CS-308-2014 Final Report

Greenhouse Temperature Regulation

Team Name: Proxymorons
Team Code: TH-1

Aditya Bhandari - 100050008
Akshay Gaikwad - 100050010
Kanishk Parihar - 100050018
Sameer Kumar Agrawal - 100050021

Table of Contents

1. Introduction

Name of the project: **Greenhouse Temperature Regulation**

Motivation:

We see that the growth of plants placed in open is hampered by excessive temperature during day time. This happens very frequently in extreme climates such as that in northern India. In summer and especially from 11 am to 4 pm the temperatures is extreme and plants cannot bear such high temperatures, some even die. So we have to place the plants in the shade during that time.

Thus the idea has its roots from here that, if we could give shade to the plant and thus reduce the amount of sunlight in a close area near the plant, the temperature will reduce. If the temperature doesn't come inside the desirable limit for the plant by giving shade, then it is almost useless. So, we decided to use a PC fan, an imitation of coolers in houses.

Working of our project:

If the temperature is below a certain threshold (T_1), the plant is kept uncovered. If the temperature increases above T_1, the rolling mechanism covers the plant with a green messy cloth (with 50% granularity). If the temperature further increases above another threshold (T_2) greater than T_1, the PC fan is switched on. In the same way, if the temperature falls down below T_2, the PC fan is switched off and if it falls below T_1, the rolling mechanism removes the cover over the plant.

Uses:

- The main aim of the project is to regulate the temperature of the model greenhouse by introducing a shade over the plant and using PC fan for air circulation.
- Our project is useful for the people living in extreme temperature conditions like Rajasthan and North India.
- Also we have a simple interface and the cost of the things is quite low so as to be used by village households.
- Our project is scalable. That is, a large number of mini-greenhouses can be controlled at once. Also the size of greenhouse can be increased as necessary.

2. Problem Statement

The problem in front of us was to maintain temperature less than a fixed threshold temperature in a mini-greenhouse. That is, the user specifies the temperature he wants to maintain in the surrounding of plant so that its growth is not affected. So our aim was to reduce the temperature of the greenhouse in case it increases in order to bring it close to the value specified by the user.

The work that we have currently finished accomplishes all of the above stated initial aims. Our system is able to detect rise of temperature and trigger the motors to roll the cover and provide shade to the plant. It is also able to turn on the cooling fan when the temperature still keeps increasing. Similarly, it is able to detect decrease in temperature to turn off the cooling fan and to unroll the shade providing cover on further decrease.

We came up with more ideas on what more (apart from the initially aimed functionalities) can be done in future, as the work progressed. But this is explained in the later part of the report.

3. Requirements

3.1 Functional Requirements

- The user specifies the temperature requirements for the plant.
- The temperature sensor (LM 35) takes the reading around the plant and initiates the appropriate reaction. According to the requirements,
    1. When the temperature rises above T_1, the shade comes over the plant. When it drops below T_1, it rolls back.
    2. When the temperature rises above T_2 ($T_2 > T_1$), the cooling fan is switched on to increase the rate of cooling. When temperature drops below T_2, it is switched off.

3.2 Non-Functional Requirements

1. We need to ensure that the plant shade does not in any way interfere with the growth of the plant in the greenhouse.
2. Opening and closing mechanism should not take much space, so currently a rolling mechanism has been used.
3. The material used for shade should be easily rollable.
4. The thread holding the shade should be highly tensile and non-elastic.

3.3 Hardware Requirements

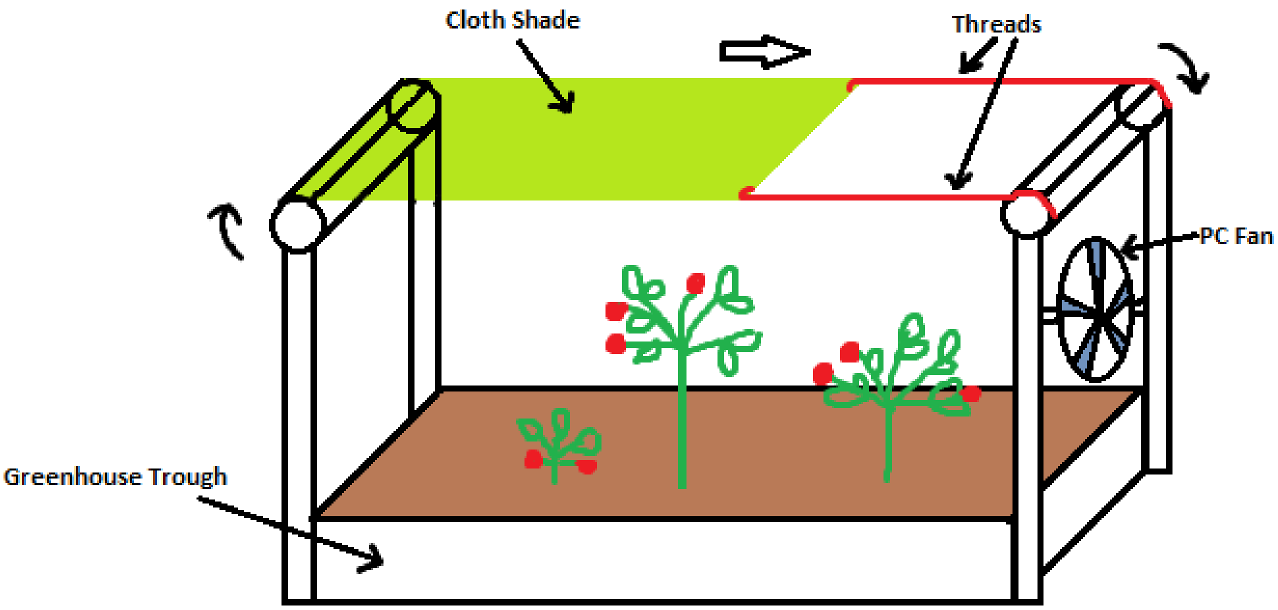
1. Greenhouse trough
2. Firebird bot (FB5)
3. L293D Motor Driver circuit
4. 2 XBEE chips and an XBEE USB module
5. Plastic (PVC) pipes for the greenhouse model
6. A green messy cloth with 50% granularity as the shade
7. Temperature sensor (LM 35)
8. PC fan
9. Threads with high tensile strength
10. 2 rotating rods (rollers)
11. 2 gears and 4 ball bearings
12. Conveyor belt

3.4 Software Requirements

1. Keil uVision4 – for building the target file for FB5
2. X-CTU – for configuring the XBEE
3. Python-2.7.6 – for running the web server
4. Python library “pyserial-2.7.win32” – for serial communication
5. Programming Languages: C, Python

4. System Design

The overall design and working of our project is shown in the diagram below:

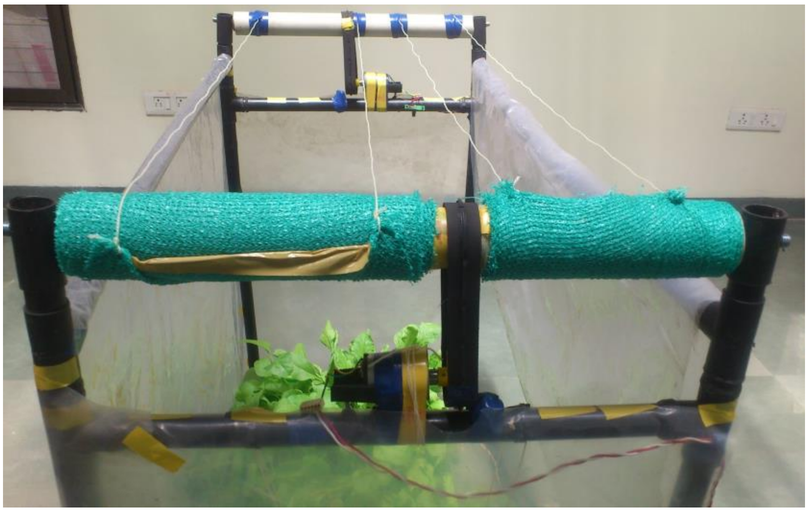


Closing of the plant shade

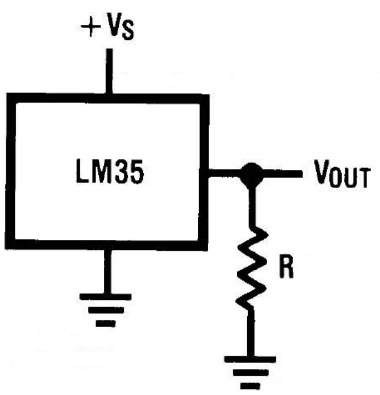
Construction:

- The greenhouse model shown was built using PVC pipes and the rolling pipes at the two ends were mounted using ball bearings mounted on the vertical pipes. The PC fan was connected using threads on one side of the model. The overall construction of the greenhouse model can be seen in the following image:

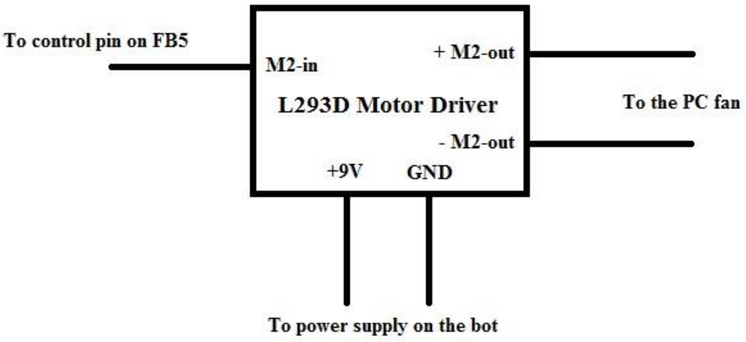
- The motors at both sides of the model were mounted just below the rolling pipes and connected to them using conveyor belts. These motors are connected to 9V power supply from their original location on the FB5 bot.

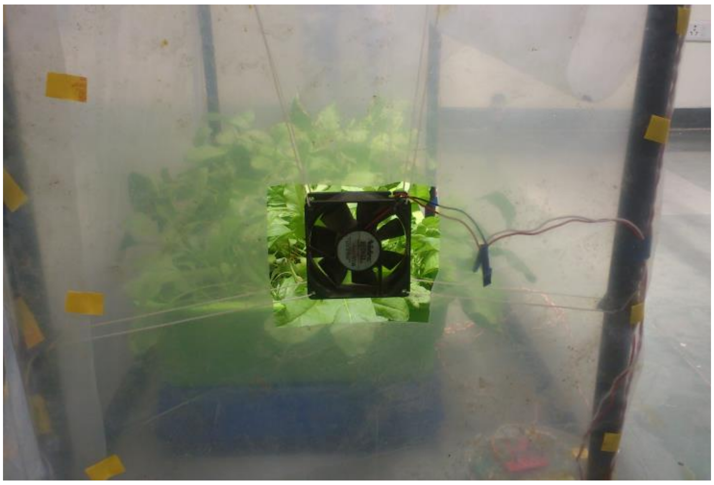


- The temperature sensor circuit consists of breadboard, the temperature sensor - LM 35 and a 4.4K resistor. The resistor is connected between V_{out} of the sensor and GND. This circuit is connected to one of the IR sensors on the Firebird (FB5) bot.

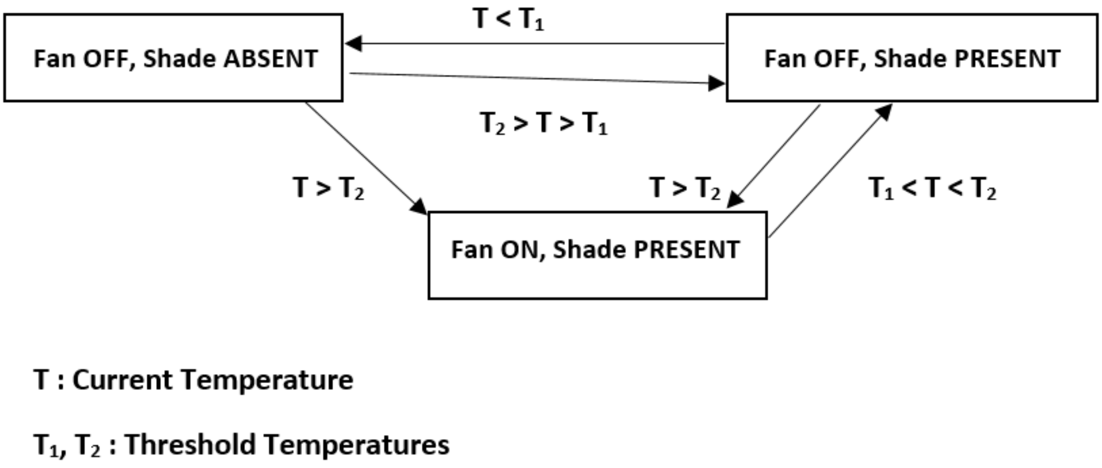


- The PC fan is connected to a motor driver circuit (L293D) and this circuit is connected to the servo pod connector on the FB5 bot.





Finite-State Machine (FSM):
The FSM of our system is shown below.



5. Working of the System and Test results

Working of the system:

The inputs to the system are the values of **two threshold temperatures ($T_1 < T_2$)** and a **temperature buffer (C)** given by the user, in addition to the **current temperature reading (T)** provided by the temperature sensor.
There are **four major scenarios** in the working of our system based on the value of T.

1. $T_2 > T > T_1+C$:
   As the current temperature increases beyond the 1st threshold temperature, both the motors will start rotating in clockwise direction thus causing the shade on top of the plant to unroll. We have used the temperature buffer (C) so as to avoid the problem of continuous opening and closing of shade that can be caused by fluctuation of the temperature around the threshold value.

2. $T > T_2+C$:
   As the temperature increases beyond the 2nd threshold value, the cooling fan connected to the side of the greenhouse will be switched on. The temperature buffer is used here as well for the same reason as mentioned above.

3. $T_1 < T < T_2$:
   For change in temperature in the reverse direction, as the temperature falls below the 2nd threshold value, the cooling fan is switched off.

4. $T < T_1$:
   As the temperature falls below the 1st threshold, both the motors will start rotating in anti-clockwise direction, thus causing the shade to close and roll back in position over the pipe.

Testing:

We used a hair dryer to control the temperature in our greenhouse model for testing purposes. According to the temperature, appropriate actions were automatically initiated.

T_1: 1st threshold temperature
T_2: 2nd threshold temperature
C: Temperature buffer
T: Current temperature (sensor reading)

The following results were obtained as expected:

Temperature value	Action
Temperature increases from low to medium level: $T > T_1 + C$	Shade comes over the plant by unrolling of the green cloth
Temperature increases from medium to high level: $T > T_2+C$	PC Fan gets switched on
Temperature decreases from high to medium level: $T < T_2$	PC Fan gets switched off
Temperature decreases from medium to low level: $T < T_1$	Shade is removed from over the plant by rolling the green shade back in place

6. Discussion of System

a) What all components of our project worked as per the plan:
- Both the components mentioned in our plan - the opening/closing of the shade and the switching on/off of the PC fan - worked. But, the rolling mechanism of the shade did not work as expected and can be improved. The belt and the motor connection gets loosen after using it many times. The threads get stretched so the curtains goes lower a bit.

b) What we added more than discussed in SRS:
- We developed a web interface for the user to input the threshold and buffer values to be sent to the Firebird.

c) Changes made in plan from SRS:
We made the following two changes to the proposed plan in the SRS:

1. **Change in the rolling mechanism**: We setup the motors at the center of rolling pipe instead of at one side thus, having to divide the shade into two parts. We did this because it was easy to fix the motor on the existing pipe of the greenhouse structure and connect the conveyor belt properly. If we would have stuck to our proposed mechanism with motors at

one side of the pipe, we would have had to create a separate rigid platform for the motor and use a longer conveyor belt.

2. **Conveyor belts**: We used conveyor belts instead of gears for connecting the rolling pipes to the motors. This was done because we were unable to get the gears of the correct size that could have been connected between the motor and rolling pipe.

7. Future Work

1. **Possible extensions:**
   Hardware:
   - **Better rolling** mechanism:
     - Fixing the motors and conveyor belts on one side of the rolling pipes instead of in the center
     - Not allowing the conveyor belt to slip out from the gear or the rolling pipe
   - Using a **light sensor** to detect the amount of sunlight and if it is above a certain threshold, cover the plant with the shade
   - Use a **humidity sensor** and switch on the fan in case of high humidity
   - Use a **water sprinkler** for better cooling in case of high temperatures or in case of low humidity
   - Use of **multiple layers of shades** which can be intelligently used by knowing the direction of the sunlight and the areas already in shade
   - Use of **two PC fans** mounted on opposite sides of the greenhouse for better ventilation
   - Look for efficient and low cost ways of cooling, for example, geothermal cooling

   Software:
   - Develop an **Android application** for the server
   - Provide pre-defined temperature parameters for different plants on the server for quick access to the user

2. **Re-usable hardware components:**
   - Greenhouse model
   - Rolling pipes and ball bearings
   - PC fan
   - Circuit components: breadboard, temperature sensor (LM 35), resistor
   - Motor driver circuit (L293D)
   - Firebird FB5

8. Conclusions

- The rolling mechanism used in our project is low cost and can be controlled easily.
- Covering the plant helps control temperature and sunlight, thus not affecting the plant's growth.
- The cooling fan can be coupled with wet "*khus*" curtains for additional cooling.
- The project is low cost and easily scalable.
- It has an intuitive and easy-to-use user interface, for which only basic knowledge of how to use a computer is required.

9. References

1. http://www.ces.ncsu.edu/depts/hort/consumer/weather/tempeffect-plants.html
2. http://www.aces.uiuc.edu/vista/html_pubs/hydro/require.html