CHAPTER 1 – BACKGROUND OF THE STUDY

Introduction

When it comes to cleaning floors, people usually clean floors manually using any available cleaning tools. If in case people cannot clean the area because they're busy, or if it is in the middle of an urgent job, people would buy an automatic floor cleaner. When it comes to repairability, these devices are often locked for repair, making it harder to replace anything. This product was designed because people want to clean faster, they are tasked to work in other things, they want an easy-to-repair floor cleaner, or they struggle to find a cheap and open-source vacuum and mop cleaner.

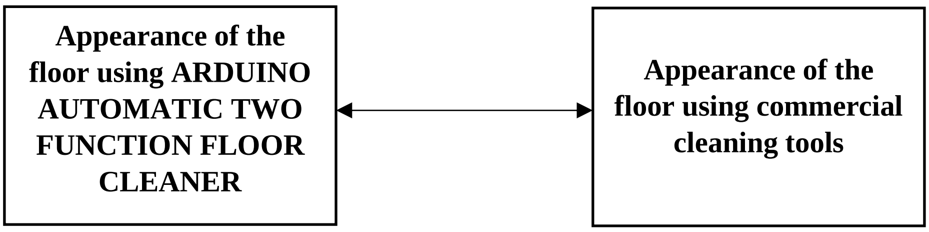
Some people in the Southeast Asia are cleaning floors to keep those areas hygienic. But others are not, based from a source from *BusinessWorld*, in the Philippines, where most areas in Metro Manila and other populous locations are filled with garbage which will end up in walkways. While this can be solved by using a mop and vacuum, with Arduino Automatic Two-function Floor Cleaner, it can clean walkways the same way as a commercial mop and vacuum.

In Oriental Mindoro, most of the floors are already clean, but schools and colleges in that province may require frequent floor cleaning, and so manual cleaning is hard to impossible due to a large number of students walking and the time allotted for janitors to clean walkways and indoor campus floor. In the case of malls/supermarkets in that province, while it is looking clean, it requires frequent cleaning. Since there are no janitors in those areas, only robot-based cleaners can clean the area, but those are expensive, as per article from *PCMag*, robotic cleaners usually cost anywhere from $279.99 (Php15,545.60 in 2023) to a thousand dollars, the

Arduino-based floor cleaners can solve some problems because it is a cheap, open-source product.

In the community, a lot of them didn't require a robotic cleaner since the houses are small, but the owners of a big mansion require the use of robotic cleaner. The Arduino-based floor cleaner can give the user a cheaper, faster and more effective cleaning.

Conceptual Framework



Statement of the Problem

Specifically, this research study answers the following question:

- Is there a significant difference between the low-cost ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER and the commercial mop and vacuum in terms of appearance of the floor after cleaning?

Statement of the Hypothesis

There is no significant difference between the ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER and the commercial cleaning tools in terms of the appearance of the floor after cleaning.

Significance of the Study

The findings of this study will be beneficial to the following:

For janitors/students

It helps janitors/students' clean floors easily, which is useful if there is an urgent job and their floors need cleanup, or the area they were about to clean is a restricted location. Students can clean floors without consuming too much time. Knowing that students nowadays are lazy.

For Youths and other People

It creates a more repairable cleaner with modular components, updatable firmware and an open-source board so repairing this product is cheap. People without personal protective equipment can clean isolation areas without actually going inside of it. People can clean large rooms faster and minimal labor. People can clean contaminated areas without getting into it. This device was currently designed for schools for now.

Objectives

The general objective of this research project is to create an alternative, open-source robotic cleaner.

Specifically,

- It used an Arduino and open-source parts instead of proprietary components to operate this product and make it easily repairable.
- To clean floors the same way as using a proprietary robot cleaner but using only open-source parts.

Scope and Delimitation

This research was all about an open-source floor polisher robot which cleaned the mess and dusty floor with very minimal manual labor. This product was used as standby for about a year and 2.3 hours when it is used with vacuum only (35-45 minutes with mop). This can only be used in office, schools, and houses with tiled cement. The duration of this product was extended by charging the product. The product estimated cost is unknown, but it is somewhere between 1000 to 2500. It uses descriptive statistical tool to determine how effective the product is.

The timeframe in conducting this study is as follows:

TABLE 1: Timeframe

PLACE OF TESTING: Baruyan, Calapan City, Oriental Mindoro, Philippines	Action
8/28/2022 – 9/2/2022	Construction of research paper
9/3/2022	Checking paper for grammar errors
9/4/2022 – 10/10/2022	Start of raising research fund
10/11/2022 – 10/13/2022	Start of gathering materials
10/14/2022 – 10/15/2022	Construction of methodology
10/16/2022	Verifying procedural steps
10/17/2022 – 10/19/2022	Making the product’s base
10/20/2022	Creation and flashing of MJLF a1.0 open-source firmware (https://github.com/477AFD/MJGF) into Arduino, finishing the main unit
10/21/2022 – 10/22/2022	Making the product’s mop module
10/23/2022	Wiring, making modifications to a battery-powered vacuum
10/24/2022 – 10/25/2022	Construction of data gathering procedure and research design
10/26/2022 – 10/27/2022	Placing the battery, beginning of testing
10/28/2022	Grading of two products based on a constructed rubric
10/29/2022	Creation of MJLF Beta and the product schematics and publishing these to GitHub

Definition of Terms

Adafruit – is a shield that receives instructions from the Arduino and moves or rotates the unit.

Adafruit Motor Shield v1.0 – used for the motors, can support up to 1.2A per motor.

Arduino – main microcontroller for the unit.

Chassis/Tupperware – used to protect and cover all circuitry in the unit.

L293D – it is an integrated circuit that supports up to 36 volts, 600mA per motor, and supports up to two motors. It is required for the shield to operate at 1.200A per motor.

Lead-acid Battery – It is a battery that uses acid chemical reaction to create electricity. It is used to power the motor shield, and therefore, geared and high-torque dynamos.

Power Bank – It is a USB rechargeable battery with a current of up to 3A. It is used solely to power the Arduino.

Relay – is an electronic switch that uses a low voltage electromagnet to turn the high-voltage switch on or off.

Servos – those are high-torque, low-speed and low-powered dynamo with data pins instead of normal positive and negative pins. Used in turning the whole unit and the directional sensor.

Ultrasonic sensor – used to measure the time lag between the sender and the receiver using sound waves greater than 100 kHz, and it is used to determine the distance of a wall or a control stick.

Variable-speed drill – also known as the hole puncher. It has a controllable speed analog trigger and direction buttons to operate the unit.

CHAPTER 2 – REVIEW OF RELATED LITERATURE AND STUDIES

RELATED LITERATURE

Arduino

According to Arduino (2018), Arduino is a free and open-source electronics platform with simple hardware and software. Arduino boards can read inputs such as a light on a sensor, a finger on a button, or a Twitter tweet and convert them into outputs such as operating a motor, turning on an LED, or posting anything online. You may direct your board by delivering a series of instructions to the board's microcontroller. You utilize the Arduino programming language (based on Wiring) and the Arduino Software (IDE) (based on Processing) to do this.

Ultrasonic Sensor

According to MaxBotix® (2023), an ultrasonic sensor is a device that uses ultrasonic sound waves to determine the distance between two objects. An ultrasonic sensor employs a transducer to emit and receive ultrasonic pulses that communicate information about the proximity of an item. High-frequency sound waves reverberate off boundaries, resulting in different echo patterns. Ultrasonic sensors operate by emitting a sound wave with a frequency higher than that of human hearing. The sensor's transducer functions as a microphone to receive and transmit ultrasonic sound. Like many others, our ultrasonic sensors employ a single transducer to emit a pulse and receive the echo. The sensor calculates the distance to a target by measuring the time between delivering and receiving the ultrasonic pulse.

Adafruit Motor Shield

Arduino is an excellent place to start learning about electronics, and with the addition of a motor shield, it can also be a neat clean platform for robotics and mechatronics. Here's a concept for a full-featured motor shield that can power a variety of easy to medium-complexity applications. (Adafruit, 2012)

Vacuum Cleaner

According to Reliance Digital (n.d.), vacuum cleaners are not only great in cleaning dust and allergens, but they are also simple to operate and save time and energy. They have suction motors and filters to collect dirt and dust. They are classified as hand-held, canister, vertical, or robot vacuum cleaners.

Mop Cleaner

A mop is a tool used to clean floors by soaking them and running the mop over them to remove dirt, dust, and other debris. (Union Service Co., Ltd. 2022)

RELATED STUDIES

An automated device called an automatic floor cleaner helps its user keep their space tidy and sanitary. To create autonomous cleaners, numerous industries are engaged in the automation sector. This essay discusses the creation of an automatic floor sweeper. The area of robotics is currently receiving a lot of attention as a way to reduce human labor. Our goal is to build a fully autonomous floor cleaner that can perform UV sterilization in addition to dry and wet cleaning. Currently, cleaners with one or two functions dominate the market. We are utilizing Arduino due to its ease and cost savings. The cleaner will be a step toward achieving a comfortable way of life by fixing the issues with conventional floor cleaning techniques. (Vyas, et al. 2020)

The study paper talks about the history of the automatic floor cleaner. Surfaces are automatically cleaned using this project in both domestic and professional settings. It travels all over the surface (a floor or other area) as it passes over it, sucking in the dust, when it is turned ON. The controller controls the motors and suction unit, and a few sensors are used to detect impediments. This could make it possible for people to live better lives. (Jain, Rawat and Morbale 2017)

To make modern living easier, we have created a low-cost, straightforward automatic floor polisher in this work. This little cleaner and polisher robot can vacuum and polish the floor automatically without running into any furniture or other obstructions. Using a Bluetooth phone, you may remotely turn on or off this robotic floor cleaner. The Bluetooth pairing process uses a straightforward, approachable user interface that is favorable to third-party programs. An Arduino microcontroller, an ultrasonic sensor, two PC fans, four DC motors, two discs, LEDs, and a Bluetooth module are all used in the construction of this project. This 3R-compliant, compact, and environmentally friendly polisher robot floor cleaner is composed of materials. (Goon, et al. 2019)

CHAPTER 3 - METHODOLOGY

Procedure

I- Preparation

Materials

The materials to be prepared are the following:

- 1 pc. Arduino Uno
- USB-A to USB-B connector
- Adafruit Motor Shield v1.0
- 1 pcs M2M/M2F/F2F wire bundle with 25 cm length
- Tupperware with 25 x 34 x 12 cm dimensions
- steel sheet with 24 x 11 cm dimensions
- steel sheet with 13.5 x 8 cm
- plywood with 13.5 x 8 cm
- one disc with 6.7 cm diameter
- 1 small spool of heavy-duty wire
- small 12-volt lead-acid battery
- 6-volt lead-acid battery
- 20000mAh (used) power bank
- pot holder
- net
- 1 pc 12-volt high-torque dynamo
- 1 pc 12-volt geared dynamo
- 4 pcs 6-volt geared dynamos

- 2 pcs servos
- 1 normally-open 5V relays 4 wheels
- wood glue
- 2 L293D motor driver chips
- dextrose hose
- 4 360-degree steel rollers
- 250 mL liquid soap
- round container with 13.7 cm diameter
- round container with 14 cm diameter
- round container with 8.5 cm diameter
- motor mounting cylinder with 14 cm diameter
- tie wire

Equipment used

The drill used in the study is the BOSCH Variable-Speed drill. It is used to create holes.

To control the unit while in semi-auto mode, a wooden stick with 1.37 meters of length with a pan on it are used.

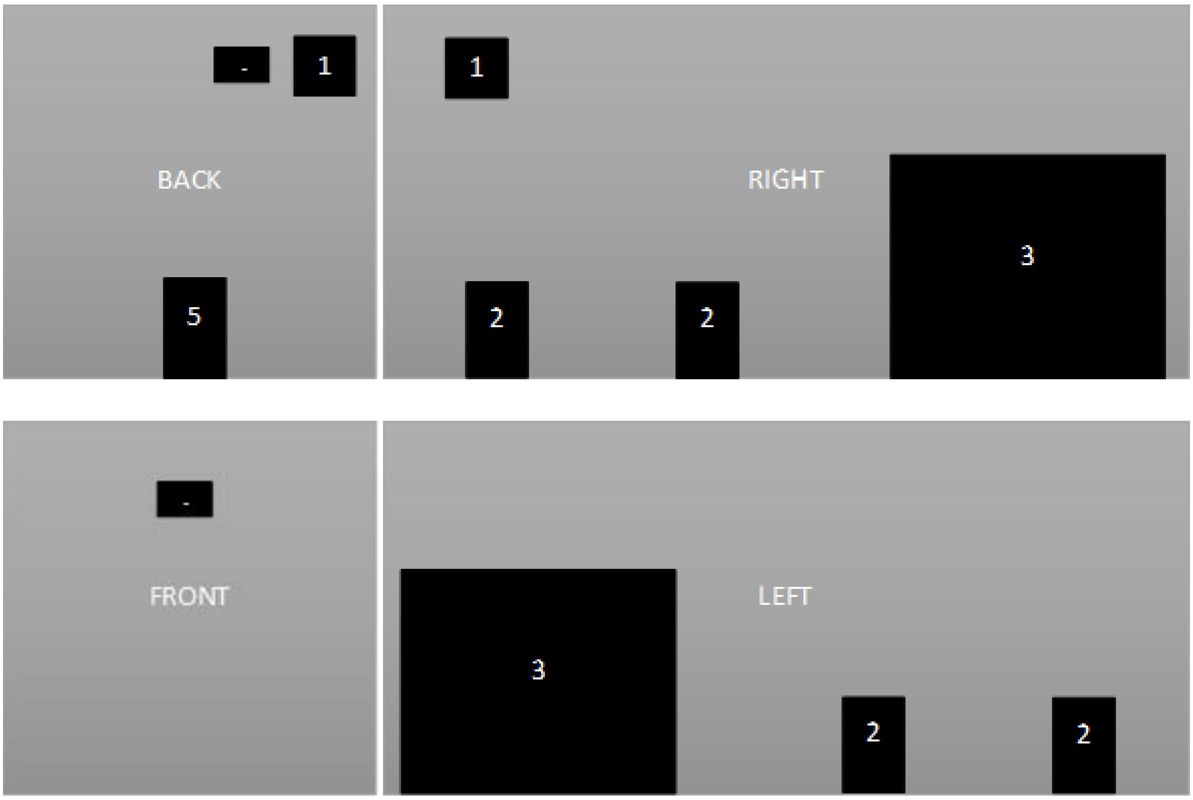
To solder the necessary wires, a soldering iron is used.

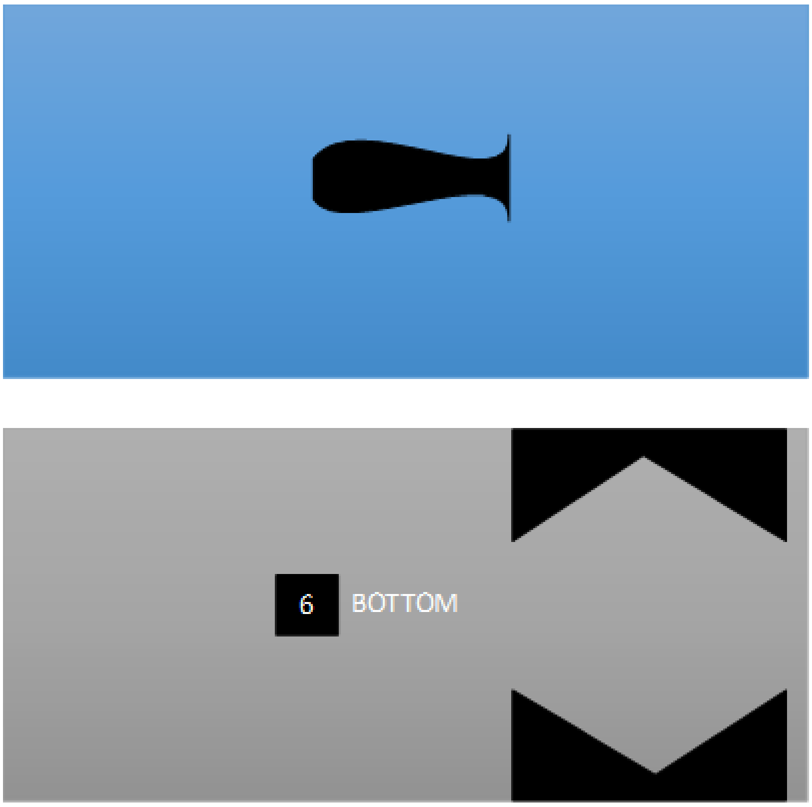
Other equipment used in this study is the scissors, which are used to cut wires; sandpaper, which are used to remove excess wood/glue and; screwdriver (with varying sizes), which are used to secure motors and assemble wires to the Adafruit.

II- Preparing the chassis

For the chassis, several holes with:

Switch (1) (2 x 1.70 cm), geared hobby motors (2), 6 x 5.4 cm for turning motor (3), (4) the same size as the wire thickness, (5) the same size as the mop motor's diameter, and (6) the same size as a dextrose hose thickness. For other holes, researchers followed the figure shown below.





TOP

FIGURE 1: Position and Size of Holes

The black parts of a figure represent areas to be discarded.

III- Preparing the Arduino and the Motor Shield

For the motor shield, pinouts were soldered to `A0-A5` holes, six `5V` and `GND` holes. Then, the `PWR` pins were desoldered and its pinouts was transferred to an additional `5V` and `GND` hole, right at the bottom of the desoldered `PWR` hole. Then, two L293D chips were placed on top of an existing chip at the exact orientation, the modified motor shield was placed on top of the Arduino, and a piece of code from `https://github.com/477AFD/MJGF` (or appendix B) was placed on a computer running Arduino IDE, the Arduino/Motor shield combo was connected to the computer, and was uploaded to it.

For the wiring, here is the pin configuration:

TABLE 2: Pin Configuration

PIN	HARDWARE	CONNECT TO
A0	Ultrasonic Sensor	`TRIG`
A1	Ultrasonic Sensor	`ECHO`
A2	Relay	`DATA`
A3	No Connection	
A4	No Connection	
A5	Switch	`GROUND`
SERV0	Servo	`Turning Gear`
SERV1	Servo	`Directional Sensor`
M1	Motor	`RB`
M2	Motor	`RT`
M3	Motor	`LB`
M4	Motor	`LT`

In this table, the `RB` (right-back), `RT` (right-front), `LB` (left-back) and `LT` (left-front) represent the left and right dynamos in the unit; `TRIG` stands for pulse trigger, which send an ultrasonic wave (USW) impulse; and `ECHO`, which receive an USW impulse.

IV- Assembling the Dynamos and the Main Board

For the switch, it looks like this: (The wires were soldered on the switch pins)

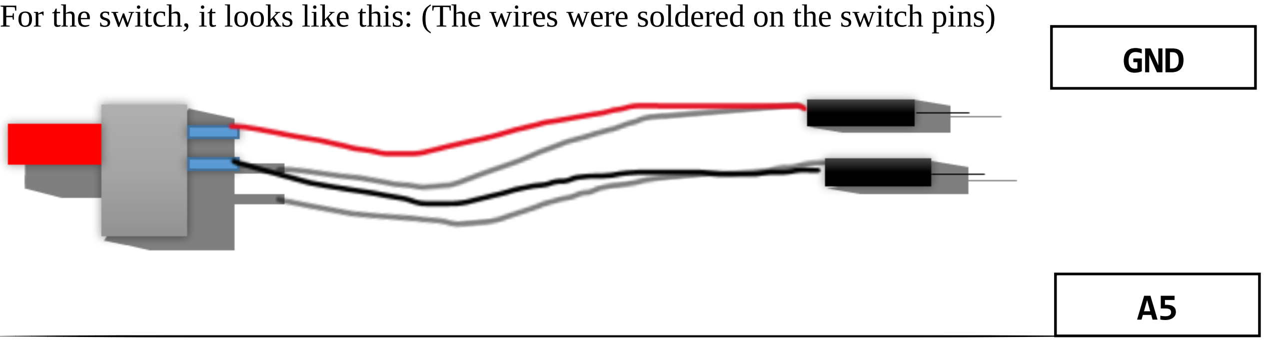
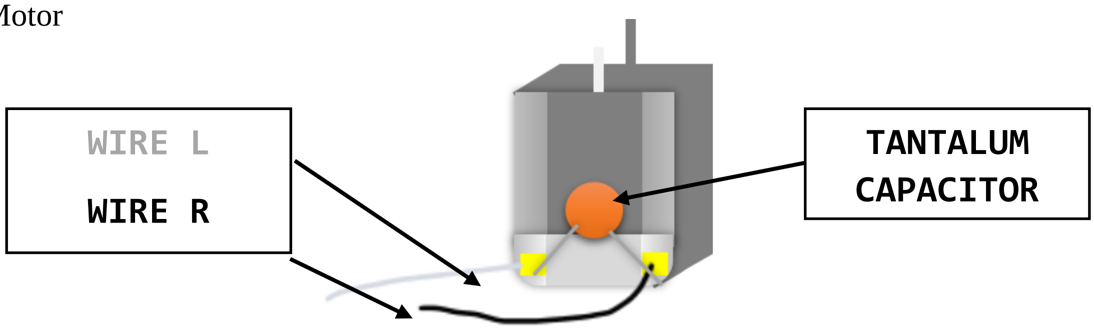


FIGURE 2: Switch and the Wire

Now, the dynamo was removed from a gearbox, and it was soldered with those:

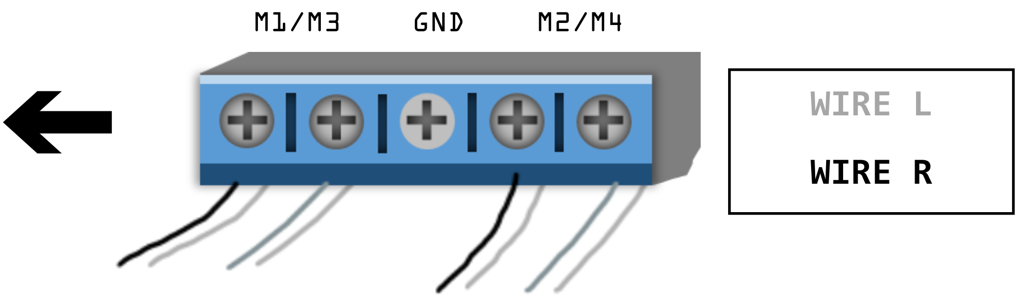
FIGURE 3: Motor



Then, the gearbox was placed in the holes and was secured with tie wire and glue, the dynamo was inserted into the gearbox, and the wire is placed like this:

FIGURE 4: Motor Shield

Sockets

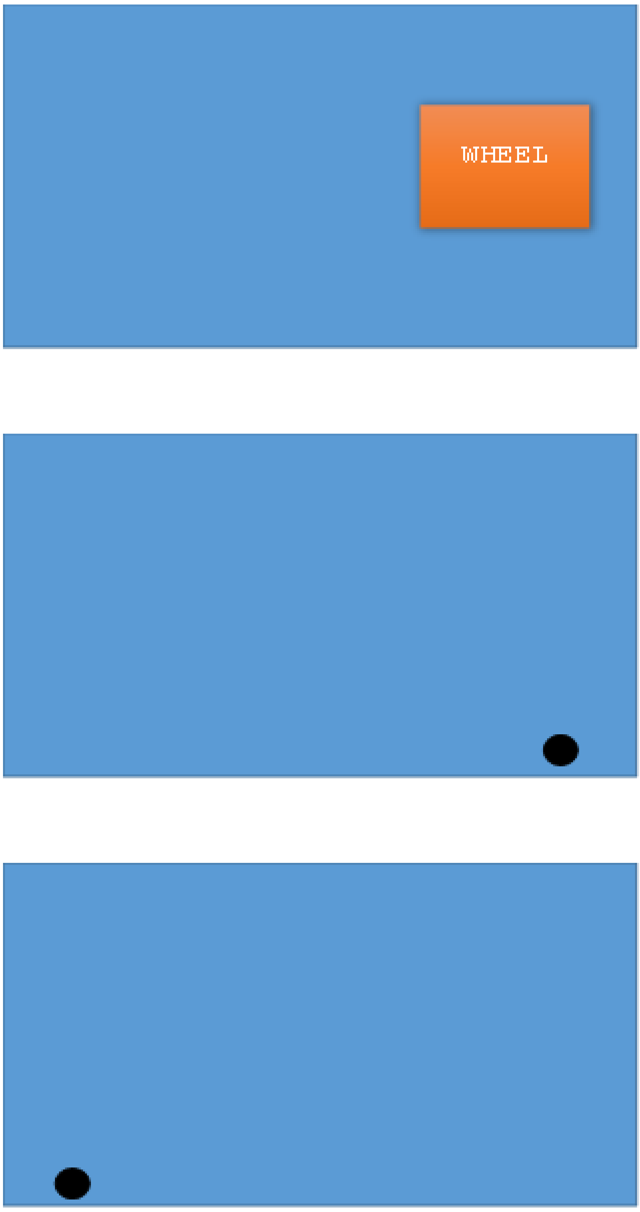


Then, the M+ socket on the back was connected to the socket (┐) on the relay, and the +6V power line was connected to the socket (⎞). For the GND socket on the motor shield, it is connected directly into the -6V/GND power line of the battery.

Now that it is connected, the VCC and the GND in the relay is plugged onto the 5V outlet and GND, respectively. The model should look like in appendix E.

V- Assembling the 12V hardware and vacuum

For the 12V battery trailer, a small box with 11 x 22 x 11 cm is prepared and researchers drilled and placed the following:



BOTTOM SIDE

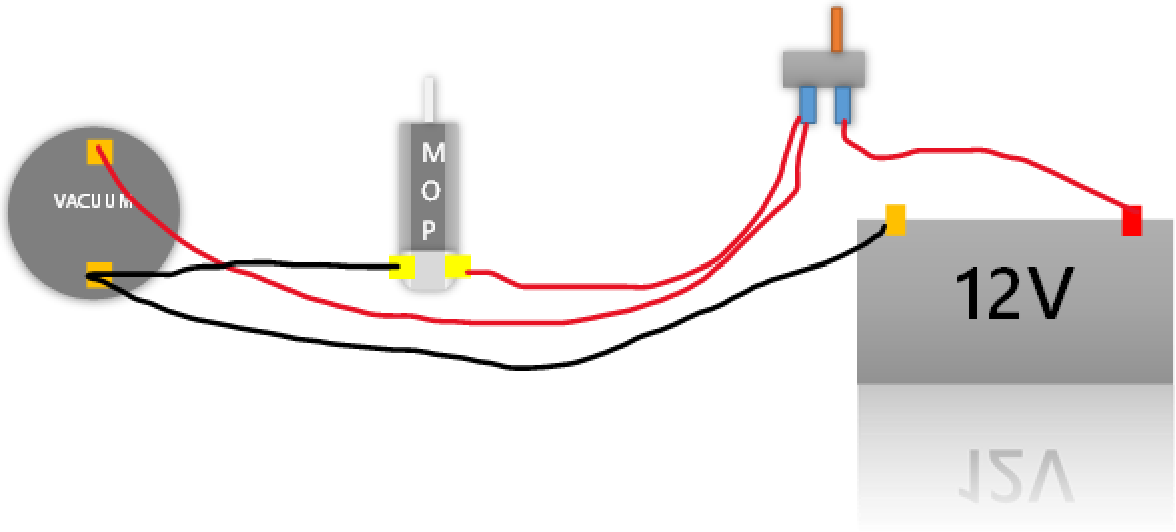
LEFT SIDE

RIGHT SIDE

FIGURE 5: Position of Holes and a Wheel

Now for the motors and a switch, here is a diagram:

FIGURE 6: Motor, Switch & Battery Diagram

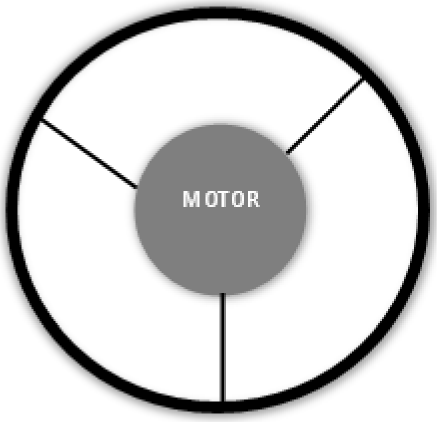


The mop motor is placed into the mop motor hole in the chassis and was secured with tie wire. And with the disc and mop, its center was drilled and plugged into the mop motor.

For the vacuum, a large round container with 6 cm diameter was prepared and its beak was cut and its base are drilled with holes. For the fan assembly, a computer fan was prepared by removing the fan blade from the fan assembly, removing the shaft from the fan blade, drilling a hole enough to fit the motor shaft, and securing the fan blade & motor shaft with glue. For the trash collector bin, a hole was drilled with 3 x 1 cm at its base, and a 360-degree steel ball roller was placed at the top of the hole.

For the fan assembly, researchers followed this format:

FIGURE 7: Vacuum Motor Layout



After that, the wires were soldered onto the pins. The pin with the black dot represents +12V. Then the wire was routed on one of the holes, to the front hole, to the switch (shown in figure 6), and the whole motor-fan assembly was placed on a large round container.

Next, the cap of the trash collector was removed and cut so that the screw part remains, and that was glued into the beak of the vacuum assembly. Then, with tire wire, tacker and epoxy glue, the vacuum assembly was placed into the plywood, then into the front of the unit.

Then the researchers checked the polarity of each wire so that the GND and +12V wires are not in contact with each other, and it was tested by flicking the red switch.

VI- Turning Unit

With this piece of steel below, researchers glued it into the chassis, drilled 2 holes and inserted a plastic tube to prevent friction.

Then, two steel alloys were prepared and researchers shaped the alloy with this:

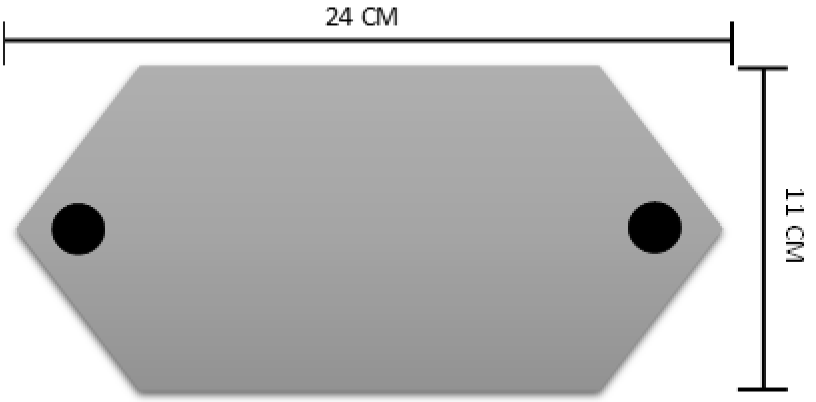


FIGURE 9: The shape of the bent alloy

X – height @ 3 cm

Y – length, looped

Z – width @ 4.2 cm

The wheels were drilled so that the Z tip fits into it. The wheel was placed there, and was secured with a large tube. The same was done for the other one, and was inserted into the unit.

After that, another steel alloy was prepared with 8 cm length and with pliers, the center was looped, its tips were bent and inserted into the loop of the other alloy with wheels.

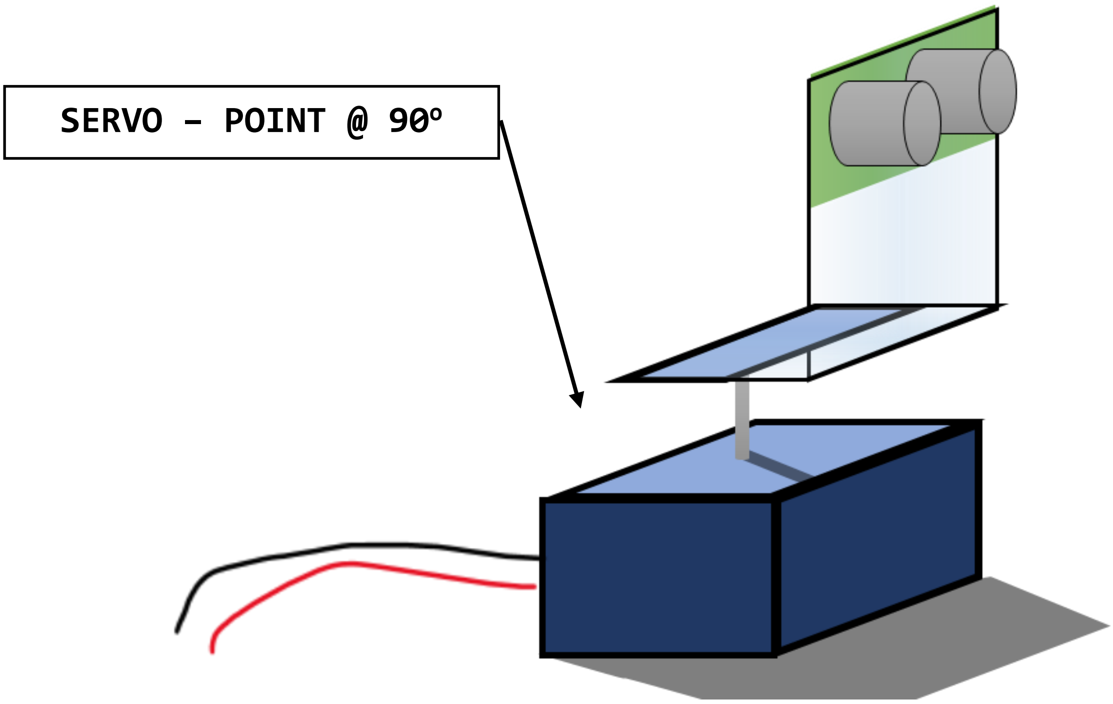
Then, a servo was prepared with one part of the wing and the shaft pointer pointing to 90^{o}. An alloy with 4.7 cm length was prepared where one end of it was glued and fitted with tie wire on both wings of a servo and the other end bent and fitted into a loop from the long alloy.

After assembly, researchers tested it so the wheels will tilt sideways.

VII- Directional Sensor

With the servo with wings pointing sideways and the pointer in 90°, the ultrasonic sensor (appendix F), the sensor holder, and foamy tape, it was assembled like this:

FIGURE 10: Directional Sensor Assembly

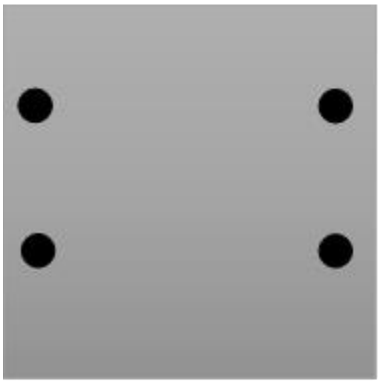


VIII- Final Assembly, Solvent Assembly

First, referring the pin configuration (TABLE 2), researchers plugged in the appropriate sockets (except directional sensor, as the vacuum must be assembled first). Then, with a small container, researchers drilled a hole enough to fit in the dextrose plug, the plug was inserted, put water sealants and glue, and was placed into the unit's cover. After that, it was tested for water-tightness/leaking test.

Then, researchers drilled four holes in the front of the unit, a steel sheet with the same dimensions as he vacuum-mounting plywood and the vacuum mounting plywood itself, like this:

FIGURE 11



And with four small bolts and nuts, the vacuum unit was secured into the unit. After that, with correct polarity and following the diagram in figure 6, the vacuum wires were installed into the 12V wires, the directional sensor assembly was installed on top of the vacuum facing the front with glue, and referring to table 1, the wires were installed onto the correct pins.

Lastly, the cover was installed with the dextrose hose being inserted into the hole on the bottom of the unit, and the 12V wires were plugged into the battery.

Data Gathering Procedure

I. Testing the Unit

To test the unit, the USB is connected to the power bank. The system will do diagnostics and testing before successful booting. After that, the white (software) switch was flicked.

II. Testing the turner unit

Researchers find out if it can turn properly without issues, such as wrong angle. If it is, researchers changed the timing in the code used for turning, and then it is re-tested.

III. Changing the distance of Reaction

The unit moves slowly and only work on flat floors, as any bumps or puddles stops the unit from moving. For the distance of reaction, researchers chose 6 centimeters.

IV. Recording the Scores

Researchers prepared the two units and the area to be tested is a room with 0.77 x 0.77 meters. There should be 8 trials, each trial 10 seconds. After each trial, researchers inspect the floor and was recorded based on a rubric below:

TABLE 3: Rubrics in Grading the Appearance of the Floor after Cleaning

CRITERIA	SCORE
Vacuum	
Amount of dust collected (30%)	
Amount of dust left (20%)	
Mop	
Amount of grime removed (30%)	
Least amount of solvent used (20%)	
Total (100%)	

V. Comparing the Unit with cleaning tools

Researchers then placed all the recorded scores in TABLE 4 and compared the unit against commercial cleaning tools based on appearance of the floor after cleaning.

TABLE 4: Data Table

Respondent #	ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER	COMMERCIAL CLEANING TOOLS
1		
2		
3		
4		
5		
6		
7		
8		
AVERAGE		

VI. Analyzing the Data

Then, to test the significance of the product against the commercial one, the statistical tool used is T-test.

Research Design

The research design was experimental and comparative as the grading of the appearance of the floor after cleaning is being tested. The statistical tool to be used is T-test.

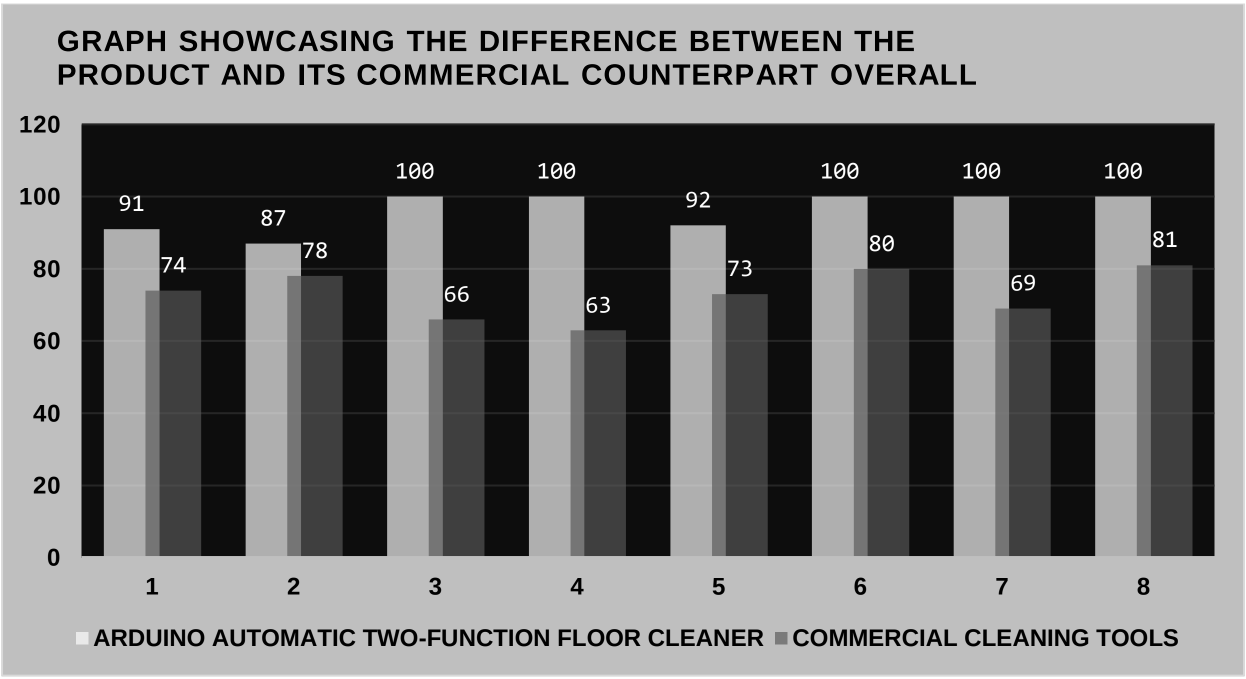
CHAPTER 4 – RESULTS & DISCUSSION

The researchers tested the mop and vacuum parts of the device, recorded the use of manual cleaning tools and grabbed its result based on the data table (table 4). The solvent used in the experiment is ethyl alcohol. The data gathered from testing was recorded into the table below:

TABLE 5: Data Table containing the Percentage of Effectiveness of two Products

Respondent #	ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER	COMMERCIAL CLEANING TOOLS
1	91	74
2	87	78
3	100	66
4	100	63
5	92	73
6	100	80
7	100	69
8	100	81
AVERAGE	96.25	73

The data table shows the score for the appearance of the floor after cleaning. Results shown that the overall appearance after using commercial cleaning tools are worse when compared to ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER based on its average. Here is the graph:



The bar graph showcases the differences in scores of the device and its commercial one. The graph shows that the floor cleaner is more effective in cleaning than its commercial counterpart. To test the significance, the statistical tool used is T-test.

TABLE 5: T-Test Table

t-Test: Two-Sample Assuming Unequal Variances		
	Variable 1	*Variable 2*
Mean	96.25	73
Variance	28.78571429	43.42857143
Observations	8	8
Hypothesized Mean Difference	0	
Degrees of Freedom	13	
t Stat	7.738492941	
P(T<=t) one-tail	1.60571×10^{-06}	
t Critical one-tail	1.770933396	
P(T<=t) two-tail	3.21142×10^{-06}	
t Critical two-tail	2.160368656	

The T-test calculation shows that there is a significance between the two products, so researchers rejected the null hypothesis. The alternative hypothesis is accepted, there is a significant difference between the ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER and the commercial cleaning tools in terms of the appearance of the floor after cleaning.

CHAPTER 5 – CONCLUSION

Based on the results, the following conclusions are drawn:

1. The robotic cleaner's mop and vacuum unit clean floors more effectively than the commercial tools provided that the solvent used to aid in cleaning is alcohol.
2. Based on the recorded data table, there is a significant difference between the device and its commercial counterpart.

Researchers reject the null hypothesis if t Stat is more than t Critical Value (t>T). The results from the calculated T-test shows that the t value is higher than its critical value (7.74 > 2.16), indicating that there is a significant difference and therefore, the null hypothesis is rejected. The alternative hypothesis is accepted, "there is a significant difference between the ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER and the commercial cleaning tools in terms of the appearance of the floor after cleaning."

After testing and comparing the difference between the ARDUINO AUTOMATIC TWO-FUNCTION FLOOR CLEANER to the commercial cleaning tools based on the appearance of the floor after cleaning, a short interval was noted. It is recommended to:

1. Use this research study along with other papers dealing with open-source robots and cleaning techniques to make a better, more environmental-friendly open-source robotic cleaner.
2. Make use of the open-source automatic cleaner in schools, hospital and large houses.
3. Make the product more functional and effective by:

Using 12V metal-geared dynamos instead of a hobby 6V dynamo for better speed and reduce weight as only one battery will be used; add suspensions to mop and wheels for better traction against bumpy surfaces; use a 3D-printed vacuum fan instead of metal fan to prevent loudness and injury; use a better, more current-tolerated Adafruit Motor Shield 2.0 to support the 12V dynamos; use a smaller case for the main unit; and use a rotatable, belted set of wheels instead of wheel-and-turner to completely alleviate friction.

REFERENCES

ada, lady. 2012. "Adafruit Motor Shield." *Adafruit.* August 27. Accessed May 15, 2023. https://learn.adafruit.com/adafruit-motor-shield/overview.

Arduino. 2018. *What is Arduino?* February 5. Accessed May 15, 2023. https://www.arduino.cc/en/Guide/Introduction.

B_E_N. n.d. *SparkFun.* Accessed November 27, 2022. https://learn.sparkfun.com/tutorials/what-is-an-arduino/all.

Colon, Alex, and Angela Moscaritolo. 2023. "The Best Robot Vacuums for 2023." *PCMag.* July 5. Accessed July 9, 2023. https://www.pcmag.com/picks/the-best-robot-vacuums.

Goon, Li Hung, Ahmad Nur Iman Md Isa, Chia How Choong, and W.A.F.W. Othman. 2019. "Development of Simple Automatic Floor Polisher Robot using Arduino." *School of Electrical & Electronic Engineering.* January 10. Accessed November 17, 2022. https://www.academia.edu/download/61073359/23-Article_Text-20-1-10-2019102520191030-84654-1na8t74.pdf.

Jain, Manya, Pankaj Singh Rawat, and Jyoti Morbale. 2017. "Automatic Floor Cleaner." *International Research Journal of Engineering and Technology (IRJET).* April. Accessed November 17, 2022. https://journal.umy.ac.id/index.php/jrc/article/view/10007.

Masigan, Andrew J. 2019. "A solution to Metro Manila's garbage problem." *BusinessWorld.* June 30. Accessed July 7, 2023. https://www.bworldonline.com/editors-picks/2019/06/30/239416/a-solution-to-metro-manilas-garbage-problem/.

MaxBotix. 2023. *How Ultrasonic Sensors Work.* March 1. Accessed May 15, 2023. https://maxbotix.com/blogs/blog/how-ultrasonic-sensors-work.

Reliance Digital. n.d. "Vacuum Cleaners – Here's Why Every Household Should Have One." *Reliance Digital.* Accessed May 15, 2023. https://www.reliancedigital.in/solutionbox/vacuum-cleaners-heres-why-every-household-should-have-one/.

Sarwar, Ismail. 2016. "Advantages and Disadvantages of Using Arduino." *Engineer Experiences.* October 1. Accessed November 27, 2022. https://engineerexperiences.com/advantages-and-disadvatages.html.

Team, The Makerkids. 2019. "What is Arduino & Why is It Important in Robotics?" *MakerKids.* November 12. Accessed November 27, 2022. https://makerkids.com/what-is-arduino-why-is-it-important-in-robotics/.

Union Service Co., Ltd. 2022. "What are the benefits of using a cleaning mop." *KnowBody.* March 1. Accessed May 15, 2023. https://knowbodycleaning.com/article/what-are-the-benefits-of-using-a-cleaning-mop.

Vyas, Akanksha, Satyam Chourasia, Shubham Antapurkar, and Raghvendra Prasad. 2020. "ARDUINO BASED DRY & WET AUTOMATIC FLOOR CLEANER." *International Journal of Engineering Technologies and Management Research.* April 25. Accessed November 21, 2022. https://www.researchgate.net/publication/341243272_ARDUINO_BASED_DRY_WET_AUTOMATIC_FLOOR_CLEANER.

APPENDIX

A – Script used in presenting methodology

[appendix A]

SPECIAL OUTLINE for METHODOLOGY ([space] per item)

- (Display the materials and equipment. Do not read them.) Here are the materials (2 sec.) [space] and here are the equipment's that are used in making this device.
- These are the holes drilled [space] during the chassis preparation. [space] [space]
- A0-A5 holes, 5V and GND holes are soldered with pinouts, and transferred the PWR pinouts to 5V-GND connector with a soldering iron.
- Here is the pin config for A0-A5 pins.
- And here is for the servo and motor pins. With the Arduino IDE, the software is flashed into the board.
- After that, the shield is plugged on top of an Arduino.
- Now, the wires were soldered into the switch pins, [space] the dynamo's pin headers were soldered with black and white wires and a tantalum capacitor, and [space] the black wire is on the left in inserting a wire. Then the Arduino was placed in the chassis.
- Here is what a wire configuration looks like with the servos, the battery and relay. The TRIG and ECHO wires were plugged in to the A0 and A1 pins, respectively.
- For the trailer, the wheel was placed at the bottom, a hole was drilled on left and right side, and a battery was placed inside. [space] Here is the schematic that was used in assembling the mop and vacuum.
- Here is a large round container that is upside-down.
- Its beak was cut and holes were drilled so that the wire fits and for ventilation.
- Here is the dynamo that was fitted into the mounting disc and a fan.
- This fan assembly was placed into the container, and its wire was inserted out of one of the holes.
- Here is a small container that are used as a trash collector bin.
- Researchers took the cap, [space] removed the top part in favor of the net, and was glued inside the fan & motor assembly.
- The trash collector was placed at the inserted cover, melted a hole with 3 x 1 cm and placed a 360-degree steel ball roller beside the longer portion. It is then placed into a plywood, and was secured with screws.
- Next, for the turner, this plate has a size of 24 x 11 cm. Researchers drilled holes on the tips.
- Here is a shaft used with a wheel and an alloy to rotate sideways.
- With the servo and an alloy attached and at 90 degrees, it was placed in the main alloy, like in the picture above.
- For the directional sensor, researchers assembled the servo, the sensor and its holder and tie wire. Here is what it looks like.
- For the final assembly, researchers prepared a small container, drilled a hole at the bottom, one end of a dextrose hose was inserted, and the other end to the bottom hole. To assemble the vacuum, a steel was prepared with 13.5 x 8 cm and drilled holes, like in this picture.
- After that, the wires were assembled and inspected, covering the unit, and here is what a final product looks like.
- (Read what it is shown on-screen except for the table)
- Here is the data table.
- The research design was experimental and comparative as the grading of the appearance of the floor after cleaning is being tested. The statistical tool to be used is T-test.

B – Code used in operating the unit

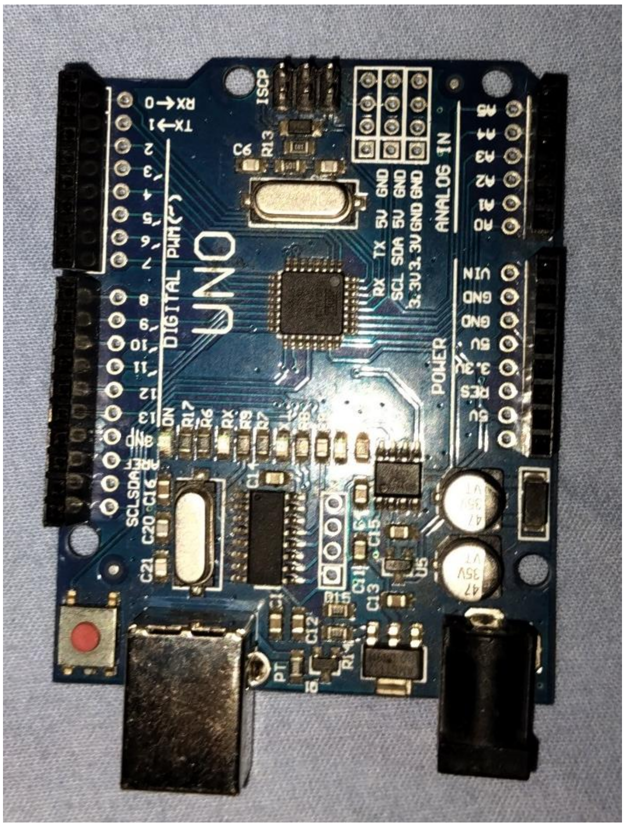
```
#include <AFMotor.h> // AF_Motor library
#include <NewPing.h> // NewPing library
#include <Servo.h> // Servo library

#define TRIG_PIN A0
#define ECHO_PIN A1
#define MAX_DISTANCE 200
#define MAX_SPEED 255 // sets speed of DC motors
#define MAX_SPEED_OFFSET 20

NewPing sonar(TRIG_PIN, ECHO_PIN, MAX_DISTANCE);

AF_DCMotor motor1(1, MOTOR12_1KHZ);
AF_DCMotor motor2(2, MOTOR12_1KHZ);
AF_DCMotor motor3(3, MOTOR34_1KHZ);
AF_DCMotor motor4(4, MOTOR34_1KHZ);
Servo myservo;
Servo turnservo;
bool goesForward=false;
int distance = 100;
int speedSet = 0;
bool power = false;
bool powerRaw = false;

int readPing() {
  delay(70);
  int cm = sonar.ping_cm();
  if(cm==0)
  {
    cm = 250;
  }
  return cm;
}

void setup() {
  pinMode(A2, OUTPUT);
  pinMode(A3, OUTPUT);
  pinMode(A5, INPUT_PULLUP);
digitalWrite(A2, HIGH);
digitalWrite(A3, HIGH);
myservo.attach(10);
turnservo.attach(9);
  myservo.write(15);
  delay(1000);
  myservo.write(165);
  delay(1000);
myservo.write(90);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
turnservo.write(90);
//digitalWrite(A2, LOW);
delay(3000);
}
```

```
int isTouched = 0;
void loop() {
 int distanceR = 0;
 int distanceL =  0;
 delay(40);
 if (digitalRead(A5) == LOW) {
  digitalWrite(A2, LOW);
  digitalWrite(A3, LOW);
 if(distance<=15)
 {
  moveStop();
  delay(105);
  distanceR = lookRight();
  delay(200);
  distanceL = lookLeft();
  delay(200); /*
  moveBackward();
  delay(900);
  moveStop();
  delay(205); */

  if(distanceR<=distanceL)
  {
    turnRight();
    moveStop();
  }else
  {
    turnLeft();
    moveStop();
  }
 }else
 {
  moveForward();
 }
 distance = readPing();
 } else {
   digitalWrite(A2, HIGH);
   digitalWrite(A3, HIGH);
 }
}

int lookRight()
{
    myservo.write(15);
    delay(500);
    int distance = readPing();
    delay(100);
    myservo.write(90);
    return distance;
}

int lookLeft()
{
    myservo.write(165);
    delay(500);
    int distance = readPing();
    delay(100);
```

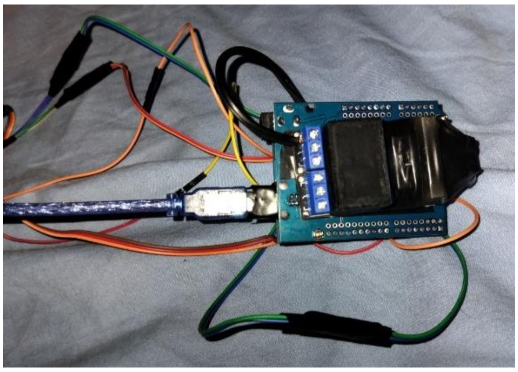
```
    myservo.write(90);
    return distance;
    delay(100);
}



void moveStop() {
  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);
  }

void moveForward() {

 if(!goesForward)
  {
    goesForward=true;
    motor1.run(FORWARD);
    motor2.run(FORWARD);
    motor3.run(FORWARD);
    motor4.run(FORWARD);
   for (speedSet = 0; speedSet < MAX_SPEED; speedSet +=2) //
slowly bring the speed up to avoid loading down the batteries
too quickly
   {
    motor1.setSpeed(speedSet);
    motor2.setSpeed(speedSet);
    motor3.setSpeed(speedSet);
    motor4.setSpeed(speedSet);
    delay(5);
   }
  }
}

void moveBackward() {
    goesForward=false;
    motor1.run(BACKWARD);
    motor2.run(BACKWARD);
    motor3.run(BACKWARD);
    motor4.run(BACKWARD);
  for (speedSet = 0; speedSet < MAX_SPEED; speedSet +=2) //
slowly bring the speed up to avoid loading down the batteries
too quickly
  {
    motor1.setSpeed(speedSet);
    motor2.setSpeed(speedSet);
    motor3.setSpeed(speedSet);
    motor4.setSpeed(speedSet);
    delay(5);
  }
}

int turnTime = 931;

void turnRight() {
  motor1.run(BACKWARD);
  motor2.run(BACKWARD);
  motor3.run(BACKWARD);
  motor4.run(BACKWARD);
  delay(turnTime);
  moveStop();
  turnservo.write(70);
  delay(500);
  motor1.run(FORWARD);
  motor2.run(FORWARD);
  motor3.run(FORWARD);
  motor4.run(FORWARD);
  delay(turnTime);
  turnservo.write(90);
  moveStop();
}

void turnLeft() {
  motor1.run(BACKWARD);
  motor2.run(BACKWARD);
  motor3.run(BACKWARD);
  motor4.run(BACKWARD);
  delay(turnTime);
  moveStop();
  turnservo.write(130);
  delay(500);
  motor1.run(FORWARD);
  motor2.run(FORWARD);
  motor3.run(FORWARD);
  motor4.run(FORWARD);
  delay(turnTime);
  turnservo.write(90);
  moveStop();
}
```

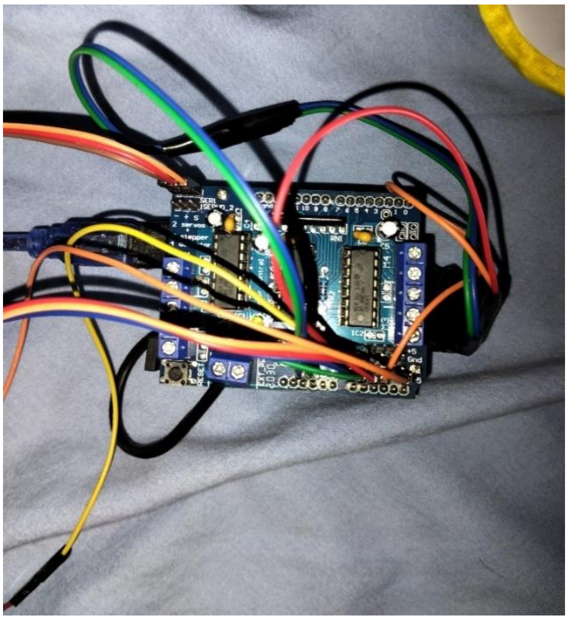
C – Product Structure (without vacuum)

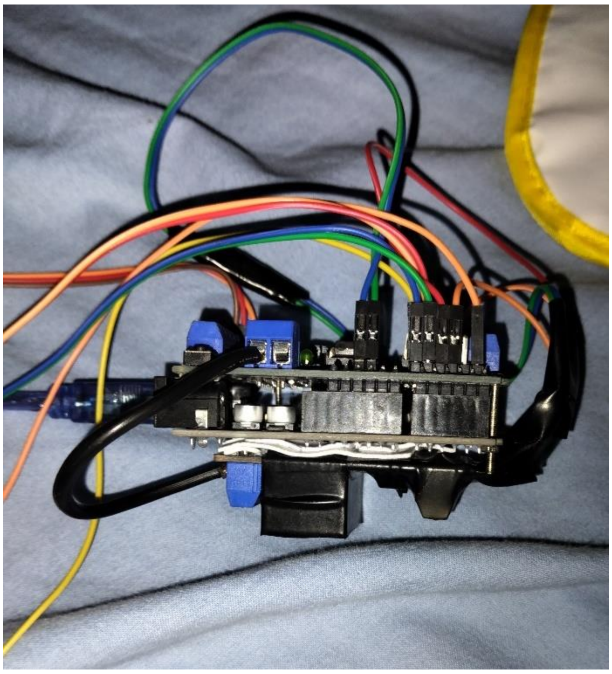
D – Arduino Uno used in making the device



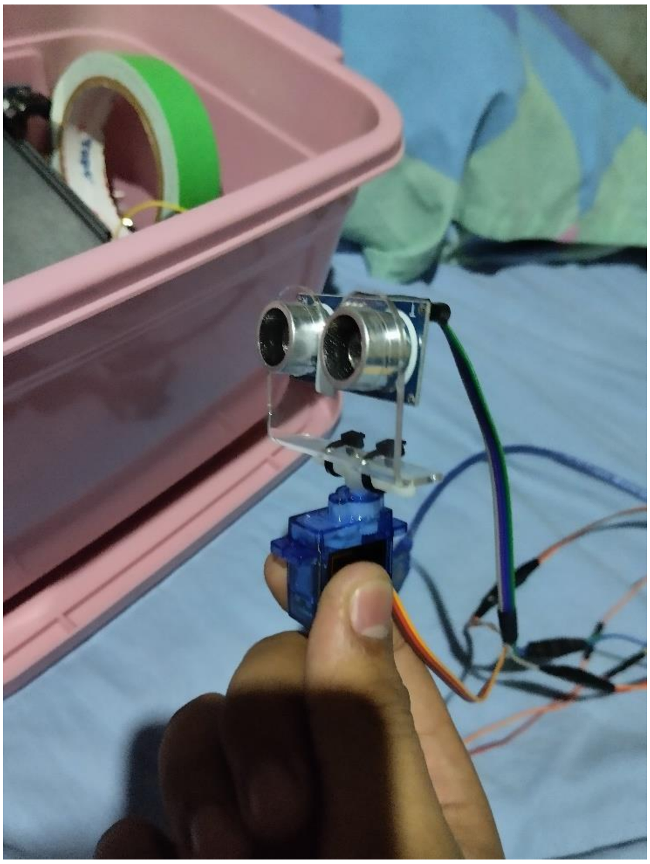
E – Arduino Uno with the relay and shield attached



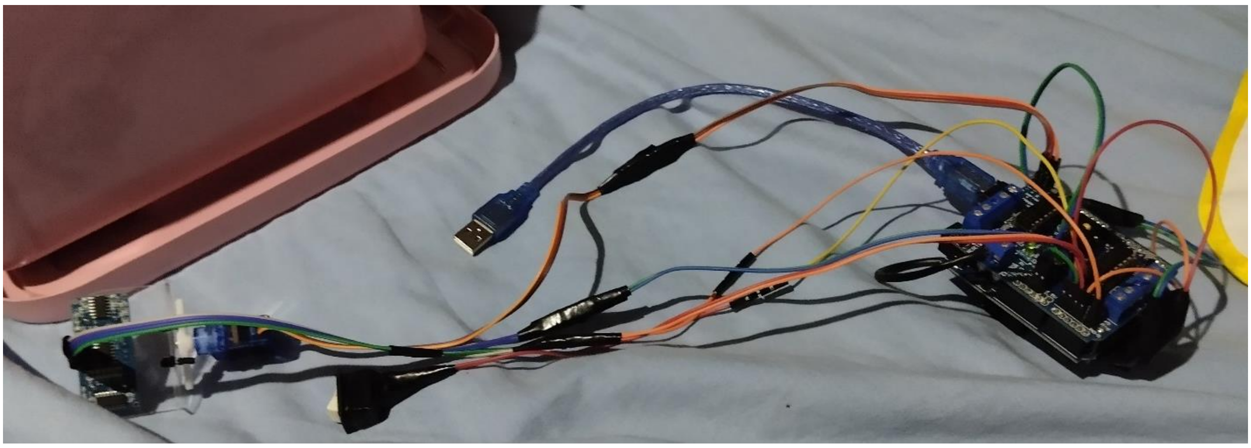




F – Ultrasonic Sensor



G – Arduino and Shield with Sensor, Switch and USB attached



H – Final Product