**CPEN 291   
Mini-Project Report**

**A. Group info**

Lab section: L2A (*L2A or L2B)* Group #: A-G4 (*Example: A-G1*)   
Group’s Lab Bench #s: 9 and 4 (*Example: 1 and 2*)   
Student names:

|  |  |
| --- | --- |
| Andrea Shao | Brielle Law |
| Matthew Chow | Matthew Stefansson |
| Matthew Yen | Sebastian Gonzalez |

All the underlined text in this template are for your info. You should remove any text that is underlined before submission to Canvas.

Please keep the format of the report as is (e.g. do not omit any section, or change the font size or margins).

In Sections B and C of the report:

* Explain the design and implementation procedures, and thoroughly provide documentation for the circuitry and software. During the project, you must have selected any method over another for some steps, describe the alternative (e.g. the second best) you considered. Include block diagrams or drawing to identify the main components and their interactions.
* You may include code segments in this part of the report only whenever needed for the explanations of the software design and approach. Your code must include comment statements, so do not repeat what is already included in the comment statements. As usual you will need to submit the complete code file separately, and also to include the complete code as an appendix to this report. The code must be readable in the first place and include sufficient comments (per code segment and per line, when needed) for documentation.

**B. Technical documentation for the main functionality**

Fully document your design and implementation for the main functionality here. In particular explain:

* What the six dance moves are, and how they are different
  + Dance 1: The robot starts facing forward. It then raises its left and right foot in sync . It then rotates its left and right foot in a half circle really fast and then comes back to the starting position slowly.
  + Dance 2: The robot starts facing forward. It taps its left foot a few times and then shakes its right foot. It then taps its right foot a few times and then shakes its left foot.
  + Dance 3: The robot starts facing forward. It then quickly turns its feet in towards the center of its body, by doing this it leaves the ground jumping up. It then quickly shifts its feet back to the starting position. This dance move looks like the bottom half of a jumping jack.
  + Dance 4: I think its two similar to 1
  + Dance 5: The robot starts facing forward. It starts by tapping its right foot a few times. It then taps its left foot a few times as well.
  + Dance 6: The robot starts facing forward. It then shifts the bottom motors to make the robot look like its standing on its tippy toes. It then shifts its knees to make it look like it is pointing its toes to the center of its body. It then returns to its original position.
* How you have implemented each of the six robot moves

Each dance move was first worked on in a separate file called dance1, dance2, dance3 etc. In this separate file the dance move was thoroughly tested and worked out. When the dance move was finished it was integrated into the main file. Where we added the threading with the piezo buzzer and ultrasonic sensor.

* What info the LCD displays and how it has been mounted/attached for optimal viewing

The LCD display was mounted on the breadboard which sits inside the robot’s shell. At the beginning it shows a smiley face that says, “hello I’m bub”. Then it changes to “danceX” where X is the dance it is currently preforming.

**C. Technical documentation for the additional functionality**

Fully document your design and implementation for the additional functionality here. Explain:

* What the additional functionalities are
  + We added the piezo buzzer to play a song while our robot is dancing
  + We added the HC-SR04 ultrasonic sensor to play a song when it is less than 4cm from an object and stop playing the song when it is more than 25cm from an object.
* Include the list of the additional components you used
  + 1 Piezo Buzzer
  + 1 HC-SR04 ultrasonic sensor
* The hardware implementation
  + The positive end of the buzzer is plug into port A0
  + The trigger is plug into D3 and the echo is plugged into D4
  + Both the buzzer and ultrasonic sensor are plugged into ground and the ultrasonic sensor is plugged into the power
* The software implementation
  + To implement this in each dance move we would make it seem like there were two thread running but in reality, there was only one. We would switch between moving the robot, checking the ultrasonic sensor and playing the buzzer. To play the buzzer we had two lists one for the note and another one for how long the note plays for. Every time a note was played the index was incremented. At the beginning of each dance move we check if the whole song has been played and if it was the song restarts and index was set to zero. For the ultrasonic sensor we would check if the distance was less than 4cm and if it was it would play a song. When playing the buzzer and using the ultrasonic sensor we would do a small part of the dance move, play the buzzer and check the ultrasonic sensor.

**D. Test and evaluations**

While doing this project our goal was to have a bug free robot and that is why testing was so important. To make sure that our integration was as easy to debug as possible every individual component was thoroughly tested before it was integrated. For example, when we were putting the robot, we tested each servo motor with the previously written raspberry pi code before putting the servo on the robot. By doing this we found out that a motor was not working.

With respect to testing with the code we would write a piece of code individually and then extensively test the code. By doing this it was easy to identify what part of the robot had a problem. After testing the code individually, we would integrate that part of the code into the main robot. By doing this it reduced the number of bugs in our code and didn’t leave integration until the very last minute.

The very first problem we had was with the power being supplied to the motors. We could only run three motors at a time. At first we tried a few capacitors to fix the noise but that did not solve it. To fix this problem we looked more closely at the documentation of the motors and the itsbitsy and realized we were supposed to use the usb power source not the 3V power source.

Explain your evaluation and testing procedures for hardware and software. Please demonstrate systematic testing, debugging and continuous integration. Include the problems you have encountered and how you resolve them, as well as best practices you have incorporated.

**E. Conclusions and Reflections**

Reflect and conclude on the lessons, tricks or interesting concepts you have learned during the project.

Also reflect on other aspects such as team work, project management, time management, ...

Our whole group loved doing this project. We learned a lot of new things. We learned the importance of teamwork and project management.

**F. References and bibliography**

Provide any relevant references.

Also include the list and description of the files submitted for this lab (including code and Fritzing breadboard schematics)

Include the following appendixes. There is no page limit for the appendixes.

**Appendix A – Robot pictures**

Include a few pictures of your robot here. The pictures should clearly show the robot as a whole, as well as all electronics, wiring and parts. Include photos taken from the top, and from the sides. Show the location/installation of circuits and components as clearly as possible.

**Appendix B – Code**

**All the code is clearly commented. In github you can see that all the different parts of the robot were initially tested in their own files. Whether that was a dance, the buzzer, the integration between the buzzer and dancing or the ultrasonic sensor.**

Include the complete CircuitPython code with comment statements. This code must be the same code as the files you demo and submit. Clearly identify the portion of the code for the main functionality and the Additional functionality.

The code must be readable, with indentation, syntax highlighting (that is, copy with colour coding), and on white background. The code must be in text (that is, absolutely no snapshots of the code).

**Appendix C - Fritzing**

Include the snapshot of your fritzing breadboard-view schematic. Include as many as you have, but clearly describe which is which. This is in addition to the fritzing file that you submit to the Canvas.

**Appendix D - GitHub**

Every group member must have reasonably and equally contributed to the project github repository. If that is not the case for any member and there is a valid reason as to why, please include an explanation here.

**Appendix E – Complete Component list**

Include the list of all the components used for the project.

If you have used any component you have purchased on own, include full info, a link to datasheet, and cost.

* One Robot hull
* 2 robot feet
* 4 servo motors
* HC-SR04 Ultrasonic Sensor
* 1 Adafruit ItsyBitsy M4
* 1 Adafruit 1.44 inch TFT Breakout
* 1 Piezo Speaker
* A whole lot of wire

**Appendix F – Answer the following questions:**

Q1 – Teamwork: Explain in details the methods your group has used to communicate effectively among team members.

To communicate with each other we created a Facebook messenger group chat. This allowed us to communicate effectively and quickly. On top of using Facebook messenger we created a ganate chart. This chart kept track of tasks that needed to be done, who was in charge of the task and when it was due.

Q2 – Design Process for the additional functionalities: Describe clearly the process you used for the following design aspects of your own additional functialities. Please spend time to carefully answer each of them.

1. **Use of process**: Describe your approach to adapt and apply a general design process for any additional feature. What was your approach?

When doing the project, we split up the group, so each person had a specific task to do. For harder parts of the project like building the robot, programing the LCD screen, working on the buzzer or ultrasonic sensor we split up into groups of two. After doing our individual parts we would review the code as groups of 2-3 people. This helped us quickly fix bugs, get a new perspective on the task and integrate quickly into the final product.

1. **Constraint identification**: Explain the constraints that you must consider in the design of the additional functionalities.
   1. **Constraint 1**: The biggest constraint we had was the amount of ports on the itsbitsy. To fix this constraint we could have used a shift register.
   2. **Constraint 2**:
2. **Solution generation**: Explain at least two possible alternative additional features that your group rejected due to technical reasons and explain why.
   1. We decided to not go with the keypad. We did this because of constraint one the amount of ports on the itsbitsy. We could have integrated this into the robot with a shift register but we decided that the amount of benefit vs the amount of work was not worth it.
   2. We decided to not go with the LED light show. We decided that an ultrasonic sensor and buzzor would be better additions because they were more work but had more impact and made the robot cooler.
3. **Solution Assessment**: Explain how you tested and assessed the viability and then correctness of your group’s additional features.
   1. Correctness was very important to our group. Before integrating any part into the project, we tested it extensively. You can see this in our GitHub repository. Every dance has its own file and was tested in that file. By testing each part individually in their own file, we could make sure there were no errors with that file and so it was very easy to make sure it was correct. We did this with all parts of the robot wheater it was a dance, the buzzer, ultrasonic sensor or integration.

**Appendix G - Other**

Include any other relevant info that does not fit in any other section in the report.