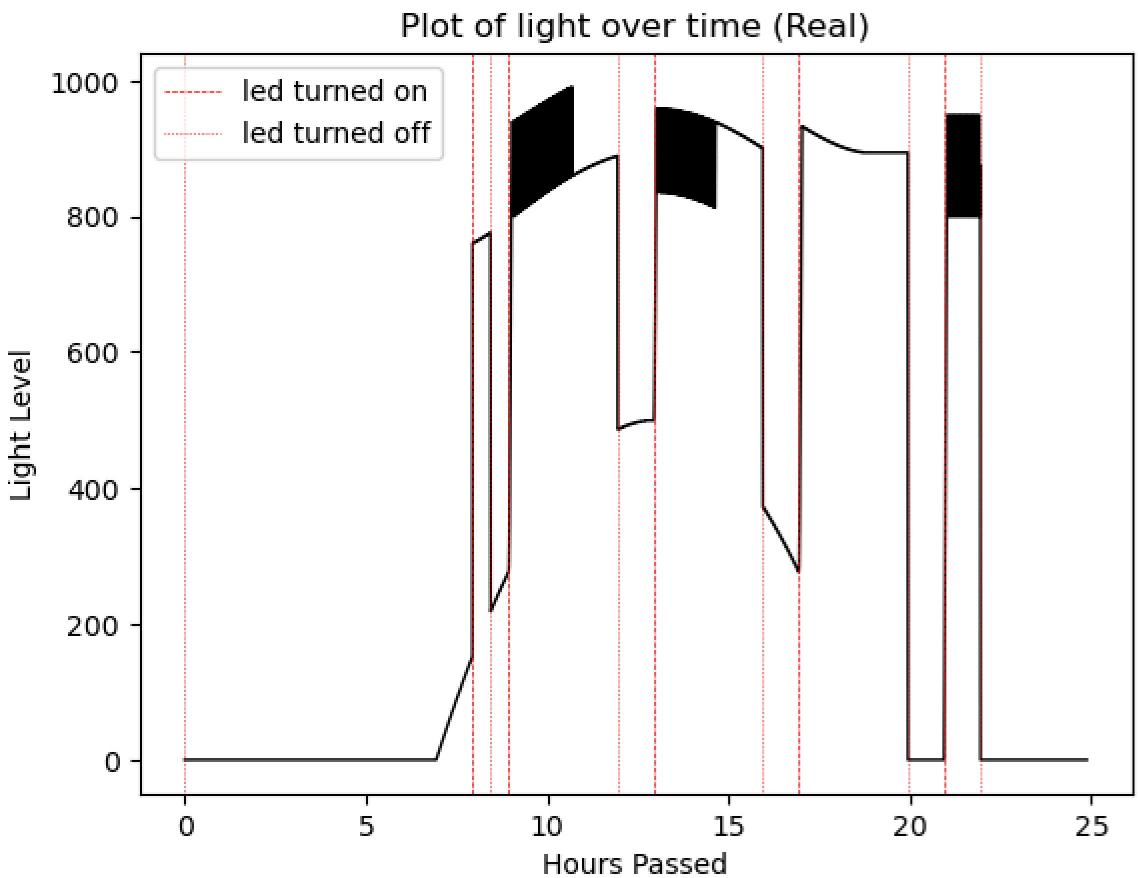
Part 4

Simulation Graphs

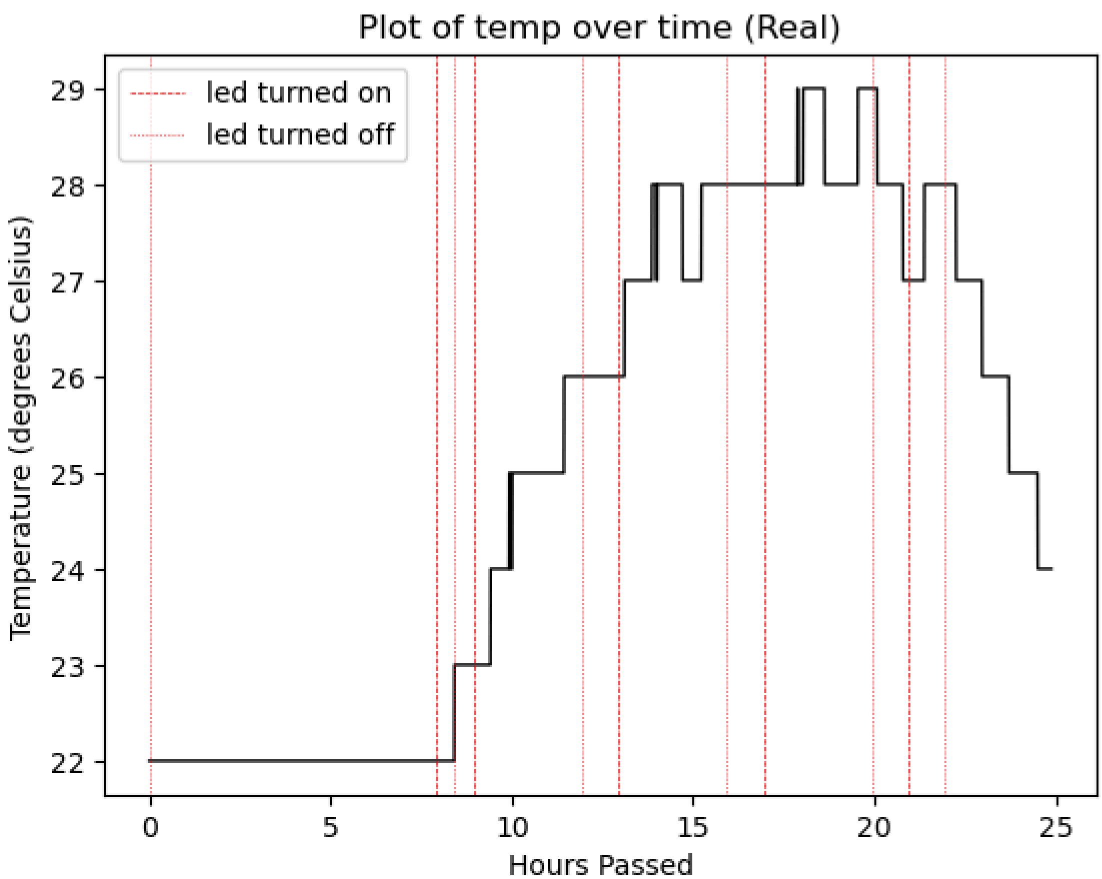
Start Time: 12:00 AM

Agent: Greenhouse Agent (FSM version, Layered)

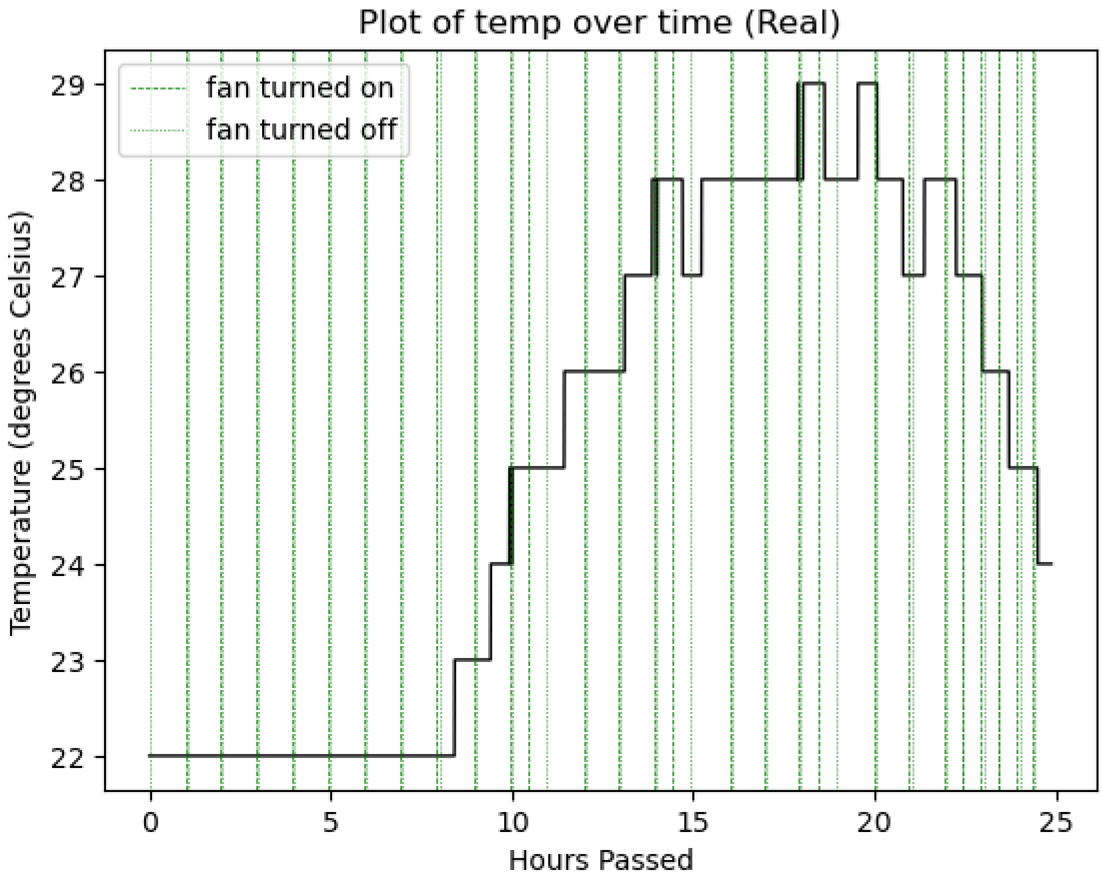
Light Level



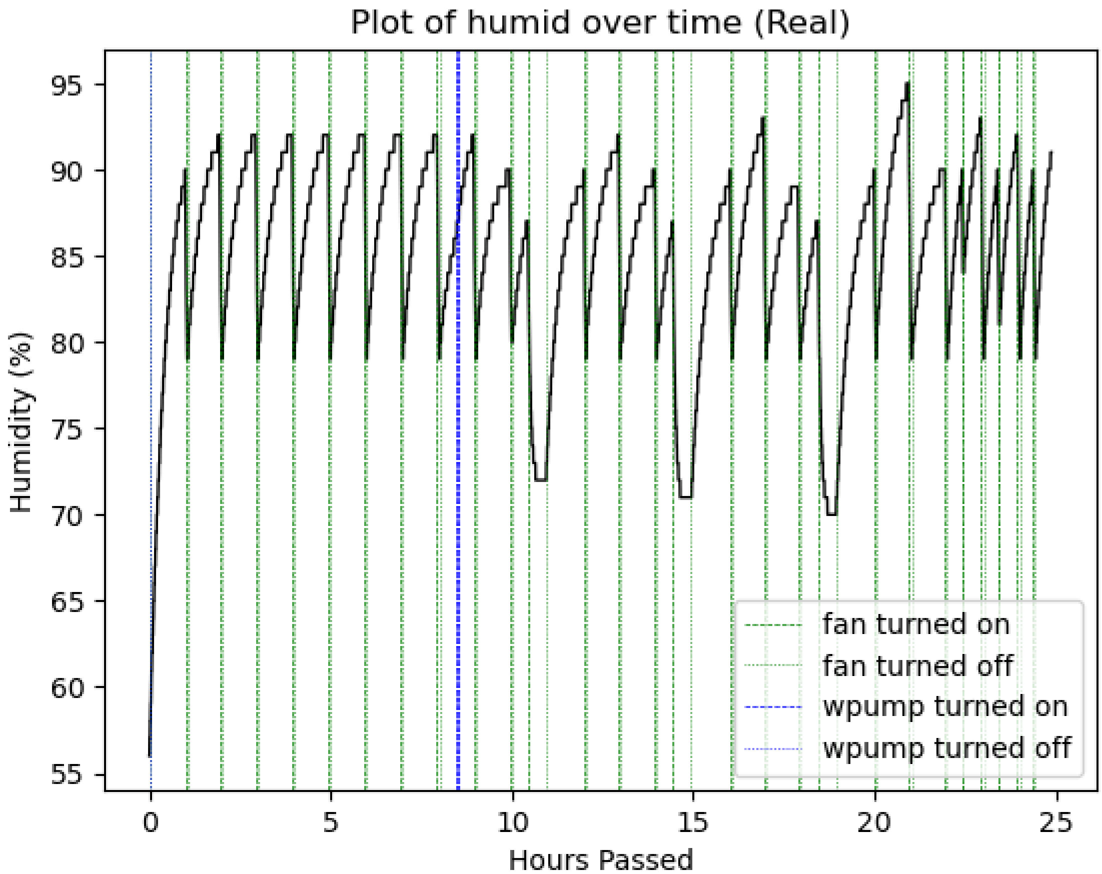
Temperature + LED



Temperature + Fan

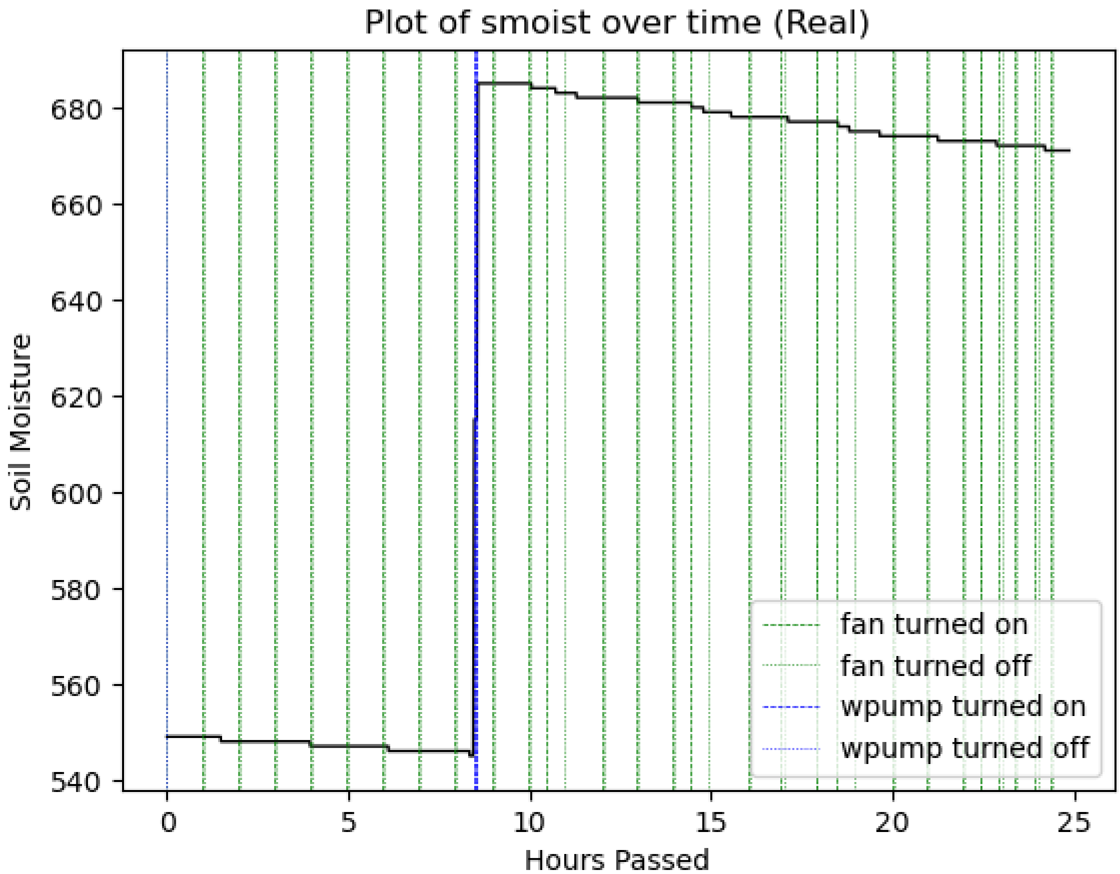


Humidity

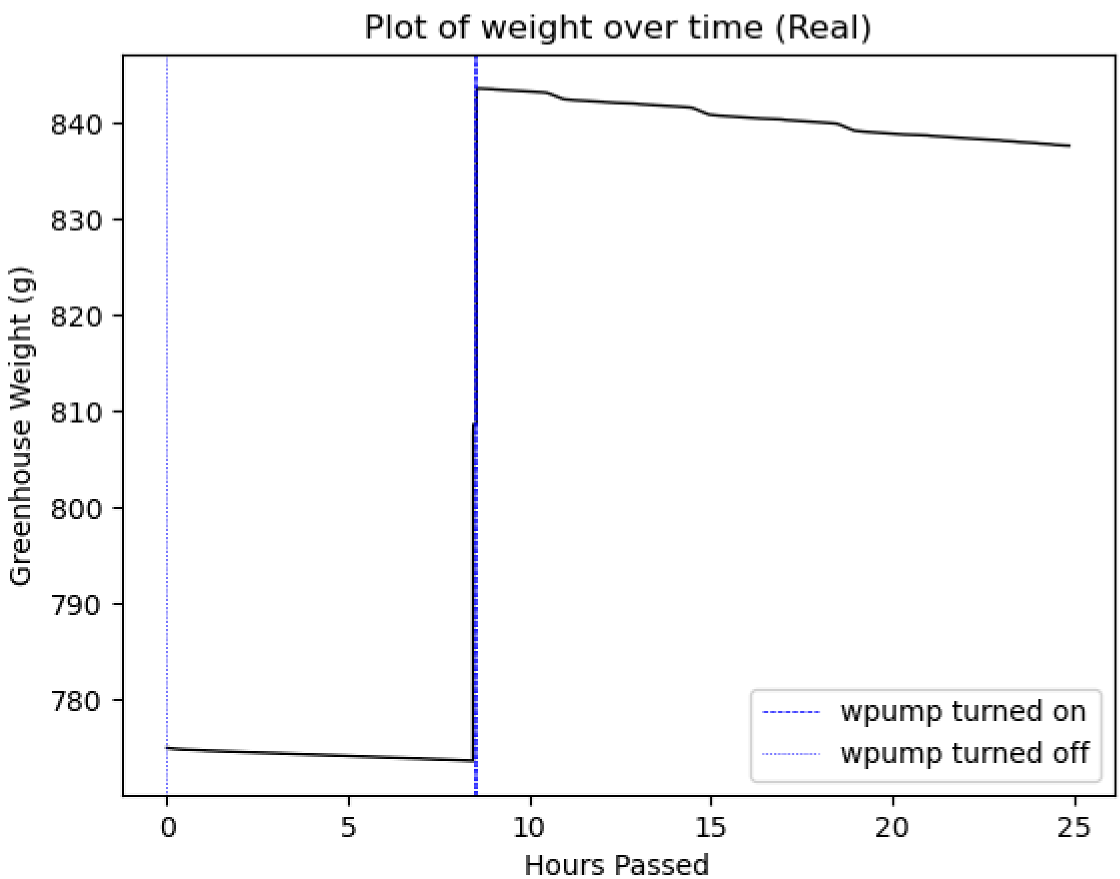


Plot of humid over time (Real)
95
90
85
80
75
70
65
60
55
Humidity (%)
0
5
10
15
20
25
Hours Passed
fan turned on
fan turned off
wpump turned on
wpump turned off

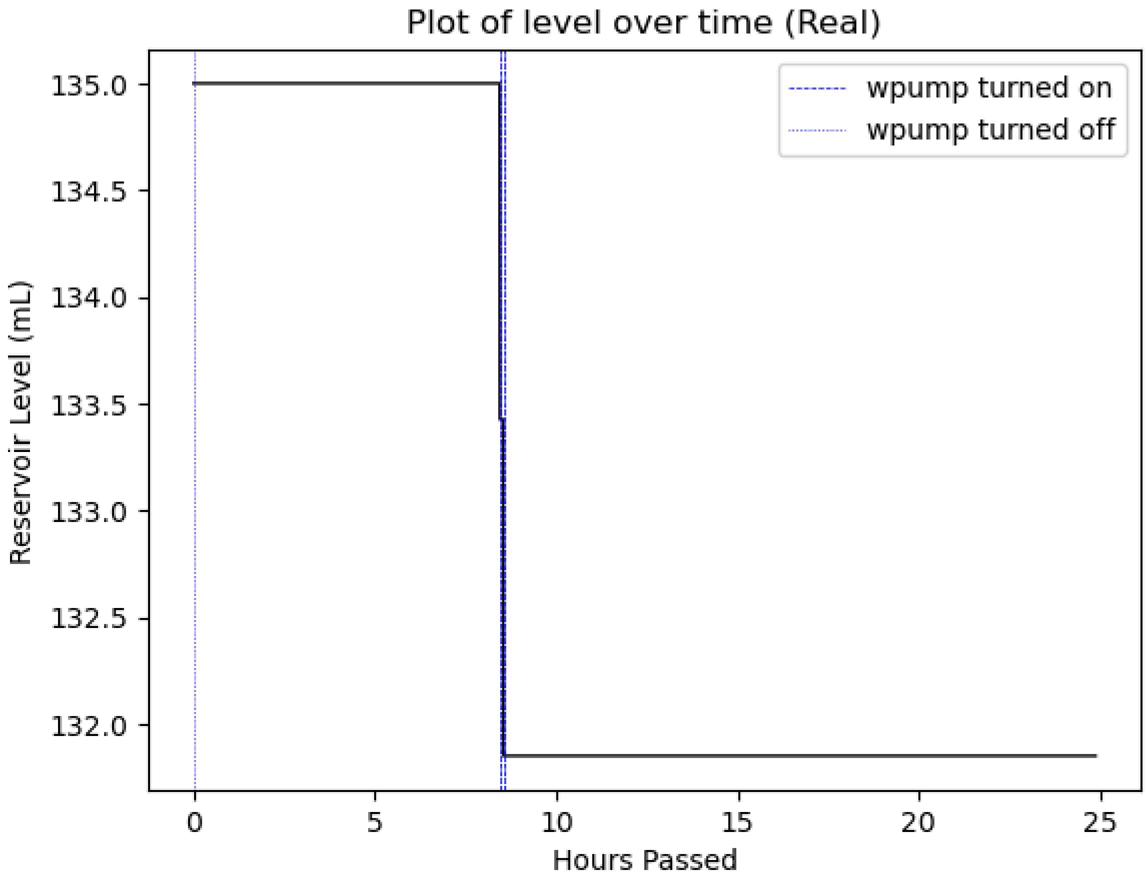
Soil Moisture



Weight


Plot of weight over time (Real)
Greenhouse Weight (g)
Hours Passed
wpump turned on
wpump turned off
840
830
820
810
800
790
780
0
5
10
15
20
25

Water Level



Plot of level over time (Real)
wpump turned on
wpump turned off
Reservoir Level (mL)
135.0
134.5
134.0
133.5
133.0
132.5
132.0
0
5
10
15
20
25
Hours Passed

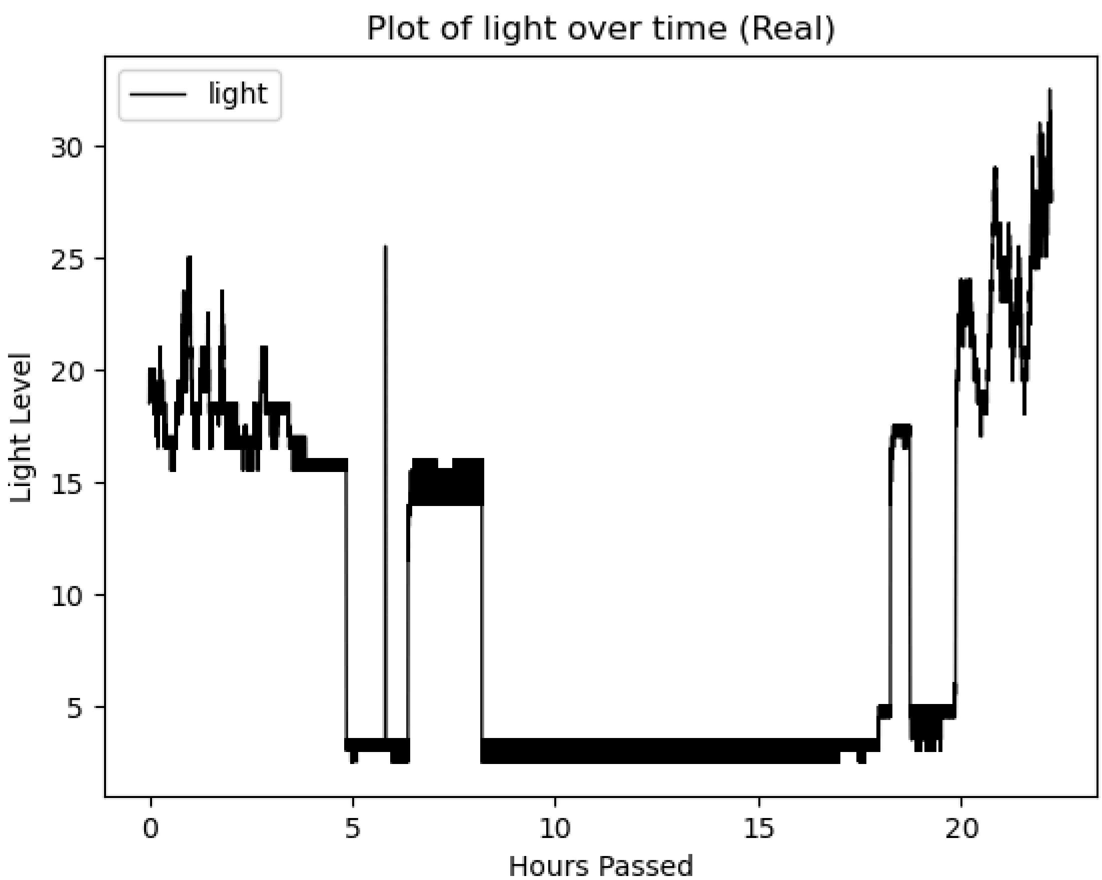
Real Hardware Graphs

Session 1

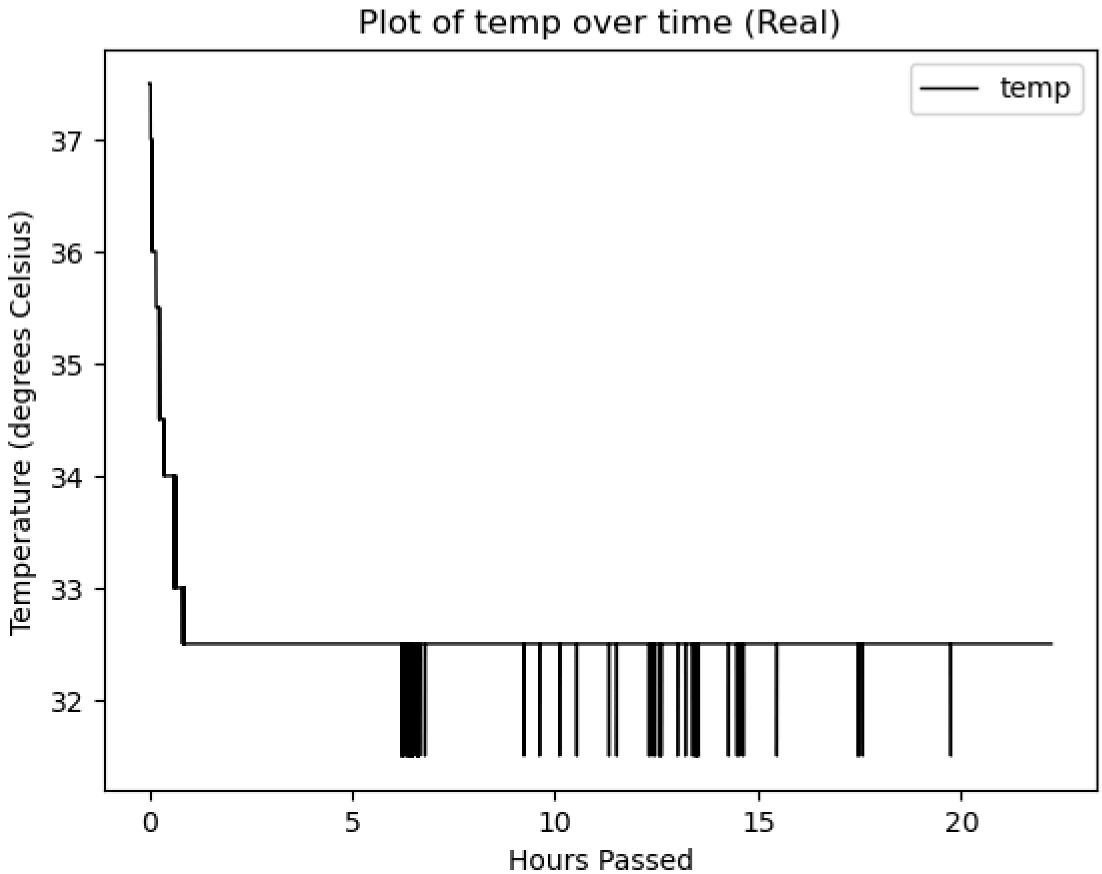
Start Time: ~3:00 PM

Agent: NONE

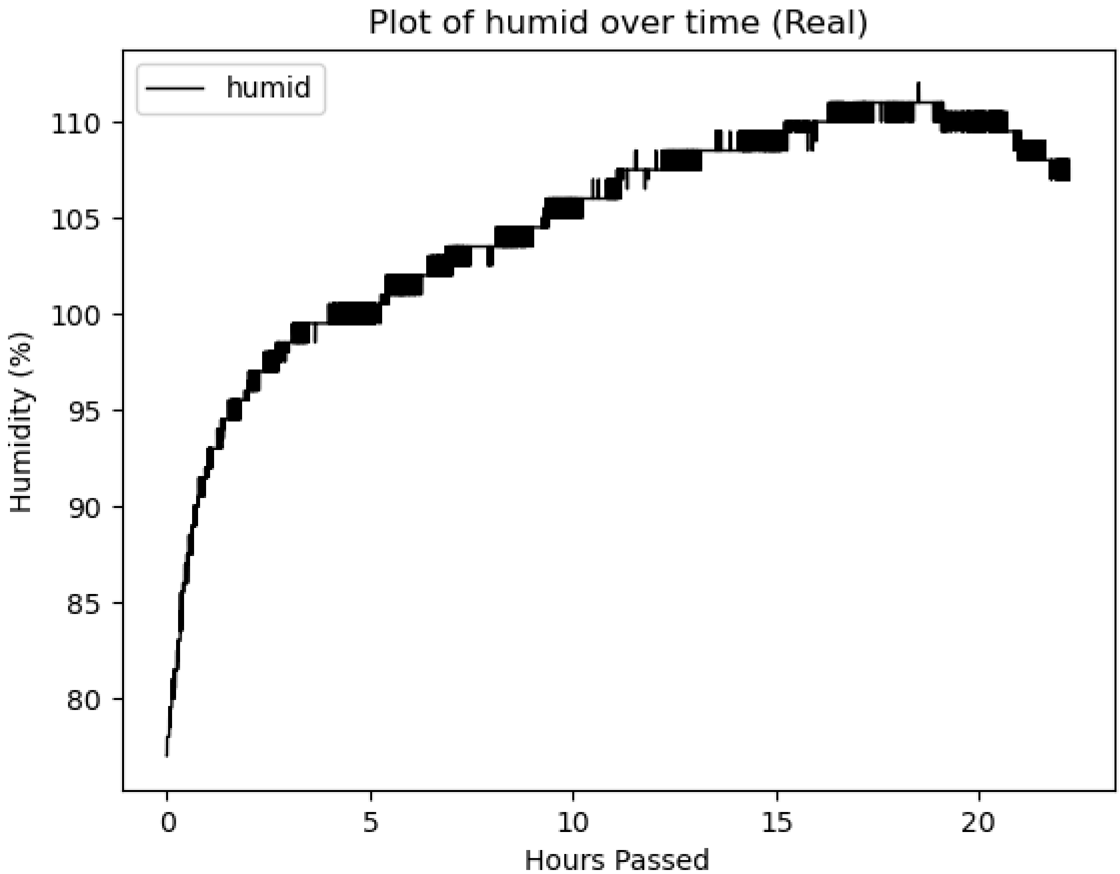
Light Level



Temperature

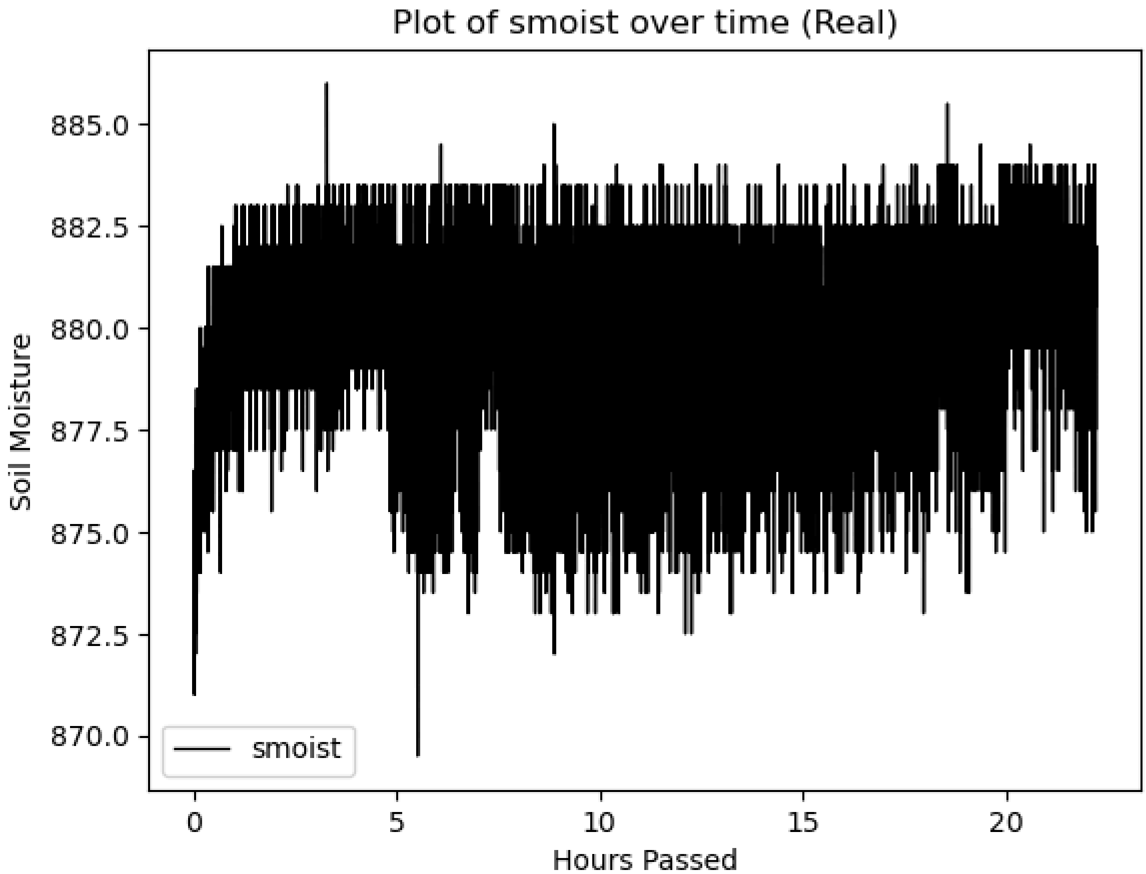


Humidity

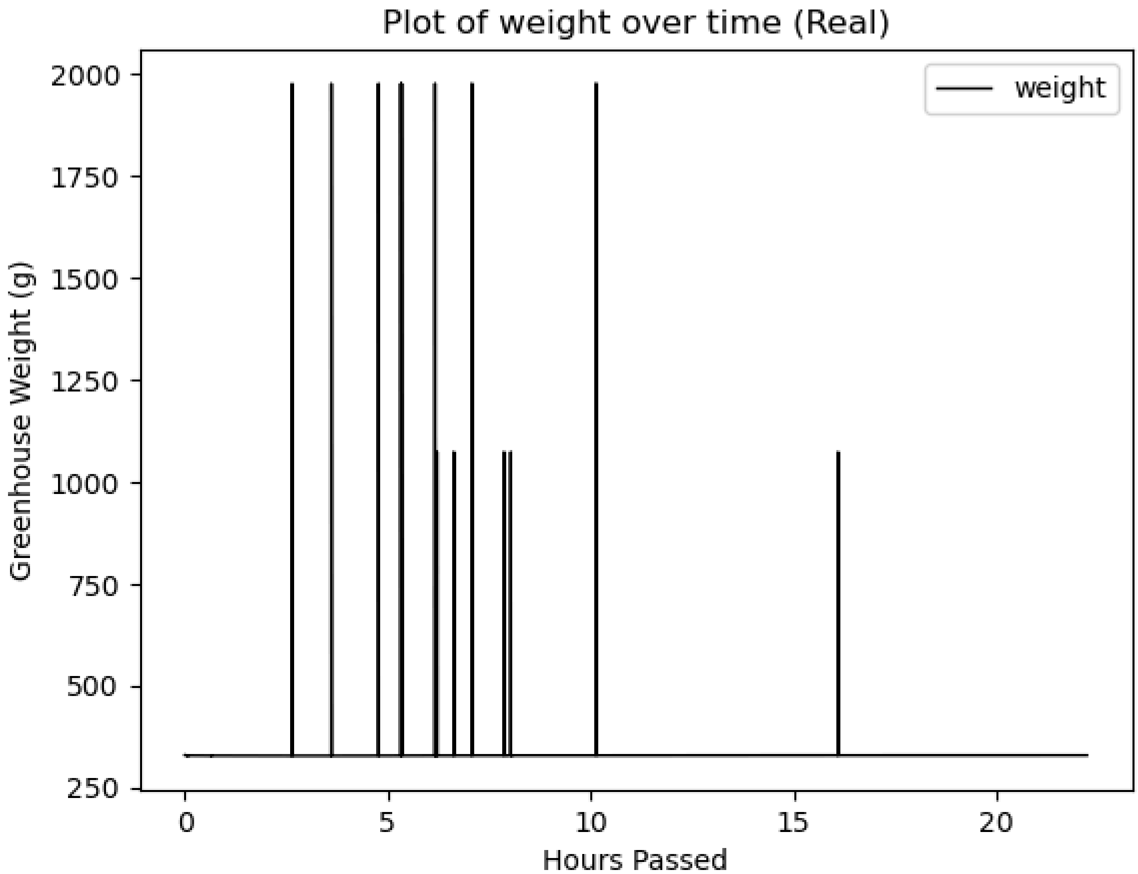


Plot of humid over time (Real)
humid
110
105
100
95
90
85
80
Humidity (%)
0
5
10
15
20
Hours Passed

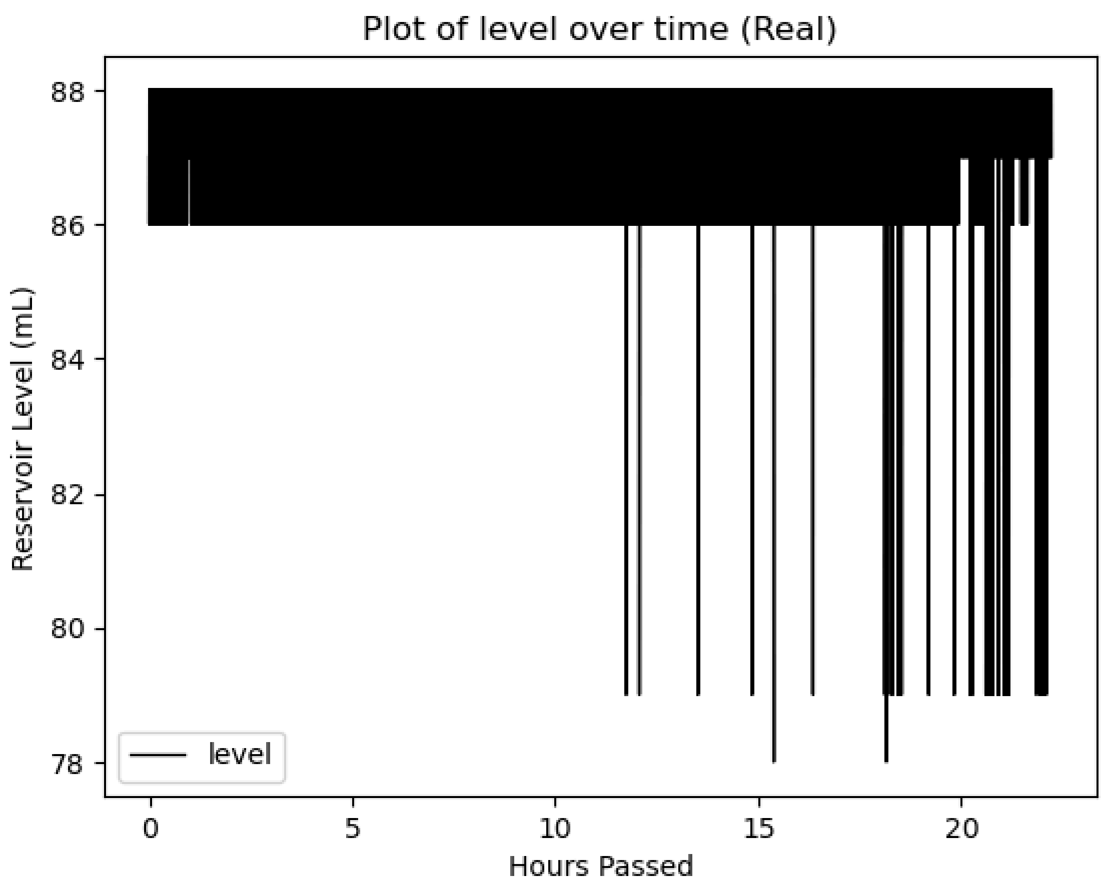
Soil Moisture



Weight


Plot of weight over time (Real)
weight
Greenhouse Weight (g)
2000
1750
1500
1250
1000
750
500
250
0
5
10
15
20
Hours Passed

Water Level

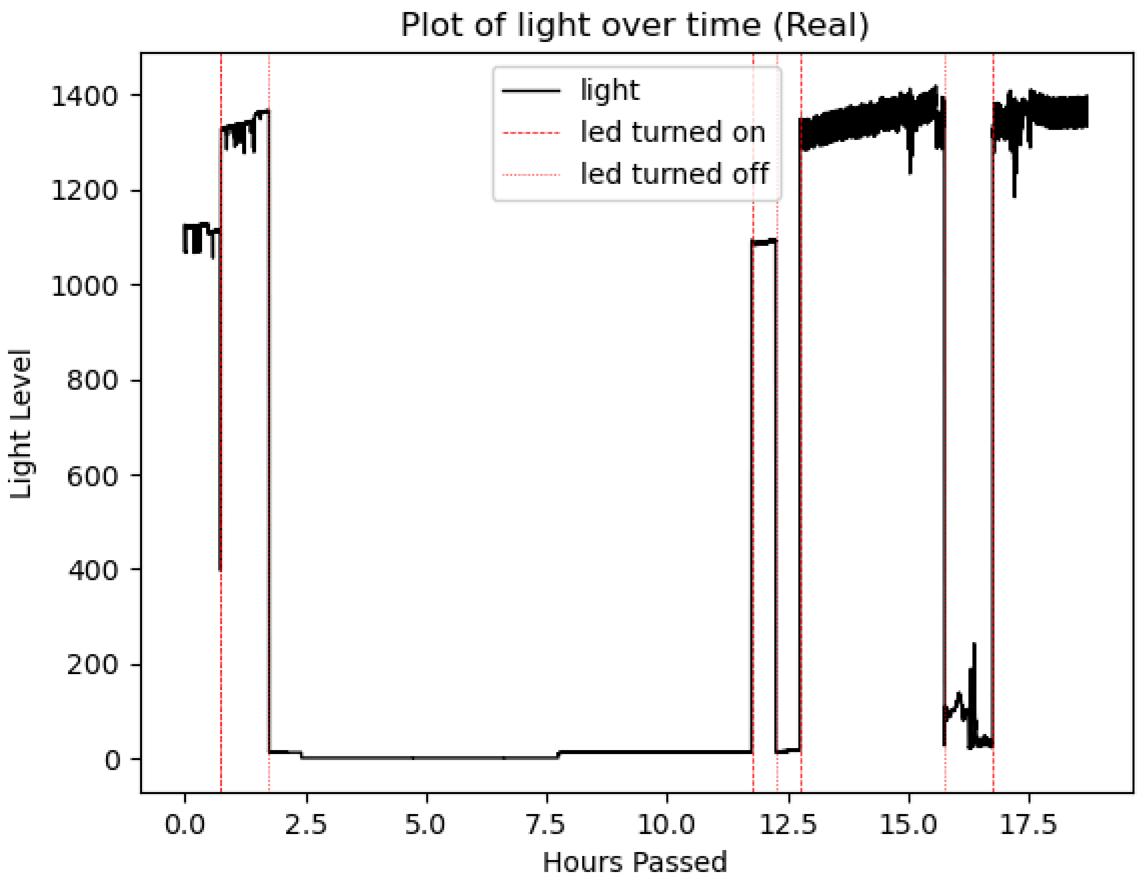

Plot of level over time (Real)
Reservoir Level (mL)
88
86
84
82
80
78
level
0
5
10
15
20
Hours Passed

Session 2

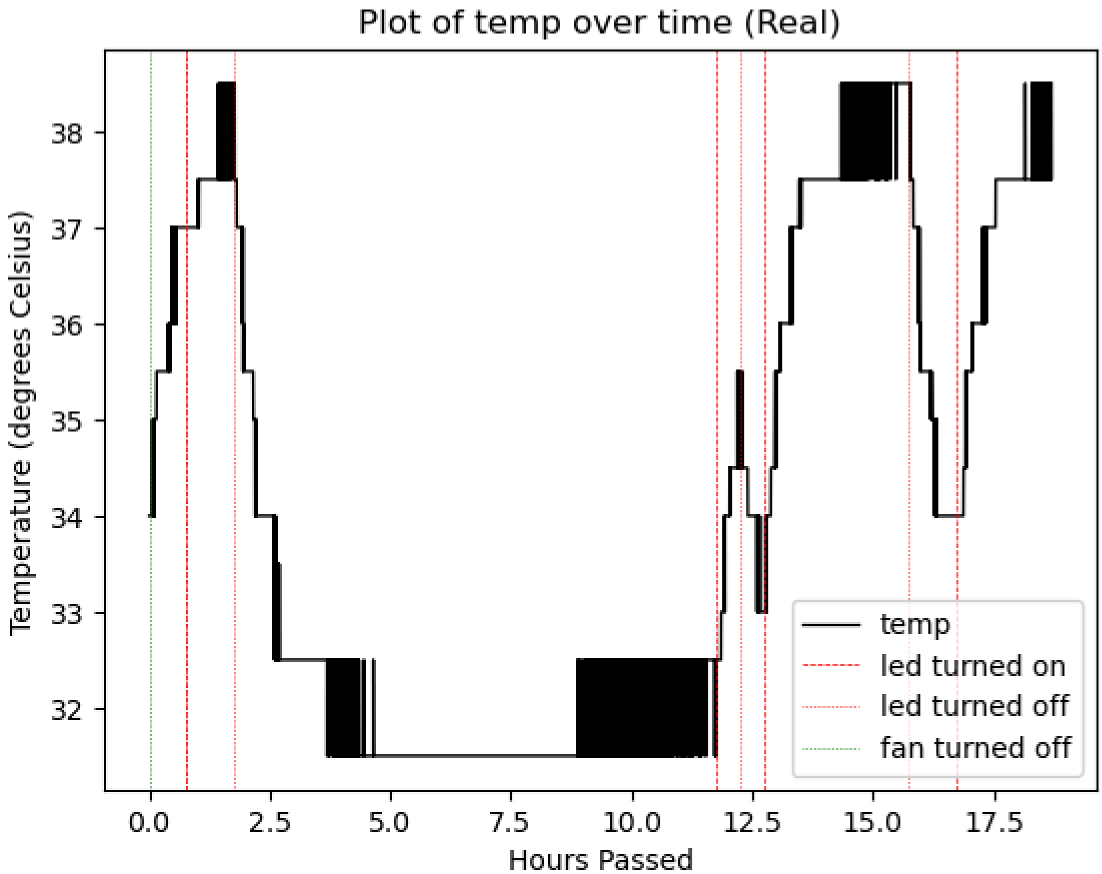
Start Time: ~7:00 PM

Agent: Greenhouse Agent (FSM version, Layered)

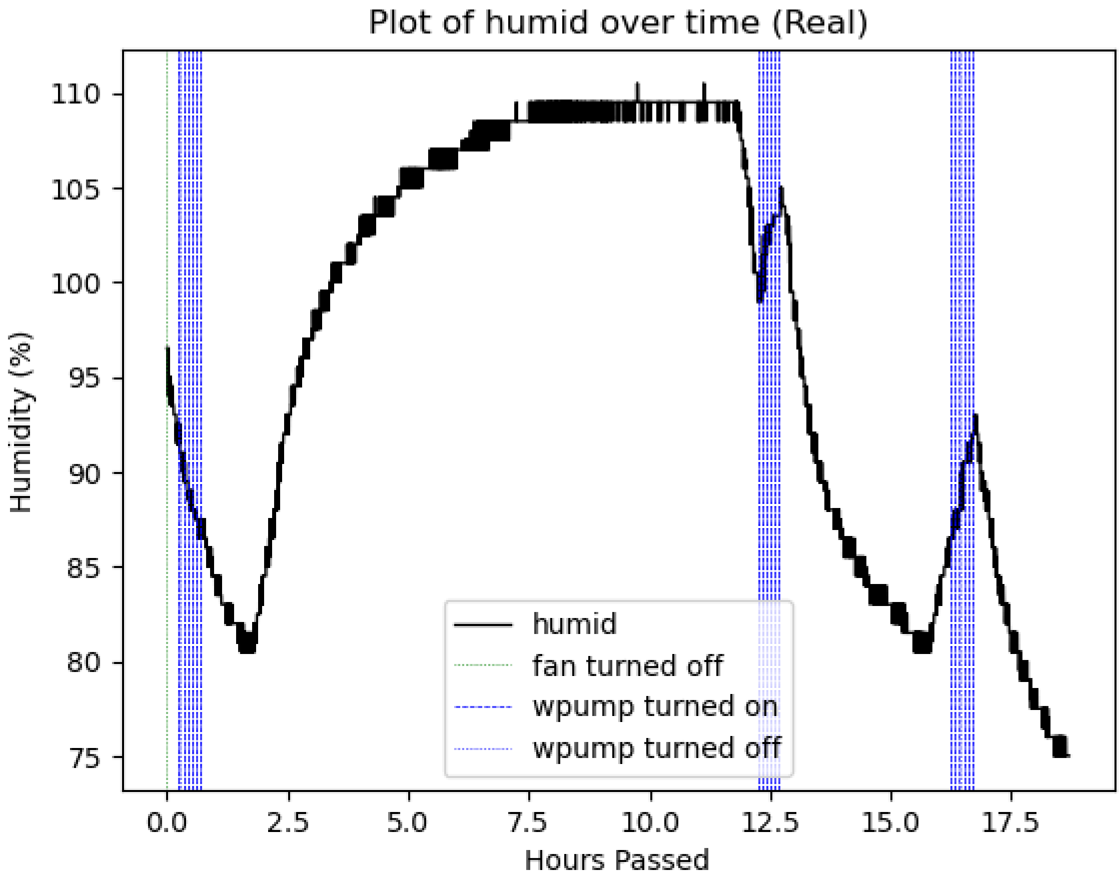
Light Level



Temperature

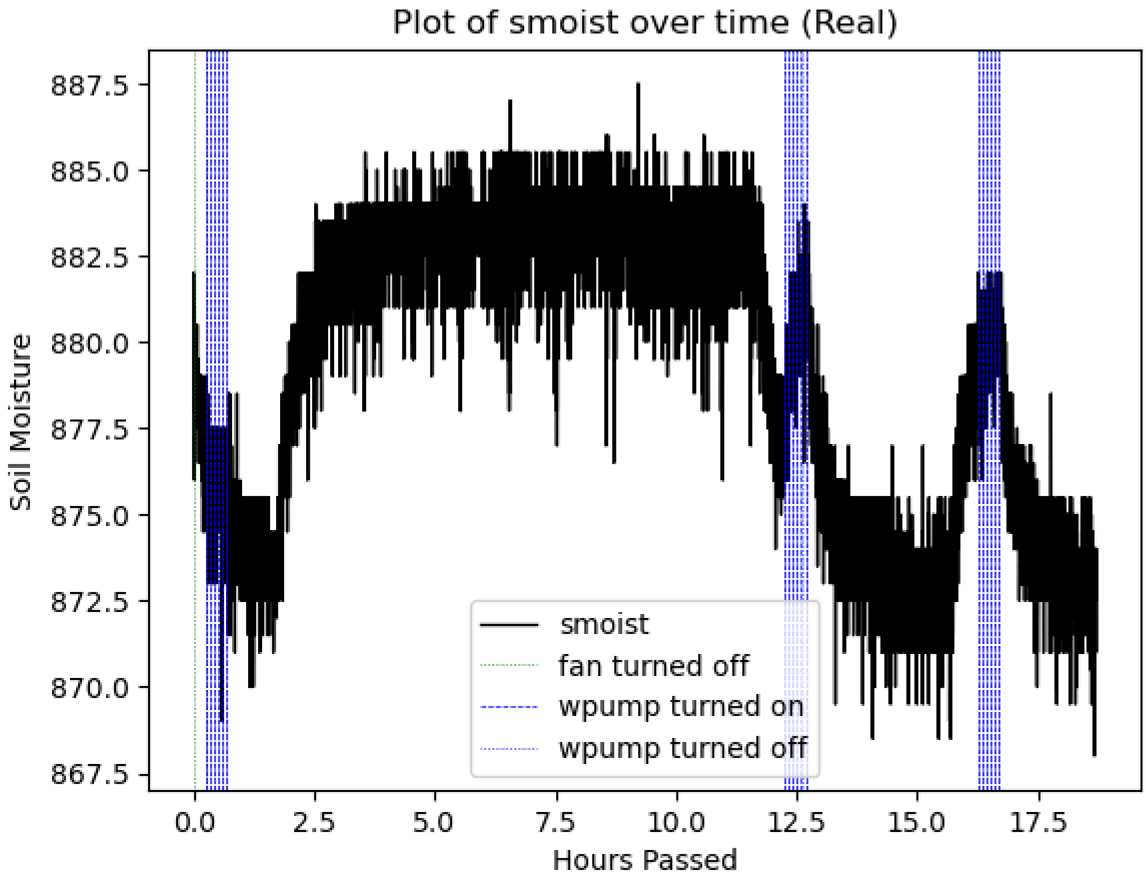


Humidity

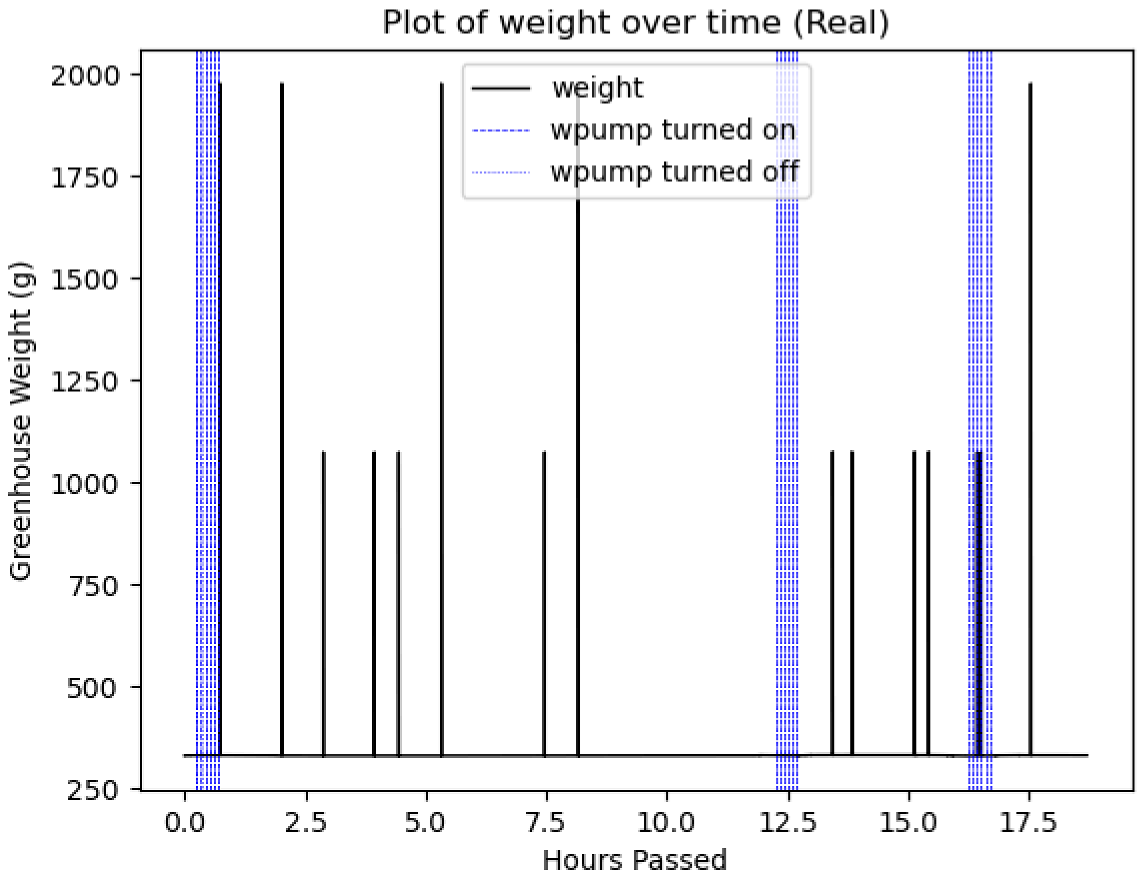


Plot of humid over time (Real)
Humidity (%)
110
105
100
95
90
85
80
75
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Hours Passed
humid
fan turned off
wpump turned on
wpump turned off

Soil Moisture

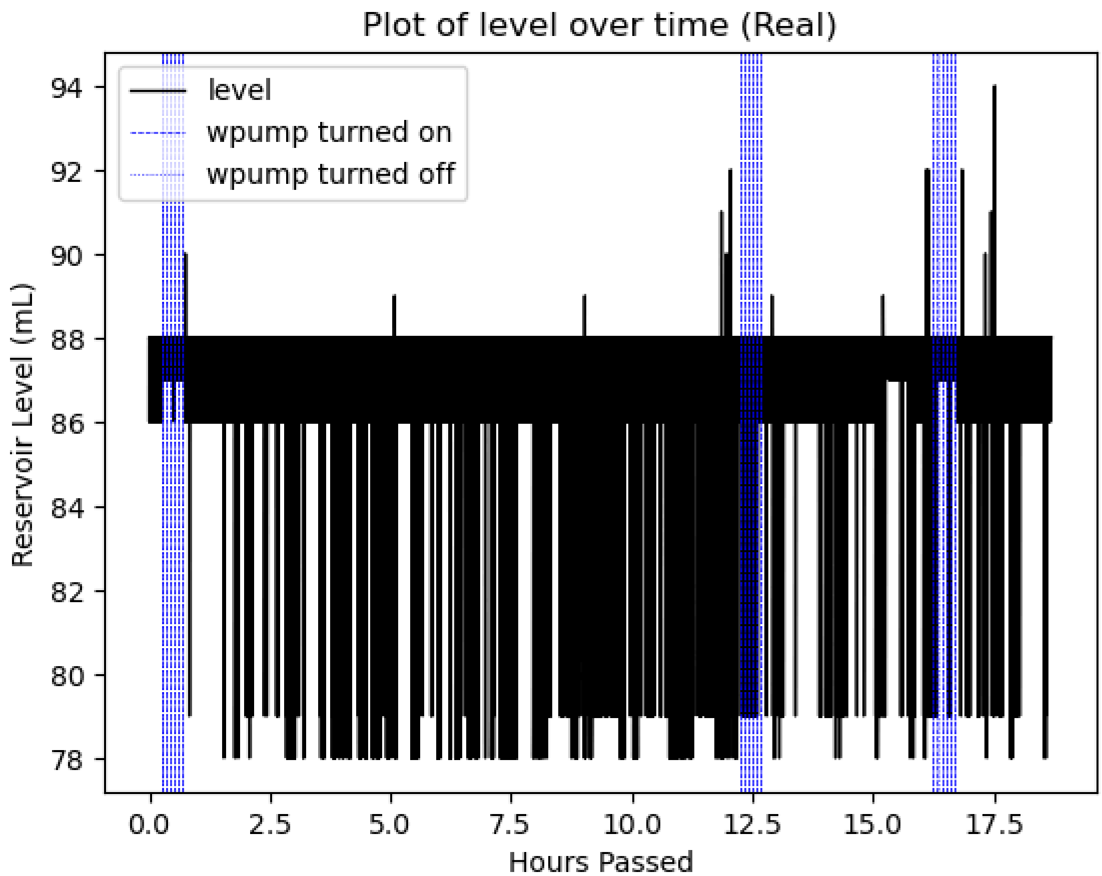


Weight



Plot of weight over time (Real)
weight
wpump turned on
wpump turned off
Greenhouse Weight (g)
2000
1750
1500
1250
1000
750
500
250
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Hours Passed

Water Level


Plot of level over time (Real)
level
wpump turned on
wpump turned off
Reservoir Level (mL)
Hours Passed
94
92
90
88
86
84
82
80
78
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

Response

1) First, understand how the actuators impact the state of the TerraBot. For example, when you plot temperature, you'll notice that the temperature should go up when the lights are on and down when the fans are on. What happens when both are on?

2) Second, compare the simulated and real data. What is similar? What is different? Do you need to update any of the limits or thresholds in your behaviors to reflect your findings?

For light levels, it increases generally throughout the day as the sun rises and sets, and also rises significantly when the LEDs are on. From the real hardware, it appears that the LEDs alone provide a light level of 1000 in the dark.

For temperature, the fan actuator was omitted from the simulation graph to maintain readability as it was triggered quite frequently. In the simulation, there was no clear indicator that the fan influenced the temperature, but for both the simulation and real hardware, temperature increased noticeably (by several degrees Celsius) once LEDs were turned on.

In the simulation, when both the fan and LEDs were on, the temperature did not change.

For the remaining sensors, the real hardware data was occasionally noisy and did not respond to actuators as expected. In the simulation, humidity would naturally increase over time and the fan would temporarily decrease its level. In the real hardware, after the water pump turns on, the humidity takes quite some time to start rising. In the simulation, the weight and soil moisture values increased immediately once the pump was turned on. This was not reflected well in the real hardware where the weight was mostly constant and the soil moisture did not appear to have a consistent response to the water pump. Lastly, in the simulation, the water level decreased immediately when the pump was on, while the sensor in the real hardware gave noisy readings. In conclusion, the water-related sensors and actuators exhibit more noisy behavior and need to be examined more closely.

Looking at the ranges of the sensor values from Session 1 and Session 2 on the real hardware, there appears to be a wide range of the light sensor when the LED is on as opposed to off. When on, the value can reach as high as 1400. When off, the value only reached as far as around 250. Humidity in the hardware often goes beyond 100 unlike the simulation, the typical temperature range is [32, 38] compared to the simulation's range of [22, 29], and the soil moisture is 200-300 units well above that of the simulation. The water level and weight range difference can be explained by differences in the amount of water in the reservoirs and the simulation having plants vs. the real hardware not yet having plants. The ranges may need to be adjusted accordingly.

Part 7

1. Checking all temperature sensors
    a. By logging into the TerraBot, we could view live the value of all the sensors
        i. This would test if the sensor is malfunctioning if there is dissimilarity between the temperature sensors.
    b. Using logs, check if it is a sudden spike or a gradual increase in temperature.
        i. If there was a spike in the temperature (say, current temperature is 2 times higher than temperature from 30 minutes ago), there is more evidence for sensor malfunction.
        ii. Otherwise, there is more evidence towards outside factors that caused temperature increase.
2. Check if the actuators or a behavior is what caused the temperature to rise
    a. Use the logs to view which behaviors are active/inactive during the temperature spike.
        i. Seeing which behaviors are active/inactive leads us to view which actuators are inactive/active as well as the double-checking conditions for behaviors (e.g. behaviors are in the correct state if it's a finite state model, value thresholds, etc.)
    b. This would indicate if it the sensor or the code that's messing up
        i. If there is an actuator command that would cause the rise in temperature (e.g. LED increase) then the problem is with the code rather than the sensor.
        ii. If there isn't that would indicate a sensor malfunction

The second test is better, because it captures more meaningful information behind conditions that cause the temperature increase if it occurs. If the issue was due to a behavior, then the first test would not be able to uncover this, while the second test would. It's also possible that all sensors malfunction which the first test can't capture.

Second Test Pros & Cons vs. First Test

Pros	**Cons**
● More meaningful interpretations of why the temperature increase occurred ● Able to know which behavior(s) contributed to the temperature increase ● Able to create diagnosing behavior depending on which behavior(s) failed. ● Able to capture cases where both	● If the sensor malfunctioned, the first test would be quicker to diagnose, while the second test would take longer to figure out.

temperature sensors.	