User-space Applications In AJIT-LINUX-IMAGE

Saurabh Bansode

22 October 2020

1 Prerequisite

1. Create a directory for the application inside ../ajit-toolchain/src/buildroot-20xx.xx/packages/<pkg_name>

2. Write a Config.in file for the package describing what it does:

   Name the application and declare dependencies, if any.

```
config BR2_PACKAGE_AJITTEST
        bool "ajittest"
        help
          This is a helloworld program for ajit processor

          https://github.com/saurabhb17/ajit
```

3. Write a <pkg_name.mk >file which describes:
   Where to get the source code from (local copy/git/svn etc.)
   How to build the source files
   Name of the executable to be created
   Where to install it
   Note: in the <pkg_name>_SITE parameter, specify the path where the source code is kept, considering your PWD is ajit-toolchain/src/buildroot-2014.08/

```
AJITTEST_VERSION = 2.0
AJITTEST_SITE = ./package/ajittest/src
AJITTEST_SITE_METHOD = local

define AJITTEST_BUILD_CMDS
        $(MAKE) CC="$(TARGET_CC)" LD="$(TARGET_LD)" -C $(@D)
endef

define AJITTEST_INSTALL_TARGET_CMDS
        $(INSTALL) -D -m 0755 $(@D)/ajittest $(TARGET_DIR)/usr/bin
```

```
endef

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

4. Add this package name inside the buildroot/packages/Config.in under Miscellaneous/appropriate section as shown in below lines:

```
menu "Miscellaneous"
        source "package/ajittest/Config.in"
        source "package/asecondtest/Config.in"
        source "package/aespipe/Config.in"
        source "package/bc/Config.in"
        source "package/collectd/Config.in"
        source "package/empty/Config.in"
        source "package/googlefontdirectory/Config.in"
        source "package/haveged/Config.in"
        source "package/mcrypt/Config.in"
        source "package/mobile-broadband-provider-info/Config.in"
        source "package/shared-mime-info/Config.in"
        source "package/snowball-init/Config.in"
        source "package/sound-theme-borealis/Config.in"
        source "package/sound-theme-freedesktop/Config.in"
endmenu
```

   The text `source "package/ajittest/Config.in"` will add the "ajittest" application to buildroot's list.

5. Coming to the source files:
   Keep your source files in the directory mentioned in earlier steps.

```
#include<stdio.h>
int a =8;
int main()
{
        printf("Hi world:%d\r\n",a);
        a++;
        printf("hello world:%d\r\n",a);
        return 0;
}
```

6. Write a Makefile for your source code, compile it locally with the ajit toolchain to see if the code compiles and you get an executable.

```
CC=sparc-buildroot-linux-uclibc-gcc
CFLAGS=-I.

all: *.c
```

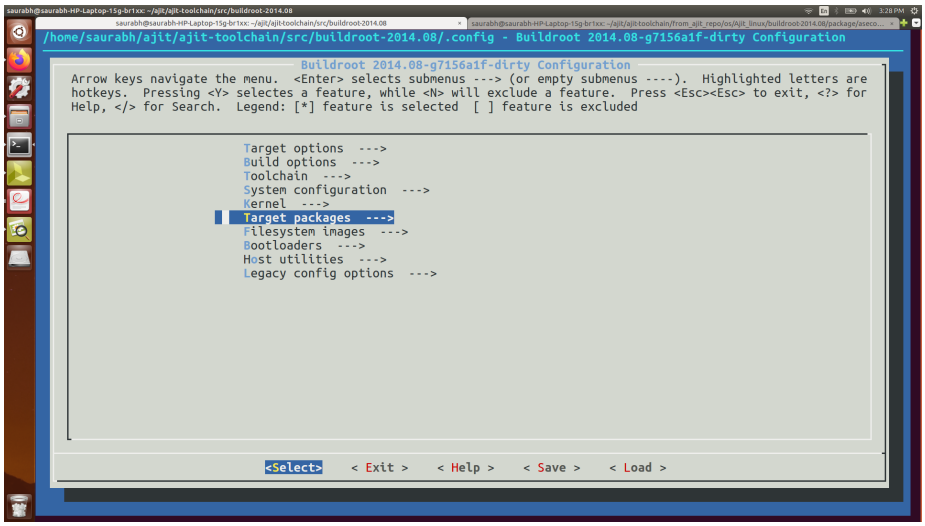

Figure 1: Menu prompted by `make menuconfig`: Select Target Applications

```
        $(CC) ajittest.c -o ajittest

clean:
        rm -f a.out ajittest
        rm -f *.o
```

2 Procedure: How to add the packages

1. Make sure that the contents of the top level directory i.e `ajit-toochain/src/buildrooot-2014.xx/.con` file reflects the changes mentioned in Prerequisites.

2. Do a `make menuconfig` in the ../src/buildrooot-xx/ directory, under `target packages`, select the application you wanted from the category under which you have sourced your file in the `../buildrooot-2014.xx/packages/Config.in` Newly added applications will be highlighted with a `NEW` text next to its name.
   For this **ajittest** application I have placed it under Miscellaneous section.

3. An acknowledgement of completion of above step can be checked by examining the ../buildroot-2014.xx/**.config** file.

4. Run ./src/clean.sh followed by ./src/setup.sh at the top directory.

5. Run `make menuconfig` inside the /os/Ajit_linux/buildroot-20xx.xx/ once.
   Verify .config file
   run `make`

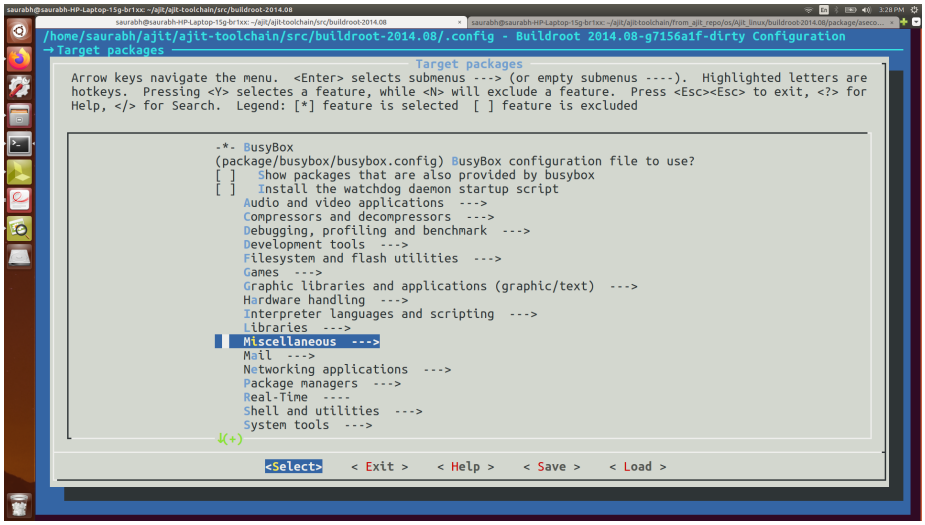

Figure 2: Select Miscellaneous as the subcategory of the application

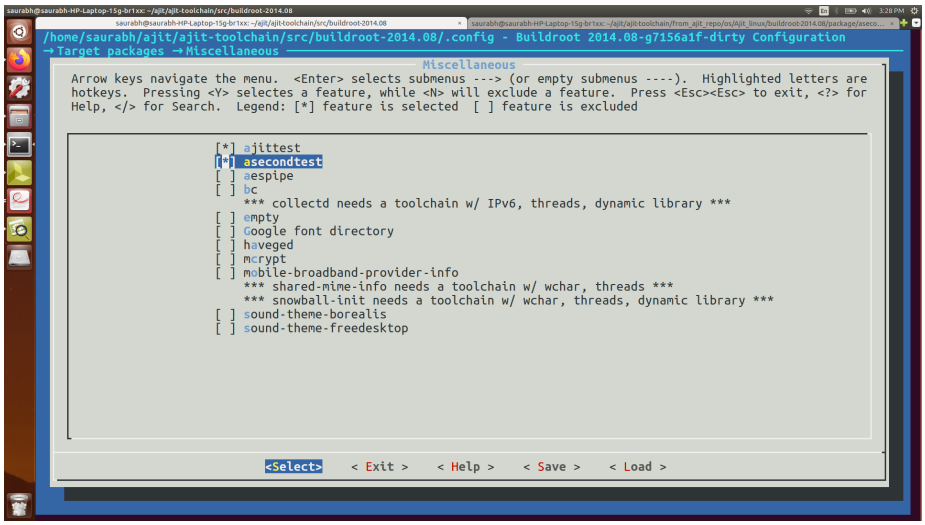

Figure 3: Select application(s)

6. Source the ajitbashrc from the ajit-release in the same terminal and come back to the Ajit_linux dir .
7. Run the script `GeneratememMapForAjit.sh` followed by `vmLinuzToMemmapAtf0004000.sh`
8. Now copy the boot_loader_plus_kernel.mmap image from the Generated_memory_map.f0004000/Output.f0004000/ directory.
9. Place this file in Ajit_linux/linux_boot/boot_loader_plus_kernel directory
10. cd linux_boot
11. do chmod +x compile.sh, followed by ./compile.sh
12. The final output will be in ../linux_boot/flash_contents directory.