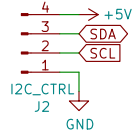


Power applied to pins for ATmega328P compatibility if needed

### Microcontroller

Responsible for controlling the entire system. Receives speed from an input and drives the MOSFET driver accordingly.

Uses internal analog comparator to deduce rotor position using the BEMF.



### Control Connectors

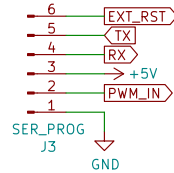
Either system can be used. PWM is standard for commercial ESCs. 1 to 2ms duty, 20 ms period. It is integrated in the Serial programmer header.

I added I2C in the event I want to retrieve data from the ESC or adjust speed at a rate >50Hz.

### Power Pads

These connections are for the batter coming in as well as either 5V coming out of the ESC or in for the control system if BEC is not installed locally.

Test pads are just placeholders for the ends of the exposed traces. Solder wires to exposed regions.



### Serial Programming

Used to upload Arduino sketches using a serial programmer.

### I2C Address Jumpers

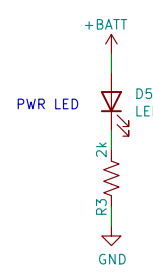
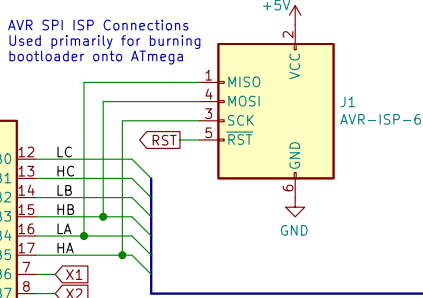
Shorting these [to GND] sets the I2C address of the module. Allows for up to eight units on a shared I2C bus, without additional changes.

**Reverse Jumper**  
Shorting it [to GND] reverses the direction of the motor

### MOSFET Bootstrapping System

Used to create a floating voltage higher than the supply to drive the n-channel MOSFETs on the highside.

**Analog Comparator**  
Multiplexes the floating phase to compare to the "zero" of the motor to deduce position in rotation cycle

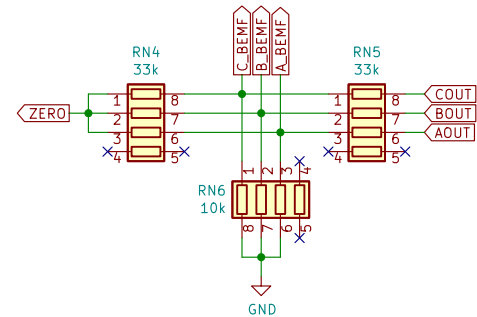


### Motor Control

FAN7x88 MOSFET driver drives each phase's half bridge according to the signals from ATmega.

Uses n-channel MOSFETs on high and low side due to their lower Rds and cost. However this requires "bootstrapping" their gates on the high-side for optimal performance.

MOSFETs specified are rated for 117A continuous current. However the traces are not! Add solder for safe operation.



### BEMF Dividers

Used to determine the position of the the rotor based on the back electromotive force (BEMF) measured in the floating phase of the motor. I won't delve into the theory here.

Resistor tolerances must be tight, so all phases have their voltage divided similarly to ensure fair comparison to the motor's "zero".

BYE BYE REGULATOR

Arduino Based BEMF Electric Speed Controller for a Brushless Motor  
Designed for input between 10V and 20V, up to 100A  
System controlled by PWM or I2C

### Title: 20V 100A BEMF ESC

|   |                  |            |
|---|------------------|------------|
| Savo Bajic  | Date: 2020-08-16 | Rev: 3     |
| KiCad E.D.A. kicad 6.0.1-79c1e3a40b-116-ubuntu21.04.1 | Size: USLetter   | Sheet: 1/1 |