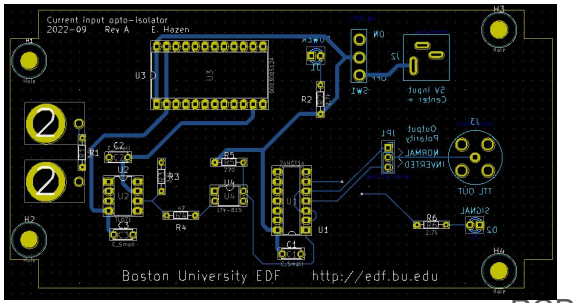
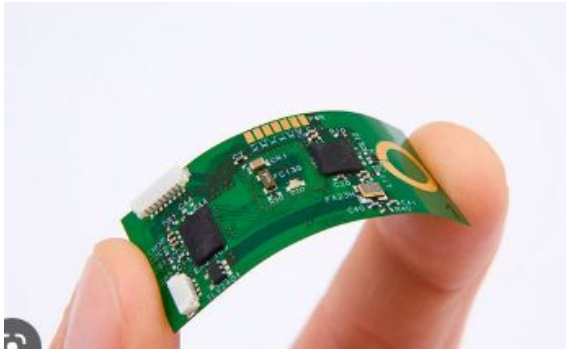
BOSTON UNIVERSITY

Design Project

High-performance analog Arduino Shield





Project Goals

- Give you a chance to exercise your skills by laying out a board
- Provide something useful for this class and other projects

This is *not* a circuit design exercise, or a detailed Arduino tutorial.
I'll give a very quick intro to Arduinos, microcontrollers, and analog/digital conversion to orient you.

The meat of the project is your work designing a simple PCB given a sketched schematic.

You will turn in:

- KiCAD project with complete schematic (required)
- KiCAD PCB layout (extra credit).

Problem Description, Specifications

Problem: The Arduino Uno (TM) is very convenient, but the analog input and output functions have (very) poor performance.

Specifications for new Shield board:

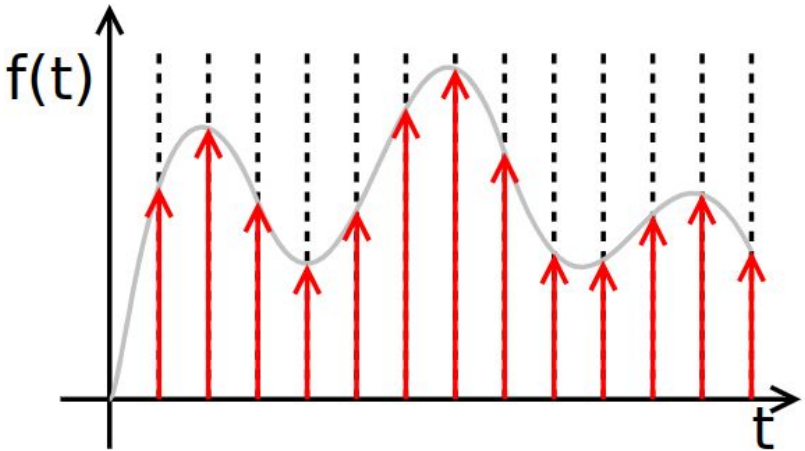
- 4 analog input and 4 analog output channels
- Improved resolution
- Improved sampling rate
- Improved S/N (signal-to-noise ratio)

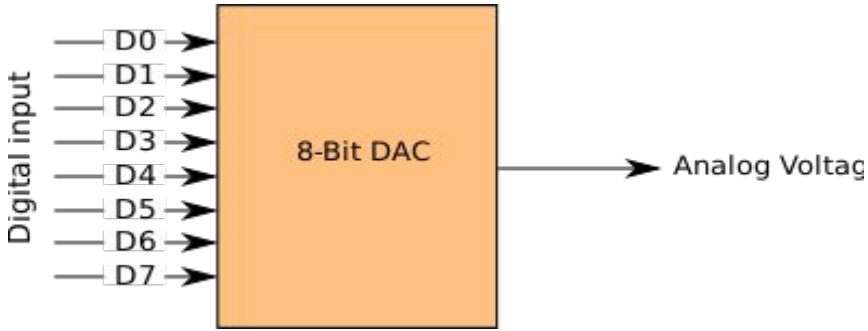
This design can easily be enhanced with active filters, etc.

Quick Introduction

To ADCs, DACs, Arduinos, etc

Digital to Analog converter (DAC)





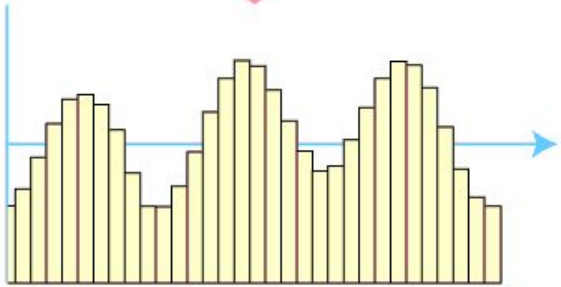
(Wikipedia: “A DAC converts an abstract finite-precision number (usually a fixed-point binary number) into a physical quantity.) In instrumentation, the output of a DAC is almost always a voltage or current.

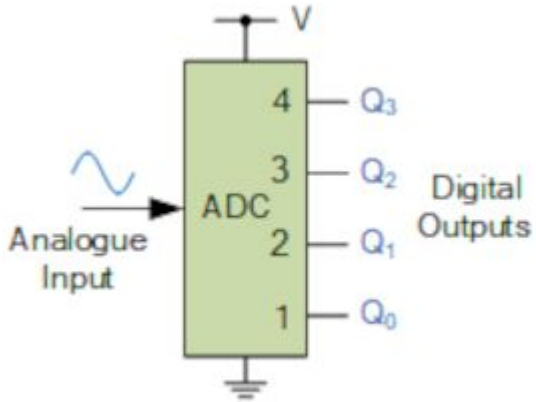
Analog(ue) to Digital converter (ADC)

fig. 1

Conversion A-D

fig. 2

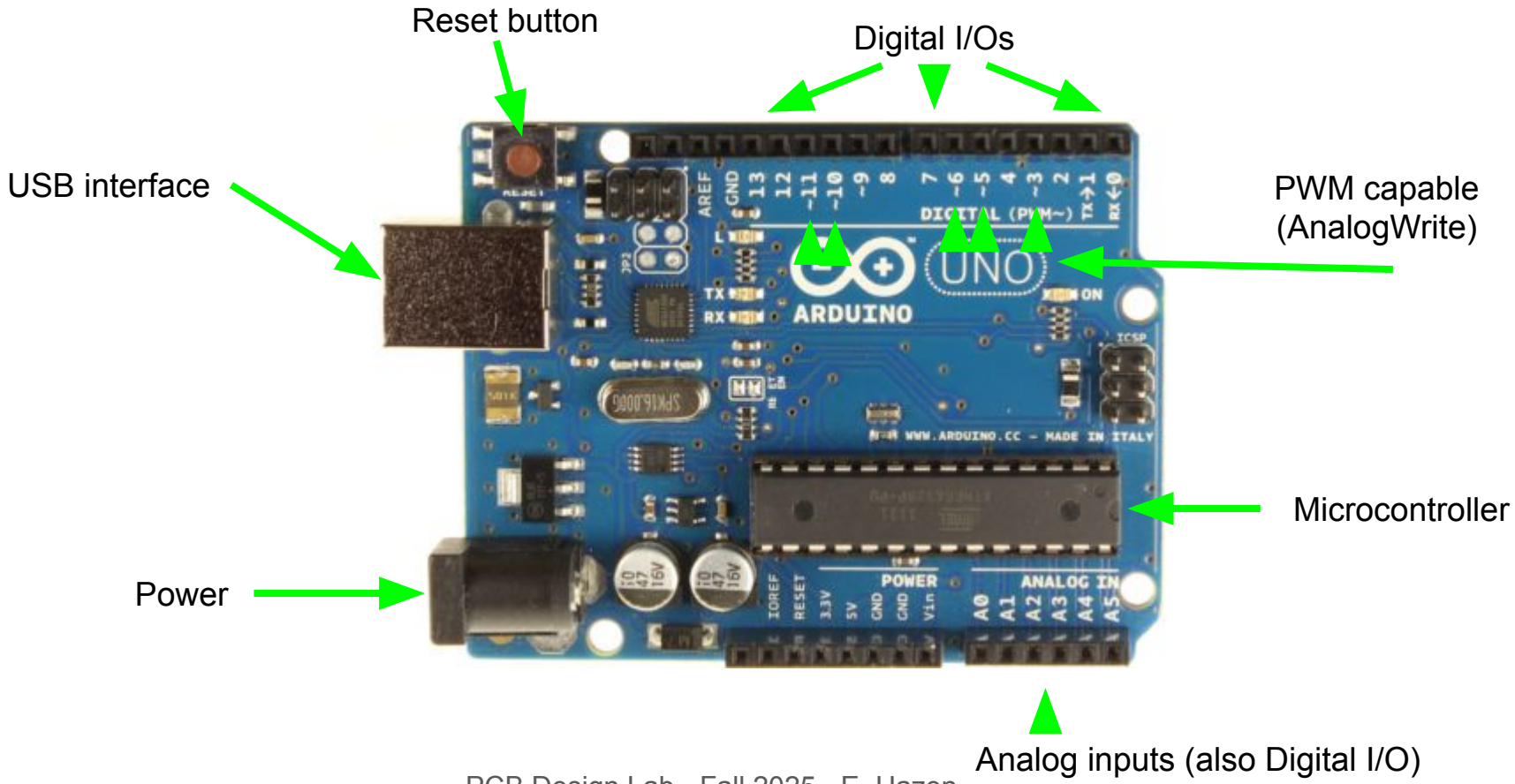




(Wikipedia: In electronics, an analog-to-digital converter (ADC, A/D, or A-to-D) is a system that converts an analog signal, such as a sound picked up by a microphone or light entering a digital camera, into a digital signal.)

Arduino Uno

Note that many types have special-functions on some pins

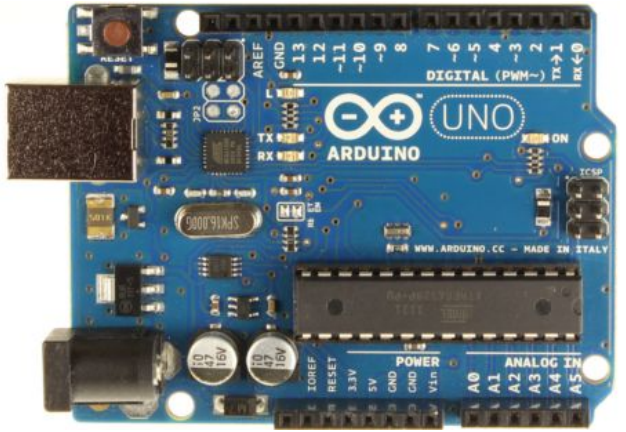


Arduino “shield” board

- Mounts on top of Arduino
- Uses power (usually) from Arduino
- connected using tall header pins

Multiple shields can be stacked (if they are compatible).

This is not a very sophisticated system but it does work for simple things and is a quick way to add electronics to a microcontroller.



What you’re going to design

Details details:

- 4 channels analog output, 12 bits, 0-5V range
  - Buffered voltage outputs
  - Up to 75kHz update rate (software/arduino dependent)
- 4 channels analog input, 12 bits, 0-5V range
  - High-impedance voltage input
  - Up to 100kHz sampling rate (software/arduino dependent)
- Screw terminal and header connectors

SPI (Serial Peripheral Interface)

SPI is a *de facto* standard for synchronous serial communication between integrated circuits.

Typically used to communicate between microcontrollers (e.g. Arduino) and peripheral devices (DACs, ADCs, displays, EEPROM etc).

The signal and device naming is in flux and somewhat confusing. I prefer the names in bold.

There is one ~~Master~~ (Main, **Controller…**) device and one or more ~~Slave~~ (Sub, **Peripheral…**) devices

SCLK / SCK	Serial clock from Main/Controller
MOSI / **COPI**	Serial data from Main to Sub Main Out Sub In Controller Out Peripheral In
MISO / **CIPO**	Serial data from Sub to Main Main In Sub Out Controller In Peripheral Out

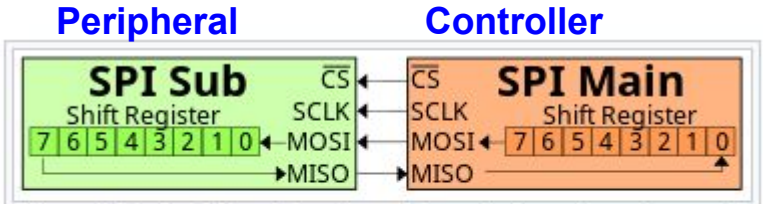
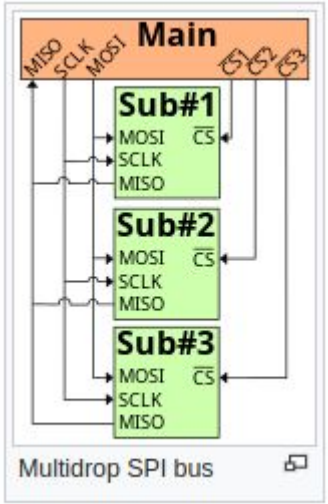


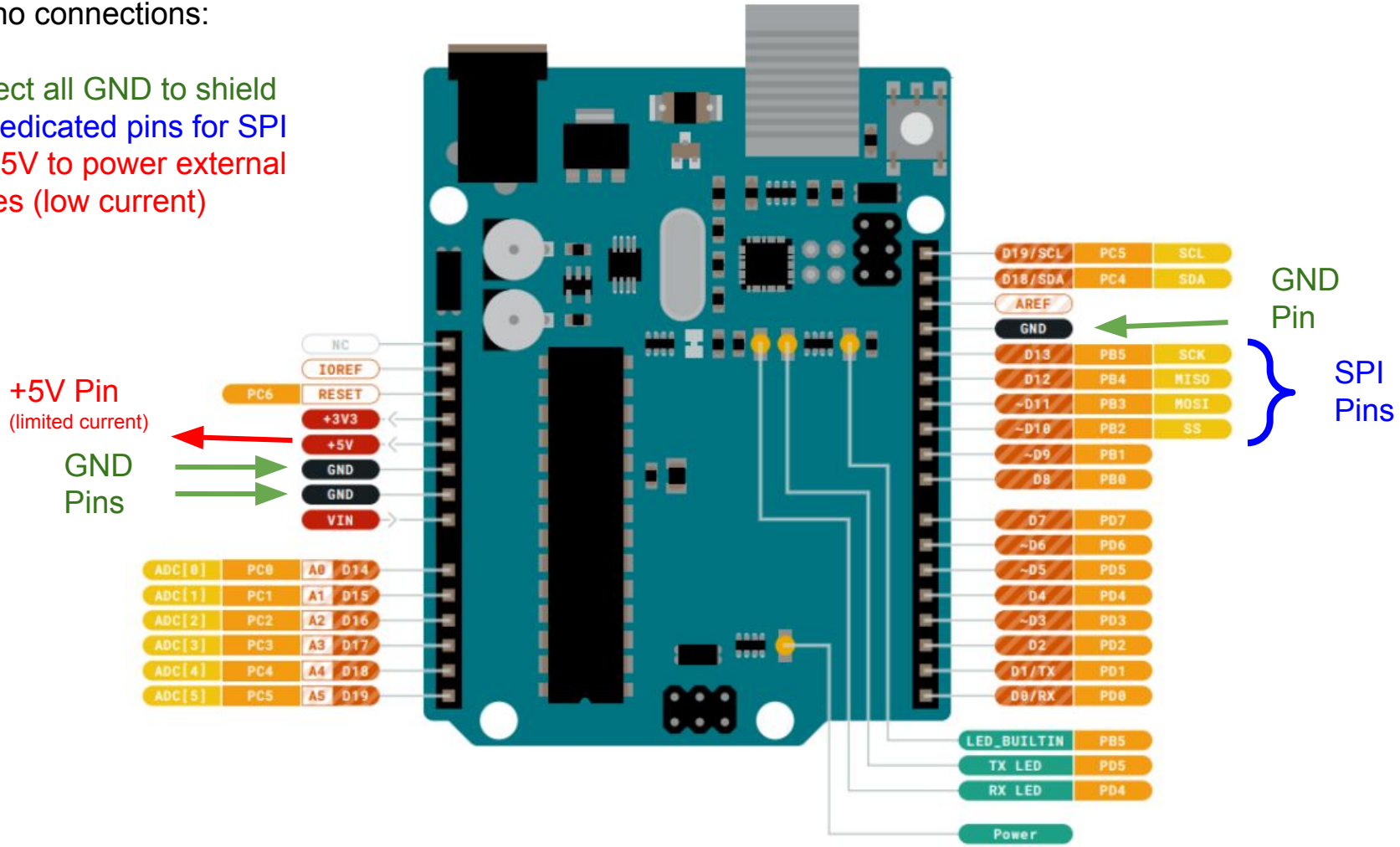
Figure 1: Basic SPI configuration using a single main and a single sub. Each device internally uses a shift register for serial communication, which together forms an inter-chip circular buffer.

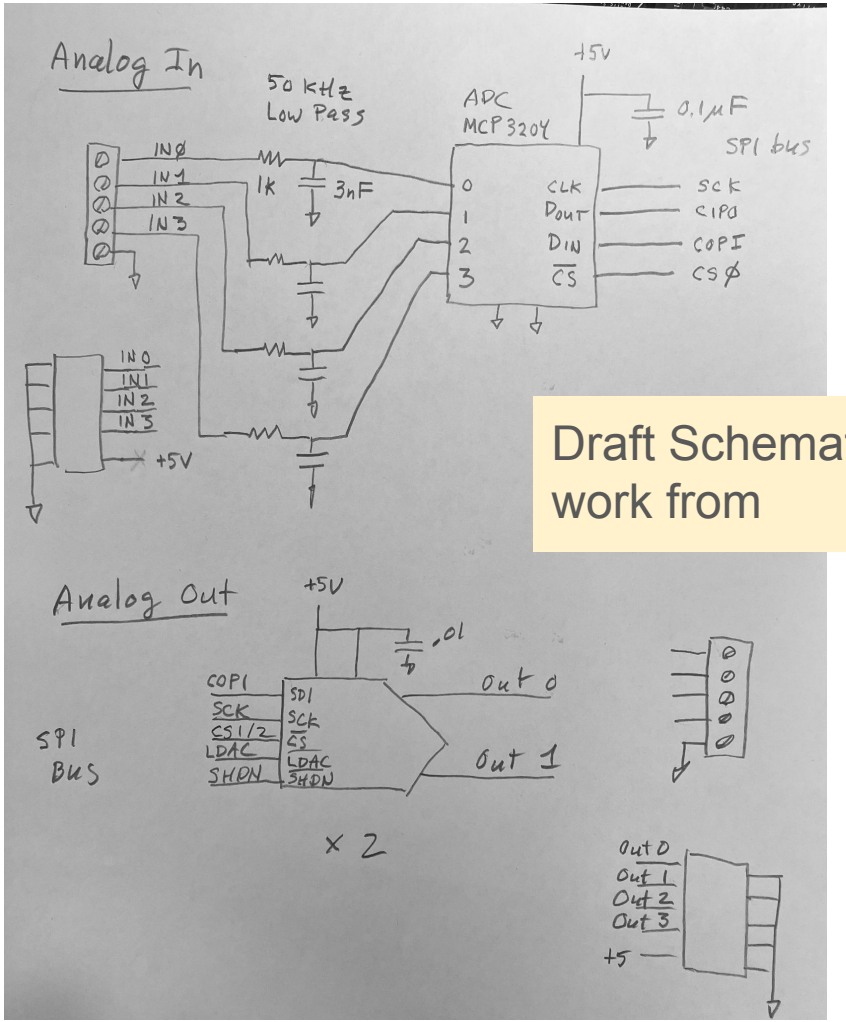


Multidrop SPI bus

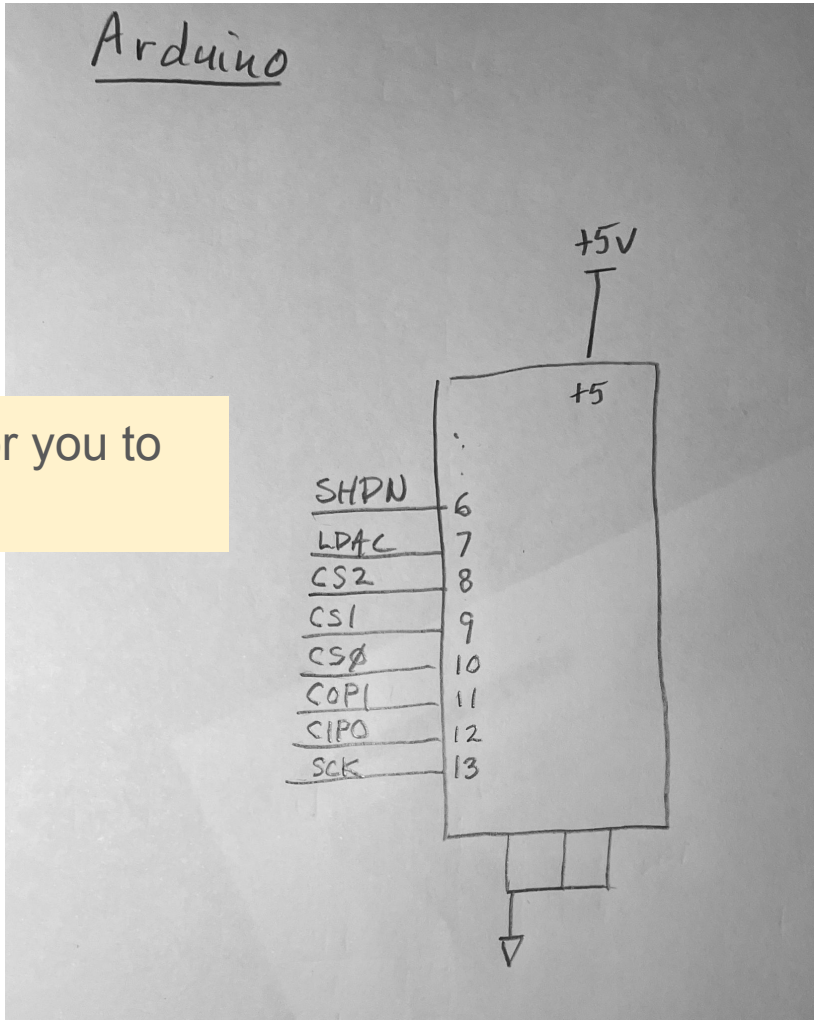
Arduino connections:

Connect all GND to shield
Use dedicated pins for SPI
Use +5V to power external devices (low current)





Draft Schematic for you to work from

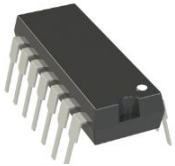


Parts

Screw terminal DigiKey ED10564-ND

A/D Converter DigiKey MCP3204-CI/P-ND

D/A Converter DigiKey MCP4922-E/P-ND

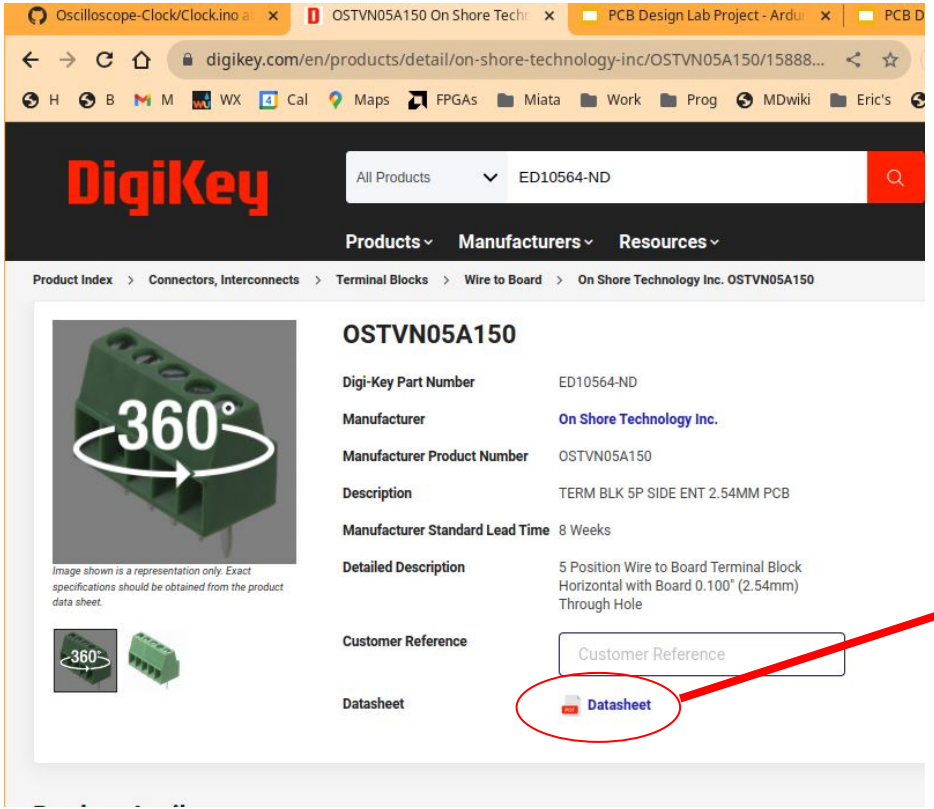


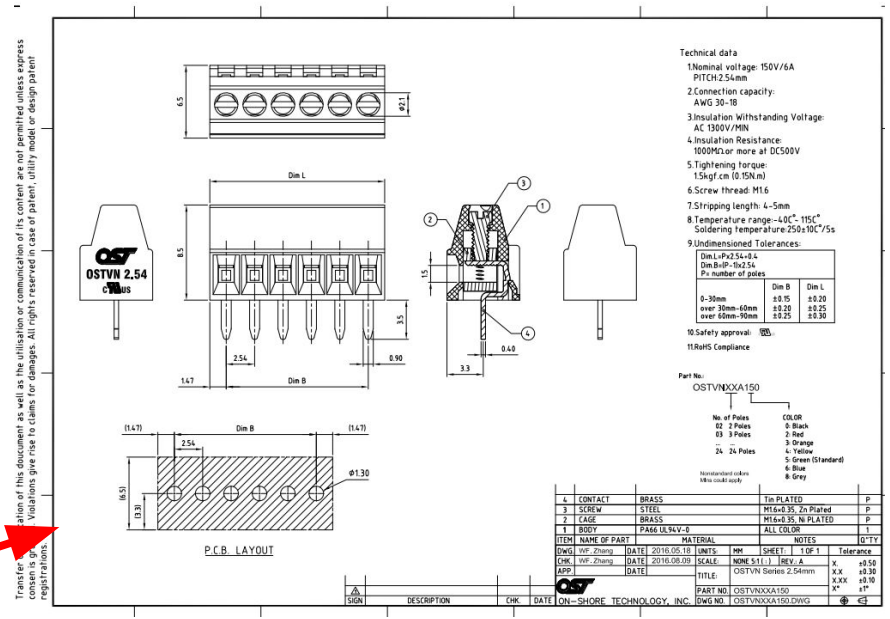
Header DigiKey 1849-FR20205VBDN-ND

I recommend the websites
of DigiKey and Mouser
for electronic parts and data

www.digikey.com

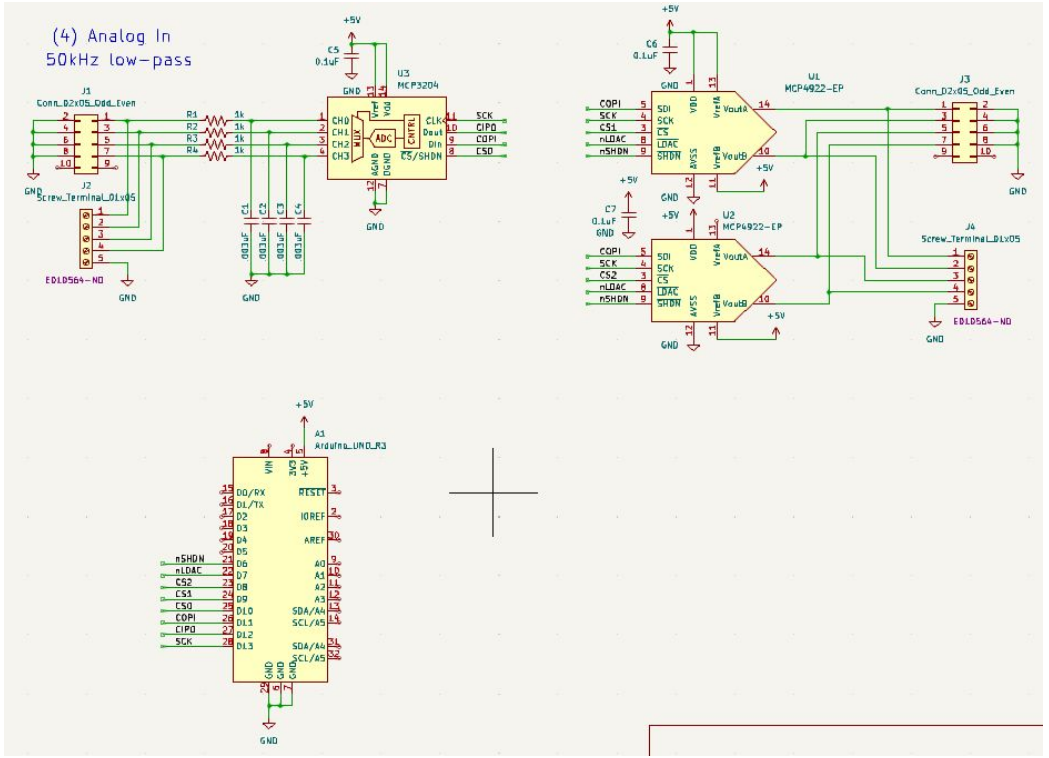
Refer to datasheet for drawings and footprint information





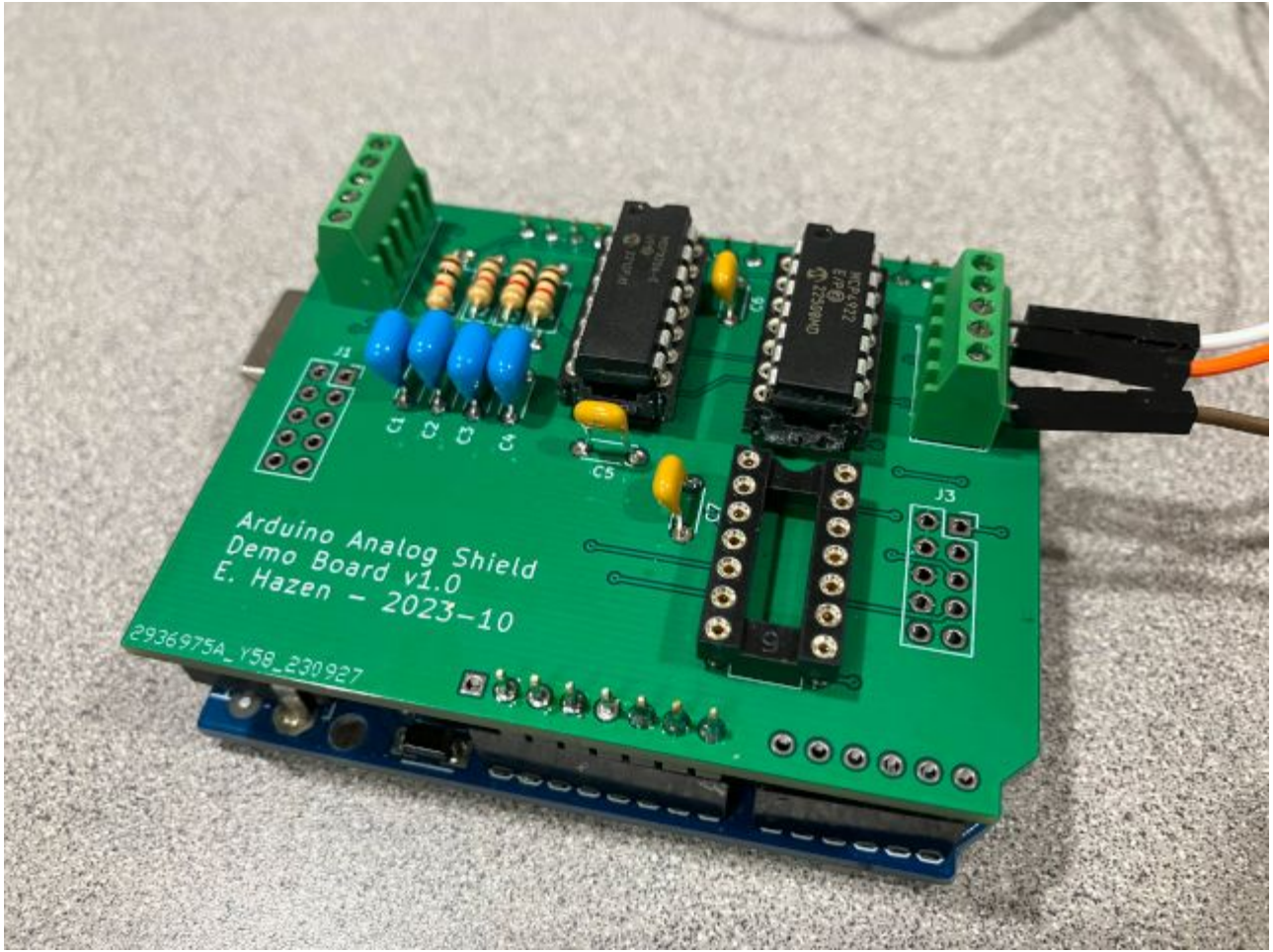
Here is my schematic.

It’s small and fuzz on purpose :)
please don’t try to copy it exactly
but make your own.



Here is the finished product…

Again, don't try to copy my layout, but it gives you an idea of one way the board can look.



Some hints

- KiCAD knows the layout of an Arduino Shield
  Use “**File→New Project from Template**” and choose the Uno shield|

- To turn in your project:
  Create a ZIP archive of the *entire project directory*
  Name it like this: **`Eric_Hazen_kicad_v1.zip`**
  Upload to the shared google drive
  If you need to update it before 10/23, please upload again
  With a new version number