# Sensors!

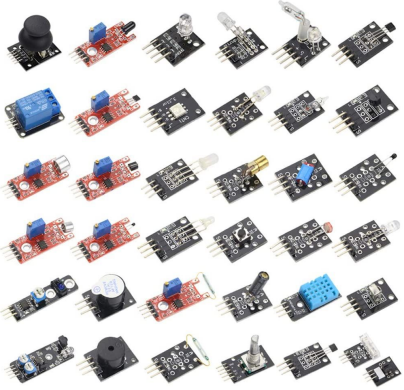
Willl Angel

Linkedin:
https://www.linkedin.com/in/william-angel/
Twitter: @DataDrivenAngel

# Overview

1. Presentation
    a. Sensors
    b. Sensor Data extraction
    c. Sensor Data Storage
2. Demo
    a. Arduino sensors
    b. ESP8266 Sensor Network Overview

# What is a sensor?



*“a device that responds to a physical stimulus (such as heat, light, sound, pressure, magnetism, or a particular motion) and **transmits** a resulting impulse (as for **measurement** or operating a control)”*
-Merriam-Webster

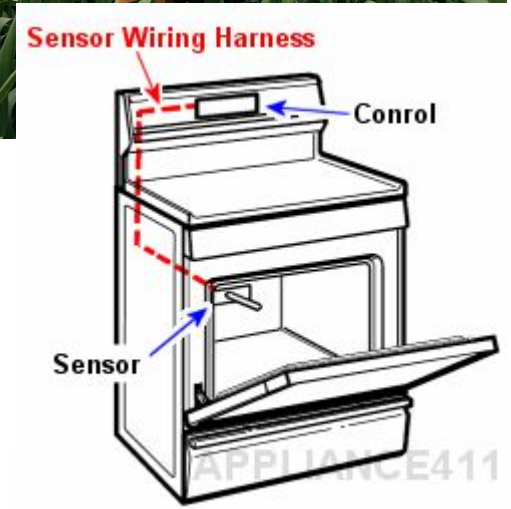
*“a device, module, machine, or subsystem whose purpose is to **detect** events or changes in its environment and **send the information** to other electronics”*
-Wikipedia

# What kinds of data do sensors collect?

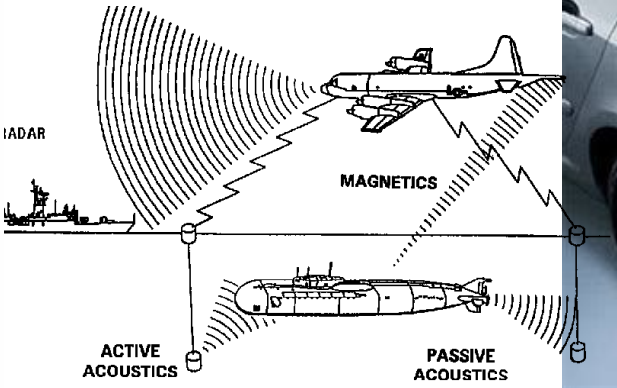
Anything!

- Temperature
- Electrical Voltage/Current
- Distance
- Time
- Force / Mass / Pressure
- Light
- Acceleration
- Images
- Sound
- Fluid Flow

# Where do we use sensors?




EVERYWHERE!


# When do we use sensors?

- When the data is necessary
- When the data can be useful
- When the sensors and controllers are so cheap that you might as well just put them in and get data.

# Microelectromechanical Systems (MEMs)

Semiconductor fabrication allows sensors to be miniaturized and produced in larger quantities at lower costs and higher qualities.

We live in a golden age of sensors.

# Microcontrollers:
# Tiny (and cheap) computers

Semiconductor Manufacturing has made computers smaller, cheaper, and more efficient.

A $3 microcontroller today has more computing power than NASA used to send astronauts to the moon.

A $40 Raspberry Pi 3 Linux Computer has **four times the power of a CREY-2 Supercomputer** (5.3 gflops vs 1.4 gflops)

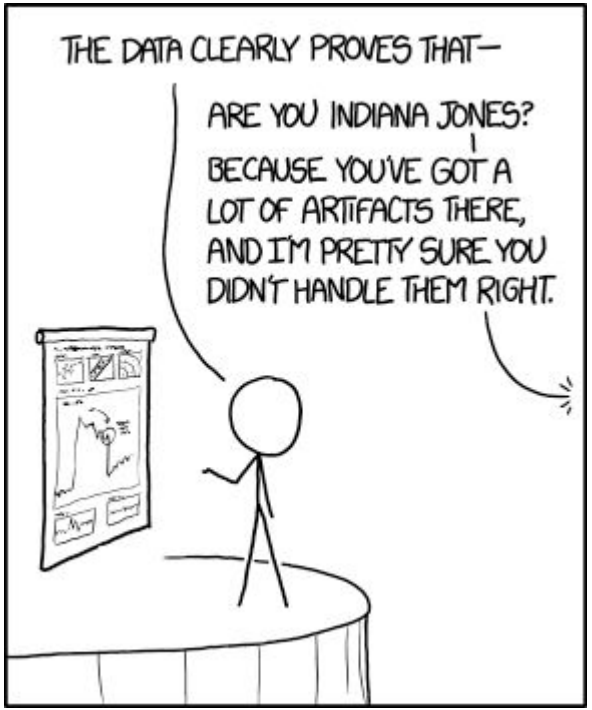
The cost of paying a college student to set one of these up is more than the hardware.

# How do useful sensors measure?

A useful sensor measures:

- Sensitively:  responds to the desired physical change in a predictable way that allows the physical property to be measured with some accuracy.
- Insensitively: does NOT respond to other physical changes that are likely to co-occur with the desired measure.
- Impact free: does NOT change the physical property it measures.

# The Pitfalls of Sensor Data



# Pitfalls of Sensor Data

Sensors are Physical devices:

- Not perfectly sensitive under all conditions: results in measurement error
  - Vibration, heat, light, moisture,  poor calibration, manufacturing defects, can chance the sensitivity of a sensor and cause it to give inaccurate readings.

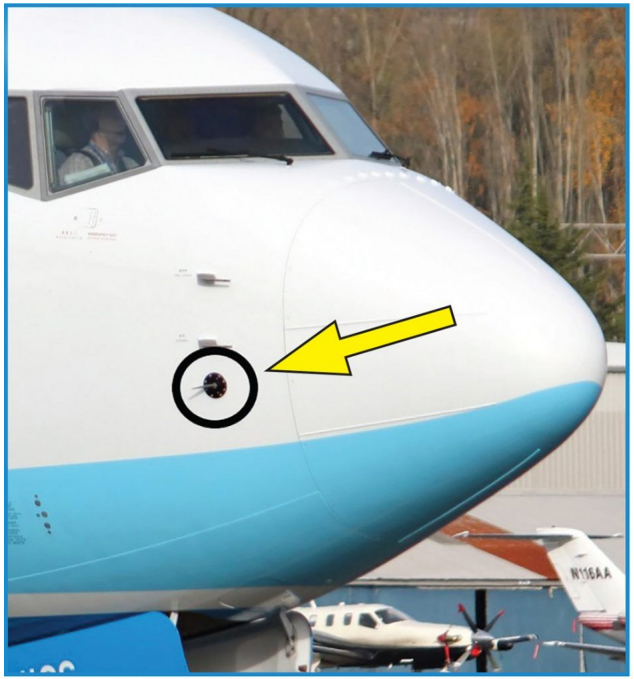
- Not always available: results in intermittent data
  - Power loss, network disruptions, Bob from IT unplugs a cable, a bird steals your camera, the sensor breaks, someone disables the smoke alarm so they can vape, someone steals the sim card from your remote sensor node and racks up €1500 in roaming charges, etc

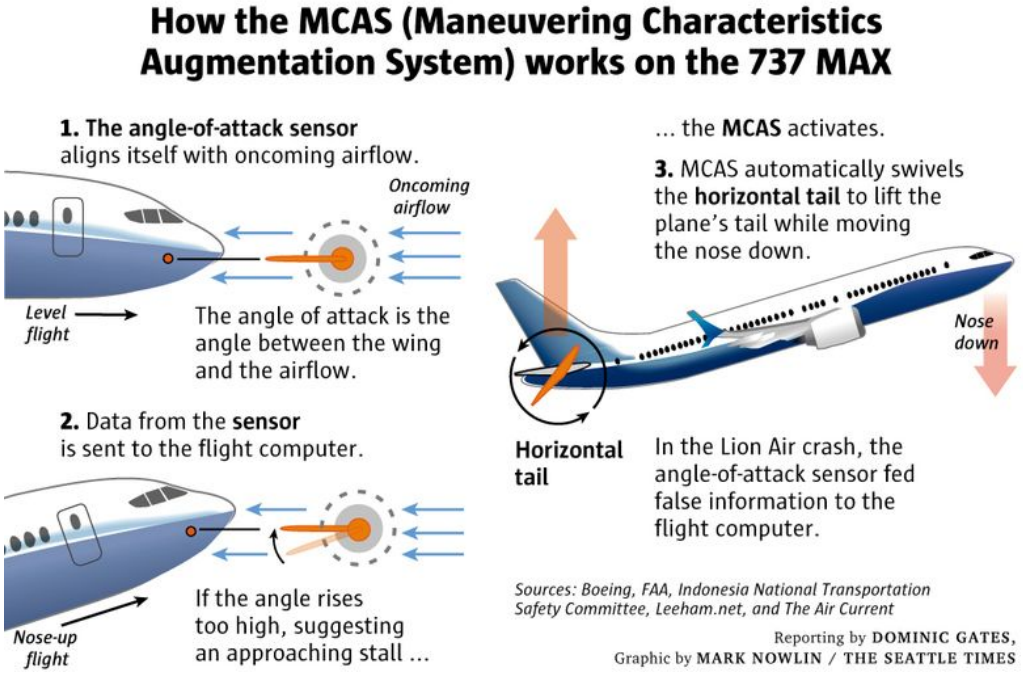
# Pitfalls of Sensor Data - Continued

Sensor data comes from systems

- Distributed computer systems are terrible:
  - Networking is hard
  - Hardware cannot be trusted
  - Things break - sometimes they only partially break
- Not always available or timely:
  - Sensors break, data gets delayed, measurements conflict,
  - Sensor data collection computers break or degrade

# 346 reasons to avoid data pitfalls





# Tips to avoid sensor data pitfalls: Trust no one, not even yourself

As a system designer/operator:

- Be Paranoid: Cultivate an Operations Mindset and anticipate potential faults
- Build redundancy and monitoring

As a data practitioner working with data from sensors:

- Understand where the data comes from and how it is collected
- Look for anomalies, missing data, incomplete data, wrong data, mislabeled data.
- Bootstrap trust: look for sensor readings that can be used to increase confidence in other sensors.

# Data Extraction

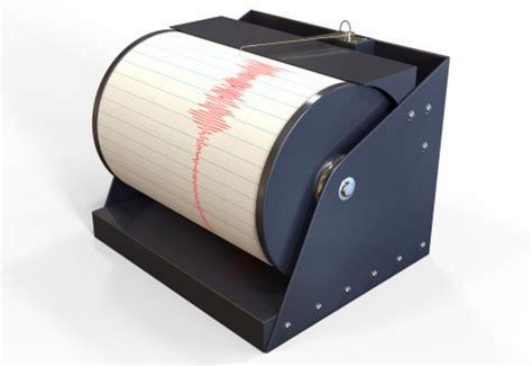
How do we get data out of a sensor anyways?

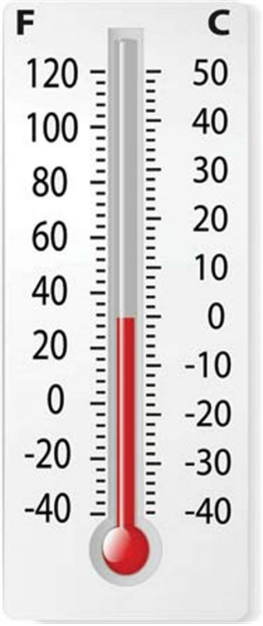
# Data Extraction

Sensor Data is useful when:

1. A human sees the data
2. A human could see the data ( but doesn’t want to)
3. An algorithm uses the data

# Methods for sensor reading:






Data can be read from a sensor:

- Manually
- Mechanically
- Electrically
- Digitally

# Methods for data extraction:

Data can be extracted from a digital sensor and sent to storage:

- By Wires:
  - Electrical
  - Optical
- By radio
  - 2G SMS
  - 3G+ Cellular data
  - Bluetooth
  - RFID
  - WiFi
  - LoRa Radio
  - Packet Radio
  - Microwave Relays
  - Satellite / space communications
- By light
  - Lasers!

By Audio:

- Phone modem
- Speaker + Microphone
- Ultrasound

# Data Storage

Where do we put all this junk data?

# Methods for data Storage:

Sensor data needs to be stored somewhere

- Put it on a flash drive ~100GB
- Put it on a hard drive ~ 10 TB
- Put it on a server ~ 100 TB
- Put it on a rack ~ 1 PB
- Put it on a data center ~ 100 PB
- Build a new google ~ 10 exabytes

# Data frequency vs data storage

Data volumes get out of control with high resolution high frequency sensors.

- ~1 TB of data per aircraft flight
- Recording a 4 byte integer at 10khz (10,000/s) is 3.2 GB per day. 1.2 TB per year. PER SENSOR.
- Large scale sensor nodes can be thousands or millions of nodes

# Datapocalypse: too much data

1. Storage systems are overwhelmed by the growth in sensor data

2. Organizations are overwhelmed by the growth in data.

3. This data will only get bigger, better, and contain more potential value

   Q.E.D. Data Related Careers will stay in demand

# Questions & Demo

Linkedin: https://www.linkedin.com/in/william-angel/
Twitter: @DataDrivenAngel
Github: DataDrivenAngel