ELECOMP CAPSTONE SUMMIT
FRIDAY, MAY 10, 2019

THE UNIVERSITY OF RHODE ISLAND

THINK BIG WE DO℠

Blank Out Sign Connectivity Over a Cellular Network

Team Members: Shane O'Connor (CPE), Khiet Pham (CPE)

Technical Director: David Connolly | Consulting Technical Director: Brenden Smerbeck ('17)

Project Motivation

The main component of this project is Blank Out Signs(BOS), which are LED arrays that can display messages to motorists. As opposed to a Dynamic Message Sign which is more akin to an LCD monitor for a computer and more complicated to implement. A BOS operates by running current across resistors to create contacts that act as switches to different sign circuits with the assistance of a low-cost microcontroller to display various predefined messages. A good example of this is a typical lane control sign on the highway which designates open and closed lanes. Since the messages are pre-programmed, it can be difficult to change what is displayed, and very costly to implement an over land communication system. SES America hopes to use modern technology to retrofit old BOS technology with wireless communication capabilities. This will simplify the process of changing the sign's message by providing a user friendly web application to send messages to the desired sign.

Key Accomplishments

Web Application Overhaul: PHP to Javascript: Last year when this project was entered in the capstone program, the team had developed the entire project in PHP and HTML with a few functions in Javascript. One of the major changes we made to the project this year is changing all of that PHP and HTML over to Javascript entirely. In this new implementation we are using the MERN web development stack. MERN stands for Mongo, Express, React, Node. MongoDB is a noSQL database. Express is a framework for routing on web servers. React is a client side development framework. Finally, Node is a web server developer framework. This change is important because it localizes the whole project to one language for every need.

Deployed Application to EC2 Instance: Deploying the application to the EC2 instance allows us and others to access the web page which is an integral portion of our project. The application is hosted on Amazon Web Service's EC2 Instance which essentially acts as a processor in the cloud that applications can run on. On this instance, we have deployed a database as well to store user information and sign information. Having this allows us to do live network testing of the application to device.

Integration of Cellular Modem Communication to Web Application: Connecting the cellular modem and Arduino system to our application was a major milestone for our group. This opens a line of communication between one master sign and the application. The cellular modem sends a series of requests to the application that returns the message desired by the user and defined by an Over the Air protocol developed by SES America for the purpose of sending data over the LTE network. The cellular modem sends out requests that constantly check whether the sign is updated or not. **(Fig. 4)**

Unique Users and Role Based Access Control: All data from the web application will follow the CIA Principles of Confidentiality, Integrity, and Availability . First, implementing secure user authentication allows us to ensure that only unique, registered users can alter their signs in our web application. Each log in and sign up request is validated with an object schema validator to guarantee that the data being entered is valid and legitimate. Each user will have a unique session with JSON Web Tokens that will be authenticated with each request. If a request token's timestamp doesn't completely match the one on file, then the user will be denied access to their account. This is to prevent multiple sign ins on the same account. In addition, we established permission levels with Role Based Access Control to control what functions the user can perform on the web application. A normal user can view and modify their signs, while the Admin can view signs and change permission levels of any user in the system. **(Fig. 3)**

Sign Display and Functionality: As described in the cellular modem section, the sign displays a message that is retrieved from the application and whose functionality is defined by the Over the Air protocol **(Fig. 2)**. This is known as the "Display Message" function and is one of a few functions of the BOSCON system. The next major function is the "Forward" function. The Forward function sends along the message in a chain from master sign to slave sign. The received message is then displayed on the slave sign. The final function of import is the "Relay" function. The Relay function is a command sent to the first slave sign that tells that sign to forward the message to another slave sign**(Fig. 1.)**. Currently, the Relay and Forward functions are being simulated with an LED circuit connected to an RF transmitter and receiver since we lack the proper hardware to fully outfit two more signs.

Anticipated Best Outcome

The Anticipated Best Outcome of this project is to successfully interface a cellular modem with a microcontroller in order to send messages from a web application to a sign. This cellular modem and microcontroller system must be able to retrofit onto the existing Blank-Out Sign architecture. The web application must be able to communicate with the cellular modem via the internet and the LTE network thereafter. The web application must also store information about a signs location, status and message. The ability to display these messages remotely is critical to achieving the best outcome

Project Outcome

The Anticipated Best Outcome was achieved. We promised a web based application capable of transmitting a message to change a Blank Out Sign's display. We exceeded this by adding the ability to simulate the relay and forward functions outlined in the OTA protocol.

Figures

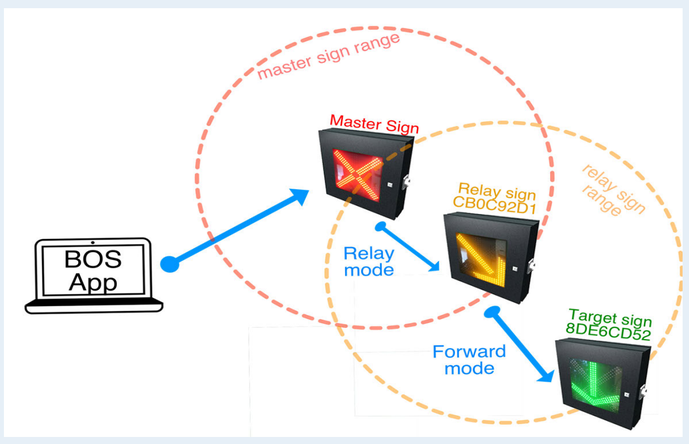

Fig. 1. The master-slave sign relationship. The application sends the command to a master sign that can relay and forward the message based off the command sent.

Commands					Values				
0	0	0	0	No Operation	0	0	0	0	Blank
0	0	0	1	Display a Message	0	0	0	1	
0	0	1	0	Return a Status Message	0	0	1	0	
0	1	0	0	Relay a Message	0	0	1	1	
1	0	0	0	Forward a Message	0	1	0	0	No Operation

Fig. 2. Table outlining the Over the Air protocol and the corresponding bit values.

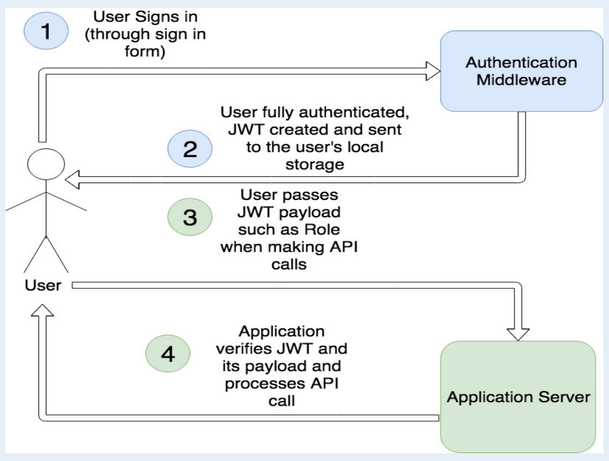

Fig. 3. This diagram describes the general flow of the security steps taken to ensure that the user performing functions on the web application is correctly verified.

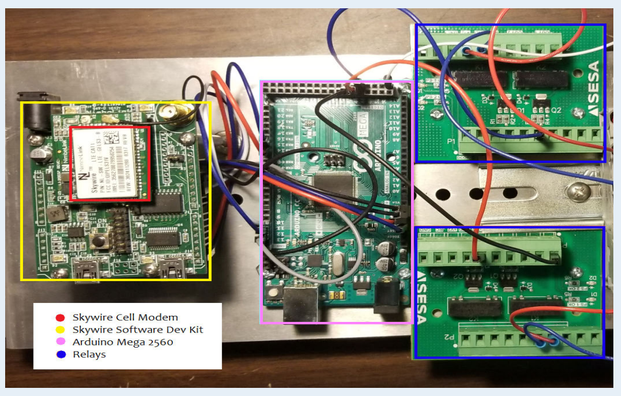

Fig. 4. Communication system for a master sign in the network.