Qt

Hands-on 2023 Linux Academy

BEZPŁATNE WARSZTATY ▪ Tworzenie **GUI** za pomocą **Qt library** na platformie SpaceSOM-8Mplus

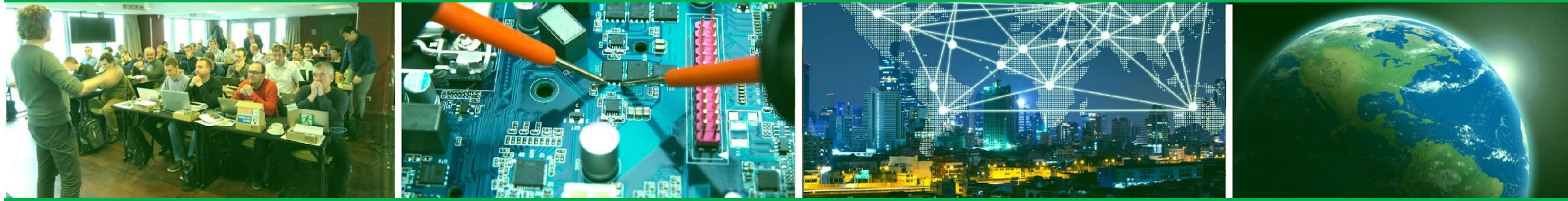
KRAKÓW **26.09.2023**

KATOWICE **27.09.2023**

elhurt
DYSTRYBUCJA

SoMLabs

TECHDAYS



Hands-on Linux Academy 2023 - programming GUI in Qt library for SpaceSOM-8Mplus

SoMLabs

SOM
System on Module

CB
Carrier Board

DK
Development Kit

Engineering
Since 2003 delivering proven designs

Agenda

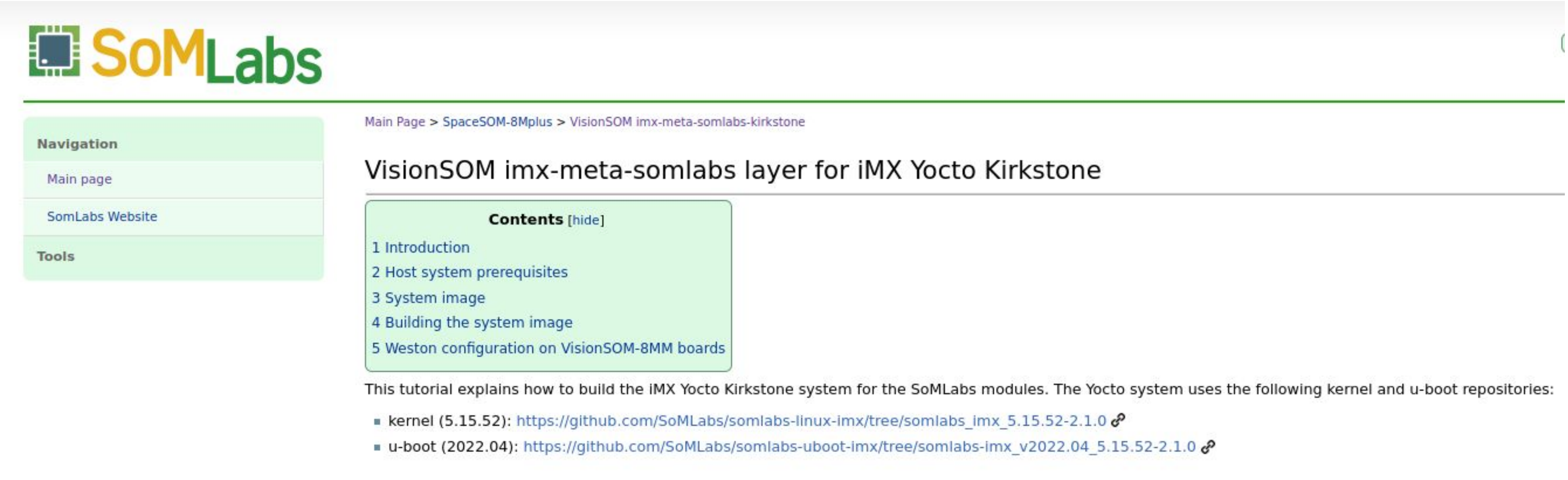
- ❑ Building Yocto system image
- ❑ Modifying kernel sources
- ❑ Programming SpaceSOM-8Mplus board
- ❑ Using Qt Creator for GUI development
- ❑ Widget application
- ❑ Sockets and slots in Qt
- ❑ Handling touch and button inputs
- ❑ QML application
- ❑ Creating Yocto recipes

Exercises

- Lab 1: compiling system image
- Lab 2: programming and connecting to SpaceSOM-8Mplus board
- Lab 3: building Qt widget application
- Lab 4: controlling LED from GUI
- Lab 5: handling GPIO input
- Lab 6: video playback
- Lab 7: building QML application
- Lab 8: controlling LED from QML GUI
- Lab 9: video playback in QML
- Lab 10: creating new Yocto recipe
- Lab 11: ???

Building system image

- https://wiki.somlabs.com/index.php/VisionSOM_imx-meta-somlabs-kirkstone



Main Page > SpaceSOM-8Mplus > VisionSOM imx-meta-somlabs-kirkstone

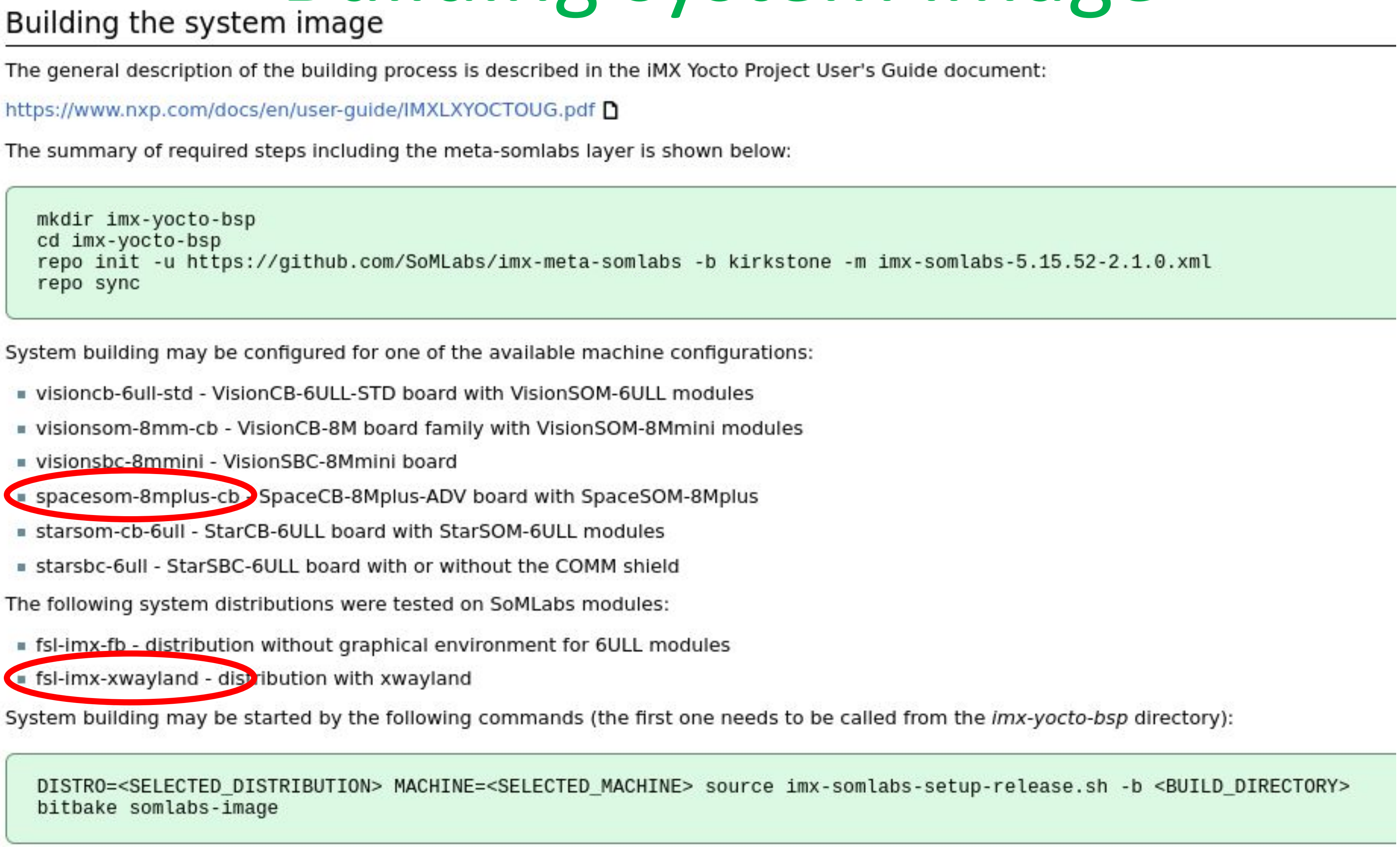
VisionSOM imx-meta-somlabs layer for iMX Yocto Kirkstone

Contents [hide]

This tutorial explains how to build the iMX Yocto Kirkstone system for the SoMLabs modules. The Yocto system uses the following kernel and u-boot repositories:

- kernel (5.15.52): https://github.com/SoMLabs/somlabs-linux-imx/tree/somlabs_imx_5.15.52-2.1.0
- u-boot (2022.04): https://github.com/SoMLabs/somlabs-uboot-imx/tree/somlabs-imx_v2022.04_5.15.52-2.1.0

Building system image



Building the system image

The general description of the building process is described in the iMX Yocto Project User's Guide document:

https://www.nxp.com/docs/en/user-guide/IMXLXYOCTOUG.pdf

The summary of required steps including the meta-somlabs layer is shown below:

```
mkdir imx-yocto-bsp
cd imx-yocto-bsp
repo init -u https://github.com/SoMLabs/imx-meta-somlabs -b kirkstone -m imx-somlabs-5.15.52-2.1.0.xml
repo sync
```

System building may be configured for one of the available machine configurations:

- visioncb-6ull-std - VisionCB-6ULL-STD board with VisionSOM-6ULL modules
- visionsom-8mm-cb - VisionCB-8M board family with VisionSOM-8Mmini modules
- visionsbc-8mmini - VisionSBC-8Mmini board
- spacesom-8mplus-cb - SpaceCB-8Mplus-ADV board with SpaceSOM-8Mplus
- starsom-cb-6ull - StarCB-6ULL board with StarSOM-6ULL modules
- starsbc-6ull - StarSBC-6ULL board with or without the COMM shield

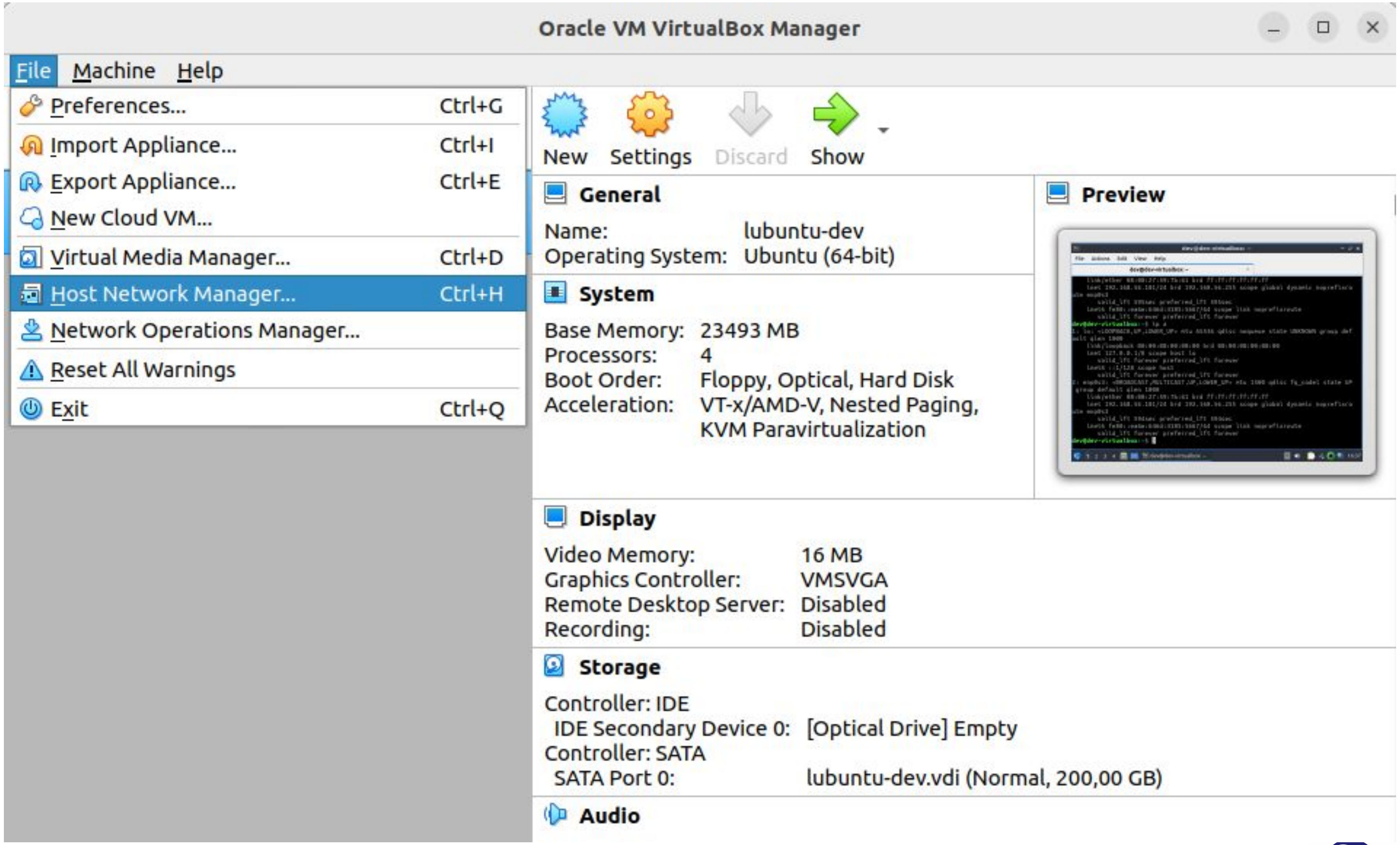
The following system distributions were tested on SoMLabs modules:

- fsl-imx-fb - distribution without graphical environment for 6ULL modules
- fsl-imx-xwayland - distribution with xwayland

System building may be started by the following commands (the first one needs to be called from the *imx-yocto-bsp* directory):

```
DISTRO=<SELECTED_DISTRIBUTION> MACHINE=<SELECTED_MACHINE> source imx-somlabs-setup-release.sh -b <BUILD_DIRECTORY>
bitbake somlabs-image
```

Virtual development machine


Oracle VM VirtualBox Manager
File
Machine
Help
Preferences... Ctrl+G
Import Appliance... Ctrl+I
Export Appliance... Ctrl+E
New Cloud VM...
Virtual Media Manager... Ctrl+D
Host Network Manager... Ctrl+H
Network Operations Manager...
Reset All Warnings
Exit Ctrl+Q
New
Settings
Discard
Show
Preview

General

Name: lubuntu-dev
Operating System: Ubuntu (64-bit)

System

Base Memory: 23493 MB
Processors: 4
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization

Display

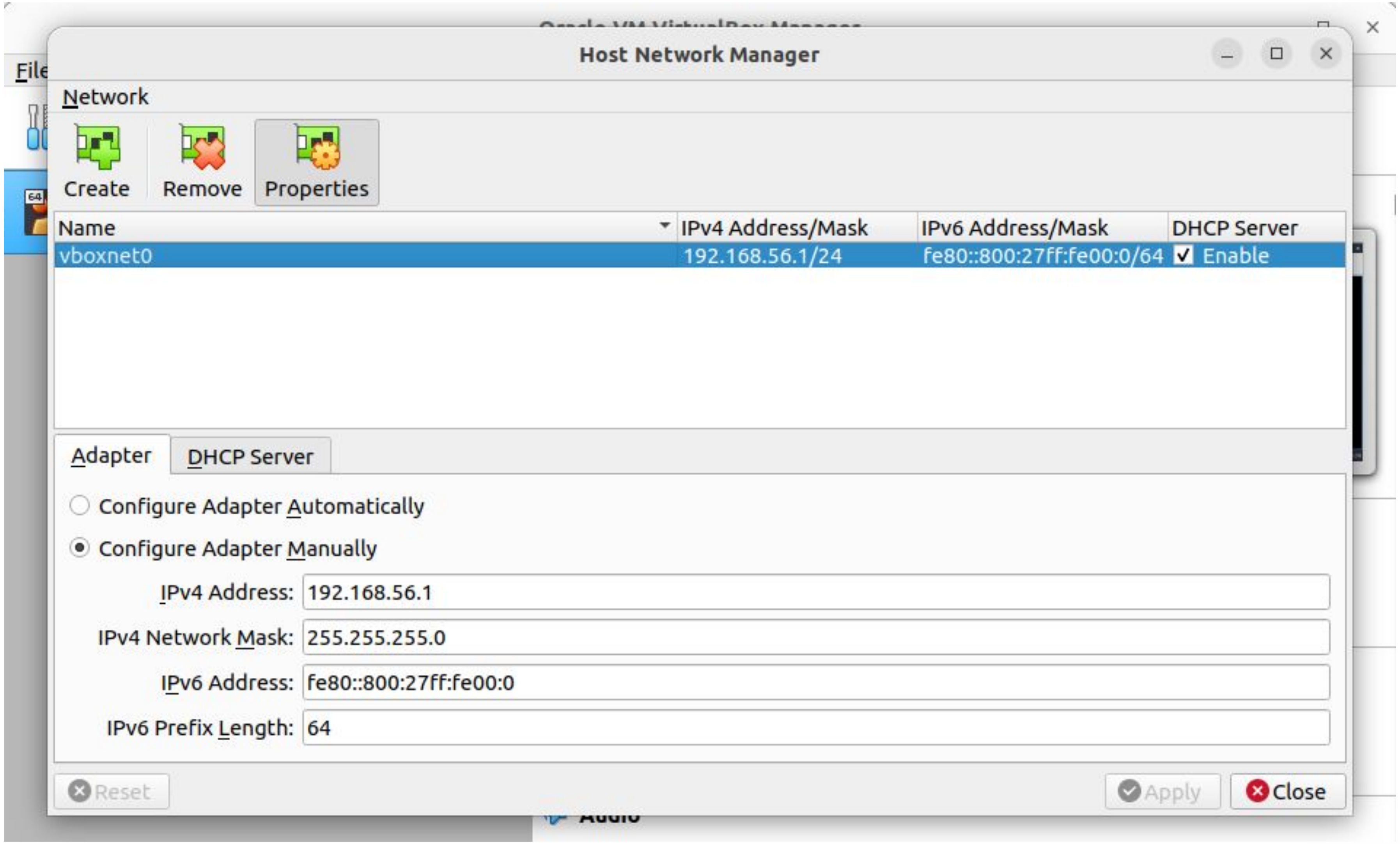
Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Secondary Device 0: [Optical Drive] Empty
Controller: SATA
SATA Port 0: lubuntu-dev.vdi (Normal, 200,00 GB)

Audio

Virtual development machine



Name	IPv4 Address/Mask	IPv6 Address/Mask	DHCP Server
vboxnet0	192.168.56.1/24	fe80::800:27ff:fe00:0/64	☑ Enable

○ Configure Adapter Automatically

● Configure Adapter Manually

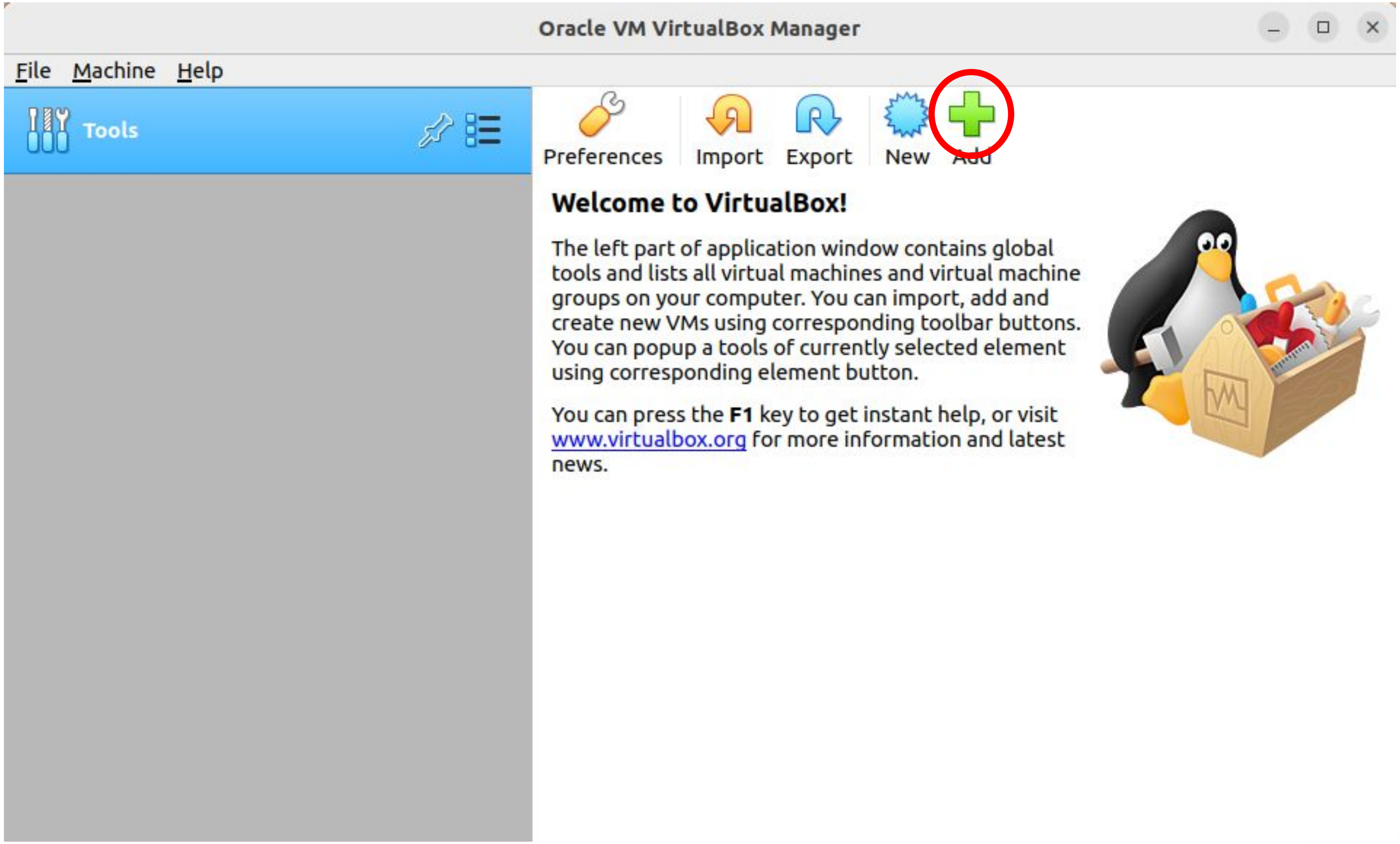
IPv4 Address: 192.168.56.1

IPv4 Network Mask: 255.255.255.0

IPv6 Address: fe80::800:27ff:fe00:0

IPv6 Prefix Length: 64

Virtual development machine

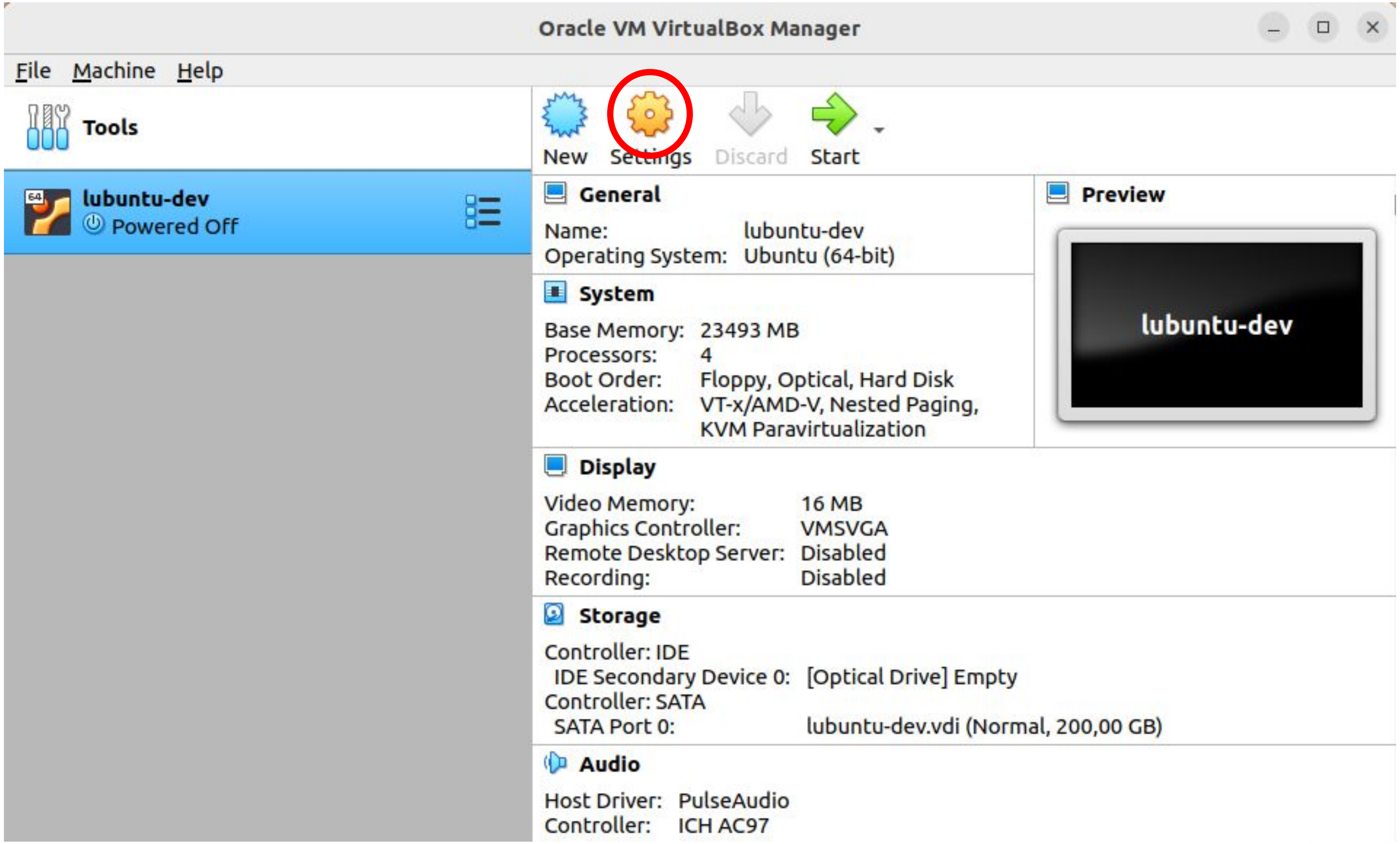


Welcome to VirtualBox!

The left part of application window contains global tools and lists all virtual machines and virtual machine groups on your computer. You can import, add and create new VMs using corresponding toolbar buttons. You can popup a tools of currently selected element using corresponding element button.

You can press the **F1** key to get instant help, or visit www.virtualbox.org for more information and latest news.

Virtual development machine



General

Name: lubuntu-dev
Operating System: Ubuntu (64-bit)

System

Base Memory: 23493 MB
Processors: 4
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization

Display

Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

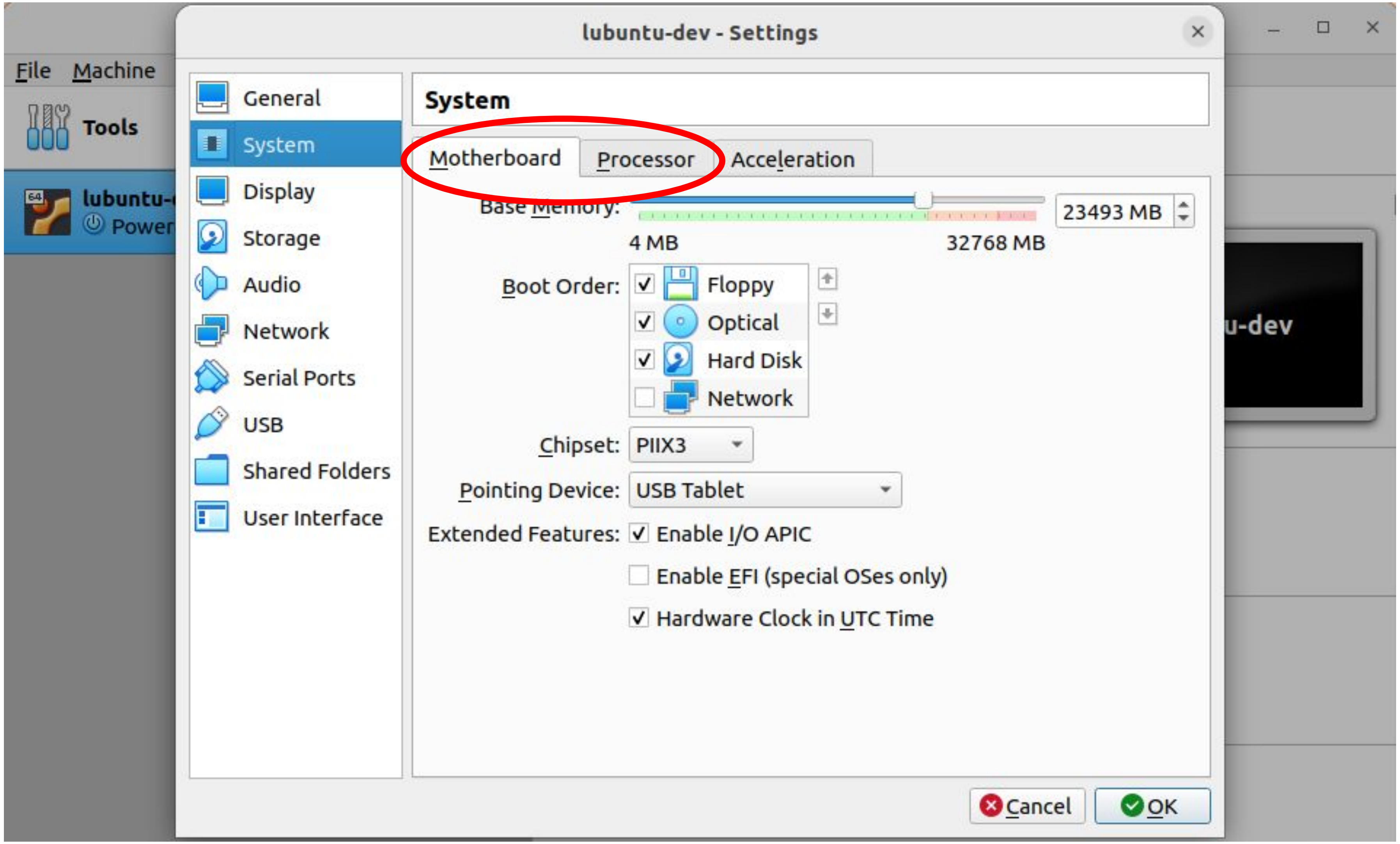
Storage

Controller: IDE
IDE Secondary Device 0: [Optical Drive] Empty
Controller: SATA
SATA Port 0: lubuntu-dev.vdi (Normal, 200,00 GB)

Audio

Host Driver: PulseAudio
Controller: ICH AC97

Virtual development machine



lubuntu-dev - Settings

System

Motherboard | Processor | Acceleration

Base Memory: 23493 MB (4 MB – 32768 MB)

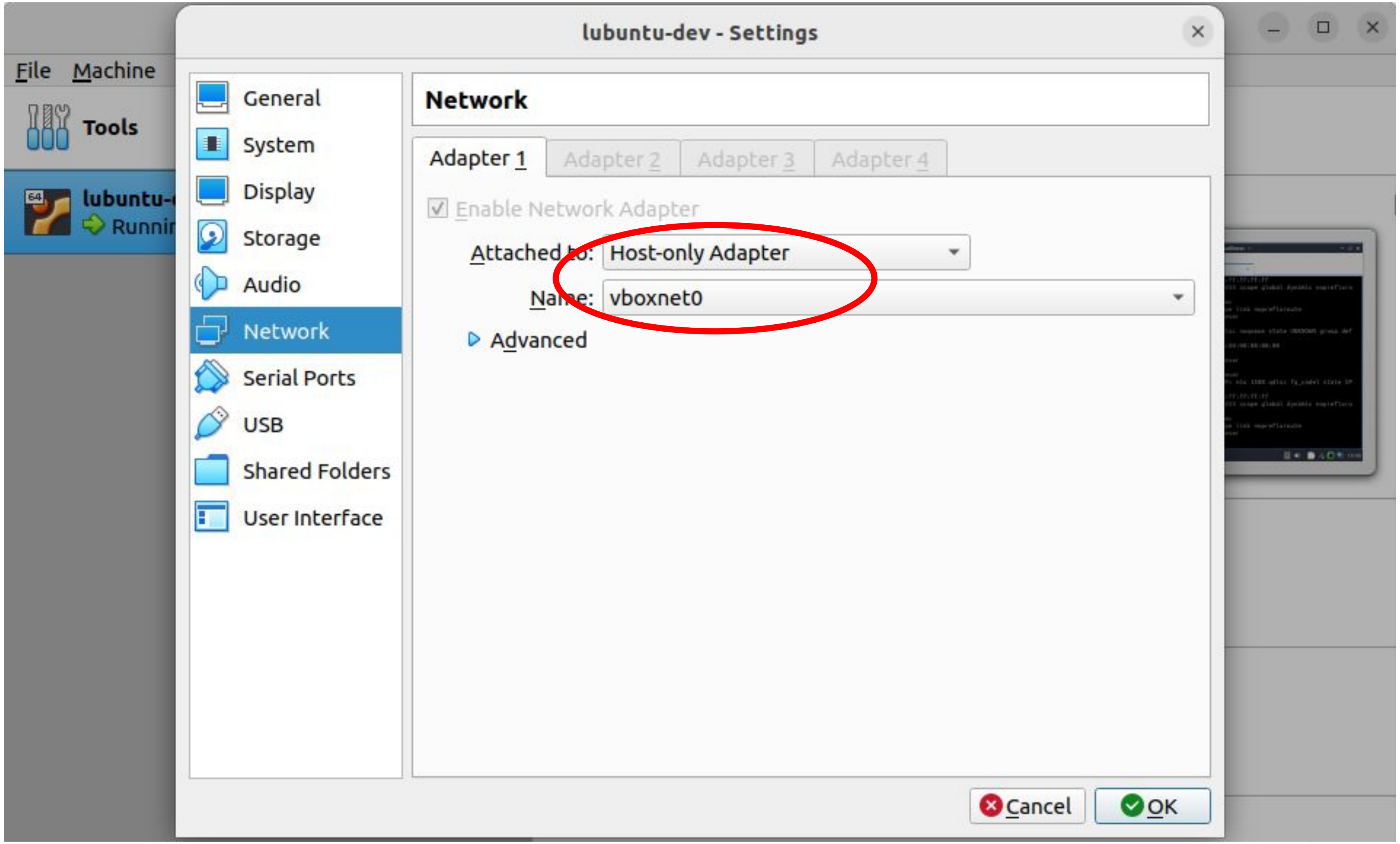
Boot Order:
- ☑ Floppy
- ☑ Optical
- ☑ Hard Disk
- ☐ Network

Chipset: PIIX3

Pointing Device: USB Tablet

Extended Features:
- ☑ Enable I/O APIC
- ☐ Enable EFI (special OSes only)
- ☑ Hardware Clock in UTC Time

Virtual development machine



Network

☑ Enable Network Adapter

Attached to: Host-only Adapter

Name: vboxnet0

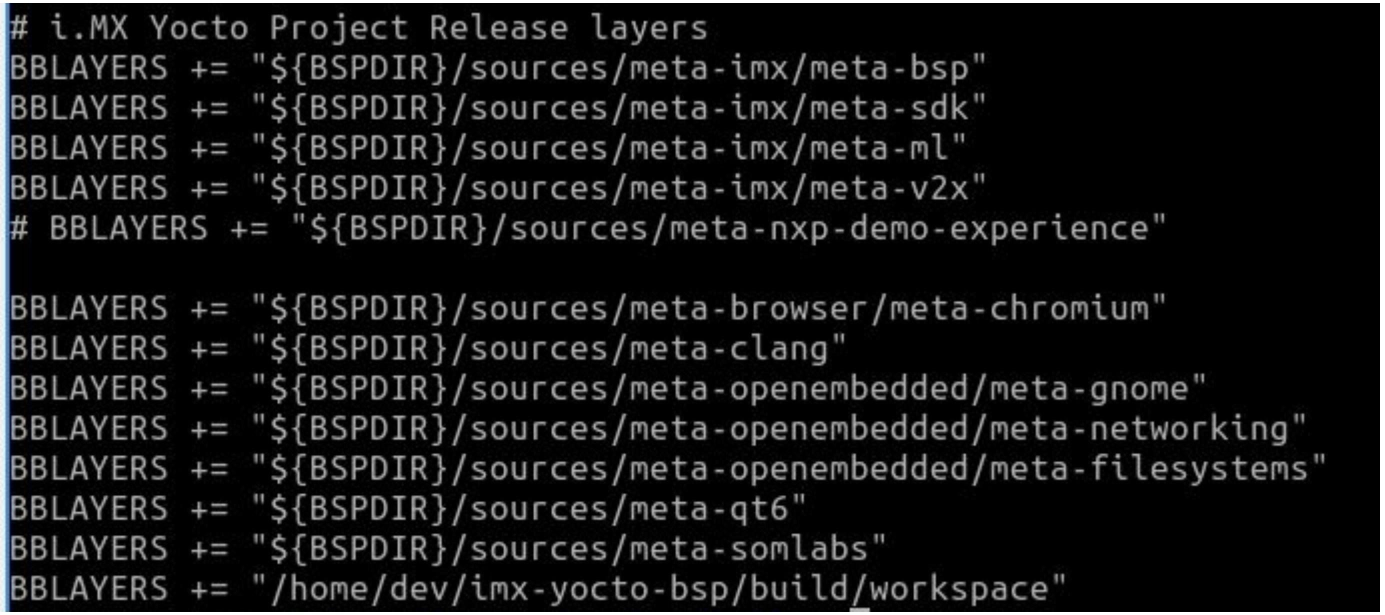
Virtual development machine

- Connecting to the virtual machine (SSH)

  -
    ```
    ssh dev@dev-virtualbox.local
    ```

Lab 1: compiling system image

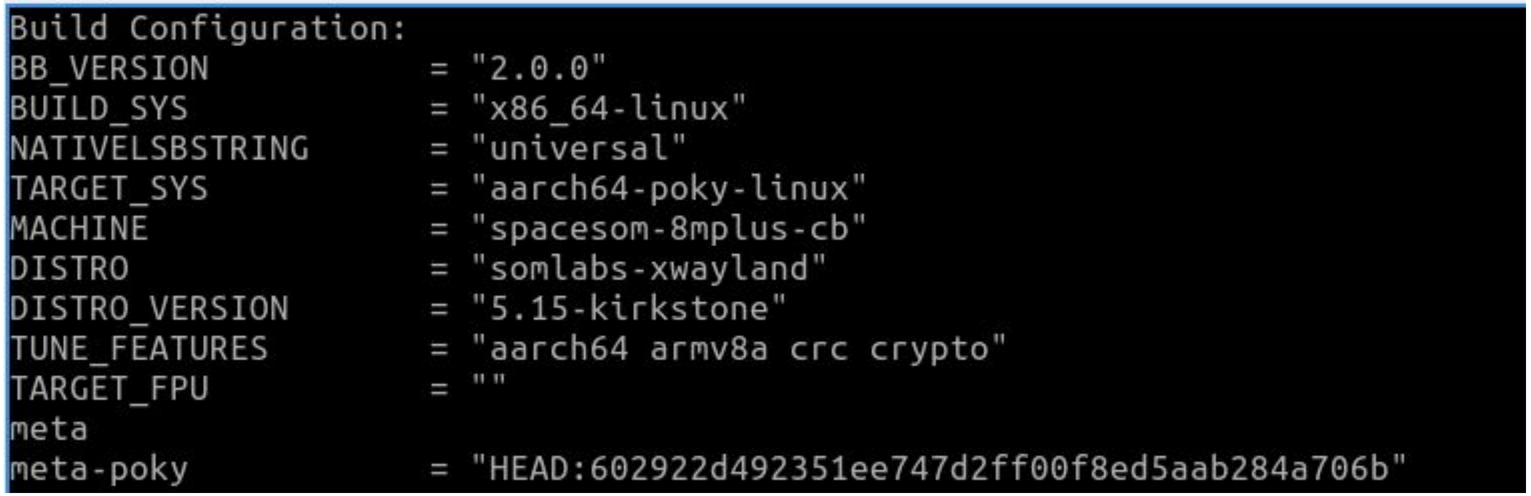
- cd ~/imx-yocto-bsp
- source setup-environment build
- conf/bblayers.conf



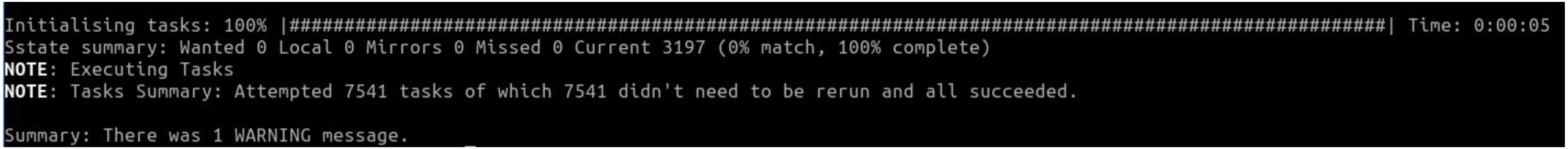
```
# i.MX Yocto Project Release layers
BBLAYERS += "${BSPDIR}/sources/meta-imx/meta-bsp"
BBLAYERS += "${BSPDIR}/sources/meta-imx/meta-sdk"
BBLAYERS += "${BSPDIR}/sources/meta-imx/meta-ml"
BBLAYERS += "${BSPDIR}/sources/meta-imx/meta-v2x"
# BBLAYERS += "${BSPDIR}/sources/meta-nxp-demo-experience"

BBLAYERS += "${BSPDIR}/sources/meta-browser/meta-chromium"
BBLAYERS += "${BSPDIR}/sources/meta-clang"
BBLAYERS += "${BSPDIR}/sources/meta-openembedded/meta-gnome"
BBLAYERS += "${BSPDIR}/sources/meta-openembedded/meta-networking"
BBLAYERS += "${BSPDIR}/sources/meta-openembedded/meta-filesystems"
BBLAYERS += "${BSPDIR}/sources/meta-qt6"
BBLAYERS += "${BSPDIR}/sources/meta-somlabs"
BBLAYERS += "/home/dev/imx-yocto-bsp/build/workspace"
```

Lab 1: compiling system image

- bitbake somlabs-image



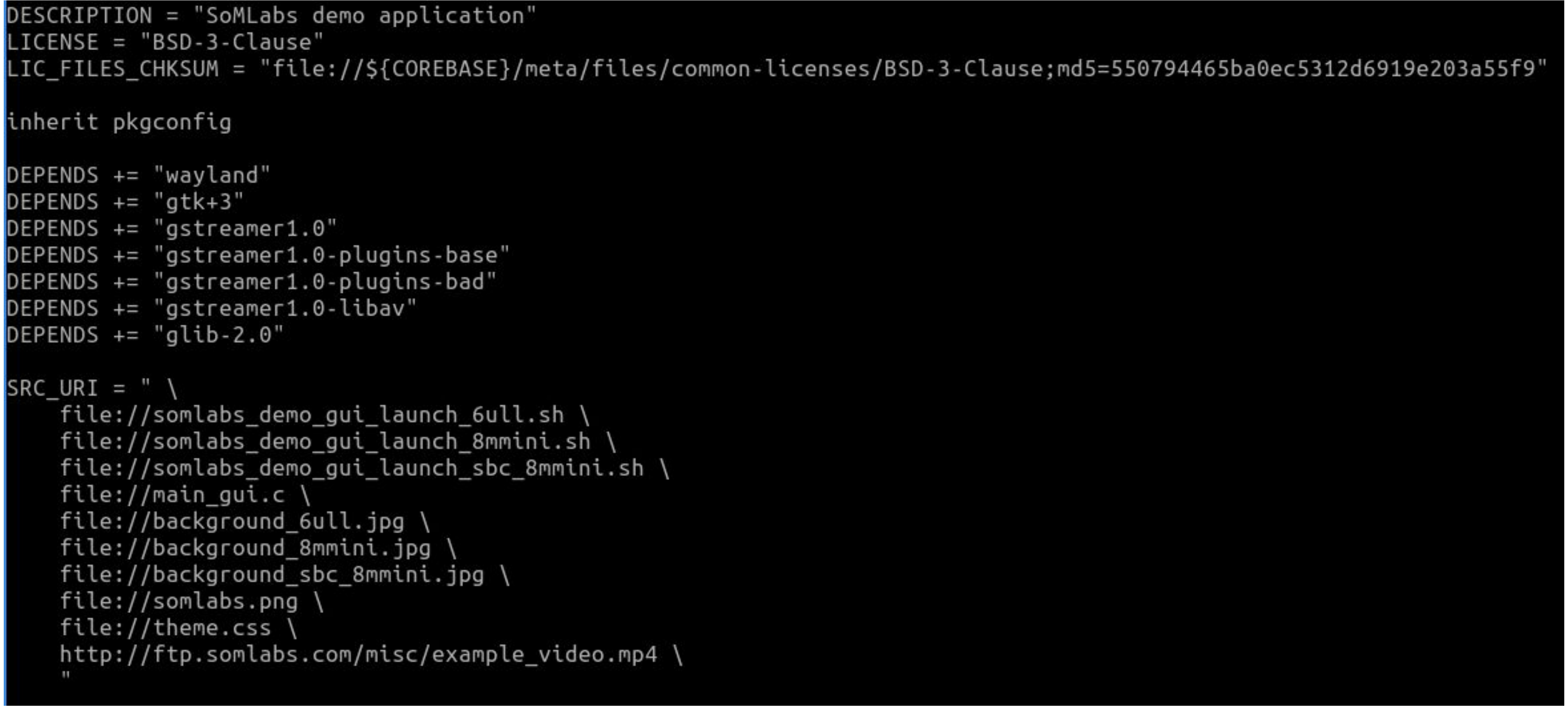
```
Build Configuration:
BB_VERSION           = "2.0.0"
BUILD_SYS            = "x86_64-linux"
NATIVELSBSTRING      = "universal"
TARGET_SYS           = "aarch64-poky-linux"
MACHINE              = "spacesom-8mplus-cb"
DISTRO               = "somlabs-xwayland"
DISTRO_VERSION       = "5.15-kirkstone"
TUNE_FEATURES        = "aarch64 armv8a crc crypto"
TARGET_FPU           = ""
meta
meta-poky            = "HEAD:602922d492351ee747d2ff00f8ed5aab284a706b"
```



```
Initialising tasks: 100% |##########################################################################################| Time: 0:00:05
Sstate summary: Wanted 0 Local 0 Mirrors 0 Missed 0 Current 3197 (0% match, 100% complete)
NOTE: Executing Tasks
NOTE: Tasks Summary: Attempted 7541 tasks of which 7541 didn't need to be rerun and all succeeded.

Summary: There was 1 WARNING message.
```

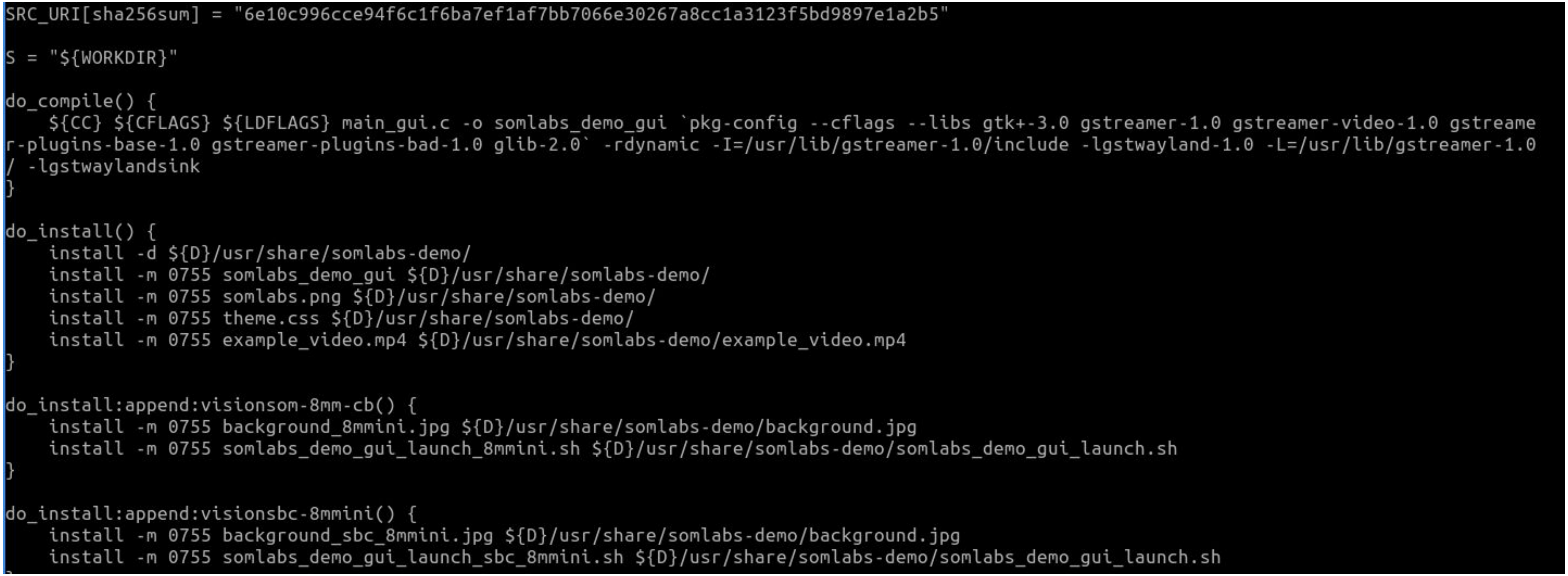
Bitbake recipes

- meta-somlabs/recipes-somlabs/somlabs-demo/somlabs-demo.bb



```
DESCRIPTION = "SoMLabs demo application"
LICENSE = "BSD-3-Clause"
LIC_FILES_CHKSUM = "file://${COREBASE}/meta/files/common-licenses/BSD-3-Clause;md5=550794465ba0ec5312d6919e203a55f9"

inherit pkgconfig

DEPENDS += "wayland"
DEPENDS += "gtk+3"
DEPENDS += "gstreamer1.0"
DEPENDS += "gstreamer1.0-plugins-base"
DEPENDS += "gstreamer1.0-plugins-bad"
DEPENDS += "gstreamer1.0-libav"
DEPENDS += "glib-2.0"

SRC_URI = " \
    file://somlabs_demo_gui_launch_6ull.sh \
    file://somlabs_demo_gui_launch_8mmini.sh \
    file://somlabs_demo_gui_launch_sbc_8mmini.sh \
    file://main_gui.c \
    file://background_6ull.jpg \
    file://background_8mmini.jpg \
    file://background_sbc_8mmini.jpg \
    file://somlabs.png \
    file://theme.css \
    http://ftp.somlabs.com/misc/example_video.mp4 \
    "
```

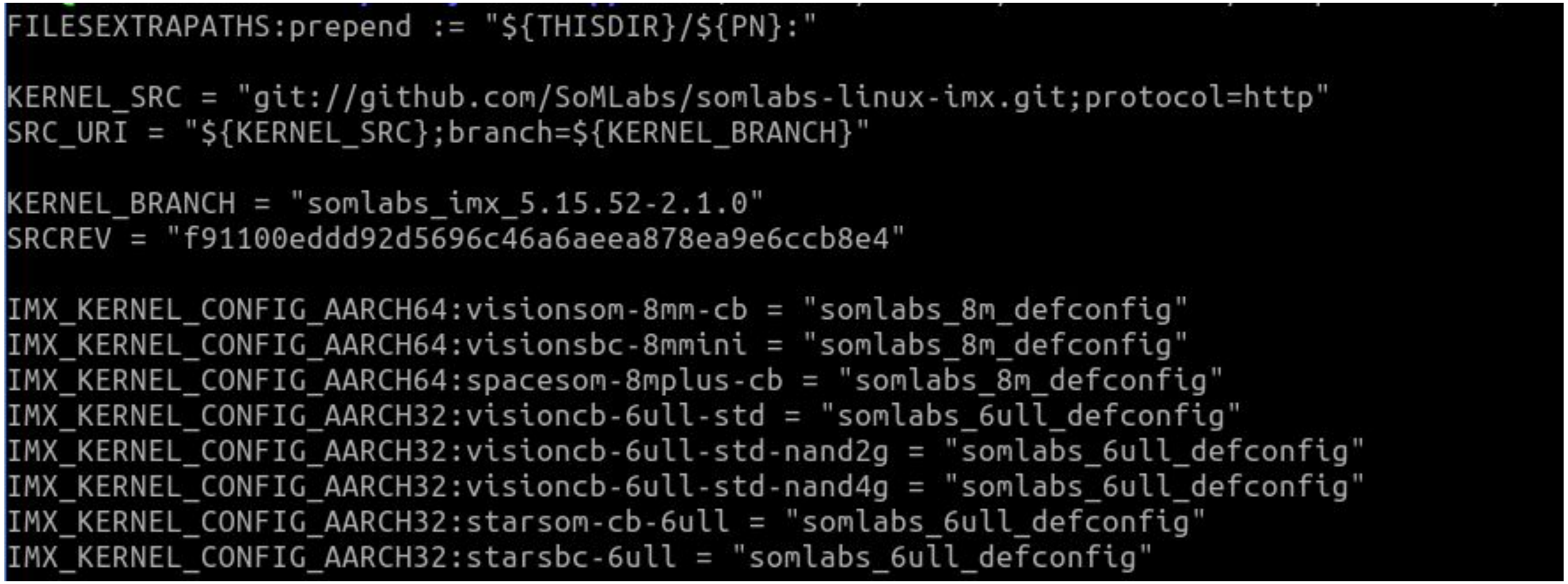
Bitbake recipes



```
SRC_URI[sha256sum] = "6e10c996cce94f6c1f6ba7ef1af7bb7066e30267a8cc1a3123f5bd9897e1a2b5"

S = "${WORKDIR}"

do_compile() {
    ${CC} ${CFLAGS} ${LDFLAGS} main_gui.c -o somlabs_demo_gui `pkg-config --cflags --libs gtk+-3.0 gstreamer-1.0 gstreamer-video-1.0 gstreame
r-plugins-base-1.0 gstreamer-plugins-bad-1.0 glib-2.0` -rdynamic -I=/usr/lib/gstreamer-1.0/include -lgstwayland-1.0 -L=/usr/lib/gstreamer-1.0
/ -lgstwaylandsink
}

do_install() {
    install -d ${D}/usr/share/somlabs-demo/
    install -m 0755 somlabs_demo_gui ${D}/usr/share/somlabs-demo/
    install -m 0755 somlabs.png ${D}/usr/share/somlabs-demo/
    install -m 0755 theme.css ${D}/usr/share/somlabs-demo/
    install -m 0755 example_video.mp4 ${D}/usr/share/somlabs-demo/example_video.mp4
}

do_install:append:visionsom-8mm-cb() {
    install -m 0755 background_8mmini.jpg ${D}/usr/share/somlabs-demo/background.jpg
    install -m 0755 somlabs_demo_gui_launch_8mmini.sh ${D}/usr/share/somlabs-demo/somlabs_demo_gui_launch.sh
}

do_install:append:visionsbc-8mmini() {
    install -m 0755 background_sbc_8mmini.jpg ${D}/usr/share/somlabs-demo/background.jpg
    install -m 0755 somlabs_demo_gui_launch_sbc_8mmini.sh ${D}/usr/share/somlabs-demo/somlabs_demo_gui_launch.sh
```

Bitbake recipes

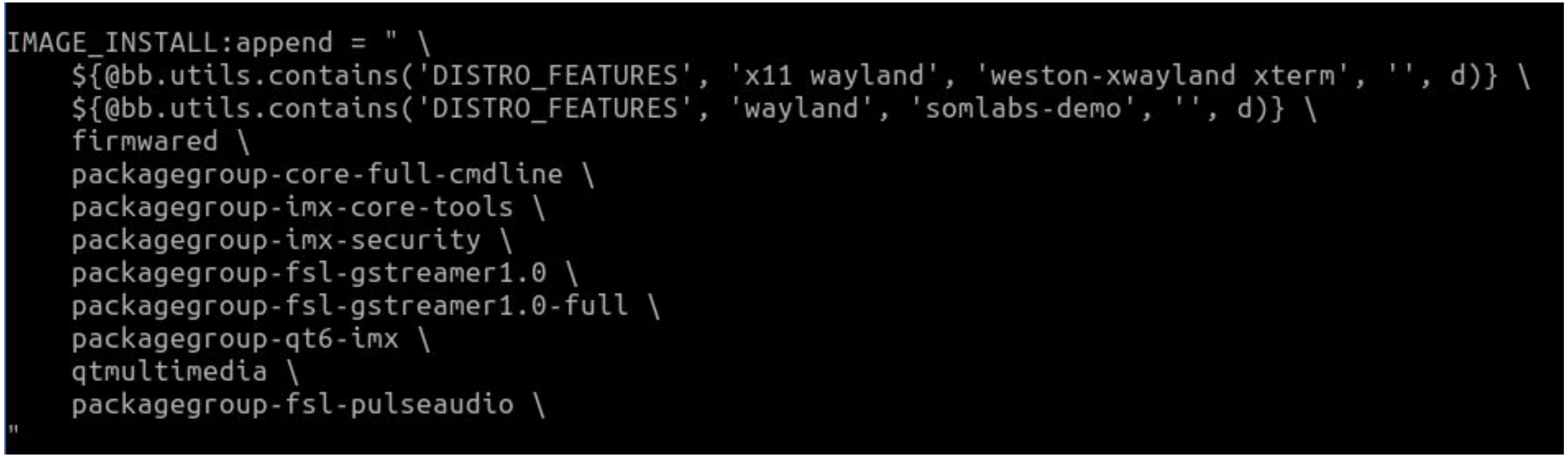
- Kernel:
  - ./meta-imx/meta-bsp/recipes-kernel/linux/linux-imx_5.15.bb
  - ./meta-imx/meta-v2x/recipes-kernel/linux/linux-imx_%.bbappend
  - ./meta-somlabs/recipes-kernel/linux/linux-imx_5.15.bbappend



```
FILESEXTRAPATHS:prepend := "${THISDIR}/${PN}:"

KERNEL_SRC = "git://github.com/SoMLabs/somlabs-linux-imx.git;protocol=http"
SRC_URI = "${KERNEL_SRC};branch=${KERNEL_BRANCH}"

KERNEL_BRANCH = "somlabs_imx_5.15.52-2.1.0"
SRCREV = "f91100eddd92d5696c46a6aeea878ea9e6ccb8e4"

IMX_KERNEL_CONFIG_AARCH64:visionsom-8mm-cb = "somlabs_8m_defconfig"
IMX_KERNEL_CONFIG_AARCH64:visionsbc-8mmini = "somlabs_8m_defconfig"
IMX_KERNEL_CONFIG_AARCH64:spacesom-8mplus-cb = "somlabs_8m_defconfig"
IMX_KERNEL_CONFIG_AARCH32:visioncb-6ull-std = "somlabs_6ull_defconfig"
IMX_KERNEL_CONFIG_AARCH32:visioncb-6ull-std-nand2g = "somlabs_6ull_defconfig"
IMX_KERNEL_CONFIG_AARCH32:visioncb-6ull-std-nand4g = "somlabs_6ull_defconfig"
IMX_KERNEL_CONFIG_AARCH32:starsom-cb-6ull = "somlabs_6ull_defconfig"
IMX_KERNEL_CONFIG_AARCH32:starsbc-6ull = "somlabs_6ull_defconfig"
```

Workshop changes

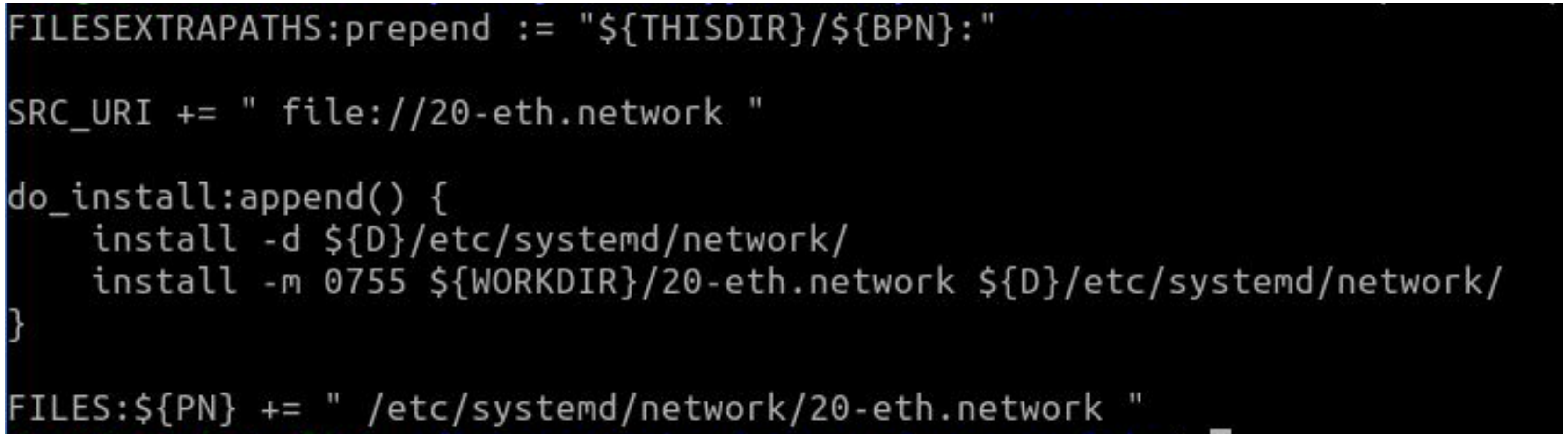
- meta-somlabs/recipes-somlabs/images/somlabs-image.bb



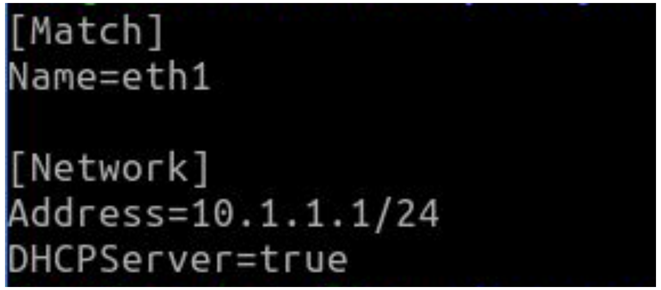
```
IMAGE_INSTALL:append = " \
    ${@bb.utils.contains('DISTRO_FEATURES', 'x11 wayland', 'weston-xwayland xterm', '', d)} \
    ${@bb.utils.contains('DISTRO_FEATURES', 'wayland', 'somlabs-demo', '', d)} \
    firmwared \
    packagegroup-core-full-cmdline \
    packagegroup-imx-core-tools \
    packagegroup-imx-security \
    packagegroup-fsl-gstreamer1.0 \
    packagegroup-fsl-gstreamer1.0-full \
    packagegroup-qt6-imx \
    qtmultimedia \
    packagegroup-fsl-pulseaudio \
"
```

Workshop changes

- meta-somlabs/recipes-core/systemd/systemd_%.bbappend
- meta-somlabs/recipes-core/systemd/systemd/20-eth.network



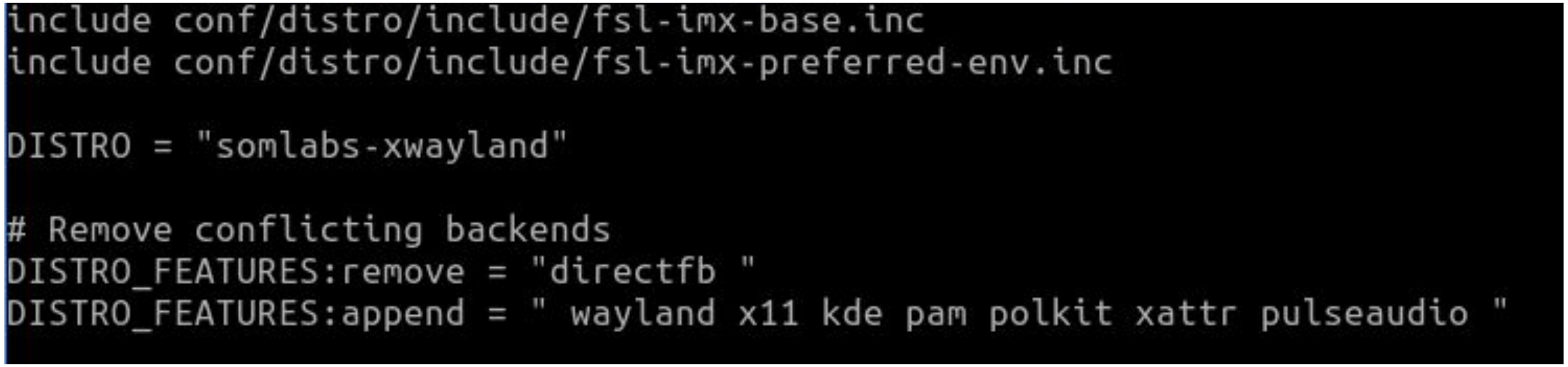
```
FILESEXTRAPATHS:prepend := "${THISDIR}/${BPN}:"

SRC_URI += " file://20-eth.network "

do_install:append() {
    install -d ${D}/etc/systemd/network/
    install -m 0755 ${WORKDIR}/20-eth.network ${D}/etc/systemd/network/
}

FILES:${PN} += " /etc/systemd/network/20-eth.network "
```



```
[Match]
Name=eth1

[Network]
Address=10.1.1.1/24
DHCPServer=true
```

Workshop changes

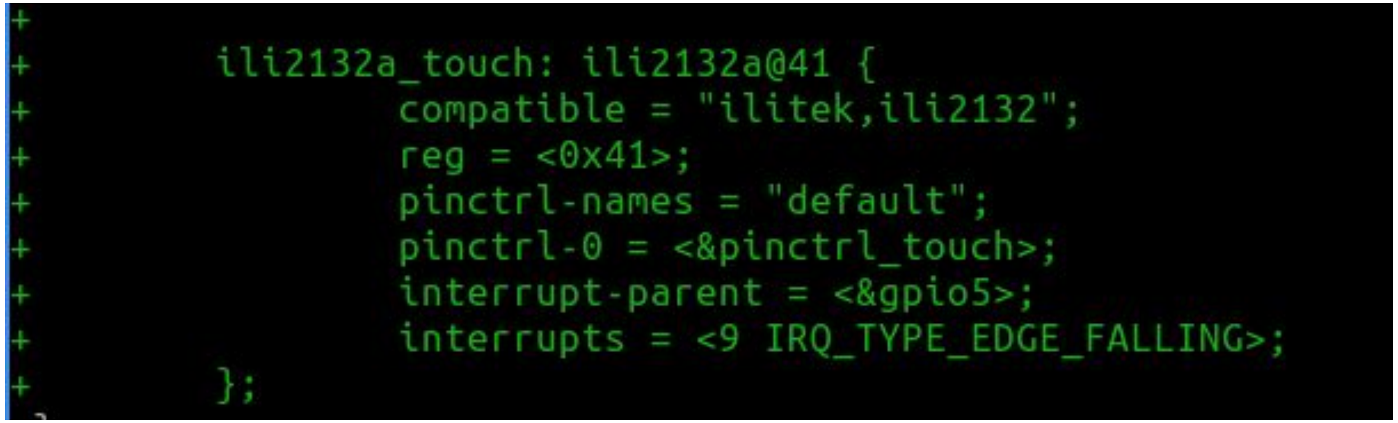
- meta-somlabs/conf/distro/somlabs-xwayland.conf



```
include conf/distro/include/fsl-imx-base.inc
include conf/distro/include/fsl-imx-preferred-env.inc

DISTRO = "somlabs-xwayland"

# Remove conflicting backends
DISTRO_FEATURES:remove = "directfb "
DISTRO_FEATURES:append = " wayland x11 kde pam polkit xattr pulseaudio "
```

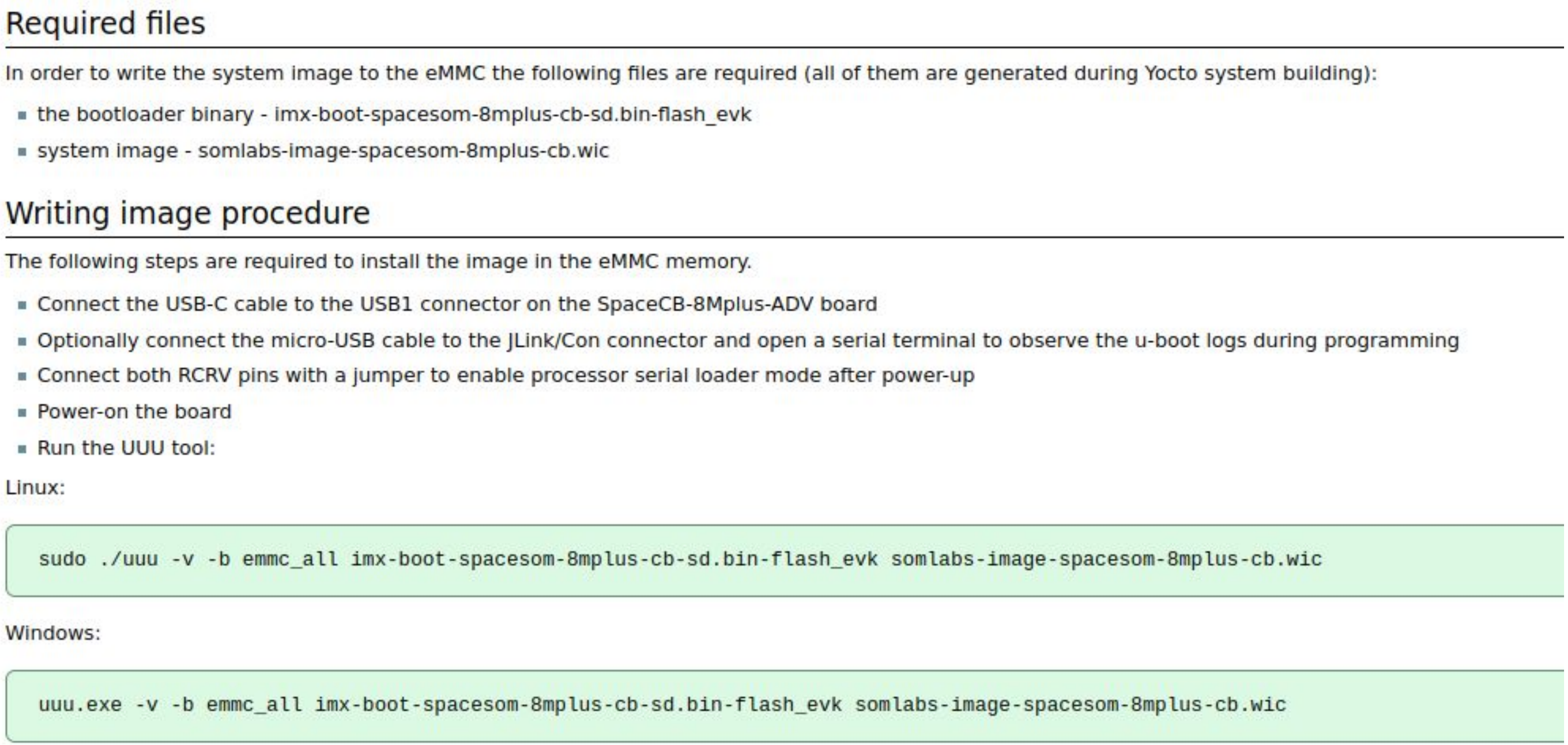
Kernel code changes

- devtool modify linux-imx
- build/workspace/sources/linux-imx
- arch/arm64/boot/dts/freescale/spacesom-8mplus-cb-adv.dts
  - disable DSI and HDMI interfaces
  - enable LVDS touchpanel ilitek,ili2132 on i2c4@41



```
+
+       ili2132a_touch: ili2132a@41 {
+               compatible = "ilitek,ili2132";
+               reg = <0x41>;
+               pinctrl-names = "default";
+               pinctrl-0 = <&pinctrl_touch>;
+               interrupt-parent = <&gpio5>;
+               interrupts = <9 IRQ_TYPE_EDGE_FALLING>;
+       };
```

Lab 2: programming SpaceSOM-8Mplus



Required files

In order to write the system image to the eMMC the following files are required (all of them are generated during Yocto system building):

- the bootloader binary - imx-boot-spacesom-8mplus-cb-sd.bin-flash_evk
- system image - somlabs-image-spacesom-8mplus-cb.wic

Writing image procedure

The following steps are required to install the image in the eMMC memory.

- Connect the USB-C cable to the USB1 connector on the SpaceCB-8Mplus-ADV board
- Optionally connect the micro-USB cable to the JLink/Con connector and open a serial terminal to observe the u-boot logs during programming
- Connect both RCRV pins with a jumper to enable processor serial loader mode after power-up
- Power-on the board
- Run the UUU tool:

Linux:

```
sudo ./uuu -v -b emmc_all imx-boot-spacesom-8mplus-cb-sd.bin-flash_evk somlabs-image-spacesom-8mplus-cb.wic
```

Windows:

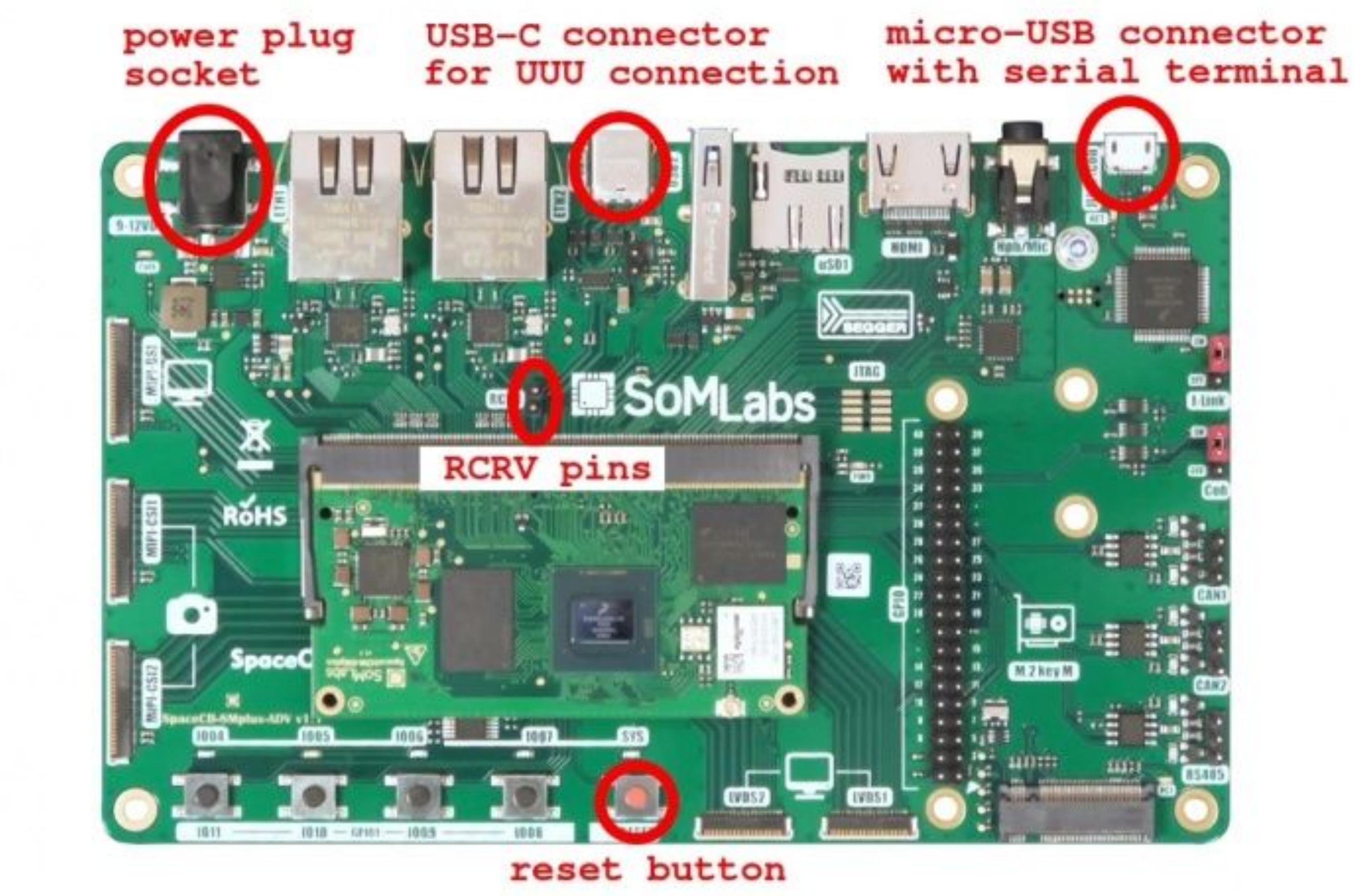
```
uuu.exe -v -b emmc_all imx-boot-spacesom-8mplus-cb-sd.bin-flash_evk somlabs-image-spacesom-8mplus-cb.wic
```

Lab 2: programming SpaceSOM-8Mplus

-
  ```
  scp dev@dev-virtualbox.local:imx-yocto-bsp/build/tmp/deploy/images/spacesom-8mplus-cb/somlabs-image-spacesom-8mplus-cb.wic.zst .
  ```

-
  ```
  scp dev@dev-virtualbox.local:imx-yocto-bsp/build/tmp/deploy/images/spacesom-8mplus-cb/imx-boot-spacesom-8mplus-cb-sd.bin-flash_evk .
  ```
- extract somlabs-image-spacesom-8mplus-cb.wic.zst
  - (Linux) `unzstd somlabs-image-spacesom-8mplus-cb.wic.zst`
  - (Windows) PeaZip

Lab 2: programming SpaceSOM-8Mplus

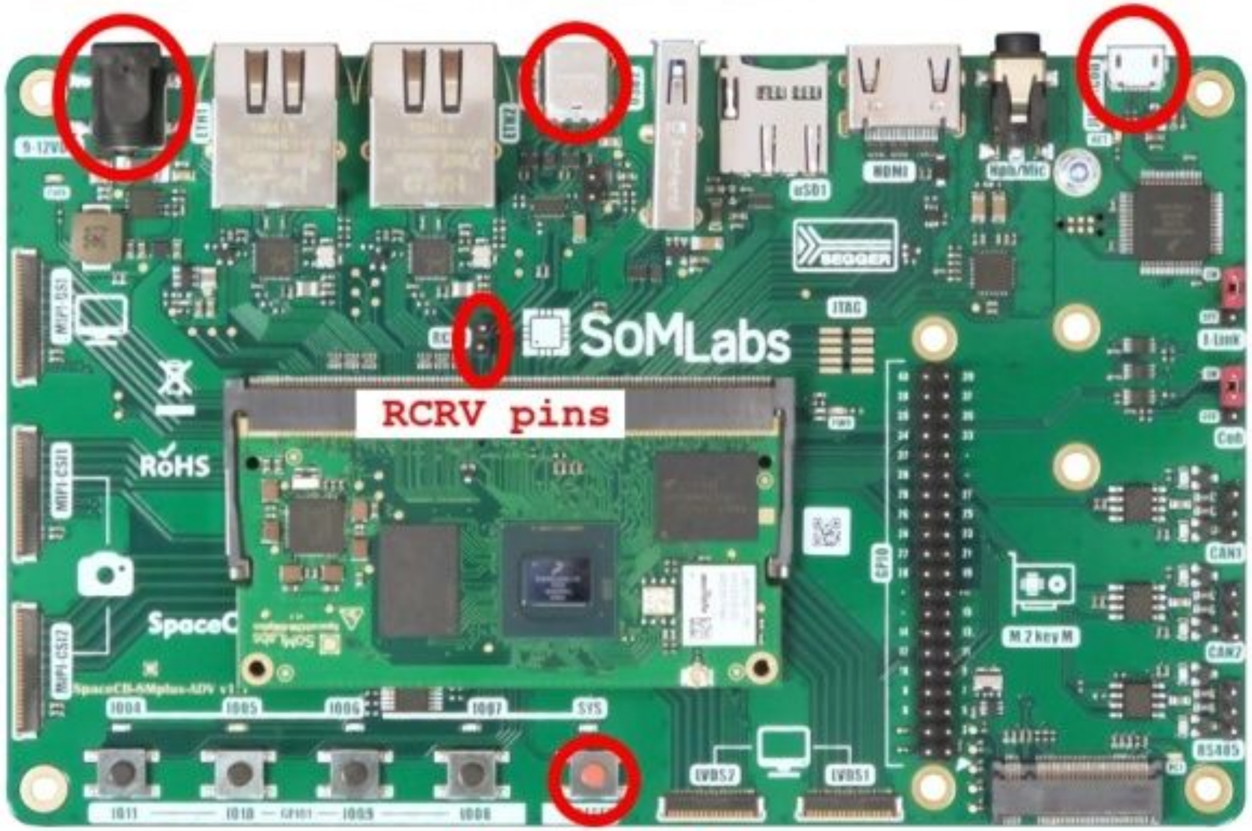

power plug socket
USB-C connector for UUU connection
micro-USB connector with serial terminal
RCRV pins
reset button

Lab 2: programming SpaceSOM-8Mplus

```
sudo ./uuu -v -b emmc_all imx-boot-spacesom-8mplus-cb-sd.bin-flash_evk somlabs-image-spacesom-8mplus-cb.wic
```

```
uuu.exe -v -b emmc_all imx-boot-spacesom-8mplus-cb-sd.bin-flash_evk somlabs-image-spacesom-8mplus-cb.wic
```

Lab 2: programming SpaceSOM-8Mplus



power plug socket

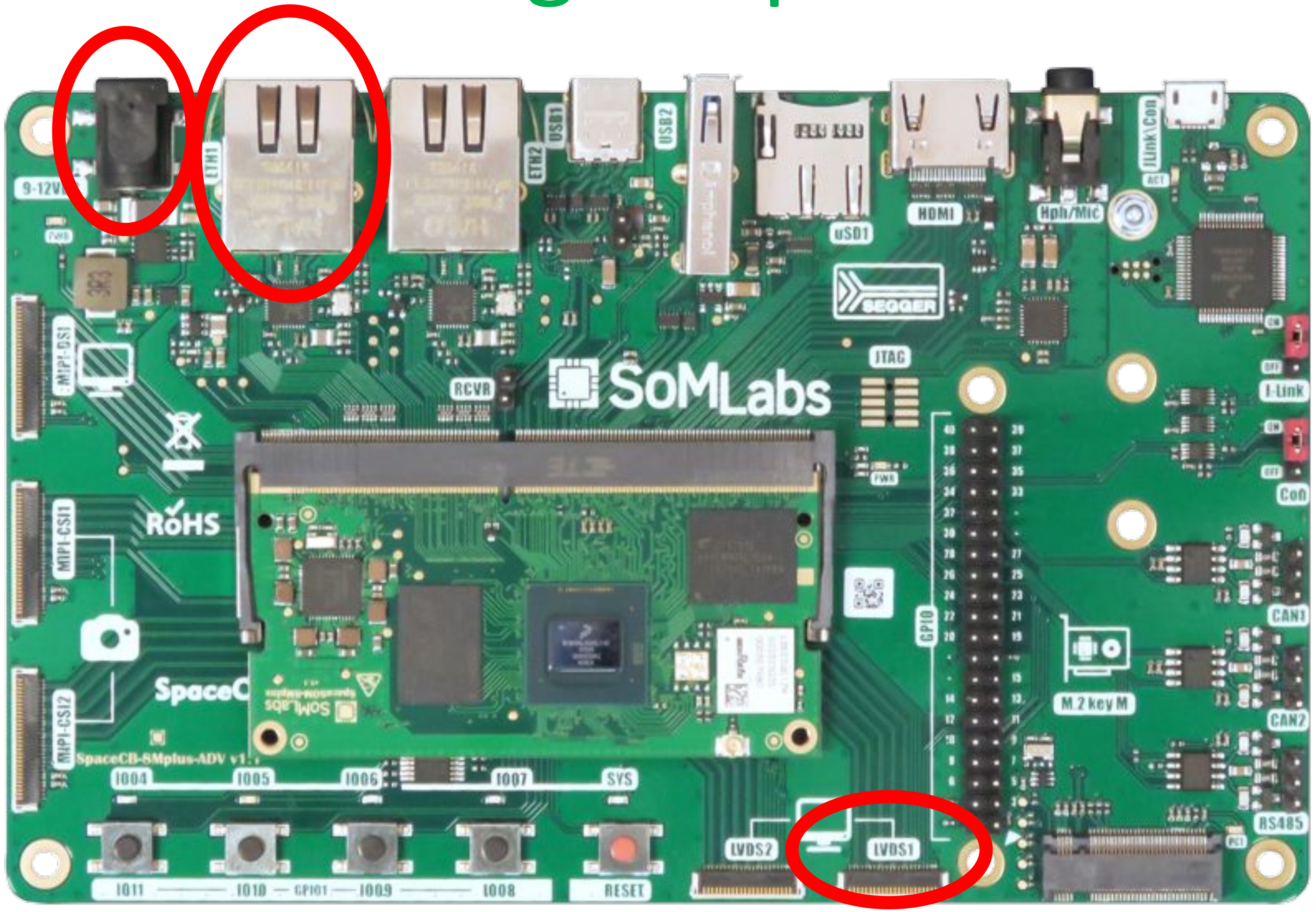
USB-C connector for UUU connection

micro-USB connector with serial terminal

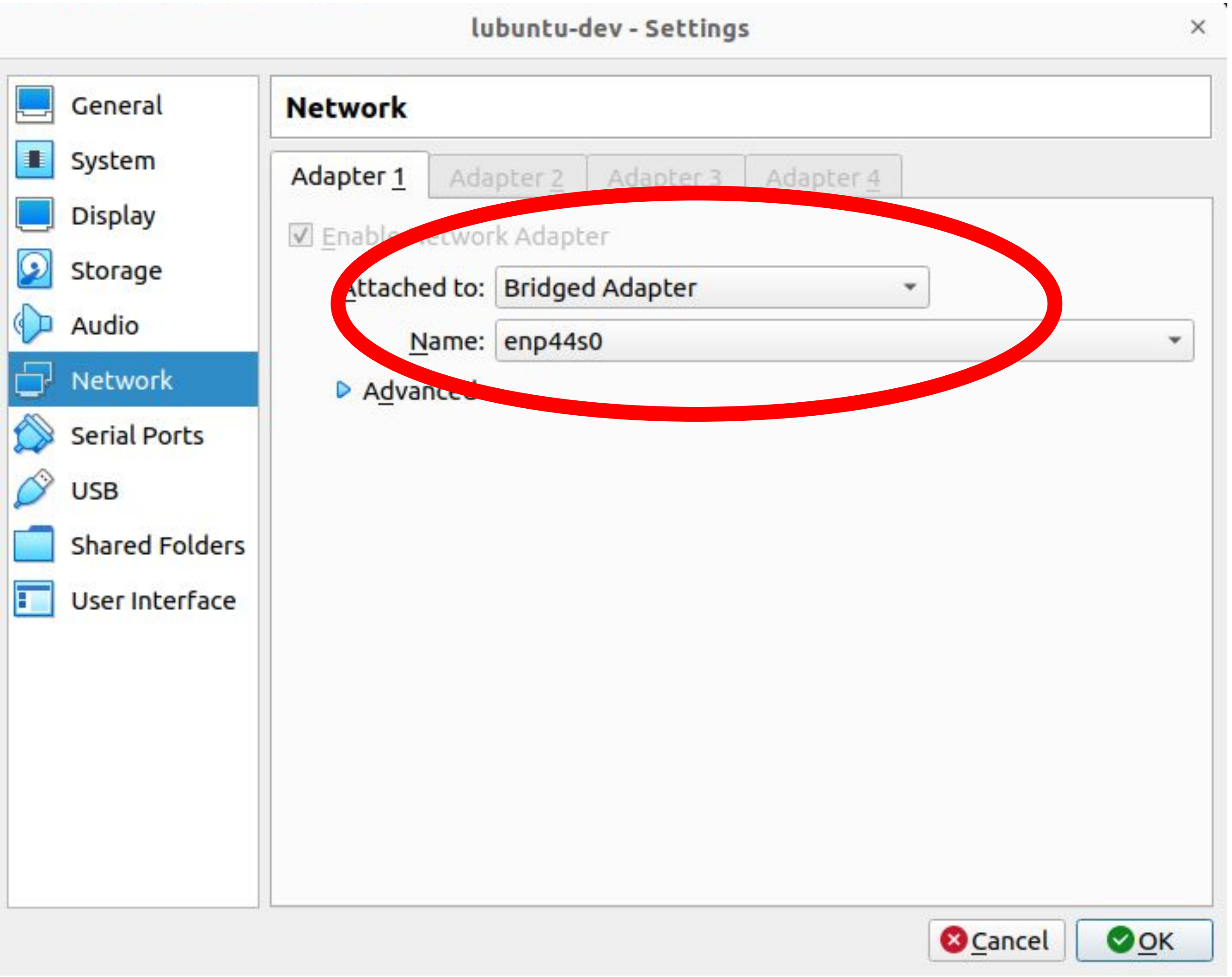
RCRV pins

reset button

Lab2: connecting to SpaceSOM-8Mplus



Lab2: connecting to SpaceSOM-8Mplus



☑ Enabl etwork Adapter

ttached to: Bridged Adapter

Name: enp44s0

▷ Advance

Lab2: connecting to SpaceSOM-8Mplus

- Connecting to the system (SSH)
  - ssh root@spacesom-8mplus-cb.local
  - ssh root@10.1.1.1

Yocto SDK

- meta-somlabs/recipes-somlabs/images/somlabs-image.bb
  - inherit core-image populate_sdk populate_sdk_qt6_base
- bitbake somlabs-image
- build/tmp/deploy/sdk
  - somlabs-xwayland-glibc-x86_64-somlabs-image-armv8a-spacesom-8mplus-cb-toolchain-5.15-kirkstone.sh

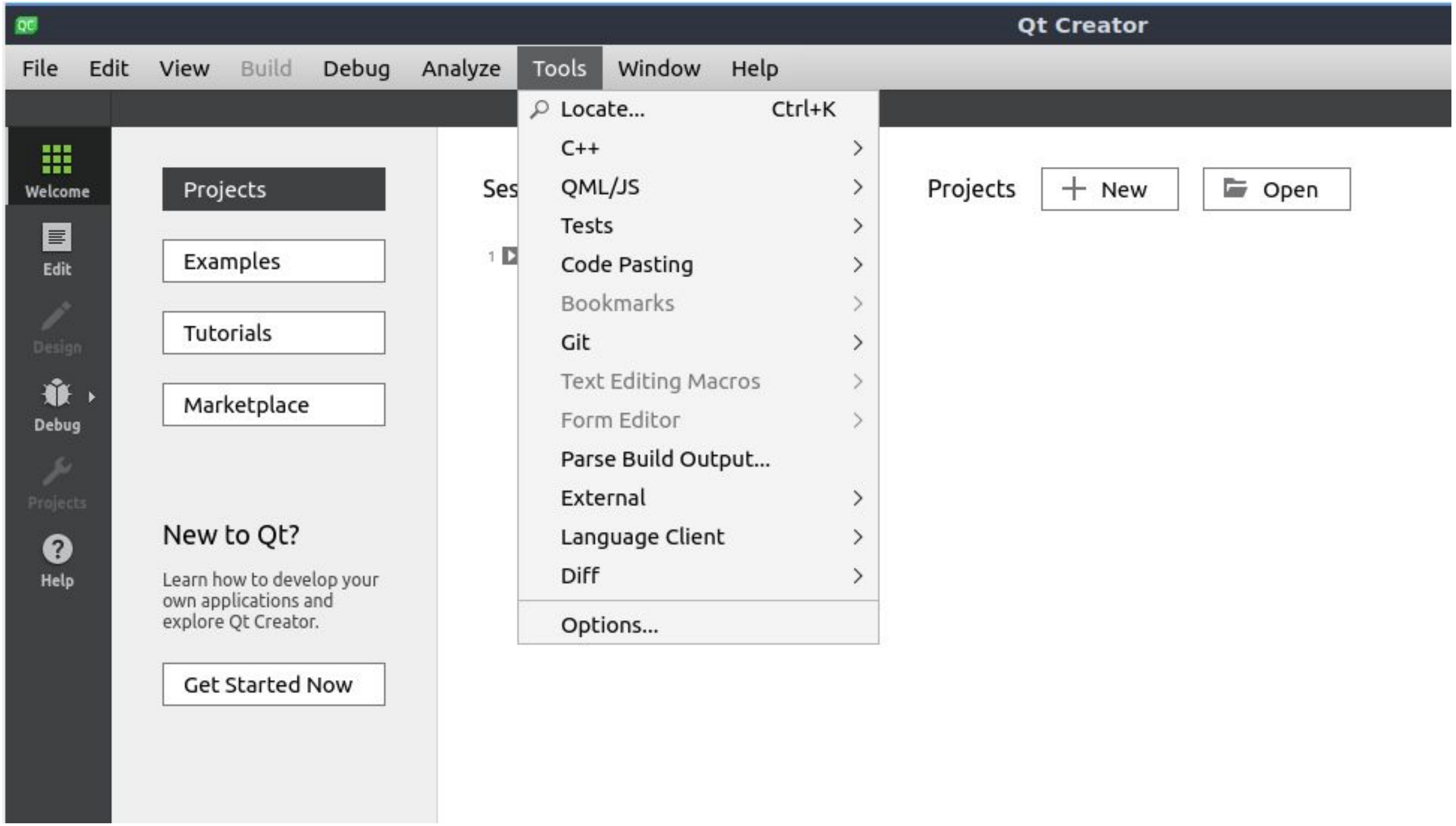
Yocto SDK

```
. /opt/somlabs-xwayland/5.15-kirkstone/environment-setup-armv8a-poky-linux
echo $CC
aarch64-poky-linux-gcc -march=armv8-a+crc+crypto -fstack-protector-strong -O2
-D_FORTIFY_SOURCE=2 -Wformat -Wformat-security -Werror=format-security
--sysroot=/opt/somlabs-xwayland/5.15-kirkstone/sysroots/armv8a-poky-linux
```

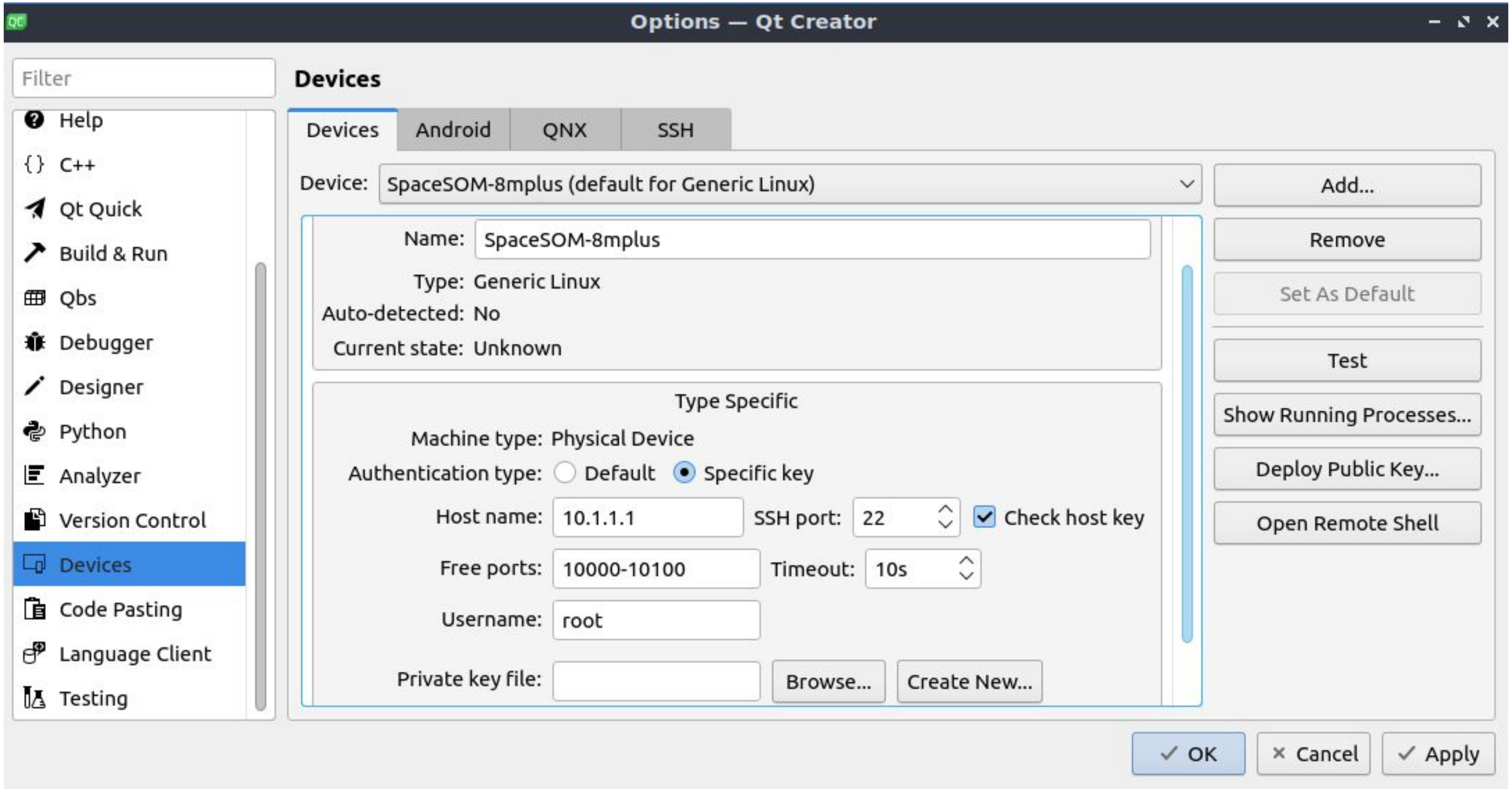
```
find /opt/somlabs-xwayland/ -name "qmake"
/opt/somlabs-xwayland/5.15-kirkstone/sysroots/x86_64-pokysdk-linux/usr/bin/qmake
```

Qt Creator

- ❑ Tools -> Options



Qt Creator

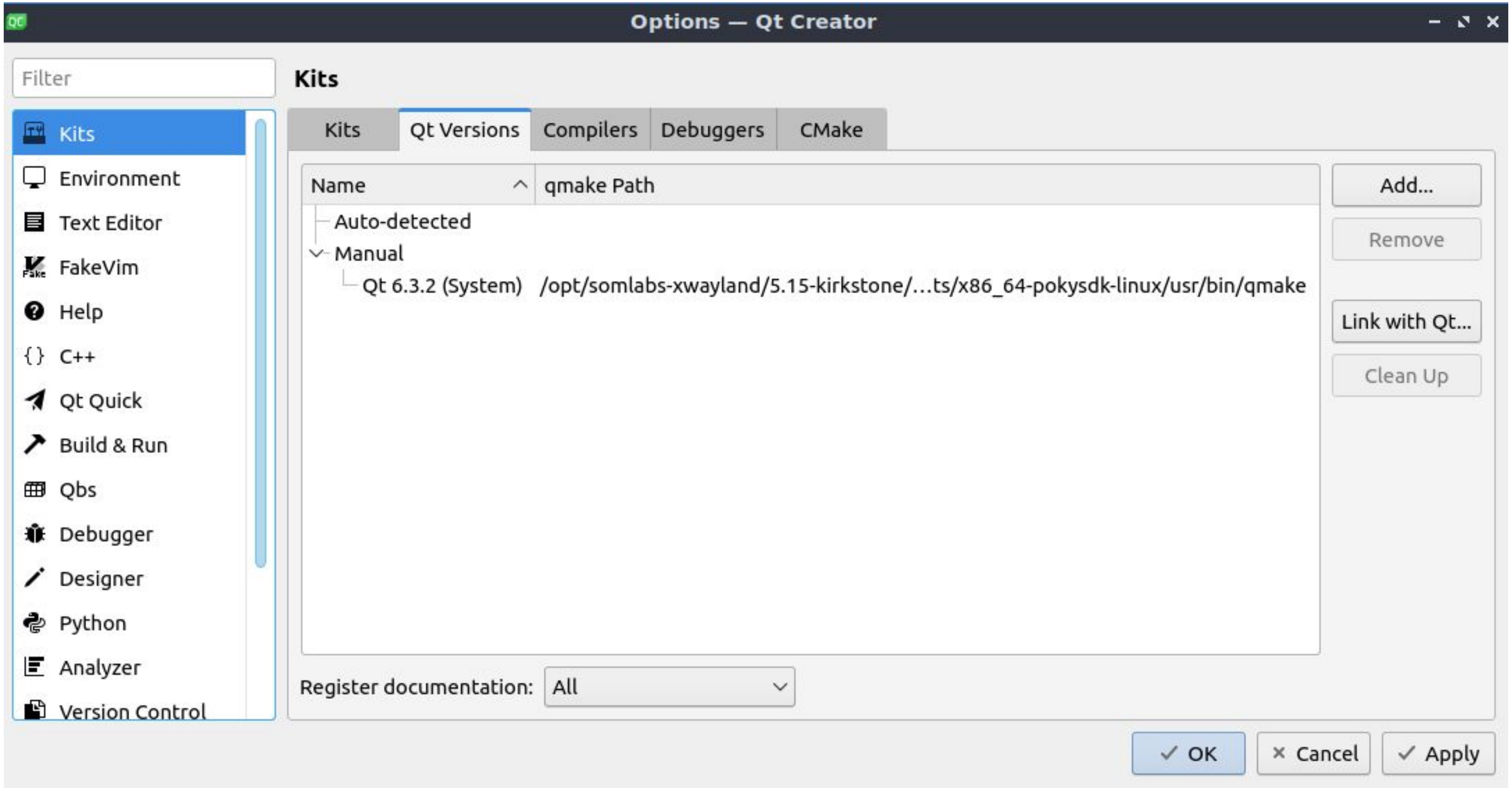


Device: SpaceSOM-8mplus (default for Generic Linux)

Name: SpaceSOM-8mplus
Type: Generic Linux
Auto-detected: No
Current state: Unknown

Machine type: Physical Device
Authentication type: ○ Default ● Specific key
Host name: 10.1.1.1 SSH port: 22 ☑ Check host key
Free ports: 10000-10100 Timeout: 10s
Username: root
Private key file:

Qt Creator

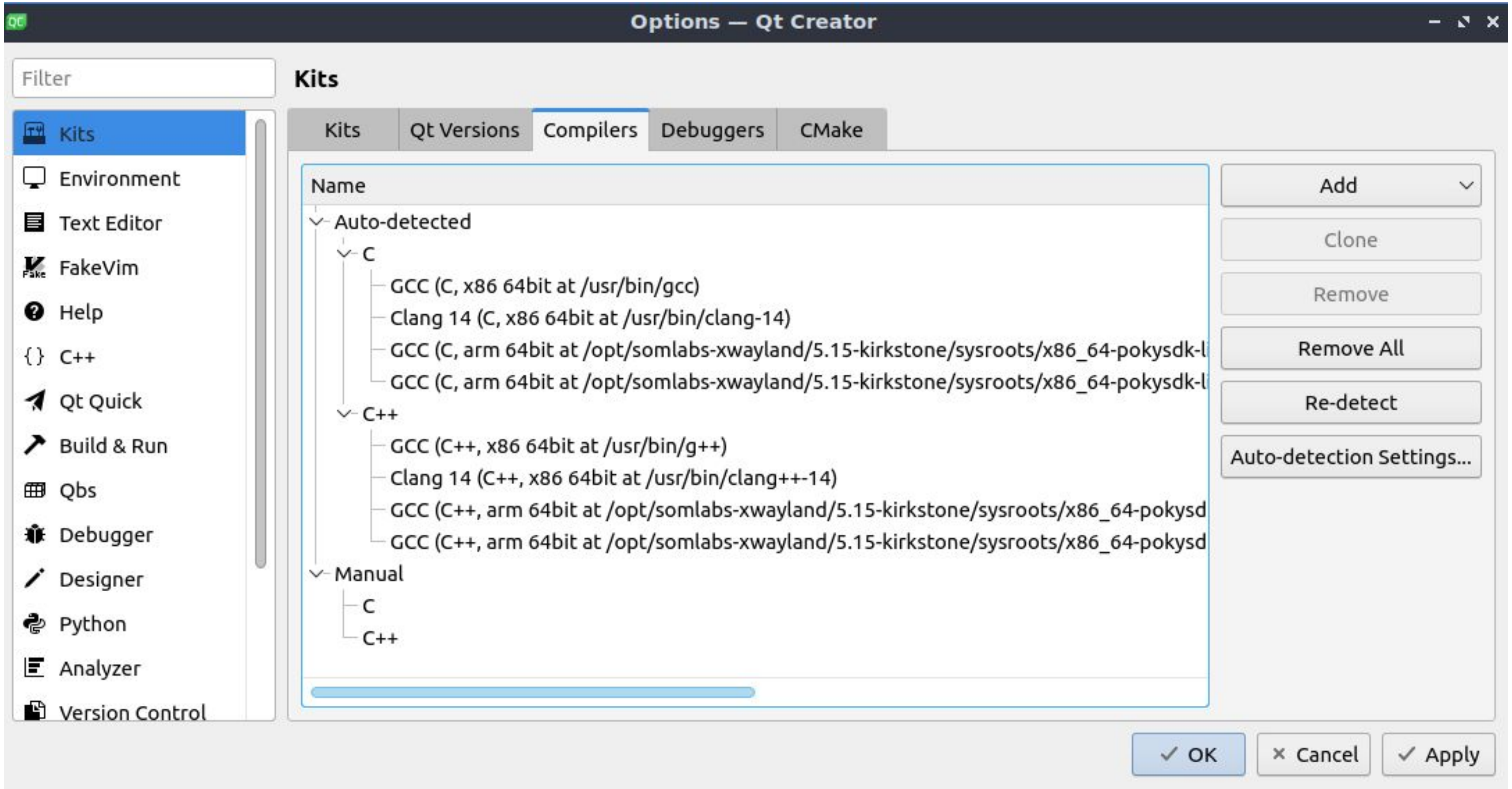


Name	qmake Path
Auto-detected	
Manual	
Qt 6.3.2 (System)	/opt/somlabs-xwayland/5.15-kirkstone/...ts/x86_64-pokysdk-linux/usr/bin/qmake

Register documentation: All

GOLD PARTNER

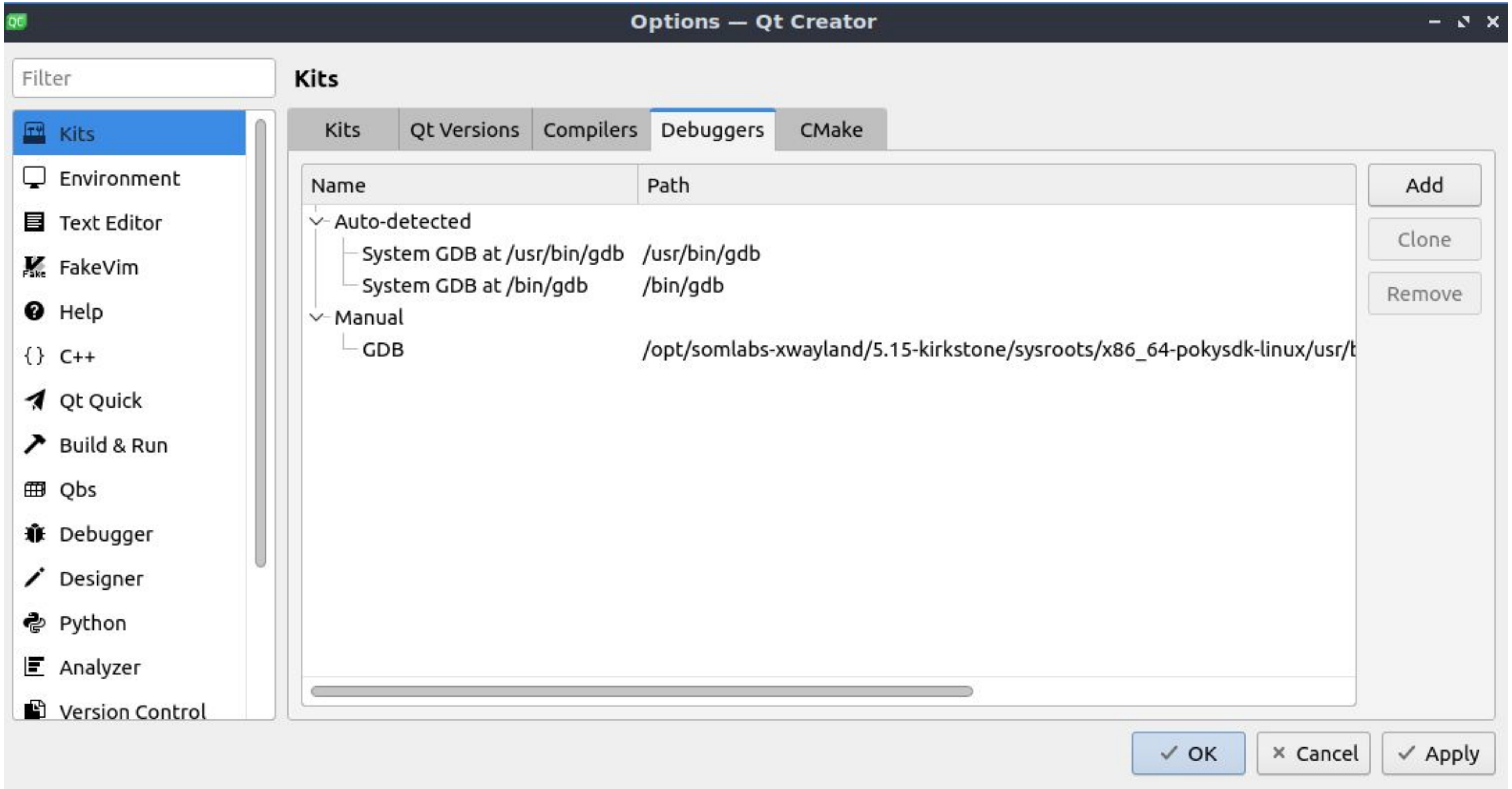
Qt Creator



Name

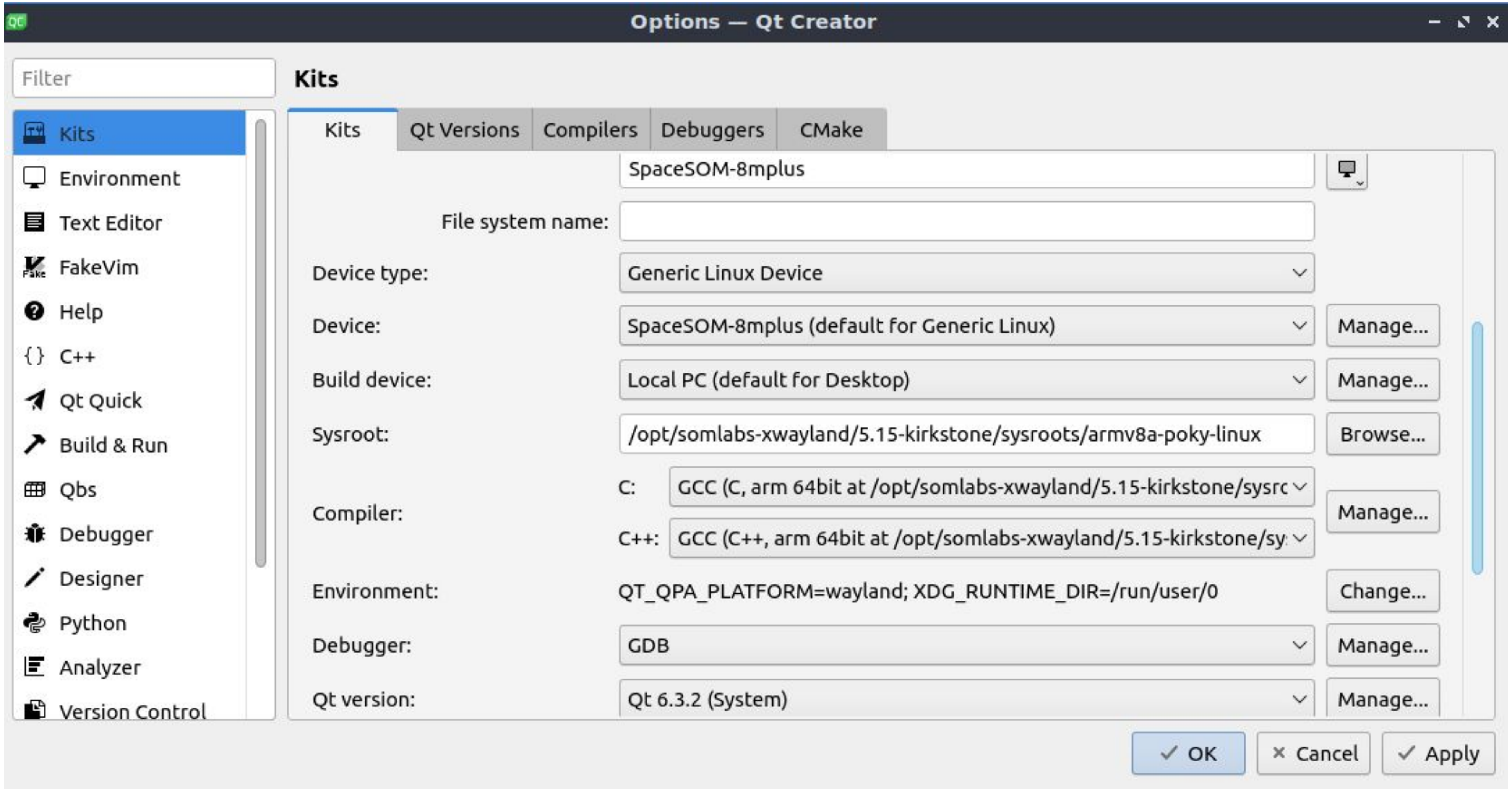
- Auto-detected
  - C
    - GCC (C, x86 64bit at /usr/bin/gcc)
    - Clang 14 (C, x86 64bit at /usr/bin/clang-14)
    - GCC (C, arm 64bit at /opt/somlabs-xwayland/5.15-kirkstone/sysroots/x86_64-pokysdk-l
    - GCC (C, arm 64bit at /opt/somlabs-xwayland/5.15-kirkstone/sysroots/x86_64-pokysdk-l
  - C++
    - GCC (C++, x86 64bit at /usr/bin/g++)
    - Clang 14 (C++, x86 64bit at /usr/bin/clang++-14)
    - GCC (C++, arm 64bit at /opt/somlabs-xwayland/5.15-kirkstone/sysroots/x86_64-pokysd
    - GCC (C++, arm 64bit at /opt/somlabs-xwayland/5.15-kirkstone/sysroots/x86_64-pokysd
- Manual
  - C
  - C++

Qt Creator



Name	Path
Auto-detected	
System GDB at /usr/bin/gdb	/usr/bin/gdb
System GDB at /bin/gdb	/bin/gdb
Manual	
GDB	/opt/somlabs-xwayland/5.15-kirkstone/sysroots/x86_64-pokysdk-linux/usr/

Qt Creator



Kits

SpaceSOM-8mplus

File system name:

Device type: Generic Linux Device

Device: SpaceSOM-8mplus (default for Generic Linux)

Build device: Local PC (default for Desktop)

Sysroot: /opt/somlabs-xwayland/5.15-kirkstone/sysroots/armv8a-poky-linux

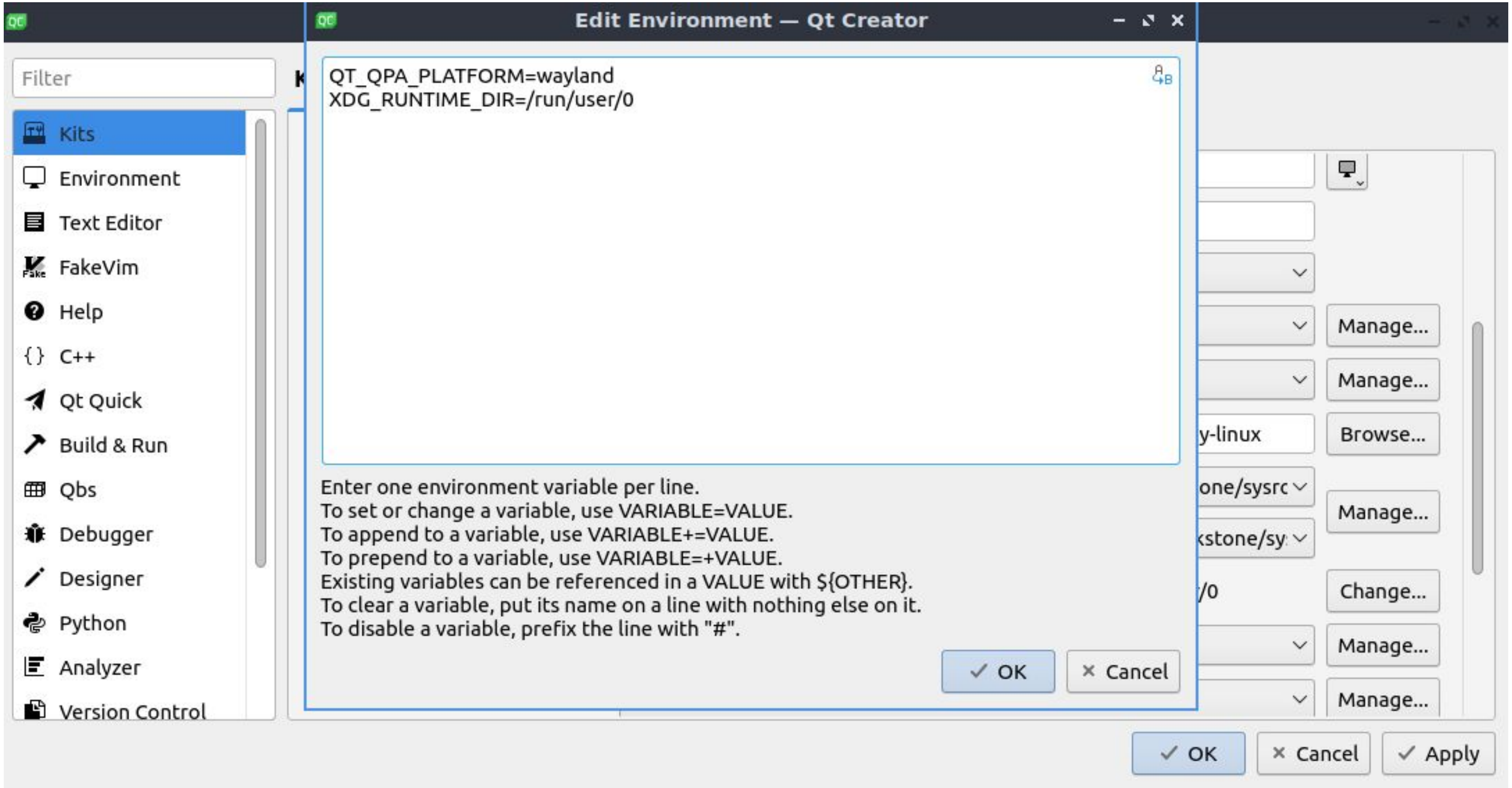
Compiler:
C: GCC (C, arm 64bit at /opt/somlabs-xwayland/5.15-kirkstone/sysrc
C++: GCC (C++, arm 64bit at /opt/somlabs-xwayland/5.15-kirkstone/sy

Environment: QT_QPA_PLATFORM=wayland; XDG_RUNTIME_DIR=/run/user/0

Debugger: GDB

Qt version: Qt 6.3.2 (System)

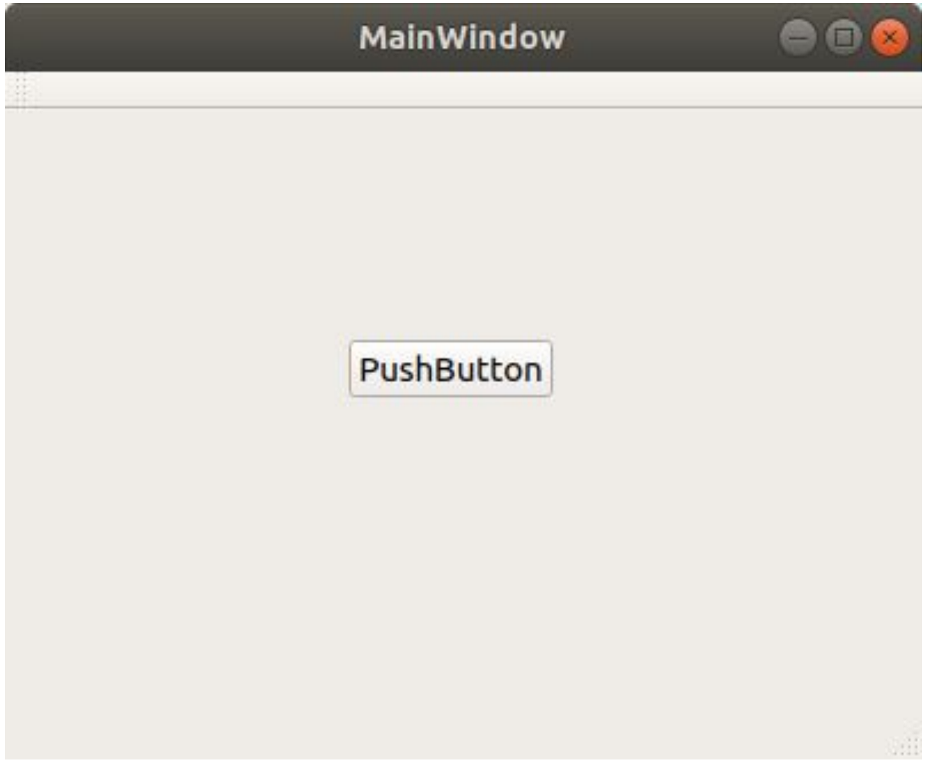
Qt Creator



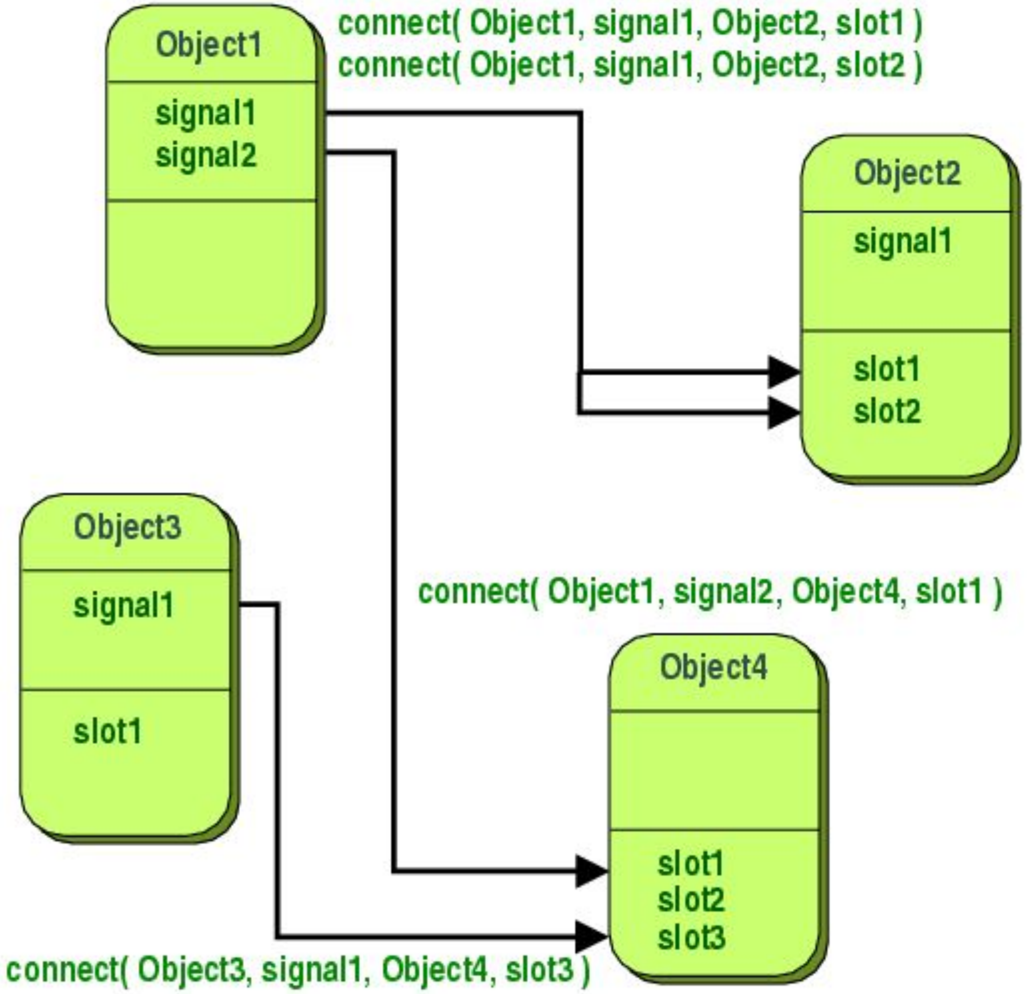
```
QT_QPA_PLATFORM=wayland
XDG_RUNTIME_DIR=/run/user/0
```

Enter one environment variable per line.
To set or change a variable, use VARIABLE=VALUE.
To append to a variable, use VARIABLE+=VALUE.
To prepend to a variable, use VARIABLE=+VALUE.
Existing variables can be referenced in a VALUE with ${OTHER}.
To clear a variable, put its name on a line with nothing else on it.
To disable a variable, prefix the line with "#".

Lab 3: building Qt widget application



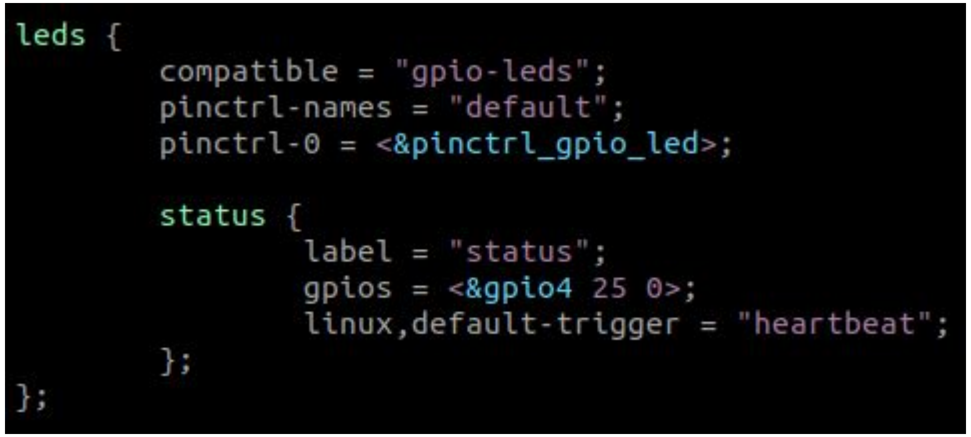
Qt Signals and Slots


Object1
signal1
signal2
connect(Object1, signal1, Object2, slot1)
connect(Object1, signal1, Object2, slot2)
Object2
signal1
slot1
slot2
Object3
signal1
slot1
connect(Object1, signal2, Object4, slot1)
Object4
slot1
slot2
slot3
connect(Object3, signal1, Object4, slot3)

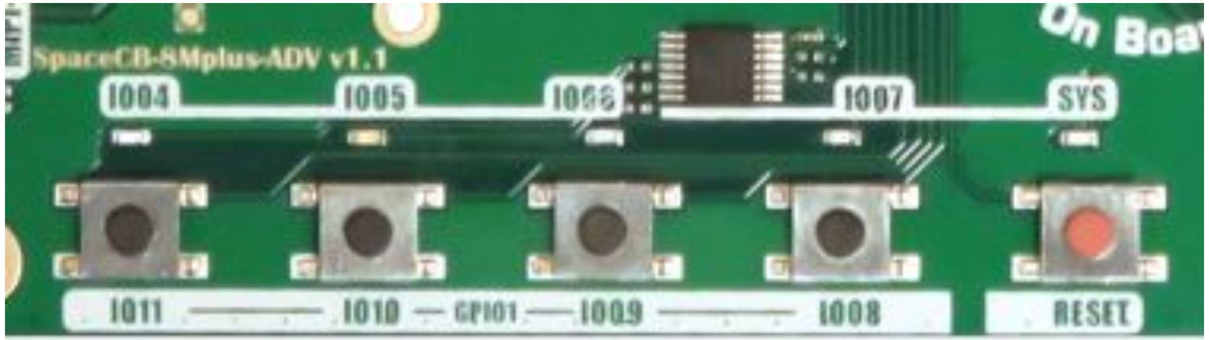
https://doc.qt.io/

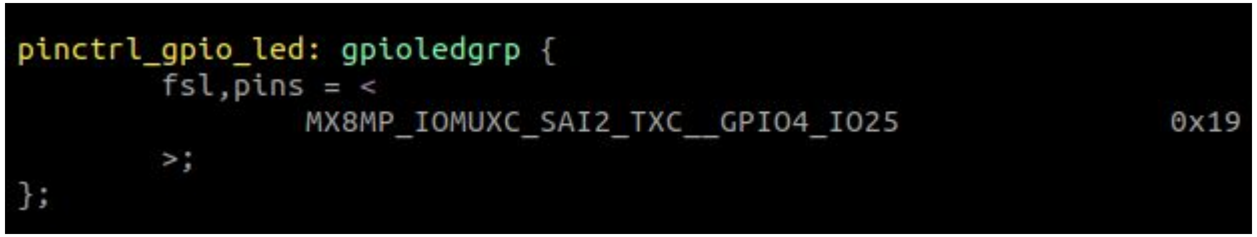
LED handling

- sysfs interface
    - /sys/class/leds/LED-IO-04/brightness



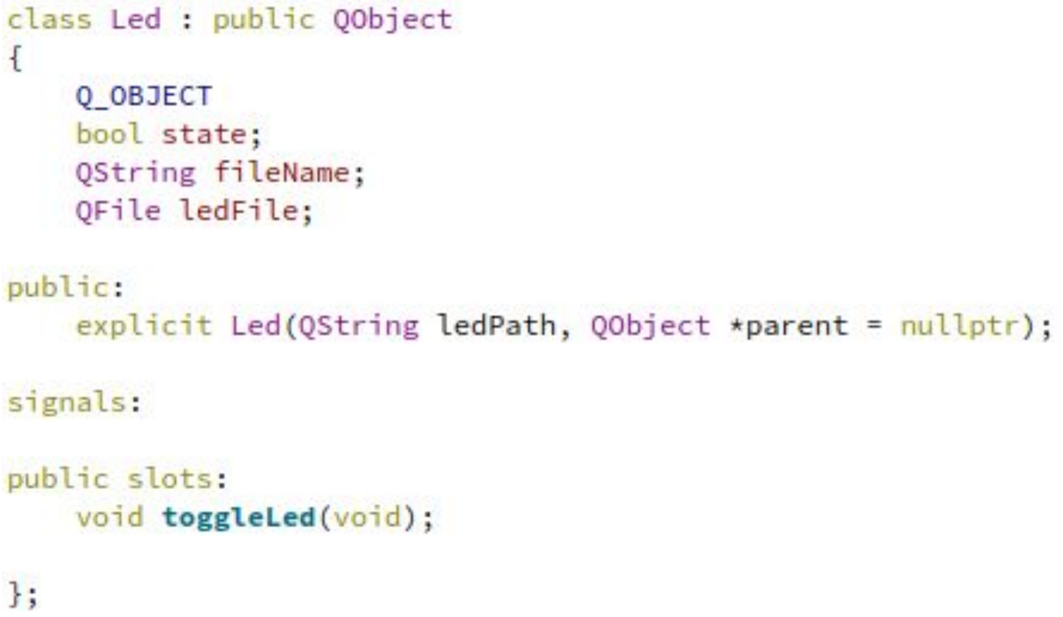
```
leds {
        compatible = "gpio-leds";
        pinctrl-names = "default";
        pinctrl-0 = <&pinctrl_gpio_led>;

        status {
                label = "status";
                gpios = <&gpio4 25 0>;
                linux,default-trigger = "heartbeat";
        };
};
```



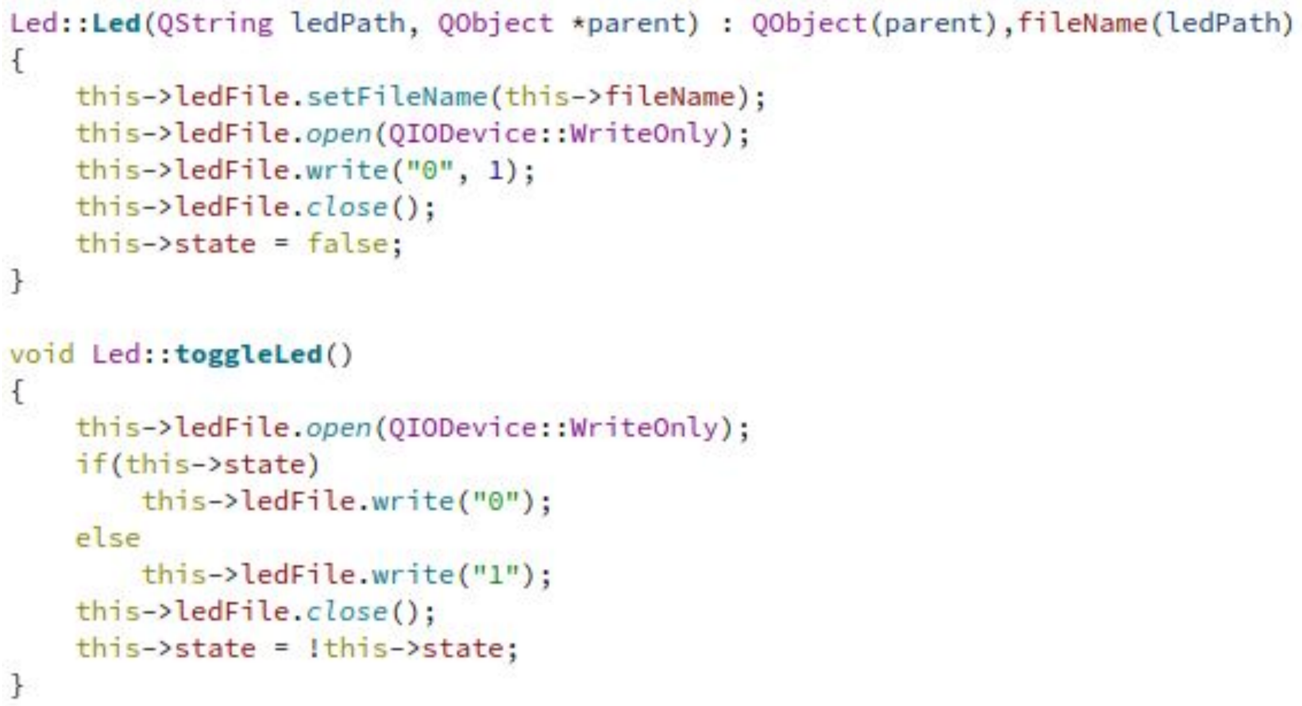


```
pinctrl_gpio_led: gpioledgrp {
        fsl,pins = <
                MX8MP_IOMUXC_SAI2_TXC__GPIO4_IO25               0x19
        >;
};
```

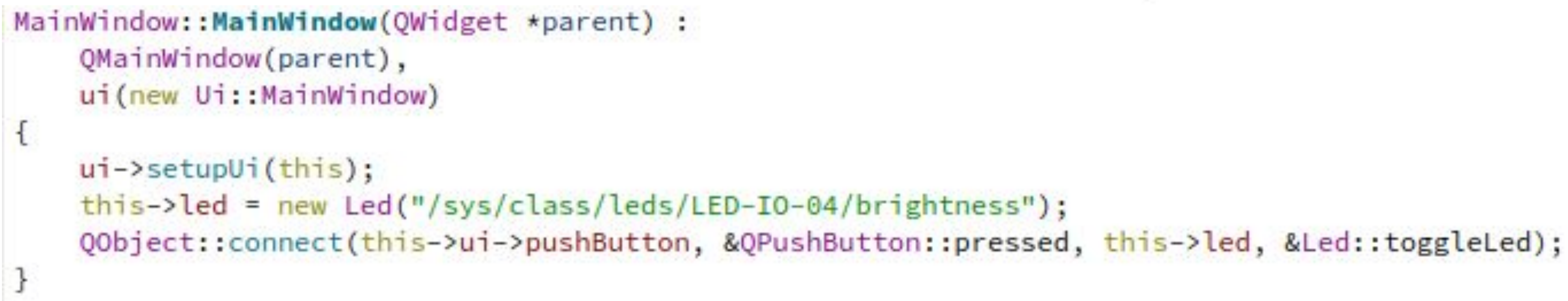
Lab 4: controlling LED from GUI



```
class Led : public QObject
{
    Q_OBJECT
    bool state;
    QString fileName;
    QFile ledFile;

public:
    explicit Led(QString ledPath, QObject *parent = nullptr);

signals:

public slots:
    void toggleLed(void);

};
```



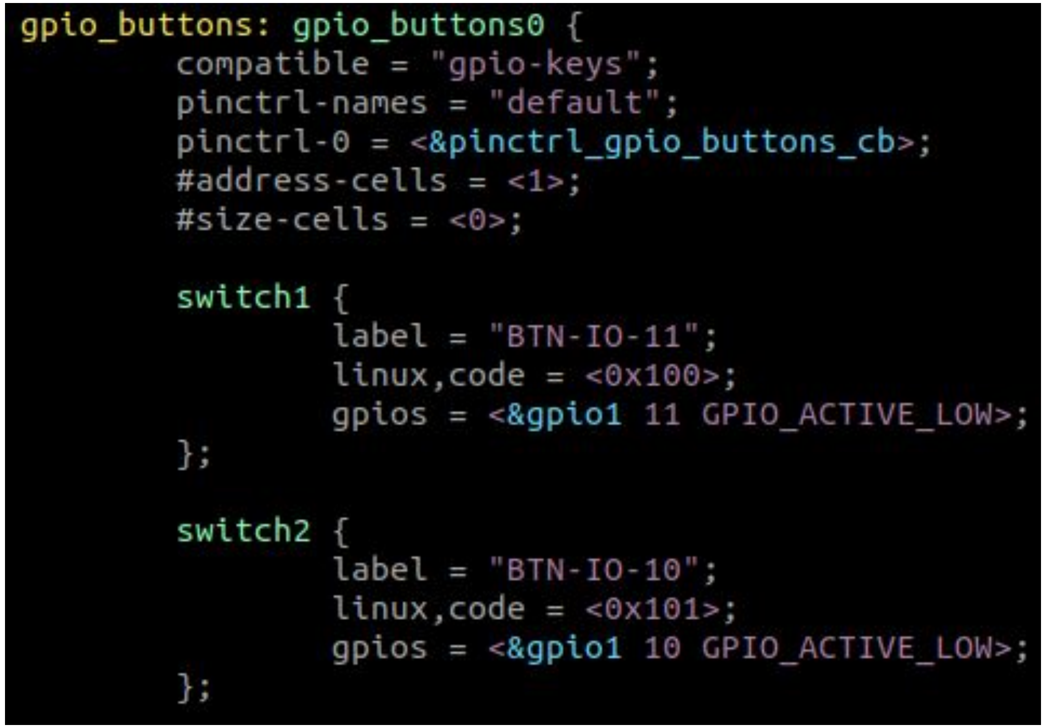
```
Led::Led(QString ledPath, QObject *parent) : QObject(parent),fileName(ledPath)
{
    this->ledFile.setFileName(this->fileName);
    this->ledFile.open(QIODevice::WriteOnly);
    this->ledFile.write("0", 1);
    this->ledFile.close();
    this->state = false;
}

void Led::toggleLed()
{
    this->ledFile.open(QIODevice::WriteOnly);
    if(this->state)
        this->ledFile.write("0");
    else
        this->ledFile.write("1");
    this->ledFile.close();
    this->state = !this->state;
}
```



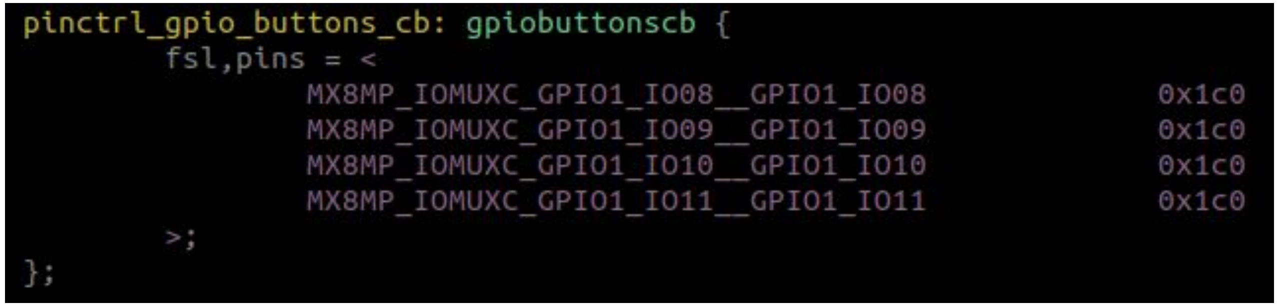
```
MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow)
{
    ui->setupUi(this);
    this->led = new Led("/sys/class/leds/LED-IO-04/brightness");
    QObject::connect(this->ui->pushButton, &QPushButton::pressed, this->led, &Led::toggleLed);
}
```

GPIO handling

- devfs interface
    - /dev/input/by-path/platform-gpio_buttons0-event

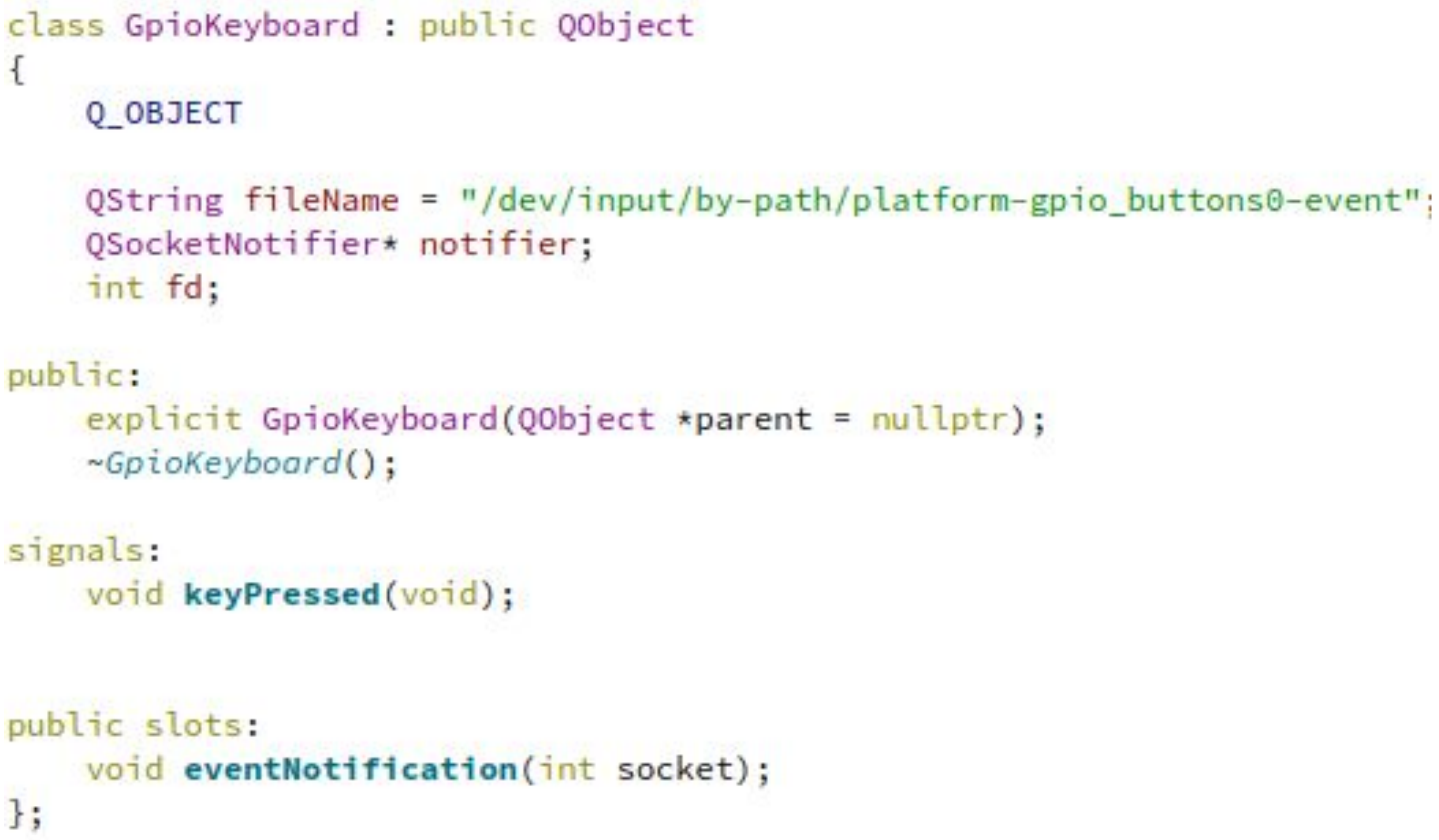


```
gpio_buttons: gpio_buttons0 {
        compatible = "gpio-keys";
        pinctrl-names = "default";
        pinctrl-0 = <&pinctrl_gpio_buttons_cb>;
        #address-cells = <1>;
        #size-cells = <0>;

        switch1 {
                label = "BTN-IO-11";
                linux,code = <0x100>;
                gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
        };

        switch2 {
                label = "BTN-IO-10";
                linux,code = <0x101>;
                gpios = <&gpio1 10 GPIO_ACTIVE_LOW>;
        };
```



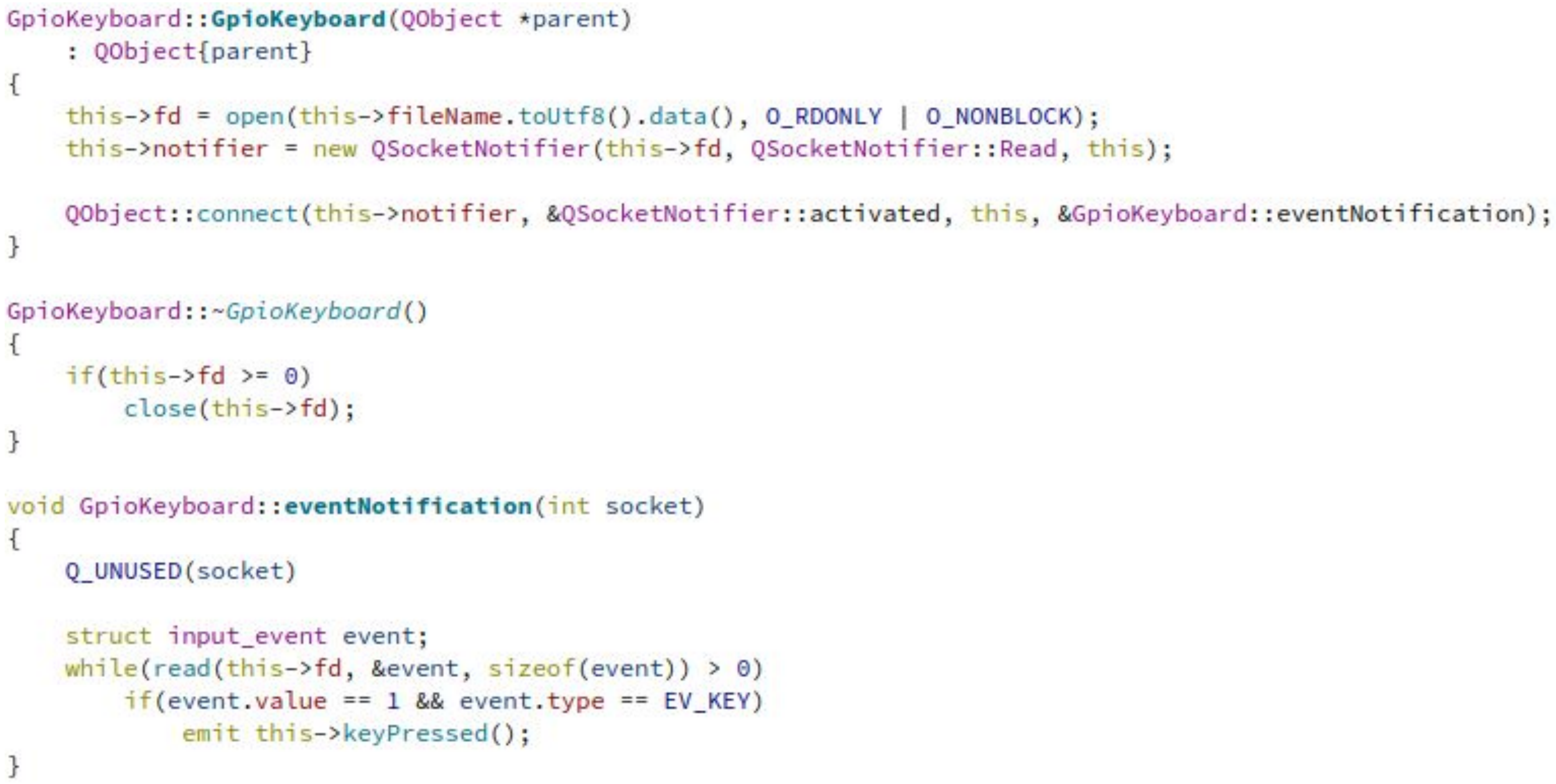
```
pinctrl_gpio_buttons_cb: gpiobuttonscb {
        fsl,pins = <
                MX8MP_IOMUXC_GPIO1_IO08__GPIO1_IO08             0x1c0
                MX8MP_IOMUXC_GPIO1_IO09__GPIO1_IO09             0x1c0
                MX8MP_IOMUXC_GPIO1_IO10__GPIO1_IO10             0x1c0
                MX8MP_IOMUXC_GPIO1_IO11__GPIO1_IO11             0x1c0
        >;
};
```

Lab 5: handling GPIO input



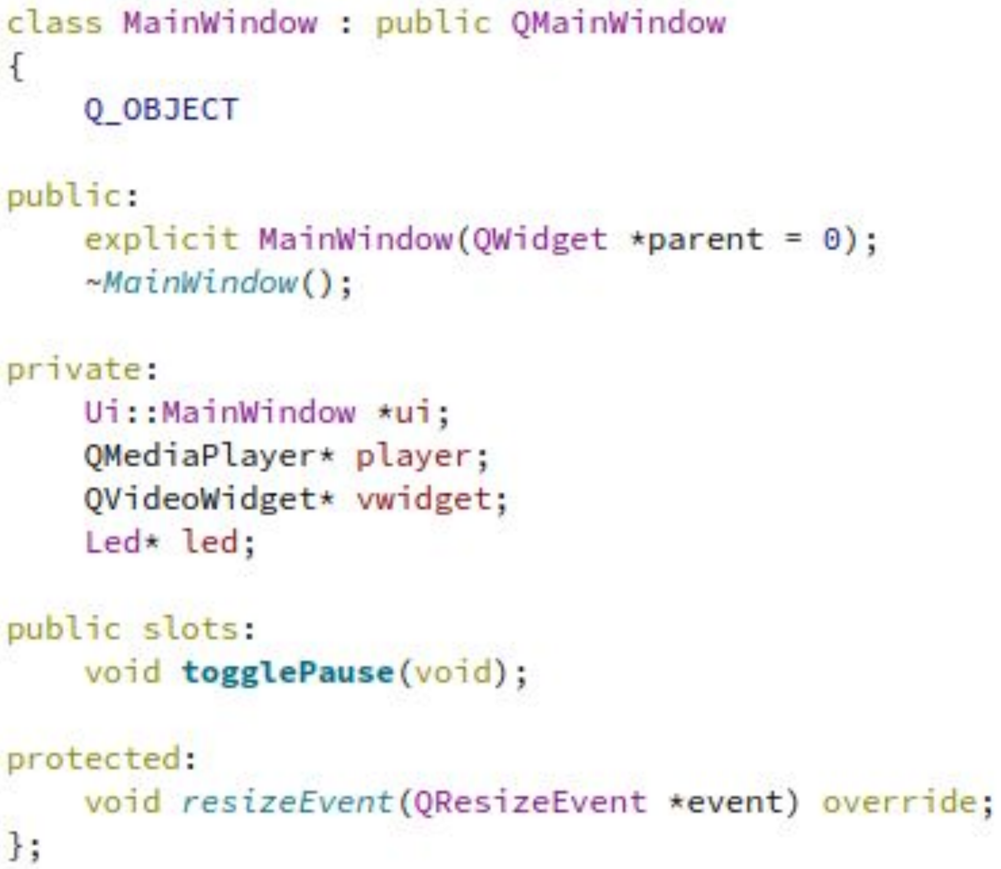
```
class GpioKeyboard : public QObject
{
    Q_OBJECT

    QString fileName = "/dev/input/by-path/platform-gpio_buttons0-event";
    QSocketNotifier* notifier;
    int fd;

public:
    explicit GpioKeyboard(QObject *parent = nullptr);
    ~GpioKeyboard();

signals:
    void keyPressed(void);


public slots:
    void eventNotification(int socket);
};
```

Lab 5: handling GPIO input



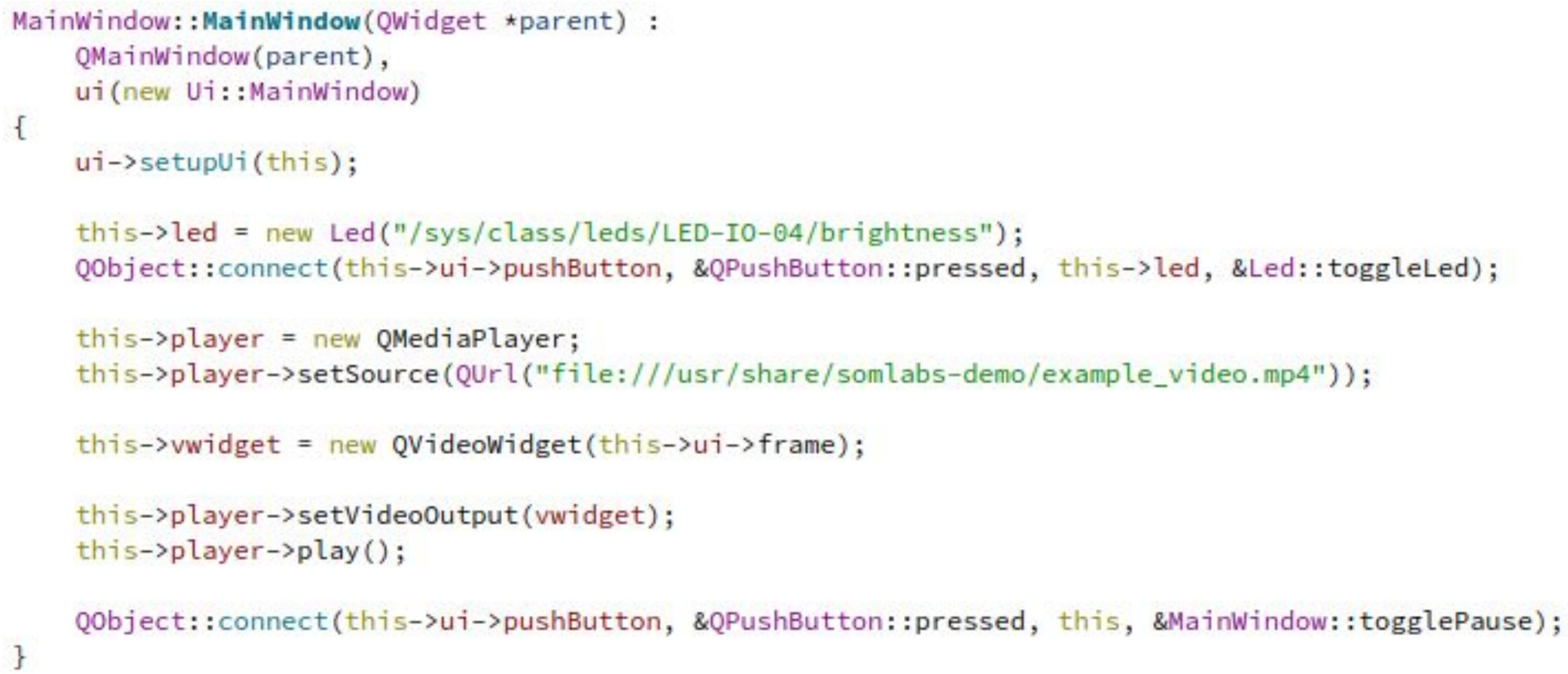
```
GpioKeyboard::GpioKeyboard(QObject *parent)
    : QObject{parent}
{
    this->fd = open(this->fileName.toUtf8().data(), O_RDONLY | O_NONBLOCK);
    this->notifier = new QSocketNotifier(this->fd, QSocketNotifier::Read, this);

    QObject::connect(this->notifier, &QSocketNotifier::activated, this, &GpioKeyboard::eventNotification);
}

GpioKeyboard::~GpioKeyboard()
{
    if(this->fd >= 0)
        close(this->fd);
}

void GpioKeyboard::eventNotification(int socket)
{
    Q_UNUSED(socket)

    struct input_event event;
    while(read(this->fd, &event, sizeof(event)) > 0)
        if(event.value == 1 && event.type == EV_KEY)
            emit this->keyPressed();
}
```

Lab 6: video playback



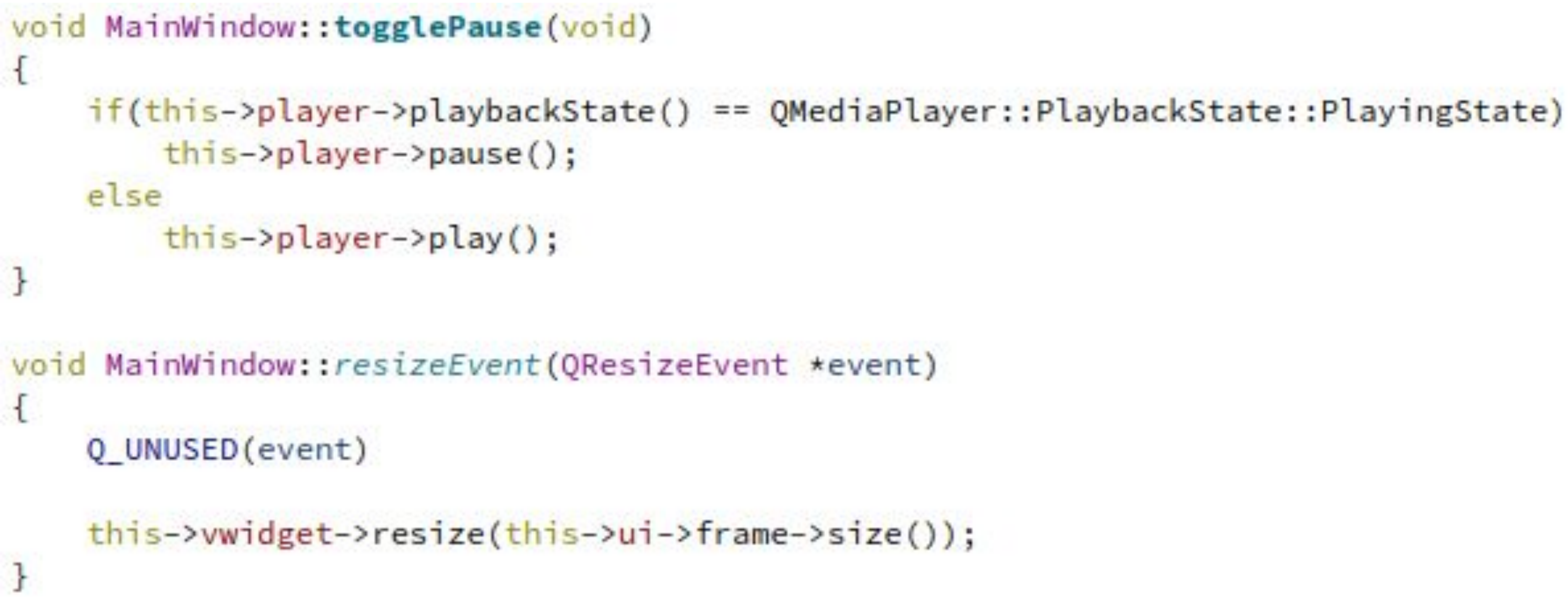
```
class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow(QWidget *parent = 0);
    ~MainWindow();

private:
    Ui::MainWindow *ui;
    QMediaPlayer* player;
    QVideoWidget* vwidget;
    Led* led;

public slots:
    void togglePause(void);

protected:
    void resizeEvent(QResizeEvent *event) override;
};
```

Lab 6: video playback



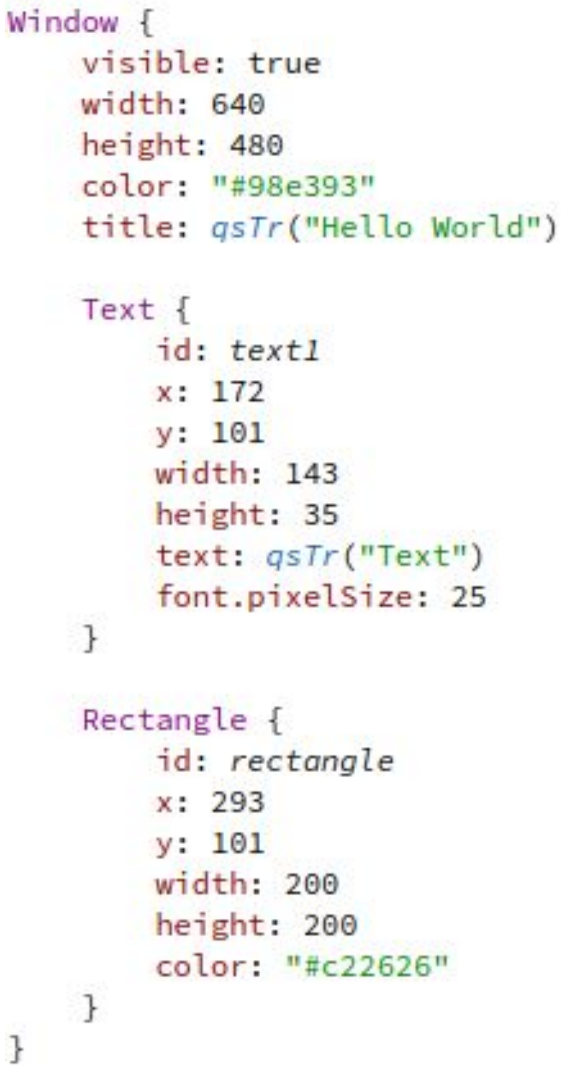
```
MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow)
{
    ui->setupUi(this);

    this->led = new Led("/sys/class/leds/LED-IO-04/brightness");
    QObject::connect(this->ui->pushButton, &QPushButton::pressed, this->led, &Led::toggleLed);

    this->player = new QMediaPlayer;
    this->player->setSource(QUrl("file:///usr/share/somlabs-demo/example_video.mp4"));

    this->vwidget = new QVideoWidget(this->ui->frame);

    this->player->setVideoOutput(vwidget);
    this->player->play();

    QObject::connect(this->ui->pushButton, &QPushButton::pressed, this, &MainWindow::togglePause);
}
```

Lab 6: video playback

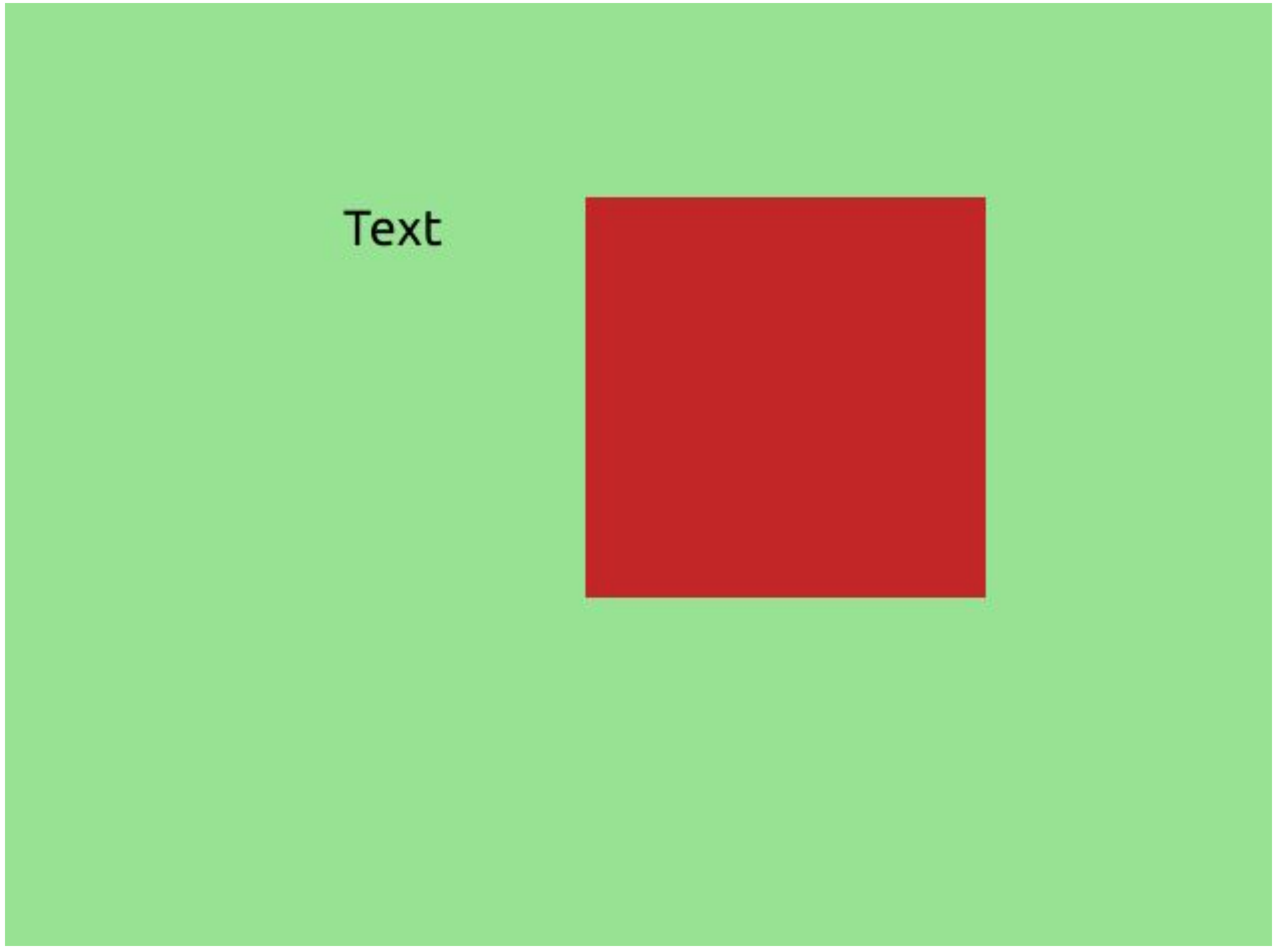


```
void MainWindow::togglePause(void)
{
    if(this->player->playbackState() == QMediaPlayer::PlaybackState::PlayingState)
        this->player->pause();
    else
        this->player->play();
}

void MainWindow::resizeEvent(QResizeEvent *event)
{
    Q_UNUSED(event)

    this->vwidget->resize(this->ui->frame->size());
}
```

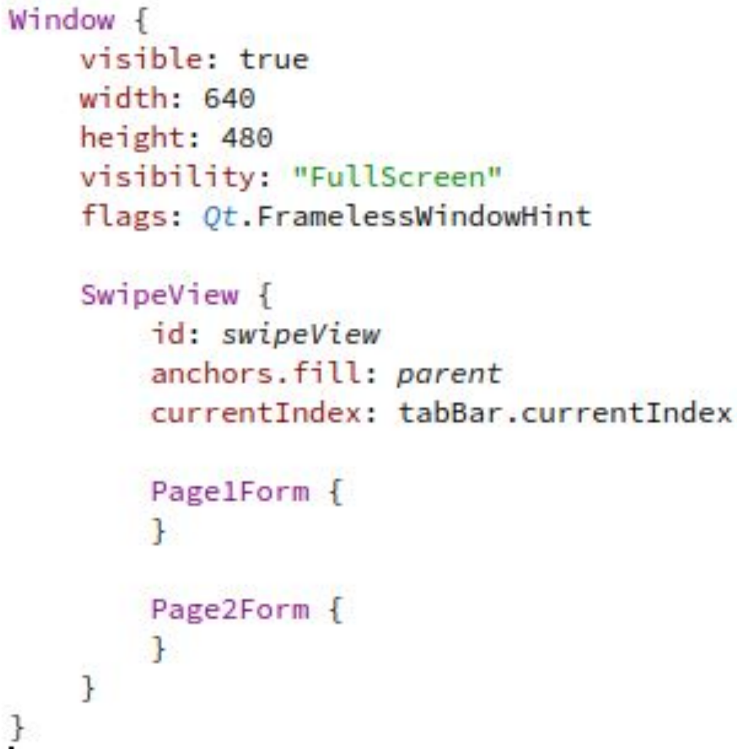
Qt QML



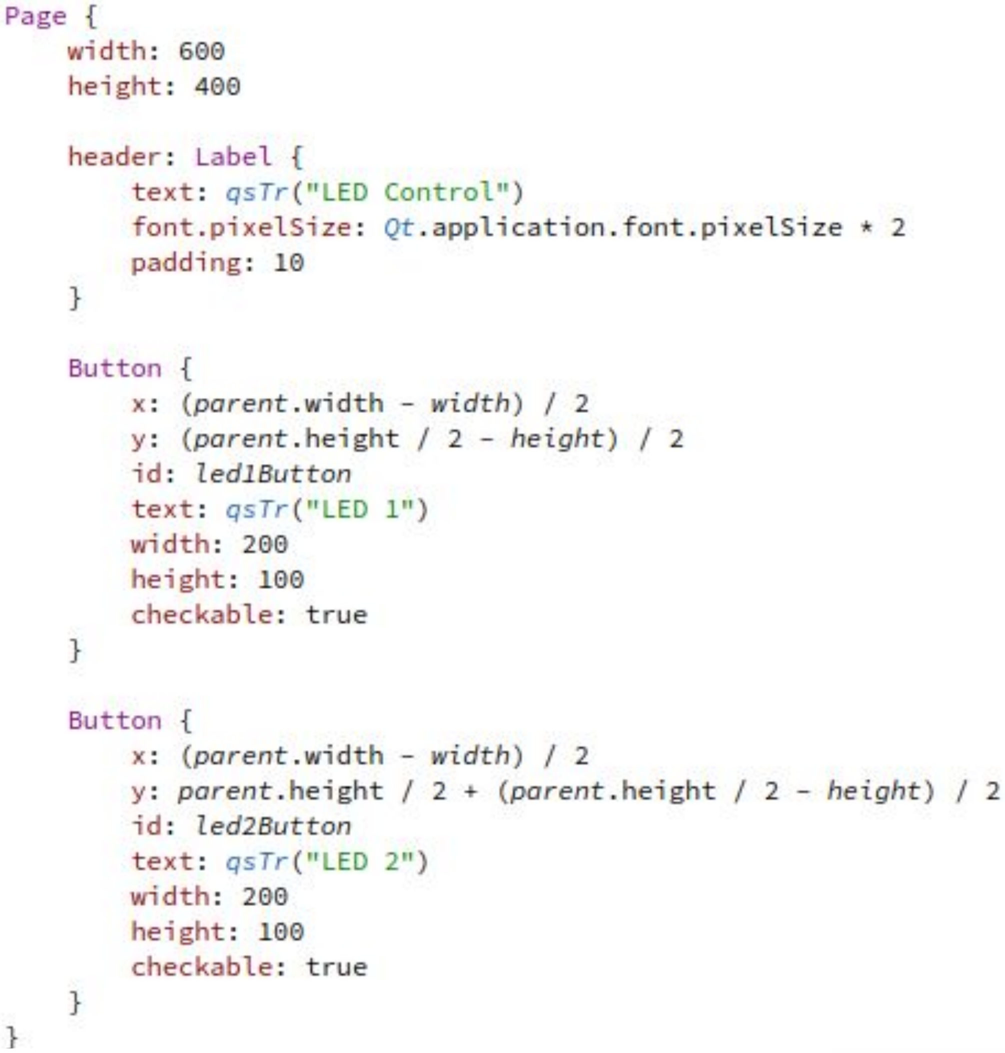
```
Window {
    visible: true
    width: 640
    height: 480
    color: "#98e393"
    title: qsTr("Hello World")

    Text {
        id: text1
        x: 172
        y: 101
        width: 143
        height: 35
        text: qsTr("Text")
        font.pixelSize: 25
    }

    Rectangle {
        id: rectangle
        x: 293
        y: 101
        width: 200
        height: 200
        color: "#c22626"
    }
}
```



Lab 7: building QML application

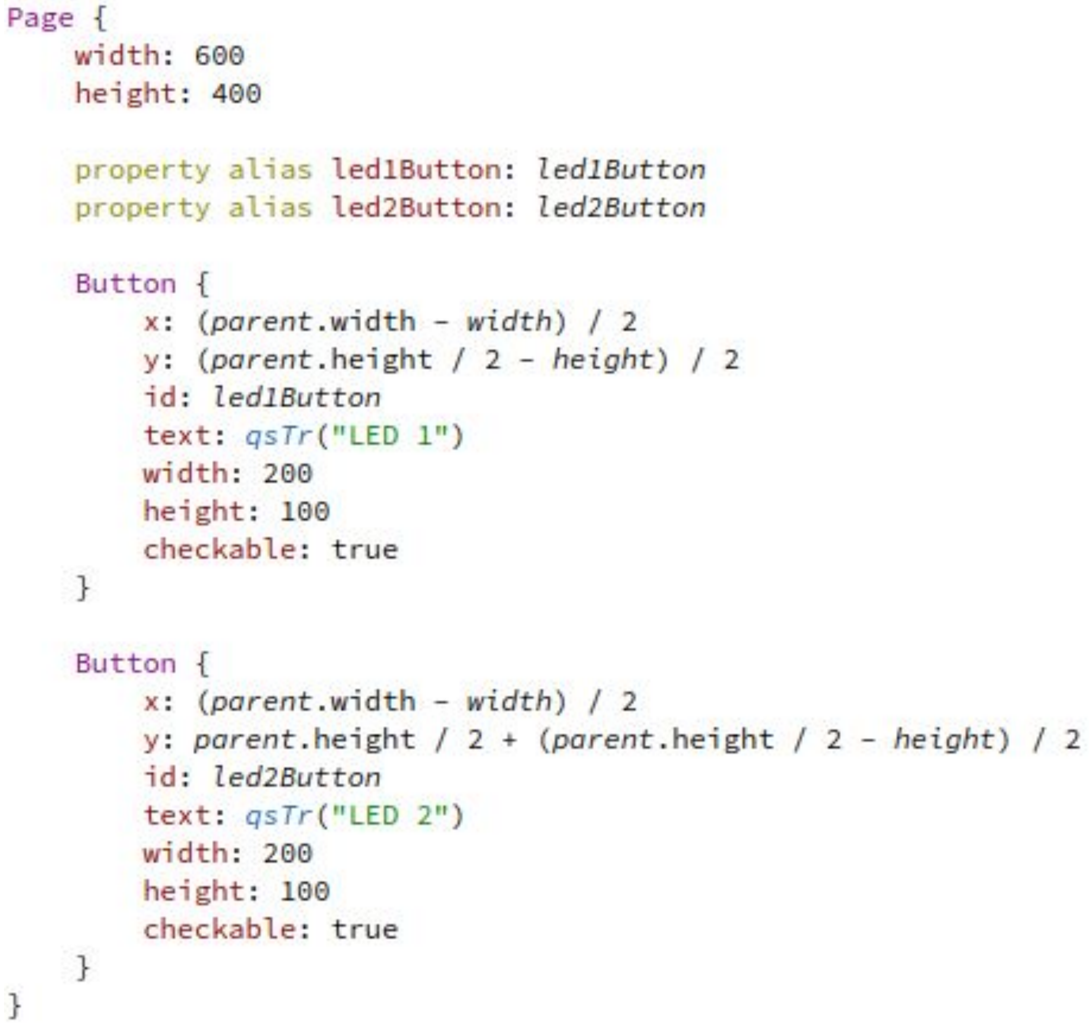


```
Window {
    visible: true
    width: 640
    height: 480
    visibility: "FullScreen"
    flags: Qt.FramelessWindowHint

    SwipeView {
        id: swipeView
        anchors.fill: parent
        currentIndex: tabBar.currentIndex

        Page1Form {
        }

        Page2Form {
        }
    }
}
```

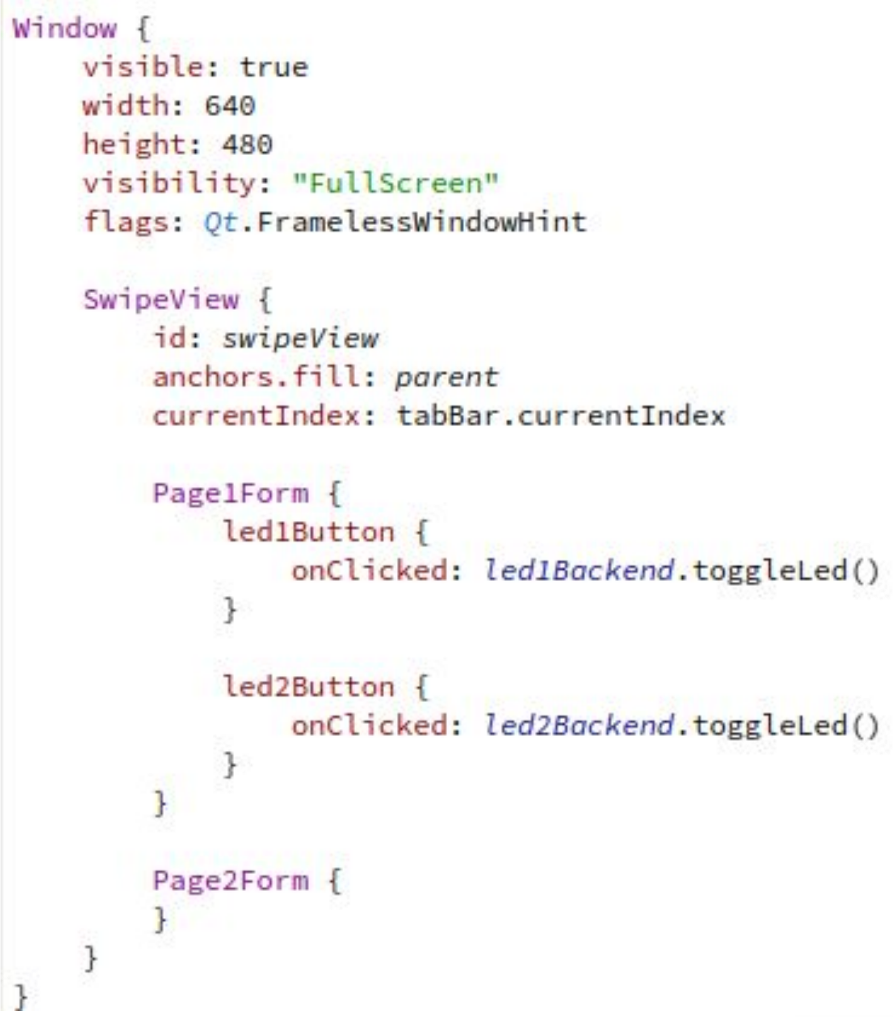


```
Page {
    width: 600
    height: 400

    header: Label {
        text: qsTr("LED Control")
        font.pixelSize: Qt.application.font.pixelSize * 2
        padding: 10
    }

    Button {
        x: (parent.width - width) / 2
        y: (parent.height / 2 - height) / 2
        id: led1Button
        text: qsTr("LED 1")
        width: 200
        height: 100
        checkable: true
    }

    Button {
        x: (parent.width - width) / 2
        y: parent.height / 2 + (parent.height / 2 - height) / 2
        id: led2Button
        text: qsTr("LED 2")
        width: 200
        height: 100
        checkable: true
    }
}
```

Lab 8: controlling LED from QML GUI

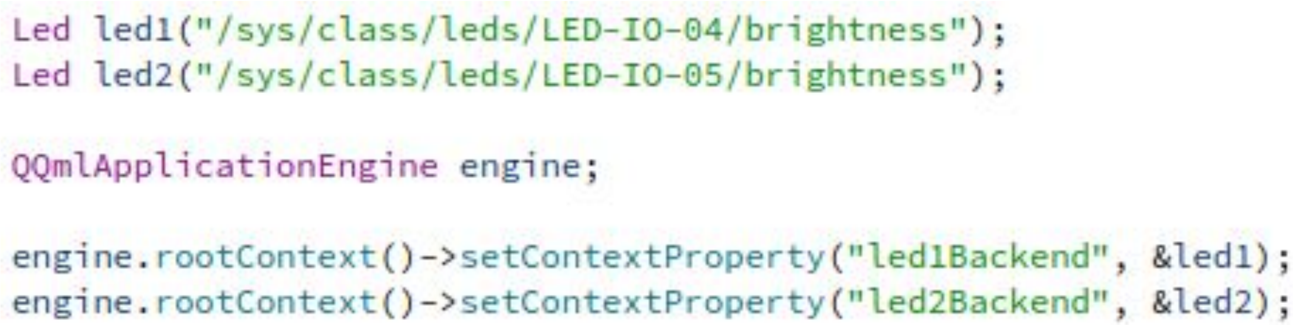


```
Page {
    width: 600
    height: 400

    property alias led1Button: led1Button
    property alias led2Button: led2Button

    Button {
        x: (parent.width - width) / 2
        y: (parent.height / 2 - height) / 2
        id: led1Button
        text: qsTr("LED 1")
        width: 200
        height: 100
        checkable: true
    }

    Button {
        x: (parent.width - width) / 2
        y: parent.height / 2 + (parent.height / 2 - height) / 2
        id: led2Button
        text: qsTr("LED 2")
        width: 200
        height: 100
        checkable: true
    }
}
```



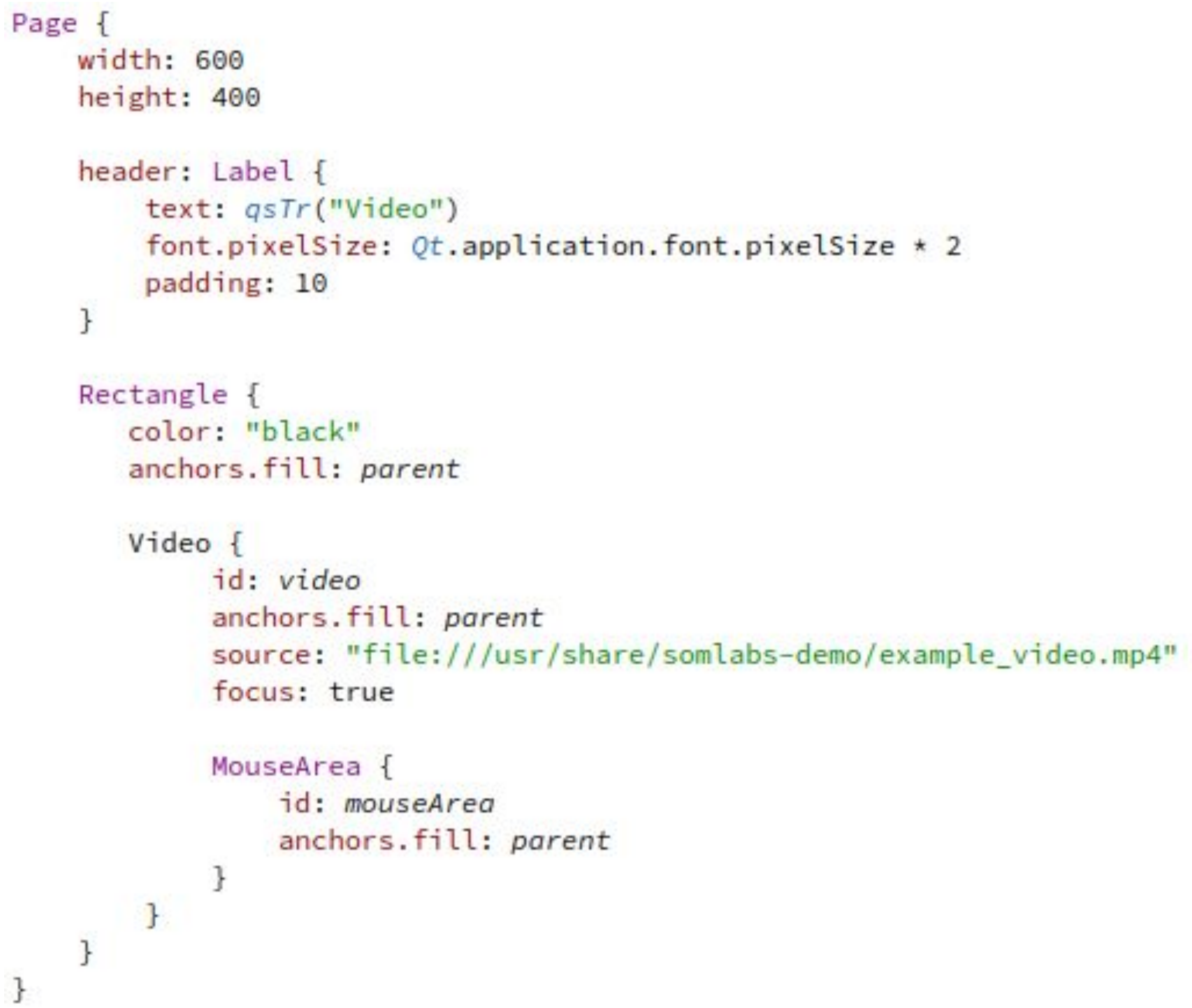
```
Window {
    visible: true
    width: 640
    height: 480
    visibility: "FullScreen"
    flags: Qt.FramelessWindowHint

    SwipeView {
        id: swipeView
        anchors.fill: parent
        currentIndex: tabBar.currentIndex

        Page1Form {
            led1Button {
                onClicked: led1Backend.toggleLed()
            }

            led2Button {
                onClicked: led2Backend.toggleLed()
            }
        }

        Page2Form {
        }
    }
}
```

Lab 8: controlling LED from QML GUI

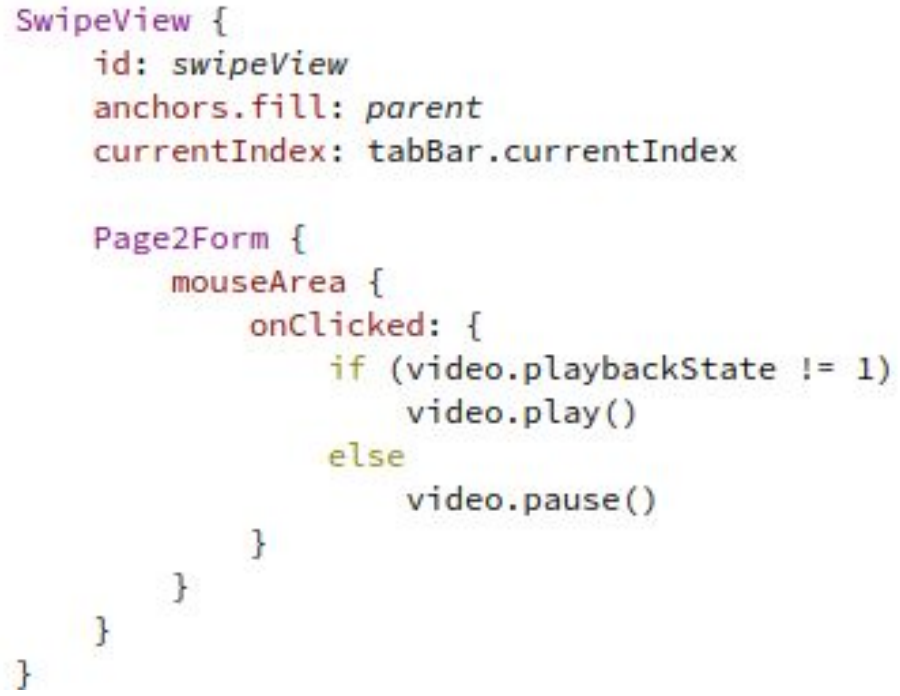


```
Led led1("/sys/class/leds/LED-IO-04/brightness");
Led led2("/sys/class/leds/LED-IO-05/brightness");

QQmlApplicationEngine engine;

engine.rootContext()->setContextProperty("led1Backend", &led1);
engine.rootContext()->setContextProperty("led2Backend", &led2);
```

Lab 9: video playback in QML



```
Page {
    width: 600
    height: 400

    header: Label {
        text: qsTr("Video")
        font.pixelSize: Qt.application.font.pixelSize * 2
        padding: 10
    }

    Rectangle {
       color: "black"
       anchors.fill: parent

       Video {
            id: video
            anchors.fill: parent
            source: "file:///usr/share/somlabs-demo/example_video.mp4"
            focus: true

            MouseArea {
                id: mouseArea
                anchors.fill: parent
            }
        }
    }
}
```



```
SwipeView {
    id: swipeView
    anchors.fill: parent
    currentIndex: tabBar.currentIndex

    Page2Form {
        mouseArea {
            onClicked: {
                if (video.playbackState != 1)
                    video.play()
                else
                    video.pause()
            }
        }
    }
}
```

Lab 10: creating new Yocto recipe

```
DESCRIPTION = "SoMLabs Qt demo application"
LICENSE = "BSD-3-Clause"
LIC_FILES_CHKSUM = "file://${COREBASE}/meta/files/common-licenses/BSD-3-Clause;md5=550794465ba0ec5312d6919e203a55f9"

DEPENDS += "qtbase"
DEPENDS += "qtmultimedia"

SRC_URI = " file://src/ "

inherit qt6-qmake

S = "${WORKDIR}/src"

do_install() {
    install -d ${D}/usr/share/somlabs-demo-qt/
    install -m 0755 qml ${D}/usr/share/somlabs-demo-qt/
}

FILES:${PN} = " /usr/share/somlabs-demo-qt/ "
```

SoMLabs

- Connecting the Future
- Customer Centric