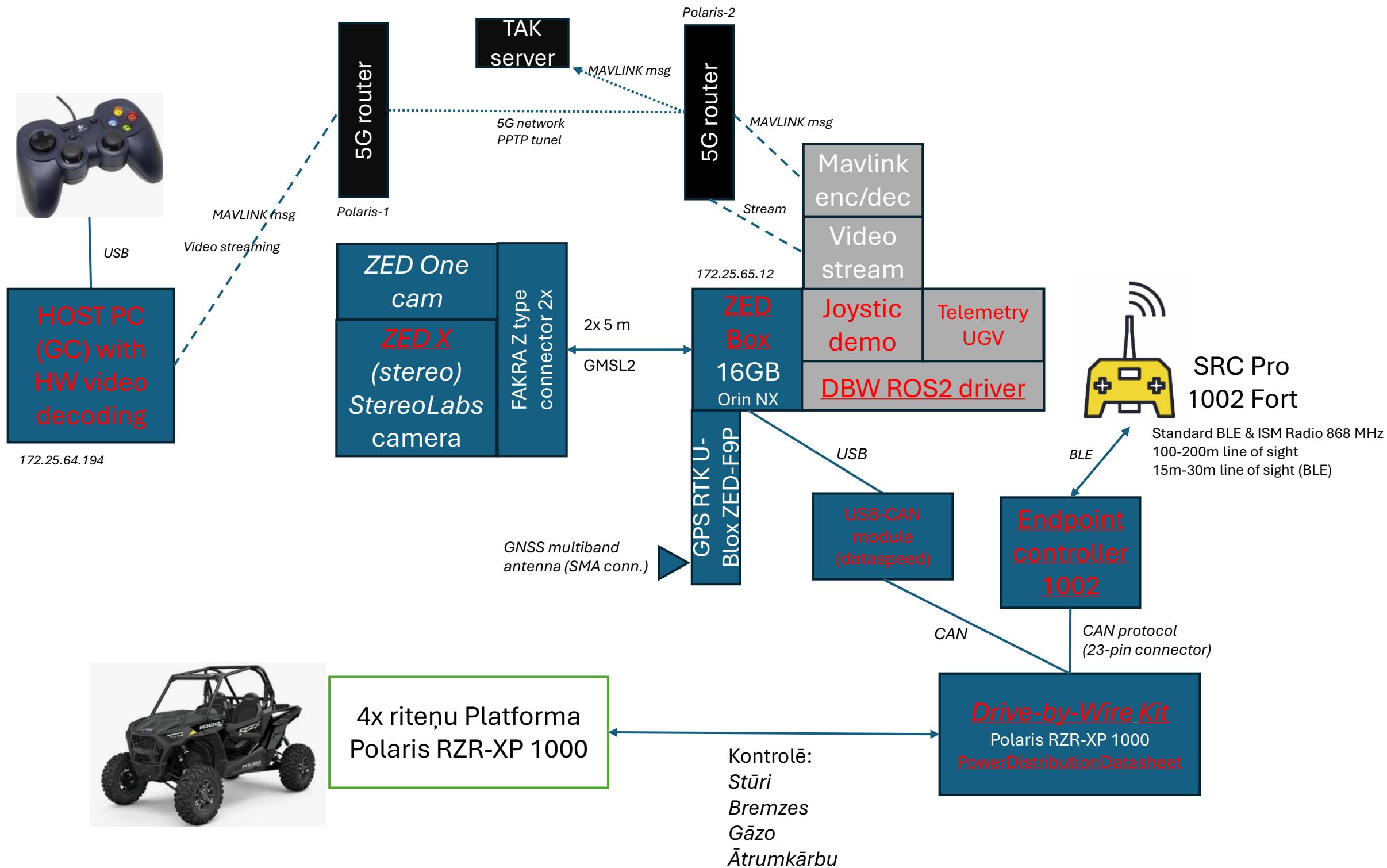
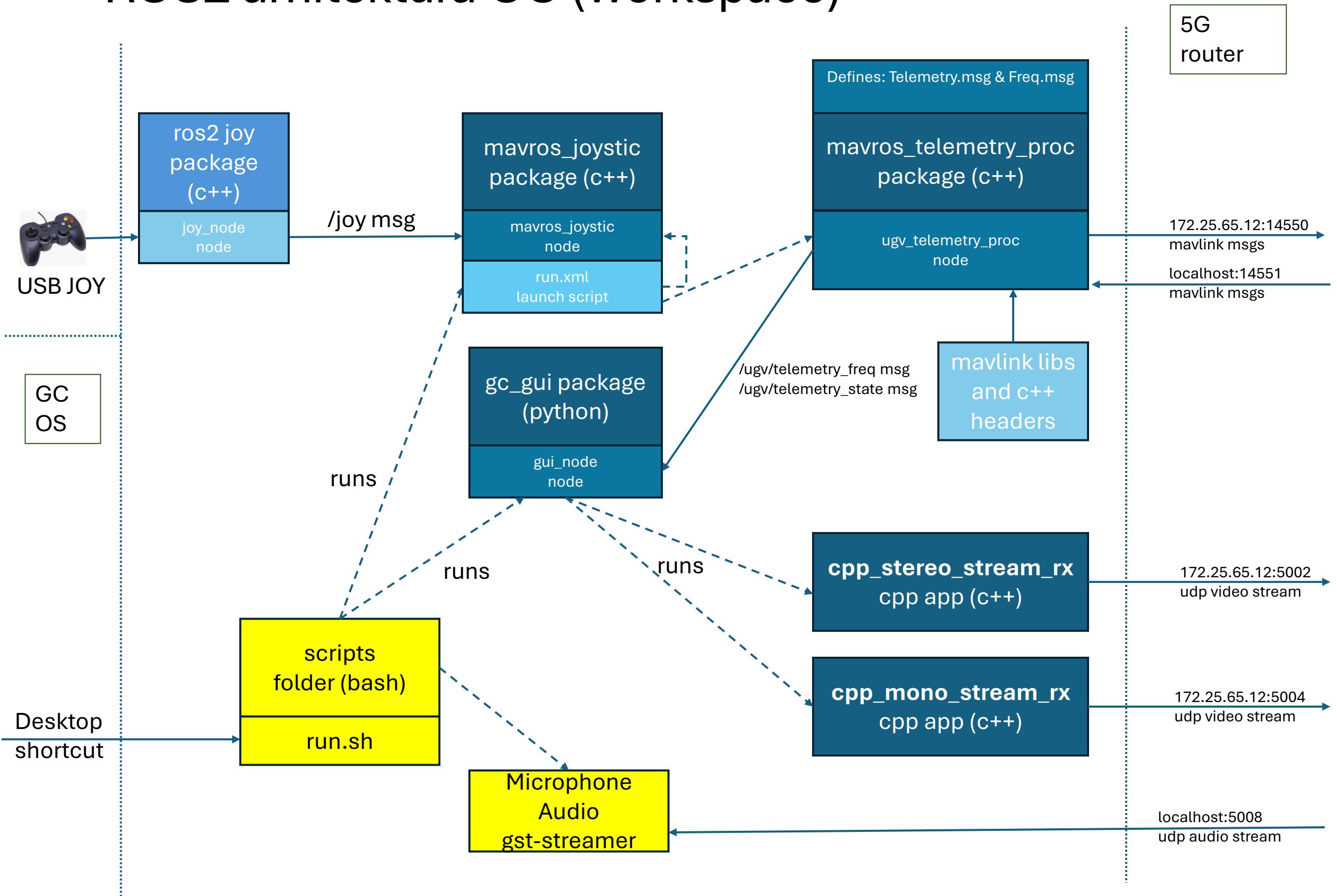


MAVLINK msg

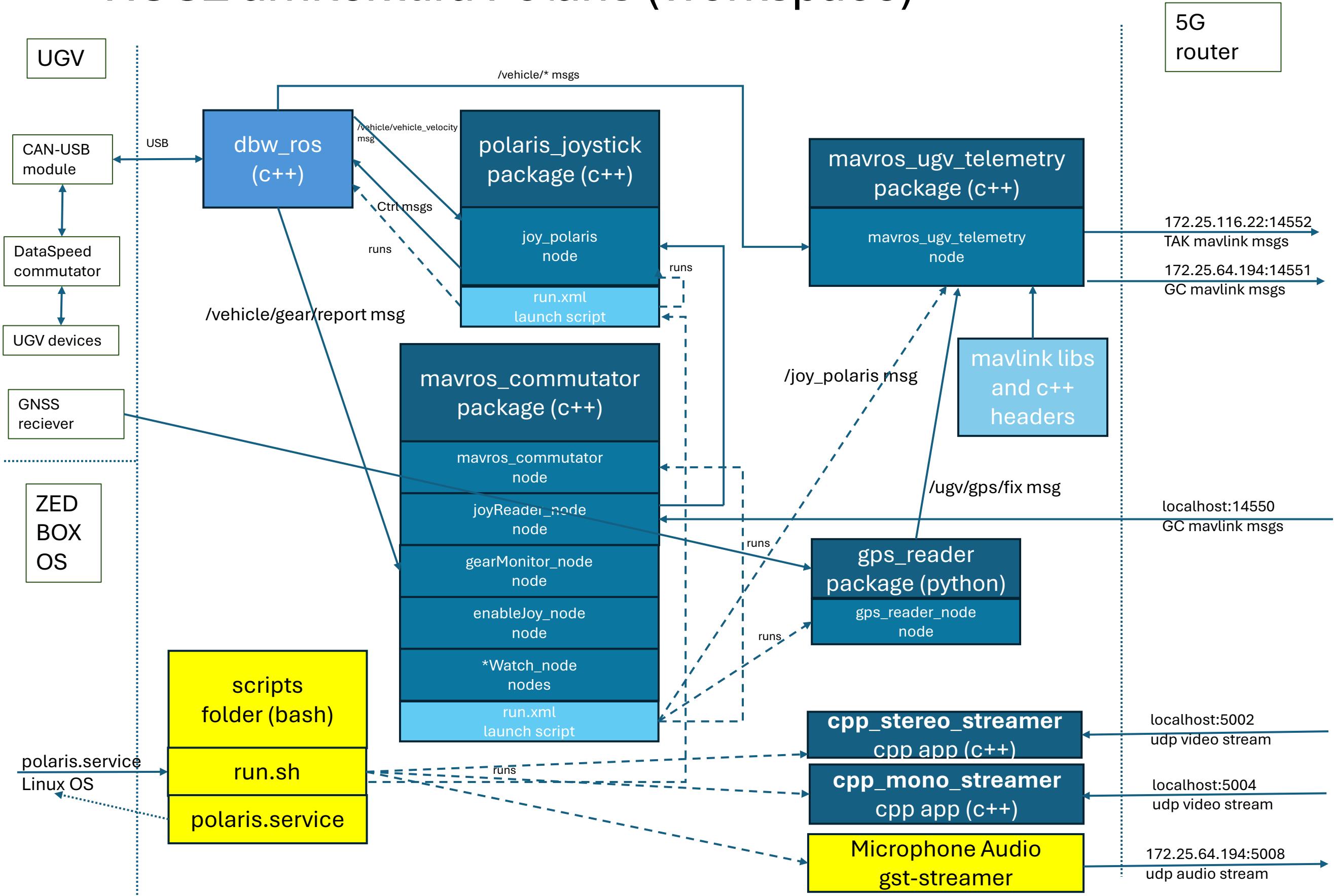
KONCEPTUĀLA SHĒMA



ROS2 arhitektūra GC (workspace)



ROS2 arhitektūra Polaris (workspace)



Setup GC on Ubuntu 24.04

Install ROS2 KILTED KAIJU (ubuntu 24.04):

Use link: [Ubuntu \(deb packages\) — ROS 2 Documentation: Kilted documentation](#)

Create ROS2 workspace:

```
mkdir -p ~/gc_ws/src  
cd gc_ws
```

Init the workspace:

```
colcon build --symlink-install  
source install/setup.bash
```

Clone LMT GC repository:

```
cd src/  
git clone git@github.com:dainissilamikelis/polaris-ground-control.git  
cd polaris-ground-control  
git submodule update --init --recursive
```

Build mavlink:

```
cd polaris-ground-control/mavlink  
cmake -Bbuild -H. -DCMAKE_INSTALL_PREFIX=~/gc_ws/install/mavlink -DMAVLINK_DIALECT=common -DMAVLINK_VERSION=2.0  
cmake --build build --target install
```

Compile ROS2 workspace's packages:

```
cd ~/gc_ws  
source install/setup.bash  
sudo rosdep init  
rosdep update  
source install/setup.bash  
colcon build --symlink-install
```

Setup Python tools:

```
sudo apt install python3-kivy
```

Setup other tools:

```
sudo apt install wmcctrl
```

CUDA 13.0 (must be compared to the version of host machine):

```
wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/x86_64/cuda-keyring_1.1-1_all.deb  
sudo dpkg -i cuda-keyring_1.1-1_all.deb  
sudo apt-get update  
sudo apt-get -y install cuda-toolkit-13-0
```

Create desktop shortcut to run GC:

Edit ~/gc_ws/src/polaris-ground-control/scripts/run.sh to set correct environment path

Edit ~/gc_ws/src/polaris-ground-control/scripts/GroundControlPolaris.desktop to set the correct path to run.sh script

```
chmod +x ~/gc_ws/src/polaris-ground-control/scripts/run.sh
```

```
cp ~/gc_ws/src/polaris-ground-control/scripts/GroundControlPolaris.desktop ~/Desktop
```

```
chmod +x ~/Desktop/GroundControlPolaris.desktop
```

Allow Launching for the shortcut (Mouse Right -> Allow Launching)

Setup Polaris ZED Orion NX 16GB on Ubuntu 22.04 (1)

Check system:

```
lsb_release -a  
dpkg-query -W nvidia-l4t-core  
nvidia-smi  
apt-cache show nvidia-jetpack
```

Original version of Orion NX 16GB: is R36 (release), REVISION: 3.0 - > JetPack 6.0 & L4T Version 36.3.0

Install ROS2 Humble: [Ubuntu \(deb packages\)](#) — [ROS 2 Documentation: Humble documentation](#)

Generate ssh key for the project repositories and register it in [github.com](#) (for a user):

```
ssh-keygen -t ed25519 -C „user-e-mail-here“
```

Install ZED StereoLabs cameras SDK (ZED SDK 5.0, CUDA 12.6, Jetpack 6.2)

More info: [here](#)

```
mkdir -p pol_ws/downloads && cd pol_ws/downloads  
wget -O ZED_SDK_Tegra_L4T36.3_v5.0.7.zstd.run  
"https://download.stereolabs.com/zedsdk/5.0/l4t36.3/jetsons?_gl=1*1bezr7q*_gcl_au*MTIwNTg0MDg0Mi4xNzU2Mjg3  
Mzcx"  
chmod +x ./ZED_SDK_Tegra_L4T36.3_v5.0.7.zstd.run  
ZED_SDK_Tegra_L4T36.3_v5.0.7.zstd.run  
# Installation path: /usr/local/zed/samples/
```

Install [ZedLink drivers](#):

(drivers [list](#))
download deb file: stereolabs-zedbox-duo_1.3.1-LI-MAX96712-all-ZEDBOX-L4T36.3.0_arm64.deb
sudo dpkg -i ./stereolabs-zedbox-duo_1.3.1-LI-MAX96712-all-ZEDBOX-L4T36.3.0_arm64.deb
sudo reboot
sudo dmesg | grep zedx

Install ZED gstreamer libraries:

Install prerequirements from [here](#)
git clone <https://github.com/stereolabs/zed-gstreamer.git>
cd zed-gstreamer
mkdir build && cd build
cmake -DCMAKE_BUILD_TYPE=Release ..
make
sudo make install
sudo apt update
sudo apt install libgstrtspserver-1.0-dev

Check the prev. install:

```
gst-inspect-1.0 zedsrc  
gst-inspect-1.0 zedxonesrc  
gst-inspect-1.0 zeddemux  
gst-inspect-1.0 zeddatamux  
gst-inspect-1.0 zeddatacsvsink  
gst-inspect-1.0 zedodoverlay
```

Connect and test the camera ([useful info](#)):

```
sudo systemctl restart zed_x_daemon  
ZED_Diagnostic -c
```

Setup Polaris ZED Orion NX 16GB on Ubuntu 22.04 (2)

Init ROS2 WS:
cd ~/pol_ws
mkdir src
colcon build

Clone LMT Polaris repository:
cd src
git clone [git@github.com:dainissilamikelis/polaris-os.git](https://github.com/dainissilamikelis/polaris-os.git)
cd polaris-os
git submodule update --init --recursive

Build mavlink:
cd mavlink
cmake -Bbuild -H. -DCMAKE_INSTALL_PREFIX=~/pol_ws/install/mavlink -DMAVLINK_DIALECT=common -DMAVLINK_VERSION=2.0
cmake --build build --target install
touch COLCON_IGNORE

Install dependences:
sudo apt update
sudo apt install ros-humble-can-msgs
cd ~/ros2_ws/src
git clone https://bitbucket.org/DataspeedInc/dataspeed_can.git
cd ..
rosdep install --from-paths src --ignore-src -r -y
colcon build --packages-select dataspeed_can_msg_filters

cd src
git clone https://github.com/autowarefoundation/ros2_socketcan.git
cd ..
rosdep install --from-paths src --ignore-src -r -y
colcon build --packages-select ros2_socketcan ros2_socketcan_msgs

Setup CAN-USB dataspeed tool device:

```
lsusb
echo 'SUBSYSTEM=="usb", ATTR{idVendor}=="6923", ATTR{idProduct}=="0112", MODE=="0666"' | \
sudo tee /etc/udev/rules.d/99-dataspeed-can.rules
sudo udevadm control --reload-rules
sudo udevadm trigger
ls /dev | grep can
modprobe can
modprobe can_raw
modprobe can_dev
ip link show
```

Setup GNNS receiver: [ZED link](#)
sudo apt install libgps-dev
cgps -s

Compile ROS2 workspace's packages:
cd ~/pol_ws
source install/setup.bash
sudo rosdep init
rosdep update
source install/setup.bash
colcon build --symlink-install

Register as service
Edit ~/pol_ws/polaris-os/scripts/polaris.service to set correct run.sh path and user name, and working dir
Edit ~/pol_ws/polaris-os/scripts/run.sh to set CG IP address
sudo cp ~/pol_ws/src/polaris-os/scripts/polaris.service /etc/systemd/system
chown user:user /home/user/pol_ws/src/polaris-os/scripts/run.sh
chmod +x /home/user/pol_ws/src/polaris-os/scripts/run.sh

```
sudo systemctl daemon-reexec
sudo systemctl daemon-reload
sudo systemctl enable polaris.service
sudo systemctl start polaris.service
Check status:
sudo systemctl status polaris.service
journalctl -u polaris.service
```

RUN commands (manual run)

Polaris ZED Box (user@admin): 172.25.65.12

Bash Terminal 1:

```
cd ~/pol_ws  
source install/setup.bash  
ros2 launch polaris_joystick joystick.launch.xml
```

Bash Terminal 2:

```
cd ~/pol_ws  
source install/setup.bash  
ros2 launch mavros_commutator run.xml
```

Bash Terminal 3:

```
cd ~/pol_ws  
source install/setup.bash  
.build/ZED_Streaming_Sender/ZED_Streaming_Sender 5002
```

Bash Terminal 4:

```
cd ~/pol_ws  
source install/setup.bash  
.build/ZED_One_live/ZED_One_streaming_sender 5004
```

CG hos (sholc@sholc): 172.25.64.194

Bash Terminal 1:

```
cd ~/gc_ws  
source install/setup.bash  
ros2 run gc_gui gui_node
```

Bash Terminal 2:

```
cd ~/gc_ws  
source install/setup.bash  
ros2 run mavros_telemetry_proc mavros_telemetry_proc_node
```

Bash Terminal 3:

```
cd ~/gc_ws  
source install/setup.bash  
ros2 launch mavros_joystick run.xml
```

Bash Terminal 4:

```
cd ~/gc_ws  
source install/setup.bash  
.build/ZED_One_live/ZED_One_streaming_receiver 172.25.65.12:5004
```

Bash Terminal 5:

```
cd ~/gc_ws  
source install/setup.bash  
.build/ZED_Streaming_Receiver/ZED_Streaming_Receiver 172.25.65.12:5002
```