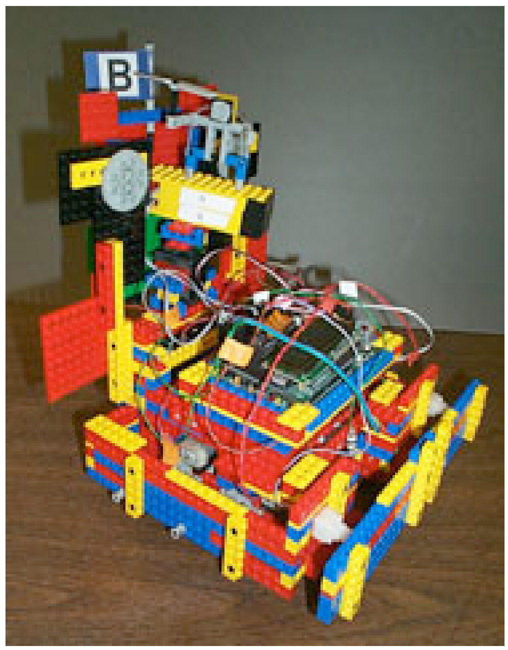
Integrating Bluetooth onto LEGO robots

Progress Proposal
February 7, 2003

Prepared for
Prof. Sree N. Sreenath and Prof. Behnam Malakooti



Prepared by

Team Members

Andrew Jones

Mike Krofcheck

Faculty Advisor(s)

Frank Merat
Associate Professor, EECS

Richard Drushel
Senior Research Associate, Biology

Project Description

EECS 375, commonly referred to as the "LEGO lab class", has become a popular course offering in the EECS department. In fact, the course has become so popular, that the course instructors, Dr. Richard F. Drushel, Dr. Hillel J. Chiel, and Dr. Randall D. Beer, have to keep a waiting list for the course.

Modeled after the original LEGO lab course, MIT 6.270, designed by Fred Martin, Randy Sargent, Anne Wright, P.K. Oberoi, *et al*. at Massachusetts Institute of Technology, EECS 375 employees a C-programmable 68HC11 microcontroller board which provides significantly more control and functionality than the LEGO Mindstorms set utilized in ENGR 131. However, the microcontroller boards are much more fragile than the robust and durable Mindstorms RCX "bricks." Through the use of 411 students over the last 15 semesters of the class's existence, the microcontroller boards have received a fair amount of use. Due to this heavy use, the boards are now beginning to show their age; components are beginning to break off the boards, cables are coming loose, and the general functional limitations of the boards have resulted in several "copy cat" designs from semester to semester. The instructors of the course are worried that many of the microcontroller boards used in the course won't be able to make it more than another one to two semesters. In addition, the original board suppliers have informed the instructors that production has ceased on the current board design used for the course.

Due to this apparent need, the student group consisting of Andrew Jones and Mike Krofcheck, have decided to continue the task of redesigning the microcontroller board for EECS 375 started during the Fall '02 semester. However, the project is to go beyond just a simple redesign of the board; three new element will be added to the microcontroller board—Bluetooth communication, a USB port, and an updated microcontroller--that will foster additional student creativity in EECS 375.

Currently, there is a transition to what some industry experts have dubbed the "wireless age." Cell phones, Wi Fi (802.11b) Local Area Networks (LANs) , and new short-range wireless technologies such as Bluetooth are beginning to transform how people communicate and interact with computers and other electronic devices.

To aid students in making the transition to wireless technology and to foster additional creative robot designs, the student team will add Bluetooth communication capabilities to the microcontroller utilized in EECS 375. The wireless communication capabilities provided by Bluetooth will allow robots designed by students to communicate with each other and with a computer to download program updates. It will also eliminate the use of serial cables currently used to program the robots. However, a more modern wired serial architecture that provides greater bandwidth, USB, will also be provided on the redesigned microcontroller board for faster serial access if so desired.

Due to the amount of work involved in such an undertaking, the project was originally broken into two semesters. The first part, conducted during the Fall '02 semester, involved investigating the wireless communication options available, deciding on a standard wireless protocol (or an original protocol), creating a design to demonstrate the wireless capabilities, and prototyping the wireless design in an effort to prove its feasibility for inclusion on a redesigned microcontroller board suitable for use in EECS 375. Due to circumstances outside of the student group's control, the student group was unable to complete all the objectives outlined. After conducting research on the available wireless protocol options, Bluetooth was chosen for inclusion on the microcontroller board due to its low cost, low power requirements, and high data bandwidth. However, acquisition of the Bluetooth modules was more difficult than originally anticipated. As a result, the second part of the project will involve integrating the Bluetooth modules onto the current microcontroller board; skipping the design of a board to demonstrate Bluetooth's capabilities. At the conclusion of the Spring '03 semester, the proposed plan will provide the following deliverables:

- A modified microcontroller board currently used in EECS 375 containing USB and Bluetooth
- Schematics for a redesigned microcontroller board used in EECS 375 that includes Bluetooth connectivity, a USB port, and an updated microcontroller; time permitting, the redesigned board will be constructed
- Software for both the computer and microcontroller board supporting the new technologies implemented on the redesigned microcontroller board

Project Objectives

Since the project involves both a hardware aspect and a software aspect, two separate requirements sets have been created.

The final software will meet the following objectives:

- The microcontroller software as well as the software on the computer will have the ability to identify other Bluetooth devices and differentiate between devices. Differentiation includes the ability to decipher robots from the same team and opposing teams.
- Direct connections will be allowed between Bluetooth devices (i.e. computer-to-micro and micro-to-micro).

- Computer software will provide an interface to program and control one or more microcontroller boards using either Bluetooth or a USB connection

The hardware will meet the following objectives:

- Communication interfaces: Bluetooth for wireless and USB for wired connections
- Two wireless devices (i.e. two robots) should be able to communicate with one another without interaction with computer.
- Wireless device (i.e. robot) should be able to communicate with a computer in both directions. Communication functions should include, but are not limited to, downloading microcontroller code (student-written programs) to the wireless device, sending signals to the computer from the wireless device, and receiving control commands from the computer.
- Wireless communication should occur at 500 Kbps or enough bandwidth to allow for 5 messages to be sent between the computer and wireless device every second (5 messages/s).
- 25 ft. circumference zone of communication; enough range for LEGO egg competition, which takes place in EECS 375, and for common updating of microcontroller code. Refer to Figure 1 for an illustration of the egg hunt arena.
- Low interference rate (i.e. packet collision rate) with other common devices (i.e. cell phones, high frequency cordless phones, Wi Fi networks, etc.)
- Abide by FCC regulations where applicable

- Wireless connectivity should have a low rate of power consumption (battery considerations on microcontroller board). Bluetooth chip should have low power mode. Operation in 50 mA range with low power mode in 100 to 200 uA range. Voltage input between 1.5 and 3 volts. The robots should be able to continue to operate non-stop for a length of time similar to current performance after wireless technology has been added.
- Low cost solution (< $150 per robot)
- Ensure data integrity and security during wireless transmission between two or more devices. Teams should not be able to "eavesdrop" on the commands sent by other teams.
- Leverage (as much as possible) current microcontroller board solution (hardware and software)

Strategy for Achieving Objectives

During the first part of the project, conducted during the Fall '02 semester, the senior project group conducted research into previous work with Bluetooth technology and past attempts at integrating wireless technology onto the microcontroller boards used in EECS 375.

Based on the research conducted by the project group, it has been determined that the best approach to solving the problem as follows:

1. Research into microcontroller and Bluetooth module architecture.
2. Integrate Bluetooth onto microcontroller development board to test optimal integration methods

3. Integrate Bluetooth module with current microcontroller board; writing the necessary software for the microcontroller.
4. Test the integrated solution to ensure the specifications are met.
5. Write supporting software for the computer allowing for student written code to be downloaded to the microcontroller board.
6. Redesign the current microcontroller board to include the Bluetooth module, a USB port, the updated microcontroller, and replace out of date parts.
7. Verify the redesigned microcontroller board using simulation software (PSpice).
8. Build redesigned microcontroller board if time permits

Team Complement

The senior project team consists of two members: Andrew Jones and Mike Krofcheck. Both Andrew and Mike will partake in the hardware and software roles of the project, although Andrew will have a larger percentage contribution to the software required for the project. Andrew Jones has been selected to act as the team leader.

Requirements

The final software will meet the following requirements:

- The microcontroller software as well as the software on the computer will have the ability to identify other Bluetooth devices and differentiate between devices. Differentiation includes the ability to decipher robots from the same team and opposing teams.

    - Bluetooth stack provided with Bluetooth module provides necessary calls to differentiate between different devices.
    - Understand stack and harness necessary function calls
- Direct connections will be allowed between Bluetooth devices (i.e. computer-to-micro and micro-to-micro).
    - Bluetooth allows you to select the device you establish a link with
- Computer software will provide an interface to program and control one or more microcontroller boards using either Bluetooth or a USB connection
    - Learn Cocoa programming language for MacOS X
    - Create Graphics User Interface wrapped around the microcontroller programming software

The hardware will meet the following requirements:

- Communication interfaces: Bluetooth for wireless and USB for wired connections
    - Research USB and Bluetooth
    - Integrate Bluetooth and USB onto microcontroller board using serial driver chip (MAX232) as go-between.
- Two wireless devices (i.e. two robots) should be able to communicate with one another without interaction with computer.
- Wireless device (i.e. robot) should be able to communicate with a computer in both directions. Communication functions should include, but are not limited to, downloading microcontroller code (student-written programs) to the

wireless device, sending signals to the computer from the wireless device, and receiving control commands from the computer.

- Wireless communication should occur at 500 Kbps or enough bandwidth to allow for 5 messages to be sent between the computer and wireless device every second (5 messages/s).
    - Bluetooth provides for 1Mbps communication speed
    - Perform validation scheme to confirm bandwidth achieved
- 25 ft. circumference zone of communication; enough range for LEGO egg competition, which takes place in EECS 375, and for common updating of microcontroller code. Refer to Figure 1 for an illustration of the egg hunt arena.
    - Bluetooth provides for 30 ft. circumference of communication
    - Perform validation scheme to confirm communication distance
- Low interference rate (i.e. packet collision rate) with other common devices (i.e. cell phones, high frequency cordless phones, Wi Fi networks, etc.)
    - Bluetooth devices change frequency 1500 per second, reducing probability of packet collisions
- Abide by FCC regulations where applicable
- Wireless connectivity should have a low rate of power consumption (battery considerations on microcontroller board). Bluetooth chip should have low power mode. Operation in 50 mA range with low power mode in 100 to 200 uA range. Voltage input between 1.5 and 3 volts. The robots should be able to

continue to operate non-stop for a length of time similar to current performance after wireless technology has been added.

  - Bluetooth module selected meets outlined power requirements

- Low cost solution (< $150 per robot)
  - Use standard parts that are readily available to keep costs down
- Ensure data integrity and security during wireless transmission between two or more devices. Teams should not be able to "eavesdrop" on the commands sent by other teams.
  - Bluetooth provides Safer+ security scheme that will be implemented
- Leverage (as much as possible) current microcontroller board solution (hardware and software)

Plan

The design team's project plan is illustrated using a Gantt chart created in Microsoft Project; this chart will be used to track the progress of team tasks. The Gantt Chart appears at the end of the report. The project is broken down into two main sections: hardware and software.

The hardware phase is broken into four phases: order and obtain parts; add Bluetooth to the microcontroller board, add USB to revised microcontroller board, layout and build microcontroller if time. Each phase contains several sub-phases. These sub-phases will be performed by Andrew Jones and Mike Krofcheck, symbolized by "AJ" and "MK" within the Gantt Chart representation, respectively.

The software phase is broken down into five sub-phases: obtain necessary hardware to programming microcontroller; modify microcontroller code for BT; create user libraries for robots; code validation; create GUI for computer. These tasks will also be performed by Andrew Jones and Mike Krofcheck.

Verification

Hardware validation will be performed using several methods. Test LEDs will be placed on prototype board. The status of these LEDs will indicate the status of various signals and chips. Individual chip signals can be checked with an oscilloscope or DMM for integrity and accuracy. The prototype will also have a self-check mode. If a problem exists, the LCD display will indicate the problem. Verification of code, downloaded to the prototype, will be performed by comparing the code in Flash memory with the original code on the host computer.

Software validation will be performed inline with development. This type of verification follows accepted software development practice. This type of testing requires that software is debugged and functioning properly before moving on to the next stage of development. The specific tests that will be run on the software will begin by feeding the application known data inputs, which will give known outputs. This will allow for the software team to verify that the application is processing data correctly. Further testing then will verify the application's robustness to handle skewed input from a user.

Implications

At this time, without more research, we anticipate that our project will have economic effects if all goes well. If we produce a viable design for wireless implementation and integrate it into a redesigned microcontroller board, we might be able to market the board to other universities with courses similar to EECS 375.

Reporting Schedule

ITEM	DEADLINE DATE
Project Definition due	Fri, February 7 by noon
Weekly progress reports & periodic project reviews	Reports due each Fri by noon
Midterm Report due	Fri, March 7 by noon
Ethics paper due	Fri, March 28 by noon
Final Project Reports due	Fri, April 25 by noon
Final Oral Presentation	Mon thru Wed, April 25, 28, 29, 30

Budget

Due to the fact that we are still in the initial stages of our project, our budget estimate is very rough. However, we do anticipate several expenses and hope that vendors will be willing to provide some supplies free of charge or for a discounted price.

ITEM	ESTIMATED COST
Chips (RF chips, memory, etc…)	$100
Protoboard	$20
Power Source (batteries)	$10
Misc. (wires, antenna, connectors, etc…)	$20
TOTAL	***$150***

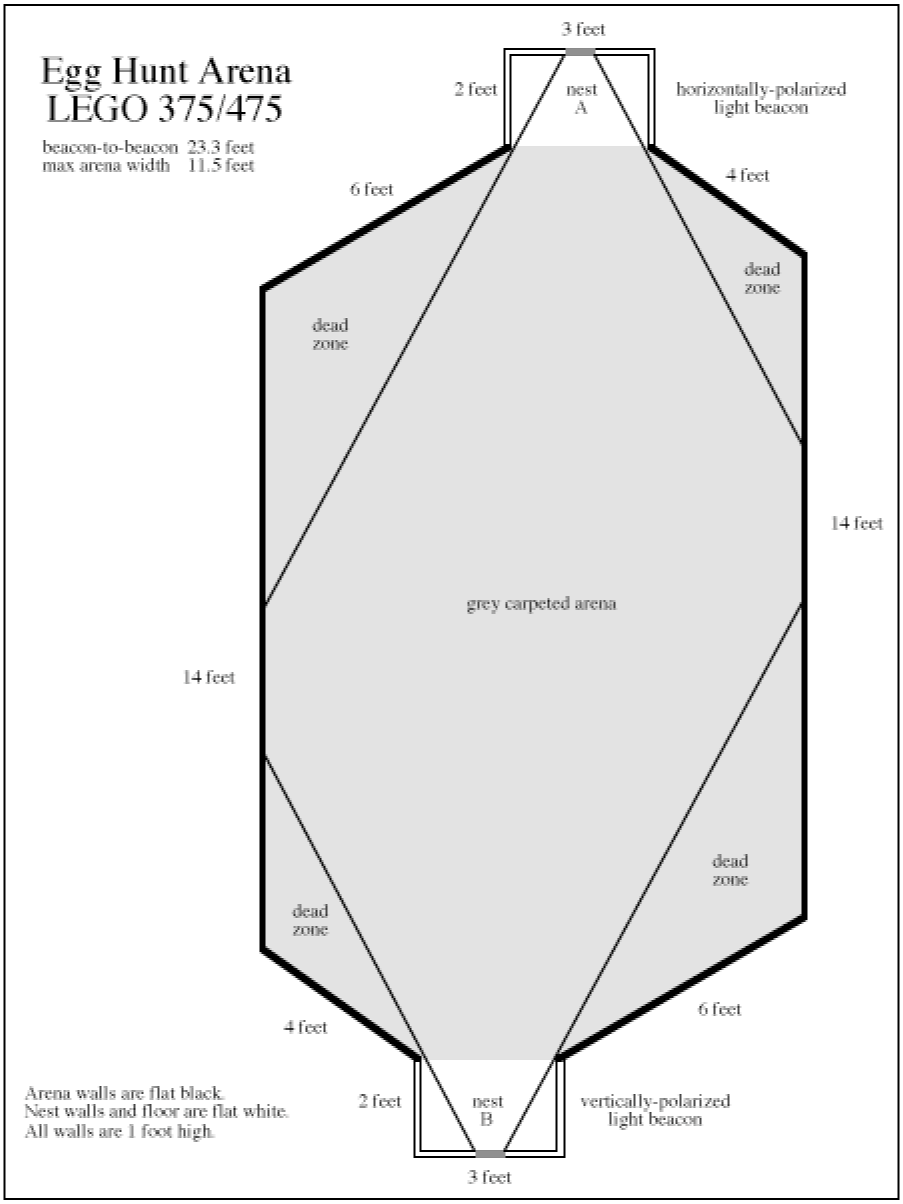


Figure 1: Arena used for EECS 375 competition