**CPEN 291   
Project 1 Report**

**A. Group info**

Lab section: *L2B* Group #: B\_G11 Group’s Lab Bench #s: 11 & 12

Student names:

|  |  |
| --- | --- |
| **Sanjeev Krishnan** | **Parsa Riahi** |
| **Arnold Ying** | **Amir Ali Barkam** |
| **Stella Wang** | **Rain Zhang** |
| **Manek Gujral** |  |

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:

* The hardware including the circuit for the reflective optical sensors, and why you chose this configuration (i.e. the number of sensors you used and the way you arranged them relative to each other)

For the reflective optical sensors, we used a 100 Ohm resistor connected to the Digital in pin and a 4100 Ohm resistor connected to the 5V pin. In total, we used 4 reflective optical sensors for our robot. They are used as sensors to support our PID controller that detects whether the robot is aligned with the track.

The four sensors are configured in two pairs that are situated tape-width apart - approximately 1 cm (See Appendix G-1 for diagram). This way, if the inner sensors detect black while the outer detect white, the robot is following the line. If both sensors in the right pair detect black while both sensors in the left pair detect white, then the robot is straying left, and vice versa for straying right.

Used in conjunction with a PID controller, we are able to generate a stream of error values based on the robot’s alignment with the track to output adjustment values to the motor, so the robot will turn.

* The algorithm for the line tracking functionality

The algorithm for the line tracking functionality is a Proportional Derivative Integral (PID) controller that takes into account the current error value, its derivative and its integral over time according to binary input from the four sensors. Each sensor outputs a bit, with 0 signifying white and 1 signifying black. Therefore, we get a 4-bit input from four sensors with the most significant bit from the left most sensor and least significant bit from the right most sensor.

Using the input, we calculate an error value that is either positive, negative, or zero which signifies an adjustment to the right, left, or none, respectively. For example, if the input is 4b’0110, then our PID outputs a value of 0, namely no adjustment is required and robot continues to move forward.

Sensors are sampled at a rate of 3000 Hz, namely approximately every 1/3000 seconds. The sample rate was decided upon after trial and error and fine tuning.

* The headless Pi use, implementation, and challenges

The headless Pi is used as the main controller for the PID algorithm and the motor speeds. It is attached to a portable battery pack and the Motor Hat and is situated on the 2WD Mobile Platform robot to allow autonomous function and can be controlled via ssh if needed after powering on.

It is at times difficult to control the RPi headless, as it can only be communicated via terminal. In the beginning, it was difficult to get used to coding through the terminal, but

* Battery-operated robot implementation and challenges

The main body of our robot is the 2WD Mobile Platform, which consists of 5 AAA batteries and 2 DC motors. The two DC motors connect to pins on the Motor Hat to allow control of voltage supply via the MotorKit (software). The battery source is attached to a power switch on the 2WD Mobile Platform before connecting to the Motor Hat.

The greatest challenge that we ran into with the robot implementation was during fine tuning of the motor and PID control. Many times, we will fine tune our code to working condition and come back the next day to have the robot not following the line properly again. This is due to a number of factors: battery pack drains slowly, ambient lighting and shadows in testing area, different track shape, etc.

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

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

* What the additional functionalities are

Mobile app

LCD Display

Live camera feed

twitter bot

wall detection

* Include the list of the additional components you used
* ultrasonic sonar sensor
* How camera is used as a part of an additional feature

* The hardware implementation
* The software implementation

**D. Test and evaluations**

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.

Since each of us started off by writing separate code in separate files for each different component, we did initial testing for each component separately. For example, one person wrote ‘motor.py’ for software controlling the DC motors, while another person wrote ‘camera.py’ for testing and controlling the camera. After ensuring hardware and software for each component was working, we started integrating two components at a time and ensured the integration was functioning before integrating additional components. For example, the first two components that we integrated together was the motor and PID controller.

To test the motors, we tried both stepper motor and motor.throttle from the MotorKit library. We started by connecting one motor to the MotorHat on the Raspberry Pi and ensured that the motor moved according to the code. We chose to use motor.throttle over stepper motor because the output movement on the motor was more consistent. Then, we attached both motors to the Motor Hat and ensured both motors moved simultaneously and according to code.

After ensuring motor control via software, we tested the integration of our PID controller. This part required lots of iteration and fine tuning from trial and error, as various factors affected the output on our robot. For example,

Finally, after ensuring the basic functionality of robot following a line, we moved on to implementing and integrating additional functionalities, starting with the camera and LCD display.

**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, …

This was a very open-ended project, which allowed more freedom and creativity, but also required more responsibility and communication. In earlier stages, it was difficult to figure out equal tasks for each team member, especially because we were not fully set on all of our ideas yet. The process was an iterative one and there were no formulas or guidelines for us to follow closely, and this lead to lots of communication - for large or small steps - during the design process, to ensure that everybody is on the same page and being productive with their time.

In the end, each member on our team rotated taking leadership depending on the component that they focused mainly on and where that fit into the design process. For example, teammates in charge of the motor and PID took charge in the earlier stages of the process, while the teammates in charge of the LCD display displayed more leadership towards the later stages in the process.

**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 view)

**Appendix A – Robot pictures**

Include 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**

Include the complete Python 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 proper 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. 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.

* Raspberry Pi
* 2WD Mobile Platform robot kit
* Motor Hat
* 2x DC Motors
* LCD Display
* RPi Camera attachment
* 5x AAA Batteries

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

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

To streamline the development process, we clearly delegated tasks to each teammate both through a README on Github as well as in person, wherever possible. We developed a group culture of openness and honesty so that every team member was able to contribute and test their ideas effectively. This also allowed for extensive constructive criticism to occur so that we could rapidly prototype effectively and develop and polish a large breadth of ideas fast into our core additional functionalities and core features.

To maintain a consistent and efficient level of communication, our team used a messenger group chat to stay in touch while not meeting face to face. This was especially practical during the reading break since many of our team members were off-campus at this time, and yet we were still able to coordinate meetings, delegate tasks, and to complete the project during this time.

Another way we ensure that everyone is working equitably for the project is through the analysis of our Github commits in terms of lines added and removed, past peer reviews, and qualitative milestones reached for the hardware and for the report. This allows for a simple diagnosis of everyone’s workload, and gives the opportunity to each teammate to see where they may be falling behind, and thus incentivizing consistent self-improvement for the teammate as well as ensuring we meet deadlines.

Q2 – Design Process for the additional functionalities: Describe clearly the process you used for the following design aspects of your own additional functionalities. 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?

We used an iterative rapid prototyping process to incrementally add new features to our line tracking robot. We began with the base requirements given to us by the lab description, then went on to address the open-ended problems we still saw with the system, adding multiple new features to improve client engagement and to be able to fully visually display the data we received from the RPi.

1. **Constraint identification**: Explain the constraints that you must consider in the design of the additional functionalities.
2. **Solution generation**: Explain at least two possible alternative additional features that your group rejected due to technical reasons and explain why.
3. **Solution Assessment**: Explain how you tested and assessed the viability and then correctness of your group’s additional features.

**Appendix G - Other**

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

G-1) Reflective optical sensor configuration diagram

