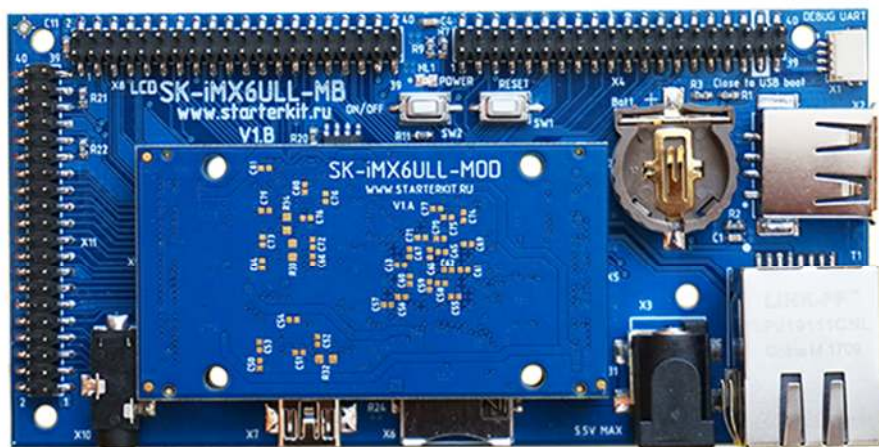
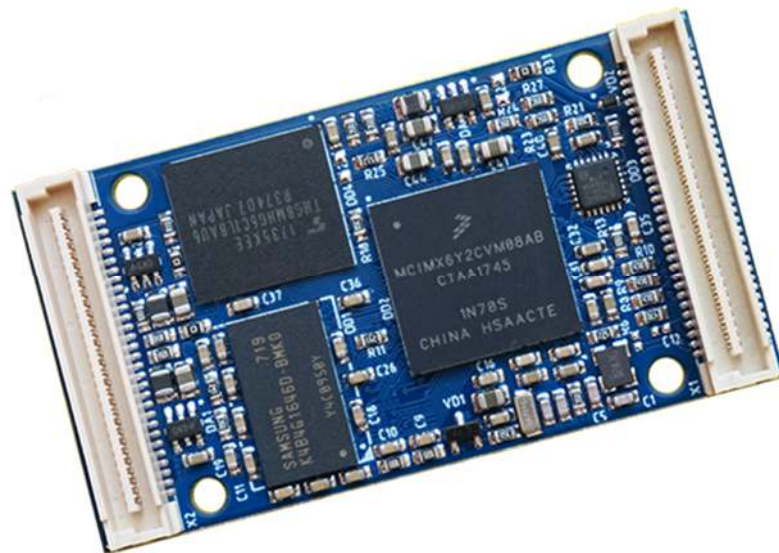


SK-IMX6ULL-MOD

ИНСТРУКЦИЯ ПОЛЬЗОВАТЕЛЯ ПРИ СОВМЕСТИМ
ИСПОЛЬЗОВАНИИ С ПЛАТОЙ SK-IMX6ULL-MB



SK-iMX6ULL-MOD

ПРОЦЕССОРНЫЙ МОДУЛЬ SK-IMX6ULL-MOD

- NXP(Freescale) ARM Cortex-A7 800МГц
- DDR3 512МБайт, DDR-800
- eMMC Flash 8ГБайт
- 100/10М Ethernet PHY
- Интерфейсы: USB, Ethernet, RMII, LCD RGB24, UART, SPI, I2C, PWM, GPIO, CAN
... общее количество возможных GPIO – 87
- Габариты: 56x30x6мм
- Температурный диапазон -40 ... +85С

МАТЕРИНСКАЯ ПЛАТА SK-IMX6ULL-MB

Материнская плата SK-iMX6ULL-MB предназначена для совместного использования с процессорным модулем SK-iMX6ULL-MOD, содержит:

- Ethernet 100/10М, USB-Host, USB-OTG, MicroSD держатель, держатель батареи для RTC
- Разъем для подключения SK-FT230, EV-FT230 – USB-UART консоль
- RS485 PHY, CAN PHY
- Аудио выход, реализован на основе PWM выхода процессора с последующей RC фильтрацией

ВОЗМОЖНОСТЬ ПОДКЛЮЧЕНИЯ МОДУЛЕЙ РАСШИРЕНИЯ

- SK-ATM0700D4-Plug – модуль расширения LCD диагональ 7 дюймов 800x480 с резистивным сенсорным экраном

ПИТАНИЕ

Напряжение питания 5В. **Внимание!!!** Обязательно от стабилизированного источника питания! Центральная жила разъема – «+».

Потребляемый ток в зависит от подключаемой периферии, сам модуль потребляет не более 0,3А.

Рекомендуемый БП: 5В/1А.

С модуля можно получить напряжение питание 3,3В нагрузкой не более 0,4А.

ПЕРВОЕ ВКЛЮЧЕНИЕ

В минимальном варианте, достаточно подключить USB кабель к разъему X7, после загрузки системы USB OTG порт платы перейдет в режим виртуального COM порта. При первом запуске потребуются установить драйвер USB-COM порта (из папки Linux/USB-COM_driver). Неудобство заключается в необходимости закрывать-открывать терминальную программу при каждом перезапуске платы. Определите в диспетчере устройств номер COM порта, который присваивается и откройте его терминальной программой (например Putty):

```
Welcome to SK-iMX6ULL-MOD
```

```
buildroot login:
```

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

SK-iMX6ULL-MOD

Из-за соображений технологичности производства, в составе системы отгружаемых изделий содержится минимальный набор утилит и сервисов, а так же форматирование eMMC осуществляется не в полном объеме.

После того как убедитесь в работоспособности платы-модуля, нужно записать вариант сборки на основе Qt5 (описание процедуры см. ниже), в контексте которого идет дальнейшее описание возможностей подключения модулей расширения, так же в системе настроены сервисы FTP и SSH, доступ к которым можно получить, подключив плату Ethernet кабелем, IP адрес платы 192.168.0.136.

ПОДКЛЮЧЕНИЕ SK-ATM0700D4-PLUG

Предварительно, необходимо скоммутировать выход сенсорного экрана:

- 1) замкнуть J2, J3, J6, J7
- 2) перевести в положение 1-2 J4, J5, J8, J9

Соединительный кабель на SK-ATM0700D4-Plug необходимо подключить к X12, на плате SK-iMX6ULL-MB к разъему X8.

Штатно, ядро Linux подразумевает совместное использование с SK-ATM0700D4-Plug, после загрузки системы на экране отобразится логотип Linux. Для демонстрации, в корневую систему включен пример Qt – affine, для его запуска достаточно выполнить `/root/affine.sh`. При первом запуске выполнится процедура калибровки экрана (для рекалибровки нужно выполнить `ts_calibrate`).

```
echo 0 > /sys/class/backlight/backlight/brightness - выключает подсветку
```

```
echo 7 > /sys/class/backlight/backlight/brightness - устанавливает максимальный уровень яркости подсветки
```

ПОДКЛЮЧЕНИЕ SK-iMX6ULL-LCD-MB

Основное отличие материнской платы SK-iMX6ULL-LCD-MB заключается в использовании встроенного в процессор контроллера резистивного сенсорного экрана, это отличие необходимо учитывать на этапе сборки ядра linux, для этого в системе добавлен отдельный профиль сборки системы, выбор которой осуществляется:

```
> cd /home/user/src/buildroot-2017.08-sk-ull  
> make menuconfig
```

Далее, необходимо зайти в меню «Bootloaders» и в селекторе «Starterkit i.MX6ULL board» выбрать: «SK-iMX6ULL-MOD + LCD-MB».

Далее, выполнить скрипт `uboot_rebuild.sh`.

SK-iMX6ULL-MOD

SK-IMX6ULL-MOD, СПОСОБЫ ЗАГРУЗКИ

Источник загрузки модуля – встроенная eMMC flash, выбор источника загрузки осуществлен программированием eFuse на этапе производства.

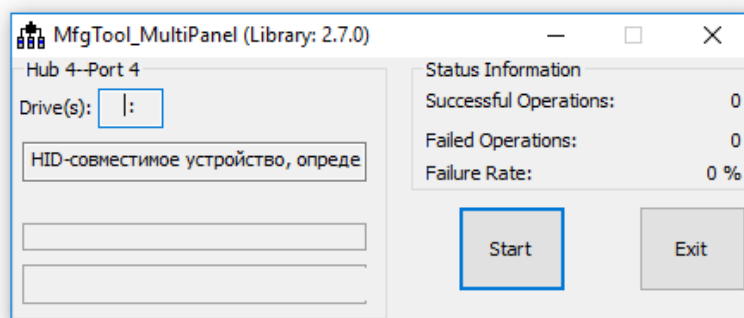
Запись во встроенную eMMC flash может осуществляться через USB-OTG порт.

ПРОГРАММИРОВАНИЕ EMMC FLASH

Для программирования используется утилита MFGTools, предварительно необходимо распаковать архив.

Замкните между собой контакты 37-38 на разъеме X4 (в пределе, это необходимо только на этапе сброса или включения питания), подключите USB кабель к разъему X7 (нажмите кнопку RESET, если до момента подключения USB кабеля модуль был запитан через разъем X3). Windows должна обнаружить новое HID устройство (установка дополнительных драйверов не требуется).

Запустите MfgTool2.exe, должно получиться:



Нажмите кнопку «Start», в консоли будет отображаться рабочий процесс, после завершения модуль отключит питание. После этого необходимо разорвать соединение между контактами 37-38 разъема X4 и можно приступать к работе.

SK-iMX6ULL-MOD

ВИРТУАЛЬНАЯ МАШИНА, ОСНОВНЫЕ ПОЛОЖЕНИЯ

Виртуальная машина VMware предназначена для сборки ядра Linux и корневой файловой системы без необходимости выделять для этого отдельный PC.

Виртуальная машина основана на Libuntu 14.10.

Сборка ядра Linux и КФС интегрирована и осуществляется в одном пакете **Buildroot**. **Buildroot** (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 собирать и запускать приложение на плате или вести пошаговую отладку.

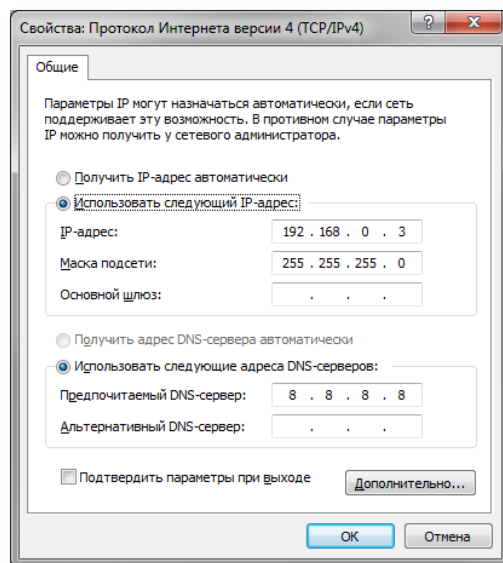
Qt наиболее предпочтительный пакет для написания приложений, т.к. в нем существует огромное количество библиотек и примеров, имеет широкую распространенность, графические приложения Qt работают значительно быстрее и потребляют гораздо меньше ресурсов (особенно памяти) системы в сравнении с приложениями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

НАСТРОЙКА ВИРТУАЛЬНОЙ МАШИНЫ

Перед началом работы необходимо скачать плеер виртуальной машины VMware, бесплатно распространяемый на сайте www.vmware.com.

Виртуальная машина имеет 2 сетевых интерфейса:

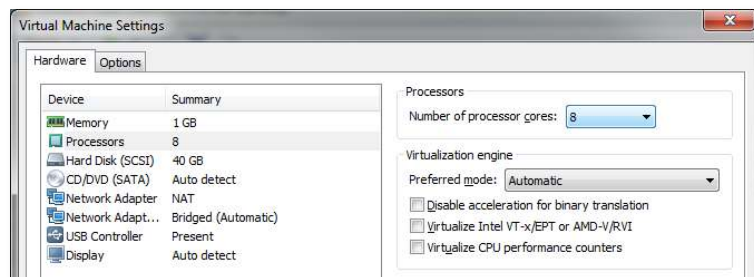
- 1) NAT – для доступа в Internet
- 2) Bridget – для взаимодействия по локальной сети, необходимо настроить VMware network adapter



SK-iMX6ULL-MOD

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.



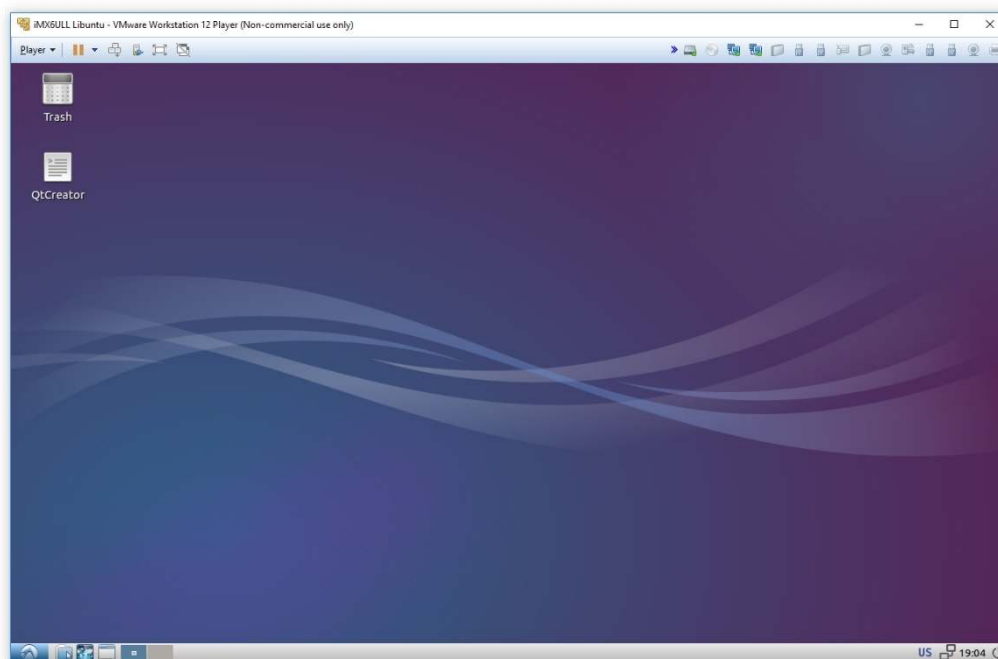
ПЕРВЫЙ ЗАПУСК ВИРТУАЛЬНОЙ МАШИНЫ

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.

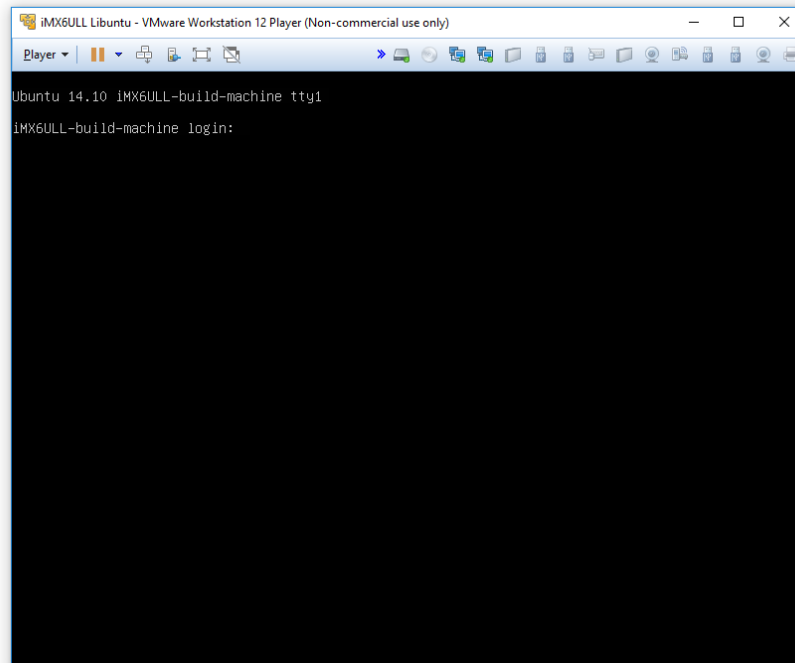


Разрешение экрана можно изменить в разделе **Perfences->Monitor Settings**

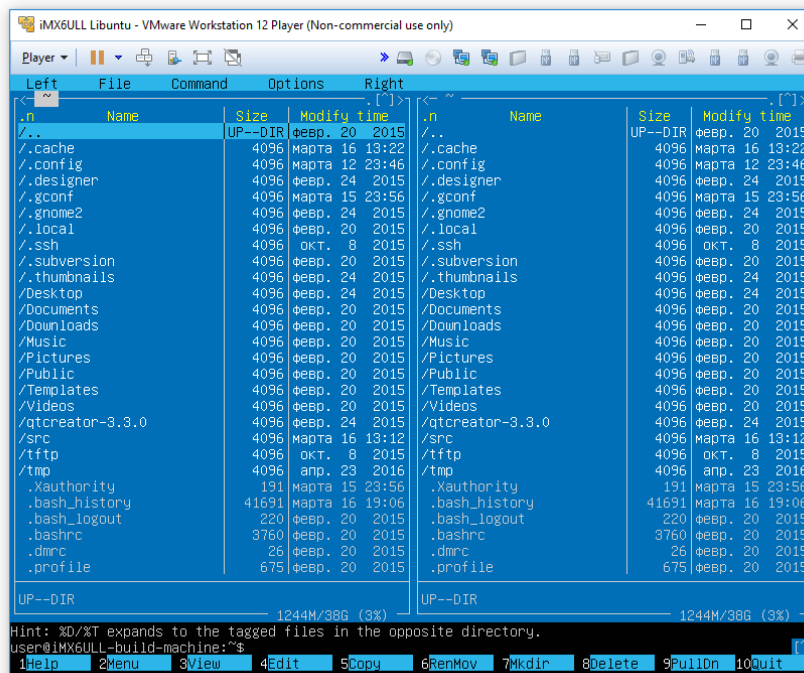
В системе присутствует один пользователь, **логин: user, пароль: 123456**
Суперпользователя в системе нет, для запуска приложений с его привилегиями необходимо использовать **sudo** (пароль 123456).

SK-iMX6ULL-MOD

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).

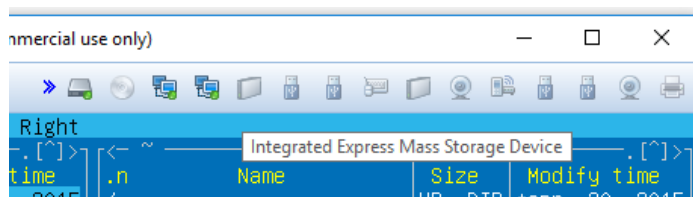


mc – Midnight Commander, файловый менеджер



SK-iMX6ULL-MOD

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.

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.

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.

BUILDROOT

Buildroot располагается в папке `/home/user/src/buildroot-2017.08-sk-ull`

Перед началом сборки необходимо сконфигурировать Buildroot, имеются следующие варианты сборки:

1. `imx6ullsk_mfg_defconfig` – вариант сборки системы для работы MFGTools утилиты (скорее всего вам он не потребуется)
2. `imx6ullsk_min_defconfig` – вариант с минимальным составом утилит
3. `imx6ullsk_qt5_defconfig` - вариант сборки включающий в себя обширный состав утилит и сервисов, а так же Qt5.

```
> cd /home/user/src/buildroot-2017.08-sk-ull
> make clean
> make imx6ullsk_qt5_defconfig
> make menuconfig
```

Выбор изделия (модуль/плата/тип материнской платы) iMX6ULL под которое будет осуществляться сборка осуществляется в меню «Bootloaders» и в селекторе «Starterkit i.MX6ULL board».

Основные команды:

- **make** – сборка системы
- **make menuconfig** – запуск меню настроек и состава требуемых пакетов
- **clean** – очистка системы, **ВНИМАНИЕ!!!** Полностью удаляется содержимое папки `output`, что удалит все изменения в исходных кодах и настройки, перед чисткой нужно позаботиться об этом.
- **make linux-menuconfig** – запуск конфигуратора ядра Linux
- **make linux-rebuild** – принудительная сборка ядра Linux
- **make busybox-menuconfig** – запуск конфигуратора Busybox
- **make busybox-rebuild** – принудительная сборка Busybox

SK-iMX6ULL-MOD

- **make uboot-rebuild** – принудительная сборка загрузчика U-boot

Длительность процесса сборки зависит от производительности вашего PC, может занять несколько часов. 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 ...

Для обновления КФС или ядра Linux на модуле SK-iMX6ULL-MOD, необходимо скопировать файл `output/images/footfs.tar` в `mfgtools\Profiles\Linux\OS Firmware\files` и провести процедуру описанную в разделе «Программирование eMMC flash».

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.

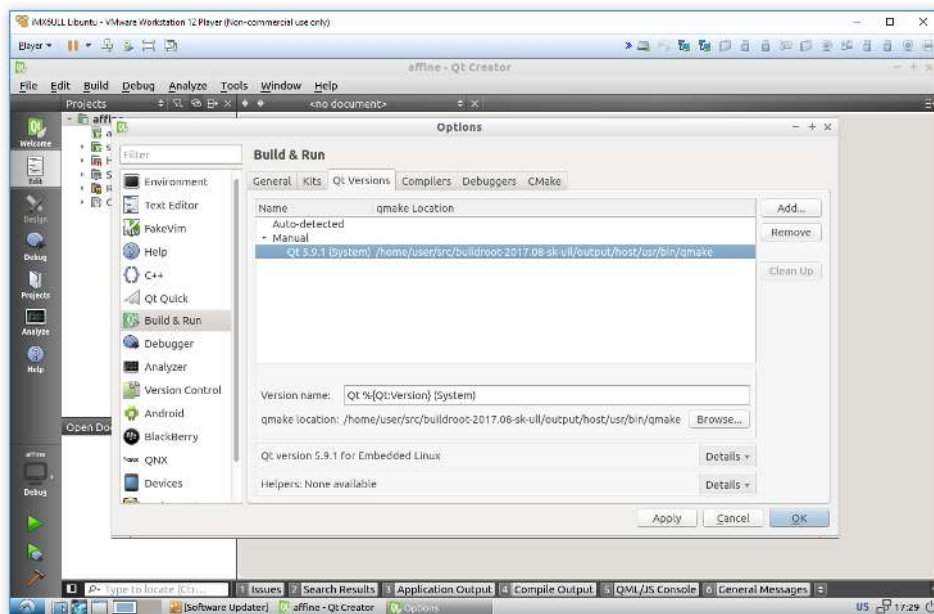
QTCREATOR

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

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

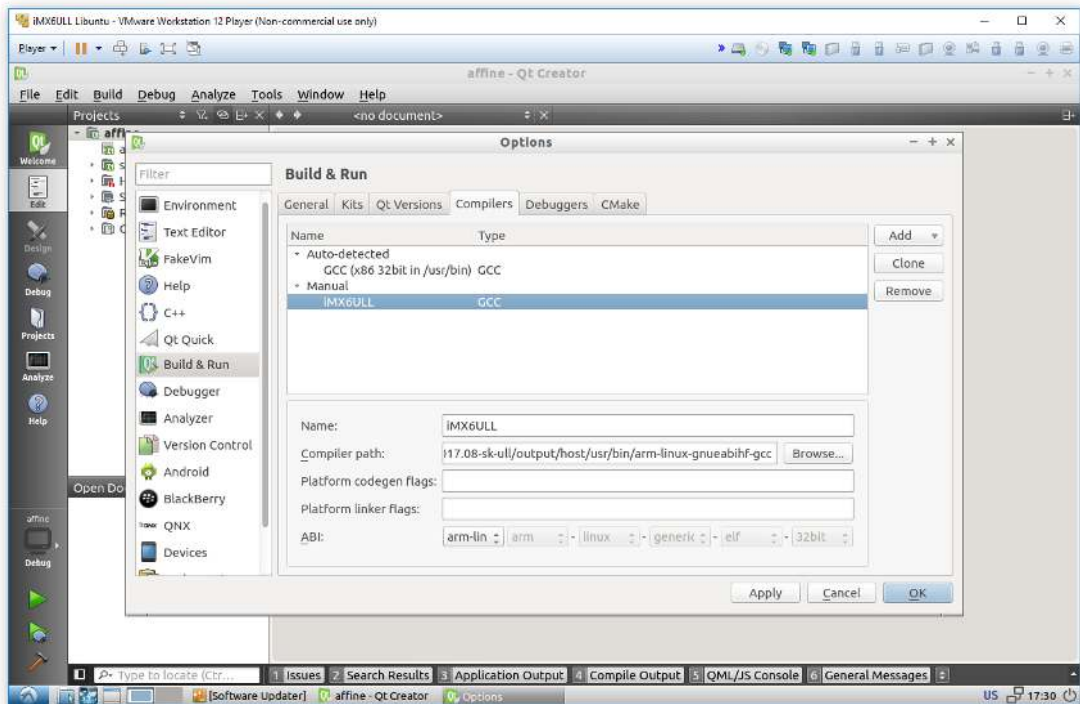
Предварительно соберите Buildroot с конфигурацией `imx6ullsk_qt5_defconfig` и запишите КФС в eMMC flash. Подключите Ethernet кабель и проверьте наличие соединения с платой, должен быть успешным ping из виртуальной машины с адресом 192.168.0.136.

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

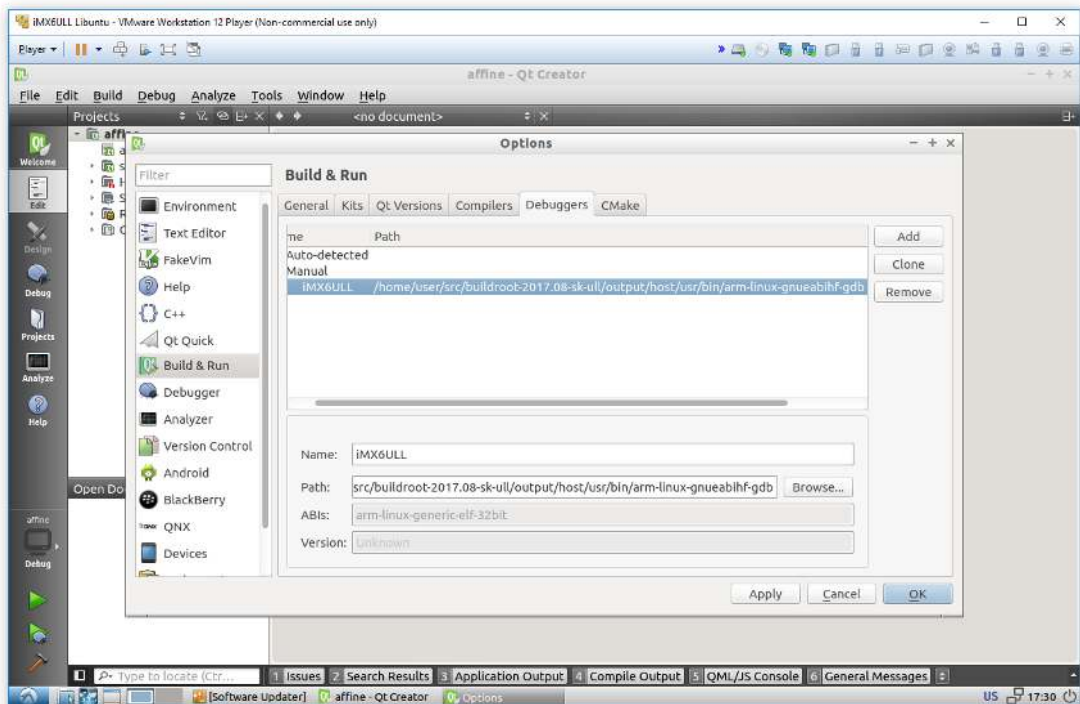


SK-iMX6ULL-MOD

В закладке Compilers добавьте компилятор:

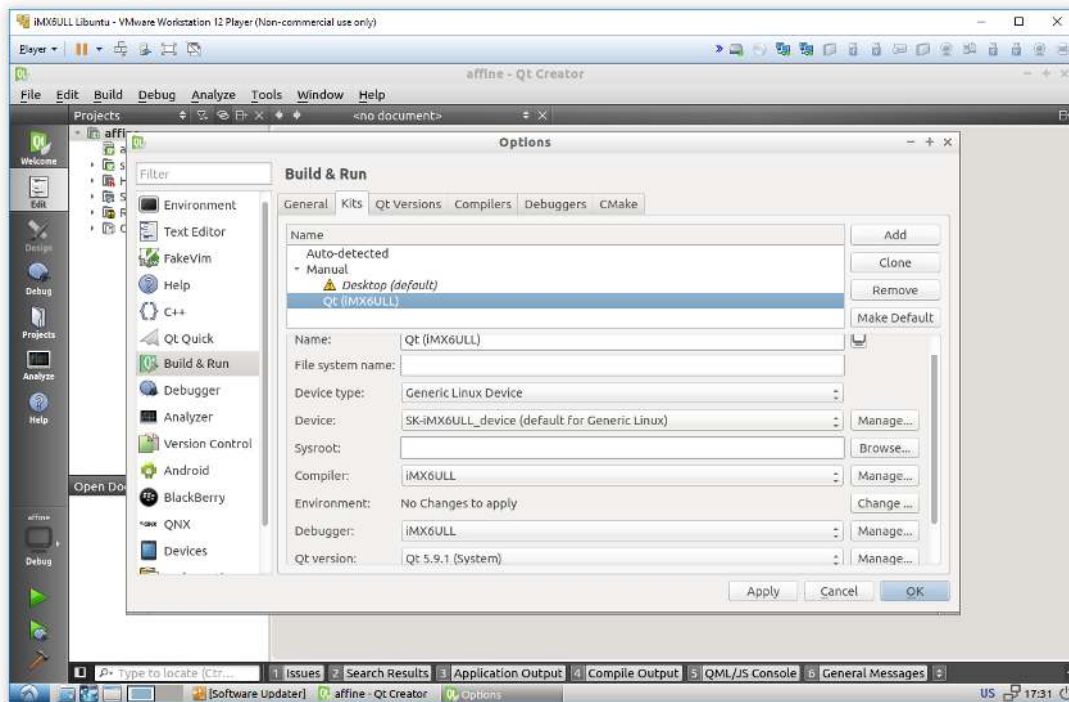


В закладке Debuggers добавьте отладчик:

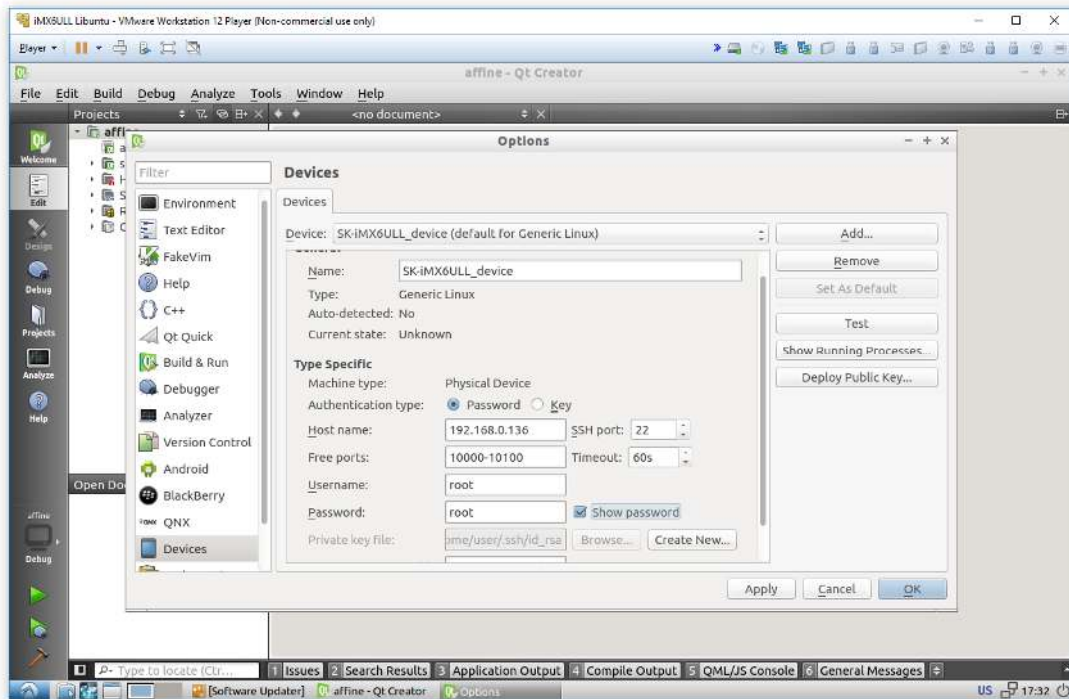


SK-iMX6ULL-MOD

В закладке Kits добавить правило:

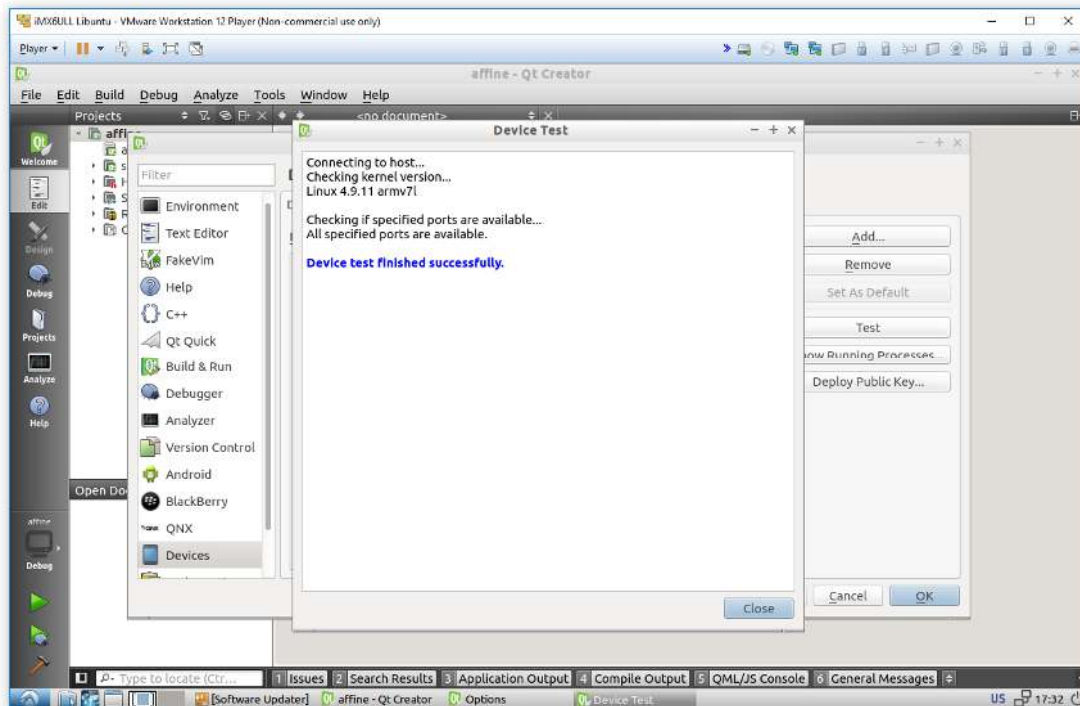


Перейти в раздел Tools->options->Devices и добавить Generic Linux Device, указать IP адрес 192.168.0.136, login:root, password:root



SK-iMX6ULL-MOD

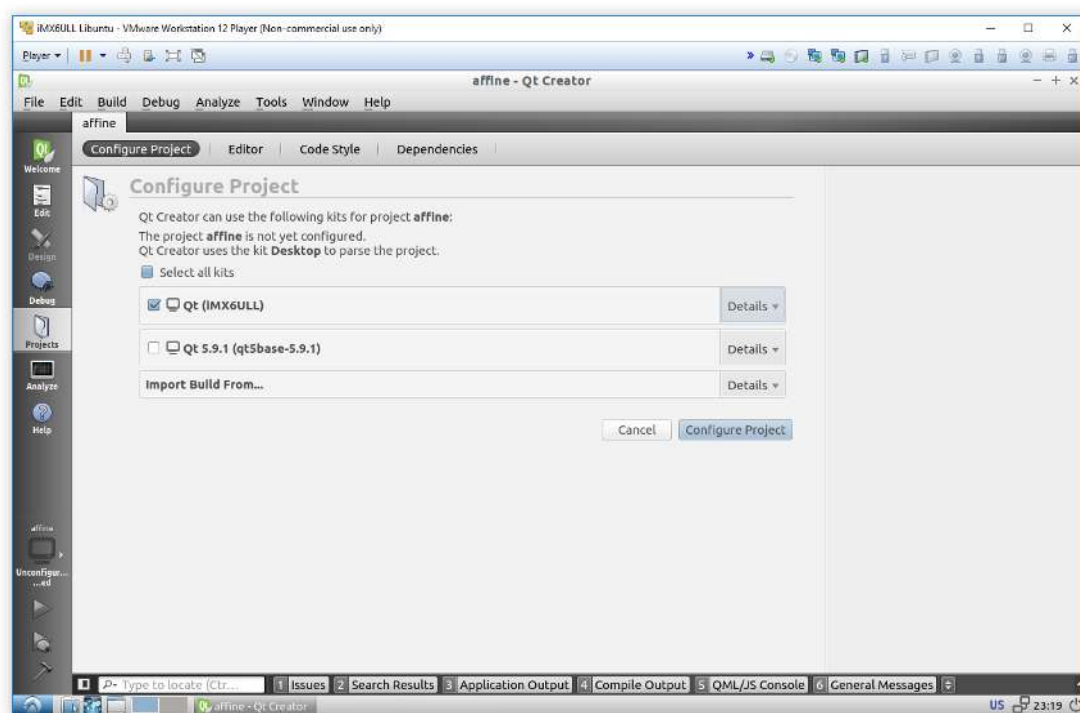
Проверить соединение с платой можно нажав кнопку Test:



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

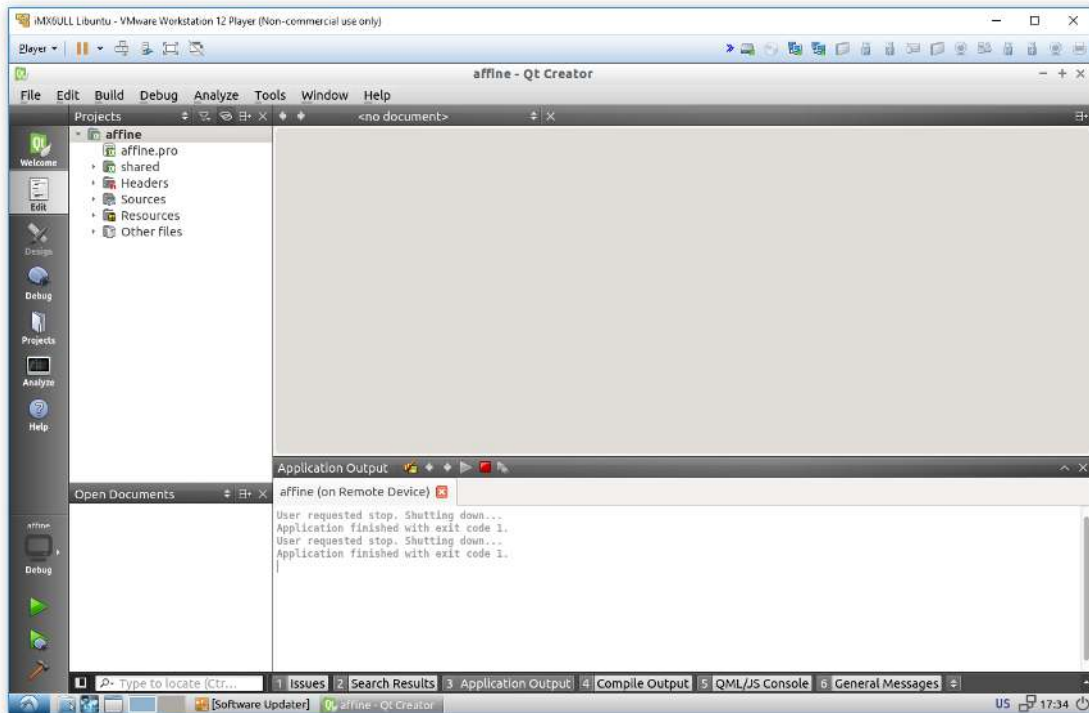
Запустите скрипт **QtCreator** на рабочем столе, зайдите в меню **File->Open File or Project ...** и выберите `/home/user/src/buildroot-2017.08-sk-ull/output/build/qt5base-5.9.1/examples/widgets/painting/affine/affine.pro`

В настройках конфигуризатора включите опцию **Qt(iMX6ULL)** и нажмите **Configure Project**.

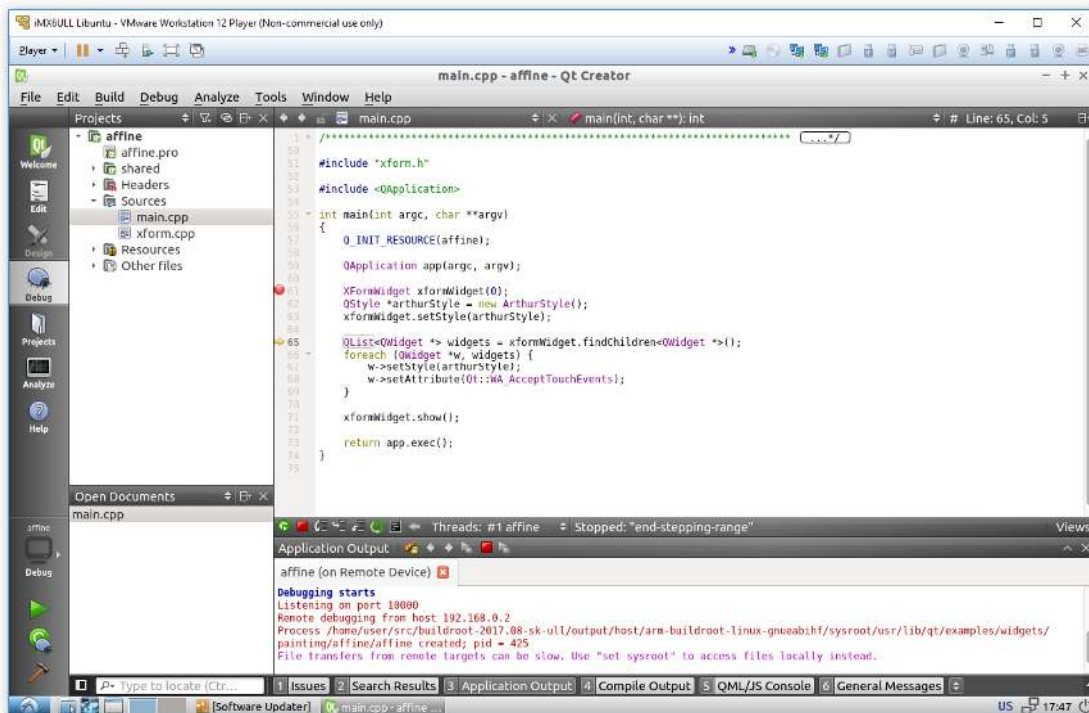


SK-iMX6ULL-MOD

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



Предварительно расставив точки останова, можно вести пошаговую отладку.



SK-iMX6ULL-MOD

ТРЕБОВАНИЯ И РЕКОМЕНДАЦИИ ПО ИНТЕГРАЦИИ МОДУЛЯ SK-iMX6ULL-MOD

- 1) Хочу обратить особое внимание на особенность включения питающих напряжений процессора iMX6ULL (впрочем, это условие есть практически у всех семейств процессоров iMX):

4.2.3 Power Supplies Usage

All I/O pins should not be externally driven while the I/O power supply for the pin (NVCC_xxx) is OFF. This can cause internal latch-up and malfunctions due to reverse current flows.

Иными словами – если при включении питания процессора на любом из GPIO будет присутствовать какое-либо напряжение, внутри процессора может что то «защелкнуться» и привести к непредсказуемым последствиям (в лучшем случае он зависнет).

Из опыта эксплуатации iMX6S/D известно, что такое «защелкивание» может приводить к короткому замыканию внутри процессора и если источник питания обладает достаточной мощностью – к выведению микросхемы процессора из строя.

Эта особенность носит статистический характер - не факт, что его последствия проявятся на 10 экземплярах, но на партиях в сотни изделий скорее всего начнут появляться модули выходящие из строя без видимых причин при включении питания.

Так же хочу обратить внимание на зависимость данного «эффекта» от технологических аспектов производства самих процессоров, т.е. бывали случаи: в одном году интегрировали без проблем 100 модулей, а через год следующие 100 модулей через один переходили в короткое замыкание.

Во избежание последствий описываемого эффекта, требуется тщательно проанализировать все GPIO модуля на предмет возможного возникновения потенциала перед включением питания, на этапе прототипирования перепроверить отсутствие каких-либо потенциалов на всех подключаемых GPIO (для выявления случаев «паразитной» запитки). Если гарантировано невозможно обеспечить нулевой потенциал на линии, обязательно поставить на этих линиях буфер, а на этапе прототипирования проверить отсутствие «паразитной» запитки.

- 2) Процессор iMX6ULL может управлять включением-выключением стабилизаторов питания, что и реализовано на модуле SK-iMX6ULL-MOD. Это повлекло за собой следующие особенности:

а) обязательно требуется присутствие питающего напряжения VBAT перед включением питания, что не всегда целесообразно (обязательно использовать батарейку, даже если часы реального времени не используются). Для удовлетворения этого требования, на материнской плате SK-iMX6ULL-MOD с помощью резистивных делителей формируется напряжение VBAT и подается на модуль в случае отсутствия батареи.

б) для правильного самовыключения (после нажатия кнопки ONOFF или командой poweroff) требуются дополнительные цепи разряда, места для размещения которых физически не нашлось на SK-iMX6ULL-MOD, поэтому необходимо на материнской плате размещать резистор номиналом 100Ом, подключаемый между напряжением 3,3В и «землей».

Если нет необходимости в автономном управлении питания, пункт «б» условий можно исключить, но пункт «а» требуется к исполнению.

SK-iMX6ULL-MOD

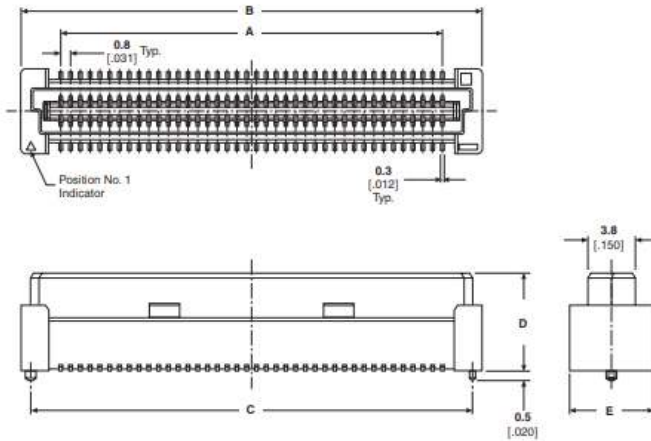
ОТВЕТНЫЕ РАЗЪЕМЫ ДЛЯ МОДУЛЯ SK-iMX6ULL-MOD

На модуле применены 60-ти контактные board-to-board разъемы с шагом 0,8мм высотой 5мм, Tусо Electronix предлагает несколько вариантов высоты ответных разъемов: 3.75, 7.75, 11.75 и 15.75мм, на материнской плате SK-iMX6ULL-MB применяются разъемы высотой 3.75мм, что в совокупности с разъемом на модуле обеспечивает расстояние от платы модуля до платы материнской платы - 5мм.

0.8 mm Free Height (FH) Connector System

PRODUCT DIMENSIONS

Receptacle



No. of Positions	Dimensions		
	A	B	C
40	15.20 .598	21.80 .858	20.20 .795
60	23.20 .913	29.80 1.173	28.20 1.110
80	31.20 1.228	37.80 1.488	36.20 1.425
100	39.20 1.543	45.80 1.803	44.20 1.740
120	47.20 1.858	53.80 2.118	52.20 2.055
140	55.20 2.173	61.80 2.433	60.20 2.370
160	63.20 2.488	69.80 2.748	68.20 2.685
180	71.20 2.803	77.80 3.063	76.20 3.000
200	79.20 3.118	85.80 3.378	84.20 3.315
Height	5H	9H	13H
Dim. D	3.75 .148	7.75 .305	11.75 .483
Dim. E	6.00 .236	6.80 .268	6.80 .268

Таблица возможных партномеров ответных:

Gender	Height	Plating Au Thickness	Package Type	Pickup Cap	40 Pos.	60 Pos.	80 Pos.	100 Pos.	120 Pos.	140 Pos.	160 Pos.	180 Pos.	200 Pos.	
Rcpt.	5H	8µ*	Tube	No	5177983-3	5177983-2	5177983-3	5177983-4	5177983-5	5177983-6	5177983-8	5177983-9	1-5177983-0	
			Tube	Yes	5179229-3	5179229-2	5179229-3	5179229-4	5179229-5	5179229-6	5179229-8	5179229-9	1-5179229-0	
			Tape and Reel	Yes	5177985-3	5177985-2	5177985-3	5177985-4	5177985-5	5177985-6	5177985-8	—	—	
		30µ*	Tube	No	6123000-3	6123000-2	6123000-3	6123000-4	6123000-5	6123000-6	6123000-8	6123000-9	1-6123000-0	
			Tube	Yes	5353999-3	5353999-2	5353999-3	5353999-4	5353999-5	5353999-6	5353999-8	—	—	
			Tape and Reel	Yes	5-5179009-3	5-5179009-2	5-5179009-3	5-5179009-4	5-5179009-5	5-5179009-6	5-5179009-8	5-5179009-9	6-5179009-0	
	9H	8µ*	Tube	Yes	—	—	—	—	—	—	—	—	—	—
			Tape and Reel	Yes	5-5179180-3	5-5179180-2	5-5179180-3	5-5179180-4	5-5179180-5	5-5179180-6	—	—	—	
			Tube	No	5084613-3	5084613-2	5084613-3	5084613-4	5084613-5	5084613-6	5084613-8	5084613-9	1-5084613-0	
		30µ*	Tube	Yes	7-5179180-3	7-5179180-2	7-5179180-3	7-5179180-4	7-5179180-5	7-5179180-6	—	—	—	
			Tube	No	5-5179010-3	5-5179010-2	5-5179010-3	5-5179010-4	5-5179010-5	5-5179010-6	5-5179010-8	5-5179010-9	6-5179010-0	
			Tape and Reel	Yes	5-5179233-3	5-5179233-2	5-5179233-3	5-5179233-4	5-5179233-5	5-5179233-6	5-5179233-8	5-5179233-9	6-5179233-0	
13H	8µ*	Tube	Yes	5084616-3	5084616-2	5084616-3	5084616-4	5084616-5	5084616-6	5084616-8	—	—	—	
		Tube	No	5-6123212-3	5-6123212-2	5-6123212-3	5-6123212-4	5-6123212-5	5-6123212-6	5-6123212-8	5-6123212-9	6-6123212-0		
		Tape and Reel	Yes	5084618-3	5084618-2	5084618-3	5084618-4	5084618-5	5084618-6	5084618-8	—	—		
	30µ*	Tube	No	5-1735480-3	5-1735480-2	5-1735480-3	5-1735480-4	5-1735480-5	5-1735480-6	5-1735480-8	5-1735480-9	6-1735480-0		
		Tube	Yes	1735481-3	1735481-2	1735481-3	1735481-4	1735481-5	1735481-6	1735481-8	1735481-9	1-1735481-0		
		Tape and Reel	Yes	1735482-3	1735482-2	1735482-3	1735482-4	1735482-5	1735482-6	1735482-8	—	—		
17H	8µ*	Tube	No	7-1735480-3	7-1735480-2	7-1735480-3	7-1735480-4	7-1735480-5	7-1735480-6	7-1735480-8	7-1735480-9	8-1735480-0		
		Tube	Yes	2-1735481-3	2-1735481-2	2-1735481-3	2-1735481-4	2-1735481-5	2-1735481-6	2-1735481-8	2-1735481-9	3-1735481-0		
		Tape and Reel	Yes	2-1735482-3	2-1735482-2	2-1735482-3	2-1735482-4	2-1735482-5	2-1735482-6	2-1735482-8	—	—		

SK-iMX6ULL-MOD

ТАБЛИЦА НАЗНАЧЕНИЯ КОНТАКТОВ SK-IMX6ULL-MOD

Большинство портов ввода-вывода процессора iMX6ULL имеет до семи периферийных функций, таблица соответствия выводов процессора и разъемов модуля SK-iMX6ULL-MOD:

SK-IMX6ULL-MOD	CPU	SK-IMX6ULL-MOD	CPU
X1.1 – 3.3V	3.3V, выход	X2.1 - GND	GND
X1.2 – USB_OTG2_VBUS	U12 - USB_OTG2_VBUS	X2.2 - LCD_HS	D9 - LCD_HSYNC
X1.3 - USB_OTG1_CHD	U16 - USB_OTG1_CHD	X2.3 - LCD_VS	C9 - LCD_VSYNC
X1.4 - USB_OTG1_VBUS	T12 - USB_OTG1_VBUS	X2.4 - GPIO4-10_PWM3	B4 – NAND_ALE
X1.5 - GND	GND	X2.5 - CSI_MCLK	F5 - CSI_MCLK
X1.6 - JTAG_TDI_OUT0_PWM6	N16 - JTAG_TDI	X2.6 - CSI_PIXCLK	E5 - CSI_PIXCLK
X1.7 - JTAG_TCK_OUT1_PWM7	M14 - JTAG_TCK	X2.7 - CSI_VSYNC	F2 - CSI_VSYNC
X1.8 – 3.3V	3.3V, выход	X2.8 - CSI_HSYNC	F3 - CSI_HSYNC
X1.9 - GPIO_8	N17 - GPIO_8	X2.9 - CSI_DATA0	E4 - CSI_DATA0
X1.10 - GPIO_0	K13 - GPIO_0	X2.10 - CSI_DATA1	E3 - CSI_DATA1
X1.11 - GPIO_5	M17 - GPIO_5	X2.11 - CSI_DATA2	E2 - CSI_DATA2
X1.12 - GPIO_4	M16 - GPIO_4	X2.12 - CSI_DATA3	E1 - CSI_DATA3
X1.13 - GPIO_3	L17 - GPIO_3	X2.13 - CSI_DATA4	D4 - CSI_DATA4
X1.14 - GPIO_9	M15 - GPIO_9	X2.14 - CSI_DATA5	D3 - CSI_DATA5
X1.15 - GPIO_2	L14 - GPIO_2	X2.15 - CSI_DATA6	D2 - CSI_DATA6
X1.16 - GPIO_1	L15 - GPIO_1	X2.16 - CSI_DATA7	D1 - CSI_DATA7
X1.17 - UART3_RX	H16 - UART3_RX	X2.17 - GPIO4-13_UART3RX	C5 - NAND_CEO
X1.18 - UART3_TX	H17 - UART3_TX	X2.18 - GPIO4-12_UART3TX	A3 – NAND_READY
X1.19 - UART3_RTS	G14 - UART3_RTS	X2.19 - UART5_RX	G14 - UART5_RX
X1.20 - UART3_CTS	H15 - UART3_CTS	X2.20 - UART5_TX	F17 - UART5_TX
X1.21 - UART4_RX	G16 - UART4_RX	X2.21 - GND	GND
X1.22 - UART4_TX	G17 - UART4_TX	X2.22 - ENET2_RX_DATA0	C17 - ENET2_RX_DATA0
X1.23 - BOOT	T10 – BOOT_MODE0	X2.23 - ENET2_RX_DATA1	C16 - ENET2_RX_DATA0
X1.24 - RESET	P8 - POR	X2.24 - ENET2_RX_ER	D16 - ENET2_RX_ER
X1.25 - ONOFF	R8 - ONOFF	X2.25 - ENET2_RX_EN	B17 - ENET2_RX_EN
X1.26 - VBAT	P12 – VDD_SNV5_IN	X2.26 - ENET2_TX_DATA0	A15 - ENET2_TX_DATA0
X1.27 – 5V	—	X2.27 - ENET2_TX_DATA1	A16 - ENET2_TX_DATA0
X1.28 – 5V	—	X2.28 - ENET2_TX_EN	B15 - ENET2_TX_EN
X1.29 – 5V	—	X2.29 - ENET2_TX_CLK	D17 - ENET2_TX_CLK
X1.30 – 5V	—	X2.30 - GND	GND
X1.31 – GND	GND	X2.31 - GND	GND
X1.32 – GND	GND	X2.32 - LCD_DATA13	B12 - LCD_DATA13
X1.33 – GND	GND	X2.33 - LCD_DATA18	A13 - LCD_DATA18
X1.34 - SD1_CLK	C1 - SD1_CLK	X2.34 - LCD_DATA08	B11 - LCD_DATA08
X1.35 - SD1_CMD	C2 - SD1_CMD	X2.35 - LCD_DATA14	A12 - LCD_DATA14
X1.36 - SD1_D3	A2 - SD1_D3	X2.36 - LCD_DATA09	A11 - LCD_DATA09
X1.37 - SD1_D2	B1 - SD1_D2	X2.37 - LCD_DATA06	A10 - LCD_DATA06
X1.38 - SD1_D1	B2 - SD1_D1	X2.38 - LCD_DATA05	B10 - LCD_DATA05
X1.39 - SD1_D0	B3 - SD1_D0	X2.39 - LCD_DATA16	C13 - LCD_DATA16
X1.40 - GND	GND	X2.40 - GND	GND
X1.41 - UART2_CTS	J15 - UART2_CTS	X2.41 - LCD_DATA23	B16 - LCD_DATA23
X1.42 - UART2_RTS	H14 - UART2_RTS	X2.42 - LCD_DATA21	B14 - LCD_DATA21
X1.43 - UART2_TX	J17 - UART2_TX	X2.43 - LCD_DATA20	C14 - LCD_DATA20
X1.44 - UART2_RX	J16 - UART2_RX	X2.44 - LCD_DATA22	A14 - LCD_DATA22

SK-iMX6ULL-MOD

X1.45 - UART1_CTS	K15 - UART1_CTS	X2.45 - LCD_DATA17	B13 - LCD_DATA17
X1.46 - UART1_RTS	J14 - UART1_RTS	X2.46 - LCD_DATA19	D14 - LCD_DATA19
X1.47 - UART1_TX	K14 - UART1_TX	X2.47 - LCD_DATA11	D12 - LCD_DATA11
X1.48 - UART1_RX	K16 - UART1_RX	X2.48 - LCD_DATA00	B9 - LCD_DATA00
X1.49 - GND	GND	X2.49 - GND	GND
X1.50 - ETH_RXN	Ethernet PHY RX “-”	X2.50 - LCD_CLK	A8 - LCD_CLK
X1.51 - ETH_RXP	Ethernet PHY RX “+”	X2.51 - LCD_DATA12	C12 - LCD_DATA12
X1.52 - ETH_TXN	Ethernet PHY TX “-”	X2.52 - LCD_DATA15	D13 - LCD_DATA15
X1.53 - ETH_TXP	Ethernet PHY TX “+”	X2.53 - LCD_DATA04	C10 - LCD_DATA04
X1.54 - GND	GND	X2.54 - LCD_DATA07	D11 - LCD_DATA07
X1.55 - USB_OTG1_DN	T15 - USB_OTG1_DN	X2.55 - LCD_DATA03	D10 - LCD_DATA03
X1.56 - USB_OTG1_DP	U15 - USB_OTG1_DP	X2.56 - LCD_EN	B8 - LCD_ENABLE
X1.57 - GND	GND	X2.57 - LCD_DATA01	A9 - LCD_DATA01
X1.58 - USB_OTG2_DN	T13 - USB_OTG2_DN	X2.58 - LCD_DATA10	E12 - LCD_DATA10
X1.59 - USB_OTG2_DP	U13 - USB_OTG2_DP	X2.59 - LCD_DATA02	E10 - LCD_DATA02
X1.60 - GND	GND	X2.60 - GND	GND

SK-iMX6ULL-MOD

СВЕДЕНИЯ ОБ ОРГАНИЗАЦИИ

www.starterkit.ru

info@starterkit.ru

Skype: starterkit.ru

Россия, г.Ижевск, ул.Телегина д.30

Тел.: (+7 3412) 478-448, +79226802173, +79226802174

www.otladka.com.ua

evodbg@gmail.com

Украина, г. Киев, ул. Желябова 8/4 офис 20

тел: 38-044-362-25-02

тел: 38-091-910-68-18

тел: 38-098-661-97-97

тел: 38-063-750-57-75