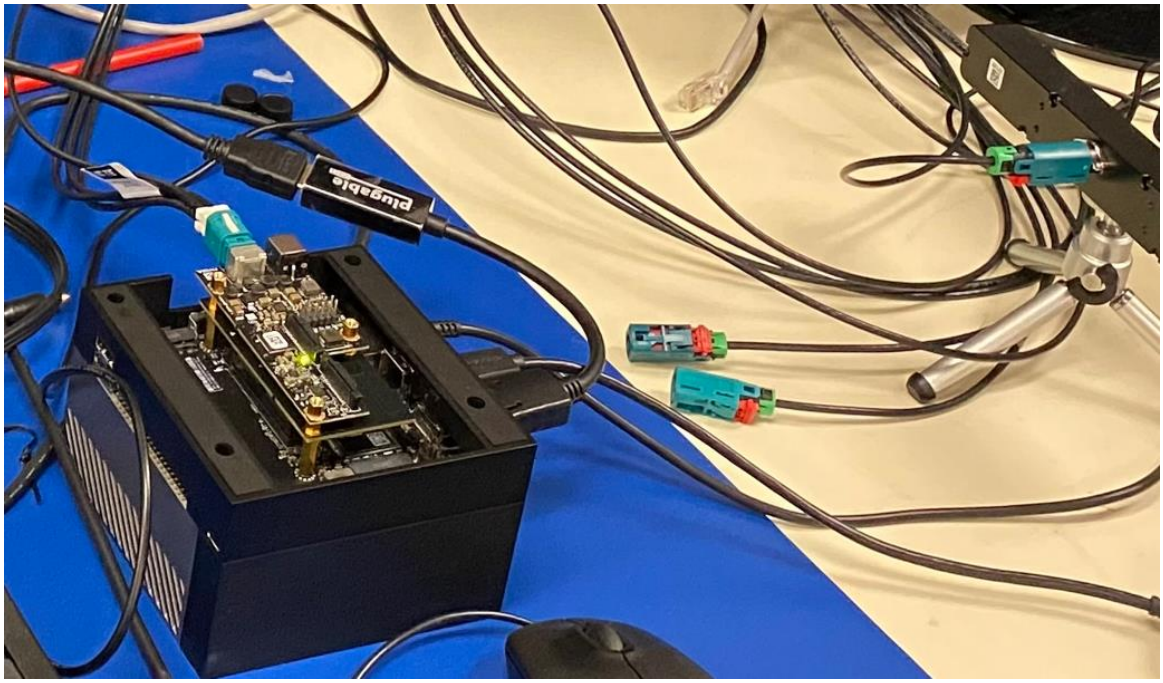
Overview

This driver is designed for LI-AR0234CS-STEREO-GMSL2-30 (Hawk) camera kit with Nvidia Jetson AGX Orin Developer kit.
This driver supports up to two LI-AR0234CS-STEREO-GMSL2-30 cameras.
This driver supports 1920x1200@30fps
This driver is based on R35.1 (Jetpack 5.0.2)

Download link

https://www.dropbox.com/sh/zz5o7rrh6y7sdnl/AACJhNaRVDru-udgD85k1b4Ma?dl=0

Platform	Camera
Nvidia Jetson AGX Orin Developer kit	1 ~ 2 x LI-AR0234CS-STEREO-GMSL2-30
Cable	**Adapter/Carrier Board**
1 x 4-in-1 Fakra cable (LI-FCB-4T-1-SS-2M-WP-A0)	1 x E3653-A03 1 x LI-JTX1-SUB-ADPT



Revision	SVN version	Release Date	Author	Tested by
20210414	TBD	04/14/2021	Xingxing Gu	Bin Liu

Updates

Revision	Description	Release Date
20210414	First Release based on R32.5	04/14/2021
20210511	Support NVS IMU driver	05/11/2021
20210705	Support autoContect Switch	07/05/2021
20210812	Support 32.6	08/12/2021
20210918	Support NVS	09/18/2021
20220301	Support 60fps	03/01/2022
20220310	Support 32.7 JP4.6.1	03/10/2022
20220316	Fix the one camera issue	03/16/2022
20220318	Fix flicker issue	03/18/2022
20220415	Support JP50	04/15/2022
20220526	Support JP50.1 add 960x600@120fps	05/26/2022
20220530	Support Orin	05/30/2022
20220715	Support IMU	07/15/2022
20220720	Support IMU timestamp	07/20/2022
20220822	Support 35.1	08/22/2022
20220922	Fix the ethernet issue when install the ko	09/22/2022
20220927	Fix the imu issue on Orin	09/27/2022
20221008	Fix the hawk 60fps issue.	10/08/2022
20221011	Fix the Xavier issue	10/11/2022
20221028	Remove second hawk imu interrupt pin	10/28/2022

Known bugs

1. If you only have one hawk camera, you can only connect on port 1

Setup Procedure 1/2

Hardware:

1. Nvidia Jetson AGX Orin Developer Kit x 1
2. E3653-A03 x 1
3. LI-JTX1-SUB-ADPT x 1
4. LI-AR0234CS-STEREO-GMSL2-30 x 1 ~ 2
5. 4-in-1 Fakra cable x 1 (LI-FCB-4T-1-SS-2M-WP-A0)
6. USB 3.0 Type-C cable x 1 (for flashing OS image)
7. Monitor with HDMI cable x 1
8. Keyboard and Mouse (with USB hub) x 1

Driver installation:

1. Download the R35.1.1 OS Image (from link below) to your Ubuntu OS on Intel x64 Host PC (we are using Ubuntu 20.04/18.04; virtual machine is fine) and follow the l4t_quick_start_guide to install the Jetpack to Orin.
   R35.1 OS Image: https://www.dropbox.com/sh/i0tzv4wwnqcp8y9/AAAt64szMoGf_3iJ0hk4uvrSa?dl=0

2. Use the sudo command to copy the tegra234-p3701-0000-p3737-0000.dtb to the /boot/dtb/kernel_ tegra234-p3701-0000-p3737-0000.dtb in your Orin platform, and use the sudo command again to copy the tegra194-p2888-0001-p2822-0000.dtb to the boot/dtb/kernel_tegra194-p2888-0001-p2822-0000.dtb such as the following:

```
sudo cp tegra234-p3701-0000-p3737-0000.dtb   /boot/dtb/kernel_tegra234-p3701-0000-p3737-0000.dtb
sudo cp tegra194-p2888-0001-p2822-0000.dtb   /boot/dtb/kernel_tegra194-p2888-0001-p2822-0000.dtb
```

3. Open a terminal and then use the commands below to remove the existing *. ko files (max96712.ko and nv_ar0234.ko) from the "/lib/modules/5.10.104-tegra/kernel/drivers/media/i2c/" directory first.

```
sudo rm /lib/modules/5.10.104-tegra/kernel/drivers/media/i2c/max96712.ko
sudo rm /lib/modules/5.10.104-tegra/kernel/drivers/media/i2c/nv_ar0234.ko
sudo rm /lib/module/5.10.104-tegra/kernel/drivers/iio/imu/bmi088/bmi088.ko
```

4. Reboot AGX Orin/Xavier kit, and then open a terminal and then perform the below commands.

```
sudo insmod  max96712.ko
sudo insmod  nv_ar0234.ko
sudo insmod bmi088.ko
```

Note:
1. The max96712.ko, nv-ar0234.ko, & bmi088.ko files are included in downloaded Driver's "Binaries" folder.
2. If you restart the Orin system, you must repeat these two commands again before using Argus Software, Gstreamer, or other camera capture applications to stream video images.

Setup Procedure 2/2

5. Then execute the command below to get live video images output.

nvgstcapture-1.0

Note:
Make sure the first camera is connected to Port 1,
then the second camera connected to Port 2, and so on.

6. Use Ctrl+C to close the video and then copy “camera_overrides.isp” from you downloaded driver’s “Binaries” directory to /var/nvidia/nvcam/settings directory and then use root (sudo) commands to change the file permission and its ownership such as the following:

sudo cp camera_overrides.isp /var/nvidia/nvcam/settings/
sudo chmod 664 /var/nvidia/nvcam/settings/camera_overrides.isp
sudo chown root:root /var/nvidia/nvcam/settings/camera_overrides.isp

```
nvidia@nvidia-desktop:~/Downloads$ sudo cp camera_overrides.isp /var/nvidia/nvca
m/settings/
nvidia@nvidia-desktop:~/Downloads$ sudo chmod 664 /var/nvidia/nvcam/settings/cam
era_overrides.isp
nvidia@nvidia-desktop:~/Downloads$ sudo chown root:root /var/nvidia/nvcam/settin
gs/camera_overrides.isp
nvidia@nvidia-desktop:~/Downloads$
```

7. Try "nvgstcapture-1.0” again. You should be able to see the image with better image quality

nvgstcapture-1.0

Run Camera

1. Argus software

Download the Multimedia package from the Dropbox link below and then copy it to the Orin system.
https://www.dropbox.com/s/qz0ey3ygvb6a6nj/jetson_multimedia_api.tar.gz?dl=0

Open a terminal, do
sudo apt-get install cmake
sudo apt-get install build-essential
sudo apt-get install pkg-config
sudo apt-get install libx11-dev
sudo apt-get install libgtk-3-dev
sudo apt-get install libexpat1-dev
sudo apt-get install libjpeg-dev
sudo apt-get install libgstreamer1.0-dev

Uncompress the downloaded zip file and then untar the file.
gunzip jetson_multimedia_api.tar.gz
tar -xvf jetson_multimedia_api.tar

Under jetson_multimedia_api/argus/cmake (cd jetson_multimedia_api/argus/cmake), do the following:
cmake ..
make
sudo make install
Do "argus_camera --device=0" to get the video.

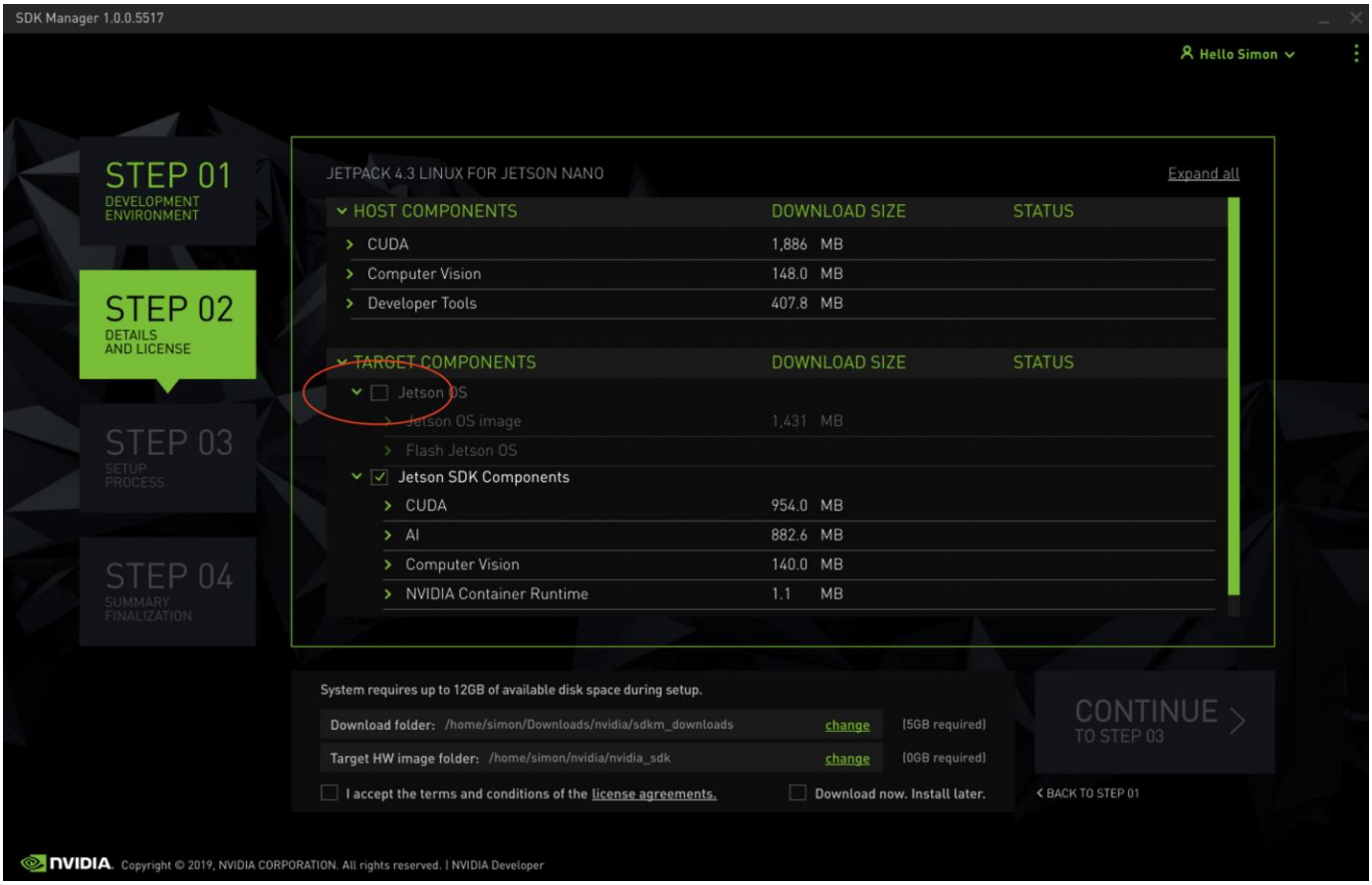
2. Gstreamer
gst-launch-1.0 nvarguscamerasrc sensor-id=0 ! 'video/x-raw(memory:NVMM), width=(int)1920, height=(int)1200, framerate=30/1' ! nvvidconv flip-method=0 ! 'video/x-raw, format=(string)I420' ! xvimagesink -e

3. v4l2-ctl capture raw
v4l2-ctl -V --set-fmt-video=width=1920,height=1200,pixelformat=RG10 --set-ctrl bypass_mode=0 --stream-mmap --stream-count=1 --stream-to=ar0234cs.raw -d /dev/video0

Note:
1) The 0 can be changed to 1 ~ 3 to run other cameras.
   Cable 1 ---- video0, Cable 2 ---- video1
   Cable 3 ---- video2, Cable 4 ---- video3
2) Please use the below commands to install v4l2 if it has not been installed to the Orin system before.
   sudo apt-get update
   sudo apt-get install v4l-utils

Note 1/2

2. Note: If you would like to install Jetpack 5.0.2 but don't want to re-flash the whole OS image, you can uncheck the Jetson OS and install the Jetson SDK components only.



2. If there are any new drivers, we will add them into the link below.

https://www.dropbox.com/sh/azymqdrce2ey9c0/AAC3vDEK24BTeyUsfvX6yml1a?dl=0

Note 2/2

3. **Compile the driver**

If you would like to recompile the driver, please follow the below steps.
Download the driver code and Tool chain from Dropbox links below.

Kernel Code:https://www.dropbox.com/s/cnmjmpecwzd6zeg/kernel_src.tbz2?dl=0
GCC ToolChain: https://www.dropbox.com/s/p7f9dyg8apmzws6/aarch64--glibc--stable-final.tar.gz?dl=0

Compile the kernel under 64-bit Ubuntu OS on Intel x64 PC. (Virtual machine is fine. We are using Ubuntu 20.04/18.04)

1) Copy compile tool "aarch64--glibc--stable-final.tar.gz" to "/opt", and then unzip it there:
   sudo tar xpf arrch64--glibc--stable-final.tar.gz

2) Copy "kernel_src.tbz2" and the two patch files (do not use sudo command to copy here) to a newly created directory called "project" under "~/Downloads"(for example: cd ~/Downloads, mkdir project, cd project) and then do the following:
   tar xvfp kernel_src.tbz2
   sudo chown -R <user_name> kernel
   sudo chown -R <user_name> hardware
   Note: <user_name> is the user name of your Ubuntu OS.
   For example: sudo chown -R leopard kernel

3) Apply the software patch files as shown below:
   patch -p1 < ar0234_dual_hawk_gmsl2_max96712_35.1_xavier_20221028_dtbs
   patch -p1 < ar0234_dual_hawk_gsml2_max96712_35.1_xavier_20221028_kernel.patch

4) To install the tool in the kernel_src.tbz2 file extracted directory, execute below commands:
   sudo apt-get install flex
   sudo apt-get install bison
   sudo apt-get install openssl
   sudo apt-get install libssl-dev

5) To compile in the kernel_src.btz2 file extracted directory and execute the following commands:
   export CROSS_COMPILE_AARCH64_PATH=/opt/bin/aarch64-buildroot-linux-gnu-
   export CROSS_COMPILE_AARCH64_PATH=/opt
   ./nvbuild.sh -o $PWD/kernel_out/
   Note: /opt / is the installation path where the compiler is decompressed.

6) To check the important compiled result files for this camera driver:

Note: Assumed the previous kernel_src.tbz2 extracted and recompiled under "~/Downloads/project".
Finally, you can get the following files at the following paths from the "~/Downloads/project":
1) tegra234-p3701-0000-p3737-0000.dtb under ~/Downloads/project/kernel_out/arch/arm64/boot/dts/nvidia/tegra234-p3701-0000-p3737-0000.dtb
2) tegra194-p2888-0001-p2822-0000.dtb under ~/Downloads/project/kernel_out/arch/arm64/boot/dts/nvidia/tegra194-p2888-0001-p2822-0000.dtb
3) max96712.ko under ~/Downloads/project/kernel_out/drivers/media/i2c/max96712.ko
4) nv_ar0234.ko under ~/Downloads/project/kernel_out/drivers/media/i2c/nv_ar0234.ko
5) bmi088.ko under ~/Downloads/project/kernel_out/drivers/media/i2c/bmi088.ko

IMU support

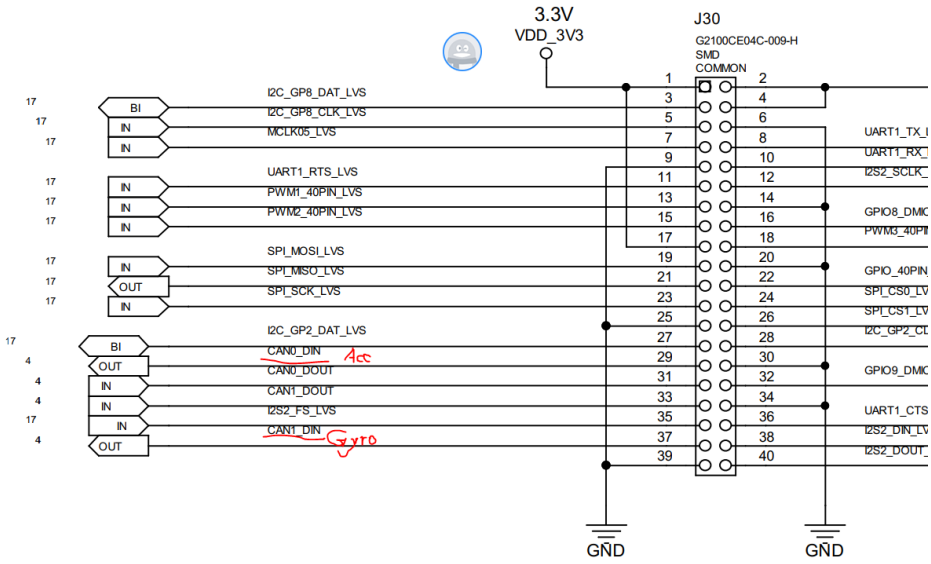
1. How to access imu data in driver

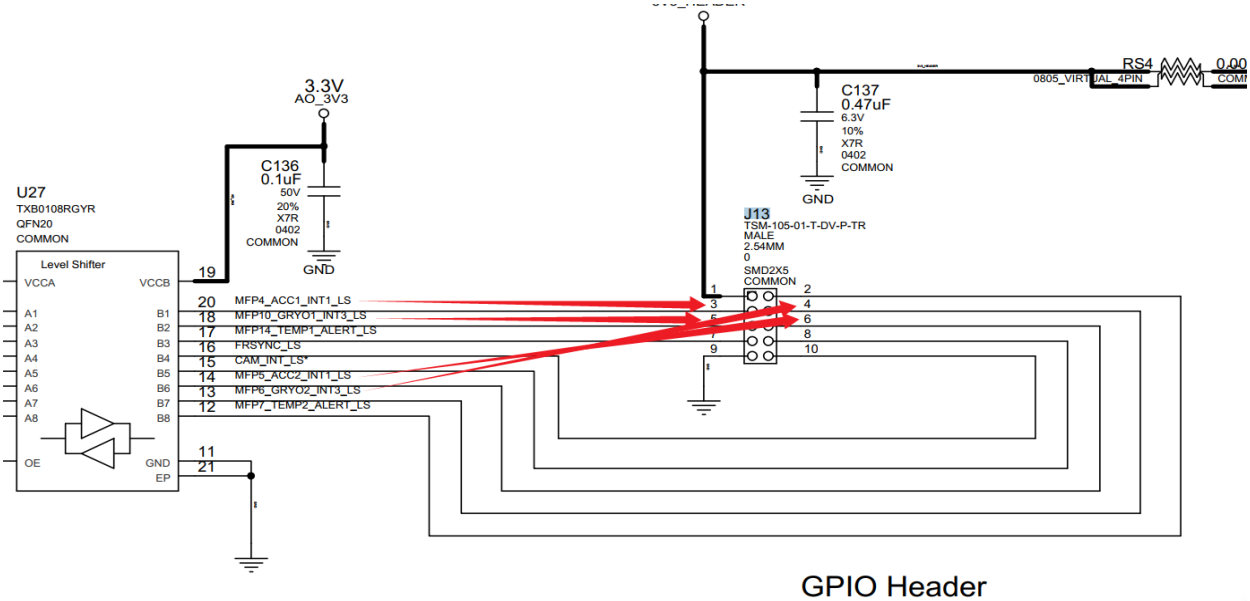
Acc:

1.cd /sys/devices/platform/3180000.i2c/i2c-2/i2c-30/30-0069/iio:device0
cd scan_elements
echo 1 >in_accel_x_en
echo 1 > in_accel_y_en
echo 1 > in_accel_z_en
echo 1 > in_timestamp_en
cd ../buffer
echo 1 > enable
cd ..
cat in_accel_x_raw
cat in_accel_y_raw
cat in_accel_y_raw

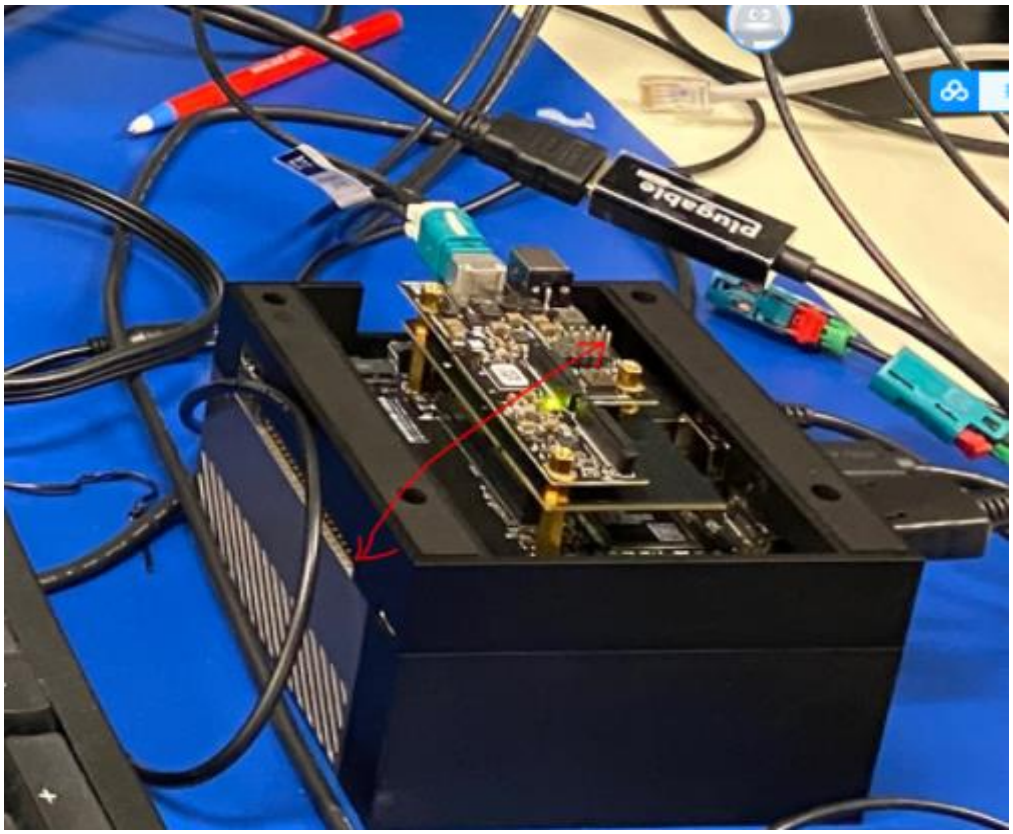
geo :

1.cd /sys/devices/platform/3180000.i2c/i2c-2/i2c-30/30-0069/iio:device1
cd scan_elements
echo 1 >in_anglvel_x_en
echo 1 >in_anglvel_y_en
echo 1 > in_anglvel_z_en
echo 1 > in_timestamp_en
cd ../buffer
echo 1 > enable
cd ..
cat in_ anglvel_x_raw
cat in_ anglvel_y_raw
cat in_ anglvel_y_raw





GPIO Header
Cable to carrier board



Note: The Red color arrows indicates the pin 1 position on the E3653-A03 and the AGX Orin system.