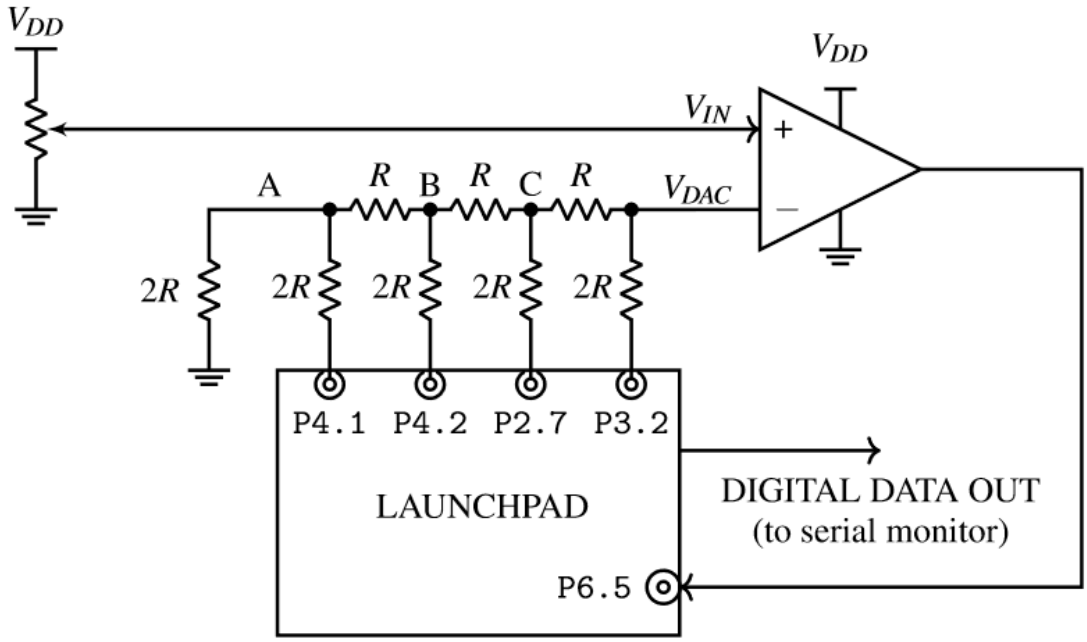
Han's ChemE Car Journal

3/19/21

- Familiarized myself with Autodesk Eagle Layers:
  - https://www.autodesk.com/products/eagle/blog/every-layer-explained-autodesk-eagle/
- Skimmed reading on PCB Design:
  - https://www.instructables.com/From-Idea-to-Reality-How-to-Design-Circuits-and-Cr/
- Started practice on Eagle building the ADC/DAC
  - 

4/2/21

- Built the schematic on Autodesk Eagle

4/9/21

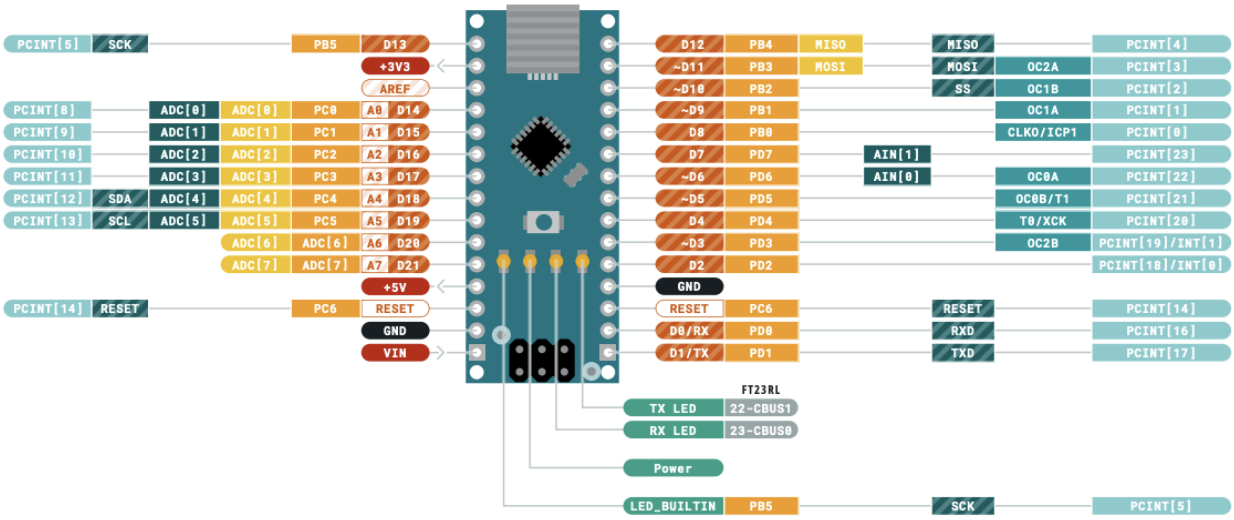
- Finished DAC ;)
- Started designing a board for:
  - Encoder read-outs
  - Voltmeter
  - SD card
    - https://www.amazon.com/gp/product/B07ZB9ZDHB/ref=ppx_yo_dt_b_asin_title_o02_s00?ie=UTF8&psc=1

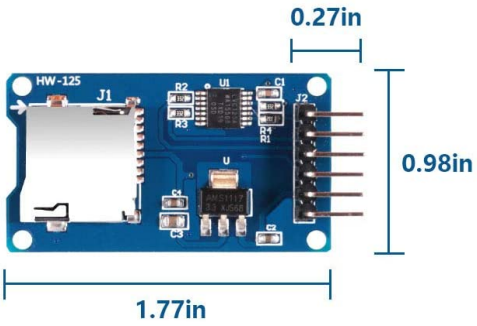
        - Datasheet
        - Tutorial
    - Extra pins for other possible peripherals

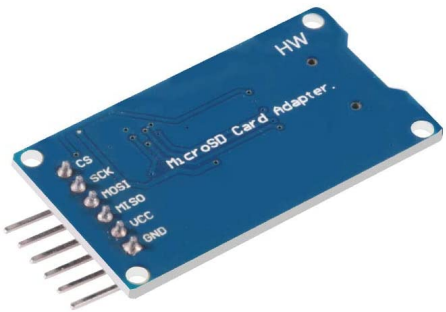
4/16/21

Unit Research

- Arduino Nano Pinout:



- SD Card:
    - 

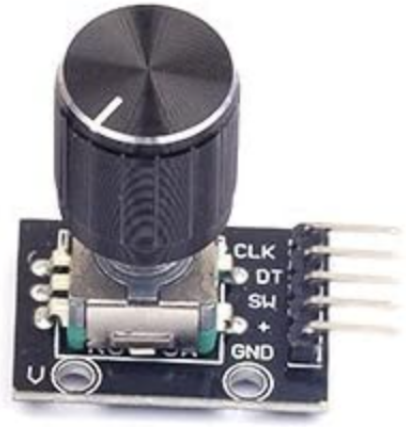
    - 

    - Supports MicroSB and MicroSDHC

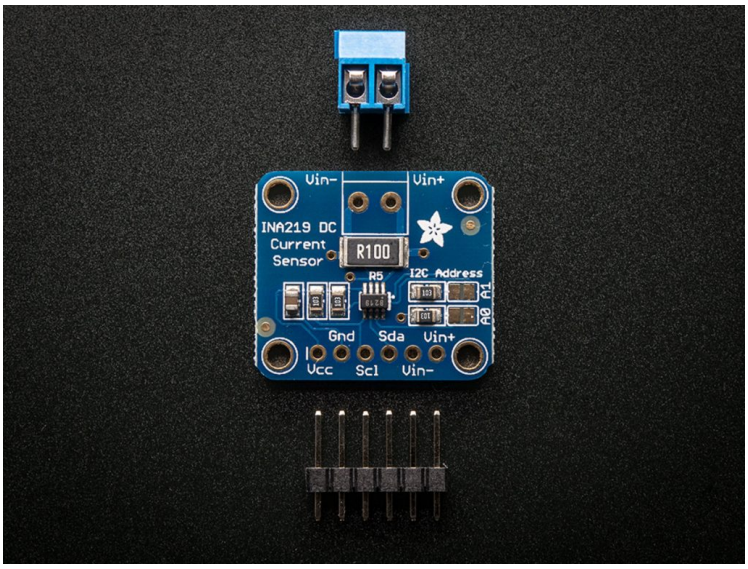
- **Vcc:** 4.5 V ~ 5.5 V
- MISO, MOSI, SCK, for SPI Bus on Arduino
  - Confirmed that Arduino Nano supports this!
    - Can't change the SPI pins tho sad
  - **MISO (Nano Pin 12):** Master In Slave Out - Slave Line for sending data to master
  - **MOSI (Nano Pin 11):** Master Out Slave In - The Master Line for sending data to the peripherals
  - **SCK (Nano Pin 13):** Serial Clock to synchronize data transmission generated by the master
  - **SS/CS (Nano Pin 10):** The pin on each device that the master uses to enable and disable specific devices
    - SS/CS Pin LOW: Slave communicates with master
    - SS/CS Pin High: Slave Ignores master
    - Allows for multiple SPI devices to share the same MISO, MOSI, and CLK/SCK lines
    - CS: Chip Select Signal Pin
- Output: 3.3 V
- Current: 0.2 - *80* - 200 mA
  - Nano: DC Current per I/O Pins = 40 mA - OK
  - Assume 5V pin current is ok ;)

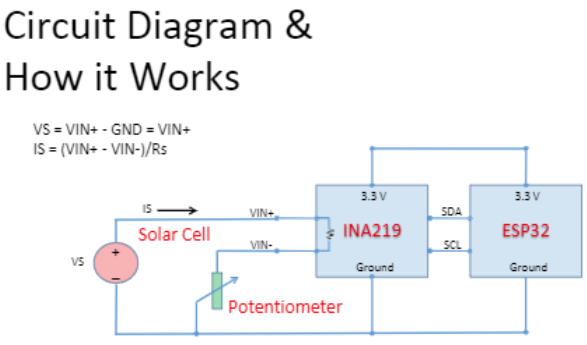
4/23/21

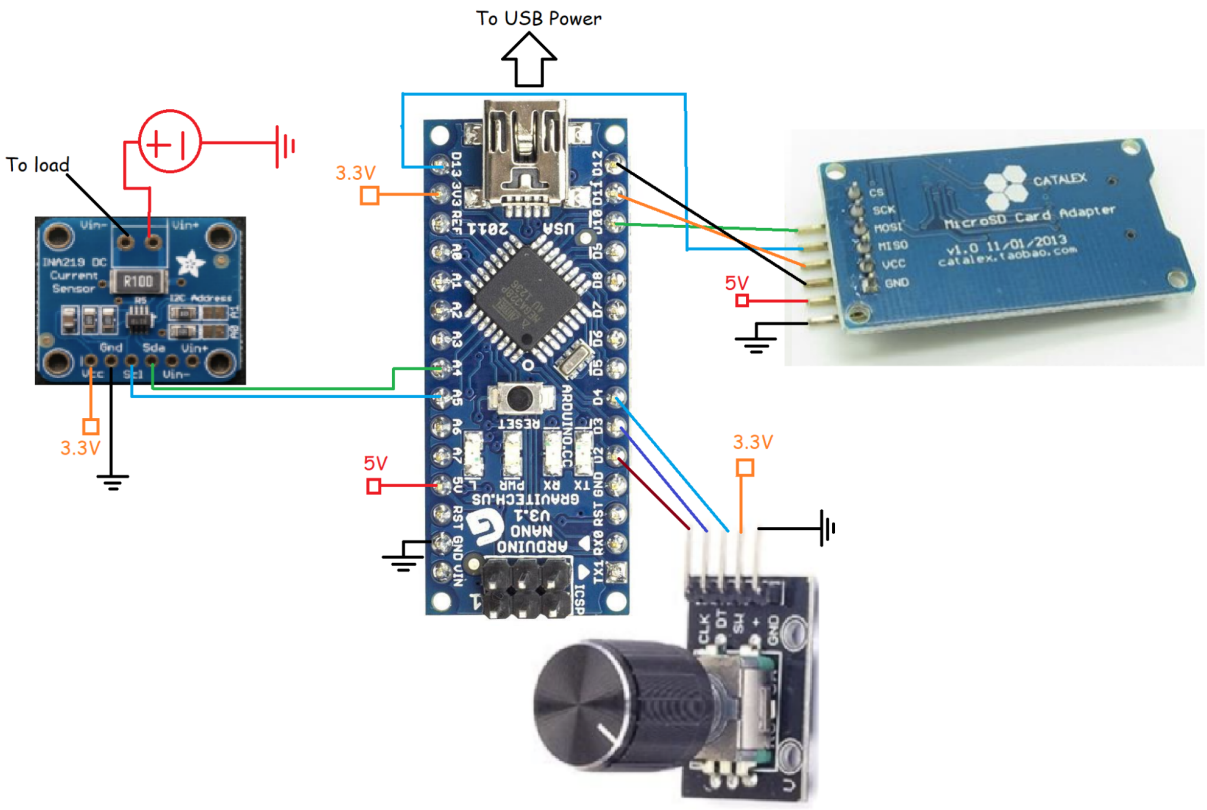
Research on encoders:

- Rotary Encoders:
  - 

  - Datasheet?
  - https://lastminuteengineers.com/rotary-encoder-arduino-tutorial/
  - **Pins:**
    - Vcc: 3.3 V - 5V
    - GND
    - CLK (Output A) is the primary output pulse for determining the amount of rotation. Each time the knob is rotated by one detent (click) in either direction, the 'CLK' output goes through one cycle of going HIGH and then LOW.
    - DT (Output B) is the same as the CLK output, but it lags the CLK by a 90° phase shift. This output can be used to determine the direction of rotation.
    - SW is the active low push button switch output. When the knob is pushed, the voltage goes LOW.
    - Note: No special communication protocol pins are needed. Use normal digitalRead pins on Arduino (No PWM).

- Voltmeter & Ammeter: [INA219]
  - 

  - https://www.adafruit.com/product/904
  - https://learn.adafruit.com/adafruit-ina219-current-sensor-breakout
  - High Side DC Current Sensor Breakout – 26V ±3.2A Max
  - Uses a precision amplifier to measure the voltage across a current-sensing resistor -> current
  - Analog current measure converted to a digital reading with onboard 12-bit ADC
  - Uses I2C communication: There is an Adafruit library for nano
  - **Pins:**
    - Vcc: 3.3 V
    - Gnd: GND
    - Scl: Nano pin A5
    - Sda: Nano pin A4
  - Ex:
    - 



Note: The INA219 requires a series connection and we can't detach this. Next time, we should use something else.

4/30/32

- Received arduino library from Nicole
    - No longer have to do my own dimensioning and create the nano on Eagle lmao
- Let's find an actual voltmeter wooooo
    - Idea: Arduino Nano analogRead() can measure 0-5V
    - Since battery is 0 - 12 V, we can use a resistive divider that will linearly scale the actual reading by a known factor
    - Measuring the output of the divider, we can determine the actual voltage by inverting the scaling factor
    - This last session tho so we won't do it right now lol