A Reproducible Mp3-Player application using Build Root

Contents

Table of figures

Description

This project is an assignment-series aimed to develop a functional Mp3 player reproducible by build root (without any manual modifications after the build) using all the knowledge and tools we got so far in the course and with Extra help from the Open source community of Linux.

Main Features of developed MP3 Player:

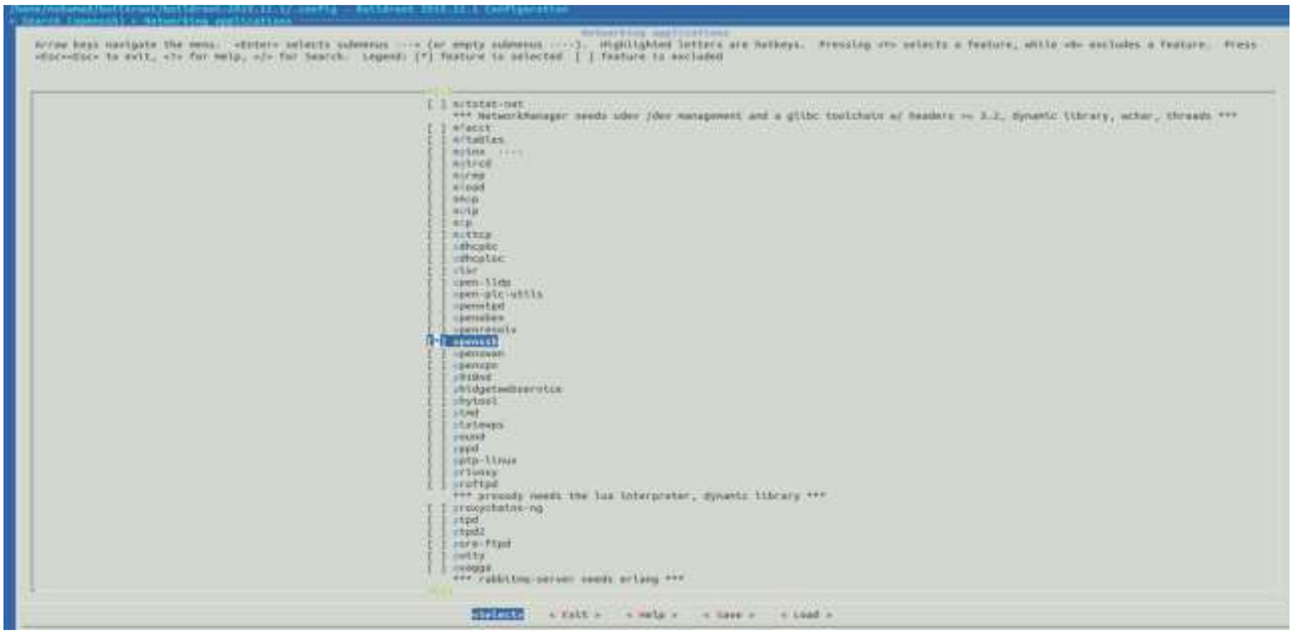
- Auto detection of mass-storage devices that can be connected via USB (e.g. Flash, USB-HDD, Memory Card).
- Manual Control of Music Player (Play/Pause, Skip, Rewind, Shuffle), either via Push Buttons or Keyboard Commands.
- Ability to play music on different output devices (3.5mm Audio Jack, Bluetooth Audio, HDMI).
- Streaming of the currently played music on the terminal (SSH terminal over Ethernet/Wi-Fi, Serial terminal).
- Audio Notifications about System changes (e.g. Inserting/removing storage/audio devices).

Work Development Steps in building the image:

Enabling the SSH

This is achieved using build root menuconfig as follows:

Target packages →Networking application → openssh



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> Search (openssh) > Networking applications
                                  Networking applications
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                                 -(-)-
                                 [ ] netstat-nat
                                     *** NetworkManager needs udev /dev management and a glibc toolchain w/ headers >= 3.2, dynamic library, wchar, threads ***
                                 [ ] nfacct
                                 [ ] nftables
                                 [ ] nginx  ----
                                 [ ] ngircd
                                 [ ] ngrep
                                 [ ] nload
                                 [ ] nmap
                                 [ ] noip
                                 [ ] ntp
                                 [ ] nuttcp
                                 [ ] odhcp6c
                                 [ ] odhcploc
                                 [ ] olsr
                                 [ ] open-lldp
                                 [ ] open-plc-utils
                                 [ ] openntpd
                                 [ ] openobex
                                 [ ] openresolv
                                 [*] openssh
                                 [ ] openswan
                                 [ ] openvpn
                                 [ ] p910nd
                                 [ ] phidgetwebservice
                                 [ ] phytool
                                 [ ] pimd
                                 [ ] pixiewps
                                 [ ] pound
                                 [ ] pppd
                                 [ ] pptp-linux
                                 [ ] privoxy
                                 [ ] proftpd
                                     *** prosody needs the lua interpreter, dynamic library ***
                                 [ ] proxychains-ng
                                 [ ] ptpd
                                 [ ] ptpd2
                                 [ ] pure-ftpd
                                 [ ] putty
                                 [ ] quagga
                                     *** rabbitmq-server needs erlang ***
                                 -(+)-

                    <Select>    < Exit >    < Help >    < Save >    < Load >
```

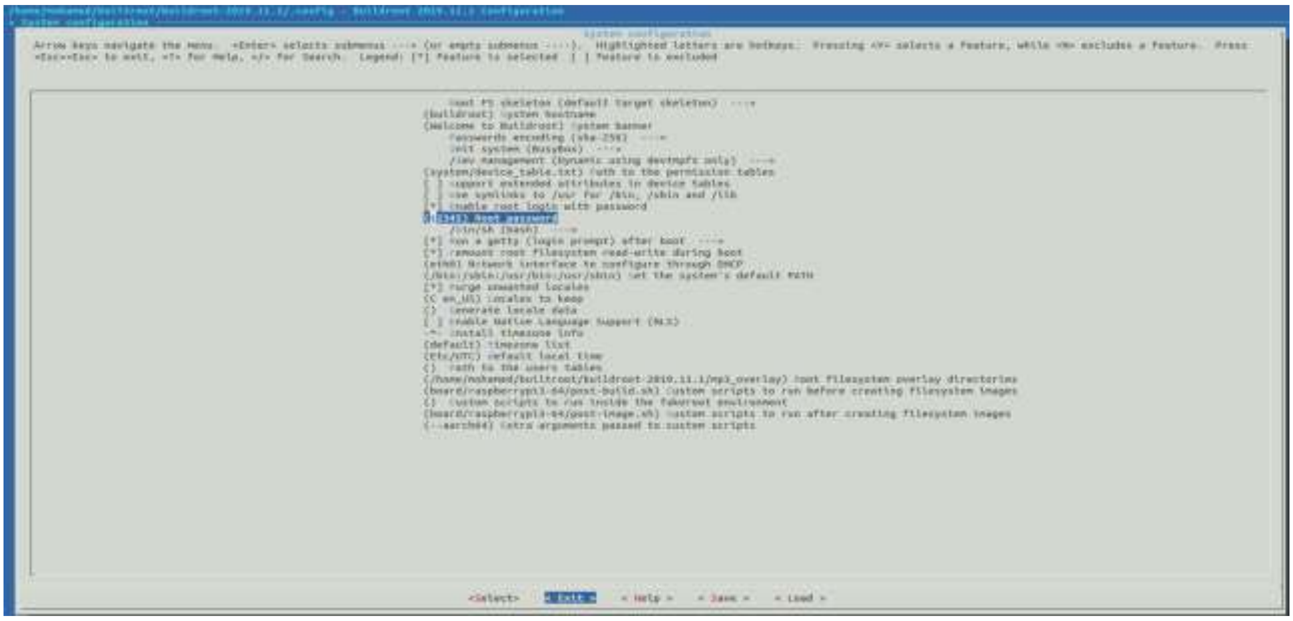
Figure 1 Enable openssh

Changing the root file system password and configuring the overlay folder

This is achieved using build root menuconfig as follows:

System configuration → root password→ put the password

System configuration → Root file systems overlay directories



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> System configuration
                                  System configuration
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                                     Root FS skeleton (default target skeleton)  --->
                                 (buildroot) System hostname
                                 (Welcome to Buildroot) System banner
                                     Passwords encoding (sha-256)  --->
                                     Init system (BusyBox)  --->
                                     /dev management (Dynamic using devtmpfs only)  --->
                                 (system/device_table.txt) Path to the permission tables
                                 [ ] support extended attributes in device tables
                                 [ ] Use symlinks to /usr for /bin, /sbin and /lib
                                 [*] Enable root login with password
                                 ([illegible]) Root password
                                     /bin/sh (bash)  --->
                                 [*] Run a getty (login prompt) after boot  --->
                                 [*] remount root filesystem read-write during boot
                                 (eth0) Network interface to configure through DHCP
                                 (/bin:/sbin:/usr/bin:/usr/sbin) Set the system's default PATH
                                 [*] Purge unwanted locales
                                 (C en_US) Locales to keep
                                 ()  Generate locale data
                                 [ ] Enable Native Language Support (NLS)
                                 -*- Install timezone info
                                 (default) timezone list
                                 (Etc/UTC) default local time
                                 ()  Path to the users tables
                                 (/home/mohamed/buildroot/buildroot-2019.11.1/mp3_overlay) Root filesystem overlay directories
                                 (board/raspberrypi3-64/post-build.sh) Custom scripts to run before creating filesystem images
                                 ()  Custom scripts to run inside the fakeroot environment
                                 (board/raspberrypi3-64/post-image.sh) Custom scripts to run after creating filesystem images
                                 (--aarch64) Extra arguments passed to custom scripts

                    <Select>    < Exit >    < Help >    < Save >    < Load >
```

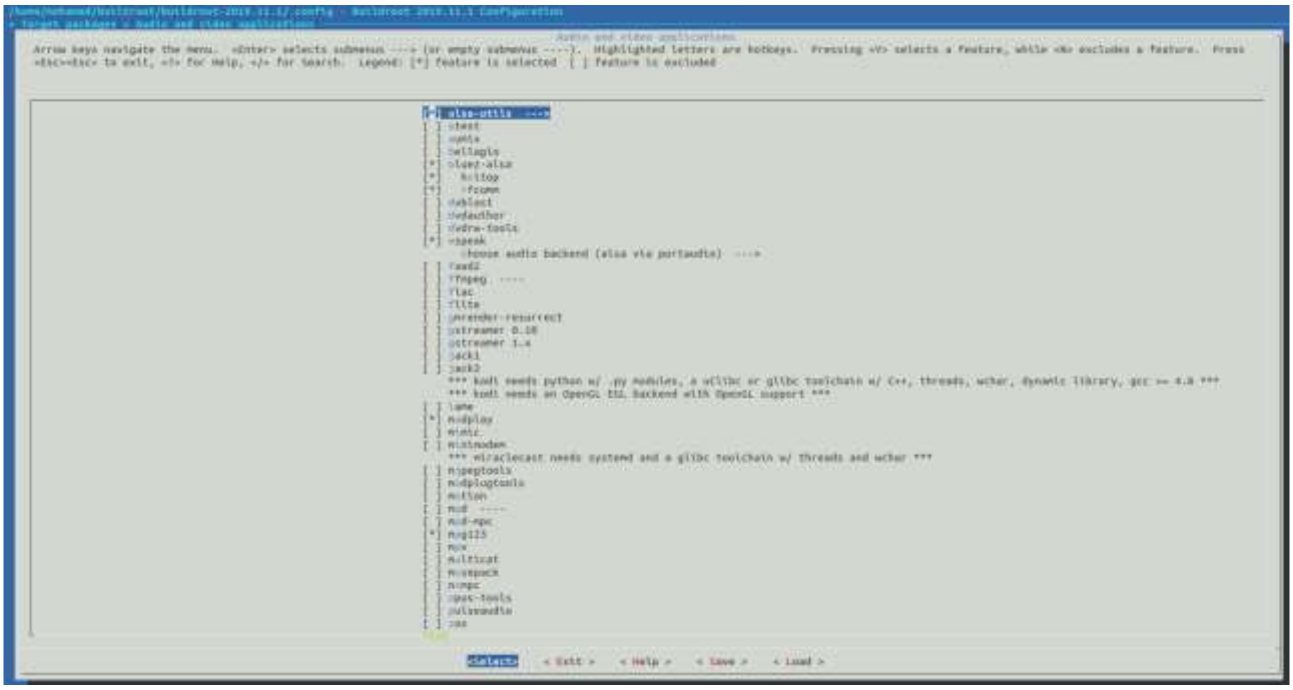
Figure 2 configure the root system password

Enabling sound driver related packages Alsa, madplay, mpg123, bluez-alsa

This is achieved using build root menuconfig as follows:

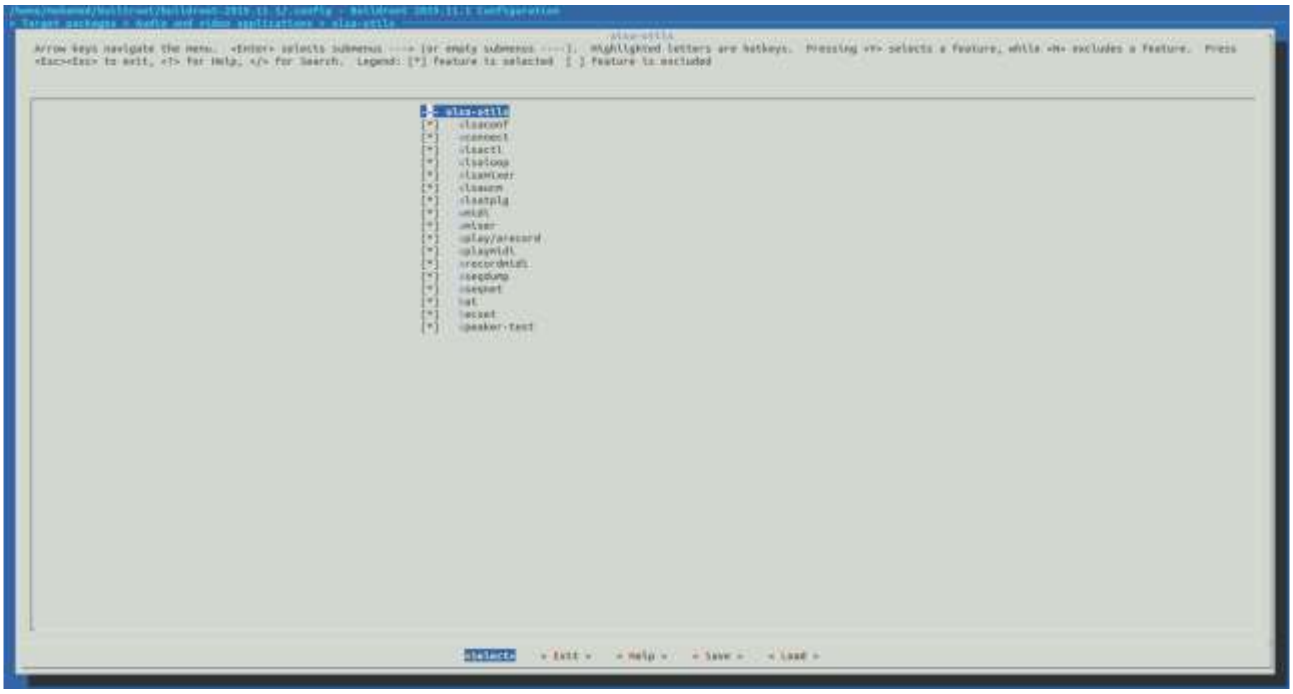
Target packages → audio and video application →

- Select Alsa utils and select all inside alsa-utils
- Select madplay
- Select mpg123
- Select bluez-alsa



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> Target packages > Audio and video applications
                                  Audio and video applications
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                     [*] alsa-utils  --->
                     [ ] atest
                     [ ] aumix
                     [ ] bellagio
                     [*] bluez-alsa
                     [*]   hcitop
                     [*]   rfcomm
                     [ ] dvblast
                     [ ] dvdauthor
                     [ ] dvdrw-tools
                     [*] espeak
                           Choose audio backend (alsa via portaudio)  --->
                     [ ] faad2
                     [ ] ffmpeg  ----
                     [ ] flac
                     [ ] flite
                     [ ] gmrender-resurrect
                     [ ] gstreamer 0.10
                     [ ] gstreamer 1.x
                     [ ] jack1
                     [ ] jack2
                         *** kodi needs python w/ .py modules, a uClibc or glibc toolchain w/ C++, threads, wchar, dynamic library, gcc >= 4.8 ***
                         *** kodi needs an OpenGL ES. backend with OpenGL support ***
                     [ ] lame
                     [*] madplay
                     [ ] mimic
                     [ ] minimodem
                         *** miraclecast needs systemd and a glibc toolchain w/ threads and wchar ***
                     [ ] mjpegtools
                     [ ] modplugtools
                     [ ] motion
                     [ ] mpd  ----
                     [ ] mpd-mpc
                     [*] mpg123
                     [ ] mpv
                     [ ] multicat
                     [ ] musepack
                     [ ] ncmpc
                     [ ] opus-tools
                     [ ] pulseaudio
                     [ ] sox

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 4 Enable madplay mpg123

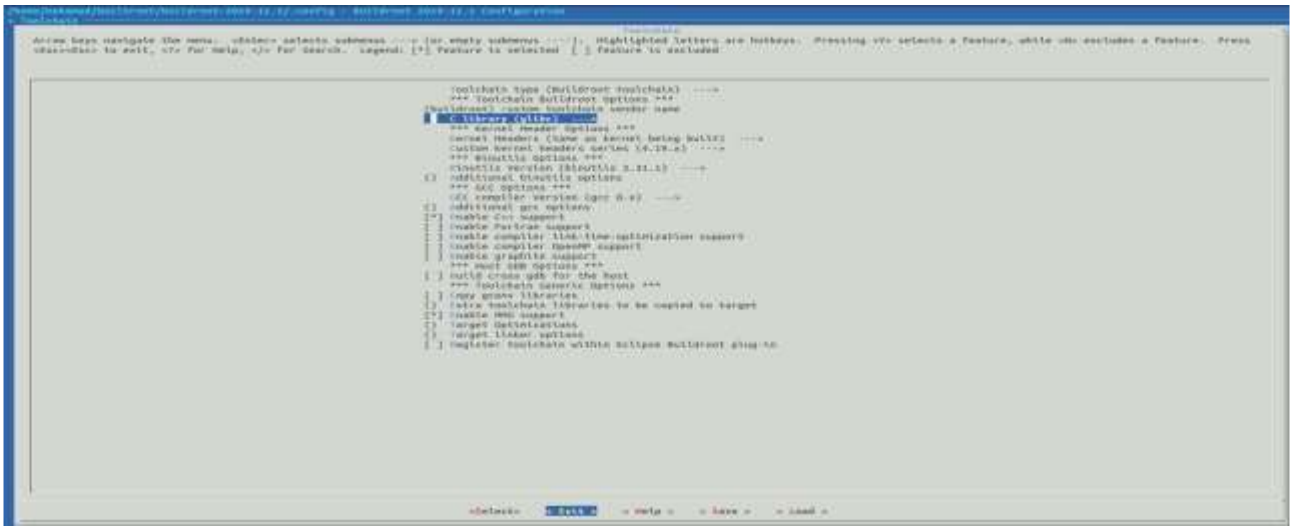


```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> Target packages > Audio and video applications > alsa-utils
                                  alsa-utils
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                     --- alsa-utils
                     [*]   alsaconf
                     [*]   aconnect
                     [*]   alsactl
                     [*]   alsaloop
                     [*]   alsamixer
                     [*]   alsaucm
                     [*]   alsatplg
                     [*]   amidi
                     [*]   amixer
                     [*]   aplay/arecord
                     [*]   aplaymidi
                     [*]   arecordmidi
                     [*]   aseqdump
                     [*]   aseqnet
                     [*]   bat
                     [*]   iecset
                     [*]   speaker-test

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 3 alsa utils

Changing the C library from ulibc to glibc

This is achieved using build root menuconfig as follows:
Toolchain c library → select glibc instate of ulibc

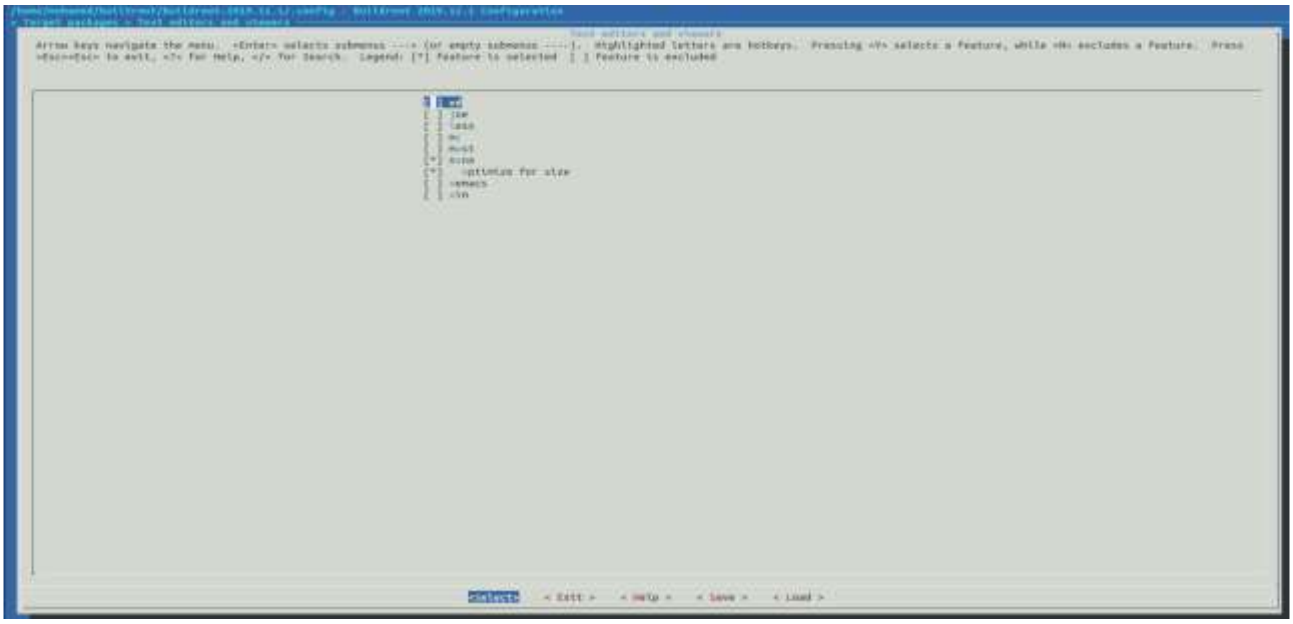


```
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
<Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

    Toolchain type (Buildroot toolchain)  --->
    *** Toolchain Buildroot Options ***
(buildroot) custom toolchain vendor name
    C library (glibc)  --->
    *** Kernel Header Options ***
    Kernel Headers (Same as kernel being built)  --->
    Custom kernel headers series (4.19.x)  --->
    *** Binutils Options ***
    Binutils Version (binutils 2.31.1)  --->
()  Additional binutils options
    *** GCC Options ***
    GCC compiler Version (gcc 8.x)  --->
()  Additional gcc options
[*] Enable C++ support
[ ] Enable Fortran support
[ ] Enable compiler link-time-optimization support
[ ] Enable compiler OpenMP support
[ ] Enable graphite support
    *** Host GDB Options ***
[ ] Build cross gdb for the host
    *** Toolchain Generic Options ***
[ ] Copy gconv libraries
()  Extra toolchain libraries to be copied to target
[*] Enable MMU support
()  Target Optimizations
()  Target linker options
[ ] Register toolchain within Eclipse Buildroot plug-in

        <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 5 changing the ulibc to ulibc

Changing the default editor from vi to nano

This is achieved using build root menuconfig as follows:
Target packages → text editors and viewers → nano



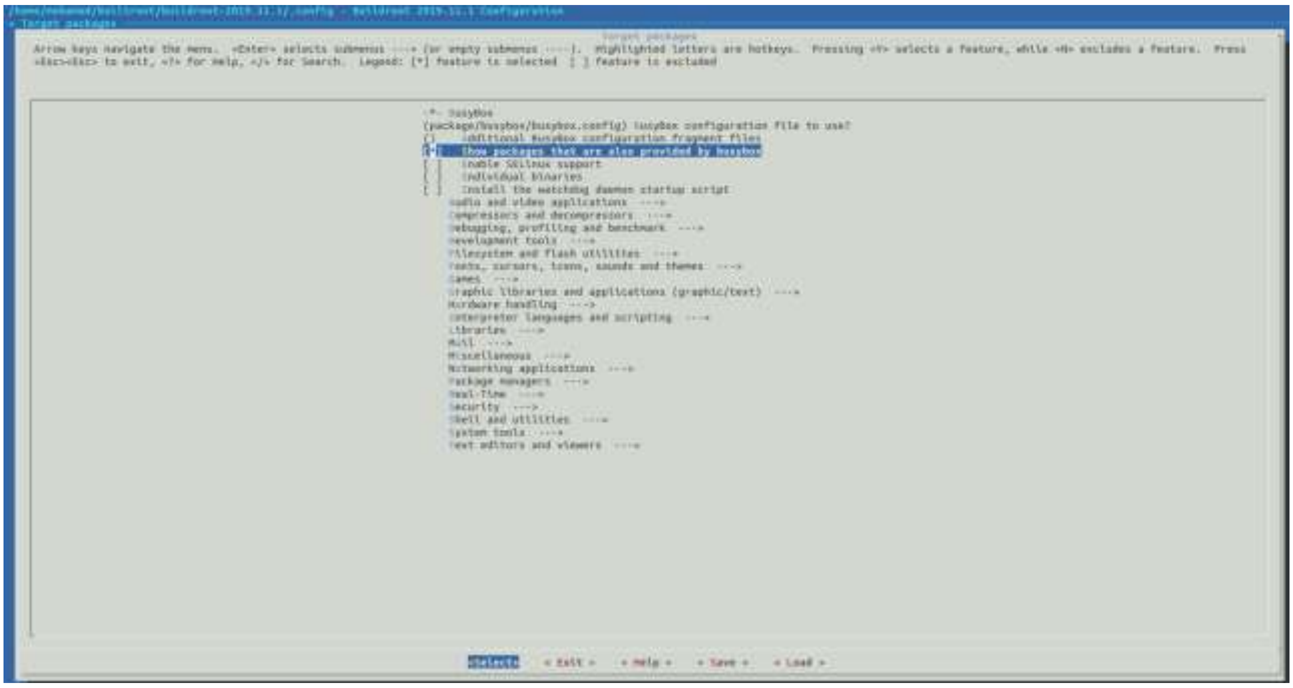
```
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
<Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

[ ] ed
[ ] joe
[ ] less
[ ] mc
[ ] mg
[*] nano
[*]   optimize for size
[ ] uemacs
[ ] vim

        <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 6 Enable nano as text editor

Adding bash language as a default shell scripting language

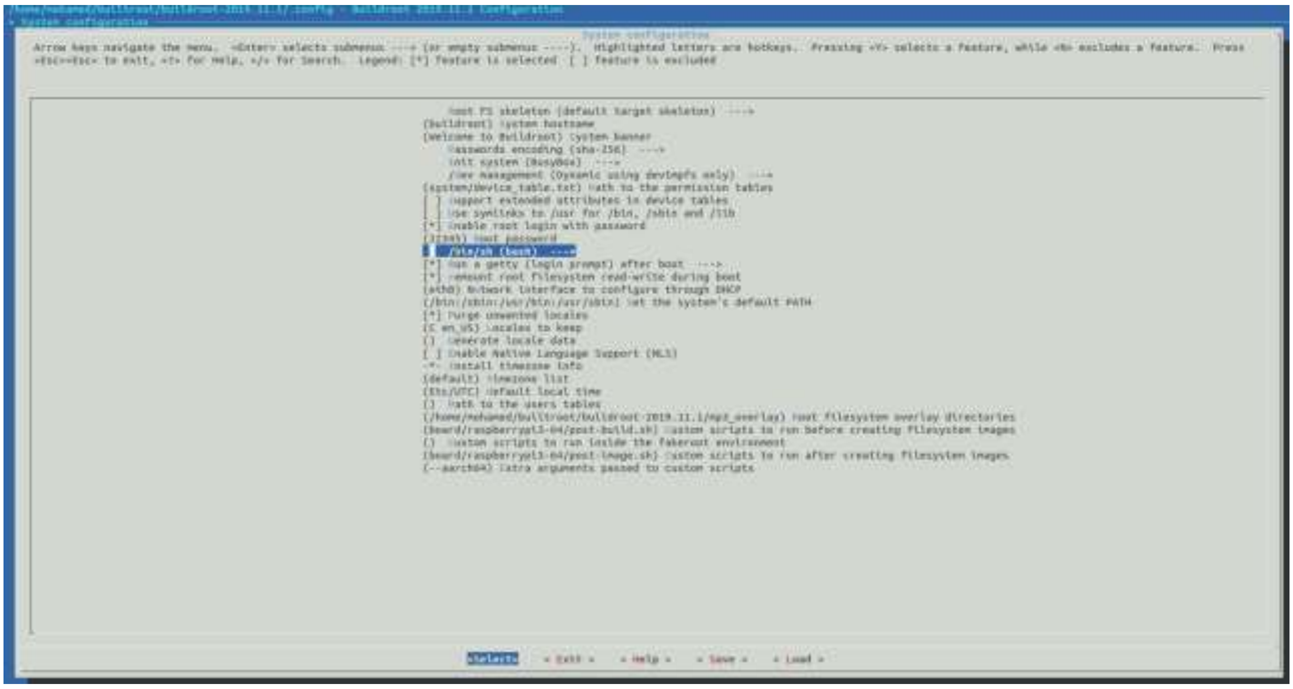
This is achieved using build root menuconfig as follows:

Target packages → select show packages that are also provided by busy box
System configuration →select bash in /*bin/*sh



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
→ Target packages
                                   Target packages
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                        -*- BusyBox
                        (package/busybox/busybox.config) BusyBox configuration file to use?
                        ()    Additional BusyBox configuration fragment files
                        [*]   Show packages that are also provided by busybox
                        [ ]   Enable SELinux support
                        [ ]   Individual binaries
                        [ ]   Install the watchdog daemon startup script
                            Audio and video applications  --->
                            Compressors and decompressors  --->
                            Debugging, profiling and benchmark  --->
                            Development tools  --->
                            Filesystem and flash utilities  --->
                            Fonts, cursors, icons, sounds and themes  --->
                            Games  --->
                            Graphic libraries and applications (graphic/text)  --->
                            Hardware handling  --->
                            Interpreter languages and scripting  --->
                            Libraries  --->
                            Mail  --->
                            Miscellaneous  --->
                            Networking applications  --->
                            Package managers  --->
                            Real-Time  --->
                            Security  --->
                            Shell and utilities  --->
                            System tools  --->
                            Text editors and viewers  --->

                    <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 7 show packages in busy box



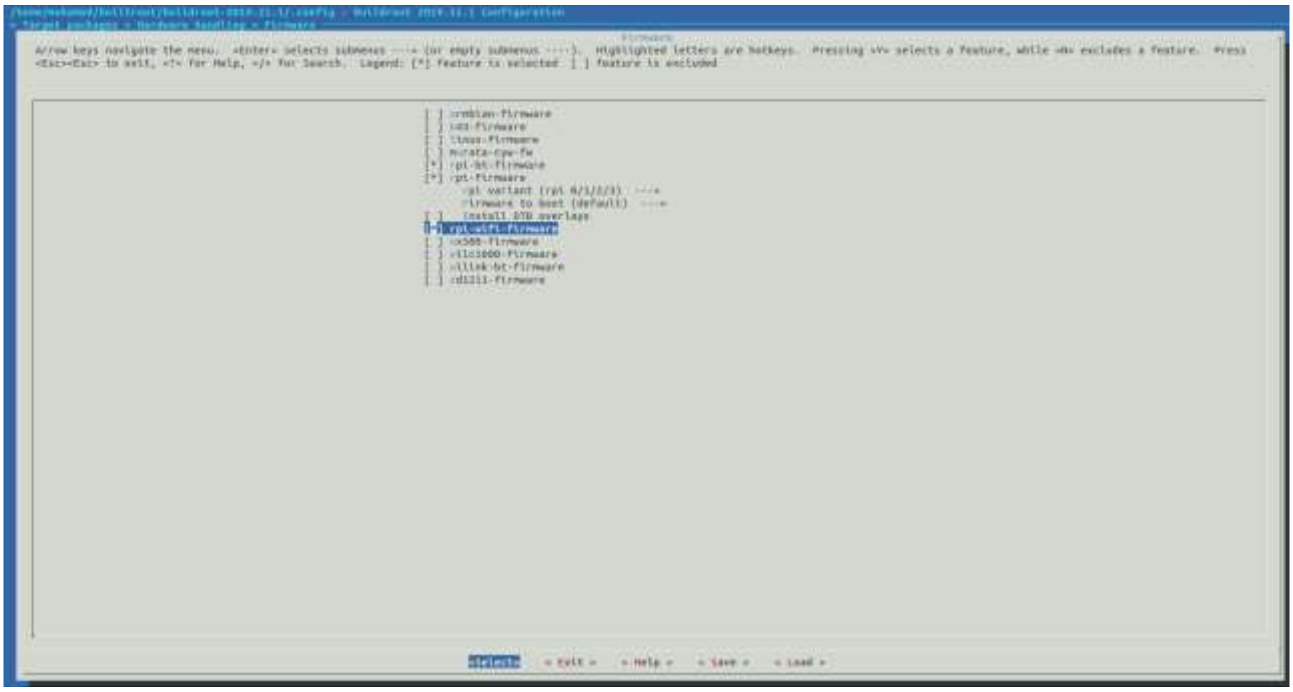
```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
→ System configuration
                                   System configuration
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                            Root FS skeleton (default target skeleton)  --->
                        (buildroot) System hostname
                        (Welcome to Buildroot) System banner
                            Passwords encoding (sha-256)  --->
                            Init system (BusyBox)  --->
                            /dev management (Dynamic using devtmpfs only)  --->
                        (system/device_table.txt) Path to the permission tables
                        [ ] Support extended attributes in device tables
                        [ ] Use symlinks to /usr for /bin, /sbin and /lib
                        [*] Enable root login with password
                        (1234) Root password
                            /bin/sh (bash)  --->
                        [*] Run a getty (login prompt) after boot  --->
                        [*] remount root filesystem read-write during boot
                        (eth0) Network interface to configure through DHCP
                        (/bin:/sbin:/usr/bin:/usr/sbin) Set the system's default PATH
                        [*] Purge unwanted locales
                        (C en_US) Locales to keep
                        ()  Generate locale data
                        [ ] Enable Native Language Support (NLS)
                        -*- Install timezone info
                        (default) Timezone list
                        (Etc/UTC) Default local time
                        ()  Path to the users tables
                        (/home/mohamed/buildroot/buildroot-2019.11.1/mp3_overlay) Root filesystem overlay directories
                        (board/raspberrypi3-64/post-build.sh) Custom scripts to run before creating filesystem images
                        ()  Custom scripts to run inside the fakeroot environment
                        (board/raspberrypi3-64/post-image.sh) Custom scripts to run after creating filesystem images
                        (--aarch64) Extra arguments passed to custom scripts

                    <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 8 Enable bash

Enabling Wifi

This is achieved using build root menuconfig as follows:

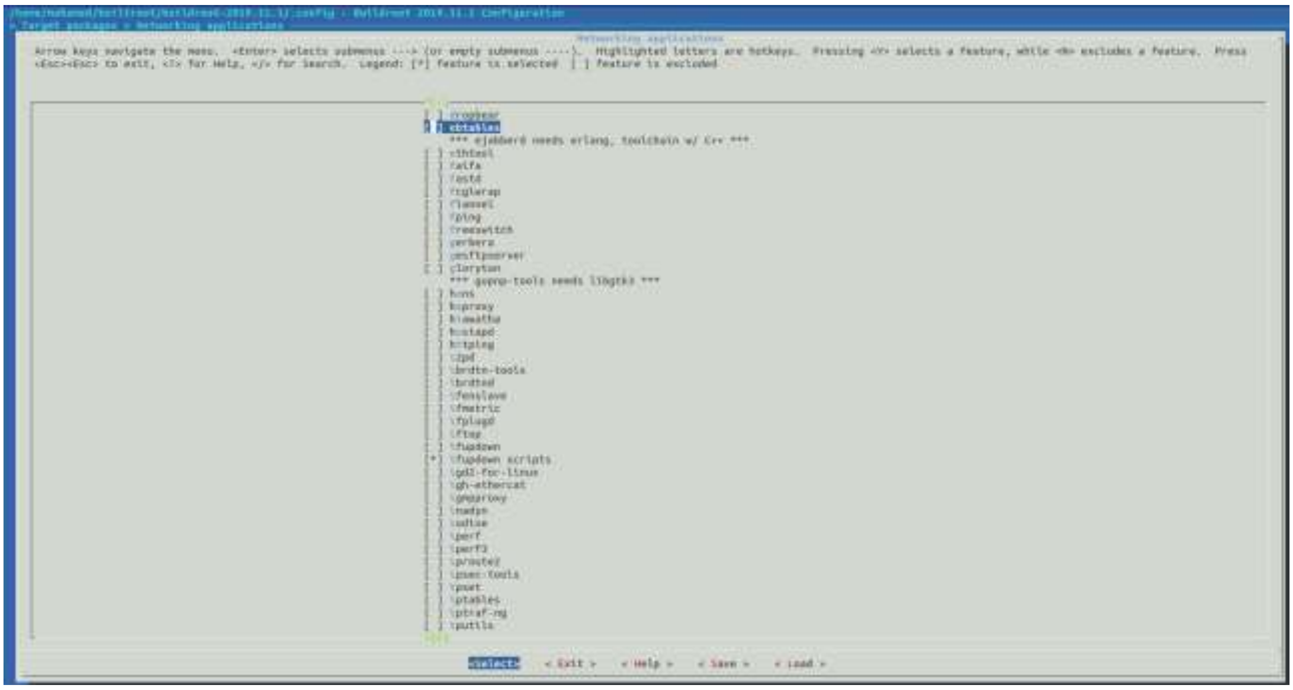
Target packages -> Hardware handling -> Firmware -> select rpi-wifi-firmware
Target packages -> Networking applications ->wpa_supplicant -> select Enable nl80211 support ,enable Autoscan



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> Target packages > Hardware handling > Firmware
                                   Firmware
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                        [ ] armbian-firmware
                        [ ] b43-firmware
                        [ ] linux-firmware
                        [ ] murata-cyw-fw
                        [*] rpi-bt-firmware
                        [*] rpi-firmware
                              rpi variant (rpi 0/1/2/3)  --->
                              Firmware to boot (default)  --->
                        [ ]   Install DTB overlays
                        [*] rpi-wifi-firmware
                        [ ] sunxi-firmware
                        [ ] ti-cc3000-firmware
                        [ ] ti-link-bt-firmware
                        [ ] zd1211-firmware

           <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 9 rpi-wifi-firmware

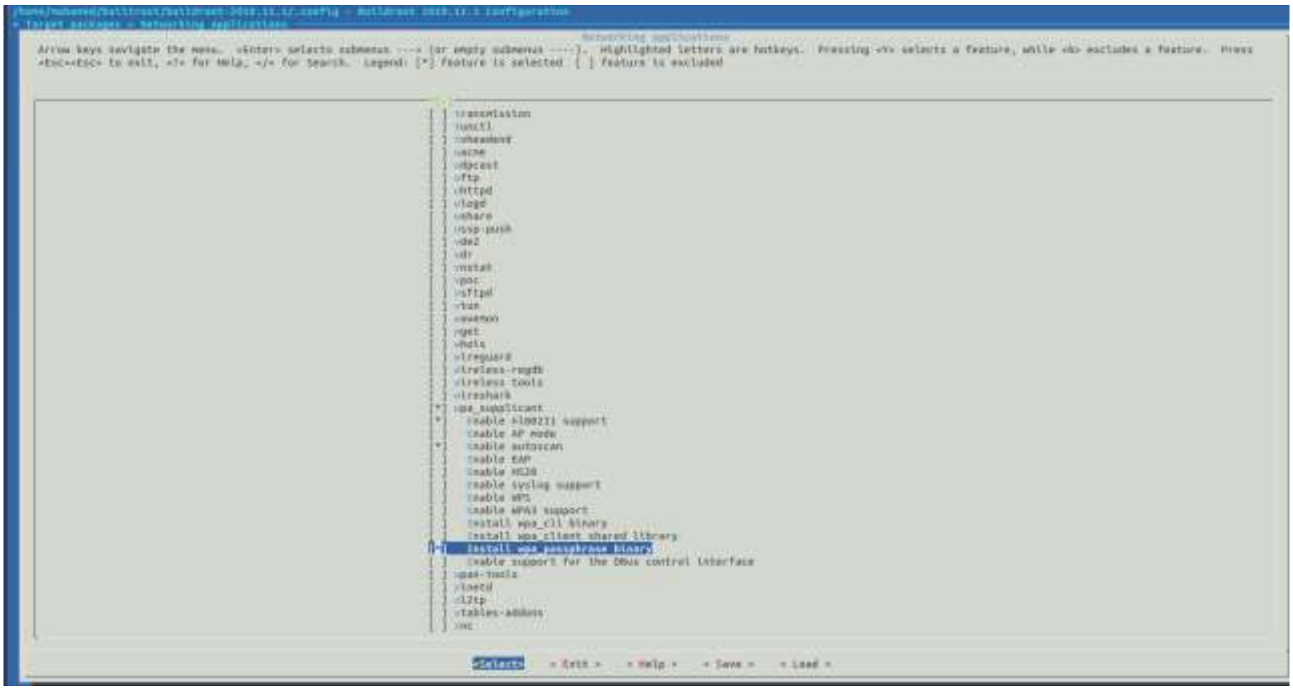
Target packages -> Networking applications ->Select ifupdown scripts and select openssh



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> Target packages > Networking applications
                           Networking applications
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                        ^(-)
                        [ ] dropbear
                        [ ] ebtables
                            *** ejabberd needs erlang, toolchain w/ C++ ***
                        [ ] ethtool
                        [ ] faifa
                        [ ] fastd
                        [ ] fcgiwrap
                        [ ] flannel
                        [ ] fping
                        [ ] freeswitch
                        [ ] gerbera
                        [ ] gesftpserver
                        [ ] glorytun
                            *** gupnp-tools needs libgtk3 ***
                        [ ] hans
                        [ ] haproxy
                        [ ] hiawatha
                        [ ] hostapd
                        [ ] httping
                        [ ] i2pd
                        [ ] ibrdtn-tools
                        [ ] ibrdtnd
                        [ ] ifenslave
                        [ ] ifmetric
                        [ ] ifplugd
                        [ ] iftop
                        [ ] ifupdown
                        [*] ifupdown scripts
                        [ ] igd2-for-linux
                        [ ] igh-ethercat
                        [ ] igmpproxy
                        [ ] inadyn
                        [ ] iodine
                        [ ] iperf
                        [ ] iperf3
                        [ ] iproute2
                        [ ] ipsec-tools
                        [ ] ipset
                        [ ] iptables
                        [ ] iptraf-ng
                        [ ] iputils
                        v(+)

           <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 10 ifupdown-script

Target packages -> Networking applications ->Install wpa_passphrase binary



```
/home/mohamed/buildroot/buildroot-2019.11.1/.config - Buildroot 2019.11.1 Configuration
> Target packages > Networking applications
                                  Networking applications
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                                  [ ] transmission
                                  [ ] tunctl
                                  [ ] tvheadend
                                  [ ] uacme
                                  [ ] udpcast
                                  [ ] uftp
                                  [ ] uhttpd
                                  [ ] ulogd
                                  [ ] ushare
                                  [ ] ussp-push
                                  [ ] vde2
                                  [ ] vdr
                                  [ ] vnstat
                                  [ ] vpnc
                                  [ ] vsftpd
                                  [ ] vtun
                                  [ ] wavemon
                                  [ ] wget
                                  [ ] whois
                                  [ ] wireguard
                                  [ ] wireless-regdb
                                  [ ] wireless tools
                                  [ ] wireshark
                                  [*] wpa_supplicant
                                  [*]   Enable nl80211 support
                                  [ ]   Enable AP mode
                                  [*]   Enable autoscan
                                  [ ]   Enable EAP
                                  [ ]   Enable HS20
                                  [ ]   Enable syslog support
                                  [ ]   Enable WPS
                                  [ ]   Enable WPA3 support
                                  [ ]   Install wpa_cli binary
                                  [ ]   Install wpa_client shared library
                                  [*]   Install wpa_passphrase binary
                                  [ ]   Enable support for the DBus control interface
                                  [ ] wpan-tools
                                  [ ] xinetd
                                  [ ] xl2tp
                                  [ ] xtables-addons
                                  [ ] znc

                   <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 11 install wpa_passphrase binary

Post build scripts and overlay configurations

Configuring a static IP address to Raspberry-pi

This is achieved using overlay folder by modifying the interfaces file in the following path
"/etc/network/interfaces"

A static IP address for Ethernet connection is configured to be able to access raspberry pi through SSH over Ethernet.

Accessing raspberry pi through SSH

This is achieved using overlay folder by modifying the sshd_config file in the following path
"/etc/ssh/sshd_config"
This step is necessary to permit the root login because it is disabled by default.

Cross compilation of Hello.c to run on the Raspberry-pi

This is achieved using the following steps:

1. Export PATH=<Buildroot_path>/output/host/bin: $PATH
2. aarch64 to build the application
3. aarch64-linux-gcc Hello.c -o Hello.o

Enabling the sound driver on raspberry pi

This is achieved using the following steps:

1. Appending the following line "dtparam=audio=on" in config.txt file in the following path: ''buildroot/package/rpi-firmware/config.txt''

2. By using the overlay folder to run script "S55audio" in the end of the boot process to enable sound driver, The script will be in the following path:"etc/init.d"

Enabling the Bluetooth driver on raspberry pi

This is achieved using the overlay folder to run script "S55audio" in the end of the boot process to enable Bluetooth driver, The script will be in the following path:"etc/init.d"

Enabling the Wifi driver op raspberry pi

This is achieved using overlay folder by modifying the interfaces file in the following path
"/etc/network/interfaces"
This step is necessary to enable the wifi interface, give this interface a static IP,

Configure the Wifi network

This is achieved using overlay folder by modifying the interfaces file in the following path "/etc/wpa_supplicant/wpa_supplicant.conf"

Enabling the final application on the raspberry pi at start up

This can be achieved by more than one methods:

1. By using the overlay folder to run script "S55audio" in the end of the boot process to run application scripts as daemons, The script will be in the following path:" etc/init.d"

2. By using the overlay folder to run script "runApplication.sh" at the beginning of accessing the terminal to run application scripts as background processes, The script will be in the following path:" etc/profile.d"