Linux内核Makefile文件组成

|  |  |
| --- | --- |
| 名称 | 描述 |
| 顶层 Makefile(Kernel Makefile) | 它是所有Makefile文件的核心，从总体上控制着内核的编译、连接 |
| arch/$(ARCH)/Makefile | 对应体系结构的Makefile，它用来决定哪些体系结构相关的文件参与内核的生成，并提供一些规则来生成特定格式的内核映像 |
| scripts/Makefile.\* | Makefile公用的通用规则、脚本等 |
| 子目录Kbuild Makefiles | 各级子目录的Makefile相对简单，被上一层Makefile.build调用来编译当前目录的文件。 |
| 顶层.config | 配置文件，配置内核时生成。所有的Makefile文件（包括顶层目录和各级子目录）都是根据.config来决定使用哪些文件的 |

 scripts/Makefile.\*中有公用的通用规则，展示如下：

scripts/Makefile.\*文件组成

|  |  |
| --- | --- |
| Makefile.build | 被顶层Makefile所调用，与各级子目录的Makefile合起来构成一个完整的Makefile文件，定义.lib、built-in.o以及目标文件.o的生成规则。这个Makefile文件生成了子目录的.lib、built-in.o以及目标文件.o |
| Makefile.clean | 被顶层Makefile所调用，用来删除目标文件等 |
| Makefile.lib | 被Makefile.build所调用，主要是对一些变量的处理，比如说在obj-y前边加上obj目录 |
| Kbuild.include | 被Makefile.build所调用，定义了一些函数，如if\_changed、if\_changed\_rule、echo-cmd |

总结：

* Linux内核Makefile体系核心的Makefile文件就两个：顶层

Makefile、scripts/Makefile.build

* 子目录中的Makefile、kbuild不是Makefile文件（完整的Makefile文件），只能算作是

Makefile的包含文件。

* 顶层Makefile文件负责将各个目录生成的\*.built-in.o、lib.a等文件连接到一起。而
* scripts/Makefile.build 包含子目录中的Makefile文件来生成这些\*.built-in.o、lib.a、\*.o等

文件。

* 执行make后顶层Makefile会根据.config里配置的宏定义选择执行编译某些模块，再调用scripts/Makefile.build，然后会调用子目录的Kbuild或者Makefile（优先调用Kbuild）

