LAMPIRAN

Sistem Komunikasi dan Sensor pada Robot Humanoid *Semi-Auto*

```
#include <SoftwareSerial.h>
#include <Servo.h>
#include "SD.h"
#include "TMRpcm.h"
#include "SPI.h"
#include <DS3231.h> //mengincludekan library DS3231
#include "RTClib.h"
RTC_DS3231 rtc;
TMRpcm tmrpcm;
Servo servo1, servo2, servo3, servo4, servo5, servo6, servo7, servo8, servo9;
#define SD_ChipSelectPin 49
#define motorkanan1 4
#define motorkanan2 6
#define motorkiri1 8
#define motorkiri2 10
#define pwm 2
#define dirPin1 34
#define stepPin1 32
char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis",
"Jum'at", "Sabtu"};
int jam, menit, detik;
int tanggal, bulan, tahun; String hari;
int pinrelay = 33;
//MENDEFINISIKAN PIN ECHO DAN TRIGGER ULTRASONIC HC-SR04
#define echo 38
#define trig 36
//MENDEFINISIKAN VARIABEL distancerobot SEBAGAI INTEGER
int distancerobot;
//MENDEFINISIKAN VARIABEL duration SEBAGAI LONG
long duration;
int pos1;
int pos2;
int pos3;
int pos4;
int pos5;
int pos6;
int pos7;
int pos8;
int pos9;
int i; //kiri
int j; //kanan
int  komunikasi;  // Variabel Komunikasi yang bernilai Integer
void(* reset) (void) = 0;
```

```
void setup() {
  Serial.begin(9600);
  tmrpcm.speakerPin = 46;
  pinMode(stepPin1,    OUTPUT);
  pinMode(dirPin1,     OUTPUT);
  pinMode(motorkanan1, OUTPUT);
  pinMode(motorkiri1,  OUTPUT);
  pinMode(motorkanan2, OUTPUT);
  pinMode(motorkiri2,  OUTPUT);
  pinMode(pwm,         OUTPUT);
  pinMode(pinrelay,    OUTPUT);
  pinMode(trig,        OUTPUT);
  pinMode(echo,        INPUT);
  digitalWrite(pinrelay, HIGH);
  servo1.attach(9);    //siku kiri
  servo2.attach(11);   //bahu kanan
  servo3.attach(13);   //lengan kanan
  servo4.attach(15);   //rotasi lengan kanan
  servo5.attach(17);   //siku kanan
  servo6.attach(19);   //bahu kiri
  servo7.attach(23);   //lengan kiri
  servo8.attach(25);   //rotasi lengan kiri
  servo9.attach(27);   //cadangan
  servo1.write(90);
  servo2.write(90);
  servo3.write(90);
  servo4.write(90);
  servo5.write(90);
  servo6.write(90);
  servo7.write(90);
  servo8.write(90);
  servo9.write(90);
  if (! rtc.begin()) {
     Serial.println("Couldn't find RTC");
     }
  if (! rtc.begin()) {}
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
}
void loop() {
   /*data();
   Serial.println(distancerobot);
   if(distancerobot<=70){
     suara();
     tmrpcm.setVolume(5);
     tmrpcm.play("ada_orang.wav");
     delay(800);
```

```
   MUNDUR();
   delay(1000);
   reset();
  }*/
 //BLUETOOTH MENGIRIMKAN KARAKTER HURUF
 if(Serial.available() > 0){
  komunikasi = Serial.read();
 if(komunikasi >10){}}
  if (komunikasi == 1){   //JIKA KARAKTER YANG DITERIMA =
KARAKTER L, MAKA ROBOT AKAN MAJU
       data();
       Serial.println(distancerobot);
       MAJU();
       adaorang();}
  if (komunikasi == 2){   //JIKA KARAKTER YANG DITERIMA =
KARAKTER D, MAKA ROBOT AKAN MUNDUR
       data();
       Serial.println(distancerobot);
       MUNDUR();}
  if (komunikasi == 3){   //JIKA KARAKTER YANG DITERIMA =
KARAKTER R, MAKA ROBOT AKAN BELOK KANAN
       data();
       Serial.println(distancerobot);
       KANAN();}
  if (komunikasi == 4){   //JIKA KARAKTER YANG DITERIMA =
KARAKTER L, MAKA ROBOT AKAN BELOK KIRI
       data();
       Serial.println(distancerobot);
       KIRI();}
  if (komunikasi == 0){   //JIKA KARAKTER YANG DITERIMA =
KARAKTER S, MAKA ROBOT AKAN DIAM (BERHENTI)
       STOP();}
  if (komunikasi == 5){
   gerakan1();
   suara();
   tmrpcm.setVolume(5);
   tmrpcm.play("PAGI.wav");
   delay(3000);
   reset();}
  if (komunikasi == 6){
   gerakan2();
   suara();
   tmrpcm.setVolume(5);
   tmrpcm.play("SIANG.wav");
   delay(3000);
   reset();}
```

```
if (komunikasi == 7){
  gerakan3();
  suara();
  tmrpcm.setVolume(5);
  tmrpcm.play("DATANG.wav");
  delay(3500);
  reset();}
if (komunikasi == 8){
  gerakan4();
  suara();
  tmrpcm.setVolume(5);
  tmrpcm.play("MAKASIH.wav");
  delay(3500);
  reset(); }
if (komunikasi == 9){
  gerakan5();
  suara();
  tmrpcm.setVolume(5);
  tmrpcm.play("SORE.wav");
  delay(3800);
  reset();}
if (komunikasi == 10){
  gerakan6();}
if (komunikasi == 11){
  gerakan7();
  reset();}
if (komunikasi == 12){
}
if(komunikasi == 13){   // Halo siapa nama kamu?
  gerakan9();
  suara();
  tmrpcm.setVolume(5);
  tmrpcm.play("SALKEN.wav");
  delay(2000);
  reset();}

if(komunikasi == 14){   //Halo Jamal
  gerakan10();
  suara();
  tmrpcm.setVolume(5);
  tmrpcm.play("ADSB.wav");
  delay(3000);
  reset();}
if(komunikasi == 15){   //Hai Jamal apa kabar?
  gerakan11();
  suara();
```

```
      tmrpcm.setVolume(5);
      tmrpcm.play("APAKABAR.wav");
      delay(2000);
      reset();}
    if(komunikasi == 16){   //Hai Jamal ini hari apa?
      gerakan12();
      RTC();
      suara();
      tmrpcm.setVolume(5);
      tmrpcm.play("hariini.wav");
      delay(1500);
      hariini();
      reset();}
delay(50);}
void MAJU(){
    digitalWrite(motorkanan1, LOW);
    digitalWrite(motorkiri1, LOW);
    digitalWrite(motorkanan2, HIGH);
    digitalWrite(motorkiri2, HIGH);
    analogWrite(pwm,120); }
void MUNDUR(){
    digitalWrite(motorkiri2, LOW);
    digitalWrite(motorkanan2, LOW);
    digitalWrite(motorkanan1, HIGH);
    digitalWrite(motorkiri1, HIGH);
    analogWrite(pwm,90);}
void KANAN(){
    digitalWrite(motorkanan1, LOW);
    digitalWrite(motorkiri1, HIGH);
    digitalWrite(motorkanan2, HIGH);
    digitalWrite(motorkiri2, LOW);
    analogWrite(pwm,120);}
void KIRI(){
    digitalWrite(motorkanan1, HIGH);
    digitalWrite(motorkiri1, LOW);
    digitalWrite(motorkanan2, LOW);
    digitalWrite(motorkiri2, HIGH);
    analogWrite(pwm,120); }
void STOP(){
    digitalWrite(motorkanan1, LOW);
    digitalWrite(motorkanan2, LOW);
    digitalWrite(motorkiri1, LOW);
    digitalWrite(motorkiri2, LOW);
    analogWrite(pwm,0);}
void gerakan1(){        //silakan tangan kanan selamat pagi
  for(pos6=90,pos7=90,i=60;i>=0;pos6++,pos7++,i--){
```

```
    servo6.write(pos6);
    servo7.write(pos7);
    delay(13);}
  delay(1500);
  for(pos6=150,pos7=150,i=0;i<=60;pos6--,pos7--,i++){
    servo6.write(pos6);
    servo7.write(pos7);
    delay(13); }}
void gerakan2(){        //silakan tangan kiri selamat siang
  for(pos2=90,pos3=90,j=0;j<=60;pos2--,pos3--,j++){
    servo2.write(pos2);
    servo3.write(pos3);
    delay(13);}
  delay(1500);
  for(pos2=30,pos3=30,j=60;j>=0;pos2++,pos3++,j--){
    servo2.write(pos2);
    servo3.write(pos3);
    delay(13);}}
void gerakan3(){        //silakan tangan kanan kiri selamat datang
  for(pos2=90,pos3=90,pos6=90,pos7=90,i=60,j=0;i>=0,j<=60;pos2--,pos3--
,pos6++,pos7++,i--,j++){
    servo2.write(pos2);
    servo3.write(pos3);
    servo6.write(pos6);
    servo7.write(pos7);
    delay(13);}
  delay(2000);
  for(pos2=30,pos3=30,pos6=150,pos7=150,i=0,j=60;i<=60,j>=0;pos2++,pos3++,
pos6--,pos7--,i++,j--){
    servo2.write(pos2);
    servo3.write(pos3);
    servo6.write(pos6);
    servo7.write(pos7);
    delay(13);} }
void gerakan4(){        //silakan tangan kanan kiri Terima kasih
  for(pos2=90,pos3=90,pos6=90,pos7=90,i=60,j=0;i>=0,j<=60;pos2--,pos3--
,pos6++,pos7++,i--,j++){
    servo2.write(pos2);
    servo3.write(pos3);
    servo6.write(pos6);
    servo7.write(pos7);
    delay(13);}
  delay(2000);
  for(pos2=30,pos3=30,pos6=150,pos7=150,i=0,j=60;i<=60,j>=0;pos2++,pos3++,
pos6--,pos7--,i++,j--){
    servo2.write(pos2);
```

```
    servo3.write(pos3);
    servo6.write(pos6);
    servo7.write(pos7);
    delay(13);} }
void gerakan5(){        //selamat sore tangan kiri melintang
  for(pos1=90,pos3=90,j=0;j<=90;pos1--,pos3--,j++){
    servo1.write(pos1);
    servo3.write(pos3);
    delay(13);}
  delay(2000);
  for(pos1=0,pos3=0,j=90;j>=0;pos1++,pos3++,j--){
    servo1.write(pos1);
    servo3.write(pos3);
    delay(13);}}
void gerakan6(){
  for(pos1=90,pos3=90,pos4=90;pos1>=30,pos3<=180,pos4<=180;pos1--
,pos3++,pos4++){
    servo1.write(pos1);
    servo3.write(pos3);
    servo4.write(pos4);
    delay(13);}}
void gerakan7(){
  for(pos3=180,pos4=180,pos1=30;pos1<=90,pos3>=90,pos4>=90;pos1++,pos3--
,pos4--){
    servo3.write(pos3);
    servo4.write(pos4);
    servo1.write(pos1);
    delay(13);}   }
void gerakan8(){}
void gerakan9(){        //memberi salam tangan kanan siapa nama kamu
  for(pos6=90,i=60;i>=0;pos6++,i--){
    servo6.write(pos6);
    delay(15);}
  delay(1500);
  for(pos8=90;pos8>=60;pos8--){
    servo8.write(pos8);
    delay(15);}
  for(pos8=60;pos8<=90;pos8++){
    servo8.write(pos8);
    delay(15);}
  delay(1000);
  for(pos6=150,i=0;i<=60;pos6--,i++){
    servo6.write(pos6);
    delay(15);}}
void gerakan10(){       //gerakan hormat tangan kiri halo jamal
  for(pos1=90;pos1>=60;pos1--){
```

```
    servo1.write(pos1);
   delay(15);}
 for(pos2=90,pos4=90;pos2>=0,pos4<=180;pos2--,pos4++){
   servo2.write(pos2);
   servo4.write(pos4);
   delay(15);}
 delay(2000);
 for(pos2=0,pos4=180;pos2<=90,pos4>=90;pos2++,pos4--){
   servo2.write(pos2);
   servo4.write(pos4);
   delay(15);}
 for(pos1=60;pos1<=90;pos1++){
   servo1.write(pos1);
   delay(15);}}
void gerakan11(){       //gerakan menyapa tangan kanan hai jamal apa kabar
 for(pos5=90,pos6=90;pos5<=150,pos6<=150;pos5++,pos6++){
   servo5.write(pos5);
   servo6.write(pos6);
   delay(10);}
 for(pos7=90,pos8=90;pos7<=180,pos8>=0;pos7++,pos8--){
   servo7.write(pos7);
   servo8.write(pos8);
   delay(10);}
 delay(800);
             for(pos7=180;pos7>=120;pos7--){
              servo7.write(pos7);
              delay(8);}
             for(pos7=120;pos7<=180;pos7++){
              servo7.write(pos7);
              delay(8);}
             for(pos7=180;pos7>=120;pos7--){
              servo7.write(pos7);
              delay(8);}
             for(pos7=120;pos7<=180;pos7++){
              servo7.write(pos7);
              delay(8);}
 delay(1000);
  for(pos7=180,pos8=0;pos7>=90,pos8<=90;pos7--,pos8++){
   servo7.write(pos7);
   servo8.write(pos8);
   delay(10);}
 for(pos5=150,pos6=150;pos5>=90,pos6>=90;pos5--,pos6--){
   servo5.write(pos5);
   servo6.write(pos6);
   delay(10);} }
void gerakan12(){       //tangan kanan menekuk hai jamal ini hari apa
```

A-8

```arduino
  for(pos1=90;pos1>=60;pos1--){
   servo1.write(pos1);
   delay(15);}
  for(pos2=90,pos4=90;pos2>=0,pos4<=180;pos2--,pos4++){
   servo2.write(pos2);
   servo4.write(pos4);
   delay(15);}
  delay(2000);
   for(pos2=0,pos4=180;pos2<=90,pos4>=90;pos2++,pos4--){
   servo2.write(pos2);
   servo4.write(pos4);
   delay(15);}
  for(pos1=60;pos1<=90;pos1++){
   servo1.write(pos1);
   delay(15);}  }
  /*digitalWrite(dirPin1, LOW);
 // Spin the stepper motor 1 revolution slowly:
  for (int i = 0; i < 30; i++) {
   // These four lines result in 1 step:
   digitalWrite(stepPin1, HIGH);
   delayMicroseconds(15000);
   digitalWrite(stepPin1, LOW);
   delayMicroseconds(15000);
  }*/
void suara(){
   if(!SD.begin(SD_ChipSelectPin)){
   Serial.println("SD fail");} }
void RTC(){
  DateTime now = rtc.now();
  jam       = now.hour();
  menit      = now.minute();
  detik      = now.second();
  tanggal     = now.day();
  bulan      = now.month();
  tahun      = now.year();
  hari       = daysOfTheWeek[now.dayOfTheWeek()];
  Serial.println(String() + hari);}
void hariini(){
  if (hari == "Senin"){
   suara();
   tmrpcm.setVolume(5);
   tmrpcm.play("senin.wav");
   delay(1500);}
  if (hari == "Selasa"){
   suara();
   tmrpcm.setVolume(5);
```

```
    tmrpcm.play("selasa.wav");
    delay(1500);}
  if (hari == "Rabu"){
    suara();
    tmrpcm.setVolume(5);
    tmrpcm.play("rabu.wav");
    delay(1500);}
  if (hari == "Kamis"){
    suara();
    tmrpcm.setVolume(5);
    tmrpcm.play("kamis.wav");
    delay(1500);}
  if (hari == "Jum'at"){
    suara();
    tmrpcm.setVolume(5);
    tmrpcm.play("jumat.wav");
    delay(1500);}
  if (hari == "Sabtu"){
    suara();
    tmrpcm.setVolume(5);
    tmrpcm.play("sabtu.wav");
    delay(1500);}
  if (hari == "Minggu"){
    suara();
    tmrpcm.setVolume(5);
    tmrpcm.play("minggu.wav");
    delay(1500);}}
long data(){
    digitalWrite(trig, LOW);
    delayMicroseconds(2);
    digitalWrite(trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(trig, LOW);
    duration = pulseIn(echo, HIGH);
    distancerobot = duration * 0.0340 / 2;}
void adaorang(){
  if (distancerobot < 60){
    STOP();
    gerakan1();
    suara();
    tmrpcm.setVolume(5);
    tmrpcm.play("GANTENG.wav");
    delay(3000);
    reset();}}
```

Lampiran 2. *Software MIT App Inventor* dan *Blocks* Program.
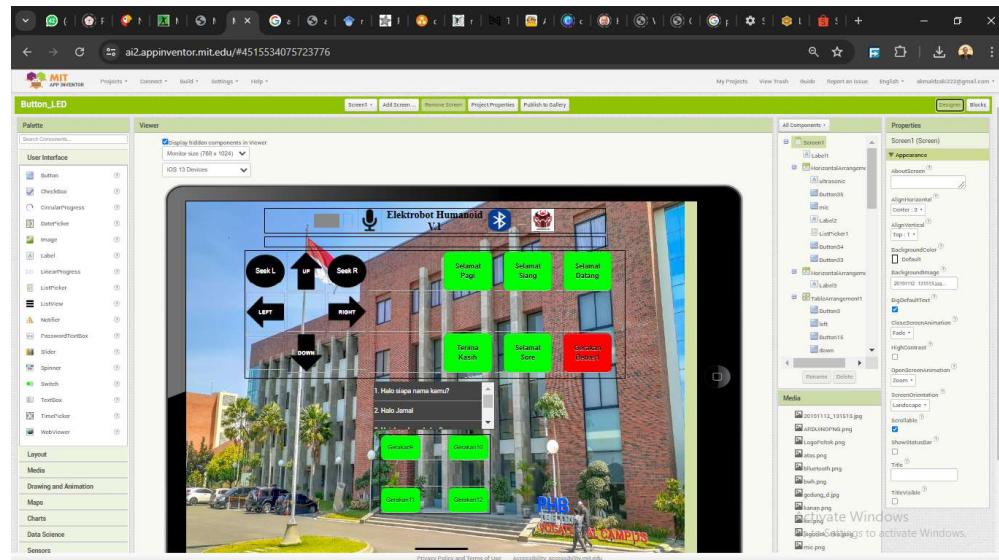
Tampilan *Software MIT App Inventor*.
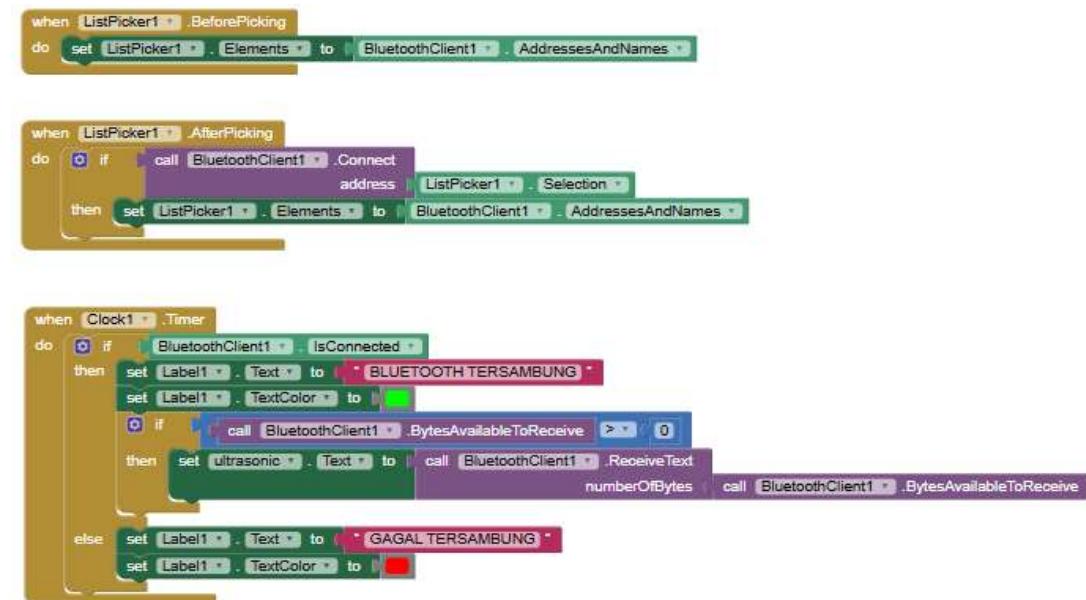
Block Konektivitas *Bluetooth*.

Block Robot Bergerak.

Block Robot Menyapa.
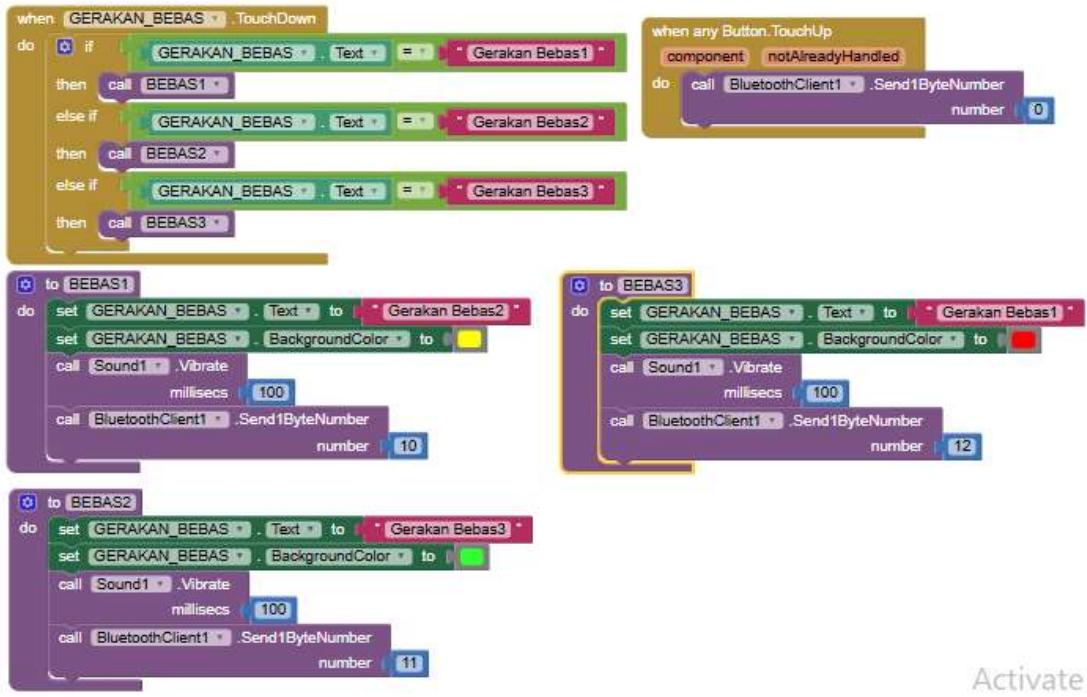
Block Gerakan Bebas.
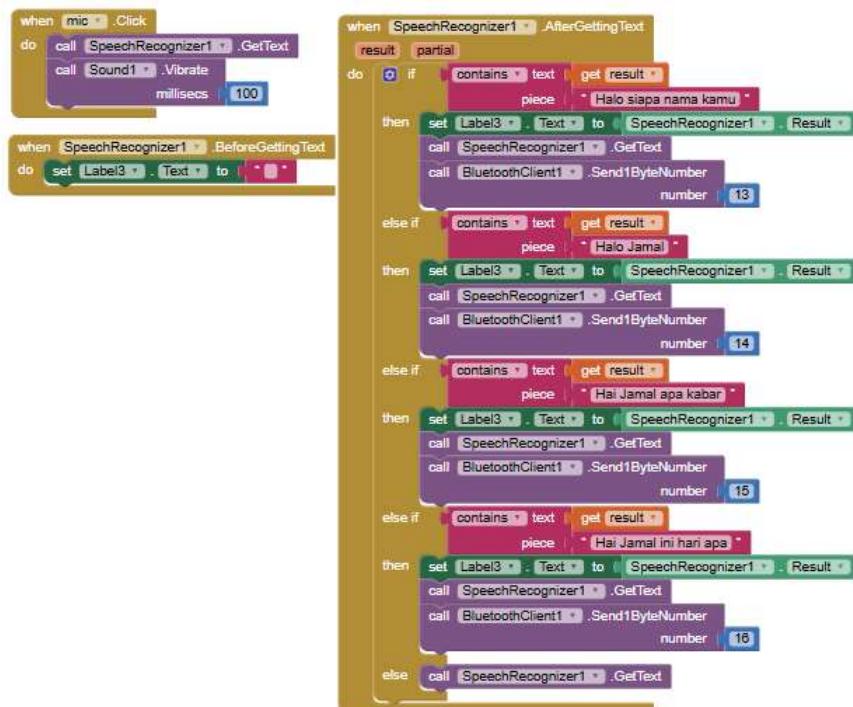
Block *Speech Recognition.*

A-13

A-15

1. *Datasheet Arduino Mega 2560 Pro*

EnM Industry
www.enmindustry.de
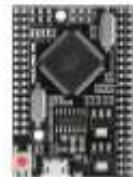
Mega2650PRO Datasheet

Mega 2650 PRO mini 5V ATMEGA 2650-16AU Development Board

- Embedded version of Mega 2560 CH340G/ATmega2560 compatible with Arduino Mega 2560 board, built on Atmel ATmega2560 microcontroller and USB-UART interface chip CH340G.
- The board has a compact size of 38x55mm. It is a good solution to make the final project on the prototype board.
- The functionality of the board is similar to the Arduino Mega2560. It is an embedded board and uses the original chip of ATmega2560 (16MHz).
- The board uses the chip CH340G as a UART-USB converter. Its function in the frequency 12MHz, leads to a stable data exchange (the driver must be installed).
- Mega PRO (Embedded) 2560 CH340G/ATmega 2560 can be connected to the computer through micro USB cable.
- Power supply of the board is via the microUSB connector or via pin headers. The voltage regulator (LDO) can handle an input voltage of 6 V to 9 V (peak 18 V) DC. The output current is over 800 mA at 5 V (please note that the higher the input voltage, the lower the output current). This will provide reliable performance in the initial projects.

Microcontroller	ATmega 2560
USB-TTL converter	CH340
Power Out	5V-800mA
Power In	5V
Power In VIN/DC Jack	5V
Power Consumption	5V 220mA
Logic Level	5V

1

2. *Datasheet Bluetooth HC-05 Modul*

HC-05

-Bluetooth to Serial Port Module

Overview

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

Specifications

Hardware features

- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

3. *Datasheet Micro SD Card Modul*

Micro SD Card Module for Arduino

The module (Micro-SD Card Adapter) is a Micro SD card reader module, and the SPI interface via the file system driver, microcontroller system to complete the Micro-SD card read and write files. Arduino users can directly use the Arduino IDE comes with an SD card to complete the library card initialization and read-write.

Features:

- Supports micro SD card (<=2G), micro SDHC card (<=32G) (high-speed card)
- Level conversion circuit board that can interface level is 5V or 3.3V
- Power supply is 4.5V ~ 5.5V, 3.3V voltage regulator circuit board
- Communication interface is a standard SPI interface
- 4 M2 screw positioning holes for easy installation
- Size: 4.1 x 2.4cm

Control Interface:

A total of six pins (GND, VCC, MISO, MOSI, SCK, CS), GND to ground, VCC is the power supply, MISO, MOSI, SCK is the SPI bus, CS is the chip select signal pin.

3.3V regulator circuit:

LDO regulator output 3.3V as level converter chip, Micro SD card supply.

4. *Datasheet Motor Servo MG996R*

MG996R High Torque
Metal Gear Dual Ball Bearing Servo
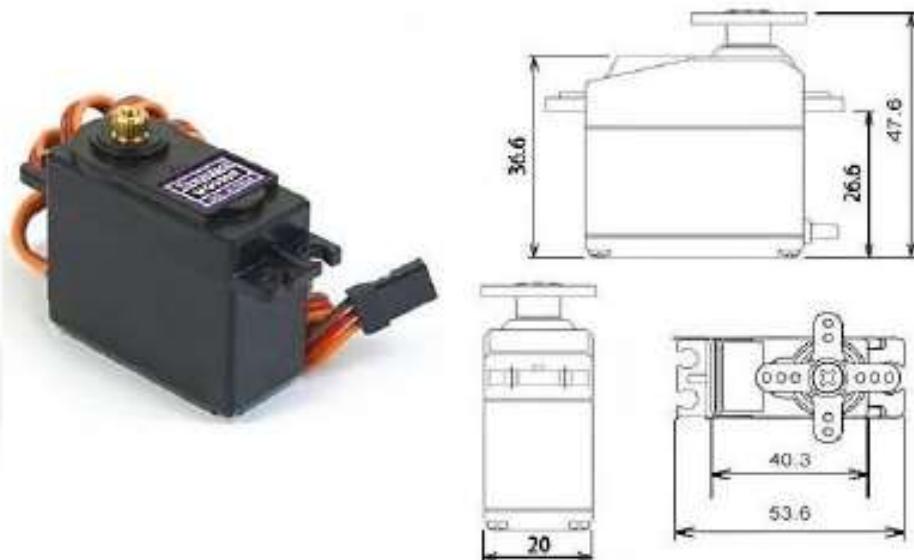
This High-Torque MG996R Digital Servo features metal gearing resulting in extra high 10kg stalling torque in a tiny package. The MG996R is essentially an upgraded version of the famous MG995 servo, and features upgraded shock-proofing and a redesigned PCB and IC control system that make it much more accurate than its predecessor. The gearing and motor have also been upgraded to improve dead bandwith and centering. The unit comes complete with 30cm wire and 3 pin 'S' type female header connector that fits most receivers, including Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum and Hitec.

This high-torque standard servo can rotate approximately 120 degrees (60 in each direction). You can use any servo code, hardware or library to control these servos, so it's great for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. The MG996R Metal Gear Servo also comes with a selection of arms and hardware to get you set up nice and fast!

Specifications

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 9.4 kgf-cm (4.8 V), 11 kgf-cm (6 V)
- Operating speed: 0.17 s/60° (4.8 V), 0.14 s/60° (6 V)

DS3231

Extremely Accurate I^{2}C-Integrated
RTC/TCXO/Crystal

Pin Description

PIN	NAME	FUNCTION
1	32kHz	32kHz Output. This open-drain pin requires an external pullup resistor. When enabled, the output operates on either power supply. It may be left open if not used.
2	V$_{CC}$	DC Power Pin for Primary Power Supply. This pin should be decoupled using a 0.1µF to 1.0µF capacitor. If not used, connect to ground.
3	INT/SQW	Active-Low Interrupt or Square-Wave Output. This open-drain pin requires an external pullup resistor connected to a supply at 5.5V or less. This multifunction pin is determined by the state of the INTCN bit in the Control Register (0Eh). When INTCN is set to logic 0, this pin outputs a square wave and its frequency is determined by RS2 and RS1 bits. When INTCN is set to logic 1, then a match between the timekeeping registers and either of the alarm registers activates the INT/SQW pin (if the alarm is enabled). Because the INTCN bit is set to logic 1 when power is first applied, the pin defaults to an interrupt output with alarms disabled. The pullup voltage can be up to 5.5V, regardless of the voltage on V$_{CC}$. If not used, this pin can be left unconnected.
4	RST	Active-Low Reset. This pin is an open-drain input/output. It indicates the status of V$_{CC}$ relative to the V$_{PF}$ specification. As V$_{CC}$ falls below V$_{PF}$, the RST pin is driven low. When V$_{CC}$ exceeds V$_{PF}$, for t$_{RST}$, the RST pin is pulled high by the internal pullup resistor. The active-low, open-drain output is combined with a debounced pushbutton input function. This pin can be activated by a pushbutton reset request. It has an internal 50kΩ nominal value pullup resistor to V$_{CC}$. No external pullup resistors should be connected. If the oscillator is disabled, t$_{REC}$ is bypassed and RST immediately goes high.
5–12	N.C.	No Connection. Must be connected to ground.
13	GND	Ground
14	V$_{BAT}$	Backup Power-Supply Input. When using the device with the V$_{BAT}$ input as the primary power source, this pin should be decoupled using a 0.1µF to 1.0µF low-leakage capacitor. When using the device with the V$_{BAT}$ input as the backup power source, the capacitor is not required. If V$_{BAT}$ is not used, connect to ground. The device is UL recognized to ensure against reverse charging when used with a primary lithium battery. Go to www.maximintegrated.com/qa/info/ul.
15	SDA	Serial Data Input/Output. This pin is the data input/output for the I^{2}C serial interface. This open-drain pin requires an external pullup resistor. The pullup voltage can be up to 5.5V, regardless of the voltage on V$_{CC}$.
16	SCL	Serial Clock Input. This pin is the clock input for the I^{2}C serial interface and is used to synchronize data movement on the serial interface. Up to 5.5V can be used for this pin, regardless of the voltage on V$_{CC}$.

Detailed Description

The DS3231 is a serial RTC driven by a temperature-compensated 32kHz crystal oscillator. The TCXO provides a stable and accurate reference clock, and maintains the RTC to within ±2 minutes per year accuracy from -40°C to +85°C. The TCXO frequency output is available at the 32kHz pin. The RTC is a low-power clock/calendar with two programmable time-of-day alarms and a programmable square-wave output. The INT/SQW provides either an interrupt signal due to alarm conditions or a square-wave output. The clock/calendar provides 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. The internal registers are accessible though an I^{2}C bus interface.

A temperature-compensated voltage reference and comparator circuit monitors the level of V$_{CC}$ to detect power failures and to automatically switch to the backup supply when necessary. The RST pin provides an external pushbutton function and acts as an indicator of a power-fail event.

Operation

The block diagram shows the main elements of the DS3231. The eight blocks can be grouped into four functional groups: TCXO, power control, pushbutton function, and RTC. Their operations are described separately in the following sections.

6. *Datasheet Modul Voice Recognition V-3*
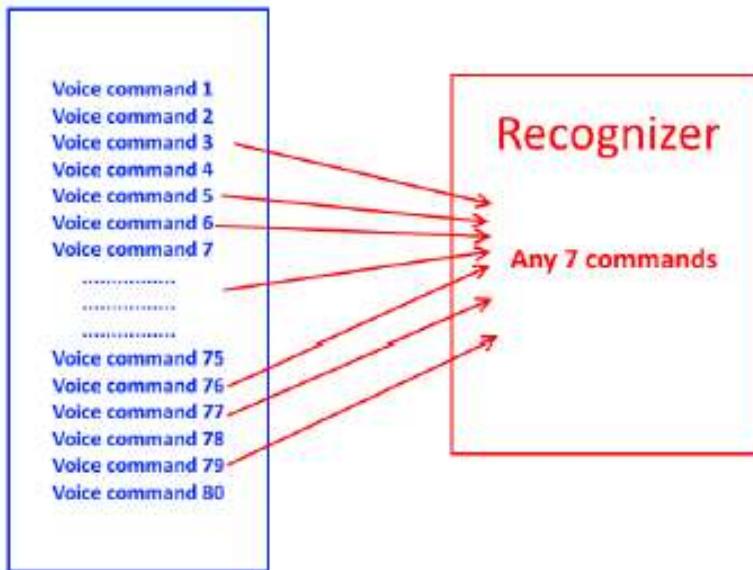
Parameter

- Voltage: 4.5-5.5V
- Current: <40mA
- Digital Interface: 5V TTL level for UART interface and GPIO
- Analog Interface: 3.5mm mono-channel microphone connector + microphone pin interface
- Size: 31mm x 50mm
- Recognition accuracy: 99% (under ideal environment)

Feature

- Support maximum 80 voice commands, with each voice 1500ms (one or two words speaking)
- Maximum 7 voice commands effective at same time
- Arduino library is supplied
- Easy Control: UART/GPIO
- User-control General Pin Output

Terminology

- VR3 -- Voice Recognition Module V3
- Recognizer -- a container where acting voice commands (max 7) were loaded. It is core part of voice recognition module. For example, it works like "playing balls". You have 80 players in your team. But you could not let them all play on the court together. The rule only allows 7 players playing on the court. Here the Recognizer is the list which contains names of players working on the court.
- Recognizer index -- max 7 voice commands could be supported in the recognizer. The recognizer has 7 regions for each voice command. One index corresponds to one region: 0~6
- Train -- the process of recording your voice commands
- Load -- copy trained voice to recognizer

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Bahrun Niam, M.T

NIPY. : 09.015.277

Jabatan : Sek. Prodi DIII Teknik Elektronika

Dengan ini menyatakan bersedia untuk menjadi Pembimbing Tugas Akhir mahasiswa berikut :

Nama : Muhammad Sholeh Akmal Nur Dzaki

NIM : 21010009

Program Studi : DIII Teknik Elektronika

Judul Laporan Tugas Akhir : SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID *SEMI-AUTO*

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 12 Juli 2024

Mengetahui,

Ka, Prodi DIII Teknik Elektronika

Calon Dosen Pembimbing I

Rony Darpono, M.T

NIPY 09.015.282

Bahrun Niam, M.T

NIPY. 09.015.277

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama   : Qirom, S.Pd, M.T

NIPY.   : 09.015.277

Jabatan   : Dosen Tetap

Dengan ini menyatakan bersedia untuk menjadi Pembimbing Tugas Akhir mahasiswa berikut :

Nama   : Muhammad Sholeh Akmal Nur Dzaki

NIM   : 21010009

Program Studi   : DIII Teknik Elektronika

Judul Laporan Tugas Akhir : SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID *SEMI-AUTO*

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 12 Juli 2024

Mengetahui,

Ka, Prodi DIII Teknik Elektronika
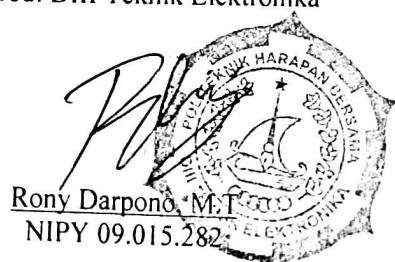
Calon Dosen Pembimbing II

Rony Darpono, M.T
NIPY 09.015.282

Qirom, S.Pd, M.T
NIPY. 09.015.281

A-23

FORM BIMBINGAN TUGAS AKHIR

Nama   : Muhammad Sholeh Akmal Nur Dzaki
NIM   : 21010009
JUDUL TA   : **SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID** *SEMI - AUTO*

Pembimbing 1

No	Hari/Tanggal	Uraian	Tanda Tangan
1	12/6 2024	Perubahan gambar skema komponen	
2.	24/06 2024	Bab I : latar bela kang,	
3	25/06 2024	Bab II : Tambahkan gambar roda	
4	02/07 2024	Bab III : Flow chat	
5	05/07 2024	Bab IV : Pembahasan.	
6.	7/7 2024	Flo chat Ok.	
7	11/7 2024	Bab IV : Hasil dan Pembahasan.	

FORM BIMBINGAN TUGAS AKHIR

Nama   : Muhammad Sholeh Akmal Nur Dzaki
NIM   : 21010009
JUDUL TA   : **SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID** *SEMI - AUTO*

Pembimbing 2

No	Hari/Tanggal	Uraian	Tanda Tangan
1.	12/6 2024	Bimbingan Judul	
2.	24/6 2024	Revisi Bab I 1) Latar Belakang 2) Permasalahan & Tujuan	
3.	25/6 2024	Bimbingan Online Program Robot Humanoid	
4.	02/07 2024	Bimbingan Bab IV Kerangka isi pembahasan	
5.	05/07 2024	Bimbingan Bab I Bab II Bab III Bab IV	
6.	11/07 2024	Perbaikan flowchart Hasil penelitian	
7.	12/07 2024	Acc Bab IV	
8.	12/07 2024	Acc Draft TA	

PENILAIAN BIMBINGAN TUGAS AKHIR INDIVIDU

Judul Tugas Akhir : SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID *SEMI-AUTO*

Nama : Muhammad Sholeh Akmal Nur Dzaki

Nim : 21010009

Kelas : 6A

I. Nilai Bimbingan Tugas Akhir (Pembimbing I)

NO	Unsur Yang Dinilai	Nilai
1.	Kedisiplinan Dalam Bimbingan	90
2.	Kreatifitas Pemecahan Dalam Bimbingan	95
3.	Penguasaan Materi Tugas Akhir	95
4.	Kelengkapan Dan Referensi Tugas Akhir	90
Total Nilai $=\left(\frac{jumlah\ nilai}{4}\right)$		92,5

II. Nilai Bimbingan Tugas Akhir (Pembimbing II)

NO	Unsur yang dinilai	Nilai
1.	Kedisiplinan Dalam Bimbingan	80
2.	Kreatifitas Pemecahan Dalam Bimbingan	80
3.	Penguasaan Materi Tugas Akhir	80
4.	Kelengkapan Dan Referensi Tugas Akhir	80
Total Nilai $=\left(\frac{jumlah\ nilai}{4}\right)$		80

Nilai Bimbingan $= \dfrac{Total\ nilai\ pembimbing\ 1 + Total\ nilai\ pembimbing\ 2}{2}$

$= 86,25$

Tegal, 12 Juli 2024

Mengetahui,

Pembimbing I,

Bahrun Niam, M.T
NIPY. 09.015.277

Pembimbing II,

Qirom, S.Pd, M.T
NIPY. 09.015.281

A-26

Turnitin

ORIGINALITY REPORT

21% 20% 6% 10%
SIMILARITY INDEX   INTERNET SOURCES   PUBLICATIONS   STUDENT PAPERS

PRIMARY SOURCES

1	eprints.poltektegal.ac.id Internet Source	5%
2	eprints.ums.ac.id Internet Source	1%
3	repository.its.ac.id Internet Source	1%
4	www.sensingthecity.com Internet Source	1%
5	repository.ub.ac.id Internet Source	1%
6	Submitted to Universitas Negeri Surabaya The State University of Surabaya Student Paper	<1%
7	eprints.polsri.ac.id Internet Source	<1%
8	widuri.raharja.info Internet Source	<1%
9	Submitted to Université Virtuelle de Tunis Student Paper	<1%

1. Ketua Penguji

FROM REVISI

UJIAN TUGAS AKHIR

NAMA	:	MUHAMMAD SHOLEH AKMAL NUR DZAKI
NIM	:	21010009
JUDUK TA	:	SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID *SEMI-AUTO*

Ketua Penguji

No	Hari/tanggal	Uraian	Tanda tangan
1	~~Se~~ kamis, 15-8-2024	Ace Projek & Laporan TA	*(tanda tangan)*
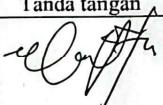
Ketua Penguji
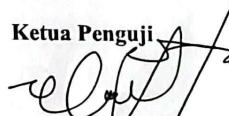
(tanda tangan)

Ubdul Akbar, M.T

2. Penguji 1

FROM REVISI

UJIAN TUGAS AKHIR

NAMA : MUHAMMAD SHOLEH AKMAL NUR DZAKI
NIM : 21010009
JUDUK TA : SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID *SEMI-AUTO*

Penguji 1

No	Hari/tanggal	Uraian	Tanda tangan
1.	Rabu 30 - 07 - 2024	Revisi Flowchart	
2.	Senin 5 - 08 - 2024	Revisi Bab 5	
3.	Senin 12 - 08 - 2024	-Acc Flowchart -Acc Bab 5	
4.	Selasa 20 - 08 - 2024	Acc	

Penguji 1

Rony Darpono M -T

3. Penguji 2

FROM REVISI

UJIAN TUGAS AKHIR

NAMA	: MUHAMMAD SHOLEH AKMAL NUR DZAKI
NIM	: 21010009
JUDUK TA	: SISTEM KOMUNIKASI DAN SENSOR PADA ROBOT HUMANOID *SEMI-AUTO*

Penguji 2

No	Hari/tanggal	Uraian	Tanda tangan
	Selasa / 14/8/2024	Acc laporan dan A1-f	

Penguji 2

...Q1.D.M.S.Pd, MT