AN IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE

by

ANGITHA K A

(2020-18-004)

DEPARTMENT OF IRRIGATION AND DRAINAGE ENGINEERING
KELAPPAJI COLLEGE OF AGRICULTURAL ENGINEERING AND TECHNOLOGY
TAVANUR - 679573, MALAPPURAM
KERALA, INDIA
2023

AN IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE

by

ANGITHA K A

(2020-18-004)

THESIS

Submitted in partial fulfilment of the requirement for the award of degree of

MASTER OF TECHNOLOGY

IN

AGRICULTURAL ENGINEERING

(Soil and Water Conservation Engineering)

Faculty of Agricultural Engineering & Technology

Kerala Agricultural University

DEPARTMENT OF IRRIGATION AND DRAINAGE ENGINEERING
KELAPPAJI COLLEGE OF AGRICULTURAL ENGINEERING AND TECHNOLOGY
TAVANUR - 679573, MALAPPURAM
KERALA, INDIA
2023

DECLARATION

I, hereby declare that this thesis entitled "**AN IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE**" is a bonafide record of research work done by me during the course of research and the thesis has not previously formed the basis for the award to me of any degree, diploma, associateship, fellowship or other similar title, of any other University or Society.

Angitha

Place: Tavanur

Date: 07/10/23

Angitha K A

2020-18-004

CERTIFICATE

Certified that this thesis entitled "**AN IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE**" is a record of research work done independently by **Ms. Angitha K A** under my guidance and supervision and that it has not previously formed the basis for the award of any degree, diploma, fellowship or associateship to her.

Place: Tavanur

Date: 07/10/23

Dr. Asha Joseph
(Major Advisor, Advisory Committee)
Professor
Department of IDE
KCAET Tavanur

CERTIFICATE

We, the undersigned members of the advisory committee of Ms. Angitha K A (2020-18-004), a candidate for the degree of Master of Technology in Agricultural Engineering with major in Soil and Water Conservation Engineering, agree that the thesis entitled "**AN IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE**" may be submitted by **Ms. Angitha K A** (2020-18-004), in partial fulfilment of the requirement for the degree.

Dr. Asha Joseph
(Major Advisor, Advisory Committee)
Professor
Department of IDE
KCAET Tavanur

Rema KP
Dr. Rema K P
(Member, Advisory Committee)
Professor and Head
Department of IDE
KCAET Tavanur

Dr. Anu Varughese
(Member, Advisory Committee)
Assistant Professor
Department of IDE
KCAET Tavanur

Dr. Josephina Paul
(Member, Advisory Committee)
Assistant Professor and Head
Department of BEAS
KCAET Tavanur

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

First of all, I offer million gratitude to God, the almighty who made me to do this task and made every job a success for me. He was the greatest source of all resources and provision, moral or without whose grace nothing is possible.

My diction would be inadequate to express my deepest sense of gratitude and heartfelt thanks to Chairman of my Advisory Committee, ***Dr. Asha Joseph****., Professor, Department of Irrigation and Drainage Engineering, Kelappaji College of Agricultural Engineering and Technology, Tavanur. Her level of guidance, constructive criticism and generous assistance at every stage of my research work is beyond measure, in fact it was the new idea and smooth dealing with the thing which motivated me to work under her guidance. It is my proud privilege to express my heartfelt indebtedness and deepest sense of gratitude for laying out the guidelines of research work. I have real admiration and regards for her whole hearted support and untiring help.*

It is my pleasure to pay tribute to ***Dr. Jayan P R.,*** *Dean of Faculty and Professor & Head, Department of Farm Machinery and Power Engineering, KCAET, Tavanur, for his valuable advices and help rendered during this study.*

I avail this opportunity to express my sincere thanks to my advisory committee members ***Dr. Rema K P****., Professor and Head, Department of Irrigation and Drainage Engineering, KCAET Tavanur,* ***Dr. Anu Varughese.,*** *Assistant Professor, Department of Irrigation and Drainage Engineering, KCAET, Tavanur and* ***Smt. Josephina Paul****., Assistant Professor (BEAS), KCAET, Tavanur for their valuable counsel, note- worthy guidance, suggestions and cordial co-operation during my research programme.*

I express my sincere and well devoted thanks to ***Dr. Abdul Jabbar P K****., Professor and Head, Instructional Farm, KCAET, Tavanur for his immense help, valuable suggestions, advices and deemed support during the polyhouse experiment to complete the research successfully.*

I feel happy to express my sincere thanks to ***Mr. Harishankar K P****., Farm Officer,* ***Ms. Shamniya K****., Farm Officer and* ***Ms. Arpana Sunil****., Farm Officer, Instructional Farm, KCAET Tavanur for their constant help, valuable suggestions, advice and guidance throughout my research work. I also acknowledge the help and support by* ***Dr. Dipak Suresh Khatawkar.***

I also feel happy to express my heartfelt thanks to my nearest and dearest friends ***Er. Jyothish Chandran, Er. Rabeea Assainar K K, Er. Nagil, Er. Anjitha Jacob, Er. Shahala M, Er. Akhina P, Er. Ranjisha K and Er. Varun Raveendranathan*** *for their boundless help and care in every step of my studies and research work.*

I express my thanks to all the ***faculty members of Library****, KCAET. Tavanur for their ever willing help and cooperation. I express my sincere thanks and gratitude to* ***Kelappaji College of Agricultural Engineering & Technology*** *for giving me an opportunity to undergo my PG studies and* ***Kerala Agricultural University*** *for all financial assistance, which offered me a chance to study in this institution.*

Words are inadequate to express my heartfelt gratitude to my beloved Father ***Mr. Anilkumar K K and*** *late mother* ***Mrs. Sheeba C S*** *for their everlasting love, affection, help, support, incessant motivation and constant encouragement throughout my life.*

I am greatly indebted to my beloved brother ***Mr. Akshay K A*** *for his valuable advice, encouragement and support during the course of study.*

One last word, since it is practically impossible to list all contributions to my work, it seems proper to issue a blanket of thanks for those who helped me directly or indirectly during the course of my study.

ANGITHA K A

(2020-18-004)

Dedicated

to

My loving Family,
Teachers and Agricultural
Engineers

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF PLATES

SYMBOLS AND ABBREVIATIONS

%	:	Percentage
&	:	And
‘	:	Minute
“	:	Second
<	:	Less than
=	:	Equal to
>	:	Greater than
°	:	Degree
°C	:	Degree Celsius
A	:	Ampere
ADC	:	Analogue to Digital Converter
AJAX	:	Asynchronous JavaScript and XML
API	:	Application Programming Interface
CO_2	:	Carbon dioxide
CoAP	:	Constrained Application Protocol
DAP	:	Days After Planting
DC	:	Direct Current
Dept.	:	Department
EC	:	Electrical Conductivity

FPGA	:	Field Programmable Gate Array
GB	:	GigaByte
GHz	:	Gigahertz
GPRS	:	General Packet Radio Service
GSM	:	Global System for Mobile Communication
GUI	:	Graphical User Interface
GWT	:	Google Web Toolkit
Hz	:	Hertz
IDE	:	Integrated Development Environment
IFTT	:	If This Then That
IMU	:	Inertial Measurement Unit
IoT	:	Internet of Things
KAU	:	Kerala Agricultural University
KCAET	:	Kelappaji College of Agricultural Engineering and Technology
kg	:	Kilogram
kg/cm^2	:	Kilogram per square centimeter
l	:	Litre
L/m^2	:	Litre per square metre
LCD	:	Liquid Crystal Display
LDR	:	Light Dependent Resistor

LED	:	Light-Emitting Diode
lh^{-1}	:	Litre per hour
lx	:	Lux
m^2	:	Square metre
mA	:	Milliampere
MAS	:	Multi Agent System
Mbps	:	Megabits per second
NTC	:	Negative Temperature Coefficient
OTA	:	Over-The-Air
PAR	:	Photosynthetically Active Radiation
PC	:	Personal Computer
PCB	:	Printed Circuit Board
pH	:	Potential of Hydrogen
PLD	:	Programmable Logic Devices
POP	:	Package of Practices
ppm	:	Parts per million
PVC	:	Poly Vinyl Chloride
RH	:	Relative Humidity
SD	:	Secure Digital
SMS	:	Short Message Service

SSR : Solid State Relay

t/ha : Tonnes per hectare

TNAU : Tamil Nadu Agricultural University

UART : Universal Asynchronous Receiver-Transmitter

UIDs : Unique Identifiers

V : Volt

VPD : Vapour Pressure Deficit

Wh/m^2 : Watt-hour per square meter

Wi-Fi : Wireless Fidelity

WSN : Wireless Sensor Network

YVM : Yellow Vein Mosaic

INTRODUCTION

CHAPTER-I

INTRODUCTION

The world population has grown to 8 billion and is predicted to increase to 9.8 billion by 2050 (FAO, 2020). The world will face a significant challenge related to ensuring the availability of food for 9.8 billion humans. Urbanisation and infrastructure development are also expanding rapidly which causes a decrease in agricultural land (conversion to housing and industry), hunger, poverty and increasing demand for healthy and fresh foods.

Monitoring of agricultural crops is accomplished primarily through the expensive, labour-intensive, and time-consuming process of crop scouting, by manual sampling and documenting the state of the field. The most important factors that determine the quality and productivity of plant growth are temperature, Relative Humidity (RH), and light intensity. Continuous monitoring of these environmental variables provides valuable information to the grower to better understand how each of these factors affects growth and how to maximize crop productiveness (Abd El-kader and El-Basioni, 2003). Smart farming technologies, such as IoT based sensors and data analytics, play a crucial role in enabling precision agriculture by acquiring and analyzing data from the field (Kagan *et al.,* 2022). A greenhouse is a structure or building specifically designed for the cultivation and protection of plants, providing an enclosed environment that can be regulated to optimize plant growth. It is typically constructed using transparent materials, such as glass or plastic, to allow sunlight to enter while trapping heat inside. The primary purpose of a greenhouse is to create a controlled microclimate that protects plants from adverse weather conditions, such as extreme temperatures, frost, wind, and pests. By manipulating various environmental factors, including temperature, relative humidity, light intensity and air circulation, greenhouse operators can create ideal growing conditions for a wide range of crops throughout the year (FAO, 2020).

Greenhouses provide a controlled environment for cultivating crops, but maintaining optimal microclimatic conditions within the structure is crucial for maximizing the yield and quality of produce (Maraveas *et al*., 2022). Here comes the importance of greenhouse automation. Greenhouse automation systems have emerged as a game-changer in the agricultural industry, offering farmers advanced tools to enhance crop productivity and resource efficiency. The greenhouse automation system is designed to streamline and optimize critical processes such as temperature control, irrigation, ventilation, and lighting. By integrating smart technology, sensors, and actuators, farmers can achieve precise control over environmental conditions and resource utilization.

In recent years, the agricultural industry has witnessed a significant transformation due to the advent of Internet of Things (IoT) technology. IoT is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with Unique Identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction (Gillis, 2021). In IoT, all the components are connected to the internet and can transmit or receive information, or both. IoT enables devices embedded with sensors to connect to and interact with each other via the internet. It is a cloud of interconnected physical devices such as microcontrollers, microprocessors, actuators and sensors, which can communicate with each other over the internet. The measured data can be observed from anywhere around the world. Devices can be remotely monitored and controlled in real-time. It also reduces the manpower requirement and the risk involved in visiting inaccessible places.

The convergence of IoT technology and greenhouse automation has transformed traditional farming practices, offering new opportunities for sustainable and efficient agriculture. The IoT-based greenhouse automation system combines sensor networks, actuators, data analytics, and cloud computing to enable precise control, optimization, and monitoring of greenhouse conditions (Dhanaraju *et al*., 2022).

An IoT-based real-time microclimate monitoring and controlling system designed specifically for greenhouse environments harness the power of interconnected sensors, actuators and data analytics to create an intelligent infrastructure that continuously collects and analyzes data related to temperature, relative humidity, light intensity and other crucial parameters. Additionally, the IoT-based system incorporates actuation mechanisms that allow farmers to remotely control various environmental factors within the greenhouse. Through a user-friendly interface, farmers can adjust parameters such as temperature, humidity, and irrigation systems based on real-time data and recommendations provided by the system. This automation minimizes human error and optimizes resource utilization, leading to increased crop productivity and reduced operational costs (Akkas and Sokullu, 2017).

The implementation of IoT based microclimate monitoring and controlling system has the potential to revolutionize greenhouse farming practices. By providing real-time insights and precise control over environmental conditions, farmers can create ideal growth conditions for their crops, resulting in higher yields and improved sustainability. By leveraging the power of interconnected sensors, data analytics, and actuation mechanisms, the system empowers greenhouse farmers with valuable insights and precise control over environmental conditions. The integration of IoT technology in greenhouse farming aligns with the broader trends of precision agriculture and sustainable food production. Thus, this innovative approach holds immense potential to improve crop yields, enhance sustainability and revolutionise the future of greenhouse farming.

In view of all the above facts the present study entitled **"An IoT based Real-time Microclimate Monitoring and Controlling System for Greenhouse"** was undertaken with the following specific objectives:

- To develop a web enabled microcontroller embedded system with sensors and IoT technology for greenhouse
- To monitor and control the various microclimate parameters in real-time
- To evaluate the system for vegetable cultivation

REVIEW OF LITERATURE

CHAPTER-II

REVIEW OF LITERATURE

The integration of Internet of Things (IoT) technology in greenhouse agriculture holds tremendous potential for advancing sustainable and efficient crop cultivation. By providing real-time insights and autonomous control, an IoT-based microclimate monitoring and controlling system can empower growers to make informed decisions, mitigate risks, and maximize yield. Hence, in this chapter, a review of the literature on various studies related to the microclimate variabilities in greenhouses, automation in greenhouses, and greenhouse automation using IoT are briefly explained.

2.1 VARIABILITY OF MICROCLIMATE IN GREENHOUSES

Ganesan (1999) conducted experiments at the M. S. Swaminathan research foundation's ecohorticulture farm at the Tamil Nadu livestock research station in Kattupakkam. He found that the greenhouse had lower Relative Humidity (RH) at 8 am and had higher daytime temperatures than the open field. Greenhouses had less light intensity than open fields. In comparison to open fields, the yield performance inside the greenhouse was highest (2145 g tomatoes/plant).

Adams *et al*. (2001) examined the effects of temperature on tomato fruit ripening and growth under controlled environmental conditions. Fruits ripened at different times (95, 65, 46 and 42 days after flower opening) when grown at temperatures of 14°C, 18°C, 22°C, and 26°C respectively. A thermal time model was developed based on these data, but it proved less effective when buds/fruits were heated at different developmental stages. Later stages of fruit maturation were more sensitive to elevated temperatures. Fruit growth rates in volume were adequately described using a Gompertz function, with low temperatures reducing absolute volume growth rates and delaying maximal growth. Manipulation of fruit temperature resulted in different growth responses. Both high (26°C) and low (14°C) temperature regimes led to small parthenocarpic fruits, along with low number of flowers and poor fruit set at 26°C, resulting in low fruit yields.

Kittas *et al.* (2001) investigated the temperature and humidity distribution in a commercial greenhouse used for cut flower production. The greenhouse was equipped with a cooling pad system and a partially shaded plastic canopy. The cooling pad system proved to be effective in achieving 80% cooling efficiency and maintained a temperature of 10°C lower than the ambient temperature. However, they observed that while the humidity was maintained, the temperature increased from the pads to the center of the greenhouse in the unshaded area. The researchers found that the cooling pad method worked well in dry climates but resulted in reduced plant transpiration in humid climates during the morning hours. Based on their findings, they suggested that in dry climates, it was advisable to avoid shading the greenhouse roof since evaporative cooling alone can effectively prevent overheating inside the greenhouse.

Kavitha *et al.* (2003) studied the morphological traits, physiological responses, and growth and yield characteristics of tomato and brinjal in a 24 m^2 polyhouse with a UV-stabilized polyethylene cover. An exhaust fan and a spinning disc sprayer powered by solar energy were used. By creating particular climatic conditions in the polyhouse, the crop response was changed. For tomatoes, within the polyhouse, shoot length increased by 96% and yield increased by 27% compared to the control. For brinjal, the yield rose by 85% and the shoot length increased by 55%.

Al-Helal (2007) investigated the influence of two ventilation rates on the environment of a shaded greenhouse equipped with a fan-pad evaporative cooler during typical summer days under extreme conditions. The greenhouse, planted with cucumbers, had two exhaust fans for providing either one air exchange per minute or 0.5 air exchange per minute. The results showed that the greenhouse with a ventilation rate of 0.5 air exchange per minute had average daytime air temperature and RH of 33.6°C and 33.5%, respectively, compared to 30.2°C and 37.5%, respectively, with a ventilation rate of one air exchange per minute. Temperature gradients were observed along the airflow direction, with the highest temperature differences occurring between locations that were located 5 and 38 meters away from the cooling pad. The water and electrical energy consumption were found 8.4 L/m^2 and 48.2 Wh/m^2 for 0.5 air exchange per minute and 14 L/m^2 and 99.6 Wh/m^2 for one air exchange per minute respectively.

Parvej *et al*. (2010) conducted an experiment in greenhouse and open field for the purpose of comparing the phenological development and production potentials of two tomato varieties. They noticed that though the soil and air temperatures were greater inside the greenhouse than the open field, the Photosynthetically Active Radiation (PAR) was 40% lower. The tomato plants grown in greenhouses exhibited an advanced flowering (3 days), fruit setting (4 days) and fruit maturity (5 days) than the open field. The yield from the greenhouse was found 81 t/ha, whereas the yield from the open field was 57 t/ha.

Umesha *et al*. (2011) conducted an experiment at the TNAU campus in Coimbatore, India, during 2009-2010 to investigate the impact of polyhouse conditions on the growth characteristics and fruit yield of tomatoes. Daily measurements of temperature, RH, and light intensity were taken at different stages of crop growth. The findings revealed that the temperature inside the polyhouse was high during the afternoon and reached 39.88°C. The RH was higher in the morning (91.06%) and lower in the afternoon (38.48%) within the polyhouse. Light intensity was highest during the afternoon (58865 lx), while low levels of light intensity were observed during the morning and evening hours. Hence, it was concluded that the changes in microclimate associated with the polyhouse positively influenced the growth and yield of tomato.

Jamaludin *et al*. (2014) conducted an experiment in a 300 m^2 tropical greenhouse with fan and pad cooling system in order to create an appropriate microclimate inside the greenhouse. The greenhouse's vertical and horizontal temperature and RH profiles were examined. The findings demonstrated that both the horizontal and vertical planes experienced a rise in temperature and a decrease in RH. It was also discovered that a greenhouse with a fan and pad cooling system was appropriate.

Gokul (2015) examined how the cowpea crop performed in rain shelters and greenhouses with natural ventilation. There was an increase of 2.7°C to 3.4°C in air temperature inside the polyhouse compared to the open field while it was 1.4°C to 2°C in the rain shelter compared to the open field.

Li *et al.* (2017) conducted field tests to test diurnal variations of temperature, RH and solar radiation and analysed the microclimate characteristics in naturally ventilated single-sloped greenhouses of different sizes. The results showed that indoor temperature and humidity under natural ventilation varied from 28.5°C to 25.8°C and 96% to 84% respectively which illustrated that ventilated greenhouses were able to create a favourable self-maintained, energy-balanced environment for vegetable growth. The relationship between the incident solar radiation and indoor air temperature was also determined. This study provided a reference for further research to reduce energy consumption and achieve a favourable greenhouse microclimate, leading to higher product quality, improved yield and shorter cultivation time in single-sloped greenhouses.

Shamshiri (2017) developed a growth response model to measure the optimal degree of air temperature and RH in a tropical greenhouse planted with tomatoes in Malaysia. A custom-designed hardware/software framework was created to interface with the model and process the collected data. The framework provided valuable insights for greenhouse climate control and management by determining the optimal conditions for specific hours, light conditions and growth stages, which in turn can help minimize cooling requirements.

Gazquez *et al.* (2018) examined the effects of several cooling techniques, including white washing, fogging, natural ventilation, and forced ventilation, on the microclimate, growth, and yield of a crop grown in a greenhouse. According to their findings, fogging was the most effective approach for regulating temperature and Vapour Pressure Deficit (VPD) values, but it was the least effective for regulating canopy temperature. According to economic analysis, bleaching was the most successful cooling procedure. Hence, it was concluded that the most effective cooling system for the Mediterranean environment is a mix of plastic cover bleaching and natural ventilation.

2.2 AUTOMATIC MICROCLIMATE MONITORING AND CONTROLLING SYSTEM IN GREENHOUSES

Ahonen *et al*. (2010) used Wireless Sensor Network (WSN) to create a monitoring and control system for greenhouses. The system made use of sensors (for temperature, RH and light intensity) and microcontroller. The system gathered data regarding greenhouse factors. The controller activated the actuators whenever a parameter reached the predetermined threshold level. The created system was tested in a greenhouse with tomato crop and worked successfully.

Bhujbal *et al*. (2012) created a microcontroller-based automation system for greenhouses. A single chip microcomputer monitored temperature, humidity, EC, and pH in a greenhouse. The collected data was saved on the MicroSD data module. The system sent an alert message and all parameter information to the greenhouse manager through a Global System for Mobile Communication (GSM) module. The greenhouse manager had the capability to update and modify the greenhouse parameters as and when needed.

Booij *et al*. (2012) created an automated climate control system for layer-by-layer microclimate management in greenhouses. The system was created for multilayer cropping inside the greenhouse. At each layer, microclimate data was measured and controlled based on predetermined values.

Dondapati and Rajulu (2012) created a sensor-based automation system for greenhouses that was capable of controlling the greenhouse without the involvement of any humans. The system was made up with various data-collection sensors, Analogue to Digital Converter (ADC), a microprocessor and actuators. It operated foggers, fans, irrigation systems, and lighting systems to adjust the greenhouse's microclimate based on real-time data gathered from various sensors. The specialists working in greenhouses benefited from the real-time microclimatic data displayed on Liquid Crystal Displays (LCD) in order to know the precise environmental conditions there.

Khandelwal (2012) designed a GSM modem-based automation system to regulate the microclimate in greenhouses. The system used sensors to gather data on temperature, RH, light intensity, rain sensors, transistor switches, and relay nodes for automation control. The data was kept on a data server. The automation system maintained the environmental conditions necessary for agricultural growth based on the crop's needs.

Kolhe and Annadate (2012) created a greenhouse automation system using an ARM7 controller. LPC2148 was the system's primary microcontroller. Light Dependent Resistor (LDR) light sensor, SYHS- 220 RH sensor and LM35 temperature sensor were the various sensors employed. The system board was based on ARM. The greenhouse atmosphere was managed and maintained by the system.

Vidyasagar (2012) created a system using WSN and GSM, to automatically monitor and adjust greenhouse climate. Sensing data was gathered and shared using WSN. The automation system was equipped with sensors that recorded data on soil moisture and temperature. Exhaust fans and miniature sprinklers were employed as two actuators. The PIC16F877A microcontroller was used. The predetermined threshold level controlled the automation system that turned ON fan and micro sprinkler. The registered mobile number received the information.

Waykole and Agrawal (2012) proposed a microcontroller-based greenhouse automation system. The system used WSN to gather greenhouse microclimate data and used Zigbee to transmit it. The microprocessor received the recorded data and, in accordance with the predetermined threshold level, activated the actuators to control the greenhouse.

Dhumal and Chitode (2013) proposed an automated greenhouse system that employed Zigbee technology and a smartphone. Real-time data was gathered by WSN and sent over Zigbee to the server. TEAM VIEWR synchronisation software was employed to keep the server and mobile device in sync. To communicate and regulate the greenhouse's settings, software was created in visual basic. The user received an SMS with information on the greenhouse settings. Internet connectivity exists between the user's server and Android phone. The user can control the automation system using the interface on an android phone.

Gayatri (2013) suggested a Psoc 3 kit-based automation system for greenhouses. The technology made use of a programmable system on chip called Psoc3. Actuators, relays, Psoc 3, temperature sensors, and RH sensors made up the proposed system. The Lm35DZ temperature sensor and the SY-HS-20 RH sensor were used. Using a transducer, physical amount was transformed into electrical quantity. Automation system controlled the greenhouse environment based on the programme and the predetermined threshold levels of parameters.

Salleh *et al*. (2013) developed a WSN and Zigbee technology-based greenhouse automation system. The designed automation system was put to test in a greenhouse. The 9V DC power supply, WSN, Zigbee transmitter, microcontroller and actuators were the components of the automation system. the collected data were shown on an LCD panel,. ‘C’ compiler and ‘MPLAB’ tools were utilised for the programming and interface-building procedure. The microprocessor controlled the actuators to control the greenhouse microclimate based on the predetermined levels.

Asholkar and Bhadade (2014) developed a hardware implementation module which was capable of measuring temperature, humidity, soil moisture, light intensity and CO_2 concentration of crops in green house. It also presented the analysis and prediction of greenhouse parameters. Crops selected for analysis and prediction were namely cucumber, tomato, brinjal, papaya and chillies. The wireless media of communication used was GSM technology to assist the control and monitoring of process plants. Five sample crops were taken and the system had tested for these crops in green house environment. Finally estimation of total power consumption and total expense consumed per annum were done. It was also seen that quality and quantity of produce with controlling action was increased more than that of the crops grown without controlling action.

Belsare *et al*. (2014) developed a greenhouse automation system that can be controlled manually or automatically and could be used with an android phone. If the actuators were to be operated automatically, the data received from the sensors was sent to a server, which send signals based on the threshold values that have been defined. If it was in a manual mode, a message was sent to the user and they could control it using their Android phone.

Dinesh and Saravanan (2014) created and implemented an automated greenhouse monitoring system. Programmable Logic Devices (PLDs) were utilised because they enabled the building of automation systems employing Field Programmable Gate Array (FPGA). An Analogue to Digital Converter (ADC), FPGA, and temperature and RH sensors were the components of the created automation system. On the LCD screen, greenhouse parameters (temperature and RH) measured were displayed. The system performed satisfactorily during the test. The developed controller activated the actuators to control the greenhouse climate whenever the greenhouse parameters went above the threshold level.

Eldhose *et al*. (2014) created an automated greenhouse monitoring system using the microcontroller PIC 16F877A. Based on the established criteria and the measured climatic factors, microcontroller regulated the greenhouse atmosphere. Thus, the automated technology reduced the labour costs associated with maintaining a greenhouse's atmosphere. It controlled the actuators and automatically gathered data on soil and environmental conditions. It had the ability to control the greenhouse for better crop production by analysing data on temperature, RH, soil temperature, and light intensity.

Martinovic and Simon (2014) designed a mobile measurement station for controlling greenhouse microclimate. For obtaining and keeping track of climatic characteristics both within and outside the greenhouse, they deployed WSN. They equipped a robot with sensors, and the robot was navigated using WSN. Based on the data gathered, they also created an expert system to control the numerous environmental control devices connected to the greenhouse. The expert system used fuzzy rules to make decisions based on multiple criteria. Based on sensed values of various factors, they created several control techniques for maintaining the greenhouse microclimate. Based on the input parameters, the most effective control technique was selected.

Parvez *et al*. (2014) developed and tested a greenhouse automation system for controlling greenhouse climate. The system successfully managed the greenhouse's microclimate. The sensor station, coordination station, and central station were the major three stations in the automation system. The automation system used wireless sensors. Both the central station and the coordination station, as well as the connection between the sensor station and the coordination station used Zigbee wireless modules.

Poyen *et al*. (2014) created an automated system to control the greenhouse's shade cover. The automation system controlled a motor that raised or lowered the shade cover based on the intensity of the light. The automation system made use of an Arduino microcontroller, a logic circuit, a light sensor, and a motor-driven shade cover. The automation system was functioned based on pre-set values.

Thenmozhi *et al*. (2014) created a remote and automated management system for greenhouses. For efficient greenhouse climate control, Zigbee and embedded system technology were used. A message of warning was transmitted to the user. Through the microcontroller and server, the user could control the actuators remotely. In the automation system, a PIC 16F877A microcontroller was employed. Based on the predetermined threshold level, the microcontroller itself controlled the greenhouse in full automatic mode. The user received a message and the server stored the data.

Asolkar and Bhadade (2015) proposed an effective method for crop monitoring in greenhouse using GSM wireless technology. The proposed greenhouse system provided impact on varieties of crop species mostly flowers, vegetable crops and fruit crops. The presented system effectively monitored and controlled the important greenhouse parameters like temperature, RH, soil moisture, light intensity and CO_2 gas. The system was tested in greenhouse environment and observations were recorded for crop analysis purpose. The crop analysis helped farmers for monthly future prediction to know the expenditure for growing crop and made effective solution for farmers to grow highly efficient and disease free crop species for commercial production.

Bajer and Krejcar (2015) created a low-cost remote-control automated greenhouse system. The system's automation was managed using an Arduino MEGA 2560 board. System components included sensors, controllers, and actuators. An LCD display allowed users to see the parameters. The designed system functioned satisfactorily.

Joteppagol and Kore (2015) created a greenhouse automation system that communicated over Controller Area Network (CAN). The system used the LM-35 temperature sensor and the SY-HS-220 RH sensor. PIC18F458 was the microcontroller utilised, and the actuators were exhaust fans for cooling and LEDs for artificial lighting. The microprocessor controlled the actuators in accordance with the predetermined conditions.

Mekki *et al*. (2015) implemented a WSN by deployed wireless sensor nodes in a greenhouse with temperature, humidity, moisture light and CO_2 sensors. The proposed model was built and tested, and the result showed an excellent improvement in the sensed parameters. To control the environmental factors, they used microcontroller programmed to control the parameters according to preset values, or manually through a user interface panel.

Atia and El-madany (2016) designed and compared four different types of greenhouse temperature control systems by using MATLAB/SIMULINK simulations. Artificial neural networks, fuzzy logic and adaptative neurofuzzy control were the approaches employed to control the greenhouse's temperature.

Testing revealed that the ‘adaptative neural fuzzy control’ method was the most effective for regulating greenhouse temperature.

In order to monitor the greenhouse environment in greenhouses, Liu *et al*. (2016) used a monitoring system. The monitoring system included a sensor-monitoring network for gathering environmental data and a specific sink node to gather data and transmit it back to the remote management centre via a communication network.

Canadas *et al*. (2017) designed a greenhouse temperature management system that could reduce the occurrence of disease by maintaining proper climate within the greenhouse. They added a real-time greenhouse management and monitoring system. In the event that a sickness was discovered, the decision support system was capable of providing instructions for changing the microclimate. The technology was able to reduce sickness as a result of the greenhouse's unfavourable climate in this way.

Jinu *et al*. (2018) conducted a comparative study on performance of automation system in controlling greenhouse microclimate. They found that the temperature inside the manually controlled greenhouse increased to 43.1°C whereas the temperature increased only to 37.6°C in the automated greenhouse. In comparison to a manually controlled greenhouse, the automated greenhouse had superior temperature control, which led to a higher production of salad cucumbers (7.54 kg/plant).

2.3 AUTOMATIC MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE USING INTERNET OF THINGS (IoT)

Sonawane *et al*. (2008) implemented an internet-based system to control and monitor environmental parameters in a polyhouse farm, ensuring optimal conditions for crop growth. They developed a concept that involved collecting thermal process data through sensors, storing and processing the data, and transmitting it online to multiple users via web browsers. The system allowed real-time control of various polyhouse parameters, such as pumps, accessories, louvers, and airflow rate, from remote monitoring stations. They also designed a user-friendly graphical interface for farmers, enabling them to operate the system easily. The system supported the transmission of process parameters and emergency signals via email or SMS, and it incorporated a chat feature to facilitate communication between users.

Rangan and Vigneswaran (2010) developed an embedded systems technique for greenhouse monitoring. In order to measure variables including humidity, water pH, soil moisture, light intensity, and temperature, they put sensors in various places. A PIC microcontroller processed and managed the measured data. The system made use of a mobile receiver, a GSM modem, and an LCD module to display the parameters and send SMS updates to the owner. The sensor outputs were continuously monitored by the microprocessor, which compared them to pre-set values. The microcontroller saved the information and alerted the farmer through SMS if any parameter varied from the preset settings. For measuring temperature, pH value, humidity, soil wetness, displaying parameters on the LCD module, serial communication, and delivering signals to the mobile receiver, an assembly code was created. They made improved hardware implementation for automatic monitoring and control of the greenhouse by applying IoT technology.

AI-Aubidy *et al*. (2014) proposed and implemented a real-time monitoring and control of a number of environmental parameters for a set of greenhouses. A wireless sensor network was imagined to have each greenhouse as a node. To monitor and manage a number of factors and maintain the required condition in each greenhouse, a single-board microcontroller-based system was developed and put into service. For each greenhouse, a rule-based fuzzy controller was created to regulate the microclimate. The proposed technology gave the farmer the ability to keep an eye on the greenhouse's internal environment. In addition, the farmer can instruct specific greenhouse devices to be turned ON or OFF via wireless communications. Both simulated and actual results were compared to illustrate system performance and real-time remote monitoring and control operations and was found satisfactory.

Qiu *et al*. (2014) designed an intelligent greenhouse environment monitoring and control system to meet the requirements in real-time, reliability and sustainability for crop-growth environment monitoring based on ZigBee and embedded technologies. The temperature, humidity and light intensity were monitored in real-time and sent through wireless network. This help to achieve intelligent control for the greenhouse fans, lights and irrigation equipment.

Halim *et al*. (2016) developed an automated scheduler system for mango by considering all optimal plant growth requirements for each phase of the plant to ensure that mango will grow perfectly. They selected Memsic and Zigbee technology to achieve greenhouse management and monitoring. The system allowed monitoring the condition of irrigation, soil moisture, humidity, temperature, and radiation of light via sensor and send the data to the microcontroller. The data were then sent from transmitter to receiver and display to device (smartphone/laptop/PDA). MPLAB and LabVIEW software were used for the programming and interfacing the process.

Mohanraj *et al*. (2016) conducted a study on field monitoring and automation using IoT in agriculture domain. Monitoring modules were demonstrated using various sensors. A prototype of the mechanism was carried out using TI CC3200 launchpad interconnected sensor modules with other necessary electronic devices. A comparative study was made between the developed IoT system and the existing systems. The system overcame limitations of traditional agricultural procedures.

Muangprathub *et al*. (2016) developed an IoT and agriculture data analysis system for smart farm. They designed and developed a control system using node sensors in the crop field with data management via smartphone and web application. The components used were hardware, web application and mobile application for data collection, manipulation and controlling. The system sent notifications through LINE API for LINE application. The performance of the system was found good.

Akkas and Sokullu (2017) developed an IoT-based greenhouse monitoring system with Micaz motes. The designed system allowed precise position real-time measurements and data transmission from the greenhouse to the interested farmer. An original routing protocol was designed and incorporated in an IoT based internet access. Data were stored in a computer and day/time- based reports and graphics were generated.

Tanappagol and Kondikopp (2017) designed and implemented a system that can monitor and control the temperature, humidity, soil moisture in the greenhouse field area. Effective management of greenhouse environment by both automatic manner and human involvement manner were done. Automatic controlling process was fully done based on coding. The sensors were connected to the Arduino UNO microcontroller. Bluetooth sensor network was used to maintain network performance at a high level. The sensors sent all the available parameters accurately.

Achouak *et al*. (2018) proposed a new design for the greenhouse using Constrained Application Protocol (CoAP) for monitoring and controlling greenhouse parameters. The monitored parameters were temperature, humidity and pH and the system included sensors, actuators, microcontrollers and an IoT.

Bharathi *et al*. (2018) developed a polyhouse automation system using RaspberryPi 3B model. The system monitored temperature, humidity and moisture values from the crop and disease infestation using a camera via microcontroller and IoT. The disease monitoring was done by image processing technique. The system monitored the parameters successfully.

Chang *et al*. (2018) proposed a prototype of smart lighting system based on narrowband-IoT communication. for connecting the back-end system and fields such as outdoor greenhouses or street lights. In the testbed, the dimming of light was scheduled and controlled automatically. The system also monitored temperature and humidity of the field .The system verified usabilities and advantages applying narrowband-IoT to the smart lighting system.

Devekar *et al*. (2018) system monitored the temperature, humidity, soil moisture and intensity of light in the polyhouse using various sensors. The system also focused on detecting the suspicious object and notified to farmer based on template matching algorithm to process the images and detected the suspicious objects in polyhouse. The whole system was controlled and monitored by IoT and mobile app.

Raja and Rajarathinam (2018) developed a wireless sensor network to gather temperature and humidity data from point to point. A Graphical User Interface (GUI) was unified for the ease of operations by the farming community. System also allowed transmission of process parameters, including emergency alarm signals via email client server or alternatively by sending SMS on a mobile phone. A conventional chat also had been integrated with the GUI to add vibrancy to inter-user communication. Such integrated approach greatly widened the socio-economic possibilities for farmers through interaction with modern technological resources.

Unsal *et al*. (2018) presented a wireless greenhouse monitoring system based on IoT technology using Arduino compatible NodeMCU and Wemos Wi-Fi modules. The proposed system collected the light, temperature, humidity, CO_2, and soil moisture values through sensors and transferred data to remote server over the internet. The data sent from the greenhouse system was accepted by a web service on the server side and was stored in the database using IoT. Thus, the data obtained through the greenhouse system could be monitored successfully through a web server.

Wang *et al*. (2018) created a greenhouse environment monitoring and control management system based on Google Web Toolkit (GWT). The system achieved functions like configuring acquisition and control parameters, adaptive database matching between gateway and server, adaptive diagnosis of monitoring parameters, adaptive warning of monitoring parameters, adaptive generation of interface, and more by using remote method call (RPC) AJAX as the communication method between browser and web server. The system's capabilities were evaluated, and the findings demonstrated that the Android app and web browser application could accomplish greenhouse environment monitoring and control in an adaptive manner depending on the information configuration.

Zafar *et al*. (2018) developed an IoT based real-time environmental monitoring system using Arduino and cloud service. They developed an environmental monitoring system which employed sensors for temperature and humidity of the surrounding area. The system was embedded with Arduino UNO board, DHT11 sensor, ESP8266 Wi-Fi module and the data were transmitted to Thingspeak. The data were used to trigger short-term actions such as remotely controlling the heating and cooling devices. The sensed data were uploaded to cloud storage and was accessed through android application.

Zou *et al*. (2018) designed a monitoring system for greenhouse with Zigbee network, embedded controller and 3G network. The design of sensor circuit mainly employed digital module sensors. Sensor circuit was controlled by relay switch. The gateway node was designed by employing high performance 32-digit embedded controller and WinCE6.0 embedded OS. Embedded SQlite database was used on WinCE6.0 for effective data management. The closed loop control was realized by employing fuzzy algorithm and the test result showed that the deviation of atmospheric temperature, atmospheric humidity, illumination intensity, CO_2 concentration and soil water content was controlled within ±0.6°C, ±1.2%, ±276 lx, ±23 ppm and ±0.9% respectively.

Devanath *et al*. (2019) designed and implemented an IoT based greenhouse environment monitoring and controlling system using Arduino platform. They developed an Arduino based checking and controlling framework with basic sensors *viz.* DHT11 sensor, Soil Moisture sensor, LDR sensor and pH sensor. A mobile application was used for observing and controlling and NodeMCU esp8266 was used to send the information.

Kitpo *et al.* (2019) developed a platform called IoT tomato that can detect temperature conditions, humidity, soil moisture and light for greenhouses planted with tomatoes. The acquired data were then transferred wirelessly using the ZigBee module to a server and stored in the cloud. Information acquired was sent to the user's smartphone via the LINE bot service.

Pandey and Chauhan (2019) proposed an automated IoT based smart polyhouse system with Hadoop technologies to bring precision in farming techniques. In order to increase the yield and quality of crops, the proposed nonmanual system helped to automate the functionality of polyhouse. The system had sensors, IoT devices and Hadoop technologies like Apache Hadoop, Flume and Hive. Using these technologies, data from various sensors were analysed to help in decision making.

Vishwakarma *et al*. (2020) developed an efficient and cost-effective system for monitoring and controlling greenhouse climate using the ESP8266 Node MCU module. The system optimized parameters soil moisture, light intensity, temperature, and humidity to meet crop production requirements. They employed soil moisture, LDR, and DHT22 sensors to collect data, which was then processed by the NodeMCU module. The system regulated water pumps, motors, exhaust, and lights based on the data calculations. The collected data was transmitted to farmers' smartphones through the IoT platform using the HTTP protocol. The system helped in achieving enhanced plant growth and increased agricultural productivity.

Wang *et al*. (2020) developed a data communication mechanism for greenhouse environment monitoring and control. A data encapsulation approach based on XML was developed to enable data inter-operability in a distributed greenhouse IoT system and to accomplish adaptive matching of data transfer between the gateway and server. In order to ensure data synchronisation, diverse data and replies were combined using the Multi-Agent System (MAS) behaviour. JADE was used to implement real-time and cumulative data synchronisation between the gateway and server. Testing carried out in a particular greenhouse revealed data loss rates of 1.52% between the data acquisition unit and gateway and 0.4% between the gateway and server, proving the viability of the data transmission method in a greenhouse IoT system.

Hemalatha *et al*. (2021) proposed an IoT-based disease-monitoring prototype for an agricultural/polyhouse application. The prototype was designed and tested to identify the disease onset in cucumbers. The work initially focused on recognizing the critical cucumber diseases in polyhouse using NodeMCU and Raspberry-Pi-based hardware model. The decisions to be made and the major changes in the sensed parameters if any will be intimated to the farmers using a specifically designed mobile application.

Kumar *et al*. (2020) developed an IoT enabled system to monitor and control greenhouse. The developed system was a combination of power electronics and power system. The system used Global System for Mobile communication (GSM), Arduino, and various sensors DHT11 sensor, Soil moisture sensor and LDR sensor. Using sensor, the system obtained the real time data of the crops and then transmitted it to the user mobile. Thus, the user could monitor and control the temperature, watering and light through mobile successfully.

MATERIALS AND METHODS

CHAPTER-III

MATERIALS AND METHODS

The materials used and methodology adopted for the development of an IoT based real-time microclimate monitoring and controlling system for greenhouse are explained in the following subheads.

3.1 DETAILS OF STUDY AREA AND WEATHER AND CLIMATE

3.1.1 Details of the Study Area

The site is situated on the cross point of 10° 51’13” N latitude and 75 °59’ 06” E longitude at an altitude of 25 m above mean sea level. Greenhouse selected for the study was a naturally ventilated polyhouse. The field experiment was conducted in a naturally ventilated polyhouse in the research plot of the Department of Irrigation and Drainage Engineering, in KCAET campus, Tavanur, Kerala. Details of the polyhouse are shown in Table 3.1.

Table 3.1 Details of the polyhouse

Area	213 m^2
Length	26 m
Width	8 m
Centre height	6 m
Gutter height	3.5 m

3.1.2 Weather and Climate

Agro - climatically, the area falls within the borderline of northern zone and central zone of Kerala. Most part of the rainfall received in this region is from southwest monsoon. The average annual rainfall varies from 2500 mm to 2900 mm. The average maximum temperature of the study area is 35ºC and the average minimum temperature is 26ºC. Average relative humidity and average light intensity are 73% and 30,000 lx respectively.

3.2 TIME FRAME OF THE EXPERIMENT

The overall experiment period was from June 2022 to May 2023. Performance evaluation of the developed automation system was carried out from

December 2022 to May 2023, first by checking the performance of the system without crop inside the greenhouse and thereafter with crop bhindi grown inside the greenhouse. Performance of the automation system without crop was evaluated from 09-01-2023 to 15-01-2023. Evaluation with crop growing inside the greenhouse was done from 20-01-2023 to 21-05-2023.

3.3 PREPARATION OF POLYHOUSE FOR EXPERIMENT

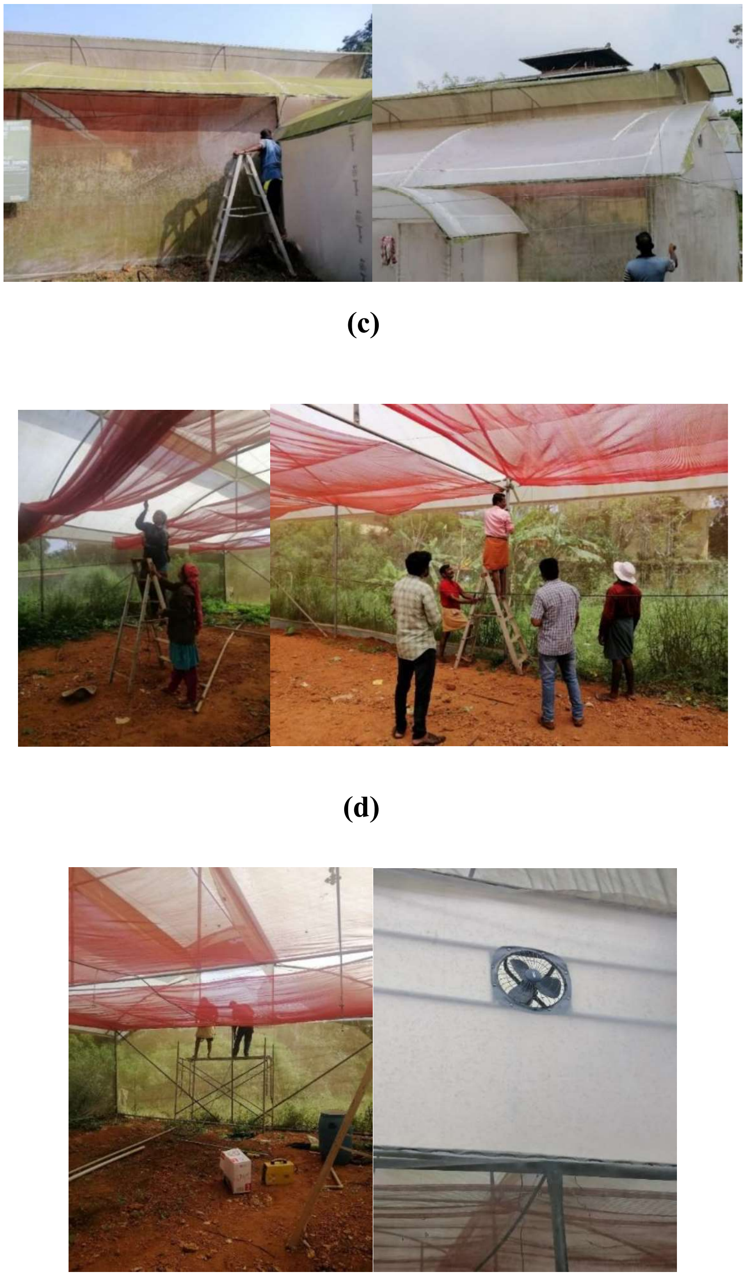
The following preparations were carried out in the polyhouse before the experiment and are shown in Plate 3.1.

1) Solar power system for the polyhouse was made ready to give power supply
2) An electric connection also established in case of solar system failure
3) Cleared the polyhouse and its premises by cutting grasses and weeds
4) UV cladding sheets of polyhouse were washed and cleaned
5) Two exhaust fans were installed inside the polyhouse
6) Shadenet and fogger were installed

An overall view of the polyhouse made ready for the experiment is shown in Plate 3.2.

(a)

(b)



(c)

(d)

(e)

Plate 3.1 Preparations for the experiment (a) Establishing electric connection (b) Cutting of grasses and weeds (c) Washing of UV cladding materials (d) Fixing shadenet inside the polyhouse (e) Fixing exhaust fans

Plate 3.2 A view of the polyhouse made ready for experiment

3.4 DEVELOPMENT OF A WEB ENABLED MICROCONTROLLER EMBEDDED SYSTEM WITH SENSORS AND IoT TECHNOLOGY FOR GREENHOUSE

This section includes the description and concept of IoT, and the steps involved in the design and installation of IoT based real-time microclimate monitoring and controlling system for greenhouse.

3.4.1 Description and Concept of Internet of Things (IoT)

IoT is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with Unique Identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction (Gillis, 2021). It is a cloud of interconnected physical devices such as microcontrollers, microprocessors, actuators, and sensors, which can communicate with each other over the internet. The measured data can be observed from anywhere around the world. Devices can be remotely monitored and controlled in real time. It also reduces the manpower requirement and the risk involved in visiting inaccessible places. IoT based greenhouse automation system consist of various sensors for collecting information about greenhouse parameters, a control device for receiving information and operating various actuators based on preset values through relays and various actuators for greenhouse environment control (Mohanty and Patil, 2013). Various sensors include sensors for air temperature, relative humidity and light intensity *etc.*

Depending upon the parameters to be controlled, the type of microcontroller can be selected. The actuators are the devices for the control of microclimate parameters such as foggers, exhaust fans, artificial lighting system *etc.* (Riahi *et al*., 2020).

3.4.2 Steps Involved in the Design and Installation of IoT based Real-time Microclimate Monitoring and Controlling System for Greenhouse

The various steps involved are as follows:

1) Selection of components
2) Calibration of sensors
3) Hardware designing
4) Software designing

3.4.3 Hardware and Software Components Selected for the Development of Iot System

IoT hardware refers to the physical components and devices that enable data collection, processing, and communication in the IoT ecosystem. They not only collect data but also respond to instructions based on the processed data. The set of programs that help to get the activities done like the data collection, processing, storage, and executing instructions based on the processed data form the IoT software. Operating systems, firmware, applications, middleware are some of the examples that fall into this category. Hardware and software components used for the study are explained in the following subheads.

3.4.3.1 Hardware components

- Microcontroller: Arduino Nano 33 IoT
- Sensors
  1) DHT22 temperature and humidity sensor
  2) BH1750 light sensor
- Liquid Crystal Display (LCD)
- Solid State Relay (SSR)
- Wi-Fi modem
- GSM module
- Hi-Link power module
- PCB dotboard

- Solar (Power Supply)
- Extension board
- Energymeter
- Actuators
    1) Fogger
    2) Exhaust fans

3.4.3.2 Software requirements

- Arduino IoT Cloud
- Arduino IDE (Integrated Development Environment)
- Arduino IoT cloud remote phone application

3.4.4 Details of Hardware Components Used

Various hardware components used in this study with its description and specifications are explained under the following subheads.

3.4.4.1 Arduino Nano 33 IoT

The Arduino Nano 33 IoT is the easiest and cheapest point of entry to enhance existing devices. It is a microcontroller with an inbuilt Wi-Fi module. The board's main processor is a low-power Arm® Cortex®-M0 32-bit SAMD21. The Wi-Fi and Bluetooth® connectivity are performed with a module from u-blox, the NINA-W10, a low-power chipset operating in the 2.4GHz range. On top of those, secure communication is ensured through the Microchip® ECC608 crypto chip. Besides that, a 6-axis IMU makes the board perfect for simple vibration alarm systems, pedometers, relative positioning of robots, *etc.* A labelled view of Arduino Nano 33 IoT is shown in Fig. 3.1 and its specifications are shown in Table 3.2.

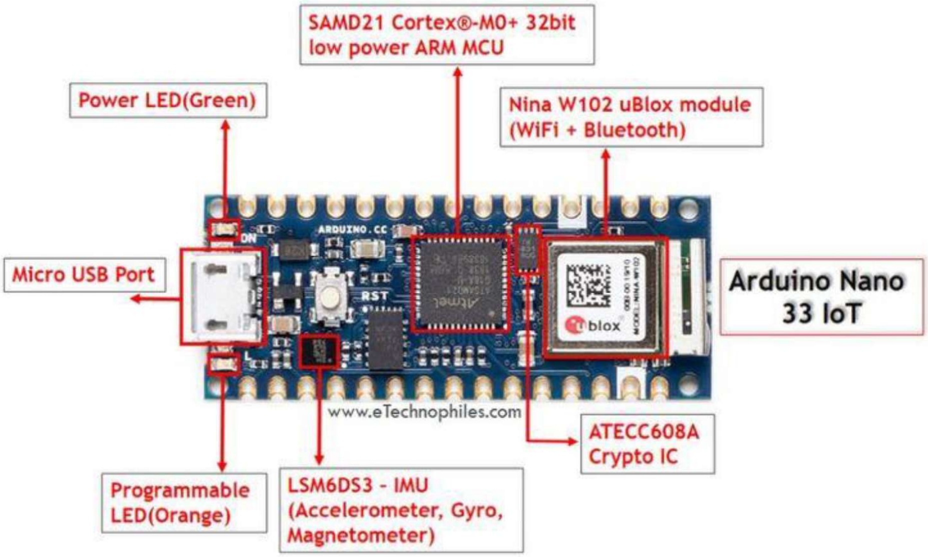


Fig. 3.1 Arduino Nano 33 IoT

Table 3.2 Specifications of microcontroller

Processor	SAMD21 Cortex®-M0+ 32bit low power ARM MCU
Operating Voltage	3.3 V
Input Voltage (limit)	21 V
DC Current per I/O Pin	7 Ma
Digital Input Pins	14
Analog Pins	8
LED_BUILTIN	13
Cost	2500/-

3.4.4.2 Sensors

Sensors are devices or instruments that detect and measure physical properties or changes in the environment and convert them into electrical signals or other readable formats. For this study, sensors were used to measure temperature, RH and light intensity inside the polyhouse.

a) DHT22 Temperature & Humidity Sensor

The DHT22 temperature & humidity sensor was used in the study to measure temperature and RH (Plate 3.2). The sensor uses a thermistor and a capacitive humidity sensor to measure the surrounding air temperature and RH respectively. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. For measuring temperature, it uses a Negative Temperature Coefficient (NTC) thermistor, which measure the resistance value. Specifications of the sensor are shown in Table 3.3.

Plate 3.2 DHT22 temperature & humidity sensor

Table 3.3 Specifications of DHT22 temperature & humidity sensor

Temperature range	-40 to 80°C
Relative Humidity range	0 to 100%
Resolution	Relative Humidity : 0.1% Temperature: 0.1°C
Operating voltage	3-6 V DC
Current supply	1-1.5 Ma
Sampling period	2 Seconds

3.4.4.3 BH1750 Light Sensor

BH1750 light sensor was used in the study to measure light intensity. It is a small and low-cost sensor that uses a photodiode. It absorbs energy from light and changes into electricity with the help of the photoelectric effect. The electricity produced is proportional to the intensity of light that falls on the sensor and sensor material. The BH1750 sensor is commonly used in applications that require automatic adjustment of the light level, such as in streetlights, backlight control for LCD displays, and indoor lighting systems. The BH1750 sensor has a built-in 16-bit analog-to-digital converter (ADC), which provides high accuracy and resolution. It also has a wide operating voltage range, typically from 2.4V to 3.6V, and consumes very low power, making it ideal for battery-powered applications. Overall, the BH1750 sensor is a reliable and accurate option for measuring light intensity and is widely used in various applications due to its low cost, high accuracy, and ease of use. A view of the BH1750 light sensor and its specifications are shown in Plate 3.3 and Table 3.4 respectively.

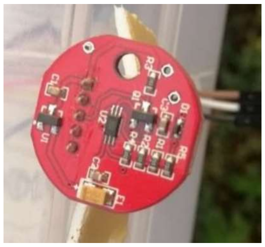


Plate 3.3 BH1750 Light sensor

Table 3.4 Specifications of BH1750 light sensor

Outer diameter	26 mm
Large edge diameter	28.5 mm
Height	26 mm (Plus light ball)
Working Voltage	DC 5V
Communication interface	I2C
Light range	0-65535 lx
Cost	Rs 130/-

3.4.4.4 Global System for Mobile Communication (GSM)

A GSM module is a device that enables mobile communication using the GSM network. It consists of a GSM modem, which is used to communicate with the network, and a microcontroller, which controls the operation of the module. SIM900A GSMGPRS was used for the study (Plate 3.4). GSM modules can be used to add mobile communication capabilities to a wide range of devices, such as alarm systems, vending machines, and remote monitoring systems. GSM modules typically support voice and Short Message Service (SMS) communication, as well as data communication over General Packet Radio Service (GPRS) or 3G/4G networks. They use a standard set of AT commands to control their operation, and can be interfaced with a wide range of microcontrollers, such as Arduino or Raspberry-Pi, Universal Asynchronous Receiver-Transmitter (UART) serial communication. Specifications of GSM are shown in Table 3.5.

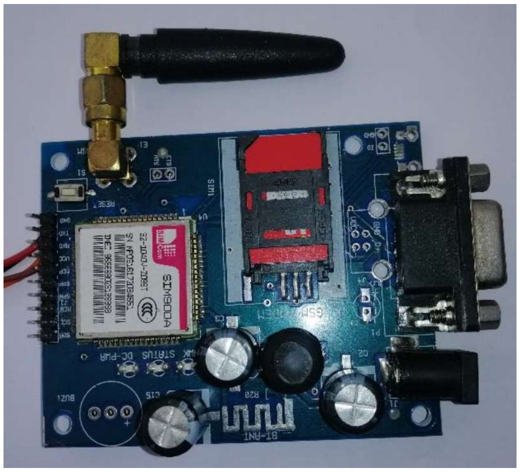


Plate 3.4 SIM900A GSMGPRS

Table 3.5 Specifications of GSM

PCB size	71.4mm X 66.0mm X1.6mm
Power supply	5V
Operation temperature	-40°C to +85°C
Cost	Rs 650/-

3.4.4.5 Hi-Link Power Module

The Hi-Link power module is a device that provides a compact and efficient solution for converting AC or DC power to a stable output voltage for various electronic applications. Hi-Link power module was used in the study to give power supply to microcontroller. Plastic enclosed PCB is mounted on the isolated switching power supply module. The series is a custom-designed power modules with high efficiency and small volume. There are many advantage of these modules, such as global input voltage range, low temperature rise, low power, high efficiency, high reliability, high security isolation, *etc.* It has been widely used in smart home, automation and control, communications equipment, instrumentation. A view of Hi-Link power module and its specifications are shown in Fig. 3.2 and Table 3.6 respectively.

Fig. 3.2 Hi-Link power module

Table 3.6 Specifications of Hi-Link power module

Model Number	HLK 5M05
Phase	Single Phase
Frequency	50Hz
Brand	HILINK
Color	Black
Input Signal	Single phase AC supply
Input Voltage	100 to 240V AC
Output Voltage	5VDC
Cost	500

3.4.4.6 LCD Display

Grove – 16 x 2 LCD is a perfect I2C LCD display for Arduino and Raspberry Pi with high contrast and easy deployment. Seedstudio Grove 16x2 LCD (White on Blue) was used for the study (Fig. 3.3). 16×2 means two lines and each line has 16 columns, 32 characters in total. With the help of Grove I2C connector, only two signal pins and two power pins are needed. Seedstudio Grove 16x2 LCD displayed values of temperature, Relative Humidity (RH) and light intensity. Specifications of Seedstudio Grove 16x2 LCD is shown in Table 3.7.

(a) **(b)**

Fig. 3.3 Seedstudio Grove 16x2 LCD (White on Blue)

Table 3.7 Specifications of Seedstudio Grove 16x2 LCD

Operating Voltage	3.3V / 5V
Operating temperature	0 to 50°C
Storage temperature	-10 to 60°C
Interface	I2C
I2C Address	0X3E
Cost	700/-

3.4.4.7 Wi-Fi Modem

Jio Wi-Fi-modem was used in the study to provide internet connection to Arduino Nano 33 IoT. Jio Wi-Fi modem is a device provided by Jio, a popular telecom operator in India, to provide high-speed internet connectivity to its customers. The Jio Wi-Fi modem, also known as JioFi, is a portable hotspot device that allows users to connect multiple devices to the internet wirelessly. It uses the 4G network to provide high-speed internet connectivity, and can also be used to make voice and video calls. To use the JioFi device, users need to insert the Jio sim card into the device and turn it on. They can then connect their devices to the Wi-Fi network created by the JioFi device using the Wi-Fi password provided with the device. Jio Wi-Fi modem and its specifications are shown in Plate 3.5 and Table 3.8 respectively.

Plate 3.5 Jio Wi-Fi modem

Table 3.8 Specifications of Jio Wi-Fi modem

Model Name	JMR541 Wi-Fi 4G Hotspot (Jio network)
Device Throughput	Up to 150 Mbps
Voice Support	No Call Support
Expandable Memory Capacity	32 GB
Cost	1500/-

3.4.4.8 Solid State Relay (SSR)

A SSR allows to control high-current AC loads from lower voltage DC control circuitry. It is an electronic switching device that switches ‘ON’ or ‘OFF’ when an external voltage (AC or DC) is applied across its control terminals. SSR-100DA Solid State Relay Module 3-32VDC/24-380VAC was used for the study (Fig. 3.4) and was used to actuate exhaust fans and foggers. Specifications of the relay are shown in Table 3.9.SSRs have several advantages over mechanical relays. One such advantage is that they can be switched by a much lower voltage and at a much lower current than most mechanical relays. Also, because there are no moving contacts, solid state relays can be switched much faster and for much longer periods without wearing out. SSR used in the study switched current loads of up to 40A with a 3-32V DC input and a zero cross trigger control method.

Features :

- Long service life and high reliability
- Reduced electromagnetic interference
- Highly reliable, compact size designed to offer users maximum simplicity

Fig. 3.4 SSR-100 DA

Table 3.9 Specifications of SSR

Input voltage	3-32V DC
Output voltage	24-380V AC
Output current	100 A
Cost	650/-

Structure and Operating Principle of SSR

SSRs use electronic circuits to transfer a signal. It consists of input circuit, drive circuit and output circuit. The structure of SSR is shown in Fig. 3.5.

SSR Working Steps:

1) When the input device (switch) is turned ON, current flows to the input circuits, the photocoupler operates, and an electric signal is transferred to the trigger circuit in the output circuits (Fig. 3.6 (a)).
2) Then switching element in the output circuit turns ON and the load current flows and the actuator (fan/fogger) turns ON (Fig. 3.6 (b)).
3) Similarly, when input device (switch) is turned OFF the photocoupler turns OFF, the trigger circuit turns OFF, which turns OFF the switching element (Fig. 3.6 (c)).

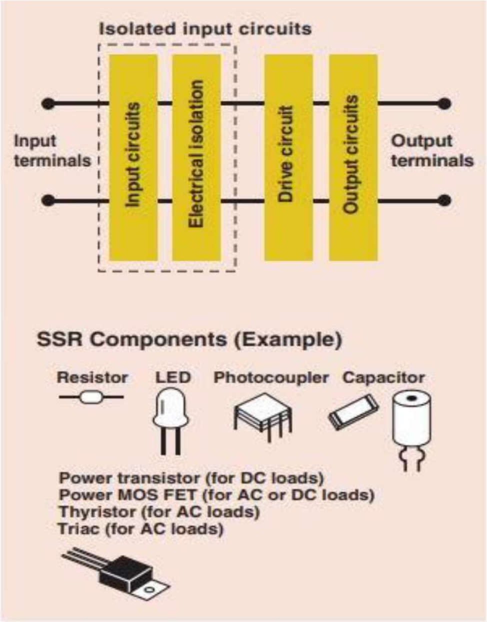
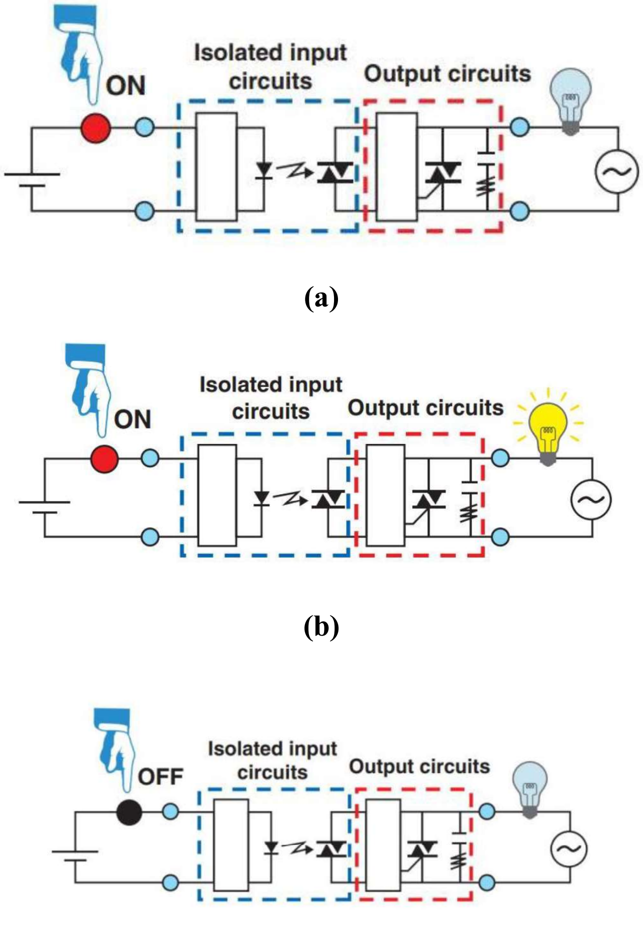


Fig. 3.5 Illustration of SSR structure



(a)

(b)

(c)

Fig. 3.6 Working of SSR

3.4.4.9 PCB Dotboard

It is a material for prototyping electronic circuits (also called DOT PCB). It is a thin, rigid sheet with holes pre-drilled at standard intervals across a grid, usually a square grid of 0.1 inches (2.54 mm) spacing (Fig. 3.7). These holes are ringed by round or square copper pads, though bare boards are also available. This is single sided Dot Matrix PCB board. It is also known as Perforated Board. This board can be used to quickly build and test circuit boards for testing and prototyping purposed or may be for making breakout boards. This board is made of high quality phenolic laminate and has holes drilled at the standard spacing of 0.1″ (2.54 mm). Specifications of PCB are shown in Table 3.10.

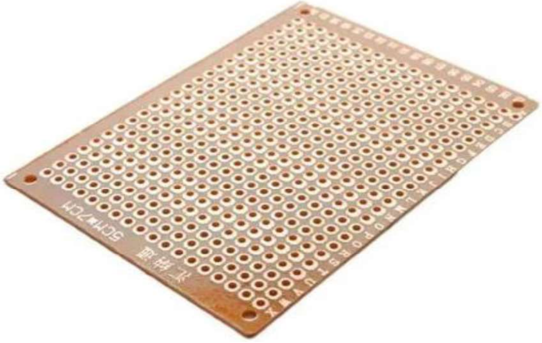


Fig. 3.7 PCB dotboard

Table 3.10 Specifications of PCB dotboard

PCB type	DOT PCB
PCB material	Phenolic
Hole spacing	2.54mm
PCB length	100mm
PCB width	75mm
Weight	25g
Cost	180/-

3.4.4.10 Extension Board

Extension boards are devices that allow multiple electrical devices (Arduino power supply, exhaust fans, fogger, charging Wi-Fi modem) to be connected to a single power outlet. A view of the extension board is shown in Plate 3.6 and its specifications are shown in Table 3.11.

Plate 3.6 Extension board

Table 3.11 Specifications of extension board

Brand	Choice
Input-output	240 V AC, 50 Hz
Item dimensions	36 x 8 x 5 Centimeters
Material	Plastic
Cost	Rs. 380/-

3.4.4.11 Energymeter

Measures the amount of electrical energy consumed (Plate 3.7). Specifications of energymeter are shown in Table 3.12.

Plate 3.7 Energymeter

Table 3.12 Specifications of energymeter

Phase	Two Phase
Output voltage	240V
Automation grade	Automatic
Display type	Digital Only
Frequency	50 Hz
Current rating	5-30 A
Cost	Rs. 475/-

3.4.4.12 Actuators Used

An actuator is a machine component or system that moves or controls the mechanism of the system. Sensors in the device sense the environment, then control signals are generated for the actuators according to the actions needed to perform. For the study, exhaust fans and foggers were used as the actuators.

a) Exhaust Fans

Two exhaust fans were used for the air exchange of greenhouse with that of outside. Whenever the temperature goes above the level of maximum set point the exhaust fans will operate**.** Exhaust fans used are shown in Plate 3.8 and its specifications are shown in Table 3.13.

(a) **(b)**

Plate 3.8 Exhaust fans

Table 3.13 Specifications of exhaust fans

Product	Fresh Air Fan
Model	Ultra Gold Hi-Speed Reversible
Quantity	1 U
Input	230 V, 50 Hz
Wattage	75 W
RPM	2200
Sweep	300 mm (12")
Color	Ash Grey
Cost	1800/-

b) Foggers

Four way foggers of flowrate pressure of 4 kg cm^{-2} were used for evaporative cooling inside the greenhouse (Plate 3.9). They were fitted at 1.5 m x 1.5 m spacing inside the greenhouse. Specifications of four way foggers are shown in Table 3.14. A one hp pump was used for the operation of fogger.

Plate 3.9 Four way foggers

Table 3.14 Specifications of fogger

Flow rate	30 lh^{-1}
Colour	Silver Gray
Nominal flow rate pressure	4 kg/cm^{2}
Nozzle size	0.61 mm
Cost	Rs 200/-

3.4.4.13 Solar Panel (Power Supply)

Uninterrupted power supply is required for the operation of the automation system. Moreover the automation system is to be automatically put ‘OFF’ during night hours. A solar panel of capacity 1 KW was used for power supply (Plate 3.10).

Plate 3.10 Solar power supply

3.4.5 Estimated Cost of IoT based Automation System

The estimated cost of the IoT based automation system is shown in Table 3.15. The developed system was found to be cost effective.

Table 3.15 Total estimated cost of IoT based automation system

Components	No.	Amount (Rs.)
Arduino Nano 33 IoT	1	Rs. 2500
DHT22 temperature and humidity Sensor	1	Rs. 400
BH1750 light sensor	1	Rs. 130
GSM module	1	Rs. 650
HI-LINK power module	1	Rs. 500
Liquid Crystal Display (LCD)	1	Rs. 700
Wi-Fi modem	1	Rs. 1500
Solid State Relay (SSR)	2	Rs. 650
PCB dotboard	1	Rs. 180
Extensionboard	1	Rs. 380
Energymeter	1	Rs. 475
Exhaust fans	2	Rs. 3600
Fogger	10	Rs. 2000
	Total	Rs. 17000 (Approx.)

3.4.6 Details of Softwares Used

Description of softwares used for the developed system are explained under the following subheads.

3.4.6.1 Arduino IoT Cloud

The Arduino IoT cloud is a platform that allows anyone to create IoT projects with a user-friendly interface, and an all in one solution for configuration, writing code, uploading and visualization. For the present study, Arduino IoT cloud showed the operating mode (automatic or manual), status of exhaust fan & fogger (ON/OFF) and it also showed reading of temperature, RH and light intensity in real-time, at intervals of one hour, one day, one week and 15 days as graphical insights. A view of the Arduino IoT cloud is shown in Fig. 3.8.

Features of Arduino IoT Cloud:

- Data Monitoring - learn how to easily monitor Arduino's sensor values through a dashboard.
- Variable Synchronisation - allow to synchronise variables across devices, enabling communication between devices with minimal coding.
- Scheduler - schedule jobs to go ON/OFF for a specific amount of time (seconds, minutes, hours).
- Over-The-Air (OTA) uploads - upload code to devices not connected to computer.
- Web hooks – integrate project with another service, such as If This Then That (IFTTT).
- Amazon Alexa Support - make project voice controlled with the Amazon Alexa integration.
- Dashboard Sharing - share data with other people around the world

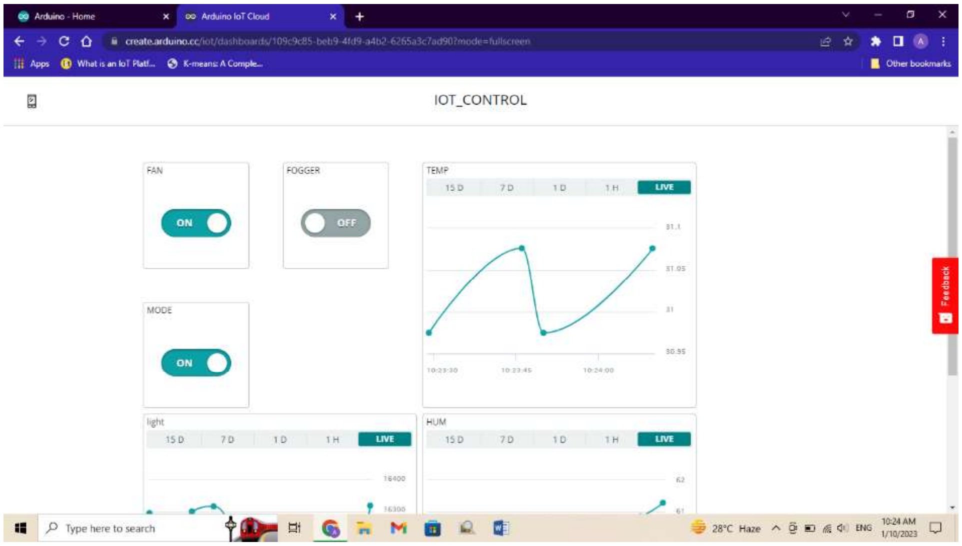
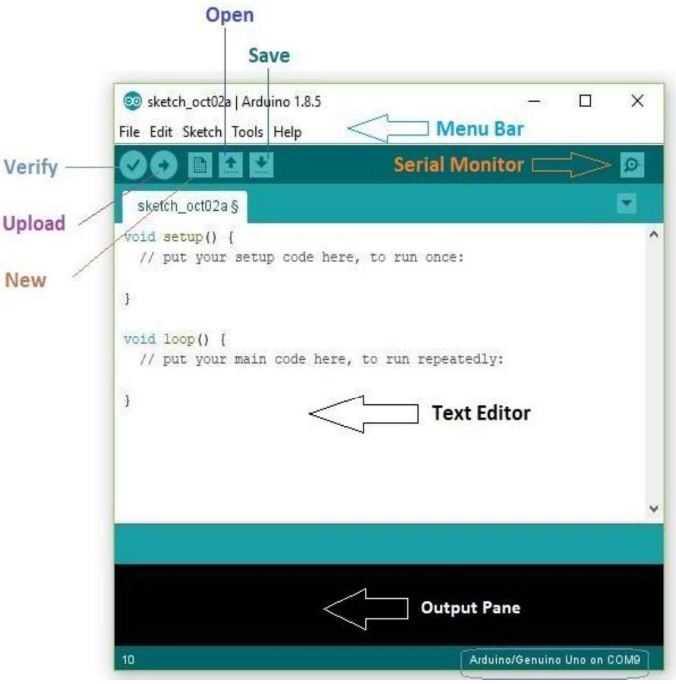


Fig. 3.8 A view of Arduino IoT cloud

3.4.6.2 Arduino Integrated Development Environment (IDE) Software

Arduino IDE was used to write code and upload it to the Arduino board (Fig. 3.9). Arduino supports C and C++ languages. C language was used here to write code. Code can be saved with the file extension *.ino*. The program written in this software can be upload to an Arduino board using a USB cable.

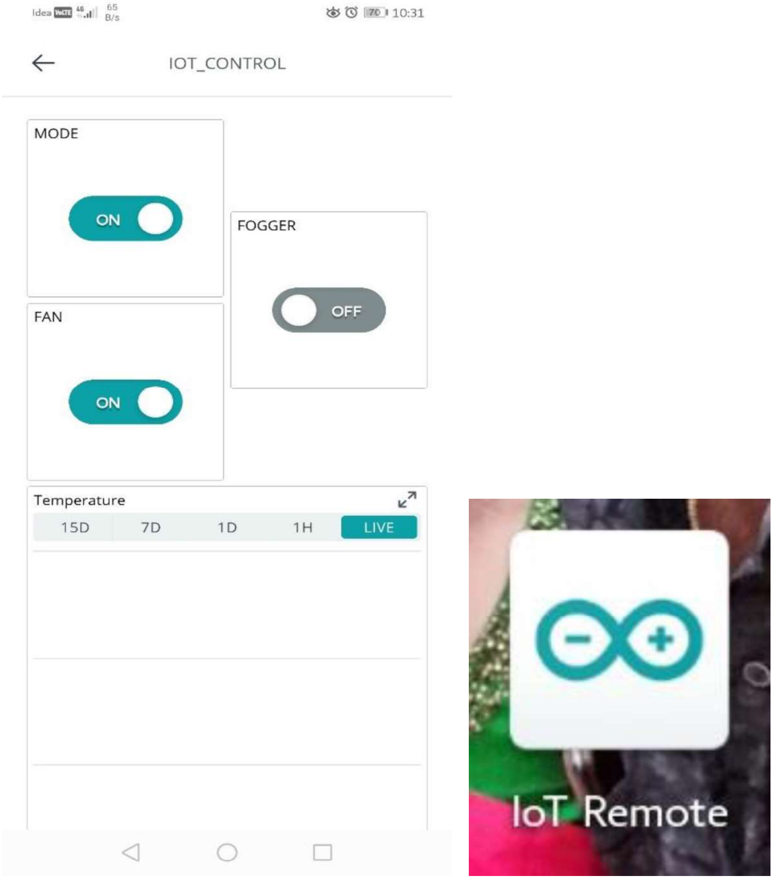


```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Fig. 3.9 View of Arduino IDE

3.4.6.3 Arduino IoT Cloud Remote Mobile Phone Application

The Arduino IoT cloud remote phone application let to control and monitor all dashboards in the Arduino IoT Cloud. It is an application which can be directly downloaded from google playstore. It is used to control and monitor all dashboards in the Arduino IoT Cloud. The data automatically stored in cloud variable are synchronized with other things. A view of the screenshot of Arduino IoT cloud remote mobile phone application with its icon is shown in Fig. 3.10.



IOT_CONTROL

MODE ON

FAN ON

FOGGER OFF

Temperature

(a) **(b)**

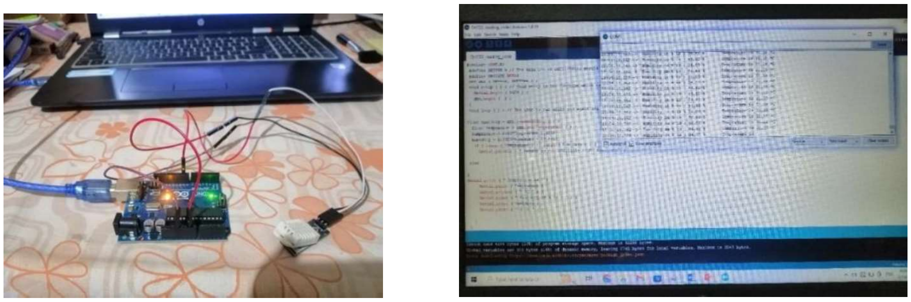
Fig. 3.10 Arduino IoT cloud remote mobile application (a) Screenshot of the IoT remote app (b) App icon

3.5 CALIBRATION OF SENSORS

Sensor calibration refers to the processes used to establish under specific circumstances, the relationship between the values of a measuring output by an instrument such as a sensor or measuring system and the corresponding values of a measuring instrument established with standards. Calibrated sensors are necessary for precise, consistent and repeatable measurement outputs. In this study, calibration was done for two sensors *i.e.*DHT22 temperature & humidity sensor and BH1750 light sensor. The procedure for the calibration of these two sensors are explained under the following subheads.

3.5.1 Calibration of DHT22 Sensor

Dry bulb & wet bulb Hygrometer were the instruments used for the measurement. Temperature and relative humidity value obtained from the sensor was compared with Hygrometer reading taken at the same time. Ambient temperature inside the polyhouse was taken from the dry bulb temperature reading in the hygrometer. Difference between dry bulb and wet bulb temperature were noted and relative humidity were calculated using a psychrometric chart. These values were recorded in an excel file and a scatter diagram was drawn with DHT22 reading in x-axis and dry bulb & wet bulb hygrometer reading in y-axis. Then trendline was added, linear option was chosen and a calibration equation was obtained. This calibration equation was used in Arduino code. A view of connection of DHT22 temperature & humidity sensor with Arduino board is shown in Plate 3.11(a) and the corresponding sensor reading is shown in Plate 3.11(b).



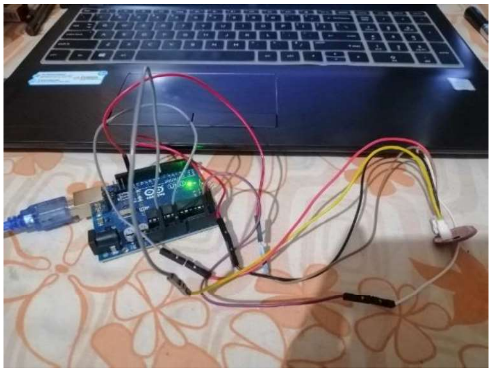
(a) **(b)**

Plate 3.11 Calibration of DHT22 temperature & humidity sensor (a) Connection (b) DHT22 sensor reading in serial monitor

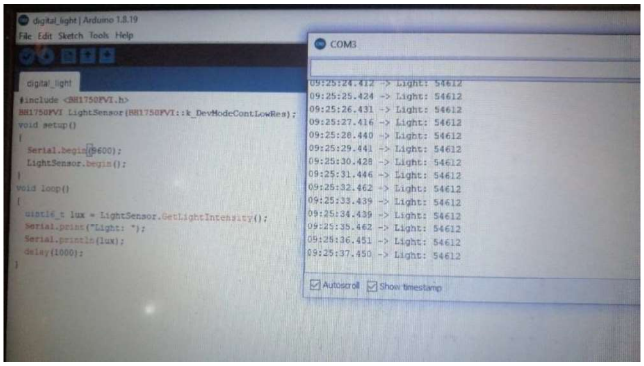
3.5.2 Calibration of BH1750 Light Sensor

.. The procedure for calibration of BH1750 light sensor is same as that of calibration of DHT22 temperature & humidity sensor. The light intensity value obtained from the BH1750 light sensor was compared with the luxmeter reading.

Scatter diagram was drawn with BH1750 light sensor reading in x-axis and luxmeter reading in y-axis. Then trendline was added, linear option was chosen and a calibration equation was obtained. This calibration equation was used in Arduino code. A view of connection of BH1750 light sensor with Arduino board is shown in Plate 3.12 (a) and the corresponding sensor reading is shown in Plate 3.12 (b).



(a)



```
#include <BH1750FVI.h>
BH1750FVI LightSensor(BH1750FVI::k_DevModeContLowRes);
void setup()
{
  Serial.begin(9600);
  LightSensor.begin();
}
void loop()
{
  uint16_t lux = LightSensor.GetLightIntensity();
  Serial.print("Light: ");
  Serial.println(lux);
  delay(1000);
}
```

```
09:25:24.412 -> Light: 54612
09:25:25.424 -> Light: 54612
09:25:26.431 -> Light: 54612
09:25:27.416 -> Light: 54612
09:25:28.440 -> Light: 54612
09:25:29.441 -> Light: 54612
09:25:30.428 -> Light: 54612
09:25:31.446 -> Light: 54612
09:25:32.462 -> Light: 54612
09:25:33.439 -> Light: 54612
09:25:34.439 -> Light: 54612
09:25:35.462 -> Light: 54612
09:25:36.451 -> Light: 54612
09:25:37.450 -> Light: 54612
```

☑ Autoscroll ☑ Show timestamp

(b)

Plate 3.12 Calibration of BH1750 light sensor (a) Connection
(b) BH1750 light sensor reading in serial monitor

3.6 ASSEMBLING AND INSTALLATION OF IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSYTEM FOR GREENHOUSE

The steps involved in the assembling and installation of IoT based automation system in the polyhouse are as follows:

1) Sensors, actuators and other components were connected to Arduino Nano 33 IoT

2) Automation system was powered by using a Hi-Link power module.
3) All these components except sensors were kept inside a plastic box to give protection from exposure to weather.

3.7 THE OVERALL SYSTEM DESIGN AND BLOCK DIAGRAM

The block diagram of the system interfaced with all hardware and software components with Arduino Nano 33 IoT microcontroller is shown in the Fig. 3.11. DHT22 temperature & humidity sensor, BH1750 light sensor, LCD display and GSM were connected to Arduino Nano 33 IoT. Actuators, exhaust fans and foggers were connected to the microcontroller by means of Solid State Relay (SSR). Arduino Nano 33 IoT had inbuilt Wi-Fi module and by using this module, data (temperature, RH and light intensity) was displayed both in mobile phone and PC as dashboards.

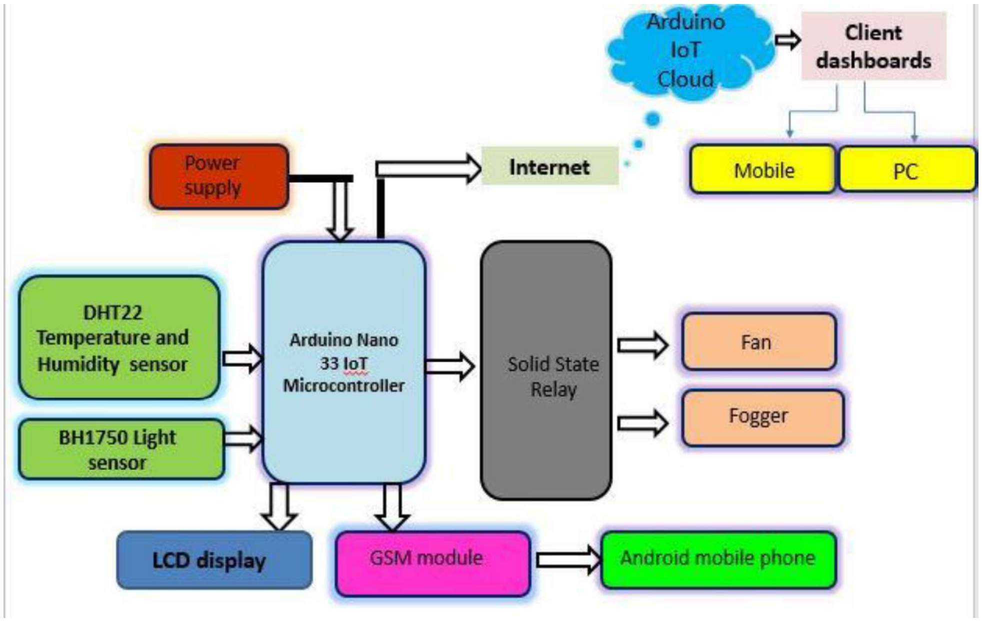


Fig. 3.11 System design and block diagram

3.8 THE VARIOUS FUNCTIONALITIES OF THE SYSTEM

The IoT based system had two sections on the basis of their functionalities *i.e.* transmitter unit and receiver unit. Various components involved in both units and their responsibilities are explained in this section.

3.8.1 Transmitter Unit

The responsibility of the transmitting unit is as follows:

1) Measure temperature and relative humidity using DHT22 temperature and humidity sensor, light using BH1750 light sensor which are interfaced with Arduino Nano 33 IoT
2) Send the data from the Arduino board to the receiving unit using the Wi-Fi module (ublox) which is inbuilt in the microcontroller.

The Transmitter unit of the system consists of the following units:

- Arduino Nano 33 IoT with inbuit Wi-Fi module
- DHT22 Sensor
- BH1750 Light sensor

3.8.2 Receiver Unit

The responsibilities of the receiver unit is as follows:

1) Receive the temperature, relative humidity and light data from the transmitting unit.
2) Receive data from the Arduino board (with inbuilt Wi-Fi module) in real-time.

The receiver unit of the system consists of the following units:

- Solid State Relay
- Arduino IoT Cloud
- Wi-Fi modem

3.9 INTERFACING VARIOUS COMPONENTS WITH ARDUINO NANO 33 IoT

Arduino Nano 33 IoT was the microcontroller used and it was the main processing unit. Sensors, actuators, SSR, GSM and LCD display were interfaced with this microcontroller. Interfacing various components with Arduino Nano 33 IoT and the source code developed is explained in the following subheads.

3.9.1 Interfacing Arduino Nano 33 IoT and DHT22 Temperature & Humidity Sensor

Various pins of Arduino Nano 33 IoT and DHT22 sensor are shown in Fig. 3.12. The DHT22 module has three pins, two of which are power pins (VDD/VCC and GND), and one is a data pin. For the study, 5V, Digital pin D2 and GND (ground) pins of Arduino were connected to VCC/VDD, DATA and GND pins of

DHT22 sensor respectively. Arduino pins and its corresponding connection with DHT22 pins is shown in Table 3.16.

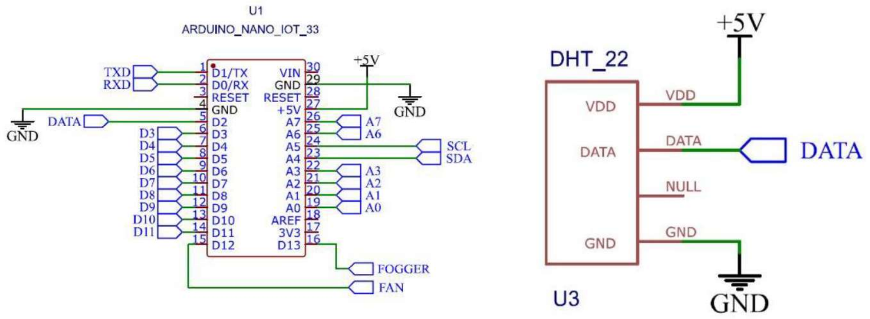


Fig. 3.12 Various pins of Arduino Nano 33 IoT, DHT22 sensor and their interfacing

Table 3.16 Arduino pins and its connection with DHT22 temperature & humidity sensor pins

Arduino pins	DHT22 temperature & humidity sensor pins
5V	VCC/VDD
D2 (Digital pin)	DATA
GND	GND

3.9.2 Interfacing Arduino Nano 33 IoT and BH1750 Light Sensor

BH1750 light sensor has four pins *i.e.* VCC, serial clock pin (SCL), serial data pin (SDA) and GND pins (Fig. 3.13). For the study, VCC pin of the sensor was connected to 5V pin of Arduino. SCL pin of sensor was connected to analog pin A5 of Arduino. SDA pin of sensor was connected to analog pin A4 of Arduino. GND pin of the sensor was connected to any of the GND pin of Arduino. Arduino pins and its corresponding connection with BH1750 light sensor pins are shown in Table 3.17.

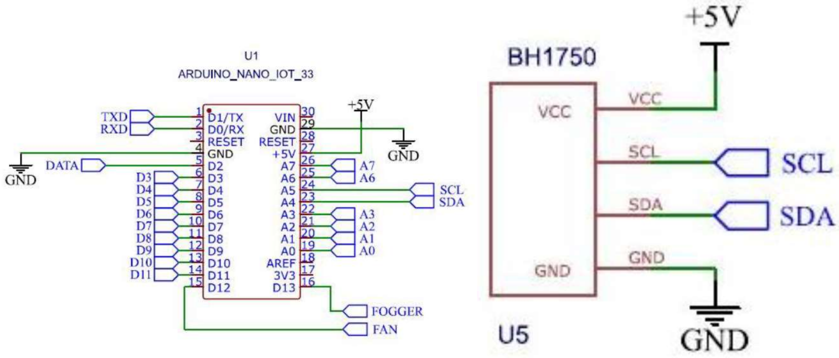


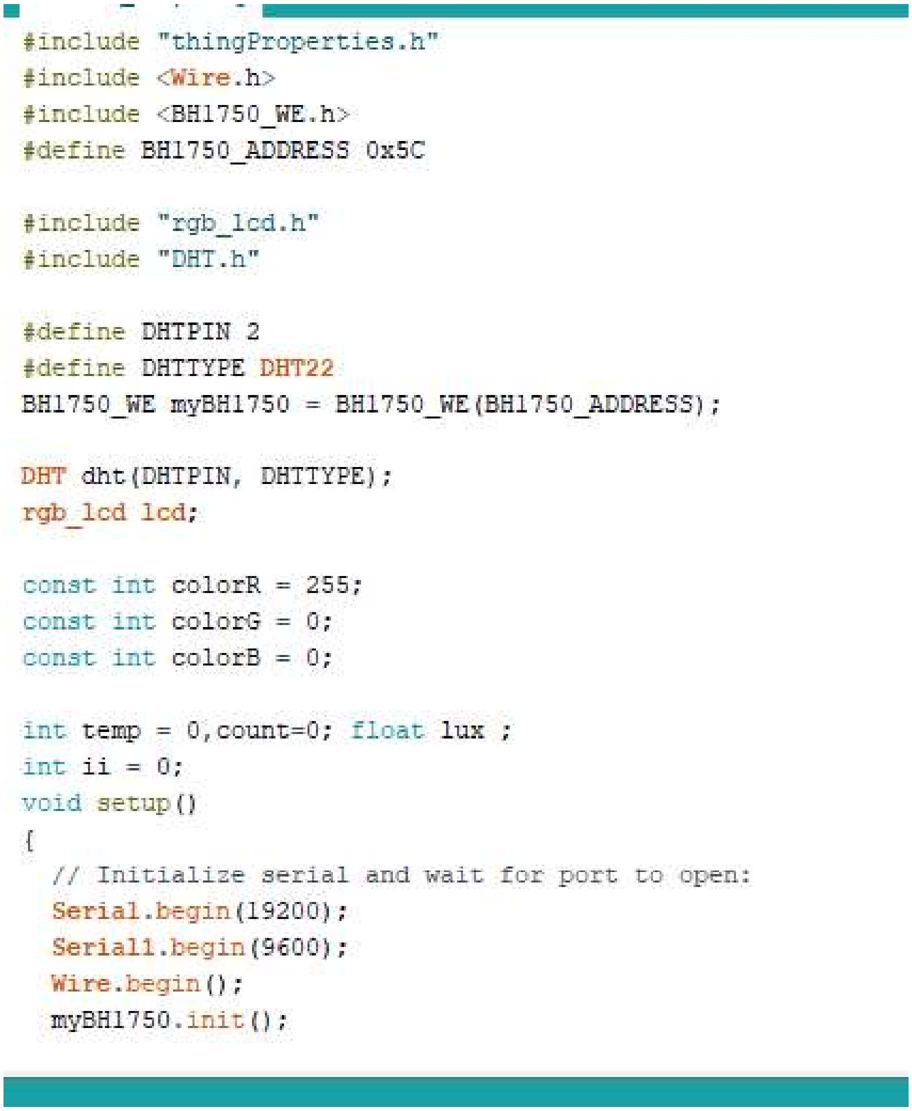
Fig. 3.13 Various pins of Arduino Nano 33 IoT, BH1750 light sensor and their interfacing

Table 3.17 Arduino pins and its connection with BH1750 light sensor pins

Arduino pins	**BH1750 Light Sensor pins**
5V	VCC
A5	SCL
A4	SDA
GND	GND

3.9.3 Source Code/ Program Developed for Reading Temperature, Relative Humidity and Light

After successful interfacing of DHT22 temperature & humidity sensor and BH1750 light sensor with Arduino Nano 33 IoT a source code was developed for sensing temperature, RH and light intensity. Source code/ program developed for reading temperature, relative humidity and light is shown in Fig. 3.14. All the necessary libraries were included and defined the sensor types and sensor pins. In the setup function serial for debugging, libraries and display were initialized as shown in Fig. 3.14(a). In the loop function temperature, RH and light intensity were measured in Degree Celsius (°C), percentage (%) and lux (lx) respectively. Calibration equations obtained for temperature, RH and light intensity was also incorporated in the Arduino code as shown in Fig. 3.14 (b).



```
#include "thingProperties.h"
#include <Wire.h>
#include <BH1750_WE.h>
#define BH1750_ADDRESS 0x5C

#include "rgb_lcd.h"
#include "DHT.h"

#define DHTPIN 2
#define DHTTYPE DHT22
BH1750_WE myBH1750 = BH1750_WE(BH1750_ADDRESS);

DHT dht(DHTPIN, DHTTYPE);
rgb_lcd lcd;

const int colorR = 255;
const int colorG = 0;
const int colorB = 0;

int temp = 0,count=0; float lux ;
int ii = 0;
void setup()
{
  // Initialize serial and wait for port to open:
  Serial.begin(19200);
  Serial1.begin(9600);
  Wire.begin();
  myBH1750.init();
```

(a)

```
  initProperties();
  dht.begin();


  lcd.begin(16, 2);
  lcd.setRGB(colorR, colorG, colorB);
  mODE = 0; delay(1000);
  ReceiveMode();
  }
  void loop()
{
  seriallEvent();
  if (temp == 1)
  {
    check();
    temp = 0;
    ii = 0;
    delay(1000);
  }

  light = myBH1750.getLux();

  HUM = 0.7203*dht.readHumidity()+12.872;
  tEMP= 1.0182*dht.readTemperature()-2.4222;
  light=0.9999*myBH1750()+3.193;
  //float f = dht.readTemperature(true);
  lcd.clear();
  displayOut();
```

(b)

Fig. 3.14 Source code/program developed for reading temperature, Relative Humidity and light intensity

3.9.4 Interfacing Arduino Nano 33 IoT and Hi-link Power Module

Hi-link power module was used to give power supply to microcontroller. A view of various pins of Arduino Nano 33 IoT, Hi-Link power module and their interfacing is shown in Fig. 3.15. Hi-link power module has four pins *i.e.* VCC pin (+VO), ground pin (-VO), neutral pin (AC) and a phase pin (AC). The various pins and its description is shown in Table 3.18. +VO,-VO, AC (phase) and AC (neutral) pins of hi-link power module was connected to 5V, GND, 230 VAC and 0VAC pins of Arduino Nano 33 IoT respectively (Table 3.19).

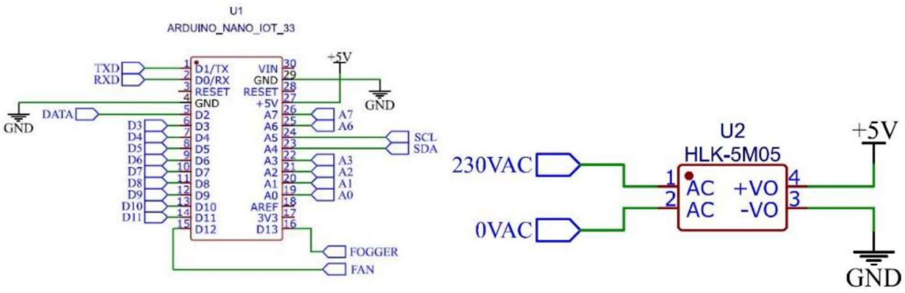


Fig. 3.15 Various pins of Arduino Nano 33 IoT, Hi-Link power module and their interfacing

Table 3.18 Various pins of Hi-Link power module and their description

Pin Number	Pin Name	Description
1	AC	Connect to the phase/live terminal
2	AC	Connect to the neutral terminal
3	-Vo	Output negative voltage here ground
4	+Vo	Outputs +5V regulated voltage

Table 3.19 Arduino pins and its connection with Hi-Link power module pins

Arduino pins	Hi-Link power module pins
5V	+VO
GND	-VO
230 VAC	AC (phase)
0VAC	AC (neutral)

3.9.5 Interfacing Arduino Nano 33 IoT with GSM

GSM was used to monitor and control microclimate parameters through mobile phone by sending SMS. GSM has four pins. They are VCC, TXD,RXD and GND pins. TXD pin is used to send the data and RXD pin is used to receive the data. Arduino Nano 33 IoT pins and GSM pins with interfacing is shown in Fig. 3.16. VCC, TXD, RXD and GND pins of GSM were connected with 5V, D1, D0 and GND pins of Arduino Nano 33 IoT (Table 3.20).

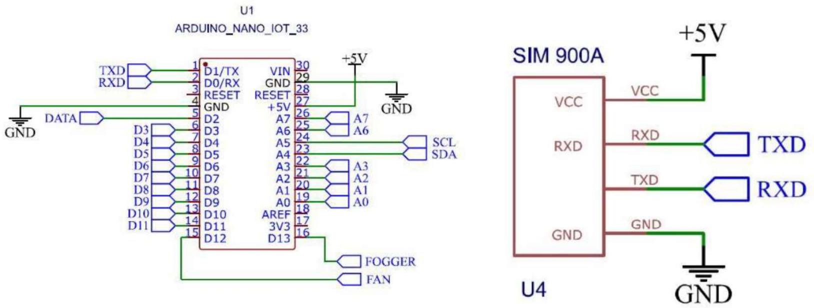


Fig. 3.16 Various pins of Arduino Nano 33 IoT, GSM and their interfacing

Table 3.20 Arduino pins and its connection with GSM pins

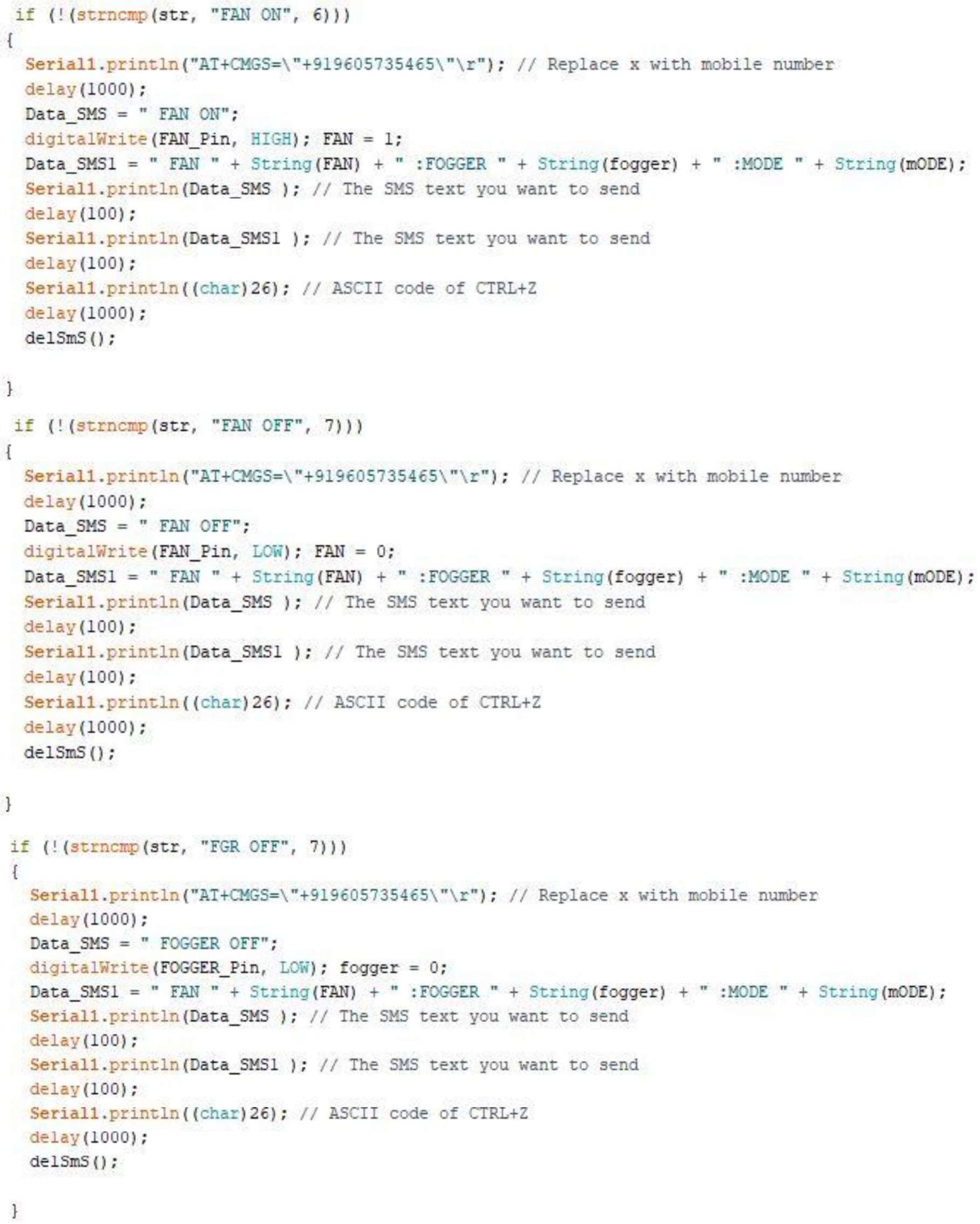
Arduino pins	GSM pins
5V	VCC
D1	TXD
D0	RXD
GND	GND

3.9.6 Source Code/ Program Developed for Connecting GSM with Arduino

A VodafoneIdea sim was inserted in the GSM. Monitoring and controlling was made possible through GSM connected mobile phone. Two modes were set for operation: Manual and Auto mode. All the pins were initialised and the mobile number to which SMS should be sent was given in the code. Various void functions were used to establish communication between Arduino Nano 33 IoT and GSM module by AT commands to send SMS to the provided mobile number.The commands used for monitoring and controlling using GSM is shown in Table 3.21.

```
void ReceiveMode()
{

  Serial1.println("AT+CNMI=2,2,0,0,0");
  delay(1000);
  Serial1.println("AT+CMGF=1");
  delay(500);
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Serial1.println("System is ready to receive commands.");// The SMS text you want to send
  delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);

}

void delSmS()
{
  Serial1.print("AT+CMGDA=\"");
  Serial1.println("DEL ALL\"");
  delay(500);
}
void delSmS()
{
  Serial1.print("AT+CMGDA=\"");
  Serial1.println("DEL ALL\"");
  delay(500);
}

void check()
{
  if (!(strncmp(str, "INFO", 4)))
  {
    Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
    delay(1000);
    Data_SMS = " TEMP:HUM > " + String(tEMP) + " C : " + String(HUM) + "% : Light" + String(light) + "lux";
    Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
    Serial1.println(Data_SMS ); // The SMS text you want to send
    delay(100);
    Serial1.println(Data_SMS1 ); // The SMS text you want to send
    delay(100);
    Serial1.println((char)26); // ASCII code of CTRL+Z
    delay(1000);
    delSmS();

  }
```

(a)



```
 if (!(strncmp(str, "FAN ON", 6)))
{
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Data_SMS = " FAN ON";
  digitalWrite(FAN_Pin, HIGH); FAN = 1;
  Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
  Serial1.println(Data_SMS ); // The SMS text you want to send
  delay(100);
  Serial1.println(Data_SMS1 ); // The SMS text you want to send
  delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);
  delSmS();

}
 if (!(strncmp(str, "FAN OFF", 7)))
{
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Data_SMS = " FAN OFF";
  digitalWrite(FAN_Pin, LOW); FAN = 0;
  Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
  Serial1.println(Data_SMS ); // The SMS text you want to send
  delay(100);
  Serial1.println(Data_SMS1 ); // The SMS text you want to send
  delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);
  delSmS();

}
 if (!(strncmp(str, "FGR OFF", 7)))
 {
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Data_SMS = " FOGGER OFF";
  digitalWrite(FOGGER_Pin, LOW); fogger = 0;
  Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
  Serial1.println(Data_SMS ); // The SMS text you want to send
  delay(100);
  Serial1.println(Data_SMS1 ); // The SMS text you want to send
  delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);
  delSmS();

 }
```

(b)

```
 if (!(strncmp(str, "FGR ON", 6)))
{
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Data_SMS = " FOGGER ON";
  digitalWrite(FOGGER_Pin, LOW); fogger = 1;
  Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
  Serial1.println(Data_SMS ); // The SMS text you want to send
  delay(100);
  Serial1.println(Data_SMS1 ); // The SMS text you want to send
  delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);
  delSmS();

}

 if (!(strncmp(str, "AUTO ON", 7)))
 {
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Data_SMS = "AUTO MODE ON";
  mODE = 1;
  Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
  Serial1.println(Data_SMS ); // The SMS text you want to send
  delay(100);
  //Serial1.println(Data_SMS1 ); // The SMS text you want to send
  //delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);
  delSmS();

 }

 if (!(strncmp(str, "AUTO OFF", 8)))
 {
  Serial1.println("AT+CMGS=\"+919605735465\"\r"); // Replace x with mobile number
  delay(1000);
  Data_SMS = "AUTO MODE OFF";
  mODE = 0;
  Data_SMS1 = " FAN " + String(FAN) + " :FOGGER " + String(fogger) + " :MODE " + String(mODE);
  Serial1.println(Data_SMS ); // The SMS text you want to send
  delay(100);
  //Serial1.println(Data_SMS1 ); // The SMS text you want to send
  //delay(100);
  Serial1.println((char)26); // ASCII code of CTRL+Z
  delay(1000);
  delSmS();

 }
}
```

(c)

Fig. 3.17 Source code/ program developed for connecting GSM with Arduino Nano 33 IoT

Table 3.21 Commands used for monitoring and controlling using GSM

Commands	Description
/INFO/	Mode, temperature, relative humidity ,light value and actuators condition
/AUTO ON/	Auto mode ON
/AUTO OFF/	Auto mode OFF
/FAN ON/	Fan ON
/FAN OFF/	Fan OFF
/FGR ON/	Fogger ON
/FGR OFF/	Fogger OFF

3.9.7 Interfacing Arduino Nano 33 IoT with LCD

Liquid Crystal Display (LCD) was used to display the characters. It has four pins VCC, GND, SCL and SDA pins. The VCC pin is used to supply the power to the LCD. GND pin is the ground pin used to connect the LCD to the ground. SCL and SDA are I2C clock pin and data pin respectively. Various pins of Arduino Nano 33 IoT, LCD and their interfacing is shown in Fig. 3.18. LCD pins connection with Arduino Nano 33 IoT pins is shown in Table 3.22.

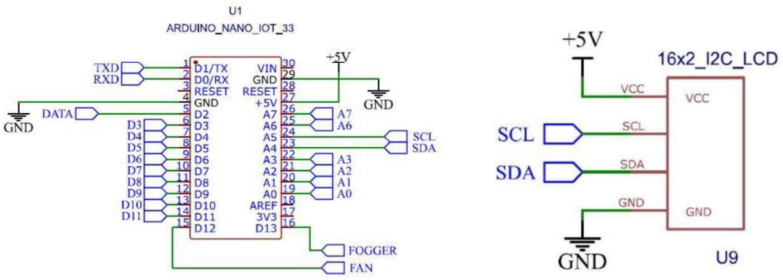


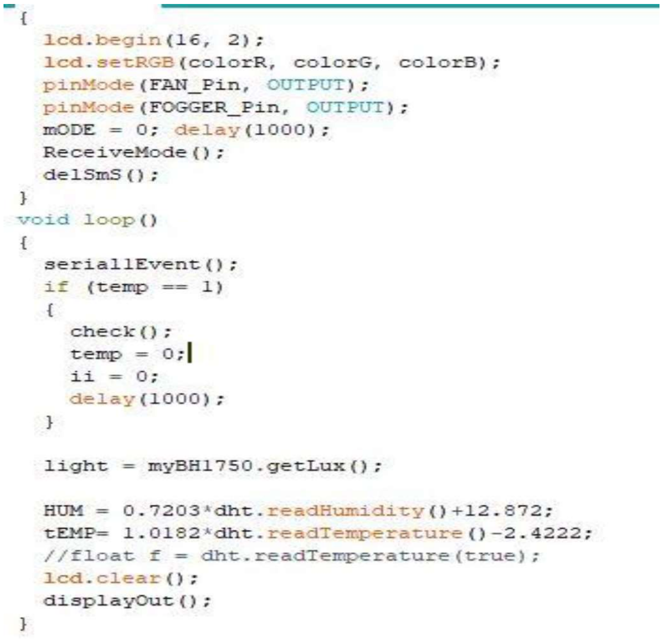
Fig. 3.18 Various pins of Arduino Nano 33 IoT, LCD and their interfacing

Table 3.22 Arduino pins and its connection with LCD pins

Arduino pins	LCD pins
5V	VCC
A5	SCL
A4	SDA
GND	GND

3.9.8 Source Code/Program Developed for Connecting LCD with Arduino

Source code/program developed for connecting LCD with Arduino is shown in Fig. 3.19. After the installation of suitable libraries values to be displayed were given the code. The address of LCD was obtained by running the address code. Readings of temperature, RH and light intensity, status of actuators, and mode was displayed in LCD.



```
{
  lcd.begin(16, 2);
  lcd.setRGB(colorR, colorG, colorB);
  pinMode(FAN_Pin, OUTPUT);
  pinMode(FOGGER_Pin, OUTPUT);
  mODE = 0; delay(1000);
  ReceiveMode();
  delSmS();
}
void loop()
{
  seriallEvent();
  if (temp == 1)
  {
    check();
    temp = 0;
    ii = 0;
    delay(1000);
  }

  light = myBH1750.getLux();

  HUM = 0.7203*dht.readHumidity()+12.872;
  tEMP= 1.0182*dht.readTemperature()-2.4222;
  //float f = dht.readTemperature(true);
  lcd.clear();
  displayOut();
}
```

Fig. 3.19 Source code/program developed for connecting LCD with Arduino Nano 33 IoT

3.9.9 Connecting SSR with Arduino for Exhaust Fan Actuation

Solid State Relays (SSRs) were connected with Arduino Nano 33 IoT for the operation of exhaust fans. The switch for both the exhaust fans were made common. FAN and GND pins of SSR were connected with D12 and GND pins of Arduino. 230VAC and 0VAC indicates phase and neutral pins (Fig. 3.20 and Table 3.23).

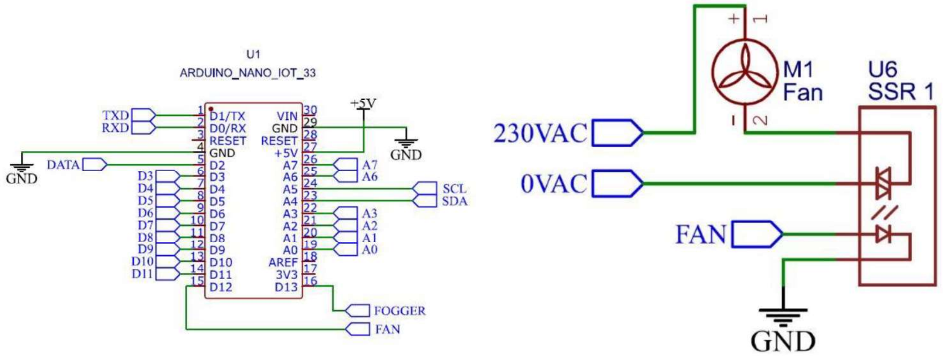


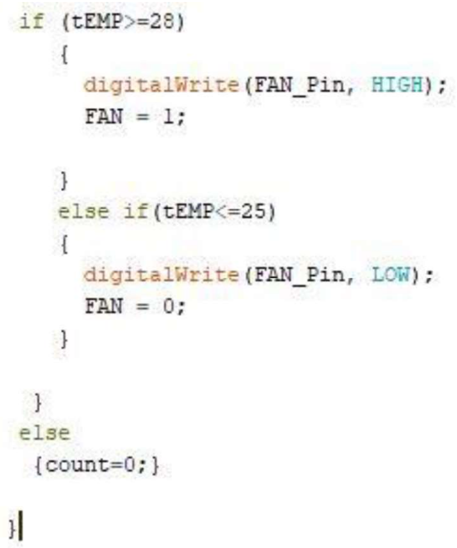
Fig. 3.20 Various pins of Arduino Nano 33 IoT, SSR pins for operating exhaust fans and their interfacing

Table 3.23 Arduino pins and its connection with SSR pins for exhaust fan actuation

Arduino pins	**SSR pins**
D12	FAN
GND	GND

3.9.10 Source Code/ Program Developed for Actuating the Exhaust Fan

For actuating the exhaust fan code was developed based on the temperature value as shown in Fig. 3.21. Both exhaust fans were controlled by one switch. The code was developed in such a way that when the temperature cross above 28°C the exhaust fan operate until it reach below 25°C.



```
 if (tEMP>=28)
     {
       digitalWrite(FAN_Pin, HIGH);
       FAN = 1;

     }
    else if(tEMP<=25)
     {
       digitalWrite(FAN_Pin, LOW);
       FAN = 0;
     }

   }
 else
   {count=0;}

}
```

Fig. 3.21 Source code/ program developed for actuating the exhaust fan

3.9.11 Connecting SSR with Arduino for Fogger Actuation

SSRs were used for operating foggers. A one hp pump was used to operate fogger. Various pins of Arduino Nano 33 IoT, SSR pins for operating foggers, their interfacing and connections are shown in Fig. 3.22 and Table 3.24 respectively.

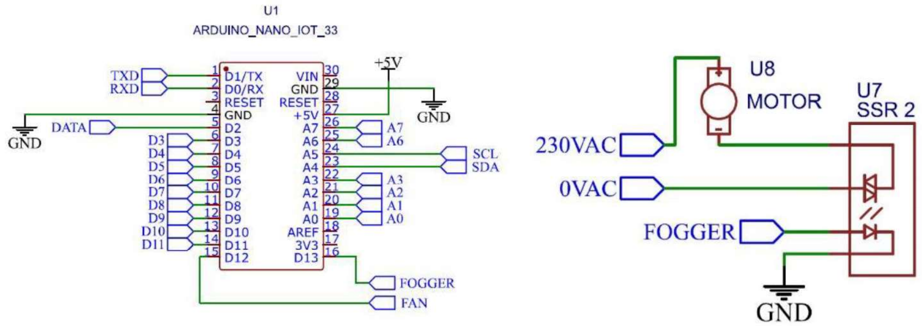


Fig. 3.22 Various pins of Arduino Nano 33 IoT, SSR pins for operating foggers and their interfacing

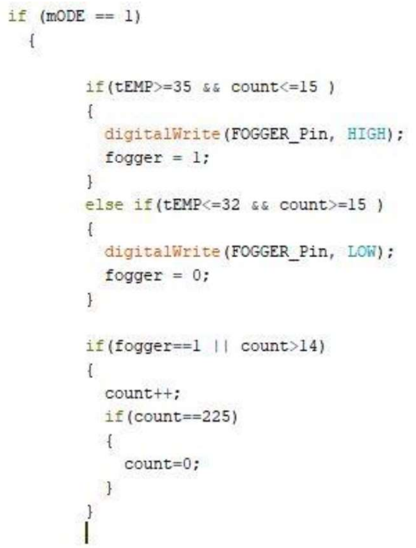
Table 3.24 Arduino pins and its connection with SSR pins for operating foggers

Arduino pins	SSR pins
D13	FOGGER
GND	GND

3.9.12 Source Code/ Program Developed for Actuation of Fogger

A program was developed for operating the fogger as shown in Fig. 3.23. The code was developed in the following way:

- The fogger operation was fixed at an interval of 15 minutes and operating time 1 minute.
- Whenever the temperature was above 35°C fogger operated for 1 min.
- If the temperature not reached 32°C after one operation the next operation was allowed only after 15 minutes.
- Whenever the temperature reached 32°C in between the operation of fogger it stops automatically before the completion.
- Otherwise the next operation of fogger starts only after 15 min and the process continues.



```
if (mODE == 1)
  {

        if(tEMP>=35 && count<=15 )
        {
          digitalWrite(FOGGER_Pin, HIGH);
          fogger = 1;
        }
        else if(tEMP<=32 && count>=15 )
        {
          digitalWrite(FOGGER_Pin, LOW);
          fogger = 0;
        }

        if(fogger==1 || count>14)
        {
          count++;
          if(count==225)
          {
            count=0;
          }
        }
```

Fig. 3.23 Source code/ program developed for fogger operation

3.10 OVERALL CIRCUIT DIAGRAM OF THE IoT BASED SYSTEM

All the components DHT22 temperature & humidity sensor, BH1750 light sensor, GSM module, Hi-link power module, LCD display and the actuators (exhaust fans and foggers) were interfaced to the Arduino Nano 33 IoT microcontroller. An overall circuit diagram showing connection of Arduino Nano 33 IoT with various components is shown in Fig. 3.24.

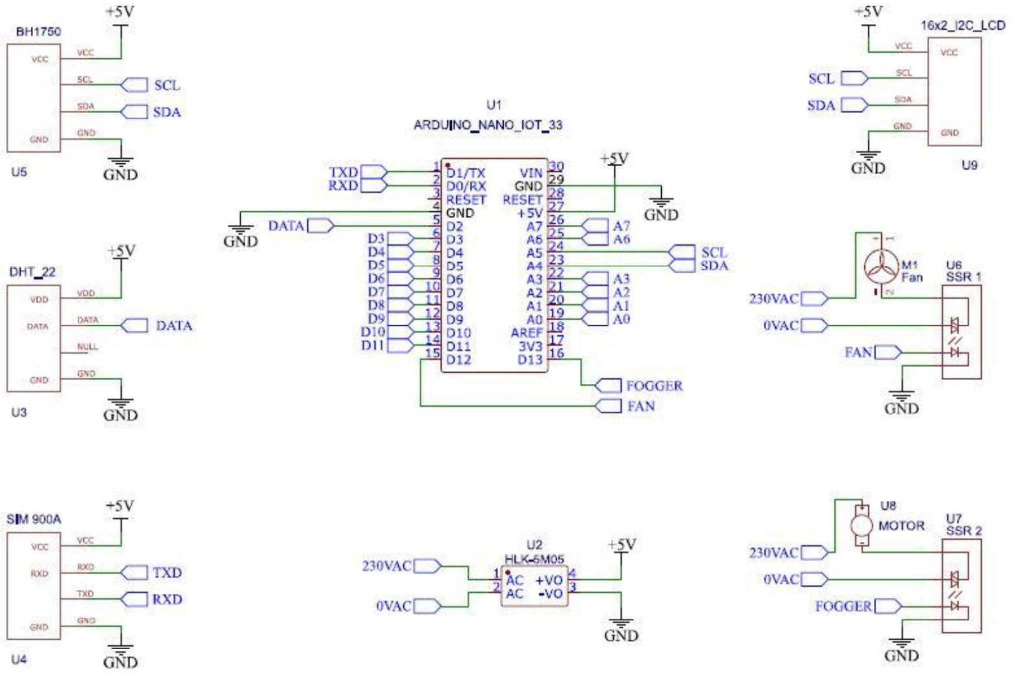


Fig. 3.24 Overall circuit diagram

3.11 SETTING UP THE ARDUINO IoT CLOUD

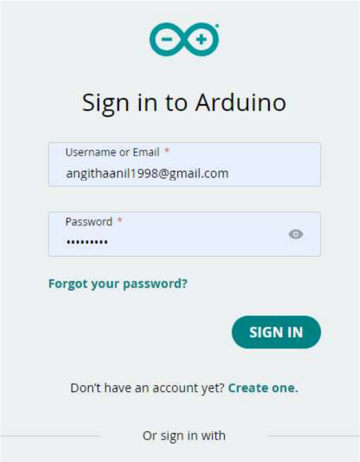
Various steps followed are:

1) Creating an Arduino account
2) Go to the Arduino IoT cloud
3) Creating a ‘Thing’
4) Configuring a device
5) Creating variables
6) Connecting to a network
7) Editing the sketch

8) Creating a Dashboard
9) Downloading the data

3.11.1 Creating an Arduino Account

In order to start an Arduino account log in/sign up in the Arduino site as shown in Fig. 3.25.



Sign in to Arduino

Username or Email *
angithaanil1998@gmail.com

Password *
•••••••••

Fig. 3.25 Arduino account creation

3.11.2 Go to the Arduino IoT Cloud

After signed up, it was possible to access the Arduino IoT Cloud from any page on 'arduino.cc' by clicking on the four dots menu in the top right corner. The same could be done by going directly to the 'Arduino IoT cloud'. Navigation to IoT cloud is shown in Fig. 3.26.

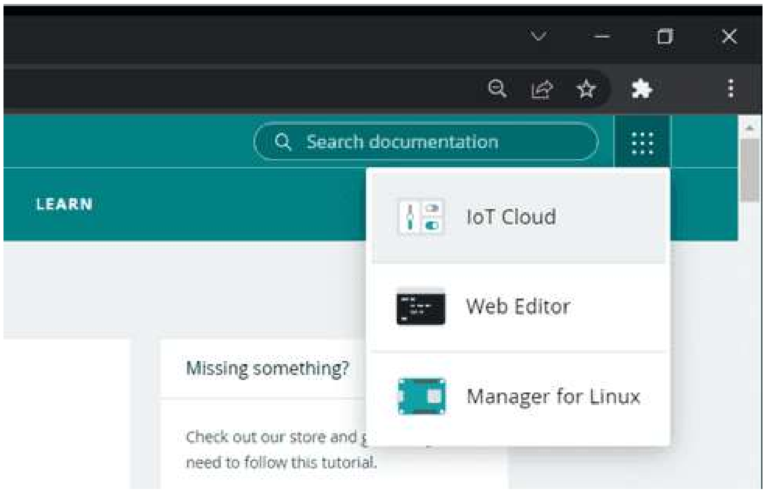
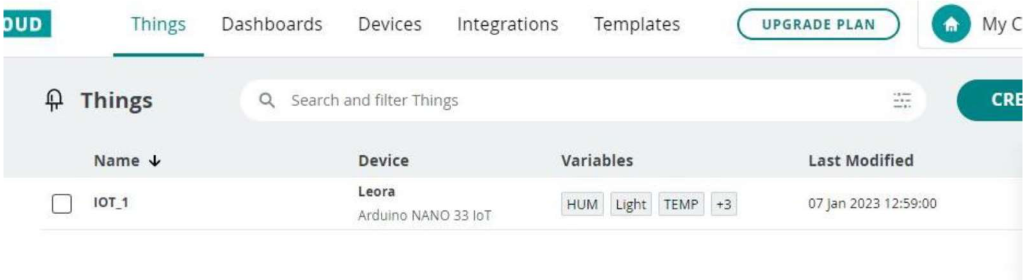


Fig. 3.26 Navigation to IoT cloud

3.11.3 Creating a ‘Thing’

In the ‘Thing overview’, what device to use, what Wi-Fi network to connect to, could be chosen and create variables that we can monitor and control. Hence, in this study, device Arduino Nano 33 IoT, Jio Wi-Fi network, the variables temperature, RH and light were selected .This was the main configuration space, where all changes were automatically generated into a special sketch file (coding). The ‘Thing overview’ is shown in Fig. 3.27.

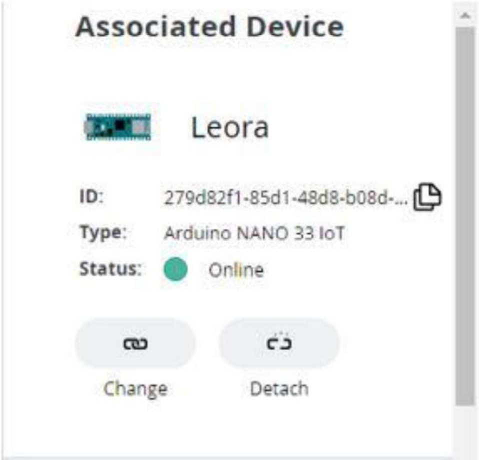


	Name	Device	Variables	Last Modified
☐	IOT_1	Leora Arduino NANO 33 IoT	HUM Light TEMP +3	07 Jan 2023 12:59:00

Fig. 3.27 The Thing overview

3.11.4 Configuring a Device

Devices can easily be added and linked to a Thing in this section. The Arduino IoT Cloud requires computer to have the ‘Arduino Create Agent’ installed. Arduino Nano 33 IoT device was added and linked to thing by clicking on the “Select device” button in the ‘Thing overview’. This could also be done by “Configure new device” option. The details are shown in Fig. 3.28.



Leora

ID: 279d82f1-85d1-48d8-b08d-...

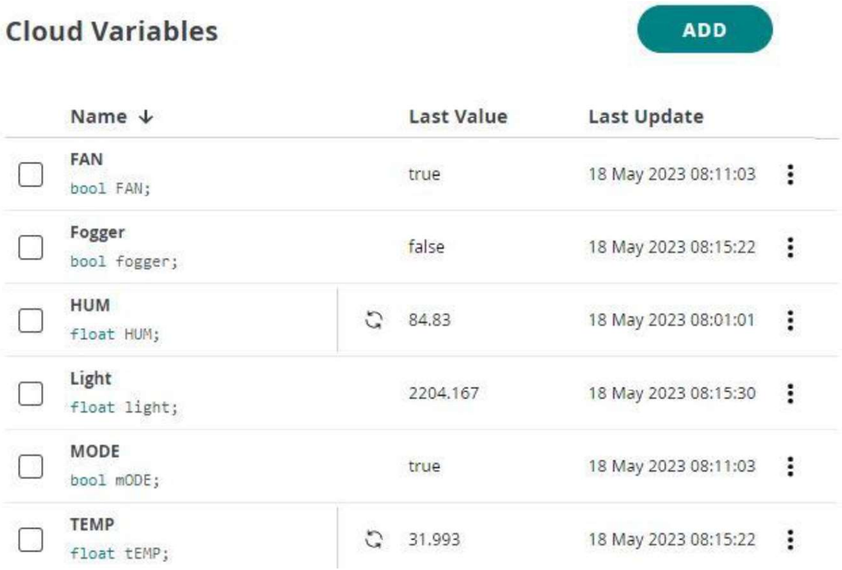
Type: Arduino NANO 33 IoT

Status: ● Online

Fig. 3.28 Configuring Arduino Nano 33 IoT

3.11.5 Creating Variables

The variables created were automatically generated into a sketch file. There are several data types such as int, float, boolean, long and char. Among these ‘float’ and ‘Boolean’ were chosen for the setup. Name, data type, update setting and interaction mode were also chosen by clicking on the “Add variable” button (Fig. 3.29).

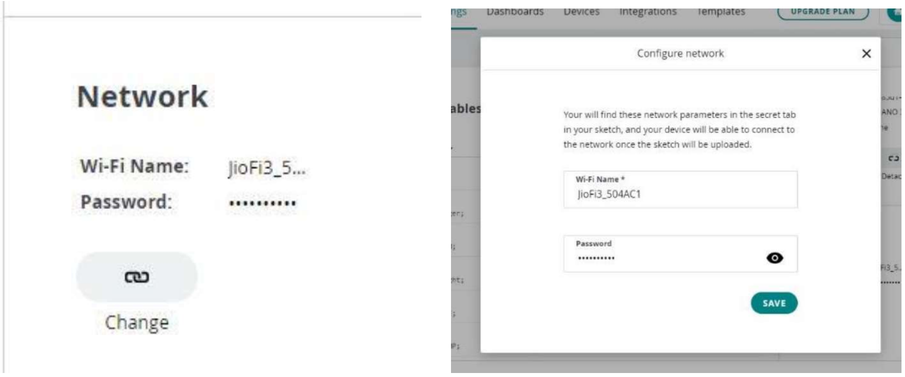


	Name ↓		Last Value	Last Update	
☐	FAN `bool FAN;`		true	18 May 2023 08:11:03	⋮
☐	Fogger `bool fogger;`		false	18 May 2023 08:15:22	⋮
☐	HUM `float HUM;`	⟳	84.83	18 May 2023 08:01:01	⋮
☐	Light `float light;`		2204.167	18 May 2023 08:15:30	⋮
☐	MODE `bool mODE;`		true	18 May 2023 08:11:03	⋮
☐	TEMP `float tEMP;`	⟳	31.993	18 May 2023 08:15:22	⋮

Fig. 3.29 Variable creation

3.11.6 Connecting to a Network

In order to connect to a Wi-Fi network, simply click the “Configure” button in the network section, enter the credentials and click “Save”. In this study “Configure” button in the network section was clicked and Jio Wi-Fi credentials were entered and saved (Fig. 3.30). This information was also generated into sketch file.



Network

Wi-Fi Name: JioFi3_5...

Password: ••••••••••

Configure network

Your will find these network parameters in the secret tab in your sketch, and your device will be able to connect to the network once the sketch will be uploaded.

Wi-Fi Name * JioFi3_504AC1

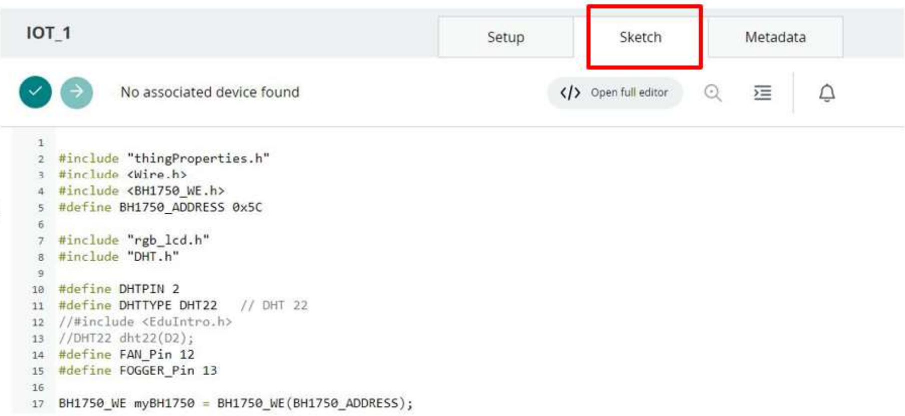
Password ••••••••••

(a) (b)

Fig. 3.30 Connecting to network (a) Enter the Wi-Fi name and password (b) Configure Jio network

3.11.7 Editing the Sketch

When the “Sketch” tab was clicked an automatically generated sketch file was found. It had the same structure as a typical .ino file, but with some additional code to make the connection to network and to the cloud. All editing could be done there (Fig. 3.31). To upload the program to board, "Upload" button was clicked. The editor also had a serial monitor tool, which could be opened by clicking the magnifying glass in the toolbar. Information regarding connection, or commands printed via Serial.print() could be viewed (Fig. 3.32). After successful uploading of code the “Serial Monitor” tab was opened to view information regarding the connection. If it was successful “connected to network_name” and “connected to cloud” were printed. If it failed to connect, it showed errors. Since it was successful in this case, dashboard was created.



```
#include "thingProperties.h"
#include <Wire.h>
#include <BH1750_WE.h>
#define BH1750_ADDRESS 0x5C

#include "rgb_lcd.h"
#include "DHT.h"

#define DHTPIN 2
#define DHTTYPE DHT22   // DHT 22
//#include <EduIntro.h>
//DHT22 dht22(D2);
#define FAN_Pin 12
#define FOGGER_Pin 13

BH1750_WE myBH1750 = BH1750_WE(BH1750_ADDRESS);
```

Fig. 3.31 Editing a sketch in the cloud editor

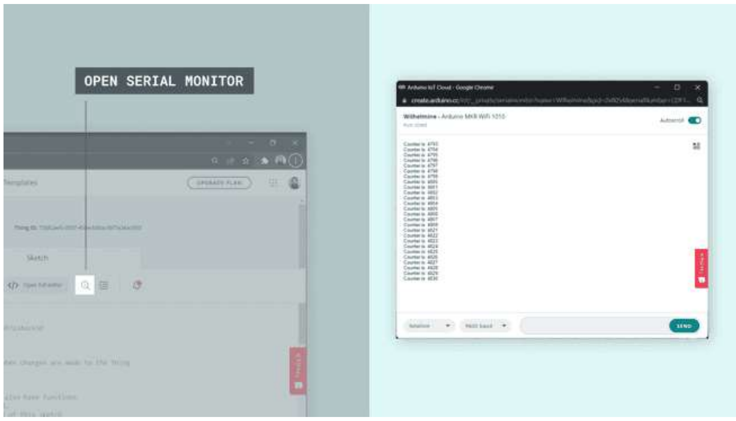
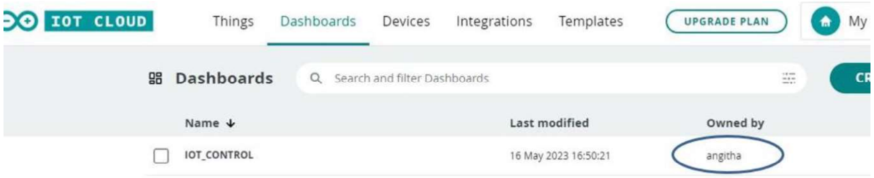


Fig. 3.32 View of serial monitor

3.11.8 Creating a Dashboard

Dashboards are visual user interface for interacting with microcontroller over the cloud. Dashboard was accessed by clicking on the "Dashboards" tab at the top of the Arduino IoT Cloud interface (Fig. 3.33). Dashboard created showed the mode of operation (automatic or manual), status of fan and fogger (ON/OFF) and live, one hour, one day, 7 days and 15 days readings of temperature, RH and light intensity (Fig. 3.34). Both monitoring and controlling was possible by the creation of dashboards.



Name	Last modified	Owned by
☐ IOT_CONTROL	16 May 2023 16:50:21	angitha

Fig. 3.33 Dashboard creation

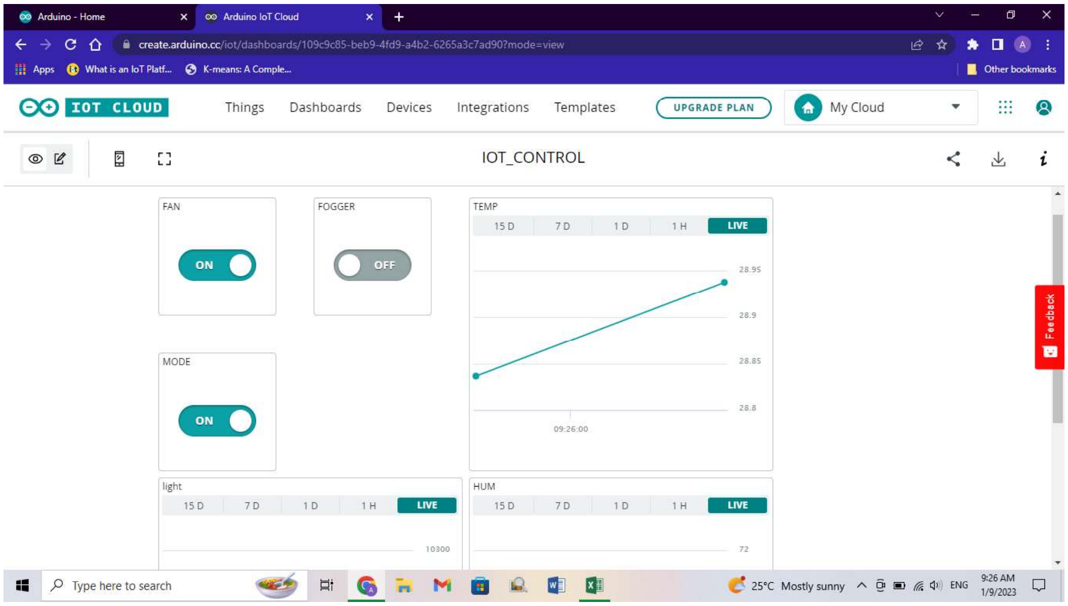
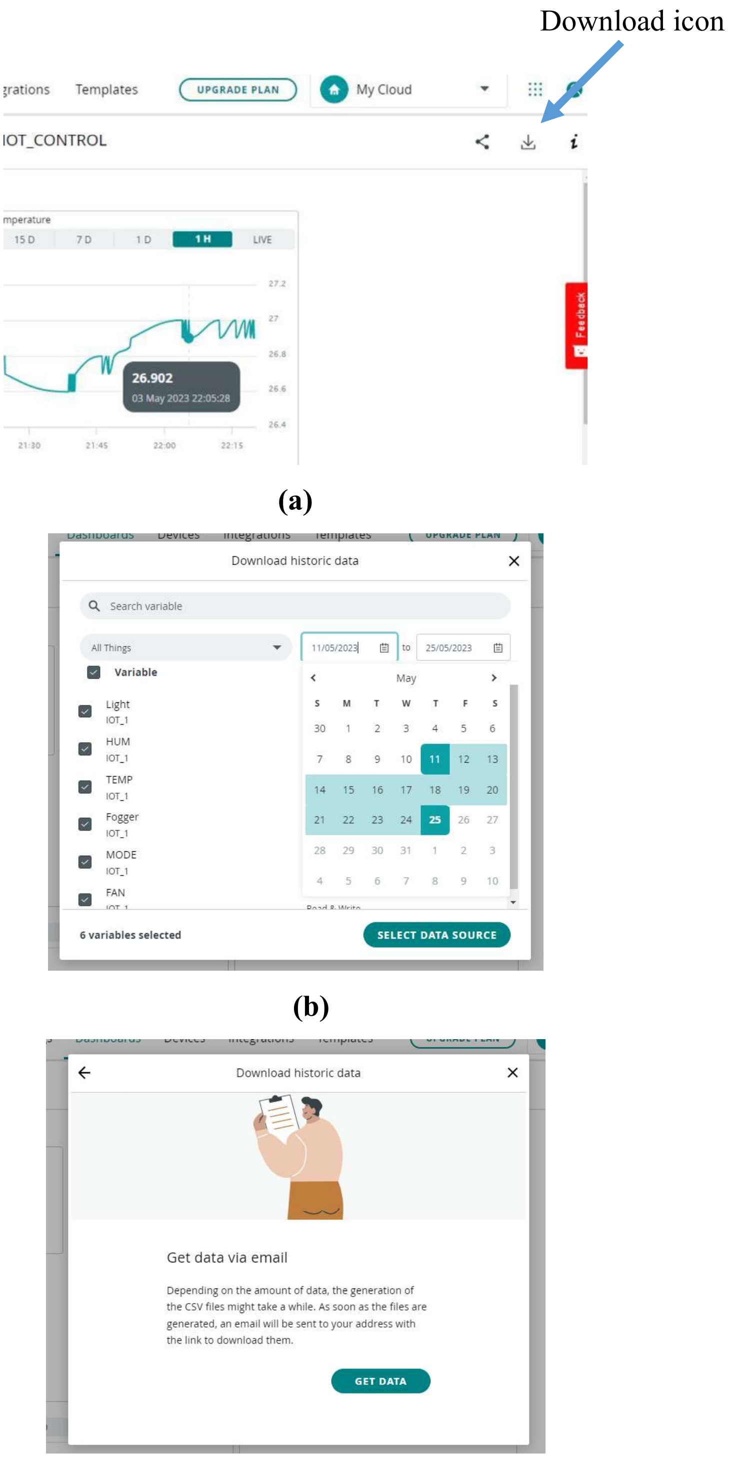


Fig. 3.34 A view of dashboard created

3.11.9 Downloading the Data from IoT Cloud

Data was downloaded from the IoT cloud using the following steps:

1) Click on the download option
2) Select the date
3) Send the data to mail id
4) Obtain data in Zulu time zone
5) Convert data from Zulu time zone to IST by adding 5h and 30 min sing excel (Fig. 3.35).



26.902
03 May 2023 22:05:28

(a)

Download historic data

11/05/2023 to 25/05/2023

☑ Variable

☑ Light IOT_1
☑ HUM IOT_1
☑ TEMP IOT_1
☑ Fogger IOT_1
☑ MODE IOT_1
☑ FAN IOT_1

6 variables selected

(b)

Download historic data

Get data via email

Depending on the amount of data, the generation of the CSV files might take a while. As soon as the files are generated, an email will be sent to your address with the link to download them.

(c)

Fig. 3.35 Downloading the data (a) Click on the download icon (b) Selection of date (c) Sending data via email

3.12 VARIOUS STEPS INVOLVED IN THE DEVELOPMENT AND WORKING OF IoT BASED SYSTEM

- Calibration of sensors
- Networking all the components with Arduino Nano 33 IoT
- Program the microcontroller using Arduino IDE software using C language for data storing process, system monitoring and controlling process
- Microclimate data acquisition using sensors (temperature, RH and light) interfaced on Arduino Nano 33 IoT microcontroller
- Information displayed on LCD screen after data processing
- Decision making and component actuation based on available information
- Polyhouse microclimate and control system status were displayed using Arduino IoT cloud
- Live monitoring and controlling from anywhere using Arduino IoT cloud and mobile application were made
- Plotted the sensor information as graphical insights in Arduino IoT cloud
- Also transmitted the results in a simplex form of communication by sending SMS to a mobile phone with the aid of a GSM module.
- Data was downloaded from Arduino IoT cloud.

3.13 FLOWCHART OF IoT BASED AUTOMATION SYSTEM

An overall workflow of the IoT based automation system is shown in Fig. 3.36 and the working of temperature and RH control is shown in Fig. 3.37. Arduino Nano 33 IoT and all other components were initialized. Calibration of sensor was done after the initialization. Sensor read the value and send it to the user. The values displayed in both mobile phone and PC. After uploading suitable code the actuators worked accordingly. Exhaust fans and foggers were the actuators used. The IoT based system checked the temperature and when it was above 28°C exhaust fans automatically switched ON by means of SSR and when the temperature reached below 25°C the exhaust fans automatically turned OFF. The fogger operation was fixed at an interval of 15 minutes and operating time 1 minute. Whenever the temperature was above 35°C fogger operated for 1 min. If the temperature not reached 32°C after one operation, the next operation was allowed only after 15 minutes. Whenever the temperature reached 32°C in between the operation of fogger it stops automatically before the completion. Otherwise, the next operation of fogger starts only after 15 min and the process continues.

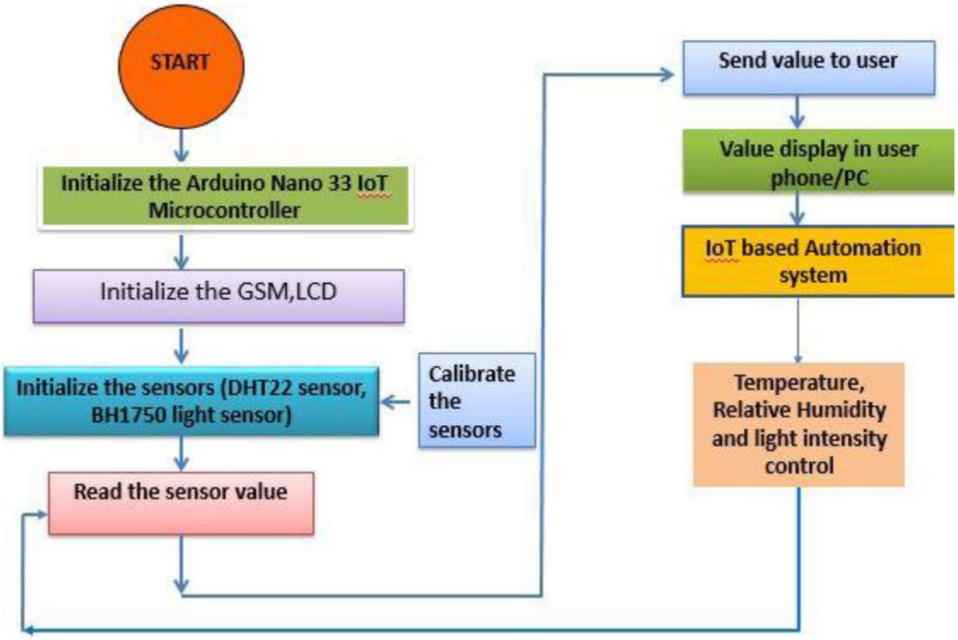


Fig. 3.36 Microcontroller programme workflow: Algorithm

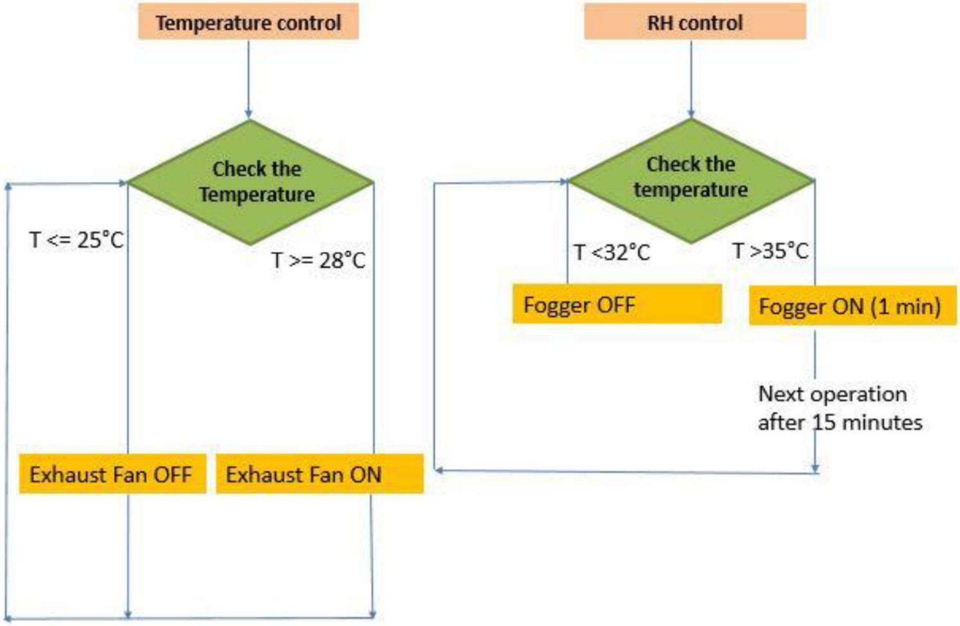


Fig. 3.37 Working of temperature and relative humidity control

3.14 TESTING OF DEVELOPED SYSTEM INSIDE THE POLYHOUSE WITHOUT CROP

Developed system was tested inside the polyhouse without crop. The observations taken were temperature, relative humidity and light intensity. Temperature and relative humidity were measured inside and outside the polyhouse using hygrometer on a daily basis. Light intensity was measured using digital luxmeter. At the same instant sensor readings were noted from the IoT based automation system. The instrument measured readings were compared with sensor readings of the IoT based automation system. Testing was done in three steps as follows.

1) Validation of sensor reading with measured reading was done for temperature, RH and light intensity inside the polyhouse
2) Comparison of temperature, RH and light intensity inside and outside the polyhouse after the installation of IoT system inside the polyhouse.
3) Checking the ability of the system for monitoring and controlling of temperature and RH inside the polyhouse.

3.15 EVALUATION OF THE SYSTEM INSIDE THE POLYHOUSE WITH CROP

The developed system inside the polyhouse was evaluated by growing a test crop. The experiment was conducted inside the naturally ventilated polyhouse during December 2022 – May 2023. The duration of the crop was four months (120 days). The polyhouse was oriented in east–west direction with an area of 208 m^2 (26 m x 8 m).

3.15.1 Details of Test Crop

‘Anjitha’ variety of bhindi released by KAU was cultivated inside the polyhouse to evaluate the performance of the IoT based automation system for real-time monitoring and controlling of microclimate. ‘Anjitha’ variety is high yielding, early maturing and resistant to Yellow Vein Mosaic (YVM).

3.15.2 Field Preparation Inside the Polyhouse

Land preparation was done inside the naturally ventilated polyhouse. Field was thoroughly tilled using a power tiller and was formed into beds and channels. Seven raised beds of 24 m length, 0.4 m width and 0.15 m height were made for cultivation of bhindi crop. Dolomite was applied prior to one week before planting for raising pH and to supply plants with calcium and magnesium needed for healthy growth. Various preparations are shown in Plate 3.13.

(a) **(b)** **(c)**

Plate 3.13 Field preparation (a) Tillage operation (b) Plant bed preparation (c) Dolomite application inside the polyhouse

3.15.3 Irrigation System

Water source was an open well from which water was pumped to an overhead tank and conveyed through the main line of 63 mm diameter PVC pipes. PVC sub main of 50 mm diameter was connected to the main line to which, low density polyethylene laterals of 16 mm diameter were connected. End caps were provided at the end of laterals. Along the laterals, online drippers of 4 l/hr were fixed. Details of drip irrigation system installed inside the polyhouse is given in Table 3.25.

Table 3.25 Details of drip Irrigation system installed inside the polyhouse

Particulars	Details
Size of the plot	26 m ×8 m
Area	5 cent
Diameter of main line	63 mm
Diameter sub main	50 mm
Diameter of lateral	16 mm
Length of each lateral from sub main	25 m
Total no.of laterals from sub main	7 nos.
Number of emitters per lateral	40 nos (Approx.)
Lateral spacing	70 cm
Emitter spacing	60 cm
Emitter type	Online
Emitter discharge rate	4 lph
No. of plants per row	40 nos.
Spacing of plants	60 × 70 cm

3.15.4 Nursery Preparation

Seeds were sown in pro trays containing mixture of coirpith, sand, perlite and cowdung in the ratio 1:1:1:2 (Plate 3.14 a, b).

(a) (b)

Plate 3.14 Pro-tray preparation and sowing

3.15.5 Transplantation of Seedlings

Seedlings were transplanted 3 weeks after germination in the polyhouse. A total of 255 plants were planted on 20/01/2023 at 70x60 cm spacing. A view of transplantation of seedlings is shown in Plate 3.15.

Plate 3.15 Transplantation of seedlings

3.15.6 Irrigation Schedule of Bhindi

Emitter discharge was 4 l/hr. Water was applied in different amount during different stages of the crop (Pavithran and Krishnan, 2018). Different crop growth stages of bhindi, water requirement and amount of water applied per plant per stage are shown in Table 3.26.

Table 3.26 Irrigation scheduling followed

Sl. No	Crop growth stage	Water requirement (l/day/plant)	Amount of water applied per plant per stage (l)
1.	Early stage (21/01/23 to 11/02/23)	0.6	13.2
2.	Mid stage (12/02/23 to 28/02/23)	1	17
3.	Late stage (01/03/23 to 21/05/23)	2	164
		Total water applied per plant	193.2

3.15.7 Fertilizer Application Schedule

Trichoderma coir pith compost, cow dung, bonemeal, jeevamrutham, urea, factomphose and potash were applied 15 Days After Planting (DAP) at different days. Fish amino acid, NPK 19:19:19 and sampoorna were applied as foliar and urea was applied 45 DAP. SOP was sprayed and hume, which is a biofertiliser was applied 45 DAP. Details of fertilizer applied with its dosage as per Package of Practices (POP) recommendation of KAU, 2016 are shown in Table 3.27.

Table 3.27 Fertilizer application schedule followed as per POP recommendation of KAU

Days After Planting (DAP)	Fertiliser recommendation applied for 255 plants inside the polyhouse
15 DAP (05/02/2023)	➢ Trichoderma coir pith compost (1.23 kg) ➢ Cow dung (10.2 kg) ➢ Bonemeal (5 kg) ➢ Jeevamrutham (50 l) ➢ Urea (1 kg) + Factomphose (3.5 kg) + Potash (2.5 kg)
30 DAP (20/02/2023)	➢ Fish Amino Acid (25 ml/5 l) ➢ NPK 19:19:19 (15 g/5l) ➢ Urea 2.37 kg ➢ Sampoorna (25 g/5l)
45 DAP (07/03/2023)	➢ Sulphate of Potash (SOP) (100 g/10 l) ➢ Hume-Biofertiliser (100 ml/10 l)

3.15.8 Cultural Practices

Various cultural practices followed during crop growth were:

- Hand weeding
- Earthing up
- Plant protection measures

The nutrient status and soil moisture status were maintained as per the Package of Practices (POP) recommendation of KAU, 2016.

3.15.9 Dates of Planting and Harvest

Details of date of planting and harvest is shown in Table 3.28.

Table 3.28 Details about dates of planting and harvest

Date of sowing	31-12-2022
Date of planting	20-01-2023
Age of seedling	21 Days
Date of first flowering	13-02-2023
First harvest of bhindi	24-02-2023
Final harvest of bhindi	21-05-2023

3.16 OBSERVATIONS ON CROP GROWTH AND YIELD PARAMETERS

Ten plants were selected randomly and tagged for observations.

3.16.1 Growth Parameters

Various growth parameters *viz.* plant height, number of leaves per plant, leaf length, leaf width, stem girth and number of primary branches per plant of the crop inside the polyhouse were noted at 15, 30, 45, 60, 75, 90, 105, 120 days after planting.

3.16.2 Yield Parameters

Harvesting was started 35 days after planting and continued regularly at alternate days. The fruit weight, fruit length, fruit girth, number of fruits per plant, yield per plant and yield per ha were noted from the 10 tagged plants for each harvest and average was taken.

RESULTS AND DISCUSSION

CHAPTER-IV

RESULTS AND DISCUSSION

An IoT based real-time microclimate monitoring and controlling system for polyhouse was developed and tested. In this study, the various microclimate parameters *viz.* temperature, relative humidity, and light intensity were monitored and controlled. The system was tested with and without crop inside the polyhouse. The biometric observations related to growth and yield parameters were recorded. The results pertaining to the development and testing of IoT based real-time microclimate monitoring and controlling system for polyhouse are discussed in this chapter under the following subheads.

4.1 CALIBRATION OF SENSORS

DHT22 temperature & humidity sensor and BH1750 light sensor were the two sensors used in this study. These two sensors were calibrated using the data acquired from the sensor and the measured data using instruments. A calibration curve was plotted in excel to get the calibration equation. This calibration equation was used in the Arduino code for accurate measurement by sensor. The results of calibration of sensors are as follows.

4.1.1 Results of Calibration of DHT22 Sensor for Temperature

Temperature reading from DHT22 sensor and drybulb thermometer reading were taken at the same time. The values recorded are shown in Table 4.1. The plot of dry bulb temperature reading *vs.* DHT22 sensor reading (calibration curve) is shown in Fig. 4.1 and a calibration equation was obtained (Eq. 4.1).

$$y = 1.0182x - 2.4222 \qquad \text{…Eq. 4.1}$$

Table 4.1 DHT22 sensor reading and drybulb temperature reading

Time	DHT22 sensor reading (°C)	Drybulb temperature reading (°C)
10.00 AM	30.5	28
10.30 AM	33.6	30
11.00 AM	39.4	38
11.30 AM	37.6	34
12.00 PM	35.6	35
12.30 PM	36.4	35.2
1.00 PM	36.5	35.1
1.30 PM	34.4	33.5
2.00 PM	34.3	32
2.30 PM	33.8	32.2
3.00 PM	32.9	31.5
3.30 PM	32.2	31
4.00 PM	31.8	30.2

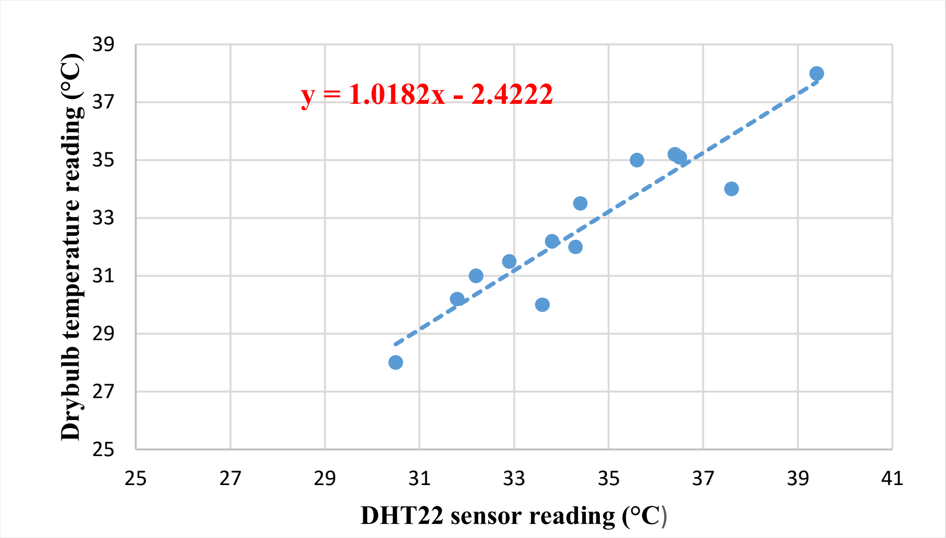


Fig. 4.1 DHT22 sensor temperature calibration curve

4.1.2 Results of Calibration of DHT22 Sensor for Relative Humidity (RH)

The RH reading from DHT22 sensor and hygrometer RH readings were taken at the same time. The values are recorded in Table 4.2. A plot of hygrometer RH reading *vs.* DHT22 sensor reading (calibration curve) was plotted (Fig. 4.2) and a calibration equation was obtained (Eq. 4.2).

$$y = 0.7203x + 12.872 \qquad \ldots\text{Eq. 4.2}$$

Table 4.2 DHT22 sensor reading and hygrometer RH reading

Time	**DHT22 sensor reading (%)**	**Hygrometer RH (%)**
10.00 AM	64.1	62.4
10.30 AM	52.2	56
11.00 AM	62.3	45
11.30 AM	52.6	54
12.00 PM	57.4	52
12.30 PM	58.4	51
1.00 PM	57.5	52.1
1.30 PM	63.5	57.55
2.00 PM	62.7	61
2.30 PM	65.9	62.9
3.00 PM	68.2	60.5
3.30 PM	69.7	69
4.00 PM	72.6	65.2

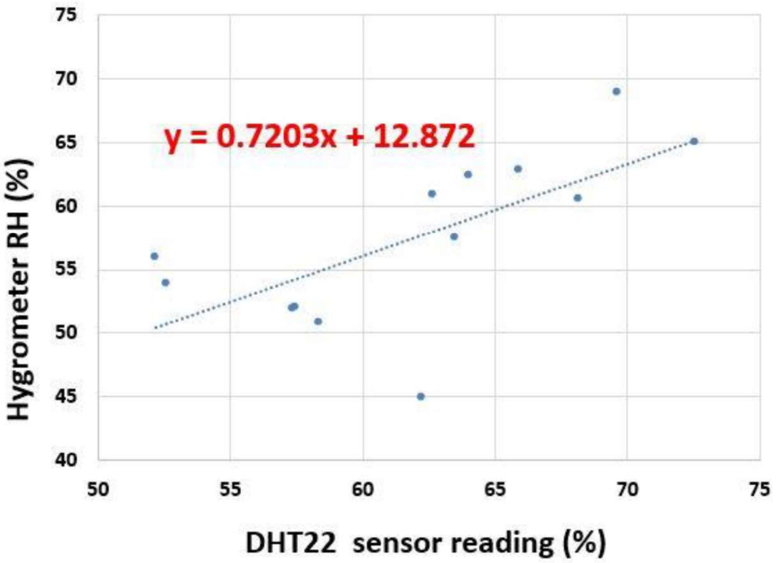


Fig. 4.2 DHT22 sensor Relative Humidity calibration curve

4.1.3 Result of Calibration of BH1750 Light Sensor for Light Intensity

Light intensity reading from BH1750 light sensor and luxmeter reading were taken at the same time which is shown in Table 4.3. Luxmeter reading *vs.* BH1750 light sensor reading (calibration curve) was plotted (Fig. 4.3) and a calibration equation was obtained (Eq. 4.3).

$$y = 0.9999x + 3.193 \qquad \text{…Eq. 4.3}$$

Table 4.3 BH1750 sensor reading and luxmeter reading

Time	BH1750 sensor reading (lx)	Luxmeter reading (lx)
8.00 AM	14099.17	14100
9.00 AM	10183.33	10185
11.00 AM	9703.33	9705
12.00 PM	8102.5	8110
1.00 PM	7495.83	7500
2.00 PM	5300	5300
3.00 PM	4401.67	4405
4.00 PM	3195.83	3200
5.00 PM	2669.17	2670

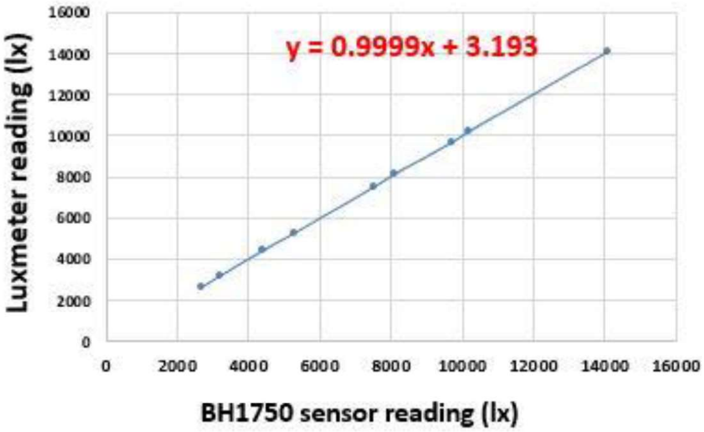
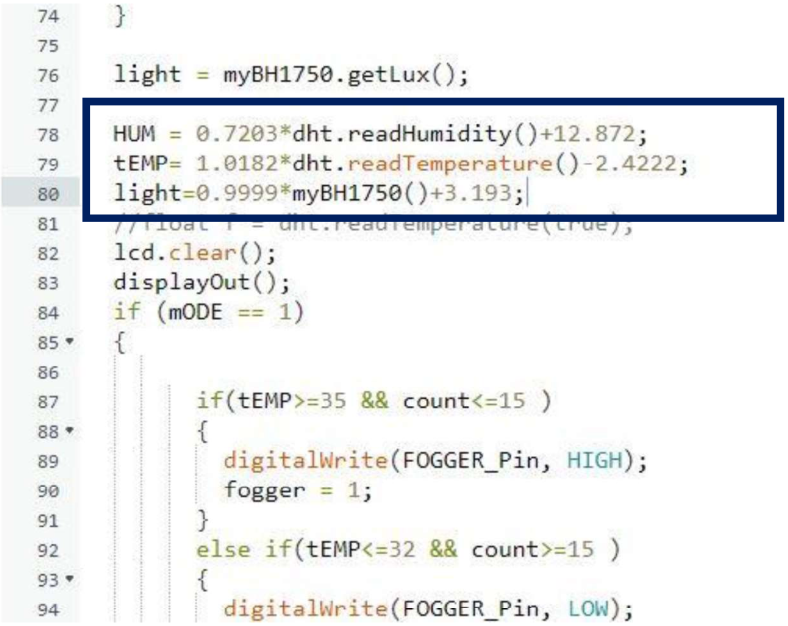


Fig. 4.3 BH1750 light sensor calibration curve

4.1.4 Inclusion of Calibration Equation in Arduino Code

Calibration equation was obtained for DHT22 temperature & humidity sensor and BH1750 light sensor. The calibration equations for temperature, RH and light intensity obtained were Eq. 4.1, Eq. 4.2 and Eq. 4.3 respectively. These equations were incorporated in the Arduino code as shown in Fig. 4.4.

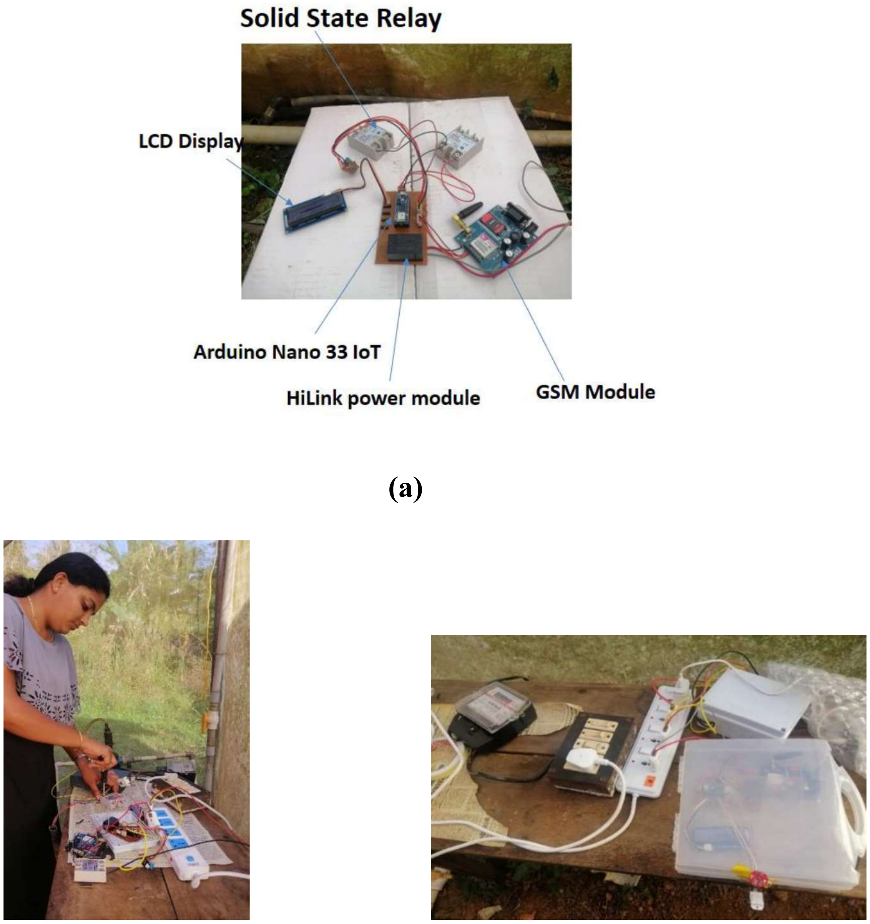


```
}

light = myBH1750.getLux();

HUM = 0.7203*dht.readHumidity()+12.872;
tEMP= 1.0182*dht.readTemperature()-2.4222;
light=0.9999*myBH1750()+3.193;
//float f = dht.readTemperature(true);
lcd.clear();
displayOut();
if (mODE == 1)
{

      if(tEMP>=35 && count<=15 )
      {
        digitalWrite(FOGGER_Pin, HIGH);
        fogger = 1;
      }
      else if(tEMP<=32 && count>=15 )
      {
        digitalWrite(FOGGER_Pin, LOW);
```

Fig. 4.4 Incorporation of calibration equations in Arduino code

4.2 INSTALLATION OF DEVELOPED IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM INSIDE THE POLYHOUSE

After successful calibration of sensors, all the hardware and software components were connected as per the circuit diagram. A view of installed system with components identified is shown in Fig. 4.5.



(a)

(b) **(c)**

Fig. 4.5 Different views of IoT based real-time microclimate monitoring and controlling system (a) Components (b) Assembling components (c) Installed system

4.3 TESTING AND EVALUATION OF DEVELOPED SYSTEM INSIDE THE POLYHOUSE WITHOUT CROP

Developed IoT based real-time microclimate monitoring and controlling system was tested inside the polyhouse without crop. Testing was done in three steps as follows.

1) Validation of sensor reading with measured reading was done for temperature, RH and light intensity inside the polyhouse
2) Comparison of temperature, RH and light intensity inside and outside the polyhouse after the installation of IoT system inside the polyhouse.
3) Checking the ability of the system for monitoring and controlling of temperature and RH inside the polyhouse

4.3.1 Validation of Sensor Reading with Measured Reading for Temperature, Relative Humidity and Light Intensity

Microclimate parameters *viz.* temperature, RH and light intensity were recorded using both instrument and sensor. Readings were taken from 8.00 AM to 5.00 PM at one hour interval for a period of one week from 09/01/2023 to 15/01/2023. Sensor reading and instrument reading were compared and deviation percentage was calculated. The values of temperature, RH & light intensity measured and deviation percentage calculated for a day (09/01/2023) are shown in Table 4.4, Table 4.5 and Table 4.6 respectively. Values for other days are shown in appendix (Table A-1 to Table A-18).

Table 4.4 Validation of sensor reading with measured reading for temperature on 09/01/2023

Time	Observed temperature (°C)	Sensor reading	Difference	Deviation %
8:00 AM	24	23.74	-0.26	-0.01
9:00 AM	28	27.1	-0.9	-0.03
10:00 AM	30	30.66	0.66	0.02
11:00 AM	32	32.8	0.8	0.02
12:00 AM	34	33.41	-0.59	-0.01
1:00 PM	34	34	0	0
2:00 PM	33.5	33.5	0	0
3:00 PM	35	35	0	0
4:00 PM	34	34	0	0
5:00 PM	31	31	0	0

Table 4.5 Validation of sensor reading with measured reading for Relative Humidity on 09/01/2023

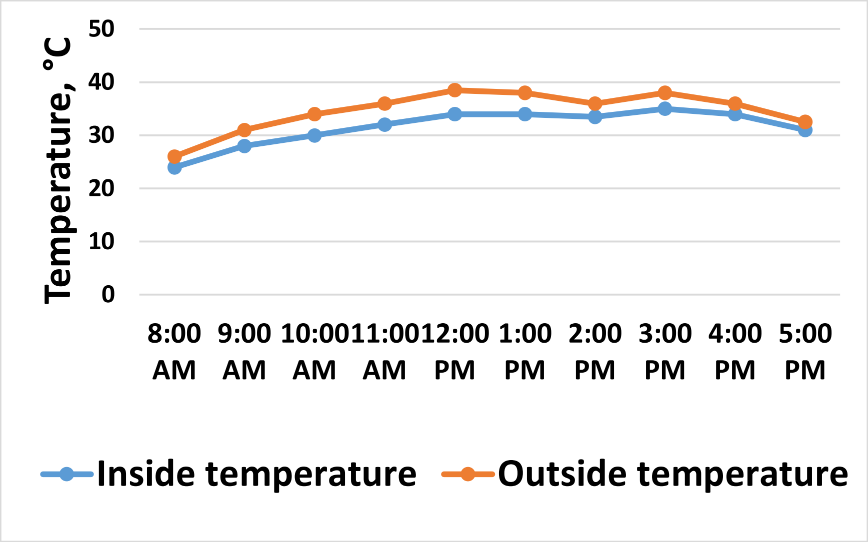
Time	Observed RH (%)	Sensor reading	Difference	Deviation %
8:00 AM	79	80.86	-1.86	0.02
9:00 AM	74	74.52	-0.52	0.007
10:00 AM	68	67.75	0.25	-0.003
11:00 AM	61	60.12	0.88	-0.01
12:00 AM	59	58.82	0.18	-0.003
1:00 PM	54	51	3	-0.05
2:00 PM	50.5	50.5	0	0
3:00 PM	47	47	0	0
4:00 PM	51	51	0	0
5:00 PM	63	63	0	0

Table 4.6 Validation of sensor reading with measured reading for light intensity on 09/01/2023

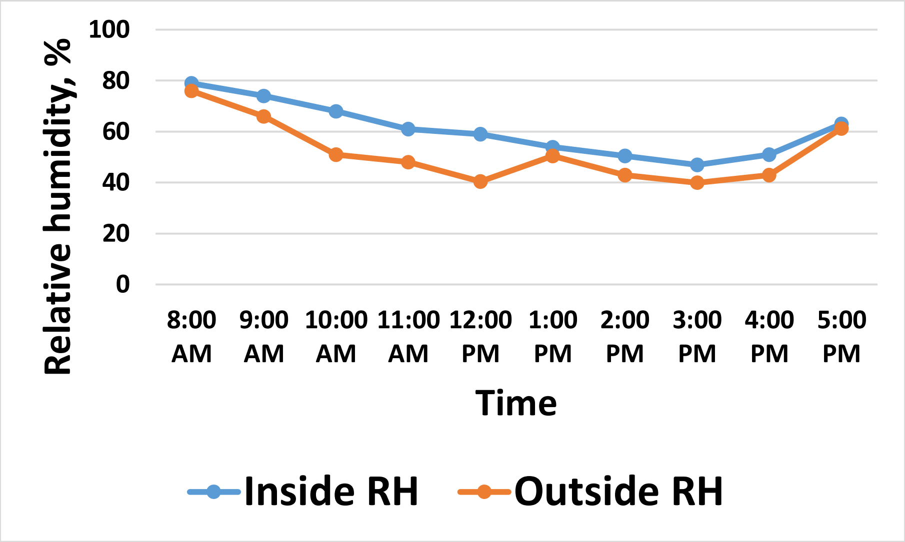
Time	Observed light intensity (lx)	Sensor reading	Difference	Deviation %
8:00 AM	4116	4115.83	0.17	$-5.4X10^{-5}$
9:00 AM	14500	14500	0	0
10:00 AM	20950	20955	-5	$3X10^{-4}$
11:00 AM	30650	30650	0	0
12:00 AM	33960	33960	0	0
1:00 PM	33530	33528.33	1.67	$-5X10^{-5}$
2:00 PM	16993.33	16994	-0.67	$3.94X10^{-5}$
3:00 PM	29840	29840	0	0
4:00 PM	19526	19526	0	0
5:00 PM	5603	5602.5	0.5	$-1X10^{-4}$

4.3.2 Comparison of Temperature, RH & Light Intensity Inside and Outside the Polyhouse After the Installation of Iot Based Microclimate Monitoring and Controlling System

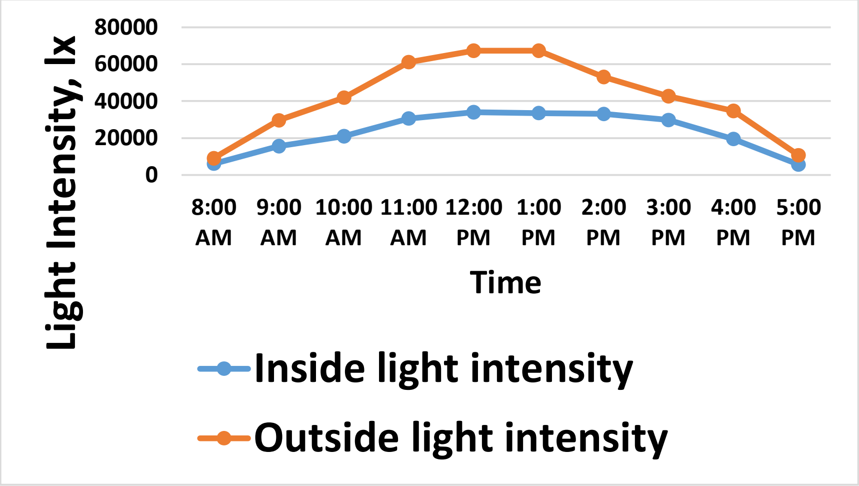
Usually temperature inside the polyhouse is higher and RH is less compared to outside of the polyhouse. Hence after the installation of the system, microclimate parameters temperature, RH and light intensity both inside and outside polyhouse were recorded using instruments, drybulb & wetbulb hygrometer and luxmeter for a period of one week from 09/01/23 to 15/01/23. After the installation of IoT based automation system it was found that there was lower temperature and higher RH inside the polyhouse than outside and are shown in Fig. 4.6 to Fig. 4.12. Hence, it can be concluded that there was a decrease of temperature and increase of RH inside the polyhouse. This might be due to the proper operation of actuators (foggers and exhaut fans). The real-time monitoring and control of microclimate parameters resulted in better microclimate control inside the polyhouse with this system than the time-based automation system developed by Jinu and Hakkim (2019) in which exhaust fans and foggers were used to control temperature and RH.


Temperature, °C
50
40
30
20
10
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Inside temperature
Outside temperature

(a)

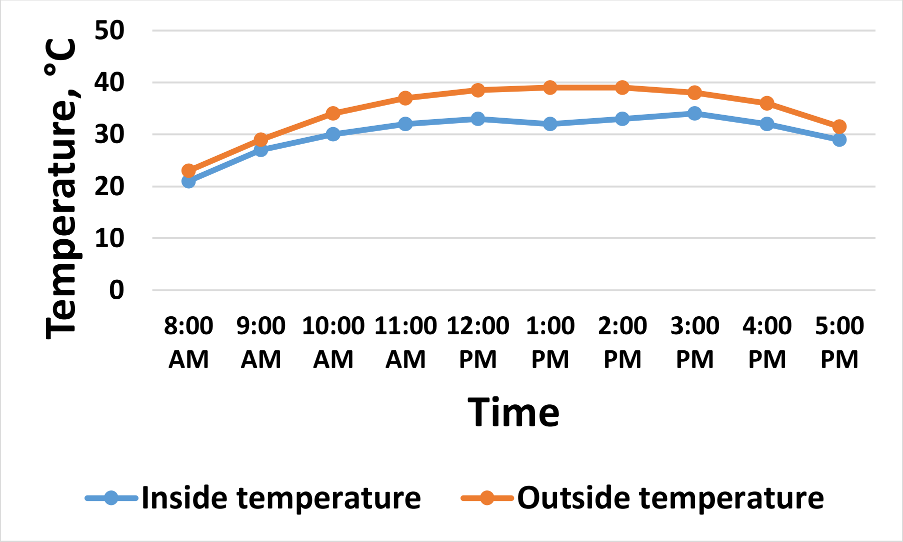

Relative humidity, %
100
80
60
40
20
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside RH
Outside RH

(b)

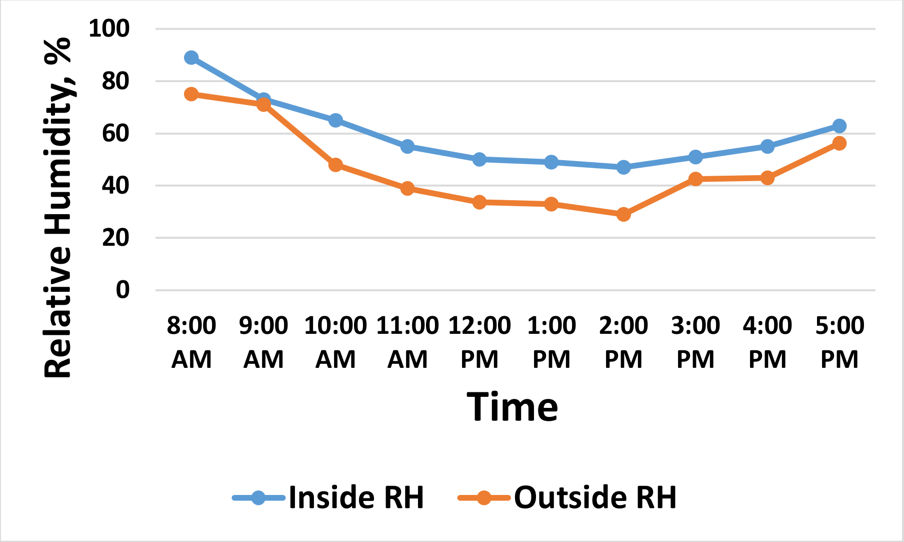


(c)

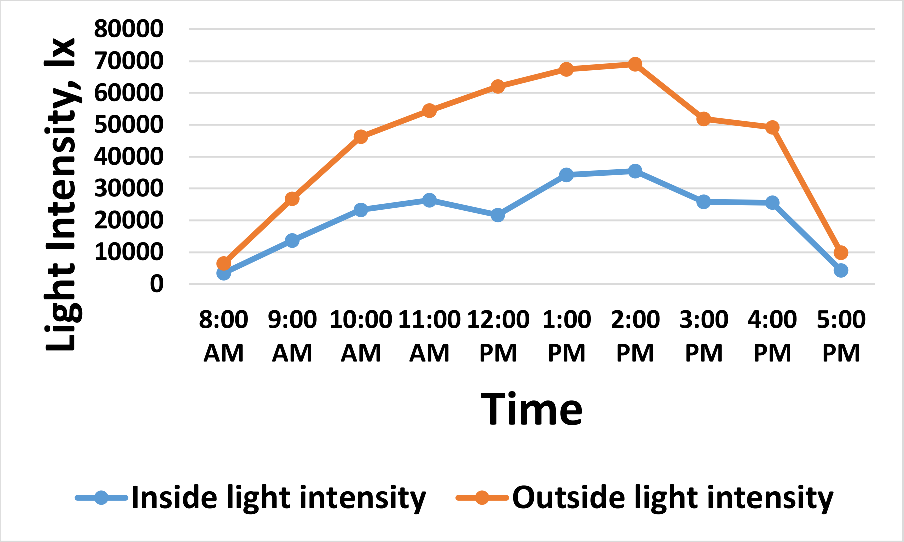
Fig. 4.6 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 09/01/2023 (a) Temperature (b) RH (c) Light intensity



(a)

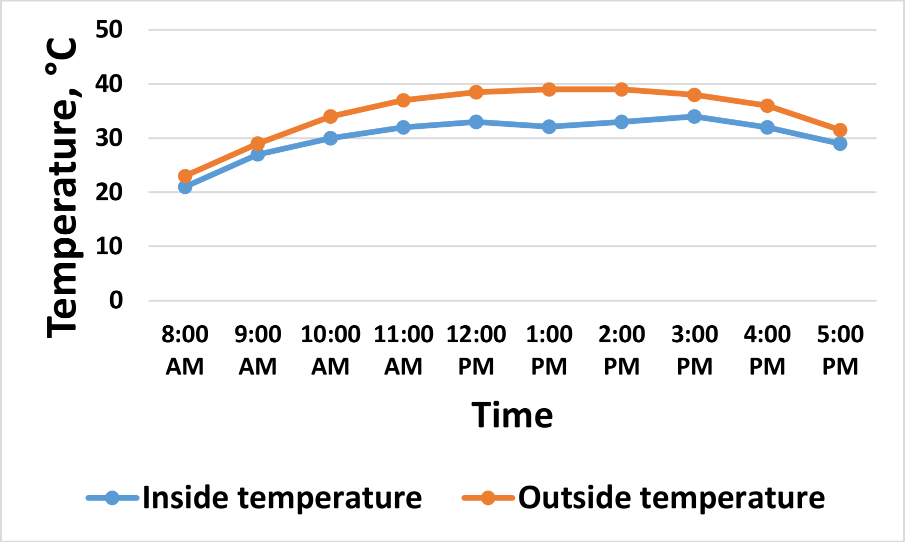


(b)

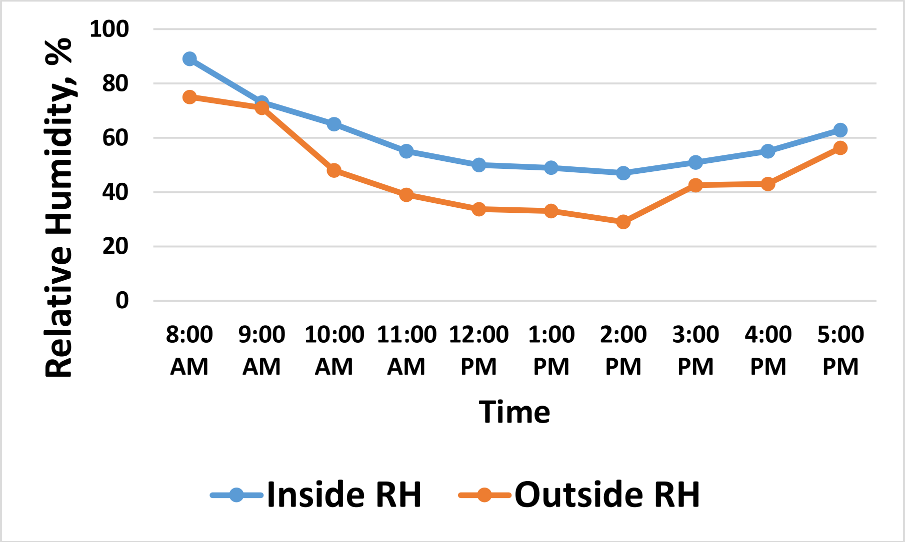


(c)

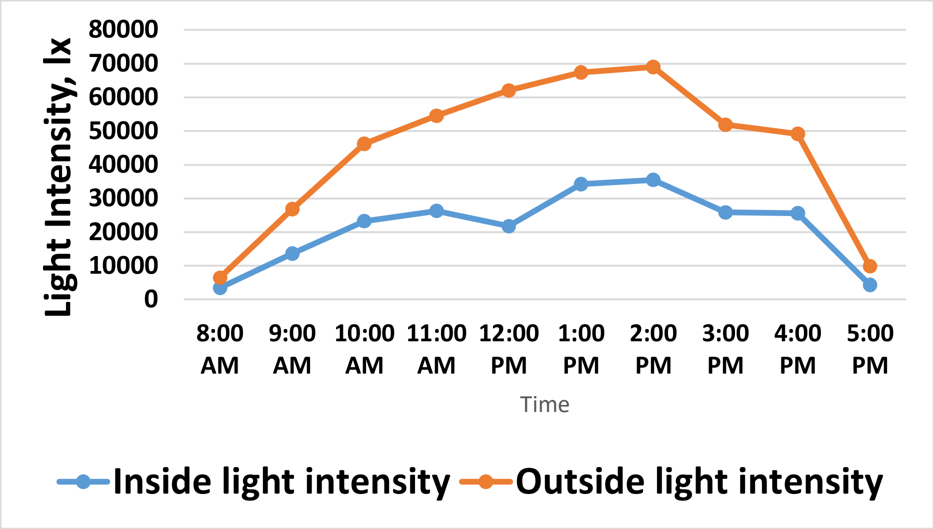
Fig. 4.7 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 10/01/2023 (a) Temperature (b) RH (c) Light intensity


Temperature, °C
50
40
30
20
10
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside temperature
Outside temperature

(a)

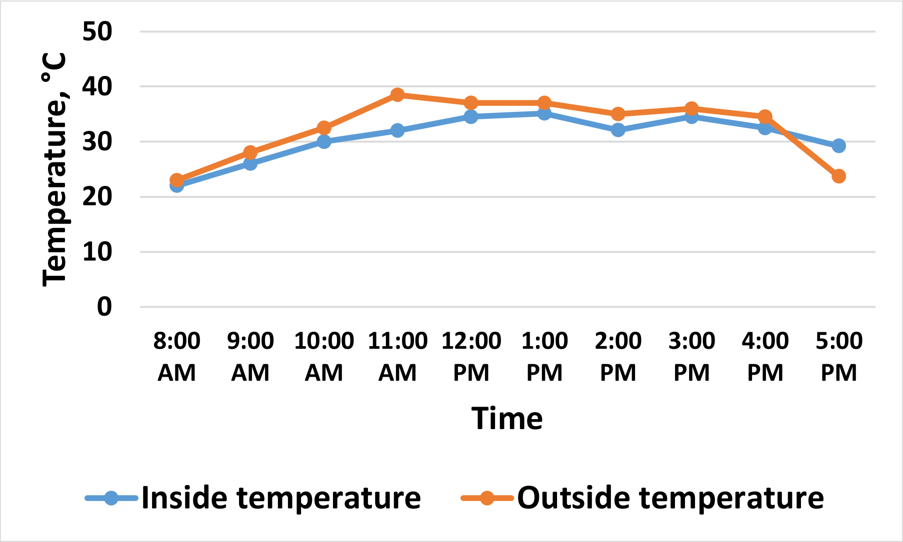

Relative Humidity, %
100
80
60
40
20
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside RH
Outside RH

(b)

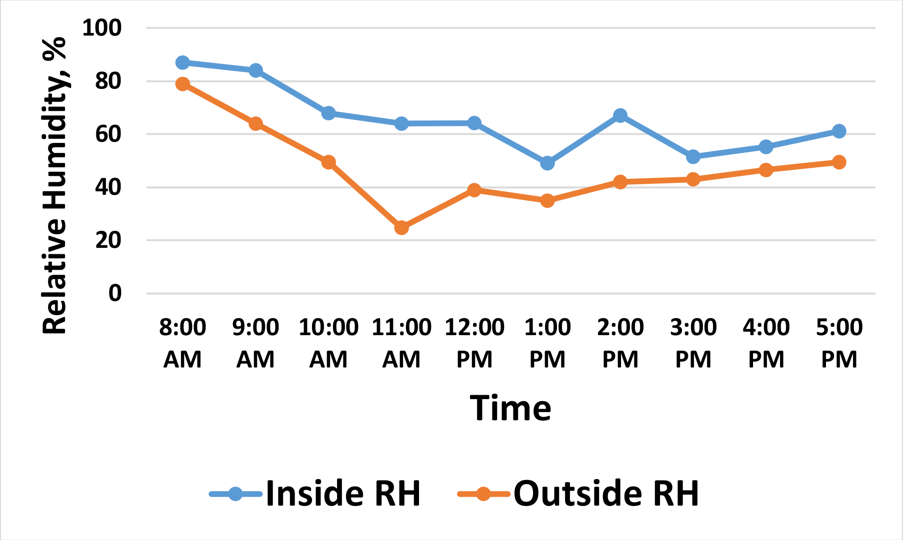


(c)

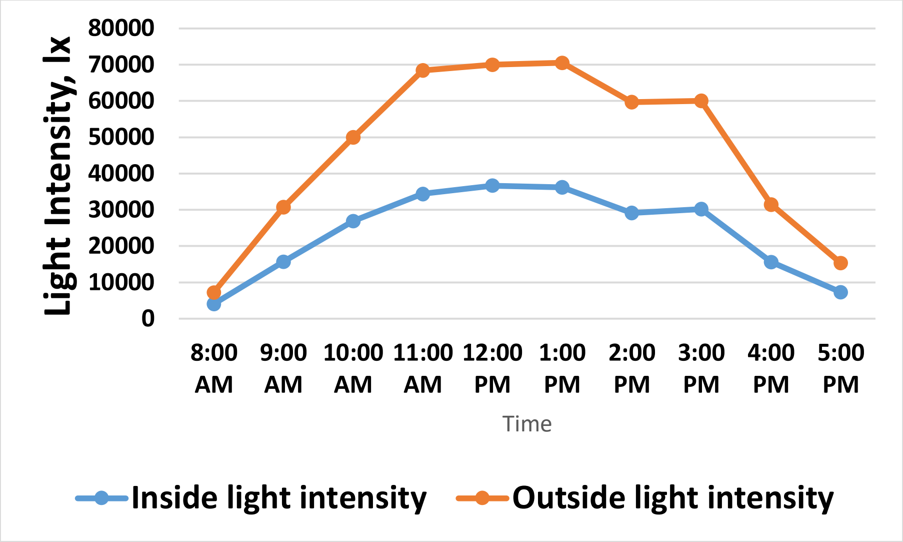
Fig. 4.8 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 11/01/2023 (a) Temperature (b) RH (c) Light intensity



(a)

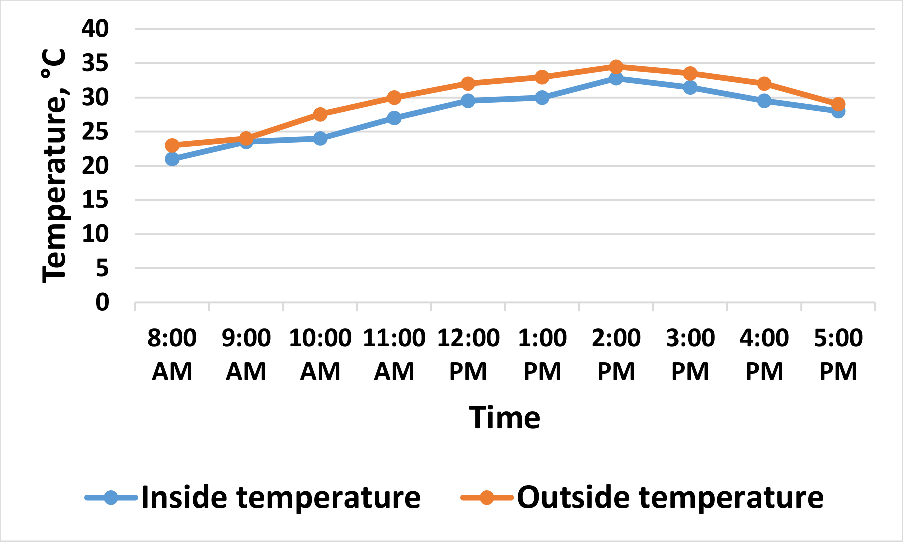


(b)

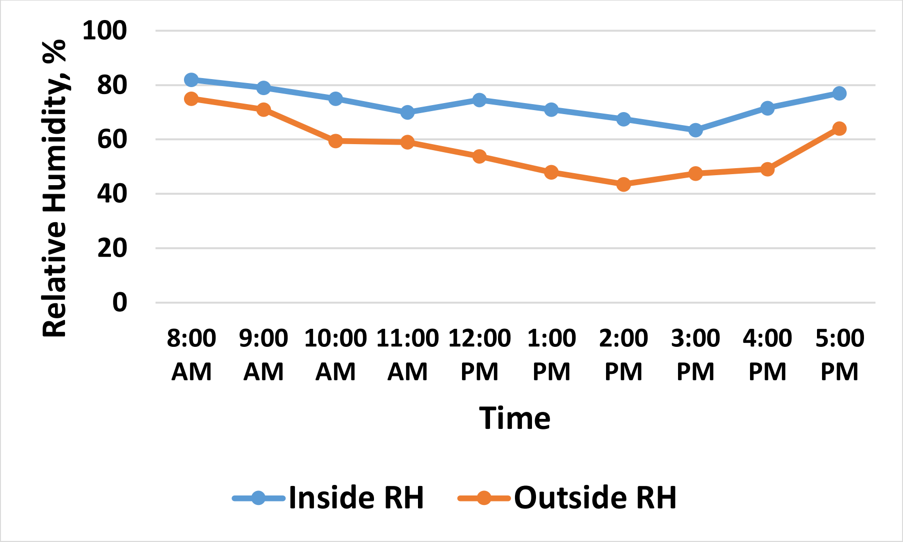


(c)

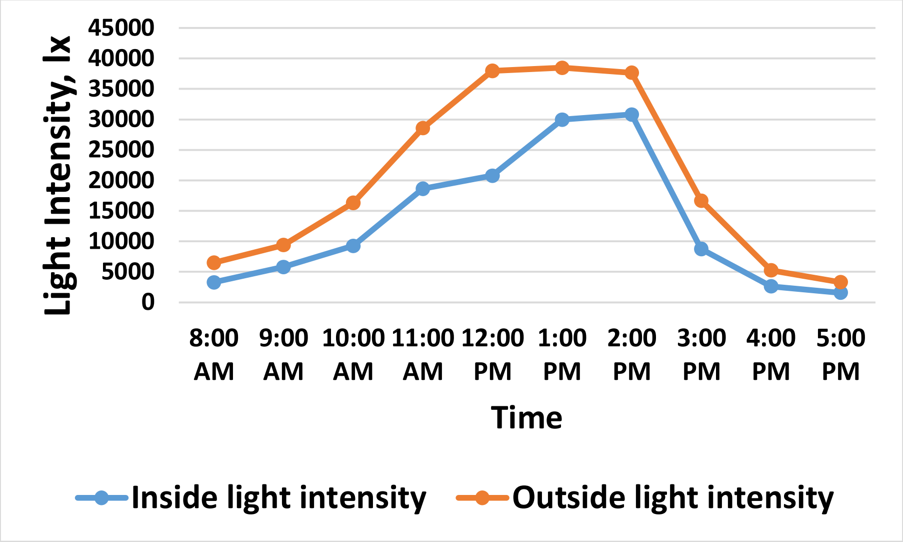
Fig. 4.9 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 12/01/2023 (a) Temperature (b) RH (c) Light intensity


Temperature, °C
40
35
30
25
20
15
10
5
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside temperature
Outside temperature

(a)

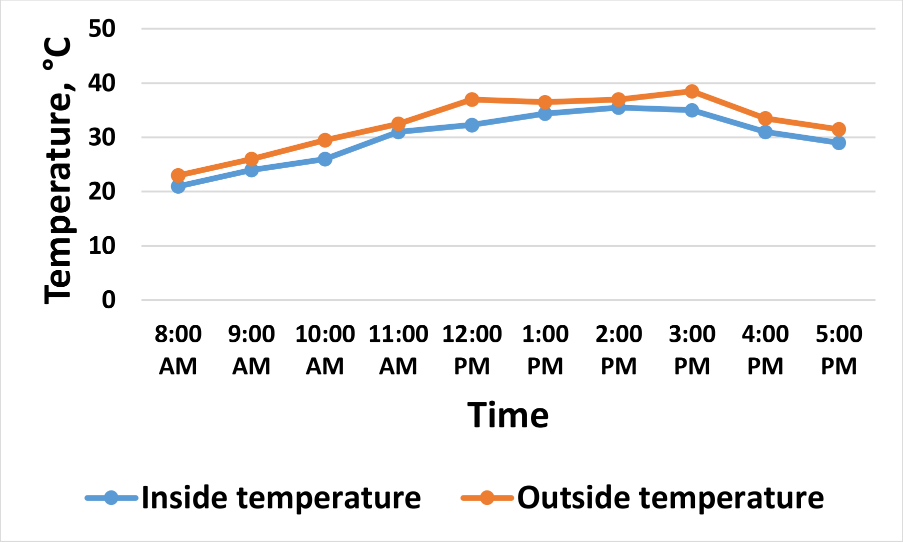

Relative Humidity, %
100
80
60
40
20
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside RH
Outside RH

(b)

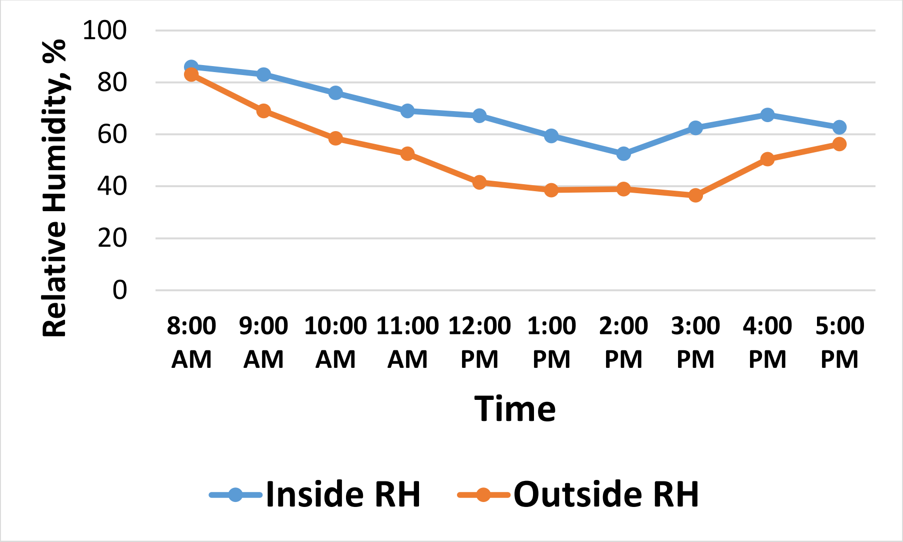


(c)

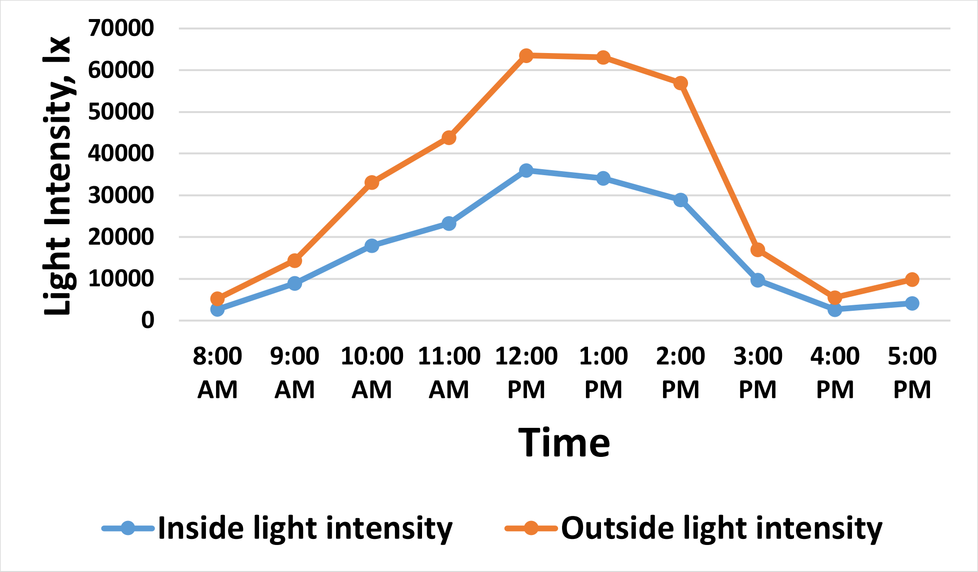
Fig. 4.10 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 13/01/2023 (a) Temperature (b) RH (c) Light intensity



(a)

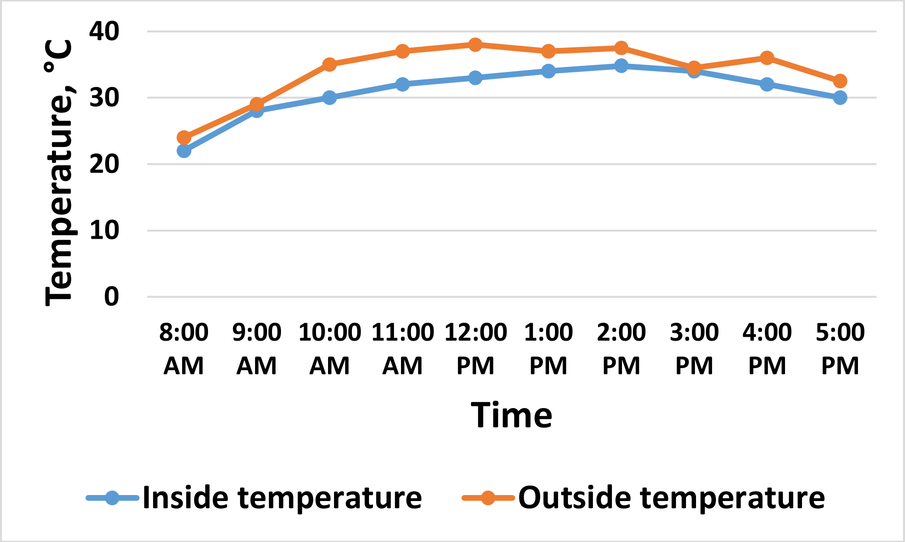


(b)



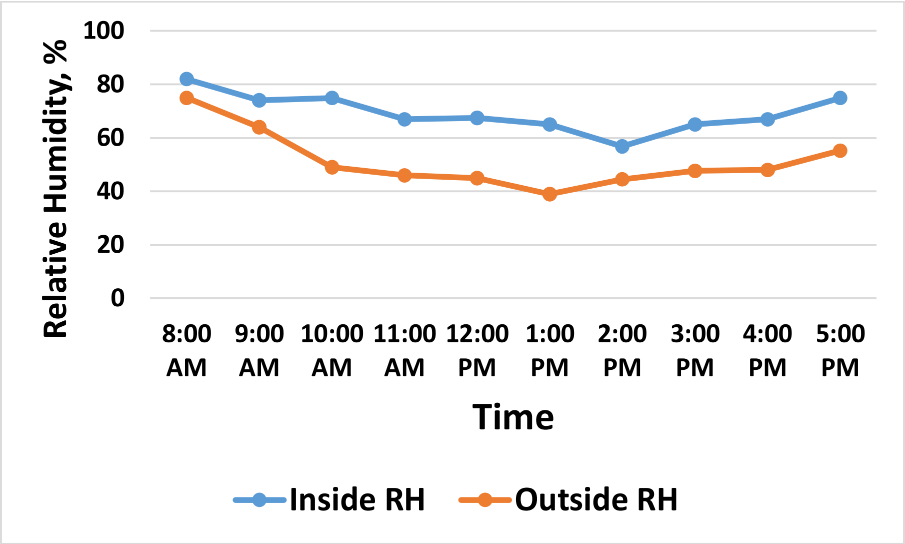
(c)

Fig. 4.11 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 14/01/2023 (a) Temperature (b) RH (c) Light intensity



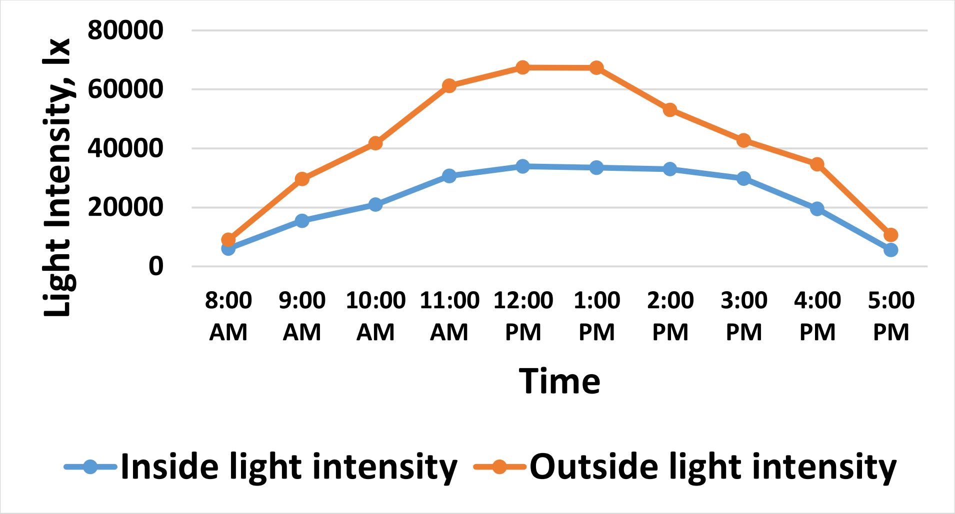
Temperature, °C
40
30
20
10
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside temperature
Outside temperature

(a)



Relative Humidity, %
100
80
60
40
20
0
8:00 AM
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
3:00 PM
4:00 PM
5:00 PM
Time
Inside RH
Outside RH

(b)



(c)

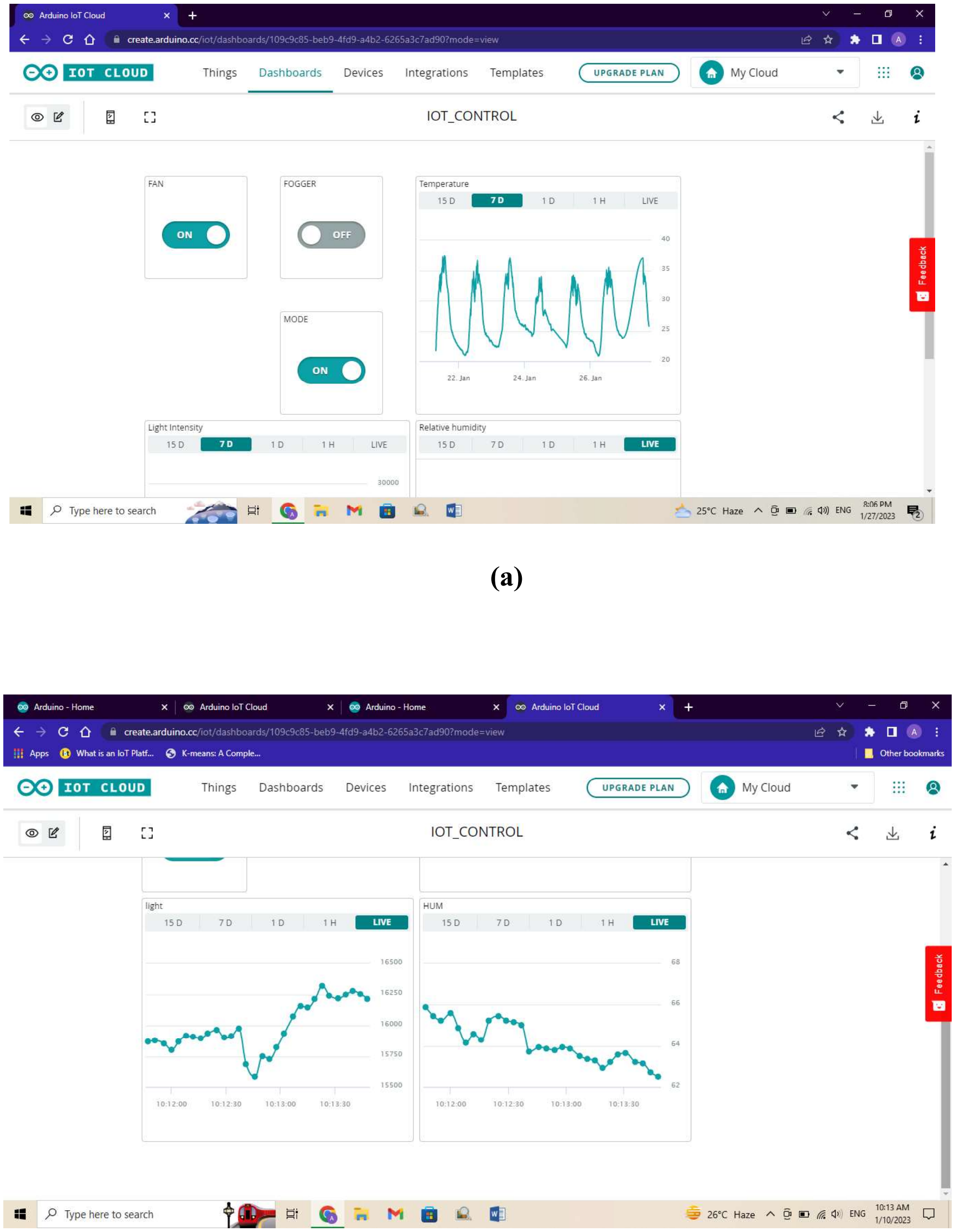
Fig. 4.12 Comparison of microclimate parameters inside (with IoT system) and outside the polyhouse on 15/01/2023 (a) Temperature (b) RH (c) Light intensity

4.3.3 Checking the Ability of the Iot System to Monitor and Control Temperature, Relative Humidity (RH) and Light Intensity

The ability of the developed IoT system to monitor and control temperature, RH and light intensity were checked and the results are explained under the following subheads.

4.3.3.1 Monitoring Live Data from Different Locations

The IoT based sytem was able to monitor temperature, RH and light intensity from different locations. Arduino IoT cloud showed the mode of operation (automatic or manual), status of fan & fogger (ON/OFF), and real-time values of temperature, RH and light intensity. Besides the continuous real-time data monitoring, it also showed past one hour, one day, seven days and 15 days interval temperature, RH and light intensity data as graphical insights. Fig. 4.13 shows the live monitoring of temperature, RH and light intensity recorded inside the polyhouse (location 1). Similarly live monitoring from ladies hostel (location 2), academic block (location 3) and from my house at Thrissur (location 4) were shown in Fig. 4.14, Fig. 4.15 and Plate 4.1 respectively.



(a)

(b)

Fig. 4.13 Monitoring live data in PC from Arduino IoT cloud inside the polyhouse on 27/01/2023 - location 1

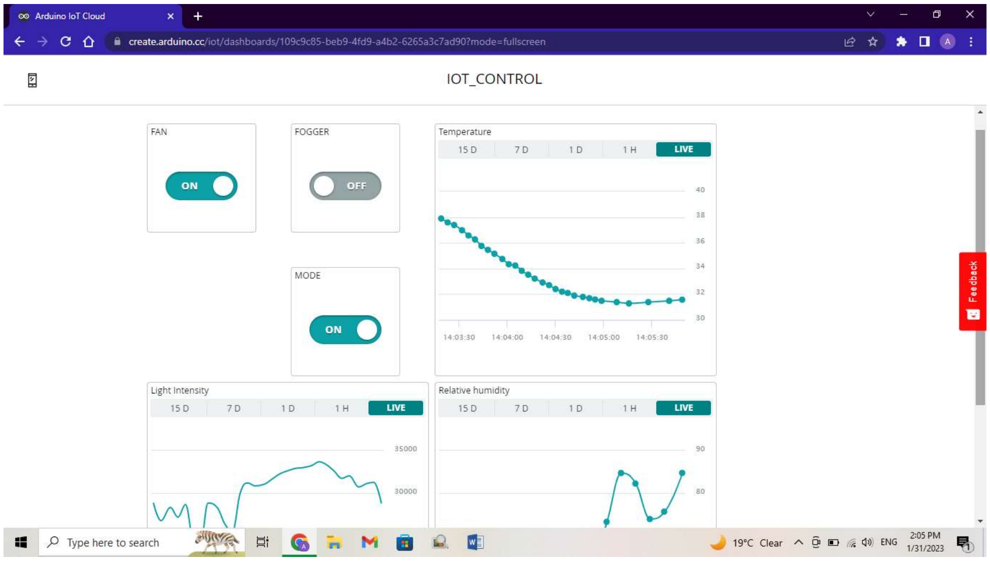


Fig. 4.14 Live monitoring on 31/01/2023 as seen from location 2

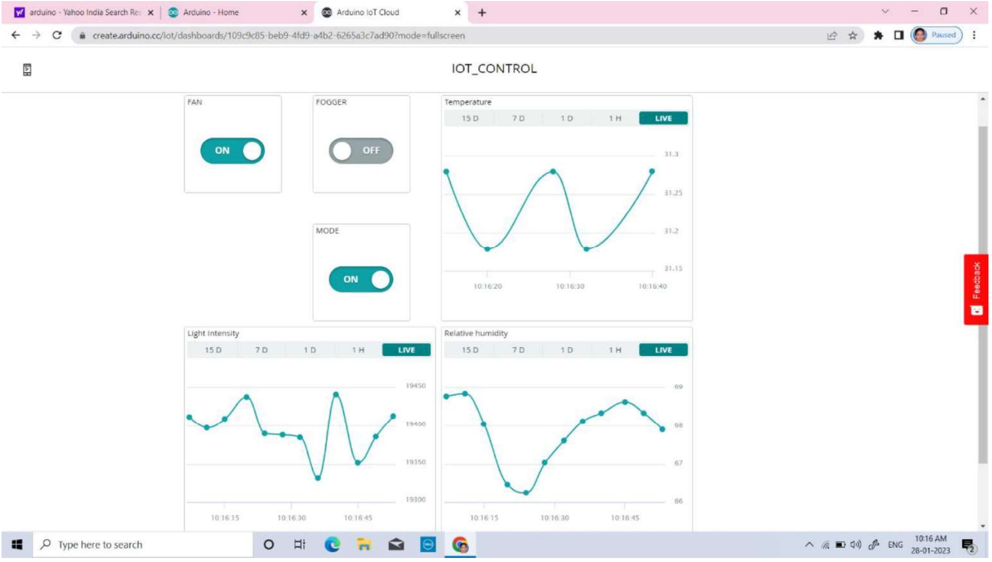


Fig. 4.15 Live monitoring on 28/01/2023- location 3

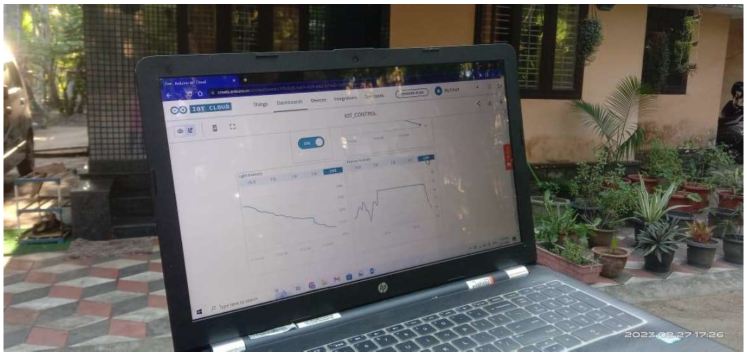
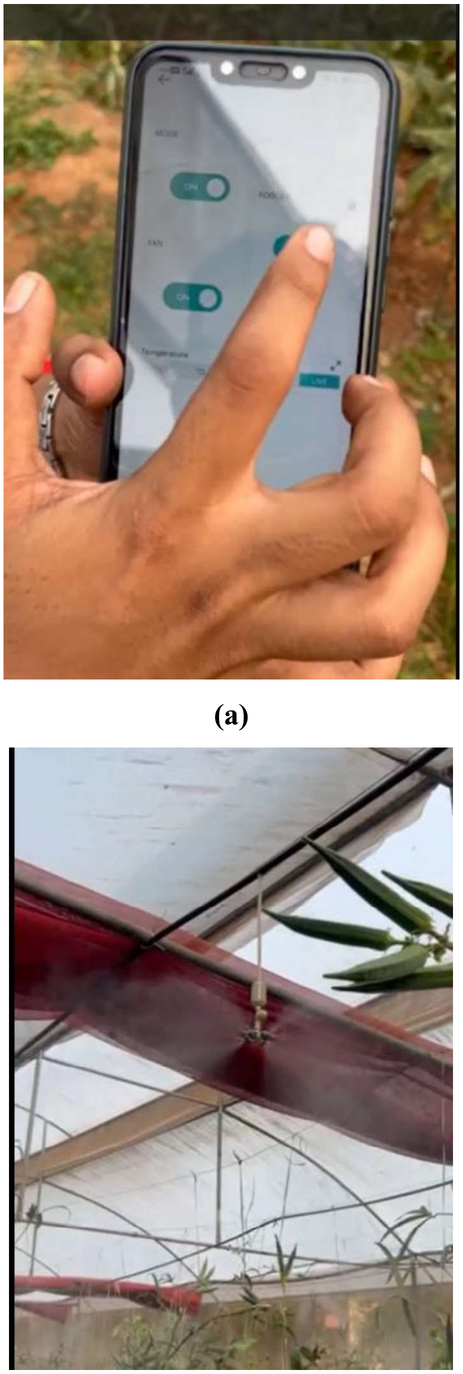


Plate 4.1 Live monitoring from Thrissur on 27/02/2023- location 4

4.3.3.2 Working of Fogger and Exhaust Fans using Iot Remote Mobile Application

Exhaust fans and foggers were the actuators used to control temperature and Relative Humidity (RH). Whenever the temperature inside the polyhouse exceeded 28°C, the controller switched 'ON' the exhaust fans, reduced the temperature, and turned 'OFF' when the temperature reached below 25°C. It was also found that the actuator 'exhaust fan' alone was not able to reduce the temperature up to the desired level, hence fogger was also connected to the system to maintain temperature as well as RH. Hence, it was found that when the temperature exceeded 35°C, fogger automatically switched 'ON' and switched 'OFF' when the temperature reached below 32°C.

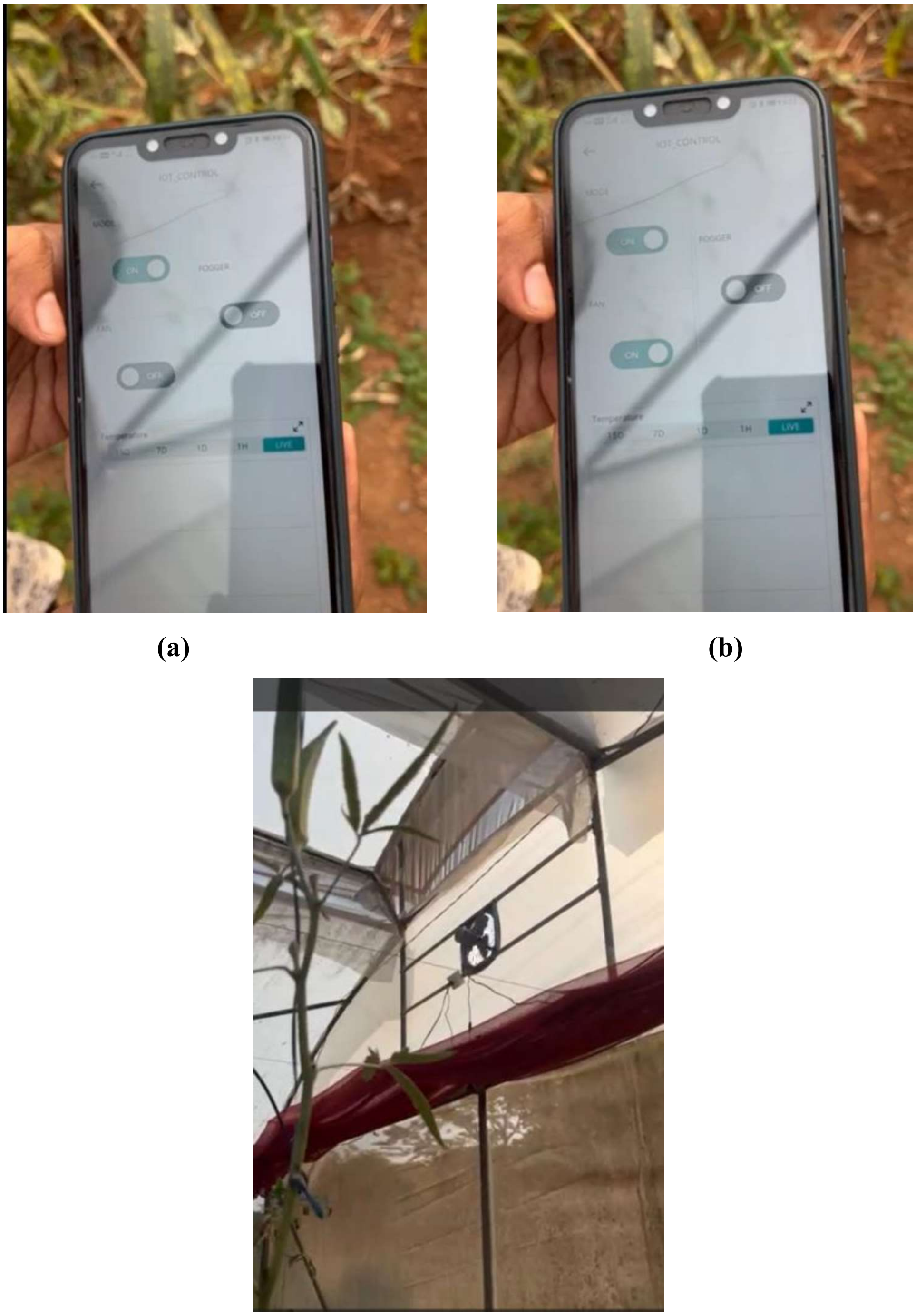
Live monitoring and controlling was also made possible through the IoT cloud remote mobile application. It showed the mode, condition of fan and fogger, and real-time value of temperature, humidity and light intensity. Fig. 4.16 (a) shows the switch to 'ON' and 'OFF' of fogger. When the fogger switch was put ON the fogger automatically operated. Operation of fogger is shown in Fig. 4.16 (b). Similarly when the mode was put in automatic mode as shown in Fig. 4.17 (a), the sensor automatically sensed the temperature and switched 'ON' the exhaust fans when the temperature was above 28°C as shown in Fig. 4.17 (b). The operation of exhaust fans is shown in Fig. 4.17 (c). The whole operation was recorded as video and was presented during the defence seminar. Incorporation of video is beyond the scope of this thesis writing.



(a)

(b)

Fig. 4.16 Different views of working of fogger (a) Switch 'ON' fogger using IoT remote (b) Working of fogger



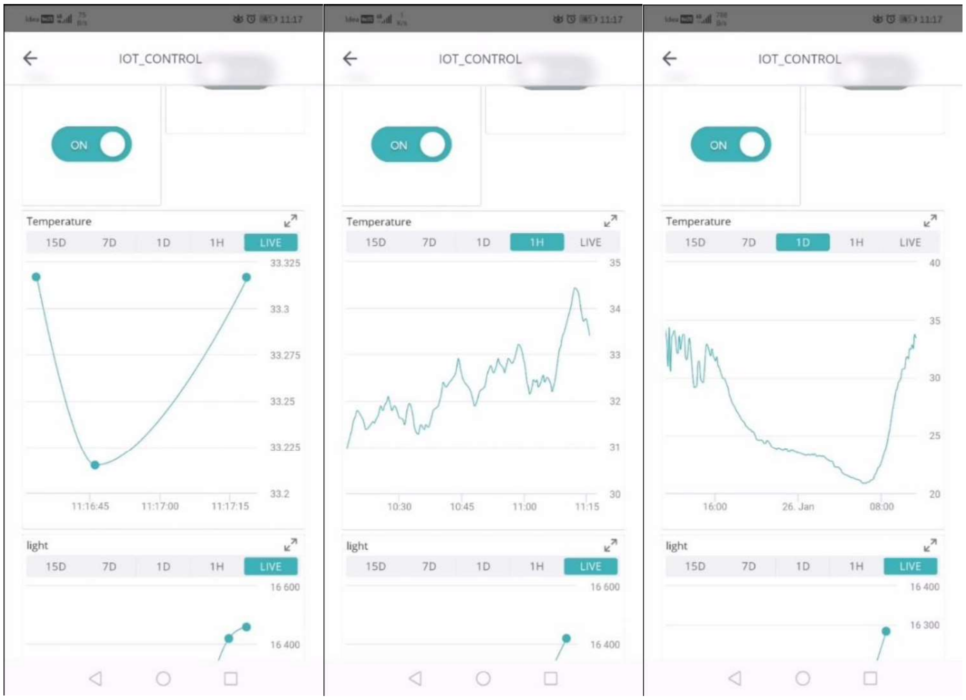
(a) (b)

(c)

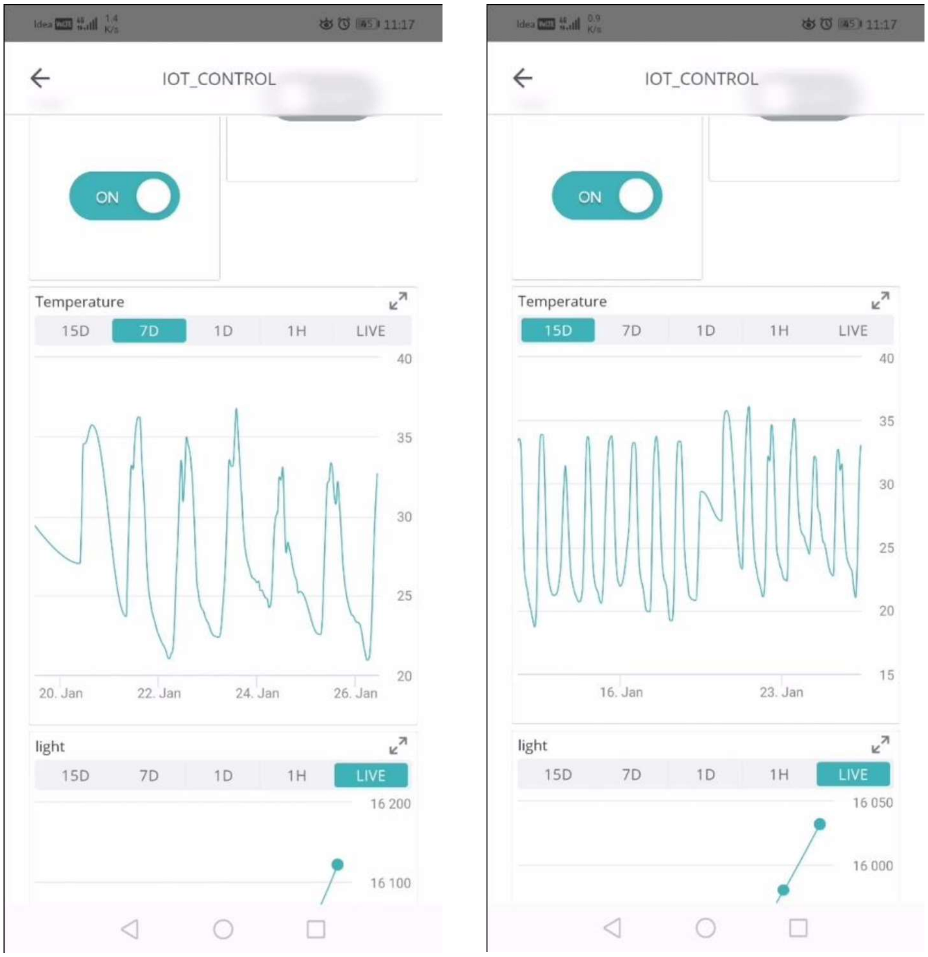
Fig. 4.17 Different views of working of exhaust fans (a) Automatic mode (b) Exhaust fan automatically switched 'ON' (c) Working of exhaust fans

4.3.3.3 Monitoring and Controlling using IoT Cloud Remote Mobile Application

Real-time microclimate data acquisition and its live monitoring and controlling was also made possible using IoT remote mobile application. Besides the continuous real-time data monitoring, it also showed past one hour, one day, seven days and 15 days interval temperature, RH and light intensity data as graphical insights. Screenshots of live monitoring of temperature, RH and light intensity at different time intervals on 30/01/2023 for demonstration purpose are shown in Fig. 4.18, Fig. 4.19 and Fig. 4.20 respectively.

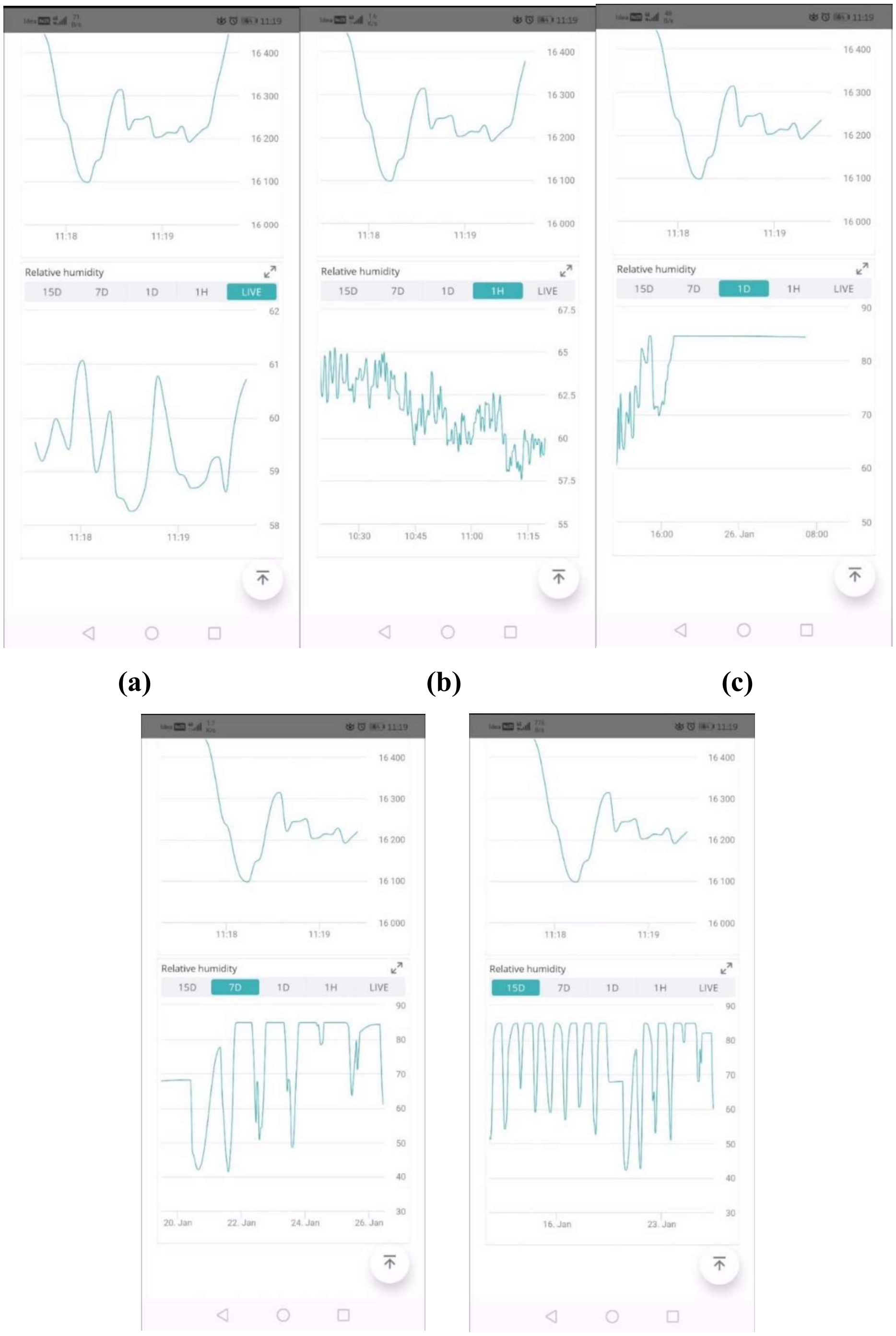


(a) **(b)** **(c)**



(d) **(e)**

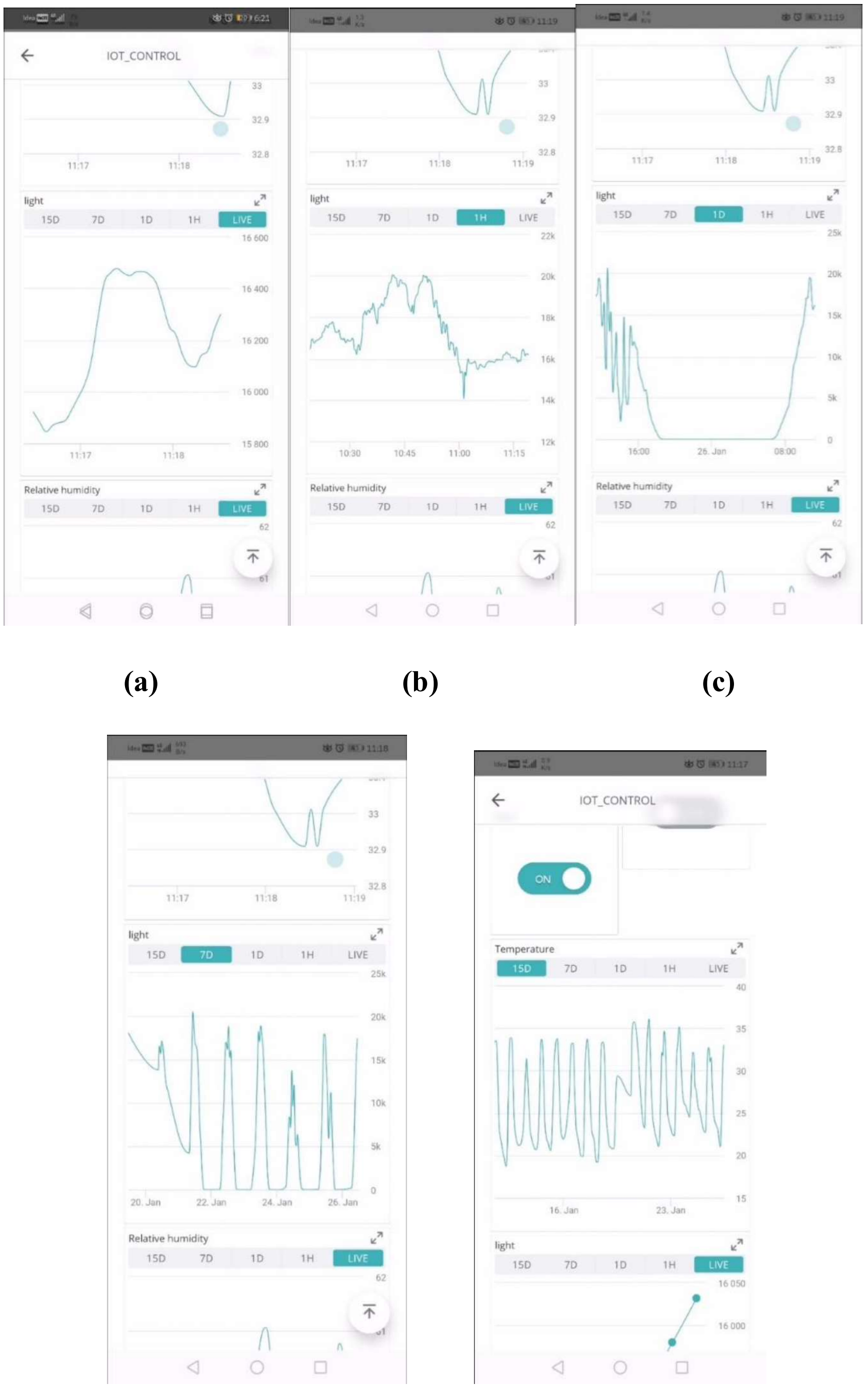
Fig. 4.18 Live monitoring of temperature using IoT remote mobile application (a) Live reading (b) one hour reading (c) one day reading (d) 7 days reading (e) 15 days reading



(a) (b) (c)

(d) (e)

Fig. 4.19 Live monitoring of RH using IoT remote mobile application (a) Live reading (b) one hour reading (c) one day reading (d) 7 days reading (e) 15 days reading



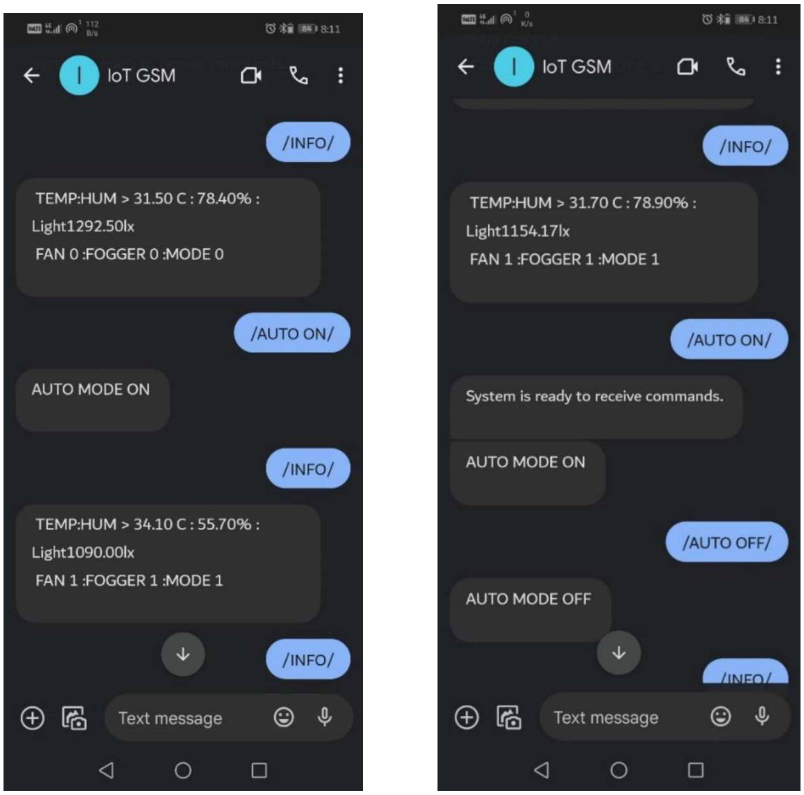
(a) (b) (c)

(d) (e)

Fig. 4.20 Live monitoring of light intensity using IoT remote mobile application (a) Live reading (b) One hour reading (c) One day reading (d) 7 days reading (e) 15 days reading

4.3.3.4 Monitoring and controlling using GSM by Sending SMS

Monitoring and controlling was also made possible by using a GSM. A Vodafone Idea sim was inserted in the GSM and it was linked to the IoT system through proper computer coding. By sending commands, it automatically sent all the informations through SMS to the connected Android mobile phone. Various screenshots of monitoring and controlling using GSM on 07/02/2023 are shown in Fig. 4.21. For actuators, fan and fogger, ‘1’ indicated ‘ON’ and ‘0’ indicated ‘OFF’ whereas for mode, ‘0’ indicated ‘manual’ mode and ‘1’ indicated ‘automatic’ mode. When the command ‘/INFO/’ was sent, system automatically sent the values of temperature, RH& light intensity, condition of fan and fogger and the mode (automatic or manual) as shown in Fig. 4.21(a) and Fig. 4.21(b). Similar monitoring of temperature and RH using GSM were conducted by Zheng *et al*. (2016).



/INFO/

TEMP:HUM > 31.50 C : 78.40% :
Light1292.50lx
FAN 0 :FOGGER 0 :MODE 0

/AUTO ON/

AUTO MODE ON

/INFO/

TEMP:HUM > 34.10 C : 55.70% :
Light1090.00lx
FAN 1 :FOGGER 1 :MODE 1

/INFO/

(a)

/INFO/

TEMP:HUM > 31.70 C : 78.90% :
Light1154.17lx
FAN 1 :FOGGER 1 :MODE 1

/AUTO ON/

System is ready to receive commands.

AUTO MODE ON

/AUTO OFF/

AUTO MODE OFF

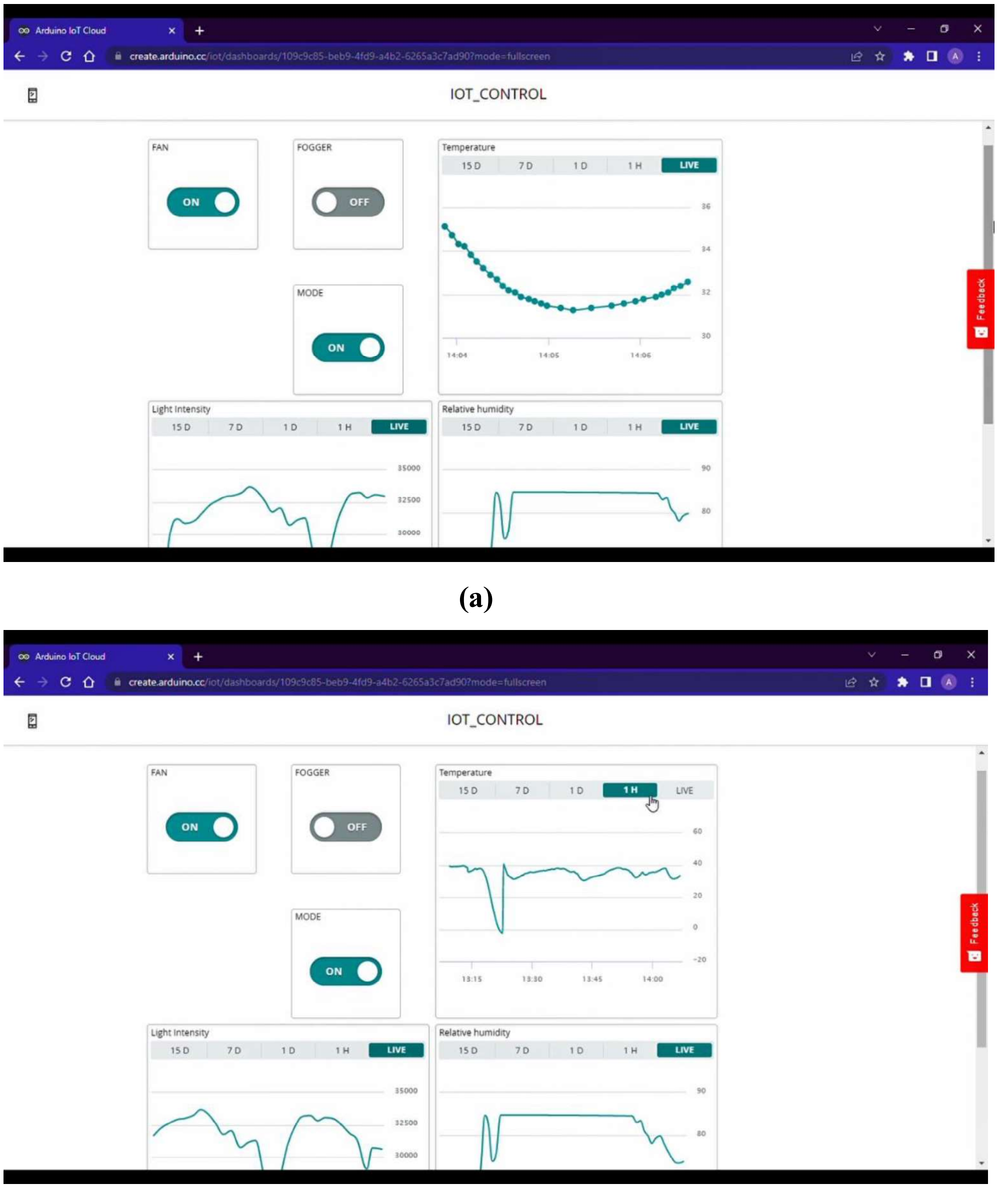
/INFO/

(b)

Fig. 4.21 Screenshots of monitoring and controlling using GSM by sending SMS

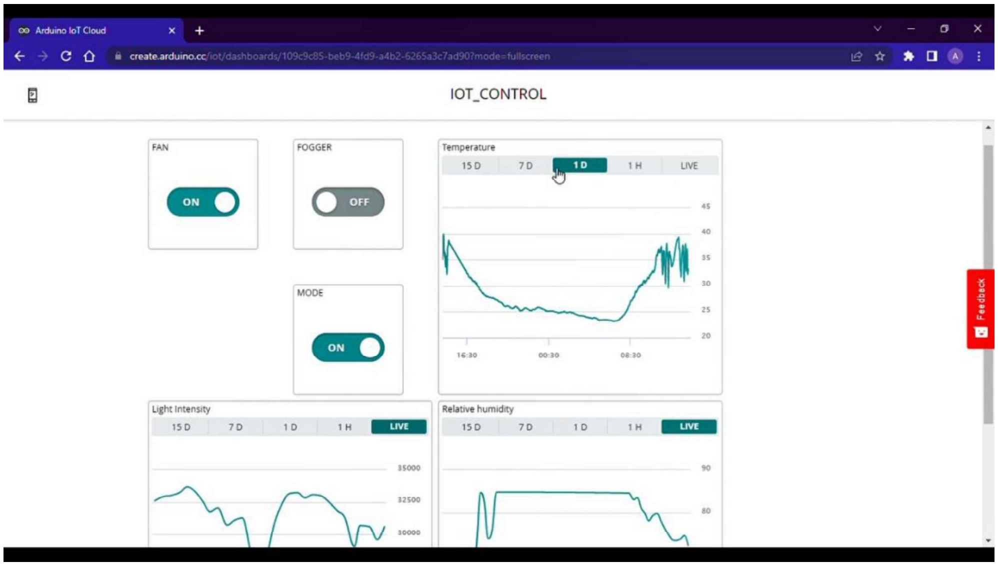
4.3.3.5 Monitoring and Controlling using Arduino IoT Cloud in Personal Computer (PC)

Real-time monitoring of temperature, RH and light intensity were also done by PC through the IoT platform 'Arduino IoT Cloud'. Using Arduino IoT cloud live reading, one hour reading, one day reading, one week reading and 15 days reading of temperature, RH and light intensity was able to monitor (Fig. 4.22 to Fig. 4.24).

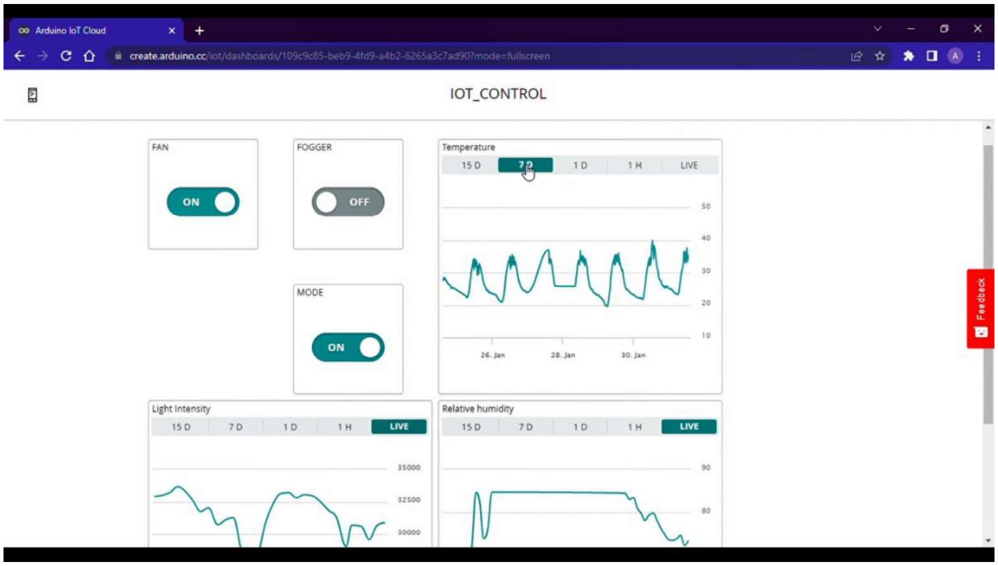


(a)

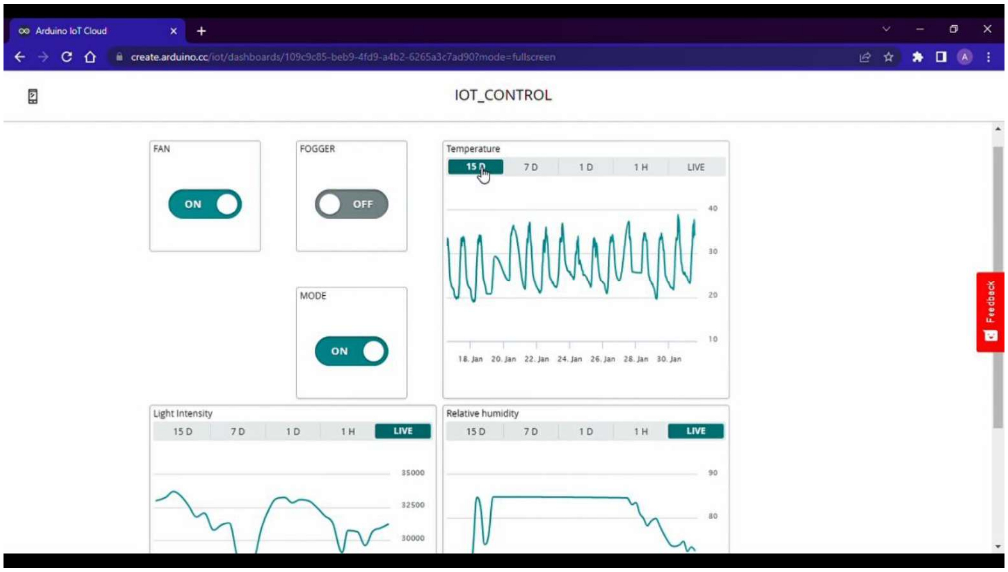
(b)


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
FAN
ON
FOGGER
OFF
MODE
ON
Temperature
15 D
7 D
1 D
1 H
LIVE
45
40
35
30
25
20
16:30
00:30
08:30
Light Intensity
15 D
7 D
1 D
1 H
LIVE
35000
32500
30000
Relative humidity
15 D
7 D
1 D
1 H
LIVE
90
80
Feedback

(c)

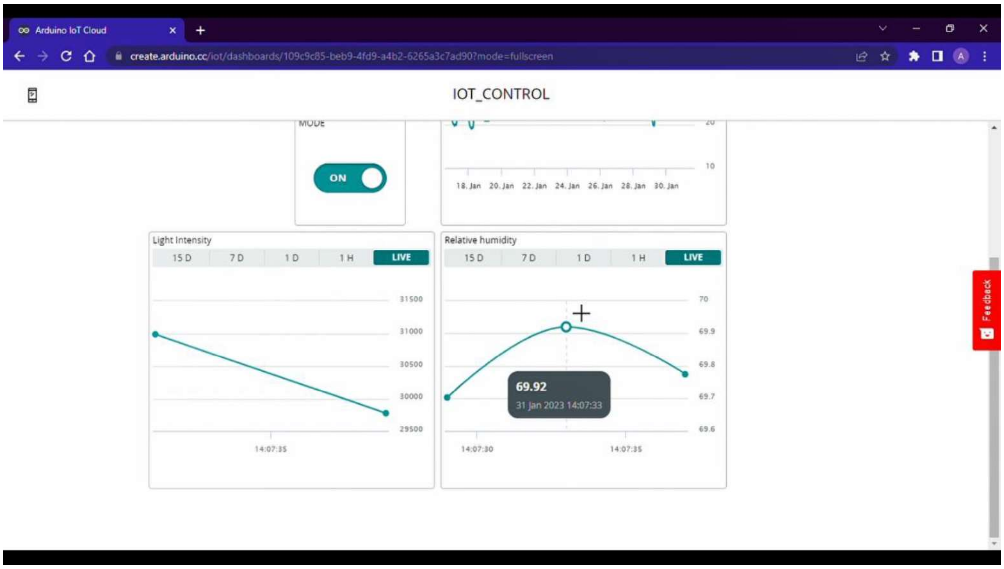

Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
FAN
ON
FOGGER
OFF
MODE
ON
Temperature
15 D
7 D
1 D
1 H
LIVE
50
40
30
20
10
26. Jan
28. Jan
30. Jan
Light Intensity
15 D
7 D
1 D
1 H
LIVE
35000
32500
30000
Relative humidity
15 D
7 D
1 D
1 H
LIVE
90
80
Feedback

(d)

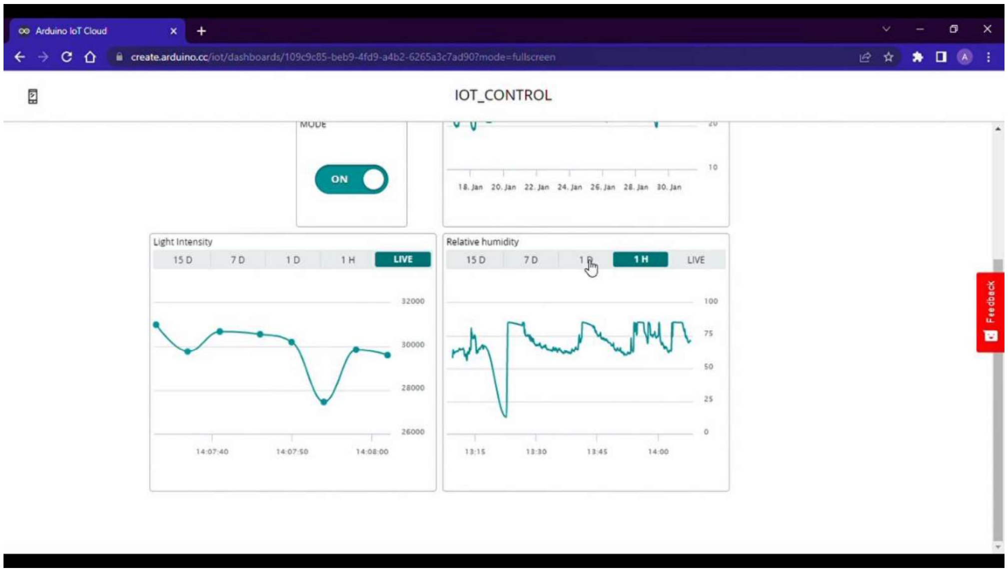


(e)

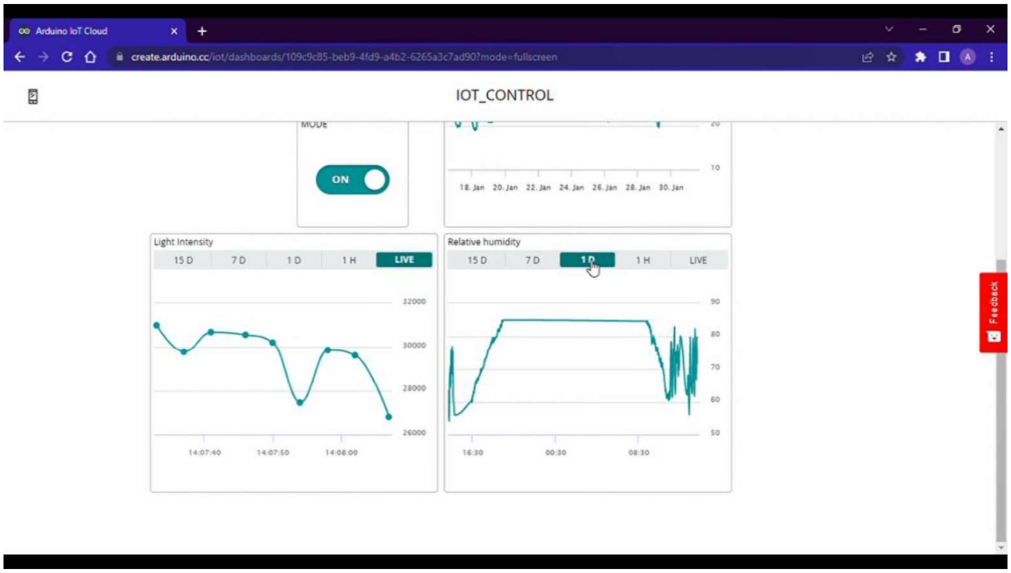
Fig. 4.22 Monitoring and controlling of temperature using Arduino IoT cloud in PC (a)Live reading (b) One hour reading (c) One day reading (d) One week reading (e) 15 days reading



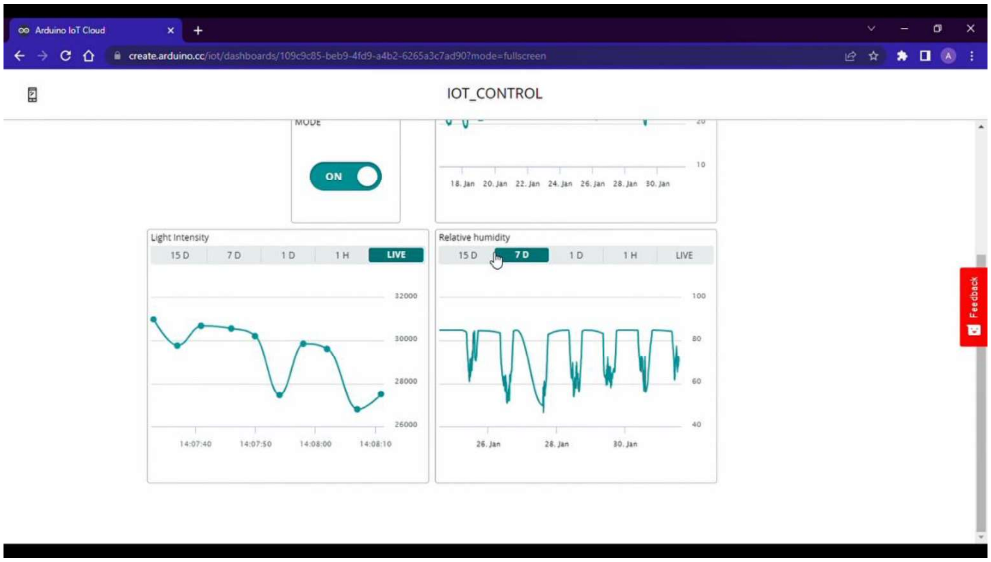
(a)


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
MODE
ON
18. Jan 20. Jan 22. Jan 24. Jan 26. Jan 28. Jan 30. Jan
10
Light Intensity
15 D 7 D 1 D 1 H LIVE
32000
30000
28000
26000
14:07:40 14:07:50 14:08:00
Relative humidity
15 D 7 D 1 D 1 H LIVE
100
75
50
25
0
13:15 13:30 13:45 14:00
Feedback

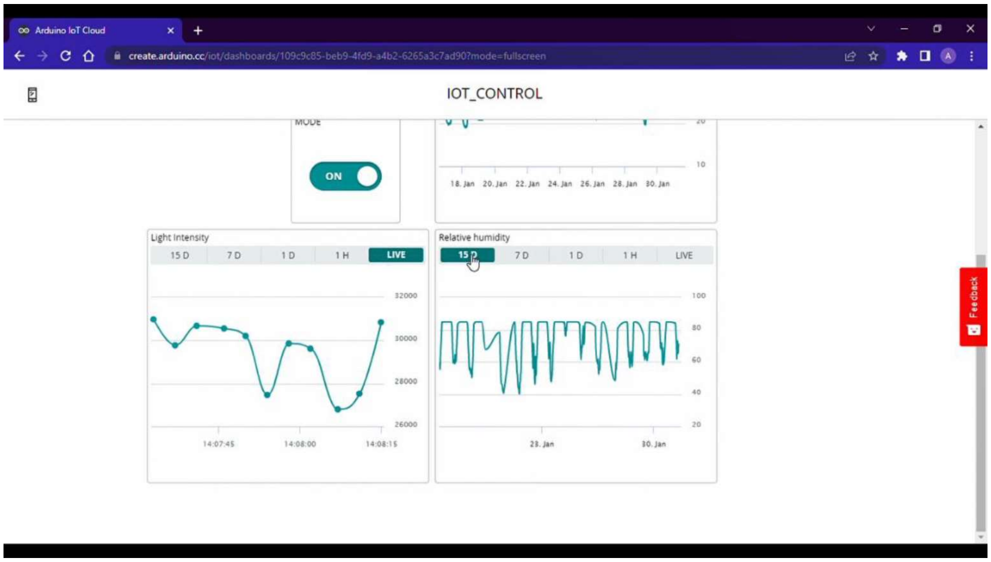
(b)


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
MODE
ON
18. Jan 20. Jan 22. Jan 24. Jan 26. Jan 28. Jan 30. Jan
10
Light Intensity
15 D 7 D 1 D 1 H LIVE
32000
30000
28000
26000
14:07:40 14:07:50 14:08:00
Relative humidity
15 D 7 D 1 D 1 H LIVE
90
80
70
60
50
16:30 00:30 08:30
Feedback

(c)

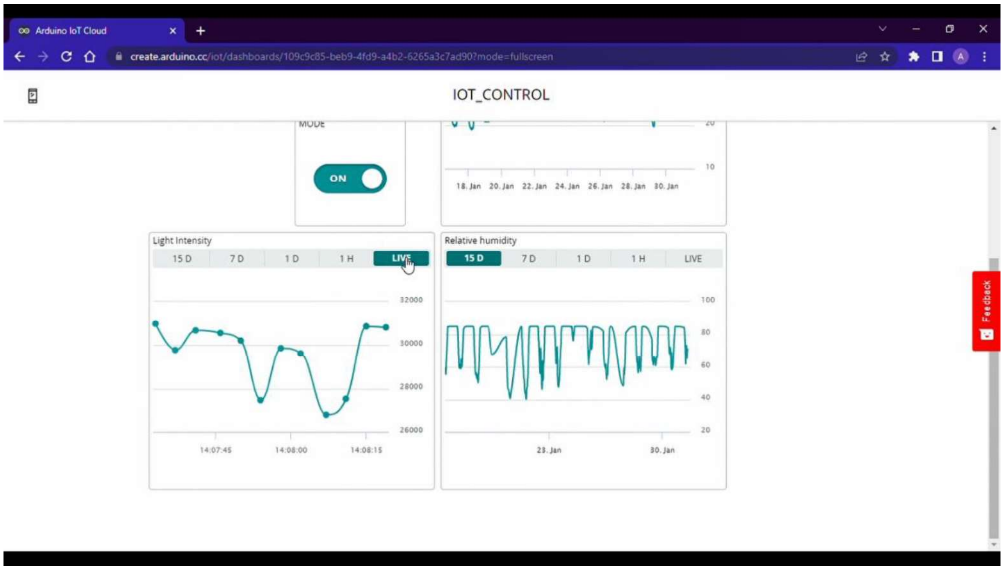


(d)

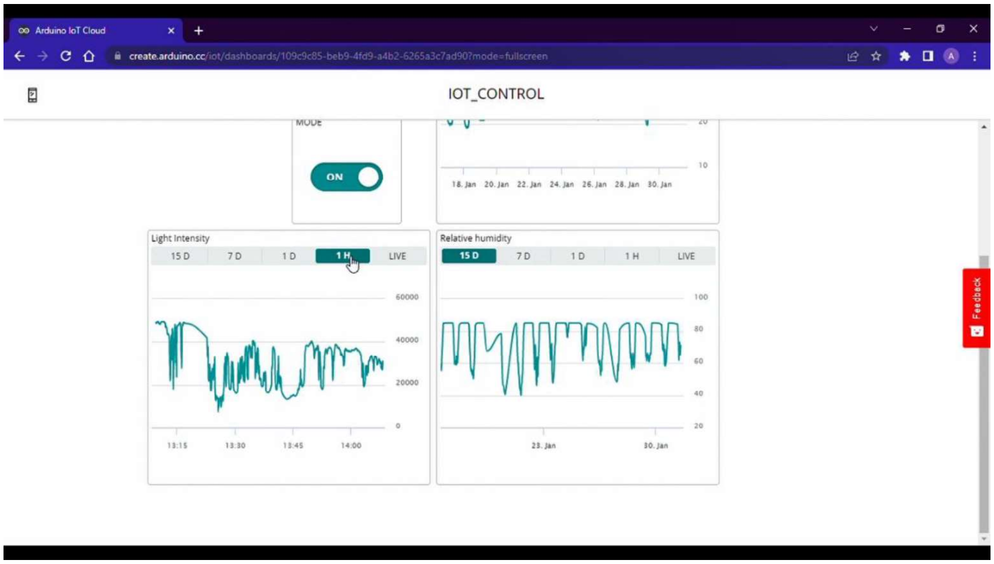


(e)

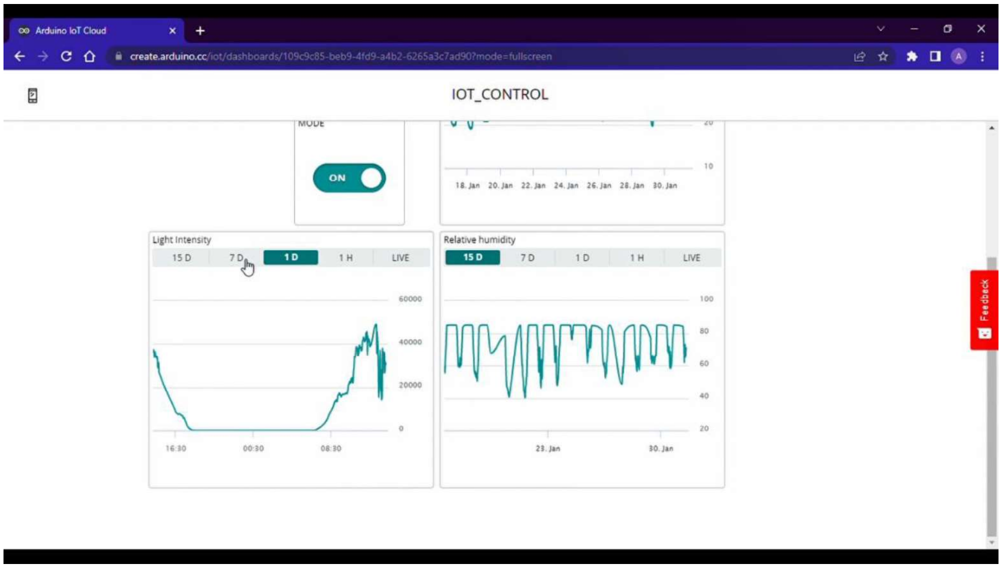
Fig. 4.23 Monitoring and controlling of RH using Arduino IoT cloud in PC (a)Live reading (b) One hour reading (c) One day reading (d) One week reading (e) 15 days reading


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
MODE
ON
18. Jan 20. Jan 22. Jan 24. Jan 26. Jan 28. Jan 30. Jan
10
Light Intensity
15 D 7 D 1 D 1 H LIVE
32000
30000
28000
26000
14:07:45 14:08:00 14:08:15
Relative humidity
15 D 7 D 1 D 1 H LIVE
100
80
60
40
20
23. Jan 30. Jan
Feedback

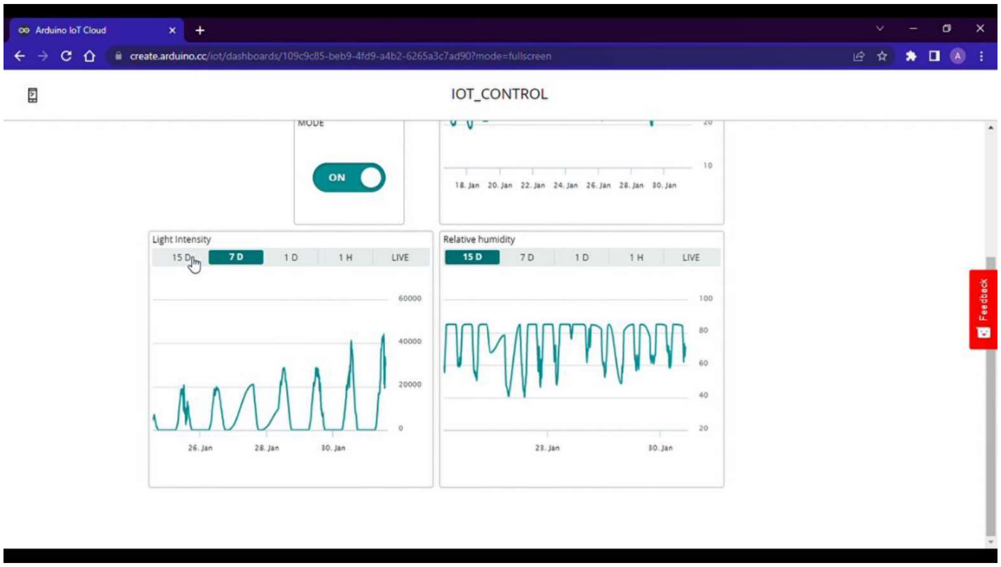
(a)


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
MODE
ON
18. Jan 20. Jan 22. Jan 24. Jan 26. Jan 28. Jan 30. Jan
10
Light Intensity
15 D 7 D 1 D 1 H LIVE
60000
40000
20000
0
13:15 13:30 13:45 14:00
Relative humidity
15 D 7 D 1 D 1 H LIVE
100
80
60
40
20
23. Jan 30. Jan
Feedback

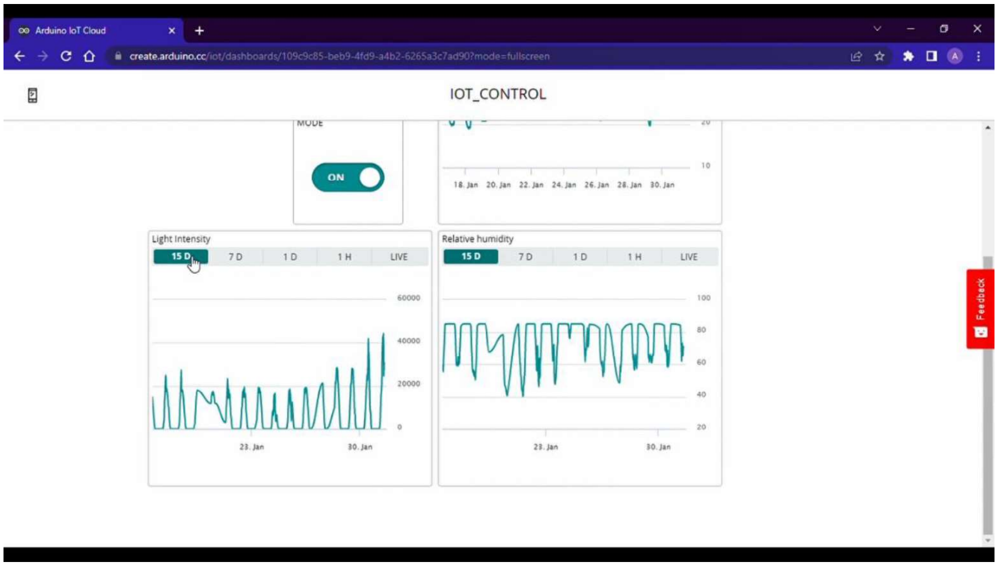
(b)


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
MODE
ON
18. Jan
20. Jan
22. Jan
24. Jan
26. Jan
28. Jan
30. Jan
10
Light Intensity
15 D
7 D
1 D
1 H
LIVE
60000
40000
20000
0
16:30
00:30
08:30
Relative humidity
15 D
7 D
1 D
1 H
LIVE
100
80
60
40
20
23. Jan
30. Jan
Feedback

(c)


Arduino IoT Cloud
create.arduino.cc/iot/dashboards/109c9c85-beb9-4fd9-a4b2-6265a3c7ad90?mode=fullscreen
IOT_CONTROL
MODE
ON
18. Jan
20. Jan
22. Jan
24. Jan
26. Jan
28. Jan
30. Jan
10
Light Intensity
15 D
7 D
1 D
1 H
LIVE
60000
40000
20000
0
26. Jan
28. Jan
30. Jan
Relative humidity
15 D
7 D
1 D
1 H
LIVE
100
80
60
40
20
23. Jan
30. Jan
Feedback

(d)

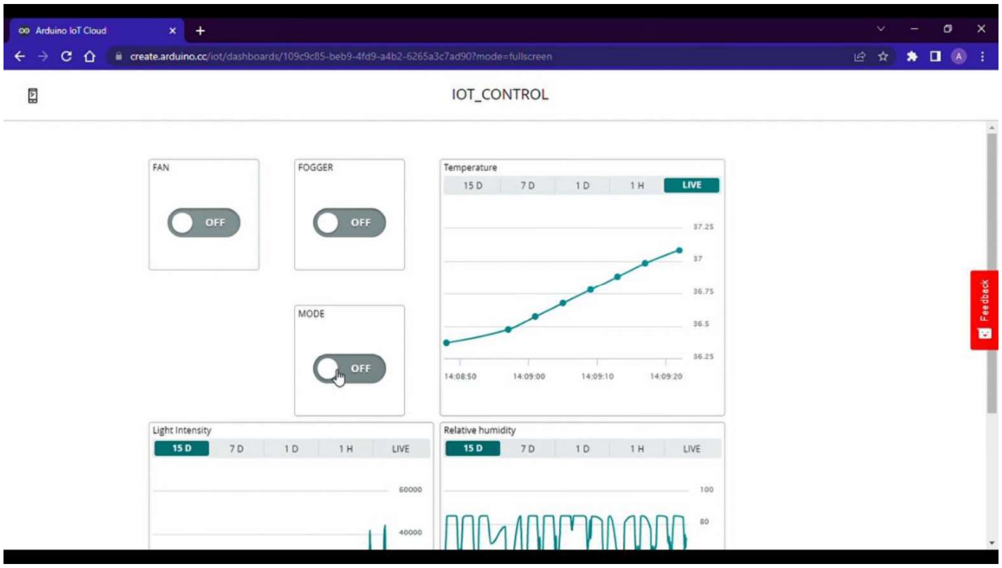


(e)

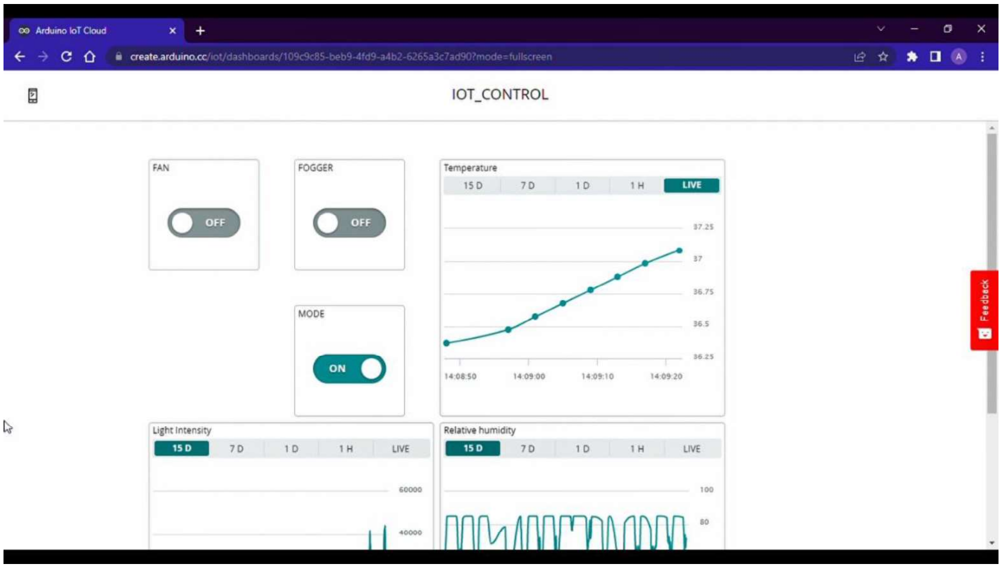
Fig. 4.24 Monitoring and controlling of light intensity using Arduino IoT cloud in PC (a)Live reading (b) One hour reading (c) One day reading (d) One week reading (e) 15 days reading

4.3.3.6 Automatic Operation of Exhaust Fans and Foggers based on Temperature Value using IoT System

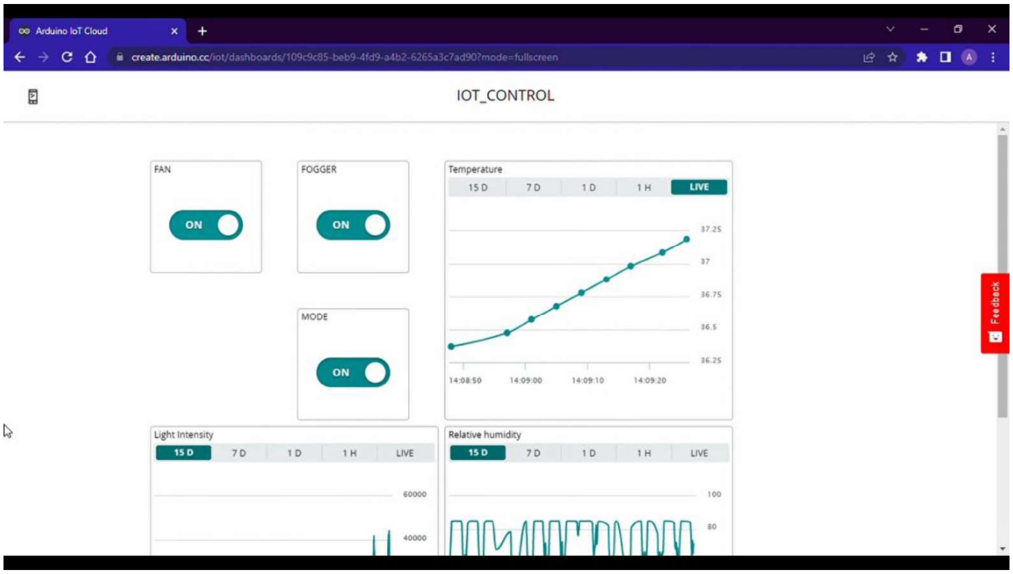
Exhaust fans and foggers were the actuators used in this system. Automatic operation of exhaust fans and foggers were made based on temperature value in the program which was added in the Arduino code. Whenever the temperature inside the polyhouse exceeded 28°C, the controller switched 'ON' the exhaust fans and reduced the temperature and turned 'OFF' when the temperature reached below 25°C. . It was also found that the actuator 'exhaust fan' alone was not able to reduce the temperature up to the desired level, hence fogger was also connected to the system to maintain temperature as well as RH. Hence, it was found that when the temperature exceeded 35°C, fogger automatically switched 'ON' and switched 'OFF' when the temperature reached below 32°C.The system was able to operate in two modes i.e, manual and automatic mode. Fig. 4.25 (a) shows the window of Arduino IoT cloud. The mode was put in 'OFF' condition that means it was in manual mode. When the mode was put 'ON', the system switched to automatic mode as shown in Fig. 4.25 (b). For example, it was found that the actuators, foggers and exhaust fans, automatically switched ‘ON’ when the temperature was 37°C as shown in Fig. 4.25 (c) which was above the threshold value of exhaust fans (28°C) and foggers (35°C).



(a)



(b)



(c)

Fig. 4.25 IoT based system for operation of fan and fogger (a) System in manual mode (MODE 'OFF') (b) Swiched to automatic mode (MODE 'ON') (c) Automatic operation of exhaust fans and foggers (FAN 'ON'; FOGGER 'ON')

4.3.3.7 Monitoring and Controlling of Temperature and RH

Temperature, RH and light intensity were monitored for a period of one week to check whether the IoT system is working properly. The monitoring and control achieved over a day and over a week are explained under different subheads. In the case of light intensity, it was monitored for a period of one week and no control was given as it was in the required range. The results of monitoring of light intensity is also explained.

a) Monitoring and Controlling of Temperature and RH Over a Day

Temperature and RH variation was observed on one day to check whether the IoT system have worked effectively by the proper working of exhaust fans and foggers. For this, microclimate parameters data were directly downloaded from the Arduino IoT cloud and was converted from Zulu time zone to IST using excel. Temperature monitored on the date 12/01/23 (from 5.30 AM to 3.00 AM on the next day) is shown in Fig. 4.26 and it was found that the temperature has increased above 35°C at 11 'O' clock and it was reduced to 32°C (approx.) after few minutes because of the automatic operation of fogger and exhaust fans. Similarly, RH monitored on 12/01/23 (from 5.30 AM to 3.00 AM on the next day) is shown in Fig. 4.27. It was found that RH was high up to 11'O' clock (80% approx.) and reduced to approximately 65% and then increased to approximately 70% due to the automatic operation of fogger. Since it was a naturally ventilated polyhouse the full control was not able to achieve. In order to achieve full climate control the polyhouse should be restructured with improved control mechanism for temperature and RH. Anyway the microclimate parameters temperature and RH was controlled to a greater extent by the system.

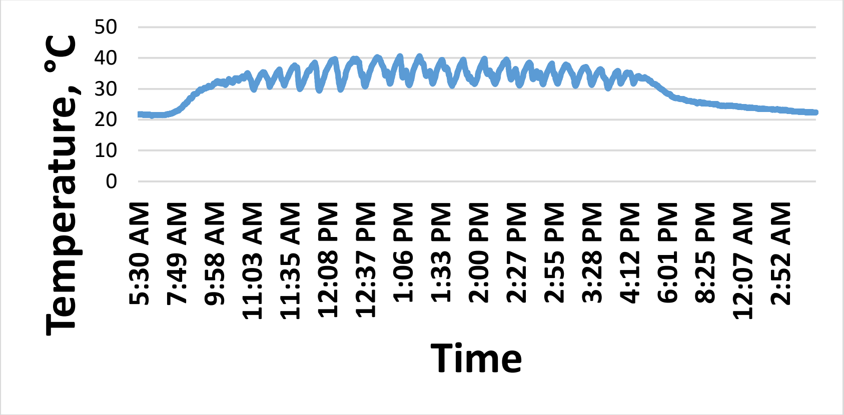


Fig. 4.26 Temperature monitored on 12/01/2023

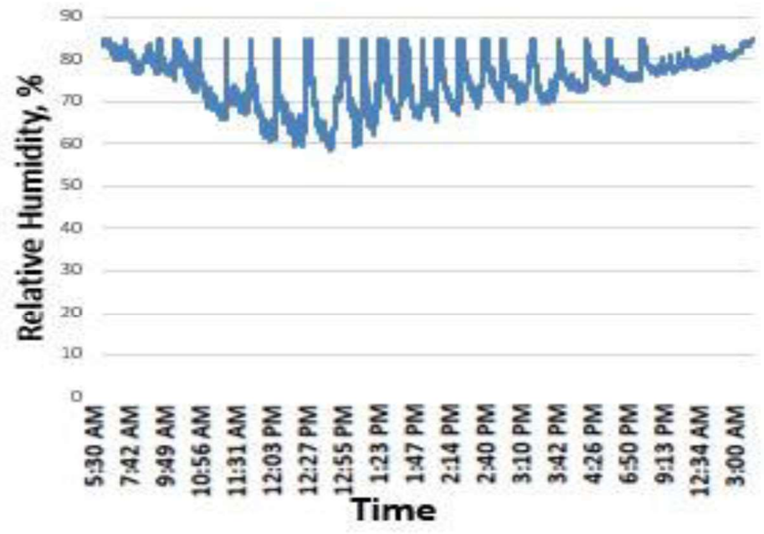
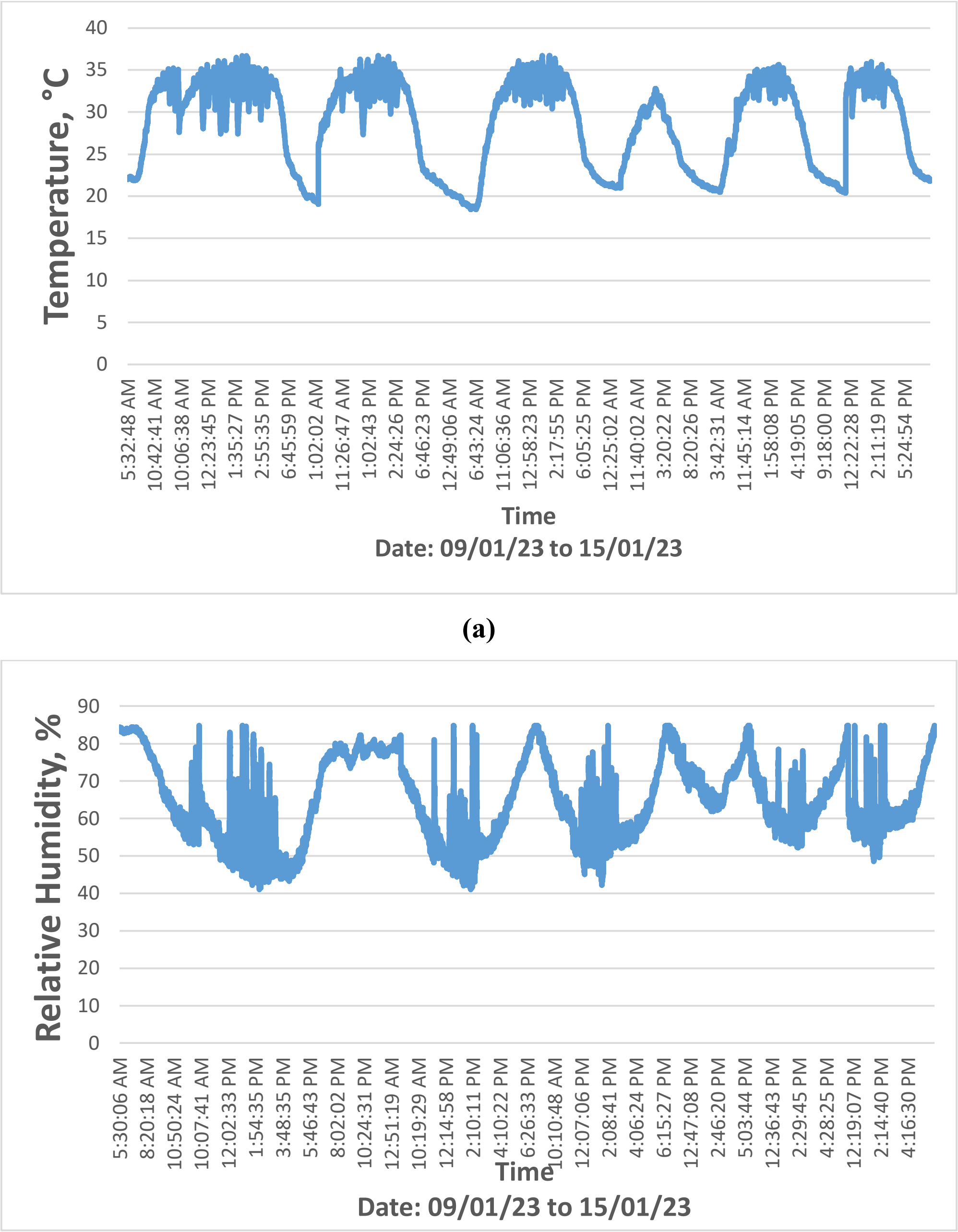


Fig. 4.27 RH monitored on 12/01/2023

b) Temperature and RH Monitored for a Period of One Week

Temperature and RH monitored and controlled during a period of one week (09/01/2023 to 15/01/2023) to know the level of control the system can achieve. For this purpose, data was downloaded from the Arduino IoT cloud, converted to IST and plotted in excel. The variation of temperature and RH monitored for the week are shown in Fig. 4.28 (a) and Fig. 4.28 (b) respectively. It was found that temperature was maintained approximately in the range of 28°C to 37°C and RH in the range of 45% to 85%. This control was achieved due to the automatic operation of fan and fogger.



(a)

(b)

Fig. 4.28 Temperature and RH monitored and controlled for one week (09/01/2023 to 15/01/2023)

c) Light Intensity Monitored During a Period of One Week

Light intensity was monitored for a period of one week from 09/01/2023 to 15/01/2023 as shown in Fig. 4.29. The maximum light intensity during the period was 38,000 lx (approx.).It was also found that the light intensity never exceeded the upper threshold (50,000 lx), so there was no need of control of light intensity in the polyhouse due to the prevailing climatic conditions during the study period.

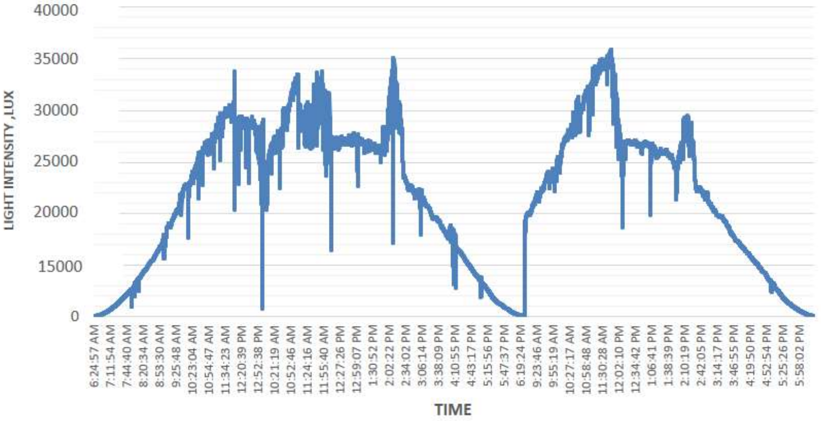


Fig. 4.29 Light intensity monitored from 09/01/2023 to 15/01/2023

4.3.3.8 Average Temperature, RH and Light Intensity Obtained Inside the Polyhouse During Testing Period Without Crop

Average temperature, RH and light intensity inside the polyhouse during the testing period without crop from 09/01/23 to 15/01/23 was obtained and is shown in Table 4.7. By referring to Fig. 4.28 and Fig. 4.29 the average temperature, RH and light intensity monitored for a period of one week were 35°C, 64.33% and 32,000 lx respectively. Hence, it was clear that the temperature was reduced by 3°C and RH increased by 4% due to the installation of IoT based microclimate monitoring and controlling system inside the polyhouse.

Table 4.7 Microclimate parameters and the change achieved inside the polyhouse during a period of one week without crop

Parameters	**Average value**	**Change achieved**
Temperature	35°C	- 3°C
RH	64.33%	+ 4%
Light Intensity	32,000 lx	-

4.4 TESTING THE IoT BASED AUTOMATION SYSTEM WITH CROP INSIDE THE POLYHOUSE

The IoT based real-time microclimate monitoring and controlling system was tested with crop ‘bhindi’ inside the polyhouse. The system has continuously monitored and controlled the microclimate parameters throughout the crop period. In order to check the daily performance of the system, temperature, RH and light intensity were monitored one day (01/02/2023) during crop growth period and is shown in Fig. 4.30. The same for one week period is shown Fig. 4.31. It was found that the system performed successfully in monitoring and controlling.

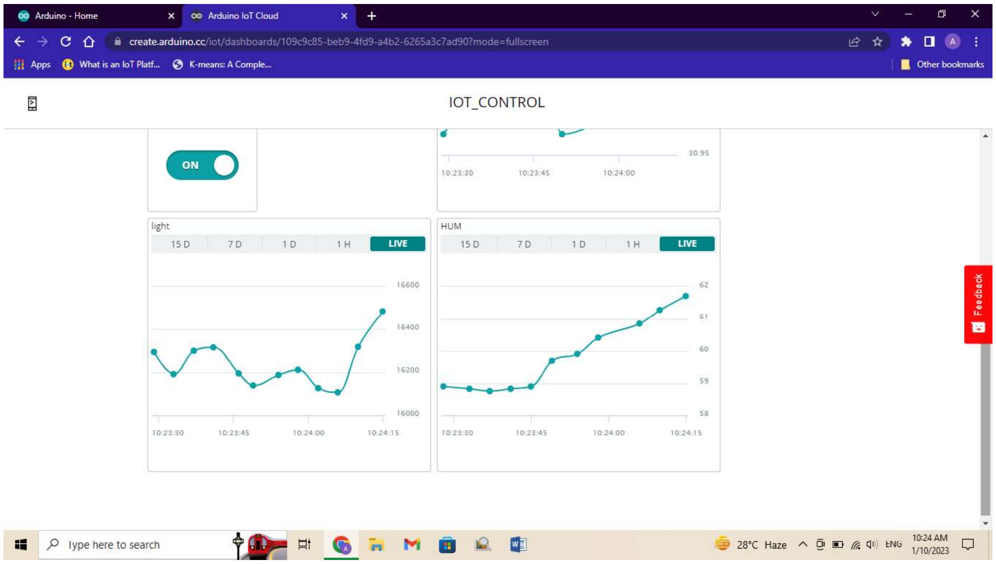


Fig. 4.30 Real-time continuous monitoring during crop growth on 01/02/2023

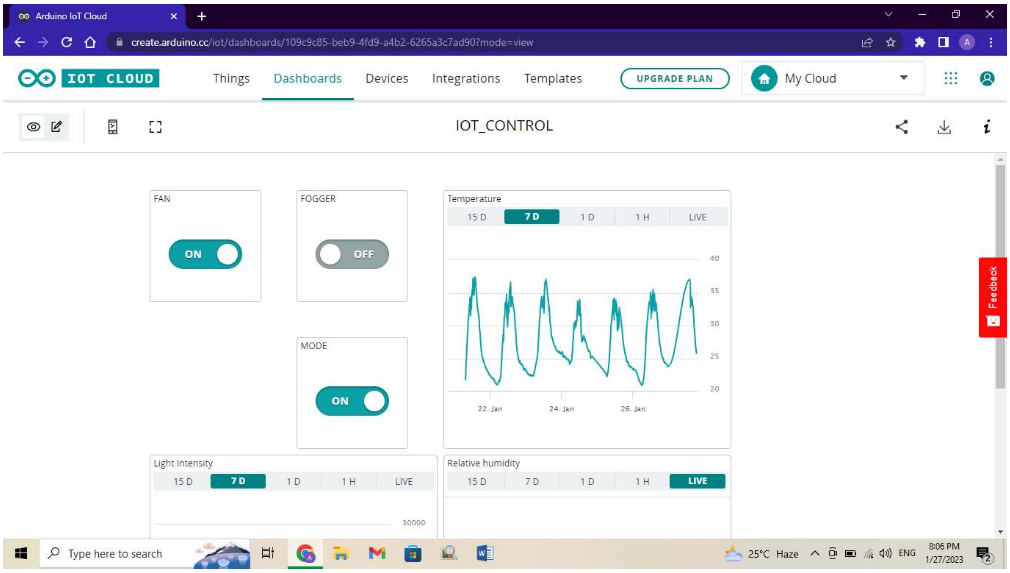


Fig. 4.31 Real-time continuos monitoring during crop period for a period of seven days (21/01/23 to 27/01/23)

4.4.1 Variation of Air Temperature and RH Readings Inside and Outside the Polyhouse During the Crop Growing Period

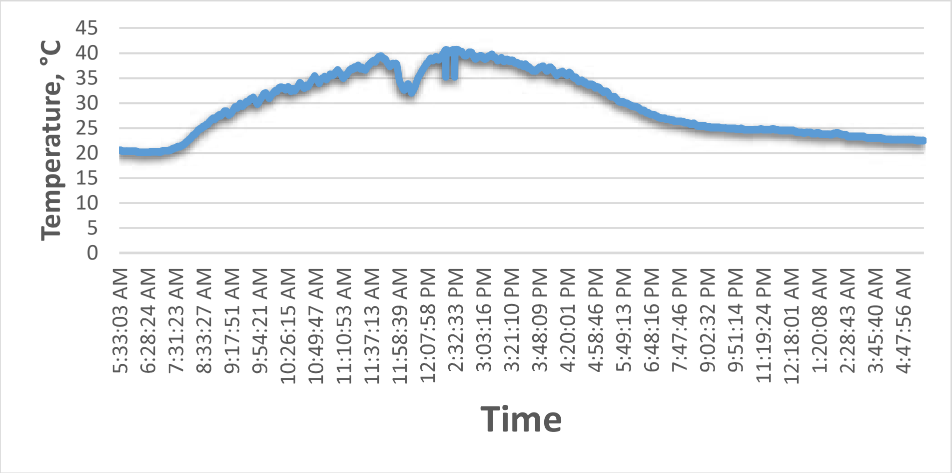
Temperature and RH values inside (with IoT system) and outside the polyhouse were measured using drybulb & wetbulb hygrometer. The measurements were taken for a period of two weeks 20/01/23 to 04/02/23 and the average values are presented in Table 4.8. Usually temperature inside the polyhouse is higher and RH is lower than the temperature and RH values respectively outside the polyhouse (Jha and Kumari, 2015). In order to compare the results, after the installation of the system, microclimate parameters temperature, RH and light intensity both inside and outside polyhouse were recorded using instruments drybulb & wetbulb hygrometer and luxmeter for a period of two weeks. After the installation of IoT based automation system, less temperature and more RH value were found inside the polyhouse than outside as shown in Table 4.8. Hence, it can be concluded that decreased temperature and increased RH readings within the polyhouse after the installation of IoT system could be due to the proper operation of actuators (foggers and exhaut fans) in real-time. The beneficial effect of real-time monitoring and control was observed during crop growth.

Table 4.8 Average values of temperature and RH readings of inside and outside polyhouse during the early stage of crop period (20/01/23 to 04/02/23)

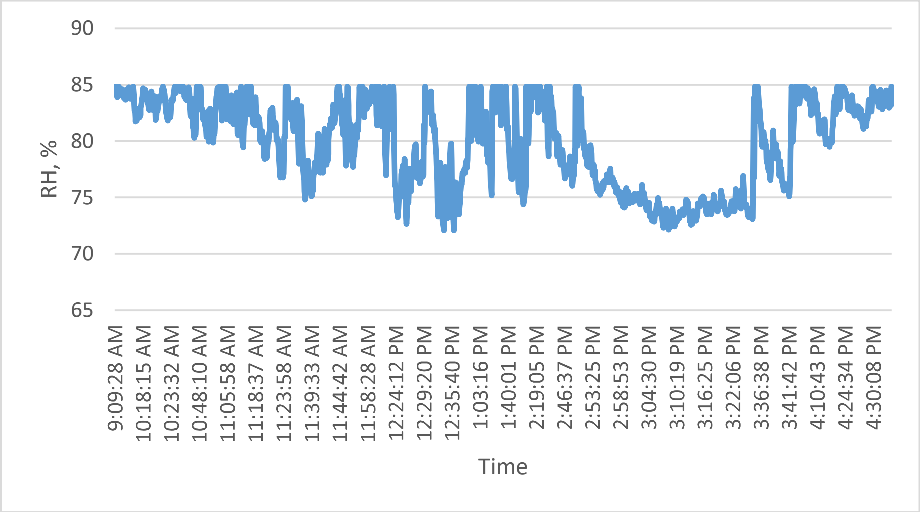
Time (h)	Temperature (°C)		RH (%)	
	Inside polyhouse	Outside polyhouse	Inside polyhouse	Outside polyhouse
9:00 AM	30.3	31	82.4	78.5
11:00 AM	34.3	35	62	59
1:00 PM	36.5	37	52	49
3:00 PM	37.5	39	49	47
5:00 PM	34.0	35	60.5	59

4.4.2 Variation of Temperature, RH and Light Intensity Monitored During Early, Mid and Late Stage of Crop

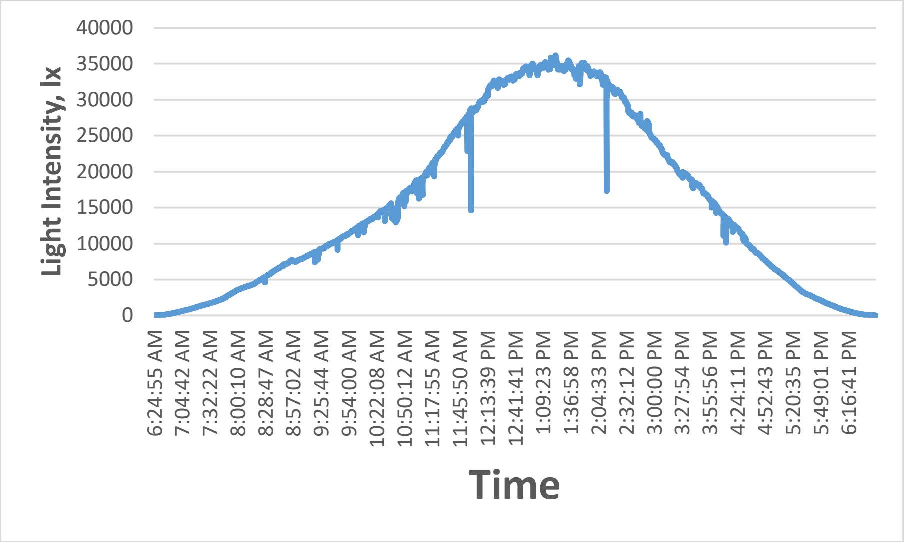
Variation of temperature, RH and light intensity were monitored during early, mid and late stage of crop. Temperature and RH were controlled to a greater extent during the crop period due to the installation of IoT based real-time microclimate monitoring and controlling system inside the polyhouse. Temperature, RH and light intensity monitored on different dates 25/01/23 (early stage), 04/03/23 (mid stage) and 30/04/23 (late stage) are shown in Fig. 4.32, Fig. 4.33 and Fig. 4.34 respectively. In the early stage, temperature was maintained in the approximate range of 24°C to 36°C and RH in the range of 75% to 80%. The maximum light intensity observed during early stage was 36,000 lx. Temperature and RH during the mid stage was maintained approximately in the range of 25°C to 38 °C and 60% to 80% respectively and the maximum light intensity was 40,000 lx. In the late stage, temperature was maintained in the range of 24°C to 39°C and RH in the range of 55% to 80%. The maximum light intensity observed during late stage was 45,000 lx.


Temperature, °C
45
40
35
30
25
20
15
10
5
0
5:33:03 AM
6:28:24 AM
7:31:23 AM
8:33:27 AM
9:17:51 AM
9:54:21 AM
10:26:15 AM
10:49:47 AM
11:10:53 AM
11:37:13 AM
11:58:39 AM
12:07:58 PM
2:32:33 PM
3:03:16 PM
3:21:10 PM
3:48:09 PM
4:20:01 PM
4:58:46 PM
5:49:13 PM
6:48:16 PM
7:47:46 PM
9:02:32 PM
9:51:14 PM
11:19:24 PM
12:18:01 AM
1:20:08 AM
2:28:43 AM
3:45:40 AM
4:47:56 AM
Time

(a)

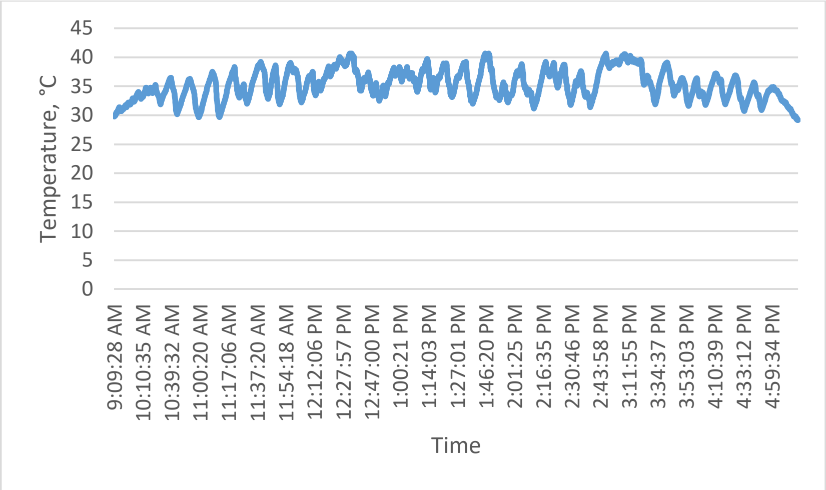

90
85
80
75
70
65
RH, %
9:09:28 AM
10:18:15 AM
10:23:32 AM
10:48:10 AM
11:05:58 AM
11:18:37 AM
11:23:58 AM
11:39:33 AM
11:44:42 AM
11:58:28 AM
12:24:12 PM
12:29:20 PM
12:35:40 PM
1:03:16 PM
1:40:01 PM
2:19:05 PM
2:46:37 PM
2:53:25 PM
2:58:53 PM
3:04:30 PM
3:10:19 PM
3:16:25 PM
3:22:06 PM
3:36:38 PM
3:41:42 PM
4:10:43 PM
4:24:34 PM
4:30:08 PM
Time

(b)

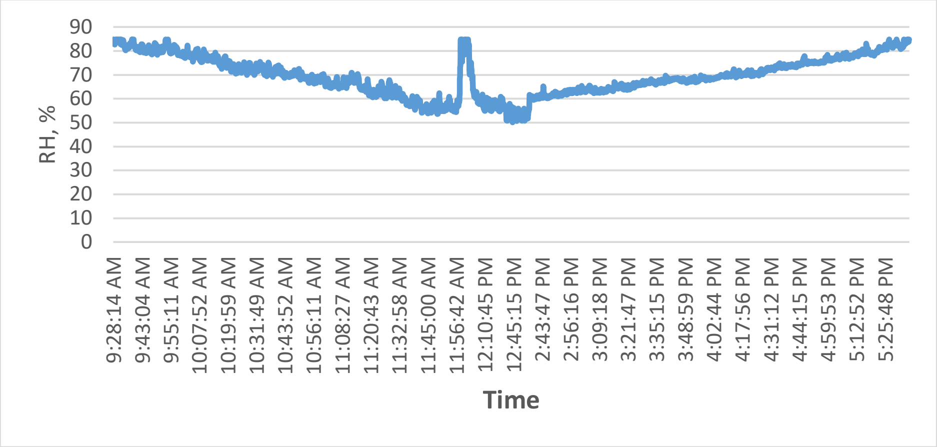


(c)

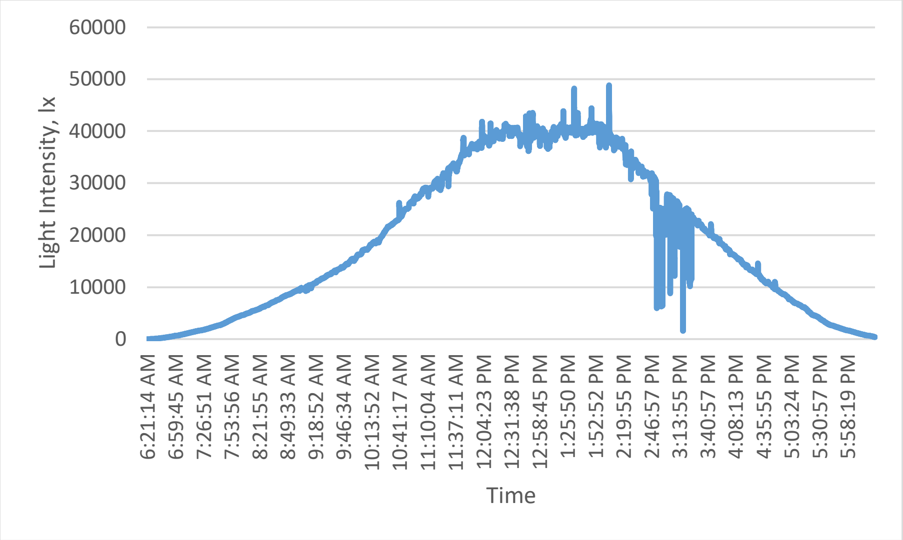
Fig. 4.32 Variation of microclimate parameters during early stage of crop on 25/01/2023 (a) Temperature (b) RH (c) Light intensity



(a)

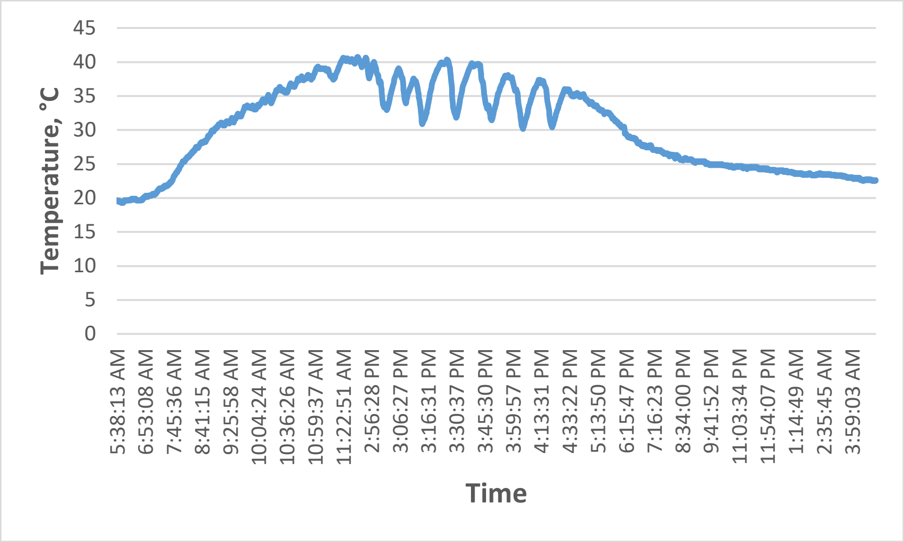


(b)

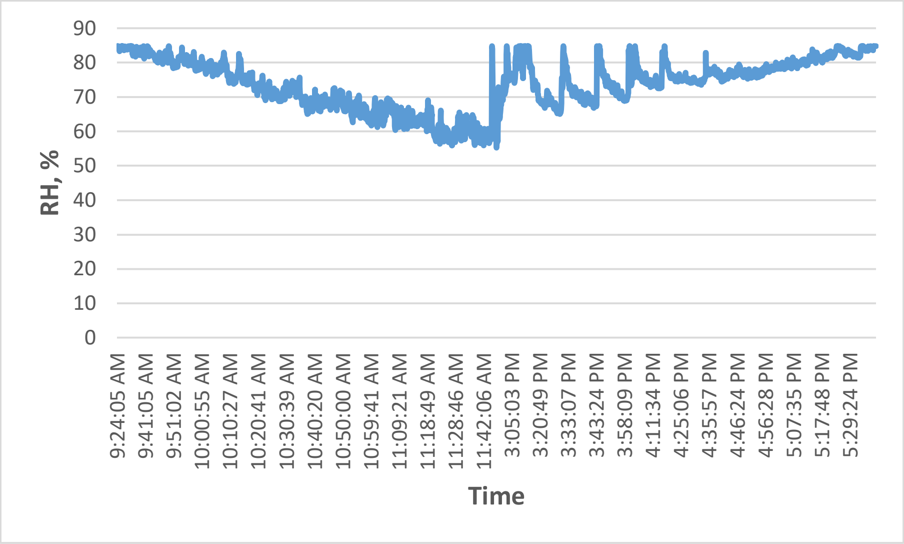


(c)

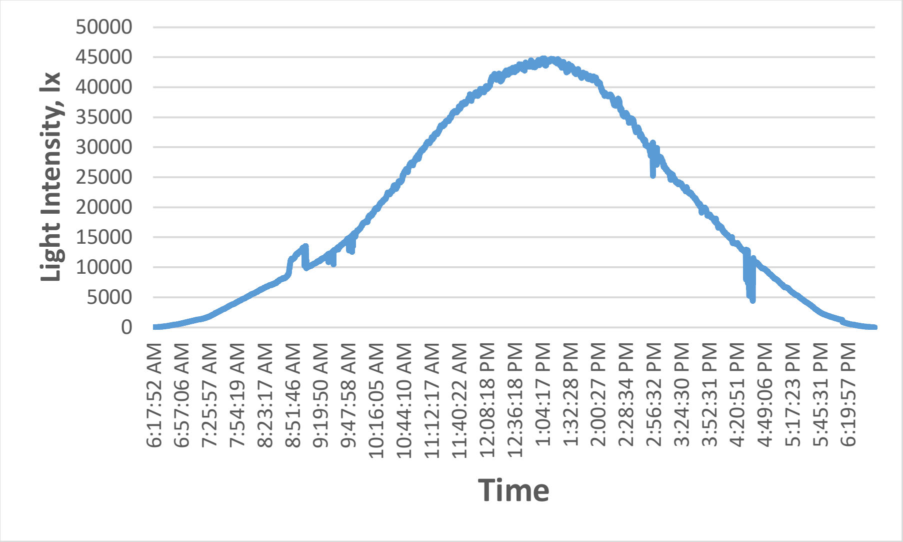
Fig. 4.33 Variation of microclimate parameters during mid stage of crop on 04/03/2023 (a) Temperature (b) RH (c) Light intensity


Temperature, °C
45
40
35
30
25
20
15
10
5
0
5:38:13 AM
6:53:08 AM
7:45:36 AM
8:41:15 AM
9:25:58 AM
10:04:24 AM
10:36:26 AM
10:59:37 AM
11:22:51 AM
2:56:28 PM
3:06:27 PM
3:16:31 PM
3:30:37 PM
3:45:30 PM
3:59:57 PM
4:13:31 PM
4:33:22 PM
5:13:50 PM
6:15:47 PM
7:16:23 PM
8:34:00 PM
9:41:52 PM
11:03:34 PM
11:54:07 PM
1:14:49 AM
2:35:45 AM
3:59:03 AM
Time

(a)


RH, %
90
80
70
60
50
40
30
20
10
0
9:24:05 AM
9:41:05 AM
9:51:02 AM
10:00:55 AM
10:10:27 AM
10:20:41 AM
10:30:39 AM
10:40:20 AM
10:50:00 AM
10:59:41 AM
11:09:21 AM
11:18:49 AM
11:28:46 AM
11:42:06 AM
3:05:03 PM
3:20:49 PM
3:33:07 PM
3:43:24 PM
3:58:09 PM
4:11:34 PM
4:25:06 PM
4:35:57 PM
4:46:24 PM
4:56:28 PM
5:07:35 PM
5:17:48 PM
5:29:24 PM
Time

(b)



(c)

Fig. 4.34 Variation of microclimate parameters during late stage of crop on 30/04/2023 (a) Temperature (b) RH (c) Light intensity

4.4.3 Hourly Variation of Temperature, RH and Light Intensity During Crop Period Monitored by IoT Based Microclimate Monitoring and Controlling System

IoT based microclimate monitoring and controlling system continuously monitored, the temperature, RH and light intensity throughout the crop period. In order to show the variations it was plotted at 15, 30, 45, 60, 75, 90, 105 and 120 Days After Planting (DAP). The developed system worked successfully throughout the crop period. The hourly variation of temperature, RH and light intensity during crop period are shown in Fig. 4.35, Fig. 4.36 and Fig. 4.37 respectively.

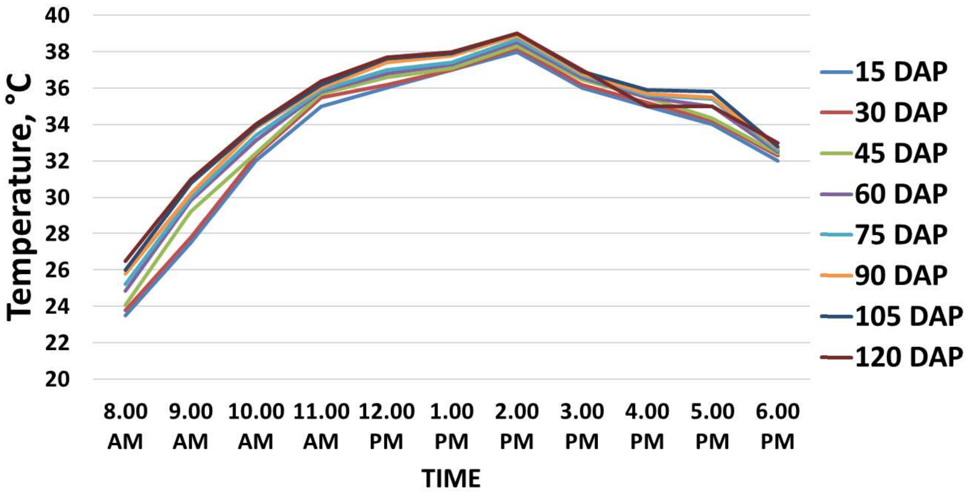


Fig. 4.35 Hourly variation of temperature during crop period monitored by IoT based microclimate monitoring and controlling system

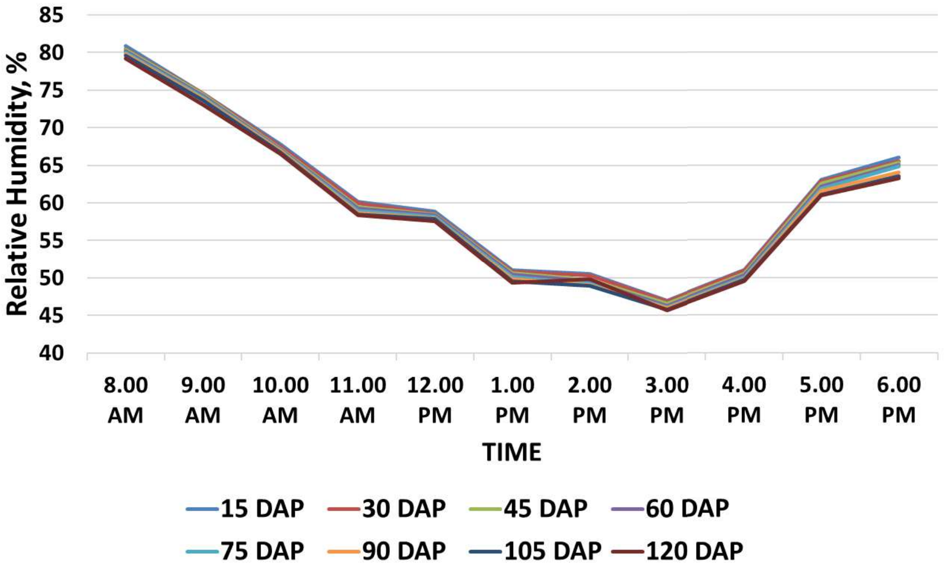


Fig. 4.36 Hourly variation of RH during crop period monitored by IoT based microclimate monitoring and controlling system

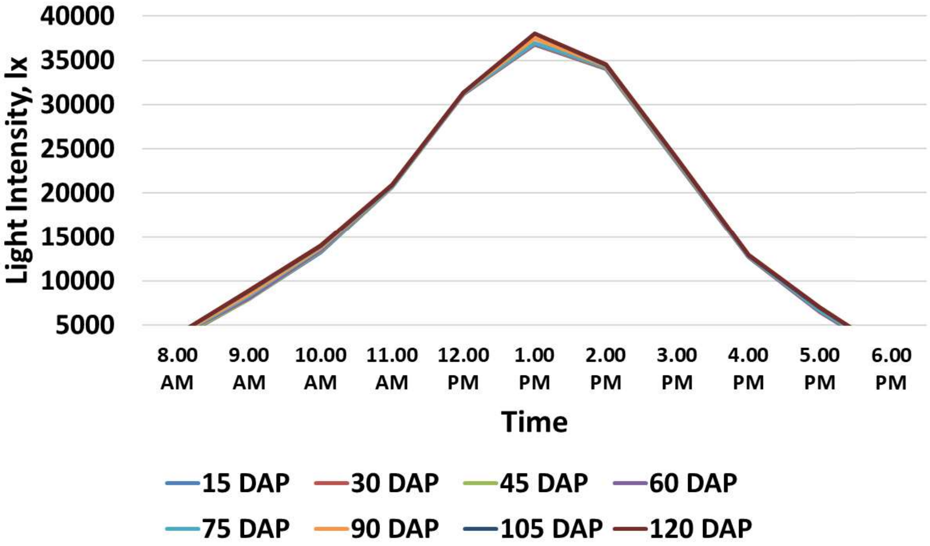


Fig. 4.37 Hourly variation of light intensity during crop period monitored by IoT based microclimate monitoring and controlling system

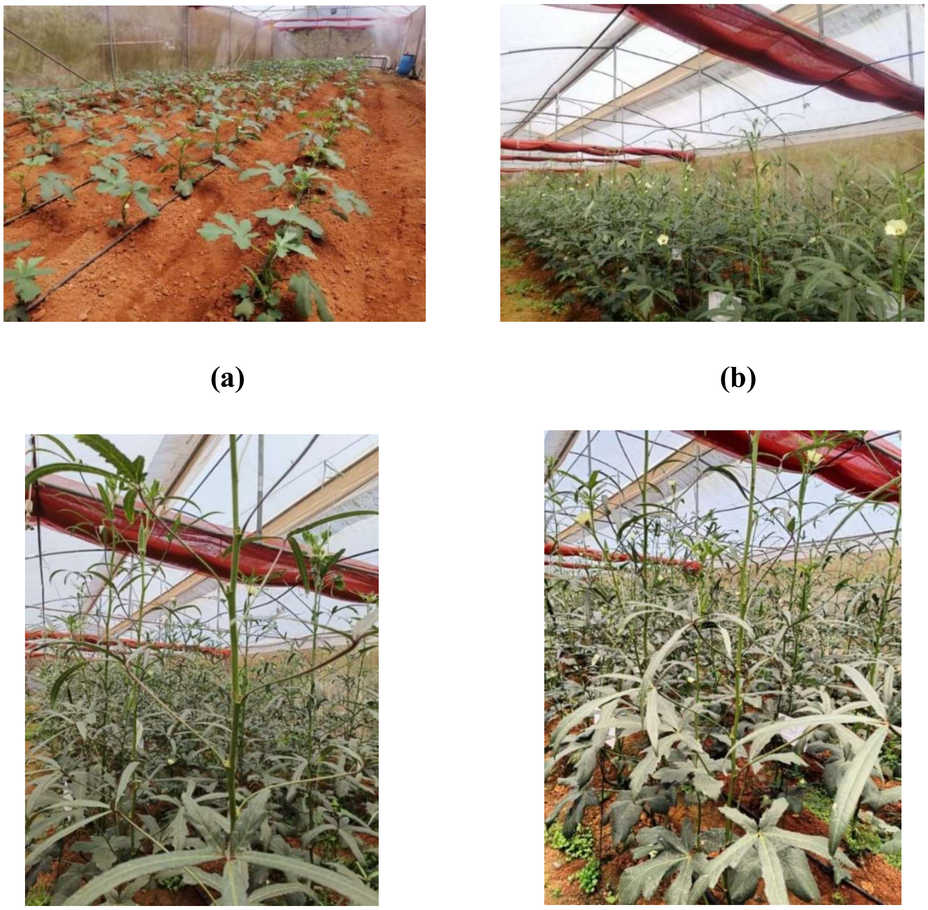
4.4.4 Average Temperature, RH and Light Intensity Obtained During the Testing Period with Crop

Average temperature, RH and light intensity inside the polyhouse obtained during the testing period with crop was obtained and is shown in Table 4.9. The average temperature, RH and light intensity monitored were 37°C, 59.22% and 35,000 lx respectively. Hence it was clear that the temperature was reduced by 3°C and RH increased by 4% due to the installation of IoT based microclimate monitoring and controlling system inside the polyhouse.

Table 4.9 Microclimate parameters and the change achieved inside the polyhouse with crop

Parameter	Average value maintained inside the polyhouse	Change achieved
Temperature	37°C	-3°C
RH	59.22%	+4%
Light Intensity	35,000 lx	-

The IoT based real- time microclimate monitoring and controlling system for polyhouse performed best in the crop growing period. Temperature, RH and light intensity were monitored successfully. Light intensity was maintained within the optimum range. The effective monitoring and controlling achieved by the IoT system benefited the crop growth. Crop showed good performance in terms of improved growth and yield parameters. Various stages of crop growth inside the polyhouse are shown in Plate 3.2.



(a) **(b)**

(c) **(d)**

Plate 4.2 Different stages of crop growth inside the polyhouse

(a) 30 DAP (b) 60 DAP (c) 90 DAP (d) 120 DAP

4.5 OBSERVATIONS ON GROWTH AND YIELD PARAMETERS

The developed IoT based real-time microclimate monitoring and controlling system was tested by growing 'Anjitha' variety of bhindi inside the polyhouse. Observations on growth and yield parameters are as follows.

4.5.1 Observations on Growth Parameters

Plant Height

The plant height from 10 selected plants was measured at 15, 30, 45, 60, 75, 90, 105 and 120 Days after Planting (DAP). The average plant height is shown in Table 4.10 and its variation is shown in Fig. 4.38. It was found that plant height has increased substantially from 15 DAP to 120 DAP. This may be due to the good microclimate conditions in the polyhouse as controlled by the developed IoT system. The average values of plant height are higher compared to the average plant height values of Bhindi in the study conducted by Fayaz *et al*., 2022.

Table 4.10 Average plant height measured at 15 days interval DAP

Days after planting	Plant height (cm)
15 DAP	14.9
30 DAP	38.1
45 DAP	96.6
60 DAP	104.6
75 DAP	125.3
90 DAP	140.5
105 DAP	179.6
120 DAP	211.8

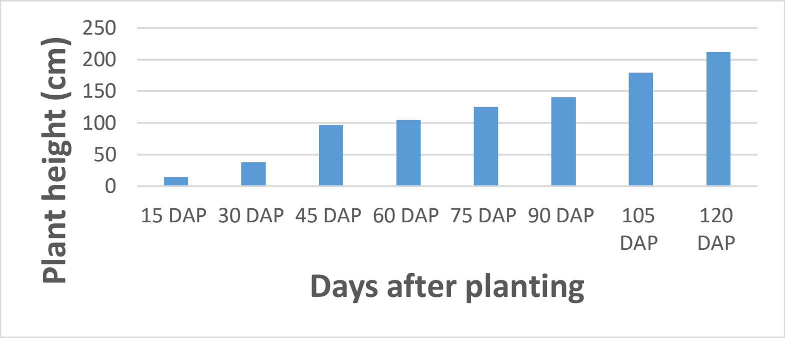


Fig. 4.38 Plant height *vs.* DAP

Number of Leaves

Number of leaves from 10 selected plants was observed at 15, 30, 45, 60, 75, 90, 105 and 120 DAP. The average number of leaves is shown in Table 4.11 and its variation is shown in Fig. 4.39. It was found that number of leaves gradually increased from 15 DAP to 120 DAP and also showed good green colour. This may be due to the controlled microclimate parameters. These values are more than the values of study conducted by Dimpka and Diepriye, 2019.

Table 4.11 Average number of leaves per plant measured at 15 days interval DAP

Days after planting (DAP)	Number of leaves per plant
15 DAP	6
30 DAP	10
45 DAP	15
60 DAP	17
75 DAP	25
90 DAP	28
105 DAP	31
120 DAP	35

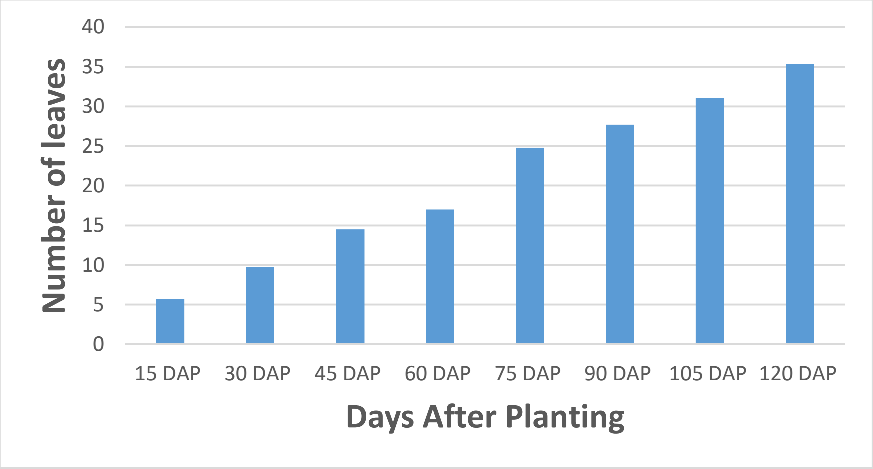


Fig. 4.39 Number of leaves *vs.* DAP

Leaf Length

Observations on length of leaves from 10 selected plants were taken at 15, 30, 45, 60, 75, 90, 105 and 120 DAP. The average values of leaf length inside the automated greenhouse is shown in Table 4.12 and its variations are shown in Fig. 4.40. Leaf length has increased gradually. This was due to the better performance of the developed IoT based system inside the polyhouse. Higher values of leaf length was measured compared to the study conducted by Mal *et al*., 2013.

Table 4.12 Leaf length measured at 15 days interval DAP

Days after planting	**Leaf length (cm)**
15 DAP	14.9
30 DAP	38.1
45 DAP	96.6
60 DAP	104.6
75 DAP	125.3
90 DAP	140.5
105 DAP	179.6
120 DAP	211.8

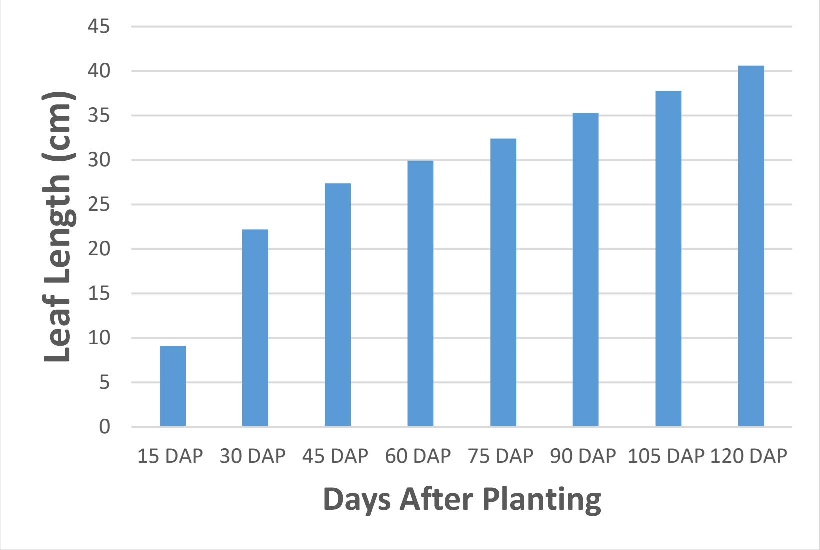


Fig. 4.40 Leaf length *vs.* DAP

Leaf Width

Observations on width of leaves from 10 selected plants were taken at 15, 30, 45, 60, 75, 90, 105 and 120 DAP. The average leaf width is shown in Table 4.13 and its variation is shown in Fig. 4.41. It was found that leaf width has increased substantially from 15 DAP to 120 DAP. This may be due to the good microclimate conditions in the polyhouse as controlled by the developed IoT system. Higher leaf width was measured compared to the study conducted by Mal *et al*., 2013.

Table 4.13 Leaf width measured at 15 days interval DAP

Days after planting	**Leaf width (cm)**
15 DAP	14.9
30 DAP	38.1
45 DAP	96.6
60 DAP	104.6
75 DAP	125.3
90 DAP	140.5
105 DAP	179.6
120 DAP	211.8

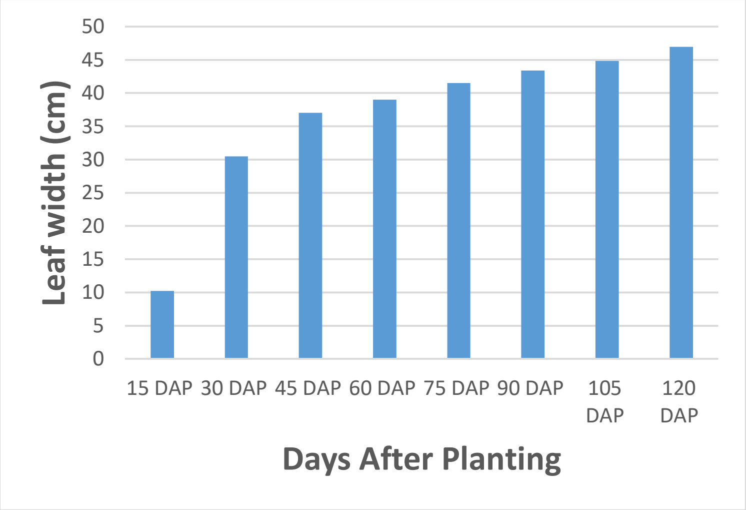


Fig. 4.41 Leaf width *vs.* DAP

Stem Girth

The stem girth was recorded at 15, 30, 45, 60, 75, 90, 105 and 120 DAP. The average stem girth is shown in Table 4.14 and its variation is shown in Fig. 4.42. Stem girth has gradually increased. This may be due to the controlled microclimate parameters inside the polyhouse. Stem girth values are higher compared to the studies done by Mahmoudi *et al*., 2020.

Table 4.14 Stem girth measured at 15 days interval DAP

Days after planting	**Stem girth (cm)**
15 DAP	2.64
30 DAP	3.59
45 DAP	4.63
60 DAP	5.205
75 DAP	5.67
90 DAP	6.19
105 DAP	6.63
120 DAP	7.25

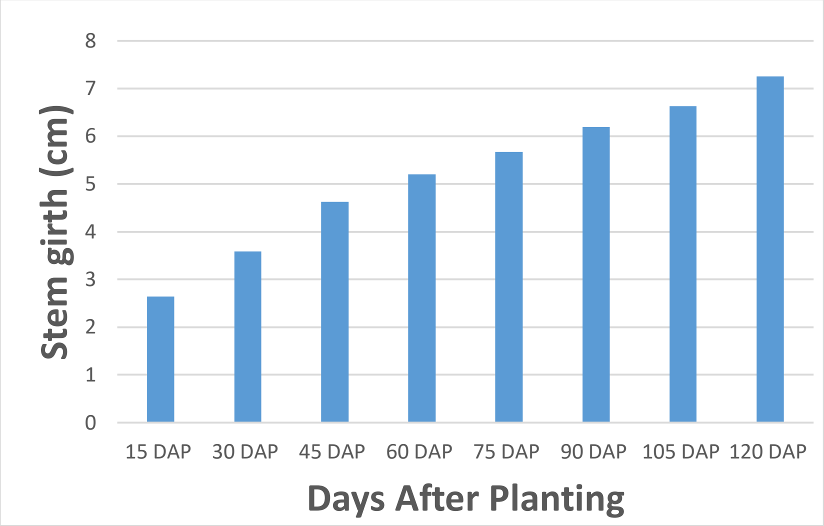


Fig. 4.42 Stem girth *vs.* DAP

Number of Primary Branches

Number of primary branches from 10 selected plants was measured at 15, 30, 45, 60, 75, 90, 105 and 120 DAP. The average values of number of primary branches per plant is shown in Table 4.15 and its variation is shown in Fig. 4.43. It was found that number of primary branches has increased substantially from 15 DAP to 120 DAP. This may be due to the good microclimate conditions in the polyhouse as controlled by the developed IoT system. Number of primary branches are more compared to the study conducted by Mal *et al*., 2013.

Table 4.15 Number of primary branches per plant measured at 15 days interval DAP

Days after planting	**Number of primary branches**
15 DAP	1
30 DAP	1
45 DAP	1
60 DAP	2
75 DAP	2
90 DAP	3
105 DAP	3
120 DAP	4

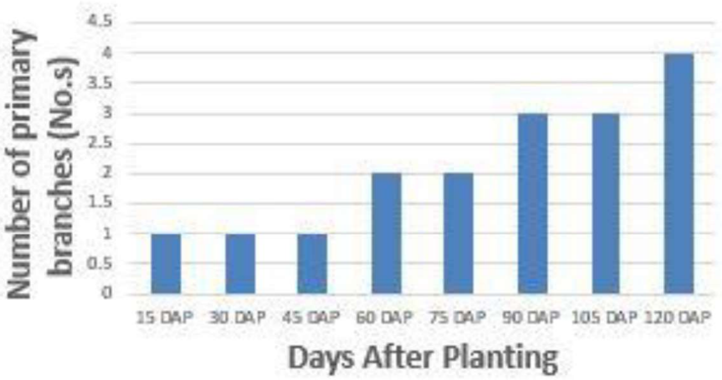


Fig. 4.43 Number of primary branches *vs.* DAP

4.5.2 Observations on Yield Parameters

The various yield parameters number of fruits per plant, fruit weight, fruit length, fruit girth and yield per plant were observed and are presented in Table 4.16. Number of fruits per plant, fruit weight, fruit length and fruit girth has increased because of the better performance and good microclimate climate control achieved by the developed IoT based automation system. Average yield per plant was also good.

Table 4.16 Observations on yield parameters

Plant	Number of fruits /Plant	Fruit weight (g)	Fruit length (cm)	Fruit girth (cm)	Yield/Plant (kg)
Plant 1	22	30.5	17.18	6.28	0.481
Plant 2	28	30.2	17.63	6.38	0.601
Plant 3	31	31	17.2	6.67	0.760
Plant 4	20	30.3	19.2	6.33	0.440
Plant 5	20	30.6	16.25	6.57	0.507
Plant 6	39	34	15.6	6.5	0.879
Plant 7	21	30.5	15.16	6.35	0.432
Plant 8	23	30.8	15.17	6.28	0.594
Plant 9	15	30.1	18.47	6.08	0.320
Plant 10	23	31	20.78	6.56	0.587
Average per plant	24	30.9	17.26 cm	6.4 cm	0.560 kg

Total Yield Per Ha

Total yield per hectare was found by multiplying yield per plant with total number of plants in one hectare and represented in tonnes per hectare (t/ha). Total yield obtained was 11 t/ha which was good and close to the average yield of the bhindi variety, ‘Anjitha’.

4.6 ENERGY CONSUMPTION

Total electricity consumed by the automation system during the entire study period measured from the energy meter was 143.2 kWh. The electricity cost for 1 kWh was taken as Rs 5. Thus, the total electricity consumed during the entire period of 135 days is only Rs 716, which is a low value. Hence, it was found that this system required only less power than any other automation system and was found to be cost-effective.

SUMMARY AND CONCLUSIONS

CHAPTER-V

SUMMARY AND CONCLUSION

The present study to develop an IoT based real-time microclimate monitoring and controlling system was conducted in a naturally ventilated poly house, Department of IDE, KCAET, Tavanur during the period December 2022 – May 2023. A web enabled microcontroller embedded system with temperature and humidity sensor, light sensor, actuators and IoT technology was developed. The developed system was able to monitor and control the microclimate parameters in real-time and can be viewed as graphical insights. Various parameters monitored were temperature, relative humidity and light. Exhaust fans and foggers were the actuators used to control temperature and RH. Upper threshold and lower threshold values of temperature were set in the code for the operation of actuators.

It was found that whenever the temperature inside the polyhouse exceeded 28°C, the controller switched ‘ON’ the exhaust fans and reduced the temperature and turned ‘OFF’ when the temperature reached below 25°C. It was also found that the actuator ‘exhaust fan’ alone was not able to reduce the temperature up to the desired level, hence fogger was also connected to the system to maintain temperature as well as RH. Hence, it was found that when the temperature exceeded 35°C, fogger automatically switched ‘ON’ and switched ‘OFF’ when the temperature reached below 32°C. The average temperature, Relative Humidity(RH) and light intensity during the test period without crop was found 34°C, 64.33% and 32,000 lx respectively, whereas the same during the crop growing period was found to be 37°C, 59.22% and 35,000 lx respectively. A reduction in temperature of 3°C and increase in relative humidity of 4% was able to achieve inside the polyhouse throughout the experiment.

Monitoring and controlling was also made possible using a GSM module where manual and automatic control was achieved using an Android mobile phone. Besides the continuous real-time data monitoring, it showed past one hour, one day, seven days and 15 days interval temperature, relative humidity and light data as graphical insights. The system was able to monitor and control both in manual and automatic mode from different locations through IoT platform. Temperature, RH and light intensity inside and outside the polyhouse were compared with and without crop after the installation of IoT based microclimate monitoring and controlling system inside the polyhouse. It was found that temperature was reduced and RH was increased inside the polyhouse by the installation of IoT based automation system. Hourly variation of temperature, RH and light intensity during crop period was monitored by IoT based system and found that temperature & light intensity increased and RH decreased during the crop growing period. The observations on growth and yield parameters of the crop were also found satisfactory. Total yield was obtained as 11 t/ha which was close to the average yield of bhindi variety ‘Anjitha’.

Hence it is concluded that

- The IoT based automation system was able to monitor and control microclimate inside a naturally ventilated polyhouse to a reasonable extent.
- The remote control of the IoT-based automation system allowed users to monitor and manage their devices and processes from anywhere with internet connectivity, offering convenience and flexibility in controlling various microclimate parameters.
- The optimum microclimate conditions could have been achieved with this system if it was a forced ventilated polyhouse with full climate control.
- The developed system with IoT applications was found to provide real-time microclimate data inside a polyhouse in smart farming.
- The monitoring and controlling of optimal conditions of crops in a protected environment using an IoT system can definitely improve the quality and yield of the crop.

Suggestions and Recommendations

- The possibility of an IoT-based automation system in a greenhouse, to achieve better resource management, increased crop yield, reduced labor costs, and improved overall productivity may be explored.
- Data fusion and Integration from multiple sensors and sources such as weather forecasts or satellite imagery to create a holistic view of the greenhouse microclimate is to be experimented to get a comprehensive understanding of the growing conditions.
- Machine learning algorithms and predictive analytics to forecast microclimate changes and optimizing the control strategies may also be researched to find optimal growing conditions.
- Introduction of advanced microprocessors like Raspberry-Pi and softwares like python may be explored for programming and interfacing process to develop a better system.
- Incorporation of carbon dioxide sensors, airflow sensors, soil moisture sensors, pH sensors, nutrient sensors and water level sensors may also be experimented to create an optimum microclimate inside the polyhouse,

REFERENCES

REFERENCES

Abd El-kader, S.M. and El-Basioni, B.M.M. 2013. Precision farming solution in Egypt using the wireless sensor network technology. *Egyptian Informatics J.* 14(3): 221-233.

Achouak, T., Khelifa, B., García-García, L., Parra-Boronat, L., Lloret, J., and Fateh, B. 2018. Sensor network proposal for greenhouse automation placed at the south of Algeria. *Network Protocols Algorithms* 10(4): 53-69.

Adams, S. R., Cockshull, K. E., and Cave, C. R. J. 2001. Effect of temperature on the growth and development of tomato fruits. *Ann. Bot.* 88(5): 869-877.

Ahonen, T., Virrankoski, R., and Elmusrati, M. 2008. Greenhouse monitoring with wireless sensor network. In: *2008 IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications,* 12-15 October 2008, Beijing [On-line]. Available: https://doi.org/10.1109/MESA.2008.4735744 [30 Dec.2008].

Akkas, M.A. and Sokullu, R. 2017. An IoT based greenhouse monitoring system with Micaz motes. *Procedia Comput. Sci.* 113: 603-608.

Al-Aubidy, K.M., Ali, M.M., Derbas, A.M., and Al-Mutairi, A.W. 2014. Real-time monitoring and intelligent control for greenhouses based on wireless sensor network. In: *2014 IEEE 11th International Multi-Conference on Systems, Signals & Devices (SSD14)*, 11-14 February 2014, Barcelona [On-line]. Available: https://doi.org/10.1109/SSD.2014.6808765 [01 May 2014].

Al-Helal, I.M. 2007. Effects of ventilation rate on the environment of a fan-pad evaporatively cooled, shaded greenhouse in extreme arid climates. *Appl. Eng. Agric*. 23(2): 221-230.

Asolkar, P.S. and Bhadade, U.S. 2014. Analyzing and predicting the green house parameters of crops. *Int. J. Comput. Appli.* 95(15): 28-39.

Asolkar, P.S. and Bhadade, U.S. 2015. An effective method of controlling the greenhouse and crop monitoring using GSM. In: *2015 International Conference on Computing Communication Control and Automation*, 26-27 February 2015, Pune [On-line]. Available: https://doi.org/10.1109/ICCUBEA.2015.47 [16 July 2015].

Atia, D.M. and El-madany H.T. 2016. Analysis and Design of Greenhouse temperature control using Adaptive Neuro-Fuzzy Inference System. *J. Electr. Syst. Inform. Technol.*133: 1-15.

Bajer, L. and Krejcar, O. 2015. Design and Realization of Low Cost Control for Greenhouse Environment with Remote Control. In: *Proceedings of the 13th IFAC Conference on Programmable Devices and Embedded Systems,* 13-15 May 2015, Cracow, Poland. [On-line]. Available: https://doi.org/10.1016/j.ifacol.2015.07.062 [26 Sep.2015].

Belsare, R., Deshmukh, K., Patil, M., and Hattarge, A.M. 2014. Smart Green House Automation. *Int. J. Comput. Sci. Eng. Technol.* 5(12):1127-1129.

Bharathi, M., Fathima,D.F., Priyanka,D., and Swetha, S. Polyhouse technique using RaspberryPi implemented in IoT. 2018. *Int. Res. J. Eng.Technol.* 5(4): 993-996.

Bhujbal, N., Vishnu, P.S., and Ratnaparkhi, M.R. 2012. Automated Green House Controller. *Int. J. Adv. Electr. Electr. Eng.* 1(1): 77-79.

Booij, P.S., Sijs, J., and Fransman, J.E. 2012. Localized climate control in greenhouses. In: 4th IFAC Nonlinear Model Predictive Control Conference. International Federation of Automatic Control, Noordwijkerhout, pp. 454-459.

Canadas, J., Molina, J.A.S., Rodriguez, F., and Aguila, I.M. 2017. Improving Automatic Climate Control with Decision Support techniques to Minimize Disease Effects in Greenhouse Tomatoes. *Inf. Processing Agric.* 4:50–63.

Chang, Y.S., Chen, Y.H., and Zhou, S.K. 2018. A smart lighting system for greenhouses based on Narrowband-IoT communication. In: *2018 13th International Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT)*, 24-26 October 2018, Taipei [On-line]. Available: https://doi.org/10.1109/IMPACT.2018.8625804 [27 Jan.2019].

Devanath, S., Kumar, H.A.R., and Shettar, R. 2019. Design and implementation of IoT based greenhouse environment monitoring and controlling system using Arduino platform. *Int. Res. J. Eng. Technol.* 6(9): 1865-1869.

Devekar, N., Raut, S., Kumbhar, H., Waidande, T., and Patil, V. 2018. Polyhouse automation using IoT & image processing. *Int. J. Innovative Res. Sci. Technol.* 5(7): 38-41.

Dhanaraju, M., Chenniappan, P., Ramalingam, K., Pazhanivelan, S., and Kaliaperumal, R. 2022. Smart farming: Internet of Things (IoT)-based sustainable agriculture. *Agric.* 12(10): 1745p.

Dhumal, Y.R. and Chitode, J.S. 2013. Green House Automation using Zigbee and Smart Phone. *Int. J. Adv. Res. Comput. Sci. Software Eng.* 3(5): 495-501.

Dimkpa, S.O.N., Tobin-West, M.D., Baraka, R.E., and Ogbowu, A.C. 2019. Field Evaluation of Some Okra (Abelmoschus esculentus (L.) Moench) Varieties in the Humid Tropics, Rivers State. *Glob. J. Agric. Res*. 7(2): 21-34.

Dinesh, M. and Saravanan, P. 2014. FPGA Based Real Time Monitoring System for Agricultural Field. *Int. J. Electr. Comput. Sci. Eng.* 1(3): 1514-1519.

Dondapati, P.P. and Rajulu, K.G. 2012. An automated multi sensored greenhouse management. *Int. J. Technol. Explorations Learning* 1(1): 21-24.

Eldhose, K.A., Rosily, A., Mini, P.K., Krishnapriya, M.N., and Neenu, M.S. 2014. Automated Greenhouse Monitoring System. *Int. J. Eng. Innovative Technol*. 3(10): 164-167.

Fayaz, Z., Masoodi, U.H., Afroza, B., Nazir, G., Wani, A.S., Jan, U., and Rashid, M. 2022. Evaluation of Growth and Yield Parameters of Okra [Abelmoschus esculentus (L.) Moench] Genotypes under Temperate Conditions of Kashmir Valley. *Env. Ecol.* 40(3A): 1368-1372.

FAO [Food and Agriculture Organization of the United Nations]. 2020. Global agriculture towards 2050. [On-line]. Available https://www.fao.org.

Ganesan, M. 1999. Effect of poly-greenhouse models on plant growth and yield of tomato. *Indian J. Agric. Sci.* 10: 586-588.

Gayatri, S.R. 2013. Greenhouse Automation System using Psoc 3. *J. Inf. Knowl. Res. Electr. Commun. Eng.* 2(2): 779-784.

Gazquez, J.C., Lopez, J. C., Baeza, E. J., Saez, M., and Parra, A. 2008. Greenhouse cooling strategies for Mediterranean climate areas. *Acta Hort.* 801: 425-432.

Gillis, A.S. 2021. What is internet of things (IoT)? [on-line]. Available: https://internetofthingsagenda.techtarget.[05 Oct.2021].

Gokul, A. 2015. Comparative evaluation of naturally ventilated greenhouse and rainshelter on the performance of cowpea. M.Tech (Agrl. Engg.) thesis, Kerala Agricultural University, Tavanur.

Halim, A.A.A., Hassan, N.M., Zakaria, A., Kamarudin, L.M., and Bakar, A.H.A. 2017. Internet of things technology for greenhouse monitoring and management system based on wireless sensor network. *ARPN J. Eng. Appl. Sci.* 11(22): 13169-13175.

Hemalatha, R., Radha, S., and Muthumeenakshi, K. 2021. *Disease Monitoring of Cucumber in Polyhouse Through IoT-Based Mobile Application* [e-book]. Artificial Intelligence and IoT-based Technologies for Sustainable Farming and Smart Agriculture. Available: https://www.igi-global.com/book/artificial-intelligence-iot-based-technologies/233938.

Jha, A. and Kumari, P. 2015. Impact of microclimatic modification on tomato quality through mulching inside and outside the polyhouse. *Agric. Sci. Digest* 35(3): 178-182.

Jamaludin, D., Ahmad, D., Kamaruddin, R., and Jaafar, H.Z.E. 2014. Microclimate inside a tropical greenhouse equipped with evaporative cooling pads. *J. Sci. Technol.* 22(1): 255-271.

Jinu, A. and Hakkim, A.V.M. 2016. Automatic Microclimate Control in Greenhouses. *Int. J. Eng. Sci. Comput.* 6(8): 2941-2946.

Joteppagol, S.M. and Kore, S.K. 2015. Greenhouse Automation Using CAN Bus. *Int. J. Res. Emerging Sci. Technol.* 2(5): 277-281.

Kagan, C.R., Arnold, D.P., Cappelleri, D.J., Keske, C.M., and Turner, K.T. 2022. Special report: The internet of things for precision agriculture (IoT4Ag). *Comput. Electr. Agric.* 196(1): 106742p.

Kavitha, E.R., Vijayaraghavan, N., and Tajuddin. A. 2003. Response of Vegetable Crops in a Solar aided Poly-house Ecosystem. *J. Inst. Eng. India* 84: 52-55.

Kerala Agricultural University 2016. ***Package of practices Recommendations: Crops*** 15th edition. Kerala Agricultural University, Thrissur-392p.

Khandelwal, S.A. 2012. Automated greenhouse management using GSM modem. *Int. J. Comput. Sci. Inf. Technol.* 3(1): 3099-3102.

Kitpo, N., Kugai, Y., Inoue, M., Yokemura, T., and Satomura, S. 2019. Internet of things for greenhouse monitoring system using deep learning and bot notification services. In: *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 11-13 January 2019, Las Vegas [On-line]. Available: https://doi.org/10.1109/ICCE.2019.8661999 [07 March 2019].

Kittas, C., Bartzanas, T., and Jaffrin, A. 2001. Greenhouse evaporative cooling: measurement and data analysis. *Trans. ASAE.* 44(3): 683-689.

Kolhe, S.A. and Annadate, S.A. 2012. Implementation of Green House Automation using ARM7 Controller. *Int. J. Comput. Appl.* 47(20): 1-5.

Kumar, A., Singh, V., Kumar, S., Jaiswal, S.P., and Bhadoria, V.S. 2022. IoT enabled system to monitor and control greenhouse. *Materials Today: Proceedings* 49: 3137-3141.

Li, A., Huang, L., and Zhang, T. 2017. Field test and analysis of microclimate in naturally ventilated single-sloped greenhouses*.* *Energy Buildings* 138: 479-489.

Liu, Q., Jin, D., Shen, J., Fu, Z., and Linge, N. 2016. A WSN-based prediction model of microclimate in a greenhouse using extreme learning approaches. *In: 2016 18th International Conference on Advanced Communication Technology*, 31 January - 03 February 2016, PyeongChang [On-line]. Available: https://doi.org/10.1109/ICACT.2016.7423609 [03 March 2016].

Mahmoudi, M., Khelil, M.N., Ghrib, R., Douh, B., and Boujelben, A. 2020. Assessment of growth and yield of okra (Abelmoschus esculentus) under surface and subsurface drip irrigation using treated waste water. *Int. J. Recycl. Org. Waste Agric.* 9(4): 349-356.

Mal, B., Mahapatra, P., Mohanty, S., and Mishra, H.N. 2013. Growth and yield parameters of okra (Abelmoschus esculentus) influenced by Diazotrophs and chemical fertilizers. *J. Crop weed* 9(2): 109-112.

Maraveas, C., Piromalis, D., Arvanitis, K.G., Bartzanas, T., and Loukatos, D. 2022. Applications of IoT for optimized greenhouse environment and resources management. *Comput. Electr. Agric.* 198(1): 106993p.

Martinovic, G. and Simon, J. 2014. Greenhouse microclimatic environment controlled by a mobile measuring station. *Njas-wageningen J. Life Sci*. 70: 61-70.

Mekki, M., Abdallah, O., Amin, M.B., Eltayeb, M., Abdalfatah, T., and Babiker, A. 2015. Greenhouse monitoring and control system based on wireless sensor network. In: *2015 international conference on computing, control, networking, electronics and embedded systems engineering (ICCNEEE)*, 07-09 September 2015, Khartoum [On-line]. Available: https://doi.org/10.1109/ICCNEEE.2015.7381396 [14 Jan. 2016].

Mohanraj, I., askumar, K., and Naren, J. 2016. Field monitoring and automation using IoT in agriculture domain. *Procedia Comput. Sci.* 93: 931-939.

Mohanty, N.R. and Patil, C.Y. 2013. Wireless sensor networks design for greenhouse automation. *Int. J. Eng. Innovative Technol.* 3(2): 257-262.

Muangprathub, J., Boonnama, N., Kajornkasirata, S., Lekbangponga, N., Wanichsombata, A., and Nillaorb, P. 2019. IoT and agriculture data analysis for smart farm. *Comput. Electr. Agric.* 156: 467-474.

Pandey, A.K. and Chauhan, M. 2019. IoT based smart polyhouse system using data analysis. In: *2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, 27-28 September 2019, Ghaziabad [On-line]. Available: https://doi.org/10.1109/ICICT46931.2019.8977665 [03 Feb.2020].

Parvej, M.R., Khan, M.A.H., and Awal, M.A. 2010. Phenological development and production potentials of tomato under greenhouse climate. *J. Agric. Sci.* 5(1): 19-31.

Parvez, Z.S., Ahmed, F.J.S, Parvez, S.S., and Gadilingappa, K. 2014. Arm Based Automated Wireless Greenhouse Climate Management System using Zigbee Technology. *IOSR J. Electr. Commun. Eng.* 9(3): 55-62.

Pavithran, A.K. and Krishnan, H. 2018. Comparative evaluation of evapotranspiration parameters of Okra in polyhouse for alternate growing media. B. Tech (Agrl. Eng.), Kerala Agricultural University, Thrissur, 27p.

Poyen, F., Bhattacharya, K., Mondal, A., Ghosh, A., and Bandyopadhyay, R. 2014. Sensor Based Automated Shading of Green House. In: International conference on Innovative Engineering Technologies, 28-29 December 2014, Bangkok [On-line]. Available: http://dx.doi.org/10.15242/IIE.E1214047.

Qiu, W., Dong, L., Wang, F., and Yan, H. 2014. Design of intelligent greenhouse environment monitoring system based on ZigBee and embedded technology. In: *2014 IEEE international conference on consumer electronics*, 09-13 April 2014, Shenzhen [On-line]. Available: https://doi.org/10.1109/ICCE-China.2014.7029857 [05 Feb.2015].

Raja, G. and Rajarathinam, D.R.P. 2018. Smart polyhouse farming using IoT environment. *Int. J. Trend Sci. Res. Dev.* 2(3): 2456-6740.

Rangan, K. and Vigneswaran, T. 2010. An embedded systems approach to monitor green house. In: *Recent Advances in Space Technology Services and Climate Change 2010 (RSTS & CC-2010)*, 13-15 November 2010, Chennai [On-line]. Available: https://doi.org/10.1109/RSTSCC.2010.5712800 [14 Feb.2011].

Riahi, J., Vergura, S., Mezghani, D., and Mami, A. 2020. Intelligent control of the microclimate of an agricultural greenhouse powered by a supporting PV system. *Appl. Sci.* 10(4): 1350p.

Salleh, A., Ismail, M.K., Mohamad, N.R., Abd Aziz, M.Z.A., Othman, M.A., and Misran, M.H. 2013. Development of Greenhouse Monitoring using Wireless Sensor Network through ZigBee Technology. *Int. J. Eng. Sci. Invention* 2(7): 6-12.

Shamshiri, R. 2017. Measuring optimality degrees of microclimate parameters in protected cultivation of tomato under tropical climate condition. *Measurement* 31: 1-9.

Sonawane, Y.R., Khandekar, S., Mishra, B.K., and Pandian, K.S. 2008. Environment monitoring and control of a polyhouse farm through internet. *World Bank: India Country Overview 2008*, pp. 1-6.

Tanappagol, R. and Kondikopp, S.V. 2017. IoT Based Energy Efficient Environmental Monitoring Alerting and Controlling System. *Int. J. Latest Technol. Eng. Manag. Appl. Sci.* 6(7): 83-86.

Thenmozhi, S., Dhivya M.M., Sudharsan, R., and Nirmalakumari, K. 2014. Greenhouse Management Using Embedded System and Zigbee Technology. *Int. J. Adv. Res. Electr. Electr. Instrumentation Eng*. 3(2): 7382-7389.

Umesha, B. and Reddy, M. 2011. Effect of weather parameters on growth and yield parameters of tomato under natural greenhouse. *Indian J. Nat. Sci.* 2(9): 654-662.

Unsal, E., Yelkuvan, A.F., Tasbasil, M., and Sonmex, O. 2018. A new greenhouse monitoring system based on Internet of Things technology. *Int. Conf. Innovative Eng. Appl.* 1(1): 1-6.

Vidyasagar, B. 2012. Green House Monitoring and Automation using GSM. *Int. J. Sci. Res. Publ.* 2(5): 1-5.

Vishwakarma, A., Sahu, A., Sheikh, N., Payasi, P., Rajput, S.K., and Srivastava, L. 2020. IoT based Greenhouse monitoring and controlling system. In: *2020 IEEE Students Conference on Engineering & Systems (SCES)*, 10-12 July 2020, Prayagraj [On-line]. Available: https://doi.org/10.1109/SCES50439.2020.9236693 [04 Nov.2020].

Wang, J., Chen, M., Zhou, J., and Li, P. 2020. Data communication mechanism for greenhouse environment monitoring and control: An agent-based IoT system. *Inf. Processing Agric.* 7(3): 444-455.

Wang, J., Zhou, J., Gu, R., Chen, M., and Li, P. 2018. Manage system for internet of things of greenhouse based on GWT. *Inf. Processing Agric.* 5(2): 269-278.

Waykole, U.A. and Agrawal, D.G. 2012. Greenhouse Automation System. In: *1st International Conference on Recent Trends in Engineering & Technology.*

Special Issue of International Journal of electronics, Communication & Soft Computing Science &Engineering March 2012.

Zafar, S., Miraj, G., Baloch, R., Murataza, D., and Arshad, K. 2018. An IoT based real time environmental monitoring system using Arduino and cloud service. *Eng. Technol. Appl. Sci. Res.* 8(4): 3238–3242.

Zheng, X.X., Li, L.R., and Shao, Y.J. 2016. A GSM-based remote temperature and humidity monitoring system for granary. In: *MATEC Web of Conferences*, 2016, China [On-line]. Available: https://doi.org/10.1051/matecconf/20164401060 [08 March 2016].

Zou, Z., Bie, Y., and Zhou, M. 2018. Design of an intelligent control system for greenhouse. In: *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 25-27 May 2018, Xian [On-line]. Available: https://doi.org/10.1109/IMCEC.2018.8469309 [23 Sept.2018].

APPENDICES

APPENDIX-I

Validation of sensor reading with measured reading

Table A-1 Validation of sensor reading with measured reading for temperature on 10/01/2023

Time	Observed temperature (°C)	Sensor reading(°C)	Difference(°C)	Deviation %
8:00 AM	21	20.08	-0.92	-0.04
9:00 AM	27	26	-1	-0.03
10:00 AM	30	29.55	-0.45	-0.01
11:00 AM	32	32.2	0.2	0.006
12:00 AM	33	32.91	-0.09	-0.002
1:00 PM	32	32	0	0
2:00 PM	33	33.01	0.01	0.0003
3:00 PM	34	33.83	-0.17	-0.005
4:00 PM	32	32.8	0.8	0.025
5:00 PM	29	28.94	-0.06	-0.002

Table A-2 Validation of sensor reading with measured reading for RH on 10/01/2023

Time	Observed RH (%)	Sensor reading(%)	Difference(%)	Deviation %
8:00 AM	89	84.69	-0.26	-0.29
9:00 AM	73	73.1	-0.9	-1.23
10:00 AM	65	66.61	0.66	1.01
11:00 AM	55	54.36	0.8	1.45
12:00 AM	50	52.27	-0.59	-1.18
1:00 PM	49	52	3	6.12
2:00 PM	47	47.23	0.23	0.48
3:00 PM	51	52.34	1.34	2.62
4:00 PM	55	55.22	0.22	0.4
5:00 PM	62.8	64.81	2.01	3.2

Table A-3 Validation of sensor reading with measured reading for light intensity on 10/01/2023

Time	Observed light intensity (lx)	Sensor reading(lx)	Difference(lx)	Deviation %
8:00 AM	3440	3440	0	0
9:00 AM	13630	13633.33	3.33	$3X10^{-4}$
10:00 AM	23270	23268.33	-1.67	$-1X10^{-4}$
11:00 AM	26280	26279.17	-0.83	$-3X10^{-5}$
12:00 AM	21717	21716.67	-0.33	$-1.7X10^{-5}$
1:00 PM	34260	34260	0	0
2:00 PM	35500	35500	0	0
3:00 PM	25870	25868.33	-1.67	$-1X10^{-4}$
4:00 PM	25574	25573	-1	$-1X10^{-4}$
5:00 PM	4330	4328.33	-1.67	$-5X10^{-4}$

Table A-4 Validation of sensor reading with measured reading for temperature on 11/01/2023

Time	Observed temperature (°C)	Sensor reading(°C)	Difference(°C)	Deviation %
8:00 AM	21	20.08	-0.92	-0.04
9:00 AM	27	26	-1	-0.03
10:00 AM	30	29.55	-0.45	-0.01
11:00 AM	32	32.2	0.2	0.006
12:00 AM	33	32.91	-0.09	-0.002
1:00 PM	32.1	32	-0.1	-0.003
2:00 PM	33	33.01	0.01	0.0003
3:00 PM	34	33.83	-0.17	-0.005
4:00 PM	32	32.8	0.8	0.025
5:00 PM	29	28.94	-0.06	-0.002

Table A-5 Validation of sensor reading with measured reading for RH on 11/01/2023

Time	Observed RH (%)	Sensor reading(%)	Difference(%)	Deviation %
8:00 AM	89	84.69	-4.31	-0.04
9:00 AM	73	73.81	0.81	0.01
10:00 AM	65	66.61	1.61	0.02
11:00 AM	55	54.36	-0.64	-0.01
12:00 AM	50	52.27	2.27	0.04
1:00 PM	49	52	3	0.06
2:00 PM	47	47.23	0.23	0.004
3:00 PM	51	52.34	1.34	0.02
4:00 PM	55	55.22	0.22	0.004
5:00 PM	62.8	64.81	2.01	0.03

Table A-6 Validation of sensor reading with measured reading for light intensity on 11/01/2023

Time	Observed light intensity (lx)	Sensor reading(lx)	Difference(lx)	Deviation %
8:00 AM	3440	3440	0	0
9:00 AM	13630	13633.33	3.33	0.0003
10:00 AM	23270	23268.33	-1.67	-0.0001
11:00 AM	26280	26279.17	-0.83	$-3.9X10^{-5}$
12:00 AM	21717	21716.67	-0.33	$-1.7X10^{-5}$
1:00 PM	34260	34260	0	0
2:00 PM	35500	35500	0	0
3:00 PM	25870	25868.33	-1.67	-0.0001
4:00 PM	25574	25573	-1	-0.0001
5:00 PM	4330	4328.33	-1.67	-0.0005

Table A-7 Validation of sensor reading with measured reading for temperature on 12/01/2023

Time	Observed temperature (°C)	Sensor reading(°C)	Difference(°C)	Deviation %
8:00 AM	22	20.69	-1.31	-0.05
9:00 AM	26	25.58	-0.42	-0.01
10:00 AM	30	29.04	-0.96	-0.03
11:00 AM	32	31.89	-0.11	-0.003
12:00 AM	34.5	34.54	0.04	0.001
1:00 PM	35.1	35.05	-0.05	-0.001
2:00 PM	32.1	32.09	-0.01	-0.0003
3:00 PM	34.5	34.23	-0.27	-0.007
4:00 PM	32.5	32.09	-0.41	-0.01
5:00 PM	29.2	29.85	0.65	0.02

Table A-8 Validation of sensor reading with measured reading for RH on 12/01/2023

Time	Observed RH (%)	Sensor reading(%)	Difference(%)	Deviation %
8:00 AM	87	86.83	-0.17	-0.001
9:00 AM	84	82.31	-1.69	-0.02
10:00 AM	68	68.5	0.5	0.007
11:00 AM	64	64.09	0.09	0.001
12:00 AM	64.25	52.49	-11.76	-0.18
1:00 PM	49.1	49.75	0.65	0.01
2:00 PM	67	67.2	0.2	0.002
3:00 PM	51.5	52.92	1.42	0.02
4:00 PM	55.25	54.87	-0.38	-0.006
5:00 PM	61.2	61.85	0.65	0.01

Table A-9 Validation of sensor reading with measured reading for light intensity on 12/01/2023

Time	Observed light intensity (lx)	Sensor reading(lx)	Difference(lx)	Deviation %
8:00 AM	4116	4115.83	-0.17	$-5.4X10^{-5}$
9:00 AM	15725	15725	0	0
10:00 AM	26952	26951.67	-0.33	$-2.3X10^{-5}$
11:00 AM	34415	34414.17	-0.83	$-4.06X10^{-5}$
12:00 AM	36675	36674.17	-0.83	$-4.6X10^{-5}$
1:00 PM	36254	36253.33	-0.67	$-4.3X10^{-5}$
2:00 PM	29172	29171.67	-0.33	$-2.3X10^{-5}$
3:00 PM	30197	30196.67	-0.33	$-3.3X10^{-5}$
4:00 PM	15668	15667.5	-0.5	$-7.4X10^{-5}$
5:00 PM	7320	7319.17	-0.83	-0.0002

Table A-10 Validation of sensor reading with measured reading for temperature on 13/01/2023

Time	Observed temperature (°C)	Sensor reading(°C)	Difference(°C)	Deviation %
8:00 AM	21	21.1	0.1	0.004
9:00 AM	23.5	23.32	-0.18	-0.007
10:00 AM	24	24.05	0.05	0.002
11:00 AM	27	27.31	0.31	0.01
12:00 AM	29.5	29.35	-0.15	-0.005
1:00 PM	30	29.85	-0.15	-0.005
2:00 PM	32.8	32.4	-0.4	-0.01
3:00 PM	31.5	31.08	-0.42	-0.01
4:00 PM	29.5	29.24	-0.26	-0.008
5:00 PM	28	27.31	-0.69	-0.02

Table A-11 Validation of sensor reading with measured reading for RH on 13/01/2023

Time	Observed RH (%)	Sensor reading(%)	Difference(%)	Deviation %
8:00 AM	82	84.83	2.83	0.03
9:00 AM	79	80.2	1.2	0.01
10:00 AM	75	75.3	0.3	0.004
11:00 AM	70	70.5	0.5	0.007
12:00 AM	74.5	74.6	0.1	0.001
1:00 PM	71	70.71	-0.29	-0.004
2:00 PM	67.4	66.68	-0.72	-0.01
3:00 PM	63.5	63.8	0.3	0.004
4:00 PM	71.5	71.93	0.43	0.006
5:00 PM	77	79.43	2.43	0.03

Table A-12 Validation of sensor reading with measured reading for light intensity on 13/01/2023

Time	Observed light intensity (lx)	Sensor reading(lx)	Difference(lx)	Deviation %
8:00 AM	3265	3265	0	0
9:00 AM	5830	5830	0	0
10:00 AM	9273	9272.5	-0.5	-9.4X10^{-5}
11:00 AM	18640	18639.17	-0.83	-9.6X10^{-5}
12:00 AM	20805	20805	0	0
1:00 PM	30003	30002.5	-0.5	-4.9X10^{-5}
2:00 PM	30838	30838	0	0
3:00 PM	8778	8777.5	-0.5	-7.3X10^{-5}
4:00 PM	2642	2641.67	-0.33	-0.0002
5:00 PM	1594	1593.33	-0.67	-0.0004

Table A-13 Validation of sensor reading with measured reading for temperature on 14/01/2023

Time	Observed temperature (°C)	Sensor reading(°C)	Difference(°C)	Deviation %
8:00 AM	21	21	0	0
9:00 AM	24	23.03	-0.97	-0.04
10:00 AM	26	26.39	0.39	0.01
11:00 AM	31	30.57	-0.43	-0.01
12:00 AM	32.3	32.3	0	0
1:00 PM	34.4	34.44	0.04	0.001
2:00 PM	35.5	35.05	-0.45	-0.01
3:00 PM	35	30.87	-4.13	-0.1
4:00 PM	31	31.08	0.08	0.002
5:00 PM	29	28.94	-0.06	-0.002

Table A-14 Validation of sensor reading with measured reading for RH on 14/01/2023

Time	Observed RH (%)	Sensor reading(%)	Difference(%)	Deviation %
8:00 AM	86	85.83	-0.17	-0.001
9:00 AM	83	84.83	1.83	0.022
10:00 AM	76	76.01	0.01	0.0001
11:00 AM	69	70.01	1.01	0.01
12:00 AM	67.15	67.25	0.1	0.001
1:00 PM	59.4	59.62	0.22	0.003
2:00 PM	52.5	53.64	1.14	0.02
3:00 PM	62.5	63.94	1.44	0.02
4:00 PM	67.5	65.5	-2	-0.02
5:00 PM	62.8	64.81	2.01	0.03

Table A-15 Validation of sensor reading with measured reading for light intensity on 14/01/2023

Time	Observed light intensity (lx)	Sensor reading(lx)	Difference(lx)	Deviation %
8:00 AM	2654	2654	0	0
9:00 AM	8914	8914	0	0
10:00 AM	17885	17884.17	-0.83	$-8.3X10^{-5}$
11:00 AM	23217	23216.67	-0.33	$-2.03X10^{-5}$
12:00 AM	35942	35941.67	-0.33	$-1.8X10^{-5}$
1:00 PM	34055	34055	0	0
2:00 PM	28907	28906.67	-0.33	$-2.07X10^{-5}$
3:00 PM	9700	9700	0	0
4:00 PM	2640	2640	0	0
5:00 PM	4122	4120	-2	-0.0006

Table A-16 Validation of sensor reading with measured reading for temperature on 15/01/2023

Time	Observed temperature (°C)	Sensor reading(°C)	Difference(°C)	Deviation %
8:00 AM	22	23.74	1.74	0.07
9:00 AM	28	29.14	1.14	0.04
10:00 AM	30	29.24	-0.76	-0.02
11:00 AM	32	32.5	0.5	0.015
12:00 AM	33	33	0	0
1:00 PM	34	33.93	-0.07	-0.002
2:00 PM	34.8	34.84	0.04	0.001
3:00 PM	34	33.8	-0.2	-0.005
4:00 PM	32	31.59	-0.41	-0.01
5:00 PM	30	29.65	-0.35	-0.01

Table A-17 Validation of sensor reading with measured reading for RH on 15/01/2023

Time	Observed RH (%)	Sensor reading(%)	Difference(%)	Deviation %
8:00 AM	82	84.83	2.83	0.03
9:00 AM	74	74.96	0.96	0.01
10:00 AM	75	74.75	-0.25	-0.003
11:00 AM	67	66.39	-0.61	-0.009
12:00 AM	67.5	66.39	-1.11	-0.01
1:00 PM	65	64.73	-0.27	-0.004
2:00 PM	56.8	58.61	1.81	0.03
3:00 PM	65	65.13	0.13	0.002
4:00 PM	67	66.17	-0.83	-0.01
5:00 PM	75	74.82	-0.18	-0.002

Table A-18 Validation of sensor reading with measured reading for light intensity on 15/01/2023

Time	Observed light intensity (lx)	Sensor reading(lx)	Difference(lx)	Deviation %
8:00 AM	6116	6115.83	-0.17	$-5.4X10^{-5}$
9:00 AM	15500	15500	0	0
10:00 AM	20950	20955	5	0.0003
11:00 AM	30650	30650	0	0
12:00 AM	33960	33960	0	0
1:00 PM	33530	33528.33	-1.67	$-9.01X10^{-5}$
2:00 PM	32993.33	32994	0.67	$3.9X10^{-5}$
3:00 PM	29840	29840	0	0
4:00 PM	19526	19526	0	0
5:00 PM	5603	5602.5	-0.5	-0.0001

APPENDIX-II

Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse during the period (09/01/23 to 15/01/23)

Table A-19: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 09/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	24	26	79	76	2830	8600
9:00 AM	28	31	74	66	7280	19780
10:00 AM	30	34	68	51	11464	35600
11:00 AM	32	36	61	48	16855	51200
12:00 PM	34	38.5	59	40.5	22960	56700
1:00 PM	34	38	54	50.5	17565	61300
2:00 PM	33.5	36	50.5	43	10810	24400
3:00 PM	35	38	47	40	9260	38000
4:00 PM	34	36	51	43	3550	11640
5:00 PM	31	32.5	63	61.25	3117	11120

Table A-20: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 10/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	21	23	89	75	3440	6480
9:00 AM	27	29	73	71	13630	26800
10:00 AM	30	34	65	48	23270	46200
11:00 AM	32	37	55	39	26280	54500
12:00 PM	33	38.5	50	33.7	21717	62000
1:00 PM	32	39	49	33	34260	67400
2:00 PM	33	39	47	29	35500	69000
3:00 PM	34	38	51	42.5	25870	51800
4:00 PM	32	36	55	43	25574	49200
5:00 PM	29	31.5	62.8	56.3	4330	9840

Table A-21: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 11/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	21	23	89	75	2440	6480
9:00 AM	27	29	73	71	8630	26800
10:00 AM	30	34	65	48	14268	46200
11:00 AM	32	37	55	39	21280	54500
12:00 PM	33	38.5	50	33.7	18715	62000
1:00 PM	32.1	39	49	33	15260	67400
2:00 PM	33	39	47	29	12600	69000
3:00 PM	34	38	51	42.5	11850	51800
4:00 PM	32	36	55	43	7572.5	49200
5:00 PM	29	31.5	62.8	56.3	3122	9840

Table A-22: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 12/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	22	23	87	79	4116	7270
9:00 AM	26	28	84	64	15725	30700
10:00 AM	30	32.5	68	49.5	26952	50000
11:00 AM	32	38.5	64	24.75	34415	68400
12:00 PM	34.5	37	64.25	39	36675	70000
1:00 PM	35.1	37	49.1	35	36254	70500
2:00 PM	32.1	35	67	42	29172	59700
3:00 PM	34.5	36	51.5	43	30197	60000
4:00 PM	32.5	34.5	55.25	46.5	15668	31500
5:00 PM	29.2	23.7	61.2	49.5	7320	15350

Table A-23: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 13/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	21	23	82	75	3265	6490
9:00 AM	23.5	24	79	71	5830	9430
10:00 AM	24	27.5	75	59.5	9273	16340
11:00 AM	27	30	70	59	18640	28600
12:00 PM	29.5	32	74.5	53.8	20805	38000
1:00 PM	30	33	71	48	30003	38500
2:00 PM	32.8	34.5	67.4	43.5	30838	37700
3:00 PM	31.5	33.5	63.5	47.5	8778	16660
4:00 PM	29.5	32	71.5	49	2642	5260
5:00 PM	28	29	77	64	1594	3320

Table A-24: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 14/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	21	23	86	83	2654	5210
9:00 AM	24	26	83	69	8914	14340
10:00 AM	26	29.5	76	58.5	17885	33100
11:00 AM	31	32.5	69	52.5	23217	43800
12:00 PM	32.3	37	67.15	41.5	35942	63500
1:00 PM	34.4	36.5	59.4	38.5	34055	63000
2:00 PM	35.5	37	52.5	39	28907	56900
3:00 PM	35	38.5	62.5	36.5	9700	17000
4:00 PM	31	33.5	67.5	50.5	2640	5500
5:00 PM	29	31.5	62.8	56.3	4122	9840

Table A-25: Comparison of temperature, Relative Humidity (RH) and light intensity inside and outside the polyhouse on 15/01/23

Time	Inside temperature (°C)	Outside temperature (°C)	Inside RH (%)	Outside RH (%)	Inside light intensity (lx)	Outside light intensity (lx)
8:00 AM	22	24	82	75	6116	9060
9:00 AM	28	29	74	64	15500	29600
10:00 AM	30	35	75	49	20950	41800
11:00 AM	32	37	67	46	30650	61200
12:00 PM	33	38	67.5	45	33960	67400
1:00 PM	34	37	65	39	33530	67300
2:00 PM	34.8	37.5	56.8	44.5	32993.33	53100
3:00 PM	34	34.5	65	47.7	29840	42700
4:00 PM	32	36	67	48	19526	34600
5:00 PM	30	32.5	75	55.25	5603	10690

AN IoT BASED REAL-TIME MICROCLIMATE MONITORING AND CONTROLLING SYSTEM FOR GREENHOUSE

by

ANGITHA K A

(2020-18-004)

ABSTRACT OF THESIS

Submitted in partial fulfilment of the requirement for the award of degree of

MASTER OF TECHNOLOGY

IN

AGRICULTURAL ENGINEERING

(Soil and Water Conservation Engineering)

Faculty of Agricultural Engineering & Technology

Kerala Agricultural University

DEPARTMENT OF IRRIGATION AND DRAINAGE ENGINEERING
KELAPPAJI COLLEGE OF AGRICULTURAL ENGINEERING AND TECHNOLOGY
TAVANUR - 679573, MALAPPURAM
KERALA, INDIA
2023

ABSTRACT

In recent years, the agricultural industry has witnessed a significant transformation due to the advent of Internet of Things (IoT) technology. IoT has revolutionized the agriculture with real-time monitoring and controlling systems. Though, greenhouse provides a controlled environment for cultivating crops, maintaining optimal microclimatic conditions inside the greenhouse in real-time is crucial for maximizing yield and quality of produce. Hence, a study was conducted to develop a web enabled microcontroller embedded system with sensors and IoT technology for greenhouse, to monitor and control the various microclimate parameters in real-time. The study was conducted in a naturally ventilated polyhouse. The web enabled system consists of microcontroller, temperature & humidity sensor, light sensor and actuators (exhaust fans and foggers). The developed system was evaluated with and without crop inside polyhouse.

The developed system was able to monitor and control the microclimate parameters in real-time, both in manual and automatic mode through IoT platform anywhere in the world. It was found that, whenever the temperature inside the polyhouse exceeded 28°C, the controller switched ‘ON’ the exhaust fans and reduced the temperature and turned ‘OFF’ when the temperature reached below 25°C. It was noticed that the actuator, exhaust fan alone was not able to reduce the temperature up to the desired level. Hence, fogger was also connected to the system to maintain temperature as well as relative humidity (RH). It was found that when the temperature exceeded 35°C, fogger automatically switched ‘ON’ and switched ‘OFF’ when the temperature reached below 32°C. The average temperature, RH and light intensity during the test period (09/01/2023 to 15/01/2023) without crop was found 34°C, 64.33% and 32,000 lx respectively, whereas the same during the crop growing period (20/01/2023 to 21/05/2023) was found to be 37°C, 59.22% and 35,000 lx respectively. A reduction in temperature of 3°C and increase in RH of 4% was able to achieve inside the polyhouse throughout the experiment.

Monitoring and controlling was also made possible using a GSM module where manual and automatic control was achieved using an Android mobile phone.

Besides the continuous real-time data monitoring, it showed past one hour, one day, seven days and 15 days interval temperature, RH and light data as graphical insights. The system was able to monitor and control both in manual and automatic mode using GSM from different locations through SMS.

Temperature, RH and light intensity inside and outside the polyhouse were compared with and without crop after the installation of IoT based system inside the polyhouse. Usually, temperature inside the polyhouse is higher and RH is less compared to outside of polyhouse. After the installation of IoT based automation system, it was found that there was lower temperature and higher RH inside the polyhouse than outside. The observations on growth and yield parameters of the crop were also found satisfactory. Crop yield of 11 t/ha, which is approximate to the average yield of bhindi for ‘Anjitha’ variety was obtained.