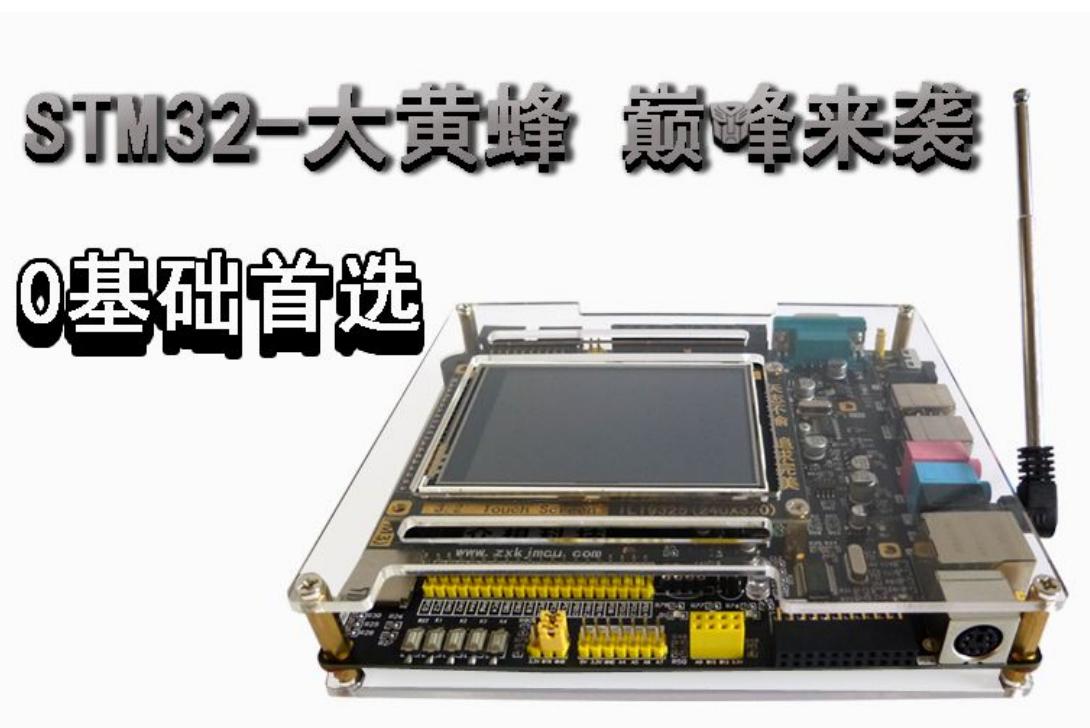


学 ARM 从 STM32 开始

STM32 开发板库函数教程—实战篇



官方网站: <http://www.zxkjmcu.com>

官方店铺: <http://zxkjmcu.taobao.com>

官方论坛: <http://bbs.zxkjmcu.com>

刘洋课堂: <http://school.zxkjmcu.com>

4.36 STM32 SD 卡读写数据块实验

4.36.1 概述

4.36.1.1 SD卡数据传输模式原理

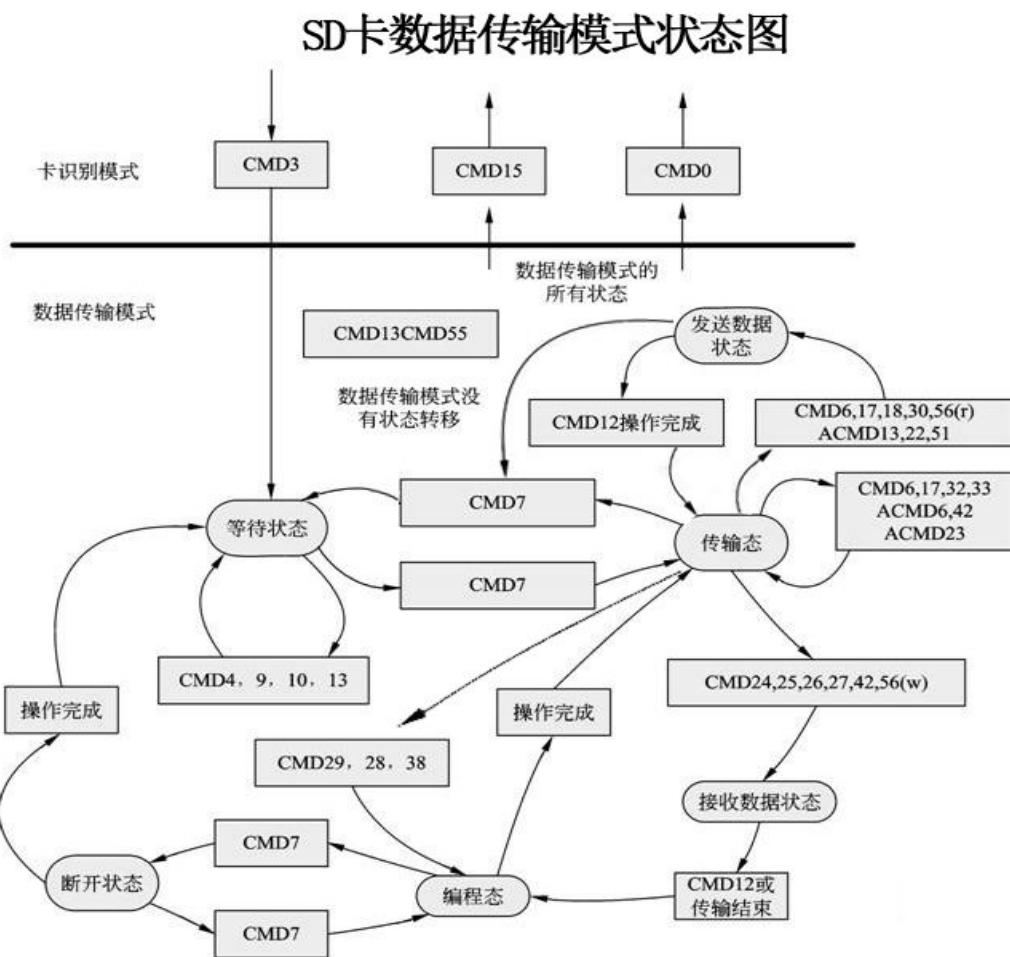


图 4.36.1 SD 卡数据传输模式状态图

4.36.2 实验目的

参考《4.35 STM32 外设篇-SD 存储卡指令协议及初始化程序设计》。

在这个基础上增加了读写 SD 卡块的功能实验，写入"LY-STM32 主讲人：刘

洋 视频教程下载地址 www.zxkjmcu.com"字符串来验证程序设计的正确性。

4. 36. 3 大黄蜂实验板触摸屏硬件设计

参考《4.35 STM32 外设篇-SD 存储卡指令协议及初始化程序设计》

4. 36. 4 软件设计

4. 36. 4. 1 软件设计说明

参考《4.35 STM32 外设篇-SD 存储卡指令协议及初始化程序设计》。

采用中断方式节省 CPU 资源。

4. 36. 5 GPIO 管脚电平控制函数

在主程序中采用 while(1) 循环语句，采用中断方式以节省 CPU 资源。

参考《4.35 STM32 外设篇-SD 存储卡指令协议及初始化程序设计》。

```
while(1);
```

4. 36. 6 main.c 文件里的内容是

```
#include "pbdata.h"

void RCC_Configuration(void);
void GPIO_COM_Configuration(void);
void NVIC_Configuration(void);
void USART_Configuration(void);

SD_Error Status=SD_OK;
extern SD_CardInfo SDCardInfo;

int fputc(int ch,FILE *f)
{
    USART_SendData(USART1, (u8)ch);
```

```
while(USART_GetFlagStatus(USART1, USART_FLAG_TXE)==RESET);  
    return ch;  
}  
  
int main(void)  
{  
    //临时中间变量，为打印输出做准备  
    u8 Buf[]="LY-STM32 主讲人：刘洋 视频教程下载地址 www.zxkjmcu.com";  
  
    RCC_Configuration(); //系统时钟初始化  
    GPIO_COM_Configuration(); //端口初始化  
    USART_Configuration();  
    NVIC_Configuration();  
  
    Status=SD_Init();  
  
    if(Status==SD_OK)  
    {  
        printf("SD 卡初始化成功!\r\n");  
    }  
    else  
    {  
        printf("SD 卡初始化失败!\r\n");  
    }  
  
    printf("制造商 ID 号: %d\r\n", SDCardInfo.SD_cid.ManufacturerID);  
    printf("卡的类型: %d\r\n", SDCardInfo.CardType);  
    printf("卡的容量: %d\r\n", SDCardInfo.CardCapacity);  
    printf("块的大小: %d\r\n", SDCardInfo.CardBlockSize);  
  
    if(Status==SD_OK)  
    {  
        Status=SD_WriteBlock(Buf, 0, 512);  
        //等待 DMA 传输结束  
        Status=SD_WaitWriteOperation();  
        //等待 SDIO 到 SD 卡传输结束  
        while(SD_GetStatus() !=SD_TRANSFER_OK);  
    }  
  
    memset(Buf, 0, sizeof(Buf));  
  
    if(Status==SD_OK)  
    {  
        Status=SD_ReadBlock(Buf, 0, 512);  
    }
```

```
//等待 DMA 传输结束
Status=SD_WaitReadOperation();
//等待 SDIO 到 SD 卡传输结束
while(SD_GetStatus() !=SD_TRANSFER_OK);

printf("%s\r\n", Buf);//打印数组
}

printf("完成! ");

while(1);
}

void RCC_Configuration(void)
{
    SystemInit(); //72m
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA, ENABLE);
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_USART1, ENABLE);
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_AFIO, ENABLE);
}

void GPIO_COM_Configuration(void)
{
    GPIO_InitTypeDef GPIO_InitStructure;
    //COM
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_9;//TX
    GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz;
    GPIO_InitStructure.GPIO_Mode=GPIO_Mode_AF_PP;
    GPIO_Init(GPIOA, &GPIO_InitStructure);

    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_10;//RX
    GPIO_InitStructure.GPIO_Mode=GPIO_Mode_IN_FLOATING;
    GPIO_Init(GPIOA, &GPIO_InitStructure);
}

void NVIC_Configuration(void)
{
    NVIC_InitTypeDef NVIC_InitStructure;

    NVIC_PriorityGroupConfig(NVIC_PriorityGroup_1);

    NVIC_InitStructure.NVIC_IRQChannel = USART1_IRQn;
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 1;
```

```
NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
NVIC_Init(&NVIC_InitStructure);
}

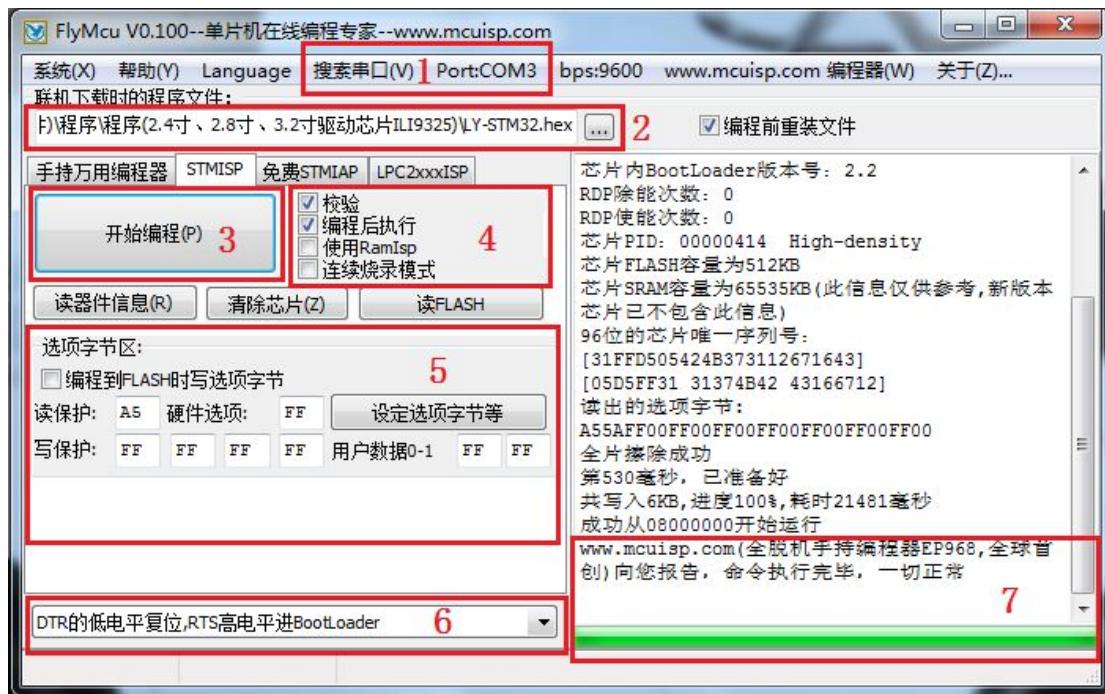
void USART_Configuration(void)
{
    USART_InitTypeDef USART_InitStructure;

    USART_InitStructureUSART_BaudRate=9600;
    USART_InitStructureUSART_WordLength=USART_WordLength_8b;
    USART_InitStructureUSART_StopBits=USART_StopBits_1;
    USART_InitStructureUSART_Parity=USART_Parity_No;
    USART_InitStructureUSART_HardwareFlowControl=USART_HardwareFlowContr
ol_None;
    USART_InitStructureUSART_Mode=USART_Mode_Rx|USART_Mode_Tx;

    USART_Init(USART1, &USART_InitStructure);
    USART_ITConfig(USART1, USART_IT_RXNE, ENABLE);
    USART_Cmd(USART1, ENABLE);
    USART_ClearFlag(USART1, USART_FLAG_TC);
}
```

4. 36. 7 程序下载

请根据下图所指向的 7 个重点区域配置。其中（1）号区域根据自己机器的实际情况选择，我的机器虚拟出来的串口号是 COM3。2）号区域请自己选择程序所在的文件夹。（7）号区域当程序下载完后，进度条会到达最右边，并且提示一切正常。（4、5、6）号区域一定要按照上图显示的设置。当都设置好以后就可以直接点击（3）号区域的开始编程按钮下传程序了。



本节实验的源代码在光盘中: (LY-STM32 光盘资料\1. 课程\2, 外设篇

\26. SD 卡读写数据块实验\程序)

4. 36. 8 实验效果图

4. 36. 8. 1 读写 SD 卡块实验效果

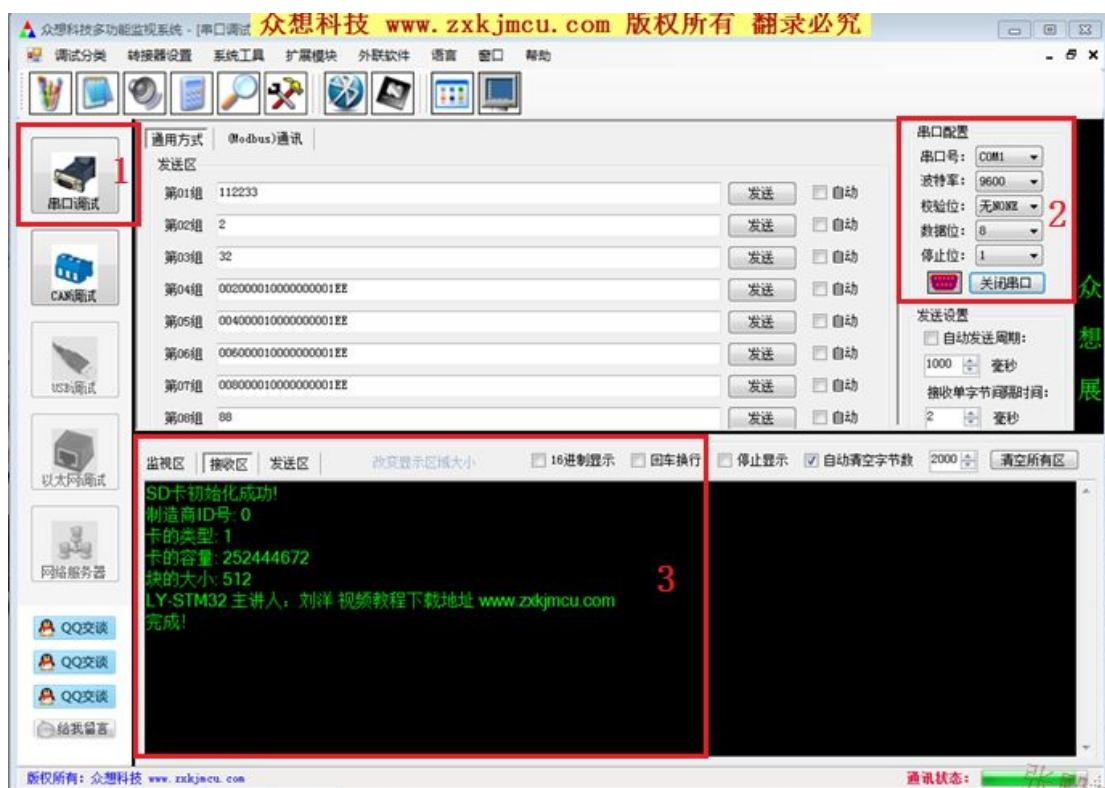


图 4.36.8 SD 卡读写块实验效果图