**PROJECT III DC MOTOR SUBSYSTEM**

Team 3

Joseph Page, Joshua Silva, Natalia Jenkins, Garrison Veith

03/14/2025

# **Executive Summary**

For Project III, exploration of the DC motor to move & point to the centroids is done. Through conducting various experiments with the DC motor’s model, planning of the motors required for the design can be done, more specifically what math & measurements are required for the system to move to the ball & “hit” it. The DC motor’s use in this project is to point to the centroid of the ball, allowing for future considerations for the “paddle” to hit the ball. Furthermore, learning how MATLAB can pipeline data to Simulink is done and can be implemented for additional motor use in the system. Lastly, more ideas for the GUI are explored, all having the intention for easy user access to the system without needing to see the code & motor models.

**Overview of Hardware-in-the-Loop**

Hardware-in-the-loop simulation is exactly what it sounds like. There is a virtual model of a system that is transferred to a controller, like an arduino or a microcontroller. The controller is connected to sensors, actuators, etc. and it uses the virtual model to process the physical inputs from the sensors and then generates an output to the actuators. An engineering computer is also connected to the controller in order to change and update the program as needed. In the case of this project, the hardware-in-the-loop is the motor, the model is then run on the arduino that controls the motor when given an input from a camera. This type of simulation is generally meant for validation of the hardware or the system. It is usually done towards the middle or the end of the design process. Since it requires the actual hardware to perform this type of simulation, it is more expensive in comparison to a full simulation. This type of simulation is also more complex than a full simulation since everything has to be physically connected and cannot be accelerated.

A full simulation, on the other hand, is only done in software. There is no hardware involved in a full simulation. This is generally done early in the design stage to test and debug the program and the logic that is being developed. This approach requires all of the hardware to be simulated. So any inputs, like those from a sensor, have to be part of the simulation. Since it is only done in software it is generally much easier to set up than hardware-in-the-loop simulation. This also means that the cost of setting up a full simulation is cheaper. Since this simulation is only on a computer, time can be accelerated in order to debug a program more quickly.

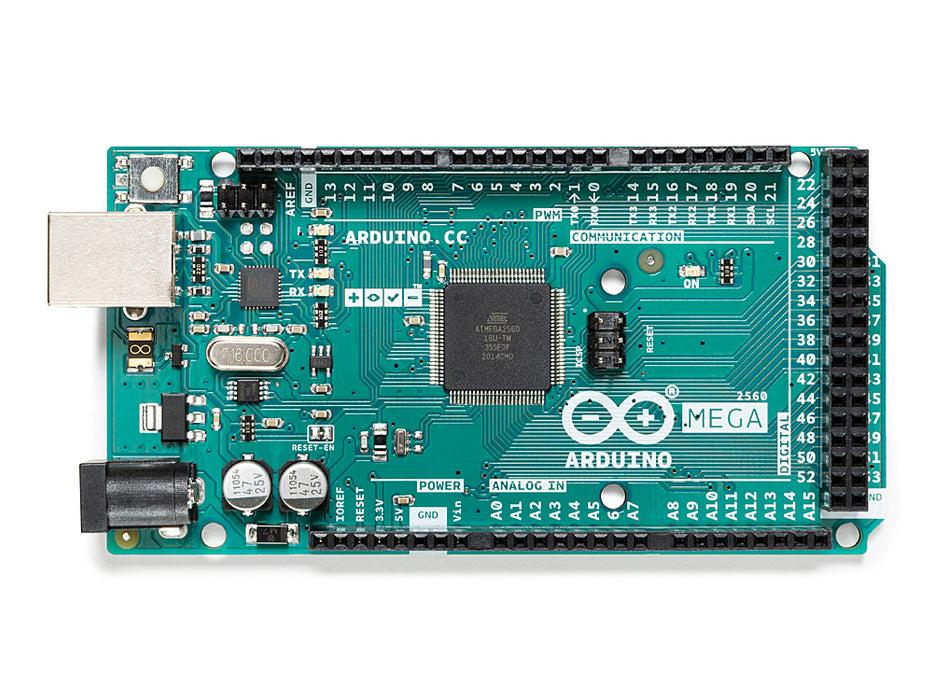
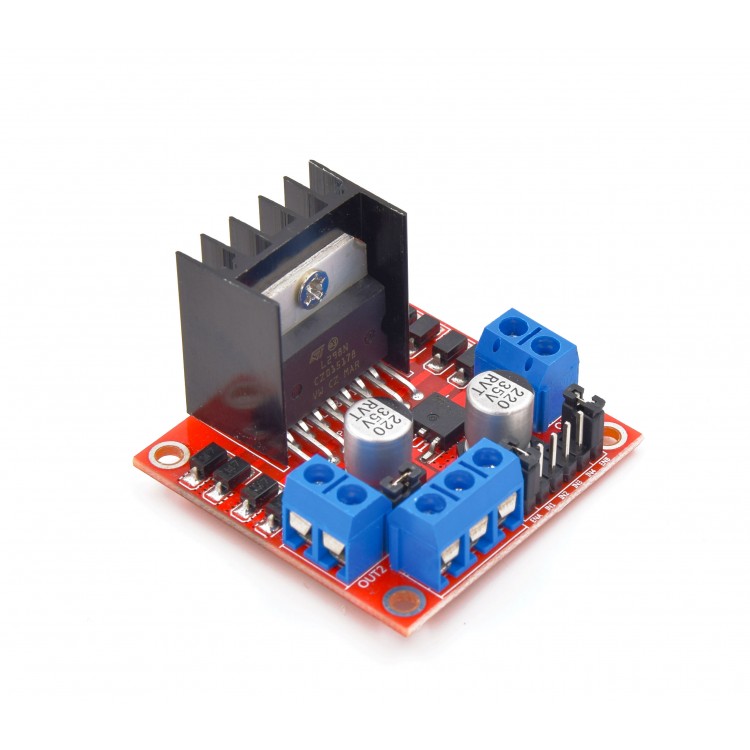
The major strength of hardware-in-the-loop simulation is that it can be used to validate a design. This is the main purpose of hardware-in-the-loop simulation. It is done after full simulation. Full simulation is for developing and debugging a program’s logic. Hardware-in-the-loop simulation is for validating the program and debugging the hardware. Its strength lies in the ability to make changes and then test those changes quickly on the actual system. This allows the programmer to validate that the program and the hardware work as expected.

**Document Hardware**

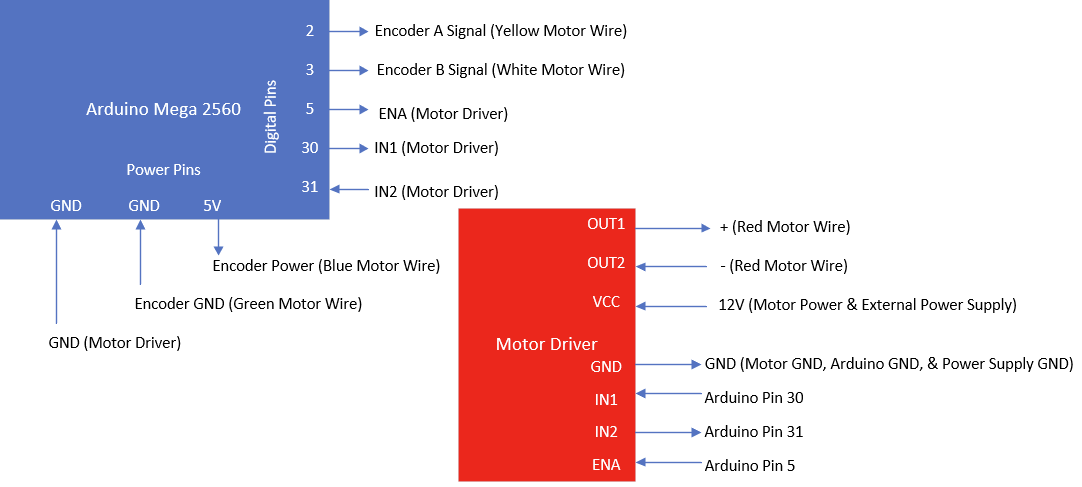
The hardware used for this project is as follows: A USB webcam, a DC motor with an encoder, an Arduino Mega 2560, and an H-bridge motor driver. Images of these parts are in figures 1-4. The wiring diagram showing the connection of the DC motor, the H bridge, and the arduino is in figure 5. Not shown in the wiring diagram is the connection from the arduino to the computer and the connection between the webcam and the computer. These connections simply use USB to connect to the computer.

** **

**Figure 1: USB Webcam Figure 2: Brushed DC Motor with encoder**

** **

**Figure 3: Arduino Mega 2560 Figure 4: H-Bridge Motor Driver**



**Figure 6: Wiring Diagram**

The image that is being used to calculate where the ball is and where the holes are is 622x320. Using the distances between the holes, the pixels per centimeter was calculated to be 2.9 pix/cm. This is about 9.90599 pix/in.

While the camera isn’t perfect, it is probably the best sensor for this project. It is easy to use in matlab and takes high quality photos. The Matlab functions for image processing make finding the holes and the ball very easy. There are, however, some caveats to the camera. It is highly sensitive to any changes in the framing of the board, if it is slightly off from the original picture, it doesn’t work very well. Because of this, the frame holding the camera needs to be robust and hold the board in the same position every time. The camera is also very sensitive to light. Due to the position of the workbench right next to the windows, at certain times during the day there is direct sunlight on the board. This causes the camera to give inconsistent results. Despite all of the downsides, the camera is the best option for this project.

**Document Software**

The flowchart in Figure 7 outlines the software system. The process begins with a pop-up asking the user if they want to take a background image of the putting green without the golf ball. After the user makes their selection, another pop-up will prompt the user to take a current picture. The current picture should be an image of the golf ball and putting green. The software will perform image subtraction (current image - background image) and other image processing techniques to detect the centroid location. The centroid location is stored as an x and y coordinate.

Next, MATLAB will compute the motor angle. The motor-to-putting green distance is added to the centroid’s y coordinate to account for offset. The angle is calculated by using the equation θ=tan−1(y adjusted/x​). The calculated motor angle is sent to the input angle block in the Simulink motor model file using the “set\_param” function in MATLAB.

Simulink will receive the calculated motor angle and will perform closed-loop motor control. As the motor moves, the model constantly compares the desired angle with the encoder feedback. A summation block will calculate the error between the reference angle and the encoder feedback and feeds the error signal into the PID controller. The PID controller will generate a control signal based on the error. The control signal is a voltage. The PID block seems to generate its output using the parallel form of the PID equation. Below is the formula in the Laplace domain:

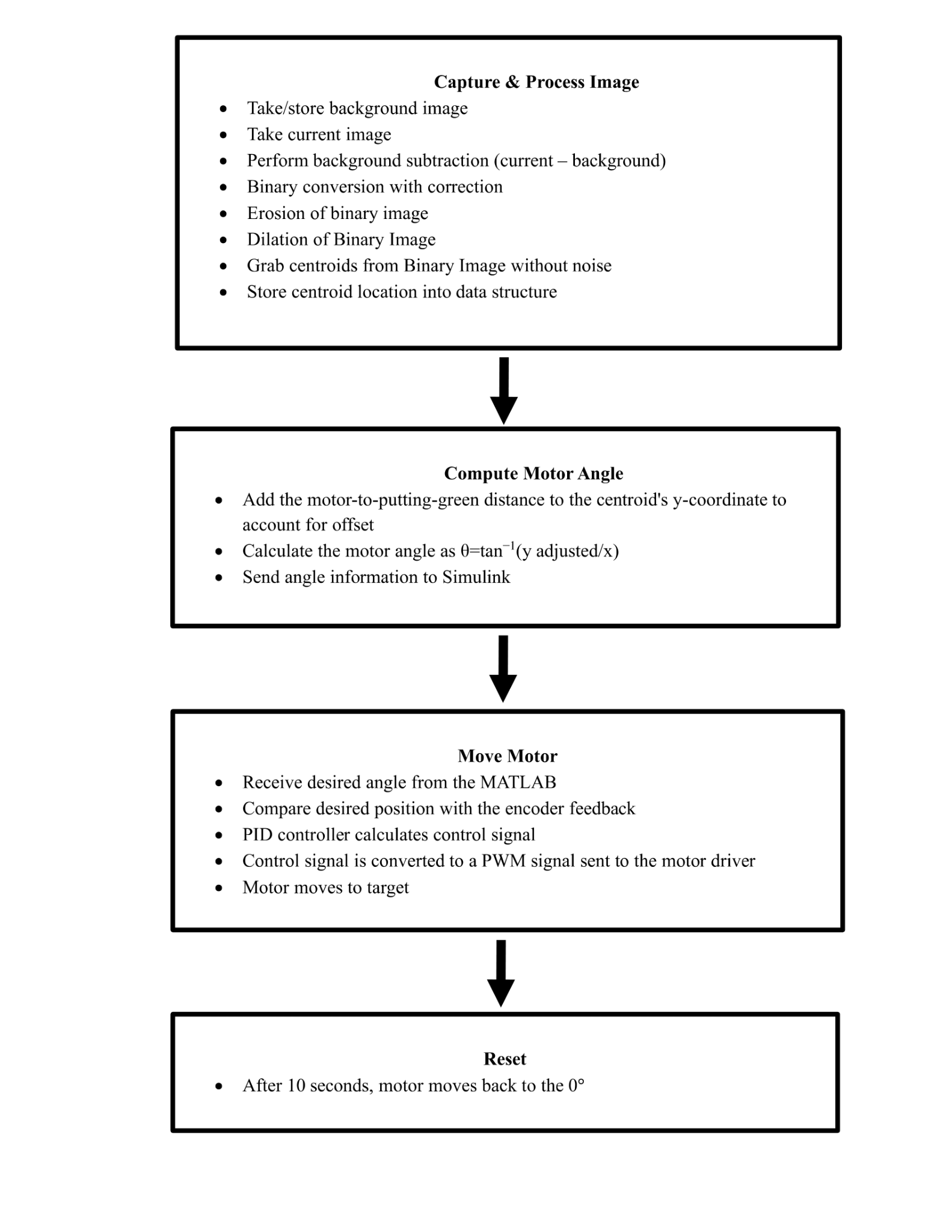


The control signal is fed into an absolute value block. The output from the absolute value block is labeled u2. The control signal is also fed into a saturation block that limits the signal range to -5 through 5 volts – ensuring the signal stays in a reasonable range. The output from the saturation block is labeled u1. U1 and u2 connect to the “Choose Direction” block. If u1 >0 and u2>3, the motor moves forward. For the motor to move forward, pin IN1 on the Arduino is set to 1, and pin IN2 is set to 0. If u1<0 and u2>3, the motor moves backward. For the motor to move forward, pin IN1 on the Arduino is set to 0, and pin IN2 is set to 1. If none of the above conditions are met, the motor stays in a constant position. For the motor to stay in a constant position, pin IN1 on the Arduino is set to 0, and pin IN2 is set to 0.

The control signal is multiplied by 255/3, representing a 0%- 100% duty cycle. This signal is sent to the Arduino's PWM pin. The motor moves faster if the control signal is higher and more slowly if it is lower.

The encoder generates two output signals - encoder A and B. Quadrature encoding is used to count the encoder ticks and the motor direction. There is an “Encoder” block in the Simulink file which will instruct the Arduino to trigger an interrupt on the rising edge on encoder A and to count the interrupts to determine the number of pulses from encoder A. Encoder B is used to determine the direction of the motor. Encoders A and B are phase-shifted by 90 degrees. The direction of the motor can be determined by comparing encoder B and encoder A. For example, if encoder A leads encoder B, then the motor is moving forward. If encoder B leads encoder A, then the motor is moving backward. The number of encoder ticks are converted to angular position in degrees using a gain block with a value of 360/2797. The encoder signal is fed back into the summation block, completing the closed-loop motor control system.

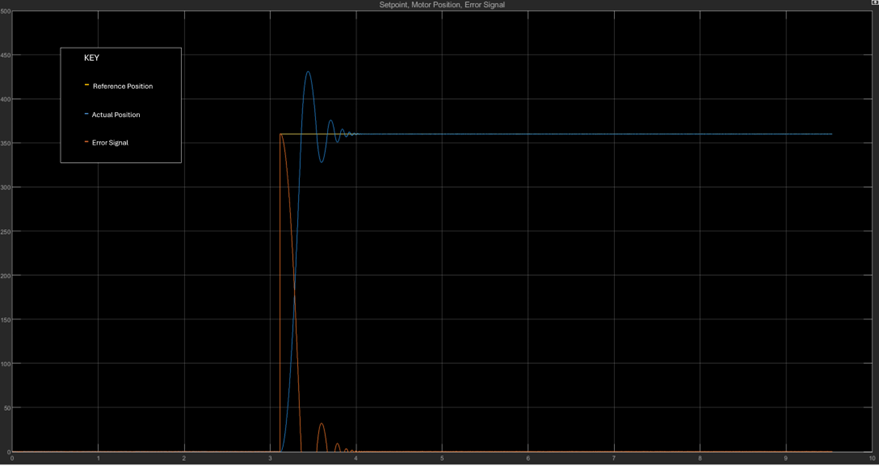
In summary, the process begins with capturing and processing images in MATLAB to determine the coordinates of the location of the golf ball on the putting green. Image processing techniques, including background subtraction, erosion, and dilation are used to determine the location of the golf ball. Next, the inverse tangent and the golf ball coordinates are used to calculate the desired motor angle. This motor angle serves as the input of the closed-loop motor-control Simulink model. The desired motor angle and the encoder feedback are used to calculate the error. The PID controller uses the error to calculate a control signal. The control signal is processed and converted to a PWM signal and sent to the motor driver. The motor moves to its target. After 10 seconds, the motor moves back to 0°.



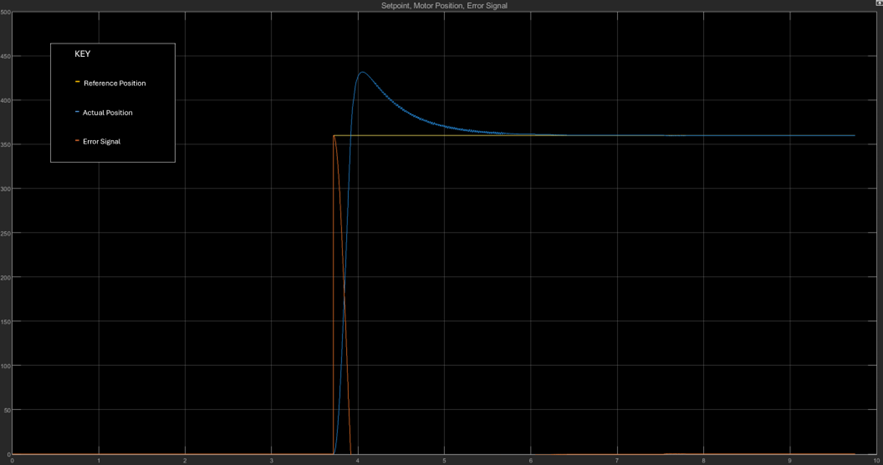
**Figure 7: Software Flowchart**

**Document Motor Experiments**

Figures 8, 9, and 10 display the effect of changing the proportional, integral, and derivative gains. For each plot, only one gain was adjusted while the other two remained at their tuned values. Increasing the proportional gain caused oscillations and overshoot. The error signal spiked up to the reference position and continued to spike upwards with values of diminishing amplitude until the error approached zero.Increasing the integral gain increased overshoot, but unlike increasing the proportional gain, increasing the integral gain did not cause oscillations. The error signal spiked initially but quickly converged to zero. Increasing the derivative gain improved stability but caused the motor to undershoot, leading to the motor never reaching its reference angle. The error signal jumped up to the reference angle and then exponentially decayed. The error signal never fully approached zero.

****

**Figure 8: Effect of Increasing Proportional Gain (P)**

****

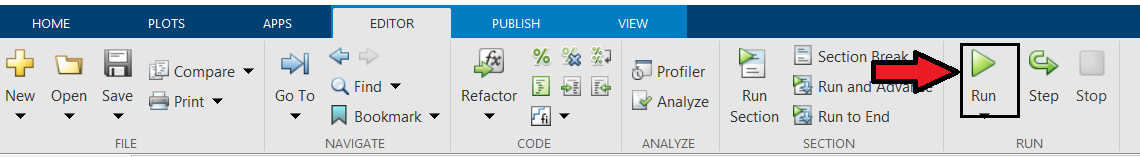
**Figure 9: Effect of Increasing Integral Gain (I)**

****

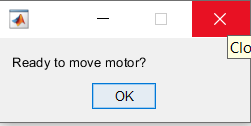
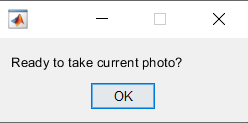
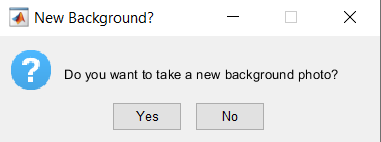
**Figure 10: Effect of Increasing Derivative Gain (D)**

**Document and Evaluate your User Interface**

For the project, a GUI assists the user in running the golf board and hitting a ball into a specific hole. At the current state, the only user interface is running a MATLAB script and seeing simple dialog boxes and message boxes. See Figures X & X below.



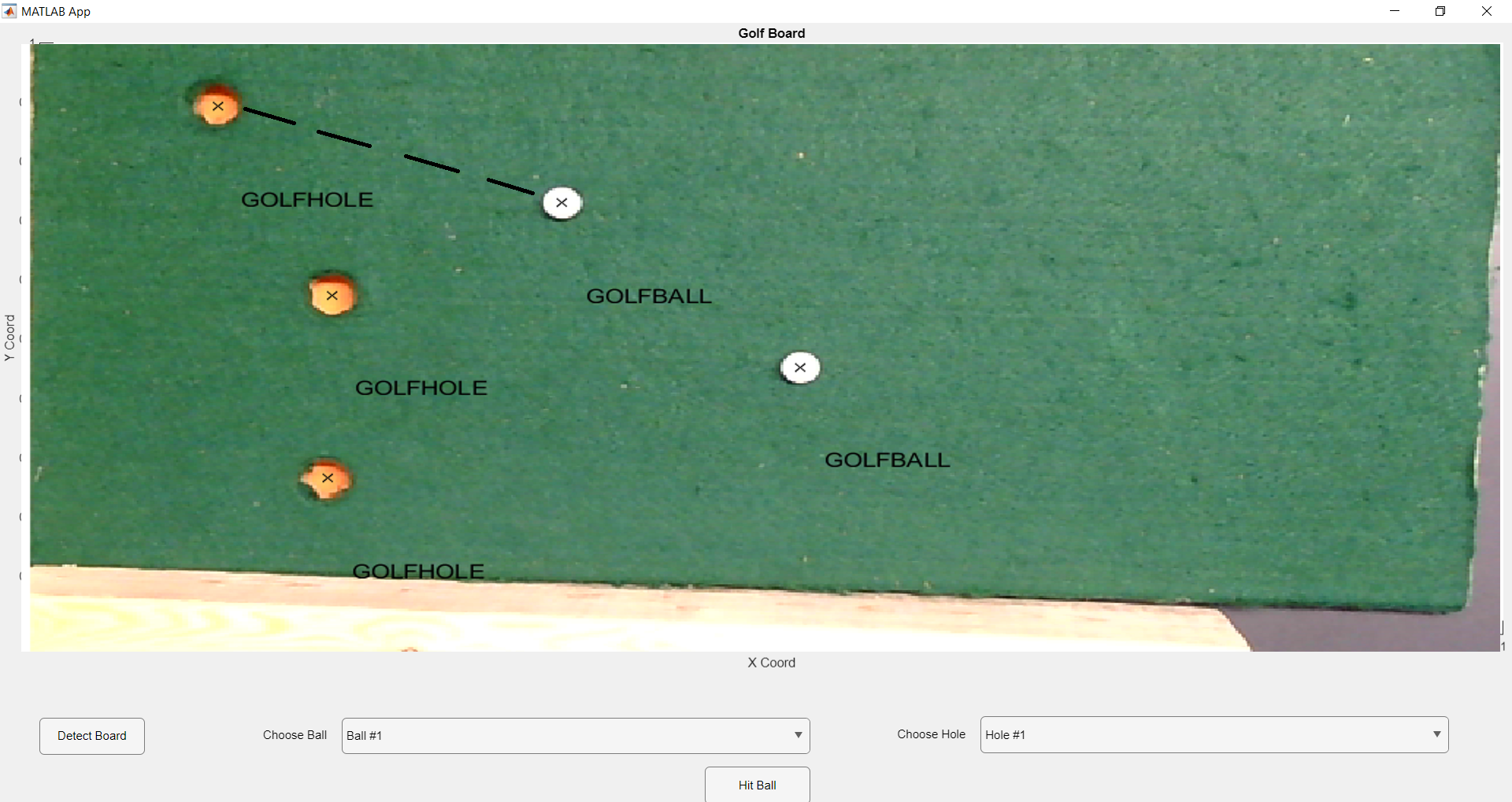
**Figure 11: Run Script**



**Figure 12: Dialog/Message Boxes**

The boxes allow the user to have enough time to set up the board for the camera, instead of instantly taking pictures as soon as the script runs. To save some time as well, the background photo does not need to be retaken every time, saving to the user's current directory. Otherwise time is allowed for the user to reset the board to take a picture of the current golf ball(s)/hole(s) locations, and to run the motor.

Below in Figure X, the prototype of the final app is shown. Ideally, it allows the user to detect the balls & holes on the board, displaying it back to the user after running the necessary script. Afterwards, the user can select the golf ball and hole desired, and then “hit” the ball in, sending the data to the motors. The idea is for the user to not see the script, allowing for easy use and less “noise” as they run the system.



**Figure 13: App Prototype**

**Financial Analysis**

Assuming Company operates in Upstate South Carolina, offering around 68K to compete with other competitors in the area (not accounting for different experience levels). Company deals with home devices, for example home alarm systems, thermostats, “smart home” etc.

**Home Security-TEK Engineering**

# **Start-up Costs**

| Personnel | 8 Engineers @ $68K/yr + President @ $75K/yr + Admin. Asst. @ $25K/yr = $644,000 |
| --- | --- |
| Fringe Benefit (FB) | A fringe benefit is a form of pay for the performance of services. For example, you provide an employee with a fringe benefit when you allow the employee to use a business vehicle to commute to and from work. Assume Fringe Benefit Package @ 36% (incl. employee's SS tax, vacation, holidays, medical, retirement (401K), dental, life insurance, relocation, unemployment insurances, etc):  (0.3 x $68,000 + $75,000 + $25,000) x 0.3 = $193,000  *Note: Federal Insurance Contributions Act (FICA) tax (Social Security and Medicare) is imposed by the federal government on both employees and employers. The entire FICA percentage of 15.3%*   * *Employee's pay 6.2% for SS and 1.45% for the Medicare (this is not included in your cost)* * *The employer is liable for 6.2% Social Security and 1.45% Medicare taxes=7.65%* |
| Building | Initially rent a suite of offices with 2 engineers/office (12' x 14'), an office/conference room for President (12' x 20'), and a reception/office area of 16' x 20'.  (4 cubicles) x (12' x 14'/cubicle) + President office of (12’x 20’)  + Reception/office area of (16' x 20') = 1,232 sq ft  Use nominal figure for office space in industrial park sectors of` Clemson area, $9.50/sq ft/mo. Then the lease rate for office space will be  $0.79/sq ft/mo x 1,232 sq ft = $973 /mo. = $11,679 /yr. |
| Furniture | Rental of a desk, chair, credenza set will run about $60/mo. Need 10 sets for a total  monthly expenditure of $600/mo = $7,200/yr  The remaining equipment, furniture and software expenses are estimated to be about  10 computers @ $1500/computer $15,000  10 sets of general software @ $1000/set $10,000  Specialized software $18,000  Copier, printer $4,000  Table and chairs for conference room $3,888  10 telephones @ $35/ea $350  Total $58,438 |
| Phone and Internet | According to Bell South, the cost of a combined voice/data line, is $70.00/mo for operation.  For 10 telephones the total cost will be $8,400 /year.  Assume that long distance calls add another 40% to this to get a total estimated annual phone cost of $11,760 |
| Travel | Cost per local trip is $3000 and the cost per out-of-state trip is $200 there will be 2 of each trip each month  $6,400/mo for the first year, or an annual total of $76,800. |
| Interest | The monthly payment is M = $74,726  Debt Service = Total interest paid in year = 11 x M - P = $21,983. |

**Cost Estimate**

Salaries $ 644,000

FB @ 30% $ 193,200

Building $ 11,679

Furniture $ 58,438

Debt service $ 21,983

Travel $ 76,800

Internet and Phone Service $ 11,760

Total Costs $ 1,017,860

## ***Overhead Calculation***

The first year, the 8 engineers will be at least 75% "sold"

8 engineers @ 75% sold $408,000

(salaries billable to clients)

FB @ 30% $122,400

(FB billable to clients)

Total Billable to Clients $530,400

Total Expenses = Total Costs - Total Billable to Clients = $530,400 (Overhead Number)

Overhead rate = ($$487,460/$530,400) x 100% = 91.90%

(Labor Cost + Fringe Benefit) \* (1+ (OH rate/100%) + (5% profit/100%)) in order to recover the costs of doing business and make a profit (assuming a 5% profit). This is the figure that you will use when estimating the cost of a contract to a customer in a proposal. An overhead rate of 150% means that for each $1.00 of direct labor budgeted for a project; $1.50 needs to be budgeted for overhead costs.

## ***Using the Overhead Number***

1 Week of Work: Bill to Client = $3347.30

**Employee Training Program**

Employee training is an essential part of running any business, let alone a start up business. Since this is a startup business many of the people working here will likely be a recent college graduate. This means that they have a lot of the foundational knowledge but they lack experience. While it would be great to be able to set aside a large sum of money for employee training, as a startup, we are strapped for money. This means that we can’t set aside too much money for training. Around 5-10% of an employees annual salary is a good amount to have for each employee’s training “budget”.

The training for engineers who will be solving problems is geared towards gaining knowledge of the technology that they will be working with and seeing how to implement the technology. Trade shows are a great way of gaining exposure to all of the technology an industry has to offer. They are also great places for ideas on how to implement the new technologies. These engineers have gotten a lot of theoretical knowledge in school but they haven’t gotten much hands on experience. They also probably have even less hands on experience in the field that they are going in to. They need to be exposed to all of the technology that can be used in the smart home and home security industry. They also need examples of these technologies being implemented in order to make effective solutions.

The training for the engineers who will be managing projects is more geared towards management and project management. There is a project management expo that goes over new and emerging ideas for how to more effectively manage projects. The management of people is emphasised much more for these engineers. Engineers don’t typically take any type of management classes or don’t take in depth communication courses so these skills need to be developed in order for them to be a successful project manager.

**Newly Graduated Engineer who will be Solving Technical Problems/Projects**

| **Activity** | **Benefit** | **Cost** |
| --- | --- | --- |
| Attending Asia World Expo Trade Show | See new technologies in smart home and home security and see different ways to implement the new technology | Travel to Hong Kong ($4000 round trip) + food and lodging for weekend ($800) + missing 3 days of work ($560) = $5360 per person |
| CEDIA training program | Get formal training on in depth smart home and home security topics | Training cost $600 |
| Pay IEEE Membership | Access to latest innovations and information in field | Membership $400 |
| Certification in relevant fields to add to focus | Would build company and personal knowledge for use in specialized projects | Average Certification Cost(Exam+Application fees): $500-800 |
| Conference on communication, electronics, electrical engineering. NY USA (ICCEEE) | Expose Design team to relevant new ideas and innovations for potential use in system designs | (assuming only 4 employees) travel, admission, boarding, employee cost:~$5850 |

**Newly Graduated Engineer who will be Managing Technical Projects**

| **Activity** | **Benefit** | **Cost** |
| --- | --- | --- |
| Attend project management courses | Get formal training on how to manage a technical project | Course $3000 |
| Attend leadership courses | Gain skills in managing people and being an effective leader | Course $3000 |
| Attend PMXPO | See new ways and techniques for managing projects | Missing 1 day of work $190 |

**Safety Analysis**

| **Description of Component or Subsystem** | **Failure Mode (Hazard)** | **Symptom** | **Effect** | **Probability of Failure** | **Severity of Effect** | **Risk Index** |
| --- | --- | --- | --- | --- | --- | --- |
| **DC Motor** | Motor failure | Motor overheating, or electrical connections compromised | Motor could stop moving altogether | D | I | ID |
|  | Incorrect Motor Control | Gain values could be received incorrectly from software | Rail may go to a position that is undesired | C | I | IC |
|  | Gear Drive Failure | This could occur due to inconsistencies in the rail | cause choppy motion and undesired outcomes and possibly damage parts | B | III | IIIB |
| **Power Supply for DC Motor** | Power Supply Shutdown | Caused by a potential overload of the supply | System is no longer functional | C | I | IC |
|  | Short Circuit | Faulty Wiring externally | power supply damage and additional hazards | B | I | IB |
|  | Thermal Overload | Excessive load over short period of time | can cause permanent damage to power supply | C | I | IC |
|  | Supply Failure | Caused by aging hardware | Loss of functionality and need for a replacement | D | II | IID |
| **Camera-as-Sensor** | Camera sensor failure | Connection Failure, overall wear | Loss of camera sensing capability | D | I | ID |
|  | Image Processing Failure | Issues with lighting conditions, software errors | could cause severe problems for the system | B | II | IIB |
|  | Misaligned camera | Improper camera mounting and angle | distortions and error in subsequent systems | B | II | IIB |
| **Software** | Software Crash | Bugs, extensively long run times | Full loss of system functionality | C | I | IC |
|  | Communication Failure | Bugs and difference in file locations | Would create minor issues in running | C | III | IIIC |
|  | User Interface Failure | Bugs within Interface, Issues communicating between different software elements | Could potentially limit the entire system if UI ran into an unexpected grouping of inputs that could cause a fault | C | II | IIC |

The above Design Failure Modes and Effects Analysis(DFMEA) Table takes a look at some reasonably common and the most important failure modes that we will encounter or already have encountered. The table looks at some things that are very much out of anyone's control. This can include software crashing, Gear Drive Failures or Supply failure. These are typically things we aren’t expecting to occur and can’t really plan for other than considering them as possible and taking precautions. That said, the rest of the failure modes on this list are things that we, as a team, can work hard to avoid by using good practice.

We have encountered some of the listed faults and continuously have been working to minimize these by constantly improving our system design and conditions that we operate under.Our Motor and Power supply are both of importance. First being our motor gain values. This is something that we accounted for by tuning the gain values to be optimal for accuracy and speed of settling time. We also observed the effects of a gain that caused an unbounded output to our motor and understand how to avoid such a problem. Motor Failure is something that we likely shouldn’t encounter but being aware of its physical limitations are likely sufficient in avoiding this problem. Faulty wiring can be an issue if we are not careful in our wiring. This is likely something that wouldn’t occur and cause damage to our power supply but critical redundancy is important in every project regardless of field.

The next things to consider are the camera subsystem and software link between all systems. While these would pose much less physical damage or hazard, they can stop the system's ability to operate. We had some minor issues with our image processing to start out. This was the issue of how we planned to distinguish between holes and balls. We found a way to determine between them using color as our method of determining the difference. Before this we could not reliably distinguish between the two. We then take a look at the possibility of a misaligned camera. Our current camera mount worked well enough for what we needed previously but looking ahead to designing a highly accurate camera sensor, it will be important to work out these issues before moving forward. The misaligned camera, while not the most severe problem, creates a significant source of error at the very beginning of our system. This will then cause the final result to be unreliable. The user interface is another thing that we must plan for in our overall design. While we haven’t yet had issues with this further development should plan for any possible issues with the user not understanding what they are doing. This would involve a number of conditional statements that would verify correct user input is given.

It is always important to consider the possible things that can go bad in any given project. There will always be unknowns and unavoidable troubles that will occur but there are also a number of things we can control. Minimizing the likelihood of these unwanted Failure Modes is the goal of any good product. In a general case, improving product reliability is always something we should strive for as this builds a reputation for those who have designed and the company involved. For many projects of higher complexity this would be a significantly larger list and we would need to consider more potential problems. The goal of our project is to have a design that is very robust given certain conditions we could encounter.

References

[1] A. Kapadia, " Closed-Loop Motor Control and Employee Training Program Project 3 Instructions," ECE 4950, Clemson University, Unpublished, 2025.

[2] A. Kapadia, "ReadMe\_Getting the Motor Model to Work Correctly," ECE 4950, Clemson University, Unpublished, 2025.

# **ECE 4950 Project 3 –Closed-Loop Motor Control, Life-Cycle Analysis, and Risk Assessment**

Use the guidelines below to complete your report and add at the end of your report.

Group Number and Member Names: Group 3 (Joseph Page, Joshua Silva, Natalia Jenkins, Garrison Veith)

| Score | Pts |  |
| --- | --- | --- |
|  | 5 | **General Format - Professional Looking Document/Preparation (whole document)**  a) Fonts, margins (11pt, times new roman, single spaced. 1" margins on all sides).  b) Spelling and grammar are correct  c) Layout of pictures – all figures need numbers and captions and must be referenced inthe text  d) Follows the page limitations below.  e)References. Use IEEE reference format.  f)This grading sheet is included as the final page. |
|  | 5 | **Page 1: Title, Group Name, Group Members, and Date**  **Executive Summary** (1 concise, well-written paragraph) Provide an overview of this project. Briefly describe what you did and what you learned. |
|  |  | **Control Subsystem Design** |
| 5 | **Page 2: Overview of Hardware-in-the-Loop** (~1/2 page) |
|  | Describe in your own words what Hardware-in-the-Loop means. What is the difference |
|  | between a full simulation and a Hardware-in-the-Loop simulation? What are the strengths of |
|  | HIL? |
| 10 | **Pages 2-3: Document Hardware** (1 page) |
|  | Describe and show images of the equipment used, connection diagrams, calculation of |
|  | resolution – pixels per square inch/cm on game board etc. Is the camera an appropriate sensor? |
| 10 | **Pages 4-5: Document Software** (2 pages) |
|  | Using Flowcharts, state diagrams, data structures etc. describe how the software is |
|  | implemented. There is no need to include the source code. |
| 10 | **Page 6: Document Motor Experiments** (~1 page) |
|  | Plot the effect of changing gains using the reference, actual position and error signals. What |
|  | happens when the proportional, derivative and/or integral gains are changed? |
|  | **Pages 7: Document and Evaluate your User Interface** (~1 pages) |
| 15 | How does the user interface connect the Camera-As-A-Sensor and the motor? What |
|  | information is provided to the user and why? Document using screenshots and similar images. |
|  | 10 | **Pages 8-9 Life Cycle Analysis (2 pages)**  You are proposing a design that consumes resources. Follow the  “Life Cycle Assessment (LCA) Exercise” for the shipping box for your project to examine the life cycle for this one part of your design. Be sure to interpret the results of the computer program. Complete this section of the report by saying that a similar analysis could be done on  the entire project to reduce environmental impact. |
|  | 10 | **Page 10-11: Financial Analysis**  Provide a financial analysis that examines turning your group into a start-up company. Use the spreadsheet provided to make calculations and report your results using the MS  Word document template and include here. |
|  | 15 | **Page 12: Employee Training Program** |
|  | 15 | **Pages 13-14: Safety Analysis**  The project must be safe for use by the customer. Perform and Document a DFMEA for your project. Document your analysis using the DFMEA Table and Risk Assessment Matrix shown in the class slides. Show that you have implemented the results of the analysis to make your design and workspace safe – that is document what changes you made to make your system  and space safe as a result of the safety analysis. Can you conclude your system is safe? |
|  |  | **Page 19: Grading Sheet** |