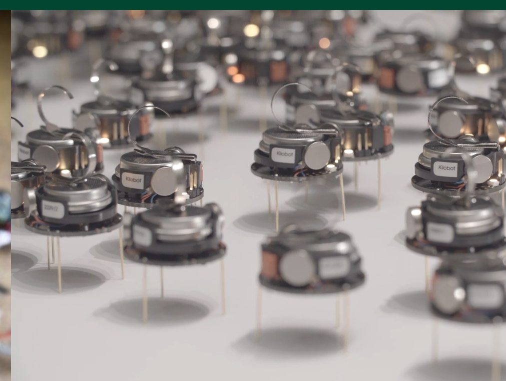
ECE693H, Spring 2025: Multi-robot System Design

"Perception 2"

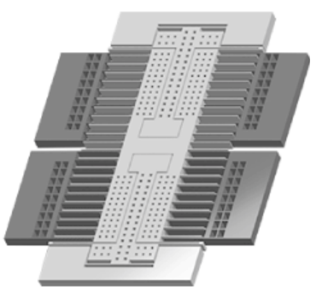
  

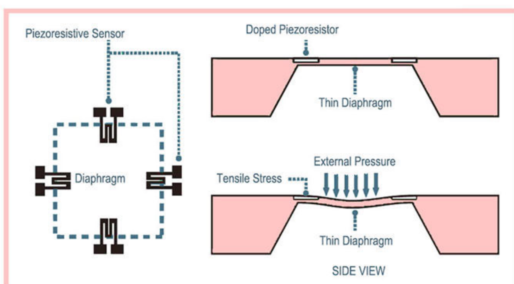
Dr. Daniel Drew

COLLEGE OF ENGINEERING
UNIVERSITY of HAWAIʻI at MĀNOA

Perception 1 Recap

1. Overview / Taxonomy
2. Sensor physics
3. Data acquisition





Active	**Distance** Time-of-flight, received angle/intensity	**LiDAR** **Ultrasonic** **Infrared**
Passive	**Intensity** Pressure Image Deflection	**IMU** **Altimeter / Barometer** **Microphone** **Camera**

Questions?

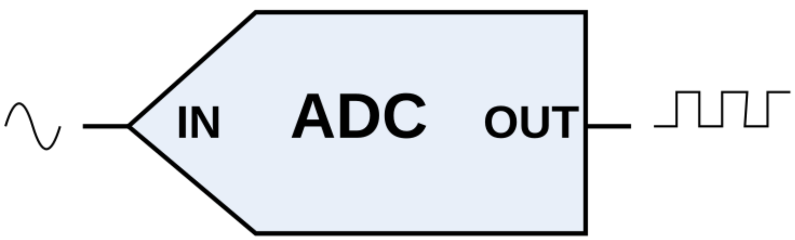
Class Overview

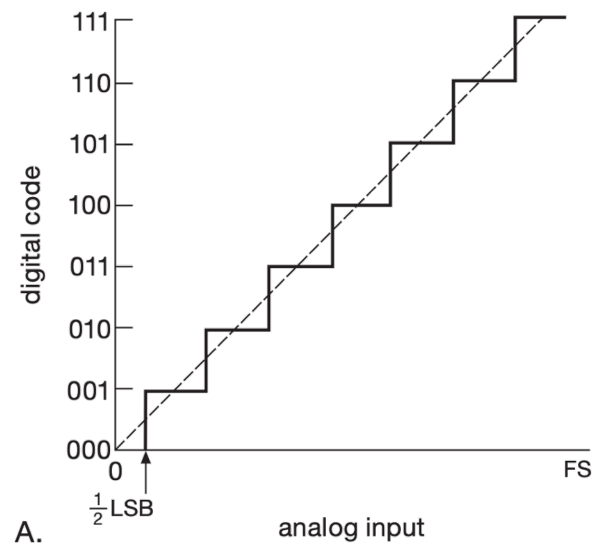
1. Lecture (~30 minutes)
2. Subsystem check-in (~30 minutes)

Perception: Lecture 2 of 2

Lecture 1: sensors → data acquisition
Lecture 2: digital signal processing

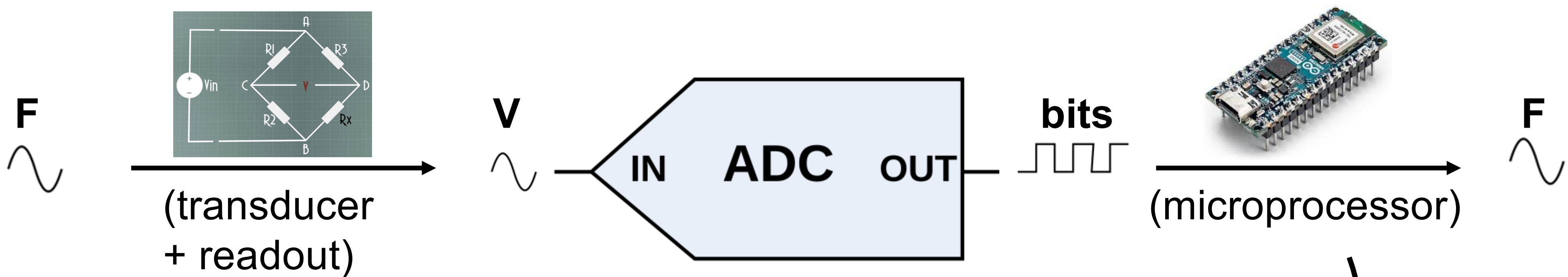
Analog-to-Digital Conversion





"3-bit" ADC:
000 to 111 in binary
0 to 7 in decimal
Resolution = $\frac{V_{ref}}{2^N}$

Sensor Scale Factors

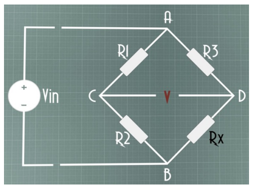


How do we know what to do?

Datasheets + Calibration

Sensor Scale Factors

F



(transducer
+ readout)

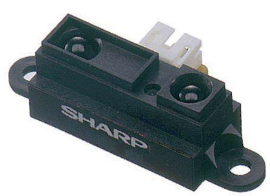
V

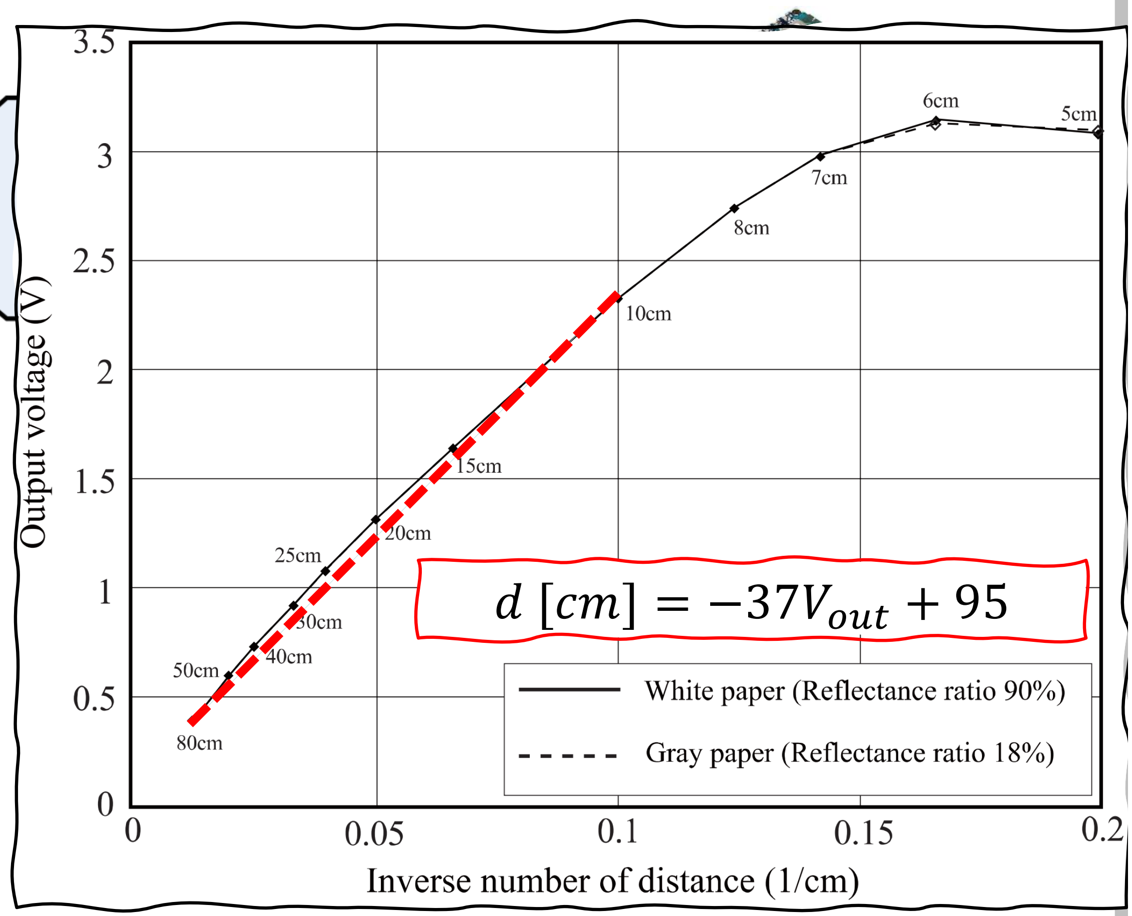
IN

SHARP GP2Y0A21YK0F

GP2Y0A21YK0F

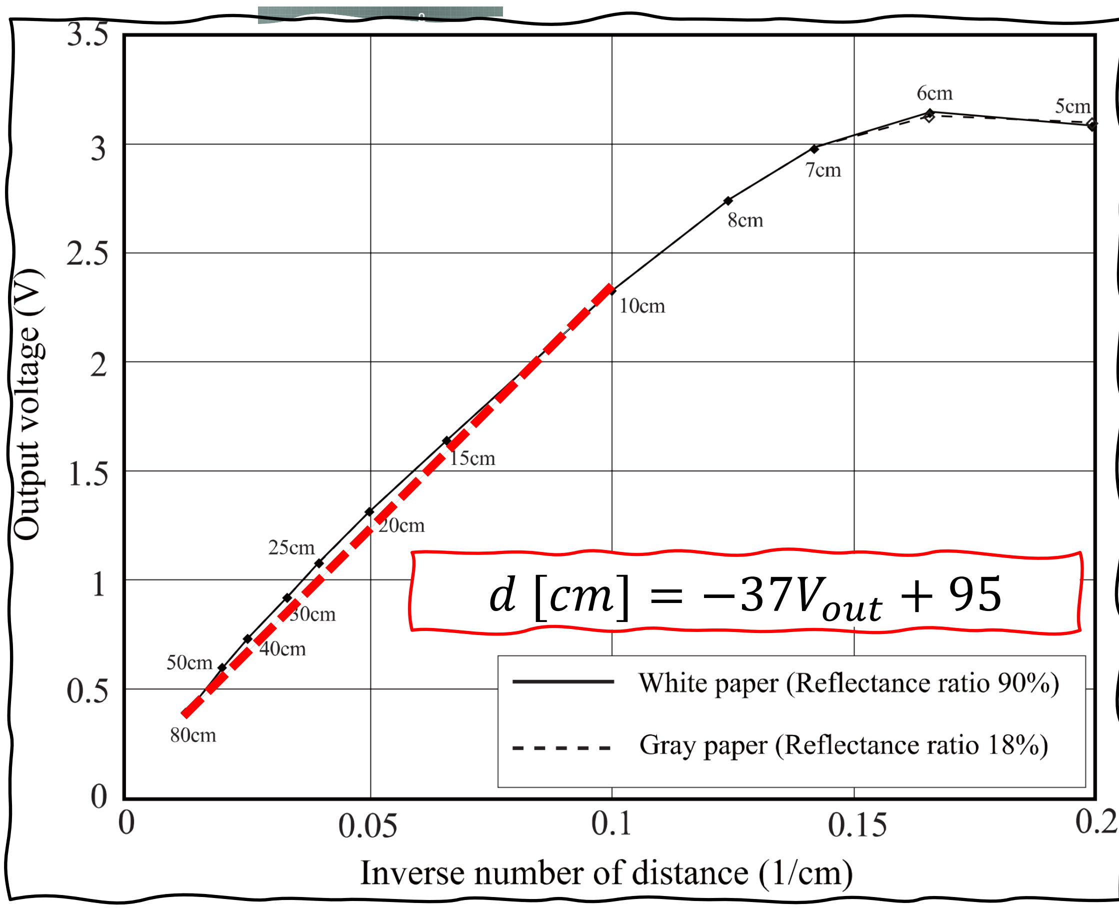
Distance Measuring Sensor Unit
Measuring distance: 10 to 80 cm
Analog output type

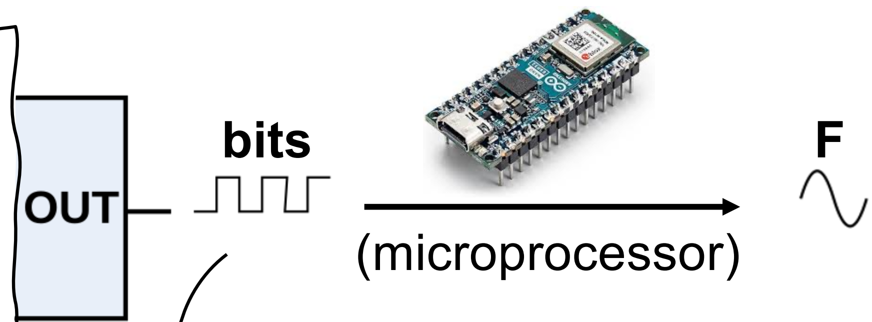




$$d\ [cm] = -37V_{out} + 95$$

Sensor Scale Factors





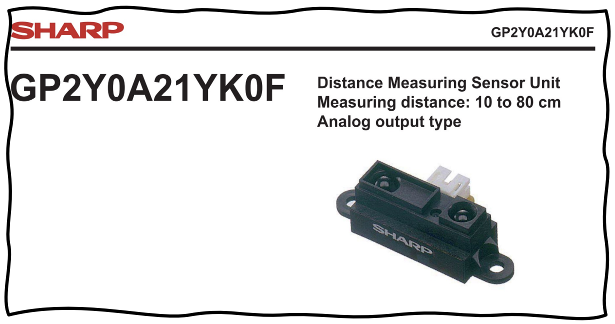
Convert bits to decimal

Convert decimal ADC_{out} to voltage:

Recall ADC resolution = $\frac{V_{ref}}{2^N}$

$$\therefore V_{out} = \frac{ADC_{out}}{2^N} V_{ref}$$

Noise and Error



Essentially, there are lots of reasons your readings may not match the datasheet values; this is true for most sensors.

White paper (Reflectance ratio 90%)

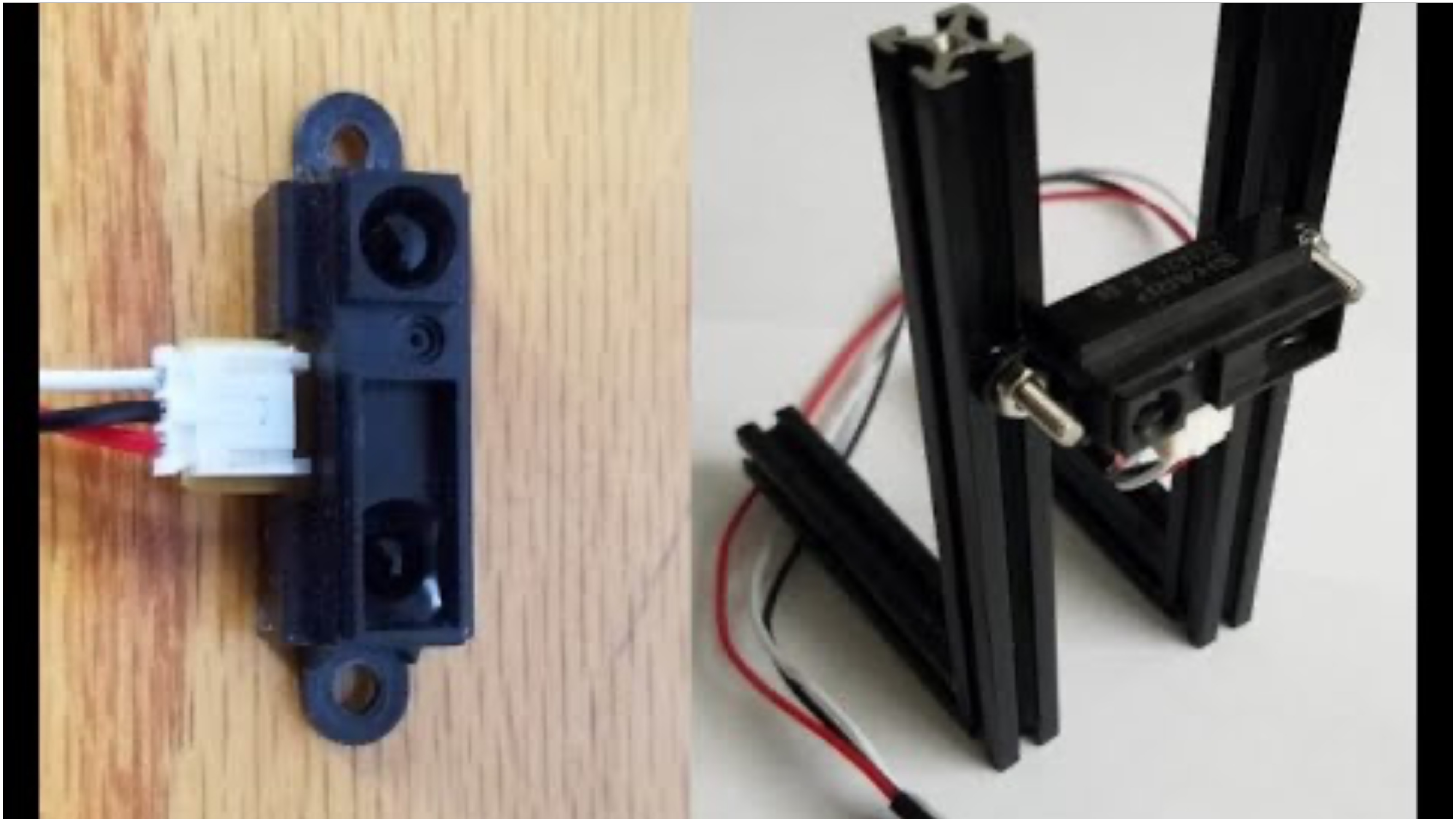
Gray paper (Reflectance ratio 18%)

● **Advice for the optics**

- The lens of this device needs to be kept clean. There are cases that dust, water or oil and so on deteriorate the characteristics of this device. Please consider in actual application.
- Please don't do washing. Washing may deteriorate the characteristics of optical system and so on. Please confirm resistance to chemicals under the actual usage since this product has not been designed against washing.

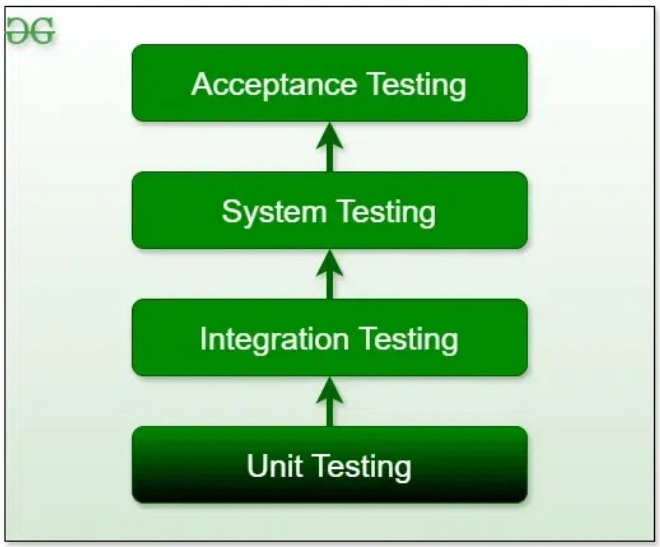
● **Advice for the characteristics**

- In case that an optical filter is set in front of the emitter and detector portion, the optical filter which has the most efficient transmittance at the emitting wavelength range of LED for this product ($\lambda = 870 \pm 70$nm), shall be recommended to use. Both faces of the filter should be mirror polishing. Also, as there are cases that the characteristics may not be satisfied according to the distance between the protection cover and this product or the thickness of the protection cover, please use this product after confirming the operation sufficiently in actual application.
- In case that there is an object near to emitter side of the sensor between sensor and a detecting object, please use this device after confirming sufficiently that the characteristics of this sensor do not change by the object.
- When the detector is exposed to the direct light from the sun, tungsten lamp and so on, there are cases that it can not measure the distance exactly. Please consider the design that the detector is not exposed to the direct light from such light source.
- Distance to a mirror reflector can not be sometimes measured exactly. In case of changing the mounting angle of this product, it may measure the distance exactly.
- In case that reflective object has boundary line which material or color etc. are excessively different, in order to decrease deviation of measuring distance, it shall be recommended to set the sensor that the direction of boundary line and the line between emitter center and detector center are in parallel.



Calibration and Testing

Experimental calibration is recommended for **every** component, especially because it also functions as a way to perform **unit tests.**

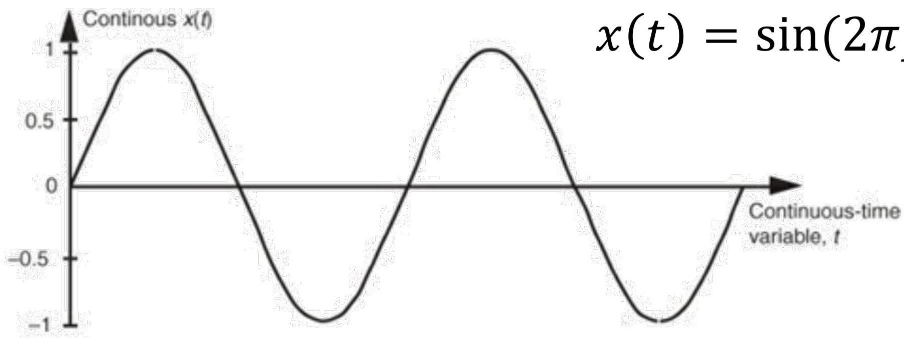


(software engineering paradigm)

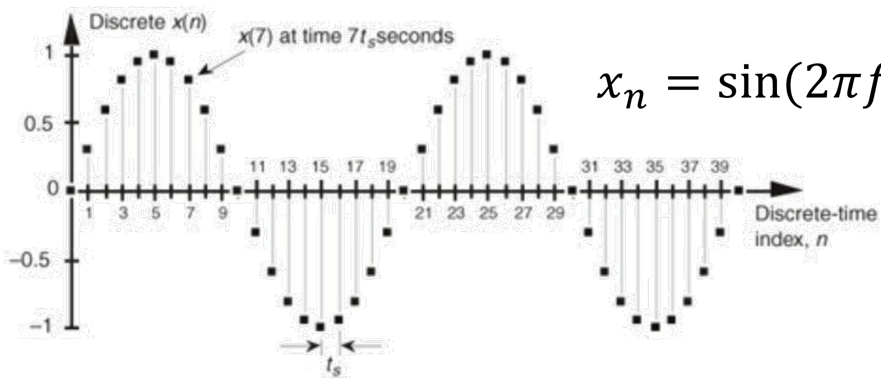
UNIT	Can I turn my motor at different speeds? Can I read from my encoders while turning the motor shaft?
INT.	Can I estimate the motor RPM from the encoder counts, as I increase motor PWM?
SYS.	Can I turn my robot 90 degrees by turning both wheels a set distance?
ACC.	Is my mechanism to program in turn angles usable by the planning team / end users?

Basic Digital Signal Processing: Signals and Systems

SIGNALS

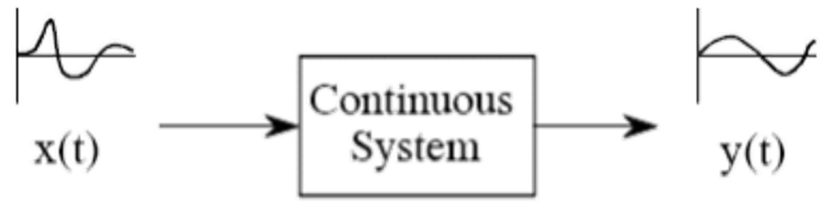


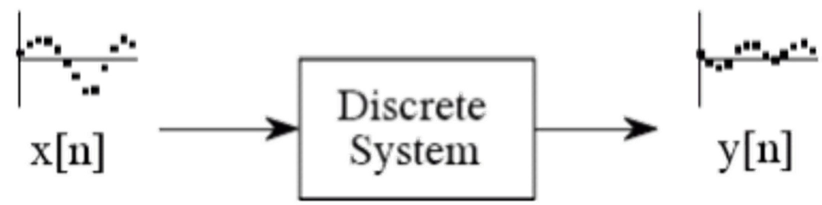
$$x(t) = \sin(2\pi f_0 t)$$



$$x_n = \sin(2\pi f_0 n t_s)$$

SYSTEMS





Basic Digital Signal Processing: LTI Systems

$$ax_1[n] + bx_2[n] \longrightarrow \boxed{\mathrm{H}} \longrightarrow ay_1[n] + by_2[n]$$

Linearity
(additivity + homogeneity)

$$x[n] \longrightarrow \boxed{\mathrm{H}} \longrightarrow y[n]$$

$$x[n - n_0] \longrightarrow \boxed{\mathrm{H}} \longrightarrow y[n - n_0]$$

Time invariance

Linear Time-Invariant (LTI) systems are typically what we design for and control;
Nonlinear control is hard, sometimes it's just as effective to linearize / assume linear!

Decomposition and Synthesis

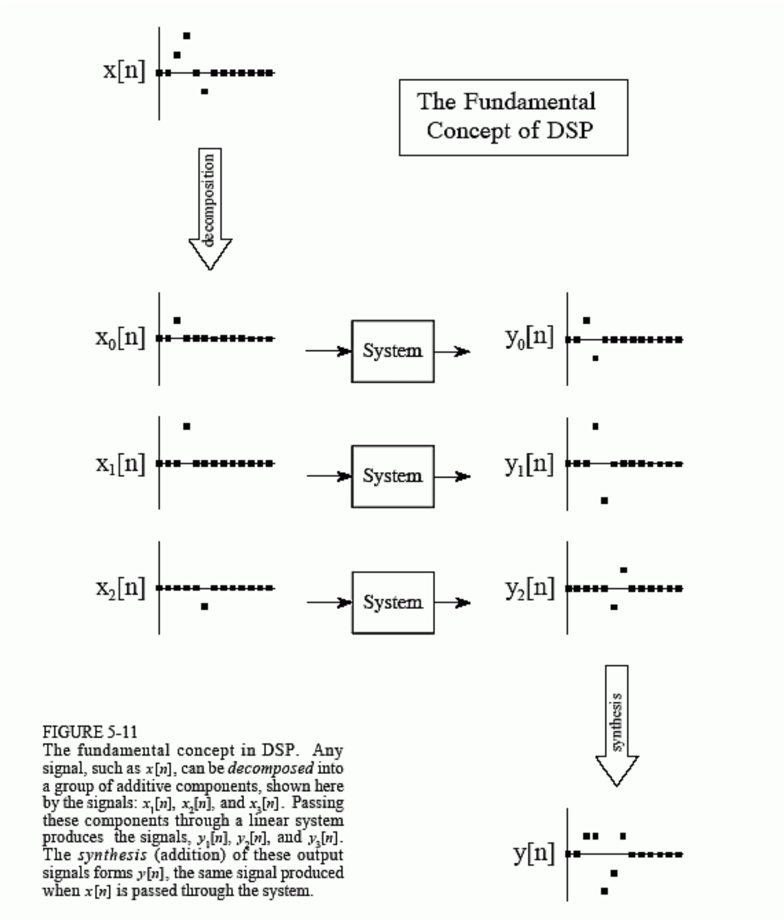
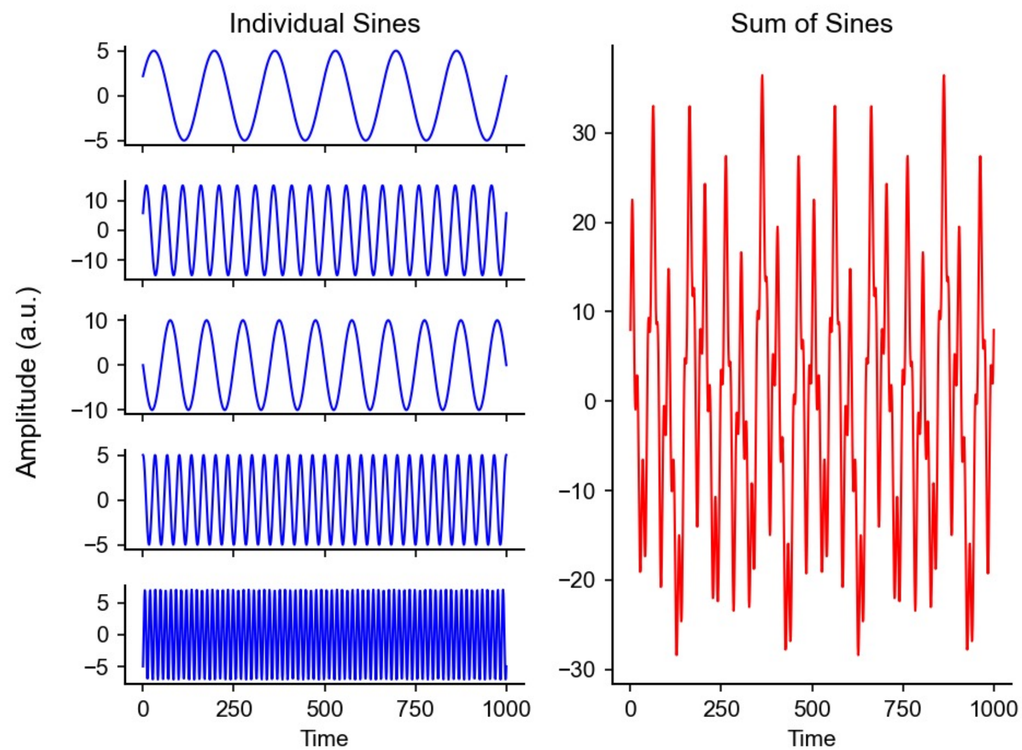


FIGURE 5-11
The fundamental concept in DSP. Any signal, such as $x[n]$, can be *decomposed* into a group of additive components, shown here by the signals: $x_1[n]$, $x_2[n]$, and $x_3[n]$. Passing these components through a linear system produces the signals, $y_1[n]$, $y_2[n]$, and $y_3[n]$. The *synthesis* (addition) of these output signals forms $y[n]$, the same signal produced when $x[n]$ is passed through the system.



Convolution

$$x[n] * h[n] = y[n]$$

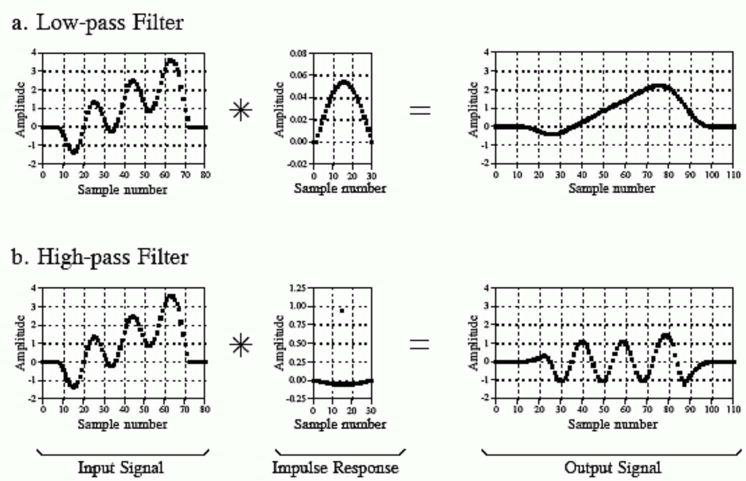


FIGURE 6-3
Examples of low-pass and high-pass filtering using convolution. In this example, the input signal is a few cycles of a sine wave plus a slowly rising ramp. These two components are separated by using properly selected impulse responses.

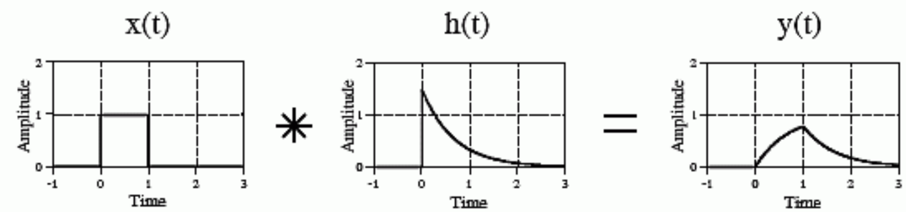


FIGURE 13-5
Example of continuous convolution. This figure illustrates a square pulse entering an RC low-pass filter (Fig. 13-4). The square pulse is convolved with the system's impulse response to produce the output.

From the convolution integral, convolution is equivalent to

$$f_1(t) * f_2(t) \equiv \int_{-\infty}^{\infty} f_1(\tau)\, f_2(t-\tau)\, d\tau$$

- Replacing the variable t with τ in both functions
- Rotating one of the functions about the y axis
- Shifting it by t
- Multiplying this flipped, shifted function with the other function
- Calculating the area under this product
- Assigning this value to $f_1(t) * f_2(t)$ at t

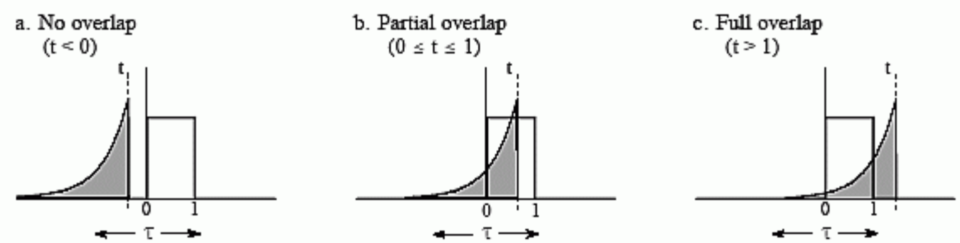
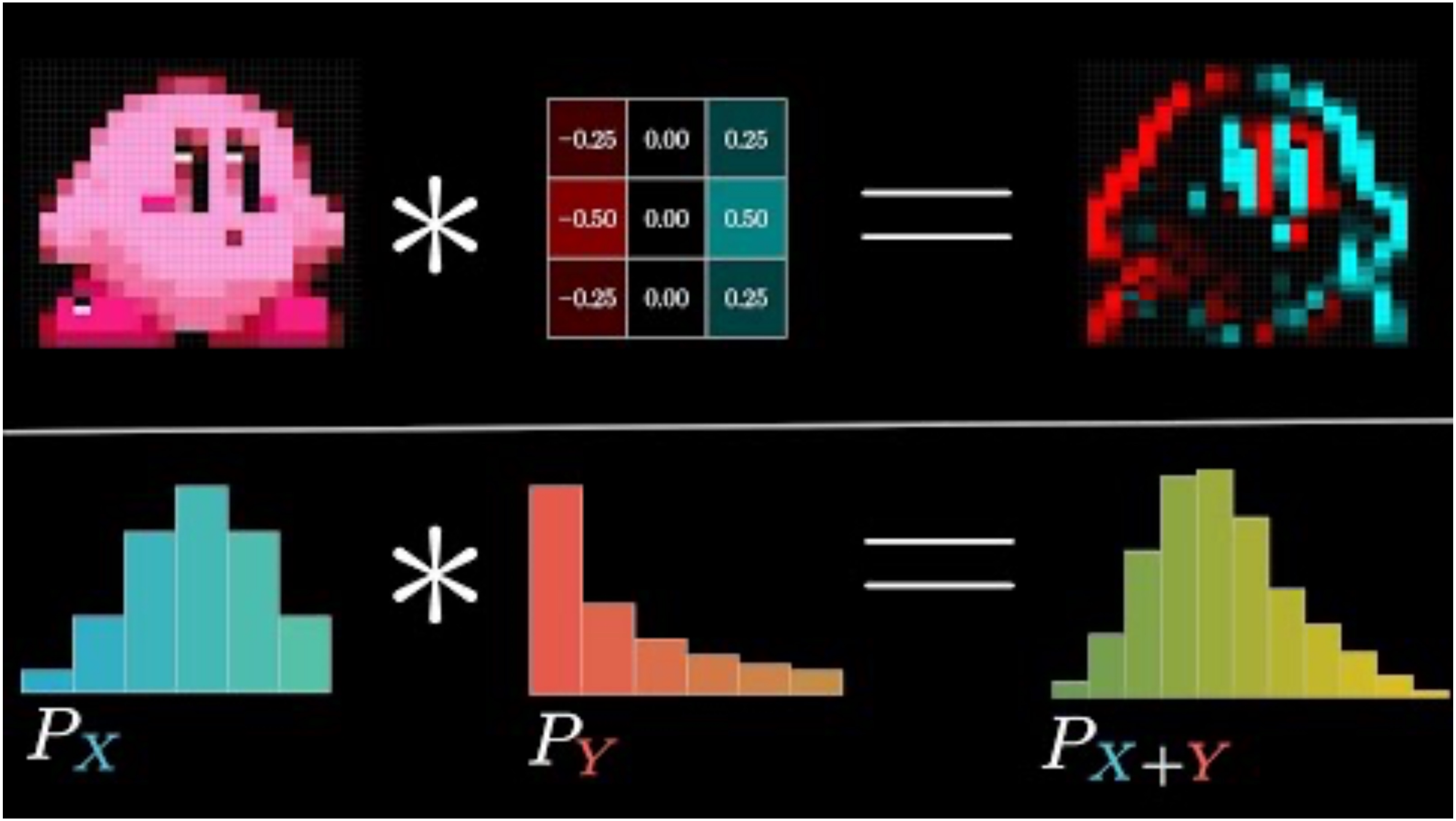
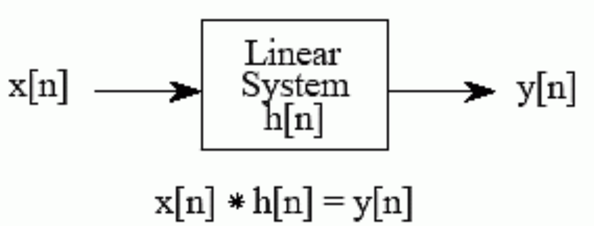


FIGURE 13-6
Calculating a convolution by segments. Since many continuous signals are defined by *regions*, the convolution calculation must be performed region-by-region. In this example, calculation of the output signal is broken into three sections: (a) no overlap, (b) partial overlap, and (c) total overlap, of the input signal and the shifted-flipped impulse response.



−0.25 0.00 0.25
−0.50 0.00 0.50
−0.25 0.00 0.25
P_X
P_Y
P_{X+Y}

Convolution



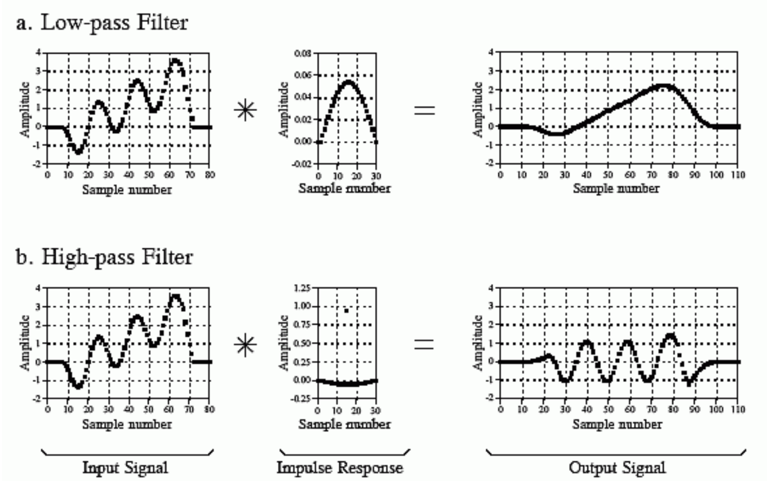
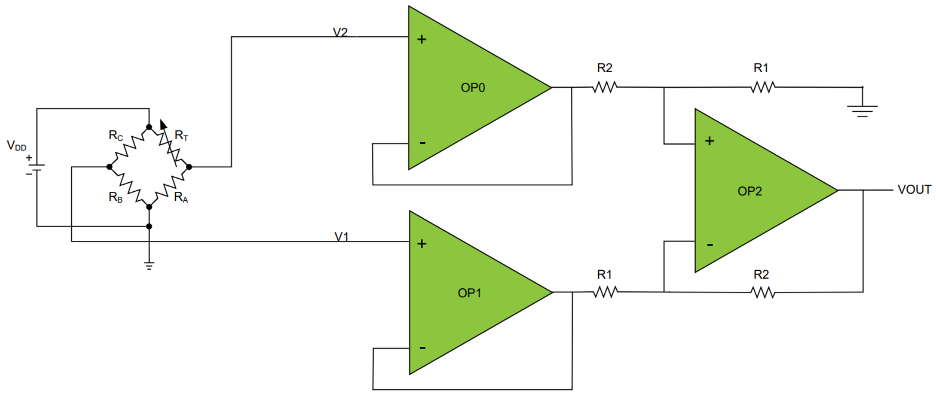


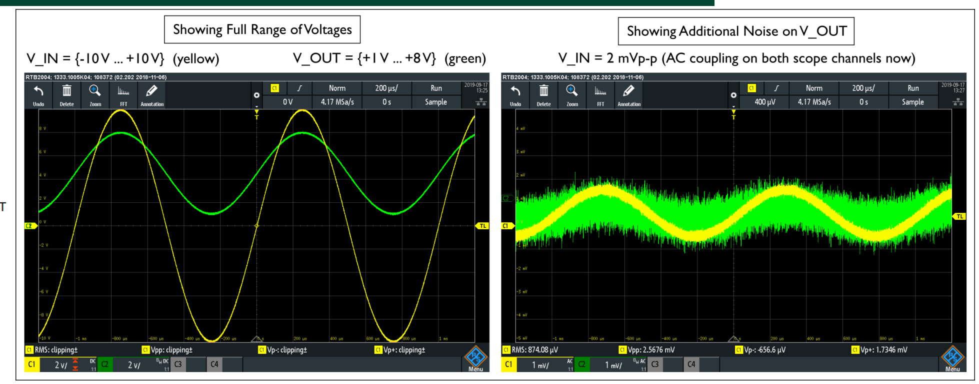
FIGURE 6-3
Examples of low-pass and high-pass filtering using convolution. In this example, the input signal is a few cycles of a sine wave plus a slowly rising ramp. These two components are separated by using properly selected impulse responses.

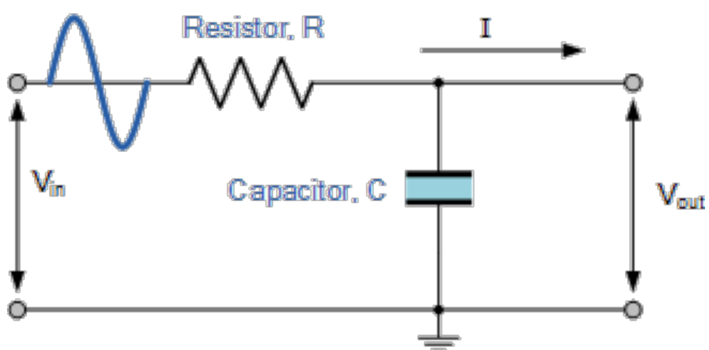
In robotics:

-Image processing (edge detection, CNNs)
-Spatial filtering (LiDAR point cloud processing)
-Odometry (motion model convolved with odometry uncertainty probability distribution)

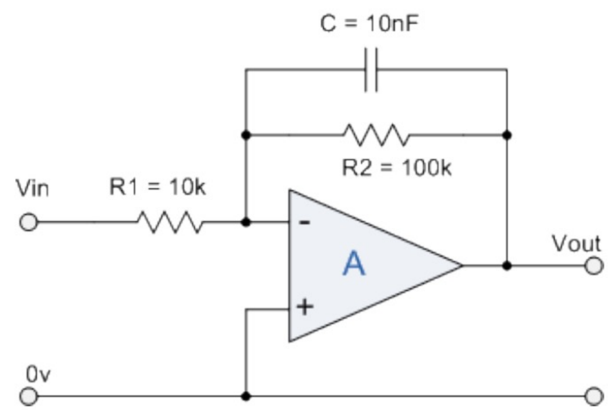
Signal Conditioning: Passive / Active Filters





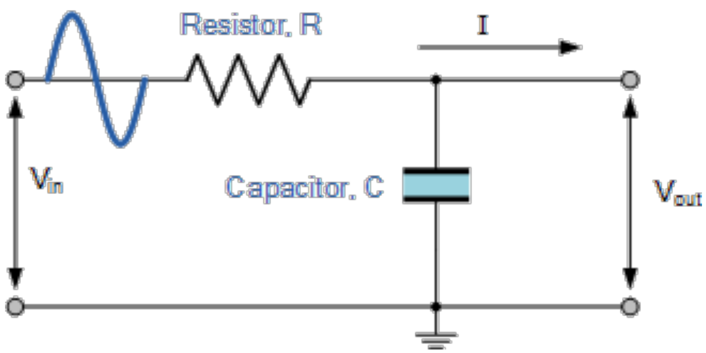


Passive filter

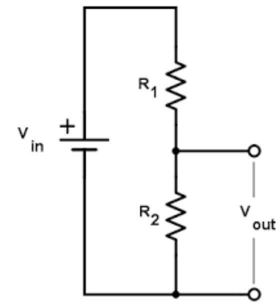


Active filter

Analog Signal Conditioning: The RC Filter



Recall the voltage divider:



$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2}$$

capacitive reactance: $X_C = \dfrac{1}{2\pi f C}$

impedance of series res. and cap.: $Z = \sqrt{R^2 + X_C^2}$

$$X_{c,100Hz} = \frac{1}{2\pi \times 100 \times 47 \times 10^{-9}} = 33{,}863\Omega$$

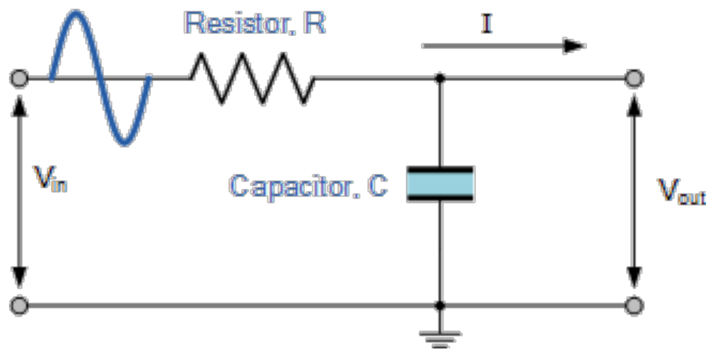
$$X_{c,10kHz} = \frac{1}{2\pi \times 10{,}000 \times 47 \times 10^{-9}} = 338.6\Omega$$

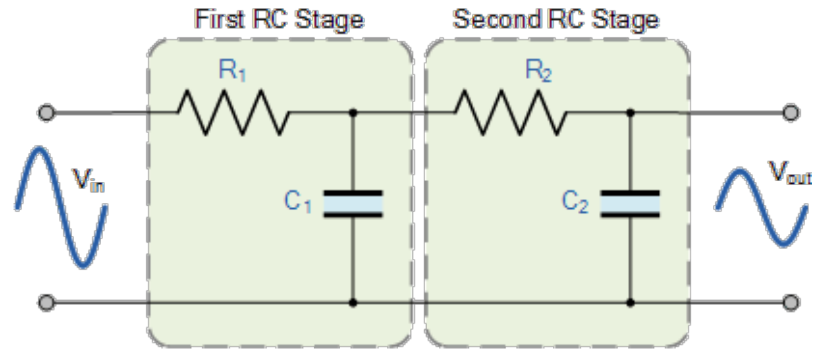
$$V_{out} = V_{in} \frac{X_C}{Z} = V_{in} \frac{X_C}{\sqrt{R^2 + X_C^2}}$$

$$V_{out,100Hz} = 9.9V \qquad V_{out,10kHz} = 0.718V$$

(10V input, 47nF cap, 4.7k resistor)

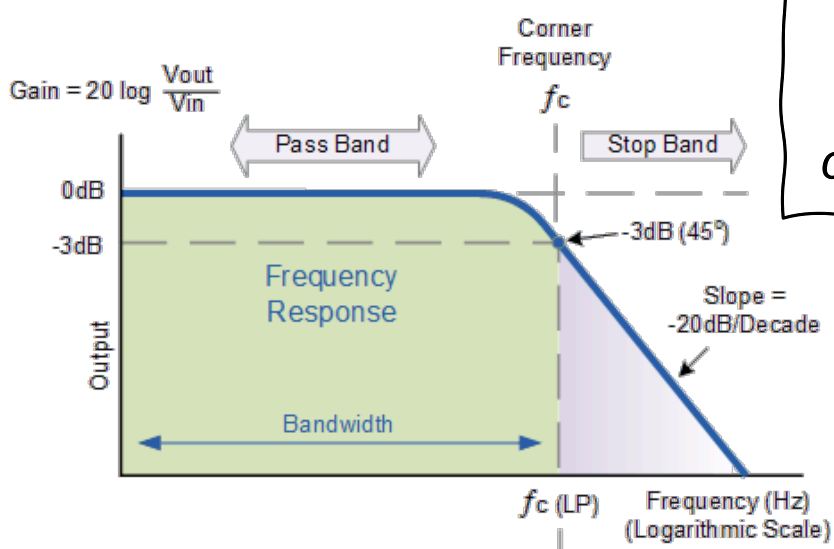
Analog Signal Conditioning: Multistage Filters

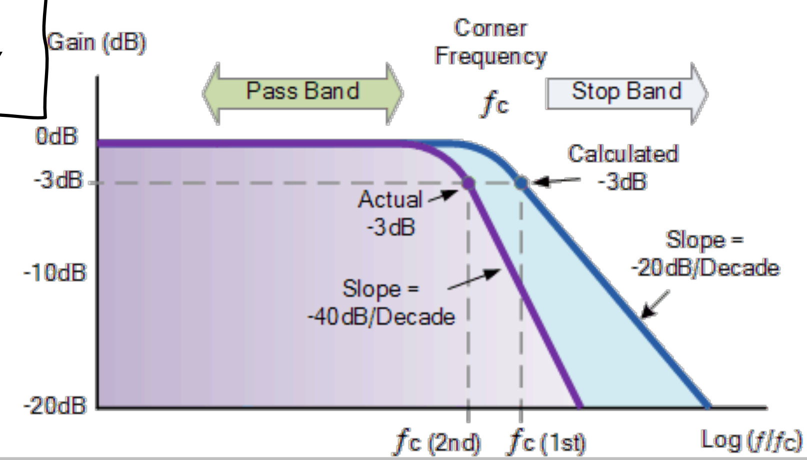




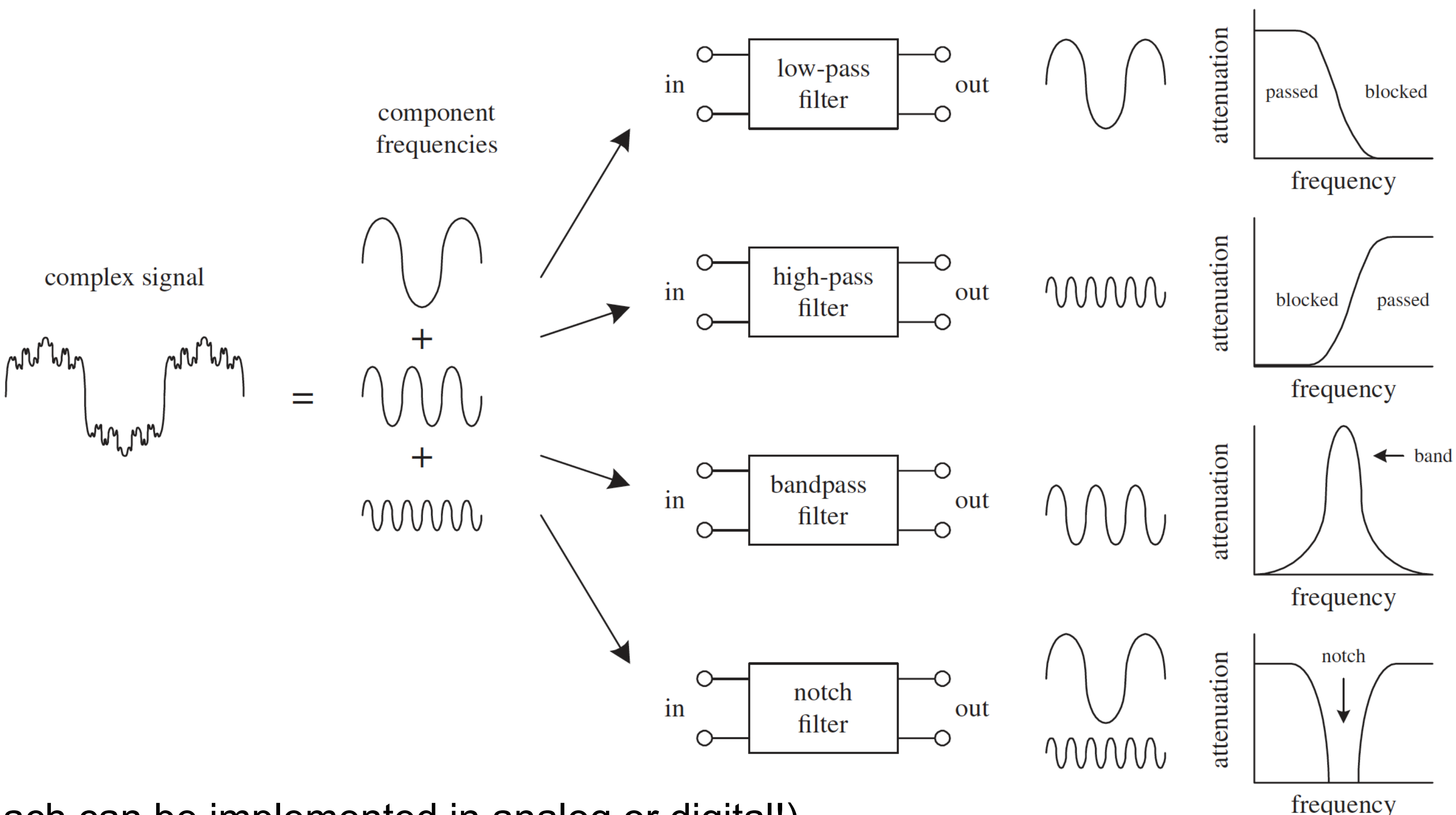
$$f_c = \frac{1}{2\pi RC}$$

cutoff frequency



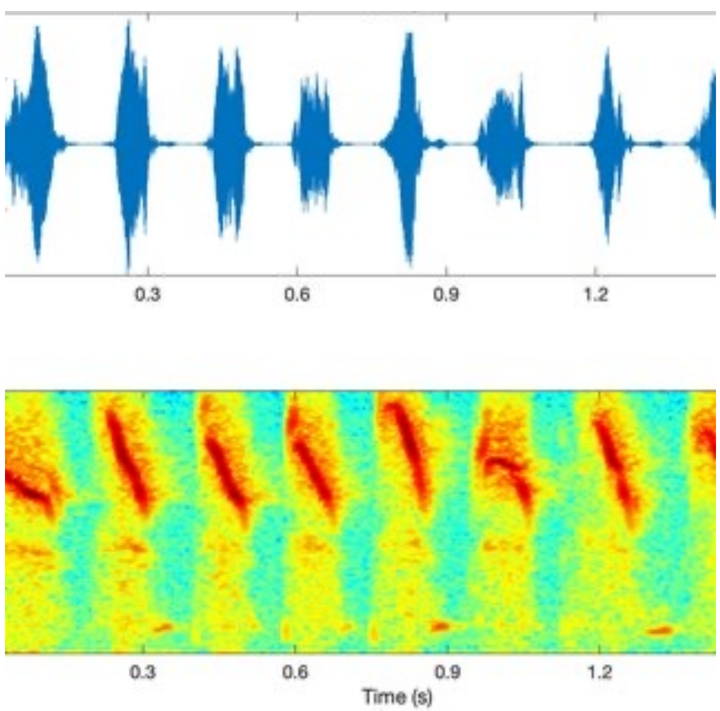


Filters at a Glance

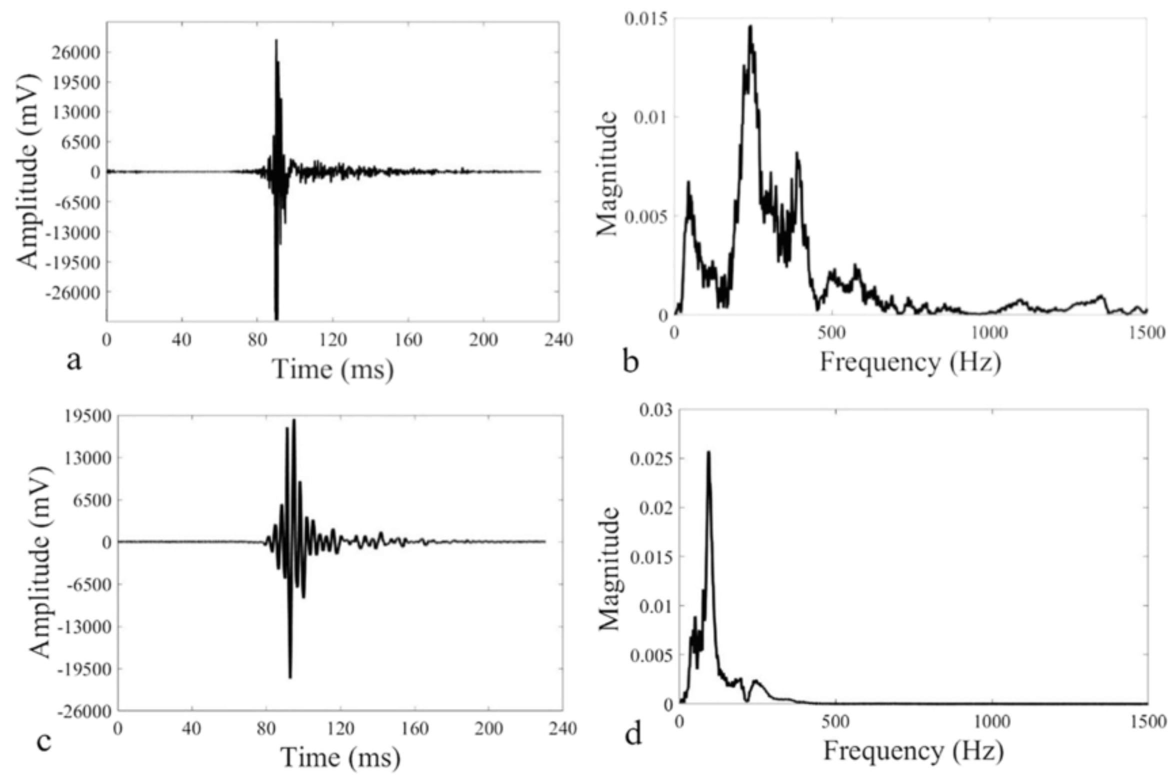


(each can be implemented in analog or digital!)

Spectral Analysis: The Frequency Domain

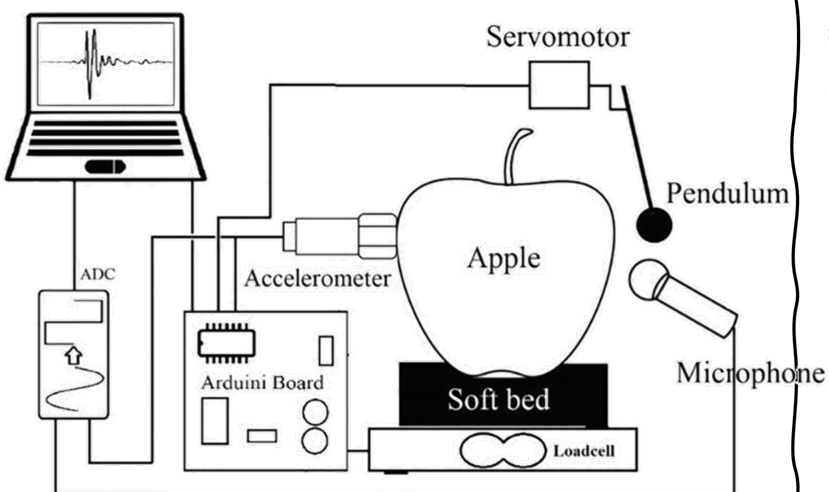


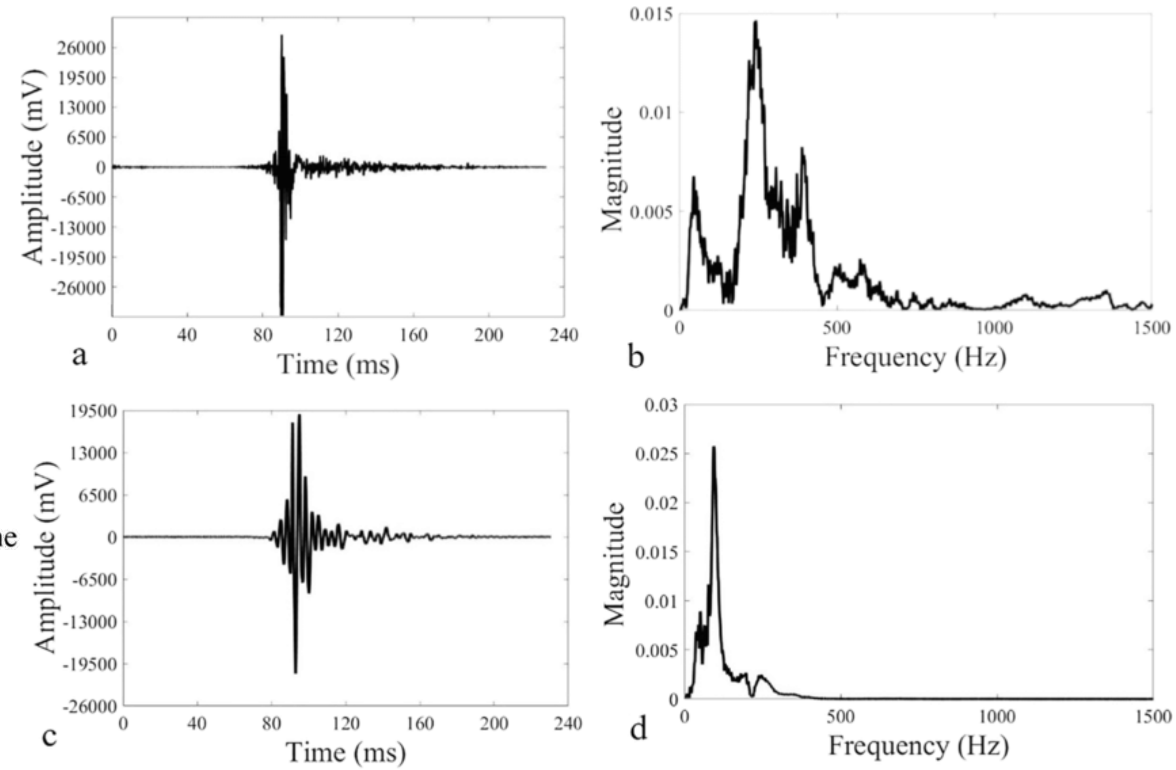
Bird chirping



Can you guess?

Spectral Analysis: The Frequency Domain





Apple Testing

Spectral Analysis: The Fourier Transform

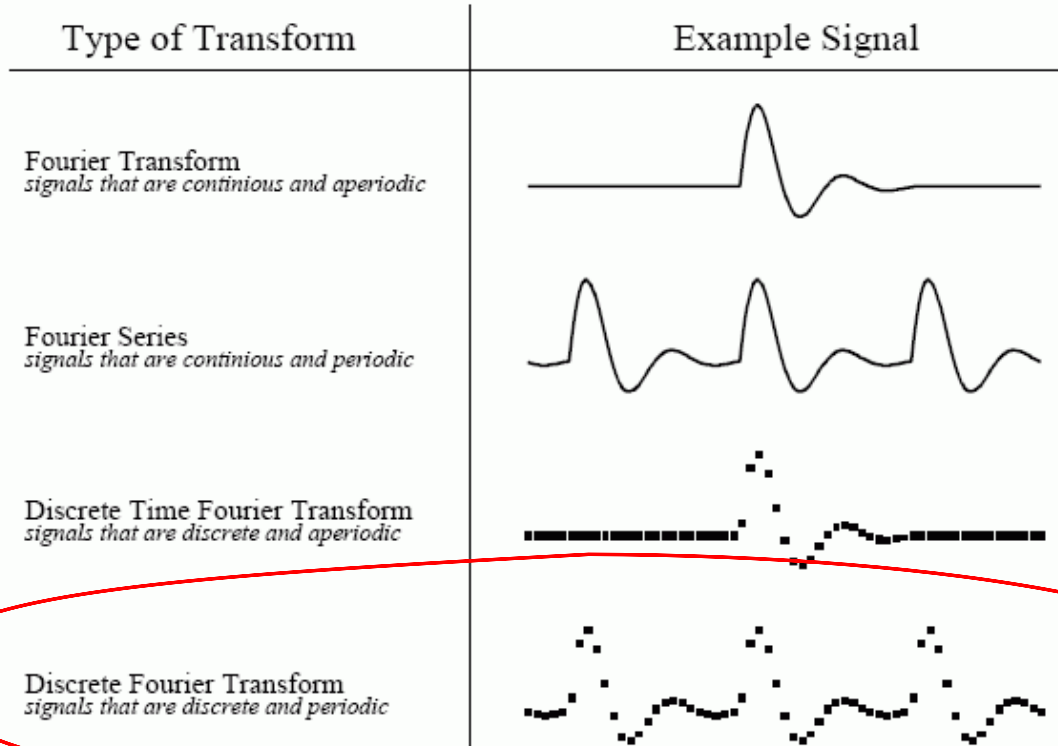
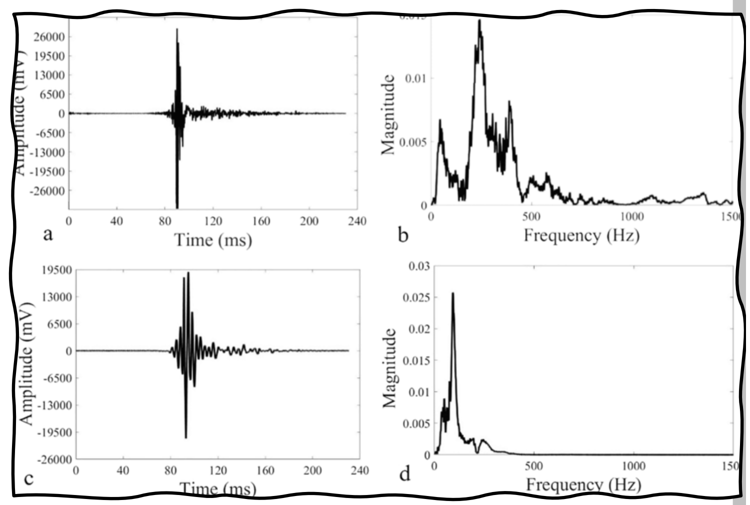
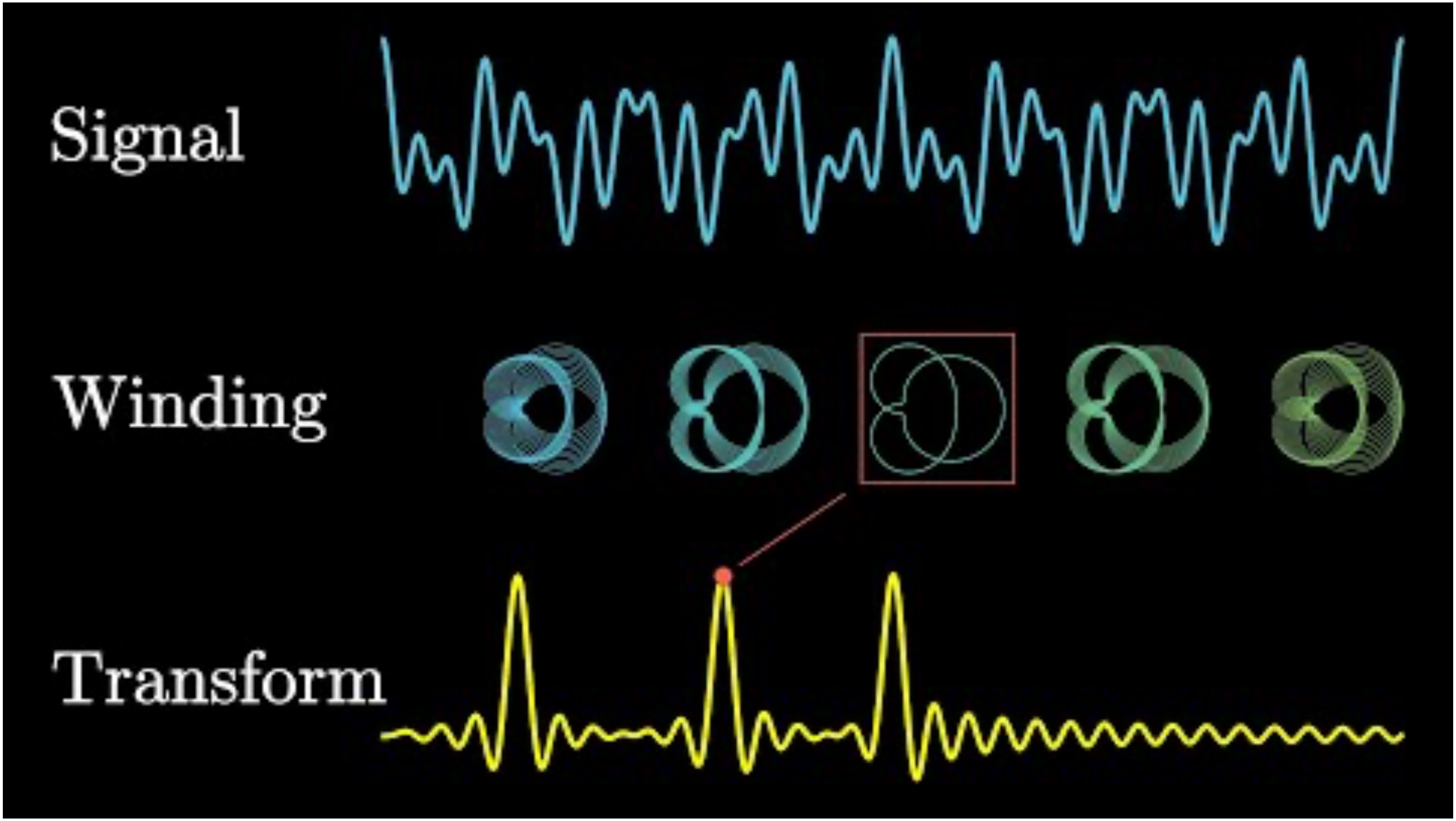
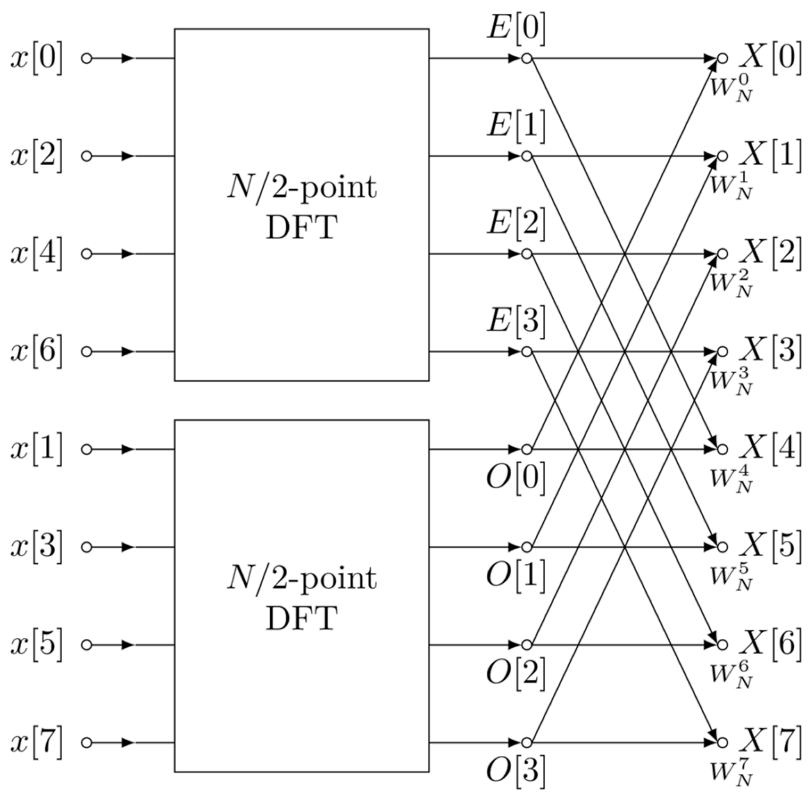


FIGURE 8-2
Illustration of the four Fourier transforms. A signal may be continuous or discrete, and it may be periodic or aperiodic. Together these define four possible combinations, each having its own version of the Fourier transform. The names are not well organized; simply memorize them.




Signal
Winding
Transform

Spectral Analysis: The FFT



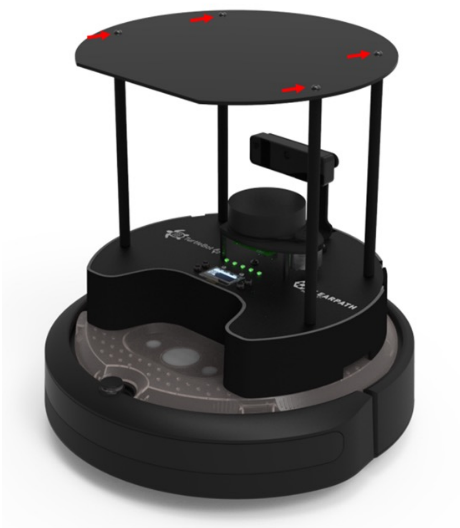
The "FFT" is a way of quickly solving the DFT with a computer.

The details of implementation are not necessary to understand*, so long as you understand Fourier analysis at a high level.

*for most applications

Popping out and applying this to robotics

Distance measurements are noisy (e.g., reflections, ambient) and need amplification;
IMU measurements are noisy (e.g., motor vibrations, people walking);
Microphone measurements pick up background noise;
Contact mechanics are temporospatial complicated, and not all useful;
Wireless communication is only reliable AFTER lots of processing



1. Lecture (~30 minutes)
2. Subsystem check-in (~30 minutes)

Questions?

Lecture 1: sensors → data acquisition
Lecture 2: digital signal processing

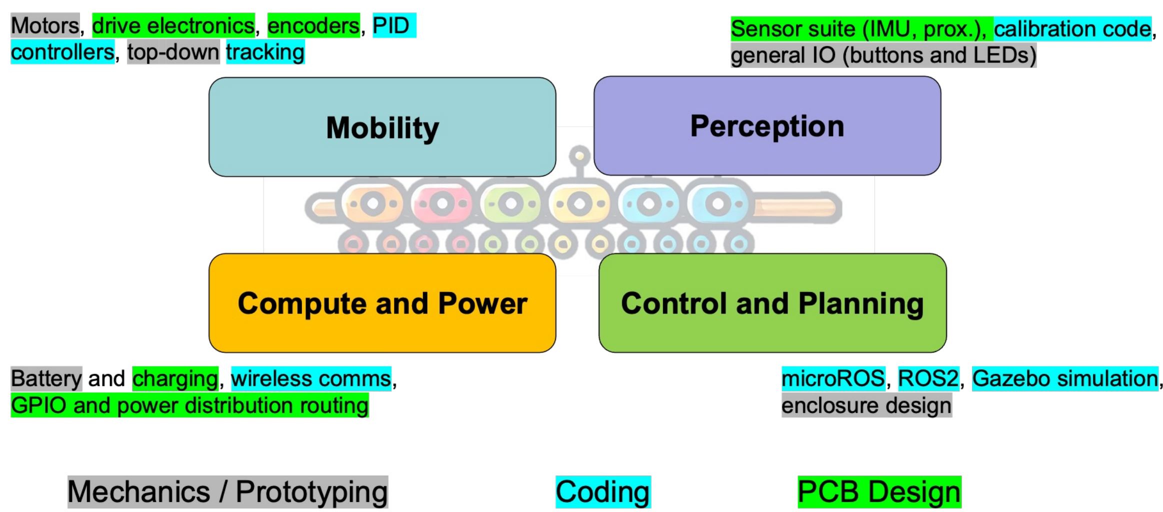
693H Robotic Subsystems

REMINDERS:

-PARTS ORDER GOING IN TONIGHT

-First Design Review Presentations are 2/10, 2/12

- -I will randomly select which two groups present which day. Be ready on 2/10.
- -Link to presentation templates are **posted on Discord + Laulima**



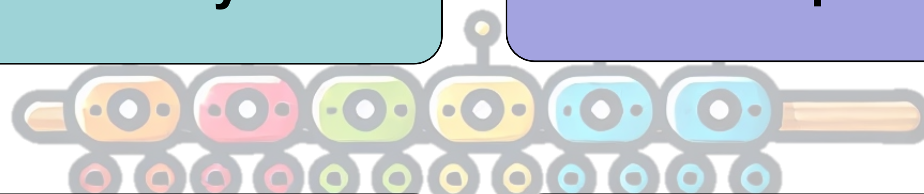
693H Robotic Subsystems

Motors, drive electronics, encoders, PID controllers, top-down tracking

Mobility

Sensor suite (IMU, prox.), calibration code, general IO (buttons and LEDs)

Perception



Compute and Power

Battery and charging, wireless comms, GPIO and power distribution routing

Control and Planning

microROS, ROS2, Gazebo simulation, enclosure design

Mechanics / Prototyping

Coding

PCB Design