

**ALAT BANTU JALAN PENYANDANG TUNANETRA UNTUK
MENDETEKSI HAMBATAN**

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh :

LEO MAHESA PRATAMA

NIRM : 0031614

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2019**

LEMBAR PENGESAHAN

JUDUL PROYEK AKHIR

**ALAT BANTU JALAN PENYANDANG TUNANETRA UNTUK MENDETEKSI
HAMBATAN**

Oleh :

Leo Mahesa Pratama/ 0031614

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Diploma III Politeknik Manufaktur Negeri Bangka Belitung

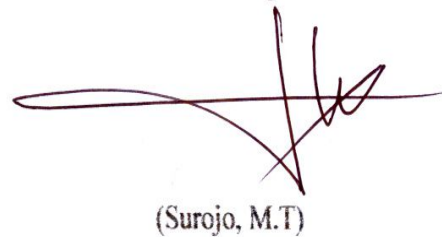
Menyetujui,

Pembimbing 1



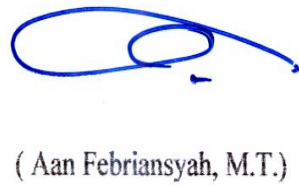
(Yudhi, M.T)

Pembimbing 2



(Surojo, M.T)

Penguji 1



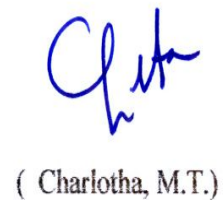
(Aan Febriansyah, M.T.)

Penguji 2



(DR.Parulian Silalahi, M.Pd)

Penguji 3



(Charlotha, M.T.)

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini :

1. Nama Mahasiswa : Leo Mahesa Pratama

NIRM : 0031614

Dengan Judul : Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata di kemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 12 September 2019

Nama Mahasiswa

Tanda Tangan

1. Leo Mahesa Pratama



ABSTRAK

Indera penglihatan adalah salah satu sumber informasi vital bagi manusia. Namun Tak sedikit masyarakat yang mengalami gangguan pada indera penglihatannya. Istilah umum yang digunakan untuk kondisi seseorang yang mengalami gangguan pada indra penglihatannya disebut juga tunanetra. Pada umumnya, penyandang tunanetra menggunakan alat bantu jalan berupa tongkat putih untuk membantu pergerakan dan meningkatkan keamanan dan kemandirian pada saat berjalan. Oleh sebab itu, diperlukan suatu sistem yang dapat menggantikan dan menyempurnakan peran alat bantu jalan yang selama ini telah ada. Maka dibuatlah Proyek akhir yang berjudul “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan”, yang dibuat dengan melakukan modifikasi terhadap tongkat tunanetra yang telah ada dengan cara memadukan teknologi sensor dan output suara terhadap alat bantu jalan yang telah ada. Tujuan utama alat ini untuk mempermudah penyandang tunanetra dalam mendeteksi hambatan yang ada di sekitarnya saat sedang berjalan. Alat ini menggunakan sensor ultrasonic sebagai sensor utama, Arduino uno sebagai kendali utama, dan modul DFPlayer Mini sebagai modul pendukung arduino sebagai output yang mengeluarkan suara. Dari hasil yang didapat alat ini bekerja dengan sesuai dengan yang diharapkan dibuktikan dengan error pendeteksian yang hanya berjumlah 1,66%

Kata kunci : Tunanetra, Alat Bantu, Hambatan

ABSTRACT

Sense of vision is a vital source of information for humans. But not a few people who experience interference with the sense of sight, a general term used for the condition of someone who has a disorder in the sense of sight is also called blind. In general, blind people use walking aids in the form of white sticks to help with movement and increase safety and independence when walking. Therefore, we need a system that can replace and perfect the role of road aids that have existed so far. So the final project, entitled "Aids for Disabled People to Detect Obstacles", was made by customizing existing blind sticks by integrating sensor technology and sound output on existing road aids. The main purpose of this tool is to make it easier for blind people to detect obstacles that are around when they are walking. This tool uses an ultrasonic sensor as the main sensor, Arduino uno as the main control, and the DFPlayer Mini module as an arduino supporting module as the output that emits sound. From the results obtained this tool works in accordance with what is expected

Keywords: Blind, Assistance, Obstacles

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat ALLAH SWT atas berkat rahmat dan hidayah-Nya jualah, sehingga pada akhirnya penulis dapat menyelesaikan makalah proyek akhir ini dengan baik.

Makalah Proyek Akhir ini disusun sebagai salah satu persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program pendidikan DIlpoma III di Politeknik Manufaktur Negeri Bangka Belitung.

Penulis mencoba untuk menerapkan ilmu pengetahuan yang telah didapatkan selama 3 tahun mengecap pendidikan di Politeknik Manufaktur Negeri Bangka Belitung dan pengalaman yang penulis dapatkan selama melaksanakan Program Kerja Lapangan pada pembuatan alat dan makalah Proyek Akhir ini.

Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada orang-orang yang telah berperan sehingga dapat terselesaikannya Proyek Akhir ini, sebagai berikut :

1. Keluarga besar yang selalu senantiasa memberikan kasih sayang, doa, dukungan moril maupun materi dan semangat.
2. Bapak Yudhi M.T. selaku pembimbing I yang telah meluangkan banyak waktu, memberi saran-saran dan solusi dari masalah-masalah yang penulis hadapi selama proses penyusunan Makalah Proyek Akhir ini dan Bapak Surojo M.T. selaku pembimbing II yang telah meluangkan banyak tenaga, dan pikiran di dalam memberikan pengarahan dalam penulisan makalah Proyek Akhir ini.
3. Bapak Sugeng Ariyono, B.Eng, M.Eng, Ph.D selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
4. Seluruh staf pengajar di Politeknik Manufaktur Negeri Bangka Belitung.
5. Rekan-rekan mahasiswa Politeknik Manufaktur Negeri Bangka Belitung yang telah banyak membantu selama menyelesaikan Proyek Akhir.
6. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung dalam pembuatan Proyek Akhir ini yang tidak dapat disebutkan satu per satu.

Penulis menyadari bahwa penulisan makalah Proyek Akhir ini masih jauh dari sempurna dikarenakan penulis adalah manusia biasa yang tidak luput dari kesalahan. Karena yang benar hanya datang dari ALLAH dan yang salah datang dari penulis sendiri. Oleh karena itu, sangat diharapkan segala petunjuk, kritik, dan saran yang membangun dari pembaca agar dapat menunjang pengembangan dan perbaikan penulisan selanjutnya. Akhir kata penulis mohon maaf atas kekurangan dalam penulisan makalah Proyek Akhir ini dan penulis dengan senang hati menerima saran dan kritik yang membangun dari pembaca.

Semoga proyek akhir ini dapat berguna untuk menambah wawasan dan wacana bagi rekan-rekan mahasiswa.

Sungailiat, 6 september 2019

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	i
PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK.....	iv
<i>ABSTRACT</i>	v
DAFTAR ISI.....	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN.....	1
1.2. Rumusan Masalah dan Batasan Masalah.....	2
1.2.1 Rumusan Masalah.....	2
1.2.2 Batasan Masalah.....	2
1.3. Tujuan	2
BAB II DASAR TEORI.....	3
2.1. Tongkat Tunanetra.....	3
2.2. Komunikasi Data Serial	4
2.3. Arduino UNO 32	5
2.3.1. Daya(power).....	6
2.4. Sensor Ultrasonic HC-SR04	7
2.5. Modul DF Player Mini.....	8
2.6. Proximity infrared Sensor	10
2.7. SD Card.....	10

BAB III METODE PELAKSANAAN.....	12
3.1. Diagram Metode Pelaksanaan	12
3.2. Konsultasi.....	14
3.3. Pengumpulan Data.....	14
3.4. Perancangan <i>Design</i> Alat.....	14
3.5. Pengerjaan Alat.....	15
BAB IV PEMBAHASAN	16
4.1. Pengerjaan Kontruksi Alat	16
4.2. Pengerjaan Kontrol Alat	17
4.3. Uji Coba.....	17
4.3.1. Pengujian Sensor Ultrasonic	17
4.3.2. Pengujian Proximity Switch E18 D80NK	18
BAB V KESIMPULAN DAN SARAN	20
5.1. Kesimpulan	20
5.2. Saran	20
Daftar Pustaka	21
LAMPIRAN.....	22

DAFTAR TABEL

Tabel 2. 1 Perintah Pengaturan Koneksi Arduino.....	4
Tabel 2. 2 Modul DfPlayer Mini	9
Tabel 4. 1 Data hasil Pengujian Sensor Ultrasonic HC-SR04.....	18
Tabel 4. 2 Data hasil Pengujian Proximity Switch E18 D80NK	19

DAFTAR GAMBAR

Gambar 2. 1 Tongkat tuna netra.....	3
Gambar 2. 2 <i>Board Arduino Uno</i>	6
Gambar 2. 3. Sensor Ultrasonic HC-SR04	8
Gambar 2. 4. Modul Dfplayer Mini	8
Gambar 2. 5. <i>Proxymity infrared</i> Sensor.....	10
Gambar 2. 6. <i>SD Card</i>	11
Gambar 4. 1. Kontruksi alat yang telah selesai.....	16
Gambar 4. 3. Skema kontrol Alat.....	17

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Riwayat Hidup

Lampiran 2 : Data Sheet Arduino Uno

Lampiran 3 : Data Sheet Sensor Ultrasonik

Lampiran 4 : Data Sheet Modul DFPayer mini

Lampiran 5 : Data Sheet Sensor E18-D80NK

Lampiran 6 : Progam Keseluruhan

BAB I

PENDAHULUAN

1.1. Latar Belakang

Indera penglihatan adalah salah satu sumber informasi vital bagi manusia. Tidak berlebihan apabila dikemukakan bahwa sebagian besar informasi yang diperoleh oleh manusia berasal dari indera penglihatan, sedangkan selebihnya berasal dari panca indera yang lain.

Tunanetra adalah istilah umum yang digunakan untuk kondisi seseorang yang mengalami gangguan atau hambatan dalam indera penglihatannya. Berdasarkan tingkat gangguannya tunanetra dibagi dua yaitu buta total (*total blind*) dan yang masih mempunyai sisa penglihatan (*Low Vision*). Akibat hilang/berkurangnya fungsi indera penglihatannya maka tunanetra berusaha memaksimalkan fungsi indera-indra yang lainnya seperti, perabaan, penciuman, pendengaran, dan lain sebagainya sehingga tidak sedikit penyandang tunanetra yang memiliki kemampuan luar biasa misalnya di bidang musik atau ilmu pengetahuan

Dengan demikian, dapat dipahami bila seseorang mengalami gangguan pada indera penglihatan, maka kemampuan aktifitasnya akan jadi sangat terbatas, karena informasi yang diperoleh akan jauh berkurang dibandingkan mereka yang berpenglihatan normal.

Pada umumnya, penyandang tuna netra menggunakan alat bantu jalan berupa tongkat putih atau anjing terlatih untuk membantu pergerakan dan meningkatkan keamanan dan kemandirian pada saat berjalan. Dengan mempunyai informasi yang cukup terhadap jalur perjalanan yang akan dilewati, penyandang tuna netra dapat lebih nyaman untuk bernavigasi pada lingkungan yang dikenal.

Untuk mengatasi permasalahan tersebut, maka dibuatlah suatu sistem yang dapat menggantikan dan menyempurnakan peran alat bantu jalan yang selama ini telah ada. Sebagai contoh alat tersebut adalah tongkat pintar yang dapat mendeteksi benda yang ada di depannya yang menjadi hambatan bagi para penyandang tuna netra [1].

1.2.Rumusan Masalah dan Batasan Masalah

1.2.1 Rumusan Masalah

Dalam pembuatan tugas akhir “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan” muncul beberapa rumusan masalah sebagai berikut:

- a. Bagaiman cara membuat “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan” ?
- b. Apa tujuan dari pembuatan “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan” ?
- c. Apa Fungsi dari “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan” ?

1.2.2 Batasan Masalah

Batasan masalah dalam pembuatan tugas akhir “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan” yaitu :

- a. Benda yang terdeteksi adalah benda yang dapat memantulkan gelombang ultrasonik
- b. Diperuntukkan bagi penyandang Tunanetra yang bisa mendengar (tidak tuli)
- c. Tongkat tidak bisa dilipat seperti kondisi awal

1.3.Tujuan

Tujuan dari pembuatan Proyek Akhir “Alat Bantu Jalan Penyandang Tunanetra Untuk Mendeteksi Hambatan” ini sebagai berikut:

- a. Merancang dan membuat alat bantu jalan bagi penyandang tunanetra yang menggunakan teknologi sensor dengan keluaran yang berupa suara.
- b. Membantu Para penyandang tunanetra. untuk lebih dalam mendeteksi hambatan di jalan yang akan di lewatinya

BAB II

DASAR TEORI

Dasar teori ini diperoleh melalui informasi dan referensi yang didapatkan melalui internet dan buku. Teori-teori yang digunakan dalam menyelesaikan proyek akhir ini, antara lain sebagai berikut:

2.1. Tongkat Tunanetra

Tongkat Tunanetra yang dikenal dengan istilah *The White Cane* memiliki jenis dan ukuran yang berbeda-beda. Berdasarkan penggunaannya, tongkat Tunanetra dibedakan menjadi dua macam, tongkat untuk anak dan tongkat untuk dewasa.

Tongkat untuk anak-anak Tunanetra memiliki panjang kurang dari 100 cm. Sedangkan tongkat untuk orang dewasa memiliki panjang antara 110 – 120 cm. Bahkan, Tunanetra di negara-negara barat menggunakan tongkat yang panjangnya di atas 120 cm. Semakin panjang tongkat, maka semakin banyak lipatannya. Tongkat dengan ukuran standart, 110 cm biasanya memiliki empat lipatan. Sementara itu, yang berukuran di atas 120 cm biasanya memiliki lima lipatan [2].



Gambar 2. 1 Tongkat tuna netraa

2.2.Komunikasi Data Serial

Komunikasi data serial merupakan komunikasi data dengan pengiriman data satu persatu pada satuan waktu. Transmisi data pada komunikasi serial dilakukan per bit. Komunikasi serial membutuhkan dua jalur yaitu tansmisit (Tx) dan receive (Rx). Komunikasi serial memiliki dua mode yaitu: sinkron dan asinkron. Mode sinkron proses pengiriman data bersamaan dengan sinyal *clock*, sehingga pengiriman satu karakter dengan karakter lainnya memiliki jeda waktu yang sama. Sedangkan mode asinkron dimana proses pengiriman data tanpa sinyal *clock*. Transmitter yang mengirim data harus menyepakati standar Universal Asynchronous Receive Transmit (UART) Atmega 328 menyediakan serial komunikasi UART TTL (5V). Software arduino mencakup sebuah serial monitor yang memungkinkan data tekstual terkirim ke dan dari *board* arduino.

Dalam pengaturan UART untuk mengkoneksikan arduino dengan perangkat lain terdapat perintah-perintah:

Tabel 2. 1 Perintah Pengaturan Koneksi Arduino

No	Perintah	Fungsi
1	Start Bit	penanda awal dimana akan dilakukan suatu proses pengiriman bit data.
2	Data Bit	merupakan data yang dikirim.
3	Parity Bit	berfungsi sebagai “ <i>flag</i> ” atau bisa dikatakan sebagai penanda.’
4	Stop Bit	sebagai penanda proses pengiriman bit data telah selesai.
5	Bit Rate	menjumlah dari bit yang terkirim atau diterima per satuan waktu (<i>second</i>).
6	Baud Rate	banyaknya perubahan data yang terjadi per satuan waktu.

Perintah terakhir yang juga dibutuhkan dalam menentukan pembacaan nomor port yang terhubung dengan PC yang dikenal dengan *Serial Communication Port Number*. Port USB (*Universal Serial Bus*) yang ada pada arduino adalah hardware yang merupakan port masukan/keluaran baru yang dibuat untuk mengatsai kekurangan-kekurangan port serial maupun paralel yang sudah ada. USB digunakan sebagai port komunikasi serial antarmuka (*interfaces*) apapun [3].

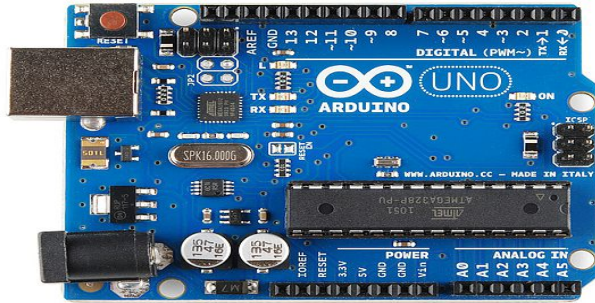
2.3.Arduino UNO 32

Arduino Uno adalah board mikrokontroler berbasis ATmega328. Memiliki 14 pin input dari output digital dimana 6 pin input tersebut dapat digunakan sebagai output PWM dan 6 pin input analog, 16 MHz osilator kristal, koneksi USB, jack power, ICSP header, dan tombol reset. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board* Arduino Uno ke computer dengan menggunakan kabel USB atau listrik dengan AC yang keadaptor-DC atau baterai untuk menjalankannya.

Uno berbeda dengan semua board sebelumnya dalam hal koneksi USB-to-serial yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter USB-to-serial berbeda dengan board sebelumnya yang menggunakan chip FTDI driver USB-to-serial.

Arduino Uno 328 memiliki fitur-fitur baru sebagai berikut:

- Pin out : menambahkan SDA dan SCL *pin* yang dekat ke *pin* aref dan dua *pin* baru lainnya ditempatkan dekat ke *pin* RESET, dengan I/O REF yang memungkinkan sebagai *buffer* untuk beradaptasi dengan tegangan yang disediakan dari *board* sistem. Pengembangannya, sistem akan lebih kompatibel dengan *prosesor* yang menggunakan AVR, yang beroperasi dengan 5V dan dengan *Arduino* karena beroperasi dengan 3,3V. Yang kedua adalah *pin* yang tidak terhubung, yang disediakan untuk tujuan pengembangannya.
- Sirkuit RESET.
- Chip Atmega 16U2 menggantikan chip Atmega 8U2 [4].



Gambar 2. 2 *Board Arduino Uno*

2.3.1. Daya(power)

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Sumber daya eksternal (non-USB) dapat berasal baik dari adaptor AC-DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan steker 2.2mm yang bagian tegahnya terminal positif ke *jack* sumber tegangan pada papan. Jika tegangan berasal dari baterai dapat langsung dihubungkan melalui header pin Gnd dan pin Vin dari konektor *Power*. Papan Arduino Uno 328 dapat beroperasi dengan pasokan daya eksternal 6 Volt sampai 20 Volt. Jika diberi tegangan kurang dari 7 Volt, maka pin 5 Volt mungkin akan menghasilkan tegangan kurang dari 5 Volt dan ini akan membuat papan menjadi tidak stabil. Jika sumber tegangan menggunakan lebih dari 12 Volt, regulator tegangan akan mengalami panas berlebihan dan bisa merusak papan. Rentang sumber tegangan yang dianjurkan adalah 7 Volt sampai dengan 12 Volt. Pin tegangan yang tersedia pada papan Arduino adalah sebagai berikut:

- Vin adalah *input* tegangan untuk papan Arduino ketika menggunakan sumber daya eksternal (sebagai “saingan” tegangan 5 Volt dari koneksi USB atau sumber daya *ter-regulator* lainnya). anda dapat memberikan tegangan melalui pin ini, atau jika memasok tegangan untuk papan melalui *jack power*, kita bisa mengakses/mengambil tegangan melauai pin ini.
- 5V: Sebuah pin yang mengeluarkan tegangan *ter-regulator* 5 Volt, dari pin ini tegangan sudah diatur (*ter-regulator*) dari *regulator* yang tersedia (*built-in*) pada papan. Arduino dapat diaktifkan dengan sumber daya, baik berasal dari *jack power* DC (7-12 Volt), konektor USB (5 Volt), atau pin Vin pada *board* (7-12 Volt).

Memberikan tegangan melalui pin 5V atau 3.3 Volt secara langsung tanpa melewati *regulator* dapat merusak papan Arduino.

- 3V3: Sebuah pin yang menghasilkan tegangan 3.3 Volt. Tegangan ini dihasilkan oleh *regulator* yang terdapat pada papan (*on-board*). Arus maksimum yang dihasilkan adalah 50Ma.
- GND: Pin *Ground* atau *Massa*.
- IOREF: Pin ini pada papan Arduino berfungsi untuk memberikan referensi tegangan yang beroperasi pada *microcontroller*, sebuah perisai (*shield*) dikonfigurasi dengan benar untuk dapat membaca pin tegangan IOREF dan memiliki sumber daya yang tepat atau mengaktifkan penerjemah tegangan (*voltage transistor*) pada *output* untuk bekerja pada tegangan 5 Volt atau 3.3 Volt [5].

2.4. Sensor Ultrasonic HC-SR04

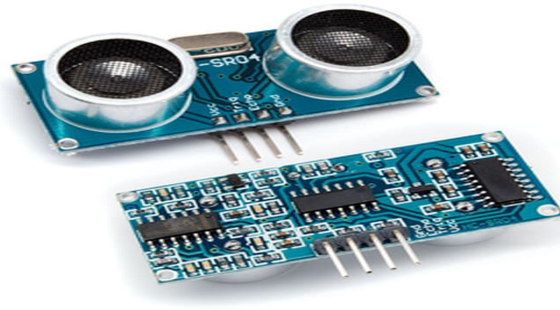
Sensor HC-SR04 adalah sensor pengukur jarak berbasis gelombang ultrasonik. Prinsip kerja sensor ini mirip dengan radar ultrasonik. Gelombang ultrasonik di pancarkan kemudian di terima balik oleh receiver ultrasonik. Jarak antara waktu pancar dan waktu terima adalah representasi dari jarak objek. Sensor ini cocok untuk aplikasi elektronik yang memerlukan deteksi jarak termasuk untuk sensor pada robot.

Sensor HC-SR04 adalah versi *low cost* dari sensor ultrasonik PING buatan parallax. Perbedaannya terletak pada pin yang digunakan. HC-SR04 menggunakan 4 pin sedangkan PING buatan parallax menggunakan 3 pin. Pada Sensor HC-SR04 pin trigger dan output diletakkan terpisah. Sedangkan jika menggunakan PING dari Parallax pin trigger dan output telah diset *default* menjadi satu jalur. Tidak ada perbedaan signifikan dalam pengimplementasiannya. Jangkauan karak sensor lebih jauh dari PING buatan parallax, dimana jika ping buatan parallax hanya mempunyai jarak jangkauan maksimal 350 cm sedangkan sensor HC-SR04 mempunyai kisaran jangkauan maksimal 400-500cm [5]

Spesifikasi:

- Jangkauan deteksi: 2cm sampai kisaran 400 -500cm
- Sudut deteksi terbaik adalah 15 derajat
- Tegangan kerja 5V DC

- Resolusi 1cm
- Frekuensi Ultrasonik 40 kHz
- Dapat dihubungkan langsung ke kaki mikrokontroler [6].



Gambar 2. 3. Sensor Ultrasonic HC-SR04

2.5. Modul DF Player Mini

Modul DFPlayer Mini merupakan sebuah modul MP3 serial yang menyediakan kesempurnaan integrasi MP3, WMV hardware decoding. Sedangkan softwarenya mendukung driver TF Card, mendukung sistem file FAT16, FAT32. Melalui perintah-perintah serial sederhana untuk menentukan memutar musik, serta bagaimana cara memutar musik dan fungsi lainnya, tidak melalui operasi yang rumit, mudah digunakan, stabil dan dapat diandalkan adalah fitur-fitur yang paling penting dari modul ini [7].



Gambar 2. 4. Modul Dfplayer Mini

Adapun spesifikasi yang dimiliki oleh modul DFPlayer Mini terdapat pada Tabel 2.2

Tabel 2. 2 Modul DfPlayer Mini

Pin	Nama	Keterangan
1	VCC	Input Voltage 3.2 V – 5 V
2	RX	UART Serial Input
3	TX	UART Serial Output
4	DAC_R	Audio Output Right Channel
5	DAC_L	Audio Output Left Channel
6	SPK2	Speaker 2
7	GND	Power Ground
8	SPK1	Speaker 1
9	IO1	Trigger Port 1
10	GND	Power Ground
11	IO2	Trigger Port 2
12	ADKEY1	Port 1
13	ADKEY2	Port 2
14	USB+	USB+ DP
15	USB-	USB-DM
16	Busy	Playing Status

2.6. Proximity infrared Sensor

Sensor ini merupakan sensor yang mendeteksi benda dengan memanfaatkan cahaya sebagai alat pendeteksinya, cahaya yang digunakan dalam sensor ini yaitu cahaya *Infra red*. sensor ini terdiri dari sebuah cahaya dan penerima (*Receptor*). jika benda yang lewat mengenai cahaya dari sensor, maka cahaya akan memantulkan kembali ke *Receptor* dan mengidentifikasi adanya benda yang tertangkap oleh sensor. Sensor ini tidak mengenal jenis bahan pada benda



Gambar 2. 5. Proximity infrared Sensor

2.7. SD Card

SD(*Secure Digital*) Card adalah kartu memori *non-volatile* yang dikembangkan oleh SD Card Association yang digunakan dalam perangkat *portable*. Saat ini, teknologi micro SD sudah digunakan oleh lebih dari 400 merek produk serta dianggap sebagai standar. Keluarga SD Card yang lain terbagi SDSC yang kapasitas maksimum resminya sekitar 2 GB, meskipun beberapa ada yang sampai 4GB. SDHC (*High Capacity*) memiliki kapasitas dari 4GB sampai 32GB. Dan SDXC (*Extended Capacity*) kapasitasnya sering kali membuat kebingungan karena masing-masing protokol komunikasi sedikit berbeda. Dari sudut pandang perangkat, semua kartu ini termasuk dalam keluarga SD. SD adapter memungkinkan konversi fisik kartu SD yang lebih kecil untuk bekerja dislot fisik yang lebih

besar pada dasarnya ini adalah alat pasif yang menghubungkan pin dari SD Card yang kecil ke pin adaptor SD Card yang lebih besar [8].



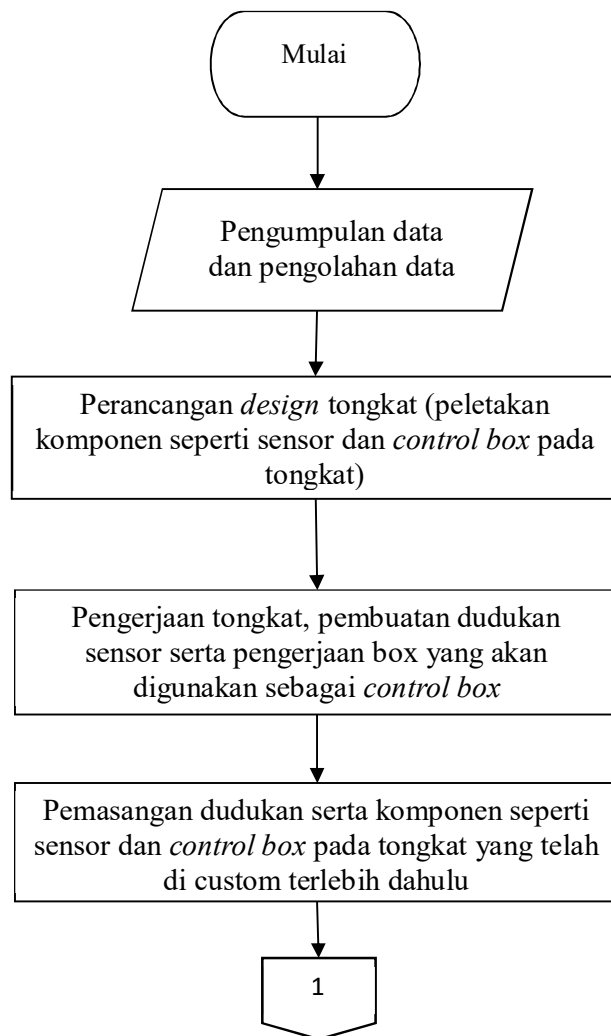
Gambar 2. 6. SD Card

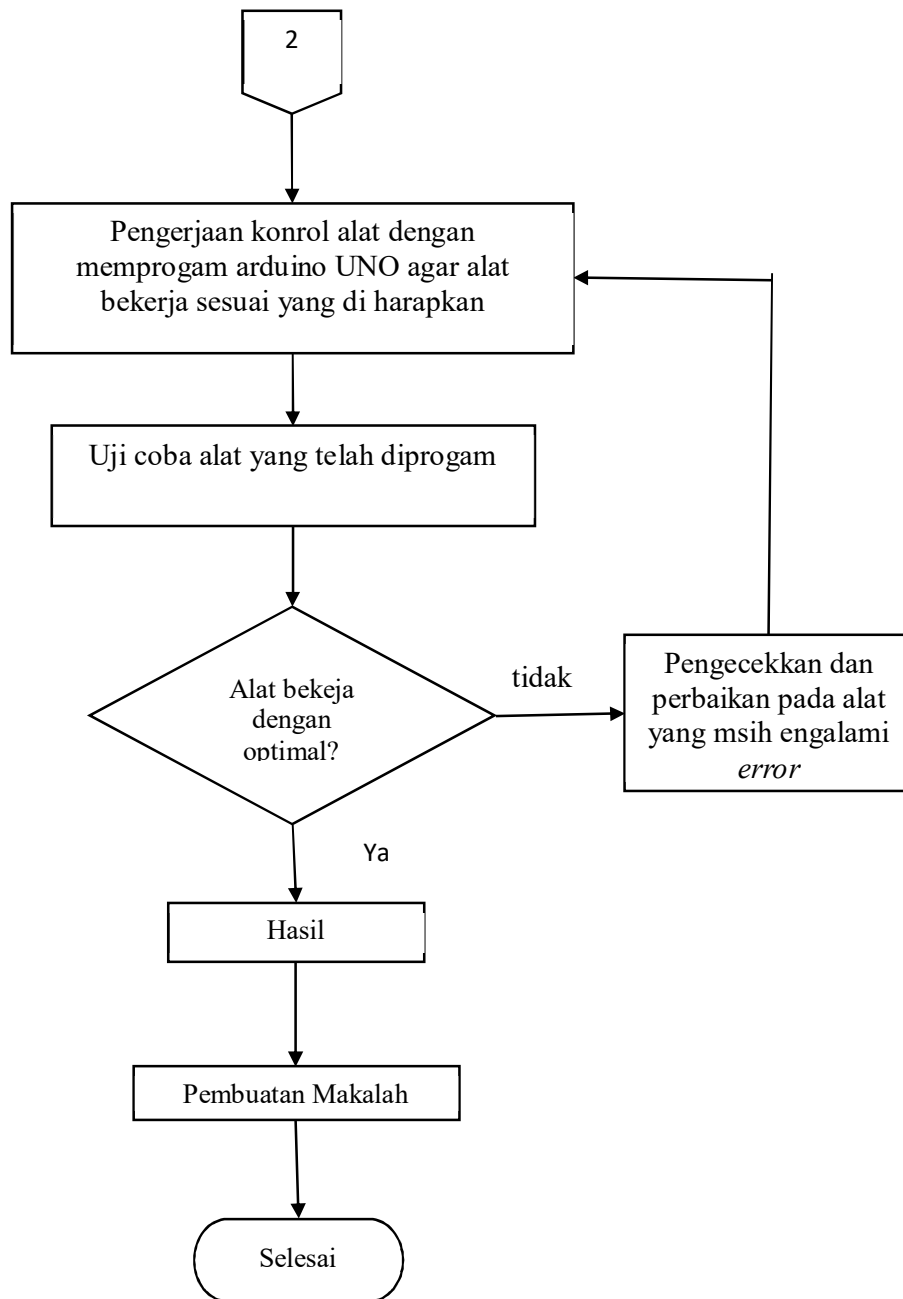
BAB III

METODE PELAKSANAAN

3.1. Diagram Metode Pelaksanaan

Dalam pelaksanaan proyek akhir ini, dilakukan tahap-tahap yang bertujuan untuk mempermudah pembuatan proyek akhir. Gambar 3.1 berikut ini adalah diagram alir tahapan-tahapan yang dilakukan.





Tabel 3. 1. Diagram Metode Pelaksanaan

3.2. Konsultasi

Konsultasi dalam hal ini adalah konsultasi kepada dosen pembimbing untuk alat tersebut sehingga di dapatkan hasil yang maksimal dan juga agar dosen pembimbing pun dapat membantu mencari solusi jika terdapat kesulitan atau permasalahan pada alat yang akan dikerjakan.

3.3. Pengumpulan Data

Pengumpulan data dapat dilakukan untuk mencari data yang akan mendukung Penelitian. Pengumpulan data dilakukan dengan 2 metode, yaitu :

1. Metode langsung

Metode langsung adalah data yang didapatkan secara langsung dari objek penelitian. Metode ini dilakukan dengan cara bimbingan / konsultasi. Bimbingan / konsultasi yang dilakukan dengan pembimbing dan pihak-pihak lain agar tujuan yang diterapkan dapat tercapai.

2. Metode Tidak Langsung

Metode tidak langsung adalah data yang didapatkan secara tidak langsung dari objek penelitian. Metode ini dilakukan dengan mengambil data dari berbagai sumber yang berasal dari buku-buku referensi, jurnal. Serta internet.

3.4. Perancangan *Design* Alat

Pembuatan desain alat ini meliputi pembuatan rancangan keseluruhan tongkat dengan menggunakan aplikasi "*Skech Up*" perancangan design itu sendiri meliputi, perancangan tata letak komponen komponen yang digunakan pada tongkat. Hal ini juga termasuk untuk merancang kontrol box yang akan digunakan sebagai tempat untuk meletakkan komponen seperti arduino, modul suara, speaker yang diletakkan di dalam box, dan baterai yang diletakkan pada bagian luar box. Sedangkan sensor dan roda akan diletakkan langsung pada bagian tongkat.

Berikut ini merupakan gambar dari desain alat secara keseluruhan yang telah dibuat menggunakan aplikasi *Skech Up*.



Gambar 3. 1. *Design* alat

3.5. Pengerjaan Alat

Proses ini meliputi pengerjaan rangkaian kontrol alat yang akan di letakan di dalam *control box*, rangkaian kontrol DFPlayer mini, dan pemasangan komponen pada tongkat sesuai dengan *design* yang telah dibuat dan juga proses pemrograman yang berfungsi untuk mengontrol alat tersebut agar dapat menjalankan fungsinya dengan baik. Pemrograman dilakukan dengan menggunakan *software ARDUINO UNO*. Pemrograman yang dilakukan meliputi pemrograman alat agar dapat melakna pendeteksian benda yang ada di sekitar serta pemrogaman output suara yang di hasilkan dari DFPlayer mini.

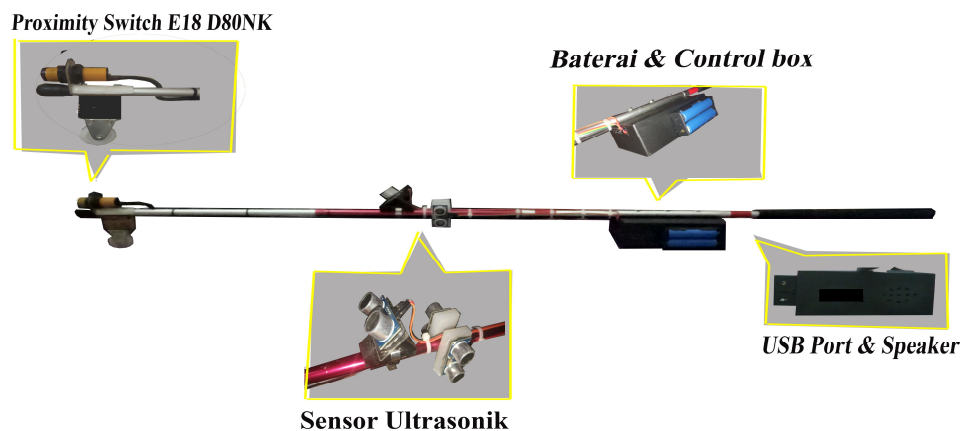
BAB IV PEMBAHASAN

Pada bab ini akan dijelaskan mengenai proses pengerjaan “ALAT BANTU JALAN PENYANDANG TUNANETRA UNTUK MENDETEKSI HAMBATAN” sehingga akhirnya dapat bekerja dengan baik dan siap untuk digunakan. Dalam proses ini terdapat beberapa tahap pengerjaan alat yaitu :

1. Pengerjaan kontruksi alat
2. Pengerjaan kontrol alat
3. Uji coba

4.1. Pengerjaan Kontruksi Alat

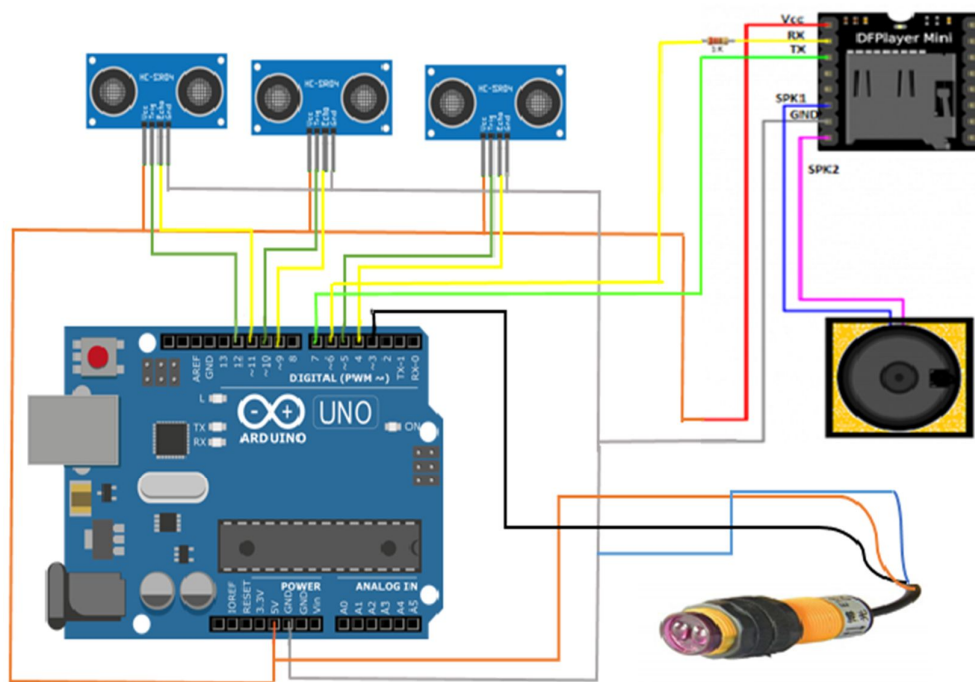
Pengerjaan kontruksi untuk alat ini tidak banyak dikarenakan setiap komponen yang diperlukan bisa didapatkan dalam bentuk telah jadi. Pengerjaan kontruksi ini meliputi pembuatan serta pemasanganudukan sensor sesuai desain, pemasangan sensor pada dudukan sensor, dan pemasangan *control box* pada tongkat. Kontruksi alat yang telah selesai dikerjakan di tampilkan pada gambar 4.1



Gambar 4. 1. Kontruksi alat yang telah selesai

4.2. Pengerjaan Kontrol Alat

Pengerjaan kontrol alat dilakukan dengan cara menghubungkan komponen komponen yang ada menggunakan kabel *jumper* dan memprogram arduino agar alat berjalan sesuai dengan yang di harapkan. dengan susunan rangkaian seperti gambar di bawah ini



Gambar 4. 2. Skema kontrol Alat

4.3. Uji Coba

4.3.1. Pengujian Sensor Ultrasonic

Pengujian sensor *ultrasonic HC-SR04* dilakukan dengan membandingkan jarak yang di tampilkan pada serial monitor dengan jarak yang di ukur menggunakan mistar/penggaris dan bisa juga menggunakan meteran.

Berdasarkan hasil pengujian terhadap sensor ultrasonik didapat data seperti tabel dibawah ini

Tabel 4. 1 Data hasil Pengujian Sensor *Ultrasonic* HC-SR04

No	Jarak	Sensor 1	Sensor 2	Sensor 3	Error %
1	10 cm	10 cm	10 cm	10 cm	0%
2	20 cm	20 cm	20 cm	20 cm	0%
3	30 cm	30 cm	30 cm	30 cm	0%
4	40 cm	40 cm	40 cm	40 cm	0%
5	50 cm	50 cm	50 cm	50 cm	0%
6	60 cm	60 cm	60 cm	60 cm	0%
7	70 cm	70 cm	70 cm	70 cm	0%
8	80 cm	81 cm	81 cm	81 cm	1,01%
9	90 cm	91 cm	91 cm	91 cm	1,01%
10	100 cm	102 cm	102 cm	102 cm	1,02%

Hasil pengujian sensor dengan jarak 10-100 cm yang dapat kita lihat pada tabel 4.1 yang merupakan hasil percobaan dari sensor ultrasonik HC-SR04 yang masih stabil saat 70 cm dan hasil pengukuran tidak mengalami *error* akan tetapi ketika memasuki jarak 80 cm *error* sudah mulai terlihat akan tetapi nilai rentang 71-81 cm *error* rata-rata 1, 01% namun hal tersebut masih dikatakan akurat dikarenakan toleransi peneliti 1 cm dan pada pengujian jarak 72-100 cm *error* rata-rata 1,66% dengan toleransi pengujian 1 cm.

4.3.2. Pengujian Proximity Switch E18 D80NK

Pengujian *Proximity Switch E18 D80NK* dilakukan dengan mengukur jangkauan deteksi benda menggunakan penggaris /mistar. Sebelumnya sensor telah di atur dengan jangkauan deteksi terendah. Dari pengujian tersebut di dapat data seperti tabel 4.2

Tabel 4. 2 Data hasil Pengujian Proximity Switch E18 D80NK

No	Jarak deteksi	Status deteksi
1	1 cm	Terdeteksi
2	2 cm	Terdeteksi
3	3 cm	Terdeteksi
4	4 cm	Terdeteksi
5	5 cm	Terdeteksi
6	6 cm	Terdeteksi
7	7 cm	Terdeteksi
8	8 cm	Terdeteksi
9	9 cm	Terdeteksi
10	10 cm	Terdeteksi
11	11 cm	Tidak Terdeteksi

Hasil pengujian sensor dengan jarak 1-10 cm yang dapat kita lihat pada tabel 4.2 yang merupakan hasil percobaan dari *Proximity Switch E18 D80NK* menegaskan pada saat sensor tersebut diatur ke jangkauan deteksi terendah sensor tersebut hanya akan mendeteksi benda hingga jarak 10 cm dan benda yang bejarak 11 cm atau lebih tidak akan terdeteksi oleh sensor tersebut

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mulai dari tahap perancangan hingga uji coba sistem maka dapat ditarik kesimpulan sebagai berikut :

1. Alat bantu jalan penyandang tunanetra untuk mendeteksi hambatan berhasil dibangun sesuai dan berfungsi seperti yang telah dibuat.
2. Penggunaan modul DFPlayer mini memberikan hasil yang baik dan sangat efektif digunakan sebagai modul pendukung arduino untuk data keluaran berupa suara.
3. Dari pengujian yang dilakukan sistem kerja dari setiap rangkaian dan komponen pada alat berfungsi dengan baik.

5.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan diperoleh beberapa hal yang dapat dijadikan saran untuk dilakukan pengembangan lebih lanjut yaitu :

1. Alat ini sebaiknya dikembangkan lagi dengan metode dan teknologi yang lebih baik agar dapat lebih banyak membantu bagi para penyandang tunanetra.
2. Proses pengembangan tidak harus berupa tongkat melainkan benda lain asal kan asih dalam konteks alat bantu jalan bagi penyandang tunanetra.

Daftar Pustaka

- [1] Hallan, D.P., Kauffman, J. M., & Pullen, P. C. (2009). *Exceptional Learners* (11th ed). USA: Pearson Education, Inc.
- [2] R.Kustiani, "Tempo.co," [Online]. Available: <https://difabel.tempo.co/read/1107384/mengenal-the-white-cane-tongkat-tunanetra-dan-aneka-jenisnya/full&view=ok>.
- [3] M. Risdianti dan N. Agustina, "Trainer Kit Piranti Elektronika Menggunakan Instruksi Suara," Sungailiat, 2018.
- [4] "Arduino," Arduino, [Online]. Available: <http://arduino.cc/en/Main/ArduinoBoardUno/>.
- [5] F. Santosa, "Arduino Uno," 14 5 2013. [Online]. Available: <https://febriadisantosa.weebly.com/knowledge/arduino-uno>.
- [6] G.INDRA, "Sensor Ultrasonik," DIGI BYTES GROUP, [Online]. Available: http://www.digibytes.com/index.php?route=product/product&product_id=96.
- [7] D.Ardianto, "belajarArduino.," juli2016. [Online]. Available: <http://www.belajarduino.com/2016/07/dfplayer-mini-serial-mp3-player-module.html>.
- [8] Machamp, "championsoffire," 25122018. [Online]. Available: <https://www.championsoffire.com/pengertian-kartu-sd-beserta-fungsinya/>.

LAMPIRAN 1
Daftar Riwayat Hidup

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Leo Mahesa Pratama
Tempat & Tanggal Lahir : Tanjung Enim, 21 September 1998
Alamat : Sungailiat
Hp.0895604253729
Email : leomahesa21@gmail.com
Status : Mahasiswa



2. Riwayat Pendidikan

SD Negeri 15 Pemali	Lulus 2010
SMP Negeri 1 Pemali	Lulus 2013
SMA Negeri 1 Pemali	Lulus 2016

3. Pengalaman Kerja

Praktik kerja lapangan di PT.VETECH Pedana Tahun 2018

4. Pengetahuan Bahasa : Bahasa Indonesia dan Bahasa Inggris

5. Hobi : Nonton Film

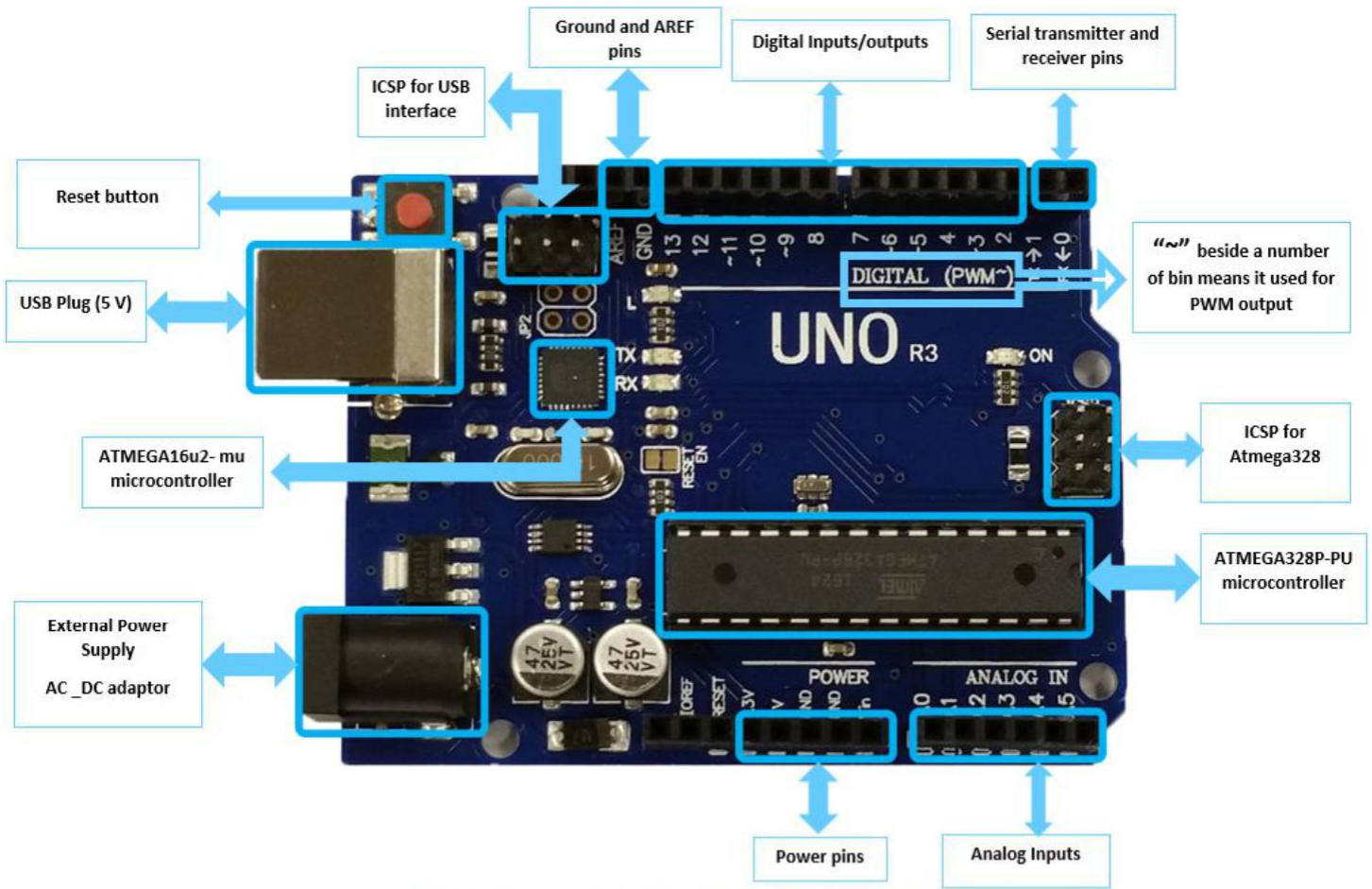
Sungailiat, Juli 2019



Leo Mahesa Pratama

LAMPIRAN 2
Data Sheet Arduino Uno

Arduino Uno R3



INTRODUCTION

Arduino is used for building different types of electronic circuits easily using of both a physical programmable circuit board usually microcontroller and piece of code running on computer with USB connection between the computer and Arduino.

Programming language used in Arduino is just a simplified version of C++ that can easily replace thousands of wires with words.

ARDUINO UNO-R3 PHYSICAL COMPONENTS

ATMEGA328P-PU microcontroller

The most important element in Arduino Uno R3 is ATMEGA328P-PU is an 8-bit Microcontroller with flash memory reach to 32k bytes. It's features as follow:

- High Performance, Low Power AVR
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Up to 20 MIPS Throughput at 20 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory
 - 256/512/512/1K Bytes EEPROM
 - 512/1K/1K/2K Bytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel 10-bit ADC in TQFP and QFN/MLF package
 - Temperature Measurement
 - 6-channel 10-bit ADC in PDIP Package
 - Temperature Measurement
 - Programmable Serial USART



- Master/Slave SPI Serial Interface
- Byte-oriented 2-wire Serial Interface (Philips I2 C compatible)
- Programmable Watchdog Timer with Separate On-chip Oscillator
- On-chip Analog Comparator
- Interrupt and Wake-up on Pin Change

• **Special Microcontroller Features**

- Power-on Reset and Programmable Brown-out Detection
- Internal Calibrated Oscillator
- External and Internal Interrupt Sources
- Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby

• **I/O and Packages**

- 23 Programmable I/O Lines
- 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF

• **Operating Voltage:**

- 1.8 - 5.5V

• **Temperature Range:**

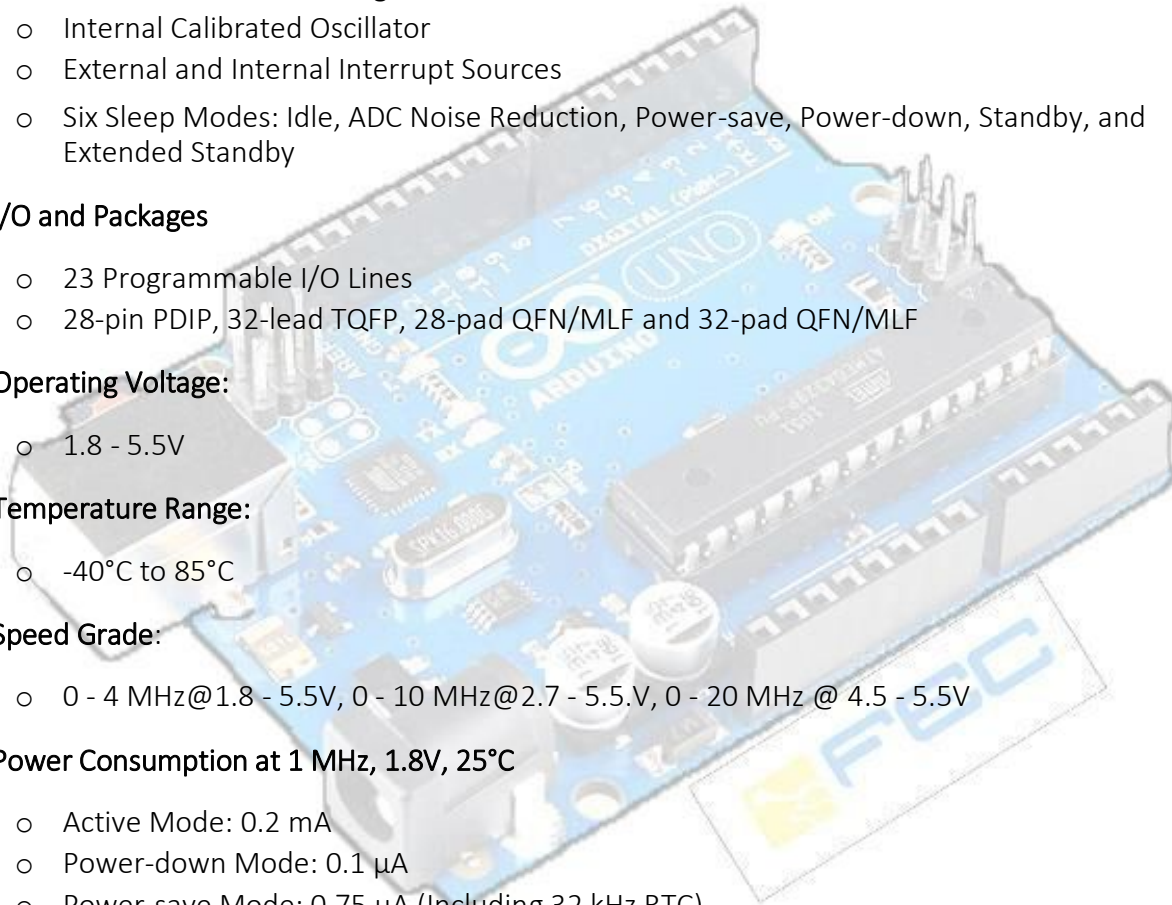
- -40°C to 85°C

• **Speed Grade:**

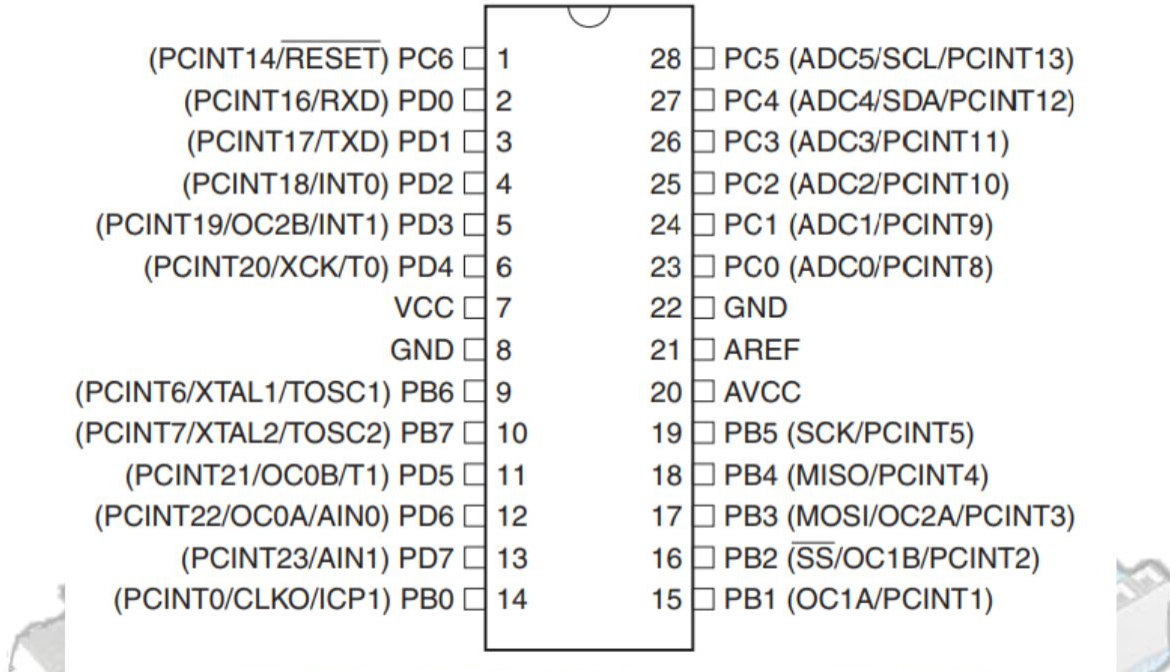
- 0 - 4 MHz@1.8 - 5.5V, 0 - 10 MHz@2.7 - 5.5.V, 0 - 20 MHz @ 4.5 - 5.5V

• **Power Consumption at 1 MHz, 1.8V, 25°C**

- Active Mode: 0.2 mA
- Power-down Mode: 0.1 μ A
- Power-save Mode: 0.75 μ A (Including 32 kHz RTC)



- Pin configuration



ATMEGA16u2- mu microcontroller

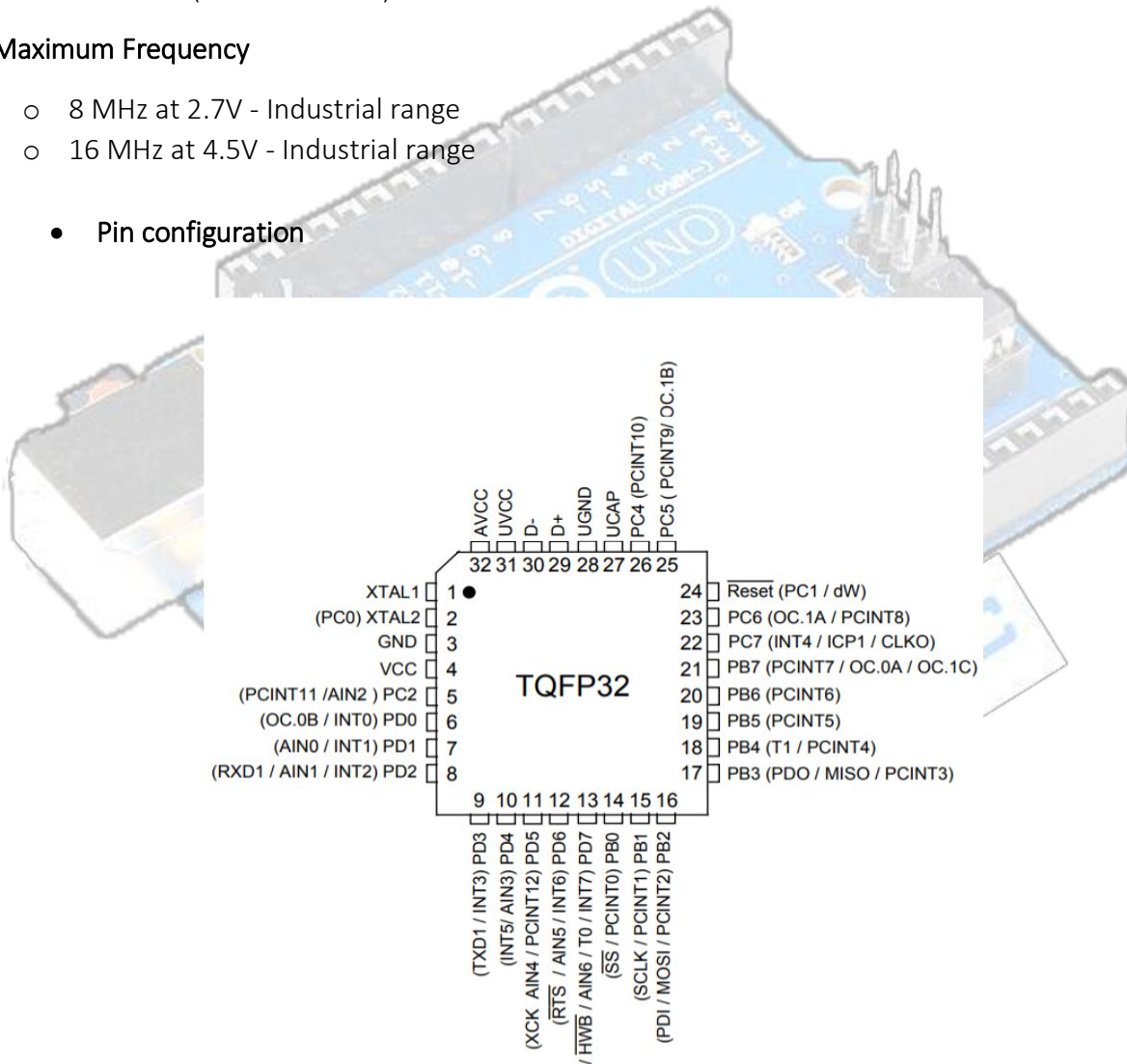
Is a 8-bit microcontroller used as USB driver in Arduino uno R3 it's features as follow:

- High Performance, Low Power AVR
- Advanced RISC Architecture
 - 125 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
- Non-volatile Program and Data Memories
 - 8K/16K/32K Bytes of In-System Self-Programmable Flash
 - 512/512/1024 EEPROM
 - 512/512/1024 Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/ 100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C



- Optional Boot Code Section with Independent Lock Bits
- In-System Programming by on-chip Boot Program hardware-activated after reset
- Programming Lock for Software Security
- **USB 2.0 Full-speed Device Module with Interrupt on Transfer Completion**
 - Complies fully with Universal Serial Bus Specification REV 2.0
 - 48 MHz PLL for Full-speed Bus Operation: data transfer rates at 12 Mbit/s
 - Fully independent 176 bytes USB DPRAM for endpoint memory allocation
 - Endpoint 0 for Control Transfers: from 8 up to 64-bytes
 - 4 Programmable Endpoints:
 - IN or Out Directions
 - Bulk, Interrupt and Isochronous Transfers
 - Programmable maximum packet size from 8 to 64 bytes
 - Programmable single or double buffer
 - Suspend/Resume Interrupts
 - Microcontroller reset on USB Bus Reset without detach
 - USB Bus Disconnection on Microcontroller Request
- **Peripheral Features**
 - One 8-bit Timer/Counters with Separate Prescaler and Compare Mode (two 8-bit PWM channels)
 - One 16-bit Timer/Counter with Separate Prescaler, Compare and Capture Mode(three 8-bit PWM channels)
 - USART with SPI master only mode and hardware flow control (RTS/CTS)
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- **On Chip Debug Interface (debug WIRE)**
- **Special Microcontroller Features**
 - Power-On Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, Power-save, Power-down, Standby, and Extended Standby
- **I/O and Packages**
 - 22 Programmable I/O Lines

- Operating Voltages
 - 2.7 - 5.5V
- Operating temperature
 - Industrial (-40°C to +85°C)
- Maximum Frequency
 - 8 MHz at 2.7V - Industrial range
 - 16 MHz at 4.5V - Industrial range
- Pin configuration



OTHER ARDUINO UNO R3 PARTS

Input and Output

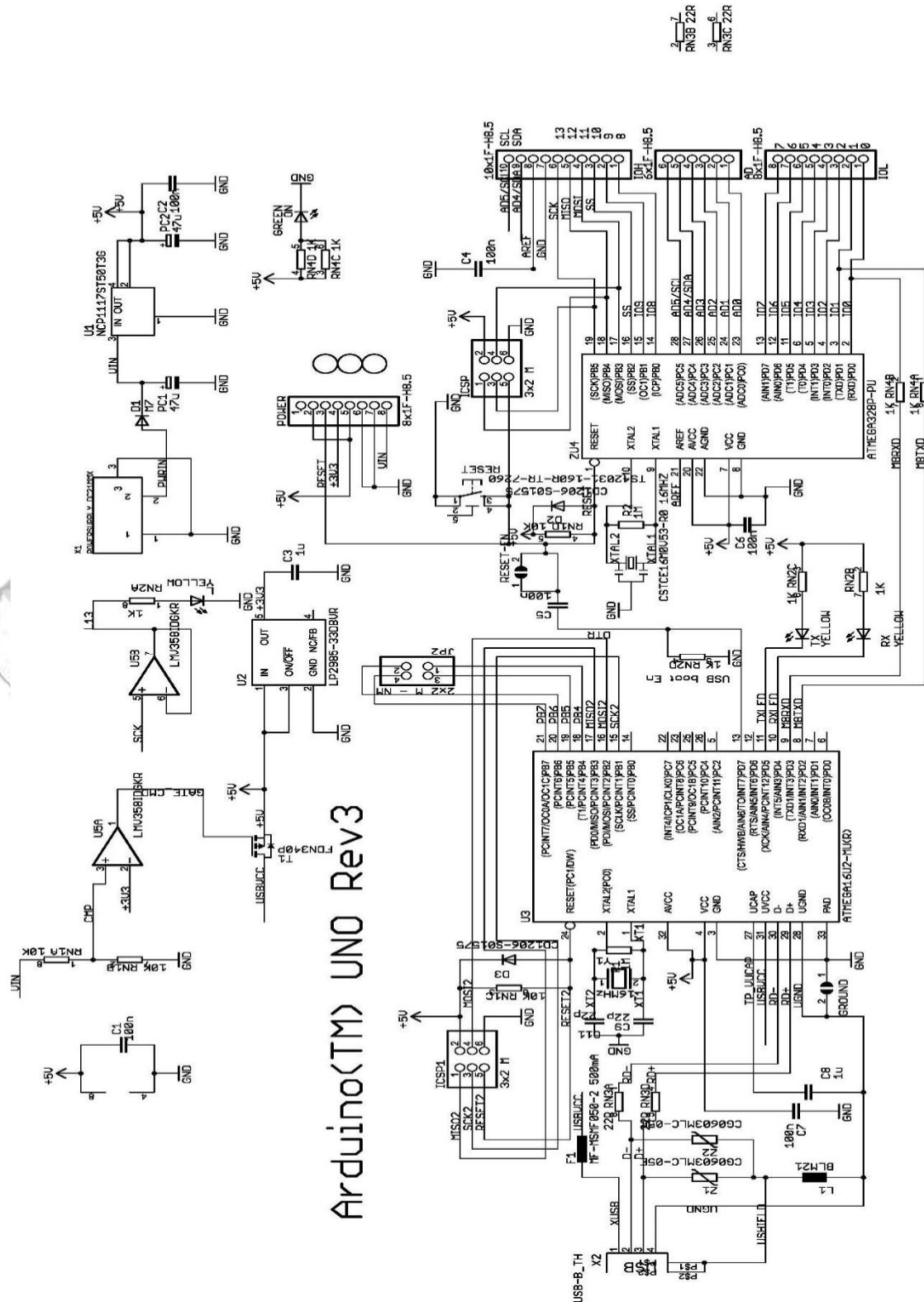
Each of the 14 digital pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 k Ohms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the `analogWrite()` function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.
- AREF: Reference voltage for the analog inputs. Used with `analogReference()`.
- Reset: Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

ARDUINO UNO R3 SCHEMATIC DIAGRAM



LAMPIRAN 3
Data Sheet Sensor Ultrasonic



Tech Support: services@elecfreaks.com

Ultrasonic Ranging Module HC - SR04

Product features:

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- (1) Using IO trigger for at least 10us high level signal,
- (2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
- (3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.

Test distance = (high level time×velocity of sound (340M/S) / 2,

Wire connecting direct as following:

5V Supply

Trigger Pulse Input

Echo Pulse Output

0V Ground

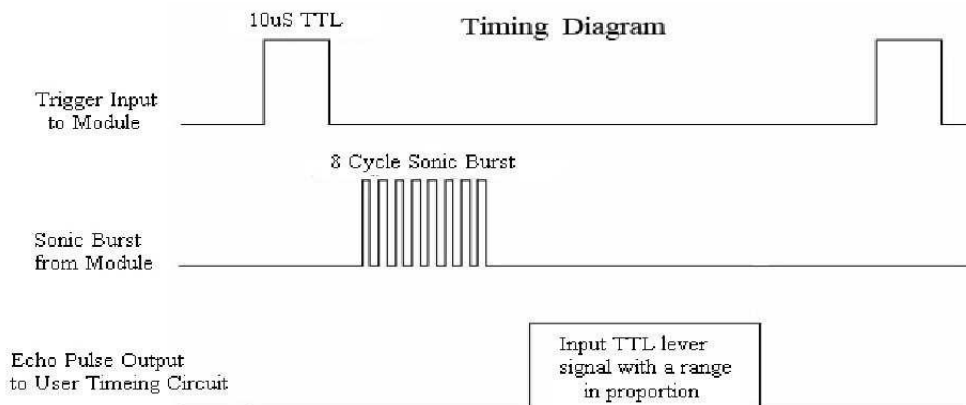
Electric Parameter

Working Voltage	DC5V
Working Current	15mA
Working Frequency	40Hz
Max Range	4m
Min Range	2cm
MeasuringAngle	15 degree
Trigger Input Signal	10uS TTL pulse
Echo Output Signal	Input TTL lever signal and the range in proportion
Dimension	45*20*15mm



Timing diagram

The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion. You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula: $\mu\text{s} / 58 = \text{centimeters}$ or $\mu\text{s} / 148 = \text{inch}$; or: the range = high level time * velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.



Attention:

The module is not suggested to connect directly to electric, if connected electric, the GND terminal should be connected the module first, otherwise, it will affect the normal work of the module.

When tested objects, the range of area is not less than 0.5 square meters and the plane requests as smooth as possible, otherwise ,it will affect the results of measuring.

www.ElecFreaks.com



LAMPIRAN 4
Data Sheet Modul DFPlayer Mini

DFPlayer Mini SKU:DFR0299

From Robot Wiki

Contents

[1 Introduction](#)

[2 Specification](#)

[3 Application](#)

[4 Pin Map](#)

[5 Work Mode](#)

[5.1 1\) Serial Mode](#)

[5.2 2\) AD KEY Mode](#)

▪ [5.3 3\) I/O Mode](#)

▪ [6 Connection Diagram](#)

▪ [7 Sample Code](#)

▪

▪

Introduction

The DFPlayer Mini is a small and low price MP3 module with an simplified output directly to the speaker. The module can be used as a stand alone module with attached battery, speaker and push buttons or used in combination with an Arduino UNO or any other with RX/TX capabilities.

Specification

- supported sampling rates (kHz): 8/11.025/12/16/22.05/24/32/44.1/48
- 24 -bit DAC output, support for dynamic range 90dB , SNR support 85dB
- fully supports FAT16 , FAT32 file system, maximum support 32G of the TF card, support 32G of U disk, 64M bytes
- NORFLASH a variety of control modes, I/O control mode, serial mode, AD button control mode
- advertising sound waiting function, the music can be suspended. when advertising is over in the music
- continue to play audio data sorted by folder, supports up to 100 folders, every folder can hold up to 255
- songs 30 level adjustable volume, 6 -level EQ adjustable

Application

- Car navigation voice broadcast;
- Road transport inspectors, toll stations voice prompts;
- Railway station, bus safety inspection voice prompts;

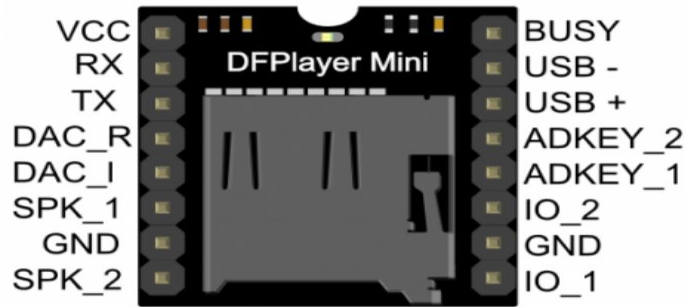


DFPlayer Mini

- Electricity, communications, financial business hall voice prompts;
- Vehicle into and out of the channel verify that the voice prompts;
- The public security border control channel voice prompts;
- Multi-channel voice alarm or equipment operating guide voice;
- The electric tourist car safe driving voice notices;
- Electromechanical equipment failure alarm;
- Fire alarm voice prompts;
- The automatic broadcast equipment, regular broadcast.

Pin Map

http://www.dfrobot.com/wiki/index.php/DFPlayer_Mini_SKU:DFR0299



Pin	Description	Note
VCC	Input Voltage	DC3.2~5.0V;Type: DC4.2V
RX	UART serial input	
TX	UART serial output	
DAC_R	Audio output right channel	Drive earphone and amplifier
DAC_L	Audio output left channel	Drive earphone and amplifier
SPK2	Speaker-	Drive speaker less than 3W
GND	Ground	Power GND
SPK1	Speaker+	Drive speaker less than 3W
IO1	Trigger port 1	Short press to play previous (long press to decrease volume)
GND	Ground	Power GND
IO2	Trigger port 2	Short press to play next (long press to increase volume)
ADKEY1	AD Port 1	Trigger play first segment
ADKEY2	AD Port 2	Trigger play fifth segment
USB+	USB+ DP	USB Port
USB-	USB- DM	USB Port
BUSY	Playing Status	Low means playing \High means no

Work Mode

1) Serial Mode

Support for asynchronous serial communication mode via PC serial sending commands

Communication Standard:9600 bps

Data bits :1

Checkout :none

Flow Control :none

■ Instruction Description

Format:	SS	VER	Len	CMD	Feedback	para1	para2	checksum	\$O
\$S	Start bit 0x7E			Each command feedback begin with \$, that is 0x7E					
VER	Version			Version Information					
Len	the number of bytes after "Len"			Checksums are not counted					
CMD	Commands			Indicate the specific operations, such as play / pause, etc.					
Feedback	Command feedback			If need for feedback, 1: feedback, 0: no feedback					
para1	Parameter 1			Query high data byte					
para2	Parameter 2			Query low data byte					
checksum	Checksum			Accumulation and verification [not include start bit \$]					
\$O	End bit			End bit 0xEF					

For example, if we specify play NORFLASH, you need to send: 7E FF 06 09 00 00 04 FF DD EF
Data length is 6, which are 6 bytes [FF 06 09 00 00 04]. Not counting the start, end, and verification.

Serial Control Cmd

CMD	Function Description	Parameters(16 bit)
0x01	Next	
0x02	Previous	
0x03	Specify tracking(NUM)	0-2999
0x04	Increase volume	
0x05	Decrease volume	
0x06	Specify volume	0-30
0x07	Specify EQ(0/1/2/3/4/5)	Normal/Pop/Rock/Jazz/Classic/Base
0x08	Specify playback mode (0/1/2/3)	Repeat/folder repeat/single repeat/ random
0x09	Specify playback source(0/1/2/3/4)	U/TF/AUX/SLEEP/FLASH
0x0A	Enter into standby – low power loss	
0x0B	Normal working	
0x0C	Reset module	
0x0D	Playback	
0x0E	Pause	
0x0F	Specify folder to playback	1~10(need to set by user)
0x10	Volume adjust set	{DH= 1:Open volume adjust } {DL: set volume gain 0~31}
0x11	Repeat play	{1:start repeat play} {0:stop play}

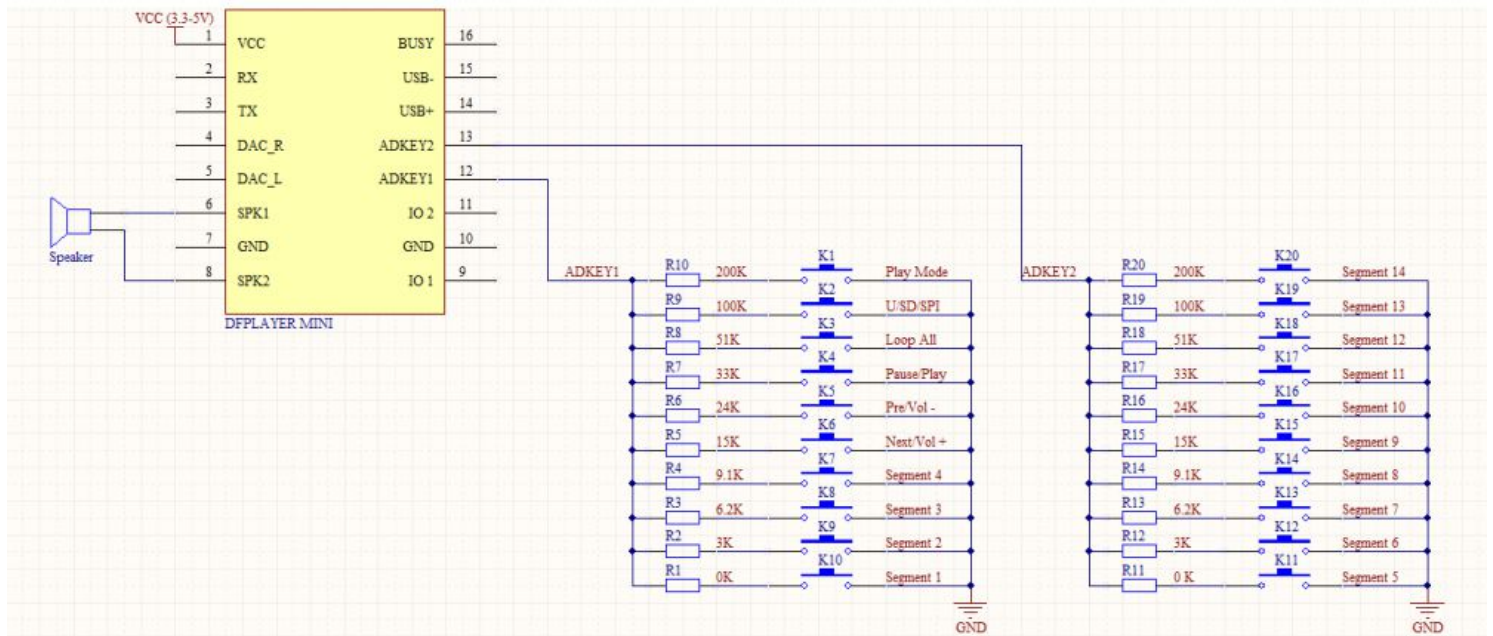
Serial Query Cmd

Commands	Function Description	Parameters(16 bit)
0x3C	STAY	
0x3D	STAY	
0x3E	STAY	
0x3F	Send initialization parameters	0 - 0x0F(each bit represent one device of the low-four bits)
0x40	Returns an error, request retransmission	
0x41	Reply	
0x42	Query the current status	
0x43	Query the current volume	
0x44	Query the current EQ	
0x45	Query the current playback mode	
0x46	Query the current software version	
0x47	Query the total number of TF card files	
0x48	Query the total number of U-disk files	
0x49	Query the total number of flash files	
0x4A	Keep on	
0x4B	Queries the current track of TF card	
0x4C	Queries the current track of U-Disk	
0x4D	Queries the current track of Flash	

2) AD KEY Mode

We use the AD module keys, instead of the traditional method of matrix keyboard connection, it is to take advantage of increasingly powerful MCU AD functionality, Our module default configuration 2 AD port, 20 key resistance distribution.

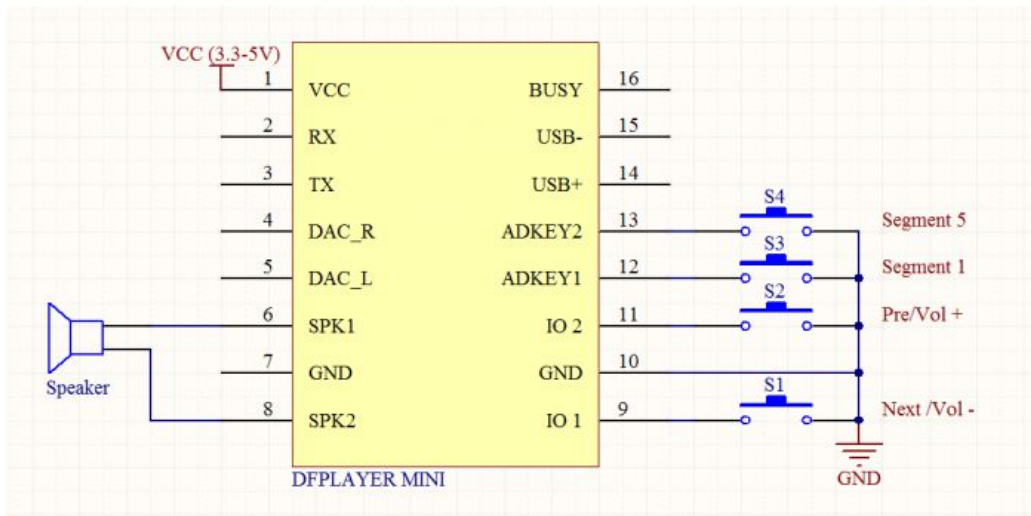
- Refer diagram



3) I/O Mode

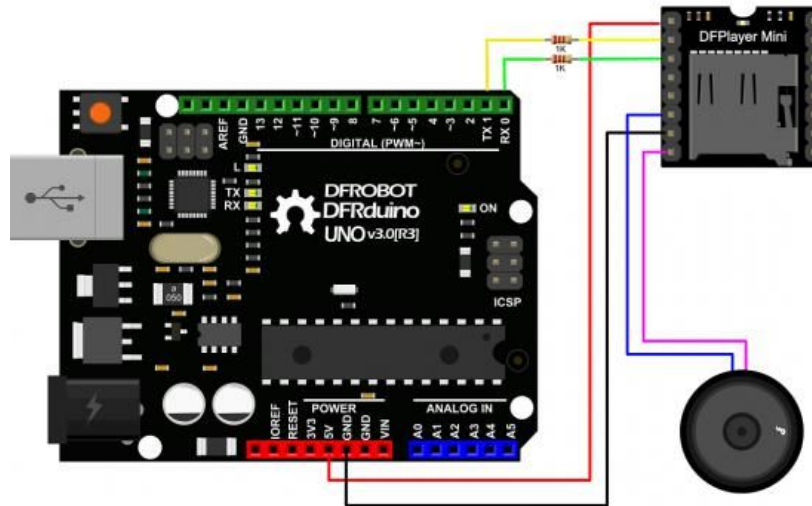
Here comes the most simple way to use this module.

- Refer diagram



note: short time press means pre/next,and long time press means vol- ,vol +

Connection Diagram



Note: For simple use ,the upper diagram is ready,But if you find the noise is quite loud, then you could attach an 1K resistor to the TX pin.

Sample Code

We've created an Arduino library for DFPlayer Mini to simplify the method for you to make it work.

Connect the hardware as the picture above shown and play with the sample code.

Please download [DFPlayer library V2.0 \(https://github.com/DFRobot/DFPlayer-Mini-mp3/archive/master.zip\)](https://github.com/DFRobot/DFPlayer-Mini-mp3/archive/master.zip).

?


```
2 * Copyright:  DFRobot
3 * name:      DFPlayer_Mini_Mp3  sample  code
4 * Author:   lisper <lisper.li@dfrobot.com>
5 *Date:     2014-05-30
6 * Description:sample code for DFPlayer Mini, this code is test on Uno
7 *          note: mp3 file must put into mp3 folder in your tf card
(4) */
9
```

http://www.dfrobot.com/wiki/index.php/DFPlayer_Mini_SKU:DFR0299

```
10
11#include <SoftwareSerial.h>
12#include <DFPlayer_Mini_Mp3.h>
13
14//
15voidsetup () {
16    Serial.begin (9600);
17    mp3_set_serial (Serial);//set Serial for DFPlayer-mini mp3 module
18    mp3_set_volume (15);
19};
20
21
22//
23voidloop () {
24    mp3_play (1);
25    delay (6000);
26    mp3_next ();
27    delay (6000);
28    mp3_prev ();
29    delay (6000);
30    mp3_play (4);
31    delay (6000);
32};
33
34/*
35 mp3_play (); //start play
36 mp3_play (5); //play "mp3/0005.mp3"
37 mp3_next (); //play next
38 mp3_prev (); //play previous
39 mp3_set_volume (uint16_t volume);//0~30
40 mp3_set_EQ (); //0~5
41 mp3_pause ();
42 mp3_stop ();
43 void mp3_get_state (); //send get state command
44 void mp3_get_volume ();
45 void mp3_get_u_sum ();
46 void mp3_get_tf_sum ();
47 void mp3_get_flash_sum ();
48 void mp3_get_tf_current ();
49 void mp3_get_u_current ();
50 void mp3_get_flash_current ();
51 void mp3_single_loop (boolean state); //set single loop
52 void mp3_DAC (boolean state);
53 void mp3_random_play ();
54 */
```

Retrieved from "http://www.dfrobot.com/wiki/index.php?title=DFPlayer_Mini_SKU:DFR0299&oldid=27096"

Categories: [Product Manual](#) | [DFR Series](#) | [Module](#) | [DFRobot](#)

■ This page was last modified on 12 August 2014, at 10:43.

This page has been accessed 2,322 times.

http://www.dfrobot.com/wiki/index.php/DFPlayer_Mini_SKU:DFR0299

LAMPIRAN 5
Data Sheet Sensor E18-D80nk

Proximity Sensor/Switch

E18-D80NK



Description

A very cheap, easy to assemble, easy to use Infrared sensor with a long detection distance and has less interference by visible light. The implementations of modulated IR signal immune the sensor to the interferences caused by the normal light of a light bulb or the sun light. This sensor has a screwdriver adjustment to set the appropriate detected distance to make it useful in many applications, and then gives a digital output when it senses something within that

range. This sensor does not measure a distance value. It can be used for collision avoidance robot and machine automation. The sensor provides a non-contact detection.

Specifications

- 1 Input voltage : +5V DC
- 1 Current consumption : > 25mA (min) ~ 100mA (max)
- 1 Dimension : 1.7cm (diameter) x 4.5cm (length)
- 1 Cable length : 45cm
- 1 Detection of objects : transparent or opaque
- 1 Diffuse reflective type
- 1 Sensing range : 3cm to 80cm (depends on obstacle surface)
- 1 NPN output (normally high)
- 1 Environment temperature : -25 °C ~ 55 °C

Pin Configuration

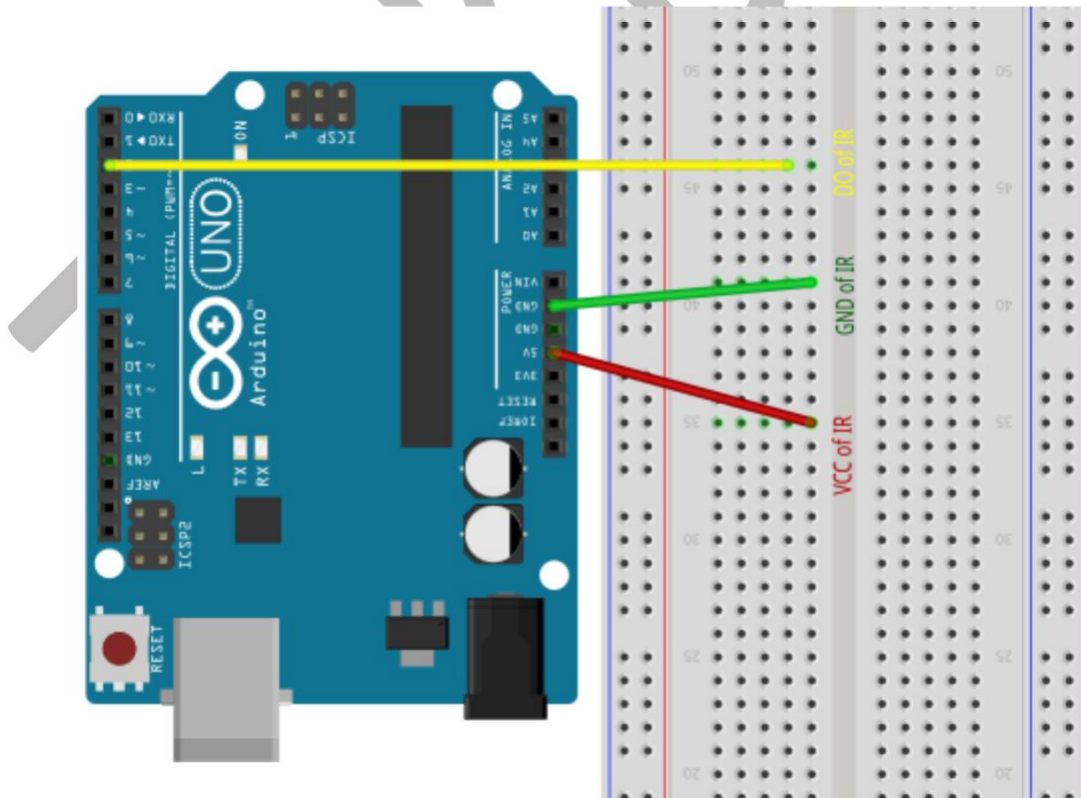
- (5) Red wire : +5V
- (6) Green wire : GND
- (7) Yellow wire : DIGITAL OUTPUT

Sample Program

```
void setup()
{
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  // make the pin 2 as an input:
  pinMode(2,INPUT);
}

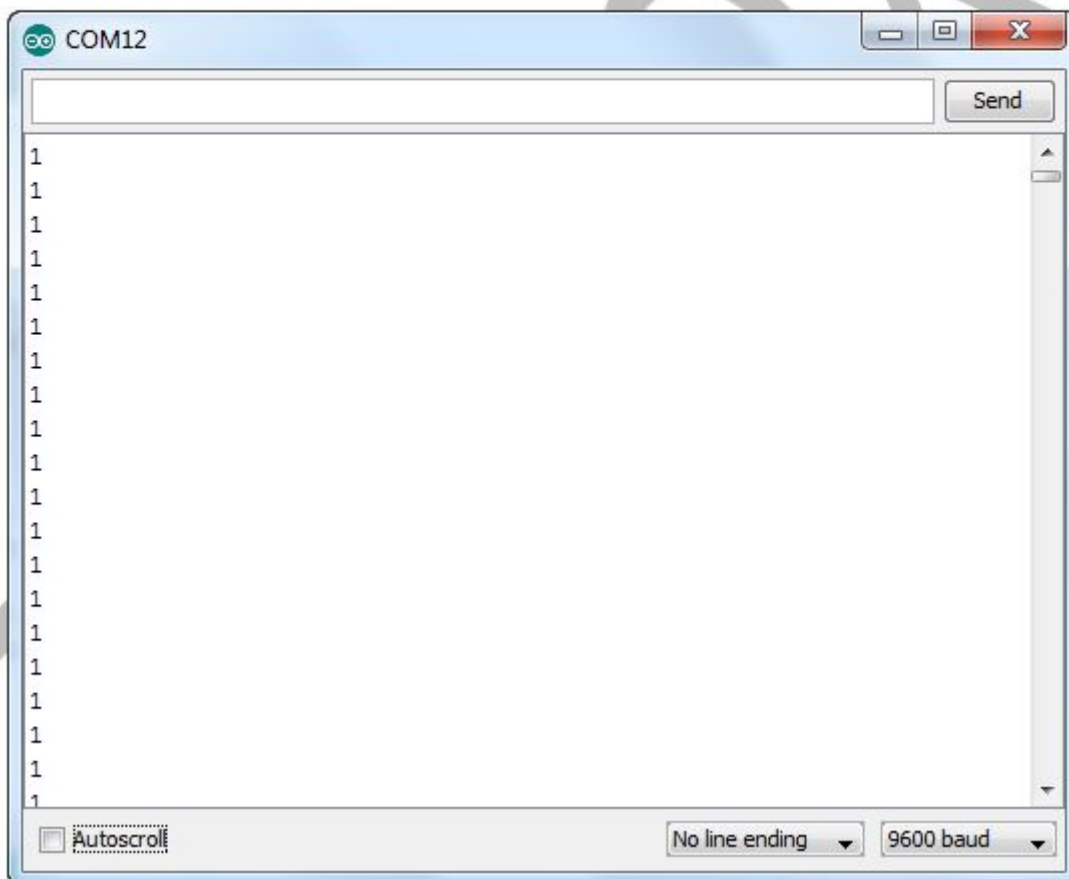
void loop()
{
  // read the input pin:
  int irread = digitalRead(2);
  // print out the reading of IR:
  Serial.println(irread);
  delay(1); // delay in between reads for stability
}
```

Wiring Diagram



Testing

- 1 Please check all the connections from the given wiring diagram.
- 2 Type the sample program in your Arduino sketch then upload.
- 3 Open the serial monitor in the upper right corner.
- 4 When there is no object in front of the proximity sensor, the output must be 1.

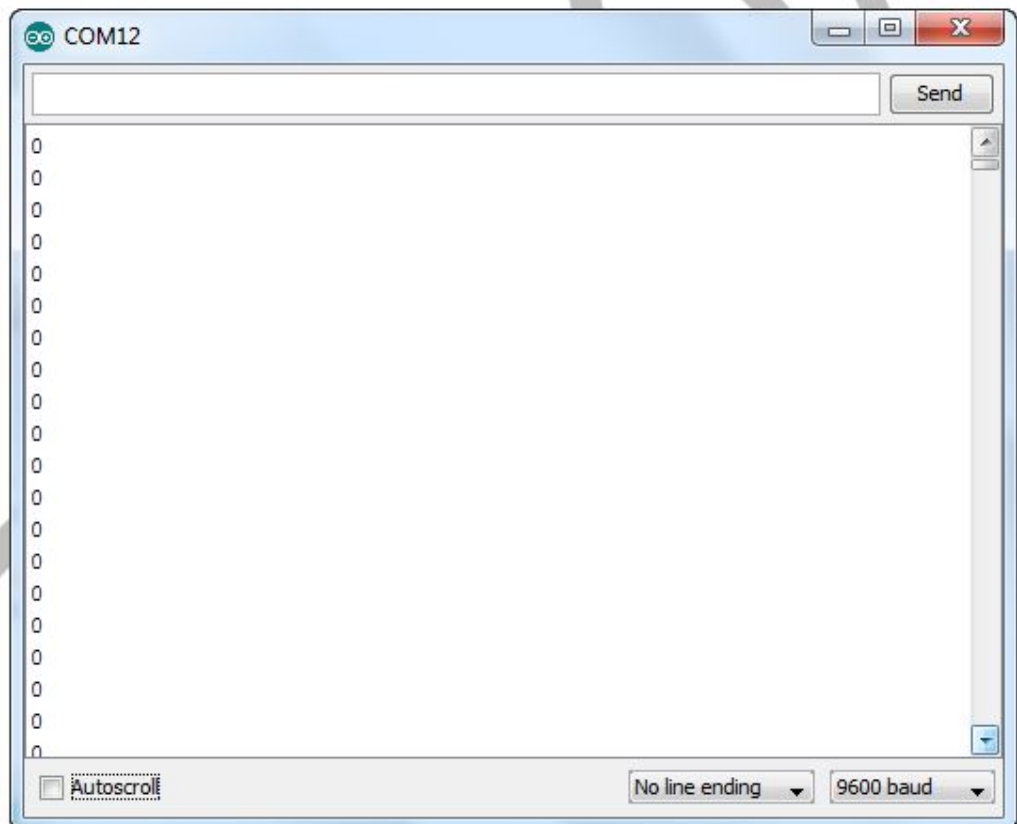


5 Try to put an object in front of the proximity sensor, the proximity sensor will light and the output will become 0.

Note:

55 If there is no change in the output of proximity sensor, try to move the object closer.

56 Adjust the screw in the proximity sensor to change the range of detection.



LAMPIRAN 6
Program Keseluruhan

```

#include <SoftwareSerial.h> //memanggil library SoftwareSerial
#include <DFPlayer_Mini_Mp3.h> //memanggil library DFPlayer mini
int trig1= 5;          // membuat variabel trig yang di set ke-pin 5
int echo1= 4;         // membuat variabel echo yang di set ke-pin 4
long durasi1, jarakka; // membuat variabel durasi dan jarak
int trig2= 9;         // membuat variabel trig yang di set ke-pin 9
int echo2= 8;         // membuat variabel echo yang di set ke-pin 8
long durasi2, jarakde; // membuat variabel durasi dan jarak
int trig3= 12;        // membuat variabel trig yang di set ke-pin 12
int echo3= 11;        // membuat variabel echo yang di set ke-pin 11
long durasi3, jarakki; // membuat variabel durasi dan jarak
SoftwareSerial mySerial(2, 3); // Declare pin RX & TX
const int switchPin = 7;
int value = 0;

void setup() {
pinMode(trig1, OUTPUT); // set pin trig menjadi OUTPUT
pinMode(echo1, INPUT);  // set pin echo menjadi INPUT
pinMode(trig2, OUTPUT); // set pin trig menjadi OUTPUT
pinMode(echo2, INPUT);  // set pin echo menjadi INPUT
pinMode(trig3, OUTPUT); // set pin trig menjadi OUTPUT
pinMode(echo3, INPUT);  // set pin echo menjadi INPUT
pinMode(switchPin, INPUT);
Serial.begin(9600);
  mySerial.begin (9600);
  mp3_set_serial (mySerial); //set softwareSerial for DFPlayer
  delay(10);
  mp3_reset(); //soft-Reset module DFPlayer
  delay(10); //wait 1 ms for respon command
  mp3_set_volume (30); //set Volume module DFPlayer

```

```

Serial.begin(9600);
delay(1000);
}

void ultrasonik_ka() {
digitalWrite(trig1, LOW);
delayMicroseconds(100);
digitalWrite(trig1, HIGH);
delayMicroseconds(100);
digitalWrite(trig1, LOW);
delayMicroseconds(100);

durasi1= pulseIn(echo1, HIGH); // menerima suara ultrasonic
jarakka= (durasi1/2) / 29.1; // mengubah durasi menjadi jarak (cm)
//Serial.println(jarakka); // menampilkan jarak pada Serial Monitor
delay(200);
}

void ultrasonik_de(){
digitalWrite(trig2, LOW);
delayMicroseconds(100);
digitalWrite(trig2, HIGH);
delayMicroseconds(100);
digitalWrite(trig2, LOW);
delayMicroseconds(100);

durasi2= pulseIn(echo2, HIGH); // menerima suara ultrasonic
jarakde= (durasi2/2) / 29.1; // mengubah durasi menjadi jarak (cm)
//Serial.println(jarakde); // menampilkan jarak pada Serial Monitor
delay(200);
}

```

```

void ultrasonik_ki(){
    digitalWrite(trig3, LOW);
    delayMicroseconds(100);
    digitalWrite(trig3, HIGH);
    delayMicroseconds(100);
    digitalWrite(trig3, LOW);
    delayMicroseconds(100);

    durasi3= pulseIn(echo3, HIGH); // menerima suara ultrasonic
    jarakki= (durasi3/2) / 29.1; // mengubah durasi menjadi jarak (cm)
    //Serial.println(jarakki); // menampilkan jarak pada Serial Monitor
    delay(200);
}
void inf(){
    value = digitalRead(switchPin);
}
void loop() {
    ultrasonik_de();
    ultrasonik_ka();
    ultrasonik_ki();
    inf();
    Serial.print(jarakde);
    Serial.print("-");
    Serial.println(jarakki);
    if (jarakde <= 115 && jarakde >=105 ){
    mp3_play(1);
    delay(1500);
    mp3_play(2);
    delay(1500);
    mp3_play(3);
    delay(1500);
}
}

```

```
}  
if (jarakki <= 50){  
  mp3_play(6);  
  delay(1300);  
  mp3_play(1);  
  delay(1500);  
  mp3_play(2);  
  delay(1500);  
  mp3_play(7);  
  delay(1200);  
  mp3_play(8);  
  delay(1200);  
  mp3_play(11);  
  delay(1500);  
  mp3_play(5);  
  delay(1500);  
}  
if (jarakka <= 50){  
  mp3_play(6);  
  delay(2000);  
  mp3_play(1);  
  delay(1500);  
  mp3_play(2);  
  delay(1500);  
  mp3_play(7);  
  delay(1500);  
  mp3_play(8);  
  delay(1200);  
  mp3_play(11);  
  delay(1500);  
  mp3_play(4);
```

```
delay(1000);
}
if (jarakde <= 50){
mp3_play(6);
delay(2000);
mp3_play(1);
delay(1500);
mp3_play(2);
delay(2000);
mp3_play(7);
delay(1500);
mp3_play(8);
delay(1500);
mp3_play(3);
delay(1500);
}

if( value == HIGH )
{
mp3_play(6);
delay(1500);
mp3_play(1);
delay(1500);
mp3_play(9);
delay(1500);
mp3_play(3);
delay(1500);
}
else
{
Serial.println("2");
```