

BusTracker

Christine Joseph

Grant Weber

Seung Goo Kang

Racine Hansen

Table of Contents

1. Introduction
2. Problem Statement and Proposed Solution
3. System Requirements
4. System Block Design
 - 4.1 Overall System
 - 4.2 Subsystem and Interface Requirements
 - 4.3 Future Enhancement Requirements
5. Networking Considerations
 - 5.1 Connection Considerations
 - 5.2 Data Flow
 - 5.3 HE910
 - 5.4 Protocol for Data Exchange
6. High Level Design Decisions
7. Open Questions
8. Major Component Costs
9. Conclusions

1 Introduction

2 Problem Statement and Proposed Solution

Buses are not always on time, which leads to problems when riders need to be somewhere at a specific time and can lead to riders standing outside waiting for a bus for extended periods of time. Though buses will inevitably fall behind schedule due to unforeseeable circumstances, the uncertainty around bus locations and arrival time can be combated with a bus tracking system. Adding a device that tracks the location of a bus and displays that GPS data for bus riders allows for better communication with bus riders and provides transit authorities with an effective, simple way to manage bus fleets. Our proposed solution would include a GPS tracking device to be placed inside buses with a web application that allows users to access real-time data about the bus's location. Riders can see where their bus is in order to know when it will be arriving at their stop--no more guessing and waiting. The web applications would also allow for transit operators to check on the status and location of their buses.

The BusTracker system outlined in this document will provide bus riders with accurate arrival predictions and bus system managers with an easy way to manage their fleets--reducing the uncertainty of static bus schedules.

3 System Requirements

The following are requirements our system will meet:

- The system will be powered from a 12V line from the bus or from the car cigarette lighter receptacle for testing purposes. Power regulation will be needed to provide 3.8V to the Telit HE910 cellular module and 3.3V to the microcontroller.

- The system is installed by setting up a SIM card with a service provider. Once power is provided to the system, the cellular chip will begin transmitting GPS coordinates to the web server. The system is used by the user pulling up our web application and seeing the bus location in real time. The box itself will have to be installed on the bus without being distracting to the driver so it should have the smallest possible form factor.
- Our system is low power and current so there are no safety concerns. All circuitry will be contained inside our box.
- None of the components are very heavy so our box will be lightweight with antennas coming through the box for reception.
- The microcontroller will need to communicate with the Telit chip via serial interface. We plan on using USB. The microcontroller will not need many I/O pin. We plan on using a PIC microcontroller with built in USB capabilities so that we do not need to use an external UART. We will need a PICKIT interface to program the microcontroller.

4 System Block Diagram

4.1 Overall System:

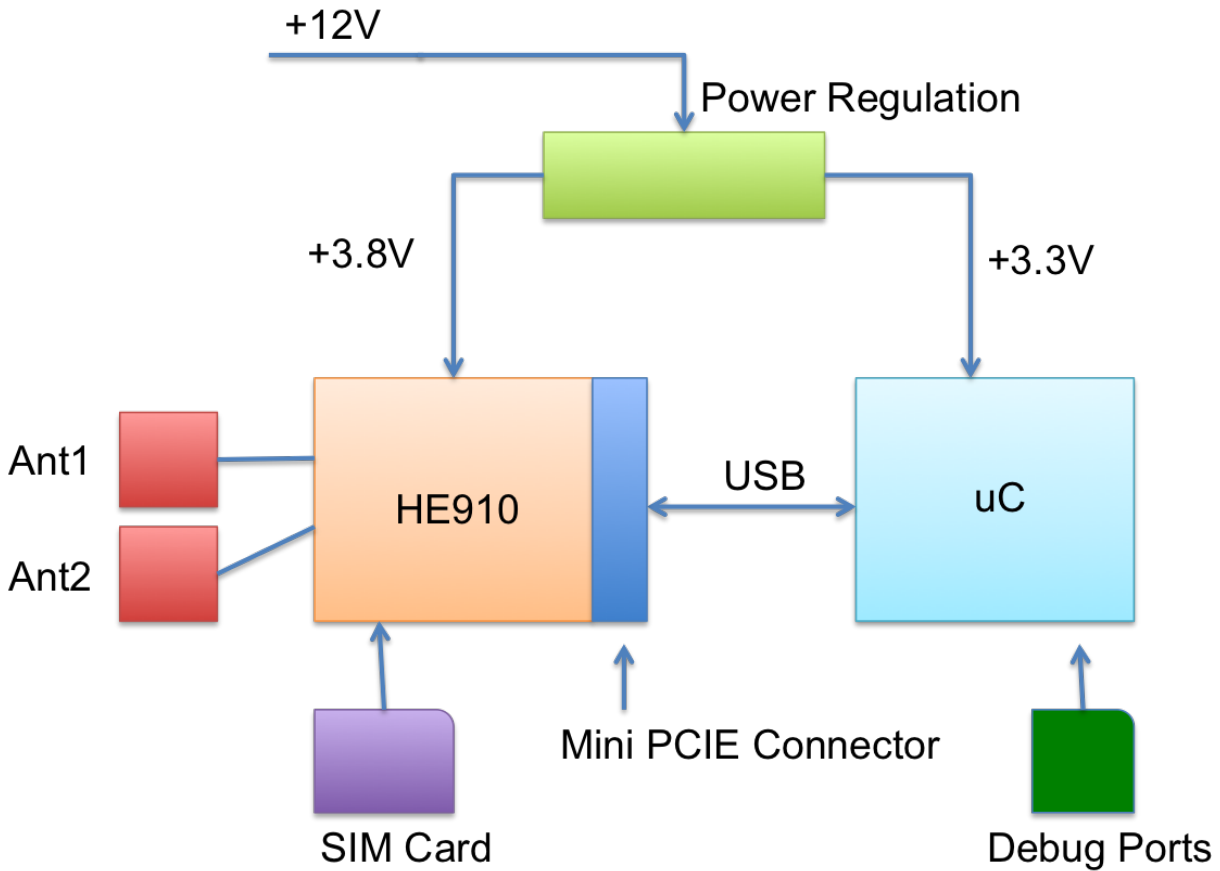


Figure 1. BusTracker System Block Diagram

4.2 Subsystem and Interface Requirements:

Subsystems:

Power

- A dual output voltage regulator will need to be used to provide both 3.3V and 3.8V.
- Other passive components will need to be placed accordingly to ensure correct output

Microcontroller

- Set up debug ports correctly so that the PIC can be programmed
- Connect passive components such as decoupling capacitors correctly
- Connect USB data lines for communication with HE910 chip

HE910

- SIM Card has unique ID for chip
- Antennas get good reception for proper communication

Programming

- GPS locations pulled from HE910 and sent to web server in real time when bus is moving
- Program microcontroller appropriately, communicates with HE910 through AT commands
- Web application allows user to see bus location and how far bus is from the next stop

4.3 Future Enhancement Requirements

We only plan on making one system, but the idea behind this product is that an entire fleet of buses would be equipped with them so that a user can see where all buses in a city are located at any given time. This would require our web server to be able to handle all the data coming in and distinguish which bus is which. Another improvement could be adding a user interface on the box so that the bus driver could be informed of useful information, such as if a road closed or if they were running behind. We could also include a backup battery in the future so that the system could transmit data even when the bus was off in a lower power mode.

5 Networking considerations

The system itself will mainly utilize cellular networking and MQTT for networking purposes. A Telit HE910 module will be used to transmit GPS data over cellular to a server utilizing MQTT. A web application integrated with the Telit platform will be used to access the GPS tracking data.

5.1 Connection Considerations

This device will be directly connected to cellular and GPS networking protocols. The Telit HE910 has integrated cellular and GPS transmission capabilities. A SIM card, most likely utilizing a minimal plan from AT&T will be used to enable cellular communications. The Telit platform will be used to build and modify the server, database and interfacing web application.

5.2 Data Flow

The device will transmit GPS data over a cellular network using MQTT. A server will be involved to store and manage the transmitted data. MQTT APIs will be used to stream, collect and integrate the data. The server will require a database so that GPS locations and timestamps can be stored. Google map APIs may also be used to integrate current GPS locations to the status of a bus's progression on its route. The amount of data would be very low as the only data being transmitted would be bytes of GPS data. When the bus is on, GPS data will be transmitted every 30 seconds.

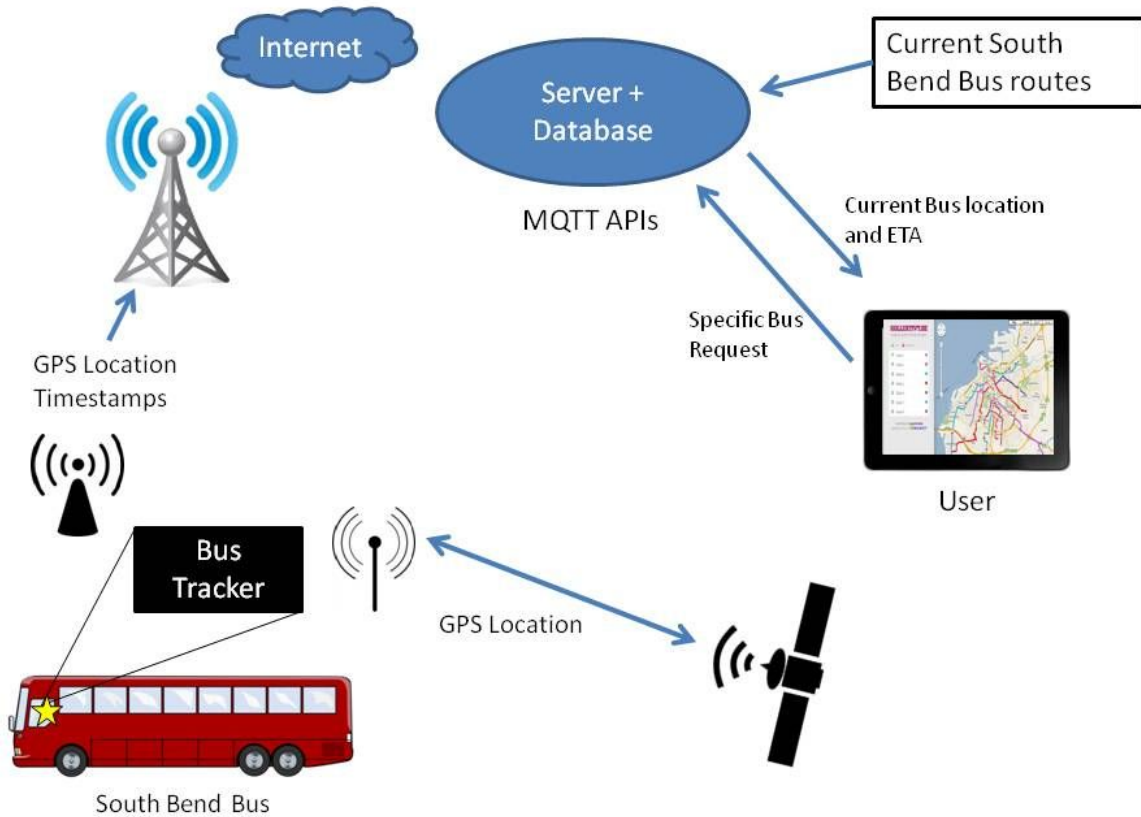


Figure 2. Data Flow Diagram

5.3 HE910

For the networking purposes required, the team will be utilizing a Telit HE910 module. This device has onboard GPS and cellular networking capabilities. The module does have GPIO pins available, but a microcontroller will still be required in order to execute the AT commands. The Telit programming APIs will be used to build and integrate the web application. The Telit APIs are embedded into the Telit HE910 module and already utilize MQTT for data management and integration, thus making it the more efficient option to program the device.

5.4 Protocol for Data Exchange

The system will utilize GPS, cellular and MQTT protocols. AT commands will be used for GPS protocols to access GPS locations and receive timestamps. Cellular and MQTT protocols will then be used to transmit the GPS data to the web server and database. As a user requests the information about a bus from the web application, MQTT will be used to poll the bus location information from the server.

6 High Level Design Decisions

HE910

In order to design a device that tracks the location of a bus, a GPS locator and connection to a cellular network are necessary. The Telit HE910 offers both the GPS location data and a cellular connection; therefore, it was the best choice for the requirements of this project.

Power

To supply power to the GPS tracking device, we will be implementing a cigarette lighter adaptor plug-in option. The cellular and GPS chip HE910 requires 3.3V and 3.8V, so a dual output voltage regulator will be needed to provide both voltage levels. In order to implement the battery and power adaptor, other passive components will be required.

Microcontroller

For this design, a PIC microcontroller will be used. A microcontroller will be required to communicate with the HE910 over a serial interface, and built in USB capability is required. Since the PIC microcontroller offers both of these, it was chosen.

Software

This project will be fairly software intensive. In order to fulfill the requirements for showing live bus locations, a web application will need to be built. To build the web application, Python and Telit's integrated APIs will be used. To interface with the PIC microcontroller, the Microchip IDE will be required, most likely the MPLAB X IDE.

7 Open Questions

1) Interface issues

Because the theme of the project is internet of things, we have to use several protocols to connect with various devices. Communication is the main issue in our project. We are uncertain about the ability to interface between Telit HE910 and server. We should understand how to utilize MQTT protocols to deliver data from Telit HE910 to the server.

2) GPS coordinate data collection

We are unsure about how accurate the GPS data tracks the bus location. Since the bus location changes fast, the GPS tracking system should update the data frequently. If not, it might not locate the coordinates accurate enough. Also, poor reception can happen in bad weather that can totally degrade the system. Time synchronization should be done so that no delays should occur between the Telit HE910 module and the server.

3) Building web application

Our team should learn how to develop mobile and web applications. The challenge is the communication with the server where the data is stored. The application should display estimated arrival time as well as the current bus location. It should consider the bus route and traffic to calculate precise arrival time.

4) Demonstration

The question that should be addressed is how to demonstrate the performance of the bus tracking system. One way we came up with was to install the tracking system in a car and run it around the Notre Dame campus. We can demonstrate the changes in car location and estimated arrival time to the destination.

8 Major Component Costs

Telit HE910 - \$120

AT&T minimum plan - \$20/month

Voltage Regulator - \$5

Microcontroller - \$5

PCB Fabrication - \$50

9 Conclusions

The primary goal of this project is to facilitate the smart use of public transportation via bus tracking system. The bus tracking system incorporates GPS data to provide the location of the bus and estimated arrival time. Mobile and web application will display such information for the users. This system poses significant interface challenges. We have to resolve any communication problems of transmitting and receiving data. We will start our project by demonstrating the GPS location of Telit HE910. In the next semester, we will work towards delivering GPS data to a server utilizing MQTT. The overall performance of the bus tracking system will be determined by how accurate it can pinpoint the bus location and how precise it can estimate the arrival time. We hope that bus tracking system can enhance the convenience in using public transportation and help make South Bend a smart city.

References

<http://www.telit.com/products/product-service-selector/product-service-selector/show/product/he910-mini-pcie-1/>

<http://www.telit.com/products/product-service-selector/product-service-selector/show/product/evk2/>