|  |
| --- |
|  |
| QE #85 (1.50 hrs) 21Mar19 **[M]**  Topic: Outlining and fixing issues found in Main v1.1.   1. Issues found during the outlining:    1. Add a display time for DisplayNumber() and DisplayString()    2. Check if need to re-initialize the scale object (done during process to get code to compile)    3. Check loop on when to close the text file (reduce if possible, saves resources)    4. Check that MakeFileName is actually doing what is designed to do. 2. Fixes:    1. Add DisplayTime()    2. The scale object needs to done the way it is…idk compiler said so    3. File opens at beginning of loop and closes just before   **Outcome(s):** Have new fixes for the issues found in the outlining. |
| QE #86 (1.00 hrs) Mar19 **[A]**  Topic: Redefining set point based on new PPR.    The PPR chosen was 512, which has a max RPM of 15000.  Given the calculations done earlier the motor would rotate about 1473 rotations in a minute, which is 9.82% of the maximum RPM for that PPR level.  This is a safe setting, i.e. the motor speed will not break the upper limit for the RPM.    **Outcome(s):** Have the set point number (126) |
| QE #87 (2.50 hrs) 21Mar19 **[A]**  Topic: Motor driver remapping of pins in order to reduce pins allocated.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Motor Driver As-Is | | | |  | Cost/Benefit of Remap | | | Name | Pin # | Needed | Comment(s) | ………. | No. of pins used | 8 | | M1PWM | 9 |  | (max is 20 kHz) |  | No. of pins needed | 3 (37.5%) | | M1DIR | 7 |  |  |  | No. of pins wasted | 5 (62.5%) | | M1FB | A0 |  | For current sense |  | |  | 4 |  |  |  | |  | 12 |  | Diagnostic output |  | | M2FB | A1 |  |  |  | | M2DIR | 8 |  |  |  | | M2PWM | 10 |  |  |  |   New pinout:    Need to rewire the model circuit and adjust values in the code in order to do the prototype review, etc.    **Outcome(s):** Have new pinout (no overlap). MOVED TO VERSION 1.2! |
| QE #88 (1.00 hrs) 21Mar19 **[D]**  Topic: Documenting display codes.    **Outcome(s):** Have tables for display codes. |
| QE #89 (2.00 hrs) 21Mar19 **[T]** (Emily, Ian, and me)  Topic: Load cell testing, and discussion)   1. Got the tare down to the 0.02-0.05 range 2. The resistance from E+ to S+ was equal to E- to S+ was equal to E+ to S- was equal to E- to S- at 261.4 ohms    1. The load cell is balanced because 3. Paper by Encompass Automation & Engineering Technologies LLC ([Link](https://www.creamcityscale.com/sites/creamcityscale.local/private_files/TechDocs/Load%20cell%20Troubleshooting.pdf))    1. Section 5: Scale Calibration       1. Part E: Zero Range          1. Specifies the total percentage of the capacity of the range that can be zeroed (typically 2%, up to 20%)    2. Notes: Maybe the range of our load cell (up to 1000 lbs) is too much for the resolution that we need/are looking for 4. Emily will email Richard (lab tech) about trouble shooting the load cell 5. Emily expressed concern about getting the load cell to reach 3 in/min without jerking or taking too long. I suggested that we add an off set (and initial, preset motor speed that will assume zero weight and then the PID can take over to keep the speed near the set point. This offset idea will need to be experimentally compared to the current design. Also, the ‘offset’ will need to be experimentally derived.   **Outcome(s):** Emily will contact for help on load cell, and we have new method to test for motor control. |
| QE #90 (1.00 hrs) 21Mar19 **[D]**  Topic: Creating testing plan for motor driver.  [EE4820\_T\_MotorDriver](../../Testing/Documentation/EE4820_TestingTable_MotorDriver.docx)    **Outcome(s):** Have required documentation for the testing plan for the motor driver. |
| QE #91 (0.50 hrs) 26Mar19 **[T]**  Topic: Creating isolated code to test motor driver and linear actuator.    **Outcome(s):** Have code to test motor driver and linear actuator. |
| QE #92 (2.00 hrs) 26Mar19 **[T]**  Topic: Testing motor driver and integration of motor driver and linear actuator. And fixing Ian and Thommy’s soldering job. (The whole was in the room, but I was testing and Thommy was timing.    **Outcome(s):** Have validation of the motor driver and green light on next integration test. |
| QE #93 (1.00 hrs) 26Mar19 **[D]**  Topic: Email with De Souza and Scalzo about trouble shooting load cell.    **Outcome(s):** Initiated dialog with De Souza so he could help with sorting the load cell. |
| QE #94 (0.50 hrs) 26Mar19 **[M]**  Topic: Adjusting set point calculation.    **Outcome(s):** Have an adjusted formulation for set point using measured time and PPR count. |
| QE #95 (1.00 hrs) 26Mar19 **[M]**  Topic: Epoxying the tube and plate together.   1. Traced outline of the tube on the plate 2. Abraded the outline of the tube and the tube end 3. Applied epoxy and wiped up the excess a bit 4. Let cure for about 30 hours (recommended 24-48)   **Outcome(s):** Have the plate and tube bonded. |
| QE #96 (0.50 hrs) 27Mar19 **[GP&A]**  Topic: Meeting with De Souza on trouble shooting the load cell. (Dylan, Emily, and me)  He just suggested to measure the signals and try and manually excite the load cell/amplifier.  **Outcome(s):** Didn’t learn anything new, I will try a little more with the load cell/amplifier and then do the long way. |
| QE #97 (1.00 hrs) 27Mar19 **[T]**  Topic: Testing the encoder. (Dylan, Emily, and me)   1. Attempted to use code that was previously written, but was wrong    1. The encoder requires checking both channel A and B 2. Got code working and reading    1. see UP and DOWN for readouts    2. Need to re check the calculations dealing with the pulse readings    3. Getting ~5000 pulses (PPR = 512) instead of the 1472 expected.   **Outcome(s):** Have new encoder code and data to begin ensuring encoder functionality. |
| QE #98 (2.00 hrs) 27Mar19 **[A]**  Topic: Fixing/ ensuring that the calculations for set point are correct.   1. The calculation in the code for the SP outputs 0…obviously it shouldn’t be 0    1. Breaking it down    2. I think C is the issue as it is very small and is probably being called 0    3. Instead of putting the fraction I put the approximate decimal equivalent and it worked    4. SP = 125.69 2. Now to get the information to see the displacement in order to get speed    1. Got the encoder reading and sending the pulse count per measured period     **Outcome(s):** Have code updated and working, and set point calculation working. |
| QE #99 (2.00 hrs) 27Mar19 **[M]**  Topic: Wiring model circuit with new pinout.    **Outcome(s):** Have model circuit with new pinout. |
| QE #100 (1.50 hrs) 28Mar19 **[GP&A]**  Topic: Prototype review for professors.   1. Mello (TA)    1. Videoed and need to get video to DG 2. Scalzo    1. Ran down subassemblies with level of completion    2. Scalzo our grade is 2 (ahead of schedule) !!!    3. Capt. David rating is 1.8 (kind of ahead of schedule)   **Outcome(s):** Have professor rating on schedule and progression. |
| QE #101 (3.00 hrs) 28Mar19 **[A]**  Topic: Getting PID library to function (i.e. checking inputs/outputs).  not working…got it working. I had to set the mode to “automatic”  Looking for PID trends in functionality (SP = 125.69):   |  |  | | --- | --- | | EncoderReadOut\_Diff: | InputMotorSpeed\_UP | | 0 | 400.00 | | 20 | -71.17 | | 40 | -82.60 | | 60 | -96.03 | | 80 | -111.46 | | 100 | -128.89 | | 120 | -148.32 | | 140 | -169.75 |   These results don’t make sense. The PID should accelerate less as it gets closer to the SP, but it shouldn’t start going backward! This test was run twice and the exact same values were produced (i.e. the library works consistently).  Below are the results of running the PID with inputs varying as follows: dE1/dM1 = 20, dE2/dM2 = 10, dE3/dM3 = 5, dE4/dM4 = 2, dE5/dM5 = 1. “P\_ON\_E” is “proportional on error” and “P\_ON\_M” is “proportional on measurement.” (see [here](http://brettbeauregard.com/blog/2017/06/introducing-proportional-on-measurement/) for an explanation on those)      Testing for the smallest time needed to delay after the .compute() I started at 100 ms, then I ran: 25 ms, 30 ms, 40 ms, 45 ms, and 50 ms. Only 50 ms produced real outputs (i.e. outputs that weren’t just the maximum and/or minimum range values).  **Outcome(s):** Have the PID to outputting and working as expected. |
| QE #102 (6.00 hrs) 28Mar19 **[A]**  Topic: Trying to figure out why amplifier isn’t working. (see EE4820\_Testing\_LC\_RawCal)  Below are the results of how the load cell readings change with respect to the change in calibration\_factor. It was noticed that the readout would respond more when initially changing the calibration\_factor and less so when the calibration\_factor became greater (further from 0).    **The SCK**:  The clock for the amplifier is set by the user and is determined by channel for how the clock speed will affect the gain. Channel A can have gains of 64 or 128 dB, whereas channel B can have only a gain of 32 dB.   1. [HX711 datasheet](https://ns-electric.com/files/datasheets/HX711.pdf)    1. Description       1. Ch. A gains: 64 or 128 (voltage range: ±20 mV, ±40 mV)       2. Ch. B gain: 32 (voltage range: ±80 mV)       3. There is on-chip oscillator (can use that in order to set clock speed to adjust gains?)       4. HX711 can output at 10-80 SPS (Sample Per Second)    2. Application Example       1. It looks like they connected “E-“ to ground and not the AGND pin on the HX711          1. I tried that and the output oscillated    3. Varying calibration\_factor       1. = 0 => inf       2. = -1 => ~ 7 digits long (7,000,000)       3. =-10 => ~5 digits long (53 K)       4. =-100 => ~5 digits long (27 K)       5. =-1000 => ~4 digits long (-1.4 K)       6. =-10000 => ~2 digits long (23)…over responds to applied force (delayed and intermittent)       7. =-100000 => ~1 digits long (5)       8. =-100000 => ~1 digits long (1.18)…somewhat responds to applied force (delayed and intermittent)       9. =-1000000 => ~1 digits long (0.1)       10. =(pos) => makes readout negative       11. =100000 => ~1 digit long (-9.9) …responsive to applied force (delayed and intermittent)    4. Digging into what calibration\_factor actually is       1. All the “.cpp” says is “SCALE=scale” (c\_f is passed to set\_scale)       2. “SCALE” is used in get\_units() in that get\_value is divided by “SCALE”       3. Get\_value returns read\_average minus OFFSET       4. OFFSET is set by the average reading found by read\_average() in tare().    5. Removing the load cell       1. After removing the load cell and only having the amplifier the readout still oscillated       3. After looking at this I can tell 2 things:          1. the tare function isn’t functioning correctly because the readout should go to zero the way its written          2. The amplifier is the most likely problem.       4. Plan going forward: solder and test with the other amplifier    6. Trying the code from [Circuits4you](https://circuits4you.com/2016/11/25/hx711-arduino-load-cell/)       1. With no load cell and no re-tare (Raw 5.0)          1. Still non-zero and fluctuating       2. With no load cell and re-tare (Raw 5.1)          1. Small dip below 0, then back to about the same range as Raw 5.0       3. With load cell and no re-tare (Raw 5.3)          1. Has a clear line, but very scattered       4. With load cell and re-tare (Raw 5.4)          1. Has a clear line, but very scattered 2. Reading on why the amplifier might not work    1. [Arduino Forum](http://forum.arduino.cc/index.php?topic=341013.0)       1. Topic: not getting good/meaningful output       2. “aarg” said either the load cell isn’t suited for such (small) weights, the load cell isn’t compatible with the HX711, or wiring is incorrect (the signal is “extremely noisy”)    2. Honeywell       1. [Datasheet](https://measurementsensors.honeywell.com/ProductDocuments/Load/Model_34_Datasheet.pdf)       2. Stated bridge resistance: 350 Ω       3. Measured bridge resistance:          1. E- \ E+: 350.9 - 351.0 Ω          2. S- \ S+: 350.1-350.2 Ω          3. E+ / S+: 263.1 Ω          4. E+ / S-: 262.8 Ω          5. E- / S+: 263.1 Ω          6. E- / S-: 262.7 Ω       4. Calibration sheet resistance: 353 Ω       5. Mentions linearity form 1000 g to 250 lbs is ±0.15%.Maybe I need to load at least 1000 g in order to get it functioning properly.       6. Also, is specifically mentioned tension, so I will load 5 lbs pulling force and try to calibrate.       7. Excitation ranges:          1. 1 kg to 10 lbs: 5 Vdc          2. 25 lbs to 1000 lbs: 10 Vdc   New testing plan for load cell/amplifier ([link](../../Testing/Documentation/EE4820_TestingTable_LoadCellCalibrationRETRY.docx))  **Outcome(s):** Have a plan to try and resolve the amplifier issue, and a new testing plan. |
| QE #103 (0.50 hrs) 29Mar19 **[GP&A]**  Topic: Discussion with my father (has his PE and works in the “controls” area specializing in SIL calculation)   1. Load cell / amplifier    1. He said what I’m getting sounds like just noise, or maybe an error in code    2. He suggested that maybe doing the tare-ing, etc., myself 2. PID    1. The PID works, but I told him about the odd response at high speed    2. He reminded me that Kd variable resists change and if set too high will freak out    3. He suggested that after tuning I may need to dampen the system in order to make it not too sensitive   **Outcome(s):** Have some good notes on trouble shooting the load cell/amplifier issue and tuning the PID. |
| QE #104 (6.00 hrs) 29Mar19 **[D/R]**  Topic: Creating testing plan for PID.   1. [Autotuner library](https://playground.arduino.cc/Code/PIDAutotuneLibrary/)    1. PID\_ATune()       1. Arguments: pointer of input and output       2. Returns: none    2. GetKp() / GetKi() / GetKd()       1. Arguments: none       2. Returns: retrieves values    3. SetNoiseBand()/GetNoiseBand()       1. Arguments: level of noise ignored before stepping output / none       2. Returns: none / retrieves value    4. SetOutputStep()/GetOutputStep()       1. Arguments: how much output will be stepped above|below starting point (>0) / none       2. Returns: none / retrieves value    5. Runtime()       1. Arguments: none       2. Returns: 0 if tuner still needs to run, 1 if tuner is done    6. SetControlType()/GetControlType ()       1. Arguments: PI (0) / PID (1)       2. Returns: none / retrieves value of control type    7. [Example sketch](https://github.com/br3ttb/Arduino-PID-AutoTune-Library/blob/master/PID_AutoTune_v0/Examples/AutoTune_Example/AutoTune_Example.pde)       1. Libraries: PID\_v1 and PID\_AutoTune\_v0       2. Set input and output (need to get averages from real run or math {i.e in=125, out=150, sp=125})       3. Makes a way to either tune from a simulation or real world (i.e. useSimulation it true|false)       4. Global Variables:          1. Input = 80          2. Output = 50          3. SP = 180          4. Kp=2 / Ki=0.5 / Kd=2       5. Setup():          1. If(Simulation): Makes theta an array and puts ints 0 -> 49          2. AUTOMATIC mode       6. Loop()          1. While tuning: Runtime() and keep doing so until tuning is done          2. When it’s done tuning it gets the coefficients and passes it to the PID          3. If(tuning): autotune, end tuning loop, set PID, then AutoTuneHelper(false)????; else compute PID          4. If(useSimulation): set theta array to output, run DOMode1?; else analogWrite output       7. Functions          1. ChangeAutoTune(): if(!tuning): sets parameters and tunes; else cancel          2. AutoTuneHelper(Boolean start): if(start): GetMode; else SetMode(2)…!0 is auto          3. SerialSend(): prints parameters          4. DoMode1(): “cycle the dead time”, then compute the input…adds some noise (0.XX)    8. [Matlab Toolkit](https://www.mathworks.com/matlabcentral/fileexchange/4652-autotunerpid-toolkit?s_tid=mwa_osa_a)       1. The [help page](file:///C:\Users\mebar_000\Desktop\2018-2019\SPRG%2019\EE%204820%20Senoir%20Design%202\Code\MatLab%20PID\autotunerPID\help\index_doc.html)       2. The autotuner uses the ISA PID structure       3. Areas v Relay Method of retrieving a process description          1. Areas—gets model of plant via simple experiment. Uses integration (noise-insensitive) and pretty accurate. Has ability to estimate delays without forcing user to define thresholds.          2. Relay—identifies a set of “distinctive” features rather than a model.    9. Test runs on the Arduino code:       1. ATEx1/2: Input v Index is giving the typical proportional on error curve       2. ATEx3: (SP=125) gave expected results       3. ATEx4: (P\_ON\_M) worked, but I was expecting a different curve. It is probably that I don’t understand.       4. ATEx5: (starting input to 125) PID freaked out a bit at the begining       5. I would like to run the autotuner on the real set up if there is time, but the coefficients [2, 0.5, 2] seem to work just fine       6. ATExT: (testing those coeff. With my code) The response freaks out as the speed jumps up…I think it’s the Kd       7. ATExT2: (found coeff., but started at input = 125) the PID does *not* like that…maybe I need to have an output value that the PID adds/subtracts to…looking at series ‘blah’ (all coeff. Set to 1) on the plot the response is only shifted up by some multiplicative factor       8. ATExT3: (PID primed as if there was readings from startup) not successful    10. I need to test the controller on the system and experimentally verify its functionality        1. What I need to get data for:           1. Have a stepped input and the read speed (for the Matlab tool for further tuning)           2. Allow PID to do “its thing” and then check if it worked or not     **Outcome(s):** Have required documentation for the testing plan for the motor driver. |
| QE #105 (2.00 hrs) 31Mar19 **[M]**  Topic: Manufacturing the plates for the battery pack.    1.Conducting plate  2.Transimission wire connection system  3.(Still to be mounted) Battery retention springs  The plate (most likely Aluminum) was cut to fit the width of the battery pack, the height, and then tapped.  **Outcome(s):** Have almost everything for the battery pack manufactured. |
| QE #106 (4.00 hrs) 31Mar19 **[T]**  Topic: “Manually” (self-written code, not using the library) calibrating the amplifier.   1. This attempt will use code that I write, using as little of the library as possible in order to get the load cell calibrated. 2. DIY    1. I printed values in the setup and limited the sampling to 10 entries for calibration\_factors of -96650 (1) and -7050 (2) respectively.    2. Results:       1. Read\_average() both returned -1638       2. Get\_value(): (1) returned 0 and (2) returned 819.00       3. Get\_units(10): (1) returned 819.00 and (2) returned -1638.00       4. Get\_scale(): both returned 1       5. Get\_offset(): (1) returned -819.00 and (2) returned 0       6. Zero\_factor: both returned -819       8. Using tare(10):          1. Read\_average() (1) returned -819 and (2) returned -1638          2. Get\_value(): (1) returned 0 and (2) returned 1637.00          3. Get\_units(10): (1) returned -819.00 and (2) returned 820          4. Get\_scale(): both returned 1          5. Get\_offset(): (1) returned 0 and (2) returned -1637          6. Zero\_factor: (1) returned 417791 and (2) returned 0    3. I have not been able to solve the issue, still need to pursue manual calibration   **Outcome(s):** I have not been able to solve the issue, still need to pursue manual calibration. |
| QE #107 (1.00 hrs) 1Apr19 **[T]**  Topic: Full scale test of battery system (me and Emily)  [doc](../../Testing/Documentation/EE4820_TestingTable_BatterySystem.docx)    Voltage output was about 12.3 Vdc (batteries not all fully charged) and the motor was driven as expected. Afterward, I attempted to attach the springs with solder, but the solder wouldn’t stick. We are going to have to find some small screws.  **Outcome(s):** Have required documentation and full scale testing done for battery system. |
| QE #108 (2.00 hrs) 1Apr19 **[A]**  Topic: Reading into the amplifier from [Honeywell](https://measurementsensors.honeywell.com/Pages/Category.aspx?cat=Honeywell&category=PRODUCTTYPES-INSTRUMENTS-INLINEAMPLIFIERS-PlasticorDieCast).   1. [U3W](https://measurementsensors.honeywell.com/ProductDocuments/Instruments/Model_U3W_Datasheet.pdf)    1. “Bridge-based” with transducer excitation at 3 or Vdc    2. Sounds like a much nicer HX711 because the broad specs are close    3. Has built-in noise reduction (65 dB signal-to-noise ratio) 2. Honeywell has several amplifiers and it looks like we can put the wires by unscrewing a plate 3. Just to make sure I’m not insane:    1. Reading through A&D’s [pdf](https://www.aandd.jp/products/weighing/loadcell/introduction/pdf/6-1.pdf) on using a load cell       1. Rated output in mV/V       2. Excitation voltage in V       3. Output range up to (rated x excitation) [in this case: 10 V x 2 mV/V = 20 mV]       4. Sensitivity:          1. Looking at the linearity from 1kg to 250 lbs and assuming that is the minimum force:          2. The is about a third of a pound (0.372 lbs)          3. The range would be from [-20, 20] mV; therefore, there would be . I want to try this as the calibration\_factor because this is what divides get\_value in order to get\_units       5. Plugged in values into the accuracy equation given by A&D and got depending on whether or not temperature change will not affect (common if taring before measure).   **Outcome(s):** Have not determined anything new, Thommy has reached out to Honeywell. |
| QE #109 (3.00 hrs) 1Apr19 **[A]**  Topic: Determining the point of error in the amplifier.   1. Hardware    1. The hardware seems to check out    2. I tested the individual devices on the chip and did ***not*** find a short circuited one    3. The voltages seemed reasonable (comparing to Honeywell’s datasheet)    4. Both chips seem to be running the same    5. **I believe that is it NOT a hardware *malfunction*, which is not to say it is not a hardware *issue* (maybe the wrong amplifier)** 2. Software    1. I noticed an interesting phenomena: when run along read\_average() returns almost always zeros (randomly a 10 digit number rarely), but when get\_units() is run with read\_average(), read\_average() becomes crazy       1. Giving a delay(100) between the read\_average(10) and the get\_units(10) did not seem to help the readings       2. Correlation between reading of r\_a and g\_u?          1. Doesn’t appear to be any…when r\_a is wild g\_u is 0.000          2. Common wild values for r\_a: 1120121200220, 210022100110, 12022211121, 120011, 21120, 10210, 2122220210, 22222201, 1010100             1. Suspect, but do not appear to be meaningful (I translated them to binary and hexadecimal and didn’t see anything particular)          3. Running just g\_u(10) I get a negative, non-zero value about every 5th reading (~80 ms, ~120 ms, ~200 ms)…this cycle repeats             1. G\_u uses 2 other functions and one as a proxy, so this pattern is not explained.          4. Running just a\_r(10) I get a positive, non-zero (very large) value about every 9-10 readings, but is not as repeatable.             1. This less repeatable pattern might be due to the structure of the a\_r, where a\_r reads for a set period and then returns          5. Running g\_u and a\_r without a “times” argument passed to them results in a less noisy result…             1. The 80/120 pattern reemerges             2. Running them together still gets a\_R to flip out, but where g\_u freaks a\_r does’t!          6. Running g\_u along, one time outputs a constant 0.000             1. G\_u(10) only returned one negative, non-zero value (1 time out of 10 times run)          7. Running g\_u(10) and adding weight (5 lbs) the 80/120 pattern appears             1. Allowing to tare(10) first, then adding weight: same          8. Running g\_u(10) and adding weight (10 lbs) the pattern changed to 40/100             1. Allowing to tare(10) first, then adding weight: same          9. Running g\_u(10), allowing to tare(10) with (5 lbs), then adding 5 lbs the 40/100 pattern appears             1. I did notice that wherever there was (not always) a “weird” value there were 2 readings for the same timestamp…             2. There were several readings from the same timestamp that simple read 0.000, so this does not seem like a conclusive indicator (see [doc](../../Testing/LoadCellAmp_CALIBRATION/EE4820_Testing_LC_idk3.txt))          10. Running tare(10) and printing g\_u(10) and a\_r() and a\_r(10) ([idk4](../../Testing/LoadCellAmp_CALIBRATION/EE4820_Testing_LC_idk4.txt) and [idk5](../../Testing/LoadCellAmp_CALIBRATION/EE4820_Testing_LC_idk5.txt))              1. Idk5: the g\_u(10) builds up in a pyramid fashion and decrease significantly each pyramid. A\_r(10) repeats values..idk          11. Running with c\_f as -2690 per QE#108 and gain as 128 (HX711 [datasheet](https://ns-electric.com/files/datasheets/HX711.pdf) says that’s for ±40 mV full-scale differential input), tare(10), and g\_u(10)              1. Didn’t seem to work   **Outcome(s):** Have no new progress, but have even more data. |
| QE #110 (3.00 hrs) 2Apr19 **[T]**  Topic: Testing PID with encoder feedback and using motor driver. (Emily and me)     1. The buttons weren’t working (worked when isolated) 2. The DownHook() function was deleted because it was just used to call MotorDown() 3. I found that I needed to set the motor driver first (also, once set the motor driver continues until changed) 4. The PID would then output, but is quickly maxed out…    1. We noticed that the motor was running sluggish…(tested motors with the power source and they ran fine) 5. The piece used to attach to the encoder is loose, and rattles and shakes.    1. Emily thinks that it shouldn’t be too much of an issue, also she plans on making a longer version in order to support extra. 6. Ian apparently “forgot” to show up at 1330 (test started at 1300) like he said…when asked at 1500 where he was…he was “getting his hair cut”   **Outcome(s):** Have another failed test. Components worked individually, but not integrated. |
| QE #111 (4.00 hrs) 2Apr19 **[T]**  Topic: Testing load cell and amplifier.   1. Load cell outputs?    1. When powered by external 10 Vdc (9.95) regulator 🡪 S+/- is ~4.98 Vdc    2. When powered by amplifier: ~2.9 Vdc 2. Using external voltage regulator and the amplifier board    1. Mostly 0.000s, but a occasionally blip of crazy numbers (lasting about 1-3 seconds)    2. I’m able to elicit some response from the load cell, but the readout isn’t continuous or seemingly correct    3. It looks like the readout is close for the first 2 readouts (maybe increase the a\_v arguement    4. Alright,  its reading about 2x what it should.    5. When tare(10) with some weight of ~2.5 lbs, then added a ~5 lbs      * 1. Test setup on jerry-rigged 100 (~2.5 lbs for tare, then add ~10 lbs):      1. Gain = 64      2. Gain = 128      3. Gain = 64, but no ~2.5 lbs for tare(10)      4. Letting it tare(10) with no weight, then adding the ~2.5 lbs c\_f = -3000      5. **NEED TO CHECK WITHOUT VOLTAGE REGULATOR**  1. Load cell with 250 ohm resistor (measure over res)? **Didn’t try because I got it working…** 2. The motor driver isn’t working:    1. The driver started having ~5.9 V and 0.45 A and the output was half of that    2. We changed the Arduino (not Arduino), changed source for motor (not motor), and changed code (not code)    3. It wouldn’t work with the buttons, and then it just wouldn’t work    4. Going to get Thommy to help figure it out (lol) 3. Soldered motor driver (tested continuity once plugged into the Arduino)    1. The soldering job wasn’t…my best. Also, soldering big holes is pretty difficult and required that I tack the solder to one of the smaller holes.   **Outcome(s):** Have a working load cell/amplifier, and have switched over the motor driver. |
| QE #112 (0.00 hrs) 3Apr19 **[T]**  Topic: Determining why the motor driver ain’t working.   1. Running motor driver test code (isolated) with both motor driver boards    1. Thommy’s motor driver (TMD):  |  |  |  |  |  | | --- | --- | --- | --- | --- | | Item | Not Running | Running (-155) | Running (155) | Running (-155) | | VIN | 12.58 | 10.8 | “ | ” | | GND | 0.001 | “ | “ | “ | | M1A | 0.434 | 4.05 | “ | “ | | M1B | 0.434 | 0 | “ | “ | | PS | +12.6 | 10.8 | “ | “ |  * + 1. Seems to be working normally…idfk   1. Garrett’s motor driver (GMD):  |  |  |  |  |  | | --- | --- | --- | --- | --- | | Item | Not Running | Running (-155) | Running (155) | Running (-155) | | VIN | 12.58 | 10.8 | “ | ” | | GND | 0.001 | “ | “ | “ | | M1A | 0.434 | 4.05 | “ | “ | | M1B | 0.434 | 0 | “ | “ | | PS | +12.6 | 10.8 | “ | “ |  * + 1. Seems  1. Adding PID controller to it (first checking pinout to ensure that there are not stray signals)    1. Must run the whole function GRS iot get ero\_d…seems like the 125 sp when mt = 10    2. The grs() is returning the ero\_d, but that value isn’t being stored; and therefore, need to store the ero\_d into a variable       1. The grs() will return the ero\_d to the address in memory…       2. I could just set ero\_d to grs()…both will be tried (nope, just the variable)    3. Trying just setting ero\_d to grs(): works   **Outcome(s):** Have the motor driver sorted and GetRotationalSpeed() working. |
| QE #113 (0.50 hrs) 3Apr19 **[D]**  Topic: BTB#5    **Outcome(s):** Have required documentation for BTB#5. |
|  |