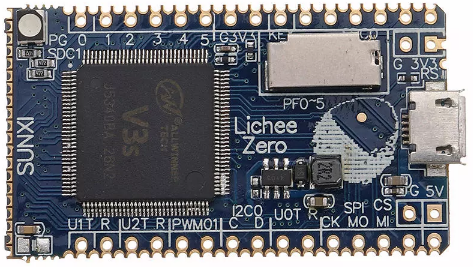
**How to compile Linux with Qt5 option using Yocto for LICHEE PI ZERO**

* Post author:[Michał Wołowik](https://www.emsyslabs.com/author/mwolo/)
* Post published:2020-04-23
* Post category:[Blog](https://www.emsyslabs.com/category/blog/) / [Linux](https://www.emsyslabs.com/category/linux/)
* Post comments:[0 Comments](https://www.emsyslabs.com/how-to-compile-linux-with-qt5-option-using-yocto-for-lichee-pi-zero/#respond)

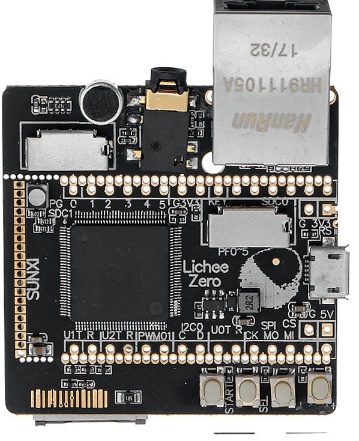
**Instruction how to build an image for Lichee Pi Zero and Lichee Pi Zero Dock in Yocto**

**Products:**

**Lichee Pi Zero Version**

[](https://github.com/voloviq/meta-licheepizero/blob/zeus/Lichee_Pi_Zero.png)

**Lichee Pi Zero Dock Version**

[](https://github.com/voloviq/meta-licheepizero/blob/zeus/Lichee_Pi_Zero_Dock.jpg)

**General Note:**

Assumed that Linux Ubuntu is installed

**List of tested elements**

WiFi  
Ethernet  
Lcd  
Touchscreen  
Led  
Backlight for Lcd

**List of not tested elements**

Bluetooth – appears during system boot up  
Microphone  
Headphone

**How to build images**

1. First, make sure to following packages are installed in the system

***sudo apt-get install gawk wget diffstat unzip texinfo gcc-multilib build-essential chrpath socat libsdl1.2-dev xterm*** ***libgmp3-dev*** ***libmpc-dev***

**Note:** More information can be found on theYocto reference manual.

2. Download necessary Yocto packaged listed below. Be sure to be in the root of the home folder.

***mkdir yocto***

***cd yocto***

***mkdir build***

***git clone git://git.yoctoproject.org/poky --depth 1 -b* dunfell**

***cd poky***

***git clone git://git.openembedded.org/meta-openembedded --depth 1 -b* dunfell**

***git clone***[***https://github.com/meta-qt5/meta-qt5.git***](https://github.com/meta-qt5/meta-qt5.git)***--depth 1 -b* dunfell**

***git clone***[***https://github.com/voloviq/meta-licheepizero***](https://github.com/voloviq/meta-licheepizero)***--depth 1 -b* dunfell**

if you have problem with download from   
***git clone git://git.openembedded.org/meta-openembedded --depth 1 -b* dunfell**

you can use follow command:

sudo apt-get install tor  
then  
***torify git clone git://git.openembedded.org/meta-openembedded --depth 1 -b* dunfell**

3. Select directory to build Linux

Zero version  
***source oe-init-build-env ~/yocto/build/licheepizero***  
Zero Dock version  
***source oe-init-build-env ~/yocto/build/licheepizero-dock***

4. Modify bblayers.conf(located in ~/yocto/build/licheepizero/conf(or licheepizero-dock/conf))

*BBLAYERS ?= " \  
${HOME}/yocto/poky/meta \  
${HOME}/yocto/poky/meta-poky \  
${HOME}/yocto/poky/meta-openembedded/meta-oe \  
${HOME}/yocto/poky/meta-openembedded/meta-networking \  
${HOME}/yocto/poky/meta-openembedded/meta-python \  
${HOME}/yocto/poky/meta-qt5 \  
${HOME}/yocto/poky/meta-licheepizero \  
"*

**Note:** Please adapt PATH of conf/bblayers.conf if necessary.

5. Modify or align following elements in local.conf(located in ~/yocto/build/licheepizero/conf(or licheepizero-dock/conf)) file

MACHINE ??= "licheepizero-dock"

or

MACHINE ??= "licheepizero"

DL\_DIR = "${HOME}/yocto/downloads"

SSTATE\_DIR = "${HOME}/yocto/sstate-cache"

TMPDIR = "${HOME}/yocto/tmp"

at the end add some option if necessary

*RM\_OLD\_IMAGE = "1"*

*INHERIT += "rm\_work"*

**Note:** Please adapt rest of conf/local.conf parameters if necessary.  
if you want to use some package (it’s for our QT project not necessary) more add follow command to the end of local.conf file

*IMAGE\_INSTALL\_append = "qtserialport minicom packagegroup-core-buildessential nano qtsvg qtsvg-plugins qtdeclarative qtdeclarative-qmlplugins qtdeclarative-plugins qtquickcontrols-qmlplugins qtquickcontrols2 qtquickcontrols2-qmlplugins qtgraphicaleffects-qmlplugins"*

*CORE\_IMAGE\_EXTRA\_INSTALL += "sqlite3"*

*#IMAGE\_INSTALL\_append = " qtbase-plugins qtbase-tools qtimageformats-plugins qtsystems qtsystems-tools qtsystems-qmlplugins qtscript qtlocation-plugins qtlocation-qmlplugins procps"*

*IMAGE\_INSTALL\_append = " minicom packagegroup-core-buildessential"*

*IMAGE\_INSTALL\_append = " qtdeclarative qtdeclarative-plugins qtdeclarative-qmlplugins qtgraphicaleffects-qmlplugins qtsvg qtsvg-plugins qtquickcontrols2 qtquickcontrols-qmlplugins qtquickcontrols2-qmlplugins"*

*#CORE\_IMAGE\_EXTRA\_INSTALL += "qtsvg"*

*#IMAGE\_INSTALL\_append = " qtsvg"*

*MACHINE\_EXTRA\_RRECOMMENDS = " kernel-modules"*

Example of this file exist in [repository](https://github.com/new/import) (local.conf)

6. Build objects

Issue from console one of the following option

console image

***bitbake console-image***

sato (X11) image

***bitbake sato-core-image***

qt5 image

***bitbake qt5-image***

qt5 toolchain sdk

***bitbake meta-toolchain-qt5***

7. After compilation images appear in

Zero version  
***~/yocto/tmp/deploy/images/licheepizero***  
Zero Dock version  
***~/yocto/tmp/deploy/images/licheepizero-dock***

8. Insert SD CARD into dedicated CARD slot and issue the following command to write an image

**Note:**  
Be 100% sure to provide a valid device name (of=/dev/**sde**). Wrong name “/dev/sde” damage Your system file !

Zero version  
***sudo dd if=~/yocto/tmp/deploy/images/licheepizero/qt5-image-licheepizero.sunxi-sdimg of=/dev/sde bs=1024***  
Zero Dock verison  
***sudo dd if=~/yocto/tmp/deploy/images/licheepizero-dock/qt5-image-licheepizero-dock.sunxi-sdimg of=/dev/sde bs=1024***

**Limitation**

rootfs-resize not working (SD CARD size can be resized manualy)

no wiringpi or similar library to control GPIO **in** C code

discover problem when WiFi connected to access point (probably some drivers issues), nevertheless WiFi works

Source link: <https://www.emsyslabs.com/how-to-compile-linux-with-qt5-option-using-yocto-for-lichee-pi-zero/>