

Musical Instrument Project

Submitted by: Richard Bennett
University of Tennessee
rbennet2@utk.edu

Rocky Top Team Project

Objectives

- Apply concepts of waves, sound, and engineering design
- Practice idea generation skills
- Effectively work in a team to accomplish a goal

Task

Design and build 2 simple musical instruments capable of playing Rocky Top (one verse and chorus).

Both instruments must be built from scratch from readily available materials (no kits allowed).

1. Instrument 1: The objective for this instrument is to build something for which you can calculate the frequencies. The calculations should show the theoretical frequency of each part (e.g. string, tube, etc) of the musical instrument. You may need to make some measurements to get the necessary data to make the calculations. The method of collecting the data should be described in the report. Any adjustments you needed to make to your instrument to get a better tone should be described. You should include references you used to obtain formulas for the calculations. There are a lot of good web sites to help with this. Simple pan pipes are NOT allowed for Instrument 1. An example of a bucket bass is given below.
2. Instrument 2: The objective for this instrument is to build something creative for which you probably will not be able to quantitatively analyze. A qualitative analysis is required. Here is an example of a [K & B Construction drum set](#).

Priorities

1. Functionality - does it work? in tune? loud enough?
2. Minimize cost
3. Originality

Deliverables

- October 24, in class: Short (2 minute) description of your project (no powerpoint); demo of both instruments playing the tune simultaneously.
- October 24, midnight: [Documentation](#)
 - Report (pdf)
 - Instrument picture (jpg)
 - Team picture (jpg)
 - Optional: Excel or Matlab file with supporting calculations.
- October 24, midnight: [Team member contribution](#)
- October 24, midnight: [Peer Review](#)

Report Contents

- Introduction
- Overview of final design, including list of materials
- Design details and calculations
- Results and conclusion
- References

References

- [Rocky Top music](#)
- [Musical note to frequency mapping](#)
- [Notes on treble clef](#)
- [Hyperphysics - Musical Instruments](#)

Example



- [Sample Report](#)
- [Sample Calculations](#)

Grading

- 10 points - Pitch
- 10 points - Loudness
- 10 points - Creativity
- 5 points - Instrument and team picture (yes/no)
- 5 points - Report in pdf format (yes/no)
- 5 points - Problem description
- 10 points - Overview of final design
- 30 points - Design details and calculations
- 10 points - Results and conclusions
- 5 points - References

Newspaper Bridge Project

Submitted by: Richard Bennett
University of Tennessee
rbennet2@utk.edu

Recitation 1.5 Statics Team Project



Objectives:

- Review concepts of statics through building a bridge
- Practice idea generation skills
- Effectively work in a team to accomplish a goal

Team Member	Contact Info (e.g. cell phone, email, Facebook).

Task 2: (5 min) Problem Statement

You are going to build a bridge using only newspaper and masking tape. The bridge must be strong enough to support a 10 pound weight located at the center of the bridge while minimizing the amount of material used. A 6 in. high x 12 in. wide object must be able to pass underneath it. The bridge must be free-standing; not attached to the wall, a piece of furniture, a person or an article of clothing. Each team will turn in a short report. Requirements of the report are given on the next page. It would be wise to read through these now.

Task 3: (5 min) Propose feasible solutions

Without discussion with other teammates, each person is to sketch out a potential solution. Strength, stability, and aesthetics should be considered in your design. You can use the whiteboard for your sketches.

Task 4: (5 min) Decide on a solution

Discuss your potential solutions with your teammates. Based on your discussion, decide on a bridge to build. This could be one of the individually developed solutions, a combination of the proposed solutions, or even an entirely new solution.

Task 5: (40 min) Build your bridge and write your report.

Using a maximum of 300 g of newspaper, and masking tape, build your bridge. You need to also write your report in this time. Figure out how to work efficiently as a team. The report needs to include a paragraph indicating why you chose the design you did, and free body diagrams showing the path the load would take to the support of the bridge. Working beyond the 40 minutes results in a disqualification.

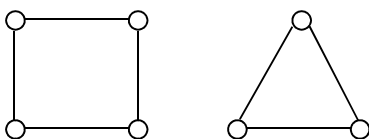
Task 6: (10 min) Test your section's bridges

Each bridge will be tested to see if it meets the requirements. Bridges will be weighed, and the lightest bridge that meets all other requirements will be declared the winner.

Task 7: (10 min) Finish report and turn in results

- Turn in a short report (1-2 pages).
- Fill out the web-based form to report on your team experience.
- Fill out the web-based form to report on your team experience (repeated since many people forget)

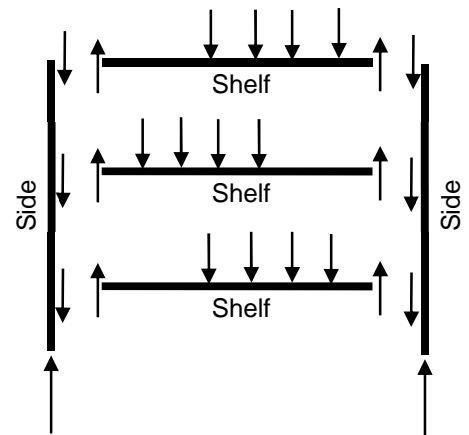
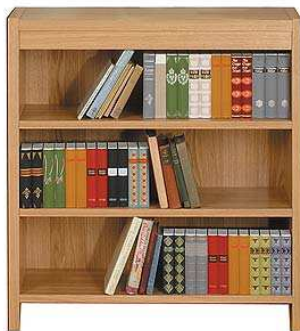
Notes on stability: Which of the following shapes is stable? Make sure you have your bridge properly braced in all directions. Lack of stability is the number 1 cause of bridges failing, both newspaper bridges and real bridges. Notice in the picture how the bridge is failing due to lack of bracing.



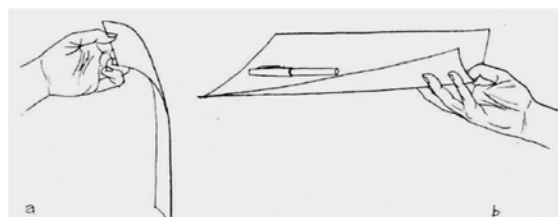
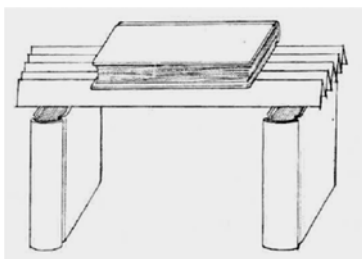
Team Project: Newspaper Bridges Grading Rubric

Grade Item	Points
Format: names of team members, problem statement, all requirements (including results) included, report well organized	10 – Excellent 8 – Good 6 – Fair 3 – Poor
Neatness: Clear writing and drawings.	10 – Excellent 8 – Good 6 – Fair 3 – Poor
Clear description of bridge, including a good sketch of your final design, free body diagrams showing the force path through the bridge, and a short paragraph of why this design was chosen.	40 points total 10 points sketch 15 points free body diagrams 15 points paragraph
Results: include results (failed, held), bridge weight, conclusions	15 points
Performance: 25 points <ul style="list-style-type: none"> • Bridge held 10 lbs (15); 7.5 lbs (12); 5 lbs (10); 2.5 lbs (8); 0 lbs (5) • 1st place: +10 points; 2nd place: +6 points; 3rd place: +3 points • 6 in. high x 12 in. wide object did not pass underneath bridge: -5 points • Excellent aesthetics: +5 points 	

Example of free body diagrams showing load path for a bookcase.



Fun Sketches (maybe even hints)



Roller Coaster Project

Submitted by: Richard Bennett
University of Tennessee
rbennet2@utk.edu

Recitation 3.4 Roller Coaster Project

Objective: Review the basics of conservation of energy and projectile motion while working together as a team to build a “roller coaster”

Materials:

- 4 ft x 8 ft sheet of pegboard
- re-usable plastic ties
- 0.0078 lb 3/8 in. diameter steel ball
- 25 ft of 5/8 in. ID 1/8 in. thick PVC tubing
- magnet.

Tasks:

0. Record your team member names and emails here:

Name	e-mail	Other info (e.g. Facebook, etc)

1. Build a roller coaster using the tubing and ties. Requirements of the coaster are:
 - A. The full 25 ft of the tubing must be used (tubing cannot be cut)
 - B. There must at least two loops in the coaster run.
 - C. The tubing cannot be more than 6 inches outside the frame.
 - D. The frame must be used horizontally. (8ft side parallel to the floor)

2. Collect the data on the attached sheet. This will be used to create an energy vs. distance plot in Excel.

3. Contest

Each coaster will be placed on a table and a TA will start the coaster down the track. Record the horizontal distance that the coaster goes beyond end of the track. The team that goes the furthest is the winner (extra credit for participation grade).

4. You must disassemble your coaster before leaving. **DO NOT CUT THE TIES.**

5. Create a plot in Excel that shows the energy vs. distance along the coaster track.
 - A. Set the datum for gravitational potential energy at the bottom of the pegboard. Determine the initial mechanical energy of the coaster.
 - B. Set the coaster on a table. Determine the final speed of the coaster based on projectile motion calculations. This can be done by knowing the angle from the horizontal that the coaster leaves the track, the height above the floor, and the distance the coaster travels before hitting the floor.
 - C. Based on the final speed and final gravitational potential energy, determine the final mechanical energy.
 - D. Plot the initial and final mechanical energy, and assume that energy losses vary linearly with distance along the track (i.e. connect the initial and final energy points with a straight line).
 - E. Plot the gravitational potential energy for every foot along the track on the same graph.
 - F. Based on the total mechanical energy (including losses) and the gravitational potential energy, plot the kinetic energy on the same graph.
 - G. From the plot of the kinetic energy, determine the maximum speed your coaster had during its run.

6. **Upload one Excel file for each team.** This file should have the one required graph, and include all the calculations.

7. **Each individual team member must fill out a web form.**

Data Collection Form:

Position Along Track (ft)	Height above Datum (ft)
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	

Final speed data:

Launch height above floor: _____

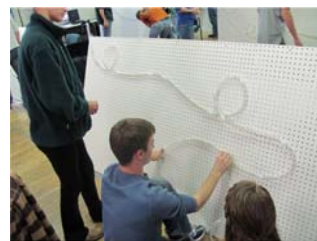
Angle of launch: _____

Horizontal distance traveled: _____

Roller Coaster Project Feedback Form

1. Enter the names of the other members on your team. (Note: every individual must complete this form.)	1. _____ 2. _____ 3. _____ 4. _____ 5. _____
2. What was the final speed of your roller coaster? (ft/sec)	_____
3. What was the maximum speed of your roller coaster? (ft/sec)	_____
4. Identify the strengths and weakness of each of your team members and describe individual contributions to the project. Include your contribution.	
5. Describe the strengths and weaknesses of your group. How could you have improved your effectiveness?	
6. Describe how your group choose the design you did.	
7. Give one suggestion for how to make this project better.	

Pictures of Roller Coaster Project



Rube Mini-Shooter Project

Submitted by: Richard Bennett
University of Tennessee
rbennet2@utk.edu

Rube Mini Shooter

Objectives:

- Solve an open-ended problem while working as a team.
- Demonstrate ways that engineers communicate - presentations, spreadsheets, written reports
- Apply principles you've learned in EF 151 and EF 105
- Have fun

Task:

Design and build a miniature [Rube Goldberg device](#) made out of readily available items that will use a complicated series of actions to trigger a final action of shooting a marble into a cup.

Inspiration:

[Melvin the Mini Machine](#)

Requirements:

- Your device must consist mainly of readily available objects.
- The final action of your device must be shooting a marble into a basket. The basket must be at least 15" away (horizontally) from where the marble is launched.
- A 2ft square piece of pegboard will be provided for each team. Use of the pegboard is optional.
- Your disassembled device must fit within a 2ft by 2ft by 1 ft imaginary rectangular volume.
- Your device (when assembled and ready to run) must fit entirely within an imaginary 3ft by 3ft by 2ft rectangular volume.
- Once started, your device must operate successfully without interaction.
- Your device must have at least 5 different steps.
- Your device must go from is disassembled to its ready to run configuration in less than 60 seconds.
- The operating time of your device must be more than 15 seconds and less than 60 seconds.
- You will complete this project outside of class time, but you may use recitation time for meeting and planning.
- Estabrook 13 will be available as a work and storage area - here are the [guidelines](#) for the use of this room.
- Your team is limited to a budget of \$20. **This cost is calculated by taking the cost difference between your used item list and your recycled item list.**
- The device must be able to be operated in a safe, clean, and non-destructive manner in Estabrook 111 (e.g. no fires, explosions, sharp projectiles, high-speed projectiles, etc.).

Suggestions:

- While we want complicated and creative - one of the challenges is to come up with a device that works but does not require a lot of time to develop.
- You are encouraged to use readily available materials.
- Divide up the work among your team members.
- Don't forget about reliability and repeatability - it is better to have a simpler device that works rather

than a complex device that doesn't.

- Try to come up with actions that are creative and that are visually appealing and that don't happen too quickly to be seen.
- Using stored mechanical energy (springs, rubber bands, things falling) is OK, but consider using other forms of energy (electrical, chemical, wind)
- Ask if you have questions.

Deliverables:

- **Tue Nov 13** (recitation)
 - A preliminary plan
 - Summary of how you have divided your tasks
 - Turn in during recitation
- **Tue Nov 27** (recitation)
 - A status report
 - Summary of your progress to date - describe what has been done and what is left to do. Include pictures if available.
 - Turn in during recitation
- **Tue, Dec 4** (recitation)
 - [Dropbox](#)
 - Presentation (.ppt or .pptx)
 - Device picture (.jpg)
 - Team picture (.jpg)
 - Project [web form](#) started with information from above.
 - Device designed and ready to assemble and demonstrate
 - Team presentation (5 minutes)
- **Wed, Dec 5** (midnight)
 - [Dropbox](#)
 - Final report (.pdf)
 - Device picture
 - Team picture
 - Device video - a video of your device being set up and successfully run.
 - Project [web form](#) completed
 - [Feedback form](#): Team members' participation and contribution
 - [Feedback form](#): Project rankings

Grading:

In general all team members will receive the same grade, although individual adjustments will be made for those students who do not participate in their team's work.

- **Preliminary Plan and Task Division:** (5%)
[An example plan](#)
- **Status Report:** (10%)
- **Device Design:** (10%)
Your adherence to the project and cost guidelines are evaluated here.
- **Device Creativity and Re-use:** (10%)
Be creative. A marble rolling down a wooden ramp isn't very creative. A McDonald's toy rolling down a calculus book demonstrates a little creativity and re-use.
- **Presentation:** (15%)
Each team will make a 5 minute presentation during testing day. [Details, rubric.](#)

- **Report:** (20%) ~1000 words
Each team will turn in a written report that provides documentation of your project. [Details](#), [rubric](#), [analysis example](#).
- **Device Performance:** (15%)
Did your device work the first time and function as intended? Did it complete the task? Was it able to function independently?
- **File upload:** (5%)
Any person on your team can upload files and enter information about your project. Doing this means that will allow us to automatically create an on-line gallery of all the projects. This portion of the grade is team credit for completing this task.
- **Team Member Evaluation:** (5%)
Each student is required to fill out a short form describing their team members' participation and contribution. This portion of the grade is individual credit for completing this task.
- **Project and Presentation Rankings:** (5%)
Each student is required to fill out a short form ranking their sections' devices and presentations. This portion of the grade is individual credit for completing this task.

Updates/Corrections

- Nov 6: Original version
- Nov 12: Add details to cost calculations

Cross – Fertilization Classroom Practice

Cross – fertilization is a creative process by which you bring ideas and concepts from one field or specialty to another.

For example, to practice generating ideas by cross-fertilization, ask yourself what would an engineer and beautician (hairdresser, hair stylist) learn if they went to lunch or dinner together that would improve themselves and/or the way they perform their jobs:

Engineer learns from the Beautician

- A. _____
- _____
- B. _____
- _____
- C. _____
- _____
- D. _____
- _____
- E. _____
- _____

Beautician learns from the Engineer

- A. _____
- _____
- B. _____
- _____
- C. _____
- _____
- D. _____
- _____
- E. _____
- _____

Many other combinations of professions would also provide growth experiences for both participants. The cross-fertilization of ideas from one group to another is a powerful method for adapting ideas from one discipline or profession to solve problems in another discipline or profession. Many times managers will bring together a small group of people from diverse (ethnic, cultural) backgrounds to interact and look at a problem and solution from many vantage points.

Other combinations of professions could include: repair person, florist, dentist, accountant, police officer, soccer coach, car designer, custodian, bellhop, cruise ship activity director, Cub Scout or Girl Scout leader, bakery café manager, construction inspector, architect, nurse, doctor, flight attendant, pilot, housekeeper, etc.

Name of Partner 1: _____

Name of Partner 2: _____

Profession 1: _____ Engineer _____ Profession 2: _____

Profession 1 _____ Engineer _____ gives to Profession 2 [_____]

A. _____

B. _____

C. _____

D. _____

E. _____

Profession 2 [_____] Gives to Profession 1 _____ Engineer _____

A. _____

B. _____

C. _____

D. _____

E. _____

EGR 100 PRESENTATION ASSIGNMENT – TRANSITION FROM HIGH SCHOOL TO ENGINEERING STUDY

Describe your transition from high school through your first five weeks as an engineering student in college in a 5-minute presentation. The presentation should be engaging, educational and creative to the audience (your classroom peers). Multimedia is encouraged (powerpoint, prezi, video, etc.) Presentations are scheduled for November 15, 2012.

Items for inclusion in your story:

--Which of the sections/topics of your textbook have made an impression on you? By impression, we mean, those that have led you to take a different course of action than you've previously taken or thought differently than you have previously.

--As result of your applying what you have learned from the sections/topics, what specific strategies are working for you or not working for you?

--What are your major challenges to implementing the strategies?

--Choose One:

What advice would you give yourself when you were a high school senior preparing for college? What advice would you give the seniors in your high school now?

--What are your next steps and goals?

A quality project will meet the following criteria:

- **Creativity** in media/method of expression
- **Level of engagement:** Is this interesting for the class? Were you able to keep the class' attention during your presentation? Note: While everyone loves chocolate, use of candy and similar incentives should not be your method of maintaining class interest.
- **Substantive and concrete expression** of how material/concepts were integrated into your experience, actions, thoughts/perceptions, and the results that came from your using the strategies. Use of quality examples, analogies, etc. is only one of many ways this can be achieved.
- **Synthesis:** Presentations are expected to go beyond Recall, Understanding, and Analysis stages and should go into the Synthesis stage. Review ch. 3, page 116 – Synthesis is a hybrid of both “Evaluating” and “Creating” categories (ex: simply *stating* what sections helped you and what resulted is basic “Recall”, and therefore is not sufficient).
- **Self-reflection:** this presentation is about you and your journey. Therefore, avoid using third person language (ex: do not use “you” and “your”...instead use “I”, “me”, etc.). Your classmates been given “advice” all quarter long – now it's time for you to talk about *yourself* and share *your* experiences in transitioning into college engineering.

Timing/Technical:

- **For the sake of time, presentations will be cut off at 5 minutes max.** There will be a 1 minute transition period between presentations.
- Please ensure that you know how to operate all equipment involved, that all technical equipment is working, etc. **Test your presentation before/after class prior to the day of the final.**

- 1) Media/Method of delivery
- 2) Synopsis of project – what should we expect when we see your presentation?
- 3) Relevant sections included
- 4) Detailed timeline for completing the project

PRESENTATION PROGRESS REPORT

Progress report due on blackboard & presented (1-2 min.) on Tuesday, November 12th

NAME _____ DATE: _____

Project Title: _____

Report needs to include:

Provide a synopsis of your presentation. What chapter and or sections are referenced in your story? Describe what medium you plan to utilize to tell your story. Who is the audience for your story?

Action items: List action items -- action items are all tasks and responsibilities which must be completed.

Completion schedule -- when tasks and responsibilities are expected to be completed (include testing the media on the classroom computer).

Presentation Synopsis

Please write a synopsis of your presentation to be due 1:30 PM, Tuesday, Dec. 4. The synopsis gives the viewer an over view of the entire presentation. The presenter tries to cover the vital points included in the presentation. The synopsis must have the following headings:

- 1) **Title** – This should be self-explanatory, right?
- 2) **Introduction** - Give a brief statement of the problem or issue your presentation focuses on explaining its importance and significance.
- 3) **Scenario** – Describe the story/game/PowerPoint/Prezi, etc. you are telling or explaining. Who are the characters? What is the setting? What is the message you are conveying? Who are the actors? What sections of the book does it reference?
- 4) **Material and Methods** – What does your presentation consist of? Is it a video, a sonnet, game, etc? What kind of video/game? Describe the tools you used to make your presentation. What is the length of the presentation?
- 5) **Conclusion** – Describe your experience in the preparing and completing the presentation. What worked and what did not work for you? What improvements would you make? How would you grade your presentation based upon effort, clarity, creativity, meeting the guidelines of the assignment?

Please submit your completed synopsis on MS Word via BB Assignments as an attachment.

Submitted by: Tim Grebner
Century College
Tim.Grebner@century.edu

ENGR 1020
Introduction to Engineering
Design of a Small Scale Electric Generator

GOAL:

Your team will design and construct a demonstration device that shows how electricity is produced from a generator. The generator will include a coil of wire, moving magnets, and a way to display that electricity is being produced from the device. You must use the Fab Lab to create the device. If successful, your device will be used in science classes at local middle schools; consequently, your device must be robust, easy to use, safe, and attractive.

OBJECTIVES:

1. Understand and use the engineering design process.
2. Exercise creativity and communication within a team effort.
3. Communicate design concepts in written and graphical forms.
4. Think like an engineer.

PROCEDURE:

Follow the design procedure outlined in the Oakes text. Maintain a notebook and **record everything** you do in that notebook, including meetings, sketches, brainstorming sessions, information searches, etc. **All team members must maintain their own notebooks.**

DELIVERABLES:

- Team Contract and Gantt Chart
- Status Report Update
- Design Report (see description below).
- PowerPoint presentation that describes your product design and development.
- Design notebooks from each team member.
- Finished Device

Design Report: Produce a word-processed design report that has a cover page, abstract, table of contents, introduction, report body, conclusions and recommendations, and appendix. A detailed description of the contents and format of the report can be found on the D2L website. Briefly, the body of the report should describe your design process and include the following sections:

1. Discussion of need and definition of problem with a clear problem statement
2. Search for information. Include your sources and what you found.
3. Constraints and design criteria.
4. Alternative solutions – **at least 3!** show hand drawn sketches of design alternatives
5. Subjective analysis of alternative solutions.
6. Decision – include a decision matrix
7. Specifications of selected design.
 - ◆ SolidWorks drawing
 - ◆ Assembly instructions
 - ◆ Complete bill of materials

CONSTRAINTS:

- All materials must be from the fab lab or readily available from a local hardware store. The design should be easily reproducible at another fab lab.
- The completed design must fit within a 12"x12"x12" box.
- Your budget for materials is \$70. Keep track of what you spend and provide documentation of all costs.

MATERIAL COSTS:

The following items will be provided to you at no cost. However, you must still count these items in your budget analysis.

- Wire (5 cents per foot) 200 ft maximum
- Magnets \$3 per magnet (maximum of 4 provided)
- LEDs 25 cents per LED
- Panel meter \$10 per meter
- Acrylic \$10 per square foot. You will be provided 4 square feet of acrylic.

EVALUATION:

The design report will be evaluated for completeness, form, writing quality and grammar. Your presentation will be evaluated on content, your delivery, and the quality of your slides. The notebooks will be graded on an acceptable/unacceptable basis. The finished device will be graded based upon how well your product satisfies the design criteria and constraints. The device will be evaluated by hands on testing and by peer and faculty evaluation. Extra points may be awarded to outstanding products.

GRADING SCHEME:

- Physical realization of device. Design and construction. 20 points
- Written Report20 points
- Oral Report5 points
- Design Notebook5 points
- Extra Credit points may be added at the discretion of the instructor.

Lego Robot Project PickNPlace

Submitted by: Tom Lombardo
Rock Valley College
T.Lombardo@RockValleyCollege.edu

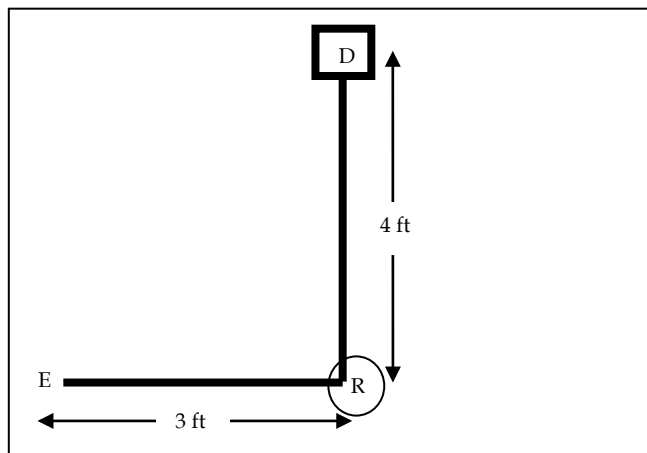
This project is worth 225 points total, broken into multiple assignments:

- Preliminary Design Document = 100 points
- Testing Report = 25 points
- Demo and final documents = 100 points

You will design and program a robot that will:

1. Begin at the robot starting point
2. Pick up a whiteboard eraser
3. Move it to the destination
4. Pick up another whiteboard eraser
5. Move it to the destination
6. Make a sound and stop.

Map:



E = Eraser (position not marked on the floor)

R = Robot starting point

D = destination (marked with a black box on the floor)

— = black tape on the floor

Eraser dimensions:

H = 5"

W = 2"

D = 1.25"

Project Teams

(Project teams are named for famous engineers. I talk about their contributions to society when I introduce the project. When I run two sections, I also have DaVinci, Edison, Lamarr, Turing, Armstrong.)

Hopper List student names here

Kamen

Nye

Wozniak

Rutan

Tesla

Individual Points

Each team member will have a particular responsibility for the project. Your individual points will be determined by how well you complete your part of the project. See the individual job descriptions (starting on the next page) for details.

If you miss any class time during this lab, your individual points will be deducted proportionally. (This includes absence and lateness.)

The following pages describe the different responsibilities for this project. As a team, you decide which person fills each responsibility.

Each person will create one or more files. Be sure to follow the specified filename format, and post your files on the college's online learning management system. See the timeline on the last page for details about posting your files.

Although we are dividing up the responsibilities, you are expected to assist your team in any way necessary.

Electronic Engineer: This person is responsible for determining which sensors are used, and finding information about how the sensors work, and how they are programmed. All parts of the software related to the sensors will be written by the electronic engineer, including the programming blocks and the comments.

The electronic engineer will write a design specification (Word document*) that includes:

- Your name at the top of the file
- List of possible sensors that could do each job
- How each of those sensors work (in general) – see the “robot sensors” lab
 - What does the sensor “measure”?
 - What is the sensor’s output?
- Pros and cons of each sensor for this particular application
- The sensors that will be used, and the reason for those choices
- How the sensors are programmed with the Mindstorms™ software for this application

For example, navigating through the course could be done by following the black line, like you did in the line-tracker lab. That would use the light sensor. Tell how the light sensor works and the pros and cons of running the course by following the black tape. On the other hand, you could use the rotation sensors to measure how far you’ve gone in each direction, and use that information to navigate the course. Tell how the rotation sensors work and the pros and cons of navigating the course by measuring distance.

Your reports should be written so that if you left the company, another engineer could read your document and easily pick up where you left off.

* TeamNameElectronicDesign.docx (Use your actual team name, not *TeamName*.)

Software Engineer: This person is responsible for the organization and structure of the program. This includes the use of decisions and loops, and the associated comments. The software engineer will write and debug the program, working with the electronic engineer regarding the sensors.

The software engineer will write the software flow in a Word document* before writing the actual software. "Software flow" means to describe in plain English what the robot is doing at any given moment. For example:

1. Move forward until 4" from wall
2. Pick up the object
3. Turn around 180 degrees
4. ...

Write is step-by-step in list form as shown above.

Put your name at the top of the file. The software engineer will use that flow document to help with writing the actual software. Comments in the software can simply be copied from the flow document and pasted into the Mindstorms™ program*. If, during the debugging stage, you change the actual software, you must also change the software flow in the Word document to match it.

Your reports should be written so that if you left the company, another engineer could read your document and easily pick up where you left off.

* TeamNameSoftwareDesign.docx (Use your actual team name, not *TeamName*.)

* TeamNameSoftware.rbt

Mechanical Engineer: This person will design and build the robot, in consultation with the electronic engineer (for sensors). The mechanical engineer will evaluate the pros and cons of different methods of picking up the object. The mechanical engineer will also assist with troubleshooting.

The mechanical engineer will write a design specification (Word document*) that includes:

- Your name at the top of the file
- List of possible ways to pick up the object (with sketches)
- Pros and cons of each possible method of picking up the object
- Choose one method and design the complete robot
- Detailed sketches that show the location of all sensors, wheels, the NXT “brick,” and the part of the robot that picks up the eraser. Show the robot from three (3) angles: front, side, and top.

If you make any modifications to the robot during the production and testing phases, you must change the sketches in the Word document to reflect the changes.

Your reports should be written so that if you left the company, another engineer could read your document and easily pick up where you left off.

* TeamNameMechanicalDesign.docx (Use your actual team name, not *TeamName*.)

For teams with four members, there's an extra responsibility. If your team has only three members, ignore this page.

Technical Writer: This person will keep a journal* of all team activities. The journal will include details about:

- The general approach to this problem – a big-picture overview of the solution. Include sketches of the robot moving through the course, and a narrative of what the robot is doing (decisions, actions, etc.) at various places in the course.
- Tests that were performed during brainstorming and troubleshooting, and the detailed results of those tests.
- Problems that were encountered (including hardware and software failures) and the solutions to those problems

Your reports should be written so that if you left the company, another engineer could read your document and easily pick up where you left off.

* TeamNameJournal.docx (Use your actual team name, not *TeamName*.)

Project Timeline

4/16 (100 points): Preliminary designs. Post preliminary design documents in the Final Team Project Preliminary Design Drop Box:

- TeamNameElectronicDesign.docx
- TeamNameSoftwareDesign.docx
- TeamNameMechanicalDesign.docx
- TeamNameJournal.docx (if applicable)

4/23 – Design Reviews. All teams present their designs to the class.

Design revisions may occur after the presentations.

4/30 (25 points): Assembly and testing. Post updated designs to the Final Team Project Testing Drop Box

5/7 (100 points): Demonstrations. After the demos, add a section to your report describing:

- What worked well
- What didn't work well
- How would you improve the design if you could do it over again?

Post the final version of all documents (design, software) to the Final Team Project Drop Box:

- TeamNameElectronicDesign.docx
- TeamNameSoftwareDesign.docx
- TeamNameSoftware.rbt
- TeamNameMechanicalDesign.docx
- TeamNameJournal.docx (if applicable)

The 100 points for the final team project are comprised of 50 points for the documents, and 50 points based on how well the robot completed the task:

Excellent Completes 2 cycles of pick-and-place perfectly (eraser completely inside the box)	Good Completes 1 cycle of pick-and-place perfectly, or completes 2 cycles almost perfectly	Marginal Nearly completes 1 cycle perfectly	Incomplete
50	40	30	0-20

Submitted by: Mark Holtzapple
Texas A&M University
m-holtzapple@mail.che.tamu.edu

ENGR 112C (526)
Water Rocket Project
Spring 2014

Objective

For each team, the objectives of this project follow:

- Construct multiple versions of the water rocket shown in Figure 1
- Launch the rockets and measure their performance
- Compare the rocket performance to a computer model.

Each water rocket is made from a 2-L soda bottle. The bottle is partially filled with water and also compressed air. As high-velocity water is expelled from the nozzle, the rocket is propelled upward. The payload consists of an altimeter that records the height as a function of time.

Teams will use the same pressure (100 psi), but will vary the following:

- Fill fraction f
- Nozzle diameter D_n
- Launch tube, yes or no

Each team will construct multiple rockets. Tables 1 and 2 show the assigned parameters for each team. Using data from all the teams, the apogee (maximum height), time-to-apogee, and time-to-impact will be recorded as a function of these parameters.

Table 1 Team assignments with launch tube

Fill fraction f	Nozzle Diameter D_n (in)		
	0.515	0.640	0.842*
0.0	1	2	3
0.1	4	5	6
0.2	7	8	9
0.3	10	11	12
0.4	13	14	15
0.5	16	17	18
0.6	19	20	21
0.7	22	23	24

* Throat diameter without nozzle

Table 2 Team assignments without launch tube

Fill fraction f	Nozzle Diameter D_n (in)		
	0.515	0.640	0.842*
0.0	25	1	2
0.1	3	4	5
0.2	6	7	8
0.3	9	10	11
0.4	12	13	14
0.5	15	16	17
0.6	18	19	20
0.7	21	22	23
0.8	24	25	

* Throat diameter without nozzle

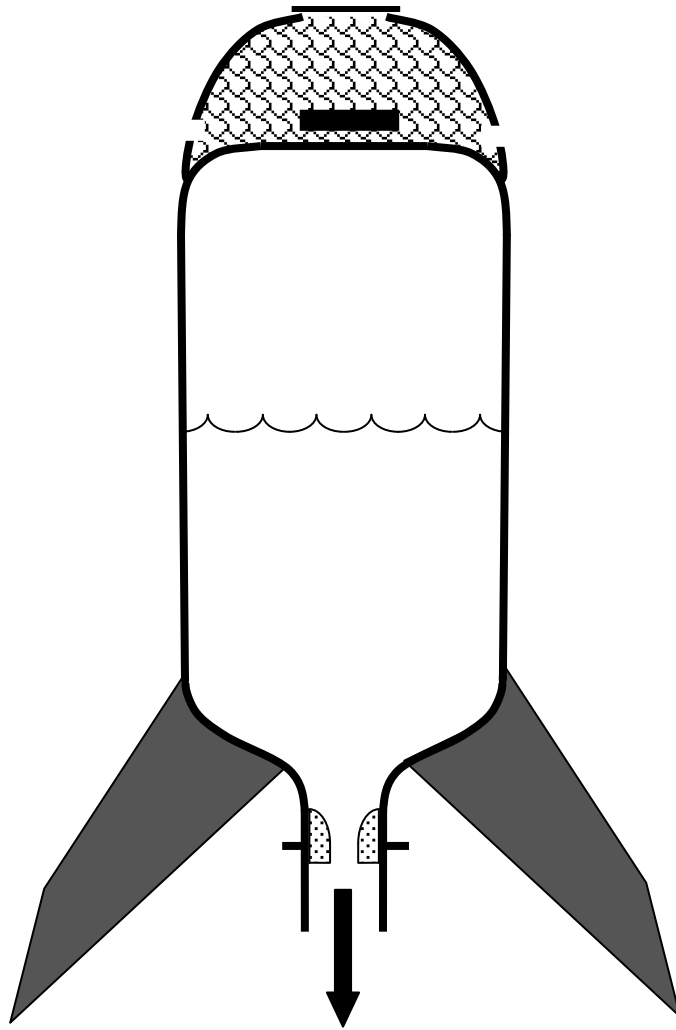


Figure 1 Schematic of water rocket.

Scheduled Activities

Three classroom periods are devoted to the project.

Class 25.1 Project Introduction

- Review project description
- Begin working on computer program

Class 28.1 Project

- Work on computer program
- Build rockets
- Fit nozzles into bottle neck

Class 30.1 Project Demonstration

- Launch rockets
- Bring the following:
 - Safety glasses
 - Bicycle pumps
 - Water-filled rockets (Note: Before filling with water, determine m_{rocket} using classroom scales or Post Office scales.

April 30

- Reports due

Rocket Fabrication

Examine Tables 1 and 2 to determine how many rockets your team will fabricate, and the parameters of each rocket. Secure the following:

Tools

- X-Acto knife or box cutter (**CAREFUL...these are sharp!**)
- Electric drill
- #49 drill bit
- Small-diameter rat-tail file
- Sand paper
- Needle-nose pliers that can fit inside the nozzle to hold it
- Screw driver
- Sharp scissors

Expendable supplies (per rocket)

- Two 2-L Pepsi bottles
- Four #2 self-tapping screws, 1/4-in length (Grainger Item 1LA93)
<http://www.grainger.com/Grainger/Wood-Screw-1LA93?Pid=search>
- Fin material, rigid foam core board, 3/16-in thick [see Note 1]
- Polyester fiberfill (used for making quilts or stuffing pillows)
- Paper towels
- Scotch tape
- RTV silicone adhesive or aquarium silicone sealant (see Note 2)
- Super glue (a.k.a. Crazy glue, cyanoacrylate glue, see Note 2)

Note 1: Alternative materials for the fins can be considered such as (1) Silly Winks foam board from Hobby Lobby, which is a flexible foam board or (2) polyethylene terephthalate (PET) sheet, which is the same material as the bottle and is flexible in thin sheets. When the rocket hits the ground, rigid fins tend to crack off the rocket whereas flexible fins do not. Flexible fins can be secured to the rocket using tabs that alternate left and right (Figure 2).

Note 2: Loctite recommends adhesives for bonding plastics:

http://www.henkelna.com/us/content_data/237471_LT2197_Plastic_Guide_v6_LR7911911.pdf

Pages 52 and 53 describe appropriate adhesives for polyethylene terephthalate (PET), the material used to construct the bottle.

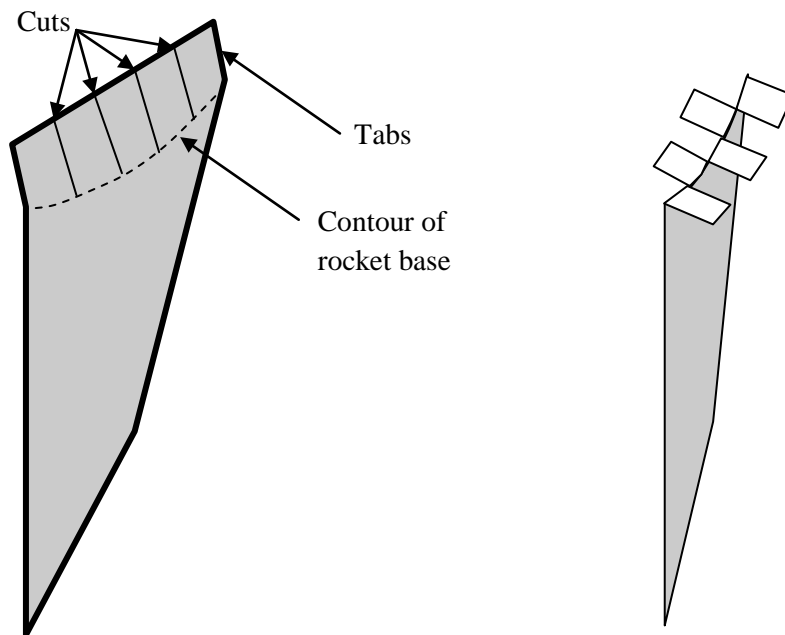


Figure 2 Fins made from flexible material.

Some expendable supplies (e.g., foam core board, polyester fiber fill, screws) are needed in small amounts per rocket. It would be helpful for industrious students to secure these supplies for the whole class, and provide them to individual teams. The details are to be resolved by the student entrepreneurs.

Figure 3 shows details of the rocket design. For each rocket, secure two empty 2-L Pepsi bottles. (*Note:* Other bottles may be used that have identical neck dimensions as the Pepsi bottle.) Save one bottle cap. Using the X-Acto knife or box cutter, carefully remove the plastic cap ring from the outer neck of the bottle, which is a remnant from the bottle cap. Cut off the neck of one bottle to create a hole that is about 1.5 inches in diameter. Cut the top off of this bottle where the cylindrical portion intersects with the round top. Using silicone adhesive or other appropriate adhesive, secure this to the bottom of the bottle, as shown in Figure 3. (*Note:* Roughen the surfaces with sandpaper to ensure good bonding.) The void space between the rounded nose and the bottle bottom holds the payload, which in this case is the altimeter supplied by the course instructor. To ensure the altimeter is not damaged, the void space is lightly packed with polyester fiberfill. To ensure the altimeter does not fall out, the hole at the top is covered with tape.

Altimeters determine height by sensing: (1) barometric pressure or (2) GPS signals. GPS does not have good resolution in altitude, so barometric pressure will be used to sense height. To record the height as a function of time, a MicroPeak altimeter will be employed. Standard models trigger once the height exceeds 30 m. In our case, we will employ a custom version that triggers when the height exceeds 10 m. (*Note:* Once the rocket reaches a height of 10 m, data are recorded from the initial launch to impact.) The data are downloaded to a computer via MicroPeak USB v2. Both these items will be supplied by the course instructor. The following website provides more information:

<http://shop.gag.com/altimeters.html>

As a back-up in case the altimeter fails, each launch will be video recorded so the time-to-apogee and time-to-impact can be determined independently.

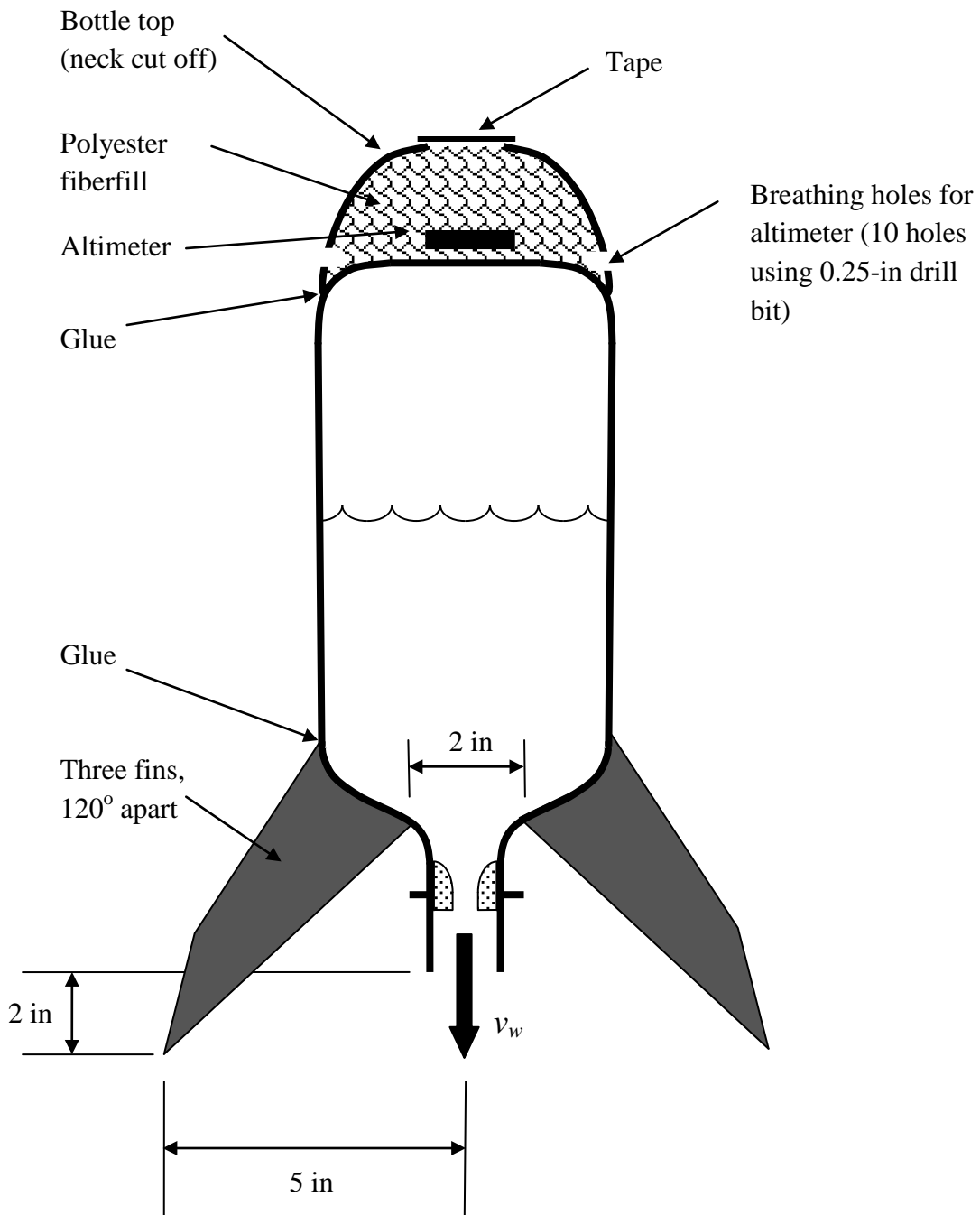


Figure 3 Details of rocket. Note 1: The above fin dimensions must be followed to ensure the rocket mounts properly to the launch stand. Note 2: Pressure readings are more accurate if the breathing holes are located on the cylindrical portion of the nosecone and not the rounded part of the nosecone.

Using a #49 drill bit, drill four holes in the bottle neck, as shown in Figure 4. **(IMPORTANT: Be sure to drill the holes with the dimensions shown in Figure 4 so they interface properly with the nozzle. Only drill holes in the bottles that have nozzles.)**

Using the rapid prototype machine, create a plastic nozzle similar to that shown in Figure 5. The nozzle diameters for your team are specified in Tables 1 and 2. If necessary, use sandpaper to smooth the surfaces. Do NOT drill holes in the nozzle until later.

Insert the nozzle into neck of the bottle. As shown in Figure 6, the nozzle must be located 0.25 inches from the edge of the neck so there is room for the O-ring seal on the launch stand. **(IMPORTANT: Be sure to locate the nozzle as shown in Figure 6 or your rocket will not seal to the O-ring.)** Hold the nozzle in place using needle nose pliers, or wedging it onto a tapered rod or a rat-tail file. Once it is in place, put super glue into the holes in the neck of the bottle and rotate the nozzle to spread the glue. Hold in place until the glue cures. After the nozzle is glued in place, drill matching holes into the nozzle. Use self-tapping screws to ensure the nozzle is mechanically bonded to the throat of the bottle.

The fins are to be constructed of foam core board (or alternative flexible materials) and secured to the bottle. Use super glue or other appropriate adhesive on the edges and fill in the corners with silicone adhesive. Roughening the bottle surface with sand paper will help ensure a good bond. Figure 3 shows the appropriate dimensions for the fins. **(IMPORTANT: Be sure to use the three fin dimensions given in Figure 3 or your rocket may not fit the launch stand.)**

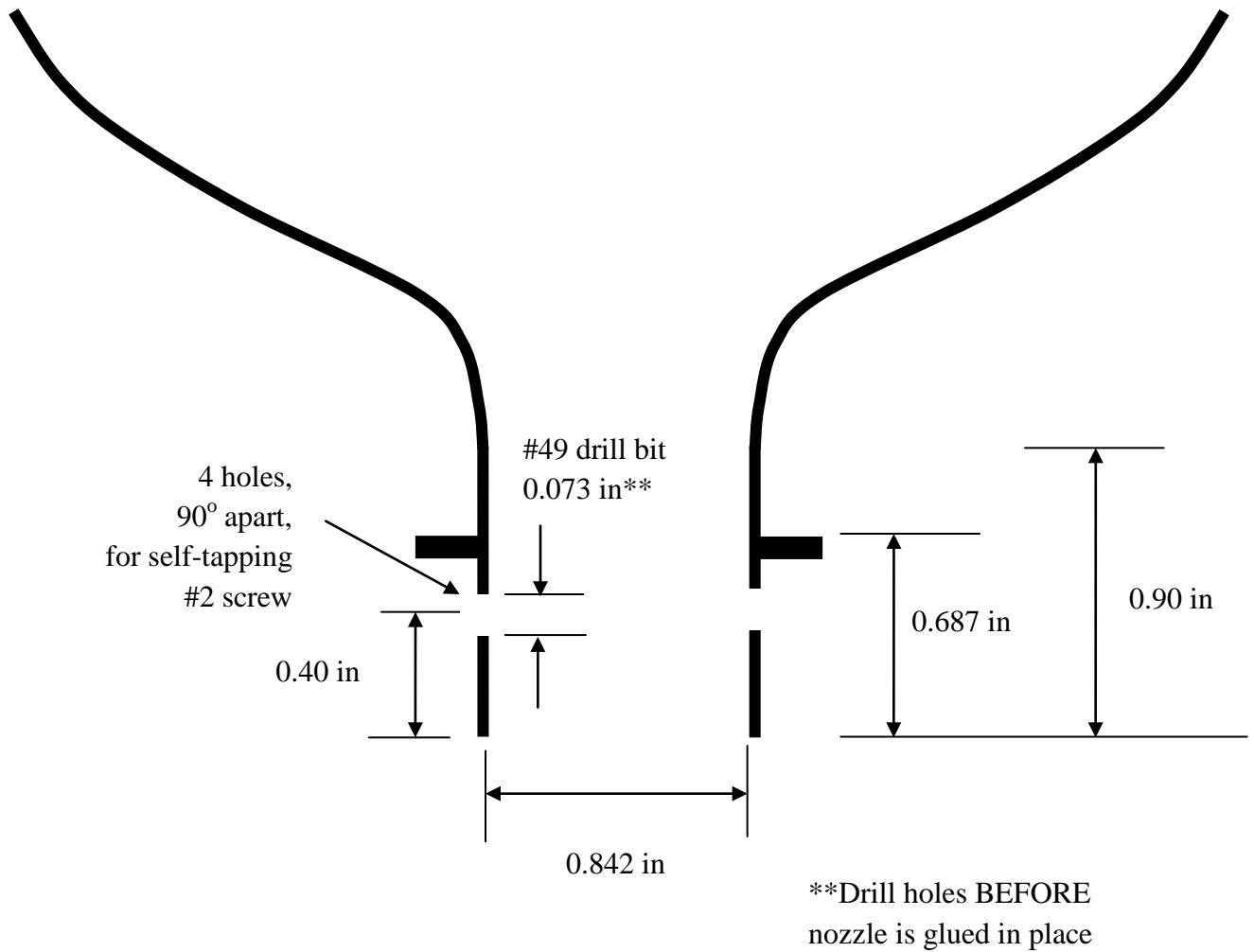


Figure 4 Detail of neck. (Note: It is critical to drill the holes in the correct location so the rocket mounts properly on the launch stand.)

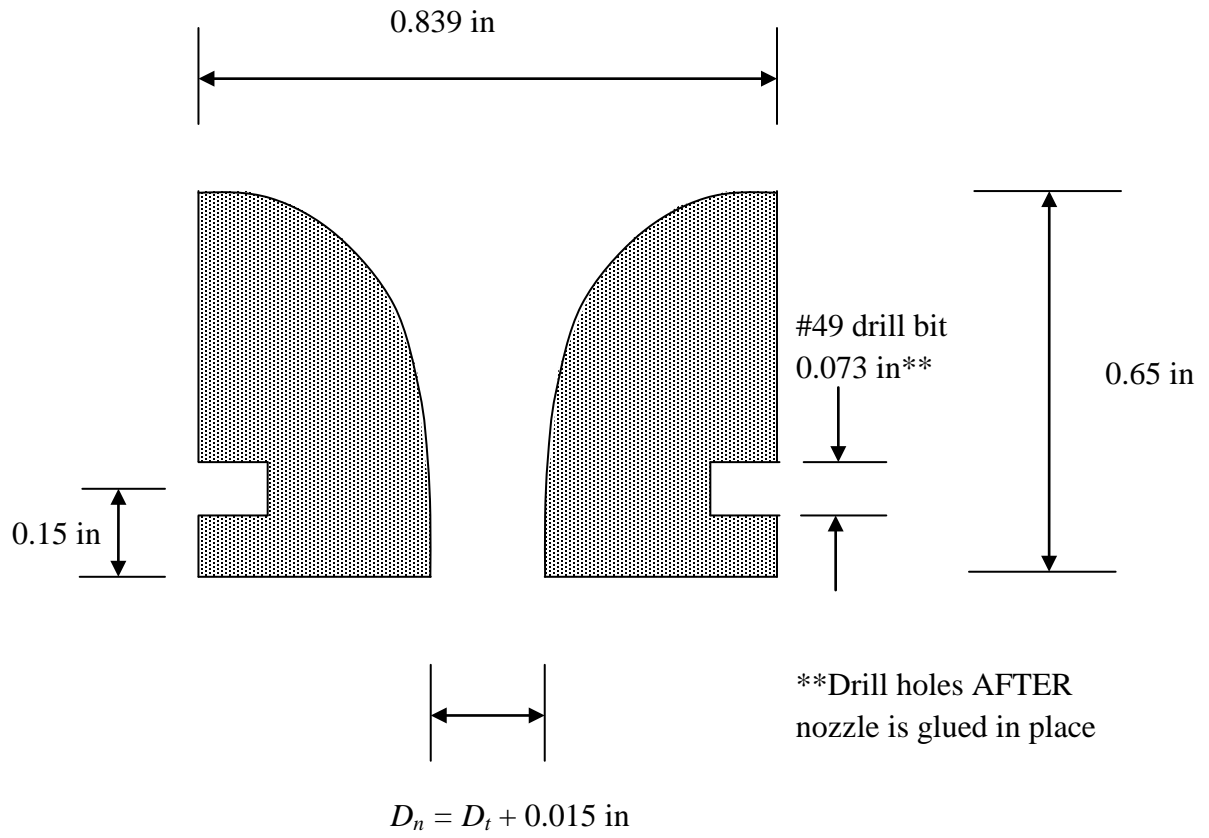


Figure 5 Detail of nozzle.

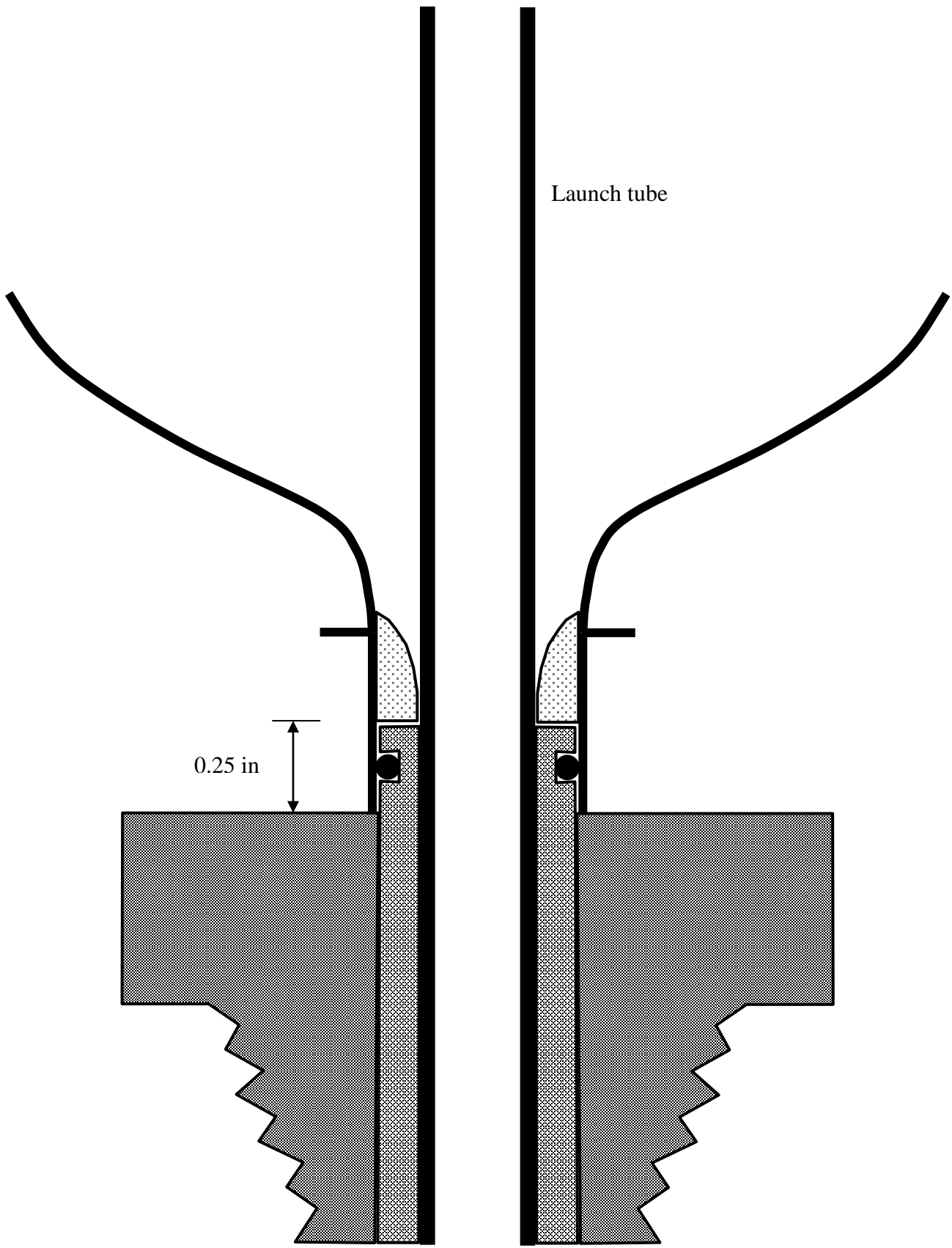


Figure 6 Bottle mounted on launch tube.

Description of Launch Apparatus

Figure 7 shows the launch apparatus, which comes in two sections: (1) base and (2) removable launch assembly (RLA). By removing the wing nuts, the RLA can be separated from the base, which is secured to the ground using stakes. The RLA contains a lanyard, which holds the bottle in place until it is ready to be launched by pulling the cord.

Figures 8 to 10 show details of the RLA, which comes in three versions.

- *Version A* has only an O-ring seal for the throat of the bottle, but it does not have a launch tube.
- *Version B* includes both the O-ring seal and a launch tube with the same diameter as the bottle throat.
- *Version C* includes both the O-ring seal and a launch tube with the same diameter as the custom nozzles made on the rapid prototype machine.

All RLA's have a detachable pipe assembly that is separated using a quick connect. The pipe assembly includes the following components:

- Pressure gauge
- Pressure relief valve to prevent overpressure
- Shrader valves that can accommodate up to 10 bicycle pumps

Figure 11 shows dimensions of the launch tube.

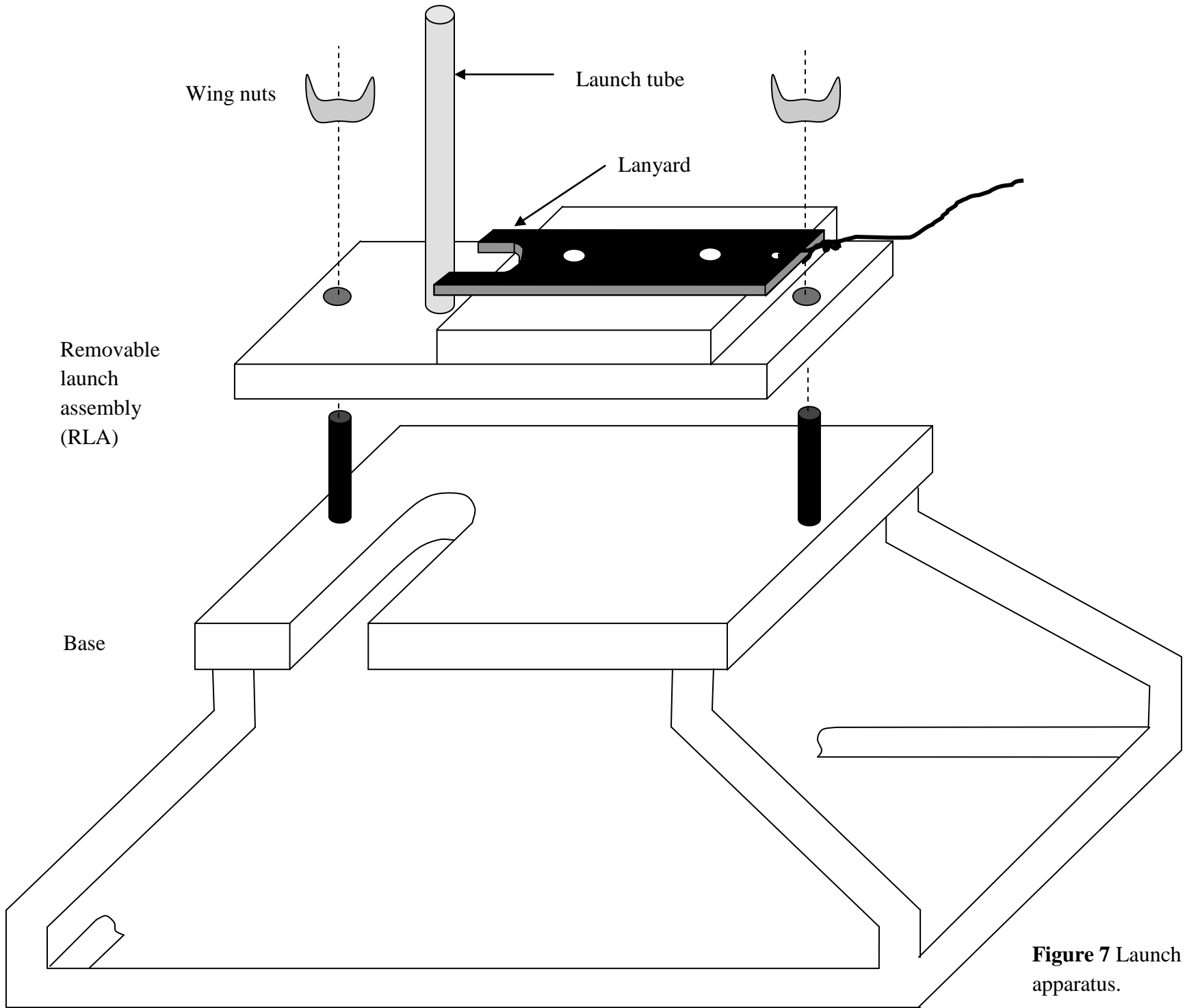


Figure 7 Launch apparatus.

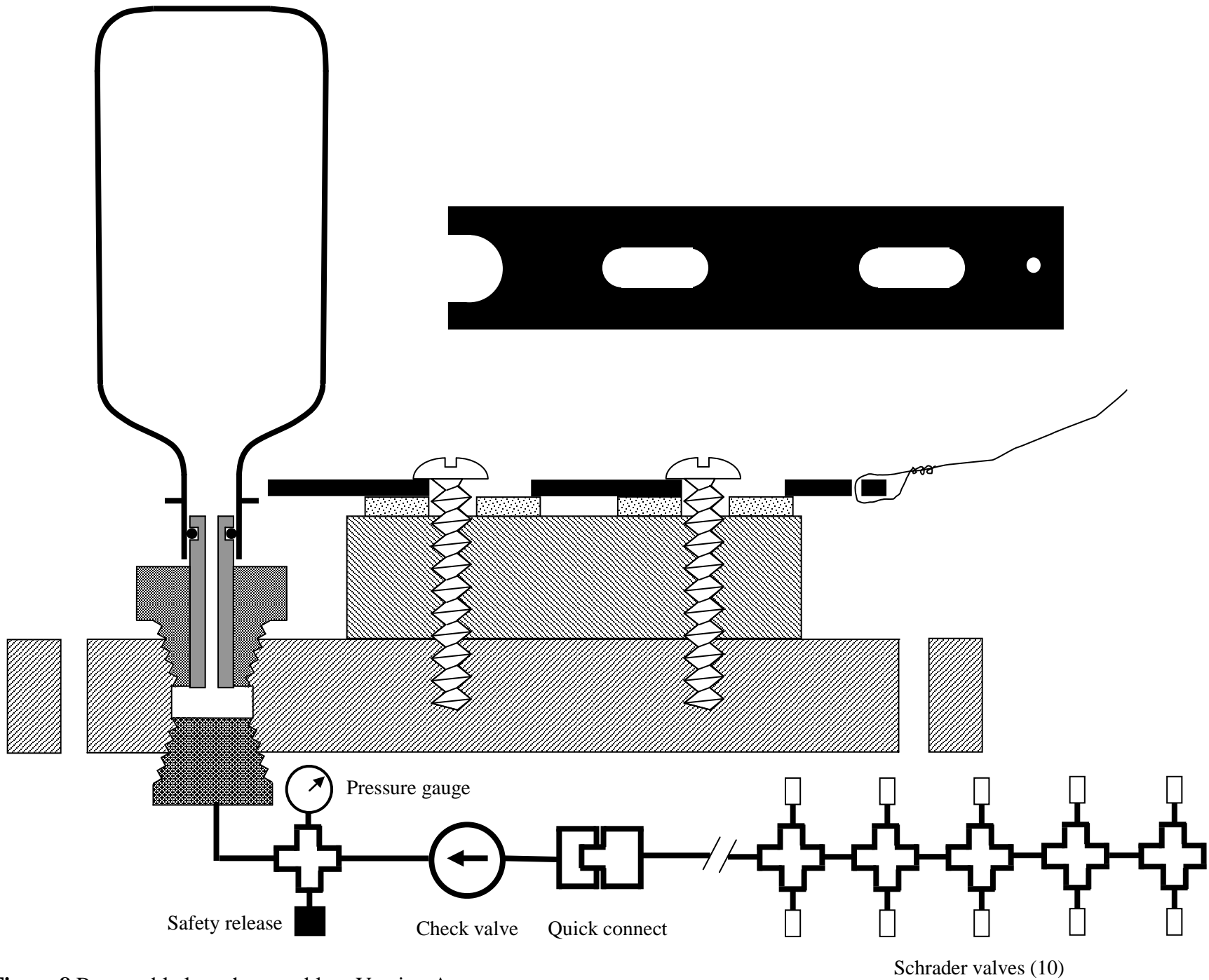


Figure 8 Removable launch assembly – Version A.

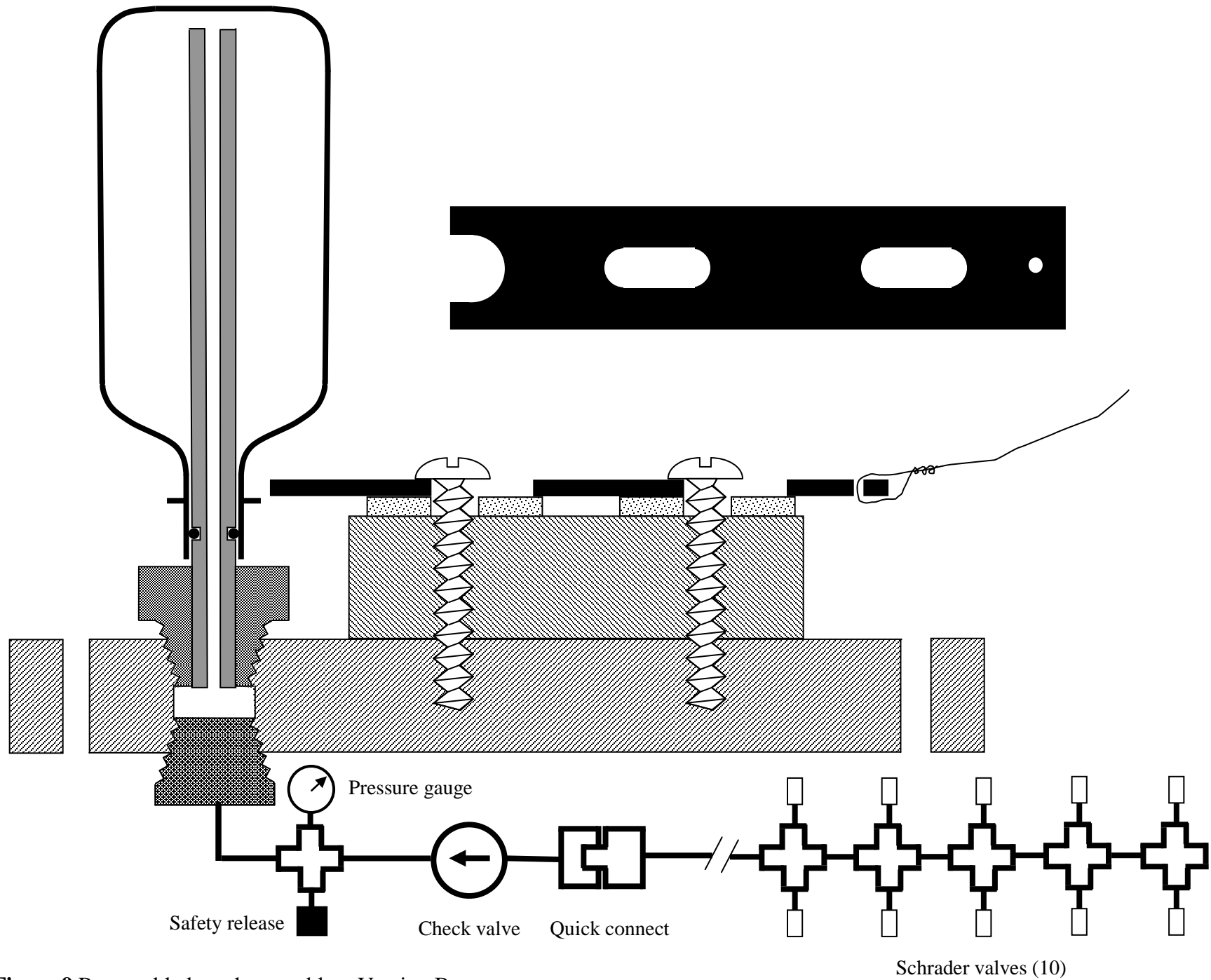


Figure 9 Removable launch assembly – Version B.

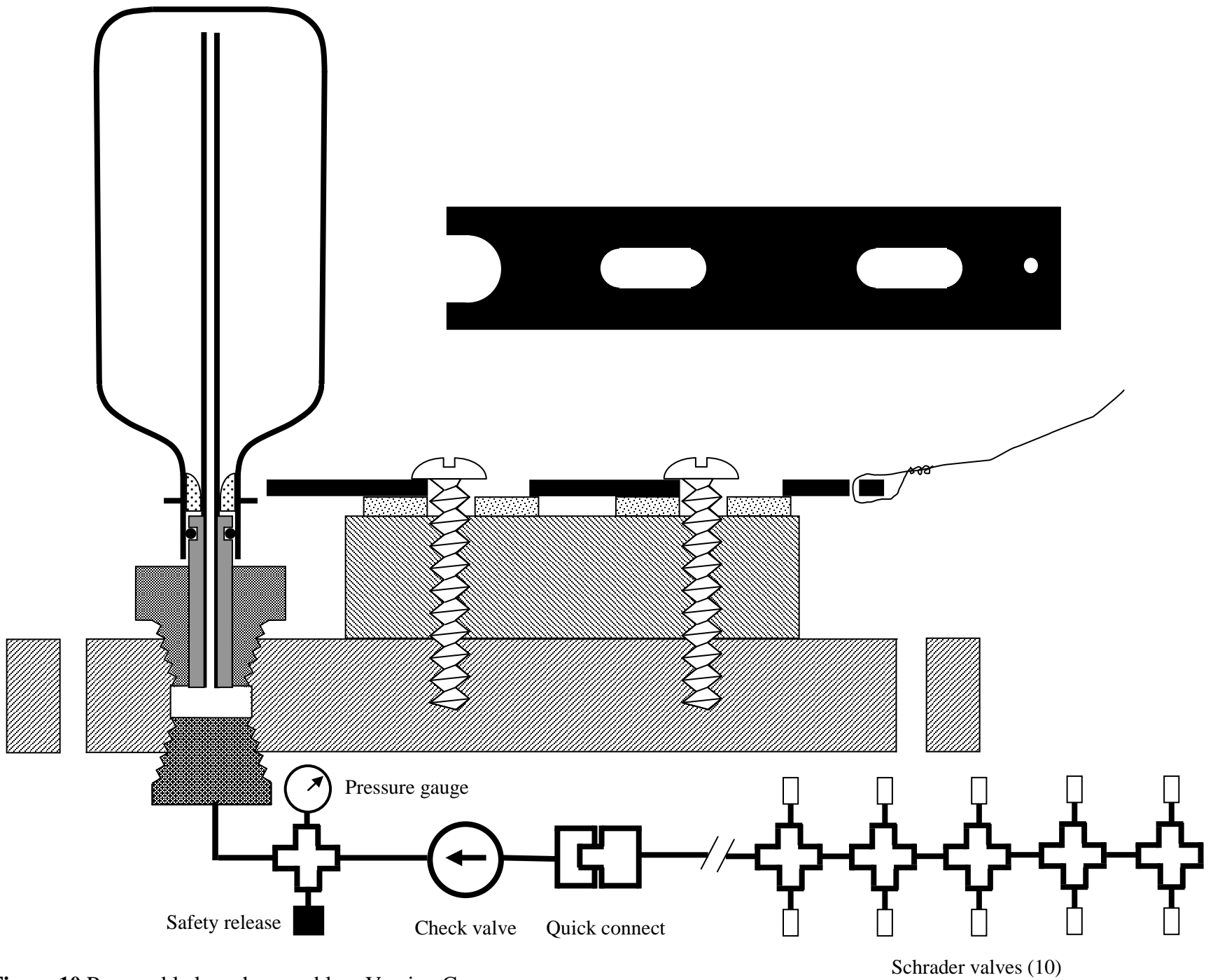


Figure 10 Removable launch assembly – Version C.

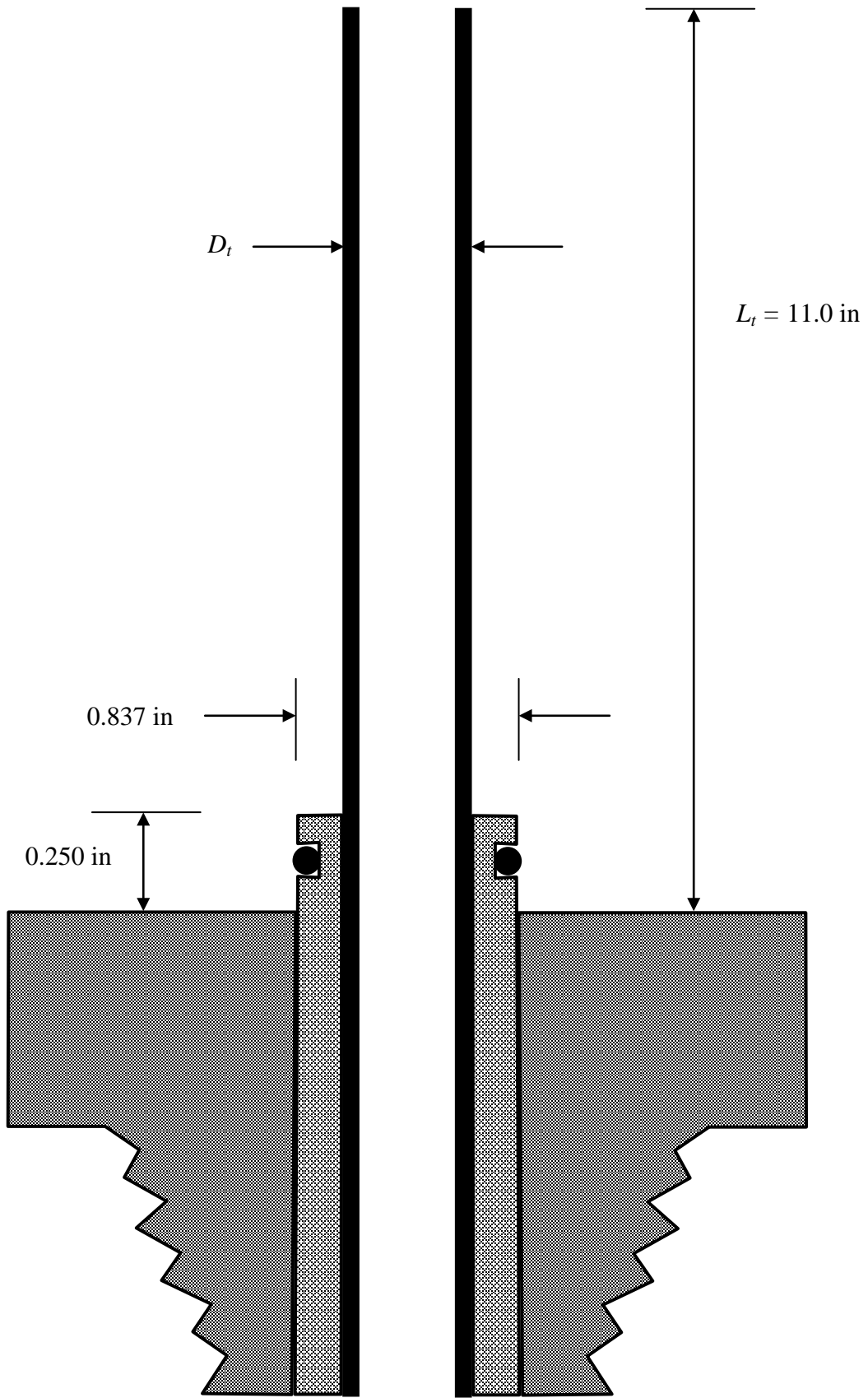


Figure 11 Launch tube dimensions.

Launch Procedures

Fill the bottle with the amount of water shown in Tables 1 and 2, where the fill fraction f is defined as follows:

$$f \equiv \frac{\text{volume of water}}{\text{empty volume of bottle}} \quad (1)$$

To ensure the water does not leak out, secure the cap on the bottle. Bring the water-filled bottle rocket to the launch site.

Setup

1. Secure base to ground using stakes
2. Attach quick connect
3. Remove wing nuts
4. Adjust bolts to level RLA
5. Install appropriate launch tube

Launch

1. Place the altimeter in the nosecone and fill the void with fiberfill. (Do not pack too tightly.) Be sure to follow altimeter directions so it provides a correct reading of the height as a function of time.
2. Remove the RLA from the base.
3. Remove the cap from the bottle. Invert the RLA and insert it into the bottle. **Ensure that the O-ring seals the neck of the bottle.**
4. Rapidly invert the bottle and place the RLA on the base.
5. Move the lanyard so it holds the neck of the bottle onto the O-ring.
6. Ensure that everyone is wearing safety goggles and is at least 10 feet from the bottle.
7. Pump the bottle to a pressure of 100 psig.
8. Shout "Fire in the hole."
9. Ensure that no one is within 10 feet of the launch pad.
10. Pull the lanyard and watch the rocket fly.
11. Recover the rocket. Remove the altimeter and transfer the height data to the computer.

Takedown

1. Remove stakes
2. Detach quick connect
3. Secure RLA to base using wing nuts
4. Remove launch tube

Safety Issues

Water rockets are widely used with young children, so the technology is very safe. NASA uses them to promote interest in science and engineering:

<http://exploration.grc.nasa.gov/education/rocket/BottleRocket/about.htm>

Maximum Pressure – Soda bottles burst at a pressure of 168 psig.

(<http://www.aircommandrockets.com/procedures.htm>)

The pressure used in our project is 100 psig, so there is a sizeable safety margin.

Safety glasses – Safety glasses will be worn during the launches.

Safe distance – No one should stand within 10 feet of the water rocket while it is being pressurized or launched.

Cuts – Scissors, knives, X-Acto knives, and box cutters are sharp, so use appropriate caution to avoid cuts.

Mathematical Model of Rocket

You have often heard the expression, “You don’t need to be a rocket scientist to understand...” Well, to do this project, you must understand something about rocket science and therefore you will become a “rocket scientist.”

The objective is to develop a mathematical model of the rocket trajectory so you can calculate the apogee, time-to-apogee, and time-to-impact as a function of independent parameters, such as the rocket mass, fill fraction, air pressure, etc.

The rocket trajectory has the following phases:

Phase 1 – If a launch tube is employed, the gas pressure will accelerate the rocket even though negligible amounts of water are expelled from the nozzle. (*Note:* If no launch tube is employed, then this phase is skipped.)

Phase 2 – Water is expelled, which provides thrust and therefore accelerates the rocket.

Phase 3 – Air is expelled, which provides thrust and therefore accelerates the rocket.

Phase 4 – The rocket has no thrust and drifts to its maximum height and then falls to the ground.

Figure 12 shows a schematic representation of the water rocket in which some of the key variables are identified.

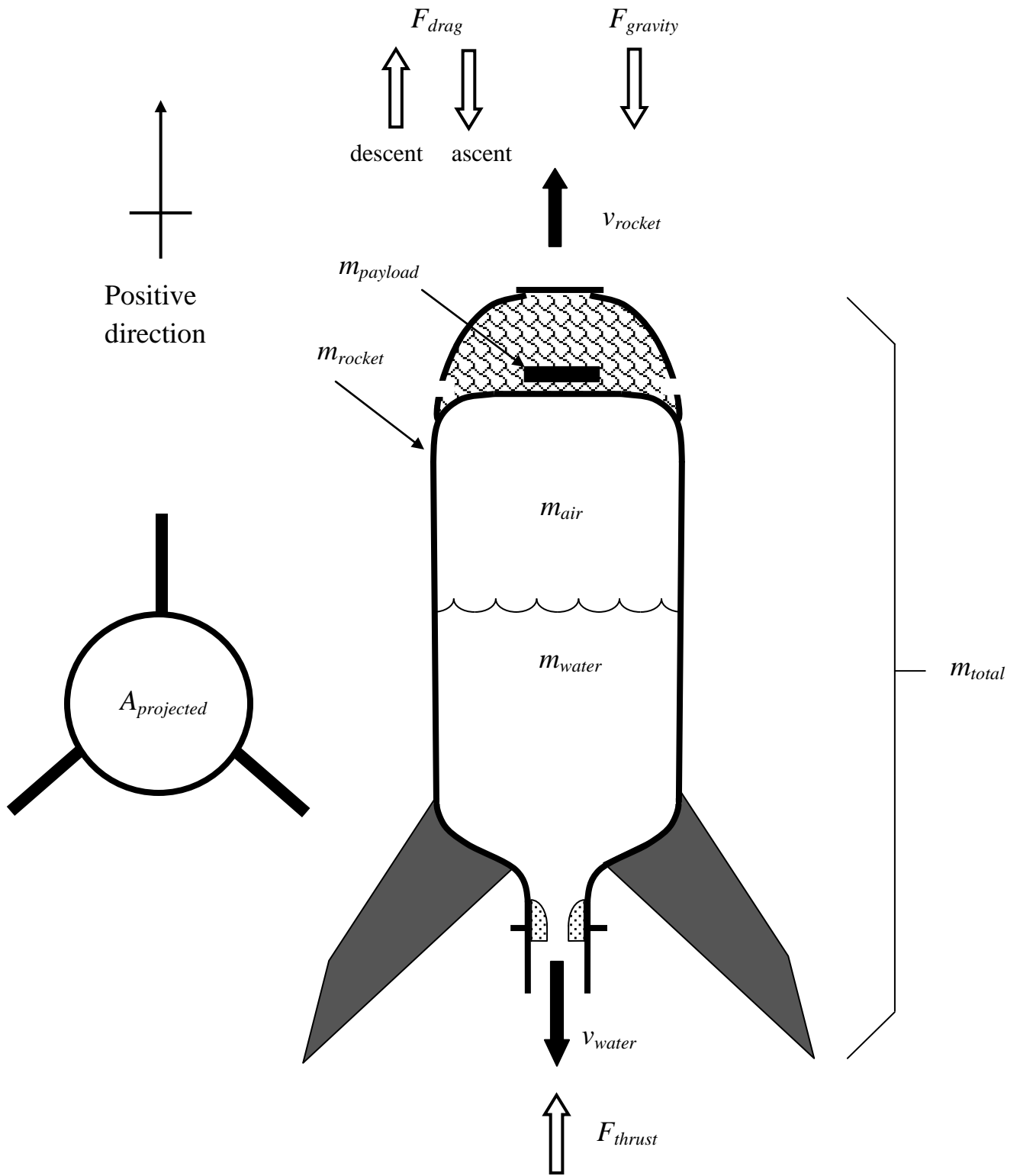


Figure 12 Some mathematical terms used to model the rocket.

Time

By definition, the time interval Δt is the elapsed time from the old time t_{i-1} to the new time t_i

$$\Delta t \equiv t_i - t_{i-1} \quad (2)$$

where the subscript i refers to “index.” Equation 2 can be rearranged as follows:

$$t_i = t_{i-1} + \Delta t \quad (3)$$

This equation states the new time t_i can be calculated from the old time t_{i-1} by adding the time interval Δt . In numerical simulations, it is common to specify a very small Δt so that it approximates a differential time interval dt , which allows differential equations to be expressed as algebraic equations that can be solved numerically. This is particularly important for complex sets of equations for which there is no analytical solution.

Position

The following equation describes the rocket velocity v_{rocket} :

$$v_{rocket} \equiv \frac{dx}{dt} \cong \frac{\Delta x}{\Delta t} = \frac{x_i - x_{i-1}}{t_i - t_{i-1}} \quad (4)$$

Equation 4 can be rearranged as follows:

$$x_i = x_{i-1} + v_{rocket} \Delta t \quad (5)$$

Equation 5 states the new position x_i can be found from the old position x_{i-1} if the velocity v_{rocket} and time interval Δt are known. An important question: What velocity is appropriate to use in this equation? Provided the time interval is small, the old velocity $v_{rocket,i-1}$ and the new velocity $v_{rocket,i}$ are virtually identical, so it is a matter of convenience which one to use.

$$x_i = x_{i-1} + v_{rocket,i-1} \Delta t \quad (6a)$$

$$x_i = x_{i-1} + v_{rocket,i} \Delta t \quad (6b)$$

Velocity

The following equation describes the rocket acceleration a_{rocket} :

$$a_{rocket} \equiv \frac{dv_{rocket}}{dt} \cong \frac{\Delta v_{rocket}}{\Delta t} = \frac{v_{rocket,i} - v_{rocket,i-1}}{t_i - t_{i-1}} \quad (7)$$

Equation 7 can be rearranged as follows:

$$v_{rocket,i} = v_{rocket,i-1} + a_{rocket} \Delta t \quad (8)$$

Equation 8 states the new velocity $v_{rocket,i}$ can be found from the old velocity $v_{rocket,i-1}$ if the acceleration a_{rocket} and time interval Δt are known. As will be described later, Newton's second law is used to determine the acceleration a_{rocket} from the net force F_{net} acting on the rocket and its instantaneous mass m_{total} .

Water Mass

The initial mass of water in the bottle $m_{water,0}$ is

$$m_{water,0} = fV_{bottle} \rho_{water} \quad (9)$$

As time progresses, the mass of water in the bottle can be determined using the Universal Accounting Equation:

$$\text{Accumulation} = \text{Input} - \text{Output} \quad (10)$$

During the launch, there is no input of water; therefore, this equation simplifies as follows

$$\text{Accumulation} = - \text{Output} \quad (11)$$

The rate that water accumulates in the bottle is

$$\frac{dm_{water}}{dt} = -\dot{m}_{water} \quad (12)$$

This differential equation can be approximated by an algebraic equation as follows:

$$\frac{dm_{water}}{dt} \cong \frac{\Delta m_{water}}{\Delta t} = \frac{m_{water,i} - m_{water,i-1}}{t_i - t_{i-1}} = -\dot{m}_{water} \quad (13)$$

Equation 13 can be rearranged as follows:

$$m_{water,i} = m_{water,i-1} - \dot{m}_{water} \Delta t \quad (14)$$

Water Flow Rate

In an ideal case, the water flow rate through the nozzle can be calculated using the Bernoulli equation, which is based on an energy balance around the nozzle in which "pressure energy" is converted to "kinetic energy." Key assumptions in the Bernoulli equation are that the fluid is not compressible (which is true for water) and that the nozzle is frictionless (which is almost true).

The Bernoulli equation follows:

$$\frac{\Delta P}{\rho} + \frac{1}{2}(\Delta v^2) + g\Delta z = 0 \quad (15)$$

The height difference across the nozzle is negligible, so the Bernoulli equation simplifies as follows:

$$\frac{\Delta P}{\rho} + \frac{1}{2}(\Delta v^2) = 0 \quad (16)$$

The pressure difference across the nozzle is the air pressure P_{air} minus the atmospheric pressure P_{atm} . Relative to the bottle frame of reference, the velocity of the water surface in the bottle is negligible, so Equation 16 simplifies to

$$\frac{P_{air} - P_{atm}}{\rho_{water}} + \frac{1}{2}(0 - v_{water}^2) = 0 \quad (17)$$

Equation 17 can be solved explicitly for the velocity of the water propellant exiting the nozzle v_{water}

$$v_{water} = \sqrt{2 \frac{P_{air} - P_{atm}}{\rho_{water}}} \quad (18)$$

When the effect of friction is included, this equation is modified as follows:

$$v_{water} = C \sqrt{2 \frac{P_{air} - P_{atm}}{\rho_{water}}} \quad (19)$$

where C is the nozzle discharge coefficient, a value less than 1.0. The value of C must be determined experimentally and depends upon factors such as surface roughness and nozzle shape. Figure 13 shows typical values of the nozzle discharge coefficient, depending upon the nozzle geometry. In the absence of experimental data, $C = 0.98$ will be employed.

Approximate coefficient of discharge, C_d for three different edge profiles

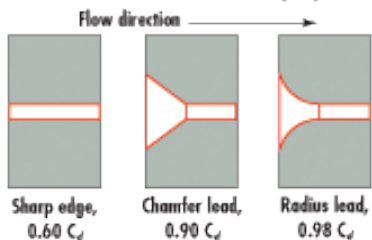


Figure 13 Nozzle discharge coefficient depends upon nozzle geometry. (<http://machinedesign.com/article/simple-orifices-have-an-edge-0318>)

At any instant of time, the rate that water mass exits the nozzle can be calculated as follows:

$$\begin{aligned}
 \dot{m}_{water} &= \rho_{water} A_{nozzle} v_{water} \\
 &= \rho_{water} A_{nozzle} C \sqrt{2 \frac{P_{air} - P_{atm}}{\rho_{water}}} \\
 &= CA_{nozzle} \sqrt{2 \rho_{water} (P_{air} - P_{atm})}
 \end{aligned} \tag{20}$$

Air volume

The initial volume of air is

$$V_{air,0} = (1 - f) V_{bottle} \tag{21}$$

At any instant of time, the air volume V_{air} is

$$V_{air} = V_{bottle} - V_{water} = V_{bottle} - \frac{m_{water}}{\rho_{water}} \tag{22}$$

Gas Temperature

As the high-pressure gas in the bottle expands, it performs work on the expelled propellant. The work comes from the thermal energy of the gas, which cools as it expands. If there is sufficient time for heat transfer, the instantaneous temperature T of the high-pressure gas warms to the ambient temperature and stays constant

$$T = T_{amb} \quad (\text{isothermal}) \tag{22a}$$

Because the rocket flight is very short, there is not sufficient time for heat transfer, so the model can be improved by assuming the rocket is perfectly insulated (*adiabatic*). In this case, the gas temperature is given by the following thermodynamic relationship, which assumes an *isentropic* (i.e., adiabatic, frictionless) expansion

$$\frac{T}{T_{init}} = \left(\frac{\hat{V}_{init}}{\hat{V}} \right)^{\gamma-1} \quad (\text{isentropic}) \tag{22b}$$

where T_{init} is the initial absolute temperature, T is the current absolute temperature, \hat{V}_{init} is the initial specific volume (V_{init} / n_{init}), and \hat{V} is the current specific volume (V / n).

$$\frac{T}{T_{init}} = \left(\frac{V_{init} / n_{init}}{V / n} \right)^{\gamma-1} \quad (22c)$$

If the number of moles is constant, Equation 22c simplifies to

$$\frac{T}{T_{init}} = \left(\frac{V_{init}}{V} \right)^{\gamma-1} \quad (\text{isentropic, constant moles})$$

$$T = T_{init} \left(\frac{V_{init}}{V} \right)^{\gamma-1} \quad (22d)$$

If the volume is constant, Equation 22c simplifies to

$$\frac{T}{T_{init}} = \left(\frac{n}{n_{init}} \right)^{\gamma-1} \quad (\text{isentropic, constant volume})$$

$$T = T_{init} \left(\frac{n}{n_{init}} \right)^{\gamma-1} \quad (22e)$$

To express the temperature in terms of mass rather than moles, the numerator and denominator can be multiplied by the molecular mass of the air M_{air}

$$T = T_{init} \left(\frac{M_{air} n}{M_{air} n_{init}} \right)^{\gamma-1} = T_{init} \left(\frac{m}{m_{init}} \right)^{\gamma-1} \quad (22f)$$

In reality, the rocket will be neither isothermal nor adiabatic; the actual temperature will fall between these extremes.

Mass of Air

The initial mass of air is

$$m_{air,0} = \rho_{air} V_{air,0} \quad (23)$$

The molar density of air can be estimated using the ideal gas equation

$$P_{air} V_{air} = n_{air} RT$$

$$\frac{n_{air}}{V_{air}} = \frac{P_{air}}{RT} \quad (24)$$

The mass density of air can be calculated using the molecular mass of air M_{air}

$$\rho_{air} = M_{air} \frac{n_{air}}{V_{air}} = \frac{M_{air} P_{air}}{RT} \quad (25)$$

Equation 25 can be substituted into Equation 23

$$m_{air,0} = \frac{M_{air} P_{air} V_{air,0}}{RT} \quad (26)$$

During Phase 3, as time progresses, the mass of air in the bottle can be determined using the Universal Accounting Equation:

$$\text{Accumulation} = \text{Input} - \text{Output} \quad (27)$$

During the launch, there is no input of air; therefore, this equation simplifies as follows

$$\text{Accumulation} = - \text{Output} \quad (28)$$

The rate that air accumulates in the bottle is

$$\frac{dm_{air}}{dt} = -\dot{m}_{air} \quad (29)$$

This differential equation can be approximated by an algebraic equation as follows:

$$\frac{dm_{air}}{dt} \cong \frac{\Delta m_{air}}{\Delta t} = \frac{m_{air,i} - m_{air,i-1}}{t_i - t_{i-1}} = -\dot{m}_{air} \quad (30)$$

Equation 30 can be rearranged as follows:

$$m_{air,i} = m_{air,i-1} - \dot{m}_{air} \Delta t \quad (31)$$

Air Flow Rate

Air is a compressible fluid, so the Bernoulli equation cannot be used to calculate the air flow rate through the nozzle. When the ratio of air pressure to atmospheric pressure is large, the flow becomes “choked,” meaning air flows through the nozzle at sonic velocity c (i.e., the speed of sound) (http://en.wikipedia.org/wiki/Choked_flow)

$$v_{air} = c = 343 \text{ m/s} \quad (32)$$

Air is choked provided the following condition is met:

$$\frac{P_{air}}{P_{atm}} > 1.9$$

This condition is met most of the time, so to keep the model simple, choked flow will be assumed at all times during Phase 3. The mass flow through a choked nozzle is given by the following equation:

$$\dot{m}_{air} = CA_{nozzle} \sqrt{\gamma \left(\frac{2}{\gamma + 1} \right)^{\frac{\gamma+1}{\gamma-1}} \rho_{air} P_{air}} \quad (33)$$

where

$$\gamma \equiv \frac{C_p}{C_v} = 1.4 \quad (\text{for air}) \quad (34)$$

Air Pressure

Using the ideal gas equation, at any instant of time, the air pressure is determined by the moles and volume of air

$$\begin{aligned} P_{air} V_{air} &= n_{air} RT \\ P_{air} &= \frac{n_{air} RT}{V_{air}} \end{aligned} \quad (35)$$

where T is the instantaneous absolute temperature. The moles of air can be calculated using the mass of air and its molar mass M_{air}

$$n_{air} = \frac{m_{air}}{M_{air}} \quad (36)$$

Substituting Equation 36 into Equation 35 gives

$$P_{air} = \frac{m_{air} RT}{M_{air} V_{air}} \quad (37)$$

Total Mass

The total rocket mass m_{total} is the sum of the empty rocket, payload, water propellant, and air propellant

$$m_{total} = m_{rocket} + m_{payload} + m_{water} + m_{air} \quad (38)$$

Thrust/Force on Rocket

According to the most proper definitions, a *force* is the resulting “push” or “pull” resulting from the interactions of two bodies, whereas a *thrust* is the resulting “push” that results when mass is ejected from a body. In the SI system, both have units of newtons (N). In common parlance, careful distinctions are not made between these two concepts. For example, see the following websites:

<http://www.grc.nasa.gov/WWW/k-12/airplane/thrust1.html>

<http://en.wikipedia.org/wiki/Thrust>

For the purposes of this project, the term F_{thrust} will be used to describe the “push” acting on the rocket from the propulsion systems during Phases 1, 2, and 3.

Phase 1 Force

Figure 14a shows a fully enclosed vessel in which internal pressure acts on the walls. Because the forces on opposite walls are perfectly balanced, the vessel does not move. In contrast, Figure 14b shows a vessel in which one of the walls is open. Because the forces are unbalanced, there is more force acting on the upper surface than the lower surface causing a net upward force.

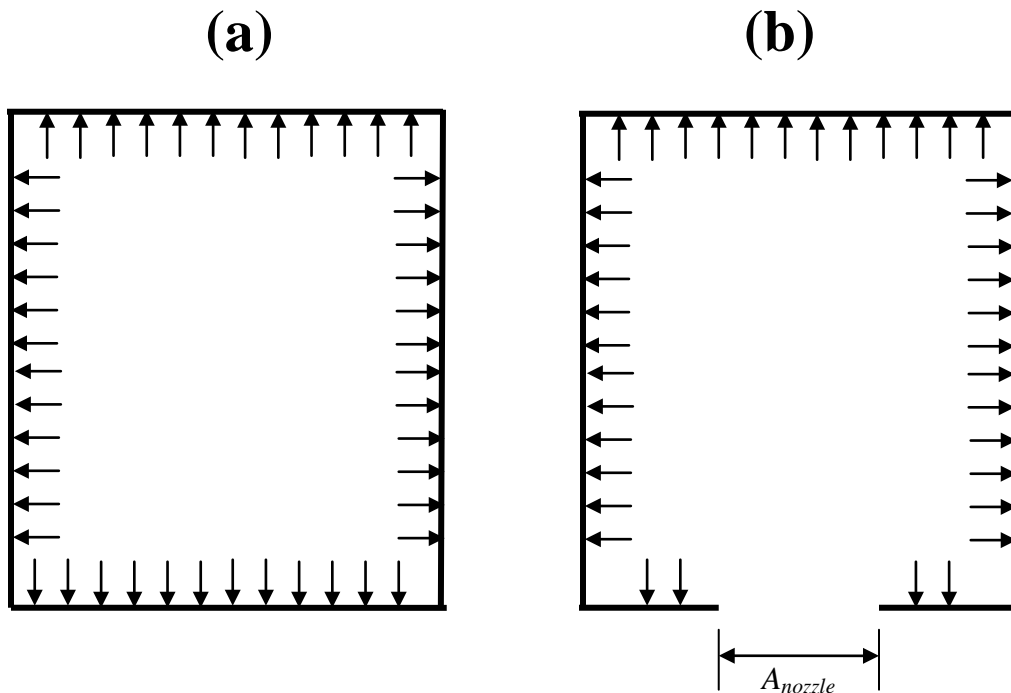


Figure 14 Pressure acting on walls of a container. (a) Fully enclosed surface. (b) Open surface. (*Note:* The arrows represent the net pressure acting on the wall, which is the internal pressure minus the external pressure. Forces from the hydrostatic head of water are neglected because they are small.)

If a launch tube is employed during Phase 1, the force acting on the rocket from unbalanced forces follow:

$$F_{thrust} = (P_{air} - P_{atm})A_{nozzle} \quad (40a)$$

Phase 2 & 3 Thrust

Figure 15 show an accounting of rocket momentum in the absence of gravity or drag. In this case, the rocket is defined as the system. The following equation accounts for the change in momentum of the rocket because mass is ejected

$$\frac{d}{dt}(m_{total}\vec{v}_{rocket}) = -\dot{m}_{out}v_{out} \quad (57)$$

$$\vec{v}_{rocket} \frac{d}{dt}(m_{total}) + m_{total} \frac{d}{dt}(\vec{v}_{rocket}) = -\dot{m}_{out}\vec{v}_{out}$$

where both \vec{v}_{out} and \vec{v}_{rocket} are velocities relative to a stationary reference frame (the earth). By accounting for the rocket mass

$$\frac{d}{dt}(m_{total}) = -\dot{m}_{out} \quad (58)$$

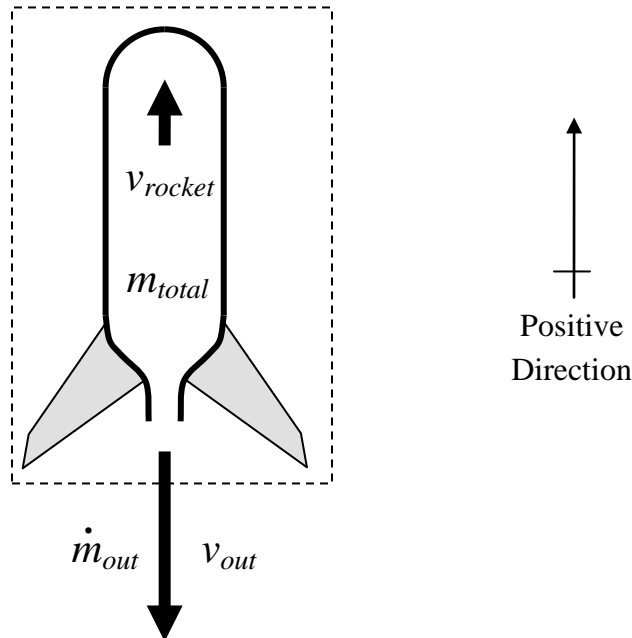


Figure 15 Accounting for momentum of the rocket.

Equation 58 can be substituted into Equation 57 to yield

$$\vec{v}_{rocket}(-\dot{m}_{out}) + m_{total} \frac{d}{dt}(\vec{v}_{rocket}) = -\dot{m}_{out} \vec{v}_{out} \quad (59)$$

By definition, the rate the velocity change is acceleration

$$\vec{v}_{rocket}(-\dot{m}_{out}) + m_{total} \vec{a}_{rocket} = -\dot{m}_{out} \vec{v}_{out} \quad (60)$$

This equation can be rearranged as follows:

$$m_{total} \vec{a}_{rocket} = \vec{v}_{rocket}(\dot{m}_{out}) - \dot{m}_{out} \vec{v}_{out} = \dot{m}_{out}(\vec{v}_{rocket} - \vec{v}_{out}) = -\dot{m}_{out}(\vec{v}_{out} - \vec{v}_{rocket}) \quad (61)$$

The term on the right describes the velocity of the propellant relative to the rocket

$$m_{total} \vec{a}_{rocket} = -\dot{m}_{out} \vec{v}_{propellant} \quad (62)$$

The term on the left describes the behavior of the rocket as if a force F_{thrust} were applied to it

$$\vec{F}_{thrust} = -\dot{m}_{out} \vec{v}_{propellant} \quad (63)$$

Eliminating the vector notation and considering the propellant velocity is in the negative direction, the following equation results:

$$F_{thrust} = \dot{m}_{out} v_{propellant} \quad (64)$$

For water propellant (Phase 2):

$$F_{thrust} = \dot{m}_{water} v_{water} \quad (40b)$$

where v_{water} is relative to the rocket.

For air propellant (Phase 3):

$$\begin{aligned} F_{thrust} &= \dot{m}_{air} v_{air} \\ &= \dot{m}_{air} c \end{aligned} \quad (41)$$

where c is the speed of sound in air, which is relative to the rocket.

Gravity

The force of gravity $F_{gravity}$ is determined from the “specific force” \hat{F} , which is the force of gravity per unit mass

$$F_{gravity} = \hat{F}m_{total} \quad (42)$$

The specific force \hat{F} is numerically identical to g which is normally interpreted as the local accelerations due to gravity (if this is confusing, please see Section 10.5.1, *Foundations of Engineering* for more explanation).

$$F_{gravity} = m_{total} g \quad (43)$$

where $g = 9.81 \text{ m/s}^2 = 9.81 \text{ N/kg}$.

Drag

The drag force F_{drag} is determined by the following equation:

$$F_{drag} = \frac{1}{2} C_d \rho_{atm} A_{projected} v_{rocket}^2 \quad (44)$$

As shown in Figure 15, the drag coefficient C_d for an object with the shape of the rocket is about 0.20. The density of the atmosphere is determined by the following equation:

$$\rho_{atm} = \frac{M_{air} P_{atm}}{RT} \quad (45)$$

The projected area $A_{projected}$ is shown in Figure 12 and the rocket velocity v_{rocket} is described by Equation 8. **Important Consideration:** The drag force always acts opposite the direction of the rocket velocity.

Net Forces Acting on Rocket

Figure 12 shows that during ascent, F_{thrust} acts in the positive direction, whereas the gravity force $F_{gravity}$ and drag force F_{drag} act in the negative direction. During ascent, the net force F_{net} acting in the positive direction is

$$F_{net} = F_{thrust} - F_{gravity} - F_{drag} \quad (\text{ascent}) \quad (39a)$$

During descent, the net force F_{net} acting in the positive direction is

$$F_{net} = F_{drag} - F_{gravity} \quad (\text{descent}) \quad (39b)$$

Acceleration

The acceleration of the rocket a_{rocket} is determined from the net force F_{net} and the total mass m_{total}

$$a_{rocket} = \frac{F_{net}}{m_{total}} \quad (46)$$

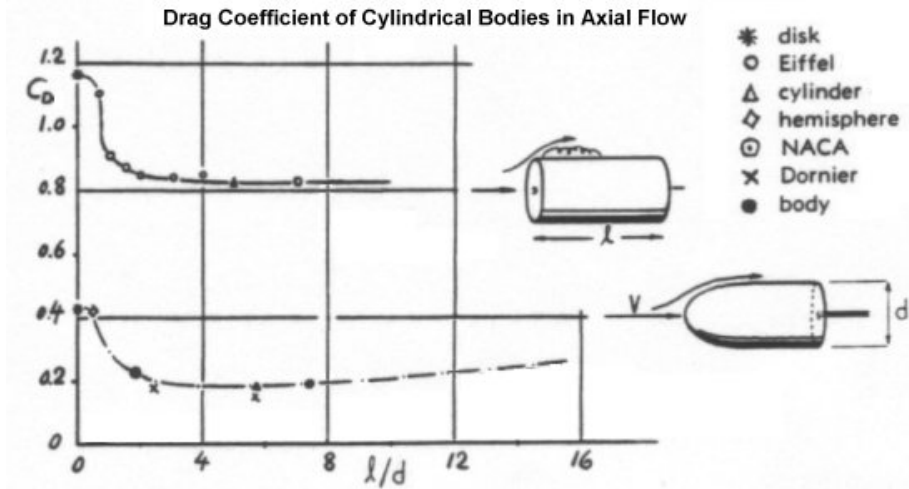


Figure 16 Drag coefficients for projectiles.

(<http://www.aerospaceweb.org/question/aerodynamics/q0231.shtml>)

Phase 1

If the launch tube is not used, at the end of Phase 1, the velocity and position of the rocket are

$$x_0 = 0 \quad (47)$$

$$v_{rocket,0} = 0 \quad (48)$$

These are the initial conditions for Phase 2.

If the launch tube is used, at the end of Phase 1, the position of the rocket is

$$x_0 = L_t \quad (49)$$

The velocity at the end of Phase 1 is determined by conducting an energy balance around the closed system depicted in Figure 17

$$\Delta E_k + \Delta E_p + \Delta U = W_{in} - W_{out} + Q_{in} - Q_{out} \quad (50)$$

There is no temperature change of the rocket, so ΔU is zero. There is no work output, so W_{out} is zero. There is no heat transfer across the system boundary, so Q_{in} and Q_{out} are zero. Therefore, the energy balance simplifies to

$$\Delta E_k + \Delta E_p = W_{in} \quad (51)$$

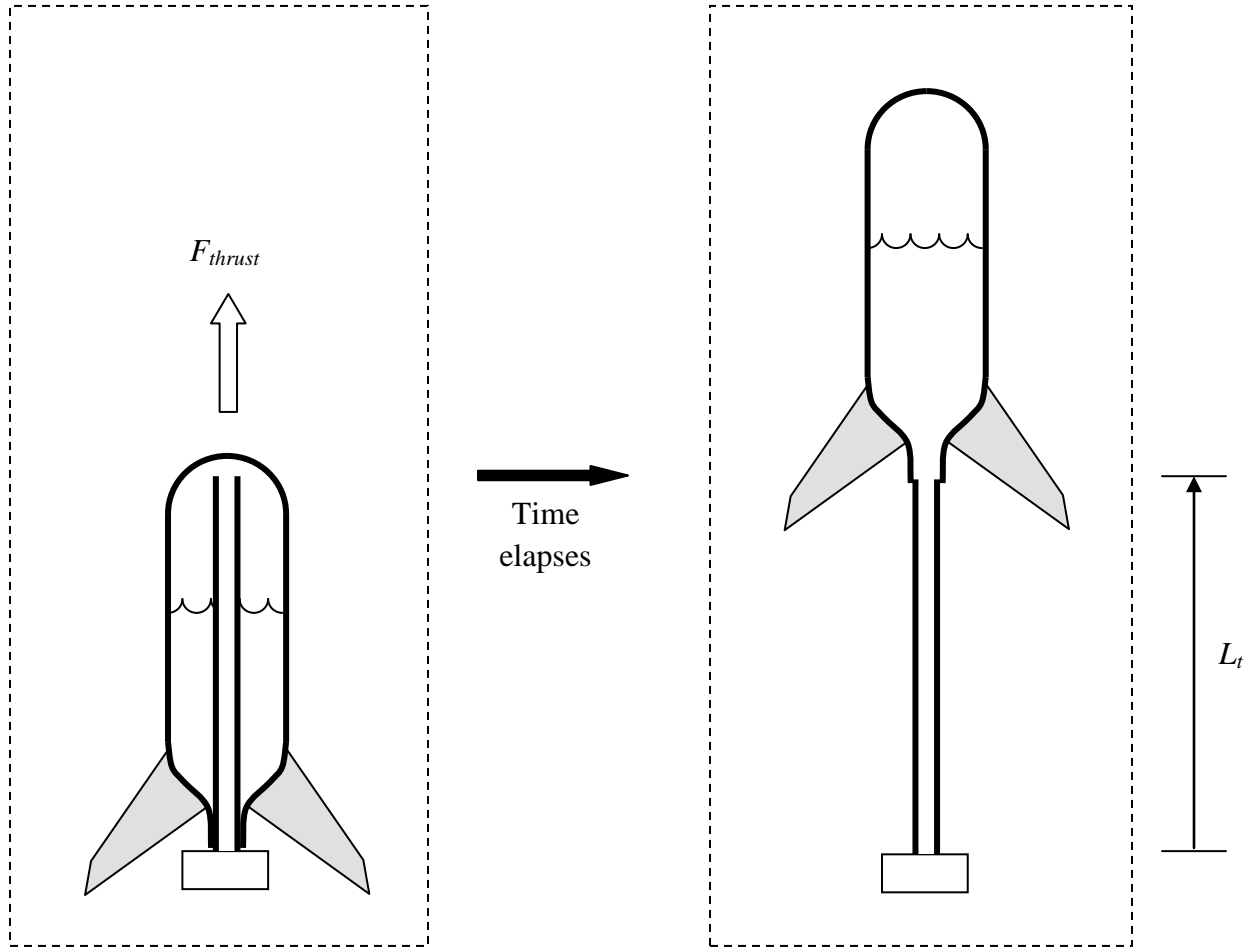


Figure 17 Rocket during Phase 1.

Each term in the energy balance may be inserted as follows:

$$\frac{1}{2} m_{total} v_{rocket}^2 + m_{total} g L_t = F_{thrust} L_t \quad (52)$$

The thrust is determined from Equation 40. Assuming there is negligible leakage, there is no loss of water, so Equation 40 simplifies to

$$F_{thrust} = (P_{air} - P_{atm}) A_{nozzle} \quad (53)$$

Equation 53 is substituted into Equation 52

$$\frac{1}{2} m_{total} v_{rocket}^2 + m_{total} g L_t = (P_{air} - P_{atm}) A_{nozzle} L_t \quad (54)$$

Equation 54 can be solved explicitly for v_{rocket}

$$v_{rocket} = \sqrt{\frac{2[(P_{air} - P_{atm}) A_{nozzle} L_t - m_{total} g L_t]}{m_{total}}} \quad (55)$$

This is the final velocity for Phase 1, which is the initial velocity of Phase 2

$$v_{rocket,0} = \sqrt{\frac{2[(P_{air} - P_{atm}) A_{nozzle} - m_{total} g] L_t}{m_{total}}} \quad (56)$$

Phases 2, 3, and 4

Because of the complexity of the equations, there is no closed-form solution in Phases 2, 3, and 4. In these phases, to determine the rocket trajectory, a computer program is required to perform numerical calculations.

Computer Program

Figure 18 shows a flowchart of the computer program.

The next sections describe details that go into key blocks.

In the previous sections, indices were used to distinguish an “old” value from a “new” value. For example,

$$t_i = t_{i-1} + \Delta t$$

In computer codes, the indices are not necessary. When a variable appears on both the left and right side of the equal sign, the variable on the left is “new” and the variable on the right is “old”; thus, in a computer program, the above equation can be written as

$$t = t + \Delta t$$

This convention will be used in the following sections, in which the equations are designed to be entered into a computer program.

Figure 19 shows the spreadsheet structure. The upper portion is where the user provides the inputs to the program. The lower portion is where the program reports the results. The user inputs information in the colored cells. If the input is a common unit, the spreadsheet calculates the SI unit, which is used in the computer program.

The time interval is set to 0.0001 s, which is short enough to approximate dt . If the results of every calculation are reported, the spreadsheet will be too large. Here, it is recommended that every 10th result be printed to the spreadsheet. This is easily accomplished by establishing a counter that cycles from 1 to 10. Printing only occurs when the counter reaches 10, and then it is reset to 1. The equations that control the printing counter are not included in the discussion below and are left to the student to program. Alternatively, printing can be controlled by the MOD operator ([http://msdn.microsoft.com/en-us/library/se0w9esz\(v=vs.80\).aspx](http://msdn.microsoft.com/en-us/library/se0w9esz(v=vs.80).aspx)).

To reduce round-off error, it is suggested to use double precision for key variables.

After completing a run of your program, report the following:

- Apogee (maximum height)
- Time-to-apogee
- Time-to-impact

The flowchart in Figure 17 does not include mechanisms to report these features. Students must use their own creativity to determine how to determine and report these numbers.

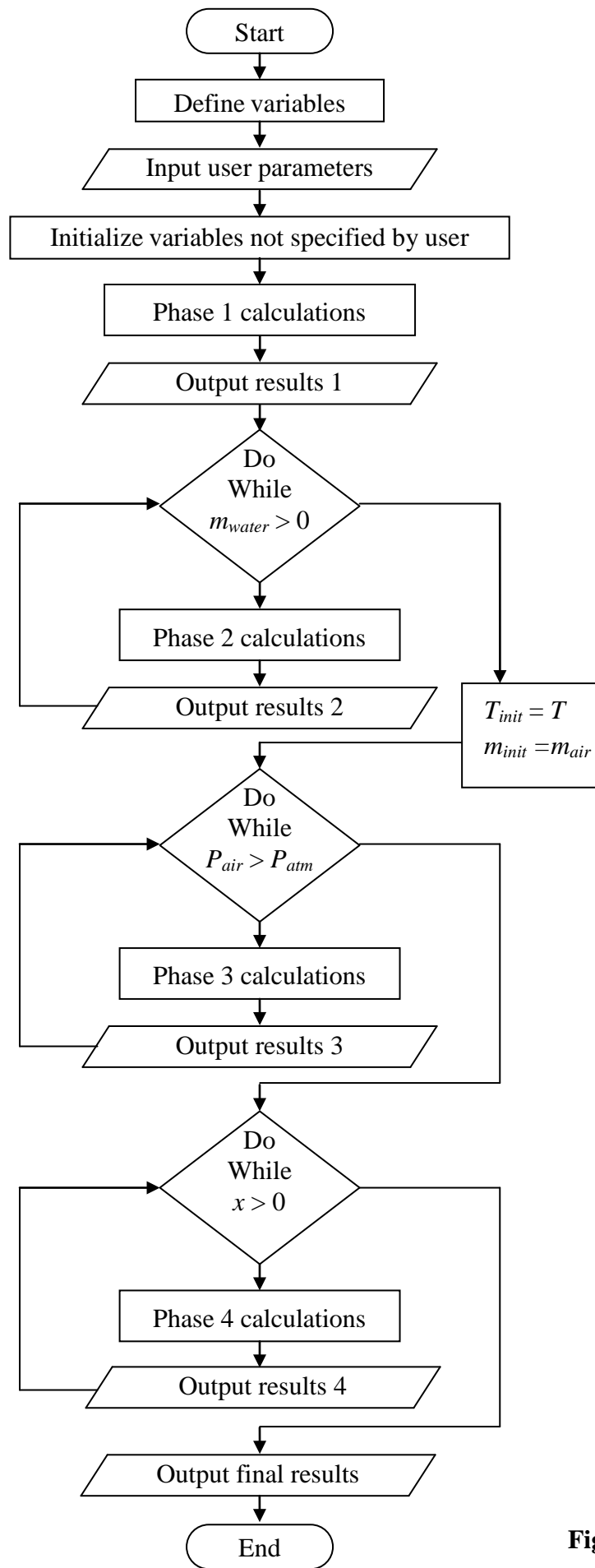


Figure 18 Flowchart.

	Common Units		SI System									
Computer program parameters												
Δt			0.0001	s								
Printing interval			10									
Rocket properties												
V_{bottle}		L		m^3								
f		–		–								
D_n		in		m								
A_{nozzle}		in^2		m^2	Calculate from nozzle diameter.							
$A_{projected}$		in^2		m^2								
$m_{payload}$		g		kg								
m_{rocket}		g		kg								
$P_{air,0}$		psig		Pa (abs)								
C_d		–		–								
C		–		–								
Launcher properties												
L_t		in		m	Zero indicates no launch tube.							
Material properties and constants												
g				m/s^2								
c				m/s								
M_{air}		g/mol		kg/mol								
γ		–		–								
P_{atm}		atm		Pa								
ρ_{water}		g/mL		kg/m^3								
T_{amb}		$^{\circ}F$		K								
R				$Pa \cdot m^3 / (mol \cdot K)$								
Key parameters during Phases 2, 3, and 4												
t	P_{air}	T	F_{thrust}	$F_{gravity}$	F_{drag}	F_{net}	m_{water}	m_{air}	m_{total}	a_{rocket}	v_{rocket}	x
(s)	(Pa)	(K)	(N)	(N)	(N)	(N)	(kg)	(kg)	(kg)	(m/s^2)	(m/s)	(m)
0												
0.001												
0.002												
...												

Figure 19 Spreadsheet structure. Users input data in yellow cells.

User Parameters

The following parameters must be specified by the user. They should be entered into cells located at the top of the spreadsheet (see Figure 19).

A_{nozzle}	= area of nozzle (m^2)
$A_{projected}$	= projected frontal area of rocket (m^2)
c	= speed of sound in air = 343 m/s
C	= nozzle discharge coefficient (dimensionless)
C_d	= drag coefficient (dimensionless)
f	= fill fraction = initial volume of water/volume of bottle
g	= local acceleration from gravity (m/s^2)
L_t	= length of launch tube (m)
$m_{payload}$	= mass of altimeter (kg)
m_{rocket}	= mass of rocket, with fiberfill, without propellant, and without payload (kg)
M_{air}	= molecular mass of air = 0.029 kg/mol
P_{air}	= initial absolute pressure of air propellant (Pa)
P_{atm}	= absolute pressure of atmosphere (Pa)
R	= ideal gas constant = $8.314 \text{ Pa}\cdot\text{m}^3/(\text{mol}\cdot\text{K})$
Δt	= time interval
T_{amb}	= ambient air temperature (K)
V_{bottle}	= empty volume of bottle (m^3)
ρ_{water}	= density of water propellant = 1000 kg/m^3
γ	= $C_p/C_v = 1.4$ for air

Initialize Variables Not Specified by User

$$\rho_{atm} = \frac{M_{air} P_{atm}}{RT_{amb}} \quad (45)$$

$$T = T_{amb}$$

$$\rho_{air} = \frac{M_{air} P_{air}}{RT} \quad (25)$$

$$V_{air} = (1 - f)V_{bottle} \quad (21)$$

$$m_{air} = \rho_{air} V_{air} \quad (23)$$

$$m_{water} = fV_{bottle} \rho_{water} \quad (9)$$

$$m_{total} = m_{rocket} + m_{payload} + m_{water} + m_{air} \quad (38)$$

$$t = 0$$

$$x = 0$$

$$v_{rocket} = 0$$

$$V_{init} = (1 - f)V_{bottle}$$

$$T_{init} = T_{amb}$$

Phase 2 Calculations

$$V_{air} = V_{bottle} - \frac{m_{water}}{\rho_{water}} \quad (22)$$

$$T = T_{init} \left(\frac{V_{init}}{V_{air}} \right)^{\gamma-1} \quad (22d)$$

$$P_{air} = \frac{m_{air} RT}{M_{air} V_{air}} \quad (37)$$

$$\dot{m}_{water} = CA_{nozzle} \sqrt{2\rho_{water} (P_{air} - P_{atm})} \quad (20)$$

$$m_{water} = m_{water} - \dot{m}_{water} \Delta t \quad (14)$$

$$m_{total} = m_{rocket} + m_{payload} + m_{water} + m_{air} \quad (38)$$

$$v_{water} = C \sqrt{2 \frac{P_{air} - P_{atm}}{\rho_{water}}} \quad (19)$$

$$F_{thrust} = \dot{m}_{water} v_{water} \quad (40b)$$

$$F_{gravity} = m_{total} g \quad (43)$$

$$F_{drag} = \frac{1}{2} C_d \rho_{atm} A_{projected} v_{rocket}^2 \quad (44)$$

$$F_{net} = F_{thrust} - F_{gravity} - F_{drag} \quad (39a)$$

$$a_{rocket} = \frac{F_{net}}{m_{total}} \quad (46)$$

$$v_{rocket} = v_{rocket} + a_{rocket} \Delta t \quad (8)$$

$$x = x + v_{rocket} \Delta t \quad (5)$$

$$t = t + \Delta t \quad (3)$$

Output Results 2

Print these results for $t = 0.001$ s, 0.002 s, etc.

t	P_{air}	T	F_{thrust}	$F_{gravity}$	F_{drag}	F_{net}	m_{water}	m_{air}	m_{total}	a_{rocket}	v_{rocket}	x
(s)	(Pa)	(K)	(N)	(N)	(N)	(N)	(kg)	(kg)	(kg)	(m/s ²)	(m/s)	(m)

Prepare for the next phase by doing the following initializations:

$$T_{init} = T$$

$$m_{init} = m_{air}$$

Phase 3 Calculations

$$m_{\text{water}} = 0$$

$$V_{\text{air}} = V_{\text{bottle}} - \frac{m_{\text{water}}}{\rho_{\text{water}}} \quad (22)$$

$$T = T_{\text{init}} \left(\frac{m_{\text{air}}}{m_{\text{init}}} \right)^{\gamma-1} \quad (22f)$$

$$P_{\text{air}} = \frac{m_{\text{air}} RT}{M_{\text{air}} V_{\text{air}}} \quad (37)$$

$$\rho_{\text{air}} = \frac{M_{\text{air}} P_{\text{air}}}{RT} \quad (25)$$

$$\dot{m}_{\text{air}} = CA_{\text{nozzle}} \sqrt{\gamma \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma+1}{\gamma-1}} \rho_{\text{air}} P_{\text{air}}} \quad (33)$$

$$m_{\text{air}} = m_{\text{air}} - \dot{m}_{\text{air}} \Delta t \quad (31)$$

$$m_{\text{total}} = m_{\text{rocket}} + m_{\text{payload}} + m_{\text{water}} + m_{\text{air}} \quad (38)$$

$$F_{\text{thrust}} = \dot{m}_{\text{air}} c \quad (41)$$

$$F_{\text{gravity}} = m_{\text{total}} g \quad (43)$$

$$F_{\text{drag}} = \frac{1}{2} C_d \rho_{\text{atm}} A_{\text{projected}} v_{\text{rocket}}^2 \quad (44)$$

$$F_{\text{net}} = F_{\text{thrust}} - F_{\text{gravity}} - F_{\text{drag}} \quad (39a)$$

$$a_{\text{rocket}} = \frac{F_{\text{net}}}{m_{\text{total}}} \quad (46)$$

$$v_{\text{rocket}} = v_{\text{rocket}} + a_{\text{rocket}} \Delta t \quad (8)$$

$$x = x + v_{\text{rocket}} \Delta t \quad (5)$$

$$t = t + \Delta t \quad (3)$$

Phase 4 Calculations

$$m_{\text{water}} = 0$$

$$V_{\text{air}} = V_{\text{bottle}} - \frac{m_{\text{water}}}{\rho_{\text{water}}} \quad (22)$$

$$P_{\text{air}} = P_{\text{atm}} \quad (37)$$

$$T = T_{\text{init}} \left(\frac{m_{\text{air}}}{m_{\text{init}}} \right)^{\gamma-1} \quad (22f)$$

$$\rho_{\text{air}} = \frac{M_{\text{air}} P_{\text{air}}}{RT} \quad (25)$$

$$m_{\text{air}} = \rho_{\text{air}} V_{\text{air}} \quad (23)$$

$$m_{\text{total}} = m_{\text{rocket}} + m_{\text{payload}} + m_{\text{water}} + m_{\text{air}} \quad (38)$$

$$F_{\text{thrust}} = 0$$

$$F_{\text{gravity}} = m_{\text{total}} g \quad (43)$$

$$F_{\text{drag}} = \frac{1}{2} C_d \rho_{\text{atm}} A_{\text{projected}} v_{\text{rocket}}^2 \quad (44)$$

$$F_{\text{net}} = F_{\text{thrust}} - F_{\text{gravity}} - F_{\text{drag}} \quad (\text{ascent}) \quad (39a)$$

$$F_{\text{net}} = F_{\text{drag}} - F_{\text{gravity}} \quad (\text{descent}) \quad (39b)$$

$$a_{\text{rocket}} = \frac{F_{\text{net}}}{m_{\text{total}}} \quad (46)$$

$$v_{\text{rocket}} = v_{\text{rocket}} + a_{\text{rocket}} \Delta t \quad (8)$$

$$x = x + v_{\text{rocket}} \Delta t \quad (5)$$

$$t = t + \Delta t \quad (3)$$

Nomenclature

a_{rocket}	= acceleration of rocket (m/s^2)
A_{nozzle}	= area of nozzle (m^2)
$A_{projected}$	= projected frontal area of rocket (m^2)
c	= speed of sound in air = 343 m/s
C	= nozzle discharge coefficient (dimensionless)
C_d	= drag coefficient (dimensionless)
C_p	= heat capacity at constant pressure ($J/(mol \cdot K)$)
C_v	= heat capacity at constant volume ($J/(mol \cdot K)$)
D_n	= diameter of nozzle (m)
D_t	= diameter of launch tube (m)
ΔE_k	= change in kinetic energy of the system (J)
ΔE_p	= change in potential energy of the system (J)
f	= fill fraction = initial volume of water/volume of bottle
F_{drag}	= force from air drag (N)
$F_{gravity}$	= force from gravity (N)
F_{net}	= net force acting on rocket (N)
F_{thrust}	= force from propellant (N)
$\hat{F}_{gravity}$	= specific force from gravity (N/kg)
g	= local acceleration from gravity (m/s^2)
L_t	= length of launch tube (m)
m_{air}	= mass of air propellant (kg)
m_{init}	= initial mass of air prior to entering a phase (kg)
$m_{payload}$	= mass of altimeter (kg)
m_{rocket}	= mass of rocket, with fiberfill, without propellant, and without payload (kg)

m_{total}	= total mass of rocket, payload, water propellant, and air propellant (kg)
m_{water}	= mass of water propellant (kg)
\dot{m}_{air}	= rate that air propellant is expelled (kg/s)
\dot{m}_{out}	= rate that propellant is expelled (kg/s)
\dot{m}_{water}	= rate that water propellant is expelled (kg/s)
M_{air}	= molecular mass of air = 0.029 kg/mol
n_{air}	= moles of air (mol)
n_{init}	= initial moles of air prior to a phase (mol)
P_{air}	= absolute pressure of air propellant (Pa)
P_{atm}	= absolute pressure of atmosphere (Pa)
P_{gauge}	= gauge pressure of air propellant (Pa)
Q_{in}	= heat input (J)
W_{out}	= heat output (J)
R	= ideal gas constant = 8.314 Pa·m ³ /(mol·K)
t	= time since rocket left launch pad (s)
Δt	= time interval (s)
T	= instantaneous temperature of high-pressure gas (K)
T_{amb}	= ambient air temperature (K)
T_{init}	= initial temperature of high-pressure gas (K)
ΔU	= change in internal energy of the system (J)
v_{air}	= velocity of air propellant relative to rocket (m/s)
v_{out}	= velocity of propellant relative to a stationary reference frame (m/s)
$v_{propellant}$	= velocity of propellant relative to rocket (m/s)
v_{rocket}	= velocity of rocket relative to earth (m/s)

v_{water} = velocity of water propellant relative to rocket (m/s)

V_{air} = volume of air propellant (m³)

V_{bottle} = empty volume of bottle (m³)

V_{init} = initial volume of air prior to entering a phase (m³)

V_{water} = volume of water propellant (m³)

\hat{V} = molar volume at an instant of time (m³/mol)

\hat{V}_{init} = initial molar volume (m³/mol)

W_{in} = work input (J)

W_{out} = work output (J)

x = vertical distance (m)

ρ_{air} = density of air propellant (kg/m³)

ρ_{atm} = density of atmospheric air (kg/m³)

ρ_{water} = density of water propellant = 1000 kg/m³

γ = $C_p/C_v = 1.4$ for air

Subscripts

0 = initial (Important: This is the number zero, not the letter O.)

i = index

Report

The report must contain the following sections in the following order. Each section must be clearly labeled. DO NOT bind the report. Simply staple it in the upper left-hand corner. Use the coversheet from the back of this document.

1. Brief description of the project and problem statement

Briefly describe the rocket and the experimental approach.

2. Description of the as-built rocket with appropriate drawings

In detail, describe the rocket. Number all drawings and use a caption. (Follow the same style used in this document.) Refer to all figures in the text.

3. Operating procedures for the rocket

Describe steps taken to prepare the rocket and to perform the launch.

4. Description of safety measures taken

Describe safety measures taken during fabrication, preparation, and launch.

5. Expenses

In a table, list the cost of the expendables used in the project.

6. Description of results from the computer simulation

For the following condition:

- $f = 0.4$
- $D_n = 0.515$ in
- Launch tube = yes

Prepare the following plots:

P_{air} vs. t

T vs. t

F_{thrust} vs. t

$F_{gravity}$ vs. t

F_{drag} vs. t

F_{net} vs. t

m_{water} VS. t

m_{air} VS. t

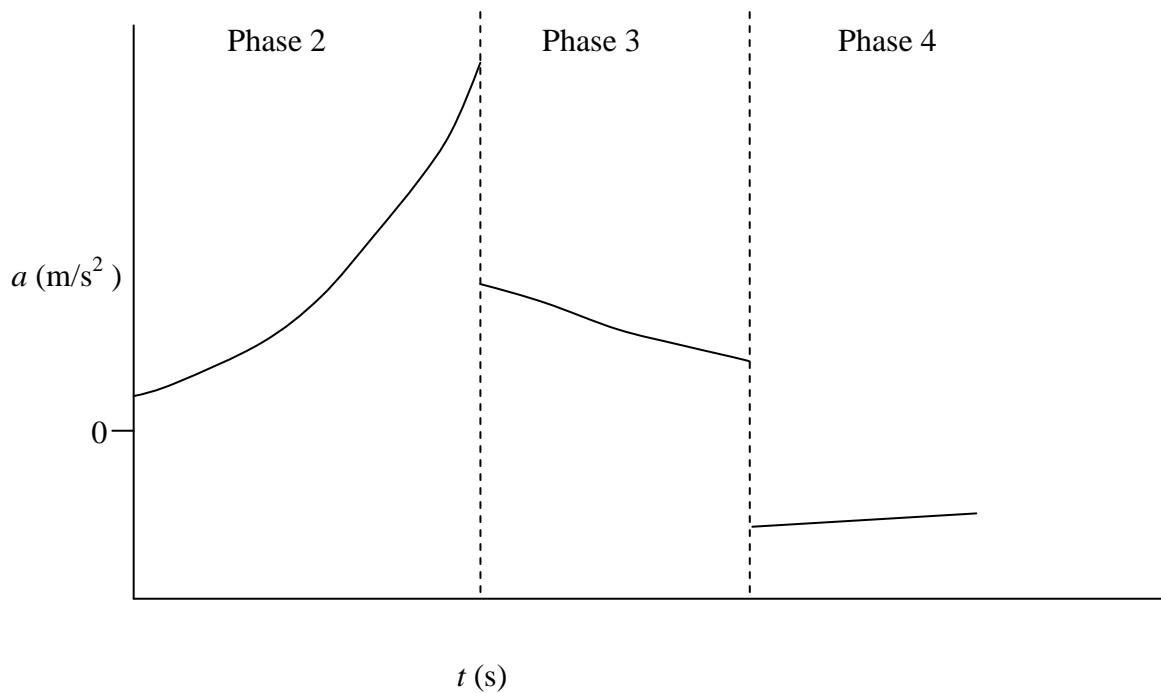
m_{total} VS. t

a_{rocket} VS. t

v_{rocket} VS. t

x VS. t

In each figure, indicate the section where Phase 2, Phase 3, and Phase 4 occurs, as shown in the example below:



Each figure must have its own figure number and descriptive caption. Each figure must be referenced and described in the text. Be sure to discuss the relationship between the figures during each phase of the trajectory. For example, in Phase 2 as water leaves the bottle, describe what happens to the mass, gas pressure, acceleration, etc.

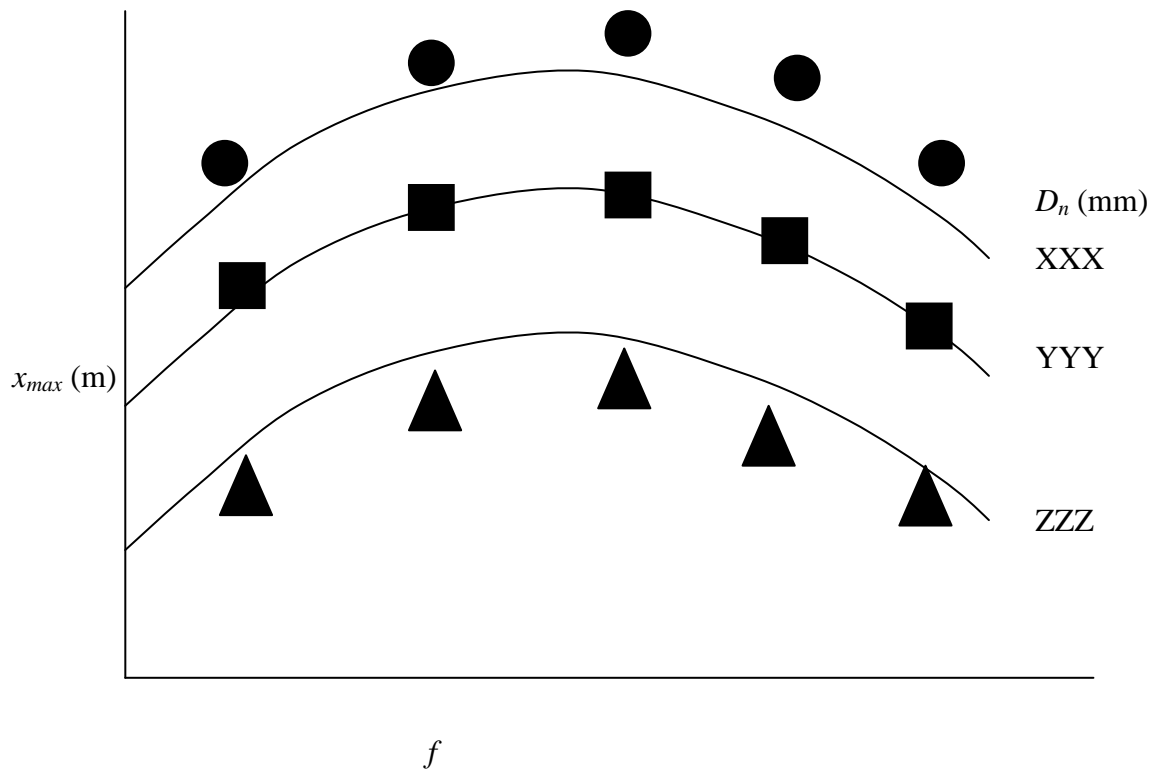
For each plot, the time period should start from the launch and end when the rocket hits the ground.

7. Comparison of computer simulation to experimental results

As a function of various independent parameters (nozzle diameter, fill fraction, launch tube), your class will generate the following data:

- Apogee (maximum height)
- Time-to-apogee
- Time-to-impact

In a graph, compare the data with the computer simulation. Shown below is an example graph for apogee:



The smooth line represents the simulation and the data points are the experimental data generated by the class. Be sure to appropriately label each curve. Use a different symbol for each nozzle diameter. In a legend, identify what each symbol means. Create one figure for the experiments performed with the launch tube and another figure for the experiments performed without the launch tube.

In the text, discuss each figure. Explicitly refer to each figure by number. Address the following issues:

- How well do the data agree with the simulation
- To obtain the highest apogee, what is the best combination of nozzle diameter and fill fraction

8. Assumptions and experimental errors

The experimental data and computer simulation will not agree exactly. In each phase of the trajectory, describe the assumption made in the computer simulation. Describe the validity of each assumption. Also, describe experimental errors that could cause disagreements between the model and data.

In particular, discuss the assumption that the rocket is adiabatic. Do you think the temperatures reported in the simulation are reasonable if the rocket is adiabatic? If you think that the simulation will yield better results assuming the rocket is isothermal, re-run the program with the isothermal assumption. Determine which assumption is a better fit to the experimental data.

9. Reflections

In the execution of this project – including modeling, rocket construction, experiment, and reporting – what skills and knowledge were applied from ENGR 111 and ENGR 112?

Appendix A – Computer Program

Use good programming technique, as follows:

- Use Option Explicit to detect spelling errors
- Define all variables and the kind of number that will be stored (e.g., integer, double precision)
- Use comments to describe what you are doing, but do not overdo it.
- Use appropriate indentation

Appendix B – Spreadsheet Printout

Follow the format in Figure 19.

Water Rocket Project

Section Number _____ (typed)

Team Number _____ (typed)

Student Name (typed)	Percent of Effort*	Student Signature

* Full effort = 100%

Item	Points Earned	Maximum Points	Description
1		5	Brief description of the project and problem statement
2		10	Description of the as-built rocket with appropriate drawings
3		5	Operating procedures for the rocket
4		3	Description of safety measures taken
5		2	Expenses
6		15	Description of results from the computer simulation
7		15	Comparison of computer simulation to experimental results
8		15	Assumptions and experimental errors
9		10	Reflections
A		15	Appendix A – Computer Program
B		5	Appendix B – Spreadsheet Printout
			Bonus for exceptional report appearance and professionalism
			Penalty for poor report appearance and lack of professionalism
			Total

Appendix A: Altimeter Operating Procedure

Hardware:

There are two pieces of hardware: (1) altimeter and (2) USB computer interface.

Software:

The altimeter instruction manual and the MicroPeak application can be found here:

<http://altusmetrum.org/MicroPeak/>

After installing the software, to automatically download the latest USB driver, it is necessary to connect the MicroPeak USB interface to the computer once. After this, it is not necessary to have an Internet connection to log flight data. (**WARNING: When you are out in a field with no Internet access, if you wait to connect the USB interface to the computer for the very first time, the software will be unable to recognize the USB interface!**)

Flight Data:

Recording flight data is simple. First, to power on, slide the altimeter switch so that it covers the “1” printed on the board (Figure A-1).



Figure A-1 Altimeter power switch. (*Note:* This device is off.)

After powering on, there is a 1-min waiting period before the altimeter checks for launch. This provides time to prepare the rocket for launch. If the rocket is launched within this 1-min window, the altimeter will not record any data.

To obtain an accurate reading, the altimeter must be placed close to the “breathing holes” in the rocket nose (Figure A-2). Ideally, the breathing holes are large (>0.25 inches).

After the rocket is launched and has landed, the altimeter automatically powers “off” to conserve battery power; however, the switch still covers the printed “1.”

To log flight data, open the MicroPeak application and connect the USB interface to the computer. Then, click the “download” button (Figure A-3), and select the MicroPeak USB interface (Figure A-4).



Figure A-2 Proper altimeter placement.

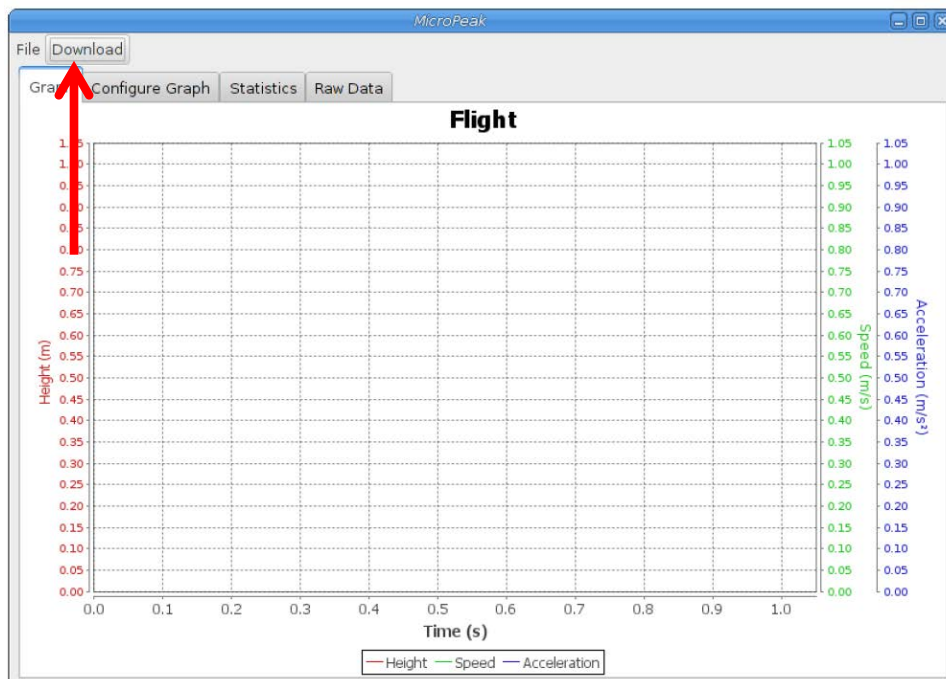


Figure A-3 MicroPeak application.



Figure A-4 Device selection.

The MicroPeak USB adapter has a small phototransistor under the hole in the center of the box. When you are ready to transfer data to the USB interface, turn the device back on by moving the altimeter switch to “0” then back to “1.” Then, quickly place the altimeter onto the USB interface (Figure A-5). The altimeter LED fits into the hole on the USB interface. (*Cautionary Note:* To prevent sunlight from interfering with data transmission, it is prudent to cover the device with your hand to block extraneous light.)

Upon successful data transmission, the MicroPeak application will automatically prompt you to save the data. Once the flight data has been saved, the altimeter should be turned off and given to the next team. The maximum height recorded during flight is displayed in the statistics tab (Figure A-6).

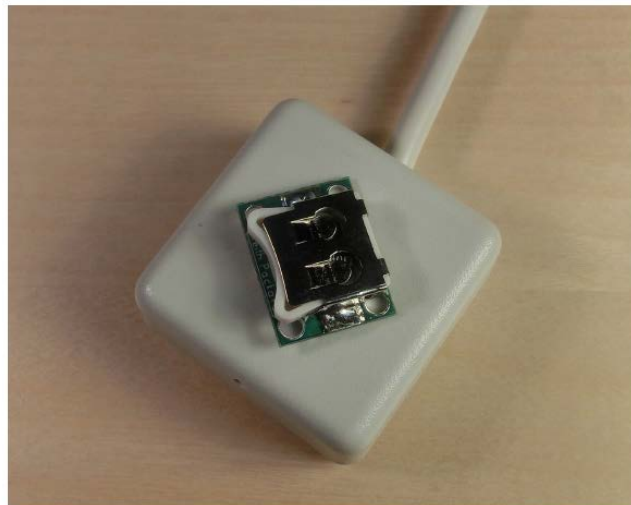


Figure A-5: Proper placement of altimeter on USB interface.

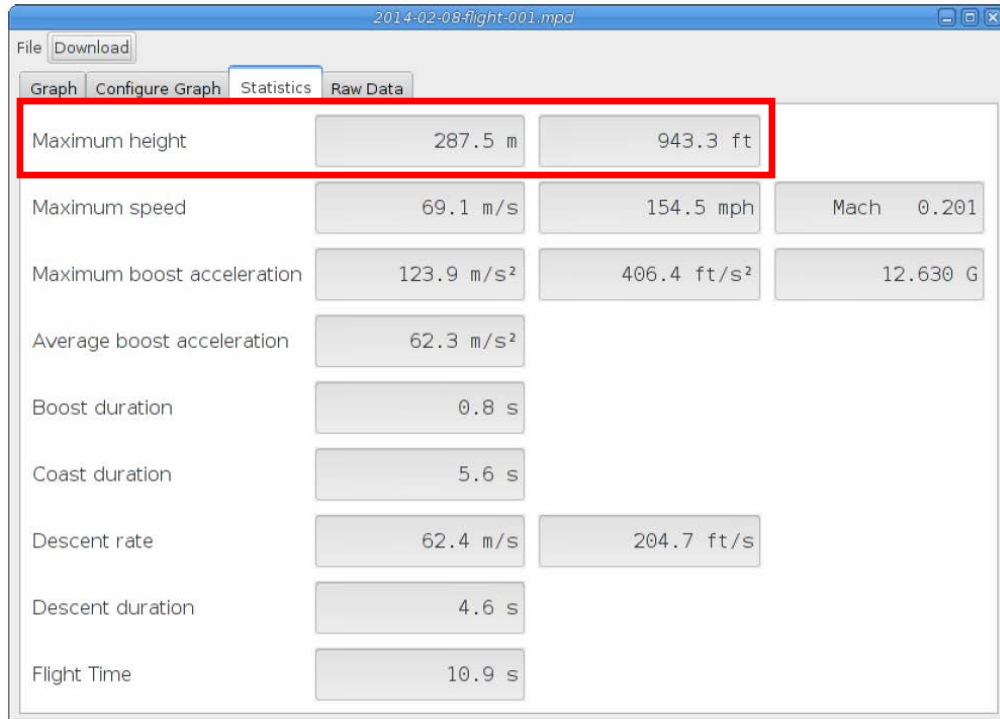


Figure A-6 Rocket flight statistics.

Reverse and Recycle: Engineering Group Project

Submitted by: Karen Groppi
Cabrillo College
kagroppi@cabrillo.edu

Engineering 5 Final Project Assignment **Reverse & Recycle - Engineering Group Project**

For this project you and your team will select an existing device, take it apart, reverse engineer it, then create something new out of the parts. 75% of your new invention should be built from pieces of the existing device.

You will be using and developing engineering skills in design, graphics, research, sustainable design, planning, teamwork, communication, time management, problem solving and more.

You will work in groups of 3 to 4 persons.

The project will have 3 phases:

Planning & Formulation

Deconstruction & Documentation, Design

Re-construction & Documentation

There will be 7 Progress Reports along the way.

The Final Report and Presentation will be due on the last day of class.

Project Due Dates:

Progress Report 1	March 28
Progress Report 2	April 18
Progress Report 3	April 25
Progress Report 4	May 2
Progress Report 5	May 9
Progress Report 6	May 16
Progress Report 7	May 30
Final Project and Presentation	June 6

Phase 1: Planning & Formulation

Progress Report 1: Team Agreement - Due March 28th

Tasks:

Meet with your group to establish a name.

Exchange contact information.

Decide when and where you can meet.

Make a Team Agreement:

- Team Name
- Mission Statement: A statement of what the group will accomplish and how they will do it.
- Values: A list of fundamental truths that will not change over time. Include values which each member cares about.
- Norms: A list of guidelines or expectations that define how each team member will treat others and be treated.
- Include: Your preferred decision making process.
- Leadership/organization style.
- Contact information for each group member.
- Signatures of each member
- Print a copy for each member and one to submit by the end of class today

Submit Progress Report 1: Hard copy of team agreement

Progress Report 2: Brainstorming and Deciding - Due April 18

- Brainstorm 10 devices of interest.
- Brainstorm 10 ways to get chosen device or product.
- Brainstorm 10 things you could reinvent from device.
- Choose your device and obtain one that is broken and no longer used.

Submit Progress Report 2: Team name and a summary of brainstorming results

Phase 2: Deconstruction & Documentation

Progress Report 3: Deconstruct and Document - Due April 25

Tasks:

1. Write a description of the original device and what it is used for
2. Disassemble the device. Sketch an assembly drawing of how the parts are arranged.
3. Learn how it works and write a description of how it operates.
Research at the library and on the internet [How Stuff Works: http://www.howstuffworks.com/](http://www.howstuffworks.com/)
4. Name each part.
Try to learn what each part is and what it does. If you don't know its actual name, make one up that reflects its function.
5. Prepare sketches of each part.
Sketches may be made by hand, AutoCAD, Google Sketch-up or any other program you'd like to try.
Sketches may be 2D or 3D.

Submit Progress Report 3: Description of how device works, assembly drawing and sketches of each part.

Progress Report 4: Sustainability - Due May 2

Analyze the sustainability of the original device and its parts:

Find out:

1. Where was the device made?
2. Who made/assembled the device?
3. Why was this device made?
4. What is each part made out of?
5. What resources are taken from the earth to make each part?
6. How long is the expected life of the device and parts?
7. What will happen to the parts when the device has exceeded its design life?
8. Are there other ways to get the benefits this device provides without the waste or negative impacts?
9. Suggest 5 ways to make this product more restorative or sustainable or to achieve its intent otherwise. (consider the earth, economic & social systems)

Submit Progress Report 4: Answers to these questions, Sustainability Analysis

Phase 3: Reconstruction & Documentation

Progress Report 5: Design Process Reconstruction - Due May 9

Brainstorm, Analyze, Select, Implement, Iterate

Tasks:

1. Use the design process to create an entirely new device from the original one.
 - Refer to brainstorming notes
 - Generate new ideas
 - Use a selection matrix to decide which new device to build. Be practical. Consider things like: Is implementation possible in time available? Does someone have the right tools? Is everyone on board?
2. Work as a team to make a plan of how to implement the new device. Help make sure everyone understands the plan by making sketches and making notes.
3. Implement... Start building your prototype.
4. Deal with problems as they come up.
5. Return to step 2 (or even step 1) until you have it working.

Submit Progress Report 5: Copies of notes and sketches, photos of the process

Progress Report 6: Design Process Reconstruction - Due May 16

Document your prototype.

Tasks:

1. Create an Assembly Drawing that shows each part used and how they are connected to make the new invention.
2. Label the parts to correspond with the parts list.
3. Make a Parts List that includes:
 - Name of the part
 - Material
 - Sheet number where drawing for individual part can be found.
 - Indication if part is from old device or new
 - Percentage of parts from old device (should be equal to or > 75%)

Submit Progress Report 6: Assembly Drawing & Parts List

Progress Report 7: Description of new device, Presentation Outline - Due May 30

Tasks:

Description of the new device and order of operations.

Discuss the evolution of your design:

- How did the thing you created change since your first idea?
- What problems did you solve along the way?
- What would you do differently next time?

Presentation Outline (order of topics and who will speak about them)

Submit Progress Report 7: Description of new device and design evolution and presentation outline

Final Report & Presentation

Final Report

Create a Final Report by assembling **revised** submittals and numbering pages.

Final report shall also include:

1. A photo and description of the original device and what it is used for
2. A photo and description of the new device and what it is used for and how it works.
3. The order of operations and how the original device worked
4. The order of operations and how the new device works
5. Analysis of what failed on the device if it was not working, and why.
6. Your suggestions for improving the design, useful life or sustainability of the old product.
7. Bibliography & references

Help for writing the report: [SJSU Help with writing research reports & term papers](#)

Presentation

Prepare a Presentation

Prepare a 10-15 minute presentation that includes visual aids and covers what is included in your report.

Explain your design process, ideas, what you tried, what didn't work and what did. Include what you gained by doing this project.

Each member of the group shall participate in the presentation.

Presentation shall include a demonstration of your New Device.

Complete the Group and Self-Assessment forms to be turned in with your report at the conclusion of your presentation.

Submit Final Report, Presentation, & Evaluation June 6

Grading Criteria: Engineering 5 Reverse, Recycle & Re-Engineer Project

For details on items listed see the Project Specifications.

Team Members:		Product:		
Preliminary Documents		Poss. Pts	Pts Rec.	Notes:
Progress Report #1 : Team Agreement		2		
Progress Report #2 : Brainstorming Documents				
10 devices to dismantle, 10 ways to get device, 10 things to reinvent		2		
Progress Report #3 : Deconstruct & Document				
Description of original device an how it works		1		
Part names & Sketches		1		
Progress Report #4 : Sustainability				
Answers to 7 questions & 5 ways to make device more restorative		2		
Progress Report #5 : Reconstruction & Documentation				
Notes, sketches, selection matrix of reconstruction		2		
Progress Report #6 : Reconstruction & Documentation				
Assembly drawing & Parts List for new device		2		
Progress Report #7 : Description of new device, design evolution , Presentation Outline				
Describe new device, Set order of topics and speakers		2		
Total Submittals 14 points				
Final Design Documents		Poss. Pts	Pts. Rec.	Notes:
Final Written Report				
Title page (name of group & members, date, Engr 5, names of devices)		2		
Old Device				
Photo & description of original device what it is used for and how it works		7		
Assembly Drawing & Parts list: names and sketches how parts fit together		7		
Sustainability Analysis for original device (discussion of 7 questions)		7		
Five ways to re-engineer this product to be more restorative		7		
Analysis of how old device failed and why		7		
Suggestions for re-engineering old device		7		
New Device				
Brainstorming ideas for new device		7		
Assembly Drawing for new device, label parts to match original parts list		7		
Include material of part and reference page for sketch				
Photo and description of new device and how it works		7		
Report is typed, organized, with page headings, page nos., good grammar & spelling		4		
Report Total				
Presentation				
Visual aids		2		
How devices work, old & new		2		
How you would re-engineer original device to improve sustainability		2		
Demonstration of new device operation		5		
Everyone participates		2		
10 - 15 minutes		2		
Presentation Total 15 points				
Evaluation forms		2		
Total Final Project points		100		
Wild Points (something done exceptionally well)				
TOTAL POINTS			0	Project is 20% of overall class grade

Robot Rat Race Tournament

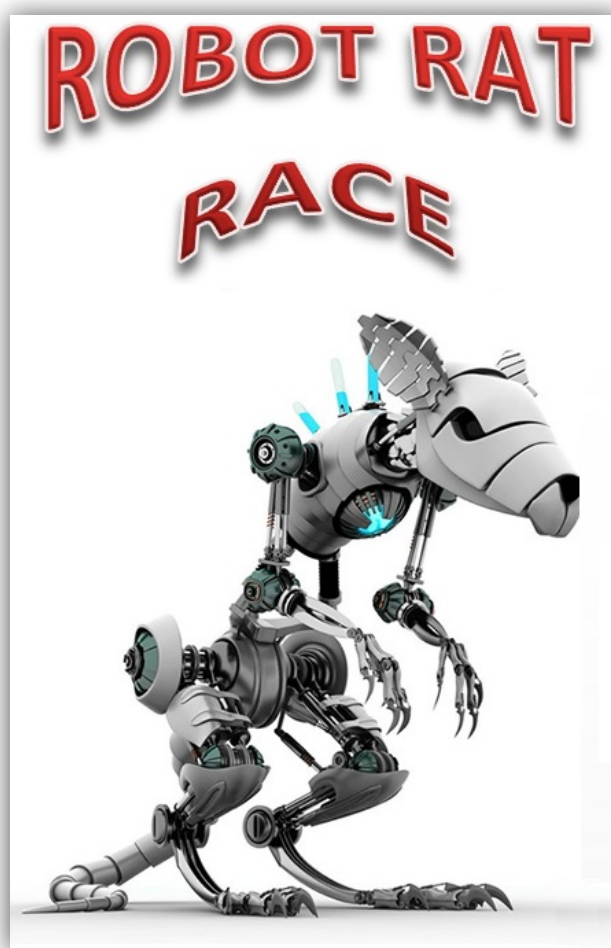
Submitted by: Carlotta Berry
Rose-Hulman Institute of Technology
berry123@rose-hulman.edu



ECE 160

Product Design Specification (PDS)

Fall 2013- 2014



Version 2.0



Design Project Overview:

Rodents, Ltd., a division of Rose Enterprises, has oversight of the **ROBOT RAT RACE (RRR) Tournament**. Rodents, Ltd. would like to solicit your team's professional expertise to design and construct a *system* for entry into the Tournament.

Your team is required to design and construct a *system* also referred to as *robot* that will execute the requirements for the **RRR Tournament** in a superior manner. The tournament playing field consists of a planar surface with a complex maze and several secure cheese vaults (CV). Your robot must retrieve valuable cheese balls (CB) from the cheese holes (CH) in the maze and deliver them to the vaults. The robot rat can earn a significant amount of points by delivering the higher point cheese balls to the more secure vaults. The robot can earn additional points by also returning cheese balls back to the cheese holes at the end of the match.

The product design specifications and contest rules have been sent out to many well-qualified and motivated teams. Therefore, your team must be deliberative and diligent in order to work hard to come up with the most creative and effective solution.

Project Competition:

The final project competition, **ROBOT RAT RACE Tournament**, will be held at **17:30 on Thursday, November 7, 2013 in the Kahn Arena** (aka, the Kahn Rooms of the Hulman Memorial Union). Each team will show off their design to the amazement of their peers. Attendance at the competition is mandatory, any conflicts must be discussed with, and approved by, your instructor before the dry run. Due to the competition, the 18th day of class (9-2) will be cancelled.

Project Awards:

There will be four awards presented based upon the competition results as well as design and team spirit. The four award categories are listed below:

- "The Big Cheese" - Overall Tournament Winner;
- "Havarti Hoarder" - Overall Tournament Runner Up;
- "Colby Creator" - Most creative, unique and elegant solution;
- "Swiss Swagger" – Most "spirited" team with an identifiable non-offensive theme;

The first two awards will be determined by the competition scoring. The last two awards are voted on by ballot during the final project competition. Instructors and students in the course are not allowed to vote. Teams that earn an award will earn bonus points on the final project of 10, 7, 5, and 3 points for The Big Cheese, Havarti Hoarder, Colby Creator, and Swiss Swagger, respectively. Some these teams will also receive certificates and trophies during the competition or final presentations.



Product Design Specification

The requirements for the *system* are as follows:

General Requirements:

1. The *system* shall be designed to perform on the competition surface shown in Figures 1 and 2.
2. Each cheese ball is the shape, size and weight of a ping pong ball.
3. The *system* shall be constructed only from LEGO® pieces included in the provided kit as listed in the kit inventory. The team may also request no more than one (1) light sensor from the ECE parts room if they have a need. In order to organize the robot cables, the team may also request two (2) meters of orange kite string from the ECE parts room or use their own tie wraps. Tie wraps may only be used to organized cables and for no other use.
4. The laptop of one team member will be allowed to be used as control device for the *system* but it is not considered a part of the *system*. This laptop may only be used during the remote control period of the match.
5. The *system* shall operate without damaging the competition surface or any of the game pieces.
6. The *system* shall be designed to operate for the entire match without any reconstruction or reconfiguration.
7. The *system* shall be designed to operate without any physical contact (this includes wired connections to the laptop) during each match of the tournament. During the remote control period, the laptop of one team member may be used to control the *system* as long as there is no physical contact during the match. All controlling devices must be operated by the same team member during the entire match.
8. The system shall collect CBs from any of the four (4) CHs within the maze. The system shall deliver CBs to any of three (3) CVs.
9. The system shall not damage the competition surface, CB, CH, CV or any other system.

Materials:

1. Student teams shall use only the parts provided in their LEGO® kit as listed on their kit inventory, parts provided by the ECE Parts Room to complete their kit inventory or as listed in this document including string and one additional light sensor, if required. One team member laptop will be allowed to be used as control device for the *system* but it is not considered a part of the *system*.
2. No additional parts shall be added to the inventory received from the ECE Stockroom without prior written approval by the ECE160 faculty.



3. Parts shall not be traded between teams.
4. Students shall not modify the kit or parts provided.

Students shall return the LEGO® NXT kit as listed in the kit inventory and all additional parts supplied with all items accounted for one day before the final presentation session held during Finals Week.

Communication/Programming requirements:

1. The *system* shall be programmed in NXT-G (the drag and drop language supplied by LEGO®) or any other appropriate programming language selected by each team. Any programming language/operating system software used must be available to all students at no cost, freeware, or available through existing Rose-Hulman licenses. All teams using any software other than NXT-G must provide information for acquiring the software to the instructor (who will publicize this to all teams).
2. During the autonomous period, the system must move to the CH, acquire CBs, move to the CVs and release CBs with no human intervention or communication with the laptop.
3. During the teleoperation (remote control) period, the laptop may be used to communicate with/control only your team's NXT within the wireless capabilities of the control devices. This will typically be done via Bluetooth. The control devices may not be positioned within, over, or under the competition surface during the match. The laptop may be placed on the surface adjacent to the competition surface during the team's competition.
4. The *system* may communicate with the control devices via wireless or wired connection during the set up period before each match, but only via wireless communication channels during the match.

Competition Surface:

- The competition will be held on a table that is 6' x 6'. The competition surface will be flat black and is shown in Figures 1 and 2. There will also be a white line on the surface for line following as shown in Figure 2. The walls of the competition table will be wood color. Figures 1 and 2 include the dimensions of the table and the locations of the CBs, CHs and CVs.
- The CH consists of 2 dimples that are $\frac{3}{4}$ " deep in the table to keep the balls from rolling. Each CH will contain two (2) CBs located in dimples on the playing surface that touch the back wall of the CH. There will be a 2" center to center separation between the CB with a $\frac{1}{2}$ " gap between them. The ping pong balls are approximately 1 $\frac{1}{2}$ " in diameter.
- The CV consists of dimples in the table that are $\frac{3}{4}$ " deep to keep the balls from rolling and there will be two $\frac{1}{8}$ " high rails along each side of the vault to confine the CBs. The front of the vault will be open to accommodate the robot driving in and out.



- The competition surface and CBs will be provided for practice during the project design and construction phase. The competition surface shall not be modified or damage in any way during practice or competition.

Competition Rules:

1. The *system* shall be defined to be that device constructed by the team to perform the designated tasks.
 - a. The *system* shall not include a computer, infrared port or external wires, or any part of the competition surface.
 - b. The *system* shall include the constructed device and any software loaded into the NXT.
 - c. All components of the system must remain connected to all other parts of the system during the match. If parts become detached from the system during the match they may not be removed from the competition surface until the conclusion of the match. Detached part(s) may be reattached during subsequent setup periods or between matches.
2. The *system* may be started from any point on the competition surface. No part of the *system* may extend over the outer walls of the competition surface.
3. The tournament is comprised of a seeding round and several elimination rounds. Round 1 is used to determine seeding for the second round of the tournament and it will be based upon the **dry run** conducted on day 17 (9-1) of class. During the dry run each team will complete their first match of round 1 and earn a score. Since this also serves as the practice day, if a team so chooses, they may elect to run twice and keep the higher of the two scores. Since the dry run is also the **design freeze**, once the team completes their first run, they are not allowed to make any changes to the physical hardware design of their system although the program may still be modified.
4. During the second day of the competition (Thursday night), there will be paired matches or heats consisting of two competition tables with one team per table in the elimination rounds. The pairings will be based upon the seeding determined from the maximum of the dry run scores.
5. Each match will consist of a 1 minute setup period followed by a cheese retrieval period lasting up to 3 minutes. At the beginning of the setup period, a die will be rolled for each table to determine the point values for the CBs at each CH. During the 1 minute setup period the *system* may be programmed or reprogrammed. By the completion of the setup period, the system should be turned on but must not move before the 180 second competition clock is started by the officials.
6. The cheese retrieval period will consist of a 30 second autonomous period and a 2 ½ minute teleoperation period. The match will end 3 minutes after the setup period, or when both systems have stopped moving, whichever comes first. Teams have the



option of designing the system to be fully autonomous for the 3 minutes and this will be addressed in more detail later in this document.

7. A roll of a die at the beginning of the setup period will establish the data set (1-6) to set the points values for the CBs at each CH. The roll of the die indicates the CH for the 7 point, 5 point, 3 point, and 1 point CBs, respectively. If the die reads 6 or blank, the team can choose which data set to use. The coding for the roll is indicated by Table 1. For example if you roll a 4 then CH 4 has 7 pt balls, CH 3 has 5 pt balls, CH 2 has 3 pt balls and CH 1 has 1 pt balls.

Table 1: Die and Cheese Hole Code/Data Set

Roll	Code/CH #
1	1234
2	2413
3	3142
4	4321
5	1432
6	team's choice

8. Immediately following the roll of the die, tournament officials will place the CBs at the proper CH according to the code listed in Table 1. The CBs will be white small ping pong balls with small, dark, numeric symbols indicating their value.
9. The competition time begins once the CBs are placed and tournament officials give the starting signal. The system should immediately autonomously navigate the maze for 30 seconds to retrieve CBs from the CHs and deliver them to the CV.
10. The CV has a security value (SV) that is inversely proportional to the width of the CV. For example, the narrowest CV has a SV = 5, the intermediate CV has a SV = 3 and the widest CV has a SV = 1.
11. The narrowest CV can hold two (2) CBs; the intermediate CV can hold four (4) CBs, the widest CV can hold six (6) CBs. Therefore, there are 12 possible vault locations where the CBs can be delivered.

Scoring Rules:

1. The competition will be a single elimination tournament where the scores reset after every round. After round 1, the teams will be ranked to determine their seeding for round 2. After round 2, the teams will be ranked based upon their round 2 match score. The top half of the teams will move on to round 3. This will continue every round until we have the competition winner.
2. The value of each CB delivered to a SV will be calculated by using the following formula:

$$CB * SV$$



The total score for all of the CBs delivered to the vault will be determined by the following formula:

$$CB_del = \sum CB * SV$$

For example, if a 3 point CB is retrieved from a CH and delivered to a CV with a 5 point SV, then the $CB_del = 3 * 5 = 15$. CBs can only be scored once for placement in the CV.

3. Additional scoring can only occur if all CBs have been successfully moved to a CV. Then a team can score additional points by returning the CBs to a CH. A successful CB return occurs only when the CB is positioned on a CH dimple and remains at rest and not in contact with the system for at least 10 seconds. A CB can only be scored once when placed within the CH.
4. If a CB is returned to the original CH then the score will be double the delivery score as determined by the following formula:

$$CB_retA = 2 * CB * SV$$

For example, if you return a 3 point CB to the original CH from a CV with a 5 point SV, then $CB_retA = 2 * 3 * 5 = 30$.

5. If a CB is returned to a CH other than the original then the score will be the same as the delivery score determined by the following formula:

$$CB_retB = CB * SV$$

For example, if you return a 3 point CB from a 5 point CV to any CH other than the original, then $CB_retA = 3 * 5 = 15$.

6. The total score for all of the CBs returned to the CH from the CV will be determined by the following formula:

$$CB_ret = \sum CB_retA + \sum CB_retB$$

7. All scores are doubled during the 30 second autonomous period. A team may choose to complete the entire match autonomously and scores will be doubled for the entire match.
8. Scoring for a team's match within a round of the competition will be calculated as the sum of the scores of the CBs delivered to the CV and CBs returned to the CH as calculated by the following formula,

$$Match_Score = \sum CB_del + \sum CB_ret$$

9. If there is a tie which prohibits the judges from determining the team that will move on to the next round then there will be a sudden victory playoff match. During the sudden victory round, the first team to score a point moves on. If there is a 3 way tie, the team that moves on will be determined by the flip of a coin.

**Judging:**

1. Judges shall be provided by Rodents, Ltd., and the ECE Department.
2. Judges will serve as announcers, officials and scorekeepers. Therefore, judges will be responsible for insuring that the teams abide by the rules of competition, scoring each round and giving the score sheet to the official scorekeeper. Officials, announcers, score keepers along with team members will be responsible for field reset.
3. The decision of the officials shall be final. Decisions may be challenged for factual or interpretive errors; but, the officials have the final decision on the validity of challenges.
4. Teams shall ask the instructor for rule interpretations in a timely fashion. Requests for rule interpretations made after 5:00 p.m. on November 7, 2013 will be rejected.

Other Requirements:

1. All students shall participate with their team in the competition.
2. All students shall participate with their team in the design process for this project.
3. All team members shall participate equally in the project work.
4. All team members shall receive a grade on the project based upon the entire team's work as well as their individual work.
5. Students shall respect the creativity of others.
6. Students shall in no way interfere with another team's project work or interfere with another team during the competition.

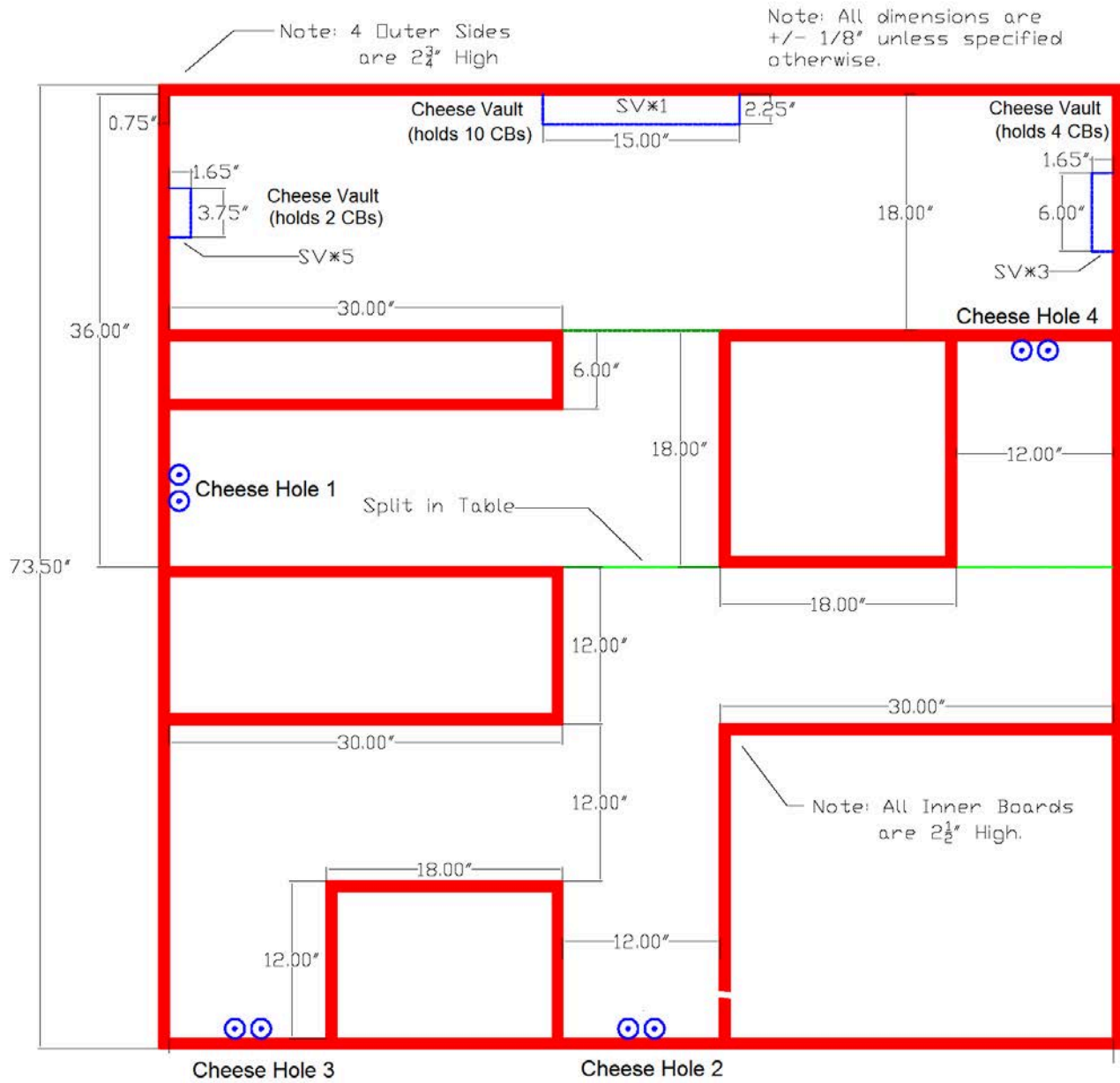


Figure 1: Competition Surface with Dimensions

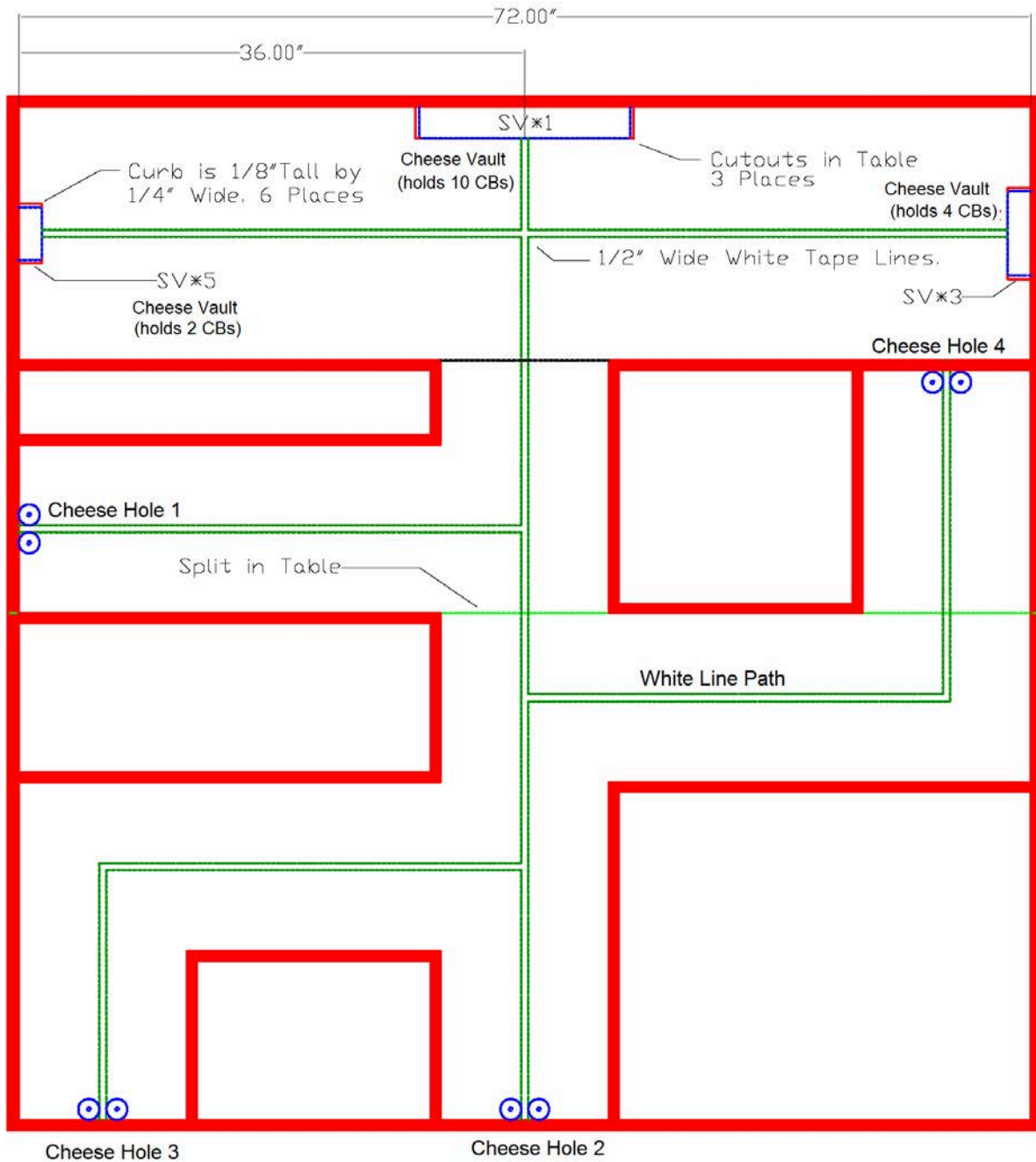


Figure 2: Competition Surface with Line Path



Acronym Glossary

CB – Cheese Ball
CH – Cheese Hole
CV – Cheese Vault
SV – Security Value
RRR – Robot Rat Race

Tremendous Towers Tournament

Submitted by: Carlotta Berry
Rose-Hulman Institute of Technology
berry123@rose-hulman.edu

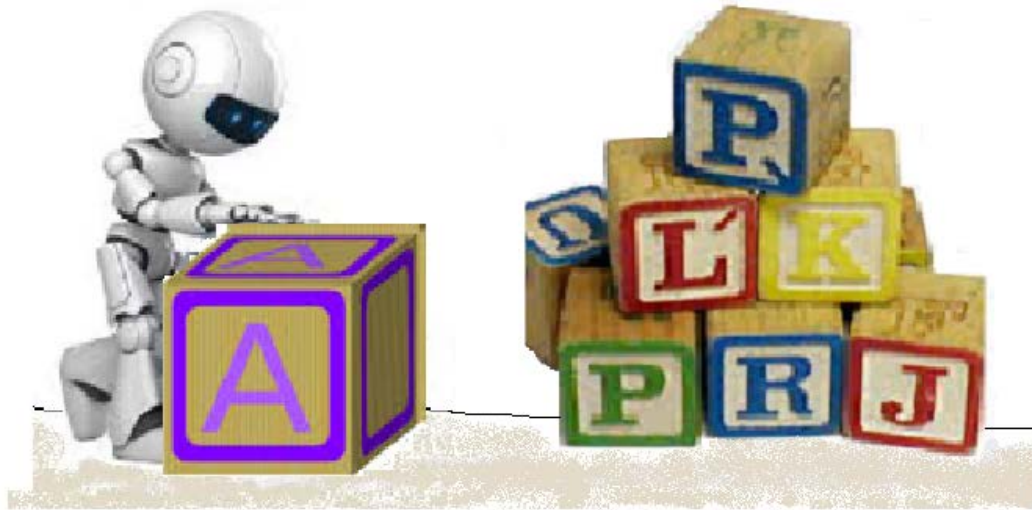


ECE 160

Product Design Specification (PDS)

Fall 2014- 2015

Tremendous Towers Tournament



Version 3.0

**Design Project Overview:**

Edison and Wozniak, Ltd., a division of Rose Enterprises, has oversight of the **TREMENDOUS TOWERS TOURNAMENT (TTT or T³)**. Edison and Wozniak, Ltd. would like to solicit your team's professional expertise to design and construct a robotic system for entry into the tournament. Your team is required to design and construct a system that will execute the requirements for the tournament in a superior manner. The tournament playing field consists of a planar surface with a complex maze that includes block storage and tower spaces. Your robot must retrieve blocks from the storage area and use them to build towers in the tower space. The product design specifications and contest rules have been sent out to many well-qualified and motivated teams. Therefore, your team must be deliberative and diligent in order to work hard to come up with the most creative and effective solution.

Project Competition:

The final project competition, **TREMENDOUS TOWERS TOURNAMENT**, will be held at 17:30 on Thursday, November 6, 2014 in the Kahn Arena (aka, the Kahn Rooms of the Hulman Memorial Union). Each team will show off their design to the amazement of their peers. **Competition results do not impact your course grade; however attendance at the competition is mandatory.** Any conflicts must be discussed with, and approved by, your instructor before the dry run. Due to the competition, the 18th day of class (9-2) will be cancelled.

Project Award Categories:

- “The Tower Terror” - Overall Tournament Winner;
- “The Block Boss” - Overall Tournament Runner Up;
- “Brilliant Builder” - Most creative, unique and elegant solution;
- “Charismatic Constructors” – Most “spirited” team with an identifiable non-offensive theme;



Product Design Specification

The *system* requirements are as follows:

General Requirements:

1. The *system* shall be designed to perform on a 6' x 6' competition surface.
2. Each building block is the size, shape and weight of a baby block. All blocks are 1 ¾" one each side with a mass of 35g.
3. The *system* shall be constructed only from LEGO® pieces included in the provided kit as listed in the kit inventory; one additional light sensor may be requested by each team; two (2) meters of orange string may be requested at the ECE stockroom by each team. A laptop of one team member will be allowed to be used as a control device for the *system* but it is not considered a part of the *system*. Teams may have the option of switching drivers during the match if they desire.
4. The *system* shall operate without damaging the competition surface, any of the game pieces or other systems.
5. The *system* shall be designed to operate for the entire match without any reconstruction or reconfiguration.
6. The *system* shall be designed to operate without any physical contact (this includes wired connections to the laptop) during each match of the tournament.
7. During the remote control period, the laptop of one team member may be used to control the *system* as long as there is no physical contact during the match. Teams have the option of switching drivers during the match if they desire. Teams also have the option of tethering a video game controller to the laptop to communicate with the system.
8. The system shall collect building blocks from storage in the maze and use them to build towers in the tower space.
9. There are no height or weight restrictions on the *system*.

Materials:

1. Student teams shall use only the parts provided in their LEGO® kit as listed on their kit inventory, parts provided by the ECE Parts Room to complete their kit inventory or as listed in this document including string and one additional light sensor, if required.



2. No additional parts shall be added to the inventory received from the ECE Stockroom without prior written approval by the ECE160 faculty.
3. Parts shall not be traded or shared between teams.
4. Students shall not modify the kit or parts provided.
5. Students shall return the LEGO® NXT kit as listed in the kit inventory and all additional parts supplied during the class period before the final presentation to be held during final exams week.

Communication/Programming requirements:

1. The *system* shall be programmed in NXT-G (the drag and drop language supplied by LEGO®) or any other appropriate programming language selected by each team. Any programming language/operating system software used must be available to all students at no cost, freeware, or available through existing Rose-Hulman licenses. All teams using any software other than NXT-G must provide information for acquiring the software to the instructor (who will publicize this to all teams).
2. During the autonomous period, the system must move to the blocks, pick them up, move them to the tower space and build one or more towers. This entire task must be completed without human intervention or communication with the laptop.
10. During the teleoperation (remote control) period, the laptop may be used to communicate with and control only your team’s *system* by using Bluetooth. The control devices may not be positioned within, over, or under the competition surface during the match. The laptop may be placed on the surface adjacent to the competition surface during the team’s competition. Teams also have the option of tethering a video game controller to the laptop to communicate with the system. The driver will be required to remain in the driver’s box during remote control.
3. The *system* may communicate with the control devices via wireless or wired connection during the set up period before each match, but only via wireless communication channels during the match.

Competition Surface:

The competition will be held on a fixed operating surface that is 6’ x 6’. The competition surface will be flat black with white lines. The competition surface and blocks will be provided for practice during the project design and construction phase. The



competition surface shall not be modified or damaged in any way during practice or competition. The game field with the dimensions is shown in Figure 1.

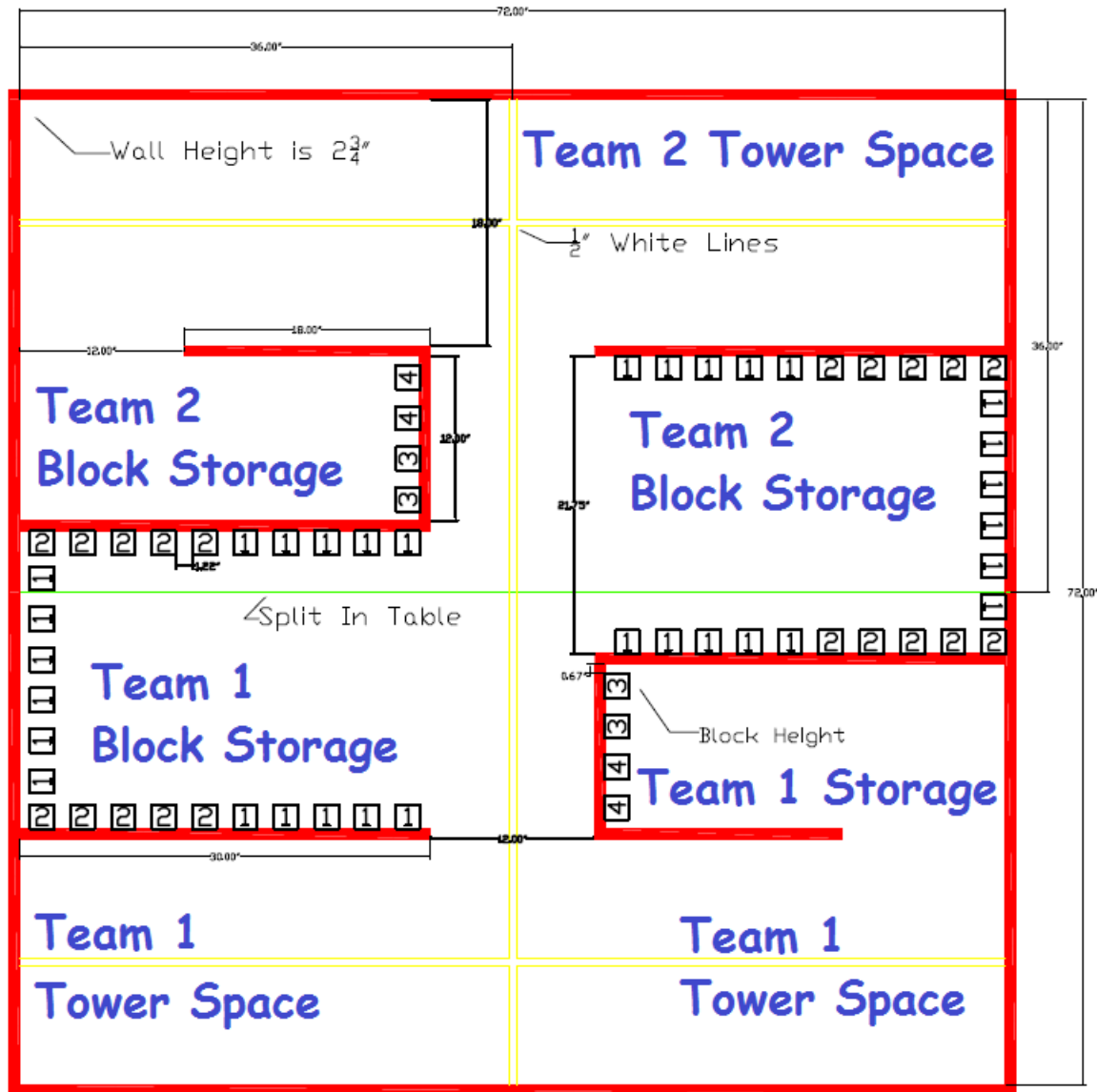


Figure 1: Competition Surface with Dimensions and Line Path

Legend

Red Lines - wood walls with a height of $2 \frac{3}{4}$ "

Yellow Lines – white path on painted black surface

Number – number of blocks in each stack

**Competition Rules:**

1. The *system* shall be defined to be that device constructed by the team to perform the designated tasks.
 - a. The *system* shall not include a computer, infrared port or external wires, or any part of the competition surface.
 - b. The *system* shall include the constructed device and any software loaded into the NXT.
 - c. All components of the system must remain connected to all other parts of the system during the match by parts supplied in the LEGO® kits.
2. The tournament is comprised of a seeding round and several elimination rounds. Round 1 (the **dry run** conducted on day 17 of class) is used to determine seeding for the second round of the tournament.
3. During the second day of the competition, there will be paired matches or heats consisting of two competition tables with two teams per table in the elimination rounds.
4. Each match will consist of a 1 minute setup period followed by a tower building period lasting up to 3 minutes. The tower building period will consist of a 30 second autonomous period and a 2 ½ minute teleoperation period. The match will end 3 minutes after the setup period, or when both systems have stopped moving, whichever comes first. Teams also have the option of designing the system to be fully autonomous for the entire 3 minutes.
5. During the 1 minute setup period the *system* may be programmed or reprogrammed. However, parts may not be added to or removed from the *system* during the elimination rounds. The design freeze for the system occurs at the beginning of the dry run.
6. If parts become detached from the *system* during the match, the team has 5 seconds to remove them from the competition surface. Detached part(s) may be reattached during subsequent setup periods or between matches.
7. The *system* may be started from any point on the competition surface but no part can touch a block, wall or extend over the outer walls of the competition surface.
8. Each team will have 50 blocks in their two storage areas and can use these to build any number of towers in their space.



9. Teams can build towers on any part of the space on their end of the table.
10. Teams will share the middle lane to go into the maze to retrieve the building blocks.
11. Once a team has used all of their assigned blocks, they are allowed to steal blocks from the opposing team's storage space. A block is not considered used until it is included in a tower that is at least 2 blocks high. Teams cannot take blocks from the opposing team's towers or system.

Scoring Rules:

1. A block on the bottom level of a tower will earn 2^0 points per block, a block on the 2nd level will earn 2^1 points per block and the n th level would be $2^{(n-1)}$ points per block. A tower must contain at least 2 blocks in order to earn a score. Figure 2 shows sample tower scores.
2. All block scores will double during the autonomous period. For example a block on level 0 would be $2(2^0)$ per block and a block on the n th level would be $2(2^{(n-1)}) = 2^n$ points per block. A tower must contain at least 2 blocks in order to earn a score. Figure 2 shows sample tower scores.
3. The system may use autonomous control for the first 30 seconds of the match or for the entire three minutes. A team will earn double points for the amount of time that the robot operates in autonomous mode. If a team chooses autonomous control for the entire match, they cannot switch to remote control at any point.
4. The match score will be based upon the number of blocks remaining on the tower at the end of a 30 second period. If a block falls from the tower before or after the 30 second period, it is out of play. Please see the *penalties* section for more details on this rule.
5. The tower must be built in the tower space to be counted in the final score.
6. The tower must be free standing and not touching the system or walls in order to be counted in the tower score.
7. Once a block leaves the competition surface, it is out of play for the match.
8. The match score will be used to determine the teams that move on to the next round. This is not a head to head match and all teams will be ranked at the end of each round to determine which teams will move on. All teams will compete in round 1 and 2 before



potential elimination occurs. Round 1 score is used to determine match order for Round 2.

9. The match score will reset at the end of every round.

10. In the event of a tie, the teams will complete in a head to head 3 minute match and the team with the tallest tower at the end of the period will win.

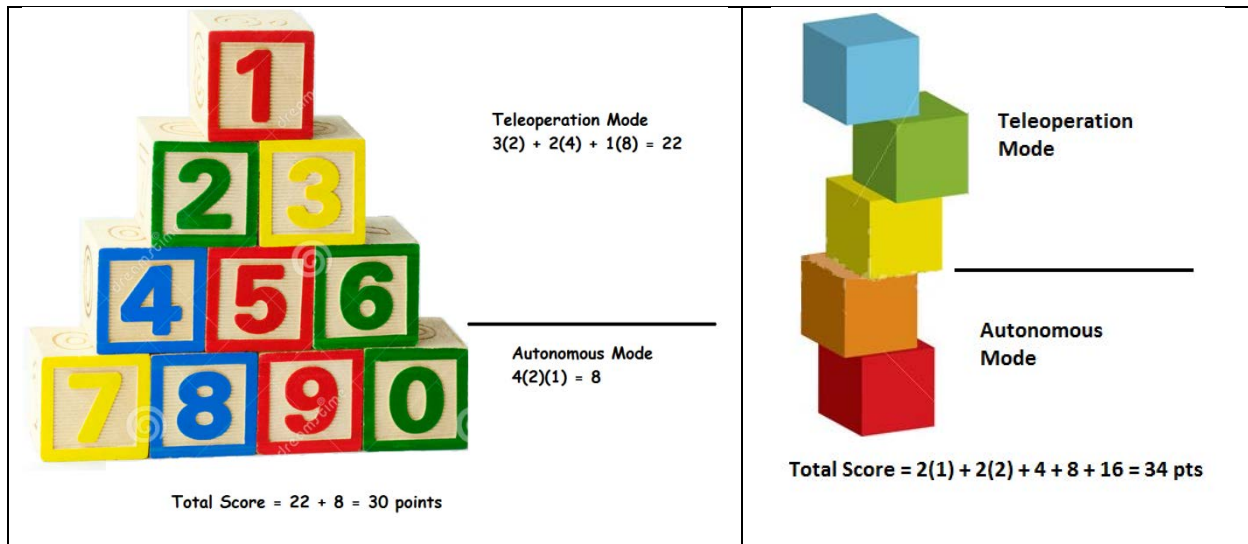


Figure 2: Sample Tower Scores

Penalties:

1. A team will receive a 5 point penalty for any of the following infractions:
 - Each block knocked off of an opposing team’s tower
 - System damaging the course or game pieces
 - System touching wall or block at the start of the match
2. A team will receive a 5 point penalty for blocking an opposing system’s path for more than 5 seconds. If it appears that the system is blocking an opposing system repeatedly or being overly aggressive there will be a 5 point penalty immediately even if it is less than 5 seconds.
3. A team will receive a 5 point penalty if there is contact with the opposing system for more than 5 seconds. If it appears that both systems are in forward motion and there is simultaneous contact there will be off setting penalties.
4. A team will receive a 5 point penalty if parts detached from the system during game play and they are not removed from the field within 5 seconds.



5. A team will incur a 5 point penalty each time the driver leaves the driver's box.

Judging:

1. Judges shall be provided by Edison and Wozniak, Ltd and the ECE department.
2. Judges will serve as officials and scorekeepers. Therefore, judges will be responsible for insuring that the teams abide by the rules of competition, scoring each round, assigning penalties and giving the score sheet to the official scorekeeper. Officials, announcers, score keepers along with team members will be responsible for field reset.
3. The decision of the officials shall be final. Decisions may be challenged for factual or interpretive errors; but, the officials have the final decision on the validity of challenges. In order to challenge a rule interpretation, one member of the team should approach the head judges to request clarification within 5 minutes of the decision.
4. Teams shall ask the instructor for rule interpretations in a timely fashion. Requests for rule interpretations made after 5:00 p.m. on November 6, 2014 will be rejected.

Other Requirements:

1. All students shall participate with their team in the competition.
2. All students shall participate with their team in the design process for this project.
3. All team members shall participate equally in the project work.
4. All team members shall receive a grade on the project based upon the entire team's work as well as their individual work.
5. Students shall respect the creativity of others.
6. Students shall in no way interfere with another team's project work or interfere with another team during the competition.



Final Project

Simon Says

Objective:

Build a circuit to simulate the Simon Says electronic game. If you are not familiar with this game please review the following link: <https://www.youtube.com/watch?v=vLi1qnRmpe4>

Learning Outcomes:

At the conclusion of the project, the student should be able to

- wire a circuit with LEDs, resistors, buzzers, buttons, potentiometer and a LCD to create the Simon Says Game
- create a properly commented program to control the game and display game statistics
- demonstrate the working project and adequately explain the functionality
- completely document what was learned in weekly memos, a final report and presentation

Preparation:

In order to learn how to wire a circuit with the basic Simon Says game, please build Circuit#16 from the SparkFun Inventor's Kit Guide. Also, review Circuit#15 from the SparkFun Inventor's Kit Guide to recall how to build a circuit with an LCD. Make sure to completely review the code and comments for both circuits to fully understand how they work. It is important to understand them because you will need to modify these to complete the final project. It would also be helpful to review the mini-projects completed before.

Parts:

Arduino Microcontroller

2 - Breadboards

4 - LEDs

4 - 330 Ω resistors

4 - pushbuttons

32 – wires

1 – piezobuzzer

1 – potentiometer

1 - LCD

Procedure:

Part I – Original Simon Says

1. Build Circuit #16 from the SparkFun Inventor's Kit Guide (see Figure 1).



2. Create the program to simulate the Simon Says game as described in the SparkFun Inventor's Kit Guide.

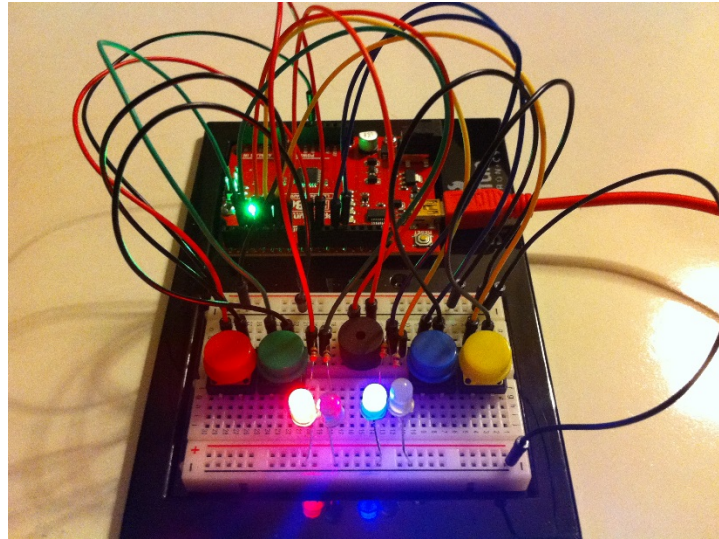


Figure 1: Original Simon Says Circuit

Part II – Charlieplex Resistors

1. In this part, you will modify the circuit and program to free up some digital pins so that you can add an LCD in the next part. This is necessary because the LCD circuit requires 6 digital pins and only 4 pins are available.
2. The first step is to reduce the number of digital pins used by the piezo buzzer. Disconnect the wire from digital pin 7 and connect it to ground. The buzzer now has one wire connected to digital pin 4 and the other wire connected to ground. The buzzer should still play all of the same game sounds although music may sound a little differently. Digital pin 7 will now be used for the LCD circuit.
3. Make sure to remove all of the references to BUZZER2 in the code. Run the program and confirm that it still works correctly. If not, debug and correct the program until it does.
4. In order to get the additional pin, the LEDs are going to be connected by using a new technique called *Charlieplexing*. In this method, 4 LEDs can be controlled by using 3 resistors and 3 digital pins. See the following video for more details on *Charlieplexing* (<https://www.youtube.com/watch?v=ECXk95qYZu8>).
5. The diagram and table in Figure 2 shows how the *Charlieplexing* would work with 6 LEDs and 3 pins. Figure 3 shows how the Charlieplex circuit looks on the breadboard.

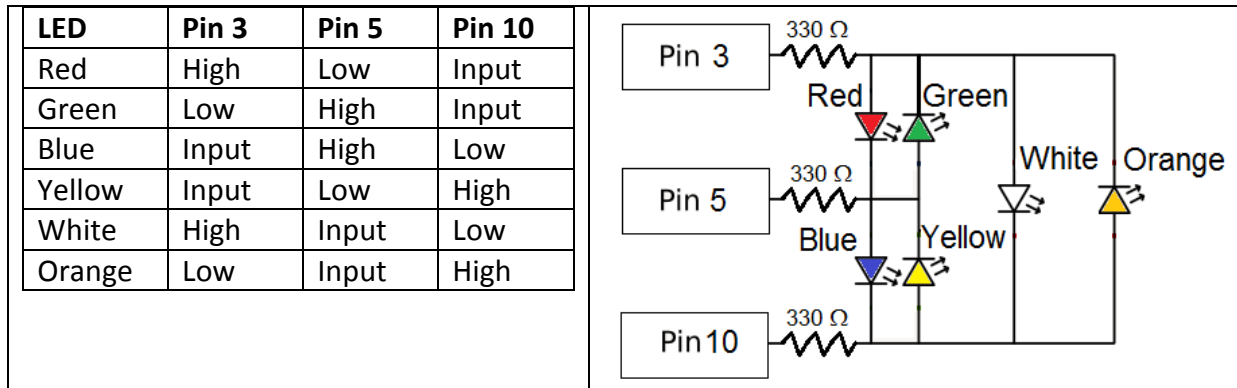


Figure 2: Charlieplexing 6 LEDs

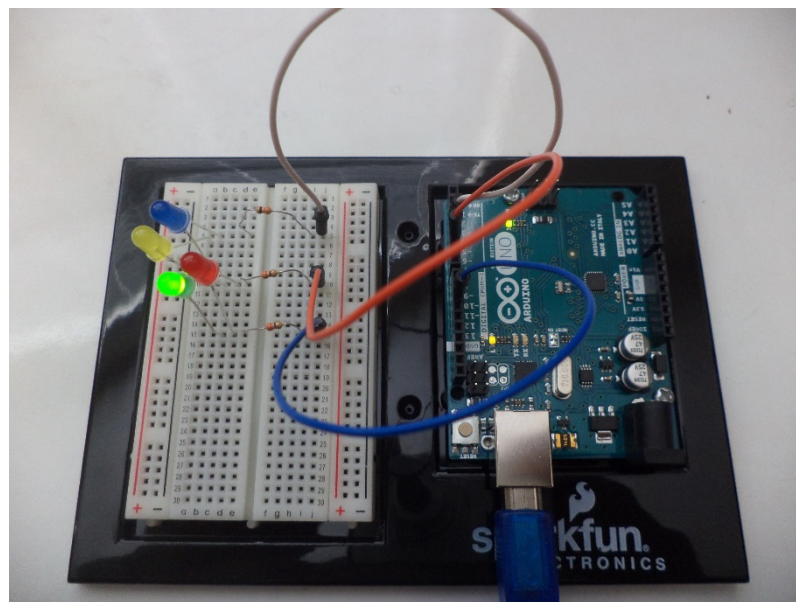


Figure 3: 4 LEDs in Charlieplex circuit

6. The good news is that there is actually an Arduino library that will charlieplex LEDs for you. You will need to download and install the library as described on the website. Please read more information about this at the following link:

<http://playground.arduino.cc/Code/Charlieplex>

7. On a new breadboard, build the circuit in Figure 3 which is the same as Figure 2 except the white and orange LEDs are removed. Compile and upload the program to the Arduino microcontroller. **Note that there will be a library error when you attempt to compile the example because it was made for an older version of Arduino software.** This can be corrected by opening the file, "C:\Program Files (x86)\Arduino\libraries\CharliePlex\Charlieplex.h" in a text editor. Next, replace the line, #include <WProgram.h> with the following code:

```
#if defined(ARDUINO) && ARDUINO >= 100
#include "Arduino.h"
#else
#include "WProgram.h"
#endif
```




8. Save and close the file. The program should now compile successfully. Upload the program to the Arduino microcontroller. If some of the LEDs flash on and off, the program is working correctly. Now modify the code to work with 4 LEDs instead of 3 LEDs.
9. Now integrate the charlieplexing LEDs into the Simon Says circuit and modify the program so that the LEDs connected to pins 3, 5, and 10 work the way they did before (see Figure 4).
10. Note that there may be some small differences in the Simon Says game operation because with charlieplexing you can only have one LED on at a time.
11. Make sure you include in-line comments for all changes to the program. Make sure to make your code as modular as possible by using relevant functions, constants, and variables with logical names. Remember magic numbers are always a bad practice in programming.

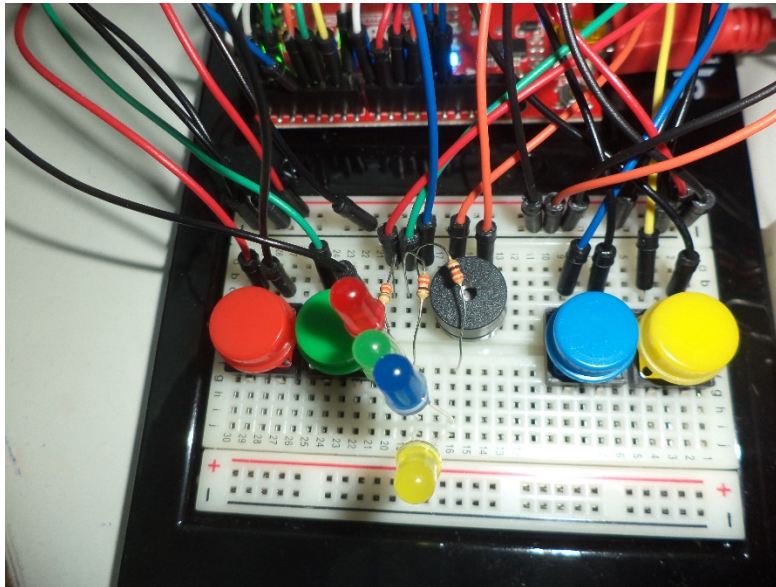


Figure 4: Simon Says Game with Charlieplexed Resistors

Part III – Simon Says with statistics

1. You will now have two additional free pins (13, 7) thus there are now 6 digital pins (0, 1, 7, 8, 11, 13) available to connect to the LCD mounted to a separate breadboard.
2. Mount the LCD and potentiometer to a separate breadboard and connect it to the digital pins on the microcontroller (see Figure 5).
3. Review the Arduino library for the LCD at the following link for more details about how to use it. <https://www.arduino.cc/en/Reference/LiquidCrystal>
4. Write a program to print the game statistics to the LCD screen. You should include the game number, length of each game sequence, number of games won and number of games lost since the reset button was pressed.
5. For the `MODE_MEMORY`, all of the statistics should work as described in step 4.
6. For the `MODE_BATTLE`, the player that starts the sequence wins if the opposite player gets out otherwise the player that starts the sequence loses. The LCD should show the statistics for the player that starts the sequence. The other game number and length of each game sequence should work as described in step 4.



7. For the `MODE_BEEGEES`, the only statistic to record is the game number. This value should represent the number of times the song was played.
8. Make sure you include in line comments for all changes to the program. Make sure to make your code as modular as possible by using relevant functions, constants, and variables with logical names. Remember magic numbers are always a bad practice in programming.

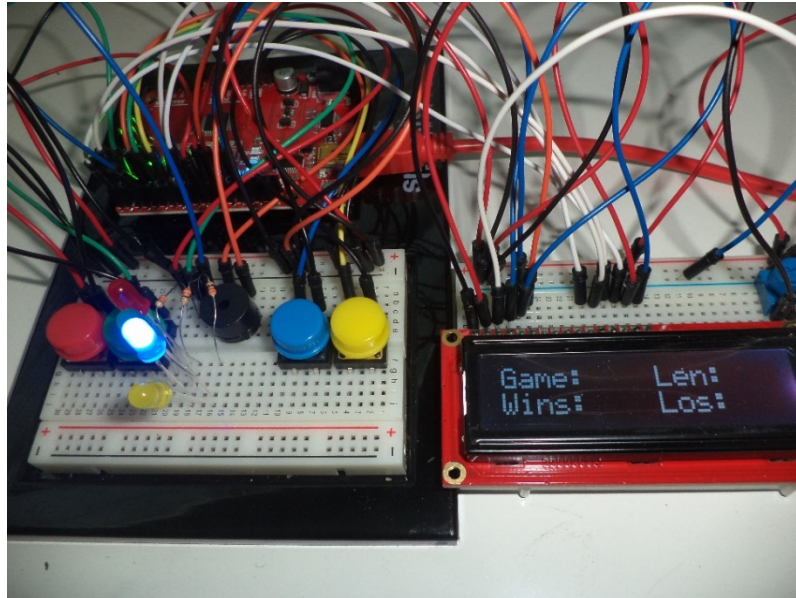


Figure 5: Simon Says Game with Statistics LCD

Part IV – Simon Says Rewind

1. In this part you will add a 4th game mode (`MODE_REWIND`) that is selected when the red button is held while the reset button is pressed.
2. You will also add the option to select the game mode (`MODE_MEMORY`) by holding the blue button as the reset buttons is pressed. There should now be 4 modes including the yellow button (`MODE_BEEGEES`) and green button (`MODE_BATTLE`).
3. Modify your program to make all four of the modes work with a button press. Make sure the light for the button that is pressed flashes and then is not illuminated briefly when the program starts similar to how the `MODE_BATTLE` and `MODE_BEEGEES` currently work.
4. In the end, all four of the buttons should cause the game to work in a different mode. Make sure to modify your program to make sure that happens. For example the `play_beegees()` function can be modified to play something else or at a minimum do something distinctive from the other three modes.
5. For `MODE_REWIND`, the game should work in the reverse of `MODE_MEMORY`. The player is required to repeat the light sequence in reverse.
6. Write a program to implement the `MODE_REWIND` portion of the game.
7. For the `MODE_REWIND`, all of the statistics should work as the same as they do for `MODE_MEMORY`.
8. Make sure you include in line comments for all changes to the program. Make sure to make your code as modular as possible by using relevant functions, constants, and variables with logical names. Remember magic numbers are always a bad practice in programming.



Submission Requirements:

Demonstration

You will demonstrate the final project in parts with specific milestones. Show your working circuit and program to the professor or lab assistant by the due dates.

Code

Upload your properly documented code to the Moodle DropBox by the due date. Properly documented code has a header comment with the name of the file, name of the author, date created and a description of how the code works. The header comment should also have a description of any key functions or variables in the program. There should be no magic numbers or numeric values not related to a constant or variable to make it clear what it does. There should also be in line comments to describe all major functions as well as lines of code as needed. In addition, as your code becomes larger and more complicated, you should create functions with logical names that are called from the main function (`void loop()`, `void setup()`) to make it readable and easier to follow. Someone unfamiliar with your project should be able to read your code and understand the functionality based upon your comments.

Memo

Create a weekly memo in Google Docs which meets the following requirements:

- ✓ Format
 - Begins with Date, To , From, Subject
 - Header includes team name, member names, faculty name, date and subject
 - Memo addressed to the course faculty member
 - Font no larger than 12 point font
 - Spacing no larger than double space
 - Written as a paragraph for the purpose and overview
 - Key decisions or progress updates can be a bulleted list
 - No longer than three pages of text
- ✓ Writing
 - Memo is organized in a logical order
 - Writing is direct, concise and to the point
 - Written in first person from team members
 - Correct grammar, no spelling errors
- ✓ Content
 - Starts with a statement of purpose
 - Summarizes key problems and accomplishments for the prior week
 - Provide major action items and list of issues or work to be done the next week
 - Discusses the strategy or pseudocode for implementing the project
 - Current progress on the project



- Current contribution and time completed on the project by each team member
- Discusses the tests and methods performed
- Clear statement of conclusions

Final Report

Create a final report on the final project that meets the following requirements:

- The document should have default Word settings with respect to font and margins
- All pages should be numbered
- All headings must be numbered, left-justified, bolded, and capitalized at the beginning of the section.
- All figures must have a number and title (caption) underneath (images of circuits, breadboards)
- All tables must have a number and title (caption) above it
- The cover page should have title, partner names, course number and title, date
- The report should be organized and free of spelling, grammatical, and punctuation errors.
- All graphics should be detailed and high quality
- The report should have the following sections:
 - a. Executive Summary
 - b. Introduction
 - c. Requirements
 - d. Research/Literature Review
 - e. Design Method
 - f. Verification and Results
 - g. Conclusions
 - h. Recommendations
 - i. References
 - j. Appendices (Code, Flowcharts)

Final Presentation

Give a final presentation on the project that meets the following requirements:

- Team Introduction with name and major of each member
- All members have a professional, business casual appearance
- Title slide has title of presentation with member names
- Quality graphics with minimum animations/transitions used in the slides
- Presentation has a problem statement and project overview
- Presentation has a clear method, verification, results, conclusions, and recommendations
- All members participate in the presentation
- The presentation is practiced with minimal reading of note cards or slides
- Team members show knowledge of the project



- Team recommends a final project for future courses
- Team handles questions wells

Grading Criteria:

The project score will be scored based upon the following rubric:

	100%	80%	60%	0%	Points
Demonstration	All parts of the project work and can be completely explained	Most parts of the project work and can be mostly explained	Very few parts of the project work and the functionality could not be explained	No demo completed	60
Code	The code includes header and in line comments with clear variables and modular functions as needed	The code is missing some necessary comments, variables and is not clear or modular as needed	The code has very few comments, variables, functions or modularity	No code submitted	80
Memo	Follows all of the guidelines with respect to the writing, content, and format	Follows most of the guidelines with respect to the writing, content, and format	Follows few of the guidelines with respect to the writing, content, and format	No memo submitted	60
Report	Follows all of the required guidelines for the final report	Follows most of the required guidelines for the final report	Follows few of the required guidelines for the final report	No final report submitted	100

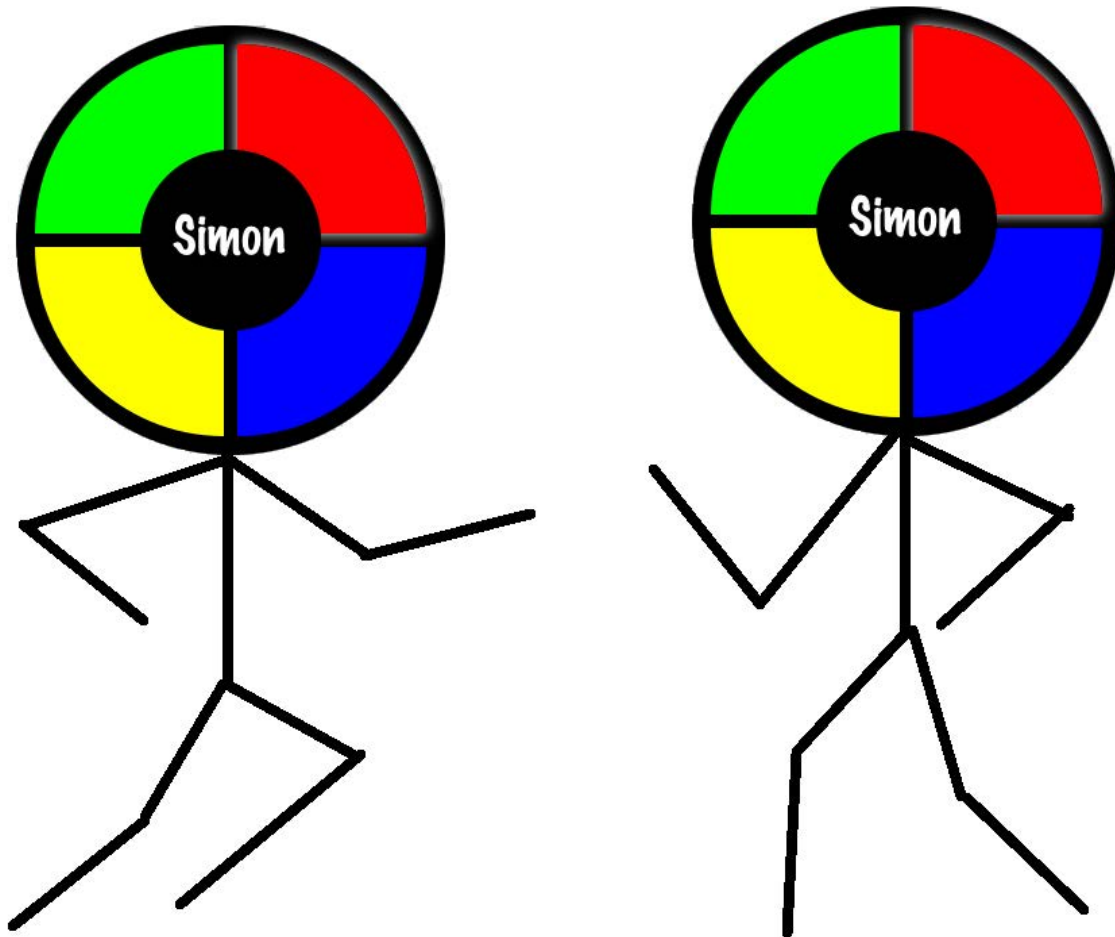


Presentation	Follows all of the required guidelines for the final presentation	Follows most of the required guidelines for the final presentation	Follows few of the required guidelines for the final presentation	No final presentation given	100
--------------	---	--	---	-----------------------------	-----

Please refer to the course schedule for the due date for all assignments. Assignments not submitted on time will incur a 20% late penalty per day. Assignments more than one day late will not be accepted.



Simon Says Spectacular



Competition



General Description

Each team will construct and program a Simon Says electronic memory game system by using the Arduino Microcontroller. Student teams will use their system to compete against other teams in the *Simon Says Spectacular*. Team scores will be based upon speed and accuracy as they complete 4 rounds of competition with increasing levels of difficulty. The four rounds of the competition are described below:

- Round 1 – MEMORY – Team members alternate completing as many memory games as possible in a two minute match.
- Round 2 – ROUND ROBIN – Team members alternate completing each step in a sequence of colors in as many games as possible in a two minute match.
- Round 3 – REWIND – It is similar to Round 1 except team members alternate completing as many memory games as possible in a two minute match, but they must remember the colors in reverse order.
- Round 4 – BATTLE – Team members alternate competing in a head to head match with a member of another team to complete as many memory games as possible in a two minute match.

Competition Details

- Thursday, November 5, 2015, 5:30 pm in HMU Kahn Room
- Attendance at the entire 2-hour event is **mandatory**. All members of the team must be present for the entire event. Any conflicts with attendance must be discussed with and approved by the course instructor at least one week before the event.
- **Competition results do not negatively impact your course grade.** However, the top 2 teams in the competition will earn extra credit on their final project grade.
- Due to the competition, class on Thursday and Friday of ninth week will be cancelled.

Awards

The top 2 teams in the competition will earn extra credit on their final project grade, a trophy and a certificate. The team with the most spirit during the event will also receive a trophy and certificate. The award categories are:

- “Simon Smasher” – Overall Tournament Winner
- “Simon Snuffer” – Overall Tournament Runner Up
- “Simon Sunshine” – Most spirited team with an identifiable theme during the event.

Timing and Seeding

- The practice and seeding will be in the class period before the competition.
- During the practice and seeding, each team will complete each round of the competition twice.
- The higher of the two scores for each round will be used to calculate an overall score which will be used to determine match order for the Thursday night competition (see Scoring section).
- There will be approximately 30 teams competing in the final project competition.
- There will be two 6 ft. competition tables with two teams per table. The two teams will compete on opposite ends of each of the tables.
- There will be 4 rounds with 8 matches per round with 4 teams per match.



- The number of games per match is variable based upon the speed and accuracy of the team members.
- Each team will get a one minute setup period followed by a two-minute match.
- Any team not ready to start the match at the end of the setup period will forfeit their turn.
- During the BATTLE round, each team will have two members competed in a two minute match where one member starts every sequence on their board and the opposing team member starts every sequence on their board.

Scoring

During the competition, teams will be scored based upon the following criteria. The accuracy score will be based upon the amount of steps completed in the sequence of a game before the player misses one or times out. A player loses the game when he or she selects a wrong color in the sequence or the game times out. The default setting for the entry time for selecting a color before the game times out is 3 seconds. The default number of steps for the player to win the game is 13. The accuracy score for a game will be calculated as 3^{m-1} where m is the number of correct steps in the sequence since the beginning of the game. The number of steps resets at the beginning of each game. The accuracy score for Game Z would be

$$\text{Game Z Accuracy Score} = 3^{m-1}$$

The speed score for the rounds is based upon the number of games played in a two minute match. Note that this value is a variable based upon the speed and accuracy of the players. The speed score will be calculated as 2^{n-1} where n is the number of complete games played since the beginning of the match. The number of games resets at the beginning of each match. The speed score for Match Y would be

$$\text{Match Y Speed Score} = 2^{n-1}$$

The length or number of steps must be at least 3 to earn the accuracy and speed score.

The match score for round Y will be the sum of the accuracy scores and speed score,

$$\text{Match Y Score} = \Sigma \text{Game Z Accuracy Score} + \text{Match Y Speed Score}$$

The team score for the competition will be the sum of the match score for each of the 4 rounds.

$$\text{Team Score} = \Sigma \text{Match Scores}$$

If the final bell rings and the team is in the middle of a game, the team will not earn points for that game. The team will only earn scores for games that are **completed** during the two minute match.

Round 4 – BATTLE

The scoring system will be a little different for round 4. If the player wins the game, the team gets the accuracy and speed score. If the player loses the game, the team does not earn any points for that game. Since this is a head to head battle, the two opposing team members will share and play on one board. The team member will start every sequence on their board.



For example, assume that there was a total of 10 games played in 2 minutes. If Team 1 won 4 games with steps of 4, 5, 6, and 13, their match 4 score would be

$$\text{Team 1 Match 4 Score} = (3^{4-1} + 3^{5-1} + 3^{6-1} + 3^{13-1}) + 2^{4-1} = 531,792 + 8 = 531,800$$

If Team 2 won 6 games with steps of 3, 5, 7, 13, 13, 2, their match 4 score would be

$$\text{Team 2 Match 4 Score} = (3^{3-1} + 3^{5-1} + 3^{7-1} + 3^{13-1} + 3^{13-1} + 0) + 2^{6-1} = 1,063,701 + 32 = 1,063,733$$

If the one minute bell to switch sides rings in the middle of a game, the team will immediately abandon the game to switch sides and no points will be assigned for that game. If the final bell rings and the team is in the middle of a game, neither team will get the points for that game. The teams will only earn points for games that are **completed** during the two one-minute matches.

Default Program/Game Settings

The system must abide by all of the following requirements:

- The game must have the MEMORY, BATTLE and REWIND modes and they must run without changing the current software on the system.
- It must be possible to change the game mode by pressing the reset button while holding down one of the 4 buttons.
- The game must be started with a button click. The player will click the button when the game clock sounds.
- The game must have an LCD that shows the game statistics (game number, sequence length, number of wins, number of losses).
- The BATTLE mode must show the wins and losses on the LCD for the player that started the sequence.
- There must be a distinctive sound and light sequence when a game is won.
- There must be a distinctive sound and light sequence when a game is lost.
- The `ROUNDS_TO_WIN` constant must be set to 13 in the software.
- The `ENTRY_TIME_LIMIT` constant must be set to 3000 in the software.
- The `playback_time` constant must be set to 150 in the software.

Rules and Penalties

All teams must abide by the following list of competition rules and penalties:

- Teams cannot add any additional parts to the Simon Says circuit that would give their team an unfair advantage in the competition. The system must abide by all of the requirements described in the final project assignment.
- Teams cannot make any additional changes to the Simon Says control software that would give their team an unfair advantage in the competition. The system must abide by all of the requirements described in the final project assignment.
- Any team found to be using a system that does not abide by these rules will be immediately disqualified from the competition and there may also be a negative effect on their final project grade.



- The only items allowed on the competition surface during a round are the Arduino microcontroller, 2 breadboards with circuits and one laptop computer. There will be a 5 point penalty per occurrence for extraneous parts on the field.
- Team members cannot have hands or arms over the competition surface during a round. The only exception will be the team member who is currently controlling the game. There will be a 5 point penalty per occurrence for team member body parts crossing the plane of the field.
- Teams must decide the competing order of the members before the competition begins. This order must be used for all rounds and cannot be changed during the competition. Any team that goes out of order during a round will forfeit the score earned.
- Only one team member may touch the game board at a time. There will be a 5 point penalty for each touch by another team member.
- During the MEMORY and REWIND rounds, the team members will change after each game based upon the pre-determined line up.
- During the ROUND ROBIN round, the team members will change after each successful step in the sequence based upon the pre-determined line up.
- During the BATTLE round, one member from two opposing teams will alternate completing a game on the same game board. Once the game is complete, the next member from each team compete on the same game board. The game board will be used for the team that starts the sequence. After the first minute of the two minute match, the teams will switch to the opposing teams board and a member of that team will start the remaining sequences.
- Students shall in no way interfere with another team's project before, after, or during the competition.
- During the competition, students must be respectful and display good sportsmanship at all times. Any student or team found to be exhibiting inappropriate behavior will be immediately disqualified from the competition and there may also be a detrimental effect on their final project grade.

Judging

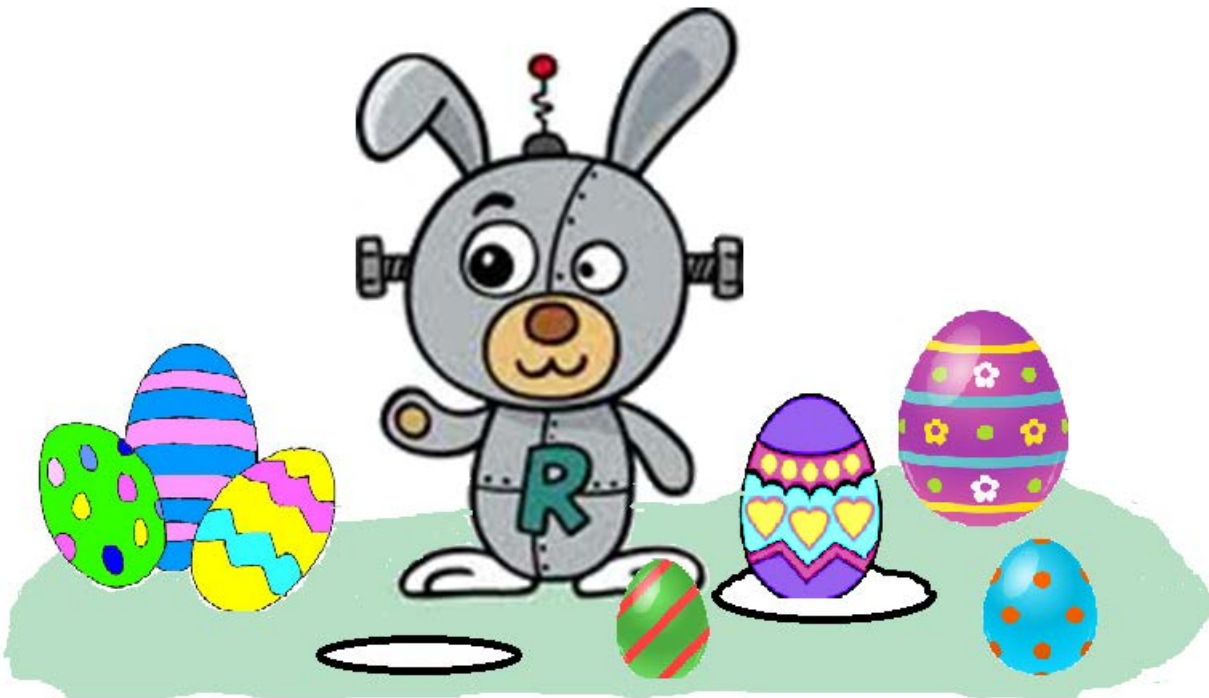
- Judges for the competition will be ECE department faculty and staff.
- Course faculty members will be the master judges.
- Judges will serve as the officials and scorekeepers. They are responsible for making sure that all teams abide by the rules of the competition, scoring each round, assigning penalties and giving score sheets to the official scorekeeper.
- The decisions of the judges shall be final. Decisions may be challenged for factual or interpretive errors but the officials get the final decision regarding the validity of the challenges. In order to challenge a rule interpretation, one member of the team must approach a master judge to request clarification within 5 minutes of the decision.
- All teams must ask the course instructor for any rule interpretation in a timely fashion. All requests for rules clarification must be made by the end of the class period preceding the competition. Requests for rule interpretations once the competition has commenced will be denied.

RoBunny Rumble

Submitted by: Carlotta Berry
Rose-Hulman Institute of Technology
berry123@rose-hulman.edu



RoBunny Rumble



Winter 2015

Version 2.0



Final Project

RoBunny Rumble

Learning Objective:

Design a system to enable a mobile robot to collect and deposit plastic eggs in rabbit holes.

Learning Outcomes:

At the conclusion of this project, the student team should be able to

- Program a robot to use a gripper to retrieve eggs and deposit them in various hole sizes
- Program a robot to navigate autonomously in a cluttered environment
- Attach peripherals to a mobile robot and create a program to wirelessly control it
- Attach sensors to a mobile robot to create a program to detect walls, floors, eggs, and/or holes
- Create properly commented code with modular functions to achieve all of the objectives of the final project
- Demonstrate the working project and adequately explain the functionality
- Completely document what was learned and achieved through weekly memos, a final report and presentation

Parts:

- Arduino Shield Bot with Gripper
- Arduino Uno Microcontroller
- 2 – infrared sensor
- 2 – sonar sensor
- 2 – 5 x 10 NeoPixel panel
- 1 – infrared receiver
- 1 – breadboard
- 1 – LCD
- Battery charger
- 2-nRF24L01 Wireless Transceiver
- Joystick
- Remote Control
- PlayStation Controller & Bluetooth Dongle
- Pixy Camera or CMUCam (optional, limited availability)

**Preparation:**

In order to prepare for the final project, review the following links as well as any supplemental materials provided for this course including the mini-projects and midterm projects. The team should also conduct any online research necessary to help complete the final project. Make sure to take good notes and keep good records of any resources used because this will be part of the final report literature review.

- Arduino Website www.arduino.cc
- Shield Bot <http://learn.parallax.com/ShieldRobot>
- IR Remote <https://www.arduino.cc/en/Tutorial/RobotRemoteControl>
- Sonar Sensor <https://www.arduino.cc/en/Tutorial/Ping>
<http://playground.arduino.cc/Code/NewPing>
<http://www.instructables.com/id/Easy-ultrasonic-4-pin-sensor-monitoring-hc-sr04/>
- Sharp IR Sensor <http://playground.arduino.cc/Main/SharpIR>
- Linearize Sharp IR Sensor <https://acroname.com/articles/linearizing-sharp-ranger-data>,
<http://projectsfromtech.blogspot.com/2014/10/linearizing-sharp-ir-ranger-2yoa21-with.html>
- PlayStation Controller Library <http://playground.arduino.cc/Main/PSXLibrary>
<http://playground.arduino.cc/Main/AnalogPSXLibrary>
https://www.youtube.com/watch?feature=player_embedded&v=xlupRVF_6W8
- CMUcam5 Pixy Camera
[http://cmucam.org/projects/cmucam5/wiki/Hooking_up_Pixy_to_a_Microcontroller_\(like_an_Arduino\)](http://cmucam.org/projects/cmucam5/wiki/Hooking_up_Pixy_to_a_Microcontroller_(like_an_Arduino))

Procedure:

Phase I – Tetris Game Demonstration

- Create a program on two 10 x 50 NeoPixel panels to play one game of Tetris for one minute or until the team member loses
- Show statistics on how many rows were cleared and how long the game was played on an LCD or serial monitor
- Play musical soundtrack for the game
- A program flowchart, state diagram or pseudocode must be included in the final report

Phase II – Remote Control Demonstration

- Attach at least 2 of the following device setups to the robot (PlayStation Controller & Bluetooth Dongle, Joystick & 2 Transceivers, Remote Control & Infrared Receiver).
- Use each device setup to wirelessly drive the robot, retrieve and deposit plastic eggs in the various rabbit holes



- Use a decision matrix to determine which method the team will use to remotely control the robot for the final competition. The decision matrix must be included in the project final report
- A program flowchart, state diagram or pseudocode must be included in the final report

Phase III – Autonomous Control Demonstration

- Attach at least 2 sensors to the robot. You should try a variety of mounting locations and orientations.
- Perform tests on the sensor data to relate distance information to analog data read by the program. Use an equation to linearize the data and use the expression in your code. You must include an Excel table and graph of the sensor distance and analog data in the appendix of the final report
- Use the sensors to perform tests on whether the robot can accurately detect eggs, holes, and walls.
- Use a decision matrix to determine which position, orientation, and sensor the team will use for the robot's autonomous control
- Write a program so that the robot operates for 30 seconds in autonomous mode to gather eggs and deliver them to the rabbit holes. You must use the sensor data as part of the program.
- Submit a program flowchart, state diagram or pseudocode to be included in the final report

Submission Requirements:

Demonstration

You will demonstrate the final project in parts with specific milestones. You will show your working project to the professor or lab assistant by the due dates.

Code

Upload your properly documented code to the Moodle DropBox by the due date. Properly documented code has a header comment with the name of the file, name of the author, date created and a description of how the code works. The header comment should also have a description of any key functions or variables in the program. There should be no magic numbers or numeric values not related to a constant or variable to make it clear what it does. There should also be in line comments to describe all major functions as well as lines of code as needed. In addition, as your code becomes larger and more complicated, you should create functions with logical names that are called from the main function (`void loop()`, `void setup()`) to make it readable and easier to follow. Someone unfamiliar with your project should be able to read your code and understand the functionality based upon your comments.

Memo

Create a weekly memo in Google Docs which meets the following requirements:



- ✓ Format
 - Begins with Date, To , From, Subject
 - Header includes team name, member names, faculty name, date and subject
 - Memo addressed to the course faculty member
 - Font no larger than 12 point font
 - Spacing no larger than double space
 - Written as a paragraph for the purpose and overview
 - Key decisions or progress updates can be a bulleted list
 - No longer than three pages of text
- ✓ Writing
 - Memo is organized in a logical order
 - Writing is direct, concise and to the point
 - Written in first person from team members
 - Correct grammar, no spelling errors
- ✓ Content
 - Starts with a statement of purpose
 - Summarizes key problems and accomplishments for the prior week
 - Provide major action items and list of issues or work to be done the next week
 - Discusses the strategy or pseudocode for implementing the project
 - Current progress on the project
 - Current contribution and time completed on the project by each team member
 - Discusses the tests and methods performed
 - Clear statement of conclusions

Final Report

Create a final report on the final project that meets the following requirements:

- The document should have default Word settings with respect to font and margins
- All pages should be numbered
- All headings must be numbered, left-justified, bolded, and capitalized at the beginning of the section.
- All figures must have a number and title (caption) underneath (images of circuits, breadboards)
- All tables must have a number and title (caption) above it
- The cover page should have title, partner names, course number and title, date
- The report should be organized and free of spelling, grammatical, and punctuation errors.
- All graphics should be detailed and high quality
- The report should have the following sections:
 - a. Executive Summary



- b. Introduction
- c. Requirements
- d. Research/Literature Review
- e. Design Method
- f. Verification and Results
- g. Conclusions
- h. Recommendations
- i. References
- j. Appendices (Code, Flowcharts)

Final Presentation

Give a final presentation on the project that meets the following requirements:

- Team Introduction with name and major of each member
- All members have a professional, business casual appearance
- Title slide has title of presentation with member names
- Quality graphics with minimum animations/transitions used in the slides
- Presentation has a problem statement and project overview
- Presentation has a clear method, verification, results, conclusions, and recommendations
- All members participate in the presentation
- The presentation is practiced with minimal reading of note cards or slides
- Team members show knowledge of the project
- Team recommends a final project for future courses
- Team handles questions wells

Grading Criteria:

The project score will be scored based upon the following rubric:

Score	5	3	1	0
Demonstration	All parts of the project work and can be completely explained	Most parts of the project work and can be mostly explained	Very few parts of the project work and the functionality could not be explained	No demo completed
Code	The code includes header and in line comments with clear variables and modular	The code is missing some necessary comments, variables and is not clear or	The code has very few comments, variables, functions or modularity	No code submitted



	functions as needed	modular as needed		
Memo	Follows all of the guidelines with respect to the writing, content, and format	Follows most of the guidelines with respect to the writing, content, and format	Follows few of the guidelines with respect to the writing, content, and format	No memo submitted
Report	Follows all of the required guidelines for the final report	Follows most of the required guidelines for the final report	Follows few of the required guidelines for the final report	No final report submitted
Presentation	Follows all of the required guidelines for the final presentation	Follows most of the required guidelines for the final presentation	Follows few of the required guidelines for the final presentation	No final presentation given

Please refer to the course schedule for the due date for all assignments. Assignments not submitted on time will incur a 20% late penalty per day. Assignments more than one day late will not be accepted.

Final Project Competition

RoBunny Rumble

The final project competition, **RoBunny Rumble**, will be held at 5:30 pm on Thursday, February, 11, 2016 in the Hulman Memorial Union Kahn Rooms. Each team will show off their design to their peers, administrators, and faculty. Any conflicts must be discussed with, and approved by, your instructor before the dry run. Due to the competition, the next day of class will be cancelled.

Competition results do not impact your course grade; however attendance at the competition is mandatory. Any individual who misses any part of the final project competition without prior permission will receive a deduction in their final project grade.

Project Award Categories:

- RoBunny Renegade - Overall Tournament Winner
- RoBunny Rebel - Overall Tournament Runner Up
- Robunny Razzle Dazzle - Most Spirited Team

Field and Game Pieces:

All teams are required to use the same mobile robot shown in Figure 1. All teams must use the sensors and peripherals provided by the instructor. Teams are not allowed to add any of their



own custom parts unless they are provided to all of the competition teams. The mobile robot must be able to pick up the five eggs shown in Figure 2. All of the eggs are approximately 2 ¾" x 1 ¼". Table 1 provides a list of the number of eggs and the score for each type.

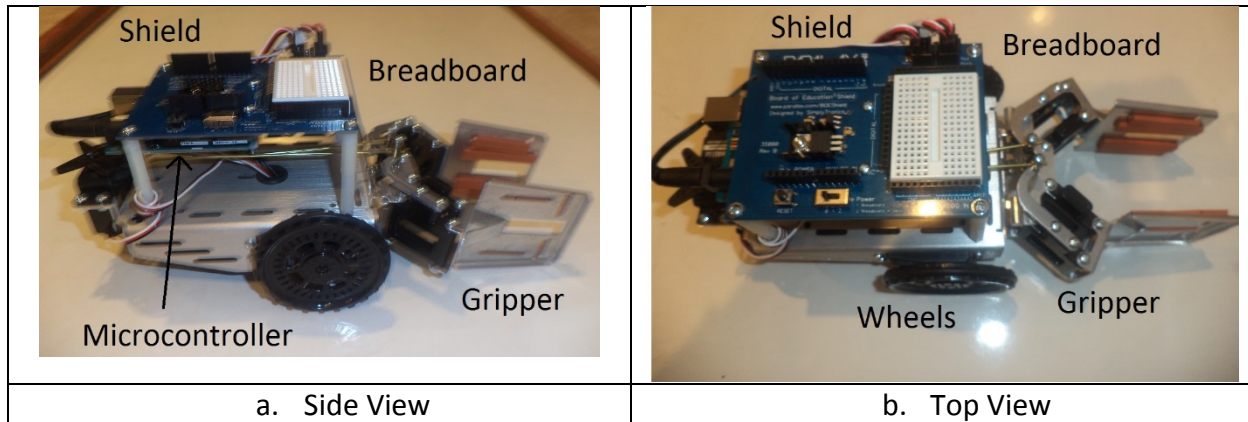


Figure 1: Shield Bot with Gripper

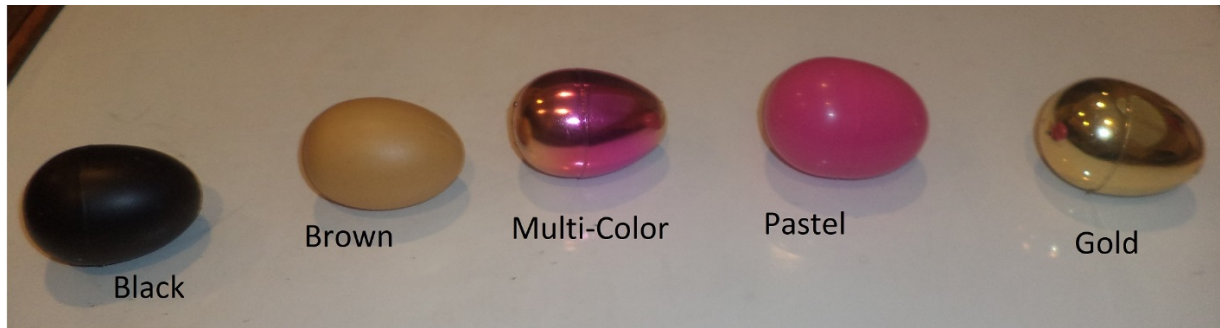


Figure 2: Egg Types

Table 1: Egg Inventory

Type	Number	Points
Black	12	-10 pts
Brown	12	+5 pts
Multi-Color	12	+10 pts
Pastel	10	+15 pts
Gold	6	+20 pts
TOTAL	52	

The mobile robot must be able to operate on the 6' x 6' competition surface as shown in Figure 3. The competition surface has 4 holes with increasing diameters to deposit eggs. There will be 2 tables available during the competition and 26 eggs on each table. The different types of eggs will be evenly divided between each table. Table 2 provides a summary of the score for the hole diameters. The competition surface and eggs will be provided for practice during the project design phase. The competition surface and eggs shall not be modified or damaged in any way



during practice or competition. Figure 4 at the end of this document provides a detailed description of the competition table with dimensions.

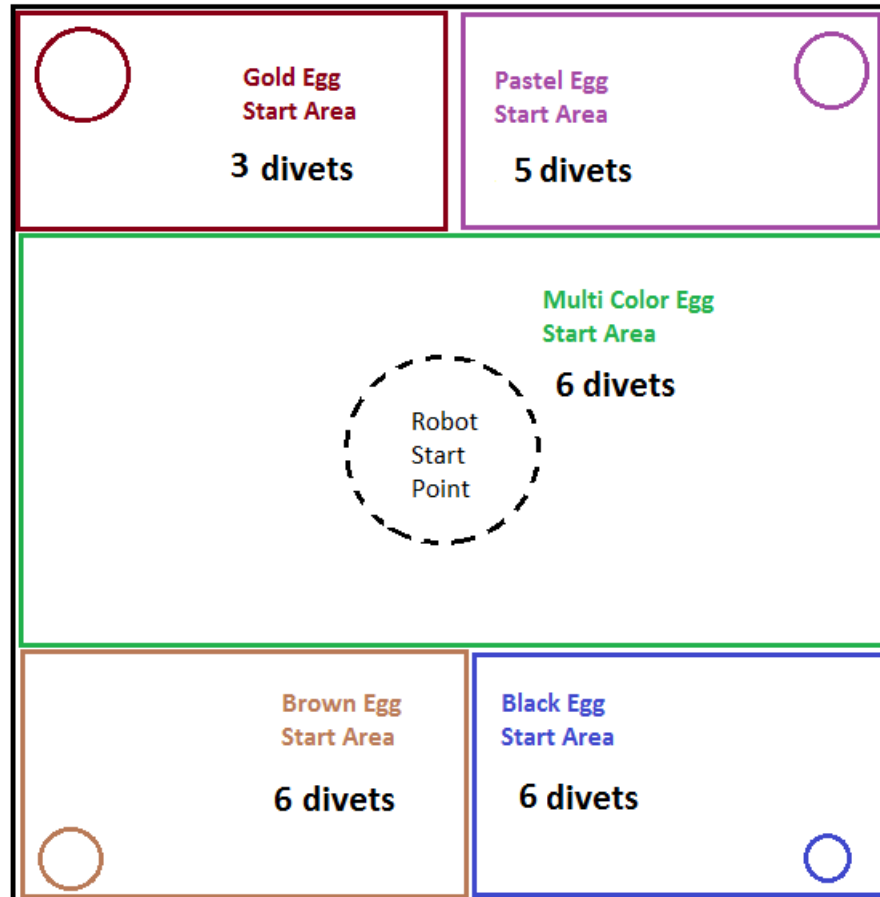


Figure 3: Competition Table

Table 2: Rabbit Hole Points

Circle Diameter	Points
2 ¼"	20 pts
2 ¾"	15 pts
3 ¼"	10 pts
4"	5 pts
TOTAL	



Timing:

Each team will compete in 5 rounds including the dry run. The dry run (round 1) will be used for seeding and will be completed during the final class before the competition. During the dry run each team will demonstrate their final complete working project 3 times and the highest of the 3 scores will be used for seeding during the Thursday night event. The competition order for Thursday night will be determined by the seeding.

There will be 3 matches in each round except for the dry run. Each match will have one team competing on each table. The timing for the match will include a 30 second autonomous period and 150 second teleoperation period. There will be a one minute setup period before each match. The scores for each round except for the dry run will be cumulative. After round 5, the overall tournament winner and runner up will be selected.

Competition Rules:

1. Each match will consist of a 1 minute setup period following by a 30 second autonomous period and a 2 ½ minute teleoperation period. The match will end 3 minutes after the end of the setup period.
2. During the 1 minute setup period, the robot may be programmed or reprogrammed. However, parts may not be added to or removed from the robot once the elimination rounds begin.
3. If parts become detached from the robot, the team has 5 seconds to remove them from the competition surface. Detached parts may be reattached during the subsequent setup periods or between matches.
4. Each team will have 26 eggs on the table at the beginning of the match. There will be a penalty for any damage to the playing surface or eggs during the match.
5. Teams will drive the robot from a driver table adjacent to the competition field.

Scoring Rules:

1. A match score will be based upon the number of eggs delivered to each of the 4 rabbit holes as well as the point value of the egg.
2. The team will earn double points for the first 30 seconds when the robot operates in autonomous mode. If the robot starts the autonomous mode holding an egg there will be a 100 point deduction in the match score.
3. The score will only be based upon eggs that have been delivered to the rabbit hole before the bell rings. Once an egg leaves the competition surface, it is out of play for the match.
4. The robot must start in the dead center of the table facing any direction.
5. This is not a head to head match, all teams will be ranked at the end of each round and the overall tournament score is cumulative based upon rounds 2 through 5. Round 1 score is used to determine the match order for Round 2.
6. In the event of a tie, the teams will compete in a head to head 1 minute match and the team with the most eggs delivered at the end will be the overall tournament winner.
7. Each match will be scored based upon the following formula.



$$\text{MATCH SCORE} = (\text{Egg Points} \times \text{Rabbit Hole Points}) - (\text{Penalty Points})$$

Penalties:

- A team will incur a 100 point penalty for each robot part, sensor or peripheral that is not removed from the field within 5 seconds.
- If the robot gets stuck in a hole and cannot extract itself, the team will incur a 100 point penalty for each manual extraction.
- A team will incur a 100 point penalty for each game piece or part of the playing field that is damaged during the match.
- A team will incur a 100 point penalty each time a team member's body part crosses the plane of the competition field during gameplay. This penalty includes manually extracting an egg stuck in the robot and placing it in front of the robot's gripper.
- A team will incur a 100 point penalty each time the robot driver leaves the driver's box.

Judging:

1. Judges shall be provided by the Rose-Hulman ECE department and other faculty.
2. Judges will serve as officials and scorekeepers. Therefore, judges will be responsible for insuring that the teams abide by the rules of competition, scoring each round, assigning penalties and giving the score sheet to the official scorekeeper. Officials, announcers, score keepers along with team members will be responsible for field reset.
3. The decision of the officials shall be final. Decisions may be challenged for factual or interpretive errors; but, the officials have the final decision on the validity of challenges. In order to challenge a rule interpretation, one member of the team should approach the head judges to request clarification within 5 minutes of the decision. The head judges are the course instructors.
4. Teams shall ask the instructor for rule interpretations in a timely fashion. Requests for rule interpretations made after 5:00 p.m. on February 11, 2016 will be rejected.

Other Requirements:

1. All students shall participate with their team in the competition.
2. All students shall participate with their team in the design process for this project.
3. All team members shall participate equally in the project work.
4. All team members shall receive a grade on the project based upon the entire team's work as well as their individual work.
5. Students shall respect the creativity of others.
6. Students shall in no way interfere with another team's project work or interfere with another team during the competition.

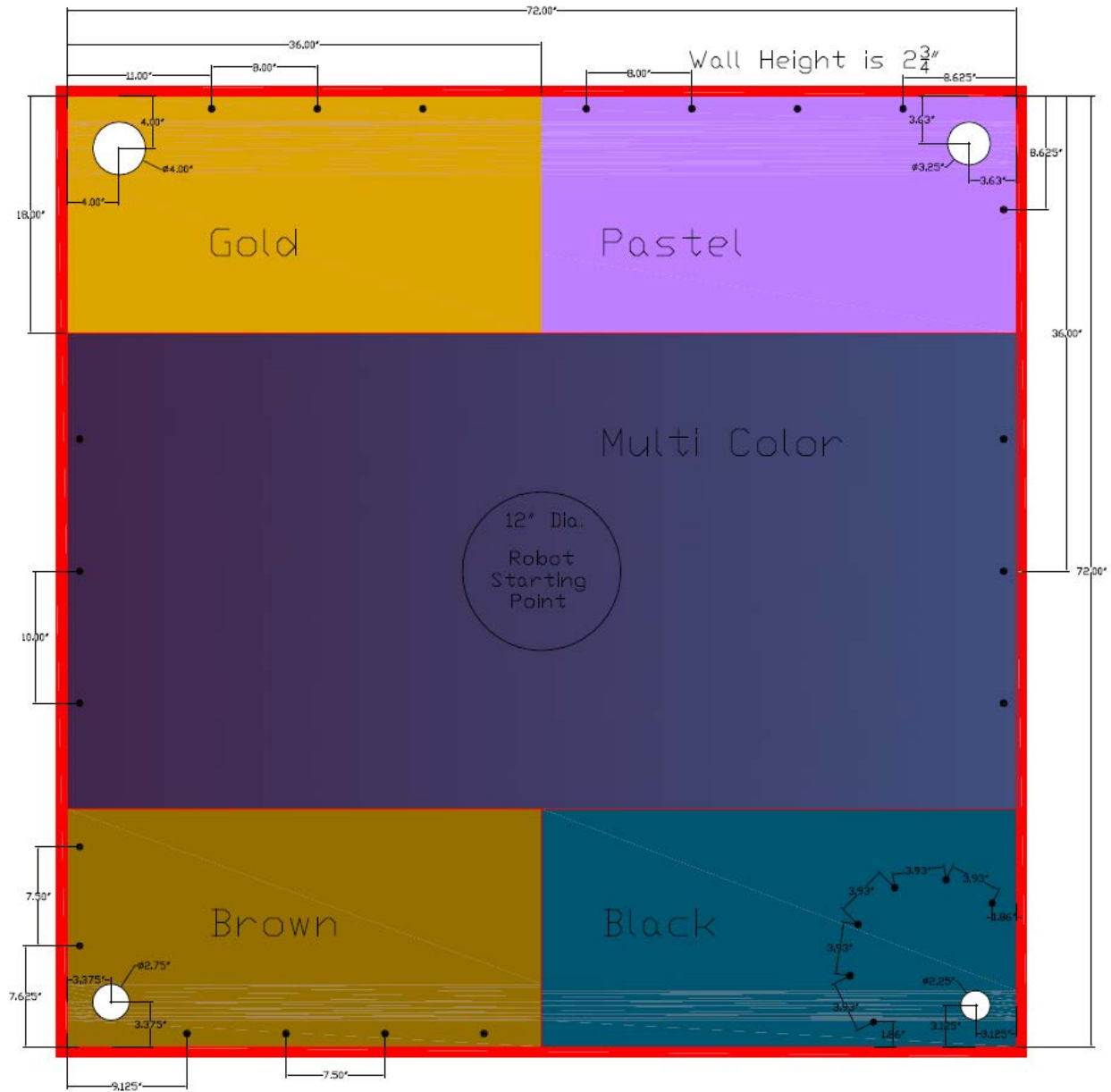


Figure 4: Competition Table with Dimensions

NeoPixel Group Project

Submitted by: Carlotta Berry
Rose-Hulman Institute of Technology
berry123@rose-hulman.edu

**NeoPixel Group Assignment****Purpose:**

- The primary purpose of this assignment is to enable the group members to develop teamwork skills by learning to work together on a common purpose while delegating tasks, holding each other mutually accountable, using effective decision making, and documenting their progress.
- The team will also learn how to use some basic hardware, the NeoPixel, which they will use as a component of the final project.
- The team will also write various programs to create a music light show that will be demonstrated in class.

Materials:

- 2 – 5 x 10 NeoPixels
- 1 – pushbutton
- 1 – buzzer
- 1 – joystick

Preparation:

1. Review the information at the following link to learn about the magic of NeoPixels.
<https://learn.adafruit.com/adafruit-neopixel-uberguide/overview>
2. Download the Adafruit_NeoPixel library from the following link by clicking Download ZIP.
<https://learn.adafruit.com/adafruit-neopixel-uberguide/arduino-library>
3. Install the library by unzipping it to [Documents\My Documents\Arduino\Libraries](#)
4. Review the image on the link in step 2 for help on how to wire the NeoPixel to your Arduino Uno microcontroller. Note that you don't need an external power supply, you can use the +5V and GND pins on the microcontroller. The 5V lead on the NeoPixel is marked. There is also a female header pin for chaining multiple NeoPixels together. Connect the middle pin on the NeoPixel to digital IO pin 6.
5. Wire an active-low pushbutton to the microcontroller. Active-low means that the pin where the pushbutton is connected reads a HIGH value until it is pressed and then it reads a LOW value. This button uses an internal pull up resistor so you don't have to wire it in series with a 10 kΩ resistor. Connect one pin to digital IO pin 2 and connect the other pin to ground.
6. Open the "buttoncycler.ino" example program. Review the code and comments and make sure you fully understand how the program works. Change the PIXEL_COUNT to 50. Run the program and examine the results.



7. Open the “simple.ino” example program. Review the code and comments and make sure you fully understand how the program works. Change NUMPIXELS to 50. Run the program and examine the results.
8. Open the “strandtest.imo” example program. Review the code and comments and make sure you fully understand how the program works. Add a global constant NUMPIXELS and set it to 100. Change the following line as shown below:

```
Adafruit_NeoPixel strip = Adafruit_NeoPixel(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
```
9. Daisy chain the two NeoPixels together by inserting the female header of one into the male header pin of the other. **MAKE SURE THAT THE TWO 5V PINS ARE CONNECTED!**
10. Run the program and examine the results.

Procedure:


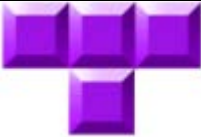





Since this assignment is primarily focused on the team members learning the art of collaboration, it is meant to be challenging. The team should keep this in mind as they attempt to achieve the following 5 challenges. The team will earn one point for each challenge that is successfully completed. Each level of the challenge has an increasing level of difficulty. The final challenge should look similar to the Tetris game play. You can see an example of the game at the following link: <https://www.youtube.com/watch?v=QQ5U-rN7Veg>. You can play a round of Tetris at the following link: <http://tetris.com/play-tetris/>.

As usual the code should be modular with logical function names. Each challenge should have a different function and the program may have more functions as needed. It is important to get all of the subsystems working individually in order to be more successful in integrating them at the next level.

Challenge 1 - Tetrimino (Tetris Pieces)

1. Create the 7 pieces shown in Table 1. There should be 7 distinct colors but they do not necessarily have to match those shown on the table.
2. Write a program so that they randomly appear based upon button presses.

Table 1: Tetrimino Pieces

			
a. Yellow block	b. Purple T	c. Blue L	d. Orange L
			
e. Green Z	f. Red Z	g. Cyan stick	

Challenge 2 - Rotation

1. Attach a joystick to the analog and digital pins on the Arduino Uno microcontroller.
2. Write a program that randomly displays the 7 Tetrimino based upon the button press.
3. Use the up motion on the joystick to incrementally rotate the Tetrimino by 90 degrees clockwise on the display.

Challenge 3 - Movement

1. Attach both NeoPixels front to back using the male and female header pins so that the size is now 100 or 5 x 20 NeoPixels.
2. Modify the program so that the pushbutton starts or pauses the program. When the button is pressed and the program starts one of the 7 Tetrimino should randomly appear in the top center of the display.
3. The piece should then fall to the bottom of the screen at a regular speed.
4. If the up arrow on the joystick is pressed, the piece should rotate as it falls and lands on the bottom of the screen.

Challenge 4 – Stacking

1. Modify the program so that when the Tetrimino fall from the top to the bottom they either stop at the bottom or stack on any pieces already there.
2. If there are gaps in the row, the piece should fit in the gap or lay on the row or create gaps as required.

Challenge 5 – Music and Gameplay

1. Modify the program to create the last elements of the game by clearing a row when it is successfully completed.
2. Also modify the program to keep statistics such as how many rows were cleared and how long the user played the game. This information can be displayed on the serial monitor during and at the conclusion of the game.
3. Attach the buzzer to a digital pin on the Arduino Uno microcontroller.
4. Use the buzzer to create a musical soundtrack for the game with distinctive sounds for clearing a row, winning a game, and losing a game.
5. The player should win the game if they successfully play the game for one minute. The player should lose the game if the Tetrimino stack reaches the top of the 5 x 20 grid.
6. Here is some inspiration:

<https://www.youtube.com/watch?v=Mn1iOMAMRw8&feature=youtu.be>



Rules, Documentation and Submission:

1. Since this is an extra credit assignment, the team members should not seek assistance from any other individuals including the instructor. There will be a negative penalty if the assignment is not attempted.
2. You must have meeting minutes for every team meeting. There is no journal entry but the meeting minutes should include details such as date, time, location, member names, activities, progress made, goals for next meeting, delegated tasks (action items), what was learned, etc. [see sample on Moodle and Google Drive]
3. You must demonstrate the assignment during class on the due date.
4. You must upload the properly commented code to the Moodle DropBox by midnight on the due date.