

## 5. 4K210 录音机

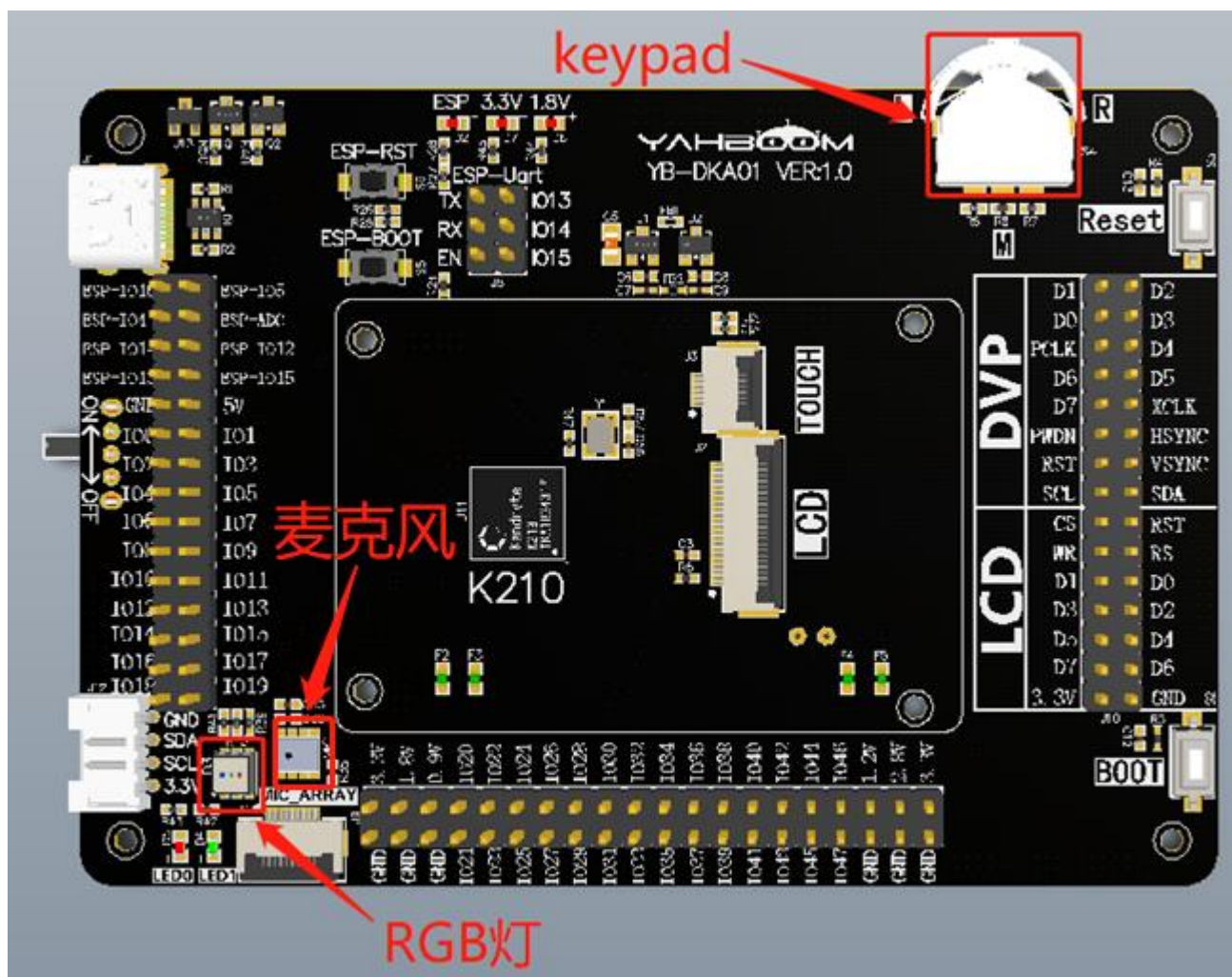
### 一、实验目的

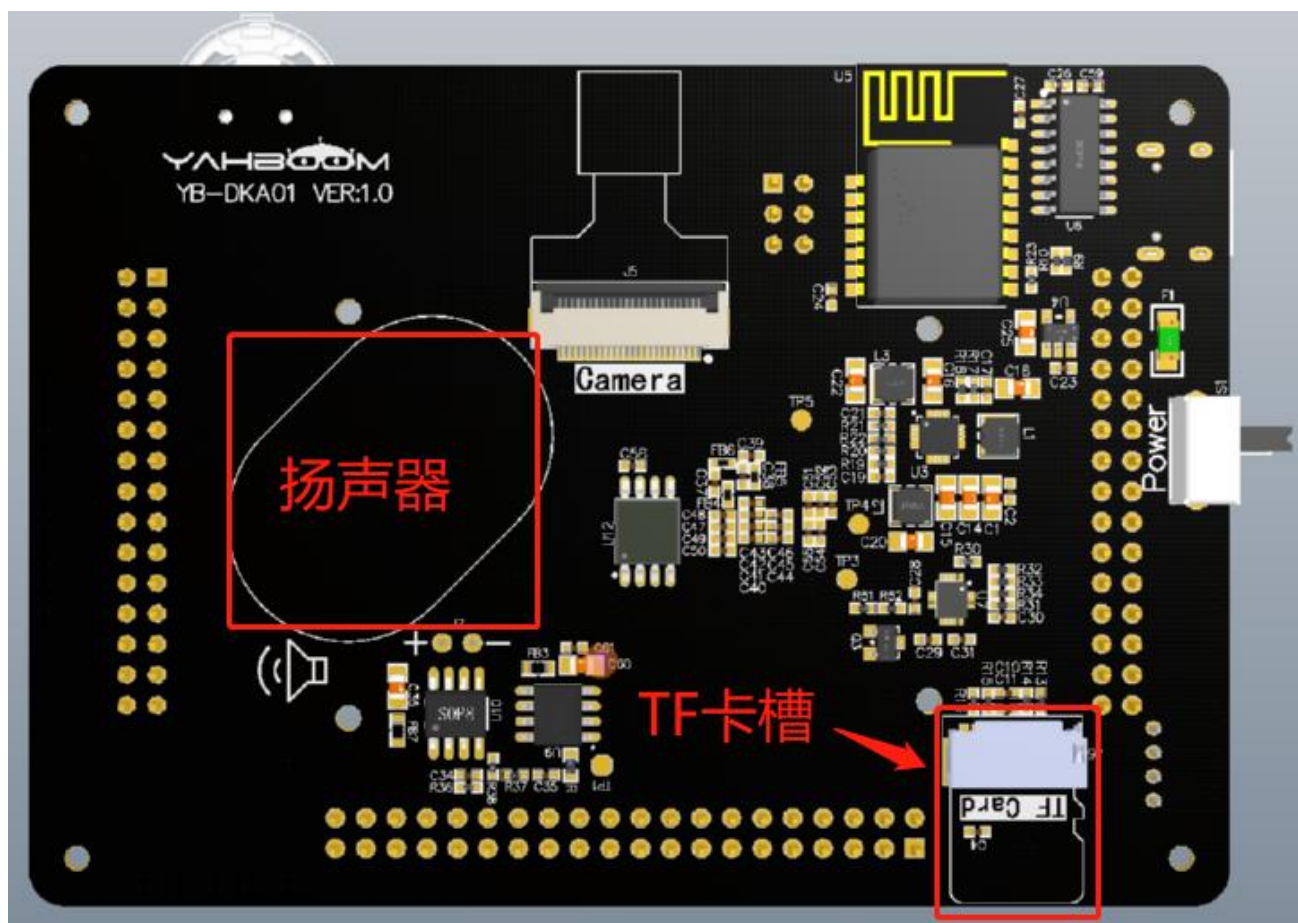
本节课主要学习 K210 如何打开麦克风录音，保存到 TF 卡中，再从 TF 卡读取录音的内容，经过扬声器播放出来。

### 二、实验准备

#### 1. 实验元件

麦克风、拨轮开关 keypad、RGB 灯、扬声器、TF 卡槽和 TF 卡





### 三、实验原理

各个元器件的工作原理前面的基础课程已经讲过，现在来说说整个代码的设计思路，录音的过程是先通过 I2S 通讯读取麦克风的缓存数据，每次读一部分数据就会直接写入到 TF 卡中，然后保存录音文件。播放录音则是从 TF 读取 wav 音频文件，经过解析文件得到数据，再通过 I2S 发送给 DAC 功放电路转化信号并放大，最后扬声器播放声音。

wav 格式文件储存原理：wav 格式文件是微软公司开发的一种声音文件格式，它符合 RIFF 文件规范，标准格式的 wav 文件与 CD 格式一样采用 44.1K 采样率，音质与 CD 相近，所以被广泛应用，wav 文件分为两个部分，第一部分为 wav 头文

件，记录 wav 文件的格式和信息等，第二部分是实际储存音频的 PCM(脉冲编码调制)数据。



PCM 数据与实际录音的数据相关，只要按照读取的数据存入即可。

wav 头文件内容请参考以下表格：

偏移地址	字节	数据内容	备注
00H~03H	4	RIFF	资源交换文件标志
04H~07H	4	Wav 数据大小	从下个地址开始到文件尾的总字节数
08H~0BH	4	WAVE	WAV 文件标志
0CH~0FH	4	fmt	波形格式标志，最后一位是空格。
10H~13H	4	过滤字节	一般为 00000010H，若为 00000012H 则说明数据头携带附加信息。
14H~15H	2	格式种类	值为 1 时，表示数据为线性 PCM 编码
16H~17H	2	通道数	单声道为 1，双声道为 2
18H~1BH	4	采样频率	标准采样率为 44.1K
1CH~1FH	4	数据传输速率	波形数据传输速率（每秒平均字节数）
20H~21H	2	块对齐	数据块对齐长度
22H~23H	2	PCM 位宽	每次采样时，每个通道的样本大小（数量）单位 bit，注意不是字节，是比特，1 字节=8 比特
24H~27H	4	data	表示数据开始
28H~2BH	4	data 数据量	从以下地址开始到结束的总字节数

#### 四、实验过程

1. 首先根据上面的硬件连接引脚图，K210 的硬件引脚和软件功能使用的是 FPIOA 映射关系。

```
static void hardware_init(void)
{
    /* TF卡 */
    //io26--miso--d1
    //io27--clk---sclk
    //io28--mosi--d0
    //io29--cs----cs
    fpioa_set_function(PIN_TF_MISO, FUNC_TF_SPI_MISO);
    fpioa_set_function(PIN_TF_CLK,  FUNC_TF_SPI_CLK);
    fpioa_set_function(PIN_TF_MOSI, FUNC_TF_SPI_MOSI);
    fpioa_set_function(PIN_TF_CS,   FUNC_TF_SPI_CS);

    /* 扬声器 */
    fpioa_set_function(PIN_I2S_WS,  FUNC_I2S2_WS);
    fpioa_set_function(PIN_I2S_DA,  FUNC_I2S2_OUT_D1);
    fpioa_set_function(PIN_I2S_BCK, FUNC_I2S2_SCLK);

    /* 麦克风 */
    fpioa_set_function(PIN_MIC0_WS,  FUNC_I2S0_WS);
    fpioa_set_function(PIN_MIC0_DATA, FUNC_I2S0_IN_D0);
    fpioa_set_function(PIN_MIC0_SCK, FUNC_I2S0_SCLK);

    /* keypad */
    fpioa_set_function(PIN_KEYPAD_LEFT,  FUNC_KEYPAD_LEFT);
    fpioa_set_function(PIN_KEYPAD_MIDDLE, FUNC_KEYPAD_MIDDLE);
    fpioa_set_function(PIN_KEYPAD_RIGHT,  FUNC_KEYPAD_RIGHT);

    /* RGB灯 */
    fpioa_set_function(PIN_RGB_R, FUNC_RGB_R);
    fpioa_set_function(PIN_RGB_G, FUNC_RGB_G);
    fpioa_set_function(PIN_RGB_B, FUNC_RGB_B);
}
```

2. 设置系统 PLL 时钟。

```
/* 系统时钟设置 */
sysctl_pll_set_freq(SYSCTL_PLL0, 320000000UL);
sysctl_pll_set_freq(SYSCTL_PLL1, 160000000UL);
sysctl_pll_set_freq(SYSCTL_PLL2, 45158400UL);
//uarths_init();
```

3. 初始化系统中断，使能系统全局中断，初始化 keypad 和 RGB 灯。

```

hardware_init();    // 硬件引脚初始化
dmac_init();        // dmac初始化
plic_init();        // 中断初始化
sysctl_enable_irq(); // 使能全局中断
keypad_init();       // keypad初始化
rgb_init(EN_RGB_ALL); // LED灯初始化

```

4. 要使用 TF 卡前必须初始化，并且判断 TF 卡的格式。

```

if(sdcard_test())
{
    printf("SD card err\r\n");
    return -1;
}
if(fs_test())
{
    printf("FAT32 err\r\n");
    return -1;
}

```

5. 读取 keypad 的状态，如果是中间的键被按下，则开始录音。

```

case EN_KEY_MIDDLE_DOWN: //录音
    printf("EN_KEY_MIDDLE_DOWN\r\n");
    if(g_record_state != 1)
    {
        bsp_recorder_start(_T("0:REC.wav"));
        g_record_state = 1;
    }

    break;

```

6. keypad 向左拨动保存录音文件。

```
//获取按键值
key_value = key_out_fifo();
switch(key_value)
{
    case EN_KEY_LEFT_DOWN: //停止和保存录音
        printf("EN_KEY_LEFT_DOWN\r\n");
        if(g_record_state == 1)
        {
            bsp_recorder_stopsave("0:REC.wav");
            g_record_state = 2;
        }
        app_rgb_red_state(LIGHT_OFF); //关闭红灯
        break;

    case EN_KEY_MIDDLE_DOWN: //录音
        printf("EN_KEY_MIDDLE_DOWN\r\n");
        if(g_record_state != 1)
        {
            bsp_recorder_start(_T("0:REC.wav"));
            g_record_state = 1;
        }

        break;
}
```

7. keypad 向右拨动播放录音。

```
case EN_KEY_RIGHT_DOWN: //播放录音（仅在非录音状态下有效）
    printf("EN_KEY_RIGHT_DOWN\r\n");
    app_rgb_blue_state(LIGHT_ON);
    if(g_record_state == 2)
    {
        bsp_play_wav(_T("0:REC.wav"));
        //g_record_state = 2;
    }
    break;
```

8. 编译调试，烧录运行

把本课程资料中的 record\_play 复制到 SDK 中的 src 目录下,然后进入 build 目录，运行以下命令编译。

```
cmake .. -DPROJ=record_play -G "MinGW Makefiles"
make
```



```
[ 80%] Linking C executable record_play
Generating .bin file ...
[100%] Built target record_play
PS C:\K210\SDK\kendryte-standalone-sdk-develop\build>
```

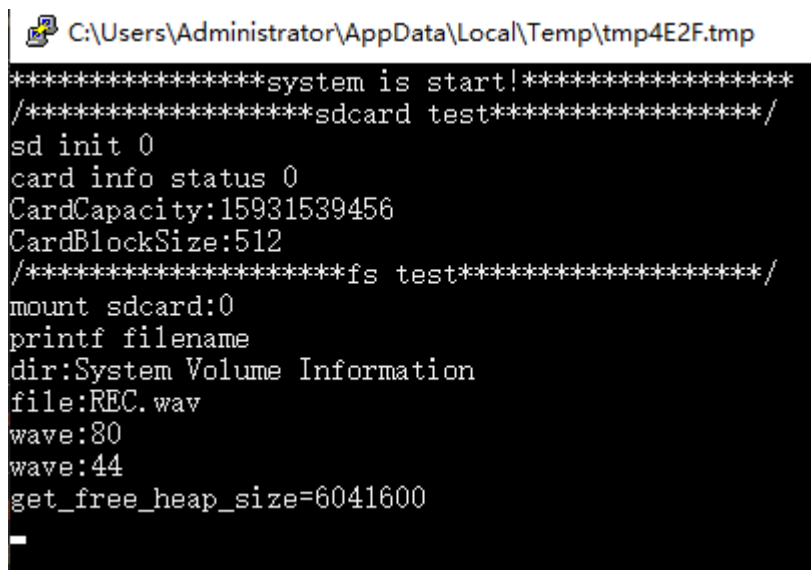
编译完成后，在 build 文件夹下会生成 record\_play.bin 文件。

使用 type-C 数据线连接电脑与 K210 开发板，打开 kflash，选择对应的设备，再将程序固件烧录到 K210 开发板上。

## 五、实验现象

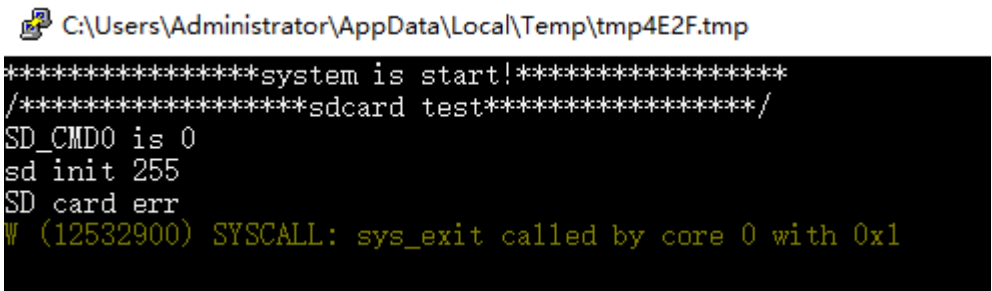
烧录完成固件后，系统会弹出一个终端界面，如果没有弹出终端界面的可以打开串口助手显示调试内容。

终端会打印 SD 卡的信息。



```
C:\Users\Administrator\AppData\Local\Temp\tmp4E2F.tmp
*****system is start!*****
/*****sdcard test*****/
sd init 0
card info status 0
CardCapacity:15931539456
CardBlockSize:512
/*****fs test*****/
mount sdcard:0
printf filename
dir:System Volume Information
file:REC.wav
wave:80
wave:44
get_free_heap_size=6041600
_
```

如果出现以下图片的情况，请检查是否 TF 卡正确插入卡槽，再把 K210 开发板的电源断开后重新打开即可。



```
C:\Users\Administrator\AppData\Local\Temp\tmp4E2F.tmp
*****system is start!*****
/*****sdcard test*****/
SD_CMD0 is 0
sd init 255
SD card err
W (12532900) SYSCALL: sys_exit called by core 0 with 0x1
```

当我们按一下 keypad 中间的按键时,RGB 红灯开始闪烁,此时已经开始录音,可以对着麦克风讲话;然后把 keypad 向左拨动一下,红灯熄灭,录音保存;最后把 keypad 向右拨动一下,播放刚刚录音的内容。

## 六、实验总结

1. 录音的文件是保存到 TF 卡中的.wav 文件,可以使用读卡器放到电脑上播放。
2. 有时候烧录完程序后,会出现 TF 卡读取失败的问题,重新拨动 K210 开发板的电源开关,重新上电就可以。