Set up PC to communicate with LCM

Prerequisites:

This Guide assumes that you are running Ubuntu 22.04 and have ROS2 Humble installed.

Setting up udev rules:

1. Disable brltty udev rules.

bash

```
for f in /usr/lib/udev/rules.d/*brltty*.rules; do
    sudo ln -sf /dev/null "/etc/udev/rules.d/$(basename "$f")"
done
```

2. Create udev rules that pick up our LCM and rename it's interface to ttyPCB

- create file `/etc/udev/rules.d/rs-robot-tty.rules` with permissions `644` and owned by `root` with the following contents:

> field **`OWNER` should have the name user that will launch PCB bridge** (usually the default user that one logs in with)

```
SUBSYSTEM=="tty",ATTRS{idProduct}=="7523",ATTRS{idVendor}=="1a86",SYMLINK+="ttyPCB",OWNER+="robot",MODE+="0666"
SUBSYSTEM=="tty",ATTRS{idProduct}=="7523",ATTRS{idVendor}=="9986",SYMLINK+="ttyPCB",OWNER+="robot",MODE+="0666"
```

- to set the permissions to the udev rule file: `sudo chmod 644 /etc/udev/rules.d/rs-robot-tty.rules`
  3. reload udev rules:

bash

```
# Reload udev management tool
sudo udevadm control --reload-rules
sudo udevadm trigger
```

4. Reboot.

5. Verify by plugging in the PCB in computer with USB and checking that `/dev/ttyPCB` exists and points to device with correct permissions:

   - `ls -lah /dev/ttyPCB`
   - you should see similar output to what is below.

     ```
     lrwxrwxrwx 1 root root 7 Mar 22 14:03 /dev/ttyPCB -> ttyUSB0
     ```

   - Make sure that the device `ttyPCB` points to (in this case ttyUSB0) has correct owner and permissions using `ls -lah /dev/ttyUSB0`
     - check that owner of this usb file matches what you chose (in this case `robot`) for `OWNER` field in the udev rule.

     ```
     crw-rw-rw- 1 robot dialout 188, 0 Mar 22 12:17 /dev/ttyUSB0
     ```

Setting up PCB bridge

1. Set up udev rules.

2. copy .deb files to computer.

3. in the folder with these deb files, run `sudo apt install ./ros-humble-robot-*jammy_amd64.deb`

   > if this fails, try moving .deb files to `/tmp` and running this command there.

4. Now this ros node can be ran just like any other ros node with ether `ros2 run` or from launch files. e.g. `ros2 run robot_pcb_bridge robot_pcb_bridge`

   > if the node runs only as root (by running `sudo ros2 run robot_pcb_bridge`) there Likely is issue with `/etc/udev/rules.d/rs-robot-tty.rules`