Andrew Dircks

CEE 4795

15 April 2022

Final Project Brainstorm

I propose to build a simple "smart parking" application for my house's lot. The three major components of this system include

1. Calibrate and connect ultrasonic distance sensors to determine *taken* or *available* for each of the four spots in the lot
    - Depending on the location of the sensors, different thresholds for recorded distances would determine a spot's availability. For example, a sensor can be put on the ground below a spot and pointed upwards – if the sensor records 1 foot, it is likely that a car is parked above it and the spot is *taken.*
2. Connect an Arduino to the internet (via Ethernet Shield) and post sensor measurements at a specified time interval to a custom web API via HTTP
3. Build a simple data processing and frontend site to display spot availability in real time

I work as a software engineer in the IoT field (specifically monitoring/operating energy storage assets). My exposure with sensors has been purely on the software side – streaming data from various sources and processing it accordingly, most closely related to the third step listed above. It would be very valuable for me to build an entire (simple) IoT application from scratch. The process of monitoring a physical measurement with a sensor, hooking an Arduino up to the internet, and building a small webapp to display this information is super interesting to me – this was my main goal in taking this class.

Additionally, I want to build something with some practical value. The emerging "smart parking" industry has interested me for a while. The concept is extremely simple, and it adds major convenience to everyday life. I live in a house with 13 people (including Bennett McCombe and Alicia Wagner, in this class), and our parking situation is far from ideal. Using simple, lightweight sensors to monitor the four parking spots in our lot would provide actual value to the residents, and it would be an incredibly fun experience to build!

Ultrasonic Ranging Module HC – SR04

Datasheet: https://www.mouser.com/datasheet/2/813/HCSR04-2307007.pdf

Purchase: https://www.mouser.com/ProductDetail/SparkFun/SEN-15569?qs=P1JMDcb91o46Sr4O2RLYiA%3D%3D&mgh=1&gclid=Cj0KCQjwxtSSBhDYARIsAEn0thTEv9n5MpDXeFTHqC2YZW6F6SCmgMgVvKgE6IPK3t3A_418l1ERscoaAgRpEALw_wcB

Output Format: Analog

Digitization Range: The distance is proportional to the duration of the "echo pulse", which results from the module's sonic burst echoing off an object in front of the sensor. The conversion is

$$distance\ (cm) = \frac{echo\ pulse\ duration\ (\mu s)}{58}$$

The digitization range would thus change depending on how the sensor was mounted. If the sensor was mounted below where a car would park, we would need to measure from roughly 15cm to 60cm, which correspond to echo pulse durations of 870 μs and 3,480 μs. If measurements were made with one micro-second granularity, there would be 3,480 – 870 = 2,610 possible values, which corresponds to about an 11-bit digitization range.

Sampling Rate: It is recommended to use "over a 60ms measurement cycle". For the proposed application, however, this granularity is far too fine – I would make measurements about once a minute.

Processor: The Arduino Uno would be more than sufficient for this project. The Arduino Ethernet Shield would be used to stream this data via HTTP.

Recording Time Length: Ideally, this would be a "real time" application, so the sensor should be running for an extended period. The measurements would likely go on for about a month before I took down the system.

Cost: $3.95

Distance Range: 2cm – 1m