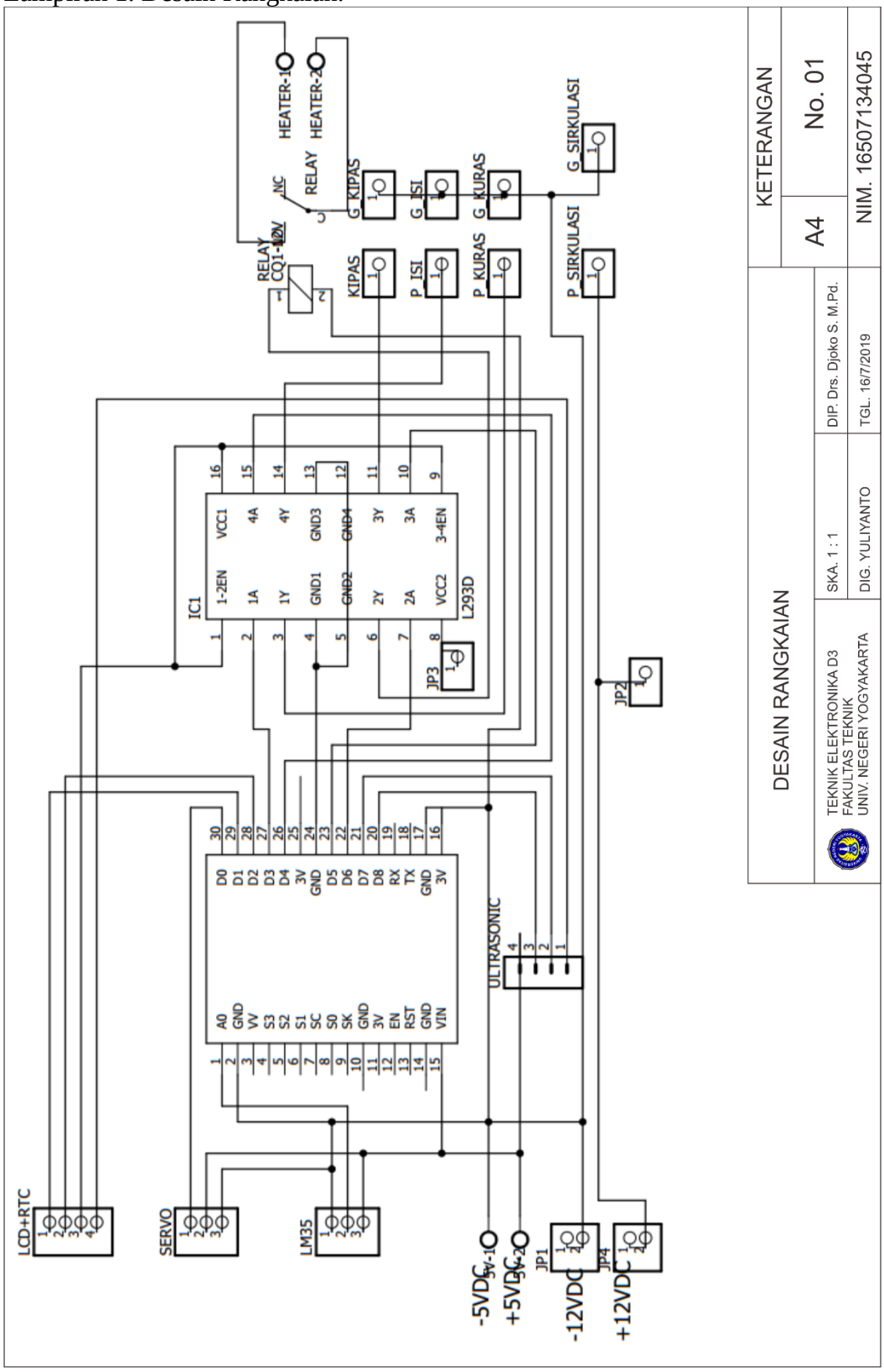
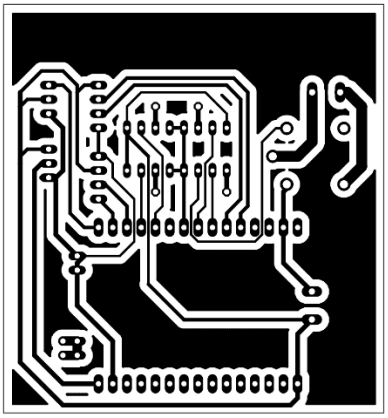
LAMPIRAN

Lampiran 1. Desain Rangkaian.

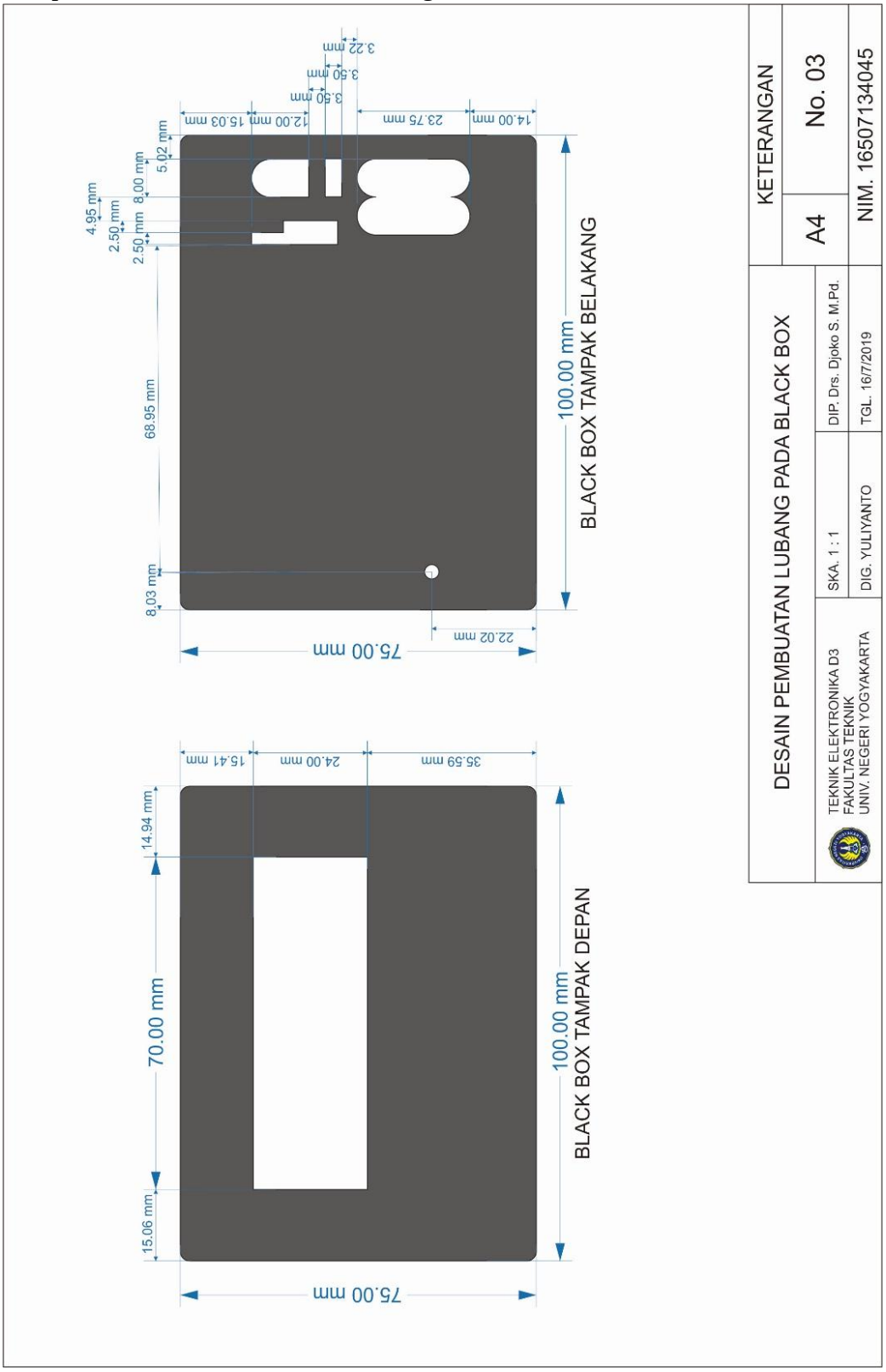


Lampiran 2. Desain PCB.

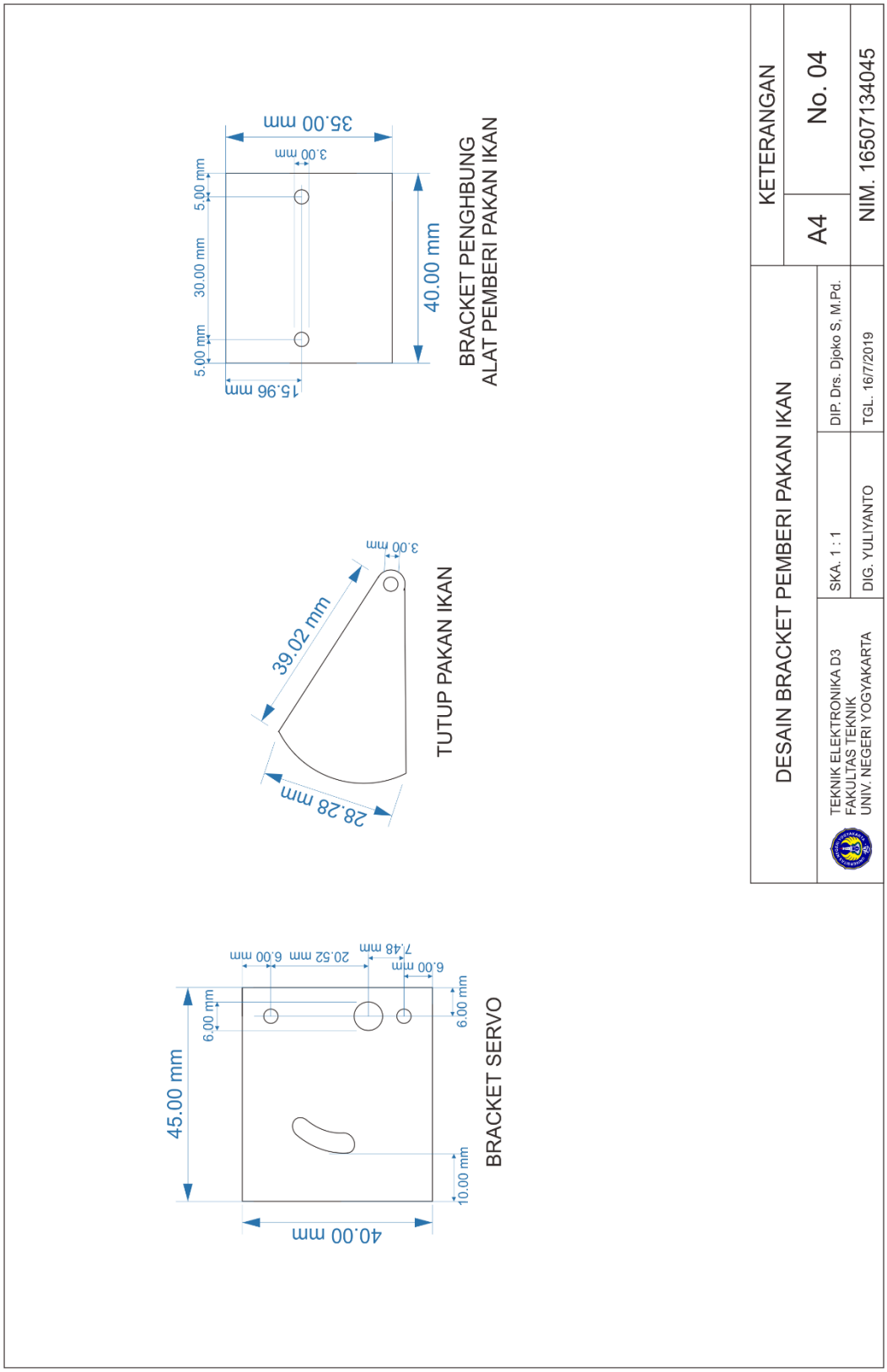


DESAIN PCB		KETERANGAN	
TEKNIK ELEKTRONIKA D3 FAKULTAS TEKNIK UNIV. NEGERI YOGYAKARTA	SKA. 1 : 1	A4	No. 02
	DIG. YULIYANTO	DIP. Drs. Djoko S. M.Pd.	
	TGL. 16/7/2019	NIM. 16507134045	

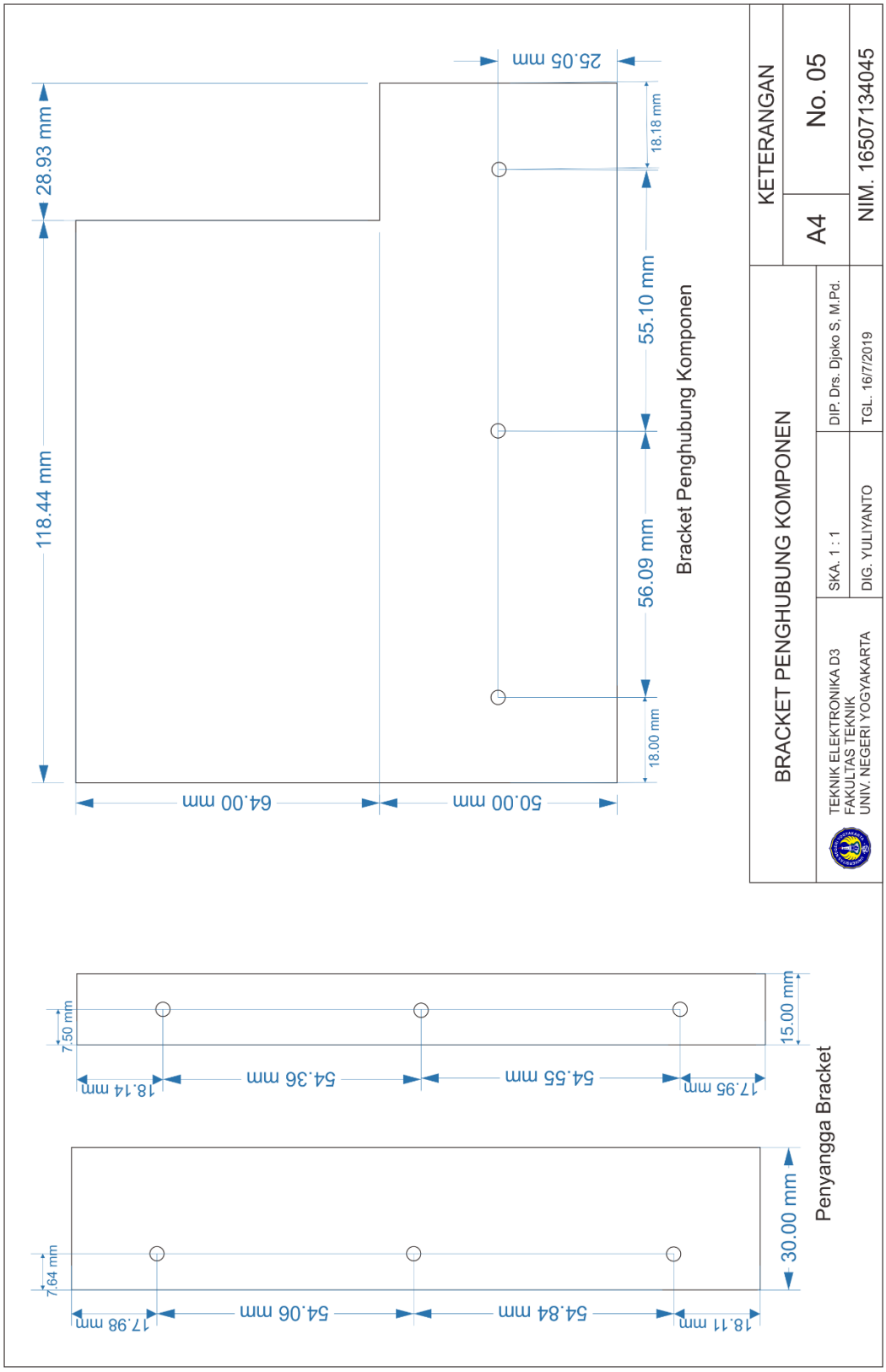
Lampiran 3. Desain Pembuatan Lubang Pada Black Box.



Lampiran 4. Desain Bracket Pemberi Pakan Ikan.



Lampiran 5. Desain Bracket Penghubung Komponen.



Lampiran 6. *Part List*

No.	Kebutuhan	Nama Komponen	Jumlah
1.	Kendali	Nodemcu ESP8266	1
2.	Pembuka dan penutup pakan	*Micro Servo SG90*	1
3.	*Driver* kendali	IC L293D	1
4.	Pompa air	*Brushless DC pump*	3
5.	Sensor suhu	LM35	1
6.	Pencacah waktu	RTC DS3231	1
7.	Pendingi air	*DC Brushless FAN*	1
8.	Pemanas air	*Aquarium Heater*	1
9.	Sensor ketinggian air	HC-SR04	1
11.	Tampilan	LCD 2x16	1
12.	*Switch* kontak	*Relay* 12VDC	1
13.	Sumber daya 12VDC	*Power Supply* 12VDC	1
14.	Sumber daya 5VDC	Modul *Power Supply Step Down 12VDC to 5VDC*	1

Lampiran 7. Petunjuk *(Manual)*

Cara pengoprasian alat:

1. Hubungkan stopkontak ke sumber tegangan.
2. Mengaktifkan *wifi* untuk koneksi *internet* Nodemcu.
3. Membuka aplikasi Blynk pada ponsel.
4. Mengisi pakan ikan pada tabung pakan.
5. Selesai.

Lampiran 8. Gambar Alat.

Lampiran 9. Program

```
#include <ESP8266WiFi.h>
#define BLYNK_PRINT Serial
#include <SimpleTimer.h>
#include <BlynkSimpleEsp8266.h>
#include <LiquidCrystal_I2C.h>

#include "Wire.h"
#include "uRTCLib.h"
// uRTCLib rtc;
uRTCLib rtc(0x68);

#include <Servo.h>
Servo myservo;
int pos = 00;

LiquidCrystal_I2C lcd(0x27, 16, 2);
BlynkTimer timer;

char auth[] = "ea518fe985a7487bafd07b56c27fe59b";
char ssid[] = "daginganjing";
char pass[] = "12344213";

// defines pins numbers
const int trigPin = 15;  //D1
const int echoPin = 13;  //D2
// defines variables
long duration;
int distance;

  int kuras = D3;
  int isi = D4;
  int fan = D5;
  int heater = D6;
```

```
  int LM_35 = A0;
  int input_val = 0;
  int temperatur;

void setup()
{
 Serial.begin(9600);
 Blynk.begin(auth, ssid, pass);
 timer.setInterval(1500L, suhu);
 timer.setInterval(1500L, jarak);

 myservo.attach(16); //D0

 lcd.begin(16,2);
 lcd.init();
 lcd.backlight();

 #ifdef ARDUINO_ARCH_ESP8266
   Wire.begin(4, 5); // D3 and D4 on ESP8266
 #else
   Wire.begin();
 #endif

 // rtc.set(0, 27, 18, 7, 6, 7, 19);
  //  RTCLib::set(byte second, byte minute, byte hour, byte dayOfWeek, byte dayOfMonth, byte month, byte
year)
}

 void loop()
{
  Blynk.run();
  timer.run();
 jam();
  suhu();
```

```
  jarak();
  servo();
  }

void jam()
{
rtc.refresh();
  Serial.print("RTC DateTime: ");
  Serial.print(rtc.year());
  Serial.print('/');
  Serial.print(rtc.month());
  Serial.print('/');
  Serial.print(rtc.day());

  Serial.print(' ');

  Serial.print(rtc.hour());
  Serial.print(':');
  Serial.print(rtc.minute());
  Serial.print(':');
  Serial.print(rtc.second());

  Serial.print(" DOW: ");
  Serial.print(rtc.dayOfWeek());

 Serial.print(" - Temp: ");
  Serial.print(rtc.temp());

  Serial.println();

  delay(1000);
      lcd.setCursor(0,0);
      lcd.print("Time:  ");
      lcd.print(rtc.hour(), DEC);
```

```
    lcd.print(':');
    lcd.print(rtc.minute(), DEC);
    lcd.print(':');
    lcd.print(rtc.second(), DEC);
 }

void suhu(){

  input_val = analogRead(LM_35);            // Reading analog input
  temperatur = (5.0 * input_val * 100.0) / 2024;   // Some calculation to convert analog value to temperature

    if(temperatur <= 26){
      digitalWrite(heater, HIGH);
      }
    else if(temperatur >=29){
      digitalWrite(fan, HIGH);
      }
    else if(temperatur == 27 || 28){
      digitalWrite(heater, LOW);
      digitalWrite(fan, LOW);
      }
  Serial.print("Temperature LM35 : " );
  Serial.print(temperatur);
  Serial.print("C" );
  Serial.println();
   lcd.setCursor(0,1);
   lcd.print("Tem: ");
   lcd.print(temperatur);
   lcd.print("C");
  Blynk.virtualWrite(V0, temperatur);
}

void jarak(){
pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
```

```
pinMode(echoPin, INPUT); // Sets the echoPin as an Input Serial.print("Temperature LM35 : " );


// Clears the trigPin
digitalWrite(trigPin, LOW);
delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
duration = pulseIn(echoPin, HIGH);

// Calculating the distance
distance= duration*0.034/2;
if(rtc.dayOfWeek() == 6 && rtc.hour() == 10){
    digitalWrite(kuras, HIGH);
    }
  else if(distance >= 16){
    digitalWrite(kuras, LOW);
    digitalWrite(isi, HIGH);
    }
  else if(distance <= 5){
    digitalWrite(isi, LOW);
    }

// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.print(distance);
Serial.println("cm" );
Serial.println();
```

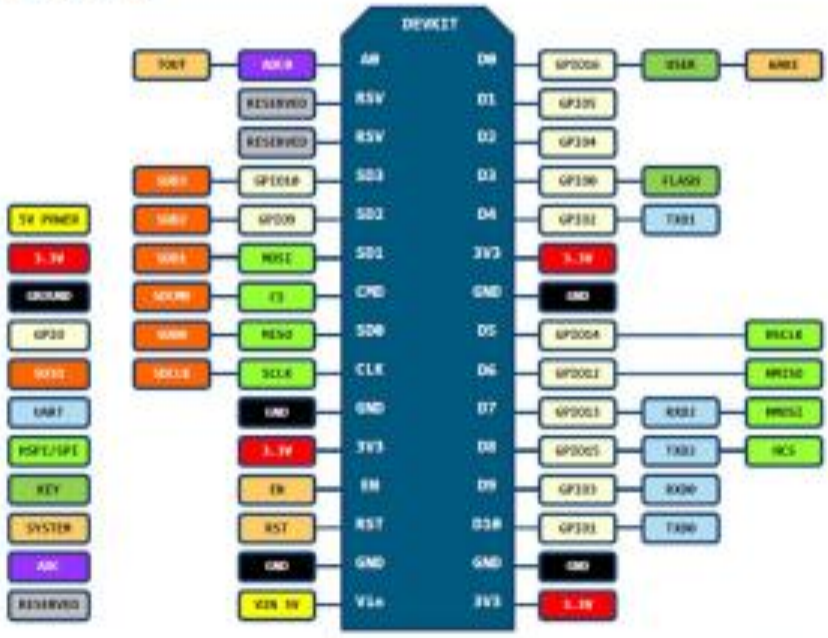
```
      lcd.setCursor(9,1);
      lcd.print("Dis:");
      lcd.print(distance);
      lcd.print("Cm");

  Blynk.virtualWrite(V1, distance);
}
void servo(){
    if(rtc.hour() == 6 || rtc.hour() == 17)
    {
        myservo.write(pos);
        delay(25);
        myservo.write(40);
        delay(25);
        myservo.write(pos);
        delay(25);
        myservo.write(40);
        delay(25);
        myservo.write(pos);
        delay(25);  }
     else {
        myservo.write(pos);
     }

  delay(1000);
}
```

Lampiran 10. Datasheet Nodemcu.

1. Specification:

- Voltage:3.3V.
- Wi-Fi Direct (P2P), soft-AP.
- Current consumption: 10uA~170mA.
- Flash memory attachable: 16MB max (512K normal).
- Integrated TCP/IP protocol stack.
- Processor: Tensilica L106 32-bit.
- Processor speed: 80~160MHz.
- RAM: 32K + 80K.
- GPIOs: 17 (multiplexed with other functions).
- Analog to Digital: 1 input with 1024 step resolution.
- +19.5dBm output power in 802.11b mode
- 802.11 support: b/g/n.
- Maximum concurrent TCP connections: 5.

2. Pin Definition:



D0(GPIO16) can only be used as gpio read/write, no interrupt supported, no pwm/I2C/one supported.

3. Using Arduino IDE

Lampiran 11. Datasheet DS3231.

DS3231

Extremely Accurate I²C-Integrated RTC/TCXO/Crystal

General Description

The DS3231 is a low-cost, extremely accurate I²C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The DS3231 is available in commercial and industrial temperature ranges, and is offered in a 16-pin, 300-mil SO package.

The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. Two programmable time-of-day alarms and a programmable square-wave output are provided. Address and data are transferred serially through an I²C bidirectional bus.

A precision temperature-compensated voltage reference and comparator circuit monitors the status of V_{CC} to detect power failures, to provide a reset output, and to automatically switch to the backup supply when necessary. Additionally, the $\overline{RST}$ pin is monitored as a pushbutton input for generating a µP reset.

Benefits and Features

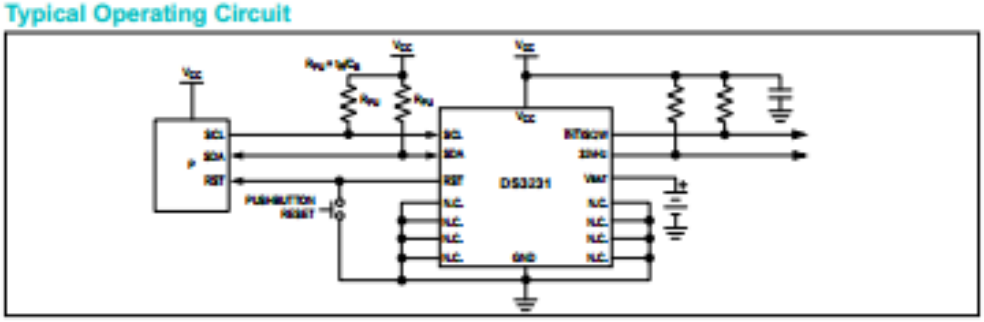
- Highly Accurate RTC Completely Manages All Timekeeping Functions
  - Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year, with Leap-Year Compensation Valid Up to 2100
  - Accuracy ±2ppm from 0°C to +40°C
  - Accuracy ±3.5ppm from -40°C to +85°C
  - Digital Temp Sensor Output: ±3°C Accuracy
  - Register for Aging Trim
  - $\overline{RST}$ Output/Pushbutton Reset Debounce Input
  - Two Time-of-Day Alarms
  - Programmable Square-Wave Output Signal
- Simple Serial Interface Connects to Most Microcontrollers
  - Fast (400kHz) I²C Interface
- Battery-Backup Input for Continuous Timekeeping
  - Low Power Operation Extends Battery-Backup Run Time
  - 3.3V Operation
- Operating Temperature Ranges: Commercial (0°C to +70°C) and Industrial (-40°C to +85°C)
- Underwriters Laboratories® (UL) Recognized

Applications

- Servers
- Telematics
- Utility Power Meters
- GPS

Ordering Information and Pin Configuration appear at end of data sheet.

Typical Operating Circuit



Underwriters Laboratories is a registered certification mark of Underwriters Laboratories Inc.

19-5170; Rev 10; 3/15

maxim integrated

Lampiran 12. Datasheet Servo.

Servo Motor

Servo Motor Pinout (Wires)

[Click the image to enlarge it]

Wire Configuration

Wire Number	Wire Colour	Description
1	Brown	Ground wire connected to the ground of system
2	Red	Powers the motor typically +5V is used
3	Orange	PWM signal is given in through this wire to drive the motor

TowerPro SG-90 Features

- Operating Voltage is +5V typically
- Torque: 2.5kg/cm
- Operating speed is 0.1s/60°
- Gear Type: Plastic
- Rotation : 0°-180°
- Weight of motor : 9gm
- Package includes gear horns and screws

SG-90 Servo Motor Equivalent

MG90S Metal Gear, MG995 High Torque Metal Gear, VTS-08A Analog Servo

Selecting your Servo Motor

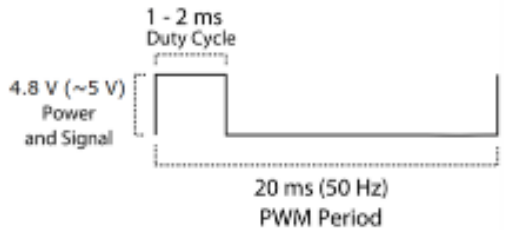
There are lots of servo motors available in the market and each one has its own speciality and applications. The following two paragraphs will help you identify the right type of servo motor for your project/system.

Most of the hobby Servo motors operates from 4.8V to 6.5V, the higher the voltage higher the torque we can achieve, but most commonly they are operated at +5V. Almost all hobby servo motors can rotate only from 0° to 180° due to their gear arrangement so make sure you project can live with the half circle if no, you can prefer for a 0° to 360° motor or modify the motor to make a full circle. The gears in the motors are easily subjected to wear and tear, so if your application requires stronger and long running motors you can go with metal gears or just stick with normal plastic gear.

Next comes the most important parameter, which is the **torque** at which the motor operates. Again there are many choices here but the commonly available one is the 2.5kg/cm torque which comes with the Towerpro SG90 Motor. This 2.5kg/cm torque means that the motor can pull a weight of 2.5kg when it is suspended at a distance of 1cm. So if you suspend the load at 0.5cm then the motor can pull a load of 5kg similarly if you suspend the load at 2cm then can pull only 1.25. Based on the load which you use in the project you can select the motor with proper torque. The below picture will illustrate the same.

How to use a Servo Motor

After selecting the right Servo motor for the project, comes the question how to use it. As we know there are three wires coming out of this motor. The description of the same is given on top of this page. To make this motor rotate, we have to power the motor with +5V using the Red and Brown wire and send PWM signals to the Orange colour wire. Hence we need something that could generate PWM signals to make this motor work, this something could be anything like a 555 Timer or other Microcontroller platforms like Arduino, PIC, ARM or even a microprocessor like Raspberry Pie. Now, how to control the direction of the motor? To understand that let us a look at the picture given in the datasheet.

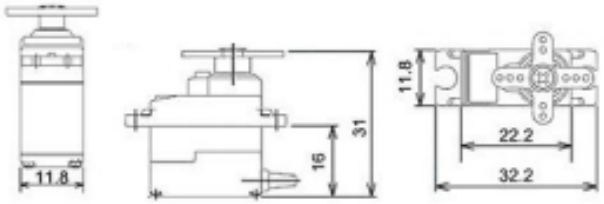


From the picture we can understand that the PWM signal produced should have a frequency of 50Hz that is the PWM period should be 20ms. Out of which the On-Time can vary from 1ms to 2ms. So when the on-time is 1ms the motor will be in 0° and when 1.5ms the motor will be 90°, similarly when it is 2ms it will be 180°. So, by varying the on-time from 1ms to 2ms the motor can be controlled from 0° to 180°

Applications

- Used as actuators in many robots like Biped Robot, Hexapod, robotic arm etc..
- Commonly used for steering system in RC toys
- Robots where position control is required without feedback
- Less weight hence used in multi DOF robots like humanoid robots

SG90 Servo Motor Dimensions



Lampiran 13. Datasheet LM35.

LM35 Datasheet

P. MARIAN

IC datasheet LM35

The LM35 datasheet specifies that this ICs are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature.
The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centi-grade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1/4°C at room temperature and ±3/4°C over a full −55 to +150°C temperature range.

LM35 datasheet

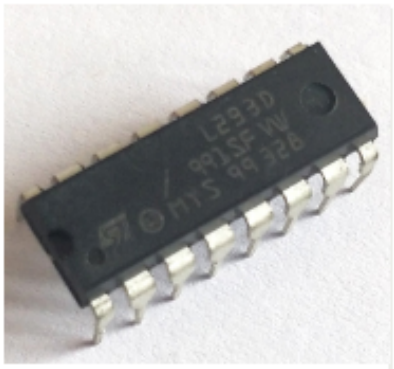
LM35 Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear + 10.0 mV/°C scale factor
- 0.5°C accuracy guaranteeable (at +25°C)
- Rated for full −55° to +150°C range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than 60 µA current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only ±1/4°C typical
- Low impedance output, 0.1 Ω for 1 mA load

Lampiran 14. Datasheet IC L293D.

L293D Motor Driver IC

8 October 2017 - 0 Comments



L293D Motor Driver IC | Motor Driver IC L293D Pinout

[Click the image to enlarge it]

L293D Pin Configuration

Pin Number	Pin Name	Description
1	Enable 1,2	This pin enables the input pin Input 1(2) and Input 2(7)
2	Input 1	Directly controls the Output 1 pin. Controlled by digital circuits
3	Output 1	Connected to one end of Motor 1
4	Ground	Ground pins are connected to ground of circuit (0V)
5	Ground	Ground pins are connected to ground of circuit (0V)
6	Output 2	Connected to another end of Motor 1
7	Input 2	Directly controls the Output 2 pin. Controlled by digital circuits
8	Vcc2 (Vs)	Connected to Voltage pin for running motors (4.5V to 36V)
9	Enable 3,4	This pin enables the input pin Input 3(10) and Input 4(15)
10	Input 3	Directly controls the Output 3 pin. Controlled by digital circuits
11	Output 3	Connected to one end of Motor 2
12	Ground	Ground pins are connected to ground of circuit (0V)
13	Ground	Ground pins are connected to ground of circuit (0V)
14	Output 4	Connected to another end of Motor 2
15	Input 4	Directly controls the Output 4 pin. Controlled by digital circuits
16	Vcc2 (Vss)	Connected to +5V to enable IC function

Lampiran 15. Datasheet HC-SR04.

HC-SR04 Datasheet

P. MARIAN

HC-SR04 is an ultrasonic ranging module that provides 2 cm to 400 cm non-contact measurement function. The ranging accuracy can reach to 3mm and effectual angle is < 15°. It can be powered from a 5V power supply.

Sources for downloading HC-SR04 pdf datasheet

source 1
source 2

HC-SR04 Specifications

- Working Voltage: DC 5V
- Working Current: 15mA
- Working Frequency: 40Hz
- Max Range: 4m
- Min Range: 2cm
- Measuring Angle: 15 degree
- Trigger Input Signal: 10µS TTL pulse
- Echo Output Signal Input TTL lever signal and the range in proportion
- Dimension 45 * 20 * 15mm