University of British Columbia
Electrical and Computer Engineering
EECE 281/282

Electromagnetic Tether Robot

Introduction

For the second project of EECE281/EECE282, you will design, build, program, and test a small autonomous robot. The robot must be battery operated and controlled using a microcontroller system. The robot must keep a fixed distance from an electromagnetic beacon. If the beacon changes position, the robot must adjust its position to keep a constant distance from the beacon.

Project Requirements

The project **must** include the following components and/or functionality:

1. **LP51B Microcontroller Systems:** For this project, you are required to use the LP51B board you assembled in the previous project.
2. **Battery operated:** Your robot must be battery operated. Batteries are neither provided in any of the kits nor will be provided to you at any moment because they are very expensive. You need to buy your own batteries. An AA battery holder is included in the project kit.
3. **Robot construction:** You can use any material you find available for the chassis of the robot (paper, cardboard, wood, plastic, metal, etc.). You can also use the materials and tools available in the MCLD workshop (the technicians may charge you for expensive material). If you plan to use the workshop, you need to come with a sketch of the work you want to do. If you do not do so, the technician(s) in charge may kick you out of the workshop! Additionally, you are required to wear safety glasses and safety shoes while in the workshop. The parts listed below have been used successfully in similar projects in the past and are included in the project kit.

Part #	Description
Solarbotics GM4	Gear Motor 4 - Clear Servo
Lynxmotion Servo Wheel	2.63" x 0.35" (pair)
Tamiya 70144	Ball Caster
4 x AA	Battery holder
1 x 9V cable	9V battery clip
Unfolded chassis	Aluminum chassis made using the water jet cutter.

You can buy the project kit for 40$ and pick it up in MCLD112b.

4. **C programming:** The code for this module must be completed using the C programming language. The integrated development environment CrossIDE available in the course's web page includes a C compiler that works for the AT89LP51RB2 microcontroller. A few C programming examples are provided with CrossIDE.

5. **MOSFET drivers:** To drive and control the motors of your robot you must use MOSFETs (Metal Oxide Semiconductor Field Effect Transistors). Both NMOS and PMOS transistors were included in the course parts kit. If you find the motors are too noisy (electronically noisy, that is) you may want to consider isolating them by using optoisolators to control the MOSFETs. The LTV847 optoisolator was provided in the course parts kit. Tip: your robot should be able to move forward and backward.
6. **Transmitter:** While one microcontroller system is used to drive the robot, another microcontroller system is used to transmit an electromagnetic signal or beacon. The robot reads this signal using a couple of inductive sensors and determines its distance and direction from the beacon adjusting its position accordingly. Additionally, the transmitter must be able to send commands to the robot. Your design must support at least these commands:

   a) Move closer. When the user presses a button in the transmitter, it commands the robot to move closer.

   b) Move farther. When the user presses a button in the transmitter, it commands the robot to move farther.

   c) Rotate 180^o. When the user presses a button in the transmitter, it commands the robot to rotate 180^o.

   d) Parallel park. When the user presses a button in the transmitter, it commands the robot to parallel park in a space that is 1.5 times the length of the robot.

   At least four preset distances should be available in your robot for the “move closer” and “move farther” operations.
7. **Distance**: The robot must be able to track the transmitter when they are separated by at least 50 cm. The farther your robot can track the transmitter the better. To design your transmitter/receiver you can use Microchip’s application note 232: “Low-Frequency Magnetic Transmitter Design” available at

   http://ww1.microchip.com/downloads/en/AppNotes/00232B.pdf

Project Evaluation

The evaluation of this project consists of a demonstration (worth 35% of the final mark) and a written project report (worth 5% of the final mark). In the project demonstration, your design is evaluated using the following criteria:

Mark	The project is:
10	Exceptional, did everything it was supposed to do well, plus maybe some additional functionality. The project looks great and it has **original/innovative ideas**!
9.0-9.5	Did everything required. Circuitry / project well designed and constructed. It could use a little improvement. The project has some original/innovative ideas.
8.0-9.0	Did everything required, lesser quality but still worked. The project lacks originality/innovation.
7.0-8.0	Mostly worked, not entirely, not the greatest design.
6.0-7.0	Didn't really work, ok design but didn't really come together.
4.5-6.0	Didn't work, not very good design.
0.5-4.5	Didn’t work, poor design.
0	What project?

The project demonstration evaluation will be carried out by the course instructor and/or one or more laboratory TA(s) on April 3 & 4, 2014. This demonstration should not take more than 4 or 5 minutes, if everything is working reasonably well. The project report is due by April 7, 2014. You need to submit the project report using the format described in the following section.

Project Report Format

The project report should be written for a reasonably expert reader such as a project manager (an engineer) in a company for whom you might have designed this prototype product. The project report should have sufficient detail that someone skilled in the art could reproduce or improve upon your results. The number of pages for the report should be ≤ 20 (not including the title page and appendices, double spaced, ‘Arial’ or ‘Times New Roman’ font size 12 for text, and ‘Courier New’ font size 8 or 10 for the source code, approximately one inch margin for the top, bottom, left, and right margins) and include the following sections:

1. **Title Page –** It should include the course name and number, instructor name, section, project name, group number, names and student number of the students in the group, and the date of submission.

2. **Table of Contents**

3. **Introduction –** Design objective and specifications. Overview of the overall design approach including **system block diagrams for both the hardware and software designs**.

4. **Investigation –** This section must include the following subsections:

A. **Idea Generation** – Describe how your group generated ideas and working hypotheses.
B. **Investigation Design** – Describe how you group perform the design investigations involving information and data gathering, analysis, and/or experimentation.
C. **Data Collection –** Describe how your group used appropriate procedures, tools, and techniques to collect and analyze data.
D. **Data Synthesis –** Describe how your group synthesized data and information to reach appropriate conclusions.
E. **Analysis of Results –** Describe how your group appraised the validity of conclusions relative to the degrees of error and limitations of theory and measurement.

5. **Design –** This section must include the following subsections:

A. **Use of Process –** Describe how your group adapted and applied general design processes, accesses to design systems and components, or processes to solve open-ended complex problems as relevant for this project.

B. **Need and Constraint Identification –** Describe how your group identified customer, user, and enterprise needs, and applicable constraints.

C. **Problem Specification –** Describe how your group specified additional design requirements based on needs and constraints presented in the point above.

D. **Solution Generation –** Briefly describe potential design solutions suited to meet functional specifications. If possible, include even those that didn't work!

E. **Solution Evaluation** - Perform systematic evaluations of the degree to which several design concept options meet project criteria. Clearly explain why you choose the final design.

F. **Detailed Design –** Explain how you applied appropriate engineering knowledge, judgment, and tools, in creating and analyzing design solutions. This has to be one of biggest parts of the report. In this section you must include the description and evaluation of each block (e.g. "A-stable Circuit", or "Counter Initialization"): Describe the approach taken to design each block. For circuits, include a detailed circuit diagram and describe how it works. For programs, include the source code in the appendices, and refer to it while you describe it.

G. **Solution Assessment –** Describe how you assessed the design performance based on requirements, needs, and constraints. This section must include an **evaluation of the complete system** by mean of tests you carried out including plots of performance, reproducibility numbers, tables, etc. as judged appropriate for this project. Describe how each relevant part of your design was tested and the testing results. Also in this section, the strengths and weaknesses of the design must be pointed out.

6. Live-Long Learning – Identify a specific learning need or knowledge gap. For example, did your learn something new by yourself? You didn't take a course that would have helped with the project? You found that one of the courses you took was particularly useful for the project?

7. Conclusions – Summarize the design and functionality of your project. Summarize also the problems you encounter, and how many hours of work the project took.

8. References – A specific book, paper, datasheet, or website is referred to in the **body** of the report at the point at which you say something about it, by a numerically-ordered, square-bracketed number, the first one being [1]. Then, at the end of the Report in a section called **REFERENCES** located just before the **Appendices** section**,** the same square-bracketed number is followed by the Author List, Article Title, Journal or Book Title, Volume, Number, Pages, ISBN Number, Publisher, Date of Publication. Although the Reference list can be listed alphabetically by author, instead we do not recommend this for an Engineering Report. With an alphabetical listing, the location in the body where any particular reference is discussed is then hard to find, since the references are no longer in order of appearance. Examples of references are [1] and [2] (note that the numbers in the square brackets here refer to the appropriate numbers in the Reference list). The Reference list itself might look like:

> **REFERENCES**
> [1] Smith, J, and F. Jones, "Designing an universal logic circuit", Journal of Impossibly Wonderful Electronic Circuits, v.3, n.1, pp. 21-35, March, 1910.
> [2] Jones, F and J. Smith, "Why universal logic circuits are impractical" , …

9. Bibliography – Items in a section at the end of a report called **BIBLIOGRAPHY** are NOT referred to in the body of the report. It is a list of appropriate background or additional reading and is located after the **References** section and before the **Appendices** section. The items in the Bibliography are usually ordered by last name of the first author. It is sometimes appropriate to have BOTH a Reference list and a Bibliography list. An example Bibliography looks like:

> **BIBLIOGRAPHY**
> • Sedra, A., and K.C. Smith, Microelectronic Circuits, 4th Edition, Oxford University Press, 1998.

10. Appendices – Supporting documents such as extensive theoretical analyses, mechanical drawings, and source code. Your source code should be properly documented and indented. Do not append datasheets, compiler manuals, or other already published material to the report.