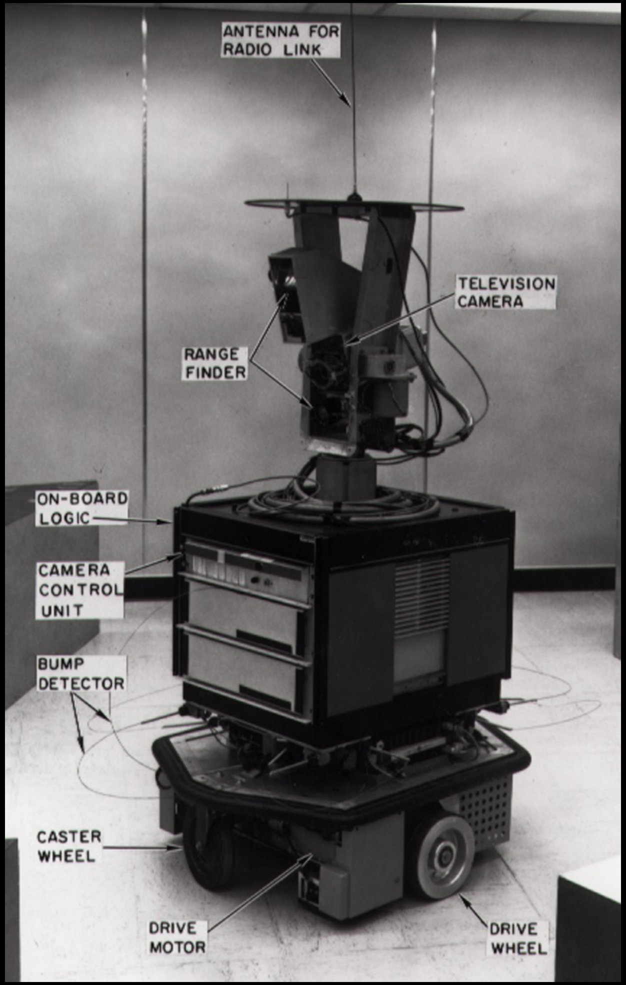
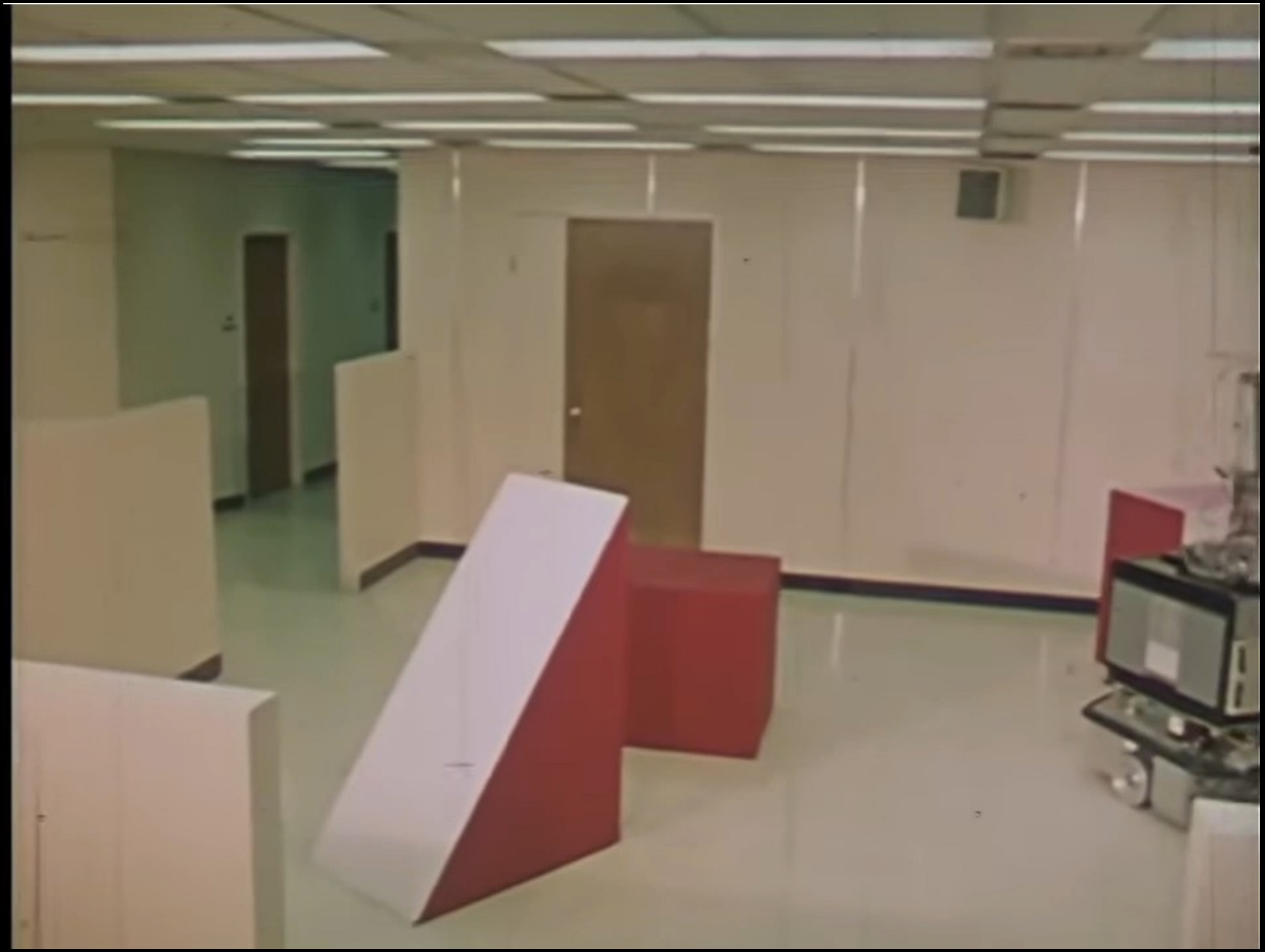
ECE 4160/5160
MAE 4910/5910

Dr Jonathan Jaramillo
jdj78@cornell.edu

# Fast Robots
# Sensors & Bluetooth

# History

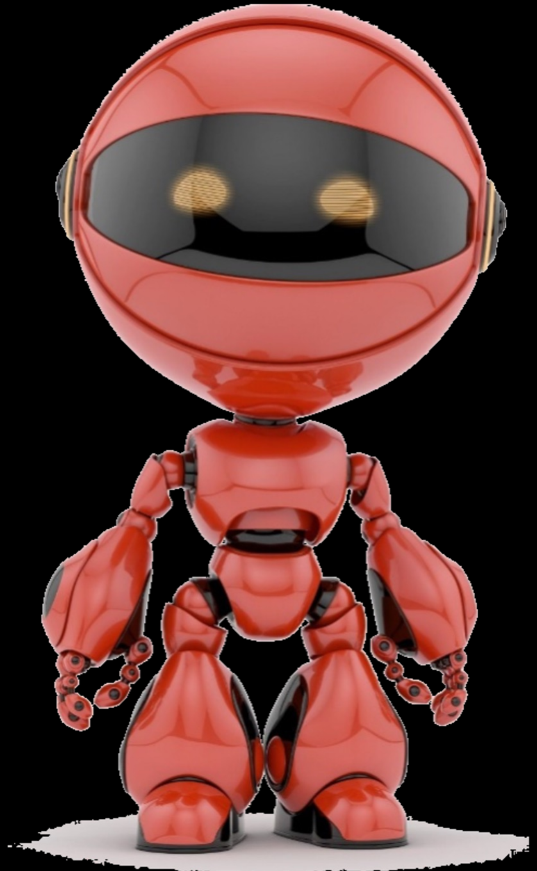




Shakey: Experiments in Robot Planning and Learning (1972), SRI

# Sensor Classification

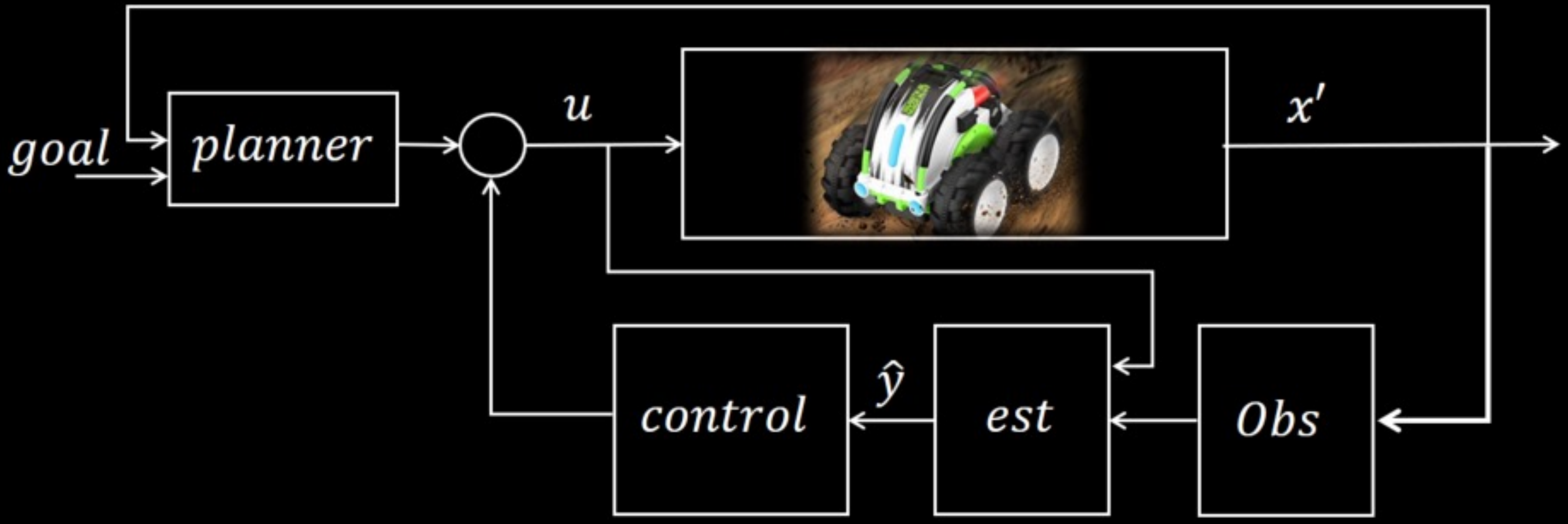
- **Proprioceptive**
  - Motor speed, wheel load, joint angles, battery voltage
- **Exteroceptive**
  - distance measurements, light intensity, sound amplitude
- **Passive Sensors**
  - Measure ambient environmental energy
  - E.g. temperature probes, microphones, light sensors
- **Active Sensors**
  - Senses reaction to emitted energy
  - E.g. wheel quadrature encoders, ultrasonic sensors, laser rangefinders



# Classification

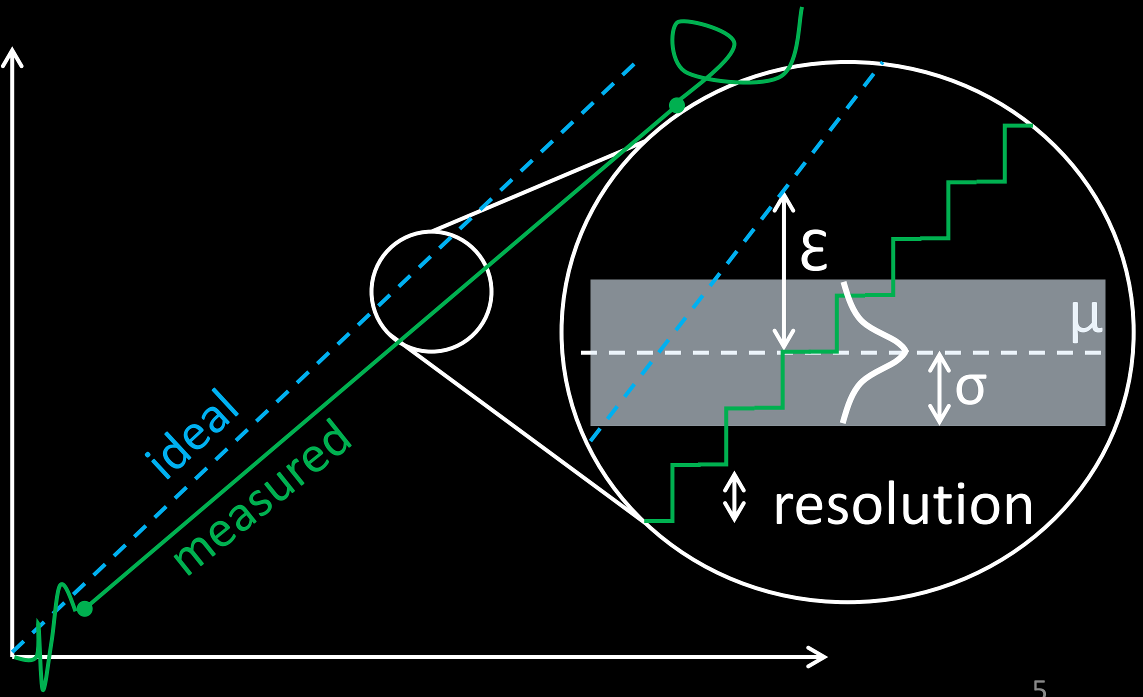
| Type | Sensor | Prop/Exte | Passive/Active |
|---|---|---|---|
| Tactile (contact/closeness) | Contact switches, bumpers,<br>Break beams, proximity<br>Capacitive | Exteroceptive<br>Exteroceptive<br>Exteroceptive | Passive<br>Active<br>Both |
| Wheel/motor | Brush encoders<br>Potentiometers<br>Optical encoders<br>Magnetic/inductive/capacitive encoders | Proprioceptive<br>Proprioceptive<br>Proprioceptive<br>Proprioceptive | Passive<br>Passive<br>Active<br>Active |
| Active ranging | Reflectivitiy sensors, ultrasonic, laser rangefinders, optical triangulation, etc. | Exteroceptive | Active |
| Heading | Compass<br>Gyroscopes | Exteroceptive<br>Proprioceptive | Passive<br>Passive |
| Ground based beacons | GPS, RF, reflective beacons | Exteroceptive | Active |
| Motion/speed | Doppler radar, sound | Exteroceptive | Active |
| Vision | CCD/CMOS | Exteroceptive | Passive |

# Sensor Characteristics



## *Name some examples*

- Dynamic Range
- Range
- Resolution
- Linearity
- Bandwidth / Sampling Frequency
- Sensitivity
- Cross-sensitivity
- Accuracy
- Precision
- Error
  - Systematic
  - Random
    - Power consumption
      - Size, price, etc…



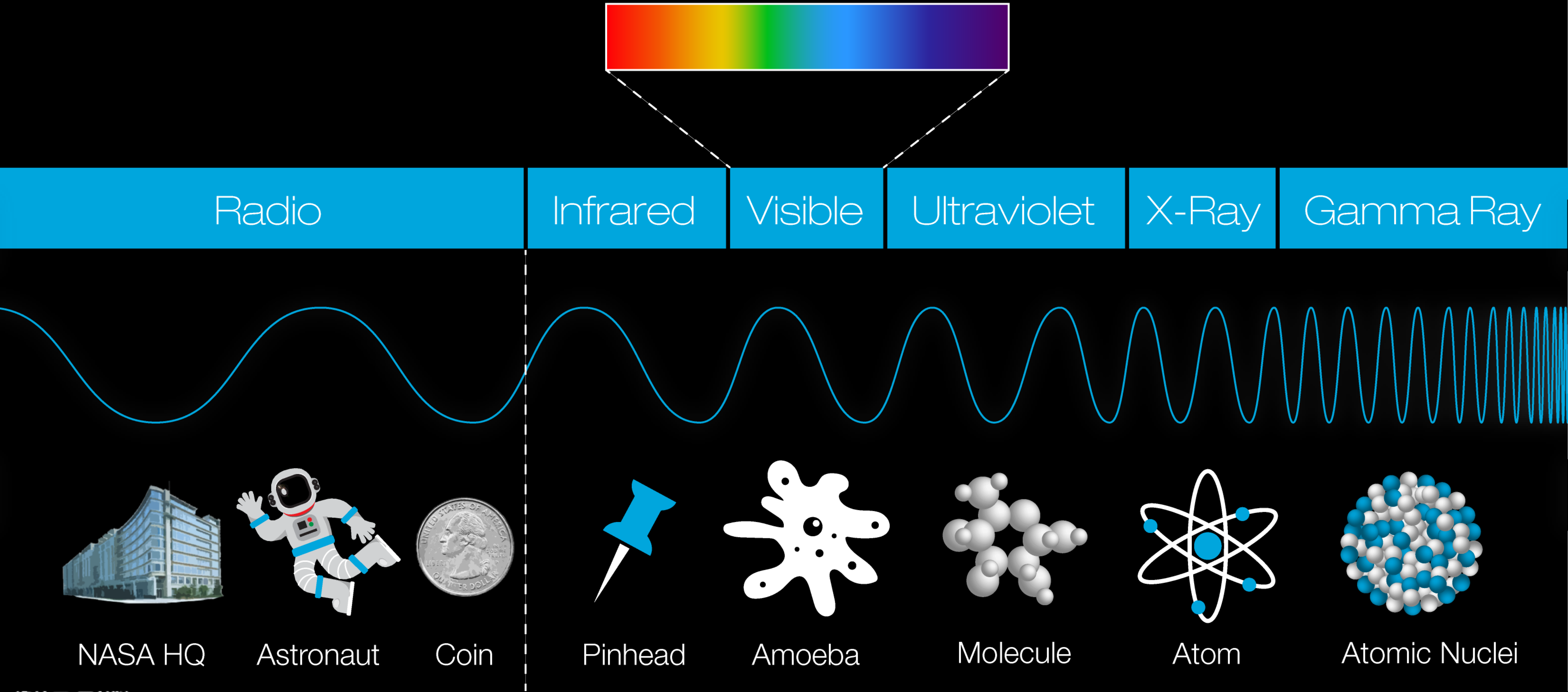
Dr Jonathan Jaramillo
jdj78@cornell.edu

# Distance Sensors

# DIY-level Distance Sensors

| Technology | Application | Pros | Cons |
|---|---|---|---|
| Amplitude-based IR | <10cm | • ~ 0.5 USD<br>• Small form factor | • Depends on target reflectivity<br>• Does not work in high ambient light |
| IR triangulation | <1m | • Insensitive to surface color/texture/ambient light | • ~ 10 USD<br>• Does not work in high ambient light<br>• Bulky (1.75” × 0.75” × 0.53”)<br>• Low sample rate (26Hz) |
| IR Time of Flight | 0.1 - 4m | • High sample rate (4kHz)<br>• Small form factor<br>• Insensitive to surface color/texture/ambient light | • ~ 6.5 USD<br>• Complicated processing<br>• Low sampling frequency: 7-30Hz |
| Ultrasonic | 0.2 – 10m | • Low cost<br>• Insensitive to ambient light and surface color<br>• Works in rain and fog | • ~4 USD<br>• Complicated processing<br>• Resolution trade off with max range<br>• Output depends on surface/geometry/humidity<br>• Bulky, sample time (tens of milliseconds)<br>• Hard to achieve a narrow FoV |

# The Electromagnetic Spectrum



Radio

Infrared

Visible

Ultraviolet

X-Ray

Gamma Ray

NASA HQ

Astronaut

Coin

Pinhead

Amoeba

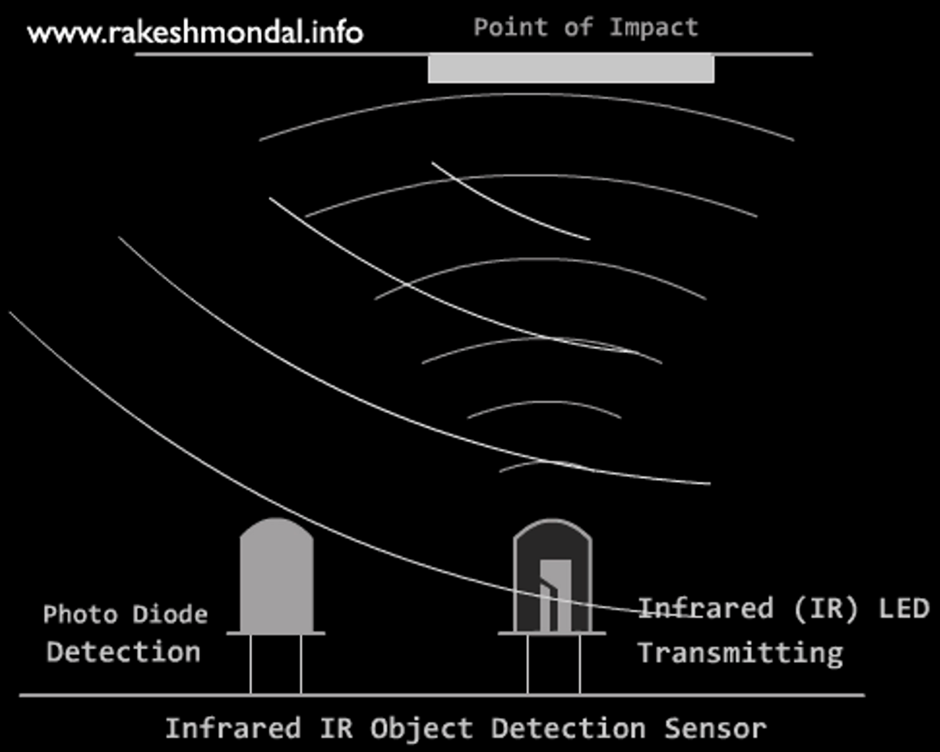
Molecule

Atom

Atomic Nuclei

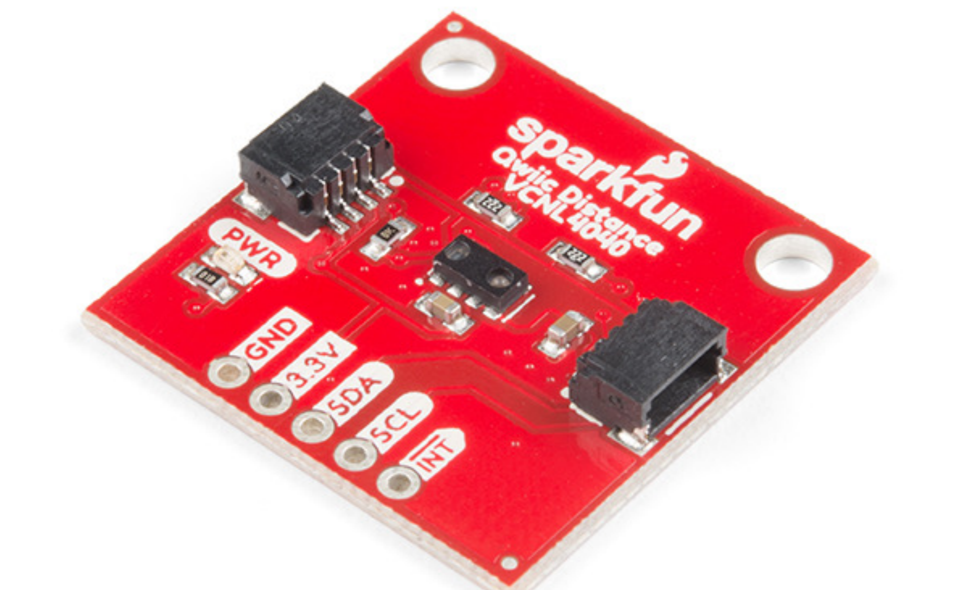
# Amplitude –Based IR Distance Sensors

- Very cheap
- Very simple circuitry
- Works reasonably well for
  - Object detection
  - Break beam sensors
  - Classifying greyscale intensity at a fixed distance
  - Short-range distance sensor
- Range <0.5m
- Sensitive to surface color, texture, and ambient light


www.rakeshmondal.info
Point of Impact
Photo Diode Detection
Infrared (IR) LED Transmitting
Infrared IR Object Detection Sensor

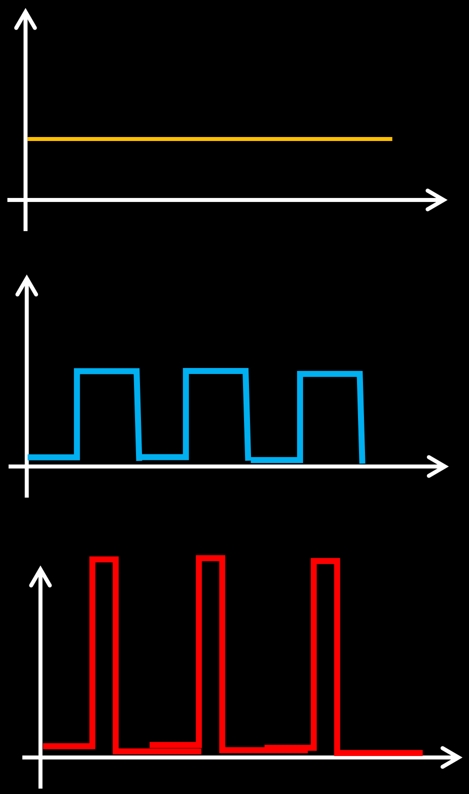
**VCNL4040**

- $3.34
- Range 20cm
- Ambient light sensor
- Programmable DC



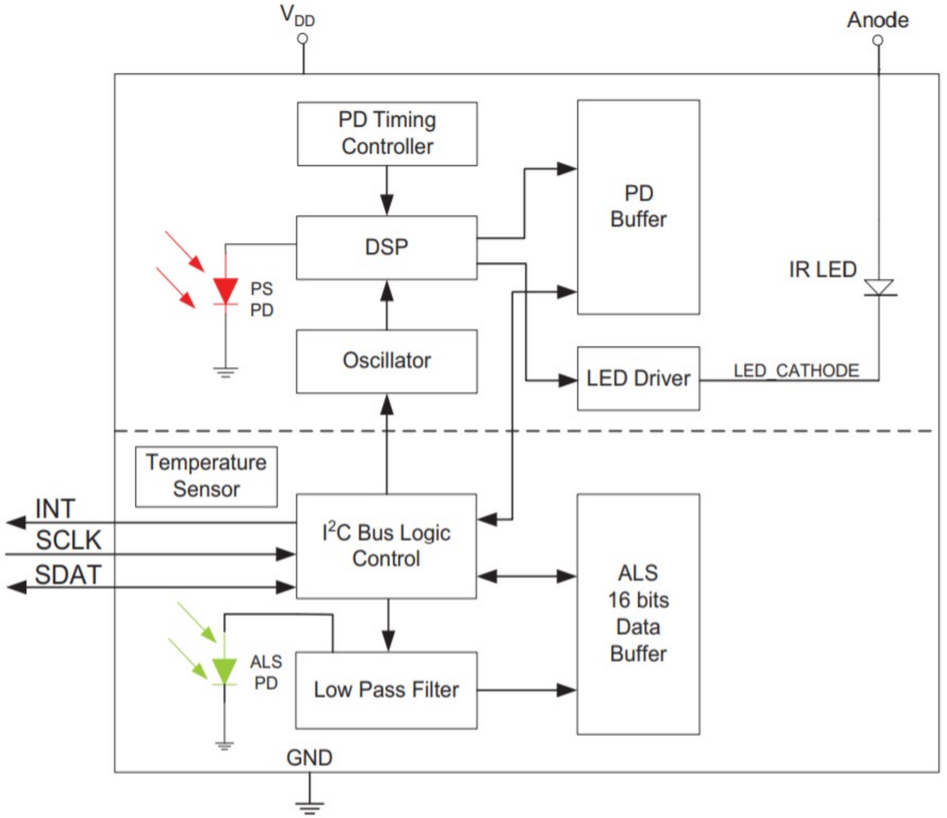
# Amplitude –Based IR Distance Sensors

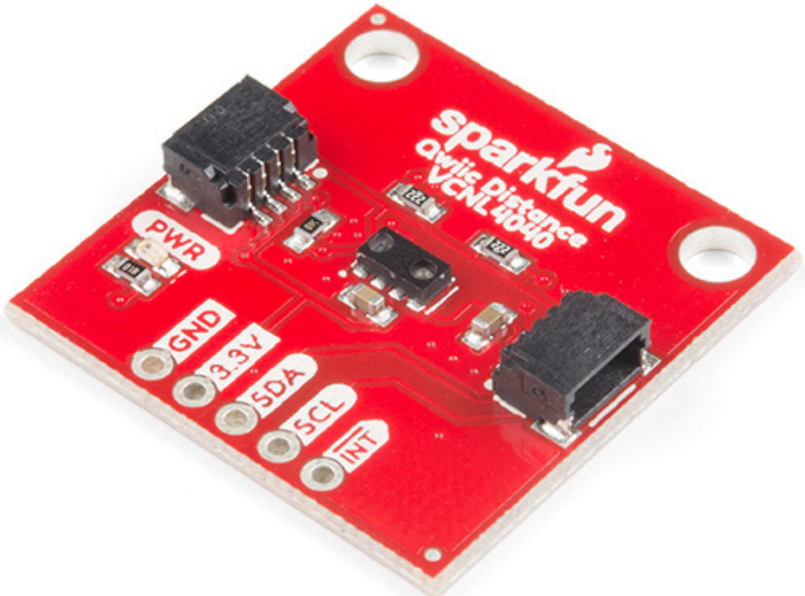
- Very cheap
- Very simple circuitry
- Works reasonably well for
  - Object detection
  - Break beam sensors
  - Classifying greyscale intensity at a fixed distance
  - Short-range distance sensor
- Sensitive to surface color, texture, and ambient light



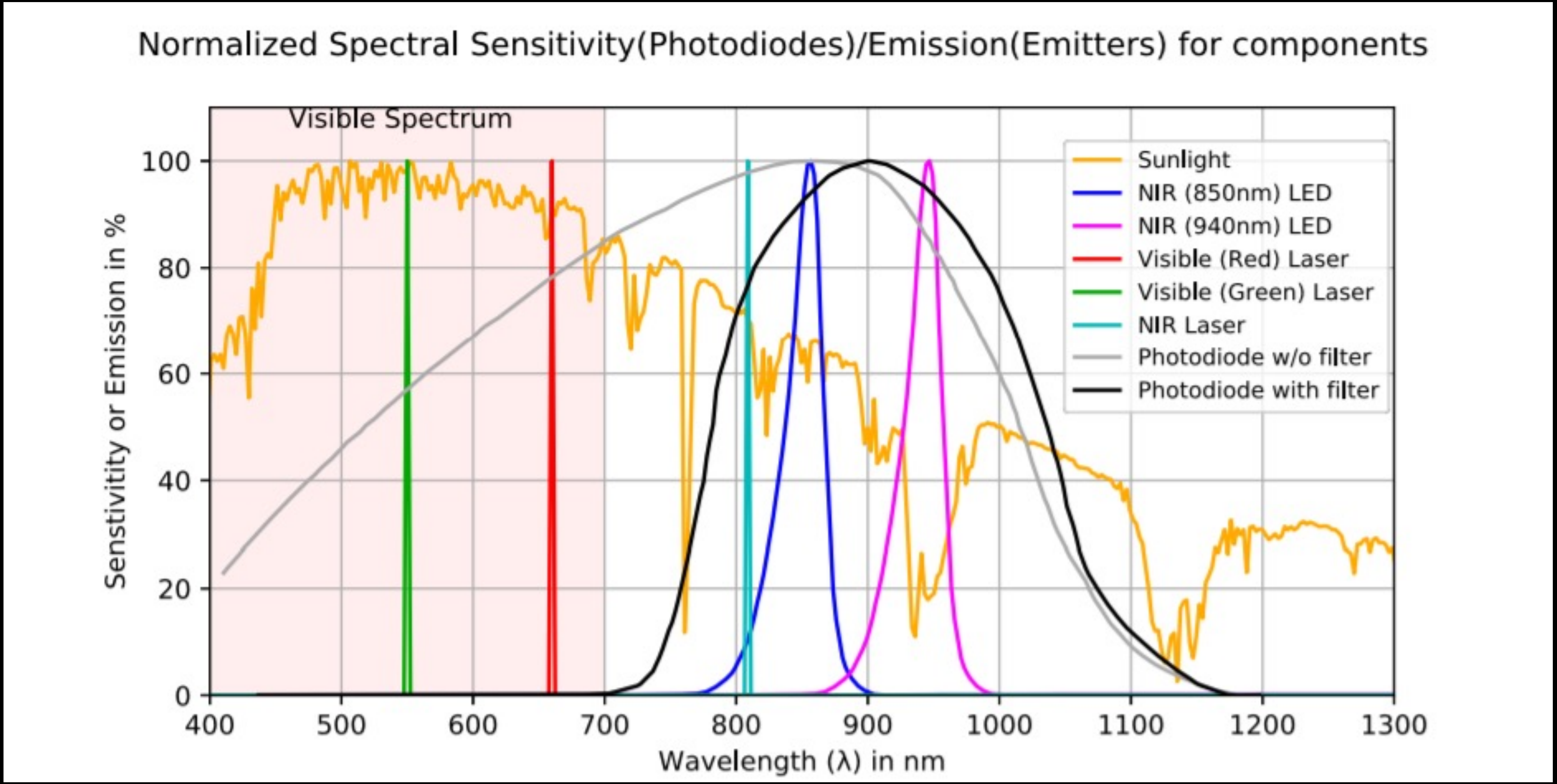
**VCNL4040**

- $3.34
- Range 20cm
- Ambient light sensor
- Programmable DC



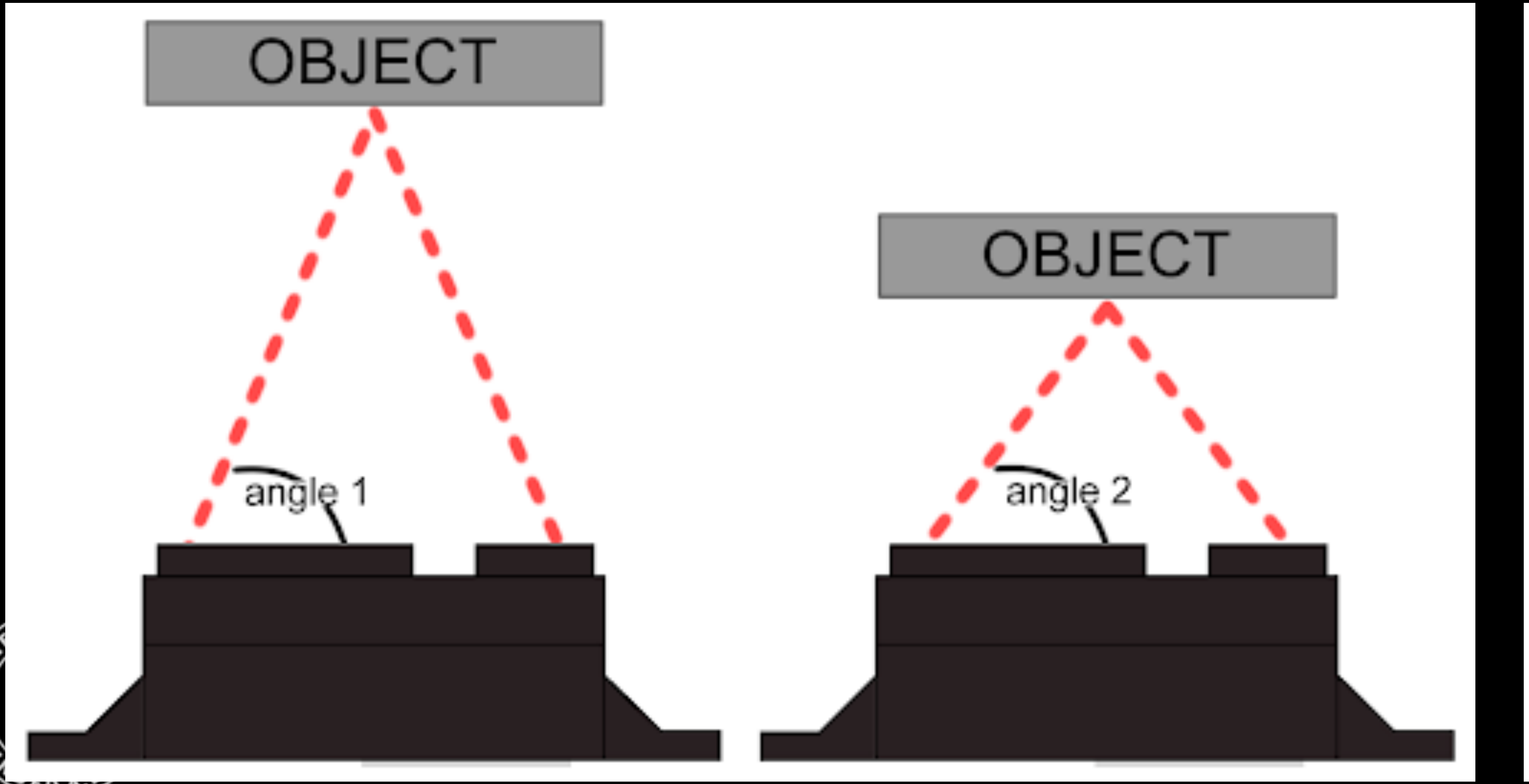


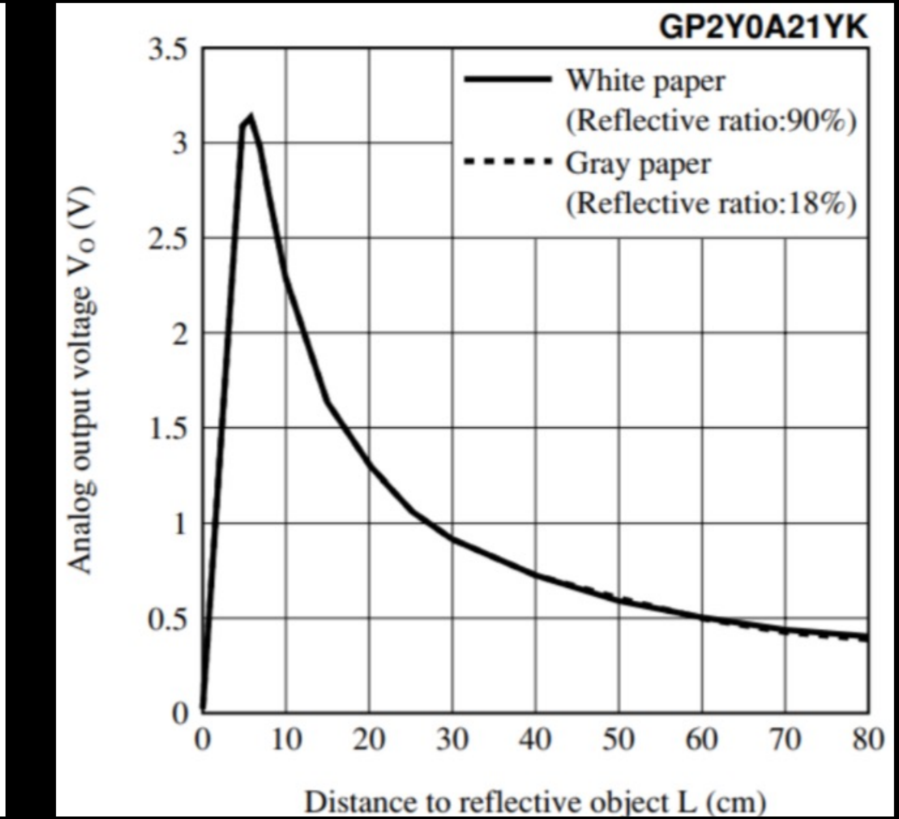
# Amplitude –Based IR Distance Sensors



# Triangulation–Based IR Distance Sensors

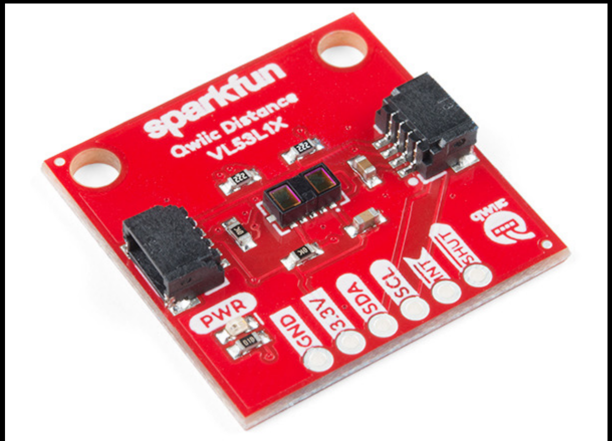
- Very simple circuitry
- Less sensitive to color, texture, ambient light
- Medium range (0.05 - 1 m)
- Cost 5-25 USD

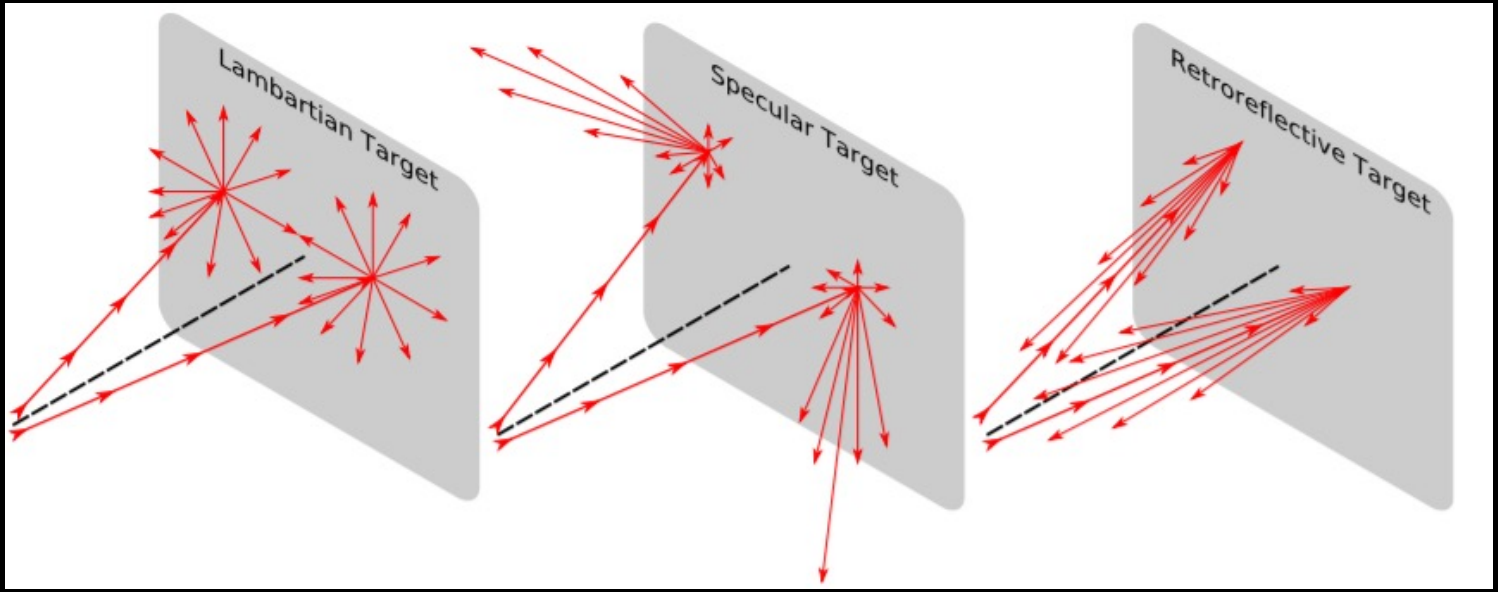


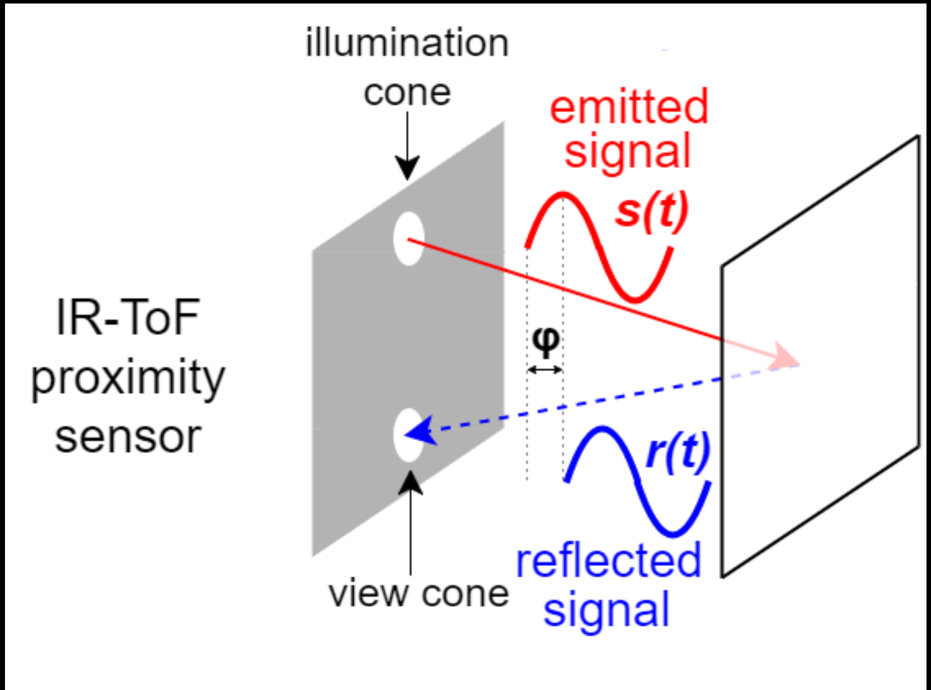


# Time of Flight IR Sensor

- Emit a pulse modulated signal, record time *t* until return!
  - r = t*c/2
  - c = speed of light = 299,792,458 m/s
- Mostly insensitive to texture, color, ambient light

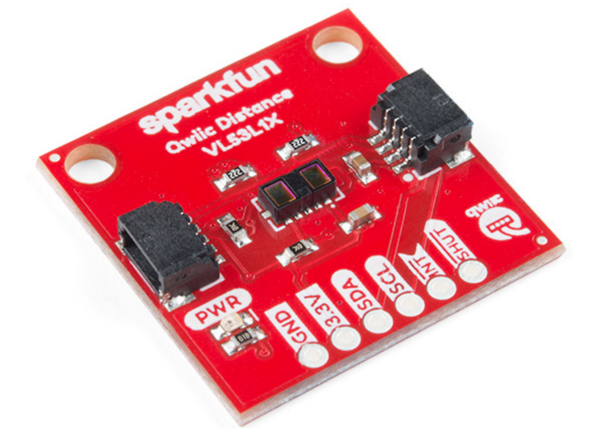


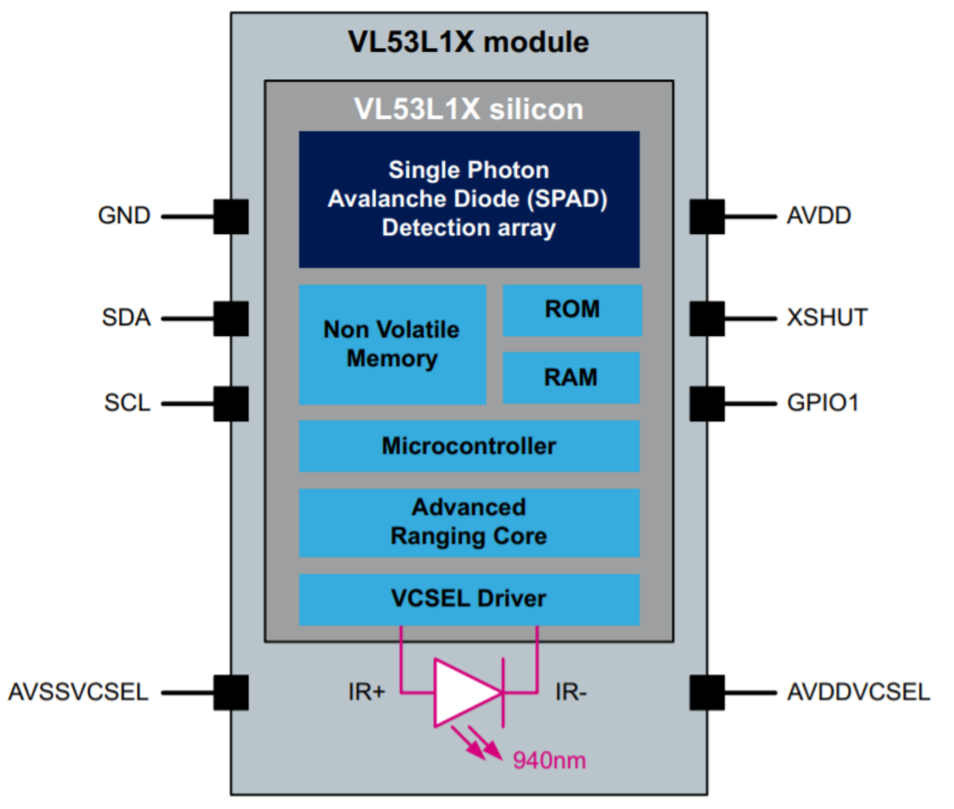


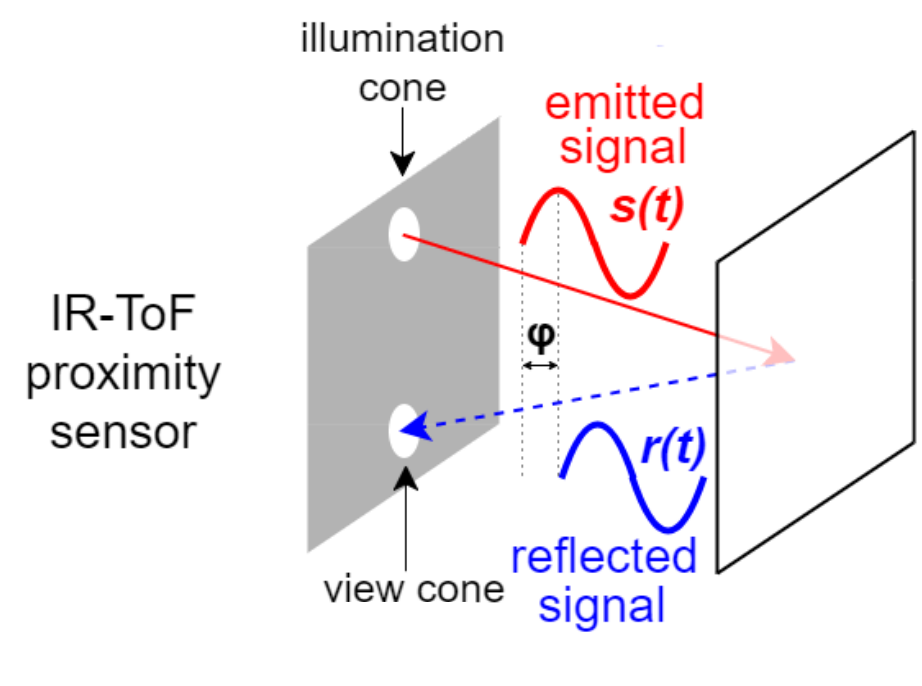


# Time of Flight IR Sensor

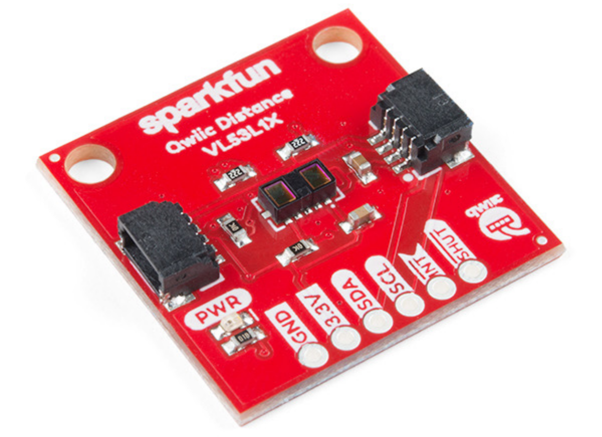
- Emit a pulse modulated signal, record time *t* until return!
  - r = t*c/2
  - c = speed of light = 299,792,458 m/s
- Mostly insensitive to texture, color, ambient light
- Outputs (Distance in mm, return signal rate, ambient signal rate, range status)





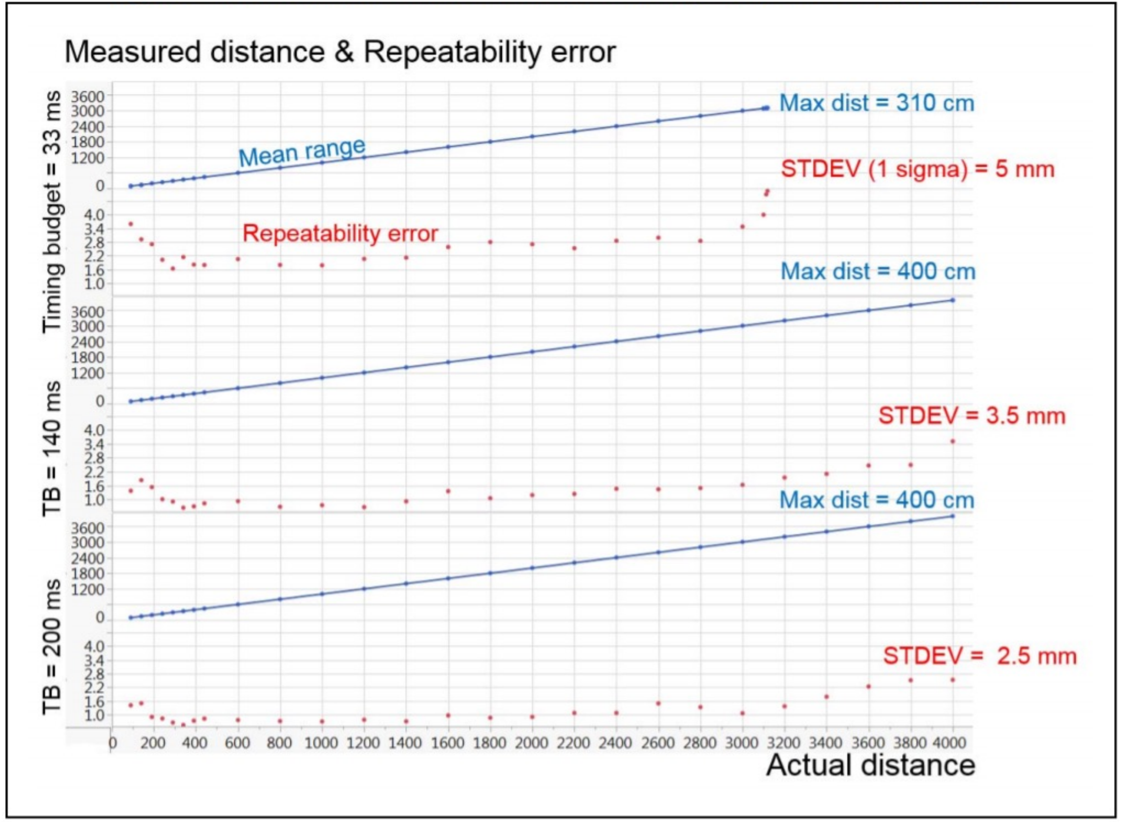


# Time of Flight IR Sensor



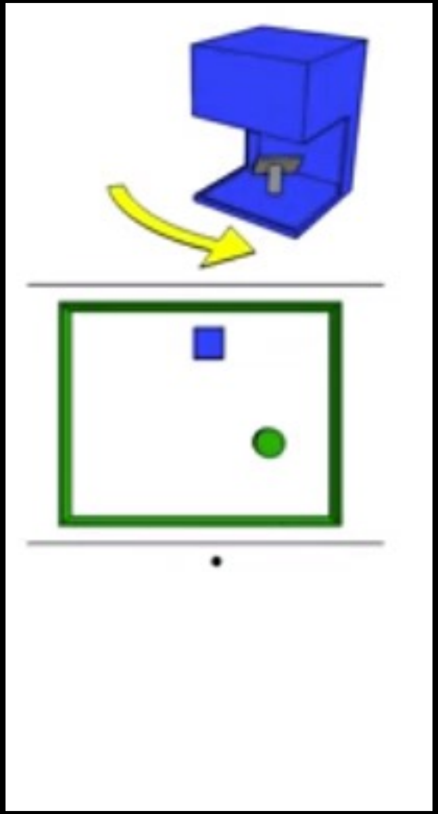
- Emit a pulse modulated signal, record time *t* until return!
  - r = t*c/2
  - c = speed of light = 299,792,458 m/s
- Mostly insensitive to texture, color, ambient light
- Outputs (Distance in mm, return signal rate, ambient signal rate, range status)
- Programmable FOV
- Distance mode (~1, 2, 4m)
- Timing budget
  - 20ms: short distance mode (0.05 - 1.3m)
  - 33ms: all distance modes (0.05 - 3.6m)
  - 140ms: improve reliability errors
- Newest developments
  - ToF Imager (64 pixels)

**Figure 6. Maximum distance and repeatability error vs. timing budget**

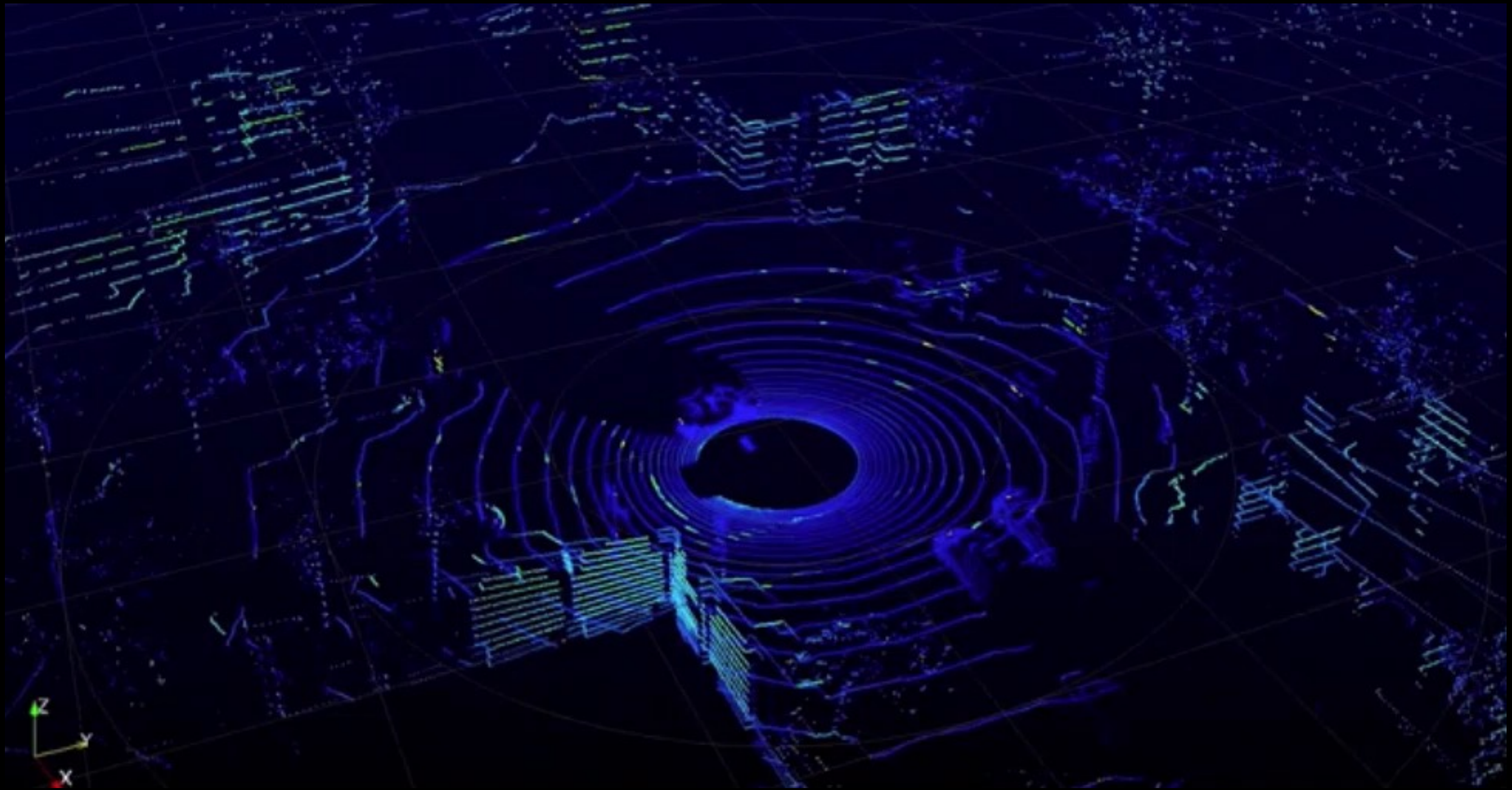


# Light Detection and Ranging Sensors

- Most common sensors on autonomous cars and robots
- Single points, line scans, full 3D
- $$$

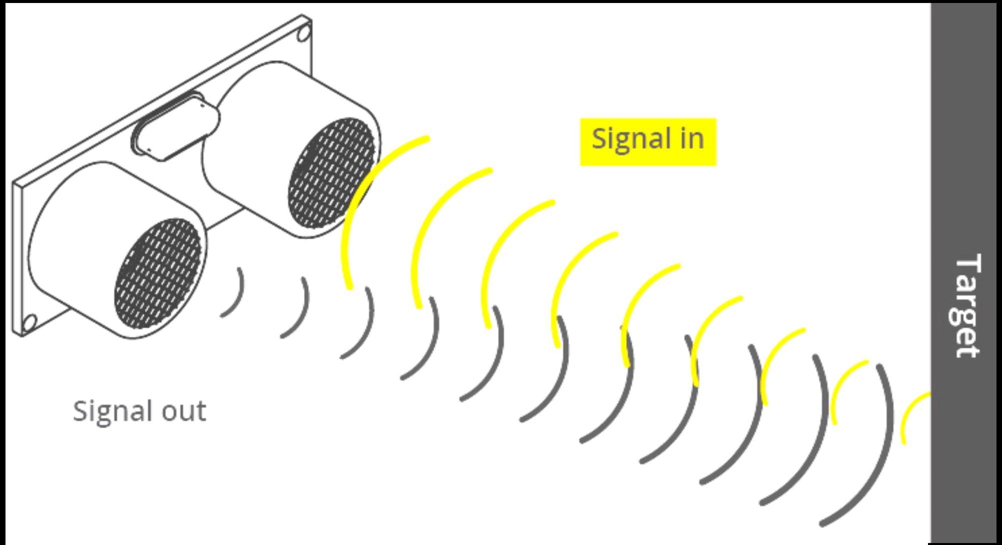


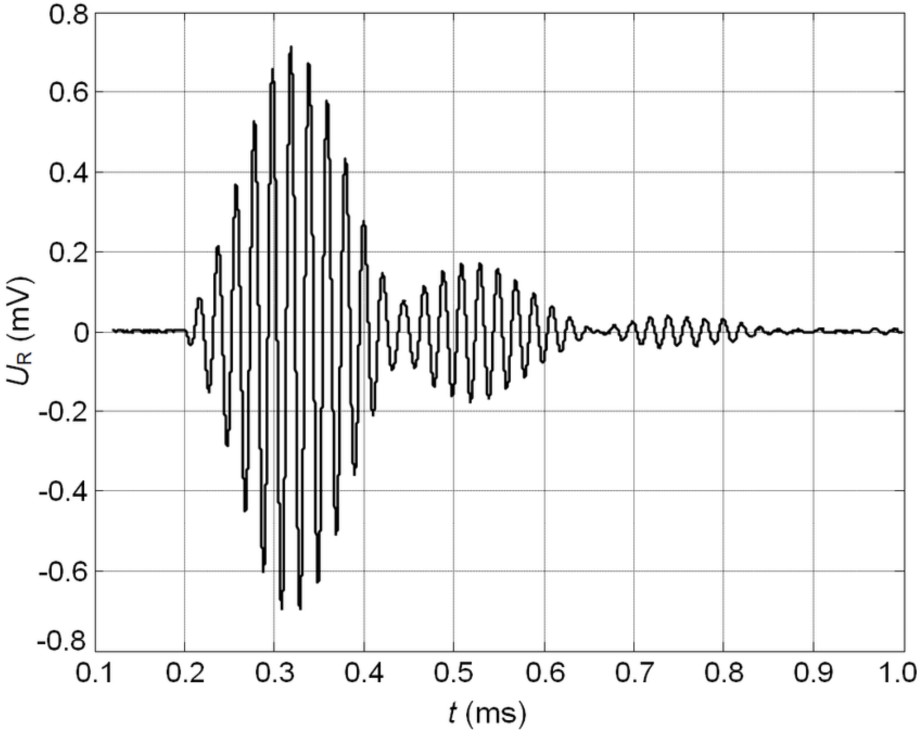
*What does the color represent?*

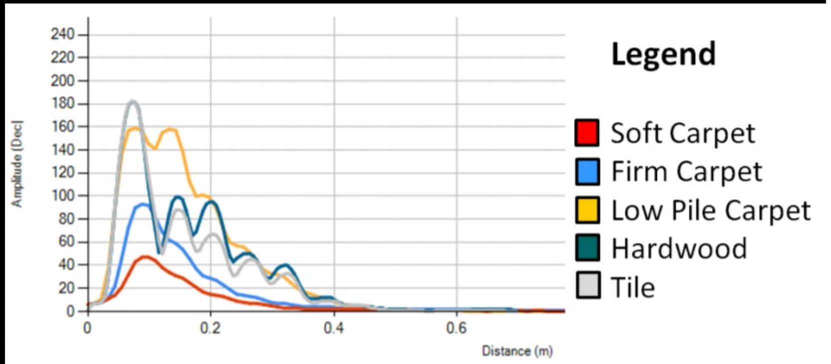


# Ultrasound (Time of Flight) Distance Sensors

- Measure the reflections of an emitted sound wave
  - $r = t * c_{sound}/2$
  - $c_{sound} = 343$ m/s
- Frequency versus resolution and range
  - 58kHz: cm resolution, range < 11m
  - 300kHz: mm resolution, range < 0.3m
- Cost is low (Sparkfun module: 4-12 USD)
- Insensitive to color, texture, glass, fog, dust, etc.
- Sensitive to humidity, temperature, audible noise, and geometry

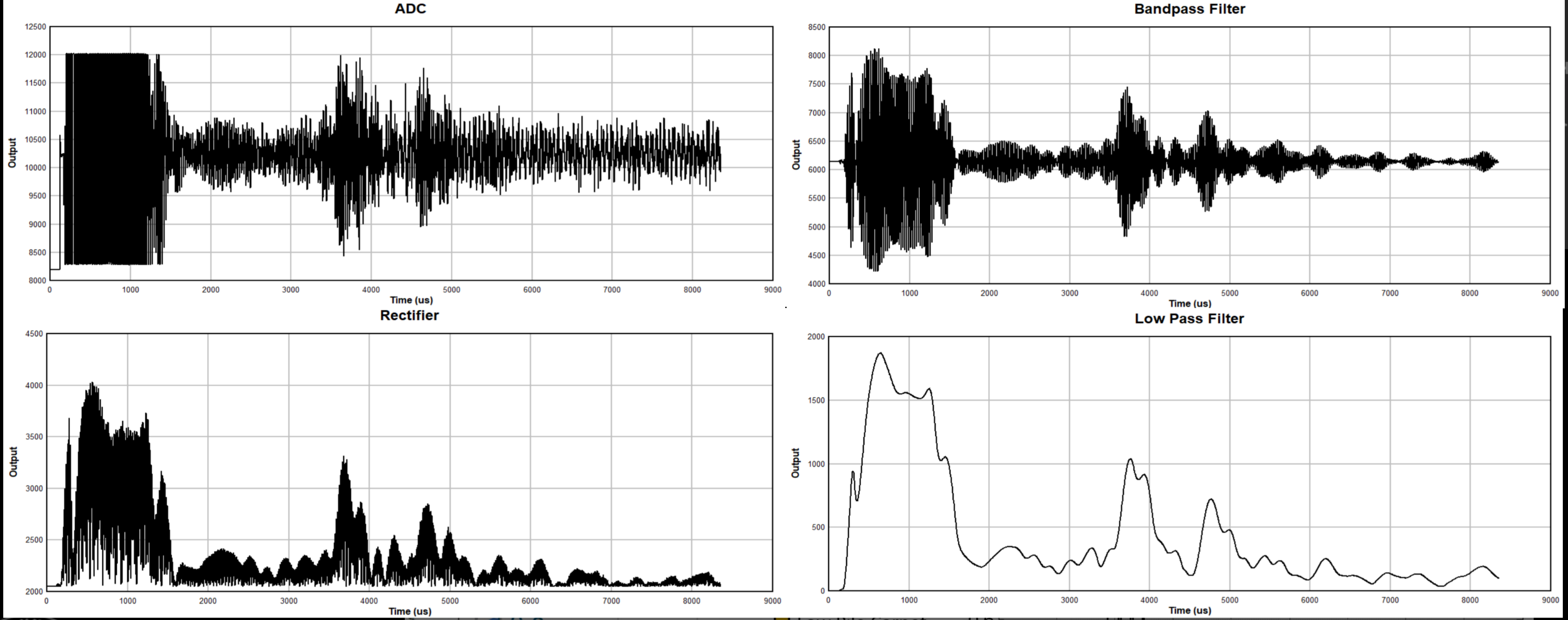


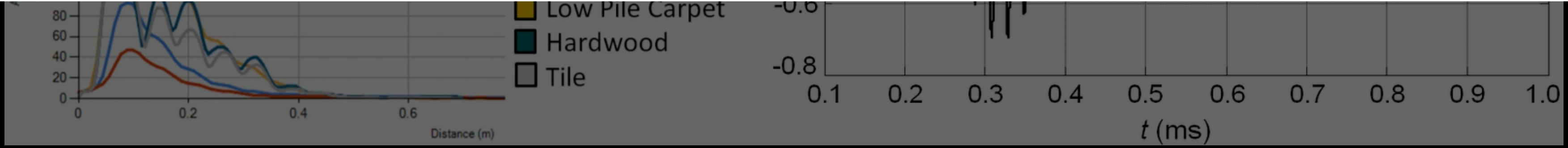




# Ultrasound (Time of Flight) Distance Sensors

- Measure the reflections of an emitted sound wave





# DIY-level Distance Sensors

| Technology | Application | Pros | Cons |
|---|---|---|---|
| Amplitude-based IR | <10cm | • ~ 0.5 USD<br>• Small form factor | • Depends on target reflectivity<br>• Does not work in high ambient light |
| IR triangulation | <1m | • Insensitive to surface color/texture/ambient light | • ~ 10 USD<br>• Does not work in high ambient light<br>• Bulky (1.75″ × 0.75″ × 0.53″)<br>• Low sample rate (26Hz) |
| IR Time of Flight | 0.1 - 4m | • Small form factor<br>• Insensitive to surface color/texture/ambient light | • ~ 6.5 USD<br>• Complicated processing<br>• Low sampling frequency: 7-30Hz |
| Ultrasonic | 0.2 – 10m | • Low cost<br>• Insensitive to ambient light and surface color<br>• Works in rain and fog | • ~4 USD<br>• Complicated processing<br>• Resolution trade off with max range<br>• Output depends on surface/geometry/humidity<br>• Bulky, sample time (tens of milliseconds)<br>• Hard to achieve a narrow FoV |

## Sources and References

- http://www.cs.cmu.edu/~rasc/Download/AMRobots4.pdf
- https://www.ti.com/lit/ug/sbau305b/sbau305b.pdf?ts=1599417595209&ref_url=https%253A%252F%252Fwww.google.com%252F
- https://hmc.edu/lair/ARW/ARW-Lecture01-Odometry.pdf
- Matlab Tech Talks on Sensor Fusion (https://www.youtube.com/watch?v=6qV3YjFppuc)

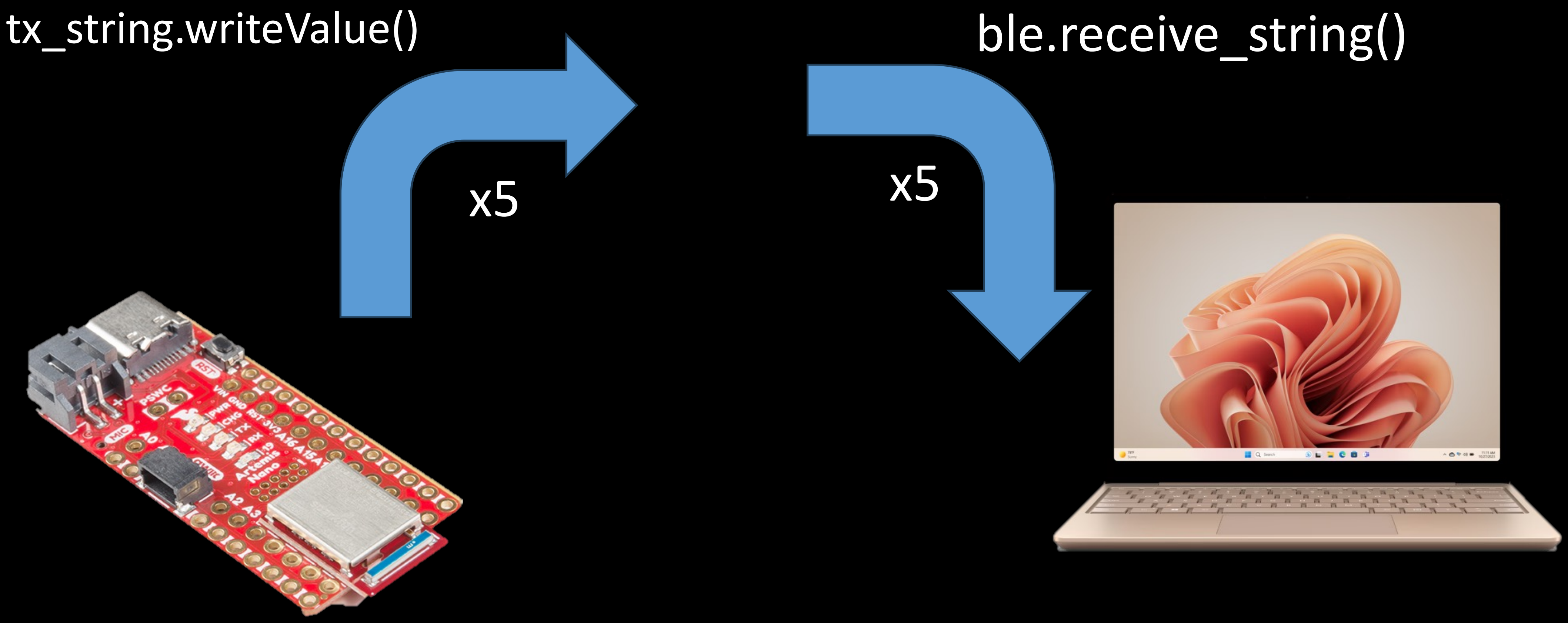
## Introduction to Lab 2

- Last 10min of class

Dr Jonathan Jaramillo
jdj78@cornell.edu

# Asynchronous Programming (sort of)

# Callback Functions



tx_string.writeValue()

x5

x5

ble.receive_string()

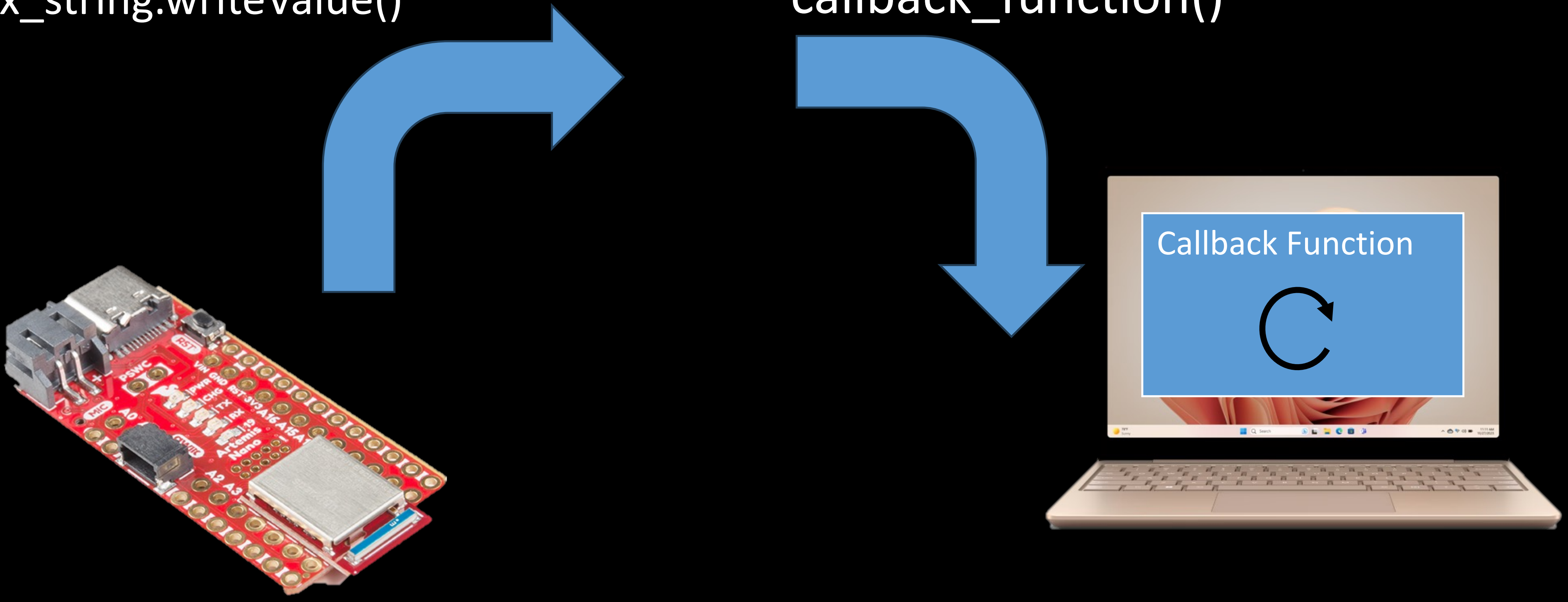
***What if you don't know how many messages get sent?***

# Callback Functions

ble.start_notify(callback_function)

tx_string.writeValue()

callback_function()



Callback Function

# How Long do Functions Take?

- Add/subtract two ints
- Add/subtract two floats
- 3x3 matrix multiply
- Read IMU accelerometer data
- Read IMU gyro data
- Set PWM value on pin
- Send a single string over BLE
- Receive a single string over BLE
- Time of Flight sensor reading

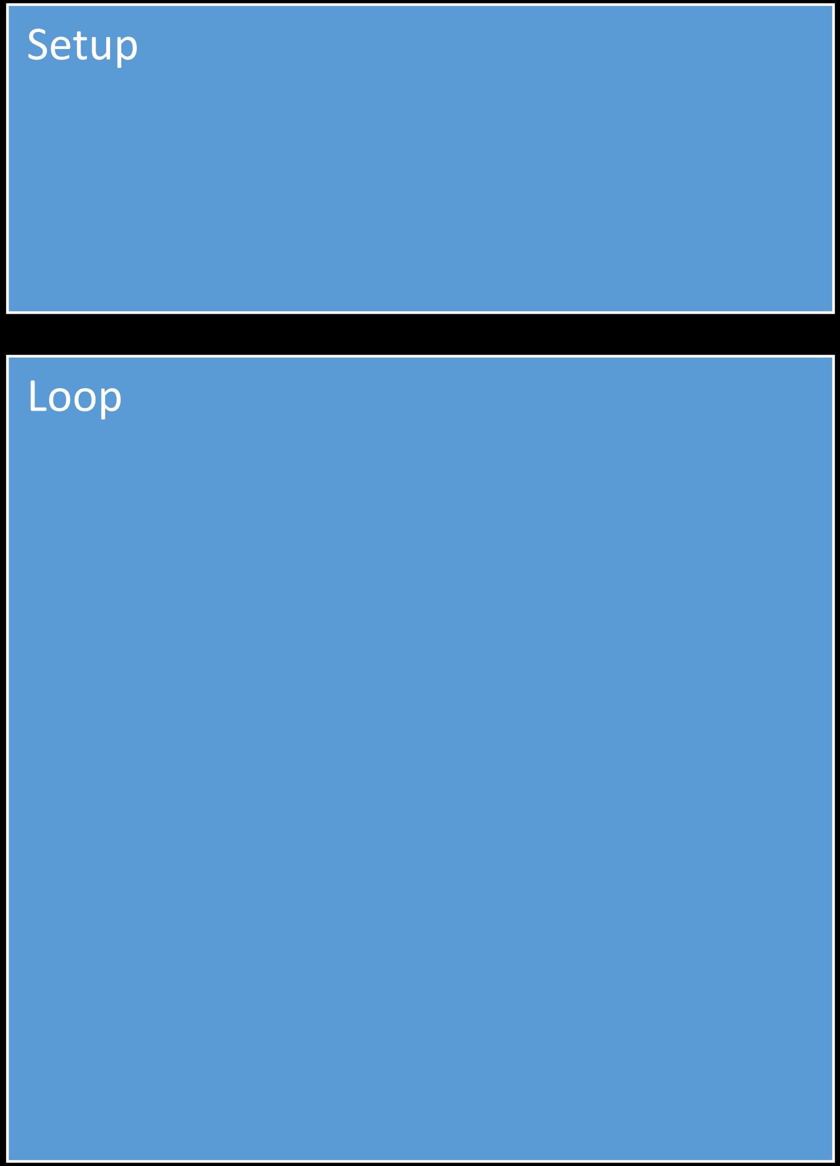
Slow

Medium

Fast

# How Fast do Functions Take?

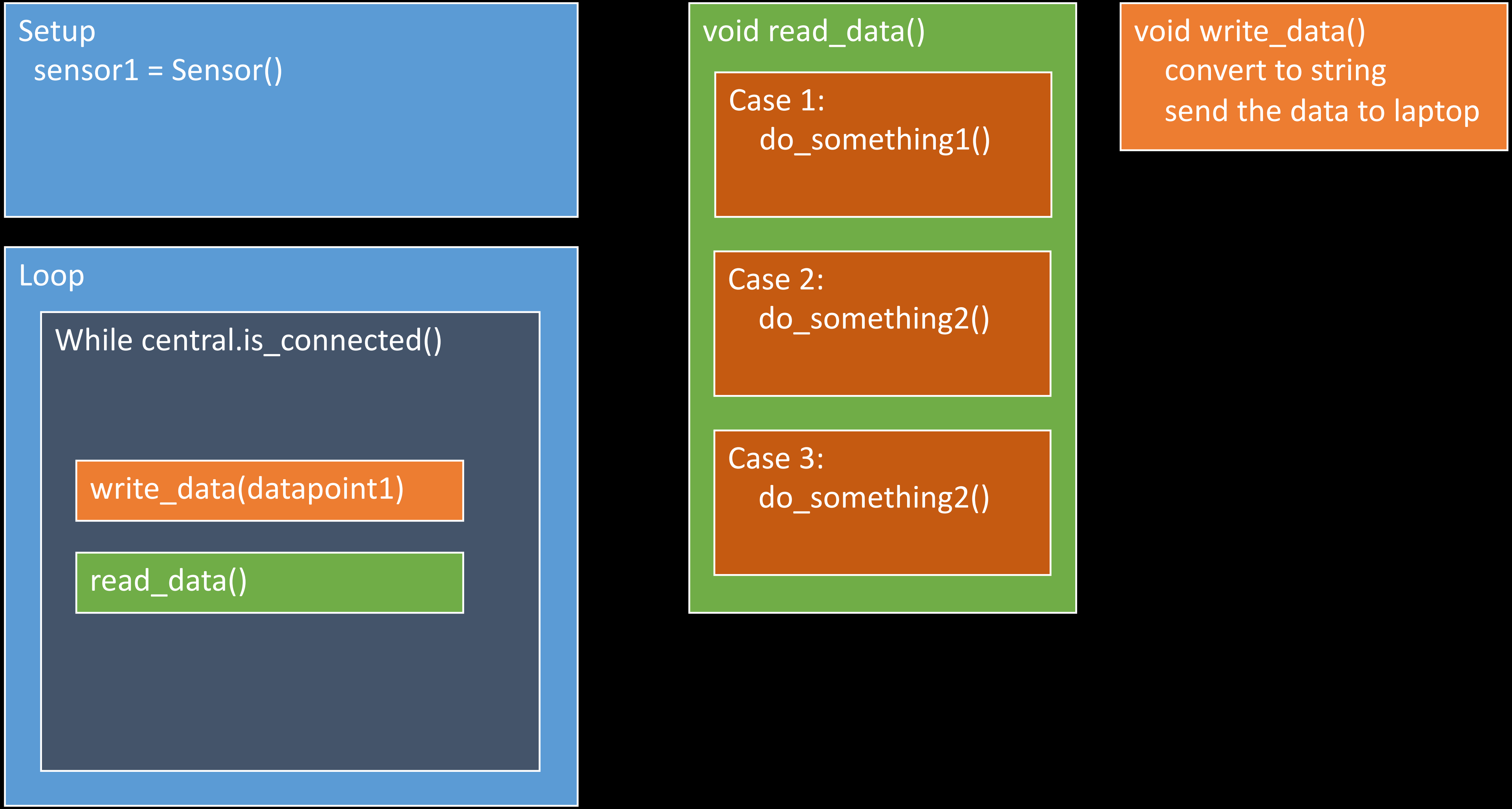
- Collect Data
  - Tof
  - IMU
- Signal Processing
  - Sensor Fusion
  - Kalman filter
  - Interpolation
- PID control
  - Distance
  - Orientation



Setup

Loop

# Sources and References



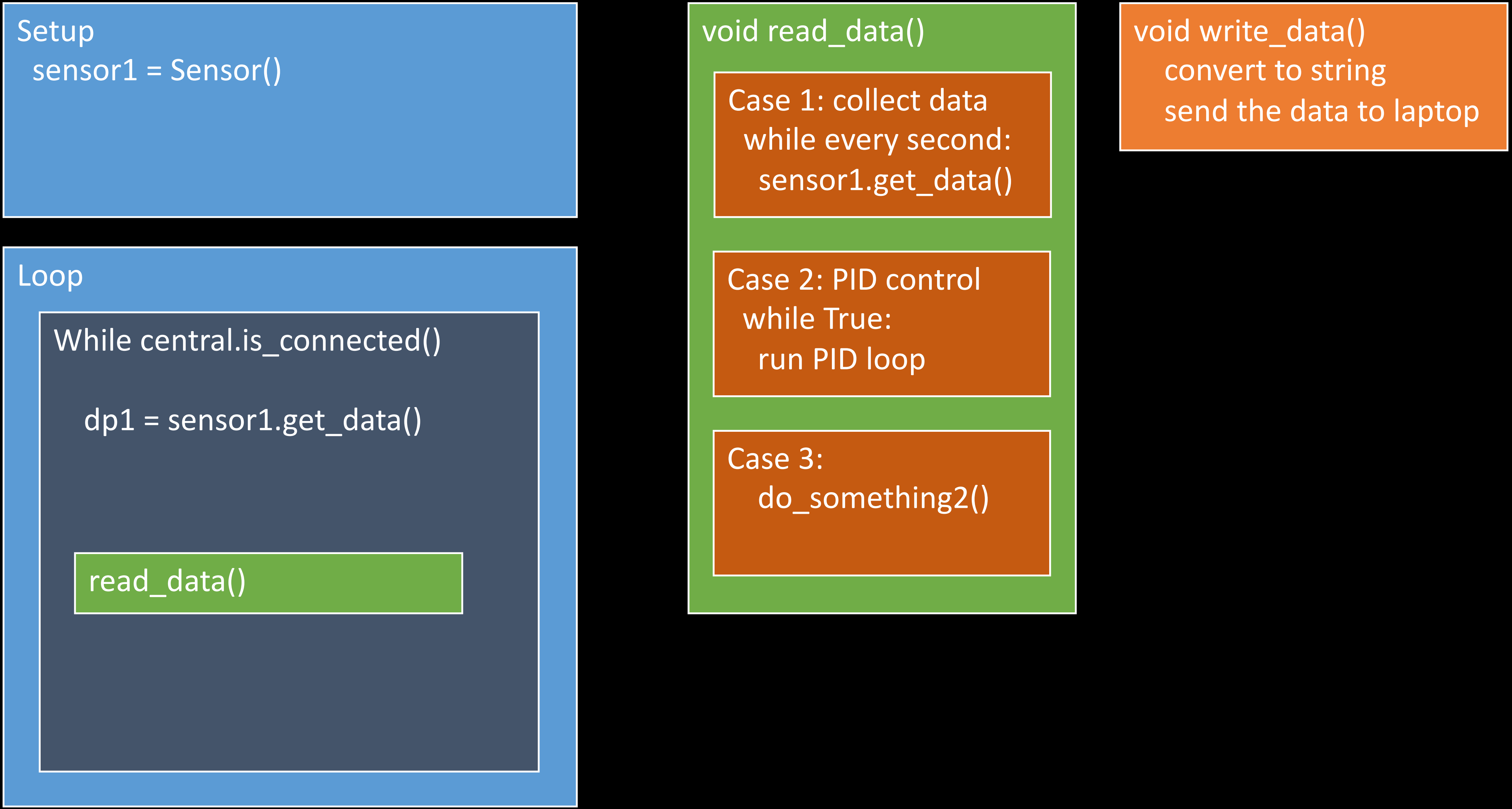
```
Setup
  sensor1 = Sensor()

Loop
  While central.is_connected()
    write_data(datapoint1)
    read_data()
```

```
void read_data()
  Case 1:
    do_something1()
  Case 2:
    do_something2()
  Case 3:
    do_something2()
```

```
void write_data()
    convert to string
    send the data to laptop
```

# Sources and References



```
Setup
  sensor1 = Sensor()
```

```
Loop
  While central.is_connected()

    dp1 = sensor1.get_data()

    read_data()
```

```
void read_data()
  Case 1: collect data
    while every second:
      sensor1.get_data()
  Case 2: PID control
    while True:
      run PID loop
  Case 3:
      do_something2()
```

```
void write_data()
    convert to string
    send the data to laptop
```

# Sources and References

```
Setup
    Sensor sensor1 = Sensor()
    bool start_record = false
    int data[10]
```

```
Loop
    While central.is_connected()
        if start_record
            int dp1 = sensor1.get_data()
            data[i] = dp1
            i++

        read_data()
```

```
void read_data()
    Case 1: start
        start_record = true

    Case 2: stop
        start_record = false

    Case 3: send data
        for value in data
            write_data(value)
```

```
void write_data()
    convert to string
    send the data to laptop
```

# Sources and References

```
Setup
   Sensor sensor1 = Sensor()
   bool start_record = false
   bool start_pid = false
   int data[10]
```

```
Loop
   While central.is_connected()
      int dp1 = sensor1.get_data()
      if start_pid
         compute_pid_values()
         set_moto_values()
      if start_record
         data[i] = dp1
         i++

      read_data()
```

```
void read_data()
   Case 1: start
      start_record = true

   Case 2: stop
      start_record = false

   Case 3: send data
      for value in data
         write_data(value)
```

```
void write_data()
   convert to string
   send the data to laptop
```

# Lab 2 Bluetooth

- *NB: The lab has changed slightly from last year!*
- Good example from last year
  - Owen Deng: https://qd39l.github.io/fast-robots/lab2.html
- Lab 2
  - https://cei-lab.github.io/FastRobots-2023/Lab2.html
  - Prelab (setup VM/Jupyter and Artemis, read through the code base)
  - Tasks
    - Change the MAC address/UUID
    - Send/receive ECHO
    - Get_Time_Millis()
    - Notification handler
    - Get_Temp_5s (T:1500|C:24|T:2657|C:24.5 …etc)
    - Get_Temp_Rapid
    - Consider Artemis storage

# ECE Colloquium 4.30pm Feb 13th PH233

- Carmel Majidi (http://sml.me.cmu.edu/)