Building the LINUX kernel for LEON

LINUX build process overview

Written by Konrad Eisele, Daniel Hellstrom

LinuxBuild
Version 1.0.4
May 2012

Kungsgatan 12
413 11 Gothenburg
Sweden

tel +46 31 7758650
fax +46 31 421407
www.aeroflex.com/gaisler

LINUX build process overview

Konrad Eisele, Daniel Hellstrom

Table of Contents

1. LINUXBUILD

1.1. Introduction

This document describes how the Linuxbuild utility is used to build one of more of the components listed below. The Linuxbuild utility consists of small Makefile-scripts for building and configuring some of the Linux tools that Aeroflex Gaisler provides patches for.

- Linux Kernel
- Buildroot user-space root file system
- Linux Kernel RAM image
- PROM/FLASH image

The Linuxbuild utility is a quick way of getting started with Linux development for the LEON architecture, it ties different components together to build a complete Linux environment. Each component is designed to be used separately from each other, one can see Linuxbuild as an example utility that provides a quick way of getting started with Linux development using the different tools and components. Currently the following components are supported in Linuxbuild.

- Linux Kernel + LEON Linux patches
- LEON Linux RAM loader (**mklinuximg**)
- Buildroot + LEON patches
- MKPROM2

Settings for standard LEON Linux configurations are available within the Linuxbuild package and can also be created by the user. Multiple build directories can be managed in order to test different configuration. The predefined configurations can be found in the `gaisler/configs` directory.

Note also that since each component is configured separately is is sometimes needed to set the same configuration option in multiple locations.

1.2. Requirements

- SPARC/LEON Linux Toolchain (Buildroot can be used to build a toolchain)
- MKPROM2 - for creating PROM/FLASH images
- wget
- git
- Internet access

Buildroot requires a number of tools such as bison, flex, msgfmt, makeinfo, etc. please see respective tool's homepage for requirements.

1.2.1. Installing Toolchain

Unless a custom toolchain is built with `crosstool-ng` or the Buildroot tool, the standard SPARC/LEON Linux toolchain should be installed before proceeding.

The GCC-4.4.2 multilib based toolchain is downloaded from the Aeroflex Gaisler FTP server at ftp://ftp.gaisler.com/gaisler.com/linux/linux-2.6/toolchains/sparc-linux-4.4.2/.

The toolchain is installed into the /opt directory creating the resulting directory `/opt/sparc-linux-x.y.z-toolchains/multilib`. The `bin` directory containing sparc-linux-gcc should be added to the PATH variable:

```
$ export PATH=/opt/sparc-linux-4.4.2-toolchains/multilib/bin:$PATH
$ which sparc-linux-gcc
/opt/sparc-linux-4.4.2-toolchains/multilib/bin/sparc-linux-gcc
```

Note that the toolchain path is hardcoded and cannot be installed to another directory.

1.3. Download Location

Table 1.1. Build steps

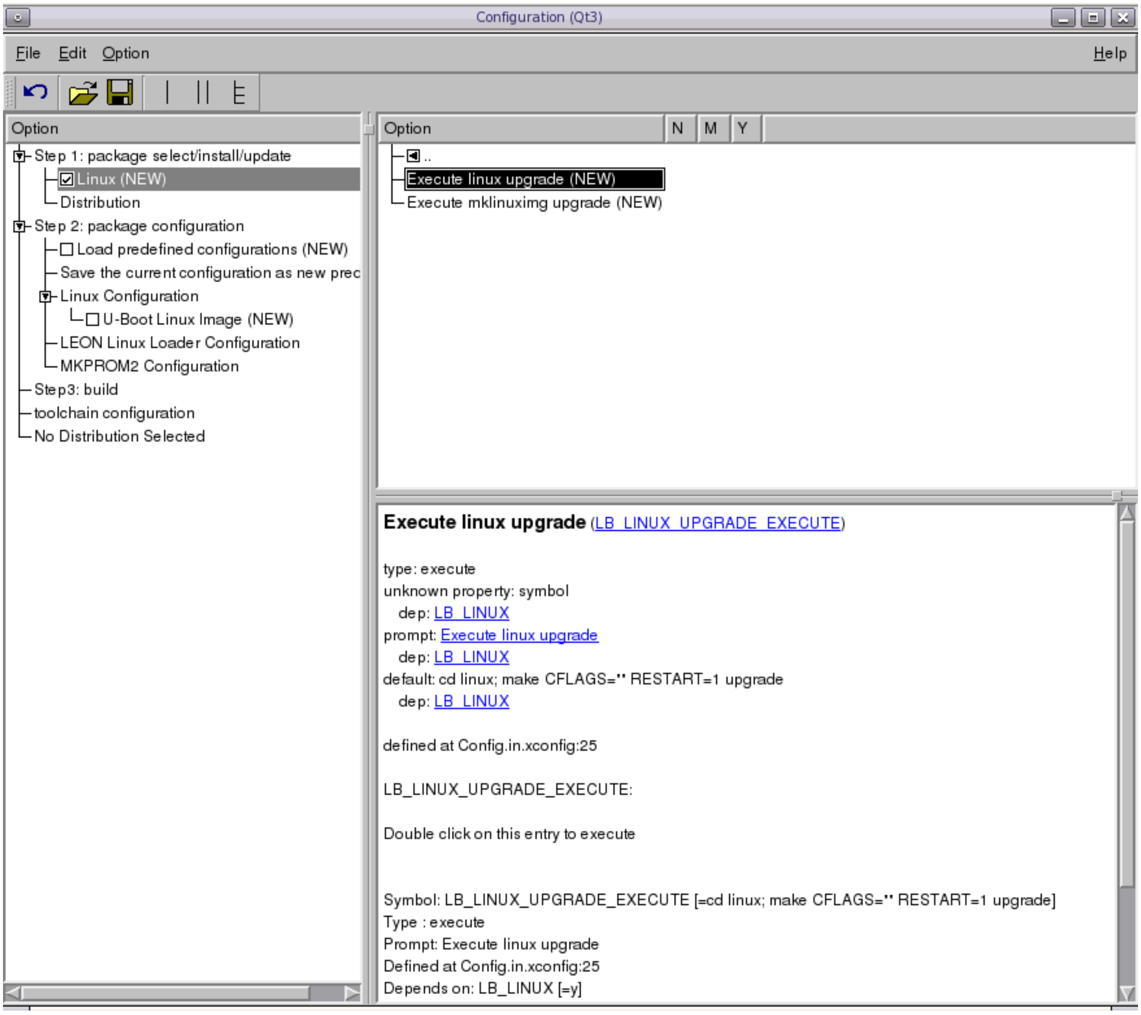
Tool	Download location
LINUXBUILD	ftp://gaisler.com/gaisler.com/linux/linux-2.6/linuxbuild/linuxbuild-x.y.z.tar.bz2
Linux toolchain	ftp://ftp.gaisler.com/gaisler.com/linux/linux-2.6/toolchains/sparc-linux-4.4.2/
MKPROM2	ftp://ftp.gaisler.com/gaisler.com/mkprom2/linux/mkprom2-2.0.36.tar.gz

2. Installing

Before installing Linuxbuild the SPARC/LEON Linux toolchain must be installed, unless Buildroot is used to build a uClibc toolchain. See Section 1.2.1.

After downloading the Linuxbuild package it is extracted using **tar -xf**, and the components are installed the first time by using the "upgrade" functionality of the KConfig GUI:

```
$ tar -xf linuxbuild-x.y.z.tar.bz2
$ cd linuxbuild-x.y.z
$ make xconfig
... do SELECT/INSTALL/UPGRADE PACKAGES...
```


Configuration (Qt3)
File
Edit
Option
Help
Option
Step 1: package select/install/update
☑ Linux (NEW)
Distribution
Step 2: package configuration
☐ Load predefined configurations (NEW)
Save the current configuration as new prec
Linux Configuration
☐ U-Boot Linux Image (NEW)
LEON Linux Loader Configuration
MKPROM2 Configuration
Step3: build
toolchain configuration
No Distribution Selected
Option
N
M
Y
..
Execute linux upgrade (NEW)
Execute mklinuximg upgrade (NEW)

Execute linux upgrade (LB_LINUX_UPGRADE_EXECUTE)

type: execute
unknown property: symbol
dep: LB_LINUX
prompt: Execute linux upgrade
dep: LB_LINUX
default: cd linux; make CFLAGS='' RESTART=1 upgrade
dep: LB_LINUX

defined at Config.in.xconfig:25

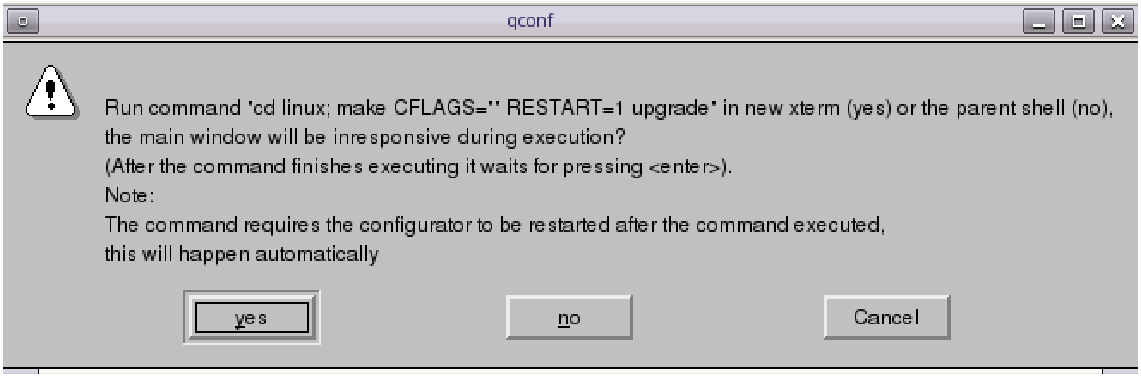
LB_LINUX_UPGRADE_EXECUTE:

Double click on this entry to execute

Symbol: LB_LINUX_UPGRADE_EXECUTE [=cd linux; make CFLAGS='' RESTART=1 upgrade]
Type : execute
Prompt: Execute linux upgrade
Defined at Config.in.xconfig:25
Depends on: LB_LINUX [=y]

Figure 2.1. Selecting Linux components for installing/upgrading

The configurator is modified from the original Kconfig to execute commands from within the GUI. I.e. when double-clicking on "Execute linux upgrade" a dialog will pop up asking weather to exeute the upgrade command in a new xterm or in the parent shell:



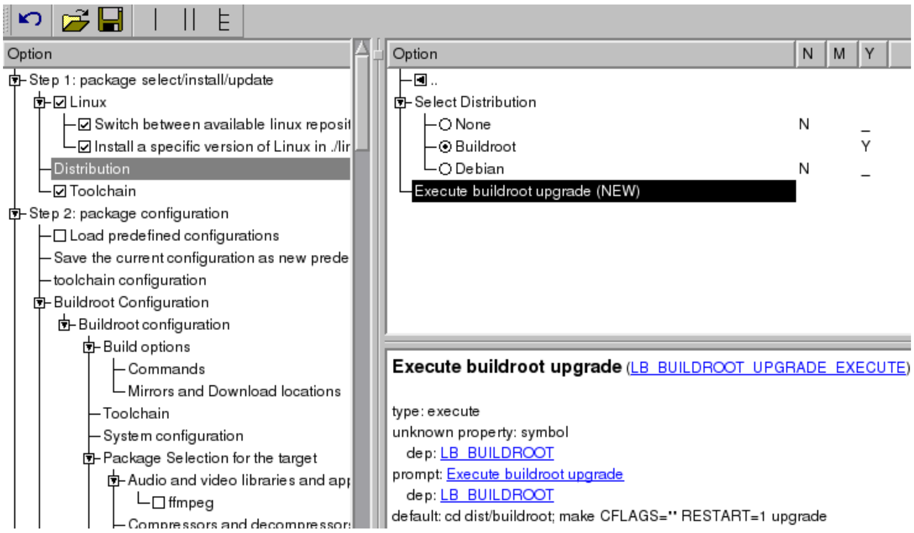
Run command 'cd linux; make CFLAGS='' RESTART=1 upgrade' in new xterm (yes) or the parent shell (no), the main window will be inresponsive during execution?
(After the command finishes executing it waits for pressing <enter>).
Note:
The command requires the configurator to be restarted after the command executed, this will happen automatically

Figure 2.2. Dialog asking weather to execute a command from within the GUI

When doing install/upgrade and when permanently saving a configuration the configurator is restarted automatically so that the configuration change is shown in the configurator.

2.1. Buildroot distribution

Below picture shows the upgrade command for the Buildroot sources:


Option
Step 1: package select/install/update
☑ Linux
☑ Switch between available linux reposit
☑ Install a specific version of Linux in ./lir
Distribution
☑ Toolchain
Step 2: package configuration
☐ Load predefined configurations
Save the current configuration as new prede
toolchain configuration
Buildroot Configuration
Buildroot configuration
Build options
Commands
Mirrors and Download locations
Toolchain
System configuration
Package Selection for the target
Audio and video libraries and app
☐ ffmpeg
Compressors and decompressors
Option
N
M
Y
..
Select Distribution

○ None N
● Buildroot Y
○ Debian N
Execute buildroot upgrade (NEW)

Execute buildroot upgrade (LB_BUILDROOT_UPGRADE_EXECUTE)

type: execute
unknown property: symbol
dep: LB_BUILDROOT
prompt: Execute buildroot upgrade
dep: LB_BUILDROOT
default: cd dist/buildroot; make CFLAGS='' RESTART=1 upgrade

Figure 2.3. Installing/upgrading distribution components

2.2. Debian distribution

A Sparc32 wheezy Debian package repository has been created at http://developer.gaisler.net/debian. The process of retriving the packages and installing them is done using the Multistrap utility from http://wiki.debian.org/Multistrap. To use multistrap by hand you would create a configuration file m-sparc.config

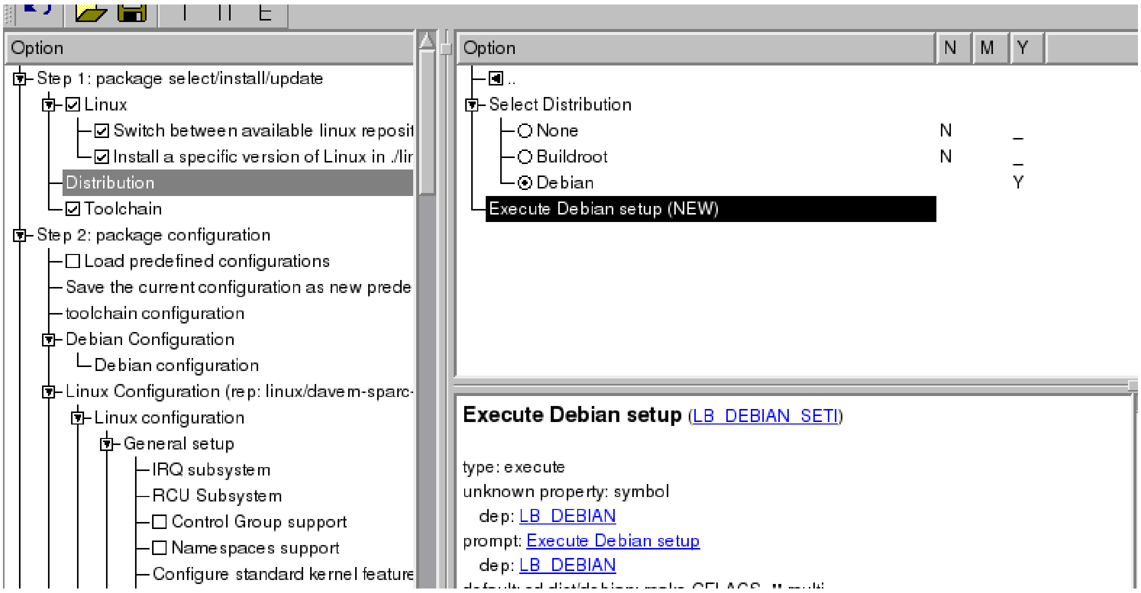
```
[General]
arch=sparc
directory=/tmp/dist-multistrap-w
cleanup=true
noauth=true
unpack=true
aptsources=Grip Updates
debootstrap=Debian

[Debian]
packages=
source=http://developer.gaisler.net/debian
suite=wheezy
```

and issue a

```
$multistrap -f m-sparc.config
```

The command run from Linuxbuild does some more steps:

- On a system that lacks apt-get download and install under [linux-build]/dist/debian/usr the apt and dpkg suite
- Multistrap requires perl modules Config::Auto, Parse::Debian::Packages and Locale::gettext. Linuxbuild will try to install them from CPAN.
- Download and install under [linux-build]/dist/debian/usr the newest version of tar and fakeroot
- Run multistrap
- Patch the created image directory with various /etc/* to be able to boot
- Create 2 versions, one [linux-build]/dist/debian/dist-multistrap-w-nfs that is big and that can be used to NFS boot, the other version [linux-build]/dist/debian/dist-multistrap-w-cpio that has some files removed to reduce size (still 20 MB).


Option
Step 1: package select/install/update
☑ Linux
☑ Switch between available linux reposit
☑ Install a specific version of Linux in ./lir
Distribution
☑ Toolchain
Step 2: package configuration
☐ Load predefined configurations
Save the current configuration as new prede
toolchain configuration
Debian Configuration
Debian configuration
Linux Configuration (rep: linux/davem-sparc-
Linux configuration
General setup
IRQ subsystem
RCU Subsystem
☐ Control Group support
☐ Namespaces support
Configure standard kernel feature
Option
N
M
Y
..
Select Distribution
○ None
○ Buildroot
● Debian
N
N
Y
Execute Debian setup (NEW)

Execute Debian setup (LB_DEBIAN_SETI)

type: execute
unknown property: symbol
dep: LB_DEBIAN
prompt: Execute Debian setup
dep: LB_DEBIAN

Figure 2.4. Setup from the debian package repository

After the setup the root cpio file should be located at [linuxbuild]/dist/debian/rootfs.cpio. Both [linux-build]/dist/debian/dist-multistrap-w-nfs and [linux-build]/dist/debian/dist-multistrap-w-cpio dont have all packages installed. To install new packages you can use dpkg. Here is a list summary of how to use dpkg:

- dpkg --contents [package].deb : list the files that are in the package
- dpkg --info [package].deb : list the control header, with dependencies etc.
- dpkg --extact [package].deb dir : extracts the archive to [dir]
- dpkg -i [--force-all] [package].deb : install a package (not using apt-get). If --force-all is given, dependencies are not checked.
- dpkg -l : list all installed package and status
- dpkg --search [full-path] : determine the package [full-path] belongs to

You can also manually extract the content of a package by using "ar xv [package]". This will extract data.tar.gz from [package] which is an tar archive relative to /.

3. Configuring

After the selected components have been downloaded or upgraded, Linuxbuild and each selected component is configured using one of the following make targets (**make xconfig**):

- xconfig - Qt based GUI (Qt-3/4 libs required)
- gconfig - GTK based GUI
- menuconfig - ncurses based terminal interface

The buildroot and Linux configuration trees are displayed as subtrees inside the main screen:

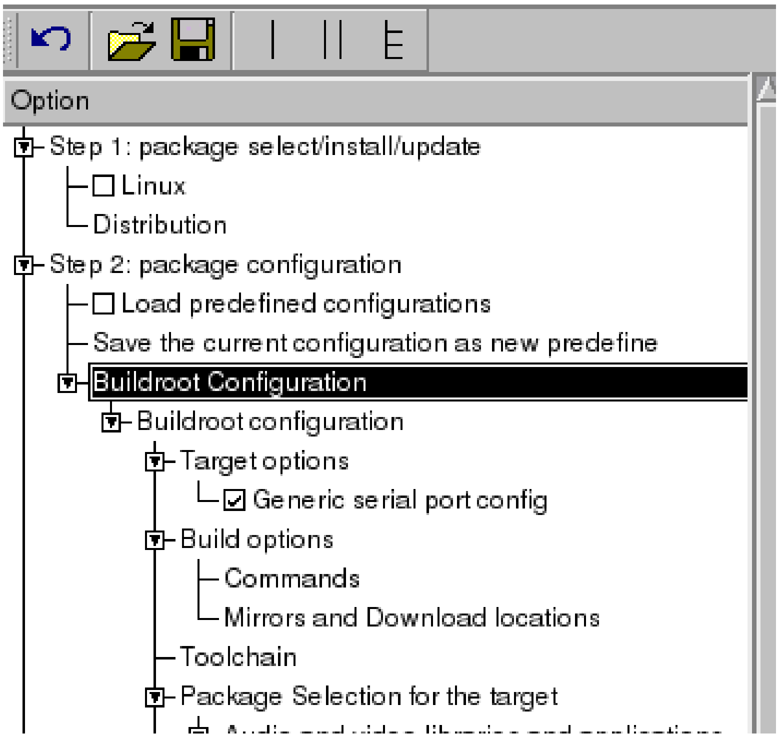
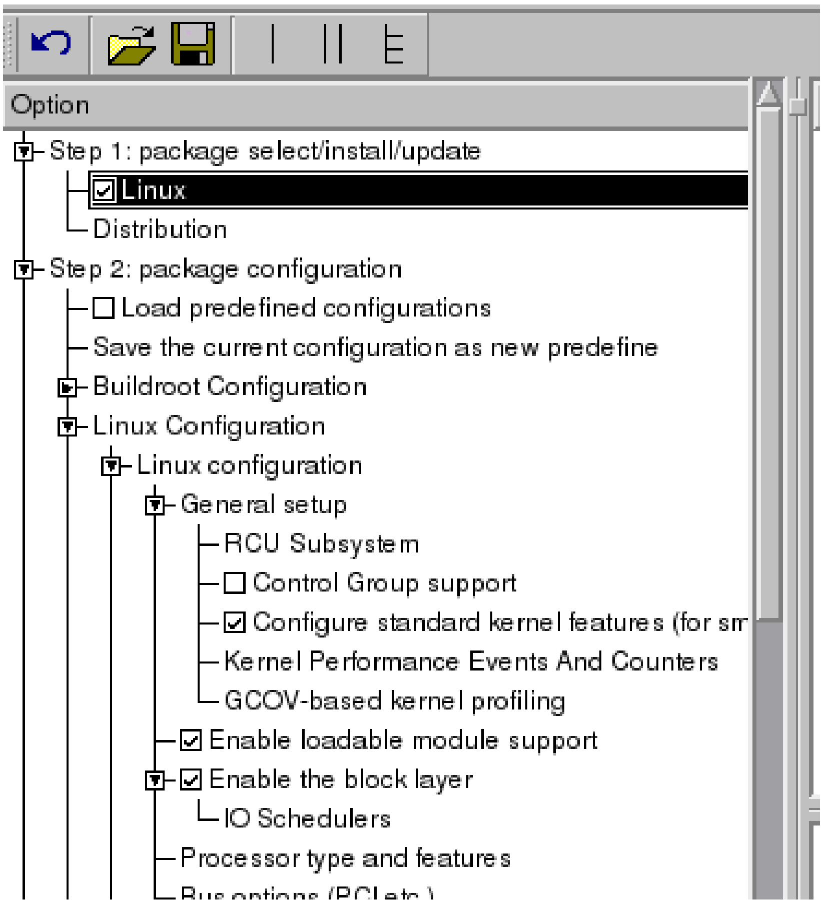

Option
Step 1: package select/install/update
☐ Linux
Distribution
Step 2: package configuration
☐ Load predefined configurations
Save the current configuration as new predefine
Buildroot Configuration
Buildroot configuration
Target options
☑ Generic serial port config
Build options
Commands
Mirrors and Download locations
Toolchain
Package Selection for the target

Figure 3.1. The buildroot configuration tree inside the main configuration tree



- Step 1: package select/install/update
  - ☑ Linux
  - Distribution
- Step 2: package configuration
  - ☐ Load predefined configurations
  - Save the current configuration as new predefine
  - Buildroot Configuration
  - Linux Configuration
    - Linux configuration
      - General setup
        - RCU Subsystem
        - ☐ Control Group support
        - ☑ Configure standard kernel features (for sm
        - Kernel Performance Events And Counters
        - GCOV-based kernel profiling
      - ☑ Enable loadable module support
      - ☑ Enable the block layer
        - IO Schedulers
      - Processor type and features

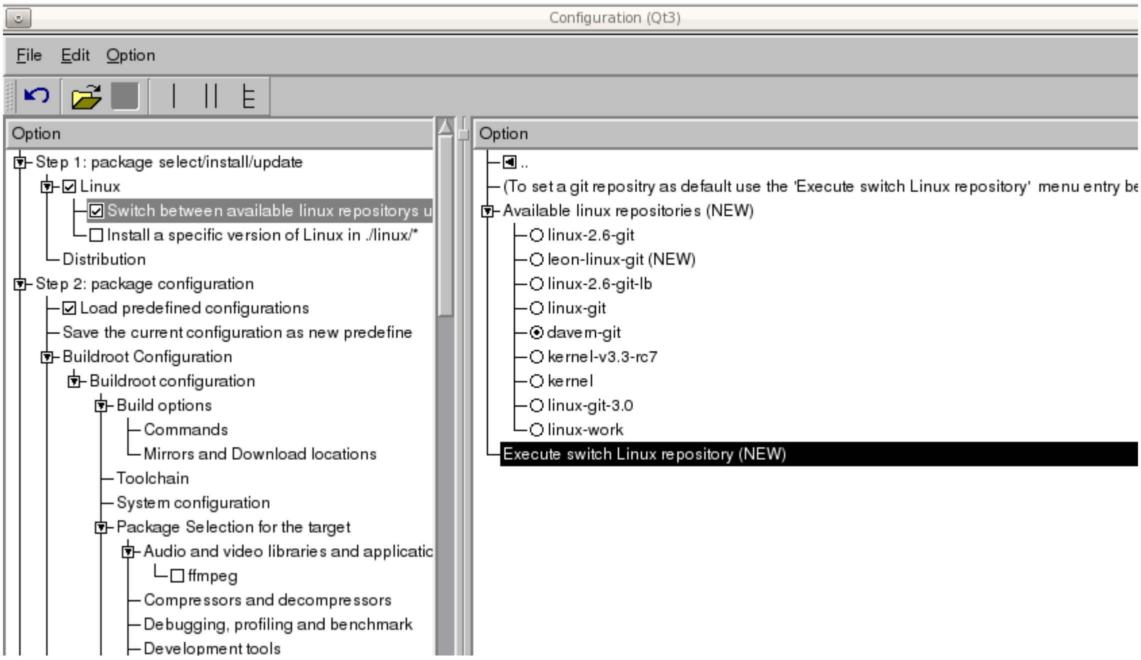
Figure 3.2. The LINUX configuration tree inside the main configuration tree

Prepared LEON configurations can be found in `gaisler/configs`, they can be loaded by Linuxbuild under the "Save/Load Configuration" in the GUI. Loading a configuration updates the current configuration of all components. For this to work all enabled components must have been installed/upgraded.

Note that since each component is configured separately is is sometimes needed to set the same configuration option more than once in different configuration GUIs.

3.1. Multiple Linux repositories

When starting the configurator the subdirectory ./linux/ is scanned for available git repositories. The symlink ./linux/linux-work will point to the active Linux repository, which is also the one where the Linux configuration tree will be loded from. To switch ./linux/linux-work execute the "Execute switch Linux repository" entry as shown below:



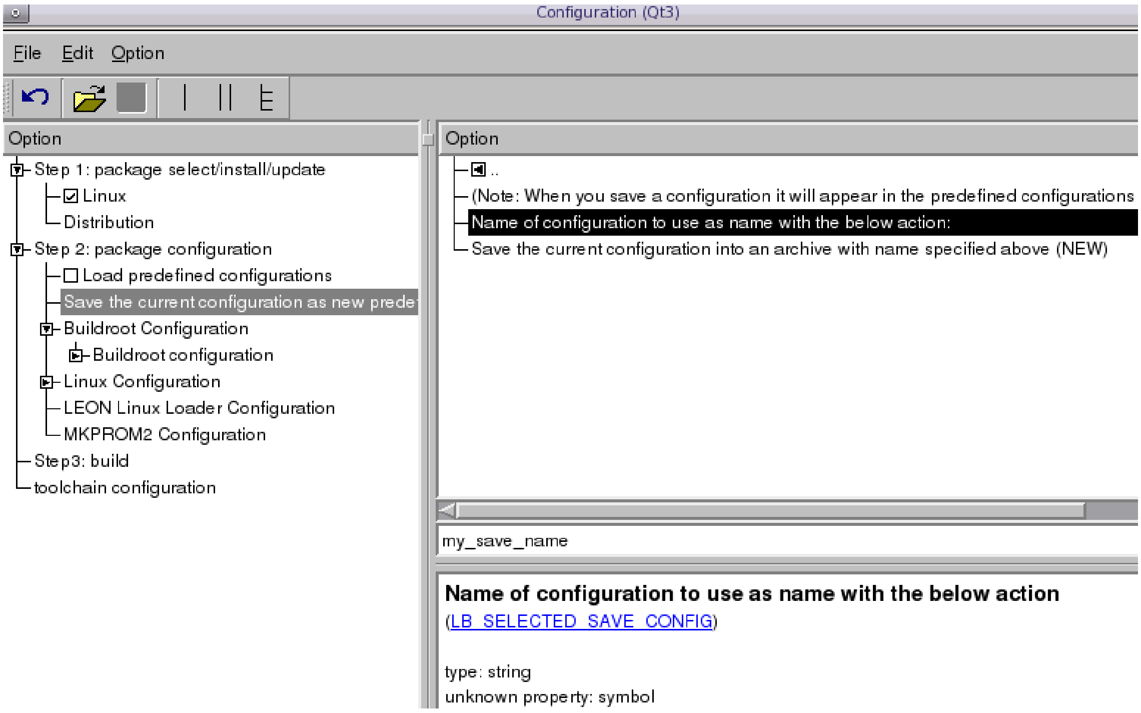
Configuration (Qt3)
File Edit Option
Option
Step 1: package select/install/update
☑ Linux
☑ Switch between available linux repositorys u
☐ Install a specific version of Linux in ./linux/*
Distribution
Step 2: package configuration
☑ Load predefined configurations
Save the current configuration as new predefine
Buildroot Configuration
Buildroot configuration
Build options
Commands
Mirrors and Download locations
Toolchain
System configuration
Package Selection for the target
Audio and video libraries and applicatic
☐ ffmpeg
Compressors and decompressors
Debugging, profiling and benchmark
Development tools
Option
..
Available linux repositories (NEW)
○ linux-2.6-git
○ leon-linux-git (NEW)
○ linux-2.6-git-lb
○ linux-git
● davem-git
○ kernel-v3.3-rc7
○ kernel
○ linux-git-3.0
○ linux-work
Execute switch Linux repository (NEW)

(To set a git repositry as default use the 'Execute switch Linux repository' menu entry be

Figure 3.3. Select the repository to set as default

3.2. Save/Load configurations

To save the current configuration first specify a name:



Configuration (Qt3)
File Edit Option
Option
Step 1: package select/install/update
☑ Linux
Distribution
Step 2: package configuration
☐ Load predefined configurations
Save the current configuration as new prede
Buildroot Configuration
Buildroot configuration
Linux Configuration
LEON Linux Loader Configuration
MKPROM2 Configuration
Step3: build
toolchain configuration
Option
..
Name of configuration to use as name with the below action:
Save the current configuration into an archive with name specified above (NEW)

(Note: When you save a configuration it will appear in the predefined configurations

my_save_name

Name of configuration to use as name with the below action

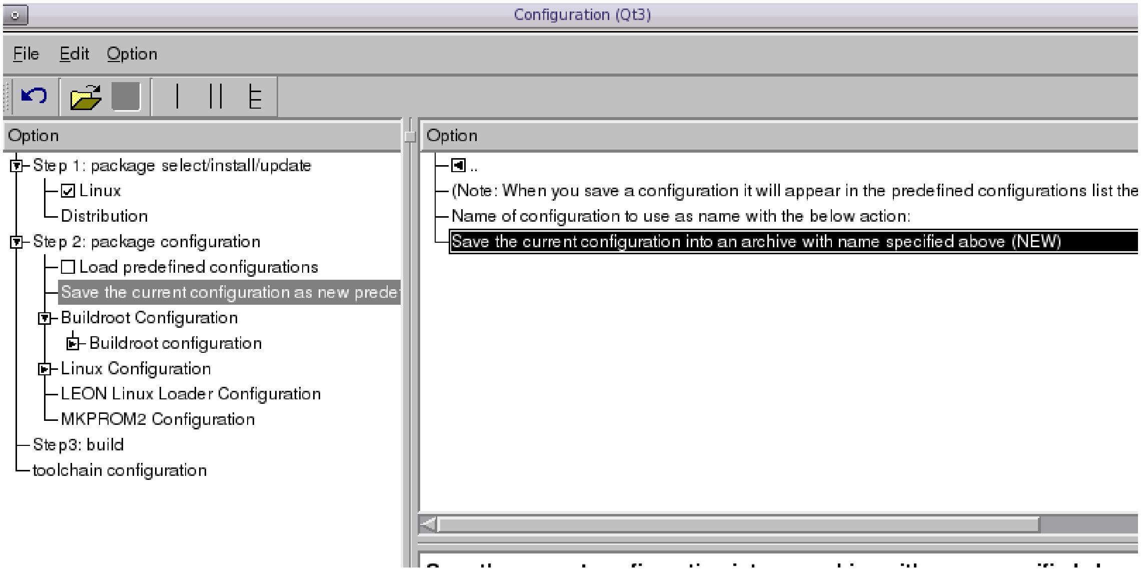
(LB_SELECTED_SAVE_CONFIG)

type: string

unknown property: symbol

Figure 3.4. Specify the name of the configuration to save

After that execute the save command:

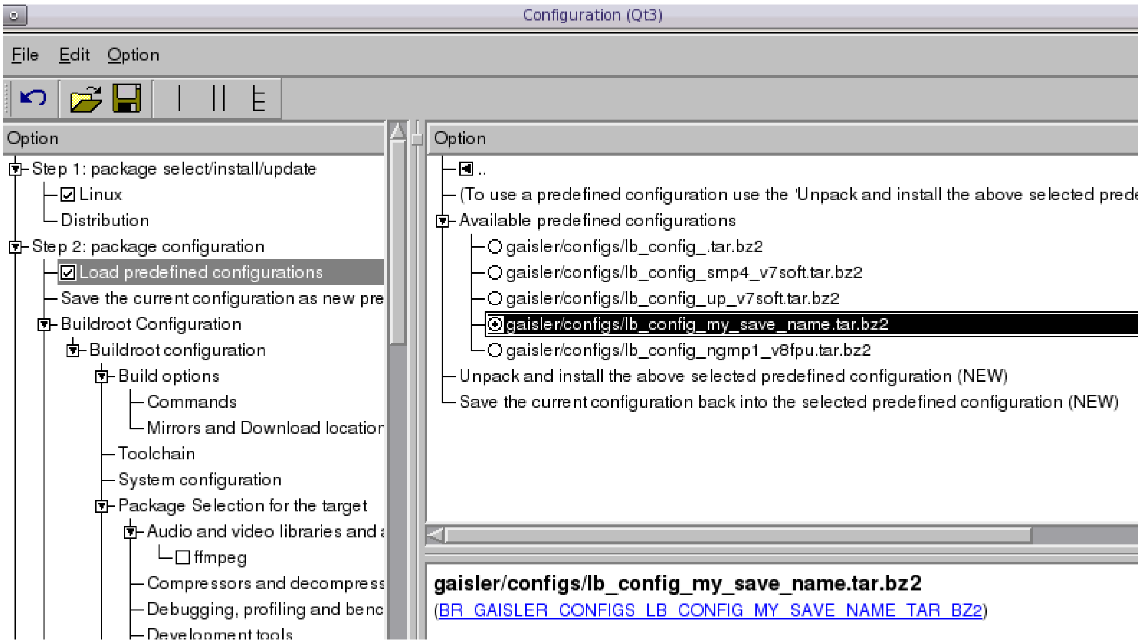


- Step 1: package select/install/update
  - ☑ Linux
  - Distribution
- Step 2: package configuration
  - ☐ Load predefined configurations
  - Save the current configuration as new predef
  - Buildroot Configuration
    - Buildroot configuration
  - Linux Configuration
  - LEON Linux Loader Configuration
  - MKPROM2 Configuration
- Step3: build
- toolchain configuration

..
(Note: When you save a configuration it will appear in the predefined configurations list the
Name of configuration to use as name with the below action:
Save the current configuration into an archive with name specified above (NEW)

Figure 3.5. Execute the save command

The configurator will restart automatically. After restart the newly saved configuration is accessible in the "Load predefined configurations" entry:



- Step 1: package select/install/update
  - ☑ Linux
  - Distribution
- Step 2: package configuration
  - ☑ Load predefined configurations
  - Save the current configuration as new pre
  - Buildroot Configuration
    - Buildroot configuration
      - Build options
        - Commands
        - Mirrors and Download locatior
      - Toolchain
      - System configuration
      - Package Selection for the target
        - Audio and video libraries and a
          - ☐ ffmpeg
        - Compressors and decompress
        - Debugging, profiling and benc
        - Development tools

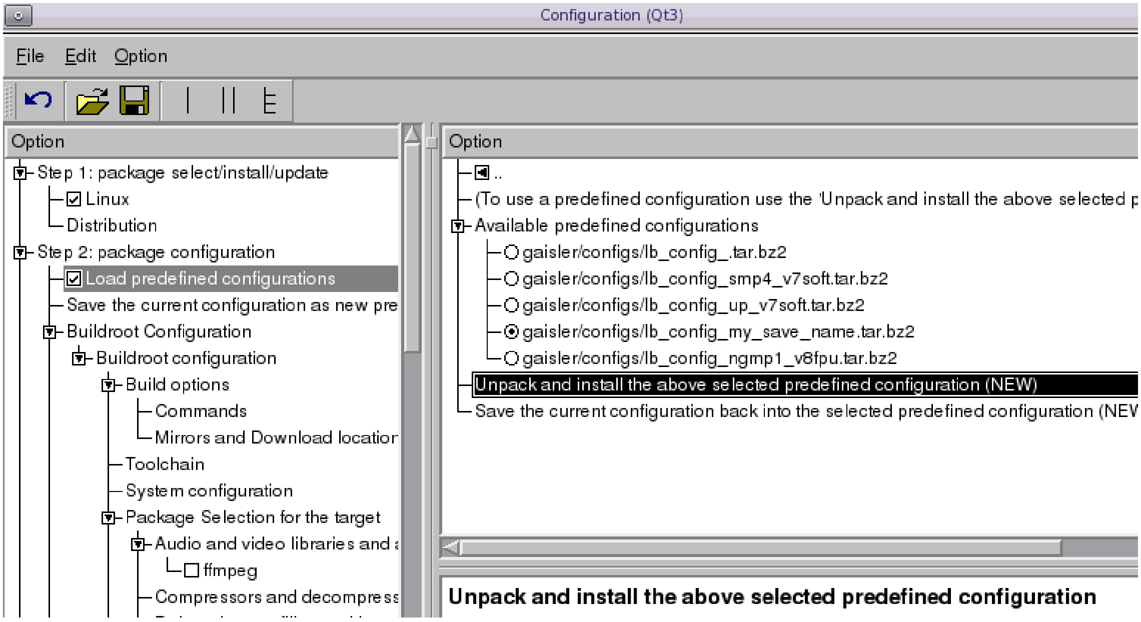
..
(To use a predefined configuration use the 'Unpack and install the above selected prede
Available predefined configurations
- ○ gaisler/configs/lb_config_.tar.bz2
- ○ gaisler/configs/lb_config_smp4_v7soft.tar.bz2
- ○ gaisler/configs/lb_config_up_v7soft.tar.bz2
- ● gaisler/configs/lb_config_my_save_name.tar.bz2
- ○ gaisler/configs/lb_config_ngmp1_v8fpu.tar.bz2

Unpack and install the above selected predefined configuration (NEW)
Save the current configuration back into the selected predefined configuration (NEW)

gaisler/configs/lb_config_my_save_name.tar.bz2
(BR_GAISLER_CONFIGS_LB_CONFIG_MY_SAVE_NAME_TAR_BZ2)

Figure 3.6. List of saved configurations

You can unpack and load the selected configuration:



Step 1: package select/install/update
☑ Linux
Distribution
Step 2: package configuration
☑ Load predefined configurations
Save the current configuration as new pre
Buildroot Configuration
Buildroot configuration
Build options
Commands
Mirrors and Download locatior
Toolchain
System configuration
Package Selection for the target
Audio and video libraries and a
☐ ffmpeg
Compressors and decompress

..
(To use a predefined configuration use the 'Unpack and install the above selected p
Available predefined configurations
○ gaisler/configs/lb_config_.tar.bz2
○ gaisler/configs/lb_config_smp4_v7soft.tar.bz2
○ gaisler/configs/lb_config_up_v7soft.tar.bz2
● gaisler/configs/lb_config_my_save_name.tar.bz2
○ gaisler/configs/lb_config_ngmp1_v8fpu.tar.bz2
Unpack and install the above selected predefined configuration (NEW)
Save the current configuration back into the selected predefined configuration (NEV

Unpack and install the above selected predefined configuration

Figure 3.7. Load a saved configuration

3.3. Buildroot

Buildroot is used to build user-space applications and tool chains, Aeroflex Gaisler does not distribute uClibc tool chains at the time of writing, in order to build one self Buildroot can be used. In either case a toolchain used to build user-space applications must be selected, see the table below for a number of prepared configurations. The Buildroot configuration GUI is entered by selecting LB_BUILDROOT_CONFIGURE (first selecting Buildroot LB_BUILDROOT).

Table 3.1. Prepared Buildroot Toolchain Configurations

Config	**Builds Toolchain**	**Libc**	**Target**	**Toolchain Location**
sfleon_shared_basic	YES	uClibc	soft-float/ v7	Buildroot `builddir/output/staging`
sfleonv8_shared_basic	YES	uClibc	soft-float/ v8	Buildroot `builddir/output/staging`
hfleon_shared_basic	YES	uClibc	hard-float/ v7	Buildroot `builddir/output/staging`
hfleonv8_shared_basic	YES	uClibc	hard-float/ v8	Buildroot `builddir/output/staging`
hfleonv8_glibc_basic	NO	GLIBC	hard-float/ v8	`/opt/sparc-linux-toolchains/hfleonv8`
sfleon_glibc_basic	NO	GLIBC	soft-float/ v7	`/opt/sparc-linux-toolchains/sfleon`
sfleon_multilib_glibc_basic	NO	GLIBC	soft-float/ v7	`/opt/sparc-linux-toolchains/multilib`
sfleonv8_multilib_glibc_basic	NO	GLIBC	soft-float/ v8	`/opt/sparc-linux-toolchains/multilib`
hfleon_multilib_glibc_basic	NO	GLIBC	hard-float/ v7	`/opt/sparc-linux-toolchains/multilib`
hfleonv8_multilib_glibc_basic	NO	GLIBC	hard-float/ v8	`/opt/sparc-linux-toolchains/multilib`

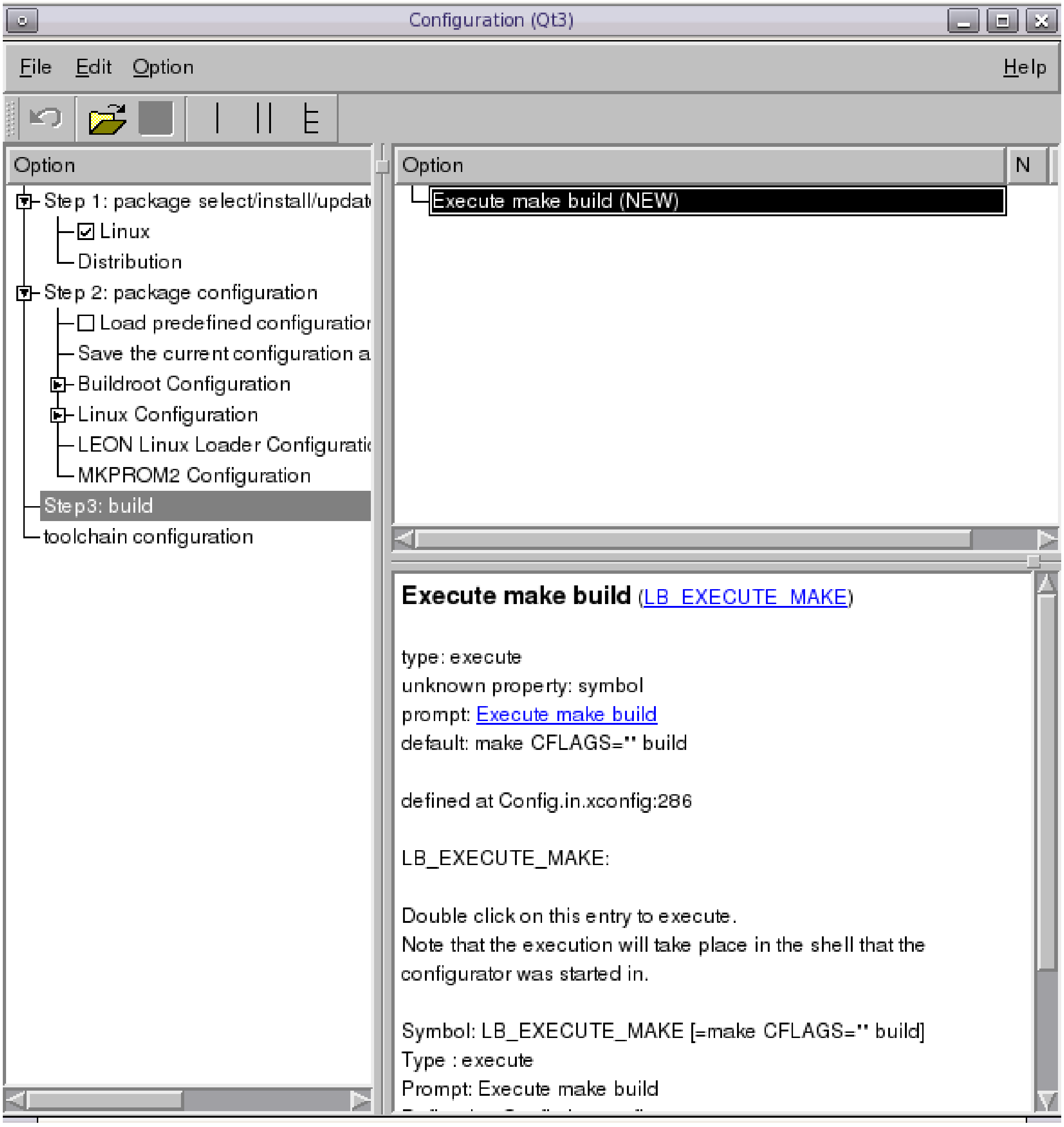
3.3.1. Network Configuration

After building the Buildroot file system the first time the file system content is located in the Buildroot build directory `build-br/target`. Adding network settings for the network interfaces can be done by editing the `/etc/network/interfaces` file, for example setting eth0 in DHCP and eth1 to a static IP address is done by editing the interfaces file as follows:

```
# Configure Loopback
auto lo
iface lo inet loopback

# Do DHCP for ETH0
auto eth0
iface eth0 inet dhcp

# Static IP for ETH1
auto eth1
iface eth1 inet static
    address 192.168.1.207
    network 192.168.1.0
    netmask 255.255.255.0
    broadcast 192.168.1.255
    gateway 192.168.1.1
```

4. Building

After configuring the Linuxbuild and all the selected components the build process is started by typing **make build** or by doubleclicking the build command inside the GUI:



Execute make build (LB_EXECUTE_MAKE)

type: execute
unknown property: symbol
prompt: Execute make build
default: make CFLAGS='' build

defined at Config.in.xconfig:286

LB_EXECUTE_MAKE:

Double click on this entry to execute.
Note that the execution will take place in the shell that the configurator was started in.

Symbol: LB_EXECUTE_MAKE [=make CFLAGS='' build]
Type : execute
Prompt: Execute make build

Figure 4.1. Execute build from within the GUI

If the build process fail it may be due to that a required tool is missing. The resulting images and file system images are found in the `output/` directory.

```
$ make build
...
```

5. Upgrading

Individual components can be upgraded when Aeroflex Gaisler release new updated packages, the upgrade process can be started from the KConfig GUI (make xconfig).

If a package requires the Linuxbuild itself to be upgraded the user must do that manually. So save time the `linux/linux-git` and `dist/buildroot/buildroot-git` trees can be moved from the old Linuxbuild directory to the new, also the current configuration can be saved from the xconfig GUI and loaded into the new Linuxbuild installation.

6. Support

For Support, contact the Aeroflex Gaisler support team at support@gaisler.com.