# HIPER Foot Gripper – Guide

## Software and Motor Control

GitHub for my code: <https://github.com/leggedrobotics/hiper_low_level_control>

What do you need for 1 finger:

* 1 Hiper Finger / 3 Maxon motors + 1 Mosrac motor with encoders
* 3x EPOS4 Micro EtherCAT Driver board
* EPOS4 EvalBoard
* Power Supply 24V & 2A
* USB Micro Cable
* Ethernet Cable (to PC, so maybe also adapter to USB)
* EPOS Studio Software (only on Windows)
* Linux Computer (no WSL or virtual machine)

## Connect and Configure the motors and drivers

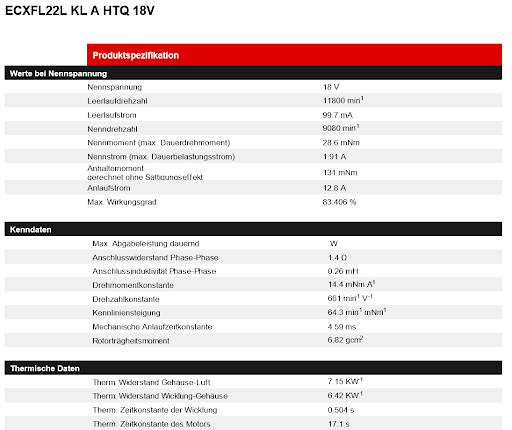
This step is necessary if you have new motors/different drivers. You can try directly running the linux communication interface programm and see if this is enough to configure the driver so it works. If problems arise (e.g. driver is in fault state) it helps doing the setup via EPOS studio first. This can also help in troubleshooting to reset the driver completely in EPOS studio and set it up newly. A handy feature is the directory overview where you can read all the values in the drivers registers which helps debugging.

1. Download EPOS Studio: <https://www.maxongroup.com/medias/mmc_webmedia/installers/EPOS/EPOS-2-4-IDX-Setup.zip> (check for newest version!)
2. Wire Motor to Eval Board as well as plug EPOS4 Micro diver into the EvalBoard, connect EvalBoard via USB to computer, power EvalBoard with DC power supply (**24V, limit at 2A**)
3. Start EPOS Studio, should auto-detect EPOS4 Eval Board
4. Configure the motor parameters, controller settings  
   Video Playlist can help: <https://www.youtube.com/playlist?list=PLmklAQtFT_ZINeRxFG2BschY0BM_2DYqI>  
   Take following screenshots as orientation for parameters. (if using the same motors and drivers, the control gains technically don’t need to be tuned again, and need to be set, since they are stored in the yaml file of the motor which will then be transmitted to the driver when running the code on the linux system for communication interface)

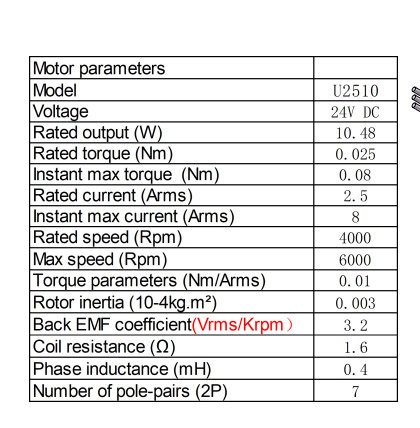
## Motor Parameters

|  |  |  |
| --- | --- | --- |
|  | Maxon | Mosrac |
| Working Voltage | 24V (18V) | 24V |
| Pole Pairs | 6 | 7 |
| Speed Constant | 661 /min/V | 312.5 rpm/V |
| Torque Constant | 14.4 mNm/A | 0.03 Nm/A |
| Max Current | 12800mA | 8000mA |
| Nominal Current | 1910mA | 2500mA |
| Motor Rated Torque | 28.6mNm | 0.025 Nm |
| Min position | 0 (for now) | 0 (for now) |
| Max position | 0 (for now) | 0 (for now) |
| Max Profile Velocity | 9440 rpm | 4800rpm (20%margin) |
| Thermal Time Constant Winding | 0.504s | 2s (estimate) |
| Max Speed | 11800 rpm (8333 in epos studio) | 6000rpm |
| Max Acceleration | 1837000rpm | 477500 rpm/s |
| Sensor | Hall Sensor | SSI absolute encoder |

Maxon Motor Sheet:



Mosrac Motor Sheet:



Mosrac Motor Encoder: follow the pictures for setup

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

If the driver is not reading any encoder data, don’t forget to adapt the switch to the device position:

A screenshot of a computer program

AI-generated content may be incorrect.

Additionally, to configure the encoder offset value refer to the instructions from maxon starting on p.131 and following.

<https://www.maxongroup.com/medias/sys_master/root/8884071235614/EPOS4-Application-Notes-Collection-En.pdf>

For tuning the control parameters this instructions from maxon can help:

<https://support.maxongroup.com/hc/en-us/articles/4411824545682-EPOS4-IDX-Manual-control-parameter-tuning-hints>

## Installation of Communication Program

1. Install Ubuntu 20.04 (not as dual-boot or WSL2)
2. Run: sudo apt update && sudo apt upgrade -y
3. Install ROS Noetic <https://wiki.ros.org/noetic/Installation/Ubuntu>

Probably this is already enough and now you can just use Rosinas code, but just in case I did the following steps to stup the workspace:

1. Create Catkin Workspace:  
   mkdir -p ~/catkin\_ws/src

cd ~/catkin\_ws

catkin init

source devel/setup.bash  
echo "source ~/catkin\_ws/devel/setup.bash" >> ~/.bashrc

1. Clone required repos:

cd ~/catkin\_ws/src

git clone https://github.com/leggedrobotics/message\_logger.git

git clone https://github.com/leggedrobotics/soem\_interface.git

git clone https://github.com/leggedrobotics/ethercat\_sdk\_master.git

git clone https://github.com/leggedrobotics/ethercat\_device\_configurator.git

git clone https://github.com/leggedrobotics/maxon\_epos\_ethercat\_sdk.git

cd ../

catkin build

A screenshot of a computer

AI-generated content may be incorrect.run: sudo bash -c ‘source /catkin\_ws\_hiper\_hand/devel/setup.bash; rosrun ethercat\_device\_configurator hiper\_hand catkin\_ws\_hiper\_hand/src/maxon\_epos\_ethercad\_sdk/config/setup.yaml’

## Most important files

The most importatant files are marked in red circles:  
 hiper\_hand.cpp and maxon.yaml, mosrac.yaml and setup.yaml

In the hiper\_hand you can adapt the code for control in the   
cyclicUserInteraction() function. And the yaml files are most important  
to store the motor parameters and the system setups. At the moment in the   
setup.yaml 2 motors are activated, just copy paste the code if you  
use more motors and adapt the type from maxon to mosrac if necessary

## Assembly

For the assembly process please look in the CAD, the pictures and the explanation videos in 07\_Manufacturing -> Assembly (<https://drive.google.com/drive/folders/1-OgwTjXhhlKgsBbEZcmYA51tPH-j6haS?usp=sharing>)

Also make sure to check out the BOM file, it is updated to the 12.08.25 and include all parts needed for assembly: <https://docs.google.com/spreadsheets/d/1gtcY6w_a1Wsbm2QDcyvMIakDZR44d4pNBZoDtqCXC7s/edit?usp=sharing>