

Install L3.0.35_1.1.0_121218_source.tar.gz into 32bit-ubuntu 10.10

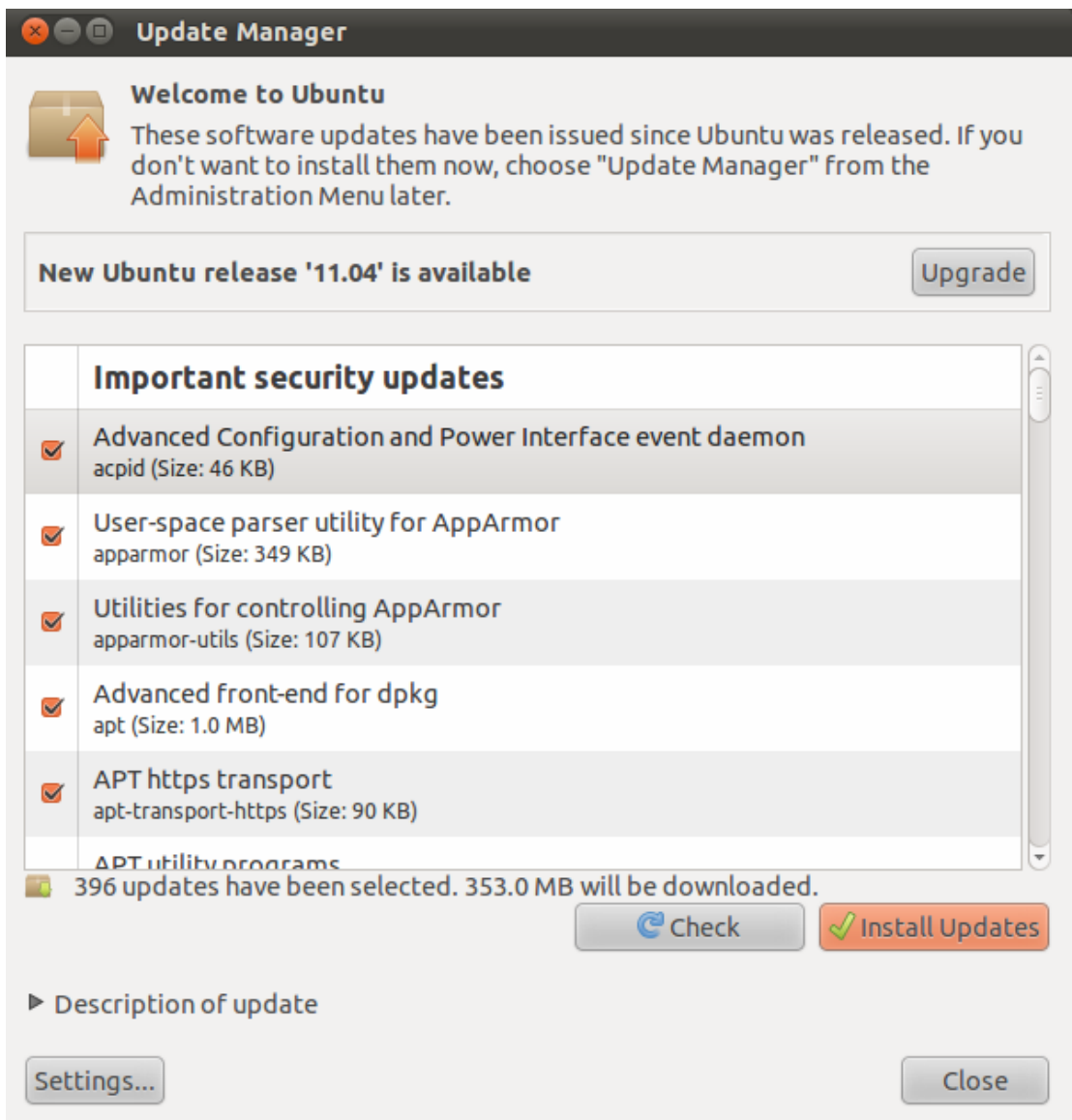
<Note>32bit ubuntu OS is a new system-- **ubuntu-10.10-desktop-i386.iso**

1.Add root password to system

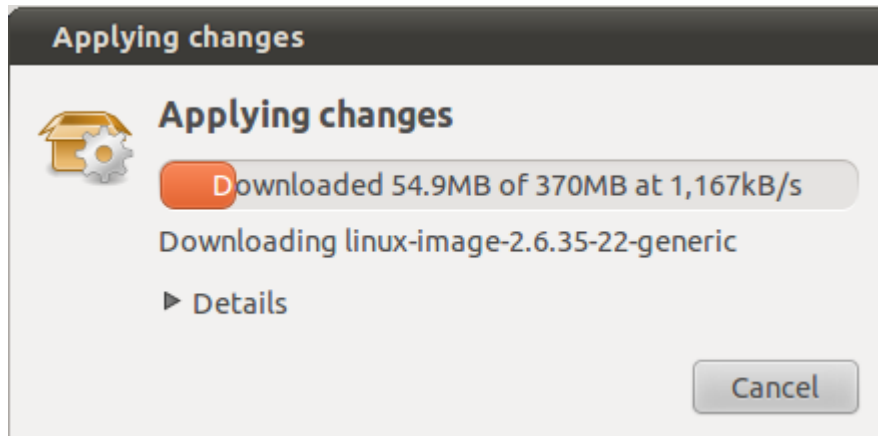
```
weidong@ubuntu:~$ sudo passwd root
[sudo] password for weidong:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
weidong@ubuntu:~$
```

2. Update ubuntu packages

(1)Using Update Manager to update packages that system checks



Press "Install Updates" and input root password, system will begin to update packages



(2) Update source list

```
#sudo cp /etc/apt/sources.list /etc/apt/sources.list.bak (backup original file)
```

```
#sudo gedit /etc/apt/sources.list
```

Delete all item in this file, then copy the following server addresses to it :

```
deb http://mirror.lupaworld.com/ubuntu/ maverick main universe restricted multiverse
deb-src http://mirror.lupaworld.com/ubuntu/ maverick main universe restricted multiverse
deb http://mirror.lupaworld.com/ubuntu/ maverick-security universe main multiverse restricted
deb-src http://mirror.lupaworld.com/ubuntu/ maverick-security universe main multiverse restricted
deb http://mirror.lupaworld.com/ubuntu/ maverick-updates universe main multiverse restricted
deb-src http://mirror.lupaworld.com/ubuntu/ maverick-updates universe main multiverse
restricted
deb http://mirror.lupaworld.com/ubuntu/ maverick-backports universe main multiverse restricted
deb-src http://mirror.lupaworld.com/ubuntu/ maverick-backports universe main multiverse
restricted
deb-src http://mirror.lupaworld.com/ubuntu/ maverick-updates universe main multiverse restricted
```

Save it and Exit .

Run command to update source list:

```
#sudo apt-get update
```

3. Install Samba Service

```
#sudo apt-get install samba
```

```
#sudo apt-get install smbfs
```

```
#chmod 777 /home/weidong
```

<Note> /home/weidong is a sharing directory with windows

```
#sudo gedit /etc/samba/smb.conf
```

Add the following lines at the end of this file:

```
[ubuntu10_Share]
comment = Shared Folder with username and password
path = /home/weidong
public = yes
writable = yes
available = yes
browseable = yes
```

Then modify security=user to security=share(No 102 line) , save the file and exit. Like:

in the samba-doc package for details.

```
security = share
```

You may wish to use password encryption. See the section on

using following command to restart samba service .

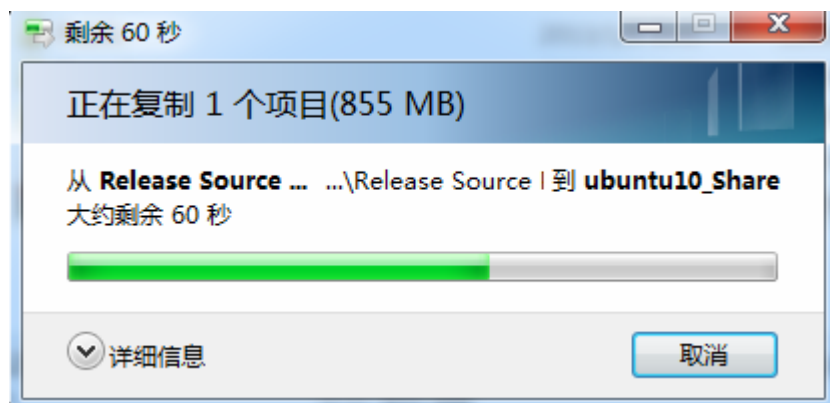
```
#sudo /etc/init.d/smbd restart
```

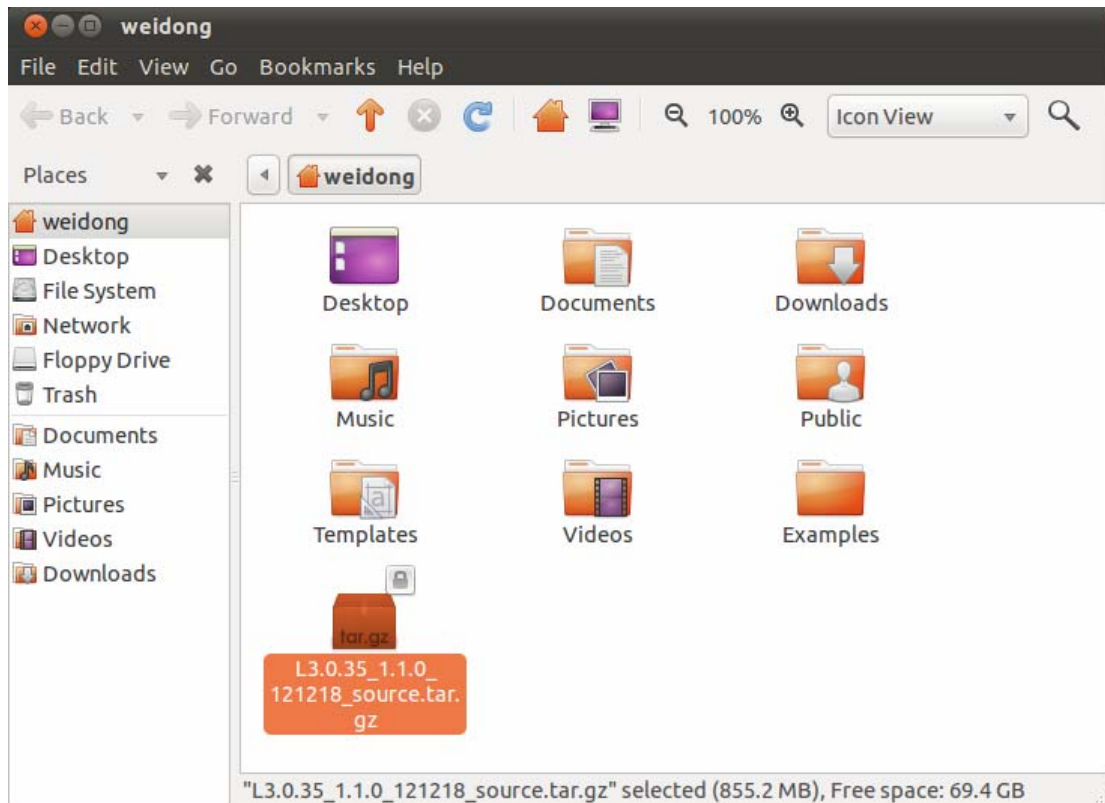
Access sharing directory that you have been set

Open 'run' on windows 7 , input `\\192.168.1.120` , enter, you will find a sharing directory named ubuntu10_share is displayed at right side. It indicates we successfully configured the samba server.

You can copy a file to ubuntu host to test your samba's configuration.

Copy L3.0.35_1.1.0_121218_source.tar.gz from windows to ubuntu host :





3. Install LTIB

(1)uncompress tar ball

```
#cd ~/
```

```
#tar -zxvf L3.0.35_1.1.0_121218_source.tar.gz
```

Then all files are uncompressed into path ~/L3.0.35_1.1.0_121218_source

```
weidong@ubuntu:~$ cd L3.0.35_1.1.0_121218_source/
weidong@ubuntu:~/L3.0.35_1.1.0_121218_source$ ls
EULA  install  ltib.tar.gz  package_manifest.txt  pkgs  redboot_201003.zip  tftp.zip
weidong@ubuntu:~/L3.0.35_1.1.0_121218_source$
```

```
#cd L3.0.35_1.1.0_121218_source
```

(2)Install ltib

```
#./install
```

You are about to install the LTIB (GNU/Linux Target Image Builder)

Before installing LTIB, you must read and accept the EULA (End User License Agreement) which will be presented next.

Do you want to continue ? Y/n

Y

.....
.....

If the Licensed Software includes software and hardware developed by Cambridge Silicon Radio, Inc. ("CSR"), you must separately obtain rights beyond evaluation and demonstration for the CSR software from CSR.

LAOPT 27
Update 9/2011

I have read and accept the EULA (yes/no):

yes

The LTIB files are extracted from a tar file which includes the prefix ltib. After installation you will find LTIB in:

/home/weidong/L3.0.35_1.1.0_121218_source/ltib

Where do you want to install LTIB ? (/home/weidong/L3.0.35_1.1.0_121218_source)

/home/weidong

Then ltib will be installed /home/weidong/ltib directory

Installation complete, your ltib installation has been placed in /home/weidong/ltib, to complete the installation:

```
cd /home/weidong/ltib
```

```
./ltib
```

```
#cd /home/weidong/ltib
```

```
#!/ltib
```

****errors occur A**

```
weidong@ubuntu:~/ltib$ ./ltib
```

```
sh: g++: not found
```

ltib cannot be run because one or more of the host packages needed to run it are either missing or out of date or not in ltib's standard path. Please install/upgrade these packages on your host. If you have your own utilities in non-standard paths, please add an entry into the .ltibr file for example:

```
%path_std
```

```
/usr/local/bin:/usr/bin:/bin:/usr/bin/X11:/usr/X11R6/bin:/my/own/exes
```

Package	Minimum ver	Installed info
libstdc++	0	not installed
gcc-c++	2.96	not installed
zlib-devel	0	not installed
rpm	0	not installed
rpm-build	0	not installed
ncurses-devel	0	not installed
m4	0	not installed

bison	0	not installed
patch	0	not installed

Died at ./ltib line 1409.

traceback:

```
main::host_checks:1409
main:554
```

Started: Tue Jan 8 02:03:45 2013
Ended: Tue Jan 8 02:03:46 2013
Elapsed: 1 seconds

Build Failed

Exiting on error or interrupt

Resolve : Let us install packages mentioned above:

```
#sudo apt-get install gettext libgtk2.0-dev rpm bison m4 libfreetype6-dev
#sudo apt-get install libdbus-glib-1-dev liborbit2-dev intltool
#sudo apt-get install ccache ncurses-dev zlib1g zlib1g-dev gcc g++ libtool
#sudo apt-get install uuid-dev liblzo2-dev
#sudo apt-get install tcl dpkg
Continue to run ./ltib
#./ltib
```

****errors occur B**

I ran the command: sudo -S -l which returned:

```
[sudo] password for weidong: Sorry, try again.
[sudo] password for weidong: Sorry, try again.
[sudo] password for weidong: Sorry, try again.
sudo: 3 incorrect password attempts
```

This means you don't have sudo permission to execute rpm commands as root without a password. This is needed for this build script to operate correctly.

To configure this, as root using the command "/usr/sbin/visudo", and add the following line in the User privilege section:

```
weidong ALL = NOPASSWD: /usr/bin/rpm, /opt/freescale/ltib/usr/bin/rpm
```

traceback:

```
main::check_sudo_setup:2467
```

main:host_checks:1413

main:554

Started: Tue Jan 8 02:14:20 2013

Ended: Tue Jan 8 02:14:26 2013

Elapsed: 6 seconds

Build Failed

Exiting on error or interrupt

weidong@ubuntu:~/ltib\$

Resolve :

Copy " weidong ALL = NOPASSWD: /usr/bin/rpm, /opt/freescale/ltib/usr/bin/rpm" to /usr/sbin/visudo

#sudo /usr/sbin/visudo

The following is contents of this file:

/etc/sudoers

#

This file *MUST* be edited with the 'visudo' command as root.

#

See the man page for details on how to write a sudoers file.

#

Defaults env_reset

Host alias specification

User alias specification

Cmnd alias specification

User privilege specification

root ALL=(ALL) ALL

weidong ALL = NOPASSWD: /usr/bin/rpm, /opt/freescale/ltib/usr/bin/rpm

Allow members of group sudo to execute any command

(Note that later entries override this, so you might need to move

it further down)

%sudo ALL=(ALL) ALL

#

#includedir /etc/sudoers.d

Members of the admin group may gain root privileges

<i>^G Get Help</i>	<i>^O WriteOut</i>	<i>^R Read File</i>	<i>^Y Prev Page</i>
<i>^K Cut Text</i>	<i>^C Cur Pos</i>		
<i>^X Exit</i>	<i>^J Justify</i>	<i>^W Where Is</i>	<i>^V Next Page</i>
<i>^U UnCut Text</i>	<i>^T To Spell</i>		

Save and Exit , Continue to run ltib

```
#!/tlib
```

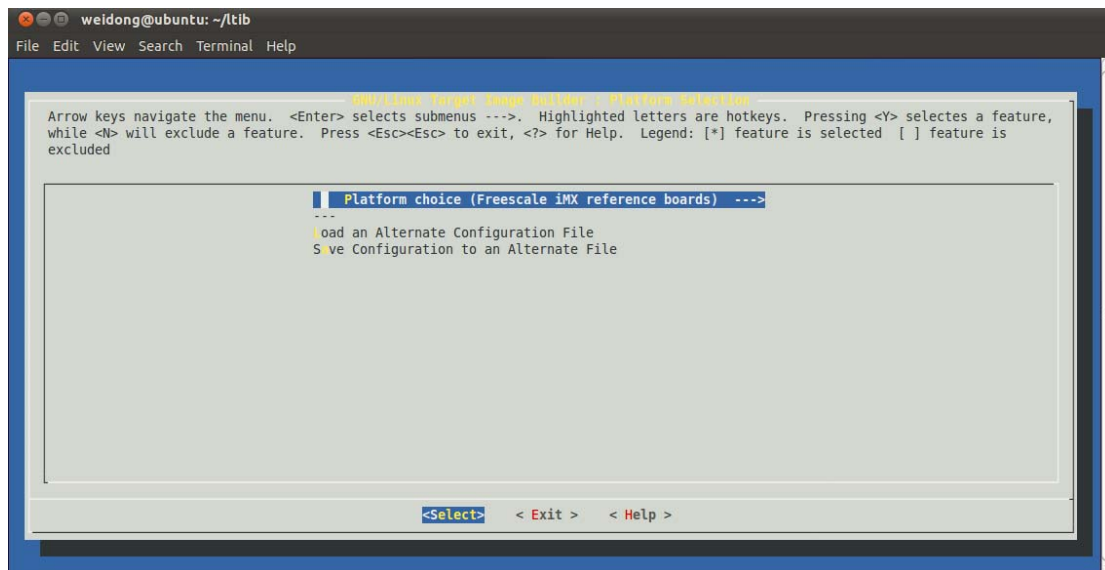
```
weidong@ubuntu:~/ltib$ ./ltib
```

Installing host support packages.

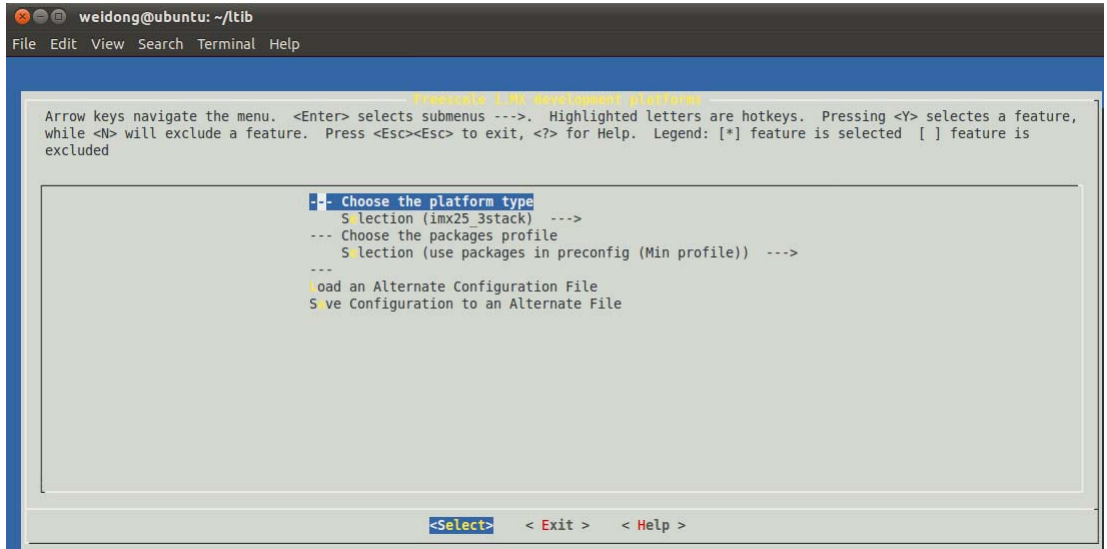
This only needs to be done once per host, but may take up to an hour to complete ...

If an error occurs, a log file with the full output may be found in: /home/weidong/ltib/host_config.log

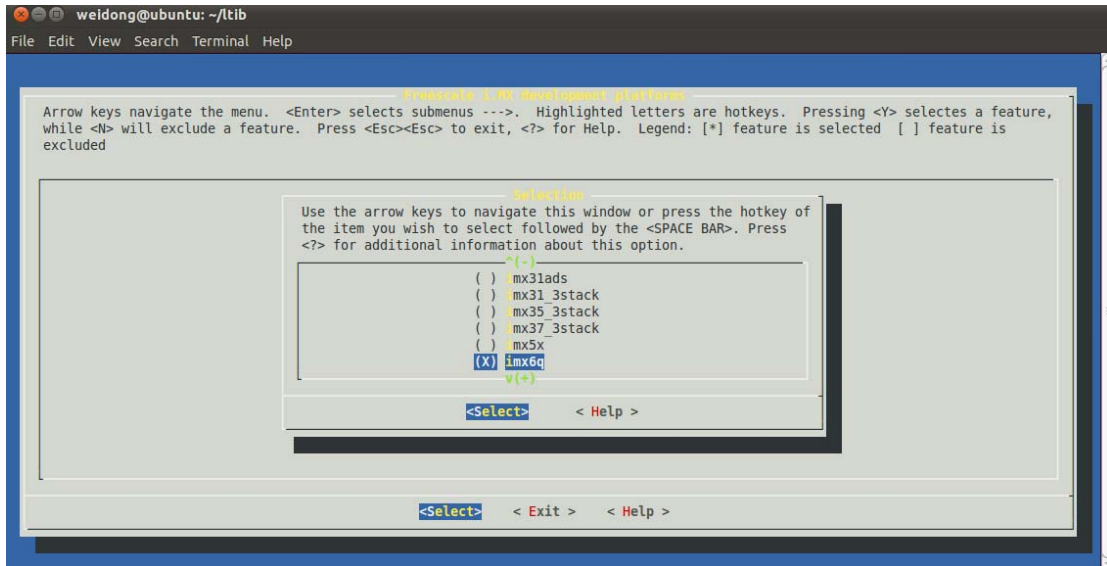
--No errors occurred , the following menu will appear :



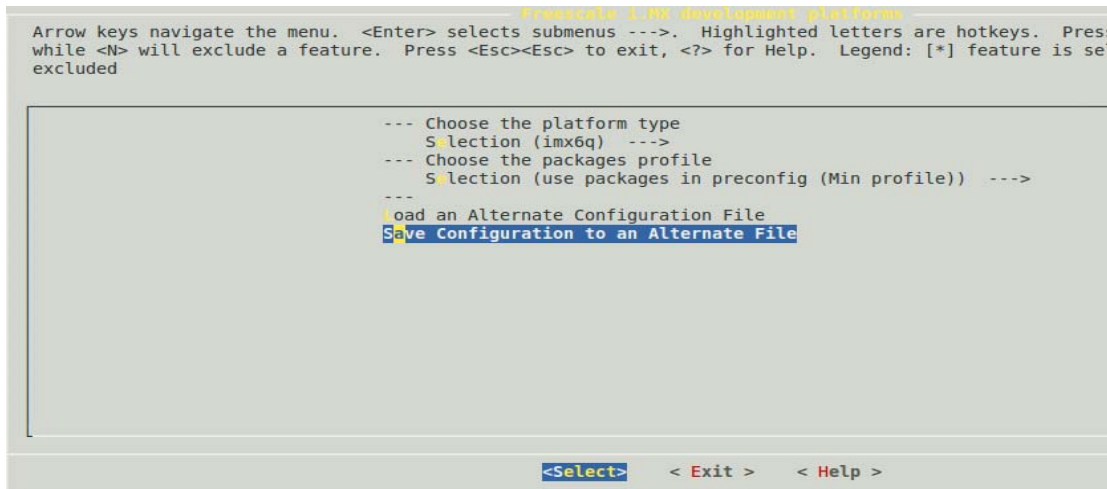
It indicates LTIB was installed successfully ! Select Exit and Save config , go into next step :



-Selection (imx25_3stack)---->
 ---(x)imx6q



Come back and continue :
 Save and Exit



Go to next step :

```
weldong@ubuntu: ~/l1ib
File Edit View Search Terminal Help

Freescale iMX6x Based Boards
Arrow keys navigate the menu. <Enter> selects submenus ---. Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> will exclude a feature. Press <Esc><Esc> to exit, <?> for Help. Legend: [*] feature is selected [ ] feature is excluded

[*] mx6q Platform
--- L1IB settings
  System features --->
--- Choose the target C library type
  Target C library type (glibc) --->
  C library package (from toolchain only) --->
  Toolchain component options --->
--- Toolchain selection.
  Toolchain (ARM, gcc-4.6.2, multilib, neon optimized, gnuabi/eglibc2.13) --->
(-02 -march=armv7-a -mcpu=vfpv3 -mfloat-abi=softfp) Enter any CFLAGS for gcc/g++ (NEW)
--- Choose your bootloader for U-Boot
  Bootloader (u-boot) --->
--- Choose your board for u-boot
  Board (mx6q_arm2) --->
--- Choose your Kernel
  Kernel (Linux 3.0.35-imx) --->
[ ] Always rebuild the kernel
v(+)

<Select> < Exit > < Help >
```

bootloader (u-boot)

board (mx6q_arm2)---->

----mx6q_sabresd

```
Freescale iMX6x Based Boards
Arrow keys navigate the menu. <Enter> selects submenus ---. Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> will exclude a feature. Press <Esc><Esc> to exit, <?> for Help. Legend: [*] feature is selected [ ] feature is excluded

^(-)
  C libr
  Toolch
  Toolch
(-02 -march=armv7-a -mcpu=vfpv3 -mfloat-abi=softfp) Enter any CFLAGS for gcc/g++ (NEW)
--- Choose your bootloader for U-Boot
  Bootloader (u-boot) --->
--- Choose your board for u-boot
  Board (mx6q_arm2) --->
  Board
  ( ) mx6q_sabrelite
  ( ) mx6q_sabreauto
  ( ) mx6q_sabreauto_nand
  ( ) mx6q_sabreauto_weimnor
  (X) mx6q_sabreauto_spi-nor
  v(+)
  <Select> < Help >

[ ] Always rebuild the kernel
[ ] Produce cscope index
--- Include kernel headers
[ ] Configure the kernel
--- Leave the sources after building
[ ] Build mfg firmware (NEW)
--- Package selection
  Package list --->
--- Target System Configuration
  Options --->
v(+)

<Select> < Exit > < Help >
```

Come back and Continue:

Kernel (linux 3.0.35-imx)

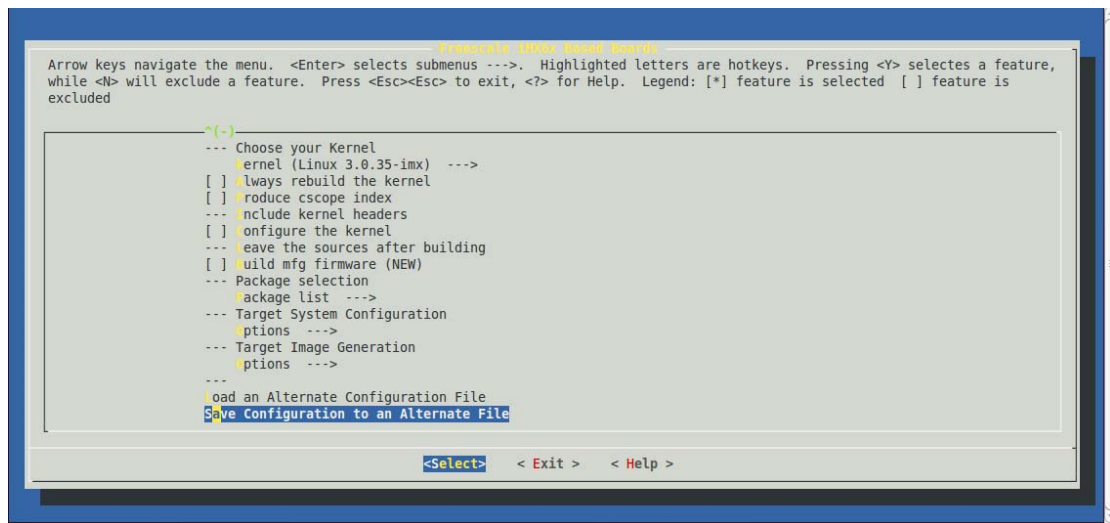
```
Freescale iMX6x Based Boards
Arrow keys navigate the menu. <Enter> selects submenus ---. Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> will exclude a feature. Press <Esc><Esc> to exit, <?> for Help. Legend: [*] feature is selected [ ] feature is excluded

^(-)
(-02 -march=armv7-a -mcpu=vfpv3 -mfloat-abi=softfp) Enter any CFLAGS for gcc/g++ (NEW)
--- Choose your bootloader for U-Boot
  Bootloader (u-boot) --->
--- Choose your board for u-boot
  Board (mx6q_sabresd) --->
--- Choose your Kernel
  Kernel (Linux 3.0.35-imx) --->
[ ] Always rebuild the kernel
[ ] Produce cscope index
--- Include kernel headers
[ ] Configure the kernel
--- Leave the sources after building
[ ] Build mfg firmware (NEW)
--- Package selection
  Package list --->
--- Target System Configuration
  Options --->
v(+)

<Select> < Exit > < Help >
```

As for other Items ,Such as Package list, customer can select them according to his requirments.

Save and Exit configuration , Then ltib will begin to compile .



*** End of configuration.

*** Check the top-level Makefile for additional configuration options.

```
+ cd -
```

```
/home/weidong/ltib
```

```
+ [ -n config/platform/imx/imx6q.cf ]
```

```
+ cp config/platform/imx/imx6q.cf config/platform/imx/.config
```

```
+ chmod +w config/platform/imx/.config
```

```
+ [ ! -f config/platform/imx/.config -a -f config/platform/imx/defconfig ]
```

```
+ [ -n ]
```

```
+ cd config/platform/imx
```

```
+ mconf main.lkc .config
```

```
../../userspace/packages.lkc:1709:warning: enable is only allowed with boolean and tristate symbols
```

```
.config:524: trying to assign nonexistent symbol CAP_FSL_INT
```

*** End of configuration.

*** Check the top-level Makefile for additional configuration op

,