Objective-Based Laser Tag with Player-Sense Technology

Project Proposal

By Sean Morris, Jonathon Jellison, Josh Seelye
Advised by Dr. Aleksander Malinowski
Published November 13, 2018

Table of Contents

Abstract

Laser tag is a game that simulates combat for the goal of eliminating enemy players and complete objectives. In its current state of the art its comercial implementation involves infrared (IR) emitters and receivers. “Bullets” of IR light are sent when a player presses their trigger in the hopes of hitting their opponents IR sensors located on either a wearable vest, hat. If the “Elimination” protocol was sent and is successfully recognized by the receiving player, a “hit” command is then acknowledged. When an elimination protocol is received by a vest, that player’s vest will turn off for an amount of time *X* until they respawn. Within each fired protocol, the ID of the player who fired the “eliminate” command is received and saved by the hit player until the end of the game when the guns are plugged into the main system.Their recorded data from the game is then uploaded for the end-of-game stats. Each Laser Tag system also has RGB LEDs and a speaker to allow for the player to know when they are hit with an “eliminate” protocol. Currently, this is how most Laser Tag systems work, and there are no systems that record and communicate data with players in real-time.

1. Introduction

A. Laser Tag Fundamentals

In order for the players to have a successful laser tag game and experience, a few fundamentals must first be established.

<u>IR Gun</u>
Each player shall be equipped with a gun that is capable of shooting IR "bullets". This gun must also have a power source, and be connected to a vest that can determine if a player has been shot. If so, the gun is rendered useless, and the player must respawn.

<u>Laser Tag Vest.</u>
The vest must have multiple IR sensors on the front and back so that it is capable of detecting if a player has been shot by an IR "bullet". If so, the vest must communicate back to the gun that the player has been shot. The vest should also light up to alert all surrounding players that the player is dead.

<u>Data Sharing</u>
At the end of the game, it is important for the players to know simple game statistics such as how many Eliminations and points each player got and which team won the game.

<u>Objective</u>
There must be something for players to work towards and win. Whether it is eliminating all enemy players, capturing a flag, or holding down an area, there needs to be an end goal that the players work towards. Otherwise, the experience is unsatisfactory.

B. Real world application

Most laser tag businesses use the aforementioned essentials and put their own spin on objectives. For instance, some locations have bases for each team, and inside the base is an IR sensor. If a player gets into the opponent's base and "destroys" the IR sensor by shooting it multiple times, a large amount of points is awarded to that team. Other places have physical objects that you must interact with to capture or enter a secret code to deactivate a threat. It is also possible that a business has one game mode in which the team with the most eliminations at the end of 10 minutes wins.

2. Problem Statement

Current laser tag systems do not have a way of actively communicating with or monitoring players/objectives in real-time on the playing field. Additionally, present day laser tag objectives do not have autonomy; meaning the player must physically interact with an objective. This means the player must take their eyes and aim off the playing field, leaving them as an open target to the enemy.

In general, realism has always been a problem with present-day laser tag systems. In certain cases where a player has been eliminated, the player will respawn after a given period of time causing the issue of being able to wait around and respawn in the same location they were eliminated. This allows the possibility of continuing to capture an objective or to immediately eliminate an enemy with little to no risk of getting eliminated themselves. Realism issues also tend to show up in the statistics gathering system. While a game is in progress, a player has little to no information of the game as a whole, and only a relative idea of their own statistics. This abstracts the player from the immersive atmosphere and could possibly affect their playing experience making the overall system flawed and less fluid.

3. Solution

In order to handle the issues listed, this laser tag system will utilize wireless communication such as IR and Wi-Fi to acquire real-time data, create a unique laser tag experience with autonomous objectives, and communicate in-game statistics to and from individual players.

A. Systems

Communication:

Wi-Fi will be the primary method of communication between players and objectives. Each gun must continually be notified of game updates so that the LCD is able to reflect the current game status. Flags must also be kept up to date on game events so that colors of flags can be changed appropriately depending on which team owns the flag. See figure 1 for the high-level communication diagram.

Laser Gun and Receiver:

Each player will possess a receiver and laser gun. They will be connected via communicating MCUs and contain subsystems such as a Wi-Fi module, speaker, IR sensors, IR LEDs, and an RGB LCD. Two different IR sensors will be mounted on the players gun for each of the different carrier frequencies. One for the flags' IR opcodes, and another for the guns' IR opcodes. If this is accomplished and goals are met via the schedule, a vest will be implemented to allow for the user to have multiple IR receivers for registering a hit. See figure 2 for the high-level diagram of the laser gun.

Flags:

Each flag will consist of an IR Emitter, Wi-Fi module, and an LED light ring. The IR emitter will emit a frequency that differs from the guns', so they do not interfere with each other. In addition, a flag will have the ability to change the power level of the IR LEDs therefore directly affecting their transmit range. This will allow for the flags to be more modular and adaptive in different environments. The Wi-Fi module will have to be used for transmitting data to other flags/players. The color of the LED ring will reflect which team currently owns the flag.

B. Subsystems

IR Sensors and LEDs:

Each gun will have IR sensors to detect a "bullet" and inform the player when they have been shot. The gun will have IR LEDs to shoot a "bullet" at enemy players, as well as IR sensors to receive a different frequency from the "bullet" frequency. These sensors will handle the communication with the flags to tell whether or not the player is capturing a point.

LCD Display:

Each gun will have an LCD mounted on the stock that displays information about the current game. This will be done through multiple menus that will be available on the LCD. Figure 3 gives an example of what some of the menus will look like, as well as the information that the LCD will display. One menu will present information about player statistics such as eliminations, times eliminated, and the number of flags captured. The other menu will provide the status of all flags. While on either of the menus, the player will be able to see their remaining ammo and health. At any given time, the player will be able to seamlessly switch between each of these menus by tapping on the resistive touch LCD.

Speaker:

A speaker will be mounted on each gun to inform the player on major game changes such as the enemy capturing a flag. The speaker will also be used to create in-game audio cues, such as shooting, reloading, getting shot, or needing a medic. The speaker will play audio files in unison with the LCD as it updates.

Power System:

Custom power systems will be designed to power the individual subsystems on the gun/vest, as well as the flags. Due to the fact that each system will be modular and mobile, rechargeable battery packs will be used as power sources, and battery charging circuitry may have to be constructed.

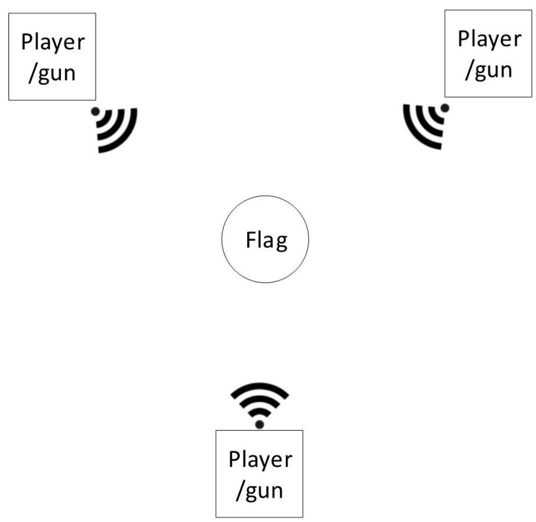


Figure 1. High-level communication diagram.

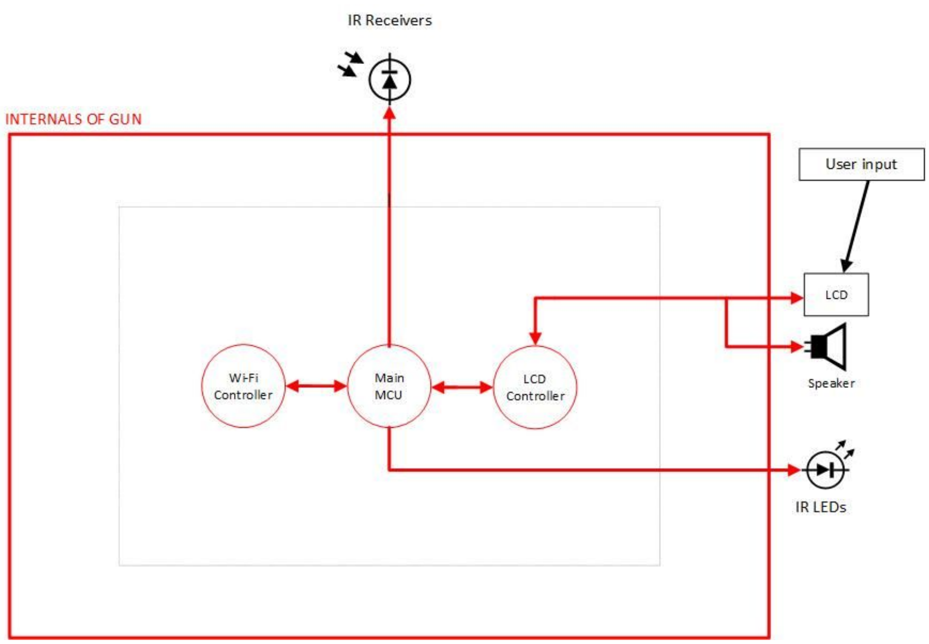


Figure 2. High-level LCD communication diagram.

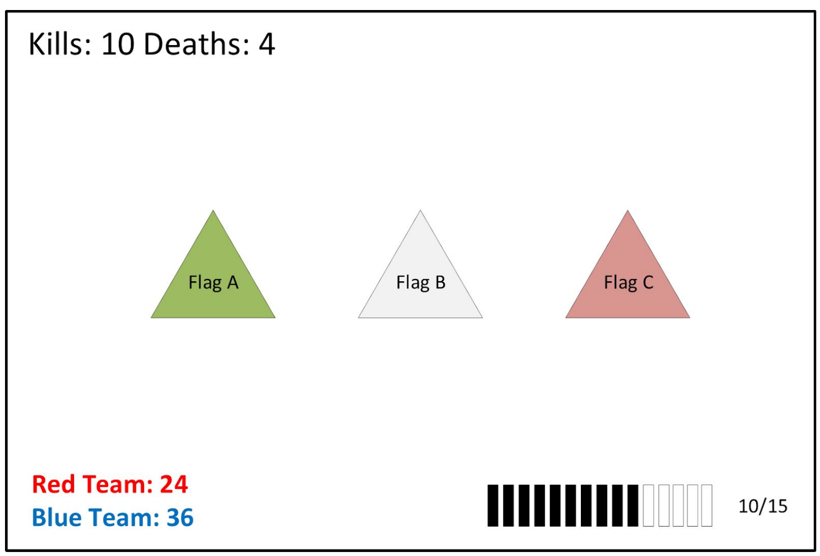


Figure 3. LCD user interface

Preliminary Research

To handle the issues as stated above with our solution the following sections outline the preliminary research that needed to be done to achieve our teams desired goals.

A. Game Communication

For the main internal game communication, it was undecided whether to use Bluetooth or Wi-Fi. The research was done into both to see if Bluetooth would be capable of handling the connection between many devices and whether or not Wi-Fi was deemed excessive under the specifications of game communication. After some discussion, it was decided that the best solution was to use Wi-Fi due to the fact that Wi-Fi is better for handling multiple connections and can be implemented without using the master/slave setup inherent to bluetooth.

With Wi-Fi, it was agreed upon that the best board for the situation would be the ESP8266 because it satisfies all project requirements including cost and the project team is already familiar with the platform. This board has 17 GPIO pins and is capable of long-distance communication of up to 300 meters line-of-sight[1]. Tests were done on this board to see how long it could read through a concrete wall. It was found that at approximately 45 ft the two devices could see each other. This test was conducted by putting one ESP8266 in a room behind a concrete wall then

These boards will be initially connected in a master/slave network meaning that one board will be the master and all other boards will be access points to that board. This will allow all of the boards to communicate back to the master. Once the boards are all connected to each other they will enter setup mode and wait for the command from the master with the SSID and password for the home network. Once the boards have the SSID and password for the home network the all of the boards will then connect to the home network allowing for communication between all of the boards on network.

For communication between the boards, a custom TCP-based protocol is going to be used to simulate a global ping on the network. This will allow all the devices to receive the information from each elimination and flag capture. To simulate the global ping, the device needing to send information will connect to each of the

[1] https://cdn-shop.adafruit.com/product-files/2471/0A-ESP8266__Datasheet__EN_v4.3.pdf

other devices on the network through TCP. This will allow all of the devices to keep track of each other and compare statistics.

B. IR Gun

Initial research was done to understand how IR signals are used. IR takes advantage of protocol signals sent as groups of words.

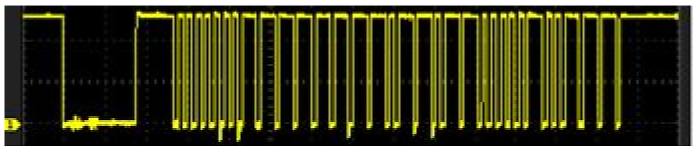


Figure 4: IR protocol

Each protocol is made up of a carrier frequency that usually ranges between 30 kHz to 60 kHz. Variations in the length of each group of pulses as shown in figure 4 of this carrier frequency results in either a ‘1’ or ’0’ is transmitted. Time for transmitting is relatively long as a word of data could take up to 10ms to send.

One of the more common industry standard protocols utilizing this process, as previously explained, is the NEC protocol[2]. It relies heavily on pulse distance for logic ‘1’ or ‘0’.

Using a signal between 40-42 kHz is a fast enough carrier frequency that the MCU’s can create and still allow for enough processing time for other information to still be given to the CPU for another task such as peripheral control.

IR signals are sent with the NEC protocol and are received by choice of two different IR receivers, active or passive. Passive IR receivers simply digitize the raw data of each ‘1’ and ‘0’, showing every pulse that makes up each bit. Active IR receivers take the same raw bit data and converts that data into one single pulse. This is much more useful for converting the IR signal to a digital signal with software. Figure 5 shows both types of receivers results.

[2] https://en.wikipedia.org/wiki/Consumer_IR

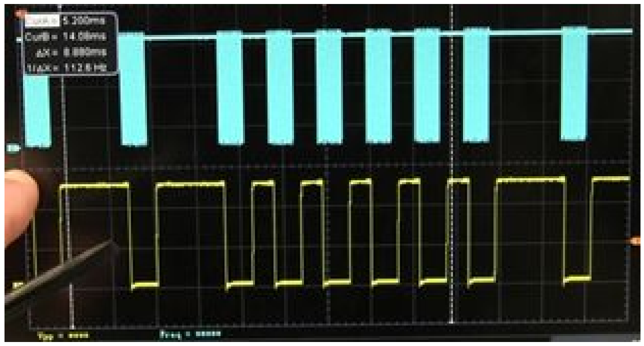


Figure: 5: Active and Passive IR receiver data

The yellow channel in figure 5 shows the results of an active IR receiver. The blue channel shows the results of a passive IR receiver. For the purposes of this project, active IR receivers will be used.

C. Peripherals

For systems such as the speaker and LCD, foundational research was done to help better understand how the LCD and speaker were going to integrate with the IR gun and the communication system. To do this, research was done to find the best LCD that was suited to handle these tasks.

The LCD that was chosen was the Gen4-uLCD-dt[3]. This LCD contains a Diablo16 which is a 16-bit embedded graphics processor capable of controlling the LCD and more using its clock that can be set as high as 160 MHz. Using a 30-pin zero insertion force (ZIF) cable, the Diablo16 can communicate and interface with an external microcontroller using any of its 16-GPIO pins.

This graphics processor is also able to process and output audio from one of the pins on the ZIF cable. In addition, the Diablo16 also has the ability to store files on a MicroSD card and can read, display, and play these files. The files must use common format such as JPEG, PNG, or WAV.

The Gen4-uLCD-dt also has a built in resistive touch screen with fairly accurate precision. Utilizing this property, the player will be able to directly interface with the current game giving them the ability to switch screens to any requested information.

[3] https://www.mouser.com/datasheet/2/451/gen4-uLCD-35D_datasheet_R_1_3-1380121.pdf

Research is also being done into whether or not the Diablo16 processor would be precise enough to handle IR. If this research was found to be credible, this would likely make the Diablo16 the primary microcontroller on board the gun.

D. Radio Frequency Identification (RFID)

Understanding RFID is crucial for player-sense technology to work properly. RFID technology is wirelessly reading information off of a small microchip embedded in an RFID tag. Reading the chip is accomplished by emitting electromagnetic waves from an antenna that is powered by an RF source (emitter). These waves propagate through free space, or air, carrying power which is slowly lost as they propagate further. When these waves hit the antenna of an RFID tag, assuming they have enough power, they provide the microchip with a tiny amount of current that is used to modulate the same RF wave with the chip's embedded information. The modulated wave is then sent back through the tag's antenna for the emitter to receive and decode.

Different applications require different frequencies to be used. In general, higher frequencies (840-920 MHz) will yield further read ranges, while lower frequencies (125 kHz) will give a shorter read range. The I-pass system implemented at most Illinois tollways uses high-frequency RFID to scan a tag embedded in the I-pass that sits in the windshield. Grocery store security systems also use low frequency RFID to scan for tags inside of expensive items.

There are two different types of RFID tags that can be used, active and passive. Active tags have an internal battery that enables them to actively broadcast their embedded information. This increases the read distance, as the tag does not rely on the power from the emitter to modulate the information, but increases cost as they are more complex, and are only good for a couple years due to the limiting factor of the battery. Passive tags do not have a battery and only broadcast their embedded information when a wave from the emitter strikes its antenna. This lowers the cost dramatically, but read range also suffers. Tag size, regardless of active or passive, also plays a role in reading range. Generally, a bigger tag has a bigger antenna, which allows more of the emitter's power to be acquired and reflected back.

5. Preliminary Results

A. IR Gun

The first goal of this portion of the project was to determine if an IR protocol could be made based on NEC protocol. To do this a carrier frequency had to be determined and created. A 40 kHz carrier frequency was chosen. Figure 5 shows that carrier frequency.

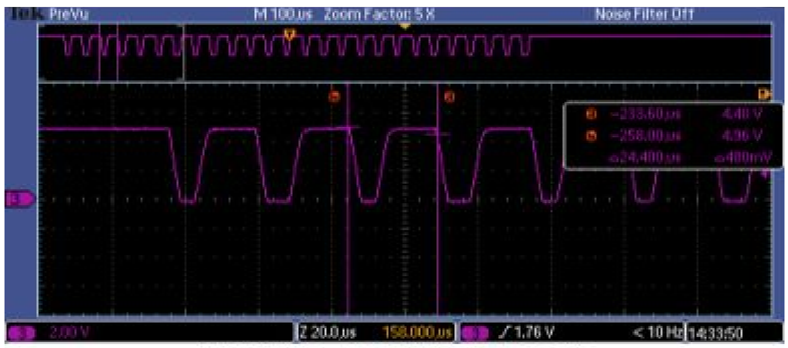


Figure 6: 40kHz Frequency

The frequency was confirmed by determining the time between waves to be 24.4us. Assuming some error in cursor placement, T was determined to be 40983.

$$F = \frac{1}{T} \quad (1)$$

The issue was allowing the MCU to create both the carrier frequency and send the IR elimination data packet. An ATmega128 is our prototyping MCU. It having a 16 MHz clock, the MCU can handle creating that carrier frequency. To allow for CPU optimization, the carrier frequency is toggle on and off only when elimination protocols are being sent. Each protocol only takes about 8 ms to send, allowing for plenty of processing time for future tasks such as decoding received signals. Figure 6 below shows the MCUs sent protocol to the LED in yellow and the received signal in pink.

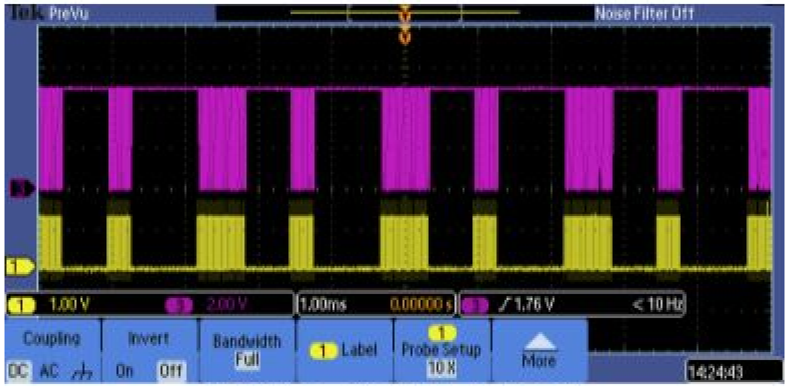


Figure 7: Sent and Received IR Binary Command

Within the sent NEC protocol, an 8-bit data package is sent. The first 3 bits starting with the LSB will determine the identification number of the player that is sending that Elimination protocol along with their team. The next two bits determine what type of gun, in terms of the damage that will be given to that player when they are hit. The last 3 bits are the actual Elimination command. The exact binary library for all of these options within the Elimination protocol are still being created. However, Figure 7 shows the received '1' and '0' NEC protocol confirming that we are able to send binary commands.

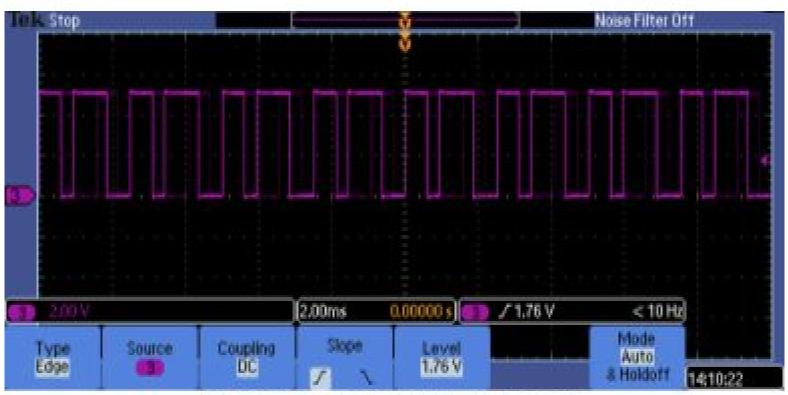


Figure 8: Binary '1' and '0' Send Commands

In terms of the IR LEDs that will be used to get an optimal distance to send the Elimination protocol over, the exact wattage LED is still being decided through distance testing of the received protocol. The goal is to successfully be able to recognize the Elimination protocol 100 feet away from the sent origin. Testing is being done to determine if multiple IR LEDs, for amplification, or a single high wattage LED is more suited for this task in terms of both data quality and power consumption.

B. Peripherals

When initially setting up the LCD it was set up to be programmed by converting RS-232 to UART using a MAX3232. With this setup, the workshop 4 IDE could not see the LCD and was not able to program the board. Tests were then done by using a USB to UART cable to see if the issue resided in the setup with the MAX3232. This setup also could not see the board. At this point, it was decided that to communicate with the LCD, the LCD needed the special programming cable that had access to the reset pin. After receiving the programming cable and looking at the reset pin on the oscilloscope, it was clear that the reset pin was toggled 700 milliseconds before the data was transmitted on the TX line.

When working on implementing the LCD, tests were done to see how often the screen refreshed. To test this, a program was written to have a ball that bounced around the screen and see if we could see any tearing of the ball. In this test, the ball appeared to be flashing after every update but moved smoothly without any tearing of the ball. From this, it was concluded that no moving parts should be displayed on the LCD.

The research was conducted to analyze whether or not the board would also be able to handle all of the IR without using an external microcontroller. After writing test code and trying to get the timers to work within the IR range of 30-60 kHz for the carrier frequency, problems arose with the accuracy of the timer. The LCDs timer resolution was only accurate to the millisecond whereas for the IR carrier frequency the microcontroller needed to be accurate to the microsecond.

C. Player-Sense Technology

a. RFID

To implement Player-Sense using RFID, assume a flag (beacon) will be placed inside of a room. The flag will have an RFID antenna mounted on it so that it can scan for RFID tags within the capture zone. Each player/gun will be fitted with an RFID tag, so that when they walk into the capture zone, the antenna can read the tag, and register that a player is capturing the flag.

Having far read distance, large coverage area, and consistent results were crucial for RFID to be used in creating the Player-Sense technology. The antenna had to read tags within a 5-15 foot radius, as well as read the tags consistently and accurately. If the antenna did not meet these specifications, then RFID could not be used to properly implement Player-Sense.

The SparkFun Simultaneous RFID Reader-M6E Nano was used to test the different requirements that needed to be met. The Universal Reader Assistant software, provided by ThingMagic, was also used to open a COM port and communicate with the M6E. A Max3232 chip was used to convert RS-232 from the COM port to UART logic that the M6E could tolerate. Attached to the M6E was the Taoglas Limited TI.09.A.0 external antenna. On power-up, the board started reading tags. However, the read range was limited to five feet, which did not meet the requirements. It was also noted that the orientation of the tag with respect to the antenna heavily affected the tag's readability. If the tag was vertical, it was able to be read. If the tag was horizontal, the antenna could not read it. This was caused due to the polarization of the antenna being linear.

In searching for a new antenna, the APAES915R80C16-T was chosen due to its high gain of 5dBic (compared to the original antenna of 0dB), as well as right-hand circular polarization. This would solve the aforementioned range issues as well as tag orientation, as circularly polarized antennas can read tags at any orientation. After constructing an apparatus to securely hold the antenna, the read range had increased to 5-6 feet, but it still did not meet the requirements. After consulting Nick Schmidt, it was found that the gauge of the wire that was used to supply power to the M6E was too small, therefore, not allowing enough current to flow. After replacing the 24 gauge wire with 18 gauge wire, as well as adding some capacitors along the power lines, the read distance was increased to ten feet. However, reading was very suggestive, and did not work at all points within a ten-foot radius.

Different capacitors, as well as lower gauge wire, were tested to aid in increasing the read range. However, all of these methods proved useless. The last items to experiment with were the RFID tags themselves. A sample RFID tag pack was bought to test different tags and see if read range was increased. Surprisingly, some tags had read ranges of 15-20 feet. Which solved the read range problem, however the actual reading of the tags was still suggestive, and could not be a reliable source of information for player position data. Due to the fact that two of the three requirements could not be met, it was decided to discard RFID as a solution to Player-Sense, and consider a different technology.

b. Received Signal Strength Indicator (RSSI)

To implement Player-Sense using RSSI, assume a flag (beacon) will be placed inside of a room. Due to the nature of the project, an ESP8266 will already be inside the flag as well as inside of the gun/player to handle communication between subsystems. The module in the beacon, or host module, will ping any devices around it and receive the signal strength of the ping in decibels. Using an in-house mapping function, the distance between the two modules can be acquired, and player position data can be determined. With this data, the system will be able to determine if a player is within the capture zone.

In order to design an in-house mapping function, data needed to be collected and analyzed. Two ESP8266 modules were put on the ground, one foot away from each other. The host then pinged the slave once every second and printed the RSSI data to the console. Fourteen data points were taken, then the distance was increased by one foot. This process was repeated up to seventeen feet. After all the data was collected, the average RSSI per distance measurement was calculated.

During data collection, it was noticed that having a cellular phone near the ESP8266 heavily affected RSSI, returning erroneous values. All cell phones were put into a neighboring room, and the data collection process was started from scratch. Once all averages were computed, they were plotted on a graph of RSSI vs. Distance (see Figure 9). As shown, the results were completely unreliable. Due to the time constraints of the project, it was decided that RSSI would not be a viable solution for Player-Sense.

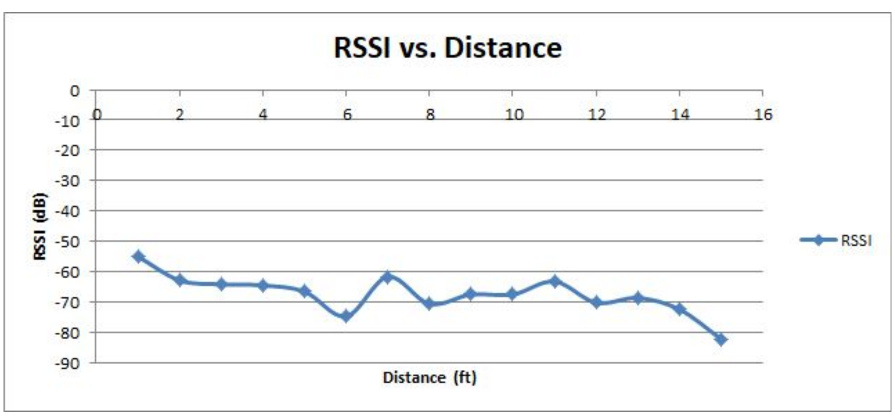


Figure 9. Plot of RSSI vs. Distance of two ESP8266.

c. Infrared (IR)

To implement Player-Sense using IR, assume a flag (beacon) will be placed inside of a room. The flag will have IR LEDs mounted on the top of the beacon. These LEDs will send packets of data within a radius specified by the user. If a player walks into the specified radius or capture zone, then the flag will begin to be captured by the player's team.

When interfacing with the IR LEDs, it was noticed that the signal from the IR receiver was reflecting the data transmitted from the microcontroller (MCU); despite confirming on the oscilloscope that the MCU was producing the correct carrier frequency (56 kHz). Through various tests, it was found that the IR LEDs that were under test did not follow the conventional standard for determining the difference between Anode and Cathode. On most LEDs, the longer lead signifies Anode or positive terminal, and the shorter signifies Cathode, or negative terminal. Once the LED was switched around in the circuit, everything was working correctly, and the data transmission was able to be seen on the oscilloscope in figure Jon2. The blue channel (channel 2) is the output from the IR receiver (TSOP18456), while the yellow channel (channel 1) is the output from the MCU.

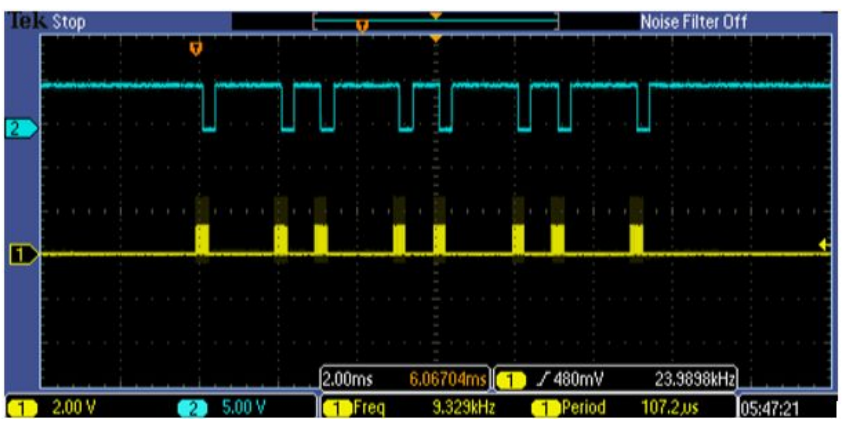


Figure J2. Output from MCU and TSOP18456

6. Parts List

A. IR Gun

a. IR Active Receiver TSOP4440

i. Vin 5V.

ii. Carrier Frequency 40kHz

b. IR LED

i. View angle 30 Degrees
ii. 50 mW
iii. 100 mA

c. ATmega128 - Main Processor

- 128KB Flash, 4KB SRAM, 4KB EEPROM
- Two 16-bit and 8-bit timers
- Two UART channels
- 53 I/O pins.

d. GEN4-ULCD-28DT

- Diablo 16 Microcontroller
  - 32Kb of SRAM
  - 16-bit microcontroller
  - Timer resolution of 1ms
  - 16 I/O pins
  - 16-bit PWM
  - 160 MHz clock
- 16-bit Audio output
- 240 x 320 Resolution
- 4-5 V operating Range
- Resistive Touch Screen

e. Speaker

- .25W, 8 Ohms

f. ESP8266 -Communication

- Wi-Fi communication module
- 17 I/O pins
- 32-bit Microcontroller
- 80 MHz clock default can be doubled to 160 MHz
- 1 MB of flash

B. Flag

a. ATmega128 - Main Processor

- 128KB Flash, 4KB SRAM, 4KB EEPROM
- Two 16-bit and 8-bit timers
- Two UART channels
- 53 I/O pins.

b. ESP8266 - Communication

- Wi-Fi communication module
- 17 I/O pins
- 32-bit Microcontroller
- 80 MHz clock default can be doubled to 160 MHz
- 1 MB of flash

c. NeoPixel Ring -24

- 24 RGB PWM controlled LEDs

7. Technical specifications

A. Flag

1. The onboard RGB LEDs will reflect current team ownership, and/or configuration settings.
   a. If no team owns the flag, then the LEDs shall be white.
   b. If the blue/red team owns the flag, the color shall be blue/red respectively.
   c. While setting up the game, the LEDs shall blink or flash different colors according to setup instructions/tutorials.
2. The IR LEDs will broadcast a one-byte long opcode in all directions that contains the flag ID and current flag ownership if any.
3. The flag will adjust the broadcast range, or capture area, of the IR LEDs based on user input. The maximum broadcasting range will be a 15ft radius from the center of the flag.
4. When a player enters the capture area, the flag shall communicate this to the rest of the players so the LCD can reflect the changes in the environment.
5. When a player leaves the capture area, the flag shall communicate this to the rest of the players so the LCD can reflect the changes in the environment.
6. To capture the flag, players must stand in line-of-sight of the IR LEDs on the flag.
7. Players must wait inside the capture area for T seconds to capture the flag. T will be a function based on the number of players of both teams inside the capture area.
8. The flag must be able to operate on a battery with no glitches for a minimum of 1 hour under “normal” use.
9. The flag must be able to be picked up and moved around while the game is “in session” without any glitches in the game flow or flag operation.

B. LCD

1. LCD must be able to display information going on in the current game.
   a. Captured Flags
   b. Team Points
   c. Ammo
   d. Eliminations
2. LCD needs to be able to communicate with the esp8266 to receive information to be displayed.

3. Be able to input SSID and password to connect to external network.

IR Gun Code Spec

1. IR eliminate protocol will consist of an 8 bit message containing team, player number, and damage points.
2. Carrier frequency for IR protocol will be in the range of 39-41 khz.
3. Active IR receivers will be placed on each gun for both elimination protocols and flag recognition.
4. MCU will take the signal sent from the active IR sensor

IR LED

A. TBD (Needs to transmit a signal 100 feet)

1. Use multiple LEDs to amplify

   –Low or high wattage TBD

2. Use one high wattage LED

3. Use one LED and overload

   –Pulse very quickly

IR Receiver

A. Active 40 kHz carrier carrier frequency.

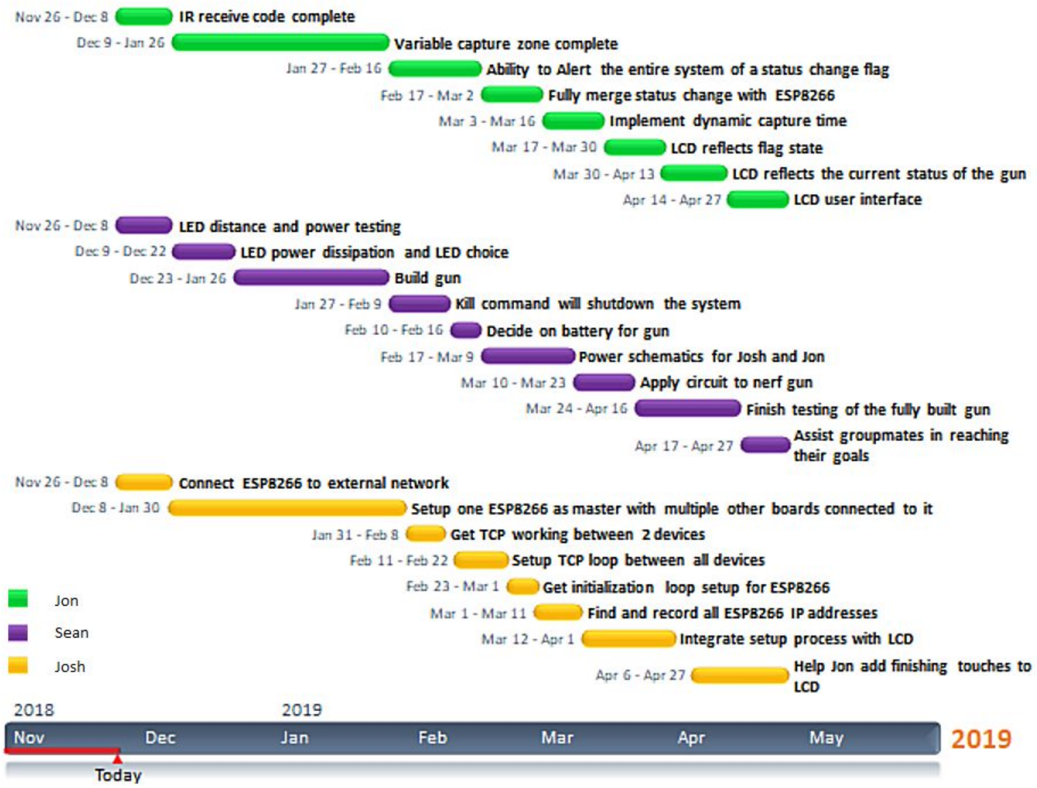
   a. Active IR receivers will be placed on each gun for both elimination protocols and flag recognition.

B. Active 50 khz carrier frequency.

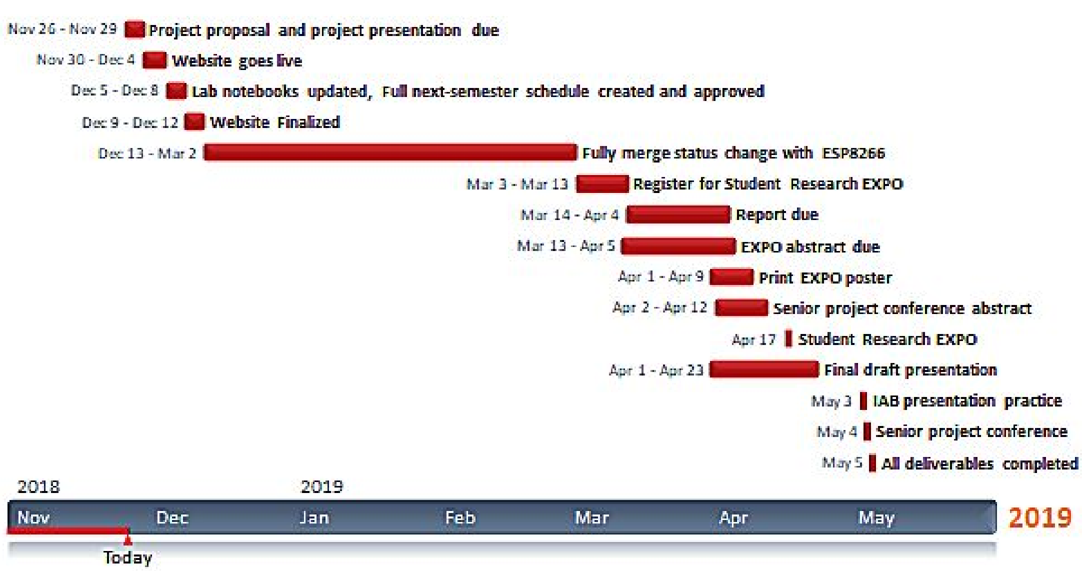
C. IR receivers are active.

8. Schedule

Individual Schedule



Group Schedule



8. Summary

While laser tag and IR is not new technology, the way its used and implemented to create a better gaming experience for the user can become very complex but innovative. Allowing for a more realistic playing style brings laser tag beyond the scope of a child's game and allows for agencies such as the military and law enforcement to use it as an ideal model for real-life situations. Implementing a user interface that is both innovative and informative and creating a dynamic environment that keeps track of location data on players will revolutionize the way laser tag is played.

This laser tagging system will include x2 laser guns, and 1 flag that are both operational and communicating with one another. Power systems will be also be created for portable use of both the flag and guns.

In conclusion, the preliminary results on research and development show that this project scope and deliverables are both feasible and achievable in the time allowed. Our group holds the right to change project components as long as deliverables and project scope is still achieved.

9. References

1. Bergmans, San. “NEC Protocol.” *SB-Projects - IR - NEC Protocol*, www.sbprojects.net/knowledge/ir/nec.php.
2. “JQuery(".Christmas-Banner’).Load('/Get-Banner'); NEC Infrared Transmission Protocol.” *Common ESD Prevention Techniques | Online Documentation for Altium Products*, techdocs.altium.com/display/FPGA/NEC+Infrared+Transmission+Protocol.
3. “Remote Sensors | Earthdata.” *NASA*, NASA, 18 Nov. 2018, earthdata.nasa.gov/user-resources/remote-sensors.
4. *TSOP44-ND*, 8 Feb. 1997, www.vishay.com/docs/82459/tsop48.pdf.
5. *ESP8266EX Datasheet*. Dec. 2005, cdn-shop.adafruit.com/product-files/2471/0A-ESP8266__Datasheet__EN_v4.3.pdf.
6. *ATmega-128*, 2011, ww1.microchip.com/downloads/en/DeviceDoc/doc2467.pdf.
7. *4D Systems*, 2016, www.mouser.com/ds/2/451/gen4-uLCD-28D_datasheet_R_1_3-951996.pdf.