|  |  |
| --- | --- |
|  |  |
|  | Sensor Network Playground Monitoring System |
|  |  |
|  |  |
|  | WSN Lab (WS 13/14)  Group number 1 Project Guide: Eugen Berlin  Group Members :  Arjun Maryankandy Rajendran  Konstantin März  Mohammed Refat Chowdhury  Pratyush Agnihotri  Rick Nitsche |
|  |  |





# 

# Introduction

The project is basically an implementation to degrade the greenhouse effect on plant using Wireless Sensors Networks. Concentration of greenhouse gases increased in last 10 years.. In result, gases like CO2 and CO increase with higher rate that can cause adverse effect on plants such as requires more water, some plants reduce their ability to do photosynthesis, plants become more vulnerable to insects, pests, and diseases.

## Problem Definition

To over the problem as mentioned above, we have to build a monitoring system which can save the plants by monitoring plant’s surrounding enviornement and maintaining the ideal conditions for plant.

## Goal

To accomplish this task, we learnt about living conditions for to known plants : Peperpmia and Kalachoe. Furthermore, we built a sensing system to monitor air humidity, temperature, CO2 concentration, soil moisture and light exposure and expolit actuators (fan, heater and lamp) to emulate those condition.

# Device / Driver information and plant living condtion

## Device/Driver Information

To use sensor in better way, it was required to study them both theoratically and pratically. We studied about the drivers and programmed them to know the behavior of them. We used following sensor and nodes[[1]](#_References) do develop a monitoring system :

|  |  |  |
| --- | --- | --- |
| Basic Node | : | XM1000 (aka TelosB) as basic node |
| On Board Sensors | : | Light sensors (Hamamatsu) |
|  |  | Temperature and humidity sensors (SHT11) |
| Co2 Sensor | : | DS1000 as gas sensor (CO, CO2 ) |
| Soil Moisture Sensor | : | VH400 as soil moisture sensor |
| Base Station | : | Raspberry Pi as base station |

## Plant living Condition

To create a better monitoring, it was required to know the ideal living conditions[[2]](#_References) for given plants. We indulged ourselves into a lot of research before we started the actual work on the “Save the Plant !”. We studied about the different kinds of necessary living requirements for plants as air humidity, CO2 concentration, temperature, soil moisture and light exposure that were needed to build our application.

# Soldering of Sensors

## Xm1000 pin structure

The XM1000 includes an expansion connector that allows the access to a number of pins in the microcontroller[[3]](#_References).

## Soldering Soil Moisture sensor



Soil moisture sensor (VH400) has three wire which has specific identity as

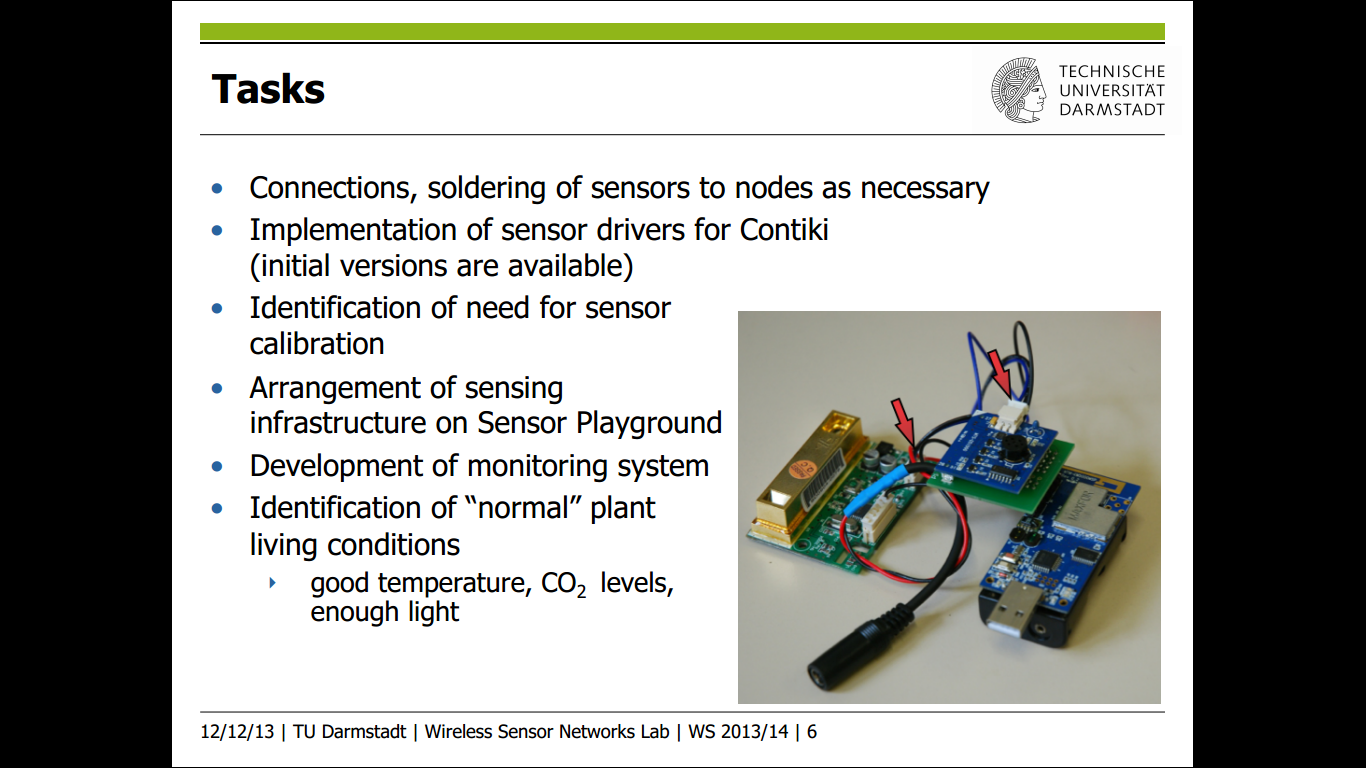
Bare : Ground , Red : Power, Black output. We used xm1000 pin structure to

solder moisture sensor. We soldered PIN 1(DVCC) : Red wire, PIN 3(ADC0) :

Black wire and PIN 9(GND) : Bare wire[[3]](#_References).

Figure 1 : Soldered moisture sensor

## Soldering CO2 sensor

****

For soldering Co2 sensor required xm1000 nodes, CO2 sensor (DS1000), PCB

board and SH-300-DG. We soldered all these compontent together and we used

DC 9V adapter jack to connect 5. DS1000 Board and SH-300-DC for power supply[[3]](#_References).

Figure 2 : Soldered CO2 sensor

# Test Bed Set up

To create better monitoring system, it is necessary to manage and place all nodes and sensor in such way so that monitoring can be done in efficient way. Therefore we proposed our test bed setup[[4]](#_References).However, a common test bed setup used by us. There were total 6 nodes. Two nodes are placed a left and right sides of the plants which were used as a sink. One CO2 sensor was placed between two plants and one on board sensor was placed at left side of plant. However, two dedicated soil moisture were used for left and right plants. [3009] On board Sensor (light, humidity and temperature), [3002] Co2 and [3007 and 3003] Moisture sensor (both left and right) collected a values and did

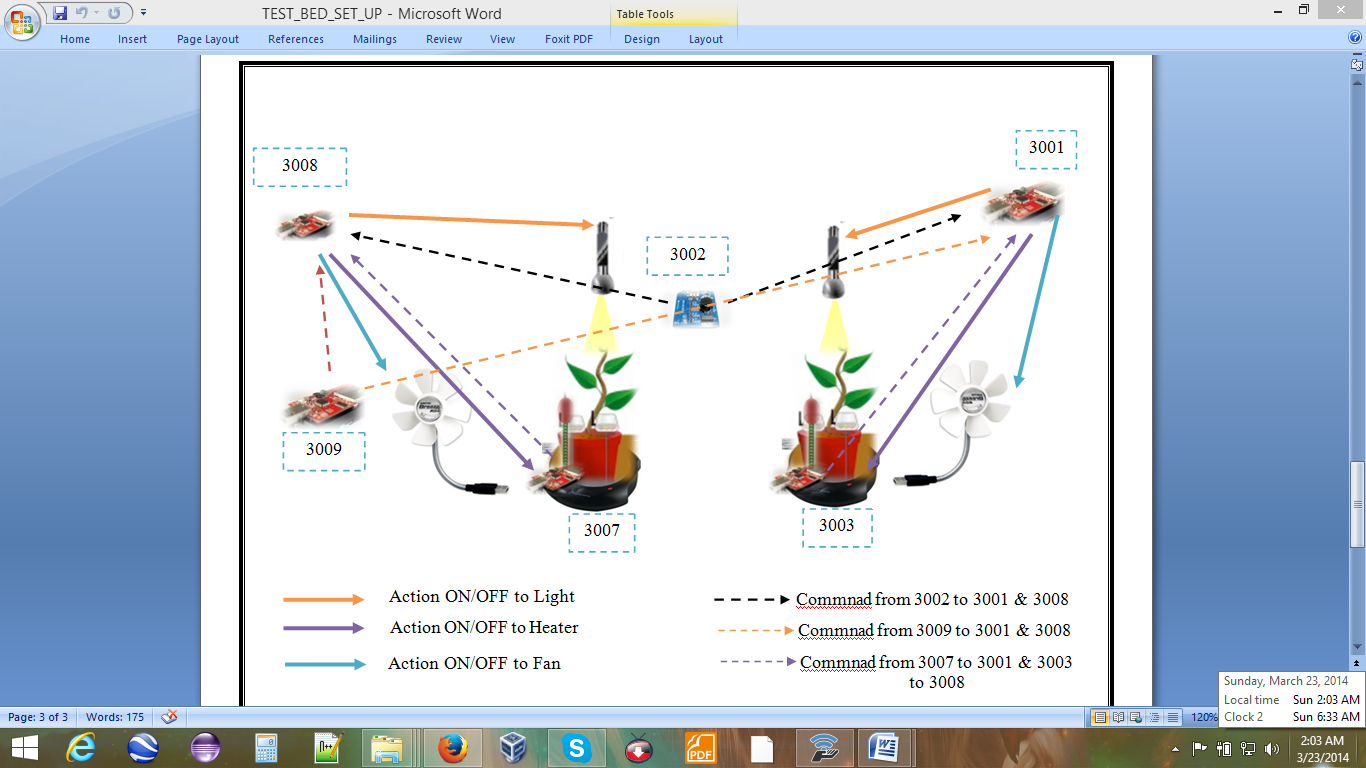


Figure 3 : Test Bed Setup

sampling. If value was below /Ok/ higher than threshold value then send command to sink 3001 and

3008. When commands was received by sink 3001 and 3008, it sent ON/OFF action to raspberry pi base station to handle corresponding actuators. Current states(actuator On/Off) stored in the sink 3001 and 3009 for respective plants. No action was sent by sink until new command was received.

# Formulas for calculation

Monitoring requires converting raw data into human readable forms which we result in efficient sampling and visualization. We used various formulas to convert raw data into correspondind sensor values as raw data of Light into LUX[[5]](http://en.wikipedia.org/wiki/Device_driver#cite_note-dev1-1), Humidity into percentage[[6]](http://en.wikipedia.org/wiki/Device_driver#cite_note-dev1-1), Temperature into ºC[[7]](http://en.wikipedia.org/wiki/Device_driver#cite_note-dev1-1), Co2 into ppm[[8]](http://en.wikipedia.org/wiki/Device_driver#cite_note-dev1-1), and soil moisture into VWC (Volumetric Water Content) [[9]](http://en.wikipedia.org/wiki/Device_driver#cite_note-dev1-1). We converter these value for monitoring and visualization purpose while we used raw data for comparison with threshold value and sending command correspondingly.

# Usage of Actuator

We used raw data instead of converted values for comparison with threshold value. When value was different from idea value that ON/OFF[[10]](http://en.wikipedia.org/wiki/Device_driver#cite_note-dev1-1) command was sent to Raspberry Pi ( base station) on respective actuators. When soil moisture was high than PG: HEAT ON was sent other wise PG: HEAT OFF was sent. Similiarly, if light was low below threshold value than PG: LIGHT ON otherwise PG: LIGHT OFF and CO2 was high than PG: FAN ON else PG: : HEAT OFF. We can use actuators for temperature and humidity but temperature and humidity don’t change so frequently. They were almost constant. Hence, we only monitored temperature adn humidity but no command was used for actuators.

# Principal of Application

To support modularity each sensor type has its own .c-file. It leaves a possibility to add additional sensor types in the future. A settings file is used to define constant values such as node ID's, value thresholds, measuring periods etc. After 120 seconds of initialization the nodes begin to operate. Every 5 seconds measurements are taken; depending on the node-ID a different measurement procedure is invoked. To reduce energy consumption calculations are done on the sensor nodes, rather than sending measured values to the sink node every time measurements are taken. Commands to manipulate the actuators are only send to the sink node if a certain value is above or below its threshold. To prevent packet losses reliable single-hop unicast (runicast) primitive is used. The runicast primitive uses acknowledgments and retransmissions to ensure that packets are successfully received. The maximum number of retransmissions used in the project was 10. Runicast doesn’t provide carrier sensing, hence different and short delays before sending were implemented for each sensor type in order to prevent occasional message loses due to simultaneous transmissions on the carrier.

# Sampling and Visualization

To be done

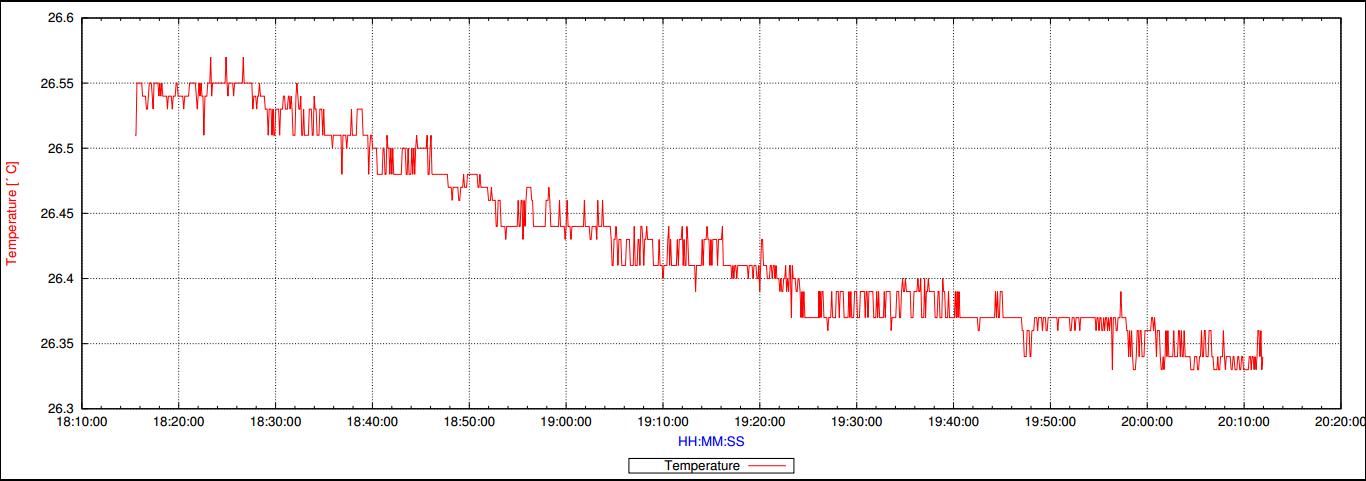


Figure 4 : Temperature

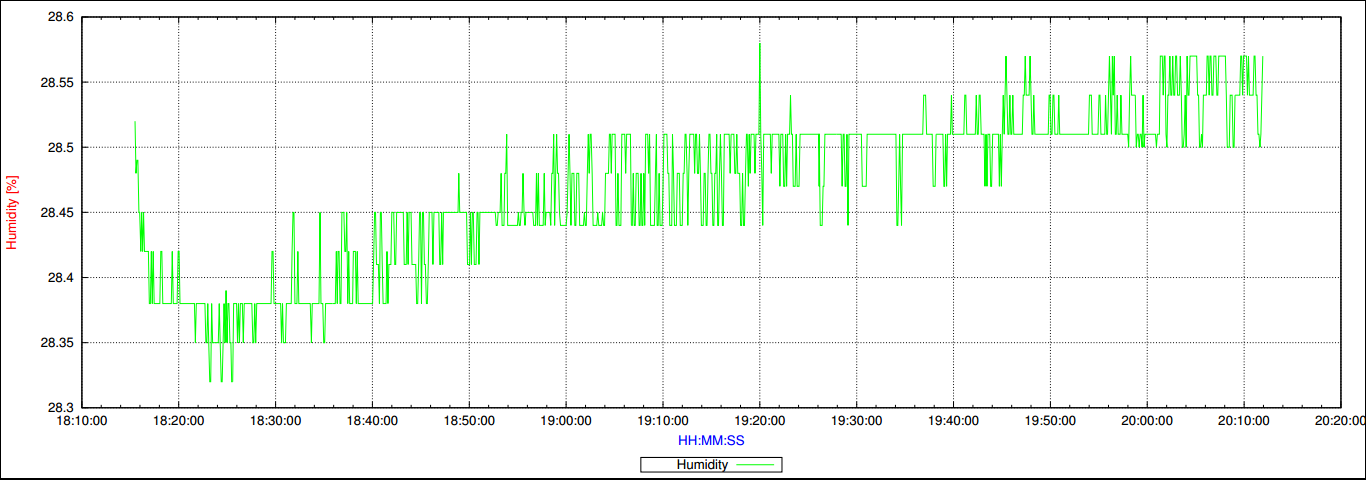


Figure 5 : Humidity

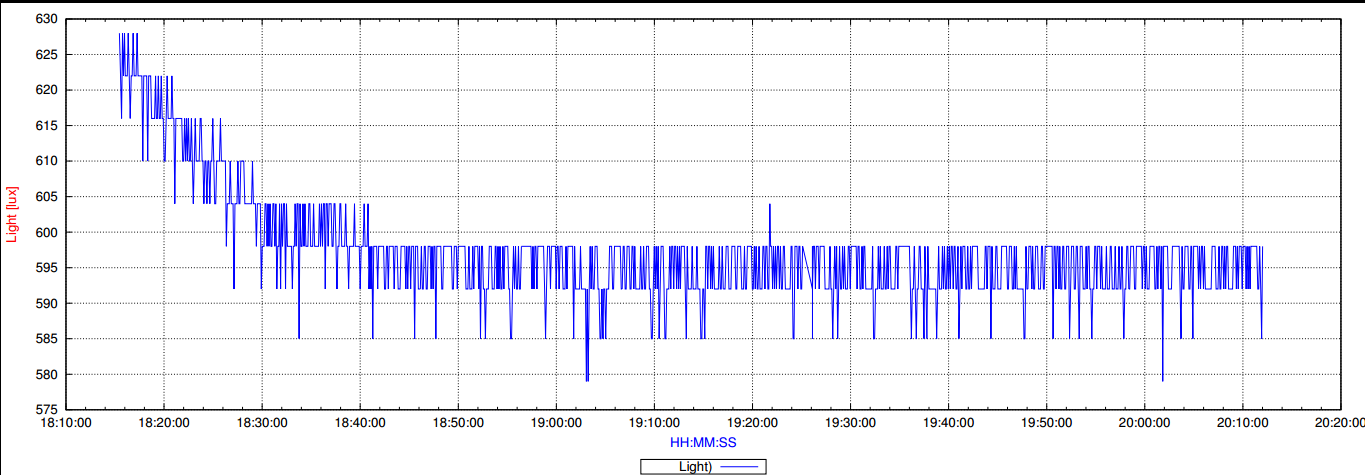


Figure 6 : Light

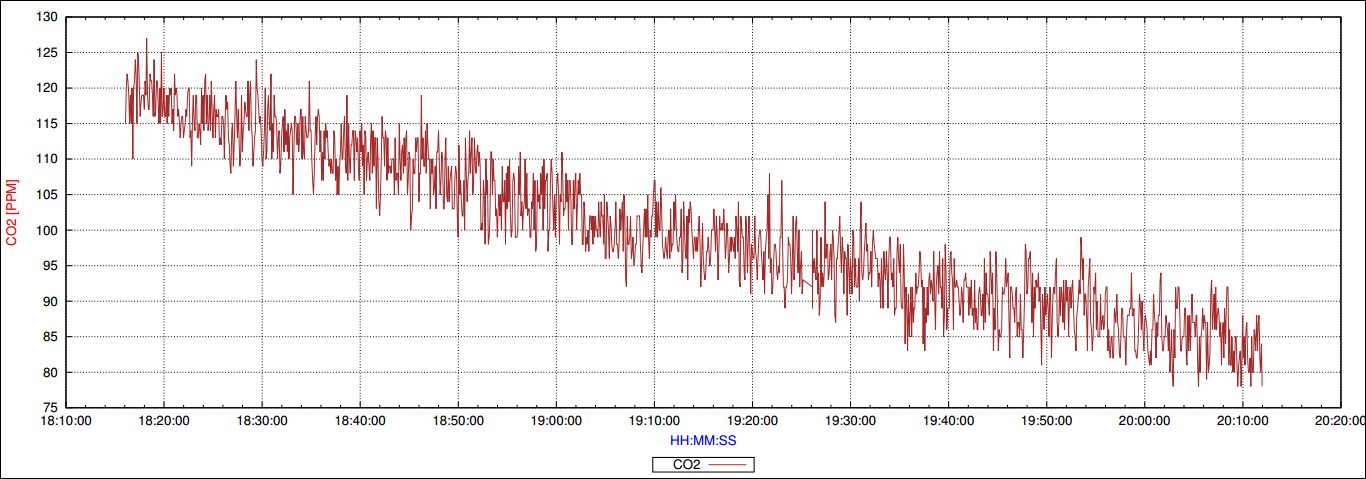


Figure 7 : CO2

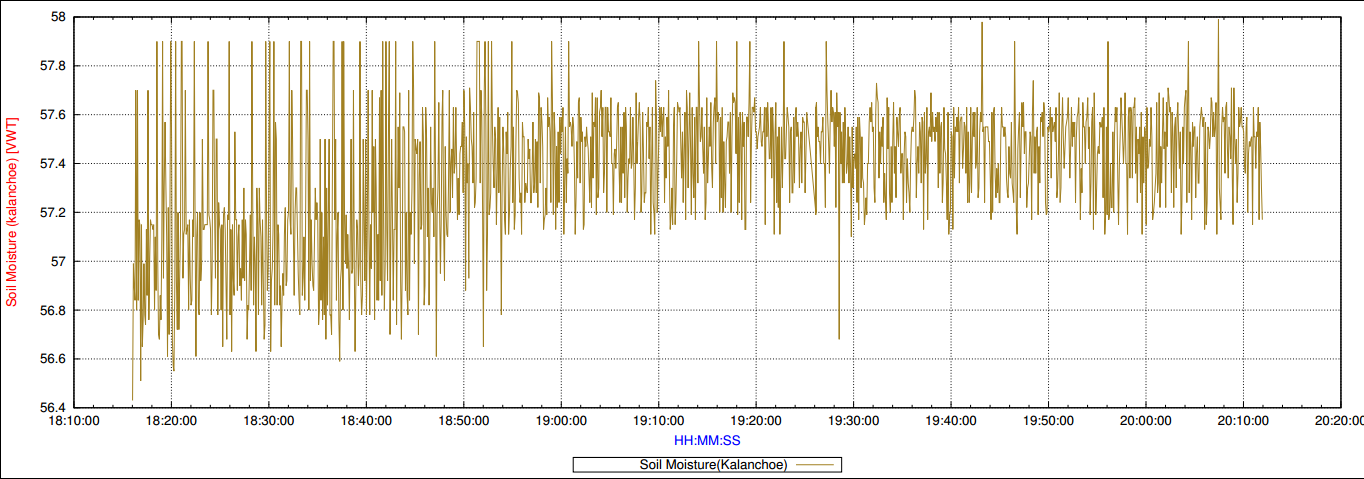


Figure 8 : Soil Moisture Kalanchoe

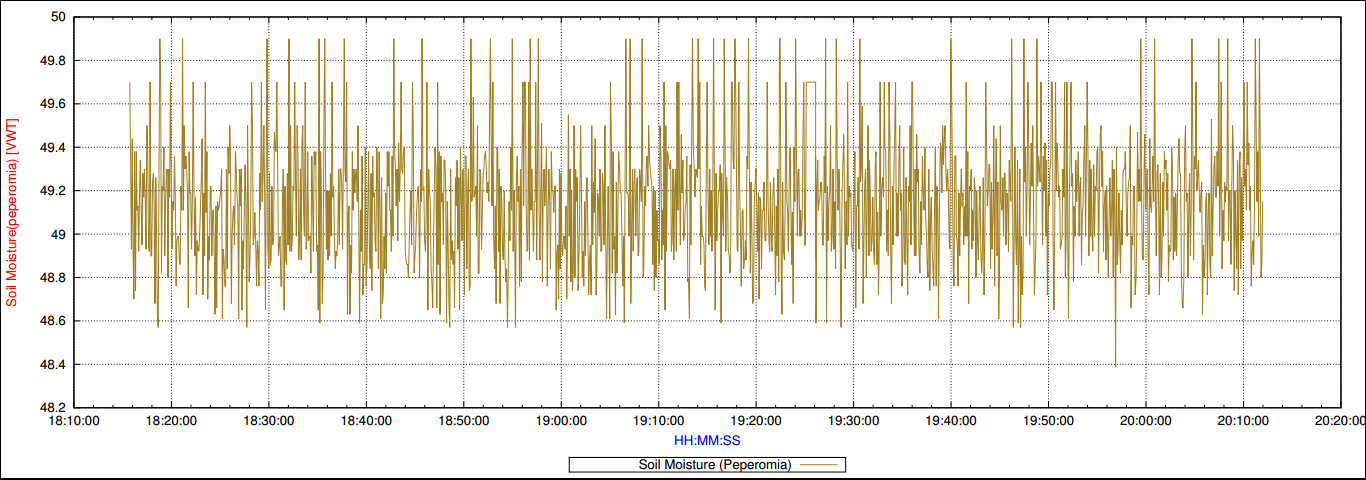


Figure 9 : Soil Moisture Peperomia

# Testing of Application

To make system more reliable and efficient, it is required to do testing of application. Testing does not only help in solving bugs but also make code more optimizie and provide efficient application. We tested our application in various senarios by applying different test cases[[11]](#_References) to check how actuators behave in these scenarios. Testing included checking the actuator functionality, run-time monitoring and system responsiveness by increasing/decreaseing soil moisture, lowering/highering light exposure, increasing/decreasing CO2 level and rigorous testing.

# References

1. Nodes and Sensors - ..\Project\_Documents\Plant\_condition\
2. Plant Living Condition - ..\Project\_Documents\Plant\_condition\
3. Soldering of Sensors - ..\Project\_Documents\Plant\_condition\
4. TEST\_BED\_SET\_UP - .. \Project\_Documents\Test bed Step\
5. http://www.advanticsys.com/wiki/index.php?title=Hamamatsu%C2%AE\_S1087\_Series
6. http://www.advanticsys.com/wiki/index.php?title=Sensirion%C2%AE\_SHT11
7. DS1000 data sheet - ..\Project\_Documents\Sensors\_Datasheet\
8. SHT11 data sheet - ..\Project\_Documents\Sensors\_Datasheet\
9. http://www.vegetronix.com/Products/VH400/VH400-Piecewise-Curve.phtml
10. Actuators condition - ..\\Project\_Documents\Actuators\_condition\
11. Test\_Cases - ..\Project\_Documents\Test\_cases\