System Architecture

Electrical Components

In order to power the Arduino, it was connected to a 12 V external power supply. Additionally, the Arduino was connected to a laptop via a USB A/B cable. The magnet circuit seen in figure 2, which was created for project 1, was used to control the electromagnet. This circuit allows the electromagnet to be powered while preventing back current from entering the Arduino. The circuit was built using a breadboard, allowing for quick changes to be made during prototyping.

With the system mostly built, the next step was to implement the motor so that the gameboard could rotate its position according to the program’s algorithm. However, the code was written to take the shortest path possible. In order for the physical system to be able to match the decisions made by the code, the motor would need to rotate in both directions but can only do so with the help of a motor driver. To remedy this, an L298N motor driver was purchased and incorporated into the system to give the motor the flexibility to rotate in both directions.

Since the group used a 12 V power supply into the arduino, the input voltage (Vin) was connected to the L298N so that the motor can be powered and any other low-powered electronics connected are not affected by the source. The output, of the motor driver, is then connected to the motor so it is sufficiently powered. With the components connected, the system was given power to ensure that they functioned by themselves so that they were not an issue in further project development.

The cost of the electrical components used include $0.20 for two 1k ohm resistors, $17.05 an L298N motor driver, $38.50 for an Arduino mega, $12 for 12V external power supply, $0.20 for an IN4001 diode, $0.22 for a 2N3904 npn transistor, $2.90 for the breadboard, and $1.95 for a set of 20 wires. This results in a total cost of $70.12 for the electrical components. All of these prices come from Digikey.

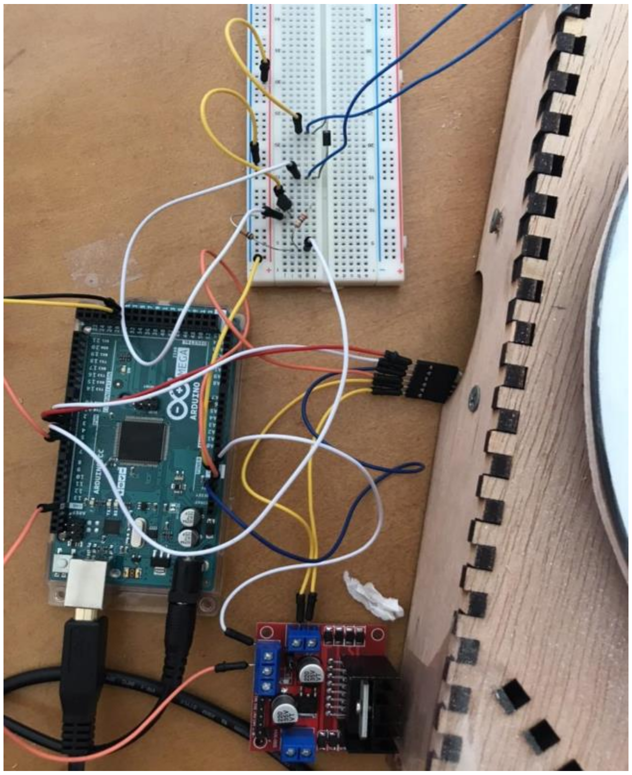

Figure 1: Final Circuit

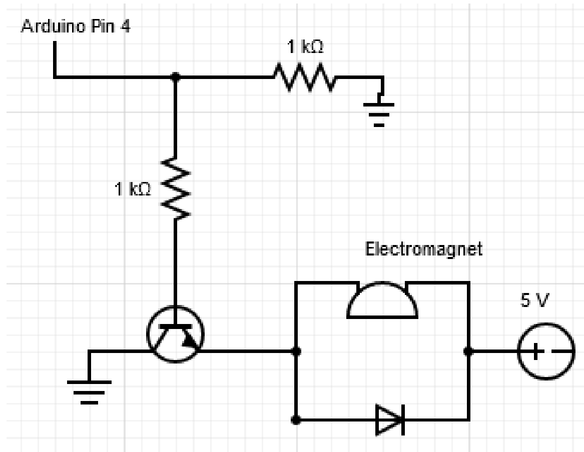

Figure 2: Electromagnet Circuit

Mechanical Components:

The overall system consists of a box to house the motor and support the stands for the electromagnet and camera, and a circular game board attached to the motor shaft, via epoxy, so that the gameboard is centered on the box. It is also critical that the gameboard is assured to be level at this stage, otherwise tuning the system to pick up only the intended washer will be much more difficult if not impossible. The game board will rotate and move the wells to be directly under the electromagnet. In order to help background subtraction the top of the game board was painted white. A stand for the electromagnet is attached to the center of one side of the board. This stand will hold the electromagnet 30 mm from the end of the box. This will result in the magnet being directly over the wells. A stand to hold the camera is attached to the opposite side of the board. This stand will be 900 mm tall and suspend the web camera over the center of the game board. The camera is this high so that it can have a clear view of the entire game board. The exact dimensions of this design can be seen in figures 4 and 5. In addition a picture of the final system can be seen in figure 3.

The cost of the wooden parts of the system include: $5 for the box, $2 for the gameboard, $0.5 for the electromagnet stand, and $4 for the camera stand. In addition $4 for the white spray paint. These prices came from Home Depot. The prices for the mechanical components of the system include $29 for the 12V DC motor, $7.50 for the 5V electromagnet. These prices came from Digikey. This results in a total cost of $52.

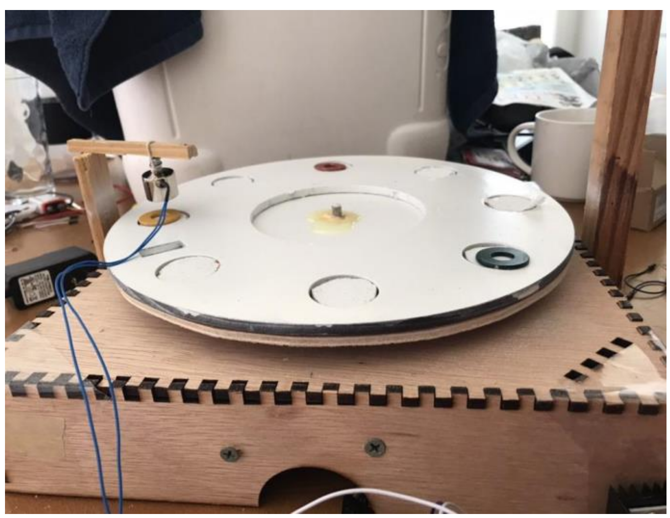

Figure 3: Picture of the Final Design

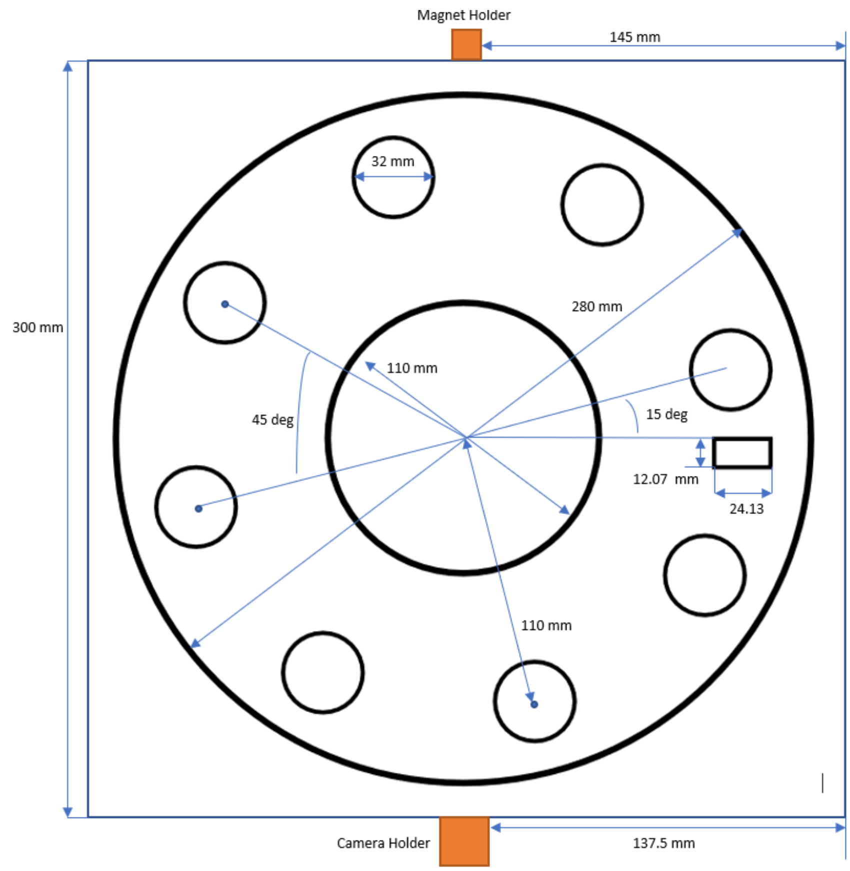

Figure 4: Overhead view of the device

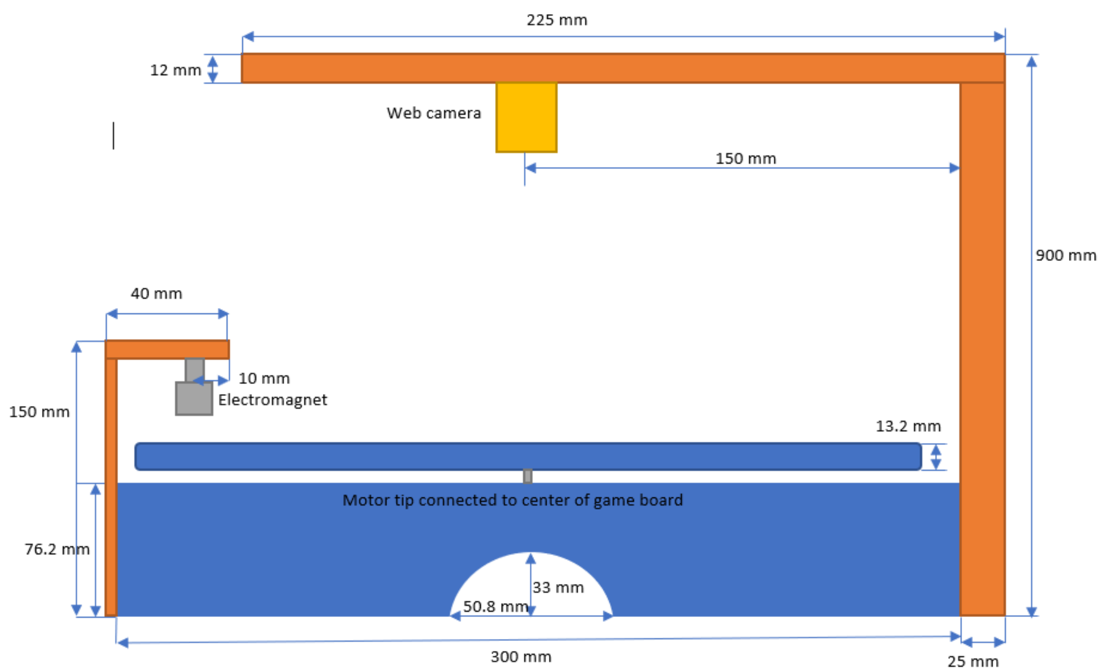

Figure 5: Side view of the Device

Software Components:

The Docooler USB 2.0 HD Camera takes pictures that are 640 x 480 pixels. Two pictures need to be stored, one for the background image and another to calculate the current game state, thus the device needs 614400 pixels. With 8 bits per pixel this results in the system needing 614.4 KB to store these images.

Matlab and simulink were used to implement this system. Matlab was used to create the GUI, to perform background subtraction, and to calculate how to sort the washers. The simulink diagram used can be seen in figure 7. To program the Arduino the simulink code was converted to c and the Matlab code was run on a real-time workstation.

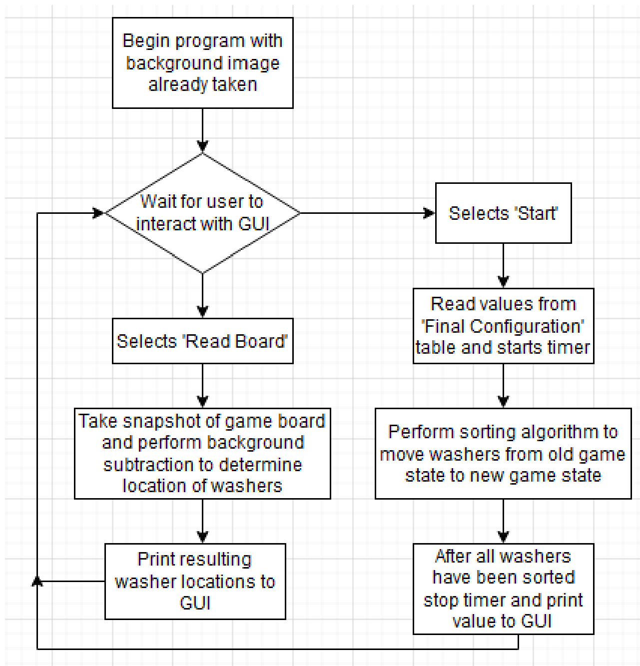

Figure 6: Flowchart for Washer Sorter Software

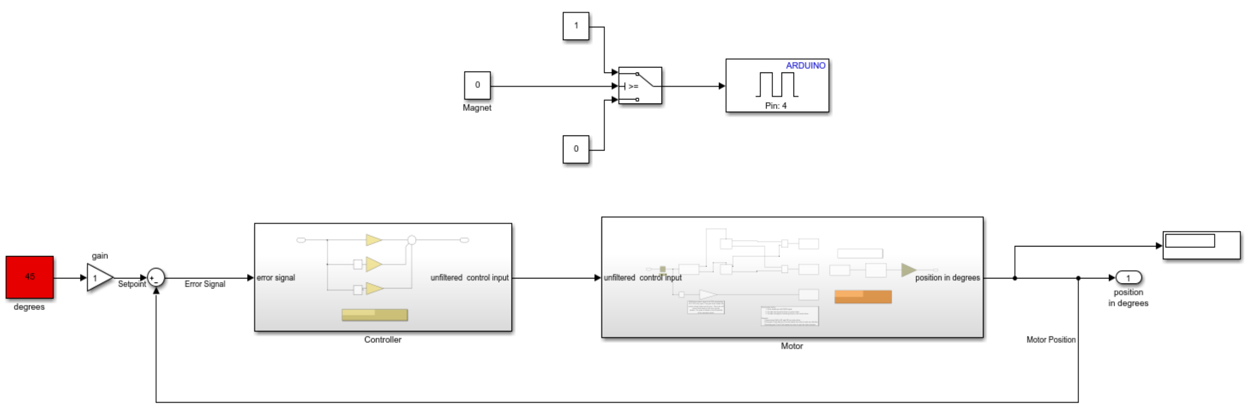

Figure 7: Simulink model for both closed-loop motor control and electromagnet activation

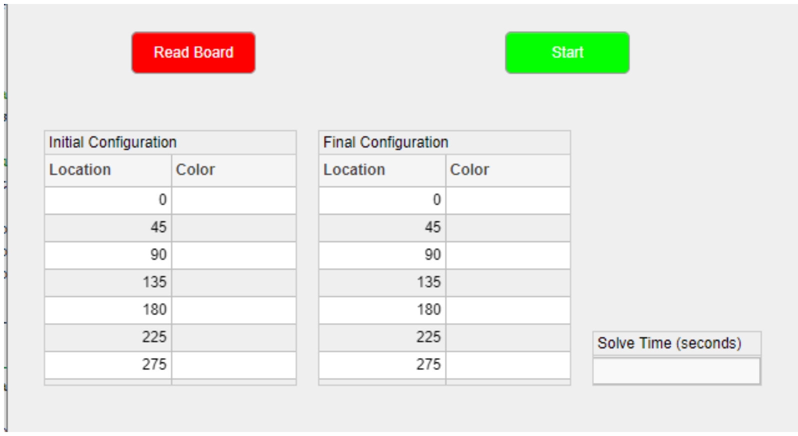

Figure 8: GUI for Motor System

The data structure created for this project is named gamestate and contains the following information:

gamestate.loc	(x,y) pixel location of every colored sticker centroid
gamestate.angle	Angle in degrees of every colored sticker
gamestate.color	Color of every sticker
gamestate.back_Orig	Background Image (game board without stickers)
gamestate.current	Original Image (game board with stickers)
gamestate.diff	Background Subtraction Image
gamestate.diffrem	Binary Background Subtraction Image

Testing of the image processing software involved taking a background picture, then placing various combinations of washer placements. These placements include placing one washer of each color on the board, placing multiple washers of the same color on the board, placing washers of the same color adjacent to each other, and across the board from each other. For each of these combinations background subtraction was performed and the result was printed to the GUI in the 'Initial Configuration' table. It was then verified whether background subtraction had succeeded or not.

To test the software for the system the simulink model was disable to prevent the motor and the magnet from turning on. Then a typical session of the system was performed. That is a background image of the board was taken followed by placing washers on the board. Then the GUI was started and the current gamestate was read. After this a new combination of washers was inserted into the 'Final Configuration' table.

The cost to implement this software was the cost of a Matlab license and $16 for the camera. An annual license costs $860 and a perpetual licence costs $2,150. The students were able to implement this license for no additional cost as the University provides an academic version of the software for use by all students while attending Clemson University. Thus the costs for the software components was $16.

System Integration

The total cost for the entire system was $122.12.

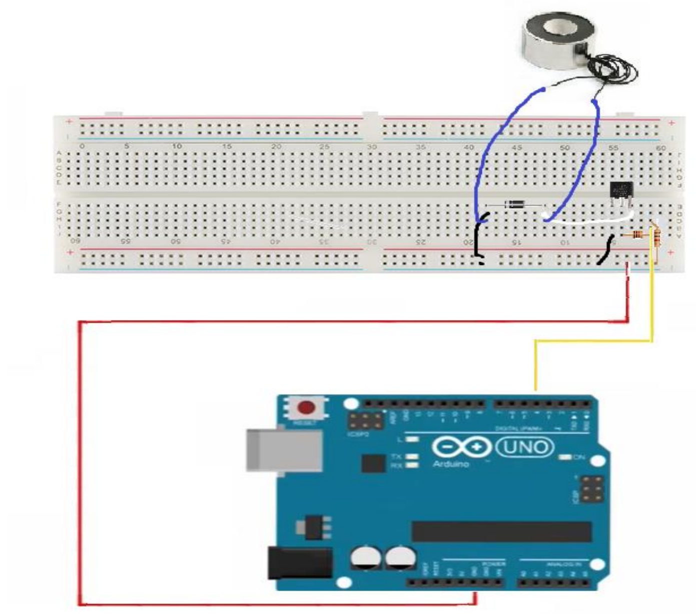

Figure 9: Electromagnet Wiring Diagram

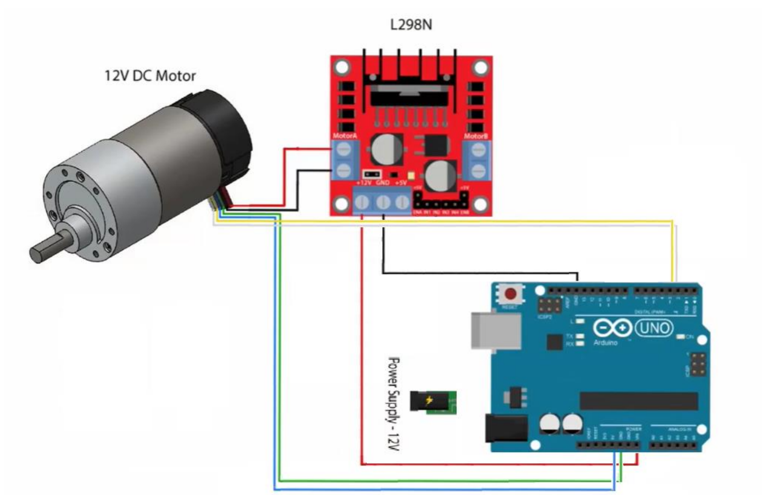

Figure 10: Motor, Motor Drive, and Arduino Wiring Diagram