Power Demand Response System

Team 21 - Antonio Martinez and William Widjaja
TA: Evan Widloski

1. Introduction

1.1 Problem

Stability of the grid is an increasingly difficult problem with the introduction of renewable energy sources. From a 2018 study on consumer energy management, 68% (up 3% from 2016) of residential consumers are very concerned about climate change and their carbon footprints while 7 in 10 businesses say their customers are demanding that a certain percentage of their electricity comes from renewable sources (up 9 points from 2017) [1]. It is clear that there is a growing demand by both consumers and businesses for an increase in renewable energy sources. The problem is that these renewable energy sources, wind and solar, are not stable and hard to predict in terms of energy output. Each new source creates another point which must be balanced against the demand. Therefore, a system of energy management must be put in place in order to maintain balance. Traditionally, energy is balanced from the perspective of the utility company.

In 2015, the California Public Utilities Commission ordered its state's investor-owned utilities to adopt time-of-use (TOU) rates by 2019 [2]. This method of energy management has consumers being charged more during peak demand hours and rewards consumers who limit their energy during said hours. This alleviates some of the stress put on the power grid, but it also creates expensive pricing. It is still up for debate how beneficial this TOU method is compared to other ways of managing power consumption such as tier-rates where consumers pay prices dependent on how their usage compares to overall average usage.

1.2 Solution

It is, however, possible to balance the grid strictly from the demand side while still having monetary incentivisation. Time of Use methods all share a dependance on the consumer to consciously and actively manage their own consumption. Instead, with our functioning communication system, consumers may one day be able to opt-in to have their utilities be effectively controlled from the demand side.

Implementing demand response systems requires significant coordination of a large number of varying loads drawing power from the same grid. By developing a centralized means of communicating to loads on the grid, a utility company could make the consumer side demand more predictable, lowering the total capacity required and passing the economic benefit on to the consumers who participate. We aim to develop this communication system to vary the effective duty cycle of several loads on the grid and coordinate the demand to reduce the peak load.

1.3 Visual Aid

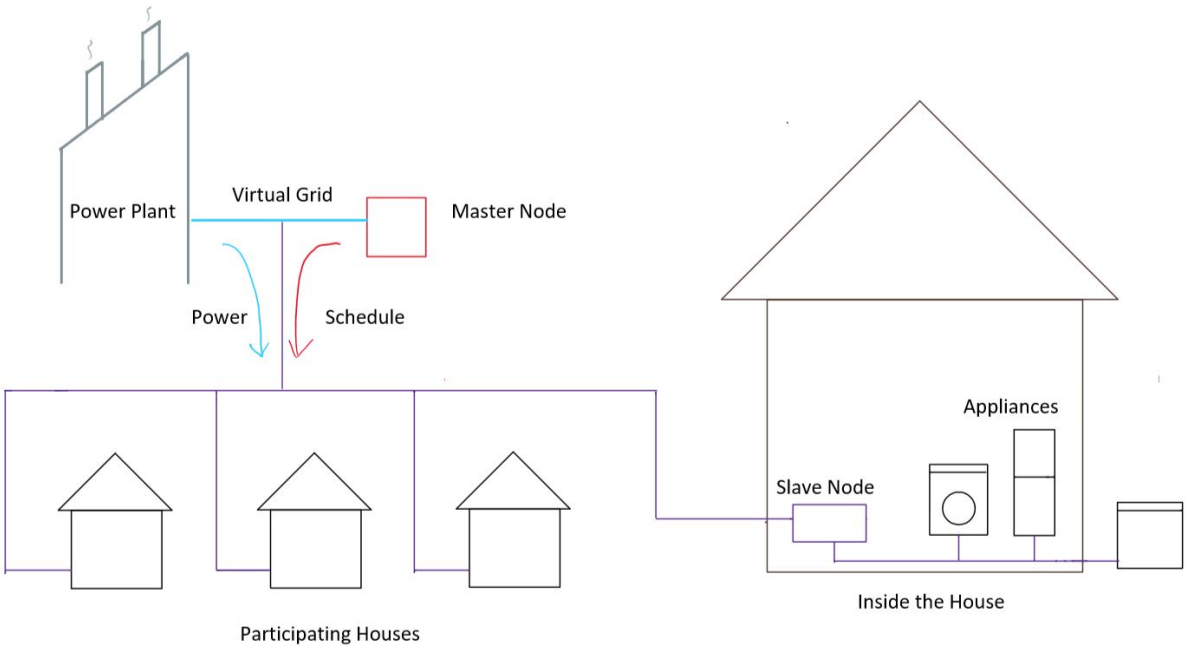

Figure 1. System Diagram

The figure above illustrates how we foresee our system being used with the power grid. The power source sends power to the houses and a signal for the Master Node to interpret. Once interpreted, scheduling announcements are sent to the Slave Nodes of each participating house, and each slave node commands power usage of connected appliances based on the schedule. Blue lines represent power, red represents scheduling data, and purple represents both.

1.4 High Level Requirements

1. Successfully transmit data from the power source across the powerline (the virtual grid). An n-bit signal should be able to ride a carrier wave to a receiver on the load end.

2. Effectively dictate the on/off states of connected appliances based on schedules received from transmitted data.
3. Be able to change schedules based on seasonal demand. For example, the demand for the Fall is different from Winter, so these seasons require different schedules.

2. Design

2.1 Block Diagram

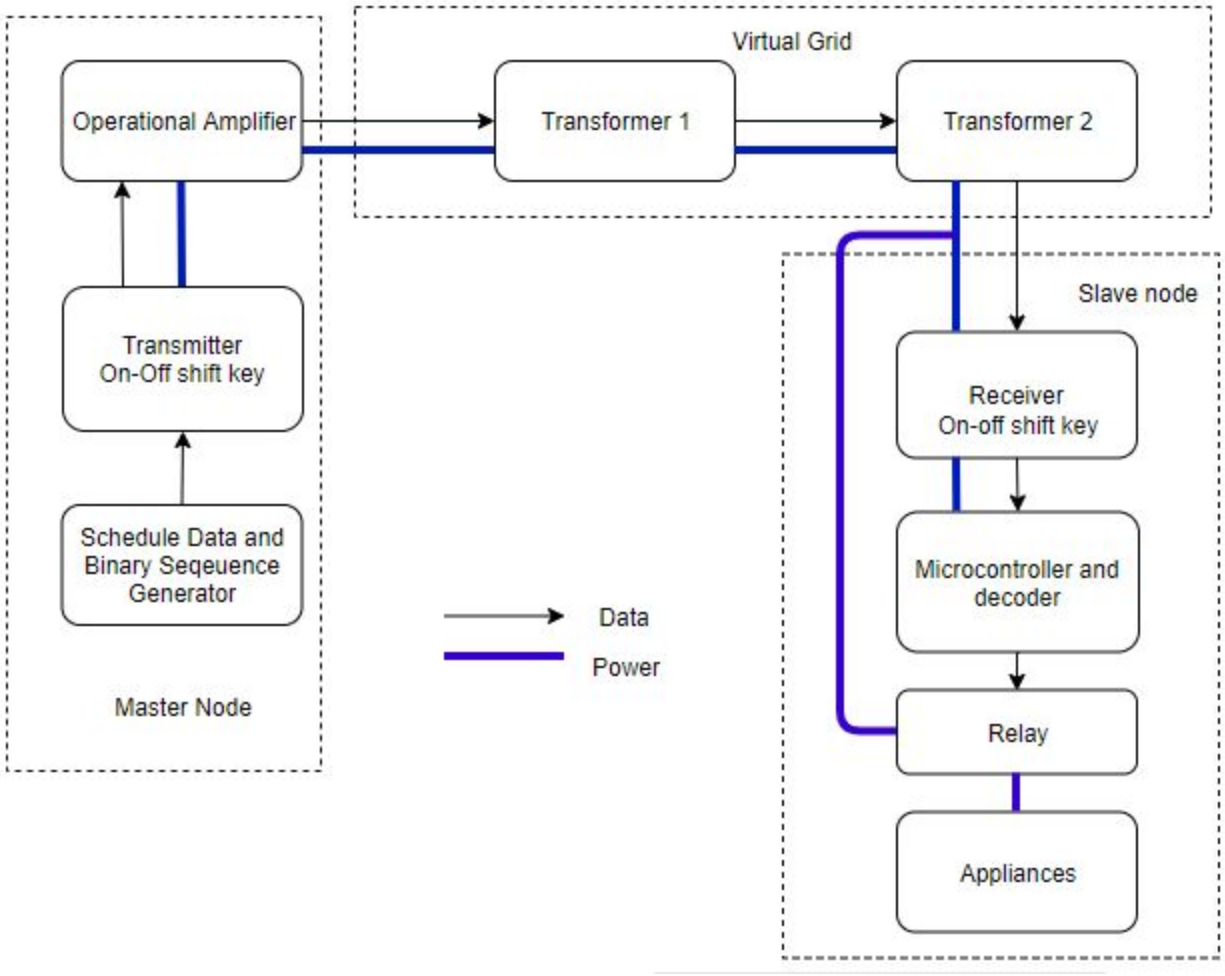

Figure 2. Block Diagram

Figure 2: For the successful operation of our system, we require the three subsystems: the Master Node, the Virtual Grid, and the Slave node. The Master Node processes scheduling data via on-off shift keying and an amplifier to send it across the Virtual Grid. At the other end, the Slave node receives the scheduling data where a microcontroller then commands connected appliances according to the schedule. New scheduling data

sent to the slave node will appropriately change the microcontroller's command on appliances.

2.2 Master Node: The initial control at the substation is a series of signals which will eventually control all loads on the grid. It also ensures there is enough power on the grid for the sum of all cycles it governs.

2.2.1 Schedule Data and Binary Sequence Generator: Signal in the form of a binary sequence to be modulated with a carrier signal. We must assign the bits of a binary sequence to represent data understood by the microcontroller in the Slave Node. Each new schedule requires a new or updated binary sequence. We simulate this data using a pulse generator, but the final design would send signals with a microcontroller.

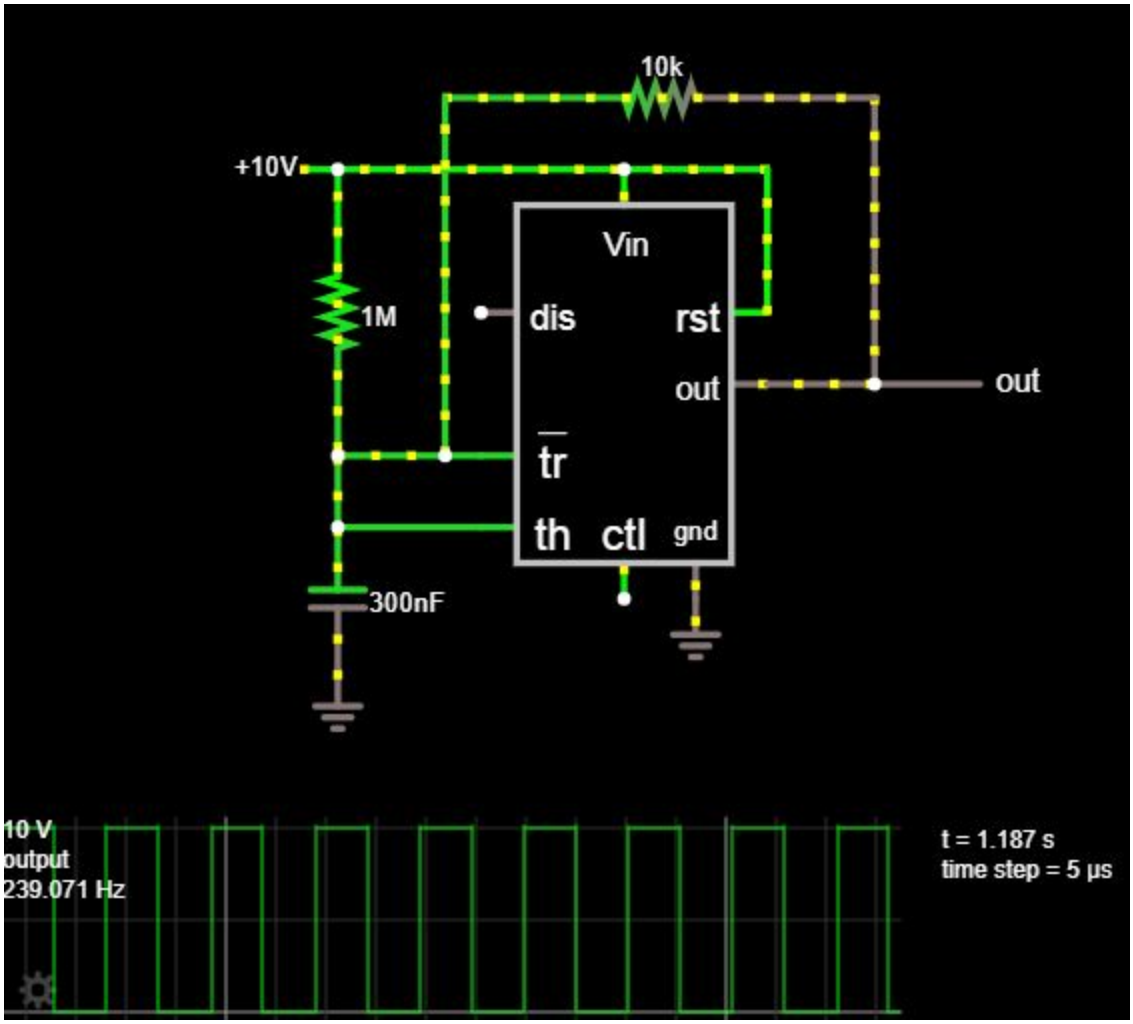

Figure 3a. Simulation of Binary Sequence Generator Circuit (BSG) In the BSG, the value of the 10k resistance can be adjusted dynamically to create a specific bit stream.

A higher resistance would increase the width of the pulse. For example, a 20k resistor would double the pulse width. By manipulating this resistor, we can generate a binary sequence to modulate a carrier signal.

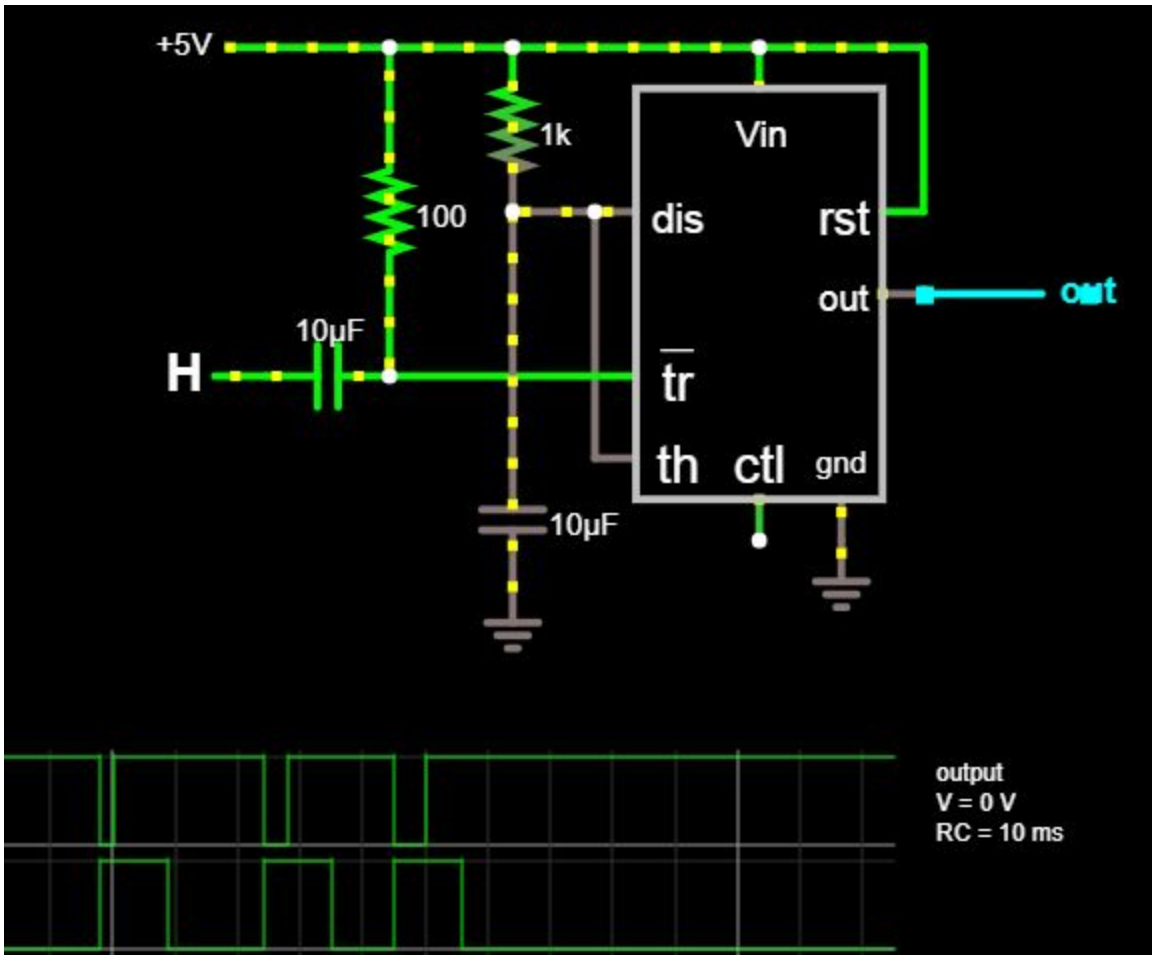

Figure 3b. Simulation of Binary Sequence Generator Circuit with varied pulse widths of low voltages. This will represent the data words. The RC constant is the width of the pulse and is equal to the product of the 1kohm resistor and 10uF capacitor close to ground. The device is active low and the data would come from the point labelled H.

Requirements	Verification
Desired input matches desired output	Tap 3 times to send “111.” Scope 3 pulses at the output of the BSG

2.2.1 On-off shift keying (OOK): This is the simplest form of amplitude shift keying modulation and will be our method of representing a carrier wave signal as digital data.

A high frequency carrier signal is modulated with a binary sequence, creating a signal that represents the original carrier signal whenever the binary signal was 1, and a flatline signal whenever the binary signal was 0. This is illustrated in the figure below. For simplistic purposes, the frequency and phase of the carrier signal will remain constant.

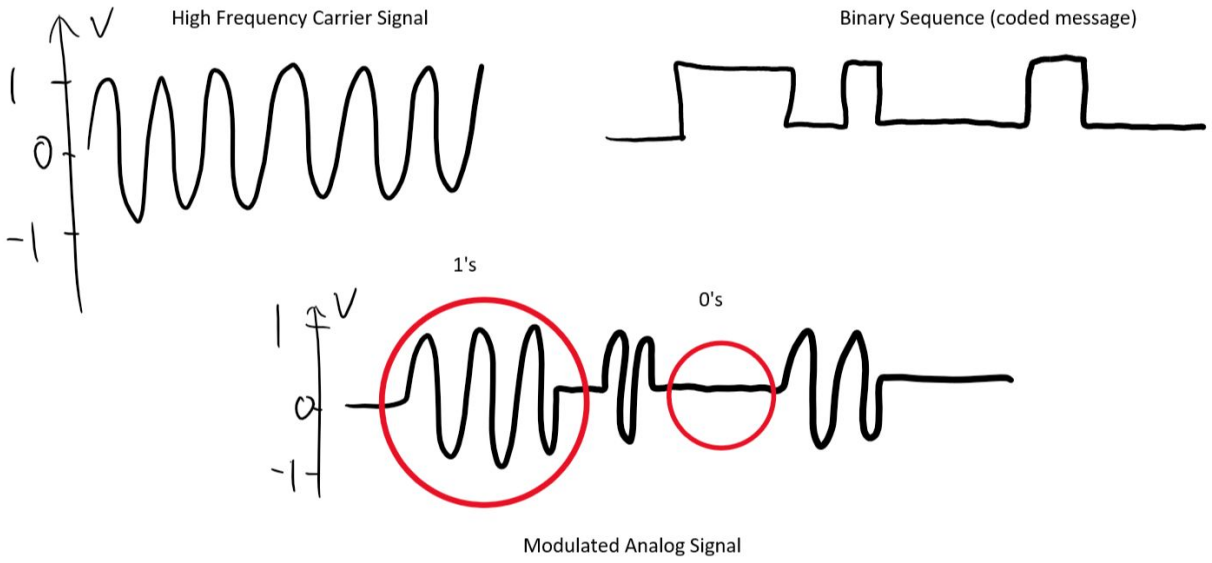

Figure 4. OOK Visual Representation

We must establish a period T based on the frequency that represents 1 bit. This is to differentiate the carrier wave oscillating from -1 to 1 from the flatline portions of the modulated wave. So say peak-to-peak oscillation takes 1s; we establish 1s = 1bit. Now say the 3 initial flatline portions are 2s, 1s and 3s long respectively, then our modulated signal represents 00111011000. Now that our digital code is represented in analog form, it can be transmitted through the virtual grid.

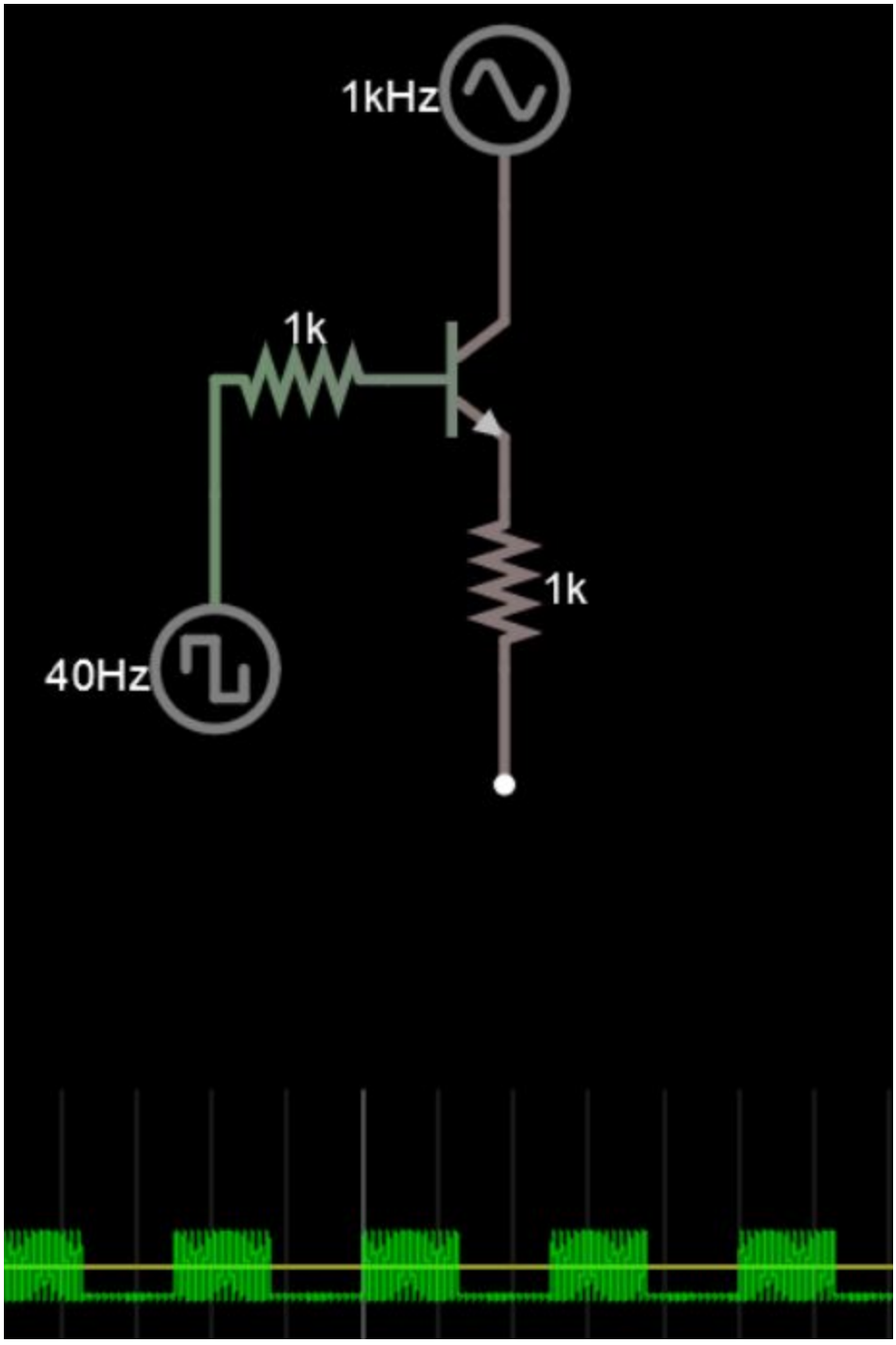

Figure 5. Falstad Simulation of OOK Circuit with Measured Output

As shown in the figure above, the non-zero input by the digital signal (40Hz square wave) will activate the transistor so that the carrier signal (1kHz AC source) is the output. For a zero input by the digital signal, there is a ~-1V output. Ideally, the output would be zero, but with some offsetting, we can see how the modulator it intended to function.

Requirements (On-Off shift keying)	Verification
Correct modulation of carrier signal	Example: Scope both the carrier and the modulated signal and read “1011” and compare it to the input of the binary sequence generator

2.2.2 Transmitter: The transmitter will connect to an amplifier which sends the modulated signal across the virtual grid. As its name suggests, it moves the data from the utility substation onto the grid.

Requirements	Verification
Must send signal without detrimental distortion.	Example: Scope output of “1011” at the wire which goes into the operational amplifier, with 10% variances in pulse widths

2.2.3 Amplifier: Must produce gain of 20 V/V or greater without clipping. This is closer to a “data sheet” parameter but is worth mentioning because a common source of problems arises from incorrect or non-functioning amplifiers (e.g, not powered or not connected)

2.3 Virtual Grid: The path which our signals are traveling on. For our design we will treat this electrically as a wire with a very small resistance.

2.3.1 Transformers: On our block diagram, these transformers represent the step up/down transformers used by the grid. For our project, since we are working on a virtual grid, we will not use these transformers. Instead we will be utilizing 1:1 isolating transformers.

Requirements	Verification
All signals made by the system must be under 500 kHz by U.S. law. No radio waves in free space if unshielded wiring is used.	Measure signals at each electrical node and ensure there are no frequency components above 500 kHz

2.4 Slave Node:

2.4.1 Receiver: Acts as an asynchronous demodulator to convert the input analog signal into a digital output readable by the microcontroller. As shown in Figure 5., this module consists of a half-wave rectifier to deliver a positive half output into a low-pass filter. This filter delivers the envelope of the signal which is shaped more into a digital signal. Finally, the envelope is sent to a comparator to finalize the digital signal.

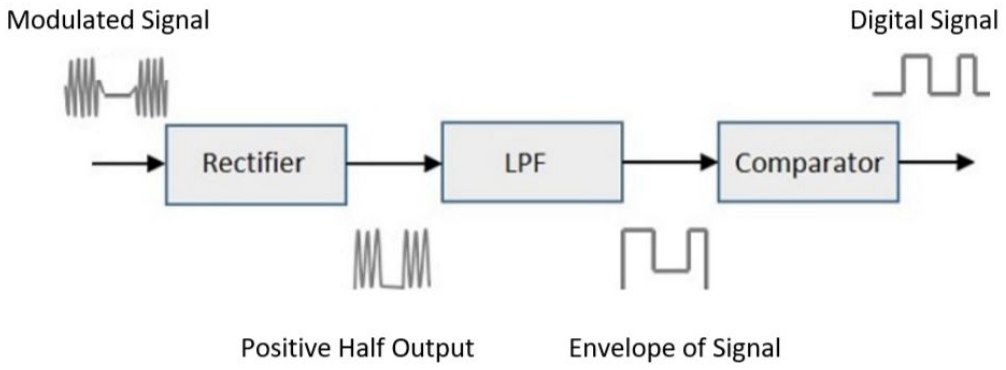

Figure 6. Demodulator Diagram

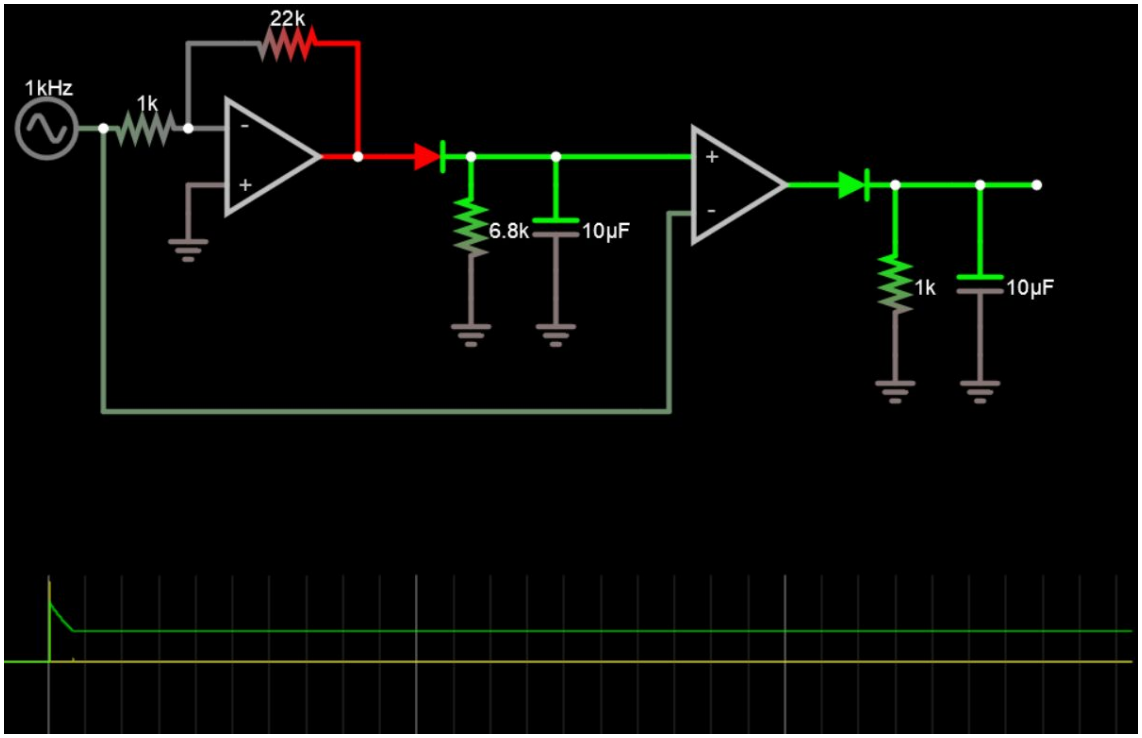

Figure 7. Falstad Simulation of Demodulator

We can see from Figure 7. That the final output is a clean unit step due to the input not being modulated.

Requirements	Verification
Must decode the information from the transmitter correctly, at least in an ideal situation where the transmitted signal is perfect/directly in communication with the receiver. For example, if 1011 is transmitted, 1011 should be received.	Scope the output of the demodulator and the output of the transmitter simultaneously. The graphs should be very similar to within 5% rise and fall time delay. Find correct values for resistors and capacitors to properly filter signals using these equations: $V\,ripple = Iload/fc$ $V\,cap = V\,source\,e^{-t/RC}$ To minimize noise while having reasonable capacitor discharge times

2.4.2 Microcontroller: Essentially opens or closes the relay for the load based on initial binary sequence representing schedule commands being properly transmitted and received

The data which is being sent through the virtual grid is an array of arrays. Since this is akin to a “broadcast” on the grid, all device schedules are actually available in this array. The following pseudocode illustrates this:

allSchedules = [A, B, C, D, E …] //This is the array with all schedules

B = [deviceId = “B”, changeSchedule = true, newSchedule = [12pm: on, 1pm: off, default: off], currentTime = 12pm] //This is an example of one of the subarrays. This subarray contains scheduling information for a load with serial identification “B.” Only the device with matching serial ID = “B” will follow the instructions set by array B. In this specific array, the device B will change its schedule and receive power from 12pm to 1pm, where it will then turn off and stay off until a new instruction arrives.

Requirements	Verification
Must store correct bit representations as intended by transmitter/receiver and properly control the relay switch with those signals.	Send a test input to the microcontroller and then view the output through 4 LEDs or a console. The 4 LEDs represent the next 4 hours of operation and LED on = power and LED off = no power. The console command would just output a 4 bit array where 1 = power and 0 = no power

2.4.3 Relay: The switch that opens or closes depending on the commands of the microcontroller so that a load (appliance) is connected to power. This is also closer to a “data sheet” parameter, but worth mentioning since it controls power to the appliance, the final step and goal of the data word sent.

Requirements	Verification
Must respond to a signal with no more than 5s delay.	Send a test input and measure the time it takes for the relay to react

2.5 Tolerance Analysis and Feasibility

Because our communication system is designed to send information from distant power sources to the devices they are powering, we must keep in mind the propagation of higher frequency signals across a power line. Our biggest challenges will come from filtering out noise and decoding a signal which might have dirty power, such as incorrect voltage amplitudes or extra frequency components. We need to consider variances in pulse width and variances in filters, two keys to the virtual grid power line communication of our project.

First, we revisit Figure 3b, but this time, we choose a resistance of 900 ohms (instead of the planned 1000 ohms) and a capacitance of 9 uF (instead of 10 uF). This represents a 10% error on each part. We then send a sequence of 4 bits manually:

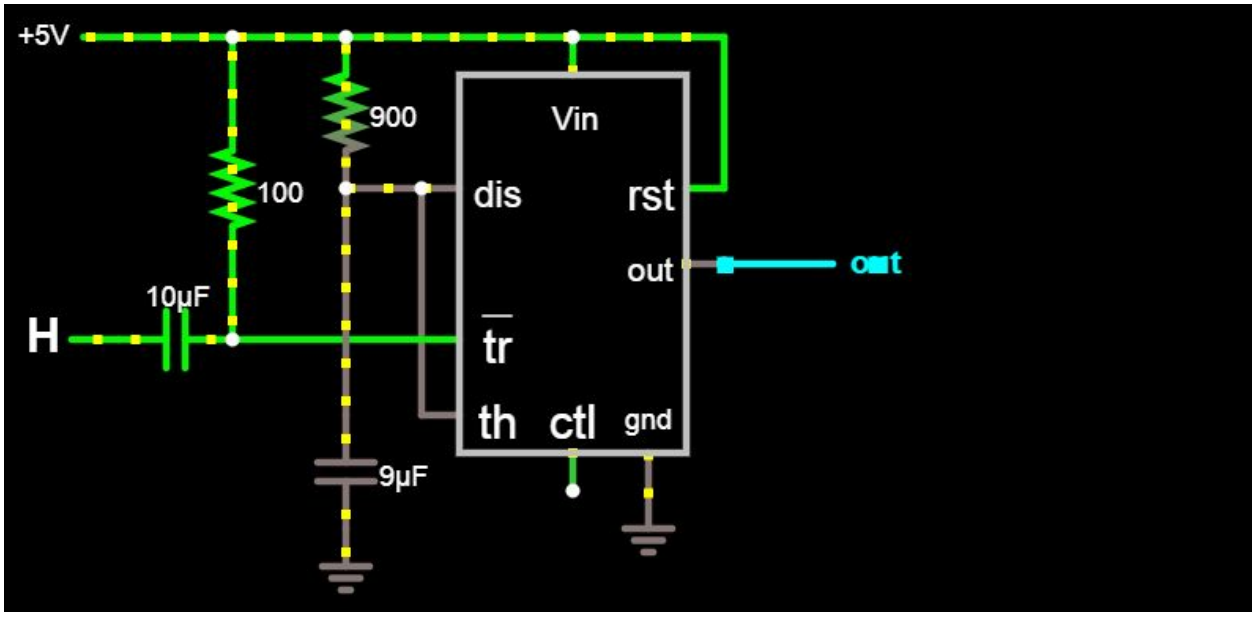

Figure 8a. Binary sequence generator with 10% error on components.

Figure 8b. The graph of the 4 bits sent in Figure 8a, the circuit with 10% error in parts.

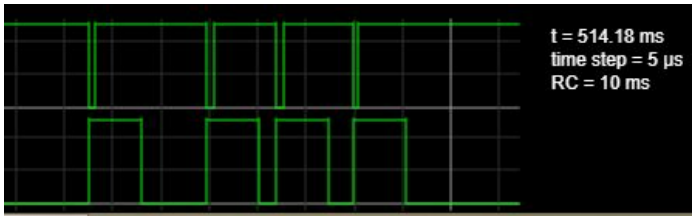

Figure 8c. The graph of the 4 bits sent on the circuits with ideal parts.

We can see that the errors stack multiplicatively. The resulting RC time constant, which dictates the pulse width, is a full 19% shorter than the planned pulse width. This change in pulse width must be accounted for in order to transmit correct information. This is evidence that true measurements need to be taken after board assembly to determine the specific circuit's true tolerance. For later parts of the design, the scheduling algorithms must be able to handle inputs with at least +/-(1-n^m) tolerance, where n is the highest tolerance of an element in the circuit and m is the total number of elements contributing to the filters of the signal. In practical terms, this means the receiver should

be able to decode 20% shorter or 20% longer width logical 1s than the 1s generated at the transmitter.

While not explicitly seen in the images, the voltage level could also be altered unexpectedly due to tolerance issues, or parasitic components. It is reasonable to assume a voltage level could change by as much as 10%. We can obviate this issue by defining the high/low voltage cutoff up to 20% below the targeted high analog voltage.

The active-low design of the binary sequence generator already accounts for varying pulse width inputs. The next important point is the filtering, which is also dependent on resistor and capacitor tolerances. For the high pass filter, which will be important for filtering our desired signal in, we have a cutoff frequency determined by 1/(2*pi*R*C). We consider again, a 10% tolerance per part and see that the true cutoff could be up to 23.5% higher than desired or 17% lower than desired. Assuming the frequency of the modulated signal is 1000 Hz, we would set a high pass filter to 806 Hz or lower to account for the possibility of a much higher true frequency to avoid cutting our signal.

2.5.1 Feasibility

In order to justify the benefit of the system, it is helpful to characterize the load profile from the power plant's perspective and simulate it from the superposition of many residential units throughout the year.

We can simulate this with a load profile generation in Python. First, import math, plotting and graphical libraries:

```
%matplotlib inline

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import enlopy as el
```

Now, define the percentage of the week in which the loads will be operating on a "work day" (Mon-Fri inclusive). Weekends are taken from a different set of tables and have different load curves. Call this the workTime variable.

```
workTime = .75 # 75% weekdays and 25% weekends

Load1 =  el.gen_load_from_daily_monthly(monthlyLoad, dailyWorkingLoad, dailyNonWorkingLoad,
Weight)
```

```
Load1.name = 'firstHouse'
Load1.plot(figsize=(16,3), linewidth =.2, grid = False, perc_list=[[1,99], [25,75], 50], color='orange'));
```

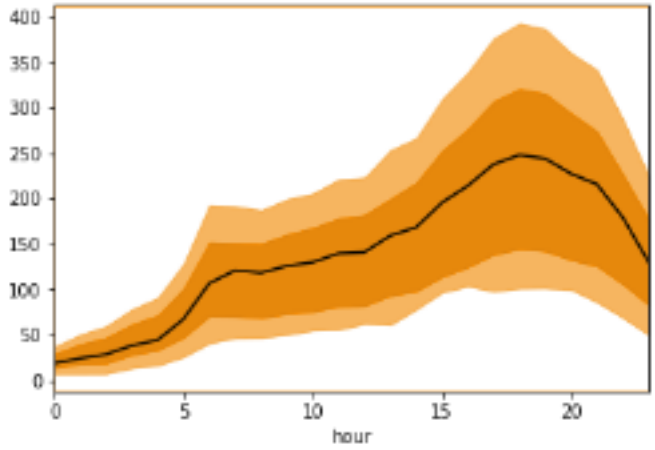

Figure 9a. The load duration curve for a peak summer day. The peak load is at nearly 350 watts. The other color shades are percentiles of deviation.

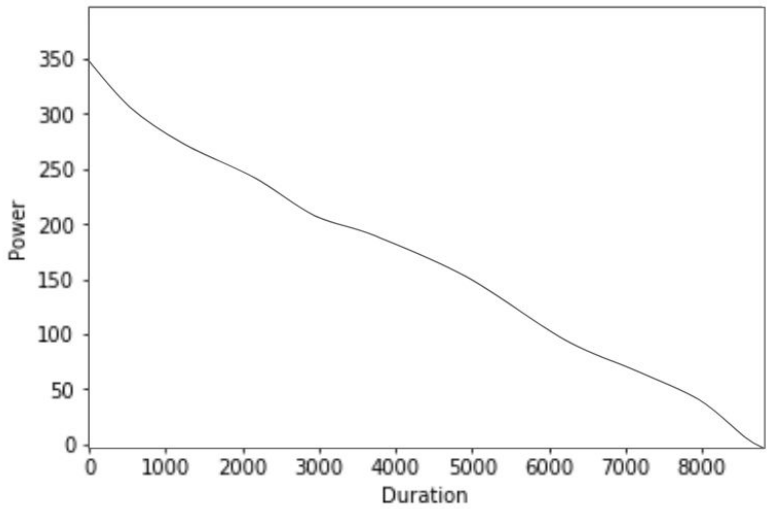

Figure 9b. The load duration curve for one year without our system.

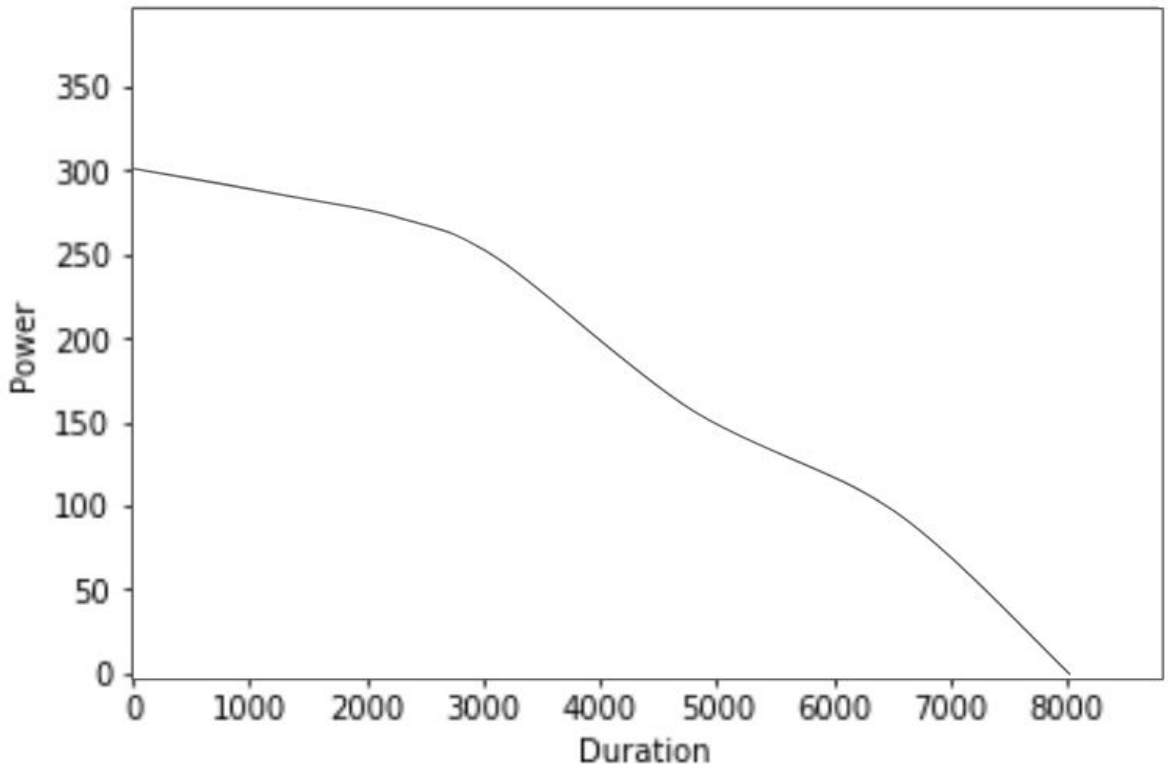

Figure 9c. The load duration curve for one year with our system. More time is spent at a lower maximum power throughout the year. Physically, energy is conserved, but peak load is reduced, freeing up power for the Utilities.

2.6 Cost Analysis:

2.6.1 Estimated Total Labor Cost

Engineer	Hourly Rate	Hours (12 hours/week x 12 weeks)	Hourly Rate x Hours x 2.5
Antonio Martinez	$40.00	144	$14,400
William Widjaja	$40.00	144	$14,400
			Total: $28,800

2.6.2 Estimated Total Parts Cost

Description (hyperlinked)	Manufacturer	Part Number	Quantity	Unit Cost	Total Cost (prices may vary on other days)
Assorted Resistors, Capacitors, Wires	Texas Instruments	Various	5 of each from 10, 20, 50, 100, 1k, 10k and 1M	$0.40	$14.00
Isolating Transformer (N-48X)	Triad Magnetics	237-1856-ND	2	$14.62	$29.24
Step Up/ Step Down Transformer(3FD-210)	Tamura	MT2096-ND	2	$5.73	$11.46
1N4004 Diodes	Diodes Incorporated	1N4004DICT-ND	6	$0.18	$1.08
Arduino Uno Rev3	Mouser Electronics	805833349090	1	$22.00	$22.00
Microcontroller	Microchip Technology	ATMEGA328-PU-ND	2	$1.96	$3.92
Operational Amplifier	Mouser Electronics	LM741	3	$1.00	$3.00
555 Timer Chip	Texas Instruments	LM555CMM/NOPB	2	$1.00	$2.00
					Total: $86.70

Grand Total: $28,886.70

2.7 Schedule:

Week of	Antonio	William
9/30	Design Document, Simulate Circuit	Design Document, Order Parts
10/7	Acquire Parts, Create and test breadboard circuit of modulator	Acquire Parts, Create and test breadboard circuit of bit generation
10/14	Design modulator and demodulator circuits on EagleCAD. Order PCB	Data generation circuit created and program written for sending bits. Order PCB.
10/21	Acquire and test PCB. Revise if necessary. In either case, send another order for more nodes.	Combine the modulating PCBs and the data generation. Begin testing with real DC loads. Test at least one DC load.
10/28	Test with multiple nodes. Last round of PCBs submitted to test multiple nodes, just in case they are not passing tests	Test with multiple nodes. Last round of PCBs submitted to test multiple nodes, just in case they are not passing tests
11/4	Fix any PCB issues that may have come up	Software encoding/decoding and scheduling algorithms complete
11/11	Continue fixing PCB issues, mount PCBs, help William	Fix any software issues, else, prepare for Mock Demo
11/18	Prepare for Mock Demo	Prepare for Mock Demo
11/25	Work on Final Paper, Resolve Remaining Issues	Work on Final Paper, Resolve Remaining Issues
12/2	Final Demo and Mock Presentation	Final Demo and Mock Presentation
12/9	Final Presentation and Final Paper	Final Presentation and Final Paper

3. Ethics and Safety

Due to the nature of our project as a broad communication system, there is a need for proper security of the information being sent. Our project is designed to eventually help dictate the usage of compliant consumers, so it is our responsibility to come up with a secure method of sending and receiving data and being transparent with our understanding of how secure the system is.
One specific consideration is that the data being sent contains scheduling information for everyone's devices. For real applications, it would be important to implement an encryption protocol, such as public key cryptography, where the private key is safely

stored on the hardware of a load device. This would secure the data being sent and ensure that each load only has access to its own scheduling data. The public key encryption would need to occur before the data enters the virtual grid.

In compliance to the IEEE Code of Ethics and the ACM Code of Ethics and Professional Conduct [3][4], we will hold the security and safety of users paramount in our development of this project to exemplify the specific codes 1.1,1.3, 1.6, and 1.7. Note that we do not restrict ourselves to these only these codes but are stating that these codes are more relevant to the nature of our project. We do not foresee any ethical dilemmas created by our project. The system is intended to be used by consumers properly educated of what our communication system does and how it is intended to control their consumption. It will not be marketed or sold to consumers who would suffer compromises to their health from the rate limited use of electricity, such as a person on hospice.

Lastly, we will be careful to follow all lab procedures described by the ECE 445 guidelines and will not hesitate to ask questions or express concerns to faculty and classmates during the process of building this project. As partners, we will hold each other accountable to follow all codes of ethics especially in holding high standards of competence, conduct, and ethical practice, knowing and respecting existing rules pertaining to laws in the power and communications industry, and assisting each other during this project development, and to support each other in following these codes.

References

[1] Deloitte. 'Deloitte Resources 2018 Study', 2018. [Online]. Available: https://www2.deloitte.com/content/dam/insights/us/articles/4568_Resource-survey-2018/DI_Deloitte-Resources-2018-survey.pdf

[2] H. Trabish. 'An emerging push for time-of-use rates sparks new debate about customer and grid impacts' *UtilityDive'*, 2019. [Online]. Available: https://www.utilitydive.com/news/an-emerging-push-for-time-of-use-rates-sparks-new-debates-about-customer-an/545009/

[3] Ieee.org. 'IEEE IEEE Code of Ethics', 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html.

[4] acm.org. 'ACM Code of Ethics and Professional Conduct', 2018. [Online] Available: https://www.acm.org/code-of-ethics