



Department of Electrical & Computer
Engineering
AMERICAN UNIVERSITY *of*
KUWAIT

FALL 2020

Smart Gardening System



Prepared by:

Zainab Mohammed S00043640

Fajer Al-Moqata S00049134

Mubarak AlTemimi S00031459

Supervisor:

Dr.Mohammed El-Abd

Declaration

We certify that this project work titled “*Smart Gardening System*” is our own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources has been properly acknowledged / referred.

Project Members

Mubarak AlTemimi	S00031459
Fajer Al-Moqata	S00049134
Zainab Mohammed	S00043640

Plagiarism Certificate (Turnitin Report)

This thesis has been checked for Plagiarism. Turnitin report endorsed by the Supervisor(s) is attached.

Mubarak AlTemimi	S00031459
Fajer Al-Moqata	S00049134
Zainab Mohammed	S00043640

Signature of Supervisor(s):

Copyright Statement

- Copyright in text of this project report rests with the student authors. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the authors and lodged in the Library of AUK. Details may be obtained by the Librarian. This page must form part of any such copies made. Further copies (by any process) may not be made without the permission (in writing) of the authors.
- The ownership of any intellectual property rights which may be described in this project report is vested in AUK's Department of Electrical & Computer Engineering, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the AUK's Department of Electrical & Computer Engineering, which will prescribe the terms and conditions of any such agreement.
- Further information on the conditions under which disclosures and exploitation may take place is available from the Library of AUK, Kuwait.

Acknowledgements

It has been a great opportunity to learn new experience in a real-life project development, followed by the knowledge of all project development lifecycle including project definition, analysis, and implementation. We want to thank everybody who made this possible to us and special thanks and our deepest gratitude to our mentor and supervisor “Dr. Mohammed EL Abd” for the great efforts he did to provide all required and useful information and making the path clear to us. Furthermore, we appreciate our supervisor for her patience and guidance along the semester.

Mubarak AlTemimi S00031459

Fajer Al-Moqata S00049134

Zainab Mohammed S00043640

Abstract

Intelligent and smart systems became a trend in most of business domains where human considered the main resource in managing and running the related duties. Make systems automated means that a technique of using computers and information technology in monitoring and controlling parameters of day-to-day life. Using the concept of software and hardware development such as IoT and Arduino, to make sensors or any hardware components to communicate with each other smartly and perform the desired functions.

The important aspect of this project design is that it saves cost, ensures safety, and helps in monitoring plants and their progress. When farmers or normal people want to make plantings and prepare their own garden, their concerns are mainly in maintenance especially at the beginning stages of planting. During days and due to lack in following up with plants progress and periodical maintenance, plants get destroyed. This project prototype will help people to automated monitoring the parameters and their progress, in addition to the planting and watering process will be implemented to the garden system. It plays a vital role and serves as a good companion for plants. The system is an autonomous solution for various problems, and it allows things to be sensed or controlled remotely using the proper set of components and sensors.

Keywords—*Gardening, Planting System, plant progress, Seeds, smart watering.*

Table of contents

DECLARATION	i
PLAGIARISM CERTIFICATE (TURNITIN REPORT)	ii
COPYRIGHT STATEMENT	iii
ACKNOWLEDGEMENTS	iv
ABSTRACT	v
TABLE OF CONTENTS	ERROR! BOOKMARK NOT DEFINED.
LIST OF FIGURES	ERROR! BOOKMARK NOT DEFINED.i
LIST OF TABLES	ERROR! BOOKMARK NOT DEFINED.
CHAPTER1: INTRODUCTION	1
1.1 BACKGROUND.....	1
1.2 PROBLEM STATEMENT	2
1.3 AIMS AND OBJECTIVES OF THE PROJECT	3
1.4 SIGNIFICANCE, SCOPE AND DEFINITIONS	3
1.5 SWOT ANALYSIS.....	3
1.6 REPORT OUTLINES	4
CHAPTER2: LITERATURE REVIEW	5
2.1 HISTORICAL BACKGROUND.....	5
2.2 A CHERRY-TOMATO HARVESTING ROBOT.....	5
2.3 iPLANT: THE GREENHOUSE ROBOT	6
2.4 SMART GARDENING SYSTEM	6
2.5 A PLANT MONITORING AND SMART GARDENING SYSTEM USING IoT.....	8
2.6 INDOOR ROBOT GARDENING: DESIGN AND IMPLEMENTATION.....	9
2.7 FOOD COMPUTER AUTOMATED GARDENING SYSTEM.....	10
2.8 SUMMERY AND IMPLICATIONS	12
CHAPTER3: METHODOLOGY, DESIGN, AND ANALYSIS	14
3.1 METHODOLOGY	14
3.2 RESEARCH DESIGN.....	14
3.2.1 <i>Design Alternative1</i>	15
3.3 SOFTWARE AND HARDWARE.....	15
3.3.1 <i>Switching Power Supply 24V</i>	16
3.3.2 <i>RS485 Soil Moisture Sensor</i>	16
3.3.3 <i>NEMA 17 Stepper Motor</i>	17
3.3.4 <i>DRV8825 Stepper Motor Driver Carrier, High Current</i>	18
3.3.5 <i>5v 5A Step down voltage regulator D24V50F5:</i>	19
3.3.6 <i>12V 4.5A Step down voltage regulator D36V50F12:</i>	19
3.3.7 <i>2 channel 5V Relay Module</i>	20
3.3.8 <i>Plate, Bracket, and Axis</i>	20
3.3.9 <i>Plastics for Cable Support and cable Carriers</i>	21
3.3.10 <i>Pressure Regulator</i>	22
3.3.11 <i>Solenoid Valve, Vacuum Pump, and Tubes</i>	23
3.3.12 <i>NBT to barb adapter, Lock Adapter, Lock Needles</i>	24

3.3.13 <i>Arduino MEGA</i>	25
3.3.14 <i>Seed Holder</i>	25
3.3.15 <i>Aluminum Beams for X, Y and Z axis</i>	26
3.4 ANALYSIS	26
3.4.1 <i>System Flowchart</i>	29
3.5 COST ANALYSIS	31
3.6 ETHICS AND IMPLICATIONS	32
CHAPTER4: IMPLEMENTATION	33
4.1 PROJECT DESIGN DETAILS	33
4.2 HARDWARE IMPLEMENTATION	35
4.3 SOFTWARE IMPLEMENTATION	43
4.4 IEEE STANDARDS	45
CHAPTER5: EVALUATION	46
5.1 PROJECT SURVEY	46
CHAPTER6: CONCLUSION & FUTURE WORKS.....	51
6.1 CONCLUSION	51
6.2 FUTURE WORKS	51
REFERENCE	53
APPENDIX	55
PROJECT SOURCE CODE	55

List of Figures

FIGURE 1 BLOCK DIAGRAM [5]	7
FIGURE 2 SYSTEM ARCHITECTURE DESIGN	9
FIGURE 3 ROBOTIC SYSTEM PROTOTYPE [7]	10
FIGURE 4 SYSTEM BLOCK DIAGRAM [8]	12
FIGURE 5 WATERFALL METHODOLOGY	15
FIGURE 6 POWER SUPPLY.....	16
FIGURE 7 SOIL MOISTURE SENSOR.....	17
FIGURE 8 STEPPER MOTOR.....	18
FIGURE 9 STEPPER MOTOR DRIVER CARRIER, HIGH CURRENT	18
FIGURE 10 STEP-DOWN REGULATOR	19
FIGURE 11 12V, 4.5A STEP-DOWN VOLTAGE REGULATOR	19
FIGURE 12 2 CHANNEL 5V RELAY MODULE	20
FIGURE 13 PLATE, BRACKET, AND AXIS PARTS.....	20
FIGURE 14 PLASTICS FOR CABLE SUPPORT AND CABLE CARRIERS.....	22
FIGURE 15 PRESSURE REGULATOR	23
FIGURE 16 SOLENOID VALVE, VACUUM PUMP, AND TUBES	23
FIGURE 17 NBT TO BARB ADAPTER, LOCK ADAPTER, LOCK NEEDLES	24
FIGURE 18 ARDUINO MEGA.....	25
FIGURE 19 SEED HOLDER.....	25
FIGURE 20 THREE TYPES OF ALUMINUM BEAMS.....	26
FIGURE 21 SYSTEM BLOCK DIAGRAM	28
FIGURE 22 SYSTEM FLOWCHART	30
FIGURE 23 SYSTEM PROTOTYPE - INITIAL DESIGN	33
FIGURE 24 CIRCUIT DIAGRAM.....	35
FIGURE 25 STEPPER MOTOR (Z AXIS)	36
FIGURE 26 STEPPER MOTOR (RIGHT X AXIS)	36
FIGURE 27 SYSTEM Y AXIS TRACK MOVEMENT WITH STEPPER MOTOR.....	36
FIGURE 28 Z AXIS TRACK	36
FIGURE 29 LEFT X AXIS TRACK	36
FIGURE 30 RIGHT X AXIS TRACK.....	36
FIGURE 31 SYSTEM Y AXIS BRIDGE ALUMINUM	37
FIGURE 32SOLENOID ELECTRONIC VALVE WITH LIFT CHECK WATER VALVE	38
FIGURE 33 PROTOTYPE DESIGN - SIDE VIEW	38
FIGURE 34 PROTOTYPE DESIGN - BACK SIDE VIEW	39
FIGURE 35 ARM WITH CHAIN AND AXISES TRACK.....	39
FIGURE 36 PLANTING SEEDS AND IRRIGATION PROCESS	40
FIGURE 37THREE AXISES ROTATIONS	41
FIGURE 38 SYSTEM FINAL RESULTS	42
FIGURE 39 ARDUINO IDE	44
FIGURE 40 SURVEY Q1.....	46
FIGURE 41 SURVEY Q2.....	47
FIGURE 42 SURVEY Q3.....	48
FIGURE 43 SURVEY Q4.....	48
FIGURE 44 SURVEY Q5.....	49

FIGURE 42 SURVEY Q6.....	49
FIGURE 43 SURVEY Q7.....	50
FIGURE 44 SURVEY Q8.....	50

List of Tables

TABLE 1 PROJECT SWOT ANALYSIS 4
TABLE 2 COST ANALYSIS 31
TABLE 3 FUNCTIONAL REQUIREMENTS PRIORITIZATION 46

CHAPTER1: INTRODUCTION

Of course, there are many products and projects have been implemented that are related to the automated gardening system which aims to improve the quality of planting, reduce wasting waters and fruit items, and increase the interaction between people and their greenhouse planting system in a way that emphasize the usefulness of the project.

Some of these patents are related to gardening systems those monitor the plants and their progress, some are performed planting and watering process, some are detecting and check the soil moisture, and some those perform harvesting to the items which are repined, and some projects are combining between all these features in one interface. But, in our project, and in such part of the planting process, it requires the human intervention specifically when the items get ripened and have to be picked up. Our prototype design will use the advantages of automation systems technology to improve the user experience in gardening operation field through decrease user intervention, make it more reliable, and provide different features using the scope of help people to perform planting and observe the plants during the growth stage until harvest to utilize the gardening process in a maximum level.

1.1 Background

Gardening is the concept of growing food products and plants in our houses and local gardens. About 95% of plants, either food crops or any other types of corps are grown in open fields. Automated Gardening Systems provide favorable environmental conditions for the plants, since it is designed to protect the plants from the adverse climatic conditions such as wind, excessive radiation, precipitation, insects, extreme temperature, and diseases. Gardening systems are framed or inflated structures with transparent or translucent material that is sufficiently large to grow crops under partial or fully controlled environmental conditions to achieve optimum growth and productivity. [1]

So, as long as the proposed concept requires the user or the cultivator machine used by farmers always to keep track of the growing plants, to observe and monitor, to water the crops according to specific time. Hence, this requires a manual check to implement all these. So, what about making a Smart Gardening System which can offer support to people to monitor the growth

of plants and to keep them updated with the status? For such purpose and as discussed before, Smart Gardening System concept can be achieved by implement the automation of gardening processes without the intervention of any human to perform a set of operations such as monitor the plants, water the plants, or even decide if crops got ripened or not yet using the technique of image processing.

Our main goal in this prototype is to develop an autonomous greenhouse automated system consisting of autonomous device enhanced with computation, sensing, and communication. The combination between the hardware components and plants transforms energy, water until get into products and plants get ripened. In this type of precision agriculture system planting and watering will be delivered locally on-demand and plants will be harvested when they are ripe, but the harvesting part will be considered in the future works.

The idea of the Smart Gardening System is basically a device acting as robot with arms which can move in different directions to give ability planting, then watering plants according to soil moisture sensor detection, observing the plants during their growth and monitor the stages, and finally when the plants is ready the farmer will pick them up.

1.2 Problem Statement

In the area of agriculture, three main concerns and issues are stuck in this area that prevent to manage the planting process in a better way. First problem is to keep observing plants during their growth and lifecycle, which may lead crops sometimes to get destroyed due to lack in monitoring. Secondly, spaces are not being used well during cultivation process.

Third and finally, the Irrigation which is considered the biggest concern of the developing countries in agriculture. The main explanation for this is low rainfall, since more land is not irrigated. Another very significant explanation for this is the unplanned usage of water supplies, so that more water is lost. Water is only supplied to the plant zone via the drip system or drop specific amount of water that needed at specific time and through planned number of watering times, thereby saving a considerable amount of water. To automate the irrigation process means that system can supply water to plants whenever they are in need of water at regular power supply intervals. Here no need of turning ON/OFF of valves.

1.3 Aims and Objectives of the Project

The main objective of designing and building Smart Gardening System is to provide the ability to planting, irrigate, and monitoring if it needs water or not. The system will be based on the soil moisture to plant and watering, and to achieve the following objectives:

- Part of the system will be responsible to sows the seeds and watering.
- Soil moisture will observe if the plants need water or not based on if the soil wet or dry.
- Autonomously planting seeds in specific points of the garden.

1.4 Significance, Scope and Definitions

Each person has small garden or greenhouse project, is required to perform a lot of activities, keep tracking and observing plants, and show high attention to the planting process during the plant's lifecycle. The process is not that easy, and to guarantee the safety of crops and that the results are close to the expectations, farmer should follow specific rules in a specific time and to keep following the progress of plants continuously. All these considerations led us to think seriously to find a practical solution for this matter, especially with the increasing in the number of greenhouses or personal gardens. From another point of view, and with the huge development of technologies, we've been encouraged to subject the available technologies and resources to serve the humanity in the agriculture field as an attempt to design an autonomous plant monitoring and smart gardening system with the help of controllers such as Raspberry Pi and Arduino to detect planting process parameters and take the proper actions.

1.5 SWOT Analysis

As seen in table1, the SWOT analysis shows a brief view of the Strengths, Weaknesses, Opportunities, and Threats predicted for this project. It is important for us as a team to look at these crucial points to help us make decisions throughout the project in order to overcome the different problems and to achieve the project's goals and objectives.

Table 1 Project SWOT Analysis

<ul style="list-style-type: none"> ● Strengths <ul style="list-style-type: none"> - Accessibility. - Autonomous - Easy to use. - Cost-effective project. Doesn't cost a lot to implement the system. - Maintainability. Components are available and can be easily maintained - New idea to be implemented 	<ul style="list-style-type: none"> ● Weaknesses <ul style="list-style-type: none"> - Limitations in some capabilities. Not designed to harvest and pick up the crops after ripened, also will not serve to classify between crops with the same characteristics - Size of system prototype. The size of the prototype is limited
<ul style="list-style-type: none"> ● Opportunities <ul style="list-style-type: none"> - The need for smart gardening and manages planting process in a better way is increasing significantly in different communities. - High probability for the system to be applied in the real environment. 	<ul style="list-style-type: none"> ● Threats <ul style="list-style-type: none"> - Users confidence to use the system. - System protection from natural phenomena in case project implemented since some parts of the project will be installed in outside.

1.6 Report Outlines

Each section in this report clarifies a set of project elements and attributes in different ways. Chapter1 define an introduction about the system and brief overview, problem statements, project objectives, and a summary of the design process and SWOT analysis. Chapter2, illustrates the literature review of the project, previous works and similar projects, and also describe the ethical and environmental issues related to the project. Chapter3 is mainly clarifying the methodology, proposed solution design in brief, and finally analysis and preliminary cost.

Chapter4 is about the project detailed design by describing each part of the system and its role, hardware implementation, and software implementation, and finally clarifying ethics and limitations.

Chapter5 is about the evaluation and survey conducted with a random segment of people to study how much system requirements are matching user's need. In Chapter6, discussing the project conclusion and future works.

CHAPTER2: LITERATURE REVIEW

This part of the report compares the idea with similar projects that have been implemented before, and to clarify the functionalities of these projects and what are the software and hardware components used to achieve the desired outcome. This approach usually used in order to observe similar ideas, detect the strengths and weaknesses of these projects to help in following best practice and avoid issues or problems that may appear during the project development. It also includes some reviews and general concerns and standards that were addressed and mentioned by others regarding the implementation of this idea. The projects surveyed may not reflect the exact same system as our project, but they have many similarities in terms of the devices used and the way they are connected.

2.1 Historical Background

Greenhouse planting and commercial agriculture have reached a high level of automation and the use of implementing the information technology in this field, although mainly for broad-land crops, such as using the autonomous combiners.

A similar research was done in as tomato greenhouse project on the south of Italy by Mancuso. Sensors used in the project are the air temperature, humidity and soil temperature, and all these are measured using Sensicast devise with wireless sensor network. [2]

Also, Web-based plant monitoring application has been developed to control and monitor readings of the system, where the Greenhouse user can read the measurements over the Internet connection. If some measurable variable changes rapidly, then an alarm will be sent to the owner through SMS notification or GPRS. [2]

2.2 A Cherry-Tomato Harvesting Robot

The main objectives behind this project are that the robot picks out the cherry tomatoes that already ripened without causing any damage to the surroundings, and leaves the ones that are

not ripe enough. Moreover, the robot identifies the ripe cherry tomatoes by image sensing using a camera. After that, the robot picks the ripe cherry tomato and places it in a basket. The robot repeats the previous steps until there are no cherry tomatoes left on the trees. The robot has additional features, such as picking the rotten cherry tomatoes and placing them in a separate basket. [3]

The system contains rows of cherry tomato plants. The space between the rows is fixed. Moreover, the space between any two cherry tomato trees is also fixed. The robot is placed in front of the first tree that is located in the first row. The robot takes a snapshot to locate the cherry tomatoes that are on the first tree using the camera. Then the arm of the robot will pick the cherry tomatoes and place it in the basket. After that, the robot moves toward the next tree for a specific distance. So, when finish harvesting, robot moves the same fixed distance every time to reach the next tree. When the robot finishes harvesting cherry tomatoes from the first row, then it will move to the next row to start harvesting following the same process. The robot also distinguishes between the ripe cherry tomatoes and the damaged one, and place them into two different baskets. [3]

2.3 iPlant: The Greenhouse Robot

iPlant is a robotic system employer that is easy to use, effortless, and saves time. The system has the ability to monitor and broadcast plants' state, water level, and nutrients. Therefore, iPlant system provides a great reduction in both financial and physical resources. [4]

The project designed to be used in green-houses and help in providing suitable and safe agricultural environments. The system consists of two parts, the first part responsible to control the environment of the greenhouse including temperature, humidity, light, in addition to the security system that detects smoke and sends SMS alerts to the owner, and provide a daily report about the conditions of the green-house. The second part of the system is to move the robot to detect the soil moisture, in addition to watering the plants and planting seeds. [4]

2.4 Smart Gardening System

The Smart gardening system aims to develop techniques in agriculture automation to flourish and deliver its full potential. This system designed by using Arduino microcontroller to overcome limitations of agriculture farming about supplying of water to plants by drip system with

the available water tables. Components used in this project design are Arduino microcontroller, motor pump, soil moisture sensor. The motor pump works according to the soil condition that is soil wet or dry; if soil is wet motor stops otherwise motor run to give water to plants. [5]

The status of motor is displayed on LCD. Different sensors are connected to the microcontroller to verify the temperature and CO gas exist. SMS will be sent to the owner in case of any critical situation. To keep the strangers away from the field an electric fence is included with slight shock. The electric fence will be deactivated when an authorized person presence is identified near the field through the RFID as shown in figure1. [5]

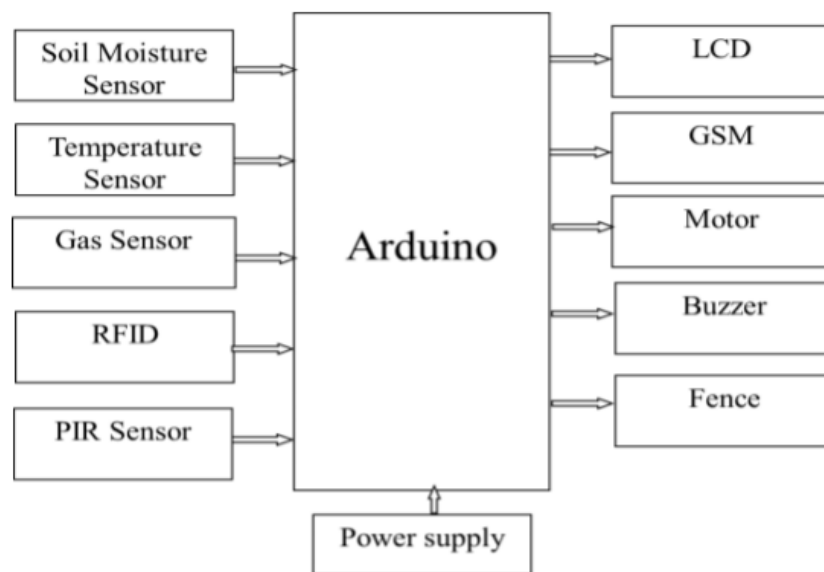


Figure 1 Block diagram [5]

The target behind this system is to create a sensor network based on low-cost soil moisture, temperature monitoring system which helps to track the moisture of the soil and temperature in real time and allow water to the plants based on the detecting values and the type of crop.

System starts working by sensors detects the values and provide it to the Arduino microcontroller. It converts these values into its appropriate form that is executing in it and gives the output in the form water flow according to the input values. In case, critical situation is detected through these sensors then an SMS is sent to the owner. An electrical fence is included to keep the strangers away from the field by its slight shock. Presence of a person is detected through PIR sensor and a led indication with buzzer alert will be given in case of unauthorized person's appearance. Authorized person's presence is identified using RFID module interfaced to the

controller since the RFID card is being carried by that candidate, in such case electric fence will be deactivated hence it doesn't produce any shock. [5]

2.5 A Plant monitoring and smart gardening system using IoT

This project automates plant monitoring and smart gardening using IoT in the Raspberry Pi platform. The main purpose of automation is to provide comfort to the people by reducing the manual work and to improve the overall performance of any system without the user interaction. All the sensors (Temperature, moisture, humidity, LDR, pH) used in this project interface with the Raspberry Pi controller. All this information about the garden can be directly monitored and controlled by the owner of the garden through a smart phone application using IoT technology. A camera is provided in this system to monitor the garden through the smart device. [6]

The proposed Smart Gardening System as shown in figure2 will provide convenience and comfort to the user by sensing and controlling the parameters of the garden without their physical presence. Any android supported device can be used to install the smart gardening application. The software's used are PHP, CSS, HTML, Apache 2, Python, and SQL. The CSS and PHP software's are used to develop and design a web page. Apache 2 is used as a web server in Linux system. The Python environment provides the space for coding in Raspberry Pi. SQL is used in programming and designed for managing data held in a relational database management system. All data are stored in the database and can be retrieved at any time. This will help the user to understand the relation between the plant growth and the mentioned garden parameters. [6]

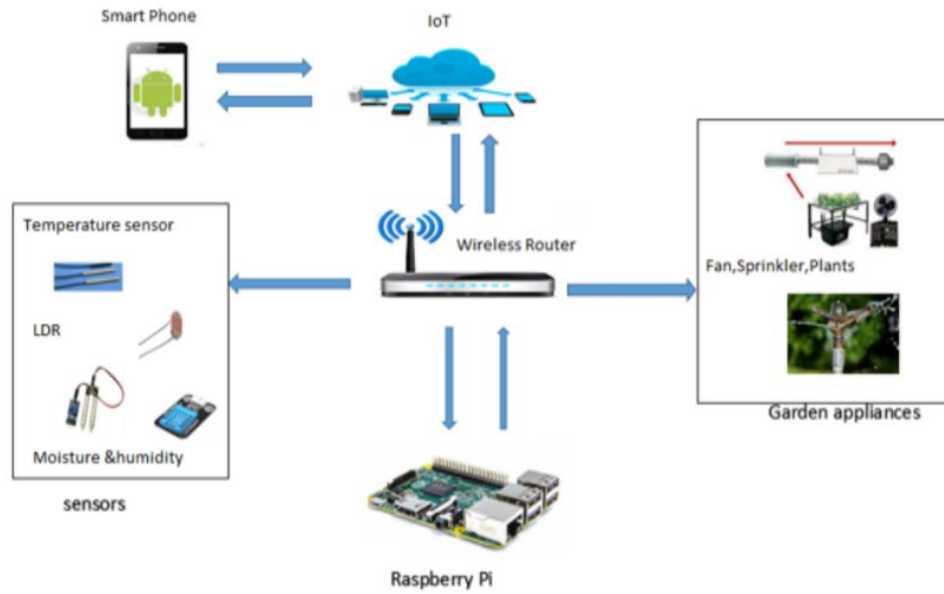


Figure 2 System Architecture Design

Figure2 illustrates the architecture design for plant monitoring and smart gardening system using IoT with the help of a controller Raspberry Pi. All the parameters of the garden like temperature, humidity, moisture, light intensity is controlled with the help of sensors like humidity sensor, moisture sensor, LDR, temperature sensor is interfaced with the Raspberry Pi board.

2.6 Indoor Robot Gardening: Design and Implementation

The main goal behind this project is to develop an autonomous greenhouse consisting of autonomous robots and pots and plants enhanced with computation, sensing, and communication. The network of robots, pots, and plants transforms energy, water and nutrients into produce and fruits. This precision agriculture system, the water and nutrients will be delivered locally on-demand and fruit will be harvested when they are ripe. [7]

The system is a distributed autonomous gardening system as shown in figure3 with applications in urban/indoor precision agriculture. The garden is a mesh network of robots and plants. The gardening robots are mobile manipulators with an eye-in-hand camera. [7]

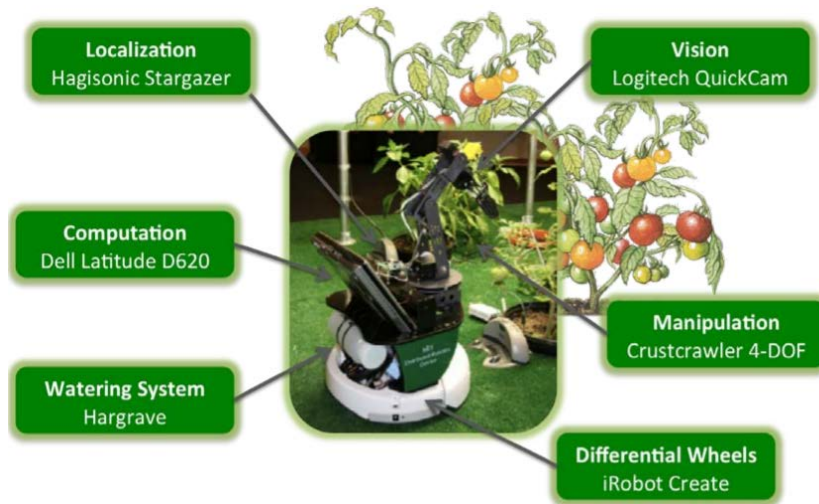


Figure 3 Robotic System Prototype [7]

The system architecture contains a set of components and networking equipment clarified such as follows:

- Notebook computer running Ubuntu Linux,
- An iRobot Create providing a bumper sensor (left/right), four cliff sensors, a wall sensor and an infra-red sensor on the robot base. The robot equipped with a rear-end caster wheel to improve odometry, it also has been powered by an external battery and is outfitted with a laser-cut Delrin structure to attach its peripherals. The robot is controlled from the notebook using an USB-to-Serial connection.
- Servo board, which controls the arm, provides an analog-digital converter for the force sensor, and PWM control for the water pump using an USB-to-Serial connection
- Water tank and pump (Hargraves) connected to the SSC-32
- 4-DOF arm with gripper and resistive force sensor
- Lithium-Polymer battery pack
- Logitech Quick cam connected to the notebook using USB.
- Atheros PCMCIA radio card

2.7 Food Computer Automated Gardening System

The Automated Gardening System can be used to monitor a garden as well as automate the irrigation process with minimum usage of water. It is an Internet of Things (IOT) based system which allows gardening by using sensors to determine soil moisture, soil pH level and light

intensity. The water moisture sensor which shows the moisture level. The pH level indicates the solubility of essential nutrients and the light intensity sensor tells us whether the plants are getting enough sunlight or not. The system sends data over the internet using a Wi-Fi module through which the user can monitor and decide the plant requirements based on the communicated data. [8]

As shown in the block diagram figure4, the Smart Gardening System consists of various sensors doing different jobs simultaneously according to the need. As illustrated in the block diagram, system has a set of sensors where the first would be the moisture level checking sensors which are used to check water content in the soil so that it can pump in the water if needed. The second would be the temperature sensor to monitor the flow or fluctuating temperature type. The humidity sensor comes then. Plants need photosynthesis to be done with the help of the light or in the natural term called as sunlight. Hence when the climate is not sunny and if it requires a certain amount of light, then an extra light emitting sensors are required. The display of the information from sensors like the moisture level, temperature level, and the humidity level is displayed using and mobile application. [8]

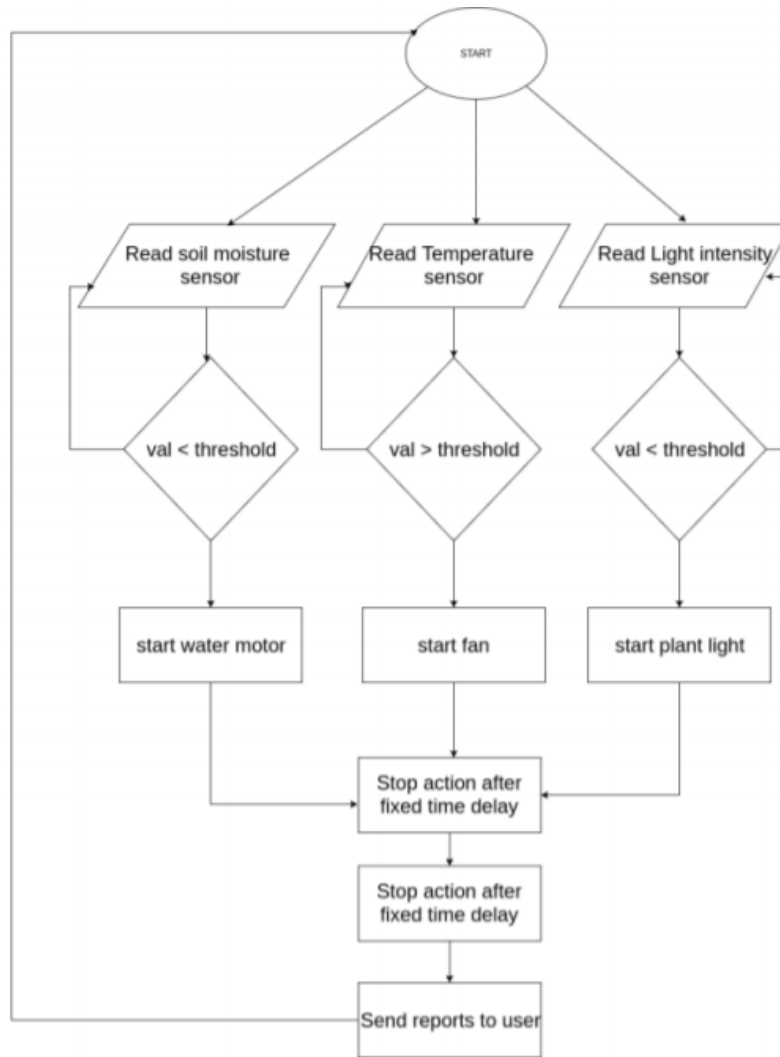


Figure 4 System Block Diagram [8]

2.8 Summery and Implications

As mentioned in this chapter, there are many projects that have been implemented in the real world and proposing some features that may be similar to ours or may not, such as planting and watering process, monitoring crops and plants in a way that guarantee the harvesting of these crops, and conduct observing to these plants in the garden. However, each surveyed project implemented using different algorithms and applied with different components, and some of them are sharing the same components such as sensors or even the same microcontroller.

In this project, we are introducing a smart gardening and autonomous planting system with intelligent capability of managing planting and watering process. The proposed smart gardening

can enable better experience in dealing with plants and conduct monitoring and observing to verify the progress of plants and take a proper action.

CHAPTER3: METHODOLOGY, DESIGN, AND ANALYSIS

The analysis and design part of the project is to provide a detailed description about objectives identified during the requirement specification phase at the beginning of the project, so our team could easily understand business needs and project scope. This will make objectives clearer and will help to implement the required functionalities as expected. Creating the analysis model requires many steps; the first step is to create an initial flowchart diagram, which is updatable during the project analysis phase.

In this section, the system architecture and proposed functionalities will be discussed. The design details will be discussed in this section also, which represents the requirements that will be implemented during development phase. In addition, to list all hardware components that will be used in the implementation and review their specifications as well.

3.1 Methodology

During our research and studying the project requirements, we have conducted a survey including a list of questions aims to clarify how much the project idea is acceptable among public people, and how will the system affect the environment and the social life positively. The methodology implemented in the Survey, which is a systematic and logical method for answering a list of questions submitted to provide more clarifications about this study, it determines the influence of the dependent variables on the independent ones (Panndey & Pandey, 2015).

3.2 Research Design

The methodology was selected to implement this study. Baum, MacDougall, and Smith (2006) according to Crotty (1998) clarified that the research methodology is a plan of action, process or design lying behind the choice and the use of particular methods to execute studies. This study used quantitative method.

Quantitative analysis is scientific approach to decision making. This approach starts with raw data, these data are analyzed and processed into valuable information to make a decision. The processing and manipulating of raw data into meaningful information is the heart of quantitative analysis (Render et al., 2015)

3.2.1 Design Alternative1

Experimental Method is a scientific method as the study of the relationships among variables give more precise, accurate and reliable results. It acts on an observation of single variable and its studies cause and impact relationship (Panndey & Pandey, 2015).

3.3 Software and Hardware

To build our project, we are following a standard project management methodology called Waterfall. Waterfall is a sequential development process, in which the project progress is flowing steadily downwards, and each stage's output is considered as input for the next stage in the project's lifecycle as shown in figure5.

We started with gathering information about the project through online research, then we moved to designing the project structure according to the proposed requirements, and we determined project components, hardware, and software requirements to build our project.

Then we will be moving to the implementation phase, where designing system prototype and programming part through writing the required code instructions to achieve the expected outcome. Finally, we will test and validate the system before we move to the final stage of the final prototype design, connecting all hardware components in one circuit, and integrate the hardware part with the software programming part.

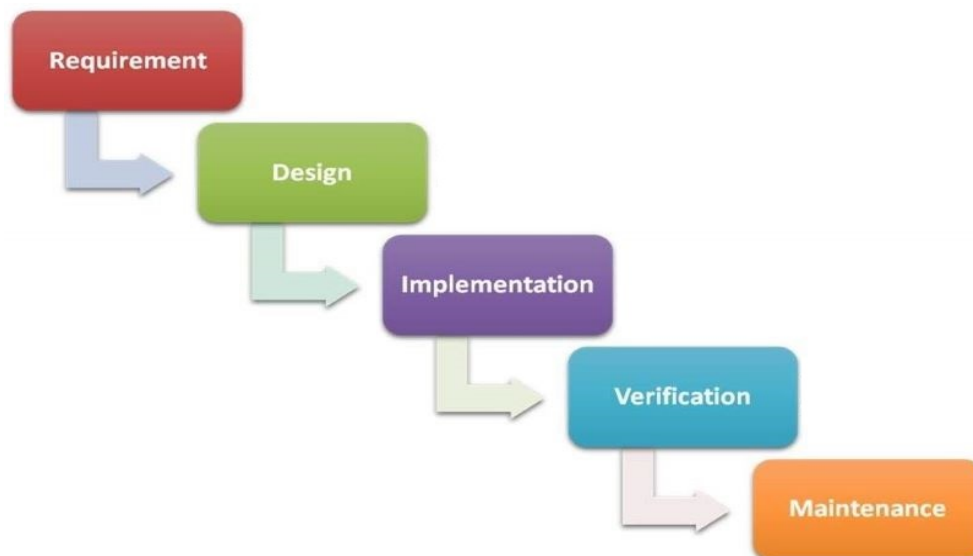


Figure 5 Waterfall Methodology

The reasons for adopting the Waterfall approach in our project is that it is simple and is most suitable for project idea since we are managing this project in a short amount of time. In addition, the Waterfall approach is easy to manage since processes are sequentially and each phase has specific deliverables. It also fits and works well for small to medium projects as ours, where requirements are well understood.

3.3.1 Switching Power Supply 24V

As shown in figure6, this component responsible to provide the needed power to the system with the following specification:

- DC Voltage: 24V.
- Rated Current: 6.25A (150Watts)
- Ac input:100~265VAC
- Protections: Short circuit / Overload / Over voltage/ Over temperature
- Waterproof rating: IP67



Figure 6 Power Supply

3.3.2 RS485 Soil Moisture Sensor

The Soil Moisture Sensor is a simple breakout for measuring the moisture in soil and similar materials. The soil moisture sensor is pretty straight forward to use. The large exposed pad

function as probes for the sensor, and acting as a variable resistor. The more water that is in the soil means the better the conductivity between the pads will be and will result in a lower resistance, and a higher SIG out.



Figure 7 Soil Moisture Sensor

This contact-less RS485 Soil Moisture Sensor as illustrated in figure7 adds a local voltage regulator to mitigate the voltage drop and RS485 transceiver to accommodate long wires, also reverse polarity and an additional ESD protection level is added.

3.3.3 NEMA 17 Stepper Motor

The NEMA 17 stepper motors as shown in figure8, it comes with step angle 1.8deg It has 4 wires, each phase draws 1.68A, with holding torque 5Kg/Cm (70 oz.in). 4 wire stepper motor has 2 leads in each phase, H Bridges with some driving topologies do have static friction effects, but this effect can be reduced by dithering stepper motor signals at higher frequencies. This powerful 200 resolution stepper motor allows the robot to move precisely in the X, Y, and Z directions.

- Rated Voltage: 12Volt
- Rated Current 1.68A
- Torque: 5Kg/Cm



Figure 8 Stepper Motor

3.3.4 DRV8825 Stepper Motor Driver Carrier, High Current

This breakout board for microstepping bipolar stepper motor driver features adjustable current limiting, over-current and over-temperature protection, and six microstep resolutions. This carrier as shown in figure9, operates from 8.2 V to 45 V and can deliver up to approximately 1.5 A per phase without a heat sink or forced air flow. This driver can be used as a higher-performance drop-in replacement for other boards in many applications. This board ships with 0.1" male header pins included but not soldered in, and lets you control one bipolar stepper motor at up to 2.2 A output current per coil.

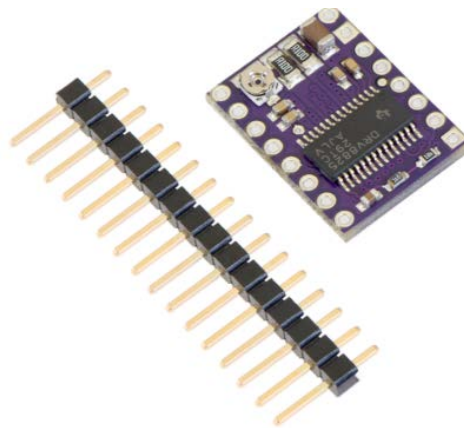


Figure 9 Stepper Motor Driver Carrier, High Current

3.3.5 5v 5A Step down voltage regulator D24V50F5:

The step-down regulator as shown in figure10 generates a fixed 5 V output from input voltages up to 38 V. It is a switching regulator or DC-to-DC converter which has a normal efficiency between 85% to 95%, which is much more efficient than linear voltage regulators, mostly when the difference between the input and output voltage is large.



Figure 10 Step-Down Regulator

At light loads, the switching frequency automatically changes to maintain high efficiencies. The regulator has a typical quiescent current draw of less than 1 mA, and the ENABLE pin can be used to put the board in a low-power state that reduces the quiescent current

3.3.6 12V 4.5A Step down voltage regulator D36V50F12:

As illustrated in figure11, this voltage regulators generate lower output voltages from input voltages as high as 50 V. This is a switching regulator OR switched-mode power supplies (SMPS), which makes them much more efficient than linear voltage regulators, especially when the difference between the input and output voltage is large. These regulators can typically support continuous output currents between 2 A and 9 A, depending on the input voltage and output voltage.



Figure 11 12V, 4.5A Step-Down Voltage Regulator

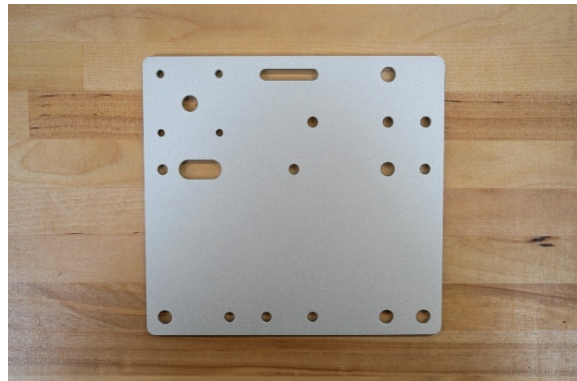
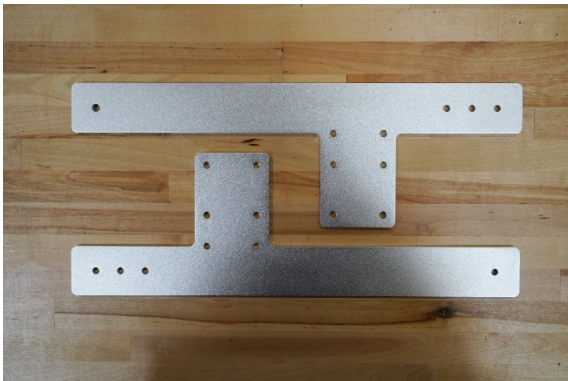
3.3.7 2 channel 5V Relay Module

This module as shown in figure12 is used to control various appliances and equipment with large current. This is a LOW Level 5V 2-channel relay interface board, and each channel needs a 15-20mA driver current. It is equipped with high-current relays that work under AC250V 10A or DC30V 10A. It has a standard interface that can be controlled directly by microcontroller. This module is optically isolated from high voltage side for safety requirement and also prevent ground loop when interface to microcontroller.



Figure 12 2 channel 5V Relay Module

3.3.8 Plate, Bracket, and Axis



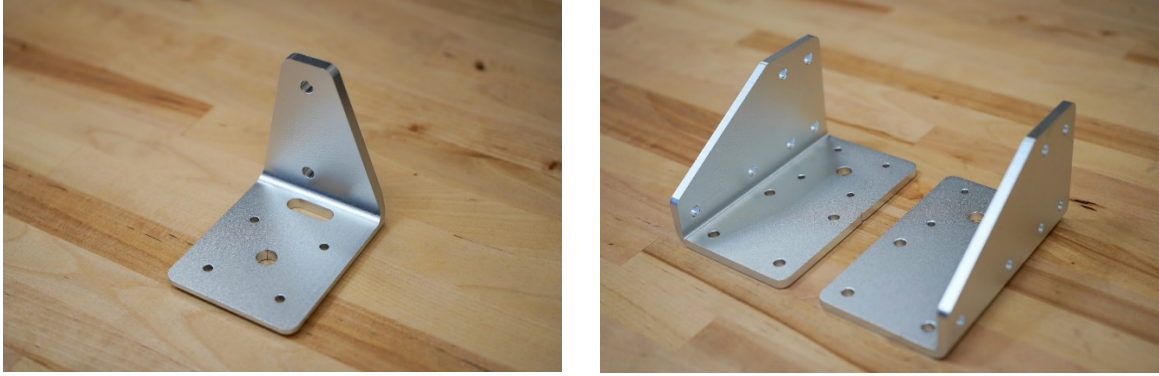


Figure 13 Plate, Bracket, and Axis Parts

As shown in figure 13, these parts are ordered from outside Kuwait, and we ordered them from FarmBot site because they are very complicated to be designed here, first we have Wheel Plates: these X Axis plates for both sides of the Robot, Second we have Corner Brackets: it's for the connecting X Axis Aluminum bars with the Y axis Aluminum bar, third, we have Cross Slide Plate: this part is responsible of connecting Z Axis Aluminum bar to the Y Axis Aluminum bar, finally, the Z Axis motor mount: this component securely attached the Z Axis motor to the Z Axis extrusion.

3.3.9 Plastics for Cable Support and cable Carriers

As shown in figure14, these parts are used to mount and support both the x-axis and y-axis cable carriers. The gusset also functions as an area for cables, tubes, and LED light strips to be routed through. And housings protect the gantry and cross-slide motors and vacuum pump from rain. The following figure16 also shows cable carrier which is a hollow plastic chains are used throughout the robot to manage cables and tubing along each of the three axes, and 4-wire waterproof cables connect each NEMA 17 stepper motor to the Robot. Finally, the Robot Head that shown in the figure combines the watering, weeding and see injection functions of the robot into one static head.

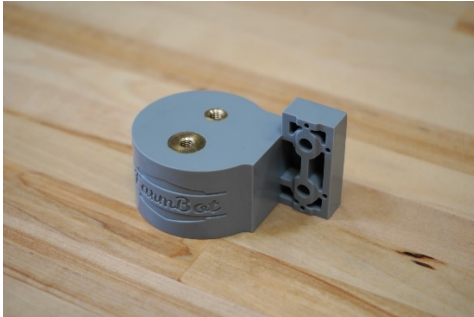
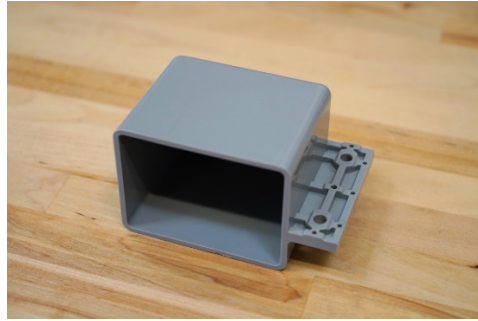
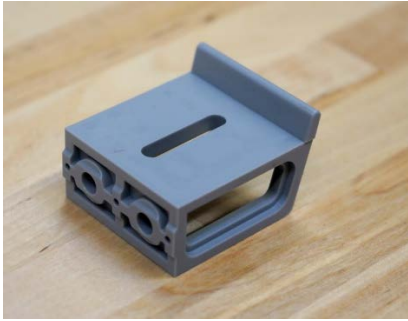


Figure 14 Plastics for Cable Support and cable Carriers

3.3.10 Pressure Regulator

This pressure regulator as illustrated in figure15 is responsible to reduce the pressure from a municipal water supply down to 15 PSI (103 kpa, 1.03 bar) so that our robot can more easily control the amount of water it disperses



Figure 15 Pressure Regulator

3.3.11 Solenoid Valve, Vacuum Pump, and Tubes

Figure 16 shows the Solenoid Valve, Vacuum Pump, and Tubes parts. The Solenoid Valve has 24V valve controls the flow of water from the water source to our robot tubing. The 24V vacuum pump sucks air through the seeder's lock needle in order to suction-hold a seed on the needle tip during planting. And these flexible silicone tubes can carry water to the tool head and suck vacuum air





Figure 16 Solenoid Valve, Vacuum Pump, and Tubes

3.3.12 NBT to barb adapter, Lock Adapter, Lock Needles



Figure 17 NBT to barb adapter, Lock Adapter, Lock Needles

The NBT adapters are used on the inlet of the solenoid valve and the outlet of the pressure reducer, For the Lock adapter screws into the robot head and allows various lock needles to be quickly attached for seed injection, and for the Lock Needles, these needles are used for seed injection.

3.3.13 Arduino MEGA

The Arduino Mega 2560 as shown in figure18 is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.



Figure 18 Arduino MEGA

3.3.14 Seed Holder

The seed holder used to hold the different types of seeds we used in our project prototype, which was designed and printed using 3D printing technology according to their positions as C point contains Lettuce, B contains Rocca, and A contains Spinach as shown in figure19.



Figure 19 Seed Holder

3.3.15 Aluminum Beams for X, Y and Z axis

The Aluminum beams are made from aluminum extrusions. They can be scaled to be taller or shorter. The two large spaces inside the beams are used to conceal and protect the GT2 timing belts that run along the tracks and up to the stepper motors pulleys. There are three Types of the bars that we used: for both sides of X Axis 20mm x 60mmx 500mm V-Slot, 20mm x 60mm x 1000mm V-Slot for Y Axis, 20mm x 20mm x 1000mm V-Slot for Z Axis. the figure below shown the three types of beams.



Figure 20 three types of Aluminum beams

3.4 Analysis

Here is a design of plant monitoring and smart gardening system with the help of Arduino microcontroller as control unit. All the parameters of the garden like moisture controlled with the help of sensors like moisture sensor, which are interfaced with the Arduino microcontroller board.

The system proposes a set of controlling features, mainly focusing on controlling and managing the planting and watering processes. The system provides the ability to watering plants based on the humidity of the soil that will be identified using the proper sensor.

The approach that we are following in the project development is to test and verify each hardware component at first level, test functions and features separately to confirm that every little part of the project is working properly before integrating them all together, instead of proceeding with development with errors or bugs may occur during the implementation. In addition to analyzing the efficiency of each component to implement the expected system behavior.

System block diagram as shown in figure21, clarifies that there are components on the left side of the diagram which represent the input into the control unit, and components on the right side which represent the output. In the middle, there is the micro controller unit (MCU) block between the input the output parts and considered as the brain of the system where all needed processing implemented in this unit.

The first part in the project as illustrated in figure21 which provides the power to the system is the power supply, which converts AC to DC so it could be connected to the electricity which gives 24V DC to feed the whole system and not limited to the Arduino microcontroller.

3 stepper motors as an output to be used to move the handle of the system into the three axis, X, Y, and Z which represents the directions of left, right, up, down, in addition to forward and backward.

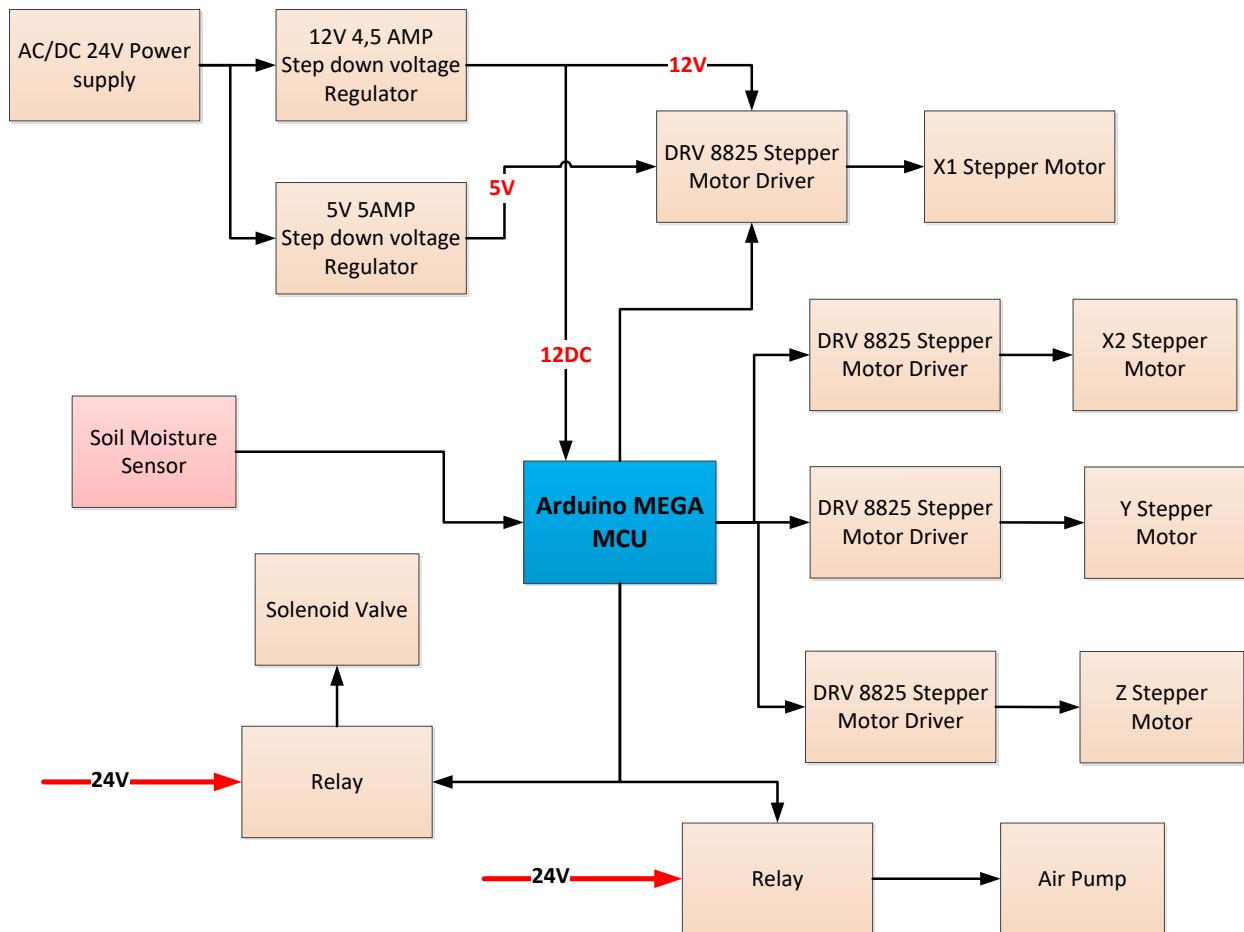


Figure 21 System Block Diagram

As illustrated in the block diagram, an AC power supply is needed to provide 24V DC which has to be decreased into 12V and 5V. The 5V needed for operating sensors, 12V for motors, 24V for solenoid valve and the air pump so be able to hold the seeds. The microcontroller has an output to stepper motor driver, which needs signal to turn it On/Off, and 5V for running, and 12V for the stepper motor. For the relay, it receives signal from Arduino and needs 24V to feed the air pump.

The soil moisture sensor has an input to the microcontroller, which used in the project instead of real-time clock due to the need to make the irrigation process running automatically instead of manual process, so irrigation will start operating based on the humidity values detected by soil moisture sensor, so, in case humidity is less than required, so will turn on the irrigation process, otherwise will not.

3.4.1 System Flowchart

The following figure22, represents the system flowchart that clarifying how system flows from the beginning until the end of the process. The workflow as shown in the figure6 starts with selecting the type of plant and specify how many seeds are to be planted in the garden. Then, the flow goes into planting process where the head start moving in specific path and planting the seeds in specific locations until completing all seeds.

According to the soil moisture sensor and based on the specific time will be developed in the project, for example at 6AM, system will then measure soil humidity and start irrigation process based on the measurements resulted.

After the irrigation process, system will continue to monitor the crops until they grow and keep detecting them repeatedly until it reaches to start with the observation and irrigation processes and finally, when the crops are ready the farmer will pick them up.

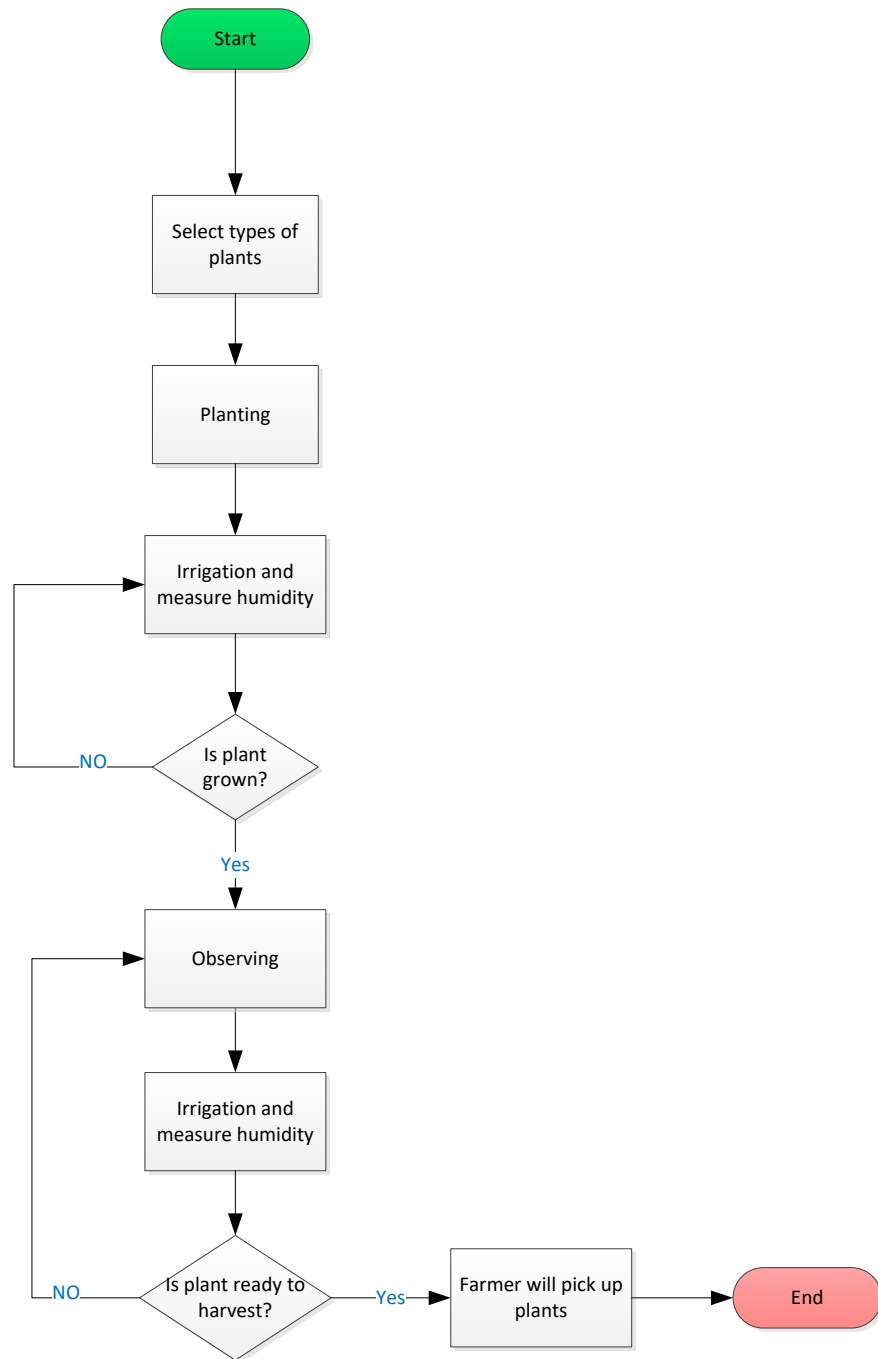


Figure 22 System Flowchart

3.5 Cost Analysis

Cost analysis part is mainly aiming to ensure that project meets the estimated cost and does not exceed the budget. Mainly, some of the hardware components have been ordered online through electronic components provider and some have been ordered from local electronic shops in Kuwait. These shops contain the most usable components in the area of engineering. As shown in table2, all hardware components will be used in the project are listed, the needed quantity for each component and price as found on the website. Total price as mentioned in the table is 271.820 KD including the shipping fees and taxes.

Table 2 Cost Analysis

#	Item Name	Qty	Cost - KD
	Switching Power Supply 12V 40A	1	KWD 12.500
	Concave Button – Red	1	KWD 00.750
	RS485 Soil Moisture Sensor	1	KWD 6.250
	Arduino MEGA	1	KWD 12.000
	5v 5A Step down voltage regulator D24V50F5	1	KWD 7.000
	12V 4.5A Step down voltage regulator D36V50F12	1	KWD 15.000
	2 channel 5V Relay Module	2	KWD 3.000
	Seeds holders	3	KWD 25.000
	DRV8825 Stepper Motor Driver Carrier, High Current	4	KWD 10.000
	Wheel Plate	2	KWD 6.11
	Corner Bracket	2	KWD 6.11
	Cross Slide Plate	1	KWD 4.58
	Z-Axis Motor Mount	1	KWD 4.58
	Plastics for Cable support and protection	1	KWD 1
	Robot Head	1	KWD 6.11
	Nema 17 Stepper Motor	4	KWD 5.50
	Motor Cables	4	KWD 4.58
	Cable Carrier	3	KWD 10.69
	NPT to Barb Adapter	2	KWD 1.53
	Pressure Regulator	1	KWD 2.44
	Lock Adapter	1	KWD 1.53
	Lock needles	3	KWD 1
	Aluminum Beams	4	KWD 4.58

Screws	1	KWD 9.16
V-Wheel	10	KWD 2
Timing Belt	3	KWD 16.79
Pulley	3	KWD 2
Shaft Coupling	1	KWD 2
8mm LeadScrew	1	KWD 10.69
LeadScrew Block	1	KWD 2.250
24Volt Power Supply	1	KWD 18.250
Solenoid Valve	1	KWD 2.500
Vacuum Pump	1	KWD 4.500
Tubes	3	KWD 9.250
Shipping	1	KWD 30.500
Tax	1	KWD 10.000
TOTAL		271.820

3.6 Ethics and Implications

1. Smart Gardening System supposed to work properly, since it focuses on different main concerns, planting, watering, observing, etc.
2. Regardless of fault or issues, the smart gardening system should behave ethically to be protected and not to expose people around to risks.
3. To accept and offer any criticism of technical work from our team or other users, to acknowledge and correct errors, and to credit properly the contributions of others.
4. Controllers of the system should pay constant attention to the safety and protection aspects.
5. The automated system can be developed by following the basic standards in the software and hardware solutions development, which will cover ethical situations.
6. The ultimate goal is to improve managing planting process with protecting plants and soil of getting damaged without observing it properly, and then ensuring that the automated project behave in acceptable ways is critical to earning the public's trust of these new technologies.
7. Engineers should conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor and usefulness of the profession.

CHAPTER4: IMPLEMENTATION

In this part, system design implementation is discussed which contains the hardware and software implementation for the smart gardening project. We also clarify all development activities starting from connecting the hardware components, create the main circuit, preparing the software development IDE, and installing the libraries that will be called to write the required code instructions including a set of classes and methods. Finally, the result of the project will be discussed in this chapter as well, also the final view of the system prototype design, describing in details the mechanical part of the prototype, and discussing the testing activities that will be conducted to confirm that all expected functionalities are operating properly.

4.1 Project Design Details

At the beginning of project design implementation, we will consider the dimensions, length, width, and height, and based on that, we will order some of the hardware components that fit to the project design requirements. After receiving the components, will start then with the mechanical part to design the prototype and assembling the electronic components such as sensors, motors, microcontrollers, etc., and then will use the 3D printing design technology for designing and printing the prototype. The initial design of the project prototype that was presented is shown in figure23, which clarifies the dimensions of the prototype with 1m width and 1.5m length.

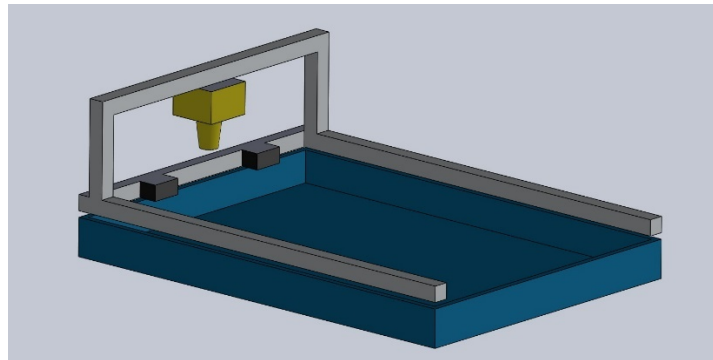


Figure 23 System Prototype - Initial Design

Regarding designing, implementation and testing, it has been structured to cover three main parts, mechanical, electronic, and software programming. At the beginning, tested all received components individually and after connecting them all together. We have divided the testing phase mainly into two levels. First level including testing the microcontroller and other hardware

components, to check and test if the Arduino and each component with the main functions are working properly. Also testing the signals that coming from the sensor and actuator connected to the main board, and this part have been completed successfully.

For the mechanical hardware part, aluminum parts have been ordered from local market in Kuwait, as long the main idea and design of the project has been developed based on **FarmBot** CNC farming project, which is a web app that can be loaded on any computer, tablet, or smartphone with a modern web browser, giving the controller the power to manage garden from anywhere at any time.

Based on the FarmBot design, all measurements, dimensions, and prototype design perspectives such as chain and wires, are have been already provided by FarmBot, so we can implement the same idea that was designed. All components are connected together based on the recommendations and directions by FarmBot, and then we connected all axes 2X, 1Y, and 1Z axis. The two X axes were attached to the edges of the belt, and the robot moves on the belt itself as well.

The electronic part design was implemented by connecting the Arduino microcontroller to the stepper motor driver which is responsible to move the system prototype. The stepper motor driver also responsible to control all motors in the system, which are 4 motors connected to the Arduino, and have used DC power supply as power source using 24V. Through the voltage regulator, system was able to reduce the voltage provided from 24V into 5V for those components need 5V to operate such as the two pins, Step and DIR. Also, using another voltage regulator to reduce the voltage from 24V into 12V for the stepper motors.

The programming part at this stage covered running the motors and the operation of moving the three axes X, Y, and Z, and then after to complete the part of put the seeds into the soil and perform the watering part of the system. The following figure24 shows the circuit diagram describing the wiring and the connecting the microcontroller to the stepper motor drivers carrier.

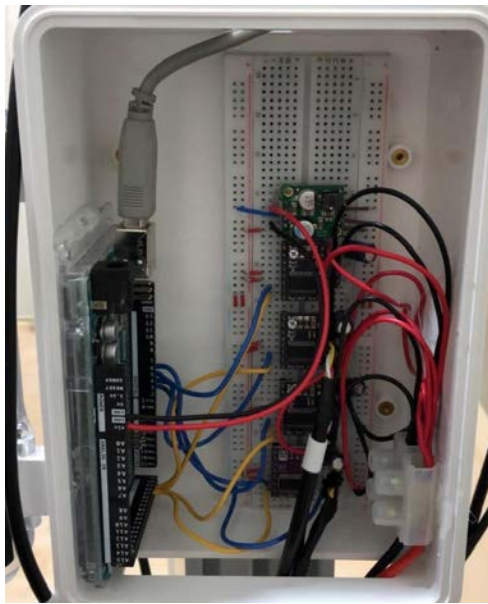
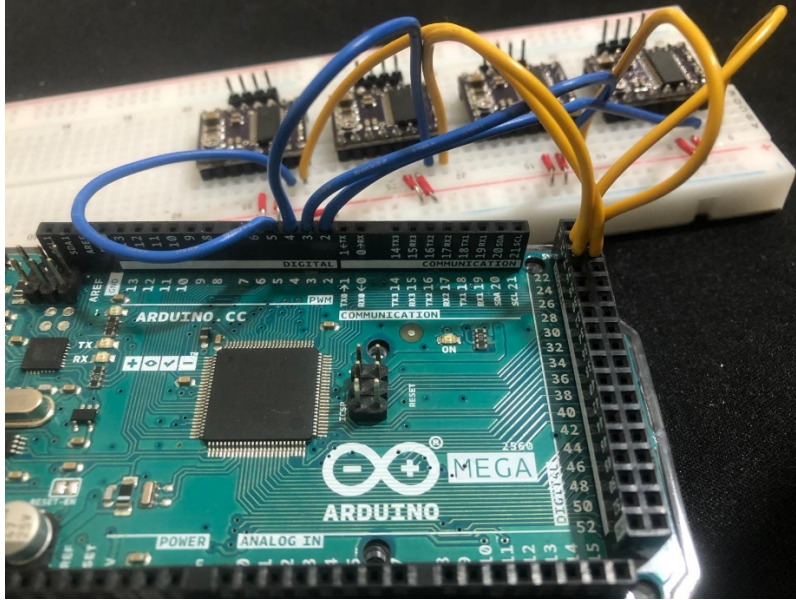


Figure 24 Circuit Diagram

4.2 Hardware Implementation

The Z axis of the prototype is the head of the project which responsible of planting process and considered the most complicated part in the project design. The Z Axis it includes one stepper motor. The Z axis motor is responsible for moving the arm vertically up and down. The Z Axis is attached to the connector and includes the Shaft, aluminum board which holding the track where X axis moves. There are 2X Axis X1 and X2 and each one has stepper motors; they move together

in specific direction to avoid interference. The Y Axis it includes one stepper motor and move horizontally, right and left. The following figures illustrate X, Y, and Z axis.



Figure 25 Stepper Motor (Z axis)

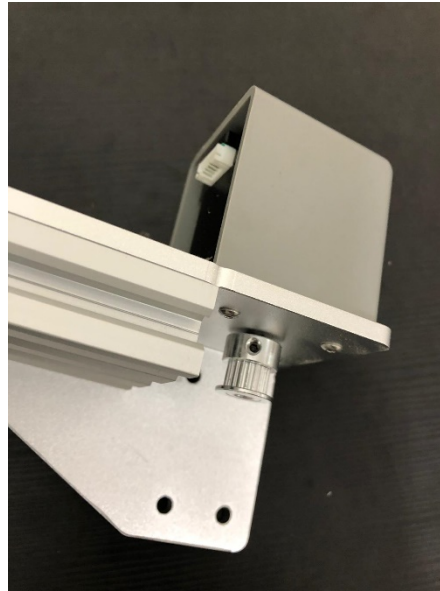


Figure 26 Stepper Motor (right X axis)

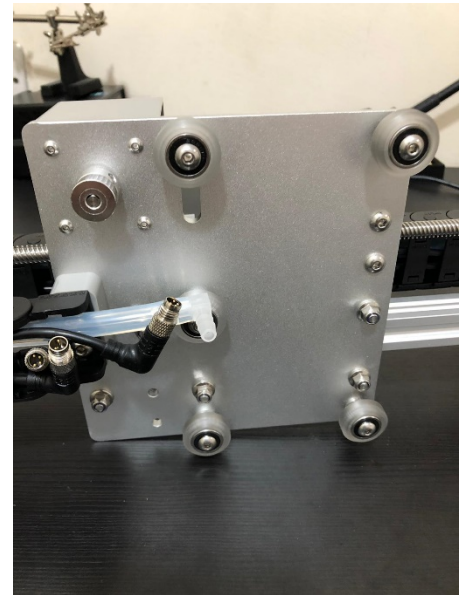


Figure 27 System Y Axis Track Movement with Stepper Motor



Figure 28 Z Axis Track

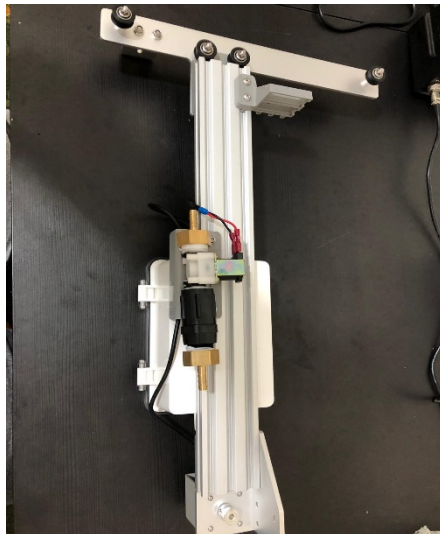


Figure 29 Left X Axis Track

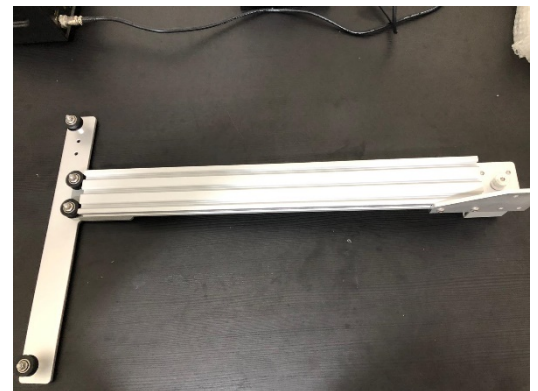


Figure 30 Right X Axis Track

As illustrated in figure25, the circuit diagram includes 4 stepper motor drivers, 2 drivers for X Axis, one for Y axis, and last one for Z axis. Each driver connected to certain wires and all these 4 wires connected to the pins as described in the datasheet. The circuit also contains the Voltage Regulator which is responsible to reduce the voltage from 24V into 5V and that is needed to run the Arduino and to operate the chips. But, for the stepper motor it normally takes 12V.

It's been used also the Connector to link both Positive represented with RED wires, and Negative with BLACK one, and connected to all pins. Each motor driver connected in the circuit has two connections from the Arduino microcontroller, one for STEP and one for DIR. The Y axis also connected to the motor through the connector which is responsible to move the prototype horizontally in Y axis.

Project has been implemented according to the requirements and functionalities that was defined at the initial stage of this project. Some changes applied during the project implementation, but the main idea completed successfully. Results were as expected, we connected all hardware components to the control unit represented with the Arduino MEGA microcontroller, connecting prototype circuits with stepper motors, and finally we wrote the required programming instructions using Arduino IDE, and uploaded these programs into the microcontroller board.

System prototype contains mainly three axes, X, Y, and Z, and as mentioned previously, the Z axis considered the main part of the project. the prototype includes stepper motors responsible to moving the arm in different direction, up, down, left, and right, attached to aluminum bridge as shown in figure 31.



Figure 31 System Y axis bridge aluminum

Prototype also includes Solenoid electronic valve with lift check water valve as shown in figure29, bearing movement tires, and three types of tubes, Tubes for water and for powering the motors, Tubes connecting with filters, and finally Tubes case for power and water tubes.

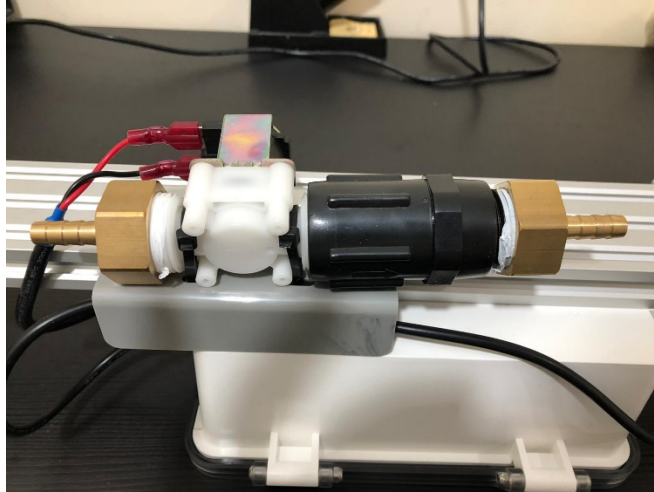


Figure 32 Solenoid electronic valve with lift check water valve

Figures 33, 34, and 35, show system prototype design after assembling hardware components including stepper motors, the three axis's, chain, Solenoid valve, tubes, main power supply, in addition to the main system electronic circuit connected with the Arduino MEGA microcontroller.



Figure 33 Prototype design - side view



Figure 34 Prototype Design - back side view

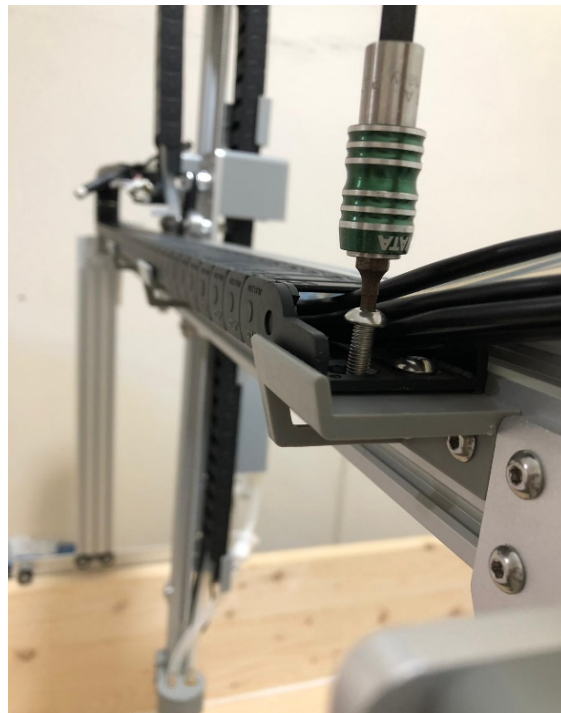


Figure 35 Arm with chain and axes track

The project prototype design has been divided mainly into 9 parts, each part has specific plant, and have 3 types of seeds, A, B, C. A on the left side, B in the middle, and finally C on the

right side. The motor start from a starting point defined in the system, take the seed, then move back to the starting point again then to be able to move into the targeted point where the seed should be planted, and so on until finishing all the seeds, considering the rotation between parts, and to do finally the irrigation process. Planting seeds and irrigation process steps are illustrated in the following figures.

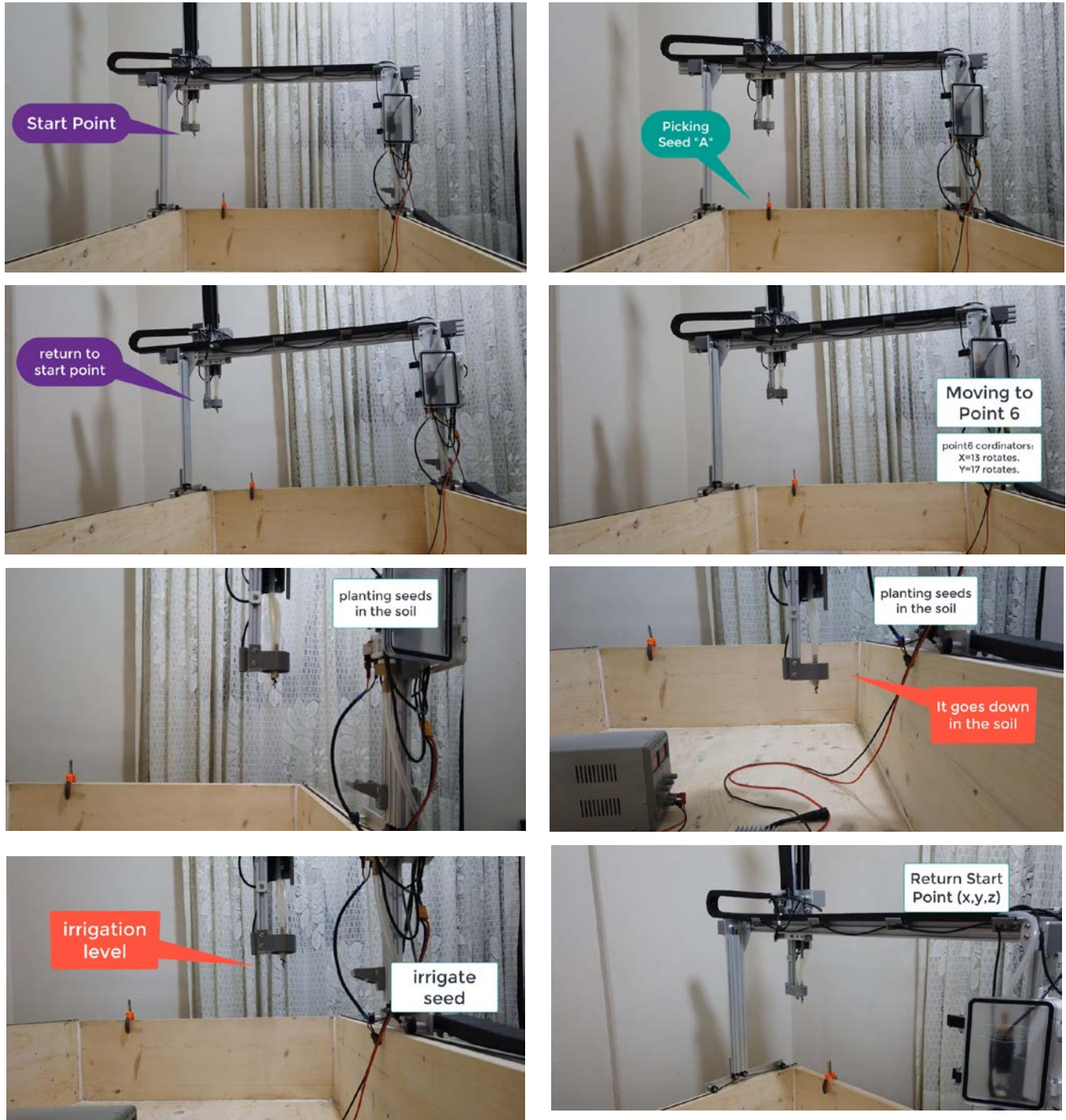


Figure 36 Planting seeds and irrigation process

Figure37 describes in details the rotations for all axes X, Y, and Z, which helped us to do the required calculation. As illustrated, numbers have been calculated based on measurement we conducted on the device movement from the starting point until end of the bed such as the X axis. During the testing of the project, we found that we need 25 rotation for X axis. So, which means to make the device move 25 rotation, we had to divided it on three sides since we have three rows with three columns. Finally, we figured out that we need the rotations to be 25 for X axis, 19 for Y axis, and 55 for Z axis.

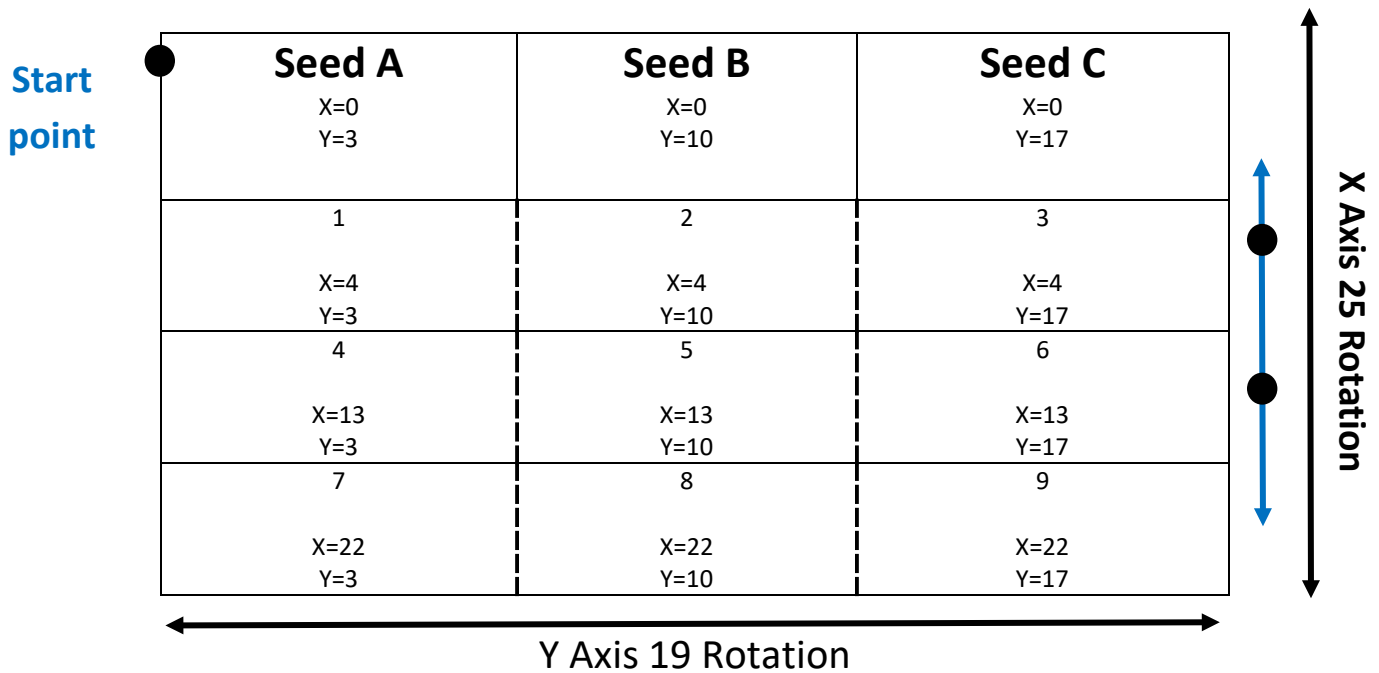


Figure 37 Three Axes Rotations

At the beginning, the device will move to point B which contains Rocca. Items order as C point contains Lettuce, and A contains Spinach. So, Z axis will move down to pick up number of seeds not more than 4 from point B using the air pump, and then will turn back to the starting point and move for example to point 4 as shown in figure36 where the device will plant the Rocca seeds. And here the air pump relay will turn on through the microcontroller so the device can keep holding the seeds.

Once the device's arm reached to a certain point in the soil, the relay will turn off the air pump will drop the seeds inside the soil and move back again to the irrigation position for watering

and turn on the water valve to spray water. finally, the Z axis will move back again to the starting point and move to pick seeds from position C, and follow the same process as was in B position.

Lettuce will be picked up from C position and move randomly to point 8 to planting the seeds and back again to irrigation position for watering purposes.

The below figure38 showing a full cycle of system function start from planting seeds into the soil, irrigation process, and showing the plants growing up in the prototype bed.



Figure 38 System Final Result

We found the suitable current for our project by using the Laboratory power supply, this device can supply from 0 to 30V (DC) and from 0 to 5 A, it is linear digital power supply. In addition, when we use this device, we found that our project needs a total current around 2 A to 2.3A. We know that our stepper motors need 12 V to operate, so we set the value of the voltage and we got the maximum value of the current as 0.75 A. We follow the same procedure to find the current for the pump and the valve.

Power consumption for stepper motor drains 0.75A maximum and supplied by 12V:

- $P_{\text{motor}} = V.I \rightarrow 12 * 0.75 = 9 \text{ WATT}$

We have 4 stepper motors then:

- $P_{\text{steppers}} = 4 * P_{\text{motor}} \rightarrow 36 \text{ WATT}$

$\rightarrow P_{\text{motors}} = 36 \text{ WATT}$

Power Consumption for Pump and valve:

- $P_{\text{pump}} = V.I_{\text{pump}} = 24 * 0.5 = 12 \text{ WATT}$
- $P_{\text{valve}} = V.I_{\text{valve}} = 24 * 0.16 = 3.84 \text{ WATT}$
- $P_{24\text{total}} = 15.84$

4.3 Software Implementation

In our project, we have one microcontroller used, the Arduino microcontroller which using micro-C programming language. So, the programming part for the Arduino microcontroller using micro-C programming under Arduino integrated development environment (IDE) as shown in figure39.

The Arduino Integrated Development Environment (IDE) is used to develop and write the code instructions required to implement the expected functionalities from other sensors and motors connected to the Arduino board.

To use the Arduino IDE, first step was connecting the microcontroller and open the IDE to select the Arduino type we will work on, which is Arduino MEGA in our project. Then, IDE will open a text editor for writing code, message area, text console, and toolbar with buttons for

common functions. The programs written on Arduino are called Sketches. These sketches output is with extension *ino*, and the IDE compiler is able to show errors and other information when running the program.

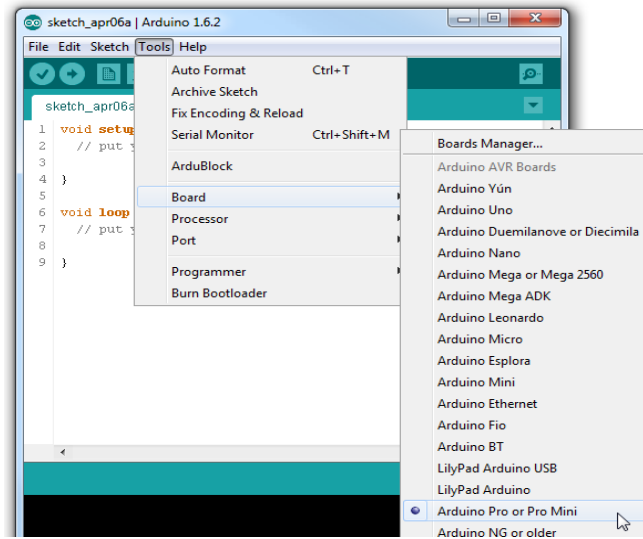


Figure 39 Arduino IDE

This IDE is used for all Arduino boards, and instructions written for the sensors, motors, or even for the connection setup and configuration between components and other controllers.

At the beginning, we define all libraries needed to deal with Arduino and motor drivers. We define the motor resolution as 200 STEPS, and then we define all Axis's RPM values X, Y, and Z with 120, also defined X, Y, and Z motors in the code body. In the body code, it's been also declared the *MultiDriver* library that takes the parameters of Stepper X and Y axes, to ensure that both are moving at the same time and to avoid any interference between them. But for the Z axis, it has been considered alone to not let them moving together.

Then, a loop method of type *Void* has been declared to loop the rotation infinitely and run moving of motors repeatedly according to expected behavior.

For testing, the moving of three axes motors has been tested well to assure the arm is moving properly in different directions accordingly. Each motor will move according to the specified axis, X and Y for the horizontal movement, and the Z axis for vertical. To test the operation of the system, we connected both Positive and Negative wires to the power supply through connector, and then we ran the system to assure the movement of all motors into the required directions.

4.4 IEEE Standards

1. To accept responsibility in making engineering decisions consistent with the safety, health and welfare of the public, and to disclose promptly factors that might endanger the public or the environment.
2. To avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist.
3. To be honest and realistic in stating claims or estimates based on available data.
4. To reject bribery in all its forms.
5. To improve the understanding of technology, its appropriate application, and potential consequences.
6. To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations.
7. To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others.
8. To treat fairly all persons regardless of such factors as race, religion, gender, disability, age, or national origin.
9. To avoid injuring others, their property, reputation, or employment by false or malicious action.
10. To assist colleagues and co-workers in their professional development and to support them in following this code of ethics.

CHAPTER5: EVALUATION

This section clarifies the prioritization for the main functional requirements of the system. The following Table3 listing the main functions of the system and the priority for each function to be considered in the project planning and implementation, so the development team take in consideration which functions should be released first.

Table 3 Functional Requirements Prioritization

Functional Requirements	Priority	Completion
Design system prototype	HIGH	Partially
Create main electrical circuit	HIGH	NO
Create full integration circuit of the Arduino controller	HIGH	NO
Planting process	HIGH	NO
Watering the plants	HIGH	NO
Check soil moisture	HIGH	NO
Final testing	HIGH	NO

5.1 Project Survey

Following is a list of the survey questions that posed to a segment of people selected randomly and their responses for each question. The first question asked was about the gender. The results as shown in the following figure were 62.07% Female and 37.93% Male.

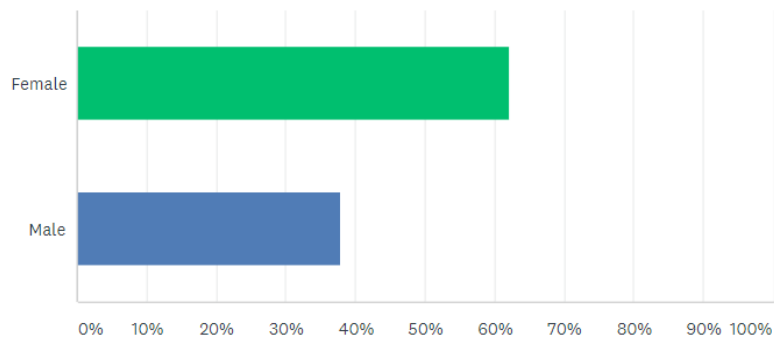


Figure 40 Survey Question1

ANSWER CHOICES	RESPONSES	
Female	62.07%	18
Male	37.93%	11
TOTAL		29

The second question was about the age group of people. 29 people answered this question, 3.45% under 18 years, 17.24% between 18 to 20 years, 68.97% between 21 to 30 years, 3.45% between 31 to 40, and 6.90% with 41 years or older as shown in the following figure.

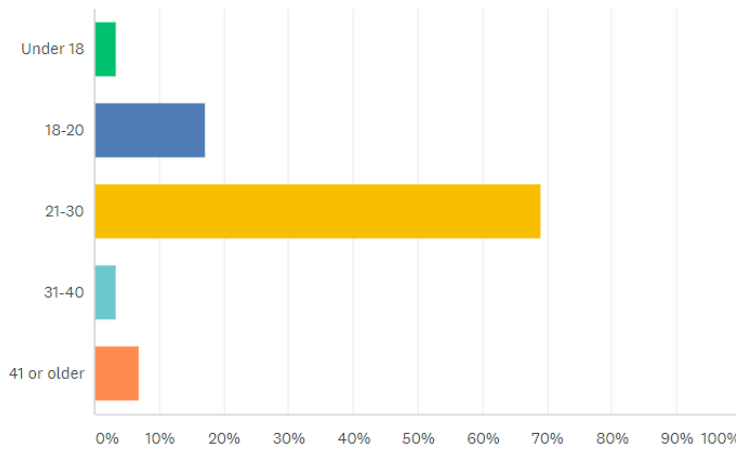


Figure 41 Survey Question2

ANSWER CHOICES	RESPONSES	
Under 18	3.45%	1
18-20	17.24%	5
21-30	68.97%	20
31-40	3.45%	1
41 or older	6.90%	2
TOTAL		29

The third question as shown below was about if they have a private garden at home or not. Results were that 58.62% of people with Yes, and 41.38% with No they don't.

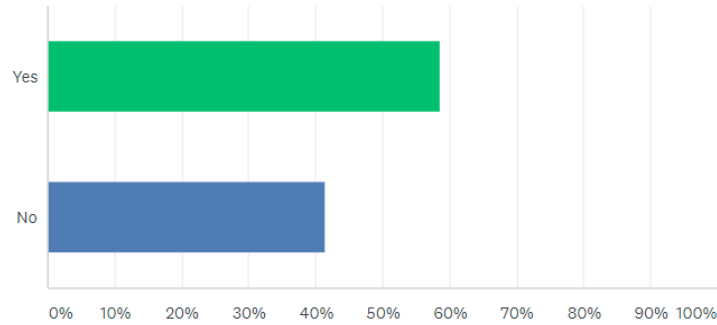


Figure 42 Survey Question3

ANSWER CHOICES	RESPONSES	
Yes	58.62%	17
No	41.38%	12
TOTAL		29

Then, the fourth question was if they usually have a space time during the day to follow up and monitoring the garden progress and keep watering it according to the standards. Answers were only 17.86% Yes, they do have, and most people with 82.14% No they don't have enough time space during the day to do such a task as shown below.

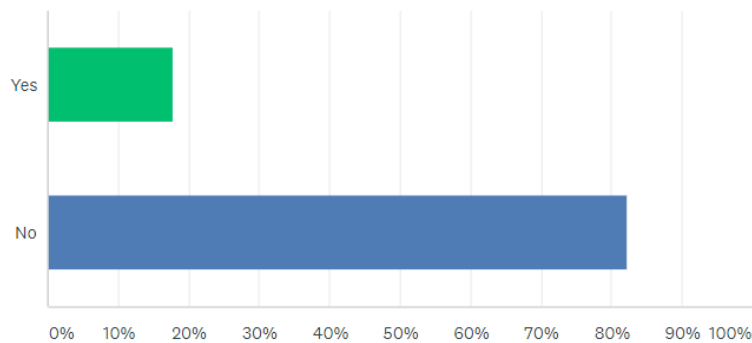


Figure 43 Survey Question4

ANSWER CHOICES	RESPONSES	
Yes	17.86%	5
No	82.14%	23
TOTAL		28

Question 5 as below was if they are interested and a good idea to have a gardening system which can planting and perform watering for different kind of plants. This question was targeting of the proposed solution will encourage them to have a garden at home or not. The results were encouraging where 96.55% with Yes, they do, and only 3.45% with No, they don't.

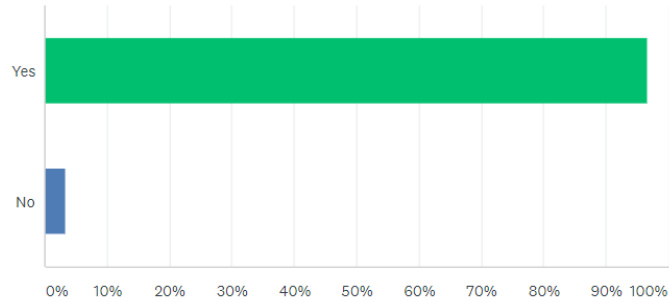


Figure 44 Survey Question5

ANSWER CHOICES	RESPONSES	
Yes	96.55%	28
No	3.45%	1
TOTAL		29

Then, as shown below question number 6 that if this device became available in the market, would they buy it or not? The answers also were matching exactly the results in the previous question, 96.55% with Yes and only 3.45% with No, they don't.

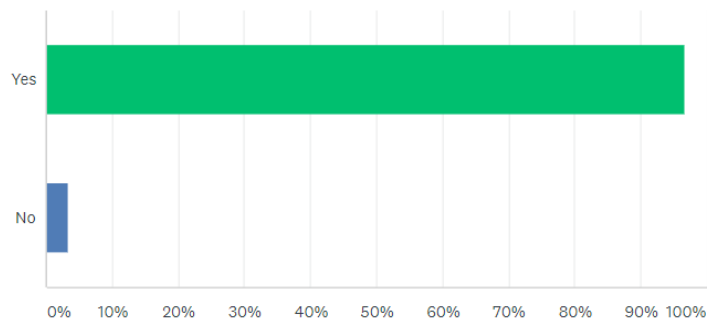


Figure 45 Survey Question6

ANSWER CHOICES	RESPONSES	
Yes	96.55%	28
No	3.45%	1
TOTAL		29

As described in the figure below, question number 7 was if they think that this device will be saving time and money instead of going to the supermarket to buy some vegetables. The results were also encouraging with 82.14% of Yes and only 17.86% of No.

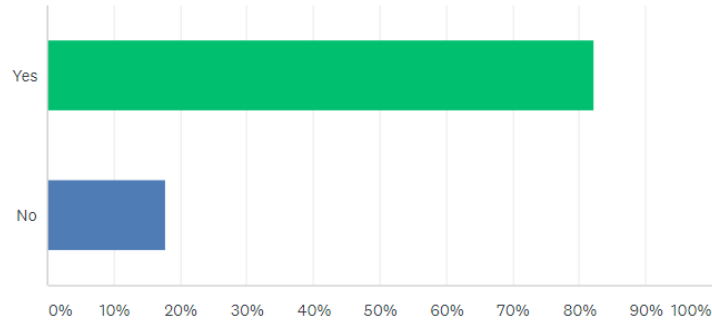


Figure 46 Survey Question7

ANSWER CHOICES	RESPONSES	
Yes	82.14%	23
No	17.86%	5
TOTAL		28

The final question 8 as show in the figure below was what is the maximum price they would pay for this device? Answers were according to different categories as 500, 1000, 1500, and 2000 KWD. 55.17% were with 500 KWD, 37.93% will pay 1000, only 6.90% will pay 1500, and no one with 0% will pay 2000.

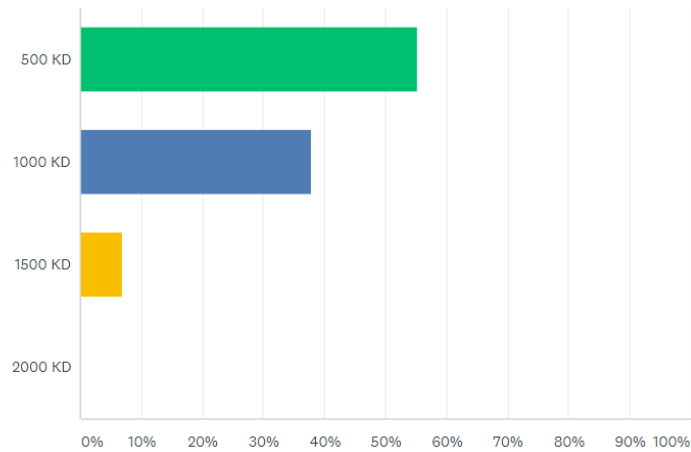


Figure 47 Survey Question8

ANSWER CHOICES	RESPONSES	
500 KD	55.17%	16
1000 KD	37.93%	11
1500 KD	6.90%	2
2000 KD	0.00%	0
TOTAL		29

CHAPTER6: CONCLUSION & FUTURE WORKS

6.1 Conclusion

Smart Gardening System using the proper control unit such as Raspberry Pi controller or Arduino board which will help to ease the most tedious job and tasks of gardening for farmers and plants lovers who always looking to care about their plants and stay updated with their progress. This design aims to monitor various garden parameters, perform watering process.

This system also helps to solve many issues occurring in the existing plant watering and gardening system. Through the soil moisture sensor to detecting the humidity level, system will decide the need to run irrigation process, so system helps to save water which means utilization for bills and save water sources. Plant monitoring and smart gardening proposed in this design with the help of microcontroller and proper sensors such as soil moisture will bring more convenience and comfort to people's lives for taking care of their garden.

The microcontroller that has been used in the system Arduino MEGA which will provide an economic and efficient platform to implement and perform the desired functions such as plant monitoring and smart gardening operations. The main advantage of the smart gardening system is that the user can manage the gardening process easily and can be updated with the status of the plants whenever necessary.

6.2 Future Works

The concept of plants monitoring, and smart gardening is far more than observing crops or just manage watering process as clarified in the project definition. The future works of the smart gardening that we are planning to perform are using the proper components that fit to the project requirements and which will make a significant change in the way of traditional planting process, Monitor the environment of the garden will be available for the user through a developed application displaying all needed information and notifications by using IOT.

The system as planned will be able to monitor, observe, planting, watering, and notify users with the required status information. At the next stage of the project, which is the implementation, we may stick to project requirements that already clarified in this paper, or we may exceed the limits and thinking to add more valuable features, such as develop mobile application and monitor

the plants and the whole garden through live camera in addition to control and do the required action through the mobile app. Also, monitoring the plant when get ripened and ready to be picked up, in addition to notify the user with the status of the plants in the garden. This might be as future work of the project, so farmer or garden owner can stay connected to the garden and the plants as if farmer physically exist.

Reference

- [1] A. P and P. R, "Design and implementation of automatic plant watering system, "International Journal of Advanced Engineering and Global Technology, vol. 04, no. 01, Jan. 2016.
- [2] J. Zhou, T. Leppanen, E. Harjula, M. Ylianttila, T. Ojala, C. Yu, H. Jin. Cloudthings: A common architecture for integrating the internet of things with cloud computing. In CSCWD, 2013. IEEE.
- [3] F. Taqi, F. Al-Langawi, H. Abdulraheem, M. AlAbd, "A Cherry-Tomato Harvesting Robot". Electrical and Computer Engineering Department American University of Kuwait, Kuwait. July, 2017.
- [4] M. AlAbd, S. AlDosari, B. AlMesbah, B. AlBeeshi, "iPlant: The Greenhouse Robot". Department of Electrical and Computer Engineering American University of Kuwait, Kuwait. May, 2015.
- [5] Mithya V, Aishwarya M, Gayathri S, Mahalakshmi L S, Pavithra S, "Smart Gardening System". International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-5S March, 2019.
- [6] Ramkumar.E, Nagarani.S, Roger Rozario A. P, Arjuman Banu.S, "Tomen:A Plant monitoring and smart gardening". Sri Ramakrishna Institute of Technology, Coimbatore. March 2018.
- [7] Correll N., Bollini M., Charrow B., Johnson L., "Indoor Robot Gardening: Design and Implementation". Article of Intelligent Service Robotics, University of Colorado Boulder, Department of Computer Science (CS). Published on October 2010.
- [8] Anitson TT, Saji J., Dubey R., Saravanakumar K., "Food Computer Automated Gardening System". Department of Computer Science, Christ University, Bangalore, India. Published on March 2, 2017.
- [9] Baum, F., MacDougall, C., & Smith, D. (2006). Participatory action research. *Journal of Epidemiology and Community Health*, 60(10), 854–857.
<https://doi.org/10.1136/jech.2004.028662>

- [10] Chen, Joy Iong Zong, Yuan-Chen Chen, and Shien-Dou Chung. "Implementation of a Greenhouse Crop Remote Monitoring System with IOT Technology."
- [11] Gopinath Shanmuga Sundaram, Bhanuprasad Patibandala and Harish Santhanam, "Bluetooth communication Using a Touchscreen Interface with the Raspberry Pi", Southeast on, 2013 Proceedings of IEEE Phil. vol., pp. 1-4, April 2013
- [12] Kidd, CoryD. et, al. "The Aware Home: A Living Laboratory for Ubiquitous Computing Research". Springer Berlin Heidelberg (1999)
- [13] J. Zhou, T. Leppanen, E. Harjula, M. Ylianttila, T. Ojala, C. Yu, and H. Jin (2013). Cloudthings: A common architecture for integrating the internet of things with cloud computing. In CSCWD, 2013. IEEE
- [14] William Issac , Abdullah Na, "On-the-go Soil Nitrogen Sensor Based on Near Infrared Spectroscopy".
- [15] S. V. Devika, S. Khamuruddeen, S. Khamurunnisa, J. Thota, and K. Shaik, "Arduino based automatic plant watering system," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 4, no. 10, pp. 449–456, 2014. [Online]. Available: http://www.ijarcsse.com/docs/papers/Volume_4/10_October2014/V4I10-0228.pdf. Accessed: Feb. 15, 2017
- [16] A. P and P. R, "Design and implementation of automatic plant watering system," International Journal of Advanced Engineering and Global Technology, vol. 04, no. 01, Jan. 2016.
- [17] A. Vashista, H. Rathore, and G. Jain, "Automatic gardening system using Arduino," SSRG International Journal of Electronics and Communication Engineering, vol. 3, no. 8, 2016. [Online]. Available: <http://www.internationaljournalssrg.org/IJECE/2016/Volume3-Issue8/IJECE-V3I8P130.pdf>. Accessed: Feb. 15, 2017
- [18] Farmbot. Available from: <https://farm.bot/>

Appendix

Project Source Code

```
#include <Arduino.h>

#include "BasicStepperDriver.h"

#include "MultiDriver.h"

#include "SyncDriver.h"

long mydeg = 360;

// Motor steps per revolution. Most steppers are 200 steps or 1.8 degrees/step

#define MOTOR_STEPS 200

// Target RPM for X axis motor

#define MOTOR_X_RPM 120

// Target RPM for Y axis motor

#define MOTOR_Y_RPM 120

// Target RPM for Z axis motor

#define MOTOR_Z_RPM 120

// X motors

#define DIR_X1 23

#define STEP_X1 3

#define DIR_X2 24

#define STEP_X2 4
```

```

// Y motor

#define DIR_Y 25

#define STEP_Y 5

// Z motor

#define DIR_Z 26

#define STEP_Z 6

#define MICROSTEPS 1

#define p_delay 3000

#define s_delay 3000

#define Air_Pump 27

#define Valve 28

int Moist;

BasicStepperDriver stepperX1(MOTOR_STEPS, DIR_X1, STEP_X1);

BasicStepperDriver stepperX2(MOTOR_STEPS, DIR_X2, STEP_X2);

BasicStepperDriver stepperY(MOTOR_STEPS, DIR_Y, STEP_Y);

BasicStepperDriver stepperZ(MOTOR_STEPS, DIR_Z, STEP_Z);

MultiDriver MotorXY(stepperX1, stepperX2, stepperY);

void setup() {

    pinMode(Air_Pump, OUTPUT);

    pinMode(Valve, OUTPUT);

```

```
digitalWrite(Air_Pump, HIGH);

digitalWrite(Valve, HIGH);

stepperX1.begin(MOTOR_X_RPM, MICROSTEPS);

stepperX2.begin(MOTOR_X_RPM, MICROSTEPS);

stepperY.begin(MOTOR_Y_RPM, MICROSTEPS);

stepperZ.begin(MOTOR_Z_RPM, MICROSTEPS);

delay(15000);

}

void loop() {

  point_A();

  Pick_Seeds();

  A_home();

  delay(500);

  point_1();

  Plant_Seeds();

  irrigate();

  home_1();

  delay(500);

  point_B();

  Pick_Seeds();
```

```
B_home();  
  
delay(500);  
  
point_2();  
  
Plant_Seeds();  
  
irrigate();  
  
home_2();  
  
delay(500);  
  
point_C();  
  
Pick_Seeds();  
  
C_home();  
  
delay(500);  
  
point_3();  
  
Plant_Seeds();  
  
irrigate();  
  
home_3();  
  
delay(500);  
  
point_4();  
  
MoisCheck();  
  
if (Moist < 300) {  
  
    irrigate();
```

```
}  
  
home_4;  
  
delay(500);  
  
point_5();  
  
MoisCheck();  
  
if (Moist < 300) {  
    irrigate();  
}  
  
home_5;  
  
delay(500);  
  
point_6();  
  
MoisCheck();  
  
if (Moist < 300) {  
    irrigate();  
}  
  
home_6;  
  
delay(500);  
  
  
  
point_7();
```

```
MoisCheck();

if (Moist < 300) {

    irrigate();

}

home_7;

delay(500);

point_8();

MoisCheck();

if (Moist < 300) {

    irrigate();

}

home_8;

delay(500);

point_9();

MoisCheck();

if (Moist < 300) {

    irrigate();

}

home_9;

delay(10000);
```

```
}  
  
void point_1() {  
  
    MotorXY.rotate(360 * 4, -360 * 4, 360 * 3);  
  
    delay(p_delay);  
  
}  
  
void home_1() {  
  
    MotorXY.rotate(-360 * 4, 360 * 4, -360 * 3);  
  
}  
  
void point_2() {  
  
    MotorXY.rotate(360 * 4, -360 * 4, 360 * 10);  
  
    delay(p_delay);  
  
}  
  
void home_2() {  
  
    MotorXY.rotate(-360 * 4, 360 * 4, -360 * 10);  
  
}  
  
void point_3() {  
  
    MotorXY.rotate(360 * 4, -360 * 4, 360 * 17);  
  
    delay(p_delay);  
  
}  
  
void home_3() {
```

```
MotorXY.rotate(-360 * 4, 360 * 4, -360 * 17);

}

////////////////////////////////////

void point_4() {

    MotorXY.rotate(360 * 13, -360 * 13, 360 * 3);

    delay(p_delay);

}

void home_4() {

    MotorXY.rotate(-360 * 13, 360 * 13, -360 * 3);

}

void point_5() {

    MotorXY.rotate(360 * 13, -360 * 13, 360 * 10);

    delay(p_delay);

}

void home_5() {

    MotorXY.rotate(-360 * 13, 360 * 13, -360 * 10);

}

void point_6() {

    MotorXY.rotate(360 * 13, -360 * 13, 360 * 17);
```

```
    delay(p_delay);
}

void home_6() {

    MotorXY.rotate(-360 * 13, 360 * 13, -360 * 17);

}

////////////////////////////////////

void point_7() {

    MotorXY.rotate(360 * 22, -360 * 22, 360 * 3);

    delay(p_delay);

}

void home_7() {

    MotorXY.rotate(-360 * 22, 360 * 22, -360 * 3);

}

void point_8() {

    MotorXY.rotate(360 * 22, -360 * 22, 360 * 10);

    delay(p_delay);

}

void home_8() {

    MotorXY.rotate(-360 * 22, 360 * 22, -360 * 10);

}
```

```
void point_9() {  
  
    MotorXY.rotate(360 * 22, -360 * 22, 360 * 17);  
  
    delay(p_delay);  
  
}  
  
void home_9() {  
  
    MotorXY.rotate(-360 * 22, 360 * 22, -360 * 17);  
  
}  
  
/////////  
  
void point_A() {  
  
    MotorXY.rotate(360 * 0, -360 * 0, 360 * 3);  
  
    delay(p_delay);  
  
}  
  
void A_home() {  
  
    MotorXY.rotate(-360 * 0, 360 * 0, -360 * 3);  
  
}  
  
void point_B() {  
  
    MotorXY.rotate(360 * 0, -360 * 0, 360 * 10);  
  
    delay(p_delay);  
  
}
```

```
void B_home() {  
  
    MotorXY.rotate(-360 * 0, 360 * 0, -360 * 10);  
  
}
```

```
void point_C() {  
  
    MotorXY.rotate(360 * 0, -360 * 0, 360 * 17);  
  
    delay(p_delay);  
  
}
```

```
void C_home() {  
  
    MotorXY.rotate(-360 * 0, 360 * 0, -360 * 17);  
  
}
```

```
////////////////////////////////////
```

```
void Pick_Seeds() {  
  
    stepperZ.rotate(360 * 38);  
  
    digitalWrite(Air_Pump, LOW);  
  
    delay(s_delay);  
  
    stepperZ.rotate(-360 * 38);  
  
}
```

```
void Plant_Seeds() {  
  
    stepperZ.rotate(360 * 55);
```

```
digitalWrite(Air_Pump, HIGH);

delay(s_delay);

stepperZ.rotate(-360 * 55);

}

void irrigate() {

    stepperZ.rotate(360 * 27);

    digitalWrite(Valve, LOW);

    delay(1000);

    digitalWrite(Valve, HIGH);

    stepperZ.rotate(-360 * 27);

}

void MoisCheck() {

    stepperZ.rotate(360 * 50);

    delay(5000);

    Moist = analogRead(A0);

    delay(500);

    stepperZ.rotate(-360 * 50);

}
```