

**PERANCANGAN ALAT PENDETEKSI DETAK JANTUNG
1 LEAD BERBASIS *INTERNET OF THINGS***

SKRIPSI

Disusun Untuk Memenuhi Sebagian Persyaratan
Dalam Rangka Menyelesaikan Pendidikan Program Studi
Teknologi Rekayasa Elektro-medis
Program Sarjana Terapan



Disusun Oleh:

TARISA SUKMAWATI
2019050018

FAKULTAS SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI SAINS DAN KESEHATAN
PKU MUHAMMADIYAH SURAKARTA
2023

LEMBAR PERSETUJUAN

Penelitian dengan judul “Perancangan Alat Pendeteksi Detak Jantung 1 *Lead* Berbasis IoT”, telah diperiksa dan disetujui untuk diujikan dihadapan Tim Penguji Tugas Akhir Program Studi Teknologi Rekayasa Elektro-medis
Program Sarjana Terapan
Fakultas Sains dan Teknologi
ITS PKU Muhammadiyah Surakarta



Nama Pembimbing

Tanda Tangan

Tanggal

Fitri Anindyahadi, S.T., M.T.
NIDN. 0616099101


.....

8/9 '23
.....

Eko Nugroho, S.T., M.Eng.
NIDN. 0608128404


.....

8/9 '23
.....




LEMBAR PENGESAHAN

**PERANCANGAN ALAT PENDETEKSI DETAK JANTUNG 1 LEAD
BERBASIS *INTERNET OF THINGS***



Penelitian ini telah diseminarkan dan diujikan
Pada tanggal: 26 Agustus 2023

Susunan Tim Penguji:

No.	Nama	Jabatan dalam tim	Tanda Tangan
1	<u>Kuat Supriyadi, BE, SE, ST, MM, MT</u> NIDK. 8901500020	Ketua Penguji	
2	<u>Fitri Anindyahadi, S.T., M.T.</u> NIDN. 0616099101	Penguji 1	
3	<u>Eko Nugroho, S.T., M.Eng.</u> NIDN. 0608128404	Penguji 2	

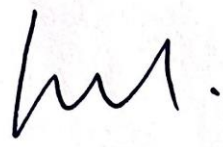
Mengetahui,

Dekan Fakultas Sains dan Teknologi



Dr. Ida Untari, S.K.M., M.Kes.
NIDN/ 0629037604

Ka.Prodi DIV Elektro-medis



Ipin Prasojo, S.Pd.T., M.Kom.
NIDN. 0614128002

SURAT PERNYATAAN KEASLIAN

Penulis menyatakan dengan sesungguhnya bahwa penelitian sebagai tugas akhir dengan judul:

PERANCANGAN ALAT PENDETEKSI DETAK JANTUNG 1 *LEAD* BERBASIS *INTERNET OF THINGS*

merupakan asli karya penulis sendiri. Isi dalam penelitian ini tidak terdapat karya yang pernah diajukan oleh orang lain atau kelompok lain untuk memperoleh gelar akademis disuatu Institusi Pendidikan, dan sepanjang pengetahuan penulis juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain atau kelompok lain kecuali yang secara tertulis dituangkan dalam naskah ini dan disebutkan dalam daftar pustaka.

Surakarta, 2 Agustus 2023

Tarisa Sukmawati

KATA PENGANTAR

Bismillahirrohmannirrohim

Puji syukur penulis ucapkan atas kehadiran Tuhan Yang Maha Esa karena dengan Rahmat karunia, serta taufik dan hidayah-Nya penulis dapat menyelesaikan skripsi yang berjudul “Perancangan Alat Pendeteksi Detak Jantung 1 *Lead* Berbasis *Internet of Things*” sebagai salah satu syarat menyelesaikan pendidikan.

Penulis berharap skripsi ini dapat berguna dalam rangka menambah wawasan serta pengetahuan kita mengenai teori yang digunakan. Penulis juga menyadari sepenuhnya bahwa di dalam mengerjakan skripsi ini terdapat kekurangan dan jauh dari apa yang diharapkan. Untuk itu, penulis berharap dengan adanya kritik, saran, dan usulan demi perbaikan di masa yang akan datang, mengingat tidak ada sesuatu yang sempurna tanpa sarana yang membangun. Penyusunan skripsi ini sepenuhnya tidak terlepas dari bimbingan dan bantuan moral dari berbagai pihak. Maka, pada kesempatan ini penulis ucapkan terima kasih kepada:

1. Bapak Ipin Prasajo, S.Pd.T., M. Kom., selaku Ketua Program Studi DIV Teknologi Rekayasa Elektro-medis yang telah membantu dalam proses akademik dan praktik selama menempuh pendidikan di ITS PKU Muhammadiyah Surakarta.
2. Ibu Fitri Anindyahadi, S.T., M.T., selaku Pembimbing I Skripsi yang telah membimbing dengan penuh kesabaran, memberikan waktu dan tenaganya di sela kesibukan beliau sehingga skripsi ini dapat terselesaikan.
3. Bapak Eko Nugroho, S.T. M.Eng., selaku Pembimbing II Skripsi yang telah membimbing dengan penuh kesabaran, memberikan waktu dan tenaganya di sela kesibukan beliau sehingga skripsi ini dapat terselesaikan.
4. Bapakku Sukiman dan Ibuku Sri Lestari tercinta yang selalu memberikan dukungan do'a, lahir, batin, dan materi serta selalu memberikan semangat di saat penulis mulai kehilangan asa.

5. Adikku Elya, yang telah berkenan memberikan begitu banyak semangat dan merelakan waktunya untuk menjadi probandus berkali-kali.
6. Mbah Kakung, Mak Mariyam, dan Mbah Giyek, terimakasih atas dukungannya.
7. Teman-teman NikPik, “Muhammad Fikri, Naufal Setya Budi, Rizki Bagus P, Syahda Aulia P, Gustarika Kusuma Y” dan Arkhan Nur Sayogo yang sudah mau berteman dekat, berjuang bersama-sama, membantu selama perkuliahan, tempat berkeluh kesah selama 4 tahun ini, terima kasih atas kehangatannya.
8. Orang-orang yang menginspirasi di kehidupanku.
9. Teman seangkatan di ITS PKU Muhammadiyah Surakarta.
10. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang telah membantu sehingga terselesaikannya skripsi ini.

Akhir kata semoga Allah Subhanallahu Wa Ta’ala memberikan balasan atas budi baik dan bantuan mereka serta semoga skripsi ini dapat memberikan manfaat bagi penulis sendiri dan pembaca. Allahumma Aamiin...

Surakarta, 2 Agustus 2023

Penyusun

Tarisa Sukmawati

ABSTRAK

Perancangan Alat Pendeteksi Detak Jantung 1 *Lead* Berbasis *Internet of Things*

Tarisa Sukmawati¹, Fitri Anindyahadi², Eko Nugroho³

Penyakit jantung merupakan penyakit yang umum diderita dari berbagai kalangan usia saat ini. Perlunya mencegah penyakit jantung dilakukan sejak dini dengan tujuan mengurangi tingkat kematian yang disebabkan oleh jantung, terutama gagal jantung. Seringnya pasien melakukan pengecekan detak jantung yang dicetak menggunakan kertas mengakibatkan menumpuknya limbah kertas. Oleh karena itu diperlukan adanya suatu sistem untuk memonitoring kesehatan detak jantung tanpa harus mencetak hasil rekaman jantung. Penelitian ini merupakan penelitian pengembangan alat monitoring detak jantung manusia dengan menggunakan Electrocardiograph (ECG) yang ditanamkan dalam sistem IoT untuk dimonitoring melalui aplikasi. Alat ini menggunakan sensor AD8232 sebagai sensor detak jantung dan ESP32 sebagai mikrokontroler untuk pemrosesan datanya. Data yang diperoleh dalam bentuk sinyal dikonversi ke dalam satuan bpm dan dikirim ke aplikasi *Blynk* melalui koneksi Wi-Fi. Alat ini sudah dilengkapi dengan baterai yang dapat diisi ulang agar penggunaan alat ini dapat bersifat portabel. Hasil penelitian menunjukkan bahwa tingkat *error* alat pendeteksi detak jantung yaitu berdasarkan data pengukuran yang dikonversi ke dalam satuan bpm yaitu 1,6 % dan nilai *error* yang didapat saat perhitungan manual menggunakan sinyal detak jantung yaitu 2,3 % hasil ini menunjukkan bahwa alat layak digunakan.

Kata Kunci: pendeteksi, detak, IoT

1. Mahasiswa program studi ITS PKU Muhammadiyah Surakarta
2. Dosen program studi ITS PKU Muhammadiyah Surakarta
3. Dosen program studi ITS PKU Muhammadiyah Surakarta

ABSTRACT

Design of 1 Lead Heart Rate Detection Device Based on Internet of Things

Tarisa Sukmawati¹, Fitri Anindyahadi², Eko Nugroho³

Heart disease is a common disease suffered from various ages today. The need to prevent heart disease is done early with the aim of reducing the death rate caused by the heart, especially heart failure. The frequency of patients checking heart rate printed using paper results in the accumulation of paper pollution. Therefore, it is necessary to have a system to monitor heart rate health without having to print the results of heart recordings. This research is a development of a human heart rate monitoring tool using an Electrocardiograph (ECG) embedded in an IoT system to be monitored through an application. This tool uses the AD8232 sensor as a heart rate sensor and ESP32 as a microcontroller for data processing. The data obtained in the form of signals is converted into bpm units and sent to the Blynk application via Wi-Fi connection. This tool is equipped with a rechargeable battery so that the use of this tool can be portable. The results showed that the error rate of the heart rate detection device is based on measurement data converted into bpm units, which is 1,6% and the error value obtained when manual calculation using heart rate signals is 2.3%, these results indicate that the device is feasible to use.

Keyword: detection, rate, IoT

- 1. Students of ITS study program PKU Muhammadiyah Surakarta*
- 2. Lecturer of the ITS PKU Muhammadiyah Surakarta study program*
- 3. Lecturer of the ITS PKU Muhammadiyah Surakarta study program*

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN	iii
SURAT PERNYATAAN KEASLIAN	iv
KATA PENGANTAR	v
ABSTRAK	vii
<i>ABSTRACT</i>	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN.....	1
A. Latar Belakang	1
B. Rumusan Masalah	3
C. Tujuan Penelitian.....	3
D. Manfaat Penelitian	3
E. Keaslian Penelitian	4
BAB II TINJAUAN PUSTAKA.....	7
A. Penelitian Yang Relevan	7
B. Landasan Teori.....	9
BAB III METODOLOGI PENELITIAN.....	19
A. Metode Penelitian	19
B. Alat dan Bahan Penelitian	20
C. Tempat dan Waktu Penelitian	21
D. Teknik Analisis Data.....	22
BAB IV HASIL DAN PEMBAHASAN	28
A. Hasil Pembuatan Alat.....	28
B. Pengujian Alat	29
C. Pembahasan	33
D. Keterbatasan.....	41
BAB V PENUTUP.....	42

A. Simpulan	42
B. Saran.....	42
DAFTAR PUSTAKA	43
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
Gambar 2. 1. Gelombang Sinyal Electrocardiograph.	12
Gambar 2. 2. Sensor AD8232.	13
Gambar 2. 3. Lokasi sadapan elektroda, biru LA, hitam RA, Merah RL.	13
Gambar 2. 4. Sadapan ekstremitas utama dan tata letak elektrodanya.	14
Gambar 2. 5. ESP32.	15
Gambar 2. 6. Skema Rangkaian Pin Out ESP32.	16
Gambar 2. 7. Saklar 3 kaki.	16
Gambar 2. 8. Oled.	17
Gambar 2. 9. Baterai Lithium Ion 18650.	18
Gambar 3. 1. Metode Perancangan.	22
Gambar 3. 2. Skema Rangkaian Alat.	23
Gambar 3. 3. Perancangan Sistem Alat.	24
Gambar 3. 4. Flowchart Sistem.	26
Gambar 4. 1. Perakitan Alat.	28
Gambar 4. 2. Hasil Akhir Alat.	28
Gambar 4. 3. Tampilan Pengukuran Detak Jantung di Blynk.	33
Gambar 4. 4. Grafik Hasil Pengukuran: (a)Pertama, (b)Kedua, (c)Ketiga, (d)Keempat, (e)Kelima, (f)Keenam.	34
Gambar 4. 5. Sinyal Detak Jantung Pengukuran Pertama.	35
Gambar 4. 6. Sinyal Detak Jantung Pengukuran Kedua.	36
Gambar 4. 7. Sinyal Detak Jantung Pengukuran Ketiga.	37
Gambar 4. 8. Sinyal Detak Jantung Pengukuran Keempat.	38
Gambar 4. 9. Sinyal Detak Jantung Pengukuran Kelima.	39

DAFTAR TABEL

	Halaman
Tabel 2. 1. Detak jantung normal pada pria.	10
Tabel 2. 2. Detak jantung normal pada Wanita.	10
Tabel 2. 3. Acuan Perhitungan bpm.	11
Tabel 2. 4. Spesifikasi ESP32.	15
Tabel 3. 1. Alat dan Bahan.	20
Tabel 3. 2. Jadwal Penelitian.	21
Tabel 4. 1. Data Pengujian Responden Pertama.	30
Tabel 4. 2. Data Pengujian Responden Kedua.	30
Tabel 4. 3. Data Pengujian Responden Ketiga.	31
Tabel 4. 4. Data Pengujian Responden Keempat.	31
Tabel 4. 5. Data Pengujian Responden Kelima.	32
Tabel 4. 6. Data Pengujian Responden Keenam.	32

BAB I

PENDAHULUAN

A. Latar Belakang

Penyakit jantung merupakan salah satu penyakit yang paling banyak diderita saat ini karena berbagai faktor penyebabnya, seperti tekanan darah tinggi, diabetes, fluktuasi kolesterol, kelelahan dan banyak lagi lainnya. Penyakit jantung terbagi menjadi beberapa jenis diantaranya penyakit jantung coroner, gangguan irama jantung, penyakit jantung bawaan, gagal jantung, penyakit katup jantung. Salah satu gangguan kardiovaskuler (gangguan berkenaan dengan jantung dan pembuluh darah) yang paling banyak terjadi yakni gagal jantung yang memiliki angka mortalitas (kematian) dan morbiditas (kesakitan) yang tinggi baik di negara maju maupun di negara berkembang seperti Indonesia (Idu et al., 2021).

Hal yang dilakukan untuk mendeteksi adanya gangguan pada jantung, salah satunya dengan mengecek parameter konduktivitas (sistem listrik yang menggerakkan siklus) jantung (Kukuh Setyohadi, Ibrahim, 2021). Dari pengukuran detak jantung yang didapat dapat mendiagnosa apakah ada penyakit yang diderita oleh pasien. Selain itu, pengukuran ini juga dapat menganalisa tingkat detak per menit dari pasien, dimana itu dapat mengurangi stress berlebih yang dapat mengganggu aktifitas. Cara mengetahui detak jantung bisa dilakukan secara manual (dengan menghitung detak nadi di pergelangan tangan selama satu menit) dan secara elektrik yaitu dengan menggunakan ECG (Sufri & Aswardi, 2020).

Perancangan ECG untuk pengecekan denyut jantung yang *paperless* dengan tujuan untuk mengurangi penularan penyakit yang disalurkan oleh kertas dan mengurangi sampah kertas di dunia. Perkembangan teknologi yang sangat pesat tersebut, memungkinkan hadirnya teknologi *Internet of Things* (IoT), dimana teknologi ini bisa menghubungkan antara alat dengan jaringan internet dan saling terintegrasi. Teknologi ini dapat menampilkan hasil pengecekan ECG

melalui android atau laptop user, sehingga tidak perlu pencetakan hasil rekaman ECG *paperless*. Penggunaan komputer di masa datang mampu mendominasi pekerjaan manusia dan mengalahkan kemampuan komputasi manusia seperti mengontrol peralatan jarak jauh menggunakan media internet. Perkembangan teknologi informasi sekarang ini ditandai oleh hadirnya IoT. IoT memungkinkan pengguna mengelola dan mengoptimalkan elektronik dan peralatan listrik yang menggunakan internet. IoT merupakan layanan internet yang terintegrasi dengan pemanfaatan jenis sensor tertentu, hal ini juga memungkinkan pemantauan faktor manusia termasuk bagi kesehatan, kebugaran, perilaku, dan data lain yang berguna dalam meningkatkan kualitas hidup individu sehari-hari (Hermansyah et al., 2022).

Perkembangan ilmu pengetahuan dan teknologi yang sangat pesat, dapat mempermudah perubahan dalam bidang apapun, terutama dunia kesehatan. Seiring berkembangnya teknologi, ternyata penyakit juga berkembang, dimana berbagai macam penyakit muncul di sekitar kita. Beragam penyakit ini muncul begitu saja tanpa kita sadari, terutama pada jantung dimana jantung berperan penting dalam organ tubuh manusia. Pengecekan kondisi detak jantung secara berkala perlu dilakukan untuk mendeteksi adanya gangguan pada jantung.

Berdasarkan uraian tersebut, maka pada tugas akhir ini dibuat "Perancangan Alat Pendeteksi Detak Jantung 1 *Lead* Berbasis IoT". Sistem ini akan merekam dan mencatat secara *real-time* kondisi medis pasien berupa detak jantung. Sistem ini berbasis IoT dan terintegrasi dengan aplikasi android.

B. Rumusan Masalah

Berdasarkan masalah di atas maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana merancang alat Pendeteksi Detak Jantung berbasis IoT?
2. Bagaimana proses monitoring detak jantung dengan Pendeteksi Detak Jantung berbasis IoT?
3. Bagaimana validasi alat Pendeteksi Detak Jantung berbasis IoT?

C. Tujuan Penelitian

Tujuan dari penelitian:

1. Mampu membuat dan merancang alat Pendeteksi Detak Jantung berbasis IoT.
2. Memonitoring denyut jantung melalui web server.
3. Membandingkan alat Pendeteksi Detak Jantung berbasis IoT dengan alat Pendeteksi Detak Jantung yang sudah ada.

D. Manfaat Penelitian

Manfaat dari penelitian:

1. Untuk dapat mendeteksi diagnosa penyakit jantung dengan pengembangan teknologi alat Pendeteksi Detak Jantung berbasis IoT.
2. Mengurangi penggunaan kertas (*paperless*).

E. Keaslian Penelitian

No.	Keaslian Penelitian	
1	Nama Peneliti/Tahun Judul Desain dan Variabel Penelitian Hasil Persamaan Perbedaan	: Sufri & Aswadi/2020 : Alat Pendeteksi Detak Jantung dan Kesehatan Berbasis Arduino : Konversi sinyal analog ke digital analog to digital converter : Hasil dari penelitian ini disimpulkan bahwa pengukuran detak jantung dengan menampilkan hasil di <i>Liquid Crystal Display</i> . Monitoring detak jantung menggunakan pulse sensor diperoleh rata-rata kesalahan pulse sensor sebesar 1,5547% : Bertujuan untuk menampilkan monitoring detak jantung : <ol style="list-style-type: none"> a. Peneliti menggunakan <i>Liquid Crystal Display</i> b. Peneliti menggunakan ATmega328 sebagai mikrokontroler c. Menggunakan pulse sensor sebagai sensor detak jantung
2	Nama Peneliti/Tahun Judul Desain dan Variabel Penelitian Hasil Persamaan Perbedaan	: Ria Hariri, dkk/2020 : Sistem Monitoring Detak Jantung Menggunakan Sensor AD8232 : Perancangan alat, pengujian alat dan menganalisis hasil : Hasil dari penelitian ini disimpulkan bahwa pengukuran detak jantung dengan menampilkan hasil dengan satuan BPM dan didapat presentase error antara pengukuran menggunakan sistem usulan dengan alat dari medis menunjukkan sebesar 1,21% : Menggunakan sensor AD8232 : Peneliti menggunakan ESP 8266 sebagai mikrokontroler
3	Nama Peneliti/Tahun	: Indra Agustian/2019

	Judul	:	Rancang Bangun Pemantau Detak Jantung dan Suhu Tubuh Portabel dengan Sistem IoT
	Desain dan Variabel Penelitian	:	Desain prototipe untuk pengukuran kondisi jantung
	Hasil	:	Hasil dari penelitian ini disimpulkan bahwa pengukuran detak jantung dengan menampilkan hasil yang dikirim ke <i>cloud server IoT ThingSpeak.com</i> dan dapat dipantau secara <i>offline</i> , pada penelitian ini diperoleh standar deviasi perbedaan pengukuran detak jantung adalah 1,3676 dan standar deviasi perbedaan pengukuran suhu 0,5916
	Persamaan	:	Menggunakan sensor AD8232
	Perbedaan	:	<ul style="list-style-type: none"> a. Peneliti mennggunakan ESP 8266 sebagai mikrokontroller b. Peneliti menggunakan web server <i>ThingSpeak.com</i>
4	Nama Peneliti/Tahun	:	Momin/2021
	Judul	:	Rancang Bangun Elektrokardiograf Berbasis IoT
	Desain dan Variabel Penelitian	:	Kalibrasi sinyal grafik EKG
	Hasil	:	Hasil dari penelitian ini disimpulkan bahwa berdasarkan pengujian nilai BPM pada tiga objek, diperoleh error pada objek pertama sebesar 1,47%, error onjek kedua sebesar 4,49%, dan error pada objek ketiga sebesar 0,53%, sehingga didapatkan rata-rata error sebesar 2,16%.
	Persamaan	:	Menggunakan sensor AD8232
	Perbedaan	:	<ul style="list-style-type: none"> a. Peneliti mennggunakan ESP 8266 sebagai mikrokontroller b. Peneliti menggunakan <i>Spreadsheets</i>

5	<p>Nama Peneliti/Tahun Judul</p> <p>Desain dan Variabel Penelitian Hasil</p> <p>Persamaan Perbedaan</p>	<p>: Firman Maulana/2021</p> <p>: Implementasi Sistem Telemedicine Untuk Monitoring Detak Jantung Berbasis Sensor AD8232</p> <p>: Perancangan alat, pengujian alat dan menganalisis hasil</p> <p>: Hasil dari penelitian ini disimpulkan bahwa berdasarkan pengujian monitoring detak jantung ketika sensor AD8232 mengirim data ke smartphone diperoleh <i>delay</i> dengan lonjakan rata-rata nilai maksimum sebesar 43,01 <i>ms</i>.</p> <p>: Menggunakan sensor AD8232</p> <p>: Peneliti menggunakan web server yang berbeda</p>
---	---	--

BAB II

TINJAUAN PUSTAKA

A. Penelitian Yang Relevan

Penulisan penelitian ini akan coba dikaitkan dengan karya ilmiah terdahulu. Adapun karya ilmiah yang dimaksud adalah sebagai berikut:

- a. Penelitian pada Jurnal Teknik Elektro Indonesia (JTEI) yang disusun oleh Sufri & Aswardi, (2020) dengan judul: Alat Pendeteksi Detak Jantung dan Kesehatan Berbasis Arduino. Penelitian ini membahas tentang modul yang digunakan untuk memonitoring proses pengukuran detak jantung pada posisi normal dengan menggunakan Arduino, dimana hasil dari pengukuran detak jantung akan ditampilkan pada *Liquid Crystal Display*. Mikrokontroler yang digunakan adalah ATmega328 berfungsi untuk mengontrol keseluruhan kerja sistem pemantau detak jantung. Berdasarkan hasil monitoring detak jantung menggunakan pulse sensor diperoleh rata-rata kesalahan pulse sensor sebesar 1,5547% (Sufri & Aswardi, 2020).
- b. Penelitian pada Jurnal Zetroem yang disusun oleh Ria Hariri, dkk (2020) dengan judul: Sistem Monitoring Detak Jantung Menggunakan Sensor AD8232. Penelitian ini membahas tentang monitoring detak jantung berbasis ESP 8266 sebagai mikrokontroler, pengukuran pada sensor AD8232 dilakukan sebanyak 3 kali untuk mendapatkan hasil yang dapat dianalisa berfungsi baik tidaknya alat tersebut dengan satuan BPM. Berdasarkan hasil pengukuran monitoring detak jantung diperoleh presentase error antara pengukuran menggunakan sistem usulan dengan alat dari medis menunjukkan sebesar 1, 21%(Ria Hariri et al., 2020).
- c. Penelitian pada Jurnal Amplifier yang disusun oleh Indra Agustian (2019) dengan judul: Rancang Bangun Pemantau Detak Jantung dan Suhu Tubuh Portabel Dengan Sistem IoT. Penelitian ini membahas tentang pemantauan detak jantung secara *offline* dan *online*, dimana jika secara *offline* hasil akan tampak melalui LCD dengan indikator menunjukkan jika tubuh manusia

normal atau abnormal, sedangkan secara *online* data pengukuran detak jantung dikirim ke *cloud server* IoT ThingSpeak.com melalui ESP01 sehingga data dapat dipantau secara *online*. Berdasarkan hasil penelitian diperoleh standar deviasi perbedaan pengukuran detak jantung adalah 1,3676 dan standar deviasi perbedaan pengukuran suhu 0,5916 (Agustian, 2019).

- d. Penelitian pada Jurnal Fisika yang disusun oleh Momin, dkk (2021) dengan judul: Rancang Bangun Elektrokardiograf Berbasis IoT. Penelitian ini membahas alat elektrokardiograf sebagai pengukur detak jantung manusia dengan IoT menggunakan ESP8266 yang telah terkonfigurasi dengan *Platform spreadsheet* untuk meminimalisir hilangnya data pada grafik ECG (Electrocardiograph). Berdasarkan pengujian nilai BPM (*Beat Per Minutes*) pada tiga objek, diperoleh error pada objek pertama sebesar 1,47%, error objek kedua sebesar 4,49%, dan error pada objek ketiga sebesar 0,53%, sehingga didapatkan rata-rata error sebesar 2,16% (Momin et al., 2021).
- e. Penelitian pada Jurnal Semnas Fortei Regional 7 yang disusun oleh Firman Maulana (2021) dengan judul: Implementasi Sistem Telemedicine Untuk Monitoring Detak Jantung Berbasis Sensor AD8232. Penelitian ini membahas monitoring detak jantung yang diawali dengan pengambilan data oleh sensor AD8232 yang selanjutnya data akan diolah dengan mikrokontroler ESP32 dan mengirimkan notifikasinya ke smartphone. Berdasarkan pengukuran *delay* menggunakan RMSE (*Root Mean Square Error*) dengan alat *moving average* 5 hingga *moving average* 15 diperoleh lonjakan *delay* dengan rata-rata nilai maksimum sebesar 43,01 ms (FM., 2021)

B. Landasan Teori

Dalam proses pembuatan tugas akhir ini digunakan beberapa rujukan pada proses pembuatan alat agar dapat berfungsi untuk monitoring proses pengukuran detak jantung normal.

1. Detak Jantung Manusia

Heart rate (HR) merupakan detak jantung per satuan waktu yang dihitung dengan satuan *beat per minute* (bpm). Pengambilan data HR dapat dilakukan dengan metode ECG (Novani et al., 2019). Detak jantung normal dapat dihitung ketika seseorang dalam kondisi istirahat atau tidak sedang melakukan aktivitas fisik. Untuk mengetahui kecepatan detak jantung seseorang dapat dilakukan dengan *pulse rate* yaitu dengan cara menghitung perubahan tiba-tiba dari tekanan yang dirambatkan sebagai gelombang pada dinding darah. Detak jantung juga dapat dihitung menggunakan *stethoscope* yang penggunaannya masih manual dan pengoperasiannya hanya bisa dilakukan oleh orang yang ahli di bidangnya (Sufri & Aswardi, 2020).

Detak jantung merupakan kecepatan jantung berdetak berdasarkan jumlah detak jantung per satuan waktu atau *beats per minute* (bpm). Detak jantung *beats per menit* (bpm) ini merupakan parameter untuk menunjukkan kondisi jantung, dan salah satu cara untuk mengetahui kondisi jantung seseorang adalah dengan cara mengetahui frekuensi detak jantung. Detak jantung manusia dapat diukur melalui denyut nadi dengan cara menempelkan ujung telunjuk dan jari tengah disisi pergelangan tangan. Setelah itu dilakukan perhitungan selama 10 detik dan dikalikan dengan 6, hasilnya akan berupa denyut nadi *beats per minute* (Hermansyah et al., 2022).

a. Detak jantung normal pada pria

Tabel 2. 1. Detak jantung normal pada pria.

No	Usia	Kondisi Tubuh	
		Istirahat	Aktifitas
1	0-10 tahun	100-130 bpm	130-170 bpm
2	10-17 tahun	80-100 bpm	120-160 bpm
3	>18 tahun	60-90 bpm	100-150 bpm

(sumber: Sufri & Aswardi, 2020)

b. Detak jantung normal pada wanita

Tabel 2. 2. Detak jantung normal pada Wanita.

No	Usia	Kondisi Tubuh	
		Istirahat	Aktifitas
1	0-10 tahun	105-130 bpm	135-170 bpm
2	10-17 tahun	83-100 bpm	123-160 bpm
3	>18 tahun	65-90 bpm	105-150 bpm

(sumber: Sufri & Aswardi, 2020)

c. Acuan perhitungan bpm

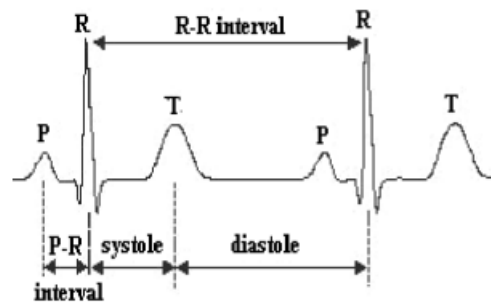
Tabel 2. 3. Acuan Perhitungan bpm.

ACUAN PENGHITUNGAN BPM	
ms	bpm
200	300.00
225	266.67
250	240.00
275	218.18
300	200.00
325	184.62
350	171.43
375	160.00
400	150.00
425	141.18
450	133.33
475	126.32
500	120.00
525	114.29
550	109.09
575	104.35
600	100.00
625	96.00
650	92.31
675	88.89
700	85.71
725	82.76
750	80.00
775	77.42
800	75.00
825	72.73
850	70.59
875	68.57
900	66.67
925	64.86
950	63.16
975	61.54
1000	60.00

(sumber: www.firebitlab.com)

2. Sensor AD8232

Sensor AD8232 adalah *board* untuk membaca atau mengukur aktivitas listrik jantung menggunakan elektroda. Aktivitas listrik ini sebagai ECG dan *output* sebagai pembacaan analog. Sensor AD8232 membantu dalam mendapatkan sinyal interval PR dan QT yang jelas karena sinyal ECG bisa sangat bising. Sensor AD8232 dirancang untuk mengekstrak, menyaring, dan memperkuat sinyal biopotensial kecil pada kondisi bising seperti gerakan atau akibat dari penempatan elektroda terpencil (Ria Hariri et al., 2020). Untuk menghitung detak jantung dapat menggunakan pendeteksian puncak gelombang R pada hasil EKG. Hal ini karena gelombang R memiliki puncak tertinggi dari puncak gelombang lainnya, sehingga akan sangat mudah untuk dideteksi (FM., 2017).



Gambar 2. 1. Gelombang Sinyal Electrocardiograph.

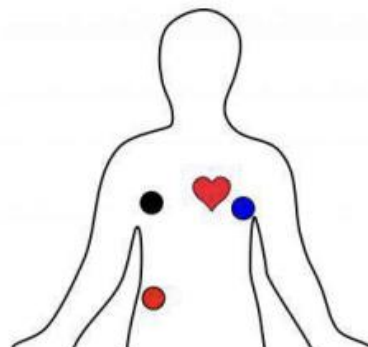
Ada tiga konfigurasi rangkaian utama yang dapat digunakan berdasarkan lokasi penempatan elektroda, pertama rangkaian untuk pengukuran detak jantung dengan lokasi deteksi dekat dengan area jantung, rangkaian ini hanya menggunakan dua elektroda. Kedua, rangkaian dengan lokasi elektroda pada lengan yang biasa digunakan untuk mendeteksi denyut nadi, menggunakan tiga elektroda. Ketiga, rangkaian untuk “*cardiac monitoring*” ini dirancang untuk memantau bentuk gelombang ECG (Agustian, 2019).



Gambar 2. 2. Sensor AD8232.

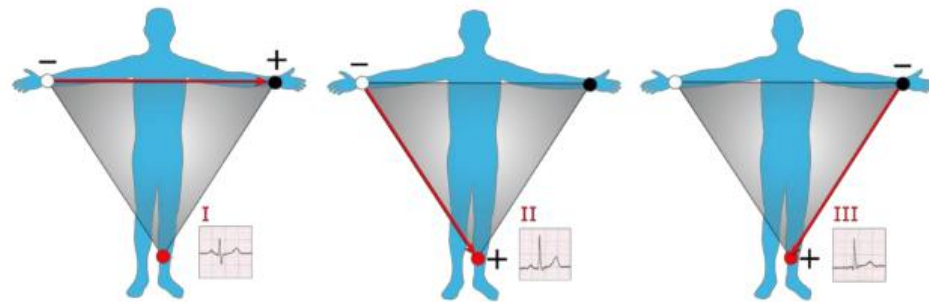
(sumber: www.ebay.com)

Posisi peletakkan pad elektroda pada dada pasien merupakan hal penting. Dalam transmisi sinyal aukultasi, dimana komponen terpenting adalah data yang diambil dari auskultasi jantung. Maka tahapan yang pertama dilakukan adalah mengambil data auskultasi dengan meletakkan sensor pada pergelangan kedua tangan dan pergelangan kaki kanan atau pada perut bagian kanan. Posisi jantung manusia adalah pada tulang iga manusia keenam disebelah kiri dada manusia, atau 5 cm di atas ulu hati di sebelah kiri. Peletakkan sensor sangat berpengaruh, karena apabila sensor tidak diletakkan pada bagian yang tepat maka data yang akan diterima berupa data *noise* (Isma et al., 2020).



Gambar 2. 3. Lokasi sadapan elektroda, biru LA, hitam RA, Merah RL.

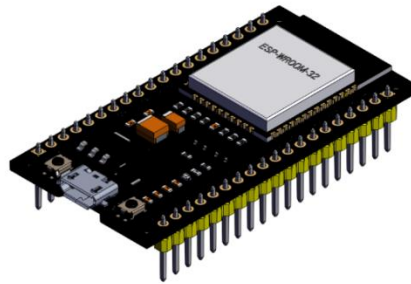
(sumber: *Agustian, 2019*)



Gambar 2. 4. Sadapan ekstremitas utama dan tata letak elektrodanya.

3. ESP32

ESP32 adalah nama dari mikrokontroler yang dirancang oleh perusahaan yang berbasis di Shanghai, China. ESP32 menawarkan solusi jaringan Wi-Fi yang mandiri sebagai jembatan dari mikrokontroler yang ada ke jaringan Wi-Fi. ESP32 menggunakan prosesor *dual core*. (Kusumah & Pradana, 2019). ESP32 digunakan sebagai *controller* pada sistem yang dibangun untuk mengolah data *input* dari sensor dan memberikan instruksi terhadap perangkat *output* sesuai dengan kondisi yang dijabarkan oleh perangkat *input* (Anggraini et al., 2023). Sistem pengoordinasian dan pengendali utama yaitu: sebagai tempat pengolahan data pada subsistem akuisisi data, mempercepat dan mempermudah dalam membuat sistem control, pengiriman data yang sudah diolah oleh akuisisi data ke data penyimpanan *online* melalui Wi-Fi dengan koneksi internet. Fungsi dari ESP32 yaitu: merubah sinyal hasil pembacaan sensor yang masih analog dan dikonversi menjadi digital, memberi perintah modul-modul untuk melakukan aksi terhadap input mikrokontroler, mengirimkan data melalui jaringan Wi-Fi



Gambar 2. 5. ESP32.
(sumber: www.iotkece.com)

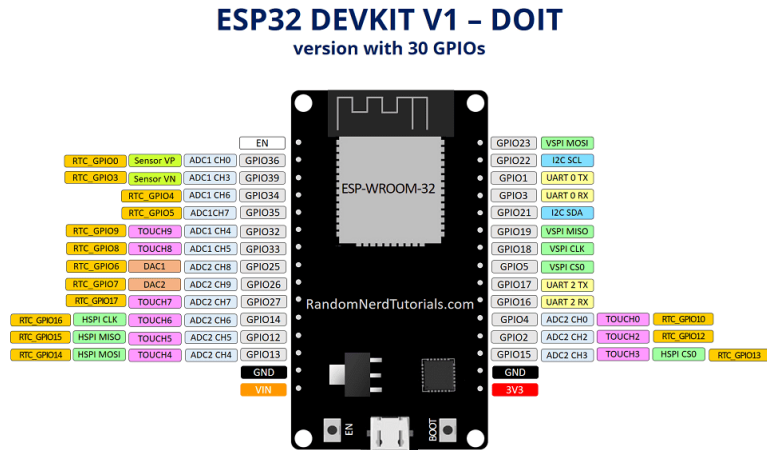
Spesifikasi ESP32 ditampilkan pada tabel 2.4.:

Tabel 2. 4. Spesifikasi ESP32.

No	Atribut	Detail
1	Tegangan	3.3 Volt
2	Prosesor	Tensilica L108 32 bit
3	Kecepatan prosesor	Dual 160MHz
4	RAM	520K
5	GPIO	34
6	ADC	7
7	Dukungan 802.11	11b/g/n/e/i
8	Bluetooth	BLE (Bluetooth Low Energy)
9	SPI	3
10	I2C	2
11	UART	3

(sumber: Kusumah & Pradana, 2019)

Jika dilihat dari spesifikasi pada tabel maka mikrokontroler ESP32 dapat dijadikan pilihan untuk digunakan pada alat peraga *interface* mikrokontroler karena mikrokontroler ini memiliki *interface* yang lengkap, juga memiliki Wi-Fi yang sudah tertanam pada mikrokontroler sehingga tepat untuk digunakan pada alat peraga atau *trainer IoT* (Kusumah & Pradana, 2019). Skema pin out pada mikrokontroler ESP32 terdapat pada gambar 2.6.



Gambar 2. 6. Skema Rangkaian Pin Out ESP32.

(sumber: www.iotkece.com)

4. Saklar

Saklar 3 kaki digunakan sebagai penghubung dan pemutus arus serta mengatur putaran kanan dan kiri pada motor listrik. Dengan spesifikasi 6A 250 VAC dan 10 A 125 VAC (Syaefudin et al., 2022).

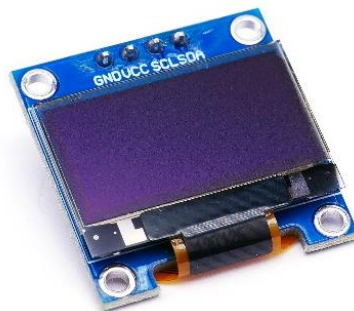


Gambar 2. 7. Saklar 3 kaki.

(sumber: www.aliexpress.com)

5. *Oled*

Oled display disebut juga *Organic LED* adalah *display* grafik dengan ukuran 0.96 inci dan resolusi 128x64 *pixel* menggunakan teknologi *Oled display*. *Oled* biasanya terbuat dari karbon dan hidrogen. Untuk komunikasi dengan mikrokontroler arduino menggunakan komunikasi I2C, menggunakan 2 pin yaitu pin sda dan pin scl sehingga menghemat pin (Saniman et al., 2020).



Gambar 2. 8. *Oled*.
(sumber : www.reddit.com)

6. *Baterai*

Baterai adalah sebuah alat yang dapat merubah energi kimia yang disimpannya menjadi energi listrik yang dapat digunakan oleh suatu perangkat elektronik(Sufri & Aswardi, 2020). Tipe baterai yang digunakan pada perancangan alat pendeteksi detak jantung pada penelitian ini menggunakan tipe baterai litium 18650 yang dirancang secara paralel. Kelebihan dari baterai Lithium Ion, tegangan (voltase) baterai Lithium 18650 mencapai 3,6-3,7V DC dan memiliki kapasitas maksimal sebesar 3600mAH. Pengisian baterai Lithium dengan charger dapat menerima voltase 4,2V DC. *Battery rechargeble* (baterai dapat diisi ulang).



Gambar 2. 9. Baterai Lithium Ion 18650.

7. *Blynk*

Blynk merupakan sebuah *platform* aplikasi di *iOS* dan Android untuk mengendalikan berbagai mikrokontroler seperti Arduino, Raspberry Pi, dan sejenisnya melalui internet. *Blynk* juga merupakan sebuah dasbor digital dimana dapat membangun sebuah antarmuka grafis untuk sebuah alat yang dibuat hanya dengan “*drag and drop*” *widget*. *Blynk* dibuat untuk menunjang pengembangan *Internet of Things* (IoT) yang mampu mengendalikan dari jarak jauh, *blynk* dapat menampilkan data sensor, menyimpan data, dan memvisualisasikannya, serta masih banyak hal lainnya yang dapat dilakukan oleh *blynk* (Kusumah & Pradana, 2019).

Ada tiga komponen utama dalam *platform* ini, yaitu *Blynk App*, *Blynk Server*, dan *Blynk Library*. *Blynk App* adalah aplikasi *Blynk* yang ada pada platform Android atau *iOS* di *smartphone* untuk dapat membuat sebuah antarmuka dari alat yang dibuat dengan menggunakan berbagai *widget* yang disediakan. Lalu *Blynk Server* adalah sebuah *server Blynk* yang bertanggung jawab atas semua komunikasi antara aplikasi *Blynk* pada *smartphone* dengan *hardware* atau alat yang dibuat. Sedangkan *Blynk Library* adalah sebuah *library* yang memberi kemampuan *hardware* atau alat yang dibuat untuk dapat berkomunikasi dengan *Blynk server* dan memproses semua data dari *input output*.

BAB III

METODOLOGI PENELITIAN

A. Metode Penelitian

Penelitian ini menggunakan metode *research and development* atau penelitian dan pengembangan, yaitu metode yang digunakan untuk membuat suatu produk dan menguji keefektifan dari hasil produk tersebut. Untuk mengetahui kesesuaian alat dan keakuratan alat dilakukan perbandingan dengan alat pembanding. Dapat dilakukan perhitungan yang dituliskan secara sistematis (Puspasari et al., 2020).

Tujuan perancangan alat ini adalah untuk membuat sebuah alat yang dapat mendeteksi detak jantung manusia dengan menempelkan *pad* atau elektroda ke bagian tubuh pasien. Penelitian ini dilakukan melalui beberapa tahap yaitu perancangan alat, pengujian alat, dan menganalisis hasil untuk ditarik kesimpulan. Nilai BPM dapat diperoleh dengan menghitung 60.000 dibagi dengan waktu BPM (Firman Maulana, 2021).

$$\text{BPM} = \frac{60.000}{\text{ms BPM}}$$

Keterangan :

BPM = *Beat Per Minutes*

ms BPM = waktu yang diperoleh untuk mencapai puncak (ms)

Hasil nilai yang didapat, dapat digunakan untuk menghitung nilai BPM yang tertera pada rumus di atas. Untuk nilai persentase *error* adalah (Puspasari et al., 2020):

1. Rata-rata

$$x = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n}$$

2. Nilai *Error*

$$Error = \frac{\text{nilai uji} - \text{nilai standar}}{\text{nilai uji}} \times 100\%$$

Sumber dari MK PK Alat Kesehatan yang diterbitkan oleh Direktorat Jenderal Pelayanan Kesehatan pada tahun 2018 menyatakan bahwa nilai *error* tidak boleh lebih dari $\pm 5\%$.

B. Alat dan Bahan Penelitian

Pada penelitian pengembangan alat Pendeteksi Detak Jantung Berbasis IoT dibutuhkan alat dan bahan yang diperlukan antara lain:

Tabel 3. 1. Alat dan Bahan.

No.	Perangkat Keras	Perangkat Lunak
1	Sensor AD8232	Arduino IDE
2	ESP32	Fritzing
3	Saklar	Blynk
4	Kabel <i>Jumper</i>	Ms. Word
5	<i>Box</i>	Ms. Excel
6	<i>Stick Glue</i>	
7	<i>Project Board</i>	
8	Tenol	
9	<i>Oled Display</i>	
10	Bolpoin	
11	Penggaris	
12	Solder	
13	Laptop	
14	Baterai 18650	
15	Kikir	
16	Amplas	
17	Modul <i>Powerbank</i>	
18	Modul <i>Step Up</i>	

C. Tempat dan Waktu Penelitian

Penelitian ini dilakukan:

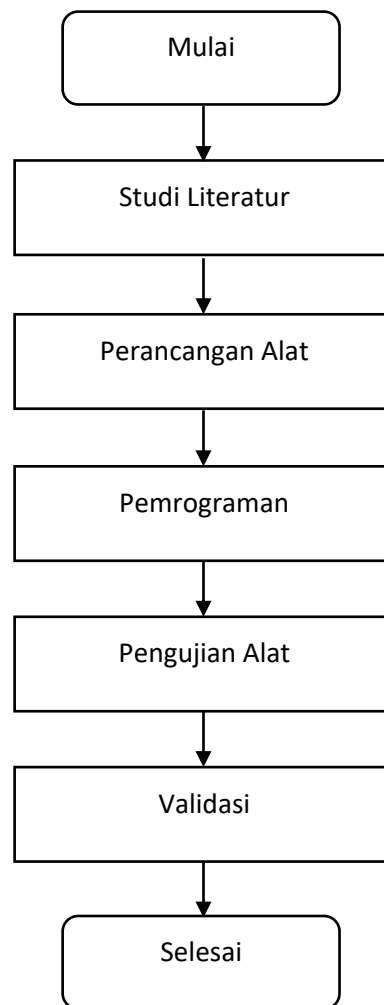
1. Tempat penelitian:
Di Boyolali dan Surakarta
2. Tempat pengujian alat:
Di Boyolali

Tabel 3. 2. Jadwal Penelitian.

No	Kegiatan	Waktu Penelitian											
		2022			2023								
		10	11	12	1	2	3	4	5	6	7	8	9
1	Perencanaan												
2	Pembuatan Proposal												
3	Seminar Proposal												
4	Pembuatan Alat												
5	Pembuatan Skripsi												
6	Seminar Hasil												
7	Revisi												
8	Pengumpulan Buku												

D. Teknik Analisis Data

Pada teknik analisis data dibagi menjadi beberapa tahap. Adapun tahapan tersebut akan menggambarkan dalam bentuk diagram blok seperti pada gambar 3.1. berikut:



Gambar 3. 1. Metode Perancangan.

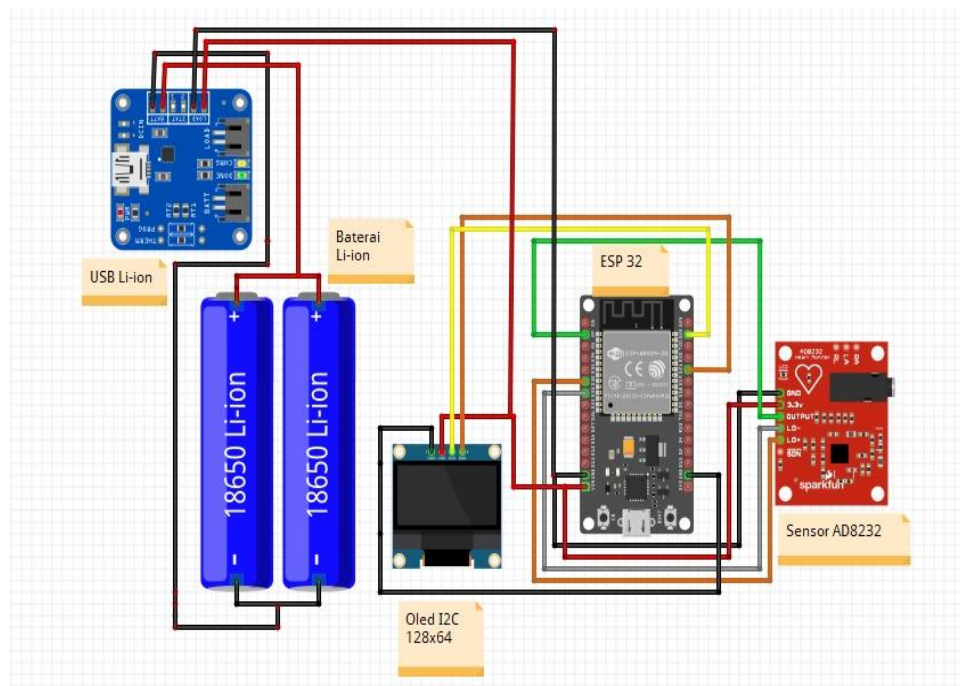
Kemudian untuk penjabaran mengenai masing-masing tahap pada blok diagram pada gambar 3.1. akan dijelaskan sebagai berikut:

1. Studi Literatur

Studi literatur dilakukan dengan cara mendapatkan dan mengumpulkan data dengan membaca buku-buku, jurnal, dan artikel yang berkaitan dengan permasalahan yang dibahas dalam penelitian ini.

2. Perancangan Alat

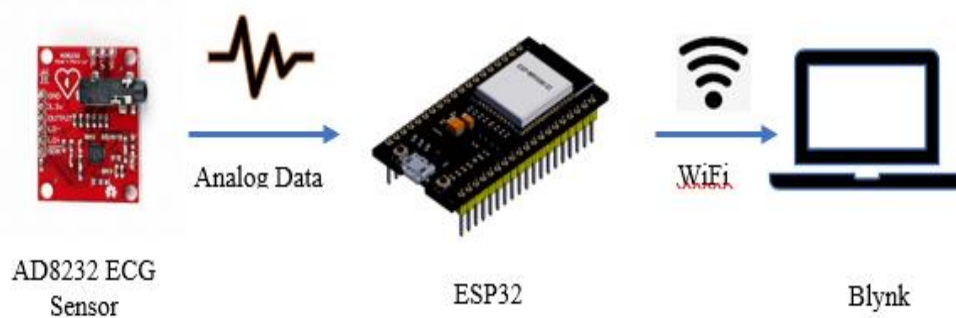
Perancangan alat yang dilakukan bertujuan untuk mencari bentuk rangkaian model alat yang akan dibuat dengan mempertimbangkan berbagai faktor permasalahan dan kebutuhan yang sudah ditentukan. Langkah selanjutnya adalah menyediakan alat dan bahan yang digunakan untuk membuat alat Pendeteksi Detak Jantung Berbasis IoT.



Gambar 3. 2. Skema Rangkaian Alat.

3. Pemrograman

Pembuatan program dilakukan menggunakan *Software* Arduino IDE, *software* ini berguna untuk membuat, mengedit, mengatur, meng-*coding* serta meng-*upload* program ke *board* Arduino agar dapat melakukan pembacaan sensor AD8232, dapat menampilkan hasil dari pengukuran detak jantung.



Gambar 3. 3. Perancangan Sistem Alat.

Pada gambar 3.3. menunjukkan sistem monitoring pendeteksi detak jantung dengan menggunakan ESP32 yang bertujuan untuk mentransmisikan data menuju *Blynk*. Data detak jantung yang diperoleh dari sensor AD8232 yang berupa sinyal analog akan diubah menjadi sinyal digital pada mikrokontroler ESP32. Pada ESP32 akan dilakukan proses penghitungan detak jantung dengan satuan *beat per minutes* (bpm), klasifikasi detak jantung dan juga enkripsi data untuk dikirimkan ke *Blynk* melalui Wi-Fi.

Pengiriman data dari ESP32 menuju *Blynk* menggunakan kode perintah pada Arduino IDE sebagai berikut:

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h> //library blynk
BlynkTimer timer;
void setup (){
  //koneksi wifi
  WiFi.begin(ssid,pass);
```

```

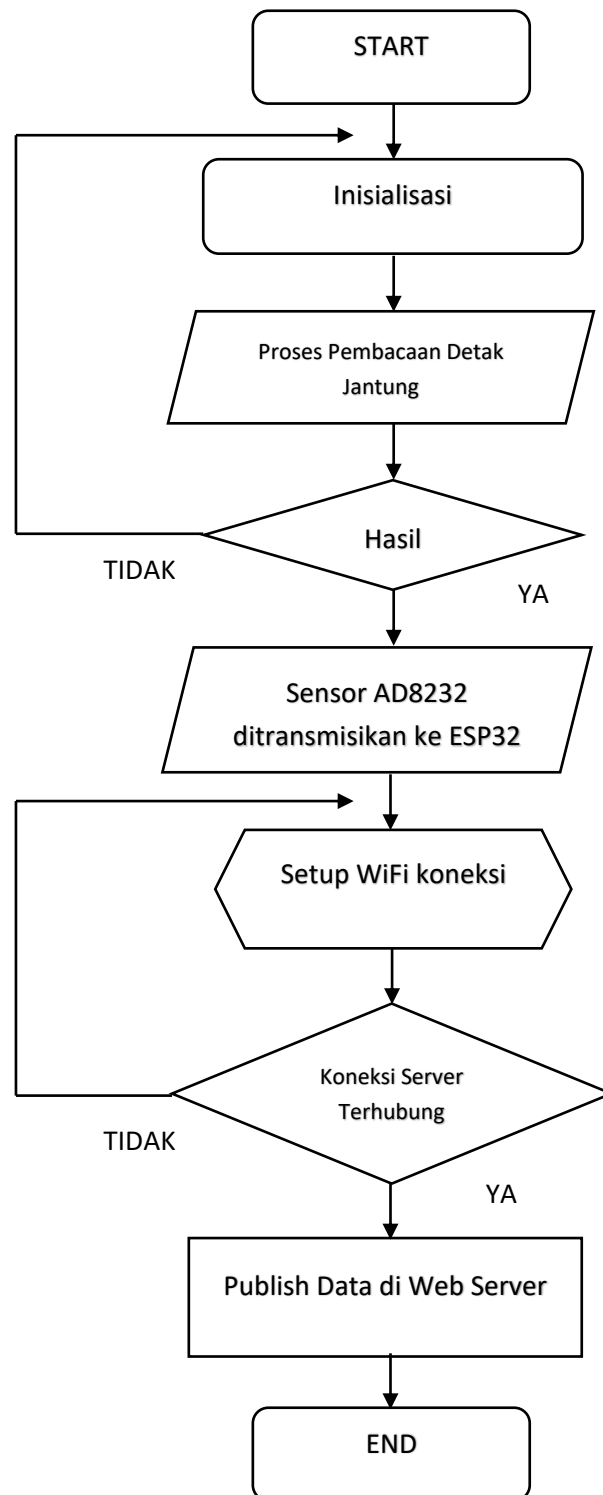
// Inicializa Blynk
Blynk.begin(auth, ssid, pass);
//Inialisasi pin untuk tampilan ke blynk
Blynk.virtualWrite(V0, "Persiapan...");
timer.setInterval(1000,updateDisplay);
timer.setInterval(1000,sendData); }
void loop(){
  Blynk.run();
  timer.run();
  //kirim data ke widget
  Blynk.virtualWrite(V1, mavBPM);
  Blynk.virtualWrite(V5,heartRateValue); }

```

4. Pengujian Alat

Pengujian dilakukan untuk mengetahui apakah alat berfungsi sesuai dengan desain yang telah direncanakan, mulai dari pengukuran melalui sensor sampai pada penampilan data serta hasil yang diberikan oleh alat pendeteksi detak jantung. Pengujian alat ini dilakukan dengan cara melakukan pengambilan data menggunakan *Pulse Oximetry*.

Kemudian untuk mengetahui jalannya sistem pada alat yang akan dibuat, maka akan dijelaskan dalam bentuk *flowchart* sistem seperti pada gambar 3.4. berikut:



Gambar 3. 4. Flowchart Sistem.

5. Validasi

Pengujian pada validasi dilakukan dengan menggunakan alat yang sudah ada dan berkaitan dengan alat pada penelitian ini, yaitu *Pulse Oximetry*. Pada pengujian ini bertujuan untuk membandingkan *output* dari masing-masing alat yaitu alat *Pulse Oximetry* dan alat pendeteksi detak jantung. Kemudian dari hasil yang didapatkan akan dianalisis untuk mendapatkan berapa tingkat *error* yang dihasilkan. Nilai *error* yang dihasilkan dapat menjadi tolak ukur sebagai kelayakan dari pengoperasian alat.

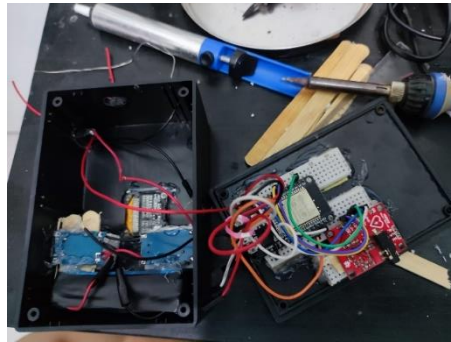
BAB IV

HASIL DAN PEMBAHASAN

A. Hasil Pembuatan Alat

1. Alat

a. Perakitan Alat



Gambar 4. 1. Perakitan Alat.

b. Hasil Akhir Alat



Gambar 4. 2. Hasil Akhir Alat.

2. Spesifikasi Alat

- a. Tegangan : 5 V
- b. 3 channel : merah (RA), kuning (LA), hijau (RL)
- c. Dimensi : 12,5x8,5x5 cm
- d. Berat : 0,75g
- e. Power Supply : DC
- f. Pengukuran : *Heart Rate* langsung dikonversi ke bpm
- g. *Natural Light-Emitting Diode (OLED)* 128x64 pixel

- h. *One-Touch Operation (monitoring)*
- i. *Battery recharge with Type C*
- j. *Monitoring via Blynk*

3. Langkah Penggunaan Alat

- a. Pastikan baterai sudah terisi penuh.
- b. Pastikan jaringan WiFi di dalam jangkauan dan stabil.
- c. Tekan tombol saklar pada alat.
- d. Colokkan kabel elektroda ke alat.
- e. Gunakan elektroda yang baru.
- f. Letakkan elektroda pada tubuh pasien, dengan ketentuan warna merah diletakkan di dada kanan, kuning di dada kiri, dan warna hijau di bawah pusar bagian kanan.
- g. Buka aplikasi *Blynk* di *smartphone* pengguna.
- h. Kemudian pilih Menu=> “Heart Rate bpm”
- i. Alat siap digunakan.

B. Pengujian Alat

Pengujian alat dilakukan di Boyolali pada bulan Juli 2023 dengan mengambil sebanyak 6 orang untuk dilakukan masing-masing orang 5 kali percobaan dan didapat data 30 kali percobaan, dengan rincian sebagai berikut:

a. Responden Pertama

Tabel 4. 1. Data Pengujian Responden Pertama.

Responden	Usia	Klaster	Pulse Oximetry (bpm)	Alat (bpm)	Blynk (bpm)	Error
Zahfa	8 thn	1	110	107	107	3%
		2	111	107	107	4%
		3	109	108	108	1%
		4	108	109	109	1%
		5	109	110	110	1%
Rata-rata	109	108	108	1%		

Hasil dari responden pertama didapatkan nilai rata-rata *error* 1%

b. Responden Kedua

Tabel 4. 2. Data Pengujian Responden Kedua.

Responden	Usia	Klaster	Pulse Oximetry (bpm)	Alat (bpm)	Blynk (bpm)	Error
Elya	16 thn	6	93	93	93	0%
		7	94	93	93	1%
		8	94	93	93	1%
		9	95	93	93	2%
		10	96	94	94	2%
Rata-rata	94	93	93	1%		

Hasil dari responden kedua didapatkan nilai rata-rata *error* 1%

c. Responden Ketiga

Tabel 4. 3. Data Pengujian Responden Ketiga.

Responden	Usia	Klaster	Pulse Oximetry (bpm)	Alat (bpm)	Blynk (bpm)	Error	—
Sukiman	53	11	80	83	83	4%	—
		12	80	84	84	5%	—
		13	81	84	84	4%	—
		14	83	84	84	1%	—
		15	80	84	84	5%	—
Rata-rata	81	84	84	4%			

Hasil dari responden ketiga didapatkan nilai rata-rata *error* 4%

d. Responden Keempat

Tabel 4. 4. Data Pengujian Responden Keempat.

Responden	Usia	Klaster	Pulse Oximetry (bpm)	Alat (bpm)	Blynk (bpm)	Error	—
Sri Lestari	54	16	72	73	73	1%	—
		17	71	73	73	3%	—
		18	72	73	73	1%	—
		19	72	75	75	4%	—
		20	74	73	73	1%	—
Rata-rata	72	73	73	2%			

Hasil dari responden keempat didapatkan nilai rata-rata *error* 2%

e. Responden Kelima

Tabel 4. 5. Data Pengujian Responden Kelima.

Responden	Usia	Klaster	Pulse Oximetry (bpm)	Alat (bpm)	Blynk (bpm)	Error
Mariyem	70	21	72	74	74	3%
		22	72	71	71	1%
		23	70	70	70	0%
		24	71	68	68	4%
		25	72	69	69	4%
Rata-rata	71	70	70	1%		

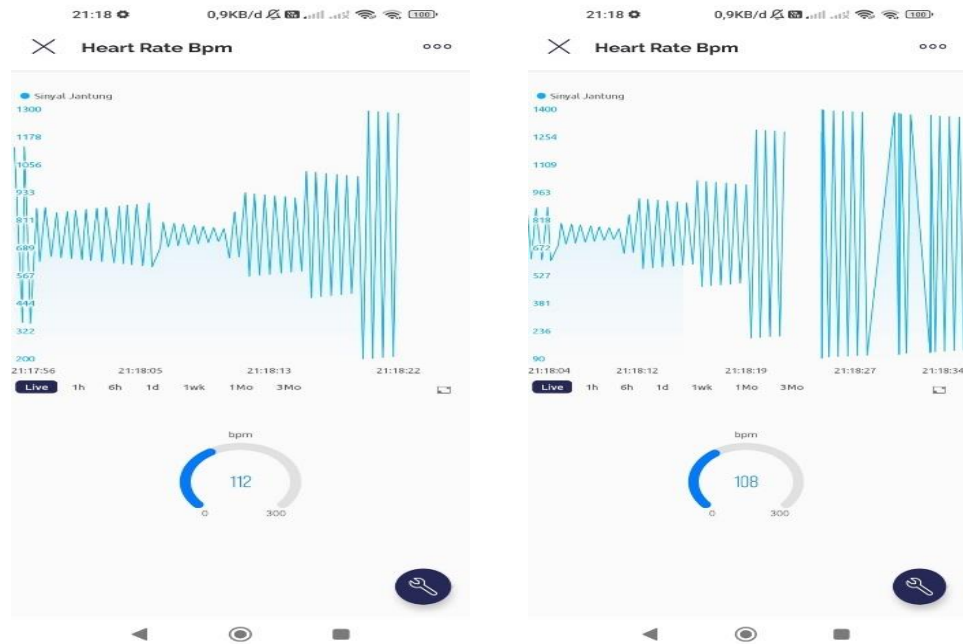
Hasil dari responden kelima didapatkan nilai rata-rata *error* 1%

f. Responden Keenam

Tabel 4. 6. Data Pengujian Responden Keenam.

Responden	Usia	Klaster	Pulse Oximetry (bpm)	Alat (bpm)	Blynk (bpm)	Error
Wakiyo	77	26	72	69	69	4%
		27	69	69	69	0%
		28	69	70	70	1%
		29	71	70	70	1%
		30	70	69	69	1%
Rata-rata	70	69	69	1%		

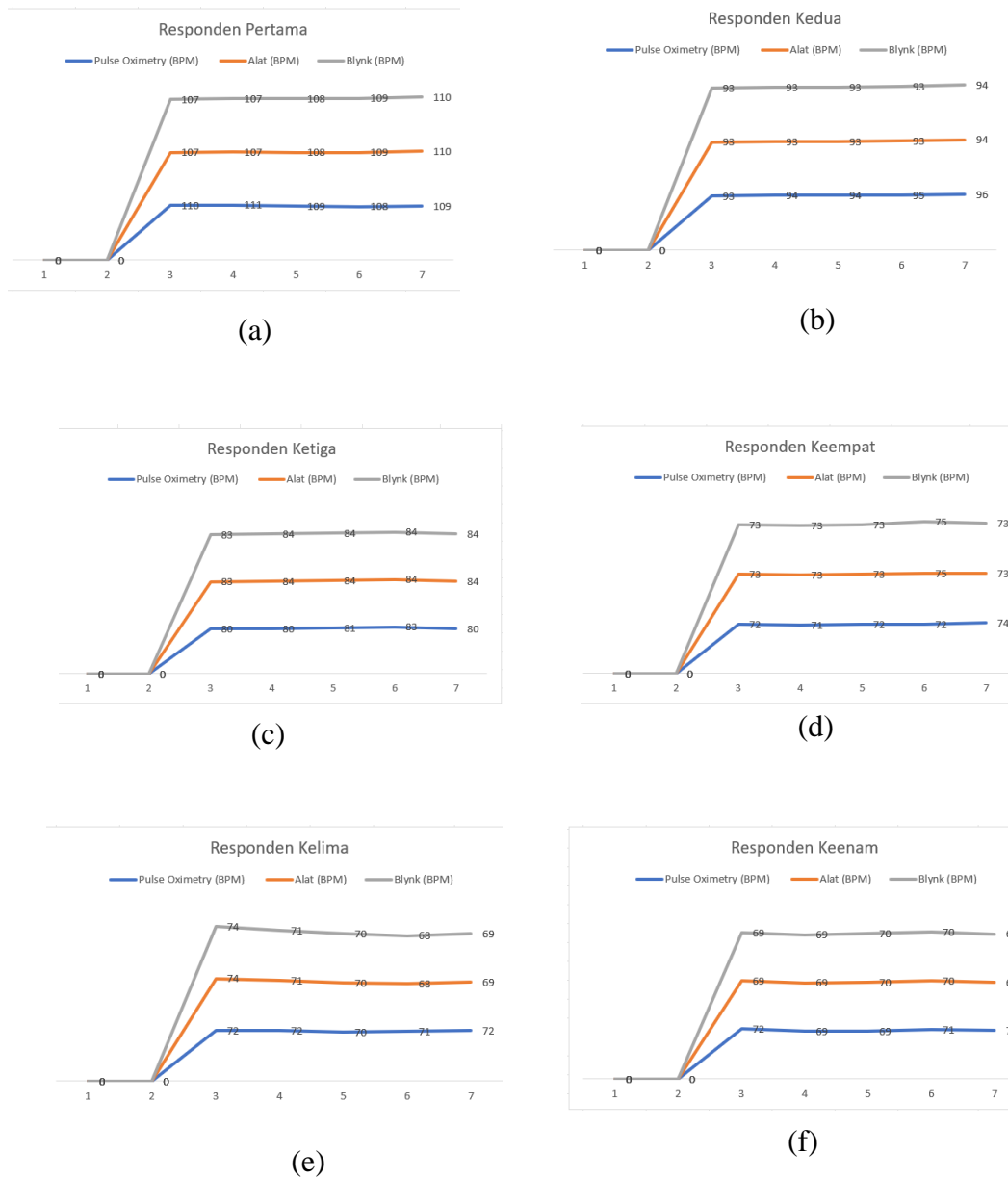
Hasil dari responden keenam didapatkan nilai rata-rata *error* 1%



Gambar 4. 3. Tampilan Pengukuran Detak Jantung di *Blynk*.

C. Pembahasan

Berdasarkan pengujian dari 6 responden menunjukkan presentase *error* dari alat pendeteksi detak jantung. Adanya nilai *error* yang masih muncul bisa dianalisis dari kurang rekatnya pad elektroda yang menempel pada tubuh responden. Pada responden pria usia 0-10 tahun menunjukkan hasil pengukuran bpm yang normal berkisar antara 100-130 bpm. Pada responden wanita usia 10-17 tahun menunjukkan hasil pengukuran bpm yang normal berkisar 83-100 bpm dan responden usia >18 tahun dimana responden pria berkisar 60-90 bpm, serta responden wanita berkisar 65-90 bpm. Dari hasil pengujian alat diubah menjadi nilai rata-rata ke dalam bentuk grafik berikut:



Gambar 4. 4. Grafik Hasil Pengukuran: (a)Pertama, (b)Kedua, (c)Ketiga, (d)Keempat, (e)Kelima, (f)Keenam.

Perhitungan konversi dari sinyal grafik detak jantung ke satuan bpm didapat dari (Firman Maulana, 2021):

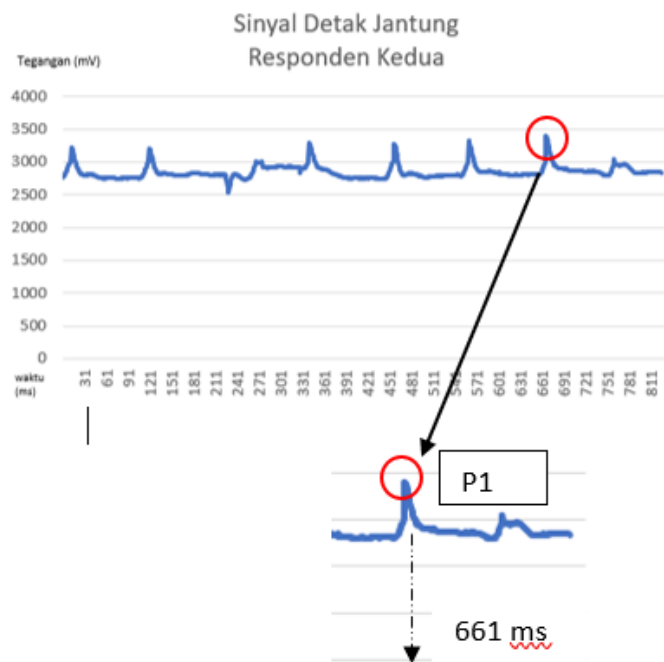
$$BPM = \frac{60.000}{ms \text{ BPM}}$$

Keterangan:

BPM = Beat Per Minutes

ms BPM = waktu yang diperoleh untuk mencapai puncak (ms)

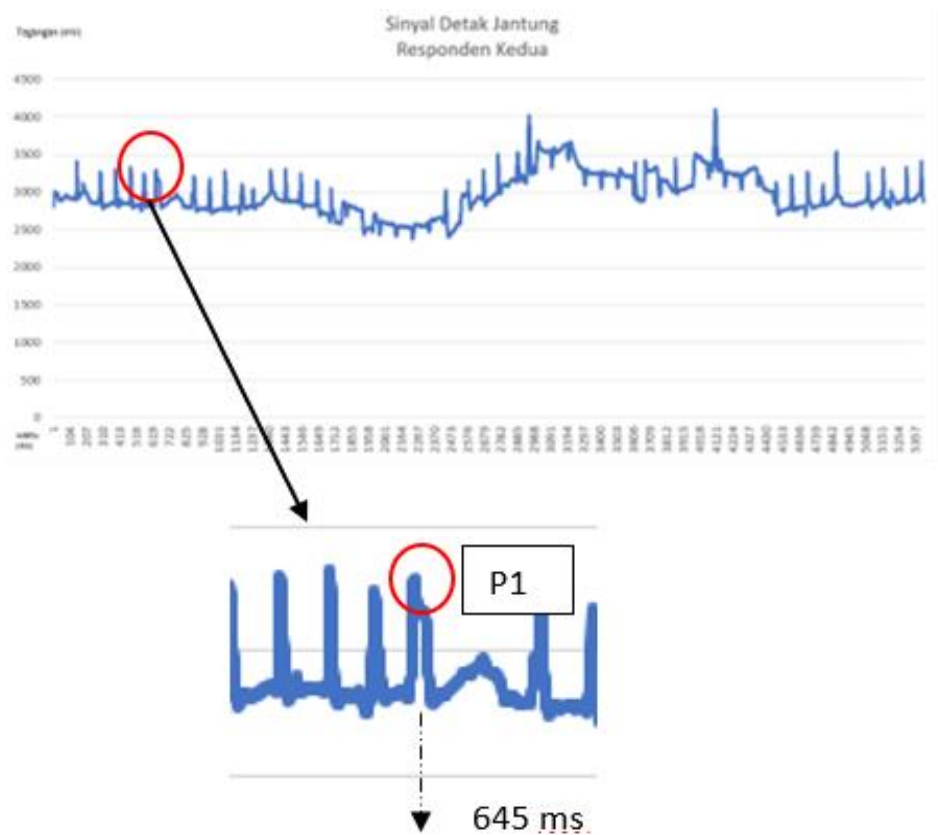
Pengambilan sampel pengukuran terhadap dua responden dengan pengujian secara berkala, dengan data sebagai berikut:



Gambar 4. 5. Sinyal Detak Jantung Pengukuran Pertama.

Perhitungan bpm pada gambar 4.5.:

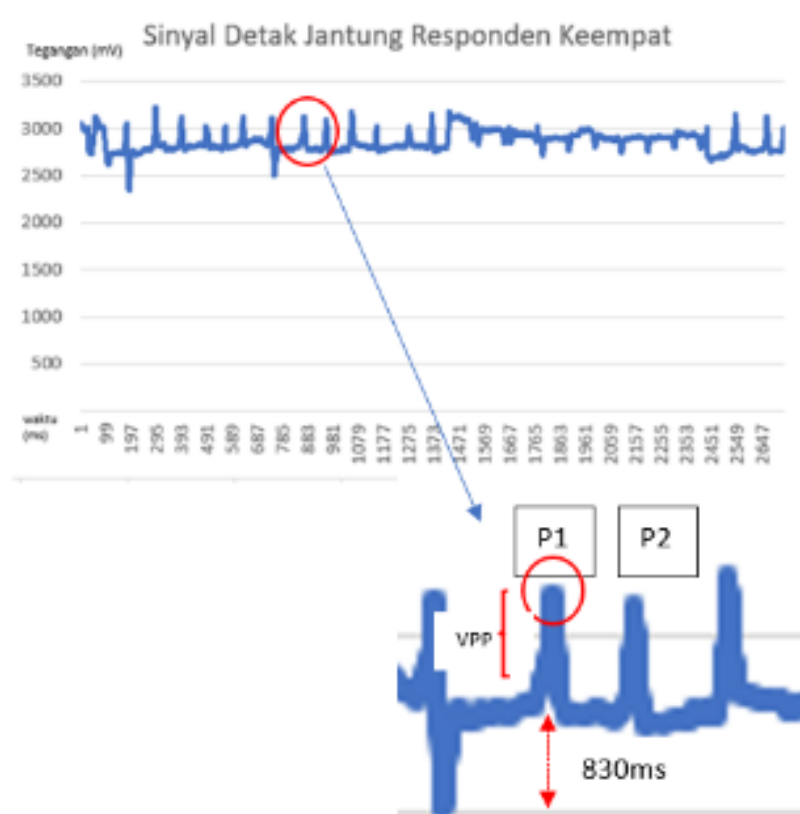
$$\text{BPM} = \frac{60.000}{661 \text{ ms}} = 90,09 \text{ bpm}$$



Gambar 4. 6. Sinyal Detak Jantung Pengukuran Kedua.

Perhitungan bpm pada gambar 4.6.:

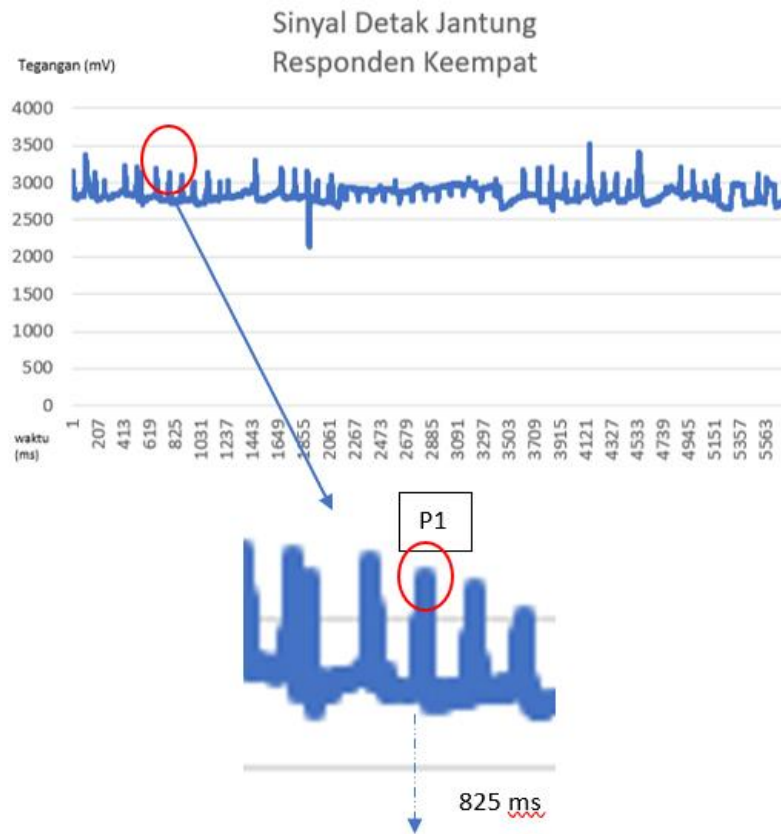
$$\text{BPM} = \frac{60.000}{645 \text{ ms}} = 93,02 \text{ bpm}$$



Gambar 4. 7. Sinyal Detak Jantung Pengukuran Ketiga.

Perhitungan bpm pada gambar 4.7.:

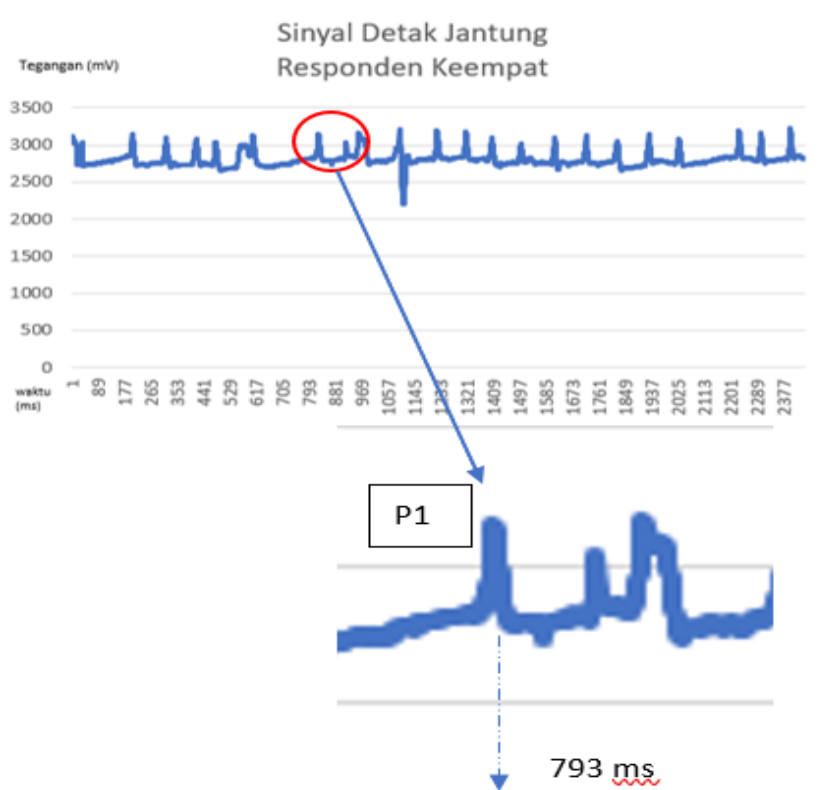
$$\text{BPM} = \frac{60.000}{830 \text{ ms}} = 72,3 \text{ bpm}$$



Gambar 4. 8. Sinyal Detak Jantung Pengukuran Keempat.

Perhitungan bpm pada gambar 4.8.:

$$\text{BPM} = \frac{60.000}{825 \text{ ms}} = 72,7 \text{ bpm}$$



Gambar 4. 9. Sinyal Detak Jantung Pengukuran Kelima.

Perhitungan bpm pada gambar 4.9.:

$$\text{BPM} = \frac{60.000}{793 \text{ ms}} = 75,6 \text{ bpm}$$

Pengolahan sinyal didapat dari acuan ms pada tabel 2.3., - Hasil dari beberapa pengukuran didapati perbedaan dari perhitungan secara manual (dengan pembacaan sinyal) dan perhitungan instan (langsung dikonversi ke satuan bpm) dengan nilai *error* yang sudah dihitung pada Tabel 4.1. sampai dengan Tabel 4.6., sedangkan pada perhitungan sinyal detak jantung dihitung nilai *error* antara *Pulse Oximetry* dan perhitungan sinyal secara manual sebagai berikut:

$$\text{Error} = \frac{\text{nilai uji} - \text{nilai standar}}{\text{nilai uji}} \times 100\%$$

$$Error_{P1} = \frac{90,09-94}{90,09} \times 100\% = 4,3\%$$

$$Error_{P2} = \frac{93,02-94}{92,02} \times 100\% = 1,05\%$$

$$Error_{P3} = \frac{72,3-72}{72,3} \times 100\% = 0,4\%$$

$$Error_{P4} = \frac{72,7-72}{72,7} \times 100\% = 0,96\%$$

$$Error_{P5} = \frac{75,6-72}{75,6} \times 100\% = 4,7\%$$

Pengukuran alat menghasilkan nilai *error* dengan rumus rata-rata:

$$x = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n}$$

$$x = \frac{4,3 + 1,05 + 0,4 + 0,96 + 4,7}{5}$$

$$x = 2,3 \%$$

Perhitungan pada nilai *error* rata-rata saat pengukuran langsung dari keenam responden dengan konversi BPM yaitu:

$$x = \frac{1 + 1 + 4 + 2 + 1 + 1}{6}$$

$$x = 1,6 \%$$

Berdasarkan data perhitungan di atas nilai *error* yang didapat saat nilai *error* yang didapat saat perhitungan manual dengan sampel 2 responden dan diambil 5 kali pengukuran yaitu 2,3 % dan pengukuran langsung terhadap keenam responden pada konversi BPM yaitu 1,6 % hasil ini menunjukkan nilai *error* alat tidak lebih dari 5% dan menunjukkan bahwa alat layak digunakan.

D. Keterbatasan

Penelitian ini menemukan bahwa keterbatasan yang dilakukan adalah menampilkan sinyal detak jantung dengan ukuran yang besar pada layar dan adanya *noise* saat sinyal detak jantung tampil pada aplikasi *Blynk*.

BAB V

PENUTUP

A. Simpulan

Setelah dilakukan pengujian dan analisis terhadap alat pendeteksi detak jantung 1 *Lead* berbasis IoT, dapat ditarik kesimpulan sebagai berikut:

1. Perancangan alat untuk pendeteksi detak jantung dapat diterapkan dengan baik dan bekerja sesuai fungsinya.
2. Cara kerja monitoring detak jantung diawali dengan pengambilan data oleh sensor AD8232 yang kemudian data akan diolah dengan mikrokontroler ESP32 dan mengirimkan hasilnya ke aplikasi *Blynk*.
3. Tingkat *error* sebagai validasi yang dihasilkan oleh alat relatif rendah jika dibandingkan dengan *Pulse Oximetry*.

B. Saran

Berdasarkan pengujian dan analisis pada alat pendeteksi detak jantung berbasis IoT mempunyai beberapa saran dalam pengembangan alat ini kedepannya yaitu: menggunakan layar dengan ukuran yang lebih besar, menambahkan analisis sinyal yang terbaca dengan menggunakan *spektrum analyzer*, menambahkan *filter* untuk mengurangi *noise* dari sinyal detak jantung yang dihasilkan, dan menambahkan *delay* pada saat pengiriman hasil ke aplikasi *Blynk*.

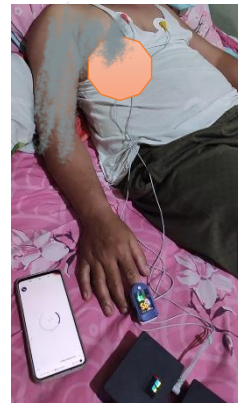
DAFTAR PUSTAKA

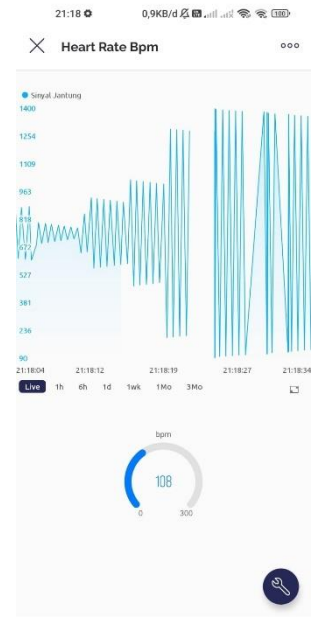
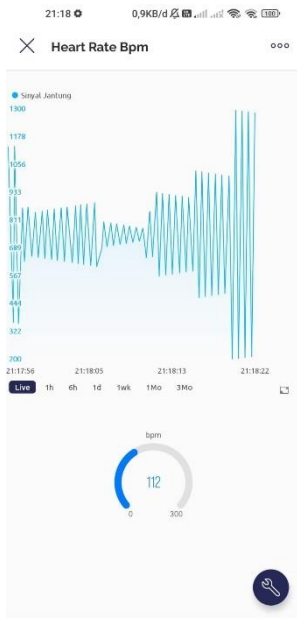
- Agustian, I. (2019). Rancang Bangun Pemantau Detak Jantung dan Suhu Tubuh Portabel Dengan Sistem IoT. *Jurnal Amplifier : Jurnal Ilmiah Bidang Teknik Elektro Dan Komputer*, 9(2), 14–18. <https://doi.org/10.33369/jamplifier.v9i2.15378>
- Analog Devices. (2013). AD8232 Single-Lead ECG. *Data Sheet*, 1–28. www.analog.com/AD8232
- Anggraini, I., B, N. I. H., Ramadhani, I. P., & Hadis, M. S. (2023). 2816 Words Crossref Posted Content database Crossref database 9 % Submitted Works database Excluded from Similarity Report Internet database Quoted material Sistem Akses Pintu Berbasis Face Recognition Menggunakan ESP32 Module dan Aplikasi Telegram.
- FM., R. (2017). Implementasi Sistem Telemedicine Untuk Monitoring Detak Jantung Berbasis Sensor Ad8232. *Sinar Fe* 7, 317–320.
- Hermansyah, A., Hardiyanti, R., & Prasetyo, A. P. P. (2022). Sistem Perekam Detak Jantung Berbasis Internet Of Things (IoT) dengan Menggunakan Pulse Heart Rate Sensor. *JTEV (Jurnal Teknik Elektro Dan Vokasional)*, 8(2), 338. <https://doi.org/10.24036/jtev.v8i2.116677>
- Idu, C. J., Tamaela, J. M., & Wicaksana, A. L. (2021). Pemanfaatan Teknologi Dalam Mengurangi Readmission Pada Pasien Dengan Gagal Jantung: Systematic Review. *Jurnal Kesehatan*, 14(1), 48–58. <https://doi.org/10.23917/jk.v14i1.11442>
- Isma, T. W., Yuliza, M., Angraini, T., & Susanti, R. (2020). Efektifitas Sensor Elektrokardiograf (EKG) AD8232 Untuk Mendeteksi Kelelahan Pada Saat Penggunaan Smartphone. *Elektron : Jurnal Ilmiah*, 12(1), 7–11. <https://doi.org/10.30630/eji.12.1.148>

- Kukuh Setyohadi, Ibrahim, R. R. (2021). Jurnal Ilmiah Wahana Pendidikan. *Jurnal Ilmiah Wahana Pendidikan* <https://Jurnal.Unibrah.Ac.Id/Index.Php/JIWP>, 7(1). <http://jurnalmahasiswa.unesa.ac.id/index.php/jurnal-penelitian-pgsd/article/view/23921>
- Kusumah, H., & Pradana, R. A. (2019). Penerapan Trainer Interfacing Mikrokontroler Dan Internet of Things Berbasis Esp32 Pada Mata Kuliah Interfacing. *Journal CERITA*, 5(2), 120–134. <https://doi.org/10.33050/cerita.v5i2.237>
- Momin, A., Hartono, H., & Aziz, A. N. (2021). Rancang Bangun Elektrokardiograf Berbasis IoT. *Jurnal Fisika*, 11(2), 60–76. <https://journal.unnes.ac.id/nju/index.php/jf/article/view/31950%0Ahttps://journal.unnes.ac.id/nju/index.php/jf/article/viewFile/31950/12550>
- Novani, N. P., Arief, L., & Anjasmara, R. (2019). Analisa Detak Jantung dengan Metode Heart Rate Variability (HRV) untuk Pengenalan Stres Mental Berbasis Photoplethysmograph (PPG). *JITCE (Journal of Information Technology and Computer Engineering)*, 3(02), 90–95. <https://doi.org/10.25077/jitce.3.02.90-95.2019>
- Ria Hariri, R. H., Lutfi Hakim, L. H., & Riska Fita Lestari, R. F. L. (2020). Sistem Monitoring Detak Jantung Menggunakan Sensor AD8232. *Journal Zetroem*, 2(2), 1–5. <https://doi.org/10.36526/ztr.v2i2.1017>
- Saniman, S., Ramadhan, M., & Zulkarnain, I. (2020). Rancang Bangun Smart Glass Telemetry Tegangan Menggunakan Teknik Simplex Berbasis Arduino Nano. *J-SISKO TECH (Jurnal Teknologi Sistem Informasi Dan Sistem Komputer TGD)*, 3(1), 12. <https://doi.org/10.53513/jsk.v3i1.191>
- Sufri, S., & Aswardi, A. (2020). Alat Pendeteksi Detak Jantung dan Kesehatan Berbasis Arduino. *JTEIN: Jurnal Teknik Elektro Indonesia*, 1(2), 69–75. <https://doi.org/10.24036/jtein.v1i2.31>

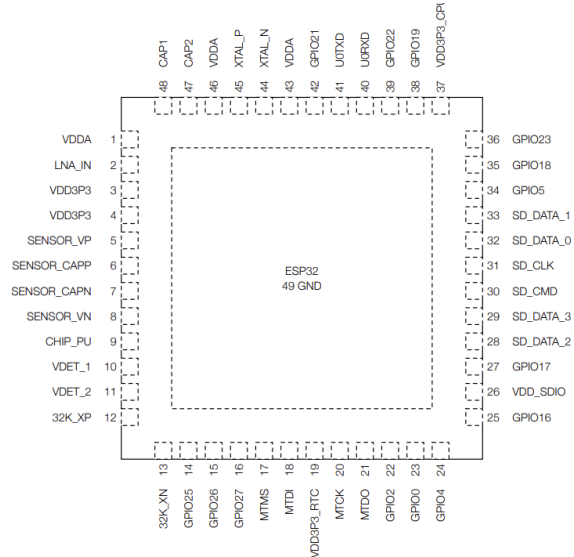
Syaefudin, E. A., Jumhur, A. A., & Wulandari, D. A. (2022). *ALAT BANTU SENAI PORTABEL Required Power Analysis in The Design and Fabrication of Portable Senai Equipment*. 7, 77–86.

LAMPIRAN 1: Dokumentasi





LAMPIRAN 2: DATA SHEET ESP32

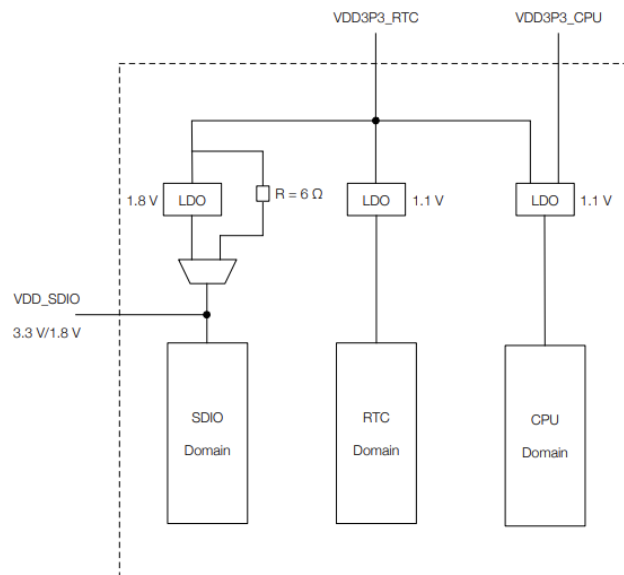


Name	No.	Type	Function
Analog			
VDDA	1	P	Analog power supply (2.3 V ~ 3.6 V)
LNA_IN	2	I/O	RF input and output
VDD3P3	3	P	Analog power supply (2.3 V ~ 3.6 V)
VDD3P3	4	P	Analog power supply (2.3 V ~ 3.6 V)
VDD3P3_RTC			
SENSOR_VP	5	I	GPIO36, ADC1_CH0, RTC_GPIO0
SENSOR_CAPP	6	I	GPIO37, ADC1_CH1, RTC_GPIO1
SENSOR_CAPN	7	I	GPIO38, ADC1_CH2, RTC_GPIO2
SENSOR_VN	8	I	GPIO39, ADC1_CH3, RTC_GPIO3
CHIP_PU	9	I	High: On; enables the chip Low: Off; the chip shuts down Note: Do not leave the CHIP_PU pin floating.
VDET_1	10	I	GPIO34, ADC1_CH6, RTC_GPIO4
VDET_2	11	I	GPIO35, ADC1_CH7, RTC_GPIO5
32K_XP	12	I/O	GPIO32, ADC1_CH4, RTC_GPIO9, TOUCH9, 32K_XP (32.768 kHz crystal oscillator input)
32K_XN	13	I/O	GPIO33, ADC1_CH5, RTC_GPIO8, TOUCH8, 32K_XN (32.768 kHz crystal oscillator output)
GPIO25	14	I/O	GPIO25, ADC2_CH8, RTC_GPIO6, DAC_1, EMAC_RXD0
GPIO26	15	I/O	GPIO26, ADC2_CH9, RTC_GPIO7, DAC_2, EMAC_RXD1
GPIO27	16	I/O	GPIO27, ADC2_CH7, RTC_GPIO17, TOUCH7, EMAC_RX_DV
MTMS	17	I/O	GPIO14, ADC2_CH6, RTC_GPIO16, TOUCH6, EMAC_TXD2, HSPICLK, HS2_CLK, SD_CLK, MTMS
MTDI	18	I/O	GPIO12, ADC2_CH5, RTC_GPIO15, TOUCH5, EMAC_TXD3, HSPIQ, HS2_DATA2, SD_DATA2, MTDI
VDD3P3_RTC	19	P	Input power supply for RTC IO (2.3 V ~ 3.6 V)
MTCK	20	I/O	GPIO13, ADC2_CH4, RTC_GPIO14, TOUCH4, EMAC_RX_ER, HSPID, HS2_DATA3, SD_DATA3, MTCK
MTDO	21	I/O	GPIO15, ADC2_CH3, RTC_GPIO13, TOUCH3, EMAC_RXD3, HSPICS0, HS2_CMD, SD_CMD, MTDO

Name	No.	Type	Function
GPIO2	22	I/O	GPIO2, ADC2_CH2, RTC_GPIO12, TOUCH2, HSPWIP, HS2_DATA0, SD_DATA0
GPIO0	23	I/O	GPIO0, ADC2_CH1, RTC_GPIO11, TOUCH1, EMAC_TX_CLK, CLK_OUT1,
GPIO4	24	I/O	GPIO4, ADC2_CH0, RTC_GPIO10, TOUCH0, EMAC_TX_ER, HSPHID, HS2_DATA1, SD_DATA1
VDD_SDIO			
GPIO16	25	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
VDD_SDIO	26	P	Output power supply: 1.8 V or the same voltage as VDD3P3_RTC
GPIO17	27	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
SD_DATA_2	28	I/O	GPIO9, HS1_DATA2, U1RXD, SD_DATA2, SPIHD
SD_DATA_3	29	I/O	GPIO10, HS1_DATA3, U1TXD, SD_DATA3, SPIWP
SD_CMD	30	I/O	GPIO11, HS1_CMD, U1RTS, SD_CMD, SPICS0
SD_CLK	31	I/O	GPIO6, HS1_CLK, U1CTS, SD_CLK, SPICLK
SD_DATA_0	32	I/O	GPIO7, HS1_DATA0, U2RTS, SD_DATA0, SPIQ
SD_DATA_1	33	I/O	GPIO8, HS1_DATA1, U2CTS, SD_DATA1, SPID
VDD3P3_CPU			
GPIO5	34	I/O	GPIO5, HS1_DATA6, VSPICS0, EMAC_RX_CLK
GPIO18	35	I/O	GPIO18, HS1_DATA7, VSPICLK
GPIO23	36	I/O	GPIO23, HS1_STROBE, VSPID
VDD3P3_CPU	37	P	Input power supply for CPU IO (1.8 V ~ 3.6 V)
GPIO19	38	I/O	GPIO19, U0CTS, VSPIQ, EMAC_TXD0
GPIO22	39	I/O	GPIO22, U0RTS, VSPWIP, EMAC_TXD1
U0RXD	40	I/O	GPIO3, U0RXD, CLK_OUT2
U0TXD	41	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
GPIO21	42	I/O	GPIO21, VSPHID, EMAC_TX_EN
Analog			
VDDA	43	P	Analog power supply (2.3 V ~ 3.6 V)
XTAL_N	44	O	External crystal output
XTAL_P	45	I	External crystal input
VDDA	46	P	Analog power supply (2.3 V ~ 3.6 V)
CAP2	47	I	Connects to a 3.3 nF (10%) capacitor and 20 kΩ resistor in parallel to CAP1

Name	No.	Type	Function
CAP1	48	I	Connects to a 10 nF series capacitor to ground
GND	49	P	Ground

SKEMA POWER ESP32



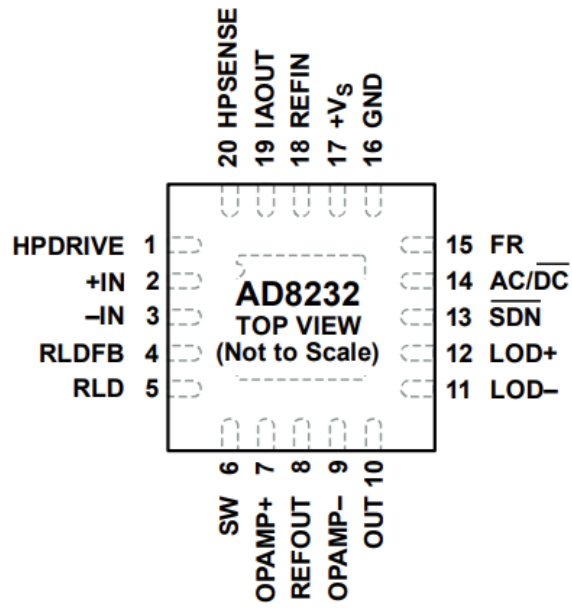
A. Spesifikasi:

1. Operating voltage : 3.3V
2. Input voltage : 7-12V (Vin)
3. Digital IO Pin (DIO) : 25
4. Analog Input Pin (ADC) : 6
5. Analog Output Pin (DAC) : 2
6. UART : 3
7. SPI : 2
8. I2C : 3
9. Flash Memory 4 MB
10. SRAM : 520 KB
11. Clock Speed : 240 Mhz
12. Wi Fi : IEEE 802.11 b/g/n/e/i
13. Mode supported : AP, STA, AP+STA
14. CP2102 USB ccontroller

B. Aplikasi:

1. IOT
2. Smart lighting
3. Wireless sensor networks
4. Home automation
5. Telemetry
6. WiFi / Bluetooth Audio player
7. WiFi camera
8. MP3 decoding
9. WiFi range extender

LAMPIRAN 3: DATA SHEET AD8232



SPECIFICATIONS

$V_S = 3\text{ V}$, $V_{REF} = 1.5\text{ V}$, $V_{CM} = 1.5\text{ V}$, $T_A = 25^\circ\text{C}$, operating temperature (T_{OPR}) = -40°C to $+105^\circ\text{C}$ for the W grade, FR = low, SDN = high, and AC/DC = low, unless otherwise noted.

Table 1.

Parameter	Symbol	Test Conditions/ Comments	A Grade			W Grade			Unit
			Min	Typ	Max	Min	Typ	Max	
INSTRUMENTATION AMPLIFIER									
Common-Mode Rejection Ratio, DC to 60 Hz	CMRR	$V_{CM} = 0.35\text{ V to }2.85\text{ V}$, $V_{DIFF} = 0\text{ V}$	80	86					dB
		T_{OPR}				80			dB
		$V_{CM} = 0.35\text{ V to }2.85\text{ V}$, $V_{DIFF} = \pm 0.3\text{ V}$		80			80		dB
Power Supply Rejection Ratio	PSRR	$V_S = 2.0\text{ V to }3.5\text{ V}$	76	90					dB
		T_{OPR}				76	90		dB
Offset Voltage (RTI) Instrumentation Amplifier Inputs	V_{OS}			3	8				mV
		T_{OPR}				3	8		mV
DC Blocking Input ¹				5	50		5	50	μV
Average Offset Drift Instrumentation Amplifier Inputs				10			10		$\mu\text{V}/^\circ\text{C}$
		T_{OPR}					10		$\mu\text{V}/^\circ\text{C}$
DC Blocking Input ¹				0.05			0.05		$\mu\text{V}/^\circ\text{C}$
Input Bias Current	I_b			50	200		50	200	pA
		$T_A = 0^\circ\text{C to }70^\circ\text{C}$		1					nA
		T_{OPR}						3.75	nA
Input Offset Current	I_{OS}			25	100		25	100	pA
		$T_A = 0^\circ\text{C to }70^\circ\text{C}$		1					nA
		T_{OPR}						1	nA
Input Impedance Differential				10 7.5			10 7.5		$\text{G}\Omega \text{pF}$
Common Mode				5 15			5 15		$\text{G}\Omega \text{pF}$
Input Voltage Noise (RTI) Spectral Noise Density		$f = 1\text{ kHz}$		100			100		$\text{nV}/\sqrt{\text{Hz}}$
		$f = 0.1\text{ Hz to }10\text{ Hz}$		12			12		$\mu\text{V p-p}$
Peak-to-Peak Voltage Noise		$f = 0.5\text{ Hz to }40\text{ Hz}$		14			14		$\mu\text{V p-p}$
Input Voltage Range		$T_A = 0^\circ\text{C to }70^\circ\text{C}$	0.2		$+V_S$				V
		T_{OPR}				0.2		$+V_S$	V
DC Differential Input Range	V_{DIFF}		-300		+300				mV
		T_{OPR}				-300		+300	mV
Output Output Swing		$R_L = 50\text{ k}\Omega$	0.1		$+V_S - 0.1$				V
		T_{OPR}				0.1		$+V_S - 0.1$	V
Short-Circuit Current	I_{OUT}			6.3			6.3		mA
Gain	A_V			100			100		V/V
Gain Error		$V_{DIFF} = 0\text{ V}$		0.4			0.4		%
		$V_{DIFF} = -300\text{ mV to }+300\text{ mV}$		1	3.5			3.5	%
		T_{OPR}						7.9	%
Average Gain Drift		$T_A = 0^\circ\text{C to }70^\circ\text{C}$		12					$\text{ppm}/^\circ\text{C}$
		T_{OPR}					12		$\text{ppm}/^\circ\text{C}$
Bandwidth	BW			2			2		kHz
RFI Filter Cutoff (Each Input)				1			1		MHz

Parameter	Symbol	Test Conditions/ Comments	A Grade			W Grade			Unit
			Min	Typ	Max	Min	Typ	Max	
OPERATIONAL AMPLIFIER (A1)									
Offset Voltage	V_{OS}	T_{OPR}	1	5		1	5	mV	
Average TC		$T_A = 0^\circ\text{C to } 70^\circ\text{C}$ T_{OPR}	5			5		$\mu\text{V}/^\circ\text{C}$ $\mu\text{V}/^\circ\text{C}$	
Input Bias Current	I_b	$T_A = 0^\circ\text{C to } 70^\circ\text{C}$ T_{OPR}	100			100		pA nA	
Input Offset Current	I_{OS}	$T_A = 0^\circ\text{C to } 70^\circ\text{C}$ T_{OPR}	100			2.5		nA pA nA	
Input Voltage Range		T_{OPR}	0.1		$+V_S - 0.1$	0.1		V	
Common-Mode Rejection Ratio	CMRR	$V_{CM} = 0.5\text{ V to } 2.5\text{ V}$	100			100		dB	
Power Supply Rejection Ratio	PSRR		100			100		dB	
Large Signal Voltage Gain	A_{VO}		110			110		dB	
Output Voltage Range		$R_L = 50\text{ k}\Omega$ T_{OPR}	0.1		$+V_S - 0.1$	0.1		V V	
Short-Circuit Current Limit	I_{OUT}		12			12		mA	
Gain Bandwidth Product	GBP		100			100		kHz	
Slew Rate	SR		0.02			0.02		V/ μs	
Voltage Noise Density (RTI)	e_n	$f = 1\text{ kHz}$	60			60		nV/ $\sqrt{\text{Hz}}$	
Peak-to-Peak Voltage Noise (RTI)	$e_{n\text{ p-p}}$	$f = 0.1\text{ Hz to } 10\text{ Hz}$ $f = 0.5\text{ Hz to } 40\text{ Hz}$	6 8			6 8		$\mu\text{V p-p}$ $\mu\text{V p-p}$	
RIGHT LEG DRIVE AMPLIFIER (A2)									
Output Swing		$R_L = 50\text{ k}\Omega$	0.1		$+V_S - 0.1$	0.1		V	
Short-Circuit Current	I_{OUT}		11			11		mA	
Integrator Input Resistor			120	150	180	120	150	180	k Ω
Gain Bandwidth Product	GDP		100			100		kHz	
REFERENCE BUFFER (A3)									
Offset Error	V_{OS}	$R_L > 50\text{ k}\Omega$	1			1		mV	
Input Bias Current	I_b		100			100		pA	
Short-Circuit Current Limit	I_{OUT}		12			12		mA	
Voltage Range		$R_L = 50\text{ k}\Omega$ T_{OPR}	0.1		$+V_S - 0.7$	0.1		V V	
DC LEADS OFF COMPARATORS									
Threshold Voltage				$+V_S - 0.5$		$+V_S - 0.5$		V	
Hysteresis			60			60		mV	
Propagation Delay			0.5			0.5		μs	
AC LEADS OFF DETECTOR									
Square Wave Frequency	f_{AC}		50	100	175	50	100	175	kHz
Square Wave Amplitude	I_{AC}			200			200		nA p-p
Impedance Threshold		Between +IN and -IN	10	20		10	20		M Ω
Detection Delay				110			110		μs
FAST RESTORE CIRCUIT									
Switches		S1 and S2							
On Resistance	R_{ON}		8	10	12	8	10	12	k Ω
Off Leakage				100			100		pA
Window Comparator									
Threshold Voltage		From either rail		50			50		mV
Propagation Delay				2			2		μs
Switch Timing Characteristics									
Feedback Recovery Switch On Time	t_{SW1}			110			110		ms
Filter Recovery Switch On Time	t_{SW2}			55			55		ms
Fast Restore Reset	t_{BST}			2			2		μs

Parameter	Symbol	Test Conditions/ Comments	A Grade			W Grade			Unit
			Min	Typ	Max	Min	Typ	Max	
LOGIC INTERFACE									
Input Characteristics									
Input Voltage ($\overline{AC/DC}$ and FR)									
Low	V_{IL}			1.24			1.24		V
High	V_{IH}			1.35			1.35		V
Input Voltage (\overline{SDN})									
Low	V_{IL}			2.1			2.1		V
High	V_{IH}			0.5			0.5		V
Output Characteristics									
Output Voltage									
Low	V_{OL}			0.05			0.05		V
High	V_{OH}			2.95			2.95		V
SYSTEM SPECIFICATIONS									
Quiescent Supply Current									
		$T_A = 0^\circ\text{C to }70^\circ\text{C}$		170	230		170	230	μA
		T_{OFR}		210					μA
Shutdown Current									
		$T_A = 0^\circ\text{C to }70^\circ\text{C}$		40	500		40	500	nA
		T_{OFR}		100					nA
								612	nA
Supply Range									
Specified Temperature Range									
				2.0	3.5		2.0	3.5	V
Operational Temperature Range									
				-40	+85		-40	+105	$^\circ\text{C}$

¹ Offset referred to the input of the instrumentation amplifier inputs. See the Input Referred Offsets section for additional information.

LAMPIRAN 4: Source Code

```
#include <Wire.h> //library oled
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h> //library blynk

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

#define OLED_RESET -1 // Reset pin # (or -1 if sharing Arduino reset pin)
#define MAX_DATA_POINTS 30
#define HEART_RATE_FREQUENCY 1.5
#define HEART_RATE_AMPLITUDE 1500

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);

// 'logokampus-removebg-preview', 128x64px
const unsigned char logo_kampus [] PROGMEM = {
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0x7f, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfc, 0x1f, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xe0, 0x07, 0xff, 0xff, 0xff, 0xff, 0xff,
  0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xff,
  0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xc0, 0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xff,
  0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xff,
  0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xff,
  0xff,
  0xff, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xff,
  0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xff,
  0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xe0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0xff,
  0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0xe0, 0x00, 0x02, 0x78, 0x7b, 0x90, 0x00, 0x03, 0xff, 0xff,
  0xff, 0xff,
```

0xff, 0xff, 0xff, 0xff, 0xe0, 0x00, 0x57, 0xa8, 0x2b, 0xb9, 0x80, 0x03, 0xff, 0xff,
0xff, 0xff,
0xff, 0xff, 0xff, 0x80, 0x20, 0x10, 0x70, 0x00, 0x00, 0x1b, 0xb4, 0x03, 0x00,
0xff, 0xff, 0xff,
0xff, 0xff, 0xfc, 0x00, 0x00, 0xbc, 0x00, 0x00, 0x00, 0x00, 0x2c, 0x80, 0x00,
0x1f, 0xff, 0xff,
0xff, 0xff, 0xf0, 0x00, 0x03, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x03, 0x90, 0x00,
0x07, 0xff, 0xff,
0xff, 0xff, 0xc0, 0x00, 0x0d, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xb0, 0x00,
0x01, 0xff, 0xff,
0xff, 0xff, 0x00, 0x00, 0x78, 0x00, 0x00, 0x00, 0x00, 0x07, 0xc0, 0x0f, 0x00,
0x00, 0xff, 0xff,
0xff, 0xfe, 0x00, 0x03, 0x20, 0x79, 0xf0, 0x00, 0x00, 0x07, 0x0e, 0x02, 0x40,
0x00, 0x3f, 0xff,
0xff, 0xfc, 0x00, 0x03, 0x80, 0xe7, 0xc0, 0x00, 0x80, 0x07, 0x3e, 0x01, 0xf0,
0x00, 0x1f, 0xff,
0xff, 0xf0, 0x00, 0x00, 0x04, 0xc0, 0x00, 0x01, 0x80, 0x02, 0x38, 0x30, 0x40,
0x00, 0x07, 0xff,
0xff, 0xc0, 0x00, 0x30, 0x19, 0x8f, 0x90, 0x00, 0x80, 0x0c, 0x31, 0xf0, 0x02,
0x00, 0x03, 0xff,
0xff, 0x00, 0x00, 0x08, 0x3b, 0xfc, 0x18, 0x02, 0xa0, 0x18, 0x01, 0xc0, 0x0f,
0x00, 0x00, 0xff,
0xfc, 0x00, 0x00, 0xb0, 0xb4, 0x00, 0x05, 0x0a, 0xf0, 0xf0, 0x03, 0x8f, 0x80,
0x80, 0x00, 0x1f,
0xe0, 0x00, 0x00, 0x41, 0xa3, 0xc0, 0x0a, 0xae, 0xd3, 0xf8, 0x00, 0x1f, 0x03,
0xc0, 0x00, 0x03,
0xf0, 0x00, 0x06, 0xc3, 0x3e, 0x00, 0x0f, 0x60, 0x05, 0xf8, 0x00, 0x38, 0x01,
0x00, 0x00, 0x0f,
0xf8, 0x00, 0x03, 0x82, 0x40, 0x70, 0x0e, 0x00, 0x40, 0x70, 0x46, 0x01, 0xf0,
0xe0, 0x00, 0x0f,
0xf8, 0x00, 0x01, 0x0a, 0xbc, 0x1f, 0xf8, 0x00, 0x00, 0x0f, 0xf8, 0x03, 0xf0,
0xf0, 0x00, 0x1f,
0xfc, 0x00, 0x07, 0x1b, 0xf0, 0x0f, 0xe0, 0x1c, 0xf8, 0x07, 0xf0, 0x03, 0x00,
0x00, 0x00, 0x1f,
0xfc, 0x00, 0x1e, 0x1b, 0x00, 0x03, 0xc2, 0x61, 0x03, 0x43, 0xe0, 0x00, 0x20,
0x00, 0x00, 0x1f,
0xfc, 0x00, 0x0e, 0x1a, 0xe0, 0x01, 0x85, 0x83, 0x88, 0x81, 0x80, 0x00, 0xfe,
0x3c, 0x00, 0x3f,
0xfc, 0x00, 0x1c, 0x11, 0xc0, 0x03, 0x83, 0x21, 0x4e, 0x41, 0xc0, 0x01, 0xc8,
0x3c, 0x00, 0x3f,
0xfc, 0x00, 0x1c, 0x0f, 0x00, 0x1f, 0xff, 0x4d, 0x9c, 0x7e, 0xf8, 0x00, 0x00,
0x00, 0x00, 0x3f,
0xfc, 0x00, 0x04, 0x66, 0x41, 0xff, 0xff, 0x75, 0xff, 0x7f, 0xff, 0xc0, 0x38, 0x30,
0x00, 0x3f,
0xfc, 0x00, 0x38, 0x74, 0xc0, 0x0f, 0xe3, 0x2b, 0xb4, 0x63, 0xf8, 0x00, 0xfe,
0x1c, 0x00, 0x3f,

0xfc, 0x00, 0x3c, 0x39, 0x80, 0x00, 0x93, 0x3e, 0x64, 0xc0, 0x00, 0x00, 0x04,
0x1c, 0x00, 0x3f,
0xfe, 0x00, 0x04, 0x0f, 0x20, 0x03, 0xc8, 0xc0, 0x91, 0x81, 0xc0, 0x00, 0x30,
0x1c, 0x00, 0x3f,
0xfe, 0x00, 0x1c, 0x32, 0x70, 0x07, 0xe0, 0x70, 0x0e, 0x33, 0xf0, 0x01, 0xf8,
0x20, 0x00, 0x3f,
0xfe, 0x00, 0x10, 0x1c, 0x60, 0x0f, 0xf1, 0x0f, 0xf0, 0x0f, 0xf0, 0x00, 0x1e, 0x3c,
0x00, 0x7f,
0xff, 0x00, 0x18, 0x0c, 0xe0, 0x3f, 0xe4, 0x00, 0x00, 0x16, 0xde, 0x00, 0x00,
0x60, 0x00, 0xff,
0xff, 0xc0, 0x00, 0x01, 0xd8, 0x00, 0x0f, 0x80, 0x00, 0xf8, 0x00, 0x07, 0xe0,
0x78, 0x01, 0xff,
0xff, 0xf8, 0x00, 0x06, 0x5c, 0x00, 0x0f, 0xfc, 0x1f, 0x50, 0x00, 0x03, 0xf0,
0x70, 0x0f, 0xff,
0xff, 0xff, 0x07, 0x87, 0xbc, 0x00, 0x1f, 0xae, 0xf2, 0xa8, 0x00, 0x00, 0x70,
0xc0, 0xff, 0xff,
0xff, 0xfe, 0x02, 0x01, 0xd9, 0x80, 0x0d, 0x02, 0xa0, 0x50, 0x00, 0x3f, 0x01,
0xe0, 0x7f, 0xff,
0xff, 0xfc, 0x01, 0xe0, 0x1d, 0xc0, 0x18, 0x02, 0xa0, 0x0c, 0x00, 0x0f, 0x02,
0xc0, 0x3f, 0xff,
0xff, 0xf8, 0x00, 0xc0, 0x60, 0x6c, 0x00, 0x01, 0xc0, 0x00, 0x03, 0x87, 0x87,
0x00, 0x1f, 0xff,
0xff, 0xf8, 0x00, 0x58, 0x1f, 0xb7, 0x80, 0x69, 0x88, 0x40, 0x61, 0xf1, 0x19,
0x00, 0x1f, 0xff,
0xff, 0xf8, 0x00, 0x22, 0x00, 0x39, 0xc0, 0x3f, 0xf8, 0x00, 0x78, 0xe0, 0x3c,
0x00, 0x1f, 0xff,
0xff, 0xfc, 0x00, 0x0c, 0x00, 0x40, 0xe0, 0x00, 0x00, 0x00, 0x3c, 0x70, 0xd8,
0x00, 0x1f, 0xff,
0xff, 0xfe, 0x00, 0x07, 0x00, 0x3f, 0xf8, 0x00, 0x00, 0x00, 0x38, 0x02, 0x20,
0x00, 0x3f, 0xff,
0xff, 0xff, 0x00, 0x02, 0x70, 0x00, 0x06, 0x0e, 0x70, 0x00, 0x18, 0x01, 0xc0,
0x00, 0x7f, 0xff,
0xff, 0xff, 0x80, 0x00, 0xec, 0x00, 0x03, 0xfe, 0x7f, 0xd0, 0x00, 0x1c, 0x00,
0x00, 0xff, 0xff,
0xff, 0xff, 0xc0, 0x00, 0x1f, 0x00, 0x0f, 0xfe, 0x7f, 0xf0, 0x00, 0x64, 0x00, 0x03,
0xff, 0xff,
0xff, 0xff, 0xf0, 0x00, 0x04, 0x80, 0x00, 0x60, 0x06, 0x40, 0x01, 0xe0, 0x00,
0x07, 0xff, 0xff,
0xff, 0xff, 0xf8, 0x00, 0x00, 0x38, 0x00, 0x00, 0x00, 0x00, 0x1c, 0x00, 0x00,
0x0f, 0xff, 0xff,
0xff, 0xff, 0xfc, 0x00, 0x00, 0x11, 0xc0, 0x00, 0x00, 0x05, 0x88, 0x00, 0x00,
0x3f, 0xff, 0xff,
0xff, 0xff, 0xfe, 0x00, 0x00, 0x03, 0xb5, 0xc8, 0x49, 0xe9, 0xc0, 0x00, 0x00,
0x7f, 0xff, 0xff,
0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x1b, 0xdd, 0xdd, 0xe4, 0x00, 0x00, 0x00,
0xff, 0xff, 0xff,

```

    0xff, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0x05, 0x40, 0x00, 0x00, 0x00, 0x01,
    0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,
    0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x01, 0x80, 0x00, 0x00, 0x00, 0x01,
    0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x03, 0xc0, 0x00, 0x00, 0x7f, 0xff, 0xff,
    0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x0f, 0xf0, 0x00, 0x1f, 0xff, 0xff, 0xff,
    0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
    0xff
};
#define piLOplus_LAYellow 32 //LO+ RX2 Konek 0 DC 1
#define piLOmin_RARed 33 //LO- TX2 Konek 0 DC 1
#define paOutput 36 //LO- TX2

unsigned char dLOplus_LAYellow;
unsigned char dLOmin_RARed;
int dOutput = 0;
#define tresholdSinyal 3000
const int MAX_SIGNAL_POINTS = 128;
int heartRateSignal[MAX_SIGNAL_POINTS];
int signalIndex = 0;
unsigned long heartRateMillis = 0;
int heartRatePeriod = 1000 / HEART_RATE_FREQUENCY;

char auth [] = "IEVumpVof0BaWQG8i6ts4UxCea2doDPb";
char ssid [] = "Mi 10T";
char pass [] = "pgntauaja";

void setup()
{
    // Initialize OLED display
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // Address 0x3C for 128x64
    display.clearDisplay(); // Clear the display
    display.drawBitmap(0,0,logo_kampus,128,64,1);
    display.display();
    //Initialize sensor
    Serial.begin(115200);
    pinMode(piLOplus_LAYellow, INPUT_PULLUP);
    pinMode(piLOmin_RARed, INPUT_PULLUP);

    my_moving_ave_clear();

    //koneksi wifi

```

```

WiFi.begin(ssid,pass);
  // Inializa Blynk
Blynk.begin(auth, ssid, pass);
//Inisialisasi pin untuk tampilan ke blynk
Blynk.virtualWrite(V0, "Persiapan...");
}

```

```

int ledState = LOW;
unsigned long previousMillis = 0;
const long interval = 1 ; //1000Hz
float BPM = 0;
int mavBPM = 0;
int error = 0; //untuk kalibrasi
int msBPM;
float dataBuffer[MAX_DATA_POINTS];
int dataIndex = 0;

```

```

int flagBPM = 0;

```

```

void loop()
{
  Blynk.run();

```

```

  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= interval)
  {
    previousMillis = currentMillis;

```

```

    dLOmin_RARed = digitalRead(piLOmin_RARed);
    dLOplus_LAYellow = digitalRead(piLOplus_LAYellow);
    dOutput = analogRead(paOutput);

```

```

    //dataLOPrint();
    dataSignalPrint();

```

```

    // LEAD TIDAK TERHUBUNG
    // LED NYALA
    if (dLOmin_RARed == HIGH || dLOplus_LAYellow == HIGH)
    {
      msBPM = 0;
    }

```

```

    // LEAD TERHUBUNG
    // LED KEDIP

```

```

else {

    if (dOutput > tresholdSinyal && flagBPM == 0) {

        flagBPM = 1;

        if (msBPM > 525)
        {
            BPM = 60000.0 / msBPM;
            mavBPM = my_moving_average(BPM, 0);
            mavBPM -= error;
            //Serial.print(BPM); Serial.print(" ");
            Serial.println(mavBPM);
        }

        msBPM = 0;

        // Format the output string
        // String outputString = "BPM: " + String(mavBPM);
        // Print the output string on the OLED display
        /*display.clearDisplay();
        display.setTextSize(1.5);
        display.setTextColor(SSD1306_WHITE);
        display.setCursor(85,55);
        display.println(outputString);*/
        display.display(); // Update the display

        dataBuffer[dataIndex] = mavBPM;
        dataIndex++;
        if(dataIndex >= MAX_DATA_POINTS){
            dataIndex = 0;
        }
        display.clearDisplay();

        /*int x = 0;
        for(int i = dataIndex; i < dataIndex + MAX_DATA_POINTS; i++){
            int y = map(dataBuffer[i % MAX_DATA_POINTS], 0, 200, 63, 10); // Map the
BPM value to the OLED screen
            display.drawPixel(x, y, SSD1306_WHITE);
            x++;
        }

        display.display();*/

```



```

        heartRateSignal[signalIndex] = dOutput;
        signalIndex = (signalIndex + 1) % MAX_SIGNAL_POINTS;
        displayHeartRateSignal();
//kirim data ke widget
Blynk.virtualWrite(V1, mavBPM);
    }

    else if (dOutput < tresholdSinyal)
    {
        flagBPM = 0;
        msBPM++;
    }
}
}
if (currentMillis - heartRateMillis >= heartRatePeriod / 2)
{
    heartRateMillis = currentMillis;

    float heartRateValue = HEART_RATE_AMPLITUDE * sin(2 * PI *
HEART_RATE_FREQUENCY * heartRateMillis / 1000);
    heartRateValue += HEART_RATE_AMPLITUDE; // Shift the signal to make it
positive
    heartRateValue /= 2; // Scale the signal to fit the OLED display

    // Format the output string
    String outputString = "BPM: " + String(mavBPM) + " HR: " +
String(heartRateValue);

    // Print the output string on the OLED display
    /*display.clearDisplay();
    display.setTextSize(1.5);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(5, 20);
    display.println(outputString); */
    display.display(); // Update the display

    Blynk.virtualWrite(V5,heartRateValue);
}
}

void dataPrint() {
    //Serial.print(dLOplus); Serial.print("+");
    //Serial.print(dLOmin); Serial.print("-");
    Serial.print(dOutput);
    Serial.println();
}

```

```

void dataLOPrint() {
  Serial.print(dLOplus_LAYellow); Serial.print("LA");
  Serial.print(dLOmin_RARed); Serial.print("RA");
  Serial.println();
}

void dataSignalPrint() {
  Serial.print(dOutput);
  Serial.println();
}

void displayHeartRateSignal() {
  // Tampilkan sinyal detak jantung ke dalam OLED
  display.clearDisplay();

  for (int i = 1; i < MAX_SIGNAL_POINTS; i++) {
    int x0 = map(i - 1, 0, MAX_SIGNAL_POINTS - 1, 0, SCREEN_WIDTH);
    int x1 = map(i, 0, MAX_SIGNAL_POINTS - 1, 0, SCREEN_WIDTH);
    int y0 = map(heartRateSignal[(signalIndex + i - 1) % MAX_SIGNAL_POINTS],
0, 4095, SCREEN_HEIGHT, 0);
    int y1 = map(heartRateSignal[(signalIndex + i) % MAX_SIGNAL_POINTS], 0,
4095, SCREEN_HEIGHT, 0);
    display.drawLine(x0, y0, x1, y1, SSD1306_WHITE);
  }
  String outputString = "BPM: " + String(mavBPM);
  display.setTextSize(1.5);
  display.setTextColor(SSD1306_WHITE);
  display.setCursor(80,55);
  display.println(outputString);
  display.display();
}

//-----#MY AVERAGE
#define n_my_data 5
float my_data[n_my_data];
int i_my_data = 0;

float my_moving_average(float in_data, int debug)
{
  float my_ave;
  float my_sum;

```

```

my_data[i_my_data] = in_data;

if (my_data[n_my_data - 1] == 0)
{
    my_sum = 0;

    for (int i = 0; i <= i_my_data; i++)
    {
        my_sum += my_data[i];
    }
    my_ave = my_sum / (i_my_data + 1);
}
else
{
    my_sum = 0;
    for (int i = 0; i < n_my_data; i++)
    {
        my_sum += my_data[i];
    }
    my_ave = my_sum / n_my_data;
}

///  

if (debug)
{
    for (int i = 0; i < n_my_data; i++)
    {
        Serial.print(i); Serial.print(" "); Serial.println(my_data[i]);
    }
    Serial.print("i"); Serial.print(" "); Serial.println(i_my_data);
    Serial.print("Last"); Serial.print(" "); Serial.println(my_data[n_my_data - 1]);

    Serial.print("Sum"); Serial.print(" "); Serial.println(my_sum);
    Serial.print("Ave"); Serial.print(" "); Serial.println(my_ave);
    Serial.println();
}

i_my_data++;
if (i_my_data >= n_my_data)
    i_my_data = 0;

return my_ave;
}

```

```
void my_moving_ave_clear()
{
    //Make sure All Array is 0
    for (int i = 0; i < n_my_data; i++)
    {
        my_data[i] = 0;
    }
}
```