# “G-TAG!”
# GPS Enabled, IR Based Laser Tag System

## Project Proposal

**Team**
Bob Bray
Jerry Davidson

**Project Website**
www.cs.utah.edu/~rbray/project.html

# Table of Contents

## INTRODUCTION

Our project goal is to develop an IR based, GPS enabled, outdoor laser tag system that can download scoring information and GPS data to a PC base station in order to graphically replay the game events. This system will consist of multiple "taggers", modeled after various science-fiction like rifles. These taggers will emit the Infra-Red beams, track shots fired and hits taken, and record the location and time at set intervals. Inspired by the military's MILES 2000 system and other Laser-Tag systems, the "G-Tag!" system additionally employs a GPS interface and a base-station uplink. Once a game session has ended the taggers will send their scoring information and GPS data to a PC based base station. The game events can be reconstructed and displayed graphically, allowing the players involved a new level of feedback, analysis, and fun.

## MOTIVATION

We have various interests we want to pursue including: imbedded systems, microcontroller/PC integration, GPS, and graphics programming. The "G-Tag!" system seems like a fun and engaging way to explore and demonstrate these interests.

## SUCCESS INDICATORS

We have broken down several statements that define the successful completion of our project goal to be the following:

1. Transmit an IR beam encoded with information from an emitter to a sensor 50 meters apart.
2. Transfer data from the GPS unit to the MCU, interpret the information, and store the information in memory.
3. Increment the "shot count" field on the LCD screen when the trigger is pulled.
4. Increment the "hit count" field on the LCD screen when the IR message hits the sensor.
5. Transfer Scoring data and GPS data to the base station.
6. Our software, using the scoring and GPS data, graphically replays the game events.

## PROJECT TASKS

Our project is broken down into the following modules:

1. Core Module – The micro-controller and the LCD display.
2. Emitter/Sensor Module
3. GPS Module
4. Uplink Module
5. Enclosure
6. Replay Software

In order to build and integrate these modules we have defined several project tasks.

**Core Module** – To build the core module we will first decide whether to base our system on the Motorola 68HC11 MCU or the PIC16F648A-I/P. We will then design a basic

circuit that will allow us to program the MCU, receive output from the MCU, and display output to the LCD display. Then, using spec sheets, reference manuals, etc. we will create sample data for the E/S, GPS, and Uplink modules and make sure the MCU inputs and outputs the data properly.

**Emitter/Sensor Module** – To build the E/S module we will design a circuit incorporating a TSAL6200 infra-red LED,TSOP4840 infra-red sensor, trigger and a double-convex lens. After building and testing, we will integrate the E/S module with the core module.

**GPS Module** – The GPS module will consist of the GPS chip and parts/circuits to communicate with the core module. Once built and tested we will integrate with the core module.

**Uplink Module** – The uplink module will consist of a serial or USB connection from the tagger to the base station.

**Enclosure** – the enclosure will be built from sheet aluminum and will appear as a science-fiction type rifle.

**Replay Software** – the replay software will be run on a PC. It will take the data sent from the tagger including scoring information, player movements, shots fired, and shots taken, and graphically replay the game events.

## TESTING AND INTEGRATION

The majority of our tests will begin in the lab with mock data. It is imperative that we create a robust and bug-free communication system in which our MCU sends information to a terminal. Once this has been done, we can use this system in our testing. We will send to our MCU mock data that we will create based on datasheets, reference manuals, or other material. We will create mock data for all units including the GPS, IR emitter and sensor, and LCD.

For input units, we will send the mock data to our MCU as if the device were actually connected. The test harnesses will consist of reading the data, interpreting the data, then outputting the interpretation to a terminal. Satisfaction will be reached when the MCU makes correct interpretations for each set of mock data.

For output devices such as the LCD and emitter, we will send data out to the terminal as if we were sending the output to the device. Except we will send the information in ascii format in order to compare against data sheets and reference manuals. After this stage of testing, we will connect the components and run further tests to ensure proper integration.

The IR emitter will need to be configured to send a signal to communicate up to 80 meters away. Therefore, we need to perform needed calculations to determine power requirements. We will send a signal through the lens to a target 50 meters away and

check the sensor to see if it was hit. We can then proceed to check various ranges up to 80 meters.

The game replay software will reproduce the game events visually on a laptop or desktop including player motions, hits, shots fired, and scores. To test this, we will create mock data for a game scenario. We will reproduce the sample scenario with our software and check for accuracy.

## GROUP COMMUNICATION PLAN

We will meet together as a team each Tuesday and Thursday at 10:00 am. We will briefly review the status of the project, present completed tasks, identify any current problems or issues, and plan the work to be done that day. The rest of the time will be spent actually working on the project. We will log the results of the day to the project webpage.

## MILESTONES

Below are listed our milestones, the task leader for each milestone, and the due date.

| Milestone | Task Leader | Due date |
|---|---|---|
| Create Sample Data | Bob Bray? | May 20, 2005 |
| Build Core Module | Bob Bray? | June 17, 2005 |
| Build Emitter/Sensor Module | Jerry Davidson? | July 1, 2005 |
| Integrate Core Module and E/S Module | Jerry Davidson? | July 15, 2005 |
| Build GPS Module | Bob Bray? | August 5, 2005 |
| Integrate Core and GPS Modules | Jerry Davidson? | August 26, 2005 |
| Build PC Uplink Module | Jerry Davidson? | September 23, 2005 |
| Integrate Core and Uplink Modules | Bob Bray? | October 7, 2005 |
| Build Enclosure | Jerry Davidson? | October 21, 2005 |
| Create Graphical Replay Software | Bob Bray? | December 2, 2005 |
| Debugging | Bob Bray? | December 9, 2005 |
| Documentation | Jerry Davidson? | December 9, 2005 |

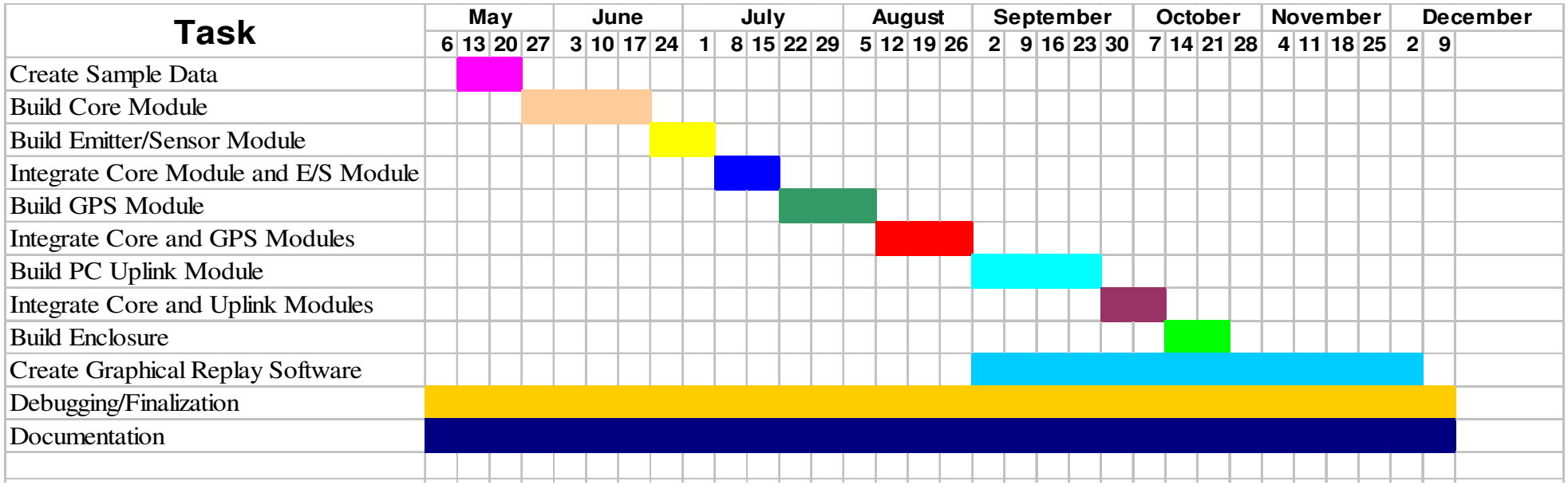
## BILL OF MATERIALS

| Item | Vendor | Risk |
|---|---|---|
| GPS | Synergy Systems | low |
| IR Emitter | Mouser | low |
| IR Sensor | Mouser | low |
| LCD | Mouser | low |
| MCU | School | low |
| Board | PCB123 | low |
| Enclosure | Self made | low |
| Lens | Edmund Scientific | low |

Various discrete components

## CONCLUSION

**SCHEDULE**


Task
May
June
July
August
September
October
November
December
6 13 20 27 3 10 17 24 1 8 15 22 29 5 12 19 26 2 9 16 23 30 7 14 21 28 4 11 18 25 2 9
Create Sample Data
Build Core Module
Build Emitter/Sensor Module
Integrate Core Module and E/S Module
Build GPS Module
Integrate Core and GPS Modules
Build PC Uplink Module
Integrate Core and Uplink Modules
Build Enclosure
Create Graphical Replay Software
Debugging/Finalization
Documentation