NH7020 User Manual

Revision	Release date	Reason	Contents
1.0	2/20/2020	First Edition	

Attention: Observe precautions for handling electrostatic sensitive devices.

1. Product Overview

The NH7020 is a networked software defined radio (SDR) that provides reliability and fault-tolerance for deployment in large-scale and distributed wireless systems. The NH7020 device simplifies control and management of a network of radios by introducing the unique capability to remotely perform tasks such as updating software, rebooting, factory resetting, self-testing, host PC/ARM debugging and monitoring system health.

The NH7020 is one of the highest channel density devices in the SDR market, offering two RX and two TX channels in a half-wide RU form factor. The RF front end used AD9361 transceivers. Each channel provides up to 56MHz of instantaneous bandwidth and covers an extended frequency range from 70 MHz to 6 GHz. The baseband processor uses the Xilinx Zynq-7020 SoC to deliver a user programmable FPGA for real-time and low latency processing and a dual-core ARM CPU for stand-alone operation. Support for, 1 GbE enables high throughput IQ streaming to a host PC or FPGA co-processor. A flexible synchronization architecture with support for clock reference, PPS time reference.

The open-source LIBIIO API reduce software development effort and integrate with a variety of industry-standard tools such as GNU Radio. Users can rapidly prototype and reliably deploy designs for a variety of SDR applications such as wireless testbeds, remote radio heads, spectrum monitoring, and more.

2. Features

Hardware：
RF ADC/DAC：AD9361
RF Range：70MHZ~6GHZ
RF Bandwidth：61.4MSPS
Ref Clock：VCTCXO 40MHZ (With 16bit DAC 0.2PPM max)+-15ppm
Power Output (CW): <3 dBm
SOC：Zynq7020
FPGA：85k
RAM：1GB
FLASH：32MB
USB：2.0
Ethernet：1000Mb
DC IN：5V +-0.5V-2A
Dimensions：75mm*102mm

Software：
OS：Linux
Interface：USB/1 GbE
API:libiio

3. Application
    1. Base station transceiver
       FM Broadcast

    2. GNSS Receiver
       https://gnss-sdr.org/

    3. GPS Simulation
       https://github.com/osqzss/gps-sdr-sim/

    4. GSM/3G Base Station
       http://openbts.org

    5. LTE Base Station
       https://www.srslte.com/

4. Package list

    1. Mainboard (no:B20191021001) 1PCS
    2. Micro USB CABLE (no:L20191021UC01) 1PCS
    3. 1GbE CABLE (no:L20191021NC01) 1PCS
    4. USB DC 5V Charger (no:P20191021DC01) 1PCS
    5. USB TO DC CABLE (no:L20191021DC02) 1PCS
    6. 2G/LTE antenna (no:A20191021L001) 1PCS
    7. GPS antenna (no:A20191021G002) 1PCS
    8. Up 3GHz antenna (no:A20191021L003) 1PCS

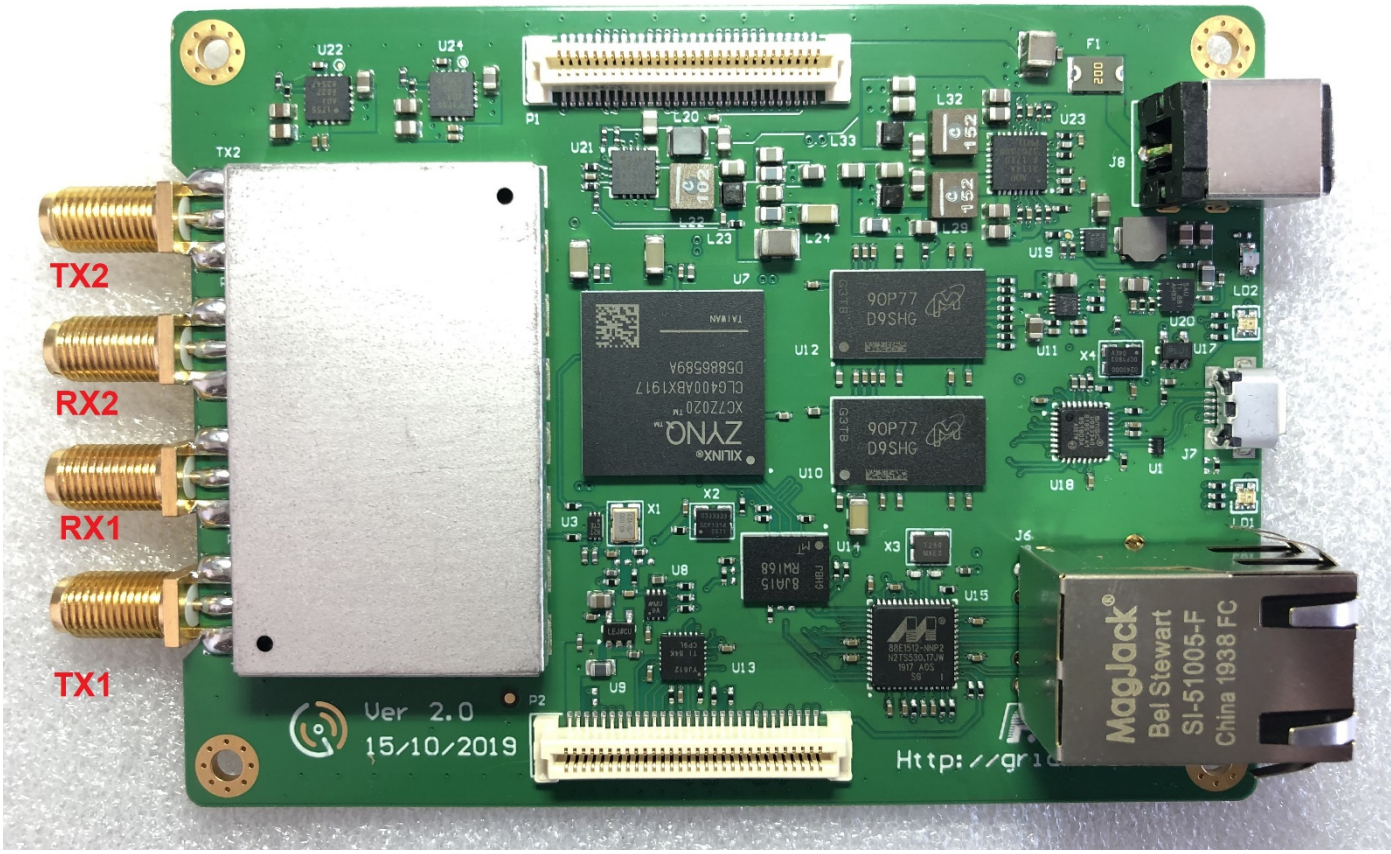
5. Getting started

1) Assembly

Connect L20191021DC02 to P20191021DC01.

Connect L20191021NC01 between B20191021001 and computer.

Connect A20191021L001/ A20191021G002 to B20191021001 RF PORT RX1 and TX1.



2) Power Up

Plug in P20191021DC01, after B20191021001 red led out, two blue (LD1/2) LED bright, System Running.

3) Configure network adapter

The NH7020 default IP address is 192.168.2.1, so you need set to the same network segment, and avoid network conflicts.

4) Install The Software

On the Windows (OS):

Install the USB driver PlutoSDR-M2k-USB-Drivers:

https://github.com/analogdevicesinc/plutosdr-m2k-drivers-win/releases

unzip the SDRSharp portable (including NH7020 plugs)：

SDRWIN.COM:

http://sdrwin.com/downloads/sdrsharp-x86_nh7020.zip

On the Ubuntu (Linux):

Gnuradio:

git clone --recursive https://github.com/gnuradio/gnuradio.git

```
cd gnuradio
git checkout maint-3.7
mkdir build && cd build
cmake ../
make
sudo make install
```

SoapySDR:

```
git clone https://github.com/pothosware/SoapySDR.git
cd SoapySDR
mkdir build && cd build
cmake ../
make
sudo make install
```

SoapyNH7020:

Dependencies:LIBIIO,libad9361.

```
git clone https://github.com/gridrf/SoapyNH7020.git
cd SoapyNH7020
mkdir build && cd build
cmake ../
make
sudo make install
```

gr-osmosdr：

patch the CMakeLists.txt

append after line 25:

set (CMAKE_CXX_STANDARD 11)

gr-iio: https://wiki.analog.com/resources/tools-software/linux-software/gnuradio

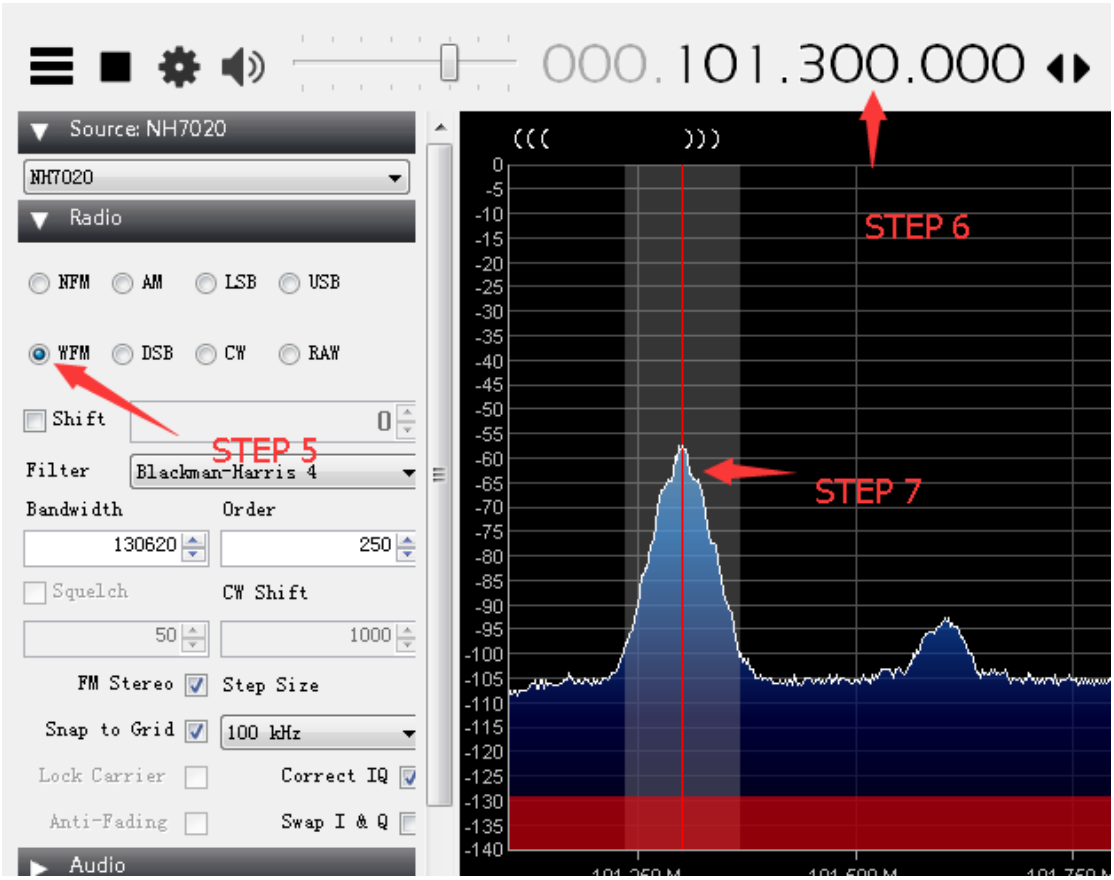
5) spectrum analysis

Open The SDRSharp software：

Selected “source” to ”NH7020” click configure button, and click connect button.

When STEP 4 button clicked System begin transport to the sdrsharp.

Change the demodulator mode to WFM and adjustment frequency and click the High Peak signal, computer will play the local FM Radio.



If you want change to the other frequency, just follow the STEP6.

Gnuradio:

Insert the gr-iio project files directory iio-examples, use the gnuradio load fmradio.grc and click running button you can play FM without GUI.

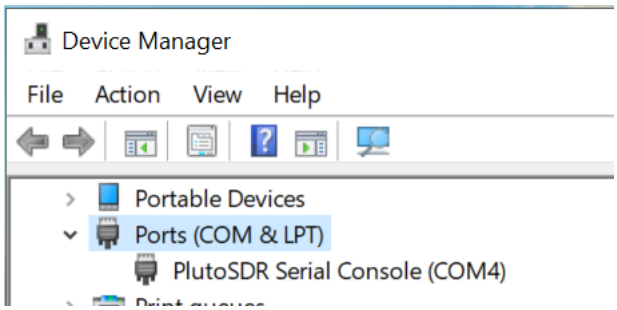
6) Test Transmitters

On the UBUNTU Install the gr-osmosdr running shell ./osmocom_siggen this software will sending IQ data through NH7020 you can see it at your RF instruments.

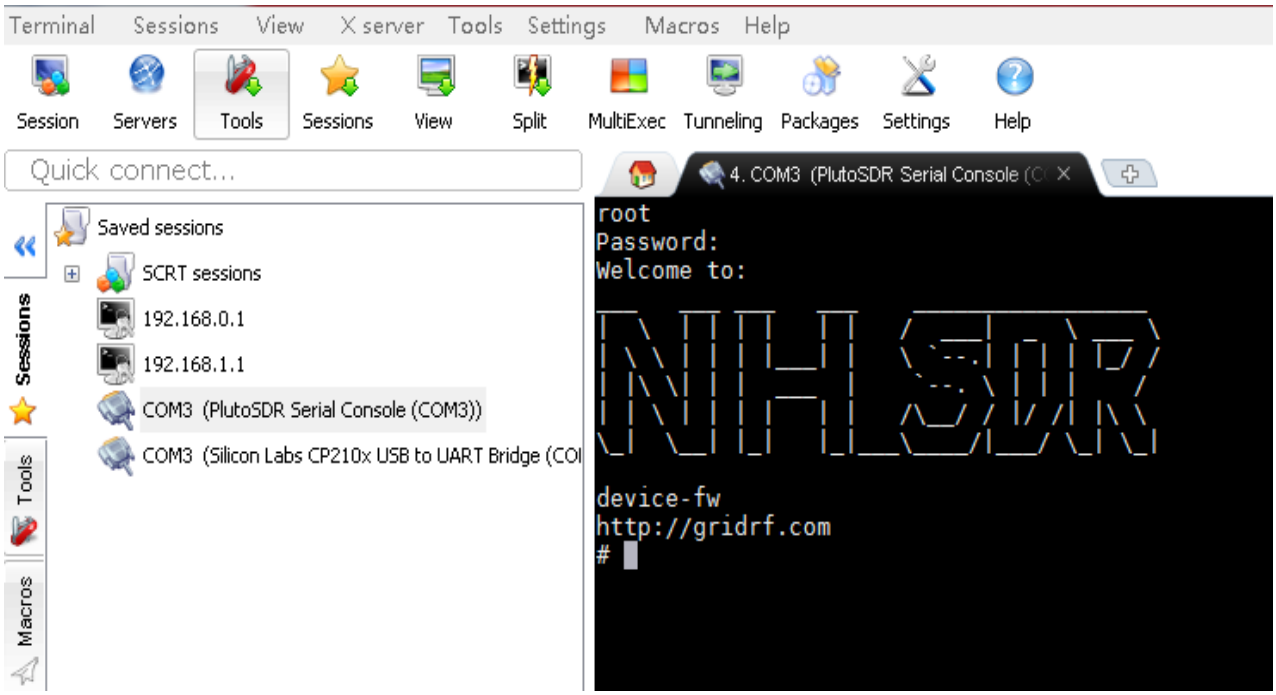
7) Embedded System

Login to NH7020:

Plug in the L20191021UC01 between NH7020 and computer,computer device manager will discover the Serial Port Device seeing like this:



Using the Serial tools software connect to the serial port, login with name root and password is gridrf.



8) SD mount (require expand board):

mount /dev/mmcblk0p1 /mnt

9) built-in app test

the Iio_rf_swap is a built-in software, this software capability to receive an radio band and transmit to the other radio band, the antennas use RX1 and TX1.

Example:
Iio_rf_swap –i 100 –t 650 –s 10

Parameters:
-i receive frequency in MHZ, the 100 mean 100MHZ.
-t transmit frequency in MHZ.
-s rf bandwith in MSPS, 10 mean 10MSPS.

When software start running, you can seeing it on the other NH7020 with sdrsharp.

6. Develop software
   1) LIBIIO
      https://wiki.analog.com/resources/tools-software/linux-software/libiio

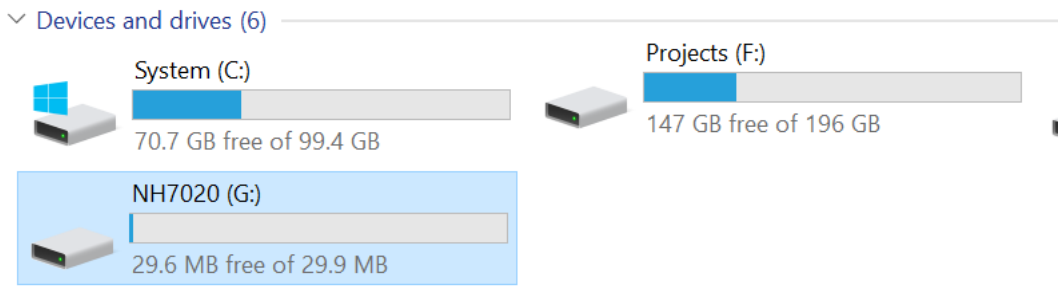
   2) Making the firmware
      https://github.com/gridrf/nh7020
      this is the NH7020 all firmware source code, after make inside build will generate .BIN,.DFU,.FRM.

   3) Custom software
      In the source directory buildroot/package/iio_rf_swap is an example.

7. Custom Develop(require expand board)
   1) schematics
      NDA require,TBD

   2) HDL Project
      Using the schematics create your own Vivado Projects, they're nothing to talk about.
      However we provide an example:
      http://sdrwin.com/downloads/NH7020_HDL_MSK_800MHZ_EXAMPLE.zip
      This example use the JTAG download to the NH7020 FPGA will sending MSK signal at 800MHZ directly, is a pure HDL control the AD9631 example, should be make some project easily.

8. Update firmware
   When NH7020 running, plug in L20191021UC01 to computer, in this pc will show the NH7020 device:



Copy file nh7020.frm into the NH7020 disk driver and right click Eject menu, wait led not flicker, system will auto reboot.
Attention: when system update proceed don't let power down.

9. Service support
   Email: billue@163.com

10. About this Manual
