Name: Ogundimu Oluwatobi Daniel
Matriculation Number: 18/ENG04/058
Department: Electrical/Electronics

Development of a software that runs the irrigation of the ABUAD farm.

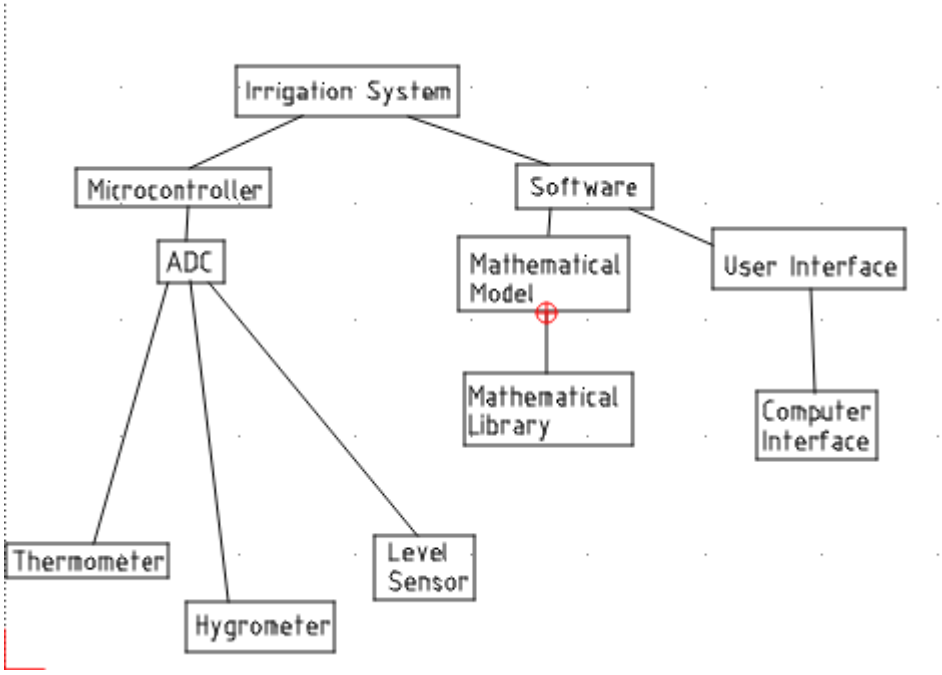
Current problem.
Manual irrigation of the farmlands is an expensive task: Workers are paid a significant amount for a simple repetitive job. The irrigation task is also a tedious one, the amount of water to add and the interval depends on a multitude of factors, including the temperature of the soil, the atmospheric moisture, sunlight and so on. These factors are difficult and inefficient for a worker to use to calculate the amount of water required. Workers are then restrained to using experience guided by their senses – which is a poor way to measure the factors affecting the rate of change of soil moisture. Because of the unreliable irrigation, it is hypothesised that the crop yield is lower than it could be, thus the need for automation.

Plan
The goal is to create a system that would automate the irrigation of the farmland by analysing the temperature of the soil, the moisture content of the soil and the atmospheric humidity. The system would also have an alarm that indicates a low level of water in the tanks for irrigation. Using a mathematical model, the program would calculate the required interval in real time ensuring that the plants get the adequate moisture for maximum yield. The system would require several input parameters including the temperature threshold, the soil moisture threshold and the water level threshold. These parameters, along with a password for the system, is set at the first run of the program: in order to change the parameters or the password, the password must be typed.

Design
For the irrigation system, we would need to build a structure that can measure the soil temperature, the soil moisture, the level of the water in the tank. For these we would require a thermometer, a hygrometer, a level sensor, an ADC. A core irrigation system needs to be purchased for automation by a microcontroller. We would be using a simple microcontroller because the task is relatively simple and repetitive and using a computer is an expensive option. The thermometer, hygrometer and level sensors would be connected to the ADC, which would be connected to the microcontroller, which is connected to the irrigation system. The microcontroller has an interface to a computer, where it is possible to alter the parameters, after inputting the password.

The software is a simple program that would be written for the microcontroller. The program would be modular in design. The mathematical model would be designed independently from the user interface and the sensor interface. However they would be compiled into a single program. Which would be loaded into the microcontroller.

The design would look like the top-down design below:



Implementation

The mathematical model is simply a function of the soil temperature, soil moisture and the parameters determined at the beginning of the program. The function returns the time left before next irrigation, in turn determining the frequency of irrigation. The amount of water added is constant and is one of the initial parameters added to the model. The function would be called multiple times each minute accounting for varying rates of change of the temperature and moisture. When the time left is less than or equal to zero, the irrigation action is begun.

The User interface would be a CLI (Command Line Interface), based on the computer based interface. The User Interface detects whether the microcontroller is connected to a computer. To use the interface, the microcontroller is first connected to a computer, then the microcontroller requests the password from the user. If the password is typed and verified, then the user is able to change the parameters, which include the minimum level of water before alarm rings, the ideal soil moisture and temperature, frequency by which the model function is run (per minute),and amount of water used for irrigation. The user is also able to terminate the program.

The software itself uses the modules to run the irrigation process, once the time left by the model function is less than or equal to zero, the irrigation function is run, irrigating the farmland.

Algorithm

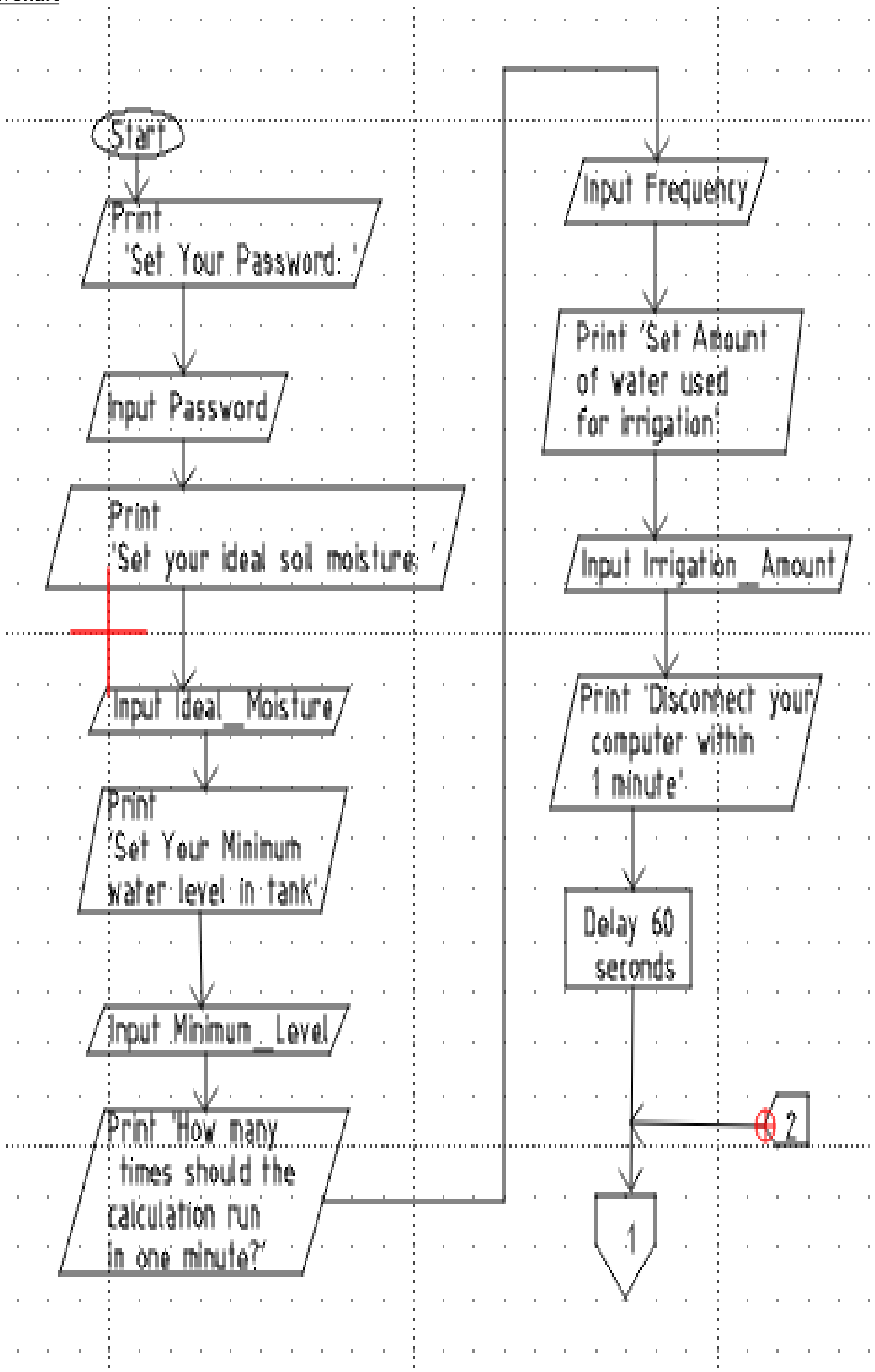
1. Start
2. Print “Set Your Password: ”
3. Input Password
4. Print “Set your ideal soil moisture: ”
5. Input Ideal_Moisture
6. Print “Set Your Minimum water level in tank”
7. Input Minimum_Level
8. Print “How many times should the calculation run in one minute?”
9. Input Frequency
10. Print “Set Amount of water used for irrigation”
11. Input Irrigation_Amount
12. Print “Disconnect your computer within 1 minute”

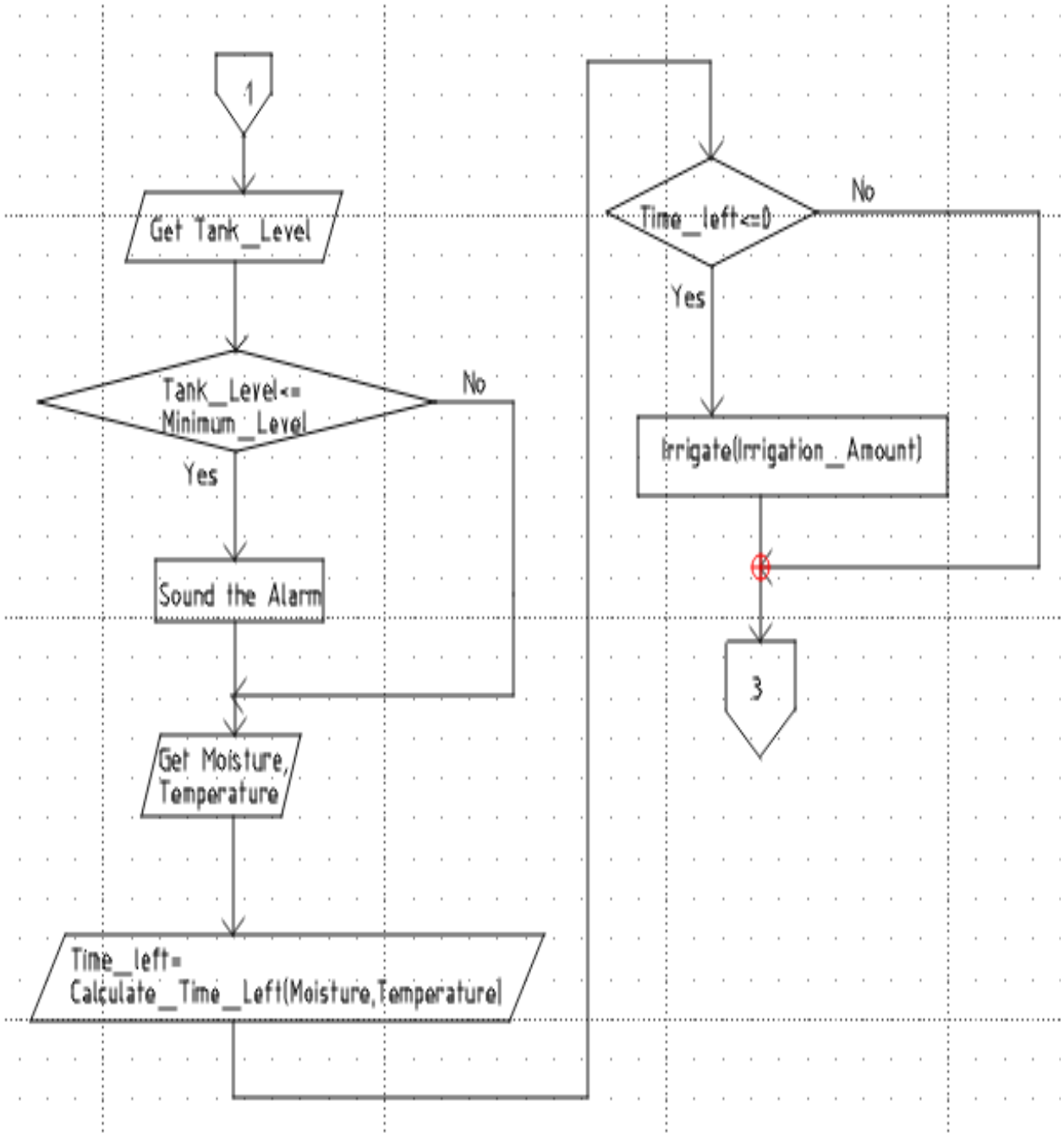
13. Delay 60 seconds
14. While True

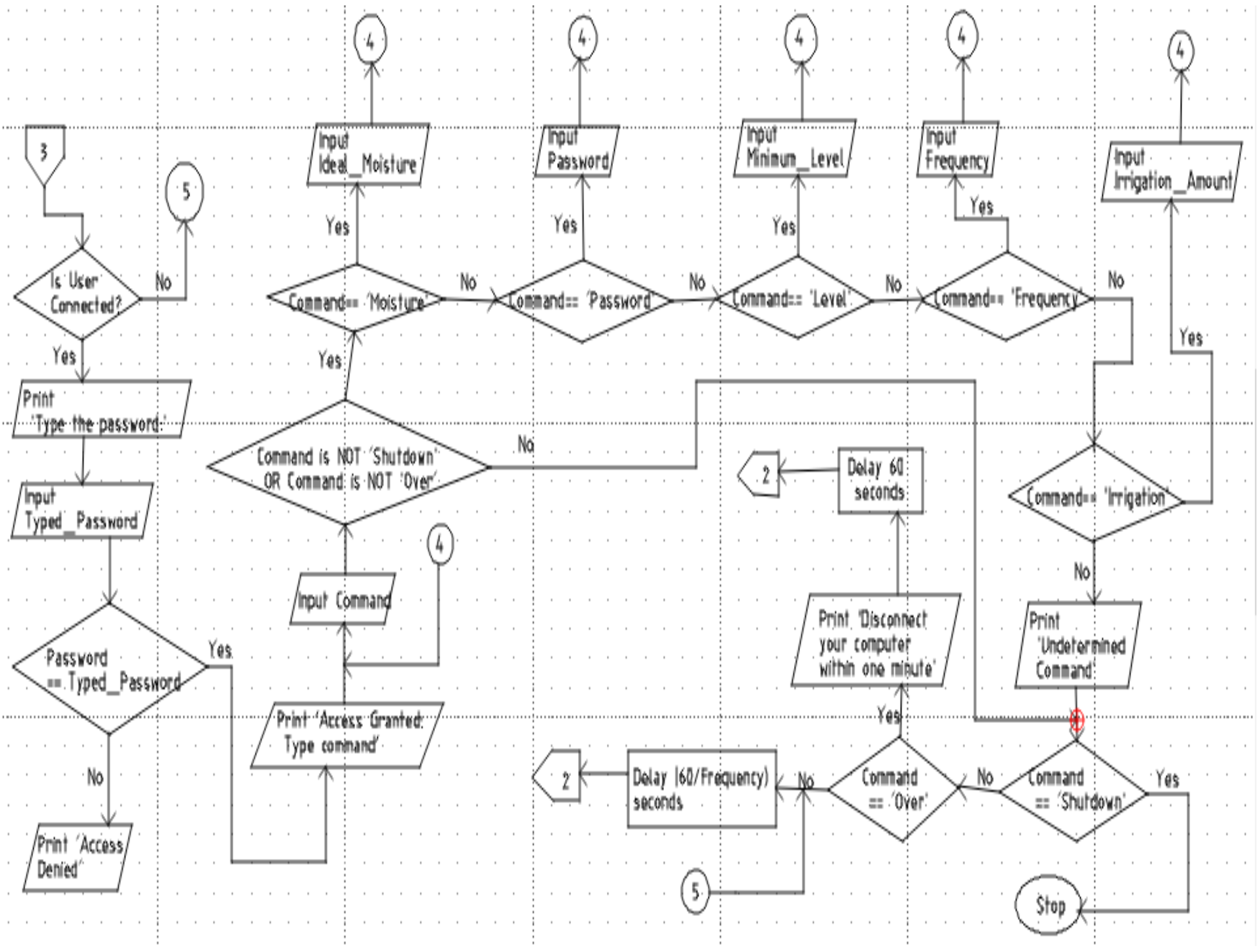
```
        Get Tank_Level          //From the level sensor
        If Tank_Level<=Minimum_Level
            Sound_Alarm()
        EndIf
        Get Moisture            //From hygrometer
        Get Temperature         //From thermometer
        Time_left=Calculate_Time_Left(Moisture,Temperature)
        If Time_left<=0
            Irrigate(Irrigation_Amount)   //Irrigate with Irrigation_amount of Water
        EndIf
        If User_Is_Connected
            Print “Type the password:”
            Input Typed_Password
            If  Password is not Typed_Password
                Print “Access Denied”
            Else
                Print “Access Granted: Type command”
                Input Command
                While Command is NOT “Shutdown” OR Command is NOT “Over”
                    If Command== “Moisture”
                        Input Ideal_Moisture
                    Else If Command== “Password”
                        Input Password
                    Else If Command == “Level”
                        Input Minimum_Level

                    Else If Command == “Frequency”
                        Input Frequency
                    Else If Command == “Irrigation”
                        Input Irrigation_Amount
                    Else
                        Print “Undetermined Command
                    EndIf
                EndWhile
                If Command == “Shutdown”
                    Break
                EndIf
                If Command == “Over”
                    Print “Disconnect your computer within one minute”
                    Delay 60 seconds
                    Continue
                EndIf
            EndIf
        EndIf
        Delay (60/Frequency) seconds
    EndWhile
```

15. Stop

Flowchart


Start
Print 'Set Your Password: '
Input Password
Print 'Set your ideal soil moisture: '
Input Ideal_Moisture
Print 'Set Your Minimum water level in tank'
Input Minimum_Level
Print 'How many times should the calculation run in one minute?'
Input Frequency
Print 'Set Amount of water used for irrigation'
Input Irrigation_Amount
Print 'Disconnect your computer within 1 minute'
Delay 60 seconds
2
1


1
Get Tank_Level
Tank_Level<=
Minimum_Level
No
Yes
Sound the Alarm
Get Moisture,
Temperature
Time_left=
Calculate_Time_Left(Moisture,Temperature)
Time_left<=0
No
Yes
Irrigate(Irrigation_Amount)
3


3
Is User Connected?
No
5
Yes
Print 'Type the password:'
Input Typed_Password
Password == Typed_Password
Yes
No
Print 'Access Denied'
Print 'Access Granted: Type command'
4
Input Command
Command is NOT 'Shutdown' OR Command is NOT 'Over'
Yes
No
Command== 'Moisture'
Yes
Input Ideal_Moisture
4
No
Command== 'Password'
Yes
Input Password
4
No
Command== 'Level'
Yes
Input Minimun_Level
4
No
Command== 'Frequency'
Yes
Input Frequency
4
No
Command== 'Irrigation'
Yes
Input Irrigation_Amount
4
No
Print 'Undetermined Command'
Command == 'Over'
Yes
Print 'Disconnect your computer within one minute'
Delay 60 seconds
2
No
Command == 'Shutdown'
Yes
Stop
Delay (60/Frequency) seconds
2
No
5

Testing

The most significant tests would be on the mathematical model. It is important that the formula is accurate and reliable enough for the given task. Before the final deployment, we would also run tests in the practical environment with various values for the parameters, and check that the soil moisture is kept to the ideal moisture with minimal variation.

Maintenance & Updates

There are possible future changes we can make, such as adding a functionality to automatically switch on the pump when the minimal level is reached. Also the model could be improved once observed for one season. The user interface could also be improved allowing for deeper control of the irrigation system.