**Upgrade the Jetson Orin Nano to 1GB NVMe:**

Download the latest Jetson Linux release package and sample file system for your Jetson developer kit from https://developer.nvidia.com/linux-tegra Notes: Jetson Linux 35.3.1 was tested

Enter the following commands to untar the files and assemble the rootfs:

$ tar xf ${L4T\_RELEASE\_PACKAGE}

$ cd Linux\_for\_Tegra/rootfs/

$ sudo tar xpf ../../${SAMPLE\_FS\_PACKAGE}

$ cd ..

$ sudo ./apply\_binaries.sh

Put your Jetson developer kit into Force Recovery Mode

Using 'lsusb' to check, 'Bus <bbb> Device <ddd>: ID 0955: <nnnn> Nvidia Corp.' means Jetson is in Recovery Mode.

Make sure the usb device in host PC not allowed to stop:

sudo su

echo 2 | sudo tee /sys/bus/usb/devices/\*/power/autosuspend >/dev/null

echo on | sudo tee /sys/bus/usb/devices/\*/power/level >/dev/null

$ sudo ./tools/kernel\_flash/l4t\_initrd\_flash.sh --external-device nvme0n1p1 \

-c tools/kernel\_flash/flash\_l4t\_external.xml -p "-c bootloader/t186ref/cfg/flash\_t234\_qspi.xml" \

--showlogs --network usb0 jetson-orin-nano-devkit internal

Connect the DCA and Radar to Jetson including two serial ports and one ethernet with power on.

**Configure AWR1642BOOST:**

Ensure the AWR1642 device has pre-flashed image in it, if not users need to use TI Uniflash tool ( http://www.ti.com/tool/UNIFLASH) to flash the device. SOP2 jumper and SOP0 jumper should be connected for starting Flash Programming Mode. The image in the directory of mmwave\_SDK like mmwave\_sdk\_<ver>\ti\demo\<platform>\mmw\<platform>\_mmw\_demo.bin.

Downlaod the profile.cfg and Radar\_configuration.py from Github https://github.com/Ubiweb-lab/mmVital/tree/main/Radar2Jestson

Only SOP0 jumper should be connected for Functional Mode

sudo apt update

sudo apt install python3

pip3 install pyserial

sudo apt install nano

python3 Radar\_configuration.py

**Configure DCA1000:**

Download mmWave Studio from <http://www.ti.com/tool/MMWAVE-STUDIO>

Intall mmWave Studio on a windows PC

Extract 'SourceCode' folder from path of installed mmWave Studio like ' C:\ti\mmwave\_studio\_<ver>\mmWaveStudio\ReferenceCode\DCA1000.'

Install GCC: #sudo apt-get update #sudo apt-get install g++

Under the path of 'SourceCode' to run 'make' command

Configure network on Jetson as Ip: 192.168.33.30 Mask: 255.255.255.0 Gateway:192.168.33.180

Update LD\_LIBRARY\_PATH using ‘export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:$pwd’ command

Downlaod configFile.json from Github https://github.com/Ubiweb-lab/mmVital/tree/main/Radar2Jestson to the path of made 'SourceCode/Release'

Configure FPGA:

Under the path of made 'SourceCode/Release' run ./DCA1000EVM\_CLI\_Control fpga configFile.json

Configure record delay:

In the same path run ./DCA1000EVM\_CLI\_Control record configFile.json

Start the record:

run ./DCA1000EVM\_CLI\_Control start\_record configFile.json

stop the record:

./DCA1000EVM\_CLI\_Control stop\_record configFile.json

The raw data will be stored as a '.bin' file.

**Extract and analyse vital sign:**

Download the program from Github https://github.com/Ubiweb-lab/mmVital/tree/main/MATLAB2Python

Install library

Using 'python HR\_RR\_main.py' to run the program in the terminal of Jetson Orin Nano