UD #: 03SP20

Automatic Print Registration Control System Project

Team Members:
Xulei Cheng
Luke Knapke
Minghao Li
Ruochen Pi
Jessica Weaver

Industry Sponsor:
Taylor Communications

Industrial Client:
David Washburn

Faculty Advisors:
Dr. Philip Appiah-Kubi
Professor Joseph Untener

April 29, 2020

Disclaimer

NOTICE

This report represents the results of a student project at the University of Dayton. Because of the nature of this project, the faculty and student team members do not warrant or guarantee the accuracy of the results nor that they are suitable for any particular purpose. The sponsor agrees that if ideas, concepts or designs from this project are implemented, the sponsor is solely responsible for the reliability, performance and safety of the concepts and designs, and shall indemnify and hold harmless the University of Dayton, employees, students, and any other representatives from and against all claims, losses or damages arising out of scope of this agreement.

Executive Summary

Taylor Communication's Webtron press in the Dayton plant has 18 print/processing units. This Webtron press currently does not consistently print at the desired quality. Sections of the printout can erroneously overlap in an undesired fashion leading to poor print quality. Currently the Webtron press uses manual operators to make print corrections. To have consistent print quality and have corrections made to the printed product as quickly as possible, automatic registration controls are needed. These controls will allow the Webtron press and printing processes to run at optimal conditions without the need for manual print corrections. A "standard" system that makes automatic corrections is available on the market but could cost over $200,000. Therefore, the purpose of this project is to build a proof of concept model that will prove whether a lower cost, modular system based on an Arduino microprocessor could be designed that would work at a lower unit cost yet improve the quality of prints as needed.

A proof of concept was constructed and laid out showing how an automatic print control could be accomplished with an Arduino microprocessor. This proof of concept model simulates one of the fourteen print stations on the Webtron press line. A sensor scans marks on a wheel simulating registration marks that would be on the printed product. The wheel spins at the same speed the paper moves on the Webtron press line. The sensor relays the information picked up from the wheel and allows the software to execute. The software will communicate with the correction motor to adjust the needed amount in order to move the output gear. The output gear is the last point of the proof of concept model, but in reality the output gear attaches to an impression cylinder. The impression cylinder is what is moved for a correction to be made. All of these different components communicate with one another to automatically correct the print unit, thus improving the print quality.

The team has performed the needed research and has built a proof of concept model. This model acts as a starting block for implementing automatic registration to the Webtron press. Additional work will need to be completed before automatic registration will be fully operational on 14 print units and 4 die-cutting/sheeting units. The team recommends to test this registration method on only a couple of units first, then slowly work up until every unit is accounted for.

Table of Contents

1. Introduction
This section contains information about the purpose of the project, the history of Taylor Communications, and the background of the project.

1.1. Purpose
The purpose of this project is to design and fabricate a proof of concept model of a low cost automatic print registration system using microprocessors or controllers such as an Arduino to control motors in order to improve the quality of images. This is expected to improve the print quality of images from the Webtron press at Taylor Communications.

1.2. Background
The Taylor Communications printing plant in Dayton, Ohio has a 14-color printing press that prints window decals for a major credit card company. However, the print quality does not always meet the desired level of quality due to unaligned print registration. As a result, Taylor Communications is seeking a new approach to improve the print quality, which may be done by utilizing a modular system based on Arduino or Raspberry Pi or other microprocessors. Even though there are existing systems in the market that could print to the desired level of quality, they cost over $200,000. Therefore, a design that would work at a much lower unit cost is highly desired. The press needs to be improved with an automatic circumferential registration control to meet customer expectations. This requires the controls to be able to monitor and modulate 13 print units, and 4 die-cut stations. The first print unit (which adds up to make 14 units in total) provides the reference mark.

1.3. Benefits and Feasibility of the Project
This section discusses the benefits and feasibility of the project. While visiting Taylor Communication's facility, it was found that the Webtron press is manually controlled by workers. The press's print towers can currently be adjusted in two different ways. The first option is to manually adjust the tower with a lever that is attached to each unit. The second option is for the operator to use a display screen that has the controls of all the presses connected in one location. The operator can adjust each unit from this one physical location. This movement is what aligns the printing and thus increases the quality of the printing. The plate cylinder is an important component when controlling a print tower's plate position and this project will help to design an automation registration system to make the Webtron press automatically adjust the position/registration to achieve better printing quality. This will greatly save manual labor, simplify the operational steps, and improve the print quality. In addition, better product quality is conducive to improving the reputation of the company and brand promotion. Last but not least, the project will reduce printing costs by minimizing the waste of material. Existing systems on the market would fix the issues the Webtron press is encountering but cost

over $200,000. By using microprocessor technology the cost will be significantly less per unit.

2. Scope

The scope of this project was to study Taylor Communication's Webtron press and develop a method to print more efficiently and accurately. The team scanned and monitored the printed product through a contrast sensor and automatically gained control of the printing location. By using a current technology microprocessor, Arduino, the motor will automatically move forward or back allowing the press to overlap with the previous layer on different unit layers. The registration control system that was designed will work for circumferential registration only. The team will not be adjusting the print registration laterally in this project. In addition, the team completed a proof of concept, setup instructions, and also a set-up of testing protocol explaining the test setup. The prototype included a trial run of 1 unit.

3. Project Specifications

This section will include the design function, design requirements, and design criteria that Taylor Communications have specified for this project.

3.1. Design Function

The intended design of our microprocessor was to be able to hold registration color to color at high production speeds. The goal of the microprocessor was to improve the print quality of pictures by picking up a signal from a sensor and automatically adjusting to the correct location.

3.2. Design Requirements

The production line runs at speeds up to 150-250 feet per minute therefore the design must be able to hold registration color to color within ± 0.005." It must also hold registration die cut/sheeting to print ± 0.030" at the same production speed.

3.3. Design Criteria

This section represents the standards of how the project will be judged. At the end of this project the proof of concept model must be able to pick up registration marks and allow the correction motor to move thus fixing a previous error. The design must be able to model and simulate repeat variable lengths. Initial set up should be easy and preferably be able to be used with a human machine interface display. Operation of the registration system should not add significant time or waste to the system but be beneficial.

3.4. Project Deliverables

This section contains information about the administrative deliverables and the technical deliverables.

3.4.1. Administrative Deliverables

This final report includes the original proposal and will allow the clients at Taylor Communication to see every aspect of the project in one location. Biweekly written reports and presentations were put together and made for stakeholders throughout the semester. In addition, a complete Gantt chart shows the project schedule that was followed can be seen in Figure 7 and 8 of this report.

3.4.2. Technical Deliverables

A fully functional program based on microprocessors or controllers was delivered to the client as well as a tested proof of concept design. Completed schematics, testing protocol explanation, part details, and set up instructions for each component of the proof of concept was also delivered.

4. General Approach

This section contains information about the research, conceptual design, decision analysis, engineering analysis, final design, schedule, costs and fees/budget.

4.1. Research

The starting research for this project was to figure out how a Webtron printing press worked. Outside of the initial client meeting, the group broke the research into three parts: sensors, microprocessors, and motors. The current type of sensor that is used on the print line is a contrast sensor. Contrast sensors are used to detect small contrasts of color while operating at high speeds (reference 2). Since the Webtron printing press operates at speeds of 150-250 feet per minute this type of sensor works best for the application. Research was also done on register sensors, that utilize a register control system to detect marks at high speeds, require complex printouts, and low contrast (reference 4). There was also brief research done on digital cameras that could be used to line up the print cylinder before even printing the color on the paper. The digital camera could have been a viable option, but the cost and the desire to use existing items that Taylor Communications already had were driving factors.

The second part of the research was conducted on microprocessors. To have an automatic registration system, we would need to utilize a type of microprocessor or programming system. The current system that Taylor Communications uses is a PLC and it would be difficult to control the system automatically with this setup because of the number of new inputs required. The group looked into using programming methods such as Arduino and Raspberry Pi. The company already uses Arduino for some of their other product lines so it could be a smart decision to stick with a familiar program. Arduino is an open-source platform that reads inputs and turns it into an output. The programmer can send a set of instructions to the microcontroller on the board that it needs to execute and it will send the output device the instruction to perform. This method was similar to that of Raspberry Pi.

The last part of the research was on the different types of motors. Currently on the print line, a synchronous low speed single shaft motor is being used. Synchronous electric motors are AC motors that run at the same frequency of the supply current and the number of rotations of the motors is equal to the number of AC cycles (reference 5). The current motor operates at 115 Volt AC with a frequency of 60 hertz. One issue was that the Arduino board would have to run with a high voltage AC motor. It was found that with the use of a solid-state relay the two components would be able to work together.

4.2. Conceptual Design

The conceptual design was broken down into a basic flowchart shown in Figure 1.

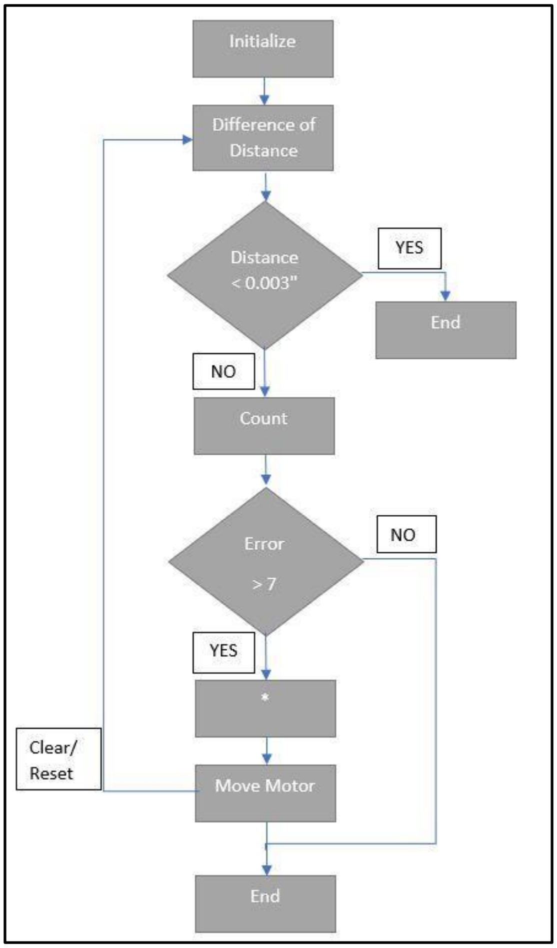


Figure 1: Basic Starting Flowchart

This flowchart was the basis of the code for the system. In this flowchart a starting point was picked on the paper for the sensor to read or initialize. Once the sensor picks up a color, it will start counting pulses until either the color stops or a new color is sensed. The number of pulses is used to determine the distance between each color dot of the print to make a correction if needed. If the dots were within 0.003 inches of each other, no correction would be made. If there are more than 7 errors in a row counted out of 0.003 inches, then the motor would be moved to make a correction. The error of 7 and distance of 0.003 inches were assumptions that were made because they were thought to be reasonable from the discussion that were had between team members. After discussion with the client, the number of errors that occurred before a correction was completed was changed to 3 errors.

Another part of the conceptual design was testing the motor and PLC ladder logic the current system uses to ensure the components that would be used in the new automatic system would work as intended and safely. When testing the PLC ladder logic, the circuit was tested in Multisim to ensure safety when using high voltage. Figures 2 and 3 show the PLC ladder logic and the Multisim simulation of the motor circuit.

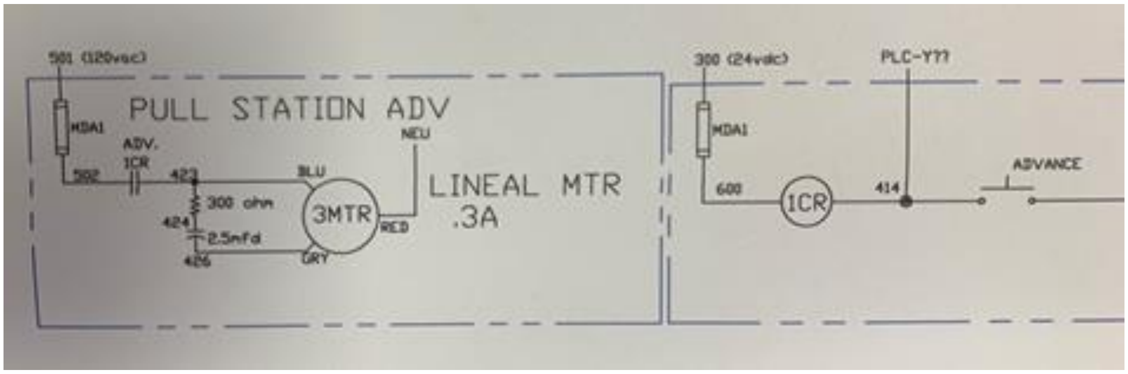


Figure 2: PLC Ladder Logic (provided from client)

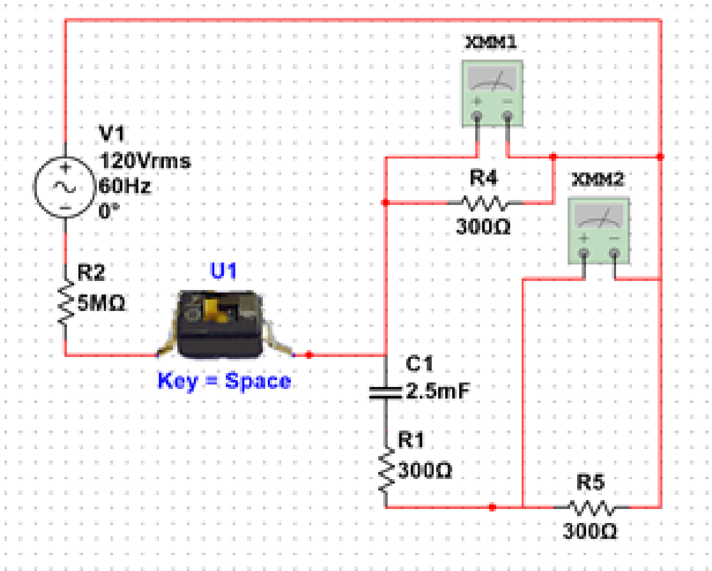


Figure 3: PLC Ladder Logic Multisim Simulation

The motor circuit shown in Figure 4 below, used 300 ohm resistors, a 2.5 microfarad capacitor, and a 5M Ohm resistor that was used as a fuse to allow the motor to run. The circuit was able to run, but since the components were not industrial grade and the lack of an actual fuse, the circuit started to smoke. However, the functionality of this circuit was proved to work. This setup was tested again using 400 ohm 30 watt resistors, a 2.5 microfarad 250 volt capacitor, and a 7 amp 250 volt fuse. This motor circuit would then be connected to the gearbox and Arduino board for a correction to be made.

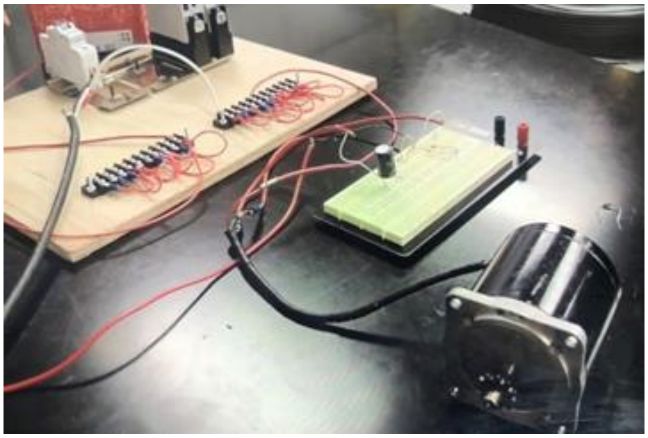


Figure 4: Motor Circuit Testing

Part of the conceptual design also included the use of a silver Buhler motor shown in Figure 5 that would be used to simulate the paper moving through the press. The silver motor acts like the line shaft rotating at 180 rpm and this is equivalent to the paper moving through the press at 150 feet per minute.

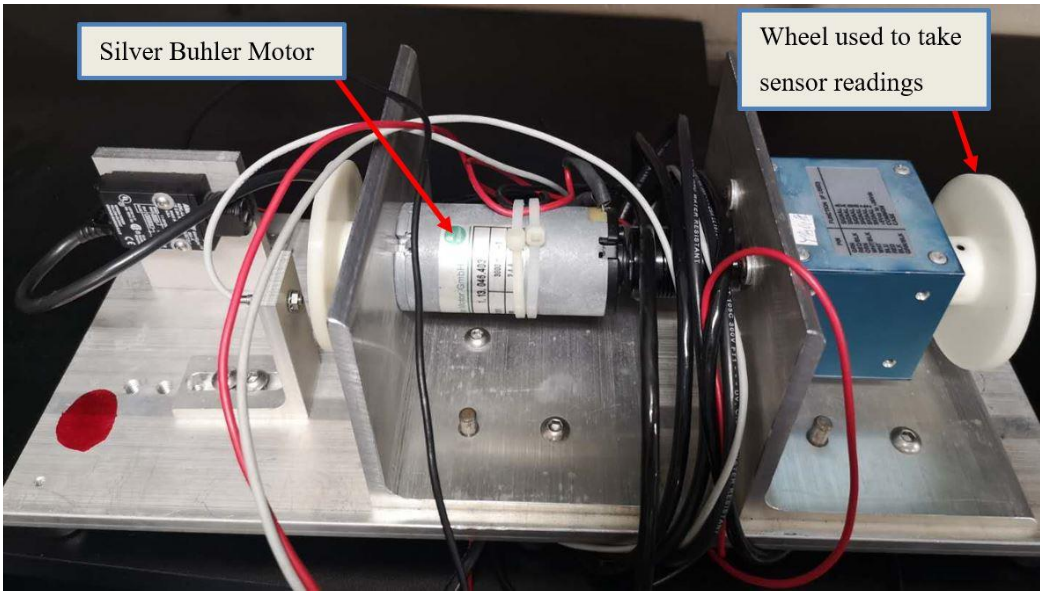


Figure 5: Silver Motor used for proof of concept design

4.3. Decision Analysis

With the research conducted, it was found that the team would not have multiple designs due to the complexity of this project. The decisions that needed to be made were whether Arduino or Raspberry Pi would be used for the programming method. The team decided to use Arduino based on the simplicity of the system, the clients prior knowledge of this program, and prior programming experience from group members using Arduino. The programming language used is C++ which is also the built-in programming language

Arduino uses. The team had prior knowledge using this method and it made it more convenient to model the system in this manner.

4.4. Engineering Analysis

The team went through a series of equations and calculations that would tell the motor whether it needed to advance or retard the impression cylinder to make print corrections on the paper. The communication that told the motor to move the needed amount was the programming that was completed via Arduino. These calculations came from the gear train that runs on the backside of the press. The way the press line operates is there is a line shaft that runs at a constant speed through each gear box at each unit of the press. There is a 11:72 tooth ratio difference between the motor and the correction shaft at each gear box. For every 2 revolutions of the line shaft, there is 1 revolution of the output gear that runs to the impression cylinder. The impression cylinders are 11 inches in circumference and if we want a 0.001 inch correction on the print, there will be 11,000 pulses per revolution. Table 1 shows the gear train calculations.

Table 1: Gear Train Calculations

Gear Train Calculations	Explanation
360 degrees/11,000 = 0.0327 degrees	Angular displacement of output gear to move impression cylinder 0.001 inches
5.333 degrees/0.0327 degrees = 163	Gear ratio divided by angular displacement of output gear
360 degrees/163 = 2.2086 degrees	Adjustment for the correction shaft to get 0.001 inch correction
2.2086 degrees x (72/11) = 14.45 degrees	Adjustment of the motor shaft to get 0.001 inch correction

The reason these numbers are important is they are needed to be incorporated into the code. Although the speed of the line shaft is constant, the motor needs to be able to be moved for a short amount of time to make corrections on the print to ensure better print quality. By using code and the Arduino microprocessor, this will allow the system to make corrections after seeing 3 errors in a row read by the sensor.

4.5. Final Design

The final design is a proof of concept shown in Figure 6 that contains three parts: a sensor, microprocessor, and a motor. The team developed code for the sensor and motor that will function together through an Arduino board. With the use of a “silver” motor to simulate the speed of the paper running through the press, this allowed the sensor to read the registration marks at the anticipated speed. If an error was picked up, it would send a signal to the Arduino board which would then relay a signal to the motor. The shaft of the correction motor would move the correction shaft and the output gear would advance or retard the impression cylinder. This correction would increase the print quality on the paper, therefore making a clearer image. The code will be provided to the client along with a user interface so that there can be a manual correction if needed by the operator. Figure 7 shows the detailed data on the paper with 5 Registration Marks of 4 colors.

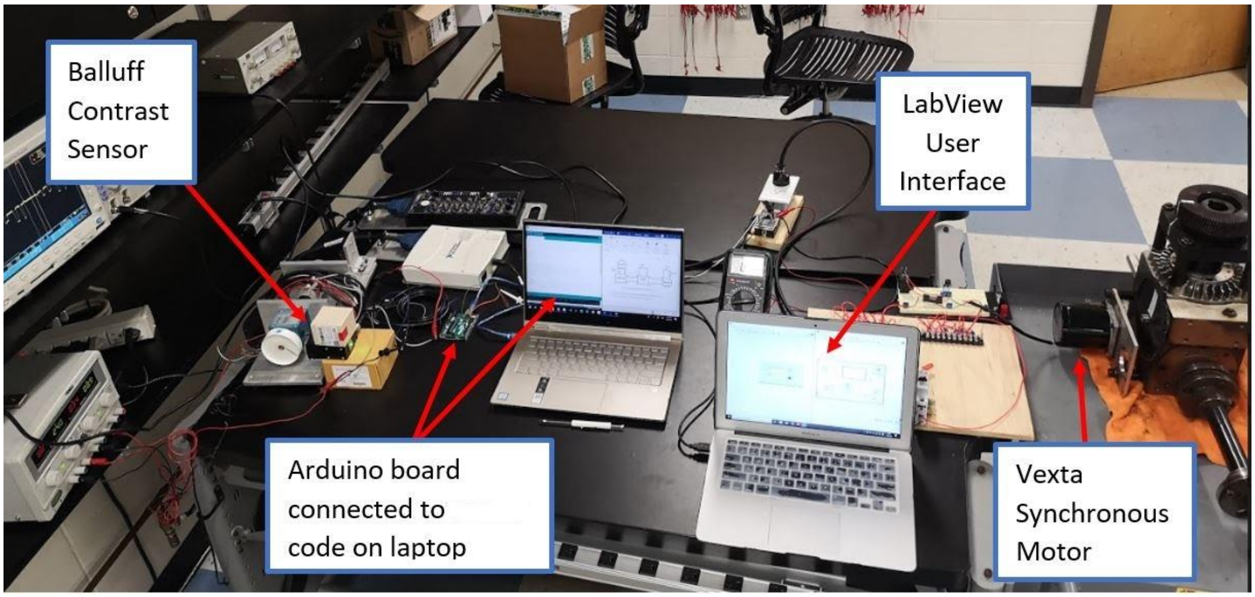


Figure 6: Proof of Concept Design

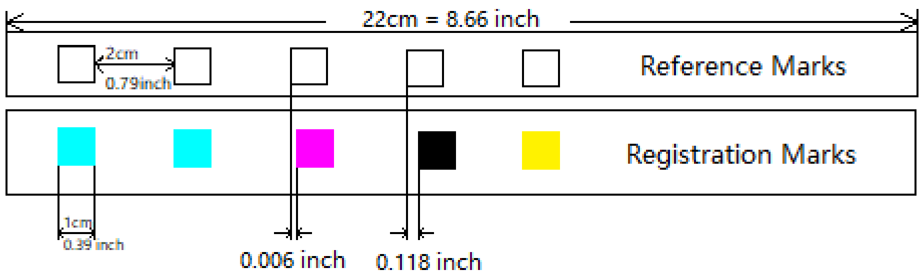


Figure 7: Reference and Registration Marks on the Paper

The design must be an automatic registration system. Therefore, the use of a microprocessor was necessary. Out of the three programming methods found, Arduino, Raspberry Pi, and PLC the group made the decision that Arduino would be best due to personal prior experience and the client having other projects based on Arduino. An advantage to this design is that it was the most practical given the team continued to use the products that are already used on similar press lines within Taylor Communication’s facility allowing the client to have familiarity.

4.6. Schedule

Figure 8 on the following page shows the schedule for the timeline that the project was completed in. Figure 9 shows the milestones of the project and their starting and ending dates.

Taylor Communications Senior Design	**73 days**	**Tue 1/21/20**	**Thu 4/30/20**
Phase I - Research and Brainstorming	11 days	Tue 1/21/20	Tue 2/4/20
First Meeting with Client	1 day	Tue 1/21/20	Tue 1/21/20
Second Meeting with Client	1 day	Wed 1/29/20	Wed 1/29/20
Phase II - Conceptual Design Development	19 days	Thu 1/30/20	Tue 2/25/20
Peer Review 1	1 day	Thu 2/6/20	Thu 2/6/20
Phase III - Design Selection and Analysis	6 days	Thu 2/20/20	Thu 2/27/20
Peer Review 2	1 day	Tue 3/3/20	Tue 3/3/20
Phase IV - Prototyping and Testing	21 days	Tue 2/25/20	Tue 3/24/20
Phase V - Technical Drawing Detailing	16 days	Tue 3/17/20	Tue 4/7/20
Peer Review 3	1 day	Thu 4/2/20	Thu 4/2/20
Phase VI - Evaluation and Reporting	13 days	Tue 3/31/20	Thu 4/16/20
Final Report	1 day	Wed 4/29/20	Wed 4/29/20
Final Peer Review and Log Books	1 day	Thu 4/30/20	Thu 4/30/20
Final Presentation	1 day	Thu 4/30/20	Thu 4/30/20

Figure 8: Schedule with Beginning and End Dates

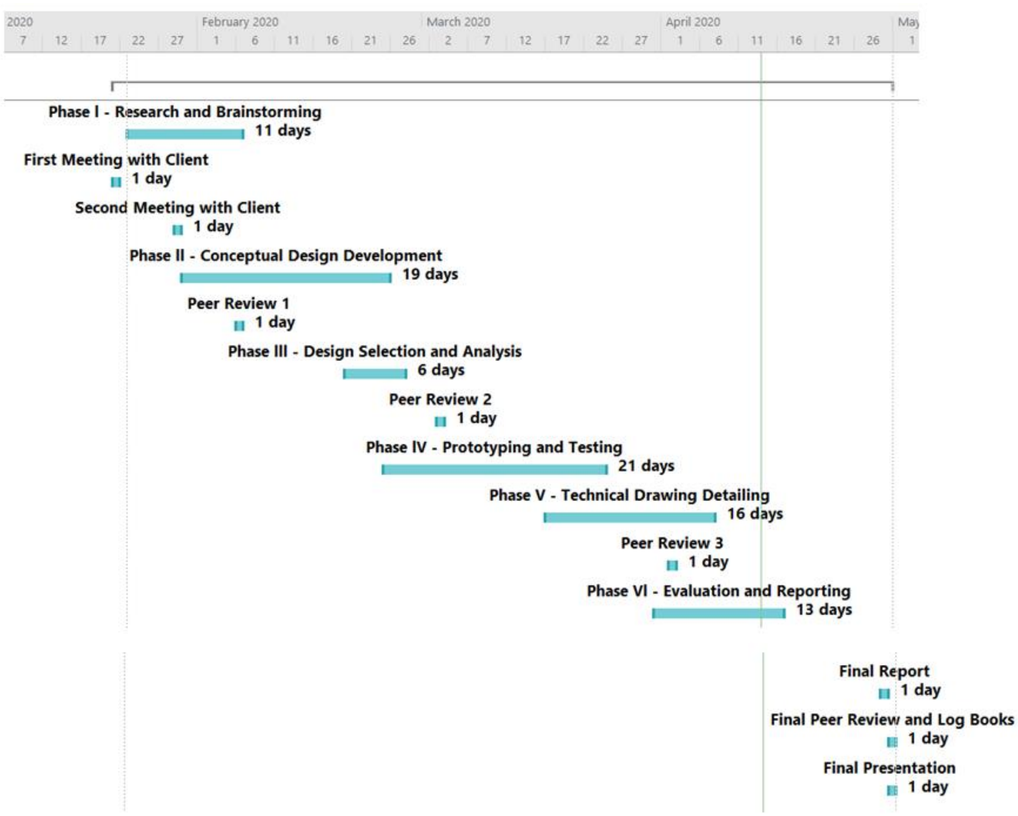


Figure 9: Final Gantt Chart

4.7. Costs and Fees/Budget

The financial budget has two different sections; the budget from the University of Dayton and the budget from the client, Taylor Communications. The University of Dayton provided $1000.00 for the project and could be used in any way that may add value or help the project succeed. This money was spent on items such as software devices, printed control boards, prototyping, and research material.

Taylor Communications has also provided a budget for the team to use for their benefit. Similarly to the University of Dayton, Taylor Communication has provided $1000.00 to be used for prototyping, trying new technologies, research, and general experimentation. In addition to this Taylor Communication is providing $1200.00 - $1500.00 to use per printing tower and die cutting unit. There are a total of 14 printing towers and 4 die cut stations for a total of 18 units. This correlates to $27,000.00 to be used on printing towers and die cutting stations.

Figure 10 shows the financial breakdown of what was spent on materials throughout the semester. The team spent $81.86 of the money that was available from the University of Dayton. $606.00 was spent from the budget that was available from Taylor Communications. Overall, as can be seen in Figure 10, $687.86 was spent between both places.

Materials						
DATE	DESCRIPTION	Purchased by (Client or IC)	Vendor	Quantity (pcs)	Cost/unit ($/pc)	TOTAL
2/5/2020	Balluff Sensor	Client		1	$ 324.08	$ 324.08
2/21/2020	Arduino	IC	Amazon	1	$ 38.99	$ 38.99
2/20/2020	Capacitor Hardware (M&R)	IC	M&R	1	$ 17.20	$ 17.20
2/21/2020	Resistors	IC	Mouser Electronics	1	$ 9.32	$ 9.32
2/28/2020	Fuses	IC	GALCO	5	$ 3.27	$ 16.35
3/6/2020	PLC Aruino Adapter Hardware	Client	Automation Direct	1	$ 281.92	$ 281.92
						$ -
						$ -
						$ -
						$ -
						$ -
						$ -
						$ -
						$ 687.86

Figure 10: Financial Breakdown of Material Costs

Figure 11 shows the transportation cost associated throughout the semester. All the transportation occurred when traveling back and forth between the University of Dayton's campus and Taylor Communication's facility.

Transportation						
DATE	DESCRIPTION	Vehicle (UD or Self)	Total Miles	Total Cost $0.50/mile		TOTAL
1/21/2020	First Client Meeting	Self-Knapke	3	$ 1.25		$ 1.25
1/29/2020	Second Client Meeting	Self-Knapke	3	$ 1.25		$ 1.25
2/5/2020	Third Client Meeting	Self-Knapke	3	$ 1.25		$ 1.25
2/17/2020	Fourth Client Meeting	Self-Knapke	3	$ 1.25		$ 1.25
2/25/2020	Fifth Client Meeting	Self-Knapke	3	$ 1.25		$ 1.25
2/25/2020	Fifth Client Meeting	Self-Weaver	3	$ 1.25		$ 1.25
3/5/2020	Sixth Client Meeting	Self-Knapke	3	$ 1.25		$ 1.25
3/5/2020	Sixth Client Meeting	Self-Weaver	3	$ 1.25		$ 1.25
				$ -		$ -
				$ -		$ -
				$ -		$ -
				$ -		$ -
						$ 10.00

Figure 11: Financial Breakdown of Transportation Costs

5. Final Design

This section will include the design features, detailed description and analysis of components, engineering analysis and calculations, prototype fabrication, prototype testing, data analysis, refinement and results.

5.1. Design Features

The group's design contains the use of a contrast sensor and a motor that is connected with software programming. A bench top design was built to simulate the three components working together. This includes using the sensor to record color marks and record any errors to send a signal to the code on the Arduino board. The Arduino will communicate with the correction motor to make a correction on the output gear. A LabVIEW user interface can be used by the operator to make any manual corrections. Below is the detailed design feature for the motor and the sensor.

5.1.1. Motor design features

The team used a VEXTA Synchronous 4CMS-111 motor, as seen in Figure 12. This AC motor is more durable than a DC motor. This motor was used because it is the same motor that is used on the Webtron Press today.

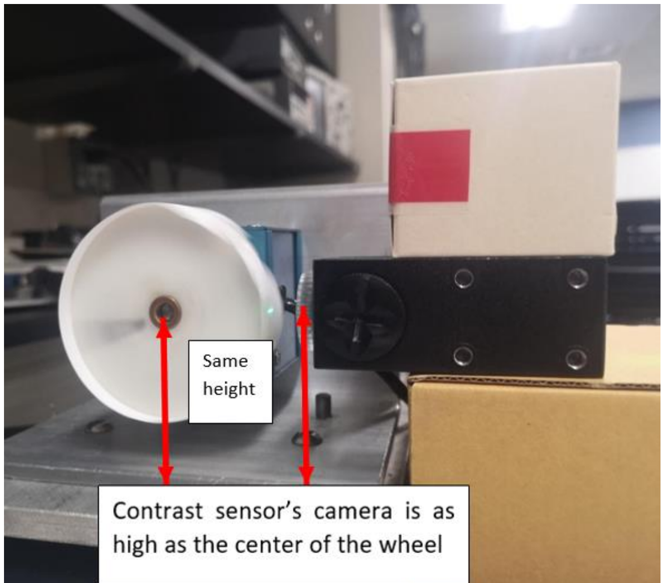
By using the same motor the team knows it is powerful enough to move the components that make up the gear train. With this verification there was no need to test a new motor to ensure it will work for the application intended. By using the same motor, money is saved on potential purchase and installation costs.

Figure 12: VEXTA Synchronous 4CMS-111 Motor (from Reference 1)

The team used an Arduino to connect the SSR (solid state relay), upload the code (shown in Appendix F) to the Arduino, and then used the SSR to control the motor. The team used the timing of the correction motor's rotation to control the angle of the correction motor shaft and the resulting linear distance the motor's rotation causes on the output gear/impression cylinder.

5.1.2. Sensor design features

The team used a Balluff contrast sensor to detect and record specific marks tied on a SSR (Solid State Relay). The group used the contrast sensor due to its high accuracy, allowing the sensor to detect and record color registration marks at a high rate. The sensor can be seen Figures 13 and 14.



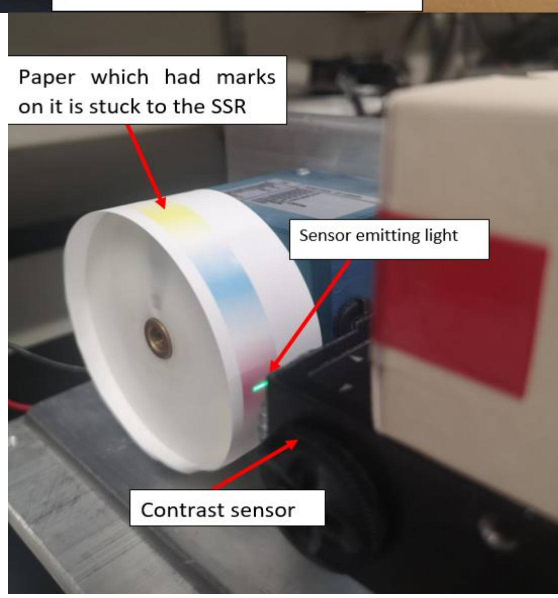


Figure 13 & 14: Contrast Sensor Readings

5.2. Detailed Description and Analysis of Components

This section mainly talked about the detailed description of motor, sensor, and complete system of the proof of concept.

5.2.1. Detailed Description and Analysis of Motor

The team used the rotation of a VEXTA Synchronous 4CSM-111 motor, also known as the correction motor, to eliminate errors as part of the final design. The following is the basic parameters and description of the sensor. In the circuit connected with the motor, as seen in Figure 15, the control switch of CCW and CW is shown. When connecting CW switches, the positive error can be eliminated. Negative errors can then be eliminated when connected to the CW switch.

Reference 1 supplied the following information: Motor Supply voltage is AC 60 Hz, 115V. The motor rate of revolution is proportional to the frequency of the power source, 72 rpm at 60 Hz. Extremely low speeds and high torque is possible without reduction gears. The motor needs a voltage of 115 volts for normal operation and the rated load of the correction motor can be no more than 0.3Amps and 13 watts. The rated speed is 72RPM. Therefore, the capacitor that is used in the motor circuit is 2.5Mfd, 250Vac, and the resistance that is used is about 400Ohm and 30Watts.

The output power of the motor is 1/113 Hp and 6.6 Watts. Insulation resistance of the motor is 100M ohm or more. The working temperature range is -10 degrees Celsius -40 degrees Celsius (14 degrees Fahrenheit -104 degrees Fahrenheit). Also, there are other requirements (Figure 16) motor general specifications. Figure 17 shows the PLC connected motor circuit.

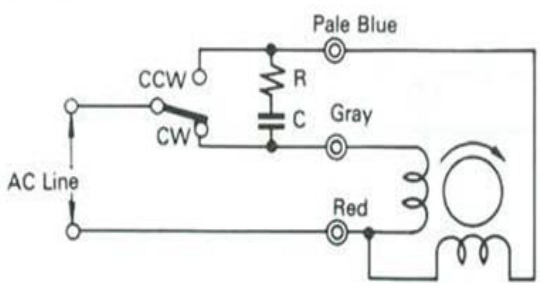


Figure 15: 4CSM-111 Motor Circuit (from reference 1)

Motor type	Output power		Hz	Volts	Rated load				Residual Torque oz.-n(N-cm)	Rotor Inertia oz.-in^2(g-cm^2)	Capacitor		Resistor	
Single Shaft	Hp	Watts			Amps	Watts	Rpm	Torque oz.-in(N-cm)			Mfd.	Vac.	Ohm	Watts
4CSM-111	1/113	6.6	60	115	0.3	25	72	125(88.3)	7.64(5.39)	6.01(1100)	2.5	250	400	30

Figure 16: Manual of 4CSM-111 Motor (from reference 1)

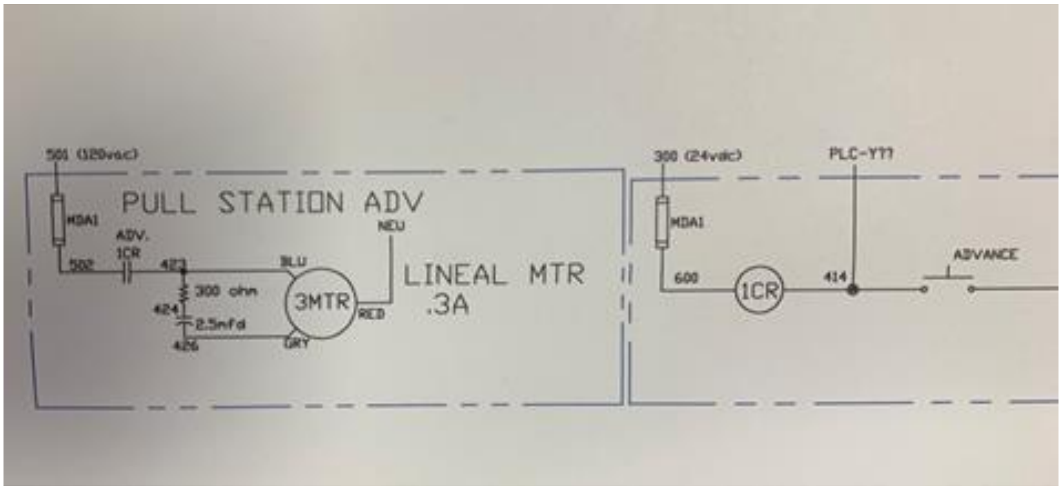


Figure 17: Motor Circuit from diagram (provided from client)

5.2.2. Detailed Description and Analysis of Sensor

The team used a Balluff contrast sensor (BKT 67M-003/004-U-S92) to achieve part of the final design. Below are basic parameters and button an introduction of this sensor, information was taken from reference 2.

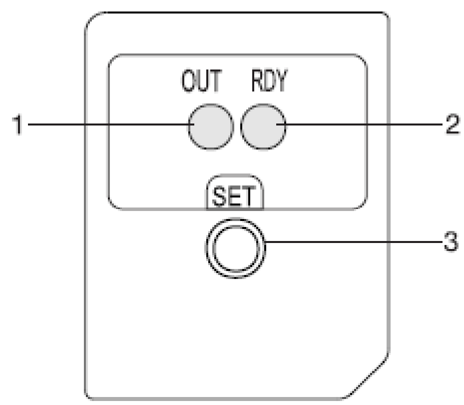


Figure 18: Display and Operating Elements (from reference 2)

The sensor has two indicator LED lights, as shown in Figure 18. The first is the yellow LED OUT which is the output function indicator and when this LED is on, it means that the output has been activated. The second is the Green LED RDY which is the ready status indicator. When it is on the sensor is ready and if the output LED light is blinking, it means the output is overloaded. There is also a key set button and this button is used for teaching the sensor to record a specific color.

This type of sensor possesses the ability to record specific color marks and comparing this recorded mark with other colors it detects to make a reaction. The sensor will generate a pulse once it detects a recorded color. The team wrote code to calculate the time of pulses by using time multiplied by the speed of rolling paper to give the distance of the measured mark.

5.2.3. Detailed Description and analysis of Complete System

The hardware of the whole system includes Balluff Contrast Sensor, VEXTA Synchronous Motor, Arduino Uno board, Solid State Relay, gear train, digital power supply, 120 VAC power socket, NI LabVIEW board and ports board, PLC trainer, two laptops, “silver motor” (simulating the speed of the paper moving on the press), resistors, capacitors, and cables. Hardware and lab setup is shown in Figure 19. The Arduino Uno board is the processor of the whole system and controls the devices with the uploaded code from the left laptop. The User Interface is shown on the right laptop programmed with a LabVIEW platform with the NI LabVIEW board and ports board, also shown in Figure 19. The operator can enter the error distance that is wanted for the motor to correct with units of thousands of inches. This user interface is shown in Figure 20. FWD (forward) or REV (reverse or retard) can be chosen to make the motor rotate clockwise or counterclockwise. “Data” shows the output value to the SSR controlled by LabVIEW (maximum is 5; minimum is 0). The Time Counter light is used to show the maximum (on) or minimum (off) of the output value. The SSR is connected with a protection circuit of the motor using a PLC trainer to connect with the socket on the SSR. The output gear on the gear box will show the final correction error distance controlled by the correction motor, which is connected to a chain drive and gear box assembly. The silver motor simulates the paper moving on the Webtron press with the same speed of 180 RPM (130feet/min).

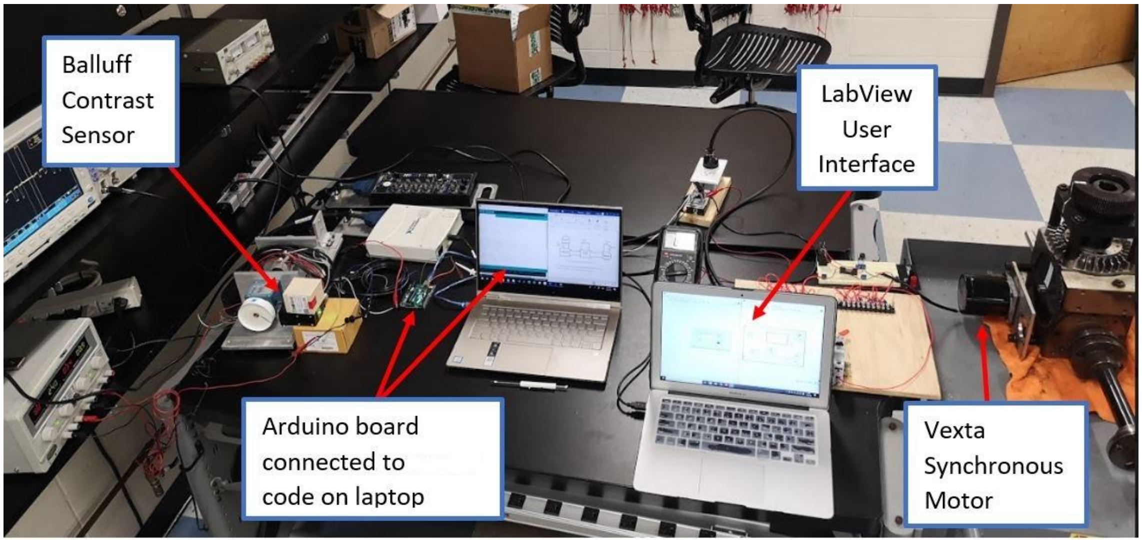


Figure 19: Hardware of the Proof of Concept

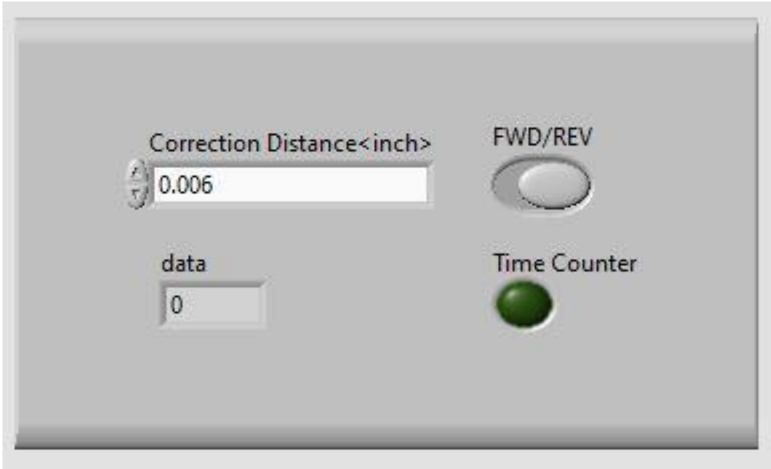


Figure 20: Front Panel of the User Interface on LabVIEW

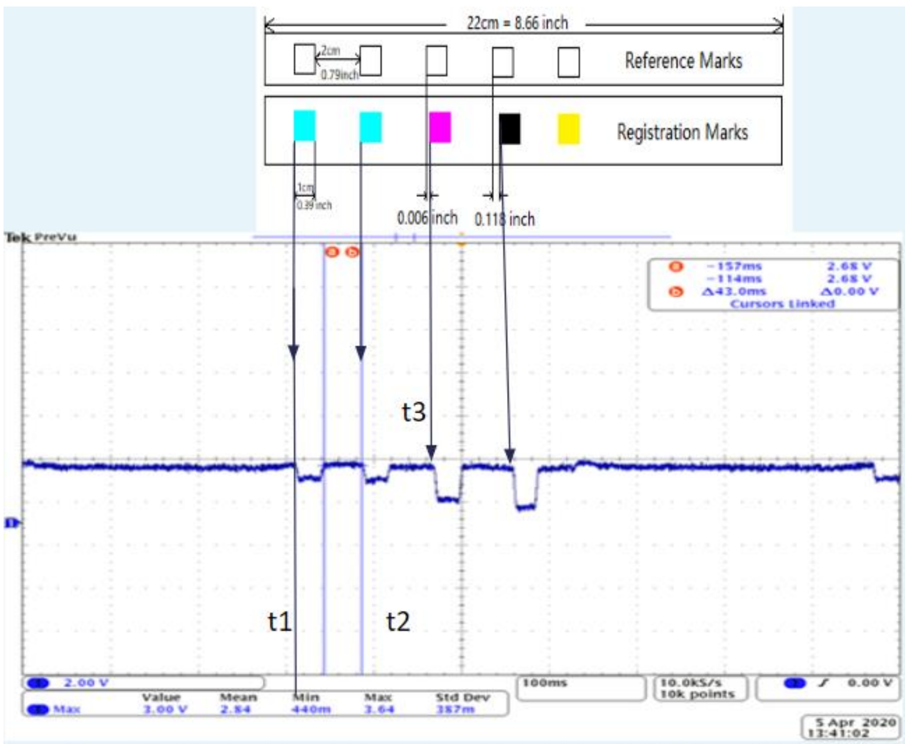


Figure 21: Relationship between Registration Marks on the Paper and the screenshot of the Oscilloscope

As for software, Figure 22 shows the detailed flow chart for the code. The complete code for the proof of concept is in Appendix F. There are four parts in the code including initialization, sensor data input of distance measurement, and motor process of error distance correction. In the Initialization part, the analogPin is the pin of the Balluff sensor output. The digitalPin is the pin connected to the digital port of SSR (Solid State Relay) with a limitation of 3 VDC to 32 VDC, which energizes/cuts off the circuit of the

VEXTA motor with 120 VAC power supply. The global variable named “count” is used to record the number of complete rotation of the silver motor. The constant named Reference_Distance with the unit of 10^-3 * inch is equal to 3cm in the proof of concept, which is shown in Figure 21. The constant named “StandardTime” is tested by the LabVIEW, which means if the error distance equals 0.001 inch, the motor will move for 0.38 second. Data[5001] and Distance_Difference[3] are two arrays with 5001 and 3 array size, which can be in decimal form with the type of float. The array named time_storage[4] has 4 variables in this integer array to store the testing time between marks.

The sensor first reads the analog output of the Balluff sensor. Due to there being an analog input with the Arduino board, the code uses the average value of every three datum as the input value to reduce testing error from the sensor. Then if the former input data is smaller than 80% of the previous data, it means the sensor tests the mark at that time. The time comes from the function called “millis()” to store in time_storage[t] array with the increasing index variable “t”. In calculations this is also called control part, time1 and time2 are the time differences between the first mark (cyan mark1) and mark2 (cyan) and mark3 (magenta). The speed of the silver motor can be calculated by dividing the reference distance (3cm=1.18inch) over time1 (from the left edge of mark1 to the left edge of mark2). Tested_distance from mark 1 to mark 3 is calculated by multiplying the speed of the paper and time2 (from the left edge of mark1 to the left edge of mark3). Finally, the Distance_Difference is the difference value from two distances between 3 marks. After the previous steps, the “count” variable will add 1 to record the first value of Distance_Difference. Finishing 3 times of the recording, the Average_Distance can be calculated by dividing the sum value of 3 Distance_Difference.

In the motor section, if the average_distance is in the range of [-0.003 inch, 0.003inch], this means the error distance can be ignored without any correction. If the average distance is a negative value, it means the motor should run REV (CW) circuit with time of multiplying Average_diatance and StandardTime (0.38s) and the gear will run counterclockwise. Otherwise, if the average distance is a positive value, it means the motor will run FWD (CCW) circuit with time of multiplying Average_diatance and StandardTime (0.38s) when the gear will run clockwise.

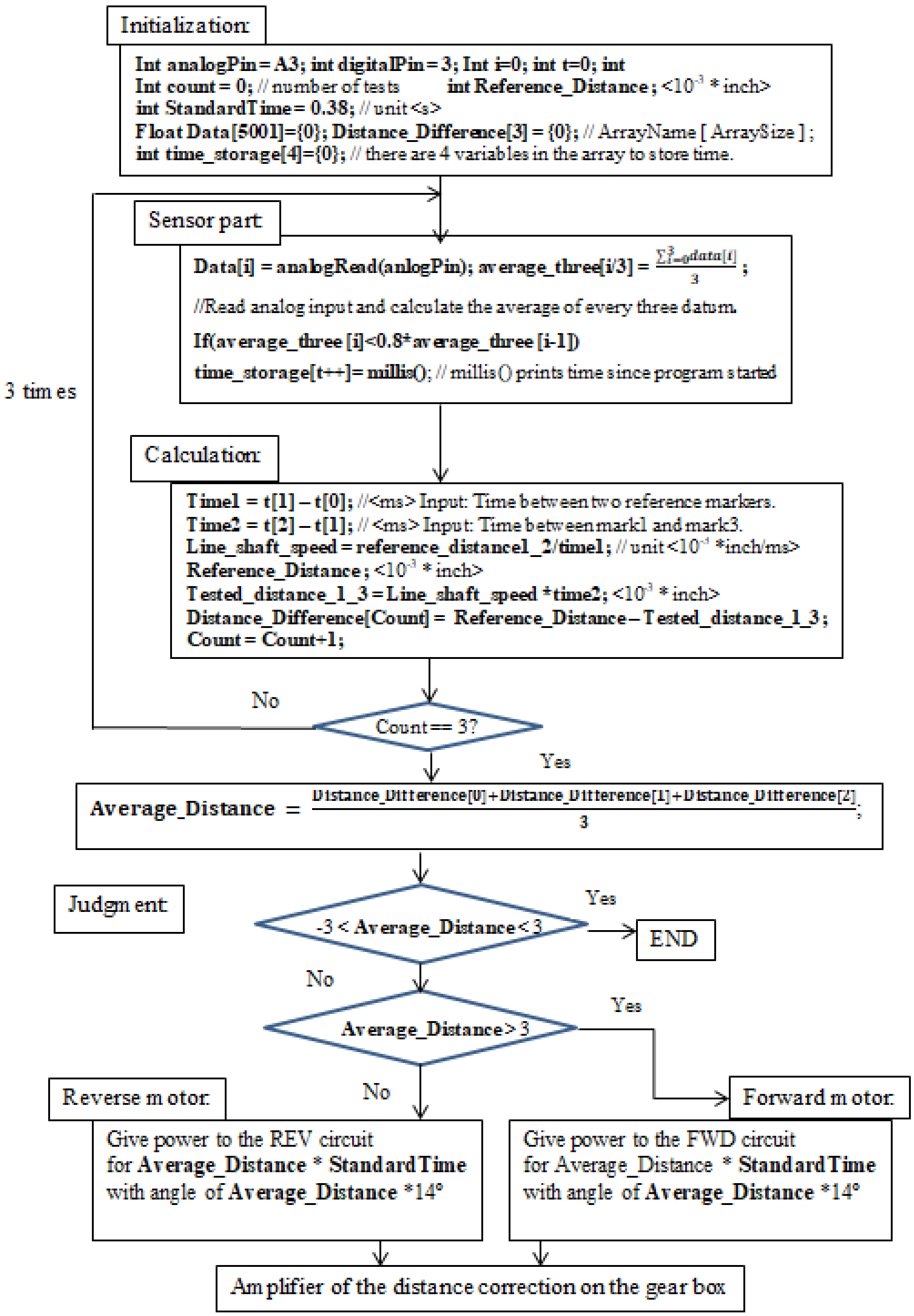


Figure 22: Detailed Logic Flow Chart

5.3. Engineering Analysis and Calculations

This section talked about the detailed engineering analysis and calculations for both motor part and sensor part.

5.3.1. Engineering Analysis and Calculations for motor

Before the team ran the correction motor to correct the error, the team needed to know how much time it takes to correct 0.001 inches by rotating the motor, and the team named this constant as “T.” Below is how the team analyzed and calculated the “T”. The team used two methods to get the “T” value.

The first method is to use the cell phone timing (Figure 23), directly control the motor with SSR, and calculate the time required for the motor to rotate 14.5 degrees. For every 14.5 degrees the motor shaft moves there will be a 0.001 inch for every error.

The second method is to directly measure the time it takes the gear train to rotate 0.001 inch. The PC is connected to the Arduino, Arduino connected to SSR, SSR controls the correction motor, and correction motor controls the gearbox. This method is more accurate because the actual measurements will include friction and other errors not taken into account in the theoretical calculations of the motor gear conversion to the gear train. Due to 0.001 inch not being easy to measure, the team calculated the 0.001 inch time of the output gear by timing the gear train to move 2 millimeters to get the “T” value. The result is achieved when the gear train rotates 180 degrees. It took 30 seconds to rotate 180 degrees and it moved 2 millimeter, which is 0.0787 inches. After calculation, the correction shaft rotated 2.28 degrees to get a 0.001 inch correction on the impression cylinder. This took .38 seconds.

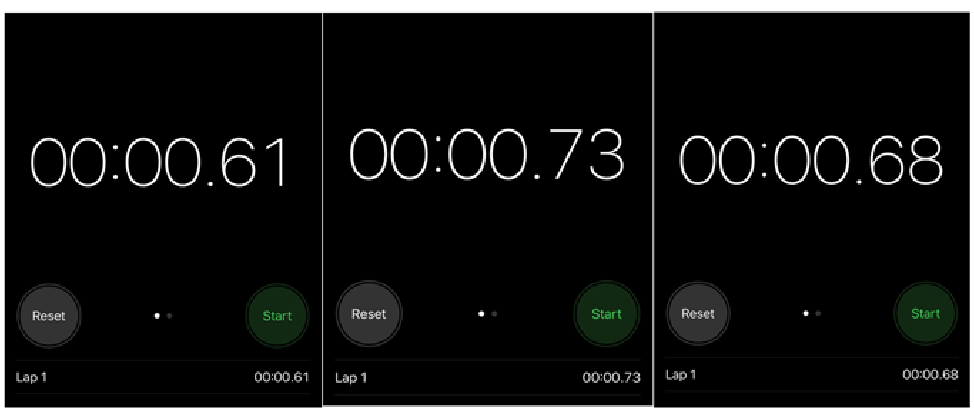


Figure 23: Counting Time

It is known that it takes little time for the gear train to move 0.001 inches and that the time the gear train is energized can be controlled.

The team used this approach to write the motor programming code. Figure 24 is the logic flowchart for motor programming. First, initialize the input port and output port and set the input port as "A3" and the output port as "~3". Then, initialize all variables and quantitative values. The data input into the Arduino is the error value transmitted by the sensor, named "e", with units of inches. From the experiment of measuring "T", the value of "T" is 0.38, a quantitative value. Initialize the value of "M" allowing "M=e/0.001" which defines how many 0.001 inches are in the data error found from the sensor. The result of the last output is the turn on time of Arduino control SSR, named "t", with units of seconds, and "t=M* t." This allows the SSR control motor rotation to eliminate M 0.001 error time of gear train.

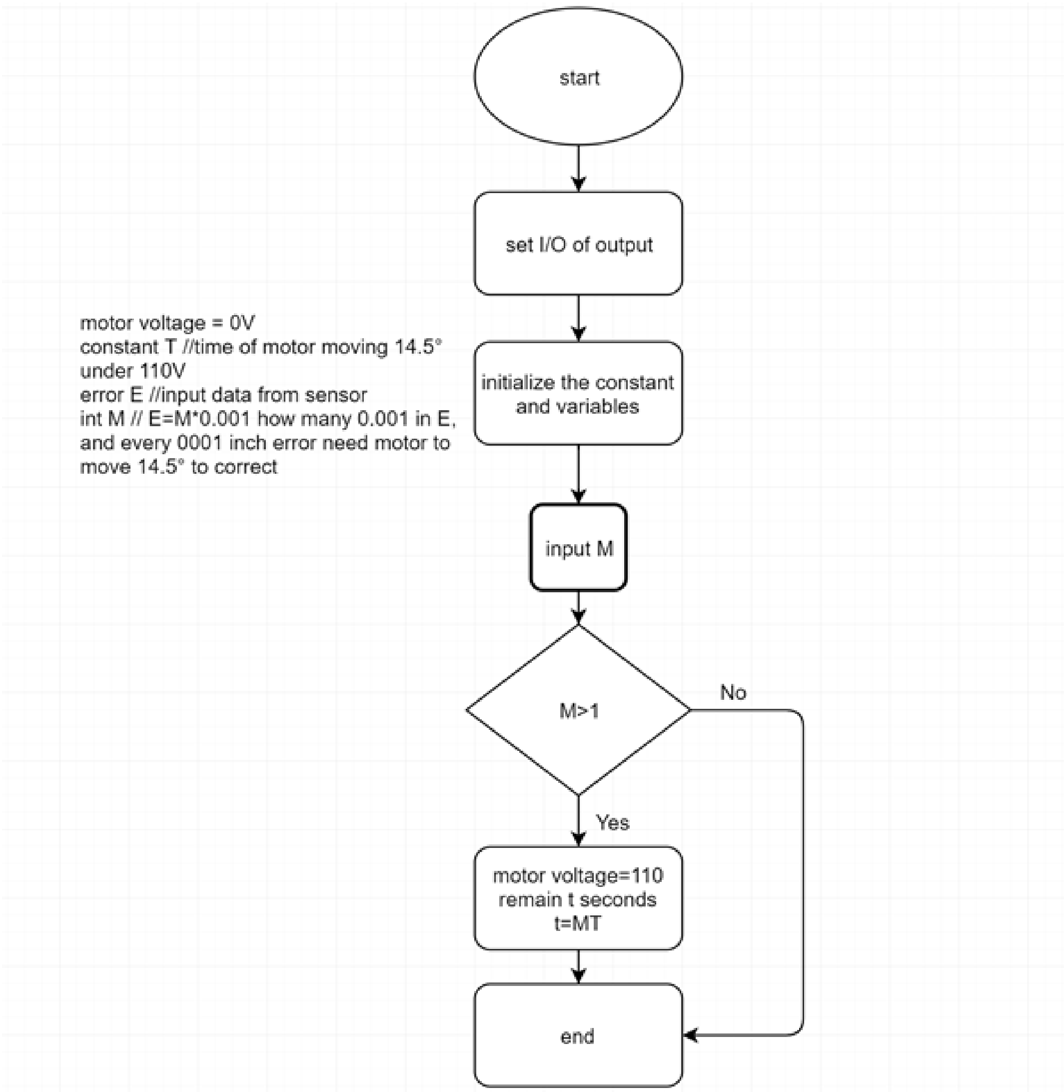


Figure 24: Motor Logic Flowchart

5.3.2. Engineering Analysis and Calculations for sensor

Figure 25 and Figure 26 show the sensor testing that was completed.

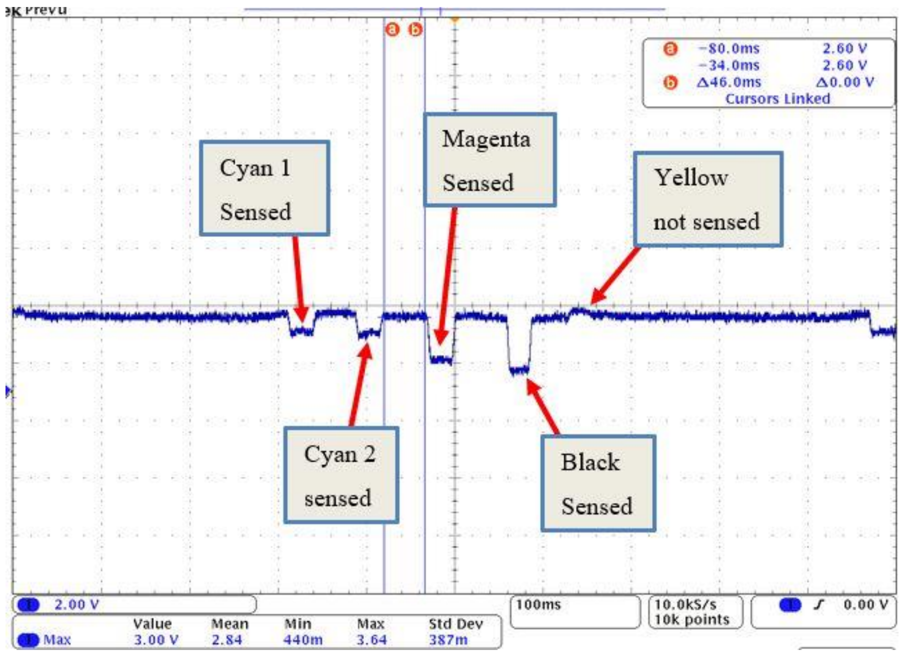


Figure 25: Sensor Testing 5 Color Strip

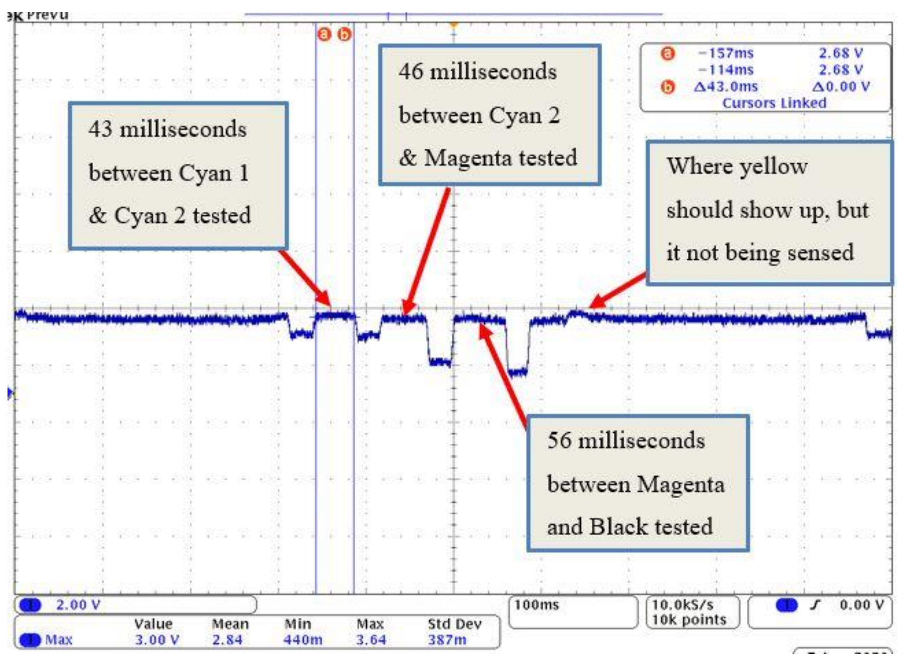


Figure 26: Time in between Sensor Color Readings

When testing the sensor 5 colors were used: blue 1, blue 2, magenta, black, and yellow. It was found that yellow did not test well with the green light given off from the sensor. Therefore yellow did not show a square wave on the oscilloscope as shown in Figure 25&26. For the color strip placed on the wheel of the silver motor, all colors were evenly distanced, except for the color black. The black square was placed 0.118 inches further from the magenta square than blue 2 was. This was used to demonstrate an error in the registration print and would allow a correction to be made with the motor. Shown in Figure 26 above you can see the time between blue 1 and blue 2 was 43 microseconds and the time between blue 2 and magenta was 46 microseconds. This shows that the sensor reading is accurate when operating at speeds of 130 feet per minute. With the offset distance between magenta and black, the time was found to be 56 microseconds. This shows that an error had occurred because the time between these two colors was larger than the other colors.

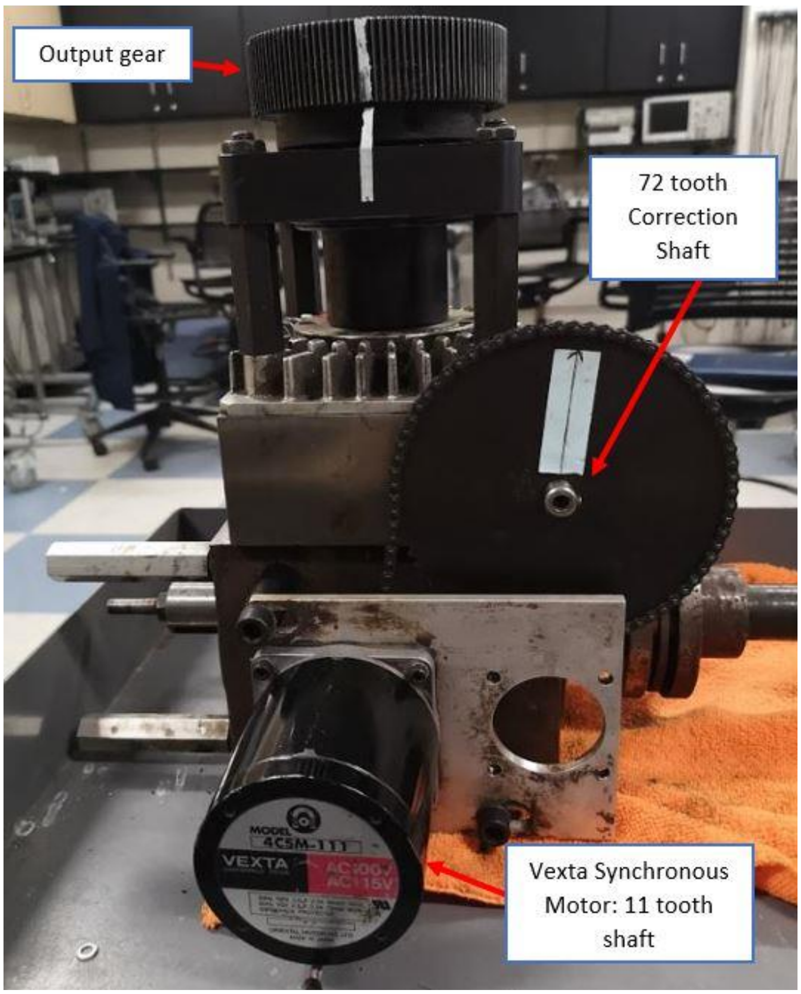


Figure 27: Gearbox & Motor

When the error is picked up, a signal is sent to the Arduino board to communicate with the motor telling how much correction is needed. The correction motor is a part of the gearbox assembly, shown in Figure 27, which will control every correction. For every 14.45 degrees that the motor shaft moves, the correction shaft will move 2.2086 degrees to get a 0.001 inch correction on the paper. For the output gear to get a 0.001 inch correction, it will move 0.0327 degrees. The output gear is directly connected to the impression cylinder and that cylinder is what will be advanced or retard for the print correction to be made. The impression cylinder has an 11 inch circumference and if that is divided by 0.001 inch for every correction then we get 11,000 pulses per revolution.

5.4. Prototype Fabrication

This section talked about the prototype fabrication of both motor and sensor part.

5.4.1. Prototype Fabrication of Motor

The team simulated the correction motor, and connected the circuit in Figure 28 according to the manual of 4CSM-111 motor (Figure 14 and Figure 15) and the motor wire diagram file from the client in Figure 17. The team simulated the circuit by using NI Multisim 14.2 first to ensure safety before connecting the physical circuit. The team connected a 2.5 M farad capacitor, MDA-1 fuse, and a 300 Ohm resistor in the motor circuit. The physical motor circuit can be seen in Figure 29.

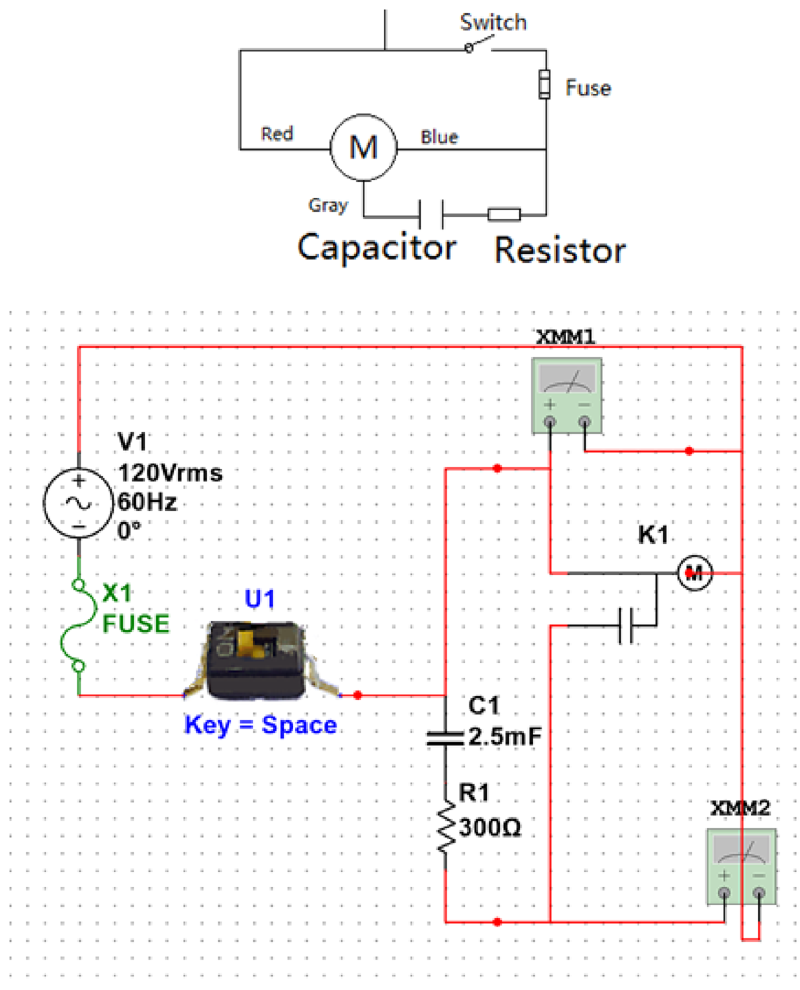


Figure 28: Schematic Diagram & Motor Circuit Simulation by NI Multisim 14.2

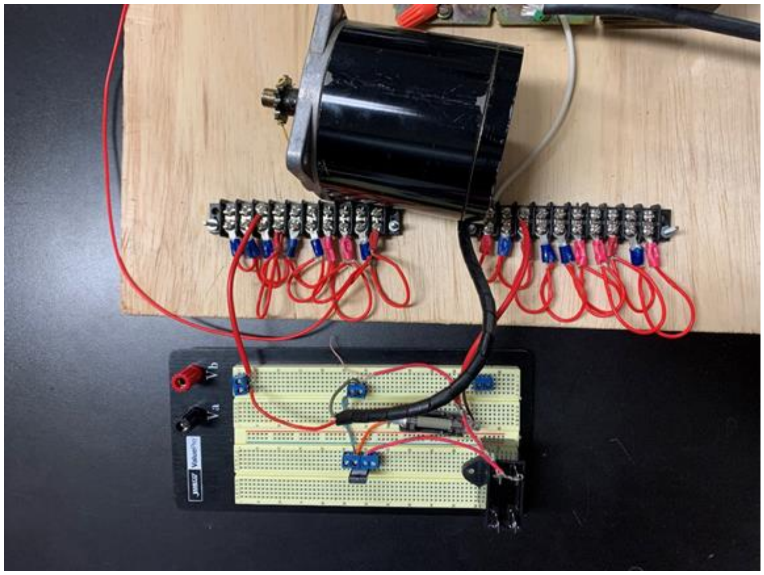


Figure 29: Physical Motor Circuit

5.4.2. Prototype Fabrication of Sensor

The team wired the sensor circuit based on the specification of BKT 67M-003/004-U-S92 shown in Figure 30. The team also made a sensor wiring diagram that is easier to follow shown in Figure 31.

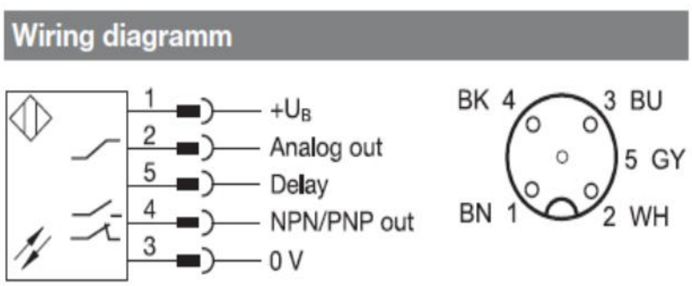


Figure 30: Wiring Diagram from Specification (in reference 2)

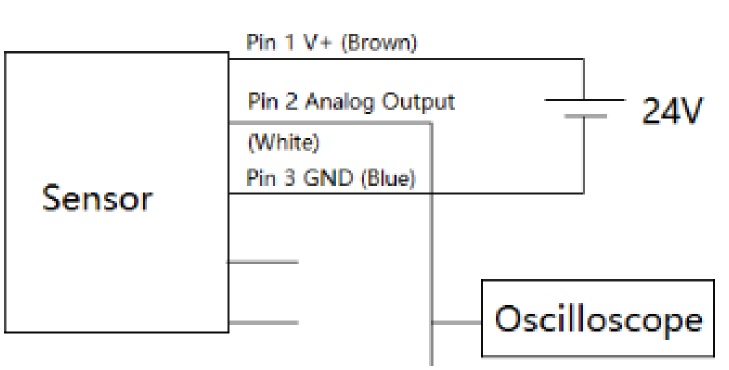


Figure 31: Actual Wiring Diagram

5.5. Prototype Testing

The prototype testing contains two portions, sensor and motor, to make sure that the sensor can accurately record and distinguish marks and confirm that the correction motor will successfully move the output gear of the gearbox in order to make corrections according to the criteria that were set. The following sections are two detailed tests regarding the sensor and motor.

5.5.1. Prototype Testing of motor

Test 1: Motor via LabVIEW

The team applied LabVIEW simulation before applying the motor code to the system. The idea behind the LabVIEW simulation can be seen in Figure 32. Connect DAQ assistant (Data Acquisition Assistant) with PC (personal computer) to transmit Data input by sensor. Through the data processing of LabVIEW and the same logic of the code in the motor portion, the motor rotation time is recorded to control the gear train, thus eliminating the error. Another DAQ assistant 2 (Data Acquisition assistant 2) connected to the PC as output controls the switch of SSR (solid state relay) with low and high levels 0 volts and 5 volts. The SSR connects the 120 volts voltage on the wall and connects the motor to control the motor's switch. When the motor turns, the gear train is driven by the gear train, and the error is eliminated by the gear train.

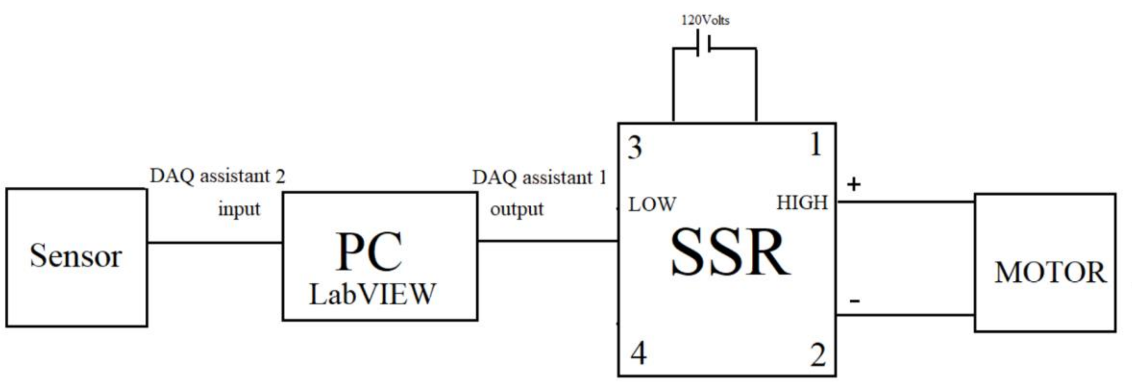


Figure 32: Schematic Diagram for LabVIEW method

In the LabVIEW file (Figures 33-35), the method of processing data is the input given by DAQ connected to the sensor, which is error, and the unit is inches. Error is divided by 0.001 to get "M," and “M” is the amount of 0.001 inch errors there are. The time to connect SSR's DAQ is controlled by multiplying "M" and "T". At the same time, an operator can set "Accurate by client." For example, when it is 3, greater than 0.003 inches, it will output 5 volts to SSR. Different precision settings meet different needs. The team succeeded in this test.

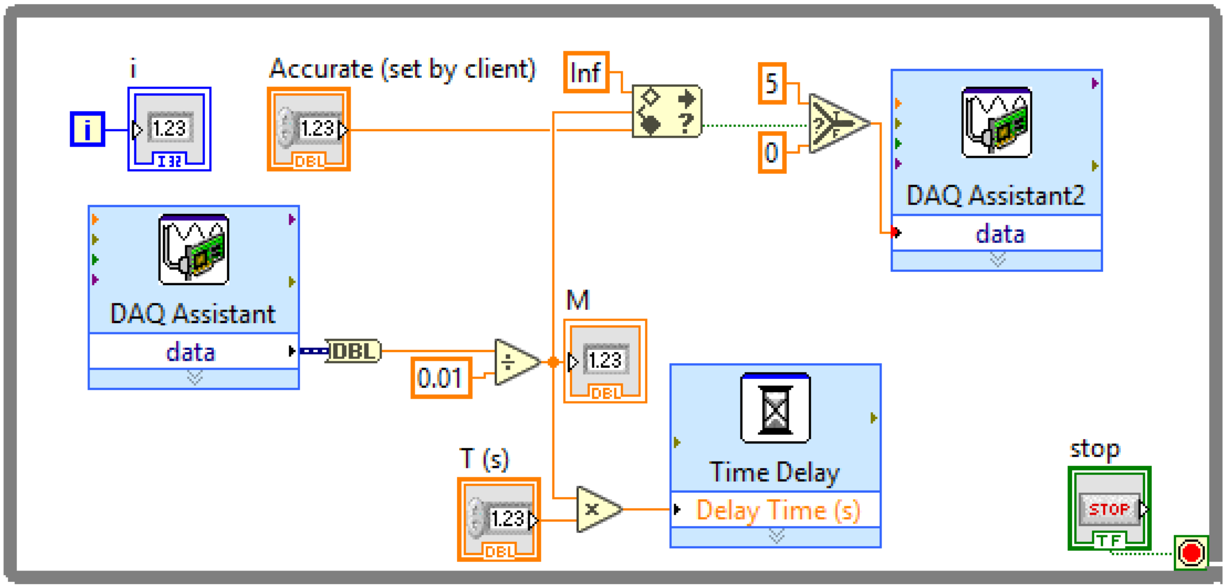


Figure 33: VI Block Diagram

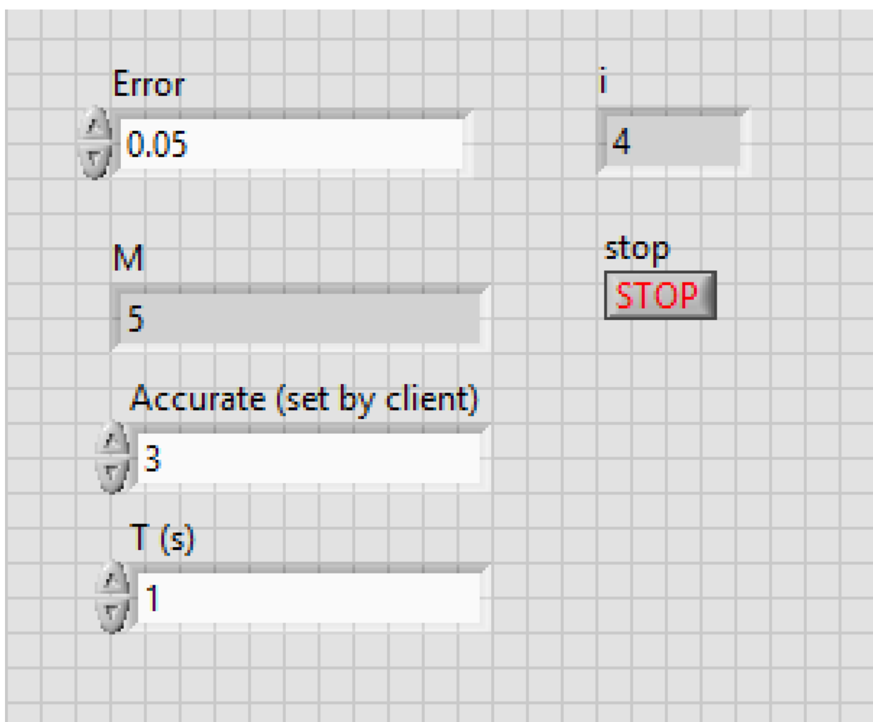


Figure 34: VI Front Panel

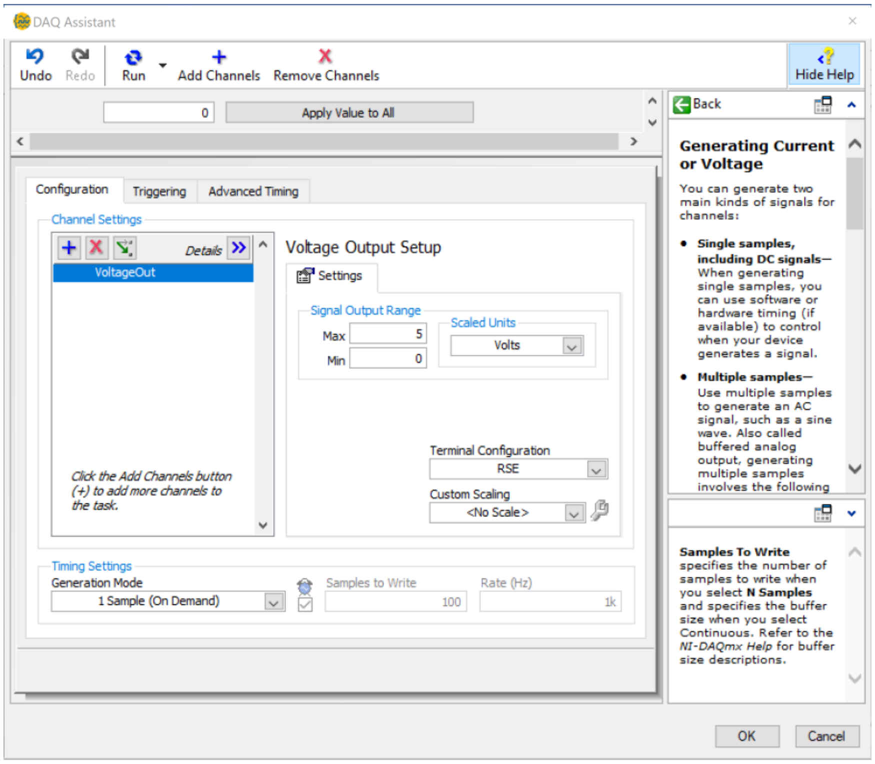


Figure 35: DAQ Channel Setting

Test 2: Motor test via Arduino

Similar to the equipment connected by LabVIEW simulation, the sensor and SSR are connected to the Arduino. The sensor inputs data, and the Arduino outputs signals to control the SSR. Meanwhile, the Arduino is connected to the PC. Voltage to the Arduino is 5 volts and the motor control code is uploaded to the Arduino. Arduino inputs high and low levels to SSR to control the motor. Similar to the LabVIEW simulation, the motor controls the gear train with a gear, and the time to control the motor's rotation and stop is calculated from the measured time "T" and data input by the sensor. This test setup thought process can be seen in Figure 36 and Figure 37 shows the physical setup.

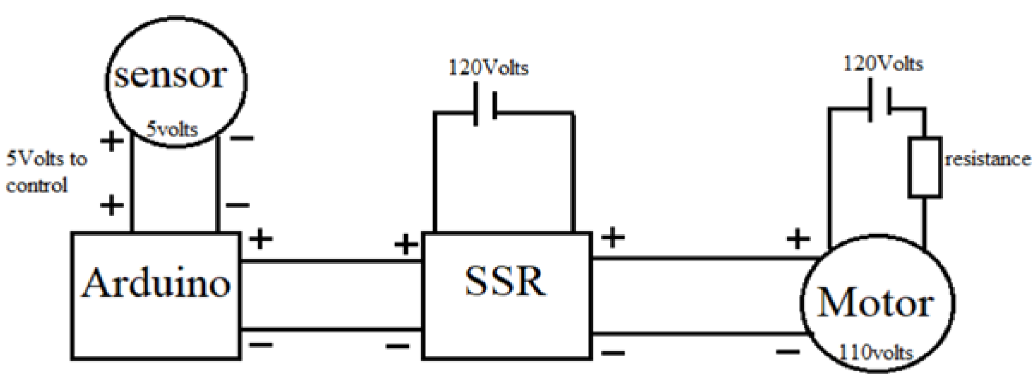


Figure 36: Schematic Diagram for Arduino SSR method

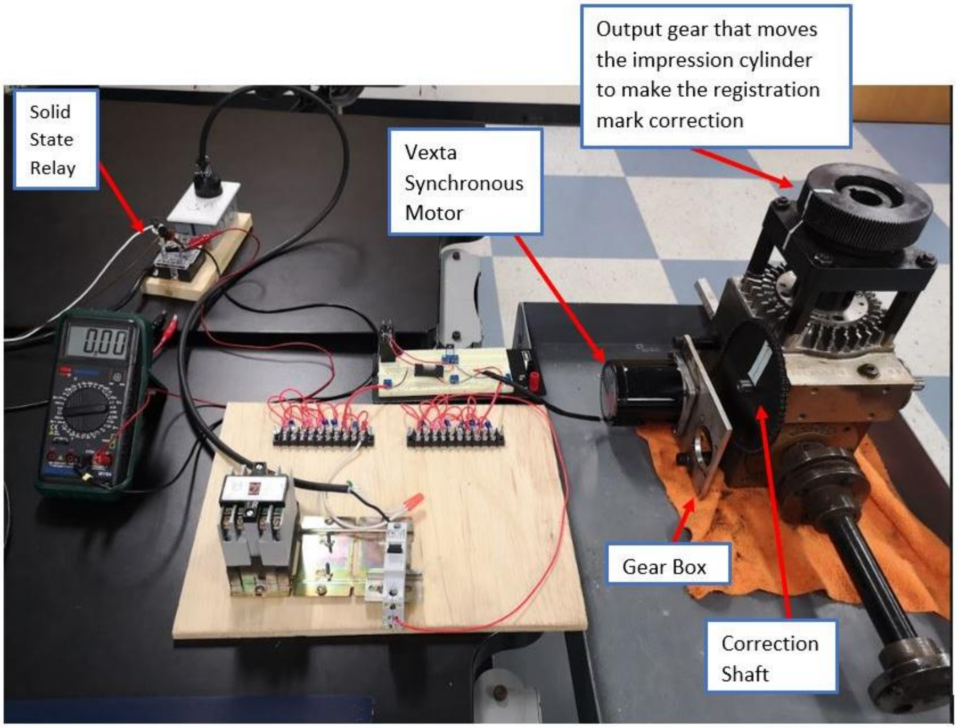


Figure 37: Motor and Gearbox Testing

5.5.2. Prototype testing of sensor

Figure 38 and Figure 39 show the physical test setup of the sensor.

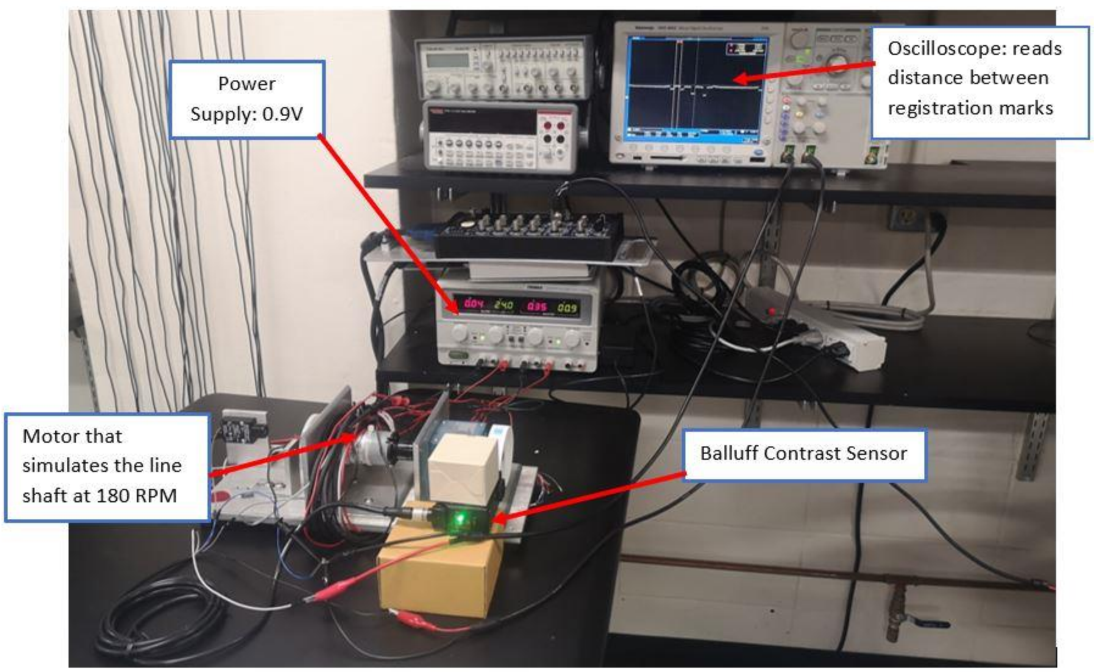


Figure 38: Sensor Testing

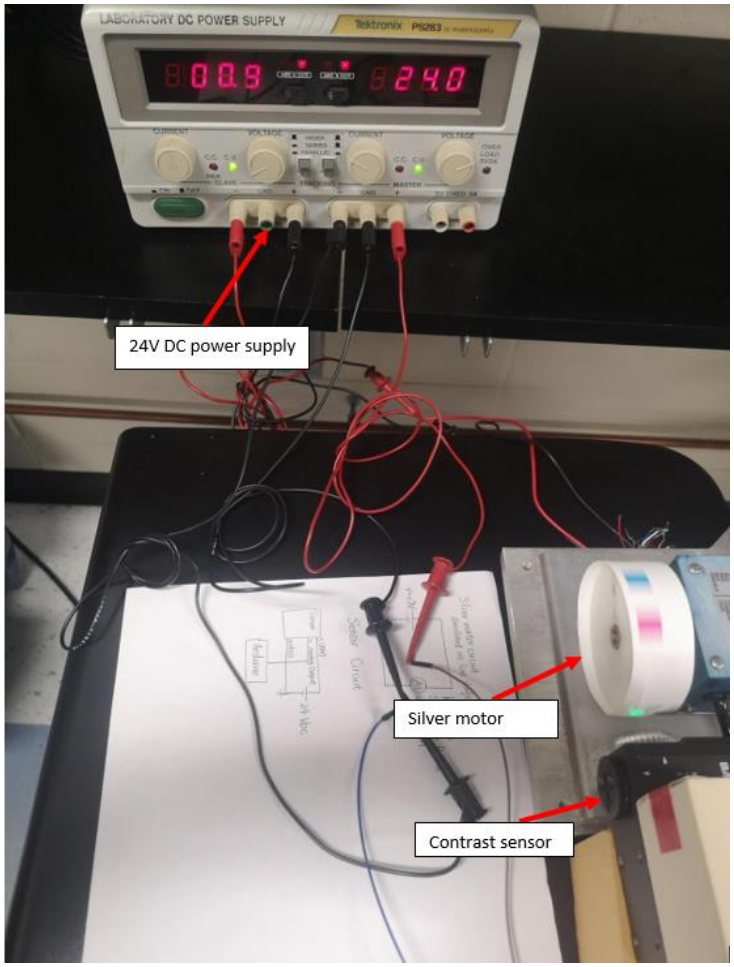


Figure 39: Sensor Testing Circuit

The team wired the sensor according to the sensor circuit, and used a 24 V power supply. An oscilloscope was used to monitor the outcome of the sensor when it detects different colors. In order to achieve accurate outputs, the distance between the registration mark and sensor's camera was within 6 mm to 12 mm. Test results were documented as shown in Table 2.

Table 2: Color Testing Limitation

Sensor light	Mark Color	Testing mark color			
		Cyan	Magenta	Yellow	Black
Green	Black	〆	√	×	√
Blue	Yellow	〆	√	√	√
Green	Magenta	×	√	×	√
Red	Cyan	√	×	×	√

Notice that the sensor will change automatically into the best testing light for the testing mark. The ⍻ mark means the voltage difference is too small, √ means it tested well and × means it did not test well. Based on the data collected, combined with the codes, the voltage value can be transferred into time. Once time is found it is multiplied by the speed of rolling paper to get distance.

5.6. Data Analysis

This section talked about the data analysis about motor part and sensor part when testing.

5.6.1. Motor Data Analysis

As for the motor, the user interface on LabVIEW is used to test the time of different error distances. Table 3 is the testing results.

$$\text{Error} = \frac{\mathit{Actual\ Distance\ on\ Gearbox} - \mathit{Input\ Error\ Distances}}{\mathit{Input\ Error\ Distances}} \times 100\% \quad (1)$$

Equation 1: Error Rate of Testing Failure Percentage

Table 3: Testing Datum of Motor Circuit by LabVIEW

Input Error Distances /inch -- mm	Actual Distance on Gearbox / mm -- inch	Error / %	Calculated Time /second	Actual Time of Correction / second	Error / %
0.118inch=3mm	3mm=0.118inch	0%	44.85	45	0%
0.0945	2.4	0%	35.91	36	0%
0.0787	2	0%	29.92	30	0%
0.0591	1.5	0%	22.44	23	2%
0.0394	1	0%	14.96	15	0%
0.0010	0.0254	0%	0.38	---	---
Average	-	0%	-	-	1%

5.6.2. Sensor Data Analysis

The oscilloscope is used to test the time difference between marks with the sensor as shown in Figures 40, 41, and 42.

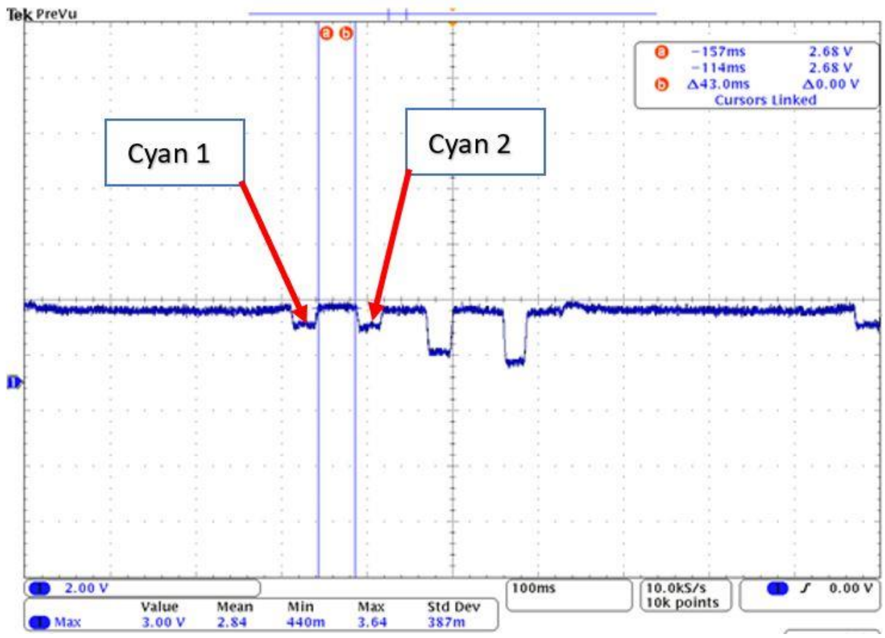


Figure 40: Sensor Time Between Cyan 1 and Cyan 2

43 milliseconds is the approximate time between Mark1 [Cyan 1] and Mark2 [Cyan 2].

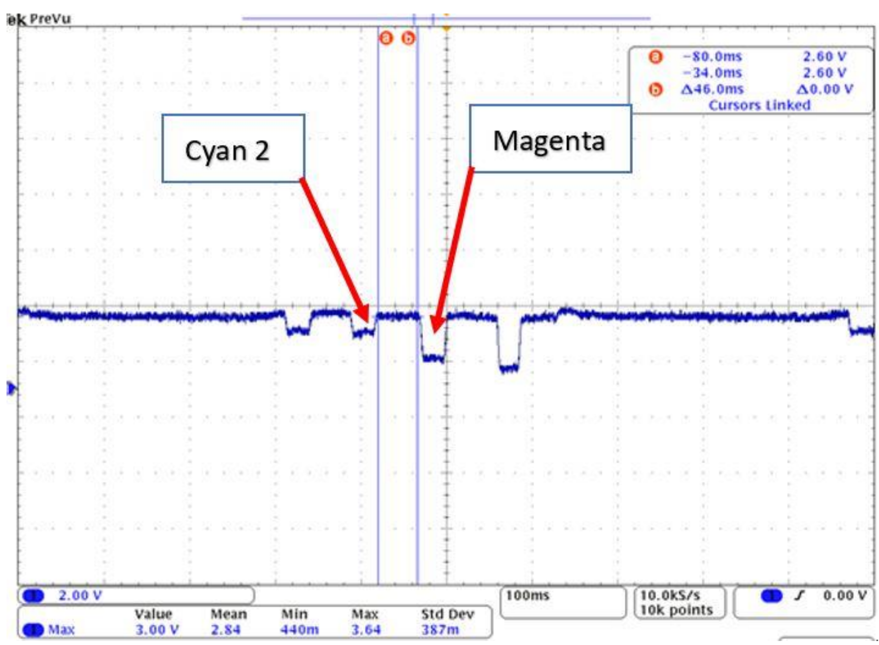


Figure 41: Sensor Time Between Cyan 2 and Magenta

46 milliseconds is the approximate time between Mark2 [Cyan 2] and Mark3 [Magenta].

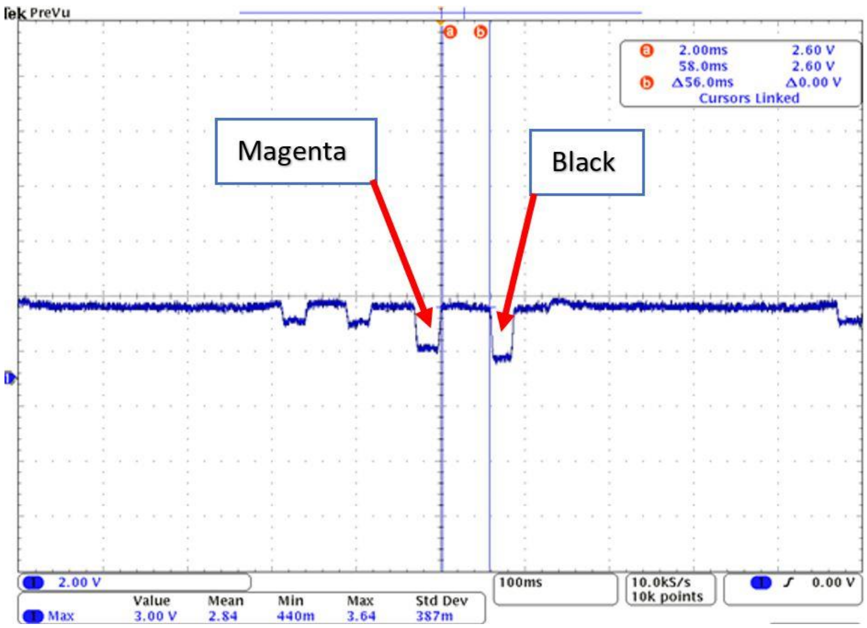


Figure 42: Sensor Time between Magenta and Black

56 milliseconds is the approximate time between Mark3 [Magenta] and Mark4 [Black].

Note: Yellow cannot be tested by the Green light from the sensor. Therefore, it doesn't show a wave on the screenshots.

5.7. Refinement

This part talked about other possibilities or options to improve our design.

5.7.1. Motor Refinement

Initially, the team thought of using PLC Arduino products from the P1AM-100 series by the AutomationDirect company because the voltage required for Arduino is 5 volts, while the normal operating voltage of motor is 115 volts. The motor can only supply 5 volts under direct control of Arduino. If P1AM series products are used, it is possible to assemble PLC Arduino of multiple slots and different slots can provide different voltages. The team found an SSR (solid state relay) that could achieve the same function to control the motor because of lower cost, ease of assembly, and smaller size.

In the LabVIEW user interface, the group added an exact precision value that the user can set. This value can be changed based on different requirements that need to be met. When the user sets the precision value to 0.003 inches, the Arduino will control the motor calibration only if the printing error is greater than 0.003 inches. It is important to note that the precision value has a lower limit that is not 0. If the precision value is set to 0.000 inches, the Arduino will control the motor to rotate continuously to control the gear train to continuously calibrate.

5.7.2. Sensor refinement

An analog output was required due to the sensor. When writing code it was not possible to use a simple function in Arduino called pulseIn() because the function requires its signal to be digital. If using the pulseIn() function it would be easy to calculate the time between sensor detections. Thus, if another sensor could be used that had a digital output signal, this would significantly save time and improve accuracy.

5.8. Results

After testing the whole system, the team manually found there is a 0.006 inch error distance from Registration Mark 3 and Reference Mark 3, and a 0.118 inch error distance from Registration Mark 1 and Reference Mark 4. This is shown graphically in Figure 43.

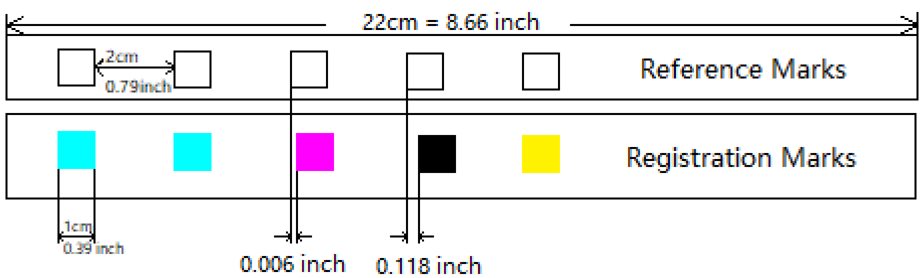


Figure 43: Result Meaning

After all the testing, the team is able to move the gear box for 0.118 inch and 0.006 inch, which proved the success of the design by correcting the errors. In order to move the gear box for 0.118 inch, it was calculated that the motor had to rotate for 45 seconds. For 0.006 inch, the motor needed to move for 2.28 seconds. A clock was successfully set to verify the time matched. Figure 44 shows the time testing setup.

Figure 44: Time testing

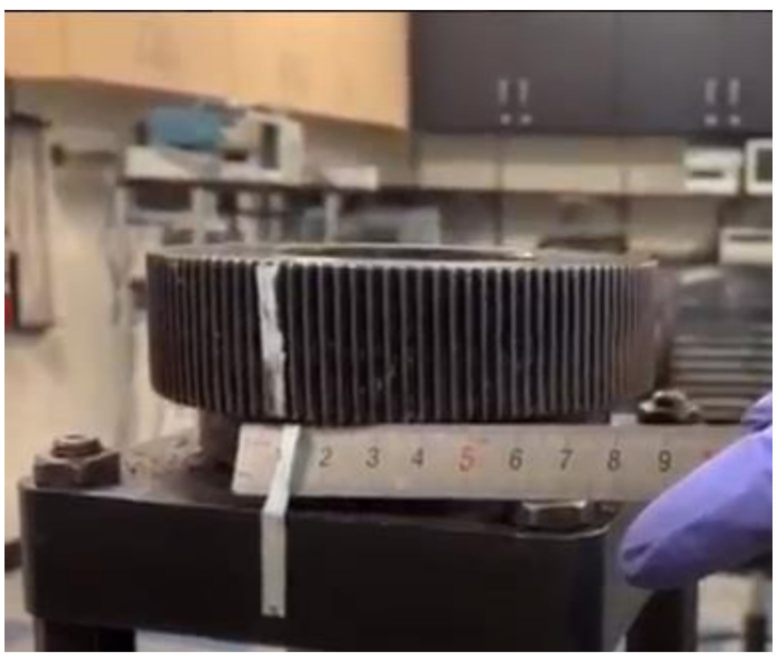


Figure 45: Measurement of Correction Distance

After testing the correction of Mark 4 (black), the ruler tested that the correction distance on the gear box was close to 3 mm. This approximately equals 0.118 inches which is similar to what the original error distance was on the physical paper wheel. This measurement can be seen in Figure 45. The team succeeded many times with 3 mm distance correction, 45 second running time, and 1.42 second waiting time.

Table 4: Testing Datum of Whole System

Times	Input Error Distances /inch	Input Error Distances /mm	Actual Distance on Gearbox / mm	Error / %	Calculated Time /second	Actual Time of Correction / second	Error / %
1	0.118	2.9972	3	0.09%	44.85	45.01	0.36%
2	0.118	2.9972	3	0.09%	44.85	45.1	0.56%
3	0.118	2.9972	3	0.09%	44.85	44.98	0.29%
4	0.118	2.9972	3	0.09%	44.85	45.01	0.36%
5	0.006	0.1524	−−−	−−−	2.28	2.41	5.70%
6	0.006	0.1524	−−−	−−−	2.28	2.49	9.21%
7	0.006	0.1524	−−−	−−−	2.28	2.52	10.53%
8	0.001	0.0254	−−−	−−−	0.38	−−−	−−−
	Average	−−−	−−−	0.09%	−−−	−−−	3.45%

Table 4 shows the testing results and the error (%) from calculated values and actual test values. According to the average of the error, we can say that the system is successful on the gear box to correct the error distance tested by the sensor.

6. Conclusion

The team was able to develop and successfully demonstrate a proof of concept model. This model showed how automatic print control could be accomplished with an Arduino microprocessor. This proof of concept model simulates one of the fourteen print stations on the Webtron press line. The following describes how the test setup operated.

1. Sensor scans marks on a wheel that simulates registration marks that would be on the printed product. The wheel spins at the same speed the paper moves on the Webtron press line.
2. The sensor relays the information picked up from the wheel and allows the programming to execute.
3. The programming code will tell the correction motor to adjust the needed amount in order to move the output gear. The correction motor is attached to a chain drive, which is attached to a tandler gearbox. The final output that the test controls is the output gear that is connected to the output shaft of the gearbox.

On the Webtron press, this output gear attaches to an impression cylinder. The impression cylinder is what is moved for a print correction to be made. All of these different components from the proof of concept communicate with one another to automatically make the output gear move the needed amount. On the Webtron press, this output gear movement would cause the print unit correction. This correction improves the print quality. Simulations of these tasks have been completed successfully in the lab.

7. Recommendations

The team has performed the needed research and has built a proof of concept model. This model acts as a starting block for implementing automatic registration to the Webtron press. Additional work will need to be completed before automatic registration will be fully operational on 14 print units and 4 die cutting/sheeting units. The team recommends to test this registration method on only a couple of units first, then slowly work up until every unit is accounted for. Using this method of testing a couple units at a time will allow the person who is installing the units to test the system and work out any errors as they happen. Troubleshooting is easier when there are fewer units involved.

8. References

1. “Low-Speed Synchronous Motors.” *Mota*, 2006, store.motadistribution.com/4csm-111.html.

2. “Product Finder.” *Balluff*, 2020, www.balluff.com/local/us/productfinder/#/ca/A0001/cg/G0103/product/F01324/variant/PV123244.

3. Project description 090519, Washburn, David (Taylor Communications), 2019.9.5 03SP2020_Taylor Communications_Automatic press print registration_SP20 _1_, Libby LoPresti

4. “Registration Sensors.” *Sort* , Sick Sensor Intelligence, 2020, sort.on.ca/sick-sensor-intelligence/registration-sensors/.

5. “Synchronous Motor.” *Synchronous Motor* , Electrical Codes, Standards, Recommended Practices and Regulations, 2010, www.sciencedirect.com/topics/engineering/synchronous-motor.

6. Taylor Communications, “The History of Taylor Communications A Tradition of Innovation in Business and the Community.” (URL: < heat.taylorcommunications.com/company/history.asp>). Accessed: 01/23/2020.

9. Engineering Drawings

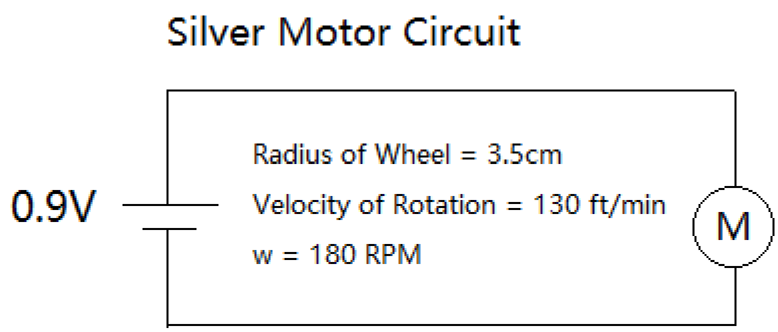


Figure 46: Silver Motor Schematic Circuit Diagram

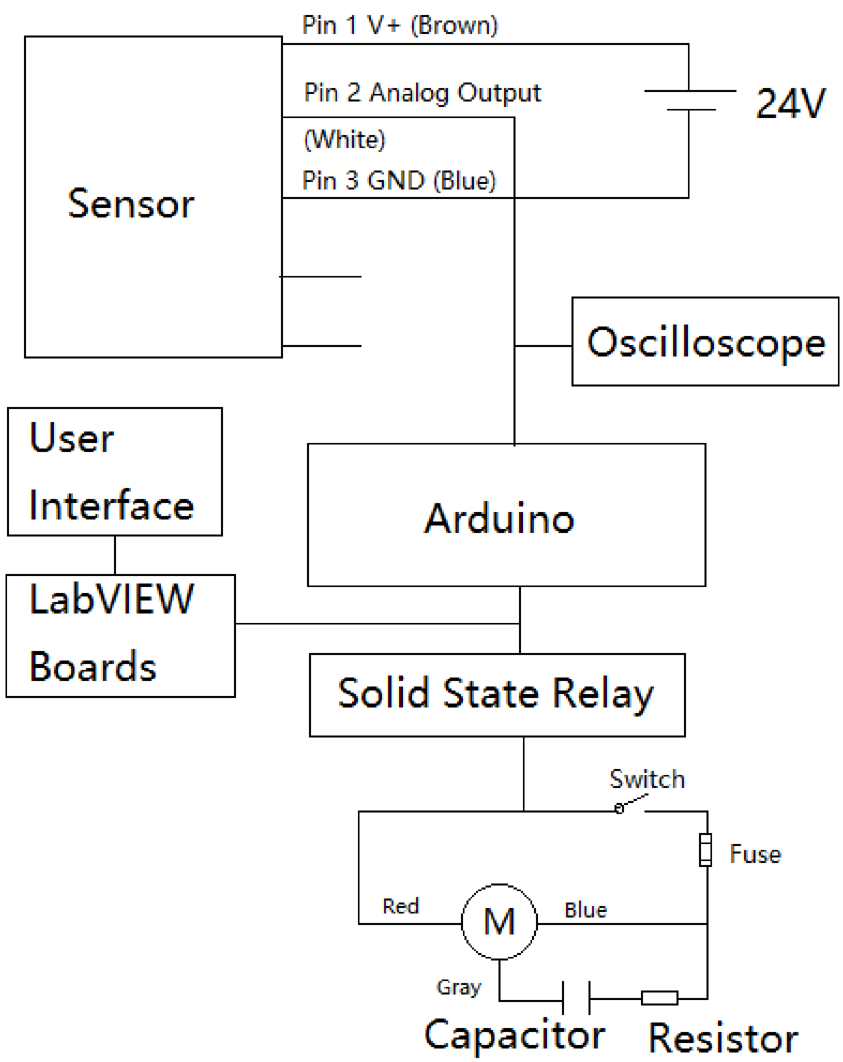


Figure 47: Sensor Schematic Circuit Diagram

10. Appendixes

Appendix A. Design Project Agreement.

This appendix shows the original project statement and explanation as provided from the industry sponsor.

Sponsor/company name	**TAYLOR COMMUNICATIONS**
Sponsor/company address	600 ALBANY ST., Dayton, Ohio 45417
Contact name	David Washburn
Phone	9372211157
Cell phone	9373611635
Email	david.washburn@taylorcommunications.com
Alternate contact name	Douglas Boyer
Alternate phone	9372211538
Alternate email	douglas.boyer@taylorcommunications.com
Purchase order contact name	David Washburn
Purchase order phone	9372211157
Purchase order email	david.washburn@taylorcommunications.com
New project schedule	Design II - MEE/ECE 432/**ET490** - 3 credit hour: $5000 - Student budget: $500-$1000 - Options to consider: research, analysis, embodiment, conceptual design selection, prototype, build, basic testing, business plan, patent research

Project title	**Automatic press print registration control**
Background - Describe why this project was initiated and the necessity	The Webtron press in the Dayton plant has 16 print/processing units that need automatic registration controls to improve the print quality. However a "standard" system available in the marketplace will cost over $200,000. Can a lower cost, modular system based on Arduino or RaspberryPi or other microprocessors be designed that would work at a much lower unit cost/
Description of project or problem	Design, assemble, test and develop a low cost, efficient automatic registration control system for the Webtron press in the Dayton IML printing plant.
Administrative deliverables	Testing schedule,
Technical deliverables - What do you expect to have at the end of the semester?	Testing protocol, set up instructions, proof of concept
Requested engineering disciplines	Computer; Electrical/Electronic
Requested skill sets	Simulation; Ergonomics/Human Factors; Appropriate Technologies; Manufacturing
Additional Skill Sets Necessary	Microprocessor and HMI programming, PID or similar control tuning
Opportunities for student learning	Dayton IML is located near campus, students are welcome to work in plant, it is planned that the team will have parts and components to prove their concepts.
Key words	Controls, logic, arduino, raspberry pi, registration, printing, Webtron, microprocessor

Appendix B. Project Participants.

This appendix shows the names, titles, e-mail addresses and phone numbers for all of those invested in the project (i.e. team members, falculity advisors, and industry sponsors).

Xulei (Celeste) Cheng
- BS in Electronic and Computer Engineering Technology
- chengx8@udayton.edu
- (937) 266-7318

Luke Knapke
- BS in Mechanical Engineering Technology
- knapkel1@udayton.edu
- (937) 234-3056

Minghao (Andrew) Li
- BS in Electronic and Computer Engineering Technology
- lim033@udayton.edu
- (937) 286-0531

Ruochen (Rachel) Pi
- BS in Electronic and Computer Engineering Technology
- pir001@udayton.edu
- (937) 286-6392

Jessica Weaver
- BS in Mechanical Engineering Technology
- weaverj9@udayton.edu
- (317)-525-4428

Joseph Untener
- Faculty Advisor
- University of Dayton Professor
- juntener1@udayton.edu
- (937) 229-2865

Dr. Philip Appiah-Kubi
- Faculty Advisor
- University of Dayton Professor
- pappiahkubi1@udayton.edu
- (937) 229-1758

David Washburn
- Industry Main Contact
- Director of Engineering
- david.washburn@taylorcommunications.com
- (937) 221-1157

Douglas Boyer

- Industry Alternative Contact
- Supervisor
- douglas.boyer@taylorcommunications.com
- (937) 221-1538

Appendix C. Proposal

This appendix has a copy of the original proposal that was created at the beginning of the project.

Taylor Communications
600 Albany Street
Dayton, Ohio 45417
January 30, 2019

Dear Mr. Washburn,

Thank you for not only the initial meeting you held with our entire group and advisor on January 21, 2020, but also for meeting with us on January 29, 2020 to clarify questions we had from the first meeting. Both of these encounters helped us grasp the background, scope, and requirements for the project. As a result of both of these meetings, all group members have expressed their excitement in being part of the team that brings an automatic print registration control system to the Webtron printing press. This project will be a challenge and everyone is looking forward to the hard work that is required to meet our common goal as well as applying the skills and knowledge we have learned thus far at the University of Dayton.

Please find attached the proposal for the automatic print registration control system. The proposal outlines the team's plans to accomplish the goals of the project, and it discusses the background of the project, scope, general approach, specifications and project deliverables. In addition, the team prepared a Gantt chart/schedule which demonstrates the sequence by which the project will be executed. Finally, information about the budget for the project has been provided. We will be happy to answer any further questions you may have after reviewing our proposal. Again, thank you for your time and the opportunity to work on this challenging but exciting project. We look forward to your feedback.

Sincerely,

Luke Knapke

Luke Knapke
Project Manager

UD #: 03SP20

Team Members:
Xulei (Celeste) Cheng
Luke Knapke
Minghao (Andrew) Li
Ruochen (Rachel) Pi
Jessica Weaver

Industry Sponsor:
Taylor Communications

Industrial Client:
David Washburn

Faculty Advisors:
Dr. Philip Appiah-Kubi
Professor Joe Untener

February 1, 2020

Executive Summary

Taylor Communications utilizes a Webtron press in the Dayton plant which has 18 print/processing units. This Webtron press currently does not print to the desired quality. Typically, sections of printout erroneously overlap in an undesired fashion which leads to poor print quality. Therefore, automatic registration controls are needed to fine-tune the Webtron press and printing processes to improve the print quality. However, a "standard" system available in the market could cost over $200,000. Therefore, the purpose of this project is to determine whether a lower cost, modular system based on Arduino or RaspberryPi or other microprocessors can be designed that would work at a lower unit cost and improve the quality of prints. The Webtron press also has four processing units that do die cutting and sheeting, not printing. The proposed registration system would be suitable for printing and die cut/sheeting. This would represent a total of seventeen (17) units to be ultimately controlled.

The scope of this project is to use Taylor Communication's Webtron press and find a method to print more efficiently and accurately. The team will use a microprocessor to allow the system to run automatically and be able to adjust for any error on its own.

The budget is $1200.00 - $1500.00 per unit. The project is expected to be completed by April 30, 2020, at which time all deliverables detailed later in this proposal will be delivered to the client. A project schedule is attached to portray the sequence of execution.

The project will present many benefits to Taylor Communications. Developing an automated registration control system for the Webtron press machine could lead to better printing quality, which will minimize cost and simplify operating procedures thus potentially leading to higher customer satisfaction.

Thank you again for the opportunity to work on this exciting project.

Table of Contents

1 Introduction

This section contains information about the purpose of the project, the history of Taylor Communications, and the background of the project.

1. Purpose

The purpose of this project is to design and fabricate a proof of concept model of a low cost automatic print registration system using micro-processors or controllers such as an Arduino to control motors in order to improve the quality of images. This is expected to improve the print quality of images from the Webtron press at Taylor Communications.

2. History

According to the *Taylor Communications Company* [1], Taylor Communications is a printing company, which was founded by John Q. Sherman in 1912. The company is located at Dayton Ohio. There are a lot of inventions created in this company. The most representative product is the Standard Register invented by Dayton native Theodore Schirmer, this invention is also called the pin-feed autographic register. Due to the strong function the register has, it quickly became universal, and there are many applications based on this invention as well. For example, because it is easy for a person to copy receipts, the register can make business transactions easier and more secure. Because of this practical invention and many other achievements the company made, Taylor Communications defeats many competitors. The company has evolved continuously as well and is capable of offering all kinds of products such as durable safety labels, computers and printing labels as well.

3. Background

The Taylor Communications printing plant in Dayton, Ohio has a 14-color printing press that prints window decals for a major credit card company. However, the print quality does not meet the desired level of quality due to unaligned print registration. As a result, Taylor Communications is seeking a new approach to improve the print quality, which may be done by utilizing a modular system based on Arduino or Raspberry Pi or other microprocessors. Even though there are existing systems in the market that could print to the desired level of quality, they cost over $200,000. Therefore, a design that would work at a much lower unit cost is highly desired. The press needs to be improved with an automatic circumferential registration control to meet customer expectations. This requires the controls to be able to monitor and modulate 13 print units, the first print unit provides the reference mark, and 4 die cut stations.

2 Scope

The scope of this project is to study Taylor Communication's Webtron press and develop a method to print more efficiently and accurately. The team is going to scan and monitor the printed product through a sensor or camera of sorts, thus automatically gaining control of the printing location. By using current technology microprocessors the motor will automatically move forward or back allowing the press to overlap with the previous layer on different unit layers. The team may use a microprocessor like Arduino or RaspberryPi to control the press allowing the low-cost modular system to operate more efficiently. Another option is to use a PLC, which is a type of computer designed specifically for industrial applications. The team is going to explore all options before making a decision. The registration control system that is designed must work for circumferential registration only. The team will not be adjusting the print registration laterally in this project. In addition, the team will complete a proof of concept, prototype proving their design, complete setup instructions, and also set up a testing protocol. The prototype includes a trial run of 1-4 units.

3 Discussion

This section presents the general approach, benefits and feasibility, project specifications, deliverables, resources, schedule, and the budget for the project. Each of these has been discussed in detail.

3.1 Procedure and General Approach

The project will be accomplished in a thought-out timeline ensuring that each task is completed and that the project continues to move at a pace in which all the work will be accomplished. Below the project is broken down into six phases. Each of the phases will be presented again in the project schedule (Figure 1 and Figure 2) further into the report.

3.1.1 Phase I – Research

The team will divide unknown concepts among each other and will research information in the areas that apply to them in the project. The team then will hold a meeting to gather information and briefly discuss the individual conclusions to ensure alignment across all team members. This information gathered will act as the foundation of the project.

3.1.2 Phase II – Conceptual Design Development

The team will conduct preliminary brainstorming, generate opinions, and select more advantageous parts of the design to form the optimal solution. After the meeting, the group will come up with the basic plan and further implement it on the basis of the correct principles. The team will look for experts such as professors to confirm the rationality and validity before proceeding to the next step.

3.1.3 Phase III – Design Selection and Analysis

The team will create a design matrix in table form. The design criteria are listed in the column, while the solutions or materials are listed in the row. The team then will assign different weights to the design criteria based on the requirements of the client. The design with the highest score after analysis will be selected as the solution for final implementation.

3.1.4 Phase IV – Proof of Concept, Prototyping, and Testing

This phase of the approach is where the design that is chosen will be proven to the client by presenting the supporting technical work and research. Not only will this phase be used to explain why our approach will work theoretically (proof of concept), but also through the use of prototyping and testing the client will be able to see the design's success first hand.

3.1.5 Phase V – Technical Drawings

Based on the schematic diagram of the machine given by the customer, the team will deliver the programming code and PLC block diagram of the prototype, which is the internal circuit diagram, may be drawn by SIMATIC Controller software.

3.1.6 Phase VI – Evaluation

The evaluation phase is when every aspect of the project comes together. Proof of concept, prototyping and testing will be already completed. All the supporting documents that prove the numbers and calculations will also be shown at this phase.

3.2 Benefits and Feasibility of the Project

This section discusses the benefits and feasibility of the project. In the visit to Taylor Communication's facility, it was found that the Webtron press is manually controlled by workers. The press's print towers can currently be adjusted in two different ways. The first option is to manually adjust the tower with a lever that is attached at each unit. The second option is for the operator to use a display screen that has the controls of all the presses connected in one location. The operator can adjust each unit from this one physical location. This movement is what aligns the printing and thus increases the quality of the printing. The plate cylinder is an important component when controlling a print tower's plate position and this project will help to design an automation registration system to make the Webtron press automatically adjust the position/registration to achieve better printing quality. This will greatly save manual labor, simplify the operational steps, and improve the print quality. In addition, better product quality is conducive to improving the reputation of the company and brand promotion. Last but not least, the project will reduce printing costs by minimizing waste of material. Existing systems on the market would fix the issues the Webtron press is encountering but cost over $200,000. By using microprocessor technology the cost will be significantly less per unit.

On the other hand, knowledge, experience, and engineering ethics may sway the approval of the project. The project is feasible because the team members have diverse backgrounds in mechanical, and electrical engineering technology, which will be a great asset in the execution of the project. The team members have relative experience working on similar microprocessor projects such as Arduino, Raspberry Pi and PLC. They are responsible and prepared to achieve the deliverables. In addition, the team will harness the expertise of Taylor Communications and the University of Dayton to provide to successfully complete the project. We believe that through systematic study and experimentation, we will successfully achieve our deliverables.

3.3 Project Specifications

This section will include the design function, design requirements, and design criteria that Taylor Communications have specified for this project.

3.3.1 Design Function

The Design Function section discusses what the final design must be able to accomplish. The intended design of our microprocessor will be able to hold registration color to color at high production speeds. The goal of the microprocessor is to improve the print quality of pictures by picking up a signal from a sensor and automatically adjusting to the correct location.

3.3.2 Design Requirements

The Design Requirements section will discuss the quantitative requirements and conditions in which the design must operate.

The production line runs at speeds up to 150-250 feet per minute therefore this design must be able to hold registration color to color within ± 0.005.” It must also hold registration die cut/sheeting to print ± 0.030” at the same production speed.

3.3.3 Design Criteria

The Design Criteria talks about the standards of how the project will be judged. At the end of this project, a rotary encoder should be able to be mounted on the press line shaft and act as a reference signal generator for the microprocessor. The design must be able to repeat variable lengths and change size on request. Initial set up should be easy and preferably be able to be used with a HMI display. Operation of the registration system should not add significant time or waste to the system.

3.4 Project Deliverables

This section contains the information about the administrative deliverables and the technical deliverables.

3.4.1 Administrative Deliverables

A final project report including this proposal will be delivered to the clients at Taylor Communication. Biweekly written reports and presentations will be put together and made for stakeholders. In addition, a complete Gantt chart that shows the project schedule will also be delivered and can be seen in Figure 1 and 2 of this proposal.

3.4.2 Technical Deliverables

A fully functional program based on microprocessors or controllers will be delivered to clients as well as a tested prototype. Completed schematics, testing protocol, part details, set up instructions and proof of concept will also be delivered.

3.5 Resources

This section is showing the personnel, facilities and equipment supported by the Automatic Print Registration Control System project.

3.5.1 Personnel

- Xulei (Celeste) Cheng: BS in Electronic and Computer Engineering Technology
- Luke Knapke: BS in Mechanical Engineering Technology
- Minghao (Andrew) Li: BS in Electronic and Computer Engineering Technology
- Ruochen (Rachel) Pi: BS in Electronic and Computer Engineering Technology
- Jessica Weaver: BS in Mechanical Engineering Technology

3.5.2 Facilities and Equipment

- Kettering Labs Rooms 108, 124, 128, 134, 136, 138, and 141
- Roesch Library at the University of Dayton
- Taylor Communications Webtron press

3.6 Schedule/Gantt Chart

Figure 1 on the following page is showing the tentative schedule of the project.

Taylor Communications Senior Design	**73 days**	**Tue 1/21/20**	**Thu 4/30/20**
Phase I - Research and Brainstorming	11 days	Tue 1/21/20	Tue 2/4/20
First Meeting with Client	1 day	Tue 1/21/20	Tue 1/21/20
Second Meeting with Client	1 day	Wed 1/29/20	Wed 1/29/20
Phase II - Conceptual Design Development	19 days	Thu 1/30/20	Tue 2/25/20
Peer Review 1	1 day	Thu 2/6/20	Thu 2/6/20
Phase III - Design Selection and Analysis	6 days	Thu 2/20/20	Thu 2/27/20
Peer Review 2	1 day	Tue 3/3/20	Tue 3/3/20
Phase IV - Prototyping and Testing	21 days	Tue 2/25/20	Tue 3/24/20
Phase V - Technical Drawing Detailing	16 days	Tue 3/17/20	Tue 4/7/20
Peer Review 3	1 day	Thu 4/2/20	Thu 4/2/20
Phase VI - Evaluation and Reporting	13 days	Tue 3/31/20	Thu 4/16/20
Final Report	1 day	Wed 4/29/20	Wed 4/29/20
Final Peer Review and Log Books	1 day	Thu 4/30/20	Thu 4/30/20
Final Presentation	1 day	Thu 4/30/20	Thu 4/30/20

Figure 1 Tentative Schedule

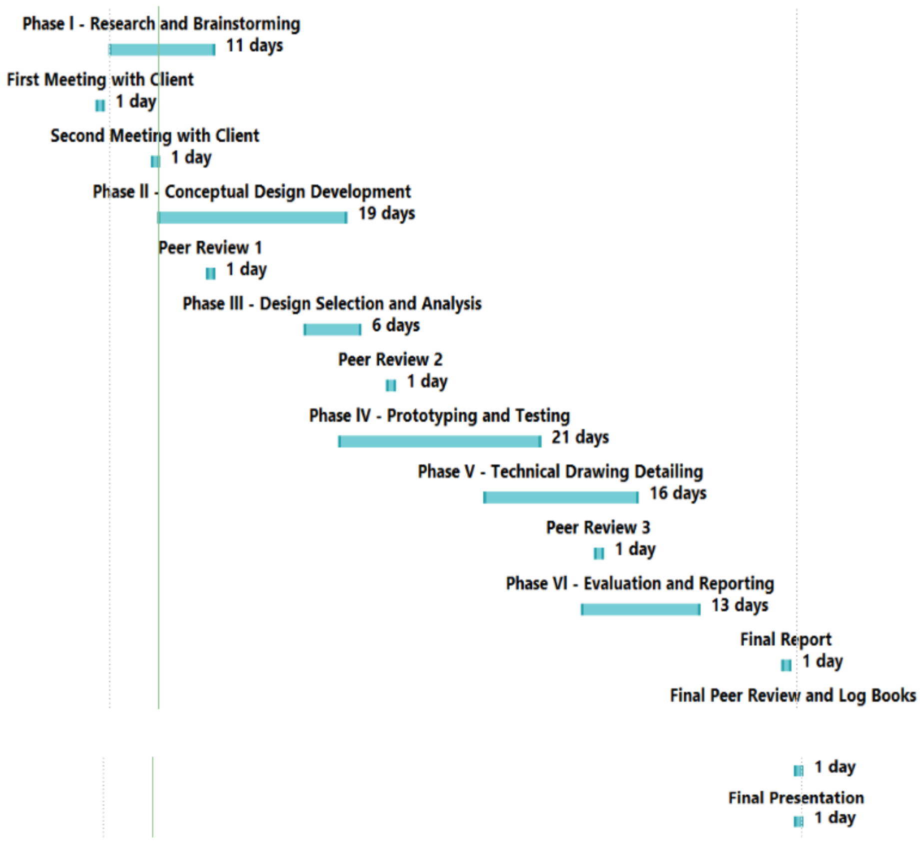


Figure 2 Tentative Gantt Chart

3.7 Costs and Fees/Budget

The financial budget has two different sections; the budget from the University of Dayton and the budget from the client, Taylor Communications. The University of Dayton provides $1000.00 for the project and can be used in any way that may add value or help the project succeed. This money could be spent on items such as software devices, printed control boards, prototyping, and research material.

Taylor Communications has also provided a budget for the team to use for their benefit. Similarly to the University of Dayton, Taylor Communication has provided $1000.00 to be used for prototyping, trying new technologies, research, and general experimentation. In addition to this Taylor Communication is providing $1200.00 - $1500.00 to use per printing tower and die cutting unit. There are a total of 13 printing towers and 4 die cut stations for a total of 17 units. This correlates to $27,000.00 to be used on printing towers and die cutting stations.

4 Conclusion

The goal of the project is to develop a microprocessor such as an Arduino for the Webtron press to accurately and efficiently print window decals for Taylor Communications. The team will use the microprocessor technology to develop an automated registration system for the Webtron press that will make the needed adjustments to improve the print quality for all 14 print units and 4 die cutters. There will be no need for manual adjustment once the project is complete. The improvement of quality, the reduction of cost and the convenience of operation will be presented in the expected products. The budget for the project is $1,200.00 to $1,500.00 per unit. Deliverables include a proof of concept, tested prototype, set up instructions, and a testing protocol. The registration control system that is designed must work for circumferential registration only. Taylor Communications will also have access to any and all engineering files such as schematics, part details and program files. The project is expected to be completed on April 30, 2020.

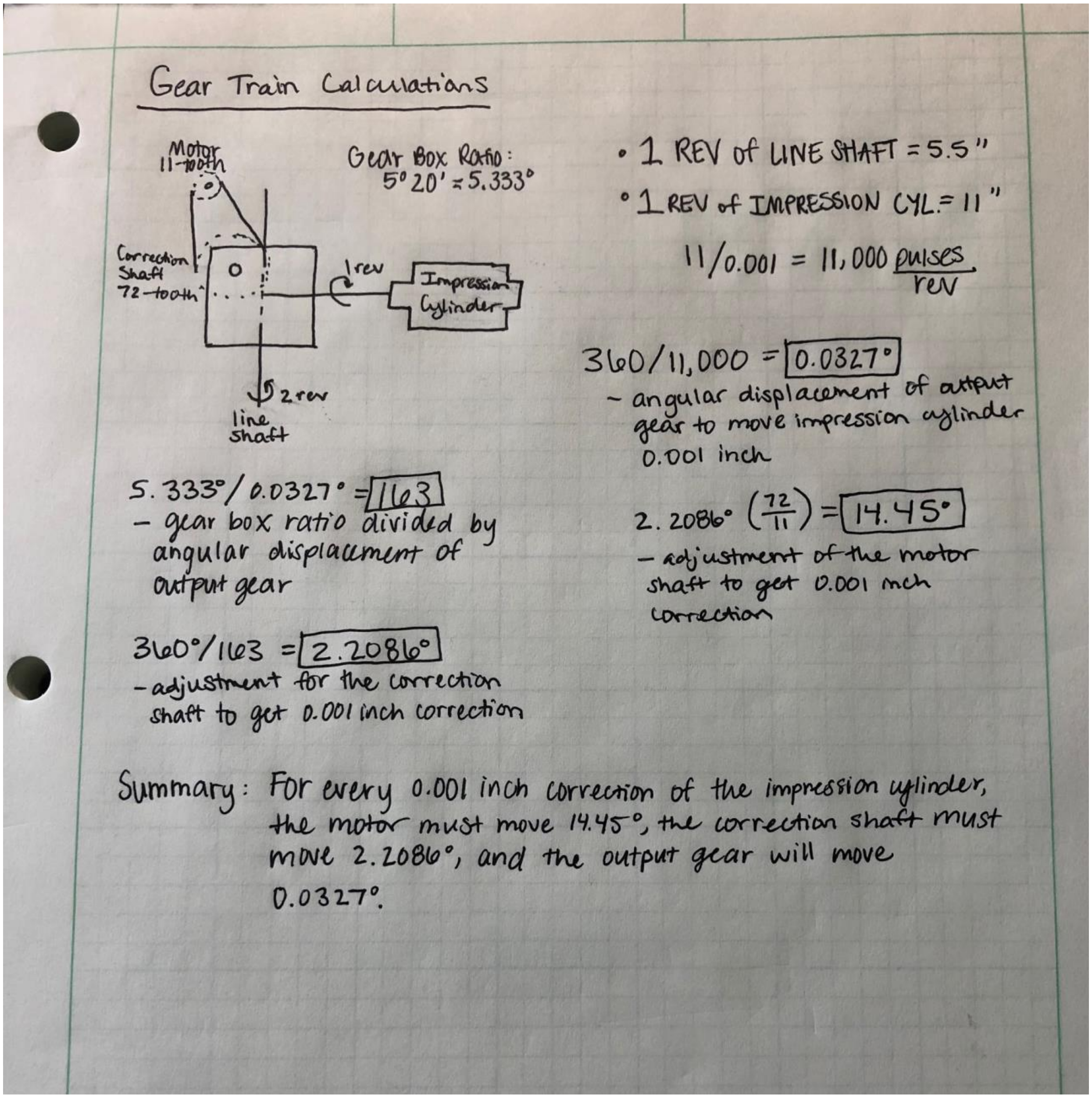
5 Reference

1. Taylor Communications, “The History of Taylor Communications A Tradition of Innovation in Business and the Community.” (URL: < heat.taylorcommunications.com/company/history.asp>). Accessed: 01/23/2020.

2. Project description 090519, Washburn, David (Taylor Communications), 2019.9.5 03SP2020_Taylor Communications_Automatic press print registration_SP20 _1_, Libby LoPresti

Appendix D. Calculations.

This appendix will show the handwritten calculations that were performed throughout the duration of the project.



Gear Train Calculations

Gear Box Ratio: $5°20' \approx 5.333°$

- 1 REV of LINE SHAFT = 5.5"
- 1 REV of IMPRESSION CYL. = 11"

$11/0.001 = 11{,}000 \dfrac{\text{pulses}}{\text{rev}}$

$360/11{,}000 = \boxed{0.0327°}$

– angular displacement of output gear to move impression cylinder 0.001 inch

$5.333°/0.0327° = \boxed{163}$

– gear box ratio divided by angular displacement of output gear

$360°/163 = \boxed{2.2086°}$

– adjustment for the correction shaft to get 0.001 inch correction

$2.2086° \left(\frac{72}{11}\right) = \boxed{14.45°}$

– adjustment of the motor shaft to get 0.001 inch correction

Summary: For every 0.001 inch correction of the impression cylinder, the motor must move 14.45°, the correction shaft must move 2.2086°, and the output gear will move 0.0327°.

Appendix E. Financial.

Individual screenshots of the financial report can be seen in section “4.7 Cost and Fees/Budget.” To view the full financial breakdown in an Excel file format, see the following link in this appendix.

Up-To Date Expense Report.xlsx

Appendix F. Source Codes

This appendix will show the code that was written for each portion of the project.

Automatic_Print_Registration_Control_System_Project | Arduino 1.8.12

File Edit Sketch Tools Help

Automatic_Print_Registration_Control_System_Project

```
/**********************************************************************
 *
 * Project: Automatic Print Registration Control System
 * Sponsor: Taylor Communications
 * Author: Xulei Cheng, Minghao Li, Ruochen Pi
 * Date: April 6, 2020
 * Function: Use a Balluff sensor controlled by Arduino Uno board
 * to test the difference of distance between 4 marks
 * to make the VETXA motor to correct the distance.
 *
 *********************************************************************/



/*********************************************************************
 *
 * Initialization
 * Function: Initialize the global variables which will be used in
 * the following functions.
 *
 *********************************************************************/
//Global variables
//sensor part
int analogPin = A3; //sensor receiving pin -- input Pin
int i; //i is a variable of index counting
//time storage
int time1 = 0;
int time2 = 0;
int time3 = 0;

//control part
int count; // count how many times did the sensor test the same distance.
float distance_difference[3] = {0};


//motor part
float M; //M is how many 0.001 in error distance
float average_error_distance; //average value of sensor input after 3 times of testing
float T; // time to launch motor to move 14.5 degree  ***constant
float correct_time; // time to launch motor to correct error
int outputV; //outputV is the output value of slot 1, channel 2
//set I/O
int digitalPin = 3; //motor control pin -- output pin


//loop for one time
void setup()
{
  //e=0.006;
  Serial.begin(115200);  // initialize serial communication at 115200 bits per second
  pinMode(digitalPin, OUTPUT);    // sets the digital pin ~3 as the output

}
```

```
//continuous loop
void loop()
{

 for(count = 0; count<3; count++)
 {
/*********************************************************************
*
* Local Variable Initialization
* Function: each loop will initialize the local vatiables.
*
*********************************************************************/
  int t=0;
  float data[5001] = {0};//array_name[array_size] with initialization of zero
  float sum_three_data[1667] = {0};// sum of every three datum from sensor
  float average_three_data[1667] = {0};// the average value of every three datum

  //time storage when mark is tested
  int time_storage[4]={0}; // there are 4 variables in the array to store time.

  float lineshaftspeed = 0;//speed should be 180RPM[r/min]=3 r/sec.
  float reference_distancel_2 = 3; // distance <cm> between reference marks [markl and mark2]
  float reference_distancel_3 = 6; // distance between reference marks [markl and mark3]
  float reference_distancel_4 = 9; // distance between reference marks [markl and mark4]
  float tested_distancel_3 = 0; // distance between markl and mark3
  float tested_distancel_4 = 0; // distance between markl and mark4

  //control part
  int count = 0; // count how many times did the sensor test the same distance.

/*******************************************************************************
 * sensor process of distance measurement
*******************************************************************************/
  for(i=1; i < 5001; i++) //use an array named data[] to store 5001 analog data.
  {
    // sensor analog input temporary storage
    data[i] = analogRead(analogPin);
    // read the voltage of sensor output as the input of Arduino

    // use average of every three datum as input value to reduce testing error
    if(i%3==0)
    {
      sum_three_data[i/3] = data[i-2]+ data[i-1] + data[i];
      average_three_data[i/3] = (sum_three_data[i/3])/3;
    }

    // judge if the sensor tests the mark
    if(average_three_data[i] < (0.8*average_three_data[i-1]))
    {
      time_storage[t] = millis(); // record the time since sensor gets the first mark.
      t++;
    }
  }
```

```
// data calculation
time1 = time_storage[1] - time_storage[0]; //time between mark1 and mark2
time2 = time_storage[2] - time_storage[0]; //time between mark1 and mark3
time3 = time_storage[3] - time_storage[0]; //time between mark1 and mark4

// line shaft speed calculation
// Give/Assume distance between reference marks[left eadge of mark1 and rigtht edge of mark2] is 3cm.
lineshaftspeed = reference_distance1_2*10/time1;//unit <mm/ms> which should be closed to 0.66mm/ms
tested_distance1_3 = lineshaftspeed*time2; //unit <mm/ms * ms = mm> [distance between mark1 and mark3]

//calculate distance between marks

// if test error = 0.006 inch = 0.1524 mm, use following judgement.
if(tested_distance1_3 > reference_distance1_3) // reference_distance1_3 = 6cm
{
  // when motor moves CCW, gear box moves CW
   distance_difference[count] = tested_distance1_3 - reference_distance1_3;
   //unit <mm/ms * ms = mm> [distance between mark1 and mark3]
}
else
{
  // when motor moves CW, gear box moves CCW
   distance_difference[count] = reference_distance1_3 - tested_distance1_3;
   //unit <mm/ms * ms = mm> [distance between mark1 and mark3]
}
/************************************************************************************
 * if test error = 0.118 inch = 3mm, use following judgement instead of the former one.
 *

   if(tested_distance1_4 > reference_distance1_4) // reference_distance1_4 = 9cm
  {
   // when motor moves CCW, gear box moves CW
    distance_difference[count] = tested_distance1_4 - reference_distance1_4;
    //unit <cm/ms * ms = cm> [distance between mark1 and mark4]
  }
  else
  {
   // when motor moves CW, gear box moves CCW
    distance_difference[count] = 6 - tested_distance1_4;
    //unit <cm/ms * ms = cm> [distance between mark1 and mark4]
  }
*/


}//count three times then move the motor if needed

//After count 3 times of distance_difference, calculate the average of error distance.
average_error_distance = (distance_difference[0]+distance_difference[1]+distance_difference[2])/3;
```



```
}//count three times then move the motor if needed

//After count 3 times of distance_difference, calculate the average of error distance.
average_error_distance = (distance_difference[0]+distance_difference[1]+distance_difference[2])/3;
```

```
/*****************************************************************************
* motor process of error distance correction
*****************************************************************************/
// data calcaulation
  T=0.38; //T is tested by LabVIEW with unit <s>
  //M = e/0.001;// M is how many 0.001 inch in average erroe distance
  M = average_error_distance/25.4/0.001; //change the unit of mm to inch
  correct_time = M*T;//calculate time of correction <s>

// data judgement
  if(M>3)
  {
    digitalWrite(digitalPin,HIGH); // write output to ssr
    delay(correct_time*1000);            // waits for a second
  }
  else if (M<-3)
  {
    digitalWrite(digitalPin,HIGH); // write output to ssr
    delay(correct_time*1000);            // waits for a second
  }
  else{
     digitalWrite(digitalPin,LOW);  // sets the analog pin 3
  }

  //e=0.001;
  average_error_distance = 0.001; // stop the motor after correction
  delay(1.42*1000);//the time of every 3 times of tesing
}
```

```
Done Saving.

Sketch uses 4084 bytes (12%) of program storage space. Maximum is 32256 bytes.
Global variables use 204 bytes (9%) of dynamic memory, leaving 1844 bytes for local variables. Maximum is 2048 bytes.

187                                                                    Arduino Uno on COM3
```