

嵌入式系统在线编程软件K-Flash

—— 快速、稳定地编程任意类型Flash

TKScope是广州致远电子有限公司2008年隆重推出上市的一款高性能通用型综合仿真开发平台，支持仿真全系列的8051、ARM、DSP、AVR、C166、C251、MX等内核；与当前全部主流IDE环境无缝嫁接，如Keil、ADS、IAR、CCS、RealView、AVRStudio、TKStudio等，保证您的开发平台始终如一，并具备其高级调试功能。同时，TKScope内嵌64路专业的逻辑分析仪，zlgLogic高级软件全面支持。

TKScope内部已经集成常用的各种内部/外部Flash算法，可以在线烧写各种ARM芯片内部的Flash，ARM芯片外挂的各种常用和非常用Flash，例如NOR / NAND / SPI DATA等。

对于特殊的Flash芯片，由于对用户公开的编程算法非常简单，因此用户可以自己完成对特殊Flash的编程，例如SD卡。

TKScope集成的Flash算法高度模块化，支持在仿真开发阶段的动态Flash烧写，同时也适合在产品量产时的快速Flash编程。

1.支持任意Flash器件的在线编程

TKScope仿真ARM内核时，支持片内/片外Flash的在线编程/调试；TKScope支持多种接口形式的片外Flash器件，如NOR、NAND、SPI等。

TKScope通过驱动设置中的【程序烧写】选项，配置需要编程的Flash信息，如图1所示。图1示例，NXP公司LPC2214芯片，外扩2片NOR Flash器件SST39VF160。根据LPC2214器件特性，外扩NOR Flash器件起始地址为0x80000000、0x81000000。

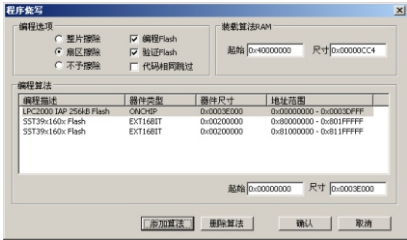


图1 程序烧写界面

图1所示【程序烧写】界面中，点击【添加算法】来添加用户需要烧写的Flash器件编程算法。TKScope提供绝大多数常用的Flash器件编程算法，存放于驱动安装路径... \TKScope\configuration\ExtFlash文件夹内。

由于市面上Flash种类比较多，而且编程算法是不通用的，TKScope不可能包含所有的Flash器件编程算法。在具体应用中，如果用户在TKScope驱动中找不到相应的Flash编程算法，可以自己添加。TKScope完全公开

Flash编程算法接口，支持用户自行添加，让用户应用更加方便自如。

2.Flash编程算法接口文件

TKScope定义了Flash编程算法的接口，包括1个描述Flash芯片的结构体和7个对Flash芯片操作的函数定义。如程序清单2.1所示。

```
// Flash 设备结构体定义
struct FlashDevice {
    unsigned short Vers; // 体系结构和版本号
    char DevName[128]; // 设备的名称及描述
    unsigned short DevType; // 设备的类型，例如：ONCHIP, EXT8BIT, EXT16BIT 等等
    unsigned long DevAddr; // 默认设备的起始地址
    unsigned long soDev; // 设备的总容量
    unsigned long soPage; // 编程的页面大小
    unsigned long Res; // 保留，以便将来扩展之用
    unsigned char valEmpty; // 设备擦除后存储单元的值
    unsigned long toProg; // 页编程函数超时的时间
    unsigned long toErase; // 扇区擦除函数超时的时间
    struct FlashSectors sectors[SECTOR_NUM]; // 扇区的起始地址及容量设置数组
};

/**
 * 函数名称: Init
 * 函数功能: 算法初始化
 * 输入参数: adr: 设备基址
 *           clk: 时钟频率
 *           fnc: 功能代码 (1 - 擦除, 2 - 编程, 3 - 校验)
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern int Init(unsigned long adr, unsigned long clk, unsigned long fnc);

/**
 * 函数名称: UnInit
 * 函数功能: 算法卸载
 * 输入参数: fnc: 功能代码 (1 - 擦除, 2 - 编程, 3 - 校验)
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern int UnInit(unsigned long fnc);

/**
 * 函数名称: BlankCheck
 * 函数功能: 查空
 * 输入参数: adr: 起始地址
 *           sz: 字节数
 *           pat: 匹配值
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern int BlankCheck(unsigned long adr, unsigned long sz, unsigned char pat);

/**
 * 函数名称: EraseChip
 * 函数功能: 整片擦除
 * 输入参数: 无
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern int EraseChip(void);

/**
 * 函数名称: EraseSector
 * 函数功能: 扇区擦除
 * 输入参数: adr: 扇区起始地址
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern int EraseSector(unsigned long adr);

/**
 * 函数名称: ProgramPage
 * 函数功能: 页编程
 * 输入参数: adr: 起始地址
 *           sz: 字节数
 *           buf: 输入缓冲区
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern int ProgramPage(unsigned long adr, unsigned long sz, unsigned char *buf);

/**
 * 函数名称: Verify
 * 函数功能: 校验
 * 输入参数: adr: 起始地址
 *           sz: 字节数
 *           buf: 输入缓冲区
 * 返回参数: 0 - 成功, 1 - 失败
 */
extern unsigned long Verify(unsigned long adr, unsigned long sz, unsigned char *buf);
```

图2 程序清单 2.1

TKScope添加Flash编程算法的实质就是填充这个结构体以及实现这7个函数。至于几个函数是如何被TKScope调用的，用户不必关心，这些是由TKScope自动管理的。只要正确的实现了这些内容，用户就可以将编译链接后的程序下载到自己的Flash芯片中去。

3.量产型在线编程软件K-Flash

K-Flash是一款用于产品量产的Flash在线编程软件，支持TKScope/ AK100系列仿真器，具有文件烧写、数据擦除、数据读取等功能。K-Flash具有如下特点：

- 工程化配置，操作简单、方便、快捷；
- 支持TKScope/AK100系列仿真器；
- 支持bin文件、hex文件、elf文件等多种类型的文件；
- 支持片内Flash和片外Flash烧写；
- 多种Flash烧写操作，可以进行擦除和读取等操作；
- 具有独立的擦除、读取功能，可指定地址范围进行操作；

- 保存配置到工程文件中，免除繁琐重复的配置操作；
- 支持各大半导体公司的芯片烧写，内置多种芯片的默认配置。

K-Flash软件界面如图3所示，界面大方美观，操作选项简单，一目了然。适合反复在线烧写Flash的场合以及工厂最终生产烧写Flash器件。



图3 K-Flash界面

K-Flash可以进行文件烧写、数据擦除、数据读取等操作，具体功能描述如下：

- 文件烧写：可以烧写bin文件、hex文件和elf文件等多种类型的文件。
- 数据擦除：擦除指定地址段内的数据。
- 数据读取：读取指定的起始地址和大小数据并保存到用户指定的路径。
- K-Flash支持工程文件的形式，形成工程文件后，相关的设备配置等内容会自动保存。这样用户再次对文件进行其它操作时，无需繁琐重复的参数设置，直接打开工程文件就可以继续进行操作。

◎ 开发工具系列产品



- 新建：新建一个空白工程文件；
- 打开文件：打开一个指定的工程文件；
- 保存文件：保存一个工程文件到指定的路径；
- 另存文件：把当前配置另存到指定的位置；
- 设备配置：配置仿真器类型、芯片参数、Flash编程算法等；
- 关于：K-Flash的相关信息。

4.在线编程Flash方法

使用K-Flash软件，TKScope可以轻松实现在线编程Flash。下面详细阐述TKScope在线编程Flash的步骤及方法：

- 用户把需要烧写到Flash中的代码保存为bin、hex或elf格式的文件。
- 查找TKScope驱动中是否提供相应Flash的编程算法。如果没有提供，用户需要自行添加Flash编程算法。
- 正确连接TKScope仿真器与用户目标板，确保硬件工作正常。
- 打开K-Flash软件，点击【设备配置】，正确配置当前使用的仿真器型号，添加需要烧写Flash器件的编程算法文件。
- 此时，调入需要烧写的代码文件就可以轻松写入Flash器件中。当然，也可以进行擦除、读取等操作。
- 用户可以把当前操作的配置信息、文件信息等保存为工程文件，下次使用直接打开工程文件即可，省去了重复配置参数的繁琐操作。



图4 K-Flash烧写Flash界面

广州致远电子有限公司

地址：广州市天河区车陂路黄洲工业区3栋2楼 电话：(020)22644372 28872453
技术支持：(020) 020-22644360 邮箱：TKS@zlgmcu.com