ems level diagram. Hardware showing voltages/currents/signals, and software showing communication General Task List

11/2

* Good, detailed syst and calculations/functions. As much detail as possible.
* Evaluate whether motor driver design is good enough.
  + Establish nominal and extreme load cases
    - Conduct simulation in SPICE
    - Hand calculations to prove that overcurrent, overheating, or other malfunctions won’t happen to FETs.
    - Prepare for motor driver circuit testing: test plan with objective, parts, tools, procedure.
      * Learn how to use gate drivers
      * Write code to control motor speed
      * Obtain test load that simulates motor ( R, L, V\_induced (if possible))
      * Assemble circuit
      * Find breaking point
* Choose speed feedback method
  + Optical or magnetic encoder
  + Hall effect sensor
  + Back-EMF from motor
* Track down connector for camera
* Optimize code for light input for speed and robustness
  + Look at output from O-Scope
* Determine what/how to store line data
* Determine what/how to store track data
  + Goals:
    - Take very little time between memory and performance modes.
    - Remember track after power down.
    - Allow it to speed up coming out of turns and slow down before turns.
* Determine what raspberry pi zero accessories we need to use it. For testing (our own use) or for implementation on the actual car.
  + Adjust steering coefficients wirelessly from camera without needing to turn off code
    - Kp/ Ki/ Kd, for steering and speed.
  + Have a remote kill switch that
    - Cuts power to motor driver and remembers track
    - Cuts power to everything
  + Make sure it powers down safely
* Look for a better chassis: braking options, lower weight or lower center of gravity
* Find cheaper place to buy motor
* Find battery