

Sunshine-A20  
Sunshine-A20-Lite  
BSP&Buildroot A20

*Руководство пользователя для изделий на основе  
Allwinner A20*

## Sunshine-A20

---

- Allwinner A20: CPU Dual-Core ARM Cortex-A7 с частотой работы ядер 1ГГц (видеопроцессор Mali400)
- DDR3 1Гбайт
- NAND 4Гбайт
- 1G Ethernet
- 2xUSB HS Host
- HDMI
- Аудио выход на наушники
- I/O: USB-OTG, LCD, LVDS, SATA, SD/MMC, SPDIF,CSI, UART, SPI, I2C, CAN, GPIO ...
- Возможность консольного подключения через EV-FT230 (USB-COM порт)
- Габариты: 109x79x16мм
- Посадочное место 2,5" HDD
- Питание: 5В/1А, «+» - центральная жила разъема, БП обязательно должен быть со стабилизированным выходом!

Отличие Sunshine-A20 от Sunshine-A20-Lite заключается в большем объеме оперативной памяти (DDR3), наличии NAND flash памяти и устанавливаемых разъемов расширения и SATA.

**Внимание!!!** Загрузочные образы плат не совместимы из-за различий в настройках памяти.

## Sunshine-A20-Lite

---

- Allwinner A20: CPU Dual-Core ARM Cortex-A7 с частотой работы ядер 1ГГц (видеопроцессор Mali400)
- DDR3 512Мбайт
- 1G Ethernet
- 2xUSB HS Host
- HDMI
- Аудио выход на наушники
- I/O: USB-OTG, LCD, LVDS, SATA, SD/MMC, SPDIF,CSI, UART, SPI, I2C, CAN, GPIO ...
- Возможность консольного подключения через EV-FT230 (USB-COM порт)
- Габариты: 109x79x16мм
- Возможность крепления 2,5" HDD
- Питание: 5В/1А, «+» - центральная жила разъема, БП обязательно должен быть со стабилизированным выходом!

## Первое включение

Предварительно необходимо подготовить загрузочную карту, собрать систему и подготовить карту согласно дальнейшим разделам данного описания, либо, можно записать готовый образ имеющийся среди публикуемых материалов к изделиям.

Подключите USB клавиатуру, Ethernet и HDMI кабель, так же можно подключить EV-FT230 к разъему X4 (DUART) – консольный доступ к системе. Включить питание, после успешной загрузки системы появится приглашение авторизации (на HDMI мониторе).

```
Welcome to Buildroot
```

```
buildroot login:
```

Для получения доступа введите, логин: root, пароль: root

В системе настроены сервисы FTP и SSH, IP адрес платы 192.168.0.136.

Для удобства операций над файловой системой, установлен MidnightCommander, введите mc для его запуска, примеры Qt расположены в папке /usr/lib/qt/examples

## Основные положения

Виртуальная  
машина  
запускается как  
приложение  
Windows

Основана на  
Libuntu 14.10

Buildroot – основа  
для сборки ядра и  
КФС

Виртуальная машина предназначена для сборки ядра Linux и корневой файловой системы (КФС) для изделий starterkit.ru на основе процессора Allwinner A20.

- **Sunshine-A20**
- **Sunshine-A20-Lite**

Виртуальная машина основана на Libuntu 14.10.

Сборка ядра Linux и КФС интегрирована и осуществляется в одном пакете **Buildroot**. **Buildroot** ([www.buildroot.org](http://www.buildroot.org)) это инструмент для сборки КФС, содержит более тысячи приложений и библиотек, состав выбирает сам пользователь.

Для удобства, в виртуальной машине установлены и настроены сервисы для взаимодействия с внешним окружением

- **FTP сервер**
- **TFTP сервер**
- **SSH сервер**
- **Samba** – взаимодействие с сетями Microsoft
- **QtCreator**

**QtCreator** – среда разработки и отладки Qt приложений, имея Ethernet соединение (TCP/IP канал связи можно организовать практически через любой интерфейс процессора) с платой можно одним щелчком мыши запускать приложение на плате или вести пошаговую отладку.

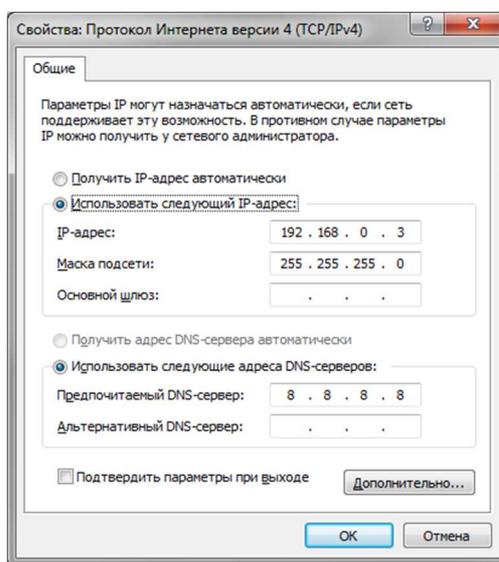
**Qt** наиболее предпочтительный пакет для написания приложений, т.к. в нем существует огромное количество библиотек и примеров, имеет широкую распространенность, графические приложения Qt работают значительно быстрее и потребляют гораздо меньше ресурсов (особенно памяти) системы в сравнении с приложениями X11.

## Настройка виртуальной машины

Перед началом работы необходимо скачать плеер виртуальной машины VMware, бесплатно распространяемый на сайте [www.vmware.com](http://www.vmware.com).

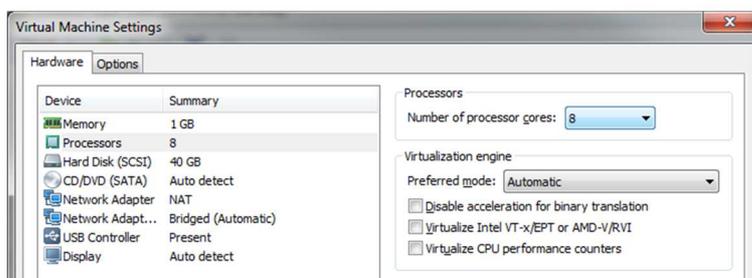
Виртуальная машина имеет 2 сетевых интерфейса:

- 1) NAT – для доступа в Internet
- 2) Bridget – для взаимодействия по локальной сети, необходимо настроить VMware network adapter



Так же необходимо настроить сетевой адаптер PC (или DHCP роутера), так, чтобы присваивался сетевой адрес в группе 192.168.0.XXX (любой кроме 1-3 и 136).

Перед запуском виртуальной машины рекомендуем зайти в ее свойства и выделить количество используемых процессорных ядер (по умолчанию, указан 1), это позволит ощутимо сократить время сборки.



Скачать плеер  
VMware

Настроить  
виртуальный  
сетевой адаптер

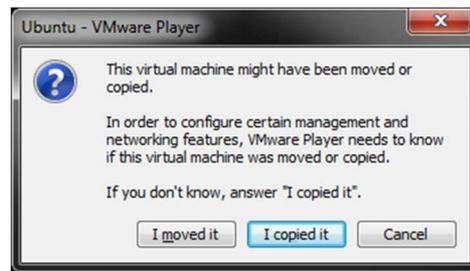
Настроить  
сетевой адаптер

Настроить  
количество  
используемых  
ядер процессора

## Первый запуск виртуальной машины

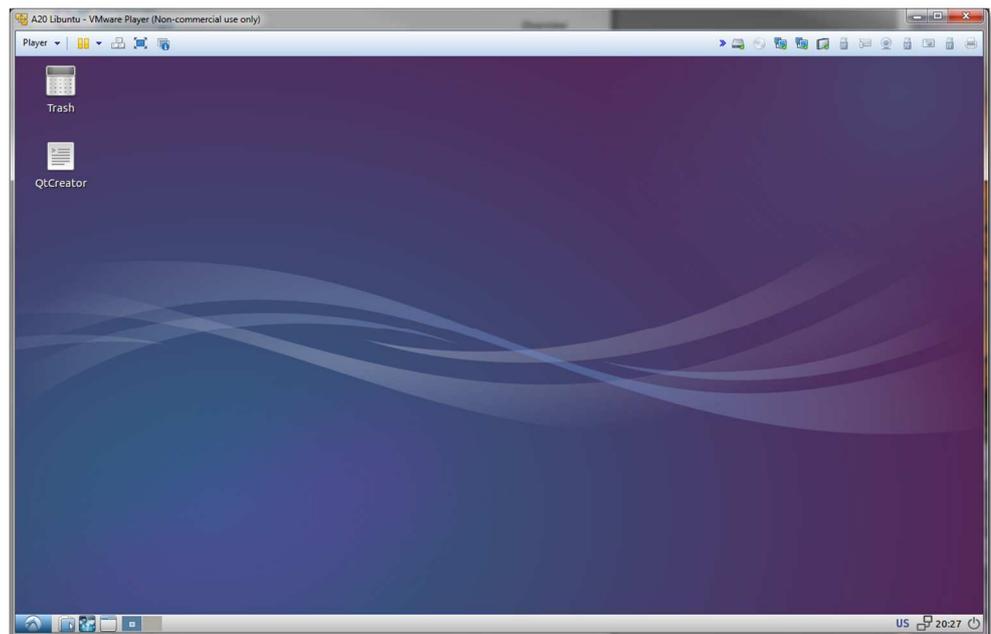
Отвечать:  
"I moved it"

При первом запуске виртуальной машины (или после копирования-перемещения), VMware Player спросит:



Нужно ответить «I moved it», это позволит сохранить сетевые настройки.

После загрузки перед вами появится рабочий стол.



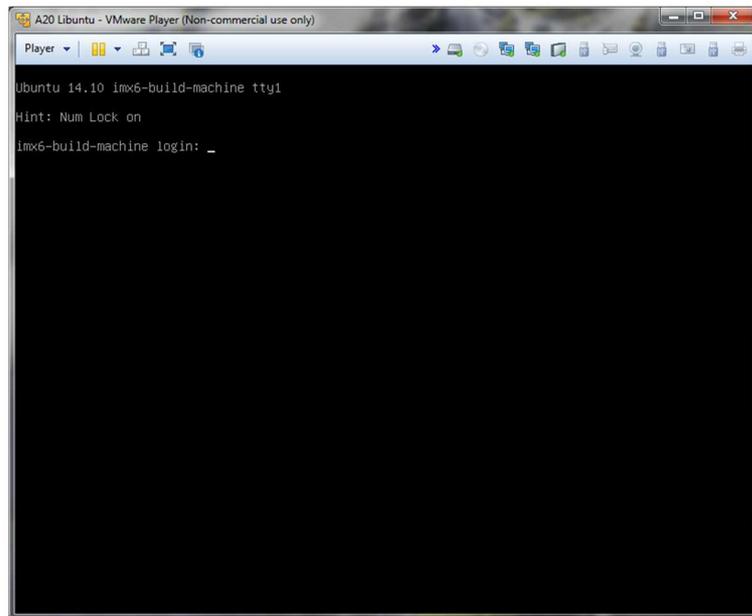
Разрешение экрана можно изменить в разделе **Perfences->Monitor Settings**

Login: user  
Pasw: 123456

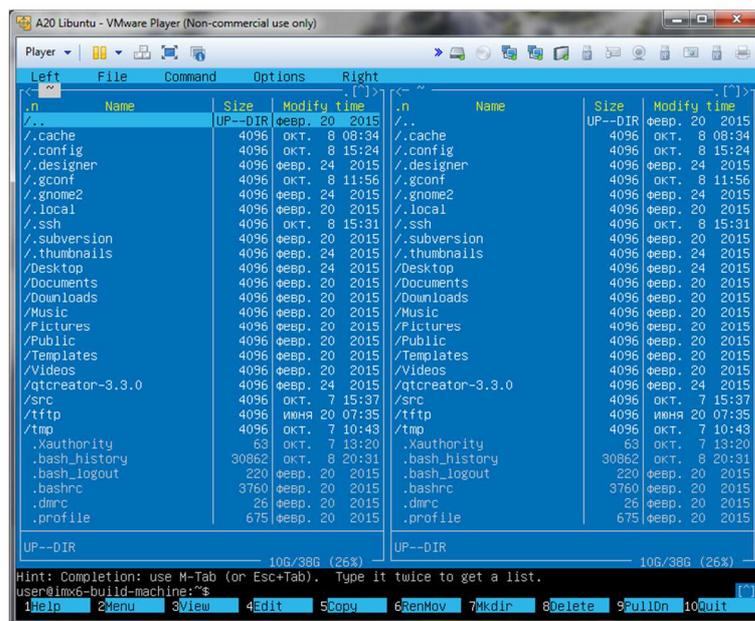
В системе присутствует один пользователь, **логин: user, пароль: 123456**  
Суперпользователя в системе нет, для запуска приложений с его привилегиями необходимо использовать **sudo**.

Login: user  
Pasw: 123456

Работать с файлами и текстами не всегда удобно через графический рабочий стол, для переключения в консольный режим необходимо нажать Ctrl+Alt+F(1-6) (Ctrl+Alt+F7 – переключение на графический рабочий стол).

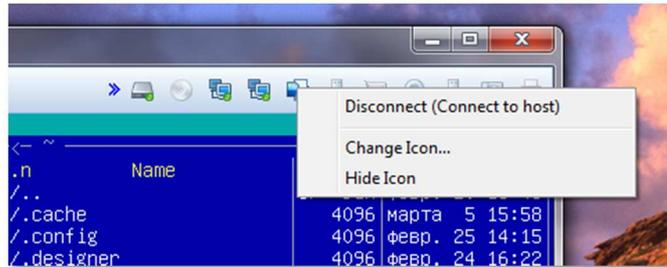


mc – Midnight Commander, файловый менеджер



Карты памяти можно записывать в виртуальной машине

Через раздел **Player>Removable devices** можно подключать-отключать к виртуальной машине различные системные устройства, например, USB устройства, картридеры и т.п. Эта же функция дублируется через графическую панель:



Функция очень полезна, потому как позволяет подключить картридер непосредственно в виртуальную машину и напрямую оперировать с картами памяти.

Во время паузы останавливается системное время!

Нет необходимости выключать виртуальную машину после завершения работы, можно перевести машину в режим паузы, а в следующий раз продолжить работу с момента паузы. **Внимание!** В режиме паузы останавливается системное время, что может негативно сказаться на сборке вновь скачиваемых архивов, во избежание этого следует подстраивать системное время, либо проводить перезагрузку виртуальной машины.

IP виртуальной машины: 192.168.0.2

При правильной настройке сетевых интерфейсов, виртуальная машина должна иметь доступ в Internet, PC должен иметь успешный ping по адресу 192.168.0.2 (адрес Bridget сетевого адаптера в виртуальной машине) и при подключенной плате должен быть успешным ping адреса 192.168.0.136.



## Первая сборка не быстрый процесс

Далее можно выйти из меню с сохранением изменений и приступить к сборке.

```
> make
```

Процесс сборки зависит от производительности вашего PC, может занять несколько часов, типичное время сборки с `a20sk_qt5gst_defconfig` на 4-х ядерном PC составляет 2 часа (при выделении 4 ядер для виртуальной машины). При повторных запусках будут собираться только вновь добавленные пакеты, что не требует много времени. **Важно!** При исключении пакета из сборки он не удаляется, остается в сборке КФС до чистки.

В результате сборки в папке **output** появится несколько новых папок:

- **build** – содержит рабочие папки собираемых пакетов, а также ядро и загрузчик
- **target** – результат сборки, скопировав сюда файл, он появится в образах КФС после выполнения `make`
- **images** – ядро, загрузчик, архив КФС ...

## Изменили исходный код или состав ядра – выполните `kernel_rebuild.sh`

Изменить состав ядра Linux можно командой **make linux-menuconfig**.

В корне `buildroot` папки присутствуют вспомогательные скрипты:

- **kernel\_rebuild.sh** – пересобирает ядро Linux  
При редактировании исходников или состава ядра Linux в папке `output/build/linux-custom` команда `make` не пересобирает ядро, скрипт устраняет данный недостаток.
- **uboot\_rebuild.sh** – пересобирает загрузчик `uboot`.
- **mk-sd.sh** – используется для подготовки загрузочных карт памяти.

## Busybox

Большинство системных утилит реализованы не отдельными программами, а специальным многофункциональным средством Busybox, в папке /bin находятся не программы, а ссылки на Busybox с указанием требуемого вызова.

Изменить состав Busybox можно командой **make busybox-menuconfig**, далее **make busybox-rebuild**.

**make clean (чистка)** – удаляет все результаты сборки

**make clean** удалит все изменения и настройки в ядре и пакетах

**Важно!!!** **make clean** удалит все в папке output, т.е. все изменения в ядре Linux и прочих пакетах, поэтому стоит внимательно относиться к применению этой команды и заблаговременно сохранять ваши изменения. Для того, чтобы изменения в ядре Linux не были безвозвратно утеряны после чистки, необходимо перед этой командой создать архив ядра и скопировать его в папку downloads:

```
> cd /home/user/src/buildroot-2015.08-a20-sk/output/build/linux-custom
> make clean
> tar cjf /home/user/src/buildroot-2015.08-a20-sk/downloads/linux-sunxi-9a1cd03-sk.tar.bz2 ./*
```

## Подключение модулей расширения

Имеется возможность подключения следующих модулей расширения:

- **SK-ATM0700D4-Plug** – модуль 7” LCD 800x480 с резистивным сенсорным экраном
- **SK-SIM800C-Plug** – модуль GSM/GPRS/Bluetooth
- **SK-SMotor-Plug** – модуль управления тремя шаговыми двигателями

**SK-SIM800C-Plug** и **SK-SMotor-Plug** можно подключить к разъему X2, предусмотрена аппаратная совместимость (питание и наличие необходимой периферии на определенных выводах), но в составе BSP на данный момент это не отражено.

SK-ATM0700D4-Plug может быть подключен двумя способами:

- 1 **RGB**, для этого необходимо соединить кабелем разъемы X14 на плате и X12 на модуле расширения. На модуле расширения необходимо разомкнуть J10, J1, J2, J3, J6, J7, а J4, J5, J8, J9 перевести в положение 2-3. Предварительно пересобрать систему, выбрав в меню настройки **Starterkit A20 board fb0 mode – LCD,WVGA 800x480**
- 2 **LVDS**, для этого необходимо соединить кабелем разъемы X13 на плате и X11 на модуле расширения. На модуле расширения необходимо замкнуть J10, разомкнуть J1, J2, J3, J6, J7, а J4, J5, J8, J9 перевести в положение 2-3. Предварительно пересобрать систему, выбрав в меню настройки **Starterkit A20 board fb0 mode – LVDS,WVGA 800x480**

После загрузки системы, необходимо откалибровать сенсорный экран, выполнив команду **ts\_calibrate**, после чего командой **ts\_test** можно оценить работу сенсорного экрана.

### **Важно!**

*J10 на LCD модуле определяет тип используемого соединения: RGB или LVDS.*

*Манипуляции с джамперами переключают сенсорный экран со встроенного контроллера на контроллер процессора A20*

## Подготовка загрузочной карты памяти

Предварительно подключите картридер к виртуальной машине, см. раздел «Первый запуск виртуальной машины», убедитесь, что карта появилась в устройствах виртуальной машины:

```
> ls /dev/sd*
```

```
/dev/sda /dev/sda1 /dev/sda2 /dev/sda5 /dev/sdb /dev/sdb1
```

Если к виртуальной машине не подключены другие накопители, устройство **sdb** и есть карта памяти.

Для подготовки карты памяти необходимо выполнить скрипт **mk-sd.sh**, а в качестве аргумента указать имя устройства карты памяти:

```
> cd /home/user/src/buildroot-2015.08-a20-sk
```

```
> sudo ./mk-sd.sh /dev/sdb
```

Команда `sudo` потребует ввести пароль суперпользователя – 123456, в результате должен получиться следующий лог операции:

```
create the bootable SD card for SK-A20 on device /dev/sdb
```

```
delete partition table      [OK]
create primary partition    [OK]
prepare u-boot              [OK]
make ext4 fs                [OK]
mount partition             [OK]
extract rootfs.tar          [OK]
umount partition            [OK]
remove tmp mount point      [OK]
```

```
all commands are completed without errors, SD is ready
```

Карта памяти готова к использованию.

## QtCreator

**QtCreator** – среда разработки позволяющая одним щелчком мыши собирать и запускать приложения пользователя на плате, а так же вести пошаговую отладку.

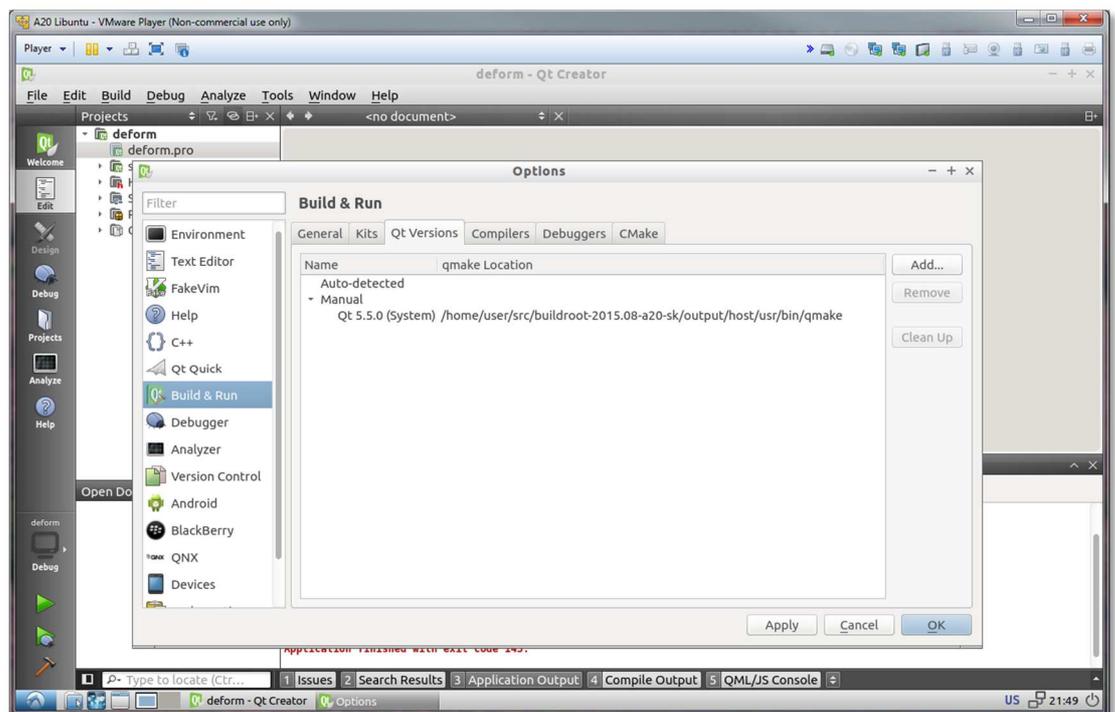
Так сложилось, что настройки Qt проектов взаимосвязаны с рабочими папками output/build, которые очищаются перед подготовкой к публикации виртуальной машины, поэтому необходимо настроить QtCreator перед началом работы.

Примечание. Вполне возможно, когда Вы читаете этот раздел, мы решили не экономить Ваш трафик (примерно 1Г) и публикуем виртуальную машину без предварительной чистки от чего настройки QtCreator не удалятся.

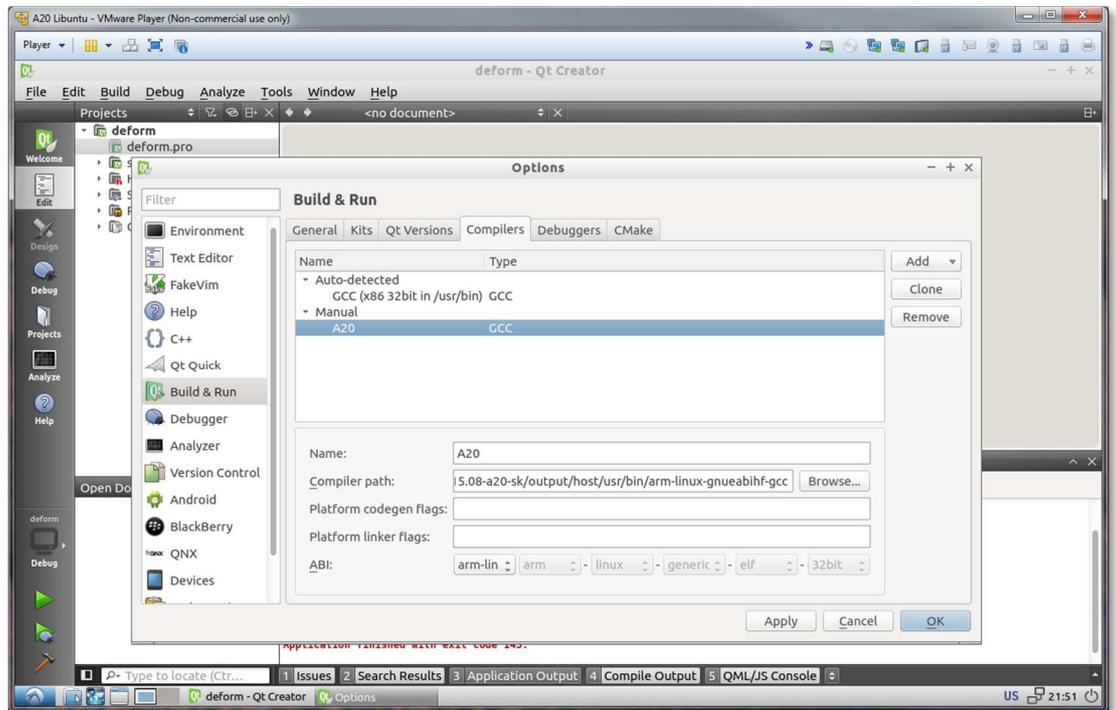
[Настроить QtCreator](#)

Предварительно соберите Buildroot с конфигурацией a20sk\_qt5gst\_defconfig и включите плату/модуль. Подключите Ethernet кабель и проверьте наличие соединения с платой, должен быть успешным ping из виртуальной машины с адресом 192.168.0.136.

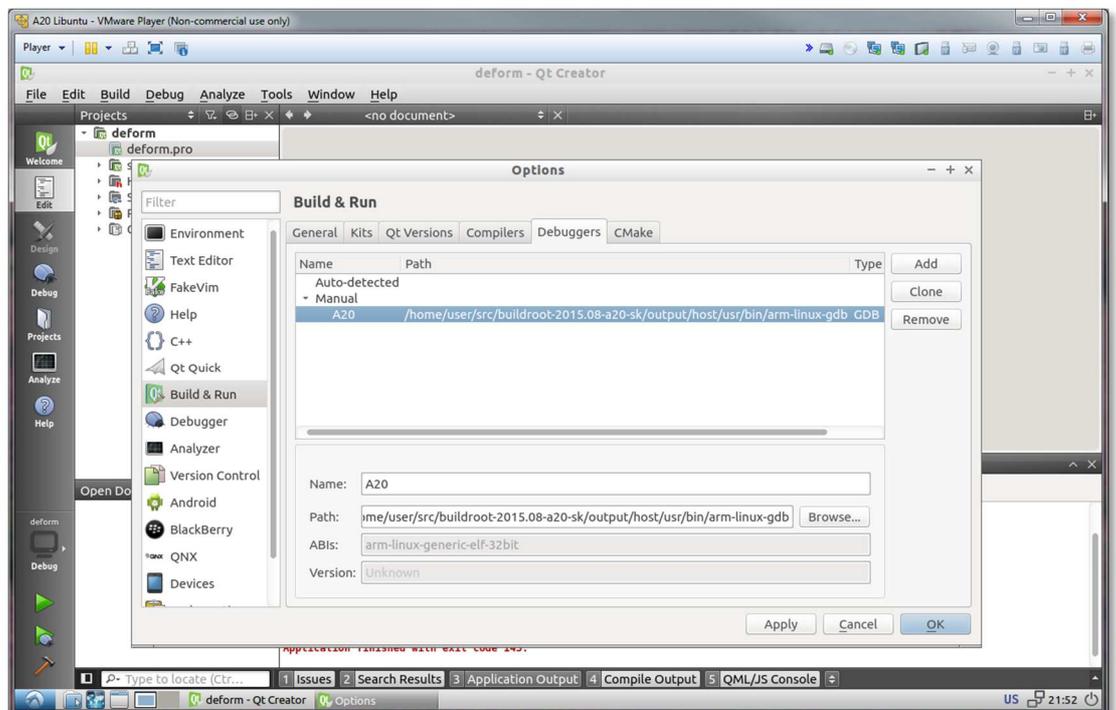
Запустите скрипт **QtCreator** на рабочем столе, зайдите в меню **Tools->Options->Build & Run**, закладка **Qt Version** – добавьте путь к **qmake**, как показано на рисунке:



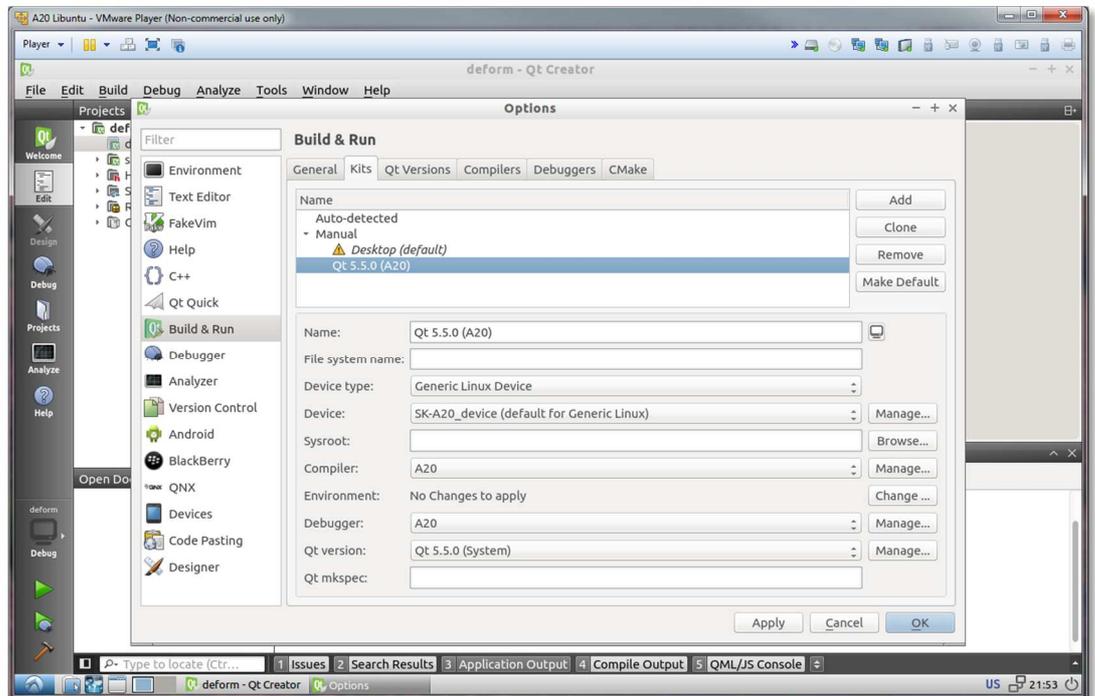
В закладке **Compilers** добавьте компилятор:



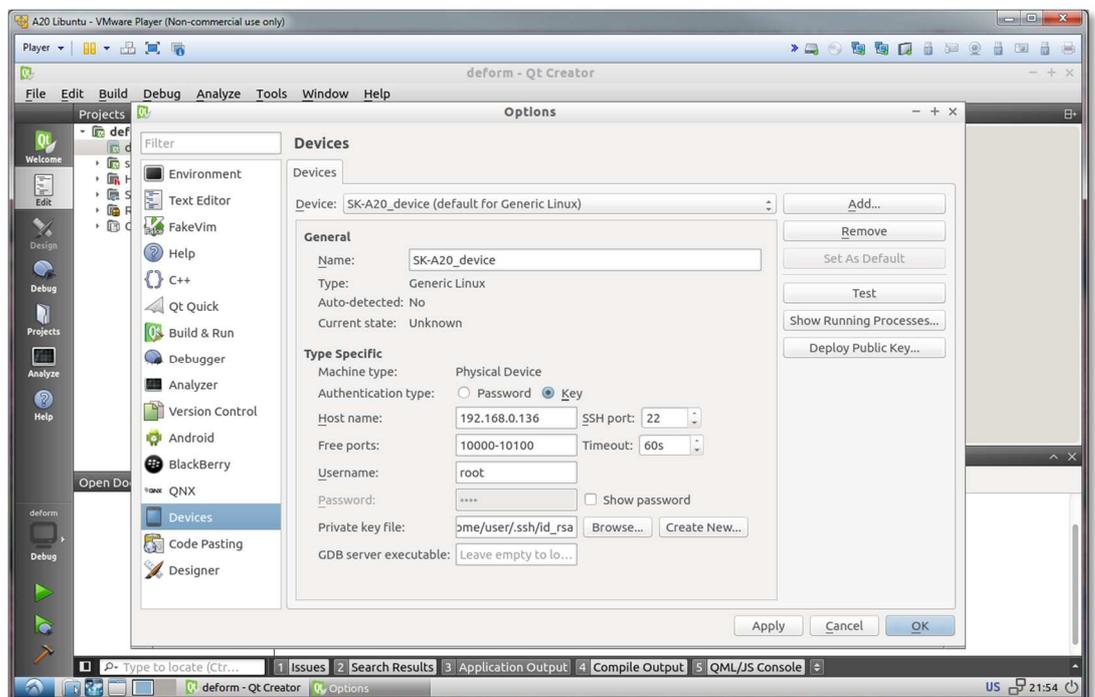
В закладке **Debuggers** добавьте отладчик:



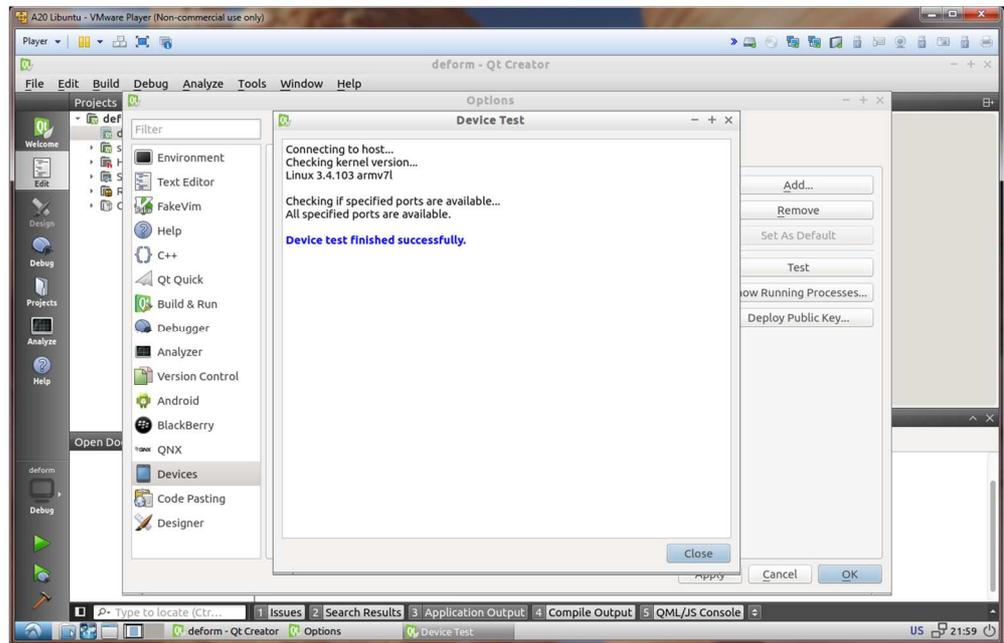
В закладке **Kits** добавить правило, как на рисунке:



Перейти в раздел **Tools->options->Devices** и добавить **Generic Linux Device**, указать IP адрес **192.168.0.136**, указать **Authentication type - key**, указать путь до ключа **Private key file: /home/user/.ssh/id\_rsa**

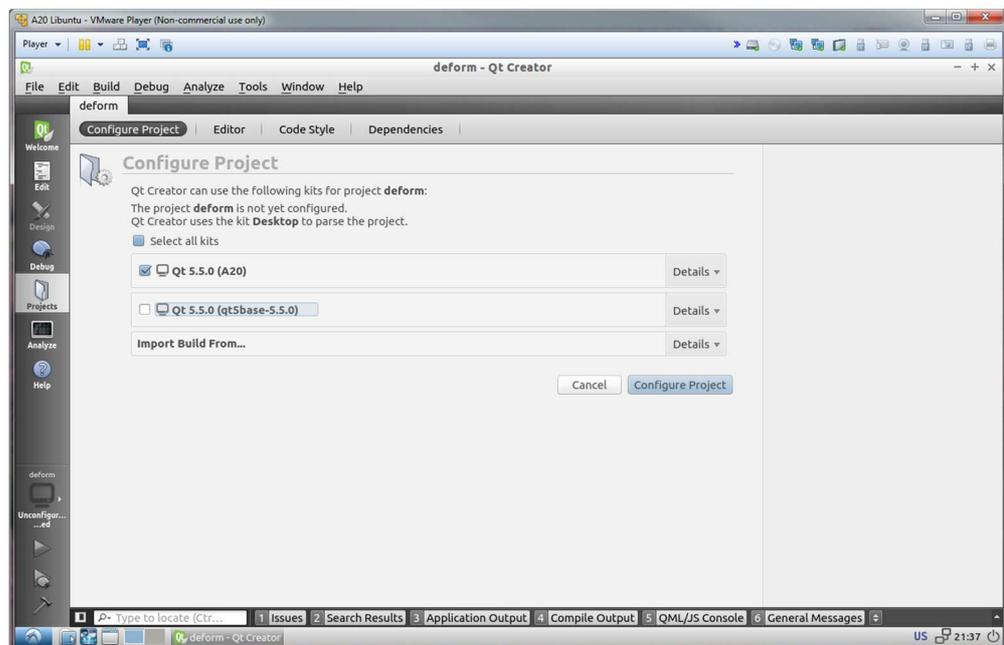


Проверить соединение с платой можно нажав кнопку **Test**.

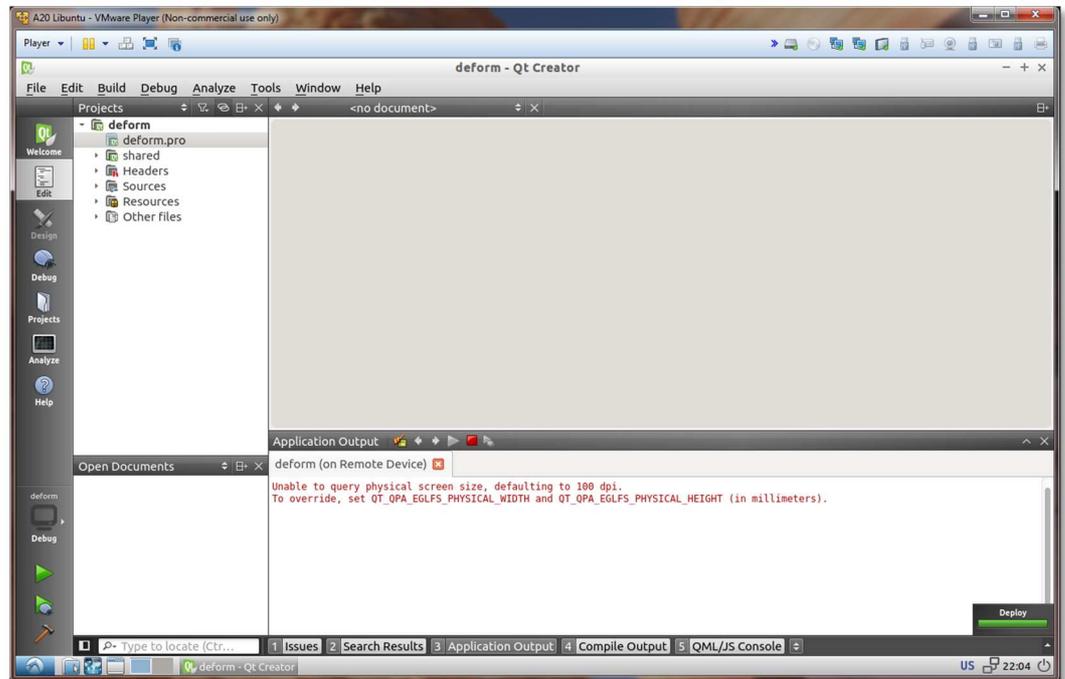


Итак, QtCreator настроен для сборки и соединения с платой, далее необходимо открыть демонстрационный проект Qt и проверить настройки.

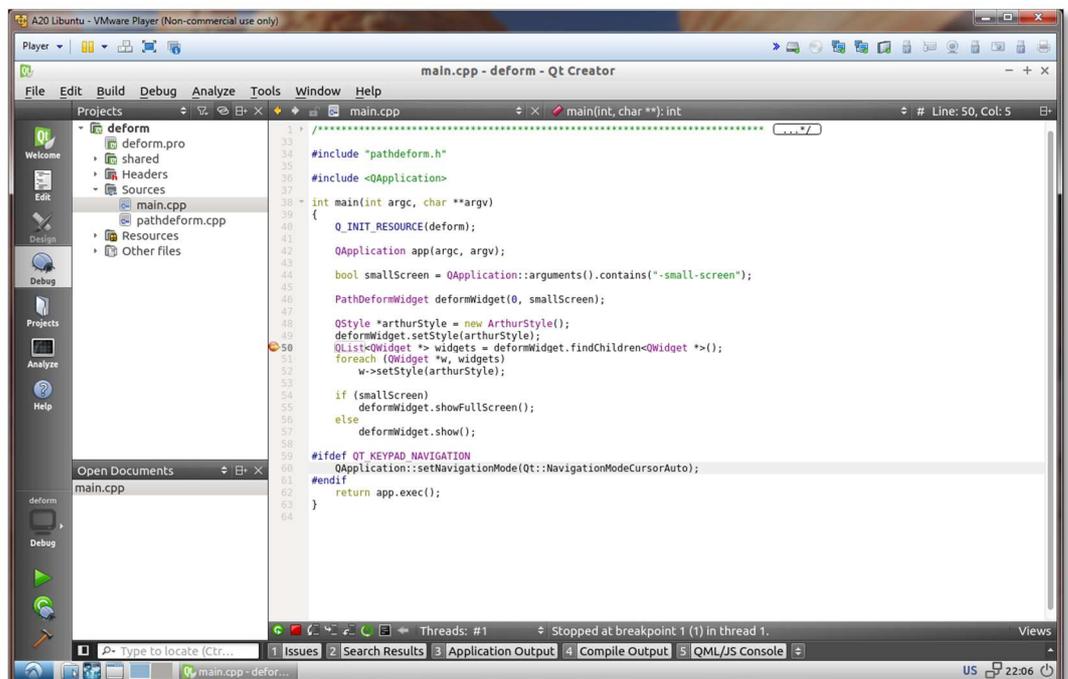
**File->Open File or Project...** выберите PRO файл примера `/home/user/src/buildroot-2015.08-a20-sk/output/build/qt5base-5.5.0/examples/...` и укажите Kit для A20:



Далее **Build->Rebuild All**, по завершении, нажать кнопку Run, на плату скопируется и запустится приложение:



Предварительно установив точки останова в исходном коде, можно вести пошаговую отладку:



## Сведения об организации

---

[www.starterkit.ru](http://www.starterkit.ru)

Россия, г.Ижевск, ул.Телегина д.30

**email:** info@starterkit.ru

**Тел.** +7-3412-478-448

[www.otladka.com.ua](http://www.otladka.com.ua)

Украина, г. Киев, ул. Молодогвардейская 7Б оф. 4

**email:** evodbg@gmail.com

**Тел.** +38-044-362-25-02