0x5C,0x24,0x2E);
```

```
};break;
```

```
case '4':
```

```
{
```

```
board1(0x00,0x1C,0x00,0x1F);
```

```
board2(0x00,0x1D,0x00,0x1F);
```

```
board4(0x00,0x1D,0x00,0x1F);
```

```
board3(0x00,0x1C,0x00,0x1F);
```

```
};break;
```

```
case '5':
```

```
{
```

```
board1(0x00,0x3E,0x00,0x0E);
```

```

        board2(0x00,0x3E,0x00,0x0E);
        board4(0x00,0x7E,0x24,0x0E);
        board3(0x00,0x7E,0x24,0x0E);

    };break;

    case '6': stop=0;break;
}
}
}

void board1(data_1,data_2,data_3,data_4)
    int data_1,data_2,data_3,data_4;
{
    outportb(ctrl_port,0x6); /* 1001(0110) prevent chaing 2 bits in the same time */

    outportb(ctrl_port,0xB); /* 0000(1011) select board 1 */
    outportb(ctrl_port,0x3); /* 1000(0011) active selected board */

    /* at this state pin 5 of U30A must be 0 */

    outportb(ctrl_port,0x7); /* 1100(0111) Writing 74HC374 mode */

    outportb(ctrl_port,0xF); /* 0100(1111) select HC374 No. 1 */
    outportb(data_port,data_1); /* send data to HC374 No.1 */
    outportb(ctrl_port,0x7); /* 1100(0111) write data to HC374 */

    outportb(ctrl_port,0xE); /* 0101(1110) select HC374 No. 2 */
    outportb(data_port,data_2); /* send data to HC374 No.2 */
    outportb(ctrl_port,0x6); /* 1101(0110) write data to HC374 */

```

```
outportb(ctrl_port,0xC);/* 0111(1100) select HC374 No. 4 */
outportb(data_port,data_4); /* send data to HC374 No.4 */
outportb(ctrl_port,0x4);/* 1111(0100) write data to HC374 */
```

```
outportb(ctrl_port,0xD); /* 0110(1101) select HC374 No. 3 */
outportb(data_port,data_3); /* send data to HC374 No.3 */
outportb(ctrl_port,0x5);/* 1110(0101) write data to HC374 */
```

```
outportb(data_port,data_2);
```

```
}
```

```
void board2(data_1,data_2,data_3,data_4)
```

```
int data_1,data_2,data_3,data_4;
```

```
{
```

```
outportb(ctrl_port,0x4);/* 1011(0100) prevent changing 2 bit in the same time */
```

```
outportb(ctrl_port,0xA); /* 0001(1010) select board 2 */
```

```
outportb(ctrl_port,0x2); /* 1001(0010) active selected board */
```

```
/* at this state pin 5 of U30A must be 0 */
```

```
outportb(ctrl_port,0x06); /* 1101(0110) Writing 74HC374 mode */
```

```
outportb(ctrl_port,0xE);/* 0101(1110) select HC374 No. 2 */
```

```
outportb(data_port,data_2); /* send data to HC374 No.2 */
```

```
outportb(ctrl_port,0x6);/* 1101(0110) write data to HC374 */
```

```

outportb(ctrl_port,0xC);/* 0111(1100) select HC374 No. 4 */
outportb(data_port,data_4); /* send data to HC374 No.4 */
outportb(ctrl_port,0x4); /* 1111(0100) write data to HC374 */

```

```

outportb(ctrl_port,0xD); /* 0110(1101) select HC374 No. 3 */
outportb(data_port,data_3); /* send data to HC374 No.3 */
outportb(ctrl_port,0x5); /* 1110(0101) write data to HC374 */

```

```

outportb(ctrl_port,0xF); /* 0100(1111) select HC374 No. 1 */
outportb(data_port,data_1); /* send data to HC374 No.1 */
outportb(ctrl_port,0x7); /* 1100(0111) write data to HC374 */

```

```

outportb(data_port,data_4);

```

```

}

```

```

void board4(data_1,data_2,data_3,data_4)

```

```

    int data_1,data_2,data_3,data_4;

```

```

{

```

```

    outportb(ctrl_port,0x0); /*1011(0000) prevent chaing 2 bits in the same time */

```

```

    outportb(ctrl_port,0x8); /* 0011(1000) select board 4 */

```

```

    outportb(ctrl_port,0x0); /* 1011(0000) active selected board */

```

```

/* at this state pin 5 of U30A must be 0 */

```



```

outportb(ctrl_port,0x4); /* 1111(0100) Writing 74HC374 mode */
outportb(ctrl_port,0xC); /* 0111(1100) select HC374 No. 4 */
outportb(data_port,data_4); /* send data to HC374 No.4 */
outportb(ctrl_port,0x4); /* 1111(0100) write data to HC374 */

```

```

outportb(ctrl_port,0xD); /* 0110(1101) select HC374 No. 3 */
outportb(data_port,data_3); /* send data to HC374 No.3 */
outportb(ctrl_port,0x5); /* 1110(0101) write data to HC374 */

```

```

outportb(ctrl_port,0xF); /* 0100(1111) select HC374 No. 1 */
outportb(data_port,data_1); /* send data to HC374 No.1 */
outportb(ctrl_port,0x7); /* 1100(0111) write data to HC374 */

```

```

outportb(ctrl_port,0xE); /* 0101(1110) select HC374 No. 2 */
outportb(data_port,data_2); /* send data to HC374 No.2 */
outportb(ctrl_port,0x6); /* 1101(0110) write data to HC374 */

```

```

outportb(data_port,data_3);

```

```

}

```

```

void board3(data_1,data_2,data_3,data_4)

```

```

    int data_1,data_2,data_3,data_4;

```

```

{

```

```

    outportb(ctrl_port,0x4); /* 1011(0100) prevent changing 2 bits in the same time */

```

```

    outportb(ctrl_port,0x9); /* 0010(1001) select board 3 */

```

```

    outportb(ctrl_port,0x1); /* 1010(0001) active selected board */

```

```
/* at this state pin 5 of U30A must be 0 */
```

```
outportb(ctrl_port,0x5); /* 1110(0101) Writing 74HC374 mode */
```

```
outportb(ctrl_port,0xD); /* 0110(1101) select HC374 No. 3 */
```

```
outportb(data_port,data_3); /* send data to HC374 No.3 */
```

```
outportb(ctrl_port,0x5); /* 1110(0101) write data to HC374 */
```

```
outportb(ctrl_port,0xF); /* 0100(1111) select HC374 No. 1 */
```

```
outportb(data_port,data_1); /* send data to HC374 No.1 */
```

```
outportb(ctrl_port,0x7); /* 1100(0111) write data to HC374 */
```

```
outportb(ctrl_port,0xE); /* 0101(1110) select HC374 No. 2 */
```

```
outportb(data_port,data_2); /* send data to HC374 No.2 */
```

```
outportb(ctrl_port,0x6); /* 1101(0110) write data to HC374 */
```

```
outportb(ctrl_port,0xC); /* 0111(1100) select HC374 No. 4 */
```

```
outportb(data_port,data_4); /* send data to HC374 No.4 */
```

```
outportb(ctrl_port,0x4); /* 1111(0100) write data to HC374 */
```

```
outportb(data_port,data_1);
```

```
}
```

## ประวัติผู้เขียน

นายชยา ลิมจิตติ เกิดวันที่ 10 ธันวาคม พ.ศ. 2513 ที่อำเภอเมือง จังหวัดสงขลา สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตรบัณฑิต เกียรตินิยมอันดับสอง ภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2534 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ.2535

