# package/Makefile

$(call subdir,$(curdir))



make prepare -j1 V=s > log.txt 2>&1

$(call stampfile,$(curdir),package,prereq,.config)

1. package/stamp-prereq:=/home/share/openwrt\_cc\_v3.0/staging\_dir/target-arm-27912x-linux-uclibc/stamp/.package\_prereq
2. $(package/stamp-prereq): /home/share/openwrt\_cc\_v3.0/tmp/.build .config
3. @+/home/share/openwrt\_cc\_v3.0/scripts/timestamp.pl -n $(package/stamp-prereq) package .config || make --no-print-directory $(package/flags-prereq) package/prereq
4. @mkdir -p $$(dirname $(package/stamp-prereq))
5. @touch $(package/stamp-prereq)
7. $(**if** ,,.SILENT: $(package/stamp-prereq))
9. .PRECIOUS: $(package/stamp-prereq) # work around a make bug
11. package//clean:=package/stamp-prereq/clean
12. package/stamp-prereq/clean: FORCE
13. @rm -f $(package/stamp-prereq)

$(call stampfile,$(curdir),package,cleanup,$(TMP\_DIR)/.build)

1. package/stamp-cleanup:=/home/share/openwrt\_cc\_v3.0/staging\_dir/target-arm-27912x-linux-uclibc/stamp/.package\_cleanup
2. $(package/stamp-cleanup): /home/share/openwrt\_cc\_v3.0/tmp/.build /home/share/openwrt\_cc\_v3.0/tmp/.build
3. @+/home/share/openwrt\_cc\_v3.0/scripts/timestamp.pl -n $(package/stamp-cleanup) package /home/share/openwrt\_cc\_v3.0/tmp/.build || make --no-print-directory $(package/flags-cleanup) package/cleanup
4. @mkdir -p $$(dirname $(package/stamp-cleanup))
5. @touch $(package/stamp-cleanup)
7. $(**if** ,,.SILENT: $(package/stamp-cleanup))
9. .PRECIOUS: $(package/stamp-cleanup) # work around a make bug
11. package//clean:=package/stamp-cleanup/clean
12. package/stamp-cleanup/clean: FORCE
13. @rm -f $(package/stamp-cleanup)

$(call stampfile,$(curdir),package,compile,$(TMP\_DIR)/.build)

1. package/stamp-compile:=/home/share/openwrt\_cc\_v3.0/staging\_dir/target-arm-27912x-linux-uclibc/stamp/.package\_compile
2. $(package/stamp-compile): /home/share/openwrt\_cc\_v3.0/tmp/.build /home/share/openwrt\_cc\_v3.0/tmp/.build
3. @+/home/share/openwrt\_cc\_v3.0/scripts/timestamp.pl -n $(package/stamp-compile) package /home/share/openwrt\_cc\_v3.0/tmp/.build || make --no-print-directory $(package/flags-compile) package/compile
4. @mkdir -p $$(dirname $(package/stamp-compile))
5. @touch $(package/stamp-compile)
7. $(**if** ,,.SILENT: $(package/stamp-compile))
9. .PRECIOUS: $(package/stamp-compile) # work around a make bug
11. package//clean:=package/stamp-compile/clean
12. package/stamp-compile/clean: FORCE
13. @rm -f $(package/stamp-compile)

$(call stampfile,$(curdir),package,install,$(TMP\_DIR)/.build)

1. package/stamp-install:=/home/share/openwrt\_cc\_v3.0/staging\_dir/target-arm-27912x-linux-uclibc/stamp/.package\_install
2. $(package/stamp-install): /home/share/openwrt\_cc\_v3.0/tmp/.build /home/share/openwrt\_cc\_v3.0/tmp/.build
3. @+/home/share/openwrt\_cc\_v3.0/scripts/timestamp.pl -n $(package/stamp-install) package /home/share/openwrt\_cc\_v3.0/tmp/.build || make --no-print-directory $(package/flags-install) package/install
4. @mkdir -p $$(dirname $(package/stamp-install))
5. @touch $(package/stamp-install)
7. $(**if** ,,.SILENT: $(package/stamp-install))
9. .PRECIOUS: $(package/stamp-install) # work around a make bug
11. package//clean:=package/stamp-install/clean
12. package/stamp-install/clean: FORCE
13. @rm -f $(package/stamp-install)

## package/upointech/testapp/Makefile

PKG\_NAME=testappv3

$(call BuildPackage,$(PKG\_NAME))



# tools/Makefile

tools\_enabled = $(foreach tool,$(sort $(tools-y) $(tools-)),$(if $(filter $(tool),$(tools-y)),y,n))

1. tools\_enabled=y y n y y n y n y n y y y y y n n y y n y y y y y y n n y n y y y y y n n y y y n n y y n n n y y

$(call stampfile,$(curdir),tools,install,,\_$(subst $(space),,$(tools\_enabled)))

1. tools/stamp-install:=/home/share/openwrt\_cc\_v3.0/staging\_dir/target-arm-27912x-linux-uclibc/stamp/.tools\_install\_yynyynynynyyyyynnyynyyyyyynnynyyyyynnyyynnyynnnyy
2. $(tools/stamp-install): /home/share/openwrt\_cc\_v3.0/tmp/.build
3. @+/home/share/openwrt\_cc\_v3.0/scripts/timestamp.pl -n $(tools/stamp-install) tools  || make --no-print-directory $(tools/flags-install) tools/install
4. @mkdir -p $$(dirname $(tools/stamp-install))
5. @touch $(tools/stamp-install)
7. $(**if** ,,.SILENT: $(tools/stamp-install))
9. .PRECIOUS: $(tools/stamp-install) # work around a make bug
11. tools//clean:=tools/stamp-install/clean
12. tools/stamp-install/clean: FORCE
13. @rm -f $(tools/stamp-install)

$(call subdir,$(curdir))



# toolchain/Makefile

$(call stampfile,$(curdir),toolchain,

install,$(TOOLCHAIN\_DIR)/stamp/.gcc-initial\_installed,,$(TOOLCHAIN\_DIR))

1. toolchain/stamp-install:=/home/share/openwrt\_cc\_v3.0/staging\_dir/toolchain-arm-27912x-linux-uclibc/stamp/.toolchain\_install
2. $(toolchain/stamp-install): /home/share/openwrt\_cc\_v3.0/tmp/.build /home/share/openwrt\_cc\_v3.0/staging\_dir/toolchain-arm-27912x-linux-uclibc/stamp/.gcc-initial\_installed
3. @+/home/share/openwrt\_cc\_v3.0/scripts/timestamp.pl -n $(toolchain/stamp-install) toolchain /home/share/openwrt\_cc\_v3.0/staging\_dir/toolchain-arm-27912x-linux-uclibc/stamp/.gcc-initial\_installed || make --no-print-directory $(toolchain/flags-install) toolchain/install
4. @mkdir -p $$(dirname $(toolchain/stamp-install))
5. @touch $(toolchain/stamp-install)
7. $(**if** ,,.SILENT: $(toolchain/stamp-install))
9. .PRECIOUS: $(toolchain/stamp-install) # work around a make bug
11. toolchain//clean:=toolchain/stamp-install/clean
12. toolchain/stamp-install/clean: FORCE
13. @rm -f $(toolchain/stamp-install)

$(call subdir,$(curdir))



 

package/libs/libubox/download:

@+ $(SUBMAKE) -r -C package/libs/libubox download BUILD\_VARIANT=""

# aliases

package/libubox/download: package/libs/libubox/download

include/download.mk

DownloadMethod/git

make package/libs/libubox/download -j1 V=s --just-print

# Makefile执行过程

4.1 include/host.mk

1. .PRECIOUS: $(TMP\_DIR)/.host.mk
2. $(TMP\_DIR)/.host.mk: $(TOPDIR)/include/host.mk
3. @mkdir -p $(TMP\_DIR)
4. @( \
5. HOST\_OS=`uname`; \
6. **case** "$$HOST\_OS" in \
7. Linux) HOST\_ARCH=`uname -m`;; \
8. Darwin) HOST\_ARCH=`uname -m`;; \
9. \*) HOST\_ARCH=`uname -p`;; \
10. esac; \
11. GNU\_HOST\_NAME=`gcc -dumpmachine`; \
12. [ -z "$$GNU\_HOST\_NAME" -o "$$HOST\_OS" = "Darwin" ] && \
13. GNU\_HOST\_NAME=`$(TOPDIR)/scripts/config.guess`; \
14. echo "HOST\_OS:=$$HOST\_OS" > $@; \
15. echo "HOST\_ARCH:=$$HOST\_ARCH" >> $@; \
16. echo "GNU\_HOST\_NAME:=$$GNU\_HOST\_NAME" >> $@; \
17. **if** gfind -L /dev/null || find -L /dev/null; then \
18. echo "FIND\_L=find -L \$$(1)" >> $@; \
19. **else** \
20. echo "FIND\_L=find \$$(1) -follow" >> $@; \
21. fi \
22. ) >/dev/null 2>/dev/null
24. endif

4.2 include/toplevel.mk:182

<http://blog.chinaunix.net/uid-20564848-id-225173.html>

1. %::
2. @+$(PREP\_MK) $(NO\_TRACE\_MAKE) -r -s prereq  /\*OPENWRT\_BUILD= QUIET=0 remake V=s -r -s prereq\*/
3. @( \
4. cp .config tmp/.config; \
5. ./scripts/config/conf --defconfig=tmp/.config -w tmp/.config Config.in > /dev/null 2>&1; \
6. **if** ./scripts/kconfig.pl '>' .config tmp/.config | grep -q CONFIG; then \
7. printf "$(\_R)WARNING: your configuration is out of sync. Please run make menuconfig, oldconfig or defconfig!$(\_N)\n" >&2; \
8. fi \
9. )
10. @+$(ULIMIT\_FIX) $(SUBMAKE) -r $@ $(**if** $(WARN\_PARALLEL\_ERROR), || { \
11. printf "$(\_R)Build failed - please re-run with -j1 to see the real error message$(\_N)\n" >&2; \
12. **false**; \
13. } )

4.3 include/toplevel.mk:167

1. prereq:: prepare-tmpinfo .config
2. @+$(NO\_TRACE\_MAKE) -r -s $@

4.3 include/toplevel.mk:68

1. prepare-tmpinfo: FORCE
2. @+$(MAKE) -r -s staging\_dir/host/.prereq-build $(PREP\_MK)  /\*remake -r -s staging\_dir/host/.prereq-build OPENWRT\_BUILD= QUIET=0\*/
3. mkdir -p tmp/info
4. $(\_SINGLE)$(NO\_TRACE\_MAKE) -j1 -r -s -f include/scan.mk SCAN\_TARGET="packageinfo" SCAN\_DIR="package" SCAN\_NAME="package" SCAN\_DEPS="$(TOPDIR)/include/package\*.mk $(TOPDIR)/overlay/\*/\*.mk" SCAN\_DEPTH=5 SCAN\_EXTRA=""
5. $(\_SINGLE)$(NO\_TRACE\_MAKE) -j1 -r -s -f include/scan.mk SCAN\_TARGET="targetinfo" SCAN\_DIR="target/linux" SCAN\_NAME="target" SCAN\_DEPS="profiles/\*.mk $(TOPDIR)/include/kernel\*.mk $(TOPDIR)/include/target.mk" SCAN\_DEPTH=2 SCAN\_EXTRA="" SCAN\_MAKEOPTS="TARGET\_BUILD=1"
6. **for** type in package target; **do** \
7. f=tmp/.$${type}info; t=tmp/.config-$${type}.in; \
8. [ "$$t" -nt "$$f" ] || ./scripts/metadata.pl $${type}\_config "$$f" > "$$t" || { rm -f "$$t"; echo "Failed to build $$t"; **false**; **break**; }; \
9. done
10. [ tmp/.config-feeds.in -nt tmp/.packagefeeds ] || ./scripts/feeds feed\_config > tmp/.config-feeds.in
11. ./scripts/metadata.pl package\_mk tmp/.packageinfo > tmp/.packagedeps || { rm -f tmp/.packagedeps; **false**; }
12. ./scripts/metadata.pl package\_feeds tmp/.packageinfo > tmp/.packagefeeds || { rm -f tmp/.packagefeeds; **false**; }
13. touch $(TOPDIR)/tmp/.build

4.4 include/toplevel.mk:140 staging\_dir/host/.prereq-build

1. staging\_dir/host/.prereq-build: include/prereq-build.mk
2. mkdir -p tmp
3. rm -f tmp/.host.mk
4. @$(\_SINGLE)$(NO\_TRACE\_MAKE) -j1 -r -s -f $(TOPDIR)/include/prereq-build.mk prereq 2>/dev/null || { \
5. echo "Prerequisite check failed. Use FORCE=1 to override."; \
6. **false**; \
7. }
8. ifneq ($(realpath $(TOPDIR)/include/prepare.mk),)
9. @$(\_SINGLE)$(NO\_TRACE\_MAKE) -j1 -r -s -f $(TOPDIR)/include/prepare.mk prepare 2>/dev/null || { \
10. echo "Preparation failed."; \
11. **false**; \
12. }
13. endif
14. touch $@   /\*touch staging\_dir/host/.prereq-build\*/

# gdb编译

make package/devel/gdb/compile

make package/devel/gdb/install -j1 V=s

staging\_dir/target-arm-27912x-linux-uclibc/root-zx279128/usr/bin