Intro to ECSE Lecture Notes: Applications of Signals and Systems

11/10/2022

Section 02, Prof. Patterson

How are EE, CSE, and CS different?

https://ecse.rpi.edu/academics/undergraduate-programs/what-are-ee-and-cse

- Where do EE and CSE overlap?

The two majors have many similarities. In either one, you'll get a solid foundation in applied math and physics, computer programming, circuit theory and electronics, engineering design, and professional development. You can take a look at our curriculum and templates for more details.

- How is EE different than CSE?

EEs generally focus more on hardware and physics. This is where you'll learn about machines, devices, and systems that consume a lot of power, and how that power is managed. This is also where you'll build circuits and systems at a small scale, like designing and fabricating your own microchips and sensors.

- How is CSE different than EE?

CSEs generally focus more on algorithms and systems. Compared to an EE, you'll do more computer programming and learn more mathematical theory related to areas like artificial intelligence. This is where you can learn about the Internet of Things, computer vision, communication networks, and robotics.

How are EE, CSE, and CS different?

- Can I do both?

Yes! Many students comfortably get an EE/CSE dual major where you'll get a broad education in both areas. You'll have a lot of freedom to choose electives in your senior year that focus on what you want to learn more about.

- How is CSE different than Computer Science?

A simple answer is that Computer Science is more focused on what computers can do, and Computer Systems Engineering is more focused on how computers are built. You can get a CS degree without taking your hands off the keyboard, but in CSE you will get them dirty with all sorts of sensors and devices! You'll learn a lot of ways of applying math to model the real world in the CSE major that you won't see in a CS degree.

- Can I do both?

Yes! It's extremely common for students to dual major in CSE and CS to get exposed to both ways of thinking.

Examples of Projects in Signals and Systems

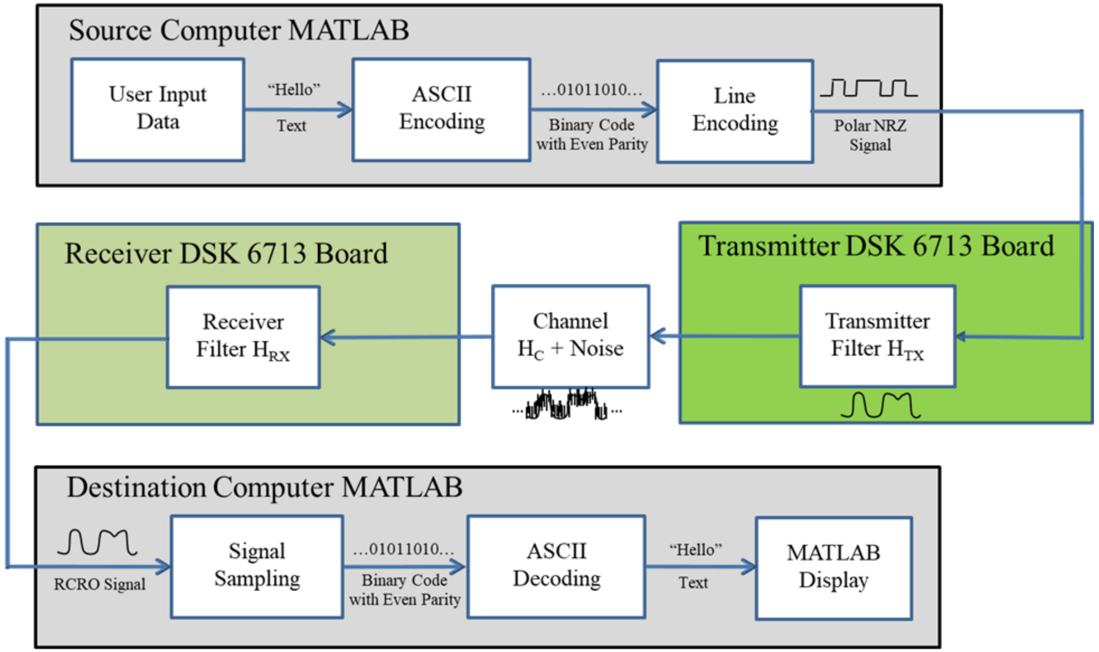
- Although EE and CSE deal more with hardware, they are not limited to building hardware hands-on

- 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

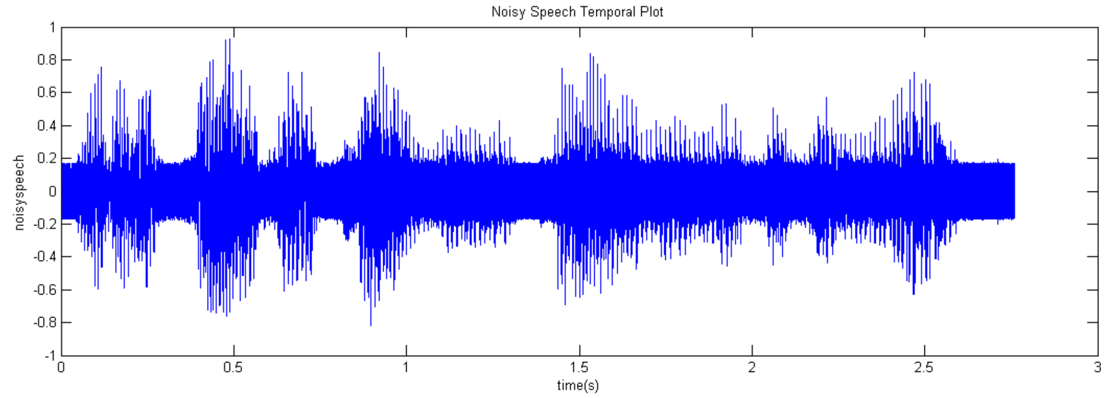
- The fields of signal processing (topic of lab 03) and control systems are examples of this

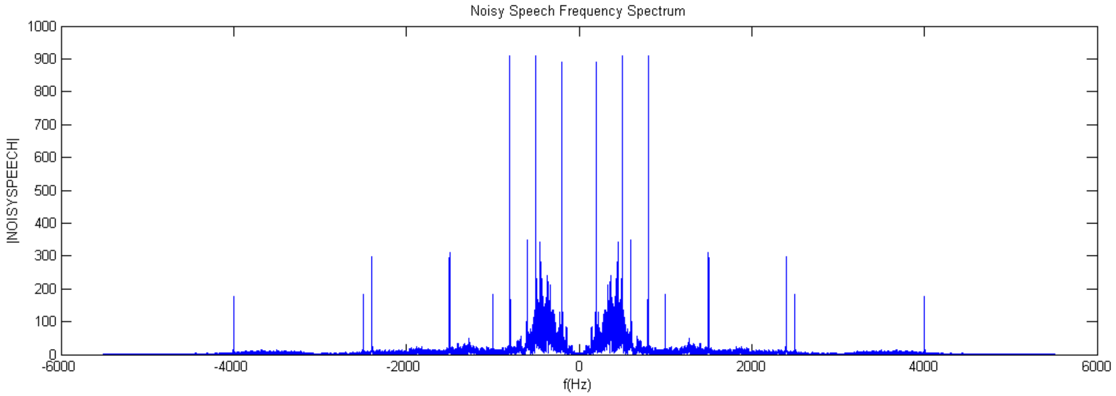
Digital Communication System



Example: Filtering Noisy Signals

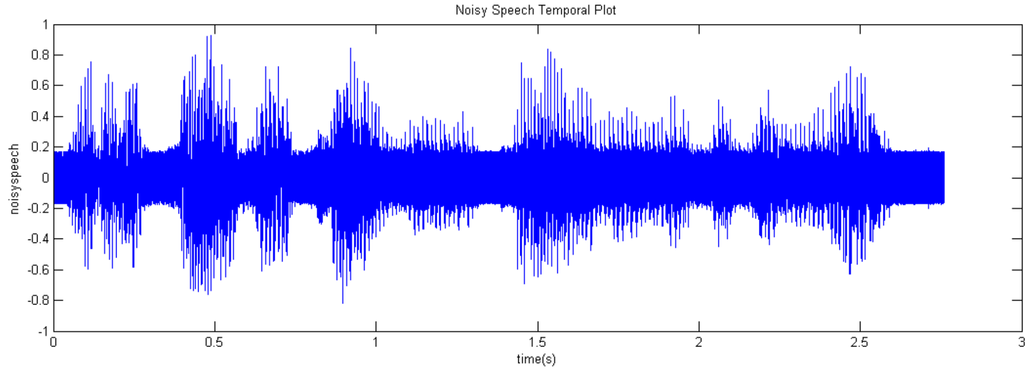
- Signals are functions that convey information
- Signals, such as a recording of speech, often contain noise that make it more difficult to understand that information
- It is possible to use a representation of a signal in the frequency domain to gain a different perspective on the information in the 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

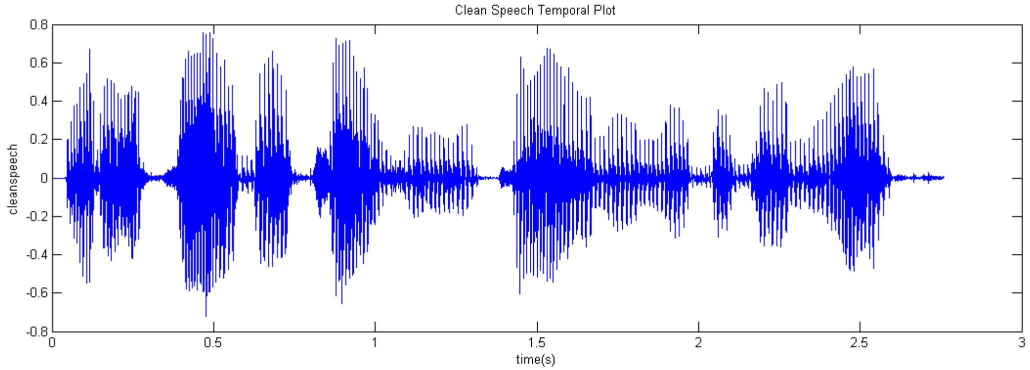


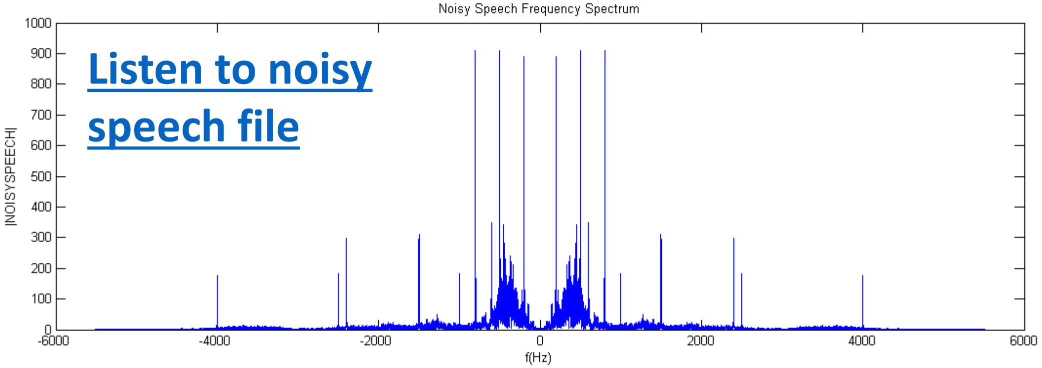


Example: Filtering Noisy Signals

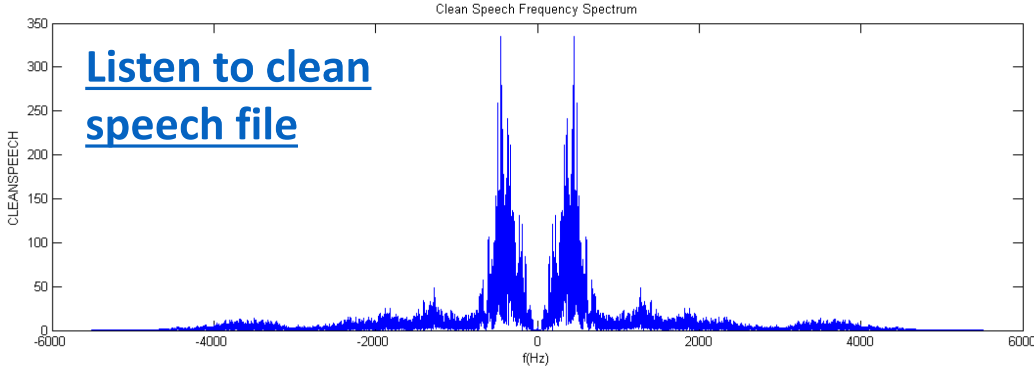
- The top plot shows a noisy signal in the time domain; the bottom plot, the frequency domain
- The top plot shows the “clean” signal in the time domain; the bottom plot, the frequency domain







Listen to noisy speech file



Listen to clean speech file

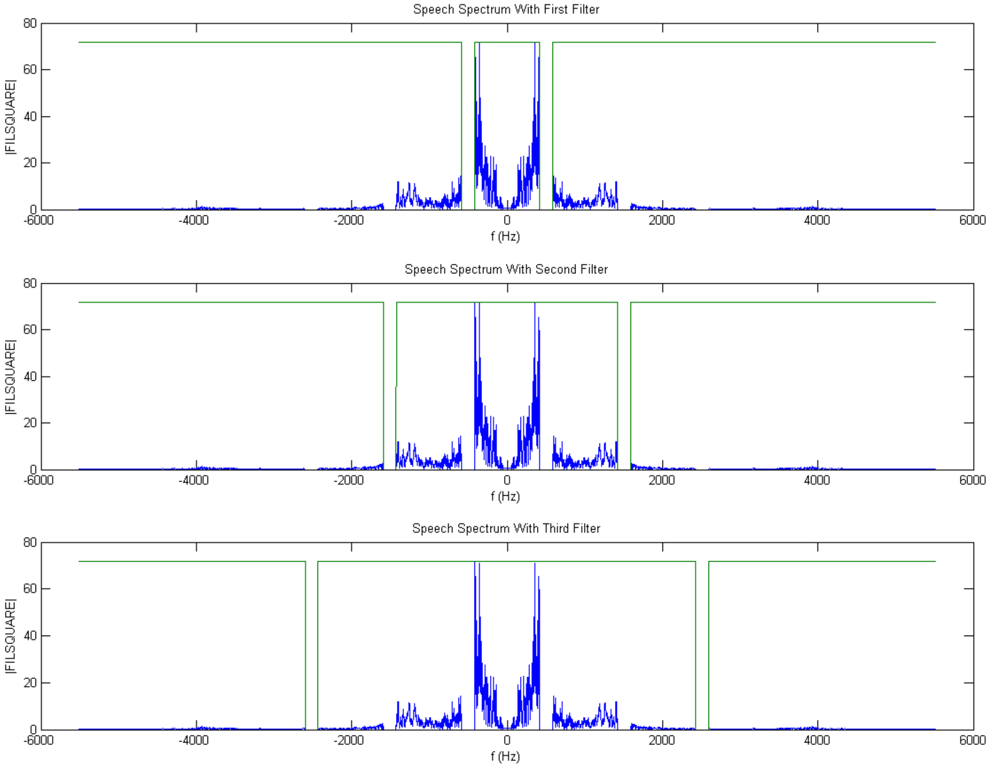
Notice that the frequency spectrum of the noisy signal has large spikes that the clean signal does not

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

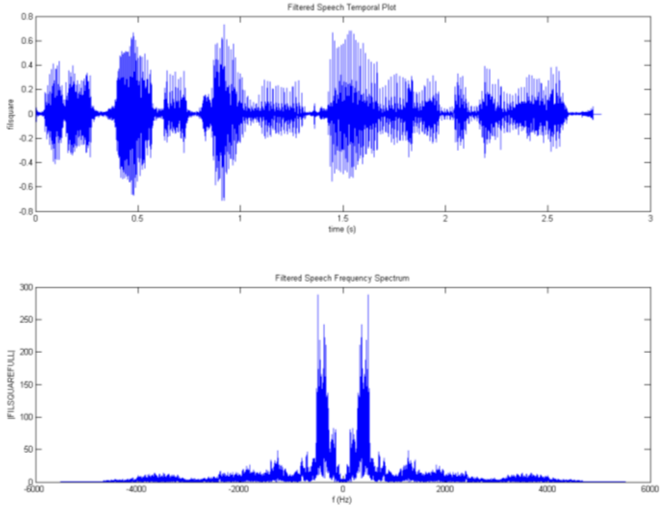
Example: Filtering Noisy Signals

- 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

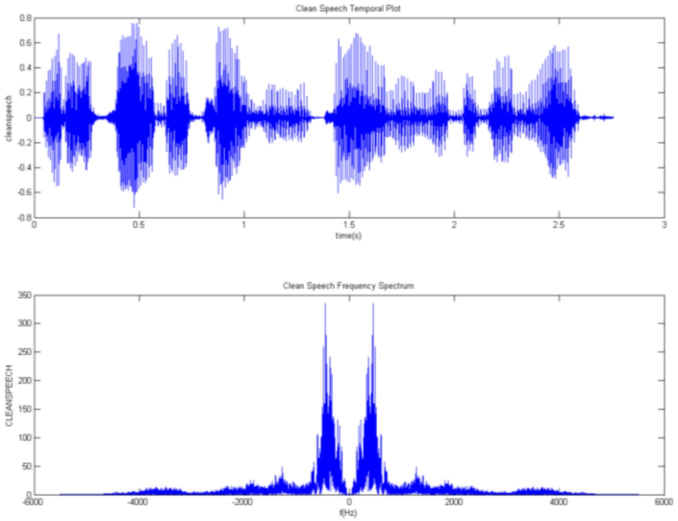
Filters



Filtered Noisy Signal



Original Clean Signal



[Listen to filtered speech file]

[Listen to clean speech file]

Example: Digital Communication System

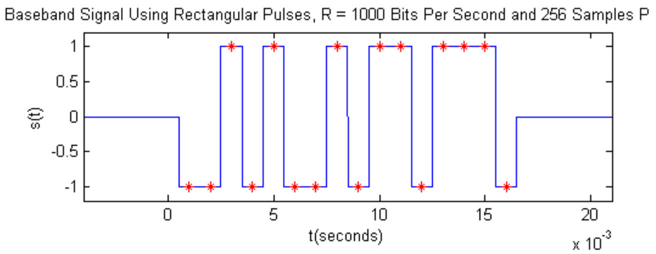
- Signals contain information, but to convey that information, they need to be transported from one place to another: this is the purpose of a communication system. How do they work?

1. User inputs data (text): “hello”

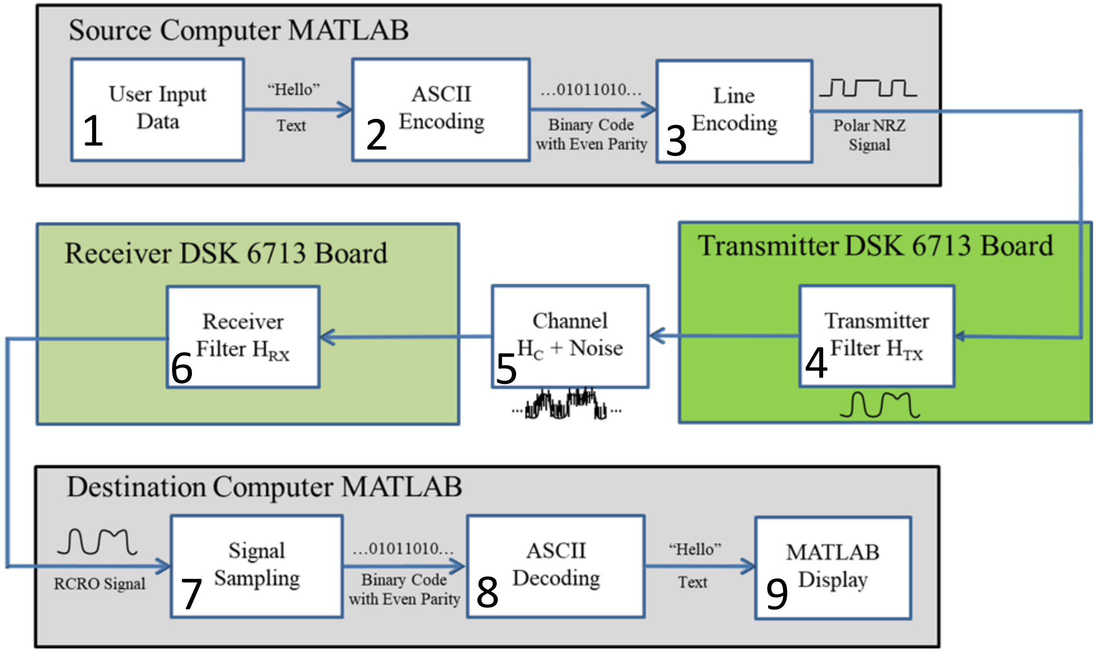
2. User data converted to binary (0’s and 1’s) via ASCII encoding.

01101000	01100101	01101100	01101100	01101111
H	E	L	L	O

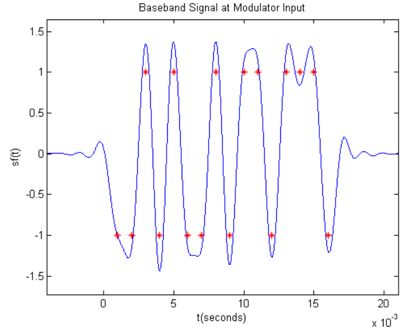
3. Binary data converted to digital voltage signal, such as:



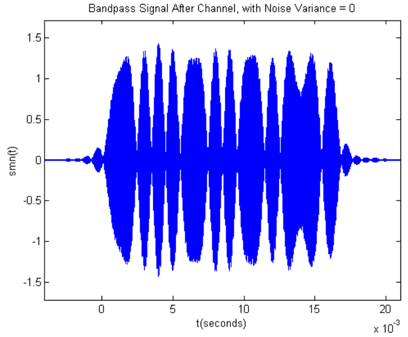
Text-to-Text Digital Communication System



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



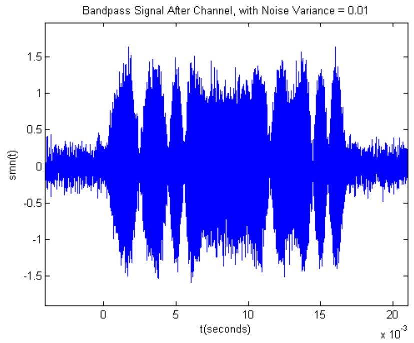
Modulate signal for transmission



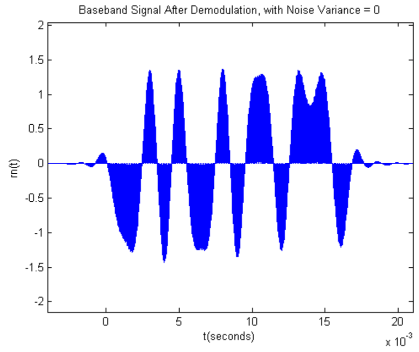
Example: Digital Communication System

- 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?

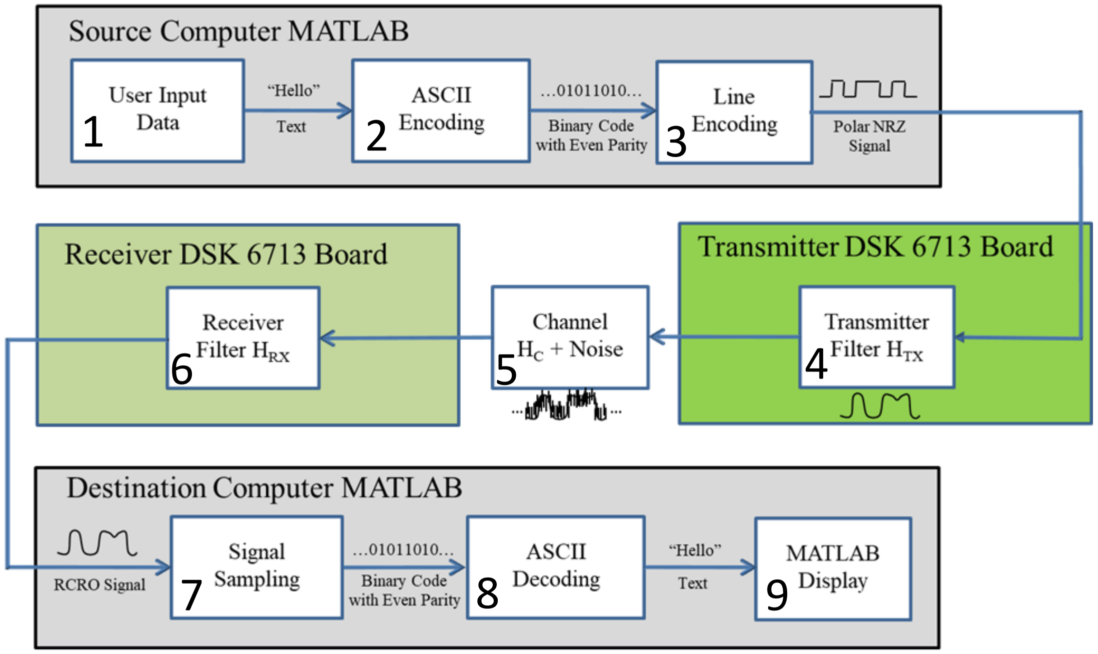
5. Signal is transmitted through a channel, gains noise



6. Receiver filters demodulates signal

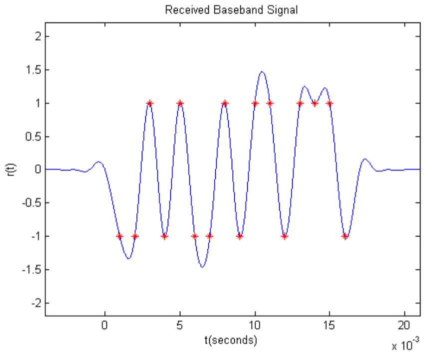


Text-to-Text Digital Communication System



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)

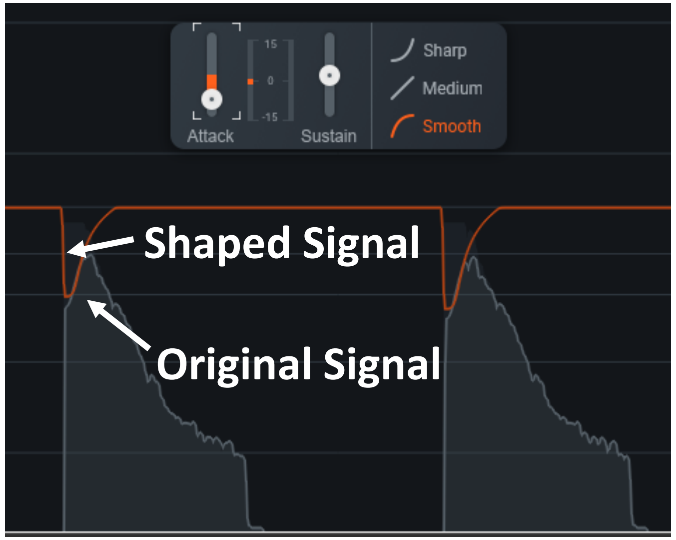
Example: Audio Engineering

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

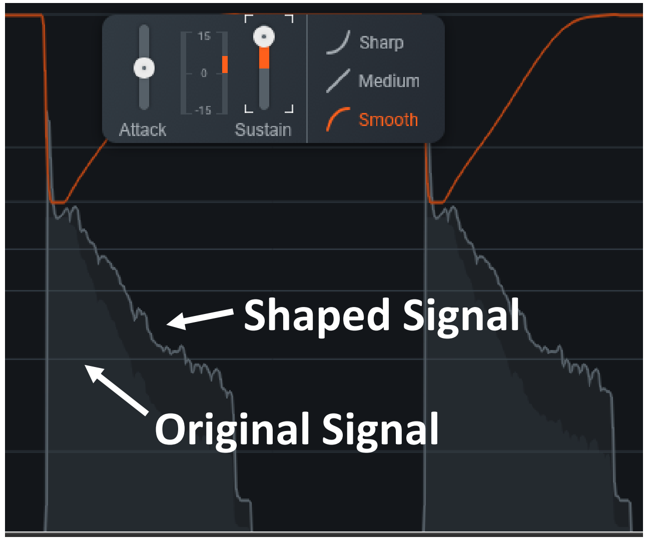
- Another area rich in signal processing applications is audio engineering & music production. Think of signal processing as a set of tools for creating auditory art (music). Examples:
  - **Transient shaping**: reduce/enhance an initial sound (attack) or the lingering sound after (sustain)

Original Sample

Reduced Attack



Enhanced Sustain



Listen to original snare sample

Listen to reduced attack snare sample

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.

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

Listen to Clean Guitar

Guitar Input

Listen to "Plexi" Guitar

Audio Output

Guitar Input

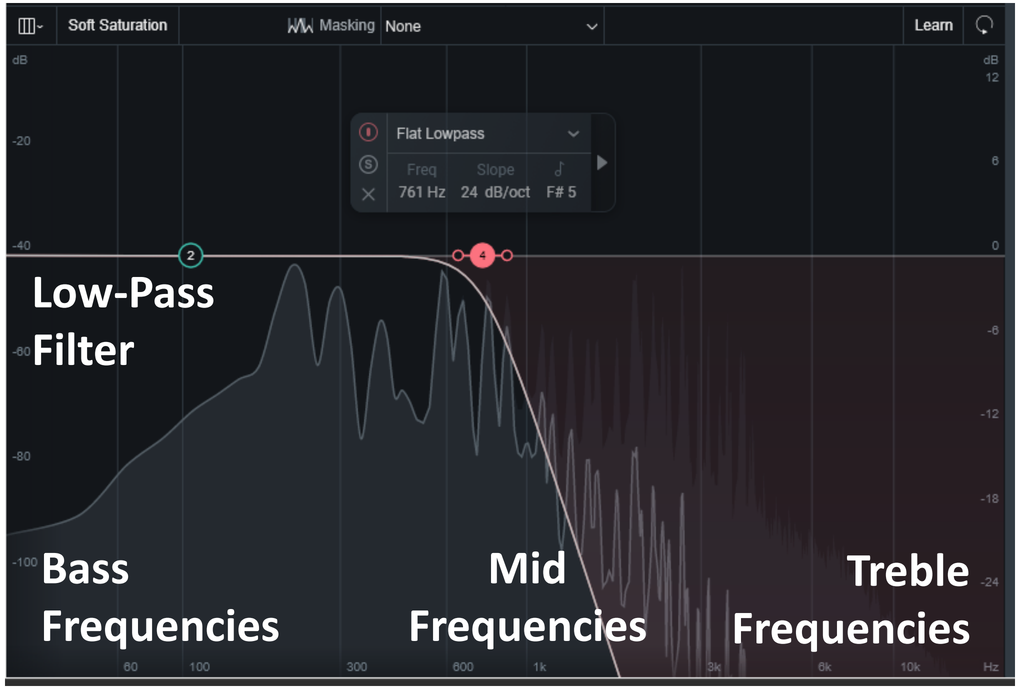
Listen to "Sitar" Guitar

Audio Output

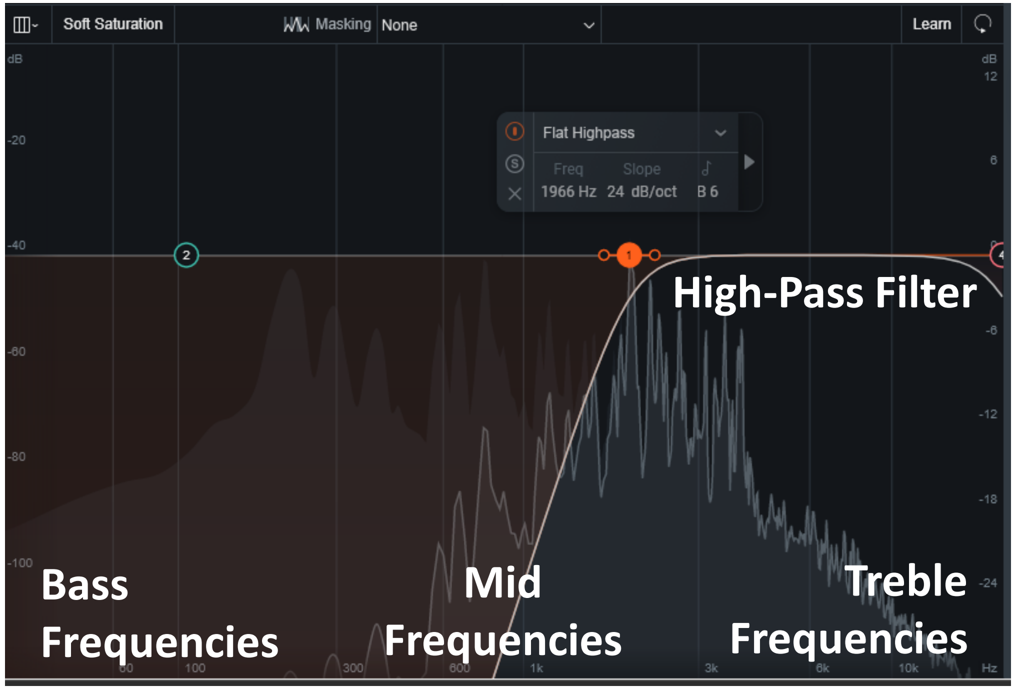
Example: Audio Engr.: Equalization

- **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]

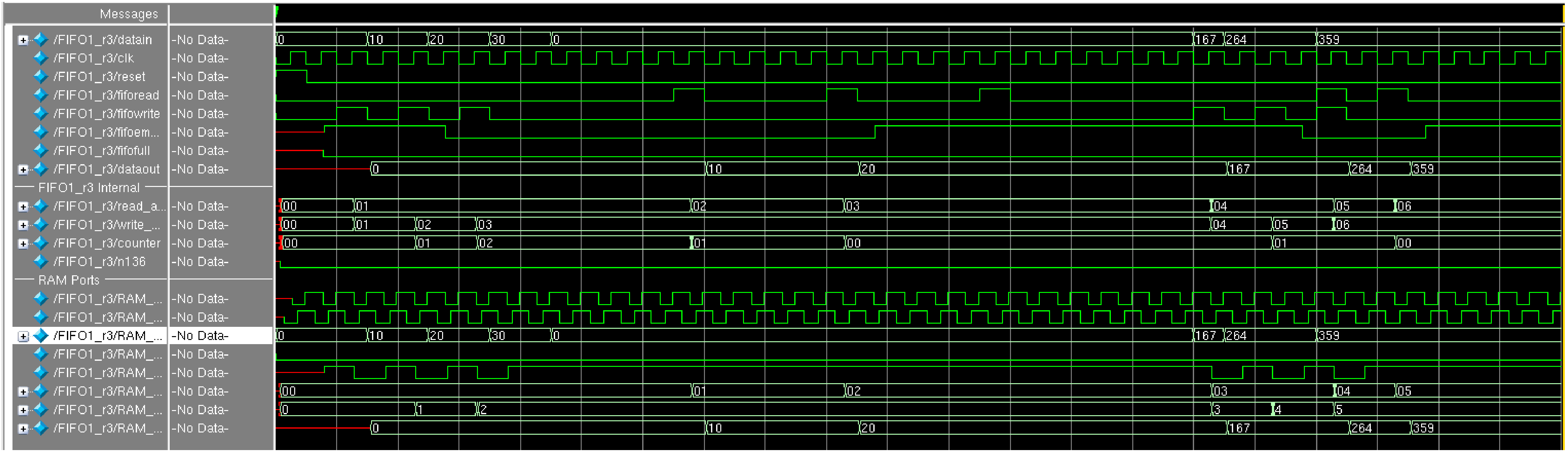
  [Listen to Square Wave Synth]

- 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