

## 目录

1、 介绍.....	1
2、 bootloader 实现.....	1
3、 运行效果.....	4

## 1、 介绍

之前文档曾使用 sd 卡来升级程序。这种方法每次都要插 sd 卡到电脑，将程序拷到 sd 卡里，然后再插到 mcu 的 sd 卡槽才能更新程序。这种更新方法似乎有些麻烦，所以我们实现一种更方便的办法，它不再需要插拔 sd 卡。使用 mcu 的 usb 功能，将 mcu 识别为存储 U 盘，存储介质则是 sd 卡。当我们需要更新程序，只要将 mcu 的 usb 接口连接到电脑，电脑识别出以后，将需要烧录的程序复制进去，然后 bootloader 就会识别该文件，然后升级应用程序。Bootloader 检测的是文件的改动而不是文件是否存在。程序本身逻辑是一旦 sd 卡产生了读写操作，才检测是否存在需要的更新文件（a000.bin 文件）。也就是说，如果 sd 卡里已经有 a000.bin 文件，是不会更新应用程序，只有将这个 a000.bin 用其他的 a000.bin 覆盖，才会执行更新应用程序的操作。

## 2、 bootloader 实现

SD 卡与单片机连接原理图如下，板子使用 SDHC 模块来与 SD 卡通信。

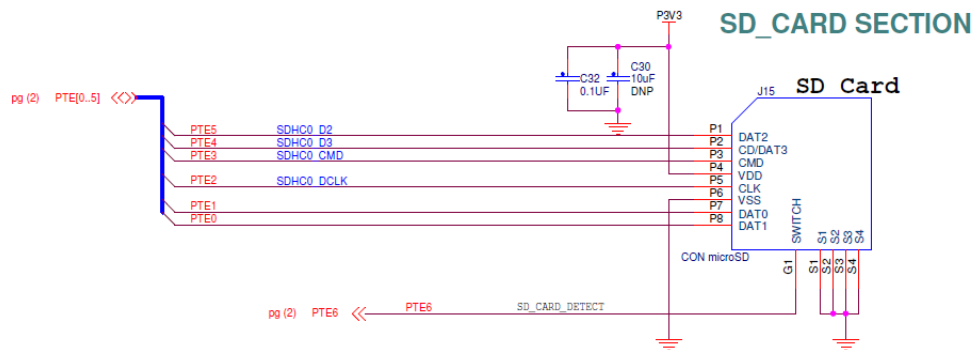


图 1.SD 卡原理图

我们使用 2.6.0 版本的 FRDM-K64F 固件，你可以从这个链接下载固件。  
“mcuxpresso.nxp.com”。

Usb 连接图如下

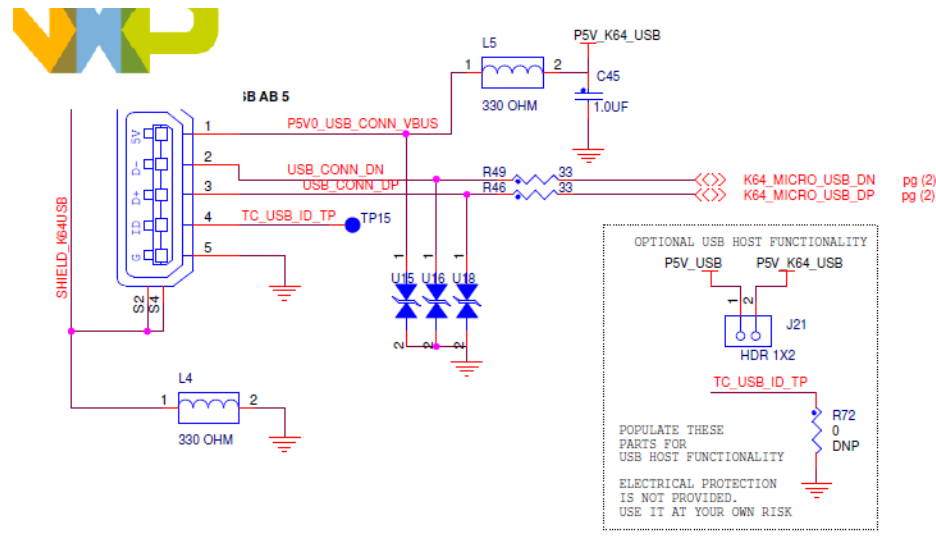


图 2.USB 连接图

Bootloader 使用了 SDHC, fatfs 文件系统, usb, flash 操作, 所以我们应该添加一些文件来支持, 我们的代码基于 usb 例程下的 usb\_device\_msc\_sdcard\_lite, 将所需文件添加到该工程。

主代码中, 主要就是将 usb 初始化完成, sd 卡初始化完成, 挂载文件系统完成, 然后电脑会与 mcu 用 usb 进行通讯, 最后电脑就会识别出一个盘符

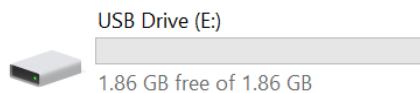


图 3.盘符

其他关于如何更新代码到 flash 的具体解释参考 <https://community.nxp.com/docs/DOC-344903>

检测文件改动代码, 通过定义一个标志 wrFlag, 当将文件放入 U 盘, 会调用 write10 命令, 此时将标志位置 1 表示有文件改动。

```

usb_status_t USB_DeviceMscProcessUfiCommand(usb_device_msc_struct_t *mshandle)
{
    usb_status_t error = kStatus_USB_Error;
    usb_device_msc_ufi_struct_t *ufi = NULL;
    ufi = &mshandle->mscUfi;
    if (USB_DEVICE_MSC_REQUEST_SENSE_COMMAND != mshandle->mscCbW->cbwcb[0])
    {
        ufi->requestSense->senseKey = USB_DEVICE_MSC_UFI_NO_SENSE;
        ufi->requestSense->additionalSenseCode = USB_DEVICE_MSC_UFI_NO_SENSE;
        ufi->requestSense->additionalSenseQualifier = USB_DEVICE_MSC_UFI_NO_SENSE;
    }
    ufi->thirteenCase.hostExpectedDataLength = mshandle->mscCbW->dataTransferLength;
    ufi->thirteenCase.hostExpectedDirection = (uint8_t)(mshandle->mscCbW->flags >> USB_DEVICE_MSC_CBW_DI);
    /*The first byte of all ufi command blocks shall contain an Operation Code, refer to ufi spec*/

    switch (mshandle->mscCbW->cbwcb[0])
    {
        /* ufi command operation code*/
        case USB_DEVICE_MSC_INQUIRY_COMMAND: /*operation code : 0x12*/
            error = USB_DeviceMscUfiInquiryCommand(mshandle);
            break;
        case USB_DEVICE_MSC_READ_10_COMMAND: /*operation code : 0x28 */
        case USB_DEVICE_MSC_READ_12_COMMAND: /*operation code : 0xA8 */
            error = USB_DeviceMscUfiReadCommand(mshandle);
            break;
        case USB_DEVICE_MSC_REQUEST_SENSE_COMMAND: /*operation code : 0x03*/
            error = USB_DeviceMscUfiRequestSenseCommand(mshandle);
            break;
        case USB_DEVICE_MSC_TEST_UNIT_READY_COMMAND: /*operation code : 0x00 */
            error = USB_DeviceMscUfiTestUnitReadyCommand(mshandle);
            break;
        case USB_DEVICE_MSC_WRITE_10_COMMAND: /*operation code : 0x2A */
        case USB_DEVICE_MSC_WRITE_12_COMMAND: /*operation code : 0xAA */
            error = USB_DeviceMscUfiWriteCommand(mshandle);
            break;
        case USB_DEVICE_MSC_PREVENT_ALLOW_MEDIUM_REM_COMMAND: /*operation code :0x1E */
            error = USB_DeviceMscUfiPreventAllowMediumCommand(mshandle);
            break;
    }
}

```

图 4.修改标志位

当检测到该标志以后，打开“a000.bin”文件，开始升级应用程序，然后反初始化设备，最后跳转到应用程序。

```

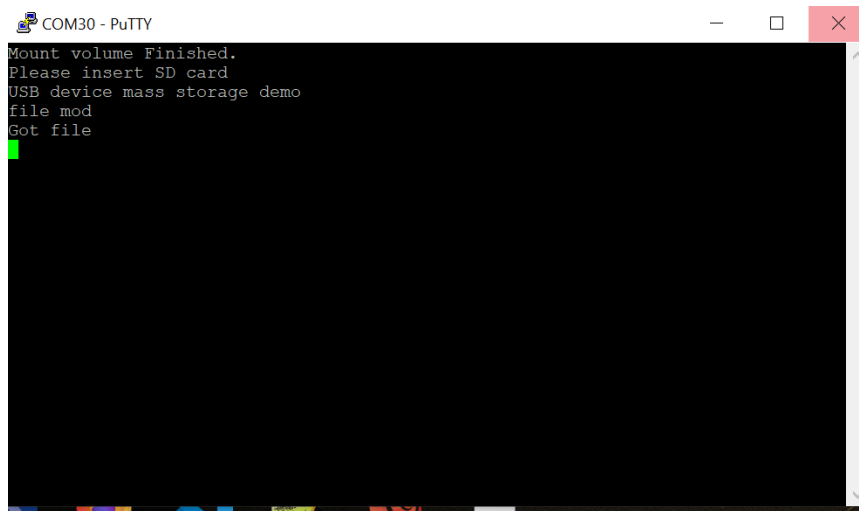
FRESULT USB_DeviceMscApp(void)
{
    FRESULT error = FR_OK;
    if(wrFlag == 1)
    {
        SDMMCHOST_Delay(300); //because When put file into the disk, the commar
        wrFlag = 0;
        PRINTF("file mod\r\n");
        error = f_open(&g_fileObject, _T("/a000.bin"), (FA_WRITE | FA_READ));
        if(FR_NO_FILE == error)
        {
            PRINTF("No file\r\n");
        }
        if(FR_OK == error)
        {
            PRINTF("Got file\r\n");
            update_bootloader(&g_fileObject);
            f_close(&g_fileObject);
            USB_DeviceStop(g_msc.deviceHandle);
            USB_DeviceDeinit(g_msc.deviceHandle);
            //jump to App();
            deinit();
            SD_Deinit(&g_sd);
            static void (*farewellBootloader)(void) = 0;
            farewellBootloader = (void (*)(void))(APP_VECTOR_TABLE[1]);
            __set_MSP(APP_VECTOR_TABLE[0]);
            __set_PSP(APP_VECTOR_TABLE[0]);
            SCB->VTOR = (uint32_t)APP_VECTOR_TABLE;
            farewellBootloader();
            while(1);
        }
        // f_close(&g_fileObject);
    }
    /*TO DO*/
    /*add user code*/
    return error;
}

```

### 3、运行效果

- 1、下载 bootloader
- 2、我们修改了 led\_blinky 代码作为升级 app，将生成的 bin 文件改成 a000.bin 文件放入 u 盘

此时串口打印的消息



- 3、程序开始运行，led 灯开始闪烁

