AUTOMATED RAINWATER HARVESTING SYSTEM

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
OF THE DEGREE OF

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

BY

Ismail Pawaskar

Deepak Ramchandani

Taman Poojary

UNDER THE GUIDANCE OF

PROF. SUVARNA ARANJO

(Department of Information Technology)

DEPARTMENT OF INFORMATION TECHNOLOGY

XAVIER INSTITUTE OF ENGINEERING

UNIVERSITY OF MUMBAI

2019 – 2020

XAVIER INSTITUTE OF ENGINEERING

MAHIM CAUSEWAY, MAHIM, MUMBAI - 400016.

CERTIFICATE

This to certify that

Ismail Pawaskar	(XIEIT171840)
Deepak Ramchandani	(XIEIT171848)
Taman Poojary	(XIEIT171842)

Have satisfactorily carried out the SNL MINI-PROJECT work titled **"Automated Rainwater Harvesting System"** in partial fulfillment of the degree of Bachelor of Engineering as laid down by the University of Mumbai during the academic year 2019-2020.

Prof. Suvarna Aranjo

Internal Examiner **External Examiner**

Date:

Place: MAHIM, MUMBAI

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' Ideas or words have been included, I have adequately cited and referenced the original sources.

I also declare that I have adhered to all the principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission.

I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which thus have not been properly cited or from whom proper permission have not been taken when needed.

Ismail Pawaskar (XIEIT171840) ----------

Deepak Ramchandani (XIEIT171848) ----------

Taman Poojary (XIEIT171842) ----------

Date:

TABLE OF CONTENTS

LIST OF FIGURES

Acknowledgement

We would like to thank Fr. (Dr.) John Rose S.J. (Director of XIE) for providing us with such an environment so as to achieve goals of our project and supporting us constantly.

We express our sincere gratitude to our Honorable Principal Mr. Y.D.Venkatesh for encouragement and facilities provided to us.

We would like to place on record our deep sense of gratitude to Prof. Chhaya Narvekar, Head of Dept Of Information Technology, Xavier Institute of Engineering, Mahim, Mumbai, for her generous guidance help and useful suggestions.

With deep sense of gratitude we acknowledge the guidance of our project guide Prof. Suvarna Aranjo. The time-to-time assistance and encouragement by her has played an important role in the development of our project.

We would also like to thank our entire Information Technology staff who have willingly cooperated with us in resolving our queries and providing us all the required facilities on time.

Ismail Pawaskar ----------

Deepak Ramchandani ----------

Taman Poojary ----------

Subject: Sensor Network Mini-projects Lab

Lab Outcomes

At the end of the course student will be able to:		Bloom Level
ITL 504-1 (LO1)	Identify the requirements for the real world problems.	Understand
ITL 504-2 (LO2)	Conduct a survey of several available literatures in the preferred field of study.	Understand
ITL 504-3 (LO3)	Study software/ hardware skills and build the project successfully by hardware requirements, coding, emulating and testing.	Apply, Implement
ITL 504-4 (LO4)	To report and present the findings of the study conducted in the preferred domain	Apply, Implement
ITL 504-5 (LO5)	Demonstrate an ability to work in teams and manage the conduct of the research study.	Apply

XAVIER INSTITUTE OF ENGINEERING

Department of Information Technology

Class/ Sem/ A.Y: TE IT/ VI/ 2019-20

Course Name: Sensor Network Mini-Project Lab

Group No/ Roll No: Group No. 16
(XIEIT171840), (XIEIT171842), (XIEIT171848)

Chapter 1: Introduction to Wireless Network						
LO1: Identify the requirements for the real world problems.						
Rubrics For Laboratory Work						
Roll No.	**Name of the Student**	**Knowledge/ Understanding (05)**	**Design/ Coding (05)**	**Punctuality & lab ethics (02)**	**Performance/ Presentation (03)**	**Total (15)**
XIEIT171840	Ismail Pawaskar					
XIEIT171842	Taman Poojary					
XIEIT171848	Deepak Ramchandani					

Ms. Suarvna Aranjo

1. Introduction to Wireless Sensor Networks

Some of the Real World Problems.

- **Ability to monitor security**
- **Any information you need from a device**
- **Energy saving**
- **Health devices connected to smartphones diagnose health conditions quickly**
- **Asset management**
- **Manufacturers using crowdsourcing to build out their manufacturing floor**

(1.1) Requirements of the above problems.

Ability to monitor security

An effective monitoring strategy is required so that actual or attempted security breaches are discovered and there are appropriate processes in place to respond. Good monitoring is more than simply the collection of logs. It is also the use of appropriate tools and skilled analysis to identify indicators of compromise in a timely manner so that corrective action can be taken.

Cloud monitoring provides an easier way to identify patterns and pinpoint potential security vulnerabilities in cloud infrastructure. As there's a general perception of a loss of control when valuable data is stored in the cloud, effective cloud monitoring can put companies more at ease with making use of the cloud for transferring and storing data.

Any information you need from a device in order to perform a particular action.

Examples provided:

1. Recloser on the distribution side of a power line that gets hit by lightening can be closed, checked and reopened by machine after reading the information on the site.
2. Measure temperature and flow of a pipeline to ensure everything is working as expected or be notified is variances outside the norm are seen.
3. Solar back-up to devices that may have power or battery issues.

Energy saving. A lot of devices are left on overnight, or longer. Interact with buildings and homes to save energy. Efficient energy use, sometimes simply called energy efficiency, is the goal to reduce the amount of energy required to provide products and services. For example, insulating a home allows a building to use less heating and cooling energy to achieve and maintain a comfortable temperature. Installing LED lighting, fluorescent lighting, or natural skylight windows reduces the amount of energy required

Fig1.1: Energy Saving

Health devices connected to smartphones diagnose health conditions quickly. You can take pictures and obtain diagnostics to share with health professionals around the world. Enables the collection and sharing of data in an affordable way. Allows inventors to think about use cases. Digitizes the power grid. Play with how energy is being served. Every device in your house will give you an energy profile. Enterprises will benefit from the digitization of devices and enable the next wave of digitization.

Fig1.2: Health Care Devices

Asset management - how to engage information to run control systems. Understand the health of the asset producing the work. Know the health and diagnostics of the machine to reduce downtime and proactively provide maintenance. Ability to tie the supply chain into the process and provide information back to manufacturing thus reducing costs and expense.

The term is commonly used in the financial sector to describe people and companies who manage investments on behalf of others. Those include, for example, investment managers that manage the assets of a pension fund. It is also increasingly used in both the business world and public infrastructure sectors to ensure a coordinated approach to the optimization of costs, risks, service/performance and sustainability.

Fig1.3: Asset Management

Manufacturers using crowdsourcing to build out their manufacturing floor. Consumer wireless routers are only secure for a couple of years. Consumer products have a short life expectancy with consumers. Whereas industrial companies need to have an ongoing relationship with their customers since they have service contract and the products often need ongoing service. In healthcare alone IoT has already made incredible contributions saving lives, giving doctors the ability to see a spectrum of health conditions across a large number of people. It will enable more self-care by patients. Clinical trials are now being based on data received from IoT devices thus accelerating time to market. Industrial is incredibly influential because of the buy in from so many big players like IBM, Cisco and GE.

Fig1.4: Building the manufacturing floor

(1.2) Wireless Sensor Networks Applications

- These networks are used in environmental tracking, such as forest detection, animal tracking, flood detection, forecasting and weather prediction, and also in commercial applications like seismic activities prediction and monitoring.
- Military applications, such as tracking and environment monitoring surveillance applications use these networks. The sensor nodes from sensor networks are dropped to the field of interest and are remotely controlled by a user. Enemy tracking, security detections are also performed by using these networks.
- Health applications, such as Tracking and monitoring of patients and doctors use these networks.
- The most frequently used wireless sensor networks applications in the field of Transport systems such as monitoring of traffic, dynamic routing management and monitoring of parking lots, etc., use these networks.
- Rapid emergency response, industrial process monitoring, automated building climate control, ecosystem and habitat monitoring, civil structural health monitoring, etc., use these networks.

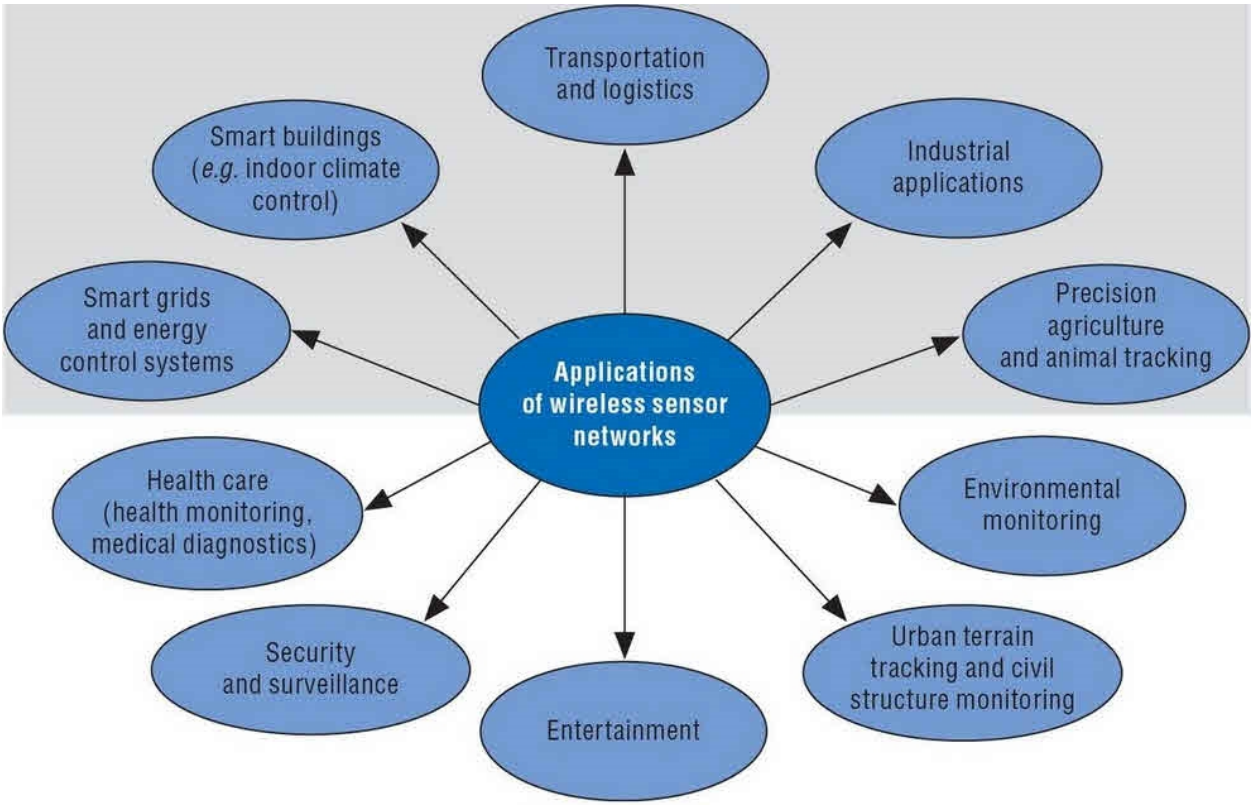


Fig.1.5: Applications of Wireless Sensor Networks

Smart Grids

Smart grids are another area of IoT technology that stands out. A smart grid basically promises to extract information on the behaviours of consumers and electricity suppliers in an automated fashion to improve the efficiency, economics, and reliability of electricity distribution. 41,000 monthly Google searches is a testament to this concept's popularity.

A **smart grid** is an electrical grid which includes a variety of operation and energy measures including smart meters, smart appliances, renewable energy resources, and energy efficient resources. Electronic power conditioning and control of the production and distribution of electricity are important aspects of the smart grid.

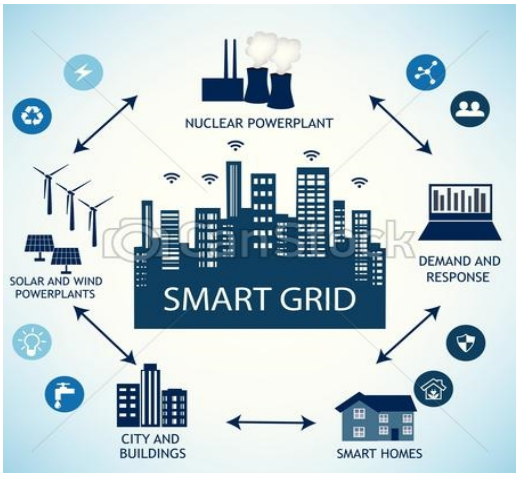


Fig1.6: Smart Grid

Connected Car

Connected car technology is a vast and an extensive network of multiple sensors, antennas, embedded software, and technologies that assist in communication to navigate in our complex world. It has the responsibility of making decisions with consistency, accuracy, and speed. It also has to be reliable. These requirements will become even more critical when humans give up control of the steering wheel and brakes to the autonomous vehicles that are being tested on our highways right now.

A connected car is a car that is equipped with Internet access, and usually also with is wireless local area network (LAN). This allows the car to share internet access, and hence data, with other devices both inside and outside the vehicle.

Fig1.7: Connected Car

Connected Health (Digital Health/Telehealth/Telemedicine)

IoT has various applications in healthcare, which are from remote monitoring equipment to advance and smart sensors to equipment integration. It has the potential to improve how physicians deliver care and also keep patients safe and healthy. Healthcare IoT can allow patients to spend more time interacting with their doctors, which can boost patient engagement and satisfaction. From personal fitness sensors to surgical robots, IoT in healthcare brings new tools updated with the latest technology in the ecosystem that helps in developing better healthcare. IoT helps to revolutionize healthcare and provide pocket-friendly solutions for both the patient and healthcare professional.

Fig1.8: Connected Health

Smart Retail

Retailers have started adopting IoT solutions and using IoT embedded systems across a number of applications that improve store operations, increasing purchases, reducing theft, enabling inventory management, and enhancing the consumer's shopping experience. Through IoT physical retailers can compete against online challengers more strongly. They can regain their lost market share and attract consumers into the store, thus making it easier for them to buy more while saving money.

Fig1.9: Smart Retail

Smart Supply Chain

Supply chains have already been getting smarter for a couple of years. Offering solutions to problems like tracking of goods while they are on the road or in transit or helping suppliers exchange inventory information are some of the popular offerings. With an IoT enabled system, factory equipment that contains embedded sensors communicate data about different parameters, such as pressure, temperature, and utilization of the machine. The IoT system can also process workflow and change equipment settings to optimize performance.

The chance to know more—and understand more—about our customers, their buying habits, and the trends associated with them is invaluable. It allows businesses to form tighter connections with customers and, inevitably, market to them in new and better ways. Beyond the use of data for improved efficiencies noted above, for instance, businesses can get creative with supply chain transparency. They can build a reputation of social responsibility by allowing customers to access—and with AR, even see—where their product came from, who made it, and the conditions in which those workers lived.

Fig1.10: Smart Supply Chain

XAVIER INSTITUTE OF ENGINEERING

Department of Information Technology

Class/ Sem/ A.Y: TE IT/ VI/ 2019-20

Course Name: Sensor Network Lab Mini-Project

Group No/ Roll No: Group No. 16
(XIEIT171840) (XIEIT171842) (XIEIT171848)

Chapter 2: Introduction to Automated Rainwater Harvesting System						
LO2: Conduct a survey of several available literatures in the preferred field of study.						
Rubrics For Laboratory Work						
Roll No.	**Name of the Student**	**Knowledge/ Understanding (05)**	**Design/ Coding (05)**	**Punctuality & lab ethics (02)**	**Performance/ Presentation (03)**	**Total (15)**
XIEIT171840	Ismail Pawaskar					
XIEIT171842	Taman Poojary					
XIEIT171848	Deepak Ramchandani					

Ms. Suvarna Aranjo

2. Introduction to Automated Rainwater Harvesting System

(2.1) Problem Definition:

This project focuses on the design, construction and analysis of a rainwater harvesting system.
Ever since mankind came into existence, the proper use as well as management of water has been a topic of much debate as well as caution for everyone. One of the crucial challenges of water management as well as conservation in a city is to determine the amount of water that any particular city is going to utilize during the next day, figuring out the right amount of water that must be present in the reservoirs and overhead tanks during any particular point of time.

Municipality already supply water at our homes, thus we all have our homes connected to supply grid and have water storage tanks at home. A common issue user faces is that either he forgets to power on the pump to fill the tank or he forgets to power off the pump after the tank is filled leading to overflowing and water wastage. The Smart Water Tank takes care of these problems by transferring the water to the reserve tanks when water level falls below a certain level and powering it off when water tank is filled and simultaneously monitors the water level and sends the data on mobile.

The availability of clean, safe and sufficient amount of water to the daily activities of human being is very crucial. Water plays an important role in transforming or changing the lives of people to better and healthier one. Access to sufficient amount of water is a major problem to most people living in rural and urban areas. The aim of this project is to contribute ideas and knowledge on harvesting rain water so as to compliment the deficiency of water need faced in
today's world.

(2.2) Aims and Objectives:

The process of saving the rainwater in a separate tank like arrangement without making it get mixed with drainage and seawater is called rainwater harvesting.

The main aim and objective of rain water harvesting is to increase is to collect, filter and store rainwater in tanks and reservoirs for irrigation and other purposes such as for drinking and for daily chores and the water level can be monitored to be used for daily chores. Collected rain water through rain water harvesting can be purified in aquifers to be consumed as drinking water.

Rainwater harvesting is the collection and storage of precipitation for later human use. This project focuses on the design, construction and analysis of a rainwater harvesting system. The information collected from this project will be used to build a template for designing a rain water harvesting system that can be placed in remote areas outside with a specific look at the applicability of rain water harvesting in developing regions.
Other aims and objectives related to Rainwater Harvesting:

- To save the water that falls down on earth in the form of rain.
- To eradicate "water shortage" during the summer season.
- If houses are constructed along with rainwater harvesting plan, it will avoid the huge loss caused due to erosion which is caused by runoffs.
- It reduces the draining of available groundwater and increases the level of groundwater."
- In rural areas it can be used to detect the wastage of water and monitoring can be done to avoid the unnecessary overflowing of water in areas where monitoring is difficult task.

(2.3) Scope of project:

1. The purpose of this project is to have a smarter way of water management in order to conserve water resources and energy.
2. The collection offers a better and efficient utilization of energy resource. It is important because portable water is usually not renewable thus reducing wastage.
3. Rain runoff recharging is useful if water level of groundwater is not very deep otherwise the water diverted will get lost in the soil-moisture zone or would require high energy to lift from deep depths & will not be economically beneficial.
4. It helps in preventing urban flooding due to excess rain.
5. Human efforts will be completely eliminated, as the water will be automatically flowing from the main tank to the reserve tanks and keeps on monitoring the level of rainwater.

(2.4) Features of project:

1. The purpose of this project is to have a smarter way of water management in order to conserve water resources and energy.
2. The collection offers a better and efficient utilization of energy resource. It is important because portable water is usually not renewable thus reducing wastage.
3. The simplest rainwater harvesting systems are non-pressurized systems, such as rain barrels, where the pipes run from rain gutters into a tank.
4. Enable users to set their own water goal. Water goal determine how much water they want to use, once the actual water consumption exceed their water goal.
5. If by any chance there is a shortage of water in the main tank then the water flows form the reserve tank to the main tank.
6. There are water sensors at the bottom and top of the tank which will detect the low level and high level of water respectively and accordingly the pump will power on or off.
7. Water Sensor at the top of the tank detects the water and the pump will power off to prevent the overflow of water.

XAVIER INSTITUTE OF ENGINEERING

Department of Information Technology

Class/ Sem/ A.Y: TE IT/ VI/ 2019-20

Course Name: Sensor Network Mini-Project Lab

Group No/ Roll No: Group No. 16
(XIEIT171840) (XIEIT171842) (XIEIT171848)

Chapter 3: Review of Literature

LO2: Conduct a survey of several available literatures in the preferred field of study.

Rubrics For Laboratory Work

Roll No.	Name of the Student	Knowledge / Understanding (05)	Design/ Coding (05)	Punctuality & lab ethics (02)	Performance/ Presentation (03)	Total (15)
XIEIT171840	Ismail Pawaskar					
XIEIT171842	Taman Poojary					
XIEIT171848	Deepak Ramchandani					

Ms. Suvarna Aranjo

3. REVIEW OF LITERATURE

The following are the reviews of literature given below:

There are many problems related to water crisis which can be solved by rainwater harvesting. The fundamental point and target of downpour water collecting is to decrease the utilization of ground water in this manner expanding its levels. This project focuses on the idea that rainwater is collected in the main tank from where it gets transferred to other tanks mainly used in rural areas. It includes gathering, separating and putting away water in tanks and supplies for water system and different purposes, for example, for drinking. Gathered downpour water through downpour water collecting can be filtered in springs to be expended as drinking water [1].

The expense of an expertly introduced water gathering framework can begin around Rs.2000, yet normal expense is around Rs.5000 for a framework that will supply a lot of water. Rainwater Harvesting has been utilized all over the world for thousands of years. Along the quality of the product cost should be effective enough so that it can be bearable by everyone. With the increasing population and demand of water increasing at twice the rate of population growth, it is a viable solution for solving water issues around the world [2].

Boers and Ben-Asher (1982) defined rainwater harvesting as a method to induce, collect, store, and conserve local surface runoff for agriculture in arid and semi- arid regions. The general goal of this project is to evaluate the measure of precipitation in the examination region and to decide the most extreme likely measure of water that can be harvested. It can be majorly used in rural as well as in the urban areas and mostly in remote areas where water scarcity is a major problem [3].
The accessibility of perfect, protected and adequate measure of water to the everyday exercises of person is pivotal. Access to adequate measure of water is a noteworthy issue to the vast majority living in provincial and urban territories [3].

A servo motor is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through servo mechanism. This motor is used to open the lid of the tanks at a particular angle when water level will reach at its threshold level. All motors have three wires coming out of them. Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent from the MCU [4].

Here is a tested Arduino project that uses 3 sensor probes as water level indicator with 3 buzzers a simple controller that turns on a motor when the water has reached the desired level (the highest in our case). A buzzer will beep an alarm up to a threshold level of water and opens the lid with the help of servo motor due to which water flows to the other tanks [5].

The water tanks in our homes are getting overflown often right? How shall we solve it? Let us consider that there is a floating object in your tank. Form a circuit in such a way that when the floating object hits an switch/sensor, an alarm should be activated. It will give indication that water level has increased and flow of water should be stopped [6].

Over centuries, people in diverse geographical positions relied on rainwater and developed indigenous knowledge and techniques to harvest it. This paper introduces Traditional Ecological Knowledge and Indigenous Knowledge and provides an overview of some of the traditional rainwater harvesting methods. Bamboo drip irrigation and rice-fish farming in India are reviewed as case studies. It is vital to take into account and to learn from what local people already know and do and apply this knowledge for our planet's benefits. These traditional rainwater harvesting practices may

have a few challenges to overcome, but they can provide water conservation strategies, especially in vulnerable regions [7].

In arid and semi-arid regions of the world, water is the limiting factor for food production; especially in areas where rain-fed agriculture dominates. Rainwater harvesting systems have the potential to provide a sustainable source of water, while helping to achieve food security and combat soil erosion and flood hazards, simultaneously, if designed correctly. Therefore, land-based rainwater harvesting systems are able to increase crop yields significantly. A variety of techniques for micro and macro-catchment rainwater harvesting schemes are reviewed, with emphasis on design factors which will ensure a functional system [8].

This document presents a comprehensive review of research that has been conducted on rainwater harvesting throughout the world. In some cases collecting runoff from other surfaces, such as parking lots, sidewalks and landscaped areas, is referred to as rainwater harvesting; however, in this paper only systems collecting roof runoff are discussed. Previous studies on harvested rainwater quality have produced contradictory conclusions, with some claiming harvested rainwater was severely polluted while others concluded that it was unpolluted. The need for conserving public potable water supplies continues to increase throughout the world and RWH is a valuable tool that may be used to fulfil this need; however, the lack of knowledge regarding the quality of harvested rainwater has prevented widespread use of this practice. This section presents a review on the origin, transport and fate of potential RWH contaminants, including sediment, nutrients, heavy metals and other chemicals, and the implications these pollutants and processes have on the use of RWH as a supplemental water source [9].

This link provided with the feature and the scope of the project. This link had a complete description of what rainwater harvesting actually is, it provided us with many other projects that has been made in the past. It provided us with some design recommendation regarding the project [10].

The pins on the Arduino board can be configured as either inputs or outputs. We will explain the functioning of the pins in those modes. It is important to note that a majority of Arduino analog pins, may be configured, and used, in exactly the same manner as digital pins. This makes the pins useful for such tasks as implementing a capacitive touch sensor or reading an LED as a photodiode [11].

Buzzer sensor is a audio signaling device which emits sound. We can code the working of buzzer using Arduino using embedded C Programming. It gives beep sound depending on the condition provided. Buzzers can be found in alarm devices, computers, timers and confirmation of user input such as a mouse click or keystroke [12].

XAVIER INSTITUTE OF ENGINEERING

Department of Information Technology

Class/ Sem/ A.Y: TE IT/ VI/ 2019-20

Course Name: Sensor Network Mini-Project Lab

Group No/ Roll No: Group No. 16
(XIEIT171840) (XIEIT171842) (XIEIT171848)

Chapter 4: System Description

LO3: Study software/ hardware skills and build the project successfully by hardware requirements, coding, emulating and testing.
LO4: To report and present the findings of the study conducted in the preferred domain
LO5: Demonstrate an ability to work in teams and manage the conduct of the research study.

Rubrics For Laboratory Work

Roll No.	Name of the Student	Knowledge / Understanding (05)	Design/ Coding (05)	Punctuality & lab ethics (02)	Performance/ Presentation (03)	Total (15)
XIEIT171840	Ismail Pawaskar					
XIEIT171842	Taman Poojary					
XIEIT171848	Deepak Ramchandani					

Ms. Suvarna Aranjo

4. SYSTEM DESCRIPTION

(4.1) Design

Block Diagram:

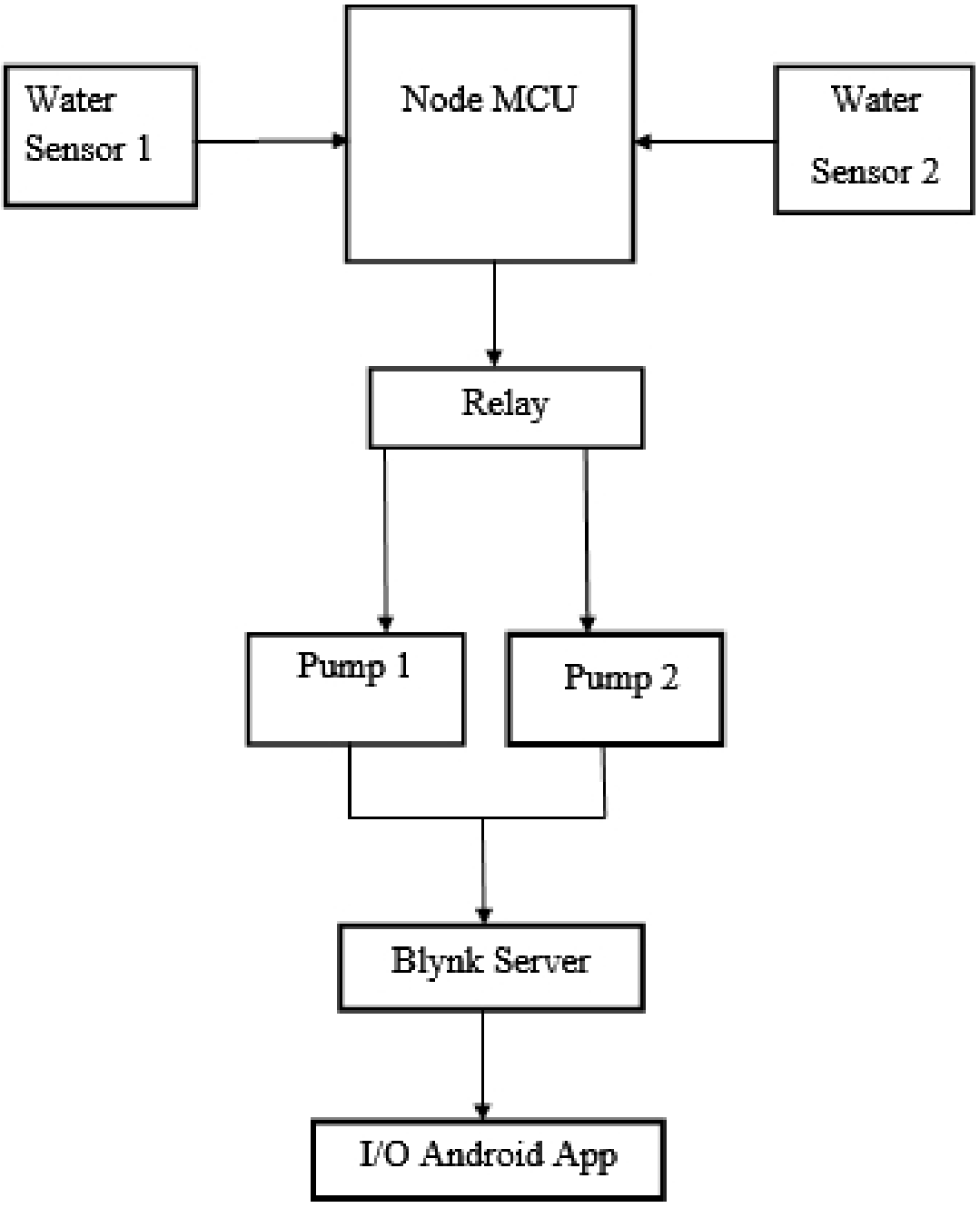


Fig.4.1: Block Diagram

(4.2) Hardware and Software used

Hardware required:-

1) Node MCU
2) Breadboard
3) Relay for controlling pumps
4) Sensors:-
    i. Water Sensors
    ii. Pumps
    iii. Ultrasonic Sensor

Software required:-

1) Arduino software (IDE)
2) Windows Operating System

Cloud Platforms used:-

Blynk Server

Description:

Hardware:-

1. Node MCU

NodeMCU is an open source firmware for which open source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit). The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits

The firmware uses the Lua scripting language. The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially was based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications

2. Breadboard

A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate.

3. Sensors

i. Water Sensor

A water sensor is a device used in the detection of the water level for various applications. Water sensors can come in several variations that include ultrasonic sensors, pressure transducers, bubblers, and float sensors

ii. Pumps

A pump is a device that moves fluids (liquids or gases), or sometimes slurries, by mechanical action. Pumps can be classified into three major groups according to the method they use to move the fluid: direct lift, displacement, and gravity pumps. The range of the pumps that it can work is 3V to 6V.

iii. Ultrasonic Sensor

The HC-SR04 ultrasonic sensor uses SONAR to determine the distance of an object just like the bats do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package from 2 cm to 400 cm or 1” to 13 feet.

The operation is not affected by sunlight or black material, although acoustically, soft materials like cloth can be difficult to detect. It comes complete with ultrasonic transmitter and receiver module.

Software:-

1. Arduino software (IDE)

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.

2. Windows Operating System
Arduino IDE works on Windows Operating System and Arduino IDE is used for writing Embedded C programming code.

3. Blynk Server
Blynk is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets.
There are three major components in the platform:

- **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.
- **Blynk Server** - responsible for all the communications between the smartphone and hardware. You can use our Blynk Cloud or run your private Blynk server locally. Its open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.
- **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and outcoming commands.

(4.3)Implementation Methodology

Rain Water Harvesting is one of the solution to India's Water Crisis. India is facing the worst water crisis in its history. As per a report by NitiAyog 21 cities in India will run out of groundwater by 2020. According to a report published by Water Aid around 80% of India's surface water is polluted. A majority of the population is dependent on the groundwater which is again struggling to keep pace with the needs of the rising population; the situation will likely get worse as the population will increase.

Rain-water harvesting can save gallons of water for daily household or office use. For every 1000 sq feet of roof space, approximately 620 gallons of water can be saved every time it rains. This source of non-potable water can be used in flushing toilets, laundry etc. The saved water can fulfill at least 70% of the water demand in a household of 3 people during a drought year.

Rainwater harvesting has been adopted by many countries as a viable means to save water. As it is too difficult to measure the amount of consumption of water sometimes the wastage of water starts due to overflowing or by deficiency of water. With the increasing population and dependence on water, it becomes pertinent for households to start investing in rain-water harvesting systems (RWH). Govt. both at the center and state must take a proactive step towards making it mandatory for buildings and complexes to install Rainwater Harvesting System. Gathering the information regarding the water level collected in the tank makes it easy for the persons to make the efficient use of water.

Collection system: Roof surface and gutters to capture the rainwater and send it to the storage system
Inlet filter: Screen filter to catch large debris

Storage tank: Storage tanks composed of food-grade polyester resin material approved by the U.S. Food and Drug Administration (FDA), which is green in color and helps to reduce bacterial growth

Overflow: Drainage spout that allows for overflow if the storage tank gets full

Controls: Control system that monitors water level and filtration system

Treatment system: Filtration and disinfection system that treats the water to non-potable or potable standards

Pump: Pump to move water through the system to where it will be used

Backflow prevention: Backflow preventer to ensure that under negative pressure water cannot flow backwards through the system into the make-up water system

Flow meter: Flow meter (with data logger) to measure water production

Power supply: Systems may use either conventional power sources or, to improve off-grid capabilities, alternative sources such as stand-alone or grid-tied solar systems
Water level indicator: Monitors the water level in the storage tank

Step by step implementation of prototype:

There are total 3 tanks which are used in this prototype
One is main tank (through which water reaches building and apartments) and other 2 are reserved storage tanks which are used to store or reserve rainwater.

Step1) Firstly the rain water is collected in the two reserve tanks by installing an incline surface above through which rain water flows into the tanks or directly by rainwater.

Step2) Assuming the initial condition that the level of water in the main tank is low which is detected by water sensor at the bottom and therefore the pumps are powered on and henceforth the water which gets collected in the reserve tanks will be transferred to the main tank and simultaneously the level of the water is detected by the ultrasonic sensor and displayed on the mobile.

Step 3) Water will continue to flow in the main tank until it reaches its threshold level i.e. at the top. As soon as the water level reaches up to its threshold level, water sensor at the top of the tank will detect it and power off the pumps to prevent the overflowing of water from the main tank.
As the water level increases or decreases gradually in the tank it is detected or monitored by the ultrasonic sensor which is located at the top of the tank and consistently the consumption of water or level of water in the tank is given the mobile.

Step 4) Now there will be a condition when all the three tanks becomes full and the pumps will be off to prevent the flow of water and the indication of the same is given on the mobile. Then the water from the main tank is used by the society or other commercial purposes. After sometime again the water level becomes low and volume of level of consumption of water is given on the mobile and consistently it is detected by the water sensor at the bottom and the water from the reserve tanks are taken.

(4.5) Code:

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#define TRIGGERPIN D1
#define ECHOPIN    D2

int pumppin1=6; // on digitalpin6
int pumppin2=11;
int sensorval1;
int sensorval2;

char auth[] = "WcwUnqu-vZ5HnmeDppvY2jxkEAkpQPbK";

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "JioFi4_1353F8";
char pass[] = "3nkvsvjv3o";

WidgetLCD lcd(V1);

void setup()
{
        Serial.begin(9600);
        pinMode(A8,INPUT);// A8 BOTTOM ONE
        pinMode(A11,INPUT);// A11 UPPER ONE
        pinMode(pumppin1,OUTPUT);
        pinMode(pumppin2,OUTPUT);

        Serial.begin(9600);
        pinMode(TRIGGERPIN, OUTPUT);
        pinMode(ECHOPIN, INPUT);
        Blynk.begin(auth, ssid, pass);

        // You can also specify server:
        //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 8442);
        //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8442);

        lcd.clear();
        lcd.print(0, 0, "Distance in cm");
}

void loop()
{
        sensorval1=analogRead(A8);
        sensorval2=analogRead(A11);
        Serial.println(sensorval1);
        Serial.println(sensorval2);
```

```
        if (sensorval1 > 300 && sensorval2 > 300)   //both dry
        {
                digitalWrite(pumppin1, HIGH);
                digitalWrite(pumppin2, HIGH);
        }

        if(sensorval1 < 300 && sensorval2 >300 )  //one wet and second dry
        {
                servo1.write(0);
                Serial.println(pos);
        }

        if(sensorval1 < 400 && sensorval2 < 400 )  //both wet
        {
                digitalWrite(pumppin1, LOW);
                digitalWrite(pumppin2, LOW);
        }

        lcd.clear();
        lcd.print(0, 0, "Distance in cm");
        long duration, distance;
        digitalWrite(TRIGGERPIN, LOW);
        delayMicroseconds(3);

        digitalWrite(TRIGGERPIN, HIGH);
        delayMicroseconds(12);

        digitalWrite(TRIGGERPIN, LOW);
        duration = pulseIn(ECHOPIN, HIGH);
        distance = (duration/2) / 29.1;
        Serial.print(distance);
        Serial.println("Cm");
        lcd.print(7, 1, distance);
        Blynk.run();

        delay(3500);
}
```

(4.4) Hardware circuit diagram

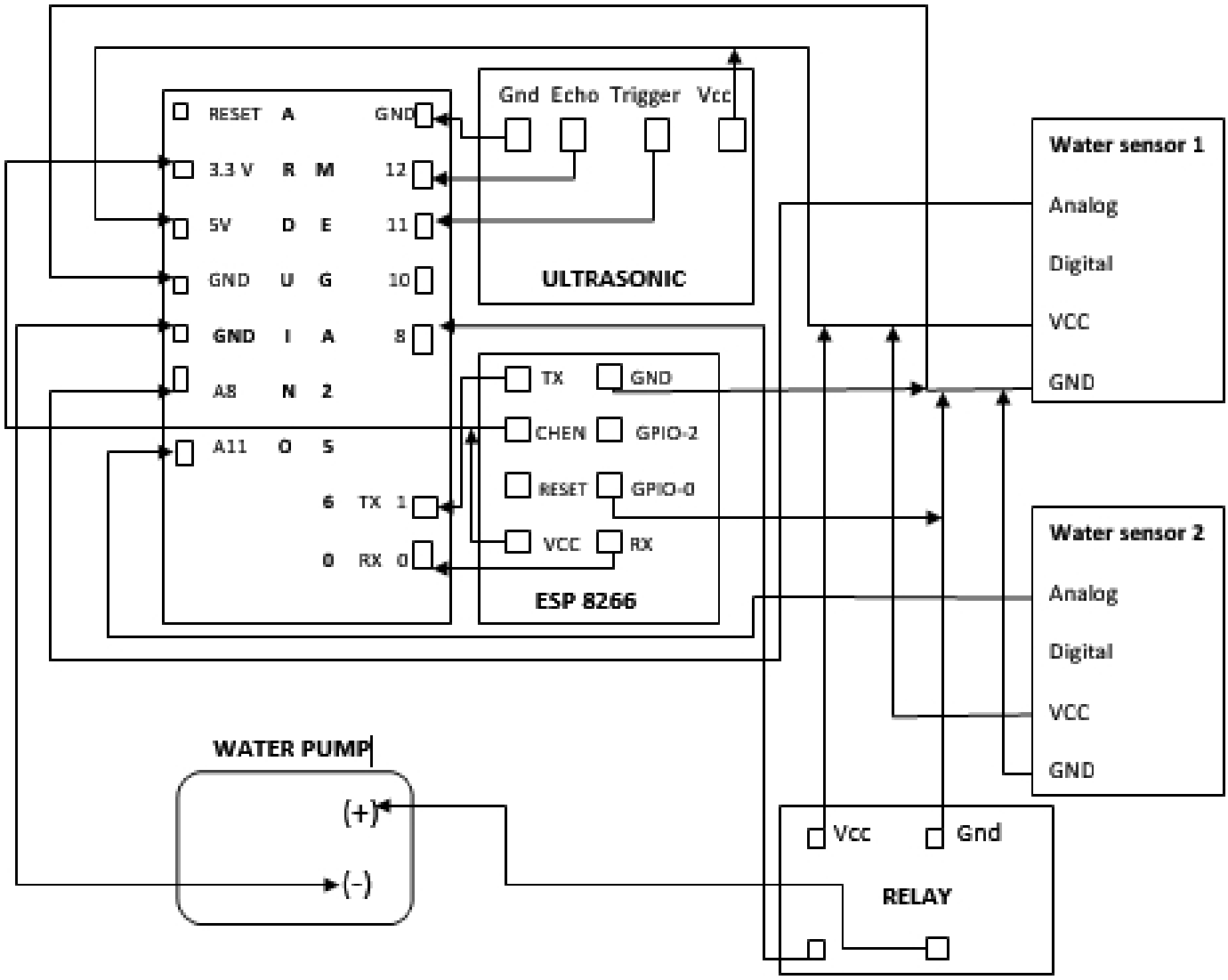


Fig.4.2: Circuit Diagram

(4.6) Final Prototype:

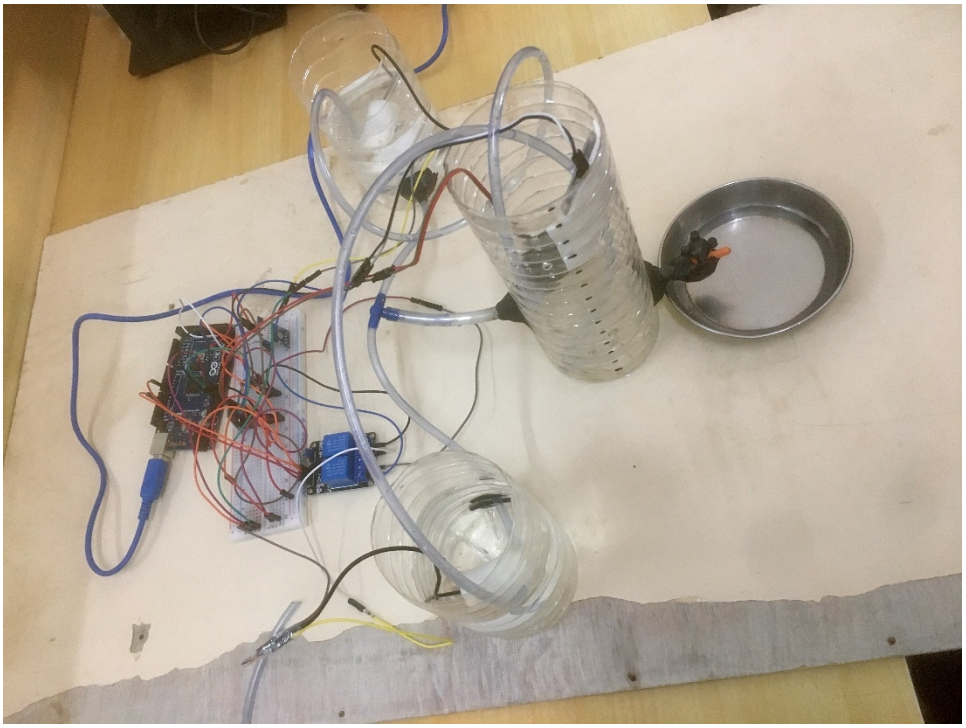


Fig 4.3: Final prototype (Image 1)

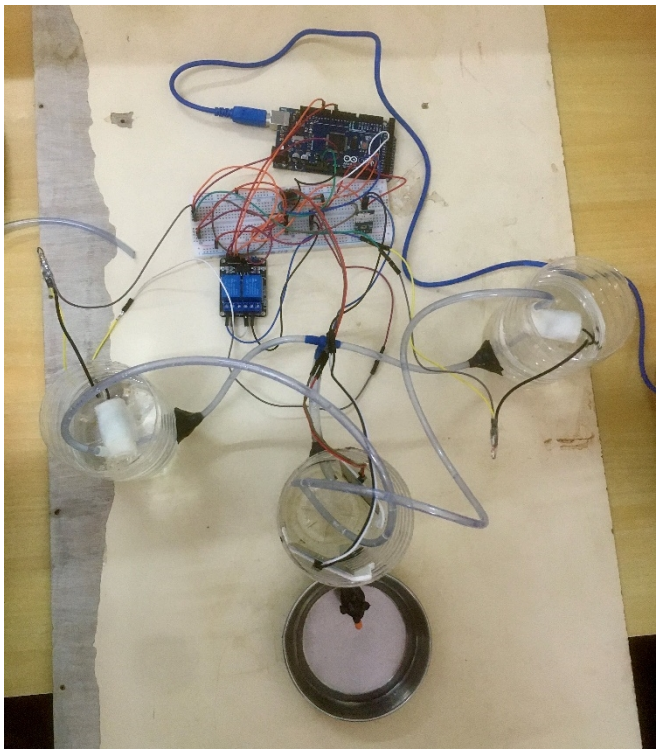


Fig 4.4: Final Prototype (Image 2)

(4.7) Conclusion and Future Scope

Future Scope:

1. Rainwater harvesting systems can currently reduce mains water consumption by 30%, and prototype systems are often developed with a view toward further reductions which is meant to monitor the level of the water in the reserved tanks.
2. With further population growth expected in India, logic dictates that rainwater harvesting systems will become the norm for domestic and commercial users alike. This demand for systems will also increase if north India experiences extended periods of decreased rainfall, as predicted by climate change analysts.
3. The future dictates that rainwater harvesting systems will need to be easy to retrofit, make use of water-storage space, eliminate the need for fossil fuel-driven pumping technology, offer protection against flooding, and be reliable.
4. Rainwater harvesting can help overcome the problem of flood, while also helping in storing excess water by knowing the level of water in the tanks anywhere and at any time that can help in providing water to the water scarce region.
5. This renewable source of water management can help in overcoming major water-related problems, that are not only plaguing the world right now but can have a severe impact on the global population and environment in the future.
6. India had enough systems in place in the past to recover the rainwater. Many places in Rajasthan such as Jaisalmer — a city that built a flourishing civilization and a stunning fort of yellow sandstone despite receiving only 50-100 mm of rain — is currently reeling under water shortages. It could built a water secure future as city planning in the past involved its rainwater being harvested from rooftops to tanks. This helped keep shortages at bay during the punishing dry summer months.

Conclusion:

Rainwater harvesting is people-centric. It will continue to remain largely on the drawing board if people are not actively involved. This because primarily, the Government cannot harvest rain; people will have to do it and for that, their involvement is necessary as it has to be done in every house; every colony; every village; and for every catchment. This system is mainly useful when there is scarcity of water. The incentive to do this only comes when we are dependent on groundwater for our needs. Also it leads to know regarding the level of water present under the ground with the help of the app provided in the mobile. This makes the system very efficient to use as it leads to know the unnecessary wastage of water.

(4.8) Constraints for real time deployment

When Rainwater Harvesting System is deployed to the real time environment the various constraints are as follows:

- **Design consideration**

  The main design parameters of a Rainwater Harvesting system are rainfall, catchment area, collection efficiency, tank volume and water demand. The appropriate design and evaluation of Rainwater Harvesting system is necessary to improve system performance and the stability of the water supply.
- **Regular Maintenance**

  Rainwater harvesting systems require regular maintenance as they may get prone to rodents, mosquitoes, algae growth, insects and lizards. They can become as breeding grounds for many animals if they are not properly maintained.
- **Storage Limits**

  The collection and storage facilities may also impose some kind of restrictions as to how much rainwater you can use. During the heavy downpour, the collection systems may not be able to hold all rainwater which ends in going to drains and rivers.
- **Water quality** related issues and available technologies for disinfection and filtration of rainwater.

  Water security has been defined as "accessibility, reliability and timely availability of adequate safe water to satisfy basic human need". Rainwater in rural areas – being situated far from atmospheric and industrial pollution is fairly clean except for some dissolved gases. On the other hand urban areas are characterized by a high traffic and industry impact and are therefore contaminated by particles, heavy metals and organic air pollutants.
- **Social challenges** and effect of **climate change** that might affect the system and the purity of the water used for drinking and other purposes.
- **Limited financial power** of farmers has resulted in low maintenance of old tanks and also decreased the capacity of farmers of purchasing materials and equipments to utilize the technology.
- **Unpredictable Rainfall** and sometimes little or no rainfall can limit the supply of rainwater. It is not advisable to depend on rainwater alone for all your water needs in areas where there is limited rainfall. Rainwater harvesting is suitable in those areas that receive plenty of rainfall.
- Only those who own an independent and large roof area and have sufficient open space for the installation of storage tanks can afford to set up such a system.
- Neither the master and zonal plans of the cities nor the building bye laws had any stipulation regarding rooftop rainwater harvesting and therefore gutter, inflow system and storage tanks were not conceived when the buildings were designed norhas no incentive structure yet been developed for adding such facilities on the existing buildings.

References:

1. https://brainly.in/question/1657794 accessed on 22.04.2020, 7:40pm

2. http://www.southeastgreen.com/index.php/contactadevertise-with-us/archives/175-rainwater-harvesting-the-solution-to-our-water-crisis accessed on 24.04.2020, 8:00pm

3. http://erepository.uonbi.ac.ke/bitstream/handle/11295/65441/Mwakilembe_Rainwater/ accessed on 26.08.2019, 8:15pm

4. https://circuitdigest.com/article/servo-motor-basics accessed on 29.04.2020, 8:40am

5. https://www.electroschematics.com/9964/arduino-water-level-indicator-controller/ accessed on 29.04.2020, 9:30am

6. http://www.robotictutorials.com/tutorial/kit-creatives/electrical-project-water-level indicator-f.html accessed on 29.04.2020, 2:30pm

7. https://www.ruvival.de/rainwater-harvesting-literature-review/ accessed on 2.05.2020, 9:45am

8.https://tore.tuhh.de/bitstream/11420/1497/1/RUVIVAL_Publication_Series_Volume_2.pdf#page=36 accessed on 6.5.2020, 10:00am

9.https://pdfs.semanticscholar.org/d9c9/186c7a1287d279c8ebb1ef2f1026ebf60beb.pdf accessed on 11.5.19, 10:15am

10. https://www.britannica.com/technology/rainwater-harvesting-system accessed on 12.5.2020, 10:00am

11. https://www.youtube.com/watch?v=JhH7DEMEkdE accessed on 18.05.2020, 9:30am

12. https://www.youtube.com/watch?v=8lM13lE2zyQ accessed on 26.05.2020, 9:45am

XAVIER INSTITUTE OF ENGINEERING

Department of Information Technology

Class/ Sem/ A.Y: TE IT/ VI/ 2019-20

Course Name: Sensor Network Mini Project Lab

Group No/ Roll No: Group No. 16
(XIEIT171840) (XIEIT171842) (XIEIT171848)

Assignment No. 1: Case study of Contiki Cooja

LO1: Identify the requirements for the real world problems.
LO2: Conduct a survey of several available literatures in the preferred field of study.
LO3: Study software/ hardware skills and build the project successfully by hardware requirements, coding, emulating and testing.
LO4: To report and present the findings of the study conducted in the preferred domain
LO5: Demonstrate an ability to work in teams and manage the conduct of the research study.

Rubrics For Assignment Work

Roll No.	Name of the Student	Knowledge & Content (03)	Neatness and Timeline (02)	Total (05)
XIEIT171840	Ismail Pawaskar			
XIEIT171842	Taman Poojary			
XIEIT171848	Deepak Ramchandani			

Ms. Suvarna Aranjo

APPENDIX

Assignment No. 1

Study of opensource simulator – **Contiki Cooja**

Description:
Contiki is an operating system for networked, memory-constrained systems with a focus on low-power wireless Internet of Things devices. Extant uses for Contiki include systems for street lighting, sound monitoring for smart cities, radiation monitoring, and alarms. It is open-source software released under a BSD license.

Contiki was created by Adam Dunkels in 2002 and has been further developed by a worldwide team of developers from Texas Instruments, Atmel, Cisco, ENEA, ETH Zurich, Redwire, RWTH Aachen University, Oxford University, SAP, Sensinode, Swedish Institute of Computer Science, ST Microelectronics, Zolertia, and many others. Contiki gained popularity because of its built in TCP/IP stack and lightweight preemptive scheduling over event-driven kernel which is a very motivating feature for IoT. The name Contiki comes from Thor Heyerdahl's famous Kon-Tiki raft.

Contiki provides multitasking and a built-in Internet Protocol Suite (TCP/IP stack), yet needs only about 10 kilobytes of random-access memory (RAM) and 30 kilobytes of read-only memory (ROM).A full system, including a graphical user interface, needs about 30 kilobytes of RAM.

Hardware:
Contiki is designed to run on types of hardware devices that are severely constrained in memory, power, processing power, and communication bandwidth. A typical Contiki system has memory on the order of kilobytes, a power budget on the order of milliwatts, processing speed measured in megaHertz, and communication bandwidth on the order of hundreds of kilobits/second. Such systems include many types of embedded systems, and old 8-bit computers.

Networking:
Contiki provides three network mechanisms: the uIP TCP/IP stack, which provides IPv4 networking, the uIPv6 stack, which provides IPv6 networking, and the Rime stack, which is a set of custom lightweight networking protocols designed for low-power wireless networks. The IPv6 stack was contributed by Cisco and was, when released, the smallest IPv6 stack to receive the IPv6 Ready certification. The IPv6 stack also contains the Routing Protocol for Low power and Lossy Networks (RPL) routing protocol for low-power lossy IPv6 networks and the 6LoWPAN header compression and adaptation layer for IEEE 802.15.4 links.

Rime is an alternative network stack, for use when the overhead of the IPv4 or IPv6 stacks is prohibitive. The Rime stack provides a set of communication primitives for low-power wireless systems. The default primitives are single-hop unicast, single-hop broadcast, multi-hop unicast, network flooding, and address-free data collection. The primitives can be used on their own or combined to form more complex protocols and mechanisms.

Low power operation:
Many Contiki systems are severely power-constrained. Battery operated wireless sensors may need to provide years of unattended operation and with little means to recharge or replace batteries. Contiki provides a set of mechanisms to reduce the power consumption of systems on which it runs. The default mechanism for attaining low-power operation of the radio is called ContikiMAC. With

ContikiMAC, nodes can be running in low-power mode and still be able to receive and relay radio messages.

Simulation:
The Contiki system includes a sensor simulator called Cooja, which simulates of Contiki nodes. The nodes may belong to either of three classes a) emulated nodes, where the entire hardware of each node is emulated, Cooja nodes, b) where the Contiki code for the node is compiled for and executed on the simulation host, or c) Java nodes, where the behavior of the node must be reimplemented as a Java class.

One Cooja simulation may contain a mix of sensor nodes from any of the three classes. Emulated nodes can also be used to include non-Contiki nodes in a simulated network.

Programming model:
To run efficiently on small-memory systems, the Contiki programming model is based on protothreads. A protothread is a memory-efficient programming abstraction that shares features of both multithreading and event-driven programming to attain a low memory overhead of each protothread. The kernel invokes the protothread of a process in response to an internal or external event. Examples of internal events are timers that fire or messages being posted from other processes. Examples of external events are sensors that trigger or incoming packets from a radio neighbor.

Protothreads are cooperatively scheduled. Thus, a Contiki process must always explicitly yield control back to the kernel at regular intervals. Contiki processes may use a special protothread construct to block waiting for events while yielding control to the kernel between each event invocation.

Features:
Contiki supports per-process optional preemptive multithreading, inter-process communication using message passing through events, as well as an optional graphical user interface (GUI) subsystem with either direct graphic support for locally connected terminals or networked virtual display with Virtual Network Computing (VNC) or over Telnet.

A full installation of Contiki includes the following features:

1)Multitasking kernel
2)Optional per-application preemptive multithreading
3)Protothreads
4)Internet Protocol Suite (TCP/IP) networking, including IPv6
5)Windowing system and GUI
6)Networked remote display using Virtual Network Computing
7)A web browser (claimed to be the world's smallest)
8)Personal web server
9)Simple telnet client
10)Screensaver

Contiki is supported by popular SSL/TLS libraries such as wolfSSL, which includes a port in its 3.15.5 release.

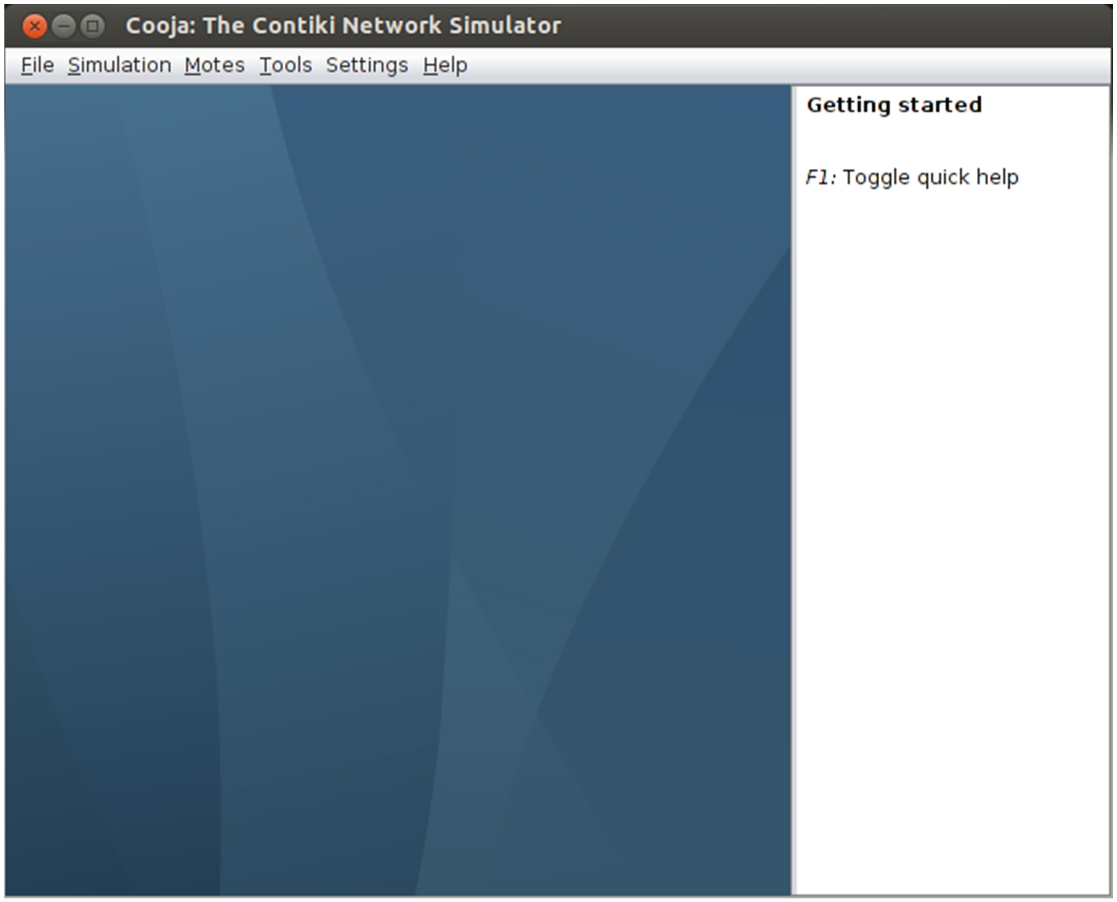
Running Cooja Simulator:

Cooja Simulator is a network simulator specifically designed for Wireless Sensor Networks. A summary of how Cooja executes binaries of different platforms (different types of nodes) and a few details about its structure can be found here.

The simple way of running Cooja is executing it inside its own directory (we will see later how to execute Cooja from any working directory):

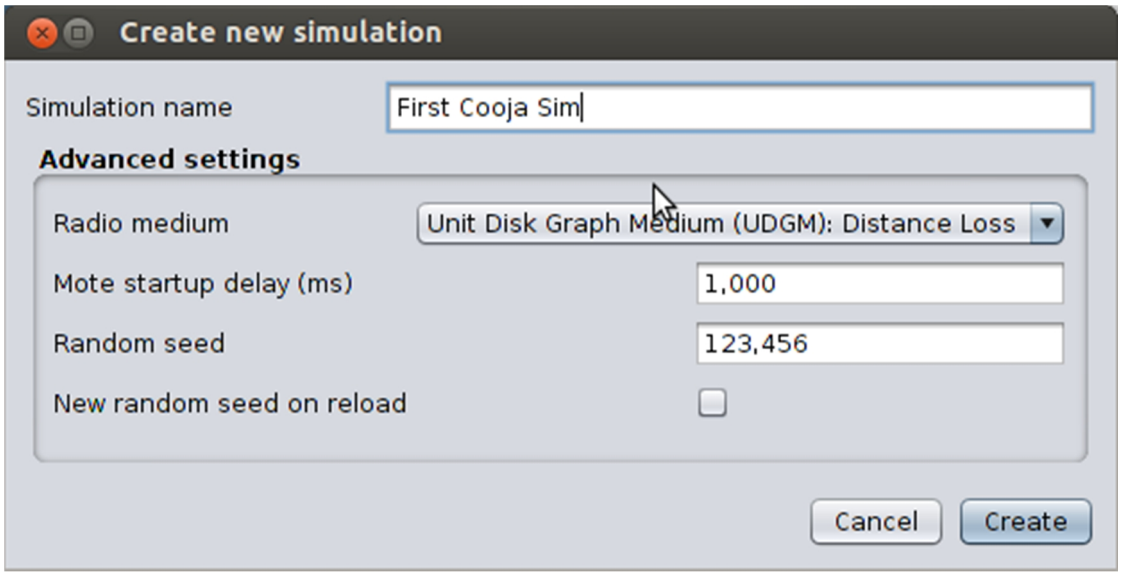
cd contiki/tools/cooja
ant run

When you execute Cooja the following window pops up.



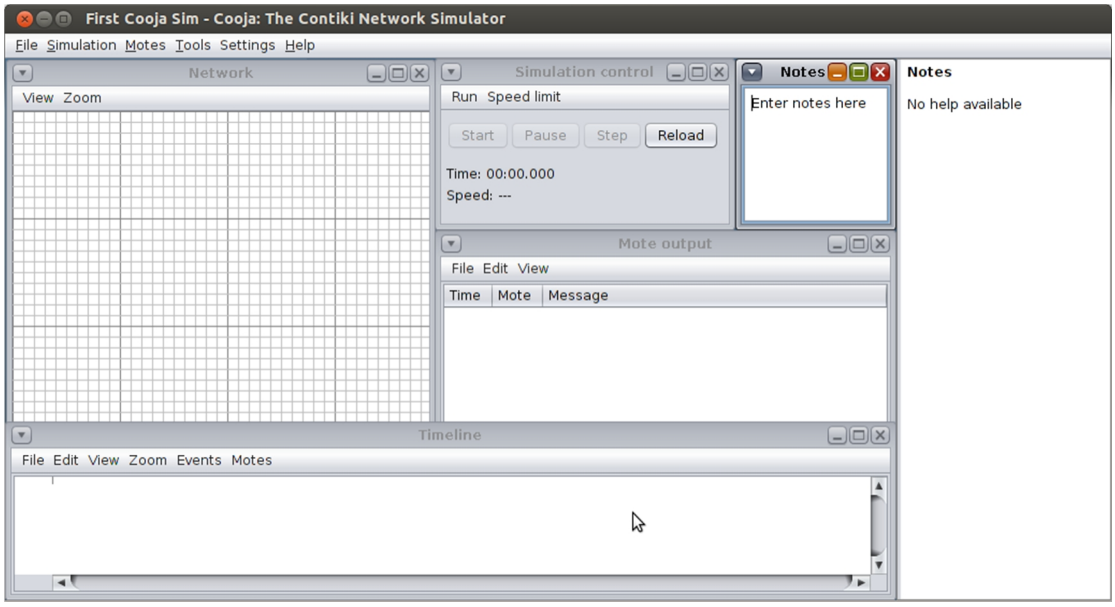
1) Creating a new simulation:

In the File menu you can start a new simulation or open an existing one. At this moment we will start a new one. You should select: File > New simulation... The following window should show up.



In the Simulation name box you should enter an identifier for the new simulation, and in Advanced settings you can choose parameters of the simulation such as Radio medium, node startup delay and random seed generation. We are going to create a simulation called "First Cooja Sim", as shown above.

After creating a new simulation, Cooja's window is filled with the main simulating tools, as shown in the next image.



Here we briefly describe the functionalities of each tool:

- **Network** - Shows the location of each node in the network. Can be used to visualize the status of each node, including LEDs, mote IDs, addresses, lof outputs, etc. Initially this window is empty and we need to populate it with our sensors.

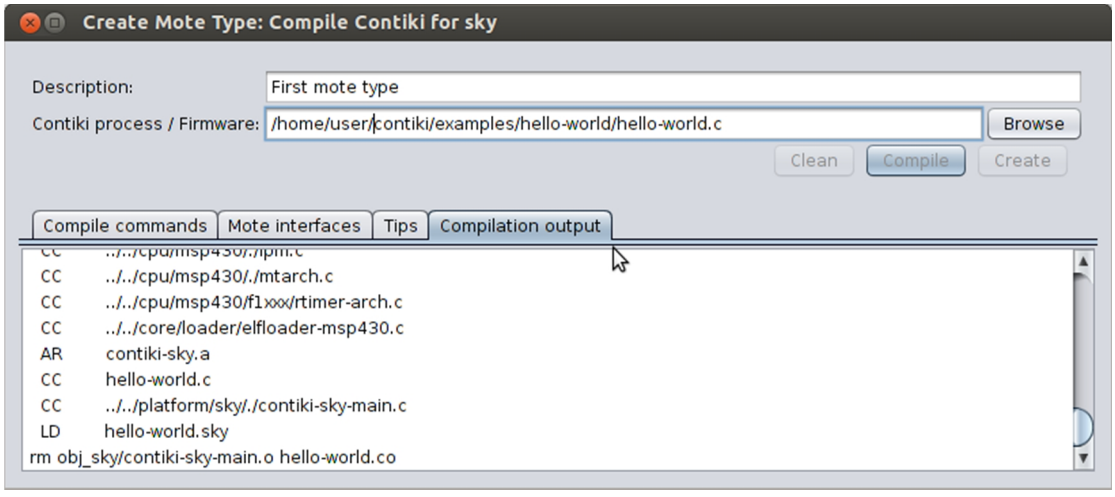
- **Simulation Control** - This panel is used to Start, Pause, Reload or execute Steps of the simulation. It shows the time of execution and the speed of simulation. It means that we can run the events several times faster than it would take in real-time execution.
- **Notes** - This is a simple notepad for taking notes about the simulation.
- **Mote output** - Shows all output of serial interface of the nodes. It is possible to enable one window of Mote output for each node in the simulation.
- **Timeline** - Simulation timeline where messages and events such as channel change, LEDs change, log outputs, etc are shown.

In addition to the default tools, it is possible to exhibit other tools such as Breakpoints, Radio messages, Script editor, Buffer view and Mote duty cycle, which can be enable in the Tools menu.

2) Creating a new mote type:

You need to create a new mote type before starting any simulation. You can do this in the menu Motes > Add motes > Create new motes type. Let's select Sky mote in order to create a mote of the same type as the used Tmote Sky.

The window that shows up (see below) asks for the Description of the new mote type and the Contiki process / Firmware. You can name your mote type as First mote type and you can select the firmware that will be used during the simulation using the Browse button. After selecting the desired firmware you can test the compilation click the Compile button. In this example we will use the Hello World fimware, typically located at /contiki/examples/hello-world/hello-world.c. If the compiling process is successed, you will see a final message: LD hello-world.sky in the Compilation output tab.



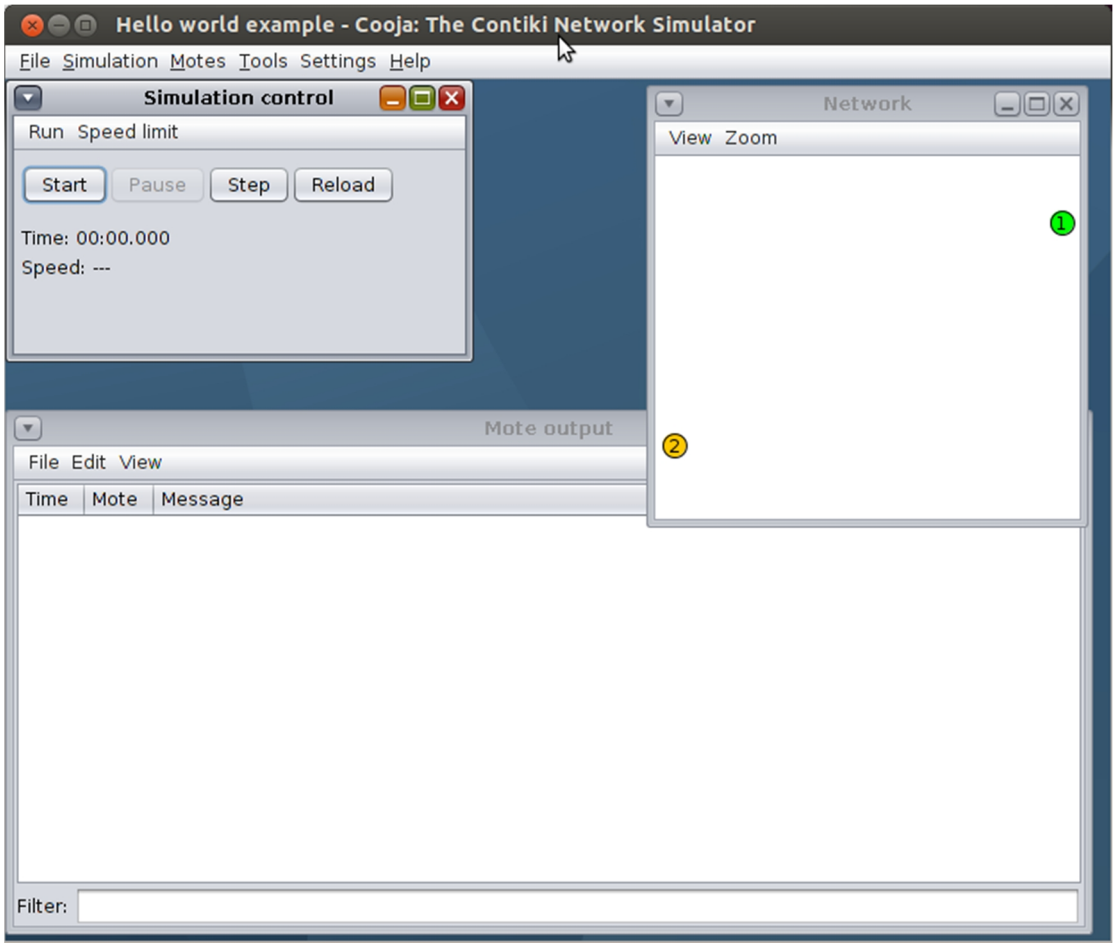
3) Saving Simulation file:

The simulation configuration and parameters such as number of nodes, type of nodes, firmware used, location of nodes, etc. can be stored in a file for future simulations. You can save your simulation configuration in File > Save simulation as.... The generated file has extension ".csc". In the case of Hello World firmware there already exist a .csc file inside contiki/examples/hello-world, called hello-world-example.csc, which contains 2 nodes. We are going to use this pre-existing Cooja file to show how to debug an application.

Debugging with Cooja Simulator:
The easiest way to debug an application is to initially create a proper .csc simulation file, as described above. Then, inside the folder with the simulation file and all firmware source code (.c and .h files), you need to execute the following command.

make TARGET=cooja <simulation_file>.csc

The command above will initialize Cooja Simulator loaded with the desired simulation file and will make Cooja recognized the source code of the application we intend to debug. In our case we are going to simulate and debug the Hello Worls application, so we need to go into /contiki/examples/hello-world folder and execute the command make TARGET=cooja hello-world-example.csc. The window shown below should be displayed.



If we click on Start button the simulation begins and we can see both nodes printing out log messages on the Mote output window.

All debugging tools can be accessed by right-clicking on a node and selecting Mote tools for Sky in the pop-up menu. The two main debugging tools that we are going to use are: Msp Code Watcher and Msp Stack Watcher. The first one can be used for analysing the source code of the application in a step-by-step fashion, with the possibility of setting breakpoints. The second tool is used to read and write the variables used throughout the execution.

XAVIER INSTITUTE OF ENGINEERING

Department of Information Technology

Class/ Sem/ A.Y: TE IT/ VI/ 2019-20

Course Name: Sensor Network Mini-Project Lab

Group No/ Roll No: Group No. 16
(XIEIT171840) (XIEIT171842) (XIEIT171848)

Assignment No. 2: Case Study of Cupcarbon
LO1: Identify the requirements for the real world problems. **LO2: Conduct a survey of several available literatures in the preferred field of study.** **LO3: Study software/ hardware skills and build the project successfully by hardware requirements, coding, emulating and testing.** **LO4: To report and present the findings of the study conducted in the preferred domain** **LO5: Demonstrate an ability to work in teams and manage the conduct of the research study.**

Rubrics For Assignment Work

Roll No.	Name of the Student	Knowledge & Content (03)	Neatness and Timeline (02)	Total (05)
XIEIT171840	Ismail Pawaskar			
XIEIT171842	Taman Poojary			
XIEIT171848	Deepak Ramchandani			

Ms. Suvarna Aranjo

Assignment No. 2

Study of opensource simulator - **Cupcarbon**

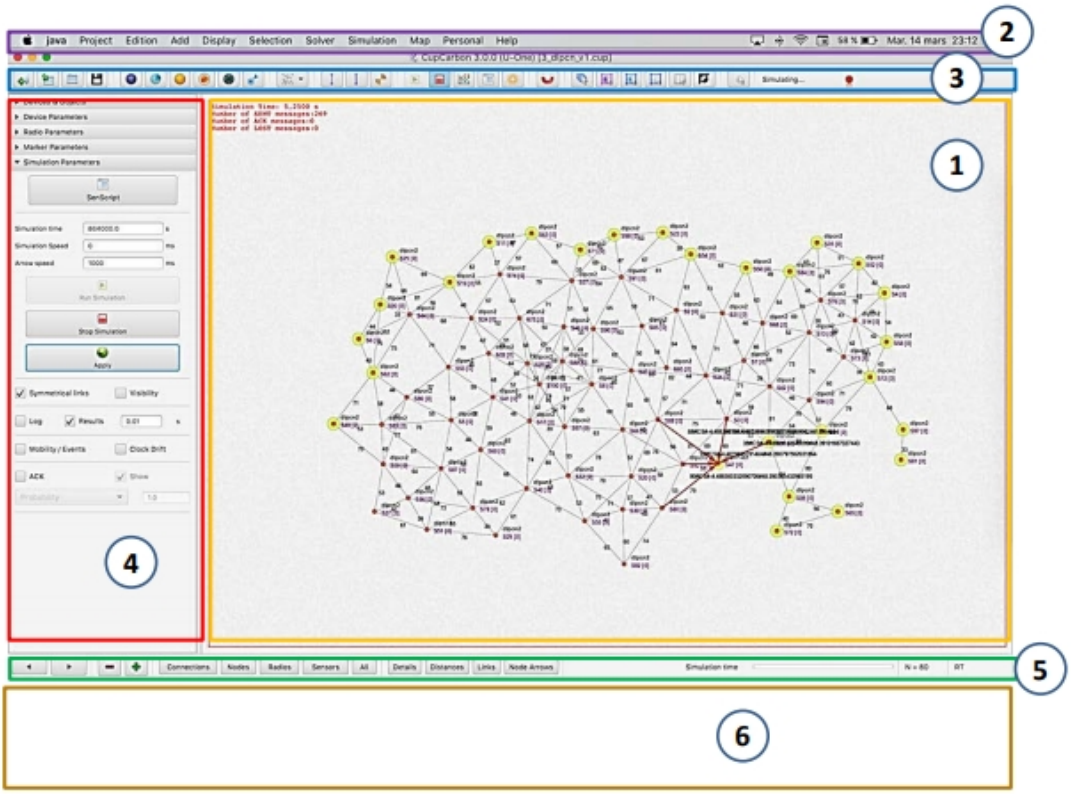
Description:
This section presents the CupCarbon network simulator [20, 21], which is the tool under evaluation. The section presents the main characteristics and functionalities of this network simulator, the graphical user interface and an example performed in CupCarbon.

Cupcarbon is a simulator specially designed for WSN networks. Its objective is the design, visualization, and validation of algorithms for environmental monitoring, data collection, etc. It allows creating scenarios with environmental stimulus as fires, gases, and mobile objects within scientific and educational projects. It offers two simulation environments. On the one hand, it enables the design of scenarios with mobility and generation of natural events and on the other, the simulation of discrete events in WSNs.

Cupcarbon is a simulator specially designed for WSN networks. Its objective is the design, visualization, and validation of algorithms for environmental monitoring, data collection, etc. It allows creating scenarios with environmental stimulus as fires, gases, and mobile objects within scientific and educational projects. It offers two simulation environments. On the one hand, it enables the design of scenarios with mobility and generation of natural events and on the other, the simulation of discrete events in WSNs.

The energy consumption can be calculated and graphically displayed as a function of the simulation time. In addition, it allows observing the visibility of propagation and interference models. CupCarbon is able to simulate the ZigBee, LoRa and WiFi protocols. On the other hand, although CupCarbon does not stand out for having a wide variety of protocols integrated, the fact that it has been implemented using Java permits the creation of different algorithms.

As Figure 2 shows, the CupCarbon Graphical User Interface (GUI) is composed by six main parts: (1) The map (in the centre), (2) The menu bar (on the top), (3) The Toolbar (bellow the menu), (4) The parameter menu (on the left), (5) The state bar (at the bottom) and (6) the console (at the bottom).



Within the selection section of the radio parameters, we can choose between ZigBee, Wifi and Lora. By default, the IEEE 802.15.4 (ZigBee) standard is automatically set for each new added sensor. One of the most interesting competences of this tool is the possibility of deploying the network on different types of maps on which Google Maps stands out. In this way, a more realistic simulation can be achieved. The following devices can be displayed on the different maps:

i)Sensor Nodes: nodes in charge of taking the variables of the medium.

ii)Media Sensor Nodes: sensor capable of taking variables from the environment and its sensing unit is directional. It has a form of a cone that can be modified with the SenScript.

iii)Base Station (Sink): It is like a sensor node with the peculiarity that its battery is Infinite.

iv)Mobile: the mobile device can be driven through a route created with markers.

v)Markers: these markers can be used for different tasks such as:

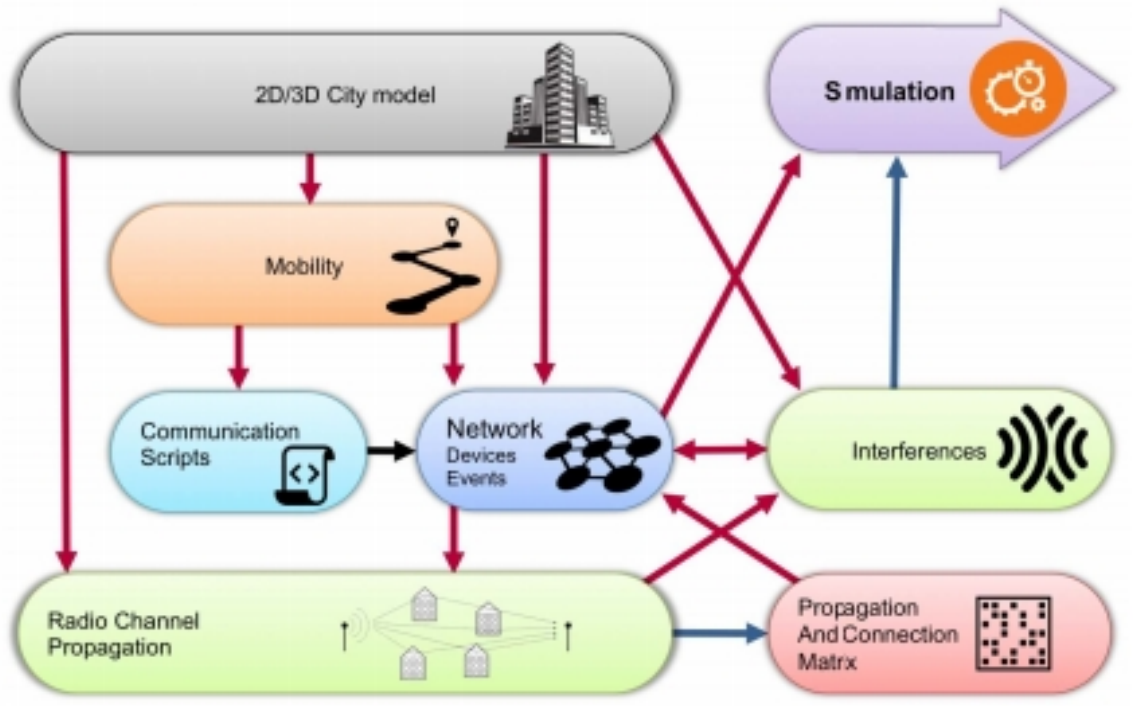
- Adding sensors can be done randomly delimiting the area on which you want to deploy.
- Creating routes.
- Adding or drawing buildings.

Moreover, by extending the Solver section of the menu, we can use different algorithms such as the Jarvis algorithm that, given a cloud of points, surrounds them with lines in the positive direction.

Objectives of Cupcarbon:

1) Study the deployment of wireless sensor networks by taking into account the mobility and the availability of the spectrum.
2) Simulate performances and services of a wireless sensor network in a 2D/3D realistic environment.
3) Study the feasibility of communication, the reliability of the network and its cost.
4) Detect any interference zones in order to improve the quality of the deployment.
5) Simulate accurately and quickly the radio propagation in a real urban environment.
6) Visualize the simulation result in order to debug and validate a developed algorithm.

Cupcarbon Architecture:



Example Performed in Cupcarbon:
The most attractive feature that CupCarbon has for WSN networks is the ability to graph the energy consumption and the battery level after the simulation. For this purpose, the simulation should be launched with the results option, of the simulation parameter selection menu, checked.

To test this functionality, we have executed one of the examples provided by developers (See Figure 3) during 25s. As we can see, the network contains 4 nodes with different roles. On the one hand, we have a node detector that will be in charge of monitoring the presence of the mobile node and transmitting the signal to the rest of the nodes. Router 1 and Router 2 are two nodes that only work by forwarding the signal to the sink node. Finally, the sink node receives all the signals from the rest of the network.

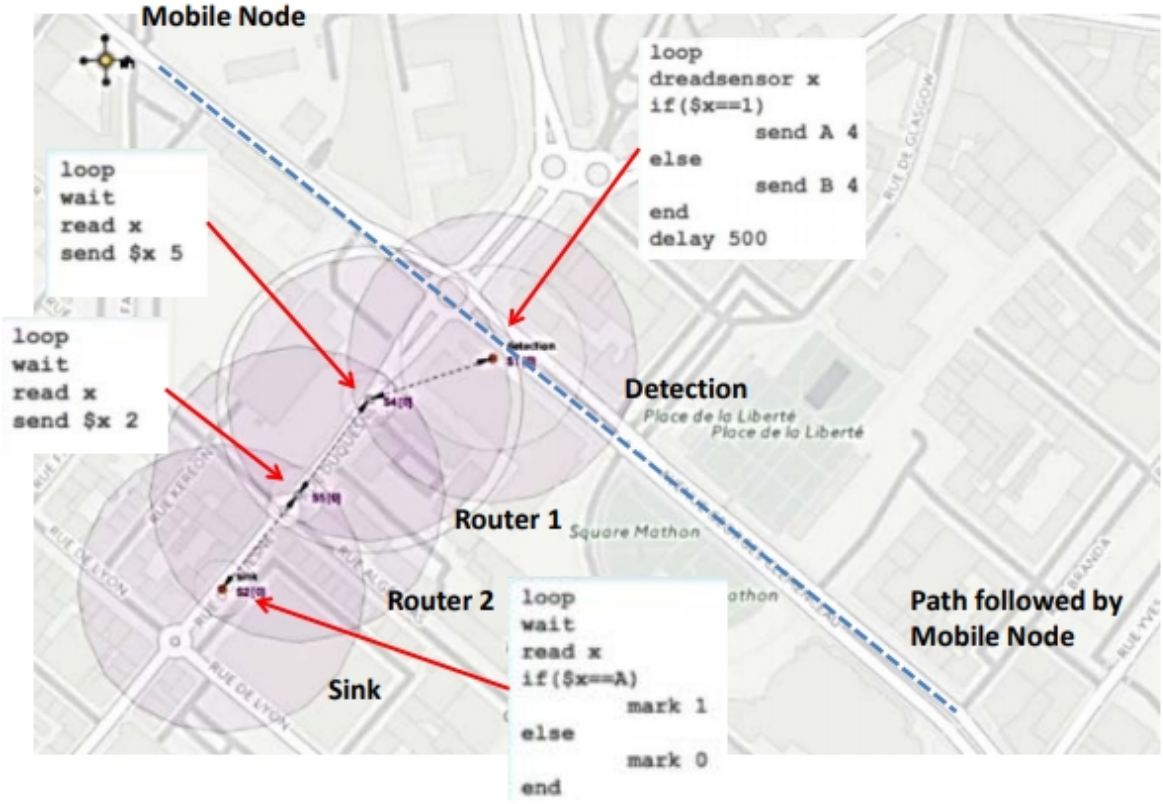
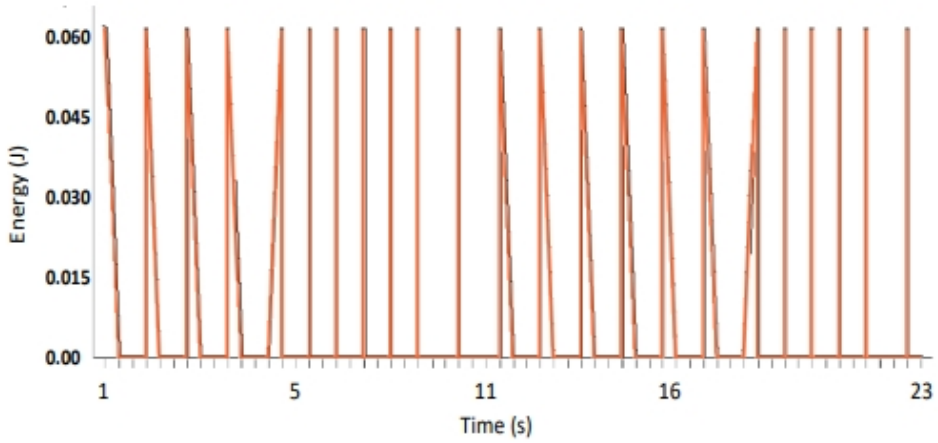
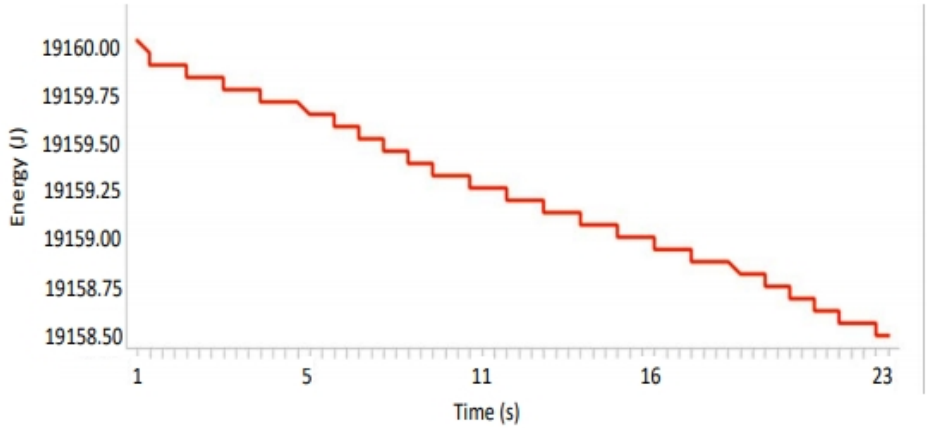


Figure 4 shows the scenario in different phases of the simulation. Figure 4a shows the proposed scenario during the simulation when the mobile node has not been detected, while Figure 4b shows the proposed scenario during the simulation when the mobile node has been detected. As we can see, when the mobile node is detected, the coverage area of the Detection node lights up in yellow.

a) b)

As a result, we obtain Figure 5 which shows the energy consumption in Joules of node 1 and Figure 6 which shows the remaining energy of node 1 as a function of the time. As we can see, when the node sends a message, it consumes 0.06 J (Figure 5). The node consumes energy due to the process of sending/receiving data and its own electric operation. So, the remaining energy in node will decrease as a function of the time (see Figure 6).





To define the behavior of each of the nodes that composes the network, it is necessary to create scripts using the SenScript window. The user guide describes the commands included in the SenScript language. However, if additional complex algorithms are required, they can be written in Java. Finally, the window called "Natural Event Generator" allows generating values that can be detected by sensors nodes such as temperature, humidity, etc.

Conclusion: Thus the operation, functions and working of different WSN opensource emulators like Cupcarbon and Contiki Cooja have been successfully studied and implemented.