Politecnico di Milano

Dipartimento di Elettronica, Informazione e Bioingegneria

Master of Science in Automation and Control Engineering

Automated Smart Indoor Farming using IoT

Supervisor

Prof. Marco MARCON

Candidate

Shahrukh Ahmed IRFANI – 10559842

Academic Year 2019 – 2020

Acknowledgements

I would like to express my sincere gratitude to my supervisor Professor Marco Marcon at Politecnico di Milano University for the continuous support of my master thesis research project, for his patience, motivation, enthusiasm, and immense knowledge. It would have been difficult to achieve my goals without my supervisor's constant feedback.

Secondly, I cannot thank enough to Politecnico di Milano University for giving me an opportunity to experience this roller coaster ride of master's degree in Automation and Control Engineering. Furthermore, I am grateful to all the professors who taught me various subjects of my academic curriculum and lab supervisors who gave me immense knowledge of the subjects in both practical and theoretical aspects.

To conclude, I would like to thank my family members Khursheedunnisa Begum, Zaheda Begum, Vasiur Rehman, Khamarunnisa Begum and Natasha Amjad for their support, motivation, and prayers throughout the journey of my life. Finally, I am extremely thankful and blessed to have my only best friend and my dear brother Mohammed Shahbaz Irfani, for helping me overcome every milestone.

Sommario

L'Internet delle Cose e gli *smart device* che generano e raccolgono dati tramite sensori per inviarli al Cloud fanno parte della vita degli utenti in molti contesti, tra cui la gestione agricola intelligente (*smart farming* e l*'intelligent farm)*. Questo elevato volume di dati viene archiviato ed elaborato nel Cloud allo scopo di ottenere conoscenze e informazioni preziose per le diverse organizzazioni. Il concetto di *coltivazione indoor* è molto recente e sta crescendo in un vasto mercato multimilionario. Con l'avvento delle tecniche di apprendimento automatico (machine learning), la pre-elaborazione dei dati e le capacità decisionali possono essere ampiamente automatizzate: un'idea simile viene proposta per coltivare specifiche erbe e piante utilizzando condizioni di crescita automatizzate quali l'intervallo di umidità del suolo, i livelli di intensità della luce e la miscela di acqua e nutrienti oltre alla frequenza di pompaggio.

Il progetto inizialmente si concentra sul confronto delle tecnologie esistenti, valide sia da un punto di vista scientifico che commerciale. Una revisione dettagliata dello stato dell'arte identifica i problemi chiave della ricerca e il progetto si concentra principalmente sulla progettazione e sull'implementazione di un sistema di agricoltura indoor intelligente automatizzato tramite interfaccia cloud opensource che utilizzando la piattaforma ThingsSpeak IoT. Il sistema è interamente gestito utilizzando un computer a scheda singola, Raspberry Pi Model 3B + con sistema operativo Raspbian Linux. Una suite di sensori di umidità del suolo composta da sensori resistivi e capacitivi, in aggiunta al sensore di temperatura e umidità forniscono misurazioni accurate dei dati che vengono inviati al cloud in tempo reale. L'algoritmo implementato utilizzando il linguaggio di programmazione Python gestisce l'automazione del sistema e i segnali logici intelligenti vengono trasmessi ai relè per l'irrigazione istantanea e alle luci utilizzate per la crescita delle piante. Il sistema IoT proposto automatizza il ciclo di crescita delle piante (quali colture o erbe aromatiche) in base a specifiche condizioni di crescita (temperatura, umidità e livelli di intensità luminosa) di quella particolare pianta senza intervento manuale da parte degli utenti.

Parole chiave: l'intelligent Farm, Agricoltura Indoor, Internet of Things, Raspberry pi, controllo in tempo reale, rilevamento in tempo reale, Thingspeak, Automazione, idroponica, aeroponica

Abstract

The Internet of Things and the smart devices that generate and collect data through sensors to send it to the Cloud are part of the life of users in many contexts, including smart farming and intelligent farm. This volume of data is stored and processed in the Cloud, with the purpose of obtaining knowledge and valuable information for organizations. The concept of Indoor farming is very new and is growing into a vast multi-million-dollar market. With the advent of machine learning techniques, data pre-processing and decision-making capabilities can be widely automated, a similar idea is proposed to grow custom user preferred herbs using automated growing conditions like soil moisture range, Light intensity levels and water-nutrient mixture pumping frequency.

The project initially focuses on comparing the existing technologies both scientifically and commercially viable. A detailed review of the state of the art identifies the key research problems and the project mainly focuses on design and implementation of an Automated smart indoor farming system with enhanced opensource cloud dashboard interface using ThingsSpeak IoT platform. The system is entirely operated using the Raspberry Pi Model 3B+ single board computer with Raspbian Linux operating system. A soil moisture sensor suite comprising of resistive and capacitive sensor along with the temperature and humidity sensor provide accurate data measurements which are sent over to the cloud in real time. The smart algorithm implemented using Python programming language handles the automation of the system and smart logic signals are conveyed to the relays for timely irrigation and switching the grow lights for plant growth. The proposed IoT system automates the growth cycle of the plants(crops, herbs) based on specific growing conditions(temperature, moisture and light intensity levels) of that particular plant without manual intervention from the users.

Keywords: Intelligent Farm, Indoor Farming, Internet of Things, Raspberry pi, Remote Control, Remote Sensing, Thingspeak, Automation, Hydroponics, Aeroponics

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The current situation of the world demands the urgent need for sustainable food production. Sustainability means the ability to be maintained at a certain growth level. According to recent statistics[40], the current food production is carried out on deforested lands, smothering crops with toxic pesticides, and killing natural habitat of the wildlife. Urbanization and Population growth further impacts the problem of food production. Furthermore, we must protect the welfare and financial security of farmers because farming is the backbone of country's economic growth. The problem of water scarcity is age old and every year countries are affected due to drought and shortage of water resources. In several parts of the world, Industrial wastes are released in the nearby areas which causes infertility in the soil. The right choice of soil and sufficient supply of water can lead to a healthy growth of plantlife. Indoor Farming is a method of growing crops or plants usually on a large scale entirely indoors. It can be further classified based on the choice of growing medium. Hydroponic systems as the name suggests uses water as the primary growing medium and Aeroponic systems utilizes water-nutrient mixture as the growing medium instead of soil. The main advantages of Indoor Farming over traditional farming methods are listed below:

Resource Optimization: Aeroponic Systems requires 95% less water as compared to traditional farming. This technology is a huge breakthrough as freshwater shortage is a huge problem and currently 1% of the worlds water is accessible for drinking. The future trends indicate that the water scarcity issue is going to worsen in the coming years.

Faster Supply Chain: Shorter suppy chains can tackle the program of food wastage. According to a recent survey[40], currently 30% of food waste is lost in the supply chain. This also promotes farmers to grow crops yearround which means they don't have to rely on buying in produce from the overseas.

Eliminating the usage of Fertlizers and Pesticides: The usage of fertilizers and pesticides have increased the production yield of crops but on the other hand is a major public health concern and also affects the natural habitat of insects and wildlife. Indoor Farming takes inspiration from organic farming eliminating the usage of toxic chemicals and growing crops in controlled environment.

Climate Control: Climate Change and natural disasters have caused a huge loss to the crop production worldwide every year. Unpredictable weather conditions and torrential rains have also devasted the crop yield and caused huge economic losses throughout the world. In controlled growing environment, the food productions can be consistent irrespective of changing weather conditions. The plantlife is subjected to artificial growing conditions like growing lights which can be powered through renewable energy sources.

1.1 Structure

The structure of the thesis includes:

- **Chapter 1 - Introduction** contains basics of Humidity Sensors, types and classifications based on working principle. It also lists down the pro and cons of using various sensors and sensor calibration methods based on water-soil characteristics. The chapter also discusses the possibility to integrate NIR Sensor which is the best in class sensor for applications involving measuring accurate moisture levels. The scientific analysis discussed later in this chapter helps use to the identify the key design problems and gives an insight to tackle the research problem.

- **Chapter 2 - Motivation** for the proposed thesis work and its background history. This chapter presents the problem statement and project description giving an initial understanding of the key concepts and basic ideas of the project.

- **Chapter 3 - System Design and Implementation** is presented with a high level emphasis on Hardware and Software Architecture. This chapter also discuss about the design elements of the system which are the hardware components and the software tools used in the project.

- **Chapter 4 - System testing and Performance Evaluation** is the main focus of this Chapter. The experimental prototype of the system is illustrated and key functionalities are presented. The system test results based on the desired expectations have been described by extensive testing under various test cases. The graphical visualizations of the sensor data is shown and inferences have been outlined by visual inspections. Finally, the Energy costs and the overall cost analysis of the system are calculated which are the essential parameters in evaluating the system performance.

- **Chapter 5 - Future Scope and Conclusion**. This chapter concludes the thesis work by identifying the goals achieved and the future scope of the project.

- **Acronyms** - This section lists down the abbreviations used in this thesis work.

- **Appendix** - The Thesis project software code and the system configuration is documented in this section.

- **Bibliography -** This section is composed of the research citations and references used during the thesis work.

1.2 Humidity Sensors

Humidity Sensor (Hygrometer)[17] senses, measures and reports both moisture and air temperature. Relative Humidity (RH) is the ratio of partial pressure of water vapor to the equilibrium vapor pressure of water vapor at a given temperature.

Humidity sensors work by detecting changes that alter electrical currents or temperature in the medium.

1.2.1 Capacitive Humidity Sensor

A capacitive humidity sensor measures relative humidity by placing a thin strip of metal oxide between two electrodes. The metal oxide's electrical capacity changes with the relative humidity of the medium.

Working Principle

A simple Capacitive RH Sensor can be made from an air-filled capacitor as the moisture in the atmosphere changes its permittivity. But for practical applications, air as a dielectric is not feasible. Hence, the space between the capacitor plates is usually filled with an appropriate dielectric material (isolator), whose dielectric constant varies when it is subjected to change in humidity. The common method of constructing a capacitive RH sensor is to use a hygroscopic polymer film as dielectric and depositing two layers of electrodes on the either side. Figure 1.1 demonstrates the construction of Capacitive Humidity Sensors.

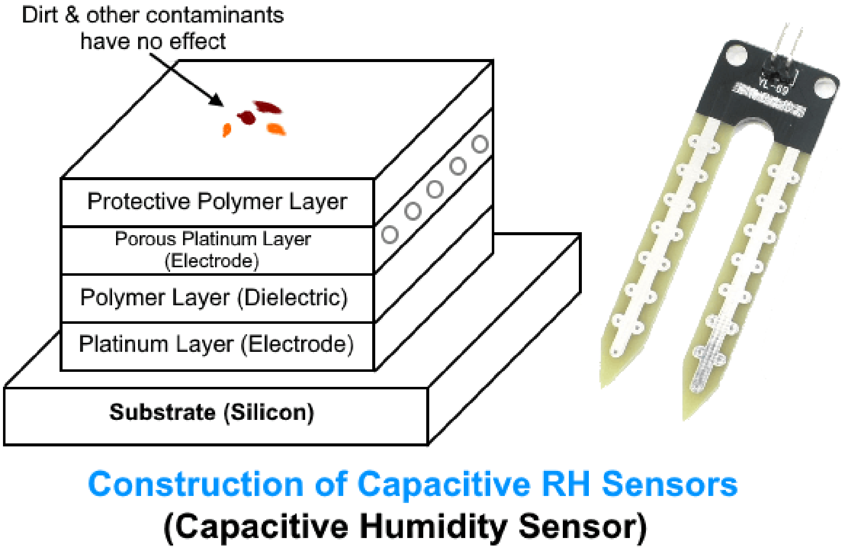

Dirt & other contaminants have no effect
Protective Polymer Layer
Porous Platinum Layer (Electrode)
Polymer Layer (Dielectric)
Platinum Layer (Electrode)
Substrate (Silicon)
Construction of Capacitive RH Sensors
(Capacitive Humidity Sensor)

Figure 1. 1 Capacitive RH sensors

Another way to use the capacitive RH sensors is to observe the changes in the frequency of the oscillator constructed using a capacitor with RH sensitive test subject as dielectric [17]. The test samples are placed between two plates (which form the capacitor electrodes) to form a capacitor in the LC Oscillator circuit. The frequency of the oscillator changes with humidity surrounding the test sample. The figure 1.2 illustrates the changes in frequency of oscillator when a test sample is placed between the capacitor plates.

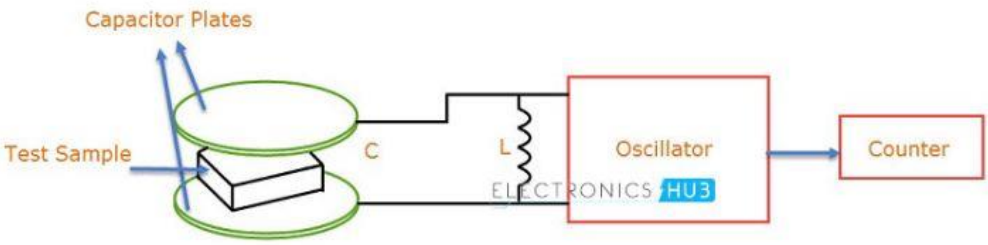


Figure 1. 2 RH sensitive test subject as dielectric to observe the changes in the frequency of oscillator.

The construction of a thin thermostat polymer film based capacitive RH Sensor is fabricated on a silicon substrate. On this substrate, two metal electrodes made of either aluminium, platinum or chromium are deposited. The shape of these electrodes is carved out such that, the electrodes form an interdigitated pattern. On top of this layer, a dielectric layer is deposited. Figure 1.3 shows a top and cross section view of the capacitive humidity sensor. **Note:** The two temperature sensitive resistors are deposited on the same substrate to provide temperature compensation.

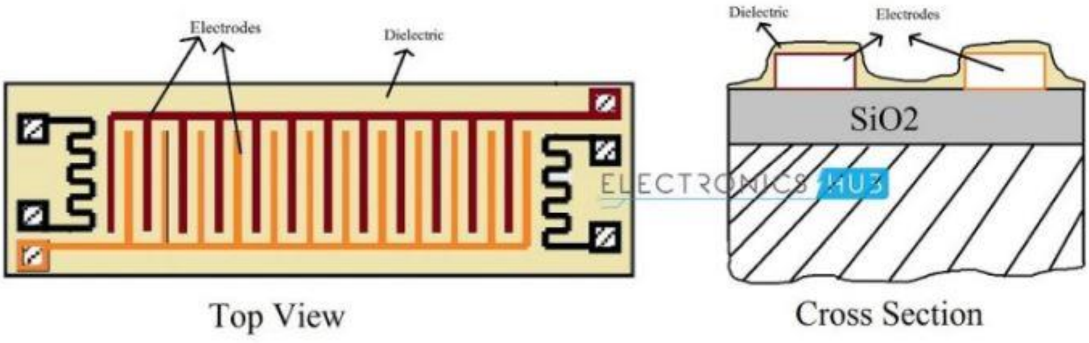


Figure 1. 3 Top and cross section view of capacitive humidity sensor.

Advantages

- The capacitive type sensors are linear and can measure relative humidity from 0% to 100%.
- These are the only types of full-range relative humidity measuring devices down to 0% relative humidity.
- This low temperature effect often leads them to being used over wide temperature ranges without active temperature compensation.
- With proper calibration, Capacitive type humidity sensors can yield precise process measurements.

Disadvantages

- The distance from the sensor and signaling circuit is very limited.

1.2.2 Resistive Humidity Sensors

Resistive Humidity Sensors utilize ions in salts to measure the electrical impedance of atoms. As humidity changes, the resistance of the electrode changes on either side of the salt medium.

Working Principle

The Resistive Humidity Sensor is usually made up of materials with relatively low resistivity and this resistivity changes significantly with changes in humidity. The relationship between resistance and humidity is inversely exponential. The low resistivity material is deposited on top of the two electrodes as depicted in figure 1.4.

The electrodes are placed in interdigitated pattern to increase the contact area. The resistivity between the electrodes changes when the top layer absorbs water, and this change can be measured with the help of a simple electrical circuit.

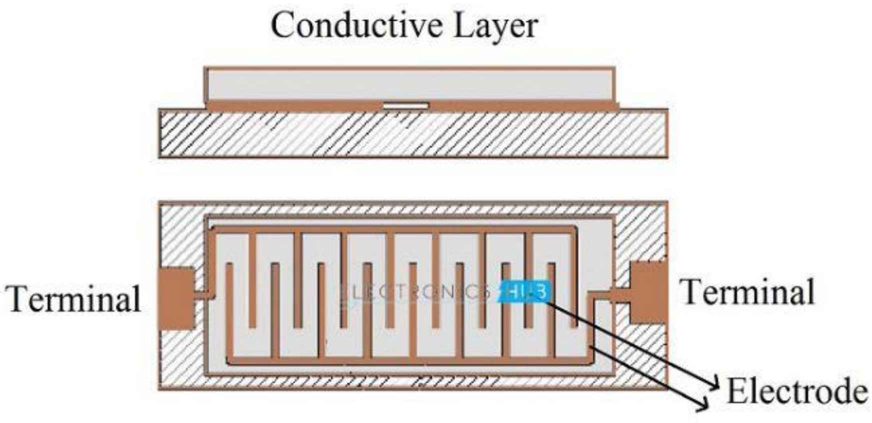


Figure 1. 4 Resistive Humidity Sensor

Some of the commonly used materials are salt, specially treated substrates, solid poly-electrolytes and conductive polymers. Modern Resistive Humidity Sensors are coated with ceramic substance to provide extra protection. The electrodes in the sensor are usually made of noble metals like gold, silver or platinum.

Advantages

- They are compact in size and are low cost.
- The distance between the sensor and signal circuit can be large (suitable for remote operations).
- They are highly interchangeable as there are no calibration standards.

Disadvantages

- Resistive Humidity Sensors are sensitive to chemical vapours and other contaminants.
- The output readings may shift if used with water soluble products.

1.2.3 Thermal Humidity Sensors

Thermal Conductivity Humidity Sensors are also known as Absolute Humidity (AH) Sensors as they measure the Absolute Humidity. Absolute Humidity (AH) is the ratio of mass of the water vapour to the volume of the air. If m is the mass of the water vapour and V is the total volume i.e. volume of air and water vapour mixture, then Absolute Humidity AH is given by

$$AH = m/V$$

Absolute Humidity doesn’t take temperature in to account but it changes with temperature and pressure. Thermal Conductivity Humidity Sensors measure the thermal conductivity of both dry air as well as air with water vapor. The difference between the individual thermal conductivities can be related to absolute humidity.

Working Principle

The best component to accomplish thermal conductivity-based humidity sensor is thermistor. Hence, two tiny thermistors with negative temperature coefficient are used to for a bridge circuit. In that way , one thermistor is hermetically sealed in a chamber filled with dry Nitrogen while the other is exposed to open environment through small venting holes. When the circuit is powered on, the resistance of the two thermistors is calculated and the difference between those two values is directly proportional to Absolute Humidity (AH).

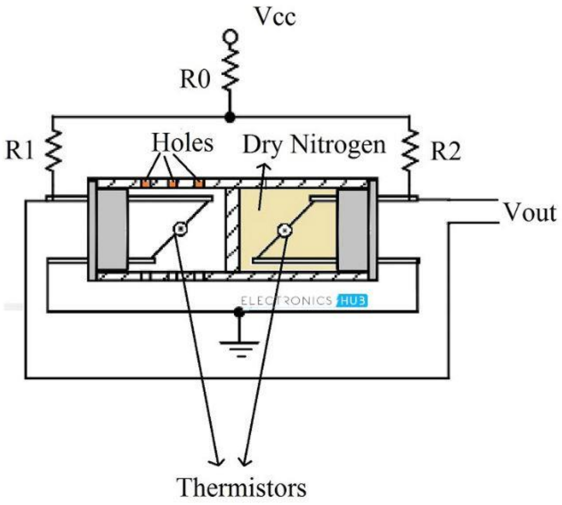


Figure 1. 5 Thermal Conductivity Humidity Sensor

Advantages

- It is suitable for high temperature environments and high corrosive stations.
- It is durable and has higher resolution compared to other types of Humidity sensors.

Disadvantages

- When exposed to any gas with thermal properties different than Nitrogen, it might affect the process measurements.

1.3 Sensor Parameters

Accuracy: Every sensor has its own calibration curve, based on a 9 point system. It basically pitches the pros against the cons of the particular sensor.

Linearity: It indicates the voltage deviation from the best fit-straight line (BFSL) value and the measured output Voltage converted to relative humidity.

Reliability: The measurements often cause the sensor to fall out of sync. However for a sensor applications, it should always provide reliable measurements.

Repeatability: The measurements from a sensor shouldn't drift apart. Repeatability is the measurement of drift among measurements of single quantity.

Response Time: Typically the time taken by a sensor to rise to 66%(rise time) or fall to 33%(fall time) of maximum output voltage is known as the response time.

1.4 Sensor Calibration Gravimetric Method

The measurement procedure for sensor calibration is as follows:

First, empty container is weighted. It is then filled with dry soil samples and weighted. Then, soak the container contraining the dry soil sample with water for 24 hours until the soil becomes saturated. The soil in the container is weighted once a day during 8 days data retrieval. Degradation of the soil weight in the container is defined as the water that evaporates within one day of experiment. Then, the soil moisture content is calculated with the following equation:

$$\boldsymbol{Soil\,moisture}(\%) = \frac{\text{Weight of water contained in the soil}}{\text{Weight of dry soil samples}}$$

This method is sometimes also used as a direct method used in measurement of soil-moisture content.

1.5 Water-Soil Characteristics

Water availability in the soil either as ground water or soil water is influenced by physical properties of the soil such a s structure, texture, porosity and fraction. These properties indirectly effect the soil-water holding capacity. The availability of soil-water can be measured by direct or indirect method. Measurement of soil-water level by direct method can be determined by using the mass difference of soil sample before and after dried. While determining the soil water using indirect method is usually conducted by using the calibrated variable which has relation to the water content.

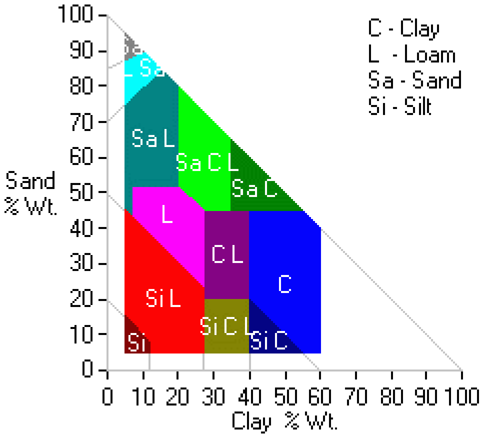


Figure 1. 6 Soil Texture Triangle (USDA Natural Resources Conservation Service)

SPAW model[18] developed by United States Department of Agriculture is a water budgeting tool for farm fields, ponds, and inundated wetlands. It is used to simulate soil water tension, conductivity and water holding capability based on the soil texture, with adjustments to account for gravel content, compaction, salinity, and organic matter. Soil texture triangle is a graphical tool to visualize and understand the meaning of soil texture names.

1.6 Indirect Method

It predicts the soil moisture content of the sample based on the change of soil-water electrical properties which varies according to the changes of soil validity.
Indirect measurement is divided into two methods:

- **Conductance Method:** A sample is placed between two electrodes. Electrical current is flowed between the two electrodes.

- **Capacitance Method:** The same sample is again placed between both electrodes. By using two parallel plates, which function as capacitor while the sample works as a dielectric material. The moisture content of the sample is predicted from the capacitance of the capacitor.

The sensor measures the amount of water in the soil by volume using the properties of capacitance. It rapidly applies voltage to the positive electrode, then removes it. This induces electromagnetic field in the soil, and the time it takes for the charge between the electrodes to stabilize is given as:

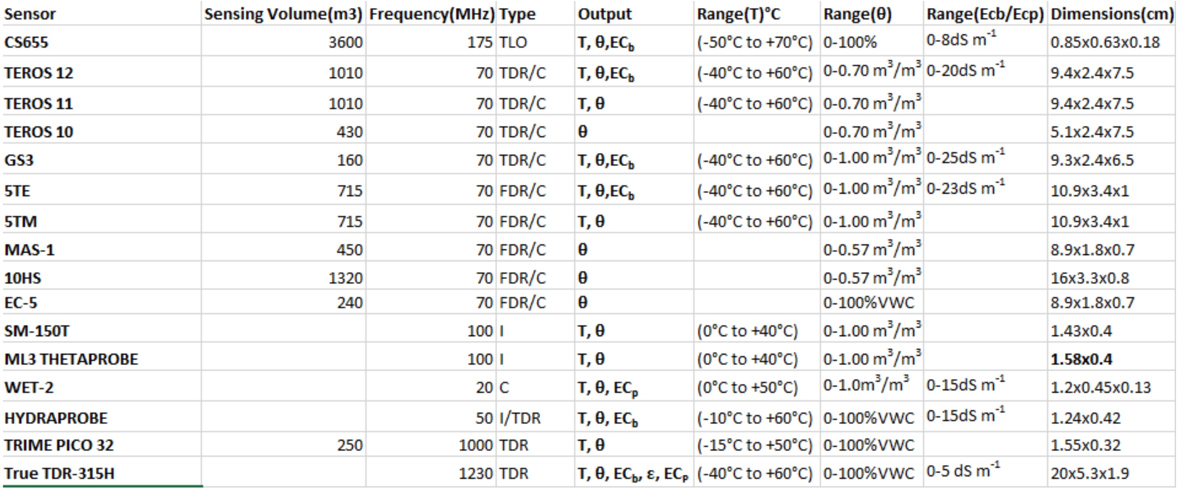
$$t = RC \ln \frac{\mathrm{V} - \mathrm{Vf}}{\mathrm{Vi} - \mathrm{Vf}}$$

R: Soil resistance
C: Capacitance
V: measured voltage at time instant t
Vi: Voltage at the beginning of the process
Vf: Applied Voltage

Since the capacitance of a capacitor is dependent on the relative dielectric permittivity of the material between the electrodes, varying this parameter will change the total capacitance and thus the time it takes to bring it up to charge. Water has a higher relative dielectric permittivity than minerals present in the soil, which changes the capacitance.

1.7 Commercial Soil Moisture Sensor

A few of the widely used commerically available moisture sensors are classified based on various parameters like sensing volume, frequency, type, measurement range and dimensions and listed down in the table 1.1



Sensor	Sensing Volume(m3)	Frequency(MHz)	Type	Output	Range(T)°C	Range(θ)	Range(Ecb/Ecp)	Dimensions(cm)
CS655	3600	175	TLO	T, θ,EC$_b$	(-50°C to +70°C)	0-100%	0-8dS m^{-1}	0.85x0.63x0.18
TEROS 12	1010	70	TDR/C	T, θ,EC$_b$	(-40°C to +60°C)	0-0.70 m^3/m^3	0-20dS m^{-1}	9.4x2.4x7.5
TEROS 11	1010	70	TDR/C	T, θ	(-40°C to +60°C)	0-0.70 m^3/m^3		9.4x2.4x7.5
TEROS 10	430	70	TDR/C	θ		0-0.70 m^3/m^3		5.1x2.4x7.5
GS3	160	70	TDR/C	T, θ,EC$_b$	(-40°C to +60°C)	0-1.00 m^3/m^3	0-25dS m^{-1}	9.3x2.4x6.5
5TE	715	70	FDR/C	T, θ,EC$_b$	(-40°C to +60°C)	0-1.00 m^3/m^3	0-23dS m^{-1}	10.9x3.4x1
5TM	715	70	FDR/C	T, θ	(-40°C to +60°C)	0-1.00 m^3/m^3		10.9x3.4x1
MAS-1	450	70	FDR/C	θ		0-0.57 m^3/m^3		8.9x1.8x0.7
10HS	1320	70	FDR/C	θ		0-0.57 m^3/m^3		16x3.3x0.8
EC-5	240	70	FDR/C	θ		0-100%VWC		8.9x1.8x0.7
SM-150T		100	I	T, θ	(0°C to +40°C)	0-1.00 m^3/m^3		1.43x0.4
ML3 THETAPROBE		100	I	T, θ	(0°C to +40°C)	0-1.00 m^3/m^3		**1.58x0.4**
WET-2		20	C	T, θ, EC$_p$	(0°C to +50°C)	0-1.0m^3/m^3	0-15dS m^{-1}	1.2x0.45x0.13
HYDRAPROBE		50	I/TDR	T, θ, EC$_b$	(-10°C to +60°C)	0-100%VWC	0-15dS m^{-1}	1.24x0.42
TRIME PICO 32	250	1000	TDR	T, θ	(-15°C to +50°C)	0-100%VWC		1.55x0.32
True TDR-315H		1230	TDR	T, θ, EC$_b$, ε, EC$_p$	(-40°C to +60°C)	0-100%VWC	0-5 dS m^{-1}	20x5.3x1.9

Table 1. 1 Commercial Humidity Sensor classification

All units of measurement in SI system. All the above sensors are in compliance with SDI-12 communication protocols.

ABBREVIATIONS:

EC$_b$	Bulk Electrical Conductivity
EC$_p$	Pore water Electrical Conductivity
TDR	Time Domain Reflectrometry
TLO	Transmission Line Oscillation
ε	real di-electric permittivity
I	Impedance
C	Capacitance
R	Resistance
T	Temperature
θ	Volumetric Water Content(VWC)

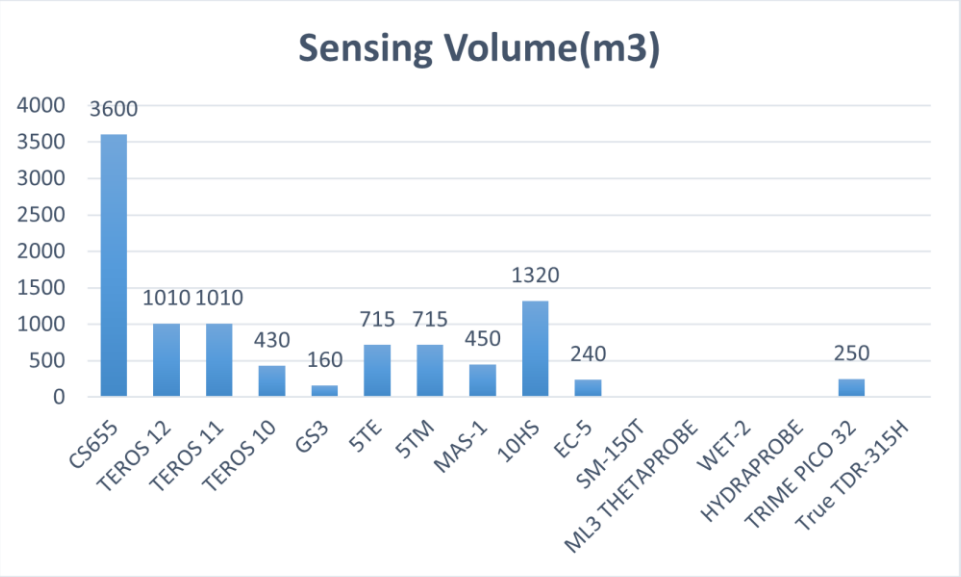


Figure 1. 7 A comparison of Sensing Volume(m3) vs commercially available sensor

Clearly, from figure 1.7 the campbell scientific CS655 is superior than other sensors interms of sensing volume. It is multiparameter smart sensor that uses innovative techniques to monitor soil volumetric-water content, bulk electrical conductivity, and temperature. It outputs an SDI-12 signal that many data loggers can measure.

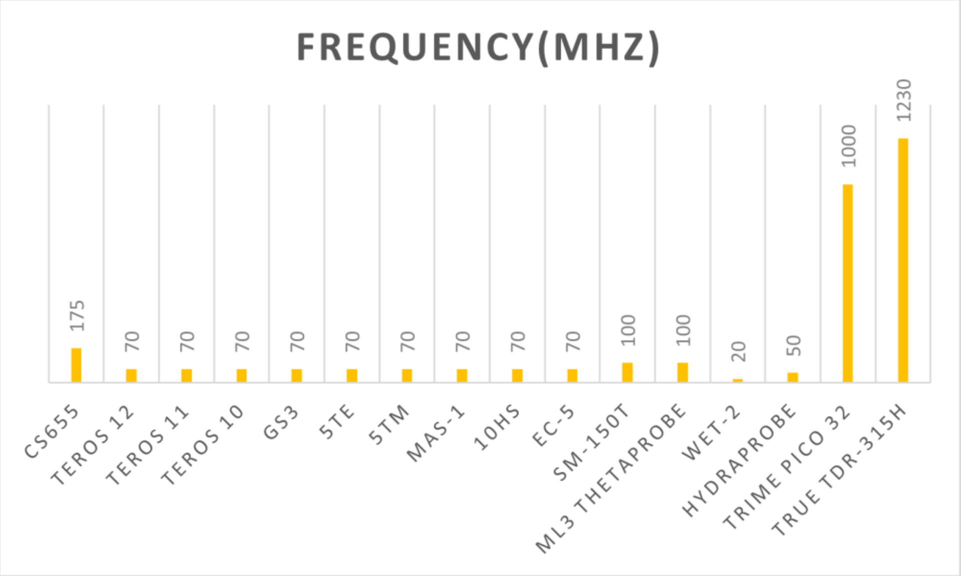


Figure 1. 8 A comparison of operating frequency (MHZ) vs commercially available sensor

Figure 1.8 illustrates that the IMKO’s TRIME probes and ACCLIMA’s TRUE TDR sensor measure moisture and conductivity very precisely at a frequency of 1GHz with a better and more exact separation of moisture and conductivity in comparison to capacitive probes with lower frequencies.

The commercial moisture sensors are classified based on it working type.

1. Resistive
2. Dielectric
    - ➢ TDR(Time Domain Reflectometry)
    - ➢ FDR(Frequency Domain Reflectometry)
3. Thermal
4. Neutron Probe
5. Gamma Ray Attenuation
6. Near Infrared Reflectometry.

- **Gravimetric Method:** Gold standard , most reliable, high accuracy and reliability. Despite its advantages, it is time consuming, destructive and unrepeatable.
- **TDR** and **FDR** type sensors can only be used to measure Volumetric soil moisture after proper calibration. Soil sensitivity poses unpredictable effects on soil moisture measurement.
- **Capacitance technique:** It is commercially available. Its performance is dependent on the size of the sensor.
- **Neutron Probe** requires trained operator due to the use of Radioactive source which is potentially hazardous to the environment.
- **Gamma Ray Attenuation method** can determine the moisture content at soil surfaces(up to 1-2 cm) but high cost and difficulty to use limit its applicability.
- **Near Infrared Reflectometry** is the potential technology which can factor out the problems resulting from other technologies. A custom NIR sensor, discussed in the following section is the key solution interms of accuracy and portability in most of the applications.

1.8 NIR Sensor

Soil moisture has absorption bands of 970, 1200, 1450 and 1940 nm in the NIR spectrum. Two strong adsorption bands have been reported at and 1450 and 1940 nm due to first overtone of OH-bending band and the combination of OH-stretching band and OH-bending band. A higher moisture content is correlated with deeper water absorption band and vice versa. From a study by Ben-Dor et al 2008 and Zhu et al 2010 [19], they predicted soil moisture content for three types of soil using wavebands 1400,1940 and 2250 nm. The results indicate most significant correlation between soil moisture and reflectance was identified at 1400nm for disturbed soil samples(R^2 = 0.996, RMSE = 0.01 cm^3 cm^{-3}) and at 1940nm for both cored (R^2 = 0.969, RMSE = 0.019 cm^3 cm^{-3}) and surface soil samples(R^2 = 0.895, RMSE = 0.059 cm^3 cm^{-3}).

Expected Outcomes

- To achieve strong liear correlation between soil moisture and relative absorption depths for different soils.
- The reflectance model is dependent of soil type.
- Soil moisture is predicted using an algorithm which is designed for estimating soil moisture using the relative absorption depth from reflectance data of 1800 nm and 1940 nm Wavelength.
- The results agree well with measurements obtained from gravimetric method.
- It can be used in laboratory and field applications.

Sensor Design

The Sensor is composed of Infrared Led as light source generating wavelengths of 1940 nm (Strong water absorption band) and 1800 nm(Weak water absorption band). The Photodiodes are used to receive the reflectance. Sensor case(PVC) of dimensions 2cm diameter at the open end, 5.5cm diameter at the lower end and 2.5cm in height with a possibility to scale down the dimensions. A Light detector is installed at the top of the sensor case, six LED's are arranged circularly around the light detector. All of the LED's are angled at 45° to the normal so that the beam of light all LED's are focused on a plane resulting in central irradiated area of approximately 8mm in diameter at the soil surface.

Typical Layout Configuration

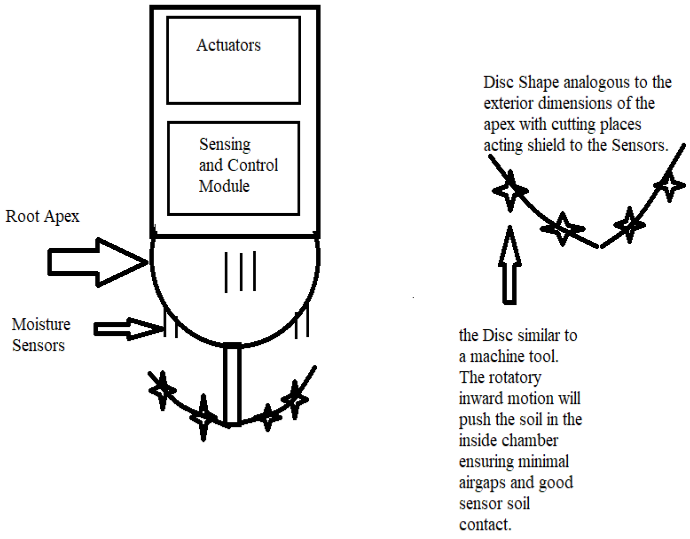

Actuators
Sensing and Control Module
Root Apex
Moisture Sensors
Disc Shape analogous to the exterior dimensions of the apex with cutting places acting shield to the Sensors.
the Disc similar to a machine tool. The rotatory inward motion will push the soil in the inside chamber ensuring minimal airgaps and good sensor soil contact.

Figure 1. 9 Thermal Conductivity Humidity Sensor

The sensor comprises of 2 modules – Actuating Module, Sensing and Control Module. The disc shaped protector comprising sharp blades is affixed at one end of the sensor, this also acts as a shield to the modules of the sensor. The sensor design makes it easy to penetrate inside the growing medium or hostile applications.

Working Prinicple

During operation, the radiations from the LED's are directed to the soil surface at a fixed distance of 15mm. After interacting with the soil surface, a fraction of the reflected light is reflected to the receiving detector and the rest is lost due to scattering absorption and transmission.

The detector has wide angular response to incident radiation and no light reaches the detector directly from the LED's. To generate maximum average power, the infrared LED emits a pulse signal to produce 600mA of current for 700µS. Normally all the LED's at both wavelength of 1940 nm and 1800 nm are under off status. When measurement is started, reference LED's is turned on for a short period of 700µs and the reflectance of the reference light is detected and collected by the control unit. Afterwards the measuring LED's is turned on and the reference LED is turned off after 700µs .

The procedure is repeated once every one second. Upon receiving the light within the spectral response range, the receiving photodiode produces a Voltage signal that is proportional to the amount of incoming light energy. The output Voltage is sent directly to the low-noise amplifier before its input to the data storage unit. The amplifier – a high pass filter can attenuate signals with lower frequencies.

The InGaAsSb photodiode has a spectral response range of 900-2400 nm and a 0.3mmx0.3mm square sensing area with peak sensitivity at 2100 nm. The InGaAsSb photodiode is very sensitive at wavelengths of 1800 nm and 1940 nm.

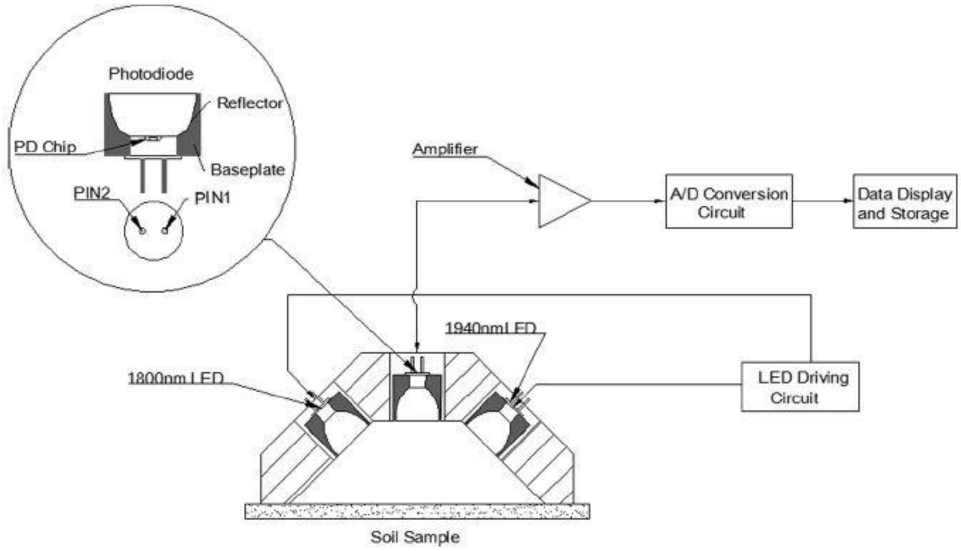


Figure 1. 10 NIR Sensor case schematic

NIR Sensor [19] utilizes the power of opto-electronics and Infra-red spectroscopy to measure soil moisture content, moisture data plays a key role in irrigation scheduling and health of an indoor farming plant.

Robots deployed in irrigation fields can be embedded with NIR sensor, this removes the process of manual intervention problem of traditional sensors. Typically, the holes should be drilled inside the soil based on the dimensions of the sensor and it is manually deployed. The main draw back associated with NIR Sensor is high cost of opto-electronics and is currently limited to research-oriented applications.

In future, the NIR sensor can be deployed in smart farming applications and a custom sensor setup for this applications is identified and depicted below.

Design Setup

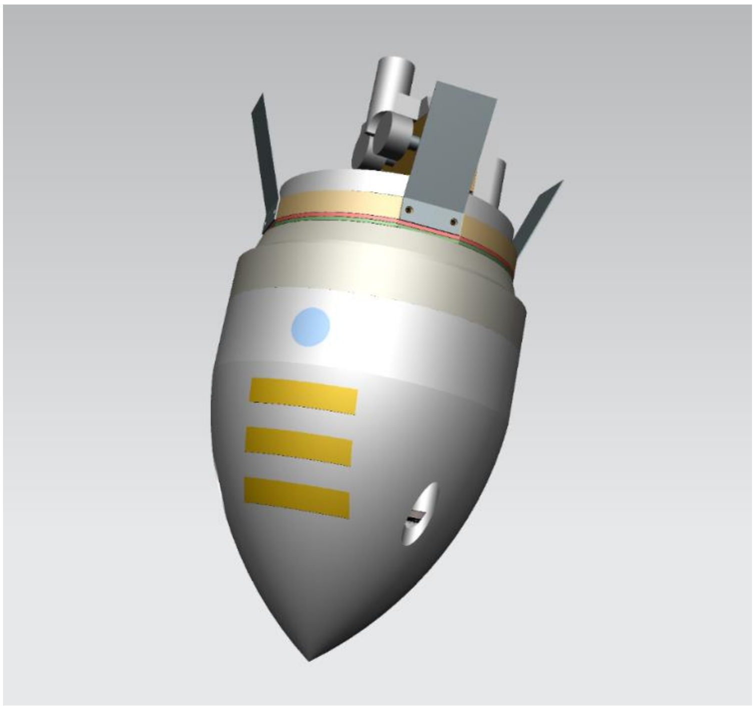


Figure 1. 11 NIR Sensor embedded in a Robotic Root Tip of diameter 2cm.

In some research oriented applications, NIR sensor can be advantageous interms of design dimensional requirements. In figure 1.11, the opposite side of the Blue circle is where the custom NIR Sensor is placed.

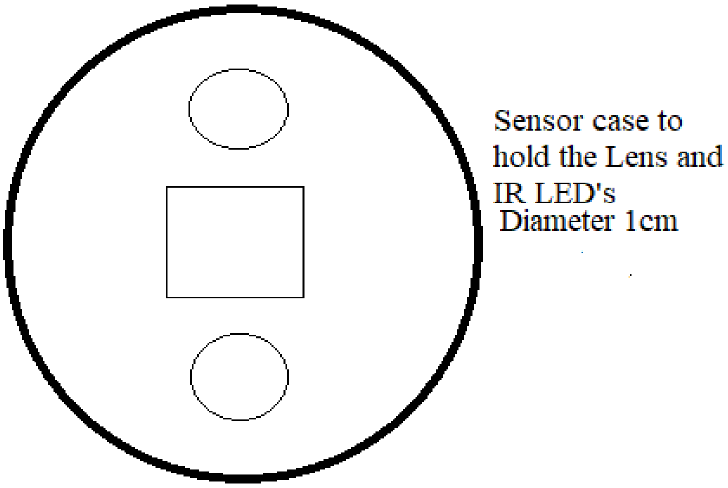


Figure 1. 12 NIR Sensor housing case.

An example of NIR sensor housing is shown in figure 1.12.

Dimensions:

1800nm IR LED: 3mm

Fresnel Lens: 5mm

1900nm IR LED: 3mm

During operation, the radiations from the LED's are directed to the soil surface at a fixed distance of 15mm. After interacting with the soil surface, a fraction of the reflected light is reflected to the photodiode through the Fresnel lens and the rest is lost due to scattering absorption and transmission. The photodiode is placed at the focal point of the Fresnel lens. To generate maximum average power, the infrared LED emits a pulse signal to produce 600mA of current for 700µS. Normally all the LED's at both wavelength of 1940 nm and 1800 nm are under off status. During the operation, the reference LED's is turned on for a short period of 700µs and then the reflectance of the reference light is detected and collected by the control unit. Afterward the measuring LED's is turned on and the reference LED is turned off after 700µs . The procedure is repeated once every one second. Upon receiving the light within the spectral response range, the receiving photodiode produces a voltage signal that is proportional to the amount of incoming light energy. The output voltage is sent directly to the low-noise amplifier before its input to the data storage unit. The amplifier – a high pass filter can attenuate signals with lower frequencies. The amplified output is passed through an I2C Analog to Digital Converter and the output is connected to the ports of the CY8CKIT-059 prototyping kit.

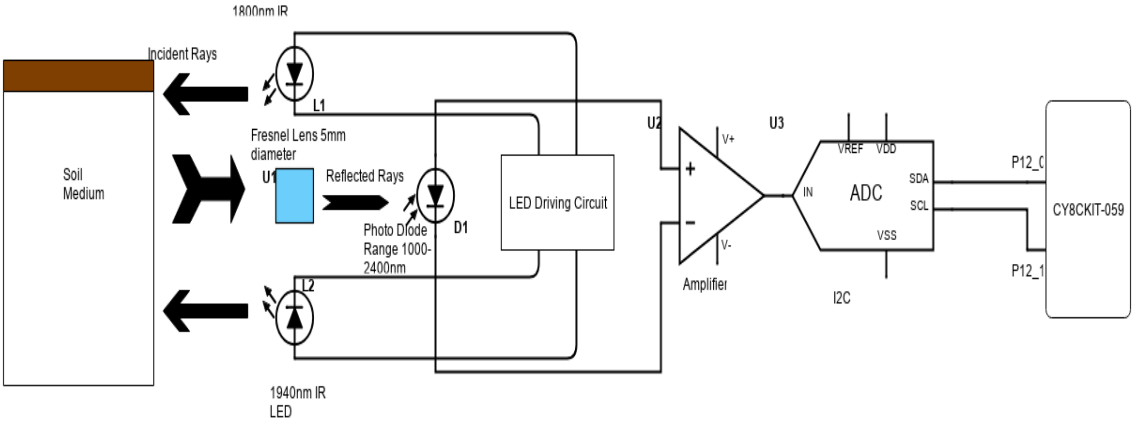


Figure 1. 13 NIR Sensor Schematic Layout

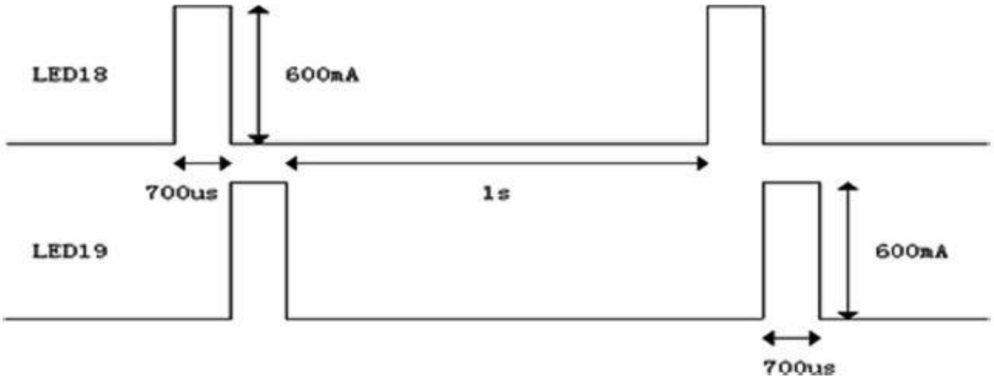


Figure 1. 14 Pulse pattern for LED’s

1.9 Scientific Analysis

Name of Research Paper	***Description***
Internet of Things (IoT): A vision, architectural elements, and future directions [1]	Through this paper, the author presents a vision and motivates the application domain of IoT following a new approach. Cloud technologies and WSN's are used. It also depicts the ways of dealing with challenges and future innovations by compiling with the technologies involving cloud.
IoT based smart agriculture[2]	In this paper, the author shows how the advancement in agricultural field is controlled by the use of various IoT technologies like Smart system GPS & remote controlled robots for performing activities like weeding, spraying, moisture sending, bird-animal scaring, keeping vigilance etc. It also shows how real time accurate field data is obtained through Smart irrigation. In this paper, Smart Warehouse System is also shown.
Design and Development of IoT and Cloud Based Smart Farming System for Optimum Water Utilization for Better Yield[3]	The author generates a statistical analysis of a microcontroller program which helps in maintaining the soil balance after a certain level of dryness that is contained in the soil. The use of sensors helps to balance and monitor the temperature, moisture and humidity of the soil for increasing the yielding capacity. The cloud facility involved is used for storing the data obtained.
Development of Raspberry Pi & IoT Based monitoring & controlling devices for agriculture[4]	In this paper, the authors explain about the sensor networks and its applications in agricultural fields. The data collected is stored in cloud and monitored using IoT device. Improvement in efficiency of agricultural sector is handled using raspberry pi & sensors. The result focuses on soil moisture, humidity & temperature of the field.

Name of Research Paper	***Description***
Sensor Network Data Acquisition & Task Management for Decision Support of Smart Farming[5]	In this paper, the author focuses on applied agriculture using comprehensive models of IoT. Data acquisition & data analysis are handled using sensors and measure of control and task management are taken care. This analyzed data is further used in modeling and system design.
IoT based Smart Water Management with Sensors Agriculture Stick for Live Temperature and Moisture monitoring[6]	The author shows the Implementation of smart water management system involving certain areas of agriculture which involves distribution of water evenly between the crops for proper plantations. This paper also shows the use of a smart stick based on agricultural field is used for depicting activities like temperature, humidity, moisture etc. It is proposed for validating system accuracy with real time parameters.
Plant monitoring using image processing, Raspberry pi & IoT[7]	The author in this paper describes the need of a device that is required to identify diseases affecting the growth criteria of plants. They have used technologies like Digital image processing in order to identify the type of disease caused by the pathogen and IoT device and sensors.
Design and Implementation of a Connected Farm for Smart Farming System[8]	Through this paper, the author shows the research and analysis which provide a suitable environment for growing crops by proposing a design for smooth deployment of IoT device into connected farms. In this paper, there are basically three components – sensors (temperature, humidity, CO2, illumination) & controllers (sprinkler, LED light, heater, AC); IoT gateway (&Cube) and IoT service Platform (Mobius).

Name of Research Paper	***Description***
Smart Farming using IoT[9]	The author depicts a better way of irrigation by penetrating water system which helps in gathering information regarding the crops such as humidity, moisture and dampness present in the soil.
Decision support system for smart farming with hydroponic style[10]	The authors in this paper describe the need and implementation of Smart farming with the help of hydroponic farming technique which results in aiding facilities that could help farmers to grow crops in areas where it is impossible to cultivate crops. This method aims to lower decaying and toxicity of plant allowing fishes to breed with ease.
Technology based streamlined agro-farming techniques[11]	This paper focuses on cultivating a specific(Tomato) plant using technologies like smart farming and IoT. Yield improvement is enhanced by implementing Machine Learning Algorithms and Intelligent data acquisition.
A smart irrigation system to automate irrigation process using IoT and Artificial Neural Network[12]	In this paper, the author focuses on creating a low-cost smart device which can be used to collect the data of moisture in soil at different places, pH and temperature of soil. Arduino is used along with some sensors and modules to collect these data at different places in farm. After collecting this data this device will send this data to server with the help of a Wi-Fi module attached to the device. Artificial Neural Network model is used for the analysis of this data for generating high accuracy results. Finally based on the results the device triggers a valve to automatically start and stop the flow of water in the farm.

Name of Research Paper	***Description***
IoT and machine learning based approach for fully automated greenhouse[13]	The parameters that impact the yield of crops like humidity, CO2 levels, light intensity, soil moisture, temperature are monitored, controlled and coordinated using Raspberry Pi and Arduino. Internet of Things enables real-time data collection from the Smart Greenhouse and visualization on ThingSpeak platform. This paper proposes a fully automated greenhouse embedded with hydroponics and vertical farming and with excellent security provisions and surveillance to become a highly advanced and diverse version of currently prevailing models.
Smart Harvest Analysis using Raspberry pi based on IoT[14]	The SHARP project features a responsive dashboard and graphs to control motors, monitor topographical data which can be viewed from any device such as mobile phones, tablets or computers. SHARP includes water level management, automatic irrigation and predictive analysis of farm lands. Cloud/Raspberry Pi server with database (stores the data) has been deployed for the prediction of crop yield by Machine learning algorithm. It also provides the manual and automatic control of the motor from the dashboard to further help farmers to manage their farms irrigation system. This will also help in water resource management (no wastage of water due to automatic control).
A control system in an intelligent farming by using Arduino technology[15]	An intelligent farming system (IF) is proposed to improve the production process in planting. IF composes of two main parts which are a sensor system and a control system. In this paper, the author focuses on the control part which are watering and roofing systems of an outdoor farm based on the statistical data sensed from the sensor systems (including temperature, humidity, moisture and light intensity sensors) Since the sensed data would not be always accurate due to noises, Kalman filtering is used to smooth the data before using as an input in the decision-making process. A decision tree model is generated to predict the weather condition. Then, a set of decision rules based

Name of Research Paper	***Description***
	on both the sensed data and the predicted weather condition is developed to automatically make a decision on whether watering and roofing system should be on or off. The function for users to manually control the watering and roofing systems is provided via mobile application.
An extended Kalman Filter for low-cost positioning system in agricultural vehicles[16]	This paper proposed an Extended Kalman filter by adopting artificial bee colony (ABC) algorithm for dynamic tuning. It is an efficient and mathematical algorithm that processes imprecise observation of input data and creates an optimal estimate by providing a prediction model and an observation model is proposed. The first phase is prediction stage and the second phase is the update stage in the system to produce an a posteriori state estimate, by adjusting the previous a priori estimate. ABC optimization algorithm is used to generate an optimal precise output. the developed model is implemented in the working platform of MATLAB and output is compared with the existing technique to evaluate the performance.

Table 1. 2 Scientific Analysis

Chapter 2

Overview

2.1 Motivation

The Thesis work derives its roots from **PLANTOID** – an EU funded H2020 project [20] which aims at taking bio-inspiration and biomimicry of plant roots to develop a new generation of robots and ICT technologies in sensing, actuation and distributed adaptive intelligence for tasks of soil exploration and monitoring and the **SMASH project** [21], a research project aimed to create collaborative and integrated robotic technology solutions to solve issues related to food safety and environmental sustainability funded by Tuscany Region (Tuscany POR FESR 2014-2020).The goal of SMASH (Smart Machine for Agricultural High tech Solutions) project is to build a smart ecosystem for monitoring the crop field with the help of sensors to monitor light, humidity, temperature, soil moisture and to automate the irrigation system. It also aims at collecting data for monitoring and managing grapevine and spinach yields.

SMASH project is expertise from Tuscany academics and industry partners, expert in the agri-food field. The SMASH consortium combines 11 partners focused on the design and implementation of robotic devices equipped with sensors, actuators, gateways, integrated with a cloud-based platform including data analytics, remote management, web, and mobile applications. The SMASH platform includes four different modules: crop-field monitoring (**AgroBot**), soil monitoring (**Plantoide**), flying sensors (**FlyBot**) and ancillary equipment in the field (**AncillaryBot**). The major benefits of the real-time data collection and processing include crop health imaging, integrated mapping, and surveying of agricultural land. From the data processing, growers can draw insights regarding plant health indices, plant counting and yield prediction, plant height measurement, quantity of fertilizers, drainage mapping, weed pressure mapping, etc.

The SMASH Consortium partners are: **E.D.I. Progetti e Sviluppo srl**, **Yanmar R&D Europe srl**, **Avmap srl**, **Base srl**, **Seintech srl**, the **Dipartimento di Gestione dei Sistemi Agrari, Alimentari e Forestali dell’Università degli studi di Firenze**, the **Istituto Italiano di Tecnologia - The BioRobotics Institute**, **Copernico srl**, **Giuntini Filippo and Dorian srl**.

2.2 Brief Description of Plantoid Project

The initial work of the Thesis work was carried out remotely in collaboartion with the **Istituto Italiano di Tecnologia - The BioRobotics Institute.** The paper [20] describes the design of a minuaturized mechatronic system inspired by plant roots for soil exploration is studied and the main research challenges of accurate soil moisture measurements is identified. Another main issue addressed was to embed the sensors in a mechatronic root apex of 2cm in diameter. The current existing scientific technologies and commercial solutions make it challenging task to achieve the design goal. The NIR sensor discussed in the previous chapter can solve the design issue and provide robustness even in hostile working conditions.

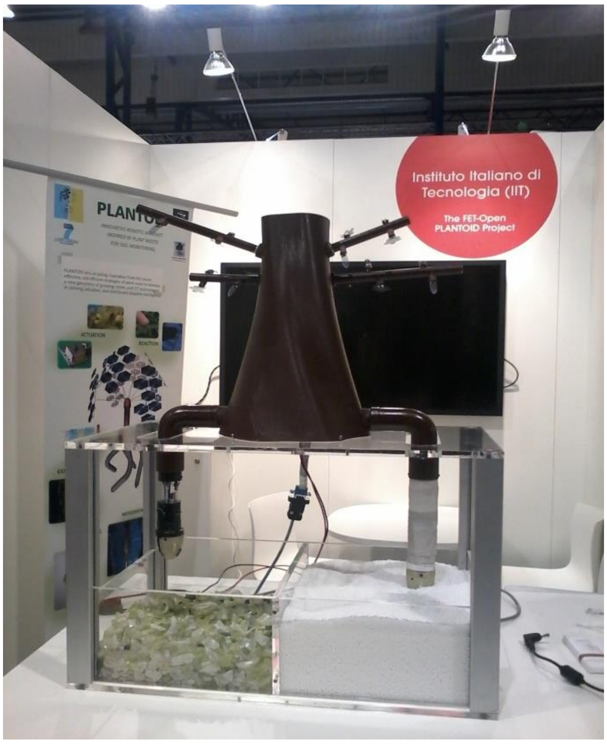

Instituto Italiano di Tecnologia (IIT)
The FET-Open PLANTOID Project
PLANTOID

Figure 2. 1 Prototype of Plantoid Project

The Plantoid is consisted of the following main elements – Robotic roots, Trunk and the Active leaves. The robotic root tips are embedded with MEMS(micro-electromechanical systems) sensor for moisture, temperature and humidity sensing. The roots are penetrated inside the growing medium through a mechanically linked osmotic actuator. overall architecture: a trunk, used to collect data from the roots and to exchange data with an external PC, and a total of five roots. The root apex is the same for all the roots, while the actuation system varies there are two roots for growing and three for bending (with the soft bending actuator)

2.3 Problem Statement

In the literature review, existing papers focus on the following main ideas – Plants classification, landcover identification, identifying growth defects and diseases, pest and weed detection. The latest papers focus on crop prediction using machine learning techniques. The proposed system is a new concept of growing user specific herbs by automating the crop's growing conditions. There are very few solutions in the consumer market which is closely like the proposed design but lacks the automation in comparison to the low-cost effective and energy efficient design of the proposed model.

2.4 Project Description

The proposed project focuses on an entirely new emerging concept of Indoor Farming. It deals with the concept of growing a user specified herb with automated growing conditions using IoT. It consists of the following technologies such as:

- **Sensors** for measuring soil moisture monitoring, Light intensity levels, Temperature and humidity.
- **Single Board Computer** for implementing a smart control system for the growing pod.
- **Growing Medium** such as aeroponics and hydroponics.
- **Actuators** for transporting water-nutrient mixture to the growing medium.
- **Communication technologies** like the advanced wireless communications.
- **Data Analytics** for decision making and prediction.
- **Cloud Computing** for remote database and smart decision making for the automated software.

Chapter 3

System Design and Implementation

The smart indoor farm can be used to grow herbs in indoor conditions without manual intervention. The entire process of cultivation, irrigation and growing is completely automated. The chapter discusses the design setup, hardware configuration and software architecture of the system. A classical approach of systems design engineering is presented based on the design objectives and the problem statement.

3.1 Hardware Components

3.1.1 Raspberry Pi – Model 3B+ (single board computer)

The Raspberry pi Model 3B+ single board computer [22] is the latest product in the Raspberry pi 3 range of single board computer, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE capability via a separate PoE HAT. The dual-band wireless LAN comes with modular compliance certification, allowing the board to be designed into end products with significantly reduced wireless LAN compliance testing, improving both cost and time to market.

Raspberry Pi 3B+ Specifications

- **SoC:** Broadcom BCM2837B0 quad-core A53 (ARMv8) 64-bit @ 1.4GHz
- **GPU:** Broadcom Videocore-IV
- **RAM:** 1GB LPDDR2 SDRAM
- **Networking:** Gigabit Ethernet (via USB channel), 2.4GHz and 5GHz 802.11b/g/n/ac Wi-Fi
- **Bluetooth:** Bluetooth 4.2, Bluetooth Low Energy (BLE)
- **Storage:** Micro-SD
- **GPIO:** 40-pin GPIO header, populated
- **Ports:** HDMI, 3.5mm analogue audio-video jack, 4x USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)
- **Dimensions:** 82mm x 56mm x 19.5mm, 50g

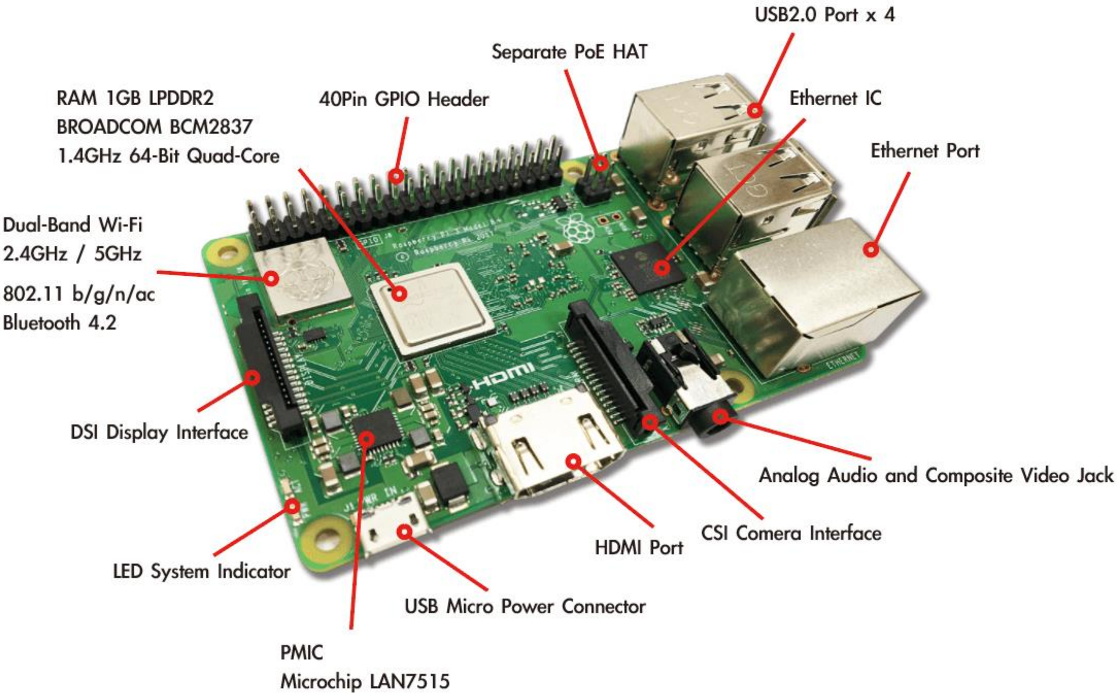


Figure 3. 1 Raspberry Pi Model 3B+ - Parts

Raspberry Pi 3B+ Features

Faster 1.4GHz CPU: It has the same design as the predecessor. The new Raspberry pi 3B+'s BCM2837 system-on-chip(SoC) features a heat-speader which have helped boost the performance from 1.2GHz to 1.4GHz.

Figure 3. 2 Broadcom BCM2837 SoC

- **Faster Ethernet(Gigabit via USB):** Due to the use of single USB channel, the ethernet controller maximises the gigabit connectivity of upto 300Mb/s.

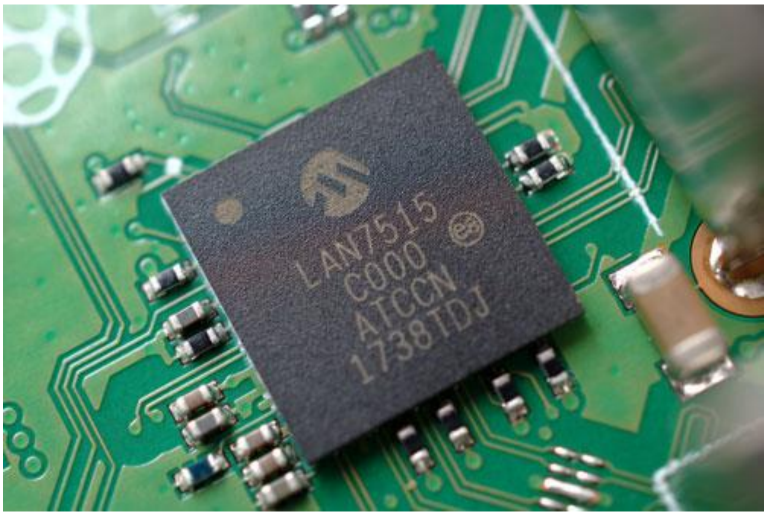


Figure 3. 3 Gigabit Ethernet Controller

- **Dual-band 2.4GHz and 5GHz wireless LAN:** The new dual-band radio module and antenna design improves the Pi 3B+'s connectivity compared to its previous models.

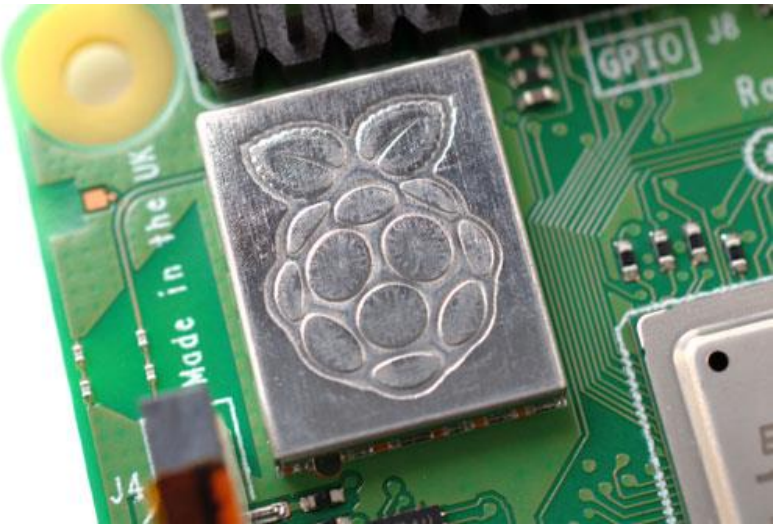


Figure 3. 4 dual-band radio module

- **Power over Ethernet(PoE) ready:** An optional Power-over-Ethernet(POE) HAT module can power the Pi 3B+.

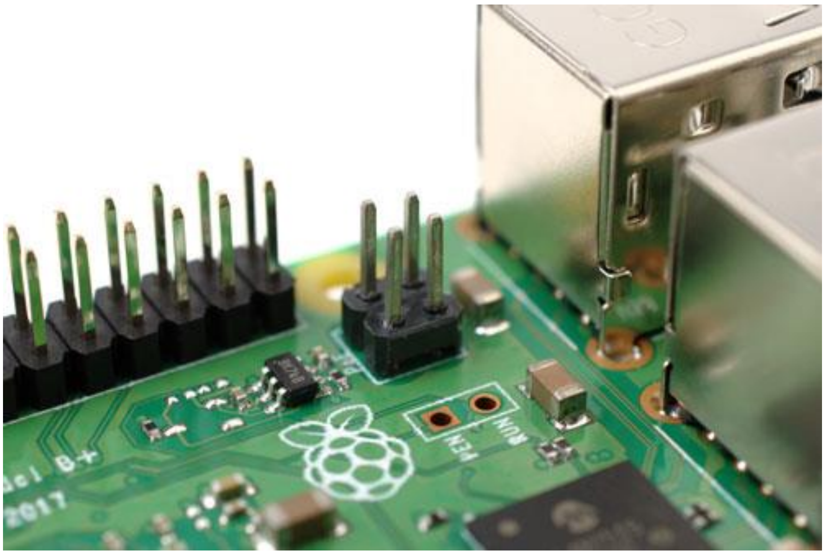


Figure 3. 5 Power-over-Ethernet HAT Module pins

- **Power Management:** A smoother power delivery is provided by the Power Management Inegrated Circuit(PMIC). It replaces the discrete components as compared to its previous models.

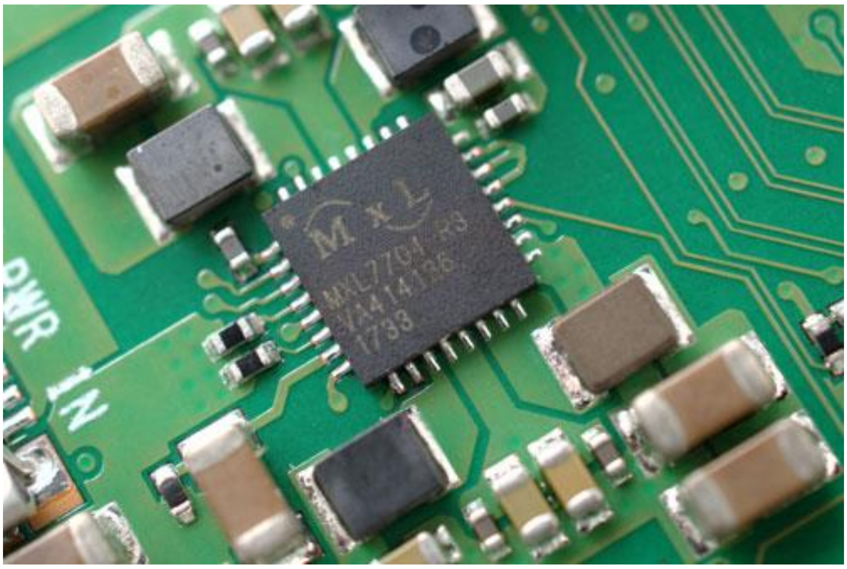


Figure 3. 6 Power Management Integrated Circuit (PMIC)

Raspberry Pi Model 3B+ Pinout with GPIO Functions

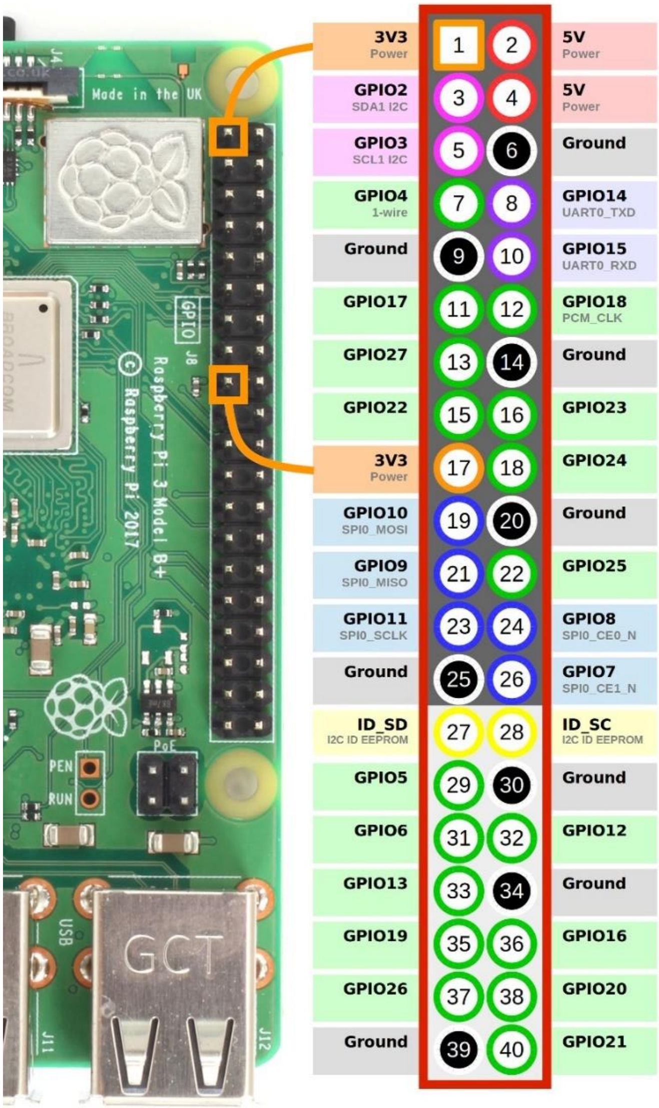


Figure 3. 7 Raspberry Pi Model 3B+ Pinout

Raspberry Pi Model 3B+ consists of 40-pin GPIO like previous pi models. GPIO stands for General Purpose Input Output pins, these pins are used to connect the Raspberry pi board to external input/output devices. A standard interface for connecting a single-board computer or a microprocessor to other devices is through these GPIO pins, the pins don't have a specific function but they can be customized using the software.

Raspberry Pi 3B+ Power Pins

The model B+ board consists of two 5V pins, two 3V3 pins, and 9 Ground pins (0V), which are unconfigurable.

5V: The 5v pins are used to directly deliver the 5v supply coming from the mains adaptor. This pin is used to power up the Raspberry Pi, and it can also be used to power up other 5v devices.

3.3V: The 3v pin is used to provide a stable 3.3v supply to external components and to test LEDs.

GND: Ground pin is commonly referred to as GND. All the voltages are measured with respect to the GND voltage.

Input/Output pins

A GPIO pin set as **Input** reads the signal received by the Raspberry Pi, sent by the device connected to this pin. Any voltage between 1.8V and 3.3V is read as HIGH and voltage lower than 1.8V as LOW by the Raspberry Pi. In a case where an input voltage above 3.3V when connected to any of the GPIO pins can severely damage the circuitry of the board.

A GPIO pin set as an **output** pin sends the voltage signal as high (3.3V) or low (0V). When this pin is set to HIGH, the voltage at the output is 3.3V and when set to LOW, the output voltage is 0V.

The GPIO pins can also perform a variety of specific functions listed below.

PWM (pulse-width modulation) Pins on Model 3B+

- Hardware PWM is available on these pins: GPIO12, GPIO13, GPIO18 and GPIO19.
- Software PWM is available on all pins.

SPI (Serial Peripheral Interface) Pins on Model 3B+

SPI (Serial Peripheral Interface) is another protocol used for master-slave communication. It is used by the Raspberry pi board to quickly communicate between one or more peripheral devices. Data is synchronized using a clock (**SCLK** at GPIO11) from the master (RPi) and the data is sent from the Pi to the SPI device using the **MOSI** (Master Out Slave In) pin. If the SPI device needs to communicate back to Raspberry Pi, then it will send data back using the **MISO** (Master In Slave Out) pin.

The Raspberry pi needs the following 5 pins to establish an SPI communication.

- **GND**: All GND pins from all the slave components and the Raspberry Pi 3 board should be connected.
- **SCLK**: Clock of the SPI. All SCLK pins should be connected.
- **MOSI**: It stands for Master Out Slave In. This pin is used to send data from the master to a slave.
- **MISO**: It stands for Master In Slave Out. This pin is used to receive data from a slave to the master.
- **CE**: It stands for Chip Enable, one CE pin per slave (or peripheral devices) should be connected in the circuit. By default, there are two CE pins but more CE pins from the other available GPIO pins can be configured.

SPI pins on Raspberry Pi Model 3B+

- **SPI0:** GPIO9 (MISO), GPIO10 (MOSI), GPIO11 (SCLK), GPIO8 (CE0), GPIO7 (CE1)
- **SPI1:** GPIO19 (MISO), GPIO20 (MOSI), GPIO21 (SCLK), GPIO18 (CE0), GPIO17 (CE1), GPIO16 (CE2)

I2C Pins on Raspberry Pi Model 3B+

I2C is used by the Raspberry Pi board to communicate with devices that are compatible with Inter-Integrated Circuit (a low-speed two-wire serial communication protocol). This communication standard requires master-slave roles between both the devices. I2C has two connections: **SDA (Serial Data)** and **SCL (Serial Clock)**. They work by sending data to and using the SDA connection, and the speed of data transfer is controlled via the SCL pin.

- **Data**: (GPIO2), Clock (GPIO3)
- **EEPROM Data**: (GPIO0), EEPROM Clock (GPIO1)

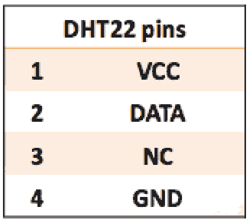
UART Pins on Raspberry Pi Model 3B+

Serial communication or the **UART** (Universal Asynchronous Receiver / Transmitter) pins provide a way to communicate between two microcontrollers or the computers. TX pin is used to transmit the serial data and RX pin is used to receive serial data coming from a different serial device.

- TX (GPIO14)
- RX (GPIO15)

3.1.2 DHT22 AM2302 Digital Temperature and Humidity Sensor Module (Digital Output)

The AM2302 [23] is a wired version of the DHT22 embedded in a large plastic body. It is commonly used low-cost digital temperature and humidity sensor which uses a capacitive humidity sensor and a thermistor to measure the surrounding air. The output is a digital signal on the data pin (no analog input pins needed).



DHT22 pins	
1	VCC
2	DATA
3	NC
4	GND

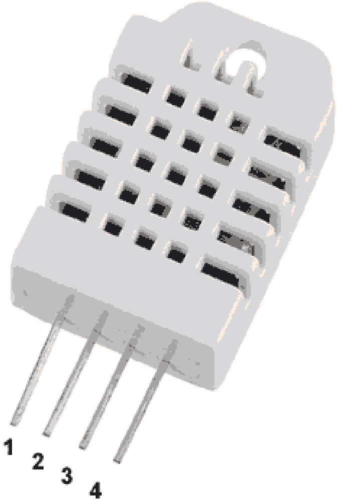


Figure 3. 8 DHT22 Pinout

DHT22 Specifications

- power supply: 3.3V – 6V DC
- output signal: single bus
- sensing element: polymer humidity capacitor & DS18B20
- measuring range: humidity 0-100% RH / temperature -40°C – 125°C
- accuracy: humidity ±2% / temperature ±0.2°C
- sensing period: ~2s

DHT22 Features

- **Digital signal collecting technique and humidity sensing technology:** It outputs a clean digital signal without the need for Analog to Digital converters and can be connected directly to one the GPIO pins of the Raspberry pi.
- Small and compact size
- Low power consumption
- Long transmission distance(upto 20m)
- One 100nF capacitor can be added between VDD and GND for wave filtering.

3.1.3 Capacitive Soil Moisture Sensor v1.2 (Analog Output)

This capacitive soil moisture sensor v1.2 [24] measures soil moisture levels by capacitive sensing technique rather than resistive sensing like other types of moisture sensor. It is made of corrosion-resistant material and has a long service life.

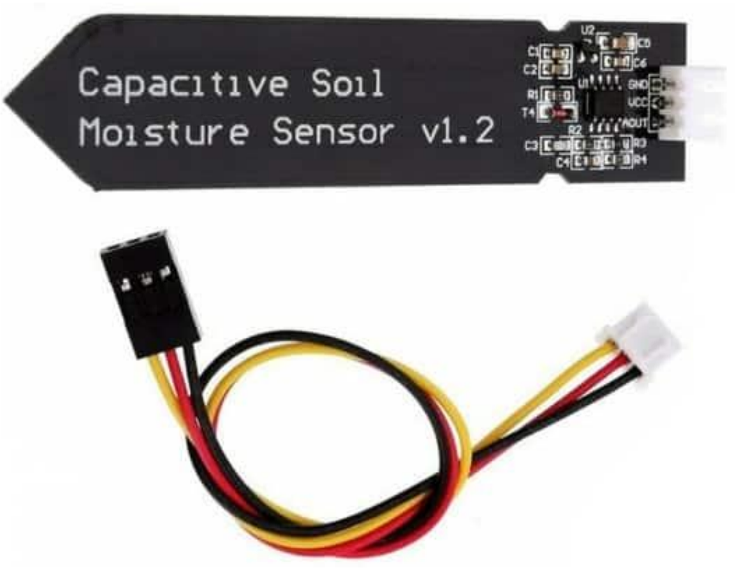


Figure 3. 9 Capacitive Soil Moisture Sensor v1.2

Capacitive Soil Moisture Sensor v1.2 Specifications

- Operating Voltage: DC 3.3-5.5V
- Output Voltage: DC 0-3.0V
- Interface: PH2.0-3P
- Size: 99x16mm/3.9×0.63″

Capacitive Soil Moisture Sensor v1.2 Features

- It Supports 3-Pin Gravity Sensor interface.
- Analog output: It requires an Analog to Digital converter(A/D) while connecting to the Raspberry pi as pi lacks inbuilt A/D converter.

3.1.4 Resistive Soil Moisture Sensor (Analog Output)

The resistive soil moisture sensor [25] consists of two probes which are used to measure the volumetric content of water. The two probes allow the current to pass through the soil and then it gets the resistance value to measure the moisture value. When there is more water, the soil will conduct more electricity which means that there will be less resistance. Therefore, the moisture level will be higher. Dry soil conducts electricity poorly, so when there will be less water, then the soil will conduct less electricity which means that there will be more resistance. Therefore, the moisture level will be lower.

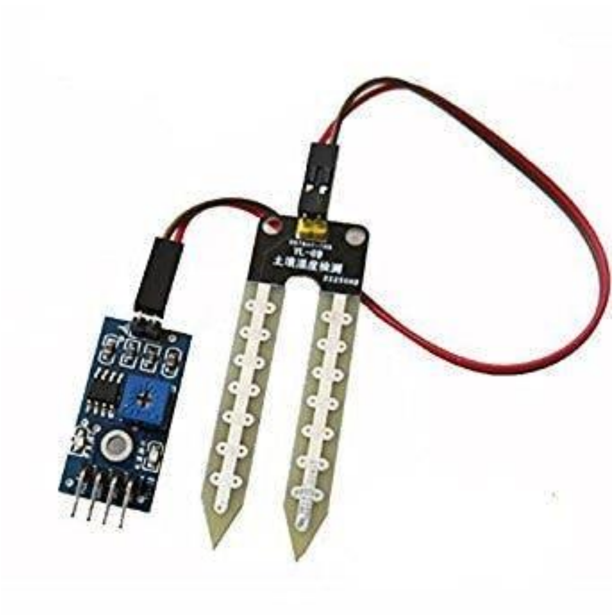


Figure 3. 10 Resistive Soil Moisture Sensor

Resistive Soil Moisture Sensor Specifications

- Operating voltage: 3.3V~5V
- Panel PCB Dimension: Approx.3cm x 1.5cm
- Soil Probe Dimension: Approx. 6cm x 3cm
- Cable Length: Approx.21cm
- VCC: 3.3V-5V
- GND: GND
- DO: digital output interface (0 and 1)
- AO: analog output interface

Resistive Soil Moisture Sensor Features

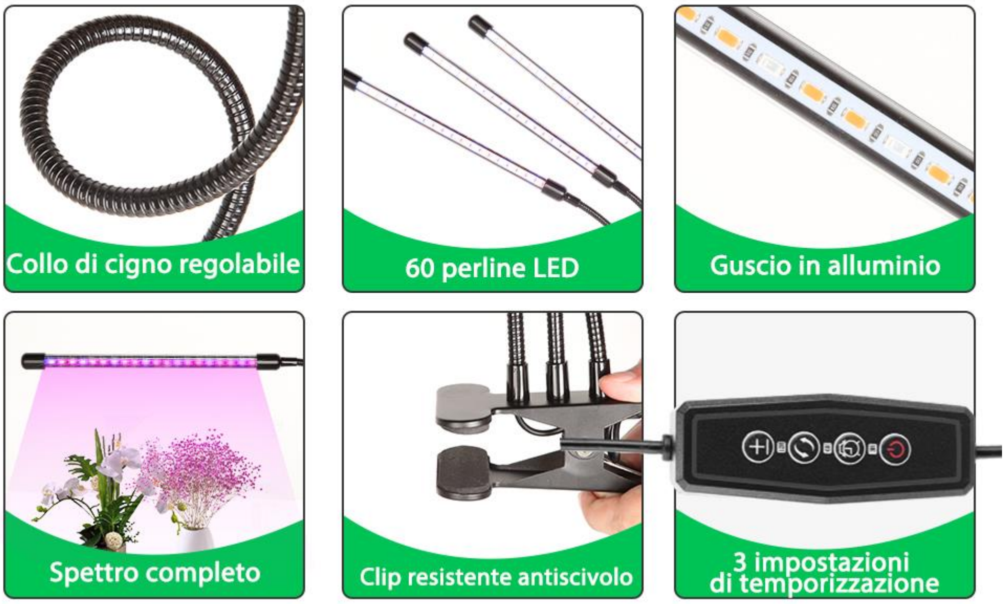
- Dual output mode, analog output more accurate
- A fixed bolt hole for easy installation
- With power indicator (red) and digital switching output indicator (green)
- Having LM393 comparator chip, stable

3.1.5 LED indoor grow lights

The LED indoor grow lights by Infinity [26] are commercially available but some minor design changes are applied because it is the primary actuator used with the switching relays for scheduled grow period. The plant grow light emits targeted wavelength lights that can be fully absorbed by plants to help accelerate growth. Ideal for all types of veg and flowering indoor plants in all stages of growth.

Figure 3. 11 Infinity full spectrum LED Plant grow lights

LED Grow Lights Features

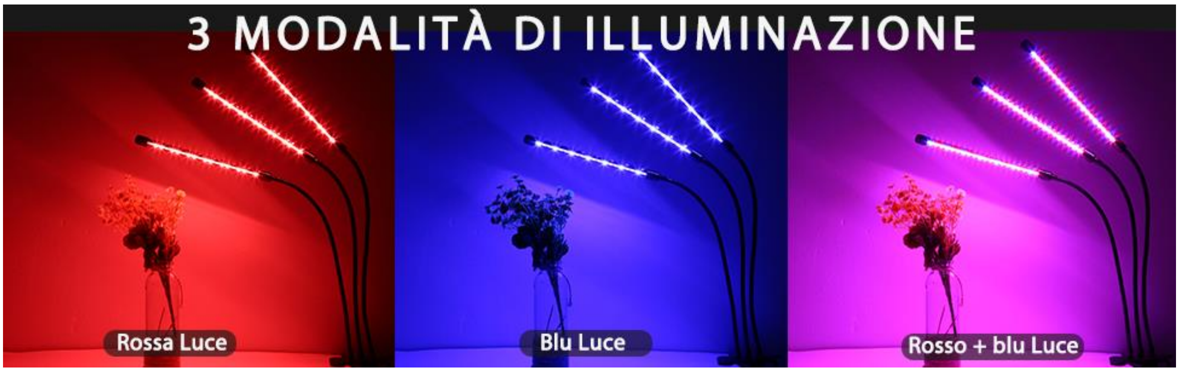


Collo di cigno regolabile
60 perline LED
Guscio in alluminio
Spettro completo
Clip resistente antiscivolo
3 impostazioni di temporizzazione

Figure 3. 12 LED Plant grow lights – advanced features.

- **Three Heads Improved Plant Light:** Its design feature includes three goosenecks which solves the installation and inflexibility problem of traditional led plant light, but also provides a much larger coverage area than grow light clip with two arms. The coverage area with one head is approximately 24-27 inches depending on the height of the head location.

- **Led Grow Light Full Spectrum:** According to the type and light output (PAR) needed to support photosynthesis, this LED grow light with 3 color modes (red light, blue light and mixed light) and 5 brightness settings from 20% to 100%, which effectively complements the lack of natural sunlight and favored plant growth at all stages



. Figure 3. 13 LED Plant grow lights – illumination modes

- **360 degree adjustable:** It is powered by USB or AC power socket (USB power supply included), it is portable and convenient to use at home or office. The gooseneck can be rotated 360 degrees which makes it easy to control the angle and distance between the grow light and the plants.

- **Wide Usage:** The led plant grow light is suitable for indoor plant seedlings, potted plants, leaf cuttings, greenhouse vegetables, etc, especially when the plant needs extra light when it rains, snows, it is dark in the covered, etc. Plant light help promote the growth, flowering and growth stages of plants.

3.1.6 MCP3008 Analog to Digital Converter IC

The MCP3008 10-bit Analog-to-Digital Converter (ADC) [27] combines high performance and low power consumption in a small package, making it ideal for embedded control applications. The MCP3008 features a successive approximation register (SAR) architecture and an industry-standard SPI serial interface, allowing 10-bit ADC capability to be added to any PIC® microcontroller. The MCP3008 features 200k samples/second, 8 input channels, low power consumption (5nA typical standby, 425µA typical active), and is available in 16-pin PDIP and SOIC packages. Applications for the MCP3008 include data acquisition, instrumentation and measurement, multi-channel data loggers, industrial PCs, motor control, robotics, industrial automation, smart sensors, portable instrumentation and home medical appliances.

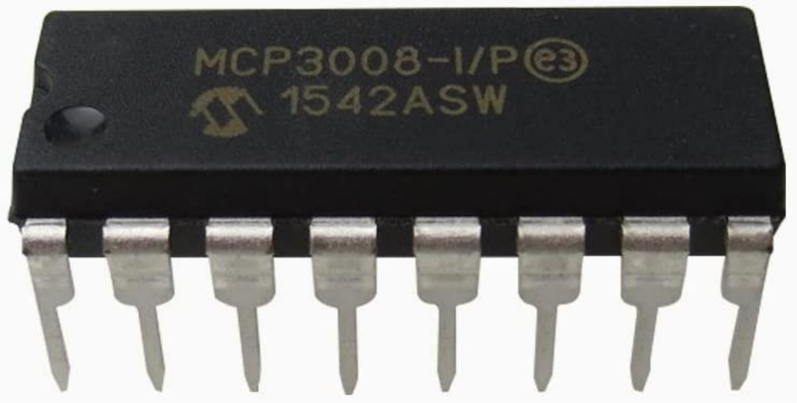


Figure 3. 14 MCP3008 Analog to Digital Converter IC

MCP3008 A/D Converter IC Pinout

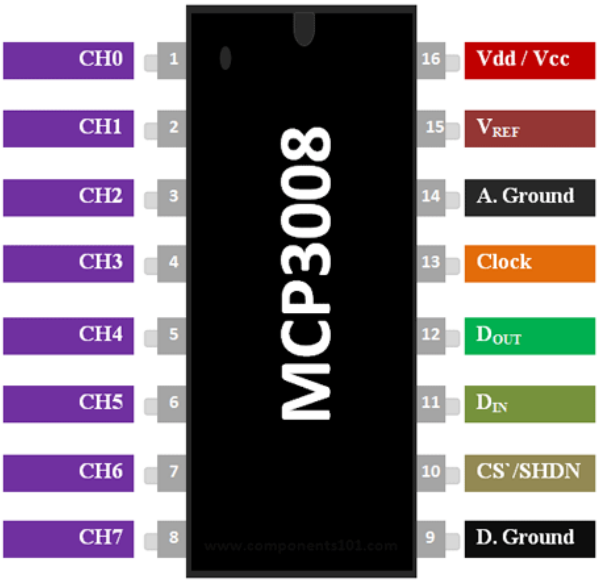


Figure 3. 15 MCP3008 Analog to Digital Converter IC – Pinout

Pin Number	Pin Name	Description
1,2,3,4,5,6,7,8	Analog Input Channels	These are the 8 Input pins, to which the analog voltage which must be measured is provided.
9	Digital Ground	Connected to the Ground of the circuit
10	Chip Select / Shutdown (CS`/SHDN)	This pin is connected to GPIO pin or MCU for turning on or off the IC
11	Serial Data In (DIN)	Used for SPI communication
12	Serial Data Out (DOUT)	Used for SPI communication
13	Serial Clock (CLK)	Used to provide clock signal for SPI communication
14	Analog Ground	Connected to Ground of the reference voltage
15	Reference Voltage (VREF)	Connected to reference voltage for ADC Conversion

Table 3. 1 MCP3008 Analog to Digital Converter IC – Pinout

MCP3008 A/D Converter IC Specifications

- Max Sample Rate(ksamples/sec): 200
- Max Supply Current(µA): 500
- Input Type: Single-ended
- Number of Input channels: 8
- Resolution(bits): 10
- Interface: SPI
- Temperature Range(°C): -40 to +85°C
- Input Voltage Range(V): 0 to 5.5V

MCP3008 A/D Converter IC Features

- Analog inputs programmable as single-ended or pseudo-differential pairs
- On-chip sample and hold
- SPI serial interface (modes 0,0 and 1,1)
- Low power CMOS technology
- 5 nA typical standby current, 2 µA max.
- 500 µA max. active current at 5V
- Available in PDIP, SOIC and TSSOP packages

3.1.7 Micro Submersible Mini Water Pump DC 3-6 V (Actuator)

The micro submersible pump [28] acts as an actuator delivering water/nutrient solution from the base vessel to the plant roots. It is connected to the relay modules and switches automatically based on the thresold moisture values and is used for irrigation scheduling. The water level of the refill pump is higher than that of the pump when submerged in water. If the water level is too low, the pump temperature is too high, and the pump noise is too loud. To keep the water clean, the pump must be cleaned regularly, and the pump must be kept clean, to avoid blocking the pump rotor. Brushless pumps are particularly sensitive to voltage and burn easily so the primary voltage supply should match the pump voltage specifications.

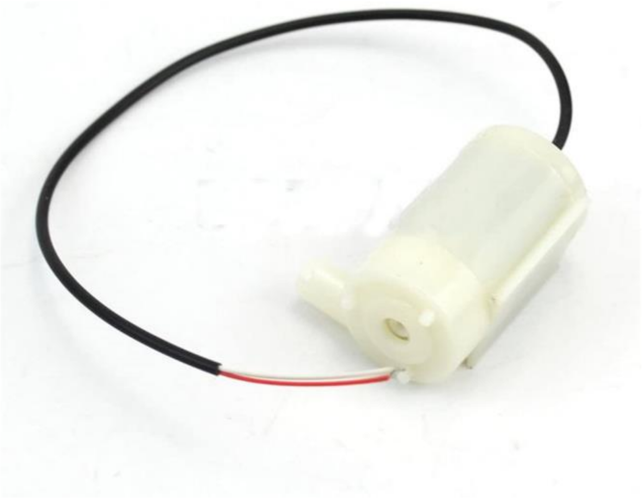


Figure 3. 16 Micro Submersible Mini Water Pump

Micro Submersible Mini Water Pump DC 3-6 V Specifications

- DC Voltage:2.5-6V
- Maximum lift: 40-110cm / 15.75"-43.4"
- Outside diameter of water outlet: 7.5mm / 0.3"
- Inside diameter of water outlet: 4.7mm / 0.18"
- Size:
  Diameter: Approx. 24mm / 0.95"
  Length: Approx. 45mm / 1.8"
  Height: Approx. 33mm / 1.30"

Micro Submersible Mini Water Pump DC 3-6 V Features

- Mini submersible pump is very compact, long lead flow, quiet and used in aquariums, solar water, fountains, computer water cooling.

3.1.8 Li-Polymer Battery Bank 10000 mAh

The external battery bank [29] provides uninterrupted power supply to the Raspberry pi 3B+. Lack of power supply and power flluctuations can severly degrade the performance of on-board electronics and interruption in sensor data acquision. Hence, the presence of an external battery bank ensure fail safe mode as the Rapberry pi lacks digital data clock readings when it is powered off unexpectedly.

Figure 3. 17 Syska 10000 mAh battery bank

Syska 10000 mAh Battery Bank Specifications

- Connectivity: USB 2.0 (2Ports)
- Battery: Li-ion
- Power: Input(5V, 2A), Output(5V,2.1A)
- Capacity: 10000 mAh

Syska 10000 mAh Battery Bank Features

- LED Indicators
- Fast Charging

3.1.9 Cooling Fan DC

The cooling fan [30] plays an important role in maintaining the temperatures of the embedded electronics.

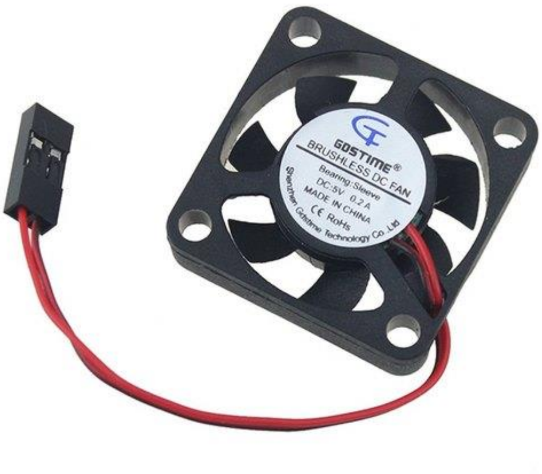


Figure 3. 18 Cooling Fan DC

Cooling Fan Specifications

Input: 5V, 0.2A

Cooling Fan Features

- Perfectly fit on the extruder
- Noiseless Performance
- Very low Current consumption
- Easy to install and connect.

3.1.10 Relay Board Module - 4 Channel 5V

The 4 channel 5V Relay Board [31] can ccontrol various appliances, and other equipment with large current. It is equipped with high-current relay, AC250V 10A; DC30V 10A. It has 4 channels and each one needs 50-60mA Driver Current. Its typical applications include MCU control, PLC control and Smart home control etc. It also has inbuilt indication LEDs for Relay output status.

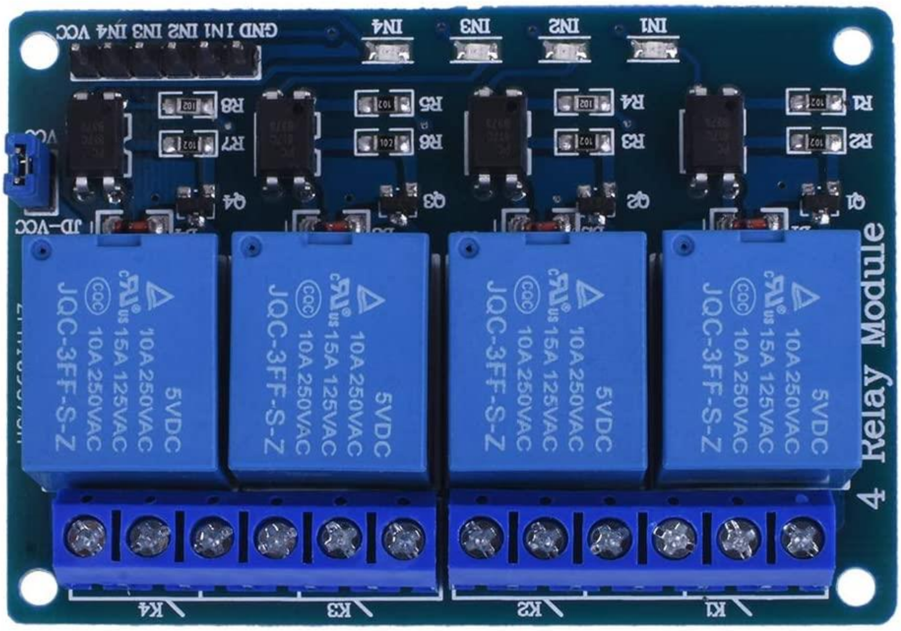


Figure 3. 19 4 Channel 5V Relay Board

4 Channel 5V Relay Board Module Specifications

- Power supply: 5V
- Module size: L75 * W55 * H18.5mm
- Net weight: 61 g
- 4 screw fixing hole: hole diameter: 3.1mm
- Spacing: 68 * 48mm

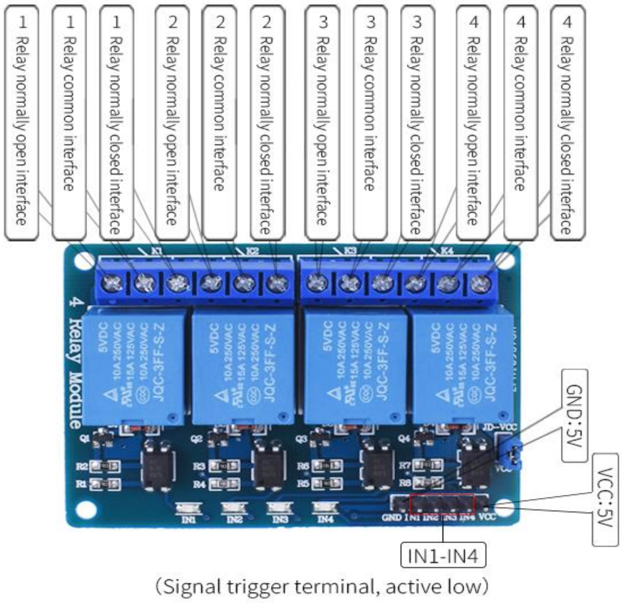

1 Relay normally open interface
1 Relay common interface
1 Relay normally closed interface
2 Relay normally open interface
2 Relay common interface
2 Relay normally closed interface
3 Relay normally open interface
3 Relay common interface
3 Relay normally closed interface
4 Relay normally open interface
4 Relay common interface
4 Relay normally closed interface
K1
K2
K3
K4
4 Relay Module
5VDC
JQC-3FF-S-Z
JD-VCC
GND:5V
VCC:5V
GND IN1 IN2 IN3 IN4 VCC
IN1
IN2
IN3
IN4
IN1-IN4
(Signal trigger terminal, active low)

Figure 3. 20 4 Channel 5V Relay Board – Working Modes

4 Channel 5V Relay Board Module Features

- The module complies with international safety standards. Independently designed on loading and control area.
- The PCB board designed with FR4 dual substrate, more secure and stable.
- The module can directly control various devices and charge.
- It can be fully open or fully closed.
- Point to point control and cycle control.
- 4 relay outputs, each of terminal 5.08-3P (the output is normally open and normally closed)

3.2 Design Setup

The Block diagram in figure 3.21 depicts the architecture of the proposed system. The single board computer Raspberry pi model 3B+ is integrated with the **Sensor modules** – DHT22(Temperature and Humidity sensor), Capacitive Soil moisture sensor v1.2, Resistive Soil Moisture Sensor, **Actuators** - LED grow lights, DC Water Pump. The moisture sensors are placed in the growing medium. The contactless Temperature and Humidity Sensors are placed above the growing medium. The LED grow lights are hooked to a metallic frame or a holder with its goosenecks facing the plant frontal area. The orientation of the LED grow lights is crucial to ensure the plant receives maximum light spectrum intensity levels. The real-time data is sent from the Raspberry pi to the cloud web server for data processing, prediction and decision making using the smart automation software logics. The user can access the system status using the web application. The user can monitor humidity, temperature and moisture values using the interactive time series dashboard.

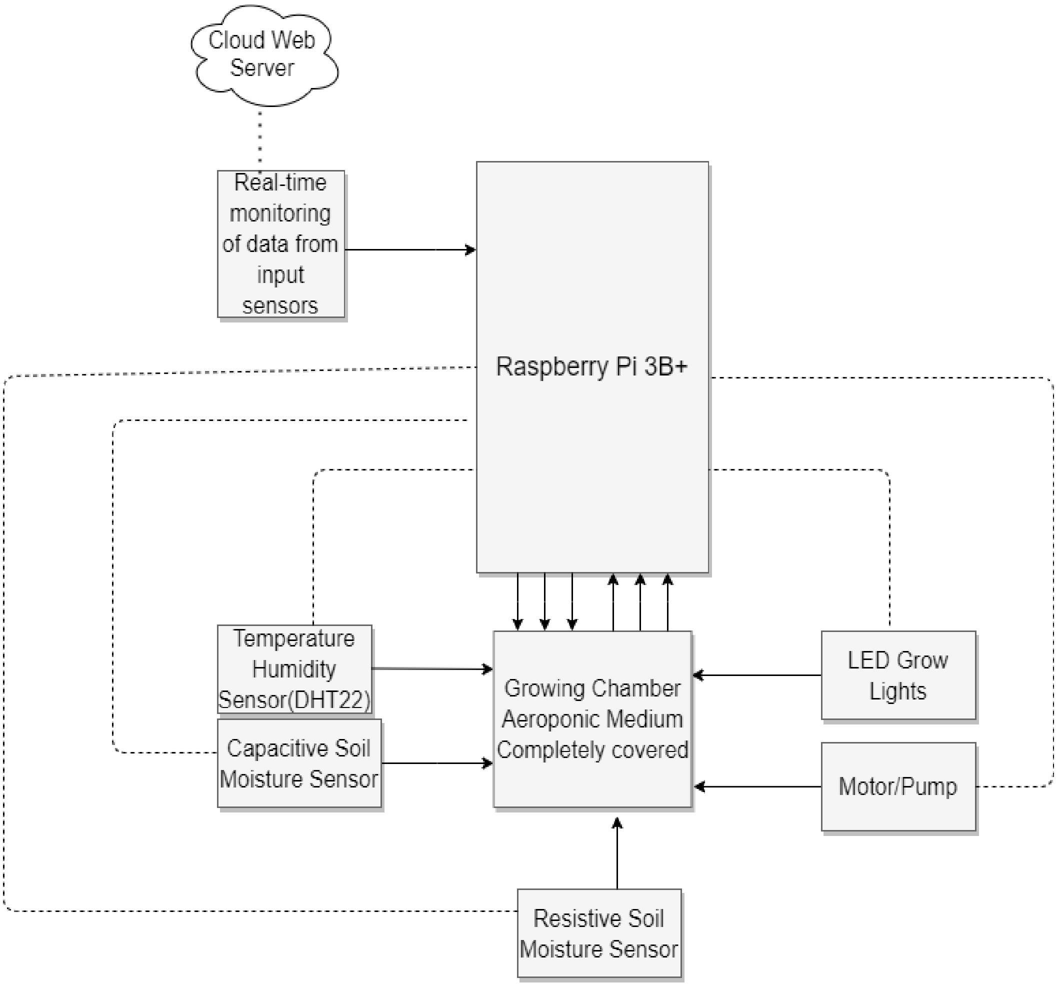

Cloud Web Server
Real-time monitoring of data from input sensors
Raspberry Pi 3B+
Temperature Humidity Sensor(DHT22)
Capacitive Soil Moisture Sensor
Growing Chamber Aeroponic Medium Completely covered
LED Grow Lights
Motor/Pump
Resistive Soil Moisture Sensor

Figure 3. 21 System Level Block Diagram

3.3 Hardware Architecture

The section describes the hardware architecture of the project and discusses extensively the circuit connections of the entire harware components.

3.3.1 EasyEDA

EasyEDA [33] is web based electronic design automation tool used to design, simulate and discuss schematics, simulations and printed circuit boards. This tools enables hardware design engineers in creation of a bill of materials, gerber files and pick and place documentary outputs in PDF, PNG and SVG formats. It also allows the designers to create and edit printed circuit board layouts and optionally, the manufacture of printed circuit boards. It features cross-platform imports from ALTIUM Designer, CircuitMaker, Eagle, Kicad and LTspice file formats as well as generic SPICE netlists.

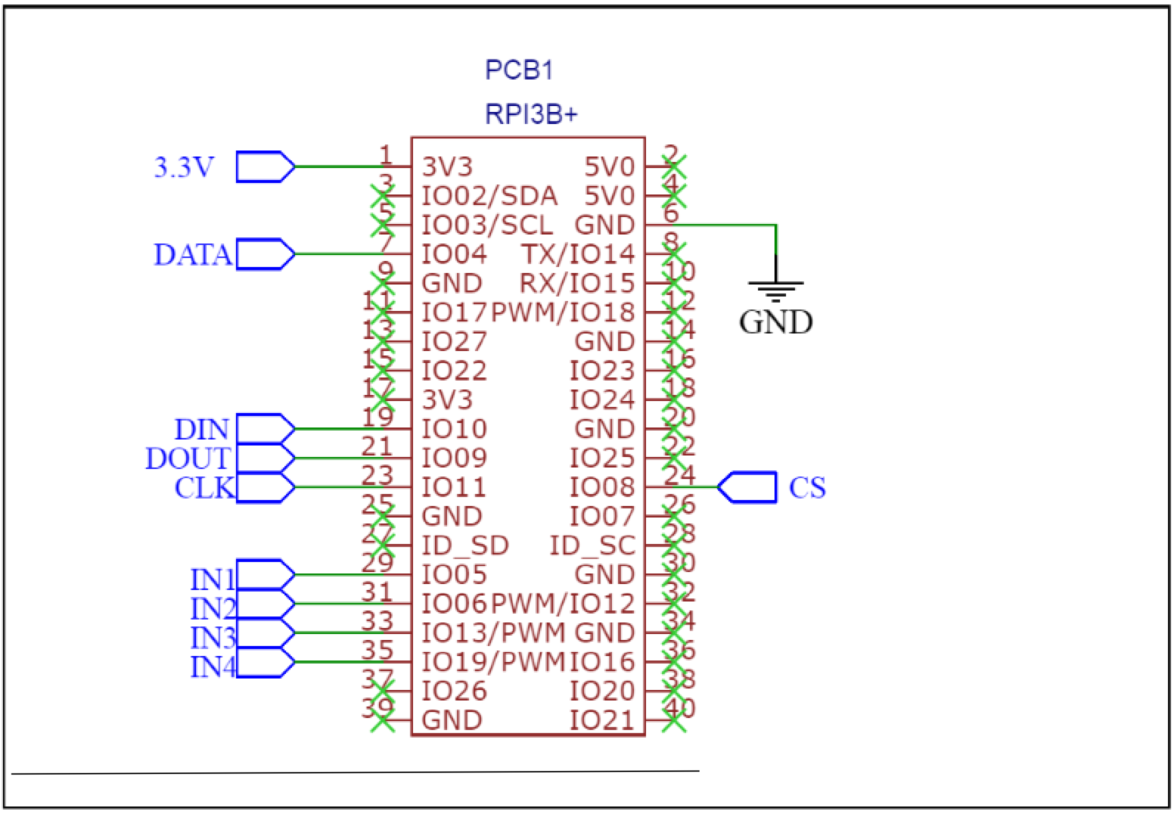


Figure 3. 22 Circuit Connections – Raspberry Pi 3B+

Figure 3.22 shows the connections between the different hardware components and the Raspberry Pi 3B+ board. As discussed earlier, the Pi is powered through a 5v DC supply connected via the AC mains adapter. The Pi can also be powered through the external battery bank delivering stable 5V 2.1A power supply to ensure fail safe mode. The circuit connections are administered through a Breadboard(Protyping Board). The 3V3 pin provides a stable power supply to the external components, it is connected to one of the common rail of the prototyping board. The GND pin is connected to the common ground rail of the prototyping board. The digital output from the DHT22(Temperature/Humidity Sensor) data pin is connected to GPIO4.

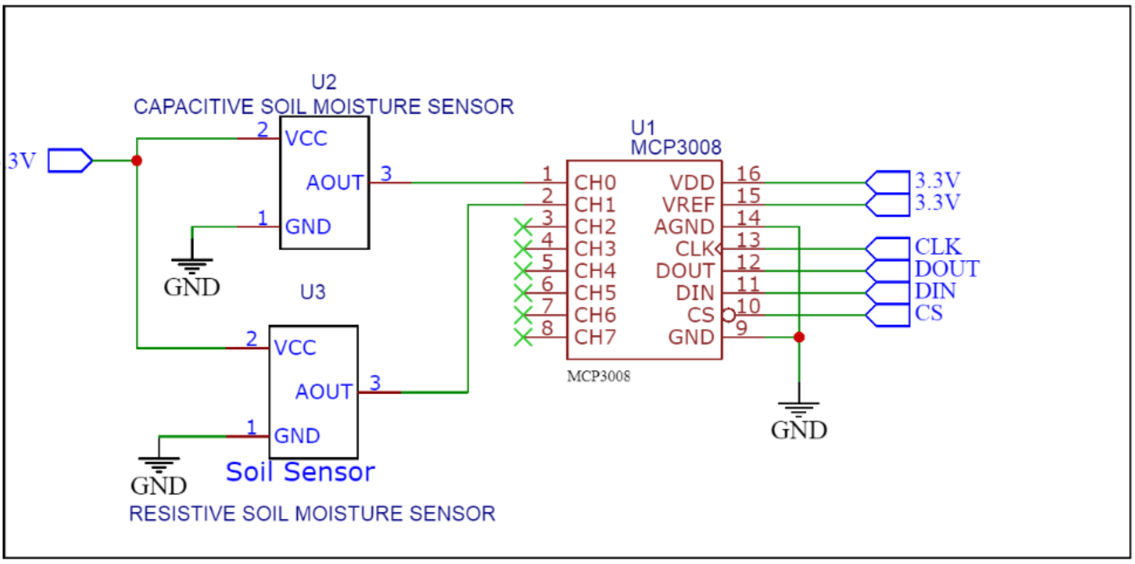


Figure 3. 23 Circuit Connections – Sensor Interfacing with MCP3008

Figure 3.23 shows the circuit connections of the two sensors – Capacitive soil mositure sensor and Resistive soil moisture sesnor interfaced with MCP3008 Analog to Digital Converter connected to the Raspberry Pi 3B+ board. The common VCC of both the sensors are connected to the common incomming 3V3 supply from the Pi to the prototyping board.Similarly, the common GND from the sensors is connected to the common GND on the prototyping board. The Analog outputs from both the sensors are connected to the CH0 and CH1 of the MCP3008 IC hooked on the prototyping board. Only 2 analog channels of the 8 input channels of the MCP3008 are used allowing the possibility to interface 6 additional sensors for future scope of the project dicussed in the Chapter titled Future Scope. The VDD and VREF ports of the MCP3008 are connected to the common 3V3 of the prototyping board. The AGND is connected to the common GND terminal on the prototyping board. The CS(Chip Select Pin) of the MCP3008 IC is connected to GPIO8 of the Pi, it allows us to turn on/off the IC. The pins DOUT, DIN and CLK are connected to the GPIO9, GPIO10 and GPIO11 respectively. These pins are used to establish an SPI(Serial-Peripheral Interface Connection) with the Raspberry Pi 3B+ board. The SCLK pin is used to provide a clock signal for SPI Communication.

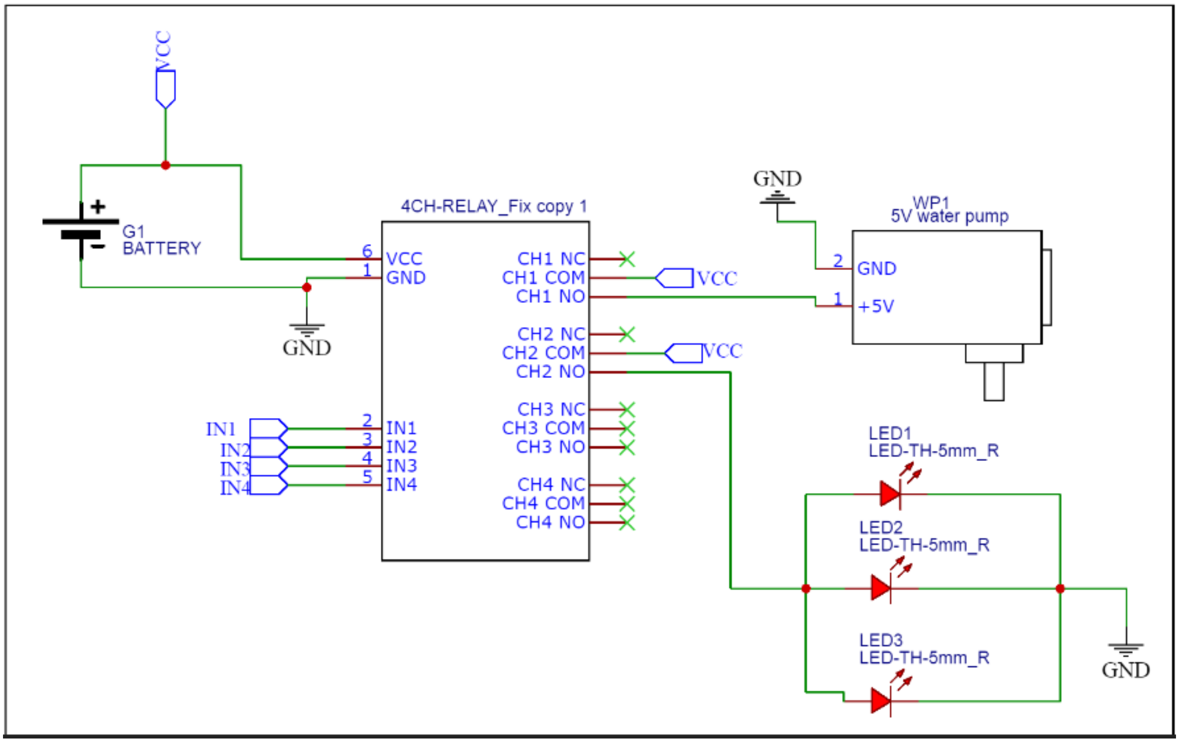


Figure 3. 24 Circuit Connections – Actuators with the Relay Module

Figure 3.24 shows the circuit connections of the Actuators – LED Growing lights and the DC motor pump to the 4 Channel Relay Module controlled by the Raspberry Pi 3B+. The Relay can be powered by the external battery bank since the actuator power requirements are minimal and requires stable 5V, 2.1A supply for consisted operation or alternatively using a 5V, 2.1A power adapter connected to the AC mains supply. A common VCC 5V is chosen on the prototyping board and is connected to the VCC of the Relay Module Board and GND is connected to the common GND on the prototyping board. Similarly, the GND terminals of both the actuators are connected to the common GND on the prototyping board. The 5V terminal of the DC pump is connected to the NO(Normally open) terminal of Channel 1 on the Relay board module and the COM(Common interface) of Channel 1 on the Relay board module is connected to the common VCC on the prototyping board. The LED growing lights connections are similarly attributed to that of the DC motor pump with its 5V terminal connected to the NO(Normally open) terminal of Channel 2 on the Relay board module and the COM(Common interface) of Channel 2 on the Relay board module is connected to the common VCC on the prototyping board. The Relay Inputs IN1, IN2, IN3 and IN4 are connected to GPIO5, GPIO6, GPIO13 and GPIO19 of the Pi respectively. These inputs triggers active low signal on the respective ports used for switching action in the relay board. These respective GPIO’s can be programmed through the software discussed later. Also, GPIO6, GPIO13 and GPIO19 are also used as hardware PWM(Pulse-width modulation) pins discussed earlier.

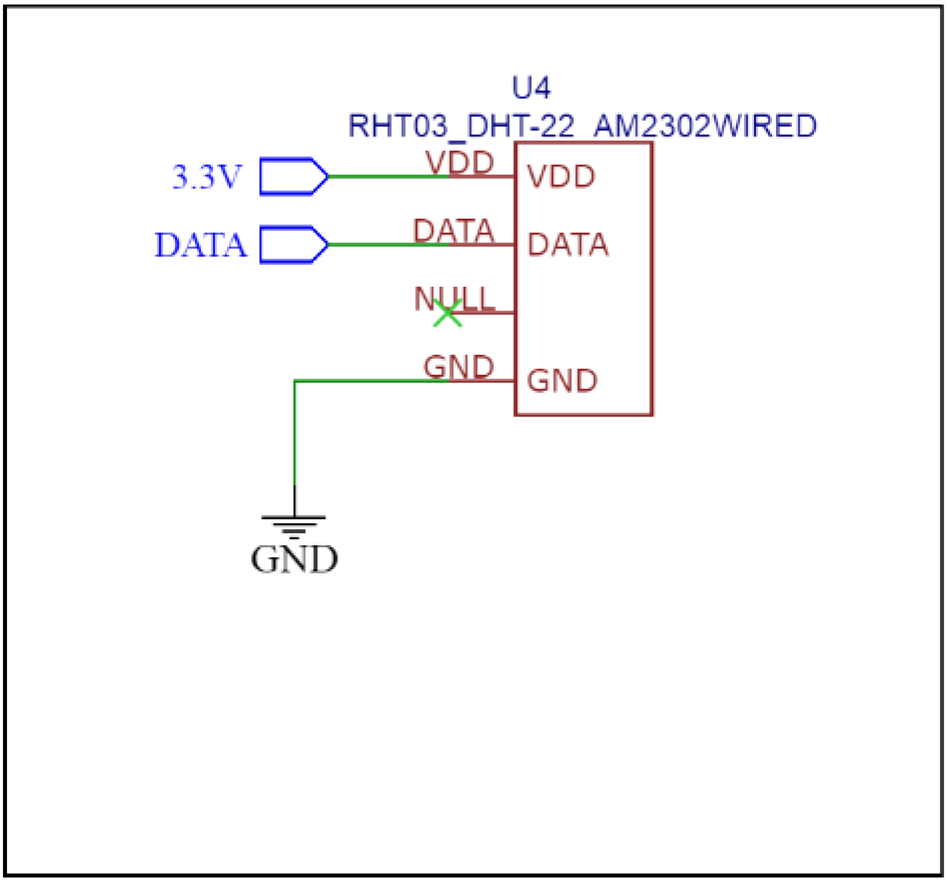


Figure 3. 25 Circuit Connections – DHT22 Temperature/Humidity Sensor

Figures 3.25 shows the circuit between the DHT22 Temperature/Humidity Sensor and the Raspberry Pi 3B+ board. The connections are straightforward with the GND terminal of the sensor connected to the common ground of the prototyping board. The sensor requires a minimun of 3.3V for stable operation which is provided by connecting the VDD port of the sensor to the 3V3 common rail on the prototyping board. All the sensors in the project are powered by the incomming supply from the Raspberry Pi 3B+ board.

3.4 Software Architecture

This Section describes the Software tools and architecture of the connected hardware components. The figure 3.26 shows the flow chart diagram of the project and pictorally depicts the automation logics of the project.

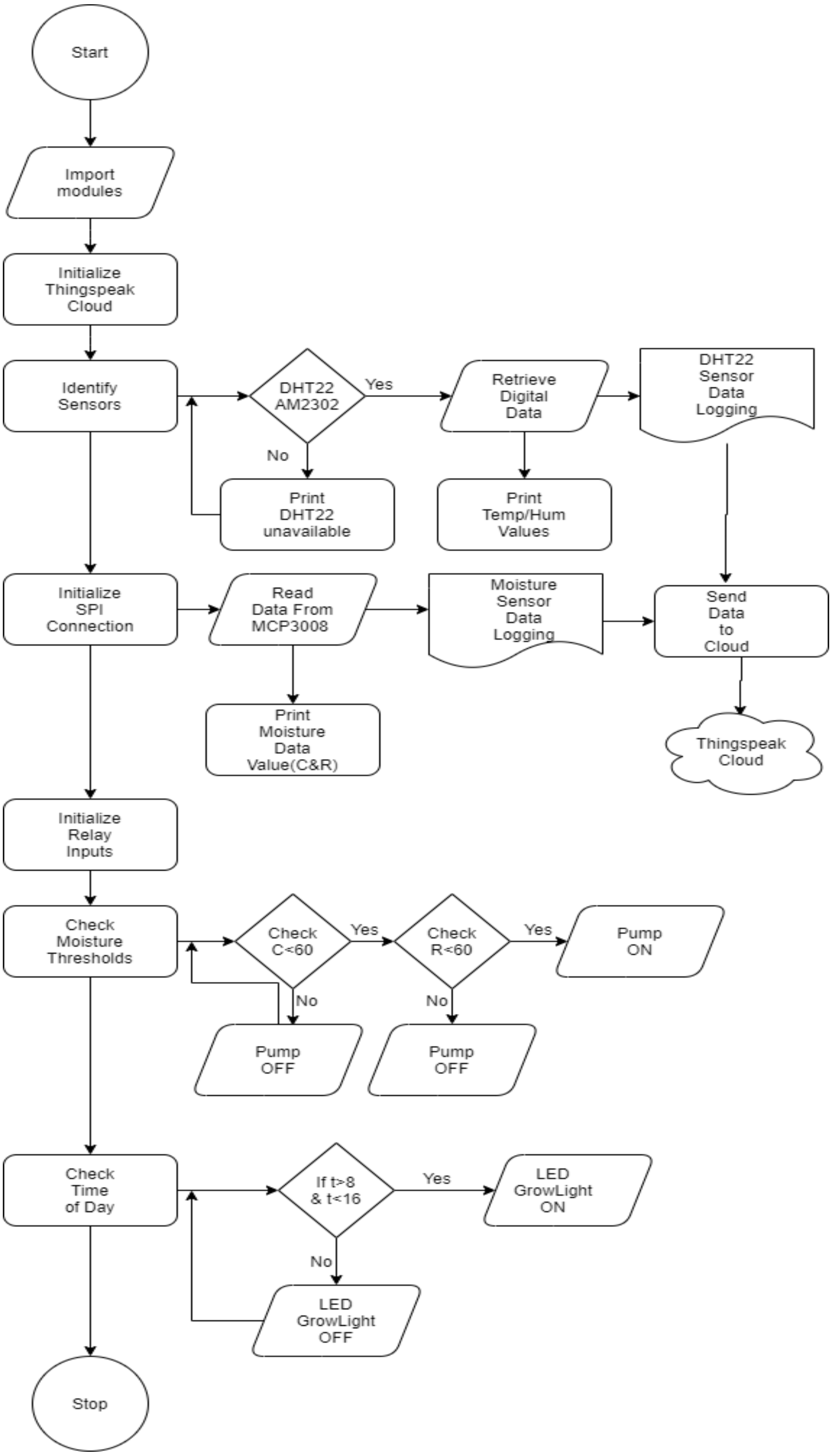


Figure 3. 26 Automated Smart Indoor Farming using IoT – Flowchart Diagram

The figure 3.26 illustrates the flow process of the Software code. The program starts getting executed initially by importing the modules(sooftware packages and libraries). It then initializes the Thingspeak Cloud for future data transmission. The next sequence of the code is to identify the sensors integrated with the system.It checks for DHT22 – AM2302 sensor and if it is connected, it automatically retrieves the digital readings of the ambient temperature and humidity. The code allows the functionality of automatic data logging. If the DHT22 sensor is not connected, it prints the message stating that the sensor is unavailable. This print function is useful aslo in the case when the sensor is damaged or malfunctioning

The next step in the code execution involves the process of establishing an SPI(Serial Peripheral Interface) communication protocol which is important for retrieving the capacitive and restistive mositure sensor readings.The digital moisture data comming in from the MCP3008 Analog to Digital Converter IC is printed and the data readings are automatically logged. The data loggings of all the sensors are now sent to the Thingspeak cloud. The next process involves the initialization of Relay inputs for switching action. The code then check the automation logics – threshold moisture values of Capacitive Mositure Sensor(C) and Resistive Moisture Sensor(R) which are cascaded with the AND logic. If both the moisture sensor readings falls below the threshold value of 60 meaning mositure percentage less than 60%, the pump is turned ON. If any of the moisture sensor or both the moisture sensor values are greater than 60, the pump remains in OFF state. This technique ensures robustness of the system incase if any of the moisture sensors is malfunctioning.

The last step of the code execution is to check the time of the day for controlling the LED Growlights. If the time(t) > 8:00 and time(t) < 16:00, the LED Growlights turn ON and remains in the OFF state otherwise. The automation logics can be easily changed anytime by just modification of few lines of code based on the design contraints or depending upon the growing conditions of a particular crop/herb.

3.4.1 Remote Desktop Connection

The Rapberry Pi Remote Desktop can be accessed with many techiniques discussed earlier. Alternatively, the Pi Desktop can be accessed by connecting the HDMI port of the Pi to the external Monitor. Connecting the Pi to the external monitor is not a viable option as most oftenly the Pi is deployed in remote areas giving user the power to acess the Pi desktop remotely.

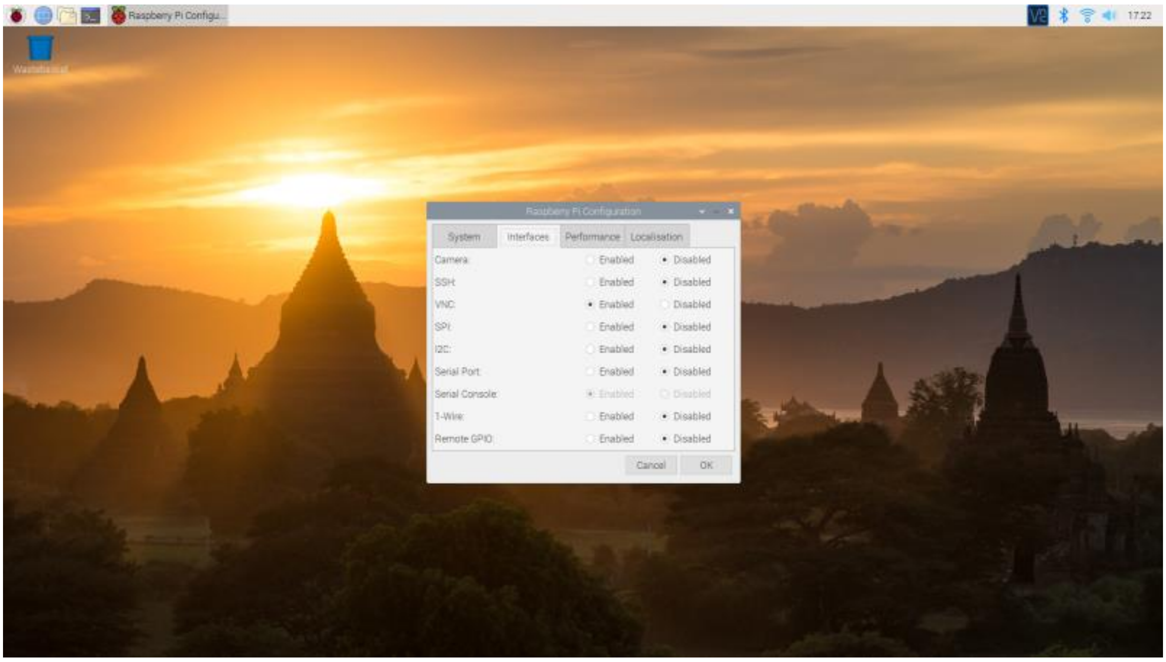
Establishing Remote Desktop Connection

Method 1: RealVNC Server

The following steps are performed to establish remote desktop connection using RealVNC Server [34].

Step 1: Enable VNC

Firstly, the Raspberry Pi and the other computer used for remote connection are connected to the same network. In Raspbian OS, click the applications menu icon (raspberry) at the top-left of the screen and select Preferences > Raspberry Pi Configuration.



Camera:	○ Enabled	● Disabled
SSH:	○ Enabled	● Disabled
VNC:	● Enabled	○ Disabled
SPI:	○ Enabled	● Disabled
I2C:	○ Enabled	● Disabled
Serial Port:	○ Enabled	● Disabled
Serial Console:	● Enabled	○ Disabled
1-Wire:	○ Enabled	● Disabled
Remote GPIO:	○ Enabled	● Disabled

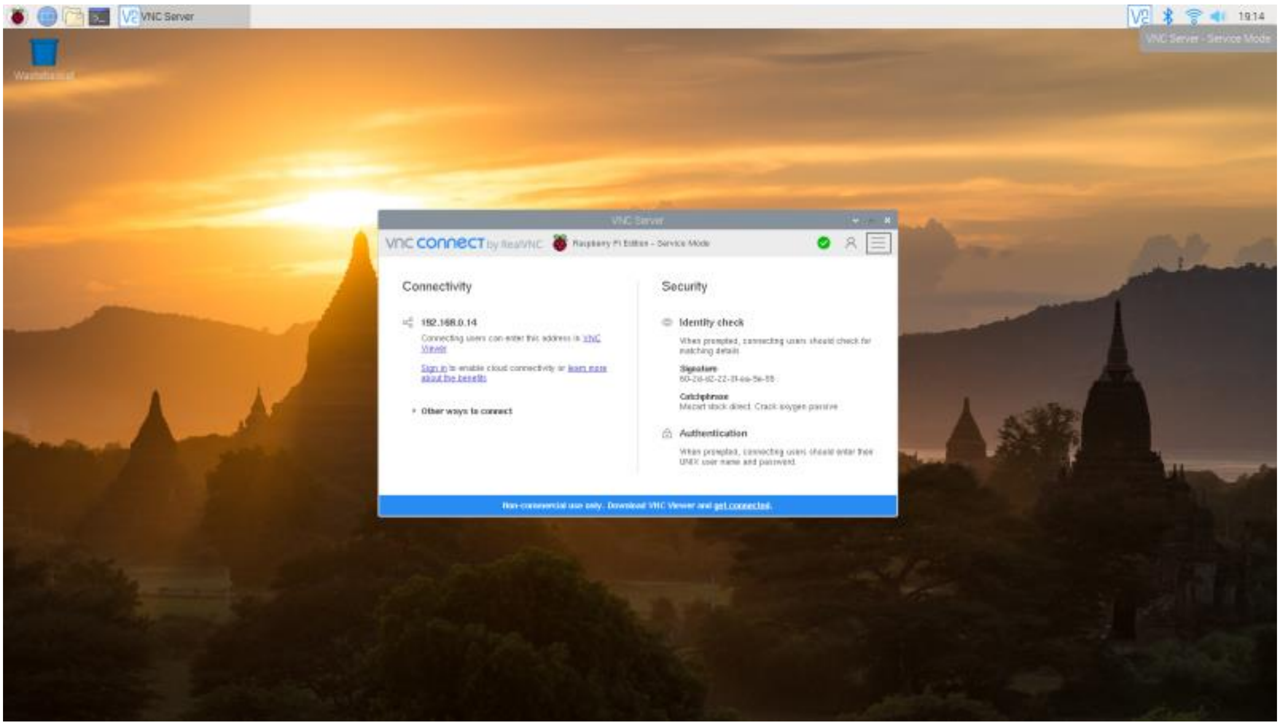
Figure 3. 27 Raspberry Pi Configuration – Enabling VNC Server

The Raspbian password from the default ‘raspberry’ can be changed in this setting. The option ‘Change Password’ allows the user to modify the default password.

In the Interfaces taps, the radio button next to VNC is enabled and at the end of the process a VNC Button appears in the top right corner of the display screen. Clicking on it enables the user to open the VNC Server. The IP (Internet Protocol) address will appear in the VNC Server Window (below connectivity). The four digits displayed will ensure the user to locate the Raspberry Pi over the network. The network can be a standard home LAN network or a Wi-Fi network.

Step 2: Open VNC Viewer

The Raspberry Pi is now ready to be connected from other Windows PC, MAC, Linux PC or even another Raspberry Pi on the same network. Since the VNC Server is now established, an application VNC Viewer is required to view the remote desktop shared across the VNC server. VNC viewer is available for macOS, Linux, Android, and iOS platforms along with a web interface.


VNC Server
Wastebasket
19:14
VNC Server - Service Mode
VNC Server
VNC CONNECT by RealVNC
Raspberry Pi Edition – Service Mode

Connectivity

192.168.0.14

Connecting users can enter this address in VNC Viewer

Sign in to enable cloud connectivity or learn more about the benefits

Other ways to connect

Security

Identity check

When prompted, connecting users should check for matching details

Signature
60-28-82-22-31-ea-5e-55

Catchphrase
Mozart stock direct. Crack oxygen passive

Authentication

When prompted, connecting users should enter their UNIX user name and password.

Non-commercial use only. Download VNC Viewer and get connected.

Figure 3. 28 Raspberry Pi Configuration – VNC Server

Step 3: Connect to Raspberry Pi

The IP address of the Raspberry Pi (the four digits displayed in the VNC Server) is entered into the search bar of VNC Viewer. The Return option in the application is chosen to connect to the Raspberry Pi. Initially a warning message appears stating that VNC has no record of connecting to this VNC Server so it is identity cannot be checked. When continue option is chosen, the Pi prompts the user to enter the default access credentials used to sign into the Pi. Further selecting remember option simplifies the process of establishing a reconnection in the future.

Step 4: Remote Control

A window appears on the Windows PC displaying the Raspbian desktop interface from the Raspberry Pi. The Pi is now securely connecting via the remote VNC server and it can now be remotely controlled over the network.

Method 2: TeamViewer

TeamViewer [35] is one of the most commonly used software to access the screen of some other computer remotely. It is similar to RealVNC but with additional features like text messaging and video conferencing while accessing the systems. This application is frequently used by support teams as it makes it easier to diagnose and fix the problems when the pi is deployed remotely. This is the best way to manage the system through multiple users at different locations.

Installing TeamViewer on Raspberry Pi

Step 1: Checking whether installed packages and OS are up to date.

Firstly, the terminal is opened in the Raspberry Pi OS.

The following command is entered in the terminal, it ensures that the package list and all installed packages are up to date.

```
sudo apt-get update
sudo apt-get upgrade
```

Step 2: Downloading the software packages from Teamviewer

After upgrading the packages, the software packages are download from TeamViewer server using the wget command. The downloaded file will be a .deb file which is an archive file containing all the required packages for TeamViewer.

```
wget
http://download.teamviewer.com/download/linux/teamviewer-
host_armhf.deb
```

Step 3: Installing the downloaded file

The downloaded file can be installed using dpkg command.

```
sudo dpkg -i teamviewer-host_armhf.deb
```

Step 4: Fixing errors

After installing the teamviewer, there might be some erros and it can be fixed using the package manager to repair the files.

```
sudo apt –fix-broken install
```

If the entire process is carried out perfectly, the TeamViewer Host will automatically start with every boot of the Raspberry Pi.

Step 5: Configuring Teamviewer on Raspberry Pi

The setup for Teamviewer can be done through either the command line or using the GUI.

In terminal, the following command is entered.

```
sudo teamviewer passwd <password>
```

After setting up password, a Teamviewer id is required to establish the connection between the Raspberry Pi and the system. To view the Teamviewer id, the following command is entered in the terminal

```
teamviewer info
```

This id and password can be used to enter in the teamviewer application remotely.

3.4.2 Software Design – an approach using Python Programming Language

The first important step before identifying the software architecture requires the software configuration of the Raspberry Pi 3B+ board discussed in the section titled Configuring the Rapsberry Pi discussed earlier. Once the software configuration of the Pi is completed, the heavyduty of programing is handled by the Python Programming language in the GNU Nano editor preinstalled on the Raspbian Linux OS.

Installing the latest Python 3 on Raspberry Pi

Python is a very powerful but easy to learn programming language. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python has very efficient high-level data structures and a simple but effective approach to (OOP) object-oriented programming. Python's design philosophy is based on code readability with its notable use of significant indentation. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms like windows, macOS, Linux.

To install the Python 3 on Raspberry Pi, the following procedure is carried out.

Step 1: Installing the dependencies needed to build.

To install the dependencies needed to build, the following command is entered.

```
sudo apt-get update
sudo apt-get install -y build-essential tk-dev libncurses5-dev libncursesw5-dev libreadline6-dev libdb5.3-dev libgdbm-dev libsqlite3-dev libssl-dev libbz2-dev libexpat1-dev liblzma-dev zlib1g-dev libffi-dev
```

Step 2: Compilation

The following command is entered to compile.

```
wget https://www.python.org/ftp/python/3.8.0/Python-3.8.0.tar.xz
```

```
tar xf Python-3.8.0.tar.xz
cd Python-3.8.0
./configure --prefix=/usr/local/opt/python-3.8.0
make -j 4
```

Step 3: Installation

Finally, the installation requires the following code is entered.

```
sudo make altinstall
```

Step 4: Removing the files

The files which are created during the installation procedure can be deleted using the following command.

```
cd ..
sudo rm -r Python-3.8.0
rm Python-3.8.0.tar.xz
. ~/.bashrc
```

Step 5: Enabling Python 3.8 the default version and verification.

To enable Python 3.8 as the default version, the following command is entered.

```
sudo update-alternatives --config python
```

It can then be verified using the following command.

```
python -V
```

3.4.3 Interfacing MCP3008 Analog to Digital Converter IC with the Raspberry Pi

To enable the SPI communication protocol [36] using the Raspberry pi, the following steps are required.

Step 1: Open the Terminal of the Pi

The below command opens the configuration setting of the Pi.

```
sudo raspi-config
```

Step 2: Interfacing Options

The interfacing options are selected from the command center of the configuration settings.

Step 3: Enabling SPI Interface

The SPI interface is enabled by toggling the enable option to form the sub-menu of the Interfacing options.

Step 4: Installing NumPy

NumPy is used to scale the values.

NumPy is installed by typing the below command in the Terminal.

```
sudo apt-get install python-numpy
```

The '`interp(output, [0, 1023], [0, 100])`' function is used to scale the values from 0-1023 to 0-100. The output we get from the MCP3008 ADC is in the range of 0-1023 but we can give a duty cycle of 0-100. Hence, the values are scaled down in this range.

3.4.4 Interfacing the DHT22(Temperature/Humidity Sensor) with the Raspberry Pi

For the DHT22(Temperature/Humidity Sensor) to work properly and be recognized by the Pi, a suitable Library and build packages [37] should be installed. The following procedure illustrates the installation of the build packages and libraries.

The following commands are entered in the terminal command center to install the build packages and libraries.

```
sudo apt update
sudo apt install build-essential python-dev
git                                                        clone
https://github.com/adafruit/Adafruit_Python_DHT.git
pi@raspberrypi:~/Desktop/Libraries $ cd Adafruit_Python_DHT
pi@raspberrypi:~/Desktop/Libraries/Adafruit_Python_DHT     $
sudo python setup.py install
```

3.4.5 Interfacing the Raspberry Pi to the ThingSpeak IoT Cloud Platform

The Raspberry Pi can log the sensor data and can also send the sensor data to the cloud. For the Raspberry pi to send data in real-time to the cloud dashboard, the thingspeak application should be installed.

The following commands are entered in the terminal command center to install the Thingspeak cloud interface [38].

```
sudo pip install thingspeak
```

Configuring the ThingSpeak IoT Cloud Platform

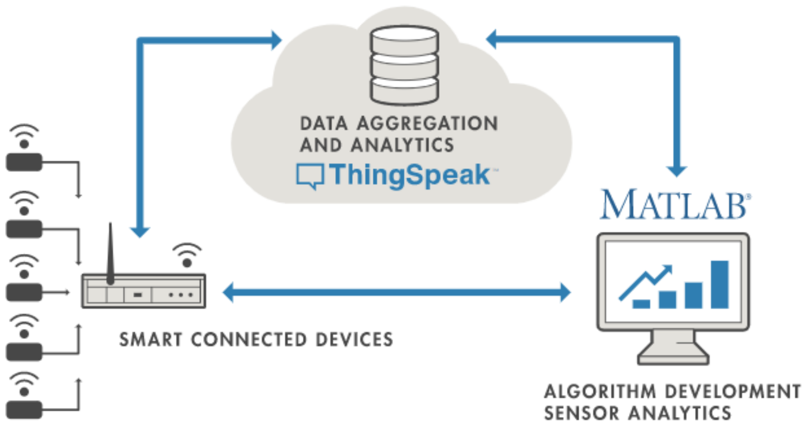


Figure 3. 29 Thingspeak IoT Cloud Platform - Architecture

ThingSpeak Configuration

Step 1: Sign up for new User Account or use existing MathWorks account.
http://thingspeak.com/users/sign_up

Step 2: Create a new Channel.
By Selecting “Channels", "My Channels", and then "New Channel".

Step 3: Enter Channel Information
Name: data_sensor
Description: DHT22 temperature/ humidity sensor and moisture sensors driven by a Raspberry Pi 3B+ running a Python script.

Field 1: Temperature (°C)
Field 2: Humidity (%)
Field 3: Resistive Sensor (VWC%)
Field 4: Capacitive Senor (VWC%)

Step 4: Record Write API Key
Once the Channel is created, click on "API Keys" and the "Write API Key" is noted for future use in the software script.
The Write API Keys are entered in the Software code and it automatically gets enabled when the code imports the thingspeak module during execution.

Chapter 4

System Testing and Performance Evaluation

This chapter discusses the system testing and performance evaluation under different test case scenarios. The desired results and the experimental results of the testing process are compared and Analyzed. The Figure 4.1 illustrates the functional test results of one of the possible test cases and the complete experimental setup of the IoT System.

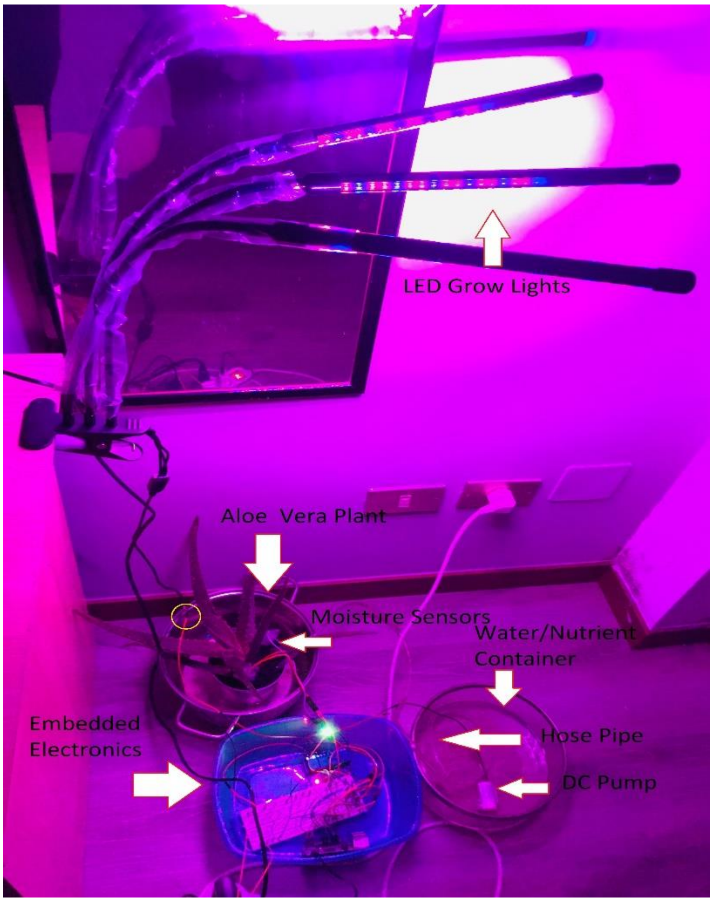


Figure 4. 1 Experimental Setup - Prototype

In the Experimental Setup show in Figure 4.1, an Aloe-Vera Plant is considered, and its growing conditions include a minimum humidity levels of 60%, temperature of 20°C. These conditions are fed to the Automation Algorithm which fully automates the system for perfect harvest. The capacitive and Soil Moisture Sensors are placed inside the Soil. These sensors collect the moisture values at a sample period of 20ms. The blue box contains the embedded electronics – the Raspberry Pi single board computer, the Relay module board and the prototyping board electronically connected. The 3 Goosenecks of the LED Grow Lights deliver the Light Spectrum Intensity Levels. The yellow circle indicates the modification of the Grow Lights to perform the desired switching actions based on the Automation Algorithm. The glass container consists of water/nutrient solution with a micro submersible pump inside it. The pump is connected to a PVC water hose pipe. The pump delivers the water/nutrient solution based on the threshold moisture values. The pumping frequency is automatic and solely depends on the moisture levels of the sensors.

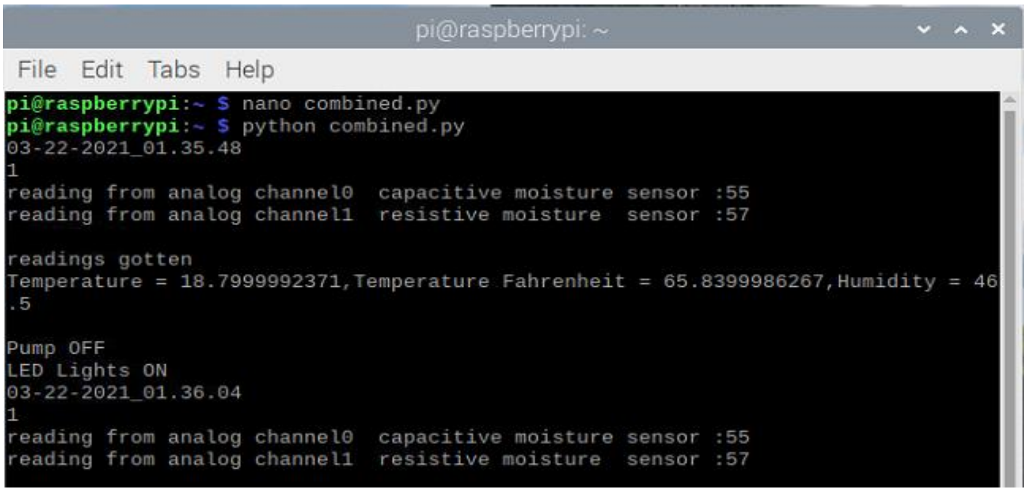
4.1 System Testing

The moisture sensor values(mv) correspond to both the capacitive moisture sensor and resistive moisture sensor reading since to reduce the complexity of the software algorithm and for robustness – AND logic is used. This means that both the sensors should fall above/below the threshold limit(θ) and preassigned time interval ($t > 0$ and $t <8$) between 00:00 mid-night and 08:00 in the morning for switching action. If any of the moisture sensor fails during operation, the automation algorithm will only work when both the sensors are giving meaningful values. Otherwise, the system may malfunction and might also cause water/nutrient flooding and damage the plant and indoor environment.

The System State is classified under the following four Test conditions:

I. Test Case: $mv > \theta \wedge (t > 0 \wedge t < 8)$

If the moisture sensor values are above the threshold and the time of the day is in the preassigned time interval. The figures 4.2 illustrates the simulation results of this test case.



```
pi@raspberrypi:~ $ nano combined.py
pi@raspberrypi:~ $ python combined.py
03-22-2021_01.35.48
1
reading from analog channel0  capacitive moisture sensor :55
reading from analog channel1  resistive moisture  sensor :57

readings gotten
Temperature = 18.7999992371,Temperature Fahrenheit = 65.8399986267,Humidity = 46
.5

Pump OFF
LED Lights ON
03-22-2021_01.36.04
1
reading from analog channel0  capacitive moisture sensor :55
reading from analog channel1  resistive moisture  sensor :57
```

Figure 4. 2 Test Case - $mv > \theta \wedge (t > 0 \wedge t < 8)$

The Capacitive and Resistive moisture sensor values are above the threshold limit of 60 and the LED Lights are turned ON because the time of the day falls in the preassigned time interval.

Desired Behavior:

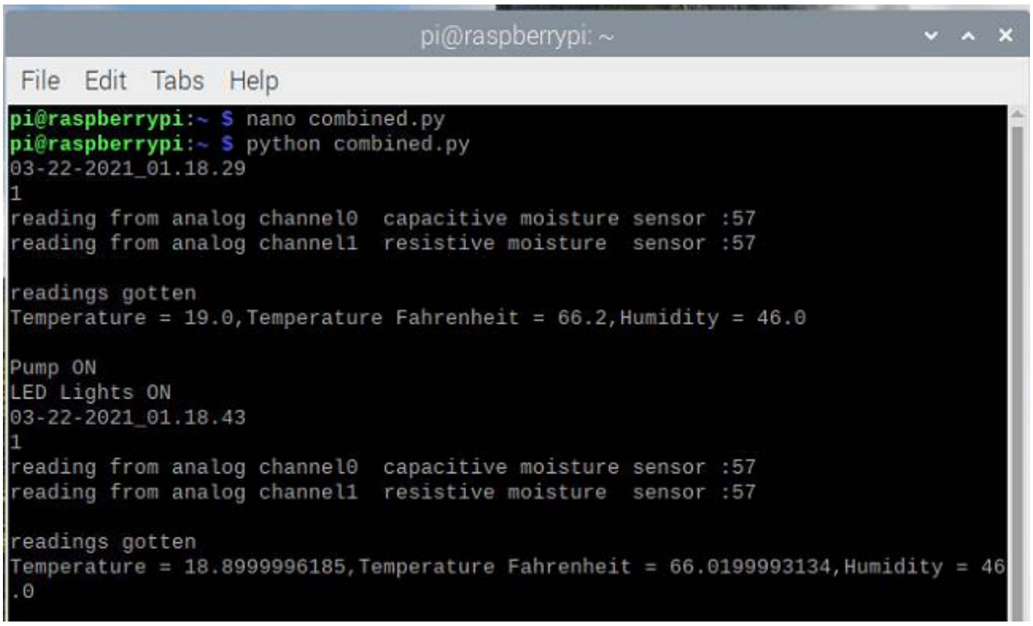
Pump OFF

Lights ON

Clearly, the simulation results coincide with the desired behavior of the system in this test case.

II. Test Case: mv < θ ∧ (t > 0 ∧ t < 8)

If the moisture sensor values are below the threshold and the time of the day is in the preassigned time interval.



```
pi@raspberrypi:~ $ nano combined.py
pi@raspberrypi:~ $ python combined.py
03-22-2021_01.18.29
1
reading from analog channel0  capacitive moisture sensor :57
reading from analog channel1  resistive moisture  sensor :57

readings gotten
Temperature = 19.0,Temperature Fahrenheit = 66.2,Humidity = 46.0

Pump ON
LED Lights ON
03-22-2021_01.18.43
1
reading from analog channel0  capacitive moisture sensor :57
reading from analog channel1  resistive moisture  sensor :57

readings gotten
Temperature = 18.8999996185,Temperature Fahrenheit = 66.0199993134,Humidity = 46
.0
```

Figure 4. 3 Test Case - mv < θ ∧ (t > 0 ∧ t < 8)

The Capacitive and Resistive moisture sensor values are below the threshold limit of 60, so the pump is ON and the LED Lights are turned ON because the time of the day falls in the preassigned time interval.

Desired Behavior:

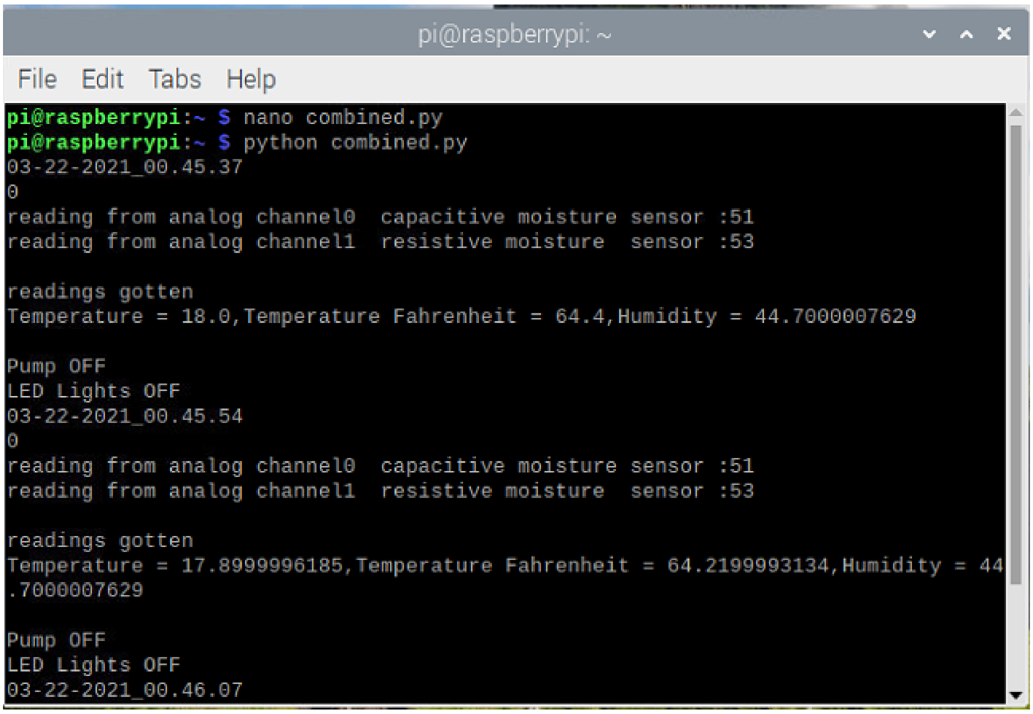
Pump ON

Lights ON

Clearly, the simulation results coincide with the desired behavior of the system in this test case.

III. Test Case: mv > θ ∧ ⌐(t > 8 ∧ t < 16)

If the moisture sensor values are above the threshold and the time of the day is not in the preassigned time interval. Also, in this test case the threshold moisture value is set to 50 and the preassigned time interval is changed to (t > 8 and t <16) between 08:00 in the morning and 16:00 in the evening for switching action.



```
pi@raspberrypi:~ $ nano combined.py
pi@raspberrypi:~ $ python combined.py
03-22-2021_00.45.37
0
reading from analog channel0  capacitive moisture sensor :51
reading from analog channel1  resistive moisture  sensor :53

readings gotten
Temperature = 18.0,Temperature Fahrenheit = 64.4,Humidity = 44.7000007629

Pump OFF
LED Lights OFF
03-22-2021_00.45.54
0
reading from analog channel0  capacitive moisture sensor :51
reading from analog channel1  resistive moisture  sensor :53

readings gotten
Temperature = 17.8999996185,Temperature Fahrenheit = 64.2199993134,Humidity = 44
.7000007629

Pump OFF
LED Lights OFF
03-22-2021_00.46.07
```

Figure 4. 4 Test Case - mv > θ ∧ ⌐(t > 8 ∧ t < 16)

The Capacitive and Resistive moisture sensor values are above the threshold limit of 50, so the pump is OFF and the LED Lights are turned OFF because the time of the day does not fall in the preassigned time interval.

Desired Behavior:

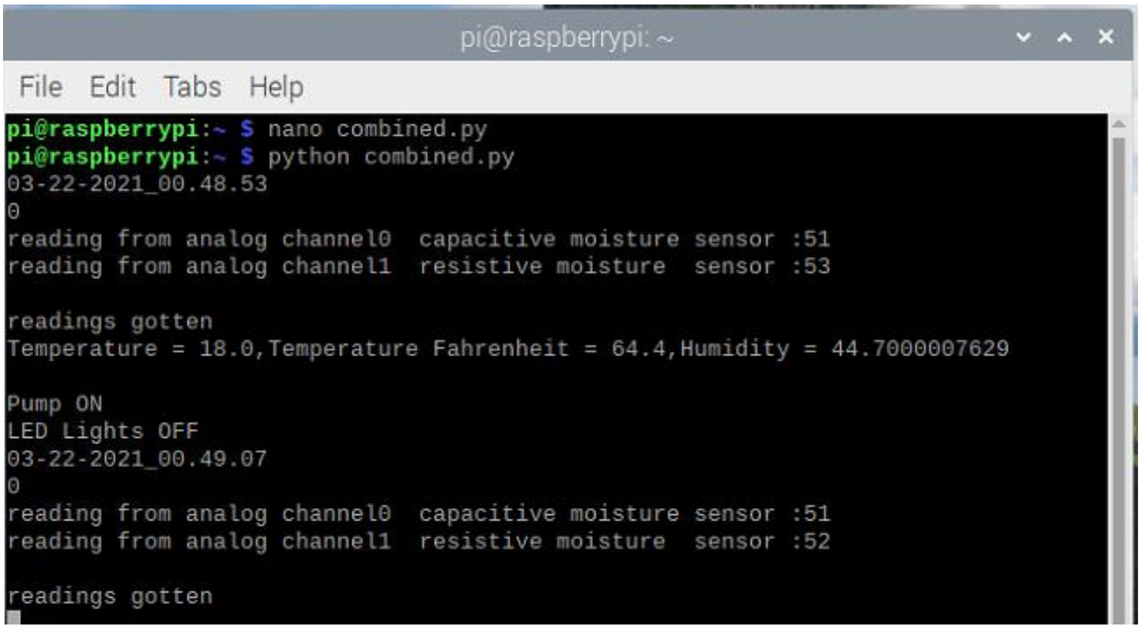
Pump OFF

Lights OFF

Clearly, the simulation results coincide with the desired behavior of the system in this test case.

IV. Test Case: mv < θ ∧ ⌐(t > 8 ∧ t < 16)

If the moisture sensor values are below the threshold and the time of the day is not in the preassigned time interval. Also, in this test case the threshold moisture value is set to 80 and the preassigned time interval is changed to (t > 8 and t <16) between 08:00 in the morning and 16:00 in the evening for switching action.



```
pi@raspberrypi:~ $ nano combined.py
pi@raspberrypi:~ $ python combined.py
03-22-2021_00.48.53
0
reading from analog channel0  capacitive moisture sensor :51
reading from analog channel1  resistive moisture  sensor :53

readings gotten
Temperature = 18.0,Temperature Fahrenheit = 64.4,Humidity = 44.7000007629

Pump ON
LED Lights OFF
03-22-2021_00.49.07
0
reading from analog channel0  capacitive moisture sensor :51
reading from analog channel1  resistive moisture  sensor :52

readings gotten
```

Figure 4. 5 Test Case - mv < θ ∧ ⌐(t > 8 ∧ t < 16)

The Capacitive and Resistive moisture sensor values are below the threshold limit of 80, so the Pump is ON and the LED Lights are turned OFF because the time of the day does not fall in the preassigned time interval.

Desired Behavior:

Pump ON

Lights OFF

Clearly, the simulation results coincide with the desired behavior of the system in this test case.

All the test cases are summarized in the table 4.1 below.

Test Case / Act. State	**Pump**	**LED Lights**
mv > θ ∧ (t > 0 ∧ t < 8)	OFF	ON
mv < θ ∧ (t > 0 ∧ t < 8)	ON	ON
mv > θ ∧ ⌐(t > 8 ∧ t < 16)	OFF	OFF
mv < θ ∧ ⌐(t > 8 ∧ t < 16)	ON	OFF

Table 4. 1 Test Cases vs Actuator State

4.2 ThingSpeak IoT Data Visualization

ThingSpeak is an Internet of Things (IoT) platform enabling the collection, storage, analysis, and visualization of data gathered from sensors integrated to the system. In the project, there are 3 sensors collecting 4 data channels.

ThingSpeak stores data in Channels, each of which can have up to 8 data fields, as well as location and status fields. Data can be sent to ThingSpeak at a rate of at every 15 seconds or longer.

DHT22 AM2302 Temperature/ Humidity Sensor

This sensor collects both the temperature and humidity values.

Temperature Values are sent to Field 1 of the ThingSpeak channel (Figure 4.6)

Humidity Values are sent to Field 2 of the ThingSpeak channel (Figure 4.7)

Resistive Soil Moisture Sensor

This sensor collects the moisture values.

Resistive Moisture Sensor Values are sent to Field 3 of the ThingSpeak channel (Figure 4.8)

Capacitive Soil Moisture Sensor

This sensor collects the moisture values.

Capacitive Moisture Sensor Values are sent to Field 4 of the ThingSpeak channel (Figure 4.9)

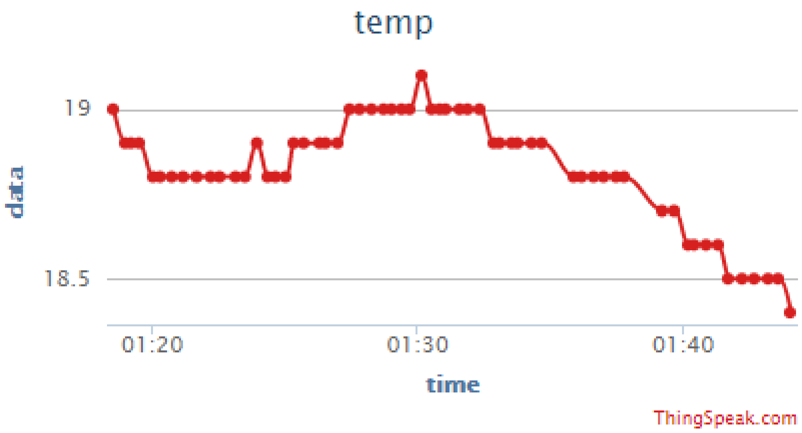


Figure 4. 6 DHT22 Sensor (Temperature) – Time Series Visualization

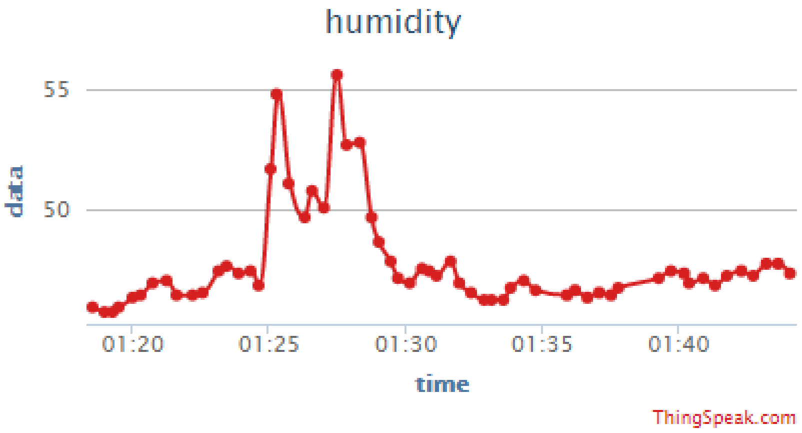


Figure 4. 7 DHT22 Sensor (Humidity) – Time Series Visualization

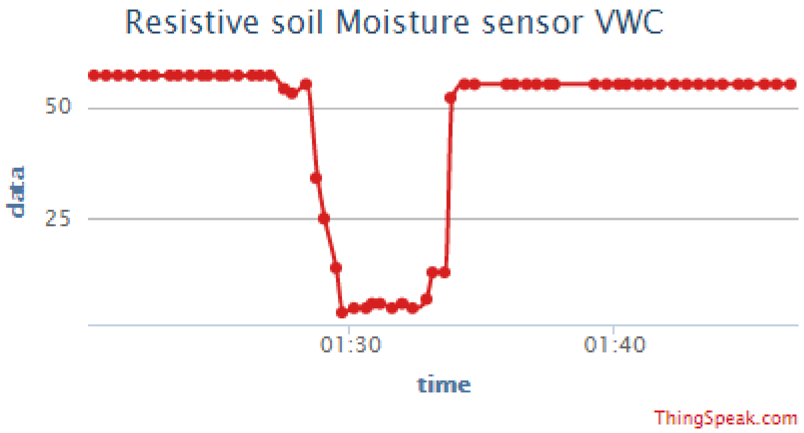


Figure 4. 8 Resistive Soil Moisture Sensor (Moisture %) – Time Series Visualization

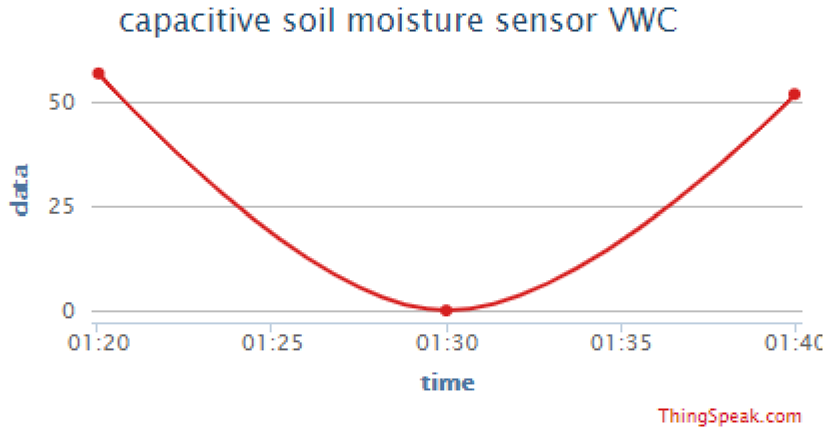


Figure 4. 9 Capacitive Soil Moisture Sensor (Moisture %) – Time Series Visualization

From the above graphical visualizations, we can conclude that the automation system tries to stabilize the moisture values of the system near the threshold value of 60%. In the test case at 01:30 time, dry soil was added and immediately the algorithm detects a zero-moisture reading. Hence, the irrigation was scheduled by automatically pumping the water/nutrient mixture to maintain desired moisture levels.

4.3 System Power Consumption and Energy Cost Calculation

4.3.1 Power Consumption and Energy Cost of Raspberry Pi

The following table 4.2 lists the Power consumption of Raspberry Pi Model 3B+ [39] based on its state:

Pi State	**Power Consumption**
Idle	350 mA (1.9W)
ab -n 100 -c 10 (uncached)	950 mA (5.0W)
400% CPU load (stress –cpu 4)	980 mA (5.1 W)

Table 4. 2 Pi State vs Power Consumption

The energy E in kilowatt-hours (kWh) per day is equal to the power P in watts (W) times number of usage hours per day t divided by 1000 watts per kilowatt:

$E_{(kWh/day)} = P_{(W)} \times t_{(h/day)} / 1000_{(W/kW)}$

$E_{(kWh/day)} = (5,0 \times 24 / 1000)$

$E_{(kWh/day)} = 0,12$ kWh/day

Energy cost calculation

The energy cost per day in euros is equal to the energy consumption E in kWh per day times the energy cost of 1 kWh in cents/kWh divided by 100 cents per euro:

Energy cost of 1 kWh in cents/kWh in Italy = 22,26 cents/kWh

$Cost_{(€/day)} = E_{(kWh/day)} \times Cost_{(cent/kWh)} / 100_{(cent/€)}$

$Cost_{(€/day)} = (0,12 \times 22,26 / 100)$ €

$Cost_{(€/day)} = 0,026712$ € per day.

$Cost_{(€/month)} = (0,026712 \times 30)$

$Cost_{(€/month)} = 0,80136$ € per month.

4.3.2 Power Consumption of Actuators

There are two actuators used in the system – **DC Pump and LED GrowLights.**

The Power(W) of the DC pump is 0.8W and due to its minimal power consumption, its energy cost can be neglected.

The Power(W) of the LED GrowLights is 30W and it is turned on for minimum 8 hours a day $t_{(h/day)}$.

The energy E in kilowatt-hours (kWh) per day is equal to the power P in watts (W) times number of usage hours per day t divided by 1000 watts per kilowatt:

$E_{(kWh/day)} = P_{(W)} \times t_{(h/day)} / 1000_{(W/kW)}$

$E_{(kWh/day)} = (30 \times 8 / 1000)$

$E_{(kWh/day)} = 0{,}24$ kWh/day

Energy cost calculation

The energy cost per day in euros is equal to the energy consumption E in kWh per day times the energy cost of 1 kWh in cents/kWh divided by 100 cents per euro:

Energy cost of 1 kWh in cents/kWh in Italy = 22,26 cents/kWh

$Cost_{(€/day)} = E_{(kWh/day)} \times Cost_{(cent/kWh)} / 100_{(cent/€)}$

$Cost_{(€/day)} = (0{,}24 \times 22{,}26 / 100)$ €

$Cost_{(€/day)} = 0{,}053424$ € per day.

$Cost_{(€/month)} = (0{,}053424 \times 30)$

$Cost_{(€/month)} = 1{,}60272$ € per month.

The total Energy Cost of the entire System is approximately 2,4 € per month (Energy cost of Raspberry Pi + Energy cost of the Actuators) which is very minimal and hence, the entire system is Energy and Cost efficient.

4.3.3 Autonomy on Battery Bank

To ensure Fail Safe operation mode, the Raspberry Pi should be continously powered. In areas of frequent power outages, the Raspberry Pi can be powered by the Battery Bank.

From the table 4.2, Power consumption of the Raspberry Pi = 950 mA (5,0W)

Battery Capacity(mAh) = 10000 mAh

Total Autonomy in Hours = Battery Capacity(mAh)/ Power Consumption(mAh)

Total Autonomy = (10000/950)h = 10,5 hours

Therefore, the Raspberry Pi can be powered for a minimum of 10,5 hours on a Battery Bank in areas of frequent power outages.

4.4 Cost Analysis

Table 4.3. *Cost analysis of the IoT System*

Parts	
Raspberry Pi 3B+	€37,65
Micro SD card	€7,99
DHT22 Sensor	€2,89
Capacitive Soil Moisture Sensor v1.2	€3,51
Resistive Soil Moisture Sensor	€2,89
Infinity LED Grow Lights	€8,00
MCP3008 Analog to Digital Converter IC	€1,99
Micro submersible DC pump(3-6V)	€1,58
Syska 10000 mAh battery bank	€6,12
Cooling Fan DC	€1,01
4 Channel 5V Relay Module Board	€3,61
Power Cables	€1,50
Jumper Wires	€2,51
Prototyping Board	€1,21
Water Hose	€1,02
Total Cost per Unit	**€83,48**

Table 4. 3 Cost Analysis

From the table 4.3, **the total cost of the sytem is less than €100 making it cost effective solution for Indoor Farming applications.**

Chapter 5

Future Scope and Conclusion

Humans are already on the verge to settle on Mars. Accoring to a recent arcticle, the SpaceX founder Elon Musk hinted that the first colonies could use hydroponics to grow plants indoors. These installations would likely sit inside the specially-built habitats powered by the underground solar panels , a necessary component of Martian life before terraforming occurs while also establishing basic life support systems. Indoor Farming could ensure the first human visitors get a healthy dose of green veggies.

Figure 5. 1 NASA Vision of Indoor Farming on Mars.

The global indoor farming technology market is estimated at USD 14.5 billion in 2020 and is projected to grow at a CAGR of 9.4%, to reach USD 24.8 billion by 2026[40].

Indoor farming technology is ready to cater the local food supplies and provide nutritious, fresh produce to consumers. This method of farming controls plant-fertilizing nutrients making the food highly delicious and nutritious.

In indoor farming, a wide variety of crops such as leafy vegetables, herbs, vegetables, fruits, microgreens, and flowers can be grown. Indoor vertical farming systems provide organic food that is not contaminated with agrochemicals. These form the major driving forces for the indoor vertical farming along with the increasing consumer demand for pesticide- and herbicide-free food.

According to a survey "The fruits & vegetables in the crop type segment is estimated to account for the largest share in 2020 in the indoor farming technology market."

Future Scope of the project can be broadly identified into the following possibilities:

- A Renewable Energy source can be used for powering the IoT System.
- A Raspberry Pi camera can be interfaced with the IoT System for identifying growth defects in the Plants.
- Computer Vison and Machine Learning Algorithms can pre-identify growth defects and improve the overall yield of the herbs or crops.
- A web and mobile application can be developed where the user can select a desired crop from a list or image recognition from the Pi camera and the algorithm automatically adopts the plant growing conditions.
- A large green house project can be implemented by adopting similar design techniques used in this project.
- ThingSpeak IoT Platform gives the possibility for MATLAB Analysis and implement Machine Learning Algorithms to predict sensor values and schedule smart irrigations.

In Conclusion, the problem of food scarcity, deforestation, and infertile lands due to industrial pollution can be tackled by developing novel innovative solutions for smart farming. **The project titled "Automated Smart Indoor Farming using IoT" is an initiative to tackle these problems by allowing plant growers to develop crops, fruits, herbs, and vegetables throughout the year irrespective of changing climatic conditions**. The automated plant growing conditions simulation is a very new concept and there exists very few scientifically and commercially proposed solutions.

Acronyms

IoT	Internet of Things
EC_b	Bulk Electrical Conductivity
EC_p	Pore water Electrical Conductivity
TDR	Time Domain Reflectrometry
FDR	Frequency Domain Reflectrometry
NIR	Near Infrared Reflectrometry
TLO	Transmission Line Oscillation
ε	Real di-electric permittivity
I	Impedance
C	Capacitance
R	Resistance
T	Temperature
θ	Volumetric Water Content(VWC)
RH	Relative Humidity
AH	Absolute Humidity
LC	Sinusoidal Oscillator
BFSL	Best Fit Straight Line
SPAW	Soil and Water Characteristics Model
SDI-12	Asynchronous Serial Communications Protocol
PVC	Poly Vinyl Chloride – a polymer used in hose pipes
LED	Light Emitting Diode
InGaAsSb	Indium Gallium Arsenic Antimony Photodiode
WSN	Wireless Sensor Networks
GPS	Global Positioning System
IF	Intelligent Farming
SMASH	Smart Machines for Agricultural Solutions Hightech
MEMS	Micro Electro-Mechanical System
LAN	Local Area Network
PoE	Power over Ethernet
SoC	System-on-Chip
GPU	Graphical Processing Unit
RAM	Random Access Memory
GPIO	General Purpose Input Output

USB	Universal Serial Bus
CSI	Camera Serial Interface
Wi-Fi	Wireless Fidelity
DSI	Display Serial Interface
ARM	Advanced RISC Machines
PMIC	Power Management Integrated Circuit
GND	Ground
VREF	Voltage Reference
VDD	Voltage Drain Drain
VCC	Voltage Common Collector
PWM	Pulse Width Modulation
SPI	Serial Peripheral Interface
SCLK	Serial Clock
MOSI	Master OUT Slave IN
MISO	Master IN Slave OUT
CE	Chip Enable
SDA	Serial Data
SCL	Serial Clock
UART	Universal asynchronous receiver transmitter
TX	Transmitter
RX	Receiver
DC	Direct Current
AC	Alternating Current
CMOS	Complementary metal-oxide semiconductor
PCB	Printed Circuit Board
MCU	Multipoint Control Unit
PLC	Programmable Logic Controller
FAT32	File Allocation Table
VNC	Virtual Network Computing
SSH	Secure Shell Communication Protocol
I2C	Inter Integrated Circuit

Appendix

A.1 Configuring the Raspberry Pi

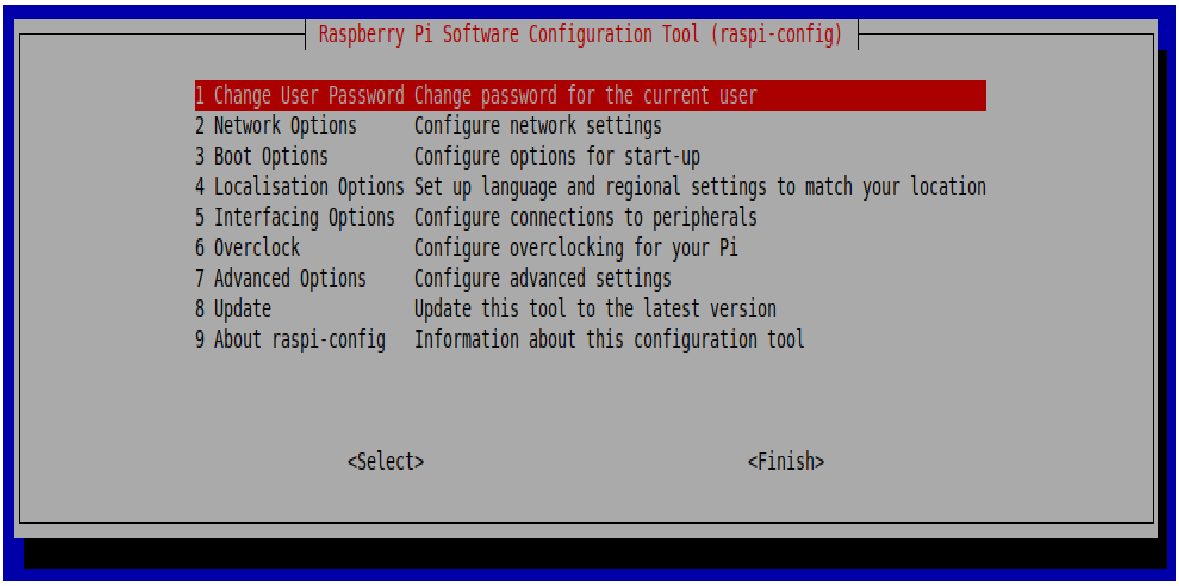
A micro SD card and a high quality 2.5A micro USB power supply is required to configure the Raspberry pi. The following steps are adopted:

1. The micro SD is formnatted in FAT32 mode using a compatible flashing software such as SD card formatter.
2. The Raspbian OS is downloaded from the offical website and it is the image is flashed on the SD card using a Software – Balena Etcher.
3. The micro USB cable is connected to the Pi's power supply port which acts both for powering up and data transfers.

The Raspberry pi displays the configuration tool [22] when it is booted for the first time after installing the Raspbian OS. It can also be accessed by entering the following command on the Raspberry Pi Linux Operating System.

```
sudo raspi-config
```

The term sudo is required to change the files that are not owned by the pi user.



```
              Raspberry Pi Software Configuration Tool (raspi-config)

1 Change User Password Change password for the current user
2 Network Options      Configure network settings
3 Boot Options         Configure options for start-up
4 Localisation Options Set up language and regional settings to match your location
5 Interfacing Options  Configure connections to peripherals
6 Overclock            Configure overclocking for your Pi
7 Advanced Options     Configure advanced settings
8 Update               Update this tool to the latest version
9 About raspi-config   Information about this configuration tool


        <Select>                              <Finish>
```

Figure A. 1 Raspberry Pi Software Configuration Tool

It has the following top level options available:

1. **System Options:** It configures system settings.
2. **Display Options:** It configures connections to peripherals.
3. **Performance Options:** It configures performance settings.
4. **Localisation Options:** It configures language and regional settings.
5. **Advanced Options:** It configures advanced settings.
6. **Updates:** It updates the tool to the latest version.
7. **About raspi-config:** It displays information about the configuration tool.

System Options

The System options allows the user to make changes to boot, login and networking processes.

Wireless LAN

This option allows the user to access wireless LAN SSID and passphrase.

Audio

The audio output destination can be changed by selecting this option.

Password

By default, Raspberry pi can be accessed with the following credentials-

Username - `pi`

Password – `raspberry`

The access credentials can be changed with this setting.

Hostname

This option sets the visible name for the Pi on a network.

Boot/Auto Login

It allows the user to boot to console or desktop. If auto login is selected, default Pi user is logged in.

Network at Boot

The option allows the user to wait for a network connection before letting the user to proceed to the boot screen.

Splash Screen

This option enables/disables the splash screen displayed at the boot time.

Power LED

This setting is compatible based on the pi model and allows the user to change the behavior of the power LED.

Display Options

This setting allows the user to change the following display options.

Resolution: The default HDMI/DVI video resolution is used when the system boots without a TV or a monitor is connected. This setting can effect the RealVNC, if the VNC option is enabled.

Underscan

This option allows the user to scale the display text when the screen disappears off the edge.

Pixel Doubling

This option enables/disables 2x2 pixel mapping.

Composite Video

This option is enabled by default on Raspberry Pi 3B+ and all other previous models.

Screen Blanking

This option enables/disables screen blanking.

Interfacting Options

The pi has a wide range of interfaces and this option allows the user to configure these interfaces manually.

Camera

This option enables/disables the CSI camera interface.

SSH

SSH allows the user ro remotely acess the command line of the pi from another computer. The **SSH** option is disabled by default.

VNC

This option enables/disables RealVNC virtual network computing server.

SPI

This option enables/disables SPI interfaces and automatic loading of the SPI kernel module for products such as PiFace.

I2C

This option enables/disables I2C interfaces and automatic loading of the I2C kernel module.

Serial

This option enables/disables shell and kernel messages on the serial connection.

1-wire

This option enables/disables the Dallas 1-wire interface. This option is usually used for DS18B20 temperature sensors.

Remote GPIO

This option enables/disables access to the GPIO pins.

Performance Options

The following list of performance options can be configured using this setting.

Overclock

It is possible to overclock Raspberry Pi’s CPU on few of the models using this tool. The overclocking may result in instability. This option can be enabled/disabled solely based on user’s discretion.

GPU Memory

This option allows the user to change the memory allocation available to the GPU.

Overlay File System

This option allows user to enable/disable read-only filesystem.

Fan

This option sets the behavior of the GPIO connected Fan.

Localisation Options

The Localisation options allows the user to configure the following settings.

Locale

This option allows the user to select a locale, for example `en_GB.UTF-8 UTF-8`

Time Zone

This option allows the user to select the local time zone.

Keyboard

This option allows the user to select the keyboard layout.

WLAN Country

This option sets the country code for the selected wireless network.

Advanced Options

The Advanced options are not suitable for a basic user. The following list of advanced option can be accessed and reconfigured.

Expand Filesystem

This option when selected begins partion expansion immediately.

GL Driver

This option enables/disables the experimental GL desktop graphic drivers.

GL(Full KMS)

This option enables/disables the experimental OpenGL Full KMS(Kernel mode setting) desktop graphics driver.

GL(Fake KMS)

This option enables/disables the experimental Fake KMS desktop graphics driver.

Legacy

This option enables/disables the original legacy non-GL VideoCore desktop graphics driver.

Compositor

This option enables/display the xcompmgr composition manager.

Network Interface Names

This option enables/disables predictable network interface names.

Network Proxy Settings

This option configures the network's proxy settings.

Boot Order

This option allows the user whether to boot from USB or network if the SD card isn't inserted.

Bootloader Version

This option allows the user to use the latest boot ROM software, or by default to the factory version if the the latest version causes problems.

Update

This option allows the user to update the Raspberry pi configuration tool to the latest version.

Using the terminal

The terminal is a really useful application: it allows you to navigate file directories and control your Raspberry Pi using typed commands instead of clicking on menu options.

A few of the commonly used terminal commands are listed as follows:

```
ls
```

The ls command lists the content of the current directory (or one that is specified).

```
cd
```

Using cd changes the current directory to the one specified.

```
pwd
```

The pwd command displays the name of the present working directory.

```
mkdir
```

This command is used to create a new directory.

```
rmdir
```

This command is used to remove empty directories.

```
rm
```

The command rm removes the specified file (or recursively from a directory when used with -r).

`cp`

Using cp makes a copy of a file and places it at the specified location (this is similar to copying and pasting).

`mv`

The mv command moves a file and places it at the specified location (so where cp performs a 'copy-paste', mv performs a 'cut-paste'). The usage is similar to cp.

`touch`

The command touch sets the last modified time-stamp of the specified files or creates it if it does not already exist.

`cat`

The command cat is used to list the contents of files

`head`

The head command displays the beginning of a file. It can be used with -n to specify the number of lines to show (by default ten), or with -c to specify the number of bytes.

`tail`

The opposite of head, tail displays the end of a file. The starting point in the file can be specified either through -b for 512 byte blocks, -c for bytes, or -n for number of lines.

`chmod`

The command chmod is used to change the permissions for a file.

`chown`

The chown command changes the user and/or group that owns a file.

`ssh`

ssh denotes the secure shell. It allows to connect to another computer using an encrypted network connection.

`scp`

The scp command copies a file from one computer to another using ssh.

`sudo`

The sudo command is used to run a command as a superuser, or another user. Example: sudo -s for a superuser shell.

`dd`

The dd command copies a file converting the file as specified. It is often used to copy an entire disk to a single file or back again.

`df`

The df command is used to display the disk space available and used on the mounted filesystems.

`unzip`

The unzip command extracts the files from a compressed zip file.

`tar`

The command tar is used to store or extract files from a tape archive file. It can also reduce the space required by compressing the file similar to a zip file.

`pipes`

A pipe command allows the output from one command to be used as the input for another command.

`tree`

The tree command is used to show a directory and all subdirectories and files indented as a tree structure.

`&`

Any command can run in the background with &, which allows freeing up the shell for future commands.

`wget`

This command allows to download a file from the web directly to the computer with wget.

`curl`

The curl command is used to download or upload a file to/from a server. By default, it will output the file contents of the file to the screen.

`man`

This command shows the manual page for a file.

Search commands

`grep`

The command grep is used to search inside files for certain search patterns.

`awk`

awk is a programming language useful for searching and manipulating text files.

`find`

The find command searches a directory and subdirectories for files matching certain patterns.

`whereis`

The whereis command is used to find the location of a command. It looks through standard program locations until it finds the requested command.

Networking commands

`ping`

The ping utility is usually used to check if communication can be made with another host.

`nmap`

nmap is a network exploration and scanning tool. It can return port and OS information about a host or a range of hosts.

`hostname`

The hostname command displays the current hostname of the system.

`ifconfig`

ifconfig command is used to display the network configuration details for the interfaces on the current system when run without any arguments (i.e. ifconfig).

A.2 Using the GNU Nano Editor for Software Scripting.

GNU Nano is a text editor for Unix-like computing systems or operating environments using a command-line interface. It is an easy-to-use editor that is versatile and simple having the following shortcuts and their respective functions.

File handling

Ctrl+S	Save current file
Ctrl+O	Offer to write file ("Save as")
Ctrl+R	Insert a file into current one
Ctrl+X	Close buffer, exit from nano

Search and replace.

Ctrl+Q	Start backward search
Ctrl+W	Start forward search
Alt+Q	Find next occurrence backward
Alt+W	Find next occurrence forward
Alt+R	Start a replacing session

Deletion

Ctrl+H	Delete character before cursor
Ctrl+D	Delete character under cursor
Alt+Bsp	Delete word to the left
Ctrl+Del	Delete word to the right
Alt+Del	Delete current line

Operations

Ctrl+T	Execute some command
Ctrl+J	Justify paragraph or region
Alt+J	Justify entire buffer
Alt+B	Run a syntax check
Alt+F	Run a formatter/fixer/arranger
Alt+:	Start/stop recording of macro
Alt+;	Replay macro

Moving around

Ctrl+B	One character backward
Ctrl+F	One character forward
Ctrl+←	One word backward
Ctrl+→	One word forward
Ctrl+A	To start of line
Ctrl+E	To end of line

Ctrl+P	One line up
Ctrl+N	One line down
Ctrl+↑	To previous block
Ctrl+↓	To next block
Ctrl+Y	One page up
Ctrl+V	One page down
Alt+\	To top of buffer
Alt+/	To end of buffer

Special movement

Alt+G	Go to specified line
Alt+]	Go to complementary bracket
Alt+↑	Scroll viewport up
Alt+↓	Scroll viewport down
Alt+<	Switch to preceding buffer
Alt+>	Switch to succeeding buffer

Information

Ctrl+C	Report cursor position
Alt+D	Report line/word/character count
Ctrl+G	Display help text

Various

Alt+A	Turn the mark on/off
Tab	Indent marked region
Shift+Tab	Unindent marked region
Alt+N	Turn line numbers on/off
Alt+P	Turn visible whitespace on/off
Alt+V	Enter next keystroke verbatim
Ctrl+L	Refresh the screen
Ctrl+Z	Suspend nano

A.3 Implementing the Software Architecture and the Automation Logics of the System using Python 3 and GNU Nano Editor.

The entire system is implemented using the Python 3 programming language and the GNU Nano editor. A few of the key functionalities of the system is discussed in this section along with their corresponding pseudo-codes.

#Importing modules

For the Python code to run desirably, all the key modules are identified and imported into the main code.

#To Import the Thingspeak application

```
import thingspeak
```

# To communicate with SPI devices

```
import spidev
```

# To scale the values using NumPy

```
from numpy import interp
```

# To add delay

```
from time import sleep
```

# To use GPIO pins

```
import RPi.GPIO as GPIO
```

#To use the Adafruit DHT Library

```
import Adafruit_DHT
```

#To use Date and Time modules

```
import time
import datetime
```

# Channel ID of the Thingspeak Cloud

```
channel_id =  1293615
```

# Input the Write Key of Thingspeak Cloud

```
write_key  = 'RAVJ0HIM3D5KB28K'
```

#Input the Read Key of Thingspeak Cloud

```
read_key   = '9EZ7E0918UVVAGAY'
```

#DHT22 AM2302 Sensor Identification

```
sensor = Adafruit_DHT.AM2302
```

#To retrieve the digital data of the DHT22 sensor connected to GPIO4

```
sensor_pin = 4
```

#To create a variable to control the while loop

```
running = True
temperaturef = 0
```

#To create DHT22 sensor data logging in the storage memory of the Pi
#A.txt file created with header

```
file = open('sensor_readings.txt', 'w')
file.write('time and date, temperature (C),temperature (F),
humidity\n')
```

#To start an SPI connection
#To create an object

```
spi = spidev.SpiDev()
spi.open(0,0)
```

#To initialize the Relay Inputs for switching action using the Raspberry Pi
#IN1 → GPIO5
#IN2 →GPIO6
#IN3 →GPIO13
#IN4 →GPIO19

```
relay1_pin = 5
relay2_pin = 6
relay3_pin = 13
relay4_pin = 19

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(relay1_pin, GPIO.OUT)
GPIO.setup(relay2_pin, GPIO.OUT)
GPIO.setup(relay3_pin, GPIO.OUT)
GPIO.setup(relay4_pin, GPIO.OUT)
```

#Relay 1 off state

```
GPIO.output(relay1_pin, GPIO.HIGH)
```

#Relay 2 off state

```
GPIO.output(relay2_pin, GPIO.HIGH)
```

#Relay 3 off state

```
GPIO.output(relay3_pin, GPIO.HIGH)
```

#Relay 4 off state

```
GPIO.output(relay4_pin, GPIO.HIGH)
```

#Threshold Values of Moisture Sensors
#Below the Threshold values, Actuator DC Pump is switched on for Irrigation.
#threshold_water == Capacitive Soil Moisture Sensor Threshold
#threshold_soil ==Resistive Soil Mositure Sensor Threshold

```
threshold_water = 30
threshold_soil = 30
```

# To read the digital data from the MCP3008 Analog to Digital Converter IC

```
def analogInput(channel):
spi.max_speed_hz = 1350000
adc = spi.xfer2([1,(8+channel)<<4,0])
data = ((adc[1]&3) << 8) + adc[2]
return data
while True:
date = datetime.datetime.now().strftime('%m-%d-%Y_%H.%M.%S')
print (date)
date2 =int( datetime.datetime.now().strftime('%H'))
print(date2)
output = analogInput(0) # Reading from CH0
output = abs( int( interp(output, [0, 1023], [100, 0]) -16))
print("reading from analog channel0  capacitive moisture
sensor :"+str(output))
output2 = analogInput(1) # Reading from CH1
output2 =int( interp(output2, [0, 1023], [100, 0]))
print("reading from analog channel1  resistive moisture  sensor
:"+str(output2)+"\n")
sleep(2)
```

#To read the temperature and humidity digital values from the DHT22 Sensor

```
humidity,   temperature   =   Adafruit_DHT.read_retry(sensor,
sensor_pin)
```

# To convert to temperature reading into Fahrenheit

```
try:
temperature_f =( temperature * 9/5) + 32
print("readings gotten")
except TypeError:
print ('temp error')
```

#To send the temperature and humidity reading to the Thingspeak Cloud

```
channel          =          thingspeak.Channel(id=channel_id,
write_key=write_key, api_key=read_key)
response = channel.update({'field1': temperature, 'field2':
humidity,'field3':output,'field4':output2})
```

#To get valid readings incase of reading error or null result

```
if humidity is not None and temperature is not None:
```

#To print temperature and humidity digital values as output readings

```
print('Temperature = ' + str(temperature) +','+ 'Temperature
Fahrenheit = ' + str(temperature_f) +',' + 'Humidity = ' +
str(humidity)+'\n')
```

#To save time, date, temperature in Celsius, temperature in Fahrenheit and humidity in .txt file for data logging in the storage memory of the Raspberry Pi

```
file.write(time.strftime('%H:%M:%S  %d/%m/%Y')  +  ',  '  +
str(temperature)   +   ',   '+   str(temperature_f)+','   +
str(humidity) + '\n')
time.sleep(10)
else:
print('Failed to get reading. Try again!')
time.sleep(1)
```

#To check if the mosture reading are below threshold limits

```
if output<threshold_water  and output2<threshold_water:
print ('PUMP ON')
```

# DC pump turns on for irrigation

```
GPIO.output(relay1_pin, GPIO.LOW)

else:
print('PUMP OFF')
```

#DC pump turn off after irrigation

```
GPIO.output(relay1_pin, GPIO.HIGH)
```

#To control the LED grow lights based on threshold time limits of the Day

```
 if date2>8 and date2< 16:
print('LED LIGHTS ON')
```

#LED Lights Turn on based on time of the Day

```
GPIO.output(relay2_pin,GPIO.LOW)

else:

print('LED LIGHTS OFF')
```

#LED Lights Turn off based on time of the Day

```
GPIO.output(relay2_pin,GPIO.HIGH)
```

The entire software script is saved in the main directory of the Raspberry Pi as combined.py It can be executed from the terminal command center by typing the following command.

```
python combined.py
```

Any changes can be made to the code by entering the following command from the terminal command center.

```
nano combined.py
```

Bibliography

[1] Gubbi, Jayavardhana, et al. "Internet of Things (IoT): A vision, architectural elements, and future directions." *Future generation computer systems* 29.7 (2013): 1645-1660.

[2] Gondchawar, Nikesh, and R. S. Kawitkar. "IoT based smart agriculture." *International Journal of advanced research in Computer and Communication Engineering* 5.6 (2016): 838-842.

[3] S. A. Nagaonkar | Dr. S. D. Bhoite | 30129 "Design and Development of IoT and Cloud Based Smart Farming System for Optimum Water Utilization for Better Yield" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470

[4] Balamurugan, C., and R. Satheesh. "Development of raspberry pi and IoT based monitoring and controlling devices for agriculture." *Journal of Social, Technological and Environmental Science* 6.2 (2017): 207-15.

[5] Suakanto, Sinung, et al. "Sensor networks data acquisition and task management for decision support of smart farming." *2016 International Conference on Information Technology Systems and Innovation (ICITSI)*. IEEE, 2016.

[6] S. A. Shete, "IoT based Smart Water Management with Sensors Agriculture Stick for Live Temperature and Moisture Monitoring", International Journal of Science and Research(IJSR), Volume 8 Issue 3, March 2019, 633 – 634

[7] Patil, Bhavana, et al. "Plant Monitoring using image processing, raspberry Pi & Iot." *International Research Journal of Engineering and Technology (IRJET)* 4.10 (2017).

[8] Ryu, Minwoo, et al. "Design and implementation of a connected farm for smart farming system." *2015 IEEE SENSORS*. IEEE, 2015.

[9] Punjabi, Hariharr C., et al. "Smart farming using IoT." International Journal of Electronics and Communication Engineering and Technology 8.1 (2017): 58-66.

[10] Thakare, Anuradha, et al. "DECISION SUPPORT SYSTEM FOR SMART FARMING WITH HYDROPONIC STYLE." *International Journal of Advanced Research in Computer Science* 9.1 (2018).

[11] Gupta, Arushi, Debangsi Satapathy, and Sanjeev Thakur. "Technology based Streamlined Agro-Farming Techniques." *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. IEEE, 2020.

[12] Aggarwal, Sachin, and Anil Kumar. "A smart irrigation system to automate irrigation process using IOT and artificial neural network." *2019 2nd International Conference on Signal Processing and Communication (ICSPC)*. IEEE, 2019.

[13] Jaiswal, Himanshu, Ram Singuluri, and S. Abraham Sampson. "IoT and Machine Learning based approach for Fully Automated Greenhouse." *2019 IEEE Bombay Section Signature Conference (IBSSC)*. IEEE, 2019.

[14] Negi, Dhira, et al. "Smart Harvest Analysis Using Raspberry Pi Based on Internet of Things." *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*. IEEE, 2018.

[15] Putjaika, Narayut, et al. "A control system in an intelligent farming by using arduino technology." *2016 Fifth ICT International Student Project Conference (ICT-ISPC)*. IEEE, 2016.

[16] Kumar, Nenavath Ravi, and E. Nagabhooshanam. "An extended Kalman filter for low-cost positioning system in agricultural vehicles." *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 2016.

[17]https://www.electronicsforu.com/tech-zone/electronics-components/humidity-sensor-basic-usage-parameter

[18]https://www.nrcs.usda.gov/wps/portal/nrcs/detailfull/national/?&cid=stelprdb1045331

[19] Yin, Zhe, et al. "A near-infrared reflectance sensor for soil surface moisture measurement." *Computers and electronics in agriculture* 99 (2013): 101-107.

[20] Sadeghi, Ali, et al. "A plant-inspired robot with soft differential bending capabilities." *Bioinspiration & biomimetics* 12.1 (2016): 015001.

[21] https://bsr.iit.it/funding/smash

[22] https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/

[23] https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf

[24]https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0193_Web.pdf

[25]https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0114_Web.pdf

[26]https://www.amazon.it/infinitoo-Coltivazione-Luminosit%C3%A0-Fioritura-Fruttificazione/dp/B08LDFFXHN

[27]https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf

[28]https://media.digikey.com/pdf/Data%20Sheets/Seeed%20Technology/108990019_Web.pdf

[29]https://www.amazon.in/Syska-Power-Port100-10000mAH-Lithium-Ion/dp/B07JHWDJ73

[30] http://www.farnell.com/datasheets/2310930.pdf

[31] https://www.handsontec.com/dataspecs/4Ch-relay.pdf

[32] https://www.raspberrypi.org/documentation/configuration/

[33] https://easyeda.com/

[34] https://www.realvnc.com/en/connect/download/vnc/raspberrypi/

[35] https://www.teamviewer.com/en/solutions/remote-desktop/raspberry-pi/

[36] https://hackaday.io/project/173288-how-to-interface-mcp3008-with-raspberry-pi

[37] https://pimylifeup.com/raspberry-pi-humidity-sensor-dht22/

[38] https://github.com/iobridge/thingspeak

[39] https://www.pidramble.com/wiki/benchmarks/power-consumption

[40] Research and Markets"Global Indoor Farming Technology Market by Growing System (Hydroponics, Aeroponics, Aquaponics, Soil-based, Hybrid), Facility Type, Component, Crop Type (Fruits & Vegetables, Herbs & Microgreens, Flowers & Ornamentals), and Region - Forecast to 2026"

[41] https://s.campbellsci.com/documents/es/manuals/cs650.pdf

[42]http://publications.metergroup.com/Manuals/20587_TEROS11-12_Manual_Web.pdf

[43] http://library.metergroup.com/Manuals/20429_GS3_Web.pdf

[44] http://publications.metergroup.com/Manuals/20424_5TM_Manual_Web.pdf

[45] http://publications.metergroup.com/Manuals/20435_5TE_Manual_Web.pdf

[46] https://www.metergroup.com/environment/products/mas-1/

[47] http://publications.metergroup.com/Manuals/20426_10HS_Manual_Web.pdf

[48] https://www.metergroup.com/environment/products/ec-5-soil-moisture-sensor/

[49]https://delta-t.co.uk/wp-content/uploads/2017/01/SM150T-user-manual-version-1.0.pdf

[50] https://delta-t.co.uk/wp-content/uploads/2017/02/ML3-user-manual-version-2.1.pdf

[51] https://delta-t.co.uk/product/wet-2-horticulture/

[52] https://www.stevenswater.com/resources/documentation/hydraprobe/HydraProbe_Manual_Jan_2018.pdf

[53] https://www.imko.de/en/trime-pico-32/

[54] https://acclima.com/wp-content/uploads/TDR-User-Manual-2.pdf