LAMPIRAN

Listing program Mikrokontroller 1(Transmitter, LCD, waterflow)

```
=====================================================
#include <VirtualWire.h>
/*
Liquid flow rate sensor -DIYhacking.com ArvindSanjeev
Measure the liquid/water flow rate using this code.
Connect Vcc and Gnd of sensor to arduino, and the
signal line to arduino digital pin 2.

 */
bytestatusLed    = 13;

bytesensorInterrupt = 0;  // 0 = digital pin 2
bytesensorPin       = 2;
floatcalibrationFactor = 4.5;
volatile byte pulseCount;
floatflowRate;

unsigned long oldTime;
// include the library code:
#include <LiquidCrystal.h>
// initialize the library with the numbers of the interface pins
LiquidCrystallcd(9, 8, 7, 6, 5, 4 );
void setup()
{
vw_set_tx_pin (10);
vw_setup (2000);
```

```
  // Initialize a serial connection for reporting values to the host
Serial.begin(9600);
  // Set up the status LED line as an output
pinMode(statusLed, OUTPUT);
digitalWrite(statusLed, HIGH);  // We have an active-low LED attached
pinMode(sensorPin, INPUT);
digitalWrite(sensorPin, HIGH);
lcd.begin(16, 2);
  // Print a message to the LCD.
lcd.print(" abdur.fadillah  ");
lcd.setCursor(0, 0);
pulseCount       = 0;
flowRate         = 0.0;
attachInterrupt(sensorInterrupt, pulseCounter, FALLING);
}
void loop(){


detachInterrupt(sensorInterrupt);
flowRate = ((1000.0 / (millis() - oldTime)) * pulseCount) / calibrationFactor; //
rumusflowmeter


oldTime = millis();
unsignedintfrac;
    // Print the flow rate for this second in litres / minute
Serial.print("Flowrate: ");
Serial.print(int(flowRate));  // Print the integer part of the variable
Serial.print(".");   // Print the decimal point
lcd.setCursor(0,1);
lcd.print("Flow rate: ");
lcd.print(int(flowRate));  // Print the integer part of the variable
lcd.print(".");            // Print the decimal point
   // Determine the fractional part. The 10 multiplier gives us 1 decimal place.
frac = (flowRate - int(flowRate)) * 10;
Serial.print(frac, DEC) ;      // Print the fractional part of the variable
Serial.println("L/min");
lcd.print(frac, DEC) ;      // Print the fractional part of the variable
lcd.println("L/min");

if (flowRate>10)
{ kirimPerintah("OKE");
  }
else if (flowRate<10)
{ kirimPerintah("NO");
  }
   // Reset the pulse counter so we can start incrementing again
pulseCount = 0;

   // Enable the interrupt again now that we've finished sending output
attachInterrupt(sensorInterrupt, pulseCounter, FALLING);
delay (1000);
  }

/*
Insterrupt Service Routine
 */
voidpulseCounter()
{
 // Increment the pulse counter
pulseCount++;
}
voidkirimPerintah(char*perintah)
{
Serial.print (perintah);
vw_send ((uint8_t*)perintah,strlen (perintah));
vw_wait_tx();


}
```

Listing Program Mikrokontroller 2 (RF Receiver)

```
=========================================

#include <VirtualWire.h>


constint PIN_LED = 13; // Normal
constint PIN_LED1 = 12; //Bocor
bytewadah [VW_MAX_MESSAGE_LEN];
bytepanjangPesan = VW_MAX_MESSAGE_LEN;
void setup ()
{
pinMode (PIN_LED,OUTPUT);

pinMode (PIN_LED1,OUTPUT);

vw_set_rx_pin (9);
vw_setup (2000); // bit perdetik
vw_rx_start (); // memulaipenerimaan data
Serial.begin (9600);
Serial.print ("Siapmenerimaperintah");
}
void loop()
{
if (vw_get_message (wadah, &panjangPesan))
{ //peroleh string perintah
charperintah [VW_MAX_MESSAGE_LEN+1];
int j;
for (j=0; j<panjangPesan; j++)
perintah [j]= wadah [j];
perintah [j]=0; // pengakhir string
  //proses perintah
if (strcmp(perintah,"OKE")==0)
  {
Serial.print ("LED ON!");

digitalWrite(PIN_LED,HIGH);
digitalWrite(PIN_LED1,LOW);
  }
else  if (strcmp (perintah,"NO" )==0)
 {
Serial.print (" LED OFF!");

digitalWrite(PIN_LED,LOW);
digitalWrite(PIN_LED1,HIGH);
  }
else
  {
Serial.println ("Maafsayatidakmengenalperintah:");
Serial.println(perintah);
  }
  //reset ke VW_MAX_MESSAGE_LEN
  //untukmengatasi bug di pustakaVirtualWire
panjangPesan = VW_MAX_MESSAGE_LEN;
}
}
```