RAFIK HARIRI UNIVERSITY

Climbing Robot

Done by

NADER ALAYAN

ALI AJEENA

SAMI HOJAIRY

MOHAMMAD OMAR

Submitted to

DR. HASSAN HARIRI

This Senior Project is submitted in Partial Fulfillment of the Requirements of the BE Degree of Mechatronics and Mechanical Engineering Major of the College of Engineering at Rafik Hariri University

MECHREF-LEBANON

May 2019

Nader Alayan (2014-0205)

Ali Ajeena (2015-0244)

Sami Hojairy (2014-0251)

Mohammad Omar (2013-6019)

ACKNOWLEDGMENT

As Stephen King once said, “Books are a uniquely portable magic.” With these simple yet inspiring words, we devote this book to our guide on this journey Dr. Hassan Hariri, for all his support and advice, and for all the information he provided us and motivation to pick up. Likewise, we would like to show our deepest appreciation to all educators and staff at Rafik Hariri University who contributed and advised in this project.

In this small clause, we would also like to express our deepest gratitude to exceptional people who indirectly contributed in our venture. Much gratitude goes to Dr. NadimDiab and Dr. Bassam Moslem for being our juries and for all they beneficial recommendations and comments.

Last but not least, we would like to dedicate this book and all our prosperity to our devoted guardians who were of great support. Our guardians, who gave us their all and never asked for anything in return, who raised us to be what we are today.

ABSTRACT

This book describes the entire procedure and execution of a Climbing Robot. The Climbing Robot is proposed to function as a surveillance product. Moreover,this book demonstrates the entire procedure of building up the Climbing Robot (Design, Modeling and simulation) with its surveillance tools including outline, mechanical and electrical parts, figuring, and codes, and a full depiction of the arrangement of equipment utilized as a part of the robot.

The surveillance climbing robot has several advantages over other autonomies surveillance means such as quad copter.It can stand in field for long time, its power efficient beside of its ability to reach difficult points and its ability to carry surveillance equipment’s.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

LIST OF TABLES

Table Page

TIME PLAN

	Task	Deadline	Status
Fall 2018	Literature Review	Week 8	Completed
	Component Search	Week 10	Completed
	Components Purchase	Week 12	Completed
	BE1 Report	Week 14	Completed
	BE1 Presentation	Week 15	Completed
Spring 2019	Components Arrival	Week 3	Completed
	Test Electrical Components	Week 4	Completed
	Components Connection	Week 6	Completed
	3D Print Robot Chassis and Whegs	Week 6	Completed
	Demonstration and Progress Report	Week 7	Completed
	PowerPoint Presentation	Week 8	Completed
	Robot Assembly	Week 10	Completed
	Project Testing and Modifying	Week 13	Completed
	BE2 Report	Week 14	Completed
	BE2 Presentation	Week 15	Completed

Table 1: Time Plan

CHAPTER 1

INTRODUCTION

1.1 Background Information

In the last three decades the need for surveillance vehicle rise in order to fulfill many application such as spying, topographic mapping and indoor mapping. Many engineers tended to use drones to do this task specifically quad copters. Quad copters shown in figures 1.1 and 1.2 are unmanned vehicles which can be controlled by remote control from a distance or autonomously without the human interference. They come with different sizes and they are available commercially in acceptable prices. They can reach almost any point, even when carrying equipment's, in addition to that, they are fast and the small sized ones can be used in indoor applications. However, the usage of electric motors for their propellant driving consumes a lot of electric power in order to move and rotate. Even when maintaining a fixed position the motors consumes lot of power which makes them perform for short time. On the other hand, a robot which can climb surfaces using adhesive materials can reach any point and will perfectly operate for surveillance tasks. A climbing robot will consume less electric power compared to quad copters when operating, while on the other hand, the robot will not consume any electric power when maintaining a fixed position due to its adhesion. This special feature makes the climbing robot more efficient than the quad copter and more desirable for surveillance tasks. In addition to that, the climbing robot will totally operate in a silent mode, thus, giving it a great advantage while doing its task (surveillance; more clearly spying) in contrast to the quad copter.

Figure 1.1: Commercial and smal size Quadcopters.

1.2 Motivation

In unmanned vehicles there is a persistent need for it to reach difficult places, last long in field of operation, carry different equipment's and to be efficient enough in power consumption. Therefore we need to build a vehicle that combines all of what was mentioned above.

1.3 Literature Review

This report introduces the design, modeling and the experimental verification of the climbing robot. In addition to that, the components, codes and standards, constraints and requirements are also introduced. The robot is expected to be useful and beneficial in monitoring and photography, in other words "Surveillance".

Currently there are many researches have been devoted to wall climbing robots, and as a fact numerous types of climbing robots have been suggested. However, there was two technical issues on climbing robots which are: adhesion and climbing mechanisms.

With respect to the adhesion, the climbing robots can use the following adhesions: magnetic, electric, gripping, suction cups adhesions or dry adhesion. Each adhesion type has it disadvantages.

For example:

- Magnetic adhesions are strong but they only work on ferrous surfaces and it’s heavy.
- Electro-adhesive adhesion requires high voltage to generate strong adhesion.
- Gripping adhesion requires the robot to look for randomly located handholds.
- Suction cups adhesion requires the use of compressed air making it noisy.
- Dry adhesion is lightweight and can work on multiple smooth surfaces, it is passive and operationally quiet, making it suitable for our robot surveillance task.

The types of adhesives used in climbing robots is shown in Figure 1.2 below:

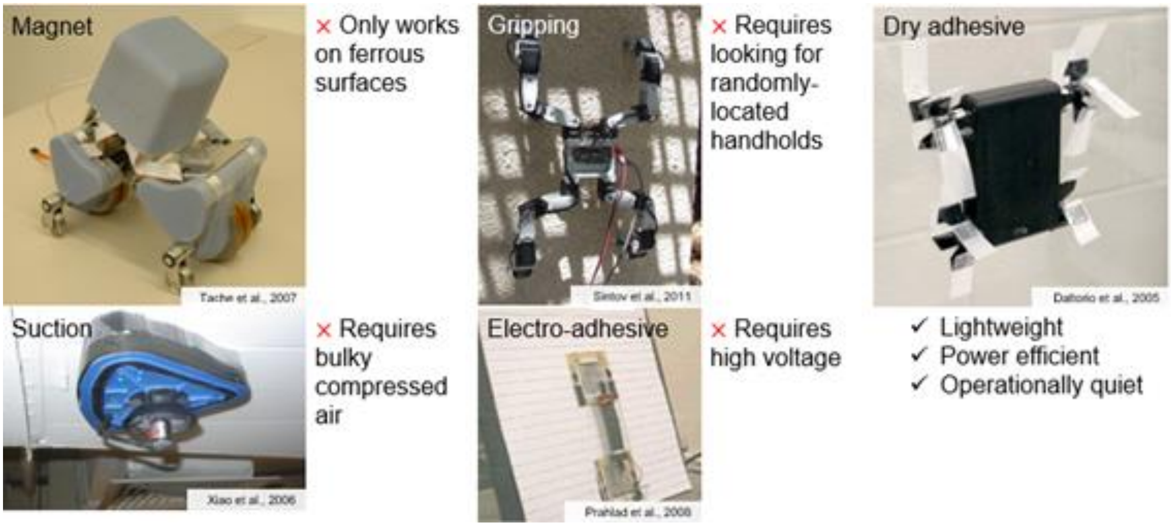


Figure 1.2: Adhesion Types.

Going to the climbing mechanism, several mechanisms have been used with climbing robots such as:

- Track based mechanism
- Legged mechanism
- Wheel-leg (wheg) mechanism.

The track based mechanism is too bulky and the legged mechanism is too complex with too many actuators which is not ideal for our power efficient tasks. However wheg mechanism have the needed advantages due to its simplicity and ability to make internal transitions.

The climbing mechanisms are shown in Figure 1.3 below:

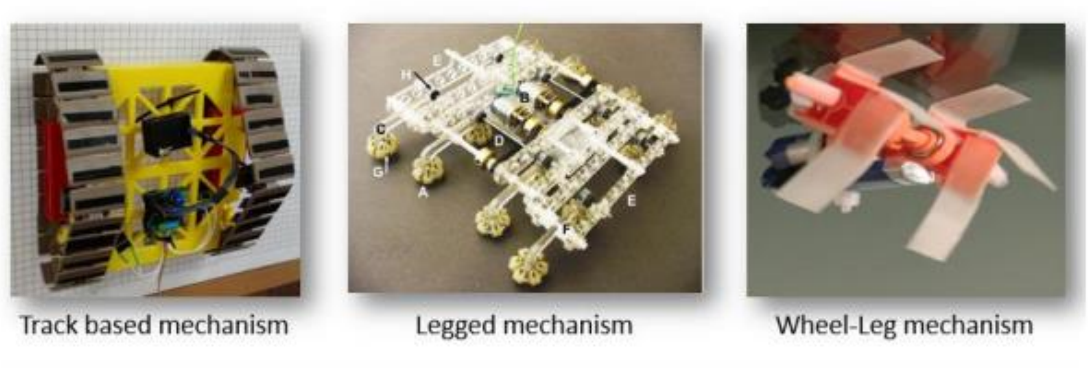


Figure 1.3: Climbing Mechanisms.

For our climbing robot we chose a combination of adhesive and climbing mechanism which makes the robot most efficient in power consumption. Therefore, we chose the wheg as climbing robot mechanism and the dry adhesive for attachment task.

Several successful climbing robots have been developed recently. Each one uses certain climbing mechanism and different type of adhesive. They are designed to carry different loads of equipment’s needed to accomplish their tasks. In our project the approach is to focus on power efficiency when designing our robot. Recent climbing robot which uses the mechanism and adhesion we chose previously (Wheg and Dry adhesive) is “ORION Robot” that was designed and developed by: Dr. Hassan Hariri and his team.

In ORION I [1] the robot used the wheg mechanism and the PDMS as a dry adhesive as shown in Figure 1.5 below:

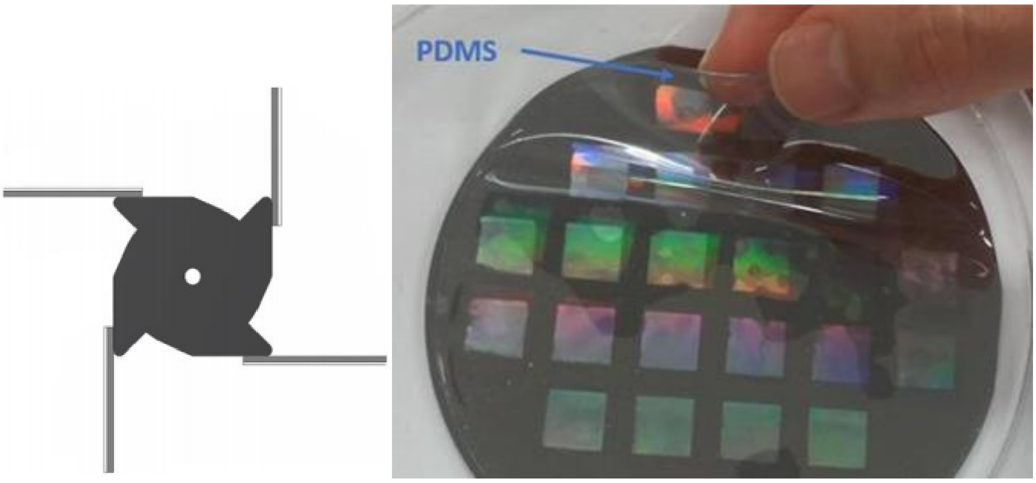


Figure 1.5: Wheg mechanism and PDMS dry adhesive.

The robot has two whegs connected to its body. The whegs provides the robot with the ability to move and to climb in all angles.ORION I uses the dry adhesive as an attachment means which is called PDMS.

The PDMS or Polydimethylsiloxane belongs to a group of polymeric organosiliconcompounds that are commonly referred to as silicones. PDMS is the most widely used silicon-based organic polymer, and is particularly known for its unusual rheological (or flow) properties. PDMS is optically clear, and, in general, inert, non-toxic, and non-flammable.This dry adhesive (PDMS) provide enough adhesion force to attach the robot on acrylic surfaces at all angles (except that its normal force is weak).

In ORION II Dr. Hassan Hariri and his team used almost the same design of robot wheg mechanism and chassis and PDMS as a dry adhesive but they added a micro suction tape. As a result the distribution of the force on the micro suction will eventually improve the performance of the climbing process uniformly.

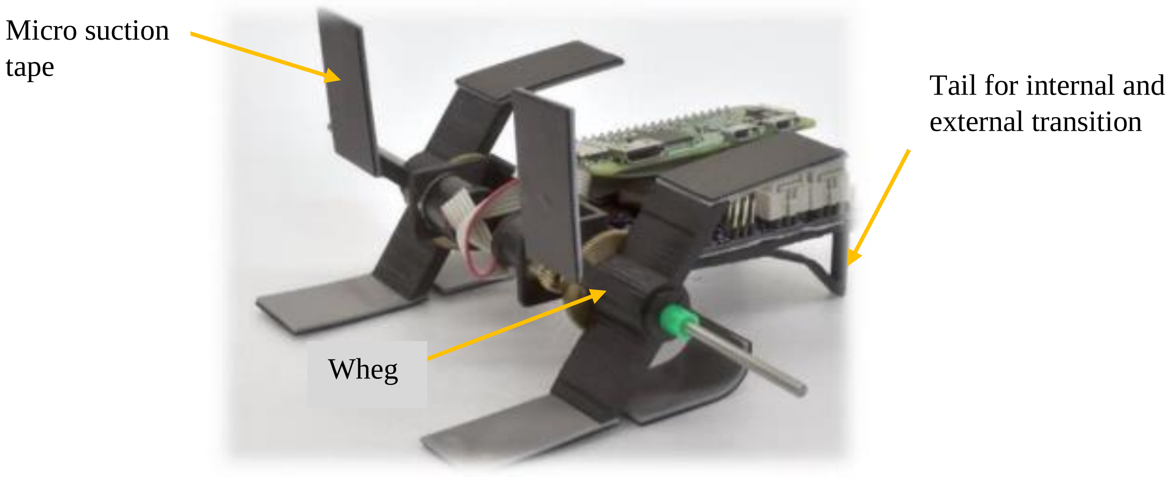


Figure 1.8: ORION Final Prototype.

The adhesion and climbing mechanism of our robot are the micro suction tape and the wheg mechanism that are adopted from “ORION Robot” that was designed by Dr. Hassan Hariri and his team.

1.5Project Aim

This project aims to model a climbing robot that will serve as a surveillance robot that would help in capturing and recording scenes. The robot will carry a camera and an SD card module that will enable it to record and save recorded videos and captured photos.

CHAPTER 2

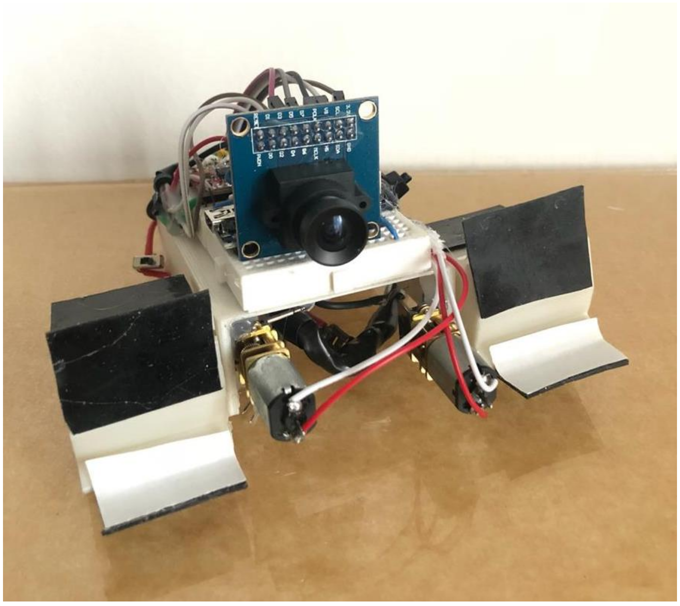
PROJECT OVERVIEW

2.1 Project Description

This project is a long term project and is divided into two phases. The first phase of the project will deal with designing and modeling the climbing robot, whereas the second phase will be dealing with the experimental verification of the robot.

This final year project will address both phases of the project, where all the project aspects from the design, model, simulation, experimental verification and results obtained are discussed in this book.

Firstly, the design and modeling of the climbing robot will be and secondly, the experimental verification of the robot and its assembly will be followed up in this final year project. Briefly the chassis and the robot whegs will be 3D printed and made from 15 % plastic and 85 % air to provide low weight. Two motors will be needed to provide the necessary force for the adhesion to stick on the surface. These motors will be controlled by the Arduino Nano Microcontroller.



CHAPTER 3

MODELING AND DESIGN

In this chapter, we dig into the details of two major parts:

- Requirements and Constraints
- Modeling and Design

3.1 Requirements and Constraints

In order to reach our desired goal which is represented by building our robot and making it able to hold the operation tools and to perform in its field (surveillance tasks), we had limitations by some requirements and constraints.

3.1.1 Requirements

- The robot must climb acrylic surfaces at any angle.
- The robot will must carry a camera that will be saving recorded videos and captured photos.
- The robot weight must not exceed 150 g.

3.1.2Constraints

- Availability of low weight battery in Lebanon.

3.2Modeling of the Robot

Before implementing the robot in hardware we need to check if our calculation results and our components are valid or not and whether the robot could be optimized or not.

In our project we used MATLAB Simscape in order to build the robot model. Simscape™ enables you to rapidly create models of physical systems within the Simulink® environment. With Simscape, you build physical component models based on physical connections that directly integrate with block diagrams and other modeling paradigms. You model systems such as electric motors, bridge rectifiers, and hydraulic actuators by assembling fundamental components into a schematic. Simscape add-on products provide more complex components and analysis capabilities.

Simscape helps you develop control systems and test system-level performance. You can create custom component models using the MATLAB® based Simscape language, which enables text-based authoring of physical modeling components, domains, and libraries. You can parameterize your models using MATLAB variables and expressions, and design control systems for your physical system in Simulink. To deploy your models to other simulation environments, including hardware-in-the-loop (HIL) systems, Simscape supports C-code generation.

In our model first we build the mechanical part of the robot to check the normal and shear forces on the robot body. Our mechanical model of the robot was based on ORION climbing robot. In our case we provided our model with rotational speed source instead of the electric motor to examine whether the mechanical model is valid or not.

The mechanical model consists of an assembly part to build the robot. Starting by angular speed source which represent the output speed of the motor connected to two gear boxes which are configured to have the gear rotation taken from the motor datasheet. The two gear boxes are connected to two wheels that represent the robot Whegs.The Whegs are connected to the vehicle body to represent the body of the robot associated with the external forces acting on the system such as gravitational force, normal force, friction force and wind force. The vehicle body is configured to have mass of 150 g and the dimension (length, width) same as ORION and a center of gravity (CG) in the position of ORION CG.This approximation has done on the belief that the center of gravity will not move a lot because our design will be much similar to ORION design. The vehicle body is connected to an angle of incline which can be manipulated in order to represent different angle of incline of the surface that the robot is climbing on.

Below is the mechanical model of the climbing robot.

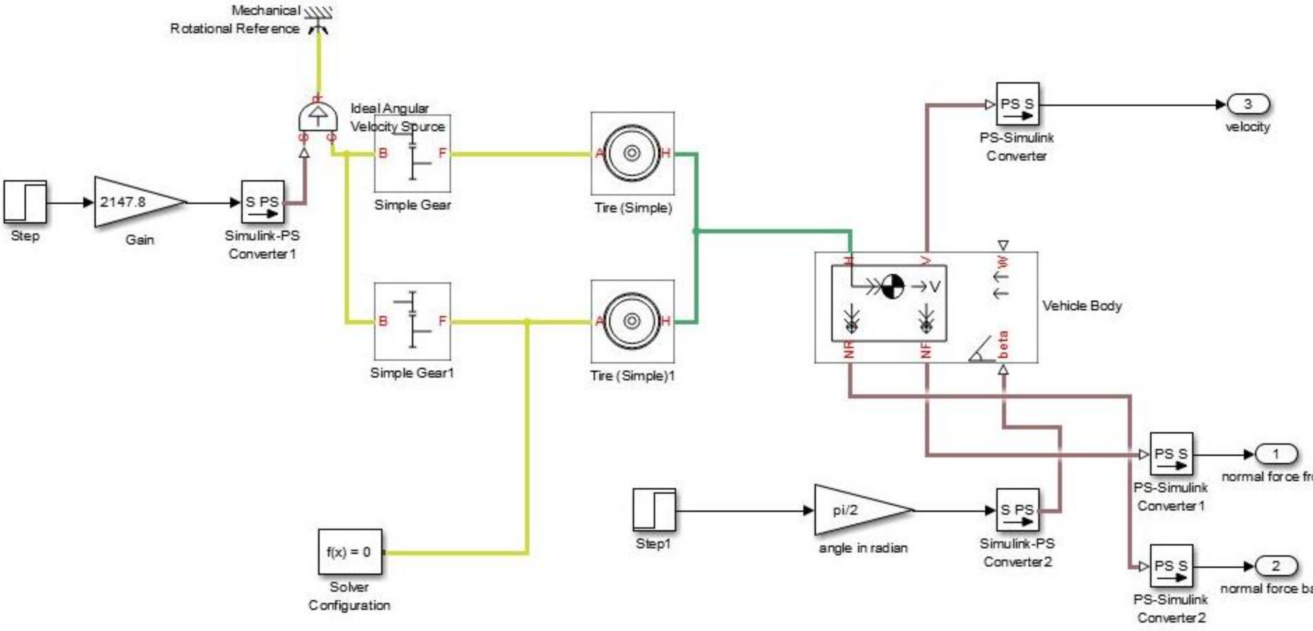


Figure 3.1: Mechanical model.

Vehicle body configurations are shown in Figure 3.2. This Figure represents the body of the robot by connecting it to the whegs it will represent all the system. It act like a free body diagram (mass, inertia and external forces acting on it) with parameters that can be varied to fit our robot conditions.

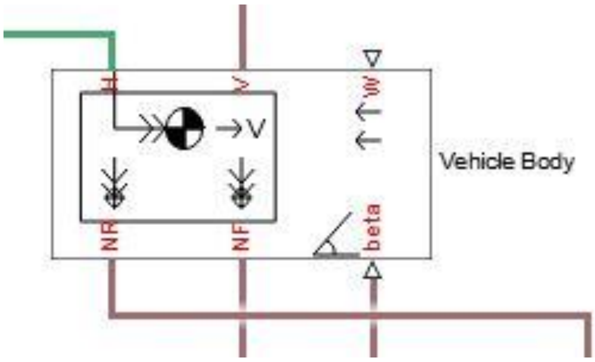
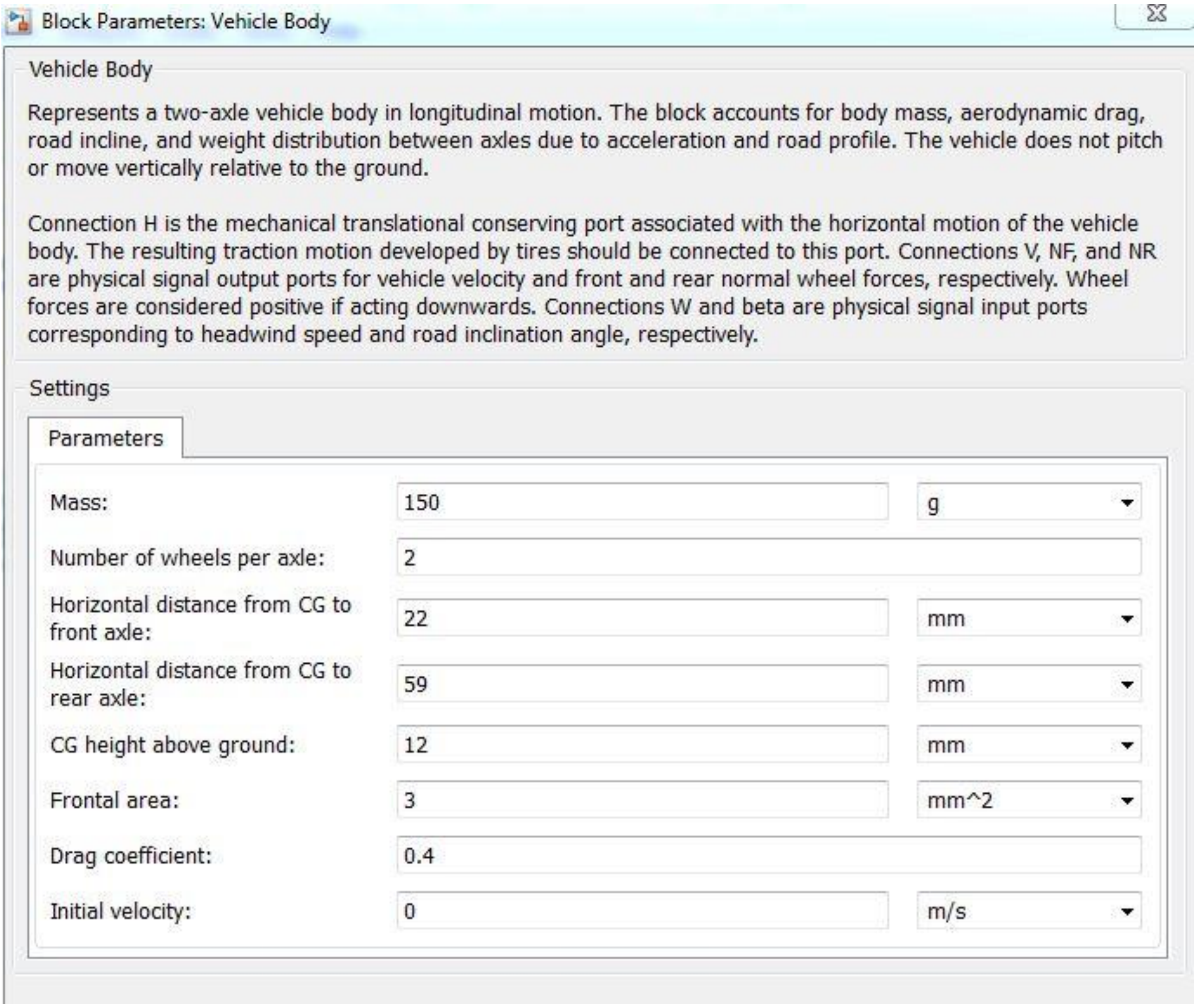


Figure 3.2: Vehicle body configurations.



Block Parameters: Vehicle Body

Vehicle Body

Represents a two-axle vehicle body in longitudinal motion. The block accounts for body mass, aerodynamic drag, road incline, and weight distribution between axles due to acceleration and road profile. The vehicle does not pitch or move vertically relative to the ground.

Connection H is the mechanical translational conserving port associated with the horizontal motion of the vehicle body. The resulting traction motion developed by tires should be connected to this port. Connections V, NF, and NR are physical signal output ports for vehicle velocity and front and rear normal wheel forces, respectively. Wheel forces are considered positive if acting downwards. Connections W and beta are physical signal input ports corresponding to headwind speed and road inclination angle, respectively.

Settings

Parameters

Mass:	150	g
Number of wheels per axle:	2	
Horizontal distance from CG to front axle:	22	mm
Horizontal distance from CG to rear axle:	59	mm
CG height above ground:	12	mm
Frontal area:	3	mm^2
Drag coefficient:	0.4	
Initial velocity:	0	m/s

Figure 3.3: Block Parameters.

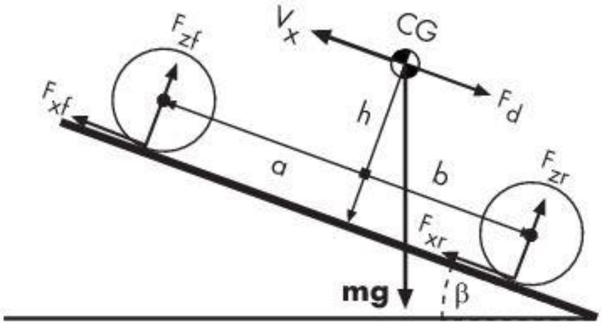


Figure 3.4: Climbing robot configurations.

3.3Normal force and Adhesion area sizing

Our robot has only two legs however we approximate our model by what is available in Simscape library which represent vehicle with four wheels.

After running the simulation several times we obtained the normal force on the front wheels at several inclination angles and we plot the graph shown in Figure 3.6.

Then we compare the data with the data obtained from ORION experiment .The comparison was in the normal force between 0 to 360 inclination angles. The ORION results are shown in Figure 3.5.

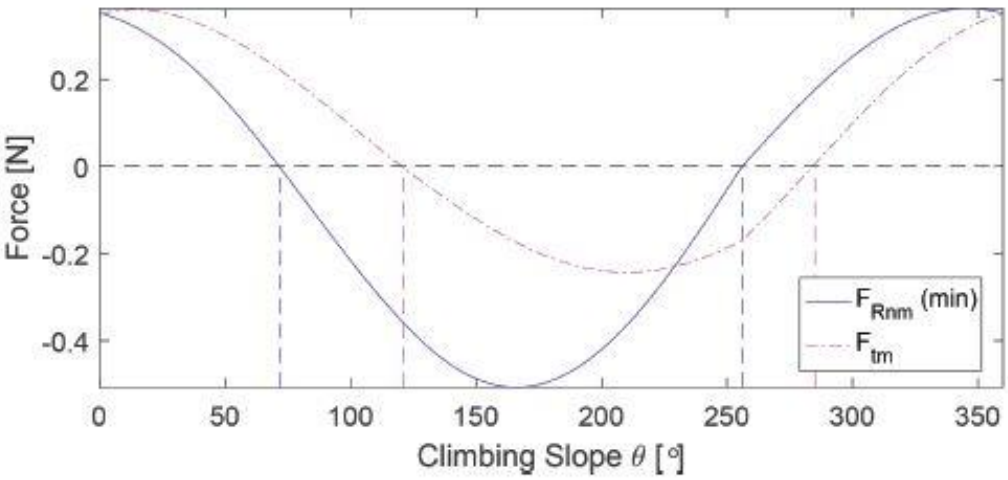


Figure 3.5: ORION Normal Force vs. Incline angle.

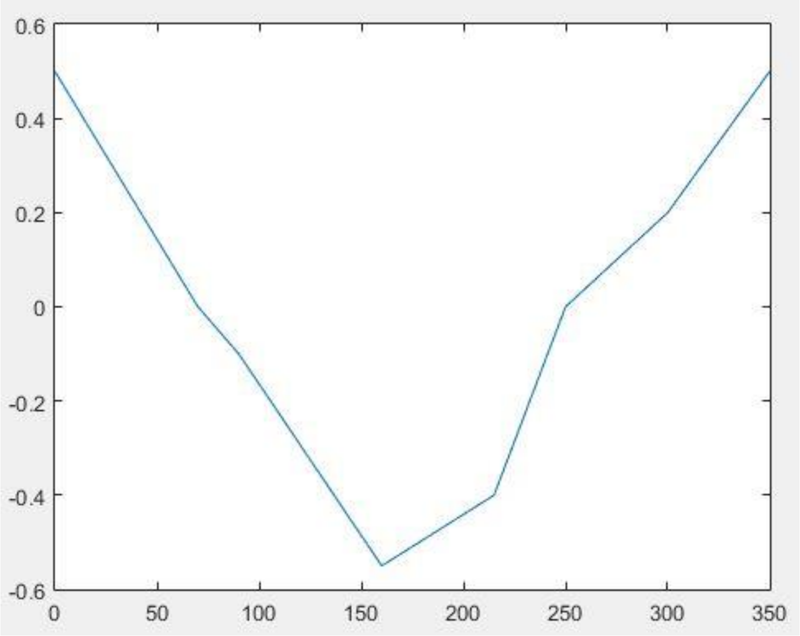


Figure 3.6: Normal Force vs. Incline angle.

The maximum normal force occurs at 180 degree which is 0.55N which means that we need 0.55N force on each wheg in order to keep the robot attached to the surface at 180 degree.

Which means that we need 0.55N force to on each wheg in order to keep the robot attached to the surface at 180 degree. As a fact, the micro suction can provide 2mN/ mm^2 and thus we can calculate the contact surface area of each wheg as follows:

$$minimum\ area = \frac{force}{adhison\ per\ area} = \frac{550mN}{2mN/mm\hat{}2} = 275mm\hat{}2$$

As engineers we decided to have a factor of safetyand to stay as the safe side, thus we chose the wheg size to be: 12.5mm in length and 35mm in width

Thus the overall area or the contact surface area of the robot with the acrylic would be: 12.5 * 35 = 412.5 mm^2.

3.4Torque and Motor selection

In order to obtain the torque needed to run our robot we run the simulation several times and we record the torque delivered to the whegs at each angle of incline. A plot of Torque vs. the incline angle is shown in Figure 3.7.

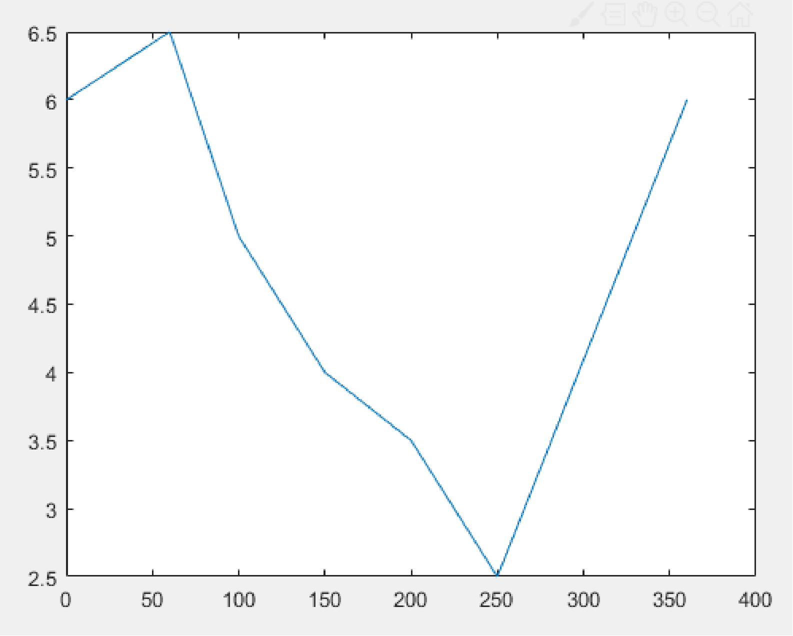


Figure 3.7: Torque vs. Incline angle.

We compare our result with the result of OROIN shown in Figure 3.8. We can see that they are approximately the same so our model is verified .The maximum torque from Figure 3.7 is 6.5mN.m so we need motor that can provide this amount of torque for each wheg.

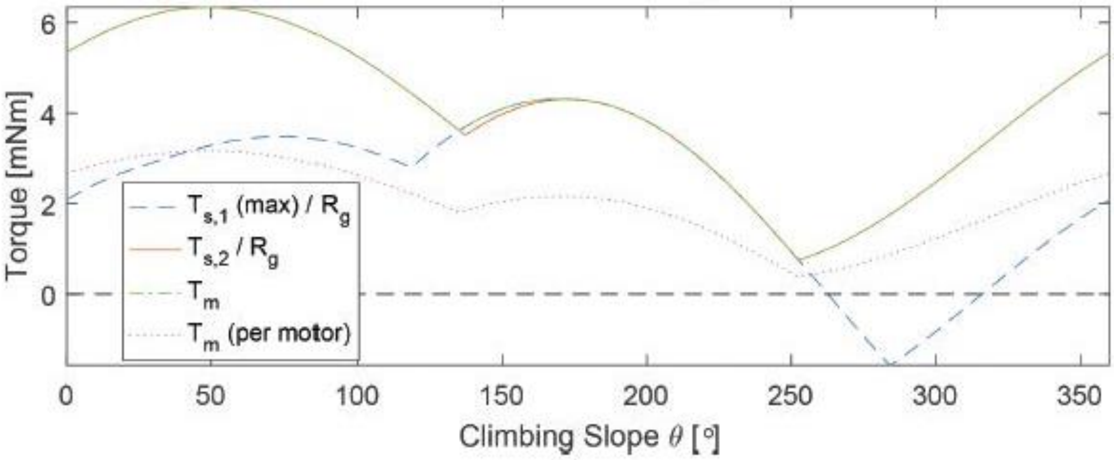
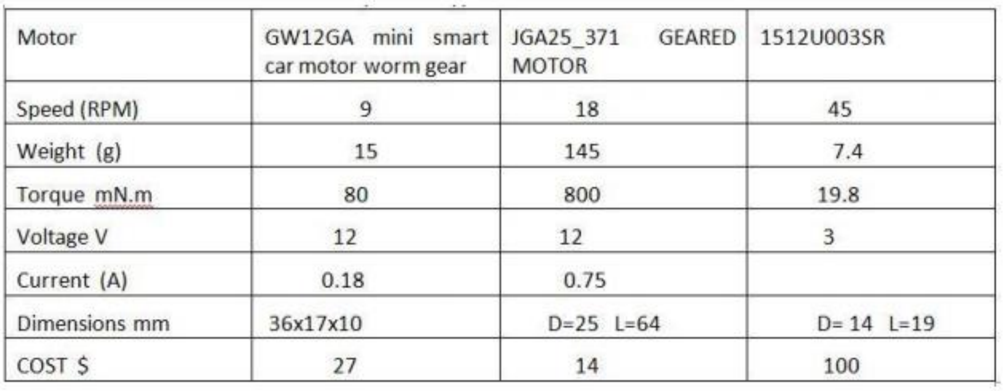


Figure 3.8: ORION Torque results.

The Figure below shows several Dc motors which we had to select one of them which will meet our design constraints.



Motor	GW12GA mini smart car motor worm gear	JGA25_371 GEARED MOTOR	1512U003SR
Speed (RPM)	9	18	45
Weight (g)	15	145	7.4
Torque mN.m	80	800	19.8
Voltage V	12	12	3
Current (A)	0.18	0.75	
Dimensions mm	36x17x10	D=25 L=64	D= 14 L=19
COST $	27	14	100

Figure 3.9: Several DC Motors DATA.

We Choose GW12GA because of its light weight which is 15g ,high torque, high efficiency and its low price which at the same time can provide us with the required torque for the whegs.

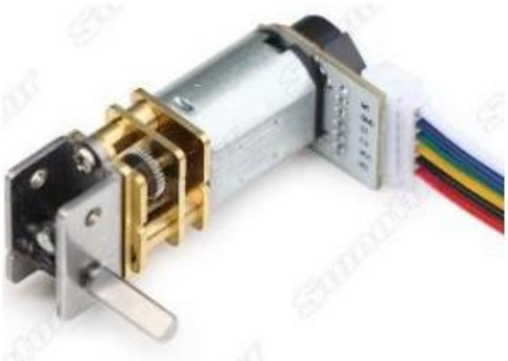


Figure 3.10: Motor selected with its Encoder.

3.5Final Prototype Dimensions

After selecting the motor needed for our robot, we have known all the other electric components needed.

- Battery
- Motor Driver
- DC to DC Boost Converter
- Arduino Nano Microcontroller

Based on the components specifications (dimensions, weight) we have known the proper dimension of the chassis and the wheg.

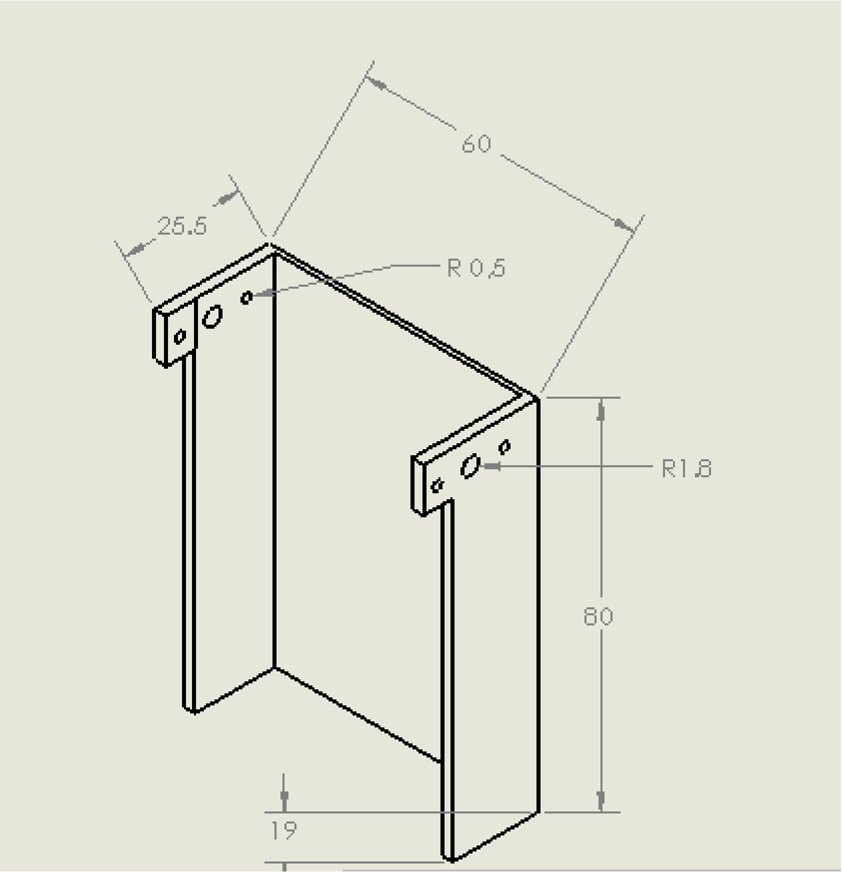


Figure 3.11: Chassis of the Robot.

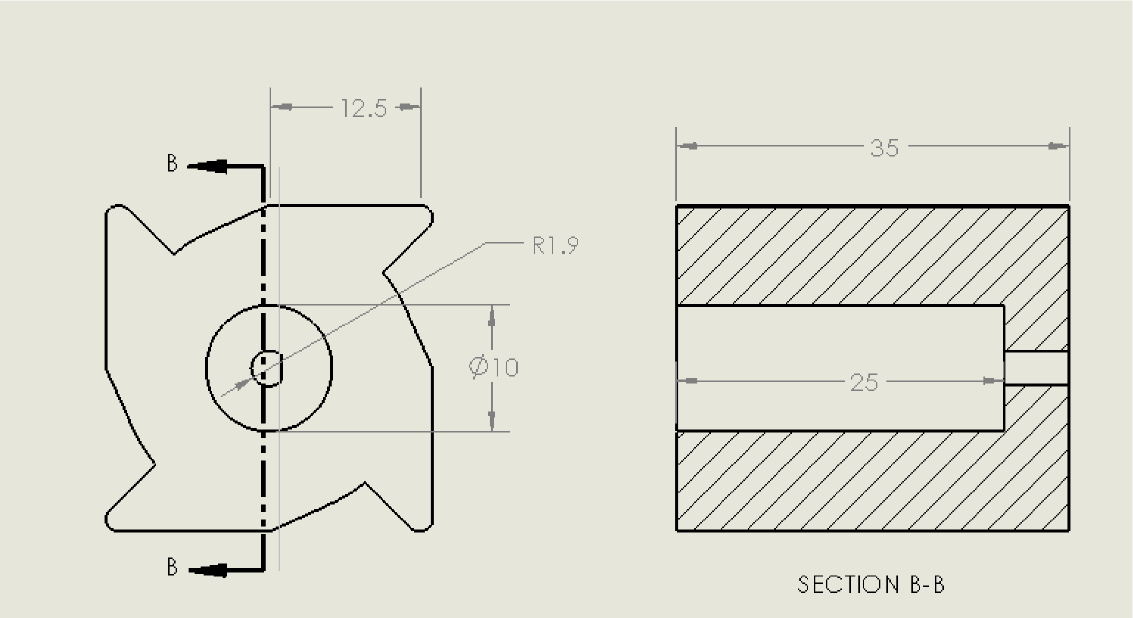


Figure 3.12: Side View of the Robot WHEG.

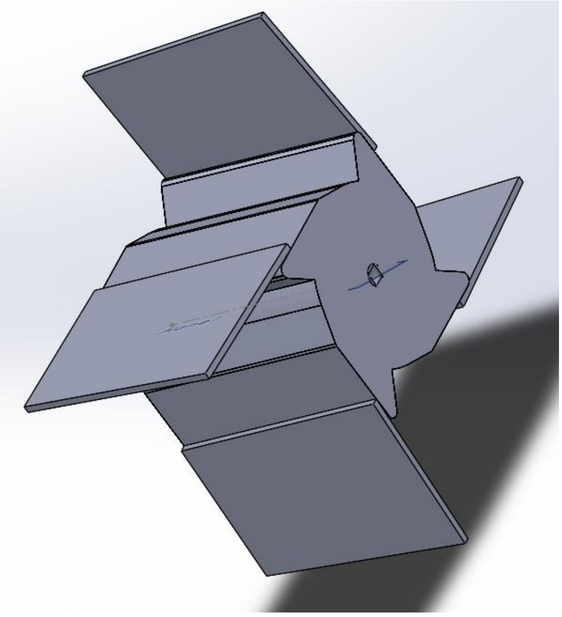


Figure 3.13: Wheg of the Robot.

3.6 Power Efficiency

Each motor draws 0.18A current when the robot is moving. The 0.18A is converted to 1.2A via the boost converter. This means that each motor will draw 0.59A (two motors = 2*0.59 = 1.2) from the battery at 3.7V. The battery is rated at 2Ah which means that it can provide 2A of current at the rated voltage which is 3.7V.

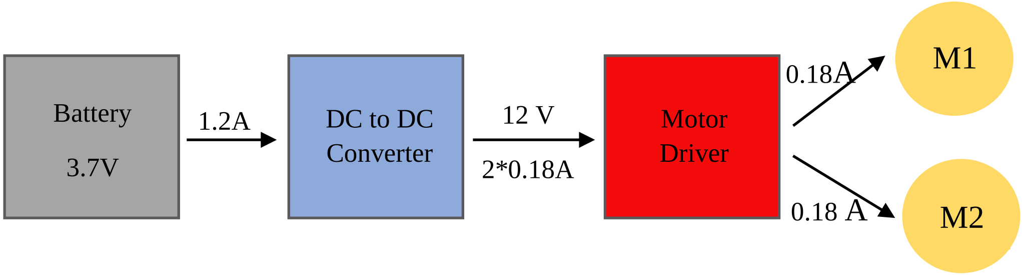
When a 0.59A current is being drawn from the battery, the battery can last for almost 100 min according to the equation shown below:

$$time = \frac{capacity\ of\ the\ baterry}{current\ drown\ from\ the\ battary}$$

$$time = \frac{2Ah}{1.2A} = \frac{2A * 60min}{1.2A} = \frac{120A * min}{1.2A} = 100min$$

This means that the robot will be able to provide a continuous motion for 100 min.

Below is a schematic of the power efficiency of the robot.



CHAPTER 4

EXPERIMENTAL VERIFICATION

This chapter shows the real prototype of our robot,trails done to test the robot angles it can climb and the overall weight of the robot.

4.1 Real Prototype

Below are several photos of the climbing robot during the assembly of it.

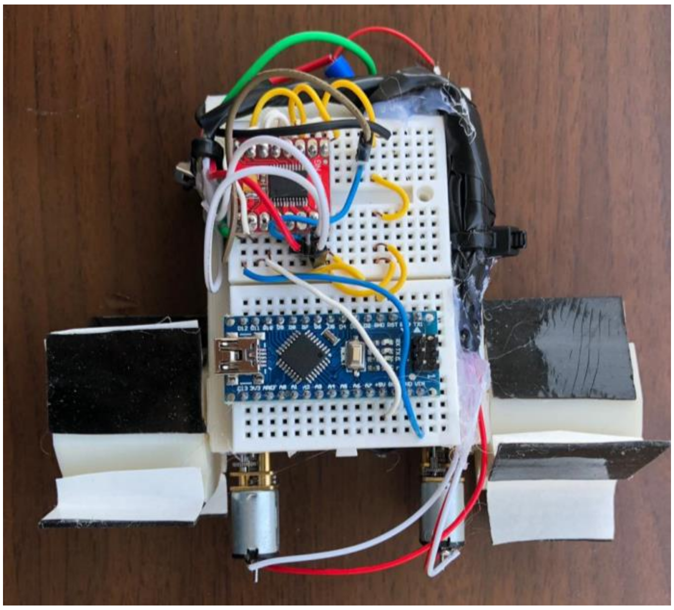


Figure 4.1: Climbing Robot without Surveillance Camera.

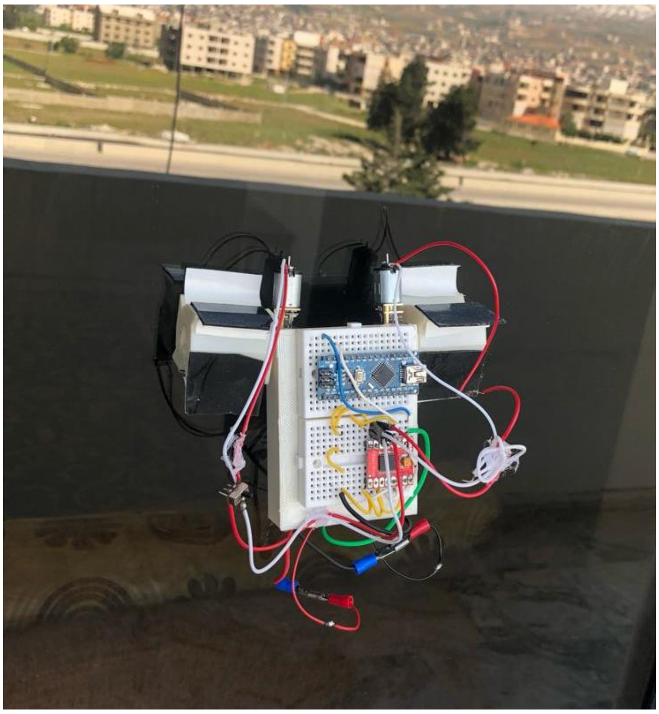


Figure 4.2: Climbing Robot view on glass.

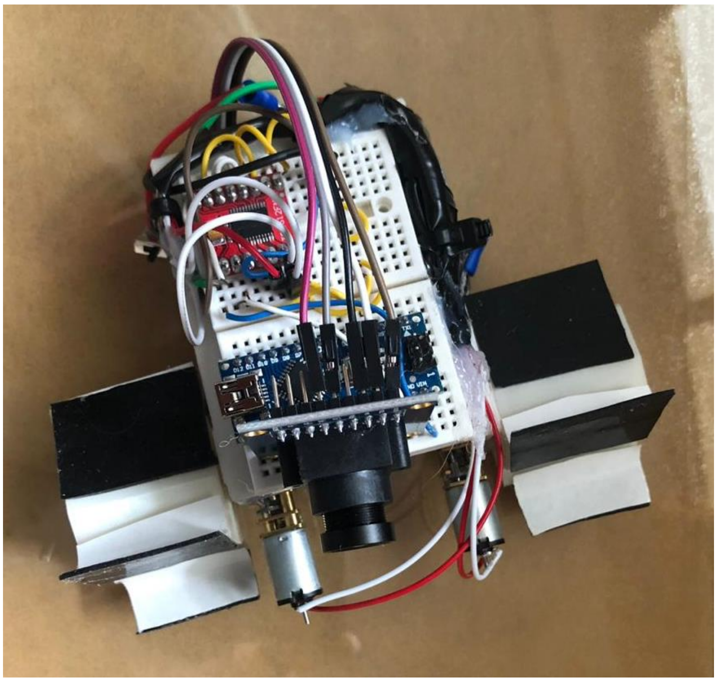


Figure 4.3: Climbing Robot Top View.

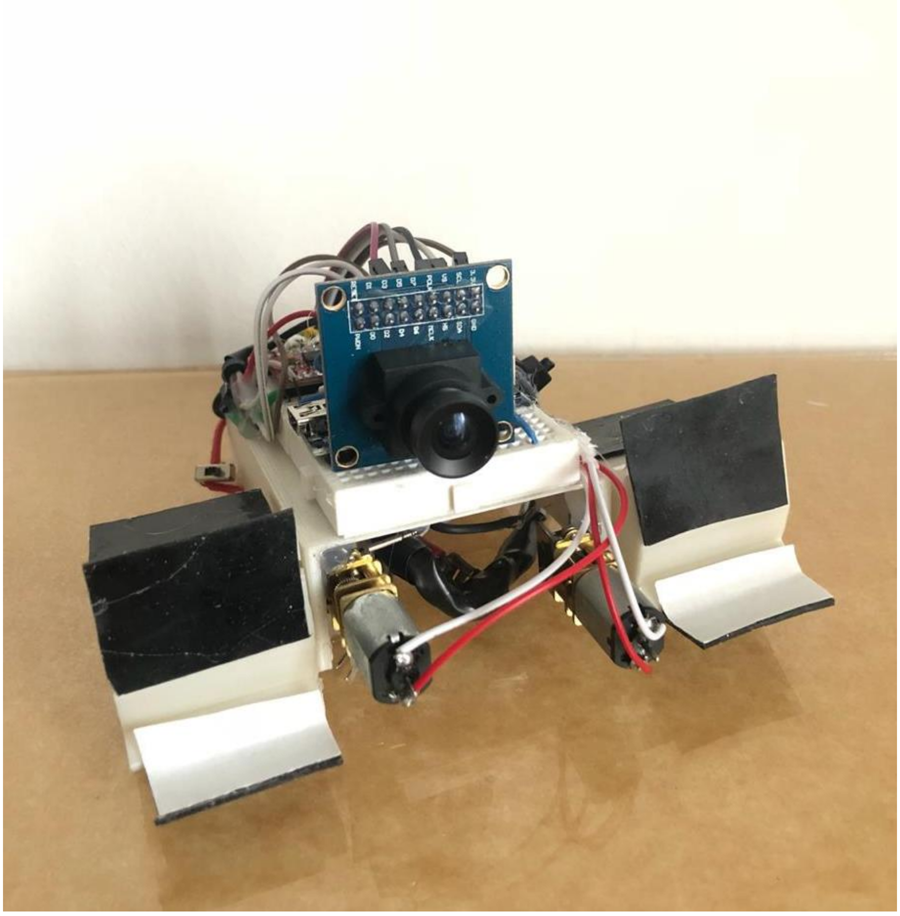


Figure 4.4: Climbing Robot Final Prototype.

4.2 Testing the Robot

After building up the robot and finalizing the assembly part, it's time to test the robot and make several trails to come up with results.

As a fact, we have tested the robot on both glass and acrylic. On glass the robot at first functioned normally on horizontal plane (0 degree) and continue to operate quietly till 89 degree. On the vertical plane (90 degree) the robot completed one revolution (complete rotation of the wheg / 4 steps forward) and failed. This failure happened due to the incorrect synchronization of the motors.

Another trail was made but this time on the acrylic, the results was much better. The robot successfully climbed the acrylic (0-85 degrees), at 90 degree the robot kept on climbing the acrylic and more than one revolution was recorded. But at last the robot fail due to the same reason which is the synchronization of the motors.

4.3 Climbing Angles

The robot climbed successfully all angles between 0 and 90degree.After that we kept on testing and surprising results was recorded.

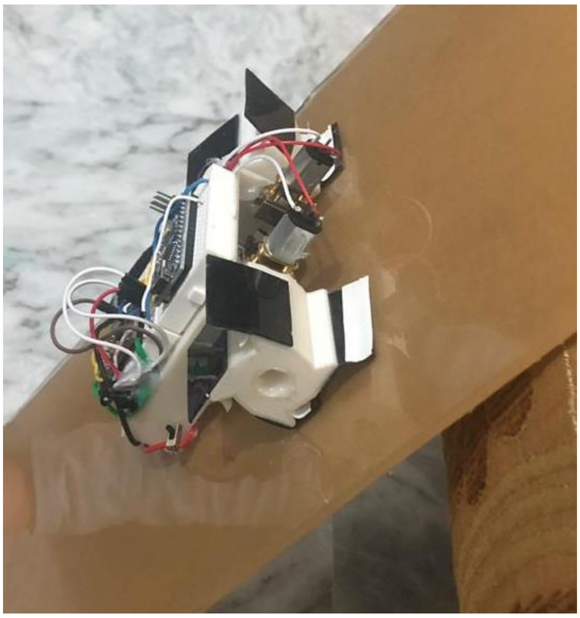


Figure 4.5: Climbing Robot at 45 degree.

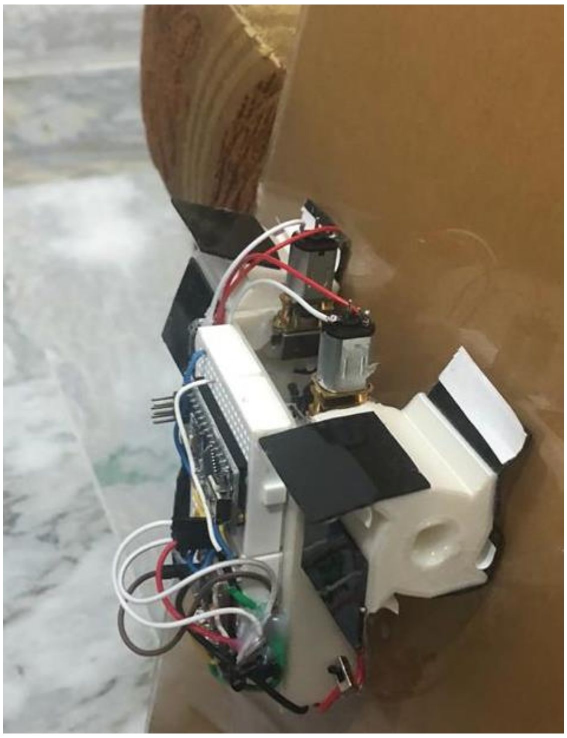


Figure 4.6: Climbing Robot at 80 degree.

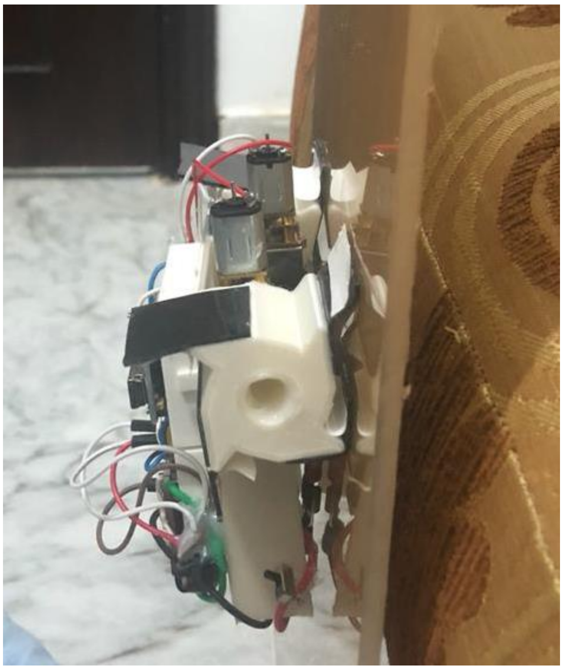


Figure 4.7: Climbing Robot at 90 degree.

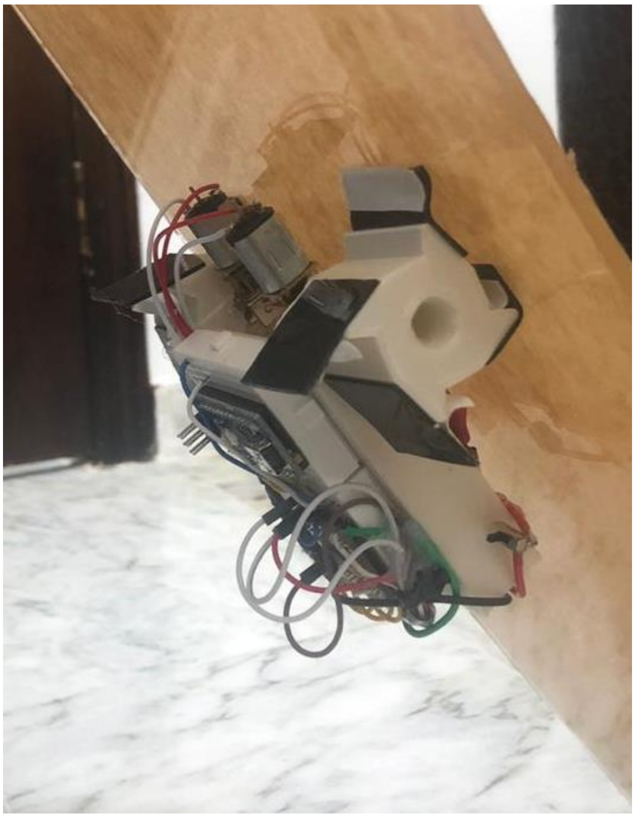


Figure 4.8: Climbing Robot at 120 degree.

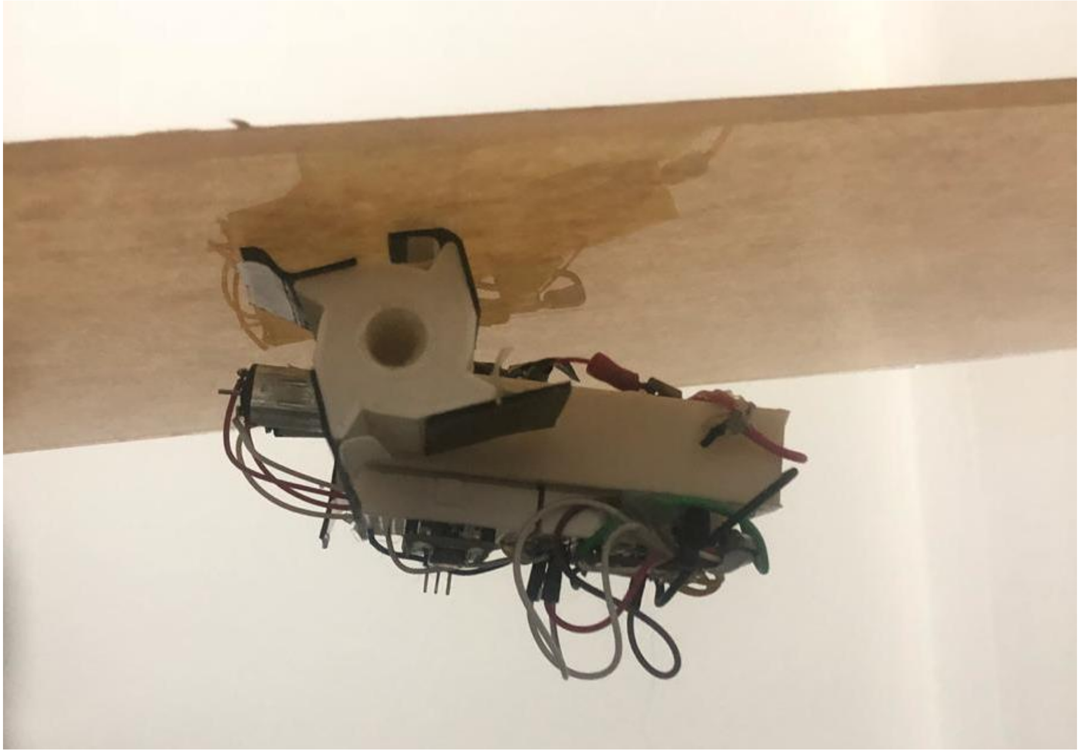


Figure 4.9: Climbing Robot at 180 degree.

After several trails it was obvious that the climbing robot was able to climb from 0 degree till 180 degree successfully without any failure.

4.4 Overall Weight

COMPONENT	**WEIGHT**
MOTOR	28 g
DC TO DC CONVERTER	4 g
BATTERY	42g
ARDUINO NANO	6 g
CAMERA	4 g
CHASSIS AND WHEGS	40 g
MOTOR DRIVER	2 g
DOUBLE SIDE AND MICRO SUCTION TAPE	4 g
BREAD BOARD AND WIRES	18 g
TOTAL	**148 g**

After summing up the overall weight of the components, it was obvious that the robot weight haven’t exceeded 150g which means that the requirement was accomplished.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

The aim of this project in its first phase was to build a functional climbing robot that is able to reach and climb specifically acrylic surfaces at any angle. By comparing our climbing robot to other autonomous quads available in the market, we can see that our robot is much cheaper and executes the same basic commands such as reaching difficult place, recording and saving data. This addresses the constraint for designing and building a relatively cheap and functional climbing robot as stated in the project aim. The robot will be used for surveillance tasks; In other words, the robot will be recording videos and capturing photos and at the same time will be saving them.

Building a climbing robot that can climb all angles is a complicated and troublesome mission. For this, and since the final year project was not only to apply what we have learned during our educational journey, our team had to dive into new fields and independently gather and apply new knowledge necessary for completing the project. For example, our team had to learn excessively how to properly model and simulate in order to get the exact values needed. Moreover, and since this was the first attempt to the team members to work on a climbing robot, we had to learn all the stuff from the very beginning including modeling and simulation, calibrating boost controller, and working on previously built robots (ORION I, ORION II, ORION III) available. Such an experience teach young engineers the value of team work and proper research.

This project is widely open for future improvements and is a solid base for future projects, in which it can be applied and used in different applications. For example, it can be used for monitoring cracks on several types of appliances, or for military applications such as spying. Moreover, this robot has a second phase which is improving its maneuverability to allow it to steer in order to reach other places were obstacles occur. Moreover, we highly recommend to use spur gears motors instead of worm gears using one shaft between the two motors, this will help in synchronizing the motors. Future SLP students are greatly encouraged to dive through this field, and to work on completing the second phase of the project.

APPENDIX A

ADDRESSING STUDENT OUTCOMES' KPIs

	How was it addressed in the SLP	**Where was it addressed in the SLP**
a. An ability to apply knowledge of mathematics, science, and engineering.		
a.1 An ability to apply knowledge of Mathematics	Using equation to calculate the battery current needed and the power efficiency of the robot	Section 3.6 Appendix C
a.2 An ability to apply knowledge of Science	Simulation of the robot dynamics	Section 3.2
a.3 An ability to apply knowledge of Engineering	CAD drawing	Section 3.5
b. An ability to design and conduct experiments, as well as to analyze and interpret data		
b.1 An ability to design experiments	Changing the robot inclination angle to prove its ability to clamp in deferent angels	Section 3.2 Section 4.3
b.2 An ability to conduct experiments	after the different simulations we was able to have normal force and motor torque data	Section 3.2 Section 3.4
b.3 An ability to analyze and interpret data	According the simulation result we sized the robot motor and whegs areas	Section 3.4 Section 3.3
c. An ability to design a system, component, or process to meet desired needs within		

realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability		
c.1 Design a system/component of a system or a process to meet specific engineering constraints	Robot assembly and components	Section 4.1
c.2 Modify a system/component of a system or a process to adhere to applicable economic, environmental, safety and sustainability constrains	We modified our choice of the battery to what we have in our country	APPENDIX C
d. An ability to function on multidisciplinary teams		
d.1 Ability to plan and organize multidisciplinary team tasks collectively	All team members cooperate to reach the project success by dividing the tasks respectively.	
d.2 Ability to carry out tasks assigned by a team	The robot was able to climb in all angles as it should be.	Section 4.3
e. An ability to identify, formulate, and solve engineering problems		
e.1 Pinpoint the existence of an engineering problem	The ability to reach any point needed in order to tack any picture needed	Section 1.1
e.2 Ability to model an engineering problem	Simulation of the robot using simmechanics	Section 3.2
e.3 Ability to justify a solution to an engineering problem	The interpretation of the simulation data.	Section 3.2 Section 3.3 Section 3.4
f. An understanding of professional and ethical responsibility		
f.1 Differentiate between ethical/unethical behaviors using applicable engineering code of ethics	Using power efficiency motors to save energy .And reference the other work (not plagiaries)	Section 3.6
f.2 Differentiate between professional and	I2C protocol for the	APPENDIX

unprofessional behaviors	camera module	G
g. An ability to communicate effectively		
g.1 Ability to write a well-structured formal report/technical document that addresses an assigned task	Submitting several technical reports, progress reports, and meetings’ minutes to the advisers and jury	Progress Report SLPI, SLPI Final Report, Progress Report SLPII, SLPII Final Report, and Minutes (Appendix C)
g.2 Ability to deliver a well-structured formal presentation that addresses an assigned task	Delivering several formal presentations throughout the SLP Project in front of the advisors and the assigned jury.	SLPI Final Presentation, SLPII Progress Presentation, and SLPII Final Presentation
h. The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context		
h.1 Identify global, economic, environmental, and societal impact of implementing engineering solutions	The robot can be applied and used in different applications related to monitoring, inspection.	CHAPTER 1
h.2 Explain global, economic, environmental, and societal impact of implementing engineering solutions	The robot can be used for surveillance application and reduce the power consumption and last long in the field	CHAPTER 1
i. A recognition of the need for, and an ability to engage in life-long learning		
i.1 Recognize the need to engage in life-long learning	Attending the workshops and Career guidance seminars organized by the MME	

	department; Recognizing the need of learning Arduino language and adhesions types	
i.2 Ability to engage in life-long learning through participation in workshops, seminars, and extracurricular activities	Using adhesion and whegs mechanisms which are not covered in any course in the MME department	
j. A knowledge of contemporary issues		
j.1 Ability to identify and discuss contemporary issues	Nowadays, autonomous surveillance machines are one of the hottest topics around	Chapter 1
k. An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice		
k.1 Identify necessary techniques, skills, tools of modern engineering practice to solve a problem at hand	In the purpose of making our robot able to climb we needed to learn different climbing mechanisms	CHAPTER 1, and CHAPTER 2 Section 3.5
k.2 Apply appropriate techniques, skills, tools of modern engineering practice to a problem at hand	use lightweight materials and electrical components for the robot	Section 4.4

APPENDIX B

BILL OF MATERIALS

The following table shows all the components purchased by our team used while implementing the project, their quantities, and their prices.

Components	**Quantity**	**Price**
3D printed Chassis	1	$0
3D printed Whegs	2	$0
Motor (DC Geared with Encoder)	3	$72.87
Motor Drive	1	$0
A4 Micro suction tape	4	$80
Double side tape	1	$2.84
Arduino Camera	2	$5
USB Module	1	$4
Arduino Nano	1	$2.78
1500 mA Battery	1	$6.5
DC to DC converter	5	$3.17
Wires + Switches + Mini Bread Board	5	$5
Acrylic	1 (65cm*20cm)	$14
Total		**$196.16**

APPENDIX C

COMPONENTS

Motor Driver

We chose TB6612FNG motor driver module which is dual bidirectional motor driver and it is based on the very popular TB6612FNG Dual H-Bridge Motor Driver IC. This module will allow us to easily and independently control two motors up to 1A each in both directions. It is ideal for robotic applications and well suited for connection to a microcontroller requiring just a couple of control lines per motor.

Figure C1: Motor Driver module.

Battery

We chose LiPo battery which can provide 3.7 Volt and 1400mAh current. It has low weight (only 28 g) which make it suitable for the climbing robot. A lithium polymer battery, or more correctly lithium-ion polymer battery (abbreviated as LiPo, LIP), is a rechargeable battery of lithium-ion technology using a polymer electrolyte instead of a liquid electrolyte. High conductivity semisolid (gel) polymers form this electrolyte.

These batteries provide higher specific energy than other lithium battery types and are used in applications where weight is a critical feature, like mobile devices and quad copters.

Unfortunately, we did not find it in Lebanon, so we bought another battery that can provide somehow same performance.

Figure C2: LiPo Battery.

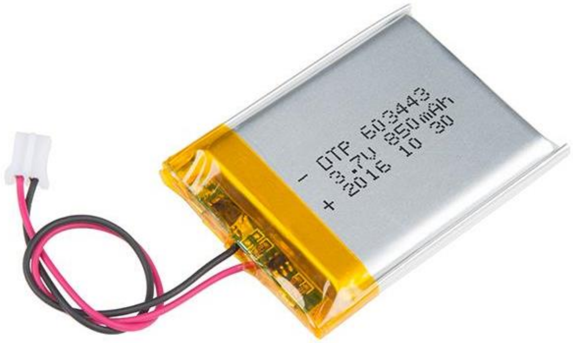


Figure C3: DTP Battery used in our Robot.

As a fact, we finally used the DTP battery that we bought from Lebanon. This battery can provide 850 mAh and is rated at 3.7 V, which almost is suitable to use for our climbing robot.

DC to DC Boost Converter

A boost converter (step-up converter) is a DC-to-DC power converter that steps up voltage (while stepping down current) from its input (supply) to its output (load). It is a class of switched-mode power supply (SMPS) containing at least two semiconductors (a diode and a transistor) and at least one energy storage element: a capacitor, inductor, or the two in combination.

In order to provide the motor with the rated voltage 12v we need to step up the battery voltage from 3.7v to 12v .we need an output current of 0.18A.

I out/ I in =V in /V out Where V in=3.7and V out =12

I out=0.18 and thus I in =0.59A.

We choose MT3608. This module features the MT3608 2 Amp step up (boost) converter, which can take input voltages as low as 2V and step up the output to as high as 28V. The MT3608 features automatic shifting to pulse frequency modulation mode at light loads. It includes under-voltage lockout, current limiting, and thermal overload protection.

This module has a multi-turn trimpot (potentiometer) that we can use to adjust the output voltage. Since the trimpot has 25 turns of adjustment we can easily adjust the output of the module to exactly the voltage we need.

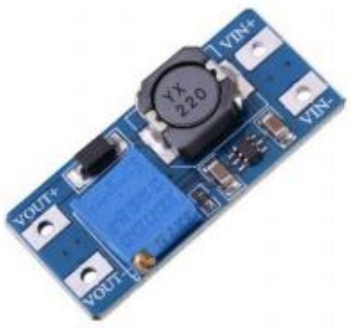


Figure C4: DC to DC Boost Converter.

Arduino Nano Microcontroller

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328P (Arduino Nano 3.x). It has more or less the same functionality of the Arduino Duemilanove, but in a different package. It works with a Mini-B USB cable instead of a standard one.

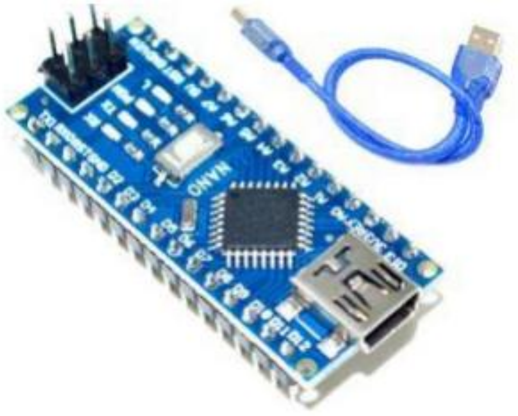


Figure C5: Arduino Nano Microcontroller with USB cable.

Double Side Tape

The tape will be placed between the surface of the wheg and the micro section tape.

Figure C5: Double Side Tape.

Micro Suction Tape

Figure C6: Micro Suction Tape.

Surveillance Camera

Firstly we have asked for a surveillance camera that is called: OV7670 CAMERA with FIFO as shown in Figure C7. Unfortunately, the camera hasn’t arrived. This camera was proposed to work via an Arduino code specialized for it in order to record and save directly to the SD card.

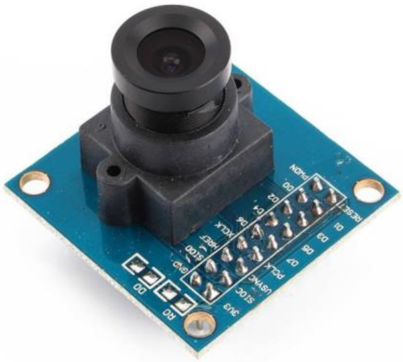


Figure C7: OV7670 Camera with FIFO.

Thus we have bought another camera from Lebanon the works without a code. This camera have its own battery and SD card. After you turn the battery on the camera will automatically start recording videos or capturing photos and send them to the SD card.

APPENDIX D

CODES AND STANDARDS

We used AWG 22 wire size because it’s suitable for prototyping application. The American wire gauge (AWG), also known as the Brown & Sharpe wire gauge, is a logarithmic stepped standardized wire gauge system used since 1857 predominantly in North America for the diameters of round, solid, nonferrous, electrically conducting wire. Dimensions of the wires are given in ASTM standard B 258. The cross-sectional area of each gauge is an important factor for determining its current-carrying capacity.

In this project we used the red wires for Vcc and black wires for GND according to IEC standards. The International Electro technical Commission-IEC; is an international standards organization that prepares and publishes International Standards for all electrical, electronic and related technologies – collectively known as "electro technology". IEC standards cover a vast range of technologies from power generation, transmission and distribution to home appliances and office equipment, semiconductors, fiber optics, batteries, solar energy, nanotechnology and marine energy as well as many others. The IEC also manages three global conformity assessment systems that certify whether equipment, system or components conform to its International Standards.

In our project we used High efficiency DC motor according to IE2 standards. The International Efficiency 60034-30-1 the following efficiency classes are defined for motors:

- IE1 (Standard Efficiency)
- IE2 (High Efficiency)
- IE3 (Premium Efficiency)
- IE4 (Super Premium Efficiency)

The Efficiency of IE 3 Motors are greater than IE2 motors.

To achieve the higher efficiency with IE3 motors, the actual design of the motors is different from your standard IE2 motor. This is mainly achieve by using more material in the IE3 motors i.e. more copper, steel etc.

APPENDIX E

MEETINGS' MINUTES

MINUTES OF MECA 595B MEETING (1)

COLLEGE OF ENGINEERING – MME Department– RHU

Group I

ON JANUARY 28th, 2019 AT 11:00AM

Present: DR. Hassan Hariri
Ali Ajeena
Nader Alayan
Sami Al Hojairy
Mohammad Omar

Absent: NONE.

The meeting came to order at 11:00 am.

First Meeting.

1. Updates

In this week, we have drawn the CAD of the wheg and chassis on SOLIDWORKS which present the mechanical parts of the robot.

2. Advisor Comments and Recommendations

DR. Hassan Hariri recommended us to use a C clip in order to fix the motor shaft with the center of the wheg to prevent it from slipping.

3. Expected Deliverables for Next Meeting

Drawing the C clip CAD

4. Assessment

NONE

This minute was taken by: Sami Hojairy

MINUTES OF MECA 595B MEETING (2)

COLLEGE OF ENGINEERING – MME Department– RHU

Group I

ON FEBRUARY 5th, 2019 AT 11:00AM

Present:	DR. Hassan Hariri Ali Ajeena Nader Alayan Sami Al Hojairy Mohammad Omar
Absent:	NONE.

The meeting came to order at 11:00 am.

Minutes of previous meeting (x) approved as corrected.

5. Updates

We Asked DR Hariri about several technical aspects of the project (climbing robot) such as the climbing technique that the robot use in order to climb, also we have drawn the C clip CAD and we start printing the parts on 3D printer.

6. Advisor Comments and Recommendations

DR. Hassan Hariri recommended us to use light weight material in order to achieve a better locomotion when using the robot.

7. Expected Deliverables for Next Meeting

PowerPoint Presentation for testing.

8. Assessment

NONE.

This Minute was taken by: Sami Hojairy

MINUTES OF MECA 595B MEETING (3)

COLLEGE OF ENGINEERING – MME Department– RHU

Group I

ON FEBRUAY 7th, 2019 AT 11:00AM

Present: DR. Hassan Hariri
Ali Ajeena
Nader Alayan
Sami Al Hojairy
Mohammad Omar

Absent: NONE.

The meeting came to order at 11:00 am.

Third Meeting.

9. Updates

In this week, we have 3D printed the chassis, left wheg and one C clip and we did presentation on the motivations and applications of the climbing robot .

Regarding the application of the climbing robot it well be used to in surveillance applications because of its ability to reach almost any point and it ability to last long on the field.

10. Advisor Comments and Recommendations

DR. Hassan Hariri recommended us to make some changes on the power point presentation and to find a suitable name for the climbing robot.

11. Expected Deliverables for Next Meeting

The right wheg and name of the robot.

12. Assessment

NONE

This minute was taken by: Mohammad Omar

MINUTES OF MECA 595B MEETING (5)

COLLEGE OF ENGINEERING – MME Department– RHU

Group I

ON FEBRUAY 21th, 2019 AT 11:00AM

Present: DR. Hassan Hariri
Ali Ajeena
Nader Alayan
Sami Al Hojairy
Mohammad Omar

Absent: NONE.

The meeting came to order at 11:00 am.

Fifth Meeting.

13. Updates

In this week, we have 3D printed the chassis and the two whegs of our robot. Plus we have bought the battery for our robot (Lithium ion battery 1700mAh).

14. Expected Deliverables for Next Meeting

Progress Report

15. Assessment

NONE.

This minute was taken by: Nader Alayan

MINUTES OF MECA 595B MEETING (6)

COLLEGE OF ENGINEERING – MME Department– RHU

Group I

ON March 7th, 2019 AT 11:00AM

Present: DR. Hassan Hariri,
Ali Ajeena
Nader Alayan
Sami Al Hojairy
Mohammad Omar

Absent: NONE.

The meeting came to order at 11:00 am.

Fifth Meeting.

16. Updates

In this week, we have bought our Lithium ion battery from Lebanon. The battery weighs 40 g thus being as a constraint for our climbing robot.

17. Advisor Comments and Recommendations

Dr. Hariri recommended us to work more on our power point presentation skills and presentation. He updated and inform us with several advice regarding that.

18. Deliverables for Next Meeting

BE. Power point Presentation in front of our juries; Dr. NadimDiab and Dr. Bassam Moslem.

19. Assessment

NONE.

This minute was taken by: Nader Alayan

MINUTES OF MECA 595B MEETING (7)

COLLEGE OF ENGINEERING – MME Department– RHU

Group I

ON March 12th, 2019 AT 11:00AM

Present: DR. Hassan Hariri
Ali Ajeena
Nader Alayan
Sami Al Hojairy
Mohammad Omar

Absent: NONE.

The meeting came to order at 11:00 am.

20. Updates

In this week, we have presented our BE power point presentation in front of our Advisor Dr. Hasan Hariri and our juries Dr. NadimDiab and Dr. Bassam Moslem.

21. Advisor Comments and Recommendations

Dr. NadimDiab said few comments regarding our presentation. He asked for:

1. Battery and overall weight of our parts.
2. Our Simulink graphs numbers
3. How much our robot can perform (Lifetime)

While, Dr. Bassam Moslem comment was he wanted something functional at last. In addition to that, Dr. Hasan Hariri (advisor) also commented on our presentation. These comments are represented by:

1. Orion gives the impression that it is our robot
2. Talk more about the 4 bar mechanism
3. Talk more about the contact surface area
4. Talk more about the robot dimensions and specifications.

22. Expected Deliverables for Next Meeting

BE report.

23. Assessment

NONE.

This minute was taken by: Nader Alayan

APPENDIX F

CAREER GUIDANCE SEMINARS

The department of Mechanical and Mechatronics Engineering at Rafik Hariri University had organized, in collaboration with the American Society of Mechanical Engineers Student Branch (ASME) at RHU a series of seminars focused on introducing students in general, and graduating students in particular, to the career life of an engineer, responsibilities to hold in an industry, and obstacles he/she might face in the work place, in an act to prepare the students to their career after graduation as engineers. Moreover, each seminar had an RHU alumna as a guest speaker, who shared with the attendees his/her experience in the workplace, and gave several advices based on his/her experience. Our team had attended several seminars of the series in an act to form a preliminary idea about the life as an engineer.

APPENDIX G

CAMERA CODE

```
//
// Source code for application to transmit image from ov7670 to PC via USB
// By SiarheiCharkes in 2015
// http://privateblog.info
//

#include <stdint.h>
#include <avr/io.h>
#include <util/twi.h>
#include <util/delay.h>
#include <avr/pgmspace.h>

#define F_CPU 16000000UL
#define vga   0
#define qvga  1
#define qqvga   2
#define yuv422  0
#define rgb565  1
#define bayerRGB  2
#define camAddr_WR  0x42
#define camAddr_RD  0x43

/* Registers */
#define REG_GAIN    0x00  /* Gain lower 8 bits (rest in vref) */
#define REG_BLUE    0x01  /* blue gain */
#define REG_RED       0x02  /* red gain */
```

```
#define REG_VREF    0x03  /* Pieces of GAIN, VSTART, VSTOP */
#define REG_COM1    0x04  /* Control 1 */
#define COM1_CCIR656  0x40    /* CCIR656 enable */

#define REG_BAVE    0x05  /* U/B Average level */
#define REG_GbAVE0x06  /* Y/Gb Average level */
#define REG_AECHH   0x07  /* AEC MS 5 bits */
#define REG_RAVE    0x08  /* V/R Average level */
#define REG_COM2    0x09  /* Control 2 */
#define COM2_SSLEEP         0x10  /* Soft sleep mode */
#define REG_PID           0x0a  /* Product ID MSB */
#define REG_VER           0x0b  /* Product ID LSB */
#define REG_COM3    0x0c  /* Control 3 */
#define COM3_SWAP         0x40  /* Byte swap */
#define COM3_SCALEEN          0x08  /* Enable scaling */
#define COM3_DCWEN          0x04  /* Enable downsamp/crop/window */
#define REG_COM4    0x0d  /* Control 4 */
#define REG_COM5    0x0e  /* All "reserved" */
#define REG_COM6    0x0f  /* Control 6 */
#define REG_AECH    0x10  /* More bits of AEC value */
#define REG_CLKRC   0x11  /* Clocl control */
#define CLK_EXT           0x40  /* Use external clock directly */
#define CLK_SCALE   0x3f  /* Mask for internal clock scale */
#define REG_COM7    0x12  /* Control 7 */ //REG mean address.
#define COM7_RESET          0x80  /* Register reset */
#define COM7_FMT_MASK         0x38
#define COM7_FMT_VGA          0x00
#define COM7_FMT_CIF          0x20  /* CIF format */
#define COM7_FMT_QVGA         0x10  /* QVGA format */
```

```
#define COM7_FMT_QCIF        0x08  /* QCIF format */
#define COM7_RGB         0x04  /* bits 0 and 2 - RGB format */
#define COM7_YUV         0x00  /* YUV */
#define COM7_BAYER         0x01  /* Bayer format */
#define COM7_PBAYER        0x05  /* "Processed bayer" */
#define REG_COM8    0x13  /* Control 8 */
#define COM8_FASTAEC         0x80  /* Enable fast AGC/AEC */
#define COM8_AECSTEP         0x40  /* Unlimited AEC step size */
#define COM8_BFILT    0x20  /* Band filter enable */
#define COM8_AGC    0x04  /* Auto gain enable */
#define COM8_AWB    0x02  /* White balance enable */
#define COM8_AEC    0x01  /* Auto exposure enable */
#define REG_COM9    0x14  /* Control 9- gain ceiling */
#define REG_COM10   0x15  /* Control 10 */
#define COM10_HSYNC        0x40  /* HSYNC instead of HREF */
#define COM10_PCLK_HB        0x20  /* Suppress PCLK on horiz blank */
#define COM10_HREF_REV         0x08  /* Reverse HREF */
#define COM10_VS_LEAD        0x04  /* VSYNC on clock leading edge */
#define COM10_VS_NEG         0x02  /* VSYNC negative */
#define COM10_HS_NEG         0x01  /* HSYNC negative */
#define REG_HSTART    0x17  /* Horiz start high bits */
#define REG_HSTOP   0x18  /* Horiz stop high bits */
#define REG_VSTART    0x19  /* Vert start high bits */
#define REG_VSTOP   0x1a  /* Vert stop high bits */
#define REG_PSHFT   0x1b  /* Pixel delay after HREF */
#define REG_MIDH    0x1c  /* Manuf. ID high */
#define REG_MIDL    0x1d  /* Manuf. ID low */
#define REG_MVFP    0x1e  /* Mirror / vflip */
#define MVFP_MIRROR        0x20  /* Mirror image */
```

```
#define MVFP_FLIP   0x10  /* Vertical flip */


#define REG_AEW           0x24  /* AGC upper limit */
#define REG_AEB           0x25    /* AGC lower limit */
#define REG_VPT           0x26  /* AGC/AEC fast mode op region */
#define REG_HSYST   0x30  /* HSYNC rising edge delay */
#define REG_HSYEN   0x31  /* HSYNC falling edge delay */
#define REG_HREF    0x32  /* HREF pieces */
#define REG_TSLB    0x3a  /* lots of stuff */
#define TSLB_YLAST    0x04  /* UYVY or VYUY - see com13 */
#define REG_COM11   0x3b  /* Control 11 */
#define COM11_NIGHT         0x80  /* NIght mode enable */
#define COM11_NMFR          0x60  /* Two bit NM frame rate */
#define COM11_HZAUTO          0x10  /* Auto detect 50/60 Hz */
#define COM11_50HZ          0x08  /* Manual 50Hz select */
#define COM11_EXP   0x02
#define REG_COM12   0x3c  /* Control 12 */
#define COM12_HREF          0x80  /* HREF always */
#define REG_COM13   0x3d  /* Control 13 */
#define COM13_GAMMA         0x80  /* Gamma enable */
#define COM13_UVSAT         0x40  /* UV saturation auto adjustment */
#define COM13_UVSWAP          0x01  /* V before U - w/TSLB */
#define REG_COM14   0x3e  /* Control 14 */
#define COM14_DCWEN         0x10  /* DCW/PCLK-scale enable */
#define REG_EDGE    0x3f  /* Edge enhancement factor */
#define REG_COM15   0x40  /* Control 15 */
#define COM15_R10F0         0x00  /* Data range 10 to F0 */
#define COM15_R01FE         0x80  /*      01 to FE */
#define COM15_R00FF         0xc0  /*      00 to FF */
```

```
#define COM15_RGB565          0x10  /* RGB565 output */
#define COM15_RGB555          0x30  /* RGB555 output */
#define REG_COM16   0x41  /* Control 16 */
#define COM16_AWBGAIN         0x08  /* AWB gain enable */
#define REG_COM17   0x42  /* Control 17 */
#define COM17_AECWIN          0xc0  /* AEC window - must match COM4 */
#define COM17_CBAR          0x08  /* DSP Color bar */
/*
* This matrix defines how the colors are generated, must be
* tweaked to adjust hue and saturation.
*
* Order: v-red, v-green, v-blue, u-red, u-green, u-blue
* They are nine-bit signed quantities, with the sign bit
* stored in0x58.Sign for v-red is bit 0, and up from there.
*/
#define REG_CMATRIX_BASE  0x4f
#define CMATRIX_LEN           6
#define REG_CMATRIX_SIGN  0x58
#define REG_BRIGHT    0x55  /* Brightness */
#define REG_CONTRAS         0x56  /* Contrast control */
#define REG_GFIX    0x69  /* Fix gain control */
#define REG_REG76   0x76  /* OV's name */
#define R76_BLKPCOR         0x80  /* Black pixel correction enable */
#define R76_WHTPCOR         0x40  /* White pixel correction enable */
#define REG_RGB444          0x8c  /* RGB 444 control */
#define R444_ENABLE         0x02  /* Turn on RGB444, overrides 5x5 */
#define R444_RGBX   0x01  /* Empty nibble at end */
#define REG_HAECC1    0x9f  /* Hist AEC/AGC control 1 */
#define REG_HAECC2    0xa0  /* Hist AEC/AGC control 2 */
```

```
#define REG_BD50MAX        0xa5  /* 50hz banding step limit */
#define REG_HAECC3    0xa6  /* Hist AEC/AGC control 3 */
#define REG_HAECC4    0xa7  /* Hist AEC/AGC control 4 */
#define REG_HAECC5    0xa8  /* Hist AEC/AGC control 5 */
#define REG_HAECC6    0xa9  /* Hist AEC/AGC control 6 */
#define REG_HAECC7    0xaa  /* Hist AEC/AGC control 7 */
#define REG_BD60MAX        0xab  /* 60hz banding step limit */
#define REG_GAIN    0x00  /* Gain lower 8 bits (rest in vref) */
#define REG_BLUE    0x01  /* blue gain */
#define REG_RED          0x02  /* red gain */
#define REG_VREF    0x03  /* Pieces of GAIN, VSTART, VSTOP */
#define REG_COM1    0x04  /* Control 1 */
#define COM1_CCIR656         0x40  /* CCIR656 enable */
#define REG_BAVE    0x05  /* U/B Average level */
#define REG_GbAVE0x06  /* Y/Gb Average level */
#define REG_AECHH   0x07  /* AEC MS 5 bits */
#define REG_RAVE    0x08  /* V/R Average level */
#define REG_COM2    0x09  /* Control 2 */
#define COM2_SSLEEP        0x10  /* Soft sleep mode */
#define REG_PID          0x0a  /* Product ID MSB */
#define REG_VER          0x0b  /* Product ID LSB */
#define REG_COM3    0x0c  /* Control 3 */
#define COM3_SWAP        0x40  /* Byte swap */
#define COM3_SCALEEN         0x08  /* Enable scaling */
#define COM3_DCWEN         0x04  /* Enable downsamp/crop/window */
#define REG_COM4    0x0d  /* Control 4 */
#define REG_COM5    0x0e  /* All "reserved" */
#define REG_COM6    0x0f  /* Control 6 */
#define REG_AECH    0x10  /* More bits of AEC value */
```

```
#define REG_CLKRC   0x11  /* Clocl control */
#define CLK_EXT           0x40  /* Use external clock directly */
#define CLK_SCALE   0x3f  /* Mask for internal clock scale */
#define REG_COM7    0x12  /* Control 7 */
#define COM7_RESET          0x80  /* Register reset */
#define COM7_FMT_MASK         0x38
#define COM7_FMT_VGA          0x00
#define COM7_FMT_CIF          0x20  /* CIF format */
#define COM7_FMT_QVGA         0x10  /* QVGA format */
#define COM7_FMT_QCIF         0x08  /* QCIF format */
#define COM7_RGB    0x04  /* bits 0 and 2 - RGB format */
#define COM7_YUV    0x00  /* YUV */
#define COM7_BAYER          0x01  /* Bayer format */
#define COM7_PBAYER         0x05  /* "Processed bayer" */
#define REG_COM8    0x13  /* Control 8 */
#define COM8_FASTAEC          0x80  /* Enable fast AGC/AEC */
#define COM8_AECSTEP          0x40  /* Unlimited AEC step size */
#define COM8_BFILT    0x20  /* Band filter enable */
#define COM8_AGC    0x04  /* Auto gain enable */
#define COM8_AWB    0x02  /* White balance enable */
#define COM8_AEC    0x01  /* Auto exposure enable */
#define REG_COM9    0x14  /* Control 9- gain ceiling */
#define REG_COM10   0x15  /* Control 10 */
#define COM10_HSYNC         0x40  /* HSYNC instead of HREF */
#define COM10_PCLK_HB         0x20  /* Suppress PCLK on horiz blank */
#define COM10_HREF_REV          0x08  /* Reverse HREF */
#define COM10_VS_LEAD           0x04  /* VSYNC on clock leading edge */
#define COM10_VS_NEG          0x02  /* VSYNC negative */
#define COM10_HS_NEG          0x01  /* HSYNC negative */
```

```
#define REG_HSTART    0x17  /* Horiz start high bits */
#define REG_HSTOP   0x18  /* Horiz stop high bits */
#define REG_VSTART    0x19  /* Vert start high bits */
#define REG_VSTOP   0x1a  /* Vert stop high bits */
#define REG_PSHFT   0x1b  /* Pixel delay after HREF */
#define REG_MIDH    0x1c  /* Manuf. ID high */
#define REG_MIDL    0x1d  /* Manuf. ID low */
#define REG_MVFP    0x1e  /* Mirror / vflip */
#define MVFP_MIRROR         0x20  /* Mirror image */
#define MVFP_FLIP   0x10  /* Vertical flip */
#define REG_AEW           0x24  /* AGC upper limit */
#define REG_AEB           0x25  /* AGC lower limit */
#define REG_VPT           0x26  /* AGC/AEC fast mode op region */
#define REG_HSYST   0x30  /* HSYNC rising edge delay */
#define REG_HSYEN   0x31  /* HSYNC falling edge delay */
#define REG_HREF    0x32  /* HREF pieces */
#define REG_TSLB    0x3a  /* lots of stuff */
#define TSLB_YLAST    0x04  /* UYVY or VYUY - see com13 */
#define REG_COM11   0x3b  /* Control 11 */
#define COM11_NIGHT         0x80  /* NIght mode enable */
#define COM11_NMFR          0x60  /* Two bit NM frame rate */
#define COM11_HZAUTO          0x10  /* Auto detect 50/60 Hz */
#define COM11_50HZ          0x08  /* Manual 50Hz select */
#define COM11_EXP   0x02
#define REG_COM12   0x3c  /* Control 12 */
#define COM12_HREF          0x80  /* HREF always */
#define REG_COM13   0x3d  /* Control 13 */
#define COM13_GAMMA         0x80  /* Gamma enable */
#define COM13_UVSAT         0x40  /* UV saturation auto adjustment */
```

```
#define COM13_UVSWAP         0x01  /* V before U - w/TSLB */
#define REG_COM14   0x3e  /* Control 14 */
#define COM14_DCWEN         0x10  /* DCW/PCLK-scale enable */
#define REG_EDGE    0x3f  /* Edge enhancement factor */
#define REG_COM15   0x40  /* Control 15 */
#define COM15_R10F0         0x00  /* Data range 10 to F0 */
#define COM15_R01FE         0x80  /*      01 to FE */
#define COM15_R00FF         0xc0  /*      00 to FF */
#define COM15_RGB565         0x10  /* RGB565 output */
#define COM15_RGB555         0x30  /* RGB555 output */
#define REG_COM16   0x41  /* Control 16 */
#define COM16_AWBGAIN         0x08  /* AWB gain enable */
#define REG_COM17   0x42  /* Control 17 */
#define COM17_AECWIN         0xc0  /* AEC window - must match COM4 */
#define COM17_CBAR         0x08  /* DSP Color bar */

#define CMATRIX_LEN            6
#define REG_BRIGHT    0x55  /* Brightness */
#define REG_REG76   0x76  /* OV's name */
#define R76_BLKPCOR         0x80  /* Black pixel correction enable */
#define R76_WHTPCOR         0x40  /* White pixel correction enable */
#define REG_RGB444         0x8c  /* RGB 444 control */
#define R444_ENABLE         0x02  /* Turn on RGB444, overrides 5x5 */
#define R444_RGBX   0x01  /* Empty nibble at end */
#define REG_HAECC1    0x9f  /* Hist AEC/AGC control 1 */
#define REG_HAECC2    0xa0  /* Hist AEC/AGC control 2 */
#define REG_BD50MAX         0xa5  /* 50hz banding step limit */
#define REG_HAECC3    0xa6  /* Hist AEC/AGC control 3 */
#define REG_HAECC4    0xa7  /* Hist AEC/AGC control 4 */
```

```
#define REG_HAECC5    0xa8  /* Hist AEC/AGC control 5 */
#define REG_HAECC6    0xa9  /* Hist AEC/AGC control 6 */
#define REG_HAECC7    0xaa  /* Hist AEC/AGC control 7 */
#define REG_BD60MAX         0xab  /* 60hz banding step limit */
#define MTX1           0x4f  /* Matrix Coefficient 1 */
#define MTX2           0x50  /* Matrix Coefficient 2 */
#define MTX3           0x51  /* Matrix Coefficient 3 */
#define MTX4           0x52  /* Matrix Coefficient 4 */
#define MTX5           0x53  /* Matrix Coefficient 5 */
#define MTX6           0x54  /* Matrix Coefficient 6 */
#define REG_CONTRAS         0x56  /* Contrast control */
#define MTXS           0x58  /* Matrix Coefficient Sign */
#define AWBC7           0x59  /* AWB Control 7 */
#define AWBC8           0x5a  /* AWB Control 8 */
#define AWBC9           0x5b  /* AWB Control 9 */
#define AWBC10            0x5c  /* AWB Control 10 */
#define AWBC11            0x5d  /* AWB Control 11 */
#define AWBC12            0x5e  /* AWB Control 12 */
#define REG_GFI           0x69  /* Fix gain control */
#define GGAIN           0x6a  /* G Channel AWB Gain */
#define DBLV           0x6b
#define AWBCTR3          0x6c  /* AWB Control 3 */
#define AWBCTR2          0x6d  /* AWB Control 2 */
#define AWBCTR1          0x6e  /* AWB Control 1 */
#define AWBCTR0          0x6f  /* AWB Control 0 */

structregval_list{
  uint8_t reg_num;
uint16_t value;
```

```
};

conststructregval_list qvga_ov7670[] PROGMEM = {
{ REG_COM14, 0x19 },
{ 0x72, 0x11 },
{ 0x73, 0xf1 },

{ REG_HSTART, 0x16 },
{ REG_HSTOP, 0x04 },
{ REG_HREF, 0xa4 },
{ REG_VSTART, 0x02 },
{ REG_VSTOP, 0x7a },
{ REG_VREF, 0x0a },


/*  { REG_HSTART, 0x16 },
{ REG_HSTOP, 0x04 },
{ REG_HREF, 0x24 },
{ REG_VSTART, 0x02 },
{ REG_VSTOP, 0x7a },
{ REG_VREF, 0x0a },*/
{ 0xff, 0xff }, /* END MARKER */
};

conststructregval_list yuv422_ov7670[] PROGMEM = {
{ REG_COM7, 0x0 },  /* Selects YUV mode */
{ REG_RGB444, 0 },  /* No RGB444 please */
{ REG_COM1, 0 },
{ REG_COM15, COM15_R00FF },
{ REG_COM9, 0x6A }, /* 128x gain ceiling; 0x8 is reserved bit */
{ 0x4f, 0x80 },   /* "matrix coefficient 1" */
{ 0x50, 0x80 },   /* "matrix coefficient 2" */
{ 0x51, 0 },    /* vb */
{ 0x52, 0x22 },   /* "matrix coefficient 4" */
{ 0x53, 0x5e },   /* "matrix coefficient 5" */
{ 0x54, 0x80 },   /* "matrix coefficient 6" */
{ REG_COM13, COM13_UVSAT },
{ 0xff, 0xff },   /* END MARKER */
};

conststructregval_list ov7670_default_regs[] PROGMEM = {//from the linux driver
{ REG_COM7, COM7_RESET },
{ REG_TSLB, 0x04 }, /* OV */
{ REG_COM7, 0 },  /* VGA */
  /*
  * Set the hardware window.  These values from OV don't entirely
  * make sense - hstop is less than hstart.  But they work...
  */
{ REG_HSTART, 0x13 }, { REG_HSTOP, 0x01 },
{ REG_HREF, 0xb6 }, { REG_VSTART, 0x02 },
{ REG_VSTOP, 0x7a }, { REG_VREF, 0x0a },

{ REG_COM3, 0 }, { REG_COM14, 0 },
  /* Mystery scaling numbers */
{ 0x70, 0x3a }, { 0x71, 0x35 },
{ 0x72, 0x11 }, { 0x73, 0xf0 },
{ 0xa2,/* 0x02 changed to 1*/1 }, { REG_COM10, 0x0 },
  /* Gamma curve values */
```

```
{ 0x7a, 0x20 }, { 0x7b, 0x10 },
{ 0x7c, 0x1e }, { 0x7d, 0x35 },
{ 0x7e, 0x5a }, { 0x7f, 0x69 },
{ 0x80, 0x76 }, { 0x81, 0x80 },
{ 0x82, 0x88 }, { 0x83, 0x8f },
{ 0x84, 0x96 }, { 0x85, 0xa3 },
{ 0x86, 0xaf }, { 0x87, 0xc4 },
{ 0x88, 0xd7 }, { 0x89, 0xe8 },
  /* AGC and AEC parameters.  Note we start by disabling those features,
then turn them only after tweaking the values. */
{ REG_COM8, COM8_FASTAEC | COM8_AECSTEP },
{ REG_GAIN, 0 }, { REG_AECH, 0 },
{ REG_COM4, 0x40 }, /* magic reserved bit */
{ REG_COM9, 0x18 }, /* 4x gain + magic rsvd bit */
{ REG_BD50MAX, 0x05 }, { REG_BD60MAX, 0x07 },
{ REG_AEW, 0x95 }, { REG_AEB, 0x33 },
{ REG_VPT, 0xe3 }, { REG_HAECC1, 0x78 },
{ REG_HAECC2, 0x68 }, { 0xa1, 0x03 }, /* magic */
{ REG_HAECC3, 0xd8 }, { REG_HAECC4, 0xd8 },
{ REG_HAECC5, 0xf0 }, { REG_HAECC6, 0x90 },
{ REG_HAECC7, 0x94 },
{ REG_COM8, COM8_FASTAEC | COM8_AECSTEP | COM8_AGC | COM8_AEC
},
{ 0x30, 0 }, { 0x31, 0 },//disable some delays
  /* Almost all of these are magic "reserved" values.  */
{ REG_COM5, 0x61 }, { REG_COM6, 0x4b },
{ 0x16, 0x02 }, { REG_MVFP, 0x07 },
{ 0x21, 0x02 }, { 0x22, 0x91 },
{ 0x29, 0x07 }, { 0x33, 0x0b },
{ 0x35, 0x0b }, { 0x37, 0x1d },
{ 0x38, 0x71 }, { 0x39, 0x2a },
{ REG_COM12, 0x78 }, { 0x4d, 0x40 },
{ 0x4e, 0x20 }, { REG_GFIX, 0 },
  /*{0x6b, 0x4a},*/{ 0x74, 0x10 },
{ 0x8d, 0x4f }, { 0x8e, 0 },
{ 0x8f, 0 }, { 0x90, 0 },
{ 0x91, 0 }, { 0x96, 0 },
{ 0x9a, 0 }, { 0xb0, 0x84 },
{ 0xb1, 0x0c }, { 0xb2, 0x0e },
{ 0xb3, 0x82 }, { 0xb8, 0x0a },

  /* More reserved magic, some of which tweaks white balance */
{ 0x43, 0x0a }, { 0x44, 0xf0 },
{ 0x45, 0x34 }, { 0x46, 0x58 },
{ 0x47, 0x28 }, { 0x48, 0x3a },
{ 0x59, 0x88 }, { 0x5a, 0x88 },
{ 0x5b, 0x44 }, { 0x5c, 0x67 },
{ 0x5d, 0x49 }, { 0x5e, 0x0e },
{ 0x6c, 0x0a }, { 0x6d, 0x55 },
{ 0x6e, 0x11 }, { 0x6f, 0x9e }, /* it was 0x9F "9e for advance AWB" */
{ 0x6a, 0x40 }, { REG_BLUE, 0x40 },
{ REG_RED, 0x60 },
{ REG_COM8, COM8_FASTAEC | COM8_AECSTEP | COM8_AGC | COM8_AEC |
COM8_AWB },

  /* Matrix coefficients */
{ 0x4f, 0x80 }, { 0x50, 0x80 },
{ 0x51, 0 },    { 0x52, 0x22 },
{ 0x53, 0x5e }, { 0x54, 0x80 },
{ 0x58, 0x9e },
{ REG_COM16, COM16_AWBGAIN }, { REG_EDGE, 0 },
{ 0x75, 0x05 }, { REG_REG76, 0xe1 },
{ 0x4c, 0 },     { 0x77, 0x01 },
{ REG_COM13, /*0xc3*/0x48 }, { 0x4b, 0x09 },
{ 0xc9, 0x60 },   /*{REG_COM16, 0x38},*/
{ 0x56, 0x40 },

{ 0x34, 0x11 }, { REG_COM11, COM11_EXP | COM11_HZAUTO },
{ 0xa4, 0x82/*Was 0x88*/ }, { 0x96, 0 },
{ 0x97, 0x30 }, { 0x98, 0x20 },
{ 0x99, 0x30 }, { 0x9a, 0x84 },
{ 0x9b, 0x29 }, { 0x9c, 0x03 },
{ 0x9d, 0x4c }, { 0x9e, 0x3f },
{ 0x78, 0x04 },

  /* Extra-weird stuff.  Some sort of multiplexor register */
{ 0x79, 0x01 }, { 0xc8, 0xf0 },
{ 0x79, 0x0f }, { 0xc8, 0x00 },
{ 0x79, 0x10 }, { 0xc8, 0x7e },
{ 0x79, 0x0a }, { 0xc8, 0x80 },
{ 0x79, 0x0b }, { 0xc8, 0x01 },
{ 0x79, 0x0c }, { 0xc8, 0x0f },
{ 0x79, 0x0d }, { 0xc8, 0x20 },
{ 0x79, 0x09 }, { 0xc8, 0x80 },
{ 0x79, 0x02 }, { 0xc8, 0xc0 },
{ 0x79, 0x03 }, { 0xc8, 0x40 },
{ 0x79, 0x05 }, { 0xc8, 0x30 },
{ 0x79, 0x26 },
{ 0xff, 0xff }, /* END MARKER */
};



voiderror_led(void){
  DDRB |= 32;//make sure led is output
while (1){//wait for reset
    PORTB ^= 32;// toggle led
    _delay_ms(100);
  }
}

voidtwiStart(void){
  TWCR = _BV(TWINT) | _BV(TWSTA) | _BV(TWEN);//send start
while (!(TWCR & (1 << TWINT)));//wait for start to be transmitted
if ((TWSR & 0xF8) != TW_START)
error_led();
}

voidtwiWriteByte(uint8_t DATA, uint8_t type){
  TWDR = DATA;
  TWCR = _BV(TWINT) | _BV(TWEN);
while (!(TWCR & (1 << TWINT))) {}
if ((TWSR & 0xF8) != type)
error_led();
}

voidtwiAddr(uint8_t addr, uint8_t typeTWI){
  TWDR = addr;//send address
  TWCR = _BV(TWINT) | _BV(TWEN);    /* clear interrupt to start transmission */
while ((TWCR & _BV(TWINT)) == 0); /* wait for transmission */
if ((TWSR & 0xF8) != typeTWI)
error_led();
}

voidwrReg(uint8_t reg, uint8_t dat){
  //send start condition
twiStart();
twiAddr(camAddr_WR, TW_MT_SLA_ACK);
twiWriteByte(reg, TW_MT_DATA_ACK);
twiWriteByte(dat, TW_MT_DATA_ACK);
  TWCR = (1 << TWINT) | (1 << TWEN) | (1 << TWSTO);//send stop
  _delay_ms(1);
}

static uint8_t twiRd(uint8_t nack){
if (nack){
    TWCR = _BV(TWINT) | _BV(TWEN);
while ((TWCR & _BV(TWINT)) == 0); /* wait for transmission */
if ((TWSR & 0xF8) != TW_MR_DATA_NACK)
error_led();
return TWDR;
  }
else{
    TWCR = _BV(TWINT) | _BV(TWEN) | _BV(TWEA);
while ((TWCR & _BV(TWINT)) == 0); /* wait for transmission */
if ((TWSR & 0xF8) != TW_MR_DATA_ACK)
error_led();
return TWDR;
  }
}

uint8_trdReg(uint8_t reg){
uint8_tdat;
twiStart();
twiAddr(camAddr_WR, TW_MT_SLA_ACK);
twiWriteByte(reg, TW_MT_DATA_ACK);
  TWCR = (1 << TWINT) | (1 << TWEN) | (1 << TWSTO);//send stop
  _delay_ms(1);
twiStart();
twiAddr(camAddr_RD, TW_MR_SLA_ACK);
dat = twiRd(1);
  TWCR = (1 << TWINT) | (1 << TWEN) | (1 << TWSTO);//send stop
  _delay_ms(1);
returndat;
}

void wrSensorRegs8_8(conststructregval_listreglist[]){
  uint8_t reg_addr, reg_val;
conststructregval_list *next = reglist;
while ((reg_addr != 0xff) | (reg_val != 0xff)){
reg_addr = pgm_read_byte(&next->reg_num);
reg_val = pgm_read_byte(&next->value);
wrReg(reg_addr, reg_val);
next++;
  }
}
```

```
voidsetColor(void){
  wrSensorRegs8_8(yuv422_ov7670);
}

voidsetRes(void){
wrReg(REG_COM3, 4); // REG_COM3 enable scaling
  wrSensorRegs8_8(qvga_ov7670);
}

voidcamInit(void){
wrReg(0x12, 0x80);
  _delay_ms(100);
  wrSensorRegs8_8(ov7670_default_regs);
wrReg(REG_COM10, 32);//PCLK does not toggle on HBLANK.
}

voidarduinoUnoInut(void) {
cli();//disable interrupts

    /* Setup the 8mhz PWM clock
  * This will be on pin 11*/
  DDRB |= (1 << 3);//pin 11
  ASSR &= ~(_BV(EXCLK) | _BV(AS2));
  TCCR2A = (1 << COM2A0) | (1 << WGM21) | (1 << WGM20);
  TCCR2B = (1 << WGM22) | (1 << CS20);
  OCR2A = 0;//(F_CPU)/(2*(X+1))
  DDRC &= ~15;//low d0-d3 camera
  DDRD &= ~252;//d7-d4 and interrupt pins
  _delay_ms(3000);


    //set up twi for 100khz
  TWSR &= ~3;//disable prescaler for TWI
  TWBR = 72;//set to 100khz


    //enable serial
  UBRR0H = 0;
  UBRR0L = 1;//0 = 2M baud rate. 1 = 1M baud. 3 = 0.5M. 7 = 250k 207 is 9600 baud
rate.
  UCSR0A |= 2;//double speed aysnc
  UCSR0B = (1 << RXEN0) | (1 << TXEN0);//Enable receiver and transmitter
  UCSR0C = 6;//async 1 stop bit 8bit char no parity bits
}



voidStringPgm(const char * str){
do{
while (!(UCSR0A & (1 << UDRE0)));//wait for byte to transmit
      UDR0 = pgm_read_byte_near(str);
while (!(UCSR0A & (1 << UDRE0)));//wait for byte to transmit
  } while (pgm_read_byte_near(++str));
}


static void captureImg(uint16_t wg, uint16_t hg){
uint16_t y, x;


StringPgm(PSTR("*RDY*"));


while (!(PIND & 8));//wait for high
while ((PIND & 8));//wait for low


   y = hg;
while (y--){
      x = wg;
    //while (!(PIND & 256));//wait for high
while (x--){
while ((PIND & 4));//wait for low
         UDR0 = (PINC & 15) | (PIND & 240);
while (!(UCSR0A & (1 << UDRE0)));//wait for byte to transmit
while (!(PIND & 4));//wait for high
while ((PIND & 4));//wait for low
while (!(PIND & 4));//wait for high
   }
   //  while ((PIND & 256));//wait for low
  }
   _delay_ms(100);
}


void setup(){
arduinoUnoInut();
camInit();
setRes();
setColor();
wrReg(0x11, 11); //Earlier it had the value: wrReg(0x11, 12); New version works better
for me :) !!!!
}
void loop(){
captureImg(320, 240);
}
```

REFERENCES

1. A. G. Dharmawan, P. Xavier, D. Anderson, K. B. Perez, H. H. Hariri, G. S. Soh, A. Baji, R. Bouffanais, S. Foong, H. Y. Low, K. L. Wood,"A bio-inspired miniature climbing robot with bilayer dry adhesives: Design, Modeling, and experimentation", Proceedings of the ASME 2018 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2018, August 26-29, 2018, Quebec City, Quebec, Canada.

2. Nguyen, A. D., & Shimada, A. (2015). Equilibrium control on four-limbed climbing robot. *Applied Mechanics and Materials, 799-800*, 1021-1027. doi:http://dx.doi.org/10.4028/www.scientific.net/AMM.799-800.1021

3. Tlale, N. S. (2006). A MODULAR DESIGN OF A WALL-CLIMBING ROBOT AND ITS MECHATRONICS CONTROLLER. *South African Journal of Industrial Engineering, 17*(2), 197-208. Retrieved from https://search.proquest.com/docview/199308361?accountid=158790

4. Shen, W., Gu, J., &Shen, Y. (2006). Permanent magnetic system design for the wall-climbing robot. *Applied Bionics and Biomechanics, 3*(3), 151-159. Retrieved from https://search.proquest.com/docview/200191888?accountid=158790

5. "Design and Analysis of A Miniature Two-Wheg Climbing Robot with Robust Internal and External Transitioning Capabilities" Darren C. Y. Koh1;*, Audelia G. Dharmawan1;*, Hassan H. Hariri1;2, Gim Song Soh1, Shaohui Foong1, Roland Bouffanais1, Hong Yee Low1, and Kristin L. Wood1.

6. "ORION-II: A Miniature Climbing Robot with Bilayer Compliant Tape for Autonomous Intelligent Surveillance and Reconnaissance" Hassan H. Hariri, Darren Koh C. Yung, HoongChing Lim, Audelia G. Dharmawan, Van Duong

Nguyen, Gim Song Soh*, ShaohuiFoong, Roland Bouffanais, Hong Yee Low and Kristin L. Wood

7. “A Bio-Inspired Miniature Climbing Robot with Bilayer Dry Adhesives: Design, Modeling, and Experimentation” Hassan H. Hariri, Darren Koh C. Yung, HoongChing Lim, Audelia G. Dharmawan, Van Duong Nguyen, Gim Song Soh*, ShaohuiFoong, Roland Bouffanais, Hong Yee Low and Kristin L. Wood.

8. Fengyu, X., Jingjin, S., &GuoPing, J. (2015). Kinematic and dynamic analysis of a cable-climbing robot. *International Journal of Advanced Robotic Systems, 12*Retrieved from https://search.proquest.com/docview/1735266029?accountid=158790

9. Nansai, S., & Mohan, R. E. (2016). A survey of wall climbing robots: Recent advances and challenges. *Robotics, 5*(3), 14. doi:http://dx.doi.org/10.3390/robotics5030014

10. Xu, F., Wang, B., Shen, J., Hu, J., & Jiang, G. (2018). Design and realization of the claw gripper system of a climbing robot. *Journal of Intelligent & Robotic Systems, 89*(3-4), 301-317. doi:http://dx.doi.org/10.1007/s10846-017-0552-3

11. Xuyan, H., Yilin, S., Shengyuan, J., Long, L., Chen, T., Lining, S., &Zongquan, D. (2018). Adhesion mechanism of space-climbing robot based on discrete element and dynamics. *Advances in Mechanical Engineering, 10*(4) doi:http://dx.doi.org/10.1177/1687814018772934

12. Xu, Y., & Liu, R. (2017). Concise method to the dynamic modeling of climbing robot. *Advances in Mechanical Engineering, 9*(2) doi:http://dx.doi.org/10.1177/1687814017691670

Useful Websites:

1. https://www.mathworks.com/products/simscape.html.

2. https://www.mathworks.com/matlabcentral/fileexchange/64227-matlab-and-simulink-robotics-arena-walking-robot.

3. https://www.youtube.com/watch?v=CnlGG5lIp-0.