

# Variables

## Auto override

```
# Create an automatic OVERRIDE for everything under src/
AUTO_OVERRIDES = $(notdir $(wildcard $(BR2_EXTERNAL)/src/*))
do_override = $(1)_OVERRIDE_SRCDIR = $(BR2_EXTERNAL)/src/$(2)
$(foreach override,$(AUTO_OVERRIDES),\
    $(eval $(call do_override,$(call
UPPERCASE,$(override)),$(override))))

#(this snippet is licensed GPLv2+).
#by Arnout Vandecappelle (Essensium/Mind)
```

## Variables

\$(@D)	which contains the directory in which the package source code has been uncompressed.
\$(FOO_DIR)	same as \$(@D)
\$(FOO_PKGDIR)	contains the path to the directory containing the foo.mk and Config.in files.
\$(HOST_DIR)	
\$(STAGING_DIR)	
\$(TARGET_DIR)	

## Cross compilation

Variables are defined in *buildroot/package/Makefile.in*

\$(TARGET_CC)	cross compiler gcc binary (i.e.: TARGET_CC=/home/user/project/out/host/usr/bin/ccache /home/user/project/out/host/usr/bin/arm-linux-gcc)
\$(TARGET_AR)    \$(TARGET_AS)    \$(TARGET_CPP)    \$(TARGET_CXX)    _FC_LD_NM   _RANLIB   _READELF   _OBJCOPY    _OBJDUMP	
\$(TARGET_CC_NOCCACHE)	
\$(TARGET_CXX_NOCCACHE)	
\$(TARGET_CFLAGS)	(i.e: TARGET_CFLAGS=-D_LARGEFILE_SOURCE -D_LARGEFILE64_SOURCE - D_FILE_OFFSET_BITS=64 -pipe -Os -g2)
\$(TARGET_CPPFLAGS)	
\$(TARGET_CXXFLAGS)	
\$(TARGET_LDFLAGS)	
\$(TARGET_MAKE_ENV)	

It overrides search path, to get cross compiler first.  
(i.e:PATH="/home/user/project/out/host/bin:/home/user/project/out/host/ ...) |

\$(TARGET_DIR)	device root filesystem (i.e: needed for installing packages into \$(TARGET_DIR)/usr/bin)
\$(STAGING_DIR)	development target root filesystem (contains all needed libraries to build other packages, contains debug info and not stripped binary)

Additional variables (not needed in normal use)

\$(CCACHE)	full path to ccache binary or empty (i.e: CCACHE=/home/user/project/out/host/usr/bin/ccache)
\$(TARGET_CROSS)	prefix with full path to cross compiler (i.e: TARGET_CROSS=/home/user/project/out/host/usr/bin/arm-linux-)
\$(ABI)	gnueabi[hf] eabi[hf]
\$(GNU_TARGET_NAME)	GNU_TARGET_NAME=arm-buildroot-linux-uclibcgnueabi

Example

```
#####
####
#
# genromfs
#
#####
####

GENROMFS_VERSION = 0.5.2
GENROMFS_SITE =
http://downloads.sourceforge.net/project/romfs/genromfs/$(GENROMFS_VERSION)
GENROMFS_LICENSE = GPLv2+
GENROMFS_LICENSE_FILES = COPYING

define GENROMFS_BUILD_CMDS
    $(TARGET_MAKE_ENV) $(MAKE) -C $(@D) \
        CC="$(TARGET_CC)" \
        CFLAGS="$(TARGET_CFLAGS)" \
        LDFLAGS="$(TARGET_LDFLAGS)"
endef

define GENROMFS_INSTALL_TARGET_CMDS
    $(TARGET_MAKE_ENV) $(MAKE) -C $(@D) PREFIX=$(TARGET_DIR) install
endef

define HOST_GENROMFS_BUILD_CMDS
    $(HOST_MAKE_ENV) $(MAKE) -C $(@D)
endef

define HOST_GENROMFS_INSTALL_CMDS
    $(HOST_MAKE_ENV) $(MAKE) -C $(@D) PREFIX=$(HOST_DIR) install
endef

$(eval $(generic-package))
```

```
$(eval $(host-generic-package))
```

## LUA

```
LUAPOSIX_CONF_OPTS = --libdir="/usr/lib/lua/$(LUAINTERPRETER_ABIVER)" --  
datarootdir="/usr/share/lua/$(LUAINTERPRETER_ABIVER)"
```

```
define LUAEXPATUTILS_INSTALL_TARGET_CMDS  
$(INSTALL) -m 0644 -D $(@D)/lua/doc.lua  
>$(TARGET_DIR)/usr/share/lua/$(LUAINTERPRETER_ABIVER)/lxp/doc.lua  
endef
```

```
define LUA_MSGPACK_NATIVE_BUILD_CMDS  
$(TARGET_CC) $(TARGET_CFLAGS) -fPIC -shared -o $(@D)/msgpack.so  
$(@D)/mp.c  
endef
```

```
define LUA_MSGPACK_NATIVE_INSTALL_TARGET_CMDS  
$(INSTALL) -m 755 -D $(@D)/msgpack.so \  
$(TARGET_DIR)/usr/lib/lua/$(LUAINTERPRETER_ABIVER)/msgpack.so  
endef
```

```
$(eval $(generic-package))
```

From:

<https://niziak.spox.org/wiki/> - **niziak.spox.org**

Permanent link:

<https://niziak.spox.org/wiki/buildroot:variables>

Last update: **2024/01/02 16:41**

