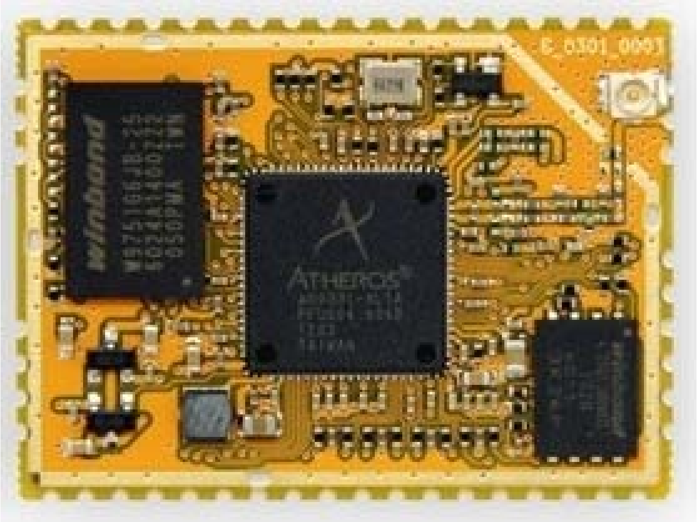
Carambola 2 manual

Table of Contents

Introduction

8devices Carambola 2 is a member of Carambola wireless modules family and is based on Qualcomm/Authors AR9331 SoC. Carambola 2 is a surface mountable, single sided, Wi-Fi enabled Linux module, featuring the lowest power consumption in the industry. Module comes bundled with an open source development board for a quick start of your new Internet of Things era project. 8devices is providing OpenWRT linux distribution source code with necessary patches on GitHub https://github.com/8devices and is supported by our growing community on http://www.8devices.com/community forum.



Carambola 2 specification overview

CPU	AR9331, 400 MHz
Memory	16 MB SPI Flash with 64 MB DDR2 RAM
Frequency	2.4 GHz
Max output power	21 dBm
Wireless standard	802.11 bgn
Antenna (port)	U.FL connector
Power supply	3.3 V, power consumption 0.5 W
Interfaces	USB host/slave, serial port, 2 x Ethernet, I2S, SLIC, SPDIF, 23 x GPIO
Size	28 by 38 mm
Software	OpenWRT

Antennas:

12dBi omni-directional antenna	
Model:	2400A-12
Gain-dBi	12
Manufacturer:	Anteny24

3dBi built-in ceramic antenna	
Model:	ACA-5036-A2-CC-S
Gain-dBi	**3**
Manufacturer:	INPAQ TECHNOLOGY CO., LTD.

Where to start

Install packages to your PC

```
sudo apt-get install subversion g++ ncurses-term zlib1g-dev gawk flex patch
openssh-server minicom picocom tftp tftpd gettext libncurses5 libncurses5-
dev unzip quilt git git-doc git-gui
```

Get latest sources

```
git clone https://github.com/8devices/carambola carambola cd carambola
```

Get older revision

For some reason you might wish to get older source revision. To see available tags execute these commands:

```
git tag -l
```

You should see something like this

```
carambola/v1.0 carambola/v1.1 carambola/v1.1-rc1 carambola/v2.0
carambola/v2.0-rc1 carambola/v2.1 carambola/v2.1-rc1 carambola/v2.1-rc2
carambola/v2.2-rc1 carambola/v2.3-rc1
```

In order to get older version, type this command (but this should be rare case)

```
git checkout carambola/v1.0
```

Stable version

At this point (2012-11-11) stable version is **carambola/v2.3-rc4**. To get this version type command:

```
git checkout carambola/v2.3-rc4
```

More packages

Edit file:

```
vi feeds.conf.default
src-git packages git://github.com/8devices/packages.git #src-git luci
https://github.com/8devices/luci.git src-git luci src-git
git://nbd.name/luci.git #src-svn packages
svn://svn.openwrt.org/openwrt/packages src-svn xwrt http://x-
wrt.googlecode.com/svn/trunk/package #src-svn luci
http://svn.luci.subsignal.org/luci/trunk/contrib/package #src-svn phone
svn://svn.openwrt.org/openwrt/feeds/phone #src-svn efl
svn://svn.openwrt.org/openwrt/feeds/efl #src-svn xorg
svn://svn.openwrt.org/openwrt/feeds/xorg #src-svn desktop
svn://svn.openwrt.org/openwrt/feeds/desktop #src-svn xfce
svn://svn.openwrt.org/openwrt/feeds/xfce #src-svn lxde
svn://svn.openwrt.org/openwrt/feeds/lxde #src-link custom
/usr/src/openwrt/custom-feed
```

Type

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

I suppose you would also like to add luci.

Build image

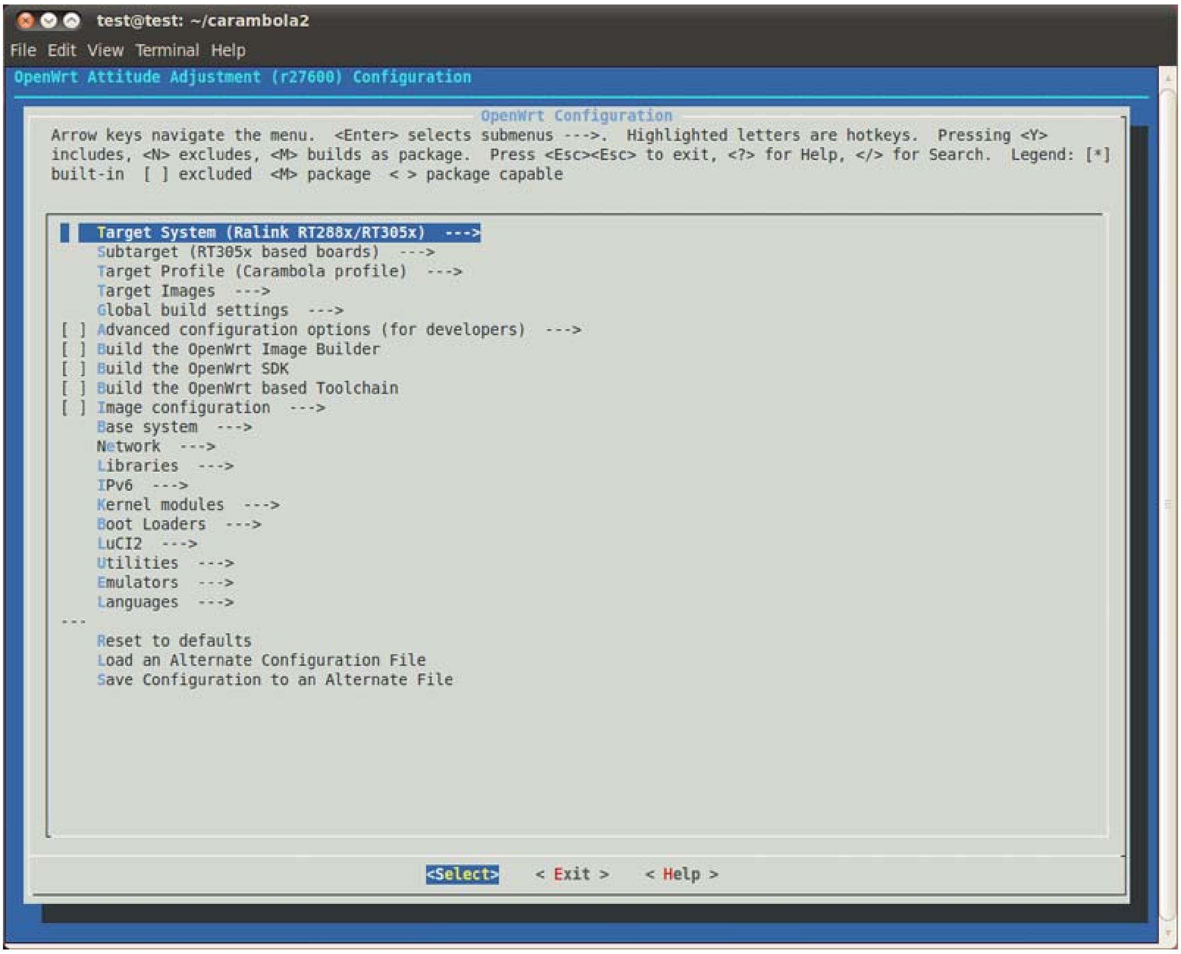
If you wish you can change kernel settings, but at this time skip this step.

```
make kernel_menuconfig
```

Type

```
make menuconfig
```

You should see configuration window like this:



```
OpenWrt Attitude Adjustment (r27600) Configuration
                                  OpenWrt Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus --->.  Highlighted letters are hotkeys.  Pressing <Y>
  includes, <N> excludes, <M> builds as package.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*]
  built-in  [ ] excluded  <M> package  < > package capable

      Target System (Ralink RT288x/RT305x)  --->
      Subtarget (RT305x based boards)  --->
      Target Profile (Carambola profile)  --->
      Target Images  --->
      Global build settings  --->
  [ ] Advanced configuration options (for developers)  --->
  [ ] Build the OpenWrt Image Builder
  [ ] Build the OpenWrt SDK
  [ ] Build the OpenWrt based Toolchain
  [ ] Image configuration  --->
      Base system  --->
      Network  --->
      Libraries  --->
      IPv6  --->
      Kernel modules  --->
      Boot Loaders  --->
      LuCI2  --->
      Utilities  --->
      Emulators  --->
      Languages  --->
  ---
      Reset to defaults
      Load an Alternate Configuration File
      Save Configuration to an Alternate File

                          <Select>    < Exit >    < Help >
```

Let's leave everything untouched for now and and exit.

Type

```
make
```

for standard building procedure or

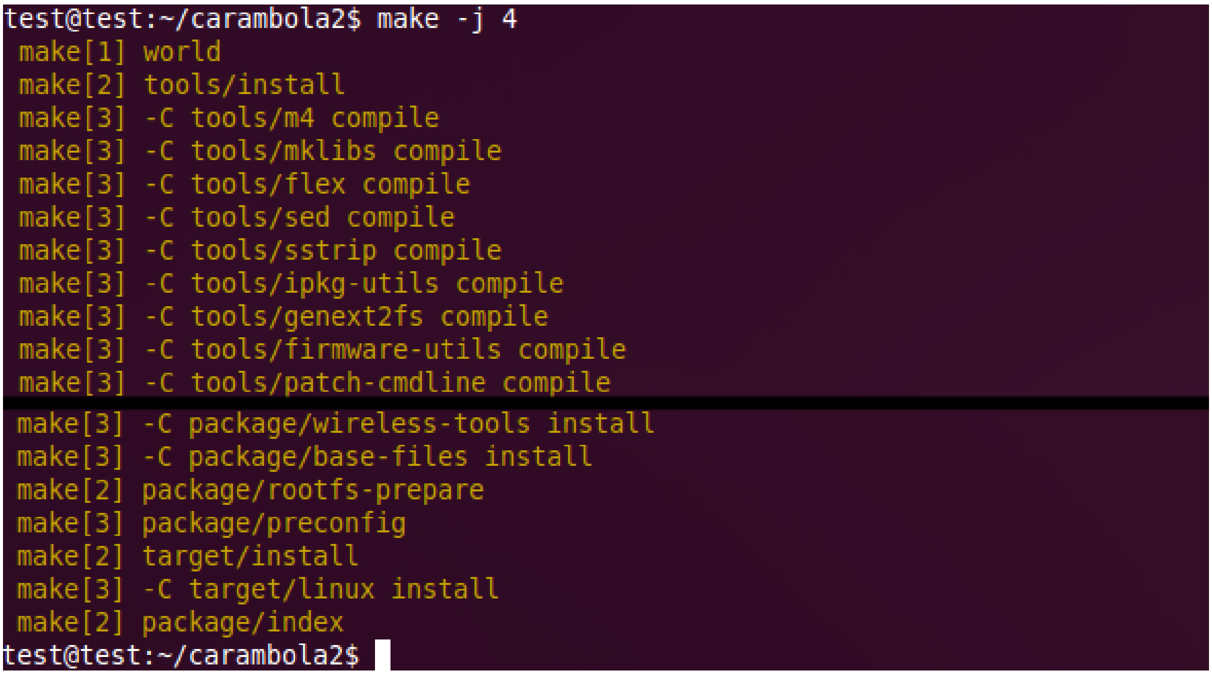
```
make -j 2
```

if you want to use more than 1 thread and you have more than 1 CPU core on your system. This process takes long time (it took ~1 hour on 4 core machine) or

```
make V=99
```

if you need or want to see more details.

You should see build log like this:



```
test@test:~/carambola2$ make -j 4
 make[1] world
 make[2] tools/install
 make[3] -C tools/m4 compile
 make[3] -C tools/mklibs compile
 make[3] -C tools/flex compile
 make[3] -C tools/sed compile
 make[3] -C tools/sstrip compile
 make[3] -C tools/ipkg-utils compile
 make[3] -C tools/genext2fs compile
 make[3] -C tools/firmware-utils compile
 make[3] -C tools/patch-cmdline compile
 make[3] -C package/wireless-tools install
 make[3] -C package/base-files install
 make[2] package/rootfs-prepare
 make[3] package/preconfig
 make[2] target/install
 make[3] -C target/linux install
 make[2] package/index
test@test:~/carambola2$
```

Or if you feel really advanced, you can get email notification upon completing build proces. For this you need to install and configure email client. After you are done, type this long command:

```
date > /tmp/text.txt; make -j 2 >> /tmp/text.txt; date >> /tmp/text.txt;
mail -s "done" email_where_to_send_report@server.com < /tmp/text.txt
```

If you wish to observe current progress on computer, type in another console

```
tail -f /tmp/text.txt
```

Connect using serial cable

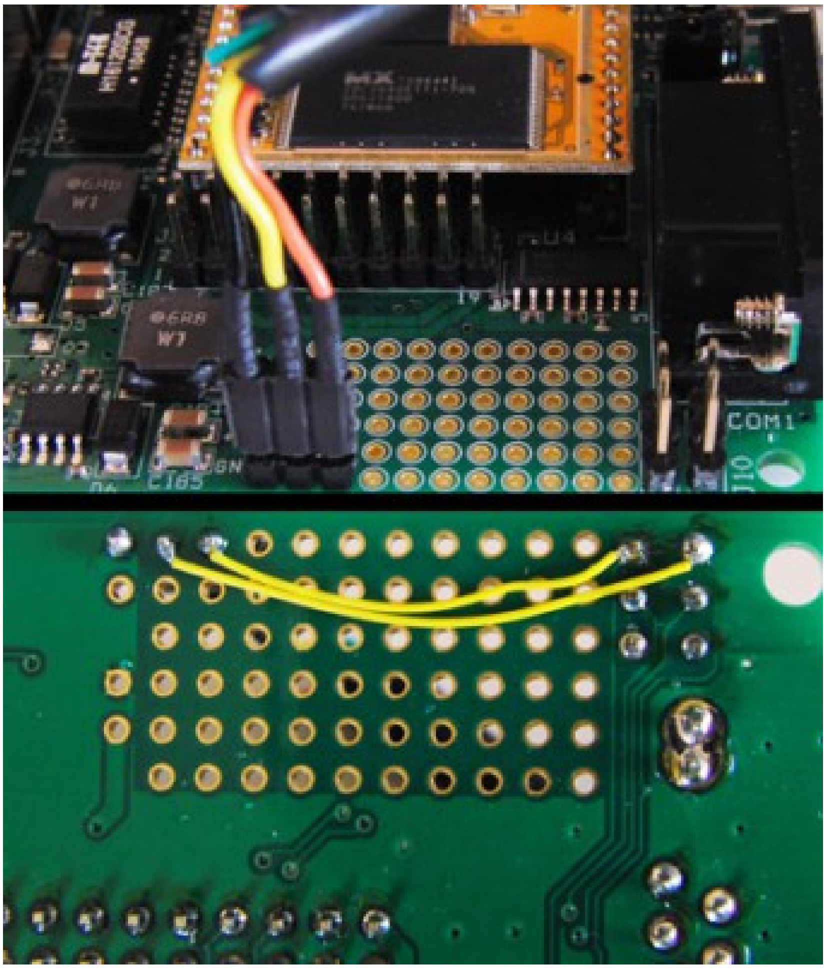
RS-232 level

It is best to connect to carambola using standard RS-232 male cable. Just plug cable to carambola, set 115200 8N1 settings without hardware flow control and power up your carambola.

TTL 3.3V level

However, if you have TTL level USB to serial converter, you should do some soldering.

Remove J9 and J10 jumpers and connect FTDI 3.3V TTL serial to USB cable. J9.1 – RX(input), J10.1 – TX(output). I used FTDI 3.3V USB to serial converter.



Software

You can use any terminal program you wish to communicate.

Start terminal

Connection settings

- 115200 8N1, no hardware flow control.

Windows

I recommend using V-term32 (no need to install just unzip and run) or TeraTerm program.

Linux minicom

TODO

Linux picocom

```
picocom -f n -p n -b 115200 -i -r -l /dev/ttyUSB0
picocom -b 9600 /dev/ttyUSB0
```

- Picocom man
- To exit, press and hold CTRL, A, Q.

Other Linux uart settings and useful commands

If you do not have installed `stty` you should get it by typing

```
opkg install coreutils-stty
```

To set options:

```
stty -F /dev/ttyUSB0 cs8 9600 ignbrk -brkint -icrnl -imaxbel -opost -onlcr
-isig -icanon -iexten -echo -echoe -echok -echoctl -echoke noflsh -ixon -
crtscts
```

To get options:

```
stty -a -F /dev/ttyUSB0
```

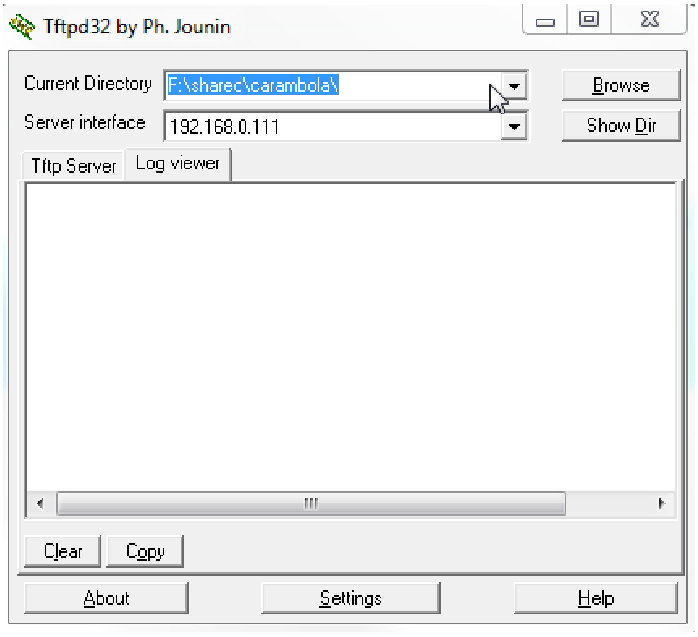
To read from device

```
tail -f /dev/ttyUSB0
```

Set up TFTP server

Windows

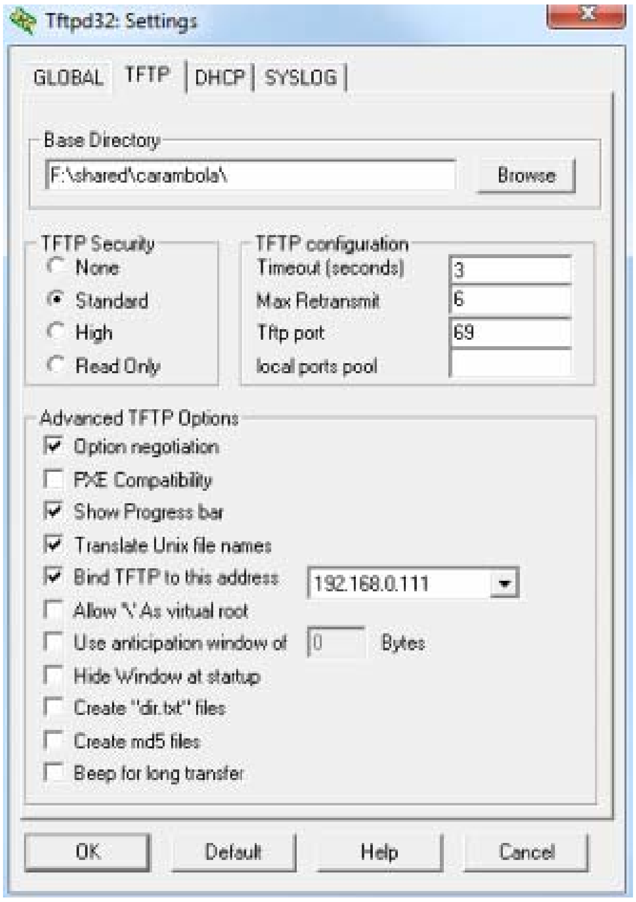
You can set up TFTP server very quickly using TFTP32 sofware. It is free.


Tftpd32 by Ph. Jounin

Current Directory: F:\shared\carambola\

Server interface: 192.168.0.111

Enable TFTP server. Your settings should look like:


Tftpd32: Settings

Base Directory: F:\shared\carambola\

TFTP Security
- ○ None
- ● Standard
- ○ High
- ○ Read Only

TFTP configuration
- Timeout (seconds): 3
- Max Retransmit: 6
- Tftp port: 69
- local ports pool:

Advanced TFTP Options
- ☑ Option negotiation
- □ PXE Compatibility
- ☑ Show Progress bar
- ☑ Translate Unix file names
- ☑ Bind TFTP to this address 192.168.0.111
- □ Allow '\' As virtual root
- □ Use anticipation window of 0 Bytes
- □ Hide Window at startup
- □ Create "dir.txt" files
- □ Create md5 files
- □ Beep for long transfer

Linux

If you wish, you can use linux machine to do the same task.

```
sudo apt-get install xinetd tftpd tftp
sudo touch /etc/xinetd.d/tftp
sudo nano /etc/xinetd.d/tftp
```

Paste this code:

```
service tftp {  protocol = udp  port = 69  socket_type = dgram  wait = yes
user = nobody  server = /usr/sbin/in.tftpd  server_args = /home/tftpboot
disable = no }
sudo mkdir /home/tftpboot
sudo chmod -R 777 /home/tftpboot
sudo chown -R nobody /home/tftpboot
sudo service xinetd restart
```

Upload new firmware to Carambola 2

This topic covers few methods how to upgrade firmware on Carambola. Some methods require only serial connection, some only LAN, some LAN and Serial. Also you might need to install/setup other software to perform these tasks.

sysupgrade

sysupgrade is best ever way to upgrade firmware. For this to work you must be connected to Carambola terminal. It does not matter using Serial, LAN ir Wireless connection.

mtd

mtd method is similar to sysupgrade. Use any one you like.

Serial connection only

Using serial cable is most dangerous one. There many things that can go wrong. Ensure you have good serial cable (cheaper usb-serial adapters don't work), rock solid power supply and steel nerves :)

TFTP

TFTP method to upgrade firmware requires you serial connection to Carambola and configured TFTP server over LAN. Wireless connection does not work.

Upgrading bootloader

If for some reason you have older bootloader and wish to upgrade, you must proceed here. It is even more dangerous than upgrading firmware using serial connection only. If something fails, there are big chances you render your module useless.

Web interface / LUCI

This method is recommended for novices. Connect Carambola web interface, browse for firmware upgrade, select file and press upgrade.

Terminal programs for various OS, settings and usage examples

terminals

Carambola2 automated flashing (linux version)

Pyton based flash utility for Carambola2

Change default configuration

Change Network parameters

Edit:

```
carambola/target/linux/ramips/base-files/etc/config/network
```

Cahnge default banner (welcome screen)

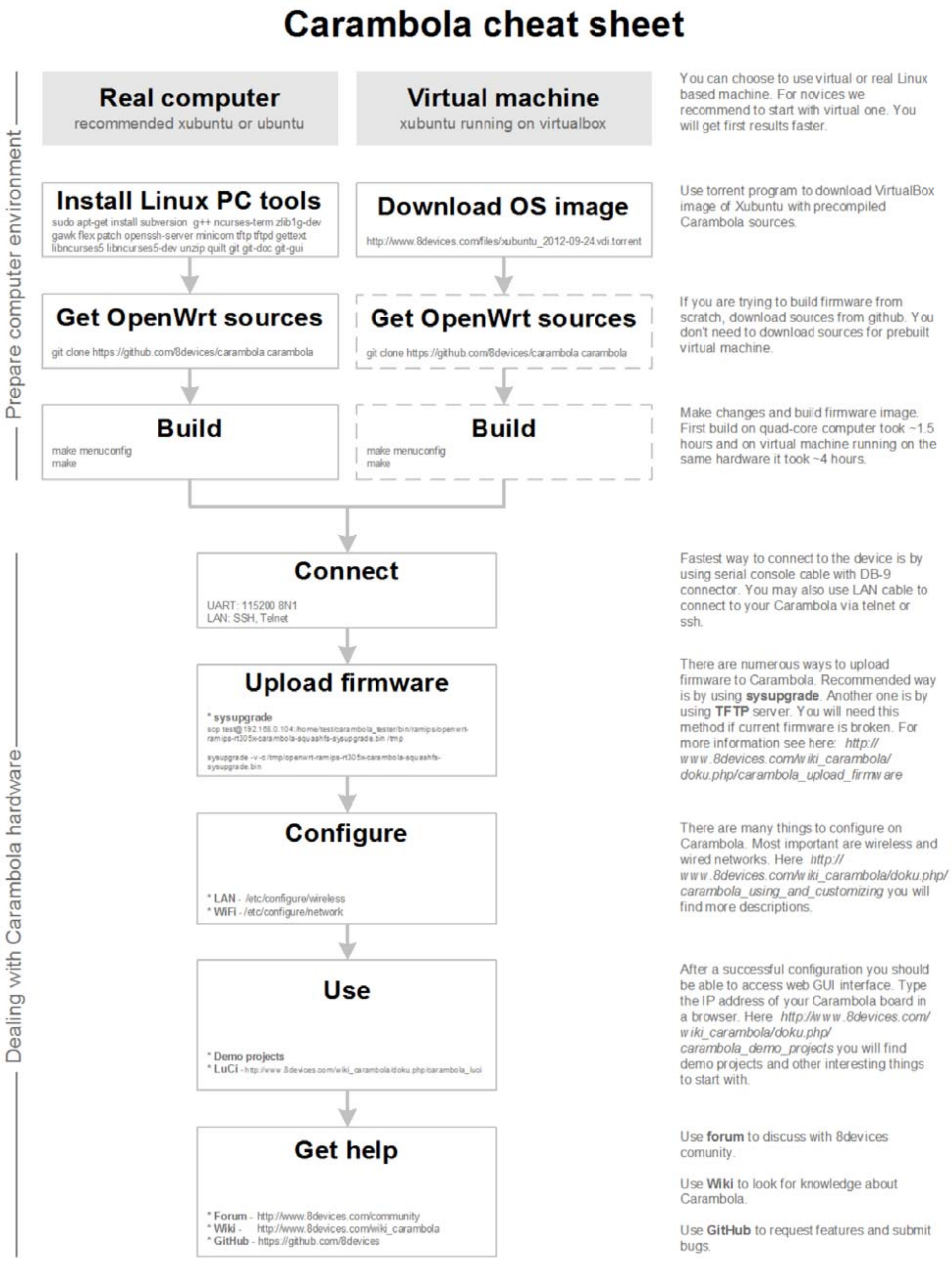
Change welcome screen example

Edit:

```
carambola/package/base-files/files/etc/banner
```

Appe

If you are



Carambola cheat sheet

Prepare computer environment

Real computer
recommended xubuntu or ubuntu

Virtual machine
xubuntu running on virtualbox

You can choose to use virtual or real Linux based machine. For novices we recommend to start with virtual one. You will get first results faster.

Install Linux PC tools
sudo apt-get install subversion g++ ncurses-term zlib1g-dev gawk flex patch openssh-server minicom tftp tftpd gettext libncurses5 libncurses5-dev unzip quilt git git-doc git-gui

Download OS image
http://www.8devices.com/files/xubuntu_2012-09-24.vdi.torrent

Use torrent program to download VirtualBox image of Xubuntu with precompiled Carambola sources.

Get OpenWrt sources
git clone https://github.com/8devices/carambola carambola

Get OpenWrt sources
git clone https://github.com/8devices/carambola carambola

If you are trying to build firmware from scratch, download sources from github. You don't need to download sources for prebuilt virtual machine.

Build
make menuconfig
make

Build
make menuconfig
make

Make changes and build firmware image. First build on quad-core computer took ~1.5 hours and on virtual machine running on the same hardware it took ~4 hours.

Dealing with Carambola hardware

Connect
UART: 115200 8N1
LAN: SSH, Telnet

Fastest way to connect to the device is by using serial console cable with DB-9 connector. You may also use LAN cable to connect to your Carambola via telnet or ssh.

Upload firmware
* **sysupgrade**
scp test@192.168.0.104:/home/test/carambola_tester/bin/ramips/openwrt-ramips-rt305x-carambola-squashfs-sysupgrade.bin /tmp

sysupgrade -v -c /tmp/openwrt-ramips-rt305x-carambola-squashfs-sysupgrade.bin

There are numerous ways to upload firmware to Carambola. Recommended way is by using **sysupgrade**. Another one is by using **TFTP** server. You will need this method if current firmware is broken. For more information see here: *http://www.8devices.com/wiki_carambola/doku.php/carambola_upload_firmware*

Configure
* **LAN** - /etc/configure/wireless
* **WiFi** - /etc/configure/network

There are many things to configure on Carambola. Most important are wireless and wired networks. Here *http://www.8devices.com/wiki_carambola/doku.php/carambola_using_and_customizing* you will find more descriptions.

Use
* **Demo projects**
* **LuCi** - http://www.8devices.com/wiki_carambola/doku.php/carambola_luci

After a successful configuration you should be able to access web GUI interface. Type the IP address of your Carambola board in a browser. Here *http://www.8devices.com/wiki_carambola/doku.php/carambola_demo_projects* you will find demo projects and other interesting things to start with.

Get help
* **Forum** - http://www.8devices.com/community
* **Wiki** - http://www.8devices.com/wiki_carambola
* **GitHub** - https://github.com/8devices

Use **forum** to discuss with 8devices comunity.

Use **Wiki** to look for knowledge about Carambola.

Use **GitHub** to request features and submit bugs.

FCC Warning Statement:

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.
This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:
(1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.
This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following
measures:
Reorient or relocate the receiving antenna.
Increase the separation between the equipment and receiver.
Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
Consult the dealer or an experienced radio/TV technician for help.

FCC Radiation Exposure Statement

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator include antenna and your body.

Labeling requirement

The Host device of OEM integrator must be labeled with ‘Contains FCC ID: Z9W-CM2’

IC warning

This device complies with Industry Canada licence-exempt RSS standard(s).
Operation is subject to the following two conditions:
(1) This device may not cause interference and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence.L'exploitation est autorisée aux deux conditions suivantes :
(1) l'appareil ne doit pas produire de brouillage, et
(2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

Under Industry Canada regulations, this radio transmitter may only operate using an antenna of a type and maximum (or lesser) gain approved for the transmitter by Industry Canada. To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the equivalent isotropically radiated power (e.i.r.p.) is not more than that necessary for successful communication.
This radio transmitter (identify the device by certification number, or model number if Category II) has been approved by Industry Canada to operate with the antenna types listed below with the maximum permissible gain and required antenna impedance for each antenna type indicated. Antenna types not included in this list, having a gain greater than the maximum gain indicated for that type, are strictly prohibited for use with this device.