

**MENGUKUR SATURASI OKSIGEN DALAM DARAH  
DENGAN SENSOR MAX30100  
BERBASIS ARDUINO**

**TUGAS AKHIR**  
Diajukan untuk Memenuhi Salah Satu Syarat Kelulusan  
Program Pendidikan Sarjana

Oleh:  
Karin Azalia Sutianto  
2017130034



JURUSAN INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER – LIKMI  
BANDUNG  
2021

**MENGUKUR SATURASI OKSIGEN DALAM DARAH  
DENGAN SENSOR MAX30100  
BERBASIS ARDUINO**

Oleh :  
Karin Azalia Sutianto  
2017130034

Bandung, 3 Agustus 2021  
Menyetujui,

Sudimanto, S.T., M.Kom.  
Pembimbing

Dhanny Setiawan, S.T., M.T.  
Ketua Jurusan

JURUSAN INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER – LIKMI  
BANDUNG  
2021

## ABSTRAK

Oksigen mempunyai peranan penting terhadap tubuh manusia karena tanpa suplai darah yang mengandung oksigen organ tubuh tidak bisa bekerja dengan baik, maka dari itu perlu dipantau secara teratur. Terdapat dua cara pengukuran saturasi oksigen yaitu *invasive* dan *non-invasive*. *Invasive* merupakan pengukuran saturasi oksigen dengan cara mengambil darah dari arteri kemudian dianalisa memakai alat laboratorium, sedangkan *non-invasive* merupakan suatu tindakan yang dilakukan tanpa memasukkan alat ke dalam tubuh dan tidak menyebabkan kerusakan kulit dan organ manapun pada tubuh manusia. Salah satu alat untuk mengukur saturasi oksigen yang beredar di masyarakat yaitu *pulse oximetry*, alat ini juga yang akan dipakai sebagai alat pembanding dalam pengujian ini.

Pengukuran saturasi oksigen ini akan direalisasikan menggunakan sebuah sensor MAX30100 sebagai pendeteksi saturasi oksigen yang dirancang menggunakan Arduino UNO sebagai sistem kontroler dan LCD (*Liquid Crystal Display*) sebagai alat untuk menampilkan hasil dari data berupa angka saturasi oksigen dengan satuan persen serta kondisi status angka saturasi oksigen. Penelitian ini menunjukkan bahwa alat ini dapat mendeteksi saturasi oksigen yang dimana usia, berat badan, jenis kelamin maupun kegiatan yang dilakukan tidak mempengaruhi tingkat saturasi oksigen.

## KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas berkat-Nya yang melimpah, sehingga penulis dapat menyelesaikan tugas akhir ini dengan judul “Mengukur Saturasi Oksigen Dalam Darah Dengan Sensor MAX30100 Berbasis Arduino” dengan baik.

Tugas akhir ini dapat selesai tidak terlepas dari dukungan berbagai pihak, baik secara langsung maupun tidak langsung. Secara khusus, penulis mengucapkan terima kasih kepada :

1. Bapak Sudimanto, S.T., M.Kom. selaku dosen pembimbing yang telah banyak membantu penulis dalam memberikan masukan, saran, kritik, dan arahan selama penulis menyelesaikan tugas akhir.
2. Bapak Dhanny Setiawan, S.T., M.T. selaku ketua jurusan Informatika yang telah memberikan bantuan berupa ide, saran, dan masukan kepada penulis.
3. Keluarga penulis yang telah memberikan dukungan dalam segi moril dan materil selama penulis menyelesaikan tugas akhir ini.
4. Teman-teman gereja yang sudah membantu penulis menjadi objek pengujian dalam proses penyusunan tugas akhir ini.
5. Lydia dan Fernando yang telah membantu penulis dalam memberi saran dan dukungan, serta berjuang bersama dalam penyusunan tugas akhir.
6. Seluruh teman angkatan 2017 dan seluruh teman penulis yang tidak dapat penulis sebutkan satu persatu, yang selalu mendukung dan membantu penulis dalam menyelesaikan tugas akhir ini.
7. Seluruh dosen dan staf STMIK LIKMI yang telah memberi ilmu dan dukungan untuk penulis selama ini.

Akhir kata, penulis menyadari banyak kekurangan dalam penyelesaian tugas akhir ini, oleh karena itu saran, kritik, dan masukan yang membangun sangat penulis harapkan dalam penyempurnaan tugas akhir ini. Semoga tugas akhir ini dapat memberikan pembelajaran dan wawasan bagi pihak-pihak yang membutuhkan.

Bandung, 3 Agustus 2021

Penulis

## DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR .....	ii
DAFTAR ISI.....	iv
DAFTAR GAMBAR .....	vii
DAFTAR TABEL .....	viii
DAFTAR SIMBOL .....	viiix
DAFTAR LAMPIRAN .....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan Penelitian .....	3
1.4 Batasan Masalah .....	3
1.5 Kegunaan Hasil.....	3
1.6 Metodologi Penelitian.....	3
1.7 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	5
2.1 Darah.....	5
2.2 Saturasi Oksigen.....	5
2.3 Mikrokontroler.....	6
2.3.1 Mikrokontroler ATmega328 .....	7
2.3.2 Konfigurasi Pin ATmega328 .....	8
2.4 Sensor .....	9
2.5 LCD ( <i>Liquid Crystal Display</i> ).....	11
2.6 Arduino .....	13

2.7	Arduino IDE ( <i>Integrated Development Environment</i> ).....	14
2.8	Bahasa Pemrograman C.....	15
2.9	OOAD ( <i>Object Oriented Analysis and Design</i> ) .....	16
2.10	UML ( <i>Unified Modeling Language</i> ) .....	16
2.10.1	<i>Use Case Diagram</i> .....	17
2.10.2	<i>Class Diagram</i> .....	18
2.10.3	<i>Deployment Diagram</i> .....	18
2.10.4	<i>State Machine Diagram</i> .....	19
BAB III ANALISIS DAN PERANCANGAN .....		20
3.1	Gambaran Umum.....	20
3.2	Spesifikasi Kebutuhan.....	20
3.2.1	Kebutuhan Perangkat Keras .....	20
3.2.2	Kebutuhan Perangkat Lunak.....	21
3.3	<i>Deployment Diagram</i> .....	21
3.4	<i>Class Diagram</i> .....	22
3.5	<i>State Machine Diagram</i> .....	22
3.6	Skematik Rancangan .....	23
BAB IV IMPLEMENTASI DAN PENGUJIAN .....		24
4.1	Implementasi antar Muka Perangkat Keras.....	24
4.2	Pengujian .....	26
4.2.1	Pengujian Sensor MAX30100 .....	26
4.3.1	Pengujian Sistem .....	26
BAB V KESIMPULAN DAN SARAN .....		30
5.1	Kesimpulan.....	30
5.2	Saran.....	30

DAFTAR PUSTAKA.....	31
LAMPIRAN.....	34



## DAFTAR GAMBAR

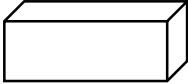

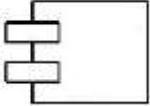

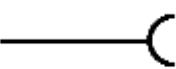
Gambar 1.1 <i>Pulse Oximetry</i> .....	2
Gambar 1.2 Blok Diagram Mikrokontroler ATmega328.....	8
Gambar 1.3 Konfigurasi Pin ATmega328P.....	8
Gambar 2.1 Sensor MAX30100.....	10
Gambar 2.2 16x2 LCD <i>Module</i> .....	11
Gambar 2.3 Tampilan Arduino IDE.....	14
Gambar 3.1 <i>Deployment</i> Diagram Alat ukur saturasi oksigen.....	21
Gambar 3.2 <i>Class</i> Diagram Alat Ukur saturasi oksigen.....	22
Gambar 3.3 <i>State Machine</i> Diagram Alat Ukur saturasi oksigen.....	22
Gambar 3.4 Skematik Rancangan.....	23
Gambar 4.1 Papan Arduino Uno.....	24
Gambar 4.2 Sensor MAX30100.....	24
Gambar 4.3 LCD ( <i>Liquid Crystal Display</i> ).....	25
Gambar 4.4 Alat ukur saturasi oksigen berbasis Arduino.....	25
Gambar 4.5 Hasil pengujian Sensor MAX30100.....	26

## DAFTAR TABEL

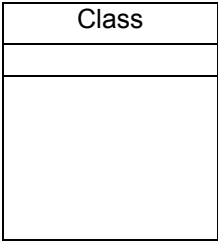

Tabel 2. 1 Kategori Nilai Saturasi Oksigen .....	6
Tabel 2. 2 <i>Pin Configuration</i> .....	12
Tabel 2. 3 Spesifikasi papan Arduino Uno.....	13
Tabel 3. 1 Kategori Nilai Saturasi Oksigen .....	20
Tabel 4. 1 Tabel Alat Ukur Saturasi Oksigen.....	27

## DAFTAR SIMBOL





### *Deployment Diagram*

<b>Nama Simbol</b>	<b>Simbol</b>	<b>Keterangan</b>
<i>Node</i>		Menggambarkan elemen fisik dari suatu modul perangkat keras
<i>Link</i>		Relasi antar <i>node</i>
Komponen		Menggambarkan komponen yang terdapat pada perangkat keras
<i>Required Interface</i>		Mendefinisikan keterkaitan kebutuhan komponen dan <i>node</i>
<i>Ball-and-Socket Joint</i>		Mendefinisikan keterkaitan kebutuhan komponen dan <i>node</i>

### *Class Diagram*

<b>Nama Simbol</b>	<b>Simbol</b>	<b>Keterangan</b>
Kelas / <i>Class</i>		Menggambarkan kelas pada struktur sistem.
Asosiasi / <i>Association</i>		Menggambarkan relasi antar kelas.

### State Machine Diagram

Nama Simbol	Simbol	Keterangan
Status Awal / <i>Start / Initial State</i>		Menggambarkan kondisi awal pada saat sistem mulai dinyalakan.
Status Akhir / <i>End / Final State</i>		Menggambarkan kondisi akhir dari sebuah kejadian pada <i>state machine</i> .
<i>Transition</i>		Merupakan perpindahan dari state ke state berikutnya.
<i>State</i>		Menggambarkan keadaan sistem pada waktu tertentu.

## DAFTAR LAMPIRAN

Listing Program Mikrokontroler.....	33
<i>Datasheet</i> Arduino Uno R3.....	36
<i>Datasheet</i> Sensor MAX30100.....	44
<i>Datasheet</i> LCD 16x2.....	49

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Darah merupakan komponen vital dalam tubuh manusia. Hampir semua gejala penyakit dalam dapat dideteksi melalui pemeriksaan darah. Secara umum darah dibagi kedalam empat golongan yaitu: A, AB, B, dan O. Darah merupakan alat transportasi dalam tubuh manusia yaitu untuk mengangkut bahan/zat makanan, oksigen, sisa-sisa metabolisme, dan hormon di dalam tubuh manusia (Adrian, Alodokter, 2020). Mengingat darah memiliki tugas yang sangat penting, maka kesehatannya perlu dipantau secara teratur.

Selain darah, oksigen pun mempunyai peranan penting terhadap tubuh manusia. Pasalnya tanpa suplai darah yang mengandung oksigen, organ tubuh tidak bisa bekerja dan tubuh akan mudah lemas. Menurut website "<https://www.alodokter.com/oksigenasi-bisa-membantu-menyembuhkan-penyakit-dan-luka>" adalah sebagai berikut : "*jika kadar oksigen turun hingga dibawah 90 persen, fungsi organ dan jaringan tubuh akan ikut menurun sehingga dapat menyebabkan gangguan kesehatan yang serius.*" (Adrian, 2018)

Terdapat dua cara untuk mengukur saturasi oksigen dalam darah yaitu *invasive* dan *non-invasive*. Dalam glosarium.com *invasive* merupakan pengukuran saturasi oksigen dengan cara mengambil darah dari arteri kemudian dianalisa memakai alat laboratorium, sedangkan *non-invasive* merupakan suatu tindakan yang dilakukan tanpa memasukkan alat ke dalam tubuh dan tidak menyebabkan kerusakan kulit dan organ tubuh manusia lainnya (Pann, Glosarium, 2019).

Salah satu alat untuk mengukur saturasi oksigen yang beredar di masyarakat yaitu *pulse oximetry*. Cara pengukuran alat ini dengan cara menjepit jari tangan dengan alat oksimetri nadi. Menurut Dr. Whittamore dalam artikel kompas.com yaitu permintaan *pulse oximetry* meningkat sejak pandemi covid-19, sehingga masyarakat harus berhati-hati terhadap *pulse oximetry* yang diperjualbelikan secara bebas di internet baik yang harganya tinggi ataupun rendah serta tidak memiliki sertifikat (Perkasa, 2020). Terdapat

juga kelemahan dari alat pulse oximetri, di dalam website <https://www.alomedika.com/prinsip-kerja-pulse-oximetry-dan-keterbatasannya> “Ada kondisi dimana pulse oximetri tidak dapat diandalkan, yaitu pada pasien dengan saturasi oksigen dibawah 70 persen”. (Fauziah, 2017)



Gambar 1.1  
*Pulse Oximetry*

(Sumber:<https://www.alomedika.com/prinsip-kerja-pulse-oximetry-dan-keterbatasannya>)

Berdasarkan latar belakang yang disampaikan di atas, penulis akan membuat alat pengukur saturasi oksigen dalam darah yang bersifat *non-invasive* menggunakan sensor MAX30100 berbasis arduino. Maka, penulis akan merancang sebuah Tugas Akhir dengan judul “**MENGUKUR SATURASI OKSIGEN DALAM DARAH DENGAN SENSOR MAX30100 BERBASIS ARDUINO**”.

## 1.2 Rumusan Masalah

Hal-hal yang menjadi masalah untuk diselesaikan dalam tugas akhir ini yaitu:

1. Bagaimana cara membuat alat yang dapat mengukur saturasi oksigen dalam darah berbasis arduino?
2. Bagaimana agar alat dapat memberikan informasi mengenai saturasi oksigen dalam darah?

### **1.3 Tujuan Penelitian**

Tujuan penulisan tugas akhir ini adalah sebagai berikut :

1. Membuat alat ukur saturasi oksigen (SpO<sub>2</sub>) dengan memakai sensor MAX30100 berbasis arduino untuk mengukur saturasi oksigen dalam darah.
2. Sebagai salah satu syarat kelulusan program Sarjana di STMIK LIKMI.

### **1.4 Batasan Masalah**

Adapun ruang lingkup bahasan yang akan dibatasi pada laporan pembuatan tugas akhir ini adalah sebagai berikut :

1. Sensor yang digunakan untuk mengukur saturasi oksigen adalah sensor MAX30100.
2. Dalam penelitian ini hanya mengukur saturasi oksigen.
3. Hasil pengukuran saturasi oksigen ditampilkan pada LCD 16x2.

### **1.5 Kegunaan Hasil**

Kegunaan hasil dari penelitian pada tugas akhir ini digunakan untuk:

1. Bidang Kesehatan  
Untuk mengukur saturasi oksigen dalam darah, sehingga dapat mencegah efek buruk yang terjadi dalam tubuh karena kekurangan oksigen.
2. Masyarakat  
Untuk mempermudah pengguna mengetahui kondisi kesehatannya tanpa harus melakukan pengecekan ke rumah sakit serta pengguna dapat mengetahui saturasi oksigennya.

### **1.6 Metodologi Penelitian**

1. Studi Pustaka

Tahap ini merupakan tahap mengumpulkan informasi yang berkaitan dengan penelitian ini, berupa jurnal, buku referensi, artikel, dan berbagai sumber lainnya.



## 2. Interview

Tahap ini merupakan tahap mengumpulkan informasi dengan bertanya kepada perawat dan dokter mengenai penelitian ini.

## 3. Perancangan alat

Tahap ini merupakan tahap perancangan perangkat keras dari komponen-komponen yang dibutuhkan sesuai dengan penelitian.

## 4. Pengujian alat

Pada tahap ini penulis akan menguji alat ukur saturasi oksigen dan akan mencatat setiap hasil yang ada. Pengujian ini akan didiskusikan dengan dosen pembimbing untuk menarik kesimpulan dan memberikan saran pada tugas akhir ini.

### 1.7 Sistematika Penulisan

#### BAB I PENDAHULUAN

Bab ini berisi mengenai latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, kegunaan hasil, metodologi penelitian dan sistematika penulisan.

#### BAB II LANDASAN TEORI

Bab ini berisikan teori saturasi oksigen, darah, mikrokontroler, arduino, sensor, LCD, bahasa pemrograman arduino, OOAD, dan UML yang digunakan dalam perancangan alat ukur saturasi oksigen.

#### BAB III ANALISIS DAN PERANCANGAN

Bab ini berisi mengenai gambaran umum, spesifikasi kebutuhan perangkat keras dan perangkat lunak, serta diagram-diagram alat ukur saturasi oksigen diantaranya *deployment* diagram, *class* diagram, dan *state machine* diagram.

#### BAB IV IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi mengenai implementasi perangkat keras, pengujian sensor MAX30100 dan pengujian sistem.

#### BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari penelitian yang dilakukan dan saran untuk pengembangan lebih lanjut.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Darah**

Darah memiliki istilah lain dari bahasa Yunani yaitu "Haima" yang berarti darah (Novel, 2012:113). Di dalam darah mengandung berbagai komponen, baik yang padat berupa sel-sel darah maupun cairan yang berupa plasma darah. Maka dari itu darah merupakan cairan tubuh yang sangat penting bagi tubuh manusia (Firani, 2018:1). Darah memiliki tiga fungsi utama yaitu untuk (Eduka, 2016:333):

1. Transpor (oksigen dari paru-paru ke seluruh tubuh, karbon dioksida dari seluruh tubuh ke paru-paru, zat makanan, urea menuju ginjal, hormon, panas);
2. Melindungi terhadap patogen;
3. Mengatur temperatur tubuh.

Jumlah darah dalam tubuh manusia kurang lebih 4 liter- 5 liter atau setara dengan sepertiga belas berat badan orang dewasa. Di dalam darah juga terdapat antigen dan antibodi dimana reaksi dua hal tersebut merupakan faktor penentu dari golongan darah (Dewi, 2013:1).

Dalam darah terdapat salah satu komponen yang tidak kalah penting yaitu hemoglobin (Hb), salah satu fungsi dari hemoglobin adalah membawa oksigen yang telah diterima untuk disalurkan ke seluruh tubuh (Adzani, 2019). Untuk mengetahui kadar oksigen yang beredar dalam tubuh dapat diukur melalui saturasi oksigen. Saturasi oksigen merupakan presentasi hemoglobin yang mengikat oksigen dalam arteri (Suherlim et al., 2018:27).

#### **2.2 Saturasi Oksigen**

Oksigen diperlukan untuk menutrisi sel-sel yang hidup di jaringan manusia. Pasokan oksigen yang rendah bisa mengakibatkan kerusakan jaringan manusia, dan bisa juga berujung pada stroke. Saturasi oksigen (SpO<sub>2</sub>) atau dapat disebut juga saturasi

*oxyhemoglobin* (HbO<sub>2</sub>) merupakan rasio dari HbO<sub>2</sub> dengan *deoxyhemoglobin* (Hb) yang menunjukkan jumlah oksigen darah. Oleh karena itu hal ini adalah indeks yang penting untuk sistem sirkulasi manusia. SpO<sub>2</sub> dikategorikan kedalam 3 jenis yaitu:

1. SaO<sub>2</sub> merupakan saturasi HbO<sub>2</sub> pada arteri
2. SpO<sub>2</sub> merupakan saturasi HbO<sub>2</sub> yang dapat dihitung oleh *pulse oximetry*
3. SvO<sub>2</sub> adalah campuran saturasi HbO<sub>2</sub> pada vena.

*Pulse oximetry* merupakan alat yang digunakan untuk mengukur SpO<sub>2</sub>, posisi pengukuran berada di lapisan pembuluh darah kulit daerah perifer yaitu di jari tangan atau kaki, dengan nilai SpO<sub>2</sub> normal mulai dari 95%-98%. Jika tingkat SpO<sub>2</sub> seseorang turun dibawah 90%, tekanan darah akan menurun dan akan membahayakan nyawanya (Tsai et al., 2014:2620). Tingkat kadar SpO<sub>2</sub> tergantung pada kondisi fisik seseorang dan tidak berpengaruh terhadap usia atau jenis kelamin (Tsai et al., 2014:2627). Kondisi dimana kadar oksigen dalam darah rendah dinamakan hipoksemia. Hipoksemia membutuhkan penanganan medis segera karena kondisi ini bisa berlanjut menjadi hipoksia (Adrian, 2019). Pada Tabel 2. 1 merupakan kategori nilai saturasi oksigen yang diinterpretasikan:

Tabel 2. 1  
Kategori Nilai Saturasi Oksigen

Nilai SpO <sub>2</sub>	Interpretasi
95-100%	Normal
91-94%	Hipoksemia Ringan
86-90%	Hipoksemia Sedang
<85%	Hipoksemia Berat

(Sumber: (Jahan et al., 2014))

### 2.3 Mikrokontroler

Menurut Hari Arief Dharmawan dalam buku yang berjudul "*Mikrokontroler: Konsep Dasar dan Praktis*" definisi mikrokontroler adalah:

*"Mikrokontroler merupakan chip mikrokomputer yang secara fisik berupa sebuah IC (Integrated Circuit)." (Dharmawan, 2017:1)*

Mikrokontroler merupakan jenis mikroprosesor yang telah memiliki piranti lengkap seperti memori internal, ADC (*Analog to Digital Converter*), perangkat *input output* (I/O) dalam satu paket IC (Efendi, 2015:6).

Menurut Drs. Imam Mudan, S.T, M.T dalam buku yang berjudul "*Elektronika Dasar*" pengertian mikrokontroler adalah:

*"Mikrokontroler adalah single chip computer yang memiliki kemampuan untuk diprogram dan digunakan untuk tugas-tugas yang berorientasi kontrol."* (Mudan, 2013:149)

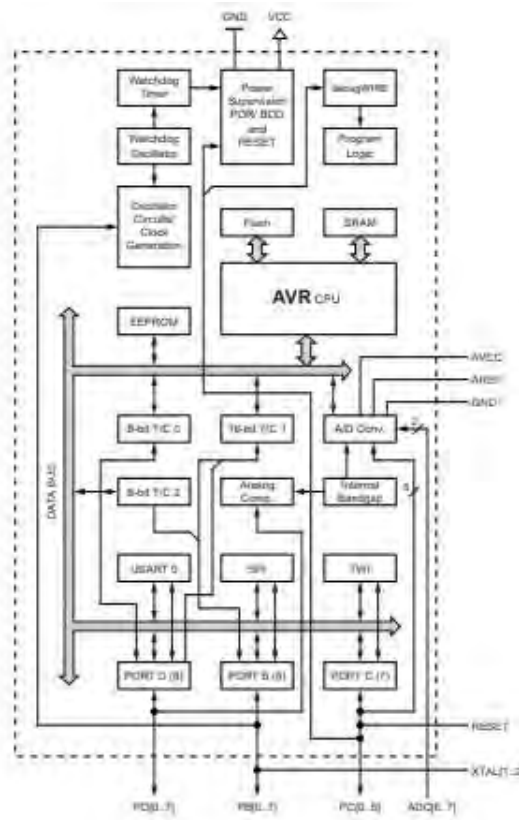
Sehingga dapat disimpulkan bahwa mikrokontroler merupakan sebuah komputer kecil dalam bentuk *chip* yang dikemas dalam sebuah *Integrated Circuit* (IC) yang dirancang untuk melakukan tugas atau operasi tertentu.

### **2.3.1 Mikrokontroler ATmega328**

Mikrokontroler ATmega328 adalah mikrokontroler dengan *Complementary Metal Oxide Semiconductor* (CMOS) 8-bit buatan Atmel yang berbasis arsitektur RISC yang berdaya rendah dan banyak digunakan saat ini.

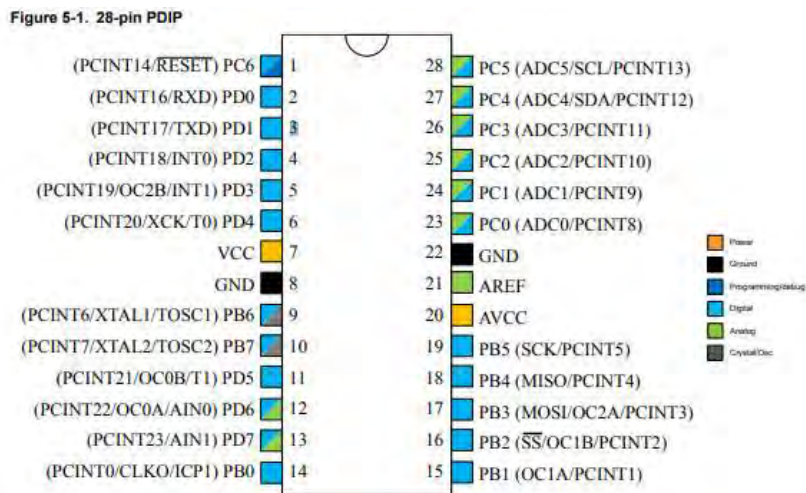
Mikrokontroler Atmega328 menyediakan fitur-fitur sebagai berikut:

1. Memiliki *Flash Memory* 32 Kb
2. Memiliki 131 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus *clock*.
3. Memiliki kecepatan eksekusi mencapai 16 MIPS dengan clock 16 MHz.
4. 32 x 8 register bekerja.
5. Memiliki EEPROM sebesar 1 Kb sebagai tempat penyimpanan data semi permanen.
6. Memiliki SRAM (*Static Random Access Memory*) sebesar 2 Kb.



Gambar 1.2  
Blok Diagram Mikrokontroler ATmega328  
(Sumber: Datasheet Atmega328)

### 2.3.2 Konfigurasi Pin ATmega328



Gambar 1.3  
Konfigurasi Pin ATmega328P  
(Sumber : Atmel, 2016:9)

Setiap pin mikrokontroler ATmega328 memiliki fungsinya masing-masing, berikut rincian dan fungsi dari susunan pin ATmega328P adalah (Atmel, 2016:12-13):

1. VCC merupakan pin yang berfungsi sebagai jalan masuknya catu daya listrik.
2. GND merupakan pin *ground*.
3. *Port B* (PB0-PB7) adalah *port* masukan/keluaran dua arah 8-bit dengan resistor internal. *Buffer* keluaran *port B* memiliki karakteristik penggerak simetris dengan kemampuan sumber yang tinggi.
4. *Port C* adalah *port* masukan/keluaran dua arah 7-bit dengan resistor *pull-up* internal. *Buffer* keluaran *PC* memiliki karakteristik penggerak simetris.
5. PC6 / *Reset* merupakan pin yang digunakan untuk mengatur ulang mikrokontroler.
6. *Port D* adalah *port* masukan/keluaran dua arah 8-bit dengan resistor *pull-up* internal. *Buffer* keluaran *port D* memiliki karakteristik penggerak simetris..
7. AVCC adalah pin tegangan untuk menyuplai konverter A/D, PC[3:0], dan PE[3:2]. Ini harus terhubung secara eksternal ke VCC, meskipun ADC tidak digunakan, ADC harus dihubungkan ke VCC melalui filter *low-pass*.
8. AREF adalah pin referensi analog untuk konverter A/D.
9. ADC (TQFP dan VFQFN) adalah pin yang berfungsi sebagai input analog ke konverter A/D. pin ini diberi daya dari suplai analog dan berfungsi sebagai saluran ADC 10-bit.

#### 2.4 Sensor

Menurut Setyawan P. Sakti dalam buku yang berjudul "*Pengantar Teknologi Sensor: Prinsip Dasar Sensor Besaran Mekanik*" definisi sensor adalah:

*"Sensor merupakan salah satu produk teknologi yang telah lama dikembangkan dan terus berkembang seiring dengan perkembangan teknologi yang lain."* (Sakti, 2017:1)

Dalam buku yang berjudul "*Buku Ajar Pengukuran dan Instrumentasi*" yang ditulis oleh Abadi Jading, Reniana, dan Bertha Ollin Paga pengertian sensor adalah:

“Sensor adalah suatu peralatan yang berfungsi untuk mendeteksi, mengukur sesuatu nilai besaran dalam bentuk energi listrik, fisika, kimia, biologi, energi mekanik, dan sebagainya. Sensor meniru panca indra manusia (mata, telinga, hidung, kulit, dan lidah).” (Jading et al., 2020:103)

Berdasarkan definisi di atas, dapat disimpulkan bahwa sensor merupakan suatu alat yang sangat beragam tergantung nilai besaran yang akan diukur dan sebuah teknologi yang akan terus dikembangkan.

Sensor MAX30100 memiliki sensor suhu on-chip untuk (opsional) mengkalibrasi ketergantungan suhu dari subsistem SpO<sub>2</sub>. Sensor MAX30100 merupakan sensor yang digunakan untuk mengukur saturasi oksigen (SpO<sub>2</sub>) dalam darah dan mengukur detak jantung dengan tidak memasukkan alat ke dalam tubuh atau dapat disebut juga dengan cara *non-invasive*. Sensor MAX30100 merupakan modul yang terdiri dari LED merah, LED inframerah dan fotodiode serta merupakan integrasi dari *pulse oximetry*.

MAX30100 dapat mengkonfigurasi arus LED yang dapat di program dari 0mA hingga 50 mA dan resolusi ADC dari 13 bit hingga 16 bit. Serta data keluaran digital yang disimpan di FIFO dapat menampung 16 sampel. FIFO memungkinkan sensor MAX30100 untuk dihubungkan ke mikrokontroler atau mikroprosesor pada bus bersama, di mana data tidak sedang dibaca terus menerus dari register perangkat



Gambar 2.1  
Sensor MAX30100

(Sumber: <http://www.jogjarobotika.com/sensor-lainnya/2054-max30100-pulse-oximeter-heart-rate-sensor-module.html>)

## 2.5 LCD (*Liquid Crystal Display*)

Pengertian LCD menurut Widodo Budiharto dalam buku berjudul *“Panduan Praktikum Mikrokontroler”* adalah:

*“LCD (Liquid Crystal Display) ialah modul penampil yang banyak digunakan karena tampilannya menarik.”* (Budiharto, 2013:44)

LCD (*Liquid Crystal Display*) merupakan salah satu jenis media yang menggunakan *liquid crystal* sebagai tampilan utama. LCD berfungsi untuk menampilkan status pengoperasian instrumen (Pane et al., 2020:47).

LCD 16x2 merupakan LCD yang mempunyai batas maksimum 16 digit horizontal dan 2 digit vertikal atau bisa disebut juga 16 baris dan 2 kolom (Parastiwi et al., 2018:114).

Berdasarkan definisi di atas, dapat disimpulkan bahwa LCD merupakan salah satu jenis media atau modul yang berfungsi untuk menampilkan keluaran dari suatu proses dengan keluaran berupa karakter dan angka.



Gambar 2.2  
16x2 LCD Module

(Sumber: <https://components101.com/16x2-lcd-pinout-datasheet>)



Tabel 2. 2  
Pin Configuration

Pin No	Pin Name	Description
1	Vss (Ground)	Ground pin connected to system ground
2	Vdd (+5 Volt)	Powers the LCD with +5V (4.7V – 5.3V)
3	VE (Contrast V)	Decide the contrast level of display. Grounded to get maximum contrast.
4	Register Select	Connected to Microcontroller to shift between command/data register
5	Read/Write	Used to read or write data. Normally grounded to write data to LCD
6	Enable	Connected to Microcontroller Pin and toggled between 1 and 0 for data acknowledgement
7	Data Pin 0	Data pins 0 to 7 forms a 8-bit data line. They can be connected to Microcontroller to send 8-bit data. These LCD's can also operate on 4-bit mode in such case Data pin 4,5,6 and 7 will be left free.
8	Data Pin 1	
9	Data Pin 2	
10	Data Pin 3	
11	Data Pin 4	
12	Data Pin 5	
13	Data Pin 6	
14	Data Pin 7	
15	LED Positive	Backlight LED pin positive terminal
16	LED Negative	Backlight LED pin negative terminal

(Sumber: <https://components101.com/16x2-lcd-pinout-datasheet>)

Terdapat fitur dari modul LCD 16x2 adalah sebagai berikut

(<https://components101.com/16x2-lcd-pinout-datasheet>):

1. Membutuhkan tegangan sebesar 4,7V – 5,3V
2. Mengonsumsi arus sebanyak 1mA tanpa lampu latar
3. Tampilan modul LCD alfanumerik, artinya dapat menampilkan huruf dan angka

4. Terdiri dari 2 baris dan setiap baris dapat mencetak 16 karakter
5. Setiap karakter dibangun dengan kotak 5 x 8 piksel
6. Dapat bekerja pada mode 8-bit dan 4-bit
7. Dapat juga menampilkan karakter yang dibuat khusus
8. Tersedia dalam lampu latar hijau dan biru

## 2.6 Arduino

Arduino merupakan perangkat keras sekaligus perangkat lunak yang memungkinkan siapa saja melakukan pembuatan prototipe suatu rangkaian elektronika yang berbasis mikrokontroler dengan mudah dan cepat (Kadir, 2016:2).

Arduino menurut Becky Stewart dalam buku *“Adventures in Arduino”* adalah:

*“The arduino is a tool for building computers that can interact with the physical world around you.”* (Stewart, 2015:1)

Arduino adalah sebuah piranti elektronik yang menggunakan chip mikrokontroler Atmega 328 yang terdiri dari CPU, *memory*, dan I/O yang bisa di kontrol dengan memprogramnya. I/O juga sering disebut dengan GPIO (*General Purpose Input Output Pins*) yang berarti : pin yang bisa diprogram sebagai input atau output sesuai kebutuhan (Saputra, 2019:23). Spesifikasi Arduino Uno ditampilkan dalam tabel 2.3 sebagai berikut:

Tabel 2. 3  
Spesifikasi papan Arduino Uno

<i>Microcontroller</i>	Atmega328P-8 bit AVR family microcontroller
<i>Operating Voltage</i>	5V
<i>Recommended Input Voltage</i>	7-12V
<i>Input Voltage Limits</i>	6-20V
<i>Analog Input Pins</i>	6 (A0-A5)
<i>Digital I/O Pins</i>	14 (out of which 6 provide PWM output)
<i>DC Current on I/O Pins</i>	40 mA
<i>DC Current on 3.3V Pin</i>	50 mA
<i>Flash Memory</i>	32 KB (0.5 KB is used for Bootloader)

SRAM	2 KB
EEPROM	1 KB
<i>Frequency (Clock Speed)</i>	16 MHz

(Sumber: <https://components101.com/microcontrollers/arduino-uno>)

## 2.7 Arduino IDE (*Integrated Development Environment*)

Pengertian arduino IDE menurut Fauzan dan Adiputri dalam buku yang berjudul “*Tutorial Membuat Prototipe Prediksi Ketinggian Air (PKA) Untuk Pendeteksi Banjir Peringatan Dini Berbasis IoT*” adalah:

“*Arduino IDE (Integrated Development Environment) adalah software yang digunakan untuk memprogram di arduino, dengan kata lain Arduino IDE sebagai media untuk memprogram board arduino.*” (Fauzan, 2020:5)

Pada umumnya editor programming memiliki fitur untuk *cut* atau *paste* dan *find* atau *replace* teks, demikian juga pada arduino IDE. Untuk lebih jelasnya dibawah ini akan dijelaskan fungsi toolbar dalam *software* ini (Fauzan, 2020:6-8).



Gambar 2.3  
Tampilan Arduino IDE

(Sumber: (Fauzan, 2020:7))

1. *Verify* untuk memverifikasi terlebih dahulu *sketch* yang dibuat, jika ada kesalahan pada *sketch* nanti akan muncul *error*.
2. *Upload* untuk mengupload *sketch* ke *board* arduino. walaupun tidak mengklik tombol *verify*, maka *sketch* akan di-compile, kemudian langsung di *upload* ke *board*.
3. *New Sketch* untuk membuka *window* dan membuat *sketch* baru.
4. *Open Sketch* untuk membuka *sketch* yang sudah pernah dibuat.
5. *Save Sketch* untuk menyimpan *sketch*, tapi tidak disertai dengan mengkompilasi.
6. Serial Monitor untuk membuka *interface* untuk komunikasi serial
7. Keterangan Aplikasi pesan-pesan yang dilakukan aplikasi akan muncul di sini
8. Konsol Log pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini
9. Baris *Sketch* bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*
10. Informasi *Board* dan *Port* bagian ini menginformasikan *port* yang dipakai oleh *board* arduino.

## 2.8 Bahasa Pemrograman C

Bahasa C pertama kali dikembangkan pada tahun 1967 oleh Martin Richards yang awalnya adalah bahasa BCPL, lalu pada tahun 1970 Ken Thomas mengembangkannya menjadi bahasa B kemudian dikembangkan kembali oleh Dennis Ritchie menjadi bahasa C di *Bell Telephone Laboratories Inc.* pada tahun 1970-an. Komputer Digital *Equipment Corporation* PDP-11 merupakan yang pertama kali menggunakan bahasa C di sistem operasi UNIX C (Iswanto, 2015:31).

Bahasa C ini istimewa karena dapat mendukung Pemrograman Berorientasi Objek (OOP/*Object Oriented Programming*) serta PHP dan java memakai sintaks yang mirip dengan bahasa C. Arduino IDE merupakan salah satu *software* yang dapat digunakan untuk memprogram bahasa C (Fauzan, 2020:4).

Dapat disimpulkan bahwa bahasa pemrograman C merupakan bahasa pemrograman yang dapat digunakan untuk membuat berbagai aplikasi dan arduino IDE merupakan salah satu software untuk memprogram bahasa C.

## **2.9 OOAD (*Object Oriented Analysis and Design*)**

Teknik pemodelan arsitektur pada dasarnya digolongkan ke dalam dua cara yaitu pemodelan terstruktur dan pemodelan berbasis objek (*Object Oriented Analysis and Design* atau yang disingkat OOAD). Pendekatan yang digunakan oleh pemodelan berbasis objek yaitu terhadap masalah dari perspektif objek, sedangkan pemodelan terstruktur berorientasi terhadap proses dan komponen pemrograman terstruktur (Sulianta, 2019:229).

OOAD menurut Raul Sidnei Wazlawick dalam bukunya yang berjudul "*Object-Oriented Analysis and Design for Information Systems*" adalah sebagai berikut :

*"Object-oriented analysis and design is taught and practiced in some places, it can be concluded that many professionals simply adopt an object-oriented programming language, or use fragments of an object-oriented based development process."*  
(Wazlawick, 2014:2)

Berdasarkan definisi di atas, dapat disimpulkan bahwa *object-oriented analysis and design* merupakan suatu metode analisis yang memeriksa permasalahan dari kumpulan objek yang saling berinteraksi dimana objek tersebut mewakili beberapa entitas, seperti kelas, elemen data dan perilaku dari objek tersebut .

## **2.10 UML (*Unified Modeling Language*)**

Dalam buku yang berjudul "*UML @Classroom: An Introduction to Object-Oriented Modeling*" yang ditulis oleh Martina Seidl, Marion Scholz, Christian Huemer, dan Gerti Kappel pengertian UML sebagai berikut :

*"UML is a general purpose modeling language, meaning that its use is not restricted to a specific application area."* (Seidl, 2015:14)

Sedangkan menurut Vivian Siahaan dan Rismon Hasiholan Sianipar dalam buku yang berjudul "*Buku Pintar Pemrograman C# Untuk Pelajar dan Mahasiswa*" UML adalah:

*"UML merupakan bahasa grafis yang digunakan programmer untuk mempresentasikan sistem berorientasi objek dengan cara yang terstandarisasi."*  
(Siahaan , 2020:35)

Sehingga dapat disimpulkan bahwa UML merupakan sebuah metode pemodelan yang digunakan oleh programmer untuk mempresentasikan aplikasi dalam perancangan sistem berorientasi objek.

### **2.10.1 Use Case Diagram**

Pengertian *use case diagram* menurut Ian Sommerville dalam buku berjudul "*Software Engineering, Tenth Edition*" adalah:

*"A UML diagram type that is used to identify use-case and graphically depict the users involved. It must be supplemented with additional information to completely describe use-cases."* (Sommerville, 2016:773)

Pengertian *use case diagram* menurut Rod Stephens dalam buku yang berjudul "*Beginning Software Engineering*" adalah:

*"In UML, a diagram that represents a user's interaction with the system. Use case diagram show stick figures representing actors (someone or something that performs a task) connected to tasks represented by ellipses."* (Stephens, 2012:436)

Pengertian *use case diagram* menurut Hassan Gomaa dalam buku yang berjudul "*Software Modeling & Design*" adalah:

*"A use case defines a sequence of interactions between the actor and the system."* (Gomaa, 2011:15)

Pengertian *use case diagram* menurut Rosa A. Sukamto dan M. Shalahuddin dalam buku yang berjudul "*Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*" adalah:

“Use case atau diagram use case merupakan pemodelan untuk kelakuan (*behavior*) yang akan dibuat, use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat” (Sukamto, 2013:155)

Berdasarkan definisi diatas, dapat disimpulkan bahwa use case diagram merupakan diagram yang merepresentasikan interaksi pengguna dengan sistem.

### 2.10.2 Class Diagram

Menurut Ian Sommerville dalam buku “*Software Engineering, Tenth Edition*” menuliskan bahwa definisi dari diagram kelas adalah:

“*Class diagrams are used when developing an object-oriented system model to show the classes in a system and the associations between these classes.*” (Sommerville, 2016:149)

Diagram kelas atau *class diagram* merupakan penggambaran struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki dua pembagian yaitu atribut dan operasi atau metode, berikut merupakan pembagian dari kelas (Sukamto, 2013:141):

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Berdasarkan definisi diatas, dapat disimpulkan bahwa *class diagram* merupakan struktur sistem berorientasi objek yang menunjukkan kelas-kelas dalam sistem asosiasi antar kelas.

### 2.10.3 Deployment Diagram

Pengertian *deployment diagram* dalam buku yang berjudul “*Memahami Metode Omax dan Promethee pada Sistem Pendukung Keputusan*” yang ditulis oleh Lusiana V. Aprillian, Muhammad Y. H. Setyawan, dan Mohamad H. K. Saputra adalah:

“*Deployment diagram merupakan suatu tampilan atau kinerja diri sebuah sistem yang baru sesuai dengan perancangan data yang diambil dari beberapa objek.*” (Aprillian et al., 2020:99)

Sedangkan menurut Yurindra dalam buku yang berjudul “Software Engineering” definisinya adalah sebagai berikut:

“*Deployment* diagram merupakan gambaran proses-proses berbeda pada suatu sistem yang berjalan dan bagaimana relasi di dalamnya.” (Yurindra, 2017:32)

Pengertian *deployment* diagram menurut Hassan Gomaa dalam buku yang berjudul “*Software Modeling & Design*” adalah:

“*A deployment diagram shows the physical configuration of the system in terms of physical nodes and physical connections between the nodes, such as network connections.*” (Gomaa, 2011:23)

Sehingga dapat disimpulkan bahwa *deployment diagram* merupakan sebuah struktur diagram yang menggambarkan proses-proses dari sebuah sistem untuk menampilkan bagian-bagian aplikasi pada perangkat keras.

#### **2.10.4 State Machine Diagram**

Pengertian *state machine* diagram menurut Rod Stephens dalam buku yang berjudul “*Beginning Software Engineering*” adalah:

“*A state machine diagram shows the states through which an object passes in response to various events. States are represented by rounded rectangles.*” (Stephens, 2012:112)

Menurut Hassan Gomaa dalam buku yang berjudul “*Software Modeling & Design*” definisi state machine diagram adalah sebagai berikut:

“*A UML depiction of a finite state machine or statechart.*” (Gomaa, 2011:534)

Sehingga dapat disimpulkan bahwa *state machine diagram* merupakan penggambaran status yang dilalui suatu objek sebagai respon dari masalah yang ada.



## BAB III

### ANALISIS DAN PERANCANGAN

#### 3.1 Gambaran Umum

Alat ukur saturasi oksigen merupakan alat yang digunakan untuk memonitor kadar saturasi oksigen (SpO<sub>2</sub>) dalam darah. Alat ini dibuat dengan menggunakan sensor MAX30100 sebagai sensor pendeteksi saturasi oksigen serta detak jantung pada manusia, sensor ini menggabungkan dua LED yaitu inframerah dan merah. Alat ini juga menggunakan papan Arduino Uno yang dilengkapi dengan mikrokontroler Atmega328 yang berfungsi untuk mengolah data masukan dari sensor dan LCD yang berperan untuk menampilkan hasil keluaran dari sensor.

Cara kerja dari alat ukur kadar oksigen ini adalah dengan cara meletakkan jari diatas sensor, kemudian cahaya dari LED inframerah akan menyerap hemoglobin yang mengandung oksigen, sedangkan LED merah hanya akan menyerap hemoglobin yang tidak mengandung oksigen. Setelah proses tersebut data yang didapat akan ditampilkan pada LCD. Jika saturasi oksigen berada di angka 95-100% maka akan masuk dalam kategori normal, apabila 91-94% maka masuk dalam kategori hipoksemia ringan, apabila 86-90% maka masuk dalam kategori hipoksemia sedang, sedangkan <85% masuk dalam kategori hipoksemia berat.

Tabel 3. 1  
Kategori Nilai Saturasi Oksigen

Nilai SpO <sub>2</sub>	Interpretasi
95-100%	Normal
91-94%	Hipoksemia Ringan
86-90%	Hipoksemia Sedang
<85%	Hipoksemia Berat

(Sumber: (Jahan et al., 2014))

#### 3.2 Spesifikasi Kebutuhan

##### 3.2.1 Kebutuhan Perangkat Keras

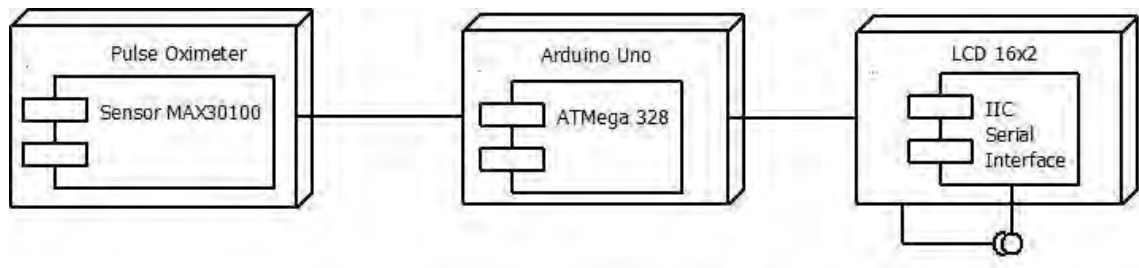
Adapun kebutuhan perangkat keras yang digunakan adalah:

1. Papan Arduino Uno
2. Sensor MAX30100
3. LCD (*Liquid Crystal Display*) 16 x 2 karakter.

### 3.2.2 Kebutuhan Perangkat Lunak

1. Arduino IDE: dibutuhkan untuk membuat program yang akan diunggah ke papan Arduino UNO.
2. Windows 10: sistem operasi dari laptop.

### 3.3 Deployment Diagram



Gambar 3.1  
Deployment Diagram Alat ukur saturasi oksigen

Penjelasan pada Gambar 3. 1 *deployment* diagram yang digunakan pada alat ukur saturasi oksigen adalah sebagai berikut:

1. Sensor MAX30100

Sensor MAX30100 adalah sensor yang akan mendeteksi saturasi oksigen dan detak jantung. Sensor ini mendeteksi dengan cara menaruh jari diatas sensor.

2. Papan Arduino Uno

Papan Arduino Uno yang digunakan pada alat ukur ini berfungsi sebagai pusat pengendali yang mengolah data dari sensor dengan menggunakan mikrokontroler Atmega328.

3. LCD (*Liquid Crystal Display*)

LCD digunakan untuk menampilkan hasil dari pengukuran saturasi oksigen dalam darah. Data yang ditampilkan pada LCD berupa hasil pengukuran saturasi oksigen dalam bentuk persen serta pengukuran detak jantung.

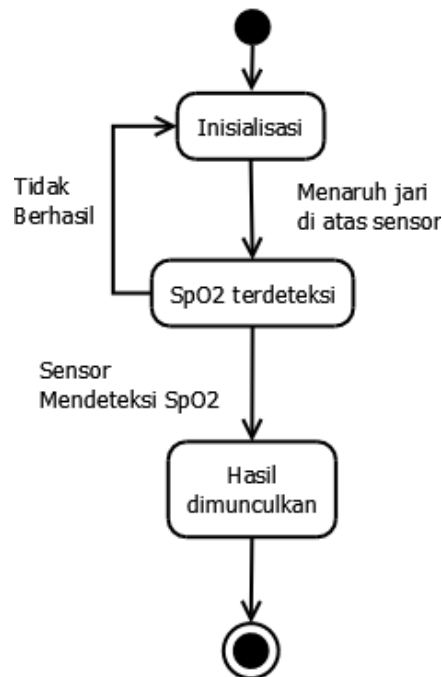
### 3.4 Class Diagram



Gambar 3.2  
Class Diagram Alat Ukur saturasi oksigen

Gambar 3.2 merupakan diagram kelas alat ukur saturasi oksigen yang dirancang. diagram kelas ini memiliki 3 kelas yaitu sensor MAX30100, Arduino Uno, dan LCD. Masing-masing dari kelas tersebut memiliki metode dan atribut berupa pin-pin yang akan dipakai.

### 3.5 State Machine Diagram

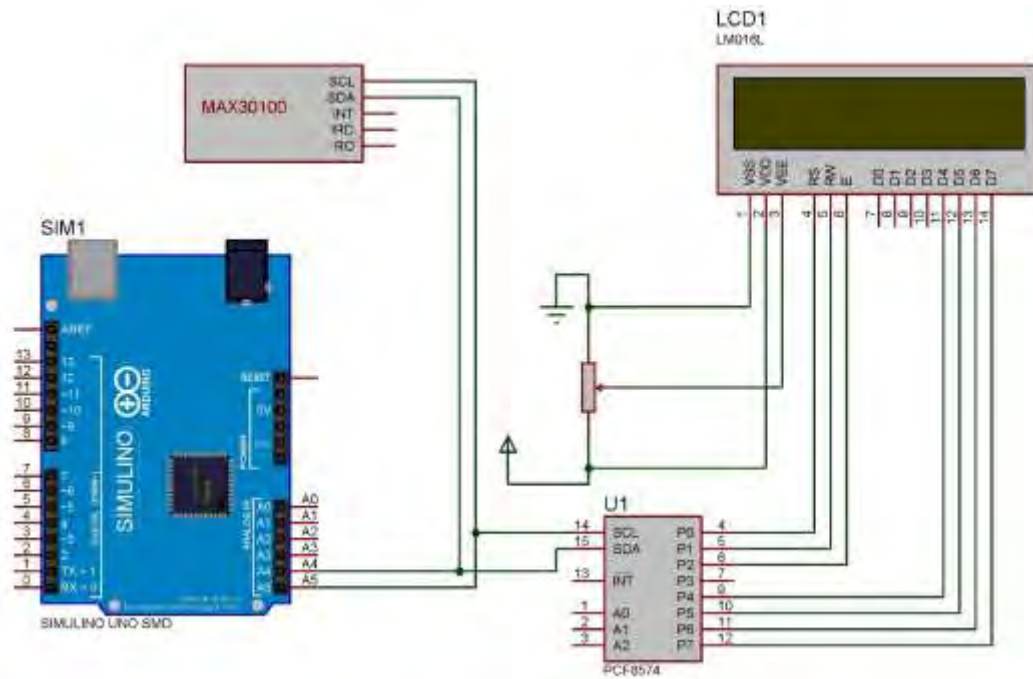


Gambar 3.3  
State Machine Diagram Alat Ukur saturasi oksigen

Gambar 3.3 merupakan *state machine* diagram alat ukur saturasi oksigen. Pertama, sistem akan menunjukkan status inisialisasi, setelah sistem sudah siap digunakan sistem akan menunggu masukan berupa jari yang diletakkan di atas sensor

tersebut. Setelah itu LED inframerah yang terdapat dalam sensor akan mendeteksi SpO<sub>2</sub> yang terdapat dalam darah. Jika SpO<sub>2</sub> gagal terdeteksi, maka status sistem akan kembali ke status inialisasi. Jika SpO<sub>2</sub> terdeteksi, maka hasil SpO<sub>2</sub> ditampilkan melalui LCD.

### 3.6 Skematik Rancangan



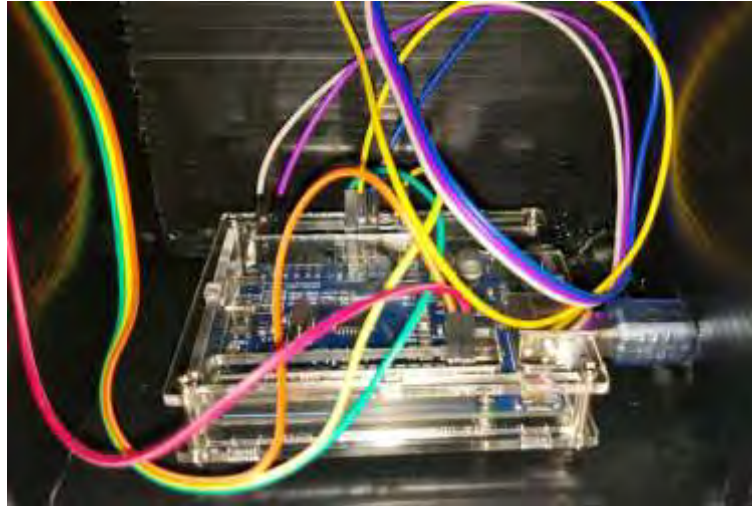
Gambar 3.4  
Skematik Rancangan

## BAB IV

### IMPLEMENTASI DAN PENGUJIAN

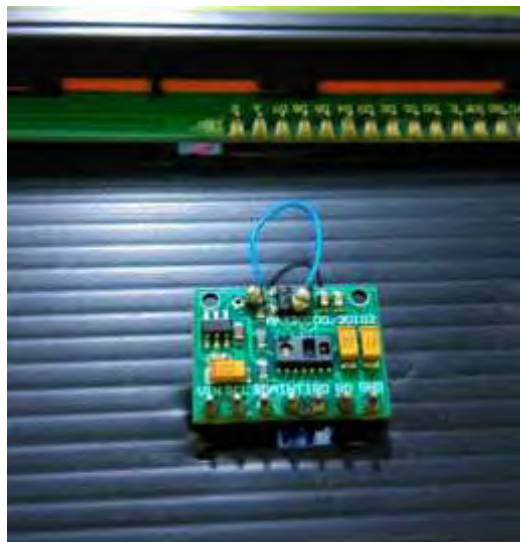
#### 4.1 Implementasi antar Muka Perangkat Keras

Gambar 4.1 merupakan foto papan Arduino Uno yang sudah dipasang di dalam kotak.



Gambar 4.1  
Papan Arduino Uno

Gambar 4.2 merupakan foto sensor MAX30100 yang sudah dipasang di atas kotak dan sudah dihubungkan dengan papan Arduino.



Gambar 4.2  
Sensor MAX30100

Gambar 4.3 adalah foto LCD (*Liquid Crystal Display*) yang sudah dipasang diatas kotak.



Gambar 4.3  
LCD (*Liquid Crystal Display*)

Gambar 4.4 merupakan foto hasil akhir dari alat ukur saturasi oksigen berbasis Arduino. Perancangan alat ini menggunakan satu buah papan Arduino Uno, satu buah sensor MAX30100, dan sebuah LCD.



Gambar 4.4  
Alat ukur saturasi oksigen berbasis Arduino

## 4.2 Pengujian

### 4.2.1 Pengujian Sensor MAX30100

Pada sub bab ini akan membahas mengenai pengujian modul dari sensor MAX30100 yang digunakan untuk mendeteksi saturasi oksigen dalam darah dan hasil pengukuran akan muncul pada LCD. Pada gambar 4.5 saturasi oksigen yang dibaca oleh sensor MAX30100 sebesar 99% yang berarti berada di kondisi normal, dan pada alat pulse oximetry saturasi oksigen sebesar 98%. Berikut tampilan saturasi oksigen pada LCD:



Gambar 4.5  
Hasil pengujian Sensor MAX30100

### 4.3.1 Pengujian Sistem

Pengujian sistem ini bertujuan untuk mengetahui apakah hasil saturasi oksigen dari sensor MAX30100 akurat atau tidak dengan menggunakan *pulse oximetry* sebagai alat pembanding saturasi oksigen dalam persen (%).

Tabel 4. 1  
Tabel Alat Ukur Saturasi Oksigen

Nama	Usia	Jenis Kelamin	Berat Badan	Kegiatan	SpO2 dari Sensor MAX30100	SpO2 dari Pulse Oximetry	Selisih	Status pada LCD	Status pada tabel	Hasil Pengujian
Ivone	21	P	47 kg	Tidak berolahraga	98%	98%	0	Normal	Normal	Sesuai
Angel	22	P	44 kg	Tidak berolahraga	99%	98%	0.01	Normal	Normal	Sesuai
Yulius	43	L	70 kg	Tidak berolahraga	99%	96%	0.03	Normal	Normal	Sesuai
Rocky	32	L	110 kg	Tidak berolahraga	98%	92%	0.06	Normal	Normal	Sesuai
Clairine	23	P	40 kg	Tidak berolahraga	98%	98%	0	Normal	Normal	Sesuai
Stephanny	21	P	55 kg	Tidak berolahraga	99%	98%	0.01	Normal	Normal	Sesuai
Santi	54	P	54 kg	Tidak berolahraga	97%	97%	0	Normal	Normal	Sesuai
Yosep	20	L	83 kg	Tidak berolahraga	98%	93%	0.05	Normal	Normal	Sesuai
Denis	34	L	90 kg	Tidak berolahraga	98%	96%	0.02	Normal	Normal	Sesuai
Honey	48	P	48 kg	Tidak berolahraga	98%	98%	0	Normal	Normal	Sesuai
Enrico	25	L	59 kg	Tidak berolahraga	98%	96%	0.02	Normal	Normal	Sesuai
Yoses	26	L	64 kg	Tidak berolahraga	98%	98%	0	Normal	Normal	Sesuai
Christa	15	P	43 kg	Tidak berolahraga	98%	98%	0	Normal	Normal	Sesuai
Wildan	24	L	50 kg	Tidak berolahraga	98%	93%	0.05	Normal	Normal	Sesuai
Grace	23	P	52 kg	berolahraga	98%	96%	0.02	Normal	Normal	Sesuai
Karin	22	P	40 kg	Sebelum berolahraga	97%	98%	0.01	Normal	Normal	Sesuai
				Setelah berolahraga	98%	98%	0	Normal	Normal	Sesuai
Yanto	56	L	71 kg	Sebelum berolahraga	98%	95%	0.03	Normal	Normal	Sesuai
				Setelah berolahraga	97%	96%	0.01	Normal	Normal	Sesuai
Christian	25	L	64 kg	Sebelum berolahraga	98%	97%	0.01	Normal	Normal	Sesuai
				Setelah berolahraga	97%	96%	0.01	Normal	Normal	Sesuai
Richard	21	L	61 kg	Sebelum berolahraga	98%	98%	0	Normal	Normal	Sesuai
				Setelah berolahraga	97%	95%	0.02	Normal	Normal	Sesuai
Garry	23	L	56 kg	Sebelum berolahraga	99%	98%	0.01	Normal	Normal	Sesuai
				Setelah berolahraga	99%	98%	0.01	Normal	Normal	Sesuai
Ardi	25	L	63 kg	Sebelum berolahraga	98%	98%	0	Normal	Normal	Sesuai



Nama	Usia	Jenis Kelamin	Berat Badan	Kegiatan	SpO2 dari Sensor MAX30100	SpO2 dari Pulse Oximetry	Selisih	Status pada LCD	Status pada tabel	Hasil Pengujian
				Setelah berolahraga	98%	98%	0	Normal	Normal	Sesuai
Jeremy	24	L	55 kg	Sebelum berolahraga	99%	98%	0.01	Normal	Normal	Sesuai
				Setelah berolahraga	93%	95%	0.02	Hipoksemia Ringan	Hipoksemia Ringan	Sesuai
Ferdy	25	L	53 kg	Sebelum berolahraga	98%	90%	0.08	Normal	Normal	Sesuai
				Setelah berolahraga	95%	97%	0.02	Normal	Normal	Sesuai
Timotius	25	L	65 kg	Sebelum berolahraga	98%	95%	0.03	Normal	Normal	Sesuai
				Setelah berolahraga	98%	95%	0.03	Normal	Normal	Sesuai
<i>Error Rate</i>							0.01834%			

Jenis kelamin "L" untuk laki-laki sedangkan "P" untuk perempuan. Selisih yang tertera pada tabel 4.1 merupakan selisih dari angka saturasi oksigen dari sensor MAX30100 dan dari *pulse oximetry*. Status pada LCD merupakan status yang dihasilkan dari sensor ketika pengujian dilakukan sedangkan keterangan pada tabel merupakan keterangan yang diperoleh berdasarkan tabel dari sumber resmi.

Hasil pengujian yang tertera pada tabel 4.1 adalah hasil perbandingan yang diperoleh antara status yang ditampilkan di LCD dengan status dari tabel sumber resmi apakah sesuai atau tidak sesuai. *Error rate* adalah hasil perhitungan dari rumus berikut:

$$\text{Error rate \%} = \frac{|\text{Hasil Sensor MAX30100} - \text{Hasil Pulse oximetry}|}{\text{Hasil Pulse oximetry}} \times 100$$

Kesimpulan dari hasil pengujian yang sudah dilakukan adalah dalam pengukuran saturasi oksigen ini usia, berat badan, jenis kelamin, maupun kegiatan yang dilakukan tidak mempengaruhi tingkat saturasi oksigen. Namun pada saat dilakukannya tes saturasi oksigen ini penempatan jari pada sensor berpengaruh terhadap angka keluaran yang dihasilkan dan ukuran, pergerakan, maupun ketebalan kulit pada jari pun berpengaruh terhadap angka keluaran yang dihasilkan.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Kesimpulan yang dapat diperoleh berdasarkan hasil pengujian yang sudah dilakukan adalah sebagai berikut:

1. Tingkat saturasi oksigen tidak terpengaruh terhadap usia, berat badan, jenis kelamin dan kegiatan yang dilakukan.
2. Ketika melakukan pengujian saturasi oksigen penempatan dan pergerakan jari pada sensor dapat mempengaruhi angka saturasi oksigen yang dihasilkan.

#### **5.2 Saran**

Perancangan alat ukur saturasi oksigen ini masih memiliki banyak kekurangan dan jauh dari sempurna sehingga alat ini butuh untuk dikembangkan dengan menambahkan fitur-fitur yang lain. Oleh karena itu, penulis menyampaikan saran-saran sebagai berikut:

1. Mempertimbangkan faktor-faktor lain yang mempengaruhi saturasi oksigen seperti gangguan pernapasan atau penyakit lainnya serta faktor dari lingkungan.
2. Perlunya mendesain tempat sensor dengan lebih baik agar saat pengukuran hasil pembacaan akurat.

## DAFTAR PUSTAKA

- Adrian, K. (2019). *Hipoksemia, Ketika Darah Kekurangan Oksigen*. <https://www.alodokter.com/penyebab-hipoksemia-dan-cara-mengukurnya>
- Adzani, F. (2019). *Mengenal Fungsi Hemoglobin, Si "Pengantar" Oksigen ke Seluruh Tubuh*.
- Aprilian, L. V., Setyawan, M. Y. H., & Saputra, M. H. K. (2020). *Memahami Metode Omax dan Promethee pada Sistem Pendukung Keputusan*. CV. Kreatif Industri Nusantara.
- Atmel. (2016). *ATmega328/P Summary*. 9.
- Budiharto, W. (2013). *Panduan Praktikum Mikrokontroler AVR ATmega16*. Elex Media Komputindo.
- Dewi, H. A. C. (2013). *Makanan Berbahaya Bagi Golongan Darah A: Kesehatan*. Pt.Niaga Swadaya.
- Dharmawan, H. A. (2017). *Mikrokontroler: Konsep Dasar dan Praktis*. Universitas Brawijaya Press.
- Eduka, T. C. (2016). *SKM (Sukses Kuasai Materi) SMA Kelas XI*. Gramedia Widiasarana Indonesia.
- Efendi, B. (2015). *Dasar Mikrokontroler Atmega8535 dengan CAVR*. Deepublish.
- Fauzan, M. N., & Adiputri, L. C. (2020). *Tutorial Membuat Prototipe Prediksi Ketinggian Air (Pka) Untuk Pendeteksi Banjir Peringatan Dini Berbasis IoT*. Kreatif.
- Firani, N. K. (2018). *Mengetahui Sel-Sel darah dan Kelainan Darah*. UB Press.
- Gomaa, H. (2011). Software modeling & design. In *ACM SIGSOFT Software Engineering Notes* (Vol. 36, Issue 4). Cambridge University Press. <https://doi.org/10.1145/1988997.1989008>
- Iswanto, & Raharja, N. M. (2015). *Mikrokontroler: Teori dan Praktik Atmega 16 dengan Bahasa C*. Deepublish.
- Jading, A., Reniana, & Paga, B. O. (2020). *Buku Ajar Pengukuran Dan Instrumentasi*. Deepublish.
- Jahan, E., Barua, T., & Salma, U. (2014). an Overview on Heart Rate Monitoring. *International J. of Latest Research in Science and Technology*, 3(5), 148–152.
- Kadir, A. (2016). *Simulasi Arduino*. Elex Media Komputindo.
- Mudan, I. (2013). *Elektronik Dasar*. Gunung Samudera.
- Novel, S. S. (2012). *Superlengkap biologi SMA*. Gagas Media.
- Pane, S. F., Lase, F. S., & Mali, O. B. (2020). *Smart Conveyor Pada Outbound Dengan Arduino*. Kreatif.
- Parastiwi, A., Putri, I. R., Adhisuwignjo, S., & Rifa'i, M. (2018). *Photovoltaic Terapan: Photovoltaic Terapan, Adriani Parastiwi*. UPT Percetakan dan Penerbitan Polinema.
- Sakti, S. P. (2017). *Pengantar Teknologi Sensor: Prinsip Dasar Sensor Besaran Mekanik*.

Universitas Brawijaya Press.

Saputra, A. S. (2019). *Programmable Logic Control (PLC) & Arduino UNO: PLC & Arduino UNO*. Agil Surya Saputra.

Siahaan, V., & Sianipar, H. R. (2020). *Buku Pintar Pemrograman C# Untuk Pelajar dan Mahasiswa*. Balige Publishing.

Sommerville, I. (2016). *Software Engineering (Tenth Edition)* (10th ed.). Pearson Education Limited.

Stephens, R. (2012). Beginning Software Engineering. In *Beginning Software Engineering*. John Wiley & Sons, Inc. <https://doi.org/10.1002/9781119209515>

Stewart, B. (2015). *Adventures in Arduino*. John Wiley & Sons, Inc.

Suherlim, D., Lubis, L., & Permana, H. (2018). Korelasi kadar hemoglobin dengan saturasi oksigen pada guru besar Universitas Padjadjaran. *Bali Anatomy Journal*, 1(2), 26–29. <https://doi.org/10.36675/baj.v1i2.15>

Sukamto, R. A., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Informatika Bandung.

Sulianta, F. (2019). *Strategi Merancang Arsitektur Sistem Informasi Masa Kini*. Elex Media Komputindo.

Tsai, H. Y., Huang, K. C., Chang, H. C., Yeh, J. L. A., & Chang, C. H. (2014). A noncontact skin oxygen-saturation imaging system for measuring human tissue oxygen saturation. *IEEE Transactions on Instrumentation and Measurement*, 63(11), 2620–2631. <https://doi.org/10.1109/TIM.2014.2312512>

Wazlawick, R. S. (2014). Object-Oriented Analysis and Design for Information Systems: Modeling with UML, OCL, and IFML. In *Object-Oriented Analysis and Design for Information Systems: Modeling with UML, OCL, and IFML*. Elsevier Inc. <https://doi.org/10.1016/C2012-0-06942-6>

Yurindra. (2017). *Software Engineering*. Deepublish.

Datasheet Atmega328

Datasheet Arduino UNO

Datasheet LCD 16x2

<https://www.alodokter.com/oksigenasi-bisa-membantu-menyembuhkan-penyakit-dan-luka> diakses tanggal 21 September 2020

<https://www.alodokter.com/memahami-sistem-peredaran-darah-pada-manusia#:~:text=3.%20Darah,karbondioksida%2C%20untuk%20dikeluarkan%20dari%20tubuh.>

<https://www.alomedika.com/prinsip-kerja-pulse-oximetry-dan-keterbatasannya> diakses tanggal 21 September 2020.

<https://glosarium.org/arti-non-invasive/> diakses tanggal 21 September 2020.

[https://glosarium.org/?s=invasive&post\\_type=post](https://glosarium.org/?s=invasive&post_type=post) diakses tanggal 21 September 2020.

<https://lifestyle.kompas.com/read/2020/09/07/115535720/pulse-oximeter-bisakah-deteksi-covid-19?page=all> diakses 21 September 2020.

<https://components101.com/16x2-lcd-pinout-datasheet> diakses 3 November 2020.

<https://datasheets.maximintegrated.com/en/ds/MAX30100.pdf> diakses 3 November 2020.

<https://components101.com/microcontrollers/arduino-uno> diakses tanggal 4 November 2020.

<http://www.jogjarobotika.com/sensor-lainnya/2054-max30100-pulse-oximeter-heart-rate-sensor-module.html> diakses 4 November 2020.

<http://www.galerimedika.com/Rekomendasi-dan-Perbandingan-Fingertip-Pulse-Oximeter> diakses 4 November 2020.

## LAMPIRAN

### Listing Program MAX30100

```
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

#define REPORTING_PERIOD_MS 1000
PulseOximeter pox;
int LED = 13;
uint32_t tsLastReport = 0;

void onBeatDetected()
{
    digitalWrite(LED,!digitalRead(LED));
}

void setup()
{
    lcd.init();
    lcd.backlight();

    Serial.begin(115200);
    Serial.print("Initializing pulse oximeter..");
    lcd.begin(16,2);
    lcd.print("Initializing...");
    delay(3000);
    lcd.clear();

    if (!pox.begin()) {
        Serial.println("FAILED");
        for(;;);
    } else {
```

```

    Serial.println("SUCCESS");}
    pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
    pox.setOnBeatDetectedCallback(onBeatDetected);
}

void loop()
{
    pox.update();
    if (millis() - tsLastReport > REPORTING_PERIOD_MS) {
        Serial.print("");
        Serial.print(pox.getSpO2());
        Serial.println("%");

        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("SpO2: ");
        lcd.print(pox.getSpO2());
        lcd.print("%");

        if (pox.getSpO2()<100 && pox.getSpO2()>=95)
        {
            lcd.setCursor(0,1);
            lcd.print("Normal");
        }
        else if(pox.getSpO2()<=94 && pox.getSpO2()>=91)
        {
            lcd.setCursor(0,1);
            lcd.print("Hipoksemia Ringan");
        }
        else if(pox.getSpO2()<=90 && pox.getSpO2()>=86)
        {
            lcd.setCursor(0,1);
            lcd.print("Hipoksemia Sedang");
        }
    }
}

```



```
else if(pox.getSpO2()<=85 && pox.getSpO2())>=10)
{
    lcd.setCursor(0,1);
    lcd.print("Hipoksemia Berat");
} else
{
    lcd.setCursor(0,1);
    lcd.print("Tempelkan Jari");
}
tsLastReport = millis();
}
}
```

## Datasheet Arduino UNO R3

# Arduino UNO







### Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328P ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

### Index

<b>Technical Specifications</b>	<b>Page 2</b>
<b>How to use Arduino</b> <small>Programming Environment, Basic Tutorial</small>	<b>Page 6</b>
<b>Terms &amp; Conditions</b>	<b>Page 7</b>
<b>Environmental Policies</b> <small>half a ton of green via Impatto Zero®</small>	<b>Page 7</b>

# Technical Specification

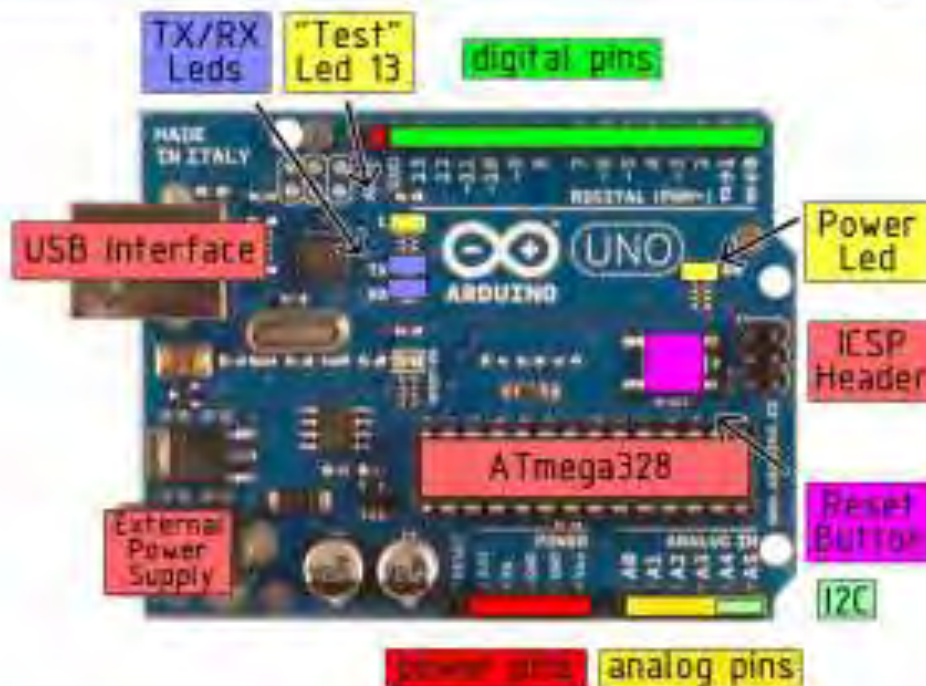


EAGLE files: <http://www.joshschneiders.com/arduino.com> Schematic: <http://www.arduino.cc/en/Reference/ArduinoProMini>

## Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

## the board



## Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts, if supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

## Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0.5 KB is used for the bootloader). It has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

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

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.



radiospares

RADIONICS





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

- **I<sup>2</sup>C: 4 (SDA) and 5 (SCL).** Support I<sup>2</sup>C (TWI) communication using the [Wire library](#).

These are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button (a shield which block the one on the board).

See also the [mapping between Arduino pins and ATmega328 pins](#).

## Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an \*.inf file is required.

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I<sup>2</sup>C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I<sup>2</sup>C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega328 datasheet.

## Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [library](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference: C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available. The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ICSP header with an external programmer (overwriting the DFU bootloader).

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this Avram Iliadis](#) for details.

## USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection: if more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

## Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



## How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](#) for the latest instructions: - <http://arduino.cc/en/Guide/HomePage>

### Linux Install

### Windows Install

### Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

### Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>  
Arduino-0017>Examples>  
Digital>Blink**

Once you have your sketch you'll see something very close to the screenshot on the right.

In **Tools>Board** select

Now you have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.



Done compiling

Press Compile button  
to mark for errors



Upload



TX RX Flashing



Blinking LED







## Terms & Conditions



### 1. Warranty

1.1 The producer warrants that its products will conform to the specifications. This warranty lasts for one (1) year from the date of the use. The producer shall not be liable for any defects that are caused by neglect, misuse or misstatement by the Customer, including improper installation or testing of it. Any products that have been altered or modified in any way by a customer, disqualify. The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the product needs to be replaced, the producer shall issue a replacement free of charge. Repaired products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

1.4 Customer agrees that prior to using any systems that include the producer's products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical assistance to design advice, quality characteristics, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional warranties or liability shall arise from the producer providing such services.

1.5 The Arduino<sup>®</sup> products are not authorized for use in safety-critical applications where failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support services and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino<sup>®</sup> products are further designed not intended for use in military or aerospace applications in environments and for automotive applications or environments. Customer acknowledges and agrees that any such use of Arduino<sup>®</sup> products which is safety critical to the Customer's work, and that Customer is solely responsible for compliance with all legal, regulatory and safety-related requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino<sup>®</sup> products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

### 2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third party losses, damages, penalties and expenses it incurs in the event directly caused by: (i) an actual breach by a Customer of the representations and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

### 3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer at any time for any special, incidental, direct, punitive, incidental, consequential or exemplary damages or compensation with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

### 4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the existence or timeliness of any features or instructions marked "reserved" or "underlined." The producer reserves these for future initiatives and shall have its responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product architecture on the Web site of Arduino is subject to change without notice. Do not freeze a design with the architecture.



## Environmental Policies



The producer of Arduino<sup>®</sup> has joined the Impatto Zero<sup>®</sup> policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.

## Datasheet Sensor MAX30100

EVALUATION KIT AVAILABLE

### MAX30100

### Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health

#### General Description

The MAX30100 is an integrated pulse oximetry and heart-rate monitor sensor solution. It combines two LEDs, a photodiode, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals.

The MAX30100 operates from 1.8V and 3.0V power supplies and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times.

#### Applications

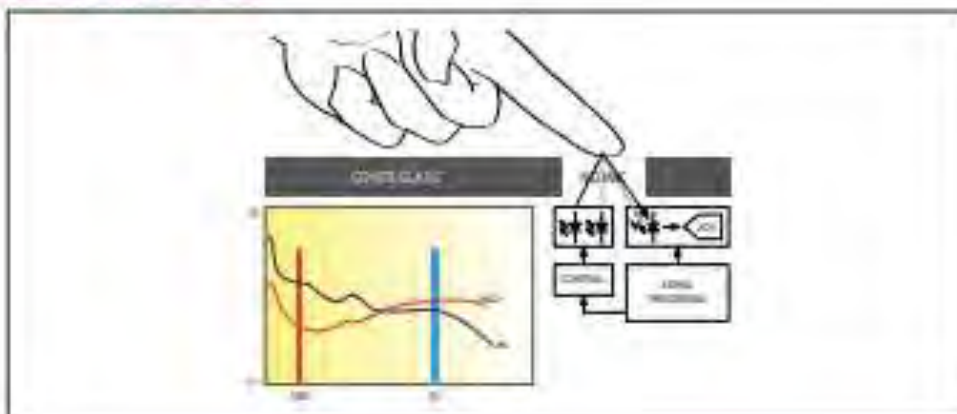
- Wearable Devices
- Fitness Assistant Devices
- Medical Monitoring Devices

#### Benefits and Features

- Complete Pulse Oximeter and Heart-Rate Sensor Solution Simplifies Design
  - Integrated LEDs, Photo Sensor, and High-Performance Analog Front-End
  - Tiny 5.8mm x 2.8mm x 1.2mm 14-Pin Optically-Enhanced System-in-Package
- Ultra-Low-Power Operation Increases Battery Life for Wearable Devices
  - Programmable Sample Rate and LED Current for Power Savings
  - Ultra-Low Shutdown Current (0.7µA, typ)
- Advanced Functionality Improves Measurement Performance
  - High SNR Provides Robust Motion Artifact Resistance
  - Integrated Ambient Light Cancellation
  - High Sample Rate Capability
  - Fast Data Output Capability

Ordering Information appears at end of data sheet.

#### System Block Diagram



01-0282 Rev. 0, 1/11

## MAX30100

Pulse Oximeter and Heart-Rate Sensor IC  
for Wearable Health

## Absolute Maximum Ratings

$V_{DD}$ to GND	-0.3V to +2.2V	Continuous Power Dissipation ( $T_A = +70^\circ\text{C}$ )	
GND to PGND	-0.3V to +0.3V	QESIP (above $5.6\text{mW}/^\circ\text{C}$ above $+70^\circ\text{C}$ )	40mW
$x\_DRV$ to LED+ to PGND	-0.3V to +0.5V	Operating Temperature Range	$-40^\circ\text{C}$ to $+85^\circ\text{C}$
All Other Pins to GND	-0.3V to +0.5V	Soldering Temperature (reflow)	$+260^\circ\text{C}$
Output Short-Circuit Current Duration	Continuous	Storage Temperature Range	$-40^\circ\text{C}$ to $+125^\circ\text{C}$
Continuous Input Current into Any Terminal	$\pm 20\text{mA}$		

## Package Thermal Characteristics (Note 1)

QESIP	
Junction-to-Ambient Thermal Resistance ( $R_{JA}$ )	100°C/W
Junction-to-Case Thermal Resistance ( $R_{JC}$ )	17°C/W

Note 1: Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to [www.maximintegrated.com/thermal-thermal](http://www.maximintegrated.com/thermal-thermal).

## Electrical Characteristics

( $V_{DD} = 1.4\text{V}$ ,  $V_{IO\_LED+} = V_{IO\_LED-} = 3.3\text{V}$ ,  $T_A = +25^\circ\text{C}$ , unless otherwise specified, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
<b>POWER SUPPLY</b>						
Power-Supply Voltage	$V_{DD}$	Guaranteed by RED and IR count tolerance	1.7	1.8	2.0	V
LED Supply Voltage ( $IO\_LED+$ to $IO\_LED-$ to PGND)	$V_{IO\_LED}$	Guaranteed by POR of LED Driver	2.7	3.3	3.0	V
Supply Current	$I_{DD}$	SpO <sub>2</sub> and heart rate modes, PW = 200 $\mu\text{s}$ , 50kps		800	1200	$\mu\text{A}$
		Heart rate only mode, PW = 200 $\mu\text{s}$ , 50kps		300	1200	
Supply Current in Shutdown	$I_{SHDN}$	$T_A = +25^\circ\text{C}$ , MODE = 0x00		0.7	10	$\mu\text{A}$
<b>SENSOR CHARACTERISTICS</b>						
ADC Resolution				14		bits
Red ADC Count (Note 3)	RED <sub>1</sub>	Proprietary ATE setup RED_PA = 0x05, LED_PW = 0x01, SPO2_SR = 0x07, $T_A = +25^\circ\text{C}$	23,000	26,000	29,000	Counts
IR ADC Count (Note 3)	IR <sub>1</sub>	Proprietary ATE setup IR_PA = 0x00, LED_PW = 0x01, SPO2_SR = 0x07, $T_A = +25^\circ\text{C}$	23,000	26,000	29,000	Counts
Dark Current Count	DC <sub>1</sub>	RED_PA = IR_PA = 0x00, LED_PW = 0x01, SPO2_SR = 0x01		0	1	Counts
DC Ambient Light Rejection (Note 4)	ALR	Number of ADC counts with light on sensor under direct sunlight (100k lux) LED_PW = 0x01, SPO2_SR = 0x01	RED LED		0	Counts
			IR LED		0	

MAX30100

Pulse Oximeter and Heart-Rate Sensor IC  
for Wearable Health

## Electrical Characteristics (continued)

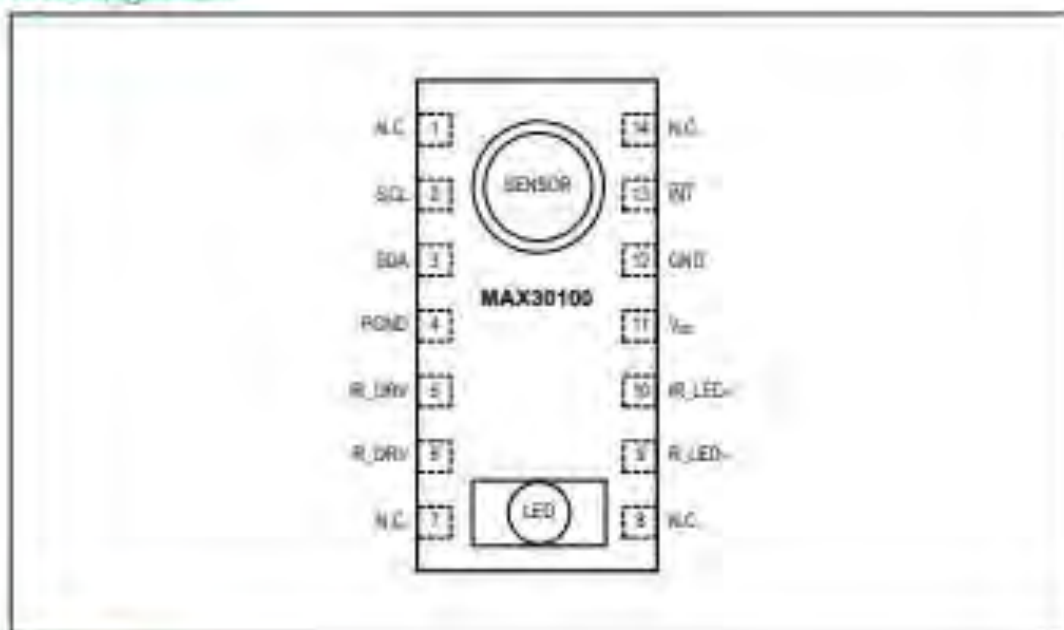
VDD = 1.8V, VIO\_LED = VIO\_LEDV = 3.3V, TA = +25°C, from package from TA = -40°C to +85°C, unless otherwise noted. (Note 7)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
IR ADC Count—PSRR (VDD)	PSRR <sub>VDD</sub>	Properly ATE setup 1.7V < V <sub>DD</sub> < 2.0V LED_PW = 0x03, SPO2_SIR = 0x03 IR_PA = 0x00, IR_RA = 0x03, TA = +25°C Frequency = DC to 100Hz, 100mV <sub>rms</sub>		0.25	2	%
RED/IR ADC Count—PSRR (X_LED)	PSRR <sub>LED</sub>	Properly ATE setup $\Delta V / V = K_{LED} \times 5V$ LED_PW = 0x03, SPO2_SIR = 0x03 IR_PA = 0x00, IR_RA = 0x03, TA = +25°C Frequency = DC to 100Hz, 100mV <sub>rms</sub>		0.25	2	%
ADC Integration Time	INT	LED_PW = 0x00		200		µs
		LED_PW = 0x03		1000		µs
<b>IR LED CHARACTERISTICS (Note 4)</b>						
LED Peak Wavelength	$\lambda_p$	I <sub>LED</sub> = 20mA, TA = +25°C	870	880	900	nm
Fall Width at half Max	$\Delta f$	I <sub>LED</sub> = 20mA, TA = +25°C		30		ns
Forward Voltage	V <sub>F</sub>	I <sub>LED</sub> = 20mA, TA = +25°C		1.4		V
Radiant Power	P <sub>O</sub>	I <sub>LED</sub> = 20mA, TA = +25°C		0.3		mW
<b>RED LED CHARACTERISTICS (Note 4)</b>						
LED Peak Wavelength	$\lambda_p$	I <sub>LED</sub> = 20mA, TA = +25°C	630	640	670	nm
Fall Width at half Max	$\Delta f$	I <sub>LED</sub> = 20mA, TA = +25°C		30		ns
Forward Voltage	V <sub>F</sub>	I <sub>LED</sub> = 20mA, TA = +25°C		2.1		V
Radiant Power	P <sub>O</sub>	I <sub>LED</sub> = 20mA, TA = +25°C		0.8		mW
<b>TEMPERATURE SENSOR</b>						
Temperature ADC Acquisition Time	T <sub>1</sub>	TA = +25°C		25		ms
Temperature Sensor Accuracy	T <sub>A</sub>	TA = +25°C		±1		°C
Temperature Sensor Minimum Range	T <sub>MIN</sub>			-40		°C
Temperature Sensor Maximum Range	T <sub>MAX</sub>			55		°C

## MAX30100

Pulse Oximeter and Heart-Rate Sensor IC  
for Wearable Health

## Pin Configuration



## Pin Description

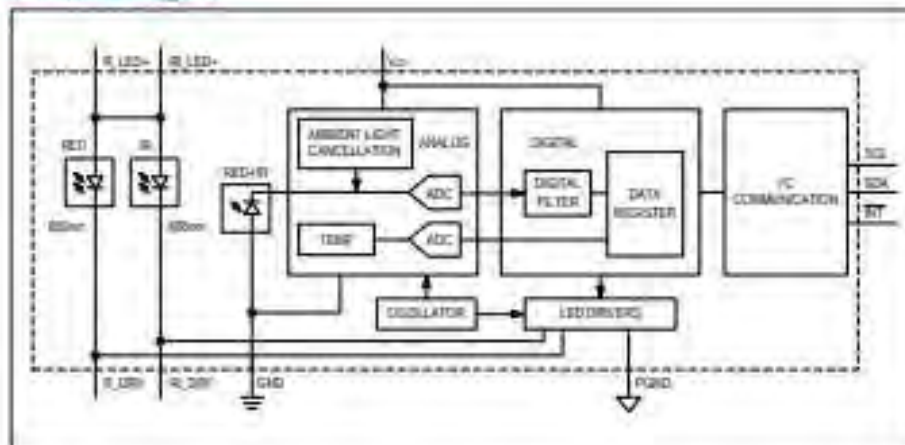
PIN	NAME	FUNCTION
1, 7, & 14	N.C.	No Connection. Connect to PCB Pad for Mechanical Stability.
2	SCL	100 kHz Clock Input
3	SDA	100 kHz Clock Data, Bidirectional (Open-Drain)
4	PGND	Power Ground of the LED Driver Stacks
5	R_DRV-	IR LED Cathode and LED Driver Connection Point. Leave floating in circuit.
6	R_DRV+	Red LED Cathode and LED Driver Connection Point. Leave floating in circuit.
9	R_LED+	Power Supply (Anode Connection) for Red LED. Bypass to PGND for best performance. Connected to R_LED+ internally.
10	IR_LED+	Power Supply (Anode Connection) for IR LED. Bypass to PGND for best performance. Connected to R_LED+ internally.
11	VDD	Operating Power Supply Input. Bypass to GND for best performance.
12	GND	Operating Ground
13	INT	Active-Low Interrupt (Open-Drain)



## MAX30100

Pulse Oximeter and Heart-Rate Sensor IC  
for Wearable Health

## Functional Diagram



## Detailed Description

The MAX30100 is a complete pulse oximetry and heart-rate sensor system solution designed for the demanding requirements of wearable devices. The MAX30100 provides very small total solution size without sacrificing optical or electrical performance. Minimal external hardware components are needed for integration into a wearable device.

The MAX30100 is fully configurable through software registers, and the digital output data is stored in a 16-deep FIFO within the device. The FIFO allows the MAX30100 to be connected to a microcontroller or microprocessor on a shared bus, where the data is not being read continuously from the device's registers.

**SpO<sub>2</sub> Subsystem**

The SpO<sub>2</sub> subsystem in the MAX30100 is composed of ambient light cancellation (ALC), 18-bit sigma delta ADC, and proprietary discrete time filter.

The SpO<sub>2</sub> ADC is a continuous time oversampling sigma delta converter with up to 18-bit resolution. The ADC full-pulse data rate can be programmed from 50Hz to 1kHz. The

MAX30100 includes a proprietary discrete time filter to reject 50Hz/60Hz interference and low-frequency residual ambient noise.

**Temperature Sensor**

The MAX30100 has an on-chip temperature sensor for (optionally) calibrating the temperature dependence of the SpO<sub>2</sub> subsystem.

The SpO<sub>2</sub> algorithm is relatively insensitive to the wavelength of the IR LED, but the red LED's wavelength is critical to correct interpretation of the data. The temperature sensor data can be used to compensate the SpO<sub>2</sub> (and with ambient temperature changes).

**LED Driver**

The MAX30100 integrates red and IR LED drivers to drive LED pulses for SpO<sub>2</sub> and HR measurements. The LED current can be programmed from 5mA to 50mA (typical only) with proper supply voltage. The LED pulse width can be programmed from 200µs to 5ms to optimize measurement accuracy and power consumption based on use cases.

## Datasheet LCD 16x2

# Datasheet I2C 1602 Serial LCD Module



### Product features:

The I2C 1602 LCD module is a 2 line by 16 character display interfaced to an I2C daughter board. The I2C interface only requires 2 data connections, +5 VDC and GND to operate

For in depth information on I2C interface and history, visit: <http://www.wikipedia/wiki/i2c>

### Specifications:

I2C Address Range	2 lines by 16 character 0x20 to 0x2F (Default=0x27, addressable)
Operating Voltage	5 Vdc
Backlight	White
Contrast	Adjustable by potentiometer on I2C
Size	interface 80mm x 36mm x 20 mm 68mm x
Viewable area	16mm

### Power:

The device is powered by a single 5Vdc connection.

### Pinout Diagram:



### Pin/Control Descriptions:

Pin #	Name	Type	Description
1	GND	Power	Supply & Logic ground
2	VCC	Power	Digital I/O or RX (serial receive)
3	SDA	I/O	Serial Data line
4	SCL	CLK	Serial Clock line
A0	A0	Jumper	Optional address selector A0 - see below
A1	A1	Jumper	Optional address selector A1 - see below
A2	A2	Jumper	Optional address selector A2 - see below
Backlight		Jumper	Jumpered - enable backlight, Open - disable backlight
Contrast		Pot	Adjust for best viewing