



ریزپردازنده

به نام او



گزارش آزمایش 7

امیرمحمد رادمهر 810195394

میلاذ عسگری 810195432

مهدی ارجمند 810195510

قسمت اول

در ابتدا یک تابع به نام **Reverse_Byte** تعریف میکنیم و از نامش مشخص است که چه می کند.

```
uint8_t reverse_byte(uint8_t x){
    static const uint8_t table[] = {
        0x00, 0x80, 0x40, 0xc0, 0x20, 0xa0, 0x60, 0xe0,
        0x10, 0x90, 0x50, 0xd0, 0x30, 0xb0, 0x70, 0xf0,
        0x08, 0x88, 0x48, 0xc8, 0x28, 0xa8, 0x68, 0xe8,
        0x18, 0x98, 0x58, 0xd8, 0x38, 0xb8, 0x78, 0xf8,
        0x04, 0x84, 0x44, 0xc4, 0x24, 0xa4, 0x64, 0xe4,
        0x14, 0x94, 0x54, 0xd4, 0x34, 0xb4, 0x74, 0xf4,
        0x0c, 0x8c, 0x4c, 0xcc, 0x2c, 0xac, 0x6c, 0xec,
        0x1c, 0x9c, 0x5c, 0xdc, 0x3c, 0xbc, 0x7c, 0xfc,
        0x02, 0x82, 0x42, 0xc2, 0x22, 0xa2, 0x62, 0xe2,
        0x12, 0x92, 0x52, 0xd2, 0x32, 0xb2, 0x72, 0xf2,
        0x0a, 0x8a, 0x4a, 0xca, 0x2a, 0xaa, 0x6a, 0xea,
        0x1a, 0x9a, 0x5a, 0xda, 0x3a, 0xba, 0x7a, 0xfa
    }
```

تابع تایمر و تابعی که ماتریس را نشان میدهد به صورت زیر تعریف میشود

```
void Timer1ISR(void){
    timer++;
    fps++;
    if(timer >= 10){
        row_flag = TRUE;
        timer = 0;
    }
    if(fps >= 1250){
        next_frame = TRUE;
        fps = 0;
    }
}

void matrix_depict(uint8_t data[8]){
    uint8_t row_holder = 7;
    if(row_flag){
        row_flag = FALSE;

        LPC_GPIO1 -> FIOCLR = (uint32_t) (0xffff<<16);
        uint8_t row = (uint8_t) (1<<row_holder);
        LPC_GPIO1 -> FIOSET = (uint32_t) (reverse_byte(data[row_holder]) << 24);

        LPC_GPIO1 -> FIOSET = (uint32_t) (row<<16);
        row_holder--;
        if(row_holder == 0)
            row_holder = 7;
    }
}

.. ..
```

در ادامه همانطور که مشخص است در شکل زیر با استفاده از قطعه کد زیر که کارت حافظه را میخوانیم و نمرات را میتوان مشاهده کرد

```
//////////////////////////////////
char ch;
fileConfig_st *source_pointer,*dest_pointer;
fileInfo fileList;
uint8_t status,sdcardType;

status = SD_Init(&sdcardType);
source_pointer = FILE_Open("GROUP-04.txt",READ,&status);
while(1){
    ch = FILE_GetCh(source_pointer);
    if(ch == EOF) break;
    UART2_TxChar(ch);
}
```

در انتها نتایج را میتوان در شکل زیر مشاهده کرد.

```
Serial port COM5 opened
0: 20 / (20)
1: 22 / (20+2)
2: 24 / (20+4)
3: 22 / (20+10)
4: 23 / (20+6)
5: 00 / (20+8)
Serial port COM5 closed
```

قسمت دوم

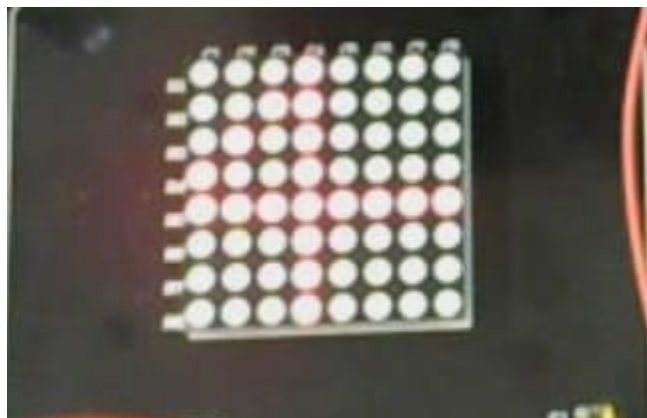
هدف این قسمت نمایش شماره گروه خود بر روی ماتریس تعریف شده است.

با استفاده از تابع زیر این کار انجام میشود

```
GP 04
uint8_t gp_num[8] = {0x18,0x14,0x12,0xff,0x10,0x10,0x10,0x10};

while(1){
    matrix_depict(gp_num);
}
```

و در انتها نتیجه به صورت زیر نمایان است.



قسمت سوم

در این قسمت برای نمایش انیمیشن از قطعه کد زیر استفاده میکنیم

```
////////// Q3 ////////////
char ch;
uint16_t cnt = 0;
uint8_t animation[272];
fileConfig_st *source_pointer;
uint8_t status,sdcardType;

status = SD_Init(&sdcardType);

source_pointer = FILE_Open("mpl.anm",READ,&status);

while(1)
{
    ch = FILE_GetCh(source_pointer);
    if(ch == EOF)break;
    animation[cnt] = ch;
    cnt++;
}
FILE_Close(source_pointer);

uint8_t frame_cnt = 0 , frame[8];
cnt = 0;

while(1)
{
    if(next_frame)
    {
        next_frame = FALSE;

        for(uint8_t i; i<8;i++)
            frame[i] = animation[frame_cnt*8+i];

        frame_cnt++;
        if(frame_cnt == 35)
            frame_cnt = 0;
    }
    matrix_depict(frame);
}
```

نتیجه این قسمت به صورت یک فیلم در فایل زیپ قرار داده شده است
در کد بالا اول با یک تابع آن را خوانده و سپس آنرا **close** میکنیم.