Applications of Signals and Systems

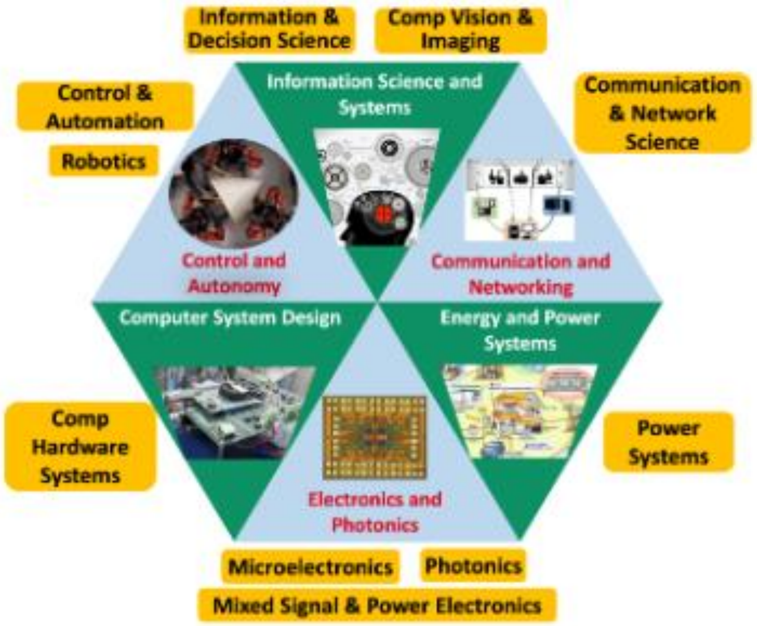
Prof. Alex Patterson
Electrical, Computer, and Systems Engineering Department

Rensselaer

Where do the Labs in ECSE 1010 Fit?

- Artificial Intelligence and Machine Learning
- Computer Vision and Image Processing **Lab03**
- Communications and Computer Networks **Lab03**
- Control Systems **Lab02** **Lab03**
- Robotics and Automation **Lab03**
- Computer Hardware Systems **Lab01**
- Electric Power and Energy **Lab01**
- Microelectronics and Photonics **Lab02***
- Mixed Signal Electronics **Lab01** **Lab02**

https://ecse.rpi.edu/academics/undergraduate-programs



Lab01 – Basics of Electric Circuits

Lab02 – Linear Systems and Op-Amps

Lab03 – Signals (Frequency & Time Domain)

***Diode Omega Lab**

Examples of Projects in Signals and Systems

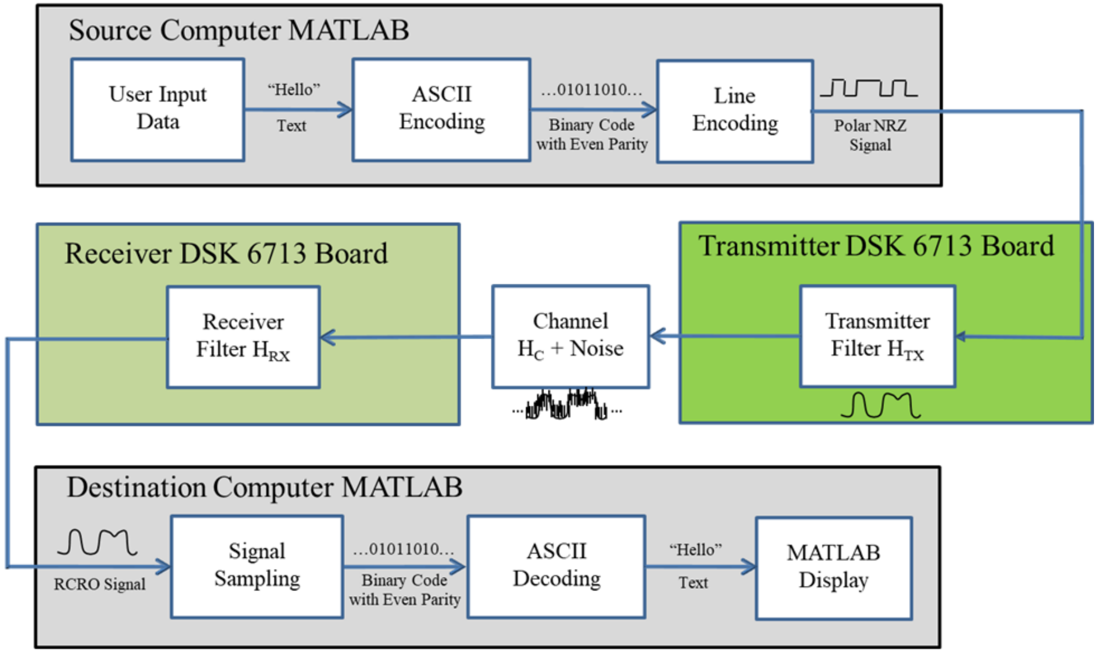
- Although EE and CSE deal more with hardware, they are not limited to building hardware hands-on

- Often the systems that EE's and CSE's deal with are virtual, and are designed on computers in software packages (Matlab, Simulink, Ltspice, Comsol Multiphysics, etc.)

- Instead of physically building these systems to test them, the experiment can actually be a simulation run in one of these programs

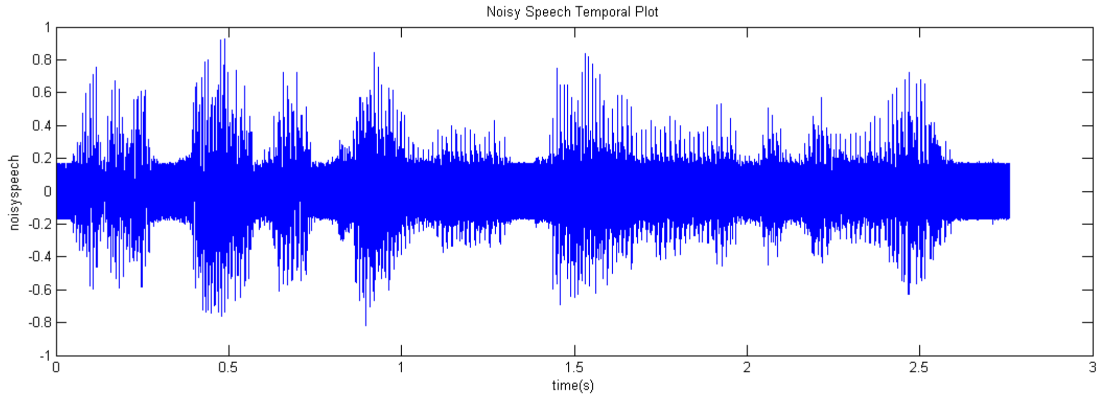
- The fields of **signal processing** (topic of lab 03) and **control systems** are examples of this

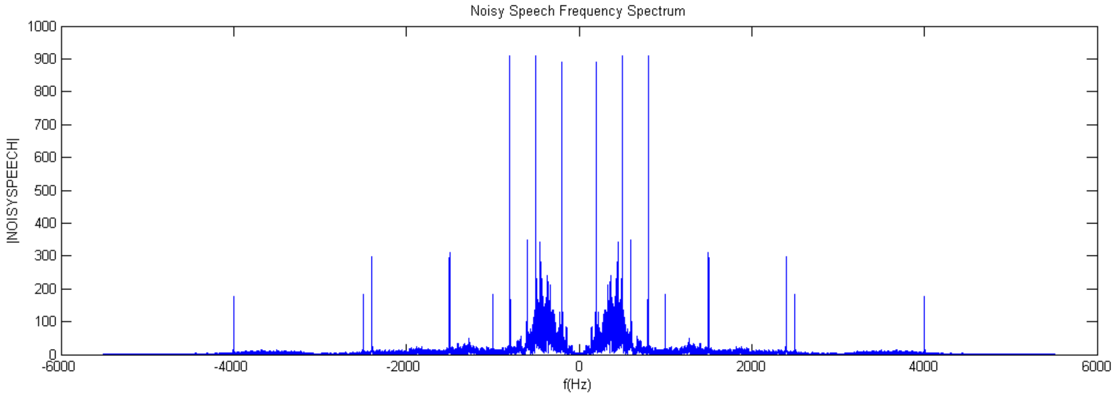
Digital Communication System



Example: Filtering Noisy Signals

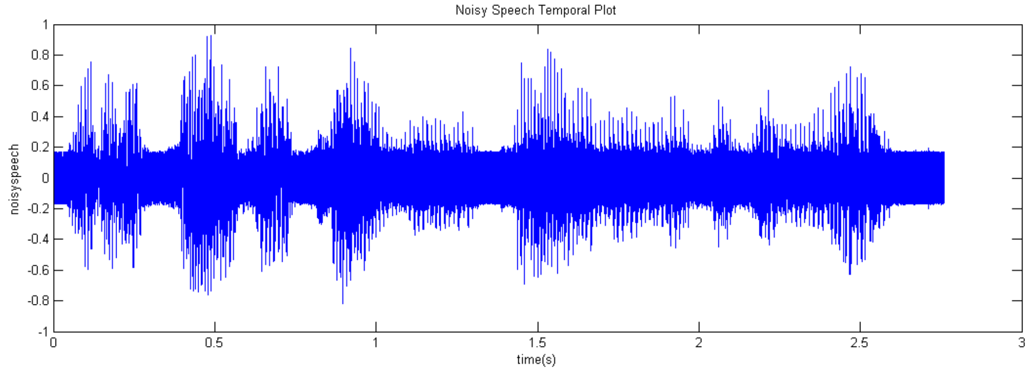
- **Signals** are functions that convey information
- Signals, such as a recording of speech, often contain **noise** that make it more difficult to understand that information
- It is possible to use a representation of a signal in the **frequency domain** to gain a different perspective on the information in the signal
- If we know which frequencies of the signals contain the information, we can use **filters** to keep only those parts of the signal
- The manipulation of signals is called **signal processing**

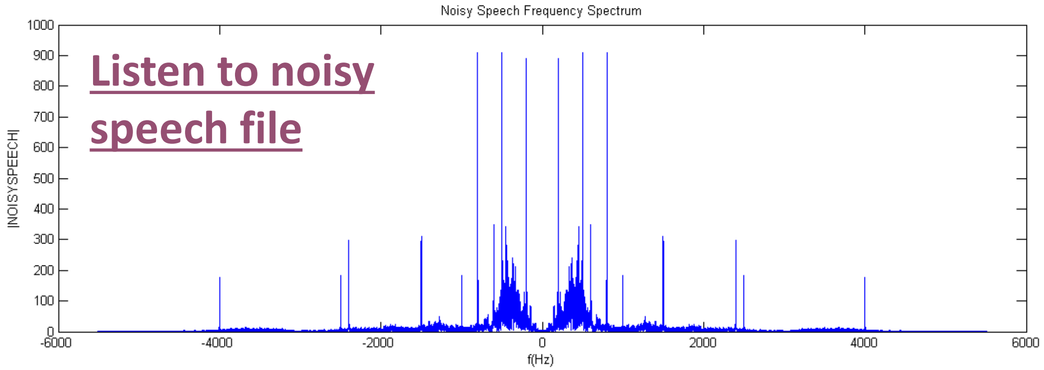




Example: Filtering Noisy Signals

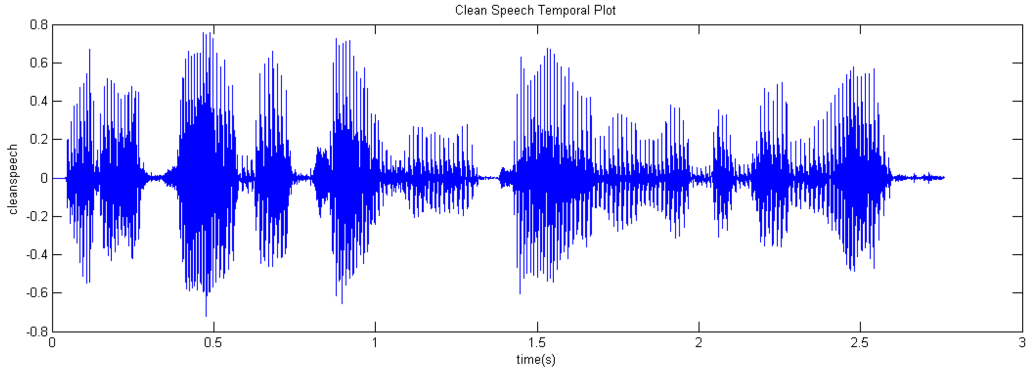
- The top plot shows a noisy signal in the **time domain**; the bottom plot, the **frequency domain**

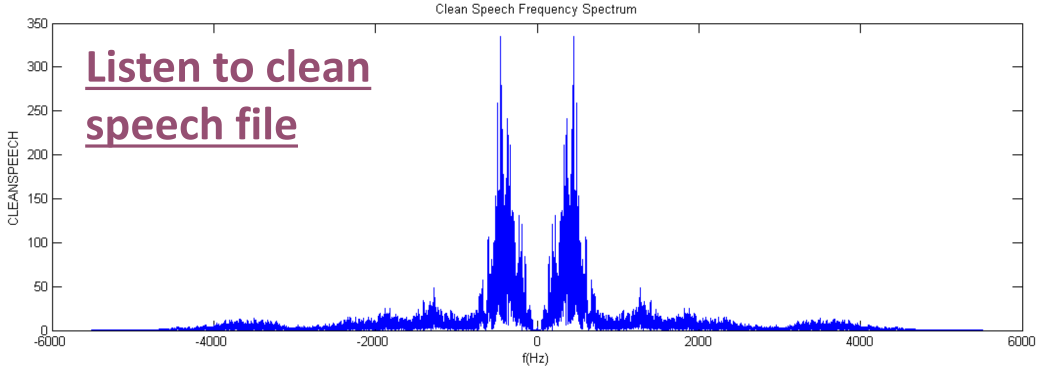




[Listen to noisy speech file]

- The top plot shows the "clean" signal in the **time domain**; the bottom plot, the **frequency domain**





[Listen to clean speech file]

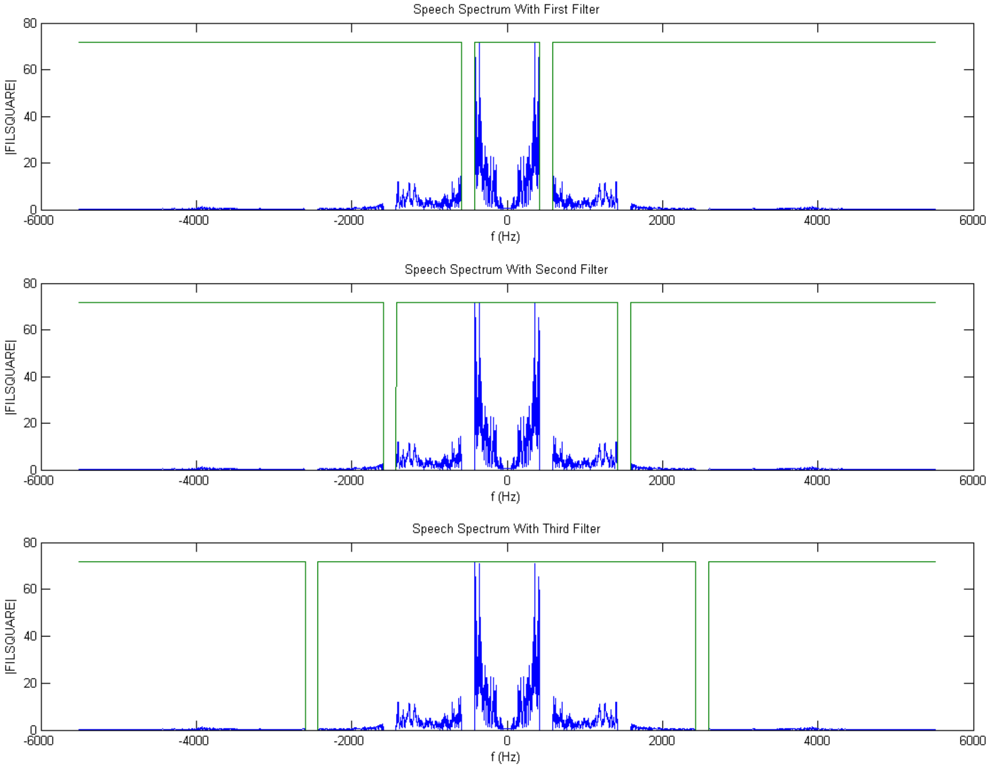
Notice that the frequency spectrum of the noisy signal has large spikes that the clean signal does not

→ These spikes are the noise and the goal is to remove them from the signal

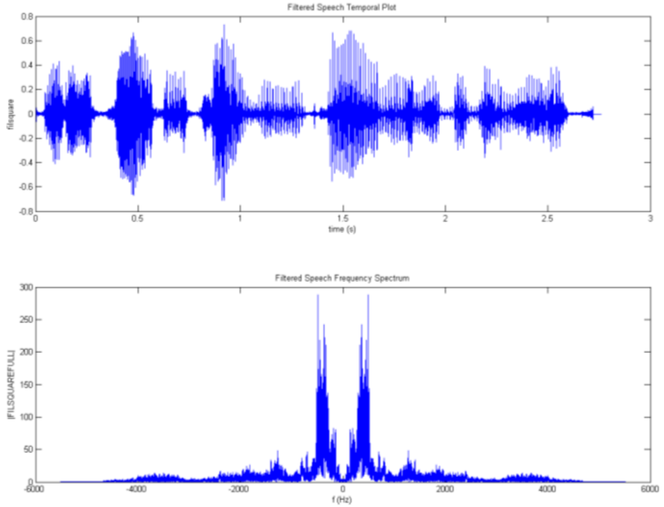
Example: Filtering Noisy Signals

- We can design **filters** (green lines) that eliminate a certain range of frequencies to remove the spikes (noise)
- After filtering, the filtered noisy signal closely resembles the original clean signal

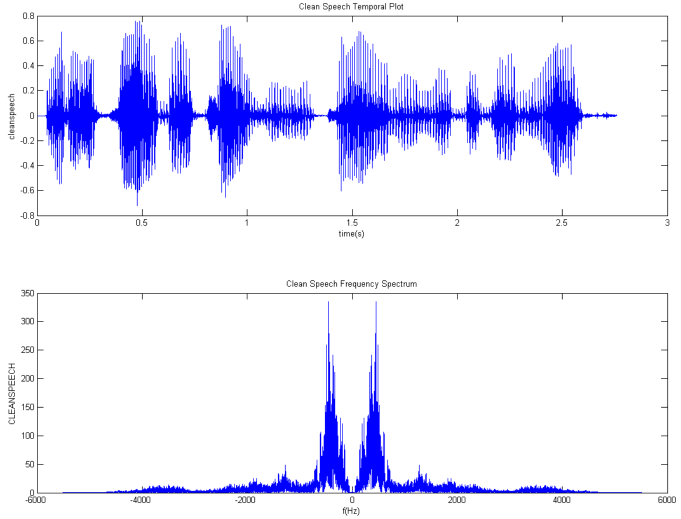
Filters



Filtered Noisy Signal



Original Clean Signal



Listen to filtered speech file

Listen to clean speech file

Example: Digital Communication System

- Signals contain information, but to convey that information, they need to be transported from one place to another: this is the purpose of a communication system. How do they work?

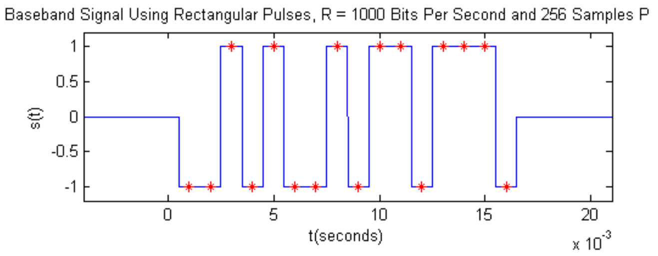
1. User inputs data (text): "hello"

2. User data converted to binary (0's and 1's) via ASCII encoding.

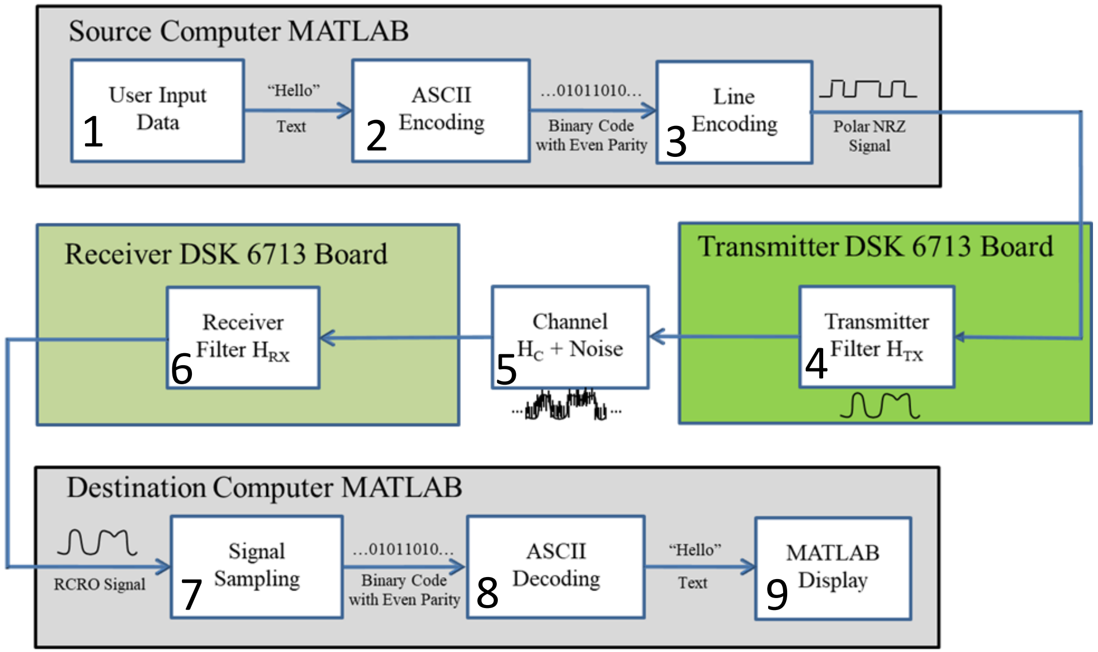
01101000 01100101 01101100 01101100 01101111

H E L L O

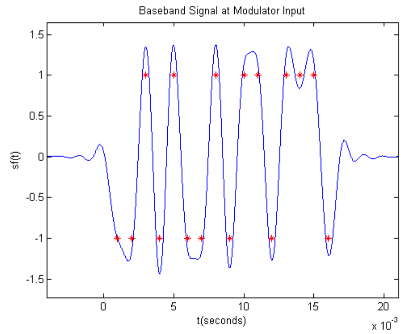
3. Binary data converted to digital voltage signal, such as:



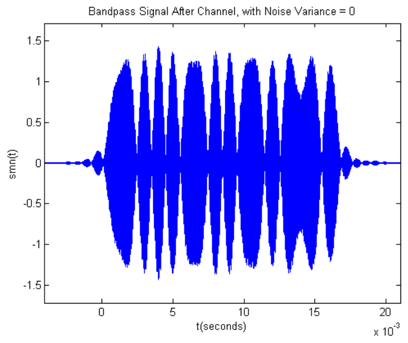
Text-to-Text Digital Communication System



4. Filter the voltage signal to ensure it keeps its shape while being transmitted through wire



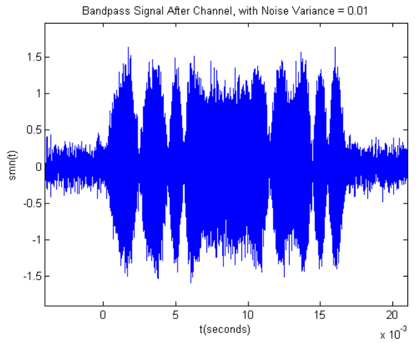
Modulate signal for transmission



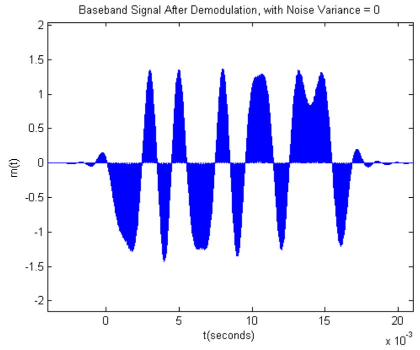
Example: Digital Communication System

- Signals contain information, but to convey that information, they need to be transported from one place to another: this is the purpose of a communication system. How do they work?

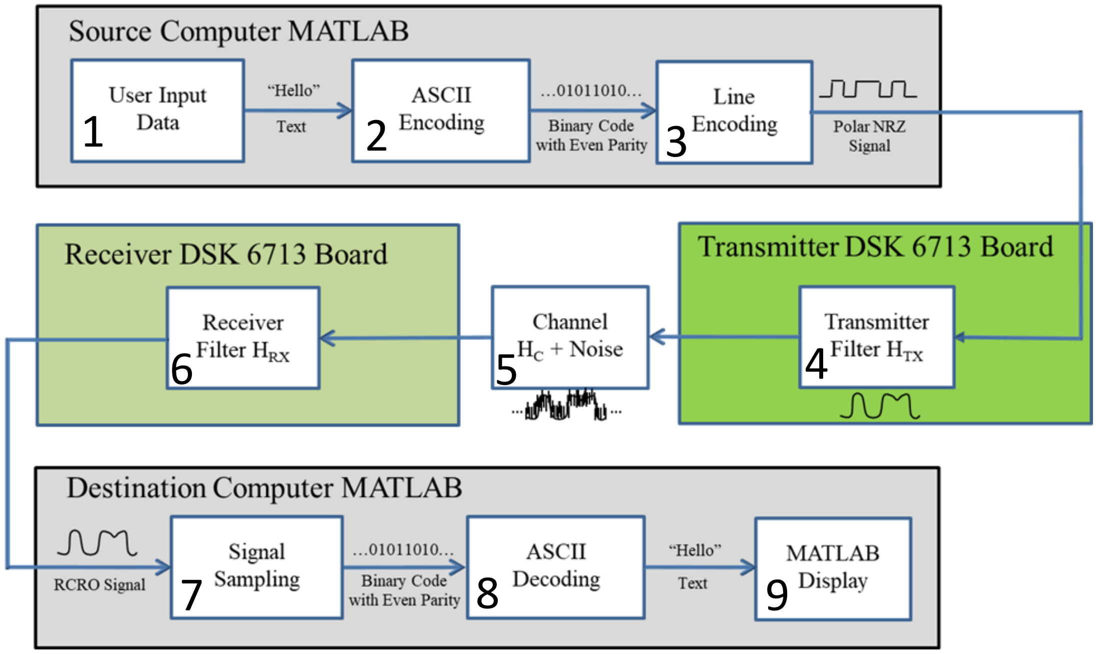
5. Signal is transmitted through a channel, gains noise



6. Receiver filters demodulates signal

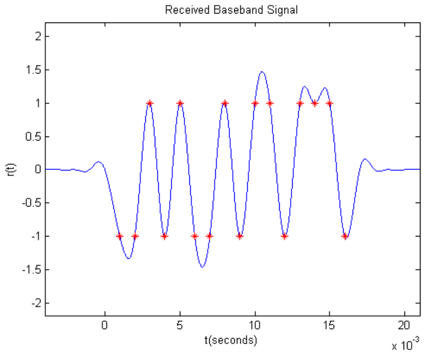


Text-to-Text Digital Communication System



Note: steps 1-4, 6-9 implemented via software!

7. Signal is sampled



8. Sampled values converted to binary

01101000 01100101 01101100 01101100 01101111

9. Binary converted to text via ASCII

H E L L O

10. Message displayed on screen

Note: although all of these examples are done in software (digital signal processing), they are also done in hardware (analog and digital signal processing)

Example: Audio Engineering

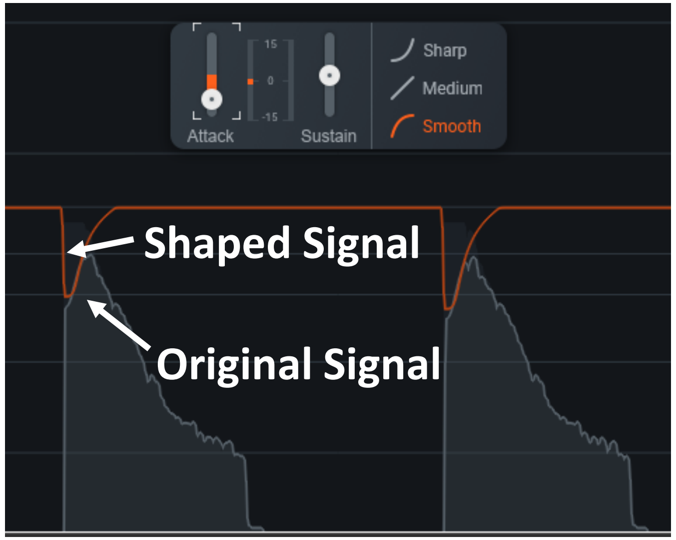
https://www.artplacer.com/essential-art-tools-every-artist-should-have/

- Another area rich in signal processing applications is audio engineering & music production. Think of signal processing as a set of tools for creating auditory art (music). Examples:
  - **Transient shaping**: reduce/enhance an initial sound (attack) or the lingering sound after (sustain)

Original Sample

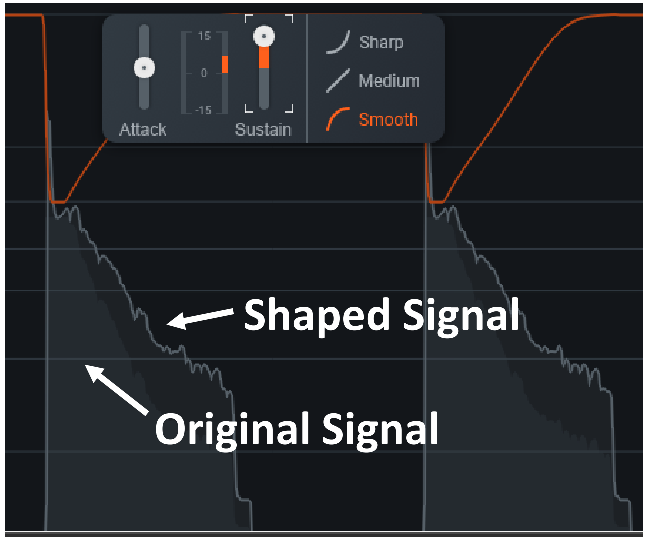
Listen to original snare sample

Reduced Attack



Listen to reduced attack snare sample

Enhanced Sustain



Listen to enhanced sustain snare sample

Example: Audio Engr.: Sound Effects

- **Sound Effects/Simulating Hardware**: changing the character of a signal's sound, by adding effects such as distortion (amplifiers), reverb, delay, chorus, flanger, phaser (guitar pedals), etc.

Listen to Clean Guitar

Digital signal processing methods used to emulate hardware (analog circuits)

Guitar Input

Listen to "Plexi" Guitar

Audio Output

Guitar Input

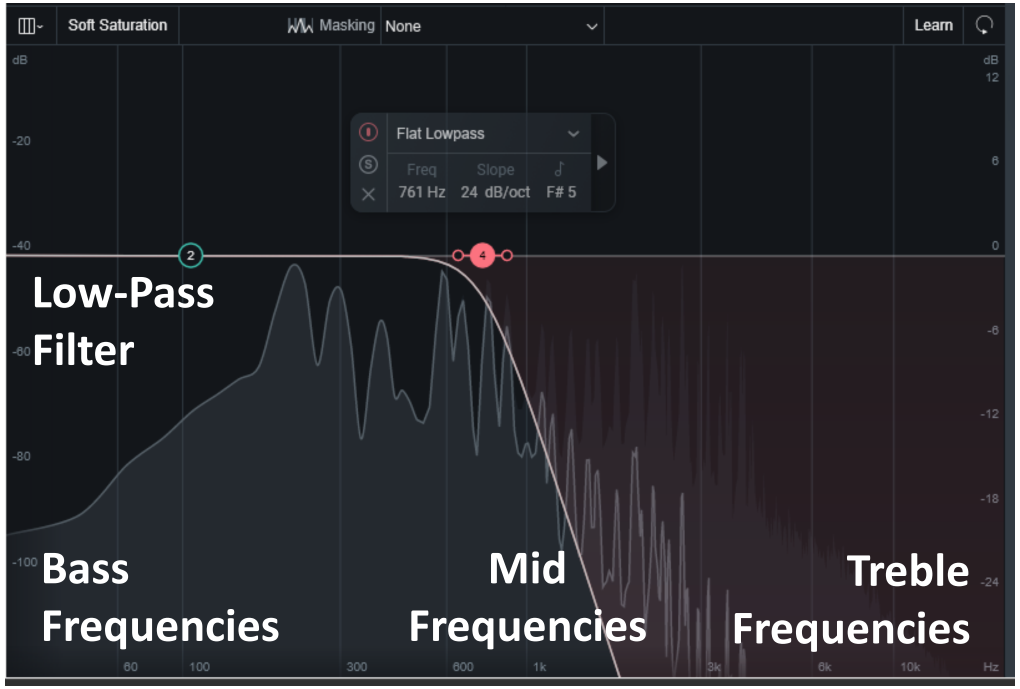
Listen to "Sitar" Guitar

Audio Output

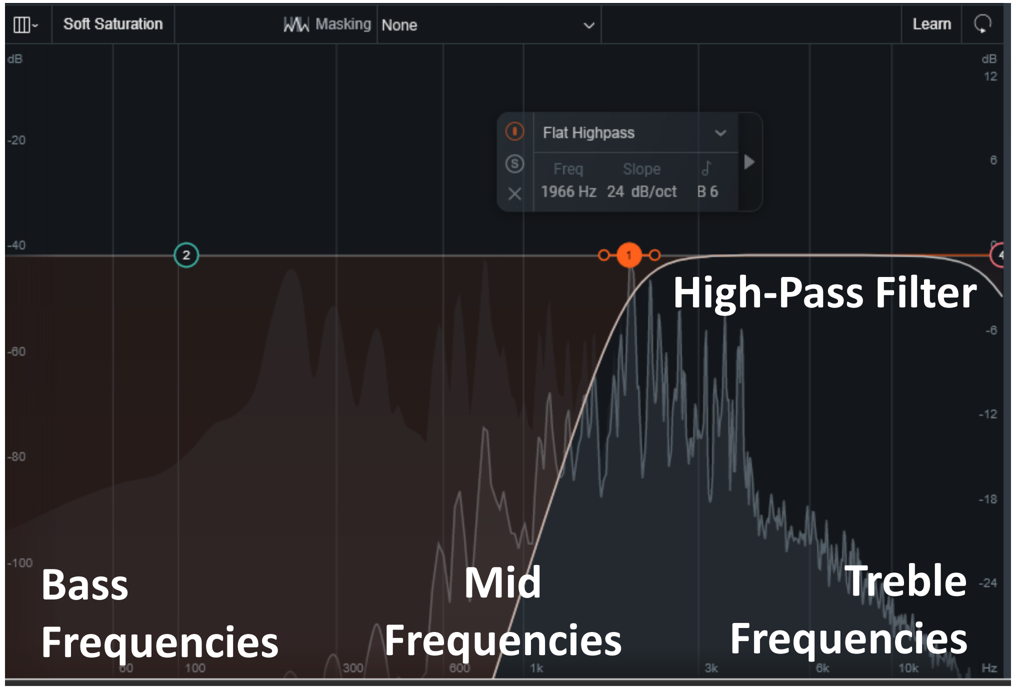
Example: Audio Engr.: Equalization

- **Equalization**: application of filters to reduce/enhance particular ranges of frequency in a signal

Listen to Unfiltered Guitar



Listen to Low-Pass Filtered Guitar (less treble)



Listen to High-Pass Filtered Guitar (less bass)

Example: Audio Engr.: Compression

- **Compression**: makes louder parts of a signal quieter to balance or even out the sound the sound.

- Compression is often a subtle effect and may be difficult to hear, but the compressed signal should sound like the volume level is more even throughout, whereas without compression it seems to get noticeably louder and softer on each strum

Listen to Guitar without Compression   Listen to Compressed Guitar

Example: Audio Engr.: Synthesizers

- With knowledge of manipulating signals, we not only can change sounds but also create (or synthesize) new ones

- Here are a few examples using familiar signals as the fundamental component of the sound

  [Listen to Sine Wave Synth]

  [Listen to Triangle Wave Synth]

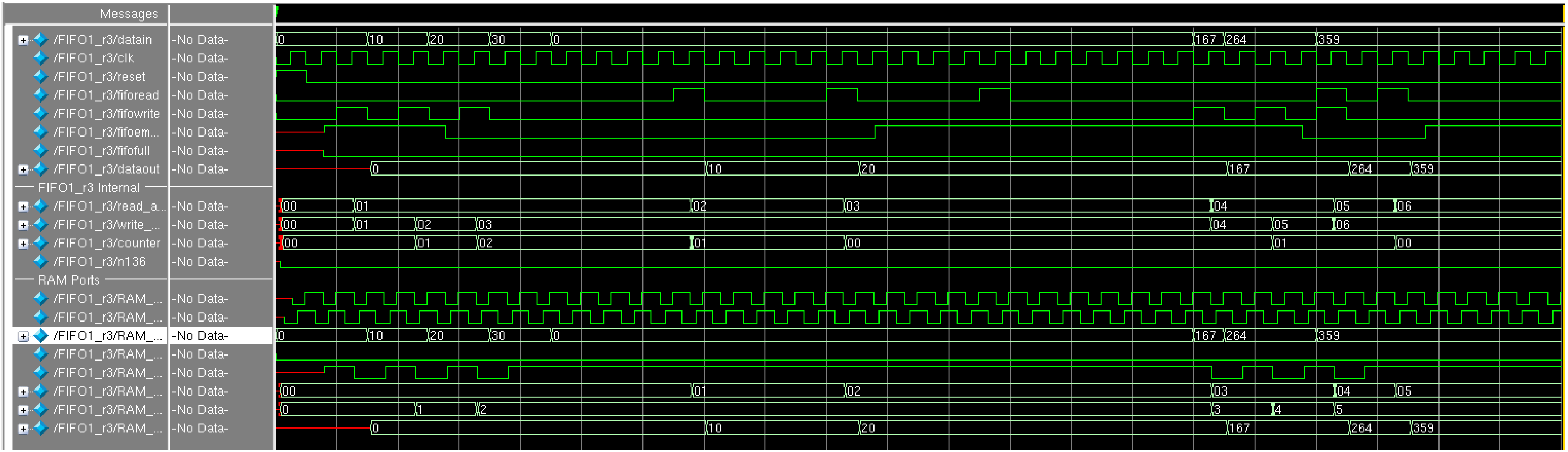
  [Listen to Square Wave Synth]

- Since additional audio processing techniques can be applied to the synthesizer signal, the possibilities are endless

  [Listen to Frequency Modulated (FM) Synth]

Example: VLSI Design (Designing Computers)

- Digital circuits in computers make decisions at the frequency of a clock signal
- Each 0 and 1 (information) is represented as a low or high voltage signal
- These signals must be sent from one component to another 1) at the correct time, and 2) with the correct shape & value. Designing such systems also requires a knowledge of signals.



Example: Cruise Control System

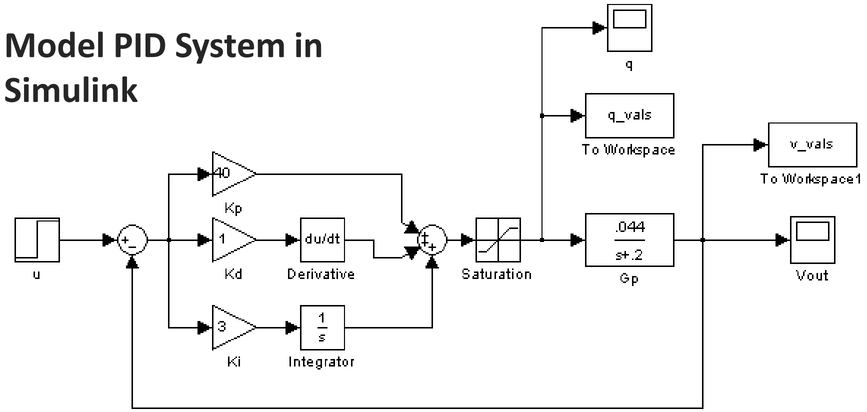
- The cruise control system of a car is an example of a linear control system

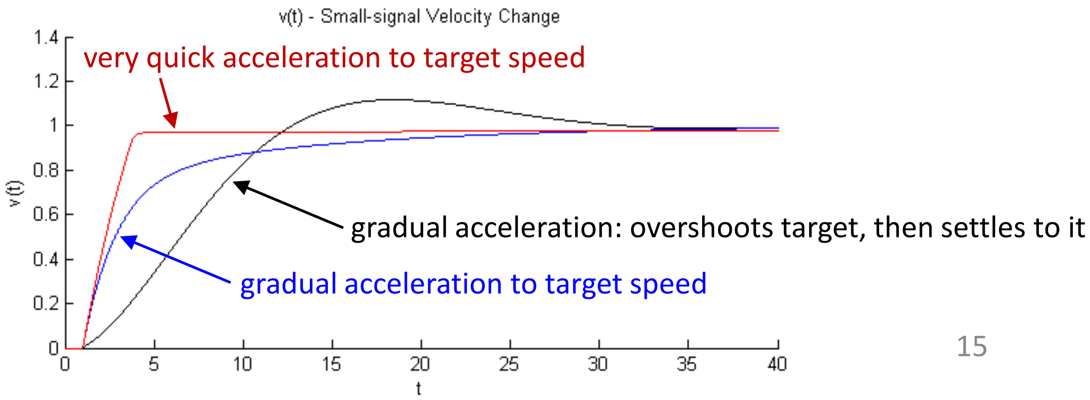
- The system takes a new target speed as input and must bring the car to that speed as quickly and smoothly as possible

- The system uses the following feedback to determine how to adjust the car's speed (PID control):
  - Difference between the current speed and target speed (the *error*)
  - Integral of the *error*
  - Derivative of the *error*

- Adjusting feedback parameters produces different velocity curves for reaching target speed

Model PID System in Simulink





So? Why care about Signals (and Systems)?

- Signals are everywhere and are the basis of **information**
- Systems can **process information to make decisions**
- **Linear systems** (remember Lab02?) are particularly convenient to work with and you will encounter them often
- Signals and systems represents the “other” side of ECSE: the (mostly) hands-off design of systems at a higher level AND the processing of information
- If you like building circuits, don’t worry – the **physical implementations** of these systems are **analog** or **digital circuits**