

	<p>KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI POLITEKNIK NEGERI SRIWIJAYA Jalan Sriwijaya Negara, Palembang 30139 Telp. 0711-353414 fax. 0711-355918 Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id</p>	 
REKOMENDASI UJIAN LAPORAN AKHIR (LA)		

Pembimbing Laporan Akhir memberikan rekomendasi kepada,

Nama : Kgs. M. Mailan
NIM : 0614 3032 0202
Jurusan/Program Studi : Teknik Elektro / Teknik Elektronika
Judul Laporan Akhir : Mobile Robot Pendeteksi Warna Dan Pengikut Bola Dengan Menggunakan Metode Image Processing.

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Akhir (LA) pada Tahun Akademik 2016 - 2017

Palembang, Juli 2017

Pembimbing I



Ir. M. Nawawi, M.T.
NIP. 19631222 199103 1 006

Pembimbing II



Dr. Eng. Tresna Dewi, ST., M.Eng.
NIP. 19771125 200003 2 001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI SRIWIJAYA

Jalan Sriwijaya Negara, Palembang 30139
Telp. 0711-353414 Fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id








LEMBAR BIMBINGAN LAPORAN AKHIR

Lembar : 1

Nama Mahasiswa : Kgs. M. Mailan
NIM : 0614 3032 0202
Jurusan/ Program Studi : Teknik Elektro/ Teknik Elektronika
Judul Laporan Akhir : Mobile Robot Pendeteksi Warna Dan Pengikut Bola
Dengan Menggunakan Metode Image Processing
Dosen Pembimbing I : Ir. M. Nawawi, M.T.
NIP : 19631222 199103 1 006

No	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1	17/ -2016 Nov	Pengajuan Judul Proposal LA	
2.	25/ -2016 Nov	ACC Judul Proposal LA	
3.	1/ -2016 Des	KONSULTASI Proposal LA	
4.	15/ -2016 Des	Konsultasi Revisi Bab 3 Proposal LA	
5	21/ -2016 Des	Konsul kata Pengantar dan Daftar isi Proposal LA	
6.	9/ -2017 Feb	Konsul masalah alat-alat LA dan Perubahan pada Judul Proposal LA	
7.	24/ -2017 mar	Konsul Progres alat LA dan melanjutkan Pembuatan alat LA.	

No	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
8.	7 ⁻²⁰¹⁷ /april	Bab I diperbaiki. BAB II Ace	
9.	3 ⁻²⁰¹⁷ /Mei	Acc Bab 1 Laporan	
10	16 ⁻²⁰¹⁷ /Juni	Konsultasi Bab 3,4,5 dan melaporkan alat.	
11	21 ⁻²⁰¹⁷ /Juni	konsultasi dan Revisi Bab 4 Laporan Akhir	
	3 Juli 2017	Ace menghidu ujian PA	

Palembang, 2 Agustus 2017

Ketua Program Studi,



Amperawan, ST., M.T.

NIP. 19670523 199303 1 002

Catatan:

Ketua Jurusan/Ketua Program Studi harus memeriksa jumlah pelaksanaan bimbingan sesuai yang dipersyaratkan dalam Pedoman Laporan Akhir sebelum menandatangani Lembar bimbingan ini, Lembar pembimbingan LA ini harus dilampirkan dalam Laporan Akhir



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI

POLITEKNIK NEGERI SRIWIJAYA

Jalan Srijaya Negara, Palembang 30139

Telp. 0711-353414 Fax. 0711-355918

Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id










LEMBAR BIMBINGAN LAPORAN AKHIR

Lembar : 1

Nama Mahasiswa : Kgs. M. Mailan
NIM : 0614 3032 0202
Jurusan/ Program Studi : Teknik Elektro/ Teknik Elektronika
Judul Laporan Akhir : Mobile Robot Pendeteksi Warna Dan Pengikut Bola
 Dengan Menggunakan Metode Image Processing
Dosen Pembimbing II : Dr. Eng. Tresna Dewi, ST., M.Eng
NIP : 19771125 200003 2 001

No	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.	17/ -16 November	Pengajuan Judul Proposal LA	
2.	21/ -2016 Nov	Acc Judul proposal LA	
3.	29/ -2016 Nov	Konsul Proposal LA	
4.	9/ -2016 Des	Revisi Bab 2 Proposal LA	
5.	23/ -2016 Des	Konsul Kata Pengantar dan Daftar Isi Serta PPT Proposal LA	
6.	8/ -2017 Feb	Konsul masalah Alat LA dan Perubahan kalimat Pada Judul Proposal LA	
7.	6/ -2017 Maret	Konsul Perkembangan Alat LA dan melanjutkan Pembuatan Alat LA.	

No	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
8	3 ¹ / _{Mei} -2017	Konsul Laporan LA Bab 1 dan Bab 2 serta konsul Alat	
9	15 ¹ / _{Mei} -2017	Konsul Laporan LA Bab 3 dan konsul Alat	
10	20 ¹ / _{Juni} -2017	Konsultasi Bab 4 dan Bab 5 Laporan Akhir.	
11	21 ¹ / _{Juni} -2017	Acc Bab 4	
12	3 ¹ / _{Juli} -2017	Konsultasi Bab 5, kata pengantar, dll	
13	4 ¹ / _{Juli} -2017	Acc Mengikuti Ujian Laporan Akhir	
14			

Palembang, 2 Agustus 2017

Ketua Program Studi,



Amperawan, ST., M.T.

NIP. 19670523 199303 1 002

Catatan:

Ketua Jurusan/Ketua Program Studi harus memeriksa jumlah pelaksanaan bimbingan sesuai yang dipersyaratkan dalam Pedoman Laporan Akhir sebelum menandatangani Lembar bimbingan ini, Lembar pembimbingan LA ini harus dilampirkan dalam Laporan Akhir



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI

POLITEKNIK NEGERI SRIWIJAYA
 Jalan Sriwijaya Negara, Palembang 30139
 Telp. 0711-353414 Fax. 0711-355918

Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id



KESEPAKATAN BIMBINGAN LAPORAN AKHIR (LA)

Kami yang bertanda tangan di bawah ini,

Pihak Pertama

Nama Mahasiswa : Kgs. M. Mailan
 NIM : 0614 3032 0202
 Jurusan : Teknik Elektro
 Program Studi : Teknik Elektronika

Pihak Kedua

Nama : Ir. M. Nawawi, M.T.
 NIP : 19631222 199103 1 006
 Jurusan : Teknik Elektro
 Program Studi : Teknik Elektronika

Pada hari ini *Jumat* tanggal *3 Maret 2017* telah sepakat untuk melakukan konsultasi bimbingan Laporan Akhir.

Konsultasi bimbingan sekurang-kurangnya 1 (satu) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari *Jumat* Pukul *09.00* tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Laporan Akhir.

Palembang, *3 Maret 2017*

Pihak Pertama,

Kgs. M. Mailan
 NIM. 0614 3032 0202

Pihak Kedua,

Ir. M. Nawawi, M.T.
 NIP. 19631222 199103 1 006

Mengetahui,
 Ketua Program Studi

Amperawan, ST., M.T.
 NIP. 19670523 199303 1 002



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI

POLITEKNIK NEGERI SRIWIJAYA

Jalan Srijaya Negara, Palembang 30139

Telp. 0711-353414 Fax. 0711-355918

Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id



KESEPAKATAN BIMBINGAN LAPORAN AKHIR (LA)

Kami yang bertanda tangan di bawah ini,

Pihak Pertama

Nama Mahasiswa : Kgs. M. Mailan
 NIM : 0614 3032 0202
 Jurusan : Teknik Elektro
 Program Studi : Teknik Elektronika

Pihak Kedua

Nama : Dr. Eng. Tresna Dewi, ST., M.Eng
 NIP : 19771125 200003 2 001
 Jurusan : Teknik Elektro
 Program Studi : Teknik Elektronika

Pada hari ini *Rabu* tanggal *8 - Maret 2017* telah sepakat untuk melakukan konsultasi bimbingan Laporan Akhir.

Konsultasi bimbingan sekurang-kurangnya 1 (satu) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari *Senin* Pukul *08:30* tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Laporan Akhir.

Palembang, *8 - Maret 2017*

Pihak Pertama,

Kgs. M. Mailan
 NIM. 0614 3032 0202

Pihak Kedua,

Dr. Eng. Tresna Dewi, ST., M.Eng
 NIP. 19771125 200003 2 001

Mengetahui,
 Ketua Program Studi

Amperawan, ST., M.T.
 NIP. 19670523 199303 1 002



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
POLITEKNIK NEGERI SRIWIJAYA
Jalan Sriwijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id



PELAKSANAAN REVISI LAPORAN AKHIR

Mahasiwa Berikut,

Nama : Kgs. M. Mailan
NIM : 0614 3032 0202
Jurusan/Program Studi : T.Elektro/T.Elektronika
Judul Laporan Akhir : Mobile Robot Pendeteksi Warna dan Pengikut Bola Dengan Menggunakan Metode Image Processing.

Telah melaksanakan revisi terhadap Laporan Akhir yang diujikan pada hari Rabu tanggal 19 bulan Juli tahun 2017. Pelaksanaan revisi terhadap Laporan Akhir tersebut telah disetujui oleh Dosen Penguji yang memberikan revisi:

No.	Komentar	Nama Dosen Penguji *)	Tanggal	Tanda Tangan
1.	Sudah direvisi	Abdurahman, S.T., M.Kom	24/7/2017	
2.		Ir. M. Nawawi, M.T.		
3.	Ace	Ir. Faisal Damsi., M.T.	25/07/2017	
4.	Tidak ada revisi	Dewi Permata Sari, S.T., M.Kom	24/7/17	
5.	Telah direvisi.	Yeni Irdayanti, S.T., M.Kom	7/Agust 17	
6.	Ace	Selamat Muslimin, S.T., M.Kom	26/7/2017	

Palembang, Juli 2017

Ketua Penguji **)

(Ir. Faisal Damsi., M.T.)

NIP. 19630218 199403 1 001

Catatan:

*) Dosen Penguji yang memberikan revisi saat ujian laporan akhir.

***) Dosen Penguji yang di tugaskan sebagai Ketua penguji saat ujian laporan akhir.

Lembaran pelaksanaan revisi ini harus dilampirkan dalam Laporan Akhir.

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI****POLITEKNIK NEGERI SRIWIJAYA****Jalan Sriwijaya Negara, Palembang 30139**

Telp. 0711-353414 Fax. 0711-355918

Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id**SURAT PEMINJAMAN DAN PEMAKAIAN ALAT DI LABORATORIUM**

Palembang, 15 Juni 2017

Hal : Permohonan Peminjaman dan Pemakaian Laboratorium

Kepada Yth.
Kepala Laboratorium Teknik Elektronika
Politeknik Negeri Sriwijaya
di Palembang

Dengan Hormat,

Sehubungan dengan penelitian dan pembuatan alat yang dilakukan guna menyelesaikan Tugas Akhir, sesuai dengan kurikulum di Program Studi Teknik Elektronika Jurusan Teknik Elektro Politeknik Negeri Sriwijaya, maka dengan ini:

Nama : Kgs. M. Mailan
NIM : 0614 3032 0202
Prodi/Jurusan : Teknik Elektronika / Teknik Elektro
Judul Laporan Akhir : Mobile Robot Pendeteksi Warna dan Pengikut Bola Dengan Menggunakan Metode Image Processing.

Dengan ini saya mengajukan permohonan izin untuk peminjaman dan pemakaian Laboratorium Teknik Elektronika Politeknik Negeri Sriwijaya, yang akan dilaksanakan pada tanggal 15. Juni 2017 sampai 19. Juni 2017. Demikian kiranya pelaksanaan penelitian ini dengan izin Bapak/Ibu dapat terlaksana dengan sebaik-baiknya. Atas perhatiannya, saya ucapkan terima kasih.

Dosen Pembimbing II

Dr. Eng. Tresna Dewi, ST., M.Eng
NIP. 19771125 200003 2 001

Mahasiswa,

Kgs. M. Mailan
NIM. 0614 3032 0202

Mengetahui,
a.n. Ketua Program Studi

Selamat Muslimin, S.T., M.Kom
NIP. 19790722 200801 1 007



Raspberry Pi 3 Model B



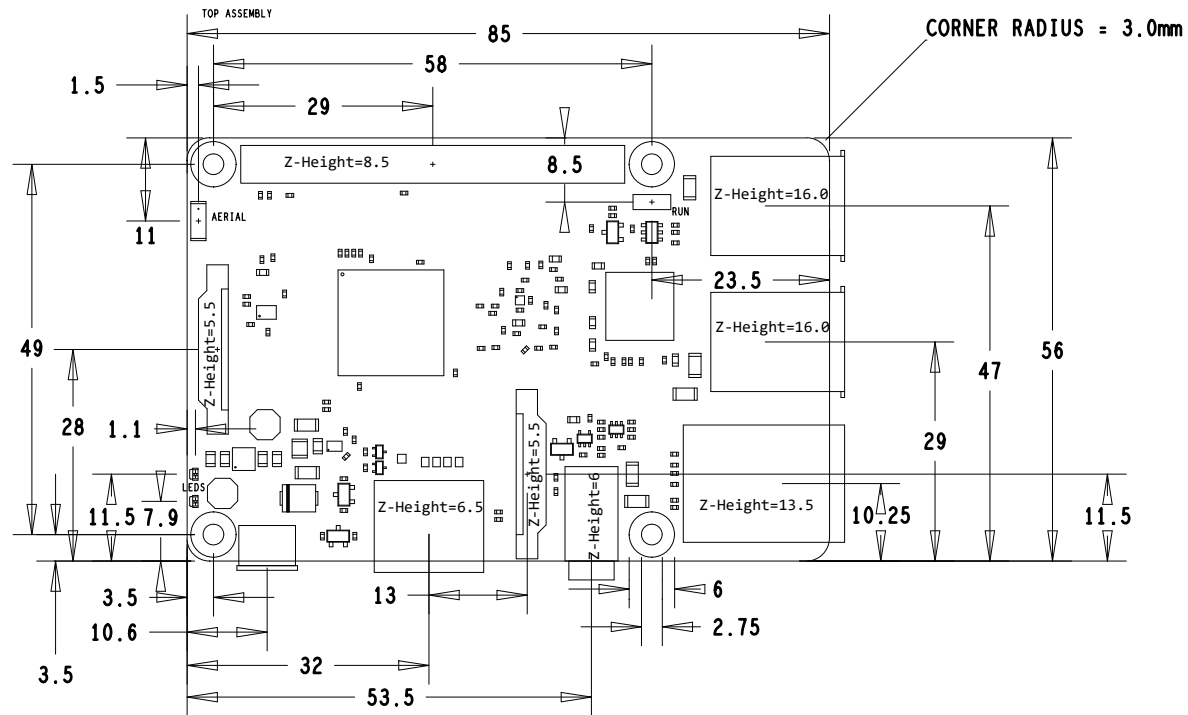
RASPBERRYPI-MODB-1GB




RPI-MODB-16GB-NOOBS

Technical Specification:

- Broadcom BCM2837 64bit ARMv7 Quad Core Processor powered Single Board Computer running at 1.2GHz
- 1GB RAM
- BCM43143 WiFi on board
- Bluetooth Low Energy (BLE) on board
- 40pin extended GPIO
- 4 x USB 2 ports
- 4 pole Stereo output and Composite video port
- Full size HDMI
- CSI camera port for connecting the Raspberry Pi camera
- DSI display port for connecting the Raspberry Pi touch screen display
- Micro SD port for loading your operating system and storing data
- Upgraded switched Micro USB power source (now supports up to 2.4 Amps)
- Expected to have the same form factor has the Pi 2 Model B, however the LEDs will change position

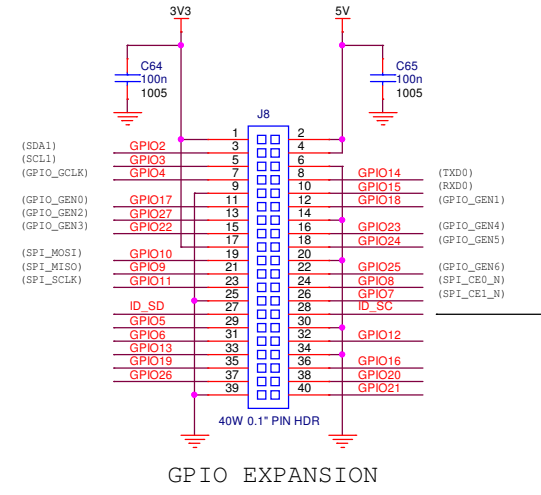
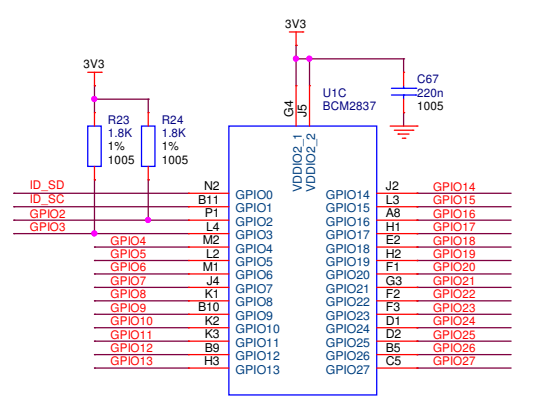
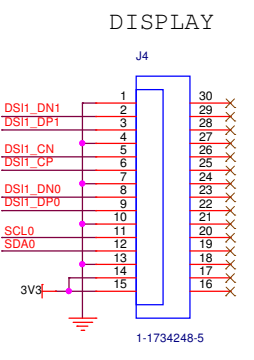
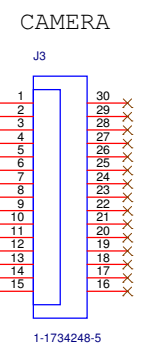
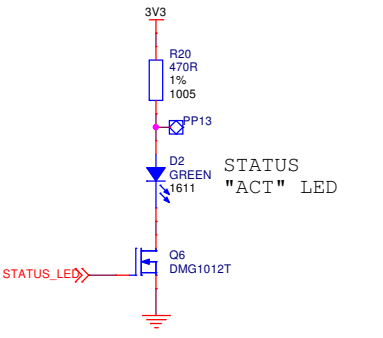
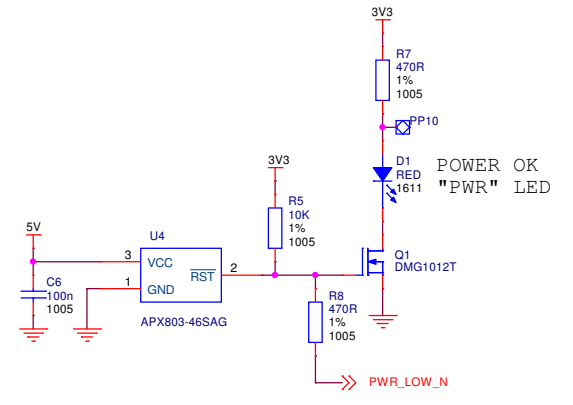
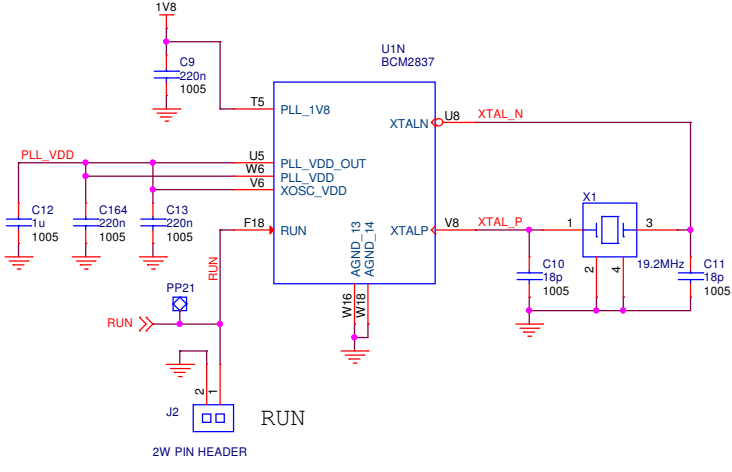
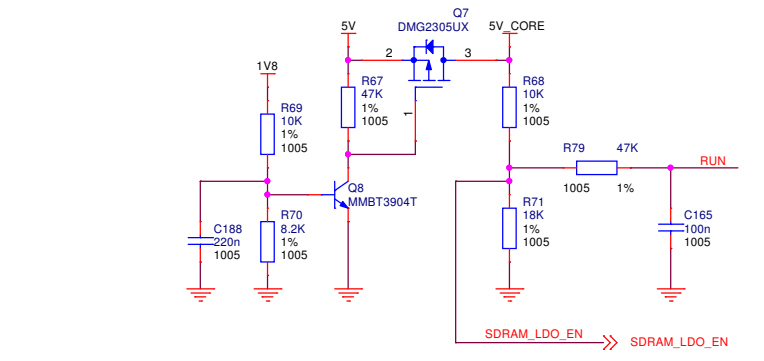
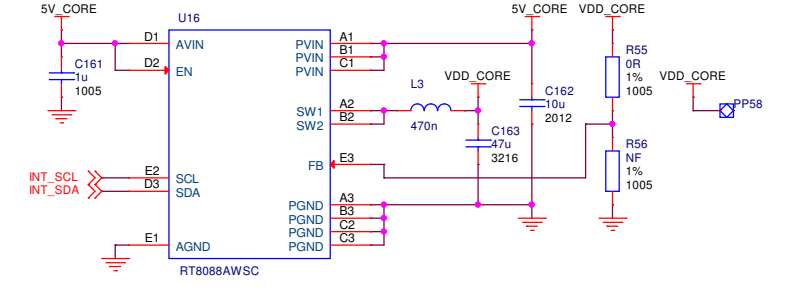
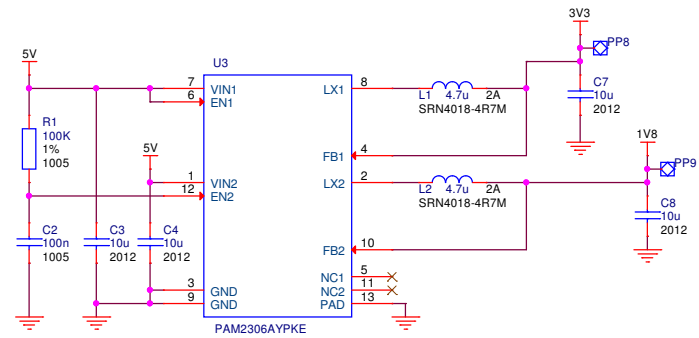
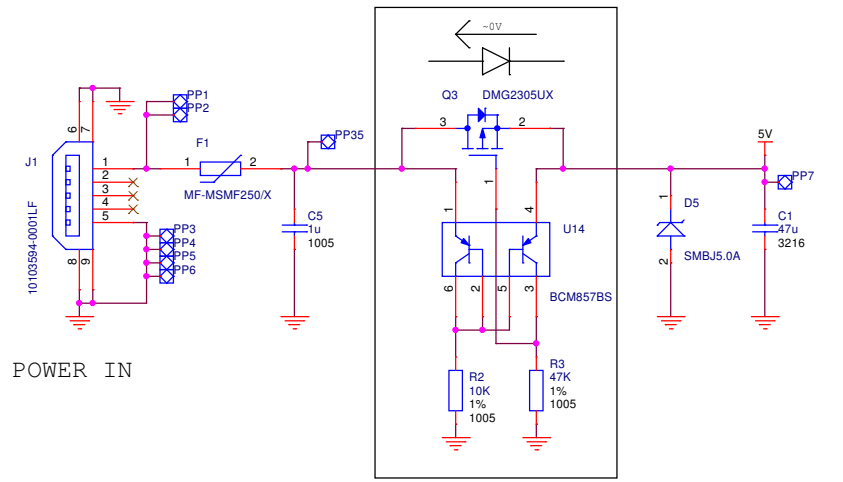


4x M2.5 MOUNTING HOLES
 DRILLED TO 2.75 +/- 0.05mm

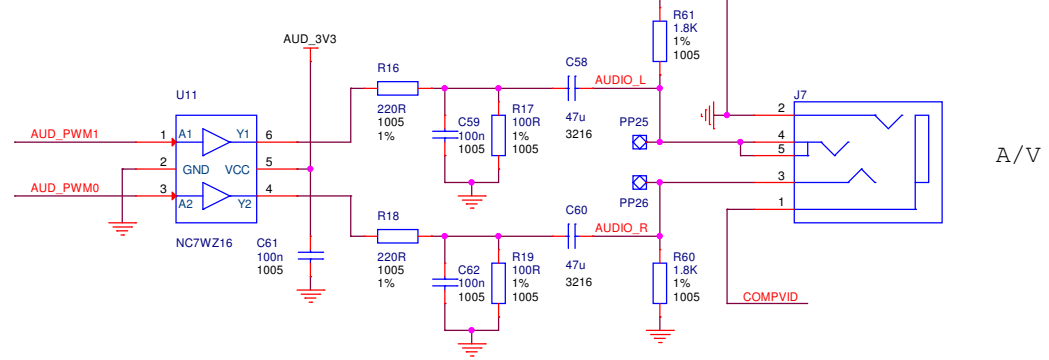
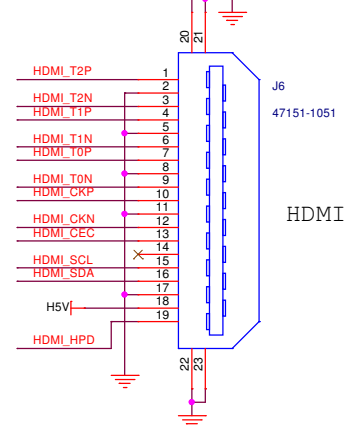
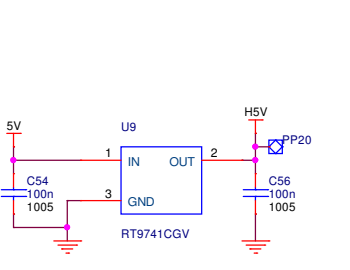


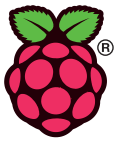
Raspberry Pi
 www.raspberrypi.org
 © Raspberry Pi 2015

TITLE	RASPERRY PI 3 MODEL B		
DATE	06/10/2015	REF	RPI-3B-V1_2
DRAWN	James Adams	APVD	James Adams



ID_SD and ID_SC PINS:
 These pins are reserved for HAT ID EEPROM. At boot time this I2C interface will be interrogated to look for an EEPROM that identifies the attached board and allows automatic setup of the GPIOs (and optionally, Linux drivers). **DO NOT USE** these pins for anything other than attaching an I2C ID EEPROM. Leave unconnected if ID EEPROM not required.





Raspberry Pi



CAMERA MODULE

Product Name	Raspberry Pi Camera Module
Product Description	High definition camera module compatible with the Raspberry Pi model A and model B. Provides high sensitivity, low crosstalk and low noise image capture in an ultra small and lightweight design. The camera module connects to the Raspberry Pi board via the CSI connector designed specifically for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor.
RS Part Number	775-7731
Specifications	
Image Sensor	Omnivision 5647 CMOS image sensor in a fixed-focus module with integral IR filter
Resolution	5-megapixel
Still picture resolution	2592 x 1944
Max image transfer rate	1080p: 30fps (encode and decode) 720p: 60fps
Connection to Raspberry Pi	15 Pin ribbon cable, to the dedicated 15-pin MIPI Camera Serial Interface (CSI-2)
Image control functions	Automatic exposure control Automatic white balance Automatic band filter Automatic 50/60 Hz luminance detection Automatic black level calibration
Temp range	Operating: -30° to 70° Stable image: 0° to 50°
Lens size	1/4"
Dimensions	20 x 25 x 10mm
Weight	3g

Accessories



▲ Raspberry Pi Model B - **756-8308**



▲ Camera case **784-6193**



▲ 8GB SD card pre-programmed with NOOBS - **779-6770**



▲ Expansion board **772-2974**



▲ WiFi dongle **760-3621**



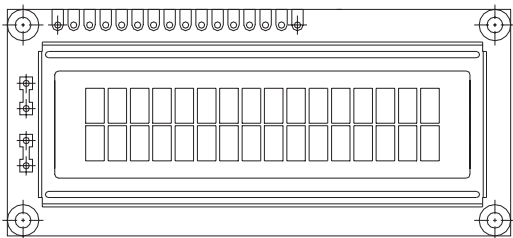
▲ 10400mAh Li-Ion battery pack **775-7517**



▲ Raspberry Pi user guide **768-6686**



16 x 2 Character LCD


FEATURES

- 5 x 8 dots with cursor
- Built-in controller (KS 0066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- B/L to be driven by pin 1, pin 2 or pin 15, pin 16 or A.K (LED)
- N.V. optional for + 3V power supply

MECHANICAL DATA		
ITEM	STANDARD VALUE	UNIT
Module Dimension	80.0 x 36.0	mm
Viewing Area	66.0 x 16.0	mm
Dot Size	0.56 x 0.66	mm
Character Size	2.96 x 5.56	mm

ABSOLUTE MAXIMUM RATING					
ITEM	SYMBOL	STANDARD VALUE			UNIT
		MIN.	TYP.	MAX.	
Power Supply	VDD-VSS	- 0.3	-	7.0	V
Input Voltage	VI	- 0.3	-	VDD	V

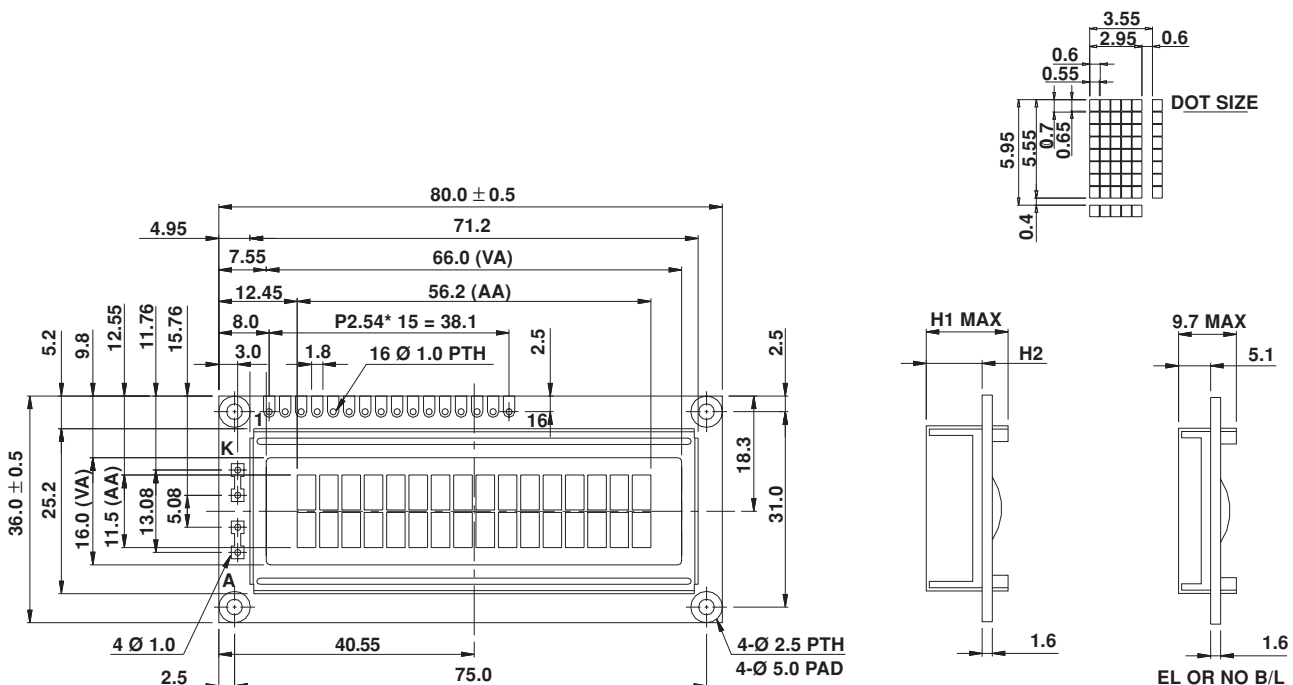
NOTE: VSS = 0 Volt, VDD = 5.0 Volt

ELECTRICAL SPECIFICATIONS							
ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT	
			MIN.	TYP.	MAX.		
Input Voltage	VDD	VDD = + 5V	4.7	5.0	5.3	V	
		VDD = + 3V	2.7	3.0	5.3	V	
Supply Current	IDD	VDD = 5V	-	1.2	3.0	mA	
Recommended LC Driving Voltage for Normal Temp. Version Module	VDD - V0	- 20 °C	-	-	-	V	
		0°C	4.2	4.8	5.1		
		25°C	3.8	4.2	4.6		
		50°C	3.6	4.0	4.4		
LED Forward Voltage	VF	25°C	-	4.2	4.6	V	
LED Forward Current	IF	25°C	Array	-	130	260	mA
			Edge	-	20	40	
EL Power Supply Current	IEL	Vel = 110VAC:400Hz	-	-	5.0	mA	

DISPLAY CHARACTER ADDRESS CODE:																
Display Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM Address	00	01														0F
DD RAM Address	40	41														4F

PIN NUMBER	SYMBOL	FUNCTION
1	Vss	GND
2	Vdd	+ 3V or + 5V
3	Vo	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/W	H/L Read/Write Signal
6	E	H → L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A/Vee	+ 4.2V for LED/Negative Voltage Output
16	K	Power Supply for B/L (OV)

DIMENSIONS in millimeters



LED - H/L B/L		
	HIGH	LOW
H1	13.2	12.1
H2	8.6	7.5

This datasheet has been download from:

www.datasheetcatalog.com

Datasheets for electronics components.


```

#include <opencv2/highgui/highgui.hpp>
#include <opencv2/imgproc/imgproc.hpp>
#include <iostream>
#include <cstdlib>
#include <stdio.h>
#include <unistd.h> //Used for UART
#include <fcntl.h> //Used for UART
#include <termios.h>
#include <string.h>
using namespace std;
using namespace cv;

int iLowH_Red = 168;
int iHighH_Red = 179;
int iLowS_Red = 20;
int iHighS_Red = 255;
int iLowV_Red = 40;
int iHighV_Red = 255;

RNG rng(12345);

int main(int argc, char** argv) {
    int uart0_filestream = -1;
    uart0_filestream = open("/dev/ttyAMA0", O_RDWR | O_NOCTTY | O_NDELAY); //Open in non
    blocking read/write mode
    if (uart0_filestream == -1) {
        //ERROR - CAN'T OPEN SERIAL PORT
        printf("Error - Unable to open UART. Ensure it is not in use by another application\n");
    }

    struct termios options;
    tcgetattr(uart0_filestream, &options);
    options.c_cflag = B9600 | CS8 | CLOCAL | CREAD; //<Set baud rate
    options.c_iflag = IGNPAR;
    options.c_oflag = 0;
    options.c_lflag = 0;
    tcflush(uart0_filestream, TCIFLUSH);
    tcsetattr(uart0_filestream, TCSANOW, &options);

    //----- TX BYTES -----
    unsigned char tx_buffer[20];
    unsigned char *p_tx_buffer;
    string pesan = "Hello CYBORG";
    p_tx_buffer = &tx_buffer[0];
    *p_tx_buffer++ = 'H';
    *p_tx_buffer++ = 'e';
    *p_tx_buffer++ = 'l';
    *p_tx_buffer++ = 'l';
    *p_tx_buffer++ = 'o';

    //----- CLOSE THE UART -----

```

```

VideoCapture stream1(0); //0 is the id of video device.0 if you have only one camera.

if (!stream1.isOpened()) { //check if video device has been initialised
    cout << "cannot open camera";
}
stream1.set(CV_CAP_PROP_FRAME_WIDTH, 120);
stream1.set(CV_CAP_PROP_FRAME_HEIGHT, 90);

//unconditional loop
while (true) {
    Mat cameraFrame;
    stream1.read(cameraFrame);
    imshow("cam", cameraFrame);
    int iLastX = -1;
    int iLastY = -1;
    //Capture a temporary image from the camera

    stream1.read(cameraFrame);

    //Create a black image with the size as the camera output
    Mat imgLines = Mat::zeros(cameraFrame.size(), CV_8UC3);
    ;

    while (true) {
        Mat imgOriginal;

        bool bSuccess = stream1.read(imgOriginal); // read a new frame from video

        if (!bSuccess) //if not success, break loop
        {
            cout << "Cannot read a frame from video stream" << endl;
            break;
        }
        Mat imgHSV;

        cvtColor(imgOriginal, imgHSV, COLOR_BGR2HSV); //Convert the stream1tured frame from
BGR to HSV

        Mat imgThresholdedRed;
        Mat imgThresholdedGreen;
        Mat imgThresholdedBlue;

        inRange(imgHSV, Scalar(iLowH_Red, iLowS_Red, iLowV_Red), Scalar(iHighH_Red, iHighS_Red,
iHighV_Red), imgThresholdedRed); //Threshold the image

        //morphological opening (removes small objects from the foreground)
        erode(imgThresholdedRed, imgThresholdedRed, getStructuringElement(MORPH_ELLIPSE,
Size(5, 5)));
        dilate(imgThresholdedRed, imgThresholdedRed, getStructuringElement(MORPH_ELLIPSE,
Size(5, 5)));

```

```

//morphological closing (removes small holes from the foreground)
dilate(imgThresholdedRed, imgThresholdedRed, getStructuringElement(MORPH_ELLIPSE,
Size(5, 5)));
erode(imgThresholdedRed, imgThresholdedRed, getStructuringElement(MORPH_ELLIPSE,
Size(5, 5)));

//Calculate the moments of the thresholded image
Moments oMomentsRed = moments(imgThresholdedRed);

double dM01R = oMomentsRed.m01;
double dM10R = oMomentsRed.m10;
double dAreaR = oMomentsRed.m00;

bool ada_bulat = false;
vector<vector<Point> > contours;
//vector<vector<Point> > result;
vector<Vec4i> hierarchy;
vector<Point> approx;
// apply pre-processing functions
findContours(imgThresholdedRed, contours, hierarchy, CV_RETR_TREE,
CV_CHAIN_APPROX_SIMPLE, Point(0, 0));

/// Draw contours
Mat drawing = Mat::zeros(imgThresholdedRed.size(), CV_8UC3);
for (int i = 0; i < contours.size(); i++) {
    //cout<<contours.size()<<endl;
    Scalar color = Scalar(rng.uniform(0, 255), rng.uniform(0, 255), rng.uniform(0, 255));
    drawContours(drawing, contours, i, color, 2, 8, hierarchy, 0, Point());
    approxPolyDP(Mat(contours[i]), approx, arcLength(Mat(contours[i]), true)*0.02, true);

    // Skip small or non-convex objects
    if (fabs(cv::contourArea(contours[i])) < 100 || !isContourConvex(approx))
        continue;
    cout << approx.size() << endl;
    if (approx.size() > 6) {
        double area = cv::contourArea(contours[i]);
        Rect r = cv::boundingRect(contours[i]);
        int radius = r.width / 2;

        if (abs(1 - ((double) r.width / r.height)) <= 0.2 &&
            abs(1 - (area / (CV_PI * (radius * radius)))) <= 0.2){
            cout << "CIRCLE" << endl;
            ada_bulat = true;
        }
    }
}
}
// if the area <= 10000, I consider that there are no object in the image and it's because of
the noise, the area is not zero
//if (dAreaR > 5000)

```

```

{
    //calculate the position of the ball
    int posXR = dM10R / dAreaR;
    int posYR = dM01R / dAreaR;
    cout << "X Red=" << posXR << " Y Red=" << posYR << endl;
    pesan = to_string(posXR) + "," + to_string(posYR) + "," + to_string(ada_bulat) + char(10);
    if (uart0_filestream != -1) {
        //int count = write(uart0_filestream, &tx_buffer[0], (p_tx_buffer - &tx_buffer[0]));
        //Filestream, bytes to write, number of bytes to write
        int count = write(uart0_filestream, pesan.c_str(), pesan.length()); //Filestream, bytes to
        write, number of bytes to write
        if (count < 0) {
            printf("UART TX error\n");
        }
    }

    if (iLastX >= 0 && iLastY >= 0 && posXR >= 0 && posYR >= 0) {
        //Draw a red line from the previous point to the current point
        //line(imgLines, Point(posX, posY), Point(iLastX, iLastY), Scalar(0,0,255), 2);
    }

    iLastX = posXR;
    iLastY = posYR;
}

imshow("Thresholded Red", imgThresholdedRed); //show the thresholded image

//cvCvtColor(img,imgGrayScale,CV_BGR2GRAY);

//thresholding the grayscale image to get better results
//cvThreshold(imgGrayScale,imgGrayScale,128,255,CV_THRESH_BINARY);

/// Show in a window
namedWindow("Contours", CV_WINDOW_AUTOSIZE);
imshow("Contours", drawing);

imgOriginal = imgOriginal + imgLines;
imshow("Original", imgOriginal); //show the original image

if (waitKey(30) == 27) //wait for 'esc' key press for 30ms. If 'esc' key is pressed, break loop
{
    cout << "esc key is pressed by user" << endl;
    break;
}
}
}
close(uart0_filestream);

return 0;
}

```

```

#define ln1 12
#define ln2 11
#define ln3 10
#define ln4 9
#define ena 6
#define enb 5
#define trig_depan 33
#define echo_depan 32
#define trig_kanan 52
#define echo_kanan 53
#define trig_kiri 30
#define echo_kiri 31
#include "Wire.h"
#include "LiquidCrystal.h";
#define btn_Ok 45 //Push Button Ok pada robot
#define btn_Up 46 //Push Button Up pada robot
#define btn_Down 47 //Push Button Down pada robot

#include <EEPROM.h>
LiquidCrystal lcd(4, 3, 36, 37, 38, 39);

String inputString = ""; // a string to hold incoming data
boolean stringComplete = false; // whether the string is complete

int jarak_depan;
int jarak_kanan;
int jarak_kiri;
int Speed;
int posX , posY, Circle;

byte menu = 1;
void setup() {
// put your setup code here, to run once:
pinMode (ln1, OUTPUT);
pinMode (ln2, OUTPUT);
pinMode (ln3, OUTPUT);
pinMode (ln4, OUTPUT);
pinMode (ena, OUTPUT);
pinMode (enb, OUTPUT);
pinMode(trig_depan, OUTPUT);
pinMode(echo_depan, INPUT);
pinMode(trig_kanan, OUTPUT);
pinMode(echo_kanan, INPUT);
pinMode(trig_kiri, OUTPUT);
pinMode(echo_kiri, INPUT);
pinMode(btn_Ok, INPUT_PULLUP); //Push Button
pinMode(btn_Up, INPUT_PULLUP);
pinMode(btn_Down, INPUT_PULLUP);
Serial.begin(9600);
Serial1.begin(9600);
Serial3.begin(9600);

```

```

inputString.reserve(200);

lcd.begin(16, 2);
lcd.setCursor(2, 0);
lcd.print(" KGS M MAILAN ");
lcd.setCursor(0, 2);
lcd.print(" 061430320202");
delay(3000); //delay 3 detik
lcd.clear();
posX = 0;
posY = 0;
Circle = 0;
Speed = 0;
}

void loop() {
  tampil_menu();
  if (digitalRead(btn_Ok) == 0) {
    delay(1000);
    if (menu == 2)start_robot();
    else if (menu == 3)setting_Speed();
    else if (menu == 4)tampil_sensor();
  }

  if (digitalRead(btn_Up) == 0) {
    delay(200);
    menu = menu + 1;
    if (menu > 4)menu = 1;
    tampil_menu();
  }

  if (digitalRead(btn_Down) == 0) {
    delay(200);
    menu = menu - 1;
    if (menu < 1)menu = 4;
    tampil_menu();
  }
}

void tampil_menu () {

  lcd.clear();
  if (menu == 1) {
    lcd.setCursor(0, 0);
    lcd.print("MOBILE ROBOT ");
    delay(200);
  }

  if (menu == 2) {
    lcd.setCursor(0, 0);
    lcd.print("Start Robot ");
  }
}

```

```

    delay (200);
}

if (menu == 3) {
    lcd.setCursor (0, 0);
    lcd.print("Setting Speed  ");
    delay(200);
}

if (menu == 4) {
    lcd.setCursor (0, 0);
    lcd.print("Baca Sensor  ");
    delay(200);
}
}

void start_robot() {
    while (digitalRead(btn_Ok) == 1) {
        baca_depan();
        baca_kanan();
        baca_kiri();

        serialEvent(); //call the function
        // print the string when a newline arrives:
        if (stringComplete) {
            //Serial.println(inputString);
            posX = getValue(inputString, ',', 0).toInt();
            posY = getValue(inputString, ',', 1).toInt();
            Circle = getValue (inputString, ',', 2).toInt();

            Serial.print("pos x = "); Serial.println(posX);
            lcd.setCursor (0, 1);
            lcd.print (posX);
            lcd.print (" ");
            Serial.print("pos y = "); Serial.println(posY);
            lcd.setCursor (5, 1);
            lcd.print (posY);
            lcd.print (" ");
            Serial.print("Circle = "); Serial.println(Circle);
            lcd.setCursor (10, 1);
            lcd.print (Circle);
            lcd.print (" ");

            if (jarak_depan <= 10) {
                henti ();
            }
            else if (jarak_kanan <= 10) {
                belok_kiri();
            } else if (jarak_kiri <= 10) {
                belok_kanan();
            }
        }
    }
}

```

```

else if (posX == 0) {
    henti();
}
else if (posX < 40) {
    belok_kiri();
}
else if (posX > 80) {
    belok_kanan();
}
else {
    if (jarak_depan <= 10 && Circle == 1) {
        henti();
    } else if (jarak_depan <= 10 && Circle == 0) {
        belok_kiri();
        delay (1500);
    }
    else {
        maju();
    }
}
inputString = "";
stringComplete = false;
}
}
delay(1000);
tampil_menu();
}

```

```

void setting_Speed () {
    lcd.setCursor(0, 0);
    lcd.print(" ");
    lcd.setCursor(0, 0);
    lcd.print(Speed);
}

```

```

while (digitalRead(btn_Ok) == 1) {
    if (digitalRead(btn_Up) == 0) {
        delay(20);
        Speed = Speed + 1;
        lcd.setCursor(0, 0);
        lcd.print(Speed);
        lcd.print(" ");
    }
    if (digitalRead(btn_Down) == 0) {
        delay(20);
        Speed = Speed - 1;
        lcd.setCursor(0, 0);
        lcd.print(Speed);
        lcd.print(" ");
    }
}
delay (1000);

```



```

    lcd.setCursor(0, 0);
    lcd.print("Set Speed Slesai");
    simpan_semua();
    delay(1000);
    tampil_menu();
}

void tampil_sensor() {          //tampil sensor ultrasonik
    lcd.setCursor(0, 0);
    lcd.print("Baca Sensor    ");
    lcd.clear ();
    delay (1000);

    while (digitalRead(btn_Ok) == 1) {
        baca_depan ();
        baca_kanan ();
        baca_kiri ();

        lcd.setCursor(5, 0);
        lcd.print(jarak_depan); lcd.print(" ");
        lcd.setCursor(1, 0);
        lcd.print(jarak_kanan); lcd.print(" ");
        lcd.setCursor(10, 0);
        lcd.print(jarak_kiri); lcd.print(" ");
    }
    delay(1000);
    tampil_menu();
}

void save_eeprom(int direccion, float num)
{
    long valor = num * 10000;

    byte cuatro = (valor & 0xFF);
    byte tres = ((valor >> 8) & 0xFF);
    byte dos = ((valor >> 16) & 0xFF);
    byte uno = ((valor >> 24) & 0xFF);

    EEPROM.write(direccion, cuatro);
    EEPROM.write(direccion + 1, tres);
    EEPROM.write(direccion + 2, dos);
    EEPROM.write(direccion + 3, uno);
}

float load_eeprom(long direccion)
{
    long cuatro = EEPROM.read(direccion);
    long tres = EEPROM.read(direccion + 1);
    long dos = EEPROM.read(direccion + 2);
    long uno = EEPROM.read(direccion + 3);
}

```

```
float num = ((cuatro << 0) & 0xFF) + ((tres << 8) & 0xFFFF) + ((dos << 16) & 0xFFFFF) + ((uno << 24) & 0xFFFFFFF);  
return (num / 10000);  
}
```

```
void simpan_semua ()  
{  
    save_eeprom (16, Speed);  
    //Serial.println("SimpanOK");  
}
```

```
void set_variabel()  
{  
    Speed = load_eeprom(16);  
    //Serial.println("Variables inicializadas");  
}
```

```
void henti() {  
    digitalWrite (In1, 1);  
    digitalWrite (In2, 1);  
    analogWrite (ena, 0);
```

```
    digitalWrite (In3, 1);  
    digitalWrite (In4, 1);  
    analogWrite (enb, 0);
```

```
}
```

```
void mundur() {  
    digitalWrite (In1, 1);  
    digitalWrite (In2, 0);  
    analogWrite (ena, Speed);
```

```
    digitalWrite (In3, 1);  
    digitalWrite (In4, 0);  
    analogWrite (enb, Speed);
```

```
}
```

```
void maju () {  
    digitalWrite (In1, 0);  
    digitalWrite (In2, 1);  
    analogWrite (ena, Speed);
```

```
    digitalWrite (In3, 0);  
    digitalWrite (In4, 1);  
    analogWrite (enb, Speed);
```

```
}
```

```
void belok_kiri() {  
    digitalWrite (In1, 1); // roda kiri mundur  
    digitalWrite (In2, 0);
```

```

analogWrite (ena, 0);

digitalWrite (In3, 0); //roda kanan maju
digitalWrite (In4, 1);
analogWrite (enb, Speed + 30);
}

void belok_kanan() {
digitalWrite (In1, 0); //roda kiri maju
digitalWrite (In2, 1);
analogWrite (ena, Speed + 30);

digitalWrite (In3, 1); //roda kanan mundur
digitalWrite (In4, 0);
analogWrite (enb, 0);

}

void baca_depan() {
digitalWrite(trig_depan, LOW);
delay(2);
digitalWrite(trig_depan, HIGH);
delay(5);
digitalWrite(trig_depan, LOW);
jarak_depan = pulseIn(echo_depan, HIGH);
jarak_depan = microsecondsToCentimeters(jarak_depan);
if (jarak_depan > 200) {
    jarak_depan = 200;
}
Serial.println(jarak_depan);
lcd.setCursor (5, 0);
lcd.print (jarak_depan); lcd.print(" ");

}

void baca_kanan() {
digitalWrite(trig_kanan, LOW);
delay(2);
digitalWrite(trig_kanan, HIGH);
delay(5);
digitalWrite(trig_kanan, LOW);
jarak_kanan = pulseIn(echo_kanan, HIGH);
jarak_kanan = microsecondsToCentimeters(jarak_kanan);
if (jarak_kanan > 200) {
    jarak_kanan = 200;
}
Serial.println(jarak_kanan);
lcd.setCursor (1, 0);
lcd.print (jarak_kanan); lcd.print(" ");

}

```

```

void baca_kiri() {
  digitalWrite(trig_kiri, LOW);
  delay(2);
  digitalWrite(trig_kiri, HIGH);
  delay(5);
  digitalWrite(trig_kiri, LOW);
  jarak_kiri = pulseIn(echo_kiri, HIGH);
  jarak_kiri = microsecondsToCentimeters(jarak_kiri);
  if (jarak_kiri > 200) {
    jarak_kiri = 200;
  }
  Serial.println(jarak_kiri);
  lcd.setCursor (11, 0);
  lcd.print (jarak_kiri); lcd.print(" ");
}
long microsecondsToCentimeters(long microseconds)
{
  return microseconds / 29 / 2;
}

```

```

String getValue(String data, char separator, int index)
{
  int found = 0;
  int strIndex[] = {0, -1};
  int maxIndex = data.length() - 1;

  for (int i = 0; i <= maxIndex && found <= index; i++) {
    if (data.charAt(i) == separator || i == maxIndex) {
      found++;
      strIndex[0] = strIndex[1] + 1;
      strIndex[1] = (i == maxIndex) ? i + 1 : i;
    }
  }

  return found > index ? data.substring(strIndex[0], strIndex[1]) : "";
}

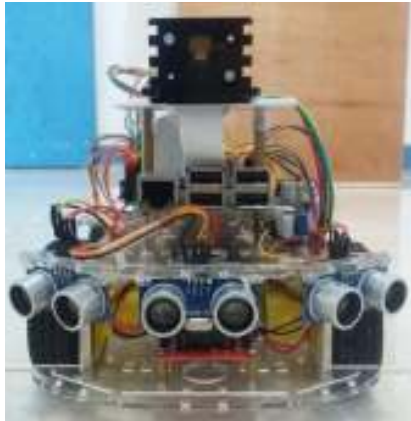
```

SerialEvent occurs whenever a new data comes in the hardware serial RX. This routine is run between each time loop() runs, so using delay inside loop can delay response. Multiple bytes of data may be available.

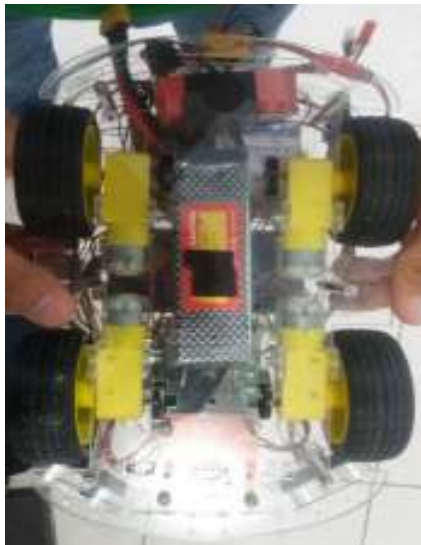
```

void serialEvent() {
  while (Serial1.available()) {
    char inChar = (char)Serial1.read();
    inputString += inChar;
    if (inChar == '\n' || inChar == '\r') {
      stringComplete = true;
    }
  }
}

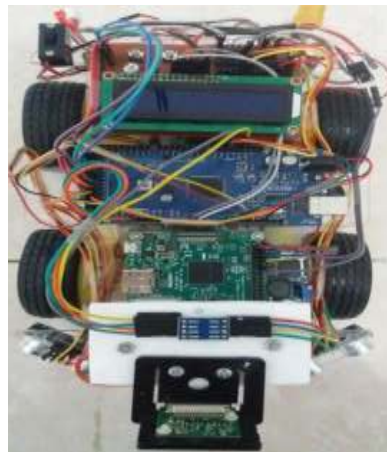
```



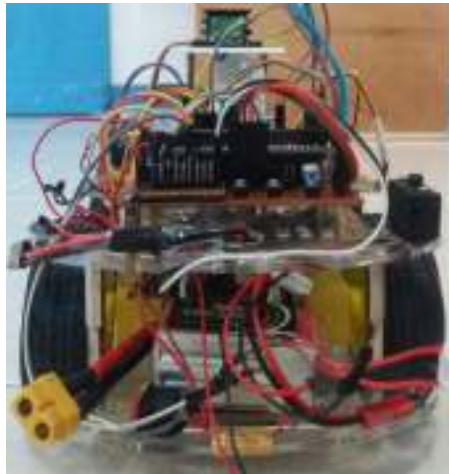
Tampak Depan Mobile Robot



Tampak Bawah Mobile Robot



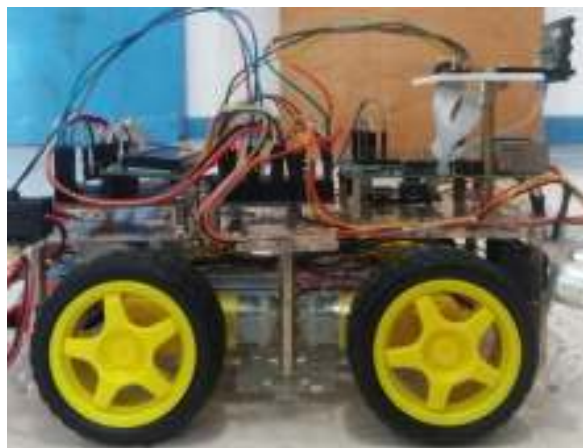
Tampak Atas Mobile Robot



Tampak Belakang Mobile Robot



Gambar 3.8 Tampak Samping Kiri Mobile Robot



Tampak Samping Kanan Mobile Robot