LAMPIRAN

Kode Program :

```
#include <TFT_eSPI.h>
#include <Arduino.h>
#include <ArduinoJson.h>
#include <SPI.h>
#include <WiFi.h>
#include <CTBot.h>

CTBot myBot;
TFT_eSPI tft = TFT_eSPI();

//flow dan led
#define SENSOR 13
#define LED 12

//turby
int pinTurb = 36;
static float V;
static float kekeruhan;
float volt;

const char *WIFI_SSID = "Maulana.M.zen";
const char *WIFI_PASSWORD = "Maulana0817";



//con telegram
String token = "5002219418:AAGy39fLNuvLol2iEFjEEFcO4ksCnu18Kgc";
const int id = 1217879404;
int botRequestDelay = 1700;
unsigned long lastTimeBotRan;

//flow
int interval = 1000;
boolean ledState = false, led = false;
float calibrationFactor = 4.5;
volatile byte pulseCount;
byte pulse1Sec = 0;
float flowRate;
unsigned int flowMilliLitres, kalkulasiAir;
unsigned long totalMilliLitres, totalMilliLitresPrev, counterKalkulasiAir,
currentMillis = 0, previousMillis = 0, previousMillis2 = 0, mainTimer = 0,
prevPerJam = 0, prevRead = 0, prevTampil = 0, millisLED = 0;
volatile int count;

void IRAM_ATTR pulseCounter()
{
  pulseCount++;
}

void Flow()
{
```

```
  count++;
}
WiFiClient net;
void connectToWiFi()
{

  WiFi.mode(WIFI_STA);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  int retries = 0;
  while (WiFi.status() != WL_CONNECTED && retries < 15)
  {
    delay(500);
    Serial.print(".");
    retries++;
  }

  if (WiFi.status() != WL_CONNECTED) {
    ledState = false;
  } else
  {
    ledState = true;
  }
}


void setup()
{
  tft.begin();
  tft.setRotation(1);
  Serial.begin(115200);
  pinMode(SENSOR, INPUT_PULLUP);
  pinMode(LED, OUTPUT);
  digitalWrite(LED, HIGH);
  attachInterrupt(0, Flow, RISING);
  delay(20);

  pulseCount = 0;
  flowRate = 0.0;
  flowMilliLitres = 0;
  totalMilliLitres = 0;
  previousMillis = 0;
  attachInterrupt(digitalPinToInterrupt(SENSOR), pulseCounter, FALLING);

  Serial.println(" mulai telegram");

  connectToWiFi();
  if (WiFi.isConnected() == true)
  {
    Serial.println("sudah terhubung");
    ledState = true;
  }
  else
  {
    connectToWiFi();
    Serial.println("Menghubungkan Ulang");
```

```
  }
  //connect wifi
  myBot.wifiConnect(WIFI_SSID, WIFI_PASSWORD);
  //set token
  myBot.setTelegramToken(token);
  //cek tele
  if (myBot.testConnection() )
    Serial.println("koneksi tele berhasil");
  else
    Serial.println ("koneksi tele gagal");


}
void loop()
{

  currentMillis = millis();
  if (currentMillis - previousMillis2 > 1000)
  {
    previousMillis2 = currentMillis;
    mainTimer++;
  }

  //READ
  if (currentMillis - previousMillis > interval)
  {
    pulse1Sec = pulseCount;
    pulseCount = 0;

    flowRate = ((1000.0 / (millis() - previousMillis)) * pulse1Sec) /
calibrationFactor;
    previousMillis = millis();

    flowMilliLitres = (flowRate / 20) * 1000;

    totalMilliLitres += flowMilliLitres;
    if (totalMilliLitres != totalMilliLitresPrev)
    {
      counterKalkulasiAir = totalMilliLitres - totalMilliLitresPrev;
      kalkulasiAir = kalkulasiAir + counterKalkulasiAir;
      totalMilliLitresPrev = totalMilliLitres;
    }
  }

  if (mainTimer >= prevPerJam + 86400)
  {
    kalkulasiAir = 0;
    prevPerJam = mainTimer;

  }
  if (mainTimer >= prevPerJam + 3600 )
  {
    flowRate = 0;
    prevPerJam = mainTimer;
```

```
}
volt = 0;
for (int i = 0; i < 200; i++)
{
  volt += ((float)analogRead(pinTurb) / 1023) * 1.25;
}
volt = volt;
int SensorKekeruhan = analogRead(pinTurb);
V = SensorKekeruhan * (5.0 / 1024);
kekeruhan = 2.8 + (V * 100.00);

Serial.print("Voltage     :");
Serial.print(volt);
Serial.println("  V");
Serial.print("Nilai ADC   :");
Serial.println(SensorKekeruhan);
Serial.print("kekeruhan   :");
Serial.print(kekeruhan);
Serial.println("  NTU");

if ((SensorKekeruhan >= 0) && (SensorKekeruhan <= 15)) {
  Serial.print("Status Air  :");
  Serial.println("AIR BERSIH");
}
else if ((SensorKekeruhan >= 16) && (SensorKekeruhan < 23)) {
  Serial.print("Status Air  :");
  Serial.println("SEDANG");
}
else {
  Serial.print("Status Air  :");
  Serial.println("AIR KERUHH");
}
delay(2000);
//display
tft.setTextSize(2);
tft.fillScreen(TFT_BLACK);
tft.setTextColor(TFT_WHITE, TFT_BLACK);

tft.setCursor(0, 00);
tft.println("==STATUS SENSOR 1==");
tft.println("_____________________");

if ((SensorKekeruhan >= 0) && (SensorKekeruhan <= 15)) {
  tft.println("STATUS AIR : BERSIH ");
  Serial.println("BERSIH");

}
else if ((SensorKekeruhan >= 16) && (SensorKekeruhan < 23)) {
  tft.println("STATUS AIR :SEDANG ");
  Serial.println("SEDANG");
}
else {
  tft.println("STATUS AIR : KERUH");
  Serial.println("KERUH");
}
tft.print("KEKERUHAN :");
```

```
tft.print(SensorKekeruhan);
tft.println(" NTU");

tft.print("DEBIT :");
tft.print(flowRate);
tft.println(" L/Min");

tft.print("AVG :");
tft.print ((float)kalkulasiAir / 1000);
tft.println (" L/Hari ");

tft.print("TOTAL : ");
tft.print((float)totalMilliLitres / 1000);
tft.println(" L ");

if (mainTimer >= prevTampil + 5)
{
  // DEBIT/MENIT
  Serial.print(" Debit              : ");
  Serial.print(int(flowRate)); // Print the integer part of the variable
  Serial.println(" L/Min ");

  // TOTAL DARI NYALA
  Serial.print(" Total              : ");
  Serial.print(totalMilliLitres);
  Serial.println(" L ");

  // RATA-RATA PERJAM
  Serial.print(" Rata-rata Air/Jam  : ");
  Serial.print(kalkulasiAir);
  Serial.println(" L/Hari");
  prevTampil = mainTimer;
}

//telegram

//flow meter 2
TBMessage msg;

if (flowRate  > 3.00)
{
  myBot.sendMessage (msg.sender.id , "PENYIRAMAN 1 BAIK  ");
}

//pesan telegram
if (myBot.getNewMessage(msg))
{
  if (msg.text.equalsIgnoreCase("/cek status"))
  {
    Serial.println("pesan2 diterima..");

    if ((SensorKekeruhan >= 0) && (SensorKekeruhan <= 15)) {
      myBot.sendMessage (msg.sender.id , "STATUS AIR : BERSIH ");

    }
    else if ((SensorKekeruhan >= 16) && (SensorKekeruhan < 23)) {
```

```
        myBot.sendMessage (msg.sender.id , "STATUS AIR :SEDANG ");
        Serial.println("SEDANG");
      }
      else {
        myBot.sendMessage (msg.sender.id , "STATUS AIR : KERUH");
      }
      //variable pesan kirim
      String balasan;
      balasan =
        (String)"Tingkat Kekeruhan :" + kekeruhan + (String) " NTU \n" +
        (String)"FLOW METER 1 :" + " \n" +
        (String)"Debit :" + (int)flowRate + (String) "L/Min \n" +
        (String)"avg/H : " + (float)kalkulasiAir / 1000 + (String) "L/hari \n" +
        (String)"Total :" + (float)totalMilliLitres / 1000 + (String) "L \n" ;
      myBot.sendMessage(msg.sender.id, balasan);
    }
    // pesan telegram kekeruhan air
    if (msg.text.equalsIgnoreCase("/kondisi air"))
    {
      Serial.print("pesan2 diterima");
      //variabel balasan kondisi air
      String balasan2;
      balasan2 =
        (String)"TINGKAT KEKERUHAN :" + kekeruhan + (String) "NTU \n" +
        (String)"NILAI ADC  : " + SensorKekeruhan + (String)" \n" ;

      if ((SensorKekeruhan >= 0) && (SensorKekeruhan <= 15)) {
        myBot.sendMessage (msg.sender.id , "KONDISI AIR BERSIH ");
      }
      else if ((SensorKekeruhan >= 16) && (SensorKekeruhan < 23)) {
        myBot.sendMessage (msg.sender.id , "SEDANG ");
        Serial.println("SEDANG");
      }
      else {
        myBot.sendMessage (msg.sender.id , "KONDISI AIR KERUH ");
      }

      myBot.sendMessage(msg.sender.id, balasan2);
    }


    if (msg.text.equalsIgnoreCase("/aliran"))
    {
      Serial.print("pesan3 diterima");
      //variabel balasan kondisi flow meter 1
      String balasan3;
      balasan3 =  (String)" == FLOW METER 1 ==" + (String)"\n" +
                  (String)"Debit :" + (int)flowRate + (String) "L/Min \n" +
                  (String)"avg/H : " + (float)kalkulasiAir / 1000 + (String)
"L/hari \n" +
                  (String)"Total :" + (float)totalMilliLitres / 1000 + (String) "L
\n" ;
      myBot.sendMessage(msg.sender.id, balasan3);
    }
  }
```

```
//ngirim mqtt
if (WiFi.isConnected() == true)
{
  {
    Serial.println("Terhubungg cok..");
    ledState = true;
  }

  {
    Serial.print("TIDAK TERHUBUNG JANCOKKKKKK.........");
    ledState = false;
    digitalWrite(LED, HIGH);
    {
      ledState = true;
    }
    {
      Serial.println("Gagal Reconnect MQTT");
      ledState = false;
    }
  }
}
else
{
  Serial.println("Menghubungkan Ulang");
  digitalWrite(LED, HIGH);
  connectToWiFi();
  if (WiFi.isConnected() == true)
  {
    {
      ledState = true;
    }
    {
      ledState = false;
    }
  }
  else
  {
    Serial.println("Wifi Gagal Reconnect");
    ledState = false;
  }
}
//LED
if (ledState == false)
{
  if (currentMillis - millisLED > 200) {
    led = ! led;
    digitalWrite(LED, led);
    millisLED = currentMillis;
  }
} else if (ledState == true) {
  if (currentMillis - millisLED > 1500) {
    led = ! led;
    digitalWrite(LED, led);
    millisLED = currentMillis;
  }
```

```
    //digitalWrite(pinLED, LOW);
  }
}
```

DataSheet Flow Meter MODEL: YF-S201

Description:

Water flow sensor consists of a plastic valve body, a water rotor, and a hall-effect sensor. When water flows through the rotor, rotor rolls. Its speed changes with different rate of flow. The hall-effect sensor outputs the corresponding pulse signal. This one is suitable to detect flow in water dispenser or coffee machine. We have a comprehensive line of water flow sensors in different diameters. Check them out to find the one that meets your need most.

Features:

Compact, Easy to Install
High Sealing Performance
High Quality Hall Effect Sensor
RoHS Compliant

Specifications:

Working Voltage: DC 4.5V~24V
Normal Voltage: DC 5V~18V
Max. Working Current: 15mA (DC 5V)
Load capacity: ≤ 10 mA (DC 5V)
Flow Rate Range: 1~30L/min
Load Capacity: ≤10mA (DC 5V)
Operating Temperature: ≤80℃
Liquid Temperature: ≤120℃
Operating Humidity: 35%～90%RH
Allowing Pressure: ≤1.75MPa
Storage Temperature: -25～+ 80℃
Storage Humidity: 25%～95%RH
Electric strength 1250V/min
Insulation resistance ≥ 100MΩ
External threads: 1/2"
Outer diameter: 20mm

Intake diameter: 9mm
Outlet diameter: 12mm

Application:

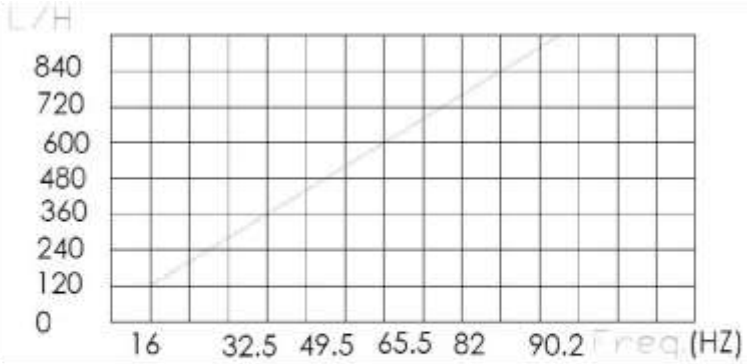
Water heaters, credit card machines, water vending machine, flow measurement device!

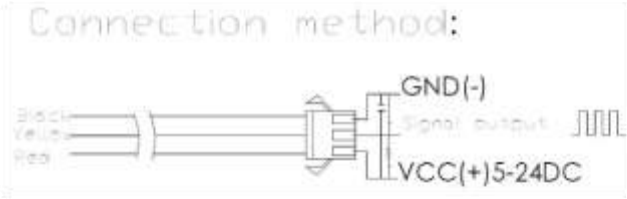
Cercuit:

Red: Positive
Black: GND
Yellow: Output signal

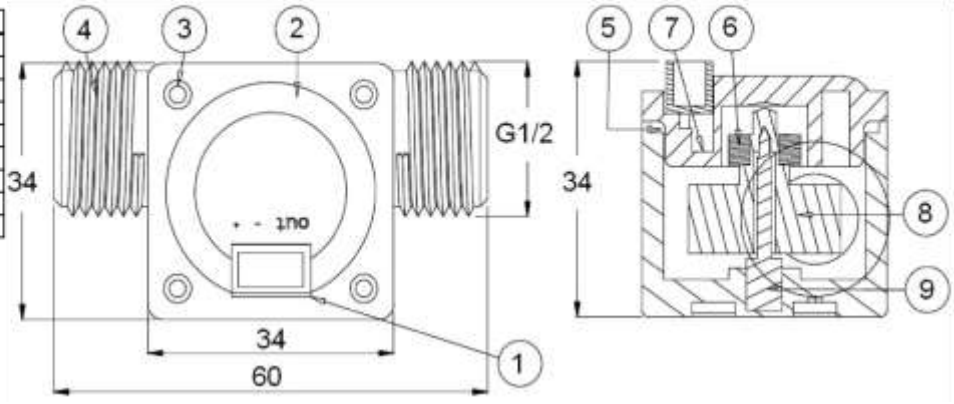
Flow Range: 100L/H-/1800H-L/H

Flow (L/H)	Frezq.(Hz)	Erro range
120	16	
240	32.5	
360	49.3	±10
480	65.5	5%
600	82	
720	90.2	



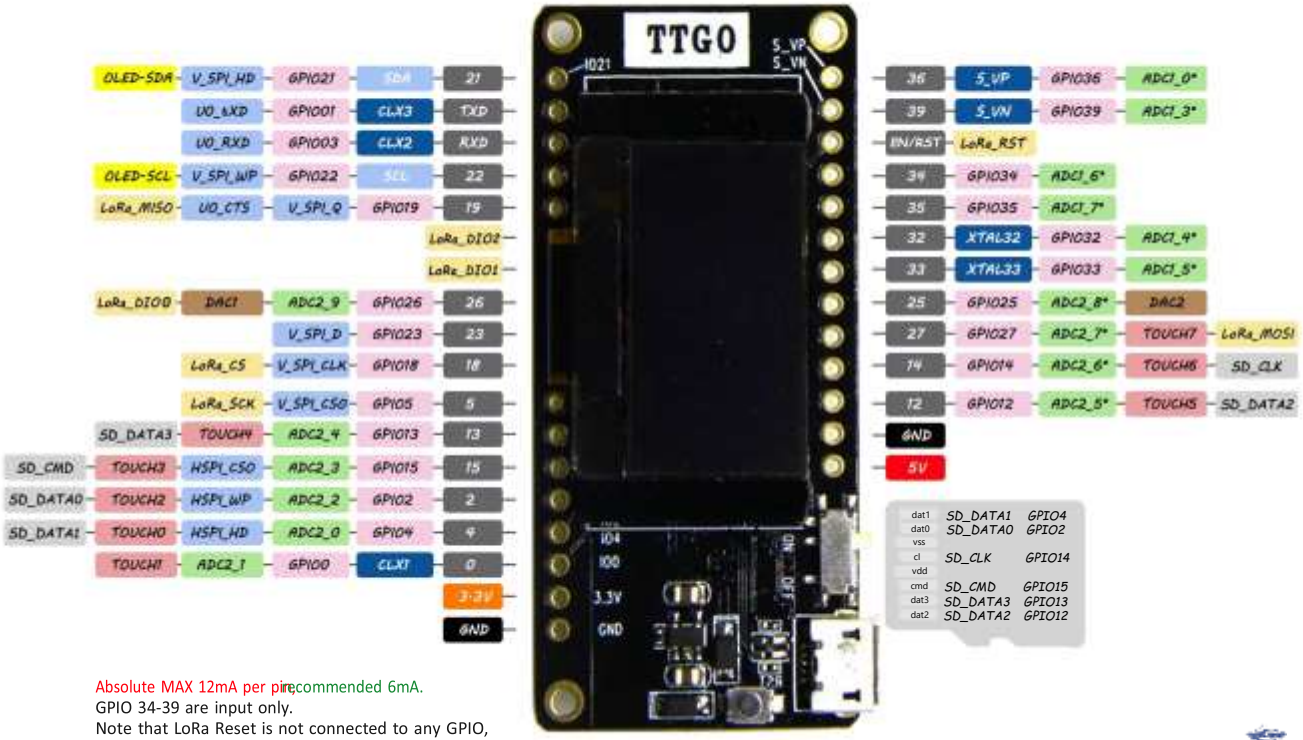


N°	Item	Material
1	Wire	PVC
2	Bonnet	PA
3	Screw	Zinc Plated
4	Valve Body	PA
5	Press Valve	
6	Magnet	
7	Hall	
8	Impeller	POM
9	Steel Sharft	SUS304



Datasheet Viscosity sensor
Datasheet Viscosity sensor

TTGO T Display V2



Absolute MAX 12mA per pin, recommended 6mA.
GPIO 34-39 are input only.
Note that LoRa Reset is not connected to any GPIO, but is hard-wired to the ESP32 Chip Enable (EN) pin.

Version 3.0

U.S.A.
PAC, LP | 101 Station Landing| Medford, MA 02155
T: +1 800.554.4836 | O: +1 781.393.6500 | F: +1 781.393.6515
sales.usa-boston@paclp.com

PAC Authorized Representatives are also located in most countries worldwide. For more information visit www.paclp.com.

SPECIFICATIONS

MODEL	VISCOSITY SENSOR 374	VISCOSITY SENSOR 501	VISCOSITY SENSOR 571
Viscosity Range	0.2-10,000cP	0.5-500cP	
Sensor Range	0.2-2cP, 0.25-5cP, 0.5-10cP, 1-20cP, 2.5-50cP, 5-100cP, 10-200cP, 25-500cP, 50-1,000cP, 100-2,000cP, 250-5,000cP, 500-10,000cP,	0.5-10cP, 1-20cP, 2.5-50cP, 5-100cP, 10-200cP, 25-500cP	
Viscosity Accuracy	± 1.0% of full scale with VISCOpro 2000		
Viscosity Repeatability	± 0.8% of Reading		
Temperature Sensor	PT100		
Wetted Components	316L / 430 Stainless Steel		
Maximum Particle Size	25-800 Microns (Range Dependent)	25-100 Microns (Range Dependent)	
Maximum Ambient Temperature	212°F (100°C)	375°F (190°C)	
Maximum Operating Pressure	1,000 psi (70.3 bar)		
Standard Cable Length	15 Feet (4.57 Meter)	3 Feet (1 Meter)	
Power	Requires VISCOpro2000 electronics	Requires VISCOpro electronics	
Heater	24VDC/66w		
System Volume	< 6ml		

Cambridge Viscosity

With more than 10,000 installations worldwide, Cambridge Viscosity is the proven leader in viscosity management technology. With over 25 years of experience, Cambridge Viscosity understands and meets the needs of laboratory researchers and process engineers in a wide range of industries whose jobs depend on the quality, accuracy, and reliability of viscosity measurement equipment. With their patented sensor technology, Cambridge Viscosity has become the gold standard in small sample viscosity measurement.

www.paclp.com