

USMART

1, USMART简介 (了解)

USMART是一个串口调试组件，可以大大提高代码调试效率

可以待用绝大部分用户直接编写的函数

占用资源少 最小：4KB FLASH, 72B SRAM

USMART特点

支持参数类型多 整数 (10/16)、字符串、函数指针

支持函数返回值显示且对格式进行设置

支持函数执行时间计算

2, USMART原理 (了解)

通过对比用户输入字符串和本地函数名，用函数指针实现调用不同的函数

用户输入字符串 解析并对比 执行函数 返回执行结果

丢弃，报错

USMART组成

usmart.c/h 核心文件，用于处理命令以及对外部交互

usmart_config.c 函数管理文件，用于添加用户需要USMART管理的函数

usmart_port.c/h 移植文件，用于USMART移植

usmart_str.c/h 字符串处理文件，用于字符串转换、参数获取等

USMART扫描函数

void usmart_scan(void)

扫描函数，解析用户输入字符串数据，进而实现函数调用和组件各个控制功能

3, USMART移植 (掌握)

- 1, 获取USMART组件 找到USMART的全部组件，从正点原子的USMART例程获得
- 2, 添加到工程 添加全部组件到工程，4个文件，设置好路径关联
- 3, 适配硬件 修改调试串口和定时器，以适配自己的硬件
- 4, 添加执行函数 添加自己需要的执行函数，在usmart_config.c中添加修改
- 5, 通过串口交互 烧录移植好的USMART组件，可以通过串口反复测试目标函数

4, USMART使用 (掌握)

USMART有7个系统命令

? : 获取帮助信息

help: 获取帮助信息

list: 可用的函数列表

id: 可用函数的ID列表

hex: 参数16进制显示，后跟空格+数字即执行进制转换

dec: 参数10进制显示，后跟空格+数字即执行进制转换

runtime 1 开启函数运行计时

runtime 0 关闭函数运行计时

注意：请按照程序编写格式输入函数名及参数并以回车键结束