SEICHE 2022
Introduction to Arduino

Instructor: Paul Frommeyer

Corporate Sponsors
- DXC Technology
- PFC Networks

YOUR INSTRUCTOR

Career

- Computer Networking Architect (30+ yrs)
  - – Apple Computer – Sony – General Magic
  - – Cisco Systems – HP/HPE/DXC [current]
- Also Linux/Unix Sysadmin
- Also C developer (entry level)
- BS Computer Science, BS Philosophy

Personal

- Ubergeek-at-large
- Computers, electronics, woodworking
  - Electronics since 6th grade
  - Computers sicne 7th grade

“Lesson” 1 – Intro and Setup

Part A

- Introduction to class format
- Overview of lesson plan
- Presentation format (monitor, camera, screen, whiteboard)
- Inventory of Arduino kits

Part B

- Review of Laptop GUI operation
- Initial review of electricity basics

Class Format
"Minimum needed to do cool stuff"

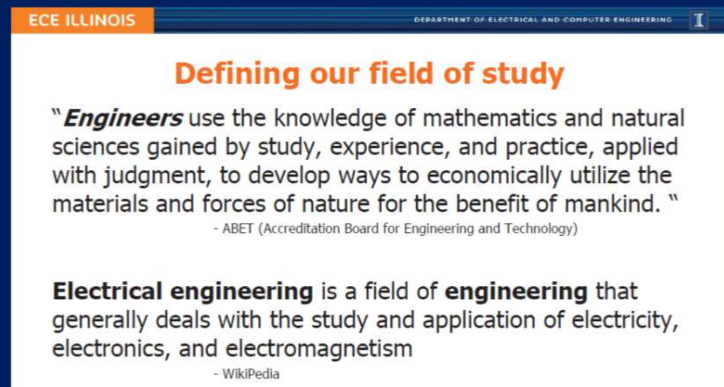
Goals

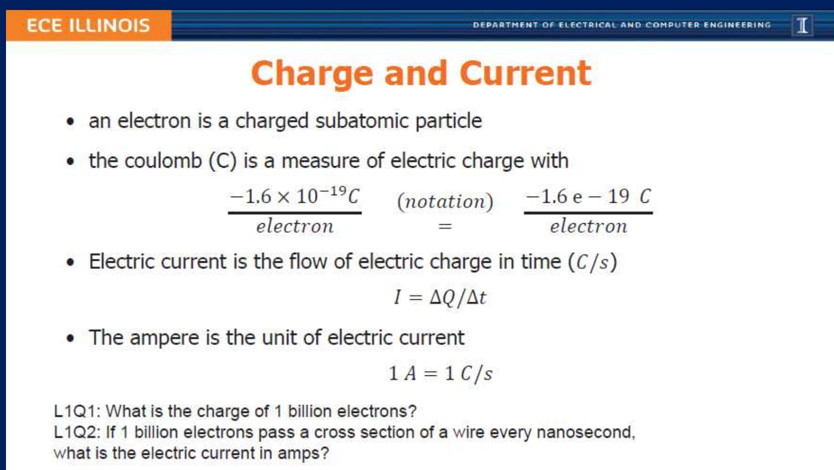
- Software installation
- Load software onto microcontroller
- Arduino software basics
- DC electricity basics
- Digital Multimeter (DMM) usage
- Electronic component basics
- Powering microcontrollers
- Connecting and using components and modules
- TL; DR – Use a microcontroller and connect things to it without blowing anything up

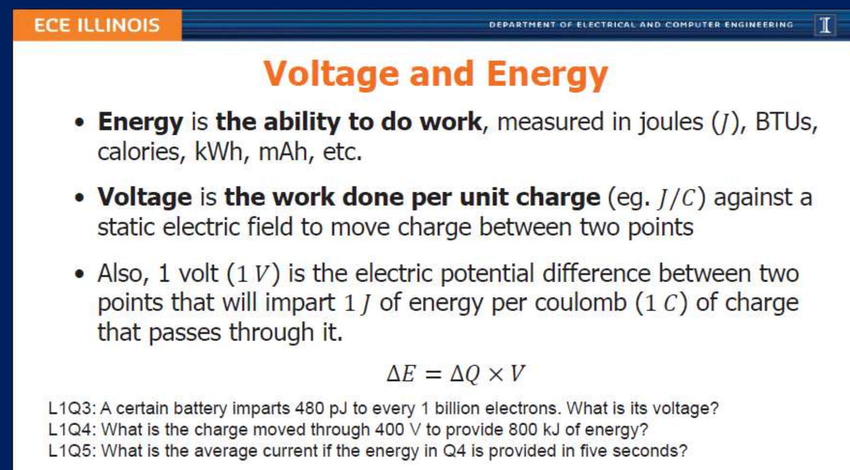
To Be Avoided

- Lots of technical details (subsequent classes)
- Extensive math, computer science, or EE

Example and Rant: What We Won't Do







Traditional Approach

- Focus on irrelevant mathematics and other high level theory
- Obsessive memorization of information that can be easily and readily looked up
- Assumption that full and complete knowledge is required before something can be built

The Maker and Hacker Approach

The maker culture is a contemporary subculture representing a technology-based extension of DIY culture that intersects with hardware-oriented parts of hacker culture and revels in the creation of new devices as well as tinkering with existing ones. The maker culture in general supports open-source hardware. Typical interests enjoyed by the maker culture include engineering-oriented pursuits such as electronics, robotics, 3-D printing, and the use of computer numeric control tools, as well as more traditional activities such as metalworking, woodworking, and, mainly, its predecessor, traditional arts and crafts.

The subculture stresses a cut-and-paste approach to standardized hobbyist technologies, and encourages cookbook re-use of designs published on websites and maker-oriented publications.[1][2] There is a strong focus on using and learning practical skills and applying them to reference designs.

– Wikipedia

The hacker culture is a subculture of individuals who enjoy the intellectual challenge of creatively overcoming the limitations of software systems to achieve novel and clever outcomes.[1] The act of engaging in activities (such as programming or other media[2]) in a spirit of playfulness and exploration is termed hacking. However, the defining characteristic of a hacker is not the activities performed themselves (e.g. programming), but how it is done[3] and whether it is exciting and meaningful.[2]

…

Richard Stallman explains about hackers who program:

> ***What they had in common was mainly love of excellence and programming. They wanted to make their programs that they used be as good as they could. They also wanted to make them do neat things. They wanted to be able to do something in a more exciting way than anyone believed possible and show "Look how wonderful this is. I bet you didn't believe this could be done."[7]***

Hackers from this subculture tend to emphatically differentiate themselves from what they pejoratively call "crackers"; those who are generally referred to by media and members of the general public using the term "hacker", and whose primary focus—be it to malign or for malevolent purposes—lies in exploiting weaknesses in computer security.[8]

– Wikipedia

Lesson Plan Overview

Lesson 1 – Intro and Setup
[may require 2 classes]
- Introduction to class format
- Overview of lesson plan
- Presentation format (monitor, camera, screen, whiteboard)
- Inventory of Arduino kits
- Inventory of USB drives
- Installation of Arduino software
- Installation of ESP32 serial port drivers (Windows only)

Lesson 2 – Laptop operation review
- Control panel/Settings location
- Home directories and folder hierarchy
- Arduino file locations
- Search functions
- Open questions and issues

Lesson 3 – IDE essentials
- Setting board type
- Loading example sketches
- Finding and selecting the Arduino USB port
- Uploading a sketch to the microcontroller
- Basic Arduino sketch (program) structure

Lesson 4 – Computer Data Formats
- Binary math
- Basic data types

Lesson 5 – DC Electricity Basics and DMM Intro
- Quick review: DC vs AC electric currents
- DC current operation
- Voltage vs Amperage (current)
- Introduction to basic DMM functions and usage
- Series vs parallel circuits
- Ohm’s law

Lesson 6 – Elementary Components
- Resistors
- Diodes and LED’s
- Capacitors: Electrolytic (polarized) vs non-electrolytic (non-polarized)
- Simplified transistor operation

Lesson 7 – ESP32 External power sources [may require 2 classes]
- Microcontroller board power requirements
- Board power input options
- Components with different voltage requirements
- External power circuit design

Lesson 8 – Connecting LED’s to the ESP32 controller board
- Microcontroller pinouts and diagrams
- Different types of pins
- Analog input
- Digital Input
- Digital output
- LEDs, constant current, and dropping resistors

Lesson 9 – Connecting smart LED strips
- Simplified serial communication protocols
- One-wire serial communication protocol
- Hardware vs software serial
- WS2812B and Neopixel connections
- External device power design

Lesson 10 – Connecting Sensors
- I2C serial protocol
- I2C connections
- I2C sensor board power design
- I2C sensor external power design

Lesson 11 – Connecting LCD and Matrix Displays
- The SPI serial protocol
- SPI connections
- SPI device external power design

Lesson 12 – Connecting other sensors [time permitting]
- Connecting buttons
- Connecting potentiometers
- Reading buttons
- Reading potentiometers

Arduino Kit Inventory

- **One (1) Flambeau blue tackle and project kit box with six (6) moveable plastic dividers**
  - Two dividers installed in upper tray
  - Remaining dividers placed on bottom of kit box or in lid storage area
- **ESP32 Microcontroller – Hel-Tec Wifi Kit 32** : installed at edge of breadboard
- **“Upgraded” breadboard kit**
  - Large rectangular plastic case : placed in bottom rear center of kit box
  - DuPont Male-Male wires : placed in upper tray front
  - DuPont Male-Female wires : placed in upper tray front
  - DuPont Female-Female wires : placed in upper tray front
  - Ceramic capacitors, electrolytic capacitors, diodes, transistors, photocells, RGB LED, low profile buttons, active buzzer : sorted and stored in blue clamshell case
  - 4N35 IC, 74HC595 IC, passive buzzer, one (1) low profile button : installed on breadboard
  - 7-13VDC breadboard power adapter : placed in middle tray right rear
  - Resistor packs : placed in upper tray left rear
  - LEDs : left in bag, stored in large rectangular plastic case
  - Set of breadboard jumper wires : left in small plastic case, stored in large rectangular plastic case
  - Two sets of long header pins : stored in large rectangular plastic case
  - Resistor color code reference key : stored in large rectangular plastic case
  - One (1) 830 series breadboard : placed in upper tray right rear
- **Two (2) 1000uf electrolytic capacitors : stored in large rectangular plastic case**
- **Four (4) MAX7219 LED matrix displays on common carrier board : placed in upper tray right rear**
- **9VDC battery clip to 5.5/2.1mm plug : placed in middle tray center right**
- **Rotary encoder board and knob : Knob attached and assembly placed in middle tray left front**
- **10K ohm linear B rotary potentiometer and knob : Knob attached and assembly placed in middle tray left front**
- **120VAC to 5VDC 2000mA (2A) power supply adapter : Placed in bottom right of kit box**
- **One (1) 2-wire screw terminal to 5.5/2.1mm female jack adapter : Attached to power adapter plug**
- **One (1) WS2812B “Neopixel” programmable LED strip, 1 meter : placed in middle tray center**
- **One (1) Four-in-one multibit screwdriver : placed in upper tray left front**
- **One (1) Digital Multimeter (DMM) : placed in bottom left of kit box**
- **One (1) set of standard test leads for DMM : placed in middle tray left**
- **One (1) 9 volt DC battery : installed in DMM**
- **One (1) dual red/black banana jack to dual red/black alligator clip lead adapter : bottom left of kit box**
- **One (1) straight mini pliers with integral wire cutter : placed bottom center front of kit box**
- **Four (4) DuPont male to mini-grabber clip test leads, 1 red, 1 black, 2 multicolor: middle tray center**
- **One (1) USB-A male to micro-USB male adapter cable : placed in upper tray**

Our Microcontroller
Hel-tec WiFi Kit 32

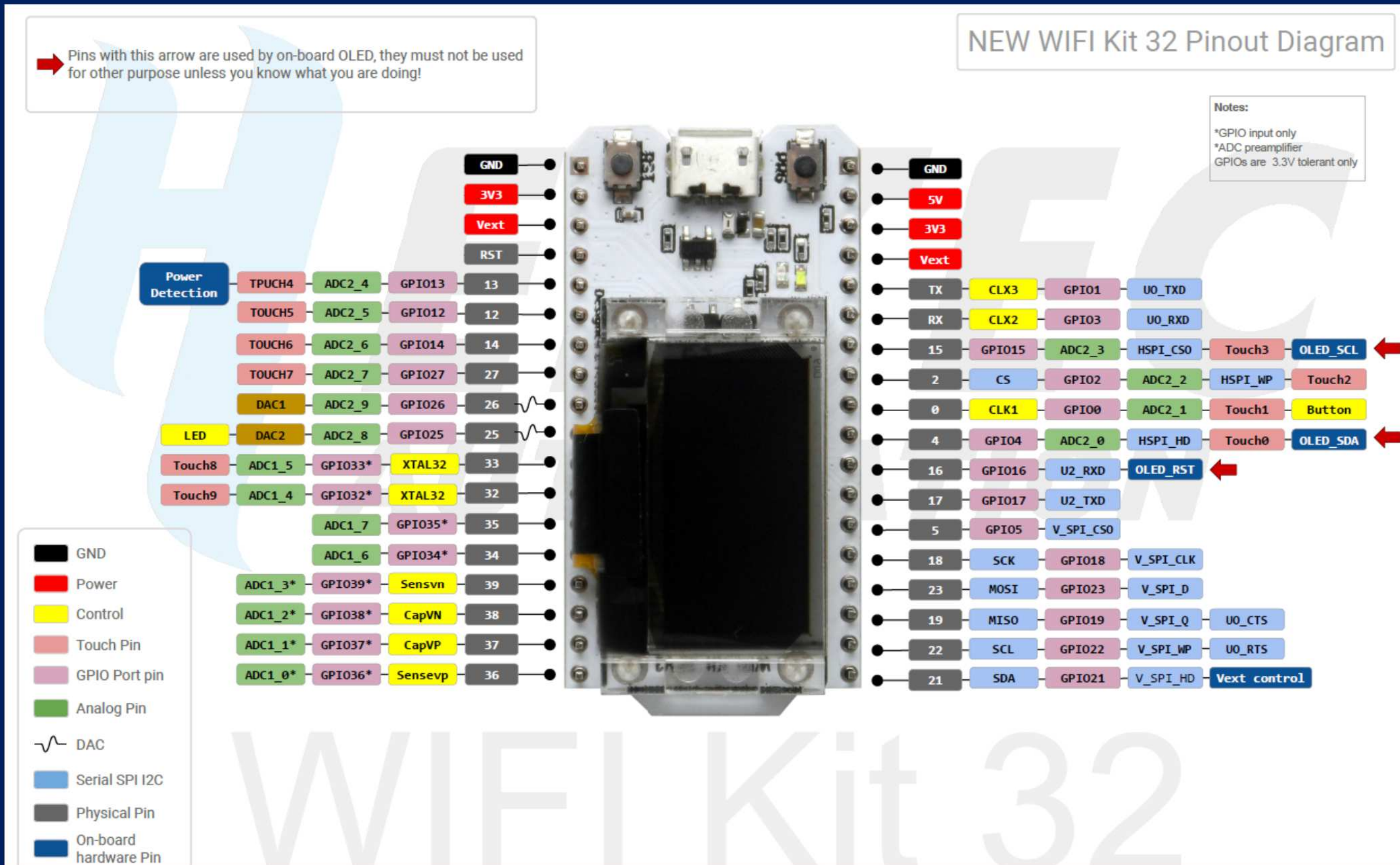
Quick Specs

- ESP32 – 32 bit processor
- Arduino IDE compatible
- Built in monochrome OLED display
- 3.3V DC power
- Integral touch sensor capability
- Integral battery charging circuitry
- Integral WiFi and Bluetooth capability

Advantages

- Built in OLED reduces display learning curve
- ESP32 is powerful with lots of room for software code
- Built-in WiFi allows for web controlled software

Our Microcontroller



Hel-Tec WiFi Kit 32

Formal End of Lesson 1

In next week's exciting episode

- Distribution of laptops
- Distribution and inventory of USB drives
- Installation of Arduino and Fritzing software
- Installation of ESP32 serial port drivers (Windows only)
- Electricity basics review (time permitting)