Title of the Capstone Project	A Customizable Persistence of Vision (POV) Display for Advertisement
Student Name	K Mafurendi
Student Number	220702330
Date	16 November 2023

This Project Report has been submitted by the author in compliance with the requirements to obtain the credits for the module towards the qualification of:

Bachelor of Engineering Technology in Electrical Engineering

1. Declaration

This Project has not been submitted previously for qualification purposes and has been created by the author during 2023 to design and build a customizable persistence of vision display for advertisement.

16 November 2023

....

Signature Date

Contents

2. Acknowledgments

The following persons and companies are acknowledged for their valued participation that contributed to the successful completion of this project:

- Mr G, Phillips: My primary mentor for my Capstone project, year 2023. He guided me throughout the entire course of the project. I also used his lecture notes from Computer Systems 3.
- Mr I, Bambiso: My secondary mentor for my Capstone project, year 2023. Provided guidance along the course of the project.
- Mr M, Wynter: Lecturer for the Capstone project module. Taught me about project management in the Research and Project Management module.
- MANTECH: supplied most of the components I used for my project.
- Plastic Concepts: supplied with the Perspex sheets that I used to house my project as well as laser cutting of the sheets.
- Mr E, Chekure: Facilitated component orders for my capstone project.
- Mr R, Ehlers: assisted with 3D printing of my Veroboard holder for my POV display. Also assisted me with technical information about the STM32 NUCLEO microcontroller whenever I encountered faults.
- Mr N, Manene: Was always available to provide project components.
- Mr F, Adlam: Taught me about different control methods in Automation Systems (including PID, PI etc)
- Mr S, Mtakati: Assisted with ideas for motor control.
- Mr T, Lalla: provided with mechanical assistance, drilling equipment at the Mechanical lab. Also referred me to Plastic Concepts for my Perspex supply and laser cutting.

3. Introduction to the Topic

3.1. What is Persistence of Vision

Perhaps, the most common example of persistence of vision is what occurs when one stares at the sun directly. Even after looking away, an afterimage of the sun remains in sight. That happens because there is a delay between the time an image is captured on the retina of the eye and the time the same image is processed in the brain. [1] Another example is the illusion of continuous motion that we experience when watching movies. The movie is made up of many still frames which are then projected on the screen at a very high rate of 24 frames per second or more. Due to the persistence of vision phenomenon we tend to perceive the frames as continuous motion. [2] [3] Therefore, persistence of vision is an optical illusion in which the eye retains for approximately 1/16th of a second an afterimage for a brief moment after the object is out of view. [4] [5]

3.2. Persistence of Vision display

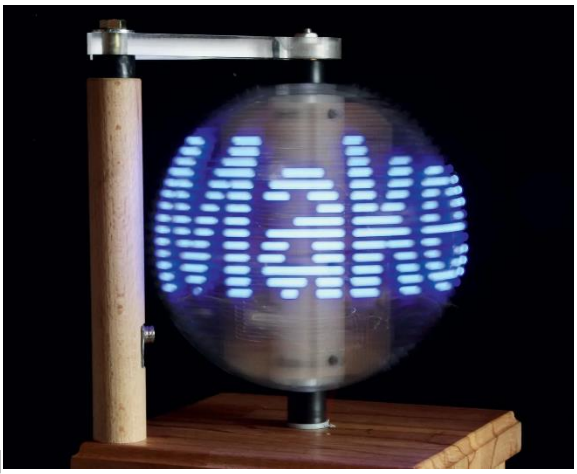
[6]

Figure 1: POV display in action.

Persistence of Vision (POV) display systems are based on the natural phenomenon described above. Usually, a row of flashing LED lights or a different light source is made to move rapidly on a trajectory which is linear or circular. [4]This can be achieved by mounting the LED row onto a frame which is then attached to a stable motor to provide rapid motion. The rotation can be in a two-dimensional or a three-dimensional space depending on the configuration of the rotating structure. [4] Through software, characters, images and even videos are made to display as if they were suspended in air when the LED strip starts to rotate. [7]The result is a stunning aesthetic display which can be useful for various applications ranging from advertisements to safety signs to entertainment. Compared to "traditional displays" POV displays come with a handful of advantages:

- Durability - most traditional display systems are fragile, but POV displays can be quite rugged hence durable.
- Power consumption – POV displays consume relatively low electrical power, they can be made to run on batteries. [1]

- Viewing angle – for most traditional displays, the viewing angle is less than 180 degrees, a POV display can be designed to offer 360 degrees view hence allowing for more viewers.

However, due to limited resolution compared to traditional displays, a lot of effort must be made to avoid blurry displays or pixilated images. Also, since POV displays rely so much on the persistence of vision illusion, very high speed needs to be achieved to eliminate motion blur and this can be dangerous if not properly done.

3.3. Persistence of Vision displays in advertisement.

This project focuses on POV display for advertisements. Due to their stunning aesthetic displays, POVs are ideal for grabbing the attention of by-passing potential customers and they can be used to display the following among others:

- The company logo
- Promotions
- New products

4. Engineering Problem

4.1. The Problem

Conventional advertisement tools like signboards, neon signs and posters are static. Once an ad is placed, it stays like that until it is replaced. This means that for every ad placed, a new display (a poster, or a signboard) must be made. Also, the same display cannot be customized to be used by another advertiser or to place a different advert. This is expensive and inefficient for the advertiser. Time is lost in designing a new display and if a new advertiser takes over a premise, they will have to place their own advertising tools from scratch.

4.2. The solution

Using a POV display that can communicate with a mobile device to update the displayed information would solve all these setbacks. The following can be updated easily in real time:

5. Literature Review

POV displays come in different designs. In this section, 3 of the various designs are proposed.

5.1. Possible Solution 1

5.1.1. Full color Spherical 3-D POV Display

[8]  [9] 

Figure 2: 3-D POV display showing world map.

Figure 3: LED configuration for 3-D POV display

Using a semicircular LED setup as shown in Figure 3, a spherical 3-D POV display can be achieved. [9] This POV display design was developed by E. Berlin over half a century ago. [9] An RGB LED array was used and this enables a full colour POV display. The design has got some advantages and disadvantages as discussed below.

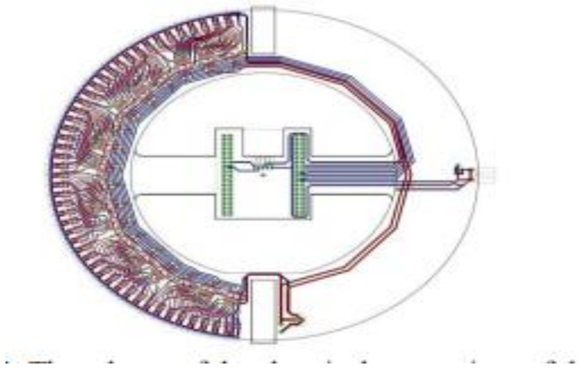
5.1.1.1. Advantages:

- Stable mechanical structure:
  Stability is a major concern when designing POV display system since the LEDs are supposed to be rapidly rotating. Using a semi-circular configuration together with the control circuitry as shown in figure 3 makes the whole structure very stable as the motor will have to be placed in a vertical orientation (which is most stable for the motor because its mass is evenly distributed in this orientation). This is further aided by the heavy base of the stationary structure. If this aspect is not well addressed, it will be difficult to produce clear displays. [10]
- Full colour display: Colour makes a huge difference between a good advert and a bad one. Much detail is lost in monochromatic displays. The use of RGB LEDs makes full colour display possible.

5.1.1.2. Disadvantages:

- Complex circuitry:

Looking at Figure 3, a less complex circuit would have been built on the stationary structure. However, that would complicate the design as it would be challenging to send control pulses to the rotating LED structure. [9] The solution of mounting all the electronic circuitry on the rotating component of the design requires more sophisticated circuitry which requires very high precision to create. Also, it comes with a higher cost, financially. The schematic for the circuit is shown below:



[9]

Figure 4: Semicircular LED design schematic

- Limited application:

Due to its spherical shape, this design is ideal for 3-D human portraits, weather map, air traffic display system or any other display that features the world map. Therefore, a spherical POV display design would not be suitable for displaying adverts for shops and other companies trying to advertise products that have nothing to do with a world map.

This design is very much ideal for an advertisement POV display even though it is relatively complex to design. To make it more versatile, instead of using a semicircular array of LEDs, they can be placed in a straight column. Instead of giving out a spherical display, the display would become cylindrical as depicted in Figure 5 below.

[11]

Figure 5: Cylindrical full color POV display.

5.2. Possible Solution 2

5.2.1. Monochromatic 2-D POV Display

This is the most common and perhaps the most traditional POV display design. LEDs of the same colour are used. The LED column is mounted either horizontally or vertically and it is made to rotate in a circular path in one plane. [4] The images below illustrate this:

[12]

[12]

Figure 7: 2-D Monochromatic POV display.

Figure 6: 2-D POV display design mechanical structure.

5.2.1.1. Advantages

- Unsophisticated design: this type of design does not require a complicated mechanical structure. The LED strip should only be set up to rotate in a 2-dimensional plane about a certain axis. Of course, the base of the structure should be a bit heavier to stabilise the whole structure when rotation starts. Since there will be no need for complex colour manipulation, the software design also comes relatively simpler than a full colour display.
- Economic: because of the simple mechanical structure, relatively simple circuitry will get the job done. Also, monochromatic LEDs are cheaper than RGB ones. Overall, a 2-D POV display is relatively cheaper to build. A very powerful microcontroller (which is expensive) will not be necessary as cheaper ones would still get the job done.
- Lower electric power consumption: Single colour LEDs consume low electric power. Cheaper microcontrollers consume low power as well.

5.2.1.2. Disadvantages

- Limited viewing angle: the viewing angle for 2-D displays is limited to less than 180 degrees. Therefore, a limited number of clients can be catered for at the same time.
- Missing colour detail: advertising using the same colour for displaying everything (text, animations, images) can be inaccurate and misleading.
- Less versatile: the colour limitation makes 2-D POV displays only ideal for displaying text characters for adverting or some simple attention-grabbing animations. Otherwise, video and images cannot be accurately displayed using such a display.

To make it idea for an advertising POV display the following changes can be implemented:

- Use RGB LEDs to enable full colour display.

- Use a more powerful microcontroller that can handle image and animation processing e.g., a Raspberry Pi.
- Constructing a very stable base for the entire structure to avoid motion blur which might distort the display.

5.3. Possible Solution 3

5.3.1. Cylindrical POV Display

[1] 

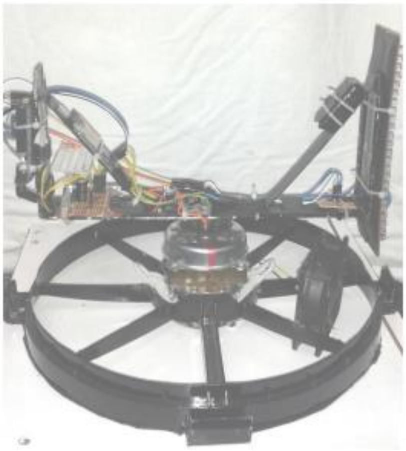
[1] 

Figure 9: Cylindrical POV display prototype.

Figure 8: Cylindrical POV display in action.

Like a spherical POV display, this design will display a virtual screen that gives a 3-D impression. [1] From the prototype in Figure 8, it is relatively easy to create a stable base for the fixed frame, hence chances of motion blur or distortion due to instability is minimal.

5.3.1.1. Features

- Gives a 3-D display impression.
- Basically, liken to a spherical POV display in many aspects but with less design sophistication.

6. Design

6.1. Hardware design

After studying various possible solutions, I eventually came up with my own chosen solution design. Figure 10 below shows the visual overview of the project design. Figure 11 is the block diagram for the same design. The design is supposed to function in the following manner:

The user enters text into a textbox in a graphical user interface (GUI) on the laptop. Before this they have to establish a connection between the laptop and the display hardware. They select a colour that they want the text to be displayed in. They have a chance to preview if the text is as they desire (looking at the colour, spellings, etc.). They then send the data to the display hardware wirelessly. A Bluetooth connection will be used for this purpose. Once the data is received on the other end, the LED columns begin to flash out the received text. The microcontroller (NUCLEO F411Re) receives the data sent via a Bluetooth module (HC-06). The microcontroller decodes the data and converts it into arrays of 1s and 0s (bits). These bits are then sent to the LED columns using Serial In, Parallel Out (SIPO) shift registers (74HC595). And then the LEDs will flash accordingly. When the flashing happens while the motor is rotating at the right speed of rotation the received message should be displayed clearly due to the phenomenon of persistence of vision.

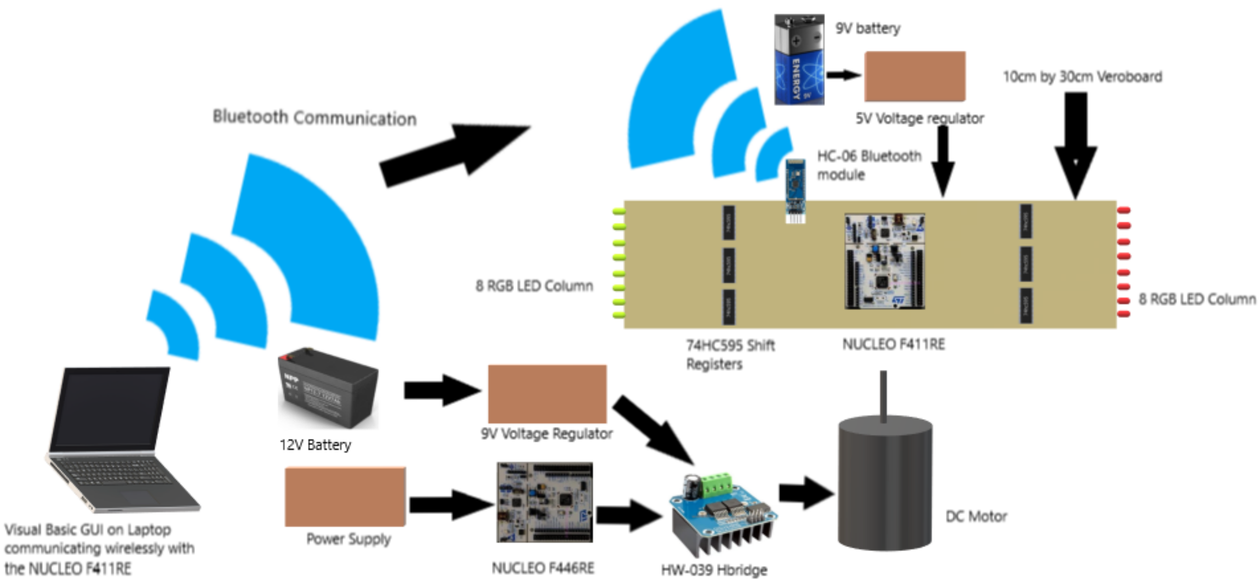


Figure 10: 2D Project design overview (the 2D diagram was rendered in the Paint 3D Windows Software)

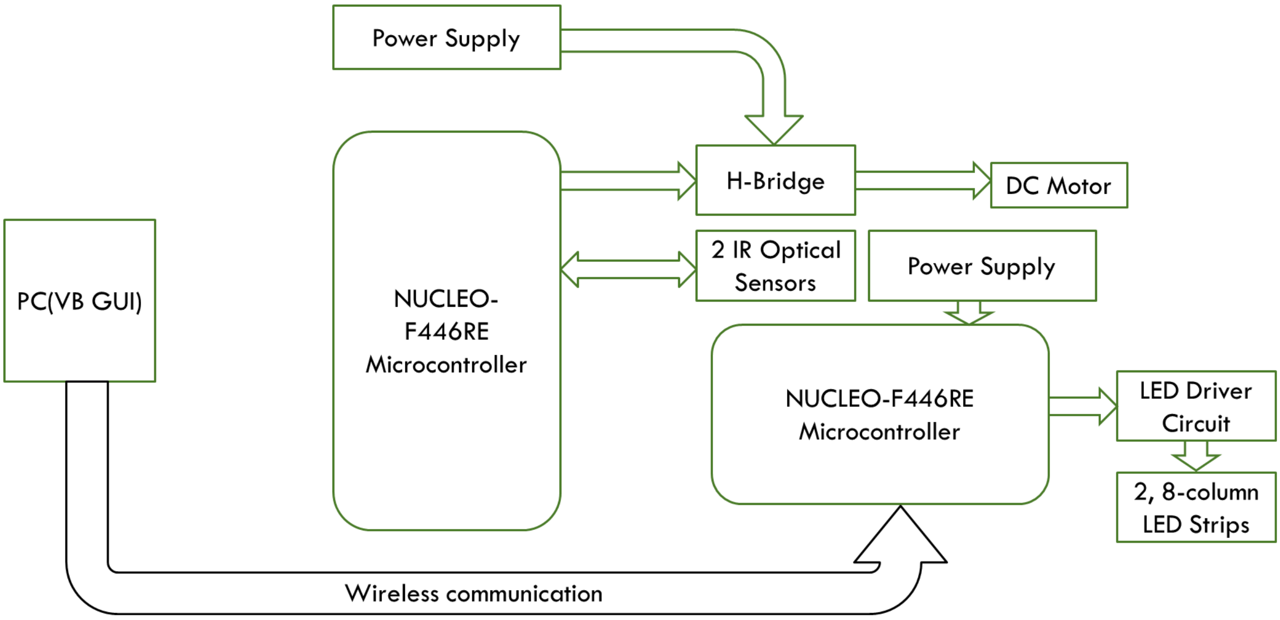


Figure 11: Project overview Block diagram

Basically, these are all the components of my project's hardware. After building all the pieces, I would then design a mechanical frame to house everything and hold all the components in their rightful positions. The following subsections dive deep into how each component was designed and developed.

6.1.1. Electrical Circuitry

My project design has 3 major electrical circuits. The LED circuit, which contains and controls the RGB LEDs, then there is a motor circuit which contains and drives a Dc motor to provide the persistence of vision illusion and the wireless communication circuit which connect a Bluetooth module to the display. There are also minor voltage regulation circuits included in the motor and LED circuits.

6.1.1.1. LED Circuit

I used the KiCad software to design the circuitry. The main reason for the choice of the design software was that it had most of the exact components I wanted to use for my project and for those that were missing it was easy to import footprints and symbols from online sources unlike with Multisim that I was used to from my first year. The learning curve was very easy as well. I found the footprints for the 74HC595 shift register at SnapEDA.com, an online CAD library.

6.1.1.2. Calculations

The only calculations for this circuit were to determine the resistor values to limit the current flowing to the RGB LEDs. From the datasheet the Absolute maximum forward current for the LEDs I used is 20mA. However, each colour pin (red, green, blue) has its own typical operating forward voltage, and the values are as follows:

Red – 2.0V

Green – 3.2V

Blue – 3.1V

Using Ohm's law, the resistance for each pin is therefore: R = V/I, where R – resistance in Ohms, V is the potential difference across the LED, and I is the forward current. V is calculated as the difference between the microcontroller output voltage and the led pin forward voltage i.e., 5V – Vf.

For:

Red, rR = (5V – 2V)/ 20mA

rR = 150 Ohm.

For:

Green, gR = (5V – 3.2)/20mA

gR = 90 Ohm.

For:

Blue, bR = (5V – 3.1)/ 20mA

bR = 95 Ohm.

However, for gR and bR these are not standard resistor values. I ended up using the 100 Ohm resistor for both pins.

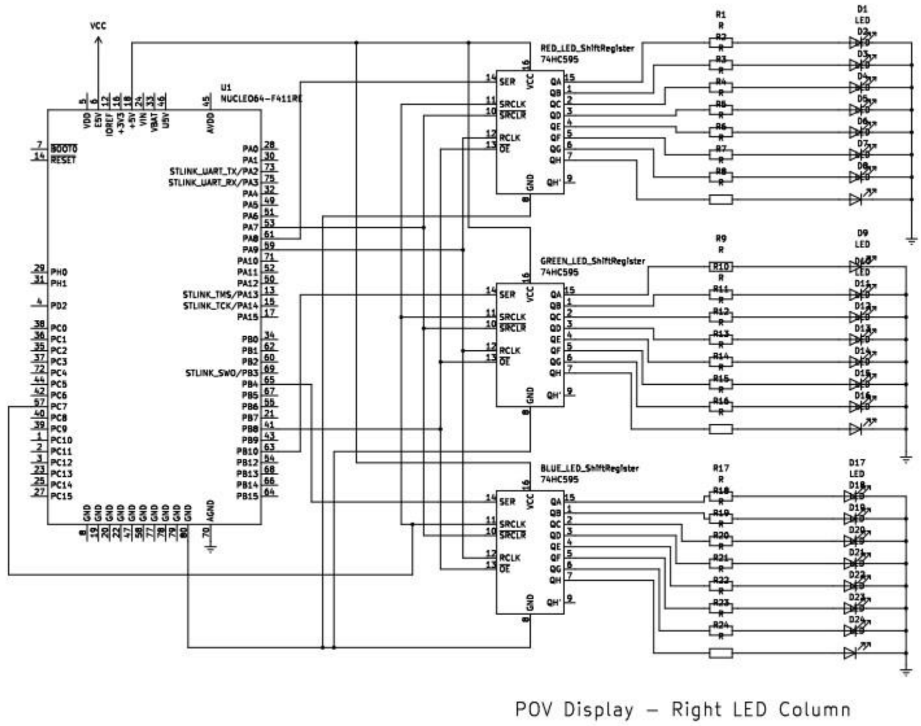


Figure 12: LED Circuit Design in KiCad – Right LED Column only

Figure 12 shows the LED circuit design in KiCad. This is only the right wing. The left wing is exactly the same, only in the opposite direction. The shift registers were not cascaded. I decided to use an individual shift register to control pins of the same colour for all the LED. On my Kicad design I used monochrome LEDs for better layout clarity instead of the 4 pin RGB LEDs. In this design, the NUCLEO F411RE microcontroller is powered 5V power supply (circuit design shown in the ‘Power Circuitry’ section). All the shift registers are powered from the microcontroller. Figures 13-17 below show the physical circuit from the bread board trials to soldering all the components on the Vero board.

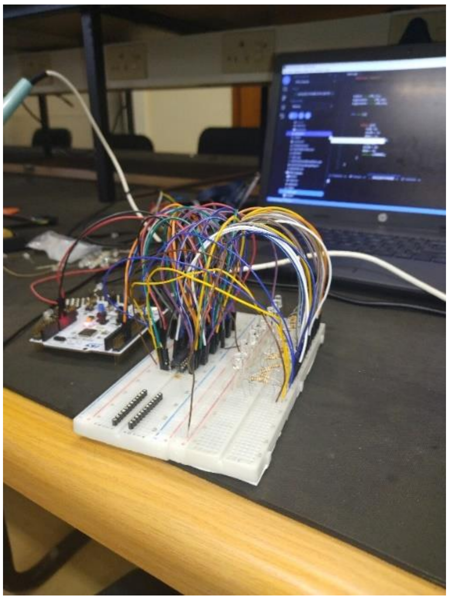


Figure 13: Bread layout for the LED circuit.

Figure 14: Soldering components on the Vero board.

.

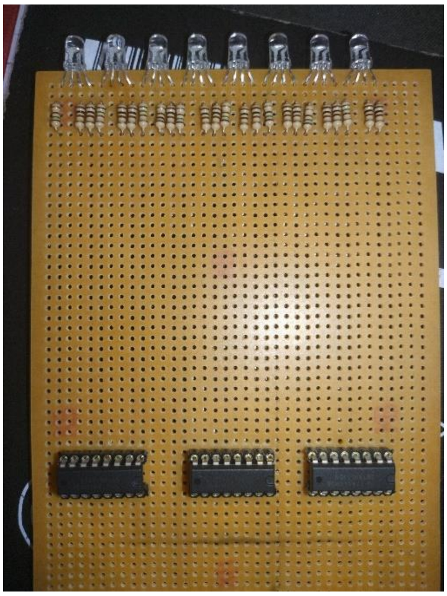


Figure 15: LED circuit soldering.

Figure 16: LED circuit soldering.

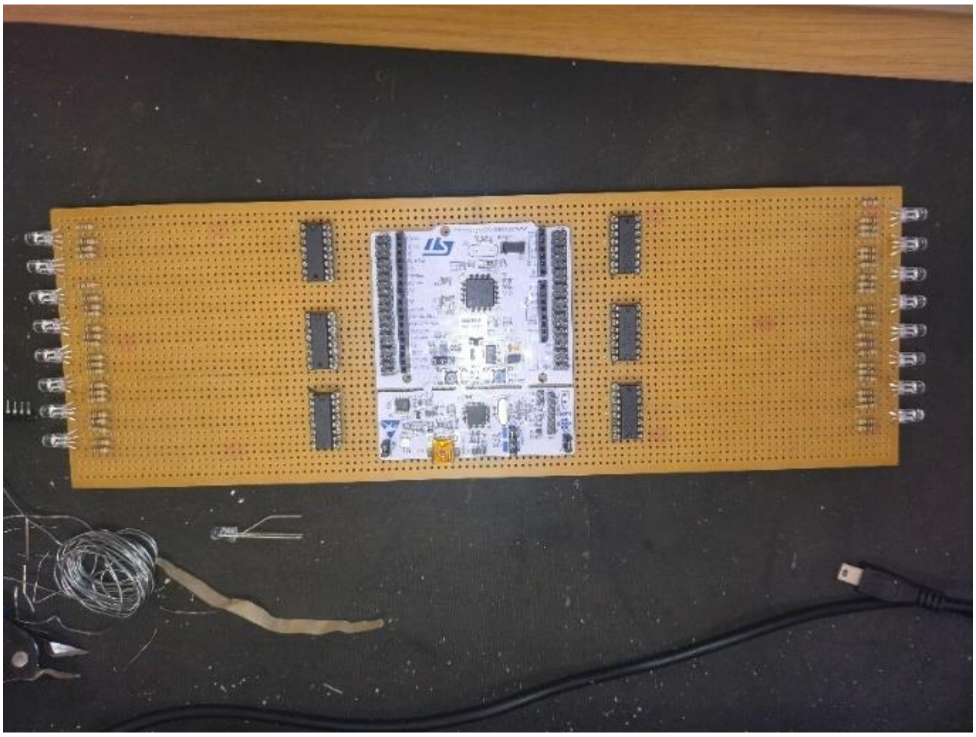


Figure 17: Completely soldered LED circuit. (NB: The microcontroller is not soldered to the Vero board.)

6.1.1.3. Motor Circuit

To provide the rotational motion, a DC brushed motor was used (part number DRS-750PM-8522-F/90).

Figure 18: 9.6V brushed DC motor.

The electrical and mechanical datasheets for the motor are shown in the appendix. The ideal motor control for a POV display project should include a dc motor driver, feedback mechanism, and PID control. This is because a high degree of precision is required in this project to provide syncronisation between the flashing of the LEDs and the position of the rotor from a reference point. [9] At the start of my project, I intended to use a brushless DC (BLDC) motor (figure 19 below). This would offer more advantages than a conventional DC motor for the following reasons:

- Durability - Because a brushless Dc motor has no brushes which can wear out under continuous rotational stress, a BLDC motor does not have brushes therefore, it can sustain continuous rotations for extended periods of time. A POV display would most likely have to be used for long periods. A BLDC motor would be the most appropriate choice.
- Built-in sensors – the BLDC I intended to use had built in position sensors therefore it would be much easier to apply the motor control precises using feedback from the sensors.
- A dedicated motor driver – There are dedicated ICs and modules designed to drive BLDC motors without having to use an h-bridge or having to design some complex circuitry.

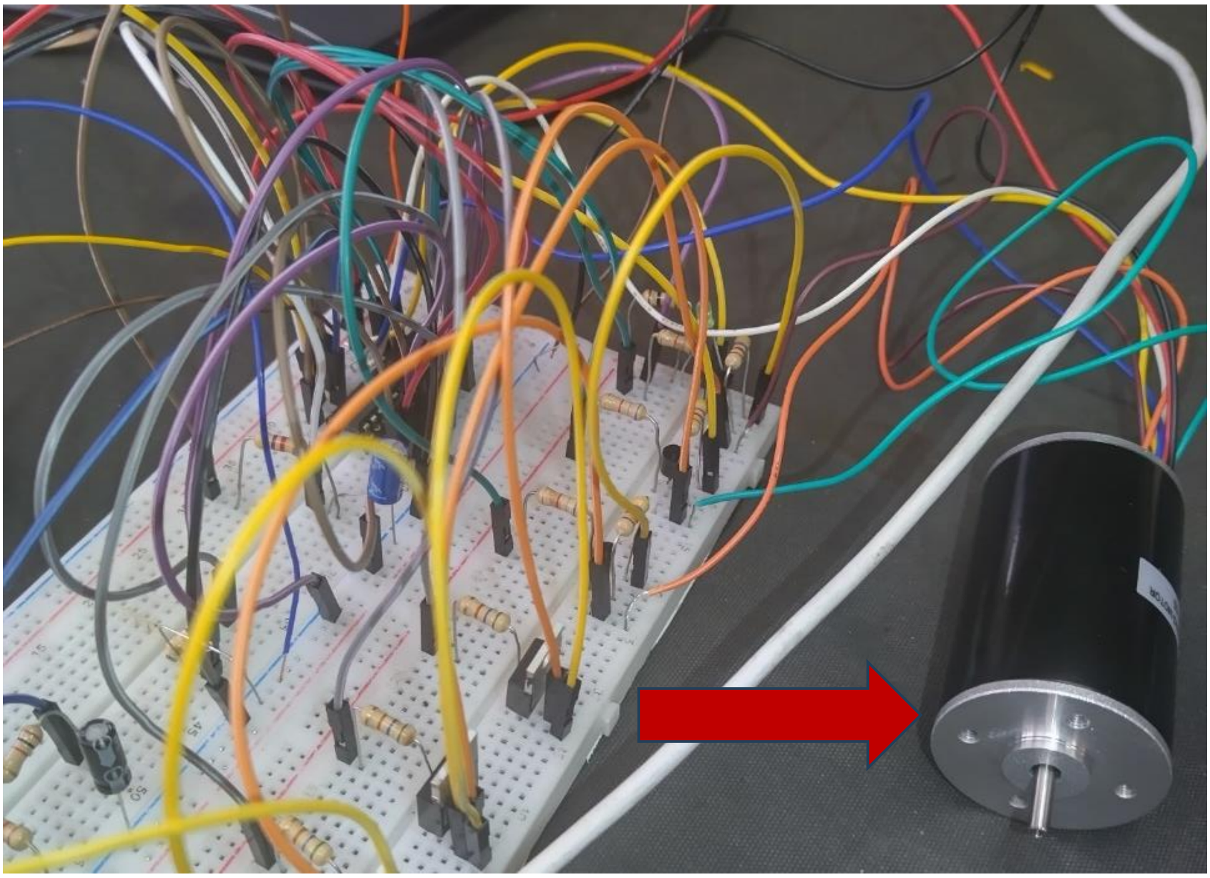


Figure 19: The BLDC motor I intended to use. Datasheet in the appendix.

For the motor driver I ordered an mc33035 IC shown in figure 20 below. This is a 24 pin IC which drives BLDC motors as shown in the schematic diagram in figure 21 below. It was only during the circuit design process that I realized that the BLDC motor that I had ordered was not compatible with this IC. Because of time constraints, I had to resort to the brushed DC motor shown inn figure 18 above. Again, at this point in time I could not implement all the motor control requirements for a POV display project. I had to design and work with a basic DC motor control circuit with an H-bridge, as shown in figure 22 below.

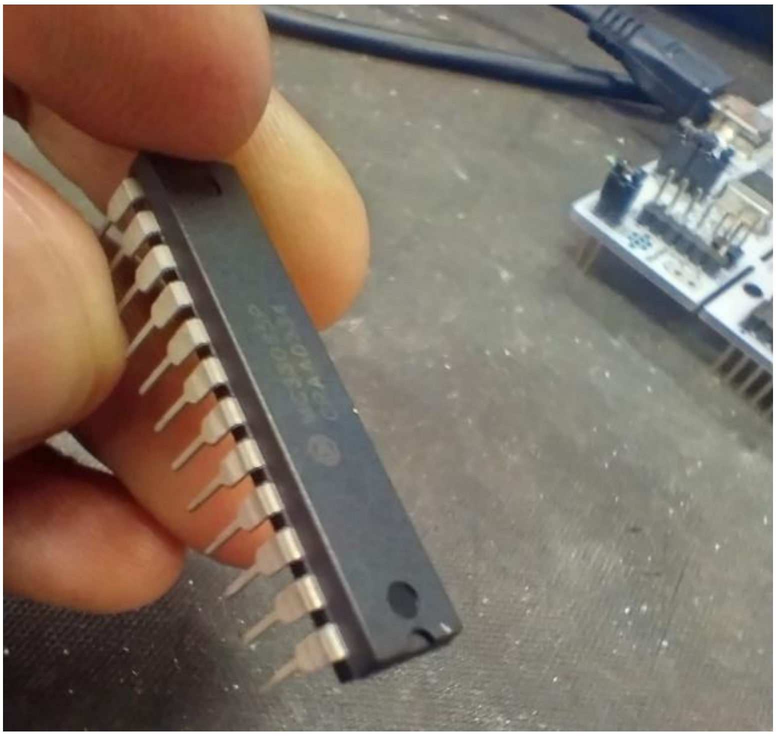
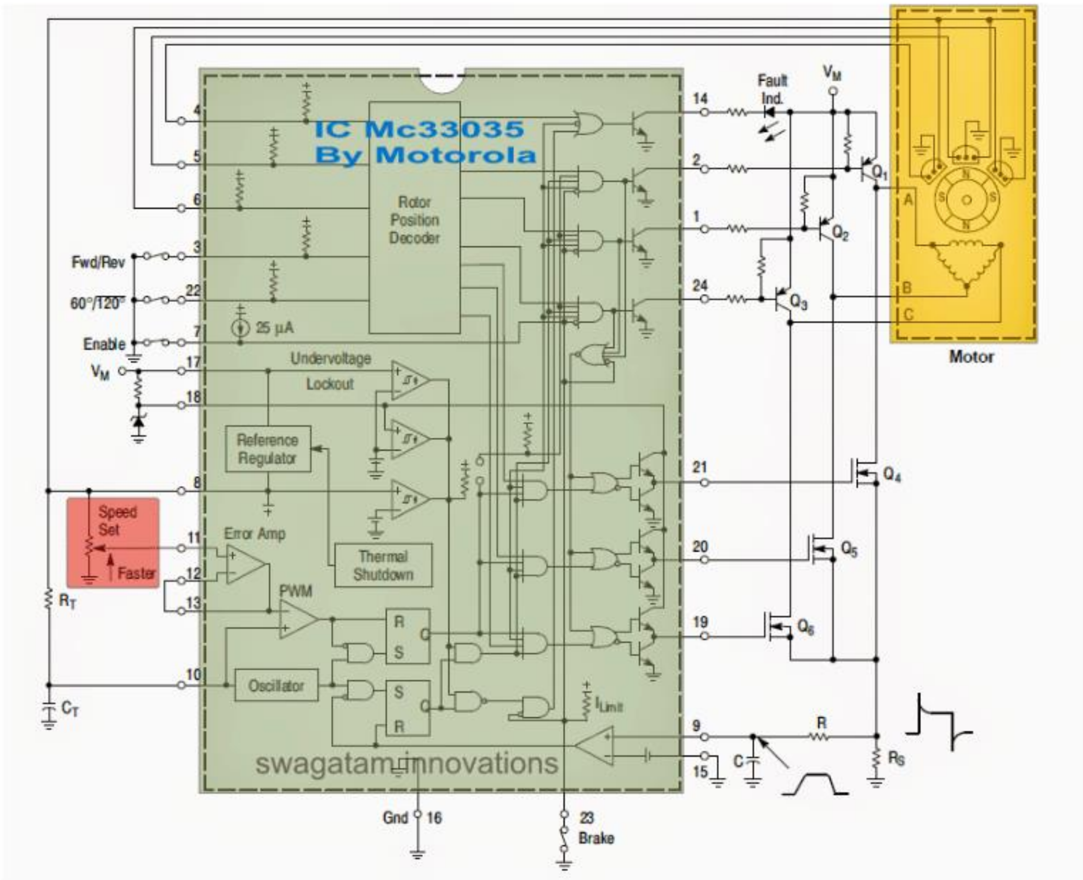


Figure 20: mc33035 BLDC motor driver IC. Datasheet in the appendix



[13]

Figure 21: MC33035 IC wiring diagram for a BLDC motor

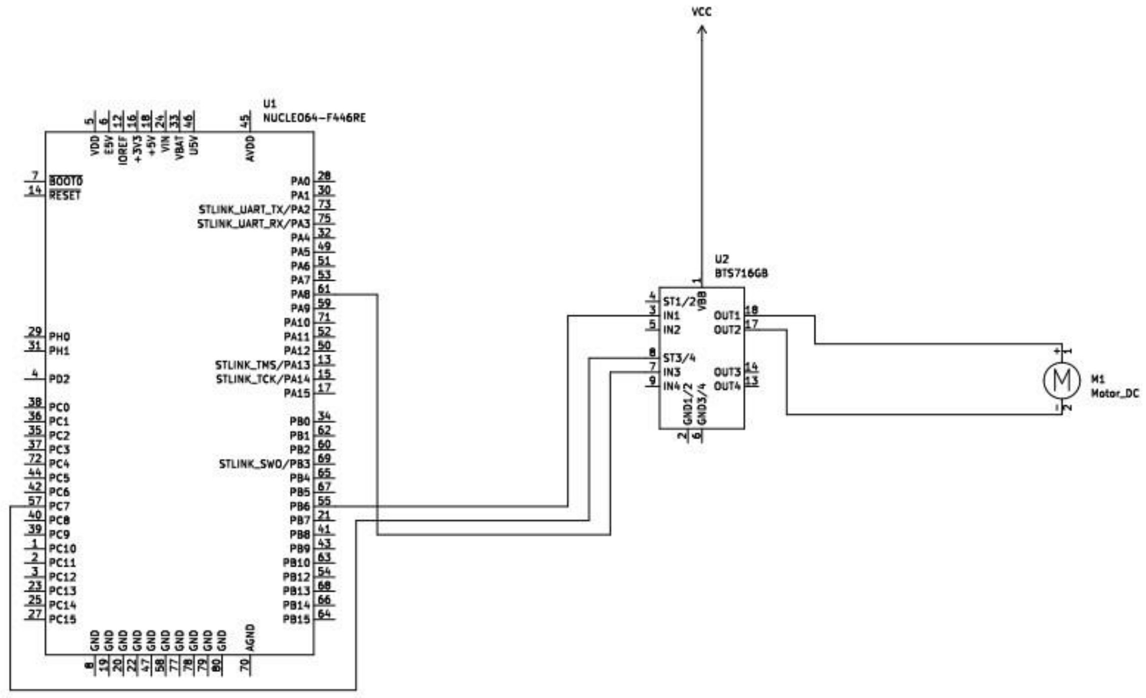


Figure 22: Kicad Motor control circuit schematic diagram.

I used the NUCLEO F446RE microcontroller to send PWM and digital signals to the BTS7690 H-bridge. The PWM signal controls the motor speed, the two digital signals would enable rotation in the two directions. Only one of them must be enabled (5V) at a time for the motor to turn either left or right. The BTS7690 can work with up to 30V maximum, therefore, any power supplier value from about 10V to 30V must be adequate to drive the motor. The electrical and mechanical information datasheets are shown in the appendix.

After connection, the circuit looks as shown in figure 23 below.

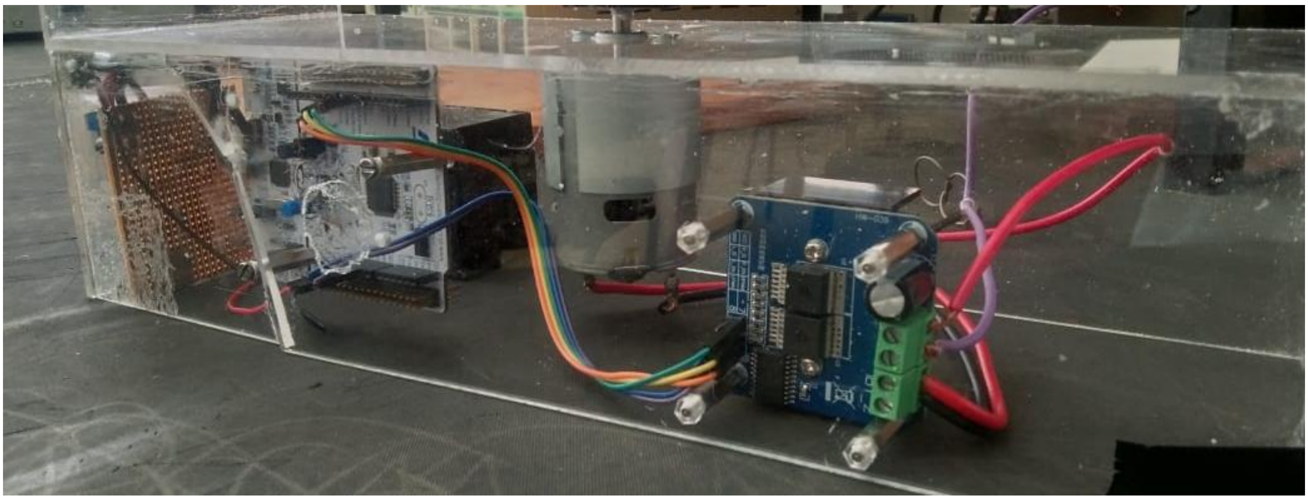


Figure 23: The physical motor control Circuit

6.1.1.4. Wireless Connection

For the wireless communication between the graphical user interface and the main microcontroller, I used a Bluetooth connection through an HC-06 module. The receiver pin (RX) of the module is connected to the PB_6 transmit pin (TX) of the NUCLEO F411RE while the transmit pin of the module is connected to the PA_10 receiver pin of the microcontroller. PB_6 and PA_10 form the first UART channel of the NUCLEO F411RE. The Bluetooth module is powered by the microcontroller.

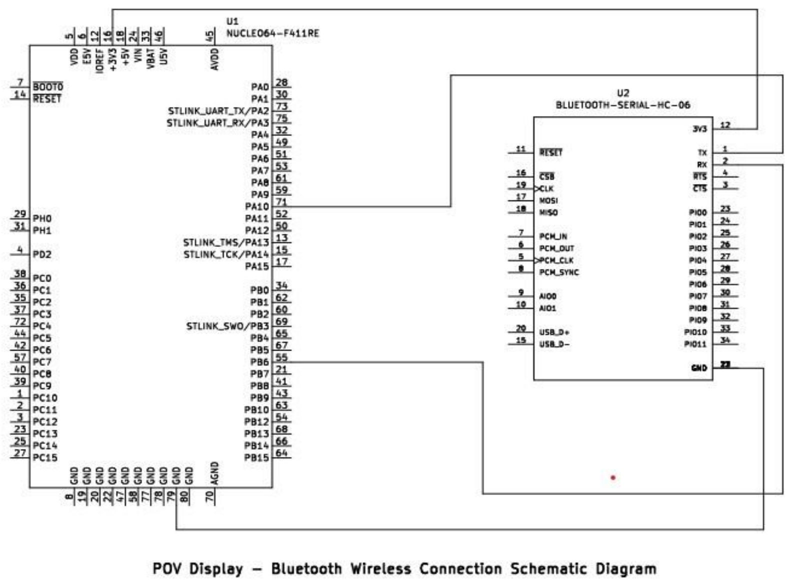


Figure 24: HC_06 Bluetooth module hardware connection schematic in KiCad

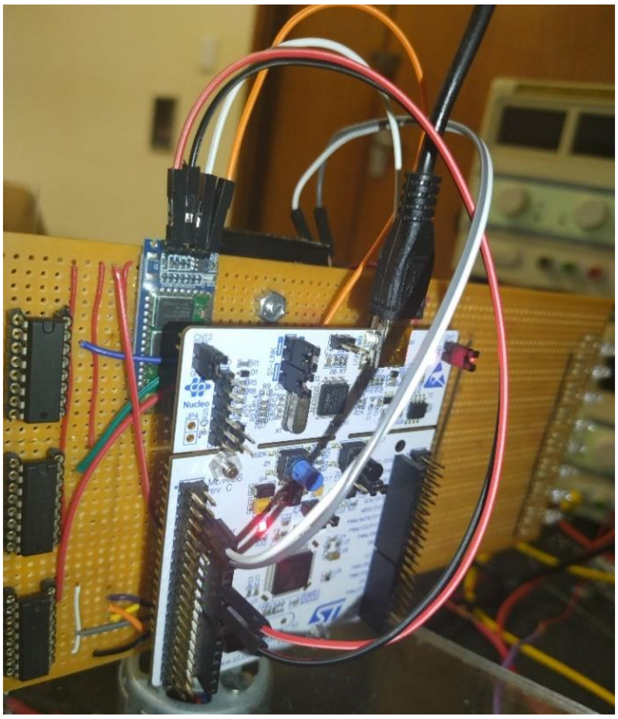
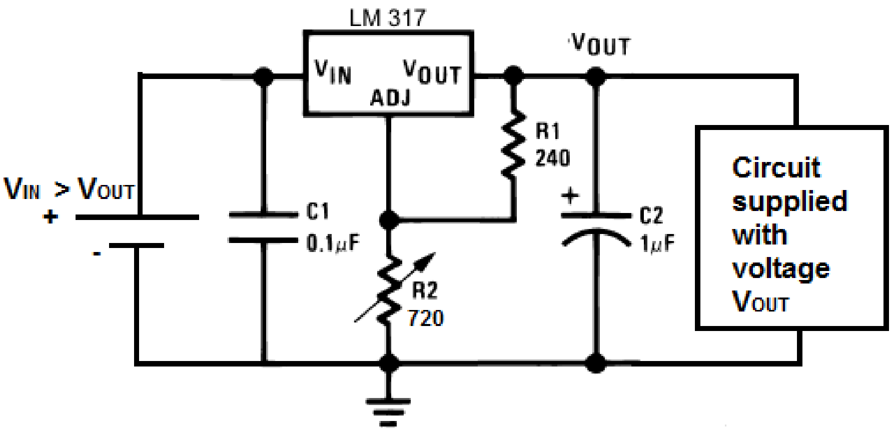


Figure 25: HC_06 module connection to the NUCLEO board

6.1.1.5. Power Supply

Due to the rotation required to create the POV illusion, all the components on the rotor required a standalone power supply. Therefore, a 9V battery and a 5V voltage regulating circuit had to be included. An LM317 voltage regulator was used for this application. The following circuit design was implemented to provide a 5V supply for the microcontroller.



[13]

Figure 26: 5V Voltage regulator circuit schematic

However, instead of a 240 Ohm resistor for R1 I used a 220 Ohm, and a 680 Ohm fixed resistor was used instead of a 720 Ohm. For C1, I used 1 uF and for C2 I used 10 uF. The resultant output voltage for the circuit was then 5.3V.

The physical circuit for the 5V voltage regulator is shown below:

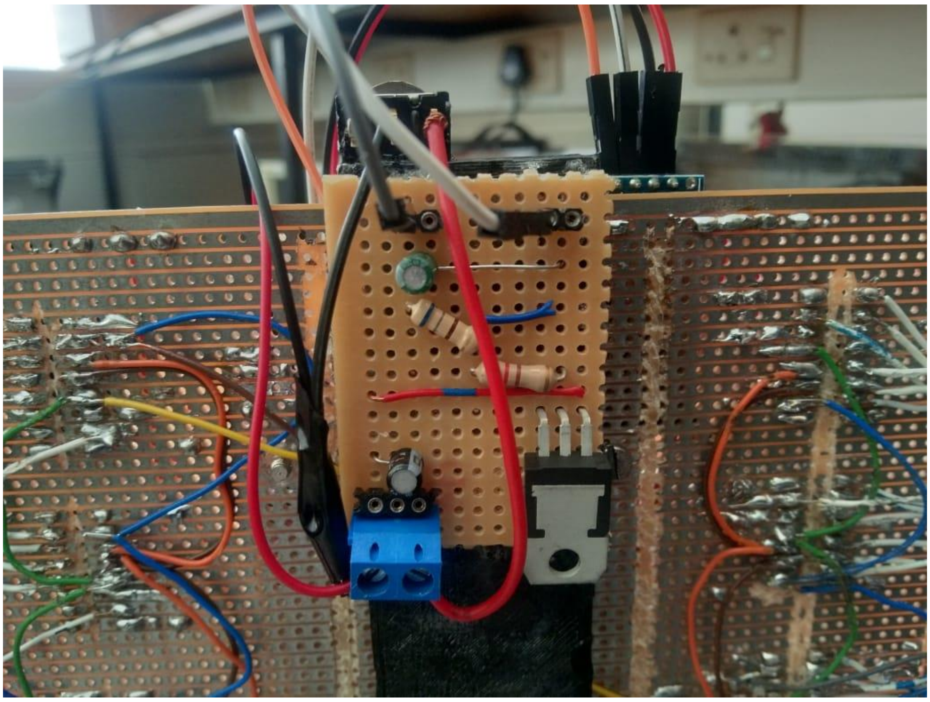


Figure 27: 5V Voltage regulator circuit mounted on POV display.

6.1.2. Mechanical Structure

To hold all the components in place, a mechanical structure was a requirement. I used SolidWorks to design the two structures, one to hold the rotating Vero board and the main one to house everything else. The Vero board holder was then 3D printed and the main frame was the constructed using Perspex sheets. The sheets were precisely cut by Plastics Concepts.

6.1.2.1. Perspex frame

The initial structure design is shown in the following figure.

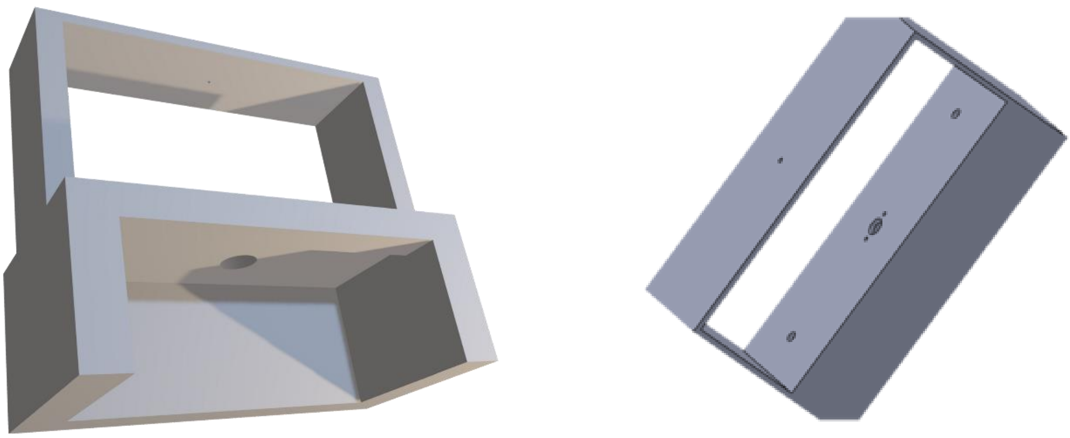


Figure 28: Mechanical structure design in SolidWorks

The design allows the motor to be fitted at the middle of the structure and the rotor to sit attached to the motor axle while rotating within a semi protected space. The components for the motor control circuit would all fit into the cavity underneath the structure and all LED control components would be on the space above. The resultant Perspex structure is shown below.

Figure 29: Perspex structure

The 'roof' was removed to reduce friction against the rotor which would otherwise minimize the motor speed. The overall dimensions of the structure are 32cm by 10cm by 20cm.

6.1.2.2. Veroboard holder

Similarly, a part to hold the entire Vero board was designed in SolidWorks. This however was 3D printed. The SolidWorks design was first converted to an STL file which is the format for 3D printing and was printed in the department.

The SolidWorks design and the resultant structure are shown in the following figure.

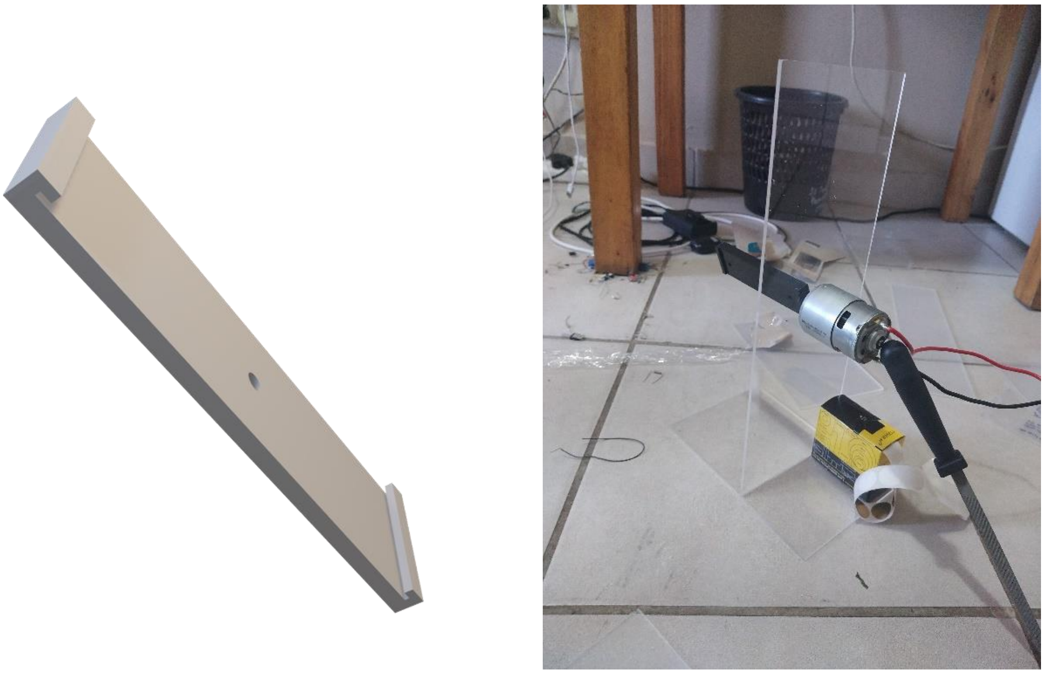


Figure 30: Vero board holder design

The holder had to be strong enough to endure the weight of the components while rotating at a high speed. 3D printing was then a good approach because 3D printed parts are usually compact and have high strength.

6.2. Graphic User Interface Design

The project is entirely operated from a Graphical User Interface. I used Visual Studio to design the GUI in Visual Basic. The GUI starts off with a splash screen that shows the application name, version, and copyright information. Then, a home page pops up. The home page consists of a check button to connect to the UART portal at the top left corner. Here, a user can connect or disconnect from the display when they want to send or stop sending data respectively. Then there are two textboxes where the user can input text characters, one for each LED column. The maximum input length is kept at 6 characters on both text boxes. To choose character colour, there are two radio button sections, one for each LED column. The available colours for selection are red, green, blue, and disco which is a random mix of the primary colours. The preview button allows for spelling checks, colour confirmation and basically previewing before sending the data to the display. Whatever the user inputs, is converted to and saved as a graphic image on the laptop and is then display as a displayed as a graphic picture in the black picture box for preview. I watched YouTube tutorials to learn about graphics manipulation. Once the user is happy with the preview a 'send to display' button transfers the data to the display via Bluetooth.

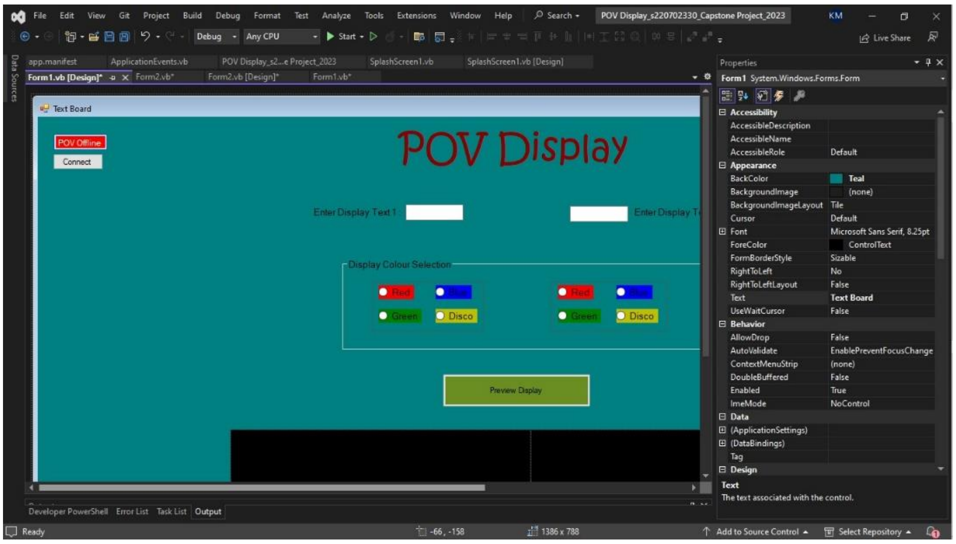


Figure 31: GUI Home page design in Visual Studio

Figure 32: Full screen of Home Page

If the GUI cannot connect to the display for some reason or a technician would like to make technical adjustments, they will press the factory settings button. This opens a settings window where one can adjust COM port numbers or baud rate. After adjusting the settings, a ‘Done’ button restores the home page to resume normal operation.

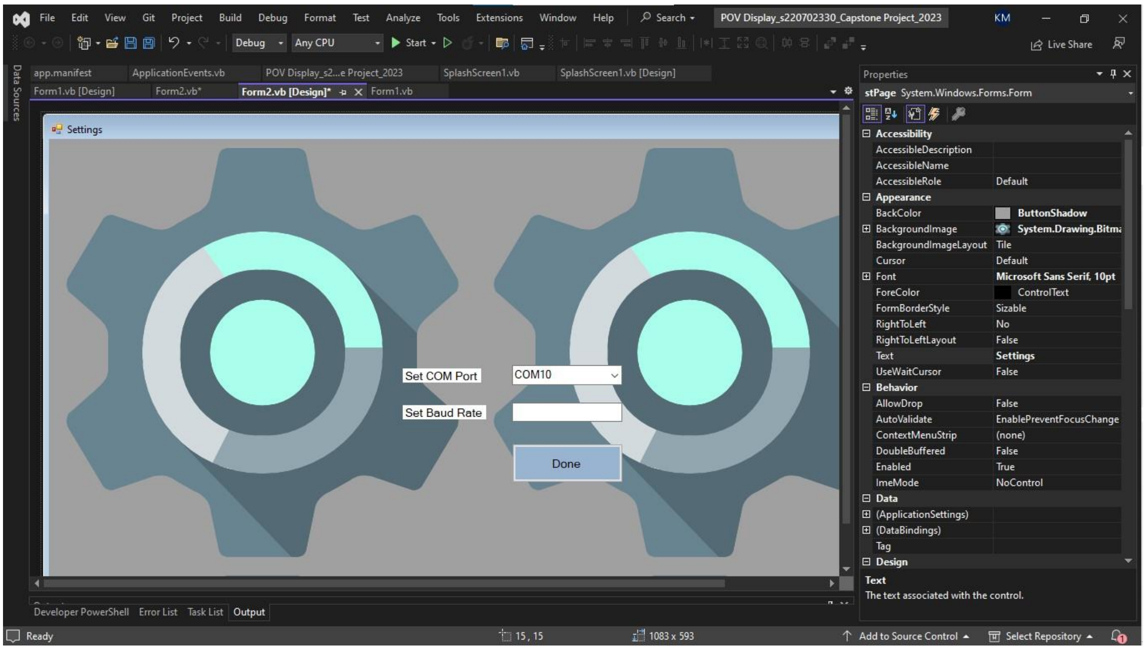


Figure 33: GUI Settings Page design in Visual Studio

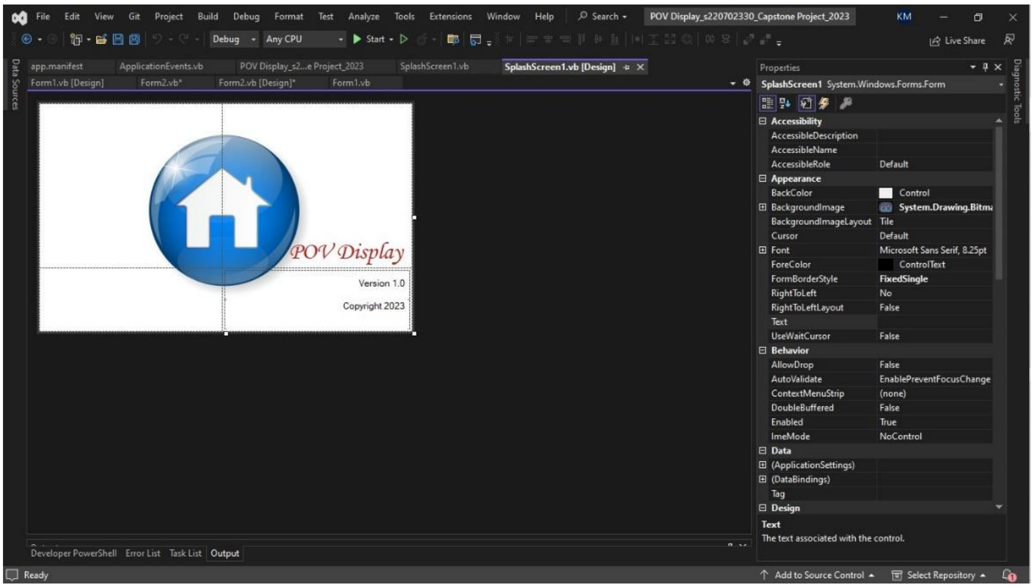


Figure 34: GUI Splash screen design in Visual Studio

For this GUI to connect to the display, the user must pair their laptop with the Bluetooth module on the display first. It is discoverable as HC-06 and the password is 1234. Two COM ports are then created automatically on the laptop and one of them allows the user to transmit data between the two devices.

In the code the UART protocol to send data is:

!TEX (text_box1_input) (text_box2_input) (colour_number_for_text_box1) (colour_number_for_text_box2)#.

Please note that the brackets are not part of the protocol but are just used to separate data.

6.3. Control Software Design

The main functionality of this project is provided by the mbed code running on a NUCLEO F411RE microcontroller.

The code is grouped into the following sections:

UART communication

Message manipulation

Shift register control.

UART communication – Using this protocol, "**!TEX (text_box1_input) (text_box2_input) (colour_number_for_text_box1) (colour_number_for_text_box2)#",** the microcontroller receives five data parameters from the GUI via a Bluetooth module.

Message manipulation – Two functions one for each LED column, AlphaL and AlphaR then receive the two words and their color numbers. They then use switch case functions to convert the input into 1s and 0s which then get shifted by the shift registers to the LEDs for display.

Shift register control – I download an already made library for the control of the 74HC595 shift register and then I modified it to suit my specific application. [14]

Another portion of mbed code runs on a separate NUCLEO F466RE to drive the DC motor via an H-bridge. The simple code set the motor duty cycle using a pulse width signal as well as the period which is 25kHz (code snippet in the appendix).

7. Testing and Analysis of Results

7.1. Graphical User Testing

When testing, the GUI works exactly as expected. I am able to establish communication with the display. Data is being sent to the display even while the rotor is in rotation.

Basically, there are two components to my graphical user interface (GUI):

1. The Bluetooth connection component

Once the display device has been switched on, I must ensure that the Bluetooth module is powered. This is shown by a rapidly flashing red LED as shown in figure 30 below. If the module is on and the display is being used with a particular computer for the first time the module must be paired first. This is done in the computer settings. If using Windows 10 or a similar version go to settings >> devices >> Bluetooth >> Add Device >> Bluetooth Device. If the module is on and properly functional the module will show as HC 06. After selecting the device input password 1234 and the module should be paired. From here the Bluetooth module is ready to connect to the computer. When I press the “connect” button on the GUI, if the port to which my module is paired to the laptop has not been selected in the GUI, a message box pops up as shown in figure 31 prompting the user to try another serial port. To make this adjustment, I must use the “Factory settings” button which opens a new window, specifically dedicated to technical adjustments for serial communication figure 32.

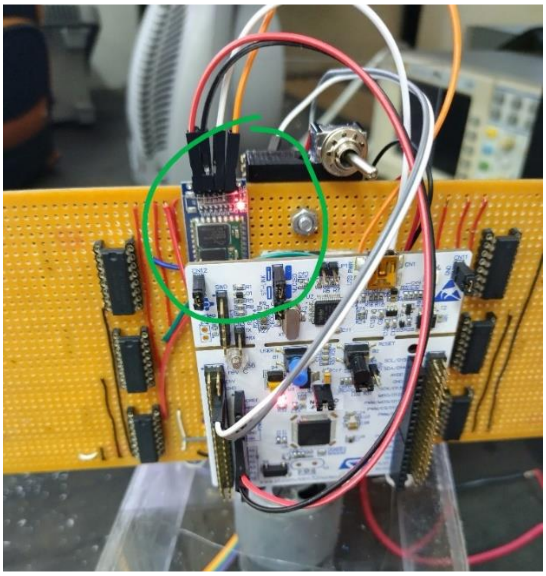


Figure 35: The red LED (in green circle) shows that the Bluetooth module is on.

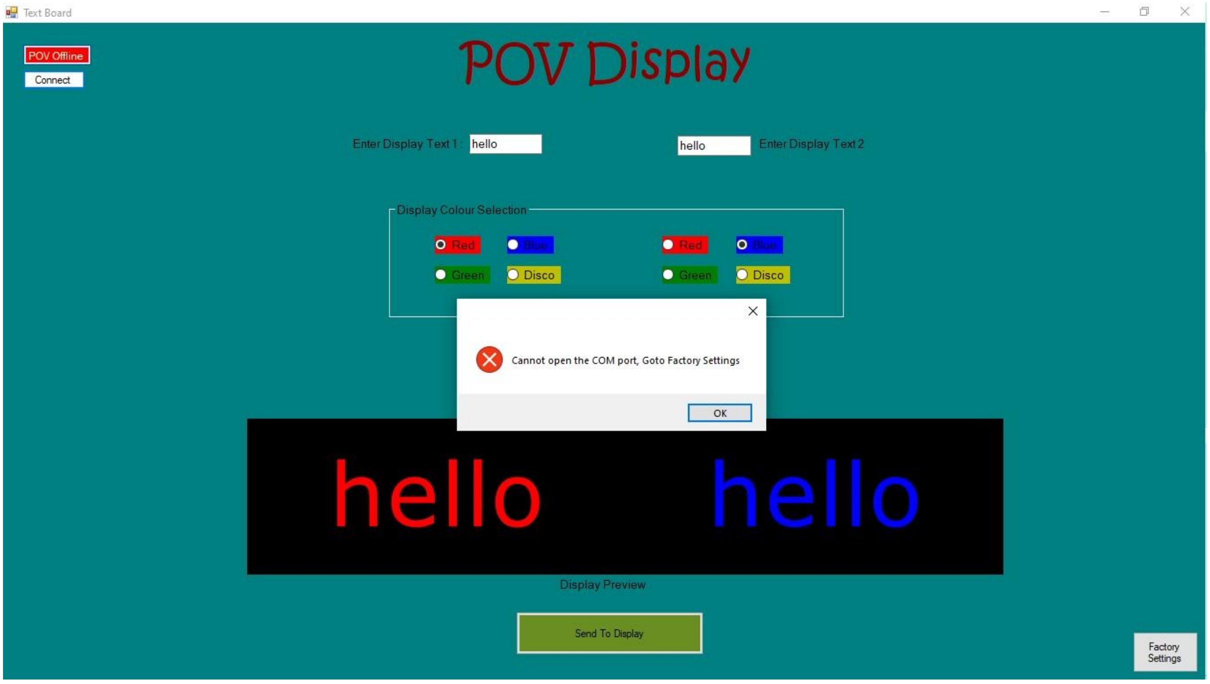


Figure 36: Message box prompting user to change COM port

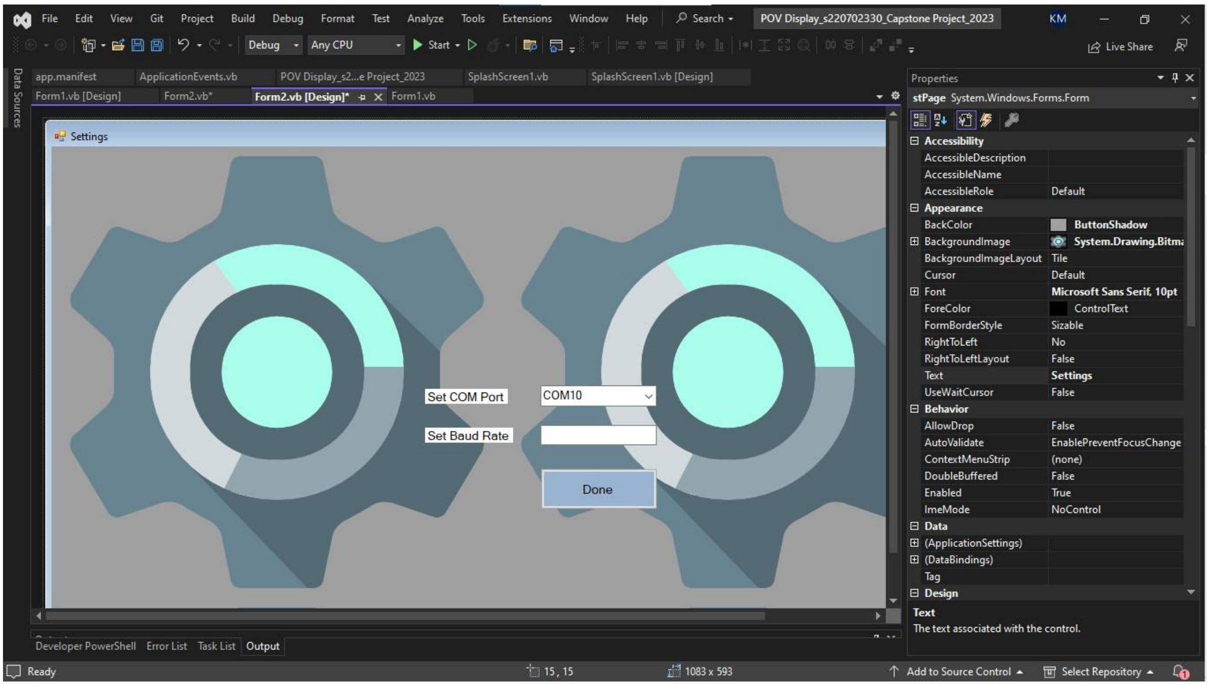


Figure 37: The settings window to adjust COM port number and baud rate. The baud rate can be typed in but default set to 9600 in code.

From the dropdown menu, I had to select the correct port number. The dropdown menu contains all the port numbers available on the computer. The default baud rate is set to 9600. When it was the first time I was trying to connect to the module, I had to do a trial and error until I found the right port number. After changing the port number, I pressed the "Done" button and it takes me back to the original window. If there are other serial ports on the computer which are non-Bluetooth for example, ports created with the STM32 microcontrollers, these ports may actually connect to the GUI (figure 33) but the red LED on the Bluetooth will keep flashing rapidly. In this case, the module is not connected to the GUI, instead it would be another serial port available on the computer.

Figure 38: The GUI is connected to port 10 and the red LED on the module is still flashing rapidly.

Also, the Bluetooth module when it pairs with the computer, sets up a pair of ports but only one of those works for serial connection with the GUI. I did not research the reason for the two different ports or their purpose. On my laptop, the right port number was COM port 7. When I set this as my port number, the LED on the module stopped flashing rapidly and stayed on, only blinking occasionally. This means that the module is successfully connected to the GUI and data can now be transmitted between the two devices.

2. The character display component

The rest of the home window is dedicated to the display component of my project. The two input textboxes cannot take more than six characters each. I did this because the overall project struggles to display a lot of characters since I did not implement synchronisation. It would then be a waste to allow the user to input many characters which they would struggle to see. So, I had to put a limit to the maximum number of input characters. For testing purposes, I started with only one character on one side and no character on the other. The input textbox could not take an empty space as input, therefore had to use a period (.) which is an invalid character in my project, on the other side. If my display receives an invalid character, it will display nothing on that particular column.

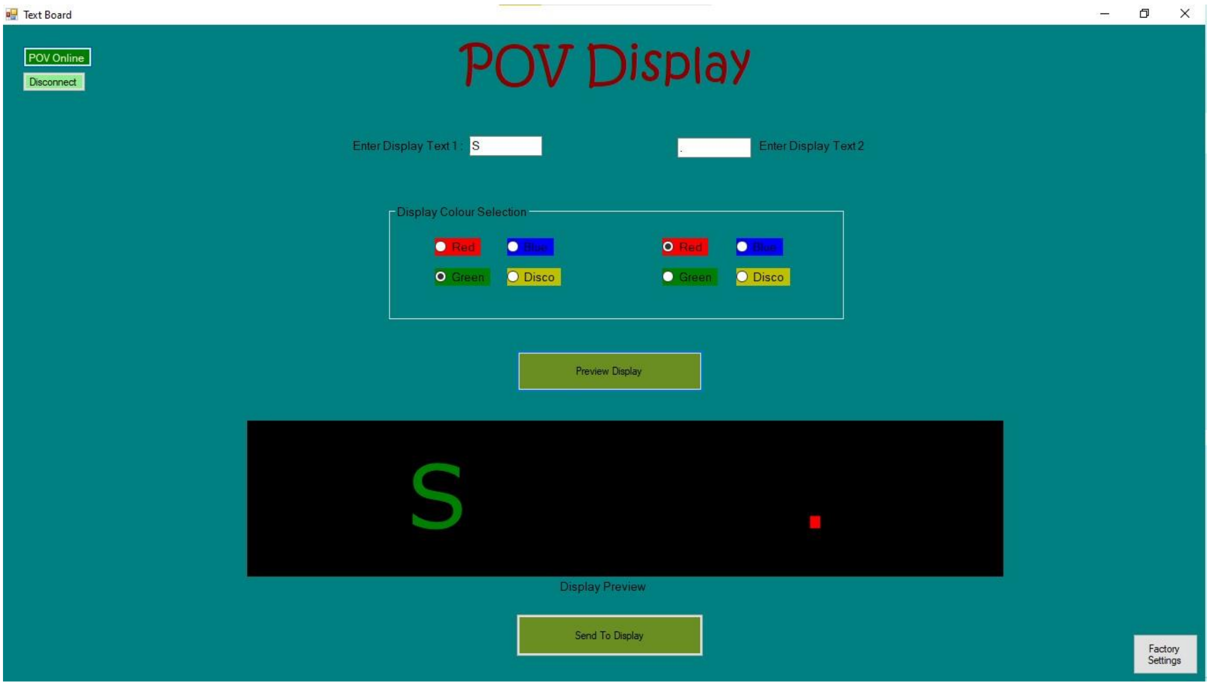


Figure 39: Inputting only one character 's' on one column and an invalid character on the other.

The preview panel (the black rectangle) allows the user to have a preview of their input before sending to the display. This allows for spelling and color confirmation in the GUI. So, after inputting the characters on both columns, I select the colors in this case green for the character 's'. The color for the invalid character must be selected only so that the computer can validate that part of the code but whatever color value is selected will not influence the display. The "Preview Display" button activate the view panel. When satisfied with the preview, clicking the "Send to Display" button will cause the computer to transmit the input data to the display. The POV display results will be shown in the following subsection. Figure 36 – 38 show some of the characters I tried from one character to five characters per column.

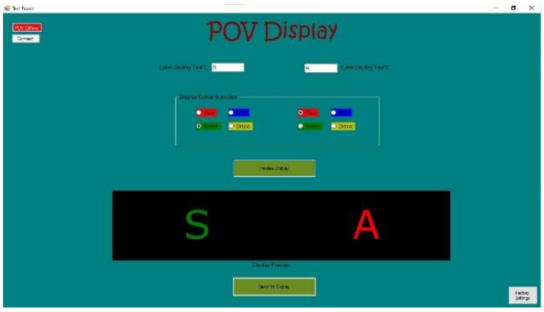


Figure 40: Preview of one character per column Input.

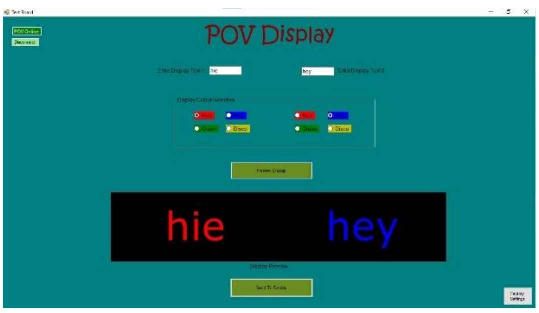


Figure 41: Preview of three characters per column Input.

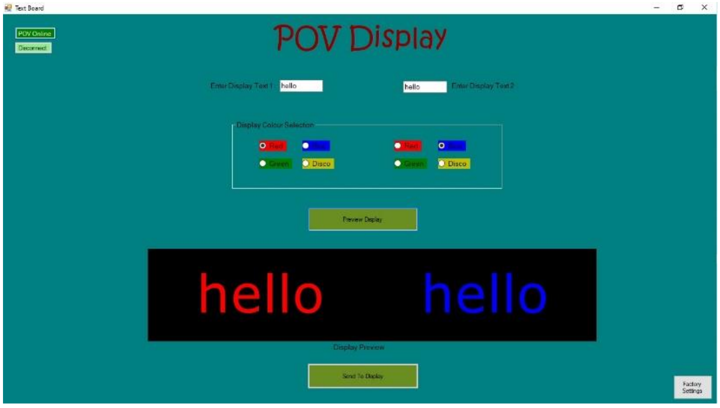


Figure 42: Preview of five characters per column Input.

As for the “Disco” color selection, I could not illustrate the idea in the GUI. This color selection displays characters with all the 8 LEDs flashing in random RGB colors. So, each character will display as patches of red here, and a little green then a blue there and so on.

7.2. Display and overall project testing

I manged to get a decent display that was eligible. However, the more characters I intend to display the less clarity the display shows when displaying the characters. Therefore, for best results, a maximum of 3 characters per column should give a good and legible display. I could not get perfect images using my camera to demonstrate my testing because the camera was not fast enough to capture the POV display. Figures 39 and 40 below were the few of my best shots.

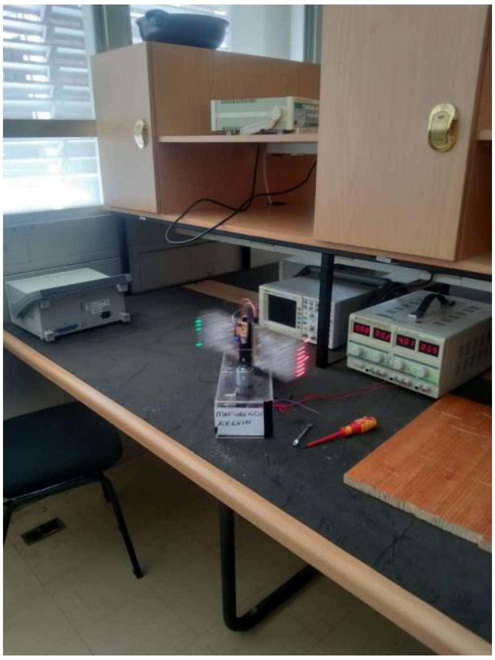
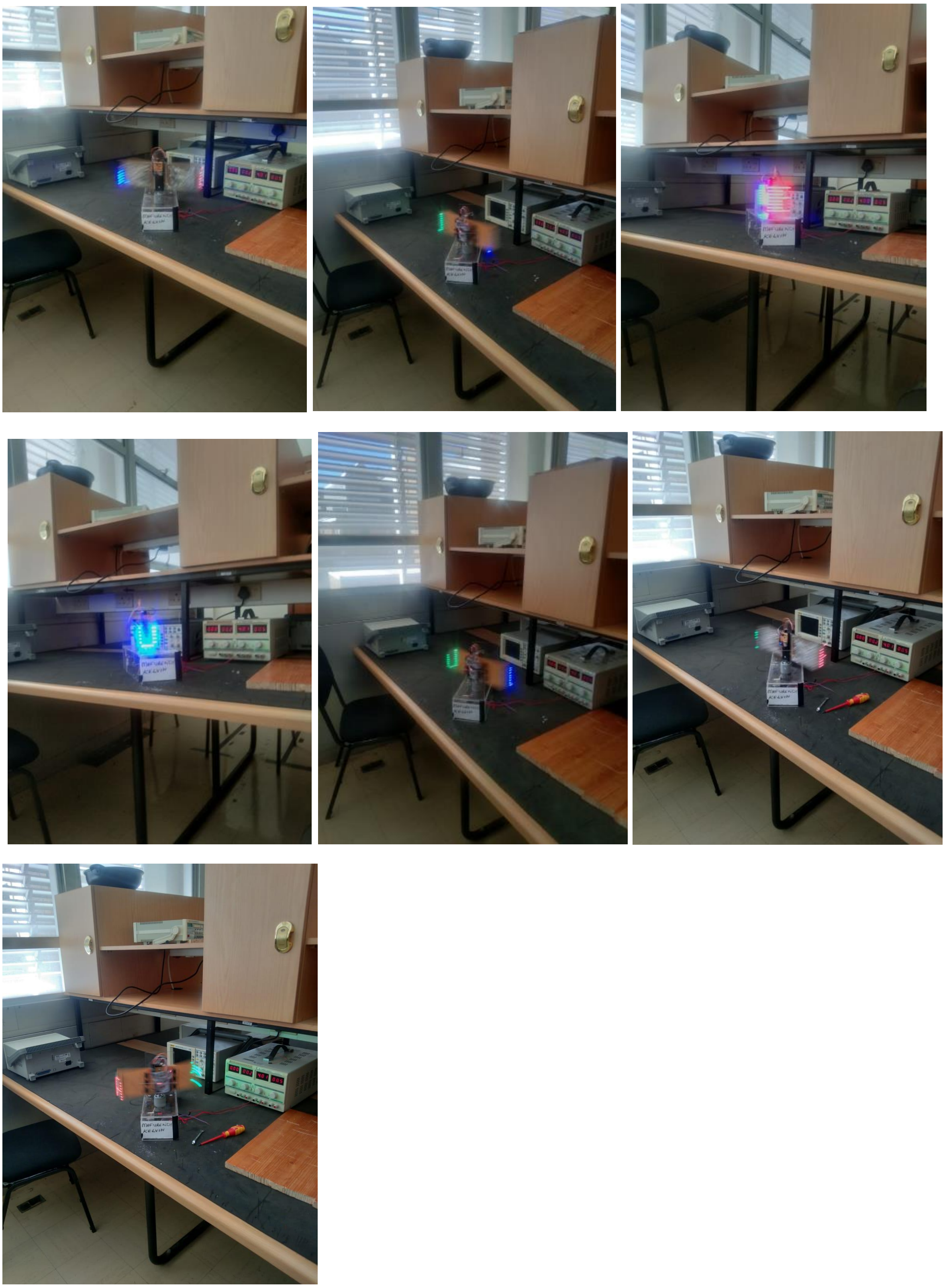


Figure 43: Testing to display characters 'S' and 'A' on either side of the columns.

Figure 44: A clearer image of the message above

When I tested with longer words like ‘Hello’ on either column, the message became more difficult to read. The images below show that with a few color variations. Please note that the images are not showing the words displayed due to the slowness of the camera to capture the LED flashing on time. Synchronization could have helped here also had I implemented it.



Collection 1: Testing the word 'Hello' on both column with color variations.

Overall, the best display results were obtained when the room became darker.

7.3. Motor rotation testing

Since my motor requires a lot of current, I struggled to keep it rotating because I could not secure a sufficient power supply. The motor would rotate at the set speed at first then, after a few minutes, the power supply would just trip. I had to connect the two channels of my power supply in parallel so that I could get a much current as possible. The highest I could get was 8A. Therefore, I had to manage this by running my motor at a lower speed, draw less current and then manipulate my LED flashing frequency to make the speed.

8. Conclusions and Recommendations

This project has been a challenge, with a lot of learning and exposure to different principles of electrical engineering. I would, therefore, make a few recommendations:

- Since the rotor drags a lot of air, that air can be channeled to cool down the heatsink of the H-bridge and any other heating components.
- Using a relatively smaller DC motor which draws relatively little current will make motor control quite easier.
- To cover the rotor entirely so that operators are adequately protected from physical harm.
- Using a position sensor to sync motor rotation and LED flashing would greatly improve efficiency and make the display more legible.

I have also learnt to use and to appreciate the importance of datasheets when working with all project components.

This project does deliver what was proposed, although it has some technical limitations like motor synchronization with LED flashing.

9. List of References

[1] B. K. H. M. A. A. Z. Waleed H. Al-Natsheh, “Design and Implementation of a Cylindrical Persistence of Vision Display,” *2019 6th International Conference on Electrical and Electronics Engineering (ICEEE),* pp. 215-219, 2019.

[2] A. P. &. A. K. K. &. K. K. &. S. Kumar, “Persistence Of Vision Display- A Review,” *IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE),* vol. 10, no. 4, pp. 36-40, 2015.

[3] A. D. S. C. Iustin Constantin, “Automatic light intensity control for a POV,” in *16th International Conference on Engineering of Modern Electric Systems (EMES)*, 2021.

[4] G. B. R. V. R. T. P. V. T. Robinson P. Paul, “Persistence of Vision Control Using Arduino,” *I.J. Intelligent Systems and Applications,,* pp. 102-111, 2014.

[5] D. S. D. S. Akshay Dhruv, “Wireless Remote Controlled POV Display,” *International Journal of Computer Applications,* vol. 115, no. 7, pp. 4-9, 2015.

[6] M. Magazine, “Build a Persistence-of-Vision LED Globe,” 11 April 2023. [Online]. Available: https://makezine.com/projects/persistence-vision-led-globe/. [Accessed 11 April 2023].

[7] R. V. R. N.DHAMODHARAN, “POV PROPELLER DISPLAY BASED ON ARDUINO,” *INTERNATIONAL JOURNAL FOR RESEARCH & DEVELOPMENT IN TECHNOLOGY,* vol. 6, no. 5, pp. 217-224, 2016.

[8] Wordpress, “Interactive Persistence of Vision Globe,” Wordpress, 12 April 2023. [Online]. Available: https://povglobe.wordpress.com/images/. [Accessed 12 April 2023].

[9] S. W. Z. B. LUNIEWSKI Slawomir, “3-D Spherical POV Display with Rotating Semicircular,” *International Journal of Electronics Communication and Computer Engineering,* vol. 8, no. 1, pp. 80-83, 2017.

[10] T. E. W. W. R.M.C.P. Karunarathna, “The persistence of vision (POV) LED globe,” in *Proceedings of the Research Symposium of Uva Wellassa University*, Sri Lanka, 2015.

[11] Hackster, “POV Cylinder with Arduino Due,” Hackster, 12 April 2023. [Online]. Available: https://www.hackster.io/hanoba_DIY/pov-cylinder-with-arduino-due-7016d5. [Accessed 12 April 2023].

[12] A. Instructibles, “Persistence of Vision (PoV) Display Using Arduino,” Autodesk Instructibles, 12 April 2023. [Online]. Available: https://www.instructables.com/Persistence-of-Vision-PoV-Display-Using-Arduino/. [Accessed 12 April 2023].

[13] L. A. Electronics, “LM317 Voltage Regulator,” Learning About Electronics, 30 October 2023. [Online]. Available: https://www.learningaboutelectronics.com/Articles/LM317-voltage-regulator. [Accessed 30 October 2023].

[14] B. R. Møklegård, “ShiftOut Class Reference,” ARM mbed, [Online]. Available: https://os.mbed.com/users/benrammok/code/ShiftOut/docs/tip/classShiftOut.html. [Accessed 25 August 2023].

10. Appendix

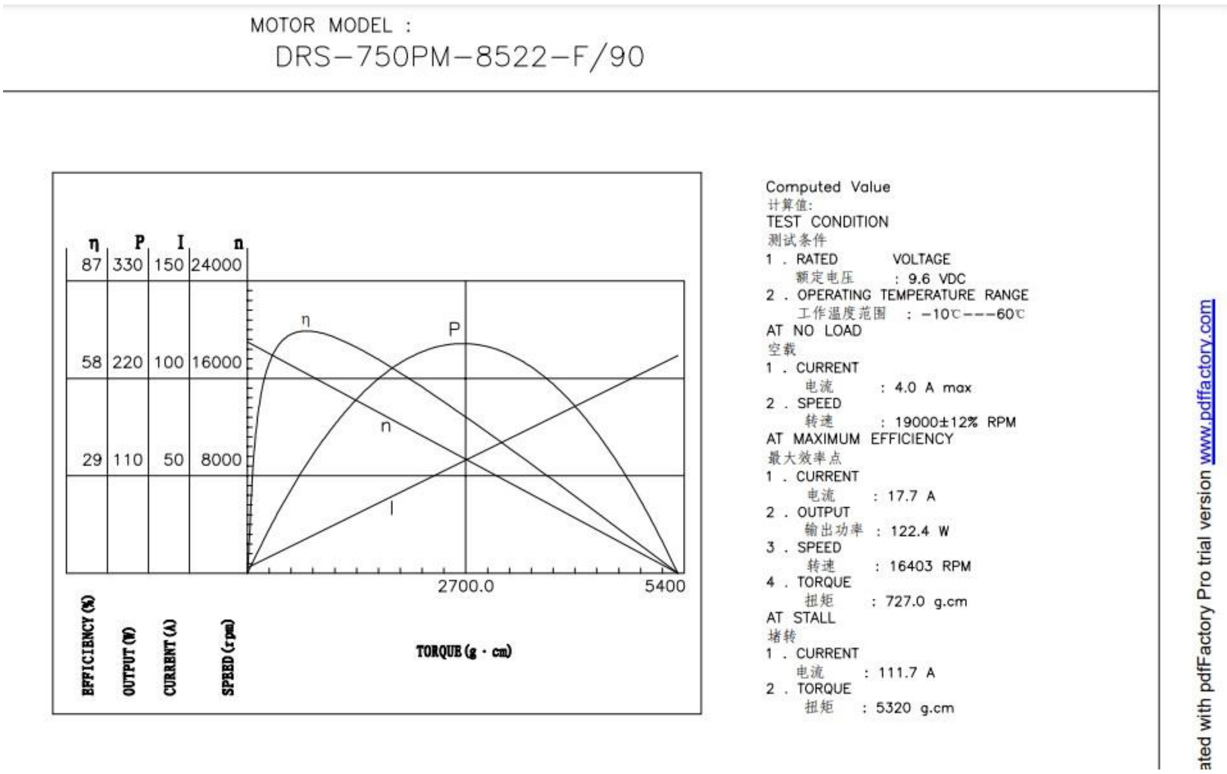


image 1: Electrical datasheet for brushed DC motor

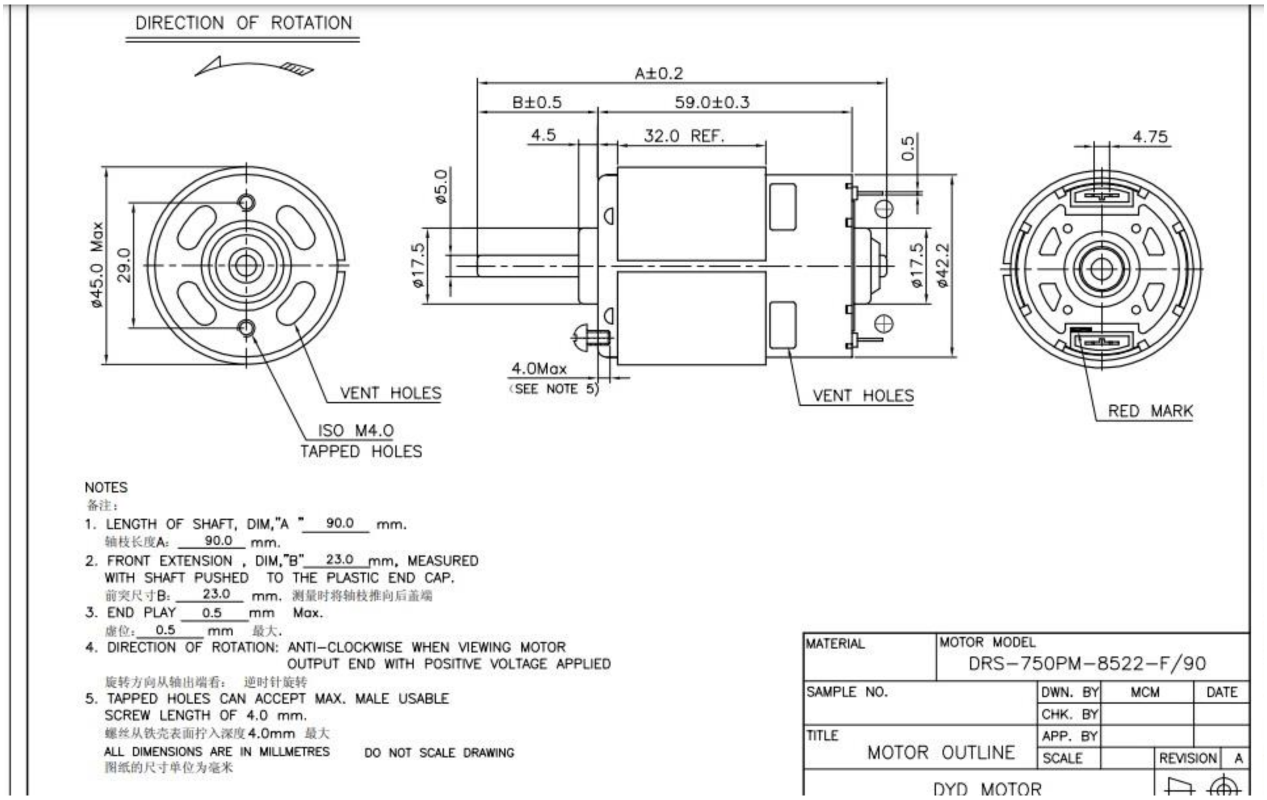
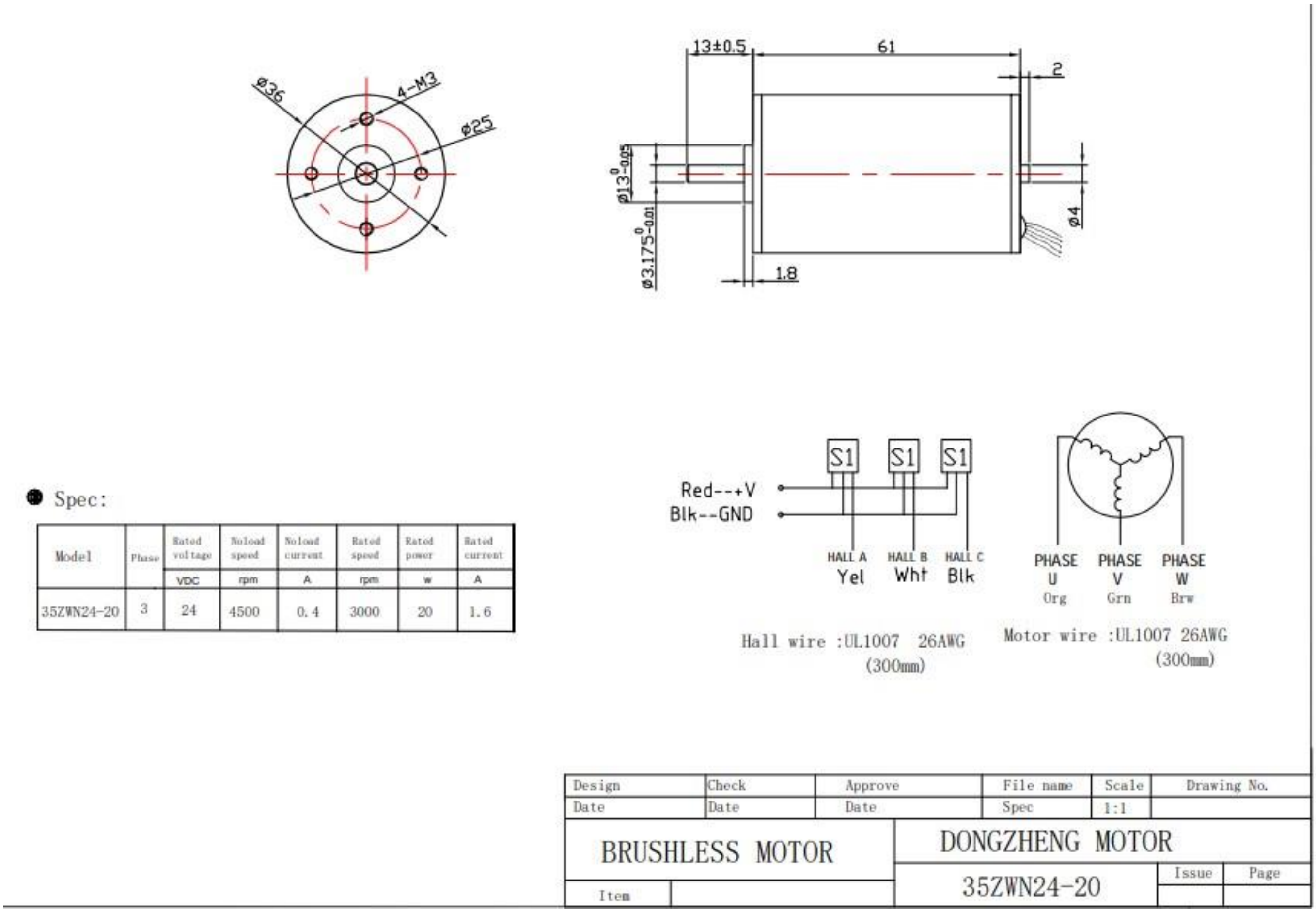


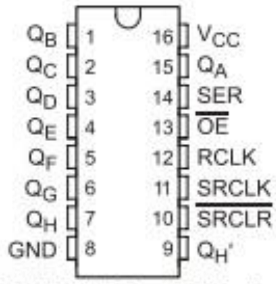
image 2: Mechanical datasheet for DC motor



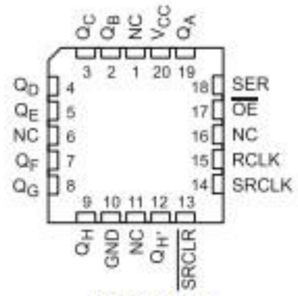
Model	Phase	Rated voltage	Noload speed	Noload current	Rated speed	Rated power	Rated current
		VDC	rpm	A	rpm	w	A
35ZWN24-20	3	24	4500	0.4	3000	20	1.6

image 3: A page from the BLDC motor datasheet

5 Pin Configuration and Functions



D, N, NS, J, DB, or PW Package
16-Pin SOIC, PDIP, SO, CDIP, SSOP, or TSSOP
Top View



FK Package
20-Pin LCCC
Top View

Table 5-1. Pin Functions

PIN			I/O(1)	DESCRIPTION
NAME	SOIC, PDIP, SO, CDIP, SSOP, or TSSOP	LCCC		
GND	8	10	—	Ground Pin
$\overline{OE}$	13	17	I	Output Enable
Q_A	15	19	O	Q_A Output
Q_B	1	2	O	Q_B Output
Q_C	2	3	O	Q_C Output
Q_D	3	4	O	Q_D Output
Q_E	4	5	O	Q_E Output
Q_F	5	7	O	Q_F Output
Q_G	6	8	O	Q_G Output
Q_H	7	9	O	Q_H Output
$Q_{H'}$	9	12	O	$Q_{H'}$ Output
RCLK	12	14	I	RCLK Input
SER	14	18	I	SER Input
SRCLK	11	14	I	SRCLK Input
$\overline{SRCLR}$	10	13	I	$\overline{SRCLR}$ Input
NC	—	1	—	No Connection
		16		
		11		
		16		
V_{CC}	—	20	—	Power Pin

(1) Signal Types: I = Input, O = Output, I/O = Input or Output.

image 4: 74HC595 Shift register datasheet.

■ Typical Optical/Electrical Characteristics 光电特性参数

Item 项目	Symbol 代号	Condition 测试条件		Min 最小值	Typ 典型值	Max 最大值	Unit 单位
Forward Voltage 正向电压	VF	IF=20mA	R	1.9	2.0	2.1	V
			G	3.1	3.2	3.3	
			B	3.0	3.1	3.2	
Light intensity 光强度	IV	IF=20mA	R	6000		8000	mcd
			G	8000		10000	
			B	5000		6000	
Wavelength 波长	WD	IF=20mA	R	620		625	nm
			G	520		525	
			B	465		470	
Reverse current 逆向电流	IR	IF=20mA		0	/	5	uA

image 5: RGB LED datasheet snippet.