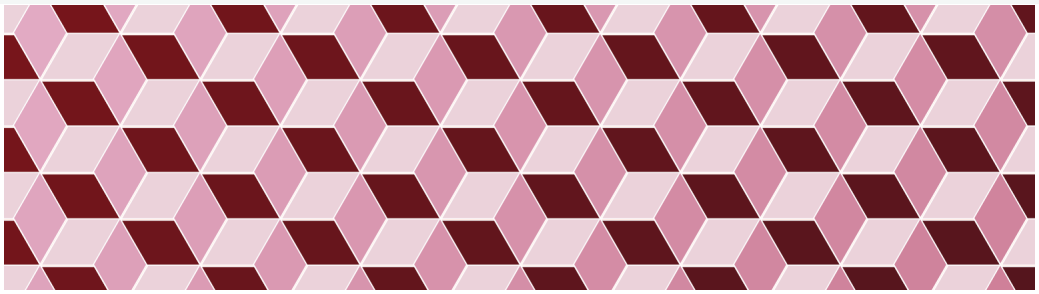


15-104 Introduction to Computing for Creative Practice

Fall 2021

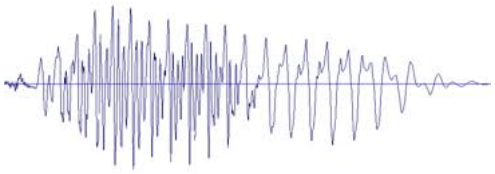
23 Sound Basics

Instructor: Tom Cortina, tcortina@cs.cmu.edu, GHC 4117, 412-268-3514

1

Sound

- Audio signals are essentially vibrations that travel through the air (and other materials).
- We characterize sound as rapid but measurable changes in pressure over time.
- The following image shows a graph of pressure vs. time, often called a "waveform":

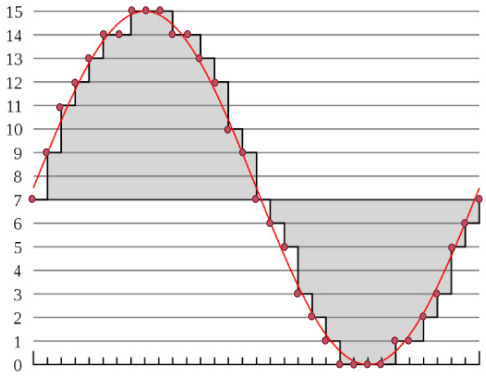


2

Digital Sound

- We represent audio waveforms digitally by measuring the amplitude or pressure of the waveform many times per second.
- The continuous (red) waveform is measured many times per second and rounded to an integer value from 0 to 15 (4 bits).

From www.jaguaraudiodesign.com



Each digitized sample of audio is assigned a value that corresponds to the amplitude of the analog wave.

3

Digital Audio

- In the previous example, 4-bit samples yield 16 different binary codes for the samples: 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111.
- The accuracy of the digital audio sequence compared to the original analog audio signal increases with:
  - Increased sampling rate, and
  - Increased number of bits per sample
- In typical uncompressed audio, such as audio on a CD, an audio "sample" is measured 44,100 times per second to a resolution of 16 bits, resulting in an integer from -32768 (-2^{15}) to 32767 ($+2^{15} - 1$).

4

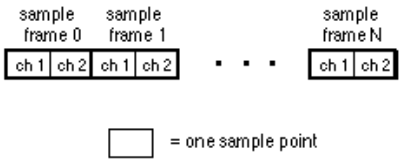
What is a Sample?

- The term "sample" as described above led to the term "sampler" — an audio synthesizer based on recording, storing, and playing back audio recordings of instrumental notes.
  - Whether it was misunderstanding, laziness, or simply the lack of a better term, people began referring to these short sound recordings as "samples," e.g. a "flute sample" or a "piano sample" or a "sample library" containing a collection of sounds.
- Even later, DJ culture began to use "samplers" to "repurpose" existing commercial recordings, and short excerpts of music began to be referred to as "samples."
- For this class, we'll usually use "sample" to mean a single number representing the amplitude of a waveform at a point in time but be aware that context is important.

5

Multichannel Audio

- Many audio files contain multiple channels. E.g. stereo has a left channel and a right channel, where a channel is just a single waveform represented by a sequence of samples.
- To store multichannel audio, we *interleave* samples so that all the samples at time 0 precede all the samples at time 1 and these are followed by samples at time 2, etc. All of the samples at a given time are called a *frame*. Thus, a multichannel audio file is a sequence of *frames*, where each frame is a sequence of *samples*.

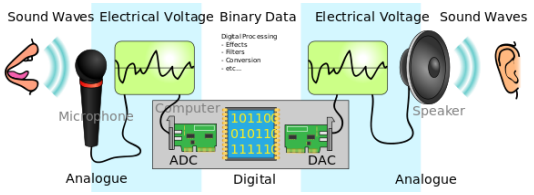


Credit: http://paulbourke.net/dataformats/audio/

6

Sample Rate

- The *sample rate* of digital audio is the number of samples per second.
- Samples are always measured periodically, that is, the time interval between successive samples is exactly the same.
  - Good converters measure timing error in picoseconds (10^{-12} second), and even the cheapest converters are probably accurate to 1 nanosecond (10^{-9} second).
- An analog-to-digital converter (ADC) samples an analog signal representing sound into a digital audio signal.
- A digital-to-analog converter (DAC) converts a digital audio signal into an analog signal representing sound.

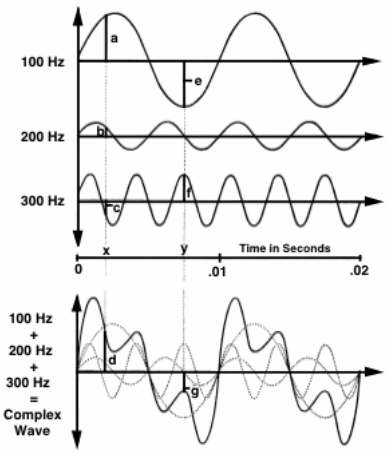


Wikibooks: Fundamentals of Data Representation: Analogue and Digital

7

Sample rate

- The sample rate determines what frequencies can be recorded.
- **To capture a frequency of X, you must sample the signal at a sample rate of 2X.** (sampling theorem)
- Speech has most of its frequencies below 4000 Hz (cycles per second), so your voice in telephone systems is sampled at 8000 Hz.
- We can hear generally up to 20,000 Hz, so music is sampled at 44,100 Hz (a little above twice the highest audible frequency).

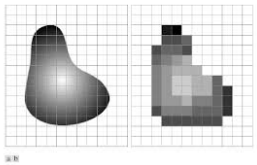


hearinghealthmatters.org

8

Sampling with Images

- Images are another example of sample-based digital representation!
- In images, we have a two-dimensional "signal" which is a function of X and Y rather than time. The samples are pixels and the "sample rate" is the number of pixels per inch.
- Sampling theory applies here too. Higher "frequencies" ("spacial frequencies") in images correspond to higher resolution. The more pixels per inch, the higher the spacial frequencies we can capture and the higher the image resolution.
- If we remove high frequencies, we lose image resolution, resulting in a blurry image.



9

Audio in p5.js Sketches

- We must use **template-all.zip** rather than template-p5only.zip .
- Basic rules for audio:
  - If you want to make sound, call `useSound()` in `setup()`.
  - If you call `useSound()`, you must define `soundSetup()`. Put all sound initialization code in `soundSetup()`. (Do not put sound code in `setup()`).
  - If you want to load pre-digitized sounds, you would have them loaded just like you did with images in `preload()`.
  - When sounds are loaded, they act as objects and have various methods that you can use to set playback features or play the sound.
- Public-domain sounds may be found at `freesound.org` .

10

Setting up a Local Server

- Browsers will likely block downloading music content just as they block images from most sites. (imgur has the right settings to allow it)
- One option using `php`:
  - Go to terminal window (on Mac) and `cd` to the directory with your sketch and sound file(s).
  - Then enter: `php –S localhost:8000`
  - Finally, point your browser to: `http://localhost:8000/`
- See here for other options: `https://github.com/processing/p5.js/wiki/Local-server`

11

Example

```
var bgcol = 200; // gray
var buzz; // this will hold a sound

function preload() {
    buzz = loadSound("http://localhost:8000/gameshow-buzzer.wav");
}

function setup() {
    createCanvas(200, 100);
    frameRate(10); // controls "flash" time
    useSound();
}

function soundSetup() {  buzz.setVolume(0.5);  }
```

12

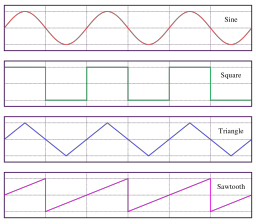
Example (cont'd)

```
function draw() {
    background(bgcol);
    bgcol = 200; // restore flash
    fill(0);
    textFont("Arial", 24);
    text("click me", 50, 50);
}
function mousePressed() {
    buzz.play();
    bgcol = 255; // bright flash
}
```

13

p5.Oscillator

- Creates a signal that oscillates between -1.0 and 1.0.
  - By default, the oscillation takes the form of a sinusoidal shape ('sine').
  - The frequency defaults to 440 oscillations per second (440Hz).
- Some methods:
  - `start()` Start an oscillator.
  - `stop()` Stop an oscillator.
  - `amp()` Set the amplitude between 0 and 1.0.
  - `freq()` Set frequency of an oscillator to a value.
  - `setType()` Set type to 'sine', 'square', 'triangle', or 'sawtooth' .
  - `disconnect()` Do not send output of this oscillator to the speakers.



14

Example

```
var myTone;

function setup() {
    createCanvas(300, 300);
    useSound();
}

function soundSetup() { // setup for audio generation
    myTone = new p5.Oscillator();
    myTone.setType('sine');
    myTone.freq(800);
    myTone.start();
}
```

15

Example (cont'd)

Volume ↔

Pitch ↕

```
function draw() {
   background(200);
   fill(0);
   ellipse(mouseX, mouseY, 20, 20);
   myTone.amp(constrain(mouseX / width, 0, 1));
   myTone.freq(constrain(200 + 1000*(mouseY / height), 200, 1200));
   if (mouseX > 2*width) {
       myTone.stop();
       noLoop();
   }
}
```

A fraction between 0 and 1 when the mouse is in the canvas bounds.

16