



**PERANCANGAN DAN IMPLEMENTASI ALAT PENDETEKSI DENYUT
NADI BERDASARKAN USIA MENGGUNAKAN *PULSE SENSOR*
BERBASIS ARDUINO UNO**

PROYEK AKHIR

Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta untuk
Memenuhi Sebagian Persyaratan Guna Memperoleh Gelar Ahli Madya Teknik



OLEH:
USWATUN KHASANAH
NIM. 13507134012

PROGRAM STUDI TEKNIK ELEKTRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2016

LEMBAR PERSETUJUAN

PROYEK AKHIR

**PERANCANGAN DAN IMPLEMENTASI ALAT PENDETEKSI DENYUT
NADI BERDASARKAN USIA MENGGUNAKAN *PULSE SENSOR*
BERBASIS ARDUINO UNO**

Oleh

USWATUN KHASANAH

13507134012

Telah diperiksa dan disetujui oleh pembimbing:

Untuk diuji



Yogyakarta, 30 September 2016

Mengetahui
Kaprodi Teknik Elketronika

Dr. Sri Waluyanti, M.Pd.
NIP. 19581218 198603 2 001

Menyetujui
Pembimbing

Dessy Irmawati, M.T.
NIP. 19791214 201012 2 002

LEMBAR PENGESAHAN

PROYEK AKHIR

**PERANCANGAN DAN IMPLEMENTASI ALAT PENDETEKSI DENYUT
NADI BERDASARKAN USIA MENGGUNAKAN *PULSE SENSOR*
BERBASIS ARDUINO UNO**


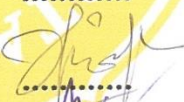

Dipersiapkan dan Disusun Oleh:

Uswatun Khasanah

13507134012

Telah Dipertahankan di depan Dewan Penguji Proyek Akhir
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
Pada tanggal 13 Oktober 2016
Dan Dinyatakan Telah Memenuhi Syarat Guna Memperoleh Gelar
Ahli Madya Teknik

SUSUNAN DEWAN PENGUJI

Jabatan	Nama Lengkap	Tandatangan	Tanggal
1. Ketua Penguji	Dessy Irmawati, M.T.		24/10-16
2. Sekretaris	Nuryake Fajaryati, M.Pd.		24/10-16
3. Penguji Utama	Muhammad Munir, M.Pd.		24/10-16

Yogyakarta, 26 Oktober 2016

Mengetahui,

Dekan Fakultas Teknik UNY



Dr. Widarto, M.Pd

NIP. 19631230 198812 1 1001 4.

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : Uswatun Khasanah
NIM : 13507134012
Program Studi : Teknik Elektronika DIII
Judul Proyek Akhir : Perancangan dan Implementasi Alat Pendeteksi
Denyut Nadi Berdasarkan Usia Menggunakan *Pulse*
Sensor Berbasis Arduino Uno

Menyatakan bahwa Proyek Akhir ini adalah hasil pekerjaan saya sendiri, dan sepanjang pengetahuan saya, tidak berisi materi yang ditulis oleh orang lain sebagai persyaratan penyelesaian studi di Universitas Negeri Yogyakarta atau perguruan tinggi lain, kecuali bagian-bagian tertentu saya ambil sebagai acuan dengan mengikuti kaidah penulisan karya ilmiah yang benar. Jika ternyata terbukti pernyataan ini tidak benar, sepenuhnya menjadi tanggung jawab saya.

Yogyakarta, 30 september 2016

Yang menyatakan



Uswatun Khasanah

NIM. 13507134012

MOTTO

SEMUA KEMBALI KEPADA KATA SYUKUR (*GRATITUDE*), SEBAB SYUKUR TIDAK SEKEDAR MENGUCAPKAN TERIMA KASIH TERHADAP SEBUAH PEMBERIAN, TETAPI CARA KITA BERTANGGUNG JAWAB TERHADAP PEMBERIAN TERSEBUT
(DR. ANDHYKA P.SETIAWAN).

"APABILA DI DALAM DIRI SESEORANG MASIH ADA RASA MALU DAN TAKUT UNTUK BERBUAT SUATU KEBAIKAN, MAKA JAMINAN BAGI ORANG TERSEBUT ADALAH TIDAK AKAN BERTEMUNYA IA DENGAN KEMAJUAN "
(Bung Karno)

"ORANG TIDAK BAIK YANG HIDUP DALAM SISTEM YANG BAIK AKAN BERUBAH MENJADI ORANG BAIK, SEBALIKNYA ORANG BAIK YANG HIDUP DALAM SISTEM YANG BURUK AKAN BERUBAH MENJADI ORANG YANG TIDAK BAIK"
(Prof.Dr.Yusril Ihza Mahendra)

LEMBAR PERSEMBAHAN

Dengan segala kerendahan hati, ingin kupersembahkan sebuah karya sederhana yang telah berhasil kuselasaikan ini kepada;

Rasa syukur kepada ALLAH SWT atas segala rahmat dan hidayah-Nya. Taburan cinta dan kasih sayang-Mu telah memberikanku kekuatan, membekaliku dengan ilmu serta memperkenalkanku dengan cinta. Atas karunia serta kemudahan yang Engkau berikan akhirnya Laporan Proyek Akhir yang sederhana ini dapat terselesaikan. Sholawat dan salam selalu terlimpahkan kepada Rasulullah Muhammad SAW.

Kupersembahkan karya sederhana ini kepada orang yang sangat kukasihi dan kusayangi. Ibu, Bapak. Sebagai tanda bakti, hormat, dan rasa terima kasih yang tiada terhingga yang telah memberikan kasih sayang, segala dukungan, dan cinta kasih yang tiada terhingga yang tiada mungkin dapat kubalas hanya dengan selembar kertas yang bertuliskan kata cinta dan persembahan. Semoga ini menjadi langkah awal untuk membuat Ibu dan Bapak bahagia karna kusadar, selama ini belum bisa berbuat yang lebih. Untuk Ibu dan Bapak yang selalu membuatku termotivasi dan selalu memberikan kasih sayang, selalu mendoakanku, selalu menasehatiku menjadi lebih baik.

Seluruh Dosen Pengajar Teknik Elektronika, terima kasih banyak untuk semua ilmu, didikan dan pengalaman yg sangat berarti yang telah kalian berikan. Untuk Dosen Pembimbing saya Dessy Irmawati, M.T. terima kasih bimbingan dan

bantuannya selama ini, atas nasihat dan pelajaran yang saya dapatkan, saya tidak akan lupa atas bantuan dan kesabaran dari ibu.

Teman-temanku di kelas B Teknik Elektronika 2013, sungguh kebersamaan yang kita bangun selama ini telah banyak merubah kehidupanku. Kemarahanmu telah menuntunku menuju kedewasaan, senyummu telah membuka cakrawala dunia dan melepaskan belenggu-belenggu ketakutanku, tetes air mata yang mengalir di pipimu telah mengajarku arti kepedulian yang sebenarnya, dan gelak tawamu telah membuatku bahagia. Sungguh aku bahagia bersamamu, bahagia memiliki kenangan indah dalam setiap bait pada paragraf kisah persahabatan kita. Bila Tuhan memberikanku umur panjang, akan aku bagi harta yang tak ternilai ini (persahabatan) dengan anak dan cucuku kelak.

*“Ya Allah, jadikanlah Iman, Ilmu dan Amal ku sebagai lentera
jalan hidupku keluarga dan saudara seimanku”*

PROYEK AKHIR

PERANCANGAN DAN IMPLEMENTASI ALAT PENDETEKSI DENYUT NADI BERDASARKAN USIA MENGGUNAKAN *PULSE SENSOR* BERBASIS ARDUINO UNO

Oleh : Uswatun Khasanah

NIM : 13507134012

ABSTRAK

Tujuan proyek akhir ini adalah sebagai alat pendeteksi denyut nadi menggunakan *pulse sensor* berdasarkan usia yang dapat memberikan informasi kondisi kesehatan seseorang. Sensor ini bekerja berdasarkan intensitas cahaya sesuai volume darah yang mengalir pada pembuluh darah kapiler. Photodiode akan mendeteksi intensitas cahaya yang dipancarkan oleh LED inframerah dan kemudian data tegangan akan dikirim ke mikrokontroler Atmega328p. Data kemudian diproses sehingga memperoleh hasil berupa jumlah denyut nadi dan kondisi normal atau abnormal (*takikardia* dan *bradikardia*) yang akan ditampilkan pada LCD dalam satuan beat per menit (BPM).

Metode yang digunakan dalam membangun alat ini terdiri atas beberapa tahap secara berurutan, yaitu analisis kebutuhan yang diperlukan untuk membuat alat, identifikasi kebutuhan, perancangan perangkat keras dan perangkat lunak, pembuatan alat, kemudian pengujian alat. Perangkat ini tersusun dari 3 bagian utama: (1). Bagian *input*, dengan komponen *pulse sensor*, dan *push button*. (2). Bagian proses, dengan komponen utama Arduino uno. (3). Bagian *output*, keluaran dari *pulse sensor* berupa sinyal analog kemudian di olah oleh mikrokontroler arduino uno sehingga dapat menampilkan sinyal BPM, data pembacaan sensor akan ditampilkan di LCD, dan LED sebagai indikator ada tidaknya denyut nadi.

Hasil pengujian dapat disimpulkan bahwa alat pendeteksi denyut nadi dapat bekerja sesuai dengan prinsip yang telah dirancang. Perangkat lunak yang ada di mikrokontroler juga dapat berjalan sesuai sintaksinya. Selain itu, sistem kerja alat yang menampilkan data denyut nadi berdasarkan usia dapat bekerja dengan baik. *Pulse sensor* memiliki tingkat pengukuran yang presisi dengan rata-rata kesalahan sebesar 1,87 %.

Kata kunci: *Photodiode*, Atmega 328p, *Pulse sensor*, *Push button*, LCD

KATA PENGANTAR

Assamu'alaikum wr. wb.

Puji syukur kehadiran ALLAH SWT atas segala rahmat dan hidayah-Nya, penulis dapat melaksanakan Laporan Proyek Akhir ini dapat terselesaikan tanpa halangan yang berarti. Sholawat dan salam semoga tercurah pada Qudwah kita Rasulullah SAW keluarga, sahabat dan orang-orang yang istiqomah di jalan-Nya.

Dalam menyusun Laporan Proyek Akhir ini penulis merasa banyak kekurangan karena terbatasnya kemampuan dan pengetahuan penulis. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih dan penghargaan yang sebesar-besarnya kepada:

1. Bapak, Ibu, Adik dan seluruh keluarga atas dukungan yang telah diberikan.
2. Ibu Dessy Irmawati, M.T. selaku Dosen Pembimbing Penyusun Laporan Proyek akhir.
3. Ibu Dr. Sri Waluyanti, M.Pd. selaku Ketua Program Studi Diploma III dan Koordinator Proyek Akhir Teknik Elektronika Universitas Negeri Yogyakarta.
4. Bapak Dr. Fatchul Arifin, M.T. selaku Ketua Jurusan Pendidikan Teknik Elektronika Universitas Negeri Yogyakarta.
5. Bapak Dr. Widarto M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
6. Seluruh Dosen Pengajar Teknik Elektronika Universitas Negeri Yogyakarta atas bekal ilmu yang diberikan kepada penulis.
7. Teman-teman seperjuangan angkatan 2013 khususnya kelas B 2013 yang telah memberikan bantuan sehingga pembuatan proyek akhir ini dapat terselesaikan.

8. Syahrul Awalludin Sidiq, Mar'atus Arifiah, Ari Widiyatmoko, Ahmad Haris ,dan Dwi Ratna Sari yang telah membantu menemani dan memberikan dukungan dalam pembuatan proyek akhir.
9. Semua pihak yang tidak dapat kami sebutkan satu persatu yang telah membantu hingga terselesainya laporan ini.

Penulis menyadari bahwa laporan ini masih banyak kekurangan dan jauh dari sempurna walaupun penulis telah berusaha untuk mendekati kesempurnaan, maka penulis berharap para pembaca memberikan saran dan kritik yang membangun.

Akhir kata penulis mohon maaf yang sebesar-besarnya apabila ada kekeliruan di dalam penulisan laporan ini.

Wassalamu'alaikum wr. Wb

Yogyakarta, 30 September 2016

Penulis



Uswatun Khasanah

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN KEASLIAN	iv
M O T T O	v
LEMBAR PERSEMBAHAN	vi
ABSTRAK	viii
KATA PENGANTAR	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
DAFTAR LAMPIRAN	xvi
BAB I PENDAHULUAN	1
A. Latar Belakang Masalah.....	1
B. Identifikasi Masalah	4
C. Batasan Masalah.....	4
D. Rumusan Masalah	5
E. Tujuan Proyek Akhir.....	5
F. Manfaat Proyek Akhir.....	6
G. Keaslian Gagasan	7
BAB II PENDEKATAN PEMECAHAN MASALAH	10
A. Sirkulasi Darah dalam Tubuh.....	10
B. Nadi.....	12
C. Pulse Sensor	17
D. Pengertian LCD (Liquid Cristal Display)	18
E. Catu Daya.....	23
F. Light Emitting Dioda (LED).....	25
G. Saklar Push On.....	26
H. Arduino Uno	27

I. Perangkat Lunak (<i>Software</i>).....	33
BAB III KONSEP PERANCANGAN ALAT	41
A. Analisa.....	41
B. Perancangan Alat	48
C. Proses Pembuatan	54
D. Implementasi	55
E. Pengujian Alat.....	56
F. Spesifikasi Alat	58
G. Perincian Alat, Bahan dan Rincian Biaya	59
BAB IV PROSES, HASIL, DAN PEMBAHASAN.....	62
A. Perancangan	62
B. Pengembangan	76
C. Implementasi Dan Pengujian	77
D. Pembahasan.....	88
E. Cara Kerja Alat Pendeteksi Denyut Nadi	91
BAB V KESIMPULAN DAN SARAN.....	93
A. Kesimpulan	93
B. Keterbatasan Alat	94
C. Saran.....	94
DAFTAR PUSTAKA	95
LAMPIRAN.....	96

DAFTAR GAMBAR

	Halaman
Gambar 1. Anatomi Jantung Manusia.....	10
Gambar 2. Komponen Perangkat Pulse Sensor	17
Gambar 3. Modul LCD 16x2	19
Gambar 4. Konfigurasi PIN LCD 16x2	20
Gambar 5 . Konfigurasi Pin pada 7805	23
Gambar 6. Simbol dan Bentuk Fisik LED	26
Gambar 7. Simbol Dan Bentuk Saklar Push On	26
Gambar 8. Arduino Uno.....	27
Gambar 9. Peta Memori Program ATmega 328	31
Gambar 10. Peta Memori Data ATmega 328	32
Gambar 11. Tampilan Jendela Arduino.	34
Gambar 12. Blok Diagram	42
Gambar 13. Blok Diagram Sistem	48
Gambar 14. Rangkaian Pengendali Arduino Uno.....	50
Gambar 15. Rangkain <i>Pulse Sensor</i>	51
Gambar 16 . Rangkaian LCD.....	53
Gambar 17. Pembuatan <i>Design</i> PCB dengan <i>Software Proteus</i>	62
Gambar 18. Desain PCB	63
Gambar 19. Perakitan Komponen di PCB	64
Gambar 20. <i>Design box</i>	64
Gambar 21. <i>Design Box</i> Adaptor	65
Gambar 22. <i>Flowchart</i> Sistem Keseluruhan	67
Gambar 23. Sub Rutin Kondisi Dewasa	70
Gambar 24. Sub Rutin Kondisi Anak	72
Gambar 25. Sub Rutin Kondisi Bayi.....	74
Gambar 26. Penempatan Komponen (depan)	76
Gambar 27. Penempatan Komponen (atas).....	76
Gambar 28. Penempatan Komponen (belakang)	77

Gambar 29. Penempatan Komponen Adaptor	77
Gambar 30. Pengujian Tegangan Menggunakan Multimeter	78
Gambar 31. Tampilan <i>Visualisasi Processing</i>	79
Gambar 32. Diagram Blok Cara Kerja Alat	91

DAFTAR TABEL

	Halaman
Tabel 1. Perbedaan Denyut Nadi Manusia.....	12
Tabel 2. Keterangan Pin-Pin LCD 16x2	21
Tabel 3. Karakteristik Regulator Tegangan Seri 78xx.....	24
Tabel 4. Spesifikasi Arduino UNO	28
Tabel 5. Identifikasi Kebutuhan Alat	47
Tabel 6. Pin Pada Mikrokontroler Atmega 328p	50
Tabel 7 . Rincian komponen yang digunakan pada rangkaian <i>Pulse sensor</i>	52
Tabel 8 . Alat dan Fungsinya	54
Tabel 9. Uji Fungsional.....	56
Tabel 10. Unjuk Kerja.....	57
Tabel 11. Daftar Alat yang Dibutuhkan	59
Tabel 12. Daftar Bahan yang Dibutuhkan	60
Tabel 13. Rincian Biaya.....	61
Tabel 14. Listing Program Utama.....	69
Tabel 15. Pengujian Tegangan Baterai 9 Volt	78
Tabel 16. Pengujian <i>Pulse Sensor</i>	80
Tabel 17. Hasil Pengujian Fungsi LCD dan <i>Push Button</i>	82
Tabel 18. Pengujian Alat Usia Dewasa.....	84
Tabel 19. Pengujian Alat Usia Anak-Anak.....	85
Tabel 20. Pengujian Alat Usia Bayi.....	86
Tabel 21. Pengujian Alat terhadap orang sakit dan setelah berolahraga.....	87
Tabel 22. Standar Penulisan Program	89

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Desain Rangkaian Alat	97
Lampiran 2. Desain Alat	98
Lampiran 4. Desain PCB.....	99
Lampiran 4. Schematic Arduino Uno R3.....	100
Lampiran 5. Datasheet LCD 16x2	101
Lampiran 6. Datasheet Pulse Sensor.....	109
Lampiran 7. Source Code.....	119

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Kesehatan merupakan bagian yang penting bagi manusia karena dengan sehat kita dapat melakukan berbagai kegiatan dan berpikir dengan baik. Jantung merupakan organ vital yang dimiliki manusia, akan tetapi banyak masyarakat yang kurang menyadari akan pentingnya organ jantung sehingga banyak muncul penderita penyakit jantung. Berdasarkan data Riset Kesehatan Dasar (Riskesdas) 2013, prevalensi penderita penyakit kardiovaskular cenderung meningkat. Data dari Badan Penelitian dan Pengembangan Kesehatan (Balitbangkes) Kementerian Kesehatan Indonesia menyebutkan, dari total angka kematian yang ada (41.590) di Indonesia, 21,1 persen di antaranya disebabkan oleh stroke dan 19,1 persen dikarenakan penyakit jantung koroner. Dari data jelas tergambar, kedua penyakit kardiovaskular masih menduduki peringkat teratas penyebab kematian terbanyak di Indonesia.

Berdasarkan wawancara yang dilakukan penulis terhadap tenaga medis Dr. Theodorus yang bekerja di apotek kimia farma UNY, untuk tahap awal pemeriksaan medis, biasanya dilakukan *medical check up* sebelum penyakit seseorang didiagnosa. Hasil *medical check up* akan diketahui apakah seseorang dalam kondisi sehat atau tidak. Pada umumnya *medical check up* yang dilakukan di rumah sakit pertama kali adalah detak jantung. Hal itu dilakukan karena jantung pada tubuh manusia merupakan organ utama, dimana fungsi kerja

jantung mempengaruhi organ-organ penting manusia lainnya. Cara sederhana untuk menghitung detak jantung adalah dengan mengukur melalui denyut nadi. Denyut nadi dapat diukur di belakang lutut, kunci paha, leher, sisi atas atau bagian dalam kaki, pelipis, pergelangan tangan, atau di bagian arteri dekat kulit. Akan tetapi jika pengukuran nadi dilakukan oleh orang awam sangat memungkinkan terjadi kesalahan penghitungan detak jantung, karena proses penghitungan dilakukan secara manual dan waktu perhitungan yang tidak tepat berbeda dengan ahli medis yang sudah berpengalaman.

Suara jantung pada umumnya dideteksi paramedis menggunakan stetoskop akustik. Secara umum, teknik mendengarkan suara tubuh untuk mendeteksi kondisi kesehatan pasien disebut *auskultasi*. *Auskultasi* dengan stetoskop akustik tidak mudah dilakukan karena bisa terganggu oleh adanya bunyi sekitar, sedangkan intensitas bunyi jantung sendiri relatif rendah. Selain itu telinga pengguna harus peka agar hasil deteksi akurat.

Pada individu yang sehat, nadi menggambarkan denyut jantung artinya, frekuensi nadi sama dengan frekuensi kontraksi ventrikel jantung. Akan tetapi pada beberapa jenis penyakit kardiovaskular, denyut jantung dan nadi dapat berbeda (Barbara Kozier, 2010 : 672)

Elektrokardiogram (EKG) digunakan untuk mendeteksi kelainan jantung dengan mengukur aktivitas listrik yang dihasilkan oleh jantung, sebagaimana jantung berkontraksi. Hasil yang diberikan oleh EKG memiliki keakuratan yang lebih baik karena menggunakan elektroda yang diletakkan pada bagian tertentu

tubuh untuk mendiagnosis adanya kelainan jantung seperti aritma jantung, pembesaran jantung, peradangan jantung, dan penyakit jantung koroner. Permasalahan dalam pemeriksaan menggunakan EKG adalah minimnya peralatan elektrokardiografi yang hanya terdapat di beberapa rumah sakit besar dengan jumlah yang terbatas.

Informasi mengenai detak jantung menjadi sangat penting dalam pemantauan kesehatan dalam beraktivitas terutama saat berolahraga. Saat berolahraga sangat penting untuk menghitung denyut nadi sebelum dan sesudah berolahraga. Hal ini bertujuan untuk memantau kesehatan kardiovaskular. Olahraga atau aktivitas fisik dapat meningkatkan jumlah denyut jantung, namun jika jumlahnya terlalu berlebihan atau diluar batas sehat dapat menimbulkan bahaya.

Berdasarkan dari permasalahan yang ada, penulis bermaksud mengembangkan teknologi kesehatan pengukur detak jantung secara otomatis. Tugas akhir ini berfokus pada pembuatan alat yang mampu mengukur denyut nadi berdasarkan usia. Tujuan pengembangan alat ini, agar hasil pengukurannya dapat dilihat secara *real time*. Oleh karena itu, penulis bermaksud merancang dan membangun proyek akhir dengan judul “ Perancangan dan Implementasi Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno ”.

B. Identifikasi Masalah

Berdasarkan uraian latar belakang masalah di atas, dapat dibuat identifikasi masalah sebagai berikut :

1. Kurangnya kesadaran masyarakat akan pentingnya kesehatan organ jantung.
2. Pengukuran denyut nadi secara manual yang dilakukan orang awam memungkinkan terjadinya kesalahan perhitungan.
3. Stetoskop akustik tidak mudah dilakukan karena bisa terganggu oleh adanya bunyi sekitar, sedangkan intensitas bunyi jantung sendiri relatif rendah. Selain itu telinga pengguna harus peka agar hasil deteksi akurat
4. Pengukuran detak jantung dengan EKG memiliki keakuratan yang lebih baik, namun alat ini sangat terbatas.
5. Belum ada alat pendeteksi denyut nadi yang murah dan mudah digunakan orang awam.

C. Batasan Masalah

Berdasarkan identifikasi masalah di atas, perlu adanya pembatasan masalah sehingga ruang lingkup permasalahan lebih jelas. Pembuatan alat pendeteksi denyut nadi menggunakan chip mikrokontroler Atmega 328p sebagai kontrol utama, *pulse sensor* sebagai sensor untuk mendeteksi denyut nadi. *Push button* digunakan sebagai tombol pemilihan usia. *Output* hasil berupa jumlah denyut nadi manusia dalam satuan BPM (*Beats Per Minute*) serta menampilkan kondisi normal atau abnormal (*bradikardia* dan *takikardia*) yang akan ditampilkan pada

media *interface* menggunakan LCD 16x2. LED *superbright* 3mm digunakan sebagai indikator ada tidaknya denyut nadi.

D. Rumusan Masalah

Berdasarkan latar belakang masalah, identifikasi masalah serta batasan masalah maka rumusan masalah yang akan dibahas dalam proyek akhir ini meliputi:

1. Bagaimana merancang dan membangun *Hardware* Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno ?
2. Bagaimana merancang dan membangun *Software* yang diterapkan pada *Hardware* Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno ?
3. Bagaimana unjuk kerja alat dan implementasi *Hardware* Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno?

E. Tujuan Proyek Akhir

Adapun tujuan dalam pembuatan proyek akhir yang berjudul “Perancangan dan Implementasi Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno”, untuk memenuhi persyaratan Guna Memperoleh Gelar Ahli Madya ini adalah sebagai berikut:

1. Merealisasikan *hardware* Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno.
2. Merealisasikan *software* Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno.
3. Mengetahui unjuk kerja alat dan mengimplementasikan Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno.

F. Manfaat Proyek Akhir

Pembuatan Proyek akhir ini diharapkan dapat bermanfaat bagi mahasiswa, lembaga pendidikan, dan industri. Adapaun manfaat yang diharapkan dari pembuatan proyek akhir ini antara lain:

1. Secara Teoritis
 - a. Bagi mahasiswa
 1. Sebagai sarana implementasi pengetahuan yang didapat dibangku pendidikan.
 2. Dapat digunakan sebagai bahan pembelajaran dan penambah wawasan tentang alat pendeteksi denyut nadi serta sebagai kajian untuk pengembangan selanjutnya.
 3. Sebagai bentuk kontribusi terhadap universitas baik dalam citra maupun daya tawar terhadap masyarakat lain.

4. Memberikan motivasi kepada mahasiswa untuk tetap berkarya dan menjadi salah satu pelaku dalam kemajuan teknologi di jaman yang serba modern ini.

b. Bagi Jurusan Pendidikan Teknik Elektronika

1. Terciptanya alat inovatif dan bermanfaat sebagai sarana ilmu pengetahuan.
2. Sebagai wujud partisipasi dalam pengembangan dibidang IPTEK.

2. Secara Praktis

a. Bagi Dunia Usaha, Industri, dan Kesehatan

1. Terciptanya alat sebagai sarana peningkatan teknologi dalam dunia industri maupun kesehatan
2. Sebagai bentuk kontribusi terhadap industri dalam mewujudkan pengembangan teknologi kesehatan.

G. Keaslian Gagasan

Proyek akhir ini berjudul “ Perancangan dan Implementasi Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan *Pulse sensor* Berbasis Arduino Uno ” merupakan asli gagasan dan rancangan penulis sendiri. Penelitian tentang pendeteksi denyut nadi manusia yang sebelumnya telah dilakukan oleh beberapa orang/ahli. Berikut ini beberapa penelitian yang mendasari penulisan tugas akhir ini.

1. Wibi Wicaksono dan I Komang Somawirata Jurusan Teknik Elektro, Intitut Teknologi Nasional Malang. Tahun 2011 yang berjudul

Perancangan dan Pembuatan Alat Penghitung Detak Jantung dengan Bipolar Standart Lead Berbasis Mikrokontroler atmega 8535. Berdasarkan hasil penelitian dapat disimpulkan sistem ini mampu mendeteksi denyut jantung manusia, namun alat ini masih banyak terjadinya *error* karena banyak komponen yang kurang mendukung.

2. I Agustinus Kalangi Program Studi Teknik Informatika, Sekolah Tinggi Manajemen Informatika Komputer Parna Raya Manado. Tahun 2014 yang berjudul : Analisa Pemanfaatan Mikrokontroller Atmega 8535 Pada Pemeriksaan Denyut Nadi Manusia. Berdasarkan hasil penelitian dapat disimpulkan sistem ini mampu mendeteksi denyut jantung manusia. Pada penelitian ini digunakan sensor jari berupa photodiode dan LED sebagai sensor detak jantung. Namun alat ini masih bersifat umum, belum adanya perbedaan antar usia.
3. Helmy Nurbani, Hafidudin ST.,MT, Sugondo Hadiyoso ST., MT, Fakultas Ilmu Terapan Universitas Telkom, yang berjudul : Perancangan Alat Pendeteksi Denyut Nadi Berbasis Mikrokontroler. Berdasarkan hasil penelitian alat ini mampu mendeteksi denyut nadi, namun tidak adanya perbedaan usia.

Dari beberapa penelitian tersebut, penelitian alat ukur dengan menggunakan teknologi arduino masih perlu dilakukan. Hal ini dikarenakan implementasi arduino lebih mudah digunakan dalam pembuatan instrumen alat ukur. Kebanyakan dari penelitian alat ukur terdahulu, masih menggunakan mikrokontroler yang berbasis pada IC Atmega 8535 dan lain-lain. Alat ukur

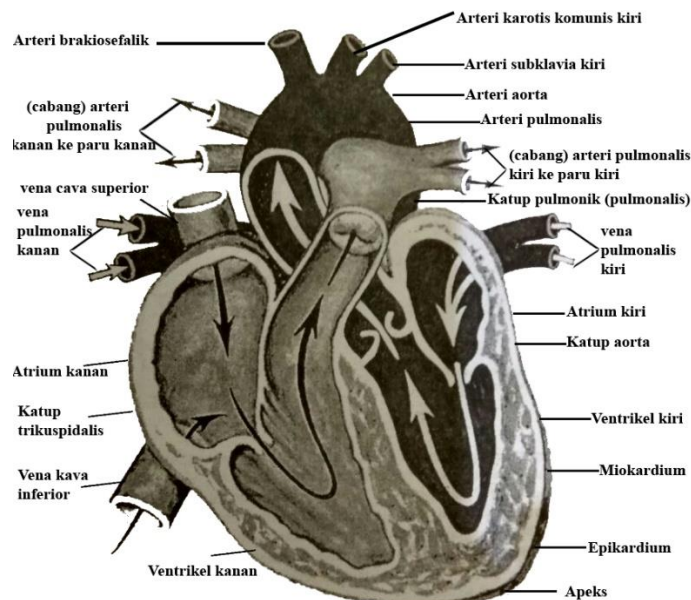
detak jantung yang digunakan dari beberapa penelitian di atas masih menggunakan photodiode, rangkaian kontrolnya masih menggunakan rangkaian analog yaitu rangkaian yang memproses sinyal analog, akan tetapi pada penelitian ini penulis menggunakan modul *pulse sensor* yaitu sensor yang memang dikhususkan untuk mengukur detak jantung sehingga rangkaiannya menjadi lebih sederhana.

BAB II

PENDEKATAN PEMECAHAN MASALAH

A. Sirkulasi Darah dalam Tubuh

Jantung adalah organ yang berupa otot, berbentuk kerucut, berongga, dengan pangkal di atas dan puncaknya di bawah miring kesebelah kiri. Jantung terletak di dalam rongga dada diantara kedua paru-paru, di belakang tulang dada, dan lebih menghadap ke kiri daripada ke kanan. Jantung berfungsi untuk memompa darah keseluruh tubuh melalui pembuluh darah. Satu detakan jantung kita menunjukkan satu pengiriman darah ke seluruh tubuh. Setiap hari, 2000 galon darah yang membawa oksigen dan nutrisi mengalir melalui pembuluh darah yang menghubungkan berbagai organ dan bagian tubuh lainnya. Inilah alasan mengapa jantung dan sistem sirkulasi darah (atau sistem kardiovaskuler) adalah penyokong utama kehidupan manusia.



Gambar 1. Anatomi Jantung Manusia
(Brunner dan Suddarth, 2010)

Pada gambar 1 adalah anatomi jantung manusia, jantung adalah sebuah pompa yang memiliki empat bilik. Dua bilik yang terletak di atas disebut Atrium, dan dua yang di bawah disebut Ventrikel. Jantung juga dapat dibagi menjadi dua bagian, yaitu bagian kanan yang bertugas memompa darah ke paru-paru, dan bagian kiri yang bertugas memompa darah ke seluruh tubuh manusia. Atrium dan ventrikel masing-masing akan dipisahkan oleh sebuah katup, sedangkan sisi kanan dan kiri jantung akan dipisahkan oleh sebuah sekat yang dinamakan dengan septum. Katup jantung berfungsi terutama agar darah yang telah terpompa tidak kembali masuk ke dalam lagi. Ketika darah dipompa keluar dari jantung pada arteri atau dikenal dengan pembuluh nadi teraba suatu gelombang denyut dan denyut ini dapat teraba pula pada tempat dimana pembuluh arteri melintas, misalnya arteri radialis yaitu di sebelah depan pergelangan tangan dan ujung jari. Saat keadaan ini volume darah pada ujung jari bertambah atau menggumpal. Kemudian sebaliknya pada saat jantung tidak memompa darah volume darah pada ujung jari menjadi lebih kecil.

Meraba gelombang denyut pada arteri, dapat dihitung kecepatan jantung yang berbeda-beda karena dipengaruhi oleh aktifitas seseorang dan juga oleh makanan, umur, dan emosi. Tabel 1 merupakan perbedaan denyut nadi manusia berdasarkan usia.

Denyut arteri dapat menjadi lebih cepat atau lebih lambat ketika seseorang sedang gelisah, hilang kesadaran atau koma, ada gangguan pada jantungnya atau menderita panas. Berdasarkan wawancara yang dilakukan penulis terhadap tenaga medis Dr. Theodorus dan Dr. Besly Sinuhaji, untuk pasien koma mempunyai

denyut jantung yang cenderung lebih lambat dibandingkan manusia normal, di bawah 60 kali per menit.

Tabel 1. Perbedaan Denyut Nadi Manusia

No	Umur	Jumlah denyut/menit (BPM)		
		Normal	Abnormal	
			<i>Bradikardia</i>	<i>Takikardia</i>
1	Bayi (0 bulan - < 1 Tahun)	80 – 160	< 80	> 160
2	Anak (1 Tahun - < 10 Tahun)	80 – 130	<80	>130
3	Dewasa (10 Tahun ke atas)	60 – 100	< 60	> 100

Sumber: Dr. Theodorus dan Dr. Besly Sinuhaji

Dalam alat pendeteksi denyut nadi manusia ini, untuk pembacaan denyut nadi per menit memanfaatkan *pulse sensor* yang ditempelkan di ujung jari manusia, seperti pada uraian di atas ujung jari manusia merupakan salah satu tempat melintasnya pembuluh *arteri* atau nadi, ujung jari dipilih karena pemasangan *pulse sensor* lebih mudah.

B. Nadi

Nadi merupakan gelombang darah yang dihasilkan oleh kontraksi ventrikel kiri jantung. Umumnya, gelombang nadi memiliki volume sekucup dan jumlah darah yang memasuki arteri pada setiap kontraksi ventrikel. Komplians arteri adalah kemampuan arteri untuk mengerut dan mengembang. Ketika arteri kehilangan distensibilitas atau daya rangsang, seperti yang terjadi pada individu lansia, dibutuhkan tekanan yang lebih besar untuk memompa darah ke dalam arteri.

Pada individu yang sehat, nadi menggambarkan denyut jantung; artinya, frekuensi nadi sama dengan frekuensi kontraksi ventrikel jantung. Akan tetapi, pada beberapa jenis penyakit kardiovaskular, denyut jantung dan nadi dapat berbeda. Sebagai contoh, jantung pasien dapat menghasilkan gelombang nadi yang sangat lemah atau kecil yang tidak dapat dideteksi pada nadi perfier yang letaknya jauh dari jantung. Pada keadaan ini, perawat harus mengkaji denyut jantung dan nadi perfier. Nadi perfier adalah nadi yang letaknya jauh dari jantung, contohnya nadi yang terdapat di kaki, pergelangan tangan, atau leher. Sebaliknya nadi apikal adalah nadi pusat yang berlokasi di apeks jantung. (Barbara Kozier, 2010 : 672)

1. Faktor Yang Mempengaruhi Nadi

Frekuensi nadi digambarkan dalam satuan beat per menit (BPM). Frekuensi nadi bervariasi berdasarkan sejumlah faktor (Barbara Kozier, 2010 : 678)

- a. Usia. Seiring peningkatan usia, frekuensi nadi akan turun secara bertahap. Umur seseorang dapat mempengaruhi jumlah denyut nadi per menit. Frekuensi jumlah denyut nadi pada bayi dibandingkan dengan orang dewasa memiliki perbedaan dimana jumlah denyut nadi pada bayi lebih cepat dibandingkan orang dewasa normal.
- b. Olahraga. Normalnya, frekuensi nadi akan meningkat dengan aktivitas. Frekuensi nadi pada atlet profesional kerap lebih rendah daripada orang biasa karena ukuran, kekuatan, dan efisiensi jantung mereka lebih besar. Aktivitas merupakan faktor yang sangat berpengaruh dalam

menentukan jumlah denyut nadi manusia. Seseorang dalam keadaan lari/berolahraga akan memiliki jumlah denyut nadi yang jauh lebih banyak dibandingkan dengan orang yang sedang duduk santai. Hal ini merupakan aktifitas tubuh yang berkaitan dengan proses oksigenasi terhadap aktivitas otot dan organ organ tubuh yang lain.

- c. Demam. Frekuensi nadi meningkat dalam merespons penurunan tekanan darah yang disebabkan oleh vasodilatasi perifer akibat peningkatan suhu tubuh dan karena peningkatan laju metabolisme. Suhu tubuh mempengaruhi proses fisiologis pada manusia termasuk frekuensi denyut nadi. Peningkatan atau penurunan suhu tubuh dapat mempercepat aktivitas organ tubuh dua kali dari normalnya.
- d. Medikasi. Sebagian obat dapat menurunkan frekuensi nadi, dan sebagian lain justru dapat meningkatkan denyut nadi. Sebagai contoh, kardiotonik (misalnya preparat digitalis) dapat menurunkan denyut jantung, sedangkan epinefrin dapat meningkatkan denyut jantung.
- e. Stres atau faktor psikis. Dalam merespons stres, stimulasi saraf simpatis akan meningkatkan aktivitas jantung secara keseluruhan. Kondisi stres meningkatkan frekuensi serta kekuatan denyut jantung. Rasa takut dan cemas serta persepsi nyeri yang hebat menstimulasi sistem saraf simpatis. Emosi seseorang dapat berpengaruh terhadap jumlah denyut nadi pada manusia. Orang yang sedang marah akan mempercepat jumlah detak jantung, yang kemudian berpengaruh pada jumlah denyut nadi. Orang dengan perasaan takut atau gugup juga dapat berdampak

pada jumlah detak jantung. Orang dengan perasaan takut atau gugup biasanya akan mempercepat frekuensi detak jantungnya.

- f. Perubahan posisi. Ketika seseorang duduk atau berdiri, darah biasanya akan berkumpul di pembuluh darah dependen pada sistem vena. Bendungan darah ini menyebabkan penurunan aliran balik vena menuju jantung yang disusul dengan penurunan tekanan darah dan peningkatan denyut nadi yang berlangsung sesaat.
- g. Patologi. Penyakit tertentu, seperti kondisi jantung atau beberapa penyakit yang mengganggu okisegenasi dapat mengubah frekuensi nadi saat istirahat.

2. Lokasi Nadi

Nadi dapat diukur pada sembilan lokasi (Barbara Kozier, 2010 : 678-679)

- a. Temporalis, tempat arteri temporalis melewati tulang temporal kepala. Lokasinya berada pada sisi superior (atas) dan lateral (jauh dari garis tengah) mata.
- b. Karotis, pada sisi leher tempat arteri karotis menjalar di antara trakea dan otot sternokleidomastoideus.
- c. Apikal, pada apeks jantung. Pada orang dewasa, arteri ini terletak di sisi kiri dada, sekitar 8 cm ke arah kiri sternum (tulang dada) dan ruang interkosta (area di antara tulang iga) keempat, kelima, atau keenam. Pada anak yang berusia 7-9 tahun, nadi apikal terletak di ruang interkosta keempat atau kelima. Brakialis, pada bagian dalam otot bisep lengan atau di tengah-tengah ruang antekubiti.

- d. Radialis, tempat arteri radialis menjalar sepanjang tulang radial, sejajar ibu jari di bagian dalam pergelangan tangan.
- e. Femoralis, tempat arteri femoralis menjalar sepanjang ligamentum inguinale.
- f. Poplitea, tempat arteri poplitea melintas di belakang lutut.
- g. Tibialis posterior, pada permukaan medial pergelangan kaki, tempat arteri tibialis posterior melewati belakang malleolus medialis.
- h. Pedis (dorsalis pedis) tempat arteri dorsalis pedis menjalar di sepanjang tulang kaki, pada garis khayal yang ditarik dari tengah-tengah pergelangan kaki menuju ruang antara ibu jari dan jari telunjuk kaki.
- i. Nadi radialis paling sering digunakan pada orang dewasa. Nadi tersebut mudah ditemukan pada kebanyakan orang dan mudah diakses.

3. Mengkaji Denyut Nadi

Normalnya, nadi dipalpasi (diraba) dengan memberi tekanan berkekuatan sedang menggunakan ketiga jari tengah. Permukaan jari paling distal merupakan area yang paling sensitif dalam mendeteksi nadi. Tekanan yang berlebihan akan menutupi nadi, sedangkan tekanan yang terlalu kecil tidak akan mampu mendeteksi nadi. Sebelum mengkaji denyut nadi saat beristirahat, harus mengambil posisi yang nyaman. Tujuan mengetahui denyut nadi seseorang adalah untuk mengetahui kerja jantung, menentukan diagnosa awal, untuk segera mengetahui adanya kelainan-kelainan pada seseorang.

Denyut jantung yang sangat cepat (misalnya, diatas 100 kali/menit) disebut *takikardia*. Denyut jantung orang dewasa 60 kali/menit atau kurang

disebut *bradikardia*. Apabila seseorang mengalami *takikardia* atau *bradikardia*, nadi apikal perlu diuji. (Barbara Kozier, 2010 : 680)

C. Pulse Sensor

Pulse Sensor pada dasarnya adalah alat medis yang berfungsi untuk memantau kondisi denyut jantung manusia. Rangkaian dasar dari sensor ini dibangun menggunakan photodiode dan LED. Sensor ini bekerja berdasarkan prinsip pantulan sinar LED. Kulit dipakai sebagai permukaan reflektif untuk sinar LED. Kepadatan darah pada kulit akan mempengaruhi reflektifitas sinar LED. Aksi pemompaan jantung mengakibatkan kepadatan darah meningkat. Pada saat jantung memompa darah, maka darah akan mengalir melalui pembuluh arteri dari yang besar hingga kecil seperti di ujung jari. Volume darah pada ujung jari bertambah maka intensitas cahaya yang mengenai photodiode akan kecil karena terhalang oleh volume darah, begitu pula sebaliknya. Keluaran sinyal dari photodiode kemudian dikuatkan oleh sebuah Op-Amp sehingga dapat dibaca oleh ADC mikrokontroler. Gambar 2 menunjukkan komponen perangkat *pulse sensor*.



Gambar 2. Komponen Perangkat Pulse Sensor
(<http://pulsesensor.com/>)

Perangkat *Pulse Sensor* terdiri dari :

1. Kabel *24-inch color-coded*, dengan konektor standar (0,1” pitch).
Dengan kabel ini perangkat *pulse sensor* pun semakin mudah dipasang dengan perangkat mikrokontroler, tanpa adanya proses menyolder.
2. Klip untuk telinga. Dimana alat dapat ditempelkan pada telinga sehingga denyut pada telinga dapat diperoleh dan diolah menjadi grafik pada layar display.
3. 2 buah Velcro dots (tempelan pada kain)
4. 3 buah stiker transparan. Dimana penggunaannya berfungsi untuk menutup komponen depan pada *pulse sensor* pada tangan yang basah agar komponen tidak rusak.

D. LCD (Liquid Cristal Display)

Display elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD adalah salah satu jenis display elektronik yang dibuat dengan teknologi CMOS logic yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari back-lit. LCD berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.



Gambar 3. Modul LCD 16x2

<http://elektronika-dasar.web.id/lcd-liquid-cristal-display/>

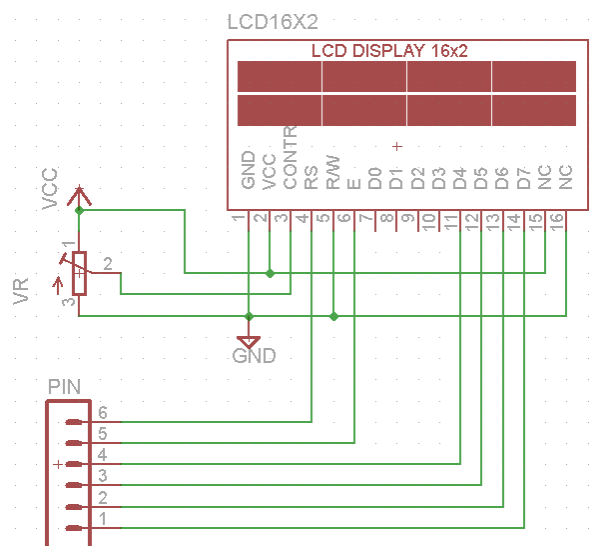
Gambar 3 adalah modul LCD, pengendali kontroler LCD dalam modul LCD terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD. Mikrokontroler pada suatu LCD dilengkapi dengan memori dan register. Memori yang digunakan mikrokontroler internal LCD adalah :

1. DDRAM (Display Data Random Access Memory) merupakan memori tempat karakter yang akan ditampilkan berada.
2. CGRAM (Character Generator Random Access Memory) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan.
3. CGROM (Character Generator Read Only Memory) merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh pabrikan pembuat LCD (Liquid Cristal Display) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam CGROM.

Register control yang terdapat dalam suatu LCD diantaranya adalah:

1. Register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD pada saat proses penulisan data atau tempat status dari panel LCD dapat dibaca pada saat pembacaan data.
2. Register data yaitu register untuk menuliskan atau membaca data dari atau ke DDRAM. Penulisan data pada register akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya. (Agus Purnomo, 2012)

1. Konfigurasi PIN LCD 16x2



Gambar 4. Konfigurasi PIN LCD 16x2

<http://www.bagusprehan.com/2013/12/konfigurasi-pin-lcd-16x2.html>

Tabel 2. Keterangan Pin-Pin LCD 16x2

No. Pin	Nama	Keterangan
Pin 1	GND	Sebagai jalur <i>power supply ground</i>
Pin 2	VCC	Sebagai jalur <i>power supply +5V</i>
Pin 3	VEE	Pengatur Kontras LCD
Pin 4	RS	Register Select, H=baca, L=instruksi
Pin 5	RW	Read/Write, H=baca, L=tulis
Pin 6	E	Enable Signal
Pin 7 - Pin 14	D0-D7	Data Bit 0 sampai Data Bit 7
Pin 15	A(+)	Anoda (back light)
Pin 16	A(-)	Katoda (back light)

(Sumber: Bagus Prehan. 2013)

Gambar 4 merupakan konfigurasi pin LCD 16x2, keterangan dari masing-masing pin dijelaskan dalam tabel 2. Berikut dijelaskan mengenai fungsi dari masing-masing PIN pada LCD, yaitu:

- a. **Pin data** adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- b. **Pin RS (*Register Select*)** berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *low* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data.

- c. **Pin R/W (*Read Write*)** berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data.
- d. **Pin E (*Enable*)** digunakan untuk memegang data baik masuk atau keluar. Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke ground, sedangkan tegangan catu daya ke LCD sebesar 5 Volt.

2. Prinsip Kerja LCD

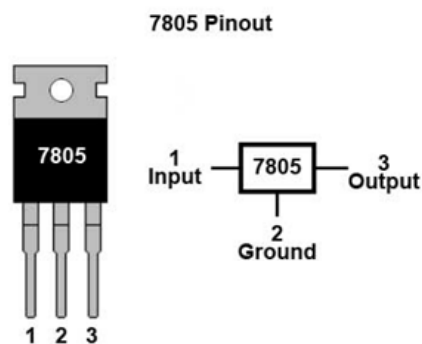
Proses *delay* karakter pada LCD diatur oleh pin E, RS, dan RW. Jalur ini digunakan untuk memberitahukan LCD bahwa sedang mengirimkan sebuah data, untuk mengirimkan sebuah data ke LCD, maka melalui program, E harus dibuat berlogika *low* “0” dan kemudian *setting* pada dua jalur kontrol yang lain yaitu RS dan RW. Ketika dua jalur yang lain telah siap, *set* kaki E dengan logika “1” dan tunggu untuk beberapa waktu tertentu (sesuai dengan *data sheet* dari LCD tersebut) dan berikutnya *set* kaki E ke logika *low* “0” lagi. Jalur RS adalah jalur *Register Select*. Ketika RS berlogika *low* “0”, data akan dianggap sebagai sebuah perintah atau instruksi khusus (seperti *clear screen*, posisi kursor). Ketika RS berlogika *high* “1”, data yang dikirim adalah data *text* yang akan ditampilkan pada *display* LCD. Sebagai contoh, untuk menampilkan huruf “T” pada layar LCD maka RS harus di-*set* logika *high* “1”. Jalur RW adalah jalur kontrol *Read/Write*. Ketika RW berlogika *low* “0”, maka informasi pada *bus data* akan dituliskan pada layar LCD. Ketika RW berlogika *high* “1”, maka program akan melakukan pembacaan memori LCD. Sedangkan pada

aplikasi umum pin RW selalu diberi logika *low* “0”, hal ini digunakan untuk *men-set* agar informasi selalu dituliskan pada LCD. Kemudian *bus data* terdiri dari 4 atau 8 jalur (bergantung pada *mode* operasi yang dipilih oleh *user*). Pada kasus bus data 8 *bit*, jalur diacukan sebagai DB0 sampai dengan DB7.

E. Catu Daya

Catu daya (*power supply*) adalah suatu rangkaian elektronika yang dapat menghasilkan tegangan keluar yang stabil, tegangan keluaran dapat berupa tegangan AC maupun tegangan DC. *Supply* daya atau tegangan catu suatu rangkaian elektronika yang berubah-ubah besarnya (baik berubah membesar maupun mengecil) dapat menyebabkan pengaruh yang sifatnya merusak rangkaian elektronika yang dicatunya.

Fungsi dari regulator tegangan merupakan komponen yang berfungsi untuk menstabilkan tegangan. Seri 78XX adalah regulator tegangan positif dengan tiga terminal seperti yang ditunjukkan pada gambar 5 Regulator ini memiliki kemampuan mengeluarkan arus yang besarnya bervariasi sesuai dengan tipe yang diberikan oleh pabrik.



Gambar 5 . Konfigurasi Pin pada 7805
<http://www.mpja.com/7805-5V-Voltage/>

Tabel 3. Karakteristik Regulator Tegangan Seri 78xx

Tipe	V Out (V)	I Out (A)			V in (V)	
		78xxC	78Lxx	78Mxx	Min	Max
7805	5	1	0,1	0,5	7,5	20
7806	6	1	0,1	0,5	8,6	21
7808	8	1	0,1	0,5	10,6	23
7809	9	1	0,1	0,5	11,7	24
7810	10	1	0,1	0,5	12,7	25
7812	12	1	0,1	0,5	14,8	27
7815	15	1	0,1	0,5	18	30
7818	18	1	0,1	0,5	21	33
7824	24	1	0,1	0,5	27,3	38

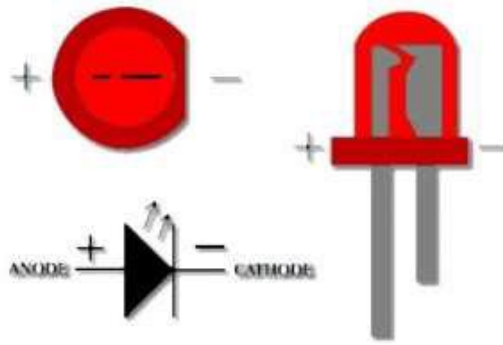
Tabel 3 merupakan karakteristik regulator tegangan seri 78XX. Dalam proyek akhir ini besarnya catu daya yang dibutuhkan sebesar +5 volt sehingga menggunakan IC regulator seri 7805 yang dapat diperoleh dalam kemasan TO-220 plastik dan logam. IC regulator 7805 dapat mengeluarkan arus melebihi 0,5A apabila dilengkapi dengan peredam *heatshink* yang memadai dan pada daya kurang atau sama dengan 15 watt.

Rangkaian terpadu (*integrate Circuit* = IC) tipe 7805 ini adalah regulator yang dapat menstabilkan tegangan searah positif dengan masukan +7,5 volt sampai +12 volt dengan keluaran +5 volt. IC regulator seri 7805 sesuai tabel 3 yang mempunyai karakteristik sebagai berikut:

1. Menstabilkan tegangan positif dengan memasukan tegangan +7,5 volt sampai +20 volt.
2. Tegangan keluaran IC 7805 adalah +5 volt DC teregulasi.
3. Arus keluaran melebihi 0,5A.
4. Penggunaan *intern* terhadap pembebasan lebih termik.
5. Tidak memerlukan tambahan komponen *ekstern*.
6. Pembatas arus hubungan singkat *intern*.

F. Light Emitting Dioda (LED)

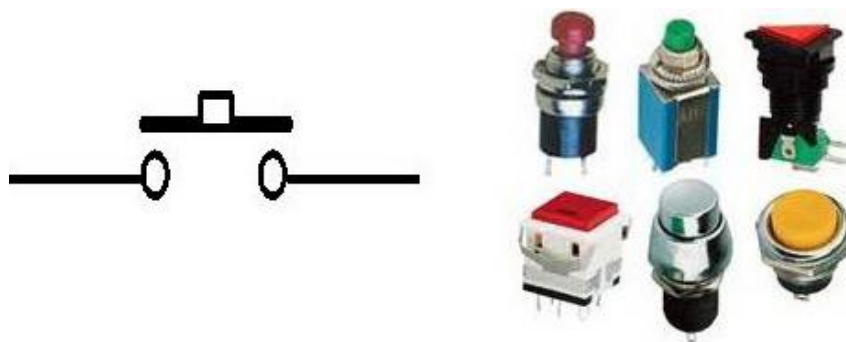
LED adalah dioda yang dapat memancarkan cahaya pada saat mendapat arus bias maju (*forward bias*). LED dapat memancarkan cahaya karena menggunakan dopping *galium*, *arsenic* dan *phosporus*. Jenis dopping yang berbeda dapat menghasilkan cahaya dengan warna yang berbeda. LED merupakan salah satu jenis dioda, sehingga hanya akan mengalirkan arus listrik satu arah saja. LED akan memancarkan cahaya apabila diberikan tegangan listrik dengan konfigurasi *forward bias*. Berbeda dengan dioda pada umumnya, kemampuan mengalirkan arus pada LED cukup rendah yaitu maksimal 20 mA. Apabila LED dialiri arus lebih besar dari 20 mA maka LED akan rusak, sehingga pada rangkaian LED dipasang sebuah resistor sebagai pembatas arus. Simbol dan bentuk fisik dari LED dapat dilihat pada gambar 6. Pada perancangan alat pengukur denyut nadi manusia digunakan 1 buah LED *superbright* 3mm dengan warna merah sebagai indikator keadaan denyut nadi.



Gambar 6. Simbol dan Bentuk Fisik LED
(<http://elektronika-dasar.web.id/komponen/led-light-emitting-dioda/>)

G. Saklar Push On

Saklar Push ON merupakan komponen elektronika yang berfungsi untuk menghubungkan dan memutuskan dua titik atau lebih dalam suatu rangkaian elektronika. Salah satu jenis saklar adalah saklar Push ON yaitu saklar yang hanya akan menghubungkan dua titik atau lebih pada saat tombolnya ditekan dan pada saat tombolnya tidak ditekan maka akan memutuskan dua titik atau lebih dalam suatu rangkaian elektronika. Simbol saklar Push ON ditunjukkan pada gambar 7 Simbol dan Bentuk Saklar Push ON.



Gambar 7. Simbol Dan Bentuk Saklar Push On
<http://elektronika-dasar.web.id/limit-switch-dan-saklar-push-on/>

Saklar push ON dapat berbentuk berbagai macam, ada yang menggunakan tuas dan ada yang tanpa tuas. Saklar push ON sering diaplikasikan pada tombol-tombol perangkat elektronik digital. Saklar push ON juga dikenal sebagai saklar push button. Salah satu contoh penggunaan saklar push ON adalah pada keyboard komputer, keypad printer, matrik keypad, tombol kontrol pada DVD player dan lain sebagainya. (Agus Purnomo, 2012)

H. Arduino Uno

Arduino Uno adalah arduino board yang menggunakan mikrokontroler ATmega328. Arduino Uno memiliki 14 pin digital (6 pin dapat digunakan sebagai output PWM), 6 input analog, sebuah 16 MHz osilator kristal, sebuah koneksi USB, sebuah konektor sumber tegangan, sebuah header ICSP, dan sebuah tombol reset. Arduino Uno memuat segala hal yang dibutuhkan untuk mendukung sebuah mikrokontroler. Hanya dengan menghubungkannya ke sebuah komputer melalui USB atau memberikan tegangan DC dari baterai atau adaptor AC ke DC sudah dapat membuanya bekerja.



Gambar 8. Arduino Uno

<http://febriadisantosa.weebly.com/knowledge/arduino-uno>

Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai *USB-to-serial converter* untuk komunikasi serial ke komputer melalui port USB. Tampak atas dari arduino uno dapat dilihat pada Gambar 8 Arduino Uno. Tabel 4 merupakan data teknis atau spesifikasi board Arduino UNO.

Tabel 4. Spesifikasi Arduino UNO

Mikrokontroler	ATmega328
Tegangan Operasi	5V
Tegangan <i>Input</i> (recommended)	7 - 12 V
Tegangan <i>Input</i> (limit)	6-20 V
Pin digital I/O	14 (6 diantaranya pin PWM)
Pin Analog <i>input</i>	6
Arus DC per pin I/O	40 Ma
Arus DC untuk pin 3.3 V	150 Ma
Flash Memory	32 KB dengan 0.5 KB digunakan untuk bootloader
SRAM	2 KB
EEPROM	1 KB
Kecepatan Pewaktuan	16 Hz

1. Pin Masukan dan Keluaran Arduino Uno

Masing-masing dari 14 pin digital arduino uno dapat digunakan sebagai masukan atau keluaran menggunakan fungsi `pinMode()`, `digitalWrite()` dan `digitalRead()`. Setiap pin beroperasi pada tegangan 5 volt. Setiap pin mampu menerima atau menghasilkan arus maksimum sebesar 40 mA dan memiliki resistor *pull-up* internal (diutus secara default) sebesar 20-30 KOhm. Sebagai tambahan, beberapa pin masukan digital memiliki kegunaan khusus yaitu:

- a. Komunikasi serial: pin 0 (RX) dan pin 1 (TX), digunakan untuk menerima(RX) dan mengirim(TX) data secara serial.

- b. External Interrupt: pin 2 dan pin 3, pin ini dapat dikonfigurasi untuk memicu sebuah interrupt pada nilai rendah, sisi naik atau turun, atau pada saat terjadi perubahan nilai.
- c. Pulse-width modulation (PWM): pin 3,5,6,9,10 dan 11, menyediakan keluaran PWM 8-bit dengan menggunakan fungsi `analogWrite()`.
- d. Serial Peripheral Interface (SPI): pin 10 (SS), 11 (MOSI), 12 (MISO) dan 13 (SCK), pin ini mendukung komunikasi SPI dengan menggunakan SPI library.
- e. LED: pin 13, terdapat built-in LED yang terhubung ke pin digital 13. Ketika pin bernilai HIGH maka LED menyala, sebaliknya ketika pin bernilai LOW maka LED akan padam.

Arduino Uno memiliki 6 masukan analog yang diberi label A0 sampai A5, setiap pin menyediakan resolusi sebanyak 10 bit (1024 nilai yang berbeda). Secara default pin mengukur nilai tegangan dari ground (0V) hingga 5V, walaupun begitu dimungkinkan untuk mengganti nilai batas atas dengan menggunakan pin AREF dan fungsi `analogReference()`. Sebagai tambahan beberapa pin masukan analog memiliki fungsi khusus yaitu pin A4 (SDA) dan pin A5 (SCL) yang digunakan untuk komunikasi *Two Wire Interface* (TWI) atau *Inter Integrated Circuit* (I2C) dengan menggunakan Wire library.

2. Sumber Daya dan Pin Tegangan Arduino Uno

Arduino uno dapat diberi daya melalui koneksi USB (*Universal Serial Bus*) atau melalui *power supply eksternal*. Jika arduino uno dihubungkan ke kedua sumber daya tersebut secara bersamaan maka arduino uno akan memilih

salah satu sumber daya secara otomatis untuk digunakan. *Power supply external* (yang bukan melalui USB) dapat berasal dari adaptor AC ke DC atau baterai. Adaptor dapat dihubungkan ke soket power pada arduino uno. Jika menggunakan baterai, ujung kabel yang dibubungkan ke baterai dimasukkan kedalam pin GND dan Vin yang berada pada konektor power.

Arduino uno dapat beroperasi pada tegangan 6 sampai 20 volt. Jika arduino uno diberi tegangan di bawah 7 volt, maka pin 5V akan menyediakan tegangan di bawah 5 volt dan arduino uno mungkin bekerja tidak stabil. Jika diberikan tegangan melebihi 12 volt, penstabil tegangan kemungkinan akan menjadi terlalu panas dan merusak arduino uno. Tegangan rekomendasi yang diberikan ke arduino uno berkisar antara 7 sampai 12 volt.

Pin-pin tegangan pada arduino uno adalah sebagai berikut:

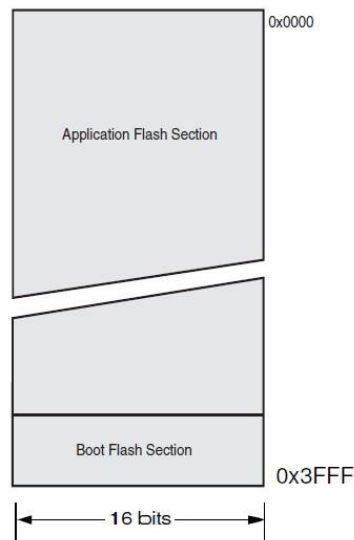
- a. Vin adalah pin untuk mengalirkan sumber tegangan ke arduino uno ketika menggunakan sumber daya eksternal (selain dari koneksi USB atau sumber daya yang teregulasi lainnya). Sumber tegangan juga dapat disediakan melalui pin ini jika sumber daya yang digunakan untuk arduino uno dialirkan melalui soket power.
- b. 5V adalah pin yang menyediakan tegangan teregulasi sebesar 5 volt berasal dari regulator tegangan pada arduino uno.
- c. 3V3 adalah pin yang menyediakan tegangan teregulasi sebesar 3,3 volt berasal dari regulator tegangan pada arduino uno.
- d. GND adalah pin ground.

3. Peta Memori Arduino Uno

Arduino Uno adalah arduino board yang menggunakan mikrokontroler ATmega328. Maka peta memori arduino uno sama dengan peta memori pada mikrokontroler ATmega328.

4. Memori Program

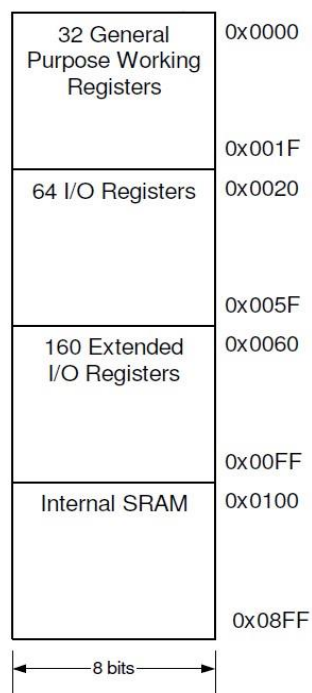
ATMega328 memiliki 32K byte *On-chip In-System Reprogrammable Flash Memory* untuk menyimpan program. Memori *flash* dibagi kedalam dua bagian, yaitu bagian program *bootloader* dan aplikasi seperti terlihat pada Gambar 9. *Bootloader* adalah program kecil yang bekerja pada saat sistem dimulai yang dapat memasukkan seluruh program aplikasi ke dalam memori prosesor.



Gambar 9. Peta Memori Program ATmega 328

5. Memori Data

Memori data ATmega328 terbagi menjadi 4 bagian, yaitu 32 lokasi untuk register umum, 64 lokasi untuk register I/O, 160 lokasi untuk register I/O tambahan dan sisanya 2048 lokasi untuk data SRAM internal. Register umum menempati alamat data terbawah, yaitu 0x0000 sampai 0x001F. Register I/O menempati 64 alamat berikutnya mulai dari 0x0020 hingga 0x005F. Register I/O tambahan menempati 160 alamat berikutnya mulai dari 0x0060 hingga 0x00FF. Sisa alamat berikutnya mulai dari 0x0100 hingga 0x08FF digunakan untuk SRAM internal. Peta memori data dari ATmega 328 dapat dilihat pada Gambar 10. Peta Memori Data ATmega 328.



Gambar 10. Peta Memori Data ATmega 328.

6. Memori Data EEPROM

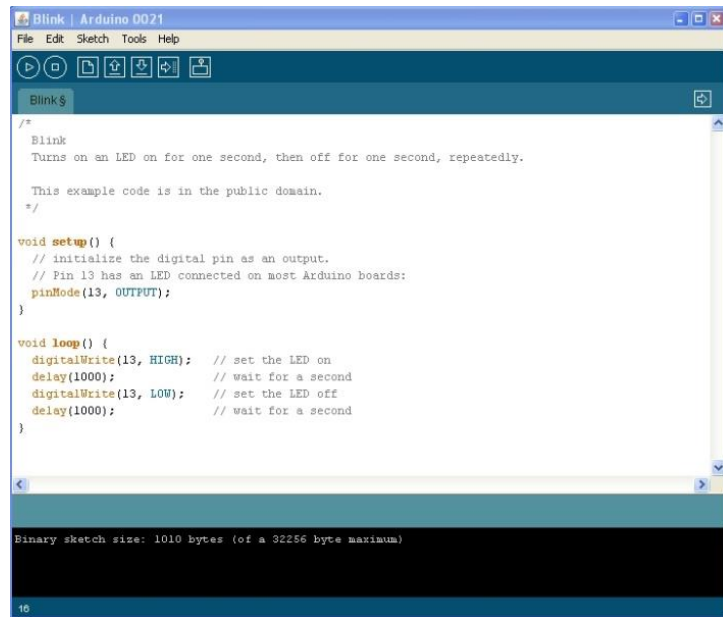
Arduino uno terdiri dari 1 KByte memori data EEPROM. Pada memori EEPROM, data dapat ditulis/dibaca kembali dan ketika catu daya dimatikan, data terakhir yang ditulis pada memori EEPROM masih tersimpan pada memori ini, atau dengan kata lain memori EEPROM bersifat *nonvolatile*. Alamat EEPROM dimulai dari 0x000 hingga 0x3FF.

I. Perangkat Lunak (*Software*)

1. Bahasa Pemrograman Arduino

Arduino IDE menggunakan bahasa pemrograman C dengan versi yang telah di sederhanakan dengan bantuan pustaka-pustaka (*libraries*) Arduino, sehingga lebih mudah dalam belajar pemrograman.

Mikrokontroler diprogram menggunakan bahasa pemrograman arduino yang memiliki kemiripan *syntax* dengan bahasa pemrograman C. Pada perancangan perangkat lunak ini menggunakan *software* Arduino IDE 1.0.5 sebagai *programmer* sekaligus *compiler* ke file .hex. Bahasa yang digunakan adalah bahasa C. Bahasa C menghasilkan objek kode yang sangat kecil dan dieksekusi sangat cepat. Bahasa C digunakan untuk sistem programming pada sistem program yang tertanam (*embedded system*). *Sketch* program bahasa C yang ditulis menggunakan *software* Arduino kemudian di verifikasi terlebih dahulu oleh program arduino tersebut. Kemudian *sketch* program tersebut diupload ke dalam alat penghitung denyut nadi. Gambar 10 merupakan tampilan jendela arduino yang di dalamnya terdapat beberapa baris program.



Gambar 11. Tampilan Jendela Arduino.
<http://www.hendriono.com/blog/post/apa-itu-arduino#isi1>)

2. Struktur

Setiap program arduino (biasa disebut *sketch*) mempunyai dua buah fungsi yang harus ada, yaitu:

- a. Void setup() { ... } ,

Semua kode didalam kurung kurawal akan dijalankan hanya satu kali ketika program arduino dijalankan untuk pertama kalinya.

- b. Void loop() { ... } ,

Fungsi ini dijalankan setelah setup (fungsi void setup) selesai. Setelah dijalankan satu kali fungsi ini akan dijalankan kembali, dan lagi secara terus menerus sampai catu daya (*power*) dilepaskan.

3. Syntax

Berikut ini adalah elemen bahasa c yang dibutuhkan untuk format penulisan:

- a. // (komentar satu baris),

Kadang diperlukan untuk memberi catatan pada diri sendiri apa arti dari kode-kode yang dituliskan. Cukup menuliskan dua buah garis miring dan apapun yang kita ketikan dibelakangnya akan diabaikan oleh program.

- b. /* (komentar banyak baris),

Jika Anda mempunyai banyak catatan, maka hal tersebut dapat dituliskan pada beberapa baris sebagai komentar. Semua hal yang terletak di antara dua simbol tersebut akan diabaikan oleh program.

- c. { ... } atau kurung kurawal,

Digunakan untuk mendefinisikan kapan blok program mulai dan berakhir (digunakan juga pada fungsi dan pengulangan).

- d. ; (titik koma),

setiap baris kode harus diakhiri dengan tanda titik koma (jika ada titik koma yang hilang maka program tidak akan bisa dijalankan).

4. Variabel

Sebuah program secara garis besar didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas. Variabel inilah yang digunakan untuk memudahkannya.

a. Int (*integer*),

Digunakan untuk menyimpan angka dalam 2 byte (16 bit). Tidak mempunyai angka desimal dan menyimpan nilai dari -23.768 s/d 32.767.

b. Long,

Digunakan ketika integer tidak mencukupi lagi. Memakai 4 byte (32 bit) dari memori RAM dan mempunyai rentang nilai dari -2.147.648 s/d 2.147.483.647.

c. Boolean,

Variable sederhana yang digunakan untuk menyimpan nilai *TRUE* (benar) atau *FALSE* (salah). Sangat berguna karena hanya menggunakan 1 bit dari RAM.

d. Float,

Digunakan untuk angka desimal (*floating point*). Memakai 4 byte (32 bit) dari RAM dan mempunyai rentang nilai dari -3,4028235E+38 s/d 3,4028235E+38.

e. Char (*character*),

Menyimpan 1 karakter menggunakan kode ASCII (misalnya 'A' = 65). Hanya memakai 1 byte (8 bit) dari RAM.

5. Operator Matematika

Operator yang digunakan untuk memanipulasi angka

- a. $=$ (sama dengan),

Membuat sesuatu menjadi sama dengan nilai yang lain (misalnya: $x = 10 * 2$, $x = 20$).

- b. $\%$ (persen),

Menghasilkan sisa dari hasil pembagian suatu angka yang lain (misalnya : $12 \% 10$, ini akan menghasilkan angka 2).

- c. $+$ (plus),

Digunakan saat operasi penjumlahan.

- d. $-$ (minus),

Digunakan saat operasi pengurangan.

- e. $*$ (asteris),

Digunakan saat operasi perkalian.

- f. $/$ (garis miring),

Digunakan saat operasi pembagian.

6. Operator Pembandingan

Digunakan untuk membandingkan nilai logika.

- a. $==$ (sama dengan),

misalnya: $12 == 10$ adalah *FALSE* (salah) atau $12 == 12$ adalah *TRUE* (benar).

b. \neq (tidak sama dengan),

misalnya: $12 \neq 10$ adalah *TRUE* (benar) atau $12 \neq 12$ adalah *FALSE* (salah).

c. $<$ (lebih kecil dari),

misalnya: $12 < 10$ adalah *FALSE* (salah) atau $12 < 12$ adalah *FALSE* (salah) atau $12 < 14$ adalah *TRUE* (benar).

d. $>$ (lebih besar dari),

misalnya: $12 > 10$ adalah *TRUE* (benar) atau $12 > 12$ adalah *FALSE* (salah) atau $12 > 14$ adalah *FALSE* (salah).

7. Struktur Pengaturan

Program sangat tergantung pada pengaturan apa yang akan dijalankan berikutnya. Berikut ini adalah elemen dasar pengaturan.

a. If ... else,

Dengan format seperti berikut ini:

If(kondisi) { ... }

Else if(kondisi) { ... }

Else { ... }

Struktur seperti diatas program akan menjalankan kode yang ada di dalam kurung kurawal jika kondisinya *TRUE*, dan jika tidak (*FALSE*) maka akan diperiksa apakah kondisi pada else if dan jika kondisinya *FALSE* maka kode pada *else* yang akan dijalankan.

b. For,

Dengan format penulisan sebagai berikut:

```
For(int i = 0; i < #pengulangan; i++) { ... }
```

Digunakan bila Anda ingin melakukan pengulangan kode program di dalam kurung kurawal beberapa kali, ganti #pengulangan dengan jumlah pengulangan yang diinginkan. Melakukan perhitungan ke atas (++) atau ke bawah (--).

8. Digital

a. pinMode(pin, mode),

Digunakan untuk menetapkan mode dari suatu pin, pin adalah nomor pin yang akan digunakan sebagai port dari 0 s/d 19 (pin analog 0 s/d 5 adalah 14 s/d 19). Mode yang bisa digunakan adalah *input* atau *output*.

b. digitalWrite(pin, value),

Ketika sebuah pin ditetapkan sebagai *output*, pin tersebut dapat dijadikan *high* (+5 volt) atau *low* (*ground*).

c. digitalRead(pin),

Ketika sebuah pin ditetapkan sebagai *input*, maka Anda dapat menggunakan kode ini untuk mendapatkan nilai pin tersebut apakah *high* (+5 volt) atau *low* (*ground*).

9. Analog

Arduino adalah mesin digital tetapi mempunyai kemampuan untuk beroperasi di dalam analog.

a. `analogWrite(pin, value)`,

Beberapa pin pada arduino mendukung PWM (*pulse width modulation*) yaitu pin 3, 5, 6, 9, 10, 11. Ini dapat merubah pin hidup (*on*) atau mati (*off*) dengan sangat cepat sehingga membuatnya dapat berfungsi layaknya keluaran analog. *Value* (nilai) pada format kode tersebut adalah angka antara 0 (0% duty cycle ~ 0 volt) dan 255 (100% duty cycle ~ 5 volt).

b. `analogRead(pin)`,

Pada saat pin analog ditetapkan sebagai *INPUT* dapat membaca keluaran voltase-nya. Keluarannya berupa angka antara 0 (untuk 0 volt) dan 1024 (untuk 5 volt).

BAB III

KONSEP PERANCANGAN ALAT

Perancangan Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan Pulse sensor Berbasis Arduino Uno menggunakan metode bangun ADDIE (Dick dan Carey, 2010). Model ADDIE (*Analysis, Design, Development, Implementation, and Evaluation*) adalah kerangka yang berisi daftar proses generik menggunakan desainer instruksional dan pengembang pelatihan. Metode ini merupakan pedoman deskriptif untuk membangun alat pelatihan dan dukungan kinerja yang efektif dalam lima tahap. Secara urut metode tersebut adalah Analisis, Desain, Pengembangan, Pelaksanaan, dan Evaluasi. Identifikasi kebutuhan yang diperlukan. Kebutuhan tersebut kemudian dianalisis untuk mendapatkan komponen secara spesifik, selanjutnya dilakukan perancangan perangkat keras dan perangkat lunak, dilanjutkan dengan pembuatan alat dan pengujian alat.

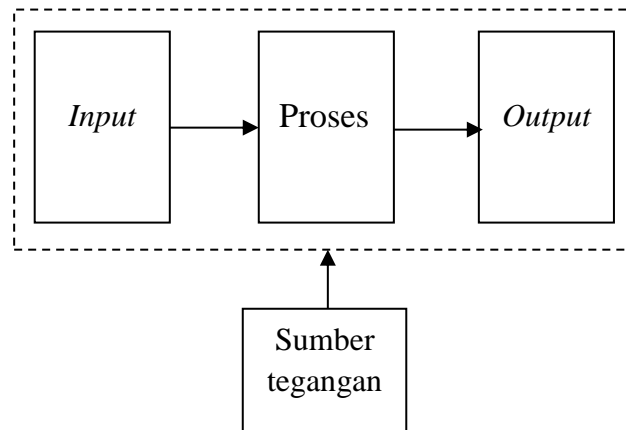
A. Analisa

Untuk merancang alat pendeteksi denyut nadi manusia dibutuhkan mikrokontroler arduino uno, *pulse sensor* atau sensor detak jantung, *push button*, LED sebagai indikator, dan LCD. Sedangkan untuk perangkat lunak dibutuhkan *software* Arduino IDE. Gambar 12 adalah blok diagram yang digunakan untuk identifikasi dan menganalisa kebutuhan dan bahan yang akan diperlukan.

1. Identifikasi Kebutuhan

Pengidentifikasian kebutuhan untuk merancang alat pendeteksi denyut nadi manusia dibedakan menjadi dua yaitu:

a. Perangkat Keras (*Hardware*)



Gambar 12. Blok Diagram

Pengidentifikasian kebutuhan dari gambar 12 dapat dikelompokkan menjadi 5 bagian:

1. Bagian *Input*

- a. Sensor denyut nadi atau *pulse sensor* yang dapat bekerja untuk menghitung jumlah denyut nadi manusia.
- b. *Push button* sebagai tombol untuk pemilihan usia, dalam hal ini usia yang ditentukan adalah bayi, anak-anak dan dewasa.

2. Bagian *Proses*

Sistem kontrol yang mempunyai algoritma untuk memproses data dan mengirimkan data jumlah denyut nadi yang terukur dalam satuan BPM (*Beats per Minute*).

3. **Bagian Output**

LCD sebagai penampil jumlah denyut nadi manusia, dan LED sebagai indikator.

4. **Bagian Catu Daya**

Catu Daya berfungsi untuk memberikan sumber daya tegangan dan arus ke rangkaian alat pendeteksi denyut nadi manusia. Sumber tegangan dan arus harus disesuaikan dengan konsumsi daya yang digunakan oleh rangkaian pada alat pendeteksi denyut nadi. Masukan tegangan yang digunakan pada alat ini sebesar 5 V.

5. **Media untuk Merealisasikan Alat**

Media sebagai tempat untuk meletakkan semua rangkaian dari alat pendeteksi denyut nadi.

b. **Perangkat Lunak (*Software*)**

Pada perancangan perangkat lunak ini menggunakan *software* Arduino IDE 1.0.5 sebagai *programmer* sekaligus *compiler* ke file .hex. Bahasa yang digunakan adalah bahasa C. Bahasa C menghasilkan objek kode yang sangat kecil dan dieksekusi sangat cepat. Bahasa C digunakan untuk sistem programming pada sistem program yang tertanam (*embedded system*). *Sketch* program bahasa C yang ditulis menggunakan *software* Arduino kemudian di verifikasi terlebih dahulu oleh program arduino tersebut. Kemudian *sketch* program tersebut diupload ke dalam alat penghitung denyut nadi.

Software Proteus ini digunakan untuk mendesain layout jalur PCB rangkaian sistem minimum ATmega 328p. Dengan menggunakan *software* ini kita bisa lebih presisi saat membuat layout jalur PCB.

2. Analisis Kebutuhan

Berdasarkan identifikasi kebutuhan di atas maka diperoleh beberapa analisis kebutuhan terhadap pengembangan alat yang akan dibuat sebagai berikut:

a. Bagian *Input*

- 1) Sensor yang diperlukan dari alat ini adalah Sensor denyut nadi atau *pulse sensor* di karenakan *Pulse Sensor* dapat memonitoring denyut nadi dalam keadaan *real time*. *Pulse Sensor* harus terintegrasi oleh Arduino untuk dapat memonitoring denyut nadi. Bagian depan sensor yang berbentuk lambang hati adalah sisi yang akan terkontak dengan kulit. Pada bagian depan pun akan terlihat sebuah lubang bulat kecil, dimana LED akan bersinar dari belakang sensor dan ada kotak persegi kecil di bawah LED. Kotak tersebut adalah *ambient light* sensor yang sama persis digunakan dalam ponsel, tablet, dan juga laptop yang dimana berfungsi sebagai penyesuai kecerahan layar dalam pengaturan cahaya rendah. LED memeberikan sinar cahaya ke ujung jari atau daun telinga, dan sensor akan membaca cahaya yang memantul kembali.

- 2) *Push button* sebagai tombol untuk memilih usia berapa yang diinginkan sebelum menghitung denyut nadi.

b. Bagian Proses

Alat ini memerlukan alat atau komponen untuk mengolah data *Input* atau data *Output* yang akan dikeluarkan. Pada bagian proses ini penulis menggunakan Arduino Uno karena implementasi arduino lebih mudah digunakan dalam pembuatan instrumen ukur dibanding dengan menggunakan teknologi sejenis lainnya. Selain itu, saat ini masih belum banyak penelitian tentang pembuatan beberapa instrumen ukur yang dikendalikan dalam sebuah board arduino. Kebanyakan dari penelitian alat ukur terdahulu, masih menggunakan mikrokontroler yang berbasis pada IC Atmega 8535 dan lain-lain. Mikrokontroler tipe ini memiliki 14 pin *Input/Output* yang mana 6 pin dapat digunakan sebagai *output* PWM, 6 analog *input*, crystal osilator 16 Mhz, koneksi USB, Jack power, kepala ICSP, dan tombol reset. Arduino Bersifat *open source* dan mempunyai bahasa pemrograman berupa bahasa C, sehingga dengan fitur tersebut sudah cukup untuk diterapkan pada alat kontrol ini.

c. Bagian Output

LCD adalah sebuah display dot matrix yang difungsikan untuk menampilkan tulisan berupa angka atau huruf sesuai dengan yang diinginkan (sesuai dengan program yang digunakan untuk mengontrolnya). Ada beberapa LCD Alphanumeric yang umum

dipakai yaitu LCD 1x16 (1 baris dan 16 karakter), 2x16 (dua baris dan 16 karakter per baris) dan 4x20 (4 baris dan 20 karakter per baris). LCD yang digunakan dalam pembuatan alat pendeteksi denyut nadi manusia menggunakan LCD 16x2. LCD jenis dapat menampilkan data berupa angka maupun huruf dengan jumlah baris 2 dan jumlah kolom 16.

d. Bagian Catu Daya

Alat ini memerlukan catu daya yang keluarannya 5 volt. Pemilihan baterai 9 volt dapat menstabilkan tegangan *output* menjadi 5 volt. Catu daya menggunakan regulator *power supply* LM7805 dengan spesifikasi pendukung kerja sistem *output* DC +5V. IC regulator LM7805 digunakan karena dengan IC ini bisa menstabilkan tegangan *output* berkisar 5V.

e. Bagian dari Media sebagai Alat Pendukung

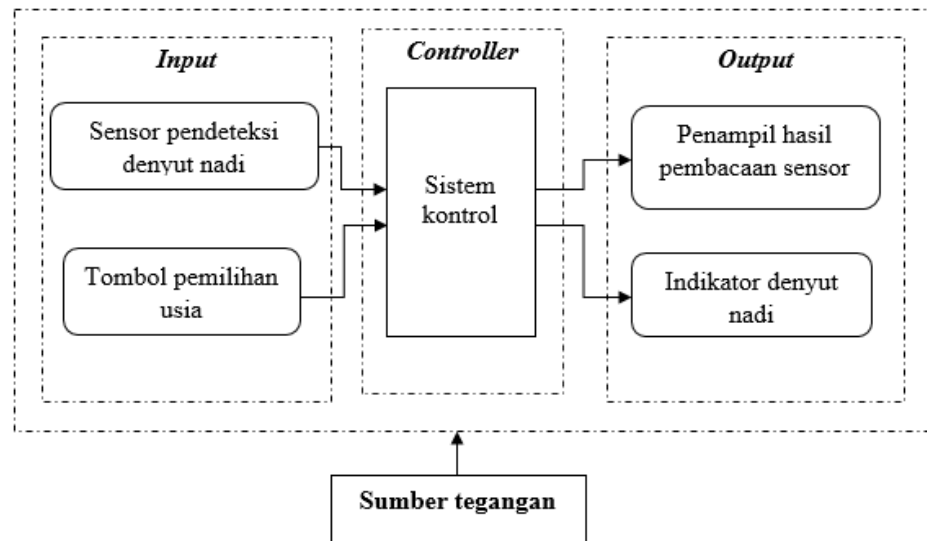
Media berupa box kotak diperlukan untuk menampung semua rangkaian. Box kotak yang dipilih berukuran 18 cm x 11 cm. Box ini sangat mudah didapatkan dipasaran selain itu harganya relatif murah. Bahan dari box ini tidak terlalu keras sehingga mempermudah untuk membuat lubang komponen dari pada bahan yang lain seperti alumunium, besi, baja, dan lainnya.

Tabel 5. Identifikasi Kebutuhan Alat

No	Rangkaian	Komponen	Spesifikasi
1	Catu daya	Baterai	9 volt
		IC Regulator	LM7805
		Saklar on/of	Ukuran kecil
		Konektor	Secukupnya
2	<i>System Minimum</i>	Mikrokontroler Arduino	Atmega328p
		Resistor	1K Ω
		Led	3mm
		Konektor	Header male, white housing
3	<i>Input</i>	Sensor <i>Pulse Sensor</i>	1. Deteksi jumlah denyut nadi permenit (BPM)
		<i>Push button</i>	Ukuran sedang
4	<i>Output</i>	LCD 16x2	<i>Alphanumeric 16x2</i>
		LED	<i>Superbright 3mm</i>
5	DII	Baut	3mm
		Mur	3mm
		<i>Box</i>	Ukuran 18 cm x 11 cm
		Kabel Pelangi	Secukupnya
		Kabel <i>jumper</i>	Secukupnya
		Pcb	Fiber FR4
		Box kotak	Ukuran besar, dan kecil

B. Perancangan Alat

Pembuatan alat pada proyek akhir ini terdiri dari beberapa perancangan. Perancangan yang dimaksud adalah meliputi perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*).



Gambar 13. Blok Diagram Sistem

Gambar 13 adalah diagram blok sistem keseluruhan yang terdiri *dari input, controller, output*, dan catu daya.

1. Blok *Input*

Blok *Input* terdiri dari sensor *pulse sensor*. Sensor ini akan mendeteksi jumlah denyut nadi kemudian akan menghasilkan *Output* berupa jumlah denyut nadi dalam satuan (BPM) yang akan dimasukkan ke pin analog dari Arduino untuk selanjutnya diproses oleh Arduino.

2. Blok *Controller*

Setelah menerima *Input* dari sensor, Arduino akan mengolah data dari *pulse sensor*. Rangkaian *pulse sensor* terdiri dari rangkaian sensor cahaya yang

berfungsi untuk mendeteksi aliran darah yang mengalir pada pembuluh nadi di ujung jari. Sensor cahaya bekerja berdasarkan prinsip pantulan sinar LED. Kulit dipakai sebagai permukaan reflektif untuk sinar LED. Kepadatan darah pada kulit akan mempengaruhi reflektifitas sinar LED. Kemudian hasil keluaran dari sensor cahaya akan dikuatkan oleh rangkaian amplifier dan dikirim ke rangkaian *Analog Digital Converter* (ADC) untuk mengubah tegangan ke sinyal digital. Sinyal digital akan diolah oleh mikrokontroler Arduino Uno untuk menghasilkan sinyal BPM.

3. **Blok *Output***

Terdapat 2 buah jenis *Output*. *Output* pertama yaitu LCD digunakan untuk menampilkan nilai dari pembacaan sensor. *Output* kedua yaitu LED sebagai indikator ada tidaknya denyut nadi.

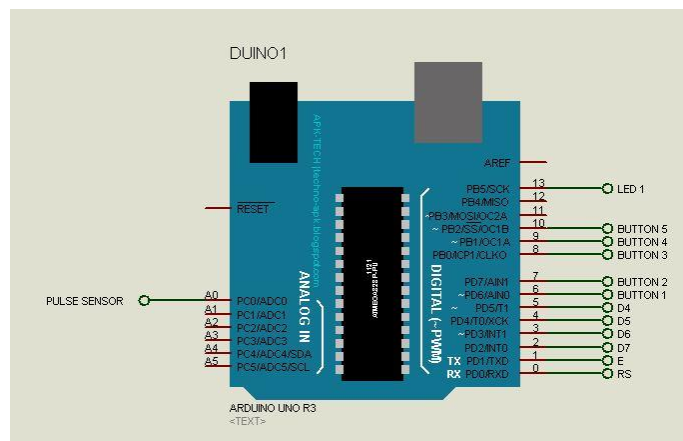
1. **Perancangan Perangkat Keras (*Hardware*)**

a. **Rangkaian Catu Daya**

power supply yang digunakan dalam perancangan alat ini menggunakan baterai 9v. Alat ini memerlukan catu daya yang keluarannya 5 volt. Agar keluaran *power supply* berada ditegangan 5v digunakan IC regulator LM7805. Pemilihan IC regulator LM7805 dapat menstabilkan tegangan *output*. Catu daya menggunakan regulator *power supply* LM7805 dengan spesifikasi pendukung kerja sistem *output* DC +5V. IC regulator LM7805 digunakan karena dengan IC ini bisa menstabilkan tegangan *output* berkisar 5V.

b. Rangkaian Pengendali berbasis Arduino

Rangkaian pengendali arduino uno berbasis Atmega 328p berfungsi sebagai otak dari alat pendeteksi denyut nadi manusia menggunakan arduino. Rangkaian ini bekerja pada tegangan 5 V yang bersumber dari rangkaian *power supply*. Gambar 14 menunjukkan rangkaian pengendali berbasis Atmega 328p. Tabel 6 menunjukkan nama pin beserta fungsinya.

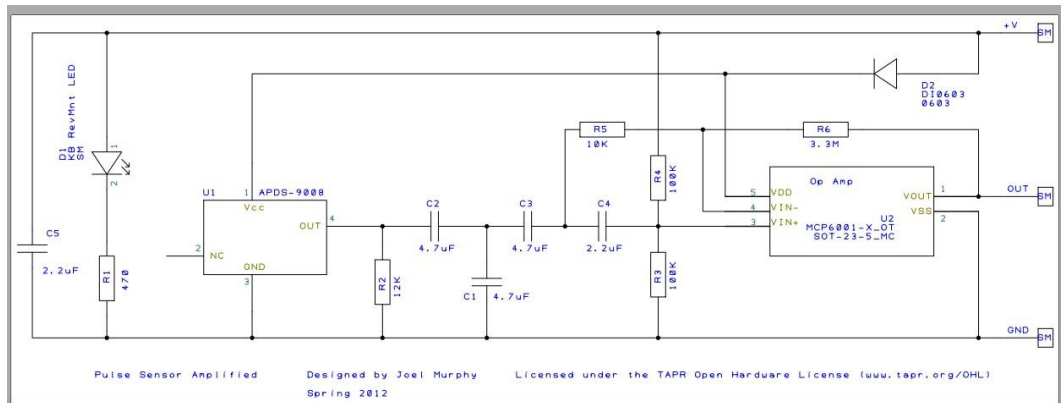


Gambar 14. Rangkaian Pengendali Arduino Uno

Tabel 6. Pin Pada Mikrokontroler Atmega 328p

No	Nama Pin	Fungsi
1	Pin A0	Masukan dari <i>Pulse Sensor</i>
2	Pin PB0,PB1,PB2	Masukan dari <i>Push button</i> untuk pemilihan usia
3	Pin PD6, dan Pin PD7	Tombol <i>start</i> dan tombol <i>close</i>
4	Pin PB5	Indikator LED untuk menentukan ada tidaknya denyut nadi
5	Pin PD0 s/d PD5	Tampilan ke LCD

c. Rangkaian *Pulse Sensor*



Gambar 15. Rangkaian *Pulse Sensor*
(<http://pulsesensor.com/>)

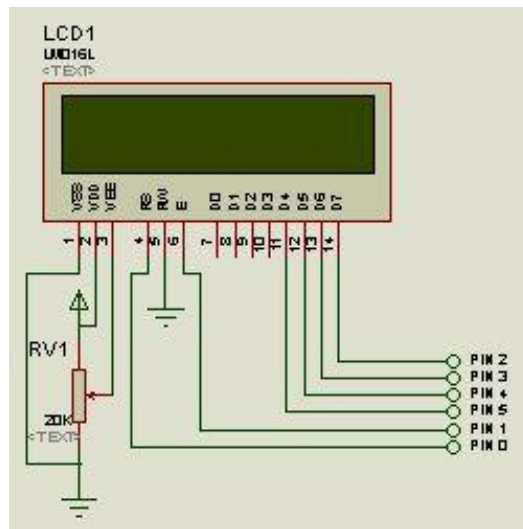
Gambar 15 adalah Rangkaian *pulse sensor* berfungsi untuk memantau kondisi denyut nadi manusia. Rangkaian dasar dari sensor ini dibangun menggunakan photodiode dan LED. Sensor ini bekerja berdasarkan prinsip pantulan sinar LED. Kulit dipakai sebagai permukaan reflektif untuk sinar LED. Kepadatan darah pada kulit akan mempengaruhi reflektifitas sinar LED. LED yang digunakan adalah jenis LED *superbright* berwarna hijau yang mampu menembus permukaan kulit. Untuk sensor cahaya digunakan IC APDS 9008 yang sangat sensitif terhadap perubahan cahaya. Untuk menguatkan hasil dari keluaran sensor APDS 9008 digunakan *Op-Amp* MCP601 menggunakan rangkaian *Inverting*. Hasil penguatan yang maksimal dibandingkan jenis *Op-Amp* lain. Keluaran dari *Op-Amp* berupa sinyal analog kemudian di olah oleh mikrokontroler arduino uno sehingga dapat menampilkan sinyal BPM. Catu daya dari sensor ini sebesar 5 V yang didapatkan dari rangkaian catu daya. Rangkaian *pulse sensor* ditunjukkan pada gambar 16, sedangkan

untuk rincian komponen yang digunakan pada rangkaian *pulse sensor* ditunjukkan pada tabel 7.

Tabel 7 . Rincian komponen yang digunakan pada rangkaian *Pulse sensor*

No.	Komponen	Fungsi pada alat
1	3,3 M Ω / ¼ W	Feedback penguat op-amp
2	10K Ω / ¼ W	Resistor input penguat op-amp
3	100K Ω / ¼ W	Resistor input penguat op-amp
4	100K Ω / ¼ W	Resistor input penguat op-amp
5	12K Ω / ¼ W	Low pass filter
6	470 Ω / ¼ W	Pembagi tegangan LED
7	4,7 F / 16V	Band pass filter
8	4,7 F / 16V	High pass filter
9	4,7 F / 16V	Low pass filter
10	4,7 F / 16V	High pass filter
11	2,2 F / 16V	Filter tegangan ripple
12	1N4004 / 1 A	Pengaman polaritas tegangan
13	LED 3mm	Indikator sensor detak jantung
14	MCP601	Penguat inverting sensor detak jantung
15	APDS9009	Sensor cahaya

d. Rangkaian LCD



Gambar 16 . Rangkaian LCD

Gambar 16 merupakan rangkaian *interface* antara Arduino dengan modul LCD. Potensio RV1 digunakan untuk mengatur kontras LCD. *Interface* antara LCD dan arduino melalui Pin seperti yang terlihat pada gambar 16. Data yang ditampilkan ke LCD adalah data pembacaan denyut nadi.

2. Perancangan Perangkat Lunak (*Software*)

Perancangan perangkat lunak menggunakan bahasa pemrograman C pada arduino, program yang telah dibuat kemudian dicompile sehingga akan diperoleh file dengan ekstensi *.ino. File ini inilah yang akan didownload ke Arduino Uno. Sebelum membuat program maka terlebih dahulu membuat algoritma sesuai dengan perancangan sistem tersebut. Kemudian algoritma program tersebut dituangkan kedalam diagram alir (*flowchart*) selanjutnya dibuat program dalam bahasa C.

C. Proses Pembuatan

Dalam proses pembuatan alat Pendeteksi denyut nadi, terlebih dahulu harus mempersiapkan beberapa alat dan bahan yang diperlukan

Tabel 8 . Alat dan Fungsinya

No.	Bahan	Fungsi
1	Komputer	Digunakan untuk menggambar rangkaian, membuat <i>layout PCB</i> dan membuat program termasuk program untuk membuat gambar rangkaian <i>schematic, layout PCB</i> , aplikasi Arduino sebagai kompiler sekaligus program untuk download ke hardware Arduino, <i>mendesign</i>
2	Multimeter	Digunakan untuk mengukur tegangan, mengecek komponen yang akan dipasang, mengecek jalu PCB.
3	IC Mikrokontroller ATmega328p	Digunakan sebagai <i>controller</i> .
4	<i>Pulse Sensor</i>	Digunakan untuk mendeteksi denyut nadi
5	<i>Push button</i>	Digunakan sebagai tombol pemilihan usia
6	LCD 16x2	Digunakan untuk menampilkan hasil penghitungan denyut nadi
7	LED 3mm	Digunakan sebagai indikator ada tidaknya denyut nadi
8	Konektor Sisir	Digunakan untuk menghubungkan rangkaian di PCB dengan board Arduino.
9	Komponen Elektronika	Digunakan sebagai hardware untuk melakukan aksi.
10	PCB	Digunakan untuk meletakkan komponen yang digunakan.
11	Baterai 9V	Digunakan sebagai sumber tegangan.
12	Saklar <i>On/Off</i>	Digunakan untuk menghidupkan atau mematikan alat deteksi denyut nadi

Adapun proses pembuatan alat sebagai berikut:

1. Merancang gambar skematik rangkaian
2. Menyiapkan alat dan bahan yang dibutuhkan
3. Mendesain layout PCB dan mencetak layout PCB
4. Memindahkan hasil cetakan pada kertas ke PCB
5. Melarutkan PCB menggunakan FeCl_3
6. Mengecek jalur PCB yang telah dilarutkan dengan menggunakan multimeter
7. Menguji kondisi komponen dengan alat ukur sebelum dipasang pada PCB, kemudian memasang dan menyolder komponen
8. Membuat program menggunakan komputer dengan program Arduino

D. Implementasi

Tahap Implementasi atau penerapan dalam proyek akhir ini mengabungkan semua komponen-komponen dari percobaan yang sudah dilakukan untuk dibuat menjadi sistem yang terintegrasi menjadi sebuah alat pendeteksi denyut nadi, serta membuat box sebagai media sebagai pendukung dengan *photoshop* terlebih dahulu. Bentuk box yang diinginkan menggunakan *box universal*.

4. Ukuran *box universal*
 - a. Panjang = 15 cm
 - b. Lebar = 10 cm
 - c. Tinggi = 7 cm

E. Pengujian Alat

Pengujian alat dilakukan untuk mendapatkan data penelitian serta mengetahui fungsi alat yang telah dibuat. Dalam pengujian dilakukan dengan dua pengujian yaitu:

1. Uji Fungsional

Pengujian dilakukan dengan cara menguji setiap bagian alat berdasarkan karakteristik dan fungsi masing-masing. Pengujian dilakukan untuk mengetahui apakah setiap bagian dari perangkat telah bekerja dengan fungsi yang diharapkan. Tabel 9 adalah uji fungsional yang dilakukan.

Tabel 9. Uji Fungsional

No	Alat	Indikator	Kebutuhan/Pengukuran
1	Catu Daya	Tegangan <i>output</i> 5 Volt	5 Volt
2	<i>Pulse Sensor</i>	Dapat menghitung jumlah denyut nadi manusia (BPM)	pengukuran dilakukan kepada a. bayi, normal = 80 – 160 /menit bradikardia = <80 /menit takikardia = >160 /menit b. anak-anak normal = 80 – 130 /menit bradikardia = <80 /menit takikardia = >130 /menit c. dewasa normal = 60 – 100 /menit bradikardia = <60 /menit takikardia = >100 /menit
3	LCD	Dapat menampilkan data jumlah denyut nadi manusia dan mengetahui kondisinya.	Mengukur denyut nadi berdasarkan usia yang telah ditetapkan.

4	<i>Push Button</i>	Dapat membedakan perbedaan usia	menampilkan perbedaan usia sesuai yang diinginkan.
5	LED	Dapat mengetahui ada tidaknya denyut nadi	Menampilkan denyut nadi

2. Uji Unjuk Kerja

Pengujian unjuk kerja alat dilakukan dengan cara melihat unjuk kerja alat. Hal-hal yang perlu diamati antara lain: rangkaian catu daya, arduino, *pulse sensor*. Tabel 10 adalah tabel uji unjuk kerja yang dilakukan.

Tabel 10. Unjuk Kerja

No	Aspek	Indikator
1	Saklar on	1. LCD <i>on</i> 2. <i>Pulse sensor on</i>
2	<i>Push button 1</i>	LCD menampilkan karakter “Dewasa”
3	<i>Push button 2</i>	LCD menampilkan karakter “Anak-anak”
4	<i>Push button 3</i>	LCD menampilkan karakter “Bayi”
5	<i>Push button START</i>	1. LED indikator denyut nadi menyala 2. LCD menampilkan karakter “Status denyut”
6	<i>Push button CLOSE</i>	Kembali ke menu utama
7	LED	1. Jika LED menyala tidak beraturan, artinya denyut nadi belum teridentifikasi, dan display LCD masih acak 2. Jika LED menyala secara beraturan atau stabil, artinya denyut nadi sudah teridentifikasi, dan LCD akan menampilkan data denyut nadi.

F. Spesifikasi Alat

Alat Pendeteksi Denyut Nadi ini mempunyai spesifikasi sebagai berikut:

1. Bahan pembuatan box alat ini menggunakan PCB dan box plastik *universal*.
2. Unit masukan :
 - a. *Pulse Sensor* untuk mengetahui denyut nadi seseorang dengan satuan *beat per minute* (BPM)
 - b. *Push button* sebagai tombol pemilihan usia
4. Sistem pengendali yang digunakan adalah ATmega328p sebagai pengendali dari *pulse sensor*
5. Unit keluaran :
 - a. LCD 16x2 untuk menampilkan hasil dari pembacaan sensor
 - b. LED sebagai indikator untuk mengetahui ada tidaknya denyut nadi seseorang.
6. Tegangan rangkaian yang digunakan adalah 5 VDC dengan baterai 9 Volt
7. Alat Pendeteksi Denyut Nadi ini akan melakukan kerjanya untuk menghitung denyut seseorang dalam satuan BPM. Pengukuran denyut nadi ini menggunakan *pulse sensor* yang ditempelkan di ujung jari manusia. Keluaran dari sensor tersebut akan dikirim dan diolah oleh mikrokontroler arduino uno. Kemudian data akan diolah dan diproses hasilnya akan

ditampilkan di LCD. Adapun beberapa ciri khas pada alat ini adalah:

- a. Menggunakan mikrokontroler Arduino, yang merupakan sebuah sistem minimum yang telah memiliki beberapa fasilitas dalam penerapannya, dapat digunakan dalam berbagai alat maupun karya lain.
- b. Menggunakan *pulse sensor* sebagai pendeteksi denyut nadi. Hasil pengolahan data dapat dilihat secara langsung melalui LCD.

G. Perincian Alat, Bahan dan Rincian Biaya

Dalam pembuatan proyek akhir ini penulis menggunakan alat dan bahan sesuai dengan rincian seperti berikut:

a. Alat yang digunakan dalam Pembuatan Proyek Akhir

Pada pembuatan alat Pendeteksi Denyut Nadi maka di perlukan alat yang ada pada tabel dibawah ini:

Tabel 11. Daftar Alat yang Dibutuhkan

No	Nama Bahan	Spesifikasi	Jumlah
1	IC Mikrokontroller	ATmega328p	1 Buah
2	<i>Pulse Sensor</i>		1 Buah
3	<i>Push Button</i>	Ukuran Sedang	5 Buah
4	LCD	Ukuran 16X2	1 Buah
5	LED	3mm	1 Buah
6	Kenektor sisir	-	Secukupnya

7	Komponen Elektronika	Resistor, IC regulator, LED, Jack DC	Secukupnya
8	Tenol	-	Secukupnya
9	PCB	FR4	Secukupnya
10	Baterai	9V	1 Buah
11	Saklar <i>On/Off</i>	Ukuran Kecil	1 Buah

b. Bahan yang Digunakan dalam Pembuatan Proyek Akhir

Pada pembuatan alat Pendeteksi Denyut Nadi maka di perlukan bahan yang ada pada tabel dibawah ini:

Tabel 12. Daftar Bahan yang Dibutuhkan

No	Nama Alat	Spesifikasi	Jumlah
1	Komputer	-	1 Unit
2	Bor mini	0.8 mm, 1 mm, 3 mm	1 Buah
3	Multimeter	Opticcom YX-360TR	1 Buah
4	Tang lancip	-	1 Buah
5	Tang potong	-	1 Buah
6	Solder	-	1 Buah
7	Atractor	-	1 Buah
8	Gergaji besi	-	1 Buah
9	Box Universal	-	2 Buah

c. Rincian Biaya

Dari tabel bahan yang digunakan pada proyek akhir ini, penulis merekap biaya yang untuk membeli bahan-bahan tersebut, dengan rincian sebagai berikut:

Tabel 13. Rincian Biaya

No	Nama Bahan	Spesifikasi	Jumlah	Harga
1	IC Mikrokontroller	ATmega328p	1 Buah	Rp. 180.000
2	<i>Pulse Sensor</i>	-	2 Buah	Rp. 300.000
3	<i>Push Button</i>	Ukuran Sedang	5 Buah	Rp. 5.000
4	LCD	Ukuran 16X2	1 Buah	Rp. 50.000
5	LED	3mm	1 Buah	Rp. 2.000
6	Kenektor sisir	-	Secukupnya	Rp. 15.000
7	Komponen Elektronika	Resistor, IC regulator, LED, Jack DC, konektor, kabel jumper, kabel strip, white hosing, dll	Secukupnya	Rp. 80.000
8	Tenol	-	Secukupnya	Rp. 12.000
9	PCB	FR4	Secukupnya	Rp. 20.000
10	Baterai	9V	1 Buah	Rp. 8.000
11	Saklar <i>On/Off</i>	Ukuran Kecil	2 Buah	Rp. 6.000
12	Box	Ukuran Kecil	1 Buah	Rp. 7.000
13	Box	Ukuran Besar	1 Buah	Rp. 15.000
TOTAL				Rp. 700.000

BAB IV

PROSES, HASIL, DAN PEMBAHASAN

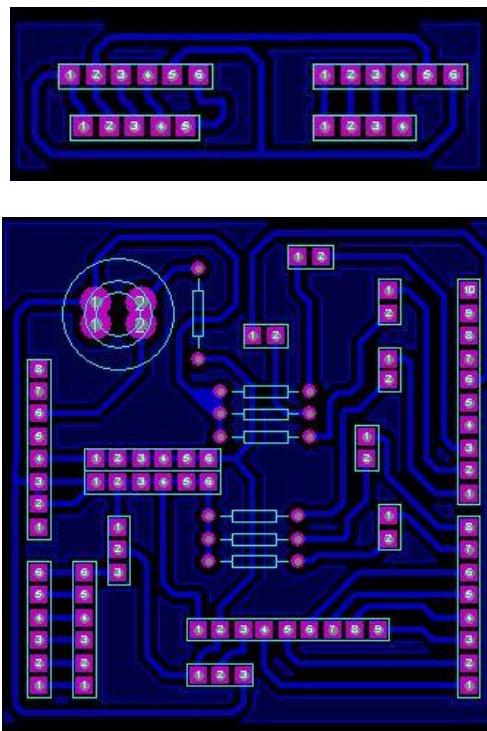
Tujuan dari pengujian dan pembahasan adalah untuk mengetahui kinerja alat baik secara per bagian blok rangkaian maupun sistem keseluruhan apakah sudah seperti yang diharapkan atau belum. Pengujian ini meliputi:

A. Perancangan

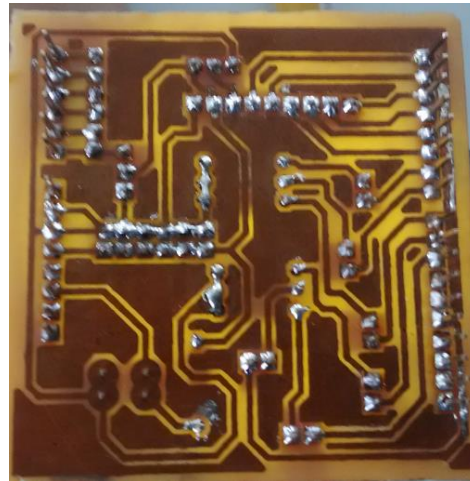
1. Perangkat Keras

a. *Layout* PCB

Setelah melakukan perancangan *layout* rangkaian menggunakan *software proteus* kemudian rancangan *layout* rangkian tersebut di *ares* supaya bisa mendesain *layout* di PCB .



Gambar 17. Pembuatan *Design* PCB dengan *Software Proteus*

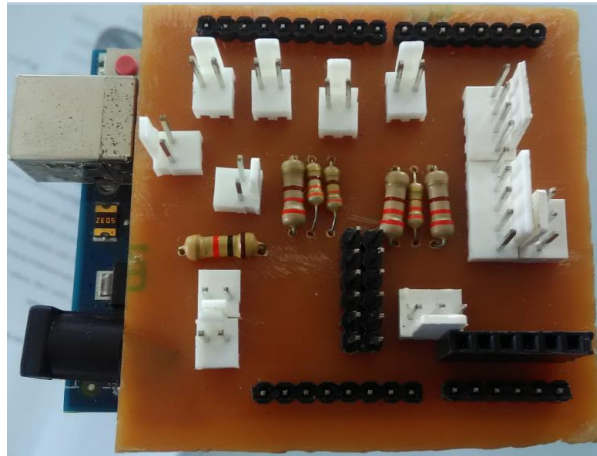


Gambar 18. Desain PCB

Sebelum melakukan pencetakan di PCB, terlebih dahulu harus mendesain dengan menggunakan *software proteus* agar hasil jalur PCB terlihat rapi, seperti yang terlihat pada gambar 17, Kemudian dilakukan pencetakan ke PCB jenis FR4 dengan teknik penyablonan, digunakannya teknik penyablonan agar mendapat hasil cetakan PCB yang maksimal, hasil pencetakan jalur ke PCB dapat dilihat pada gambar 18.

b. Perakitan Komponen

Setelah dilakukan pencetakan jalur pada PCB, langkah selanjutnya adalah pemasangan komponen dan penyolderan komponen pada board PCB:



Gambar 19. Perakitan Komponen di PCB

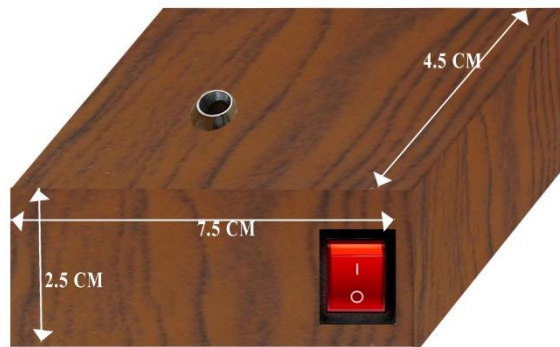
Perakitan komponen PCB dilakukan dengan menyolder terlebih dahulu komponen resistor, *jumper*, dan *white housing*. Gambar 18 menunjukkan perakitan PCB. Penyolderan dilakukan dengan sangat hati-hati agar tidak ada jalur yang *short*.

c. Perancangan Design Box

Melakukan perancangan design *box* dengan menggunakan *software photoshop*, kemudian merealisasikan design dengan menggunakan bahan *box universal*:



Gambar 20. Design box



Gambar 21. *Design Box* Adaptor

2. Perangkat Lunak

a. Algoritma Pemrograman

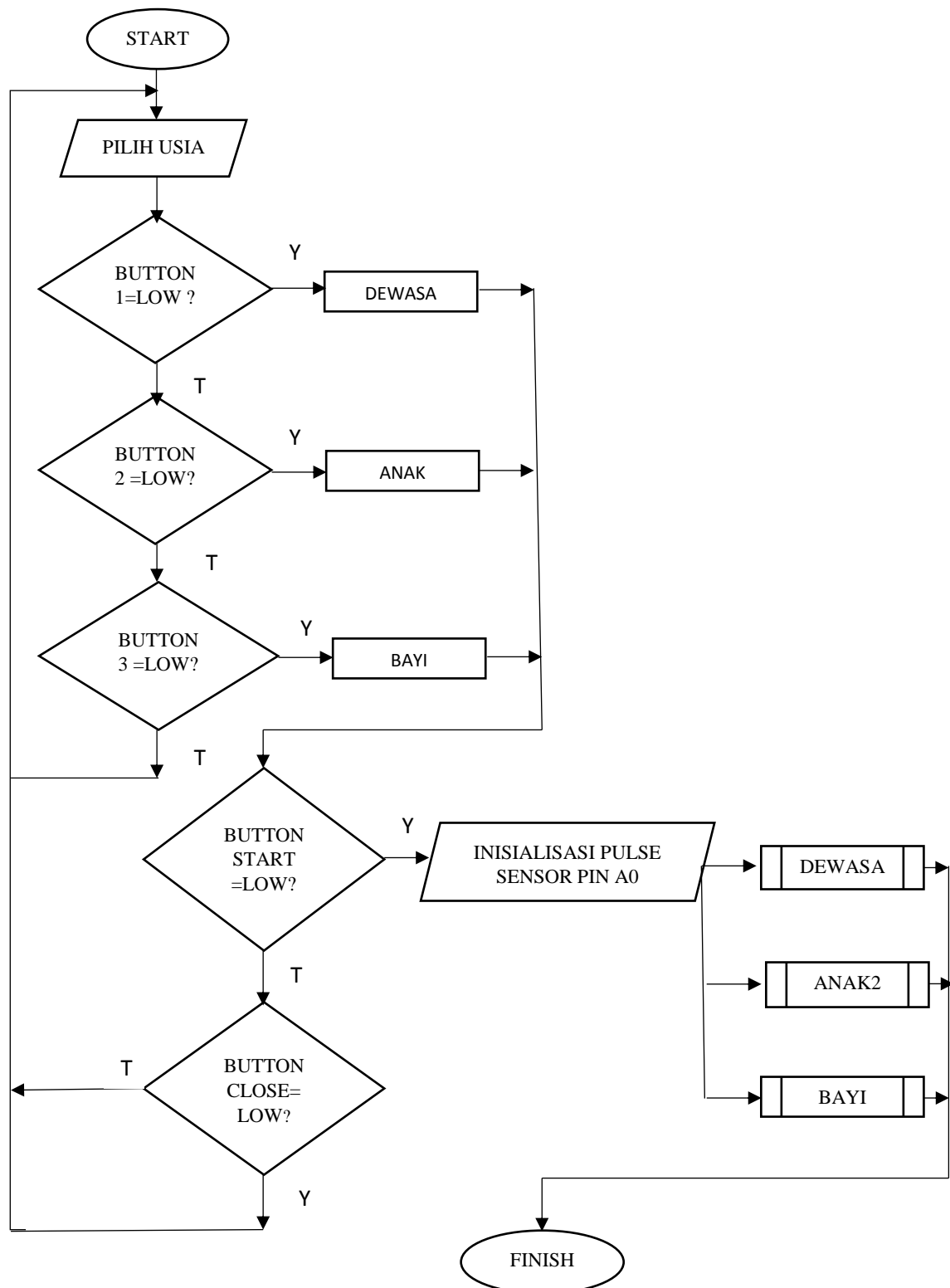
Pada proyek akhir ini yang dilakukan penulis pertama kali adalah membentuk algoritma, kemudian membuat *flowchart* yang sesuai dengan algoritma yang sudah dibuat. Setelah membuat algoritma dan *flowchart* kemudian menuliskan *pseudocode* menggunakan bahasa C Arduino.

1. Algoritma program utama

1. Memilih usia
2. Jika tombol *button* 1 ditekan maka LCD akan menampilkan “Dewasa”
3. Jika tombol *button* 2 ditekan maka LCD akan menampilkan LCD “Anak”
4. Jika tombol *button* 3 ditekan maka LCD akan menampilkan LCD “Bayi”
5. Jika tombol *button start* ditekan maka inisialisasi *pulse sensor*
6. Dan jika *button start* dan *button* 1 aktif maka masuk ke sub rutin dewasa

7. Dan jika *button start* dan *button 2* aktif maka masuk ke sub rutin anak
8. Dan jika *button start* dan *button 3* aktif maka masuk ke sub rutin bayi
9. Dan jika *button close* ditekan maka kembali ke menu pilih usia
10. Menampilkan status denyut nadi dan jumlah denyut nadi dalam satuan BPM.

2. Flowchart Program Utama



Gambar 22. Flowchart Sistem Keseluruhan

3. *Pseudocode*

Algoritma keseluruhan

Deklarasi

Nama keterangan : String, Sensor : Integer

Deskripsi

Read (pilih usia)

If *button 1* = Low then

Keterangan \leftarrow “Dewasa”

If *button 2* = Low then

Keterangan \leftarrow “Anak”

If *button 2* = Low then

Keterangan \leftarrow “Bayi”

If *button start* = Low then

Keterangan \leftarrow “Inisiaalisasi *pulse sensor*”

If *button start* = Low && *button 1* = Low then

Keterangan \leftarrow “Sub rutin dewasa”

If *button start* = Low && *button 2* = Low then

Keterangan \leftarrow “Sub rutin anak”

If *button start* = Low && *button 3* = Low then

Keterangan \leftarrow “Sub rutin bayi”

If *button close* = Low then

Keterangan \leftarrow “*return*”

Write (keterangan sensor)

4. Listing Program

Tabel 14. Listing Program Utama

Program utama	Interrupt
<pre> lcd.begin(16, 2); lcd.setCursor(0, 0); lcd.print("Pengukuran"); lcd.setCursor(0, 1); lcd.print("Denyut"); lcd.setCursor(7, 1); lcd.print("Nadi"); delay(2000); lcd.clear(); lcd.setCursor(0, 0); lcd.print("PILIH USIA"); delay(1000); lcd.clear(); pinMode(saklar1, INPUT_PULLUP); pinMode(saklar2, INPUT_PULLUP); pinMode(saklar3, INPUT_PULLUP); pinMode(mulai, INPUT_PULLUP); pinMode(berhenti, INPUT_PULLUP); pinMode(blinkPin, OUTPUT); </pre>	<pre> volatile int rate[10]; volatile unsigned long sampleCounter = 0; volatile unsigned long lastBeatTime = 0; volatile int P = 512; volatile int T = 512; volatile int thresh = 512; volatile int amp = 100; volatile boolean firstBeat = true; volatile boolean secondBeat = true; void interruptSetup(){ </pre>

Hasil program ditunjukkan pada tabel 14, menggunakan *software* Arduino. Program utama atau inisialisasi program sedangkan *interrupt* merupakan program pengaktif *pulse sensor*. *Source code* secara lengkap dapat dilihat di lampiran.

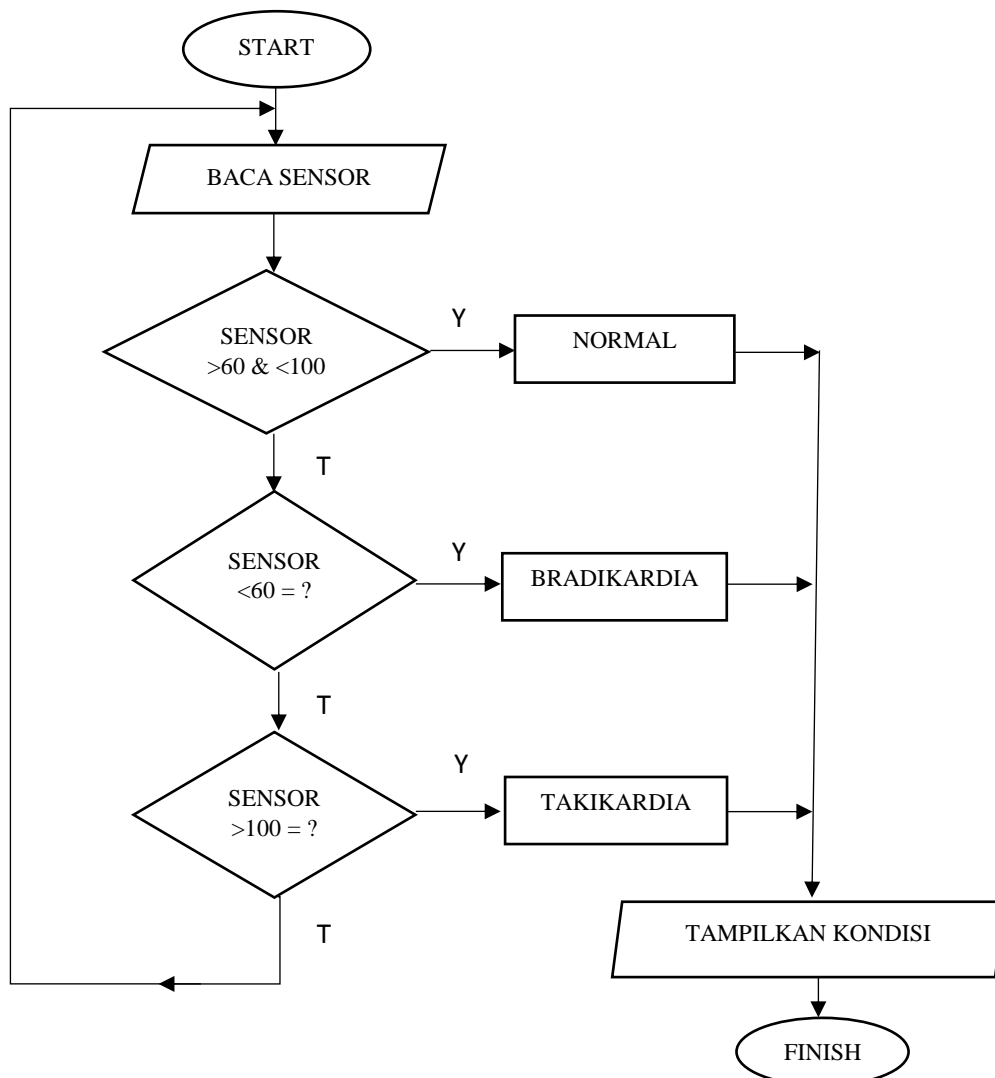
b. Sub Rutin Dewasa

1. Algoritma Sub Rutin Dewasa

Deskripsi → Sensor (BPM)

1. Baca sensor
2. Jika nilai sensor > 60 & < 100
3. LCD menampilkan kondisi "Normal"
4. Jika nilai sensor < 60
5. LCD menampilkan kondisi "Bradikardia"
6. Jika nilai sensor > 100
7. LCD menampilkan kondisi "Takikardia"
8. Tulis sensor dan keterangan

2. Flowchart Sub Rutin Dewasa



Gambar 23. Sub Rutin Kondisi Dewasa

3. *Pseudocode*

Deklarasi

Keterangan : string
Sensor : integer

Deskripsi

Read (sensor)

If sensor > 60 && < 100 then

Keterangan \leftarrow Normal

If sensor < 60 then

Keterangan \leftarrow Bradikardia

If sensor > 100 then

Keterangan \leftarrow Takikardia

Write (sensor, keterangan)

4. Listing Program

```
void dewasa() {
  lcd.setCursor(0, 0);
  lcd.print("Denyut:");
  lcd.setCursor(0, 1);
  lcd.print("Status:");
  if (BPM == 0 ) {
    lcd.setCursor(8, 1);
    lcd.print("DEAD !!");
  }
  else if (BPM < 60 ) {
    lcd.setCursor(8, 1);
    lcd.print("BRADIKARDIA  ");
  }
  else if (BPM > 100 ) {
    lcd.setCursor(8, 1);
    lcd.print("TAKIKARDIA ");
  }
  else {
    lcd.setCursor(8, 1);
    lcd.print("NORMAL      ");
  }
}
```

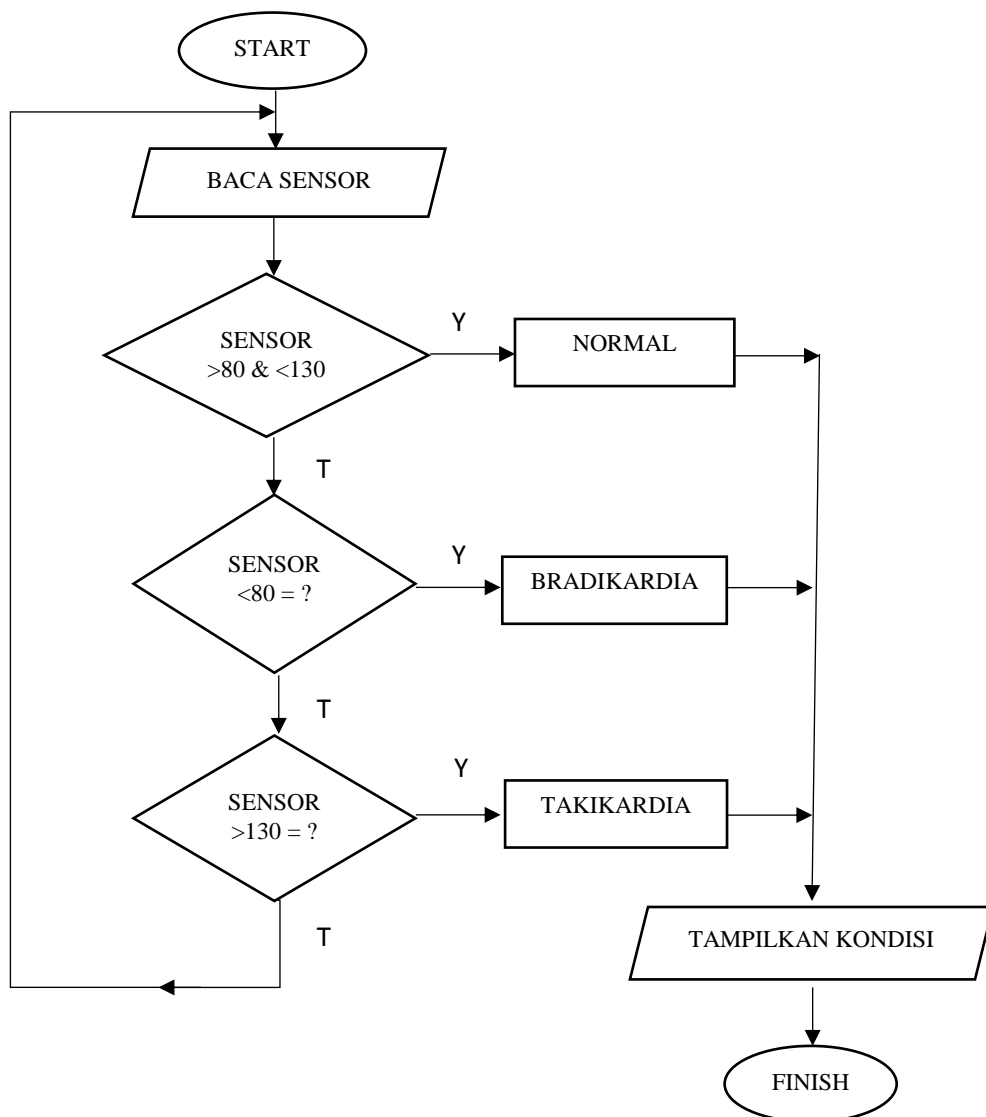
c. Sub Rutin Anak

1. Algoritma Sub Rutin Anak

Deskripsi → Sensor (BPM)

1. Baca sensor
2. Jika nilai sensor > 80 & < 130
3. LCD menampilkan kondisi "Normal"
4. Jika nilai sensor < 80
5. LCD menampilkan kondisi "Bradikardia"
6. Jika nilai sensor > 130
7. LCD menampilkan kondisi "Takikardia"
8. Tulis sensor dan keterangan

2. Flowchart Sub Rutin Anak



Gambar 24. Sub Rutin Kondisi Anak

3. *Pseudocode*

Deklarasi

Keterangan : string
Sensor : integer

Deskripsi

Read (sensor)

If sensor > 80 && < 130 then

Keterangan ← Normal

If sensor < 80 then

Keterangan ← Bradikardia

If sensor > 130 then

Keterangan ← Takikardia

Write (sensor, keterangan)

4. Listing Program

```
void anak() {
  lcd.setCursor(0, 0);
  lcd.print("Denyut:");
  lcd.setCursor(0, 1);
  lcd.print("Status:");
  if (BPM == 0 ) {
    lcd.setCursor(8, 1);
    lcd.print("DEAD !!");
  }
  else if (BPM < 80 ) {
    lcd.setCursor(8, 1);
    lcd.print("BRADIKARDIA  ");
  }
  else if (BPM > 130 ) {
    lcd.setCursor(8, 1);
    lcd.print("TAKIKARDIA ");
  }
  else {
    lcd.setCursor(8, 1);
    lcd.print("NORMAL      ");
  }
}
```

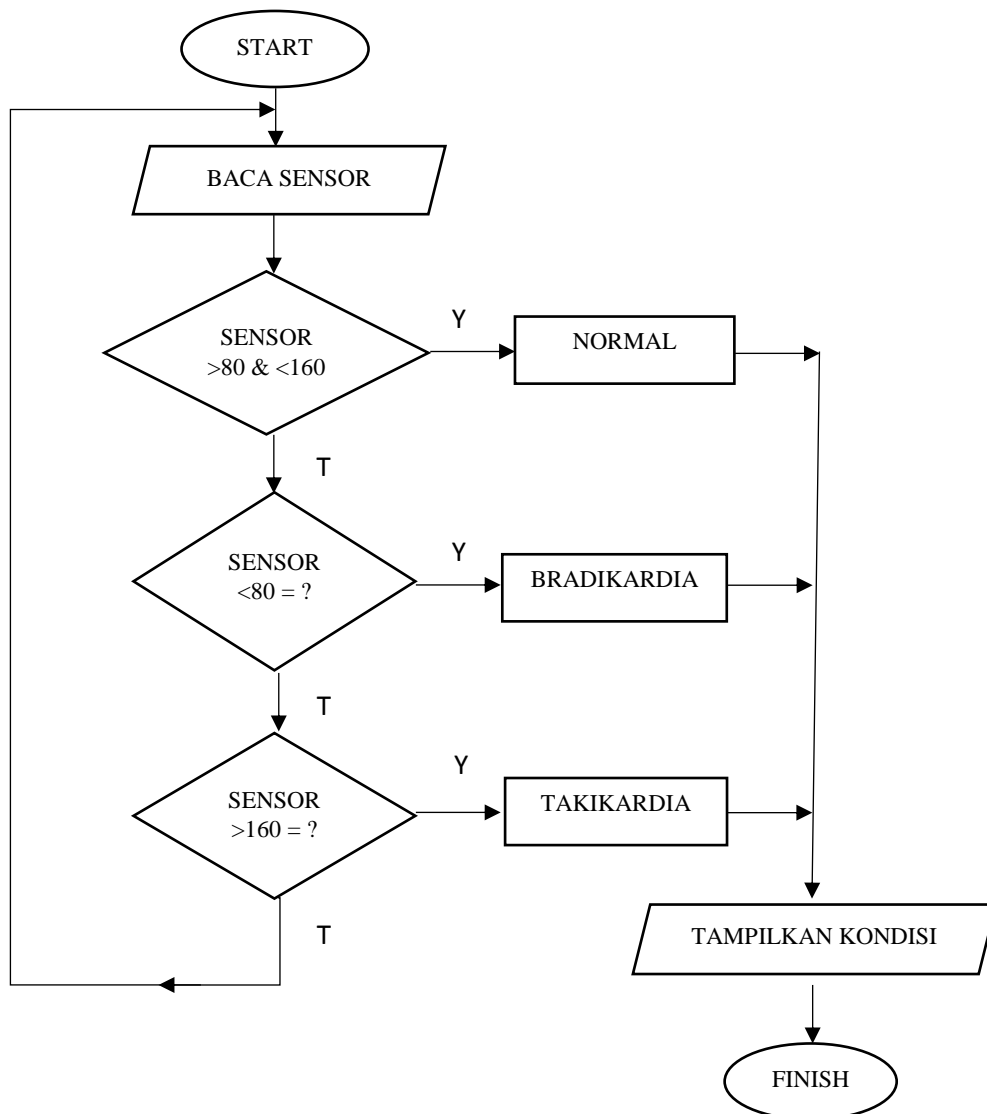
d. Sub Rutin Bayi

1. Algoritma Sub Rutin Bayi

Deskripsi → Sensor (BPM)

1. Baca sensor
2. Jika nilai sensor > 80 & < 160
3. LCD menampilkan kondisi "Normal"
4. Jika nilai sensor < 80
5. LCD menampilkan kondisi "Bradikardia"
6. Jika nilai sensor > 160
7. LCD menampilkan kondisi "Takikardia"
8. Tulis sensor dan keterangan

2. Flowchart Sub Rutin Bayi



Gambar 25. Sub Rutin Kondisi Bayi

3. *Pseudocode*

Deklarasi

Keterangan : string
Sensor : integer

Deskripsi

```

Read (sensor)

If sensor > 80 && < 160 then
    Keterangan ← Normal
If sensor < 80 then
    Keterangan ← Bradikardia
If sensor > 160 then
    Keterangan ← Takikardia

Write (sensor, keterangan )

```

4. Listing Program

```

void bayi() {
    lcd.setCursor(0, 0);
    lcd.print("Denyut:");
    lcd.setCursor(0, 1);
    lcd.print("Status:");
    if (BPM == 0 ) {
        lcd.setCursor(8, 1);
        lcd.print("DEAD !!");
    }
    else if (BPM < 80 ) {
        lcd.setCursor(8, 1);
        lcd.print("BRADIKARDIA  ");
    }
    else if (BPM > 160 ) {
        lcd.setCursor(8, 1);
        lcd.print("TAKIKARDIA ");
    }
    else {
        lcd.setCursor(8, 1);
        lcd.print("NORMAL      ");
    }
}
}

```

B. Pengembangan

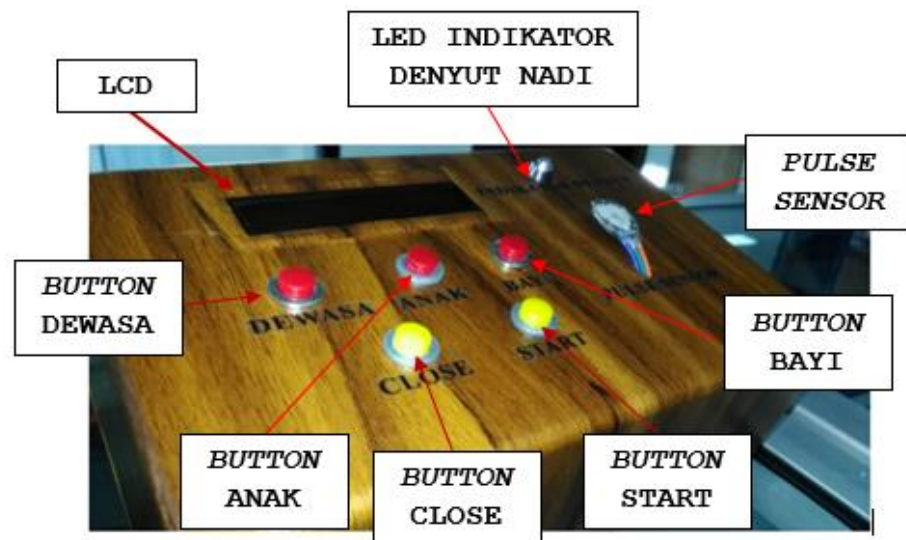
1. Perangkat Keras

a. Perangkat Keras dan Box

Setelah melakukan design box dan merealisasikannya dengan menggunakan *box universal*. Kemudian penulis memasukkan *hardware* ke dalam *box*:



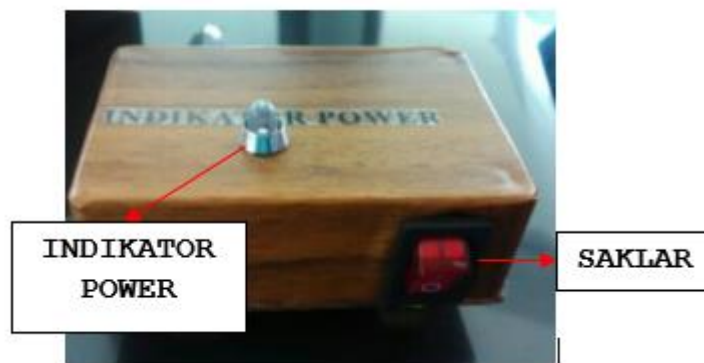
Gambar 26. Penempatan Komponen (depan)



Gambar 27. Penempatan Komponen (atas)



Gambar 28. Penempatan Komponen (belakang)



Gambar 29. Penempatan Komponen Adaptor

C. Implementasi dan Pengujian

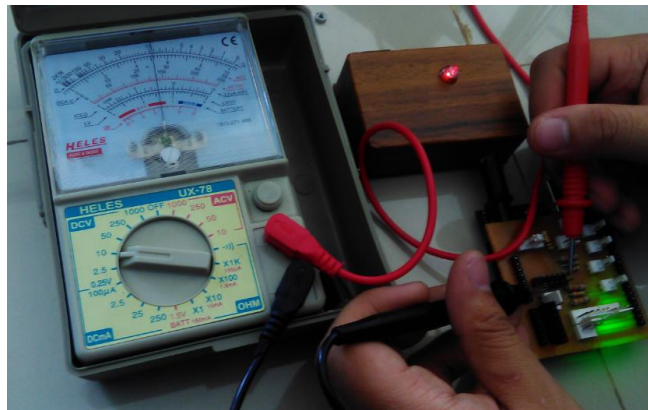
1. Hasil Pengujian Fungsional

Pengujian fungsional diperlukan sebelum melakukan uji unjuk kerja karena pengujian fungsional terdiri dari pengujian dari seluruh komponen yang akan dirakit menjadi satu, komponen tersebut terdiri dari *hardware* dan *software* dan juga untuk memastikan apakah komponen yang digunakan bekerja dengan baik atau tidak.

a. Pengujian Perangkat Keras (*Hardware*)

1. Pengujian Tegangan Baterai

Gambar 30 merupakan Pengukuran tegangan dilakukan dengan multimeter. Pengukuran dilakukan pada bagian *input output* baterai 9 Volt. Hal ini bertujuan untuk mengetahui besarnya tegangan kerja yang masuk sebelum ke sistem minimum Arduino, karena Arduino hanya beroperasi pada tegangan masukan 4,5 – 5,5 Volt. Tabel 17 adalah hasil dari pengukuran tegangan. Pengukuran dilakukan dengan melihat penurunan tegangan *input* dari 9 Volt DC.



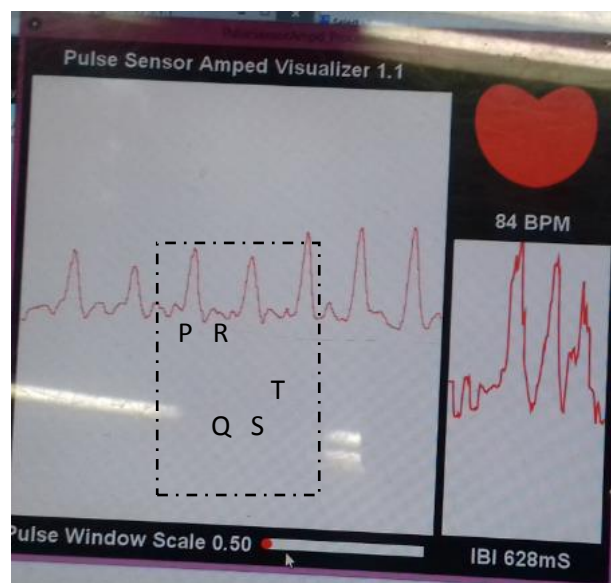
Gambar 30. Pengujian Tegangan Menggunakan Multimeter

Tabel 15. Pengujian Tegangan Baterai 9 Volt

No	Tegangan <i>Input</i>	Tegangan <i>output</i> tanpa beban	Tegangan <i>output</i> dengan beban
1	9,0 Volt	5 Volt	4,6 Volt
2	8,9 Volt	5 Volt	4,6 Volt
3	8,4 Volt	5 Volt	4,6 Volt
4	8,2 Volt	5 Volt	4,6 Volt
5	8,0 Volt	5 Volt	4,6 Volt

Berdasarkan data hasil pengukuran pada tabel 15, bisa dilihat bahwa catu daya dapat bekerja dengan baik. Tegangan masukan terukur sebanyak lima kali mengalami penurunan tegangan *input* dari 9 Volt sampai 8 Volt, disebabkan karena konsumsi rangkaian. Tegangan keluaran tanpa beban dan dengan beban yang terukur tetap stabil meskipun tegangan masukan berubah-ubah.

2. Pengujian *Pulse Sensor*




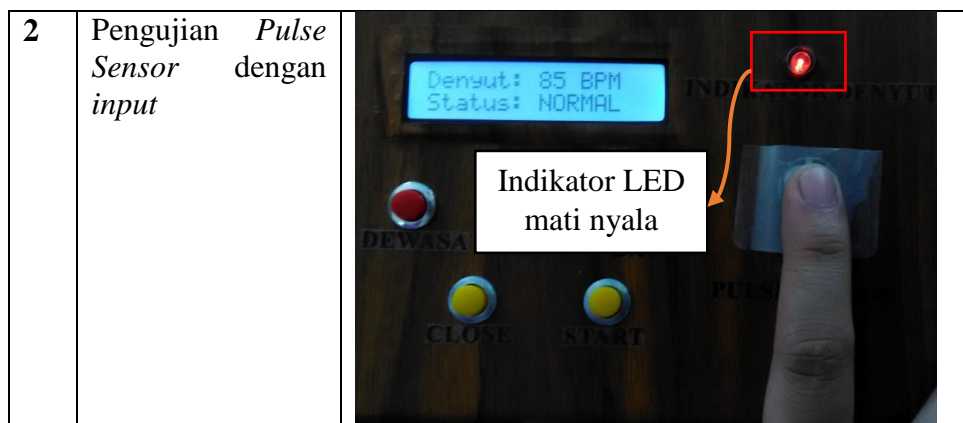
Gambar 31. Tampilan *Visualisasi Processing*

Processing adalah sebuah bahasa program *open-source* dan juga lingkungan pengembangan terintegrasi (IDE) yang dirancang untuk kebutuhan pengembangan elektronik, seni media, desain visual dengan tujuan pengajaran dasar-dasar pemrograman komputer dalam konteks visual dan sebagai dasar sketsa elektronik. Pengujian *pulse sensor* menggunakan *processing* dengan *input* orang dewasa dapat dilihat pada gambar 31. Ketika *Processing* dijalankan pada komputer / laptop terlihat bahwa fungsi dari

program *Processing* hanya mengkonversikan data yang berasal dari Arduino menjadi tampilan grafik pulsa berdasarkan data analog yang diterima oleh *Pulse Sensor*. Dari grafik tersebut diperoleh gelombang PQRS merupakan 1 denyut nadi, jika dalam 1 menit terdapat 84 denyut nadi maka banyaknya gelombang yang terjadi sebanyak 84 kali gelombang PQRS selama 1 menit. Denyut nadi yang terukur sebesar 84 BPM (*beats per minute*) dalam kondisi Normal, karena denyut nadi orang dewasa normal berkisar antara 60 - 100 BPM. Setelah mengetahui bahwa *pulse sensor* dapat bekerja dengan baik kemudian membandingkan data hasil pembacaan *processing* dengan pembacaan alat. Pada tabel 16 No 2 denyut nadi yang terukur adalah 85 BPM dengan kondisi Normal. Dari data tersebut terdapat selisih 1 angka dari pembacaan *processing* dengan alat. IBI (*Interbeat Interval*) umumnya diukur dalam satuan milidetik. Dalam fungsi jantung normal, setiap nilai IBI bervariasi dari detak ke detak. Variasi alami ini dikenal sebagai *heart rate variability* (HRV).

Tabel 16. Pengujian *Pulse Sensor*

No	Aspek	Hasil
1	Pengujian <i>Pulse Sensor</i> tanpa input	



Berdasarkan data hasil pengujian pada tabel 16, bisa dilihat bahwa pengujian *pulse sensor* tanpa *input* atau sebelum jari ditempelkan di sensor, indikator LED belum menyala dan LCD menampilkan “status : DEAD ! “. Hal ini karena sensor belum mendeteksi denyut nadi, sedangkan pengujian menggunakan *input* yaitu dengan menempelkan jari kita ke sensor, indikator LED menyala secara periodik sesuai irama denyut nadi dan LCD menampilkan denyut nadi dalam satuan BPM serta menampilkan hasil keputusan. Dari hasil pengujian *pulse sensor* dapat berfungsi dengan cukup baik.

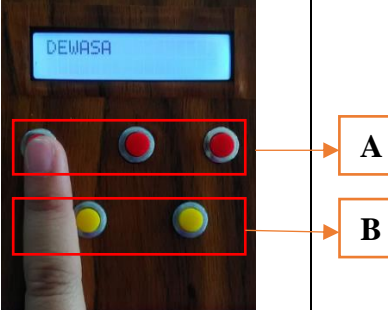



3. Pengujian Fungsi LCD & Push Button


Pengujian LCD dilakukan bertujuan untuk mengetahui apakah LCD sudah bekerja sehingga dapat menampilkan karakter sesuai dengan yang diharapkan. Pengujian LCD dilakukan dengan memprogram karakter atau tulisan yang ingin ditampilkan dan kemudian dicocokkan dengan tampilan yang ada pada layar LCD tersebut. Pengujian *Push button* juga dilakukan untuk mengetahui apakah tombol pemilihan usia dapat bekerja dengan baik atau tidak. Pengujian *Push button* terdiri dari 5 pilihan. Pilihan A terdiri dari

button dewasa, anak, dan bayi. Pilihan B terdiri dari *button start* dan *close*.

Tabel 17 merupakan hasil pengujian dari fungsi LCD dan *Push button*.

Tabel 17. Hasil Pengujian Fungsi LCD dan *Push Button*

No	Kondisi	Gambar Kondisi Alat
1	Ketika <i>button</i> 1 ditekan, maka LCD akan menampilkan “Dewasa”	
2	Ketika <i>button</i> 2 ditekan, maka LCD akan menampilkan “Anak”	
3	Ketika <i>button</i> 3 ditekan, maka LCD akan menampilkan “Bayi”	
4	Ketika <i>button start</i> ditekan, maka LCD akan menampilkan data dari pembacaan <i>pulse sensor</i>	

5	Ketika <i>button close</i> ditekan, maka LCD akan kembali ke menu pertama.	
---	--	--

2. Hasil Pengujian Unjuk Kerja Keseluruhan

Pengujian unjuk kerja secara keseluruhan bertujuan untuk mengetahui Alat Pendeteksi Denyut Nadi secara elektris bekerja dengan baik dan benar. Pengujian ini didampingi oleh ahli medis Dwi Ratna Sari mahasiswa semester 7 Universitas Respati Yogyakarta jurusan Keperawatan. Pengujian terdapat 3 aspek yaitu : pengujian denyut nadi usia dewasa, pengujian denyut nadi usia anak-anak, pengujian denyut nadi usia bayi, dan pengujian terhadap orang yang sedang sakit dan orang setelah berolahraga. Pengujian akurasi pembacaan sensor juga menghitung nilai *error* atau kesalahan. Perhitungan *error* adalah selisih antara pembacaan manual dengan jarak pembacaan sensor. Berikut ini rumus perhitungan nilai error.

$$Error = | \text{pembacaan manual} - \text{pembacaan sensor} |$$

Selanjutnya *error* yang telah diketahui digunakan untuk menghitung persentase kesalahan (% error). Rumus untuk mencari persentase kesalahan (% error) digunakan rumus sebagai berikut:

$$\% \text{ error} = \frac{\text{error}}{\text{pembacaan manual}} \times 100 \%$$

Setelah semua data diperoleh, akan dihitung rata-rata nilai % error secara keseluruhan. Hasil pengujian akurasi *pulse sensor* dapat dilihat pada tabel 18, 19, 20, dan 21.

a. Dewasa

Tabel 18. Pengujian Alat Usia Dewasa

No	Nama	Umur (tahun)	Manual (BPM)	Sensor (BPM)	Status	Error	%Error
1	Wulandari	24	88	86	Normal	2	2,2
2	Ahmad	22	79	80	Normal	1	1,2
3	April	21	86	84	Normal	2	2,3
4	Ridho	20	85	84	Normal	1	1,1
5	Syahrul	21	88	90	Normal	2	2,2
6	Romi	31	91	91	Normal	0	0
7	Junaedi	48	94	94	Normal	0	0
8	Faris	21	69	70	Normal	1	1,4
9	Beni	20	93	92	Normal	1	1,0
10	Ani	44	80	81	Normal	1	1,2

Berikut ini diambil satu sampel perhitungan *error* dan % *error* pada saat mengukur denyut nadi. Contoh perhitungan Tabel 1

Diketahui: manual = 88

Sensor = 86

Ditanya: error dan % error

Penyelesaian:

$$\begin{aligned}
 error &= |\text{pembacaan manual} - \text{pembacaan sensor}| \\
 &= |88 \text{ BPM} - 86 \text{ BPM}| \\
 &= 2 \text{ BPM}
 \end{aligned}$$

$$\begin{aligned}
 \%error &= \frac{error}{\text{Pembacaan manual}} \times 100 \% \\
 &= \frac{2}{88} \times 100 \%
 \end{aligned}$$

$$= 2,2 \%$$

Selanjutnya dicari total nilai *error* dan rata-rata *error*.

$$\text{Total \% error} = 12,6 \%$$

$$\text{Rata-rata \% error} = (\text{Total \% error}) / (\text{Banyak data})$$

$$= 12,6 \% / 10 = 1,26\%$$

Berdasarkan hasil perhitungan *Pulse Sensor* memiliki akurasi pengukuran yang presisi dengan rata-rata kesalahan sebesar 1,26%.

b. Anak-Anak

Tabel 19. Pengujian Alat Usia Anak-Anak

No.	Nama	Usia (tahun)	Manual (BPM)	Sensor (BPM)	Status	<i>Error</i>	<i>%Error</i>
1	Adam	3,5	106	104	Normal	2	1,8
2	Amira	3	110	105	Normal	5	4,5
3	Eldia	3	102	101	Normal	1	0,9
4	Fahri	3,5	101	102	Normal	1	0,9
5	Syifa	3	104	109	Normal	5	4,8
6	Nizam	3	110	116	Normal	6	5,4
7	Melvin	3	103	103	Normal	0	0
8	Laras	3,5	105	101	Normal	4	3,8

Berikut ini diambil satu sampel perhitungan *error* dan *% error* pada saat mengukur denyut nadi. Contoh perhitungan pada tabel 1

Diketahui: manual = 106 BPM

Sensor = 104 BPM

Ditanya: *error* dan *% error*

Penyelesaian:

$$\begin{aligned}
 \text{error} &= |\text{pembacaan manual} - \text{pembacaan sensor}| \\
 &= |106 \text{ BPM} - 104 \text{ BPM}| \\
 &= 2 \text{ BPM}
 \end{aligned}$$

$$\begin{aligned}
 \%error &= \frac{error}{Pembacaan\ manual} \times 100 \% \\
 &= \frac{2}{106} \times 100 \% \\
 &= 1,8 \%
 \end{aligned}$$

Selanjutnya dicari total nilai *error* dan rata-rata *error*.

$$\text{Total \% error} = 22,1 \%$$

$$\text{Rata-rata \% error} = (\text{Total \% error}) / (\text{Banyak data})$$

$$= 22,1 / 8 = 2,76 \%$$

Berdasarkan hasil perhitungan, *Pulse Sensor* memiliki akurasi pengukuran yang presisi dengan rata-rata kesalahan sebesar 2,76 %.

c. Bayi

Tabel 20. Pengujian Alat Usia Bayi

No	Nama	Umur (bulan)	Manual (BPM)	Sensor (BPM)	Keadaan	<i>Error</i>	<i>%Error</i>
1	Elisa	9	119	116	Normal	3	2,5
2	Faiz	3	120	122	Normal	2	1,6
3	Rafaza	9	117	115	Normal	2	1,7
4	Sofia	2	151	150	Normal	1	0,6

Berikut ini diambil satu sampel perhitungan *error* dan *% error* pada saat mengukur denyut nadi.

Diketahui: manual = 119 BPM

Sensor = 116 BPM

Ditanya: *error* dan *% error*

Penyelesaian:

$$\begin{aligned}
 error &= |\text{pembacaan manual} - \text{pembacaan sensor}| \\
 &= |119 \text{ BPM} - 116 \text{ BPM}| \\
 &= 3 \text{ BPM}
 \end{aligned}$$

$$\begin{aligned}
 \%error &= \frac{error}{Pembacaan\ manual} \times 100 \% \\
 &= \frac{3}{119} \times 100 \% \\
 &= 2,5 \%
 \end{aligned}$$

Selanjutnya dicari total nilai *error* dan rata-rata *error*.

$$\text{Total \% error} = 6,4 \%$$

$$\text{Rata-rata \% error} = (\text{Total \% error}) / (\text{Banyak data})$$

$$= 6,4 \% / 4 = 1,6 \%$$

Berdasarkan hasil perhitungan, *Pulse Sensor* memiliki akurasi pengukuran yang presisi dengan rata-rata kesalahan sebesar 1,6 %.

d. Pengujian terhadap orang sakit dan orang setelah berolahraga

Pengujian ini dilakukan di Rumah Sakit Purworejo dan didampingi ahli medis. Pasien atas nama Rahayu Dewi yang sedang opnam karena sakit demam berdarah dengan suhu tubuh 39°C. Sedangkan pengujian orang setelah berolahraga diambil pada saat Ahmad Haris melakukan olahraga Badminton di Pagi Hari. Pengambilan denyut nadi dilakukan 5 menit sesudah berolahraga. Data dari pengujian alat dapat dilihat pada tabel 23.

Tabel 21. Pengujian Alat terhadap orang sakit dan setelah berolahraga

No	Nama	Umur (Thn)	Manual (BPM)	Sensor (BPM)	Keadaan	Error	%Error
1	Rahayu	24	59	59	Bradikardia	0	0
2	Haris	22	118	116	Takikardia	2	1,6

D. Pembahasan

1. Perangkat Keras (*Hardware*)

Berdasarkan tahapan pengujian yang telah dilakukan maka dapat ditarik kesimpulan bahwa sistem yang telah dirancang dapat bekerja sebagaimana mestinya, meskipun terdapat *error* atau kesalahan pada beberapa bagian. Berikut adalah pembahasan dari pengujian yang telah dilakukan.

a. Catu Daya

Hasil pengukuran catu daya bekerja dengan baik, dengan melihat pada tabel 17. Tegangan masukan terukur sebanyak lima kali mengalami penurunan tegangan *input* dari 9 Volt sampai 8 Volt, disebabkan karena konsumsi rangkaian. Tegangan *output* sudah sesuai yang diharapkan dan memenuhi tegangan kerja mikrokontroler arduino uno yaitu sebesar 5 Volt tanpa beban dan 4,5 Volt dengan beban.

b. *Pulse Sensor*

Sensor denyut nadi dapat berfungsi dengan baik dan mempunyai akurasi yang cukup baik. Hasil pengujian pada tabel 20,21,22, dan 23 menunjukkan bahwa *Pulse Sensor* memiliki tingkat pengukuran yang presisi dengan rata-rata kesalahan sebesar 1,87 %. *Error* yang terjadi disebabkan karena Kepekaan *Pulse sensor* dipengaruhi oleh cahaya, oleh karena itu pemasangan *pulse sensor* di jari harus lebih diperhatikan agar tidak terpengaruh oleh cahaya luar.

2. Perangkat Lunak (*Software*)

Setiap bahasa pemrograman mempunyai standar penulisan program.

Dalam proyek akhir ini penulis menjelaskan sebagian konstruksi dari program bahasa Arduino pada tabel 22.

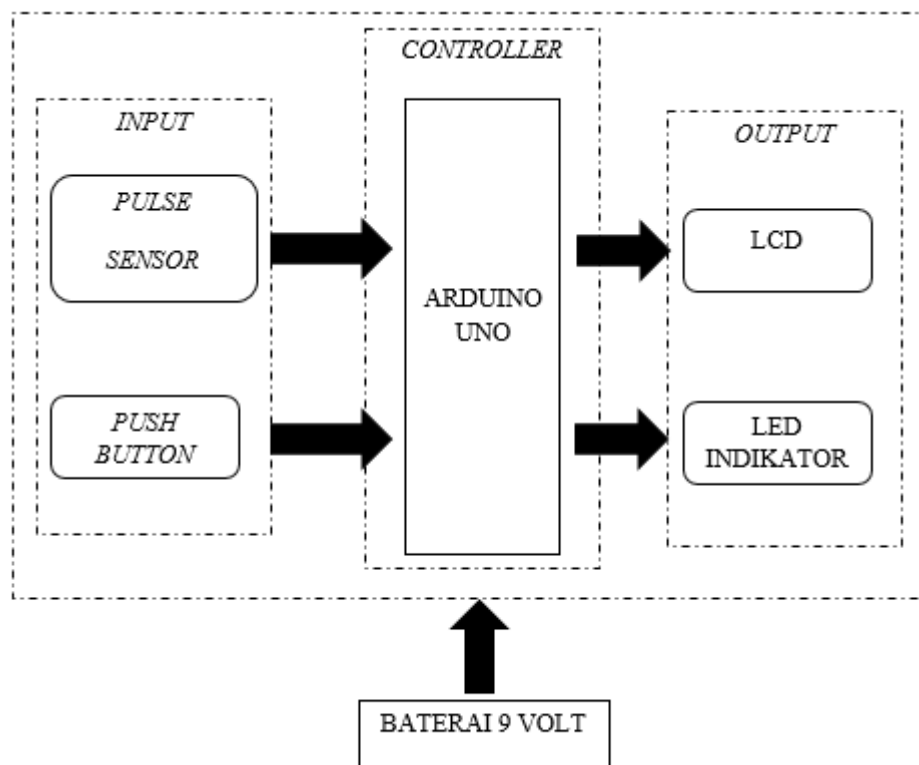
Tabel 22. Standar Penulisan Program

KONTRUKSI	PROGRAM	KETERANGAN
Definisi <i>Hardware</i>	<pre>#include <Wire.h> #include <LiquidCrystal.h> #define saklar1 6 #define saklar2 7 #define saklar3 8 #define mulai 9 #define berhenti 10 #define pulsePin A0 #define blinkPin 13</pre>	Baris disamping menyatakan bahwa <i>hardware</i> yang digunakan berupa LCD (Rs=0, E=1, D4=5, D5=4, D6=3, D7=2), led pada pin 13, pulse sensor pin A0, <i>push button</i> pin 6,7,8,9,10.
Definisi Pemrosesan Awal (<i>Preprocessor</i>)	<pre>pinMode(saklar1, INPUT_PULLUP); pinMode(saklar2, INPUT_PULLUP); pinMode(saklar3, INPUT_PULLUP); pinMode(mulai, INPUT_PULLUP); pinMode(berhenti, INPUT_PULLUP); pinMode(blinkPin, OUTPUT);</pre>	Baris disamping merupakan program pengenalan <i>input</i> dan <i>output</i> dimana inputnya adalah 5 <i>button</i> dan <i>output</i> adalah LED.
Definisi Variable		<i>Variabel</i> adalah suatu pengenalan (<i>identifier</i>) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program yang dapat diubah-ubah

		sesuai dengan kebutuhan. Nama dari <i>variable</i> terserah sesuai dengan yang diinginkan
Deklarasi Fungsi	<pre> interruptSetup(); dewasa(); sensor(); anak(); bayi(); setup(); </pre>	Fungsi merupakan bagian yang terpisah dari program dan dapat dipanggil di manapun di dalam program.
Deklarasi Variable	<pre> int kondisi = 0; int enable = 0; </pre>	
Operator	<pre> (BPM == 0) (BPM < 80) (BPM > 160) </pre>	Operator hubungan digunakan untuk membandingkan hubungan dua buah operand atau sebuah nilai / variable
Pernyataan Kondisional (IF-THEN-END IF)	<pre> if (BPM == 0) { lcd.setCursor(8, 1); lcd.print("DEAD !!"); } else if (BPM < 80) { lcd.setCursor(8, 1); lcd.print("BAKIKARDIA "); } else if (BPM > 160) { lcd.setCursor(8, 1); </pre>	<p>Pernyataan ini digunakan untuk melakukan pengambilan keputusan terhadap dua buah bahkan lebih kemungkinan untuk melakukan suatu blok pernyataan atau tidak.</p> <p><i>IF</i> pernyataan kondisi 1 <i>THEN</i></p>

	<pre> lcd.print("TAKIKARDIA"); } else { lcd.setCursor(8, 1); lcd.print("NORMAL"); </pre>	'blok pernyataan 1 yang dikerjakan bila kondisi 1 terpenuhi
--	---	---

E. Cara Kerja Alat Pendeteksi Denyut Nadi



Gambar 32. Diagram Blok Cara Kerja Alat

Alat pendeteksi denyut nadi ini menggunakan *pulse sensor*. Sensor denyut mendeteksi jumlah denyut nadi permenit (BPM). *Push button* digunakan sebagai tombol pemilihan usia. Keluaran dari sensor tersebut akan

dikirim dan diolah oleh mikrokontroler arduino uno. Didalam mikrokontroller data akan diolah dan diproses kemudian hasilnya akan ditampilkan dalam LCD. LED digunakan sebagai indikator ada tidaknya denyut nadi. LED akan menyala secara stabil ketika denyut nadi sudah ditemukan. *Power supply* berfungsi untuk memberi tegangan ke seluruh rangkaian.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan pengujian yang telah dilakukan terhadap "Perancangan Dan Implementasi Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan Pulse Sensor Berbasis Arduino Uno" maka dapat disimpulkan:

1. Perangkat Keras "Perancangan Dan Implementasi Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan Pulse Sensor Berbasis Arduino Uno" berhasil dibuat dengan Arduino yang didukung oleh perangkat lunak didalamnya dan digabung dengan beberapa rangkaian yang saling mendukung. Rangkaian pendukung alat ini adalah adaptor sebagai catu daya, mikrokontroler Arduino, *pulse sensor*, *push button*, LED, dan LCD.
2. Perangkat lunak yang digunakan untuk merealisasikan alat ini adalah *software* Arduino untuk memprogram sistem minimum. Proteus7 digunakan untuk membuat layout PCB Arduino. Secara keseluruhan program yang dibuat sudah dapat bekerja sesuai dengan tujuan. Hal ini ditunjukkan dengan berhasilnya Arduino melakukan pembacaan dan pengolahan data dari sensor.
3. Alat pendeteksi denyut nadi manusia yaang dirancang dapat digunakan untuk mengukur denyut nadi berdasarkan usia serta dapat mengetahui kondisi denyut nadi seseorang secara langsung. *Pulse sensor* memiliki tingkat pengukuran yang presisi dengan rata-rata kesalahan sebesar 1,87%

B. Keterbatasan Alat

”Perancangan Dan Implementasi Alat Pendeteksi Denyut Nadi Berdasarkan Usia Menggunakan Pulse Sensor Berbasis Arduino Uno” memiliki keterbatasan yaitu pada saat pengambilan data harus memperhatikan peletakan jari di *pulse sensor* karena kepekaan *pulse sensor* dipengaruhi oleh cahaya. *Photodiode* akan mendeteksi intensitas cahaya yang dipancarkan oleh LED inframerah berdasarkan volume darah yang mengalir pada pembuluh darah kapiler.

C. Saran

Berdasarkan hasil penelitian, penulis mengakui terdapat kekurangan terhadap alat yang dibuat ini, karena keterbatasan kemampuan dan waktu, maka penulis mengajukan saran sebagai berikut:

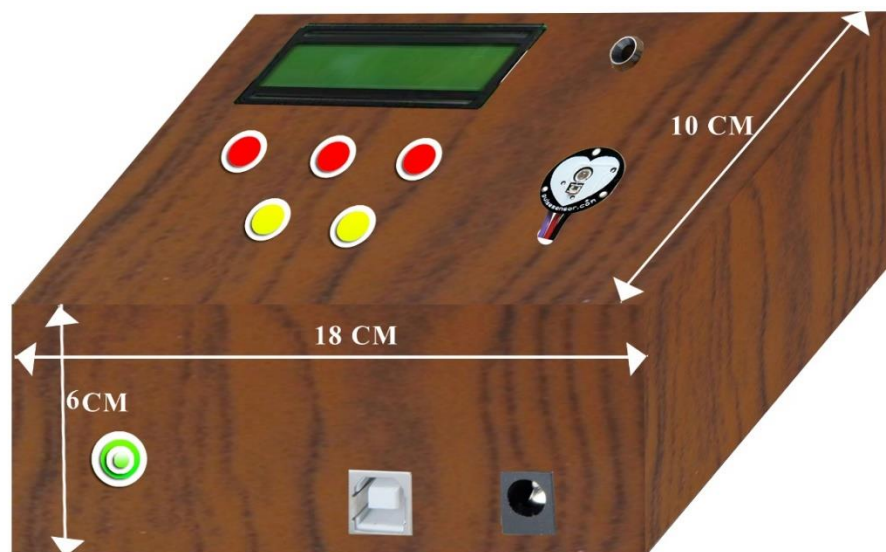
1. alat pendeteksi denyut nadi ini akan lebih baik jika ada penambahan komponen berupa *output* bunyi jantung yang nantinya akan dianalisis bunyi jantung tersebut menggunakan komputer untuk mendeteksi lebih lanjut adanya kelainan atau penyakit jantung.
2. alat yang dibuat diharapkan dapat dikembangkan lebih lanjut oleh mahasiswa Universitas Negeri Yogyakarta untuk bahan penelitian lebih lanjut.

DAFTAR PUSTAKA

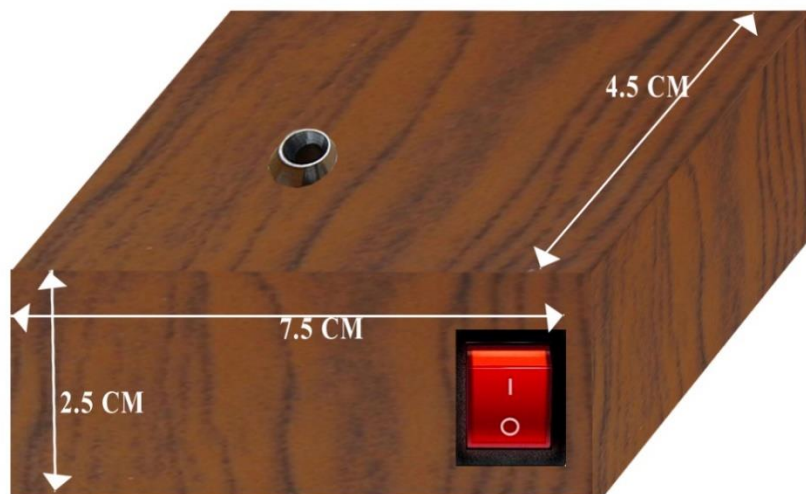
- Agus, Purnomo (2012). *Limit Switch Dan Saklar Push ON*. Diakses pada tanggal 11 Mei 2016, dari <http://elektronika-dasar.web.id/limit-switch-dan-saklar-push-on/>.
- Agus Purnomo (2012). *LCD (Liquid Cristal Display)*. Diakses pada tanggal 25 Mei 2016, dari <http://elektronika-dasar.web.id/lcd-liquid-cristal-display/>
- Agus Purnomo (2012). *LED (Light Emitting Dioda)*. Diakses pada tanggal 27 Mei 2016, dari <http://elektronika-dasar.web.id/led-light-emitting-dioda/>
- Agus Tinuskalangi. 2013. Analisa Pemanfaatan Mikrokontroler Atmega8535 Pada Pemeriksaan Denyut Nadi Manusia. Gema Teknologi Vol. 17 No. 2.
- Anita, D, dan Slamet, S. 2015. Perancangan Alat Pemantau Kondisi Kesehatan Manusia. Vol. 4 No 2.
- Aris Munandar (2012). *Prinsip Kerja LCD (Liquid Cristal Display)*. Diakses pada tanggal 11 Mei 2016, dari <http://www.leselektronika.com/2012/06/liquid-crystal-display-lcd-16-x-2.html>
- Febriadi, Santoso (2013). *Panduan Mudah Simulasi dan Praktek Mikrokontroler Arduino*. Diambil pada tanggal 23 Mei 2016, <http://febriadisantosa.weebly.com/knowledge/arduino-uno-5/14/2013>
- Gitman, Yuri. (2013). *Pulse Sensor (Sensor Detak Jantung)*. Diakses pada tanggal 9 Mei 2016, dari <http://www.pulsesensor.com/>
- Kozier,Barbara.2010. *Fundamental Keperawatan: Konsep, Proses, dan Praktik*. Jakarta : EGC
- Nurhayati (2013). Penyakit Kardiovaskular Masih Menduduki Peringkat Teratas Penyebab Kematian Terbanyak Di Indonesia. Diakses dari <http://www.koran-jakarta.com/awas-penyakit-kardiovaskuler-tak-kenal-kompromi/>
- Wibi, W dan I Komang, S. 2011. Perancangan dan Pembuatan Alat Penghitung Detak Jantung Dengan *Bipolar Standart Lead* Berbasis Mikrkontroler ATMEGA 8535. Eltek Vol. 2 No. 2

LAMPIRAN

Lampiran 2. Desain Box

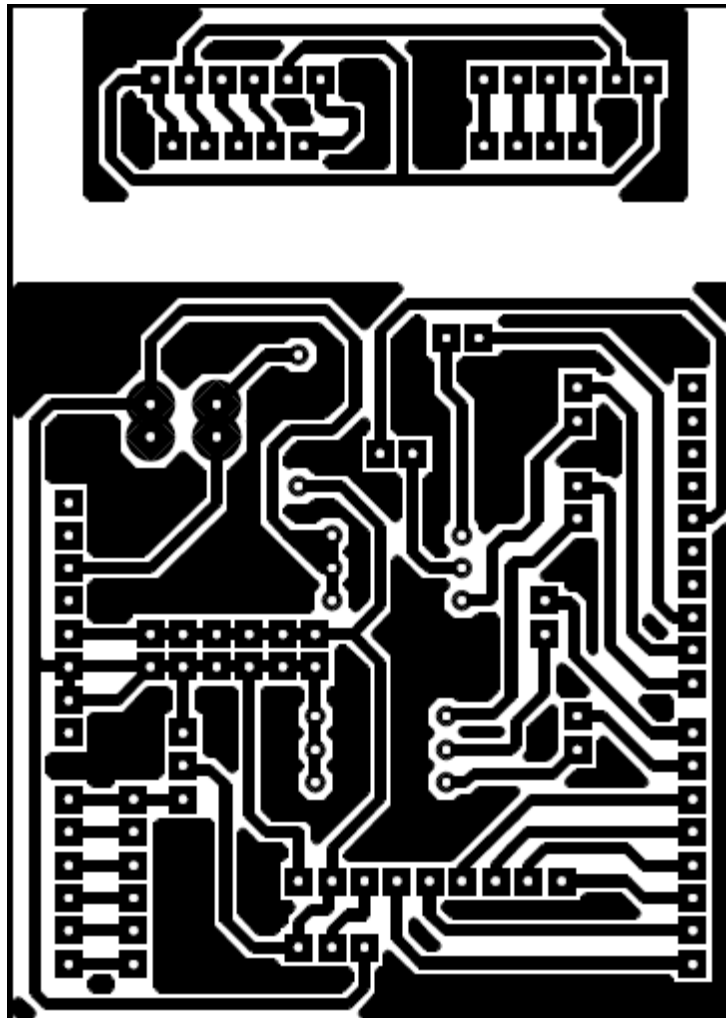


Desain Box Alat Pendeteksi Denyut Nadi

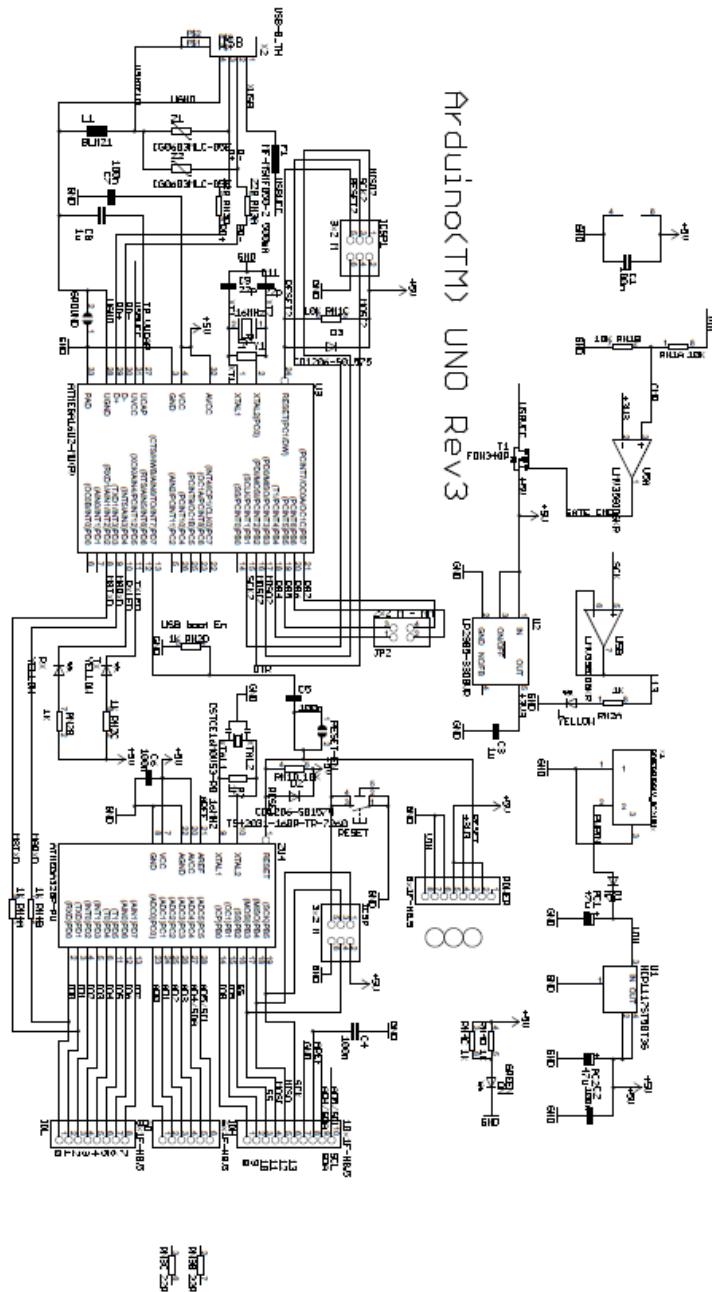


Desain Box Adaptor

Lampiran 3. Desain PCB



Lampiran 4. Schematic Arduino Uno R3



Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS," WITHOUT WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. REFERENCE DESIGNS ARE PROVIDED BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information. ARDUINO is a registered trademark.

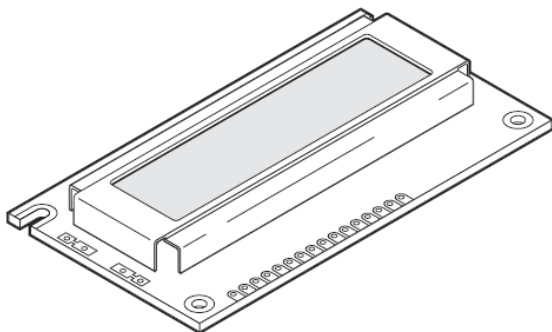
Use of the ARDUINO name must be compliant with <http://www.arduino.cc/en/Main/Policy>

Lampiran 5. Datasheet LCD 16x2

ALPHANUMERIC LCD DISPLAY (16 x 2)

Order Code

LED008	16 x 2 Alphanumeric Display
FRM010	Serial LCD Firmware (optional)



Contents

- 1 x 16x2 Alphanumeric Display
- 1 x data booklet

Introduction

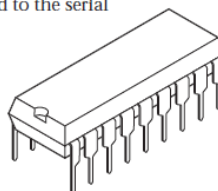
Alphanumeric displays are used in a wide range of applications, including palmtop computers, word processors, photocopiers, point of sale terminals, medical instruments, cellular phones, etc. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. A full list of the characters and symbols is printed on pages 7/8 (note these symbols can vary between brand of LCD used). This booklet provides all the technical specifications for connecting the unit, which requires a single power supply (+5V).

Further Information

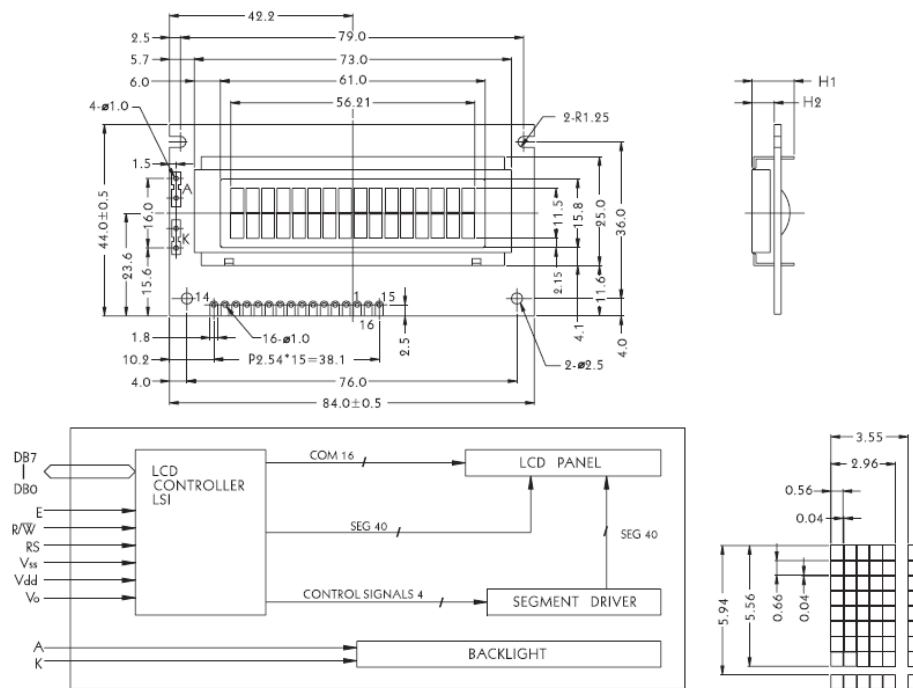
Available as an optional extra is the Serial LCD Firmware, which allows serial control of the display. This option provides much easier connection and use of the LCD module. The firmware enables microcontrollers (and microcontroller based systems such as the PICAXE) to visually output user instructions or readings onto an LCD module. All LCD commands are transmitted serially via a single microcontroller pin. The firmware can also be connected to the serial port of a computer.

An example PICAXE instruction to print the text 'Hello' using the `serout` command is as follows:

```
serout 7,T2400,("Hello")
```



Outline Dimension and Block Diagram



The tolerance unless classified $\pm 0.3\text{mm}$

MECHANICAL SPECIFICATION

Overall Size	84.0 * 44.0	Module	H2 / H1
View Area	61.0 * 15.8	W/O B/L	5.1 / 9.7
Dot Size	0.56 * 0.66	EL B/L	5.1 / 9.7
Dot Pitch	0.60 * 0.70	LED B/L	9.4 / 14.0

PIN ASSIGNMENT

Pin no.	Symbol	Function
1	Vss	Power supply (GND)
2	Vdd	Power supply (+5V)
3	Vo	Contrast Adjust
4	RS	Register select signal
5	R/W	Data read / write
6	E	Enable signal
7	DB0	Data bus line
8	DB1	Data bus line
9	DB2	Data bus line
10	DB3	Data bus line
11	DB4	Data bus line
12	DB5	Data bus line
13	DB6	Data bus line
14	DB7	Data bus line
15	A	Power supply for LED B/L (+)
16	K	Power supply for LED B/L (-)

ABSOLUTE MAXIMUM RATING

Item	Symbol	Conditions	Min.	Max.	Unit
Power Supply Voltage	Vdd-Vss	—	0	7	V
LCD Driving Supply Voltage	Vdd-Vee	—	0	13	V
Input Voltage	Vin	—	-0.3	Vdd+0.3	V
Operating Temperature	Topr	Nor.	0	50	°C
Storage Temperature	Tstg	Nor.	-20	+70	°C

ELECTRICAL CHARACTERISTICS (Vdd = +5V, Ta = 25°C)

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Logic Supply Voltage	Vdd	—	4.5	5	5.5	V
"H" Input Voltage	Vih	—	2.2	—	—	V
"L" Input Voltage	Vil	—	—	—	0.6	V
"H" Output Voltage	VoH	—	2.4	—	—	V
"L" Output Voltage	VoL	—	—	—	0.4	V
Supply Current	Idd	—	2	—	—	mA
LCD Driving Voltage	VLCD	Vdd-Vo	4.3	—	4.8	V

Electrical Characteristics

$V_{dd} = 5V \pm 5\%$
 $V_{ss} = 0V$

Item	Symbol	Condition	Standard value			Unit	Applicable terminal
			Min.	Typ.	Max.		
Power voltage	V_{dd}		4.5	5.00	5.5	V	V_{dd}
Input H - level voltage	V_{IH}		2.2	—	V_{dd}	V	RS, R/ \overline{W} , E DB0~DB7
Input L - level voltage	V_{IL}		-0.3	—	0.6	V	
Output H - level voltage	V_{OH}	$-I_{OH} = 0.205\text{mA}$	2.4	—	—	V	DB0~DB7
Output L - level voltage	V_{OL}	$I_{OL} = 1.2\text{mA}$	—	—	0.4	V	
I/O leakage current	I_{IL}	$V_{in} = 0 \sim V_{dd}$	-1	—	1.0	μA	RS, R/ \overline{W} , E DB0~DB7
Supply current	I_{dd}	$V_{dd} = 5V$	2	—	—	mA	V_{dd}
LCD operating voltage	V_{LCD}	$V_{dd} - V_0$	3.0	—	11.0	V	V_0

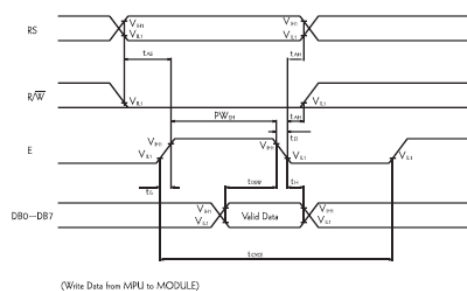
Timing Characteristics

$V_{dd} = 5V \pm 5\%$
 $V_{ss} = 0V$

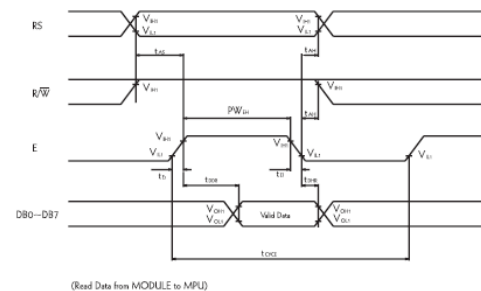
Item	Symbol	Min.	Max.	Unit
Enable cycle time	T_{CYCE}	500	—	ns
Enable pulse width	"High" level P_{WEH}	220	—	ns
Enable rise/fall time	T_{ER}, T_{EF}	—	25	ns
Set-up time	RS, R/ \overline{W} , E T_{AS}	40	—	ns
Address hold time	T_{AH}	10	—	ns
Data set-up time	T_{DSH}	60	—	ns
Data delay time	T_{DDR}	60	120	ns
Data hold time (writing)	T_H	10	—	ns
Data hold time (reading)	T_{DHR}	20	—	ns
Clock oscillating frequency	T_{OSC}	270(TYP)		KHz

Timing Chart

◆ FIG.1 WRITE OPERATION

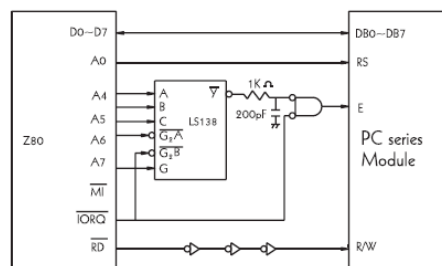
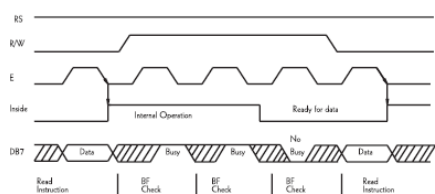


◆ FIG.2 READ OPERATION



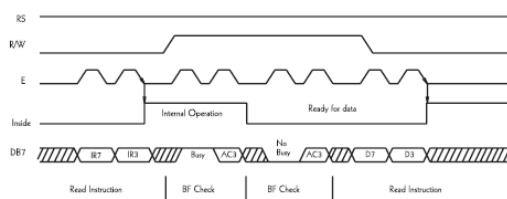
Interface with MPU

◆ Example of Interface with 8-bit MPU (Z80)

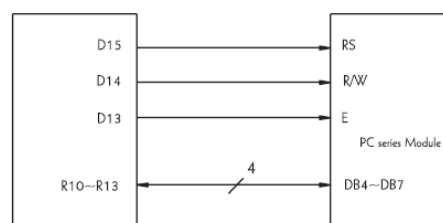


◆ Example of interface with 4-bit MPU

Interface with 4-bit MPU can be made through I/O port of 4-bit MPU. If there are enough I/O ports, data can be transferred by 8-bit, however, if there are not data transfer can be done by 4-bit in twice (select interface is 4-bit long), and timing sequence will be complicated in this case. Please take into account that 2 cycles of BF check is necessary, while 2 cycles of data transfer are also necessary.



Note: IR7, IR3: 7th bit, 3rd bit of instruction
AC3: 3th bit of Address Counter



Features

- (1) Interface with 8-bit or 4-bit MPU is available.
- (2) 192 kind of alphabets, numerals, symbols and special characters can be displayed by built-in character generator (ROM).
- (3) Other preferred characters can be displayed by character generator (RAM).
- (4) Various functions of instruction are available by programming.
 - Clear display • Cursor at home • On / off cursor
 - Blink character • Shift display • Shift cursor
 - Read / write display data.....etc.
- (5) Compact and light weight design which can be easily assembled in devices.
- (6) Single power supply +5V drive (except for extended temp. type).
- (7) Low power consumption.
 - *Interface between data bus line and 4-bit or 8-bit MPU is available.
 - Data transfer are made in twice in case of 4-bit MPU, and once in case of 8-bit MPU.

◆ If interface data is 4-bit long

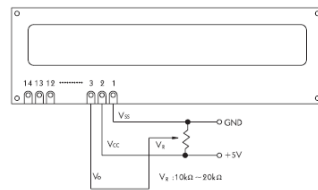
Data transfer are made through 4 bus lines from DB4 to DB7. (while the rest of 4 bus lines from DB0 to DB3 are not used.) Data transfer with MPU are completed when 4-bit data are transferred in twice. (first upper 4-bit data, then lower 4-bit data.)

◆ If interface data is 8-bit long

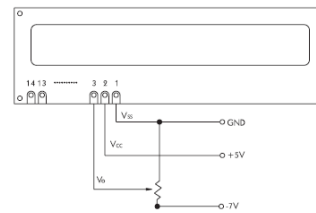
Data transfer are made through all of 8 bus lines from DB0 to DB7.

Example of Power Supply

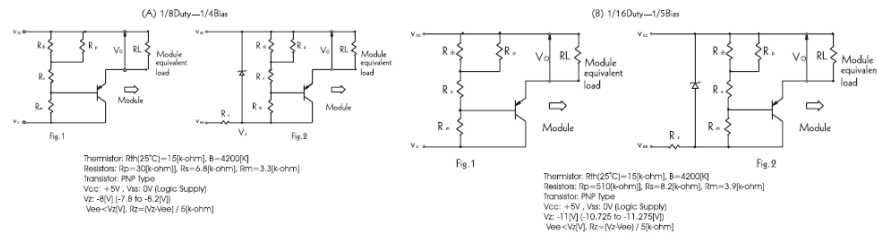
◆ Normal Temperature Type



◆ Extended Temperature Type



◆ Examples of Temperature Compensation Circuits for Extended Temp Type. (Only for reference)



Instructions

Instruction	Code										Description	Executed Time(max.)
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears all display and returns the cursor to the home position (Address 0)	1.64mS
Cursor At Home	0	0	0	0	0	0	0	0	1	*	Returns the cursor to the home position (Address 0). Also returns the display being shifted to the original position. DD RAM contents remain unchanged.	1.64mS
Entry Mode Set	0	0	0	0	0	0	0	1	1/D	S	Sets the cursor move direction and specifies or not to shift the display. These operations are performed during data write and read.	40μS
Display On / Off Control	0	0	0	0	0	0	1	D	C	B	Sets ON / OFF of all display (D), cursor NO / OFF (C), and blink of cursor position character (B).	40μS
Cursor / Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves the cursor and shifts the display without changing DD RAM contents.	40μS
Function Set	0	0	0	0	1	DL	N	F	*	*	Sets interface data length (DL) number of display lines (L) and character font (F)	40μS
CG RAM Address Set	0	0	0	1	ACG						Sets the CG RAM address. CG RAM data is sent and received after this setting.	40μS
DD RAM Address Set	0	0	1	ADD						Sets the DD RAM address. DD RAM data is sent and received after this setting.	40μS	
Busy Flag / Address Read	0	1	BF	AC						Reads Busy flag (BF) indicating internal operation is being performed and reads address counter counts.	0μS	
CG RAM / DD RAM Data Write	1	0	WRITE DATA						Writes data into DD RAM or CG RAM.	40μS		
CG RAM / DD RAM Data Read	1	1	READ DATA						Reads data from DD RAM or CG RAM.	40μS		
Code											Description	Executed Time (max)
I / D = 1: Increment I / D = 0: Decrement S = 1: With display shift S / C = 0: cursor movement R / L = 1: Shift to the right R / L = 0: Shift to the left DL = 1: 8-bit				DL = 0: 4-bit N = 1: 2lines N = 0: 1line F = 1: 5×10dots F = 0: 5×7dots BF = 1: Internal operation is being performed BF = 0: Instruction acceptable							DD RAM: Display Data RAM CG RAM: Character Generator RAM ACG: CG RAM Address ADD: DD RAM Address Corresponds to cursor address. AC: Address Counter, used for both DD RAM and CG RAM *: Invalid	fcp or fosc = 250KHz However, when frequency changes, execution time also changes Example if fcp or fosc is 270KHz, 70μS × 250 / 270 = 37μS

Power Supply Reset

The internal reset circuit will be operated properly when the following power supply conditions are satisfied. If it is not operated properly, please perform initial setting along with the instruction.

Item	Symbol	Measuring Condition	Standard Value			Unit
			Min.	Typ.	Max.	
Power Supply RISE Time	tree	—	0.1	—	10	mS
Power Supply CFF Time	toff	—	1	—	—	mS

Reset function

◆ Initialization Made by Internal Reset Circuit

HD44780 automatically initializes (resets) when power is supplied (builtin internal reset circuit). The following instructions are executed in initialization. The busy flag (BF) is kept in busy state until initialization ends. (BF=1) The busy state is 10 ms after Vdd reaches to 4.5V.

- (1) Display clear
- (2) Function set

DL= 1:8 bit long interface data

DL= 0:4 bit F= 0:5 x 7dots character font

N= 1:2 lines

N= 0:1 line
- (3) Display ON / OFF control

D= 0:Display OFF C= 0:Cursor OFF

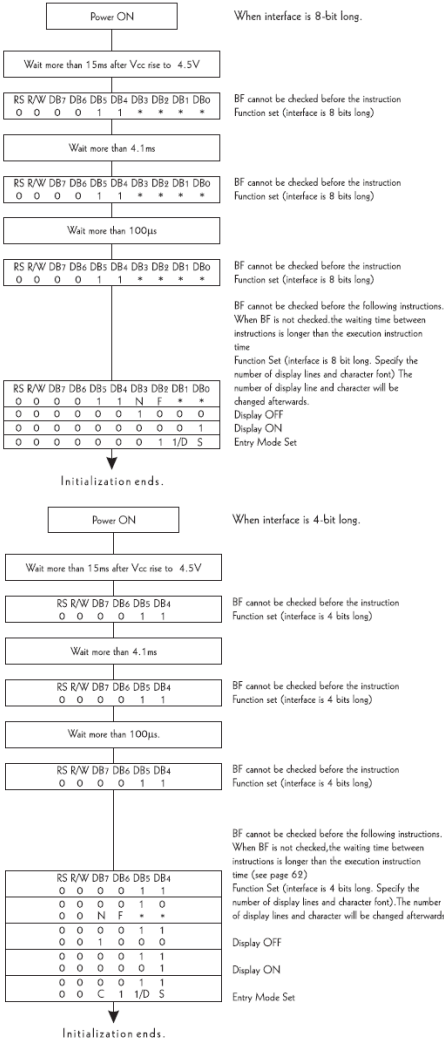
B= 0:Blink OFF
- (4) Entry mode set

1 / D= 1:+1(increment) S= 0:No shift

Note:When conditions stated in power supply conditions using internal reset circuit are not satisfied.The internal reset circuit will not operate properly and initialization will not be performed. Please make initialization using MPU along with instruction.

◆ Initialization along with instruction

If power supply conditions are not satisfied, which for proper operation of internal rest circuit, it is required to make initialization along with instruction. Please make following procedures.



Standard Character Pattern (Powertip Module)

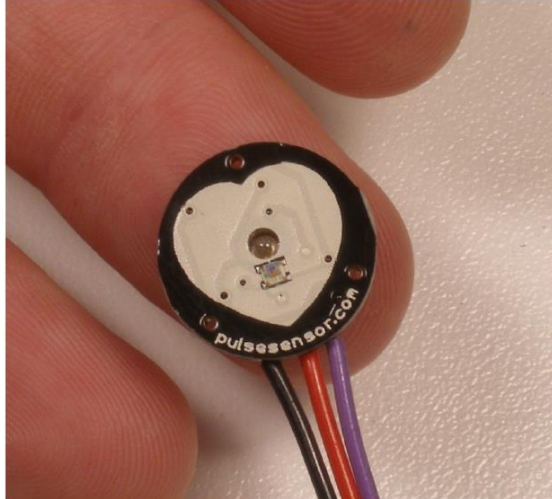
		Higher 4-bit (D4 to Character Code (Hexadecimal))															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	0	CG RAM (1)	±		0	0	P	°	P	5	é	á			í	Á	ß
	1	CG RAM (2)	≡	!	1	A	Q	a	a	Q	²	í			J	+	Y
	2	CG RAM (3)	"		2	B	R	b	r	é	°	°			ó	3	8
	3	CG RAM (4)	#		3	C	S	c	s	á	ó	ó			P	¶	e
	4	CG RAM (5)	\$		4	D	T	d	t	á	á	á			'	4	7
	5	CG RAM (6)	%		5	E	U	e	u	á	á	á			↑	∠	¶
	6	CG RAM (7)	&		6	F	U	f	v	á	Q	°			↓	0	0
	7	CG RAM (8)	'		7	G	W	g	w	°	U	R			→	À	L
	8	CG RAM (1)	(8	H	X	h	x	é	U	f			÷	÷	3
	9	CG RAM (2))		9	I	Y	i	y	é	U	i			∠	¶	À
	A	CG RAM (3)	*		*	J	Z	j	z	é	0	á			2	7	2
	B	CG RAM (4)	+		+	K	[k		°	i	R			*L	¶	¶
	C	CG RAM (5)	,		,	<	L	\	1	1	i	R			U	0	0
	D	CG RAM (6)	-		-	=	M]	m	°	i	3			°	¶	¶
	E	CG RAM (7)	.		.	>	N	^	n	°	°	0			¶	0	0
	F	CG RAM (8)	/		/	?	O	_	o	á	á	á			°	0	0

Standard Character Pattern (Elec & Eltek Module)

Upper(4bit) Lower(4bit)		LLLL	LLHL	LLHH	LHLL	LHLH	LHHH	HLLL	HLLH	HLHL	HLHH	HHLL	HHLH	HHHL	HHHH
LLLL	CG RAM (1)														
LLHH	(2)														
LLHL	(3)														
LLHH	(4)														
LHLL	(5)														
LHLH	(6)														
LHHL	(7)														
LHHH	(8)														
HLLL	(1)														
HLLH	(2)														
HLHL	(3)														
HLHH	(4)														
HHLL	(5)														
HHLH	(6)														
HHHL	(7)														
HHHH	(8)														

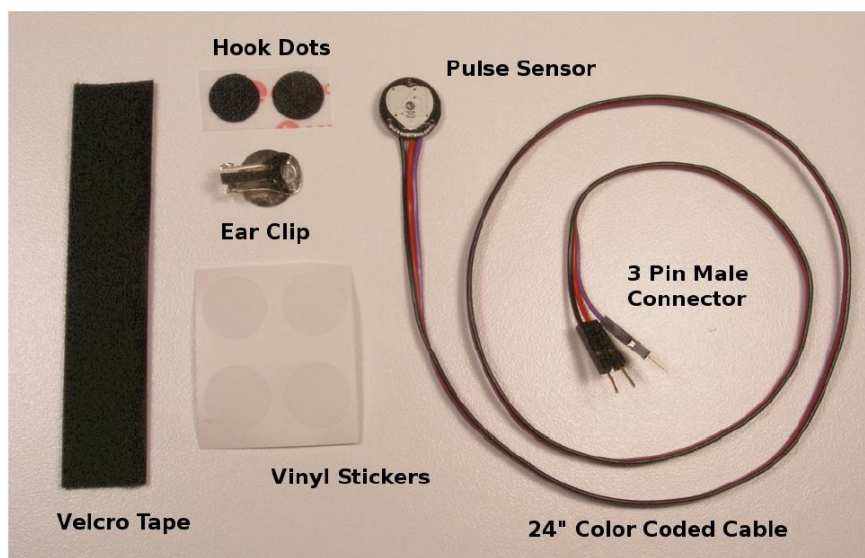
Lampiran 6. Datasheet Pulse Sensor

Pulse Sensor Getting Started Guide



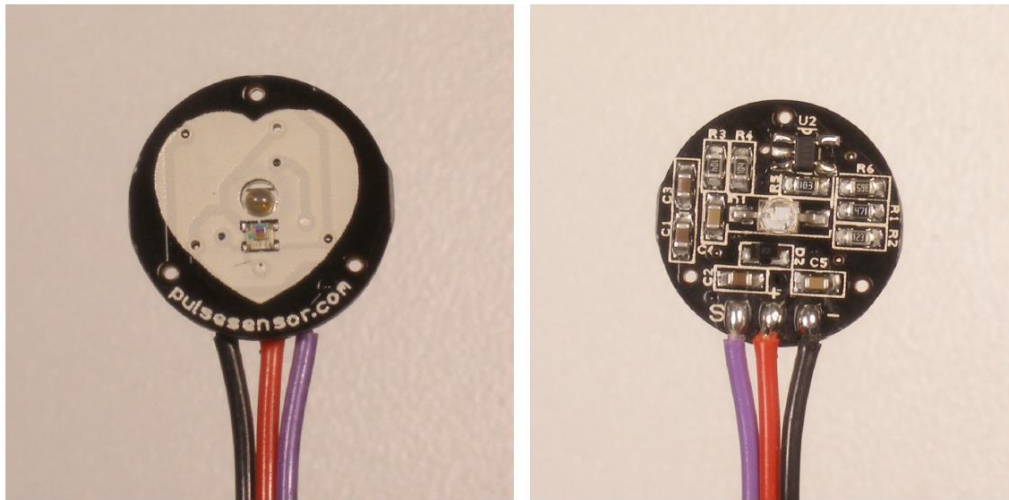
Introduction:

Pulse Sensor is a well-designed plug-and-play heart-rate sensor for Arduino. It can be used by students, artists, athletes, makers, and game & mobile developers who want to easily incorporate live heart-rate data into their projects. The sensor clips onto a fingertip or earlobe and plugs right into Arduino with some jumper cables. It also includes an open-source monitoring app that graphs your pulse in real time.



The Pulse Sensor Kit includes:

- 1) A 24-inch Color-Coded Cable, with (male) header connectors. You'll find this makes it easy to embed the sensor into your project, and connect to an Arduino. No soldering is required.
- 2) An Ear Clip, perfectly sized to the sensor. We searched many places to find just the right clip. It can be hot-glued to the back of the sensor and easily worn on the earlobe.
- 3) 2 Velcro Dots. These are 'hook' side and are also perfectly sized to the sensor. You'll find these velcro dots very useful if you want to make a velcro (or fabric) strap to wrap around a finger tip.
- 4) Velcro strap to wrap the Pulse Sensor around your finger.
- 4) 3 Transparent Stickers. These are used on the front of the Pulse Sensor to protect it from oily fingers and sweaty earlobes.
- 5) The Pulse Sensor has 3 holes around the outside edge which make it easy to sew it into almost anything.

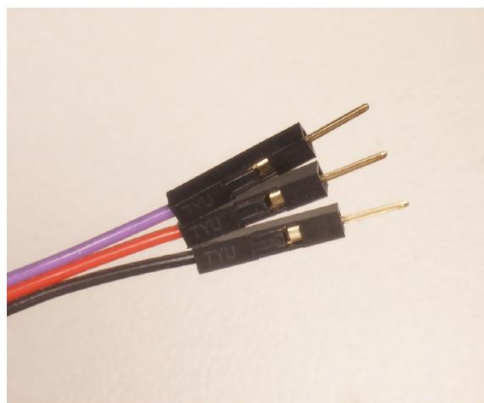


Let's get started with Pulse Sensor Anatomy

The front of the sensor is the pretty side with the Heart logo. This is the side that makes contact with the skin. On the front you see a small round hole, which is where the LED shines through from the back, and there is also a little square just under the LED. The square is an ambient light sensor, exactly like the one used in cellphones, tablets, and laptops, to adjust the screen brightness in different light conditions. The LED shines light into the fingertip or earlobe, or other capillary tissue, and sensor reads the light that bounces back. The back of the sensor is where the rest of the parts are mounted. We put them there so they would not get in the way of the of the sensor on the front. Even the LED we are using is a reverse mount LED. For more about the circuit functionality, check out the Hardware page.[\[needs link\]](#)

The cable is a 24" flat color coded ribbon cable with 3 male header connectors.

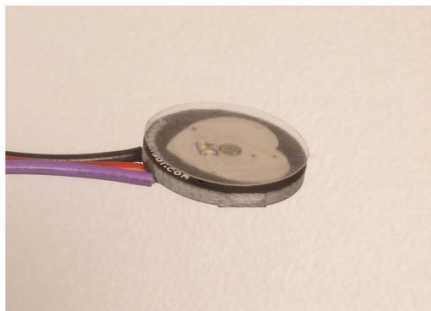
RED wire = +3V to +5V
 BLACK wire = GND
 PURPLE wire = Signal



The Pulse Sensor can be connected to arduino, or plugged into a breadboard. Before we get it up and running, we need to protect the exposed circuitry so you can get a reliable heart beat signal.

Preparing the Pulse Sensor

Before you really start using the sensor you want to insulate the board from your (naturally) sweaty/oily fingers. The Pulse Sensor is an exposed circuit board, and if you touch the solder points, you could short the board, or introduce unwanted signal noise. We will use a thin film of vinyl to seal the sensor side. Find the small page of four clear round stickers in your kit, and peel one off. Then center it on the Pulse Sensor. It should fit perfectly.



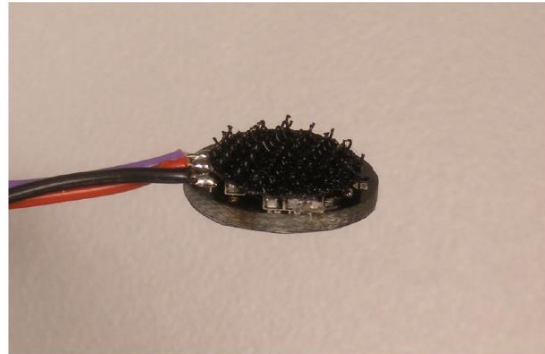
When you are happy with the way it's lined up, squeeze it onto the face all at once! The sticker (made of vinyl) will kind of stretch over the sensor and give it a nice close fit. If you get a wrinkle, don't worry, just press it down really hard and it should stick. We gave you 4, so you can replace it if necessary.

That takes care of the front side. The vinyl sticker offers very good protection for the underlying circuit, and we rate it 'water resistant'. meaning: it can stand to get splashed on, but don't throw it in the pool!

If this is your first time working with Pulse Sensor, you're probably eager to get started, and not sure if you want to use the ear-clip or finger-strap (or other thing). The back of the Pulse Sensor has even more exposed contacts than the front, so you need to make sure that you don't let it touch anything conductive or wet.

The easiest and quickest way to protect the back side from undesirable shorts or noise is to simply stick a velcro dot there for now. The dot will keep your parts away from the Pulse Sensor parts enough for you to get a good feel for the sensor and decide how you want to mount it. You'll find that the velcro dot comes off easily, and stores back on the little strip of plastic next to the other one.

Notice that the electrical connections are still exposed! We only recommend this as a temporary setup so you can get started. We show you how to better seal the Pulse Sensor later in this document.



Running The Pulse Sensor Code

Get the latest Arduino and Processing Pulse Sensor software here <http://pulsesensor.com/downloads/>

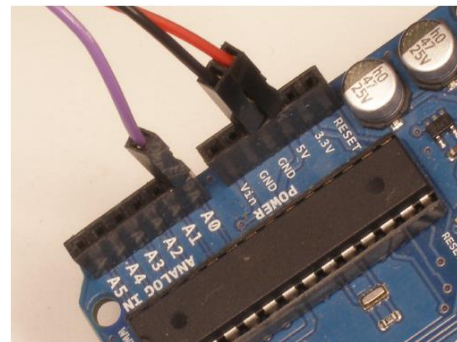
Arduino code is called "PulseSensorAmped_Arduino-xx"

The Processing code is called "PulseSensorAmped_Processing-xx"

We strongly advise that you DO NOT connect the Pulse Sensor to your body while your computer or arduino is being powered from the mains AC line. That goes for charging laptops and DC power supplies. Please be safe and isolate yourself from the power grid, or work under battery power.

Connect the Pulse Sensor to: +V (red), Ground (black), and Analog Pin 0 (purple) on your favorite Arduino, or Arduino compatible device, and upload the 'PulseSensorAmped_Arduino-xx' sketch.

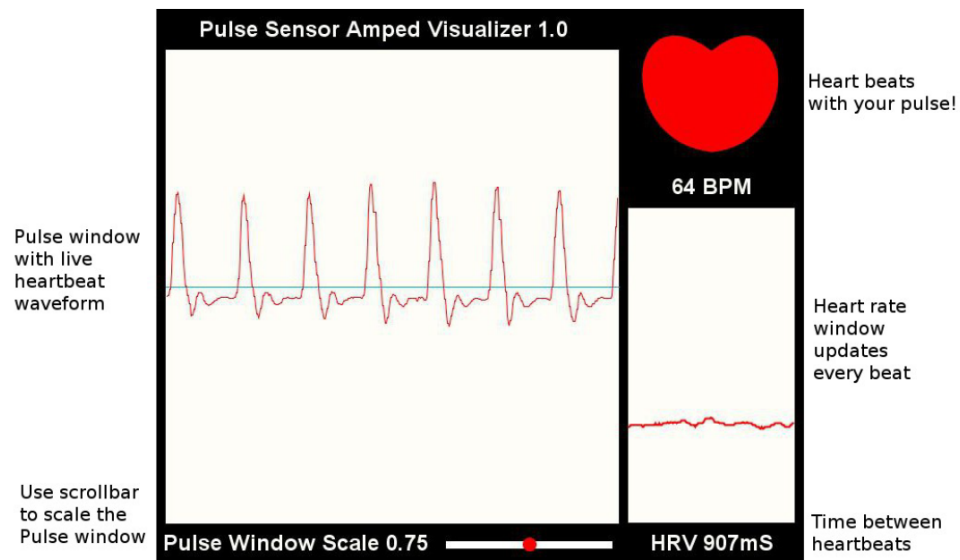
note: If you want to power Pulse Sensor Amped with low voltage (3.3V for example), make sure you have this line of code in the setup()
`analogReference(EXTERNAL);`
 Also, make sure that you apply the lower voltage to the Arduino Aref pin (next to pin 13).



After it's done uploading, you should see Arduino pin 13 blink in time with your heartbeat when you hold the sensor on your fingertip. If you grip the sensor too hard, you will squeeze all the blood out of your fingertip and there will be no signal! If you hold it too lightly, you will invite noise from movement and ambient light. Sweet

Spot pressure on the Pulse Sensor will give a nice clean signal. You may need to play around and try different parts of your body and pressures. If you see an intermittent blink, or no blink, you might be a zombie or a robot.

To view the heartbeat waveform and check your heart rate, you can use the Processing sketch that we made. Start up Processing on your computer and run the 'PulseSensorAmped_Processing-xx' sketch. This is our data visualization software, and it looks like this.



note: If you get an error when starting this code, you may need to make sure you are selecting the right serial port. Check the Troubleshooting section below..

The large main window shows a graph of raw sensor data over time. The Pulse Sensor Data Window can be scaled using the scrollbar at the bottom if you have a very large or very small signal. At the right of the screen, a smaller data window graphs heart rate over time. This graph advances every pulse, and the Beats Per Minute is updated every pulse as a running average of the last ten pulses. The big red heart in the upper right also pulses to the time of your heartbeat. When you hold the Pulse Sensor to your fingertip or earlobe or (fill in body part here) you should see a nice heartbeat waveform like the one above. If you don't, and you're sure you're not a zombie, try the sensor on different parts of your body that have capillary tissue. We've had good results on the side of the nose, middle of the forehead, palm, and lower lip. We're all different, original organisms. Play around and find the best spot on you and your friends. As you are testing and getting used to the sensor. You may find that some fingers or parts of fingers are better than others. For example, I find that when I position the sensor so that the edge of the PCB is at the bottom edge of my earlobe I get an awesome signal. Also, people with cold hands or poor circulation may have a harder time reading the pulse. Run your hands under warm water, or do some jumping-jacks!

Arduino and Processing programming environments available for download here:
www.arduino.cc www.processing.org

Sealing the Back Side of Pulse Sensor

Basic protection for the back of the Pulse Sensor
and prep for attaching Velcro Dot.

It's really important to protect the exposed Pulse Sensor circuitry so the sweat of your fingertips or earlobe (or wherever) doesn't cause signal noise or a short circuit. This How-To uses hot glue, which can be removed easily or re-worked if you want to change where you've stuck your Pulse Sensor. We love hot glue!

The other things you'll need are:

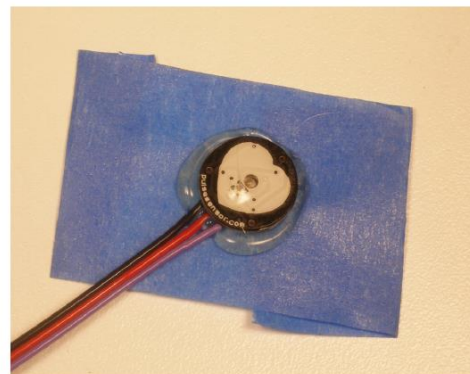
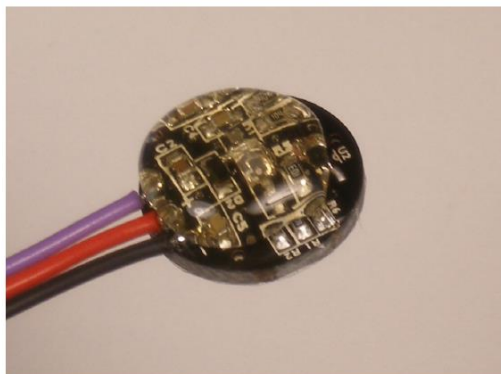
Hot Glue Gun

Blue Tape (any tape should do ok)

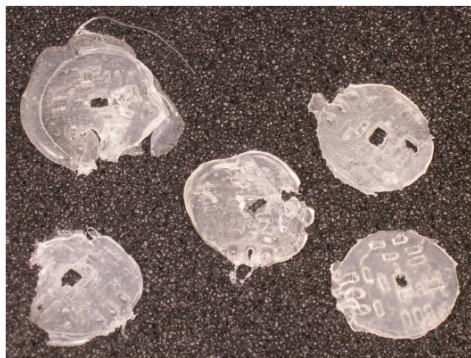
Nail Trimmers (or your favorite trimming device. Flush-cut wire snips work well too)

Read these instructions all the way through before you start!

First, attach the clear vinyl sticker to the front of your Pulse Sensor, as shown above. Then put a blob of hot glue on the back, right over the circuit. Size can be difficult to judge sometimes. What I meant was put a hot glue blob about the size of a kidney bean on the back side of the Pulse Sensor.

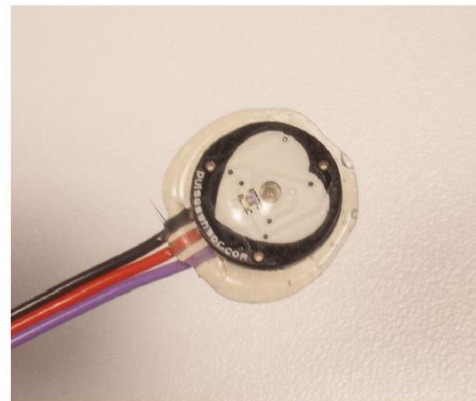
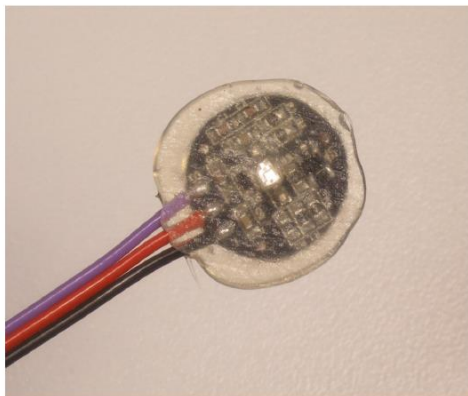
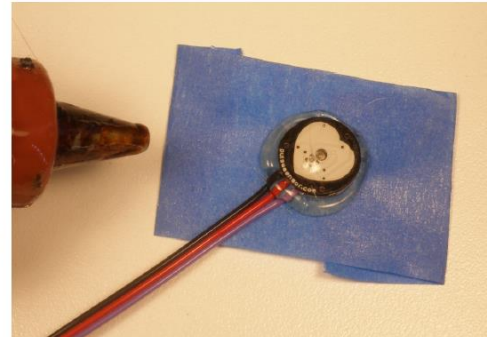


Then, while the glue is still very hot, press the Pulse Sensor glue-side-down onto the sticky side of a piece of blue tape (I believe that blue tape has magical properties, but if you don't have blue tape other kinds of tape will work just as well).

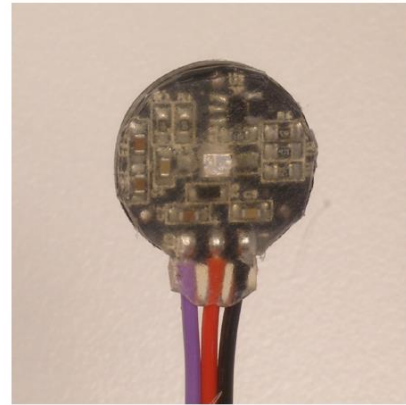


note: The tallest thing on the back of the Pulse Sensor is the green LED housing right in the middle. Use it to make the hot-glue seal thin and strong. When you press evenly until the back of the LED touches, all the conductive parts will be covered with hot glue. If the glue doesn't ooze out all around, let it cool down, then peel it from the Pulse Sensor and try again. Once the glue has cooled down and has some body, you can peel it off easily. Here's some pics of hot glue 'impressions' that I took during the making of this guide.

Next put a small dab of hot glue on the front of the cables, where they attach to the Pulse Sensor circuit board. This will bond to the other glue that's there and act as a strain-relief for the cable connection. This is important because the cable connection can wear out over time.

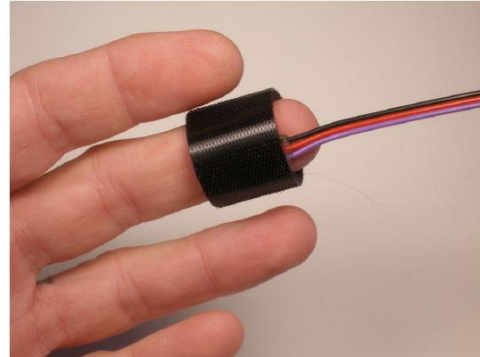
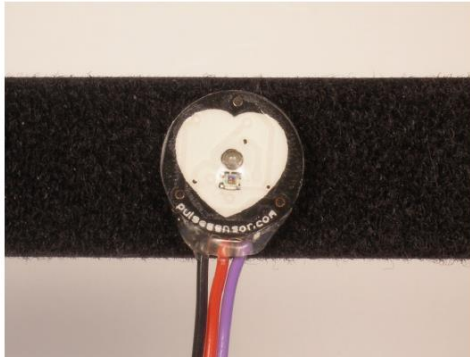


Once the hot glue has cooled (wait for it!) the blue tape will peel off very easily. Check your work to make sure that there are not exposed electrical connections! Next is trimming. I find the easiest way is to use nail clippers. Flush cutting wire snips work too. Take care not to clip the wires!!! And leave enough around the cable to act as a good strain-relief



This is the basic Pulse Sensor Hot Glue Seal, It's also got the clear vinyl sticker on the front face. We're calling this 'Water Resistant', ready to be handled and passed around from fingers to earlobes or whatever. It is not advised to submerge or soak the Pulse Sensor with this basic seal.

Now you can stick on the velcro dot (included) and make a finger strap with the velcro tape (included)!



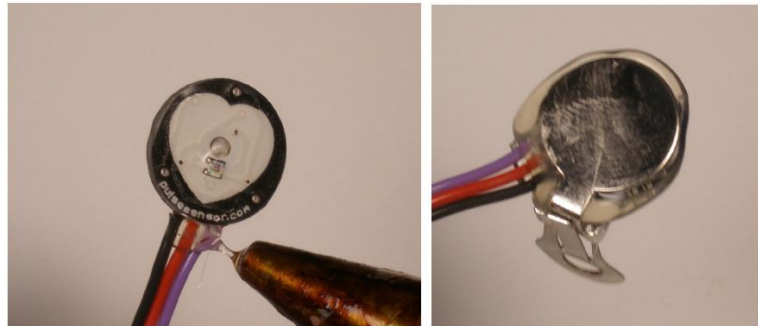
Attaching the Ear Clip

We looked all over, and were lucky enough to find an ear clip that fits the Pulse Sensor perfectly. The earlobe is a great place to attach Pulse Sensor. Here's some instruction on how to do it.

It is important to apply some strain relief to the cable connection where it meets the Pulse Sensor PCB. The little wire connections can wear out and break (or short on something) over time. We can do this with hot glue, like we did in the previous example.

First, attach a clear vinyl sticker to the front of the Pulse Sensor if you have not already. Then, put a small dab of hot glue on the front of the cables right where they meet the PCB. Get some on the edge of the PCB too, that will help. Remember, if you don't like the blob you've made for any reason, it's easy to remove once it cools down.

Next place the Pulse Sensor face down, and put a dab of glue about the size of a kidney bean on the back as illustrated above. Center the round part of the ear clip on the sensor and press it into the hot glue. The tallest component on the back is the plastic body of the reverse mount LED, and if you press it evenly it will help keep the metal of the ear clip from contacting any of the component connections.



Allow the hot glue to ooze out around the ear clip. That will ensure good coverage. Take care not to let the hot glue cover around the ear clip hinge, as that could get in the way of it working. Trimming is easy with nail clippers (as above) or your trimming tool of choice. Don't trim the wires by mistake!!!



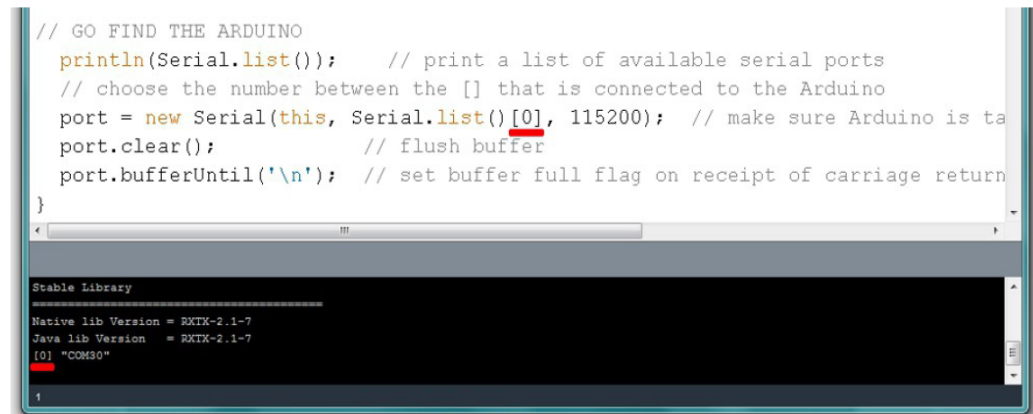
Take a good look at your work with a magnifying glass to be sure that you haven't made contact with any of the solder joints, then plug it in and test it. Hot glue is also great because it is easy to remove or re-work if you need to.

Troubleshooting:

Processing Sketch gives me a COM port error at startup.

Make sure you are plugged into an Arduino board, that it is working correctly, and running our firmware.

Check to see if you have the right serial port. The code underlined in **red** should match the correct port number in the terminal window at the bottom of Processing IDE.



```
// GO FIND THE ARDUINO
println	Serial.list(); // print a list of available serial ports
// choose the number between the [] that is connected to the Arduino
port = new Serial(this, Serial.list()[0], 115200); // make sure Arduino is ta
port.clear(); // flush buffer
port.bufferUntil('\n'); // set buffer full flag on receipt of carriage return
}

Stable Library
Native lib Version = RXTX-2.1-7
Java lib Version = RXTX-2.1-7
[0] "COM30"
```

Processing gives an RXTX mismatch warning, then nothing happens

The RXTX mismatch problem can be resolved by making sure you are running the latest version of Processing and Arduino.

If you're still having trouble, go to <http://processing.org/reference/libraries/serial/index.html> for more info.

Lampiran 7. Source Code

```

#include <Wire.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(0, 1, 5, 4, 3, 2);
int pulsePin = 0;
int blinkPin = 13;
int saklar1 = 6;
int saklar2 = 7;
int saklar3 = 8;
int mulai = 9;
int berhenti = 10;
int kondisi = 0;
int enable = 0;
volatile int BPM;
volatile int Signal;
volatile int IBI = 600;
volatile boolean Pulse = false;
volatile boolean QS = false;
void setup() {

  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("Pengukuran");
  lcd.setCursor(0, 1);
  lcd.print("Denyut");
  lcd.setCursor(7, 1);
  lcd.print("Nadi");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("PILIH USIA");
  delay(1000);
  lcd.clear();
  pinMode(saklar1, INPUT_PULLUP);
  pinMode(saklar2, INPUT_PULLUP);
  pinMode(saklar3, INPUT_PULLUP);
  pinMode(mulai, INPUT_PULLUP);
  pinMode(berhenti, INPUT_PULLUP);
  pinMode(blinkPin, OUTPUT);

```

```

}
void loop()
{
  if (digitalRead(saklar1) == LOW && digitalRead(saklar2) == HIGH &&
digitalRead(saklar3) == HIGH) {
    kondisi = 1;
    lcd.setCursor(0, 0);
    lcd.print("DEWASA");
    delay(1000);
    lcd.clear();
  }
  if (digitalRead(saklar1) == HIGH && digitalRead(saklar2) == LOW &&
digitalRead(saklar3) == HIGH) {
    kondisi = 2;
    lcd.setCursor(0, 0);
    lcd.print("ANAK");
    delay(1000);
    lcd.clear();
  }
  if (digitalRead(saklar1) == HIGH && digitalRead(saklar2) == HIGH &&
digitalRead(saklar3) == LOW) {
    kondisi = 3;
    lcd.setCursor(0, 0);
    lcd.print("BAYI");
    delay(1000);
    lcd.clear();
  }
  if (digitalRead(mulai) == LOW && digitalRead(berhenti) == HIGH) {
    enable = 1;
  }
  if (enable == 1) {
    interruptSetup();
    if (kondisi == 1) {
      dewasa();
      sensor();
    }
    if (kondisi == 2) {
      anak();
      sensor();
    }
  }
}

```

```

    if (kondisi == 3) {
        bayi();
        sensor();
    }
    if (digitalRead(berhenti) == LOW) {
        enable = 0;
    }
    if (enable == 0) {
        setup();
    }
}
}

void sensor() {
    lcd.setCursor(8, 0);
    lcd.print(BPM);
    lcd.print(" BPM ");
}

void bayi() {
    lcd.setCursor(0, 0);
    lcd.print("Denyut:");
    lcd.setCursor(0, 1);
    lcd.print("Status:");
    if (BPM == 0 ) {
        lcd.setCursor(8, 1);
        lcd.print("DEAD !!");

    }
    else if (BPM < 80 ) {
        lcd.setCursor(8, 1);
        lcd.print("BAKIKARDIA ");

    }
    else if (BPM > 160 ) {
        lcd.setCursor(8, 1);
        lcd.print("TAKIKARDIA ");

    }
    else {
        lcd.setCursor(8, 1);
        lcd.print("NORMAL ");
    }
}

```

```

    }
}
void anak() {
    lcd.setCursor(0, 0);
    lcd.print("Denyut:");
    lcd.setCursor(0, 1);
    lcd.print("Status:");
    if (BPM == 0 ) {
        lcd.setCursor(8, 1);
        lcd.print("DEAD !!");

    }
    else if (BPM < 70 ) {
        lcd.setCursor(8, 1);
        lcd.print("BAKIKARDIA ");

    }
    else if (BPM > 130 ) {
        lcd.setCursor(8, 1);
        lcd.print("TAKIKARDIA ");

    }
    else {
        lcd.setCursor(8, 1);
        lcd.print("NORMAL ");

    }
}
void dewasa() {
    lcd.setCursor(0, 0);
    lcd.print("Denyut:");
    lcd.setCursor(0, 1);
    lcd.print("Status:");
    if (BPM == 0 ) {
        lcd.setCursor(8, 1);
        lcd.print("DEAD !!");

    }
    else if (BPM < 60 ) {

```

```

    lcd.setCursor(8, 1);
    lcd.print("BAKIKARDIA ");

}
else if (BPM > 100 ) {
    lcd.setCursor(8, 1);
    lcd.print("TAKIKARDIA ");

}
else {
    lcd.setCursor(8, 1);
    lcd.print("NORMAL ");

}
}
}

```

#interrupt

```

volatile int rate[10];
volatile unsigned long sampleCounter = 0;
volatile unsigned long lastBeatTime = 0;
volatile int P = 512;
volatile int T = 512;
volatile int thresh = 512;
volatile int amp = 100;
volatile boolean firstBeat = true;
volatile boolean secondBeat = true;
void interruptSetup(){
    TCCR2A = 0x02;
    TCCR2B = 0x06;
    OCR2A = 0X7C;
    TIMSK2 = 0x02;
    sei(); }
ISR(TIMER2_COMPA_vect){
    cli();
    Signal = analogRead(pulsePin);
    sampleCounter += 2;
    int N = sampleCounter - lastBeatTime;
    if(Signal < thresh && N > (IBI/5)*3){
        if (Signal < T){ T = Signal; } }
    }

```

```

if(Signal > thresh && Signal > P){
P = Signal; }
if (N > 250){
if ( (Signal > thresh) && (Pulse == false) && (N > (IBI/5)*3) ){
Pulse = true;
digitalWrite(blinkPin,HIGH);
IBI = sampleCounter - lastBeatTime;
lastBeatTime = sampleCounter;
if(firstBeat){
firstBeat = false;
return; }
if(secondBeat){ secondBeat = false;
for(int i=0; i<=9; i++){ rate[i] = IBI; } }
word runningTotal = 0;
for(int i=0; i<=8; i++){
rate[i] = rate[i+1];
runningTotal += rate[i]; }
rate[9] = IBI;
runningTotal += rate[9];
runningTotal /= 10;
BPM = 60000/runningTotal;
QS = true;
}
}
if (Signal < thresh && Pulse == true){
digitalWrite(blinkPin,LOW);
Pulse = false;
amp = P - T;
thresh = amp/2 + T;
P = thresh; T = thresh; }
if (N > 2500){
thresh = 512;
P = 512;
T = 512;
lastBeatTime = sampleCounter;
firstBeat = true;
secondBeat = true; }
sei(); }

```