

RPM Meter using Nodemcu of Micropump Motor

Dinesh Kumar

ISRO Satellite Center

Bangalore

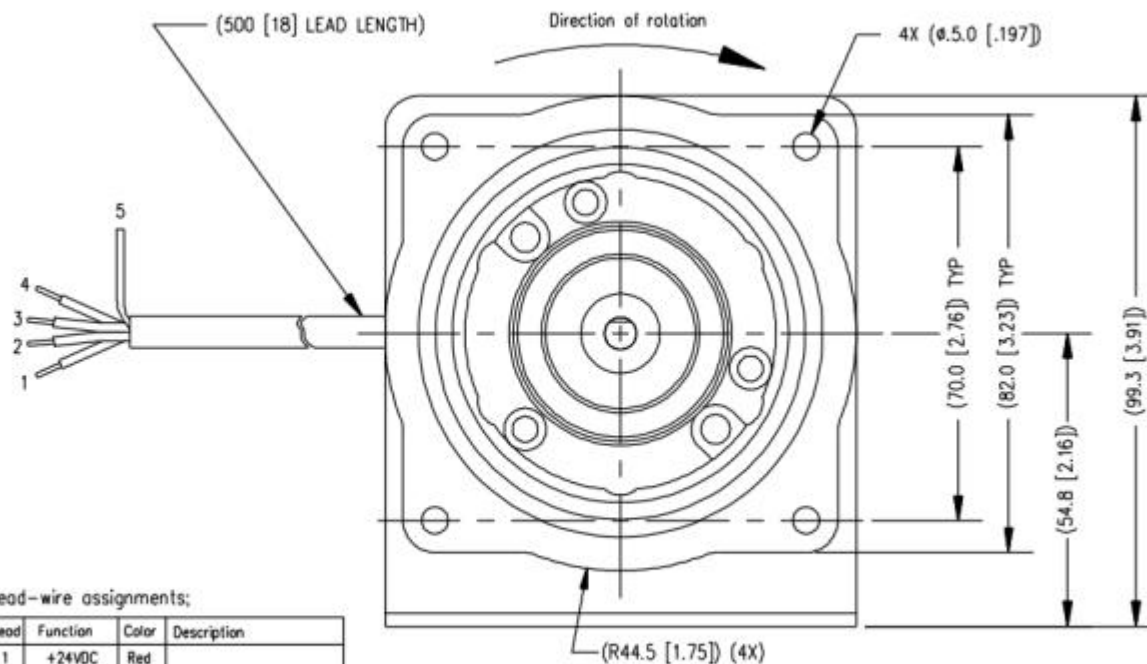


PRO PUMP

THIS PRODUCT MAY BE COVERED
BY ONE OF THE FOLLOWING
PATENTS OR PENDING PATENTS:
U.S. PAT. 4,111,814; 4,127,282;
4,183,844; 4,532,448; 4,532,449;
JAPAN PAT. 1,384,117

A-MOUNT (MM [INCHES])

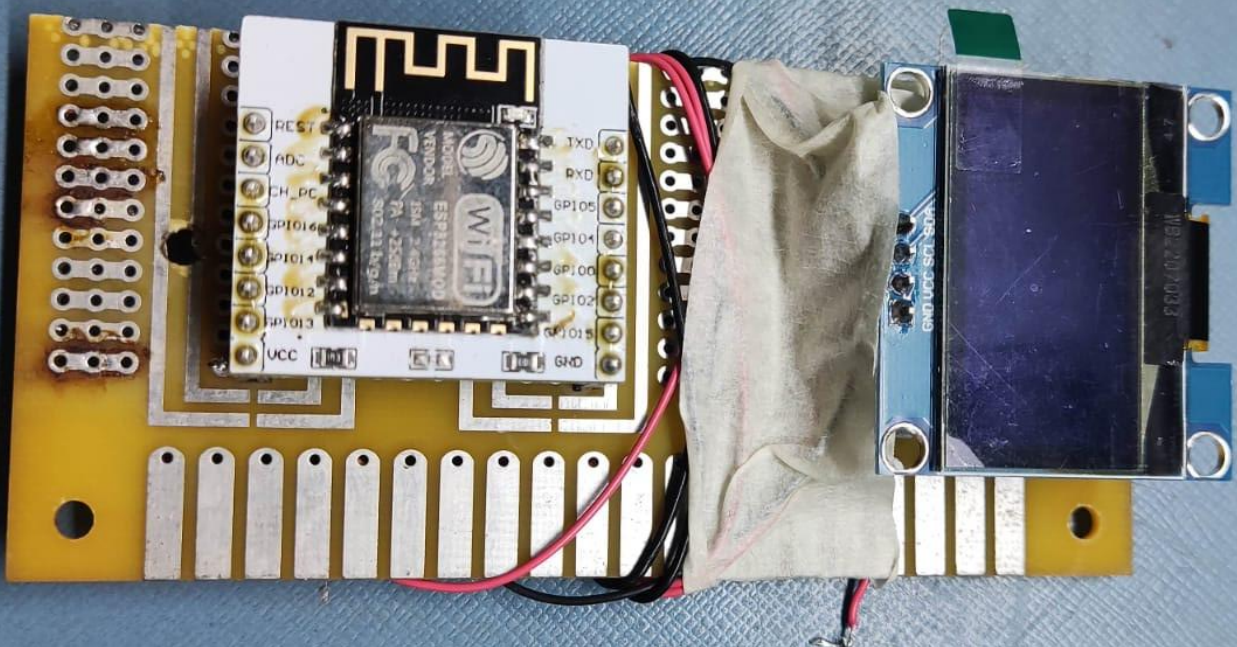
Weight: 830 $\pm$ 15 grams (29.28 $\pm$ 0.53 oz.)



Lead-wire assignments:

Lead	Function	Color	Description
1	+24VDC	Red	
2	Ground	Black	
3	Speedcontrol	White	0-5 VDC
4	Tach out	Green	2 pulses per revolution ⁽¹⁾
5	Shield/ground	-	Motor housing

1) 0-5V square wave. Output impedance = 2kOhm

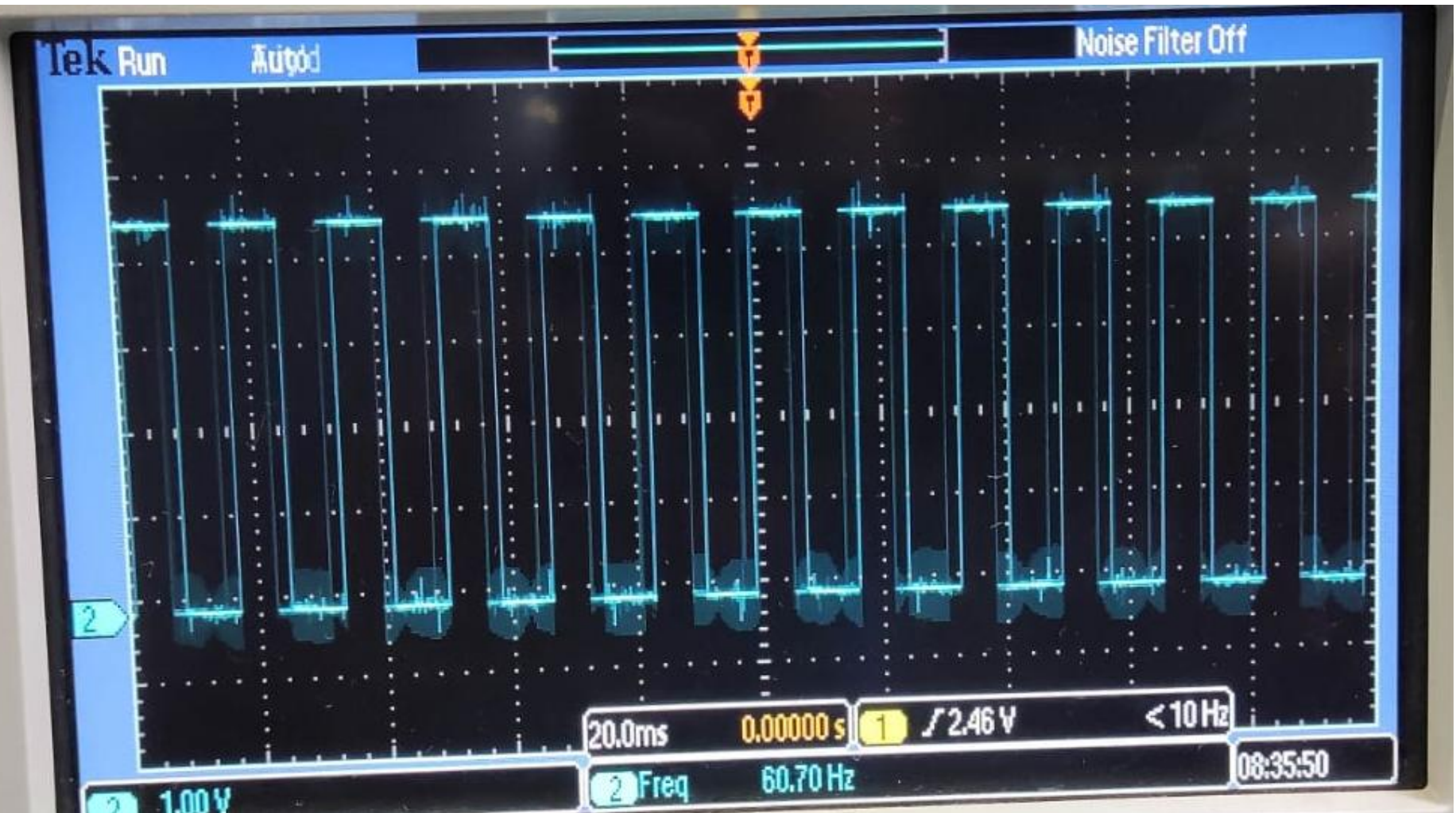


7

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SH110X.h>
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1 // Reset pin
#define SCREEN_ADDRESS 0x3C
Adafruit_SH1106G display = Adafruit_SH1106G(SCREEN_WIDTH,
SCREEN_HEIGHT, &Wire, OLED_RESET);
#define BUTTON_PIN D3
void setup()
{
  Serial.begin(9600);
  pinMode(BUTTON_PIN, INPUT);
  display.begin(SCREEN_ADDRESS, true);
  display.display();
  delay(2000);
  display.clearDisplay();
  display.setTextSize(2);
  display.setTextColor(SH110X_WHITE);
  display.setCursor(0,0);
  display.println("  RPM ");
  display.println("  AND ");
  display.println(" Pressure");
  display.println("  Meter ");
  display.display();
  delay(2000);
}
```

```
void loop()
{
float duration = pulseIn(BUTTON_PIN, HIGH);
float pulsePeriod = duration/500 ;
float frequency = 1 / pulsePeriod;
Serial.print("Time Period = ");
Serial.print(pulsePeriod);
Serial.println(" mili second");
Serial.print("Frequency = ");
Serial.print(frequency);
Serial.println(" KHz");
Serial.print("Motor RPM = ");
Serial.print(frequency * 30);
Serial.println(" rpm");
Serial.println("");
display.display();
display.clearDisplay();
display.setTextSize(2);
display.setTextColor(SH110X_WHITE);
display.setCursor(0,0);
display.println("  RPM ");
display.println("");
display.print("  ");
if(duration > 100 )
{
frequency = 0;
display.print(frequency * 30);
}
else {display.print(frequency * 30);}
delay(5000);
}
```

Getting 5v pulse from RPM pin of Motor – 60 Hz frequency



Counting pulse using Arduino and after mathematical formula we are getting RPM.
Displaying in OLED.

Frequency = 60 Hz and RPM = frequency * 30 = 1800) since 2 pulse generated every rotation.

