

Soil Productivity and Fertility Using Neuro-Fuzzy Logic Inference

[¹] Dr.K.Renganathan, [²]Vishnuvardhan J, [³]Badhrivishal RPV, [⁴]Venkatesh.K.D, [⁵]Dogiparthi Manoj
[⁶] Srinivas.R.S

[¹] Head of the department, Department of Electronics and Instrumentation Engg, Sri Sairam Engineering

College, Chennai., [²] CEO and Founder, Trend Instrumentation Pvt Ltd, Chennai

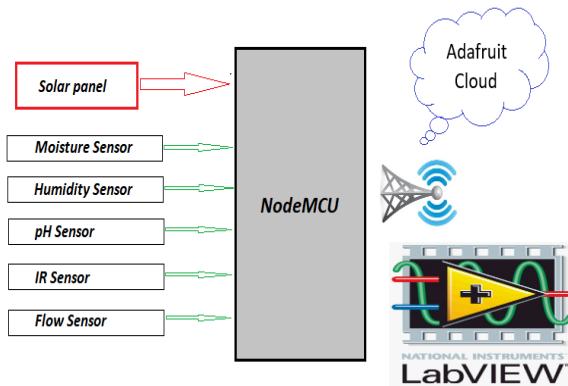
[³][⁴][⁵][⁶] UG Scholars, Department of Electronics and Instrumentation, Sri Sairam Engineering College, Chennai.

Abstract – Soil nutrients testing methodologies are still a concern as most of them are time consuming and require laborious sampling which is expensive. Soil sensing stations installed in a precision agriculture farm can generate real time soil data online to keep on tracking about soil status such as macronutrients (NPK), temperature, moisture, humidity, flow and pH. These types of soil data can be analyzed with the respective sensor and feed through the microcontroller. The system utilizes an NODEMCU for data acquisition therefore the output from the transducer is converted into a digital display reading. Wireless sensor network is one of the key technologies for monitoring and regulating crop parameters towards efficient farming as it has been shown that the sensors provides indigenous platforms in various fields. The inference rules are framed using expert knowledge in the form of IF...THEN structures. Fuzzy logic tool in LabVIEW is used for building a prediction model. Neuro-Fuzzy model is used for the prediction of soil productivity.

I. INTRODUCTION

Soil is the main source of nutrients for crops. Soil also provides support for plant growth in various ways. Knowledge about soil health and its maintenance is critical to sustaining crop productivity. The health of soils can be assessed by the quality and stand of the crops grown on them. However, this is a general assessment made by the farmers. A scientific assessment is possible through detailed physical, chemical and biological analysis of the soils. Essential plant nutrients such as N, P, K, Ca, Mg and S are called macronutrients, while Fe, Zn, Cu, Mo, Mn, B and Cl are called micronutrients. It is necessary to assess the capacity of a soil to supply nutrients in order to supply the remaining amounts of needed plant nutrients (total crop requirement - soil supply). Thus, soil testing laboratories are considered nerve centres for nutrient management and crop production systems.

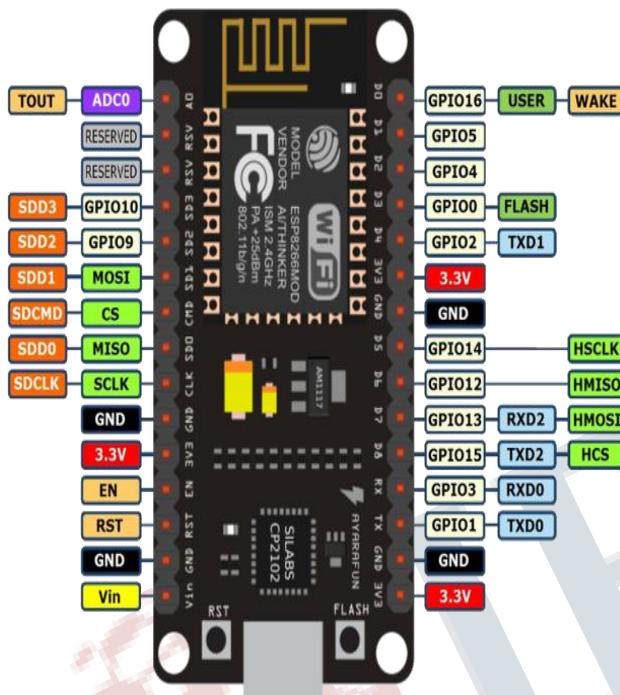
BLOCK DIAGRAM



II. NODEMCU

Node MCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the firmware rather than the development kits. The firmware uses the Lua scripting language. It is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects; such as lua-cjson and SPIFFS. Node MCU Dev Kit has Arduino like Analog (i.e. A0) and Digital (D0-D8) pins on its

board. It supports serial communication protocols i.e. UART, SPI, I2C etc. Using such serial protocols we can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards etc.



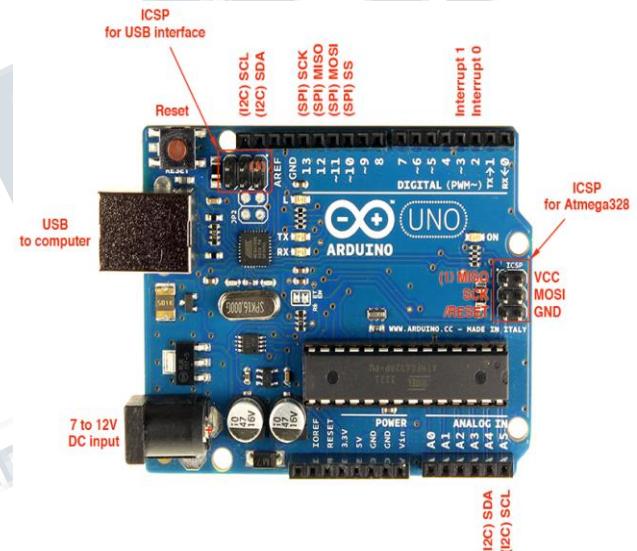
Node MCU has general purpose input output pins on its board as shown in above pin-out diagram. We can make it digital high/low and control things like LED or switch on it. Also, we can generate PWM signal on these GPIO pins. Node MCU has one ADC channel/pin on its board (A0). Node MCU based ESP8266 has Hardware SPI (HSPI) with four pins available for SPI communication. It also has SPI pins for Quad-SPI communication. With this SPI interface, we can connect any SPI enabled device with Node MCU and make communication possible with it.

Node MCU has I2C functionality support on ESP8266 GPIO pins. Due to internal functionality on ESP-12E we cannot use all its GPIOs for I2C functionality. So, do tests before using any GPIO for I2C applications Node MCU based ESP8266 has two UART interfaces, UART0 and UART1. Since UART0 (RXD0 & TXD0) is used to upload firmware/codes to board, we can't use them in applications while uploading firmware/codes.

III.ARDUINO

The Arduino UNO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc.[2][3] The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 Digital pins, 6 Analog pins, and is programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable.[4] It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts.

Relay Circuit :



Each of the 14 digital pins on the Arduino Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. pins 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip. 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write() function. 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library. 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

**International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)**
Vol 6, Issue 7, July 2019**IV. SOFTWARE PART**

Laboratory Virtual Instrument Engineering Workbench (LabVIEW) is a system-design platform and development environment for a visual programming language from National Instruments.

The graphical language is named "G"; not to be confused with G-code. Originally released for the Apple Macintosh in 1986, LabVIEW is commonly used for data acquisition, instrument control, and industrial automation on a variety of operating systems (OSs), including Microsoft Windows, various versions of Unix, Linux, and macOS. If there is enough data available to a subVI or function, that subVI or function will execute. Execution flow is determined by the structure of a graphical block diagram (the LabVIEW-source code) on which the programmer connects different function-nodes by drawing wires. These wires propagate variables and any node can execute as soon as all its input data become available. Since this might be the case for multiple nodes simultaneously, LabVIEW can execute inherently in parallel. Multi-processing and multi-threading hardware is exploited automatically by the built-in scheduler, which multiplexes multiple OS threads over the nodes ready for execution.

V. GRAPHICAL PROGRAMMING

LabVIEW integrates the creation of user interfaces (termed front panels) into the development cycle. LabVIEW programs-subroutines are termed virtual instruments (VIs). Each VI has three components: a block diagram, a front panel, and a connector panel. The last is used to represent the VI in the block diagrams of other, calling VIs. The front panel is built using controls and indicators. Controls are inputs: they allow a user to supply information to the VI. Indicators are outputs: they indicate, or display, the results based on the inputs given to the VI. The back panel, which is a block diagram, contains the graphical source code. All of the objects placed on the front panel will appear on the back panel as terminals. The back panel also contains structures and functions which perform operations on controls and supply data to indicators. The structures and functions are found on the Functions palette and can be placed on the back panel. Collectively controls, indicators, structures, and functions are referred to as nodes. Nodes are connected to one another using wires, e.g., two controls and an indicator can be wired to the addition function so that the indicator displays the sum of the two controls. Thus a virtual instrument can be run as either a program,

with the front panel serving as a user interface, or, when dropped as a node onto the block diagram, the front panel defines the inputs and outputs for the node through the connector panel. This implies each VI can be easily tested before being embedded as a subroutine into a larger program. The graphical approach also allows nonprogrammers to build programs by dragging and dropping virtual representations of lab equipment with which they are already familiar. The LabVIEW programming environment, with the included examples and documentation, makes it simple to create small applications. This is a benefit on one side, but there is also a certain danger of underestimating the expertise needed for high-quality G programming. For complex algorithms or large-scale code, it is important that a programmer possess an extensive knowledge of the special LabVIEW syntax and the topology of its memory management. The most advanced LabVIEW development systems offer the ability to build stand-alone applications. Furthermore, it is possible to create distributed applications, which communicate by a client-server model, and are thus easier to implement due to the inherently parallel nature of G.

VI.APPLICATIONS

The easiest live tracking of soil nutrient content in the given test sample and also with the amount of humidity, moisture, temperature , pH content present in the soil.

RESULT AND DISCUSSIONS

The labview programming is used with the neuro-fuzzy logic(By using IF.....THEN logic) and the soil nutrient (NPK) content for the given sample of soil is shown in the front panel diagram.

REFERENCES

- [1] Miss Maya V. Mawale, Dr.VinayChavan" Fuzzy Inference System for productivity and fertility of soil", IJEDR ,Volume 2, Issue 3 ,2014
- [2] Kailan Shang, ZakirHossen, "Applying Fuzzy Logic to Risk Assessment and Decision-Making", November 2013
- [3] L. C. Gavade, "Detection of N, P, K using Fiber Optic Sensor and PIC Controller," International Journal of Engineering Science, vol. 13787, 2017

**International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)**
Vol 6, Issue 7, July 2019

- [4] MarianahMasrie,MohamadSyamim,AizuddinRosman, Rosidah Sam ,ZuriatiJanin ,”Detection of Nitrogen, Phosphorus, and Potassium (NPK) nutrients of soil using Optical Transducer”, Proc. of the 4th IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA) 28-30 November 2017
- [5] R.Sindhuja and B.Krithiga,“Soil Nutrient Identification Using Arduino“,Asian Journal of Applied Science and Technology (AJAST) Volume 1, Issue 4, Pages 40-42, May 2017.

