**说明**

总共有五个文件:

tm4c\_build.sh : 生成Makefile & 编译 & 烧写

tm4c\_debug.sh : 生成Makefile & 编译 & 后台启动OpenOCD & 启动GDB & 完成gdb初始化操作

tm4c\_genMakefile.sh : 生成Makefile

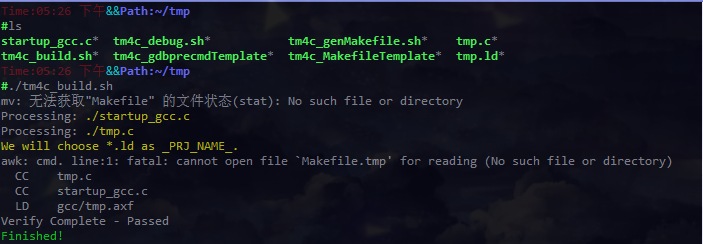
tm4c\_MakefileTemplate : Makefile的模板

tm4c\_gdbprecmdTemplate : gdb初始化操作命令的模板

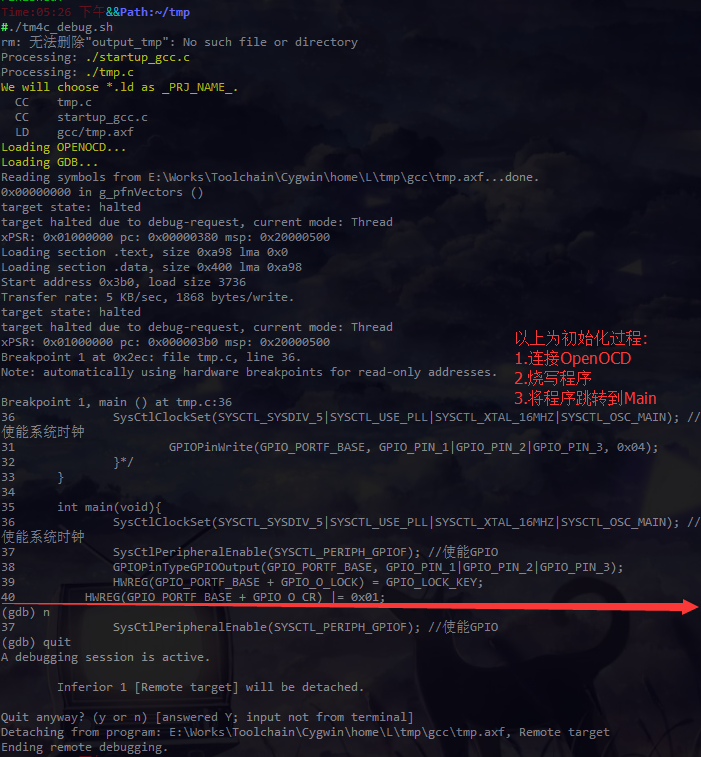
使用方法:

将脚本复制到待编译工程的根目录

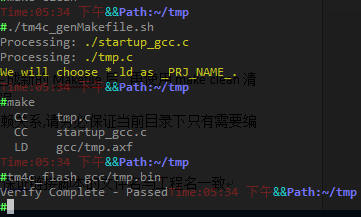
执行./tm4c\_build.sh将完成编译与烧写过程



执行./ tm4c\_debug.sh将完成编译与调试环境初始化过程



执行./tm4c\_genMakefile.sh将仅生成Makefile



注意事项:

1. 如果移动了程序目录的位置, 那么请先./tm4c\_genMakefile.sh生成新的Makefile后, 再使用make clean清空编译结果, 再执行./tm4c\_build.sh 或者 make, 否则会出现错误.
2. 工具将会检索目录下所有的c文件(包括子目录)生成对应的依赖关系,请务必保证当前目录下只有需要编译的c文件.
3. 请保证该目录下至少有主程序,链接脚本以及启动文件
4. 默认以含有main(void)函数的c文件的文件名作为工程名, 如果c文件中含有多个main(void)字符串, 将以ld文件的文件名作为工程名
5. 请保证链接脚本的文件名与工程名一致
6. 目前脚本仅处理driverlib/\*.h以及utils/\*.h的依赖
7. OpenOCD的输出被重定向至文件openocd\_output
8. 如果存在libdriver.a找不到的情况,请尝试执行cd ~/ToolChain/TivaWare && make clean && make以重新编译TivaWare.

**请切勿依赖此工具, 有时间的话学习一下Makefile的规则!**

如果使用中遇到BUG请及时反馈~

**Changelog:**

[2015-01-09 V0.22]

1. 改进识别main函数的方式,将忽略掉任何注释掉的代码

[2015-01-08 V0.21]

1. 修正grep误识别utf-8文件为二进制文件的问题

[2015-01-07 V0.2]

1. 改进对于工程名的识别方式,当检测到多个main函数时,以ld文件名作为工程名
2. 增加对于是否需要make clean的判断,减少编译出错的可能性
3. 增加tm4c\_debug的支持,轻松进入在线调试环境
4. 默认向目标文件中添加调试相关信息

**RoadMap:**

1. 采用内部变量的形式储存中间结果,优化处理流程
2. 将命令做成全局命令,处理好路径依赖

2015-01-09