Building OpenWrt image for GW-08 from sources.

Updating sources tree.

If you use this guide first time omit this step and move to "Building image from sources".

For using this patch first you need to discard changes from older patch by following command:

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

Red is path for older patch. Now you can apply fresh new patch:

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

After that you can run make command to start cross compilation process. OpenWrt build tools will apply all changes automatically.

Building image from sources.

First you need to download source code of openwrt build tools from official repository by using following command:

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

Next move to path with sources:

```
cd openwrt/
```

Next step you need to move for commit which was used while developing OrangePi zero dev board support patch:

```
git checkout d13c7acd9e34c5d64a03d8860752b4d75a6b906e
```

Apply patch. Red-lit text is the path for patch. Enter your way to the file:

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

Add additional packets:

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

Copy configuration file for OrangePi Zero board:

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

Open configuration window. This is required to create temporary configuration files in ./tmp directory. Don't change anything. Just exit that menu.

```
make menuconfig
```

Start compilation process. '-j' key sets amount of parallel processes used to speed-up compilation process. Typical value is amount of cores on you PC minus one

```
make -j 5
```

After compilation process finishes in ./bin/targets/sunxi/cortexa7/ directory you can see following files:

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

Image to write in empty SPI-NOR flash IC using programmer or from openwrt environment booted from microSD card. This image contains Uboot, DTB, image kernel and filesystem.

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

Image to write in SPI-NOR flash IC from openwrt environment using sysupgrade command. Stores configuration. Doesn't contain DTB, which can be updated manually.

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

Image to boot from microSD card. Compatible with officially supported Sunxi OrangePi Zero.

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

UBoot image to upgrade only it. Typically it is doesn't required. But can be used for following Uboot updating.

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

DTB tree explaining system configuration for updating it manually.

Flashing image.

Flashing first time.

First you need to write image to microSD card:

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

Enter you address of SD card in your PC instead of /dev/sdX

Next you need to plug microSD card to OrangePi Zero and boot from it. Then you need to download image to burn onboard SPI-NOR flash IC. You can do it using FTP-/HTTP- server on your local PC or using SCP.

Loading from FTP- or HTTP-server.

Put image file to the path of your FTP/HTTP server. For FTP use anonymous user for file loading using this protocol.

For example, your server's IP address is 192.168.1.100. Download image file in /tmp directory of your device using following command (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
```

Or using this way (for 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
```

Next you need to write image in SPI-NOR (MTD) flash IC:

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

Mtd command checks the partition name in the /proc/mtd file before flashing. The last argument in the command specifies the name of partition to flash, because the block device may differ at each load.

Now your device ready to boot without microSD card. Unplug SD card and reboot the device. IP address may be changed after boot so check it again please.

Loading using SCP.

Lets consider that IP address of your device 192.168.1.1.

Copy image file via scp:

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

Next you need to write image in SPI-NOR (MTD) flash IC.

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

Image update.

For full image upgrade in a case of changes in DTB tree or Uboot while OS configuration storing doesn't required – use way described above.

If you need to store configuration after image updating you need to rewrite DTB/Uboot manually.

To update Uboot download `openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-uboot.bin` file on your device (via SCP or using FTP/HTTP server) in /tmp directory and run following:

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

To update DTB-tree download `openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-dtb.bin` file on your device (via SCP or using FTP/HTTP server) in /tmp directory and run following:

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

After that you can upgrade other parts of image using sysupgrade command as described below.

Download `openwrt-sunxi-cortexa7-sun8i-h2-plus-orangepi-zero-squashfs-mtd-sysupgrade.bin` file on your device (via SCP or using FTP/HTTP server) and run following:

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

Image will be updated without any changes of configuration. Key '-c' tells the command to check for all modification in /etc directory and store them (by default files to be stored are in /etc/sysupgrade.comf list).

OpenWrt params by default.

IP-address, default access way, user and password.

On first boot device configured to get IP-address using DHCP. You can check for it on DHCP server. Remote access available using SSH. Username to access is **root**. No password required.

LoraWan configuration.

Config file of `packet-forwarder` library placed in /etc. Path to access it:

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

Control mechanism typical for openwrt. To launch daemon manually run following:

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

To autorun on every device power up run following:

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

/etc/init.d/lora_pkt_fwd script resets SX1301 via gpio on every power up. GPIO pinout can be found in following file:

Patch description.

List of source files modified by patch with short description:

./board-configs/orangepi-zero.config	Config file with supporting of OrangePi Zero. Booting from SPI-NOR flash IC. LoraWan packets added by default.
./package/boot/uboot-sunxi/Makefile	Uboot makefile for OrangePi Zero.
./package/boot/uboot-sunxi/patches/999-uboot-sunxi-add-spidev-boot.patch	Patch for Uboot to support linux kernel booting from SPI-NOR flash IC.
./target/linux/sunxi/patches-4.14/203-ARM-dts-sun8i-activate-SPI-spidev-and-flash.patch.diff	Linux kernel patch with support of SPI-NOR flash IC. Second SPI interface added to communicate with SX1308 (/dev/spidev1.1).
./target/linux/sunxi/config-4.14	Linux kernel configuration. Added support for SPI interface and SPI-NOR IC partitions checking.
./target/linux/sunxi/image/gen_sunxi_sdcard_img-ng.sh	Auto configuration of raw image to write on microSD card.
./target/linux/sunxi/image/Makefile	Main makefile. Added link to OrangePi's makefile.Description of steps to compile resulted images.
./target/linux/sunxi/image/cortex-a7-orangepi-zero-ng.mk	Description of steps to compile resulted images.
./target/linux/sunxi/base-files/lib/upgrade/platform.sh	Script used for image upgrade using sysupgrade. Allows remotely upgrade image without overwriting configuration.
./toolchain/musl/patches/101-added_qsort_r.patch	`qsort_r` function added to `musl` library. As lora-gateway packet uses this function but this function presents only in GLIBC or uClibc adding this function mandatory.
./package/lora/lora-gateway/	LoRaWAN packets directory.
./package/lora/packet-forwarder/	Semtech's `packet-forwarder` packet.
./files/	Default config files directory.

Detailed description of patch.

Sources used by patch.

All changes in patch were maid manually except following.

UBoot

To support booting from SPI-NOR flash IC Sunxi's official repository was used:

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

Branch that was used is `sunxi-spi`. Pin mux (gpio configuration) supporting added in patch. DTB tree changes for flash partitions configuration.

LoraWan

Lorawan packets was ported from following repository:
https://github.com/xueliu/lora-feed
Changes for local compilation. Manual sources update without using external resources. Patch from repository was applied manually. Patches were moved in applied-patches-to-src directory. You don"t need to use it anymore.

`Lora Packages` chapter was added to menuconfig to simplify navigation.

SPI-NOR flash IC partitions.

Openwrt allows you to dynamically change the structure of flash IC when the system boots. This is done for the convenience of the main subsystems, file-size requirements, technical limitations of working with NOR flash drives (no wear-leveling and inability to work as with a conventional block device).

Main filesystem in OpenWrt for linux kernel is `squashfs`, secondary filesystem, used by OpenWRT - `JFFS2`. `Squashfs` doesn't support write mode (static) and is formed at the time of image compiling, has better compression parameters. `JFFS2` supports write operations and wear-leveling to reduce the wear of flash drives, but because of this has significantly worse compression characteristics.

The basic set of partitions of the flash drive is the uboot partition and firmware partition for the main system. For simplicity of porting and compatibility of all firmware images (for SD-card and for SPI-NOR IC) a separate section for DTB on SPI-NOR is required. It is called DTB.

The base partitions are mentioned in the corresponding dts files for UBoot and the linux kernel.

With each download of Openwrt, there are two splits of partitions. The first partition is based on the name of the partition with the firmware (in the firmware section it is assumed that there is a linux kernel packed in a special image format `uImage`, created by the `mkimage` utility from `UBoot`, and `squashfs`), in which the linux kernel size is read, at the end of which the erasesize the minimum amount of erasure) of the flash drive is searched for a sign (MAGIC combination of bytes) of the beginning of `squashfs`. If the file system is found, the system dynamically creates the `kernel` and `rootfs` partitions. The `kernel` section corresponds exactly to the boundaries of the linux kernel, and `rootfs` occupies the rest of the region.
The second partition occurs when the `rootfs` section appears, at the end of which the kernel looks for (at the erasesize boundary) a pointer to the start of `JFFS2`. The pointer is placed during image compilation and is aligned to the size of the flash block size. After that, the kernel creates another partition called `rootfs_data`.
When the system is subsequently booted, the available partitions are mounted in the required order. The executable file `mount_root` is responsible for this. At the very first loading of the device, the JFFS2 file system is created on the `rootfs_data` partition.

For the usual work in an openwrt environment with the ability to write and delete files, the subsystem of the `OverlayFS` kernel is used. This file system allows you to emulate the behavior of standard systems using only one file system available for writing, all changes in the structure are stored in it.

For the convenience of updating and initial recording on the flash, the factory section is also presented, covering the entire volume of the SPI-NOR flash drive (8MiB).

A visual representation of the sections on SPI-NOR is presented below:

	factory (8 192KiB) (0x000000 0x800000)				
Available at all steps of the booting. Statically specified in DTS files UBoot and linux kernels	u-boot (512KiB) (0x000000 - 0x080000)	dtb (64KiB) (0x080000 - 0x010000)	firmware (7 616KiB) (0x090000 - 0x770000)		
The first step of splitting when the kernel is loaded			kernel (динамический)	rootfs (динамический)	
The second and final step of the splitting when the kernel is loaded.			kernel (динамический)	rootfs (динамический)	rootfs_data (динамический)