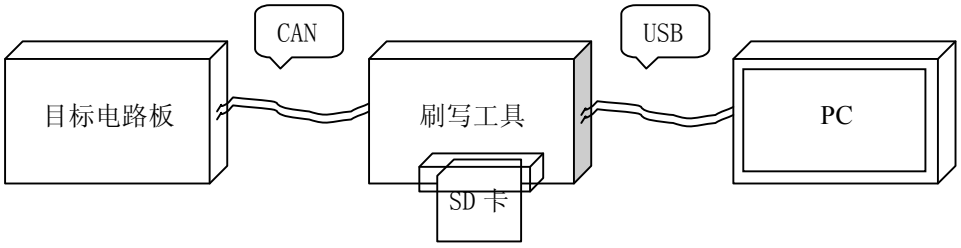
KESENS
凯晟汽车系统(上海)有限公司
地址：上海市浦东新区康桥东路 1365 弄 8 号楼
电话：21-68183636 传真：21-20980596 邮编：201315

SPC563M64 芯片刷写工具方案

刷写工具的功能

1. 通过 CAN 总线与 SPC563M64 的电路板进行通信，通过 USB 总线与上位机进行通信；
2. 可以读取 SD 卡上的数据并将其写入 SPC563M64 的电路板上；
3. 可以通过操作上位机软件将指定数据写入 SPC563M64 的电路板上；
4. 支持自定义密码；
5. 支持对 SPC563M64 内部 FLASH 的擦除、读出、检测和比较操作。



计划实施步骤

1. 根据 uFlasher 刷写工具的 CAN 线通信内容，用 STM32F207 开发板实现对 ECU 的脱机刷写。
2. 在 STM32F207 开发板上实现使用 SD 卡的脱机刷写。
3. 使 STM32F207 开发板通过 USB 线与上位机通信，并开发上位机软件，实现联机刷写。
4. 开发对 SPC563M64 内部 FLASH 的擦除、读出、检测和比较功能。

HEX 文件格式

HEX 文件是预先下载到 SPC563M64 中的 BootLoader 文件所采用的格式。该文件中的每行记录由长度、地址、类型、数据、校验五部分组成。

1. 长度：
   数据域的长度，最大不超过 0xFF。
2. 地址：
   16 位地址，实际使用时要与高 16 位的基址相加组成 32 位地址。基址默认为 0。
3. 类型：
   00 为数据，01 为文件结束；
   02 为 386 的 20 位实模式基址跳转，该值*16 后与此后所有 00 类型记录的地址相加；
   03 用于设置 CPU 的 CS 和 IP 寄存器；
   04 用于设置 32 位地址的高 16 位，并对此后所有 00 类型记录有效，直到下一个 04 跳转记录出现为止。

05 用于设置 CPU 的 EIP 寄存器。

在 32 位文件格式中，只使用上述 00、01、04、05 四种类型。

4. 校验：

1 个字节，CRC = 0xFF - (记录长度 + 存储地址 + 类型 + 数据)。

例如：

第一行 :02 0000 04 4000 BA　　用来设置地址 4000 0100 中的高 16 位。

第二行 :20 0100 00 7000 0000...　　用来在地址 4000 0100 处开始写入数据 7000 0000...

最末行 :04 0000 05 4000 0E00 A9　　跳转到 4000 0E00 执行？

:00 0000 01 FF　　结束标志。

S19 文件格式

S19 文件是向 SPC563M64 中下载的程序文件格式，该文件中的每行记录由类型、长度、地址、数据、校验五部分组成。

1. 类型：

S0 表示本文件开始，地址为 0000；

S1~S3 表示该行记录为数据，地址长度为 2~4 个字节；

S5 表示本文件中 S1~S3 记录的个数；

S7~S9 表示本文件结束，程序从指定地址执行，地址长度为 4~2 个字节。

2. 长度：

地址字节数＋数据字节数＋校验字节数。

3. 地址：

2~4 个字节，表示操作的起始地址。

4. 校验：

1 个字节，CRC = 0xFF - (记录长度 + 存储地址 + 数据)。

例如：

第一行 S0 03 0000 FC　　起始标志。

第二行 S3 19 0000 0E14 104C0093000570C0E0051CC60000009603630374 66 写入数据。

最末行 S7 05 0000 0E00 EC　　结束标志，从 0E00 开始执行。

uFlasher 的刷写流程

通常的刷写操作流程是：

CAN 总线建立连接→ 确认密码→ 发送 BootLoader 的二进制信息→

擦除 flash→ 查空→ 发送程序的二进制信息。下面逐项进行说明。

1. 发送 BootLoader（CONNECT）：

①一次性发送 FE ED FA CE CA FE BE EF。

此为刷写密码。

CAN 总线的发送 ID=0x11，接收 ID=0x01。

②一次性发送 40 00 01 00 80 01 07 D0。

其中 40 00 01 00 为刷写起始地址，8 为 VLE BIT，0 01 07 D0 为刷写数据大小。

CAN 总线的发送 ID=0x12，接收 ID=0x02。

③一次性发送 70 00 00 00 7C 01 03 A6。

对应 HEX 文件内容为:02 000004 4000 BA + :20 0100 00 7000 0000 7C01。

CAN 总线的发送 ID=0x13，接收 ID=0x03。

④一次性发送 00 00 00 00 00 00 00 00。

对应 HEX 文件内容为:02 000004 4000 BA + :20 FFE0 00 0000 0000 0000。

CAN 总线的发送 ID=0x13，接收 ID=0x03。

… …

⑤一次性发送 40 00 01 A0 04 00 01 A0。

对应 HEX 文件内容为:02 000004 4001 B9 + :10 08C0 00 4000 01A0 4000。

CAN 总线的发送 ID=0x13，接收 ID=0x03。

⑥对 HEX 文件末尾的“:04 000005 4000 0E00 A9”和“:00 000001 FF”不作任何处理。

2．擦除（ERASE）：

逐个字节发送 07 51 12 00 00 13 FF FF 74。

其中 07 51 为固定内容，12 00 00 为起始地址，13 FF FF 为结束地址，74 为校验位；

目标板返回 51。

CAN 总线的发送 ID=0x01，接收 ID=0x34。

3．查空（BLANK CHECK，非必需）：

逐个字节发送 07 76 12 00 00 13 00 00 9B。

其中 07 76 为固定内容，12 00 00 为起始地址，13 00 00 为结束地址，9B 为校验位；

目标板返回 00 56。

CAN 总线的发送 ID=0x00，接收 ID=0x34。

3．发送程序（PROGRAM）：

①逐个字节发送 07 34 00 00 00 17 FF FF 49。

其中 07 34 为固定内容，00 00 00 为起始地址，17 FF FF 为结束地址，49 为校验位；

目标板返回 52。（应该是从擦除开始累加的序号，如果加入查空，该序号可能需要加 1）

CAN 总线的发送 ID=0x00，接收 ID=0x34。

②逐个字节发送 41 FA 36 01 (DATA) (CHECK) 。

其中 4 为固定内容，含义不明；

1FA 为后续发送数据总长度（包括序号和校验位），如果数据总长度小于 0x0100，前面的

字节 4X 应省略；

36 为固定内容，含义不明；

01 为数据帧的序号，该序号从 01 开始累加到 FF 后重置为 00；

(DATA)为数据正文，每次 8 个字节，共发送 63 次，504 个字节；

(CHECK)为 1 个字节的校验位，使用校验和，从 36 开始累加，直到数据最后一个字节；

目标板返回 5X，X 等于发送序号＋1 的后面半个字节。

CAN 总线的发送 ID=0x00，接收 ID=0x34。

③重复上述动作②直到所有数据发送完成。

④逐个字节发送 01 37 37。

内容固定，但含义不明；

目标板返回 53，刷写到此结束。

CAN 总线的发送 ID=0x00，接收 ID=0x34。

关于刷写过程的可靠性

1．在发送 BootLoader 过程中，每发送一帧数据，SPC563 都会把收到的内容回传一遍，所以可以通过确认回传内容来验证该刷写过程。该机制是由 SPC563 的 BAM 模块实现。

2．在通过 BootLoader 刷写程序的过程中，BootLoader 并不回传接收到的信息，但是 BootLoader 要求发送的每帧数据末尾要有一个校验位，估计 BootLoader 会根据这个校验位对接收到的内容进行验证，如果验证出错，应该会要求重发。目前还没有进行验证。

SD 卡刷写的配置文件

使用 SD 卡脱机刷写时需要编写一个描述刷写过程的配置文件，程序中并没有固定任何刷写的细节。

配置文件会被逐行处理，基本语法为：

1．行首为“/”表示注释，本行后面全部内容将被忽略。

2．行首为“@”表示设置。

目前只有两种情况，后面是数字表示延时，否则就会查找“bs1=”、“bs2=”和“scale=”，设置 CAN 总线通信速率。

通信速率计算方法为 baudrate = 30M / ((1 + bs1 + bs2) * scale) ,虽然各种 bs1 和 bs2 的组合都可以得到同样的通信速率，但是其中“1+bs1”所占的比例一般认为 500kbps 时应该在 75%，低于 300kbps 时应该在 90%。

3．行首为“<<”表示发送内容。

格式为“<<CAN 总线 ID：16 进制数据；”，如“<<32:12AB;”，ID 和数据以冒号分隔，数据不能超过 8 个字节即 16 个字符。多项发送的数据可以用“；”分隔，写在同一行，如“<<32:12;AB;”。

数据部分中，数据之外有一些符号表示要发送计算后的数值。

包括：“0&”，表示开始计算校验和，而不是发送数据。直到“&&”时才把算好的校验和

发送出去；“LL”表示后面要发送的数据总长度，而“4L”表示 4 和发送数据总长度的组合；“NN”表示发送帧序号。

发送二进制文件时，使用“<<ID=文件名”的形式，文件名不区分大小写。文件名之后使用“%8*63”这样的形式表示每次发送 8 个字节，每帧发送 63 次，后面使用“:”与发送内容相分隔。发送的内容中，“XX”表示发送的数据，长度为 1~8 个字节。到行尾之前的内容会一直循环发送直到发送完整个二进制文件。

4. 行首为“>>”表示应该接收到的内容，以此与实际接收到的内容相比较。
“~”表示收到的内容应该与前一次发送的内容相同。

5. 整个配置文件的最后一行将被忽略，无论是什么内容。

使用 300kbps 速率，刷写 SPC563 时的配置文件内容如下：

```
        //先设置通信速率。
@ bs1=14, bs2=5, scale=5;
        //发送刷写密码。
<<11:FEEDFACECAFEBEEF;>>01:~;
        //设置刷写范围为从0x40000100开始的0x107D0字节。
<<12:40000100800107D0;>>02:~;
        //使用SPC563的BAM模块，下载由SPC563M64_300_CAN.hex文件转成的BIN文件。
<<13=LOADER300.BIN%8*1:XX;>>03:~;
        //刷写LOADER300文件之后，在刷写程序之前先等待一秒钟。
@ 1000
        //擦除Flash，地址范围为0x000000到0x7FFFFF。
<<01:07;0&;51;00;00;00;00;3F;FF;&&;>>34:51;
<<01:07;0&;51;00;40;00;00;7F;FF;&&;>>34:51;
<<01:07;0&;51;00;80;00;00;FF;FF;&&;>>34:51;
<<01:07;0&;51;01;00;00;01;7F;FF;&&;>>34:51;
<<01:07;0&;51;01;80;00;01;BF;FF;&&;>>34:51;
<<01:07;0&;51;01;C0;00;01;FF;FF;&&;>>34:51;
<<01:07;0&;51;02;00;00;02;FF;FF;&&;>>34:51;
<<01:07;0&;51;03;00;00;03;FF;FF;&&;>>34:51;
<<01:07;0&;51;04;00;00;05;FF;FF;&&;>>34:51;
<<01:07;0&;51;06;00;00;07;FF;FF;&&;>>34:51;
<<01:07;0&;51;08;00;00;09;FF;FF;&&;>>34:51;
<<01:07;0&;51;0A;00;00;0B;FF;FF;&&;>>34:51;
<<01:07;0&;51;0C;00;00;0D;FF;FF;&&;>>34:51;
<<01:07;0&;51;0E;00;00;0F;FF;FF;&&;>>34:51;
<<01:07;0&;51;10;00;00;11;FF;FF;&&;>>34:51;
```

<<01:07;0&;51;12;00;00;13;FF;FF;&&;>>34:51;
<<01:07;0&;51;14;00;00;15;FF;FF;&&;>>34:51;
<<01:07;0&;51;16;00;00;17;FF;FF;&&;>>34:51;
//设置刷写范围为0x000000到0x17FFFF。
<<00:07;0&;34;00;00;00;17;FF;FF;&&;>>34:52;
//将程序文件转成的FLASH1212.BIN文件下载到SPC563。
<<00=FLASH1212.BIN%8*63:4L;0&;36;NN;XX;&&;>>34:5N;
//刷写完成。
<<00:01;37;37;>>34:53;

刷写前的准备

SPC563有时必须复位才能开始刷写，所以建议每次刷写前都给SPC563重新上电。目前为止并未遇到过CAN总线通信线路方面的问题，所以只要给SPC563重新上电就够了。