Eye Can Hear You (E.C.H.Y.)

Group 4

Michael Cooke	**(EE)**
David Griffen	**(CpE)**
Whitney Keith	**(EE)**
Edward Romero	**(EE/CpE)**

Acknowledge: Dr. Larry Chew

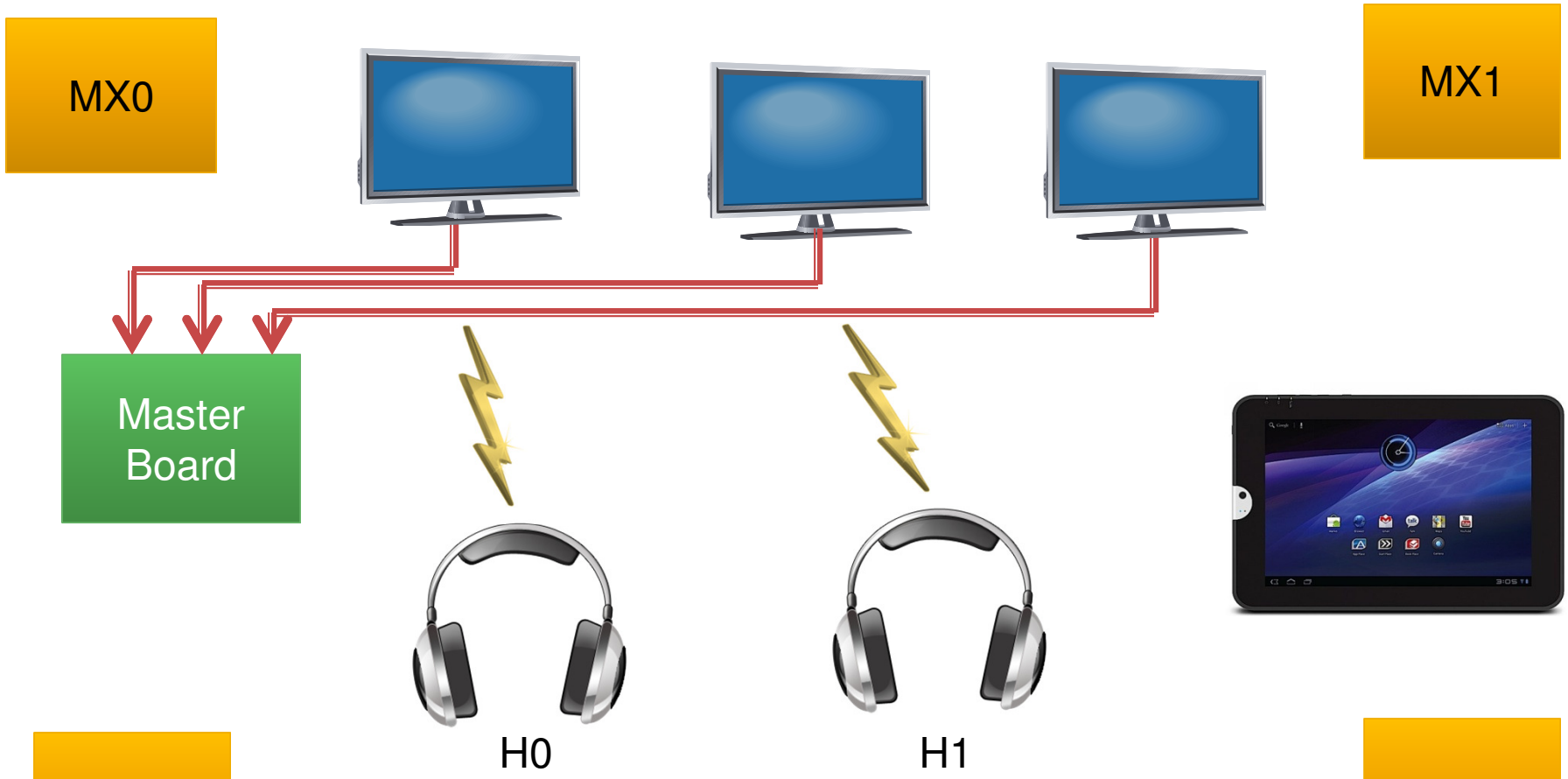
Problem

- One television's audio is broadcasted within a restaurant/gymnasium leaving all other televisions muted.
- Customers are limited to hearing one/no televisions.
- Restaurants/gyms that do have headphone plug in units, limit user mobility and comfort.

The E.C.H.Y. Project

The Eye Can Hear You Project is a network of wireless headphones that receive quality television audio through a user's line of sight.

System Layout



Network Demo: 2 Headphones & 3 Televisions

Objectives

- Efficiency at a low cost
- A high SNR
- Mobility
  - Wireless Headset and Tablet
- User Friendly
  - GUI to determine headphone location
  - Track customer use
- Comfortable and lightweight headset

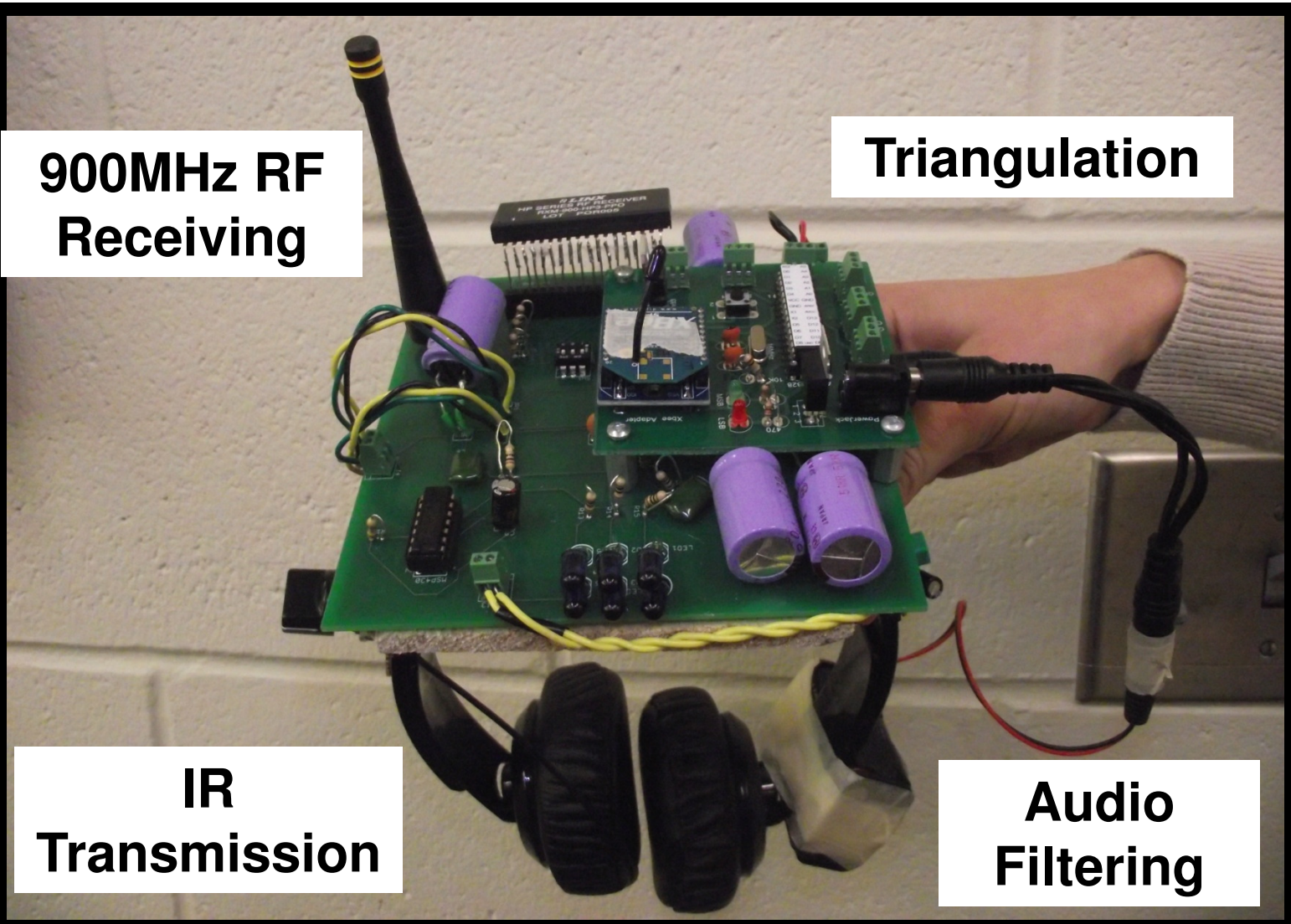
Specifications

Component	Parameter	Design Specification
Headphone	Audio Frequency Bandwidth	500-18kHz
Headphone	Weight	< 1 lb.
Battery	Operation Time	5 hours
Audio Transmission	Range	100 ft
IR detection	Range	15 ft
Triangulation	Accuracy	5 ft
GUI	Update Time	< 5 sec
Tablet Communication	Range	50 ft

Sub-sytems and Communication

- Headphone to Television Identification
  - Infrared LED PWM signal
- Audio Streaming
  - 900MHz Radio Frequency, protocol IEEE 802.11
- Triangulation
  - 2.4GHz Radio Frequency, protocol IEEE 802.15.4
- Tablet
  - 2.4GHz Radio Frequency protocol IEEE 802.15.4

Headphone Layout



Television to Headphone Identification

IR Emitter

Transmits unique 8 bit PWM for each headphone

IR Detector

Decodes PWM into 8 bit ID.

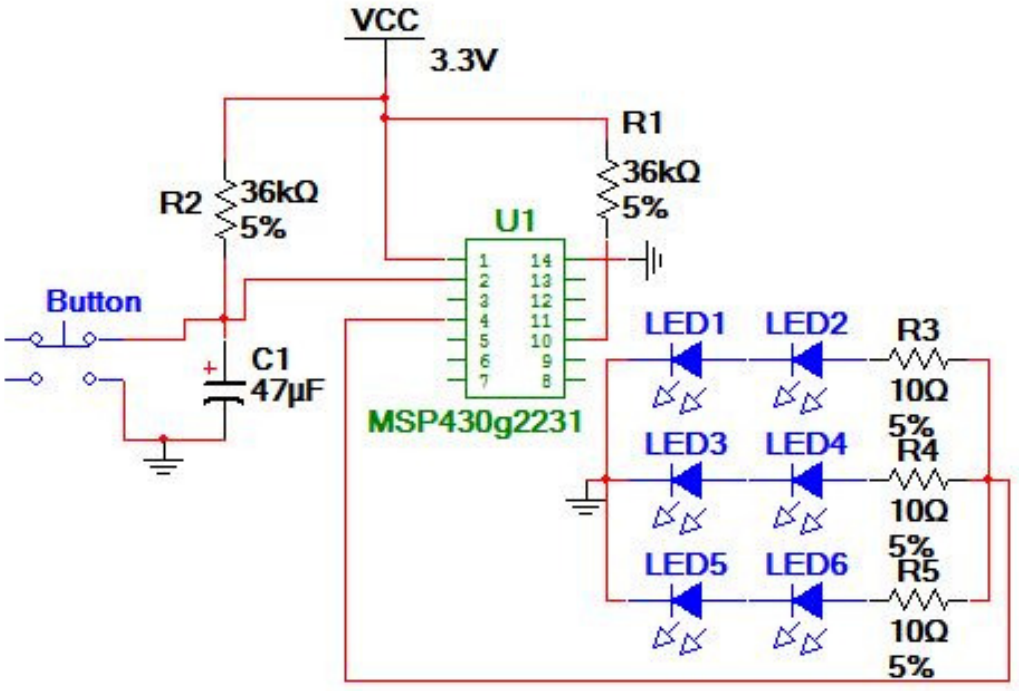
Matches ID to headphone.

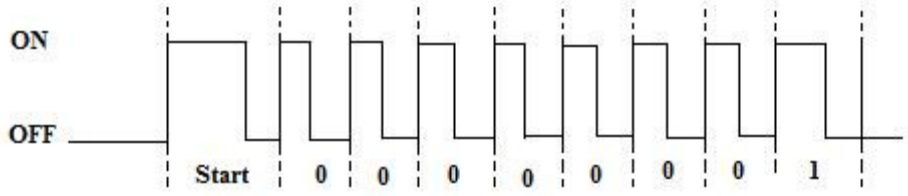
Sends ID to Master Board.

Master Board

Switches TV audio for headphones according to ID from each television

IR Emitter Circuit





Hardware

Vishay TSAL6200

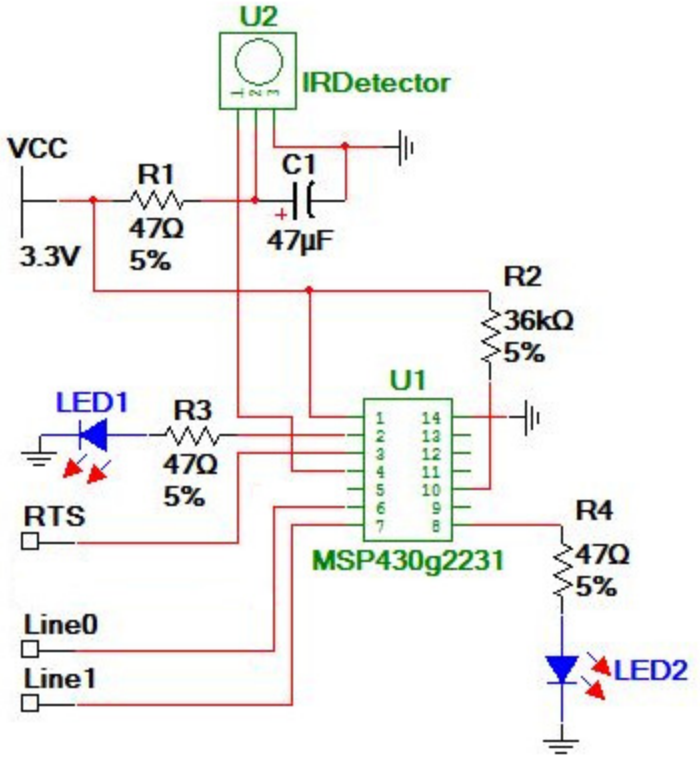
- +/- 17 deg. angle of intensity

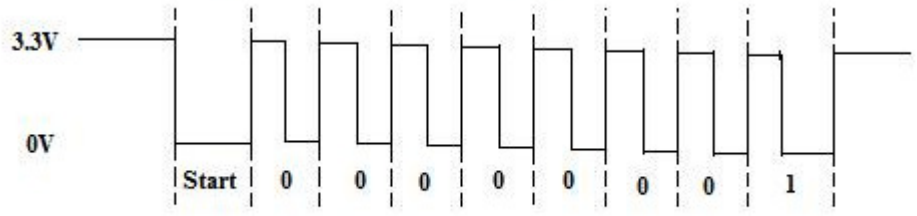
MSP430g2231

Software

1. Awaits Button press
2. Send a unique 8 bit IR PWM signal with a 38KHz carrier frequency.

IR Detector Circuit





Hardware

TSOP32338 IR Receiver Module

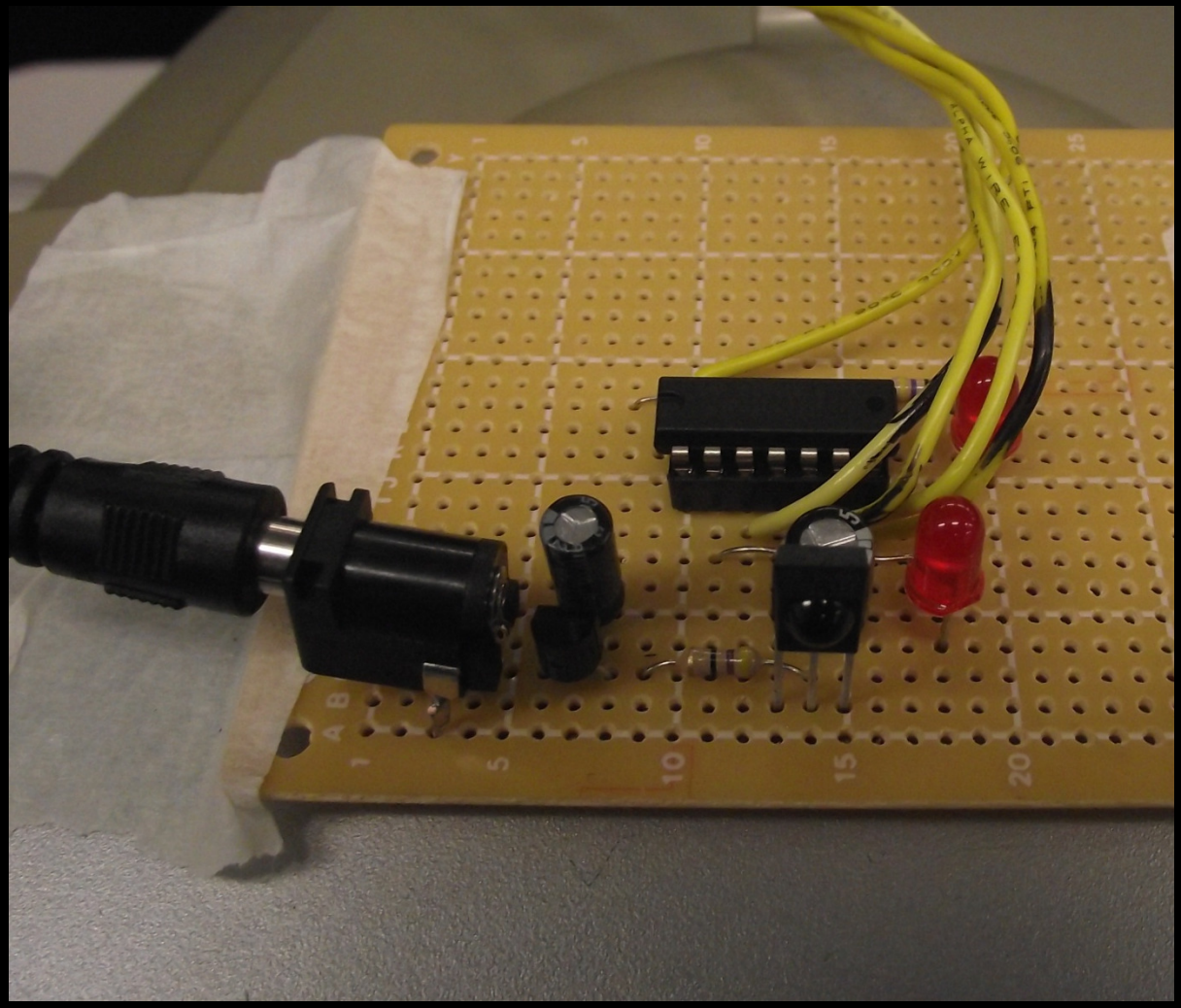
- Filters and amplifies at a carrier frequency of 38KHz
- +/- 45 deg. detector angle

MSP430g2231

Software

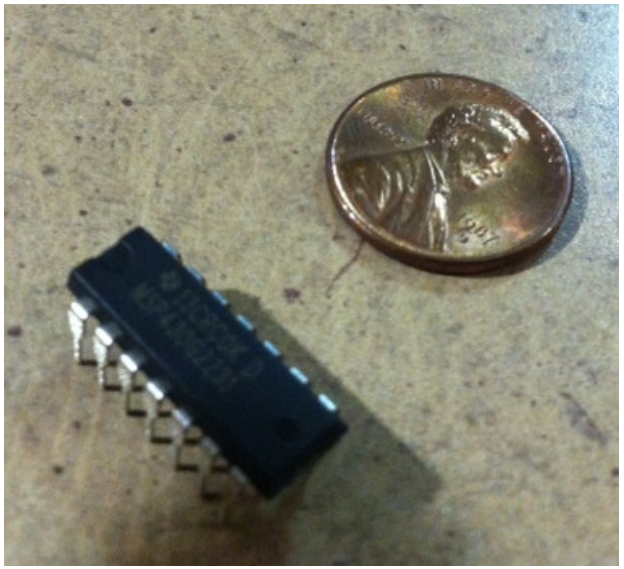
1. Checks Start bit period
2. Captures and computes following 8 bit periods, determining whether a high or low bit has been transmitted
3. Compares captured ID to headphone IDs
4. If matched sets RTS high and sets Line0 & Line1

Implemented Detector Vector Board

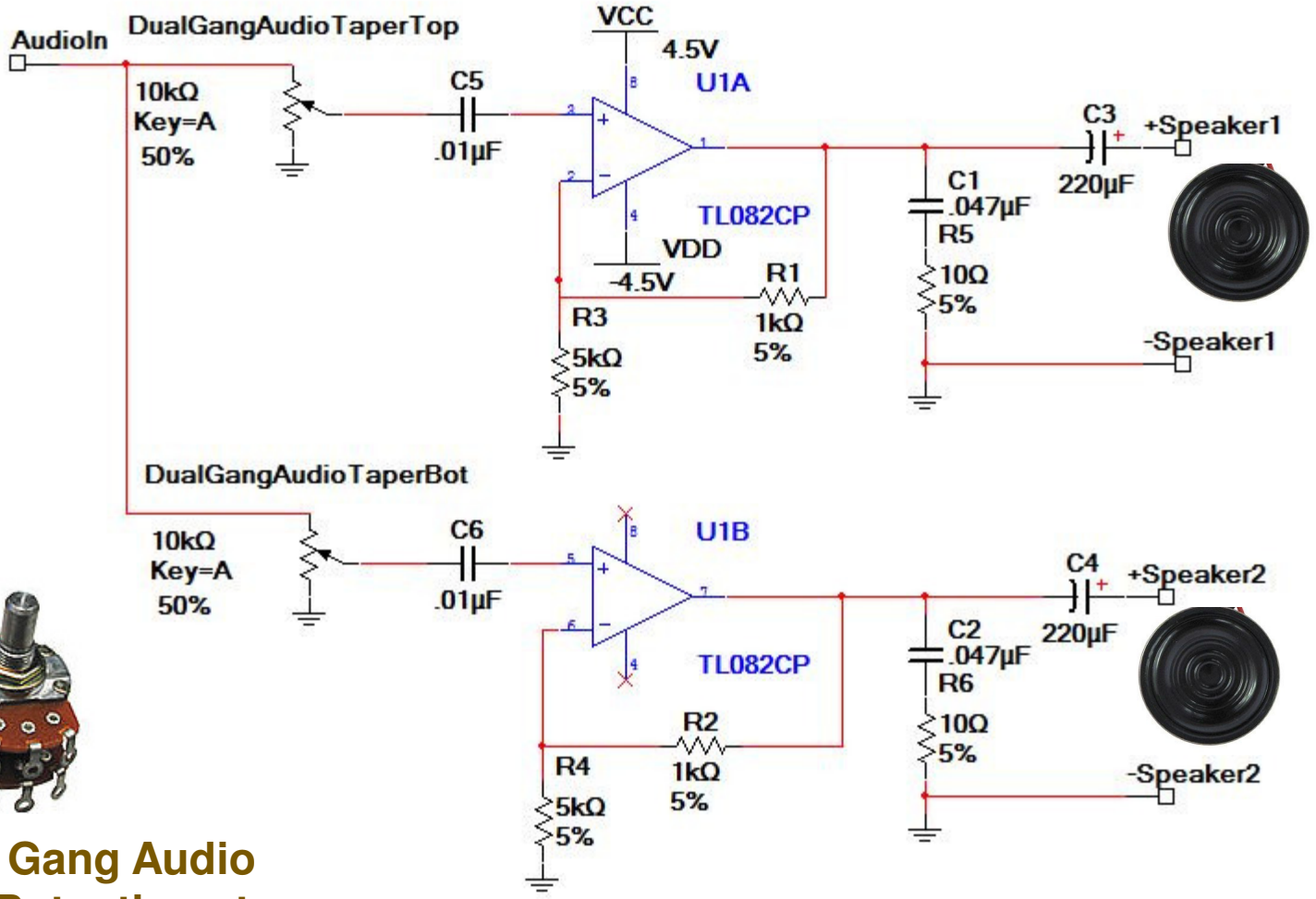


MSP430G2231

- Application
  - IR Detector and Emitter Circuit
- Pin Functions
  - PWM
  - Capture/Compare
  - Interrupts
- Description
  - Ultra Low Power
  - 10 I/O pins

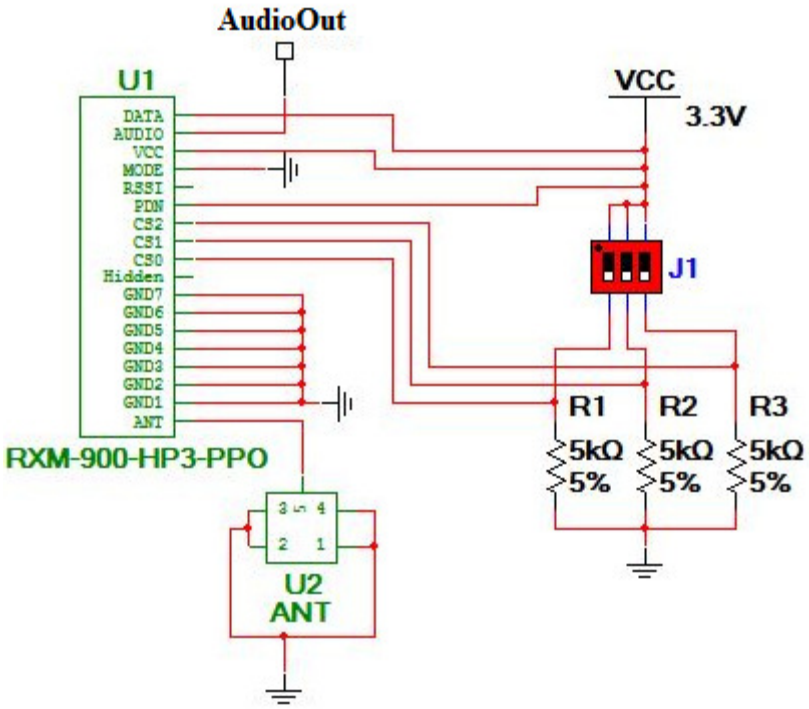


Headphone Audio Output



Dual Gang Audio Taper Potentiometer

HP3 Receiver

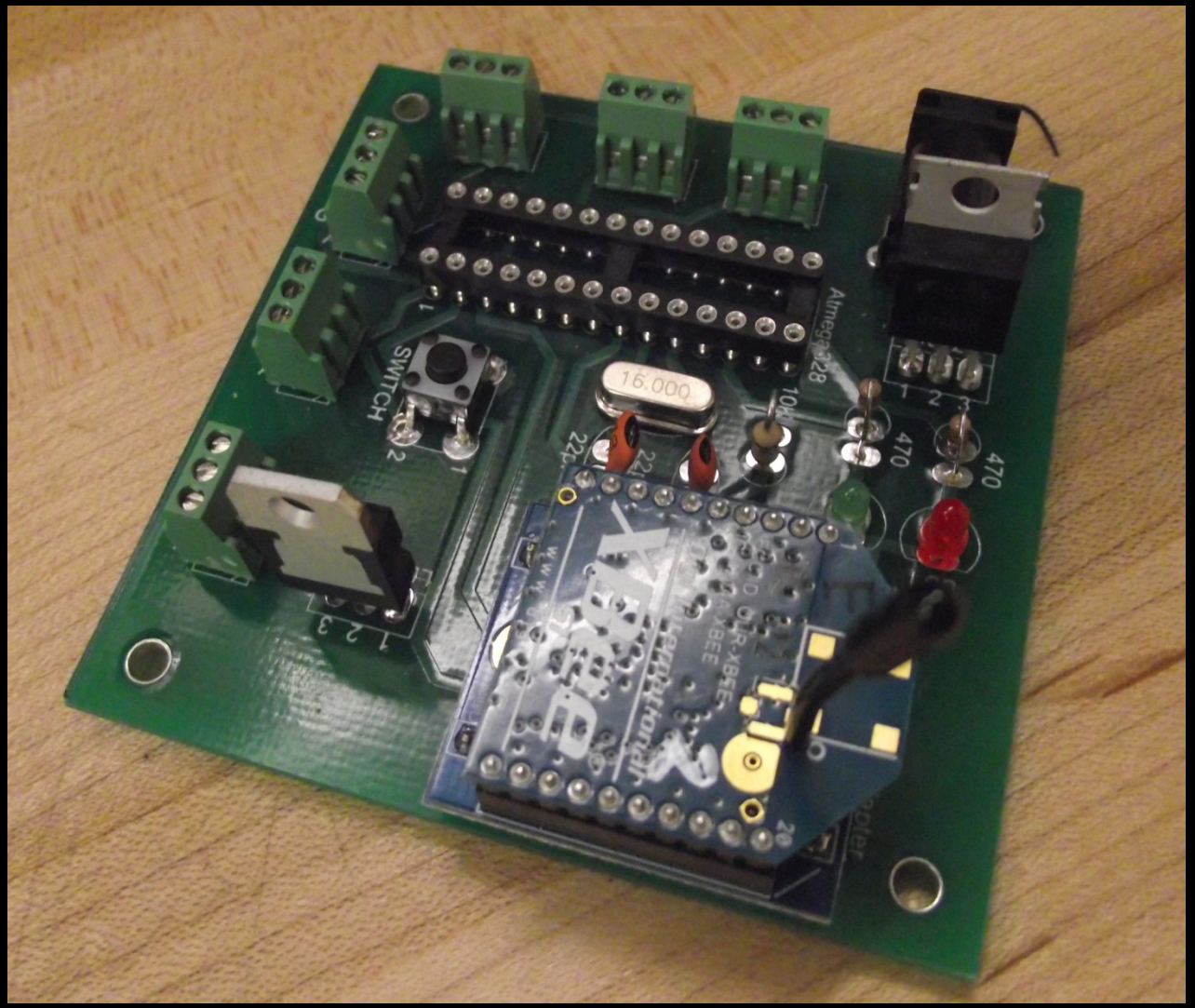


- Linx Technologies
- 900MHz Radio Frequency
- Range up to 1000 ft
- 8 channels
- Internal FM to AM conversion.
- Module, no external components for signal processing.

Triangulation

- The triangulation is performed via Radio Signal Strength Indication(RSSI) in the 2.4GHz RF range.
- The headphone is trained to find it's own location in reference to each corner of the room.
- Each corner contains a circuit board with an Xbee and power supply that each headphone can call on.
- Each headphone contains a circuit board with an atmega328, an Xbee, and a power supply.

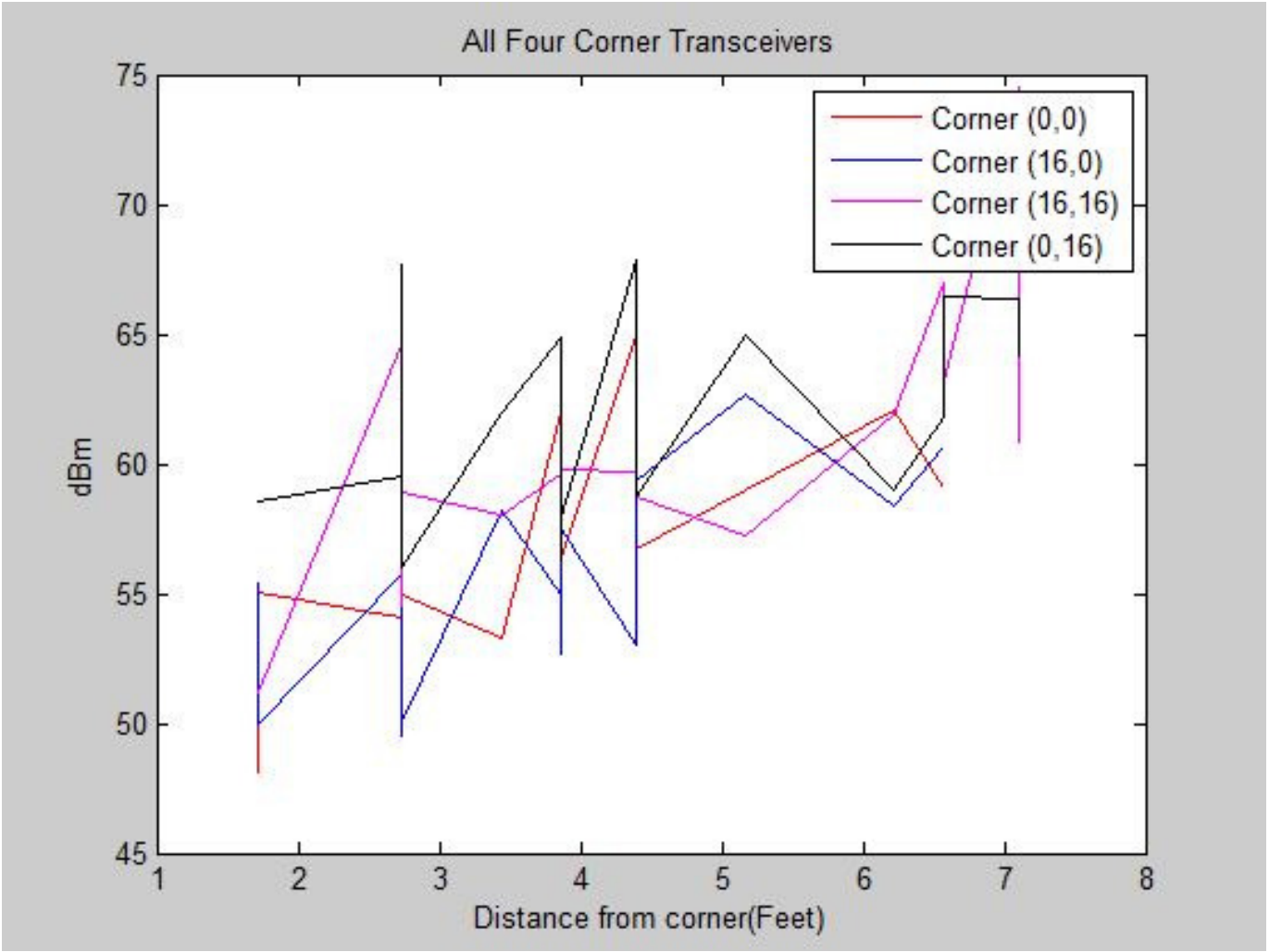
Triangulation PCB



Training

- For each corner in the room the headphone measures an RSSI value. This RSSI value is an average of 25 RSSI values measured consecutively to minimize outliers (noise).
- These four RSSI average values are saved on a table and mapped to a known (x,y) coordinate.
- This is done on a 16x16 ft grid with points 4 ft apart.

RSSI vs. Distance



Testing

- 5 points are taken for each corner and an average is measured to lower noise.
- This four averages are then compared to every four RSSI values in the training table by using a 4D distance formula as shown below.

$$D = \sqrt{(RSSI1[i] - RSSI1_{test})^2 + (RSSI2[i] - RSSI2_{test})^2 + (RSSI3[i] - RSSI3_{test})^2) + (RSSI4[i] - RSSI4_{test})^2}$$

  - i=o…n where n is the length of the training table
- The index of the minimum D found is then used to fetch the (x,y) position of the headphone from the training table.

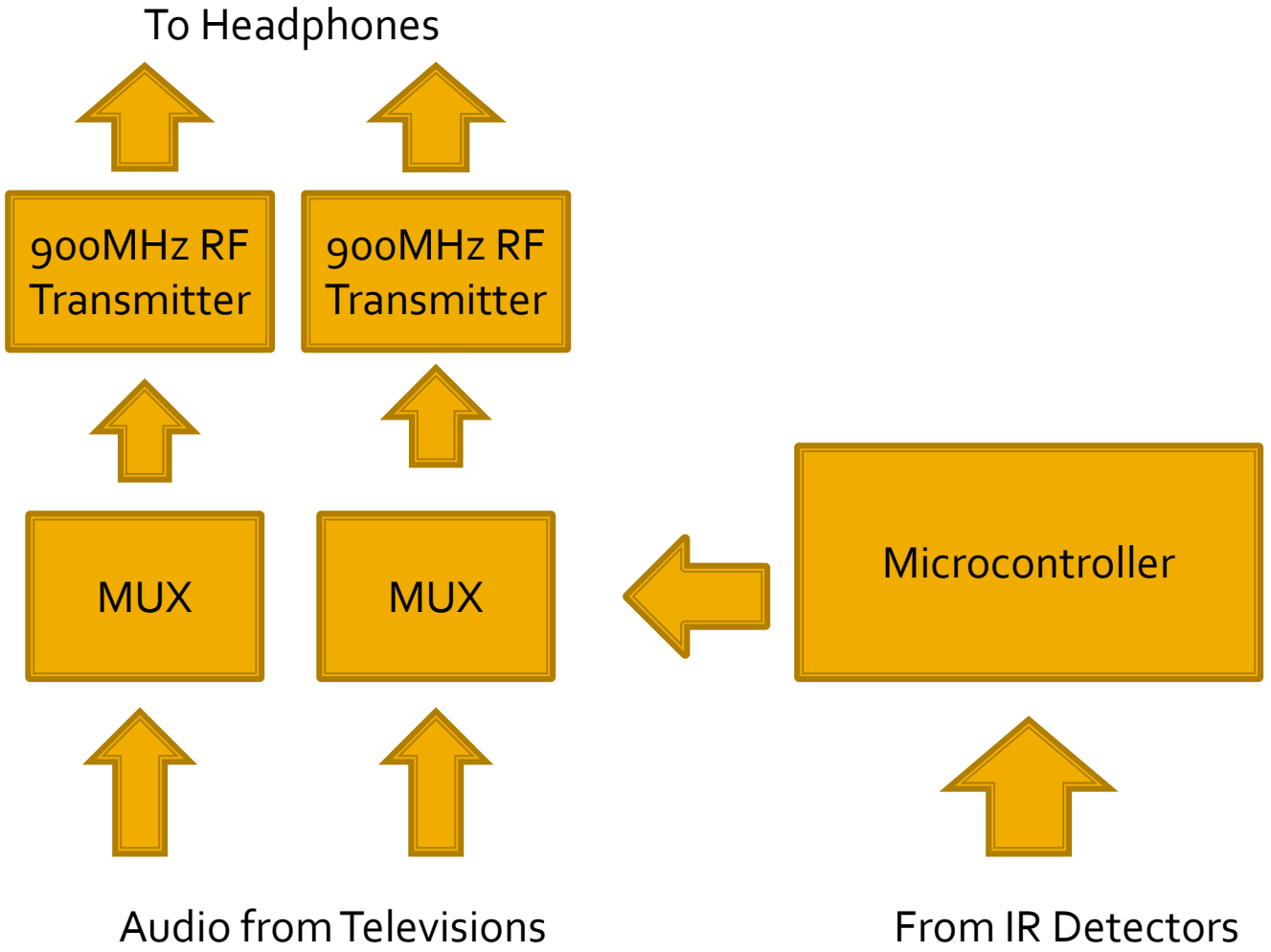
Communication to Tablet

- The (x,y) coordinate found is then translated into a serial message send via 2.4GHz RF to the table application. The message is as follows:

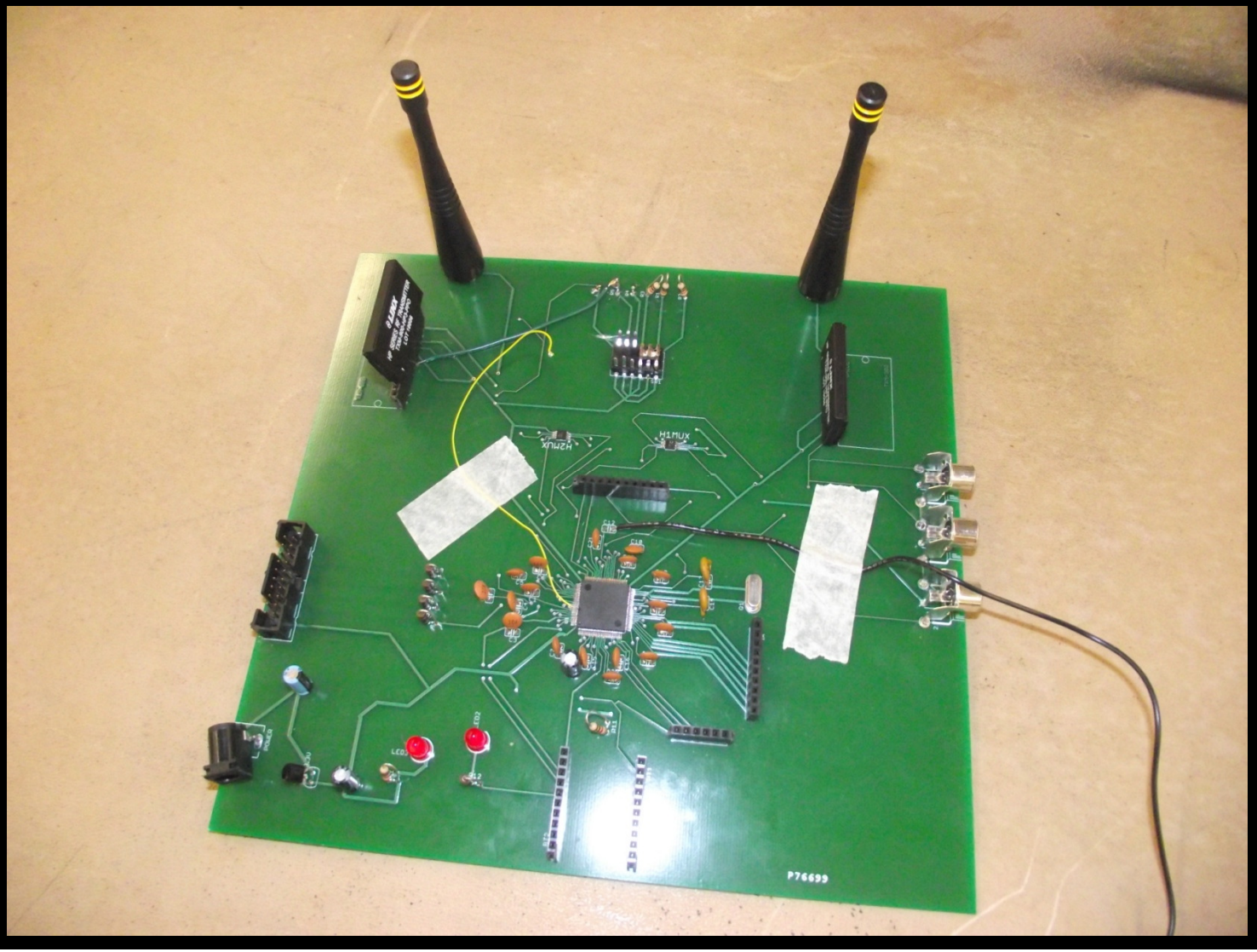
$name,x,y^

- $ - start serial message flag
- name – headphone name abbreviated. Ex: H1
- x – x coordinate in meters
- y – y coordinate in meters
- ^ - end serial message flag

Master Board



PCB Master Board



Microcontroller

- The Stellaris LM3S8962 is the center piece to the project.
- The 8962 was chosen for its high number of GPIO pins and the corresponding development board that can be used to test the code.
- The GPIO pins will be used for the MUX selector lines and the IR detectors.

4:1 MUX

- A 4:1 MUX is used to switch sound to each headphone.
- Each headphone corresponds to a MUX.
- With 3 televisions we will use input lines 1-3.
  - The Stellaris will select which television's sound will go to a particular headphone.

RF Transmitter/Audio Transceiving

- Two 900MHz transmitters are used on the master board, which pair up with a receiver on each headphone.
- The transmitter will be in the parallel state which will allow it to use multiple channels.

CS2	CS1	CS0	Channel	Frequency (Hz)
0	0	0	0	903.37
0	0	1	1	906.37
0	1	0	2	907.87
0	1	1	3	909.37
1	0	0	4	912.37
1	0	1	5	915.37
1	1	0	6	919.87
1	1	1	7	921.37

RF Transmitter/Audio Transceiving (Cont.)

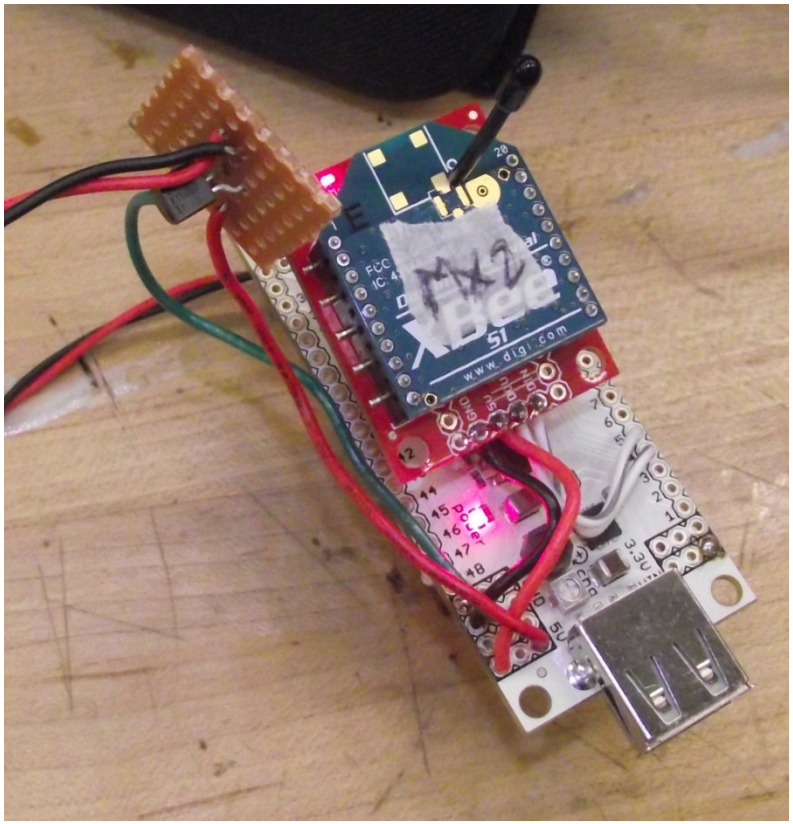
- RCA connectors will be bringing in the sound from the televisions.
- Once the sound has reached the PCB the sound will then be split between the two MUXs.

Tablet Requirements

- Needs to have access to a USB port
- Screen size needs to be between 7” – 10”
- Needs to have long battery life.
- Familiar Programming Language

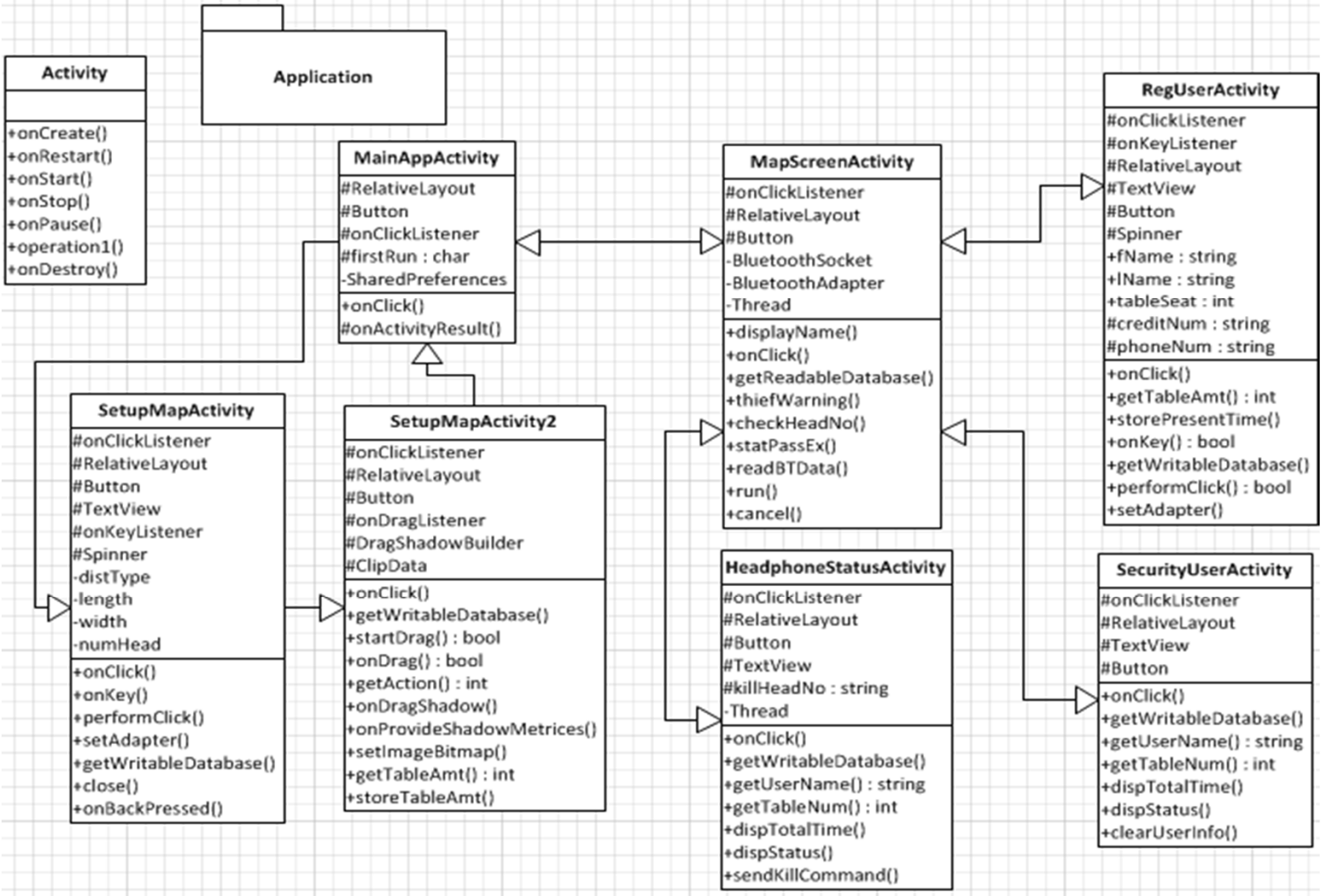
Specs	Toshiba Thrive	Acer Iconia A500	iPad (version 1)
Screen Size	10.1 inches	10.1 inches	9.7 inches
USB Ports	USB 2.0 MiniUSB	USB 2.0 MicroUSB	None
OS Stack	Honeycomb (3.2)	Honeycomb (3.2)	iOS
Language	Java	Java	Objective-C
Retail Price	$379 [$250]	$349	$399 + $99/year dev
Battery Life	~11 hours	~7 hours	~10 hours

IOIO for Android Triangulation Communication

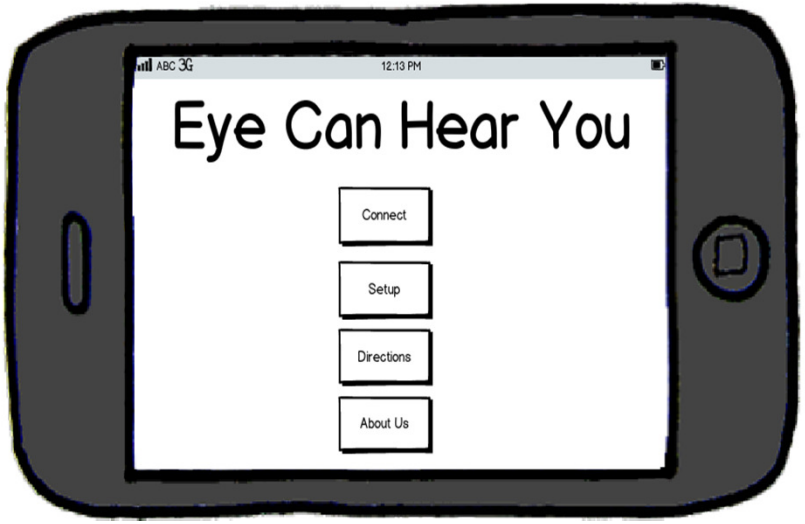


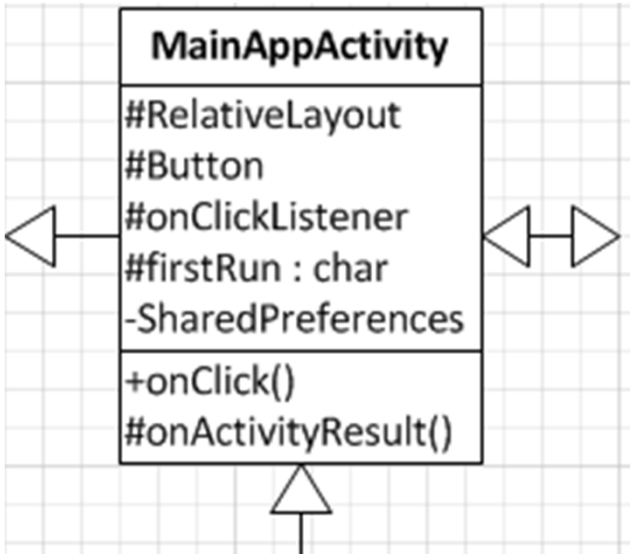
Use SparkFun's IOIO for Android device which utilizes the Android debug bridge (ADB) interface to communicate with the Xbee Triangulation system.

Class Layout – 7 Activities



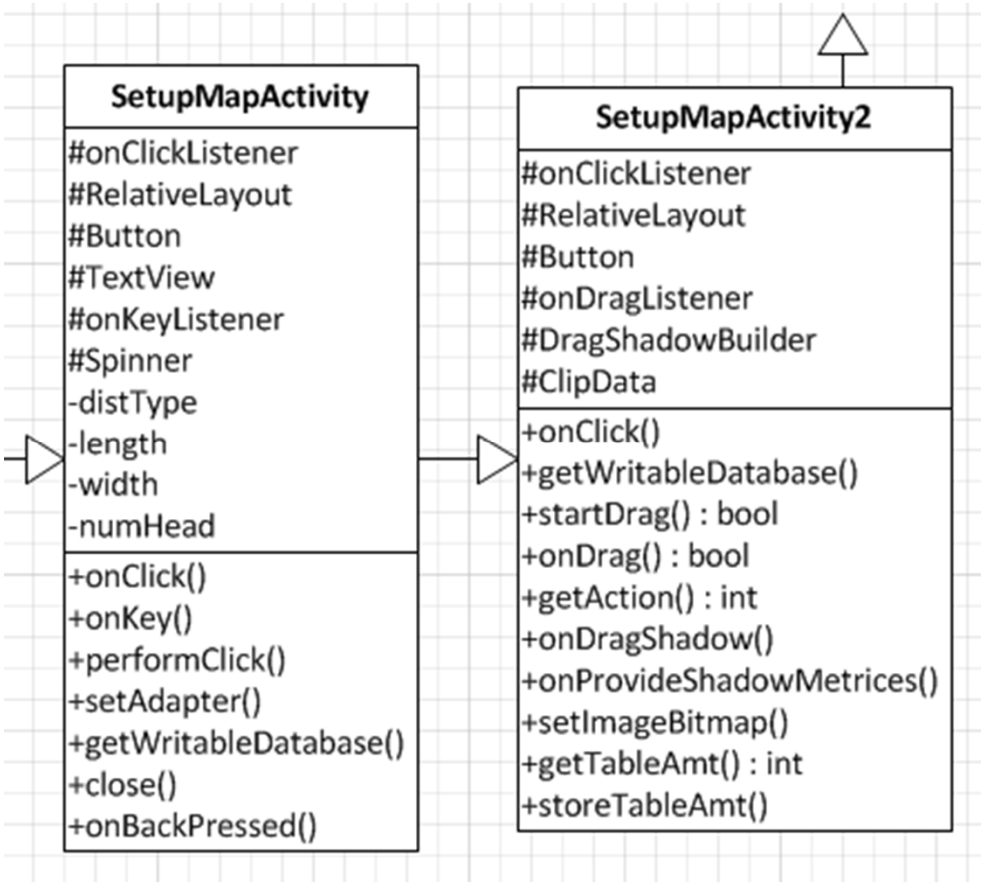
MainAppActivity





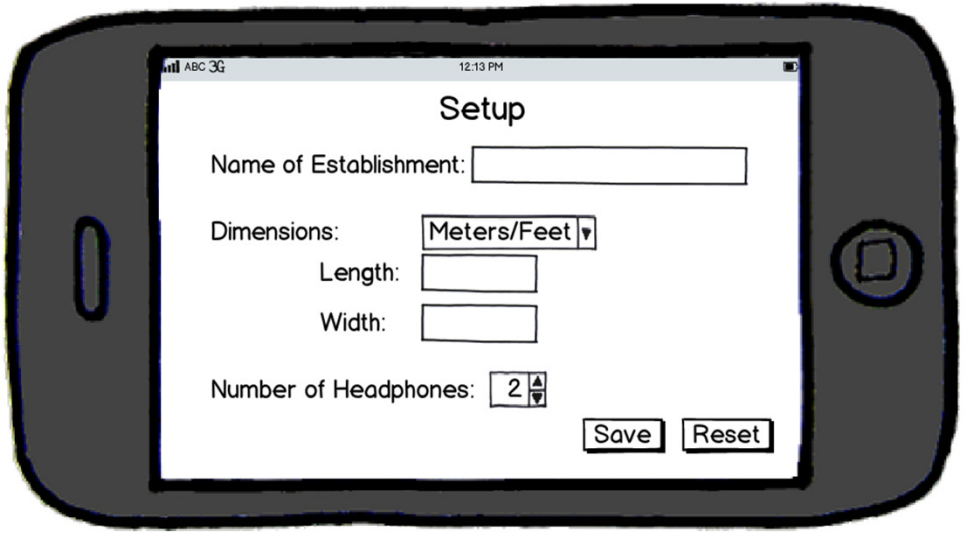
- Landing for the Application.
- User has 4 choices to make
  - Connect to the Map and IOIO Device
  - Setup the Map
  - Read Directions
  - Visit the SD website
- First Run Check
  - Connect button disabled

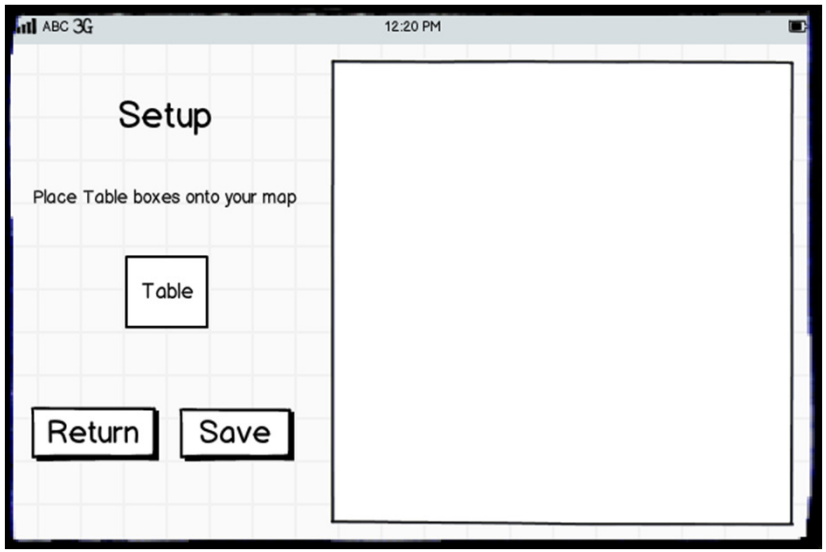
Setup Map Activities



- Two activities handle the setup phase.
  - SetupMapActivity requests general info of the restaurant and stores it into a restaurant database (SQLite).
  - SetupMapActivity2 pulls dimensions from the database and constructs a map to place tables on.
    - Using Drag-and-Drop method.

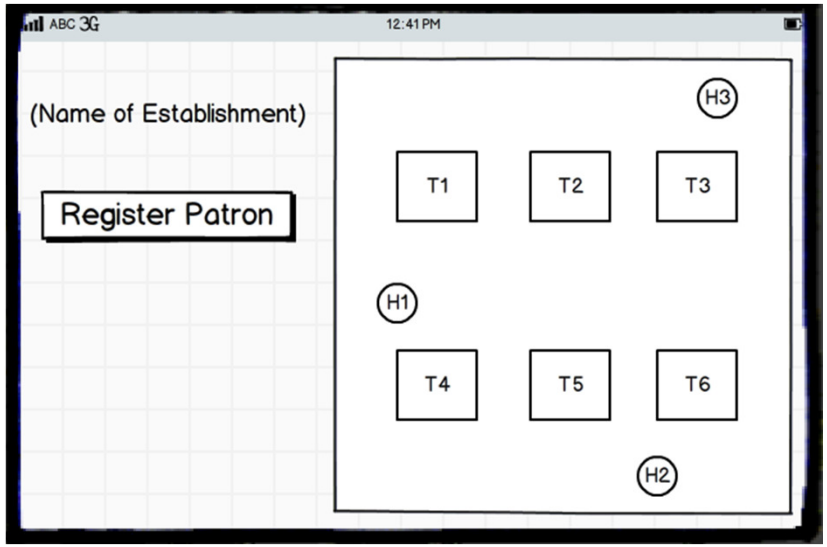
SetupMapActivities (cont.)

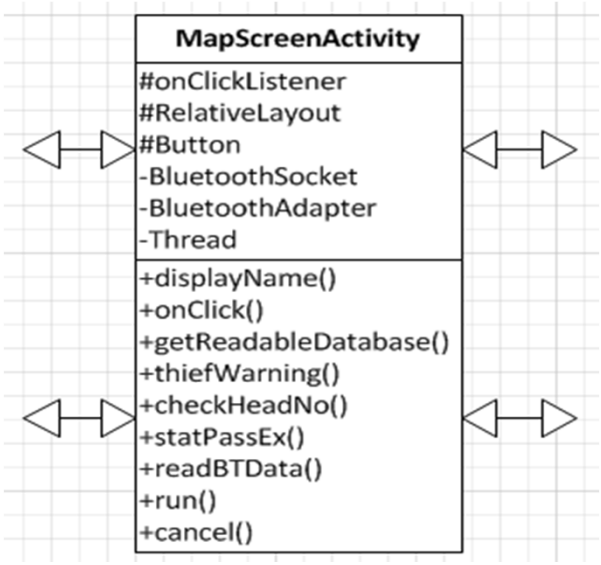




- Table squares are the objects being dragged.
- Drag Object Shadow produces (x,y) point. Stored for later.
- Once finished, both activities finish and return directly to MainAppActivity.

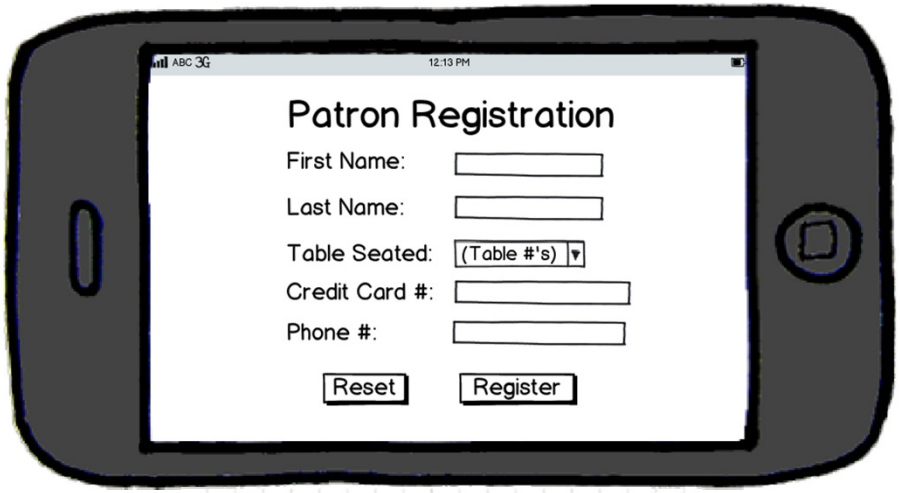
MapScreenActivity

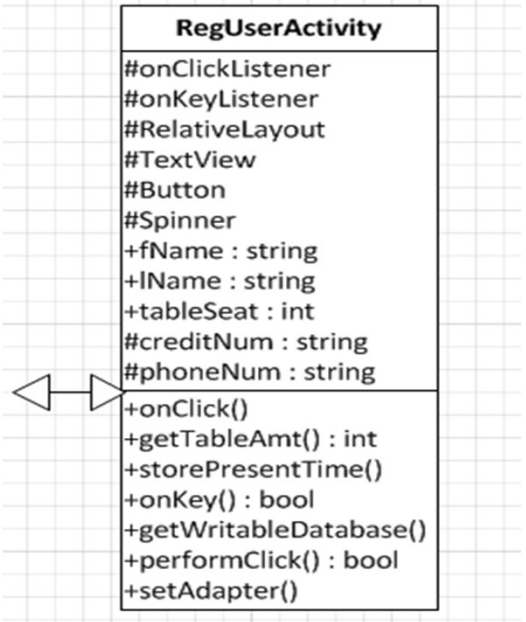




- Main function of the application.
- Monitors headset location.
  - Tables are orange squares and headphones are blue squares.
- IOIO device connection is established here.
- Security warning occurs here if headset leaves dimensions of restaurant (stolen option or “out for repairs”)

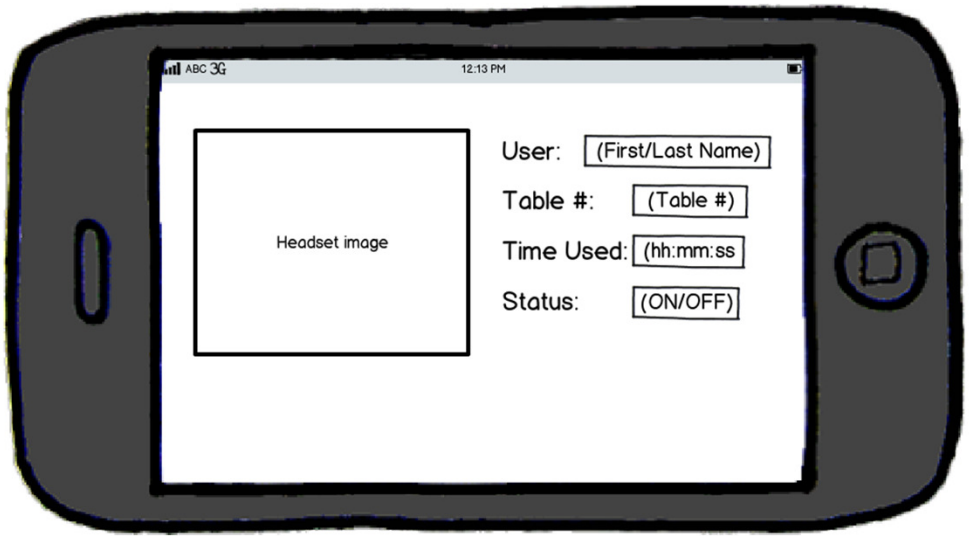
RegUserActivity

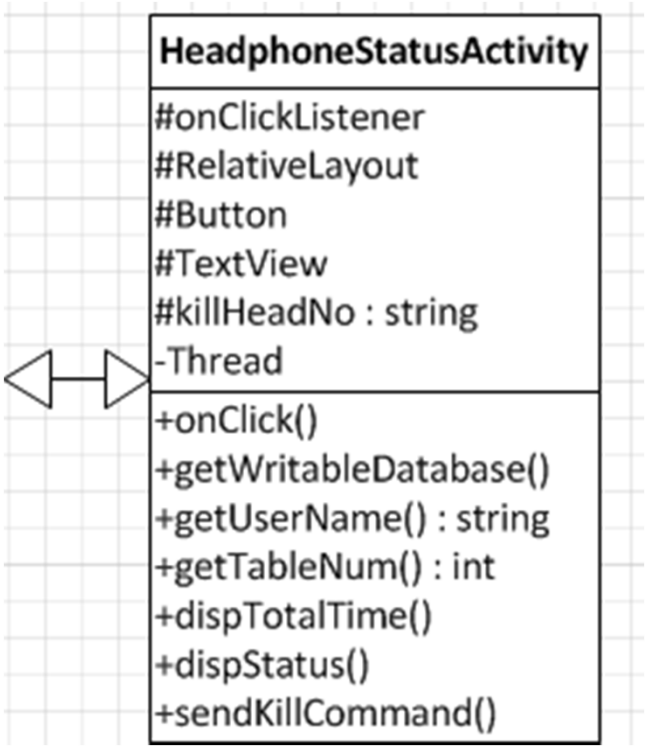




- Registration form for customers.
- Data placed into Patron SQLite database.
- Credit Card #/Phone # used for security (CC for collateral)
- Returns back to MapScreenActivity.

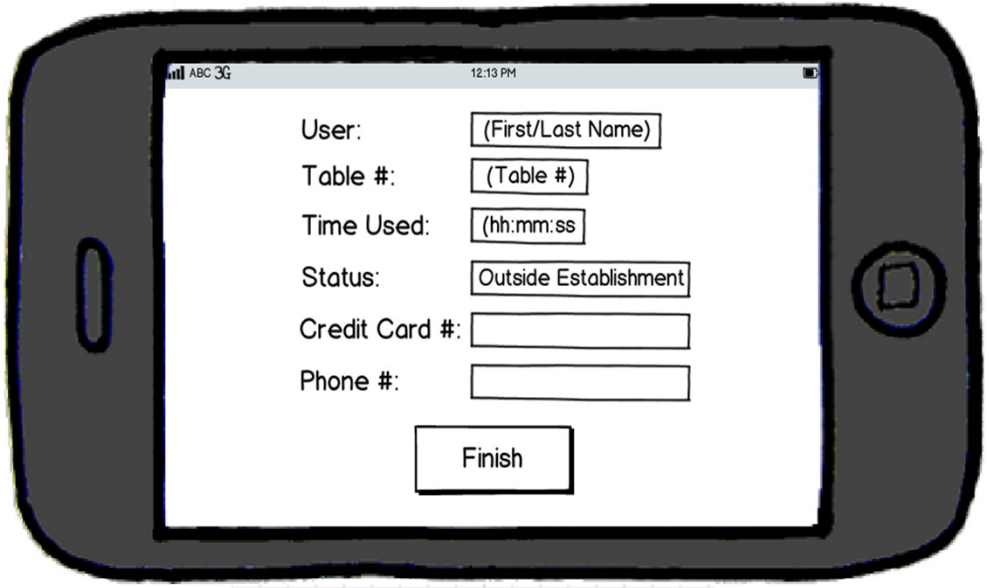
HeadphoneStatusActivity

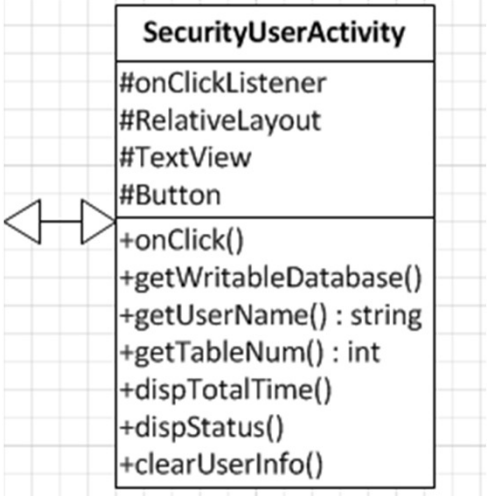




- Activity displays data about the customer.

SecurityUserActivity





- Once registered headset leaves the dimensions of restaurant, security flag is set and takes user to this activity.
- Displays user data along with the credit card and phone number.

Budget

Component	Price	Qty.	Total	We paid
IR LED	$0.36	25	$9.00	$9.00
IR Detector	$1.23	6	$7.38	$7.38
4:1 MUX	$2.84	4	$11.36	$11.36
Speaker	$0.75	6	$4.50	$4.50
MSP430	$1.62	6	$9.72	Sampled
Stellaris	$14.19	1	$14.19	$14.19
Bluetooth Module	$59.99	1	$59.99	$59.99
Xbee	$22.95	6	$137.70	Donated
900 MHz Transmitter/Receiver	60.05	2	$120.10	$120.10
3.3 Volt Voltage regulator	$0.57	10	$5.70	Sampled
5.0 Volt Voltage regulator	$0.49	10	$4.90	Sampled
Antenna	$6.75	4	$27.00	$27.00
Op-Amp	$1.42	3	$4.26	$4.26
Audio Taper Potentiometer	$1.55	3	$4.65	$4.65
Potentiometer Knob	$0.68	2	$1.36	$1.36
Antenna Connector	$2.76	4	$11.04	$11.04
9V battery straps	$0.65	2	$1.30	$1.30
2.1 mm power plug adapter	$1.36	8	$10.88	$10.88
555 Timer	$1.11	2	$2.22	$2.22
6 Position Switch for Master Board	$0.85	2	$1.70	$1.70
PCB	$219.10	1	$219.10	$219.10
3 Position Switch for Headphones	$0.75	4	$3.00	$3.00
Total			**$671.05**	**$513.03**
How muched we saved				**$158.02**

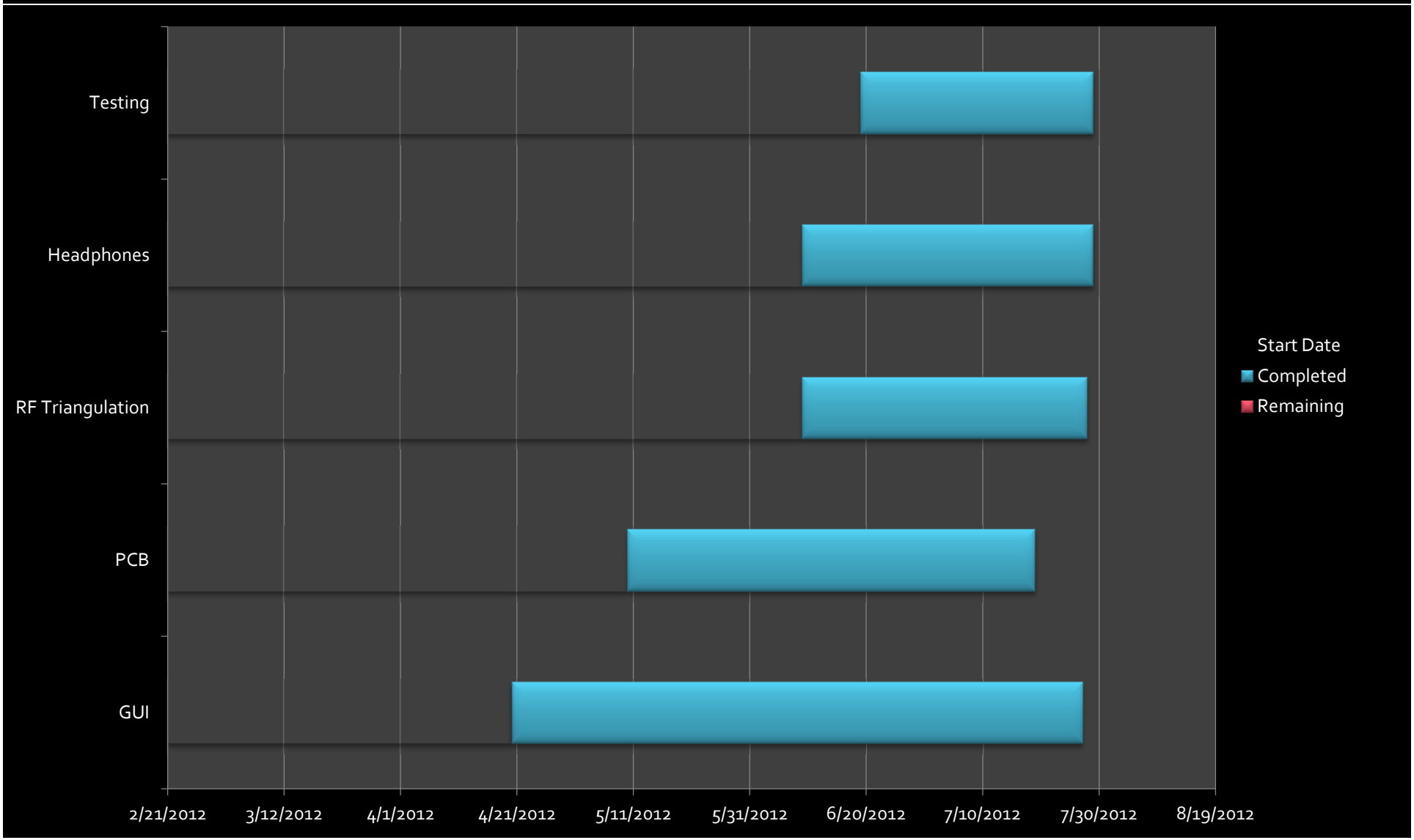
Overall Budget

Total Income	$ 800.00
Total Expenses	$ 513.00
Net	$ 287.00

Project Delegation

	Infrared Identification	Programming	Audio Transmission	Tablet	Triangulation	Audio Switching
Michael	X	X	X			X
David		X		X	X	
Whitney	X	X	X			X
Edward	X	X	X		X	X

Milestone


Testing
Headphones
RF Triangulation
PCB
GUI
2/21/2012
3/12/2012
4/1/2012
4/21/2012
5/11/2012
5/31/2012
6/20/2012
7/10/2012
7/30/2012
8/19/2012
Start Date
Completed
Remaining

Questions?