

Сборка образа OpenWrt для GW-08.

Обновление дерева исходных текстов.

Для использования нового патча необходимо сначала отменить изменения внесённые предыдущей версией патча командой:

```
patch -p1 -R <~/openwer-orangepi-zero.patch
```

Красным указан путь к файлу старого патча. После этого можно наложить новый патч командой

```
patch -p1 <~/openwrt-orangepi-zero-201804230634.patch
```

После этого достаточно выполнить команду начала компиляции прошивки. Система сборки openwrt автоматически внесёт все изменения.

Сборка прошивки с нуля.

В первую очередь нужно загрузить исходные тексты системы сборки прошивки openwrt из официального репозитория командой:

```
git clone git://github.com/openwrt/openwrt.git
```

Далее переходим в директорию с исходными текстами.

```
cd openwrt/
```

После чего необходимо перейти на коммит, в котором был разработан патч поддержки OrangePi Zero командой:

```
git checkout d13c7acd9e34c5d64a03d8860752b4d75a6b906e
```

Далее необходимо применить патч к исходным текстам openwrt. Команда подразумевает нахождение патча в домашней директории пользователя, если это не так, после < укажите свой путь к файлу.

```
patch -p1 <~/openwrt-orangepi-zero.patch
```

Стандартный шаг по включению дополнительных пакетов в систему сборки openwrt:

```
./scripts/feeds update -a  
./scripts/feeds install -a
```

Копируем конфигурационный файл для OrangePi Zero

```
cp board-configs/orangepi-zero.config ../config
```

Запускаем команду конфигурирования. Это необходимо, чтобы система сборки openwrt сформировала временные конфигурационные файлы в директории ./tmp (файлы конфигураций пакетов и целевых платформ сборки). Также на этом этапе можно внести дополнительные изменения в прошивку (выбрать необходимые пакеты для сборки).

```
make menuconfig
```

После этого запускаем непосредственно сборку. Ключ '-j' позволяет указать количество параллельных процессов для ускорения компиляции, обычно указывают значение на 1 меньше количества ядер процессора.

```
make -j 5
```

В результате в директории ./bin/targets/sunxi/cortexa7/ должны быть файлы:

```
openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin
```

Образ прошивки для записи на SPI-NOR (MTD) флеш во время производства оборудования (байт-

в-байт образ флешки для записи программатором или из окружения openwrt, которое было загружено с SD-карты. Запись из окружения openwrt выполняется командой `mtd write`). Образ включает в себя загрузчик UBoot, DTB-раздел и образ прошивки (раздел `firmware/sysupgrade` образ) с ядром linux и файловой системой.

```
openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-sysupgrade.bin
```

Образ прошивки для записи на SPI-NOR (MTD) флеш для обновления из окружения openwrt командой `sysupgrade` с сохранением конфигурации оборудования. Не содержит в себе DTB-образ, который можно обновить вручную.

```
openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-sdcard-ng.img
```

Образ прошивки для SD-карт. Формат совпадает с официально поддерживаемым производителем плат Sunxi OrangePi Zero.

```
openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-uboot.bin
```

Образ UBoot для обновления только загрузчика вручную, как правило, не требуется, но возможны случаи дальнейшего обновления только UBoot.

```
openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-dtb.bin
```

Образ DTB (файл OpenFirmware конфигурации оборудования, описывающий внутреннюю структуру компонентов системы) для обновления вручную.

Обновление и запись прошивки.

Первичная запись.

Для использования прошивки необходимо записать образ на внешний накопитель (SD-карта) командой

```
dd if=openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-sdcard-ng.img of=/dev/sdX
```

Вместо `/dev/sdX` укажите путь к устройству с SD-картой.

После этого необходимо загрузить устройство с openwrt на SD-карте. Далее скачиваем файл прошивки для SPI-NOR (MTD) на устройство. Для этого можно воспользоваться FTP- или HTTP-сервером на локальном компьютере, или воспользоваться SCP-протоколом.

Загрузка с FTP- или HTTP-сервера.

Положите файл прошивки в директорию, к которой даёт доступ сервер. Для FTP используйте пользователя `anonymous` для обмена файлами по этому протоколу.

Допустим, IP-адрес сервера `192.168.1.100`. Скачиваем файл в директорию `/tmp` на устройстве командой (для FTP):

```
wget ftp://192.168.1.100/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin -O /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin
```

Или командой (для HTTP):

```
wget http://192.168.1.100/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin -O /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin
```

Далее нам необходимо записать прошивку на SPI-NOR (MTD) флеш командой

```
mtd write /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin factory
```

Важно, команда `mtd` из состава openwrt перед записью проверяет название разделов в файле `/proc/mtd`, последний аргумент в команде указывает название раздела для записи, т.к. блочное устройство может отличаться при каждой загрузке.

После выполнения этой команды устройство готово для работы без SD-карты.

Загрузка с использованием SCP.

Допустим, IP-адрес устройства после загрузки 192.168.1.1.

Для выгрузки файла на устройство необходимо на компьютере выполнить команду

```
scp openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin  
root@192.168.1.1:/tmp/
```

После этого необходимо выполнить команду записи прошивки на флеш с самого устройства.

```
mtd write /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-factory.bin  
factory
```

Обновление.

Для полного обновления прошивки, в случае изменений в DTB (dts) файлах и/или загрузчике UBoot при этом сохранение конфигурации устройства после обновления НЕ требуется - необходимо воспользоваться командами из предыдущего раздела.

Если же требуется сохранение конфигурации при обновлении нужно записать новые образы UBoot/DTB вручную.

Для обновления UBoot раздела скачайте файл openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-uboot.bin на устройство в директорию /tmp и выполните команду

```
mtd write /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-uboot.bin u-boot
```

Для обновления DTB-раздела скачайте файл openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-dtb.bin на устройство в директорию /tmp и выполните команду

```
mtd write /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-dtb.bin dtb
```

После этого можно выполнить стандартное обновление остальной прошивки командой sysupgrade, об это ниже.

Для обновления прошивки загрузите файл прошивки openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-sysupgrade.bin на устройство с использованием wget (FTP или HTTP протокол) или scp, как в предыдущем разделе (первичная запись) и выполните команду

```
sysupgrade -c /tmp/openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-  
sysupgrade.bin
```

Команда sysupgrade обновит прошивку с сохранением текущей конфигурации устройства. Ключ '-c' указывает команде найти все изменения в директории /etc/ на устройстве и сохранить их для следующей загрузки (по-умолчанию сохраняются только файлы из списка /etc/sysupgrade.conf, в основном конфигурационные файлы в /etc/config/).

Параметры OpenWrt по-умолчанию.

IP-адрес, протокол удалённого управления, пользователь и пароль.

При первой загрузке прошивки устройство настроено получать IP-адрес автоматически (DHCP). Выданный адрес можно посмотреть на DHCP-сервер в файле выделенных lease (или в базе данных, если она используется).

Удалённый доступ к устройству предоставляется по протоколу SSH.

Имя пользователя: root

Пароль: не требуется.

Конфигурация LoraWan.

Файл конфигурации packet-forwarder находится в директории /etc, полный путь к файлу:

```
/etc/global_conf.json
```

Управление службой производится стандартным механизмом прошивки openwrt. Для запуска

демона вручную необходимо выполнить на устройстве команду:

```
/etc/init.d/lora_pkt_fwd start
```

Для включения службы в автозагрузку (автоматический запуск при включении устройства) необходимо выполнить команду:

```
/etc/init.d/lora_pkt_fwd enable
```

При запуске службы в первый раз после включения скрипт `/etc/init.d/lora_pkt_fwd` производит сброс устройства SX1301 используя GPIO. Конфигурация номеров выводов GPIO для сброса находится в файле

```
/etc/config/lora
```

Описание патча.

Список файлов в дереве исходных текстов openwrt, которые затрагивает патч, с кратким описанием.

./board-configs/orangepi-zero.config	Конфигурационный файл системы сборки openwrt для создания прошивки для устройства OrangePi Zero с поддержкой работы со SPI-NOR флеш-накопителем и включенным пакетом LoraWan
./package/boot/uboot-sunxi/Makefile	Файл сборки загрузчика для OrangePi Zero. Добавлена целевая сборка UBoot для OrangePi Zero.
./package/boot/uboot-sunxi/patches/999-uboot-sunxi-add-spidev-boot.patch	Патч для UBoot добавляющий поддержку SPI-NOR флеш в загрузчике для возможности дальнейшей передачи загрузки ядру linux.
./target/linux/sunxi/patches-4.14/203-ARM-dts-sun8i-activate-SPI-spidev-and-flash.patch.diff	Патч для ядра linux добавляющий поддержку SPI-NOR флеш-накопителя и включающий второй SPI интерфейс для работы из userspace окружения прошивки через устройство <code>/dev/spidev1.1</code>
./target/linux/sunxi/config-4.14	Конфигурация ядра linux. Включена поддержка SPI-интерфейса и доопределение разделов на SPI-NOR накопителе.
./target/linux/sunxi/image/gen_sunxi_sdcard_img-ng.sh	Файл автоматического формирования "сырого" образа прошивки для записи на SD-карту.
./target/linux/sunxi/image/Makefile	Основной файл сборки результирующих файлов прошивки для целевых платформ Sunxi. Добавлена ссылка на makefile для сборки прошивки OrangePi Zero
./target/linux/sunxi/image/cortex-a7-orangepi-zero-ng.mk	Файл, в котором описаны шаги по формированию файлов прошивки.
./target/linux/sunxi/base-files/lib/upgrade/platform.sh	Скрипт, использующийся при обновлении прошивки из окружения openwrt на самом устройстве командой sysupgrade. Позволяет удалённо обновлять прошивку с сохранением конфигурационных файлов.
./toolchain/musl/patches/101-added_qsort_r.patch	Файл добавляющий поддержку функции <code>qsort_r</code> в библиотеку стандартных функций musl. В связи с тем, что пакет lora-gateway использует эту функцию, а она есть только в библиотеках GLIBC или uClibc, добавление <code>qsort_r</code>

	обязательно для нормально работы пакетов LoraWan.
<code>./package/lora/lora-gateway/</code>	Директория с файлами для сборки библиотеки LoraWan.
<code>./package/lora/packet-forwarder/</code>	Пакет сетевого взаимодействия LoraWan.
<code>./files/</code>	Директория с файлами конфигурации прошивки по-умолчанию. При сборке образа файлы из этой директории копируются в образ корневой файловой системы openwrt.

Детальное описание изменений.

Происхождение изменений в патче.

Все изменения в патче производились вручную, кроме перечисленных ниже пакетов.

UBoot

Для поддержки загрузки прошивки с SPI-NOR флеш-накопителя в загрузчике UBoot были портированы драйвера из официального репозитория UBoot компании производителя Sunxi по адресу:

<https://github.com/StephanvanSchaik/u-boot/tree/sunxi-spi>

Ветка, в которой ведётся разработка драйвера называется sunxi-spi. В патче внесены изменения для поддержки pinmux (конфигурирование выводов ножек процессора по назначению), непосредственно драйвер sunxi_spi и добавлены необходимые правки в DTB для общей конфигурации разделов на накопителе.

LoraWan

Пакеты для поддержки устройств Lora были портированы из репозитория:

<https://github.com/xueliu/lora-feed>

Пакеты были переработаны для локальной сборки, обновление исходных текстов производится вручную, без привязки к внешним ресурсам. Патчи из репозитория применены к исходным текстам пакетов вручную, патчи перемещены в директорию applied-patches-to-src и их применение больше не требуется.

Для выбора пакетов LoraWan в menuconfig системы сборки openwrt создан раздел Lora Packages, а не стандартные пути Libraries, Network. Это сделано для удобства нахождения пакетов в конфигурационном меню.

Разделы SPI-NOR флеш-накопителя.

Openwrt позволяет динамически изменять структуру флеш-накопителей при загрузке системы. Это сделано для удобства работы основных подсистем, требовательных к размеру файлов, техническими ограничениями работы с NOR-флеш-накопителями (отсутствие wear-leveling и невозможностью работы как с обычным блочным устройством).

Основной файловой системой в среде OpenWrt для ядра linux считается squashfs, вторичная файловая система, используемая OpenWRT - JFFS2. Первая не поддерживает режим записи (статическая) и формируется в момент сборки прошивки, имеет лучшие характеристики по параметрам сжатия дерева директорий и файлов. Вторая файловая система поддерживает запись и wear-leveling для уменьшения износа флеш-накопителей, но из-за этого обладает значительно худшими характеристиками сжатия.

Базовым набором разделов на флеш-накопителе считается раздел uboot для загрузчика и раздел firmware для основной системы.

Для простоты портирования и совместимости всех образов прошивок (между SD-картами и SPI-NOR) требуется отдельный раздел для DTB на SPI-NOR, он назван dtb.

Базовые разделы упомянуты в соответствующих dts-файлах для UBoot и ядра linux.

При каждой загрузке Openwrt происходит два разделения (split) разделов (partition). Первое разделение происходит по признаку имени раздела с прошивкой (в разделе firmware подразумевается наличие ядра linux упакованного в особый формат образов ulmage, создаётся утилитой mkimage из состава UBoot, и наличие squashfs), в котором считывается размер ядра linux, в конце которого по границе erasesize (минимальный объём стирания) флеш-накопителя ищется признак (MAGIC комбинация байт) начала squashfs. Если файловая система найдена, в системе динамически создаются разделы kernel и rootfs. Раздел kernel точно соответствует границам ядра linux, а rootfs занимает всю оставшуюся область.

Второе разделение происходит при появлении раздела rootfs, в конце которого ядро ищет (по границе erasesize) указатель на начало JFFS2. Указатель помещается при сборке прошивки и выравнивается по размеру блока накопителя. После чего ядро создаёт ещё один раздел с названием rootfs_data.

При дальнейшей загрузке системы происходит монтирование доступных разделов в необходимом порядке. За это отвечает исполняемый файл mount_root. При самой первой загрузке устройства происходит создание JFFS2 файловой системы на разделе rootfs_data.

Для привычной работы в среде openwrt с возможностью записи и удаления файлов используется подсистема ядра OverlayFS. Эта файловая система позволяет эмулировать поведение стандартных систем оперируя только одной файловой системой доступной для записи, все изменения в структуре сохраняются в ней.

Для удобства обновления и первичной записи на флеш также представлен раздел factory, охватывающий весь объём SPI-NOR флеш-накопителя (8МиБ).

Наглядное представление разделов на SPI-NOR представлено ниже:

Доступно на всех этапах загрузки. Вписаны статически в DTS файлах UBoot и ядра linux	factory (8 192KiB) (0x0000000 0x8000000)				
	u - boot (512KiB) (0x0000000 - 0x0800000)	dtb (64KiB) (0x0800000 - 0x0100000)	firmware (7 616KiB) (0x0900000 - 0x7700000)		
			kernel (динамический)	rootfs (динамический)	
			kernel (динамический)	rootfs (динамический)	rootfs_data (динамический)