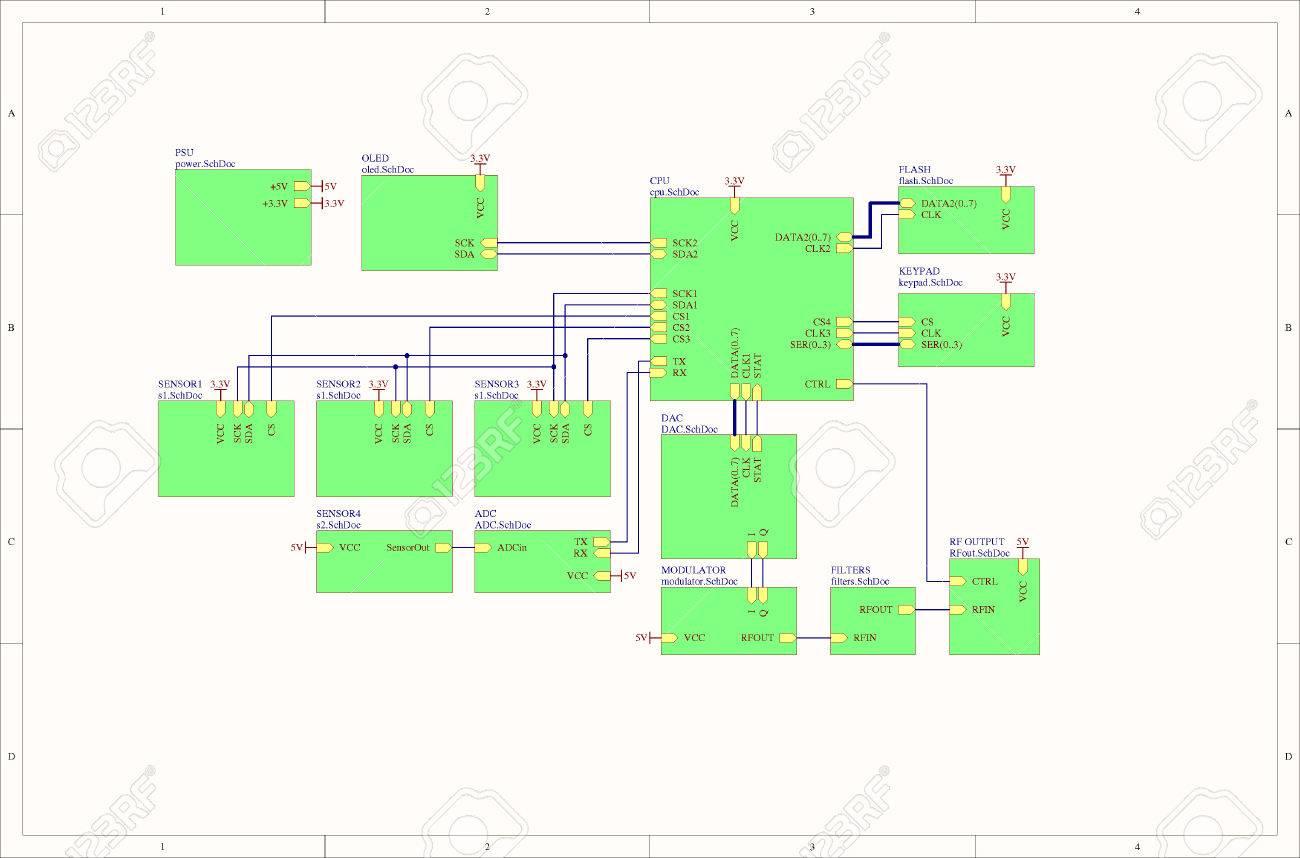
Things

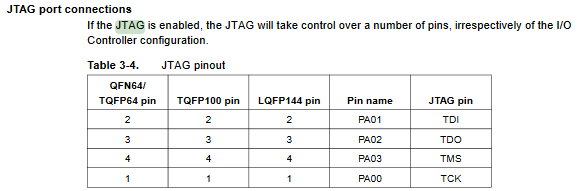
* sample block diagram  
  [1300 × 858](https://www.123rf.com/photo_52671445_real-electronic-embedded-system-concept-block-diagram.html)
  + our components: 1 CPU, 1 distance sensor, 3 IR sensors, 2 accelerometers, 2 motors, 1 display, 2 voltage regulator, batteries, 1 crystal oscillator
  + criteria: comms protocols, # of lines inside buses, connection labels
* Sparkfun line-follower schematic, we may want to reference this if we build our own module using the QRE1113GR sensors  
  <https://www.sparkfun.com/products/11769>  
  <http://cdn.sparkfun.com/datasheets/Sensors/Infrared/RedBot_Line_Sensor.pdf>
* Block Diagram components:
  + CPU
  + batteries
  + 2 voltage regulators
  + 2 distance sensors
  + 3 IR sensors
  + 2 accelerometers
  + 2 DC brushed motors
  + 1 H-bridge
  + 1 reset button
  + 1 power LED
  + 1 display mode button
  + 1 16x2 LCD character display
  + 1 CPU programmer port
    - 6-pin header
    - 
  + 1 crystal oscillator
  + 1 serial port (for log export)
  + \*\*!\*\* pin headers wherever possible
* budget $1000.00
* what pad size for SMD?
  + should be ~1.5 to 3x longer than heel-to-toe measurement
  + datasheets will often give a recommended size

Communications Protocols

* serial port descriptions: <http://www.ece.northwestern.edu/local-apps/matlabhelp/techdoc/matlab_external/ch_seri7.html>

Meeting w/ Prof. Tandon [11/7/2019 2:00 PM]

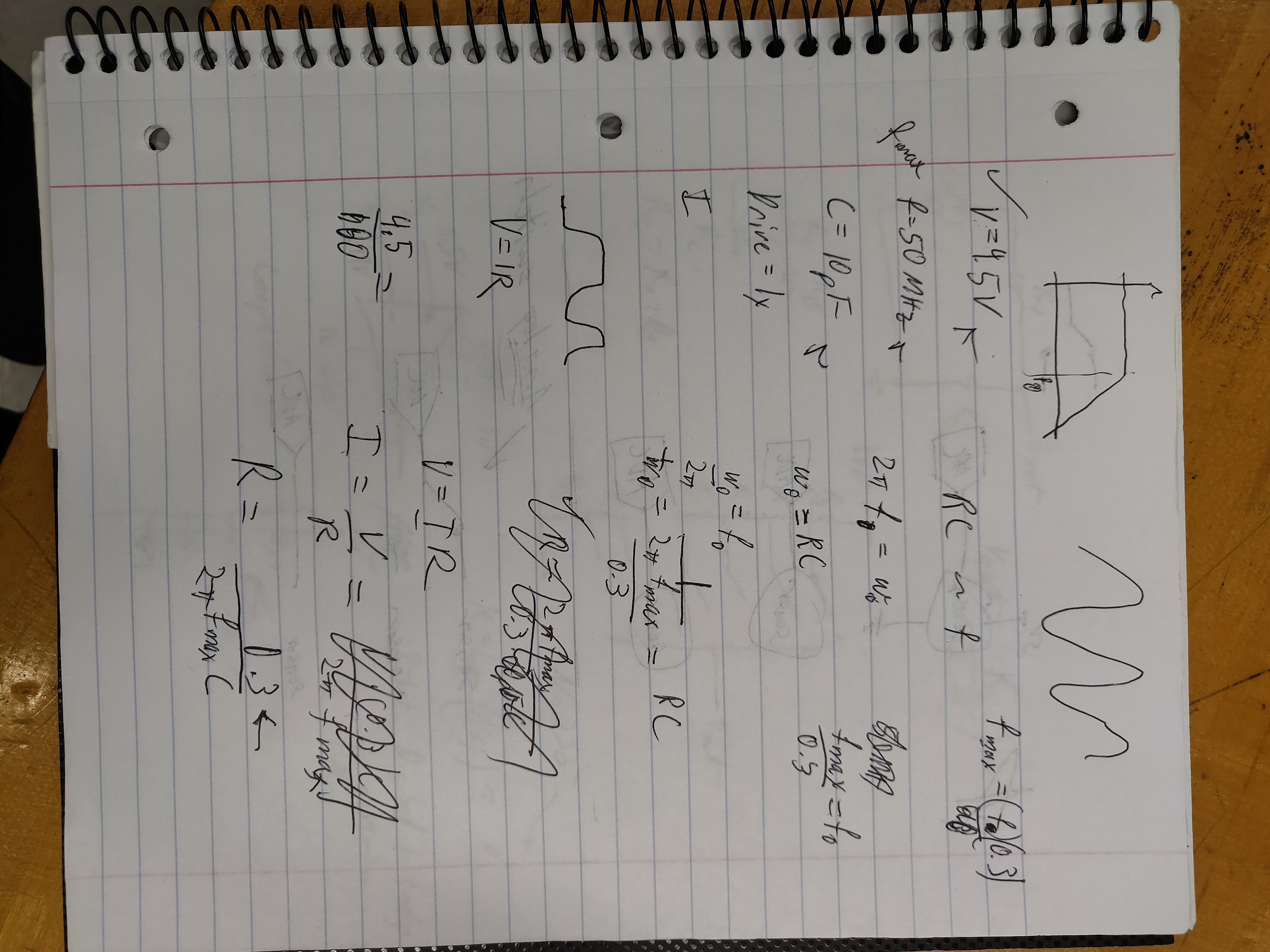
* Read up on ‘PID loops’
  + https://www.csimn.com/CSI\_pages/PIDforDummies.html
* Take into consideration the number of instructions in our Line Follower control loop, needs to be fast enough to clock our ADC to react quickly
* Build your own level shifter onto your board using the schematic of the i2c lvl shifter
* Ultrasonic lvl shifter unnecessary
* JTAG Interface on processor (32-bit AVR)



\* Dr. Tandon has a programmer we can use

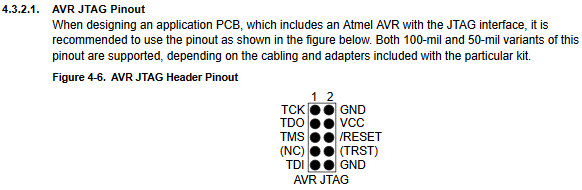
* LCD character display
  + configure serial to 9600 baud
  + connect the JST cable to the backpack as follows: Red wire goes to +5V, Black wire goes to Ground, White data wire goes to digital #2. You can change the digital pin later, but stick with #2 for this example and adjust later as desired.

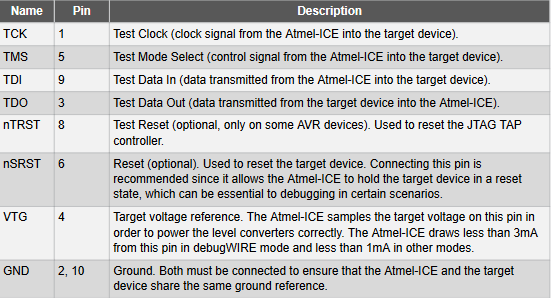
RLC Things

* RLC sizes: look for codes 0603 or larger, 0805 ideal
* [refer to table 40-6 while solving for Resistor values on buttons and LEDs]
* parts sizes
  + 0805 == 2012 metric
* 

Programming the CPU

* need to find a programmer device, should be ~$15.00 or lower
  + Maybe we can use Atmel-ICE?
    - Very expensive, 94 dollars
    - <https://www.mouser.com/ProductDetail/Microchip-Technology-Atmel/ATATMEL-ICE-BASIC?qs=KLFHFgXTQiAG498QgmqIdw%3D%3D>

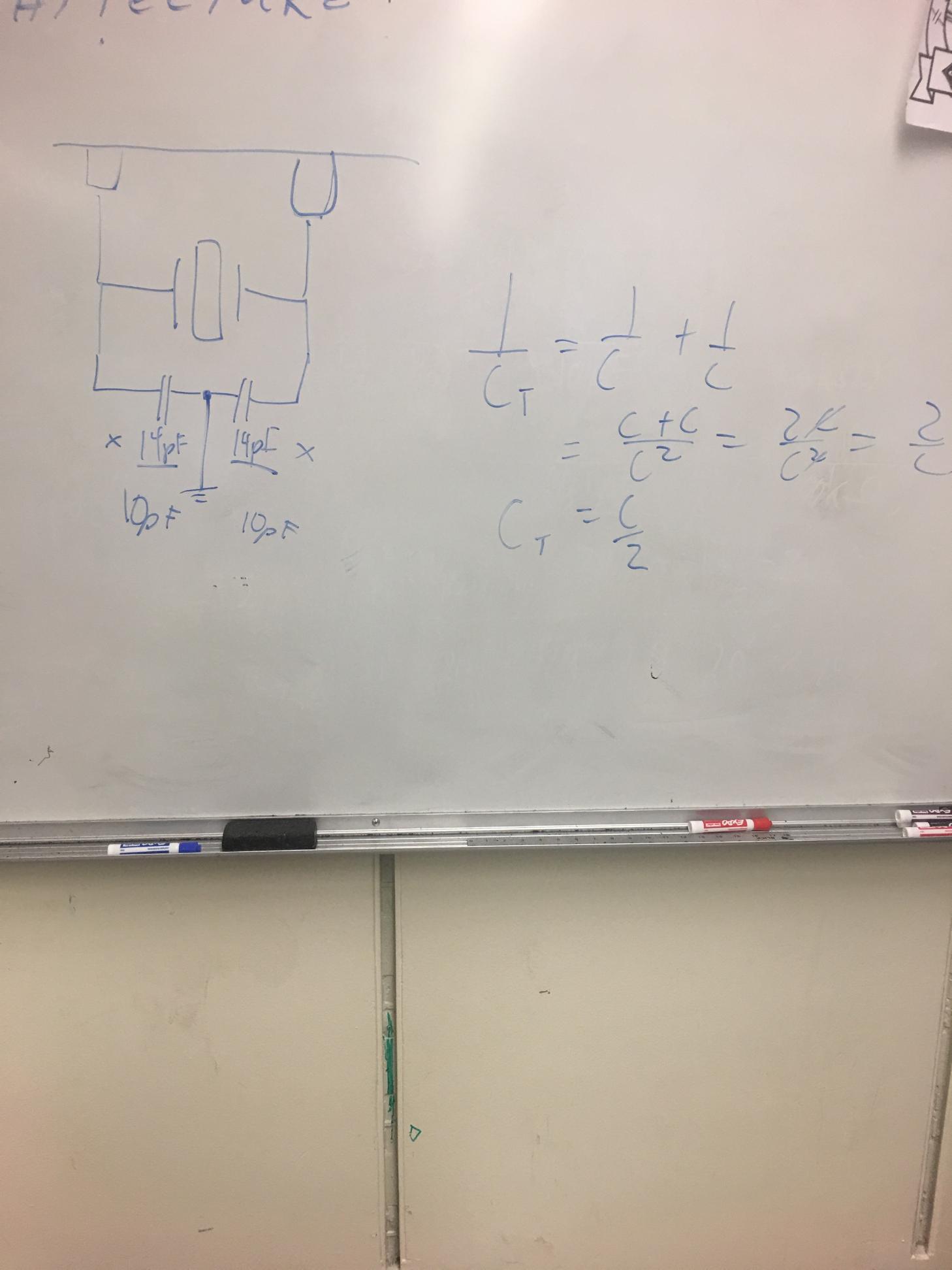




Pin Headers and Port Connectors

* DB9 DSub connector, use w/ max232 or max233
  + We’ll be using the SparkFun RS232 Shifter module
* if needed, generic DB9 connector can be found in EAGLE lib: con-subd → M09
  + Purchasing Sabrent usb to serial connector
* Need this pin header for JTAG connection
  + Programmer uses 50-mil pin spacing (1.27mm)
  + <https://www.digikey.com/product-detail/en/sullins-connector-solutions/GRPB052VWQP-RC/S9013E-05-ND/1786428>

Oscillator things

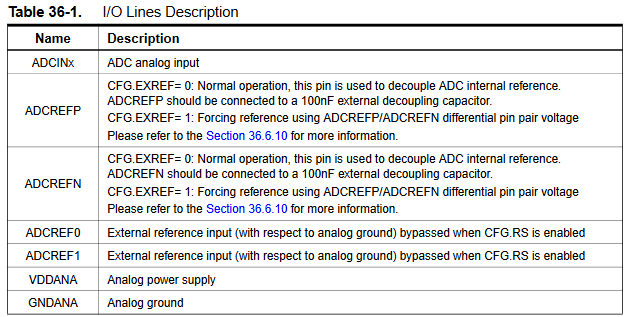
* <https://www.sparkfun.com/tutorials/95>
* External Oscillator
  + oscillator pins on pads PB30 and PB31
  + see CPU datasheet table 40-7 for freq-period-capacitance
    - max freq = 50 MHz
    - min period = 20 ns
    - crystal input capacitance = 2 pF
  + external freq must be < 4.5x system clock
  + duration of each of its levels must be longer than the CLK\_TC period
  + external clock freq must be at least 2.5x lower than the CLK\_TC
* look at shunt capacitance to get bypass capacitor values
* 
  + Crystal oscillator circuit

CPU things

* 45 GPIO pins on the processor
  + See table 3.2 in datasheet
* we’ll be handling floating point numbers for travel logging functionality
  + floating points occupy 32-bits
* we’ll need <= 32-bit CPU
* criteria:
  + <= 32-bit
  + not BGA or QFN packaging/mounting
  + interfaces: UART/USART, I2C, PWM, ADC
* attempting to calculate memory required:
  + given 10 min logging period
  + ↑ = 600 sec
  + 1 measurement = 1 floating-pt = 32 bits = 4 bytes
  + 2 measurements / second (1 measurement / sec may be sufficient)
  + 6 metrics to track (accel, vel, pos, total dist, avg spd, avg accel)
  + 4 \* 600 \* 2 \* 6 = 29 KBytes  
    bytes \* sec \* Hz \* metrics
  + ↑ + 20% for margin = 35 KBytes
* [see Figure 6-1 RE: VDD and GND pins]
* pins VBUS, DM, and DP seem to be used for USB connection stuff
* check the AVR programmer port connector shape, match your port shape to the programmer’s

ADC Things

CPU ADCs



* ADC Inputs/Outputs for the processor
  + (Apparently the built in ADC is a switching one?)
  + ADCINx - pin 7, 8, 9, 10, 11, 12, 16, 19, 20, 21, 22
    - ADCIN0, ADCIN1, …, ADCIN10 respectively
  + ADCREFN - pin
  + ADCREF0 - pin 13
  + ADCREF1 - N/A on our CPU
  + VDDANA - pin 18
  + GNDANA - pin 17
* Sect. 36.6.4  
  clues on using the two ADCs simultaneously (dual sequencer mode)
* dual sequencer mode
  + can handle conversions for two separate analog inputs
  + 8-state (bit?) resolution
* [see CPU datasheet table 40-27 for ADC sampling freq.]

External ADC

* ADC IC has sample circuits on the datasheet
* [see Paragraph 3 of Layout section on ADS7816 datasheet for bypass cap. specs.]
* inputs: Vref, +In, -In, (CS)’/SHDN, DCLOCK
* output: Dout
* DCLOCK sets the tempo for ADC comms and operations
* (CS)’/SHDN
  + relates to the tempo for when the ADC sends its conversions
  + second clock pulse after falling edge of (CS)’/SHDN enables serial output
  + new conversion is initiated only when CS is taken HIGH and returned LOW
  + falling CS signal initiates conversion and data transfer
* we may want to set the external ADC’s resolution to 8-bits (clock CS to every 8th DCLOCK cycle) to keep ADC resolution consistent w/ the CPU’s ADC resolution
* Both BypC’s necessary, keep them close to their respective terminals
* DCLOCK signal can be supplied on GPIO pins
* can probably exclude this from the schematic

IR Things

* IR reflective sensor
  + analog output (interface)
  + will need one ADC per sensor, or make use of analog pole switches
  + check how many ADCs available on CPU (There are two of them in CPU and we have added one ADC in BOM)

IMU Module Things

Ultrasonic Things

* HC-SR04 Ultrasonic Distance Sensor
  + Uses custom interface, echo-trig
* hook up to GPIO pins

Routing/Trace Things

Motor Trace Width

* load current: 145 mA
* starting current: 874 mA
* voltage: 5 V
* <https://www.4pcb.com/trace-width-calculator.html>

Battery Trace Width

* 6 V nominal

Via Usage and Plane Crossing Guidelines

* use drill holes of 3mm for mounting screw holes

Motor Control Things

Motor Noise

* keep motor Vs away from CPU/logical Vss
* motors can cause bad noise