KUWAIT COLLEGE OF SCIENCE & TECHNOLOGY
كليــة الكــويت للعلـــــوم والتكنولــوجيا
Private University جامعــــة خاصــة

An autonomous Window cleaning robot

(Prototype)

Clients: small households, people who do not have the capability to clean.

Team Number: group 3

Team Members/Roles:

Mohammed Nasser Bekhait, 181181, Ce
Shaikha Bassam AlObaidi, 181175, Ce
Jenan Waheed AlKhabaz, 182014, Ce
Shahad Fahad Allaho, 181110, Ce

Team Website:

Advisers: Dr. Ismaeel Al-Ridhawi
Revised: 20-12-2022

Table of Contents

1.1 List of Figures

1.2 List of tables

1 ABSTRACT

With the advancements in automation technology nowadays, the involvement of Artificial Intelligence (AI) and Machine Learning (ML) in everyday tasks is becoming a reality. The use of such advanced technologies leads to task fulfillment cost and time efficiency. As such, the use of robotic devices for automated task fulfillment is becoming a must nowadays. This project focuses on the design and implementation of an autonomous window cleaning robot. Such a robotic device will replace the way windows are traditionally cleaned with the aid of humans and make it more time and cost efficient to use the proposed robotic device, especially in high-rise building scenarios. The proposed robotic device uses ML algorithms that provide path planning mechanisms prior to initiation and while running. Obstacle avoidance methods are also incorporated to ensure that path routes are planned to avoid any known obstacles. Obstacles that are found in real-time would initiate the path planning algorithm to run iteratively to find more optimal paths in terms of time-efficiency. A plethora of sensors are used to support the robotic device in path detection and movement.

2 INTRODUCTION

2.1 ACKNOWLEDGEMENT

we would like to convey our gratitude to everyone who made it possible for us to finish this report. Dr. Ismaeel AlRidhawi, our final year project advisor, whose stimulating suggestions and encouragement helped us to coordinate our project, particularly in writing this report. We also want to thank Dr. Ahmed AL Saleh for assisting us with the report and providing us with additional information about the robot.

2.2 PROBLEM AND PROJECT STATEMENT

Other companies rely on the employee cleaning surfaces at high altitudes and other companies depend on installing the cleaning device through the wiring, but what about small households and small companies, our project was made with the focus to help those who don't have the huge budget that the big companies have to achieve the same outcome.

2.3 PREVIOUS WORK AND LITERATURE

Starting with the most similar research was done by Young-Ho, Jae-Youl, Jong-Deuk, & Ka-Eun in the year 2012 talks about if it's possible for small robots to clean windows [1], and the goal off the experiment at the time was to create the World's First Commercialized Window Cleaning Robot for Domestic Use, and the way the they wanted to archive their goal is by writing simple programing and relaying on math to clean windows.

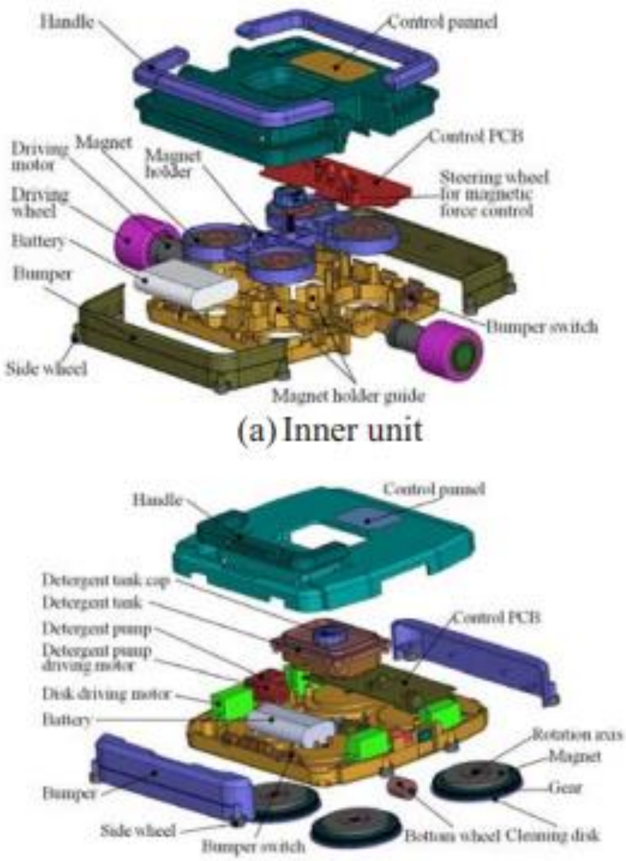

Figure 1

This figure shows the original design that the researcher's team in used in 2011 that they later have modified

2.4 OPERATIONAL ENVIRONMENT

For safety of our customers and our end users, our products was made with the safety of our customers in mind, so when designing the robot we did not modify any of the safety features that came with the robot it self like the safety bin witch if not engaged and help in please the robot would not work or the remote delay witch insure that the robot has the perfect time do a rotation to avoid falling from a window or the sound that make sure the user know when it perfect to remove the robot from any window.

2.5 INTENDED USERS AND USES

First, our users would be mostly households so we made a robot that would not be interfering with their home security system, and there will also be no cameras instead of that we will be using "IRF" sensors and algorithms so our robot can be smart, fast, easy to use with the touch off a button like any of the "IOT" devices that are currently in your house.

The use of our robot is to clean giant or small windows that are in households and are hard to reach/clean, but unlike other robots from other companies, who are unable to clean windows as efficiently as our robot, because we are using "ML" and algorithms, since that is our advantage, so we can compute, scan and map any large or small area to clean without leaving behind any stains or dirt but with a cheaper price and better deal.

2.6 ASSUMPTIONS AND LIMITATIONS

ASSUMPTIONS:

- **The robot work with any window even if does not have a basil as long as the robot did the scan beforehand.**
- **The robot the robot can work as long as the robot is connected to the wall outlet.**
- **The robot comes with different matts that can be replaced later.**
- **If the robot did not finish the scan or was interrupted during it the robot will stop in its place and make a sound to make sure that the user knows what is happing.**

LIMITATIONS:

- **The robot will only clean single glass panel at a time so it will not be able to jump from a window to a window in the house without a human interference.**
- **The other limitation that our battery version will only work for 15-25 min because we are using small batteries so if something happens to our product (if it's dropped) the batteries would not be a safety concern.**

2.7 EXPECTED END PRODUCT AND DELIVERABLES

- **Our product at this point is almost ready to be delivered to some house holds as a testing sample that we can our feedback from and since the original robot comes with allot of features from the factory so our production line can keep up with the costumes.**
  - **Features/components:**
    - **Matts: the robot comes with 4 types of matts two of them are from dry cleaning (cleaning the window just to remove the dust),and the other type is to clean the window when its wet.**
    - **Wall plug: it comes standard with a wall plug since the robot can only work when its connected to a wall outlet, and also because using batteries will add the to the total weight of the robot therefore the robot would not be as fast.**
    - **Remote control: since the remote control is the only part that we have modified our robot now uses a remote deck instead of a remote-control witch has and hold all the components that we are using to apply the algorithms.**
    - **Safety pin: it's a standard feature from the factory that comes with each robot to insure that the robot is not powered if for some reason it gets dropped of the wall.**

3 SPECIFICATION AND ANALYSIS

3.1 OBJECTIVE OF THE TASK

Our objective is to improve the quality of life inside households by integrating algorithm and machine learning by using robots which help with the day-to-day tasks like window cleaning.

Our approach included purchasing a robot with a low price that does not have the tech and the algorithm to be efficient, so we improved the robot by modifying the remote control that came with a robot to add and introduce new command that can help the robot to be more efficient and to add algorithms which help the robot to be a smart and efficient one.

3.2 FUNCTIONAL REQUIREMENTS

Non-Functional Requirements:

1. Security:

The robot and the app that will be used to control the robot will not need any access to your camera roll, contact, or any sensitive information on your device. 2. Usability:

The robot should include a remote control that can be easily navigated and the robot will be already fitted with 2-3 moods to clean windows.

2. Availability:

The app will be available 24/7 to use Furthermore any Explanations and requests would be responded by support team.

3.Storage:

• There will be a memory for saving the map services area that the robot will scan it.

4.Wi-Fi:

• Wireless communication if you are using Wi-Fi, you should indicate that you will be using the IEEE 802.11.x standard.

3.3 CONSTRAINTS CONSIDERATIONS AND ENGINEERING STARNDARDS

Constraints:

1. Battery:

• The previous report we stated that we will be using a battery but the better alternative was to use a directly connecting the robot to the wall to keep the safety features that the robot already has.

2. Safety:

• we haven't modified the robot itself instead we choose to modify the remote control to keep the safety of the robot and to avoid the security risks that we may face if we did.

3. Weight:

• the robot weight is constant with all the safety mangers and scenarios calculated and put in

mind so to avoid extra research and to save valuable time we didn't modify or add any weight to the robot itself.

4. Cost:

• The cost should be affordable for all people to cover their needs, so the approach that we took is purchasing the cheapest robot on the market so after all the modification the robot will still be lower than others.

5. Time

• In the past we had a concern with the battery life so instead we changed our method to increase the time that we can use the robot for by switching from a battery to plugging the robot straight from the wall outlet.

Engineering standards:

In our scenario we don't have any ethical problems or concerns but the only concern that we may face is the safety of our user's mobile phone so we will try to keep our application up to the safety requirement that all the other mobile phone apps have.

3.4 PREVIOUS WORK AND LITERATURE

Starting with the most similar research was done by Young-Ho, Jae-Youl, Jong-Deuk, & Ka-Eun in the year 2012 talks about if it's possible for small robots to clean windows [1], and the goal off the experiment at the time was to create the World's First Commercialized Window Cleaning Robot for Domestic Use, and the way the they wanted to archive their goal is by writing simple programing and relaying on math to clean windows. (pic)

Contact sensor-based coverage of rectilinear environments: An algorithm known as CCR is proposed to decompose an environment into rectilinear regions. Its goal is to provide a coverage of the obstacles faced by robots in this region. Most grid-based methods are complete when they are computed according to the approximate representation of the environment. This is because the grid map can be represented as an array with the cell's associated values. On the other hand, grid maps are very simple to use since they can easily mark regions in a grid map. Due to their resolution, grid maps are commonly used for coverage algorithms. Due to the small size of the area covered by grid-based coverage

methods, they are commonly used for indoor robot operations. Grid-based Coverage using the Wavefront Algorithm: presented the first grid- based method for coverage path planning. In their offline method, they use a grid representation and apply a complete coverage path planning algorithm to the grid. The method requires a start cell and a goal cell. A distance transform that propagates a wave front from goal to start is used to assign a specific number to each grid element Graph-based coverage: First, it considers that the prior map information provided as a graph might be incomplete. Second, it accounts for environmental constraints, such as restrictions in certain directions in the graph (corresponding to a one-way street, for example). Third, it provides strategies for on-line re-planning when changes in the graph are detected by the robot's sensors when performing coverage. Finally, strategies for coverage using multiple robots are provided. (pic)

Grid-based methods: Grid-based methods use a representation of the environment decomposed into a collection of uniform grid cells. This grid representation was first proposed by Moravec and Elfers to map an indoor environment using a sonar ring mounted on a mobile robot. In this representation, each grid cell has an associated value stating whether an obstacle is present or if it is rather free space. The value can be either binary or a probability. Typically, each grid cell is a square, but also different grid cell shapes can be used, such as triangles. As grid representations only approximate the shape. spanning tree until it reaches the end of the tree. At that point, the robot turns around to traverse the other side of the tree. It is worth noticing that, when coverage is completed, the robot returns to the start cell, facilitating its collection and storage. On the other hand, STC never visits any small cell twice and thus minimizes the cover time. Fig. 21(b) shows an example of a coverage path generated by the Spiral-STC algorithm. wall finding. Nonetheless, they introduce a map coordinate assignment scheme based on the history of sensor readings to improve the time-to completion by reducing the number of turns on the generated path. The generated spiral paths are then linked by an in- verse distance transform they introduce. This proposal is validated in simulation and with real-world experiments conducted in- side a room with a mobile robot. Grid mapping movement on a surface: So starting with Acar and Chose whom proposed to plan the paths of their sensor based Morse decomposition approach by relying on the boundaries of each cell, so they can minimize the dead-reckoning error, but on the other hand Tully et al used a method that is combination of three robots that move together

to minimize the localization error and that was accomplished by using two off the three robots a static robots and the third one as a beacon to communicate, and on top of that the three robot fleet that was also designed to so the roles can be interchangeable. But the problem was that they are so good on two dimensional suffuses but when it came to realty the obstacles was not consideration. So, despite what was said Using multiple robots in a CPP task still has a lot of advantages over using a single robot and some of them are that you can reduce the time it takes and improve robustness, and as failure of some members of the robot team can be compensated by other members. Most approaches extend single-robot ideas presented before to multiple robots by using a strategy to divide the workload, which can be achieved by Algorithms for the complete coverage path planning problem using a team of mobile robots on an unknown environment The algorithms use the same planar cellular decomposition as the Boustrophedon single robot coverage algorithm. In summary the article/research paper keep going over the fact that using a single robot can be good enough in a lot of scenarios with a good algorithm or with an algorithm that is specially designed for the task, but using a fleet of robot also an idea that you should consider if you are determent to be efficient In the work/task that you are trying to finish specially because using a fleet of robot also can use the same algorithms and methods that a single robot is using but on a bigger scale and can minimize the effort the time and also improve the communication between the user and the robot in some scenarios (if one of the robot was used as a beacon). 3D coverage: Most coverage path planning methods, and in particular the methods reviewed so far in this article, assume that the environment can be modelled as a simple planar surface. However, some surfaces in nature are 3-dimensional, and 3-dimensional coverage path planning is required instead to cover these surfaces. This is the case of an autonomous underwater vehicle covering the seabed or a robot spray-painting vehicle parts, for instance. Next, we review several 3-dimensional coverage methods. Indeed, in 3-dimensional coverage, covering 2-dimensional surfaces embedded in 3-dimensional space such as the boundaries of automotive parts, the boundaries of buildings, the ocean floor, rugged agricultural fields or the boundaries of the in-water part of a ship hull are the main focus. 3D coverage using a planar coverage algorithm in successive horizontal planes: 3D coverage using a planar coverage algorithm in successive horizontal planes Hertel presented a 3D coverage algorithm that is based on a planar 2-dimensional terrain-covering algorithm. Their target application

is an autonomous underwater vehicle imaging the sea bottom. Their solution applies to a 3D protectively planar environment by applying the planar terrain-covering algorithm in the successive horizontal planes laying at different depths. Artificial inlets are covered in the same way that real diversion inlets are. Therefore, aiming to make the robot navigate only in areas close to the surface, artificial obstacles (artificial islands) are introduced in the robot's map of the environment. (pic)

3D cellular decomposition considered the problem of trajectory generation for spray-painting robots. In their early work, they proposed an on-line, 3-dimensional, on-line CPP method for closed, orientable surfaces embedded in R3. Addressing their spray-painting target application, the method does not plan a coverage path on the target surface, but the coverage path is rather planned in an offset surface from which the end effector will spray the target surface. However, if the surface is non-convex and includes elements such as a bifurcation, the planner will use the critical points occurring in such shape changes to divide the surface in cells that will be covered individually. As in the on-line Morse decomposition for planar spaces, a Reeb graph is used to encode the topology of the target surface. (pic)

Random sampling-based coverage of complex 3D structures to handle this family of problems, global path planning strategies, utilizing sampling-based planning have been applied to find feasible, collision-free, paths through confined areas and obtain full coverage of a 2-dimensional target structure. It should be noted that the generated paths cover cluttered spaces where complex structures such as shafts and rudders are present. The approach is validated using sensor imagery of real vessels and with experiments conducted at sea. As discussed above, the approach by Englut and Hover first generates a set of view configurations that completely cover the target surface (by solving an instance of the art gallery problem) and then finds a path that connects them (by solving an instance of the traveling salesman problem). To tackle this problem, presented a random sampling-based algorithm that incrementally explores the robot's configuration space while constructing an inspection path until all points on the target surface are guaranteed to be covered. optimal coverage Work addressing the optimality of the generated coverage paths, in terms such as path length and time to completion, appears in the CPP literature. Huang presented an optimal line-sweep based method for cellular decomposition algorithms in planar spaces. This approach produces an optimal length coverage path by allowing different sweep

directions in the lawnmower paths used to cover each cell. This is achieved by allowing a different sweep direction in each cell. Mandira and Realities proposed an algorithm based on the Boustrophedon cellular decomposition that achieves complete coverage of known spaces while minimizing the path of the robot. Component description: For the positioning part at the start we will begin with a sweep that will map the surface than will send it to our raspberry pi that calculate and create a map in its memory which is helpful for the robot when the movement part begins, when the movement stage starts the maps we have calculate and collect it will be applied to an algorithm that will divide the surface area into small parts (maps) that will also later get their own algorithm, applied to them and this part will be done by the raspberry pi in collaboration with the sensors that will be detecting all the edges, after we have calculated our algorithm a signal will be sent by the raspberry pi to dc motors and the sensors to initiate the movement from the position that it has stopped on, the dc motors and the sensor will now work combined to move the robot and detects all the edges and will later send the signal to the raspberry pi again to record the progress and send it via Wi-Fi or Bluetooth to the iPhone.

design alternatives: KITEROBOTICS window cleaning robot [3], they do not use a path detection system it's just a mix between horizontal and vertical movements that are done by machines which pull from each corner and that can be operated by a control panel. And They also do not use an algorithm it's just pre-programmed route. There is no obstacle avoidance system, but cable system follows the contours of every surface and brush rotation direction cleans all corners and indentations.

The advantages are:

• The robot can move faster than any robot that is designed to clean household windows.

• The robot can clear gapes that are between windows because the robot is connect to wires instead of being connected to the window itself.

The shortcomings are:

• The robot is only designed to skyscrapers that have a flat face that is covers with windows.

• There is no obstacle avoidance system, but cable system follows the contours of every surface and brush rotation direction cleans all corners and indentations.

The advantages:

• Some of the most obvious advantages beside being the first to do it, is that that used simple

programing that did the job that later will help them and other programmers and inventers to develop it better because the leaded the ground for others after them.

• They have focused on a different market that no body before them have thought of which is house-holds and small company and apartment buildings which gave them the advantage because they are no competitors in their market.

The shortcomings:

• The experiment was done is a small lab with not a lot of working/experiment hours.

• The version that used only came with a battery so the work duration will not be so long compared to those whom are connected and powered by wires.

• The robot did not have an algorithm that can handle wen the shape of window is different or with size of the window get scaled by a lot.

3.5 PROPOSED DESIGN

In this report we have focused on the idea of cleaning windows, but now going in details the basic idea of the robot, the controlling / circle of the robot can be divided into five stages the first one going the input part which can be received via a remote control or via the phone app, the second stage is the positing part which can be done by using the algorithms companied with the sensors that are connected to the device which can determine which side of the surfed you are on and whether the robot should move in certain direction instead of another, and at the end the moving and distant/area covered recording happen and all the data get sent to the raspberry pi that can coordinate with all the other part of the robot. For the path detection the whole map is first covered by the mapping sweep and after that the robot generate smaller maps and select the easiest way to clean each part/segment, then for the movement algorithms we choose grid mapping (Wavefront Algorithm) because it was the most suitable for our application (uses), at last regarding the obstacle Avoidance our obstacle avoidance system is based on our algorithm which first map the surface and that create smaller maps so our robot can avoid obstacles. And the part that allow as to archive our goals (1: the mother board: it needs a 5v power switch can be done by connecting the motherboard to a computer or by connecting the board to a wall outlet.)

(2: the remote control: the remote control indeed did come with (3x 3.7v battery) which we switched in place but keep it the same to keep the remount function)

(3: our robot: our robot needs to be connected to a power source while it is working but it does need to charge after 45 min of working so it does need a wall outlet from time to time).

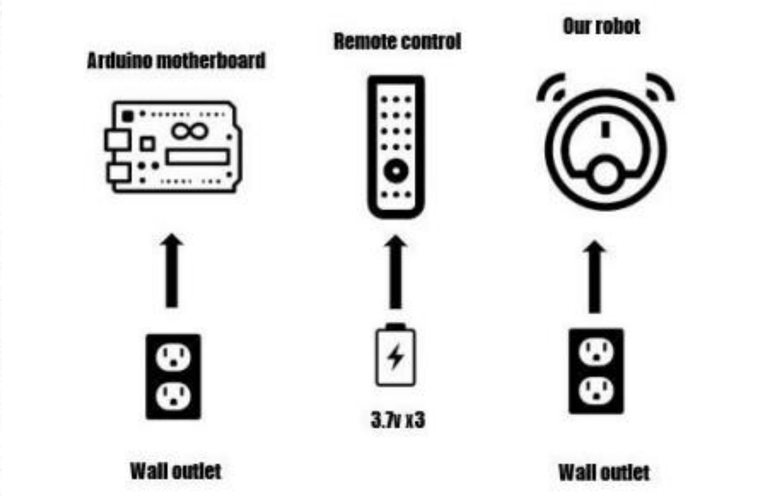

Figure 2

High level design of the integrated system:

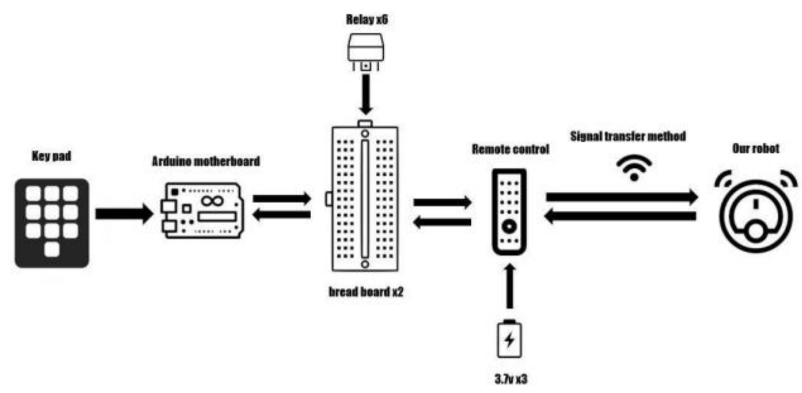

Figure 3

The basic idea of our report is that we are planning to modify the inputs that the robot is getting by the remote by using command and algorithms and stopping and forcing those inputs by using relays and switches.

3.6 TECHNOLOGY CONSIDERATIONS

Starting with one of the most important strengths of our robot is that it can save water by just using small amount of water or cleaner while its moving in a certain direction, and the second one that it can be a huge help for people who do not have the time or don't have a lot of time to waste on simple day to day home tasks, and the third on is that using and "AI" and "ML' to make our houses more smart can be a huge step if you are trying to make your house a smart house. On the other hand, there are some important weaknesses in our robot the most important one of them that the robot cannot jump from a window to window so if the robot finishes cleaning a window you have to stop what you are doing and move the robot from a window to another, the second weakness is that if the robot is powered by batteries the duration of working time wouldn't be as long as if the robot was powered by a wire straight from the wall outlet. Going over some of the important trade-offs that have been missed are using the robot to clean windows that have a walls or gaps between them without any human interference. design alternatives: KITEROBOTICS window cleaning robot [3], they do not use a path detection system it's just a mix between horizontal and vertical movements that are done by machines which pull from each corner and that can be operated by a control panel. And They also do not use an algorithm it's just pre-programmed route. There is no obstacle avoidance system, but cable system follows the contours of every surface and brush rotation direction cleans all corners and indentations.

The advantages are:

• The robot can move faster than any robot that is designed to clean household windows.

• The robot can clear gapes that are between windows because the robot is connect to wires instead of being connected to the window itself.

The shortcomings are:

• The robot is only designed to skyscrapers that have a flat face that is covers with windows.

• There is no obstacle avoidance system, but cable system follows the contours of every surface and brush rotation direction cleans all corners and indentations.

3.7 SAFETY CONSIDERATIONS

Our objective is to improve the quality of life inside households by integrating algorithm and machine learning by using robots which help with the day-to-day tasks like window cleaning. Our approach included purchasing a robot with a low price that have the tech and the algorithm to be efficient and high safety features, so we improved the robot by modifying the remote control that came with a robot to add and introduce new command that can help the robot to be more efficient and to add algorithms which help the robot to be a smart and efficient one, but our main focus was to keep some of the safety features like the safety pin which if not engaged the robot will not have any power and obstacle avoidance system which stops the robot when reaching and edge or a corner or maybe an obstacle.

4 TESTING AND IMPLEMENTATION

Since the robot doesn't have an application that we can test the simulation on, therefore we are examining our theories and algorithm in real life, so from the testing we found out that the robot needs 6.5 sec to cover one column which contains 3 cells ONLY, which means too cover all the cells equally the robot has to do 9 full loops just to have the first column covered equally 3 times and to finish all the cells for one loop it needs 19.5 sec which is too much. As shown in (figure 1)

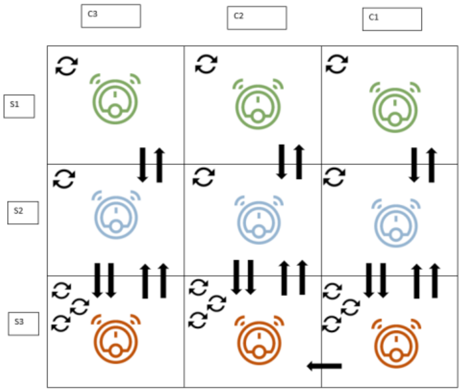

Figure 4

name	percentage	pass	Total time in each cell	Time to rotate in the cell	Total time to complete
S1	%100~	3	0.5s	3s	19.5
S2	%66~	2	0.5s	1s	
S3	%33~	1	0.5s	0s	

Table 1

Whereas the improvement we did on the robot which makes our robot more efficient if we are talking about covering cell or column equally. The time it takes to complete a full column is 2 sec and to complete all the cells is 6 sec which at the end makes our robot effective if we

are talking about covering all the cells equally (each cell should take 11%) which in total it takes 18 sec for the 3 loops. Because of our algorithm we did its more efficient than the origin algorithm.

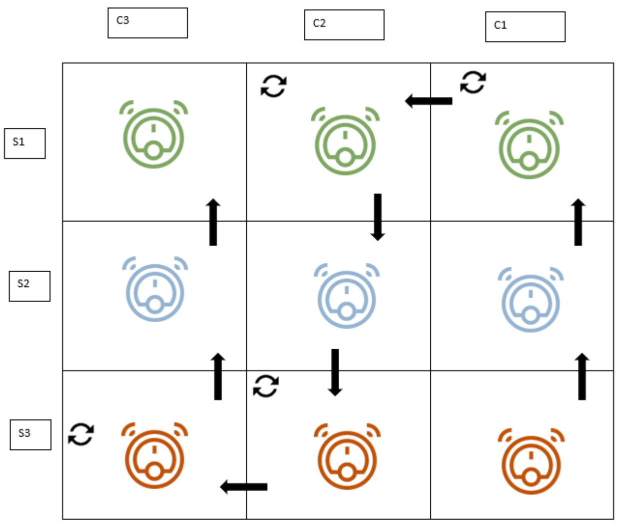

Figure 5

name	percentage	pass	Total time in each cell	Time to rotate in the cell	Total time to complete
S3	%11.111~	1	0.5s	0s	18s
S2	%11.111~	1	0.5s	0s	
S1	%11.111~	1	0.5s	1s	

Table 2

name	percentage	pass	Total time in each cell	Time to rotate in the cell	Total time to complete
S3	%11.111~	1	0.5s	1s	18s
S2	%11.111~	1	0.5s	0s	
S1	%11.111~	1	0.5s	1s	

Table 3

name	percentage	pass	Total time in each cell	Time to rotate in the cell	Total time to complete
S3	%11.111~	1	0.5s	1s	18s
S2	%11.111~	1	0.5s	0s	
S1	%11.111~	1	0.5s	0s	

Table 4

Figure 9 : the original robot coverage

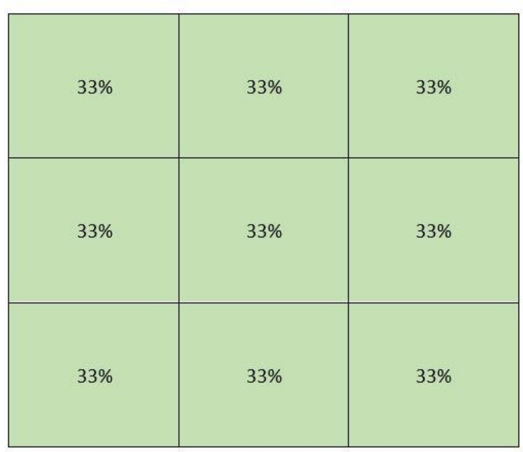

Figure 10: the coverage of the improved robot

So, from our testing we found that our robot takes less time to the over all cycle of the window and if we are talking about the loops not the cycle we can see that our robot does take only 18s(table 2, 3, 4) and that is the full cycle which is made out of 3 loops (each loop takes around 6.5s) compared to the other robot which takes 14.5 seconds to complete a loop and more that 30-40s to complete a cycle where each cell is covered 3 times

	Our robot	The original one
Total time to complete one cycle	6.5s	19.5
The improvement in each cycle	~ ± (15-17.5)%	-
Total time to complete one loop	18s	(30s – x) there is no specific time because it depends on the height of the window
The improvement in each loop	198% on average (if the window is 3x3 cells)	-
How many loops it takes to cover each cell 3 times	3 loops	9 loops
Time to cover each cell 3 times	18s	1.5-2.1 min
The improvement over all	More than 400% Because our robot takes $\frac{1}{7}$ the time that the original robot takes	-

Figure 11

4.1 INTERFACE SPECIFICATIONS

In terms of the software mainly the software one software witch I Arduino ide witch is an software that can be used to input algorithms and codes, so beside that we are only hoping that with in time we have the time to build a full application other than that we will only be going with original plan.

4.2 HARDWARE AND SOFTWARE

Hardware:

Since most of our test was done in real life, so due to nature of our project our testing was mostly done on window that was separated to 3x3 grid to be compatible with the algorithm that we choose which is the grid mapping algorithm which basically separate huge shapes into small cells that later can be calculated Separately, so to calculate the efficiency of our robot by calculating the time it takes to move from one cell to another, and after we gather our result then we create a chart with each cell numbers based on the row and column.

Software:

When testing our project due to the nature of it we don not have any software that we can apply our testing to, so most of our testing was done in real life with simple and basic tools that are easy to use like a ruler and a stop watch, but after that we used application like excel and MATLAB to calculate the error of margin and the percentage of improvement.

4.3 FUNCTIONAL TESTING

since from the factory our robot can only work with windows that do have an edge or a frame and the size of it should always be above >45cm~ so most of our testing was done on window with the dimension of 90cm~ overall which was later separated to 30x30 cells that each had a name and a number to be later inputted to our charts therefor we can calculate the improvement and since the robot moving speed is 2min/m^2 we have calculated that:

Adhesive type	:Vacuum, Square
Cleaning pattern	:Linear clean
Glass thickness limitation	:No limitation
Cleaning speed	:2 minitues/m²
Auto-cleaning mode	:3 modes
UPS(Un--interrupt Power System)	:20 mins
Frameless window	:Yes
Auto returns to starting point	:Yes
Auto-finish	:Yes
Safety rope	:Strength:150 Kgf
Workable window size	:>45cm x 45cm
Allowed surface roughness	:<1.5 mm
Machine size	:25(L) x 25 (W) x8.5 (H) cm
Weight	:1.35KG
Voltage Range	:100~240V, 50/60Hz
Power consumption	:75W
Micro-fiber cleaning cloth	:4 pcs

	NAME	PERSANTAGE	PASSES	TOTAL TIME IN EACH CELL	Rotation time in each loop	Rotation time	Total to complete the pass	Total time to complete a loop	Total time (3 loops)
	S1		1	0.5	0				
The first loop	S2	%11.1111	1	0.5	0	1s	2s		
	S3		1	1s	1				
	S1		1	1s	1				
The second loop	S2	%11.1111	1	0.5	0	1s	2.5	6.5s	18s
	S3		1	1s	1				
	S1		1	1s	1				
The third loop	S2	%11.1111	1	0.5	0	1s	2s		
	S3		1	0.5	0				

Table 5

And after a lot of testing we came to the conclusion that our robot is consistent with our testing with a margin of error that is less that "2.5%~ ((+-1%))" and since we only got the error only one time we can say that error margin is close to none, since each time there was an error the error was due to natural cusses not the algorithm or the robot itself.

4.4 NON-FUNCTIONAL TESTING

Since the robot does not have an application to test on, we are examining our theories and algorithm in real life. From the testing we found out that the robot needs 6.5 sec to cover one column which contains 3 cells ONLY, which means that overall we have improved over 15% from the original robot, and in term of security whether we are talking about psychical or software security we are up to standers, since our robot does have couple of safety major like stopping when realizing that one of the censors have went over the edge and when an obstacle is found our robot does not keep moving instead our robot makes a sound stops then takes a step back, and for software security our robot does not have a camera which can be counted as a advantage or a disadvantage for some people but since most our measurements will be taking be censors so we do not need a camera involved to keep the security our costumers houses, and in the future if we are planning to add a camera or an app we will keep the safety of our customers in mind and we will keep it up with engendering standers.

In terms of compatibility since most house hold widows are standard and since our robot can go over curved edges will just add a disclaimer to those who have odd shaped windows, to be around just for the first time when using the robot and after that the robot will have the window stored in its memory.

4.5 PROCESS

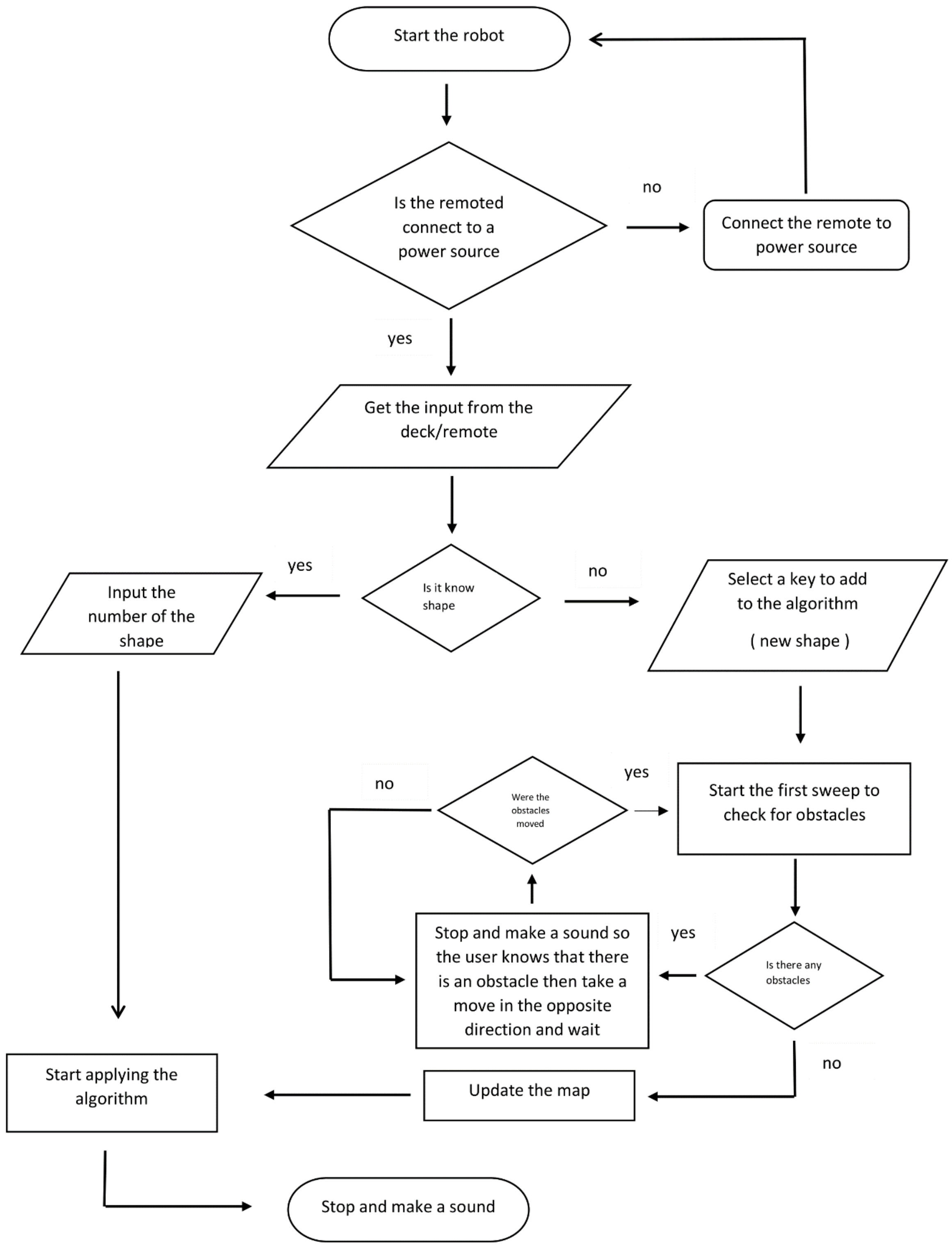

Figure 6

4.6 RESULTS

before starting with our process part here is a disclaimer that we wanted to get out the way one of them that the final robot is not the first version and we when through couple of variations to get to this point, so with that being said, our testing and implementation of the ideas that we have found be the research that we did in fyp1 started with finding the best cheapest robot that can be a good candidate to have all the appropriate upgrades for and after that we placed the order and within day we got our robot and started with our plan, the plan basically is upgraded the robot but without touching, modifying or removing any of the important part that the robot comes with like the safety for an example, and since our robot didn't have a lot of things other than safety we reached the conclusion that modifying the robot itself wouldn't be beneficial for us so we found a better alternative which is modifying the remote that moves the robot.

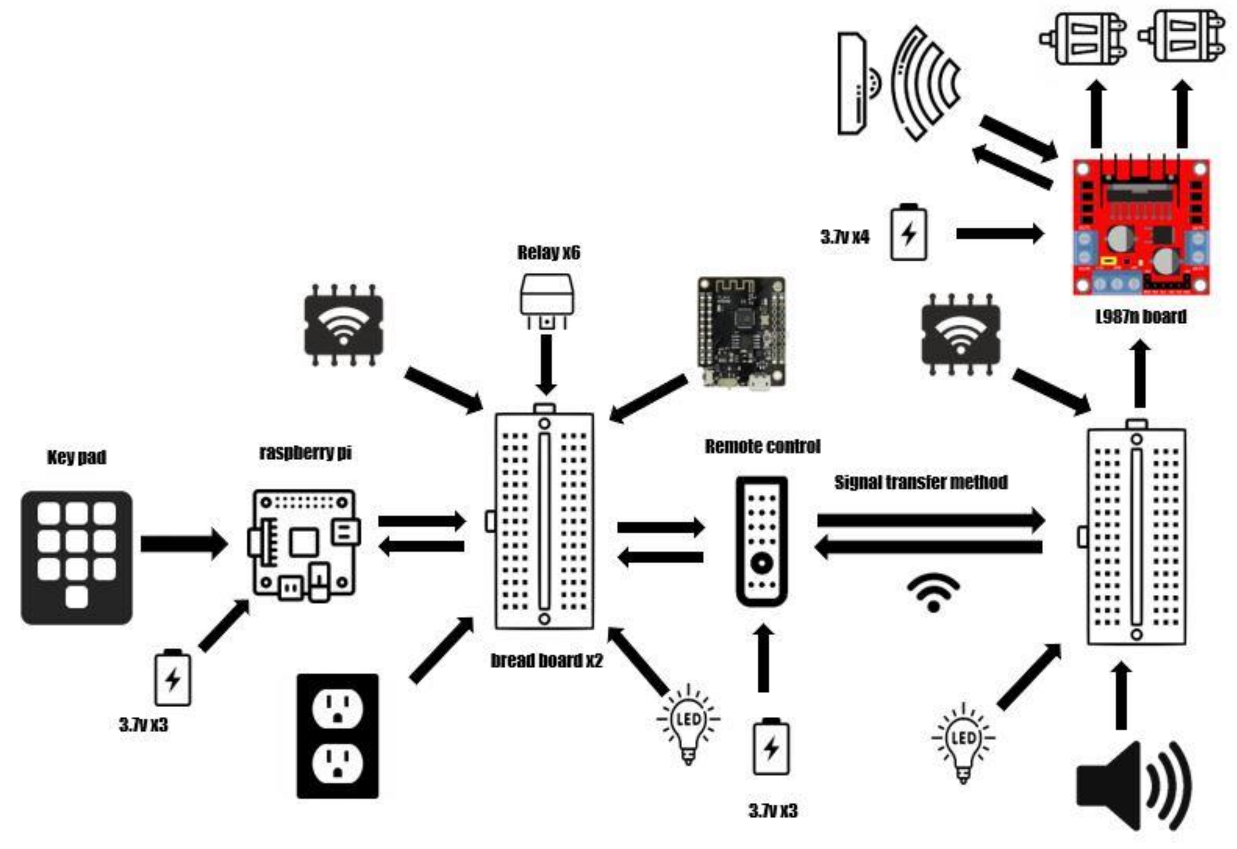

Figure 7

The first design that we came up with in fyp1

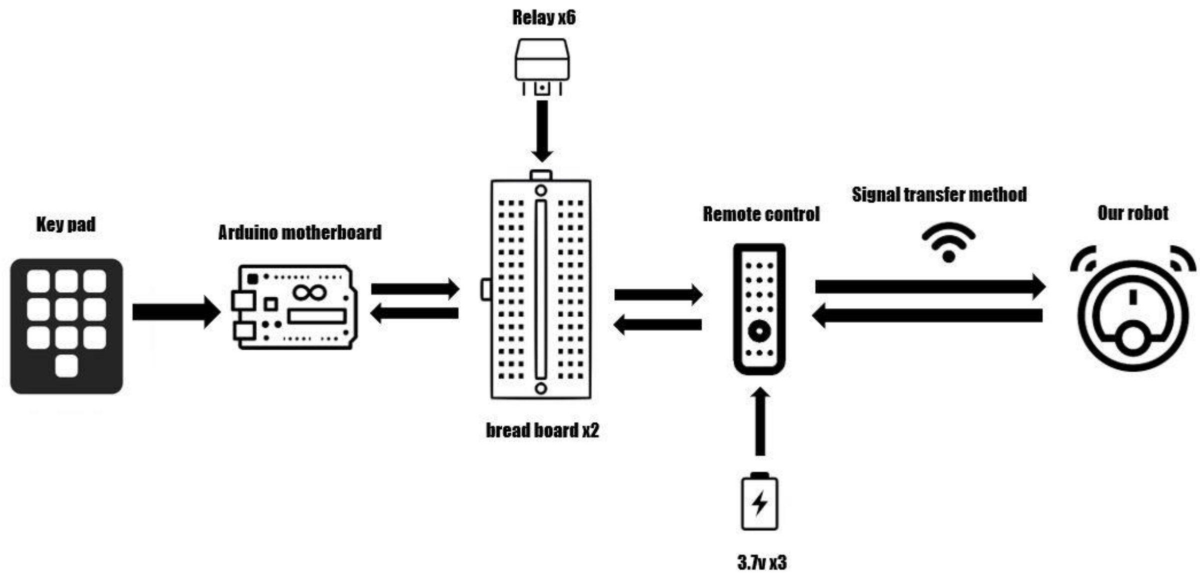

Figure 8

The final design that we came up with in fyp2

Since we have done a lot of testing and implantations of our fyp1 project in fyp2 we came to the conclusion that the robot does not need to be modified if we find the perfect fit on a low budget instead finding a robot with a high budget that cannot be modified, and our approach to this project has changed a lot with the knowledge that we gathered throughout our research so the our use of some of the part have also change from an example in fyp1 we mentioned that we are going to use a rassparry pi but after a lot of research and budget management we came with the conclusion that a small Arduino board can cost less that 1/16 of the price and be as affection as we want it to be since we are only going to apply code so we do not need a full robot with huge calculating power, and in terms speed and usability we learned that we do not need fancy apps or expensive robot to improve the robot instead we just simply and critical thinking mixed with some knowledge that we have learned from the courses that we took in our engendering journey, like using mat lab and basic math equations to calculate the improvements that we did which in the end was only 15%, but in fyp1 was estimated to be over 20% so at the end we lead that with good tools and team work so hard tasks can be done and improved.

5 PROJECT TIMELINE, ESTIMATED RESOURCES AND CHALLENGES

5.1 PROJECT TIMELINE

id	Task name	start	finish	duration	before	September					October				November					December					January			
					May-September	1-3	4-10	11-17	18-24	25-30	1-8	9-15	16-22	22-31	1-5	6-12	13-19	20-26	27-30	1-3	4-10	11-17	18-24	25-31	1-7	8-14	1-21	22-28
1	Fixing the ideas	2	9	2w																								
2	Reasrch1	11	16	1w																								
3	Research2	18	30	2w																								
4	ordering	5	29	4w																								
5	Testing1	1	13	2w																								
6	Testing 2	16	20	1w																								
7	Try and error	29	5	11w																								
8	Deliverable 1	16	18	2d																								
9	Deliverable 2	3	9	2w																								
10	Deliverable 3	10	13	3d																								
11	Final report	2	17	1w																								
12	poster	11	15	1w																								
13	Final presentation	11	1	4w																								

Table 6

5.2 FEASIBILITY ASSESSMENT

The window clean robot provides an ease of use for all ages especially elder people. The second one is the market feasibility, here with our product we are targeting Kuwait and The GCC Market because of the dusty weather conditions in particular the targeted people and those who have a lot of windows or glass surfaces which is risky to clean it and needs an effort to do it, therefore we are solving this issue by selling (window clean robot), for sure there are competitors however our aim is to sweep the market by selling our product with an affordable price, safe home usage, light weighted and high efficiency.

5.3 PERSONNEL EFFORT REQUIREMENTS

Task name	Who did it	duration
Researching the robot	G1	1W
Researching for compatible parts	G2	5D
Ordering the robot	G1	1D
Writing the software	G2	9D
Connecting the hardware	G1	2D
Putting the robot together	G1&G2	1D
Putting together the test area	G2	3D
Making the tweaks	G2	4D
Making hard ware improvements	G1	6D
trying to build the app	G2	4D
Software testing p2	G2	2D
Hardware testing p1	G1	1D
Software testing p3	G1&G2	2D
calculating the improvements	G1&G2	1D
Making sketches	G1&G2	1D
Improving the design	G1&G2	3D
Reconnecting the hardware	G1&G2	1D
Ordering new hardware	G1&G2	5D
Re-testing the software	G2	2D
Calculating the cost	G1&G2	1D
Submitting the deliverables	G1&G2	3W
Writing the final report	G1&G2	1W
Making the presentation	G1&G2	3W
Drawing and making the tables	G2	1D
Modeling and editing the report	G1	2D
Going over all what was submitted	G1&G2	1W

Table 7

5.4 OTHER RESOURCE REQUIREMENTS

There are some hardware/software resources that will not be included in the total cost of the project which are:

- **A computer, to program all the codes needed for the project.**
- **Object, to do the test for avoiding obstacles/objects feature we can use any object.**
- **Plastic, using it in case body for the device.**

The software resources that are not included in the total cost:

- **Using C++ programming language**
- **C#**
- **Python**

5.5 FINANCIAL REQUIREMENTS

Name	Store name: RTC		
	Qyt	Details	Price
LDR sensors	X3	To detect light	1.000 KD
LR sensors	X4	To detect edges and corners	4.000 KD
The robot	X1	The main body	32.000 KD
Arduino mega 2650	X1	Used as a motherboard	11.00 KD
Relay module 2 way	X1	-	1.500 KD
Relay 4-channal	X1	-	3.000 KD
Key pad	X1	-	1.500 KD
Switch (on / off)	X1	-	0.150 KD
Wires and other connection boards	X1	-	3.500 KD
Battery box AA	X1	-	0.250 KD
Battery pack AA	X1	-	0.500 KD
Case	N/A	Should be built using a 3d printer	0.400 KD
Programing	N/A	To program the controller	N/A
TOTAL PRICE			58.800 KD

Table 8

6 CLOSING MATERIAL

6.1 CONCLUSION

In conclusion, Window cleaning robots have become increasingly popular in recent years as a way to efficiently and effectively clean the windows of a home or office building. These robotic cleaners are equipped with a variety of features, such as sensors, brushes, and squeegees, to remove dirt, dust, and streaks from the surface of the glass. They are able to navigate around obstacles and reach difficult-to-access areas, making the task of window cleaning much easier and less time-consuming for the user.

One of the main benefits of using a window cleaning robot is the convenience factor. Instead of having to manually clean the windows, which can be a tedious and physically demanding task, the robot can do the work for you. This can be especially useful for those who live in high-rise buildings or have windows that are difficult to reach.

Another advantage of window cleaning robots is their ability to clean a large number of windows in a relatively short period of time. This makes them particularly useful for commercial properties, where the windows may need to be cleaned on a regular basis. In addition, these robots are typically equipped with advanced sensors and algorithms that allow them to navigate around obstacles and avoid collisions, making them both safe and efficient.

However, it's important to note that window cleaning robots do have some limitations. They typically require a nearby power source to function, which can be an issue for those with large or remote properties. In addition, they may not be able to clean certain types of windows, such as those made of wood or with intricate designs, as effectively as a human.

Overall, window cleaning robots can be a valuable addition to a cleaning routine, providing a convenient and efficient way to keep windows looking clean and clear. While they may not be able to replace human cleaners in every situation, they can certainly make the task of window cleaning much easier and more efficient.

7 REFERENCES

[1] Young-Ho, c., & Kwang-Mok, J. (2011). Windoro: The world's first commercialized window cleaning robot for domestic use. International Conference on Ubiquitous Robots and Ambient Intelligence (URAI). Retrieved 5 1, 2022, from https://ieeexplore.ieee.org/document/6145947

[2] Young-Ho, c., Jae-Youl, L., Jong-Deuk, L., & Ka-Eun, L. (2012). SMART WINDORO V1.0: Smart window cleaning robot. International Conference on Ubiquitous Robots and Ambient Intelligence (URAI). Retrieved 5 1, 2022, from https://ieeexplore.ieee.org/document/6462948

[3] kiterobotics. (22). Revolutionary facade cleaning robot for high buildings. Retrieved from meet kiterobotics: https://www.kiterobotics.com/en/

[4] Ontario Professional Engineers Act. (n.d.). Retrieved May 11, 2022, from https://engineerscanada.ca/sites/default/files/Ontario/Ontario_Professional_ Engineers_Act.pd

[5] Professional engineers act - R.R.O. 1990, reg. 941. (n.d.). Retrieved May 11, 2022, from https://engineerscanada.ca/sites/default/files/Ontario/Ontario_Reg_941.pd

[6] Code of ethics. Professional Engineers Ontario. (n.d.). Retrieved May 11, 2022, from https://peo.on.ca/licence-holders/code-ethics

[7] Improvement, H., & Outlets? S. (2022). Should You Use 12 or 14-Gauge Wire for Outlets? | Pepper's Home & Garden. Retrieved 10 May 2022, from https://peppershomeandgarden.com/12-or-14-gauge-wire-outlets/

[8] Best arduino RC cars (including Bluetooth RC cars). All3DP. (2021, September 17). Retrieved May 11, 2022, from https://all3dp.com/2/arduino-rc-carbluetooth-project Page 34 of 36

[9] Home. KITE Robotics - Veilig en accuraat glas en gevel reinigen op grote hoogte. (n.d.). Retrieved May 11, 2022, from https://www.kiterobotics.com/en/

[10] **Skyline Robotics: The future of automated window cleaning. (n.d.). Retrieved May 11, 2022, from https://www.skylinerobotics.com/**

[11]Miao, X., Lee, J., & Kang, B. (2018). Scalable Coverage Path Planning for Cleaning Robots Using Rectangular Map Decomposition on Large Environments. IEEE Access, 6, 38200-38215. doi: 10.1109/access.2018.2853146

[12]Galceran, E., & Carreras, M. (2013). A survey on coverage path planning for robotics. Robotics And Autonomous Systems, 61(12), 1258-1276. doi: 10.1016/j.robot.2013.09.004

[13]Ajeil, F., Ibraheem, I., Azar, A., & Humaidi, A. (2020). Grid-Based Mobile Robot Path Planning Using Aging-Based Ant Colony Optimization Algorithm in Static and Dynamic Environments. Sensors, 20(7), 1880. doi: 10.3390/s20071880

[14]Almadhoun, R., Taha, T., Seneviratne, L., & Zweiri, Y. (2019). A survey on multi-robot coverage path planning for model reconstruction and mapping. SN Applied Sciences, 1(8). doi: 10.1007/s42452-019-0872-y

[15]Ajeil, F., Ibraheem, I., Azar, A., & Humaidi, A. (2020). Grid-Based Mobile Robot Path Planning Using Aging-Based Ant Colony Optimization Algorithm in Static and Dynamic Environments. Sensors, 20(7), 1880. doi: 10.3390/s20071880

[16]Matt. (2022, May 6). Embrace lazy, why you need window cleaning robot. Little Robot Shop. Retrieved May 10, 2022, from https://littlerobotshop.com/are-robotwindow-cleaners-any-good/ Page 35 of 36

[17]Connaughton, N., says:, R. V., says:, N. C., & says:, H. J. L. (2016, March 9). Roomba algorithms and visualization. Code and a glass of wine. Retrieved May 10, 2022, from https://blog.niallconnaughton.com/2016/01/25/roomba-algorithms-andvisualization/

[18]Do robot window cleaners work? Robot Chores. (2021, October 4). Retrieved May 10, 2022, from https://www.robotchores.com/do-robot-window-cleaners-work/

[19]irobot. (2016, August 4). Clean Floors with the press of a button | roomba® 900 series | iRobot®. YouTube. Retrieved May 10, 2022, from https://www.youtube.com/watch?v=XIPzSmwClJ8

[20]YouTube. (2020, March 1). IRobot vacuum cleaner review in 2020: It's almost perfect! (iRobot Roomba E6 review). YouTube. Retrieved May 10, 2022, from https://www.youtube.com/watch?v=84wHv36s7tU&t=485s

[20] Window Cleaner Robot. (n.d.). American Fleet Inc. https://wexbi-home.com/products/window-cleaner-robot

7.1 APPENDICES

✗ IT DOES NOT MANAGE TO CLEAN CORNERS OF THE WINDOW WITH ROUND MECHANISM AND PADS	✓ IT CLEANS ALL CORNERS OF THE WINDOW WITHOUT LEAVING ANY STREAKS OR WATER MARKS
✗ THE CORD IS TOO SHORT TO CLEAN LARGE WINDOWS	✓ HIGHLY EFFECTIVE WITH LARGE WINDOWS
✗ THIS MACHINE CAN NOT MOVE IN UP DIRECTION	✓ ROBOT CAN MOVE IN UP DIRECTION
✗ IT WORKS VERY LOUD WHEN WASHING WINDOWS	✓ MUCH LESS NOISY THAN MANY OF THE LEADING SMART WINDOW CLEANING ROBOTS
✗ ROBOT CAN TAKE A LONG AMOUNT OF TIME TO CLEAN A WINDOW	✓ CLEANS FAST. CAPABLE OF REMAINING ATTACHED TO A WINDOW FOR THIRTY MINUTES AFTER THE BATTERY RUNS OUT
	✓ UPON PURCHASE YOU RECEIVE A GIFT - TRACK WINDOW CLEANING BRUSH

IMPORTANT SAFETY PRECAUTIONS

Read all instructions before operating the device

WARNING

.Fasten the safety rope and tie to indoor fixture before operating the device
.Check if the safety rope is broken or knot is loose before operating the device
.Set danger-warning area downstairs for no-balcony window and door
.Battery shall be fully-charged(green lamp ON) before using
.Do not use it in rainy or moist day
.Make sure appliance can stick and stand on glass surface while put onto glass
.Make sure put cleaning pad onto appliance correctly for no leakage
.Do not spray or pour water onto the appliance
.Prohibit children from using this device
.Remove anything on the glass. Do not use on cracked glass. Reflective or coating glass might be slightly scratched due to dirt on the glass while operating
.Don't use in areas with flammable or combustible liquids, gas.

Features & Specifications

Adhesive type	:Vacuum, Square
Cleaning pattern	:Linear clean
Glass thickness limitation	:No limitation
Cleaning speed	:2 minitues/m²
Auto-cleaning mode	:3 modes
UPS(Un--interrupt Power System)	:20 mins
Frameless window	:Yes
Auto returns to starting point	:Yes
Auto-finish	:Yes
Safety rope	:Strength:150 Kgf
Workable window size	:>45cm x 45cm
Allowed surface roughness	:<1.5 mm
Machine size	:25(L) x 25 (W) x8.5 (H) cm
Weight	:1.35KG
Voltage Range	:100~240V, 50/60Hz
Power consumption	:75W
Micro-fiber cleaning cloth	:4 pcs
Certification	:CE. FCC. RoHS
Available window	:Medium & Large sized window

Accessories list

Product | Power adapter | Cleaning pad | Safety rope
Instruction manual | Warranty card

Last step is to start the deck and choose a number