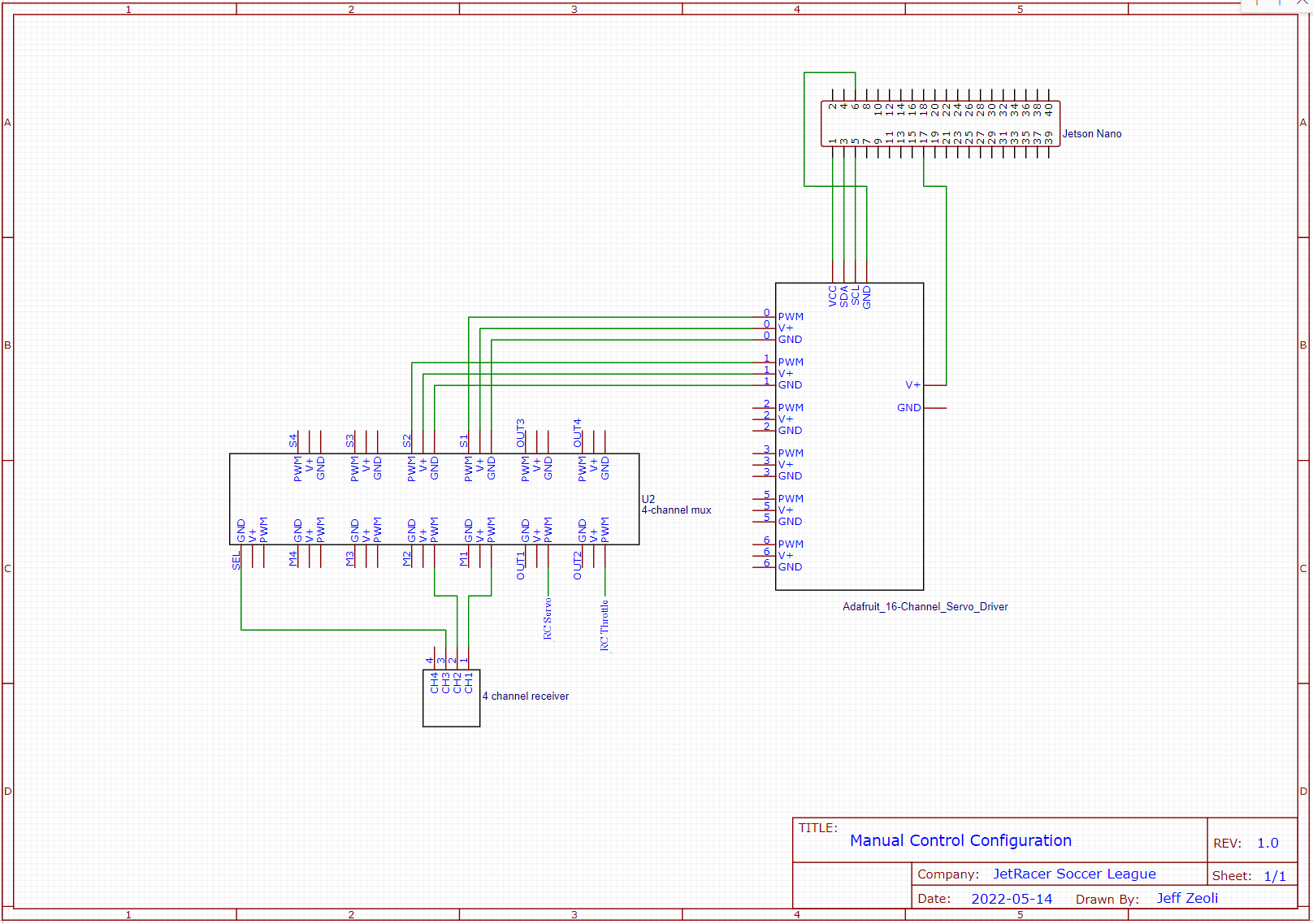
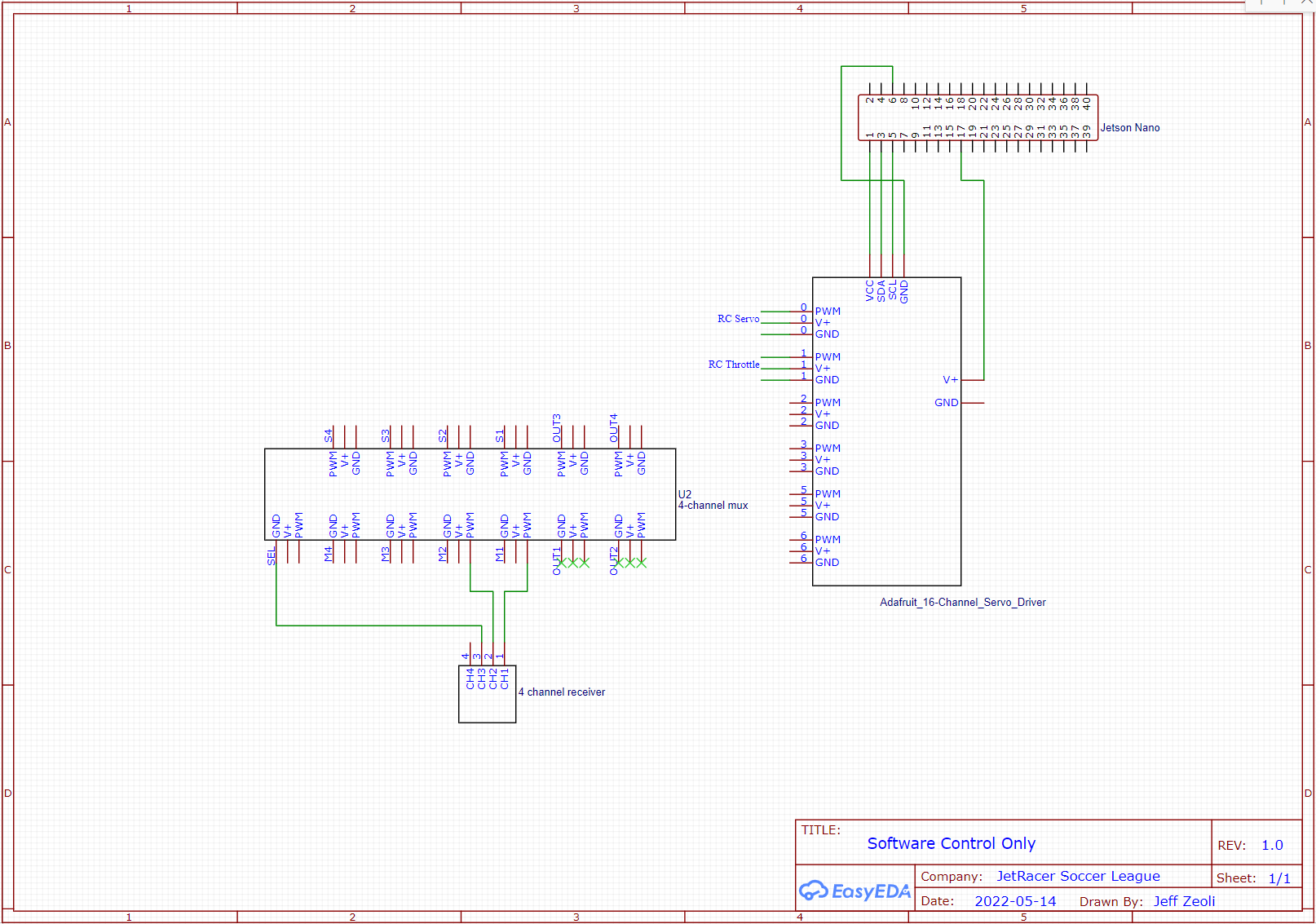
Build chassis by following [these instructions](https://www.tamiya.com/english/rc/rcmanual/tt02.pdf)

Add additional hardware following [these instructions](https://github.com/NVIDIA-AI-IOT/jetracer/blob/master/docs/tamiya/hardware_setup.md) and using [these materials](https://docs.google.com/spreadsheets/d/1pdqG1og2lO-83MmD-lcoXxOteeDbbvSXhTYG8bqrR_c/edit#gid=0)

Follow below wiring configurations for manual and software control. The manual control configuration should work for both hardware and software if the MUX is calibrated correctly.





Test that hardware is setup correctly by following the [basic motion](https://github.com/NVIDIA-AI-IOT/jetracer/blob/master/notebooks/basic_motion.ipynb) example. Below is a user-friendly guide to help you out.

car.steering and car.throttle will accept any value within range [-1.0, 1.0]. From calibration testing, we found that the car has a complete steering range of (-0.5, 0.5) with 0.1 resolution when 0 points the wheels forward. Also, the throttle must receive a value of at least 0.15 to begin moving. The JetRacer will stop when throttle is set to 0, reach max speed when set to 1, reach minimum speed when set to 0.125, and can reverse when set to a negative value after being stopped. The JetRacer will stop moving if you set throttle to a negative value when it is moving forward.

from jetracer.nvidia\_racecar import NvidiaRacecar

car = NvidiaRacecar()

car.steering\_offset = 0.1 # calibrate for each car

car.throttle = 0.0 # stops car

car.steering = 0.0 # wheels point forward

def steer\_forward():

car.steering = 0.0

# setting steering to negative makes JetRacer turn right

def steer\_max\_right():

car.steering = -1.0

# setting steering to positive makes JetRacer turn left

def steer\_max\_left():

car.steering = 1.0

def steer\_slight\_right():

car.steering = car.steering - 0.1

def steer\_slight\_left():

car.steering = car.steering + 0.1

def throttle\_max():

car.throttle = 1.0

def throttle\_min():

car.throttle = 0.15

car.throttle = 0.125

def throttle\_mid():

car.throttle = 0.15

def throttle\_accelerate():

if (car.throttle + 0.25 <= 1.0):

car.throttle = car.throttle + 0.025

def throttle\_decelerate():

if (car.throttle - 0.25 >= 0):

car.throttle = car.throttle - 0.025

def throttle\_stop():

car.throttle = 0.0

# setting throttle to negative when stopped should make the JetRacer reverse

def throttle\_reverse():

car.throttle = 0.0

car.throttle = -0.15