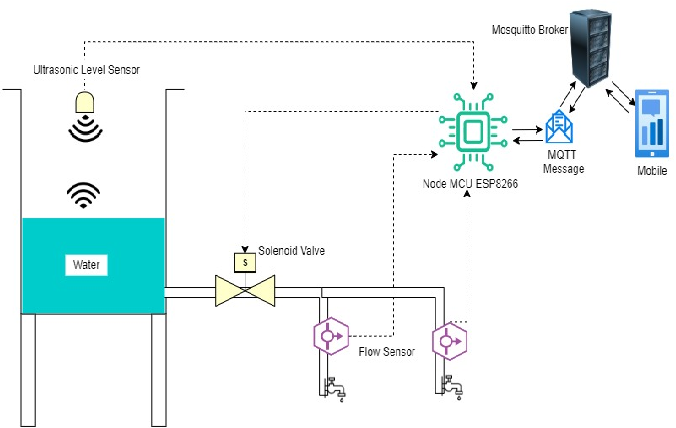
SMART WATER MANAGEMENT

***PROTOTYPE OF MQTT IMPLEMENTATION:***



SOURCE CODE FOR MQTT :

// Code to measure water level in a water tank

// Board: ESP8266 NodeMCU

// 15 May 2021

// Pablo Cruz Lemini

// portfedh@gmail.com

// Mills Function

const unsigned long event\_interval = 300000; // 5 minute interval, 300 second

//const unsigned long event\_interval = 2000; // 2 second interval. For testing

unsigned long previous\_time = 0;

// MQTT Library

#include "EspMQTTClient.h"

#define WIFI\_SSID "<Wifi\_Username\_here>" // WiFi Username

#define WIFI\_PASS "<Wifi\_Password\_here>" // Wifi Password

#define BROKER\_IP "<MQTT\_ip\_here>" // IP adress of MQTT broker

#define BROKER\_USERNAME "<broker\_username\_here>" // Broker username

#define BROKER\_PASSWORD "<broker\_password\_here>" // Broker password

#define CLIENT\_NAME "<device\_name\_here>" // MQTT client name to identify the device

#define BROKER\_PORT <mqtt\_port\_here> // MQTT Port. No "" needed

#define lastwill\_topic "<lastwill\_topic\_here>" // MQTT topic to report lastwill and testament.

#define lastwill\_text "<lastwill\_message\_here>" // MQTT memssage to report lastwill and testament.

String client\_name = CLIENT\_NAME; // MQTT Topic to report initial value

String startup\_topic = "<startup\_topic\_here>"; // MQTT Topic to report startup

String water\_level\_topic = "<reporting\_values\_topic\_here>"; // MQTT topic to report values

// Function to connect to MQTT

EspMQTTClient client(

WIFI\_SSID,

WIFI\_PASS,

BROKER\_IP,

BROKER\_USERNAME,

BROKER\_PASSWORD,

CLIENT\_NAME,

BROKER\_PORT

);

// Water Sensor pins

#define TRIG 14 //GPIO Number 14, D5

#define ECHO 12 //GPIO Number 12, D6

void setup() {

Serial.begin(115200); // Serial monitoring

// Enable debugging messages sent to serial output

client.enableDebuggingMessages();

// Enable the web updater.

client.enableHTTPWebUpdater();

// MQTT Last Will & Testament

client.enableLastWillMessage( lastwill\_topic , lastwill\_text);

// Water level sensor Pin Setup

pinMode(TRIG, OUTPUT); // Initializing Trigger Output

pinMode(ECHO, INPUT\_PULLUP); // Initializing Echo Input

}

// MQTT Innitial Connection

void onConnectionEstablished() {

client.publish(startup\_topic, String(client\_name + " is now online."));

}

void loop() {

// MQTT Loop: Must be called once per loop.

client.loop();

// Mills Loop

unsigned long current\_time = millis();

// Mills if Statement

if(current\_time - previous\_time >= event\_interval) {

// Set the trigger pin to low for 2uS

digitalWrite(TRIG, LOW);

delayMicroseconds(2);

// Send a 20uS high to trigger ranging

digitalWrite(TRIG, HIGH);

delayMicroseconds(20);

// Send pin low again

digitalWrite(TRIG, LOW);

// Read pulse times

int distance = pulseIn(ECHO, HIGH,26000);

//Convert the pulse duration to distance

distance= distance/58;

//Print Result in serial monitor

Serial.print("Distance ");

Serial.print(distance);

Serial.println("cm");

// MQTT Client Publisher

client.publish(water\_level\_topic, String(distance));

// Mills Update timing for next time

previous\_time = current\_time;

}

}

The graph shows 15 days period of water consumption:

