Design and Implementation of an IP-based Wireless Trolling System

Abstract

In this paper, the design and implementation of a low cost (around $350) Internet Protocol (IP)-based wireless trolling motor system is presented, as the output of the class project of “Wireless Networks and Mobile Computing” performed in South Dakota State University. Typical wireless trolling motors (non-IP-based) can cost upwards of $1,500, for high end models, and communication range is very limited. Corded versions can usually be purchased for around $300, but these wired trolling motors can be cumbersome.

In this design and implementation, an IP-based wireless communication channel was established between a Minn Kota trolling motor and a computer’s WiFi card. A wireless Gumstix and a designed circuit were used to control the motor. An Ad-Hoc connection enabled the Gumstix and computer to communicate with each other. Socket programs were then designed for the Gumstix and the computer, and a command string was sent across the network for each special action. The commands were sent from the computer via a Graphical User Interface (GUI) which was designed to represent a dashboard for a boat and included a throttle and turning capabilities. Google Maps was also included in the GUI.

Drew Waltner, Wei Wang and Mark Kolecka
Electrical Engineering and Computer Science
South Dakota State University
Brookings, South Dakota 57006
djwaltner@jacks.sdstate.edu wei.wang@sdstate.edu
mrkolecka@jacks.sdstate.edu

Acknowledgement
This project was supported in part by NASA EPSCoR grant NNX12AI13A on wireless research and education.

1. Introduction

A trolling motor is a secondary propulsion system used on a boat to propel it when the main motor is turned off. Trolling motors are used mainly in the fishing technique called “Trolling”. Trolling involves moving fishing lines through water from a boat at a relatively slow pace (*Trolling*, 2013). In the Midwest trolling is used to catch perch, crappie, walleye, trout and salmon. This is most commonly done in slow moving water so that the speed can be precisely controlled.

Trolling motors typically are an all-in-one unit that has a controller, motor, and propeller. They are most commonly attached to the bow of the boat and the propeller is lowered into the water when in use. Trolling motors are electric and therefore powered by batteries in the form of 12, 24, or 36 volt (Williams, 2008). This is accomplished by using common 12-Volt batteries and then putting them in a series, as need be, to acquire the correct voltage. The trolling motor that was used in this project was a Minn Kota bow mount 12-Volt Turbo Pro 812, with 42 pound thrust. The controlling unit for this trolling motor is via a foot pedal.

High end trolling motors most commonly have some type of wireless capabilities, but for older and lower end trolling motors the control unit is via a foot pedal controller. The newer trolling motors with these wireless capabilities can cost upwards of $1,500, which is not in the price range for a vast majority of fishermen ("Minn kota motors"). So therefore, the majority of the market belongs to the older and lower end systems which can commonly be found on Ebay or Craigslist for around $350. The foot pedal controllers only have the capabilities of variable speeds and turning. This corded style of controller can sometimes be difficult to use when fishermen are moving around in the boat and keeping track of multiple fishing lines.

A solution for these older trolling motors is to develop a control unit that can attach to the trolling motor and provide wireless communication at a low cost. This would allow the user to move around the boat freely and not worry about getting tangled with the controller cord.

After weighing the price difference and the functionality of different trolling motors it was determined that designing our own wireless trolling motor controller would be very beneficial. For the development of the project, our system would use wireless Gumstix, which uses a Wi-Fi connection and would act as a server ("Gumstix"). Then a socket program would be implemented to control the I/O pins, which are connected to the trolling motor. By using the Gumstix, an IP based solution would be used to control the trolling motor. Therefore there would be a variety of different devices that the client side program could utilize. These devices include smart phones, laptops or any other device that has the capability of Wi-Fi. One could even use a non-activated smart phone so that it could just stay with the trolling motor. This device would then become the controller for this trolling motor. This controller would consist of a Graphical User Interface that can send commands to the Gumstix using the Wi-Fi connection via an Ad-Hoc network. An Ad-Hoc connection communicates to other devices without the need of a centralized base station (Peterson & Davie, 2012).   By using a GUI, other features could be incorporated into the controls, such as Google Maps.

2. Problem Formulation

The designed controller will be able to control a trolling motor wirelessly from anywhere in the boat. This is easily accomplished as an average Wi-Fi connections range is approximately 50 meters, which would clearly suffice any fishing boat (Zoltek, 2000). A boat can be a cluttered place, especially when there are multiple fishermen and they all have their own equipment. As one can imagine, with everyone's fishing tackle scattered all over the place it can sometimes be difficult to walk around to access the trolling motors controller. Not having someone by the controller at all times could be hazardous to the boat and fishing equipment. For example one may not be able to access the controller before hitting a rock or breaking their fishing line from a snag. Therefore a wireless control system would be beneficial because one would have it on them at all times and be able to quickly react to a hazard.

The controls would have a simple user interface that is operated through the wireless device, which is able to react to a command at a high speed. The controller will send signals from the wireless device via a client side socket program and will then be received on a server side socket program on the wireless Gumstix. This will be accomplished by using an Ad-Hoc connection, which enables us to quickly setup a relatively cheap connection that is capable of being implemented on any Wi-Fi device (Geier, 2002). The Gumstix will be connected to the trolling motor through a designed circuit and once it receives the signal it will interpret it and change the various I/O pins accordingly. The system must be able to interpret the user's desired action and then send the command across the Ad-Hoc network to the Gumstix faster than the operator controlling the motor, making the controller a real-time system (Real-time systems, 2002). For this to happen the signal must be good and the program must be efficient and not have much overhead. The wireless connection must also be free of any interference that would cause degradation of signal or lost packets.

3. Trolling Motor

A Minn Kota bow mount 12-volt Turbo Pro 812, with 42 pound thrust trolling motor was selected to be used in the project. This motor is a regular trolling motor that attaches to the bow or front of the boat and has a corded foot control. This control has a variable speed setting and left and right turning controls. The motor and control are shown in the figures below.

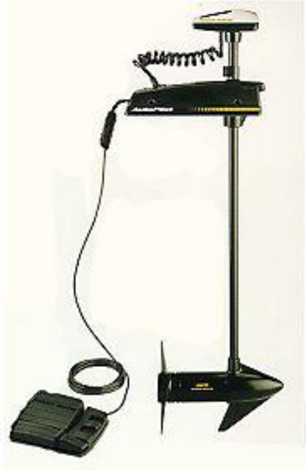


Figure 1: Minn Kota Model 812 Trolling Motor

Figure 1 shows the trolling motor with the corded foot pedal control. The motor runs on a single 12-volt electric battery and is a bow mounted unit.

Figure 2: Minn Kota Foot Pedal Control

Figure 2 is an up close shot of the foot pedal control. Shown on the right side is the variable speed setting and on the left are the turning controls to steer the motor and the throttle switch.

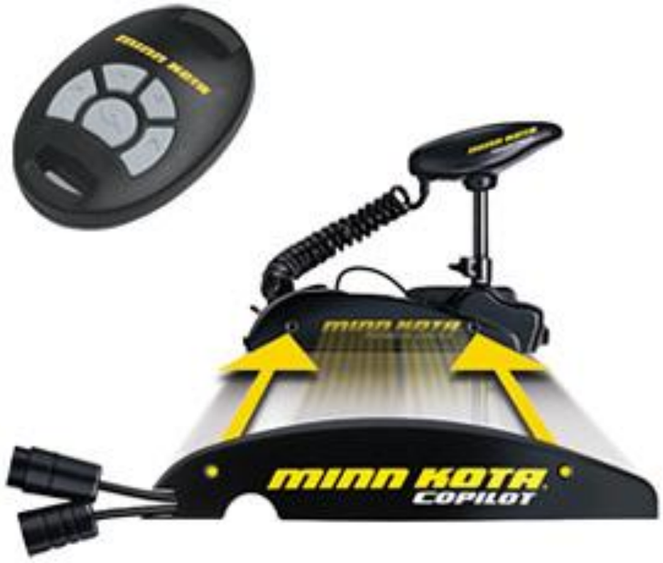


Figure 3: Wireless Minn Kota Trolling Motor

An example of a wireless trolling motor is shown above in Figure 3. The motor is virtually the same, except for the remote control being much smaller and is operated by hand.

4. Development Tools and Components

An integrated development environment or IDE was needed to design and make the GUI. Visual Studio 2010 was chosen for this project. Visual Studio allowed colorful illustrations to be put into the GUI and the addition of the fishing theme to it.

C# was chosen as the language for the GUI and client side socket program. This allowed for the creation of the GUI in Visual Studio and the ability to embed Google Maps into the application. The Gumstix runs a Linux environment, so C++ was chosen as the programming language for the server side socket program.

The wireless Gumstix was supplied in class and has many features including its own Linux operating system and a Wi-Fi antenna. Connections on the Gumstix include USB, Ethernet, various I/O pins, and a power supply ("Gumstix").

Other components used in the project were a power supply, bread board, car battery, and a designed circuit. The power supply is used to power the designed circuitry and could eventually be eliminated with the addition of a voltage regulator. The bread board is needed for the designed circuit. The car battery is needed to power the trolling motor.

5. Design, Coding and Circuit Implementation

The first part of the project was to create a GUI. This application needed the functionality of being able to take input from the user and turn it into commands that the embedded program on the Gumstix could understand. A screenshot of the GUI is shown below in Figure 4. It can then be seen that there are three main inputs from the user. These include commands to turn left, turn right and powering on or off the motor. There is also an additional feature of Google Maps. This was embedded into the application and allows the user to see the lake that they are at and determine their relative position.

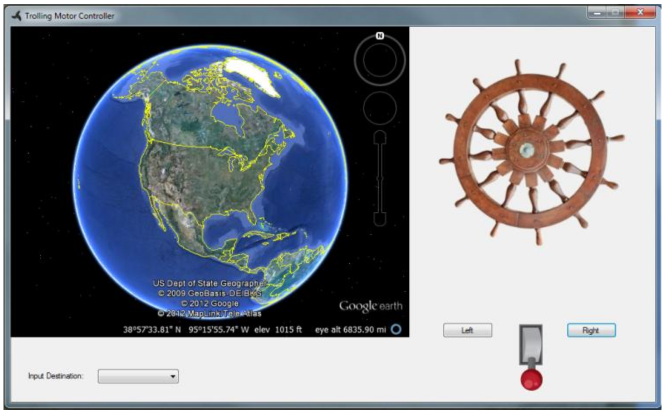

Trolling Motor Controller
N
US Dept of State Geographer
© 2009 GeoBasis-DE/BKG
© 2012 Google
© 2012 MapLink/Tele Atlas
Google earth
38°57'33.81" N 95°15'55.74" W elev 1015 ft eye alt 6835.90 mi
Left
Right
Input Destination:

Figure 4: Graphical User Interface

When implementing the turning capabilities, using the GUI, there is a command sent when the user clicks the turn button and then a command is sent when the user releases the turn button. For example, if the user wanted to turn left they would click the turn left button and then a command to turn left would be sent over the wireless network to the Gumstix. While the user is holding down the button, the motor would be turning in the desired direction, but as soon as the user releases the button the GUI would send the command to the Gumstix to stop turning.

The command to power on or off the motor works in a slightly different way. When the user wants to turn on the motor they will input the command to the GUI and it will send the command to the Gumstix. When the user releases the switch, the motor will stay turned on. The motor will not turn off until the user switches the throttle off, then the command to turn the motor off will be sent. The GUI also has the capabilities to steer the motor while not changing the speed or throttle.

The next part of the project was setting up the Ad-Hoc wireless connection. This involved booting up and running the wireless Gumstix, which can be configured through a USB connection when connected to a computer. This was done by having the Gumstix create the wireless Ad-Hoc network and setting the Gumstix IP address to a fixed address. The network would then be visible to the computer and therefore was able to be connected wirelessly by this newly created Ad-Hoc network. Once connected to the wireless network, the computer and Gumstix can easily communicate.

After a successful connection was made, client and server socket programs had to be developed. The client side was to be designated as a computer and would send commands across the network to the server socket program located on the Gumstix. The client side would be running in a windows environment, while the server side would be running on the Gumstix's Linux environment.

The client socket program would be integrated with the GUI. Like mentioned above, the GUI would interpret commands from user's input and send its corresponding command across the network through the socket. These commands would be in the form of a string and would contain a specific combination of three numbers of either one or zero for each spot.

A zero would mean that the command is to stop turning or power down. A one would mean to turn or power on the motor. Each position in the string had a specialized meaning. The first position would mean to turn right, the second position would mean to turn left, and the third position would power on or power off the motor. The different commands that can be transmitted across the network can be seen in Table 1.

STRING	COMMAND
100	Right Turn
010	Left Turn
001	Power On Motor
101	Turn right while motor is running
011	Turn left while motor is running
000	Motor is powered off

Table 1: Commands List

After the client side socket program was set up, the server side needed to be developed. The server side program works by setting certain output pins to either high or low. When a pin is set to high, the motor would turn on or steer depending on which pin was set. When the pin was set to low, the motor would turn off or quit steering depending upon the command.

The server program would turn these pins either high or low by opening certain files and changing the contents of those files. The program would interpret which pins to set high or low based on the string that was passed over the wireless network.

Pin #	Wire Color	Description
E	Black	Ground
C	Brown	Speed
A	Red	+12 V.
B	Orange	Switched 3.9 V.
F	Yellow	Steering Motor
G	Green	Steering Motor
D	White	On/Off

Table 2: Pin Out of Trolling Motor

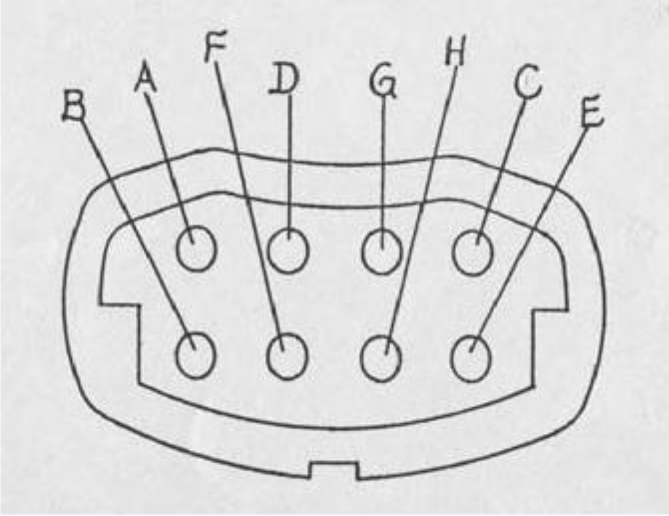


Figure 5: Pin Out of Trolling Motor

A schematic had to be set up so that the I/O pins could control the trolling motor in the desired way. This was needed to be done because various pins had to be connected to each other so that an operation would happen. Table 2 and Figure 5 show the pin out of the trolling motor control unit. After some in depth research it was determined how the pins function and therefore how to control the unit. To turn left pin F has to be pulled to pin A (12 V), and pin G has to be pulled to pin E (Ground). The opposite is done for turning right, pin G is pulled to 12V and pin F is pulled to ground. Determining the speed functionality was more complicated because to turn on the motor pin D has to be pulled to pin A (12 V). After the motor is on the speed has to be adjusted. It was determined that this was done by connecting pin C to pin B. From this configuration, it was determined that four relays were needed to switch between the different settings and therefore, four I/O pins would need to be used from the Gumstix.

After it was determined what was needed for the correct functionality of the control unit, the circuit was designed and constructed. The schematic for the circuit can be seen in Figure 6.

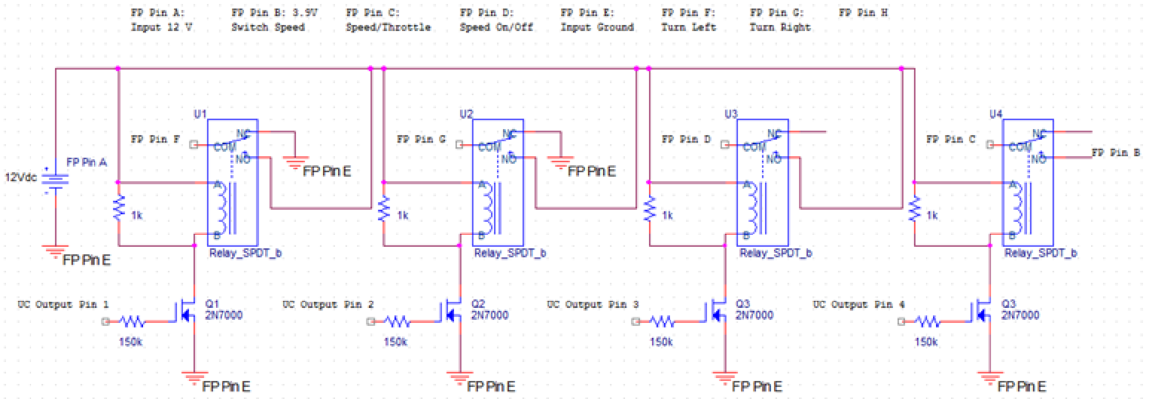


Figure 6: Designed Circuit Schematic

6. Results and Discussion

Once the circuit was set up, Figure 7, and communication was established, the trolling motor was connected and the functionality was tested as seen in Figure 8.

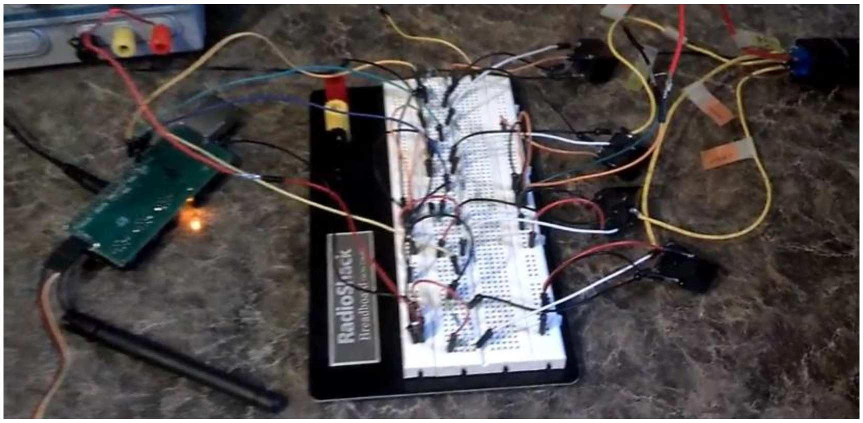


Figure 7: Designed Circuit

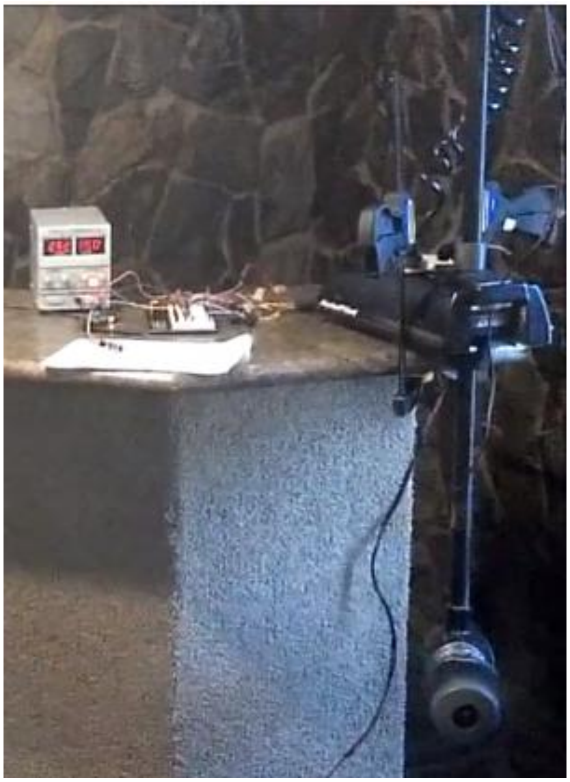
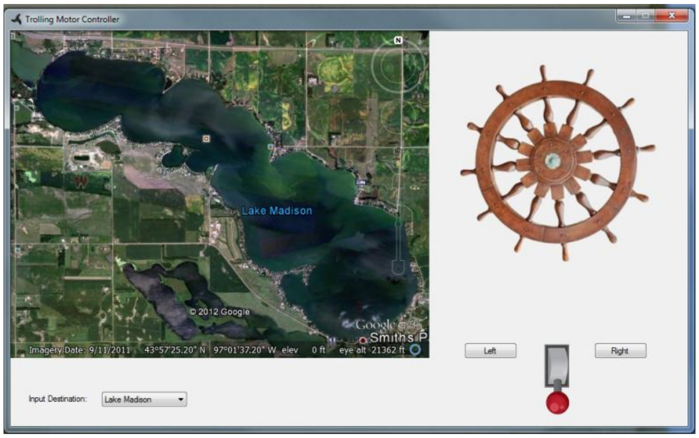


Figure 8: Final Deliverable

After it was determined that the project was fully functional, extra functionality was added to the GUI. A dropdown menu was added so that the user has the option of choosing between different lakes for Google Maps. The locations of these lakes are hard coded into the application so only a few options around the area were done. An example of a user selecting Lake Madison can be seen in Figure 9.

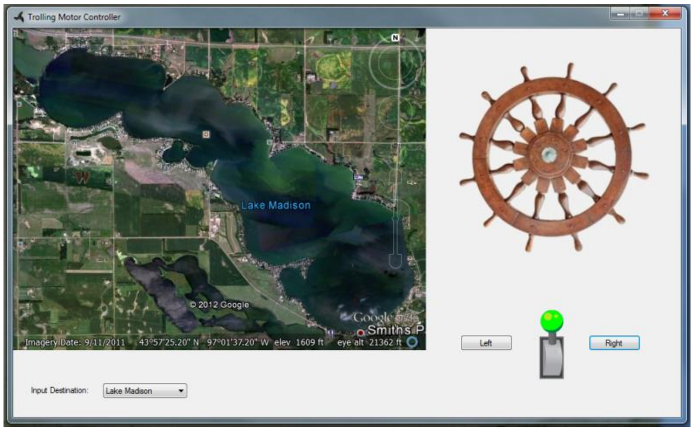

Trolling Motor Controller
N
Lake Madison
© 2012 Google
Google
Smiths P
Imagery Date: 9/11/2011 43°57'25.20" N 97°01'37.20" W elev 0 ft eye alt 21362 ft
Left
Right

Input Destination: Lake Madison

Figure 9: GUI with Lake Madison

As seen above, once the user selects the lake, Google Maps will load the lake onscreen. The user will then be able to zoom in and examine the map. This can be very useful to the users because the fishermen could see the lake as a whole and possibly determine where the best places to fish by locating hidden inlets where others might not be able to look.

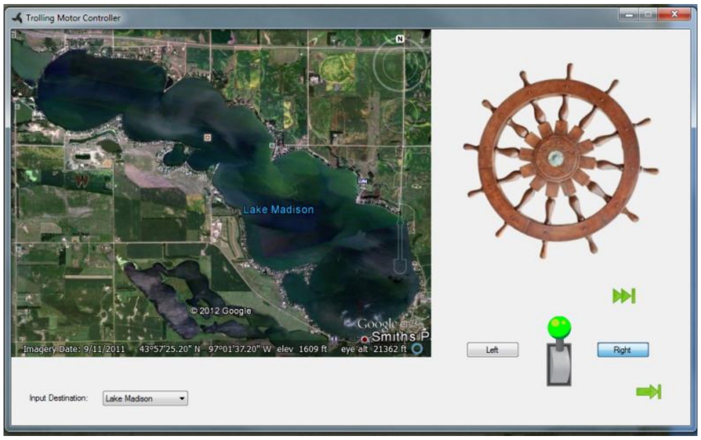
Other features were also added to the GUI to help enhance its appearance and make it user friendly. One of these features is the power switch, shown in Figure 9, between the left and right turn buttons. This switch will remain red when the motor is powered off. To turn on the motor, the user will simply click on the switch and it will turn green and flip up to indicate the motor is on. A screenshot is shown below in Figure 10 to show this.


Trolling Motor Controller
N
Lake Madison
© 2012 Google
Google
Smiths P
Imagery Date: 9/11/2011 43°57'25.20" N 97°01'37.20" W elev 1609 ft eye alt 21362 ft
Left
Right

Input Destination: Lake Madison

Figure 10: Motor Powered On

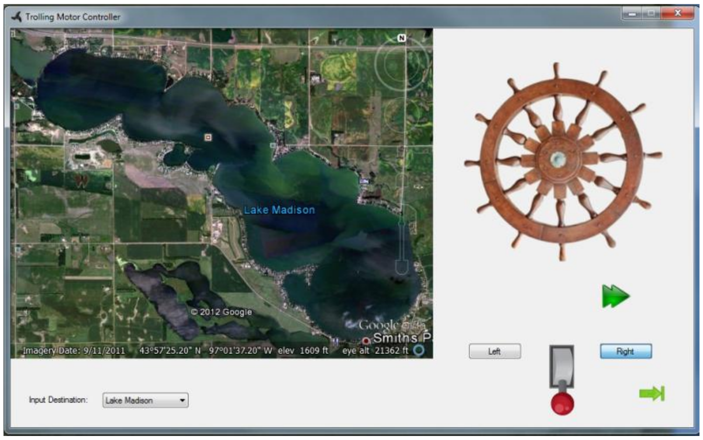
Another feature added to the GUI was sequential and blinking directional arrows. These arrows will show when the user presses down one of the turning buttons. They will remain flashing until the user releases the button to stop turning. The steering wheel also spins in the direction that the user chooses to turn. Shown in Figure 11 below is the user choosing to turn right with the motor powered on.


Trolling Motor Controller
Lake Madison
© 2012 Google
Google
Smiths P
Imagery Date: 9/11/2011 43°57'25.20" N 97°01'37.20" W elev 1609 ft eye alt 21362 ft
Left
Right

Input Destination: Lake Madison

Figure 11: Turning Right

Shown above are the flashing arrows indicating the direction chosen. Also happening simultaneously is the steering wheel spinning to the right, but is unable to be viewed in the screenshots.


Trolling Motor Controller
Lake Madison
© 2012 Google
Google
Smiths P
Imagery Date: 9/11/2011 43°57'25.20" N 97°01'37.20" W elev 1609 ft eye alt 21362 ft
Left
Right

Input Destination: Lake Madison

Figure 12: Right Turn

One last screenshot, Figure 12, indicates the user steering the motor right while the motor is powered off.

There were many difficult obstacles to overcome during this project. The first one was determining if it was even possible to control the trolling motor without any specialized equipment or knowledge of the previous design. Once it was found that it could be done with a relatively simple circuit, it was decided to go through with the project.

The next step that was frustrating was setting up the network communication with the Gumstix. There was a decent amount of troubleshooting in this area, but after it got resolved everything went smoothly with the Gumstix.

The last complicated part was communicating with the GUI and Gumstix. This was done by using the "sys/socket.h" class in Linux and then using a socket to communicate to the GUI from the Gumstix. After this two-way communication was set up it was made so that the Gumstix was blocked until it received a command from the GUI.

After this project was completed, the prices of the original system were compared to our designed system. If this were to be commercially made one would have to make sure it is able to run off of batteries. This could be done with the addition of a voltage regulator which maintains the voltage of the batteries to a desired level (*voltage regulator,* 2008). Since only a low current voltage regulator would be needed, because the components of the circuit do not take a high amount of current, the price of this would be under $5. It also would be in our best interest to not use the Gumstix for the final product but instead just use a Wi-Fi module along with a microcontroller. This would be beneficial because the Gumstix with the Wi-Fi module costs roughly $200 whereas a PIC microcontroller and a Wi-Fi module can be obtained for around $35. The only other parts that would be needed would be the resistors and the relays which can be all purchased for less than $10. Therefore, it can be seen that to make a common wired trolling motor wireless it would cost around $50.

7. Conclusion

The goal of this project was to design and develop a low cost wireless controller that could attach to a standard Minn Kota trolling motor. This was done using an Ad-Hoc network for bi-directional communication. The control module included a GUI running on a computer that communicated to the Gumstix, which was connected to the trolling motor through a circuit. I/O pins were used on the Gumstix to control different settings for the trolling motor, which would allow the user to turn and toggle power to the motor. The Gumstix interpreted signals from the GUI and then adjusted the I/O pins accordingly.

References

[1] Trolling. In (2013). Merriam Webster. Retrieved from http://www.merriam-webster.com/dictionary/troll

[2] Williams, M. (2008). Trolling motors 101. Retrieved from http://www.flwoutdoors.com/fishing-articles/tech-tackle-reviews/148958/trolling-motors-101/

[3] Minn kota motors. (n.d.). Retrieved from http://www.minnkotamotors.com

[4] Gumstix. (n.d.). Retrieved from https://www.gumstix.com/

[5] Peterson, L. L., & Davie, B. S. (2012). Computer networks, fifth edition: A Systems Approach. (5th ed., p. 134). San Francisco: Morgan Kaufmann Publishers.

[6] Zoltek, T. (2000). Comparing bluetooth and ieee 802.11. Microwaves & RF, 26-28.

[7] Real-time systems. In (2002). The McGraw-Hill Publishing Companies, Inc. Retrieved from http://encyclopedia2.thefreedictionary.com/Real-time systems

[8] Geier, J. (2002). Understanding ad hoc mode. Retrieved from http://www.wi-fiplanet.com/tutorials/article.php/1451421

[9] Voltage Regulator. In (2008). Encyclopedia Britannica Online. Retrieved from http://dictionary.reference.com/browse/voltage regulator?s=t