Documentation for EV3 controlling large DC motors

This is a guide to setup an EV3 to independently control two large DC motors via EV3.dev.

Supplies

1x [Sabretooth motor driver](https://www.dimensionengineering.com/datasheets/Sabertooth2x12.pdf) (2x12 Amp used in this guide)

2x [DC Motors](https://www.amazon.com/uxcell-Reversible-Reducing-Electric-Motor-JCF63L/dp/B0732GPB4N/ref=pd_sbs_60_2/133-7533894-0622021?_encoding=UTF8&pd_rd_i=B0732GPB4N&pd_rd_r=27cf3057-18df-47de-9720-554c5bfa28d1&pd_rd_w=hqCZV&pd_rd_wg=le9hJ&pf_rd_p=1c11b7ff-9ffb-4ba6-8036-be1b0afa79bb&pf_rd_r=TBSGZD6JXGR3MPZ75Y54&psc=1&refRID=TBSGZD6JXGR3MPZ75Y54) (24V or Below)

1x Voltage Divider (Used 2x 2.7 K-Ohms, 2x 4.7 K-Ohms, 2 potentiometers for quick tuning)

1x Low Pass Filter (1x 0.1 μF capacitor and 1x 180 Ohms Resistor)

1x Power Supply (or a 12 V, 5-amp computer power supply with adapter)

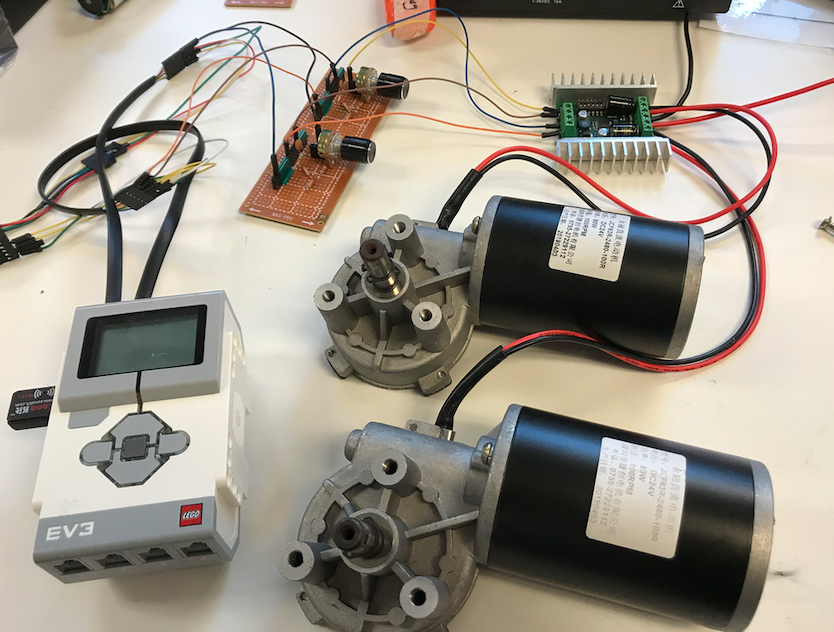
1x EV3, SD chip, Wi-fi dongle and access to a computer/Wi-fi

1x Encoder (5 VCC compatible)

Jumper Cables

EV3//DC Motor Diagram





After the EV3 has EV3.dev downloaded on a SD card inserted and a Wi-fi dongle connected to it, one can use python to control PWM signals from the EV3 to control the analog voltage interface on the Sabretooth motor driver.

1. The EV3 needs to think that it’s attached to a Lego motor to supply a 8V PWM. (reference <http://www.biasedlogic.com/index.php/running-third-party-motor-with-your-mindstorms-ev3-nxt/> for more information). There are varies ways to achieve this, but in this guide the Pololu 6 V motor w/ encoder (YC2010-24) was used without any in-line resistors. (Also, a 3.3 k-ohm resistor connected to the Vcc and SDA, yellow wire will work [see below [1]]). This allowed us to send PWM signals to the motor driver.



[1] hookup trick EV3.dev

1. The sabretooth motor driver’s dip switches need to be set properly before hooking up. (<https://www.dimensionengineering.com/datasheets/SabertoothDIPWizard/nonlithium/analog/independent/linear.htm>).
2. After dip switches are set (on 1,2,3,5,6 – off 4), the PWM signal from the EV3 needs to be run through a Low Pass RC filter to convert the PWM to an analog voltage. The sabretooth does not accept PWM cycles as a valid input. In this guide a voltage divider was used to turn the 0 to 8 V signal from the EV3 to 0 to 5 V. This is because the motor driver does not accept voltage inputs over 6 V. Accepted voltages for the motor driver are 0 to 5 V, with 2.5 V commanding the driver to stop. Anything below 2.5 (above 0 volts) will move the motors backward and anything between above 2.5 V to 5 V will move the motors forward. After the divider is setup, 5 volts are sent to the low pass filter (1x 0.1 μF capacitor and 1x 120 Ohms resistor). In the suggested setup using the resistors in this guide, PWM at ~50 will halt the motors. (This might change depending on the charge of the EV3 battery) PWM below 45 will move the motors in reverse and above 55 will move the motors forward.
3. After the filter is setup it needs to be connected to the motor driver. 0V connected to the low of the filter and the high voltage should be connected to S1 which will move one motor. If using the same signal another connection could be used to connect to S2 on the driver. This will control the second motor with the same commands. A duplicate setup (encoder, resistors, low pass filter) is required to control two motors separately. Also, make sure the filter is on the bottom of the voltage divider. The ground for controlling two motor separately needs to be at 0 V for it to work.
4. Connected the motors to the driver and the power supply. **Make sure the power from the supply/battery is correct (positive to positive and negative to negative) or the driver will break.**
5. Use an SSH connection from terminal (Mac) to use a program to control the EV3.

Code for using two motors: <https://github.com/drfricke/EV3-Robot/blob/master/control.py>

Manuel for the Sabretooth: <https://www.dimensionengineering.com/datasheets/Sabertooth2x12.pdf>