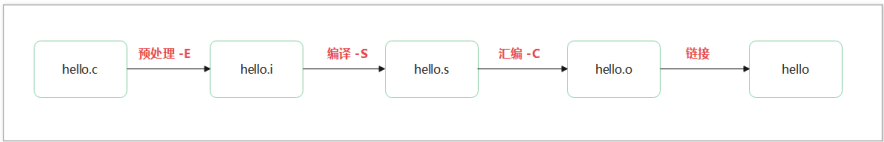


4.1 静态库原理与制作_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 4.1 静态库原理与制作涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

1. 静态库原理与制作

- 程序在编译时分为多个阶段:
 - 预处理
 - 处理所有预处理命名，包括宏定义、条件编译指令、文件包含指令
 - 编译
 - 进行词法分析、语法分析、语义分析后，将代码翻译成汇编指令
 - 汇编
 - 将汇编指令翻译成机器指令，也就是二进制，形成目标文件
 - 链接
 - 将多个目标文件进行链接，得到一个程序最后的执行文件



- 编译过程演示示例:
 - Step 1 : 预处理, 假定源文件名为 hello.c

```
gcc -E hello.c -o hello.i
```

```
gcc -S hello.i -o hello.S
```

```
gcc -c hello.S -o hello.o
```

- 函数库 是实现了某一类功能的若干个函数的集合
- 函数库可以编译独立的二进制文件，在进一步制作成静态库与动态库进行链接使用
- 静态库是函数库二进制文件的一种形式, 在 windows 与 Linux 下对应的文件类型
 - Windows :name.lib
 - Linux: libname.a
- 静态库的特点:
 - 当编译器链接静态库的时候，如果在可执行文件中有调用静态库的函数接口，则会将静态拷贝到可执行文件中
 - 由于可执行文件中有静态库中函数接口的实现代码，运行的时候不需要静态库
 - 由于链接的时候，需要将库函数接口实现代码拷贝到可执行文件中，所以生成
- 静态库的制作:
 - 将 file.c 编译成 file.o

```
gcc -c file.c -o file.o
```

```
ar -rs file.a file.o
```

- ar 命令用于制作静态库的命令, 可以使用一些常用的选项
 - -s: 将目标文件的索引符号添加到库中
 - -r: 在库中更新文件或者添加新的文件
- 静态库在链接使用时需要指定头文件的位置与静态库的位置
 - -I: 指定头文件
 - -L: 指定库的位置
 - -l: 指定链接的库的名字

```
gcc -I <头文件路径> -L <库的路径> -l <静态库的名字> -o <可执行文件名>
```

- gcc 编译器默认搜索头文件与库文件的路径
 - /usr/include 为头文件默认路径
 - /usr/lib 与 /lib 为库的默认路径
- 优点
 - 可执行程序在运行的时候, 不需要加载动态库, 可以直接运行
- 缺点
 - 多个程序链接静态库的时候, 需要拷贝多份静态库的代码, 占用的内存较多
- 练习: 设计一个用于进行算术运算的库 add.h 与 add.c, 然后制作成静态库进行链接

全文完

本文由 简悦 SimpRead 优化, 用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta, 点击查看详细说明

