***CENTRALIZED REMOTE AGRICULTURE***

***PUMP CONTROL SYSTEM FOR***

***FARMERS***

**ABSTRACT:**

As we all know that India is an agriculture oriented country, now a day’s our farmers are facing very difficulties in irrigating the land due to the lack of power supply or unable to get the required supply on-time to run the irrigation motor, or to prevent the crop from over load and wastage of water and dry run. Mostly the farmers lost their lives when turning on the motors without any precautions due to electrical shocks , they will face this type of problems in villages, very hardly they get power in night time, So to turn on the motor someone has to go to the field and turn on, and specifically in night times they don't have availability of lights.

Due to this causes by Keeping this in mind we came up with the concept called “CENTRALISED REMOTE AGRICULTURE PUMP CONTROL SYSTEM FOR FARMERS”. This device can be used to "turn ON and turn OFF" the Motor pumps Remotely by using app .It can handle many number of motor pumps , by this special feature we can "turn ON and "turn OFF" number of motors at a time . Farmers have to travel great distances to manually operate pumps, through difficult terrain, often encountering snakes and other hazards ,by accessing through app present in this device they can operate motors from their own place.

**INRODUCTION:**

"CENTRALIZED REMOTE AGRICULTURE PUMP CONTROL SYSTEM FOR FARMERS" .In the view of farmers difficulties in agriculture and losses of their lives This project is designed, to make their daily life easily,

In this project, several hardware and software components, the device is operated with the help of mobile app by pressing the buttons in screen the commands are transferred from mobile and received by the motor, the motor turns on and the water is supplied to the agriculture fields Node MCU Development board is featured with WIFI capability, analog pin, digital pins and serial communication protocols, It gives power supply to the relay module***.*** When NODEMCU is connected with relay module it receives input from user when motor is ON the LED will glow in relay module, it indicates that the motor is turned ON. when motor is OFF the LED will not glow in relay module.

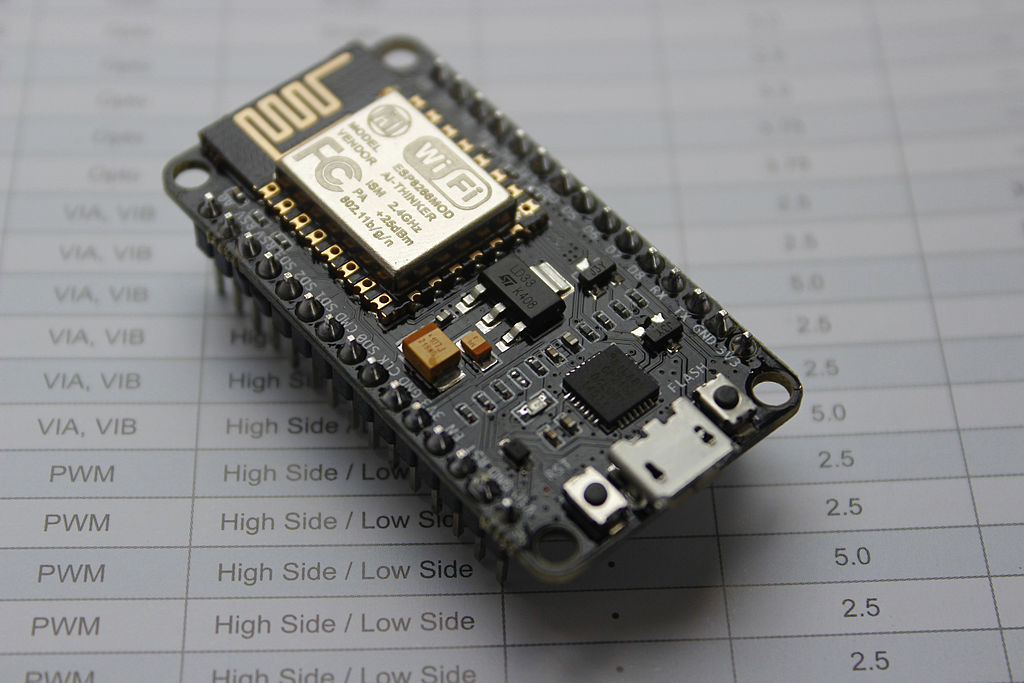
IBM cloud is used to send data and to access stored data when user gives inputs .IBM cloud computing is a set of cloud computing services for business offered by the information technology company IBM ,In IBM cloud with the feature of NODE RED we are providing device type ,device id and various information, to the cloud it is the main intermediate application in between "user" and "output" ,with help of MIT app inventor on the project basis the app is designed ,app is designing in the way that farmers can easily access through it and can understand ,if the farmers who still can't know how to hit which button to turn on or turn off the motor they can easily access through the "MIC" application by accessing through it by using their voice they can turn on or turn off the motor this feature makes this project versatile ."ARDUINO IDE" it is the application software ,when the user gives inputs the inputs are received by it, and checks whether the given input is exist or error, and gives the output to run the motors.

***HARDWARE COMPONENTS***

**NODE MCU (ESP8266)** :

**Node MCU** is an open source IOT platform. It includes firmware which runs on the ESP8266 Wi-Fi SOC from Express if Systems, and hardware which is based on the ESP-12 module. The term "Node MCU" by default refers to the firmware rather than the development kits.

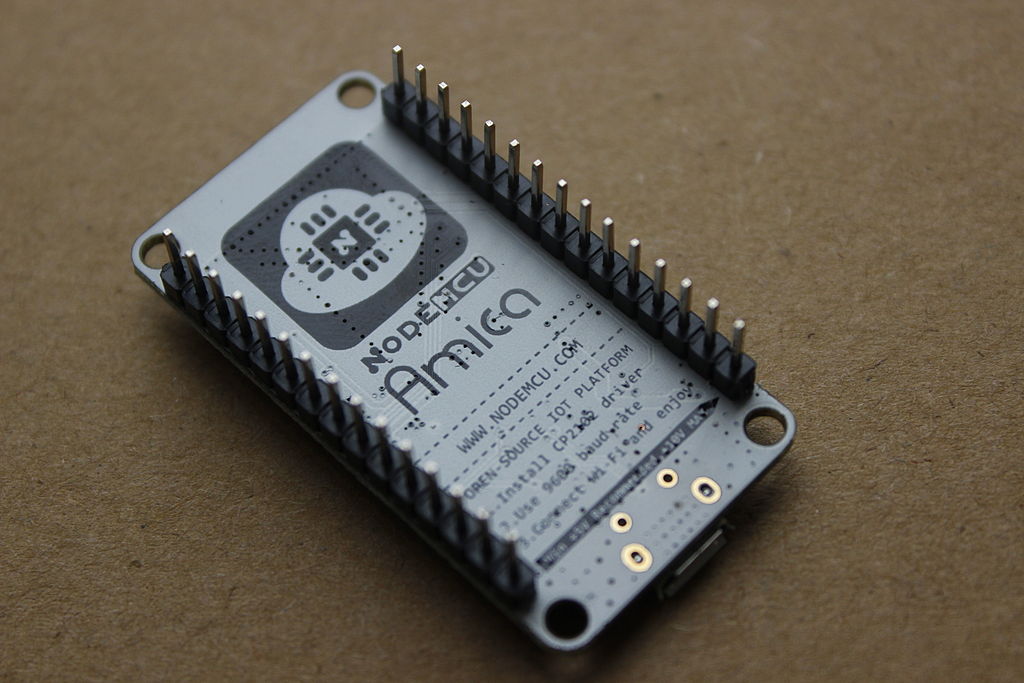
The firmware uses the LUA scripting language. It is based on the ELUA project, and built on the Express if Non-OS SDK for ESP8266. It uses many open source projects.



Node MCU was created shortly after the ESP8266 came out. On December 30, 2013, Espressif Systems began production of the ESP8266.The ESP8266 is a Wi-Fi SOC integrated with a tensilica xtensa LX106 core, widely used in IOT application. Node MCU started on 13 Oct 2014, when Hong committed the first file of node mcu-firmware to GitHub.Two months later, the project expanded to include an open-hardware platform when developer Huang R committed the gerber file of an ESP8266 board, named devkit v0.9. Later that month, Tuan PM ported MQTT client library from Contiki to the ESP8266 SOC platform, committed to Node MCU project, then Node MCU was able to support the MQTT IOT protocol, using Lua to access the MQTT broker. Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glib to Node MCU project, enabling Node MCU to easily drive LCD, Screen, OLED, even VGA displays.

ESP8266 Arduino Core,As Arduino.cc began developing new MCU boards based on non-AVR processors like the ARM/SAM MCU and used in the Arduino Due, they needed to modify the Arduino IDE so that it would be relatively easy to change the IDE to support alternate tool chains to allow Arduino C/C++ to be compiled for these new processors. They did this with the introduction of the Board Manager and the SAM Core. A "core" is the collection of software components required by the Board Manager and the Arduino IDE to compile an Arduino C/C++ source file for the target MCU's machine language. Some ESP8266 enthusiasts developed an Arduino core for the ESP8266 WiFi SoC, popularly called the "ESP8266 Core for the Arduino IDE".

This has become a leading software development platform for the various ESP8266-based modules and development boards, including Node MCUs

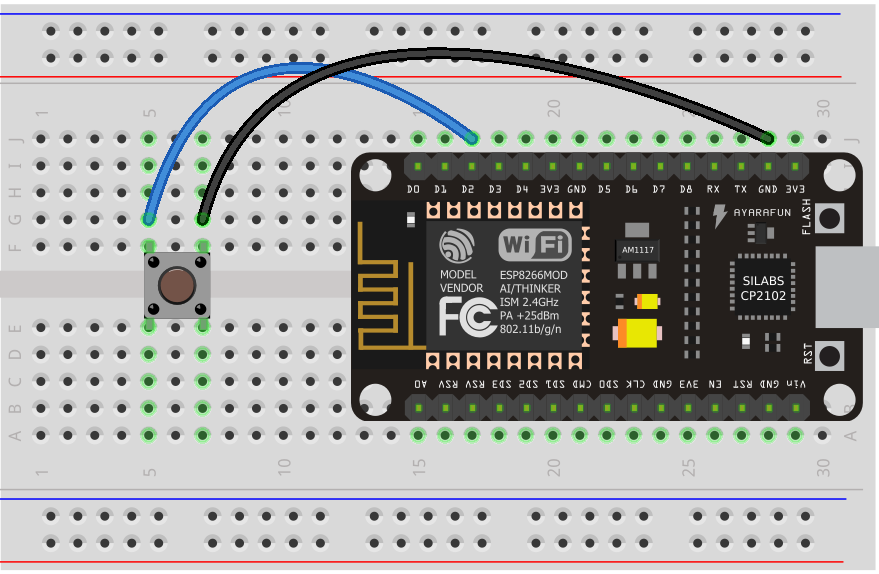


**Push buttons** :

The **Push Button** is a electronic component that connects two points in a circuit when you press it. We use these little buttons on everything!

These Miniature Single Pole Single Throw **(SPST)** switches are bread board friendly. Perfect as a tactile reset switch. These buttons are rated up to 50mA.

**Typical Push Button Switches Applications ,**Calculators, push-button telephones, various other mechanical and electronic devices, home and commercial purposes.



When the pushbutton is open (un pressed) ,there is noconnection between the two legs of the pushbutton,

so the pin is connected to 3.3v (through the pull-up resistor) and we read a HIGH.

When the pushbutton isclosed(pressed), there is **a** connection between its two legs, connecting the pin to ground,

so that we read a LOW. (The pin is still connected to 5 volts, but the resistor in-between them means that the pin is "closer" to ground.

The **first pin** goes from one leg of the pushbutton through a pull-up resistor(here 10K Ohms) to the5v supply.

The **second pin** goes from the corresponding leg of the pushbutton to Ground (GND) pin.

The **third pin** connects to a Digital I/O pin which reads the button's state

**RELAY MODULE :**

A relay is an electrically operated switch that can be turned on or off, letting the current go through or not, and can be controlled with low voltages, like the 5V provided by the Arduino pins.

This relay module has two channels (those blue cubes). There are other models with one, four and eight channels. This module should be powered with 5V, which is appropriate to use with an Arduino. There are other relay modules that are powered using 3.3V, which is ideal for ESP32, ESP8266, and other microcontrollers.



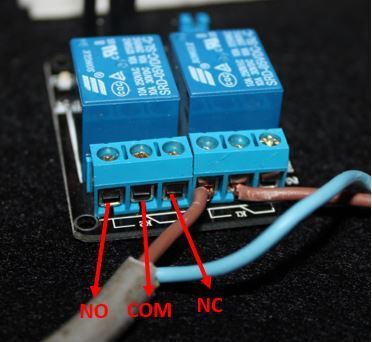
The high-voltage side has two connectors, each with three sockets: common (COM), normally closed (NC), and normally open (NO).

**COM**: common pin

**NC (Normally Closed):** the normally closed configuration is used when you want the relay to be closed by default, meaning the current is flowing unless you send a signal from the Arduino to the relay module to open the circuit and stop the current.

**NO (Normally Open):** the normally open configuration works the other way around: the relay is always open, so the circuit is broken unless you send a signal from the Arduino to close the circuit

Pin wiring ,The low-voltage side has a set of four pins and a set of three pins.

The set at the right consists of **VCC** and **GND** to power up the module, and input 1 (**IN1**) and input 2 (**IN2**) to control the bottom and top relays, respectively.

The second set of pins consists of **GND**, **VCC**, and **JD-VCC** pins. The JD-VCC pin powers the electromagnet of the relay

**GND**: goes to ground

**IN1**: controls the first relay it will be connected to an Arduino digital pin.

**IN2**: controls the second relay ,it should be connected to an Arduino digital pin if you are using this second relay. Otherwise, you don’t need to connect it.

**VCC**: goes to 5V.

# ***software components***

**IBM WATSON IOT PLATFORM :**

This service is the hub for IBM Watson IOT and lets you communicate with and consume data from connected devices and gateways. Use the built-in web console dashboards to monitor your IOT data and analyze it in real time. Then, enhance and customize your IBM Watson IOT Platform experience by building and connecting your own apps by using messaging and REST APIs.

**Connect**

Quickly and securely register and connect your devices and gateways. You can find simple step-by-step instructions for connecting popular devices, sensors, and gateways in our recipes site.

**Analyze in real time**

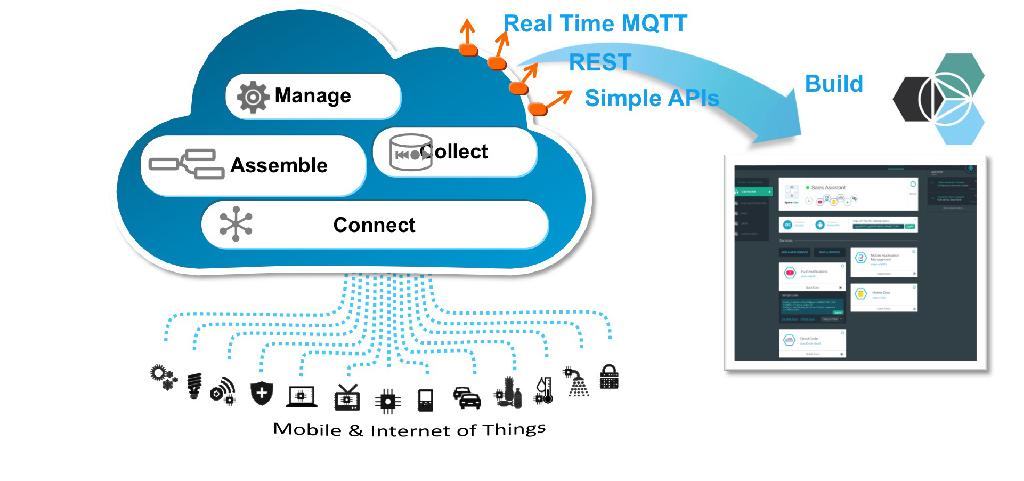
Monitor your real-time device data through rules, analytics, and dashboards. Define rules to monitor conditions and trigger automatic actions that include alerts, email, **IFTTT**, **Node-RED** flows, and external services to react quickly to critical changes.

**Information Management**

Control what happens to the data that is received from your connected devices. Manage data storage, configure data transformation actions, and integrate with other data services and device platforms.

**Risk and Security management**

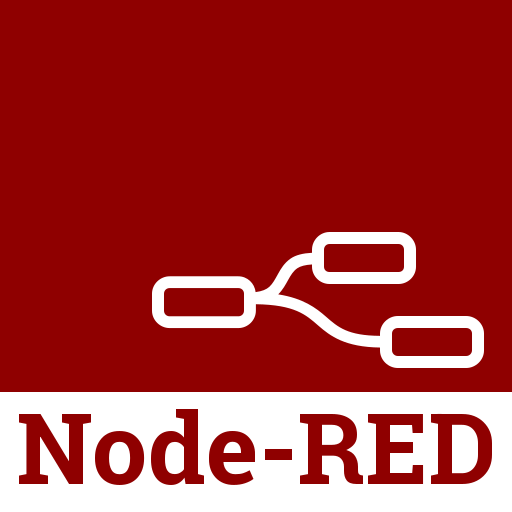
Our secure-by-design control capabilities protect the integrity of your IOT solution through secure connectivity and access control for users and applications. Extend the base security with threat intelligence for IOT to visualize critical risks and automate operational responses with policy-driven mitigation actions.



**Node-RED Starter**

**Node-RED** is a flow based development tool for visual programming developed originally IBM for wiring together hardware devices,APIs and online services as part of the internet of things.

Node-RED provides a web browser-based flow editor, which can be used to create java script functions. Elements of applications can be saved or shared for re-use. The runtime is built on node.js. The flows created in Node-RED are stored using JSON. Since version 0.14 MQTT nodes can make properly configured TLS connections



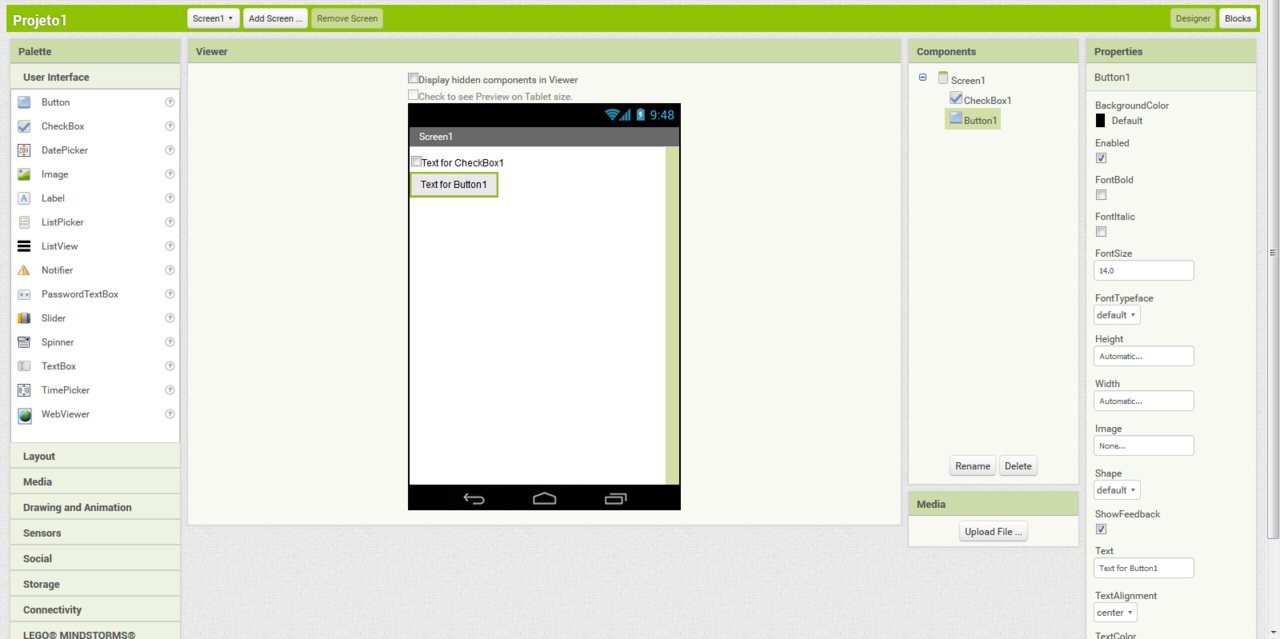
You can deploy the flows to the Node.js runtime environment with a single click. This starter application provides a version of Node-RED that is customized to run in IBM Cloud.

**MIT APP INVENTOR:**

App Inventor is a web application integrated development app inverter originally provided by google and now maintained by the (MIT). It allows newcomers to computer programming to create application software(apps) for two operating systems(OS):android, and ios, which, as of 8 July 2019[[update]](https://en.wikipedia.org/w/index.php?title=App_Inventor_for_Android&action=edit), is in final beta testing, scheduled to be released publicly in summer 2019. It free and open ource software released under dual licensing: a creative commons attribution ShareAlike 3.0 Unported license, and an apanche license 2.0 for the source code.

It uses a graphical user interface(GUI) very similar to the programming languages scratch and the star logoTNG user, which allows users to drag and drop visual objects to create an application that can run on mobile devices. In creating App Inventor, Google drew upon significant prior research in educational computing, and work done within Google on online development environments.

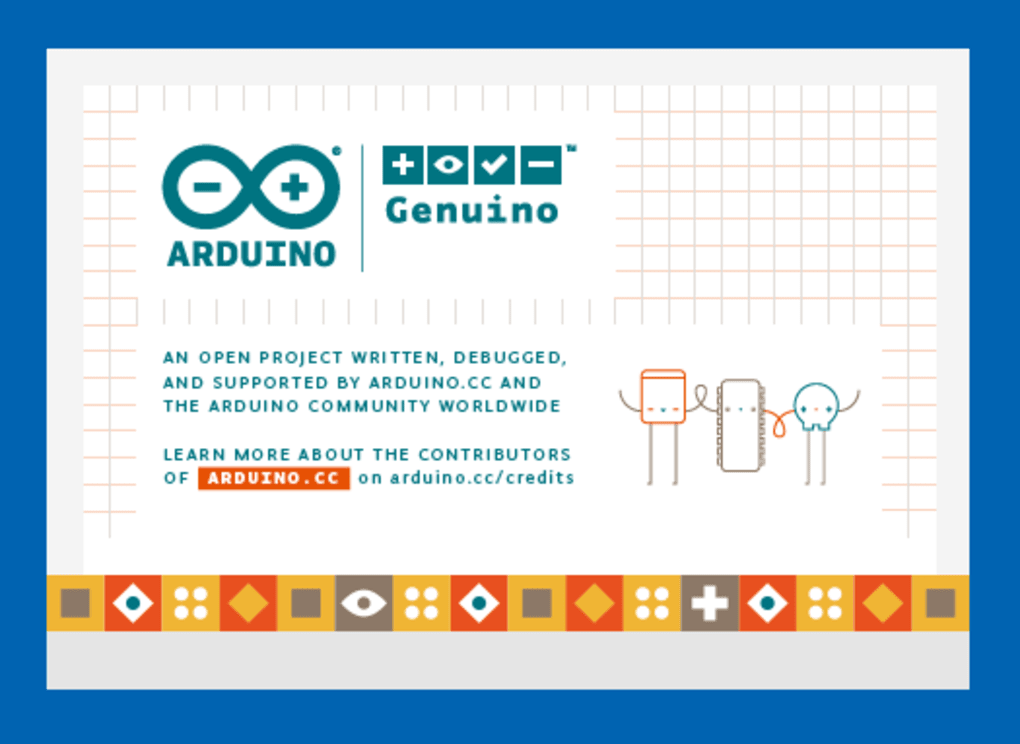
App Inventor and the projects on which it is based are informed by constructionist learning theories, which emphasize that programming can be a vehicle for engaging powerful ideas through active learning.



**ARDUINO IDE:**

The arduino integrated development environment (ide) is a cross-platform application (for windows,macos,linux) that is written in the programming language java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.

The source code for the IDE is released under the GNUgeneral public license license, version 2.The Arduino IDE supports the languages C and c++ using special rules of code structuring. The Arduino IDE supplies a software library from the wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cycle executive program with the GNU tool chain also included with the IDE distribution.



The Arduino IDE employs the program to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

***Conclusion***

In this project,we conclude that water is supplied to irrigation facilities through using relay module and mobile app.Farmers can easily do their work with the help of mobile app or mic.