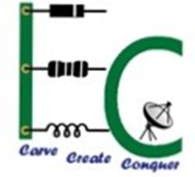
 ****

**THE NATIONAL INSTITUTE OF ENGINEERING**

**MYSURU-570008**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**IOT PROJECT [EC0311] –VII Semester**

**Report On**

SMART PUMP USING ARDUINO

**Under the guidance of:**

**PUNEETH S**

Assistant Professor, Dept. of ECE.

**Submitted By:**

SUDHANSHU SHARMA (4NI15EC109)

T S SUKRUTH (4NI15EC118)

SYED HISHAM SAADI (4NI15EC116)

TABLE OF CONTENTS

1. Acknowledgement -----------------------------------------------------3

2. Abstract -----------------------------------------------------4

3. Aim and Introduction -----------------------------------------------------5

4. Components used -----------------------------------------------------9

5. Working and Block diagram -----------------------------------------------------12

6. Code -----------------------------------------------------14

7. Application -----------------------------------------------------19

ACKNOWLEDGEMENT

We would like to express our profound gratitude to our guide **Mr. PUNEETH S** for his support, encouragement, supervision and useful suggestions throughout this project. His moral support and continuous guidance enabled us tocomplete our work successfully.

We are also thankful for non –teaching staff members who have helped us to get required equipment’s and necessary material to complete this task.

We are also grateful for the co-operation and constant encouragement from our honorable head of the department **Dr. ROHINI NAGAPADMA** for her support and encouragement throughout the project.

ABSTRACT

It is a wireless system to switch on the water pump from anywhere with the help of

SMS (Short message service) and also it acts as a notifying system which will notify

the user/farmer as to when the water pump is on/off

# AIM

To build arduino based smart water pump using GSM module (GPS included)

and a mobile phone with a SIM to test**.**

1. **INTRODUCTION**

Water pumps used in agriculture can cost a lot of money. They range from anywhere around 5000/- to 15000/-. There are times when the water overflows but when this happens and there is no one around to switch it off; it causes a lot of wastage. In this project we are trying to minimize that by sending a notification to the user (farmer) whenever the pump is on or off. Modern agricultural irrigation is a complex interplay of sustainable energy consumption, water use, market conditions, and the application of experience and knowledge to ensure the best design for irrigation applications. Understanding past practices, current water and energy issues, and developments in pump technology contributes to building pumping systems that best service the needs of modern agriculture. The agricultural market is changing rapidly, and farmers cannot rely on the technology and practices of the past.

To keep productivity high and stay competitive in the market, farmers need to focus on profitability, which includes energy optimization and better use of water resources. Pumping systems play a vital role in providing optimized solutions for energy and water use.

A pump system for today’s irrigation is not only about the pumps. Variable speed drives, intelligent control and remote management all necessitate the integration of components in an irrigation system.

Successful agriculture depends on farmers having sufficient access to water. In the middle of the last century, the common perception was that water was an infinite resource. Today, we know that water is a resource that must be managed. This is not only a question of more mouths to feed—people consume more calories and eat more meat. This requires more water to produce food.

Farmers must consider energy consumption. Energy for irrigation pumps is one of the highest single cost drivers for farmers. However, many are unaware of the potential savings from more effective and efficient energy use.

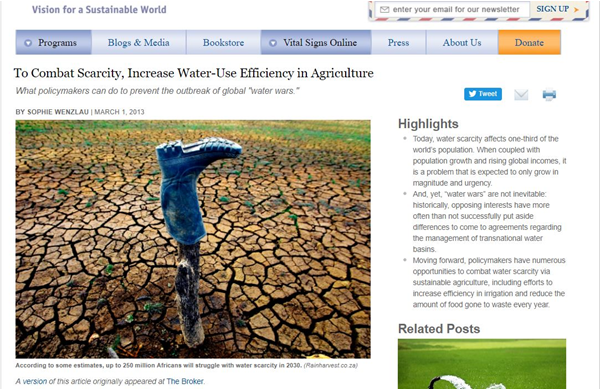
Modern agriculture requires irrigation solutions that optimize uniformity, reduce energy costs, safeguard the water resource and keep productivity at its best. The agricultural market changes require greater focus on applying knowledge, experience and total irrigation solutions integrating all components.

These pumps are required to meet changing conditions above and below ground, which have an effect on the pressure and flow required from day to day and from season to season. A pumping system must deliver the right amount of pressure and flow at the nozzle. The simple solution is to oversize the pump, so the pump is able to handle a worst case scenario. However, as a result, the pump will almost never operate at its optimal duty point. It will produce too much pressure and consume too much energy, which is not used productively in any way.

Traditionally, water has been distributed from the water source—either groundwater or surface water—at low or constant pressure from pumps operating at single speed. Delivery to the crop has been from nozzles, where the focus has been on surface coverage, without much attention placed on run-off, canopy evaporation and wind drift. Soil moisture monitoring to ensure an even spread over the irrigated area is a relatively new discipline.

In contrast, pressure management has long been an issue. Through the years, pressure reduction valves have been used to reduce pressure in the system. However, valves are costly to install and require frequent service and replacement, and their operation consumes a lot of energy.

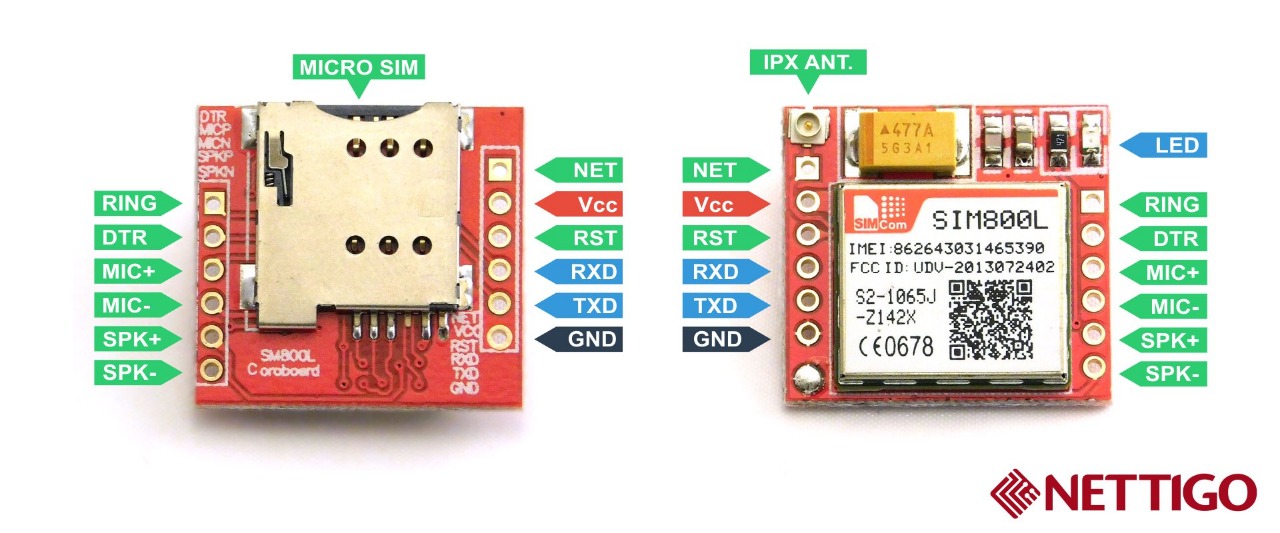
If end users think of an agricultural irrigation system as a car and the pump as the motor, would it make sense to drive the car at constant full throttle and control the speed with the brakes? This is a common approach for irrigation pumps.





1. **COMPONENTS USED**

* Arduino board – Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worring too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards. GSM Module – SIM900A (GPS module included) – It is built with dual band GSM/GPRS based SIM900A modem from SIMCOM. It works on frequencies 900/ 1800 MHz It is controlled via AT commands and the operating temperatures are -40 to +85 deg centigrade. The Arduino Uno can be programmed with the ([Arduino Software](https://www.arduino.cc/en/Main/Software) (IDE)). Select "Arduino/Genuino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the [reference](https://www.arduino.cc/en/Reference/HomePage) and [tutorials](https://www.arduino.cc/en/Tutorial/HomePage). The ATmega328 on the Arduino Uno comes preprogrammed with a [bootloader](https://www.arduino.cc/en/Hacking/Bootloader?from=Tutorial.Bootloader) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](http://www.atmel.com/Images/doc2525.pdf), [C header files](http://www.atmel.com/dyn/resources/prod_documents/avr061.zip)). You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using [Arduino ISP](https://www.arduino.cc/en/Main/ArduinoISP) or similar; see [these instructions](https://www.arduino.cc/en/Hacking/Programmer) for details.
* SIM 800L



Here are the features of the SIM800L breakout board:

* 2G quad-band @ 850/900/1800/1900 MHz
* Receive and make calls using the speaker and microphone outputs
* Receive and send SMS
* Connect to the Internet via GPRS
* Listen to FM radio broadcasts
* Accepts AT Commands

This SIM800L breakout board is ideal for projects that needs to save space. In fact, this is the one I used when I created my own cellphone. This board draws a maximum of 2 A with an input voltage of 3.7 V to 4.2 V. This means you must not connect its pins directly to a 5 V Arduino! It doesn't even run on 3.3 V.

If the power to the SIM800L is enough, the on-board LED starts blinking. The frequency of the blinking means something:

* Every second:  searching for a network.
* Every three seconds: connected to a network.
* Twice per second: connected through GPRS.

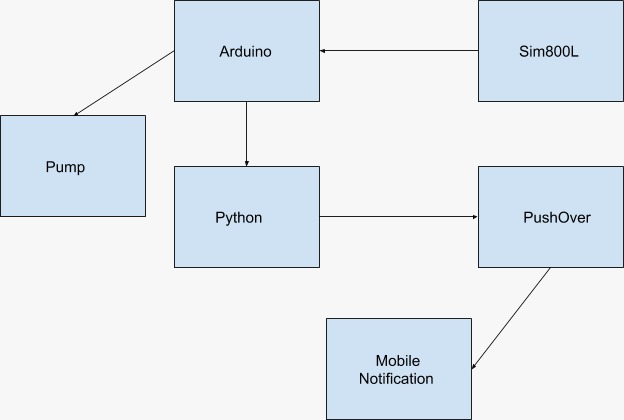
Antennas are essential for this kind of module especially if your project is indoors. Without an antenna, there would not be enough transmitting power for the SIM800L to perform GSM services such as calls and SMS.

* Breadboard or PCB - The substrate most commonly used in **printed** circuit boards is a glass fiber reinforced (fiberglass) **epoxy** resin with a **copper foil** bonded on to one or both sides.
* Power supply – Standard DC supply of 12V is used through the 12V,2A adapter which has a 2.5mmX5.5mm jack
* Connecting wires

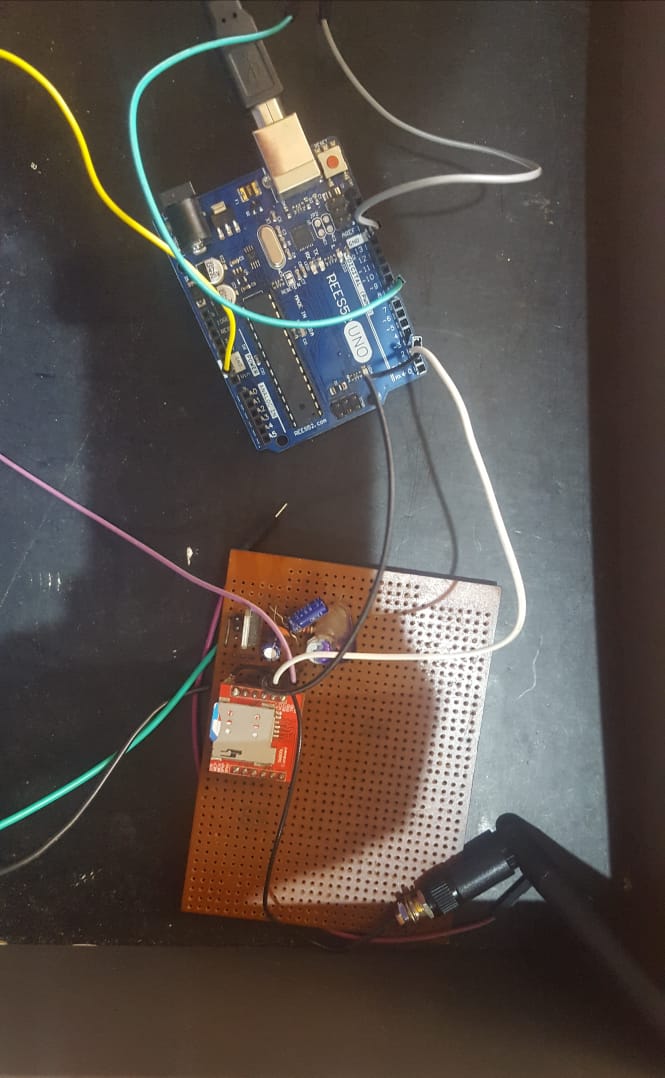
# WORKING

* It’s a two way communication device; it takes readings from user.
* If user send a SMS ‘On’ it reaches sim 800l, arduino validates the text
* Once it is validated it sends the text to python, the code will validate if it matches it will send the user a notification informing them about the status of the pump.

1. **BLOCK DIAGRAM**

****

**SMART WATER PUMP**



# CODE

# import serial #importing the serial so to connect arduino

# import httplib, urllib # importing the push serves

# ser=serial.Serial('/dev/ttyACM0',9600) # connect to arduino

# while True: # starting the infinte loop

# p=ser.readline() # start the reading the arduino value

# r=p.strip() # trimming the left and right space

# print(r) # print to see the output

# 

# if (r=='On' ): # checking the arduino value with desire value

# conn = httplib.HTTPSConnection("api.pushover.net:443")

# conn.request("POST", "/1/messages.json",

# urllib.urlencode({

# "token": "a69um92m2wefa4o84xaqiudb913iw3",

# "user": "utnxsnodqkahrd18ohpc3bf4r6s3ie",

# "message": "pump is currently switched ON",

# }), { "Content-type": "application/x-www-form-urlencoded" })

# conn.getresponse() # printing the ON and sending notification

# if (r=='Off' ): # checking the arduino value with desire value

# conn = httplib.HTTPSConnection("api.pushover.net:443")

# conn.request("POST", "/1/messages.json",

# urllib.urlencode({

# "token": "a69um92m2wefa4o84xaqiudb913iw3",

# "user": "utnxsnodqkahrd18ohpc3bf4r6s3ie",

# "message": "pump is currently switched OFF",

# }), { "Content-type": "application/x-www-form-urlencoded" })

# conn.getresponse() # printing the OFF and sending notification

# ARDUINO CODE

# #include <SoftwareSerial.h>

# SoftwareSerial SIM900(2,3);//tx conn rx conn

# char incoming\_char = 0;

# String readString;

# int ledpin =8;

# void setup()

# {

# Serial.begin(9600); // for serial monitor

# SIM900.begin(9600); // for GSM shield

# delay(2000); // give time to log on to network.

# SIM900.print("AT\r");

# SIM900.print("ATE1\r");

# delay(100);

# SIM900.print("ATE1\r");

# delay(100);

# SIM900.print("AT+CMGF=1\r");

# delay(100);

# SIM900.print("AT+CMGF=1\r");// set SMS mode to text

# delay(100);

# SIM900.print("AT+CNMI=2,2,0,0,0\r");

# // blurt out contents of new SMS upon receipt to the GSM shield's serial out

# delay(100);

# SIM900.print("AT+CNMI=2,2,0,0,0\r");

# Serial.println("Send text message");

# }

# String readSIM900()

# {

# String buffer;

# while (SIM900.available())

# {

# char c = SIM900.read();

# buffer.concat(c);

# delay(10);

# }

# return buffer;

# }

# void loop()

# {

# String buffer = readSIM900();

# 

# if (buffer.startsWith("\r\n+CMT: "))

# {

# 

# //Serial.println("\* RECEIVED SMS \*");

# 

# // Remove first 51 characters

# buffer.remove(0, 51);

# int len = buffer.length();

# // Remove \r\n from tail

# buffer.remove(len - 2, 2);

# //Serial.print("Message send :");

# Serial.println(buffer);

# 

# // Serial.println("\* END SMS \*");

# }

# //Serial.print(Serial.read());

# 

# if (buffer=="On")

# {// Serial.println("Message Recieved is ON hence Motor is ON");

# digitalWrite(ledpin, HIGH);

# }

# if(buffer=="Off")

# { // Serial.println("Message Recieved is OFF hence Motor is OFF");

# digitalWrite(ledpin, LOW);

# }

# delay(100);

# }

# 6. APPLICATIONS:

- It reduces the human efforts and also prevents the water overflow and work as automated water pump. This keeps regular check on field if the water pump is on or not.

- This idea is agriculture centrist but it can be used in other fields of agriculture itself like their field status or their vehicular status and so on.

**GUIDE**

**Mr. Puneeth S**