Window Washing Robot
Design Concept Proposal

Team B

Monday January 26th 2015

Stephanie Chen
Trevor Decker
Ian Hartwig
Ian Rosado

Contents

1 Introduction and Design Requirements

The goal of the Window Washer is to automatically clean one side of a glass or plexiglass window without scratching or damaging the surface. The robot has to accommodate any window size within the range of 3'x4' to 5'x6' with a solid frame. The robot has no required start position but the robot must be able to clear randomly applied dirt or streaking. Some of the other mandatory requirement include the following:

- 100% coverage with speed greater than or equal to 10 ft^2/min
- 2 ft^2 footprint when retracted, if applicable, with any aspect ratio
- cannot be supported from the ground
- no external structure can be installed other than a pneumatic compressor

Success will be based on visual inspection 3 feet away with no streaking or residual moisture 30 seconds after washing is complete.

As a team, it was also decided that the robot should be able to move from window to window in order to be able to clean multiple adjacent windows. To do this, the robot must be able to pass over the window frame and continue to clean. Ideally, we want our robot to be practical to use on tall buildings.

2 Functional Architecture

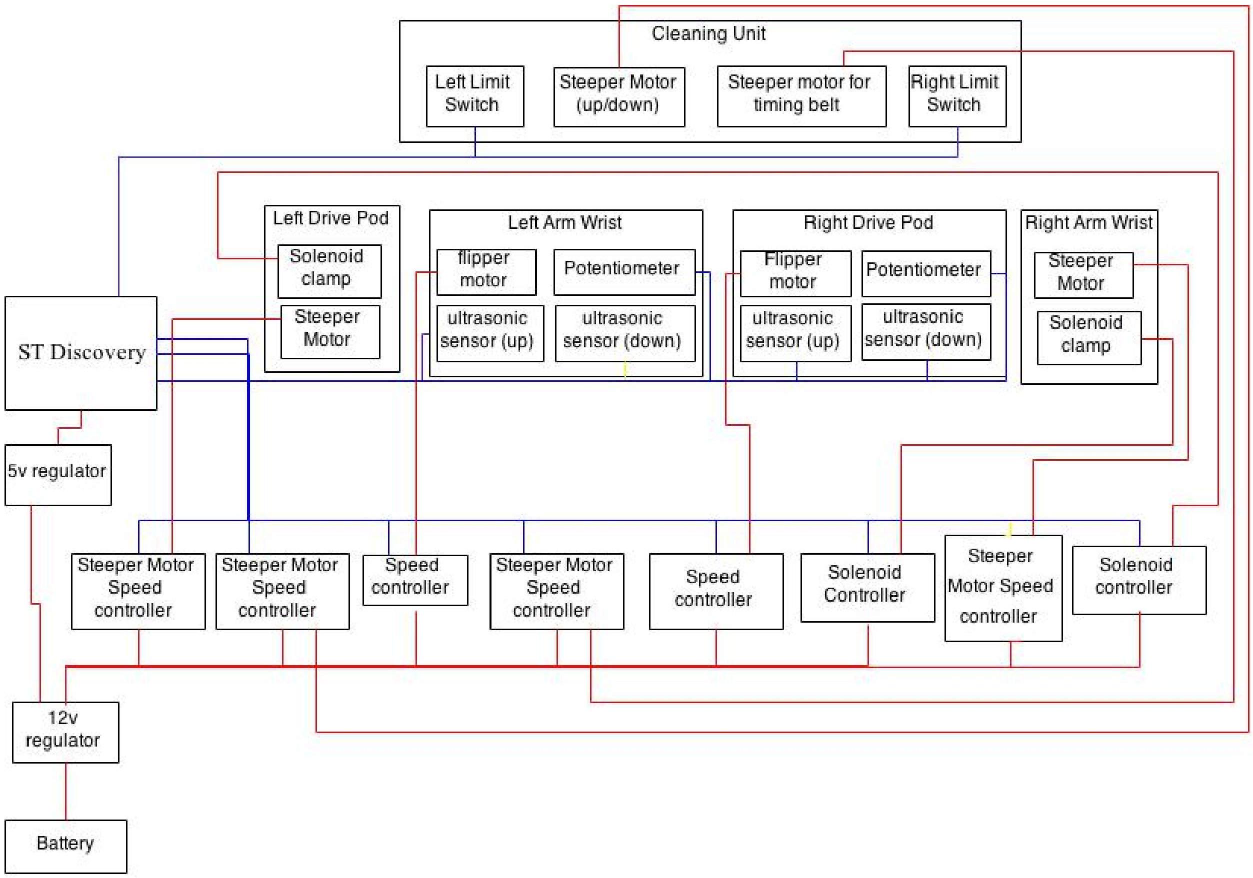

Figure 1. Functional Architecture

2.1 Control

We will be using an asynchronous parallel control system with 4 threads (localization,window path planning, safety, logging). We will be using a real time OS to allow for all four threads to run simultaneously.

2.2 Localization

The localization thread will manage the robot understanding of its location. When a new sensor event occurs a callback will fire. Causing for the internal representation of the robots internal understanding of its state to change. Each sensor input will be filtered over time using a kalman filter, giving us an evolving state estimate and confidence of each state variable. The state will be available to all of the other threads.

2.3 Window Path Planning

To minimize the travel time of our robot and to minimize the amount of resources (power/cleaning fluid) that the robot uses we will pre plan a path on the building. This can be split into a macro and micro problem.

Macro Problem

Create a graph of all window area's to be cleaned. The nodes will be the window sections that need to be cleaned. The edges will be the cost of moving between each set of adjacent window areas. An MST (minimum spanning tree) will then be created from the graph. The Robot will then follow the MST by executing a depth first search. Each window will have four nodes on it one on the top left, one on the top right, one on the bottom left, one on the bottom right. The cost of moving between each of the four corners will be very low compared to the cost of moving between the window panes. The four corners are needed incase windows of different size exist on a window panel.

Micro Problem

Move to the top or bottom of the window, snake away from the horizontal edge that the robot is currently at. While the robot is moving along the window it will be verifying that the window is cleaned.

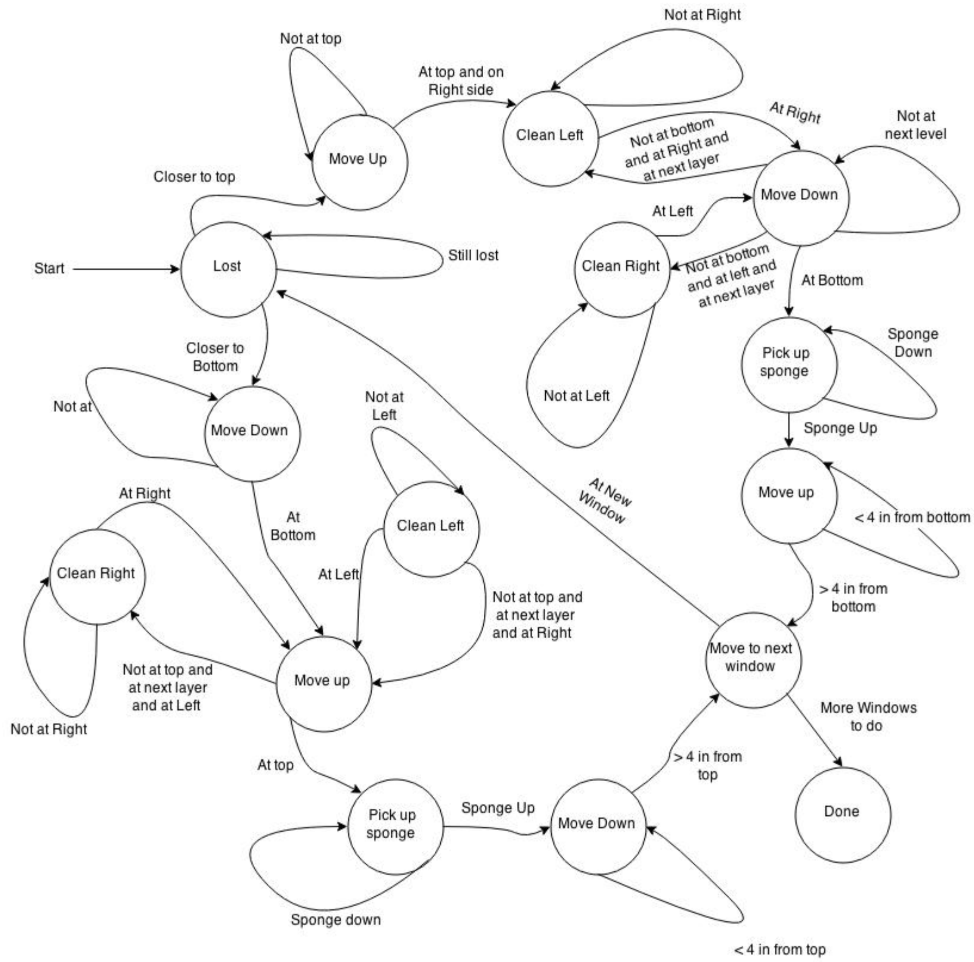

Figure 2. Window path planning of micro cleaning process

2.4 Logging System

The system logger will record the state of the system. This information will be written to disk locally and broadcast over a radio serial link so that a monitoring computer can view the current state of the robot.

2.5 Safety System

We will have two error states. Immediate stop: Shut down right away, this is what will happen when an emergency stop button (physical or virtual) is pressed or a critical error is encountered. This error state will be used sparingly as it causes for the robot to stop all operations. This can be dangerous if the robot were to be in the middle of a movement. System Error: Shutdown when statically stable: this error state is designed for when the robot encounters an error which stops it from being able to continue its job but does not impose any potential for harm to the robot or people around the robot.

A thread will be running which continually checks the following assertions, if any of them are ever false then an error state will be entered into until the robot is reset by an operator.

- physical emergency stop pressed
- emergency stop signal sent via wireless serial link from monitoring laptop
- battery power extremely low
- position estimate is outside of bounds

3 Design Trade Studies

3.1 Suction Attachment

We considered the suction attachment to the window because it would allow us to move freely to any part of the window. The base of the Winbot 730 by Ecovacs has an inner ring that is connected to a vacuum pump which holds the robot to the window. The function of the outer rings is to detect a loss of suction.

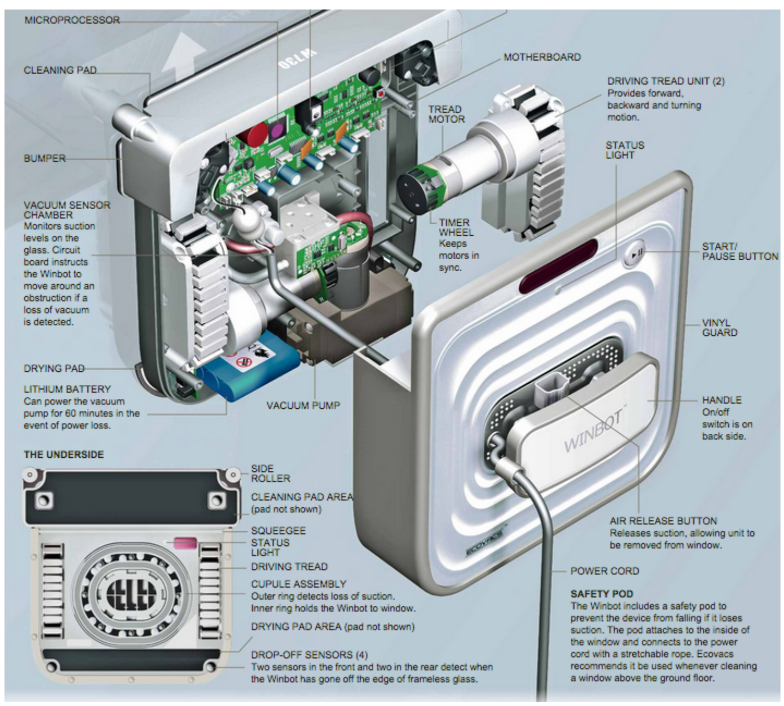

Figure 3. The internal structure of a Winbot 730 by Ecovacs [1]

One of the disadvantages of the this method include streaking caused by the suction mechanism and loss of suction due to dirtying of the surface. Also, the area under the suction mechanism will be left uncleaned in the end. Therefore, we decided to explore other attachment options.

3.2 Magnetic Attachment

We briefly discussed using magnets to secure the robot and clean both sides of the window. However, since having a magnetic attachment on the inside would be unrealistic in the cleaning of skyscrapers, we decided to pursue other avenues.

3.3 Jointed Arms

Our design relies on moving a cleaning head to a spot on the window from a fixed location on the frame. Common options for this are a mutli-jointed arm, a telescopic arm, or an x-y grid of linear slides. For this application, we would need multiple arm joins in order to keep the torsional load from the arm on the frame gripper due to gravity low enough so that our robot does not fall off the frame. With this many joints (likely 8, 4 on each side of our cleaning head) we were concerned about the mechanical complexity and play in the joints. We did consider driving multiple legs off of one motor and belt drive in an accordion configuration since we only want a linear action, but this can more aptly be accomplished with a telescoping arm.

3.4 Movement by Flight

The possibility of using a quadrotor or another type of flying mechanism as a means of locomotion was considered. This would allow for the robot to be very versatile and mobile, but in the end, issues with weight, controls, precision, and varying expected air conditions (outside of a skyscraper) rendered this idea unfeasible.

4 Physical Architecture

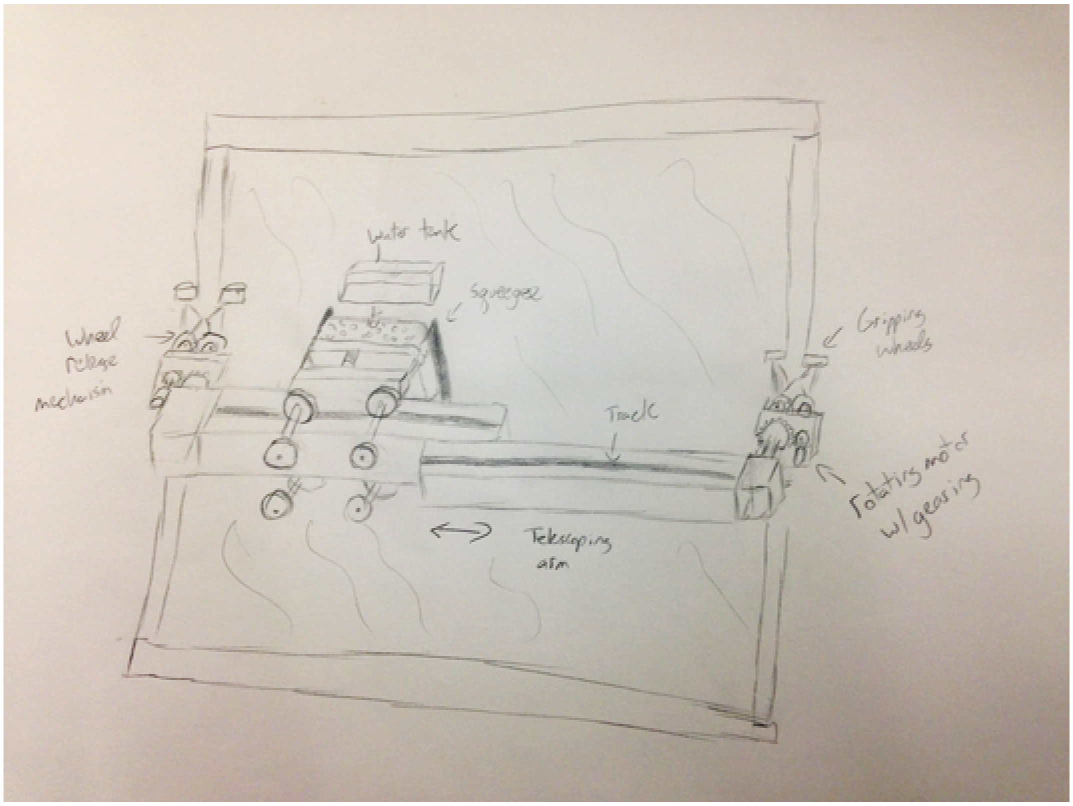

Figure 4. Sketch of proposed design

4.1 Side Gripper

The side gripper is used to hold the robot to the wooden window frame. The following figure shows a side view of the proposed gripper where the frame would fit between the passive and active wheels.

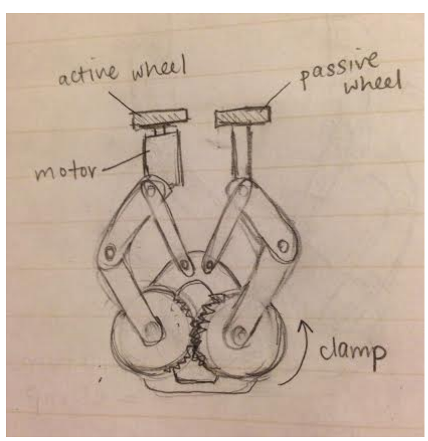

Figure 5. Clamping mechanism for holding robot on frame

In this mechanism, as the motor attached to the right gear spins counter-clockwise, the gear on the right will spin clockwise and cause the linkages to rotate about the pinned connections. This will draw in the wheels and allow them to apply a normal force to the frame. Since the robot needs to be able to be able to travel up and down the frame, the motor attached to one of the wheels will spin the wheels perpendicular to the frame to allow the robot to reach different parts of the window.

4.2 Telescoping Arm

A core feature of our design requires moving a cleaning unit horizontally on the window. To do this, we plan to build a linear slide that the cleaning unit can move on. This slide may be powered by timing belt or a winch solution. To adjust for different size windows, we need the slide to be built on a telescoping arm that can adjust between the 3 and 6 foot bounds of window size. We can likely accomplish this in 2 or 3 stages. Stepper motors with limit switches can be used to control movement on the arm with good speed and accuracy, and zeroing position after the arm has changed length is not an issue. The stiffness of the arm is definitely a concern and the appropriate size tubing or beam can be found using the following deflection equation for aluminum, assuming that our arm will be holding approximately 5 lbs at one end with a fixed connection on the other:

$$\delta = \frac{PL^3}{3EI}$$

$$\delta \geq \frac{5\,lb * (6\,ft * 12\,in/ft)^3}{3(10000000\,psi) * \pi * (d_o^4 - d_i^4)}$$

$$d_o^4 - d_i^4 \geq \frac{1.2673}{\delta}$$

By choosing an acceptable deflection, δ, for our application, we are able to calculate the outer and inner diameter of the tubing or beam that we need.

4.3 Cleaning Unit

The actual cleaning unit that will translate from side to side on the telescoping arm comprises of a large sponge and two separate squeegee edges. The entire cleaning unit can be raised and lowered from the window surface using a lead screw and a small motor. At rest, the two squeegees extend past the sponge surface, supported by springs. As the unit is driven further towards the window, the sponge will come into contact as well. With the sponge in contact with the glass surface, the unit will move across the window in order to clean it. After moving across the window, the sponge will retract, and the squeegees will remove leftover water before the entire unit moves down and cleans the next lower “layer” of the window. The sponge will be kept moist with water and cleaning fluid using two separate storage tanks located on the top and bottom of the unit. Two separate tanks are needed for when the robot flips upside down. The tanks will have small holes facing the sponge, and the fluid will be gravity fed into the sponge. The unit will be guided along the telescoping arm using small motorized wheels and limit switches to maintain knowledge of cleaning unit position.

4.4 Flipper

In order to achieve one of the coolness factors we have identified -- cleaning a grid of windows with dividers -- we need to be able to crawl from one window frame to another. We believe the most effective solution is a 180º joint where each gripper meets the telescoping arm. The joint allows movement in the plane parallel to the window. This allows us to move the main cleaning and location assembly up, down, left, or right on a grid of windows with frames. To reduce the side of motor required to move the arm, especially when it is nearly horizontal to the ground, the arm should retract completely before a latch in the gearbox unlocks from horizontal. If the robot is currently in position A, it should follow movement steps B1, B2, B3 to move up or down (position B), or steps C1, C2, C3 to move left or right (position C).

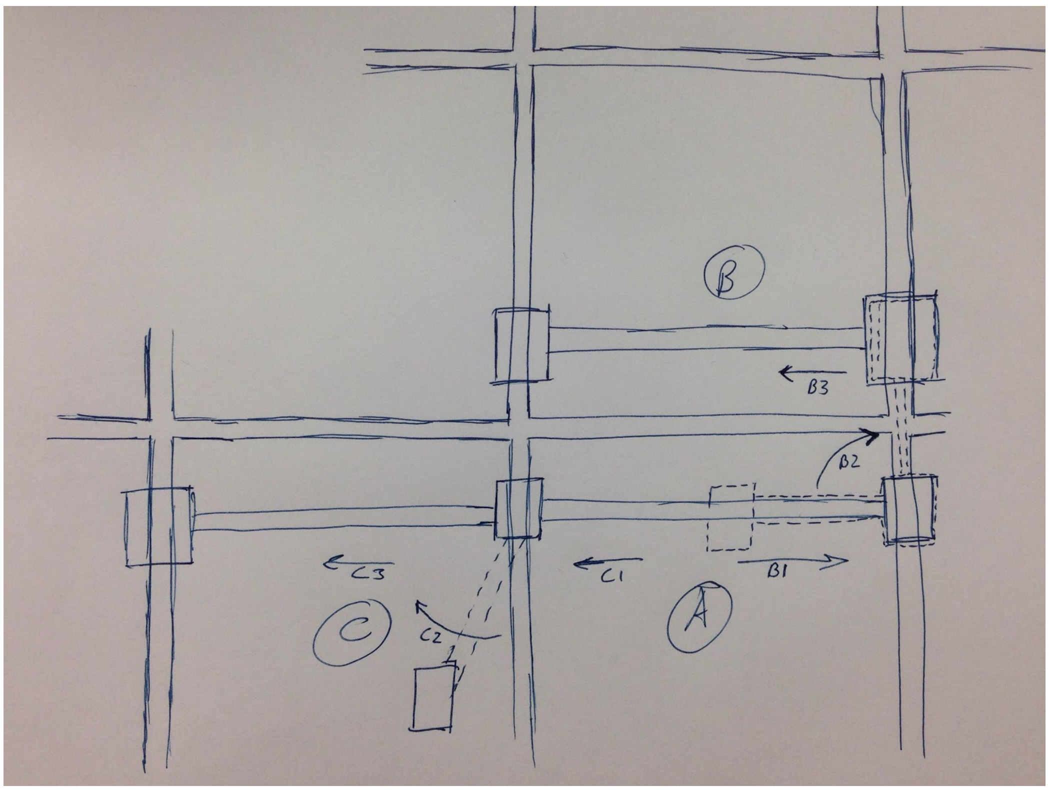

Figure 6. Diagram of window to window moving process

Another design considered was a 180º joint in the same location but with movement in a plane perpendicular to the window. This would also require us to make a symmetrical cleaning head since odd windows would see the front of the head and even windows would see the back. Additionally, this design would not have been able to reach the next row of windows below or above.

4.5 Sensors

4.5.1 Cleaning head position

The lateral motion of the cleaning unit will be powered by a stepper motor, this way we can have a pretty good understanding of the cleaning systems lateral position

At either edge of the arm we will have limit switches which the cleaning unit runs into when it reaches that side of the window. Whenever A limit switch is hit we can re zero the cleaning units lateral position.

4.5.2 Power Level

We will be monitoring and logging the robots battery level, This way we can shut down the robot while it is statically stable. The power level will be determined by monitoring the voltage of the battery and using a lookup table which we will produce of how voltage correlates to remaining runtime. The lookup table will be empirically derived.

4.5.3 Cleaning fluid level

We will be monitoring the amount of cleaning fluid that is still in the robot so that we can alert the user that continuing to run the robot would be wasteful. The sensing of the amount of cleaning fluid will be done with a pressure sensor in the cleaning fluid storage tank.

4.5.4 Position on Window

Ultrasonic sensors will be attached to each drive module point up and down, for a total of 4 sensors. The sensors will be pointing towards the top and bottom border of the window. This way we will be able to keep track of the absolute position of each drive unit. Each of the arm hinges will have a potentiometer attached to it. This way the robot will be able to track the position of the arm. To track the relative position of the robot as we move up and down the window we will use the commanded distance to stepper motors. We can then fuse the noisey

4.6 Processing

We would like to choose a microcontroller that has a good variety of complex IO pins (for multiple sets of PWM/servo and motor controller outputs) and ample processing power for the right price. We are already familiar with the AVR line of controllers from previous coursework and projects, but are more heavily considering the ARM Cortex line of microcontrollers. ST Microelectronics produces good, low cost evaluation boards for M0-M4 type controller including on-board support for USB, multiple UARTs, a real time OS, and possibly a touch screen interface. Depending on our processing workload, we may use either the ST Discovery 32F429 [2] or the ST Discovery 32F072 [3].

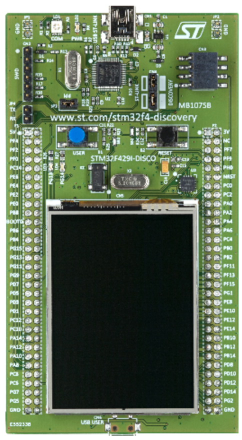

Figure 7. ST Discovery 32F429

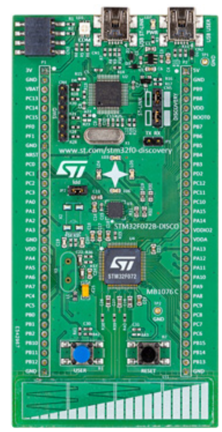

Figure 8. ST Discovery 32F072

5 Schedule

	Week															
Activity	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Website Design	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Mechanical Mockup	■	■														
Mechanism Design	■	■	■	■												
Electrical Design	■	■	■	■												
Mechanism Construction				■	■	■	■	■								
Sensor Bringup				■	■	■										
Spring Break							■									
Software Construction								■	■	■						
Mechanism Test								■	■	■						
Integration Testing									■	■	■	■				
Emergency Redesign												■	■	■	■	
Final Report														■	■	
Lab Clean Up																■

6 Estimated Budget

Description	Vendor	Estimated Cost	Quantity	Total
Aluminum Stock	McMaster-Carr	$47.75	1	$47.75
Acrylic Sheets	Amazon	$7.99	2	$15.98
Delrin Stock				
Standard Servos	Amazon	$12.88	2	$25.76
Micro Servos	Amazon	$13.69	1	$13.69
Drive Wheels				$0.00
10 Ultrasonic sensors	Amazon	$12.00	1	$12.00
Squeegee Stock (10')	McMaster	$8.80	1	$8.80
Batteries	Amazon	$39.12	4	$156.48
Battery Connectors	Amazon	$6.03	1	$6.03
Waterproof Case	Amazon	$17.82	1	$17.82
Data Radios	Amazon	$35.98	1	$35.98
Microcontroller	Digikey	$24.00	2	$48.00
SD Card Adapter	Adafruit	$14.95	1	$14.95
Small Stepper Motor	Pololu	$17.95	2	$35.90
Large Stepper Motor	Pololu	$49.95	2	$99.90
Stepper Motor Driver	Pololu	$8.95	3	$26.85
Sponge	Amazon	$5.66	6	$33.96
Flipper Motor	Servo City	$49.95	2	$99.90
			Total	$699.75

7 Team Member Responsibilities

	Trevor Decker	Stephanie Chen	Ian Rosado	Ian Hartwig
Website				
Mechanical Design		Significant Contribution	Significant Contribution	
Electrical Design	Passing Contribution			Significant Contribution
Fabrication	Passing Contribution	Significant	Significant	Passing Contribution

		Contribution	Contribution	
Software Design	Significant Contribution			Passing Contribution

8 Risk Management

8.1 Falling Off the Window

With our plan to grab or clamp on to the window frame, we are much less susceptible to losing grip and falling off of the window. But, it is still a concern. We must be careful that the gripper grabs with enough force to support our weight from sliding downwards or from torquing the gripper off the window. This has been an important factor in deciding to use a telescoping arm rather than jointed arm (to reduce the torque on this joint due to arm segments far from the window surface) and in pivoting the arm in a plane parallel to the window surface (to prevent having a 3 foot level arm hanging on the gripper friction). We feel that these risks have been adequately mitigated.

Another concern is that in the process of moving the slide up and down the frame we will drive slightly away from the surface until we fall off the frame. We think the risk of this happening is rather low, but we have not implemented any risk mitigation. Should it be a problem, we have a few options. We could modify our gripper to slightly angle one or more sets of wheels so that they will drive into the frame by a few degrees. We can also implement detection of how far we are from the window surface so that we can reset the grippers periodically when we pull too far away.

8.2 Water Resistance of Electronics

Since we are working with a robot that must operate outside it needs to be designed to handle an uncontrollable environment which may or may not include rain,snow, high winds. Moreover the window is being cleaned with water. For these reasons we need to design all of our electronics to be sealed and protected. To make our system water resistant we plan to seal all of our electrical wires in a watertight tube. The electronics will be placed in a sealed waterproof box. To avoid rust we will look into coating every metal component in our robot with a rust resistant slushing oil. Moreover we will use as many plastic materials as possible to remove the possibility of rust, moreover this will reduce the weight of the robot [4].

8.3 Gravity Fed Cleaning Pressure

At the moment, the method proposed for applying water and cleaning solution to the window is through a simple water tank mounted above the sponge/scrubber with small holes which will allow the water and cleaning solution to be gravity fed to the sponge at a relatively constant rate. A potential issue with this design is that it is not possible to adjust the flow rate of the solution into the sponge during operation, so a poorly calibrated system may leave the sponge too dry, or use up the water too quickly.

8.4 Drive Motor Synchronization

Currently we plan to use two drive modules which are separated by an arm which can reach up to 6 ft. If one of the drive modules drives slightly faster than the other, the arm will jam and the system will not be able to move. To avoid this we will be using stepper motors for our drive motors. If this problem continues to persist we can add encoders to sense the speed of each wheel and regulate each drive's speed with a PID controller.

We might also have problems with wheel slippage. To avoid this problem we will be observing the position of each drive module directly with an ultrasonic sensor. This way even if we have wheel slippage we will have an accurate estimate of the robots position on the window.

9 References

1. http://www.nytimes.com/interactive/2013/11/21/technology/how-a-winbot-730-robotic-window-cleaner-works.html?_r=1&

2. http://www.st.com/web/catalog/tools/FM116/SC959/SS1532/LN1848/PF259090

3. http://www.st.com/web/catalog/tools/FM116/SC959/SS1532/LN1848/PF259724

4. http://www.howtoprevent.com/misc/how-to-prevent-rust/