Samuel P. Kelly, Latif Masud, Nathan Young, Ziji Guo, Tyler Bohlke

EE 491  DEC 14 #03

Submitted: 8 December 2014

Bluetooth audio receiver

Final Report

# Problem and Need Statement

Currently, there are a few products that can stream audio from a Bluetooth enabled device to any speaker or audio system. However, most of these solutions are quite expensive, ranging anywhere from 40 to 70 dollars for the conversion module. The goal of our project was to build an alternative to the current competition at a cheaper price with similar product specifications and quality.

# Concept

The final product will provide a way to provide audio output into a stereo or desk speaker system, from a remote device. A Bluetooth connection will provide the means of transmission. A receiver module will be required to receive the Bluetooth transmission and provide an analog audio output. While the hardware and software are both available for purchase, the goal will be to provide a cost-effective competing product. The concept sketch for this concept is shown in Fig. 1 below

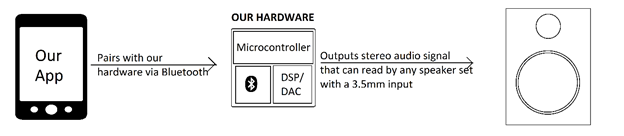


Fig. 1 Concept Sketch

# System Requirements

## Functional Requirements (Hardware)

* Microcontroller (Host Controller)
* 16-bit DAC
* Bluetooth transceiver

## Functional Requirements (Software)

* List Audio Files
* Stream Music
* Connect using Bluetooth

## Non-Functional Requirement

* Aesthetically pleasing
* Cost effective
* Small form factor

## Functional Decomposition

Overall Function

The goal of this project is to create a cost-effective universal Bluetooth to 3.5mm signal converter that would allow users to stream audio from a Bluetooth enabled device to any standard speaker set that has a 3.5mm audio connection. In addition, a mobile app will be developed to provide an intuitive, smooth audio experience with the converter.

Subfunctions

* PCB: Two-layer, easily to build the circuit and decrease the cost.
* Power supply: USB 5V DC supply
* Data transfer: Use User datagram protocol
* Connection: Transfer data from mobile device to receiver module.
* Output analog audio signal: the output goes through the buffer circuit.
* 16 bit audio(@128 kB/s)

## 

## System Analysis

The mobile device will use Bluetooth (3.0 or higher) to pair and connect to the receiver module. This will require the application to control the device’s Bluetooth transceiver and then pair with the receiver module. After the device has paired with the receiver module, the device will then need to send audio output data to the receiver module through this Bluetooth connection. The Bluetooth transceiver within the receiver module will receive the data and output the audio as a PCM (digital) signal. The signal will then need to be converted to an analog audio signal using a DAC, then this analog signal will be sent to a buffer amp circuit before being output from the receiver module.

The peripheral needs of this system include a power supply and a PCB in order to facilitate the functions that occur within the receiver module. The power supply will power the devices held and connected together on the PCB. In addition to the power supply, a voltage regulator and low-pass capacitors will allow for a conditioned power supply to the rest of the board.

On the software end, an application that will function as a media player with direct access to the Bluetooth functions of the mobile device will be needed. It will utilize Android Media Player Class in order to serve that functionality.

## Input/Output Specification

The input to the overall system is a digital audio feed (sent through a radio signal) from a Bluetooth enabled device (as well as a DC voltage from a power supply). The output of the system is a stereo analog audio signal, to be received by a stereo or desk speaker system.

The Bluetooth signal will be received by an RF transceiver-on-a-chip, and the information received will be sent, through a serial connection, to a DAC. The Bluetooth chip will need to communicate with the device it is pairing with in order to carry out the action of pairing before audio may be transmitted from the device to the system.

The DAC will be required to process the incoming serial stream from the Bluetooth Receiver and then provide a 16 bit analog stereo audio output (at line levels), which will then be output to buffer circuits. The buffer circuits will be built using a buffer amp circuit that includes a potentiometer to provide output volume control.

The outputs of the buffer circuits will then be sent through both a 3.5mm headphone jack, which can use a conversion cable to facilitate an RCA connection.

The receiver module requires a power supply. A 5V USB charger should allow for a reasonably low-cost and effective solution. These chargers are ubiquitous in the mobile electronics industry, and a variety of manufacturers have options from which to choose from.

The Bluetooth chip will be required to support Bluetooth Core Specification 3.0 or better in order to allow enough data to be reliably transmitted to receive a 128kBit/s audio signal. Core Specification 4.0 can allow for low power features but will increase the variable costs associated with the Bluetooth IC.

## User Interface

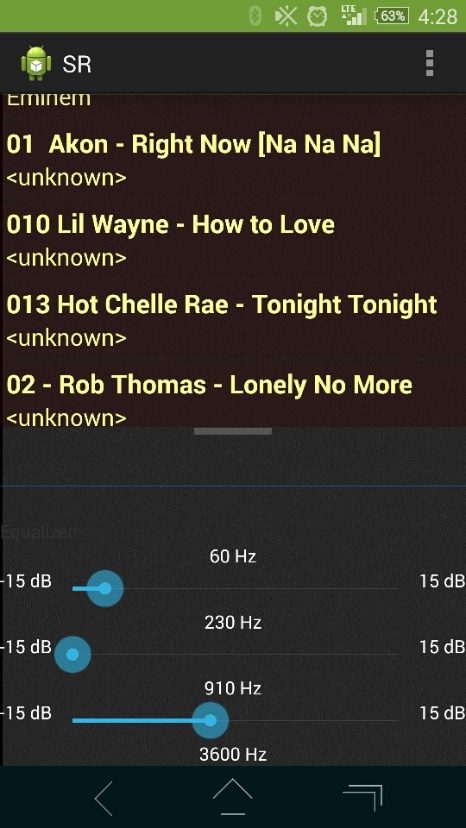
The hardware user interface will have to allow the user to turn the device on and off (a physical switch will suffice) and a Volume control, which will be realized by a passive potentiometer.

The software user interface is the UI of the mobile app. (See Deliverables)

# Deliverables

## Android Application

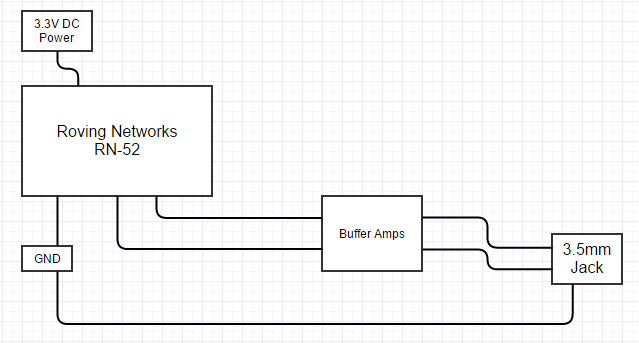
The android application would be used to provide a music player that allows for the user to play audio and pair with the receiver within the same application. This would allow for more ease of use, especially with less technologically savvy types of consumers who wish to not deal with their mobile device’s Bluetooth settings menu.

Android Application

## Proof-of-Concept/Prototype

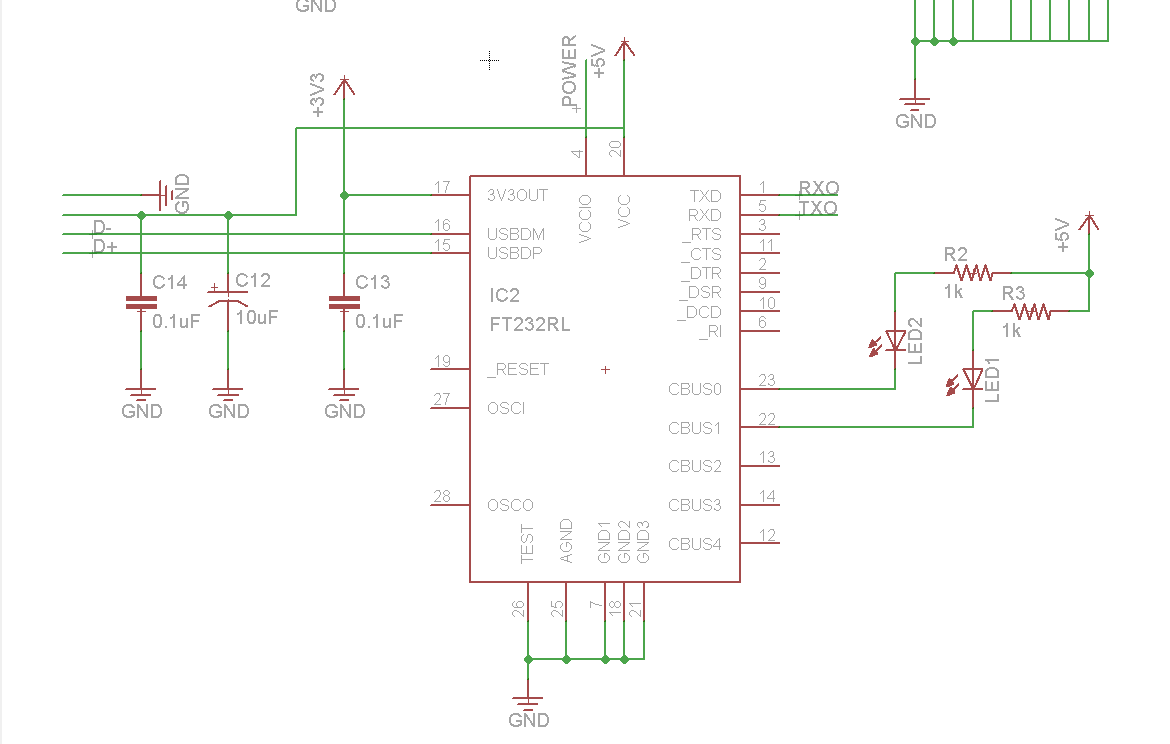
The physical product we have designed and built utilizes the RN-52 chip to provide the Bluetooth transceiver, host-controller, and DAC functionality. We were still required to provide a buffer circuit, volume control, power supply/regulator, USB connectivity, and a power switch.



Proof of Concept Block Diagram

## USB Connection

Provides a USB connection that allows for powering, interfacing, and programing the RN-52 as well as the other devices found in our circuit. Provides USB to serial UART interfacing and signal conditioning.



USB interfacing schematic

## Power

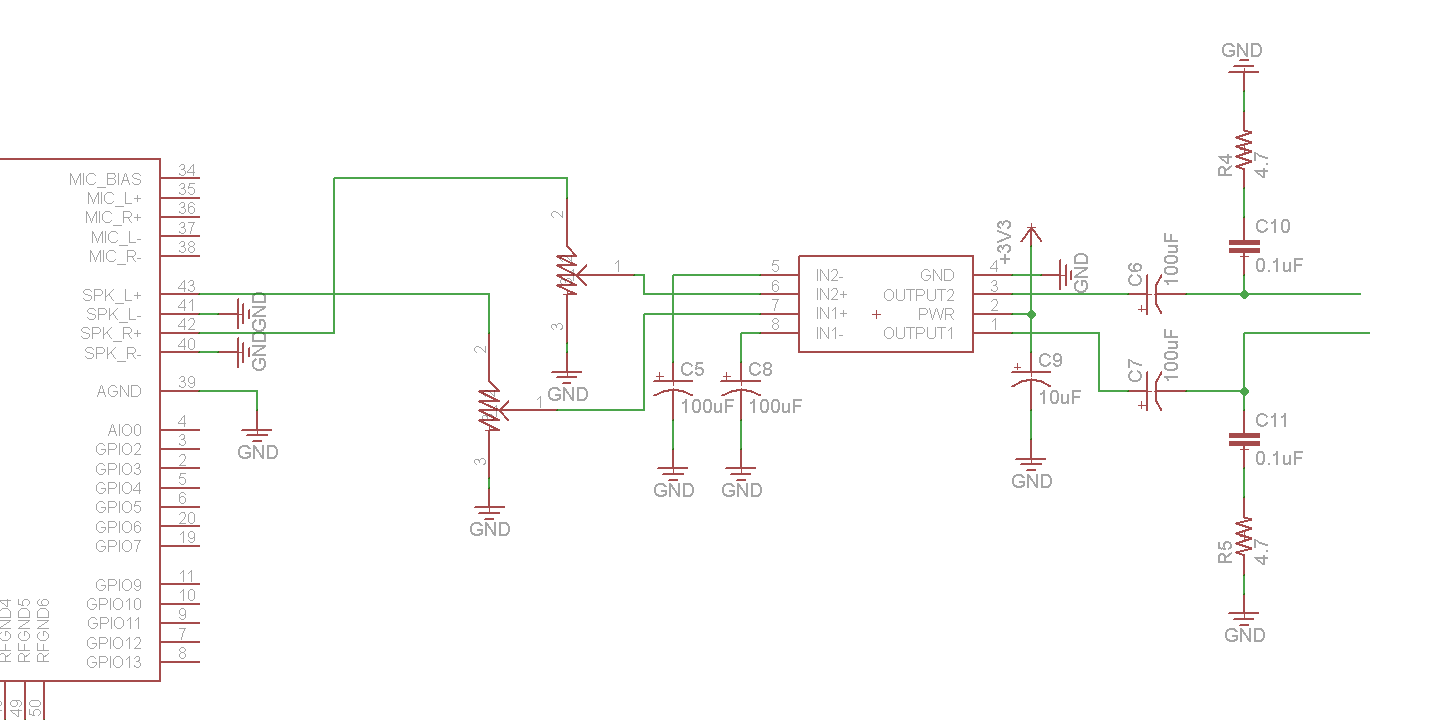
We used a voltage detector and a voltage regulator to provide a 3.3V supply from a 5V USB input.



Power Regulation Schematic

## Audio Output

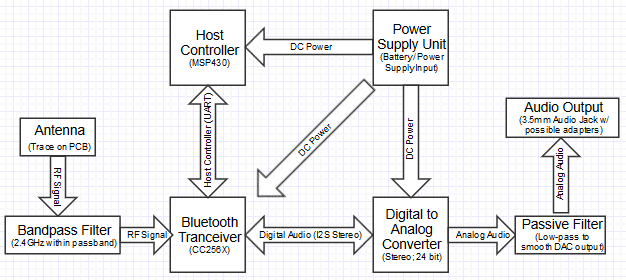
We used potentiometers to adjust volume on both outputs (Left and Right channels). A Buffer Amplifier is used to provide power amplification and to isolate the RN-52 output from the physical output socket.



Audio Output Schematic

## Final Design Concept

The real ending to our design project, given far more time, resources, and experience, would have required purchasing individual devices to provide the functionality of the RN-52. These additions would include an MSP430 Microcontroller as the host controller, a 24 bit DAC, and a CC256X transceiver. We would also be incorporating the required microwave antenna and bandpass filter directly printed on the PCB.



Actual Design Block Diagram

# Testing

We had to test multiple parts of our prototype while waiting for our parts to arrive. This lead to a compartmentalized testing of our prototype.

## RN-52

We prepared a bread-boarded circuit that supported the RN-52 with power and an audio output buffer. This allowed us to test the functionality of the RN-52 outside of the final prototype.

## Pairing with the RN-52

We first tested the ability of the RN-52 to pair with multiple mobile devices. Only twice did we fail to connect, though this was probably due to unobserved parameters of the test. After multiple iterations were performed, it was evaluated that we could reliably connect to the RN-52.

## Testing Audio Output

The need for the buffer circuit was demonstrated by testing the audio quality before and after its addition. Additionally, we conducted the subjective listening test to confirm the quality of the audio output would be satisfactory on the final prototype. This required playback of multiple music tracks, transmitted from multiple devices.

While occasionally we could notice the mild detriment of the 16-bit output (most devices manufactured in the past few years utilize a 24-bit DAC), this issue went unnoticed by most testers. To sum it up, the prototype passed this test.

## PCB Testing

While there was not much to test with the parts still on their way, we were able to confirm that the PCB design and fabrication resulted in perfect connections between every pin. For this test we used a digital multi-meter to confirm that proper connections were made from pad to pad.

# Appendix I: Operation Manual

## Initial Receiver Setup:

1. Plug power supply into a 120V (60Hz) AC Power outlet (for North American use).
2. Connect the USB connector from the power supply to the USB input on the receiver module.
3. Connect desired audio system to the audio output of the receiver module. The output port is for the 3.5 mm headphone connector. Use with systems utilizing RCA inputs will require a 3.5mm to RCA converter cable. (Hint: Older stereos that include a tape deck can be connected to using a 3.5 mm to cassette adapter)
4. **Important:** Turn the Volume knob to the “Min” position for now. This will protect the speakers and stereo from being overdriven.
5. Move power switch to the “ON” position.

## Mobile Device Setup:

1. Open the Android Application.
2. On the settings view, select “Turn On”.
3. Next, select “Get Visible”.
4. Next, select “List Devices”. This will provide a list of devices.
5. Select “RN-52” on the device list. This will command the mobile device to pair with the receiver module.
6. Navigate to the main page and select the desired track to play.

## Final Setup:

1. After you have paired with the receiver module, power on the stereo you have connected the receiver to if you haven’t done so already.
2. Adjust the stereo volume to 25% of the maximum volume.
3. Select “play” on your mobile device.
4. Slowly adjust the receiver volume until you hear the audio distorted, then decrease the volume until the distortion goes away.
5. Enjoy

# Appendix II: Other Versions

## Previous Project

In the beginning, we were assigned a different project altogether. The project in question was to design a better interface for a child with major motor control disabilities. We were excited due to the implications of the project and the capacity to give back to the world.

We initially had some promising ideas, including the recently feasible ‘mind control’ technology that was available in the DIY market. However, the client (including the child’s educators and parents) was looking for a solution that had the customer support and insurance of a larger medical supplies company. In other words, they were looking for a purchasing decision.

While we did not prefer to abandon the project, the lack of design involved meant that we would not be able to get the education we needed from our Senior Design Project. Therefore, we fired our client (respectfully and on good terms).

## Initial Bluetooth Audio Device

Our initial audio product idea was to provide an application and device for streaming audio from a single mobile device to multiple receiver units. This focus had good implications for the live audio market, especially considering the rise of DJs who favor using a mobile device in their live equipment and need to transmit to multiple audio systems within the same venue. Use of such a product would allow for a “wire-free” connection between their audio output and multiple PA systems, monitor systems, and other necessary audio equipment. This would save both hassle and clutter for the user.

Even though the implications were very good, there was one fatal flaw with the system: Bluetooth can only allow for a connection to one device at a time. While we could have constructed a separate unit that had multiple transceivers on it to handle the multiple connections, this seemed too far out of our expertise, time, and resources to reasonably complete within the course.

# Appendix III: Other Considerations

We learned a variety of things while working on this project, usually through trial-and-error method. Some major points are included below.

## Firing a Client

We learned, through a variety of situations, that problems almost always will arise during an engineering project. As stated in Appendix II, our group began EE 491 by trying to make the best out of a difficult situation.

We had to fire the initial client after it was discovered that they were looking for more of a purchasing decision. While such a task could be considered applications engineering, it wasn’t enough to be considered a design engineering project. After trying to find a way that a design service could be applied to the client’s needs, we ended up having to part with them.

## Changing Prototype Parts

Our initial prototyping scope was to build the final implementation as our prototype. However, the parts we were to use for such an implementation required Ball Grid Array Soldering, a capability we simply didn’t have.

To solve the problem, we ended up using a single-chip solution to replace many the BGA components in our prototype. This saved us from having to rely on a third party to solder our BGA components to the boards.

## Buffer Amp Circuit

When we were initially investigating the RN-52, there was an interest in using the direct output of the chip. This was due to the chip having its own DAC and a reasonable power output. While the output did produce a useable signal, we noticed that some uses led to a large current draw. In fact, it was effecting the audio quality on cheaper desk speakers.

To solve this issue, we utilized a buffering power amplifier circuit utilizing an audio op-amp IC. We also incorporated volume control into the buffering circuit with the addition of a stereo potentiometer. We also isolated the RN-52 from the receiver output, gaining protection for the chip.